
Amazon Elastic Compute Cloud

Windows Instances용 사용 설명서



Amazon Elastic Compute Cloud: Windows Instances용 사용 설명서

Copyright © 2020 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Amazon EC2이란 무엇입니까?	1
Amazon EC2의 기능	1
Amazon EC2 시작 방법	1
관련 서비스	2
Amazon EC2에 액세스	3
Amazon EC2 가격	3
PCI DSS 준수	4
기본 인프라	4
Amazon 머신 이미지와 인스턴스	4
리전 및 영역	5
스토리지	6
루트 디바이스 볼륨	7
네트워킹 및 보안	8
AWS Identity and Access Management	8
Windows Server와 Amazon EC2 Windows 인스턴스의 차이점	9
Amazon EC2 Windows 인스턴스 구동을 위한 애플리케이션 개발	10
설정	12
AWS에 가입	12
키 페어 생성	12
보안 그룹 생성	13
시작하기 자습서	16
개요	16
사전 조건	17
1단계: 인스턴스 시작	17
2단계: 인스턴스에 연결	18
3단계: 인스턴스 정리	19
다음 단계	20
모범 사례	21
Amazon 머신 이미지	24
고유 AMI 생성	24
AMI 구입, 공유 및 판매	24
AMI 등록 해제	24
AWS Windows AMI	24
초기 Windows AMI 선택	25
AMI를 최신으로 유지하기	25
가상화 유형	25
AWS Windows AMI	26
Windows 인스턴스 업데이트	27
Windows Server 새 버전으로 업그레이드 또는 마이그레이션	27
Windows AMI 알림에 대한 구독	27
AWS Windows AMI의 구성 변경	28
AWS Windows AMI 버전에 관한 세부 정보	30
Windows Server 2016 이상 AMI의 변경 사항	63
Windows Server 2016 인스턴스에서 Docker 컨테이너 총들	64
Hibernate Agent(2018.03.16 AMI)와 관련된 문제	64
STIG 규정을 준수하는 AMI	65
Windows AMI 찾기	68
Amazon EC2 콘솔을 사용하여 Windows AMI 찾기	68
Windows PowerShell용 AWS 도구를 사용하여 AMI 찾기	69
AWS CLI를 사용하여 AMI 찾기	69
시스템 관리자를 사용하여 최신 Amazon Linux AMI 찾기	69
시스템 관리자 파라미터를 사용하여 AMI 찾기	70
공유 AMI	72
공유 AMI 검색	73

퍼블릭 AMI 설정	75
지정한 AWS 계정과 AMI 공유	76
북마크 사용	79
공유 Windows AMI 지침	79
유료 AMI	80
AMI 판매	80
유료 AMI 찾기	81
유료 AMI 구입	82
인스턴스에 대한 제품 코드 가져오기	82
유료 지원 사용	83
유료 및 지원된 AMI에 대한 청구서	83
AWS Marketplace 구독 관리	83
사용자 지정 Windows AMI 생성	84
작동 방식	84
실행 중인 인스턴스에서 AMI 생성	85
Sysprep을 이용한 AMI 생성	87
EBS 지원 AMI를 통한 암호화 사용	98
인스턴스 시작 시나리오	98
이미지 복사 시나리오	101
AMI 복사	103
인스턴스 스토어 지원 AMI 복사 권한	104
교차 리전 복사	104
교차 계정 복사	105
암호화 및 복사	105
AMI 복사	107
대기 중인 AMI 복사 작업 종지	108
결제 정보 가져오기	108
AMI 결제 정보 필드	109
플랫폼 세부 정보 및 사용 작업 값	109
플랫폼 세부 정보 및 사용 작업 값 보기	110
청구서에서 결제 정보 확인	111
Windows AMI 등록 취소	111
인스턴스	114
인스턴스 유형	114
사용 가능한 인스턴스 유형	115
하드웨어 사양	118
Nitro 시스템에 구축된 인스턴스	118
네트워킹 및 스토리지 기능	119
인스턴스 제한	122
법용	122
컴퓨팅 최적화	163
메모리 최적화	168
스토리지 최적화	177
액셀러레이티드 컴퓨팅	182
인스턴스 유형 찾기	193
인스턴스 유형 변경	194
권장 사항 가져오기	198
인스턴스 구입 옵션	200
인스턴스 수명 주기 결정	201
온디맨드 인스턴스	202
예약 인스턴스	205
예약된 인스턴스	236
스팟 인스턴스	239
전용 호스트	320
전용 인스턴스	349
온디맨드 용량 예약	354
인스턴스 수명 주기	371

인스턴스 시작	372
인스턴스 중지 및 시작(Amazon EBS 지원 인스턴스에만 해당)	372
인스턴스 최대 절전 모드(Amazon EBS 지원 인스턴스에만 해당)	373
인스턴스 재부팅	373
인스턴스 만료	373
인스턴스 종료	374
재부팅, 중지, 최대 절전 모드 및 종료의 차이	374
시작	375
연결	436
중지 및 시작	440
최대 절전 모드	443
재부팅	452
만료	452
Terminate	454
복구	460
인스턴스 구성	461
EC2Launch v2	461
EC2Launch	489
EC2Config 서비스	496
PV 드라이버	523
AWS NVMe 드라이버	539
CPU 옵션 최적화	541
시간 설정	556
암호 설정	560
Windows 구성 요소 추가	560
보조 프라이빗 IPv4 주소 구성	564
시작 시 명령 실행	568
인스턴스 메타데이터 및 사용자 데이터	576
EC2의 SQL Server 클러스터링	605
Windows 인스턴스 업그레이드	611
인 플레이스 업그레이드 수행	612
자동 업그레이드 수행	616
최신 세대 인스턴스 유형으로 마이그레이션	622
Microsoft SQL Server를 Windows에서 Linux로 마이그레이션	627
업그레이드 문제 해결	634
인스턴스 식별	634
인스턴스 자격 증명 문서 검사	634
시스템 UUID 검사	634
Elastic Graphics	636
Elastic Graphics 기본 사항	636
Elastic Graphics 가격	638
Elastic Graphics 제한 사항	638
Elastic Graphics 작업	638
보안 그룹 구성	639
Elastic Graphics 액셀러레이터로 인스턴스 시작	639
Elastic Graphics용 필요 소프트웨어 설치	640
인스턴스에서 Elastic Graphics 기능 확인	640
Elastic Graphics 정보 보기	642
피드백 제출	643
CloudWatch 지표를 사용하여 Elastic Graphics 모니터링	643
Elastic Graphics 지표	643
Elastic Graphics 차원	644
Elastic Graphics에 대한 CloudWatch 지표 보기	644
Elastic Graphics 모니터링을 위한 CloudWatch 경보 생성	644
문제 해결	645
애플리케이션 성능 문제 조사	645
비정상 상태 문제 해결	646

모니터링	648
자동 및 수동 모니터링	649
자동 모니터링 도구	649
수동 모니터링 도구	650
모니터링 모범 사례	650
인스턴스 상태 모니터링	651
인스턴스 상태 확인	651
예약된 이벤트	655
CloudWatch를 사용해 인스턴스 모니터링	667
세부 모니터링 활성화	667
사용 가능한 지표 나열	669
지표 통계 가져오기	679
그래프 지표	687
경보 만들기	687
인스턴스를 중지, 종료, 재부팅 또는 복구하는 경보 만들기	688
CloudWatch 이벤트를 사용한 Amazon EC2 자동화	697
AWS CloudTrail을 사용하여 API 호출 로깅	697
CloudTrail의 Amazon EC2 및 Amazon EBS 정보	698
Amazon EC2 및 Amazon EBS 로그 파일 항목 이해	698
EC2 Instance Connect를 통해 연결하는 사용자 감사	699
.NET 및 SQL Server 애플리케이션 모니터링	700
네트워킹	702
인스턴스 IP 주소 지정	702
프라이빗 IPv4 주소 및 내부 DNS 호스트 이름	702
퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름	703
탄력적 IP 주소(IPv4)	704
Amazon DNS 서버	704
IPv6 주소	704
인스턴스에 대한 IP 주소 작업	705
다중 IP 주소	709
고유 IP 주소 가져오기	716
요구 사항	716
AWS 계정으로 주소 범위를 가져오도록 준비하기	717
AWS에서 사용할 수 있도록 주소 범위 프로비저닝	718
AWS를 통해 주소 범위 알리기	719
주소 범위 관련 작업	720
주소 범위 프로비저닝 취소	721
탄력적 IP 주소	721
탄력적 IP 주소 기본 사항	721
탄력적 IP 주소 작업	722
이메일 애플리케이션에 역방향 DNS 사용	728
탄력적 IP 주소 제한	728
네트워크 인터페이스	729
네트워크 인터페이스 기본 사항	729
인스턴스 유형별 네트워크 인터페이스당 IP 주소	730
네트워크 인터페이스 시나리오	739
네트워크 인터페이스 구성 모범 사례	741
네트워크 인터페이스 관련 작업	741
요청자 관리 네트워크 인터페이스	750
향상된 네트워킹	751
향상된 네트워킹 유형	751
인스턴스에서 향상된 네트워킹 기능 활성화	751
향상된 네트워킹: ENA	751
향상된 네트워킹: Intel 82599 VF	758
배치 그룹	761
클러스터 배치 그룹	762
파티션 배치 그룹	763

분산형 배치 그룹	764
배치 그룹 규칙 및 제한 사항	764
배치 그룹 생성	765
배치 그룹 태그 지정	766
배치 그룹으로 인스턴스 시작	768
배치 그룹의 인스턴스 설명	769
인스턴스의 배치 그룹 변경	770
배치 그룹 삭제	771
네트워크 MTU	772
점보 프레임(9001 MTU)	773
경로 MTU 검색	773
두 호스트 간 경로 MTU 확인	774
Windows 인스턴스에서 MTU 확인 및 설정	774
문제 해결	776
가상 사설 클라우드	776
Amazon VPC 설명서	776
포트 및 프로토콜	777
AllJoyn 라우터	777
디바이스로 캐스팅	778
코어 네트워킹	780
배달 최적화	793
진단 추적	794
DIAL 프로토콜 서버	794
DFS(분산 파일 시스템) 관리	794
파일 및 프린터 공유	795
파일 서버 원격 관리	798
ICMP v4 모두	798
멀티캐스트	798
원격 데스크톱	799
Windows 장치 관리	801
Windows 방화벽 원격 관리	801
Windows 원격 관리	802
EC2-Classic	802
지원되는 플랫폼 감지	802
EC2-Classic에서 사용 가능한 인스턴스 유형	804
EC2-Classic 인스턴스와 VPC 간 차이점	804
EC2-Classic과 VPC 간 리소스 공유 및 액세스	808
ClassicLink	809
EC2-Classic에서 VPC로 마이그레이션	820
보안	830
인프라 보안	830
네트워크 격리	831
물리적 호스트에서 격리	831
네트워크 트래픽 제어	831
인터넷 VPC 엔드포인트	832
인터넷 VPC 엔드포인트 생성	833
인터넷 VPC 엔드포인트 정책 생성	833
복원성	834
데이터 보호	834
저장된 암호화	835
전송 중 데이터 암호화	835
ID 및 액세스 관리	836
인스턴스에 대한 네트워크 액세스	836
Amazon EC2 권한 속성	836
IAM 및 Amazon EC2	836
IAM 정책	838
IAM 역할	890

	네트워크 액세스	898
	키 페어	900
	키 페어 생성 또는 가져오기	901
	키 페어 태그 지정	904
	키 페어에 맞는 퍼블릭 키 검색	905
	인스턴스 메타데이터를 통해 키 페어의 퍼블릭 키 검색	905
	시작 시 지정된 키 페어 식별	906
	(선택 사항) 키 페어의 지문 확인	906
	프라이빗 키를 분실한 경우 Windows 인스턴스에 연결하는 방법	907
	키 페어 삭제	907
	보안 그룹	908
	보안 그룹 규칙	909
	기본 보안 그룹	911
	사용자 지정 보안 그룹	911
	보안 그룹 작업	912
	보안 그룹 규칙 참조	919
	구성 관리	924
	업데이트 관리	925
	변경 관리	925
	규정 준수 확인	925
	감사 및 책임	926
	스토리지	927
	Amazon EBS	928
	Amazon EBS의 기능	929
	EBS 볼륨	929
	EBS 스냅샷	966
	EBS 데이터 서비스	1023
	EBS 볼륨 및 NVMe	1048
	EBS 최적화	1049
	EBS 성능	1063
	EBS CloudWatch 지표	1076
	EBS CloudWatch 이벤트	1081
	인스턴스 스토어	1091
	인스턴스 스토어 수명	1092
	인스턴스 스토어 볼륨	1093
	인스턴스 스토어 볼륨 추가	1098
	SSD 인스턴스 스토어 볼륨	1101
	파일 스토리지	1102
	Amazon S3	1102
	Amazon EFS	1104
	Amazon FSx	1104
	인스턴스 볼륨 제한	1104
	Nitro 시스템 볼륨 제한	1105
	Windows 볼륨 제한	1105
	대역폭 및 용량 비교	1105
	디바이스 명명	1106
	사용 가능한 디바이스 이름	1106
	디바이스 이름 고려 사항	1107
	블록 디바이스 매핑	1107
	블록 디바이스 매핑의 개념	1107
	AMI 블록 디바이스 매핑	1110
	인스턴스 블록 디바이스 매핑	1112
	볼륨에 디스크 매핑	1116
	Windows 디스크 관리를 이용하여 디스크 나열	1116
	Windows PowerShell을 사용하여 디스크 나열(Windows Server 2012 이상)	1118
	디스크 디바이스와 디스크 이름 간 매핑	1121
	저장소 공간 다이렉트 배포	1122

1단계: 시작 및 도메인 병합 인스턴스	1125
2단계: 인스턴스 사전 조건 설치 및 구성	1127
3단계: 장애 조치 클러스터 생성	1128
4단계: S2D 활성화	1129
5단계: 스토리지 프로비저닝	1129
6단계: S2D 리소스 검토	1130
7단계: 정리	1131
추가 리소스	1131
리소스 및 태그	1132
리소스 위치	1132
리소스 ID	1133
더 긴 ID 작업	1134
긴 ID 설정에 대한 액세스 제어	1137
리소스 목록화 및 필터링	1138
고급 검색	1138
콘솔을 이용하여 리소스 목록화	1139
콘솔을 이용하여 리소스 필터링	1140
CLI 및 API를 이용하여 목록화 및 필터링	1141
리소스에 태그 지정	1143
태그 기본 사용	1143
리소스에 태그 지정	1144
태그 제한	1147
리소스에 결제용 태그 지정	1147
콘솔을 사용한 태그 작업	1148
명령줄을 사용한 태그 작업	1151
CloudFormation을 사용하여 리소스에 태그 추가	1154
서비스 할당량	1154
현재 제한 조회	1154
증가 요청	1156
포트 25를 사용하여 전송되는 이메일 관련 제한	1156
사용 보고서	1156
자습서	1157
자습서: WordPress 블로그 배포	1157
사전 조건	1157
Microsoft 웹 플랫폼 설치 관리자 설치	1158
WordPress 설치	1158
보안 키 구성	1159
사이트 제목 및 관리자 구성	1160
WordPress 사이트를 퍼블릭으로 만들기	1160
다음 단계	1161
자습서: WAMP 서버 설치	1161
자습서: WIMP 서버 설치	1163
사전 조건	1164
인스턴스 준비	1164
IIS 웹 서버 설치	1165
MySQL 및 PHP 설치	1166
서버 테스트	1166
자습서: 애플리케이션의 가용성 향상	1167
사전 조건	1168
애플리케이션 확장 및 로드 밸런싱	1168
로드 밸런서 테스트	1170
자습서: Windows HPC 클러스터 설치	1170
사전 조건	1171
1단계: 보안 그룹 만들기	1171
2단계: Active Directory 도메인 컨트롤러 설치	1174
3단계: 헤드 노드 구성	1174
4단계: 컴퓨팅 노드 설치	1176

5단계: HPC 컴퓨팅 노드 용량 조정(선택 사항)	1177
문제 해결	1178
시작 문제 해결	1178
인스턴스 제한 초과됨	1178
부족한 인스턴스 용량	1179
인스턴스 즉시 종료	1179
Windows 시작 직후 높은 CPU 사용률	1180
인스턴스에 연결	1181
원격 데스크톱으로 원격 컴퓨터에 연결할 수 없음	1181
macOS RDP 클라이언트 사용 중 오류 발생	1184
RDP에 바탕 화면이 아닌 빈 화면 표시	1184
관리자가 아닌 사용자 계정으로 인스턴스에 원격 로그온할 수 없음	1184
AWS 시스템 관리자를 사용하여 원격 데스크톱 연결 문제 해결	1184
원격 레지스트리를 사용하여 EC2 인스턴스에서 원격 데스크톱 활성화	1187
연결할 수 없는 인스턴스 문제 해결	1188
연결할 수 없는 인스턴스의 스크린샷을 얻는 방법	1188
공통적인 스크린샷	1189
기억나지 않거나 만료된 Windows 관리자 암호 재설정	1196
EC2Config를 사용하여 재설정	1197
EC2Launch를 사용하여 재설정	1201
인스턴스 종지	1204
대체 인스턴스 생성	1204
인스턴스 종료	1205
지연된 인스턴스 종료	1205
종료된 인스턴스가 계속 표시됨	1206
인스턴스가 자동으로 시작되거나 종료됨	1206
EC2Rescue for Windows Server	1206
GUI 사용	1206
명령줄 사용	1210
시스템 관리자 사용	1215
진단 인터럽트 보내기	1217
지원되는 인스턴스 유형	1218
사전 조건	1218
진단 인터럽트 보내기	1218
일반적인 문제	1218
EBS 볼륨이 Windows Server 2016 이상에서 초기화를 수행하지 않음	1219
DSRM(Directory Services Restore Mode)로 EC2 Windows 인스턴스 부팅	1219
인스턴스의 네트워크 연결이 끊어지거나 예약된 작업이 예정 시간에 실행되지 않음	1221
콘솔 출력을 가져올 수 없음	1221
네트워크에서 Windows Server 2012 R2를 이용할 수 없는 경우	1222
일반적인 메시지	1222
>Password is not available"	1222
>Password not available yet"	1223
"Cannot retrieve Windows password"	1223
"Waiting for the metadata service"	1223
"Unable to activate Windows"	1226
"Windows is not genuine (0x80070005)"	1227
"No Terminal Server License Servers available to provide a license"	1227
"Some settings are managed by your organization.(일부 설정이 사용자의 조직에 의해 관리됩니다.)"(Windows Server 2019)	1228
Microsoft System Center DPM용 AWS 시스템 관리자	1229
기능	1229
제한 사항	1229
요구 사항	1230
시작하기	1230
설정	1230
AWS에 가입	1230

사용자에 대한 액세스 설정	1230
추가 기능 배포	1233
AWS 자격 증명 제공	1233
EC2 인스턴스 관리	1234
EC2 인스턴스 생성	1234
인스턴스 보기	1237
인스턴스에 연결	1237
인스턴스 재부팅	1238
인스턴스 중지	1238
인스턴스 시작	1238
인스턴스 종료	1238
VM 가져오기	1239
사전 조건	1239
가상 머신 가져오기	1239
가져오기 작업 상태 확인	1240
가져온 인스턴스 백업	1241
문제 해결	1241
오류: 추가 기능을 설치할 수 없음	1241
설치 오류	1241
로그 파일 확인	1242
VM 가져오기 오류	1242
추가 기능 제거	1242
AWS Management Pack	1244
AWS Management Pack for System Center 2012 개요	1244
AWS Management Pack for System Center 2007 R2 개요	1246
다운로드	1247
System Center 2012	1247
System Center 2007 R2	1248
배포	1248
1단계: AWS Management Pack 설치	1248
2단계: 감시자 노드 구성	1250
3단계: AWS Run As 계정 생성	1250
4단계: Add Monitoring Wizard 실행	1254
5단계: 포트 및 엔드포인트 구성	1259
사용	1260
보기	1260
검색	1274
모니터	1275
규칙	1276
이벤트	1276
상태 모델	1277
AWS Management Pack 사용자 지정	1279
업데이트	1279
System Center 2012	1279
System Center 2007 R2	1280
설치 제거	1280
System Center 2012	1280
System Center 2007 R2	1281
문제 해결	1281
오류 4101 및 4105	1281
오류 4513	1281
이벤트 623	1282
이벤트 2023 및 2120	1282
이벤트 6024	1282
System Center 2012 — Operations Manager에 대한 일반 문제 해결	1282
System Center 2007 R2에 대한 일반 문제 해결	1283
문서 기록	1284

지난 몇 년의 기록	1287
------------------	------

Amazon EC2이란 무엇입니까?

Amazon Elastic Compute Cloud(Amazon EC2)는 Amazon Web Services(AWS) 클라우드에서 확장식 컴퓨팅을 제공합니다. Amazon EC2를 사용하면 하드웨어에 선투자할 필요가 없어 더 빠르게 애플리케이션을 개발하고 배포할 수 있습니다. Amazon EC2를 통해 원하는 만큼 가상 서버를 구축하고 보안 및 네트워크 구성과 스토리지 관리가 가능합니다. 또한 Amazon EC2는 요구 사항이나 갑작스러운 인기 증대 등 변동 사항에 따라 신속하게 규모를 확장하거나 축소할 수 있어 서버 트래픽 예측 필요성이 줄어듭니다.

클라우드 컴퓨팅에 대한 자세한 내용은 [클라우드 컴퓨팅이란 무엇입니까? 단원](#)을 참조하십시오.

Amazon EC2의 기능

Amazon EC2는 다음의 기능을 제공합니다.

- 인스턴스: 가상 컴퓨팅 환경
- Amazon 머신 이미지(AMI): 서버에 필요한 운영체제와 여러 소프트웨어들이 적절히 구성된 상태로 제공되는 템플릿으로 인스턴스를 쉽게 만들 수 있습니다.
- 인스턴스 유형: 인스턴스를 위한 CPU, 메모리, 스토리지, 네트워킹 용량의 여러 가지 구성 제공
- 키 페어를 사용하여 인스턴스 로그인 정보 보호(AWS는 퍼블릭 키를 저장하고 사용자는 개인 키를 안전한 장소에 보관하는 방식)
- 인스턴스 스토어 볼륨: 임시 데이터를 저장하는 스토리지 볼륨으로 인스턴스 종료 시 삭제됨
- Amazon Elastic Block Store(Amazon EBS), 즉 Amazon EBS 볼륨을 사용해 영구 스토리지 볼륨에 데이터 저장
- 인스턴스와 Amazon EBS 볼륨 등의 리소스를 다른 물리적 장소에서 액세스할 수 있는 리전 및 가용 영역
- 보안 그룹을 사용해 인스턴스에 연결할 수 있는 프로토콜, 포트, 소스 IP 범위를 지정하는 방화벽 기능
- 탄력적 IP 주소(EIP): 동적 클라우드 컴퓨팅을 위한 고정 IPv4 주소
- 태그: 사용자가 생성하여 Amazon EC2 리소스에 할당할 수 있는 메타데이터
- AWS 클라우드에서는 논리적으로 격리되어 있지만, 원활 때마다 고객의 네트워크와 간편히 연결할 수 있는 가상 네트워크, Virtual Private Clouds(VPC)

Amazon EC2에 대한 자세한 내용은 [Amazon EC2제품 페이지](#)를 참조하십시오.

Amazon EC2를 사용하면 호환 가능한 모든 Windows 기반 솔루션을 비용 효율성이 높은 안정적인 고성능 클라우드 컴퓨팅 플랫폼에서 실행할 수 있습니다. 자세한 내용은 [AWS의 Windows Server](#)를 참조하십시오.

AWS에서의 웹사이트 실행에 대한 자세한 내용은 [웹 호스팅](#)을 참조하십시오.

Amazon EC2 시작 방법

먼저, Amazon EC2를 사용하도록 설정해야 합니다. 일단 설정을 하시면, Amazon EC2 시작 자습서를 따라 하실 수 있습니다. 기능에 대한 추가 정보가 필요한 경우에는 기술 문서를 참조하십시오.

실행 안내

- [Amazon EC2 설정 \(p. 12\)](#)

- 자습서: Amazon EC2 Windows 인스턴스 시작하기 (p. 16)

기본

- Amazon EC2 기본 인프라(Windows) (p. 4)
- 인스턴스 유형 (p. 114)
- Tags (p. 1143)

네트워킹 및 보안

- Amazon EC2 키 페어 및 Windows 인스턴스 (p. 900)
- 보안 그룹 (p. 908)
- 탄력적인 IP 주소 (p. 721)
- Amazon EC2 및 Amazon VPC (p. 776)

스토리지

- Amazon EBS (p. 928)
- 인스턴스 스토어 (p. 1091)

Windows 인스턴스 작업

- AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 Run Command](#)
- Windows Server와 Amazon EC2 Windows 인스턴스의 차이점 (p. 9)
- Amazon EC2 Windows 인스턴스 구동을 위한 애플리케이션 개발 (p. 10)
- AWS 시작: .NET 웹 앱 호스팅

AWS 구매에 관련된 질문은 [AWS 영업부에 문의](#)하십시오. Amazon EC2에 관련된 기술적인 문의 사항은 [Amazon EC2 forum](#) 단원을 참조하십시오.

관련 서비스

Amazon EC2를 사용하여 인스턴스, 볼륨 같은 Amazon EC2 리소스를 직접 프로비저닝할 수 있습니다. 또한 AWS의 다른 서비스를 사용하여 Amazon EC2를 프로비저닝할 수도 있습니다. 자세한 내용은 다음 설명서를 참조하세요.

- [Amazon EC2 Auto Scaling 사용 설명서](#)
- [AWS CloudFormation 사용 설명서](#)
- [AWS Elastic Beanstalk 개발자 안내서](#)
- [AWS OpsWorks User Guide](#)

Elastic Load Balancing는 애플리케이션의 인바운드 트래픽을 여러 인스턴스로 자동으로 분산해 줍니다. 자세한 내용은 [Elastic Load Balancing 사용 설명서](#)를 참조하십시오.

Amazon Relational Database Service(Amazon RDS)로 데이터베이스 인스턴스를 실행하여, AWS에 의해 관리되는, 관계형 데이터베이스를 사용하실 수 있습니다. EC2 인스턴스로 데이터베이스를 구축하는 것도 가능하지만, Amazon RDS를 선택하면, 직접 소프트웨어 패치 적용, 백업 및 백업 데이터 저장 등 데이터베이스 관리 작업을 할 필요가 없습니다. 자세한 내용은 [Amazon Relational Database Service 개발자 안내서](#)를 참조하십시오.

EC2 인스턴스의 클러스터에서 Docker 컨테이너를 더 쉽게 관리 할 수 있도록 하려면 Amazon Elastic Container Service(Amazon ECS)를 사용하십시오. 자세한 내용은 [Amazon Elastic Container Service Developer Guide](#) 또는 [Amazon Elastic Container Service AWS Fargate 사용 설명서](#)를 참조하십시오.

인스턴스와 Amazon EBS 볼륨에 관련된 기본 통계 정보를 모니터링하려면 Amazon CloudWatch를 사용합니다. 자세한 내용은 [Amazon CloudWatch 사용 설명서](#) 단원을 참조하십시오. EC2 인스턴스의 잠재적으로 승인된 사용 또는 악의적인 사용을 탐지하려면 Amazon GuardDuty를 사용합니다. 자세한 내용은 [Amazon GuardDuty 사용 설명서](#)를 참조하십시오.

Amazon EC2에 액세스

Amazon EC2는 웹 기반 사용자 인터페이스인 Amazon EC2 콘솔을 제공합니다. AWS 계정에 가입한 고객은 AWS Management 콘솔에 로그인한 후 콘솔 홈페이지에서 EC2를 선택하여 Amazon EC2에 액세스할 수 있습니다.

명령줄 인터페이스를 선호하는 고객의 경우 다음과 같은 옵션이 있습니다.

AWS 명령줄 인터페이스(CLI)

다양한 AWS 제품에서 사용되는 명령어를 제공하며 Windows, Mac, Linux를 지원합니다. 시작하려면 [AWS Command Line Interface 사용 설명서](#) 단원을 참조하십시오. Amazon EC2 명령어에 대한 자세한 내용은 [EC2\(AWS CLI Command Reference\)](#) 단원을 참조하십시오.

Windows PowerShell용 AWS 도구

PowerShell 환경에서 스크립트 작업을 선호하는 다양한 AWS 제품을 관리할 수 있도록, 명령줄 도구를 제공합니다. 시작하려면 [Windows PowerShell용 AWS 도구 사용 설명서](#) 단원을 참조하십시오. Amazon EC2용 cmdlets에 대한 자세한 내용은 [PowerShell용 AWS 도구 Cmdlet Reference](#) 단원을 참조하십시오.

Amazon EC2에서는 쿼리 API를 제공합니다. 이 리퀘스트들은, HTTP나 HTTPS의 메시지 교환 방식인 GET이나 POST이며, 미리 정해진 이름인 "Action"을 쿼리 변수로 사용합니다. Amazon EC2에 관련된 API 작업에 대한 자세한 내용은 [작업\(Amazon EC2 API Reference\)](#)을 참조하십시오.

HTTP나 HTTPS 리퀘스트를 직접 보내는 대신, 각 언어가 제공하는 고유의 API를 사용하여 애플리케이션을 빌드하는 것을 선호하는 개발자를 위해 AWS는, 라이브러리, 샘플 코드, 자습서 및 기타 리소스를 제공합니다. 이 라이브러리는 HTTP/HTTPS 리퀘스트에 암호화된 사인하기, 다시 리퀘스트를 보내기, 오류 응답 처리하기 등의, 작업을 자동화할 수 있는 기본적인 기능을 제공합니다. 자세한 내용은 [AWS SDK 및 도구](#)를 참조하십시오.

Amazon EC2 가격

AWS 가입 시 무상으로 Amazon EC2를 시작할 수 있는 [AWS 프리 티어](#)를 제공합니다.

예를 들어, Amazon EC2는 다음의 구입 옵션이 있습니다:

온디맨드 인스턴스

장기 약정이나 선결제 금액 없이 시간 단위로 사용한 인스턴스에 대한 요금을 지불하는 방식입니다.

Savings Plans

1년 또는 3년 기간 동안 시간당 USD로 일관된 사용량을 약정하여 Amazon EC2 비용을 절감할 수 있습니다.

예약 인스턴스

1년 또는 3년 기간 동안 인스턴스 유형 또는 지역을 포함해 특정 인스턴스 구성을 약정하여 Amazon EC2 비용을 절감할 수 있습니다.

스팟 인스턴스

미사용 EC2 인스턴스를 요청하여 Amazon EC2 비용을 대폭 줄일 수 있습니다.

Amazon EC2에 관련된 전체적인 요금 및 가격 목록은 [Amazon EC2 요금](#)을 참조하십시오.

예시로 주어진 프로비저닝 환경에서의 비용은 [클라우드 경제 센터](#)에서 계산할 수 있습니다.

청구 요금은 [AWS Billing and Cost Management 콘솔](#)의 결제 및 비용 관리 대시보드에서 확인할 수 있습니다. 청구서에는 요금 내역을 자세하게 확인할 수 있는 사용 보고서 링크가 포함됩니다. AWS 계정 결제에 대한 자세한 내용은 [AWS 계정 결제 단원](#)을 참조하십시오.

AWS 결제, 계정 및 이벤트에 관련된 질문은 [AWS Support](#)에 문의하십시오.

AWS 환경에서의 비용과 보안, 성능 최적화를 돋는 Trusted Advisor의 개요는 [AWS Trusted Advisor](#) 단원을 참조하십시오.

PCI DSS 준수

Amazon EC2에서는 전자 상거래 웹사이트 운영자 또는 서비스 공급자에 의한 신용 카드 데이터의 처리, 저장 및 전송을 지원하며, Payment Card Industry(PCI) Data Security Standard(DSS) 준수를 검증 받았습니다. AWS PCI 규정 준수 패키지의 사본을 요청하는 방법 등 PCI DSS에 대해 자세히 알아보려면 [PCI DSS 레벨 1](#)을 참조하십시오.

Amazon EC2 기본 인프라(Windows)

Amazon EC2를 시작하면서 기본 인프라의 구성요소와 함께 보유 중인 데이터 센터 비교, 대조되는 부분을 이해하고 있으면 사용에 도움이 됩니다.

개념

- [Amazon 머신 이미지와 인스턴스 \(p. 4\)](#)
- [리전 및 영역 \(p. 5\)](#)
- [스토리지 \(p. 6\)](#)
- [루트 디바이스 볼륨 \(p. 7\)](#)
- [네트워킹 및 보안 \(p. 8\)](#)
- [AWS Identity and Access Management \(p. 8\)](#)
- [Windows Server와 Amazon EC2 Windows 인스턴스의 차이점 \(p. 9\)](#)
- [Amazon EC2 Windows 인스턴스 구동을 위한 애플리케이션 개발 \(p. 10\)](#)

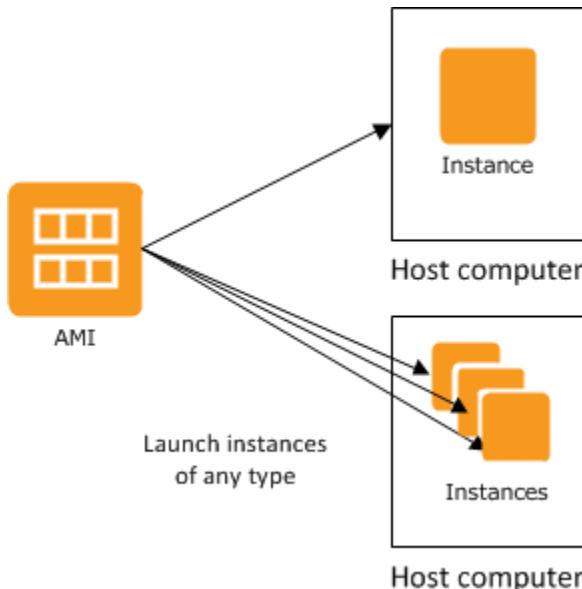
Amazon 머신 이미지와 인스턴스

Amazon 머신 이미지(AMI)는 소프트웨어 구성이 기재된 템플릿입니다(예: 운영 체제, 애플리케이션 서버, 애플리케이션). AMI를 사용해 인스턴스를 시작할 수 있으며, 이 AMI의 사본인 인스턴스는 클라우드에서 가상 서버로서 실행됩니다.

Amazon에서는 자주 사용되는 소프트웨어 구성과 함께 다양한 AMI를 공개 제공하고 있습니다. 그뿐 아니라 AWS 개발자 커뮤니티 회원들이 올린 자체 구성 AMI도 게시되어 있습니다. 사용자 지정 AMI는 누구나

생성할 수 있으며 AMI를 생성하면 필요한 기능을 모두 갖춘 새 인스턴스를 쉽고 빠르게 시작할 수 있습니다. 예를 들어 애플리케이션이 웹 사이트나 웹 서비스인 경우, AMI에 웹 서버와 관련 고정 콘텐츠, 그리고 동적 페이지에 사용할 코드를 포함할 수 있습니다. 이 AMI에서 인스턴스를 시작하면 웹 서버가 시작되고 애플리케이션에서 바로 요청을 처리할 수 있습니다.

하나의 AMI에서 다양한 인스턴스 유형을 실행할 수 있습니다. 인스턴스 유형에 따라 인스턴스에 사용되는 호스트 컴퓨터의 하드웨어가 기본적으로 결정됩니다. 각 인스턴스 유형은 서로 다른 컴퓨팅 및 메모리 특징을 제공합니다. 인스턴스에서 실행하려는 애플리케이션 또는 소프트웨어에 필요한 메모리 양과 컴퓨팅 파워를 기준으로 인스턴스 유형을 선택하십시오. Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오. 다음 그림과 같이 하나의 AMI로 여러 인스턴스를 시작할 수 있습니다.



Windows 인스턴스는 종지하거나 종료할 때까지, 혹은 오류가 발생하지 않는 한 계속 실행됩니다. 인스턴스가 실패하면 AMI에서 새로 실행할 수 있습니다.

AWS 계정당 동시에 실행할 수 있는 인스턴스 수는 제한됩니다. 해당 제한 및 추가 요청 방법에 대한 자세한 내용은 [Amazon EC2의 실행 인스턴스 한도](#)(일반 FAQ의 Amazon EC2) 단원을 참조하십시오.

리전 및 영역

Amazon EC2는 세계 각지의 여러 곳에서 호스팅되고 있습니다. 이 위치는 리전, 가용 영역, 로컬 영역 및 Wavelength Zone으로 구성됩니다. 각 리전은 개별 지리 영역입니다.

- 가용 영역은 각 리전 내에 있는 여러 격리된 위치입니다.
- 로컬 영역에서는 최종 사용자에게 가까운 여러 위치에 컴퓨팅, 스토리지 등의 리소스를 배치할 수 있는 기능을 제공합니다.
- AWS Outposts는 네이티브 AWS 서비스, 인프라 및 운영 모델을 사실상 모든 데이터 센터, 코로케이션 공간 또는 온프레미스 시설로 옮길 수 있습니다.
- Wavelength Zone을 사용하면 개발자는 5G 디바이스 및 최종 사용자에게 매우 짧은 지연 시간을 제공하는 애플리케이션을 빌드할 수 있습니다. Wavelength는 표준 AWS 컴퓨팅 및 스토리지 서비스를 통신 사업자의 5G 네트워크 엣지에 배포합니다.

AWS는 최신 기술을 탑재한 고가용성 데이터 센터를 운영하고 있습니다. 드물기는 하지만 동일한 위치에 있는 인스턴스의 가용성에 영향을 미치는 장애가 발생할 수도 있습니다. 장애의 영향을 받는 위치 한 곳에서 모든 인스턴스를 호스팅하면 인스턴스를 전혀 사용하지 못하게 될 수 있습니다.

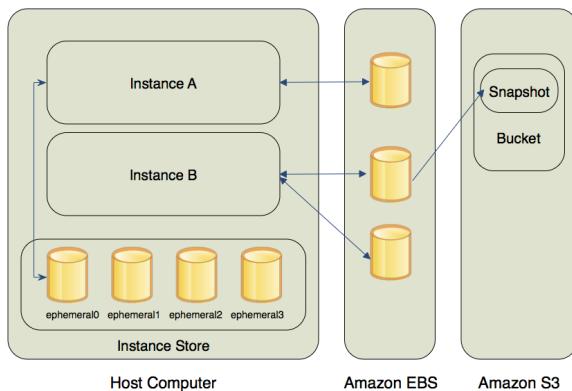
사용 가능한 리전 및 가용 영역에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [리전 및 영역](#)을 참조하세요.

스토리지

Amazon EC2를 사용하면서 저장이 필요한 데이터가 생길 수 있습니다. Amazon EC2에서 제공되는 스토리지 옵션:

- [Amazon Elastic Block Store\(Amazon EBS\)](#)
- [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#)
- [Amazon Simple Storage Service\(Amazon S3\)](#)

다음 그림은 스토리지 유형 간의 관계도입니다.

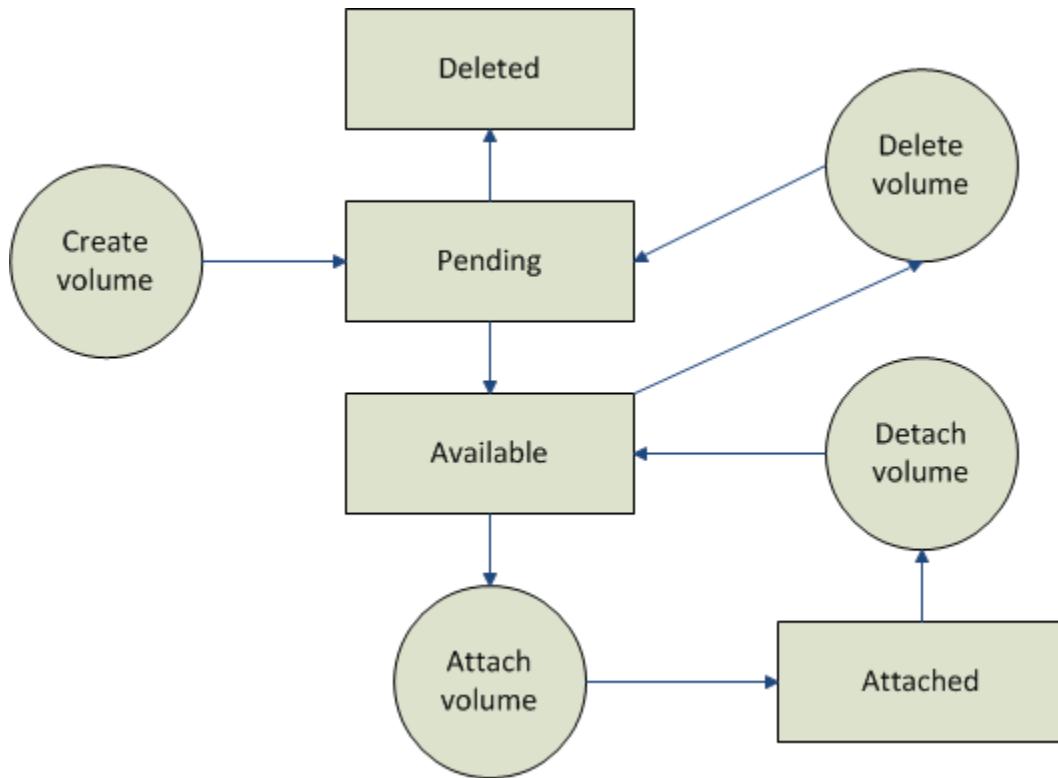


Amazon EBS 볼륨

Amazon EBS 볼륨은 대부분의 경우에 권장되는 스토리지 옵션입니다. Amazon EBS는 인스턴스를 위한 블록 수준의 영구 스토리지를 제공합니다. Amazon EBS 볼륨은 사실상 하드 디스크이며 실행하는 인스턴스에 연결할 수 있습니다.

Amazon EBS는 데이터베이스, 파일 시스템 또는 원시 블록 수준 스토리지에 대한 액세스가 필요한 애플리케이션에 특히 적합합니다.

이전 그림에서 설명된 것과 같이 여러 볼륨을 하나의 인스턴스에 연결할 수 있습니다. 또한, EBS 볼륨의 스냅샷을 생성하여 Amazon S3에 저장하면 데이터의 백업 사본을 보관할 수 있습니다. 스냅샷을 사용하면 새 Amazon EBS 볼륨을 생성하고 다른 인스턴스에 연결할 수 있습니다. 한 인스턴스에서 EBS 볼륨을 분리한 다음 이 볼륨을 다른 인스턴스에 연결하는 것도 가능합니다. 다음 그림은 EBS 볼륨의 수명 주기를 설명합니다.



Amazon EBS 볼륨에 대한 자세한 내용은 [Amazon Elastic Block Store \(p. 928\)](#) 단원을 참조하십시오.

인스턴스 스토어

인스턴스 스토어는 인스턴스에 임시 블록 수준 스토리지를 제공합니다. 이 스토리지는 호스트 컴퓨터에 물리적으로 연결됩니다. 인스턴스 스토어 볼륨의 데이터는 연결된 인스턴스를 중지하거나 종료하면 더 이상 보관되지 않습니다. 지원되는 각 인스턴스 유형에서 사용할 수 있는 인스턴스 스토어 볼륨 목록은 [인스턴스 스토어 볼륨 \(p. 1093\)](#) 단원을 참조하십시오.

인스턴스 스토어는 비용 부담 없이 사용할 수 있는 임시 스토리지입니다. 데이터의 영구 보존이 필요하지 않을 때는 인스턴스 스토어 볼륨을 사용하면 됩니다. 인스턴스 스토어 볼륨에 대한 자세한 내용은 [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#) 단원을 참조하십시오.

Amazon S3

Amazon S3는 인터넷 스토리지입니다. 단순한 웹 서비스 인터페이스로 웹상 어디서나 용량에 관계없이 데이터를 저장하고 검색할 수 있습니다. Amazon S3에 대한 자세한 내용은 [Amazon S3 제품 페이지](#)를 참조하십시오.

루트 디바이스 볼륨

인스턴스를 시작하면 인스턴스 부팅에 사용된 이미지가 루트 디바이스 볼륨에 저장됩니다. Windows 인스턴스를 시작하면 루트 EBS 볼륨이 EBS 스냅샷에서 생성되어 이 인스턴스에 연결됩니다.

기본적으로 인스턴스가 종료되면 루트 볼륨이 삭제됩니다(DeleteOnTermination 속성은 true임). 콘솔을 사용하여 인스턴스 시작 시 DeleteOnTermination을 변경할 수 있습니다. 기존 인스턴스의 속성을 변경하려면 명령줄을 사용해야 합니다.

콘솔을 사용해 인스턴스의 루트 디바이스 볼륨 유지를 설정하는 방법(인스턴스 시작 시)

1. Amazon EC2 콘솔을 엽니다.
2. Amazon EC2 콘솔 대시보드에서 인스턴스 시작을 선택합니다.
3. Amazon Machine Image(AMI) 선택 페이지에서 사용할 AMI를 선택하고 선택을 선택합니다.
4. 마법사 안내에 따라 인스턴스 유형 선택 및 인스턴스 세부 정보 구성 설정을 완료합니다.
5. 스토리지 추가 페이지에서 루트 볼륨에 대한 종료 시 삭제 확인란 선택을 해제합니다.
6. 나머지 마법사 페이지를 완료한 후 시작을 선택합니다.

인스턴스의 세부 정보 창에서 루트 디바이스 볼륨의 세부 정보를 조회하여 설정을 확인할 수 있습니다. 블록 디바이스 옆의 루트 디바이스 볼륨 항목을 선택합니다. 종료 시 삭제의 기본 설정은 `True`입니다. 기본 설정을 변경하면 종료 시 삭제의 설정 값이 `False`가 됩니다.

명령줄을 사용해 인스턴스의 루트 디바이스 볼륨 유지를 설정하는 방법

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [modify-instance-attribute\(AWS CLI\)](#)
- [Edit-EC2InstanceAttribute\(Windows PowerShell용 AWS 도구\)](#)

네트워킹 및 보안

인스턴스는 기본 VPC로 실행된 경우에만 퍼블릭 IPv4 주소가 할당되도록 기본 설정됩니다. 기본이 아닌 VPC로 실행되는 인스턴스에 퍼블릭 IPv4 주소를 할당하려면 인스턴스를 시작할 때 따로 지정하거나 서브넷의 기본 퍼블릭 IPv4 주소 할당 방식을 수정해야 합니다.

사용자가 제어할 수 없는 이유로 인해 인스턴스 오류가 발생하거나 종료될 수 있습니다. 인스턴스에 오류가 발생하여 대체 인스턴스를 시작한 경우, 대체 인스턴스에 원본 인스턴스와 다른 퍼블릭 IPv4 주소가 할당됩니다. 단, 고정 IPv4 주소가 필요한 애플리케이션은 Amazon EC2에서 탄력적 IP 주소를 제공합니다. 자세한 내용은 [Amazon EC2 인스턴스 IP 주소 지정 \(p. 702\)](#) 단원을 참조하십시오.

보안 그룹을 사용하면 인스턴스에 액세스할 수 있는 사용자를 관리할 수 있습니다. 보안 그룹은 인스턴스 접속할 수 있는 허용 프로토콜, 포트, 소스 IP 범위를 지정하는 인바운드 네트워크 방화벽과 유사합니다. 여러 보안 그룹을 생성하고 각 그룹마다 다른 규칙을 할당할 수 있습니다. 그럼 다음 각 인스턴스에 보안 그룹을 1개 이상 배정하고 규칙을 사용하여 인스턴스에 도달할 수 있는 트래픽을 지정할 수 있습니다. 지정된 IP 주소나 보안 그룹에서만 인스턴스에 액세스할 수 있도록 보안 그룹을 구성할 수 있습니다. 자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#) 단원을 참조하십시오.

AWS Identity and Access Management

AWS Identity and Access Management(IAM)을 사용하여 다음을 수행할 수 있습니다.

- AWS 계정의 사용자와 그룹 생성
- AWS 계정 사용자 각각에 고유한 보안 자격 증명 할당
- 작업 수행 시 각 사용자의 AWS 리소스 사용 권한 제어
- 다른 AWS 계정의 사용자와 AWS 리소스 공유
- AWS 계정에 적용할 규칙 생성 및 규칙을 관리할 사용자나 서비스 규정
- 엔터프라이즈의 기존 자격 증명을 사용해 AWS 리소스를 사용하는 작업 권한 허용

IAM과 Amazon EC2 함께 사용하면 조직 내 사용자별로 특정 Amazon EC2 API 작업을 사용하는 작업 수행과 특정 AWS 리소스의 사용 권한을 제어할 수 있습니다.

IAM에 대한 자세한 내용은 다음을 참조하십시오.

- IAM 그룹 및 사용자 생성 (p. 837)
- Amazon EC2에 대한 IAM 정책 (p. 838)
- Amazon EC2에 대한 IAM 역할 (p. 890)
- AWS Identity and Access Management (IAM)
- IAM 사용 설명서

Windows Server와 Amazon EC2 Windows 인스턴스의 차이점

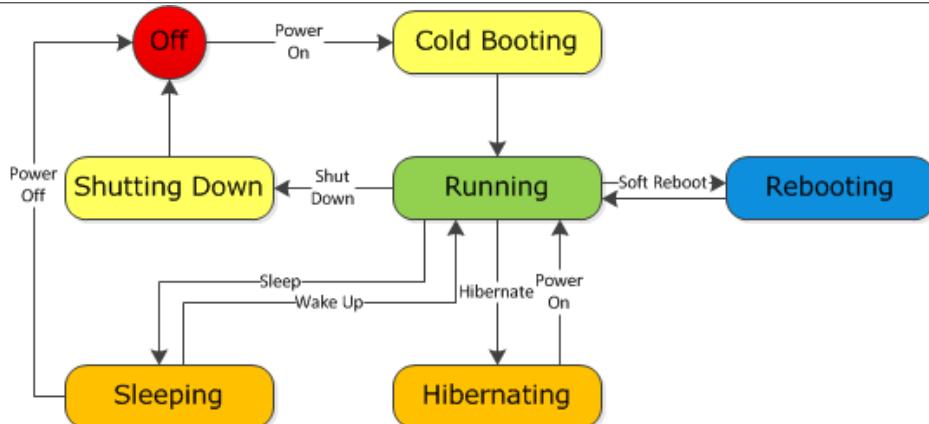
Amazon EC2 Windows 인스턴스는 시작한 후에는 기존의 Windows Server 구동 서버와 비슷하게 수행됩니다. 예를 들어, Windows Server와 Amazon EC2 인스턴스 모두 웹 애플리케이션 구동이나 배치 프로세스 (batch processing), 또는 대규모 연산 기능이 필요한 애플리케이션의 관리에 사용할 수 있습니다. 그러나 서버 하드웨어 모델과 클라우드 컴퓨팅 모델 간에는 중요한 차이가 있습니다. Amazon EC2 인스턴스의 실행 방식은 기존의 Windows Server 구동 서버과 다릅니다.

Amazon EC2 Windows 인스턴스를 시작하기 전에 클라우드에서 구동되는 애플리케이션의 아키텍처가 사용 하드웨어에서 사용 중인 기존의 애플리케이션 모델과 크게 다를 수 있다는 사실을 알아두어야 합니다. 클라우드 서버에서 애플리케이션을 구현하려면 개발 과정에서 이런 차이를 고려해야 합니다.

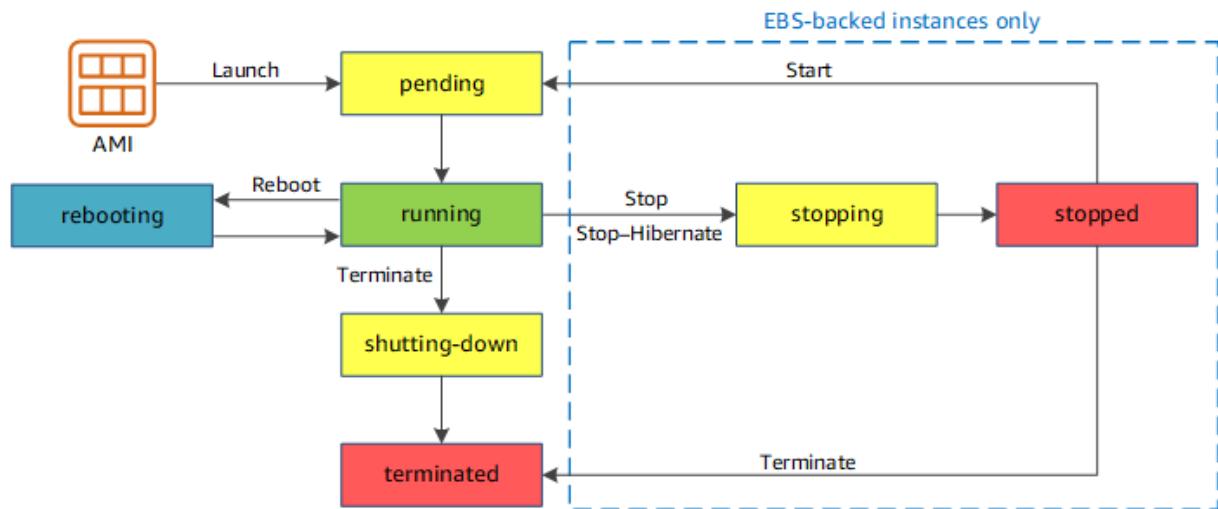
다음 표는 Windows Server와 Amazon EC2 Windows 인스턴스의 주요 차이점을 설명합니다.

Windows Server	Amazon EC2 Windows 인스턴스
리소스와 용량이 물리적으로 제한됩니다.	리소스와 용량을 확장할 수 있습니다.
인프라를 사용하지 않을 때에도 비용이 소요됩니다.	인프라를 사용한 만큼 비용을 지불합니다. 인스턴스 상태가 "중지"나 "종료"로 변경되는 즉시 요금 발생이 중지됩니다.
물리적인 공간과 정기적인 관리가 요구됩니다.	물리적인 공간이나 정기적인 관리가 필요 없습니다.
전원 버튼을 눌러 시작하는 방식입니다(콜드부팅).	인스턴스를 실행하여 시작합니다.
종료가 필요할 때까지 서버를 계속 구동하거나 절전 (sleep) 또는 수면(hibernation) 상태로 변경할 수 있습니다(이들 상태에서는 서버 전원이 꺼집니다).	서버를 계속 구동하거나 종지했다가 재시작할 수 있습니다(이 때 인스턴스는 새 호스트 컴퓨터로 이동 됩니다).
서버를 종료해도 모든 리소스가 변동 없이 종료 당시 상태로 유지됩니다. 하드 드라이브에 저장된 데 이터가 계속 보존되어 필요할 때 언제든지 액세스할 수 있습니다. 전원을 다시 공급하는 방식으로 서버를 실행 상태로 복구할 수 있습니다.	인스턴스를 종료하면 이 인스턴스의 인프라는 더 이상 사용할 수 없게 됩니다. 인스턴스를 종료한 후에는 인스턴스에 다시 연결하거나 재시작할 수 없습니다. 단, 인스턴스가 실행 중일 때 이미지를 생성하고 이 이미지를 사용해 이후 언제든지 새 인스턴스를 시작할 수는 있습니다.

기존의 Windows Server 구동 서버에서는 다음 도표의 상태를 거쳐 실행됩니다.



다음의 도표와 Windows Server 도표를 비교하면 Amazon EC2 Windows 인스턴스가 기존 Windows Server와 유사한 것을 알 수 있습니다. 인스턴스는 시작 후 잠시 보류 상태가 되었다가 등록이 완료되면 실행 상태로 변경됩니다. 인스턴스는 사용자가 중지하거나 종료할 때까지 계속 실행됩니다. 종료(termination) 후에는 인스턴스를 재시작할 수 없습니다. 인스턴스가 실행 중일 때 백업 이미지를 생성하면 이 백업 이미지를 사용해 나중에 새 인스턴스를 시작할 수 있습니다.



Amazon EC2 Windows 인스턴스 구동을 위한 애플리케이션 개발

Amazon EC2 Windows 인스턴스에서 구동할 애플리케이션을 개발하려면 이전 섹션에서 언급한 차이점을 반영하는 것이 중요합니다.

Amazon EC2용으로 구축된 애플리케이션은 특정 주기 없이 필요에 따라 기본 컴퓨팅 인프라를 사용합니다. 이런 애플리케이션은 작업 수행에 필요한 경우에만 리소스(스토리지, 컴퓨팅 등)를 임시로 사용하다가 작업을 마치면 리소스 사용 권한을 반납합니다. 그뿐 아니라 작업을 마치면 자가 종료되는 경우가 많습니다. 실행 중인 애플리케이션은 리소스 요구사항에 따라 탄력적으로 확대되거나 축소됩니다. Amazon EC2 인스턴스에서 구동되는 애플리케이션은 인프라 오작동에 대비해 종료했다가 원할 때 다양한 요소를 재생성할 수 있습니다.

Amazon EC2에서 구동할 애플리케이션을 개발할 때는 요구사항 변동에 따른 컴퓨팅 및 스토리지 리소스의 신속한 배치와 감축을 염두하여 계획해 볼 수 있습니다.

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
Amazon EC2 Windows 인스턴
스 구동을 위한 애플리케이션 개발

Amazon EC2 Windows 인스턴스를 실행할 때는 Windows Server처럼 하드웨어와 소프트웨어, 스토리지를 포함한 시스템 패키지를 엄격하게 프로비저닝하지 않아도 됩니다. 대신 다양한 클라우드 리소스를 활용해 Windows 애플리케이션의 확장성과 전체적인 성능을 향상하는 데 집중할 수 있습니다.

Amazon EC2에서 오작동이나 정전에 대비한 설계는 아키텍처에서 없어서는 안 될 중요한 부분입니다. 어느 확장 가능한 중복 시스템과 마찬가지로 이런 시스템 아키텍처 또한 컴퓨팅, 네트워크, 그리고 스토리지 오작동의 가능성에 대비해야 합니다. 오작동의 각 유형에 따른 대처 매커니즘이 애플리케이션에 구축되어 있어야 합니다. 핵심은 개별 구성요소 간에 긴밀하게 결합되지 않은 상태로 비동기적인 상호 작용이 가능하며 다른 요소를 독자적인 확장이 가능한 블랙박스로 간주하는 모듈형 시스템을 구축하는 것입니다. 그러면 이 종한 구성요소에 오류가 발생하거나 사용 중일 때 사용 시스템에 무리가 없도록 해당 요소의 인스턴스를 추가로 실행하면 됩니다.

오작동에 대비한 설계에서 또 다른 중요한 점은 애플리케이션의 지리적 배포입니다. 애플리케이션을 지리적으로 분산된 리전에 복제하면 시스템의 가용성이 향상됩니다.

Amazon EC2 인프라는 프로그램이 가능하고 스크립트를 사용해 배치 과정, 프로그램과 애플리케이션의 설치와 구성, 가상 서버의 부트스트랩을 자동화 할 수 있습니다.

Amazon EC2 Windows 인스턴스에서 구동되는 애플리케이션 아키텍처는 각 계층마다 보안을 설정해야 합니다. Amazon EC2 환경에 저장하는 것이 우려되는 민감한 데이터나 기밀 정보는 업로드하기 전에 암호화해야 합니다.

Amazon EC2 설정

Amazon EC2 인스턴스를 처음으로 설정하려면 이 단원의 작업을 완료합니다.

1. AWS에 가입 (p. 12)
2. 키 페어 생성 (p. 12)
3. 보안 그룹 생성 (p. 13)

작업을 마치면 [Amazon EC2 시작하기 \(p. 16\)](#) 자습서를 사용할 준비가 됩니다.

AWS에 가입

Amazon Web Services(AWS)에 가입하면 Amazon EC2를 포함해 AWS의 모든 서비스에 AWS 계정이 자동으로 등록됩니다. 사용한 서비스에 대해서만 청구됩니다.

Amazon EC2에서는 사용한 만큼만 지불하면 됩니다. AWS를 처음 사용하는 고객인 경우 Amazon EC2를 무료로 시작할 수 있습니다. 자세한 정보는 [AWS 프리 티어](#) 단원을 참조하십시오.

이미 AWS 계정이 있다면 다음 작업으로 건너뛰십시오. AWS 계정이 없는 경우에는 아래 단계를 수행하여 계정을 만드십시오.

AWS 계정을 만들려면 다음을 수행합니다.

1. <https://portal.aws.amazon.com/billing/signup>을 엽니다.
2. 온라인 지시 사항을 따릅니다.

등록 절차 중 전화를 받고 전화 키패드를 사용하여 확인 코드를 입력하는 과정이 있습니다.

키 페어 생성

AWS에서는 퍼블릭 키 암호화를 사용하여 인스턴스에 대한 로그인 정보를 보호합니다. 인스턴스를 시작할 때 키 페어의 이름을 지정한 다음 프라이빗 키를 제공하여 RDP를 사용하여 로그인할 수 있도록 Windows 인스턴스에 대한 관리자 암호를 가져옵니다.

키 페어를 아직 생성하지 않은 경우 Amazon EC2 콘솔을 사용하여 생성할 수 있습니다. 여러 리전에서 인스턴스를 시작하려면 각 리전에서 키 페어를 생성해야 합니다. 리전에 대한 자세한 내용은 [리전 및 영역 \(p. 5\)](#) 단원을 참조하십시오.

다음 방법 중 하나를 사용하여 키 페어를 만들 수 있습니다.

새로운 콘솔

키 페어를 만들려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Key Pairs]를 선택합니다.
3. Create key pair(키 페어 생성)를 선택합니다.

4. 이름에 설명이 포함된 키 페어 이름을 입력합니다. Amazon EC2는 키 이름으로 지정한 이름과 퍼블릭 키를 연결합니다. 키 이름에는 최대 255자의 ASCII 문자를 포함할 수 있습니다. 선행 또는 후행 공백을 포함할 수 없습니다.
5. 파일 형식에서 프라이빗 키를 저장할 형식을 선택합니다. OpenSSH에서 사용할 수 있는 형식으로 프라이빗 키를 저장하려면 pem을 선택합니다. PuTTY에서 사용할 수 있는 형식으로 프라이빗 키를 저장하려면 ppk를 선택합니다.
6. Create key pair(키 페어 생성)를 선택합니다.
7. 브라우저에서 프라이빗 키 파일이 자동으로 다운로드됩니다. 기본 파일 이름은 키 페어의 이름으로 지정한 이름이며, 파일 이름 확장명은 선택한 파일 형식에 따라 결정됩니다. 안전한 장소에 프라이빗 키 파일을 저장합니다.

Important

이때가 사용자가 프라이빗 키 파일을 저장할 수 있는 유일한 기회입니다.

기존 콘솔

키 페어를 만들려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 NETWORK & SECURITY에서 Key Pairs를 선택합니다.

Note

탐색 창은 Amazon EC2 콘솔의 왼쪽에 있습니다. 창이 보이지 않는 경우 창이 최소화되었을 수 있으니 화살표를 선택해 확대하십시오.

3. 키 페어 생성을 선택합니다.
4. 키 페어 이름에서 새 키 페어의 이름을 입력한 다음 생성을 선택합니다. 이름에는 최대 255자의 ASCII 문자를 포함할 수 있습니다. 선행 또는 후행 공백을 포함할 수 없습니다.
5. 브라우저에서 프라이빗 키 파일이 자동으로 다운로드됩니다. 기본 파일 이름은 키 페어의 이름으로 지정된 이름이며, 파일 이름 확장명은 .pem입니다. 안전한 장소에 프라이빗 키 파일을 저장합니다.

Important

이때가 사용자가 프라이빗 키 파일을 저장할 수 있는 유일한 기회입니다.

자세한 내용은 [Amazon EC2 키 페어 및 Windows 인스턴스 \(p. 900\)](#) 단원을 참조하십시오.

보안 그룹 생성

보안 그룹은 연결된 인스턴스에 대한 방화벽 역할을 하여 인스턴스 수준에서 인바운드 트래픽과 아웃바운드 트래픽을 모두 제어합니다. RDP를 사용하여 IP 주소에서 인스턴스에 연결할 수 있게 하는 규칙을 보안 그룹에 추가해야 합니다. 어디서나 인바운드 및 아웃바운드 HTTP/HTTPS 액세스를 허용하는 규칙을 추가할 수도 있습니다.

여러 리전에서 인스턴스를 시작하려면 각 리전에서 보안 그룹을 생성해야 합니다. 리전에 대한 자세한 내용은 [리전 및 가용 영역 \(p. 5\)](#)을 참조하십시오.

사전 조건

로컬 컴퓨터의 퍼블릭 IPv4 주소가 필요합니다. Amazon EC2 콘솔의 보안 그룹 편집기는 퍼블릭 IPv4 주소를 자동으로 검색할 수 있습니다. 또는 인터넷 브라우저에서 "내 IP 주소"와 같은 검색 구문을 사용하거나 [Check IP](#) 서비스를 사용할 수도 있습니다. 고정 IP 주소가 없는 방화벽 뒤나 ISP(인터넷 서비스 공급자)를 통해 연결되어 있는 경우 클라이언트 컴퓨터가 사용하는 IP 주소의 범위를 찾아야 합니다.

다음 방법 중 하나를 사용하여 사용자 지정 보안 그룹을 생성할 수 있습니다.

새로운 콘솔

최소 권한으로 보안 그룹을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 보안 그룹을 생성할 리전을 선택합니다. 보안 그룹은 리전에 고유하므로 키 페어를 생성한 리전과 동일한 리전을 선택해야 합니다.
3. 탐색 창에서 [Security Groups]를 선택합니다.
4. 보안 그룹 생성을 선택합니다.
5. Basic details(기본 세부 정보) 섹션에서 다음을 수행합니다.
 - a. 새 보안 그룹의 이름과 설명을 입력합니다. 기억하기 쉬운 이름을 사용합니다(예: 사용자 이름, 뒤에 _SG_ 및 리전 이름). 예를 들어, me_SG_uswest2로 지정할 수 있습니다.
 - b. VPC 목록에서 리전의 기본 VPC를 선택합니다.
6. 인바운드 규칙 섹션에서 다음 규칙을 생성합니다(각 새 규칙에 대해 규칙 추가 선택).
 - Type(유형) 목록에서 HTTP를 선택하고 Source(소스)가 Anywhere(위치 무관)(0.0.0.0/0)로 설정되어 있는지 확인합니다.
 - 유형 목록에서 HTTPS를 선택하고 소스가 위치 무관(0.0.0.0/0)으로 설정되어 있는지 확인합니다.
 - 유형 목록에서 RDP를 선택합니다. 필드를 로컬 컴퓨터의 퍼블릭 IPv4 주소로 자동으로 채우려면 소스 상자에서 내 IP를 선택하면 됩니다. 또는 사용자 지정을 선택하고 컴퓨터 또는 네트워크의 퍼블릭 IPv4 주소를 CIDR 표기법으로 지정해도 됩니다. 개별 IP 주소를 CIDR 표기법으로 지정하려면 라우팅 접미사 /32를 추가합니다(예: 203.0.113.25/32). 회사에서 주소를 범위로 할당하는 경우 전체 범위(예: 203.0.113.0/24)를 지정합니다.

Warning

보안상 테스트를 위해 짧은 시간 동안만 허용하는 경우를 제외하고 모든 IPv4 주소(0.0.0.0/0)에서의 인스턴스에 대한 RDP 액세스를 허용하지 않는 것이 좋습니다.

7. 보안 그룹 생성을 선택합니다.

기존 콘솔

최소 권한으로 보안 그룹을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 보안 그룹 생성을 선택합니다.
4. 새 보안 그룹의 이름과 설명을 입력합니다. 기억하기 쉬운 이름을 사용합니다(예: 사용자 이름, 뒤에 _SG_ 및 리전 이름). 예를 들어, me_SG_uswest2로 지정할 수 있습니다.
5. VPC 목록에서 리전의 기본 VPC를 선택합니다.
6. 인바운드 탭에서 다음 규칙을 생성합니다(각 새 규칙에 대해 규칙 추가 선택).
 - Type(유형) 목록에서 HTTP를 선택하고 Source(소스)가 Anywhere(위치 무관)(0.0.0.0/0)로 설정되어 있는지 확인합니다.
 - 유형 목록에서 HTTPS를 선택하고 소스가 위치 무관(0.0.0.0/0)으로 설정되어 있는지 확인합니다.
 - 유형 목록에서 RDP를 선택합니다. 필드를 로컬 컴퓨터의 퍼블릭 IPv4 주소로 자동으로 채우려면 소스 상자에서 내 IP를 선택하면 됩니다. 또는 사용자 지정을 선택하고 컴퓨터 또는 네트워크의 퍼블릭 IPv4 주소를 CIDR 표기법으로 지정해도 됩니다. 개별 IP 주소를 CIDR 표기법으로 지정하려면 라우팅 접미사 /32를 추가합니다(예: 203.0.113.25/32). 회사에서 주소를 범위로 할당하는 경우 전체 범위(예: 203.0.113.0/24)를 지정합니다.

정하려면 라우팅 접미사 /32를 추가합니다(예: 203.0.113.25/32). 회사에서 주소를 범위로 할당하는 경우 전체 범위(예: 203.0.113.0/24)를 지정합니다.

Warning

보안상 테스트를 위해 짧은 시간 동안만 허용하는 경우를 제외하고 모든 IPv4 주소(0.0.0.0/0)에서의 인스턴스에 대한 RDP 액세스를 허용하지 않는 것이 좋습니다.

7. Create를 선택합니다.

Command line

최소 권한으로 보안 그룹을 생성하려면

다음 명령 중 하나를 사용합니다.

- [create-security-group](#)(AWS CLI)
- [New-EC2SecurityGroup](#)(Windows PowerShell용 AWS 도구)

자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#) 단원을 참조하십시오.

자습서: Amazon EC2 Windows 인스턴스 시작하기

이 자습서를 사용하여 Amazon Elastic Compute Cloud(Amazon EC2)를 시작하십시오. Windows 인스턴스를 시작, 연결 및 사용하는 방법을 터득할 수 있습니다. 인스턴스는 AWS 클라우드의 가상 서버입니다. Amazon EC2를 사용하여 인스턴스에서 실행되는 운영 체제와 애플리케이션을 설정하고 구성할 수 있습니다.

Linux 인스턴스를 시작하려면 [Amazon EC2 Linux 인스턴스 시작하기](#)를 참조하십시오.

AWS에 가입할 때 [AWS 프리 티어](#)를 사용하여 Amazon EC2 사용을 시작할 수 있습니다. 12개월 이전에 AWS 계정을 생성했지만 Amazon EC2에 대한 프리 티어 혜택을 아직 다 사용하지 않은 경우 프리 티어 혜택 안에 포함된 옵션을 선택하는 데 도움이 되는 이 자습서를 무료로 이용할 수 있습니다. 그렇지 않을 경우, 유 휴 상태로 유지되더라도 인스턴스를 시작하는 시점부터 인스턴스를 종료할 때까지(이 자습서의 최종 작업) 스텐다드 Amazon EC2 사용료가 발생합니다.

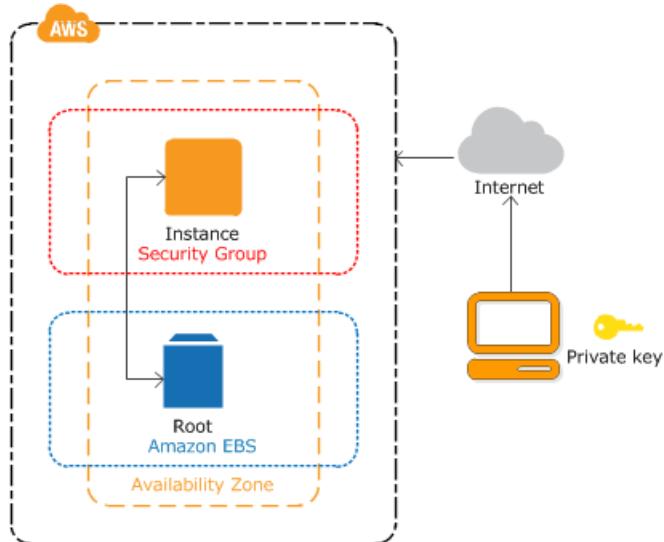
Windows Server를 실행하는 EC2 인스턴스의 특정 사용 사례에 대한 단계별 자습서는 [Windows Server를 실행하는 Amazon EC2 인스턴스에 대한 자습서 \(p. 1157\)](#) 단원을 참조하십시오.

내용

- [개요 \(p. 16\)](#)
- [사전 조건 \(p. 17\)](#)
- [1단계: 인스턴스 시작 \(p. 17\)](#)
- [2단계: 인스턴스에 연결 \(p. 18\)](#)
- [3단계: 인스턴스 정리 \(p. 19\)](#)
- [다음 단계 \(p. 20\)](#)

개요

이 인스턴스는 Amazon EBS 지원 인스턴스(루트 볼륨이 EBS 볼륨임을 의미)입니다. 인스턴스가 실행되는 가용 영역을 지정하거나 적합한 가용 영역이 Amazon EC2에서 자동으로 선택할 수 있습니다. 인스턴스를 시작할 때 키 페어와 보안 그룹을 지정하여 인스턴스 보안을 설정합니다. 인스턴스에 연결할 때는 인스턴스 시작 시 지정한 키 페어의 프라이빗 키를 지정해야 합니다.



작업

이 자습서를 완료하려면 다음 작업을 수행하십시오.

1. 인스턴스 시작 (p. 17)
2. 인스턴스에 연결합니다 (p. 18)
3. 인스턴스 정리 (p. 19)

관련 자습서

- Linux 인스턴스를 시작하려는 경우 Linux 인스턴스용 Amazon EC2 사용 설명서: [Amazon EC2 Linux 인스턴스 시작하기](#)에서 이 자습서를 참조하세요.
- 명령줄을 사용하려는 경우 AWS Command Line Interface 사용 설명서의 [AWS CLI를 통해 Amazon EC2 를 사용하는 방법](#) 자습서를 참조하십시오.

사전 조건

시작하기 전에 먼저 [Amazon EC2 설정 \(p. 12\)](#)의 단계를 완료해야 합니다.

1단계: 인스턴스 시작

다음 절차의 설명에 따라 Windows를 사용하여 AWS Management 콘솔 인스턴스를 시작할 수 있습니다. 이 자습서는 첫 번째 인스턴스를 빠르게 시작하도록 돕기 위한 것이므로 가능한 모든 옵션을 다루지는 않습니다. 고급 옵션에 대한 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하세요.

인스턴스를 시작하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 콘솔 대시보드에서 인스턴스 시작을 선택합니다.
3. Amazon Machine Image(AMI) 선택 페이지에 인스턴스에 대한 템플릿 역할을 하는 Amazon Machine Image(AMI)라는 기본 구성 목록이 표시됩니다. Windows Server 2016 Base 이상용 AMI를 선택합니다. 해당되는 AMI는 "프리 티어 사용 가능"으로 표시됩니다.

4. 인스턴스 유형 선택 페이지에서 인스턴스의 하드웨어 구성을 선택할 수 있습니다. 기본적으로 선택된 t2.micro 인스턴스 유형을 선택합니다. t2.micro 인스턴스 유형은 프리 티어로 이용할 수 있습니다. t2.micro를 사용할 수 없는 리전에서는 프리 티어 아래의 t3.micro 인스턴스를 사용할 수 있습니다. 자세한 정보는 [AWS 프리 티어](#) 단원을 참조하십시오.
5. 검토 후 시작(Review and Launch)을 선택하여 마법사가 다른 구성 설정을 완료하게 합니다.
6. 인스턴스 시작 검토 페이지의 보안 그룹 아래에서 마법사가 보안 그룹을 만들고 선택했음을 확인합니다. 이 보안 그룹을 사용하거나, 다음 단계를 이용하여 설정을 시작할 때 만든 보안 그룹을 선택합니다.
 - a. 보안 그룹 편집을 선택합니다.
 - b. 보안 그룹 구성 페이지에서 Select an existing security group(기존 보안 그룹 선택)이 선택되어 있는지 확인합니다.
 - c. 기존 보안 그룹 목록에서 보안 그룹을 선택한 다음 검토 후 시작(Review and Launch)을 선택합니다.
7. 인스턴스 시작 검토 페이지에서 시작을 선택합니다.
8. 키 페어에 대한 메시지가 나타나면 기존 키 페어 선택을 선택한 다음 설치할 때 생성한 키 페어를 선택합니다.

Warning

키 페어 없이 계속을 선택하지 마십시오. 키 페어 없이 인스턴스를 시작하면 인스턴스에 연결할 수 없습니다.

준비되면 승인 확인란을 선택한 다음 인스턴스 시작을 선택합니다.

9. 확인 페이지에서 인스턴스가 실행 중인지 확인할 수 있습니다. 인스턴스 보기 선택하여 확인 페이지를 닫고 콘솔로 돌아갑니다.
10. 인스턴스 화면에서 시작 상태를 볼 수 있습니다. 인스턴스를 시작하는 데 약간 시간이 걸립니다. 인스턴스를 시작할 때 초기 상태는 pending입니다. 인스턴스가 시작된 후에는 상태가 running으로 바뀌고 퍼블릭 DNS 이름을 받습니다. (퍼블릭 DNS(IPv4) 열이 숨겨져 있는 경우 페이지 오른쪽 상단 모서리에 있는 열 표시/숨기기(기어 모양 아이콘)를 선택한 다음 퍼블릭 DNS(IPv4)를 선택합니다.)
11. 연결할 수 있도록 인스턴스가 준비될 때까지 몇 분 정도 걸릴 수 있습니다. 인스턴스가 상태 확인을 통과했는지 확인하십시오. 상태 검사 열에서 이 정보를 볼 수 있습니다.

2단계: 인스턴스에 연결

Windows 인스턴스에 연결하려면 최초 관리자 암호를 검색한 다음(아래 2단계 참조) 원격 데스크톱을 사용하여 인스턴스에 연결할 때 이 암호를 지정해야 합니다.

관리자 계정의 이름은 운영 체제의 언어에 따라 다릅니다. 예를 들어 영어는 Administrator, 프랑스어는 Administrateur, 포르투갈어는 Administrador입니다. 자세한 내용은 Microsoft TechNet Wiki의 [Localized Names for Administrator Account in Windows](#)를 참조하십시오.

인스턴스를 도메인에 조인한 경우 AWS Directory Service에서 정의한 도메인 자격 증명을 사용하여 인스턴스에 연결할 수 있습니다. 원격 데스크톱 로그인 화면에서 로컬 컴퓨터 이름과 생성된 암호를 사용하는 대신 관리자의 정규화된 사용자 이름(예: `corp.example.com\Admin`)과 이 계정의 암호를 사용합니다.

Windows Server 운영 체제(OS) 라이선스는 관리 목적으로 두 개의 동시 원격 연결을 허용합니다. Windows 인스턴스 가격에는 Windows Server 라이선스가 포함됩니다. 2개를 초과하는 동시 원격 연결이 필요할 경우, 원격 데스크톱 서비스(RDS) 라이선스를 구매해야 합니다. 제3의 연결을 시도하면 오류가 발생합니다. 자세한 정보는 [Configure the Number of Simultaneous Remote Connections Allowed for a Connection](#)을 참조하십시오.

RDP 클라이언트를 사용하여 Windows 인스턴스에 연결하려면

1. Amazon EC2 콘솔에서 인스턴스를 선택한 다음, [Connect]를 선택합니다.

2. 인스턴스에 연결 대화 상자에서 암호 가져오기를 선택합니다(인스턴스가 시작된 후 몇 분 정도 지나야 암호를 사용할 수 있음).
3. Browse(찾아보기)를 선택하고 인스턴스를 시작할 때 생성한 프라이빗 키 파일을 탐색합니다. 파일을 선택하고 열기를 클릭하여 파일의 전체 내용을 콘텐츠 필드로 복사합니다.
4. [Decrypt Password]를 선택합니다. 콘솔에서는 [Connect To Your Instance] 대화 상자에 해당 인스턴스에 대한 기본 관리자 암호가 표시되어 이전에 표시된 [Get Password]에 대한 링크가 실제 암호로 바뀝니다.
5. 기본 관리자 암호를 기록하거나 클립보드로 복사합니다. 이 암호는 인스턴스에 연결하는 데 필요합니다.
6. [Download Remote Desktop File]을 선택합니다. 브라우저에서 .rdp 파일을 열거나 저장하라는 메시지가 표시됩니다. 어떤 옵션이든 좋습니다. 마쳤으면 [Close]를 선택하여 [Connect To Your Instance] 대화 상자를 닫습니다.
 - .rdp 파일을 연 경우에는 원격 데스크톱 연결 대화 상자가 나타납니다.
 - .rdp 파일을 저장한 경우에는, 다운로드 디렉터리로 이동해 .rdp 파일을 열면 대화 상자가 표시됩니다.
7. 원격 연결 게시자를 알 수 없다는 경고를 받을 수도 있습니다. 계속해서 인스턴스에 연결할 수 있습니다.
8. 관련 메시지가 표시되면 운영 체제 관리자 계정과 이전에 기록하거나 복사한 암호를 사용하여 인스턴스에 로그인합니다. 원격 데스크톱 연결에 관리자 계정이 이미 설정되어 있는 경우에는 다른 계정 사용 옵션을 선택해 사용자 이름과 암호를 수동으로 입력해야 할 수도 있습니다.

Note

때로는 콘텐츠를 복사하고 붙여 넣으면 데이터가 손상될 수 있습니다. 로그인할 때 "Password Failed" 오류가 발생하면 암호를 수동으로 입력해 보십시오.

9. 자체 서명된 인증서의 특성으로 인해, 보안 인증서를 인증할 수 없다는 경고 메시지가 나타날 수도 있습니다. 다음 단계에 따라 원격 컴퓨터의 자격 증명을 확인하거나, 인증서를 신뢰할 경우에는 단순히 [Yes] 또는 [Continue]를 선택하여 계속 진행합니다.
 - a. Windows PC에서 [Remote Desktop Connection]을 사용 중이라면 [View certificate]을 선택합니다. Mac에서 [Microsoft Remote Desktop]을 사용 중이라면 [Show Certificate]을 선택합니다.
 - b. 세부 정보 탭을 선택하고 Windows PC에서는 지문 항목, Mac에서는 SHA1 지문 항목이 나타날 때 까지 아래로 스크롤합니다. 이것은 원격 컴퓨터의 보안 인증서에 대한 고유한 식별자입니다.
 - c. Amazon EC2 콘솔에서 인스턴스를 선택하고 [Actions]를 선택한 다음, [Get System Log]를 선택합니다.
 - d. 시스템 로그 출력에서 RDPCERTIFICATE-THUMBPRINT라는 항목을 확인합니다. 이 값이 인증서의 지문과 일치한다면 원격 컴퓨터의 자격 증명을 확인한 것입니다.
 - e. Windows PC에서 Remote Desktop Connection을 사용 중이라면 [Certificate] 대화 상자로 돌아가서 [OK]를 선택합니다. Mac에서 [Microsoft Remote Desktop]을 사용 중이라면 [Verify Certificate]으로 돌아가서 [Continue]를 선택합니다.
 - f. [Windows] 원격 데스크톱 연결 창에서 예를 선택하여 인스턴스에 연결합니다.

[Mac OS] 메시지가 표시되면 이전에 기록했거나 복사한 기본 관리자 계정과 기본 관리자 암호를 사용하여 로그인합니다. 로그인 화면을 보려면 스페이스를 전환해야 할 수도 있습니다. 스페이스에 대한 자세한 정보는 support.apple.com/en-us/HT204100을 참조하십시오.

- g. 인스턴스에 연결을 시도하는 동안 오류가 발생한 경우 [원격 데스크톱으로 원격 컴퓨터에 연결할 수 없음 \(p. 1181\)](#) 단원을 참조하십시오.

3단계: 인스턴스 정리

이 자습서용으로 생성한 인스턴스와 볼륨을 완료한 후에는 인스턴스를 종료하여 정리해야 합니다. 정리하기 전에 이 인스턴스로 추가 연습을 수행하려는 경우 [다음 단계 \(p. 20\)](#) 단원을 참조하십시오.

Important

인스턴스를 종료하면 인스턴스가 실제로 삭제되므로 인스턴스를 종료한 후에는 인스턴스에 다시 연결할 수 없습니다.

AWS 프리 티어 밖에 있는 인스턴스를 시작한 경우 인스턴스 상태가 `shutting down` 또는 `terminated`로 변경되는 즉시 해당 인스턴스에 대한 요금 발생이 중지됩니다. 나중에 사용하기 위해 인스턴스를 보관하지만 요금이 발생하지 않도록 하려면 지금 인스턴스를 중지한 다음 나중에 다시 시작할 수 있습니다. 자세한 정보는 [인스턴스 중단](#)을 참조하십시오.

인스턴스를 종료하려면

1. 탐색 창에서 인스턴스를 선택합니다. 인스턴스 목록에서 인스턴스를 선택합니다.
2. 작업, 인스턴스 상태, 종료를 차례로 선택합니다.
3. 확인 메시지가 나타나면 예, 종료를 선택합니다.

Amazon EC2가 인스턴스를 종료합니다. 인스턴스는 종료한 후에도 잠시 동안 콘솔에서 표시된 상태로 유지되며, 그 이후 항목이 삭제됩니다.

다음 단계

인스턴스를 시작한 후 다음 연습을 시도할 수 있습니다.

- Run Command를 사용하여 EC2 인스턴스를 원격으로 관리하는 방법을 알아봅니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 Run Command](#)를 참조하십시오.
- 사용량이 프리 티어 한도를 초과하는 경우 알려 주는 CloudWatch 경보를 구성합니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 [청구 경보 생성](#)을 참조하십시오.
- EBS 볼륨을 추가합니다. 자세한 내용은 [Amazon EBS 볼륨 생성 \(p. 948\)](#) 및 [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#) 단원을 참조하십시오.
- WAMP 또는 WIMP 스택을 설치합니다. 자세한 내용은 [자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WAMP 서버 설치 \(p. 1161\)](#) 및 [자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WIMP 서버 설치 \(p. 1163\)](#) 단원을 참조하십시오.

Amazon EC2 Windows 모범 사례

이 모범 사례 목록을 사용하면 Amazon EC2에서의 Windows 실행에 대한 최상의 결과를 얻을 수 있습니다.

Windows 드라이버 업데이트

플랫 전체에 최신 문제 수정 및 성능 개선사항을 적용하려면 모든 Windows EC2 인스턴스에 최신 드라이버를 설치해야 합니다. 인스턴스 유형에 따라 AWS PV, ENA, NVMe 드라이버를 업데이트해야 합니다.

- [Trusted Advisor](#)를 이용해, Amazon EC2 AWS가 제공하는 최신 Windows 드라이버를 항상 적용하십시오.
- [SNS 주제](#)를 이용해 새로 출시된 드라이버에 관한 업데이트를 받으십시오.
- AWS 시스템 관리자 SSM 문서 [AWSSupport-UpgradeWindowsAWSDrivers](#)를 이용해 인스턴스에 업데이트를 쉽게 적용하십시오.

최신 Windows AMI를 사용하여 새 인스턴스 시작

AWS는 최신 OS 패치, 드라이버, 시작 에이전트가 포함된 신규 [Windows AMI](#)를 매월 출시합니다. 새 인스턴스를 시작하거나 자체 사용자 지정 이미지를 만들 때는 최신 AMI를 활용해야 합니다.

- 사용 가능한 최신 AMI를 이용해 이미지를 만드는 방법은 [Systems Manager 파라미터 스토어를 사용한 최신 Windows AMI에 대한 쿼리](#) 단원을 참조하십시오.

マイグ레이션하기 전 시스템/애플리케이션 성능 테스트

엔터프라이즈 애플리케이션을 AWS로 마이그레이션하는 작업은 다양한 변수와 구성을 동반할 수 있습니다. 다음을 보장하라면 EC2 솔루션의 성능을 테스트해야 합니다.

- 인스턴스 크기, 향상된 네트워킹, 테넌시(공유 또는 전용)를 포함한 인스턴스 유형이 올바르게 구성됩니다.
- 인스턴스 토플로지는 워크로드에 적합하며, 필요 시 고성능 기능(전용 테넌시, 배치 그룹, 인스턴스 스토어 볼륨, 베어 메탈)을 활용합니다.

시작 에이전트 업데이트

플랫 전체에 최신 문제 수정을 적용하려면 최신 EC2Launch v2(Windows Server 2008 이상) 에이전트로 업데이트해야 합니다. 업데이트하려면 [최신 버전의 EC2Launch v2 설치](#)의 지침을 참조하세요.

EC2Config(Windows Server 2012 R2 및 이전 버전) 또는 EC2Launch(Windows Server 2016 이상) 에이전트를 계속 사용하려면 플랫 전체에 최신 문제 수정을 적용하세요.

- EC2Config 업데이트 지침은 [최신 EC2Config 설치](#)를 참조하십시오.
- EC2Launch 업데이트 지침은 [최신 EC2Launch 설치](#)를 참조하십시오.

보안

Windows 인스턴스를 보호할 때는 Active Directory 도메인 서비스를 구현하여 확장 가능하고 안전하며 관리하기 쉬운 인프라를 분산 위치에서 사용하는 것이 좋습니다. 또한 AWS 콘솔에서 인스턴스를 시작하거나 AWS CloudFormation 등의 Amazon EC2 프로비저닝 도구를 사용한 후에는 [Microsoft Windows PowerShell DSC](#)와 같은 기본 OS 기능을 사용하여 구성 드리프트가 발생하더라도 구성 상태를 유지하는 것이 좋습니다.

AWS의 Windows 인스턴스는 다음과 같은 고차원적인 모범 사례를 준수해야 합니다.

- **최소 액세스:** 신뢰할 수 있고 예상되는 시스템 및 위치에만 액세스 권한을 부여합니다. 이는 Active Directory, Microsoft 비즈니스 생산성 서버 등 모든 Microsoft 제품 및 원격 데스크톱 서비스, 역방향 프록시 서버, IIS 웹 서버 등의 인프라 서비스에 적용됩니다. Amazon EC2 인스턴스 보안 그룹, 네트워크 ACL(액세스 제어 목록), Amazon VPC 퍼블릭/프라이빗 서브넷과 같은 AWS 기능을 사용하여 아키텍처의 여러 위치에 걸쳐 보안을 계층화합니다. 고객은 Windows 인스턴스 안에서 Windows 방화벽을 사용하여 해당 배포의 심층 방어 전략을 추가로 계층화 할 수 있습니다. 시스템이 설계된 대로 작동하는 데 필요한 OS 구성 요소 및 애플리케이션만 설치하십시오. IIS와 같은 인프라 서비스를 서비스 계정에서 실행하도록 구성하거나, 애플리케이션 폴 ID와 같은 기능을 사용하여 인프라 전체의 리소스에 로컬 또는 원격으로 액세스하도록 구성합니다.
- **최소 권한:** 인스턴스 및 계정이 담당하는 기능을 수행하기 위해 필요한 최소 권한 집합을 결정합니다. 이렇게 정의된 권한만 허용하도록 이러한 서버 및 사용자를 제한합니다. 역할 기반 액세스 제어와 같은 기술을 사용하여 관리 계정의 노출 영역을 줄이고, 작업 수행을 위해 가장 제한된 역할을 만듭니다. NTFS의 EFS(파일 시스템 암호화)와 같은 OS 기능을 사용하여 저장된 중요 데이터를 암호화하고, 그 데이터에 대한 애플리케이션 및 사용자 액세스를 제어합니다.
- **구성 관리:** 바이러스 백신, 맬웨어 방지, 침입 탐지/방지 및 파일 무결성 모니터링이 포함된 호스트 기반 보호 제품군과 최신 보안 패치를 통합하는 기본 서버 구성을 만듭니다. 현재 기록된 기준선으로 각 서버를 평가하여 편차를 식별하고 플래그를 지정합니다. 각 서버가 적절한 로그 및 감사 데이터를 생성하고 안전하게 저장하도록 구성되어 있는지 확인합니다. Windows 인스턴스 업데이트에 대한 자세한 내용은 [Windows 인스턴스 업데이트](#) 단원을 참조하십시오.
- **변경 관리:** 변경 프로세스의 완전 자동화를 목표로 서버 구성 기준의 변경 사항을 제어하는 프로세스를 만듭니다. 또한 Windows PowerShell DSC와 함께 JEA(충분한 관리)를 활용하여 관리 액세스를 최소한의 필수 기능만으로 제한합니다.
- **감사 로그:** Amazon EC2 인스턴스에 대한 액세스 및 모든 변경 사항을 감사하여 서버 무결성을 확인하고 승인된 변경 사항만 수행되도록 합니다. [IIS용 향상된 로깅](#) 등의 기능을 활용하여 기본 로깅 기능을 향상시킵니다. VPC 흐름 로그 등 AWS 기능을 사용할 수 있으며, 허용/거부된 요청 및 API 호출 등 네트워크 액세스를 AWS CloudTrail로 각각 감사할 수도 있습니다.

스토리지

- 운영 체제와 데이터에 대해 별도의 Amazon EBS 볼륨을 사용합니다. 데이터를 포함하는 볼륨이 인스턴스 종료 이후에 지속되는지 확인합니다. 자세한 내용은 [인스턴스 종료 시 Amazon EBS 볼륨 보존 \(p. 458\)](#) 단원을 참조하십시오.
- 인스턴스에서 임시 데이터를 저장하는 데 사용 가능한 인스턴스 스토어를 사용합니다. 인스턴스를 중지하거나 종료하면 인스턴스 스토어에 저장된 데이터가 삭제됩니다. 인스턴스 스토어를 데이터베이스 스토리지용으로 사용하는 경우 내결합성을 보장하는 복제 인자를 가진 클러스터가 있어야 합니다.
- EBS 볼륨 및 스냅샷을 암호화합니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

리소스 관리

- 인스턴스 메타데이터 및 사용자 지정 리소스 태그를 사용하여 AWS 리소스를 추적하고 식별합니다. 자세한 내용은 [인스턴스 메타데이터 및 사용자 데이터 \(p. 576\)](#) 및 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.
- Amazon EC2에 대한 현재 제한을 조회합니다. 실제로 필요할 시점보다 미리 제한 증가를 요청하도록 계획하십시오. 자세한 내용은 [Amazon EC2 서비스 할당량 \(p. 1154\)](#) 단원을 참조하십시오.

백업 및 복구

- [Amazon EBS 스냅샷 \(p. 966\)](#)을 사용하여 EBS 볼륨을 정기적으로 백업하고, 인스턴스에서 [Amazon 마신 이미지\(AMI\) \(p. 24\)](#)를 만들어 추후 인스턴스 시작을 위한 템플릿으로 구성을 저장합니다.
- 애플리케이션의 주요 구성 요소를 여러 가용 영역에 배포하고 데이터를 적절히 복제합니다.
- 인스턴스를 다시 시작할 때 IP 주소를 동적으로 지정하도록 애플리케이션을 설계합니다. 자세한 내용은 [Amazon EC2 인스턴스 IP 주소 지정 \(p. 702\)](#) 단원을 참조하십시오.

- 이벤트 모니터링 및 응답. 자세한 내용은 [Amazon EC2 모니터링 \(p. 648\)](#) 단원을 참조하십시오.
- 장애 조치를 처리할 수 있도록 준비해야 합니다. 기본 솔루션의 경우 네트워크 인터페이스 또는 탄력적 IP 주소를 대체 인스턴스에 수동으로 연결할 수 있습니다. 자세한 내용은 [탄력적 네트워크 인터페이스 \(p. 729\)](#) 단원을 참조하십시오. 자동 솔루션의 경우 Amazon EC2 Auto Scaling을 사용할 수 있습니다. 자세한 내용은 [Amazon EC2 Auto Scaling 사용 설명서](#)를 참조하십시오.
- 장애가 발생할 경우에 대비하여 인스턴스 및 Amazon EBS 볼륨의 복구 프로세스를 정기적으로 테스트합니다.

네트워킹

- 애플리케이션의 TTL(Time-to-Live) 값을 IPv4 및 IPv6의 경우 255로 설정합니다. 더 작은 값을 사용하면 애플리케이션 트래픽이 전송되는 동안 TTL이 만료되어 인스턴스에 연결성 문제가 발생할 수 있습니다.

Amazon 머신 이미지(AMI)

Amazon 머신 이미지(AMI)는 인스턴스를 시작하는 데 필요한 정보를 제공합니다. 인스턴스를 시작할 때 AMI를 지정해야 합니다. 동일한 구성의 인스턴스가 여러 개 필요할 때는 한 AMI에서 여러 인스턴스를 시작할 수 있습니다. 서로 다른 구성의 인스턴스가 필요할 때는 다양한 AMI를 사용하여 인스턴스를 시작하면 됩니다.

AMI는 다음을 포함합니다.

- 1개 이상의 EBS 스냅샷 또는, 인스턴스 저장 지원 AMI의 경우, 인스턴스의 루트 볼륨에 대한 템플릿(예: 운영 체제, 애플리케이션 서버, 애플리케이션)
- AMI를 사용하여 인스턴스를 시작할 수 있는 AWS 계정을 제어하는 시작 권한
- 시작될 때 인스턴스에 연결할 볼륨을 지정하는 블록 디바이스 맵핑

고유 AMI 생성

기존 AMI에서 인스턴스를 시작하고 사용자 지정한 다음, 업데이트된 이 구성을 사용자 지정 AMI로 저장할 수 있습니다. 이 새로운 사용자 지정 AMI에서 인스턴스를 시작하면 해당 AMI를 만들 때 지정한 사용자 정의 값을 포함하게 됩니다.

자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오.

AMI를 범주화하고 관리하기 위해 사용자는 AMI에 사용자 정의 태그를 할당할 수 있습니다. 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

AMI 구입, 공유 및 판매

AMI를 생성한 후 사용자는 AMI를 프라이빗으로 유지하여 자체적으로 사용하거나 특정 AWS 계정 목록과 공유할 수 있습니다. 또한 사용자 정의 AMI를 퍼블릭으로 설정하여 커뮤니티에서 사용되도록 할 수 있습니다. 간단한 몇 단계만 수행하면 간단한 프로세스를 통해 안전하고 사용이 가능하며 보안이 제공되는 퍼블릭 AMI를 구축할 수 있습니다. AMI 사용 및 공유 방법에 대한 자세한 내용은 [공유 AMI \(p. 72\)](#) 단원을 참조하십시오.

Red Hat과 같은 조직의 서비스 계약에 따라 제공되는 AMI를 포함하여 타사에서 AMI를 구입할 수 있습니다. 또한, AMI를 생성한 후 다른 Amazon EC2 사용자에게 판매할 수도 있습니다. AMI 구입 및 판매에 대한 자세한 내용은 [유료 AMI \(p. 80\)](#) 단원을 참조하십시오.

AMI 등록 해제

관련 작업이 완료되면 AMI의 등록을 해제할 수 있습니다. 등록 취소한 AMI에서는 새 인스턴스를 시작할 수 없습니다. 그 AMI에서 시작된 기존 인스턴스에는 영향을 주지 않습니다. 자세한 내용은 [Windows AMI 등록 취소 \(p. 111\)](#) 단원을 참조하십시오.

AWS Windows AMI

AWS는 Windows 플랫폼별 소프트웨어 구성이 포함되고 공개적으로 사용이 가능한 AMI를 제공합니다. 이러한 AMI를 사용하면 Amazon EC2를 사용하는 애플리케이션을 빠르게 구축 및 배포할 수 있습니다. 우선 정

요구 사항을 충족하는 AMI를 선택한 다음 AMI를 사용하여 인스턴스를 시작합니다. 관리자 계정의 암호를 확인한 다음 다른 Windows 서버에서와 마찬가지로 원격 데스크탑 연결을 사용하여 인스턴스에 로그인합니다.

Windows AMI에서 인스턴스를 시작할 때 Windows 인스턴스의 루트 디바이스는 Amazon EBS 볼륨입니다. Windows AMI는 루트 디바이스에 대한 인스턴스 스토어를 지원하지 않습니다.

일부 Windows AMI에는 Microsoft SQL Server 버전(SQL Enterprise Edition, SQL Server Standard, SQL Server Express 또는 SQL Server Web)이 포함됩니다. Microsoft SQL Server 기반 Windows AMI에서 시작한 인스턴스를 데이터베이스 서버로 운영할 수 있습니다. 다른 방법으로, Windows AMI에서 인스턴스를 시작한 다음 인스턴스에서 필요한 데이터베이스 소프트웨어를 설치할 수도 있습니다.

Microsoft는 더 이상 Windows Server 2003을 지원하지 않습니다([Microsoft Windows Server 2003 지원 종료 참조](#)). 지원되는 버전의 Windows Server를 사용하여 새 EC2 인스턴스를 시작하는 것이 좋습니다. 지원되지 않는 Windows Server 버전을 실행하는 기존 EC2 인스턴스가 있는 경우, 지원되는 Windows Server 버전으로 해당 인스턴스를 업그레이드하는 것이 좋습니다. 자세한 내용은 [Amazon EC2 Windows 인스턴스를 새 버전의 Windows Server로 업그레이드 \(p. 611\)](#) 단원을 참조하십시오.

초기 Windows AMI 선택

AWS에서 제공하는 Windows AMI를 보려면 Amazon EC2 콘솔이나 [AWS Marketplace](#)을 사용하십시오. 자세한 내용은 [Windows AMI 찾기 \(p. 68\)](#) 단원을 참조하십시오.

또한 사용자의 Windows 컴퓨터에서 AMI를 생성하는 것도 가능합니다. 자세한 내용은 다음 서비스를 참조하십시오.

- [AWS Server Migration Service](#)
- [VM Import/Export](#)

AMI를 최신으로 유지하기

AWS는 업데이트되고 완전히 패치된 Windows AMI를 Microsoft의 패치 화요일(매달 둘째 주 화요일)을 기준으로 5영업일 내에 제공합니다. 자세한 내용은 [AWS Windows AMI 버전에 관한 세부 정보 \(p. 30\)](#) 단원을 참조하십시오.

AWS Windows AMI에는 생성 당시의 최신 보안 업데이트가 포함되어 있습니다. 자세한 내용은 [Windows 인스턴스 업데이트 \(p. 27\)](#) 단원을 참조하십시오.

가상화 유형

AMI는 PV(반가상화) 또는 HVM(하드웨어 가상 머신)의 두 가지 유형의 가상화를 사용합니다. PV AMI와 HVM AMI의 주요 차이점은 부팅 방법과 더 나은 성능을 위해 특수 하드웨어 확장을 활용할 수 있는지에 있습니다. Windows AMI는 HVM AMI입니다.

HVM AMI는 이미지 루트 블록 디바이스의 마스터 부트 레코드를 실행하여 완벽하게 가상화된 하드웨어 및 부트 세트를 함께 제공합니다. 이 가상화 유형은 운영 체제 미설치 하드웨어에서 실행될 때처럼 가상 머신에서 운영 체제를 수정하지 않고 실행할 수 있습니다. Amazon EC2 호스트 시스템은 게스트에게 제공되는 기본 하드웨어의 일부 또는 모두를 에뮬레이트합니다.

HVM 게스트는 하드웨어 확장을 활용하여 호스트 시스템의 기본 하드웨어에 빠르게 액세스할 수 있습니다. 향상된 네트워킹 및 GPU 처리를 활용하려면 HVM AMI가 필요합니다. 특수 네트워크 및 GPU 디바이스에 대한 명령을 통과하기 위해 OS는 기본 하드웨어 플랫폼에 액세스할 수 있어야 하고, HVM 가상화는 이 액세스 기능을 제공합니다.

이전에는 반가상화 게스트는 I/O용 특수 드라이버를 활용하여 네트워크 및 디스크 하드웨어 에뮬레이트 오버헤드를 방지할 수 있지만, HVM 게스트는 이러한 명령을 에뮬레이트된 하드웨어로 변환해야 했기 때문에,

반가상화 게스트가 HVM 게스트보다 스토리지 및 네트워크 운영 성능이 더 뛰어났습니다. 현재는 HVM 게스트용 PV 드라이버가 제공되므로 Windows 인스턴스에서는 이 드라이버를 사용해 스토리지 및 네트워크 I/O 성능을 높일 수 있습니다. HVM 게스트는 이러한 HVM 기반 PV 드라이버를 사용하여 반가상 게스트와 동일하거나 더 나은 성능을 얻을 수 있습니다.

관리형 AWS Windows AMI

AWS는 여러 가지 버전과 구성의 Windows Server가 포함된 관리형 Amazon 머신 이미지(AMI)를 제공합니다. 일반적으로, AWS Windows AMI는 Microsoft 설치 미디어에서 사용하는 기본 설정으로 구성됩니다. 그러나 사용자 지정도 가능합니다. 예를 들어 AWS Windows AMI에는 다음과 같은 소프트웨어 및 드라이버가 함께 제공됩니다.

- EC2Config 서비스(Windows Server 2012 R2를 통해 제공)
- EC2Launch(Windows Server 2016 이상)
- AWS 시스템 관리자
- AWS CloudFormation
- Windows PowerShell용 AWS 도구
- 네트워크 드라이버(SRIOV, ENA, Citrix PV)
- 스토리지 드라이버(NVMe, AWS PV, Citrix PV)
- 그래픽 드라이버(NVidia GPU, 엘라스틱 GPU)
- 스팟 인스턴스 최대 절전 모드

그 밖의 사용자 지정에 대한 자세한 내용은 [AWS Windows AMI의 구성 변경 \(p. 28\)](#) 단원을 참조하십시오.

목차

- [Windows 인스턴스 업데이트 \(p. 27\)](#)
- [Windows Server 새 버전으로 업그레이드 또는 마이그레이션 \(p. 27\)](#)
- [Windows AMI 알림에 대한 구독 \(p. 27\)](#)
- [AWS Windows AMI의 구성 변경 \(p. 28\)](#)
- [AWS Windows AMI 버전에 관한 세부 정보 \(p. 30\)](#)
 - 공식 AWS Windows AMI에서 다루는 내용 (p. 30)
 - AWS에서 제공할 Windows AMI를 결정하는 방법 (p. 31)
 - 패치, 보안 업데이트 및 AMI ID (p. 31)
 - 연 2회 채널 릴리스 (p. 32)
 - AWS Windows AMI 버전 내역 (p. 32)
 - 2020년 월간 AMI 업데이트(현재 기준) (p. 32)
 - 2019년 월간 AMI 업데이트 (p. 36)
 - 2018년 월간 AMI 업데이트 (p. 42)
 - 2017년 월간 AMI 업데이트 (p. 48)
 - 2016년 월간 AMI 업데이트 (p. 52)
 - 2015년 월간 AMI 업데이트 (p. 56)
 - 2014년 월간 AMI 업데이트 (p. 58)
 - 2013년 월간 AMI 업데이트 (p. 59)
 - 2012년 월간 AMI 업데이트 (p. 61)
 - 2011년 이전 월간 AMI 업데이트 (p. 63)
 - [Windows Server 2016 이상 AMI의 변경 사항 \(p. 63\)](#)
 - [Windows Server 2016 인스턴스에서 Docker 컨테이너 충돌 \(p. 64\)](#)

- Hibernate Agent(2018.03.16 AMI)와 관련된 문제 (p. 64)
- STIG 규정을 준수하는 Amazon EC2 Windows Server AMI (p. 65)
 - 코어 및 기본 운영 체제 (p. 66)
 - Internet Explorer(IE) 11 STIG V1 Release 18 (p. 67)
 - Microsoft .NET Framework 4.0 STIG V1 Release 9: STIGS는 적용되지 않음 (p. 67)
 - Windows 방화벽 STIG V1 Release 7 (p. 67)
 - 버전 기록 (p. 67)

Windows 인스턴스 업데이트

Windows 인스턴스를 실행한 후에 사용자는 그 인스턴스에 업데이트를 설치할 책임이 있습니다. 관심 있는 업데이트만 수동으로 설치하거나, 현재 AWS Windows AMI에서 시작을 하고 새 Windows 인스턴스를 구축할 수 있습니다. 현재 AWS Windows AMI에 대한 자세한 내용은 [Windows AMI 찾기 \(p. 68\)](#) 단원을 참조하십시오.

Note

업데이트 시 인스턴스는 상태 비저장이어야 합니다. 자세한 내용은 [대규모 AWS 인프라 관리하기](#)를 참조하십시오.

Windows 인스턴스에 대해 다음 서비스 또는 애플리케이션에 업데이트를 설치할 수 있습니다.

- Microsoft Windows Server
- Microsoft SQL Server
- Windows PowerShell
- EC2Launch (p. 490)
- EC2Config 서비스 (p. 498)
- AWS 시스템 관리자 SSM 에이전트
- ENA (p. 753)
- NVMe 드라이버 (p. 539)
- PV 드라이버 (p. 528)
- Windows PowerShell용 AWS 도구
- AWS CloudFormation 헬퍼 스크립트

업데이트를 인스톨한 후에 Windows 인스턴스를 재부팅할 수 있습니다. 자세한 내용은 [인스턴스 재부팅 \(p. 452\)](#) 단원을 참조하십시오.

Windows Server 새 버전으로 업그레이드 또는 마이그레이션

Windows 인스턴스를 새 버전의 Windows Server로 업그레이드하거나 마이그레이션하는 방법에 대한 자세한 내용은 [Amazon EC2 Windows 인스턴스를 새 버전의 Windows Server로 업그레이드 \(p. 611\)](#) 단원을 참조하십시오.

Windows AMI 알림에 대한 구독

새로운 AMI가 배포되거나 이전에 배포된 AMI가 프라이빗으로 설정될 때 알림을 받으려면 Amazon SNS를 사용하여 이러한 알림을 구독합니다.

Windows AMI 알림에 대한 구독 방법

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 필요한 경우 탐색 모음에서 리전을 미국 동부(버지니아 북부)로 변경하십시오. 구독을 신청하는 SNS 알림이 이 리전에서 생성되었기 때문에 이 리전을 사용해야 합니다.
3. 탐색 창에서 구독을 선택합니다.
4. Create subscription을 선택합니다.
5. 구독 생성 대화 상자에서 다음과 같이 수행합니다.

a. 주제 ARN에 다음 Amazon 리소스 이름(ARN) 중 하나를 복사하여 붙여넣습니다.

- **arn:aws:sns:us-east-1:801119661308:ec2-windows-ami-update**
- **arn:aws:sns:us-east-1:801119661308:ec2-windows-ami-private**

AWS GovCloud(미국)의 경우:

arn:aws-us-gov:sns:us-gov-west-1:077303321853:ec2-windows-ami-update

- b. 프로토콜에서 이메일을 선택합니다.
 - c. 엔드포인트에서 알림을 받을 이메일 주소를 입력합니다.
 - d. Create subscription을 선택합니다.
6. 그러면 AWS Notification - Subscription Confirmation의 제목을 가진 확인 이메일을 받게 됩니다. 이메일을 열고 구독 확인을 선택하여 구독을 완료합니다.

Windows AMI가 배포될 때마다, ec2-windows-ami-update 주제의 구독자에게 알림이 발송됩니다. 배포된 Windows AMI가 프라이빗으로 설정될 때마다, ec2-windows-ami-private 주제의 구독자에게 알림이 발송됩니다. 이런 알림을 더 이상 받지 않기를 원하는 경우, 다음 절차를 수행해서 구독을 해제하십시오.

Windows AMI 알림에 대한 구독 해제 방법

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 필요한 경우 탐색 모음에서 리전을 미국 동부(버지니아 북부)로 변경하십시오. SNS 알림이 이 리전에 생성되었기 때문에 이 리전을 사용해야 합니다.
3. 탐색 창에서 구독을 선택합니다.
4. 구독을 선택한 후 작업, 구독 삭제를 선택합니다. 확인 메시지가 나타나면 삭제를 선택합니다.

AWS Windows AMI의 구성 변경

각각의 AWS Windows AMI가 다음과 같이 변경됩니다.

정리 및 준비

변경 사항	적용 대상
대기 중인 파일 이름 변경 또는 재부팅 작업이 있는지 확인 후 필요하면 재부팅	모든 AMI
.dmp 파일 삭제	모든 AMI
로그 삭제(이벤트 로그, 시스템 관리자, EC2Config)	모든 AMI
sysprep용 임시 폴더 및 파일 삭제	모든 AMI
최근 내역 정리(시작 메뉴, Windows 탐색기 등)	Windows Server 2012 R2 및 이전

변경 사항	적용 대상
바이러스 스캔 수행	모든 AMI
대기열의 .NET 어셈블리 사전 컴파일링(sysprep 전)	모든 AMI
Windows 유지 관리 도구 실행	Windows Server 2012 R2 및 이상
Internet 탐색기의 기본값 복원	모든 AMI
EC2Config의 기본값 복원	Windows Server 2012 R2 및 이전
다음 번 시작할 때 실행되도록 EC2Launch 설정	Windows Server 2016 이상
Windows 바탕 화면 재설정	모든 AMI
sysprep 실행	모든 AMI

설치 및 구성

변경 사항	적용 대상
Amazon EC2 Windows Guide 링크 추가	모든 AMI
확장 탑재 지점에 인스턴스 스토리지 볼륨 연결	모든 AMI
현재 Windows PowerShell용 AWS 도구 설치	모든 AMI
현재 AWS CloudFormation 헬퍼 스크립트 설치	모든 AMI
현재 EC2Config 및 SSM 에이전트 설치	Windows Server 2012 R2 및 이전
현재 EC2Launch 및 SSM 에이전트 설치	Windows Server 2016 이상
현재 AWS PV, ENA, NVMe 드라이버 설치	Windows Server 2008 R2 및 이상
현재 SRIOV 드라이버 설치	Windows Server 2012 R2 및 이상
현재 Citrix PV 드라이버 설치	Windows Server 2008 SP2 및 이전
현재 EC2WinUtil 드라이버 설치	Windows Server 2008 R2 및 이상
PowerShell 2.0 및 3.0 설치	Windows Server 2008 SP2 및 R2
Microsoft SQL Server가 설치되어 있는 경우:	모든 AMI
<ul style="list-style-type: none"> • 서비스 팩 설치 • 자동으로 시작되도록 구성 • SysAdmin 역할에 BUILTIN\Administrators 추가 • TCP 포트 1433 및 UDP 포트 1434 열기 	
다음 핫픽스를 적용합니다.	Windows Server 2008 SP2 및 R2
<ul style="list-style-type: none"> • MS15-011 • KB2582281 • KB2634328 • KB2800213 • KB2922223 	

변경 사항	적용 대상
<ul style="list-style-type: none"> KB2394911 KB2780879 	
ICMP 트래픽의 방화벽 통과 허용	Windows Server 2012 R2 및 이전
파일 및 프린터 공유 활성화	Windows Server 2012 R2 및 이전
Internet 탐색기의 RunOnce 비활성화	모든 AMI
원격 PowerShell 활성화	모든 AMI
시스템 볼륨에서 다음과 같이 폴더를 구성합니다.	모든 AMI
<ul style="list-style-type: none"> Windows Server 2019 - 시스템에서 관리 Windows Server 2016 - 시스템에서 관리 Windows Server 2012 R2 - 초기 크기 및 최대 크기 8GB Windows Server 2012 및 이전 버전 - 초기 크기 512MB, 최대 크기 8GB 	
사용 가능한 경우 z:에 시스템 관리 폴더를 추가로 구성합니다.	Windows Server 2012 R2 및 이전
최대 절전 모드 비활성화 및 최대 절전 모드 파일 삭제	모든 AMI
성능 극대화를 위해 성능 옵션 설정	모든 AMI
전원 설정을 고성능으로 설정	모든 AMI
화면 보호기 암호 비활성화	모든 AMI
RealTimeUniversal 레지스트리 키를 설정합니다	모든 AMI
시간대를 UTC로 설정	모든 AMI
Windows 업데이트 및 알림 비활성화	모든 AMI
대기 중인 업데이트가 없을 때까지 Windows 업데이트 실행 및 재부팅	모든 AMI
어떤 전력 체계에서도 디스플레이가 꺼지지 않도록 설정합니다.	모든 AMI
PowerShell 실행 정책을 "무제한(Unrestricted)"으로 설정합니다.	모든 AMI

AWS Windows AMI 버전에 관한 세부 정보

공식 AWS Windows AMI에서 다루는 내용

AWS에서는 모든 지원 Windows 운영 체제에 대한 다양한 구성이 포함된 AMI를 제공합니다. 각 이미지의 경우 AWS는 다음을 수행합니다.

- 모든 Microsoft 권장 Windows 보안 패치를 설치합니다. 매달 Microsoft 패치가 제공된 직후 이미지를 릴리스합니다.
- 네트워크 및 디스크 드라이버를 포함한 AWS 하드웨어의 최신 드라이버, 문제 해결을 위한 EC2WinUtil, 그리고 선택한 AMI의 GPU 드라이버를 설치합니다.
- Server 2012 R2 및 이전의 경우 [EC2 Config \(p. 496\)](#) 또는 Server 2016 이후의 경우 [EC2 Launch \(p. 489\)](#)와 같은 AWS 헬퍼 소프트웨어를 포함합니다.

- AWS Time Service (p. 558)를 사용하여 Windows 시간을 구성합니다.
- 디스플레이를 절대 꺼지지 않도록 설정하기 위해 모든 전력 체계를 변경합니다.
- 경미한 버그 수정. 일반적으로 한 줄로 된 레지스트리를 변경하여 AWS에서 성능이 개선된 기능을 활성화 또는 비활성화합니다.

위에서 나열된 것과는 별개로 AMI를 가급적 기본 설치와 동일하게 유지합니다. 따라서 "스톡" PowerShell 또는 .NET 프레임워크 버전을 기본 설정하고, Windows 기능을 설치하지 않으며, 일반적으로 AMI를 변경하지 않습니다.

AWS에서 제공할 Windows AMI를 결정하는 방법

각 AMI는 일반 퍼블릭 릴리스 전에 광범위한 테스트를 받습니다. 주기적으로 AMI 오퍼링을 간소화하여 고객 선택을 간소화하고 비용을 절감합니다.

- 새 OS 릴리스에 대해 새 AMI 오퍼링이 생성됩니다. "Base", "Core/Container" 및 "SQL Express/Standard/Web/Enterprise" 오퍼링을 영어와 기타 널리 사용되는 언어로 이용할 수 있습니다. Base 제품군과 Core 제품군의 주된 차이는 Base의 경우 데스크톱/GUI를 가지고 있는 반면, Core 제품군은 PowerShell 명령 줄 전용이라는 것입니다. Windows Server Core에 대한 자세한 내용은 <https://docs.microsoft.com/en-us/windows-server/administration/server-core/what-is-server-core>를 참조하십시오.
- 새로운 AMI 오퍼링이 생성되어 새 플랫폼을 지원합니다. 예를 들어 딥 러닝 및 "Nvidia" AMI가 생성되어 GPU 기반 인스턴스 유형(P2 및 P3, G2 및 G3 등)을 사용하는 고객을 지원합니다.
- 인기가 덜한 AMI는 제거되는 경우가 있습니다. 특정 AMI가 전체 수명 기간 도중 몇 차례만 실행되는 경우 더 널리 사용되는 옵션을 위해 이를 제거합니다.

확인하고자 하는 AMI 변형이 있는 경우 클라우드 지원을 통해 티켓을 발행하거나 [설립된 채널 중 하나](#)를 통해 피드백을 제공하여 알려주십시오.

패치, 보안 업데이트 및 AMI ID

AWS는 업데이트되고 완전히 패치된 Windows AMI를 Microsoft의 패치 화요일(매달 둘째 주 화요일)을 기준으로 5영업일 내에 제공합니다. 새로운 AMI는 Amazon EC2 콘솔의 이미지 페이지를 통해 즉시 제공됩니다. 새로운 AMI는 릴리스 후 며칠 내에 실행 인스턴스 마법사의 AWS Marketplace 및 빠른 시작 탭에서 제공됩니다.

Note

최신 Windows Server 2019 AMI에서 시작한 인스턴스에서 "Some settings are managed by your organization.(일부 설정이 사용자의 조직에 의해 관리됩니다.)"이라는 Windows 업데이트 대화 상자 메시지를 표시할 수 있습니다. 이 메시지는 Windows Server 2019의 변경 사항으로 인해 나타나며 업데이트 설정 관리 기능 또는 Windows 업데이트 동작에는 영향을 주지 않습니다.
이 경고를 제거하려면 "[Some settings are managed by your organization.\(일부 설정이 사용자의 조직에 의해 관리됩니다.\)](#)"을 참조하십시오.

고객들이 항상 최신 보안 업데이트를 갖출 수 있도록 하기 위해 AWS는 Windows AMI를 3개월 동안 제공합니다. 새로운 Windows AMI가 릴리스된 후 10일 안에 AWS는 3개월 이상 된 Windows AMI를 프라이빗으로 설정합니다. AMI가 프라이빗으로 변경된 뒤 콘솔에서 해당 AMI에서 시작한 인스턴스를 살펴보면 AMI ID 필드에 "Cannot load detail for ami-xxxxx. You may not be permitted to view it"이라고 표시됩니다. 그래도 AWS CLI 또는 AWS SDK를 사용하여 AMI ID를 검색할 수 있습니다.

각 릴리스에서 Windows AMI는 새로운 AMI ID를 가집니다. 따라서 ID보다는 이름으로 최신 AWS Windows AMI의 위치를 지정하는 스크립트를 작성하는 것이 권장됩니다. 자세한 내용은 다음 예를 참조하십시오.

- Get-EC2ImageByName (Windows PowerShell용 AWS 도구)
- 시스템 관리자 파라미터 스토어를 사용하여 최신 Windows AMI 쿼리

- 연습: Amazon 마신 이미지 ID 조회 (AWS Lambda, AWS CloudFormation)

연 2회 채널 릴리스

AWS는 AWS의 규모, 성능 및 탄력성을 [Windows Server의 연 2회 채널 릴리스 버전](#)으 새 기능과 결합하는 Windows Server 연 2회 채널 릴리스를 제공합니다.

AWS Windows AMI 버전 내역

다음 표에서는 AWS Windows AMI의 각 릴리스에 대한 변경 사항을 요약합니다. 일부 변동사항은 모든 AWS Windows AMI에 적용되지만 다른 일부 변동사항은 해당 AMI의 하위세트에만 적용된다는 점을 참고하시기 바랍니다.

내용

- [2020년 월간 AMI 업데이트\(현재 기준\) \(p. 32\)](#)
- [2019년 월간 AMI 업데이트 \(p. 36\)](#)
- [2018년 월간 AMI 업데이트 \(p. 42\)](#)
- [2017년 월간 AMI 업데이트 \(p. 48\)](#)
- [2016년 월간 AMI 업데이트 \(p. 52\)](#)
- [2015년 월간 AMI 업데이트 \(p. 56\)](#)
- [2014년 월간 AMI 업데이트 \(p. 58\)](#)
- [2013년 월간 AMI 업데이트 \(p. 59\)](#)
- [2012년 월간 AMI 업데이트 \(p. 61\)](#)
- [2011년 이전 월간 AMI 업데이트 \(p. 63\)](#)

이러한 AMI에 포함된 구성 요소에 대한 자세한 내용은 다음을 참조하십시오.

- [EC2Config 버전 기록 \(p. 509\)](#)
- [EC2Launch 버전 기록 \(p. 494\)](#)
- [시스템 관리자 SSM 에이전트 출시 정보](#)
- [Amazon ENA 드라이버 버전 \(p. 754\)](#)
- [AWS PV 드라이버 패키지 내역 \(p. 524\)](#)

2020년 월간 AMI 업데이트(현재 기준)

Microsoft 업데이트에 대한 자세한 내용은 [2020년 소프트웨어 업데이트 서비스 및 Windows Server 업데이트 서비스 변경 내용에 대한 설명](#)을 참조하십시오.

릴리스	변경
2020.9.9	<p>모든 AMI</p> <ul style="list-style-type: none">Windows 보안 업데이트(2020년 9월 8일까지 반영)AWS PV 드라이버 버전 8.3.4AWS ENA 버전 2.2.0AWS Tools for Windows PowerShell 버전 3.15.1110SQL CU 설치됨<ul style="list-style-type: none">SQL_2016_SP2: CU14SQL_2019: CU7

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
AWS Windows AMI 버전에 관한 세부 정보

릴리스	변경
	<p>Windows Server 2016/2019/1809/1903/1909/2004 AMI</p> <ul style="list-style-type: none">• EC2Launch 버전 1.3.2003155• SSM 에이전트 버전 2.3.1319.0 <p>EC2LaunchV2_Preview AMI</p> <ul style="list-style-type: none">• EC2Launch v2 버전 2.0.124
2020.8.12	<p>모든 AMI</p> <ul style="list-style-type: none">• Windows 보안 업데이트(2020년 8월 11일까지 반영)• AWS Tools for Windows PowerShell 버전 3.15.1084• G3 AMI: NVIDIA GRID 버전 451.48• EC2Launch v2 Preview AMI: EC2Launch 버전 2.0.104• SQL CU 설치됨<ul style="list-style-type: none">• SQL_2019: CU6• Amazon에서 2020년 5월 13일 이전에 발표한 Windows AMI의 이전 버전은 비공개로 만들어졌습니다.
2020.7.15	<p>모든 AMI</p> <ul style="list-style-type: none">• Windows 보안 업데이트(2020년 7월 14일까지 반영)• AWS Tools for Windows PowerShell 버전 3.15.1064• ENA 버전 2.1.5• SQL CU 설치됨<ul style="list-style-type: none">• SQL_2017: CU21• SQL_2019: CU5• Amazon에서 2020년 4월 15일 이전에 발표한 Windows AMI의 이전 버전은 비공개로 만들어졌습니다.

릴리스	변경
2020.7.01	<p>Amazon Machine Image(AMI)의 새 버전이 릴리스되었습니다. 이러한 이미지는 EC2Launch v2를 포함하고 있으며, AWS에서 현재 제공하는 모든 Windows AMI에 올해 말 기본적으로 포함되기 전에 새 시작 에이전트의 기능 미리 보기 역할을 합니다. EC2 Image Builder와 같은 일부 SSM 문서 및 종속 서비스는 EC2 Launch v2를 지원하기 위해 업데이트가 필요할 수 있습니다. 이러한 업데이트는 앞으로 몇 주 이내에 진행될 예정입니다. 이러한 이미지는 프로덕션 환경에서는 사용하지 않는 것이 좋습니다. EC2Launch v2에 대한 자세한 내용은 https://aws.amazon.com/about-aws/whats-new/2020/07/introducing-ec2-launch-v2-simplify-customizing-windows-instances/ 및 https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2launch-v2.html에서 확인할 수 있습니다. 현재 모든 Windows Server AMI는 향후 몇 달 동안 현재 시작 에이전트인 EC2Config(Server 2012 RTM/2012 R2) 또는 EC2Launch v1(Server 2016 이상)에 대한 변경 없이 계속 제공됩니다. 가까운 시일 내에 AWS에서 현재 제공하는 모든 Windows Server AMI는 월별 릴리스의 일부로 EC2Launch v2를 사용하도록 마이그레이션될 예정입니다. EC2LaunchV2_Preview AMI는 매월 업데이트되며 이 마이그레이션이 이루어질 때까지 계속 사용할 수 있습니다.</p> <p>신규 Windows AMI</p> <ul style="list-style-type: none"> EC2LaunchV2_Preview-Windows_Server-2004-English-Core-Base-2020.06.30 EC2LaunchV2_Preview-Windows_Server-2019-English-Full-Base-2020.06.30 EC2LaunchV2_Preview-Windows_Server-2019-English-Core-Base-2020.06.30 EC2LaunchV2_Preview-Windows_Server-2016-English-Full-Base-2020.06.30 EC2LaunchV2_Preview-Windows_Server-2016-English-Core-Base-2020.06.30 EC2LaunchV2_Preview-Windows_Server-2012_R2_RTM-English-Full-Base-2020.06.30 EC2LaunchV2_Preview-Windows_Server-2012_R2_RTM-English-Core-Base-2020.06.30 EC2LaunchV2_Preview-Windows_Server-2012_RTM-English-Full-Base-2020.06.30 EC2LaunchV2_Preview-Windows_Server-2019-English-Full-SQL_2019_Express-2020.06.30 EC2LaunchV2_Preview-Windows_Server-2016-English-Full-SQL_2017_Express-2020.06.30
2020.6.10	<p>모든 AMI</p> <ul style="list-style-type: none"> Windows 보안 업데이트(2020년 6월 9일까지 반영) AWS Tools for Windows PowerShell 버전 3.15.1034 cfn-init 버전 1.4.33 SQL CU 설치됨: SQL_2016_SP2: CU13
2020.5.27	<p>신규 Windows AMI</p> <ul style="list-style-type: none"> Windows_Server-2004-English-Core-Base-2020.05.27 Windows_Server-2004-English-Core-ContainersLatest-2020.05.27
2020.5.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Windows 보안 업데이트(2020년 5월 12일까지 반영) AWS Tools for Windows PowerShell 버전 3.15.1013 EC2Launch 버전 1.3.2003150

릴리스	변경
2020.4.15	모든 AMI <ul style="list-style-type: none"> • Windows 보안 업데이트(2020년 4월 14일까지 반영) • AWS Tools for Windows PowerShell 버전 3.15.998 • EC2Config 버전 4.9.4222 • EC2Launch 버전 1.3.2003040 • SSM 에이전트 버전 2.3.842.0 • SQL CU 설치됨: <ul style="list-style-type: none"> • SQL_2017: CU 20 • SQL_2019: CU 4
2020.3.18	Windows Server 2019 AMI <p>초기 OS 부팅 후 예상 시간 내에 BITS(Background Intelligent Transfer Service)가 시작되지 않아 잠재적으로 시간 초과, 이벤트 로그의 BITS 오류, 초기 부팅 후 BITS 관련 cmdlet이 빠르게 호출되지 않는 문제 등이 발생하는 2020.3.11 릴리스에서 발견된 간헐적 문제가 해결되었습니다. 다른 Windows Server AMI는 이 문제의 영향을 받지 않으며, 최신 버전은 2020.03.11로 유지됩니다.</p>
2020.3.11	모든 AMI <ul style="list-style-type: none"> • Windows 보안 업데이트(2020년 3월 10일까지 반영) • AWS Tools for Windows PowerShell 버전 3.15.969 • EC2Config 버전 4.9.4122 • EC2Launch 버전 1.3.2002730 • SSM 에이전트 버전 2.3.814.0 • SQL CU 설치됨: <ul style="list-style-type: none"> • SQL_2016_SP2: CU 12 • SQL_2017: CU 19 • SQL_2019: CU 2는 SQL 에이전트의 알려진 문제로 인해 적용되지 않습니다. • 서버 코어 1909 및 1903에 대한 OOB(Out Of Band) 보안 업데이트 (KB4551762) 가 CVE-2020-0796을 완화하기 위해 적용되었습니다. 다른 Windows Server 버전은 이 문제의 영향을 받지 않습니다. 자세한 내용은 https://portal.msrg.microsoft.com/ko-KR/security-guidance/advisory/CVE-2020-0796을 참조하십시오.

릴리스	변경
2020.2.12	<p>모든 AMI</p> <ul style="list-style-type: none"> 2020년 2월 11일 현재 Windows 보안 업데이트 AWS Tools for Windows PowerShell 버전 3.15.945 Intel SRIOV 드라이버 업데이트 <ul style="list-style-type: none"> 2019/1903/1909: 버전 2.1.185.0 2016/1809: 버전 2.1.186.0 2012 R2: 버전 1.2.199.0 SQL CU 설치됨: <ul style="list-style-type: none"> SQL_2019: CU 1 SQL_2017: CU 18 SQL_2016_SP2: CU 11 <p>Microsoft Windows Server 2008 SP2 및 Windows Server 2008 R2</p> <p>Windows Server 2008 SP2와 Window Server 2008 R2는 2020년 1월 14일에 EOS(지원 종료)에 도달했으며 더 이상 Microsoft에서 정기적으로 보안 업데이트가 제공되지 않습니다. AWS는 더 이상 Windows Server 2008 SP2 또는 Windows Server 2008 R2 AMI를 게시하거나 배포하지 않습니다. 계정에 있는 기존 2008 SP2/R2 인스턴스와 사용자 지정 AMI는 영향을 받지 않으며 EOS 날짜 이후에도 계속 사용할 수 있습니다.</p> <p>업그레이드 및 가져오기 옵션과 2020년 1월 14일을 기준으로 더 이상 게시되지 않는 AMI의 전체 목록을 포함하여 AWS의 Microsoft 서비스 종료에 대한 자세한 내용은 Microsoft 제품에 대한 EOS(지원 종료)를 참조하십시오.</p>
2020.1.15	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2020년 1월 14일까지 반영) Windows PowerShell용 AWS 도구 버전 3.15.925 ENI 버전 2.1.4 <p>Microsoft Windows Server 2008 SP2 및 Windows Server 2008 R2</p> <p>Windows Server 2008 SP2와 Window Server 2008 R2는 2020년 1월 14일에 EOS(지원 종료)에 도달했으며 더 이상 Microsoft에서 정기적으로 보안 업데이트가 제공되지 않습니다. AWS는 더 이상 Windows Server 2008 SP2 또는 Windows Server 2008 R2 AMI를 게시하거나 배포하지 않습니다. 계정에 있는 기존 2008 SP2/R2 인스턴스와 사용자 지정 AMI는 영향을 받지 않으며 EOS 날짜 이후에도 계속 사용할 수 있습니다.</p> <p>업그레이드 및 가져오기 옵션과 2020년 1월 14일을 기준으로 더 이상 게시되지 않는 AMI의 전체 목록을 포함하여 AWS의 Microsoft 서비스 종료에 대한 자세한 내용은 Microsoft 제품에 대한 EOS(지원 종료)를 참조하십시오.</p>

2019년 월간 AMI 업데이트

Microsoft 업데이트에 대한 자세한 내용은 [2019년 소프트웨어 업데이트 서비스 및 Windows Server 업데이트 서비스 변경 내용에 대한 설명](#)을 참조하십시오.

릴리스	변경
2019.12.16	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2019년 12월 10일까지 반영) Windows PowerShell용 AWS 도구 버전 3.15.903 <p>Microsoft Windows Server 2008 SP2 및 Windows Server 2008 R2</p> <p>Microsoft는 2020년 1월 14일에 Windows Server 2008 SP2 및 Windows Server 2008 R2에 대한 일반 지원을 종료합니다. 이 날짜가 되면 AWS는 더 이상 Windows Server 2008 SP2 또는 Windows Server 2008 R2 AMI를 게시하거나 배포하지 않습니다. 계정에 있는 기존 2008 SP2/R2 인스턴스와 사용자 지정 AMI는 영향을 받지 않으며 서비스 종료(EOS) 날짜 이후에도 계속 사용할 수 있습니다.</p> <p>업그레이드 및 가져오기 옵션을 포함하여 AWS의 Microsoft EOS에 대한 자세한 내용과 2020년 1월 14일부터 게시 또는 배포가 중단되는 전체 AMI 목록은 Microsoft 제품에 대한 EOS(지원 종료)를 참조하십시오.</p>
2019.11.13	<p>모든 AMI</p> <ul style="list-style-type: none"> PowerShell 버전 3.15.876용 AWS 도구 Windows 보안 업데이트(2019년 11월 12일까지 반영) EC2 Config 버전 4.9.3865 EC2 Launch 버전 1.3.2002240 SSM 에이전트 v2.3.722.0 <p>AMI의 이전 버전들은 프라이빗으로 표시되었습니다.</p> <p>신규 Windows AMI</p> <ul style="list-style-type: none"> Windows_Server-1909-English-Core-Base-2019.11.13 Windows_Server-1909-English-Core-ContainersLatest-2019.11.13 Windows_Server-2016-English-Full-SQL_2019_Enterprise-2019.11.13 Windows_Server-2016-English-Full-SQL_2019_Express-2019.11.13 Windows_Server-2016-English-Full-SQL_2019_Standard-2019.11.13 Windows_Server-2016-English-Full-SQL_2019_Web-2019.11.13 Windows_Server-2019-English-Full-SQL_2019_Enterprise-2019.11.13 Windows_Server-2019-English-Full-SQL_2019_Express-2019.11.13 Windows_Server-2019-English-Full-SQL_2019_Standard-2019.11.13 Windows_Server-2019-English-Full-SQL_2019_Web-2019.11.13
2019.11.05	<p>신규 Windows AMI</p> <p>신규 SQL AMI 사용 가능:</p> <ul style="list-style-type: none"> Windows_Server-2016-English-Full-SQL_2019_Enterprise-2019.11.05 Windows_Server-2016-English-Full-SQL_2019_Express-2019.11.05 Windows_Server-2016-English-Full-SQL_2019_Standard-2019.11.05 Windows_Server-2016-English-Full-SQL_2019_Web-2019.11.05 Windows_Server-2019-English-Full-SQL_2019_Enterprise-2019.11.05 Windows_Server-2019-English-Full-SQL_2019_Express-2019.11.05

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
AWS Windows AMI 버전에 관한 세부 정보

릴리스	변경
	<ul style="list-style-type: none">Windows_Server-2019-English-Full-SQL_2019_Standard-2019.11.05Windows_Server-2019-English-Full-SQL_2019_Web-2019.11.05
2019.10.09	<p>모든 AMI</p> <ul style="list-style-type: none">Windows PowerShell용 AWS 도구 버전 3.15.846Windows 보안 업데이트(2019년 10월 8일까지 반영)Windows Defender 플랫폼은 레지스트리를 제거하여 현재 블록 및 업데이트 블록을 업데이트한다. 자세한 내용은 https://support.microsoft.com/en-us/help/4513240/sfc-incorrectly-flags-windows-defender-ps-files-as-corrupted 단원을 참조하십시오. <p>신규 Windows AMI</p> <p>신규 ECS 최적화 AMI 사용 가능</p> <ul style="list-style-type: none">Windows_Server-2019-English-Core-ECS_Optimized-2019.10.09
2019.09.12	<p>신규 Windows AMI</p> <ul style="list-style-type: none">amzn2-ami-hvm-2.0.20190618-x86_64-gp2-mono <p>Amazon Linux 2에서 NET 애플리케이션을 LTS(Long Term Support)로 실행하도록 .NET Core 2.2, Mono 5.18 및 PowerShell 6.2가 사전 설치되었습니다.</p>

릴리스	변경
2019.09.11	<p>모든 AMI</p> <ul style="list-style-type: none">AWS PV 드라이버 버전 8.3.2AWS NVMe 드라이버 버전 1.3.2Windows PowerShell용 AWS 도구 버전 3.15.8262019 AMI에 대해 모든 OS 2012 RTM에서 NLA가 활성화됩니다.고객이 보고한 문제로 인해 인텔 82599 VF 드라이버를 버전 2.0.210.0(Server 2016) 또는 버전 2.1.138.0(Server 2019)로 복구했습니다. 이러한 문제는 인텔과 협력하십시오.Windows 보안 업데이트(2019년 9월 10일까지 반영)최신 클라이언트에서 도입된 SFC 오류로 인해 Windows Defender 플랫폼 업데이트가 레지스트리를 통해 차단되었습니다. 패치가 제공될 시점에 다시 활성화될 예정입니다. https://support.microsoft.com/en-us/help/4513240/sfc-incorrectly-flags-windows-defender-ps-files-as-corrupted를 참조하십시오. 플랫폼 업데이트 차단: HKLM\SOFTWARE\Microsoft\Windows Defender\Miscellaneous Configuration \PreventPlatformUpdate type=DWORD, value=1 <p>AMI의 이전 버전들은 프라이빗으로 표시되었습니다.</p> <p>신규 Windows AMI</p> <p>제공되는 신규 STIG 규격 AMI:</p> <ul style="list-style-type: none">Windows_Server-2012-R2-English-STIG-FullWindows_Server-2012-R2-English-STIG-CoreWindows_Server-2016-English-STIG-FullWindows_Server-2016-English-STIG-CoreWindows_Server-2019-English-STIG-FullWindows_Server-2019-English-STIG-Core <p>Windows Server 2008 R2 SP1</p> <p>Microsoft 확장 보안(ESU) 업데이트에 필요한 다음 업데이트가 포함되어 있습니다.</p> <ul style="list-style-type: none">KB4490628KB4474419KB4516655 <p>Windows Server 2008 SP2</p> <p>Microsoft 확장 보안(ESU) 업데이트에 필요한 다음 업데이트가 포함되어 있습니다.</p> <ul style="list-style-type: none">KB4493730KB4474419KB4517134

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 AWS Windows AMI 버전에 관한 세부 정보

릴리스	변경
	<p>Note</p> <p>기본 RDP 보안 태세를 강화하기 위해 이제 모든 2012 RTM, 2012 R2 및 2016 AMI에서 NLA가 활성화됩니다. NLA는 2019 AMI에서 활성화 상태로 유지됩니다.</p>
2019.08.16	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2019년 8월 13일까지 반영) CVE-2019-1181, CVE-2019-1182, CVE-2019-1222 및 CVE-2019-1226를 처리하는 KB가 포함됩니다. EC2Config 버전 4.9.3519 SSM 에이전트 버전 2.3.634.0 PowerShell 버전 3.15.802용 AWS 도구 업데이트에서 도입된 SFC 오류로 인해 Windows Defender 플랫폼 업데이트가 레지스트리를 통해 차단되었습니다. 새 패치가 사용 가능하면 업데이트가 다시 활성화됩니다. <p>Note</p> <p>기본 RDP 보안 태세를 강화하기 위해 9월부터 모든 2012 RTM, 2012 R2 및 2016 AMI에서 NLA가 활성화됩니다.</p>
2019.07.19	<p>신규 Windows AMI</p> <ul style="list-style-type: none"> Windows_Server-2016-English-Full-ECS_Optimized-2019.07.19 Windows_Server-2019-English-Full-ECS_Optimized-2019.07.19
2019.07.12	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2019년 7월 9일까지 반영)
2019.06.12	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2019년 6월 11일까지 반영) AWS SDK 버전 3.15.756 AWS PV 드라이버 버전 8.2.7 AWS NVMe 드라이버 버전 1.3.1 다음 "P3" AMI는 "Tesla" AMI로 이름이 변경됩니다. 이 AMI는 Tesla 드라이버를 사용하는 모든 GPU 지원 AWS 인스턴스를 지원합니다. P3 AMI는 이 릴리스 이후 더 이상 업데이트되지 않으며 정기적인 주기로 제거됩니다. <ul style="list-style-type: none"> Windows_Server-2012-R2_RTM-English-P3-2019.06.12가 Windows_Server-2012-R2_RTM-English-Tesla-2019.06.12로 대체 Windows_Server-2016-English-P3-2016.06.12가 Windows_Server-2016-English-Tesla-2019.06.12로 대체 <p>신규 Windows AMI</p> <ul style="list-style-type: none"> Windows_Server-2019-English-Tesla-2019.06.12 <p>AMI의 이전 버전들은 프라이빗으로 표시되었습니다.</p>

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 AWS Windows AMI 버전에 관한 세부 정보

릴리스	변경
2019.05.21	Windows Server, 버전 1903 <ul style="list-style-type: none"> 현재 AMI를 사용할 수 있습니다.
2019.05.15	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2019년 5월 14일까지 반영) EC2Config 버전 4.9.3429 SSM 에이전트 버전 2.3.542.0 AWS SDK 버전 3.15.735
2019.04.26	모든 AMI <ul style="list-style-type: none"> 인스턴스 최초 시작 시 인스턴스 손상이 발생하고 Windows에서 "Please wait for the User Profile Service(사용자 프로파일 서비스를 기다려 주십시오)"라는 메시지가 표시될 수도 있는 엣지 케이스를 해결하고자 SQL로 Windows Server 2019의 AMI를 수정함
2019.04.21	모든 AMI <ul style="list-style-type: none"> AWS PV 드라이버를 버전 8.3.0에서 버전 8.2.6으로 를 백
2019.04.10	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2019년 4월 9일까지 반영) AWS SDK 버전 3.15.715 AWS PV 드라이버 버전 8.3.0 EC2Launch 버전 1.3.2001360 <p>신규 Windows AMI</p> <ul style="list-style-type: none"> Windows_Server-2016-English-Full-SQL_2012_SP4_Standard-2019.04.10 Windows_Server-2016-English-Full-SQL_2014_SP3_Standard-2019.04.10 Windows_Server-2016-English-Full-SQL_2014_SP3_Enterprise-2019.04.10
2019.03.13	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2019년 3월 12일까지 반영) AWS SDK 버전 3.15.693 EC2Launch 버전 1.3.2001220 Deep Learning 및 P3 AMI용 NVIDIA Tesla 드라이버 버전 412.29(https://nvidia.custhelp.com/app/answers/detail/a_id/4772) <p>AMI의 이전 버전들은 프라이빗으로 표시되었습니다.</p>

릴리스	변경
2019.02.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2019년 2월 12일까지 반영) SSM 에이전트 버전 2.3.444.0 AWS SDK 버전 3.15.666 EC2Launch 버전 1.3.2001040 EC2Config 버전 4.9.3289 AWS PV 드라이버 8.2.6 EBS NVMe 도구 <p>서비스 팩 2가 포함된 SQL 2014 및 서비스 팩 1이 포함된 SQL 2016은 이 릴리스 이후로는 더 이상 업데이트되지 않습니다.</p>
2019.02.09	<p>모든 AMI</p> <ul style="list-style-type: none"> Windows AMI가 업데이트되었습니다. 새 AMI는 다음 날짜 버전에서 확인할 수 있습니다. <p>11월 "2018.11.29"</p> <p>12월 "2018.12.13"</p> <p>1월 "2019.02.09"</p> <p>AMI의 이전 버전들은 프라이빗으로 표시되었습니다.</p>
2019.01.10	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2019년 1월 10일까지 반영) SSM 에이전트 버전 2.3.344.0 AWS SDK 버전 3.15.647 EC2Launch 버전 1.3.2000930 EC2Config 버전 4.9.3160 <p>SQL Server가 있는 모든 AMI</p> <ul style="list-style-type: none"> 최근 누적 업데이트

2018년 월간 AMI 업데이트

Microsoft 업데이트에 대한 자세한 내용은 [2018년 소프트웨어 업데이트 서비스 및 Windows Server 업데이트 서비스 변경 내용에 대한 설명](#)을 참조하십시오.

릴리스	변경
2018.12.12	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 12월 12일까지 반영) SSM 에이전트 버전 2.3.274.0 AWS SDK 버전 3.15.629

릴리스	변경
	<ul style="list-style-type: none"> • EC2Launch 버전 1.3.2000760 신규 Windows AMI <ul style="list-style-type: none"> • Windows_Server-2012-R2_RTM-Japanese-64Bit-SQL_2014_SP3_Standard-2018.12.12 • Windows_Server-2012-R2_RTM-Japanese-64Bit-SQL_2014_SP3_Express-2018.12.12 • Windows_Server-2012-R2_RTM-English-64Bit-SQL_2014_SP3_Enterprise-2018.12.12 • Windows_Server-2012-R2_RTM-English-64Bit-SQL_2014_SP3_Standard-2018.12.12 • Windows_Server-2012-R2_RTM-English-64Bit-SQL_2014_SP3_Express-2018.12.12 • Windows_Server-2012-R2_RTM-English-64Bit-SQL_2014_SP3_Web-2018.12.12 • Windows_Server-2012-RTM-Japanese-64Bit-SQL_2014_SP3_Express-2018.12.12 • Windows_Server-2012-RTM-Japanese-64Bit-SQL_2014_SP3_Standard-2018.12.12 • Windows_Server-2012-RTM-English-64Bit-SQL_2014_SP3_Express-2018.12.12 • Windows_Server-2012-RTM-English-64Bit-SQL_2014_SP3_Web-2018.12.12 • Windows_Server-2012-R2_RTM-Japanese-64Bit-SQL_2016_SP2_Web-2018.12.12 • Windows_Server-2012-R2_RTM-Japanese-64Bit-SQL_2016_SP2_Express-2018.12.12 • Windows_Server-2012-R2_RTM-English-64Bit-SQL_2016_SP2_Express-2018.12.12 • Windows_Server-2012-R2_RTM-English-64Bit-SQL_2016_SP2_Web-2018.12.12 • Windows_Server-2012-R2_RTM-English-64Bit-SQL_2016_SP2_Standard-2018.12.12 • Windows_Server-2012-R2_RTM-English-64Bit-SQL_2016_SP2_Web-2018.12.12 • Windows_Server-2012-R2_RTM-English-64Bit-SQL_2016_SP2_Standard-2018.12.12 • Windows_Server-2016-Korean-Full-SQL_2016_SP2_Standard-2018.12.12 • Windows_Server-2016-Japanese-Full-SQL_2016_SP2_Enterprise-2018.12.12 • Windows_Server-2016-Japanese-Full-SQL_2016_SP2_Web-2018.12.12 • Windows_Server-2016-English-Full-SQL_2016_SP2_Web-2018.12.12 • Windows_Server-2016-Japanese-Full-SQL_2016_SP2_Standard-2018.12.12 • Windows_Server-2016-English-Full-SQL_2016_SP2_Express-2018.12.12 • Windows_Server-2016-English-Full-SQL_2016_SP2_Standard-2018.12.12 • Windows_Server-2016-English-Core-SQL_2016_SP2_Enterprise-2018.12.12 • Windows_Server-2016-English-Core-SQL_2016_SP2_Web-2018.12.12 • Windows_Server-2016-English-Core-SQL_2016_SP2_Express-2018.12.12 • Windows_Server-2016-English-Core-SQL_2016_SP2_Standard-2018.12.12

릴리스	변경
	<ul style="list-style-type: none"> • Windows_Server-2016-Japanese-Full-SQL_2016_SP2_Standard-2018.12.12 • Windows_Server-2016-Korean-Full-SQL_2016_SP2_Standard-2018.12.12 • Windows_Server-2019-Spanish-Full-Base-2018.12.12 • Windows_Server-2019-Japanese-Full-Base-2018.12.12 • Windows_Server-2019-Portuguese_Portugal-Full-Base-2018.12.12 • Windows_Server-2019-Chinese_Traditional-Full-Base-2018.12.12 • Windows_Server-2019-Italian-Full-Base-2018.12.12 • Windows_Server-2019-Swedish-Full-Base-2018.12.12 • Windows_Server-2019-English-Core-Base-2018.12.12 • Windows_Server-2019-Hungarian-Full-Base-2018.12.12 • Windows_Server-2019-Polish-Full-Base-2018.12.12 • Windows_Server-2019-Turkish-Full-Base-2018.12.12 • Windows_Server-2019-Korean-Full-Base-2018.12.12 • Windows_Server-2019-Dutch-Full-Base-2018.12.12 • Windows_Server-2019-German-Full-Base-2018.12.12 • Windows_Server-2019-Russian-Full-Base-2018.12.12 • Windows_Server-2019-Czech-Full-Base-2018.12.12 • Windows_Server-2019-English-Full-Base-2018.12.12 • Windows_Server-2019-French-Full-Base-2018.12.12 • Windows_Server-2019-Portuguese_Brazil-Full-Base-2018.12.12 • Windows_Server-2019-Chinese_Simplified-Full-Base-2018.12.12 • Windows_Server-2019-English-Full-HyperV-2018.12.12 • Windows_Server-2019-English-Full-ContainersLatest-2018.12.12 • Windows_Server-2019-English-Core-ContainersLatest-2018.12.12 • Windows_Server-2019-English-Full-SQL_2017_Enterprise-2018.12.12 • Windows_Server-2019-English-Full-SQL_2017_Standard-2018.12.12 • Windows_Server-2019-English-Full-SQL_2017_Web-2018.12.12 • Windows_Server-2019-English-Full-SQL_2017_Express-2018.12.12 • Windows_Server-2019-English-Full-SQL_2016_SP2_Enterprise-2018.12.12 • Windows_Server-2019-English-Full-SQL_2016_SP2_Standard-2018.12.12 • Windows_Server-2019-English-Full-SQL_2016_SP2_Web-2018.12.12 • Windows_Server-2019-English-Full-SQL_2016_SP2_Express-2018.12.12 <p style="margin-top: 20px;">업데이트된 Linux AMI</p> <ul style="list-style-type: none"> • amzn2-ami-hvm-2.0.20180622.1-x86_64-gp2-dotnetcore-2018.12.12
2018.11.28	<p>모든 AMI</p> <ul style="list-style-type: none"> • SSM 에이전트 버전 2.3.235.0 • 디스플레이를 절대 꺼지지 않도록 설정하기 위해 모든 전력 체계를 변경합니다.

릴리스	변경
2018.11.20	<p>Windows_Server-2016-English-Deep-Learning</p> <p>Windows_Server-2016-English-Deep-Learning</p> <ul style="list-style-type: none"> • TensorFlow 버전 1.12 • MXNet 버전 1.3 • NVIDIA 버전 392.05
2018.11.19	<p>모든 AMI</p> <ul style="list-style-type: none"> • Microsoft 보안 업데이트(2018년 11월 19일까지 반영) • AWS SDK 버전 3.15.602.0 • SSM 에이전트 버전 2.3.193.0 • EC2Config 버전 4.9.3067 • 인텔 Chipset INF 구성을 통해 새로운 인스턴스 유형을 지원합니다 <p>Windows Server, 버전 1809</p> <ul style="list-style-type: none"> • 현재 AMI를 사용할 수 있습니다.
2018.10.14	<p>모든 AMI</p> <ul style="list-style-type: none"> • Microsoft 보안 업데이트(2018년 10월 9일까지 반영) • Windows PowerShell용 AWS 도구 버전 3.3.365.0 • CloudFormation 버전 1.4.31 • AWS PV 드라이버 버전 8.2.4 • AWS PCI 직렬 드라이버 버전 1.0.0.0(베어 메탈 인스턴스에서 Windows 2008R2 및 2012 지원) • ENA 드라이버 버전 1.5.0 <p>Nano Server용 Microsoft Windows Server 2016 Datacenter 및 Standard 버전</p> <p>Microsoft는 2018년 4월 10일자로 Nano Server용 Microsoft Windows Server 2016 Datacenter 및 Standard Edition 설치 옵션에 대한 주류 지원을 종료했습니다.</p>
2018.09.15	<p>모든 AMI</p> <ul style="list-style-type: none"> • Microsoft 보안 업데이트(2018년 9월 12일까지 반영) • Windows PowerShell용 AWS 도구 버전 3.3.343 • EC2Launch 버전 1.3.2000430 • AWS NVMe 드라이버 버전 1.3.0 • EC2 WinUtil 드라이버 버전 2.0.0 <p style="color: red;">Microsoft Windows Server 2016 Base Nano</p> <p>Windows_Server-2016-English-Nano-Base의 모든 퍼블릭 버전에 대한 액세스 권한은 2018년 9월에 제거됩니다. Nano Server를 컨테이너로 시작하는 것에 관한 세부 정보를 포함해 Nano Server 수명 주기에 관한 추가 정보는 https://docs.microsoft.com/en-us/windows-server/get-started/nano-in-semi-annual-channel에서 얻을 수 있습니다.</p>

릴리스	변경
2018.08.15	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 8월 14일까지 반영) Windows PowerShell용 AWS 도구 버전 3.3.335 이제 AMI는 시간 동기화를 위해 IP 169.254.169.123에서 Amazon의 NTP 서비스를 사용하도록 기본 설정되어 있습니다. 자세한 내용은 Amazon Windows AMI에 대한 기본 NTP 설정 (p. 558)을 참조하십시오. <p style="color: red;"><a>Microsoft Windows Server 2016 Base Nano</p> <p>Windows_Server-2016-English-Nano-Base의 모든 퍼블릭 버전에 대한 액세스 권한은 2018년 9월에 제거됩니다. Nano Server를 컨테이너로 시작하는 것에 관한 세부 정보를 포함해 Nano Server 수명 주기에 관한 추가 정보는 https://docs.microsoft.com/en-us/windows-server/get-started/nano-in-semi-annual-channel에서 얻을 수 있습니다.</p>
2018.07.11	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 7월 10일까지 반영) EC2Config 버전 4.9.2756 SSM 에이전트 2.2.800.0
2018.06.22	<p>Windows Server 2008 R2</p> <ul style="list-style-type: none"> 인스턴스를 이전 세대에서 현재 세대로 변경할 경우 2018.06.13 AMI와 발생하는 문제를 해결합니다(예를 들어, M4에서 M5로 변경).
2018.06.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 6월 12일까지 반영) EC2Config 버전 4.9.2688 SSM 에이전트 2.2.619.0 Windows PowerShell용 AWS 도구 3.3.283.0 AWS NVMe 드라이버 1.2.0 AWS PV 드라이버 8.2.3
2018.05.09	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 5월 9일까지 반영) EC2Config 버전 4.9.2644 SSM 에이전트 2.2.493.0 Windows PowerShell용 AWS 도구 3.3.270.0 <p>Windows Server 버전 1709 및 Windows Server 버전 1803</p> <ul style="list-style-type: none"> 현재 AMI를 사용할 수 있습니다. 자세한 내용은 Amazon EC2용 Windows Server 버전 1709 및 1803 AMI 발표를 참조하십시오.

릴리스	변경
2018.04.11	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 4월 10일까지 반영) EC2Config 버전 4.9.2586 SSM 에이전트 2.2.392.0 Windows PowerShell용 AWS 도구 3.3.256.0 AWS CloudFormation 템플릿 1.4.30 시리얼 INF 및 intel Chipset INF 구성을 통해 새로운 인스턴스 유형을 지원합니다 <p>SQL Server 2017</p> <ul style="list-style-type: none"> 누적 업데이트 5(CU5) <p>SQL Server 2016 SP1</p> <ul style="list-style-type: none"> 누적 업데이트 8(CU8)
2018.03.24	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 3월 13일까지 반영) EC2Config 버전 4.9.2565 SSM 에이전트 2.2.355.0 Windows PowerShell용 AWS 도구 3.3.245.0 AWS PV 드라이버 8.2 AWS ENA 드라이버 1.2.3.0 Amazon EC2 Hibernate Agent 1.0(2018.03.16 AMI 릴리스의 2.1.0에서 롤백) AWS EC2WinUtilDriver 1.0.1(문제 해결용) <p>Windows Server 2016</p> <ul style="list-style-type: none"> EC2Launch 1.3.2000080
2018.03.16	<p>AWS는 Amazon EC2 Hibernate Agent에 대한 구성에서 따옴표로 끝지 않은 경로로 인한 문제 때문에 2018.03.16 날짜의 Windows AMI를 모두 삭제했습니다. 자세한 내용은 Hibernate Agent(2018.03.16 AMI)와 관련된 문제 (p. 64) 단원을 참조하십시오.</p>
2018.03.06	<p>모든 AMI</p> <ul style="list-style-type: none"> AWS PV 드라이버 8.2.1
2018.02.23	<p>모든 AMI</p> <ul style="list-style-type: none"> AWS PV 드라이버 7.4.6(2018.02.13 AMI 릴리스의 8.2에서 롤백)

릴리스	변경
2018.02.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 2월 13일까지 반영) EC2Config 버전 4.9.2400 SSM 에이전트 2.2.160.0 Windows PowerShell용 AWS 도구 3.3.225.1 AWS PV 드라이버 8.2 AWS ENA 드라이버 1.2.3.0 AWS NVMe 드라이버 1.0.0.146 Amazon EC2 HibernateAgent 1.0.0 <p>Windows Server 2016</p> <ul style="list-style-type: none"> EC2Launch 1.3.740
2018.01.12	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 1월 9일까지 반영)
2018.01.05	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2018년 1월까지 반영) Spectre 및 Meltdown 공격에 대한 완화를 활성화할 수 있는 레지스트리 설정 Windows PowerShell용 AWS 도구 3.3.215 EC2Config 버전 4.9.2262

2017년 월간 AMI 업데이트

Microsoft 업데이트에 대한 자세한 내용은 [2017년 소프트웨어 업데이트 서비스 및 Windows Server 업데이트 서비스 변경 내용에 대한 설명](#)을 참조하십시오.

릴리스	변경
2017.12.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 12월 12일까지 반영) EC2Config 버전 4.9.2218 AWS CloudFormation 템플릿 1.4.27 AWS NVMe 드라이버 1.02 SSM 에이전트 2.2.93.0 Windows PowerShell용 AWS 도구 3.3.201
2017.11.29	<p>모든 AMI</p> <ul style="list-style-type: none"> Windows Backup과의 호환성 문제로 인해 2017.11.18 및 2017.11.19에 포함된 VSS(Volume Shadow Copy Service)에서 제거된 구성 요소입니다.
2017.11.19	<p>모든 AMI</p> <ul style="list-style-type: none"> EC2 Hibernate Agent 1.0 (스팟 인스턴스에 최대 절전 모드 지원)

릴리스	변경
2017.11.18	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 11월까지 반영) EC2Config 버전 4.9.2218 SSM 에이전트 2.2.64.0 Windows PowerShell용 AWS 도구 3.3.182 ENA(Elastic Network Adapter) 드라이버 1.08 (2017.10.13 AMI 릴리스의 1.2.2에서 둘째) Amazon AMI Systems Manager 파라미터 스토어를 사용한 최신 Windows AMI에 대한 쿼리 단원을 참조하십시오. <p>Windows Server 2016</p> <ul style="list-style-type: none"> EC2Launch 1.3.640
2017.10.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 10월 11일까지 반영) EC2Config 버전 4.9.2188 SSM 에이전트 2.2.30.0 AWS CloudFormation 템플릿 1.4.24 ENA(Elastic Network Adapter) 드라이버 1.2.2. (Windows Server 2008 R2부터 Windows Server 2016)
2017.10.04	<p>Microsoft SQL Server</p> <p>Windows Server 2016 with Microsoft SQL Server 2017 AMI는 현재 모든 리전에서 퍼블릭입니다.</p> <ul style="list-style-type: none"> Windows_Server-2016-English-Full-SQL_2017_Enterprise-2017.10.04 Windows_Server-2016-English-Full-SQL_2017_Standard-2017.10.04 Windows_Server-2016-English-Full-SQL_2017_Web-2017.10.04 Windows_Server-2016-English-Full-SQL_2017_Express-2017.10.04 <p>Microsoft SQL Server 2017은 다음 기능을 지원합니다.</p> <ul style="list-style-type: none"> Python(ML 및 AI) 및 R 언어 지원이 포함된 Machine Learning 서비스 자동 데이터베이스 튜닝 클러스터링 가능성 그룹 Red Hat Enterprise Linux(RHEL), SUSE Linux Enterprise Server(SLES) 및 Ubuntu에서 실행됩니다. 자세한 내용은 Linux에서 SQL Server에 대한 설치 지침이라는 Microsoft 문서를 참조하십시오. Amazon Linux에서는 지원되지 않습니다. Windows-Linux cross-OS 마이그레이션 다시 시작 가능한 온라인 인덱스 다시 빌드 향상된 적응형 쿼리 처리 그래프 데이터 지원

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 AWS Windows AMI 버전에 관한 세부 정보

릴리스	변경
2017.09.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 9월 13일까지 반영) EC2Config 버전 4.9.2106 SSM 에이전트 2.0.952.0 Windows PowerShell용 AWS 도구 3.3.143 AWS CloudFormation 템플릿 1.4.21
2017.08.09	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 8월 9일까지 반영) EC2Config 버전 4.9.2016 SSM 에이전트 2.0.879.0 <p>Windows Server 2012 R2</p> <ul style="list-style-type: none"> 내부 오류로 인해 이러한 AMI가 이전 버전의 Windows PowerShell용 AWS 도구, 3.3.58.0과 함께 배포되었습니다.
2017.07.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 7월 13일까지 반영) EC2Config 버전 4.9.1981 SSM 에이전트 2.0.847.0 <p>Windows Server 2016</p> <ul style="list-style-type: none"> intel SRIOV 드라이버 2.0.210.0
2017.06.14	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 6월 14일까지 반영) Windows 업데이트에서 설치된 NET Framework 4.7의 업데이트 PowerShell Stop-Computer cmdlet을 사용하여 "권한 없음" 오류를 해결할 Microsoft 업데이트 자세한 내용은 Microsoft 사이트에서 Privilege not held error를 참조하십시오. EC2Config 버전 4.9.1900 SSM 에이전트 2.0.805.0 Windows PowerShell용 AWS 도구 3.3.99.0 몰입형 Internet Explorer 대신 데스크톱용 Internet Explorer 11이 기본 브라우저입니다. <p>Windows Server 2016</p> <ul style="list-style-type: none"> EC2Launch 1.3.610
2017.05.30	암호 생성과 관련된 문제를 해결하기 위해 Windows_Server-2008-SP2-English-32Bit-Base-2017.05.10 AMI가 Windows_Server-2008-SP2-English-32Bit-Base-2017.05.30 AMI로 업데이트되었습니다.

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 AWS Windows AMI 버전에 관한 세부 정보

릴리스	변경
2017.05.22	<p>일부 로그를 정리한 후 Windows_Server-2016-English-Full-Base-2017.05.10 AMI가 Windows_Server-2016-English-Full-Base-2017.05.22 AMI로 업데이트되었습니다.</p>
2017.05.10	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 5월 9일까지 반영) AWS PV 드라이버 v7.4.6 Windows PowerShell용 AWS 도구 3.3.83.0 <p>Windows Server 2016</p> <ul style="list-style-type: none"> SSM 에이전트 2.0.767
2017.04.12	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 4월 11일까지 반영) Windows PowerShell용 AWS 도구 3.3.71.0 AWS CloudFormation 템플릿 1.4.18 <p>Windows Server 2003부터 Windows Server 2012까지</p> <ul style="list-style-type: none"> EC2Config 버전 4.9.1775 SSM 에이전트 2.0.761.0 <p>Windows Server 2016</p> <ul style="list-style-type: none"> SSM 에이전트 2.0.730.0
2017.03.15	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2017년 3월 14일까지 반영) 현재 Windows PowerShell용 AWS 도구 현재 AWS CloudFormation 템플릿 <p>Windows Server 2003부터 Windows Server 2012까지</p> <ul style="list-style-type: none"> EC2Config 버전 4.7.1631 SSM 에이전트 2.0.682.0 <p>Windows Server 2016</p> <ul style="list-style-type: none"> SSM 에이전트 2.0.706.0 EC2Launch v1.3.540
2017.02.21	<p>Microsoft는 2월에 월간 패치나 보안 업데이트를 릴리스하지 않을 것이라고 최근 발표했습니다. 모든 2월 패치 및 보안 업데이트는 3월 업데이트에 포함될 예정입니다.</p> <p>Amazon Web Services는 2월에 업데이트된 Windows Server AMI를 릴리스하지 않았습니다.</p>

릴리스	변경
2017.01.11	<p>모든 AMI</p> <ul style="list-style-type: none">Microsoft 보안 업데이트(2017년 1월 10일까지 반영)현재 Windows PowerShell용 AWS 도구현재 AWS CloudFormation 템플릿 <p>Windows Server 2003부터 Windows Server 2012까지</p> <ul style="list-style-type: none">EC2Config 버전 4.2.1442SSM 에이전트 2.0.599.0

2016년 월간 AMI 업데이트

Microsoft 업데이트에 대한 자세한 내용은 [2016년 소프트웨어 업데이트 서비스 및 Windows Server 업데이트 서비스 변경 내용에 대한 설명](#)을 참조하십시오.

릴리스	변경
2016.12.14	<p>모든 AMI</p> <ul style="list-style-type: none">Microsoft 보안 업데이트(2016년 12월 13일까지 반영)현재 Windows PowerShell용 AWS 도구 <p>Windows Server 2003부터 Windows Server 2012까지</p> <ul style="list-style-type: none">EC2Config 버전 4.1.1396 릴리스ENA(Elastic Network Adapter) 드라이버 1.0.9.0(Windows Server 2008 R2만 해당) <p>Windows Server 2016</p> <p>모든 리전에서 새로운 AMI를 사용할 수 있습니다.</p> <ul style="list-style-type: none">Windows_Server-2016-English-Core-Base <p>Microsoft SQL Server</p> <p>이제 모든 리전에서 최신 서비스 팩이 포함된 모든 Microsoft SQL Server AMI를 사용할 수 있습니다. 이 새로운 AMI는 앞으로 기존 SQL Service Pack AMI를 대체합니다.</p> <ul style="list-style-type: none">Windows_Server-2008-R2_SP1-English-64Bit-SQL_2012_SP3_edition-2016.12.14Windows_Server-2012-RTM-English-64Bit-SQL_2012_SP3_edition-2016.12.14Windows_Server-2012-R2_RTM-English-64Bit-SQL_2014_SP2_edition-2016.12.14Windows_Server-2012-RTM-English-64Bit-SQL_2014_SP2_edition-2016.12.14

릴리스	변경
	<ul style="list-style-type: none"> Windows_Server-2012-R2_RTM-English-64Bit-SQL_2016_SP1_<i>edition</i>-2016.12.14 Windows_Server-2016-English-Full-SQL_2016_SP1_<i>edition</i>-2016.12.14 <p>SQL Server 2016 SP1은 주요 릴리스입니다. 전에는 Enterprise 버전에서만 제공되었던 다음 기능들을 이제 SQL Server 2016 SP1으로 Standard, Web, Express 버전에서도 사용할 수 있습니다.</p> <ul style="list-style-type: none"> 행 수준 보안 동적 데이터 마스킹 데이터 캡처 변경 데이터베이스 스냅샷 컬럼 스토어 분할 압축 인 메모리 OLTP 항상 암호화
2016.11.23	<p>Windows Server 2003부터 Windows Server 2012까지</p> <ul style="list-style-type: none"> EC2Config 버전 4.1.1378 릴리스 이번 달에 AMI가 출시되었습니다. 앞으로 EC2Config 서비스를 이용해 부팅 시간 구성을 처리하고 SSM 에이전트를 이용해 AWS 시스템 관리자 Run Command 및 Config 요청을 처리하십시오. EC2Config는 더 이상 시스템 관리자 Run Command 및 State Manager 요청을 처리하지 않습니다. 최신 EC2Config 설치 관리자는 SSM 에이전트를 EC2Config 서비스와 나란히 설치합니다. 자세한 내용은 EC2Config 및 AWS 시스템 관리자 (p. 500) 단원을 참조하십시오.
2016.11.09	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 11월 8일까지 반영) Windows 2008 R2 및 그 이후 버전용 AWS PV 드라이버 버전 7.4.3.0 릴리스 현재 Windows PowerShell용 AWS 도구
2016.10.18	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 10월 12일까지 반영) 현재 Windows PowerShell용 AWS 도구 <p>Windows Server 2016</p> <ul style="list-style-type: none"> Windows Server 2016용 AMI 릴리스됨. 이러한 AMI에는 중요한 변경 사항이 포함되어 있습니다. 예를 들어, 이러한 AMI에는 EC2Config 서비스가 포함되지 않습니다. 자세한 내용은 Windows Server 2016 이상 AMI의 변경 사항 (p. 63) 단원을 참조하십시오.

릴리스	변경
2016.9.14	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 9월 13일까지 반영) 현재 Windows PowerShell용 AWS 도구 AMI Windows_Server-2012-RTM-Japanese-64Bit-SQL_2008_R3_SP2_Standard에서 Windows_Server-2012-RTM-Japanese-64Bit-SQL_2008_R2_SP3_Standard로 이름 변경
2016.8.26	2016년 8월 11일을 기준으로 알려진 문제를 해결하기 위해 모든 Windows Server 2008 R2 AMI가 업데이트되었습니다. 2016년 8월 25일부터 새 AMI가 적용됩니다.
2016.8.11	모든 AMI <ul style="list-style-type: none"> Ec2Config v3.19.1153 Microsoft 보안 업데이트(2016년 8월 10일까지 반영) MS15-124에 대해 Internet Explorer에서 기능을 강화하는 레지스트리 키 User32 예외 핸들러가 활성화되었습니다 <p>Windows Server 2008 R2, Windows Server 2012 RTM, Windows Server 2012 R2</p> <ul style="list-style-type: none"> ENA(Elastic Network Adapter) 드라이버 1.0.8.0 ENA AMI 속성이 활성 상태로 설정되었습니다 알려진 문제로 인해 Windows Server 2008 R2용 AWS PV 드라이버가 이번 달에 다시 릴리스되었습니다. 이 문제로 인해 Windows Server 2008 R2 AMI가 제거되었습니다.
2016.8.2	7월 Windows Server 2008 R2 AMI는 AWS PV 드라이버에서 발견된 문제 때문에 모두 제거되고 2016.06.15자 AMI로 룰백되었습니다. AWS PV 드라이버 문제는 수정되었습니다. 8월 AMI 릴리스는 수정된 AWS PV 드라이버 및 7월/8월 Windows 업데이트가 포함된 Windows Server 2008 R2 AMI입니다.
2016.7.26	모든 AMI <ul style="list-style-type: none"> Ec2Config v3.18.1118 2016.07.13자 AMI에는 보안 패치가 누락되었습니다. AMI는 다시 패치되었습니다. 패치 설치가 성공적으로 진행되는지 확인하기 위한 추가 프로세스가 마련되었습니다.
2016.7.13	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 7월까지 반영) 현재 Windows PowerShell용 AWS 도구 AWS PV Driver 7.4.2.0 업데이트 Windows Server 2008 R2용 AWS PV 드라이버

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 AWS Windows AMI 버전에 관한 세부 정보

릴리스	변경
2016.6.16	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 6월까지 반영) 현재 Windows PowerShell용 AWS 도구 EC2Config 서비스 버전 3.17.1032 <p>Microsoft SQL Server</p> <ul style="list-style-type: none"> Microsoft SQL Server 2016의 64비트 버전을 포함한 10가지 AMI를 발표했습니다. Amazon EC2 콘솔을 사용하는 경우 이미지, AMI, 퍼블릭 이미지로 이동한 다음 검색 창에 다음을 입력합니다. Windows_Server-2012-R2_RTM-English-64Bit-SQL_2016_Standard. 자세한 내용은 MSDN의 SQL Server 2016의 새 기능을 참조하십시오.
2016.5.11	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 5월까지 반영) 현재 Windows PowerShell용 AWS 도구 EC2Config 서비스 버전 3.16.930 MS15-011 Active Directory 패치가 설치됨 <p>Windows Server 2012 R2</p> <ul style="list-style-type: none"> intel SRIOV 드라이버 1.0.16.1
2016.4.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 4월까지 반영) 현재 Windows PowerShell용 AWS 도구 EC2Config 서비스 버전 3.15.880
2016.3.9	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 3월까지 반영) 현재 Windows PowerShell용 AWS 도구 EC2Config 서비스 버전 3.14.786
2016.2.10	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 2월까지 반영) 현재 Windows PowerShell용 AWS 도구 EC2Config 서비스 버전 3.13.727
2016.1.25	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2016년 1월까지 반영) 현재 Windows PowerShell용 AWS 도구 EC2Config 서비스 버전 3.12.649
2016.1.5	<p>모든 AMI</p> <ul style="list-style-type: none"> 현재 Windows PowerShell용 AWS 도구

2015년 월간 AMI 업데이트

Microsoft 업데이트에 대한 자세한 내용은 [2015년 소프트웨어 업데이트 서비스 및 Windows Server 업데이트 서비스 변경 내용에 대한 설명](#)을 참조하십시오.

릴리스	변경
2015.12.15	모든 AMI <ul style="list-style-type: none">Microsoft 보안 업데이트(2015년 12월까지 반영)현재 Windows PowerShell용 AWS 도구
2015.11.11	모든 AMI <ul style="list-style-type: none">Microsoft 보안 업데이트(2015년 11월까지 반영)현재 Windows PowerShell용 AWS 도구EC2Config 서비스 버전 3.11.521최신 버전으로 업데이트된 CFN 에이전트
2015.10.26	기본 AMI의 부팅 볼륨 크기를 35GB가 아닌 30GB로 수정
2015.10.14	모든 AMI <ul style="list-style-type: none">Microsoft 보안 업데이트(2015년 10월까지 반영)EC2Config 서비스 버전 3.10.442현재 Windows PowerShell용 AWS 도구모든 SQL에 대해 SQL 서비스 팩을 최신 버전으로 업데이트이벤트 로그의 오래된 항목 제거최신 서비스 팩을 반영하기 위해 AMI 이름이 변경되었습니다. 예를 들어, Server 2012 및 SQL 2014 Standard를 지원하는 최신 AMI 이름은 "Windows_Server-2012-RTM-English-64Bit-SQL_2014_SP1_Standard-2015.10.26"이며, "Windows_Server-2012-RTM-English-64Bit-SQL_2014_RTM_Standard-2015.10.26"이 아닙니다.
2015.9.9	모든 AMI <ul style="list-style-type: none">Microsoft 보안 업데이트(2015년 9월까지 반영)EC2Config 서비스 버전 3.9.359현재 Windows PowerShell용 AWS 도구현재 AWS CloudFormation 헬퍼 스크립트
2015.8.18	모든 AMI <ul style="list-style-type: none">Microsoft 보안 업데이트(2015년 8월까지 반영)EC2Config 서비스 버전 3.8.294현재 Windows PowerShell용 AWS 도구 <p>Windows Server 2012 및 Windows Server 2012 R2의 AMI만</p> <ul style="list-style-type: none">AWS PV 드라이버 7.3.2
2015.7.21	모든 AMI <ul style="list-style-type: none">Microsoft 보안 업데이트(2015년 7월까지 반영)

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 AWS Windows AMI 버전에 관한 세부 정보

릴리스	변경
	<ul style="list-style-type: none"> • EC2Config 서비스 버전 3.7.308 • 현재 Windows PowerShell용 AWS 도구 • 일관성을 위한 SQL 이미지의 수정된 AMI 설명
2015.6.10	<p>모든 AMI</p> <ul style="list-style-type: none"> • Microsoft 보안 업데이트(2015년 6월까지 반영) • EC2Config 서비스 버전 3.6.269 • 현재 Windows PowerShell용 AWS 도구 • 현재 AWS CloudFormation 헬퍼 스크립트 <p>Windows Server 2012 R2를 포함한 AMI만 해당</p> <ul style="list-style-type: none"> • AWS PV 드라이버 7.3.1
2015.5.13	<p>모든 AMI</p> <ul style="list-style-type: none"> • Microsoft 보안 업데이트(2015년 5월까지 반영) • EC2Config 서비스 버전 3.5.228 • 현재 Windows PowerShell용 AWS 도구
2015.04.15	<p>모든 AMI</p> <ul style="list-style-type: none"> • Microsoft 보안 업데이트(2015년 4월까지 반영) • EC2Config 서비스 버전 3.3.174 • 현재 Windows PowerShell용 AWS 도구
2015.03.11	<p>모든 AMI</p> <ul style="list-style-type: none"> • Microsoft 보안 업데이트(2015년 3월까지 반영) • EC2Config 서비스 버전 3.2.97 • 현재 Windows PowerShell용 AWS 도구 <p>Windows Server 2012 R2를 포함한 AMI만 해당</p> <ul style="list-style-type: none"> • AWS PV 드라이버 7.3.0
2015.02.11	<p>모든 AMI</p> <ul style="list-style-type: none"> • Microsoft 보안 업데이트(2015년 2월까지 반영) • EC2Config 서비스 버전 3.0.54 • 현재 Windows PowerShell용 AWS 도구 • 현재 AWS CloudFormation 헬퍼 스크립트
2015.01.14	<p>모든 AMI</p> <ul style="list-style-type: none"> • Microsoft 보안 업데이트(2015년 1월까지 반영) • EC2Config 서비스 버전 2.3.313 • 현재 Windows PowerShell용 AWS 도구 • 현재 AWS CloudFormation 헬퍼 스크립트

2014년 월간 AMI 업데이트

Microsoft 업데이트에 대한 자세한 내용은 [2014년 소프트웨어 업데이트 서비스 및 Windows Server 업데이트 서비스 변경 내용에 대한 설명](#)을 참조하십시오.

릴리스	변경
2014.12.10	<p>모든 AMI</p> <ul style="list-style-type: none">Microsoft 보안 업데이트(2014년 12월까지 반영)EC2Config 서비스 버전 2.2.12현재 Windows PowerShell용 AWS 도구
2014.11.19	<p>모든 AMI</p> <ul style="list-style-type: none">Microsoft 보안 업데이트(2014년 11월까지 반영)EC2Config 서비스 버전 2.2.11현재 Windows PowerShell용 AWS 도구
2014.10.15	<p>모든 AMI</p> <ul style="list-style-type: none">Microsoft 보안 업데이트(2014년 10월까지 반영)EC2Config 서비스 버전 2.2.10현재 Windows PowerShell용 AWS 도구 <p>Windows Server 2012 R2를 포함한 AMI만 해당</p> <ul style="list-style-type: none">AWS PV 드라이버 7.2.4.1(현재는 기본 지원되는 플러그-앤-플레이 클린업(Plug and Play Cleanup)으로 인한 문제 해결)
2014.09.10	<p>모든 AMI</p> <ul style="list-style-type: none">Microsoft 보안 업데이트(2014년 9월까지 반영)EC2Config 서비스 버전 2.2.8현재 Windows PowerShell용 AWS 도구 <p>Windows Server 2012 R2를 포함한 AMI만 해당</p> <ul style="list-style-type: none">플러그-앤-플레이 비활성화(중요 정보 참조)AWS PV 드라이버 7.2.2.1(설치 제거 프로그램으로 인한 문제 해결)
2014.08.13	<p>모든 AMI</p> <ul style="list-style-type: none">Microsoft 보안 업데이트(2014년 8월까지 반영)EC2Config 서비스 버전 2.2.7현재 Windows PowerShell용 AWS 도구 <p>Windows Server 2012 R2를 포함한 AMI만 해당</p> <ul style="list-style-type: none">AWS PV 드라이버 7.2.2.1(디스크 성능 개선, 다수 네트워크 인터페이스 재연결 및 손실된 네트워크 설정으로 인한 문제 해결)

릴리스	변경
2014.07.10	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2014년 7월까지 반영) EC2Config 서비스 버전 2.2.5 현재 Windows PowerShell용 AWS 도구
2014.06.12	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2014년 6월까지 반영) EC2Config 서비스 버전 2.2.4 NVIDIA 드라이브 제거(Windows Server 2012 R2 AMI는 해당 없음) 현재 Windows PowerShell용 AWS 도구
2014.05.14	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2014년 5월까지 반영) EC2Config 서비스 버전 2.2.2 현재 Windows PowerShell용 AWS 도구 AWS CloudFormation 헬퍼 스크립트 버전 1.4.0
2014.04.09	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2014년 4월까지 반영) 현재 Windows PowerShell용 AWS 도구 현재 AWS CloudFormation 헬퍼 스크립트
2014.03.12	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2014년 3월까지 반영)
2014.02.12	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2014년 2월까지 반영) EC2Config 서비스 버전 2.2.1 현재 Windows PowerShell용 AWS 도구 KB2634328 BCDEdit <code>useplatformclock</code> 값 제거 <p>Microsoft SQL Server를 포함한 AMI만 해당</p> <ul style="list-style-type: none"> Microsoft SQL Server 2012 SP1 누적 업데이트 패키지 8 Microsoft SQL Server 2008 R2 누적 업데이트 패키지 10

2013년 월간 AMI 업데이트

Microsoft 업데이트에 대한 자세한 내용은 [2013년 소프트웨어 업데이트 서비스 및 Windows Server 업데이트 서비스 변경 내용에 대한 설명](#)을 참조하십시오.

릴리스	변경
2013.11.13	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2013년 11월까지 반영) EC2Config 서비스 버전 2.1.19 현재 Windows PowerShell용 AWS 도구 NTP가 하루에 한 번씩 시간을 동기화하도록 구성(기본 설정은 매 7일) <p>Windows Server 2012를 포함한 AMI만 해당</p> <ul style="list-style-type: none"> WinSXS를 다음 명령을 사용해서 클린업합니다. <code>dism /online /cleanup-image /StartComponentCleanup</code>
2013.09.11	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2013년 9월까지 반영) EC2Config 서비스 버전 2.1.18 현재 Windows PowerShell용 AWS 도구 AWS CloudFormation 헬퍼 스크립트 버전 1.3.15
2013.07.10	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2013년 7월까지 반영) EC2Config 서비스 버전 2.1.16 50GB까지 확장된 루트 볼륨 512MB로 페이지 파일이 설정, 필요한 경우 8GB까지 확장 가능 현재 Windows PowerShell용 AWS 도구
2013.06.12	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2013년 6월까지 반영) 현재 Windows PowerShell용 AWS 도구 <p>Microsoft SQL Server를 포함한 AMI만 해당</p> <ul style="list-style-type: none"> Microsoft SQL Server 2012 SP1 누적 업데이트 패키지 4
2013.05.15	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2013년 5월까지 반영) EC2Config 서비스 버전 2.1.15 모든 인스턴스 스토어 볼륨은 기본적으로 연결됨 기본 지원되는 원격 PowerShell 현재 Windows PowerShell용 AWS 도구
2013.04.14	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2013년 4월까지 반영) 현재 Windows PowerShell용 AWS 도구 AWS CloudFormation 헬퍼 스크립트 버전 1.3.14

릴리스	변경
2013.03.14	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2013년 3월까지 반영) EC2Config 서비스 버전 2.1.14 CPU 하트비트(heartbeat) 픽스 포함 Citrix Agent 현재 Windows PowerShell용 AWS 도구 AWS CloudFormation 헬퍼 스크립트 버전 1.3.11
2013.02.22	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2013년 2월까지 반영) KB2800213 Windows PowerShell 3.0 업그레이드 EC2Config 서비스 버전 2.1.13 시간 픽스 포함 Citrix Agent Citrix PV 드라이버(날짜: 2011.07.19) 현재 Windows PowerShell용 AWS 도구 AWS CloudFormation 헬퍼 스크립트 버전 1.3.8 <p>Microsoft SQL Server를 포함한 AMI만 해당</p> <ul style="list-style-type: none"> Microsoft SQL Server 2012 누적 업데이트 패키지 5

2012년 월간 AMI 업데이트

Microsoft 업데이트에 대한 자세한 내용은 [2012년 소프트웨어 업데이트 서비스 및 Windows Server 업데이트 서비스 변경 내용에 대한 설명](#)을 참조하십시오.

릴리스	변경
2012.12.12	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 12월까지 반영) ActiveTimeBias 레지스트리 값을 0으로 설정 네트워크 어댑터에 대한 IPv6 비활성화 EC2Config 서비스 버전 2.1.9 수입-모듈을 허용하도록 Windows PowerShell용 AWS 도구를 추가하고 정책을 설정
2012.11.15	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 11월까지 반영) EC2Config 서비스 버전 2.1.7
2012.10.10	<p>모든 AMI</p> <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 10월까지 반영)

릴리스	변경
2012.08.15	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 8월까지 반영) EC2Config 서비스 버전 2.1.2 KB2545227
2012.07.11	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 7월까지 반영)
2012.06.12	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 6월까지 반영) 페이지 파일을 4GB로 설정 설치된 언어 팩 제거 성능 옵션을 "Adjust for best performance(성능 극대화를 위해 조정)"로 설정 재시작 시 화면 보호기가 더 이상 로그온 화면을 표시하지 않도록 설정 pnputil을 사용하는 이전 RedHat 드라이버 버전 제거 bcdedit를 사용해서 종복 부트로더 제거 및 bootstatuspolicy(부트 상태 정책)을 ignoreallfailures(모든 장애 무시)로 설정
2012.05.10	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 5월까지 반영) EC2Config 서비스 버전 2.1.0
2012.04.11	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 4월까지 반영) KB2582281 EC2Config의 현재 버전 GMT 대신 UTC로 시스템 시간 표시
2012.03.13	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 3월까지 반영)
2012.02.24	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 2월까지 반영) AMI 이름 및 설명 표준화
2012.01.12	모든 AMI <ul style="list-style-type: none"> Microsoft 보안 업데이트(2012년 1월까지 반영) RedHat PV 드라이버 버전 1.3.10

2011년 이전 월간 AMI 업데이트

릴리스	변경
2011.09.11	<p>모든 AMI</p> <ul style="list-style-type: none">Microsoft 보안 업데이트(2011년 9월까지 반영)
1.04	<p>모든 AMI</p> <ul style="list-style-type: none">최신 Microsoft 보안 업데이트네트워크 드라이버 업데이트인스턴스의 시간대를 변경할 때 VPC의 인스턴스가 접속이 끊기는 문제 해결
1.02	<p>모든 AMI</p> <ul style="list-style-type: none">최신 Microsoft 보안 업데이트네트워크 드라이버 업데이트VPC의 인스턴스에 대한 라이선스 정품 인증 지원 추가
1.01	<p>모든 AMI</p> <ul style="list-style-type: none">최신 Microsoft 보안 업데이트네트워크 활성화를 기다리는 도중 부적절하게 생성된 암호로 인한 문제 해결
1.0	<p>모든 AMI</p> <ul style="list-style-type: none">최초 릴리스

Windows Server 2016 이상 AMI의 변경 사항

AWS는 Windows Server 2016 이상에 대한 AMI를 제공합니다. 이러한 AMI에는 이전 Windows AMI에 대해 다음과 같은 고급 변경 사항이 포함됩니다:

- .NET Framework에서 .NET Core로 변경을 수용하기 위해 Windows Server 2016 AMI에서 EC2Config 서비스가 중단되고 EC2Launch로 교체되었습니다. EC2Launch는 EC2Config 서비스에서 수행되는 대부분의 작업을 수행하는 Windows PowerShell 스크립트 번들입니다. 자세한 내용은 [EC2Launch를 사용한 Windows 인스턴스 구성 \(p. 489\)](#) 단원을 참조하십시오.
- 이전 버전의 Windows Server AMI에서는 EC2Config 서비스를 사용하여 EC2 인스턴스를 도메인에 조인하고 Amazon CloudWatch와 통합을 구성할 수 있습니다. Windows Server 2016 이후 AMI에서 CloudWatch 에이전트를 사용하여 Amazon CloudWatch와의 통합을 구성할 수 있습니다. CloudWatch로 로그 데이터를 보내도록 인스턴스 구성에 대한 자세한 내용은 [CloudWatch 에이전트를 사용하여 Amazon EC2 인스턴스 및 온프레미스 서버에서 지표 및 로그 수집](#)을 참조하십시오. EC2 인스턴스를 도메인에 조인하는 것에 대한 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS-JoinDirectoryServiceDomain JSON 문서를 사용하여 도메인에 인스턴스 조인하기](#)를 참조하십시오.

기타 차이점

그 밖에 Windows Server 2016 이상 AMI에서 생성된 인스턴스에 대한 중요한 차이점을 알아 두십시오.

- 기본적으로 EC2Launch는 두 번째 EBS 볼륨을 초기화하지 않습니다. 스크립트 실행을 예약하거나 사용자 데이터에서 EC2Launch를 호출하여 디스크를 자동으로 초기화하도록 EC2Launch를 구성할 수 있습니다. EC2Launch를 사용하여 디스크를 초기화하는 절차는 [EC2Launch 구성 \(p. 491\)](#)에서 "드라이브 및 드라이브 문자 매핑 초기화"를 참조하십시오.

- 이전에 로컬 구성 파일(AWS.EC2.Windows.CloudWatch.json)을 사용하여 인스턴스에서 CloudWatch 통합을 활성화한 경우 Windows Server 2016 이상 AMI에서 생성된 인스턴스에서 SSM 에이전트로 작업하도록 파일을 구성할 수 있습니다.

자세한 내용은 Microsoft.com에서 [Windows Server 2019](#)을 참조하십시오.

Windows Server 2016 인스턴스에서 Docker 컨테이너 총돌

Windows Server 2016 AMI에서 Docker 서비스를 실행하는 경우 이 서비스는 기본 내부 IP 주소 접두사 값과 다른 CIDR 값을 사용하도록 구성됩니다. 기본값은 172.16.0.0/12입니다. Windows Server 2016 AMI는 기본 Amazon EC2 VPC/ 서브넷과 총돌을 방지하기 위해 172.17.0.0/16을 사용합니다. EC2 인스턴스에 대한 VPC/서브넷 설정을 변경하지 않는 경우 어떠한 작업도 수행할 필요가 없습니다. CIDR 값이 다르기 때문에 총돌이 근본적으로 방지됩니다. VPC/서브넷 설정을 변경할 경우 이러한 내부 IP 주소 접두사 값에 주의하여 총돌을 일으키지 않도록 하십시오. 자세한 내용은 다음 단원을 참조하십시오.

Important

Windows Server 2016 인스턴스에서 Docker를 실행할 계획인 경우 다음 Amazon 머신 이미지(AMI) 또는 이름에 `Windows_Server-2016-English-Full-Containers`가 있는 이미지에 기반을 둔 AMI에서 인스턴스를 생성해야 합니다. 그렇지 않으면 다른 Windows Server 2016 AMI를 사용하면 Docker를 설치하고 Sysprep을 실행한 후 인스턴스가 올바르게 부팅되지 않습니다.

Hibernate Agent(2018.03.16 AMI)와 관련된 문제

2018.03.16 Windows AMI를 출시한 후 Amazon EC2 Hibernate Agent의 구성에서 따옴표로 둑지 않은 경로를 발견했습니다. 이 에이전트는 Windows Server 2008부터 Windows Server 2016까지의 AMI에 포함되었습니다. 이 문제는 Windows Server 2003용 AMI에는 영향을 주지 않습니다.

AWS에서 2018년 3월 16일자 Windows AMI를 제거했습니다. 새 Windows AMI가 제공될 때 알림을 받으려면 다음을 참조하십시오. [Windows AMI 알림에 대한 구독 \(p. 27\)](#).

문제를 완화하기 위해 다음 절차 중 하나를 사용하여 누락된 따옴표를 추가할 수 있습니다. 에이전트가 실행 중인 경우 에이전트를 다시 시작해야 합니다. 또는 2018.03.16 Windows AMI에서 시작한 인스턴스를 종료하고 다른 AMI를 사용하여 시작한 인스턴스로 바꿀 수 있습니다.

Windows PowerShell

1. Windows 인스턴스에서 Windows Powershell을 엽니다.
2. 다음 명령을 사용하여 누락된 따옴표를 추가하고 구성 파일을 업데이트합니다.

```
cmd /c 'sc config EC2HibernateAgent binPath="\"%ProgramFiles%\Amazon\Hibernate\EC2HibernateAgent.exe\""'
```

3. 다음 명령을 사용하여 업데이트한 구성을 확인합니다.

```
(Get-ItemProperty -Path Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EC2HibernateAgent).ImagePath
```

다음 예와 같이 응답이 따옴표로 둑였는지 확인합니다.

```
"C:\Program Files\Amazon\Hibernate\EC2HibernateAgent.exe"
```

4. 다음 명령을 사용하여 Status가 Running인지 확인합니다.

```
Get-Service EC2HibernateAgent
```

에이전트가 실행 중이면 다음 명령을 사용하여 에이전트를 다시 시작해야 변경 사항이 적용됩니다.

```
Restart-Service EC2HibernateAgent
```

명령 프롬프트

1. Windows 인스턴스에서 명령 프롬프트 창을 엽니다.
2. 다음 명령을 사용하여 누락된 따옴표를 추가하고 구성 파일을 업데이트합니다.

```
sc config EC2HibernateAgent binPath="\"%ProgramFiles%\Amazon\Hibernate\EC2HibernateAgent.exe\""
```

3. 다음 명령을 사용하여 업데이트한 구성을 확인합니다.

```
sc qc EC2HibernateAgent
```

BINARY_PATH_NAME의 경로가 다음 예에 나와 있는 것처럼 따옴표로 묶였는지 확인합니다.

```
"C:\Program Files\Amazon\Hibernate\EC2HibernateAgent.exe"
```

4. 다음 명령을 사용하여 STATE가 RUNNING인지 확인합니다.

```
sc query EC2HibernateAgent
```

에이전트가 실행 중이면 다음 명령을 사용하여 에이전트를 다시 시작해야 변경 사항이 적용됩니다.

```
sc stop EC2HibernateAgent && sc start EC2HibernateAgent
```

STIG 규정을 준수하는 Amazon EC2 Windows Server AMI

보안 기술 규현 가이드(STIG)는 Defense Information Systems Agency(DISA)에서 생성한 구성 표준으로 정보 시스템과 소프트웨어를 보호합니다. 시스템이 STIG 표준 규정을 준수하려면 다양한 보안 설정을 설치, 구성 및 테스트해야 합니다. Amazon EC2 Windows Server AMI는 STIG 규정 준수를 위해 160개가 넘는 필수 보안 설정으로 사전 구성되어 있습니다. STIG 규정을 준수하는 시스템으로는 Windows Server 2012 R2, Windows Server 2016 및 Windows Server 2019가 있습니다. STIG 규정을 준수하는 AMI에는 업데이트된 국방부(DoD) 인증서가 있어 STIG 규정 준수를 시작하고 수행하는 데 도움이 됩니다. STIG 규정을 준수하는 AMI를 사용하기 위해 별도의 변경은 필요하지 않습니다.

STIG 규정을 준수하는 Amazon EC2 Windows Server AMI는 모든 퍼블릭 AWS와 GovCloud 리전에서 제공됩니다. Amazon EC2 콘솔에서 직접 이러한 AMI의 인스턴스를 시작할 수 있습니다. 요금은 표준 Windows 요금으로 청구됩니다.

STIG 호환 Windows Server용 Amazon EC2 AMI는 인스턴스를 만들 때 커뮤니티 AMI에서 찾을 수 있습니다. AMI 이름은 다음과 같습니다(YYYY.MM.DD는 최신 버전을 나타냅니다. 날짜 접미사 없이 버전을 검색할 수 있습니다.)

- Windows_Server-2019-English-STIG-Full-YYYY.MM.DD

- Windows_Server-2019-English-STIG-Core-YYYY.MM.DD
- Windows_Server-2016-English-STIG-Full-YYYY.MM.DD
- Windows_Server-2016-English-STIG-Core-YYYY.MM.DD
- Windows_Server-2012-R2-English-STIG-Full-YYYY.MM.DD
- Windows_Server-2012-R2-English-STIG-Core-YYYY.MM.DD

규정 준수 수준

- 높음(범주 I)

가장 심각한 위협이며 기밀성, 가용성 또는 무결성의 손실을 초래할 수 있는 모든 취약성을 포함합니다.

- 보통(범주 II)

기밀성, 가용성 또는 무결성이 손실될 수 있는 모든 취약성입니다. 이러한 위험은 완화될 수 있습니다.

- 낮음(범주 III)

기밀성, 가용성 또는 무결성의 손실을 방지하기 위해 조치를 저하시키는 모든 취약성입니다.

다음 목록과 표에는 별도의 표시가 없는 한 각 운영 체제에 적용되지 않은 STIG를 카테고리별로 보여줍니다.

주제

- 코어 및 기본 운영 체제 (p. 66)
- Internet Explorer(IE) 11 STIG V1 Release 18 (p. 67)
- Microsoft .NET Framework 4.0 STIG V1 Release 9: STIGS는 적용되지 않음 (p. 67)
- Windows 방화벽 STIG V1 Release 7 (p. 67)
- 버전 기록 (p. 67)

코어 및 기본 운영 체제

아래 나타나는 STIG 설정이 적용되지 않았습니다. 다른 STIG는 모두 적용되었습니다. 전체 목록은 [STIG 문서 라이브러리](#)를 참조하십시오. 전체 목록을 보는 방법에 대한 지침은 [SRG 및 STIG를 보는 방법](#)을 참조하십시오.

Windows Server 2019 STIG V1 Release 3

V-92975, V-92977, V-93147, V-93149, V-93183, V-93185, V-93187, V-93203, V-93205, V-93209, V-93217, V-93219, V-93221, V-93227, V-93229, V-93231, V-93281, V-93283, V-93369, V-93379, V-93381, V-93437, V-93439, V-93457, V-93461, V-93473, V-93475, V-93489, V-93511, V-93515, V-93543, V-93567 및 V-93571

Windows Server 2016 STIG V1 Release 12

V-73217, V-73221, V-73223, V-73225, V-73229, V-73231, V-73233, V-73235, V-73241, V-73245, V-73259, V-73261, V-73263, V-73265, V-73273, V-73275, V-73277, V-73279, V-73281, V-73283, V-73285, V-73307, V-73401, V-73403, V-73521, V-73607, V-73623, V-73625, V-73647, V-73649, V-73701, V-78127, V-90357, and V-90357

Windows Server 2012 R2 STIG V2 Release 18

V-1072, V-1074, V-1076, V-1089, V-1112, V-1114, V-1115, V-1127, V-1145, V-2907, V-3289, V-3383, V-3472, V-3487, V-6840, V-7002, V-14225, V-15505, V-26359, V-36451, V-36658, V-36659, V-36661, V-36662, V-36666, V-36670, V-36671, V-36672, V-36678, V-36733, V-36734, V-36735, V-36736, V-40172, V-40173, V-42420, V-57637, V-57641, V-57645, V-57653, V-57655, V-57719 및 V-75915

Internet Explorer(IE) 11 STIG V1 Release 18

Internet Explorer와 관련된 STIG 설정은 모두 적용되었습니다. 전체 목록은 [STIG 문서 라이브러리](#)를 참조하십시오. 전체 목록을 보는 방법에 대한 지침은 [SRG 및 STIG를 보는 방법](#)을 참조하십시오.

Windows Server 2019, 2016 및 2012 R2

STIG 설정이 모두 적용되었습니다.

Microsoft .NET Framework 4.0 STIG V1 Release 9: STIGs는 적용되지 않음

다음 STIG 설정은 적용되지 않았습니다. 다른 STIG 설정은 모두 적용되었습니다. 전체 목록은 [STIGs U_MS_DotNet_Framework_4-0_V1R9_STIG.zip](#)을 참조하십시오. 전체 목록을 보는 방법에 대한 지침은 [SRG 및 STIG를 보는 방법](#)을 참조하십시오.

Windows Server 2019, 2016 및 2012 R2

V-7055, V-7061, V-7063, V-7067, V-7069, V-7070, V-18395, V-30926, V-30935, V-30937, V-30968, V-30972, V-30986, V-31026 및 V-32025

Windows 방화벽 STIG V1 Release 7

Windows 방화벽과 관련된 모든 STIG 설정이 적용되었습니다. 전체 목록은 https://dl.dod.cyber.mil/wp-content/uploads/stigs/zip/U_Windows_Firewall_V1R7_STIG.zip을 참조하십시오. 전체 목록을 보는 방법에 대한 지침은 [SRG 및 STIG를 보는 방법](#)을 참조하십시오.

버전 기록

다음 표에서는 STIG AMI 버전 기록 업데이트를 보여줍니다.

날짜	AMI	세부 정보
2019년 12월 6일	Windows 방화벽 STIG V1 R17	최초 릴리스.
2019년 12월 6일	Server 2012 R2 Core 및 Base V2 R17 Server 2016 Core 및 Base V1 R11 Internet Explorer 11 V1 R18 Microsoft .NET Framework 4.0 V1 R9	업데이트된 버전이며 STIG가 적용되었습니다.
2019년 9월 17일	Server 2012 R2 Core 및 Base V2 R16 Server 2016 Core 및 Base V1 R9 Server 2019 Core 및 Base V1 R2 Internet Explorer 11 V1 R17 Microsoft .NET Framework 4.0 V1 R8	최초 릴리스.

Windows AMI 찾기

인스턴스를 시작하려면 사용할 AMI를 선택해야 합니다. AMI를 선택할 때 시작할 인스턴스에 대해 다음 요구 사항을 고려하십시오.

- 리전
- 운영 체제
- 아키텍처: 32비트(i386), 64비트(x86_64) 또는 64비트 ARM(arm64)
- 공급자(예: Amazon Web Services)
- 추가 소프트웨어(예: SQL server)

Linux AMI를 찾아야 하는 경우 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux AMI 찾기](#) 단원을 참조하십시오.

목차

- [Amazon EC2 콘솔을 사용하여 Windows AMI 찾기 \(p. 68\)](#)
- [Windows PowerShell용 AWS 도구를 사용하여 AMI 찾기 \(p. 69\)](#)
- [AWS CLI를 사용하여 AMI 찾기 \(p. 69\)](#)
- [시스템 관리자를 사용하여 최신 Amazon Linux AMI 찾기 \(p. 69\)](#)
- [시스템 관리자 파라미터를 사용하여 AMI 찾기 \(p. 70\)](#)

Amazon EC2 콘솔을 사용하여 Windows AMI 찾기

Windows 콘솔을 사용하여 Amazon EC2 AMI를 찾을 수 있습니다. 시작 마법사를 사용하여 인스턴스를 시작할 때 AMI 목록에서 선택하거나 이미지 페이지를 사용하여 사용 가능한 모든 AMI를 검색할 수 있습니다. AMI ID는 AWS 리전마다 고유합니다.

시작 마법사를 사용하여 Windows AMI를 찾으려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 인스턴스를 실행할 리전을 선택합니다. 현재 위치와 관계없이 사용자가 고를 수 있는 리전을 임의로 선택합니다.
3. 콘솔 대시보드에서 인스턴스 시작을 선택합니다.
4. 빠른 시작 탭의 목록에서 자주 사용되는 AMI 중 하나를 선택합니다. 필요한 AMI가 표시되지 않는 경우 나의 AMI, AWS Marketplace 또는 커뮤니티 AMI 탭을 선택하여 추가 AMI를 찾습니다. 자세한 내용은 [1 단계: Amazon 머신 이미지\(AMI\) 선택 \(p. 377\)](#) 단원을 참조하십시오.

이미지 페이지를 사용하여 Windows AMI를 찾으려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 인스턴스를 실행할 리전을 선택합니다. 현재 위치와 관계없이 사용자가 고를 수 있는 리전을 임의로 선택합니다.
3. 탐색 창에서 AMI를 선택합니다.
4. (선택 사항) 필터 옵션을 사용하여 원하는 AMI만 표시하도록 표시되는 AMI 목록의 범위를 지정합니다. 예를 들어, AWS에서 제공하는 모든 Windows AMI를 나열하려면 퍼블릭 이미지를 선택합니다. 검색 창을 선택하고 메뉴에서 소유자를 선택한 다음 Amazon 이미지를 선택합니다. 검색 창을 다시 선택하고 플랫폼을 선택한 다음 제공된 목록에서 운영 체제를 선택합니다.
5. (선택 사항) 열 표시/숨기기 아이콘을 선택하여 표시할 이미지 속성(예: 루트 디바이스 유형)을 선택합니다. 또는 목록에서 AMI를 선택하고 세부 정보 탭에서 속성을 조회할 수 있습니다.

6. 이 AMI에서 인스턴스를 시작하려면 원하는 인스턴스를 선택한 다음 시작을 선택합니다. 콘솔을 통한 인스턴스 시작에 대한 자세한 내용은 [AMI에서 인스턴스 시작 \(p. 377\)](#)을 참조하십시오. 인스턴스를 시작할 준비가 되지 않은 경우, 나중을 위해 AMI ID를 기록해 둡니다.

Windows PowerShell용 AWS 도구를 사용하여 AMI 찾기

Amazon EC2 또는 AWS 시스템 관리자의 cmdlet을 사용하여 필요에 맞는 Windows AMI만 나열되도록 할 수 있습니다. 필요에 맞는 AMI를 찾았으면 인스턴스를 시작할 때 사용할 수 있도록 ID를 기록해 둡니다. 자세한 내용은 Windows PowerShell용 AWS 도구 사용 설명서의 [Windows PowerShell을 사용한 인스턴스 시작](#) 단원을 참조하십시오.

Amazon EC2

자세한 내용과 예제는 Windows PowerShell용 AWS 도구 사용 설명서의 [Windows PowerShell을 사용한 AMI 찾기](#) 관련 문서를 참조하십시오.

시스템 관리자 Parameter Store

자세한 내용과 예제는 [시스템 관리자 파라미터 스토어를 사용하여 최신 Windows AMI 큐리](#)를 참조하십시오.

AWS CLI를 사용하여 AMI 찾기

Amazon EC2 또는 AWS 시스템 관리자의 AWS CLI 명령을 사용하여 필요에 맞는 Windows AMI만 나열되도록 할 수 있습니다. 필요에 맞는 AMI를 찾았으면 인스턴스를 시작할 때 사용할 수 있도록 ID를 기록해 둡니다. 자세한 내용은 AWS Command Line Interface 사용 설명서의 [AWS CLI를 사용한 인스턴스 시작하기](#) 단원을 참조하십시오.

Amazon EC2

`describe-images` 명령은 파라미터 필터링을 지원합니다. 예를 들어, Amazon 소유의 퍼블릭 AMI를 표시하려면 `--owners` 파라미터를 사용합니다.

```
aws ec2 describe-images --owners self amazon
```

Windows AMI만 표시하려면 이전 명령에 다음 필터를 추가합니다.

```
--filters "Name=platform,Values=windows"
```

Important

`describe-images` 명령에서 `--owners` 플래그가 누락되면 소유 여부와 관계없이 시작 권한을 보유한 모든 이미지를 반환합니다.

시스템 관리자 Parameter Store

자세한 내용과 예제는 [시스템 관리자 파라미터 스토어를 사용하여 최신 Windows AMI 큐리](#)를 참조하십시오.

시스템 관리자를 사용하여 최신 Amazon Linux AMI 찾기

Amazon EC2는 인스턴스를 시작할 때 사용할 수 있는 AWS 유지 관리 퍼블릭 AMI에 대한 AWS 시스템 관리자 퍼블릭 파라미터를 제공합니다. 예를 들어, EC2에서 제공하는 `/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2` 파라미터는 모든 리전에서 사용할 수 있으며 항상 지정된 리전의 최신 Amazon Linux 2 AMI 버전을 가리킵니다.

Amazon EC2 AMI 퍼블릭 파라미터는 다음 경로에서 사용할 수 있습니다.

- /aws/service/ami-amazon-linux-latest
- /aws/service/ami-windows-latest

AWS CLI에서 다음 명령을 사용하여 현재 AWS 리전의 모든 Windows AMI 목록을 볼 수 있습니다.

```
aws ssm get-parameters-by-path --path /aws/service/ami-windows-latest --query Parameters[ ].Name
```

퍼블릭 파라미터를 사용하여 인스턴스를 시작하려면

다음 예제에서는 EC2에서 제공하는 퍼블릭 파라미터를 사용하여 최신 Amazon Linux 2 AMI를 사용하는 m5.xlarge 인스턴스를 시작합니다.

명령에서 파라미터를 지정하려면 `resolve:ssm:public-parameter` 구문을 사용합니다. 여기서 `resolve:ssm`은 표준 접두사이고 `public-parameter`은 퍼블릭 파라미터의 경로와 이름입니다.

이 예제에는 `--count` 및 `--security-group` 파라미터가 포함되어 있지 않습니다. `--count`의 기본값은 1입니다. 기본 VPC와 기본 보안 그룹이 있는 경우 이들이 사용됩니다.

```
aws ec2 run-instances  
  --image-id resolve:ssm:/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2  
  --instance-type m5.xlarge  
  --key-name MyKeyPair
```

자세한 내용은 AWS 시스템 관리자 사용 설명서의 [퍼블릭 파라미터 사용](#) 및 [AWS 시스템 관리자 파라미터 스토어를 사용한 최신 Amazon Linux AMI ID 쿼리](#)를 참조하십시오.

시스템 관리자 파라미터를 사용하여 AMI 찾기

콘솔에서 EC2 시작 마법사를 사용하여 인스턴스를 시작할 때 목록에서 AMI를 선택하거나 AMI ID를 가리키는 AWS 시스템 관리자 파라미터를 선택할 수 있습니다. 자동화 코드를 사용하여 인스턴스를 시작하는 경우 AMI ID 대신 시스템 관리자 파라미터를 지정할 수 있습니다.

시스템 관리자 파라미터는 시스템 관리자 파라미터 스토어에서 생성할 수 있는 고객 정의 키-값 페어입니다. 파라미터 스토어는 애플리케이션 구성 값을 외부화할 수 있는 중앙 스토어를 제공합니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 파라미터 스토어](#)를 참조하십시오.

AMI ID를 가리키는 파라미터를 생성할 때는 데이터 유형을 `aws:ec2:image`로 지정해야 합니다. 이 데이터 유형을 사용하면 파라미터가 생성되거나 수정될 때 파라미터 값이 AMI ID로 확인됩니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [Amazon Machine Image\(AMI\) ID에 대한 기본 파라미터 지원](#)을 참조하십시오.

목차

- [사용 사례 \(p. 70\)](#)
- [시스템 관리자 파라미터를 사용하여 인스턴스 시작 \(p. 71\)](#)
- [권한 \(p. 72\)](#)
- [제한 사항 \(p. 72\)](#)

사용 사례

시스템 관리자 파라미터를 사용하여 AMI ID를 가리키면 사용자가 인스턴스를 시작할 때 올바른 AMI를 더 쉽게 선택할 수 있는 동시에 관리자가 자동화 코드를 더 간편하게 유지 관리할 수 있습니다.

사용자의 사용 편의성 향상

특정 AMI를 사용하여 인스턴스를 실행해야 하며 해당 AMI가 정기적으로 업데이트되는 경우 사용자가 시스템 관리자 파라미터를 선택하여 해당 AMI를 찾도록 하는 것이 좋습니다. 사용자에게 시스템 관리자 파라미터를 선택하도록 하면 항상 최신 AMI를 사용하여 인스턴스가 시작될 수 있습니다.

예를 들어 조직에서 매달 최신 운영 체제 및 애플리케이션 패치가 포함된 새 버전의 AMI를 생성하며 사용자가 최신 버전의 AMI를 사용하여 인스턴스를 시작하도록 하려는 경우, 사용자가 최신 버전을 사용하도록 올바른 AMI ID를 가리키는 시스템 관리자 파라미터(예: `golden-ami`)를 생성할 수 있습니다. 새 버전의 AMI가 생성될 때마다 이 파라미터의 AMI ID 값을 업데이트하여 항상 최신 AMI를 가리키도록 합니다. 사용자는 AMI에 대한 정기 업데이트에 대해 알 필요 없이 매번 동일한 시스템 관리자 파라미터를 계속 선택하면 됩니다. 사용자가 시스템 관리자 파라미터를 선택하도록 하면 인스턴스를 시작할 때 사용할 AMI를 더 쉽게 선택할 수 있습니다.

자동화 코드 유지 관리 간소화

자동화 코드를 사용하여 인스턴스를 시작하는 경우 AMI ID 대신 시스템 관리자 파라미터를 지정할 수 있습니다. 새 버전의 AMI가 생성되면 최신 AMI를 가리키도록 이 파라미터의 AMI ID 값을 변경합니다. 파라미터를 참조하는 자동화 코드는 새 버전의 AMI가 생성될 때마다 수정할 필요가 없습니다. 이를 통해 자동화 유지 관리가 크게 간소화되고 배포 비용을 절감할 수 있습니다.

Note

시스템 관리자 파라미터가 가리키는 AMI ID를 변경해도 실행 중인 인스턴스는 영향을 받지 않습니다.

시스템 관리자 파라미터를 사용하여 인스턴스 시작

콘솔 또는 AWS CLI를 사용하여 인스턴스를 시작할 수 있습니다. AMI ID를 지정하는 대신 AMI ID를 가리키는 AWS 시스템 관리자 파라미터를 지정할 수 있습니다.

시스템 관리자 파라미터를 사용하여 Windows AMI를 찾으려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 인스턴스를 실행할 리전을 선택합니다. 현재 위치와 관계없이 사용자가 고를 수 있는 리전을 임의로 선택합니다.
3. 콘솔 대시보드에서 인스턴스 시작을 선택합니다.
4. 오른쪽 상단에 있는 Search by 시스템 관리자 parameter(Systems Manager 파라미터로 검색)을 선택합니다.
5. 시스템 관리자 parameter(Systems Manager 파라미터)에서 파라미터를 선택합니다. 해당 AMI ID가 Currently resolves to(현재 확인된 값) 옆에 나타납니다.
6. 검색을 선택합니다. AMI ID와 일치하는 AMI가 목록에 나타납니다.
7. 목록에서 해당 AMI를 선택하고 선택을 선택합니다.

시작 마법사를 사용하여 AMI에서 인스턴스를 시작하는 방법에 대한 자세한 내용은 [1단계: Amazon 머신 이미지\(AMI\) 선택 \(p. 377\)](#) 단원을 참조하십시오.

AMI ID 대신 AWS 시스템 관리자 파라미터를 사용하여 인스턴스를 시작하려면(AWS CLI)

다음 예제에서는 시스템 관리자 파라미터 `golden-ami`를 사용하여 `m5.xlarge` 인스턴스를 시작합니다. 이 파라미터는 AMI ID를 가리킵니다.

명령에서 파라미터를 지정하려면 `resolve:ssm:/parameter-name` 구문을 사용합니다. 여기서 `resolve:ssm`은 표준 접두사이고 `parameter-name`은 고유한 파라미터 이름입니다. 파라미터 이름은 대/소문자를 구분합니다. 파라미터 이름의 백슬래시는 파라미터가 계층 구조의 일부인 경우에만 필요합니다(예: `/amis/production/golden-ami`). 파라미터가 계층의 일부가 아닌 경우 백슬래시를 생략할 수 있습니다.

이 예제에는 `--count` 및 `--security-group` 파라미터가 포함되어 있지 않습니다. `--count`의 기본값은 1입니다. 기본 VPC와 기본 보안 그룹이 있는 경우 이들이 사용됩니다.

```
aws ec2 run-instances
--image-id resolve:ssm:/golden-ami
--instance-type m5.xlarge
...
```

특정 버전의 AWS 시스템 관리자 파라미터를 사용하여 인스턴스를 시작하려면(AWS CLI)

시스템 관리자 파라미터는 버전을 지원합니다. 파라미터의 각 이터레이션에는 고유한 버전 번호가 지정됩니다. `resolve:ssm:parameter-name:version`으로 파라미터의 버전을 참조할 수 있습니다. 여기서 `version`은 고유한 버전 번호입니다. 버전이 지정되지 않은 경우 기본적으로 최신 버전의 파라미터가 사용됩니다.

다음 예제에서는 버전 2의 파라미터를 사용합니다.

이 예제에는 `--count` 및 `--security-group` 파라미터가 포함되어 있지 않습니다. `--count`의 기본값은 1입니다. 기본 VPC와 기본 보안 그룹이 있는 경우 이들이 사용됩니다.

```
aws ec2 run-instances
--image-id resolve:ssm:/golden-ami:2
--instance-type m5.xlarge
...
```

AWS에서 제공하는 퍼블릭 파라미터를 사용하여 인스턴스를 시작하려면

Amazon EC2는 AWS에서 제공하는 퍼블릭 AMI에 대한 시스템 관리자 퍼블릭 파라미터를 제공합니다. 예를 들어, 퍼블릭 파라미터 `/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2`는 모든 리전에서 사용할 수 있으며 항상 해당 리전의 Amazon Linux 2 AMI의 최신 버전을 가리킵니다.

```
aws ec2 run-instances
--image-id resolve:ssm:/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-x86_64-gp2
--instance-type m5.xlarge
...
```

권한

인스턴스 시작 마법사에서 AMI ID를 가리키는 시스템 관리자 파라미터를 사용하는 경우 IAM 정책에 `ssm:DescribeParameters` 및 `ssm:GetParameters`를 추가해야 합니다. `ssm:DescribeParameters`는 IAM 사용자에게 시스템 관리자 파라미터를 보고 선택할 수 있는 권한을 부여하며, `ssm:GetParameters`는 IAM 사용자에게 시스템 관리자 파라미터의 값을 가져올 수 있는 권한을 부여합니다. 특정 시스템 관리자 파라미터에 대한 액세스를 제한할 수도 있습니다. 자세한 내용은 [EC2 시작 마법사 사용 \(p. 883\)](#) 단원을 참조하십시오.

제한 사항

AMI 및 시스템 관리자 파라미터는 리전별로 다릅니다. 여러 리전에서 동일한 시스템 관리자 파라미터 이름을 사용하려면 동일한 이름(예: `golden-ami`)을 가진 시스템 관리자 파라미터를 각 리전에 생성합니다. 각 리전에서 시스템 관리자 파라미터가 해당 리전의 AMI를 가리키도록 지정합니다.

공유 AMI

공유 AMI는 다른 개발자가 사용할 수 있도록 공유된 개발자 생성 AMI입니다. Amazon EC2를 처음 시작할 때 가장 손쉬운 방법 중 하나는 필요한 구성 요소를 가진 공유 AMI를 선택한 다음 개인 설정을 추가하는 것입니다. 자체 AMI를 생성하여 다른 사람과 공유할 수도 있습니다.

공유 AMI를 사용할 때는 사용자의 주의가 필요합니다. Amazon에서는 다른 Amazon EC2 사용자와 공유된 AMI의 무결성이나 보안성을 보장하지 않습니다. 따라서 공유 AMI를 사용할 때는 데이터 센터에서 외부 코드를 배포하는 경우와 마찬가지로 이런 AMI를 취급하고 그에 따라 적합한 조치를 취해야 합니다. 신뢰할 수 있는 출처의 AMI를 사용하십시오.

Amazon의 공개 이미지는 별칭을 소유주 이름으로 사용하며 계정 필드에 amazon가 표시됩니다. 따라서 Amazon에서 배포한 AMI를 쉽게 찾을 수 있습니다. 다른 사용자는 AMI 별칭을 사용할 수 없습니다.

AMI 생성에 대한 자세한 정보는 [Amazon EBS 지원 Windows AMI 생성](#)을 참조하십시오. AWS Marketplace에서 애플리케이션을 빌드하고 제공하며 관리하는 정보는 [AWS Marketplace 설명서](#)를 참조하십시오.

목차

- [공유 AMI 검색 \(p. 73\)](#)
- [퍼블릭 AMI 설정 \(p. 75\)](#)
- [지정한 AWS 계정과 AMI 공유 \(p. 76\)](#)
- [북마크 사용 \(p. 79\)](#)
- [공유 Windows AMI 지침 \(p. 79\)](#)

공유 AMI 검색

공유 AMI는 Amazon EC2 콘솔 또는 명령줄을 사용해 검색할 수 있습니다.

AMI는 리전 리소스입니다. 따라서 공유 AMI(퍼블릭 또는 프라이빗)를 검색할 경우에는 공유되고 있는 리전 내에서 검색해야 합니다. AMI를 다른 리전에서 사용할 수 있도록 하려면 AMI를 해당 리전에 복사한 후 공유하십시오. 자세한 내용은 [AMI 복사](#)를 참조하십시오.

공유 AMI 검색(콘솔)

콘솔을 사용해 프라이빗 AMI를 검색하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 AMI를 선택합니다.
3. 프라이빗 이미지를 첫 필터로 선택합니다. 사용자와 공유된 모든 AMI가 나열됩니다. 검색 결과를 좀 더 세부적으로 보려면 검색 창을 선택하여 메뉴에서 제공하는 필터 옵션을 사용하십시오.

콘솔을 사용해 퍼블릭 AMI를 검색하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 AMI를 선택합니다.
3. 퍼블릭 이미지를 첫 필터로 선택합니다. 검색 결과를 좀 더 세부적으로 보려면 검색 창을 선택하여 메뉴에서 제공하는 필터 옵션을 사용하십시오.
4. 필터를 사용하면 원하는 유형의 AMI만 검색할 수 있습니다. 예를 들어 소유자 :]를 선택한 다음 Amazon 이미지를 선택하면 Amazon의 퍼블릭 이미지만 표시됩니다.

공유 AMI 검색(Windows PowerShell용 도구)

[Get-EC2Image](#) 명령(Windows PowerShell용 도구)을 사용해 AMI를 나열합니다. 아래 예시와 같이 원하는 유형의 AMI만 나타나도록 목록을 정리할 수 있습니다.

예: 모든 퍼블릭 AMI 나열

다음 명령은 사용자가 소유한 퍼블릭 AMI를 포함한 모든 퍼블릭 AMI를 나열합니다.

```
PS C:\> Get-EC2Image -ExecutableUser all
```

예: 명시적 시작 권한으로 AMI 나열

다음 명령은 사용자가 명시적 시작 권한을 가지고 있는 AMI를 나열합니다. 이 목록에는 사용자가 소유한 AMI는 포함되지 않습니다.

```
PS C:\> Get-EC2Image -ExecutableUser self
```

예: Amazon에서 소유한 AMI 나열

다음 명령은 Amzaon 소유 AMI를 나열합니다. Amazon의 공개 AMI는 별칭을 소유주 이름으로 사용하며 계정 필드에 amazon가 표시됩니다. 따라서 Amazon에서 배포한 AMI를 쉽게 찾을 수 있습니다. 다른 사용자는 AMI 별칭을 사용할 수 없습니다.

```
PS C:\> Get-EC2Image -Owner amazon
```

예: 계정에서 소유한 AMI 나열

다음 명령은 지정된 AWS 계정에서 소유한 AMI를 나열합니다.

```
PS C:\> Get-EC2Image -Owner 123456789012
```

예: 필터를 사용하여 AMI 범위 지정

표시된 AMI 수가 너무 많다면 필터를 사용하여 원하는 유형의 AMI만 나타나도록 할 수 있습니다. 예를 들어, 다음 필터를 사용하면 EBS 기반 AMI만 나열됩니다.

```
-Filter @{ Name="root-device-type"; Values="ebs" }
```

공유 AMI 검색(AWS CLI)

`describe-images` 명령(AWS CLI)을 사용해 AMI를 나열합니다. 아래 예시와 같이 원하는 유형의 AMI만 나타나도록 목록을 정리할 수 있습니다.

예: 모든 퍼블릭 AMI 나열

다음 명령은 사용자가 소유한 퍼블릭 AMI를 포함한 모든 퍼블릭 AMI를 나열합니다.

```
aws ec2 describe-images --executable-users all
```

예: 명시적 시작 권한으로 AMI 나열

다음 명령은 사용자가 명시적 시작 권한을 가지고 있는 AMI를 나열합니다. 이 목록에는 사용자가 소유한 AMI는 포함되지 않습니다.

```
aws ec2 describe-images --executable-users self
```

예: Amazon에서 소유한 AMI 나열

다음 명령은 Amzaon 소유 AMI를 나열합니다. Amazon의 공개 AMI는 별칭을 소유주 이름으로 사용하며 계정 필드에 amazon가 표시됩니다. 따라서 Amazon에서 배포한 AMI를 쉽게 찾을 수 있습니다. 다른 사용자는 AMI 별칭을 사용할 수 없습니다.

```
aws ec2 describe-images --owners amazon
```

예: 계정에서 소유한 AMI 나열

다음 명령은 지정된 AWS 계정에서 소유한 AMI를 나열합니다.

```
aws ec2 describe-images --owners 123456789012
```

예: 필터를 사용하여 AMI 범위 지정

표시된 AMI 수가 너무 많다면 필터를 사용하여 원하는 유형의 AMI만 나타나도록 할 수 있습니다. 예를 들어, 다음 필터를 사용하면 EBS 기반 AMI만 나열됩니다.

```
--filters "Name=root-device-type,Values=ebs"
```

퍼블릭 AMI 설정

Amazon EC2에서는 소유한 AMI를 다른 AWS 계정과 공유할 수 있습니다. 모든 AWS 계정에서 공유한 AMI를 시작할 수 있도록 설정하거나(퍼블릭 AMI), 특정 계정에서만 AMI를 시작할 수 있도록 설정할 수 있습니다 ([지정한 AWS 계정과 AMI 공유 \(p. 76\)](#) 참조). 다른 AWS 계정에서 공유 AMI를 사용하면 관련 요금은 공유 AMI를 시작한 해당 계정에만 청구됩니다.

암호화된 볼륨이 있는 AMI는 퍼블릭으로 설정할 수 없습니다.

AMI는 리전 리소스입니다. 따라서 AMI를 공유하면 해당 리전에서 사용할 수 있습니다. AMI를 다른 리전에서 사용할 수 있도록 하려면 AMI를 해당 리전에 복사한 후 공유하십시오. 자세한 내용은 [AMI 복사 \(p. 103\)](#) 단원을 참조하십시오.

AMI에 제품 코드가 있거나 암호화된 볼륨의 스냅샷이 포함된 경우, 이 AMI를 퍼블릭으로 설정할 수 없습니다. AMI는 특정 AWS 계정과만 공유할 수 있습니다.

모든 AWS 계정과 AMI 공유(콘솔)

AMI를 공개한 후 콘솔을 사용하여 동일한 리전에서 인스턴스를 시작하면 커뮤니티 AMI에서 AMI를 사용할 수 있습니다. AMI 공개 후 AMI가 커뮤니티 AMI에 표시되는 데 약간의 시간이 걸릴 수 있다는 점에 유의하십시오. AMI를 다시 비공개로 바꾼 후 AMI가 커뮤니티 AMI에서 제거되는 데도 약간의 시간이 걸릴 수 있습니다.

콘솔을 사용해 퍼블릭 AMI를 공유하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 AMI를 선택합니다.
3. 목록에서 AMI를 선택한 후 작업에서 이미지 권한 설정을 선택합니다.
4. 퍼블릭을 선택한 다음 저장을 선택합니다.

모든 AWS 계정과 AMI 공유(Windows PowerShell용 도구)

각 AMI에는 소유자를 제외하고 해당 AMI를 사용한 인스턴스 시작이 허용된 launchPermission 계정을 설정할 수 있는 AWS 속성이 존재합니다. AMI의 launchPermission 속성을 변경하여 이 AMI를 퍼블릭 설정(모든 AWS 계정에 시작 권한 허용)하거나 사용자가 지정한 AWS 계정하고만 공유할 수 있습니다.

AMI의 시작 권한을 부여할 계정 ID는 목록에 추가하거나 제거할 수 있습니다. AMI를 퍼블릭 설정하려면 all 그룹을 지정합니다. 퍼블릭 권한과 명시적 시작 권한 모두 설정이 가능합니다.

퍼블릭 AMI 설정

1. 다음과 같이 [Edit-EC2ImageAttribute](#) 명령을 사용하여 지정한 AMI의 launchPermission 목록에 all 그룹을 추가합니다.

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute launchPermission -OperationType add -UserGroup all
```

2. AMI의 시작 권한을 확인하려면 [Get-EC2ImageAttribute](#) 명령을 사용합니다.

```
PS C:\> Get-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute launchPermission
```

3. (선택 사항) AMI를 프라이빗 상태로 되돌리려면 시작 권한 목록에서 all 그룹을 삭제합니다. AMI의 소유자는 언제나 시작 권한을 가지며 이 명령에 영향을 받지 않습니다.

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute launchPermission -OperationType remove -UserGroup all
```

모든 AWS 계정과 AMI 공유(AWS CLI)

각 AMI에는 소유자를 제외하고 해당 AMI를 사용한 인스턴스 시작이 허용된 launchPermission 계정을 설정할 수 있는 AWS 속성이 존재합니다. AMI의 launchPermission 속성을 변경하여 이 AMI를 퍼블릭 설정(모든 AWS 계정에 시작 권한 허용)하거나 사용자가 지정한 AWS 계정하고만 공유할 수 있습니다.

AMI의 시작 권한을 부여할 계정 ID는 목록에 추가하거나 제거할 수 있습니다. AMI를 퍼블릭 설정하려면 all 그룹을 지정합니다. 퍼블릭 권한과 명시적 시작 권한 모두 설정이 가능합니다.

퍼블릭 AMI 설정

1. 다음과 같이 [modify-image-attribute](#) 명령을 실행하고 지정한 AMI의 launchPermission 목록에 all 그룹을 추가합니다.

```
aws ec2 modify-image-attribute \
--image-id ami-0abcdef1234567890 \
--launch-permission "Add=[{Group=all}]"
```

2. AMI의 시작 권한을 확인하려면 [describe-image-attribute](#) 명령을 사용합니다.

```
aws ec2 describe-image-attribute \
--image-id ami-0abcdef1234567890 \
--attribute launchPermission
```

3. (선택 사항) AMI를 프라이빗 상태로 되돌리려면 시작 권한 목록에서 all 그룹을 삭제합니다. AMI의 소유자는 언제나 시작 권한을 가지며 이 명령에 영향을 받지 않습니다.

```
aws ec2 modify-image-attribute \
--image-id ami-0abcdef1234567890 \
--launch-permission "Remove=[{Group=all}]"
```

지정한 AWS 계정과 AMI 공유

AMI를 퍼블릭으로 설정하지 않고 지정한 AWS 계정과 공유할 수 있습니다. 이런 작업은 AWS 계정 ID만 있으면 가능합니다. 암호화되지 않은 볼륨 및 고객 관리형 CMK로 암호화된 볼륨이 있는 AMI만 공유할 수 있습니다. AMI를 암호화된 볼륨과 공유할 경우 AMI의 암호화에 사용되는 모든 CMK도 공유해야 합니다. 자세한

내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오. AWS 관리형 CMK로 암호화된 볼륨이 있는 AMI는 공유할 수 없습니다.

AMI는 리전 리소스입니다. 따라서 AMI를 공유하면 해당 리전에서 사용할 수 있습니다. AMI를 다른 리전에서 사용할 수 있도록 하려면 AMI를 해당 리전에 복사한 후 공유하십시오. 자세한 내용은 [AMI 복사 \(p. 103\)](#) 단원을 참조하십시오.

AMI에서 공유할 수 있는 AWS 계정 수에는 제한이 없습니다. 공유 AMI에 연결하는 사용자 정의 태그는 사용자 AWS 계정에서만 사용할 수 있으며 AMI가 공유되는 다른 계정에서는 사용할 수 없습니다.

AMI 공유(콘솔)

콘솔을 사용해 명시적 시작 권한을 허용하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
 2. 탐색 창에서 AMI를 선택합니다.
 3. 목록에서 AMI를 선택한 후 작업에서 이미지 권한 수정을 선택합니다.
 4. AWS 계정 번호 필드에 AMI를 공유할 AWS 계정 번호를 지정하고 권한 추가를 선택합니다.
- 이 AMI를 다수의 사용자와 공유하려면 이 단계를 반복하여 사용자를 추가합니다.
5. 스냅샷 볼륨 권한 생성을 허용하려면 권한 생성 시 연결된 스냅샷에 대해 "볼륨 생성" 권한 추가.(Add "create volume" permissions to the following associated snapshots when creating permissions.)를 선택합니다.

Note

AMI를 공유하기 위해서 해당 AMI의 레퍼런스인 Amazon EBS 스냅샷을 함께 공유할 필요는 없습니다. AMI만 공유하면 시스템에서 시작에 필요한 Amazon EBS 스냅샷 액세스를 인스턴스에 자동으로 제공합니다. 그러나 AMI가 참조하는 스냅샷을 암호화하는 데 사용되는 모든 CMK를 공유해야 합니다. 자세한 내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

6. 모두 완료했으면 저장을 선택합니다.
7. (선택 사항) AMI를 공유하는 AWS 계정 ID를 보려면 목록에서 AMI를 선택하고 권한 탭을 선택합니다. 공유되는 AMI를 찾으려면 [공유 AMI 검색 \(p. 73\)](#) 단원을 참조하십시오.

AMI를 공유하는 방법(Windows PowerShell용 도구)

`Edit-EC2ImageAttribute` 명령(Windows PowerShell용 도구)을 사용하여 다음 예시와 같이 AMI를 공유할 수 있습니다.

명시적 시작 권한 허용

다음 명령은 지정한 AWS 계정에 특정 AMI의 시작 권한을 허용하는 데 사용됩니다.

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute launchPermission -OperationType add -UserId "123456789012"
```

다음 명령은 스냅샷 볼륨 권한 생성을 허용합니다.

```
PS C:\> Edit-EC2SnapshotAttribute -SnapshotId snap-1234567890abcdef0 -Attribute CreateVolumePermission -OperationType add -UserId 123456789012
```

Note

AMI를 공유하기 위해서 해당 AMI의 레퍼런스인 Amazon EBS 스냅샷을 함께 공유할 필요는 없습니다. AMI만 공유하면 시스템에서 시작에 필요한 Amazon EBS 스냅샷 액세스를 인스턴스에 자동으

로 제공합니다. 그러나 AMI가 참조하는 스냅샷을 암호화하는데 사용되는 모든 CMK를 공유해야 합니다. 자세한 내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

특정 계정에서 시작 권한을 제거하는 방법

다음 명령은 지정한 AWS 계정에 허용했던 특정 AMI의 시작 권한을 제거하는 데 사용됩니다.

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute launchPermission -OperationType remove -UserId "123456789012"
```

다음 명령은 스냅샷 볼륨 권한 생성을 제거합니다.

```
PS C:\> Edit-EC2SnapshotAttribute -SnapshotId snap-1234567890abcdef0 -Attribute CreateVolumePermission -OperationType remove -UserId 123456789012
```

모든 시작 권한을 제거하는 방법

다음 명령은 특정 AMI의 퍼블릭 및 명시적 시작 권한을 모두 삭제하는 데 사용됩니다. AMI의 소유자는 언제나 시작 권한을 가지며 이 명령에 영향을 받지 않습니다.

```
PS C:\> Reset-EC2ImageAttribute -ImageId ami-0abcdef1234567890 -Attribute launchPermission
```

AMI를 공유하는 방법(AWS CLI)

`modify-image-attribute` 명령(AWS CLI)을 사용하여 다음 예시와 같이 AMI를 공유할 수 있습니다.

명시적 시작 권한 허용

다음 명령은 지정한 AWS 계정에 특정 AMI의 시작 권한을 허용하는 데 사용됩니다.

```
aws ec2 modify-image-attribute \
--image-id ami-0abcdef1234567890 \
--launch-permission "Add=[{UserId=123456789012}]"
```

다음 명령은 스냅샷 볼륨 권한 생성을 허용합니다.

```
aws ec2 modify-snapshot-attribute \
--snapshot-id snap-1234567890abcdef0 \
--attribute createVolumePermission \
--operation-type add \
--user-ids 123456789012
```

Note

AMI를 공유하기 위해서 해당 AMI의 레퍼런스인 Amazon EBS 스냅샷을 함께 공유할 필요는 없습니다. AMI만 공유하면 시스템에서 시작에 필요한 Amazon EBS 스냅샷 액세스를 인스턴스에 자동으로 제공합니다. 그러나 AMI가 참조하는 스냅샷을 암호화하는데 사용되는 모든 CMK를 공유해야 합니다. 자세한 내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

특정 계정에서 시작 권한을 제거하는 방법

다음 명령은 지정한 AWS 계정에 허용했던 특정 AMI의 시작 권한을 제거하는 데 사용됩니다.

```
aws ec2 modify-image-attribute \
--image-id ami-0abcdef1234567890 \
```

```
--launch-permission "Remove=[{UserId=123456789012}]"
```

다음 명령은 스냅샷 볼륨 권한 생성을 제거합니다.

```
aws ec2 modify-snapshot-attribute \
--snapshot-id snap-1234567890abcdef0 \
--attribute createVolumePermission \
--operation-type remove \
--user-ids 123456789012
```

모든 시작 권한을 제거하는 방법

다음 명령은 특정 AMI의 퍼블릭 및 명시적 시작 권한을 모두 삭제하는 데 사용됩니다. AMI의 소유자는 언제나 시작 권한을 가지며 이 명령에 영향을 받지 않습니다.

```
aws ec2 reset-image-attribute \
--image-id ami-0abcdef1234567890 \
--attribute launchPermission
```

북마크 사용

퍼블릭 AMI를 생성했거나 다른 AWS 사용자와 AMI를 공유했다면 허용된 사용자가 자신의 계정에서 즉시 인스턴스를 시작할 수 있도록 허용하는 북마크를 생성할 수 있습니다. 사용을 위해 AMI 검색에 시간을 할애할 필요 없이 AMI 레퍼런스를 공유하는 간단한 방법입니다.

AMI는 반드시 퍼블릭 AMI이거나 북마크를 보낼 사용자와 공유된 상태여야 합니다.

AMI 북마크 생성

1. 다음 정보를 참고하여 URL을 입력합니다. 여기에서 리전은 AMI가 속하는 리전입니다.

```
https://console.aws.amazon.com/ec2/v2/home?
region=region#LaunchInstanceWizard:ami=ami_id
```

예를 들어, 다음 URL은 us-east-1 리전의 ami-0abcdef1234567890 AMI에서 인스턴스를 실행합니다.

```
https://console.aws.amazon.com/ec2/v2/home?region=us-
east-1#LaunchInstanceWizard:ami=ami-0abcdef1234567890
```

2. AMI 사용을 원하는 사용자에게 이 링크를 공유합니다.
3. 북마크를 사용하려면 링크를 선택하거나 복사하여 브라우저에 붙여넣기하면 됩니다. AMI 선택이 완료된 상태로 시작 마법사가 열립니다.

공유 Windows AMI 지침

AMI의 안정성을 높이고 공격 대상 영역을 최소화하려면 다음 지침을 사용하십시오.

- 어떤 보안 지침도 포괄적일 수는 없습니다. 공유 AMI를 구축할 때는 민감한 데이터의 유출 가능성에 특히 유의하고, 충분한 시간을 할애하여 검토하십시오.
- AMI 개발, 업데이트, 리퍼블리싱을 위한 반복적인 프로세스를 개발합니다.
- 최신 운영 체제, 패키지, 소프트웨어를 사용하여 AMI를 개발합니다.
- 최신 버전의 EC2Config 서비스를 [다운로드](#)하여 설치합니다. 이 서비스 설치에 대한 자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 단원을 참조하십시오.

- `Ec2SetPassword`, `Ec2WindowsActivate`, `Ec2HandleUserData`가 활성화되었는지 확인합니다.
- 게스트 계정이나 원격 데스크톱 사용자 계정이 없는지 확인합니다.
- 필요 없는 서비스와 프로그램을 비활성화하거나 삭제하여 AMI의 공격 대상 영역을 최소화합니다.
- 키 페어와 같은 인스턴스 자격 증명을 AMI에서 제거합니다(AMI에 저장했을 경우). 안전한 위치에 자격 증명을 저장합니다.
- 관리자 암호와 다른 계정의 암호를 공유하기에 적합한 값으로 설정했는지 확인합니다. 이러한 암호는 공유 AMI를 시작하는 누구나 사용할 수 있습니다.
- AMI를 공유하기 전에 테스트합니다.

유료 AMI

유료 AMI는 개발자에게서 구입할 수 있는 AMI입니다.

Amazon EC2가 AWS Marketplace와 통합되어 개발자가 다른 Amazon EC2 사용자에게 AMI 사용 요금을 청구하거나 인스턴스에 대한 지원을 제공할 수 있습니다.

AWS Marketplace는 EC2 인스턴스를 시작하는 데 사용할 수 있는 AMI를 비롯하여 AWS에서 실행되는 소프트웨어를 구입할 수 있는 온라인 상점입니다. 요구 사항에 맞는 제품을 찾을 수 있도록 AWS Marketplace AMI는 범주(예: Developer Tools)별로 구성됩니다. AWS Marketplace에 대한 자세한 내용은 [AWS Marketplace 사이트](#)를 참조하십시오.

유료 AMI에서 인스턴스를 시작하는 것은 다른 AMI에서 인스턴스를 시작하는 것과 같습니다. 추가 파라미터가 필요하지 않습니다. AMI 소유자가 설정한 요금과 관련 웹 서비스에 대한 스탠다드 사용 요금(예: Amazon EC2에서 m1.small 인스턴스 유형 실행에 대한 시간당 요금)에 따라 인스턴스 요금이 부과됩니다. 추가 세금이 적용될 수 있습니다. 유료 AMI의 소유자는 특정 인스턴스가 해당 유료 AMI를 사용하여 시작되었는지 여부를 확인할 수 있습니다.

Important

Amazon DevPay는 더 이상 새로운 판매자 또는 제품을 수락하지 않습니다. 이제 AWS Marketplace가 AWS를 통해 소프트웨어와 서비스를 판매하는 단일 통합 전자 상거래 플랫폼입니다. AWS Marketplace에서 소프트웨어를 배포하고 판매하는 방법에 대한 자세한 내용은 [AWS Marketplace에서의 판매](#)를 참조하십시오. AWS Marketplace는 Amazon EBS 기반 AMI를 지원합니다.

내용

- [AMI 판매 \(p. 80\)](#)
- [유료 AMI 찾기 \(p. 81\)](#)
- [유료 AMI 구입 \(p. 82\)](#)
- [인스턴스에 대한 제품 코드 가져오기 \(p. 82\)](#)
- [유료 지원 사용 \(p. 83\)](#)
- [유료 및 지원된 AMI에 대한 청구서 \(p. 83\)](#)
- [AWS Marketplace 구독 관리 \(p. 83\)](#)

AMI 판매

AWS Marketplace를 사용하여 AMI를 판매할 수 있습니다. AWS Marketplace는 조직적인 쇼핑 환경을 제공합니다. 또한 AWS Marketplace에서는 Amazon EBS 지원 AMI, 예약 인스턴스 및 스팟 인스턴스와 같은 AWS 기능도 지원합니다.

AWS Marketplace에서 AMI를 판매하는 방법에 대한 자세한 내용은 [AWS Marketplace에서의 판매](#) 단원을 참조하십시오.

유료 AMI 찾기

구입 가능한 AMI를 찾는 방법은 다양합니다. 예를 들어, [AWS Marketplace](#), Amazon EC2 콘솔 또는 명령줄을 사용할 수 있습니다. 또는 개발자가 유료 AMI에 대한 정보를 제공할 수 있습니다.

콘솔을 사용하여 유료 AMI 찾기

콘솔을 사용하여 유료 AMI를 찾으려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 AMI를 선택합니다.
3. 퍼블릭 이미지를 첫 필터로 선택합니다.
4. 검색 창에서 소유자, AWS Marketplace를 차례로 선택합니다.
5. 제품 코드를 알고 있는 경우 제품 코드를 선택한 다음 제품 코드를 입력합니다.

AWS Marketplace를 사용하여 유료 AMI 찾기

AWS Marketplace를 사용하여 유료 AMI를 찾으려면

1. Open [AWS Marketplace](#).
2. 검색 상자에 운영 체제의 이름을 입력하고 이동을 클릭합니다.
3. 결과 범위를 더 자세히 지정하려면 범주 또는 필터 중 하나를 사용합니다.
4. 각 제품에는 제품 유형(AMI 또는 Software as a Service)으로 레이블로 지정됩니다.

Windows PowerShell용 도구를 사용하여 유료 AMI 찾기

다음 [Get-EC2Image](#) 명령을 사용하여 유료 AMI를 찾을 수 있습니다.

```
PS C:\> Get-EC2Image -Owner aws-marketplace
```

유료 AMI의 출력에는 제품 코드가 포함되어 있습니다.

ProductCodeId	ProductCodeType
----- <i>product_code</i>	----- marketplace

제품 코드를 알고 있는 경우 제품 코드별로 결과를 필터링할 수 있습니다. 이 예제는 지정된 제품 코드가 포함된 최신 AMI를 반환합니다.

```
PS C:\> (Get-EC2Image -Owner aws-marketplace -Filter @{"Name"="product-code"; "Value"="product_code"} | sort CreationDate -Descending | Select-Object -First 1).ImageId
```

AWS CLI를 사용하여 유료 AMI 찾기

다음 [describe-images](#) 명령(AWS CLI)을 사용하여 유료 AMI를 찾을 수 있습니다.

```
aws ec2 describe-images
```

```
--owners aws-marketplace
```

이 명령은 유료 AMI에 대한 제품 코드를 포함하여 각 AMI를 설명하는 다양한 정보를 반환합니다. `describe-images`의 출력에는 다음과 같은 제품 코드 항목이 포함됩니다.

```
"ProductCodes": [  
    {  
        "ProductCodeId": "product_code",  
        "ProductCodeType": "marketplace"  
    }  
,
```

제품 코드를 알고 있는 경우 제품 코드별로 결과를 필터링할 수 있습니다. 이 예제는 지정된 제품 코드가 포함된 최신 AMI를 반환합니다.

```
aws ec2 describe-images  
    --owners aws-marketplace \  
    --filters "Name=product-code,Values=product_code" \  
    --query "sort_by(Images, &CreationDate)[-1].[ImageId]"
```

유료 AMI 구입

AMI를 사용하여 인스턴스를 시작하려면 유료 AMI(구입)에 가입해야 합니다.

대개 유료 AMI 소유자가 가격과 해당 AMI를 구입할 수 있는 링크를 비롯하여 AMI에 대한 정보를 제공합니다. 링크를 클릭하면 AWS에 로그인하라는 메시지가 표시되고 그런 다음 AMI를 구입할 수 있습니다.

콘솔을 사용하여 유료 AMI 구입

Amazon EC2 시작 마법사를 사용하여 유료 AMI를 구입할 수 있습니다. 자세한 내용은 [AWS Marketplace 인스턴스 시작 \(p. 398\)](#) 단원을 참조하십시오.

AWS Marketplace를 사용하여 제품 구독

AWS Marketplace를 사용하려면 AWS 계정이 있어야 합니다. AWS Marketplace 제품에서 인스턴스를 시작하려면 Amazon EC2 서비스를 사용하도록 가입되어 있고, 인스턴스를 시작할 제품을 구독해야 합니다. AWS Marketplace에서 제품을 구독하는 방법은 두 가지입니다.

- AWS Marketplace 웹 사이트: 1-Click 배포 기능을 사용하여 미리 구성된 소프트웨어를 빠르게 시작할 수 있습니다.
- Amazon EC2 시작 마법사: AMI를 검색하고 마법사에서 직접 인스턴스를 시작할 수 있습니다. 자세한 내용은 [AWS Marketplace 인스턴스 시작 \(p. 398\)](#) 단원을 참조하십시오.

인스턴스에 대한 제품 코드 가져오기

인스턴스 메타데이터를 사용하여 인스턴스에 대한 AWS Marketplace 제품 코드를 검색할 수 있습니다. 메타데이터 검색에 대한 자세한 내용은 [인스턴스 메타데이터 및 사용자 데이터 \(p. 576\)](#) 단원을 참조하십시오.

제품 코드를 검색하려면 다음 명령을 사용합니다.

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/product-codes
```

인스턴스에 제품 코드가 있는 경우 Amazon EC2에서 해당 코드를 반환합니다.

유료 지원 사용

개발자가 Amazon EC2를 사용하여 소프트웨어 또는 파생 AMI를 지원할 수도 있습니다. 개발자는 사용자가 가입하여 사용할 수 있는 지원 제품을 생성할 수 있습니다. 지원 제품에 가입하는 동안 개발자가 제품 코드를 제공합니다. 이 제품 코드를 AMI와 연결해야 합니다. 개발자는 이 제품 코드를 사용하여 인스턴스가 지원 대상인지 확인할 수 있습니다. 또한 제품의 인스턴스를 실행할 때 개발자가 지정한 제품에 대한 조건에 따라 요금이 부과됩니다.

Important

지원 제품을 예약 인스턴스와 함께 사용할 수 없습니다. 항상 지원 제품의 판매자가 지정한 가격을 지불합니다.

제품 코드를 AMI와 연결하려면 다음 명령 중 하나를 사용합니다. 여기에서 `ami_id`는 AMI의 ID이고 `product_code`는 제품 코드입니다.

- [modify-image-attribute\(AWS CLI\)](#)

```
aws ec2 modify-image-attribute --image-id ami_id --product-codes "product_code"
```

- [Edit-EC2ImageAttribute\(Windows PowerShell용 AWS 도구\)](#)

```
PS C:\> Edit-EC2ImageAttribute -ImageId ami_id -ProductCode product_code
```

제품 코드 속성을 설정한 후 해당 속성을 변경하거나 제거할 수 없습니다.

유료 및 지원된 AMI에 대한 청구서

매월 말 그 달에 사용한 유료 또는 지원된 AMI에 대해 신용 카드로 청구되는 금액을 이메일로 수신하게 됩니다. 이 청구서는 정기 Amazon EC2 청구서와는 별개입니다. 자세한 내용은 [Paying For AWS Marketplace Products](#)를 참조하십시오.

AWS Marketplace 구독 관리

AWS Marketplace 웹 사이트에서 구독 정보 확인, 공급업체의 사용 지침 보기, 구독 관리 등을 수행할 수 있습니다.

구독 정보를 확인하려면

1. [AWS Marketplace](#)에 로그인합니다.
2. Marketplace 계정(Your Marketplace Account)을 선택합니다.
3. 소프트웨어 구독 관리(Manage your software subscriptions)를 선택합니다.
4. 현재 구독이 모두 나열됩니다. 사용량 제한을 선택하여 제품 사용에 대한 특정 지침(예: 실행 중인 인스턴스에 연결하기 위한 사용자 이름)을 봅니다.

AWS Marketplace 구독을 취소하려면

1. 구독에서 실행 중인 모든 인스턴스를 종료해야 합니다.
 - a. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
 - b. 탐색 창에서 인스턴스를 선택합니다.
 - c. 인스턴스를 선택하고 작업, 인스턴스 상태, 종료를 선택합니다.
 - d. 확인 메시지가 나타나면 예, 종료를 선택합니다.

2. AWS Marketplace에 로그인하고 Marketplace 계정(Your Marketplace Account)을 선택한 후 소프트웨어 구독 관리(Manage your software subscriptions)를 선택합니다.
3. 구독 취소를 선택합니다. 취소를 확인하라는 메시지가 나타납니다.

Note

구독을 취소하면 해당 AMI에서 더 이상 인스턴스를 시작할 수 없습니다. AMI를 다시 사용하려면 AWS Marketplace 웹 사이트 또는 Amazon EC2 콘솔의 시작 마법사를 통해 해당 AMI를 다시 구독해야 합니다.

사용자 지정 Windows AMI 생성

Windows AMI를 생성하려면 기존의 Windows AMI에서 인스턴스를 시작하고, 이 인스턴스를 사용자 정의한 다음, 그 인스턴스에서 새 AMI를 생성합니다.

사용자 지정 Linux AMI를 생성하려면 해당 인스턴스의 볼륨 유형에 맞는 절차를 사용하십시오. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EBS 지원 Linux AMI 생성](#) 또는 [인스턴스 스토어 지원 Linux AMI 생성](#)을 참조하십시오.

주제

- [사용자 지정 AMI 생성의 작동 방식 \(p. 84\)](#)
- [실행 중인 인스턴스에서 Windows AMI 생성 \(p. 85\)](#)
- [Sysprep을 이용한 표준 Amazon 머신 이미지 생성 \(p. 87\)](#)

사용자 지정 AMI 생성의 작동 방식

우선 만들려는 AMI와 비슷한 AMI에서 인스턴스를 시작합니다. 인스턴스에 연결하여 인스턴스를 사용자 지정할 수 있습니다. 인스턴스를 원하는 방법으로 설정한 후 AMI를 생성하기 전에 인스턴스를 종단하여 데이터 무결성을 확인한 다음 이미지를 생성합니다. 는 자동으로 등록됩니다.

AMI 생성 프로세스 중에 Amazon EC2는 인스턴스의 루트 볼륨과 인스턴스에 연결된 다른 EBS 볼륨의 스냅샷을 생성합니다. AMI 등록을 해제하고 스냅샷을 삭제할 때까지는 스냅샷에 대한 요금이 부과됩니다. 자세한 내용은 [Windows AMI 등록 취소 \(p. 111\)](#) 단원을 참조하십시오. 인스턴스에 연결된 볼륨이 암호화된 경우 새 AMI는 Amazon EBS 암호화를 지원하는 인스턴스 유형에서만 시작됩니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

볼륨의 크기에 따라 AMI 생성 프로세스를 완료하는 데 몇 분 정도 걸리지만 경우에 따라 24시간까지 걸릴 수도 있습니다. AMI를 생성하기 전에 볼륨의 스냅샷을 생성하는 것이 더 효율적일 수 있습니다. 이처럼 AMI를 생성할 때 작은 충분적 스냅샷만 만들어야 프로세스가 더 빠르게 완료됩니다. 스냅샷을 만드는 데 걸리는 전체 시간은 동일하게 유지됩니다. 자세한 내용은 [Amazon EBS 스냅샷 생성 \(p. 969\)](#) 단원을 참조하십시오.

프로세스가 완료되면 인스턴스의 루트 볼륨에서 새 AMI 및 스냅샷이 생성됩니다. 새 AMI를 사용하여 인스턴스를 시작할 때 스냅샷을 사용하여 루트 볼륨에 대한 새 EBS 볼륨을 생성합니다.

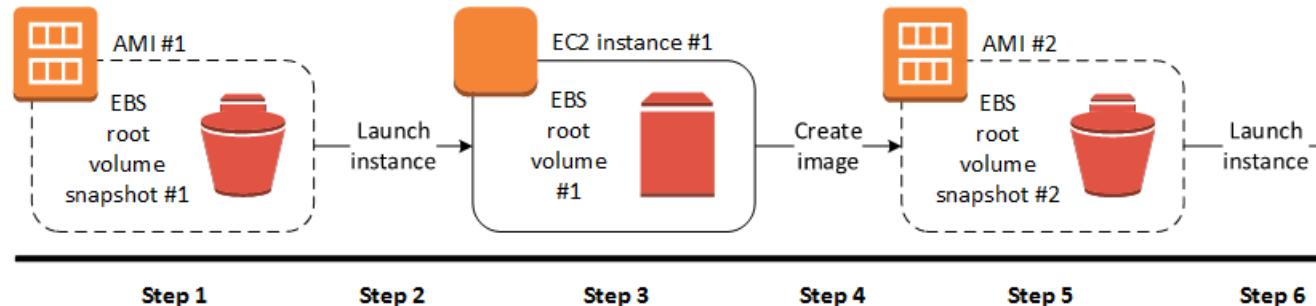
루트 디바이스 볼륨 외에도 인스턴스에 인스턴스 스토어 볼륨이나 Amazon EBS 볼륨을 추가하는 경우, 새 AMI에 대한 블록 디바이스 매핑과 새 AMI에서 자동으로 시작하는 인스턴스에 대한 블록 디바이스 매핑에 이러한 볼륨에 대한 정보가 포함됩니다. 새 인스턴스에 대한 블록 디바이스 매핑에 지정된 인스턴스 스토어 볼륨은 새 볼륨이므로 AMI를 생성하는 데 사용된 인스턴스에 대한 인스턴스 스토어 볼륨의 데이터가 포함되어 있지 않습니다. EBS 볼륨의 데이터는 유지됩니다. 자세한 내용은 [블록 디바이스 매핑 \(p. 1107\)](#) 단원을 참조하십시오.

Note

사용자 지정 AMI에서 새 인스턴스를 만들 때는 루트 볼륨과 추가 EBS 저장소 둘 다 초기화한 뒤에 프로덕션 환경에 배치해야 합니다. 자세한 정보는 [Amazon EBS 볼륨 초기화](#)를 참조하십시오.

실행 중인 인스턴스에서 Windows AMI 생성

AWS Management 콘솔 또는 명령줄을 사용하여 AMI를 생성할 수 있습니다. 다음 다이어그램은 실행 중인 EC2 인스턴스에서 를 만드는 프로세스를 요약한 것입니다. 기존 AMI로 시작하여 인스턴스를 시작한 다음 사용자 지정하고 해당 인스턴스에서 새 AMI를 생성합니다. 그런 다음 새 AMI의 인스턴스를 시작합니다. 다음 다이어그램의 단계는 아래 절차의 단계와 일치합니다. Windows 인스턴스를 이미 실행 중이라면 바로 4단계로 이동할 수 있습니다.



콘솔을 사용하여 인스턴스에서 AMI를 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 이미지와 AMI를 선택합니다.
3. 필터 옵션을 사용하여 필요에 맞는 Windows AMI만 표시되도록 AMI 목록의 범위를 지정합니다. 예를 들어 AWS에서 제공한 Windows AMI를 보려면 드롭다운 목록에서 퍼블릭 이미지를 선택합니다. 검색 창을 선택합니다. 메뉴에서 소유자를 선택하고 Amazon 이미지를 선택합니다. 메뉴에서 소스를 선택하고 필요한 Windows Server 버전에 따라 다음 중 하나를 입력합니다.
 - amazon/Windows_Server-2019
 - amazon/Windows_Server-2016
 - amazon/Windows_Server-2012
 - amazon/Windows_Server-2008

그 밖에 필요한 필터를 추가합니다. AMI를 골랐으면 해당 확인란을 선택합니다.

4. 시작을 선택합니다. 나머지 기본값을 그대로 두고 마법사를 계속 진행합니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오. 인스턴스가 준비되면 그 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
5. 인스턴스에서 다음과 같은 작업을 수행하여 인스턴스를 원하는 대로 사용자 지정할 수 있습니다.
 - 소프트웨어 및 애플리케이션 설치
 - 데이터 복사
 - 임시 파일 삭제, 하드 드라이브 조각 모음, 여유 공간 제로 클리어를 통한 시작 속도 향상
 - 추가 EBS 볼륨 연결
 - 새 계정 생성 후 Administrators 그룹에 추가

AMI를 공유하는 경우 기본 설정된 Administrator 암호를 제공할 이런 자격 증명을 통해 RDP 액세스를 허용할 수 있습니다.

- [Windows Server 2016 이상] EC2Launch를 사용하여 설정을 구성합니다. 시작할 때 무작위 암호가 생성되게 하려면 adminPasswordType 설정을 사용합니다. 자세한 내용은 [EC2Launch 구성 \(p. 491\)](#) 단원을 참조하십시오.
- [Windows Server 2012 R2 및 이전] EC2Config를 사용하여 설정을 구성합니다. 시작할 때 무작위 암호가 생성되게 하려면 Ec2SetPassword 플러그인을 활성화합니다. 그렇지 않으면 현재의 관리자 암호를 사용하게 됩니다. 자세한 내용은 [EC2Config 설정 파일 \(p. 504\)](#) 단원을 참조하십시오.

- [Windows Server 2008 R2] 인스턴스에서 RedHat 드라이버를 사용해 Xen 가상화 하드웨어에 액세스하는 경우, AMI를 생성하기 전에 Citrix 드라이버로 업그레이드해야 합니다. 자세한 내용은 [Windows Server 2008 및 2008 R2 인스턴스 업그레이드\(Redhat에서 Citrix PV로 업그레이드\) \(p. 531\)](#) 단원을 참조하십시오.
6. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다. 작업, 이미지, 이미지 생성을 선택합니다.

Tip

이 옵션이 비활성화되어 있다면 Amazon EBS 지원 인스턴스가 아님을 의미합니다.

7. 이미지에 고유한 이름을 지정하고 선택적으로 설명을 입력합니다(255자 이하).

기본적으로 Amazon EC2는 인스턴스를 종료하고, 연결된 볼륨의 스냅샷을 캡처하고, AMI를 생성하여 등록한 다음 인스턴스를 재부팅합니다. 인스턴스를 종료하지 않으려는 경우 재부팅 안 함을 선택합니다.

Warning

재부팅 안 함을 선택하는 경우 생성된 이미지의 파일 시스템 무결성을 보장할 수 없습니다.

(선택) 루트 볼륨, Amazon EBS 볼륨 및 인스턴스 스토어 볼륨을 필요에 따라 수정합니다. 다음 예를 참조하십시오.

- 루트 볼륨 크기를 변경하려면 유형 열의 루트 볼륨을 찾은 후 크기 필드에 값을 입력합니다.
- 인스턴스를 시작하는 데 사용되는 AMI의 블록 디바이스 매핑에서 지정된 Amazon EBS 볼륨을 표시하지 않으려면 목록에서 EBS 볼륨으로 이동한 다음 삭제를 선택합니다.
- Amazon EBS 볼륨을 추가하려면 새 볼륨 추가, 유형, EBS를 선택하고 필드를 작성합니다. 그런 다음 새 AMI에서 인스턴스를 시작하면 추가 볼륨이 인스턴스에 자동으로 연결됩니다. 빈 볼륨은 반드시 포맷하고 마운트해야 합니다. 스냅샷 기반 볼륨을 반드시 마운트해야 합니다.
- 인스턴스를 시작하는 데 사용되는 AMI의 블록 디바이스 매핑에서 지정된 인스턴스 스토어 볼륨을 표시하지 않으려면 목록에서 해당 볼륨으로 이동한 후 삭제를 선택합니다.
- 인스턴스 스토어 볼륨을 추가하려면 새 볼륨 추가, 유형, 인스턴스 스토어를 차례로 선택하고 디바이스 목록에서 디바이스 이름을 선택합니다. 새 AMI에서 인스턴스를 시작하면 추가 볼륨이 자동으로 시작되어 마운트됩니다. 이러한 볼륨에는 AMI를 기반으로 하는 실행 중인 인스턴스에 대한 인스턴스 스토어 볼륨의 데이터가 포함되어 있지 않습니다.

모두 마쳤으면 이미지 생성을 선택합니다.

8. AMI가 생성되는 동안 탐색 창에서 AMI를 선택하여 상태를 볼 수 있습니다. 처음에는 pending으로 표시됩니다. 몇 분 후 상태는 available로 변경됩니다.

(선택 사항) 탐색 창에서 스냅샷을 선택하여 새 AMI에 대해 생성된 스냅샷을 봅니다. 이 AMI에서 인스턴스를 시작할 때 이 스냅샷을 사용하여 루트 디바이스 볼륨을 생성합니다.

9. 새 AMI에서 인스턴스를 시작합니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오. 새로 실행되는 인스턴스에는 이전 단계에서 적용한 사용자 지정 사항이 모두 포함된 것은 물론, 인스턴스를 시작할 때 추가한 사용자 데이터(인스턴스가 시작할 때 실행되는 스크립트) 등 그 밖의 사용자 지정 사항도 포함되어 있습니다.

명령줄을 사용하여 인스턴스에서 AMI를 생성하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [create-image\(AWS CLI\)](#)
- [New-EC2Image\(Windows PowerShell용 AWS 도구\)](#)

Sysprep을 이용한 표준 Amazon 머신 이미지 생성

Sysprep(System Preparation) 도구는 Microsoft Windows의 사용자 지정 설치 반복 과정을 단순화합니다. Sysprep을 사용하여 표준화된 Amazon 머신 이미지(AMI)를 생성할 수 있습니다. 그런 다음 이 표준화된 이미지로부터 Windows용 새 Amazon EC2 인스턴스를 생성할 수 있습니다.

소프트웨어 및 설정으로 사전 설치 및 사전 구성된 안전한 맞추형 최신 “골드” 서버 이미지의 생성, 관리 및 배포를 자동화하려면 [EC2 이미지 빌더](#)를 사용하는 것이 좋습니다.

Sysprep을 사용하여 표준화된 AMI를 생성하는 경우 [EC2Launch v2 \(p. 461\)](#)와 함께 Sysprep을 실행하는 것이 좋습니다. EC2Config(Windows Server 2012 R2 및 이전 버전) 또는 EC2Launch(Windows Server 2016 및 이후 버전) 에이전트를 아직 사용하는 경우 아래의 EC2Config 및 EC2Launch와 함께 Sysprep 사용에 대한 설명서를 참조하세요.

Important

Sysprep을 사용하여 인스턴스 백업을 생성하지 마십시오. Sysprep은 시스템에 고유한 정보를 삭제하는데, 이는 인스턴스 백업에 의도하지 않은 결과를 낳을 수 있습니다.

Sysprep의 문제를 해결하려면 [??? 단원](#)을 참조하십시오.

목차

- [시작하기 전에 \(p. 87\)](#)
- [EC2Launch v2와 함께 Sysprep 사용 \(p. 87\)](#)
- [EC2Launch와 함께 Sysprep 사용 \(p. 90\)](#)
- [EC2Config와 함께 Sysprep 사용 \(p. 94\)](#)

시작하기 전에

- Sysprep을 수행하기 전에 Sysprep을 실행할 단일 관리자 계정을 제외하고 로컬 사용자 계정 및 계정 프로파일을 모두 제거하는 것이 좋습니다. 추가 계정 및 프로파일로 Sysprep를 수행하면 프로파일 데이터 손실 또는 Sysprep 완료 실패 등 예기치 못한 동작이 발생할 수 있습니다.
- Microsoft TechNet에서 [Sysprep](#)에 대해 더 자세히 알아보십시오.
- 어떤 [서버 역할이 Sysprep에 지원되는지](#) 알아보십시오.

EC2Launch v2와 함께 Sysprep 사용

이 단원에서는 이미지가 준비될 때 이루어지는 다양한 Sysprep 실행 단계와 EC2Launch v2 서비스에 의해 수행되는 작업의 세부 정보를 설명합니다. 또한 EC2Launch v2 서비스와 함께 Sysprep을 사용하여 표준화된 AMI를 생성하는 단계도 포함되어 있습니다.

EC2Launch v2와 함께 Sysprep 사용 주제

- [Sysprep 단계 \(p. 87\)](#)
- [Sysprep 작업 \(p. 88\)](#)
- [Sysprep 이후 \(p. 89\)](#)
- [EC2Launch v2와 함께 Sysprep 실행 \(p. 90\)](#)

Sysprep 단계

Sysprep은 다음과 같은 단계들을 실행합니다.

- **일반화:** 이 도구는 이미지 고유 정보 및 설정을 삭제합니다. 예를 몇 가지 들자면, Sysprep은 보안 식별자(SID), 컴퓨터 이름, 이벤트 로그, 특정 드라이버를 제거합니다. 이 단계가 완료되면 운영 체제(OS)는 AMI를 생성할 준비가 됩니다.

Note

`PersistAllDeviceInstalls` 설정이 `true`로 기본 설정되어 있기 때문에 Sysprep을 EC2Launch v2 서비스와 함께 실행할 때 시스템은 드라이버가 제거되는 것을 방지합니다.

- **특수화:** 플러그-앤-플레이가 컴퓨터를 스캔하여 검색된 장치에 대해 드라이버를 설치합니다. 이 도구는 컴퓨터 이름과 SID와 같은 OS 요건을 생성합니다. 선택 사항으로, 이 단계에서는 명령을 실행할 수 있습니다.
- **Out-of-Box Experience(OOBE):** 시스템은 Windows 설치 측면 버전을 실행하여 사용자에게 시스템 언어, 표준 시간대, 등록된 조직과 같은 정보를 입력하도록 요청합니다. EC2Launch v2와 함께 Sysprep을 실행할 때 응답 파일은 이 단계를 자동으로 실행합니다.

Sysprep 작업

Sysprep과 EC2Launch v2는 이미지를 준비할 때 다음 작업을 수행합니다.

1. EC2Launch 설정(EC2Launch settings) 대화 상자에서 Sysprep을 이용해 종료(Shutdown with Sysprep)를 선택하면 시스템은 `ec2launch sysprep` 명령을 실행합니다.
2. EC2Launch v2는 `HKEY_USERS\.DEFAULT\Control Panel\International\LocaleName`에서 레지스트리 값을 읽어 `unattend.xml` 파일의 내용을 편집합니다. 이 파일은 `C:\ProgramData\Amazon\EC2Launch\sysprep` 디렉터리에 있습니다.
3. 시스템은 `BeforeSysprep.cmd`를 실행합니다. 이 명령은 다음과 같은 레지스트리 키를 생성합니다.

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 1 /f
```

레지스트리 키는 RDP 연결이 다시 활성화될 때까지 RDP 연결을 비활성화합니다. RDP 연결을 비활성화하는 것은 필수적인 보안 조치입니다. 왜냐하면 Sysprep이 실행된 이후 첫 번째 부트 세션 동안 RDP가 연결을 허용하고 관리자 암호가 비어 있는 짧은 시간이 있기 때문입니다.

4. EC2Launch v2 서비스는 다음 명령을 실행하여 Sysprep을 호출합니다.

```
sysprep.exe /oobe /generalize /shutdown /unattend: "C:\ProgramData\Amazon\EC2Launch\sysprep\unattend.xml"
```

일반화 단계

- EC2Launch v2는 컴퓨터 이름과 SID 같은 이미지 고유 정보 및 설정을 삭제합니다. 인스턴스가 도메인의 멤버인 경우에는 도메인에서 삭제됩니다. `unattend.xml` 응답 파일은 이 단계에 영향을 미치는 다음의 설정을 포함합니다.
 - `PersistAllDeviceInstalls`: 이 설정은 Windows 설치로 하여금 장치를 제거하고 재구성하지 못하도록 함으로써 이미지 준비 과정을 가속화하는데, 이는 Amazon AMI는 특정 드라이버가 실행되는 것을 요구하고 그 드라이버들을 재검색하는 데 시간이 걸리기 때문입니다.
 - `DoNotCleanUpNonPresentDevices`: 이 설정은 현재 존재하지 않는 장비들에 대한 플러그-앤-플레이 정보를 담고 있습니다.
- Sysprep은 AMI를 생성하기 위한 준비를 하는 과정에서 OS를 종료합니다. 시스템은 새 인스턴스를 시작하거나 원본 인스턴스를 시작합니다.

특수화 단계

시스템은 컴퓨터 이름과 SID와 같은 OS 특정 요건을 생성합니다. 시스템은 또한 `unattend.xml` 응답 파일에 지정된 구성에 기반을 두어 다음 작업을 수행합니다.

- `CopyProfile`: Sysprep을 구성해 내장된 관리자 프로파일을 비롯한 모든 사용자 프로파일을 삭제할 수 있습니다. 이 설정은 내장된 관리자 계정을 보유하고 있어서 계정에 대한 어떤 사용자 지정도 새 이미지로 전달됩니다. 기본 값은 `true`입니다.

CopyProfile은 기본 프로파일을 기준의 로컬 관리자 프로파일로 바꿉니다. Sysprep를 실행한 후 로그인한 모든 계정은 첫 로그인 시 해당 프로파일 및 콘텐츠의 사본을 받게 됩니다.

새 이미지로 전달하고자 하는 사용자 프로파일에 대한 특정한 사용자 지정이 없다면 이 설정을 `False`로 변경하세요. Sysprep은 모든 사용자 프로파일을 삭제할 것입니다. 이는 시간과 디스크 공간을 절약해줍니다.

- `TimeZone`: 표준 시간대는 UTC(협정 세계시)로 기본 설정되어 있습니다.
- `Synchronous command with order 1`: 시스템은 다음 명령을 실행해 관리자 계정을 활성화하고 암호 요건을 지정합니다.

```
net user Administrator /ACTIVE:YES /LOGONPASSWORDCHG:NO /EXPIRES:NEVER /  
PASSWORDREQ:YES
```

- `Synchronous command with order 2`: 시스템은 관리자 암호를 암호화합니다. 이 보안 조치는 `ec2setpassword` 설정을 활성화하지 않으면 Sysprep이 완료된 후 인스턴스에 액세스할 수 없도록 설계되어 있습니다.

```
C:\Program Files\Amazon\Ec2ConfigService\ScramblePassword.exe" -u Administrator
```

- `Synchronous command with order 3`: 시스템은 다음 명령을 실행합니다.

```
C:\Program Files\Amazon\Ec2ConfigService\Scripts\SysprepSpecializePhase.cmd
```

이 명령은 다음과 같은 레지스트리 키를 추가하여 RDP를 재활성화합니다.

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /  
fDenyTSConnections /t REG_DWORD /d 0 /f
```

OOBE 단계

1. EC2Launch v2 응답 파일을 사용하여 다음과 같은 구성을 지정합니다.

- <InputLocale>en-US</InputLocale>
- <SystemLocale>en-US</SystemLocale>
- <UILanguage>en-US</UILanguage>
- <UserLocale>en-US</UserLocale>
- <HideEULAPage>true</HideEULAPage>
- <HideWirelessSetupInOOBE>true</HideWirelessSetupInOOBE>
- <ProtectYourPC>3</ProtectYourPC>
- <BluetoothTaskbarIconEnabled>false</BluetoothTaskbarIconEnabled>
- <TimeZone>UTC</TimeZone>
- <RegisteredOrganization>Amazon.com</RegisteredOrganization>
- <RegisteredOwner>EC2</RegisteredOwner>

Note

일반화 및 특수화 단계에서 EC2Launch v2는 OS의 상태를 모니터링합니다. EC2Launch v2는 OS가 Sysprep 단계에 있다는 것을 탐지하면 시스템 로그에 다음 메시지를 출력합니다.
`Windows is being configured. SysprepState=IMAGE_STATE_UNDEPLOYABLE`

2. 시스템이 EC2Launch v2를 실행합니다.

Sysprep 이후

Sysprep이 완료된 후 EC2Launch v2는 콘솔에 다음 메시지를 출력합니다.

Windows sysprep configuration complete.

그 다음에 EC2Launch v2는 다음 작업을 수행합니다.

1. agent-config.yml 파일의 내용을 읽고 구성된 작업을 실행합니다.
2. preReady 스테이지의 모든 작업을 실행합니다.
3. 실행이 완료된 후에는 Windows is ready라는 메시지를 인스턴스 시스템 로그에 전송합니다.
4. PostReady 스테이지의 모든 작업을 실행합니다.

EC2Launch v2에 대한 자세한 내용은 [EC2Launch v2를 사용한 Windows 인스턴스 구성 \(p. 461\)](#) 단원을 참조하세요.

EC2Launch v2와 함께 Sysprep 실행

Sysprep과 EC2Launch v2를 이용해 표준화된 AMI를 생성하려면 다음 절차를 수행하세요.

1. Amazon EC2 콘솔에서 복사하고자 하는 AMI의 위치를 지정하거나 [생성 \(p. 84\)](#)합니다.
2. 실행을 시작해서 Windows 인스턴스에 연결합니다.
3. 그 AMI를 사용자 지정합니다.
4. Windows 시작 메뉴에서 Amazon EC2Launch 설정(Amazon EC2Launch settings)을 검색하고 선택합니다. Amazon EC2Launch 설정(EC2Launch settings) 대화 상자의 옵션 및 설정에 대한 자세한 내용은 [EC2Launch v2 설정 \(p. 467\)](#) 단원을 참조하세요.
5. Sysprep을 이용해 종료(Shutdown with Sysprep) 또는 Sysprep을 이용하지 않고 종료(Shutdown without Sysprep)를 선택합니다.

Sysprep 실행 및 인스턴스 종료 여부 확인을 요청받을 때 예(Yes)를 클릭합니다. EC2Launch v2가 Sysprep을 실행합니다. 그런 다음 인스턴스에서 로그오프되고 인스턴스가 종료됩니다. Amazon EC2 콘솔에서 인스턴스(Instances) 페이지를 보면 인스턴스 상태가 running에서 stopping으로 바뀌었다가 다시 stopped로 바뀌는 것을 확인할 수 있습니다. 이 지점에서는 안전하게 현재 인스턴스에서 AMI를 생성할 수 있습니다.

다음 명령을 사용해서 명령줄에서 Sysprep 도구를 수동으로 호출할 수 있습니다.

"%programfiles%\amazon\ec2launch\ec2launch.exe" sysprep

EC2Launch와 함께 Sysprep 사용

EC2Launch는 AMI에서 이미지 준비 프로세스를 자동화하고 보호하는 기본 응답 파일과 배치 파일을 Sysprep에 제공합니다. 이 파일을 수정하는 것은 선택 사항입니다. 이 파일은 기본적으로 C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep 디렉터리에 있습니다.

Important

Sysprep을 사용하여 인스턴스 백업을 생성하지 마십시오. Sysprep은 시스템에 특정한 정보를 제거합니다. 이 정보를 제거하면 인스턴스 백업에서 의도하지 않은 결과가 발생할 수 있습니다.

EC2Launch와 함께 Sysprep 사용 주제

- [Sysprep에 대한 EC2Launch 응답 및 배치 파일 \(p. 90\)](#)
- [EC2Launch와 함께 Sysprep 실행 \(p. 91\)](#)
- [사용자 지정 AMI 시작 시 Server 2016 이후에 대한 메타데이터/KMS 경로 업데이트 \(p. 94\)](#)

Sysprep에 대한 EC2Launch 응답 및 배치 파일

Sysprep용 EC2Launch 응답 파일과 배치 파일에는 다음 내용이 포함됩니다.

Unattend.xml

이 파일은 기본 응답 파일입니다. SysprepInstance.ps1을 실행하거나 사용자 인터페이스에서 ShutdownWithSysprep을 선택하는 경우 시스템이 이 파일에서 설정을 읽습니다.

BeforeSysprep.cmd

EC2Launch가 Sysprep을 실행하기 전에 이 배치 파일이 명령을 실행하도록 사용자 지정합니다.

SysprepSpecialize.cmd

Sysprep 특수화 단계 중에 명령을 실행하려면 이 배치 파일을 사용자 지정합니다.

EC2Launch와 함께 Sysprep 실행

데스크톱 환경에서 Windows Server 2016 이후를 전체 설치할 때 EC2 Launch Settings 애플리케이션을 사용하여 EC2Launch와 함께 Sysprep을 수동으로 실행할 수 있습니다.

EC2Launch Settings 애플리케이션을 사용하여 Sysprep을 실행하려면

1. Amazon EC2 콘솔에서 Windows Server 2016 이후 AMI를 찾거나 생성합니다.
2. AMI에서 Windows 인스턴스를 시작합니다.
3. Windows 인스턴스에 연결하고 인스턴스를 사용자 지정합니다.
4. EC2LaunchSettings 애플리케이션을 검색하여 실행합니다. 기본적으로 C:\ProgramData\Amazon\EC2-Windows\Launch\Settings 디렉터리에 위치합니다.

Ec2 Launch Settings

General

Set Computer Name

- Set the computer name of the instance ip-<hex internal IP>. Disable this feature to persist your own computer name setting.

Set Wallpaper

- Overlay instance information on the current wallpaper.

Extend Boot Volume

- Extend OS partition to consume free space for boot volume.

Add DNS Suffix List

- Add DNS suffix list to allow DNS resolution of servers running in EC2 without providing the fully qualified domain name.

Handle User Data

- Execute user data provided at instance launch.
Note: This will be re-enabled when running shutdown with sysprep below.

Administrator Password

- Random (Retrieve from console)
 Specify (Temporarily store in config file)
 Do Nothing (Customize Unattend.xml for sysprep)

These changes will take effect on next boot if Ec2Launch script is scheduled. By default, it is scheduled by shutdown options below.

Sysprep

Sysprep is a Microsoft tool that prepares an image for multiple launches.

Ec2Launch Script Location: [Found](#)

92

- Run EC2Launch on every boot (instead of just the next boot).

5. 필요에 따라 옵션을 선택하거나 취소합니다. 이러한 설정은 `LaunchConfig.json` 파일에 저장되어 있습니다.
6. Administrator 암호에서 다음 중 하나를 수행합니다.
 - 임의(Random)를 선택합니다. EC2Launch는 암호를 생성하고 사용자의 키를 사용하여 암호를 암호화합니다. 인스턴스가 재부팅 또는 종지되었다가 시작된 경우 이 암호가 그대로 유지되도록 시스템은 인스턴스가 시작된 후 이 설정을 비활성화합니다.
 - 지정(Specify)을 선택하고 시스템 요구 사항을 충족하는 암호를 입력합니다. 암호는 `LaunchConfig.json`에 일반 텍스트로 저장되며 Sysprep에서 관리자 암호를 설정한 후에 삭제됩니다. 지금 종료하면 암호는 지금 바로 설정됩니다. EC2Launch는 사용자의 키를 사용하여 암호를 암호화합니다.
 - `DoNothing`을 선택하고 `unattend.xml` 파일에 암호를 지정합니다. `unattend.xml`에 암호를 지정하지 않으면 관리자 계정이 비활성화됩니다.
7. Sysprep을 이용해 종료(Shutdown with Sysprep)를 선택합니다.

EC2Launch를 사용하여 Sysprep을 수동으로 실행하려면

1. Amazon EC2 콘솔에서 복제하려는 Windows Server 2016 이후 Datacenter 에디션 AMI를 찾거나 만듭니다.
2. 실행을 시작해서 Windows 인스턴스에 연결합니다.
3. 인스턴스를 사용자 지정합니다.
4. `LaunchConfig.json` 파일에서 설정을 지정합니다. 이 파일은 기본적으로 `C:\ProgramData\Amazon\EC2-Windows\Launch\Config` 디렉터리에 위치합니다.

`adminPasswordType`에 대해 다음 값 중 하나를 지정할 수 있습니다.

Random

EC2Launch는 암호를 생성하고 사용자의 키를 사용하여 암호를 암호화합니다. 인스턴스가 재부팅 또는 종지되었다가 시작된 경우 이 암호가 그대로 유지되도록 시스템은 인스턴스가 시작된 후 이 설정을 비활성화합니다.

Specify

`adminPassword`에 지정한 암호가 EC2Launch에 사용됩니다. 암호가 시스템 요구 사항에 맞지 않으면 EC2Launch에서 임의의 암호를 대신 생성합니다. 암호는 `LaunchConfig.json`에 일반 텍스트로 저장되며 Sysprep에서 관리자 암호를 설정한 후에 삭제됩니다. EC2Launch는 사용자의 키를 사용하여 암호를 암호화합니다.

DoNothing

`unattend.xml` 파일에 지정한 암호가 EC2Launch에 사용됩니다. `unattend.xml`에 암호를 지정하지 않으면 관리자 계정이 비활성화됩니다.

5. (선택 사항) `unattend.xml` 및 기타 구성 파일에서 설정을 지정합니다. 설치에 참가하려는 경우 이 파일을 변경할 필요가 없습니다. 파일은 기본적으로 `C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep` 디렉터리에 위치합니다.
6. Windows PowerShell에서 `./InitializeInstance.ps1 -Schedule`을 실행합니다. 스크립트는 기본적으로 `C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts` 디렉터리에 위치합니다. 이 스크립트는 다음 부팅 중에 초기화하도록 인스턴스를 예약합니다. 다음 단계에서 `SysprepInstance.ps1` 스크립트를 실행하기 전에 이 스크립트를 실행해야 합니다.
7. Windows PowerShell에서 `./SysprepInstance.ps1`을 실행합니다. 스크립트는 기본적으로 `C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts` 디렉터리에 위치합니다.

인스턴스에서 로그오프되고 인스턴스가 종료됩니다. Amazon EC2 콘솔에서 인스턴스 페이지를 보면 인스턴스 상태가 `running`에서 `stopping`으로 바뀌었다가 다시 `stopped`로 바뀌는 것을 확인할 수 있습니다. 이 시점에서는 이 인스턴스에서 안전하게 AMI를 생성할 수 있습니다.

사용자 지정 AMI 시작 시 Server 2016 이후에 대한 메타데이터/KMS 경로 업데이트

사용자 지정 AMI를 시작할 때 Server 2016 이후에 대한 메타데이터/KMS 경로를 업데이트하려면 다음과 같이 하십시오.

- EC2LaunchSettings GUI(C:\ProgramData\Amazon\EC2-Windows\Launch\Settings\Ec2LaunchSettings.exe)를 실행하고 Sysprep을 사용하여 종료하는 옵션을 선택합니다.
- EC2LaunchSettings를 실행하고 Sysprep 없이 종료한 다음 AMI를 생성합니다. 그러면 다음번 부팅할 때 EC2 시작 초기화 작업을 실행하도록 설정되며, 해당 인스턴스의 서브넷을 토대로 경로가 설정됩니다.
- [PowerShell \(p. 491\)](#)에서 AMI를 생성하기 전에 EC2 시작 초기화 작업을 수동으로 다시 예약합니다.

Important

작업을 다시 예약하기 전에 기본 암호 재설정 동작을 적어 두십시오.

- Windows 정품 인증 또는 인스턴스 메타데이터 오류와의 통신이 발생하는 실행 중인 인스턴스에서 경로를 업데이트하려면 “[Windows를 정품 인증할 수 없음](#)” (p. 1226)을 참조하세요.

EC2Config와 함께 Sysprep 사용

이 단원에서는 이미지가 준비될 때 이루어지는 다양한 Sysprep 실행 단계와 EC2Config 서비스에 의해 수행되는 작업의 세부 정보를 설명합니다. 또한 EC2Config 서비스와 함께 Sysprep을 사용하여 표준화된 AMI를 생성하는 단계도 포함되어 있습니다.

EC2Config와 함께 Sysprep 사용 주제

- [Sysprep 단계](#) (p. 87)
- [Sysprep 작업](#) (p. 94)
- [Sysprep 이후](#) (p. 97)
- [EC2Config 서비스와 함께 Sysprep 실행하기](#) (p. 97)

Sysprep 단계

Sysprep은 다음과 같은 단계들을 실행합니다.

- **일반화:** 이 도구는 이미지 고유 정보 및 설정을 삭제합니다. 예를 몇 가지 들자면, Sysprep은 보안 식별자 (SID), 컴퓨터 이름, 이벤트 로그, 특정 드라이버를 제거합니다. 이 단계가 완료되면 운영 체제(OS)는 AMI 를 생성할 준비가 됩니다.

Note

PersistAllDeviceInstalls가 true로 기본 설정되어 있기 때문에 Sysprep을 EC2Config 서비스와 함께 실행할 때 시스템은 드라이버들이 제거되는 것을 방지합니다.

- **특수화:** 플러그-앤-플레이가 컴퓨터를 스캔하여 검색된 장치에 대해 드라이버를 설치합니다. 이 도구는 컴퓨터 이름과 SID와 같은 OS 요건을 생성합니다. 선택 사항으로, 이 단계에서는 명령을 실행할 수 있습니다.
- **Out-of-Box Experience(OOBE):** 시스템은 Windows 설치 측면 버전을 실행하여 사용자에게 시스템 언어, 표준 시간대, 등록된 조직과 같은 정보를 입력하도록 요청합니다. EC2Config와 함께 Sysprep을 실행할 때 응답 파일은 이 단계를 자동으로 실행합니다.

Sysprep 작업

Sysprep과 EC2Config 서비스는 이미지를 준비할 때 다음 작업을 수행합니다.

1. EC2 서비스 속성 대화 상자에서 Shutdown with Sysprep(Sysprep을 이용해 종료)을 선택하면, 시스템은 ec2config.exe –sysprep 명령을 실행합니다.

2. EC2Config 서비스는 `BundleConfig.xml` 파일의 내용을 읽습니다. 기본적으로 이 파일은 `c:\Program Files\Amazon\Ec2ConfigService\Settings` 디렉터리에 위치합니다.

`BundleConfig.xml` 파일에는 다음 설정이 포함되어 있습니다. 이 설정은 변경할 수 있습니다.

- AutoSysprep: Sysprep을 자동적으로 사용할지 여부를 지정합니다. Sysprep을 EC2 Service Properties 대화 상자로부터 실행하는 경우에는 이 값을 변경할 필요가 없습니다. 기본 값은 `No`입니다.
- SetRDPCertificate: 원격 데스크톱 서버에 대한 자체 서명된 인증서를 설정합니다. 이렇게 함으로써 원격 데스크톱 프로토콜(RDP)을 안전적으로 이용해 인스턴스에 연결할 수 있습니다. 새 인스턴스가 인증서를 사용해야 하는 경우 이 값을 `Yes`로 변경합니다. Windows Server 2008 또는 Windows Server 2012 인스턴스는 자신의 고유한 인증서를 생성할 수 있으므로 이 설정은 이 인스턴스에서는 사용되지 않습니다. 기본 값은 `No`입니다.
- SetPasswordAfterSysprep: 새로 실행된 인스턴스에 무작위 암호를 설정하고 이를 사용자 실행 키로 암호화하고 암호화된 암호를 콘솔에 출력합니다. 새 인스턴스에 무작위의 암호화된 암호를 설정해서는 안 되는 경우에는 설정 값을 `No`로 변경합니다. 기본 값은 `Yes`입니다.
- PreSysprepRunCmd: 실행할 명령의 위치 명령은 기본적으로 `C:\Program Files\Amazon\Ec2ConfigService\Scripts\BeforeSysprep.cmd` 디렉터리에 위치합니다.

3. 시스템은 `BeforeSysprep.cmd`를 실행합니다. 이 명령은 다음과 같은 레지스트리 키를 생성합니다.

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 1 /f
```

레지스트리 키는 RDP 연결이 다시 활성화될 때까지 RDP 연결을 비활성화합니다. RDP 연결을 비활성화하는 것은 필수적인 보안 조치입니다. 왜냐하면 Sysprep이 실행된 이후 첫 번째 부트 세션 동안 RDP가 연결을 허용하고 관리자 암호가 비어 있는 짧은 시간이 있기 때문입니다.

4. EC2Config 서비스는 다음 명령을 실행하여 Sysprep을 호출합니다.

```
sysprep.exe /unattend: "C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml" /oobe /generalize /shutdown
```

일반화 단계

- 이 도구는 컴퓨터 이름과 SID 같은 이미지 고유 정보 및 설정을 삭제합니다. 인스턴스가 도메인의 멤버인 경우에는 도메인에서 삭제됩니다. `sysprep2008.xml` 응답 파일은 이 단계에 영향을 미치는 다음의 설정을 포함합니다.
 - PersistAllDeviceInstalls: 이 설정은 Windows 설치로 하여금 장치를 제거하고 재구성하지 못하도록 함으로써 이미지 준비 과정을 가속화하는데, 이는 Amazon AMI는 특정 드라이버가 실행되는 것을 요구하고 그 드라이버들을 재검색하는 데 시간이 걸리기 때문입니다.
 - DoNotCleanUpNonPresentDevices: 이 설정은 현재 존재하지 않는 장비들에 대한 플러그-앤플레이 정보를 담고 있습니다.
 - Sysprep은 AMI를 생성하기 위한 준비를 하는 과정에서 OS를 종료합니다. 시스템은 새 인스턴스를 시작하거나 원본 인스턴스를 시작합니다.

특수화 단계

시스템은 컴퓨터 이름과 SID와 같은 OS 특정 요건을 생성합니다. 시스템은 또한 `sysprep2008.xml` 응답 파일에 지정된 구성에 기반을 두어 다음 작업을 수행합니다.

- CopyProfile: Sysprep을 구성해 내장된 관리자 프로파일을 비롯한 모든 사용자 프로파일을 삭제할 수 있습니다. 이 설정은 내장된 관리자 계정을 보유하고 있어서 계정에 대한 어떤 사용자 지정도 새 이미지로 전달됩니다. 기본값은 `True`입니다.

`CopyProfile`은 기본 프로파일을 기존의 로컬 관리자 프로파일로 바꿉니다. Sysprep를 실행한 후 로그인한 모든 계정은 첫 로그인 시 해당 프로파일 및 콘텐츠의 사본을 받게 됩니다.

새 이미지로 전달하고자 하는 사용자 프로파일에 대한 특정한 사용자 지정이 없다면 이 설정을 False로 변경하십시오. Sysprep은 모든 사용자 프로파일을 삭제할 것입니다. 이는 시간과 디스크 공간을 절약해줍니다.

- TimeZone: 표준 시간대는 UTC(협정 세계시)로 기본 설정되어 있습니다.
- Synchronous command with order 1: 시스템은 다음 명령을 실행해 관리자 계정을 활성화하고 암호 요건을 지정합니다.

```
net user Administrator /ACTIVE:YES /LOGONPASSWORDCHG:NO /EXPIRES:NEVER /  
PASSWORDREQ:YES
```

- Synchronous command with order 2: 시스템은 관리자 암호를 암호화합니다. 이 보안 조치는 ec2setpassword 설정을 활성화하지 않으면 Sysprep이 완료된 후 인스턴스에 액세스할 수 없도록 설계되어 있습니다.

```
C:\Program Files\Amazon\Ec2ConfigService\ScramblePassword.exe" -u Administrator
```

- Synchronous command with order 3: 시스템은 다음 명령을 실행합니다.

```
C:\Program Files\Amazon\Ec2ConfigService\Scripts\SysprepSpecializePhase.cmd
```

이 명령은 다음과 같은 레지스트리 키를 추가하여 RDP를 재활성화합니다.

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v  
fDenyTSConnections /t REG_DWORD /d 0 /f
```

OOBE 단계

1. 시스템은 EC2Config 응답 파일을 이용해 다음과 같은 구성을 지정합니다.

- <InputLocale>en-US</InputLocale>
- <SystemLocale>en-US</SystemLocale>
- <UILanguage>en-US</UILanguage>
- <UserLocale>en-US</UserLocale>
- <HideEULAPage>true</HideEULAPage>
- <HideWirelessSetupInOOBE>true</HideWirelessSetupInOOBE>
- <NetworkLocation>Other</NetworkLocation>
- <ProtectYourPC>3</ProtectYourPC>
- <BluetoothTaskbarIconEnabled>false</BluetoothTaskbarIconEnabled>
- <TimeZone>UTC</TimeZone>
- <RegisteredOrganization>Amazon.com</RegisteredOrganization>
- <RegisteredOwner>Amazon</RegisteredOwner>

Note

일반화 및 특수화 단계에서 EC2Config 서비스는 OS의 상태를 모니터링합니다. EC2Config는 OS가 Sysprep 단계에 있다는 것을 탐지하면 시스템 로그에 다음 메시지를 출력합니다.

EC2ConfigMonitorState: Windows 0개가 구성 중입니다.

SysprepState=IMAGE_STATE_UNDEPLOYABLE

2. OOBE 단계가 완료되고 나면 시스템은 C:\Windows\Setup\Scripts\SetupComplete.cmd에서 SetupComplete.cmd를 실행합니다. 2015년 4월 이전의 Amazon 퍼블릭 AMI에서는 이 파일이 비어 있었고 이미지 상에서 아무 것도 실행하지 않았습니다. 2015년 4월 이후의 퍼블릭 AMI에서 이 파일은 call "C:\Program Files\Amazon\Ec2ConfigService\Scripts\PostSysprep.cmd"라는 값을 포함합니다.
3. 시스템은 PostSysprep.cmd를 실행하여 다음 작업을 수행합니다.
 - 로컬 관리자 암호가 만료되지 않도록 설정합니다. 로컬 관리자 암호가 만료되면, 관리자가 로그인할 수 없을 수도 있습니다.

- MSSQLServer 머신 이름(설치된 경우)를 설정하여 그 이름이 AMI와 동기화되도록 합니다.

Sysprep 이후

Sysprep이 완료된 후 EC2Config 서비스는 콘솔에 다음 메시지를 출력합니다.

```
Windows sysprep configuration complete.  
Message: Sysprep Start  
Message: Sysprep End
```

그 다음에 EC2Config는 다음 작업을 수행합니다.

- config.xml 파일의 내용을 읽고 활성화된 모든 플러그인을 나열합니다.
- 모든 “Before Windows is ready” 플러그인을 동시에 실행합니다.
 - Ec2SetPassword
 - Ec2SetComputerName
 - Ec2InitializeDrives
 - Ec2EventLog
 - Ec2ConfigureRDP
 - Ec2OutputRDPCert
 - Ec2SetDriveLetter
 - Ec2WindowsActivate
 - Ec2DynamicBootVolumeSize
- 실행이 완료된 후에는 “Windows is ready”라는 메시지를 인스턴스 시스템 로그에 전송합니다.
- 모든 “After Windows is ready” 플러그인을 동시에 실행합니다.
 - AWS CloudWatch logs
 - UserData
 - AWS 시스템 관리자 (시스템 관리자)

Windows 플러그인에 대한 자세한 내용은 [EC2Config 서비스를 사용한 Windows 인스턴스 구성 \(p. 496\)](#) 단원을 참조하십시오.

EC2Config 서비스와 함께 Sysprep 실행하기

Sysprep과 EC2Config 서비스를 이용해 표준화된 AMI를 생성하려면 다음 절차를 수행하십시오.

- Amazon EC2 콘솔에서 복사하고자 하는 AMI의 위치를 지정하거나 [생성 \(p. 84\)](#)합니다.
- 실행을 시작해서 Windows 인스턴스에 연결합니다.
- 그 AMI를 사용자 지정합니다.
- EC2Config 서비스 응답 파일에서 구성 설정을 다음과 같이 지정합니다.

```
C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml
```

- Windows 시작 메뉴에서 모든 프로그램을 선택한 후 EC2ConfigService 설정을 클릭합니다.
- Ec2 서비스 속성 대화 상자에서 이미지 탭을 선택합니다. Ec2 서비스 속성 대화 상자의 옵션 및 설정에 대한 자세한 내용은 [Ec2 서비스 속성 \(p. 496\)](#)을 참조하십시오.
- 관리자 암호에 대한 옵션을 선택하고 Shutdown with Sysprep(Sysprep을 이용해 종료) 또는 Shutdown without Sysprep(Sysprep을 사용하지 않고 종료)을 선택합니다. EC2Config는 선택한 암호 옵션에 기반하는 설정 파일을 편집합니다.
 - Random(무작위): EC2Config는 암호를 생성하고 이를 사용자의 키로 암호화한 다음, 암호화된 암호를 콘솔에 표시합니다. AWS는 첫 실행 후 이 설정을 비활성화하여 인스턴스가 재부팅되거나 중단되고 시작된 경우에도 해당 암호가 계속 유지되도록 합니다.

- 지정: 암호를 암호화되지 않은 형태(클리어 텍스트)로 Sysprep 응답 파일에 저장합니다. Sysprep은 다음에 실행될 때 관리자 암호를 설정합니다. 지금 종료하면 암호는 지금 바로 설정됩니다. 서비스가 다시 시작할 때 관리자 암호는 제거됩니다. 이 암호는 나중에 다시 확인할 수 없기 때문에 이를 꼭 기억해두십시오.

- Keep Existing(기존 유지): Sysprep이 실행 중일 때 또는 EC2Config가 재시작할 때, 관리자 계정에 대한 기존 암호를 변경시키지 않습니다. 이 암호는 나중에 다시 확인할 수 없기 때문에 이를 꼭 기억해두십시오.

8. 확인을 선택합니다.

Sysprep 실행 및 인스턴스 종료 여부 확인을 요청받을 때 예를 클릭합니다. EC2Config가 Sysprep을 실행하는 것을 확인할 수 있습니다. 그 다음 인스턴스에서 로그아웃되고 인스턴스는 종료됩니다. Amazon EC2 콘솔의 인스턴스 페이지를 보면 인스턴스 상태가 running에서 stopping, 그리고 마지막으로 stopped로 바뀌는 것을 확인할 수 있습니다. 이 지점에서는 안전하게 현재 인스턴스에서 AMI를 생성할 수 있습니다.

다음 명령을 사용해서 명령줄에서 Sysprep 도구를 수동으로 호출할 수 있습니다.

```
"%programfiles%\amazon\ec2configservice\"ec2config.exe -sysprep"
```

Note

CMD 셸이 이미 C:\Program Files\Amazon\EC2ConfigService\ 디렉터리에 있는 경우에는 명령에 큰 따옴표가 필요 없습니다.

그러나 Ec2ConfigService\Settings 폴더에 지정된 XML 파일 옵션이 올바른지 주의 깊게 확인할 필요가 있습니다. 이것이 올바르게 지정되지 않은 경우는 인스턴스에 연결하지 못할 수 있습니다. 설정 파일에 대한 자세한 내용은 [EC2Config 설정 파일 \(p. 504\)](#) 단원을 참조하십시오. 명령줄에서 Sysprep을 구성한 후 실행하는 예는 Ec2ConfigService\Scripts\InstallUpdates.ps1을 참조하십시오.

EBS 지원 AMI를 통한 암호화 사용

Amazon EBS 스냅샷의 지원을 받는 AMI에서는 Amazon EBS 암호화를 활용할 수 있습니다. 데이터 볼륨과 루트 볼륨 모두의 스냅샷을 암호화하고 AMI에 연결할 수 있습니다. 전체 EBS 암호화 지원을 통해 인스턴스를 시작하고 이미지를 복사할 수 있습니다. 이러한 작업을 위한 암호화 파라미터는 AWS KMS를 사용할 수 있는 모든 리전에서 지원됩니다.

암호화된 EBS 볼륨이 있는 EC2 인스턴스는 다른 인스턴스와 동일한 방법으로 AMIs에서 시작됩니다. 또한 암호화되지 않은 EBS 스냅샷이 지원하는 AMI에서 인스턴스를 시작할 때 시작하는 동안 해당 볼륨의 일부 또는 전체를 암호화할 수 있습니다.

EBS 볼륨처럼 AMI의 스냅샷을 기본 AWS Key Management Service 고객 마스터 키(CMK) 또는 지정한 고객 관리형 키로 암호화할 수 있습니다. 어느 경우든 선택한 키에 대한 사용 권한이 있어야 합니다.

암호화된 스냅샷이 있는 AMI는 AWS 계정 간에 공유할 수 있습니다. 자세한 내용은 [공유 AMI](#) 단원을 참조하십시오.

인스턴스 시작 시나리오

Amazon EC2 인스턴스는 AWS Management 콘솔을(를) 사용하거나 Amazon EC2 API 또는 CLI를 직접 사용하여 블록 디바이스 매핑을 통해 파라미터가 입력된 RunInstances 작업을 사용하여 AMI에서 시작됩니다. 블록 디바이스 매핑에 대한 자세한 내용은 [블록 디바이스 매핑](#) 단원을 참조하십시오. AWS CLI에서 블록 디바이스 매핑을 제어하는 예제는 [시작, 목록 및 EC2 인스턴스 종료](#) 단원을 참조하십시오.

기본적으로 명시적인 암호화 파라미터가 없는 경우 RunInstances 작업은 AMI 원본 스냅샷에서 EBS 볼륨을 복원하는 동안 AMI 원본 스냅샷의 기존 암호화 상태를 유지합니다. [암호화 기본 제공 \(p. 1036\)](#)을 활성화

하면 AMI에서 생성된 모든 볼륨(암호화된 스냅샷이든 암호화되지 않은 스냅샷이든)이 암호화됩니다. 암호화 기본 제공이 활성화되지 않은 경우, 인스턴스는 AMI의 암호화 상태를 유지합니다.

암호화 파라미터를 입력하여, 인스턴스를 시작하는 동시에 결과 볼륨에 새 암호화 상태를 적용할 수도 있습니다. 결과적으로 다음의 동작이 관찰됩니다.

암호화 파라미터 없이 시작

- 암호화가 기본적으로 활성화되지 않은 경우, 암호화되지 않은 스냅샷이 암호화되지 않은 볼륨으로 복원됩니다. 이런 경우 새로 생성된 모든 볼륨이 암호화됩니다.
- 소유한 암호화된 스냅샷은 동일한 CMK로 암호화된 볼륨으로 복원됩니다.
- 소유하지 않은 암호화된 스냅샷(예: AMI가 자신과 공유됨)은 AWS 계정의 기본 CMK로 암호화된 볼륨으로 복원됩니다.

암호화 파라미터를 입력하여 기본 동작을 재정의할 수 있습니다. 사용 가능한 파라미터는 `Encrypted` 및 `KmsKeyId`입니다. `Encrypted` 파라미터만 설정할 경우 그 결과는 다음과 같습니다.

`Encrypted`이(가) 설정되었지만 `KmsKeyId`이(가) 지정되지 않은 경우의 인스턴스 시작 동작

- 암호화되지 않은 스냅샷은 AWS 계정의 기본 CMK로 암호화된 EBS 볼륨으로 복원됩니다.
- 소유한 암호화된 스냅샷은 동일한 CMK로 암호화된 EBS 볼륨으로 복원됩니다. (즉, `Encrypted` 파라미터는 아무런 효과가 없습니다.)
- 소유하지 않은 암호화된 스냅샷(즉, AMI가 자신과 공유됨)은 AWS 계정의 기본 CMK로 암호화된 볼륨으로 복원됩니다. (즉, `Encrypted` 파라미터는 아무런 효과가 없습니다.)

`Encrypted` 및 `KmsKeyId` 파라미터를 모두 설정하면 암호화 작업에 대해 기본이 아닌 CMK를 지정할 수 있습니다. 결과는 다음 동작과 같습니다.

`Encrypted`와(과) `KmsKeyId`이(가) 모두 설정된 인스턴스

- 암호화되지 않은 스냅샷은 지정된 CMK로 암호화된 EBS 볼륨으로 복원됩니다.
- 암호화된 스냅샷은 원래의 CMK가 아니라 지정된 CMK로 암호화된 EBS 볼륨으로 복원됩니다.

`Encrypted` 파라미터를 설정하지 않고 `KmsKeyId`을(를) 제공하면 오류가 발생합니다.

다음 단원에서는 기본이 아닌 암호화 파라미터를 사용하여 AMI에서 인스턴스를 시작하는 예제를 볼 수 있습니다. 이러한 각각의 시나리오에서 `RunInstances` 작업에 입력된 파라미터에 의해 스냅샷에서 볼륨을 복원하는 동안 암호화 상태의 변경이 유발됩니다.

Note

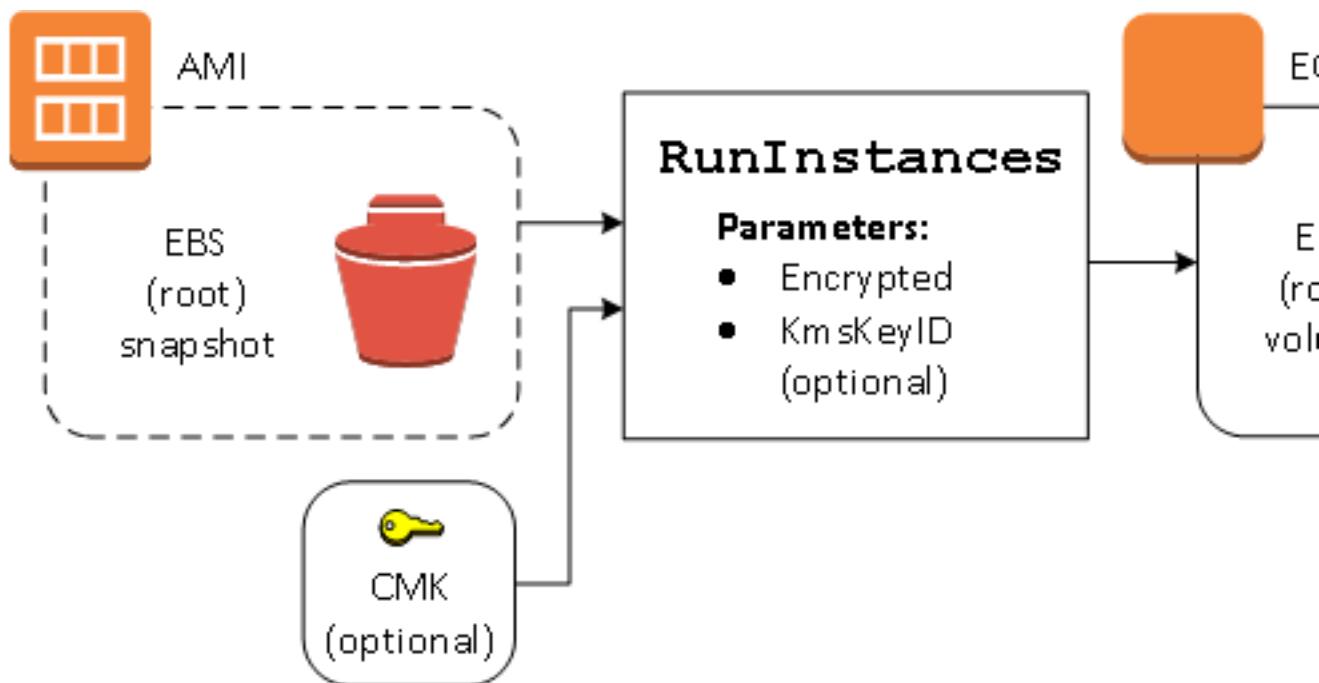
AMI에서 인스턴스를 시작하는 자세한 콘솔 절차는 [인스턴스 시작](#) 단원을 참조하십시오.

`RunInstances` API 설명서는 [RunInstances\(인스턴스 실행\)](#) 단원을 참조하십시오.

AWS Command Line Interface의 명령 `run-instances` 설명서는 [run-instances\(인스턴스 실행\)](#) 단원을 참조하십시오.

시작 중에 볼륨 암호화

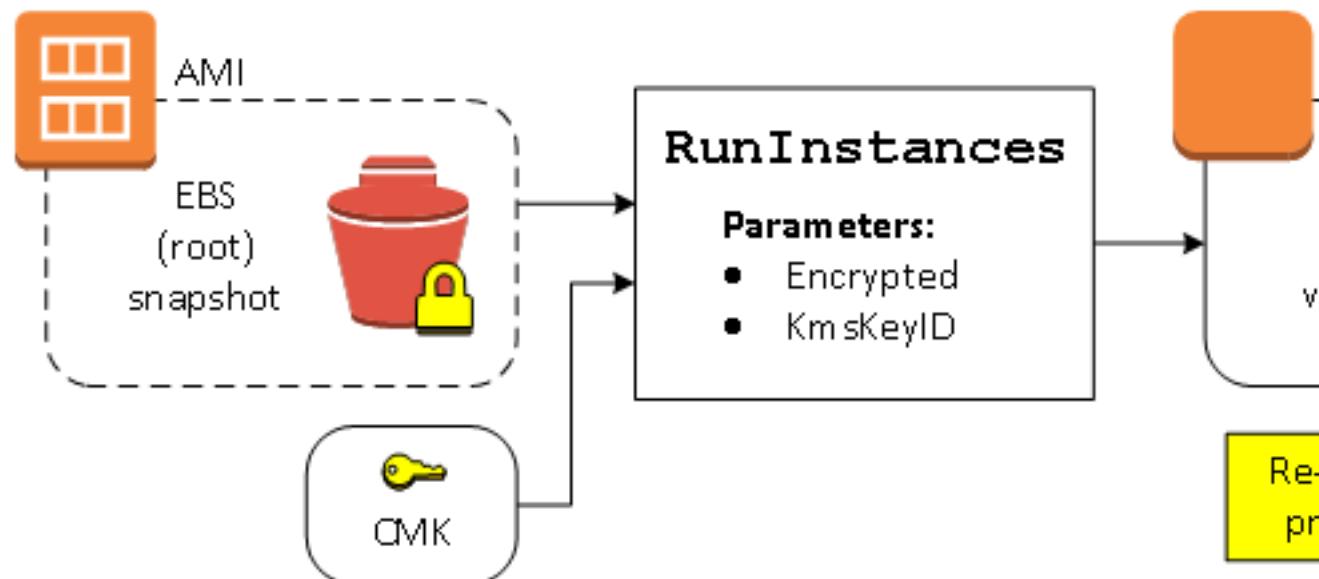
이 예제에서 암호화되지 않은 스냅샷이 지원하는 AMI는 암호화된 EBS 볼륨을 통해 EC2 인스턴스를 시작하는 데 사용됩니다.



Encrypted 파라미터만 사용하면 이 인스턴스의 볼륨이 암호화됩니다. **KmsKeyId** 파라미터는 선택 항목입니다. 키 ID를 지정하지 않을 경우 볼륨을 암호화하는 데 AWS 계정의 기본 CMK가 사용됩니다. 소유한 다른 CMK로 사본을 암호화하려면 **KmsKeyId** 파라미터를 입력합니다.

시작 중에 볼륨 재암호화

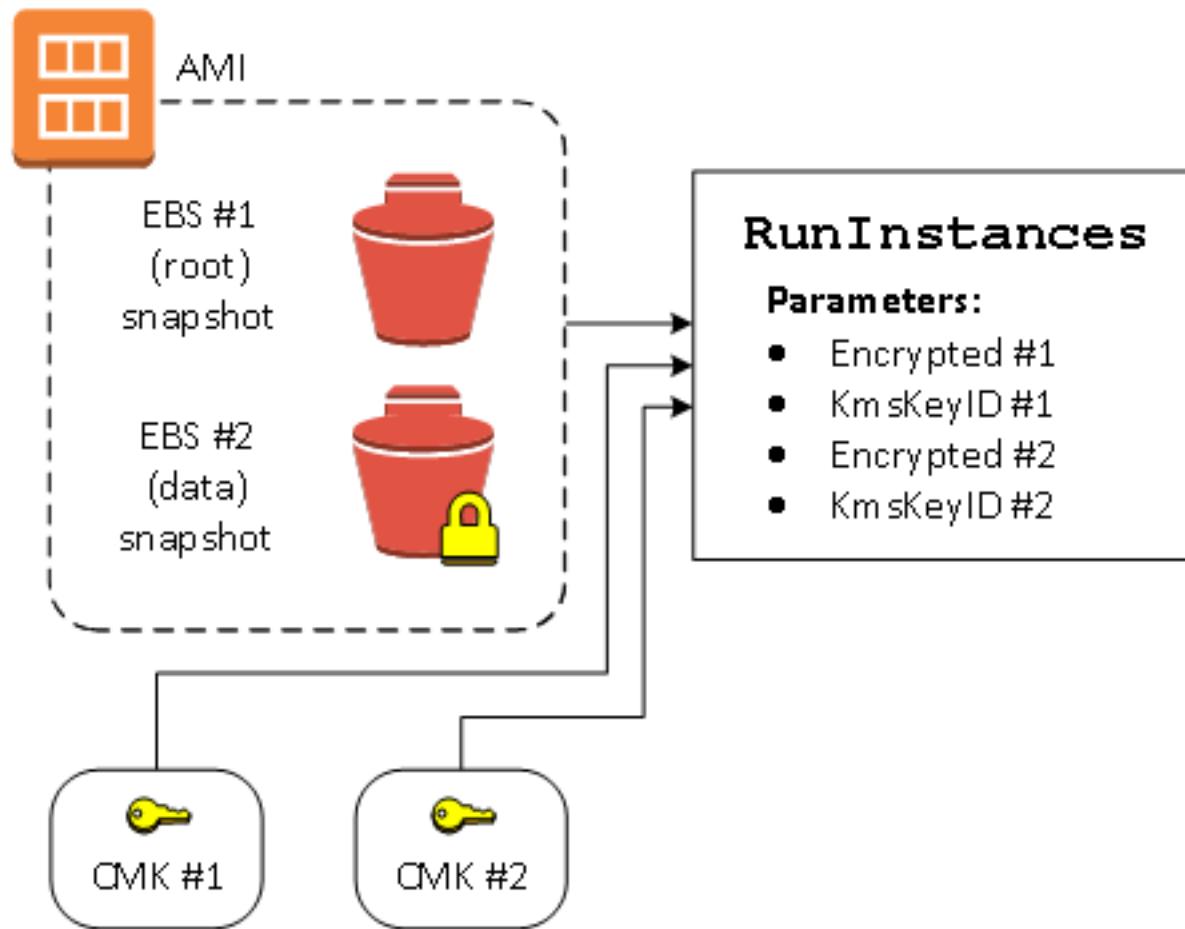
이 예제에서 암호화된 스냅샷이 지원하는 AMI는 새 CMK로 암호화된 EBS 볼륨을 통해 EC2 인스턴스를 시작하는 데 사용됩니다.



AMI를 소유한 상태에서 암호화 파라미터를 입력하지 않을 경우, 결과 인스턴스는 해당 스냅샷과 동일한 키로 암호화된 볼륨을 갖게 됩니다. AMI를 소유하지 않고 공유하며 암호화 파라미터를 입력하지 않을 경우, 볼륨이 기본 CMK로 암호화됩니다. 설명된 대로 암호화 파라미터를 입력할 경우, 볼륨이 지정된 CMK로 암호화됩니다.

시작 중에 여러 볼륨의 암호화 상태 변경

이 더 복잡한 예제에서 여러 스냅샷이 (각기 자체적인 암호화 상태를 통해) 지원하는 AMI는 새로 암호화된 볼륨과 재암호화된 볼륨을 통해 EC2 인스턴스를 시작하는 데 사용됩니다.



이 시나리오에서 `RunInstances` 작업에는 원본 스냅샷 각각에 대한 암호화 파라미터가 입력됩니다. 모든 가능한 암호화 파라미터를 지정하면, 결과 인스턴스는 AMI 소유 여부와 상관없이 동일합니다.

이미지 복사 시나리오

Amazon EC2 AMI은(는) AWS Management 콘솔을(를) 통하거나 직접 Amazon EC2 API 또는 CLI를 통해 `CopyImage` 작업을 사용해 복사됩니다.

기본적으로 명시적인 암호화 파라미터가 없는 경우, `copyImage` 작업은 복사 중에 AMI 원본 스냅샷의 기존 암호화 상태를 유지합니다. 암호화 파라미터를 입력하여, AMI를(를) 복사하는 동시에 연결된 EBS 스냅샷에 새 암호화 상태를 적용할 수도 있습니다. 결과적으로 다음의 동작이 관찰됩니다.

암호화 파라미터 없이 복사

- 암호화가 기본적으로 활성화되지 않은 경우, 암호화되지 않은 스냅샷이 또 다른 암호화되지 않은 스냅샷으로 복사됩니다. 이런 경우 새로 생성된 모든 스냅샷이 암호화됩니다.
- 소유한 암호화된 스냅샷은 동일한 키로 암호화된 스냅샷으로 복사됩니다.
- 소유하지 않은 암호화된 스냅샷(즉, AMI가 자신과 공유됨)은 AWS 계정의 기본 CMK로 암호화된 스냅샷으로 복사됩니다.

암호화 파라미터를 입력하여 이러한 모든 기본 동작을 재정의할 수 있습니다. 사용 가능한 파라미터는 **Encrypted** 및 **KmsKeyId**입니다. **Encrypted** 파라미터만 설정할 경우 그 결과는 다음과 같습니다.

Encrypted이(가) 설정되었지만 **KmsKeyId**이(가) 지정되지 않은 경우의 이미지 복사 동작

- 암호화되지 않은 스냅샷은 AWS 계정의 기본 CMK로 암호화된 스냅샷으로 복사됩니다.
- 암호화된 스냅샷은 동일한 CMK로 암호화된 스냅샷으로 복사됩니다. (즉, **Encrypted** 파라미터는 아무런 효과가 없습니다.)
- 소유하지 않은 암호화된 스냅샷(즉, AMI가 자신과 공유됨)은 AWS 계정의 기본 CMK로 암호화된 볼륨으로 복사됩니다. (즉, **Encrypted** 파라미터는 아무런 효과가 없습니다.)

Encrypted 및 **KmsKeyId** 파라미터를 모두 설정하면 암호화 작업에 대해 고객 관리형 CMK를 지정할 수 있습니다. 결과는 다음 동작과 같습니다.

Encrypted와(과) **KmsKeyId**이(가) 모두 설정된 경우의 이미지 복사 동작

- 암호화되지 않은 스냅샷은 지정된 CMK로 암호화된 스냅샷으로 복사됩니다.
- 암호화된 스냅샷은 원래의 CMK가 아니라 지정된 CMK로 암호화된 스냅샷으로 복사됩니다.

Encrypted 파라미터를 설정하지 않고 **KmsKeyId**을(를) 제공하면 오류가 발생합니다.

다음 단원에서는 기본이 아닌 암호화 파라미터를 사용하여 AMI를 복사하여 암호화 상태에 변경이 유발되는 예제를 제공합니다.

Note

AMI를 복사하는 자세한 콘솔 절차는 [AMI 복사](#) 단원을 참조하십시오.

`CopyImage` API 설명서는 [CopyImage\(이미지 복사\)](#) 단원을 참조하십시오.

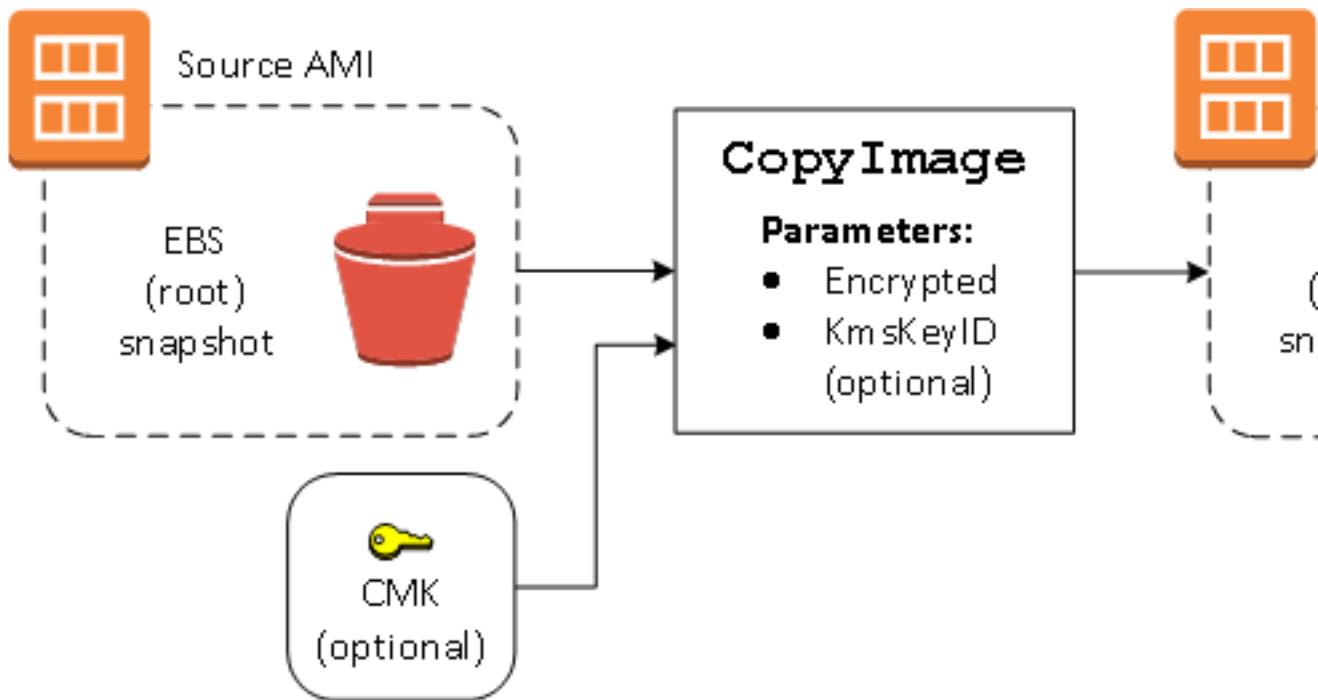
AWS Command Line Interface의 명령 `copy-image` 설명서는 [copy-image\(이미지 복사\)](#) 단원을 참조하십시오.

복사 중에 암호화되지 않은 이미지 암호화

이 시나리오에서는 암호화되지 않은 루트 스냅샷에서 지원하는 AMI가 암호화된 루트 스냅샷이 있는 AMI로 복사됩니다. `CopyImage` 작업은 CMK를 포함하여 두 암호화 파라미터를 사용하여 호출됩니다. 따라서 루트 스냅샷의 암호화 상태가 변경되므로 대상 AMI는 원본 스냅샷과 동일한 데이터를 포함하는 루트 스냅샷에 의해 지원되지만 지정된 키를 사용하여 암호화됩니다. 두 AMIs 모두의 스냅샷에 대한 스토리지 비용과 각 AMI에서 시작되는 인스턴스에 대한 비용이 발생합니다.

Note

[encryption by default\(암호화 기본 제공\)](#) (p. 1036)를 활성화하는 것은 AMI에서 모든 스냅샷에 대해 **Encrypted** 파라미터를 `true`로 설정하는 것과 동일한 효과를 가집니다.



`Encrypted` 파라미터를 설정하면 이 인스턴스에 대한 단일 스냅샷이 암호화됩니다. `KmsKeyId` 파라미터를 지정하지 않으면 스냅샷 복사본을 암호화하는 데 기본 CMK가 사용됩니다.

Note

여러 스냅샷으로 이미지를 복사하고 각 이미지의 암호화 상태를 개별적으로 구성할 수도 있습니다.

AMI 복사

AWS Management 콘솔, AWS Command Line Interface나 SDK 또는 Amazon EC2 API(모두 `CopyImage` 작업을 지원함)를 사용하여 AWS 리전 내부 또는 전체에서 Amazon 머신 이미지(AMI)를 복사할 수 있습니다. 사용자는 Amazon EBS 지원 AMIs와 인스턴스 스토어 지원 AMIs를 모두 복사할 수 있습니다. 암호화된 스냅샷을 사용하여 AMI를 복사하고 복사 과정 중 암호화 상태도 변경할 수 있습니다.

원본 AMI를 복사하면 동일하지만 고유의 식별자로 구별되는 대상 AMI가 생성됩니다. Amazon EBS 지원 AMI의 경우, 동일하지만 구분된 대상 스냅샷으로 각 지원 스냅샷이 복사되도록 기본 설정되어 있습니다. (유일한 예외는 스냅샷의 암호화 또는 재암호화를 선택할 때입니다.) 대상 AMI에 영향을 미치지 않고 원본 AMI를 변경하거나 다시 등록할 수 있습니다. 반대의 경우도 마찬가지입니다.

AMI 복사 시 부과되는 요금은 없습니다. 그러나 표준 스토리지 및 데이터 전송 요금은 적용됩니다. EBS 지원 AMI를 복사하면 추가 EBS 스냅샷의 스토리지에 대한 요금이 발생합니다.

AWS에서는 시작 권한, 사용자 정의 태그 또는 Amazon S3 버킷 권한이 원본 AMI에서 새 AMI로 복사되지 않습니다. 복사 작업이 완료된 후 시작 권한, 사용자 정의 태그 및 Amazon S3 버킷 권한을 새 AMI에 적용할 수 있습니다.

사용자가 직접 가져왔거나 사용자와 공유되었는지 여부와 상관없이 AWS Marketplace에서 가져온 AMI를 복사할 수 없습니다. 대신에 AWS Marketplace AMI를 사용하여 EC2 인스턴스를 시작한 다음, 해당 인스턴스에서 AMI를 생성합니다. 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오.

인스턴스 스토어 지원 AMI 복사 권한

IAM 사용자를 사용하여 인스턴스 스토어 지원 AMI를 복사하려면 사용자에게 s3:CreateBucket, s3:GetBucketAcl, s3>ListAllMyBuckets, s3:GetObject, s3:PutObject, s3:PutObjectAcl이라는 Amazon S3 권한이 있어야 합니다.

다음 예시 정책을 통해 사용자는 지정된 버킷의 AMI 원본을 지정된 리전에 복사할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "s3>ListAllMyBuckets",  
            "Resource": [  
                "arn:aws:s3::::*"  
            ]  
        },  
        {  
            "Effect": "Allow",  
            "Action": "s3:GetObject",  
            "Resource": [  
                "arn:aws:s3::::ami-source-bucket/*"  
            ]  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "s3>CreateBucket",  
                "s3:GetBucketAcl",  
                "s3:PutObjectAcl",  
                "s3:PutObject"  
            ],  
            "Resource": [  
                "arn:aws:s3::::amis-for-123456789012-in-us-east-1*"  
            ]  
        }  
    ]  
}
```

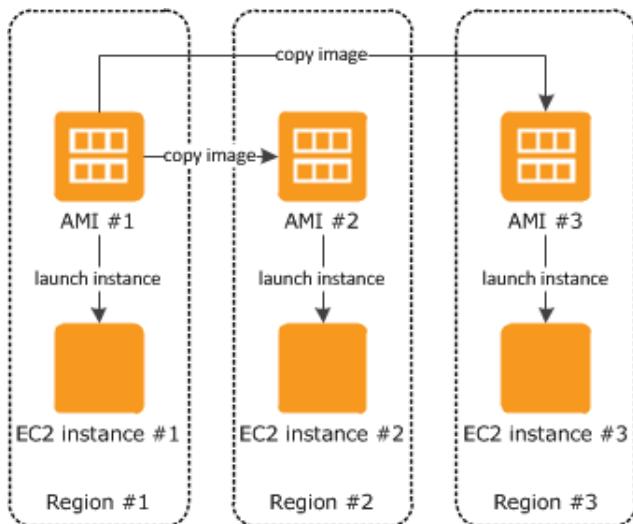
AMI 원본 버킷의 Amazon 리소스 이름(ARN)을 찾으려면 <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 열고 탐색 창에서 AMI를 선택하고 소스 옆에서 버킷 이름을 찾습니다.

교차 리전 복사

지리적으로 다른 리전 간에 AMI를 복사하면 다음과 같은 이점이 제공됩니다.

- 일관적인 글로벌 배포: 한 리전에서 다른 리전으로 AMI를 복사하면 동일한 AMI를 기반으로 하는 일관적인 인스턴스를 여러 리전에서 시작할 수 있습니다.
- 확장성: 사용자의 지역에 관계없이 요구 사항에 대응하는 글로벌 애플리케이션을 보다 손쉽게 설계하고 구축할 수 있습니다.
- 성능: 애플리케이션을 분산하여 성능을 높이고 애플리케이션의 핵심 구성 요소를 사용자에게 보다 가까이 들 수 있습니다. 또한 인스턴스 유형이나 여타 AWS 서비스와 같은 리전별 기능을 활용할 수 있습니다.
- 고가용성: 여러 AWS 리전을 포괄하는 애플리케이션을 설계하고 배포하여 가용성을 높일 수 있습니다.

다음 다이어그램은 원본 AMI 및 다른 리전에 복사된 두 개의 AMIs 간 관계와 각각에서 시작된 EC2 인스턴스를 보여 줍니다. AMI에서 인스턴스를 시작하는 경우 인스턴스는 AMI가 상주하는 동일한 리전에 상주합니다. 원본 AMI를 변경한 후 대상 리전의 AMIs에 변경 내용을 반영하려면 원본 AMI를 대상 리전으로 다시 복사해야 합니다.



먼저 인스턴스 스토어 지원 AMI를 리전에 복사하는 경우 해당 리전에 복사된 AMIs에 대한 Amazon S3 버킷이 생성됩니다. 해당 리전에 복사하는 인스턴스 스토어 지원 AMIs는 모두 이 버킷에 저장됩니다. 버킷 이름 형식은 `amis-for-account-in-region-hash`를 따릅니다. 예: `amis-for-123456789012-in-us-east-2-yhjmxvp6`.

사전 조건

AMI를 복사하기 전에 원본 AMI의 내용이 다른 리전에서 실행이 가능하도록 업데이트되었는지 확인해야 합니다. 예를 들어 데이터베이스 연결 문자열 등의 애플리케이션 구성 데이터가 적절한 리소스를 가리키도록 업데이트해야 합니다. 그렇지 않으면 대상 리전의 새 AMI에서 시작된 인스턴스가 여전히 원본 리전의 리소스를 사용하여 성능과 비용에 영향을 줄 수 있습니다.

제한

- 대상 리전은 50개의 동시 AMI 복사본으로 제한됩니다.

교차 계정 복사

AMI를 다른 AWS 계정과 공유할 수 있습니다. AMI 공유는 AMI 소유권에 영향을 미치지 않습니다. 계정 소유에는 리전의 스토리지에 대한 요금이 부과됩니다. 자세한 내용은 [지정한 AWS 계정과 AMI 공유 \(p. 76\)](#) 단원을 참조하십시오.

계정과 공유된 AMI를 복사하는 경우 계정에 있는 대상 AMI의 소유자가 됩니다. 원본 AMI 소유자에게는 표준 Amazon EBS 또는 Amazon S3 전송 요금이 청구되고 사용자에게는 대상 리전의 대상 AMI 스토리지에 대한 요금이 부과됩니다.

리소스 권한

사용자가 다른 계정에서 공유한 AMI를 복사하려면 원본 AMI 소유자는 AMI를 연결된 EBS 스냅샷이든 (Amazon EBS 지원 AMI의 경우) 연결된 S3 버킷이든(인스턴스 스토어 지원 AMI의 경우) 든 지원하는 스토리지에 대한 읽기 권한을 사용자에게 부여해야 합니다. 공유 AMI에 암호화된 스냅샷이 있는 경우, 소유자는 해당 키를 사용자와도 공유해야 합니다.

암호화 및 복사

다음 표는 다양한 AMI 복사 시나리오에 대한 암호화 지원을 보여 줍니다. 암호화되지 않은 스냅샷을 복사하여 암호화된 스냅샷을 생성할 수 있지만, 암호화된 스냅샷을 복사하여 암호화되지 않은 스냅샷을 생성할 수는 없습니다.

시나리오	설명	지원
1	암호화되지 않음-암호화되지 않음	예
2	암호화됨-암호화됨	예
3	암호화되지 않음-암호화됨	예
4	암호화됨-암호화되지 않음	아니요

Note

`CopyImage` 작업 중 암호화는 Amazon EBS 지원 AMIs에만 적용됩니다. 인스턴스 스토어 지원 AMI에서는 스냅샷을 사용하지 않기 때문에 AMI 사본을 사용하여 암호화 상태를 변경할 수 없습니다.

기본적으로(즉, 암호화 파라미터를 지정하지 않은 상태에서) AMI의 스냅샷에 대한 지원은 원래의 암호화 상태와 함께 복사됩니다. 암호화되지 않은 스냅샷에서 지원되는 AMI를 복사하면 역시 암호화되지 않은 동일한 대상 스냅샷이 생성됩니다. 원본 AMI가 암호화된 스냅샷에서 지원되는 경우 원본을 복사하면 동일한 고객 마스터 키(CMK)로 암호화된 동일한 대상 스냅샷이 생성됩니다. 여러 스냅샷에서 지원되는 AMI를 복사하면 각 대상 스냅샷에서 원본 암호화 상태가 유지됩니다.

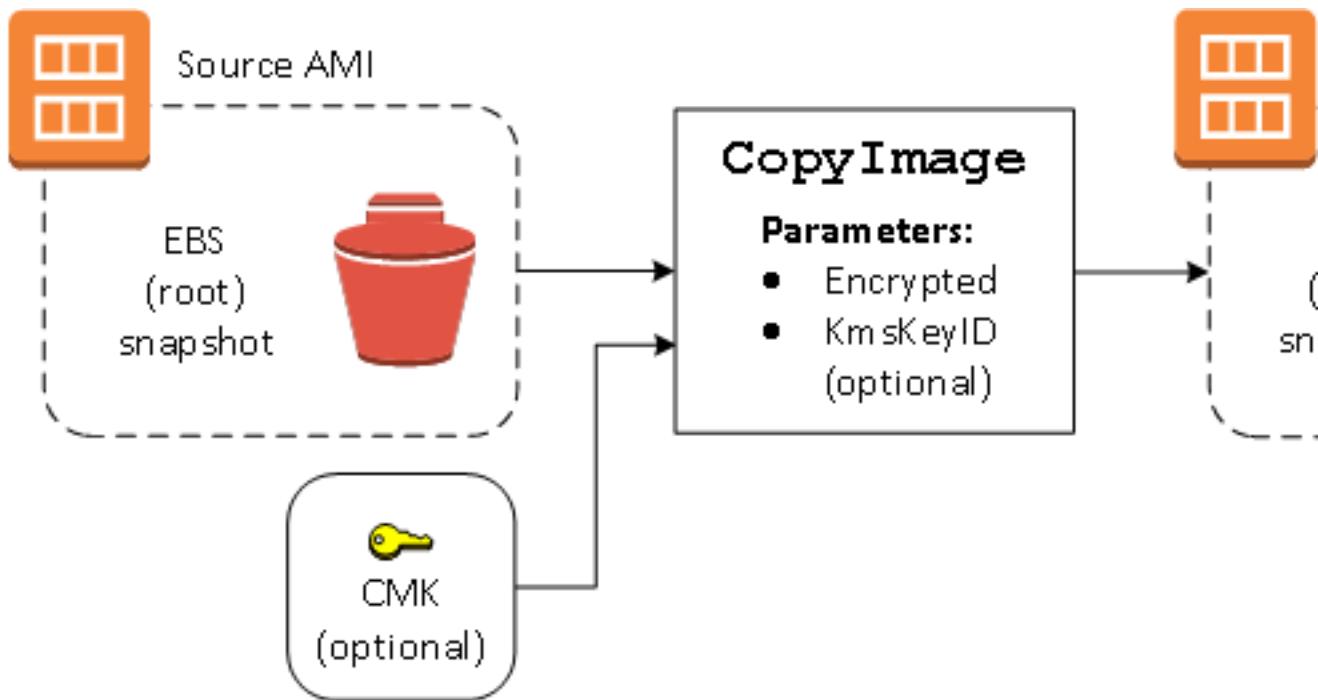
AMI을(를) 복사하는 동안 암호화 파라미터를 지정하면 해당 지원 스냅샷을 암호화 또는 재암호화 할 수 있습니다. 다음 예제에서는 대상 AMI의 암호화 상태를 변경하기 위해 암호화 파라미터를 `CopyImage` 작업으로 공급하는 기본이 아닌 사례를 보여 줍니다.

암호화되지 않은 원본 AMI을(를) 암호화된 대상 AMI(으)로 복사

이 시나리오에서는 암호화되지 않은 루트 스냅샷에서 지원하는 AMI가 암호화된 루트 스냅샷이 있는 AMI로 복사됩니다. `CopyImage` 작업은 CMK를 포함하여 두 암호화 파라미터를 사용하여 호출됩니다. 따라서 루트 스냅샷의 암호화 상태가 변경되므로 대상 AMI는 원본 스냅샷과 동일한 데이터를 포함하는 루트 스냅샷에 의해 지원되지만 지정된 키를 사용하여 암호화됩니다. 두 AMIs 모두의 스냅샷에 대한 스토리지 비용과 각 AMI에서 시작되는 인스턴스에 대한 비용이 발생합니다.

Note

[encryption by default\(암호화 기본 제공\)](#) (p. 1036)를 활성화하는 것은 AMI에서 모든 스냅샷에 대해 `Encrypted` 파라미터를 `true`로 설정하는 것과 동일한 효과를 가집니다.



Encrypted 파라미터를 설정하면 이 인스턴스에 대한 단일 스냅샷이 암호화됩니다. KmsKeyId 파라미터를 지정하지 않으면 스냅샷 복사본을 암호화하는 데 기본 CMK가 사용됩니다.

암호화된 스냅샷을 포함하는 AMIs 복사에 대한 자세한 내용은 [EBS 지원 AMI를 통한 암호화 사용 \(p. 98\)](#) 단원을 참조하십시오.

AMI 복사

다음과 같이 AMI를 복사할 수 있습니다.

사전 조건

Amazon EBS 스냅샷에서 지원되는 AMI를 생성하거나 가져옵니다. Amazon EC2 콘솔을 사용하여 AWS가 제공하는 광범위한 AMI를 검색할 수 있다는 점을 유의하십시오. 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 및 [Windows AMI 찾기 \(p. 68\)](#) 단원을 참조하십시오.

콘솔을 사용하여 AMI를 복사하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 콘솔 탐색 모음에서 AMI가 들어 있는 리전을 선택합니다. 탐색 창에서 이미지, AMI를 선택하여 리전에서 사용할 수 있는 AMIs 목록을 표시합니다.
3. 복사할 AMI를 선택하고 작업, AMI 복사를 선택합니다.
4. AMI 복사 대화 상자에서 다음 정보를 지정하고 AMI 복사를 지정합니다.
 - 대상 리전: AMI를 복사할 리전.
 - 이름: 새 AMI의 이름. AMI에 대한 세부 정보를 표시할 때 운영 체제 정보가 제공되지 않으므로, 이름에 운영 체제 정보를 넣을 수 있습니다.
 - 설명: 원본과 사본을 구분할 수 있도록 설명에는 기본적으로 원본 AMI에 대한 정보가 포함됩니다. 필요에 따라 이 설명을 수정할 수 있습니다.
 - Encryption: 대상 스냅샷을 암호화하거나 다른 키를 사용하여 다시 암호화하려면 이 필드를 선택합니다. [암호화 기본 제공](#)을 활성화한 경우 암호화 옵션이 설정되어 AMI 콘솔로부터 설정 해제할 수 없습니다.

- Master Key: 대상 스냅샷을 암호화하기 위해 사용하는 KMS 키.
5. 복사 작업이 시작되었음을 알리는 확인 페이지가 표시되고 새 AMI의 ID가 제공됩니다.
- 복사 작업의 진행 상황을 즉시 확인하려면 제공된 링크를 따라갑니다. 진행 상황을 나중에 확인하려면 완료를 선택한 후, 준비가 되면 탐색 모음을 사용하여 대상 리전으로 전환하고(해당하는 경우) AMI 목록에서 해당 AMI를 찾습니다.
- 대상 AMI의 초기 상태는 `pending`이고 작업이 완료되면 상태가 `available`이 됩니다.

AWS CLI를 사용하여 AMI를 복사하려면

`copy-image` 명령을 사용하여 AMI를 복사할 수 있습니다. 원본 리전과 대상 리전을 모두 지정해야 합니다. `--source-region` 파라미터를 사용하여 원본 리전을 지정합니다. `--region` 파라미터 또는 환경 변수를 사용하여 대상 리전을 지정할 수 있습니다. 자세한 정보는 [AWS 명령줄 인터페이스 구성](#)을 참조하십시오.

복사 중에 대상 스냅샷을 암호화하는 경우, `--encrypted` 및 `--kms-key-id` 파라미터를 추가로 지정해야 합니다.

Windows PowerShell용 도구를 사용하여 AMI를 복사하려면

`Copy-EC2Image` 명령을 사용하여 AMI를 복사할 수 있습니다. 원본 리전과 대상 리전을 모두 지정해야 합니다. `-SourceRegion` 파라미터를 사용하여 원본 리전을 지정합니다. `-Region` 파라미터 또는 `Set-AWSDefaultRegion` 명령을 사용하여 대상 리전을 지정할 수 있습니다. 자세한 정보는 [AWS 리전 지정](#)을 참조하십시오.

복사 중에 대상 스냅샷을 암호화하는 경우, `-Encrypted` 및 `-KmsKeyId` 파라미터를 추가로 지정해야 합니다.

대기 중인 AMI 복사 작업 중지

다음과 같이 대기 중인 AMI 복사를 중지할 수 있습니다.

콘솔을 사용하여 AMI 복사 작업을 중지하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음의 리전 선택기에서 대상 리전을 선택합니다.
3. 탐색 창에서 [AMIs]를 선택합니다.
4. 복사를 중지할 AMI를 선택하고 작업, 등록 해제를 차례대로 선택합니다.
5. 확인 메시지가 표시되면 계속을 선택합니다.

명령줄을 사용하여 AMI 복사 작업을 중지하려면 다음을 수행합니다.

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `deregister-image`(AWS CLI)
- `Unregister-EC2Image`(Windows PowerShell용 AWS 도구)

결제 정보 가져오기

온디맨드 인스턴스 또는 스팟 인스턴스를 시작하거나 예약 인스턴스를 구매하기 전에 Amazon Machine Image(AMI)와 연결된 플랫폼 세부 정보 및 결제 정보를 확인할 수 있습니다. 스팟 인스턴스의 경우 플랫폼

세부 정보를 사용하여 스팟 인스턴스에 대해 AMI가 지원되는지를 확인할 수 있습니다. 예약 인스턴스를 구매할 때 플랫폼에서 AMI의 플랫폼 세부 정보에 매핑할 올바른 값을 선택합니다. 인스턴스를 시작하거나 예약 인스턴스를 구매하기 전에 결제 정보를 알면 잘못된 AMI에서 인스턴스를 잘못 시작하고 계획되지 않은 비용이 발생할 가능성을 줄일 수 있습니다.

인스턴스 요금에 대한 자세한 내용은 [Amazon EC2 요금](#)을 참조하십시오.

목차

- [AMI 결제 정보 필드 \(p. 109\)](#)
- [플랫폼 세부 정보 및 사용 작업 값 \(p. 109\)](#)
- [플랫폼 세부 정보 및 사용 작업 값 보기 \(p. 110\)](#)
- [청구서에서 결제 정보 확인 \(p. 111\)](#)

AMI 결제 정보 필드

다음 필드는 AMI와 연결된 결제 정보를 제공합니다.

플랫폼 세부 정보

AMI의 결제 코드와 연결된 플랫폼 세부 정보입니다. 예: Red Hat Enterprise Linux.

사용 작업

AMI와 연결된 Amazon EC2 인스턴스 및 결제 코드의 작업입니다. 예: RunInstances:0010. 사용 작업은 AWS 비용 및 사용 보고서(CUR) 및 [AWS 가격 목록 API](#)의 `lineitem/Operation` 열에 해당합니다. 사용 작업 코드 목록은 다음 섹션의 [플랫폼 세부 정보 및 사용 작업 값 \(p. 109\)](#) 단원을 참조하십시오.

Amazon EC2 콘솔의 인스턴스 또는 AMI 페이지에 또는 [describe-images](#) 명령에 의해 반환된 응답으로 이러한 필드를 볼 수 있습니다.

플랫폼 세부 정보 및 사용 작업 값

다음 표에는 Amazon EC2 콘솔의 인스턴스 또는 AMI 페이지에 또는 [describe-images](#) 명령에 의해 반환된 응답으로 표시될 수 있는 플랫폼 세부 정보 및 사용 작업 값이 나와 있습니다.

플랫폼 세부 정보	사용 작업**
Linux/UNIX	RunInstances
Red Hat BYOL Linux	RunInstances:00g0
Red Hat Enterprise Linux	RunInstances:0010
SQL Server Enterprise	RunInstances:0100
SQL Server Standard	RunInstances:0004
SQL Server Web	RunInstances:0200
SUSE Linux	RunInstances:000g
Windows	RunInstances:0002
Windows BYOL	RunInstances:0800
SQL Server Enterprise가 설치된 Windows *	RunInstances:0102

플랫폼 세부 정보	사용 작업**
SQL Server Standard가 설치된 Windows *	RunInstances:0006
SQL Server Web이 설치된 Windows *	RunInstances:0202

* AMI에 두 개의 소프트웨어 라이선스가 연결된 경우 플랫폼 세부 정보 필드에 모두 표시됩니다.

** 스팟 인스턴스를 실행 중인 경우 AWS 비용 및 사용 보고서의 [lineitem/Operation](#)은 여기 나열된 사용 작업 값과 다를 수 있습니다. 예를 들어, [lineitem/Operation](#)에 RunInstances:0010:SV006이 표시되는 경우 Amazon EC2가 미국 동부(버지니아)의 VPC 영역 #6에서 Red Hat Enterprise Linux 스팟 인스턴스 시간을 실행하고 있는 것입니다.

플랫폼 세부 정보 및 사용 작업 값 보기

AMI 또는 인스턴스에서 AMI와 연결된 플랫폼 세부 정보 및 사용 작업 값을 볼 수 있습니다. Amazon EC2 콘솔에서 또는 AWS CLI를 사용하여 이러한 값을 볼 수 있습니다.

AMI에서

AMI(콘솔)와 연결된 플랫폼 세부 정보 및 사용 작업을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 AMI를 선택한 다음 AMI를 선택합니다.
3. 세부 정보 탭에서 플랫폼 세부 정보 및 사용 작업에 대한 값을 확인합니다.

AMI(AWS CLI)와 연결된 플랫폼 세부 정보 및 사용 작업을 보려면

`describe-images` 명령을 사용합니다.

```
$ aws ec2 describe-images --image-ids ami-0123456789EXAMPLE
```

다음 예제 출력은 PlatformDetails 및 UsageOperation 필드를 보여줍니다. 이 예에서 ami-0123456789EXAMPLE 플랫폼은 Red Hat Enterprise Linux이며, 사용 작업 및 결제 코드는 RunInstances:0010입니다.

```
{
  "Images": [
    {
      "VirtualizationType": "hvm",
      "Description": "Provided by Red Hat, Inc.",
      "Hypervisor": "xen",
      "EnaSupport": true,
      "SriovNetSupport": "simple",
      "ImageId": "ami-0123456789EXAMPLE",
      "State": "available",
      "BlockDeviceMappings": [
        {
          "DeviceName": "/dev/sda1",
          "Ebs": {
            "SnapshotId": "snap-111222333444aaabb",
            "DeleteOnTermination": true,
            "VolumeType": "gp2",
            "VolumeSize": 10,
            "Encrypted": false
          }
        }
      ]
    }
  ]
}
```

```
        },
    ],
    "Architecture": "x86_64",
    "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2",
    "RootDeviceType": "ebs",
    "OwnerId": "123456789012",
    "PlatformDetails": "Red Hat Enterprise Linux",
    "UsageOperation": "RunInstances:0010",
    "RootDeviceName": "/dev/sda1",
    "CreationDate": "2019-05-10T13:17:12.000Z",
    "Public": true,
    "ImageType": "machine",
    "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
}
]
```

인스턴스에서

AMI(콘솔)와 연결된 플랫폼 세부 정보 및 사용 작업을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택한 다음 인스턴스를 선택합니다.
3. 세부 정보 탭에서 플랫폼 세부 정보 및 사용 작업에 대한 값을 확인합니다.

AMI(콘솔)와 연결된 플랫폼 세부 정보 및 사용 작업을 보려면

인스턴스를 시작한 후에는 인스턴스 메타데이터의 `billingProducts` 필드를 검사하여 결제 정보를 찾을 수 있습니다. 자세한 정보는 [인스턴스 자격 증명 문서 \(p. 597\)](#) 단원을 참조하십시오. 또는 `describe-instances` 명령을 사용하여 인스턴스에 대한 AMI ID를 가져온 다음 위의 절차에서 설명한 대로 `describe-images` 명령을 사용하여 응답의 `PlatformDetails` 및 `UsageOperation` 필드에서 결제 정보를 가져올 수 있습니다.

청구서에서 결제 정보 확인

계획되지 않은 비용이 발생하지 않도록 AWS 비용 및 사용 보고서(CUR)의 인스턴스에 대한 결제 정보가 인스턴스 시작에 사용한 AMI와 연결된 결제 정보와 일치하는지 확인할 수 있습니다. 결제 정보를 확인하려면 CUR에서 인스턴스 ID를 찾아 `lineitem/Operation` 열에서 해당 값을 확인합니다. 이 값은 AMI와 연결된 사용 작업 값과 일치해야 합니다.

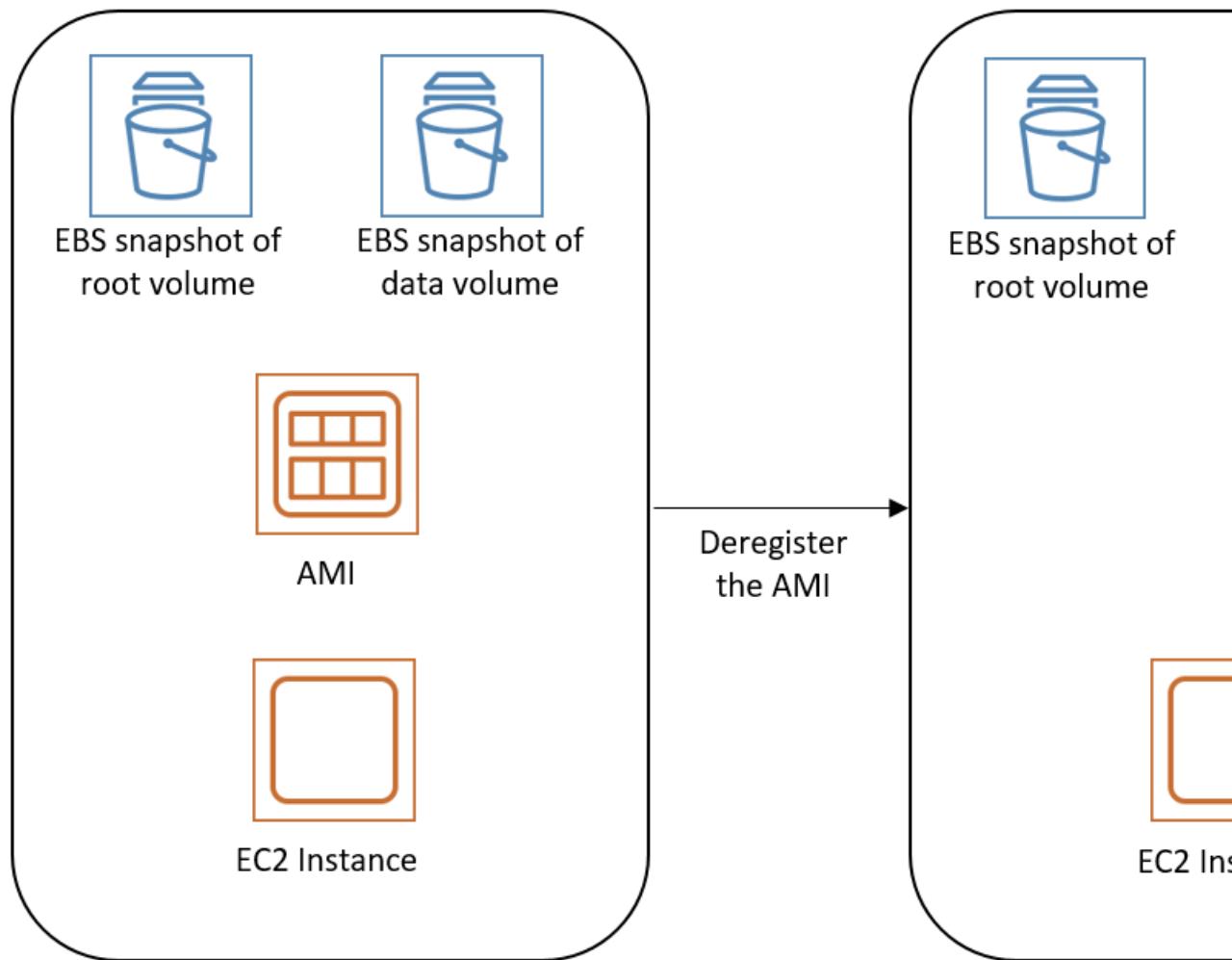
예를 들어 AMI `ami-0123456789EXAMPLE`에는 플랫폼 세부 정보 = `Red Hat Enterprise Linux` 및 사용 작업 = `RunInstances:0010`인 결제 정보가 있습니다. 이 AMI를 사용하여 인스턴스를 시작한 경우 CUR에서 인스턴스 ID를 찾아 `lineitem/Operation` 열에서 해당 값을 확인할 수 있습니다. 이 예제에서 값은 `RunInstances:0010`이어야 합니다.

Windows AMI 등록 취소

사용을 마쳤으면 Windows AMI의 등록을 취소할 수 있습니다. AMI의 등록을 취소한 이후에는 새 인스턴스를 시작하기 위해 해당 AMI를 사용하는 것을 불가능합니다.

AMI의 등록을 취소하는 경우 AMI에서 이미 시작한 인스턴스 또는 AMI 생성 프로세스 중에 생성된 스냅샷에는 영향을 주지 않습니다. 이러한 인스턴스에 대한 사용 비용과 스냅샷에 대한 스토리지 비용이 계속 발생합니다. 따라서 완료된 인스턴스를 종료하고 완료된 스냅샷을 삭제해야 합니다.

다음 다이어그램에서는 Windows AMI를 정리하는 프로세스를 보여줍니다.



Your AMI, its snapshots, and an instance launched from the AMI

Windows AMI를 정리하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 AMI를 선택합니다. AMI를 선택하고 그 ID를 메모해 놓습니다. 그러면 다음 단계에서 올바른 스냅샷을 쉽게 찾을 수 있습니다. 작업을 선택한 후 등록 취소를 선택합니다. 확인 메시지가 표시되면 계속을 선택합니다.

Note

콘솔이 목록에서 AMI를 제거하는 데 몇 분 정도 걸릴 수 있습니다. 상태를 새로 고치려면 새로 고침을 선택합니다.

3. 탐색 창에서 스냅샷을 선택한 후 스냅샷을 선택합니다(설명 열에서 AMI ID를 검색). 작업을 선택한 후 스냅샷 삭제를 선택합니다. 확인 메시지가 나타나면 예, 삭제를 선택합니다.

4. (선택 사항) AMI에서 시작한 인스턴스 관련 작업이 완료되면 해당 인스턴스를 종료합니다. 탐색 창에서 인스턴스를 선택합니다. 인스턴스를 선택하고 작업을 선택한 후 인스턴스 상태와 종료를 차례로 선택합니다. 확인 메시지가 나타나면 예, 종료를 선택합니다.

Amazon EC2 인스턴스

Amazon EC2를 처음 사용하는 경우 시작하려면 다음 단원을 참조하십시오.

- [Amazon EC2이란 무엇입니까? \(p. 1\)](#)
- [Amazon EC2 설정 \(p. 12\)](#)
- [자습서: Amazon EC2 Windows 인스턴스 시작하기 \(p. 16\)](#)
- [인스턴스 수명 주기 \(p. 371\)](#)

프로덕션 환경을 시작하기 전에 다음 질문에 답해야 합니다.

Q. 어떤 인스턴스 유형이 필요에 가장 잘 맞습니까?

Amazon EC2는 애플리케이션을 실행하는 데 필요한 CPU, 메모리, 스토리지 및 네트워킹 용량을 선택할 수 있는 다양한 인스턴스 유형을 제공합니다. 자세한 내용은 [인스턴스 유형 \(p. 114\)](#) 단원을 참조하십시오.

Q. 어떤 구매 옵션이 필요에 가장 잘 맞습니까?

Amazon EC2는 온디맨드 인스턴스(기본값), 스팟 인스턴스 및 예약 인스턴스를 지원합니다. 자세한 내용은 [인스턴스 구입 옵션 \(p. 200\)](#) 단원을 참조하십시오.

Q. 하이브리드 환경에서 EC2 인스턴스 및 머신 플릿을 원격으로 관리할 수 있습니까?

AWS 시스템 관리자를 통해 하이브리드 환경의 Amazon EC2 인스턴스, 온-프레미스 인스턴스 및 다른 클라우드 공급자가 제공하는 VM(가상 머신)을 포함하는 VM의 구성을 안전하게 원격으로 관리할 수 있습니다. 자세한 내용은 [AWS 시스템 관리자 사용 설명서](#)를 참조하십시오.

인스턴스 유형

인스턴스를 시작할 때 지정하는 인스턴스 유형에 따라 인스턴스에 사용되는 호스트 컴퓨터의 하드웨어가 결정됩니다. 각 인스턴스 유형은 서로 다른 컴퓨팅, 메모리, 스토리지 용량을 제공하는데, 이 용량에 따라 서로 다른 인스턴스 패밀리로 분류됩니다. 인스턴스에서 실행하려는 애플리케이션 또는 소프트웨어의 요구 사항에 따라 인스턴스 유형을 선택하십시오.

Amazon EC2에서는 실제로 사용되는 하드웨어에 관계없이 각 인스턴스에 일정하고 예측 가능한 CPU 용량을 제공합니다.

Amazon EC2는 호스트 컴퓨터에 있는 CPU, 메모리 및 인스턴스 스토리지 등의 일부 리소스를 특정 인스턴스에 전용으로 할당합니다. Amazon EC2는 호스트 컴퓨터의 네트워크 및 디스크 하위 시스템과 같은 기타 리소스를 여러 인스턴스와 공유합니다. 호스트 컴퓨터의 각 인스턴스가 이러한 공유 리소스 중 하나를 최대한 많이 사용하려고 할 경우 해당 리소스는 각 인스턴스에 고르게 분배됩니다. 그러나 리소스 사용률이 저조한 경우에는 리소스에 여유가 있는 한 특정 인스턴스가 해당 리소스를 더 많이 소비할 수 있습니다.

각 인스턴스 유형은 공유 리소스의 최소 성능을 더 많이 제공하거나 더 적게 제공합니다. 예를 들어 I/O 성능이 높은 인스턴스 유형에는 더 많은 뷰의 공유 리소스가 할당됩니다. 더 많은 뷰의 공유 리소스가 할당되면 I/O 성능의 변동성도 감소합니다. 대부분의 애플리케이션에 대해서는 중간 수준의 I/O 성능만으로 충분합니다. 그러나 더욱 높거나 일관적인 I/O 성능이 필요한 애플리케이션에 대해서는 I/O 성능이 높은 인스턴스 유형을 사용하는 것이 좋습니다.

내용

- [사용 가능한 인스턴스 유형 \(p. 115\)](#)
- [하드웨어 사양 \(p. 118\)](#)
- [Nitro 시스템에 구축된 인스턴스 \(p. 118\)](#)
- [네트워킹 및 스토리지 기능 \(p. 119\)](#)

- [인스턴스 제한 \(p. 122\)](#)
- [범용 인스턴스 \(p. 122\)](#)
- [컴퓨팅 최적화 인스턴스 \(p. 163\)](#)
- [메모리 최적화 인스턴스 \(p. 168\)](#)
- [스토리지 최적화 인스턴스 \(p. 177\)](#)
- [Windows 액셀러레이티드 컴퓨팅 인스턴스 \(p. 182\)](#)
- [Amazon EC2 인스턴스 유형 찾기 \(p. 193\)](#)
- [인스턴스 유형 변경 \(p. 194\)](#)
- [인스턴스 유형에 대한 권장 사항 가져오기 \(p. 198\)](#)

사용 가능한 인스턴스 유형

Amazon EC2는 각 사용 사례에 최적화된 다양한 인스턴스 유형을 제공합니다. 지원되는 리전, 컴퓨팅 리소스, 스토리지 리소스 등의 요구 사항을 충족하는 인스턴스 유형을 확인하려면 [Amazon EC2 인스턴스 유형 찾기 \(p. 193\)](#) 단원을 참조하십시오.

현재 세대 인스턴스

최상의 성능을 위해서는 새 인스턴스를 시작할 때 다음의 인스턴스 유형을 사용하는 것이 좋습니다. 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하세요.

Type	Sizes	사용 사례
C4	c4.large c4.xlarge c4.2xlarge c4.4xlarge c4.8xlarge	컴퓨팅 최적화 (p. 163)
C5	c5.large c5.xlarge c5.2xlarge c5.4xlarge c5.9xlarge c5.12xlarge c5.18xlarge c5.24xlarge c5.metal	컴퓨팅 최적화 (p. 163)
C5a	c5a.large c5a.xlarge c5a.2xlarge c5a.4xlarge c5a.8xlarge c5a.12xlarge c5a.16xlarge c5a.24xlarge	컴퓨팅 최적화 (p. 163)
C5ad	c5ad.large c5ad.xlarge c5ad.2xlarge c5ad.4xlarge c5ad.8xlarge c5ad.12xlarge c5ad.16xlarge c5ad.24xlarge	컴퓨팅 최적화 (p. 163)
C5d	c5d.large c5d.xlarge c5d.2xlarge c5d.4xlarge c5d.9xlarge c5d.12xlarge c5d.18xlarge c5d.24xlarge c5d.metal	컴퓨팅 최적화 (p. 163)
C5n	c5n.large c5n.xlarge c5n.2xlarge c5n.4xlarge c5n.9xlarge c5n.18xlarge c5n.metal	컴퓨팅 최적화 (p. 163)
D2	d2.xlarge d2.2xlarge d2.4xlarge d2.8xlarge	스토리지 최적화 (p. 177)
SharePoint 용 F1	f1.2xlarge f1.4xlarge f1.16xlarge	액셀러레이티드 컴퓨팅 (p. 182)

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
사용 가능한 인스턴스 유형

Type	Sizes	사용 사례
G3	g3s.xlarge g3.4xlarge g3.8xlarge g3.16xlarge	액셀러레이티드 컴퓨팅 (p. 182)
G4	g4dn.xlarge g4dn.2xlarge g4dn.4xlarge g4dn.8xlarge g4dn.12xlarge g4dn.16xlarge g4dn.metal	액셀러레이티드 컴퓨팅 (p. 182)
H1	h1.2xlarge h1.4xlarge h1.8xlarge h1.16xlarge	스토리지 최적화 (p. 177)
I3	i3.large i3.xlarge i3.2xlarge i3.4xlarge i3.8xlarge i3.16xlarge i3.metal	스토리지 최적화 (p. 177)
I3en	i3en.large i3en.xlarge i3en.2xlarge i3en.3xlarge i3en.6xlarge i3en.12xlarge i3en.24xlarge i3en.metal	스토리지 최적화 (p. 177)
M4	m4.large m4.xlarge m4.2xlarge m4.4xlarge m4.10xlarge m4.16xlarge	범용 (p. 122)
M5	m5.large m5.xlarge m5.2xlarge m5.4xlarge m5.8xlarge m5.12xlarge m5.16xlarge m5.24xlarge m5.metal	범용 (p. 122)
M5a	m5a.large m5a.xlarge m5a.2xlarge m5a.4xlarge m5a.8xlarge m5a.12xlarge m5a.16xlarge m5a.24xlarge	범용 (p. 122)
M5ad	m5ad.large m5ad.xlarge m5ad.2xlarge m5ad.4xlarge m5ad.8xlarge m5ad.12xlarge m5ad.16xlarge m5ad.24xlarge	범용 (p. 122)
M5d	m5d.large m5d.xlarge m5d.2xlarge m5d.4xlarge m5d.8xlarge m5d.12xlarge m5d.16xlarge m5d.24xlarge m5d.metal	범용 (p. 122)
M5dn	m5dn.large m5dn.xlarge m5dn.2xlarge m5dn.4xlarge m5dn.8xlarge m5dn.12xlarge m5dn.16xlarge m5dn.24xlarge	범용 (p. 122)
M5n	m5n.large m5n.xlarge m5n.2xlarge m5n.4xlarge m5n.8xlarge m5n.12xlarge m5n.16xlarge m5n.24xlarge	범용 (p. 122)
P2	p2.xlarge p2.8xlarge p2.16xlarge	액셀러레이티드 컴퓨팅 (p. 182)
P3	p3.2xlarge p3.8xlarge p3.16xlarge	액셀러레이티드 컴퓨팅 (p. 182)
P3dn	p3dn.24xlarge	액셀러레이티드 컴퓨팅 (p. 182)
R4	r4.large r4.xlarge r4.2xlarge r4.4xlarge r4.8xlarge r4.16xlarge	메모리 최적화 (p. 168)

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
사용 가능한 인스턴스 유형

Type	Sizes	사용 사례
R5	r5.large r5.xlarge r5.2xlarge r5.4xlarge r5.8xlarge r5.12xlarge r5.16xlarge r5.24xlarge r5.metal	메모리 최적화 (p. 168)
R5a	r5a.large r5a.xlarge r5a.2xlarge r5a.4xlarge r5a.8xlarge r5a.12xlarge r5a.16xlarge r5a.24xlarge	메모리 최적화 (p. 168)
R5ad	r5ad.large r5ad.xlarge r5ad.2xlarge r5ad.4xlarge r5ad.8xlarge r5ad.12xlarge r5ad.16xlarge r5ad.24xlarge	메모리 최적화 (p. 168)
R5d	r5d.large r5d.xlarge r5d.2xlarge r5d.4xlarge r5d.8xlarge r5d.12xlarge r5d.16xlarge r5d.24xlarge r5d.metal	메모리 최적화 (p. 168)
R5dn	r5dn.large r5dn.xlarge r5dn.2xlarge r5dn.4xlarge r5dn.8xlarge r5dn.12xlarge r5dn.16xlarge r5dn.24xlarge	메모리 최적화 (p. 168)
R5n	r5n.large r5n.xlarge r5n.2xlarge r5n.4xlarge r5n.8xlarge r5n.12xlarge r5n.16xlarge r5n.24xlarge	메모리 최적화 (p. 168)
T2	t2.nano t2.micro t2.small t2.medium t2.large t2.xlarge t2.2xlarge	범용 (p. 122)
T3	t3.nano t3.micro t3.small t3.medium t3.large t3.xlarge t3.2xlarge	범용 (p. 122)
T3a	t3a.nano t3a.micro t3a.small t3a.medium t3a.large t3a.xlarge t3a.2xlarge	범용 (p. 122)
u-xtb1	u-6tb1.metal u-9tb1.metal u-12tb1.metal u-18tb1.metal u-24tb1.metal	메모리 최적화 (p. 168)
X1	x1.16xlarge x1.32xlarge	메모리 최적화 (p. 168)
X1e	x1e.xlarge x1e.2xlarge x1e.4xlarge x1e.8xlarge x1e.16xlarge x1e.32xlarge	메모리 최적화 (p. 168)
z1d	z1d.large z1d.xlarge z1d.2xlarge z1d.3xlarge z1d.6xlarge z1d.12xlarge z1d.metal	메모리 최적화 (p. 168)

이전 세대 인스턴스

Amazon Web Services에서는 이전 세대 인스턴스 유형을 기준으로 애플리케이션을 최적화했으며 아직 업그레이드하지 않은 사용자를 위해 이전 세대 인스턴스 유형을 제공합니다. 최상의 성능을 얻으려면 현재 세대 인스턴스 유형을 사용할 것을 권장합니다. 물론 AWS에서는 다음과 같은 이전 세대 인스턴스 유형을 계속 지원합니다. 어떠한 현재 세대 인스턴스 유형으로 업그레이드하는 것이 적합한지에 대한 자세한 내용은 [이전 세대 인스턴스](#)를 참조하세요.

Type	Sizes
C1	c1.medium c1.xlarge
C3	c3.large c3.xlarge c3.2xlarge c3.4xlarge c3.8xlarge
G2	g2.2xlarge g2.8xlarge
I2	i2.xlarge i2.2xlarge i2.4xlarge i2.8xlarge
M1	m1.small m1.medium m1.large m1.xlarge
M2	m2.xlarge m2.2xlarge m2.4xlarge
M3	m3.medium m3.large m3.xlarge m3.2xlarge
R3	r3.large r3.xlarge r3.2xlarge r3.4xlarge r3.8xlarge
T1	t1.micro

하드웨어 사양

Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

요구 사항에 가장 적합한 인스턴스 유형을 확인하려면 인스턴스를 시작한 후 벤치마크 애플리케이션을 직접 사용해 보는 것이 좋습니다. 과금 기준은 인스턴스 시간이므로 여러 인스턴스 유형을 테스트해 본 후에 결정하는 것이 간편하면서도 경제적입니다.

변경이 필요할 경우 결정을 내린 후에도 인스턴스 크기를 조정할 수 있습니다. 자세한 내용은 [인스턴스 유형 변경 \(p. 194\)](#) 단원을 참조하십시오.

Note

Amazon EC2 인스턴스는 일반적으로 인스턴스 유형 제품 페이지에 지정된 대로 64비트 가상 intel 프로세서에서 실행됩니다. Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오. 하지만 64비트 CPU에 대한 업계의 이름 지정 방식 때문에 혼란이 빚어질 수 있습니다. 칩 제조업체 Advanced Micro Devices(AMD)는 최초로 intel x86 명령 집합 기반의 64비트 아키텍처를 상용화하는 데 성공했습니다. 그 결과, 이 아키텍처는 칩 제조업체와 상관없이 AMD64로 통용됩니다. Windows와 다수의 Linux 배포가 이 관례를 따릅니다. 인스턴스가 intel 하드웨어에서 실행되고 있음에도 불구하고 Ubuntu나 Windows EC2 인스턴스에 대한 내부 시스템 정보는 CPU 아키텍처를 AMD64로 표시하는 이유가 이 때문입니다.

Nitro 시스템에 구축된 인스턴스

Nitro 시스템은 우수한 성능과 고가용성, 철저한 보안을 가능하게 만드는 AWS 구축 하드웨어 및 소프트웨어 구성 요소의 모음입니다. 또한 Nitro 시스템은 가상화 오버헤드를 없애고 호스트 하드웨어에 대한 모든 액세스 권한이 필요한 워크로드를 지원하는 베어 메탈 기능을 제공합니다. 자세한 내용은 [AWS 닉트로 시스템](#)을 참조하십시오.

Nitro 구성 요소

다음 구성 요소는 Nitro 시스템의 일부입니다.

- Nitro 카드
 - 로컬 NVMe 스토리지 볼륨

- 네트워킹 하드웨어 지원
- 관리
- 모니터링
- 보안
- 마더보드에 통합된 Nitro 보안 칩
- Nitro 하이퍼바이저 - 메모리 및 CPU 할당을 관리하고, 대부분의 워크로드를 위한 베어 메탈과 구분이 어려울 정도로 뛰어난 성능을 제공하는 경량 하이퍼바이저입니다.

인스턴스 유형

다음은 Nitro 시스템에 구축된 인스턴스입니다.

- 가상화: C5, C5a, C5ad, C5d, C5n, G4, I3en, M5, M5a, M5ad, M5d, M5dn, M5n, p3dn.24xlarge, R5, R5a, R5ad, R5d, R5dn, R5n, T3, T3a, , z1d
- 베어 메탈: c5.metal, c5d.metal, c5n.metal, i3.metal, i3en.metal, m5.metal, m5d.metal, r5.metal, r5d.metal, u-6tb1.metal, u-9tb1.metal, u-12tb1.metal, u-18tb1.metal, u-24tb1.metal, and z1d.metal

자세히 알아보기

자세한 내용은 다음 비디오를 참조하십시오.

- AWS re:Invent 2017: Amazon EC2 Nitro 시스템 아키텍처
- AWS re:Invent 2017: Amazon EC2 베어 메탈 인스턴스
- AWS re:Invent 2019: 차세대 Amazon EC2 강화: Nitro 시스템 심층 분석
- AWS re:Inforce 2019: Nitro 아키텍처의 보안 이점

네트워킹 및 스토리지 기능

인스턴스 유형을 선택하면 사용할 수 있는 네트워킹 및 스토리지 기능이 결정됩니다. 인스턴스 유형을 설명하려면 [describe-instance-types](#) 명령을 사용합니다.

네트워킹 기능

- IPv6는 모든 현재 세대 인스턴스 유형과 C3, R3 및 I2 이전 세대 인스턴스 유형에서 지원됩니다.
- 인스턴스 유형의 네트워킹 및 대역폭 성능을 극대화하려면 다음을 수행해볼 수 있습니다.
 - 클러스터 배치 그룹에 대해 지원되는 인스턴스 유형을 실행하여 HPC(고성능 컴퓨팅) 애플리케이션에 맞게 인스턴스를 최적화합니다. 공통 클러스터 배치 그룹의 인스턴스는 지역 시간이 짧은 고대역폭 네트워킹이 유용할 수 있습니다. 자세한 내용은 [배치 그룹 \(p. 761\)](#) 단원을 참조하십시오.
 - PPS(Packet Per Second) 성능을 크게 높이고 네트워크 지터 및 지역 시간을 낮추려면 지원되는 최신 인스턴스 유형에 대해 향상된 네트워킹을 활성화합니다. 자세한 내용은 [Windows에서 향상된 네트워킹 \(p. 751\)](#) 단원을 참조하십시오.
- 향상된 네트워킹을 지원하는 현재 세대 인스턴스 유형에는 다음 네트워킹 성능 속성이 있습니다.
 - 동일한 리전 안에서 프라이빗 IPv4 또는 IPv6를 통한 단일 흐름 트래픽은 5Gbps를, 다중 흐름 트래픽은 최대 25Gbps를 지원할 수 있습니다(인스턴스 유형에 따라 다름).
 - 동일한 리전 내에서 퍼블릭 IP 주소 공간이나 VPC 엔드포인트를 통해 Amazon S3 버킷과 주고받는 트래픽은 사용 가능한 인스턴스 집계 대역폭을 전부 사용할 수 있습니다.
- 지원되는 최대 전송 단위(MTU)는 인스턴스 유형에 따라 다릅니다. 모든 Amazon EC2 인스턴스 유형은 표준 이더넷 V2 1500MTU 프레임을 지원합니다. 모든 현재 세대 인스턴스는 9001MTU 또는 점보 프레임을 지원하며, 일부 이전 세대 인스턴스도 이를 지원합니다. 자세한 내용은 [EC2 인스턴스에 대한 네트워크 MTU\(최대 전송 단위\) \(p. 772\)](#) 단원을 참조하십시오.

스토리지 기능

- 일부 인스턴스 유형은 EBS 볼륨과 인스턴스 스토어 볼륨을 지원하는 반면, EBS 볼륨만 지원하는 인스턴스 유형도 있습니다. 인스턴스 스토어 볼륨을 지원하는 일부 인스턴스 유형은 SSD(Solid State Drive)를 사용하여 매우 높은 랜덤 I/O 성능을 제공합니다. 일부 인스턴스 유형은 NVMe 인스턴스 스토어 볼륨을 지원합니다. 일부 인스턴스 유형은 NVMe EBS 볼륨을 지원합니다. 자세한 내용은 [Windows 인스턴스의 Amazon EBS 및 NVMe \(p. 1048\)](#) 및 [NVMe SSD 볼륨 \(p. 1101\)](#) 단원을 참조하십시오.
- 일부 인스턴스 유형은 EBS 최적화 인스턴스로 시작하면 Amazon EBS I/O 전용 용량을 더 많이 확보할 수 있습니다. 일부 인스턴스 유형은 기본적으로 EBS에 최적화되어 있습니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

네트워킹 및 스토리지 기능 요약

다음 표에서는 현재 세대 인스턴스 유형에서 지원되는 네트워킹 및 스토리지 기능을 요약합니다.

	EBS 전용	NVMe EBS	인스턴스 스토어	배치 그룹	향상된 네트워킹
C4	예	아니요	아니요	예	intel 82599 VF
C5	예	예	아니요	예	ENA
C5a	예	예	아니요	예	ENA
C5ad	아니요	예	NVMe *	예	ENA
C5d	아니요	예	NVMe *	예	ENA
C5n	예	예	아니요	예	ENA
D2	아니요	아니요	HDD	예	intel 82599 VF
SharePoint용 F1	아니요	아니요	NVMe *	예	ENA
G3	예	아니요	아니요	예	ENA
G4	아니요	예	NVMe *	예	ENA
HS1	아니요	아니요	HDD *	예	ENA
I3	아니요	아니요	NVMe *	예	ENA
I3en	아니요	예	NVMe *	예	ENA
M4	예	아니요	아니요	예	m4.16xlarge: ENA 기타 모든 크기: intel 82599 VF
M5	예	예	아니요	예	ENA
M5a	예	예	아니요	예	ENA
M5ad	아니요	예	NVMe *	예	ENA
M5d	아니요	예	NVMe *	예	ENA

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
네트워킹 및 스토리지 기능

	EBS 전용	NVMe EBS	인스턴스 스토어	배치 그룹	향상된 네트워킹
M5dn	아니요	예	NVMe *	예	ENA
M5n	예	예	아니요	예	ENA
P2	예	아니요	아니요	예	ENA
P3	예	아니요	아니요	예	ENA
P3dn	아니요	예	NVMe *	예	ENA
R4	예	아니요	아니요	예	ENA
R5	예	예	아니요	예	ENA
R5a	예	예	아니요	예	ENA
R5ad	아니요	예	NVMe *	예	ENA
R5d	아니요	예	NVMe *	예	ENA
R5dn	아니요	예	NVMe *	예	ENA
R5n	예	예	아니요	예	ENA
T2	예	아니요	아니요	아니요	아니요
T3	예	예	아니요	아니요	ENA
T3a	예	예	아니요	아니요	ENA
u-xtb1.metal	예	예	아니요	아니요	ENA
X1	아니요	아니요	SSD *	예	ENA
X1e	아니요	아니요	SSD *	예	ENA
z1d	아니요	예	NVMe *	예	ENA

* 루트 디바이스 볼륨은 Amazon EBS 볼륨이어야 합니다.

다음 표에서는 이전 세대 인스턴스 유형에서 지원되는 네트워킹 및 스토리지 기능을 요약합니다.

	인스턴스 스토어	배치 그룹	향상된 네트워킹
C3	SSD	예	Intel 82599 VF
G2	SSD	예	아니요
I2	SSD	예	intel 82599 VF
M3	SSD	아니요	아니요
R3	SSD	예	intel 82599 VF

인스턴스 제한

한 리전에서 시작할 수 있는 총 인스턴스 수에는 제한이 있으며, 일부 인스턴스 유형에는 또 다른 제한이 있습니다.

기본 제한에 대한 자세한 내용은 [Amazon EC2에서 실행 가능한 인스턴스 수](#)를 참조하십시오.

이러한 제한을 보거나 현재 제한 증가를 요청하는 방법은 [Amazon EC2 서비스 할당량 \(p. 1154\)](#) 단원을 참조하십시오.

범용 인스턴스

컴퓨팅, 메모리 및 네트워킹 리소스의 균형을 유지하는 범용 인스턴스는 광범위한 워크로드에 사용할 수 있습니다.

M5 및 M5a 인스턴스

이러한 인스턴스를 사용하면 클라우드에 배포된 광범위한 애플리케이션을 위해 컴퓨팅, 메모리, 네트워킹 리소스의 균형을 유지해 주는 이상적인 클라우드 인프라를 구축할 수 있습니다. 이러한 인스턴스는 다음의 경우에 적합합니다.

- 중소 규모 데이터베이스
- 추가 메모리가 필요한 데이터 처리 작업
- 캐싱 집합
- SAP, Microsoft SharePoint, 클러스터 컴퓨팅 및 기타 엔터프라이즈 애플리케이션을 위한 백엔드 서버

자세한 내용은 [Amazon EC2 M5 및 M5a 인스턴스](#)를 참조하십시오.

베어 메탈 인스턴스(예: m5.meta1)에서는 애플리케이션이 프로세서, 메모리 등 호스트 서버의 물리적 리소스에 직접 액세스할 수 있습니다. 이 인스턴스는 다음과 같은 경우에 적합합니다.

- 가상화된 환경에서 사용할 수 없거나 완전히 지원되지 않는 저수준 하드웨어 기능(예: intel VT)에 액세스해야 하는 워크로드
- 라이선스 또는 지원을 위해 가상화되지 않은 환경이 필요한 애플리케이션

T2, T3 및 T3a 인스턴스

이러한 인스턴스는 기본 수준의 CPU 성능 외에 버스트 기능이 있어 워크로드에 필요한 만큼 성능을 높일 수 있습니다. 무제한 인스턴스는 필요할 때마다 원하는 기간 동안 높은 CPU 성능을 유지할 수 있습니다. 자세한 내용은 [성능 순간 확장 가능 인스턴스 \(p. 129\)](#) 단원을 참조하십시오. 이 인스턴스는 다음과 같은 경우에 적합합니다.

- 웹 사이트 및 웹 애플리케이션:
- 코드 리포지토리
- 개발, 빌드, 테스트 및 스테이징 환경
- 마이크로서비스

자세한 내용은 [Amazon EC2 T2 인스턴스](#) 및 [Amazon EC2 T3 인스턴스](#)를 참조하십시오.

내용

- [하드웨어 사양 \(p. 123\)](#)
- [인스턴스 성능 \(p. 125\)](#)
- [네트워크 성능 \(p. 125\)](#)
- [SSD I/O 성능 \(p. 127\)](#)

- 인스턴스 기능 (p. 128)
- 릴리스 정보 (p. 129)
- 성능 순간 확장 가능 인스턴스 (p. 129)

하드웨어 사양

다음은 범용 인스턴스용 하드웨어 사양의 요약 설명입니다.

인스턴스 유형	기본 vCPU	메모리(GiB)
m4.large	2	8
m4.xlarge	4	16
m4.2xlarge	8	32
m4.4xlarge	16	64
m4.10xlarge	40	160
m4.16xlarge	64	256
m5.large	2	8
m5.xlarge	4	16
m5.2xlarge	8	32
m5.4xlarge	16	64
m5.8xlarge	32	128
m5.12xlarge	48	192
m5.16xlarge	64	256
m5.24xlarge	96	384
m5.metal	96	384
m5a.large	2	8
m5a.xlarge	4	16
m5a.2xlarge	8	32
m5a.4xlarge	16	64
m5a.8xlarge	32	128
m5a.12xlarge	48	192
m5a.16xlarge	64	256
m5a.24xlarge	96	384
m5ad.large	2	8
m5ad.xlarge	4	16
m5ad.2xlarge	8	32

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
범용

인스턴스 유형	기본 vCPU	메모리(GiB)
m5ad.4xlarge	16	64
m5ad.8xlarge	32	128
m5ad.12xlarge	48	192
m5ad.16xlarge	64	256
m5ad.24xlarge	96	384
m5d.large	2	8
m5d.xlarge	4	16
m5d.2xlarge	8	32
m5d.4xlarge	16	64
m5d.8xlarge	32	128
m5d.12xlarge	48	192
m5d.16xlarge	64	256
m5d.24xlarge	96	384
m5d.metal	96	384
m5dn.large	2	8
m5dn.xlarge	4	16
m5dn.2xlarge	8	32
m5dn.4xlarge	16	64
m5dn.8xlarge	32	128
m5dn.12xlarge	48	192
m5dn.16xlarge	64	256
m5dn.24xlarge	96	384
m5n.large	2	8
m5n.xlarge	4	16
m5n.2xlarge	8	32
m5n.4xlarge	16	64
m5n.8xlarge	32	128
m5n.12xlarge	48	192
m5n.16xlarge	64	256
m5n.24xlarge	96	384
t2.nano	1	0.5

인스턴스 유형	기본 vCPU	메모리(GiB)
t2.micro	1	1
t2.small	1	2
t2.medium	2	4
t2.large	2	8
t2.xlarge	4	16
t2.2xlarge	8	32
t3.nano	2	0.5
t3.micro	2	1
t3.small	2	2
t3.medium	2	4
t3.large	2	8
t3.xlarge	4	16
t3.2xlarge	8	32
t3a.nano	2	0.5
t3a.micro	2	1
t3a.small	2	2
t3a.medium	2	4
t3a.large	2	8
t3a.xlarge	4	16
t3a.2xlarge	8	32

Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

CPU 옵션 지정에 대한 자세한 내용은 [CPU 옵션 최적화 \(p. 541\)](#) 단원을 참조하십시오.

인스턴스 성능

EBS에 최적화된 인스턴스를 사용하면 Amazon EBS I/O와 인스턴스의 다른 네트워크 간의 경합을 제거하여 EBS 볼륨에 대해 일관되게 우수한 성능을 제공할 수 있습니다. 일부 범용 인스턴스는 추가 비용 없이도 EBS에 최적화할 수 있게 기본 설정되어 있습니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

네트워크 성능

지원되는 인스턴스 유형에 대해 향상된 네트워킹을 활성화하면 지연 시간을 줄이고 네트워크 지터를 낮추며 PPS(Packet Per Second) 성능을 높일 수 있습니다. 대부분의 애플리케이션은 항상 높은 수준의 네트워크 성능을 필요로 하지 않지만, 데이터를 주고 받을 때 증가된 대역폭에 액세스할 수 있을 경우 유익할 수 있습니다. 자세한 내용은 [Windows에서 향상된 네트워킹 \(p. 751\)](#) 단원을 참조하십시오.

다음은 확장 네트워킹을 지원하는 범용 인스턴스용 네트워크 성능의 요약 설명입니다.

인스턴스 유형	네트워크 성능	향상된 네트워킹
t2.nano t2.micro t2.small t2.medium t2.large t2.xlarge t2.2xlarge	최대 1Gbps	지원되지 않음
t3.nano t3.micro t3.small t3.medium t3.large t3.xlarge t3.2xlarge t3a.nano t3a.micro t3a.small t3a.medium t3a.large t3a.xlarge t3a.2xlarge	최대 5Gbps †	ENAs (p. 751)
m4.large	보통	Intel 82599 VF (p. 758)
m4.xlarge m4.2xlarge m4.4xlarge	높음	Intel 82599 VF (p. 758)
m5.4xlarge 이하 m5a.8xlarge 이하 m5ad.8xlarge 이하 m5d.4xlarge 이하	최대 10Gbps †	ENAs (p. 751)
m4.10xlarge	10Gbps	Intel 82599 VF (p. 758)
m5.8xlarge m5a.12xlarge m5ad.12xlarge m5d.8xlarge	10Gbps	ENAs (p. 751)
m5.12xlarge m5a.16xlarge m5ad.16xlarge m5d.12xlarge	12Gbps	ENAs (p. 751)
m5.16xlarge m5a.24xlarge m5ad.24xlarge m5d.16xlarge	20Gbps	ENAs (p. 751)
m5dn.4xlarge 이하 m5n.4xlarge 이하	최대 25Gbps †	ENAs (p. 751)
m4.16xlarge m5.24xlarge m5.metal m5d.24xlarge m5d.metal m5dn.8xlarge m5n.8xlarge	25Gbps	ENAs (p. 751)
m5dn.12xlarge m5n.12xlarge	50Gbps	ENAs (p. 751)
m5dn.16xlarge m5n.16xlarge	75Gbps	ENAs (p. 751)
m5dn.24xlarge m5n.24xlarge	100Gbps	ENAs (p. 751)

† 이러한 인스턴스는 네트워크 I/O 크레딧 메커니즘을 이용해 평균 대역폭 활용도를 기준으로 인스턴스에 네트워크 대역폭을 할당합니다. 해당 대역폭이 기준 대역폭 미만으로 떨어지면 크레딧이 발생하는데, 이 크레딧은 네트워크 데이터를 전송할 때 사용할 수 있습니다. 자세한 내용은 지원 사례를 개설하여 관심 있는 특정 인스턴스 유형의 기준 대역폭에 대해 문의하십시오.

SSD I/O 성능

를 사용하고 인스턴스에서 사용 가능한 모든 SSD 기반의 인스턴스 스토어 볼륨을 활용하는 경우, 다음 표와 같은 IOPS(블록 크기 4,096바이트) 성능을 얻을 수 있습니다(대기열 깊이 포화 상태에서). 그렇지 않으면 IOPS 성능이 더 낮아집니다.

인스턴스 크기	100% 임의 읽기 IOPS	IOPS 쓰기
m5ad.large *	30,000개	15,000
m5ad.xlarge *	59,000	29,000
m5ad.2xlarge *	117,000	57,000
m5ad.4xlarge *	234,000	114,000
m5ad.8xlarge	466,666	233,333
m5ad.12xlarge	700,000	340,000
m5ad.16xlarge	933,333	466,666
m5ad.24xlarge	1,400,000	680,000
m5d.large *	30,000개	15,000
m5d.xlarge *	59,000	29,000
m5d.2xlarge *	117,000	57,000
m5d.4xlarge *	234,000	114,000
m5d.8xlarge	466,666	233,333
m5d.12xlarge	700,000	340,000
m5d.16xlarge	933,333	466,666
m5d.24xlarge	1,400,000	680,000
m5d.metal	1,400,000	680,000
m5dn.large *	30,000개	15,000
m5dn.xlarge *	59,000	29,000
m5dn.2xlarge *	117,000	57,000
m5dn.4xlarge *	234,000	114,000
m5dn.8xlarge	466,666	233,333
m5dn.12xlarge	700,000	340,000
m5dn.16xlarge	933,333	466,666
m5dn.24xlarge	1,400,000	680,000

* 이러한 인스턴스의 경우, 지정된 최대 성능을 얻을 수 있습니다.

인스턴스에 대한 SSD 기반 인스턴스 스토어 볼륨에 데이터가 있는 경우, 달성 가능한 쓰기 IOPS의 수는 감소합니다. 이는 SSD 컨트롤러가 가용 공간을 찾고 기존 데이터를 다시 쓰고 미사용 공간을 삭제하여 다시 쓸 수 있는 공간을 마련하기 위해 추가적인 작업을 해야 하기 때문입니다. 이러한 폐영역 회수 과정은 SSD에 대한 내부 쓰기 작업이 증폭되는 결과를 낳게 되며, 이런 결과는 사용자 쓰기 작업에 대한 SSD 쓰기 작업의 비로 표현됩니다. 이러한 성능 감소는 쓰기 작업이 4096바이트의 배수들 또는 4096바이트 경계에 정렬되지 않은 상태로 수행되는 경우에 더 심해질 수 있습니다. 정렬되지 않은 바이트를 소량으로 쓰기 작업하는 경우, SSD 컨트롤러는 쓰려는 부분의 주변 데이터를 읽고 그 결과도 새 위치에 저장해야 합니다. 이런 패턴으로 인해 쓰기 작업이 크게 증폭되고 지연 시간 증가와 I/O 성능의 급격한 감소를 초래합니다.

SSD 컨트롤러는 여러 전략을 사용해서 쓰기 작업 증폭의 영향을 감쇄할 수 있습니다. 그 중 하나의 전력은 SSD 인스턴스 스토리지에 예약 공간을 마련해서 SSD 컨트롤러가 쓰기 작업에 사용 가능한 공간을 보다 효율적으로 관리할 수 있게 하는 것입니다. 이를 오버-프로비저닝이라고 합니다. 인스턴스에 제공된 SSD 기반 인스턴스 스토어 볼륨은 오버-프로비저닝을 위한 예약 공간을 가지고 있지 않습니다. 쓰기 작업 증폭의 영향 감쇄를 위해 최소한 볼륨의 10%를 파티션 처리되지 않은 상태로 두어서 SSD 컨트롤러가 이를 오버-프로비저닝에 사용할 수 있도록 하는 것이 좋습니다. 그러면 사용할 수 있는 스토리지는 줄어들지만, 디스크를 전체 용량에 가깝게 사용하더라도 성능은 향상됩니다.

TRIM을 지원하는 인스턴스 스토어 볼륨의 경우, TRIM 명령을 사용하여 작성한 데이터가 더 이상 필요하지 않을 때 SSD 컨트롤러에 알릴 수 있습니다. 이를 통해 컨트롤러에 더 많은 여유 공간이 제공되어 쓰기 증폭이 줄어들고 성능이 향상될 수 있습니다. 자세한 내용은 [인스턴스 스토어 볼륨 TRIM 지원 \(p. 1102\)](#) 단원을 참조하십시오.

인스턴스 기능

다음은 범용 인스턴스를 위한 기능의 요약 설명입니다.

	EBS 전용	NVMe EBS	인스턴스 스토어	배치 그룹
M4	예	아니요	아니요	예
M5	예	예	아니요	예
M5a	예	예	아니요	예
M5ad	아니요	예	NVMe *	예
M5d	아니요	예	NVMe *	예
M5dn	아니요	예	NVMe *	예
M5n	예	예	아니요	예
T2	예	아니요	아니요	아니요
T3	예	예	아니요	아니요
T3a	예	예	아니요	아니요

* 루트 디바이스 볼륨은 Amazon EBS 볼륨이어야 합니다.

자세한 내용은 다음 단원을 참조하십시오.

- [Windows 인스턴스의 Amazon EBS 및 NVMe \(p. 1048\)](#)
- [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#)
- [배치 그룹 \(p. 761\)](#)

릴리스 정보

- M5, M5d 및 T3 인스턴스는 1세대(Skylake-SP) 또는 2세대(Cascade Lake)의 3.1 GHz Intel Xeon Platinum 8000 시리즈 프로세서를 사용합니다.
- M5a, M5ad 및 T3a 인스턴스는 2.5GHz AMD EPYC 7000 시리즈 프로세서를 사용하는 것이 특징입니다.
- M4, M5, M5a, M5ad, M5d, t2.large 이상, t3.large 이상, t3a.large 및 그 이상 인스턴스 유형에는 64비트 HVM AMIs가 필요합니다. 이들은 고용량 메모리를 보유하는데, 이 용량을 활용하기 위해서는 64비트 운영 체제가 필요합니다. HVM AMI는 고용량 메모리 인스턴스 유형의 반가상화(PV) AMI보다 우수한 성능을 제공합니다. 또한 확장 네트워킹을 활용하려면 HVM AMI를 사용해야 합니다.
- [Nitro 시스템 \(p. 118\)](#)에 구축된 인스턴스에는 다음과 같은 요구 사항이 있습니다.
 - NVMe 드라이버 ([p. 1048](#))가 설치되어 있어야 합니다.
 - ENA(Elastic Network Adapter) ([p. 751](#)) 드라이버가 설치되어 있어야 합니다.

현재 [AWS Windows AMI \(p. 26\)](#)는 이러한 요구 사항을 충족합니다.

- Nitro 시스템에 구축된 인스턴스는 네트워크 인터페이스, EBS 볼륨 및 NVMe 인스턴스 스토어 볼륨을 포함해 최대 28개의 연결을 지원합니다. 자세한 내용은 [Nitro 시스템 볼륨 제한 \(p. 1105\)](#) 단원을 참조하십시오.
- 베어 메탈 인스턴스를 시작하면 기본 서버가 부팅되는데, 이때 모든 하드웨어 및 펌웨어 구성 요소를 확인합니다. 즉, 인스턴스가 실행 상태가 되어 네트워크를 통해 사용할 수 있게 될 때까지 20분이 걸릴 수 있습니다.
- 베어 메탈 인스턴스에서 EBS 볼륨 또는 보조 네트워크 인터페이스를 연결 또는 분리하려면 PCIe 기본 핫 플러그 지원이 필요합니다.
- 베어 메탈 인스턴스에서는 I/O 포트 기반 직렬 디바이스가 아닌 PCI 기반 직렬 디바이스를 사용합니다. 업스트림 Linux 커널 및 최신 Amazon Linux AMI에서는 이 디바이스를 지원합니다. 베어 메탈 인스턴스도 시스템에서 PCI 기반 직렬 디바이스를 자동으로 사용할 수 있게 해주는 ACPI SPCR 테이블을 제공합니다. 최신 Windows AMI에서는 PCI 기반 직렬 디바이스를 자동으로 사용합니다.
- 한 리전에서 시작할 수 있는 총 인스턴스 수에는 제한이 있으며, 일부 인스턴스 유형에는 또 다른 제한이 있습니다. 자세한 내용은 Amazon EC2 FAQ의 [Amazon EC2에서 실행할 수 있는 인스턴스 수는 몇 개입니까?](#)를 참조하십시오.

성능 순간 확장 가능 인스턴스

성능 순간 확장 가능 인스턴스는 기본 수준의 CPU 성능과 함께 워크로드에서 필요한 만큼 성능을 높이는 버스트 기능을 제공하도록 설계되었습니다. 성능 순간 확장 가능 인스턴스는 광범위한 범용 애플리케이션에 적합합니다. 해당되는 예로 마이크로서비스, 자연 시간이 짧은 대화형 애플리케이션, 중소 규모 데이터베이스, 가상 데스크톱, 개발, 빌드, 스테이지 환경, 코드 리포지토리, 제품 프로토타입을 들 수 있습니다.

성능 순간 확장 가능 인스턴스는 CPU 사용량에 대해 크레딧을 사용하는 유일한 인스턴스 유형입니다. 인스턴스 요금에 대한 정보와 기타 자세한 하드웨어 정보는 [Amazon EC2 요금](#) 및 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

계정이 12개월이 아직 안 된 경우 특정 사용 한도 내에서 무료로 t2.micro 인스턴스(또는 t2.micro를 사용할 수 없는 리전에서는 t3.micro 인스턴스)를 사용할 수 있습니다. 자세한 정보는 [AWS 프리 티어](#) 단원을 참조하십시오.

목차

- [성능 순간 확장 가능 인스턴스에 대한 요구 사항 \(p. 130\)](#)
- [모범 사례 \(p. 130\)](#)
- [성능 순간 확장 가능 인스턴스에 대한 CPU 크레딧 및 기준 사용률 \(p. 130\)](#)
- [성능 순간 확장 가능 인스턴스의 무제한 모드 \(p. 133\)](#)
- [성능 순간 확장 가능 인스턴스의 스탠다드 모드 \(p. 141\)](#)
- [성능 순간 확장 가능 인스턴스의 작업 \(p. 155\)](#)

- [CPU 크레딧 모니터링 \(p. 159\)](#)

성능 순간 확장 가능 인스턴스에 대한 요구 사항

다음은 이러한 인스턴스에 대한 기본 요구 사항입니다.

- 지원되는 인스턴스 패밀리는 T2, T3 및 T3a입니다.
- 이러한 인스턴스는 온디맨드 인스턴스, 예약 인스턴스, 전용 인스턴스 및 스팟 인스턴스로 사용할 수 있지만 예약 인스턴스로는 사용할 수 없습니다. 또한 전용 호스트에서도 지원되지 않습니다. 자세한 내용은 [인스턴스 구입 옵션 \(p. 200\)](#) 단원을 참조하십시오.
- 선택한 인스턴스의 크기가 운영 체제 및 애플리케이션의 최소 메모리 요구 사항을 충족하는지 확인합니다. 그래픽 사용자 인터페이스에서 많은 메모리와 CPU 리소스를 사용하는 운영 체제(예: Windows)에서는 대부분의 경우 인스턴스 크기가 t2.micro 이상이어야 합니다. 시간이 지나면서 워크로드의 메모리 및 CPU 요구 사항이 증가함에 따라 더 큰 규모의 동일한 인스턴스 유형 또는 다른 인스턴스 유형에 맞게 조정할 수 있습니다.
- 추가 요구 사항은 [범용 인스턴스 출시 정보 \(p. 129\)](#)를 참조하십시오.

모범 사례

다음 모범 사례를 따르면 성능 순간 확장 가능 인스턴스의 이점을 최대한 활용할 수 있습니다.

- 권장 AMI 사용 – 필수 드라이버를 제공하는 AMI를 사용합니다. 자세한 내용은 [릴리스 정보 \(p. 129\)](#) 단원을 참조하십시오.
- 인스턴스 복구 켜기 – EC2 인스턴스를 모니터링하고 어떤 이유로든 손상된 경우 이를 자동으로 복구하는 CloudWatch 경보를 생성합니다. 자세한 내용은 [Amazon CloudWatch 경보에 복구 작업 추가 \(p. 691\)](#) 단원을 참조하십시오.

성능 순간 확장 가능 인스턴스에 대한 CPU 크레딧 및 기준 사용률

기존 Amazon EC2 인스턴스 유형은 고정된 CPU 사용률을 제공하는 반면, 성능 순간 확장 가능 인스턴스는 기본 수준의 CPU 사용률을 제공하면서 기본 수준 이상으로 CPU 사용률을 버스트하는 기능을 제공합니다. 기준 사용률과 버스트 기능은 CPU 크레딧에 의해 좌우됩니다.

CPU 크레딧은 1분 동안 전체 CPU 코어의 100% 사용률을 제공합니다. vCPU, 사용률 및 시간의 여러 가지 조합이 CPU 크레딧 하나에 해당할 수도 있습니다. 예를 들어, CPU 크레딧 하나는 2분 동안 50%의 사용률로 실행되는 vCPU 하나 또는 2분 동안 25%의 사용률로 실행되는 vCPU 2개에 해당합니다.

목차

- [CPU 크레딧 획득 \(p. 130\)](#)
- [CPU 크레딧 획득률 \(p. 132\)](#)
- [CPU 크레딧 누적 한도 \(p. 132\)](#)
- [획득한 CPU 크레딧의 수명 \(p. 132\)](#)
- [기준 사용률 \(p. 133\)](#)

CPU 크레딧 획득

각 성능 순간 확장 가능 인스턴스는 인스턴스 크기에 따라 특정 비율의 시간당 CPU 크레딧을 지속적으로 (밀리초 수준의 시간 정밀도로) 획득합니다. 크레딧이 누적되는지 아니면 소비되는지를 결정하는 산정 프로세스도 밀리초 수준의 시간 정밀도로 수행되므로 CPU 크레딧 과소비를 염려할 필요는 없습니다. 즉, 짧은 CPU 버스트는 약간의 CPU 크레딧만을 소비합니다.

성능 순간 확장 가능 인스턴스에서 기본 사용률에 필요한 것보다 더 적은 CPU 리소스를 사용하는 경우(예: 유휴 상태) 사용하지 않은 CPU 크레딧은 CPU 크레딧 밸런스에 누적됩니다. 성능 순간 확장 가능 인스턴스

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
범용

가 기준 사용률 수준 이상으로 버스트해야 할 경우 누적된 크레딧을 소모합니다. 성능 순간 확장 가능 인스턴스가 획득한 크레딧이 많을수록 추가 CPU 사용률이 필요할 때 기준 사용률 수준을 초과하여 버스트할 수 있는 시간이 증가합니다.

다음 표에는 성능 순간 확장 가능 인스턴스 유형, 시간당 CPU 크레딧 획득률, 인스턴스가 획득할 수 있는 최대 CPU 크레딧 수, 인스턴스당 vCPU 수, 전체 코어의 백분율로 나타낸 기준 사용률(단일 vCPU 사용 시) 등이 나와 있습니다.

인스턴스 유형	시간당 지급되는 CPU 크레딧	누적 가능한 최대 지급된 크레딧*	vCPUs	vCPU당 기준 사용률
T2				
t2.nano	3	72	1	5%
t2.micro	6	144	1	10%
t2.small	12	288	1	20%
t2.medium	24	576	2	20%**
t2.large	36	864	2	30%**
t2.xlarge	54	1296	4	22.5%**
t2.2xlarge	81.6	1958.4	8	17%**
T3				
t3.nano	6	144	2	5%**
t3.micro	12	288	2	10%**
t3.small	24	576	2	20%**
t3.medium	24	576	2	20%**
t3.large	36	864	2	30%**
t3.xlarge	96	2304	4	40%**
t3.2xlarge	192	4608	8	40%**
T3a				
t3a.nano	6	144	2	5%**
t3a.micro	12	288	2	10%**
t3a.small	24	576	2	20%**
t3a.medium	24	576	2	20%**
t3a.large	36	864	2	30%**
t3a.xlarge	96	2304	4	40%**
t3a.2xlarge	192	4608	8	40%**

* 누적될 수 있는 크레딧은 수는 24시간 동안 획득할 수 있는 크레딧의 수와 동일합니다.

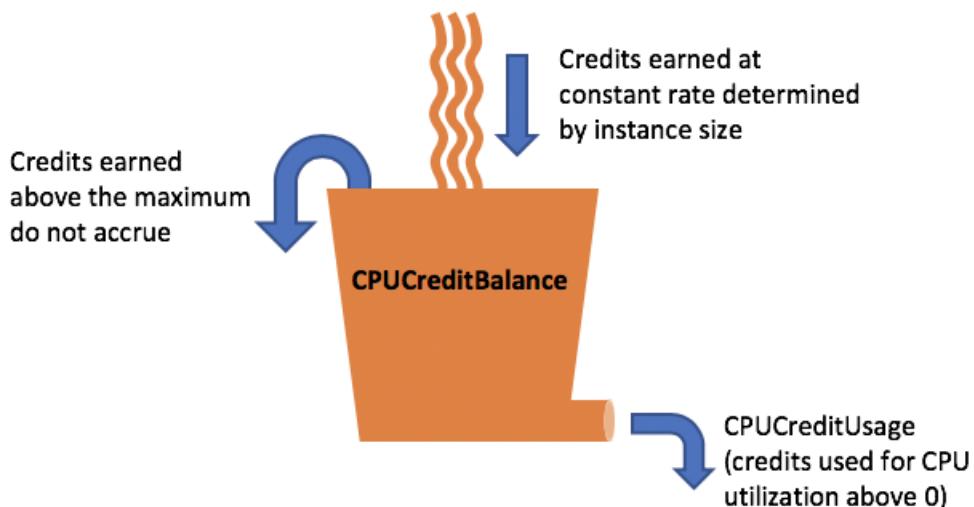
** 테이블의 기준 사용률(%)은 vCPU당입니다. CloudWatch에서 CPU 사용률은 vCPU 기준으로 표시됩니다. 예를 들어 기준 수준으로 작동하는 t3.large 인스턴스에 대한 CPU 사용률은 CloudWatch CPU 지표의 30%로 표시됩니다. 기준 사용률을 계산하는 방법에 대한 자세한 내용은 [기준 사용률 \(p. 133\)](#) 단원을 참조하십시오.

CPU 크레딧 획득률

시간당 획득하는 CPU 크레딧의 수는 인스턴스 크기에 의해 결정됩니다. 예를 들어 t3.nano는 시간당 6개의 크레딧을 획득하는 반면, t3.small은 시간당 24개의 크레딧을 획득합니다. 이전 표에는 모든 인스턴스에 대한 크레딧 획득률이 나와 있습니다.

CPU 크레딧 누적 한도

실행 중인 인스턴스에서 획득한 크레딧은 만료되지 않습니다. 하지만 인스턴스가 누적할 수 있는 획득 크레딧 수에는 한도가 있습니다. 한도는 CPU 크레딧 밸런스 한도에 따라 결정됩니다. 한도에 도달한 후에 새로 획득하는 크레딧은 다음 이미지와 같이 모두 삭제됩니다. 최대 버킷은 CPU 크레딧 밸런스 한도를 나타내고, 스필오버는 한도를 초과하여 새로 획득한 크레딧을 나타냅니다.



CPU 크레딧 밸런스 한도는 각 인스턴스 크기에 따라 다릅니다. 예를 들어 t3.micro 인스턴스는 CPU 크레딧 밸런스에서 최대 288의 획득한 CPU 크레딧을 누적할 수 있습니다. 이전 표에는 각 인스턴스에서 누적할 수 있는 최대 획득 크레딧 수가 나와 있습니다.

T2 스탠다드 인스턴스에서도 시작 크레딧을 획득합니다. 시작 크레딧은 CPU 크레딧 밸런스 한도에 포함되지 않습니다. T2 인스턴스가 시작 크레딧을 사용하지 않고 획득 크레딧을 누적하면서 24시간 동안 유튜 상태를 유지한 경우 CPU 크레딧 밸런스는 한도 이상으로 표시됩니다. 자세한 내용은 [시작 크레딧 \(p. 142\)](#) 단원을 참조하십시오.

T3 이상 인스턴스에서는 시작 크레딧을 획득하지 않습니다. 이러한 인스턴스는 `unlimited`로 시작 하도록 기본 설정되어 있으므로 시작 크레딧 없이도 시작하자마자 즉시 버스트 할 수 있습니다.

획득한 CPU 크레딧의 수명

실행 중인 인스턴스의 CPU 크레딧은 만료 기간이 없습니다.

T2의 경우 CPU 크레딧 밸런스는 인스턴스 종지와 시작 사이의 기간 동안 지속하지 않습니다. T2 인스턴스를 종지하면 인스턴스는 누적된 크레딧을 모두 상실합니다.

T3 이상의 경우 인스턴스가 종지된 후 CPU 크레딧 밸런스가 7일 동안 지속하다가 7일이 지나면 크레딧이 상실됩니다. 7일 이내에 인스턴스를 시작하면 크레딧이 상실되지 않습니다.

자세한 내용은 [CloudWatch 지표 \(p. 160\)](#) 표에서 CPUCreditBalance 항목을 참조하십시오.

기준 사용률

기준 사용률은 획득하는 CPU 크레딧 수가 사용 중인 CPU 크레딧 수와 일치할 때 순 크레딧 밸런스 0에서 CPU를 사용할 수 있는 수준입니다. 기준 사용률을 기준이라고도 합니다.

기준 사용률은 vCPU 사용률의 백분율로 표시되며 다음과 같이 계산됩니다.

$(\text{number of credits earned}/\text{number of vCPUs})/60 \text{ minutes} = \% \text{ baseline utilization}$

예를 들어 vCPU가 2개인 t3.nano 인스턴스는 시간당 6 크레딧을 획득하여 기준 사용률이 5%로, 다음과 같이 계산됩니다.

$(6 \text{ credits earned}/2 \text{ vCPUs})/60 \text{ minutes} = 5\% \text{ baseline utilization}$

vCPU가 4개인 t3.xlarge 인스턴스는 시간당 96 크레딧을 획득하여 기준 사용률이 40%입니다 ($(96/4)/60$)。

성능 순간 확장 가능 인스턴스의 무제한 모드

unlimited로 구성된 성능 순간 확장 가능 인스턴스는 필요한 경우 언제든지 원하는 기간 동안 높은 CPU 사용률을 유지할 수 있습니다. 24시간 동안 또는 인스턴스 수명(더 짧음) 동안 인스턴스의 평균 CPU 사용률이 기준 이하인 경우에 모든 CPU 사용량 급증에 대해 시간당 CPU 인스턴스 요금이 적용됩니다.

대부분의 범용 워크로드에서 unlimited로 구성된 인스턴스는 추가 요금 없이 충분한 성능을 제공합니다. 인스턴스 실행에 장기간 높은 CPU 사용률이 필요한 경우, vCPU-시간당 추가 고정 요금으로 인스턴스를 실행할 수 있습니다. 인스턴스 요금에 대한 자세한 내용은 [Amazon EC2 요금](#) 및 [Amazon EC2 온디맨드 요금 페이지](#)의 무제한 모드 요금에 대한 단원을 참조하세요.

[AWS 프리 티어](#) 혜택에 따라 t2.micro 또는 t3.micro 인스턴스를 사용하고 이 인스턴스를 unlimited 모드에서 사용하는 경우 24시간 룰링 기간 동안 평균 사용률이 인스턴스의 [기준 사용률 \(p. 133\)](#)을 초과하면 요금이 적용될 수 있습니다.

T3 이상 인스턴스는 unlimited로 시작하도록 기본 설정되어 있습니다. 24시간 동안 평균 CPU 사용량이 기준을 초과하면 잉여 크레딧에 대한 요금이 발생합니다. T3 이상 스팟 인스턴스를 unlimited로 시작하고 CPU 크레딧 발생에 대한 유지 시간 없이 즉시 짧은 기간 동안 사용하려는 경우 잉여 크레딧에 대한 요금이 발생합니다. 더 높은 비용을 지불하지 않으려면 [표준 \(p. 141\)](#) 모드에서 T3 이상 스팟 인스턴스를 시작하는 것이 좋습니다. 자세한 내용은 [잉여 크레딧으로 요금 발생 가능 \(p. 136\)](#) 및 [성능 순간 확장 가능 인스턴스 \(p. 320\)](#) 단원을 참조하세요.

목차

- [무제한 모드 개념 \(p. 134\)](#)
 - [무제한 성능 순간 확장 가능 인스턴스의 작동 방식 \(p. 134\)](#)
 - [무제한 모드 대 고정 CPU 사용 시기 \(p. 134\)](#)
 - [잉여 크레딧으로 요금 발생 가능 \(p. 136\)](#)
 - [T2 무제한 인스턴스에는 시작 크레딧이 없음 \(p. 137\)](#)
 - [무제한 모드 활성화 \(p. 137\)](#)
 - [무제한과 스탠다드 간 전환 시 크레딧에 발생하는 현상 \(p. 137\)](#)
 - [크레딧 사용량 모니터링 \(p. 138\)](#)
- [무제한 모드 예제 \(p. 138\)](#)
 - [예 1: T3 무제한에서 크레딧을 사용하는 것에 관한 설명 \(p. 138\)](#)
 - [예 2: T2 무제한에서 크레딧을 사용하는 것에 관한 설명 \(p. 139\)](#)

무제한 모드 개념

unlimited 모드는 성능 순간 확장 가능 인스턴스에 사용할 수 있는 크레딧 구성 옵션입니다. 이 모드는 실행 중인 또는 중지된 인스턴스에 대해 언제든지 활성화 또는 비활성화할 수 있습니다. 성능 순간 확장 가능 인스턴스 패밀리마다 AWS 리전별 계정 수준에서 unlimited를 기본 크레딧 옵션으로 설정하여 계정에 속한 새 성능 순간 확장 가능 인스턴스가 모두 기본 크레딧 옵션을 사용하여 시작되게 할 수 있습니다.

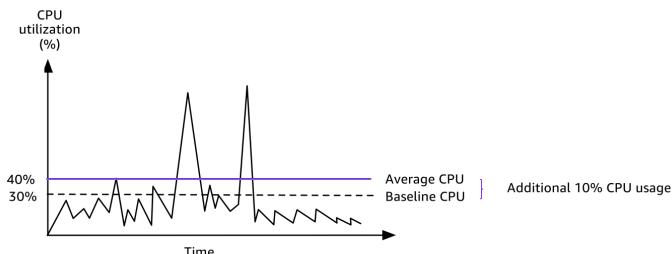
Note

T3 이상 인스턴스는 unlimited로 시작하도록 기본 설정되어 있습니다. T2 인스턴스는 standard로 시작하도록 기본 설정되어 있습니다. AWS 리전별로 계정 수준에서 기본값을 변경할 수 있습니다. 자세한 내용은 [계정의 기본 크레딧 사양 설정 \(p. 159\)](#) 단원을 참조하십시오.

무제한 성능 순간 확장 가능 인스턴스의 작동 방식

unlimited로 구성된 성능 순간 확장 가능 인스턴스의 CPU 크레딧 밸런스가 감소하면 잉여 크레딧을 사용하여 [기준 \(p. 133\)](#) 이상으로 버스트할 수 있습니다. CPU 사용률이 기준 미만으로 떨어지면 획득한 CPU 크레딧을 사용하여 이전에 소비한 잉여 크레딧을 청산할 수 있습니다. CPU 크레딧을 획득하고 잉여 크레딧을 청산하는 기능을 통해 Amazon EC2는 24시간 동안 인스턴스의 CPU 사용률을 평균 수준으로 유지할 수 있습니다. 24시간 동안의 평균 CPU 사용량이 기준을 초과하는 경우 인스턴스에 추가 사용량에 대해 vCPU 시간당 고정 추가 요금이 청구됩니다.

다음 그래프는 t3.large의 CPU 사용량을 보여줍니다. t3.large에 대한 기본 CPU 사용률은 30%입니다. 인스턴스가 24시간 동안 평균 30% CPU 사용률로 실행되는 경우 이미 인스턴스 시간당 가격으로 비용이 처리되었으므로 추가 비용이 발생하지 않습니다. 그러나 그래프에 표시된 것처럼 24시간 동안의 평균 40%의 CPU 사용률로 실행되는 경우 이 인스턴스는 추가 10% CPU 사용량에 대해 vCPU 시간당 고정 요금이 청구됩니다.



각 인스턴스 유형별 vCPU 당 기준 사용률 및 각 인스턴스 유형에서 얻은 크레딧 수에 대한 자세한 내용은 [크레딧 표 \(p. 131\)](#)을 참조하십시오.

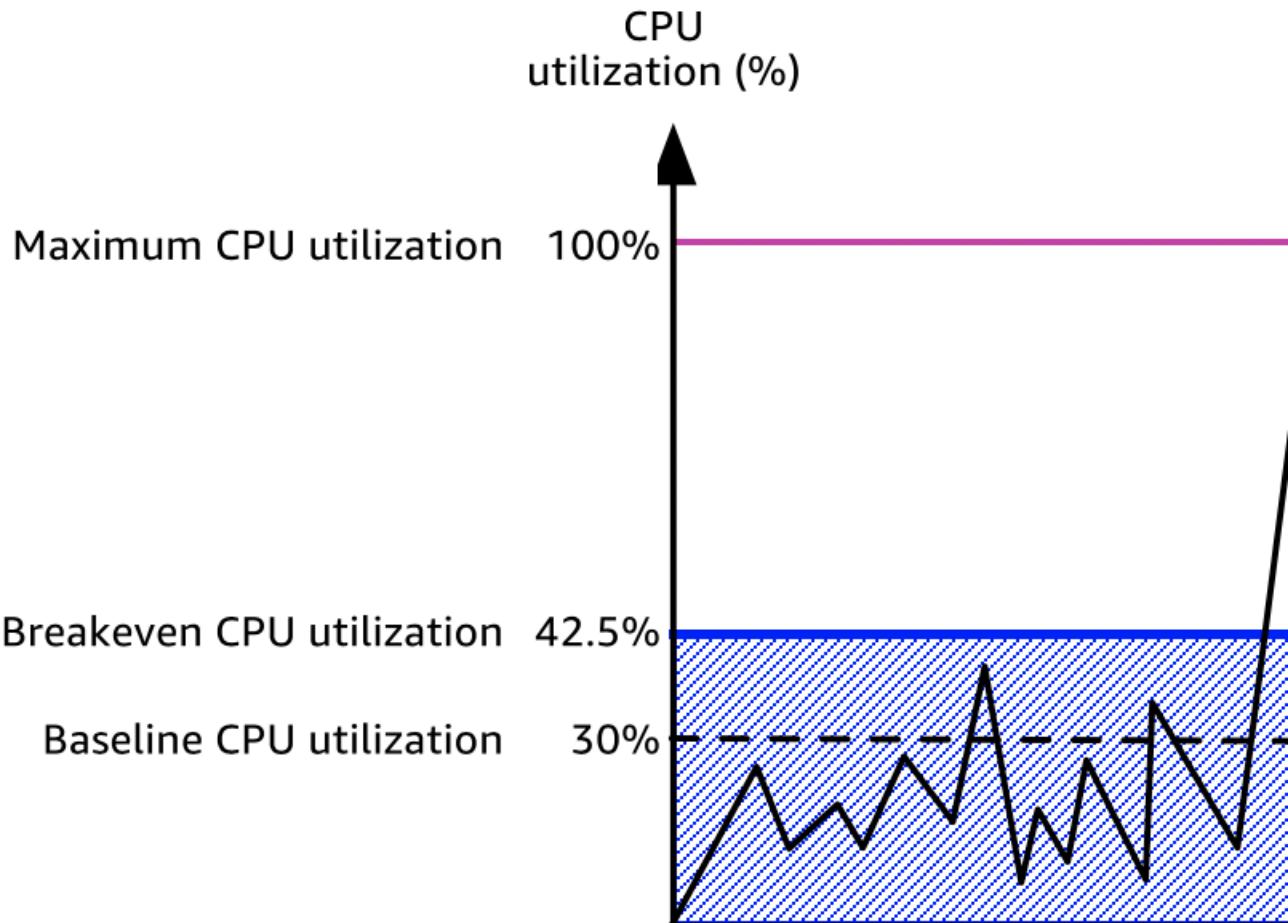
무제한 모드 대 고정 CPU 사용 시기

T3와 같은 unlimited 모드 또는 M5와 같은 고정 성능 인스턴스에서 버스트 가능한 성능 인스턴스를 사용해야 하는지 결정할 때는 손익분기 CPU 사용량을 결정해야 합니다. 버스트 가능한 성능 인스턴스에 대한 손익분기 CPU 사용량은 버스트 가능한 성능 인스턴스가 고정 성능 인스턴스와 동일한 비용을 부담합니다. 손익분기 CPU 사용량은 다음을 결정하는 데 도움이 됩니다.

- 24시간 동안의 평균 CPU 사용량이 손익분기 CPU 사용량 또는 그 이하인 경우 unlimited 모드에서 버스트 가능한 성능 인스턴스를 사용하면 버스트 가능한 성능 인스턴스의 저렴한 가격으로 혜택을 누릴 수 있으며 동시에 고정 성능 인스턴스와 동일한 성능을 얻을 수 있습니다.
- 24시간 동안의 평균 CPU 사용량이 손익분기 CPU 사용량보다 많으면 버스트 가능한 성능 인스턴스의 비용은 동등한 크기의 고정 성능 인스턴스보다 증가합니다. T3 인스턴스가 100% CPU에서 연속적으로 버스트하면 동등한 크기의 M5 인스턴스 가격의 약 1.5배를 지불하게 됩니다.

다음 그래프에서는 t3.large와 m5.large와 동일한 비용의 손익분기 CPU 사용량을 보여줍니다. t3.large에 대한 손익분기 CPU 사용량은 42.5%입니다. 평균 CPU 사용량이 42.5%인 경우 t3.large를 실행하는 비용은 m5.large와 동일하며 평균 CPU 사용량이 42.5%를 초과하면 비용이 더 많이 듭니다.

다. 작업 부하가 42.5% 미만의 평균 CPU 사용량이 필요한 경우 m5.large와 동일한 성능을 얻는 동안 t3.large의 저렴한 가격으로 혜택을 볼 수 있습니다.



다음 표는 손익분기 CPU 사용량 임계값을 계산하여 **unlimited** 모드 또는 고정 성능 인스턴스에서 버스트 가능한 성능 인스턴스를 사용하는 것이 더 경제적인 시기가 언제인지를 결정할 수 있는 방법을 보여줍니다. 테이블의 열은 A에서 K로 표시됩니다.

인스턴스 유형	vCPUs	T3 가격*/시간	M5 가격*/시간	가격 차이	vCPU 당 T3 기준 사용률(%)	잉여 크레딧에 대한 vCPU 시간당 요금	vCPU 분당 요금	vCPU 당 사용 가능한 추가 버스트 시간(분)	사용 가능한 추가 CPU%	손익 분기 CPU%
A	B	C	D	E = D - C	F	G	H = G/60	I = E/H	J = (I/60)/B	K = F + J

인스턴스 유형	vCPUs	T3 가격*/시간	M5 가격*/시간	가격 차이	vCPU 당 T3 기준 사용률(%)	잉여 크레딧에 대한 vCPU 시간당 요금	vCPU 분당 요금	vCPU 당 사용 가능한 추가 버스트 시간(분)	사용 가능 한 추가 CPU%	손익 분기 CPU%
t3.large	2	\$0.0835 USD	\$0.096 USD	\$0.0125 USD	30%	0.05 USD	\$0.000833 USD	15	12.5%	42.5%

* 가격은 us-east-1 및 Linux OS를 기준으로 합니다.

이 테이블에서는 다음 정보를 제공합니다.

- A열은 인스턴스 유형인 t3.large을 표시합니다.
- B열은 t3.large에 대한 vCPU 수를 나타냅니다.
- C열은 시간당 t3.large의 가격을 보여줍니다.
- D열은 시간당 m5.large의 가격을 보여줍니다.
- E열은 t3.large과 m5.large 사이의 가격 차이를 보여줍니다.
- F열은 30%인 t3.large의 vCPU당 기준 사용률을 보여줍니다. 기준선에서 인스턴스의 시간당 비용은 CPU 사용량 비용을 포함합니다.
- G열은 획득된 크레딧이 소진된 후 100% CPU에서 버스트되는 경우 인스턴스에 청구되는 vCPU 시간당 고정 추가 요금을 보여줍니다.
- H열은 획득된 크레딧이 소진된 후 100% CPU에서 버스트되는 경우 인스턴스에 청구되는 vCPU 시간(분)당 고정 추가 요금을 보여줍니다.
- I열은 t3.large이 시간당 100% CPU에서 버스트 가능하고 m5.large와 같은 시간당 가격을 지불하는 추가 시간(분)을 보여줍니다.
- J열은 m5.large로 동일한 가격을 지불하면서 인스턴스가 버스트 가능한 기준선에 대한 추가 CPU 사용량(%)을 보여줍니다.
- K열은 t3.large와 m5.large보다 많은 비용을 들이지 않고 버스트 가능한 손익분기 CPU 사용량(%)을 보여줍니다. t3.large 비용 및 그 어떤 비용도 m5.large보다 많습니다.

다음 테이블은 비슷한 크기의 M5 인스턴스 유형과 비교한 T3 인스턴스 유형의 손익분기 CPU 사용량(%)을 보여줍니다.

T3 인스턴스 유형	M5와 비교한 T3에 대한 손익분기 CPU 사용량(%)
t3.large	42.5%
t3.xlarge	52.5%
t3.2xlarge	52.5%

잉여 크레딧으로 요금 발생 가능

인스턴스의 평균 CPU 사용률이 기준 이하인 경우에는 인스턴스로 인해 추가 요금이 발생하지 않습니다. 인스턴스는 24시간 동안 최대 크레딧 수 (p. 131)를 획득하기 때문에(예를 들면 t3.micro 인스턴스는 24시간 동안 최대 288개의 크레딧 획득이 가능) 요금을 부과하지 않고 이 최대 값까지 잉여 크레딧을 소비할 수 있습니다.

그러나 CPU 사용률이 기준 이상으로 유지되는 경우 인스턴스는 소비한 잉여 크레딧을 청산하기에 충분한 수준으로 크레딧을 획득할 수 없습니다. 청산된 잉여 크레딧은 vCPU-시간당 추가 고정 요금으로 부과됩니다.

이전에 소비된 잉여 크레딧은 다음이 발생할 때 요금이 부과됩니다.

- 소비한 잉여 크레딧이 인스턴스가 24시간 동안 획득할 수 있는 [최대 크레딧 수 \(p. 131\)](#)를 초과하는 경우. 해당 시간이 끝날 때 최대 값 이상으로 소비한 잉여 크레딧에 요금이 부과됩니다.
- 인스턴스가 종지 또는 종료된 경우.
- 인스턴스가 `unlimited`에서 `standard`로 전환됩니다.

소비한 잉여 크레딧은 CloudWatch 지표 `CPUSurplusCreditBalance`에 의해 추적이 가능합니다. 요금이 부과된 잉여 크레딧은 CloudWatch 지표 `CPUSurplusCreditsCharged`에 의해 추적이 가능합니다. 자세한 내용은 [성능 순간 확장 가능 인스턴스에 대한 추가 CloudWatch 측정치 \(p. 159\)](#) 단원을 참조하십시오.

T2 무제한 인스턴스에는 시작 크레딧이 없음

T2 스탠다드 인스턴스에서는 [시작 크레딧 \(p. 142\)](#)을 획득하지만 T2 무제한 인스턴스에서는 시작 크레딧을 획득하지 않습니다. 24시간 동안 또는 인스턴스 수명(더 짧음) 동안 평균 CPU 사용률이 기준 이하인 경우, T2 무제한 인스턴스는 언제라도 추가 요금 없이 기준 성능 이상으로 버스트가 가능합니다. 따라서 T2 무제한 인스턴스는 시작 크레딧 없이도 시작 즉시 높은 성능을 달성할 수 있습니다.

T2 인스턴스가 `standard`에서 `unlimited`으로 전환된 경우 남은 `CPUCreditBalance`가 전달되기 전에 `CPUCreditBalance`에서 누적된 시작 크레딧이 모두 삭제됩니다.

Note

T3 이상 인스턴스는 시작 크레딧을 획득하지 않습니다.

무제한 모드 활성화

T3 이상 인스턴스는 `unlimited`로 시작하도록 기본 설정되어 있습니다. T2 인스턴스는 기본적으로 `standard`로 시작하지만 시작 시 `unlimited`를 활성화할 수 있습니다.

실행 중이거나 종지된 인스턴스에서 언제든지 `unlimited`에서 `standard`로, `standard`에서 `unlimited`로 전환할 수 있습니다. 자세한 내용은 [무제한 또는 스탠다드로 성능 순간 확장 가능 인스턴스 시작 \(p. 155\)](#) 및 [성능 순간 확장 가능 인스턴스의 크레딧 사양 수정 \(p. 158\)](#) 단원을 참조하십시오.

성능 순간 확장 가능 인스턴스 패밀리마다 AWS 리전별 계정 수준에서 `unlimited`를 기본 크레딧 옵션으로 설정하여 계정에 속한 새 성능 순간 확장 가능 인스턴스가 모두 기본 크레딧 옵션을 사용하여 시작되게 할 수 있습니다. 자세한 내용은 [계정의 기본 크레딧 사양 설정 \(p. 159\)](#) 단원을 참조하십시오.

Amazon EC2 콘솔 또는 AWS CLI를 사용하여 확장 가능한 성능 인스턴스가 `unlimited` 또는 `standard`로 구성되었는지 확인할 수 있습니다. 자세한 내용은 [성능 순간 확장 가능 인스턴스의 크레딧 사양 보기 \(p. 157\)](#) 및 [기본 크레딧 사양 보기 \(p. 159\)](#) 단원을 참조하십시오.

무제한과 스탠다드 간 전환 시 크레딧에 발생하는 현상

`CPUCreditBalance`는 인스턴스에서 누적한 크레딧 수를 추적하는 CloudWatch 측정치입니다.

`CPUSurplusCreditBalance`는 인스턴스에서 사용한 잉여 크레딧 수를 추적하는 CloudWatch 측정치입니다.

`unlimited`로 구성된 인스턴스를 `standard`로 변경하면 다음이 발생합니다.

- `CPUCreditBalance` 값은 변경되지 않은 채 전달됩니다.
- `CPUSurplusCreditBalance` 값은 즉시 요금이 부과됩니다.

`standard` 인스턴스가 `unlimited`로 전환될 경우 다음이 발생합니다.

- 누적된 획득 크레딧이 포함된 CPUCreditBalance 값이 전달됩니다.
- T2 스탠다드 인스턴스의 경우 CPUCreditBalance 값에서 모든 시작 크레딧이 삭제되고, 누적된 획득 크레딧이 포함된 나머지 CPUCreditBalance 값이 전달됩니다.

크레딧 사용량 모니터링

인스턴스가 기준 이상의 크레딧을 사용하고 있는지 여부를 확인하기 위해 CloudWatch 측정치를 사용하여 사용량을 추적할 수 있으며 시간별 경보를 설정하여 크레딧 사용량에 대한 알림을 받을 수 있습니다. 자세한 정보는 [CPU 크레딧 모니터링 \(p. 159\)](#) 단원을 참조하십시오.

무제한 모드 예제

다음은 `unlimited`로 구성된 인스턴스에 크레딧 사용을 설명하는 예입니다.

예제

- [예 1: T3 무제한에서 크레딧을 사용하는 것에 관한 설명 \(p. 138\)](#)
- [예 2: T2 무제한에서 크레딧을 사용하는 것에 관한 설명 \(p. 139\)](#)

예 1: T3 무제한에서 크레딧을 사용하는 것에 관한 설명

이 예에서는 `unlimited`로 시작된 `t3.nano` 인스턴스의 CPU 사용률과 CPU 사용률 유지를 위해 획득 및 잉여 크레딧을 어떻게 사용하고 있는지 보여줍니다.

`t3.nano` 인스턴스는 24시간 동안 144개의 CPU 크레딧을 획득하고, 이를 사용하여 144분의 vCPU 사용 시간을 확보할 수 있습니다. CPU 크레딧 밸런스(CloudWatch 측정치 CPUCreditBalance에 의해 표현)가 고갈되면 인스턴스는 아직 획득되지 않은 잉여 CPU——크레딧을 사용하여 필요한 시간 동안 버스트를 할 수 있습니다. `t3.nano` 인스턴스는 24시간 동안 최대 144개의 크레딧을 획득하기 때문에 즉시 요금을 부과하지 않고 이 최대 값까지 잉여 크레딧을 소비할 수 있습니다. 144개 이상의 CPU 크레딧을 사용하고 있는 경우에는 해당 시간이 끝날 때 그 차이만큼 비용이 부과됩니다.

이 예제는 다음 그래프를 통해 CPUCreditBalance가 감소한 이후에도 인스턴스가 잉여 크레딧을 사용하여 어떻게 버스트를 할 수 있는지 보여줍니다. 아래 워크플로는 그래프에서 번호가 매겨진 지점을 참조합니다.

P1 – 그래프의 0시간에서 인스턴스는 `unlimited`로 시작되며 즉시 크레딧을 획득하기 시작합니다. 인스턴스는 시작된 시간부터 유휴 상태로 유지되어 CPU 사용률이 0%이므로 크레딧이 사용되지 않습니다. 사용하지 않은 모든 크레딧은 크레딧 밸런스에 누적됩니다. 처음 24시간 동안 CPUCreditUsage는 0이고 CPUCreditBalance 값은 최대 144에 이릅니다.

P2 – 향후 12시간 동안 CPU 사용률은 2.5%이며, 이는 5% 기준 아래입니다. 인스턴스는 사용하는 크레딧보다 더 많은 크레딧을 획득하지만, CPUCreditBalance 값은 최대 144 크레딧을 초과할 수 없습니다.

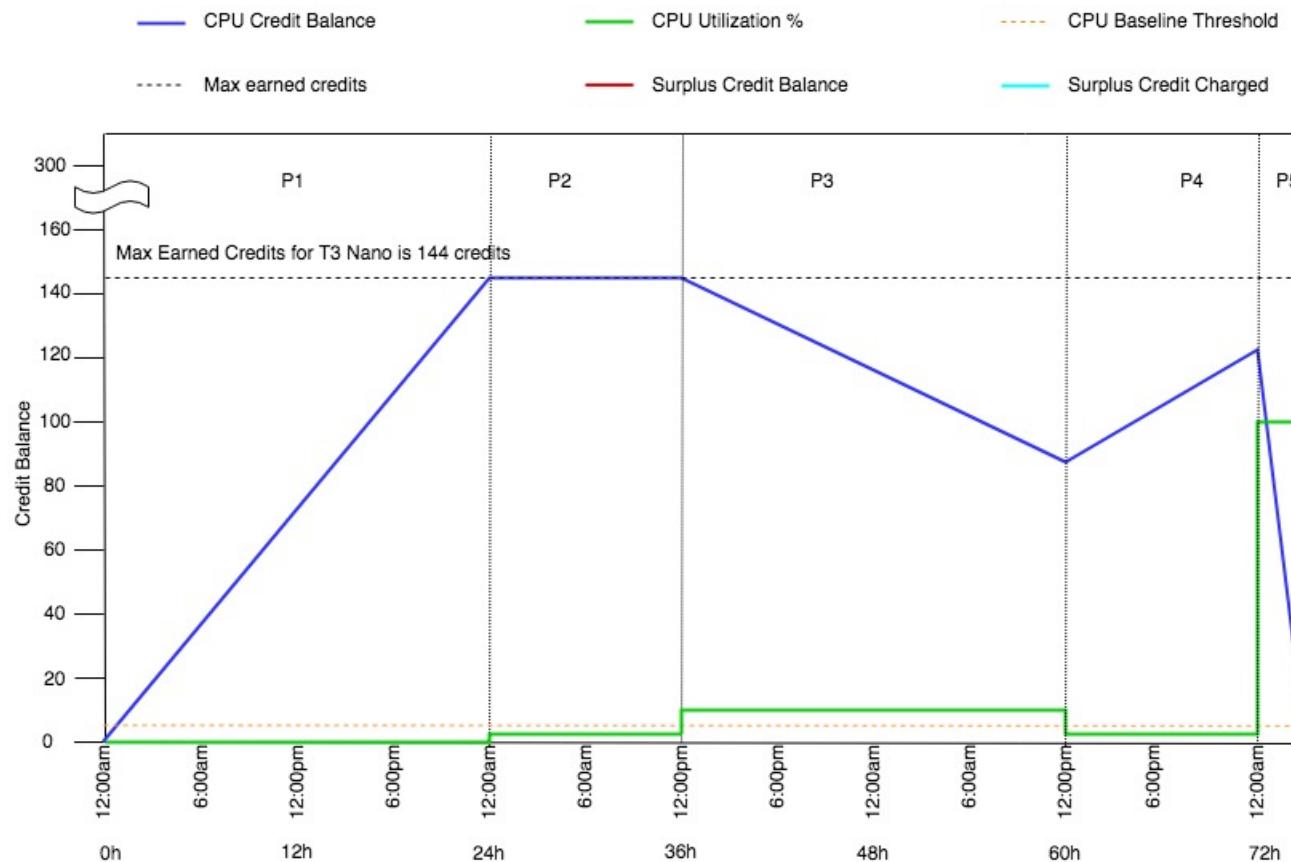
P3 – 향후 24시간 동안 CPU 사용률은 7%(기준보다 높음)이며, 이를 위해서는 57.6 크레딧을 사용해야 합니다. 인스턴스는 획득한 것보다 더 많은 크레딧을 사용하므로 CPUCreditBalance 값은 86.4 크레딧으로 감소합니다.

P4 – 향후 12시간 동안 CPU 사용률은 2.5%(기준보다 낮음)로 감소하며, 이를 위해서는 36 크레딧을 사용해야 합니다. 인스턴스에서는 동시에 72 크레딧을 획득할 수 있습니다. 인스턴스는 사용하는 크레딧보다 더 많은 크레딧을 획득하므로 CPUCreditBalance 값은 122 크레딧으로 증가합니다.

P5 – 향후 5시간 동안 인스턴스는 100% CPU 사용률로 버스트하고 이 버스트를 지속하기 위해 총 570 크레딧을 사용합니다. 이 기간 중 1시간이 지나면 인스턴스는 122 크레딧의 전체 CPUCreditBalance를 소진하고 높은 CPU 사용률을 유지하기 위해 잉여 크레딧을 사용하기 시작해 이 기간 동안 총 448 잉여 크레딧 ($570 - 122 = 448$)을 사용합니다. CPUSurplusCreditBalance 값이 144 CPU 크레딧(`t3.nano` 인스턴스는 24시간 동안 획득할 수 있는 최대 크레딧)에 이르면 이후에 사용된 모든 잉여 크레딧은 획득한 크레딧으로 상쇄되지 않습니다. 이후에 사용된 잉여 크레딧은 304 크레딧($448 - 144 = 304$)에 해당하며, 이로써 304 크레딧에 대한 시간이 종료될 때 약간의 추가 요금이 발생하게 됩니다.

P6 – 향후 13시간 동안 CPU 사용률은 5%(기준)입니다. 인스턴스는 사용하는 크레딧과 동일한 크레딧을 획득하므로 CPUSurplusCreditBalance를 청산할 여력은 없습니다. CPUSurplusCreditBalance 값은 144 크레딧을 유지합니다.

P7 – 이 예에서는 최근 24시간 동안 인스턴스가 유휴 상태로, CPU 사용률이 0%입니다. 이 기간 동안 인스턴스는 144 크레딧을 획득하고 이 크레딧은 CPUSurplusCreditBalance를 청산하는 데 사용합니다.



예 2: T2 무제한에서 크레딧을 사용하는 것에 관한 설명

이 예에서는 `unlimited`로 시작된 `t2.nano` 인스턴스의 CPU 사용률과 CPU 사용률을 유지를 위해 획득 및 잉여 크레딧을 어떻게 사용하고 있는지 보여줍니다.

`t2.nano` 인스턴스는 24시간 동안 72개의 CPU 크레딧을 획득하고, 이를 사용하여 72분의 vCPU 사용 시간을 확보할 수 있습니다. CPU 크레딧 밸런스(CloudWatch 측정치 `CPUCreditBalance`에 의해 표현)가 고갈되면 인스턴스는 아직 획득되지 않은 잉여 CPU —— 크레딧을 사용하여 필요한 시간 동안 버스트를 할 수 있습니다. `t2.nano` 인스턴스는 24시간 동안 최대 72개의 크레딧을 획득하기 때문에 즉시 요금을 부과하지 않고 이 최대 값까지 잉여 크레딧을 소비할 수 있습니다. 72개 이상의 CPU 크레딧을 사용하고 있는 경우에는 해당 시간이 끝날 때 그 차이만큼 비용이 부과됩니다.

이 예제는 다음 그래프를 통해 `CPUCreditBalance`가 감소한 이후에도 인스턴스가 잉여 크레딧을 사용하여 어떻게 버스트를 할 수 있는지 보여줍니다. 그래프의 타임 라인 시작 지점에서 인스턴스가 24시간 동안 획득할 수 있는 최대 수와 동일한 크레딧 밸런스를 누적했다고 가정할 수 있습니다. 아래 워크플로는 그래프에서 번호가 매겨진 지점을 참조합니다.

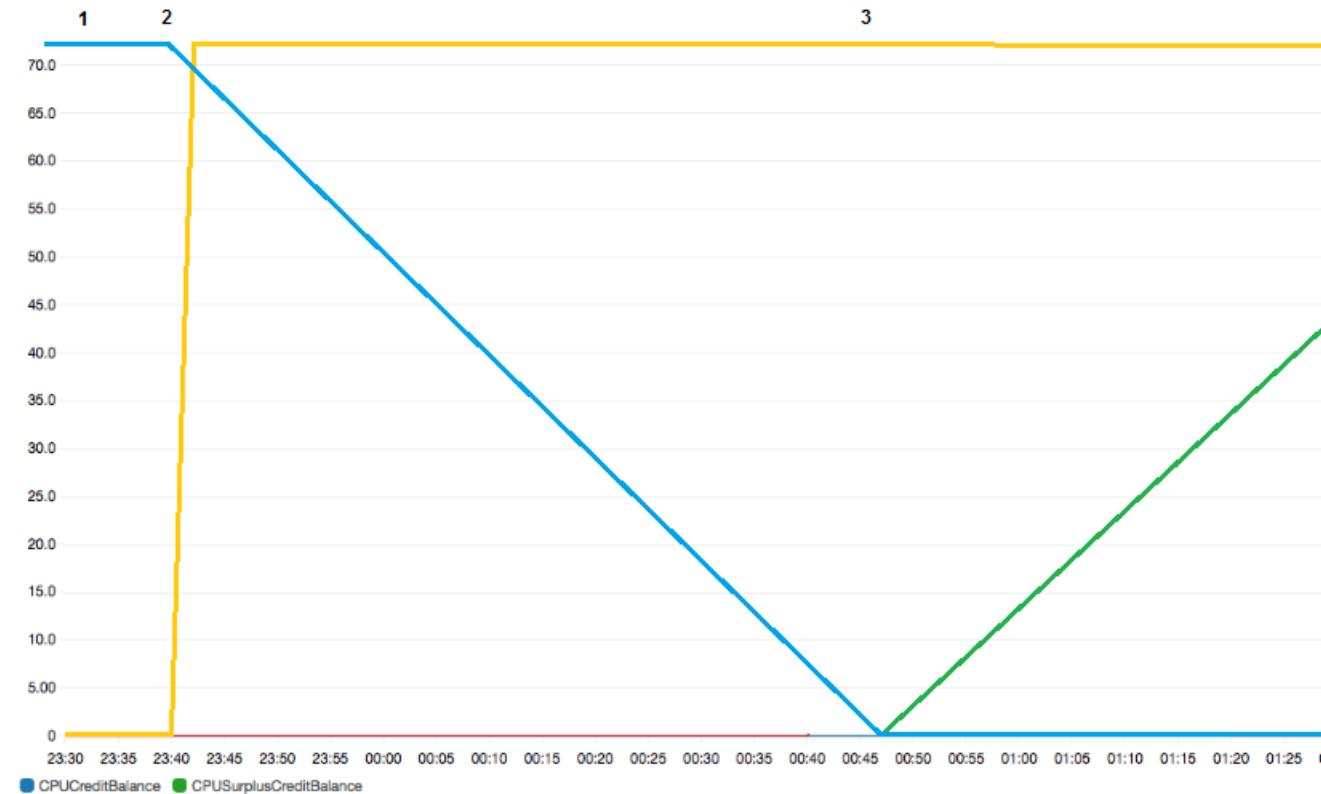
1 – 처음 10분 동안 `CPUCreditUsage`가 0이고 `CPUCreditBalance` 값이 최대 72로 유지됩니다.

2 – 23:40에 CPU 사용률이 증가하면서 인스턴스가 CPU 크레딧을 사용하고, 이에 따라 `CPUCreditBalance` 값이 줄어듭니다.

3 – 00:47경, 인스턴스에서 전체 CPUCreditBalance가 고갈되고 높은 CPU 사용률을 유지하기 때문에 잉여 크레딧을 사용하기 시작합니다.

4 – CPUSurplusCreditBalance 값이 72 CPU 크레딧에 도달하는 01:55까지 잉여 크레딧이 사용됩니다. 이는 t2.nano 인스턴스가 24시간 동안 획득할 수 있는 최대 값과 동일합니다. 이후에 사용된 모든 잉여 크레딧은 24시간 내에 획득한 크레딧으로 상쇄가 되지 않기 때문에 해당 시간이 끝날 때 약간의 추가 요금이 발생하게 됩니다.

5 – 인스턴스가 02:20경까지 잉여 크레딧을 계속해 사용합니다. 이때 CPU 사용률이 기준 이하로 떨어지면 인스턴스는 시간당 3개씩(5분마다 0.25개) 크레딧을 획득하기 시작합니다. 이는 CPUSurplusCreditBalance를 청산하는 데 사용됩니다. CPUSurplusCreditBalance 값이 줄어들어 0이 되고 나면 인스턴스는 5분마다 0.25개씩 CPUCreditBalance 획득 크레딧을 누적하기 시작합니다.



	Label	Details
■	CPUCreditBalance	EC2 * InstanceId:i-0aa4b948d7eb37d6b * CPUCreditBalance
■	CPUCreditUsage	EC2 * InstanceId:i-0aa4b948d7eb37d6b * CPUCreditUsage
■	CPUSurplusCreditBalance	EC2 * InstanceId:i-0aa4b948d7eb37d6b * CPUSurplusCreditBalance
■	CPUSurplusCreditsCharged	EC2 * InstanceId:i-0aa4b948d7eb37d6b * CPUSurplusCreditsCharged

청구서 계산

vCPU-시간당 잉여 크레딧의 요금은 0.096 USD입니다. 인스턴스는 01:55부터 02:20까지 약 25개의 잉여 크레딧을 소비했으며, 이는 0.42 vCPU-시간에 해당됩니다.

이 인스턴스에 대한 추가 요금은 $0.42 \text{ vCPU-시간} \times 0.096 \text{ USD/vCPU-시간} = 0.04032 \text{ USD}$ (반올림해서 0.04 USD).

여기 이 T2 무제한 인스턴스에 대한 월말 청구서가 나와 있습니다.

Amazon Elastic Compute Cloud running Windows		
\$0.0081 per On Demand Windows t2.nano Instance Hour	720.000 Hrs	\$5.83
Amazon Elastic Compute Cloud T2 CPU Credits		
\$0.096 per vCPU-Hour of T2 CPU credits	0.420 vCPU-Hours	\$0.04

발생하는 모든 요금을 매시간 공지하는 청구서 알림을 설정하고 필요 시 조치를 취할 수 있습니다.

성능 순간 확장 가능 인스턴스의 스탠다드 모드

standard로 구성된 성능 순간 확장 가능 인스턴스는 평균 CPU 사용률이 인스턴스의 기준 CPU 사용률보다 일관되게 낮은 워크로드에 적합합니다. 기준 이상으로 버스트하려면 인스턴스는 CPU 크레딧 밸런스에 누적한 크레딧을 사용합니다. 인스턴스가 획득한 크레딧이 부족해지면 CPU 사용률이 점차적으로 기준 수준으로 떨어지기 때문에 획득한 CPU 크레딧 밸런스가 고갈되어도 급격한 성능 저하가 발생하지 않습니다. 자세한 정보는 [성능 순간 확장 가능 인스턴스에 대한 CPU 크레딧 및 기준 사용률 \(p. 130\)](#) 단원을 참조하십시오.

내용

- [스탠다드 모드 개념 \(p. 141\)](#)
 - [스탠다드 성능 순간 확장 가능 인스턴스의 작동 방식 \(p. 141\)](#)
 - [시작 크레딧 \(p. 142\)](#)
 - [시작 크레딧 한도 \(p. 142\)](#)
 - [시작 크레딧과 획득 크레딧의 차이 \(p. 143\)](#)
- [스탠다드 모드 예제 \(p. 143\)](#)
 - [예 1: T3 스탠다드에서 크레딧을 사용하는 것에 관한 설명 \(p. 143\)](#)
 - [예 2: T2 스탠다드에서 크레딧을 사용하는 것에 관한 설명 \(p. 145\)](#)
 - [기간 1: 1 – 24시간 \(p. 146\)](#)
 - [기간 2: 25 – 36시간 \(p. 147\)](#)
 - [기간 3: 37 – 61시간 \(p. 148\)](#)
 - [기간 4: 62 – 72시간 \(p. 149\)](#)
 - [기간 5: 73 – 75시간 \(p. 151\)](#)
 - [기간 6: 76 – 90시간 \(p. 152\)](#)
 - [기간 7: 91 – 96시간 \(p. 153\)](#)

스탠다드 모드 개념

standard 모드는 성능 순간 확장 가능 인스턴스에 사용할 수 있는 구성 옵션입니다. 이 모드는 실행 중인 또는 중지된 인스턴스에 대해 언제든지 활성화 또는 비활성화할 수 있습니다. 성능 순간 확장 가능 인스턴스 패밀리마다 AWS 리전별 계정 수준에서 standard를 기본 크레딧 옵션으로 설정하여 계정에 속한 새 성능 순간 확장 가능 인스턴스가 모두 기본 크레딧 옵션을 사용하여 시작되게 할 수 있습니다.

T3 이상 인스턴스는 [unlimited](#)로 시작하도록 기본 설정되어 있습니다. T2 인스턴스는 standard로 시작하도록 기본 설정되어 있습니다. AWS 리전별로 계정 수준에서 기본값을 변경할 수 있습니다. 자세한 내용은 [계정의 기본 크레딧 사양 설정 \(p. 159\)](#) 단원을 참조하십시오.

스탠다드 성능 순간 확장 가능 인스턴스의 작동 방식

standard로 구성된 성능 순간 확장 가능 인스턴스는 실행 중 상태인 경우 시간당 특정 비율의 획득 크레딧을 지속적으로 (밀리초 수준의 시간 정밀도로) 획득합니다. T2 스탠다드 인스턴스가 중지되면 발생한 크레딧

이 모두 손실되고 크레딧 밸런스가 0으로 재설정됩니다. 인스턴스가 다시 시작되면 새로운 세트의 시작 크레딧이 지급되고 획득 크레딧이 누적되기 시작합니다. T3 이상 표준 인스턴스의 경우 인스턴스가 중지된 후 CPU 크레딧 밸런스가 7일 동안 지속하다가 7일이 지나면 크레딧이 상실됩니다. 7일 이내에 인스턴스를 시작하면 크레딧이 상실되지 않습니다.

T2 표준 인스턴스는 두 가지 유형의 CPU 크레딧을 획득합니다. 획득 크레딧 및 시작 크레딧. T2 스탠다드 인스턴스가 실행 중 상태인 경우 지속적으로 시간당 특정 비율의 획득 크레딧을 획득합니다(밀리초 수준의 시간 정밀도). 시작 시에는 아직 뛰어난 시작 환경을 위한 크레딧이 없으므로, 뛰어난 시작 환경을 제공하기 위해 획득 크레딧이 누적되는 동안 먼저 소비할 수 있도록 시작 시에 시작 크레딧이 지급됩니다.

T3 이상 표준 인스턴스는 시작 크레딧을 획득하지 않습니다.

시작 크레딧

T2 스탠다드 인스턴스는 시작 또는 재시작 시 vCPU당 30개의 시작 크레딧이 지급됩니다. 예를 들어 t2.micro 인스턴스는 1개의 vCPU에서 30개의 시작 크레딧을 획득하는 반면에 t2.xlarge 인스턴스는 4 개의 vCPU에서 120개의 시작 크레딧을 획득합니다. 시작 크레딧은 획득 크레딧을 누적하기 전에 인스턴스가 시작 즉시 버스트를 할 수 있도록 허용하는 뛰어난 시작 경험을 제공하도록 설계되었습니다.

시작 크레딧은 획득 크레딧보다 먼저 소비됩니다. 소비되지 않은 시작 크레딧은 CPU 크레딧 밸런스에 누적됩니다. 하지만 CPU 크레딧 밸런스 한도에 포함되지 않습니다. 예를 들어 t2.micro 인스턴스는 최대 144 의 CPU 크레딧 밸런스 한도를 가지고 있습니다. 시작된 후 24시간 이상 유휴 상태로 지속된 경우 CPU 크레딧 밸런스는 한도 이상인 174(시작 크레딧 30 + 획득 크레딧 144)에 도달합니다. 그러나 인스턴스가 30개의 시작 크레딧을 사용하고 나면 크레딧 밸런스가 144개를 초과할 수 없습니다. 각 인스턴스 크기별 CPU 크레딧 밸런스 한도에 대한 자세한 내용은 [크레딧 표 \(p. 131\)](#)를 참조하십시오.

아래 표에는 시작 시 획득한 초기 CPU 크레딧 할당과 vCPU의 수가 나와 있습니다.

인스턴스 유형	시작 크레딧	vCPUs
t1.micro	15	1
t2.nano	30	1
t2.micro	30	1
t2.small	30	1
t2.medium	60	2
t2.large	60	2
t2.xlarge	120	4
t2.2xlarge	240	8

시작 크레딧 한도

T2 스탠다드 인스턴스가 시작 크레딧을 획득할 수 있는 횟수에는 제한이 있습니다. 기본 한도는 24시간마다 계정, 리전 및 24시간당 모든 T2 스탠다드 인스턴스에 대해 총 100회 시작입니다. 예를 들어 한 인스턴스가 24시간 이내에 100회 중지 및 시작되는 경우, 24시간 이내에 100개의 인스턴스가 시작되는 경우 또는 기타 조합으로 100회의 시작에 도달한 경우 한도에 도달하게 됩니다. 새 계정에는 사용량에 따라 증가하는 하한이 설정되어 있을 수 있습니다.

Tip

워크로드가 항상 필요한 성능을 얻도록 하려면 [성능 순간 확장 가능 인스턴스의 무제한 모드 \(p. 133\)](#) 전환 또는 크기가 더 큰 인스턴스 사용을 고려하십시오.

시작 크레딧과 획득 크레딧의 차이

다음 표에는 시작 크레딧과 획득 크레딧의 차이가 나와 있습니다.

	시작 크레딧	획득 크레딧
크레딧 획득률	T2 스탠다드 인스턴스는 시작 또는 재시작 시 vCPU당 30개의 시작 크레딧이 지급됩니다. T2 인스턴스가 unlimited 에서 standard 로 전환되는 경우 전환되는 시점에는 이 인스턴스에서 시작 크레딧을 획득하지 않습니다.	각 T2 인스턴스는 인스턴스 크기에 따라 지속적으로 특정 비율의 시간당 CPU 크레딧을 얻습니다(밀리초 수준의 시간 정밀도로). 인스턴스 크기에 따라 지급되는 CPU 크레딧 수에 대한 자세한 내용은 크레딧 표 (p. 131) 를 참조하십시오.
크레딧 획득 한도	시작 크레딧 획득 한도는 24시간마다 계정, 리전 및 24시간당 모든 T2 스탠다드 인스턴스에 대해 총 100회 시작입니다. 새 계정에는 사용량에 따라 증가하는 하한이 설정되어 있을 수 있습니다.	T2 인스턴스는 CPU 크레딧 밸런스 한도 이상의 크레딧을 누적할 수 없습니다. CPU 크레딧 밸런스가 한도에 도달한 경우 한도 도달 이후 획득한 모든 크레딧은 삭제됩니다. 시작 크레딧은 한도에 포함되지 않습니다. 각 T2 인스턴스 크기별 CPU 크레딧 밸런스 한도에 대한 자세한 내용은 크레딧 표 (p. 131) 를 참조하십시오.
크레딧 사용	시작 크레딧은 획득 크레딧보다 먼저 소비됩니다.	획득 크레딧은 모든 시작 크레딧이 소비된 후에만 소비됩니다.
크레딧 만료	T2 인스턴스가 실행 중인 동안 시작 크레딧은 만료되지 않습니다. T2 스탠다드 인스턴스가 종단되거나 T2 무제한으로 전환될 때 모든 시작 크레딧이 삭제됩니다.	T2 인스턴스가 실행 중일 때는 누적된 획득 크레딧이 만료되지 않습니다. T2 인스턴스가 종지되면 누적된 획득 크레딧이 모두 상실됩니다.

누적된 시작 크레딧 및 획득 크레딧의 수는 CloudWatch 지표 [CPUCreditBalance](#)를 통해 추적됩니다. 자세한 내용은 [CloudWatch 지표 \(p. 160\)](#) 표에서 CPUCreditBalance 항목을 참조하십시오.

스탠다드 모드 예제

다음은 인스턴스가 [standard](#)로 구성되었을 때의 크레딧 사용을 설명하는 예입니다.

Note

[standard](#)로 구성된 T3 이상 인스턴스는 시작 크레딧을 획득하지 않습니다.

예제:

- [예 1: T3 스탠다드에서 크레딧을 사용하는 것에 관한 설명 \(p. 143\)](#)
- [예 2: T2 스탠다드에서 크레딧을 사용하는 것에 관한 설명 \(p. 145\)](#)

예 1: T3 스탠다드에서 크레딧을 사용하는 것에 관한 설명

이 예에서는 [standard](#)로 시작된 [t3.nano](#) 인스턴스가 획득 크레딧을 획득, 축적, 사용하는 방식을 보여줍니다. 이로써 누적된 획득 크레딧이 크레딧 밸런스에 반영되는 방식을 알 수 있습니다.

실행 중인 [t3.nano](#) 인스턴스는 24시간마다 144개 크레딧을 획득합니다. 크레딧 밸런스 한도는 획득 크레딧 144개입니다. 한도에 도달하면 새로 획득한 크레딧이 삭제됩니다. 획득 및 누적될 수 있는 크레딧 수에 대한 자세한 내용은 [크레딧 표 \(p. 131\)](#)를 참조하십시오.

T3 스탠다드 인스턴스를 시작하고 즉시 사용할 수 있습니다. 또는 T3 스탠다드 인스턴스를 시작하고 애플리케이션을 실행하기 전에 며칠 동안 유휴 상태로 둘 수 있습니다. 인스턴스를 사용했는지 아니면 유휴 상태로 두었는지에 따라 크레딧이 사용되는지 또는 누적되는지가 결정됩니다. 인스턴스가 시작된 시간부터 24시간 동안 유휴 상태로 유지된 경우 크레딧 밸런스는 한도에 이릅니다. 여기서 한도는 누적될 수 있는 획득 크레딧의 최대 수입니다.

이 예에서는 시작 시간부터 24시간 동안 유휴 상태로 유지된 인스턴스에 대해 설명하며, 96시간 기간 동안 7 단계 기간을 통해 크레딧이 획득, 누적, 사용되고 폐기되는 비율과 각 기간 종료 시 크레딧 밸런스의 값을 보여 줍니다.

아래 워크플로는 그래프에서 번호가 매겨진 지점을 참조합니다.

P1 – 그래프의 0시간에서 인스턴스는 standard로 시작되며 즉시 크레딧을 획득하기 시작합니다. 인스턴스는 시작된 시간부터 유휴 상태로 유지되어 CPU 사용률이 0%이므로 크레딧이 사용되지 않습니다. 사용하지 않은 모든 크레딧은 크레딧 밸런스에 누적됩니다. 처음 24시간 동안 CPUCreditUsage는 0이고 CPUCreditBalance 값은 최대 144에 이릅니다.

P2 – 향후 12시간 동안 CPU 사용률은 2.5%이며, 이는 5% 기준 아래입니다. 인스턴스는 사용하는 크레딧보다 더 많은 크레딧을 획득하지만, CPUCreditBalance 값은 최대 144 크레딧을 초과할 수 없습니다. 한도를 초과하여 획득한 모든 크레딧은 삭제됩니다.

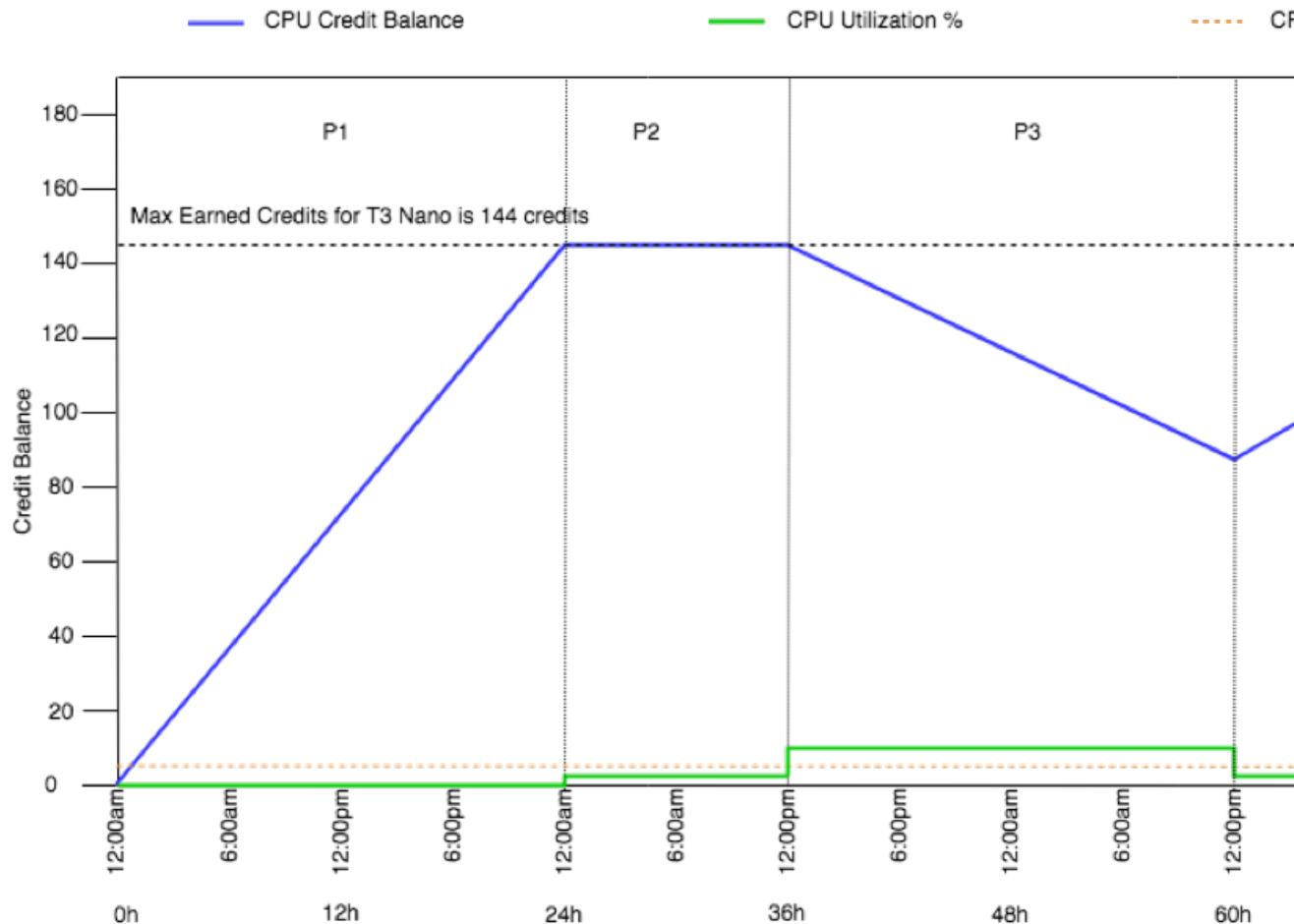
P3 – 향후 24시간 동안 CPU 사용률은 7%(기준보다 높음)이며, 이를 위해서는 57.6 크레딧을 사용해야 합니다. 인스턴스는 획득한 것보다 더 많은 크레딧을 사용하므로 CPUCreditBalance 값은 86.4 크레딧으로 감소합니다.

P4 – 향후 12시간 동안 CPU 사용률은 2.5%(기준보다 낮음)로 감소하며, 이를 위해서는 36 크레딧을 사용해야 합니다. 인스턴스에서는 동시에 72 크레딧을 획득할 수 있습니다. 인스턴스는 사용하는 크레딧보다 더 많은 크레딧을 획득하므로 CPUCreditBalance 값은 122 크레딧으로 증가합니다.

P5 – 향후 2시간 동안 인스턴스는 100% CPU 사용률로 버스트하고 전체 CPUCreditBalance인 122크레딧을 소진합니다. 이 기간이 종료되는 시점에 CPUCreditBalance가 0이고, CPU 사용률은 강제로 5%의 기준 사용률 수준으로 하락합니다. 기준 수준에서 인스턴스는 사용하는 크레딧과 동일한 크레딧을 획득합니다.

P6 – 향후 14시간 동안 CPU 사용률은 5%(기준)입니다. 인스턴스는 사용하는 크레딧과 동일한 크레딧을 획득합니다. CPUCreditBalance 값은 0을 유지합니다.

P7 – 이 예에서는 최근 24시간 동안 인스턴스가 유휴 상태로, CPU 사용률이 0%입니다. 이 기간 동안 인스턴스는 144크레딧을 획득하고 이 크레딧은 CPUCreditBalance에 누적됩니다.



예 2: T2 스탠다드에서 크레딧을 사용하는 것에 관한 설명

이 예제는 standard로 실행된 t2.nano 인스턴스가 어떻게 시작 및 획득 크레딧을 획득하고 축적하고 사용하는지를 보여줍니다. 크레딧 밸런스에 획득 크레딧의 누적뿐 아니라 시작 크레딧의 누적이 어떻게 반영되는지 볼 수 있습니다.

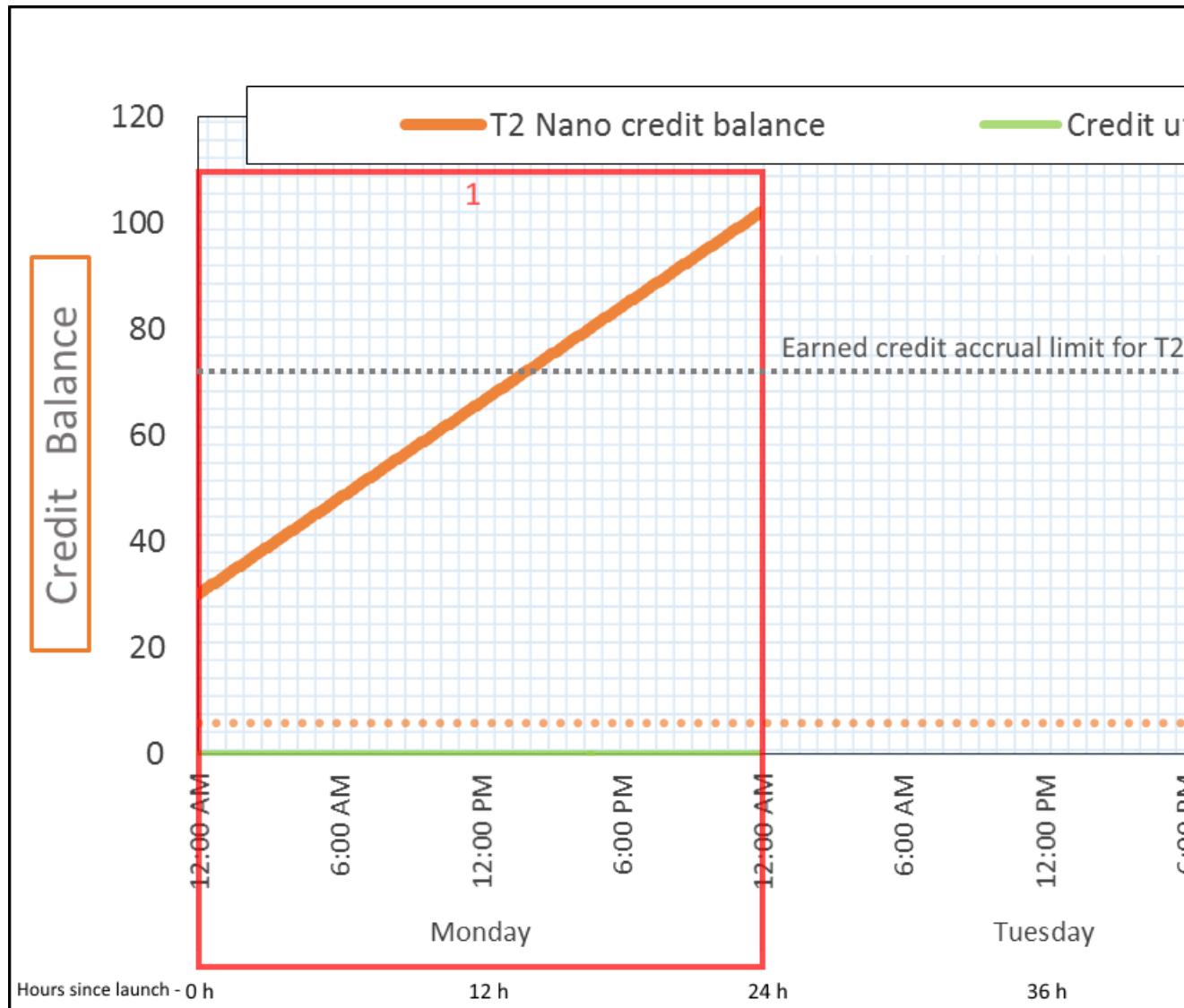
t2.nano 인스턴스는 시작 시 30개의 시작 크레딧을 받고 24시간마다 72개의 크레딧을 획득합니다. 크레딧 밸런스 한도는 획득 크레딧 72개이며, 시작 크레딧은 한도에 포함되지 않습니다. 한도에 도달하면 새로 획득한 크레딧이 삭제됩니다. 획득 및 누적될 수 있는 크레딧 수에 대한 자세한 내용은 [크레딧 표 \(p. 131\)](#)를 참조하십시오. 제한에 대한 자세한 내용은 [시작 크레딧 한도 \(p. 142\)](#) 단원을 참조하십시오.

T2 스탠다드 인스턴스를 시작하고 즉시 사용할 수 있습니다. 또는 T2 스탠다드 인스턴스를 시작하고 애플리케이션을 실행하기 전에 며칠 동안 유휴 상태로 둘 수 있습니다. 인스턴스를 사용했는지 아니면 유휴 상태로 두었는지에 따라 크레딧이 사용되는지 또는 누적되는지가 결정됩니다. 인스턴스가 시작 시간부터 24시간 동안 유휴 상태로 유지된 경우, 잔고에 획득 누적 크레딧과 시작 누적 크레딧이 모두 반영되어 잔고가 한도를 초과한 것으로 나타납니다. 하지만 CPU가 사용되면 시작 크레딧이 먼저 사용됩니다. 그 후 한도에는 누적될 수 있는 최대 획득 크레딧이 항상 반영됩니다.

이 예에서는 시작 시간부터 24시간 동안 유휴 상태로 유지된 인스턴스에 대해 설명하며, 96시간 기간 동안 7 단계 기간을 통해 크레딧이 획득, 누적, 사용되고 폐기되는 비율과 각 기간 종료 시 크레딧 밸런스의 값을 보여 줍니다.

기간 1: 1 – 24시간

그래프의 0시간에서 T2 인스턴스는 standard로 시작되며 30개의 시작 크레딧을 바로 받습니다. 인스턴스가 실행 상태일 때 크레딧을 획득합니다. 인스턴스는 시작된 시간부터 유휴—상태로 유지되어 CPU 사용률이 0%—이므로 크레딧이 사용되지 않습니다. 사용하지 않은 모든 크레딧은 크레딧 밸런스에 누적됩니다. 시작 후 약 14시간이 되면 크레딧 밸런스가 72(시작 크레딧 30 + 획득 크레딧 42)개가 되고, 이 값은 인스턴스가 24시간 안에 획득할 수 있는 값과 동일합니다. 시작 후 24시간이 경과하면 사용하지 않은 시작 크레딧이 크레딧 밸런스에 누적되기 때문에 크레딧 밸런스가 72개를 초과합니다. 즉 크레딧 밸런스는 102(시작 크레딧 30 + 획득 크레딧 72)입니다.—



크레딧 사용률	24시간당 0 크레딧(CPU 사용률 0%)
크레딧 획득률	24시간당 72 크레딧
크레딧 폐기율	24시간당 0 크레딧
크레딧 밸런스	102 크레딧(시작 크레딧 30 + 획득 크레딧 72)

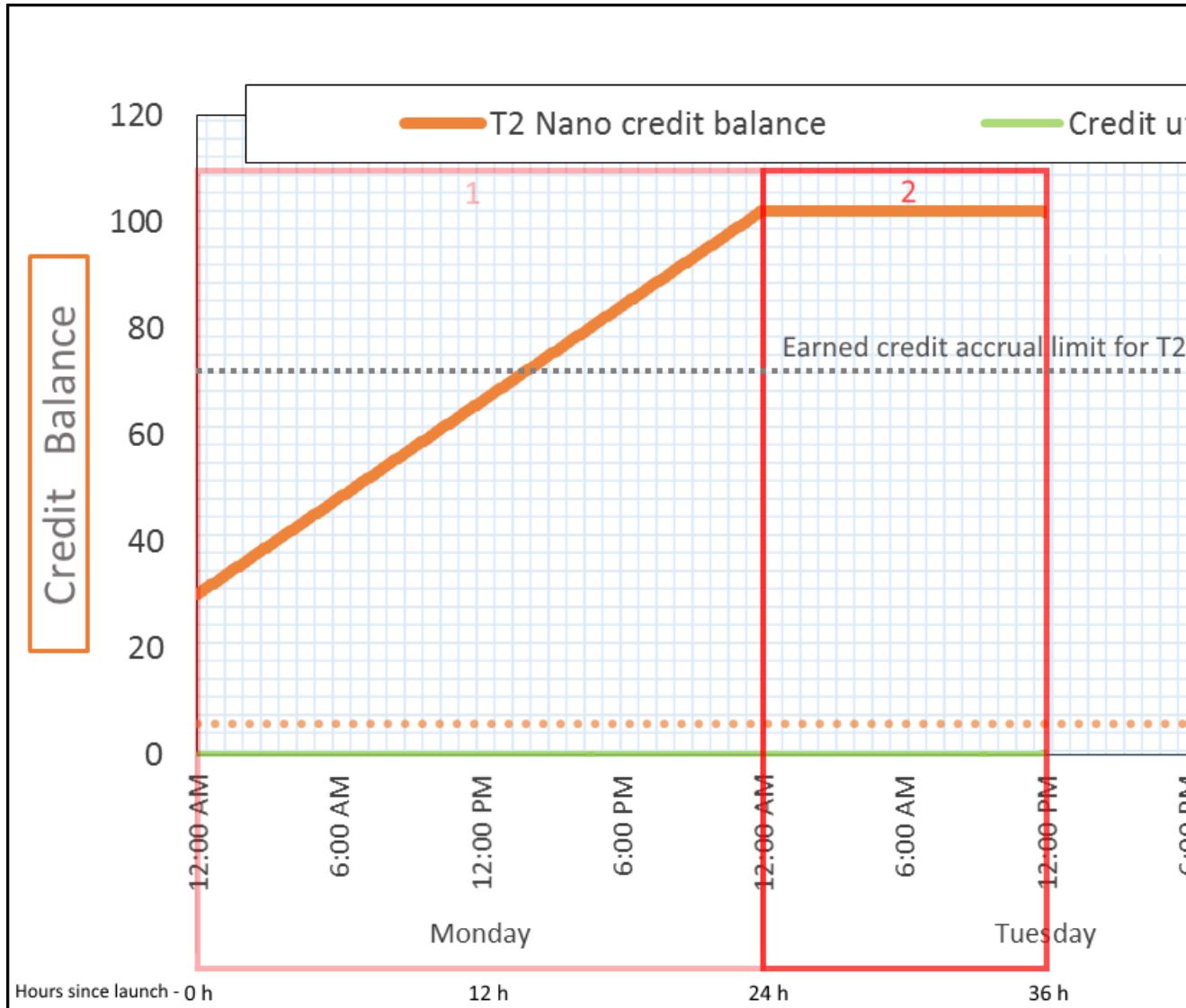
결론

시작 후 CPU를 사용하지 않으면 24시간 후에 적립할 수 있는 크레딧보다 더 많은 크레딧이 인스턴스에 적립됩니다(시작 크레딧 30 + 획득 크레딧 72 = 102 크레딧).

실제 상황에서 EC2 인스턴스는 시작 및 실행 중에 적은 양의 크레딧을 사용하므로 잔고는 이 예에서의 이론적인 최댓값에 도달하지 않습니다.

기간 2: 25 – 36시간

다음 12시간 동안 인스턴스는 계속 유휴 상태이고 크레딧을 획득하지만 크레딧 밸런스는 증가하지 않습니다. 102 크레딧(시작 크레딧 30 + 획득 크레딧 72)에서 더 이상 증가하지 않습니다. 크레딧 밸런스가 한도인 72개 획득 누적 크레딧에 도달한 경우 새로 획득한 크레딧은 버려집니다.



크레딧 사용률	24시간당 0 크레딧(CPU 사용률 0%)
크레딧 획득률	24시간당 72 크레딧(시간당 3 크레딧)
크레딧 폐기율	24시간당 72 크레딧(크레딧 획득률 100%)

크레딧 밸런스

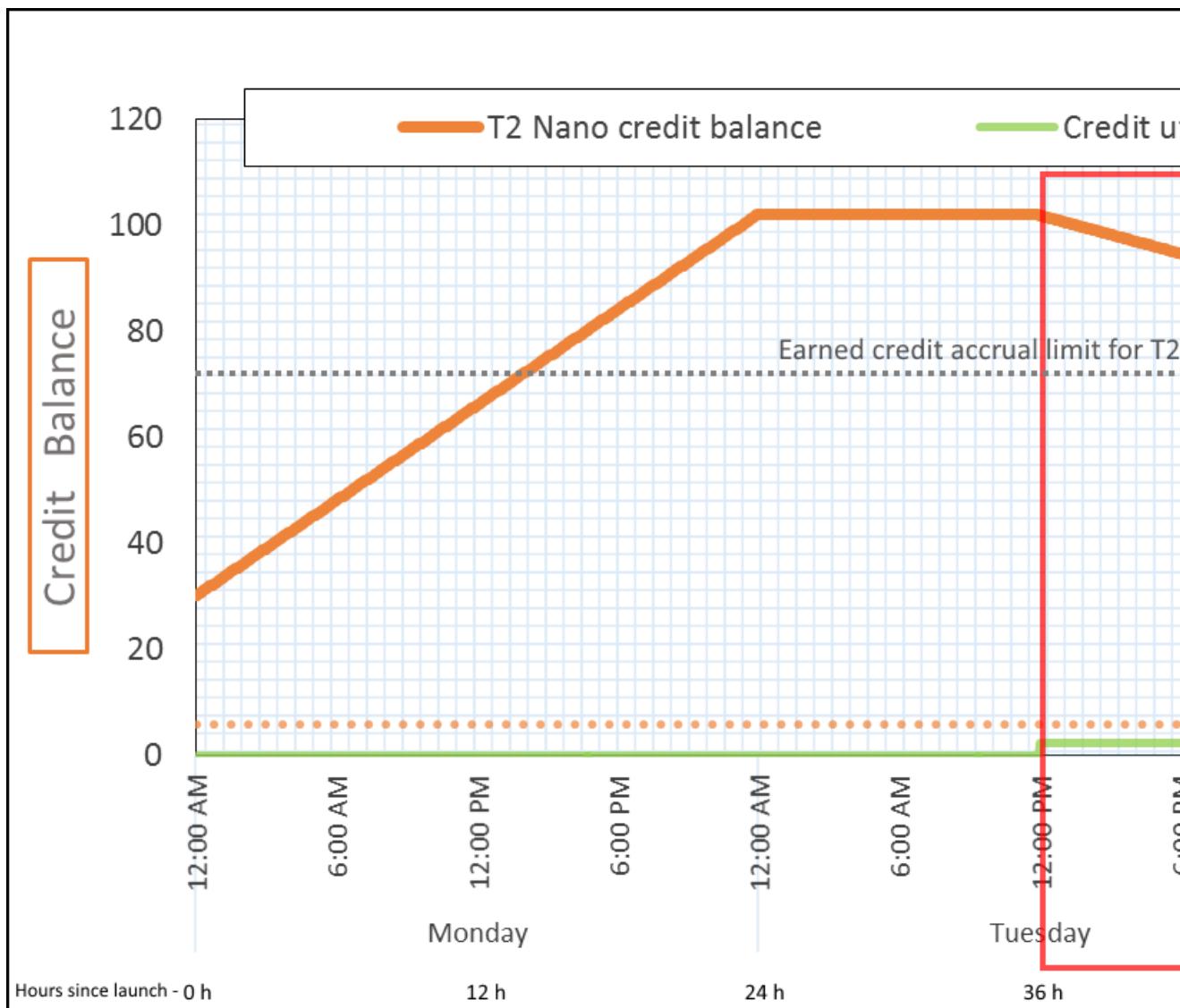
102크레딧(시작 크레딧 30 + 획득 크레딧 72) — — 잔고 변경 없음

결론

인스턴스는 크레딧을 계속 획득하지만 크레딧 밸런스가 한도에 도달하면 획득 크레딧을 더 이상 누적할 수 없습니다. 한도에 도달한 후 새로 획득한 크레딧은 버려집니다. 시작 크레딧은 크레딧 밸런스 한도에 포함되지 않습니다. 잔고에 시작 누적 크레딧이 포함되면 잔고가 한도를 초과한 것으로 나타납니다.

기간 3: 37 – 61시간

다음 25시간 동안 인스턴스는 2% CPU를 사용하며 이는 30 크레딧이 필요합니다. 동일한 기간에서 75 크레딧을 획득하지만 크레딧 밸런스는 감소합니다. 누적된 시작 크레딧이 처음 사용되고, 크레딧 밸런스가 이미 획득 크레딧 한도 72에 도달함에 따라 새로 획득한 크레딧은 버려지기 때문에 잔고가 감소합니다.



크레딧 사용률	24시간당 28.8 크레딧(시간당 1.2 크레딧, 2% CPU 사용률, 크레딧 획득률 400%) – 25시간 동안 30 크레딧—
크레딧 획득률	24시간당 72 크레딧
크레딧 폐기율	24시간당 72 크레딧(크레딧 획득률 100%)
크레딧 밸런스	72 크레딧(시작 크레딧 30개가 사용되고, 획득 크레딧 72개는 사용하지 않은 상태로 유지됨)

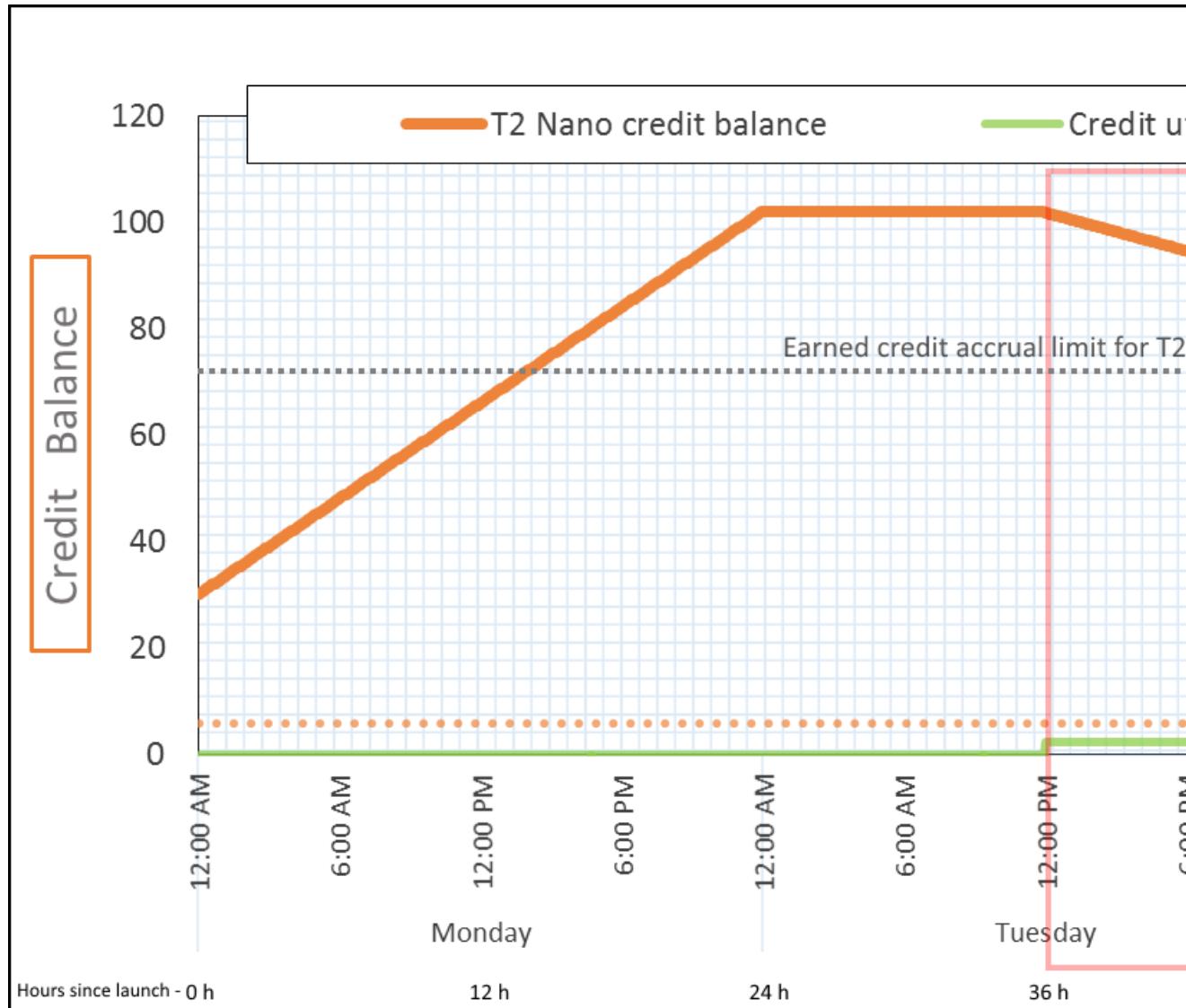
결론

인스턴스는 획득 크레딧을 사용하기 전에 시작 크레딧을 먼저 사용합니다. 시작 크레딧은 크레딧 한도에 포함되지 않습니다. 시작 크레딧이 사용된 후에는 24시간만에 획득할 수 있는 크레딧보다 잔고 더 많아지는 경우가 없습니다. 또한 인스턴스가 실행 중인 동안에는 시작 크레딧을 획득할 수 없습니다.

기간 4: 62 – 72시간

다음 11시간 동안 인스턴스는 2% CPU를 사용하며 이는 13.2 크레딧이 필요합니다. CPU 사용률은 이전 기간과 동일하지만 잔고는 감소하지 않습니다. 72 크레딧으로 유지됩니다.

크레딧 획득률이 크레딧 사용률보다 높기 때문에 잔고가 감소하지 않습니다. 인스턴스는 13.2개 크레딧을 사용하는 동안 33개 크레딧을 획득합니다. 하지만 잔고 한도는 72개이므로 이 한도를 초과하는 획득 크레딧은 버려집니다. 잔고는 72개로 유지되고, 이 값이 기간 2에서 102개 크레딧으로 유지된 것과 다른 이유는 획득 크레딧이 누적되지 않기 때문입니다.



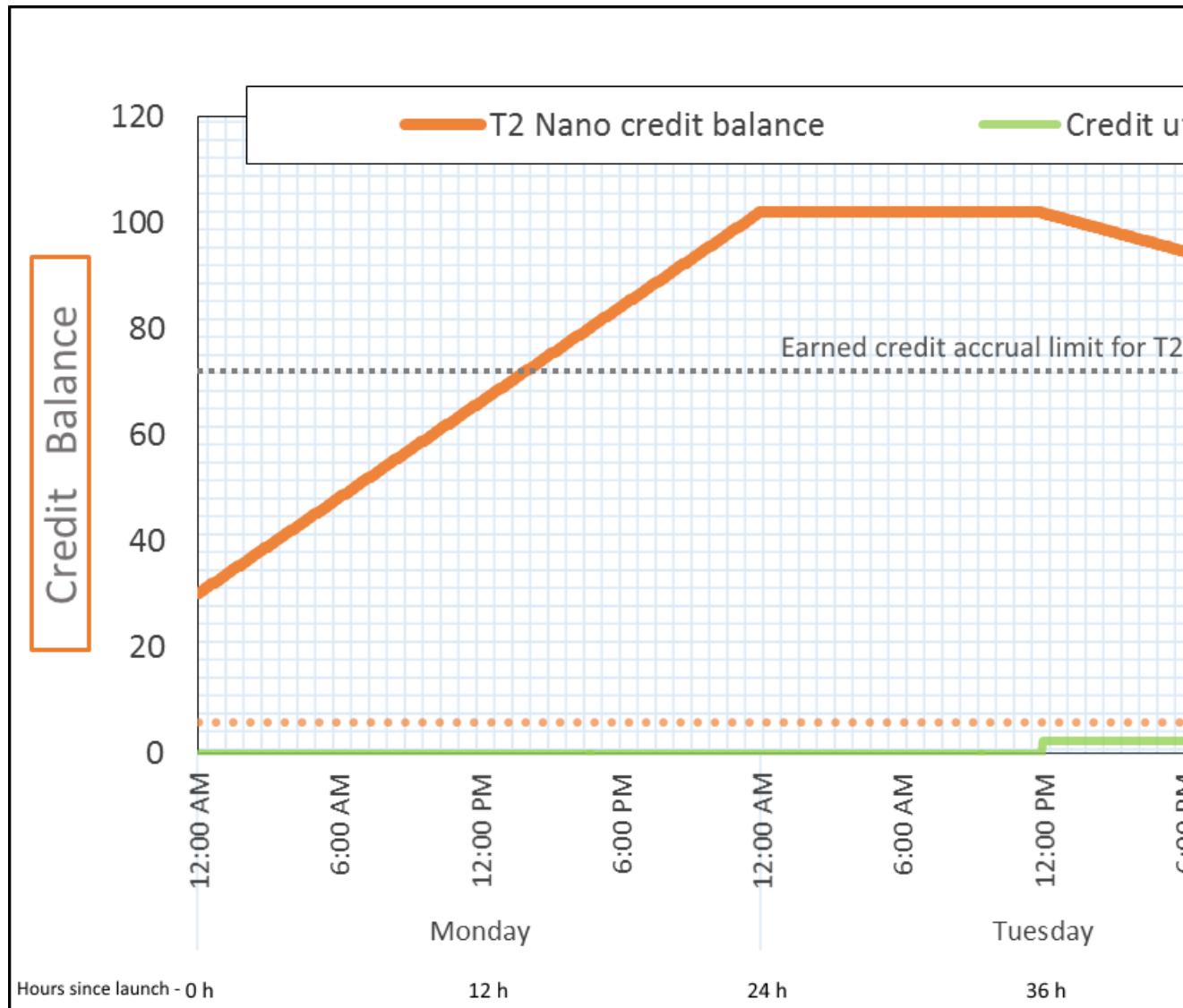
크레딧 사용률	24시간당 28.8 크레딧(시간당 1.2 크레딧, 2% CPU 사용률, 크레딧 획득률 40%) – 11시간 동안 13.2 크레딧—
크레딧 획득률	24시간당 72 크레딧
크레딧 폐기율	24시간당 43.2 크레딧(크레딧 획득률 60%)
크레딧 밸런스	72 크레딧(시작 크레딧 0, 획득 크레딧 72) – 잔고가 한도에 이룸

결론

시작 크레딧이 사용된 후에는 인스턴스가 24시간만에 획득할 수 있는 크레딧 수에 따라 크레딧 밸런스 한도가 결정됩니다. 인스턴스가 사요한 것보다 더 많은 크레딧을 획득한 경우 새로 획득한 크레딧 중 한도를 초과하는 크레딧은 버려집니다.

기간 5: 73 – 75시간

다음 3시간 동안 인스턴스의 CPU 사용률은 20%가 되고 36개의 크레딧을 사용합니다. 인스턴스는 이 3시간 동안 9개의 크레딧을 획득하므로 실제로 크레딧 밸런스는 27개가 감소합니다. 3시간이 지나면 크레딧 밸런스는 45개(획득 누적 크레딧)가 됩니다.



크레딧 사용률	24시간당 288 크레딧(시간당 12 크레딧, 20% CPU 사용률, 크레딧 획득률 400%) – 3시간 동안 36 크레딧
크레딧 획득률	24시간당 72 크레딧(3시간 동안 9 크레딧)
크레딧 폐기율	24시간당 0 크레딧
크레딧 밸런스	45 크레딧(이전 잔고(72) - 사용한 크레딧(36) + 획득한 크레딧(9)) – 24시간당 잔고 감소율 216개(사용률 288/24 + 획득률 72/24 = 잔고 감소율 216/24)

결론

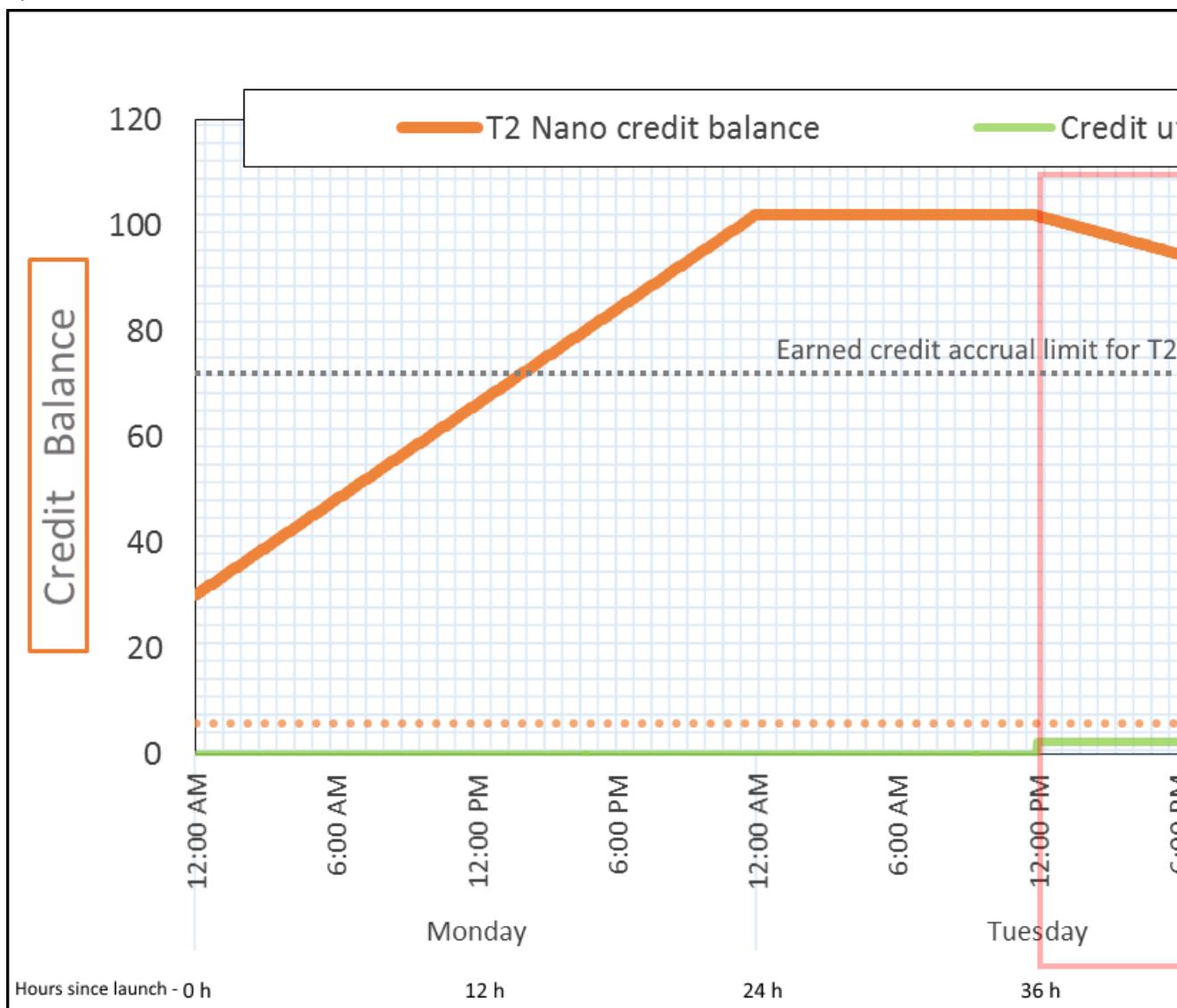
인스턴스가 획득한 것보다 더 많은 크레딧을 사용한 경우 크레딧 밸런스가 감소합니다.

기간 6: 76 – 90시간

다음 15시간 동안 인스턴스는 2% CPU를 사용하며 이는 18 크레딧이 필요합니다. 기간 3 및 4와 동일한 CPU 사용률입니다. 하지만 기간 3에서는 잔고가 감소하고, 기간 4에서는 잔고가 그대로 유지된 반면, 이 기간 동안에는 잔고가 증가합니다.

기간 3에서는 누적된 시작 크레딧이 사용되었고, 크레딧 한도를 초과하는 획득 크레딧은 모두 버려졌기 때문에 크레딧 밸런스가 감소했습니다. 기간 4에서는 인스턴스가 획득한 것보다 더 적은 크레딧을 사용했습니다. 한도를 초과하는 획득한 크레딧은 폐기되고, 잔고는 최대 72 크레딧으로 유지됩니다.

이 기간에는 누적된 시작 크레딧이 없고 잔고에 누적된 획득 크레딧이 한도보다 적습니다. 획득된 크레딧이 버려지지 않습니다. 또한 인스턴스는 사용한 것보다 더 많은 크레딧을 획득하므로 크레딧 밸런스가 증가합니다.



크레딧 사용률	24시간당 28.8 크레딧(시간당 1.2 크레딧, 2% CPU 사용률, 크레딧 획득률 40%) — 15시간 동안 18 크레딧
크레딧 획득률	24시간당 72 크레딧(15시간 동안 45 크레딧)
크레딧 폐기율	24시간당 0 크레딧
크레딧 밸런스	72 크레딧(24시간당 잔고 증가율 43.2 크레딧 — 변화율 = 사용률 28.8/24 + 획득률 72/24)

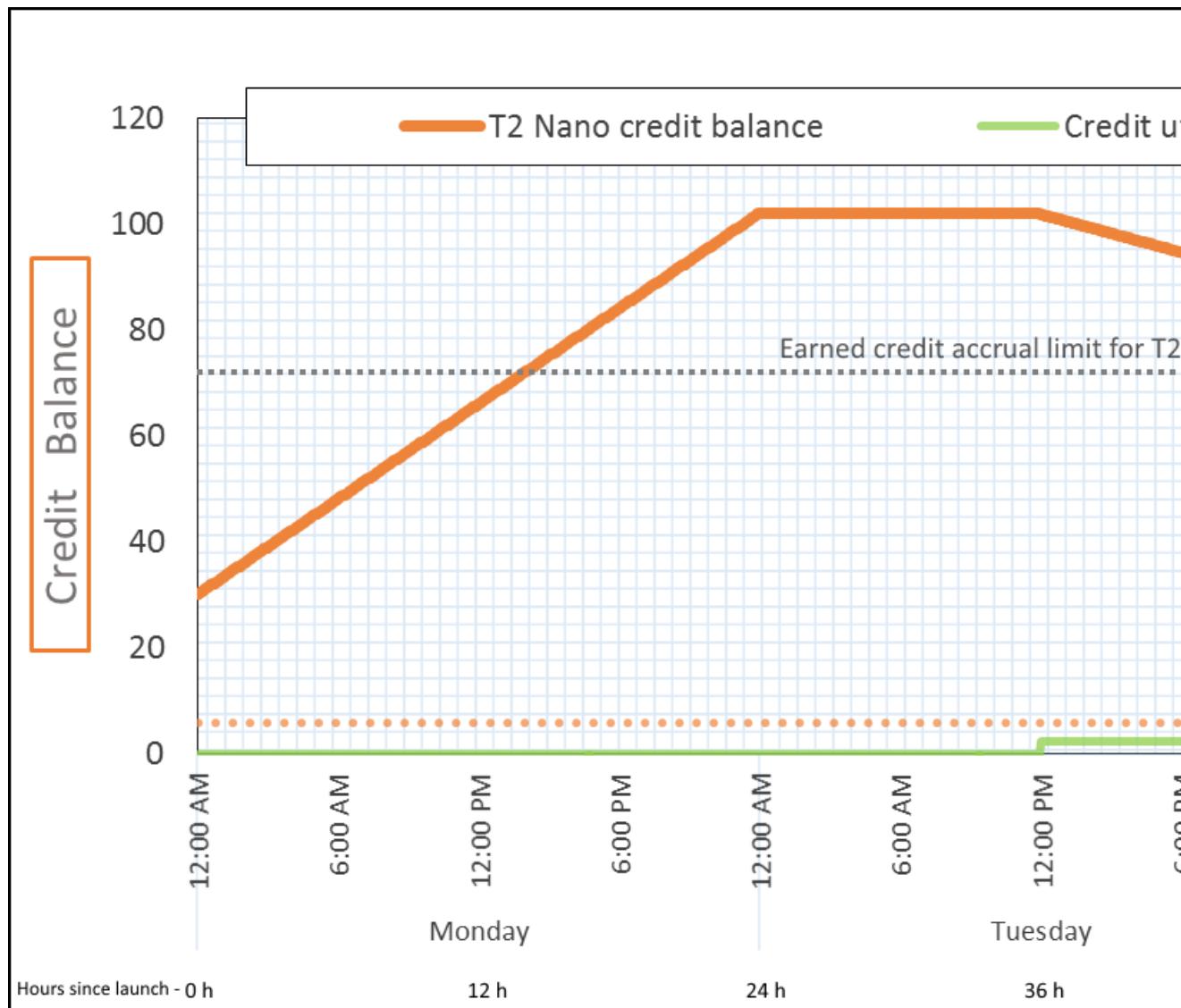
결론

인스턴스가 획득한 것보다 적은 크레딧을 사용한 경우 크레딧 밸런스가 증가합니다.

기간 7: 91 – 96시간

다음 6시간 동안 인스턴스는 유휴 상태로 유지되어 —CPU 사용률이 0%이므로 크레딧이 사용되지 않습니다. 기간 2의 CPU 사용률과 동일하지만 잔고는 102 크레딧으로 유지되지 않고 인스턴스 크레딧 밸런스 한도인— 72 크레딧으로 유지됩니다.

기간 2에서 크레딧 밸런스에는 누적된 시작 크레딧 30개가 포함됩니다. 기간 3에서는 시작 크레딧이 사용되었습니다. 실행 중 인스턴스는 더 이상 시작 크레딧을 받을 수 없습니다. 크레딧 밸런스 한도에 도달한 후, 획득 크레딧 중 한도를 초과하는 크레딧은 버려집니다.



크레딧 사용률	24시간당 0 크레딧(CPU 사용률 0%)
크레딧 획득률	24시간당 72 크레딧
크레딧 폐기율	24시간당 72 크레딧(크레딧 획득률 100%)
크레딧 밸런스	72 크레딧(시작 크레딧 0, 획득 크레딧 72)

결론

인스턴스는 크레딧을 계속 획득하지만 크레딧 밸런스 한도에 도달하면 획득 크레딧을 더 이상 누적할 수 없습니다. 한도에 도달한 후 새로 획득한 크레딧은 버려집니다. 크레딧 밸런스 한도는 인스턴스가 24시간 만에 획득할 수 있는 크레딧 수에 따라 결정됩니다. 크레딧 밸런스 한도에 대한 자세한 내용은 [크레딧 표 \(p. 131\)](#)를 참조하십시오.

성능 순간 확장 가능 인스턴스의 작업

이러한 인스턴스 시작, 모니터링 및 수정 절차는 서로 유사합니다. 주요 차이점은 인스턴스가 시작할 때 기본 적용되는 크레딧 사양입니다. 기본 크레딧 사양을 변경하지 않을 경우 기본값은 다음과 같습니다.

- T3 이상 인스턴스는 `unlimited`로 시작하도록 기본 설정되어 있습니다.
- T2 인스턴스는 `standard`로 시작하도록 기본 설정되어 있습니다.

내용

- [무제한 또는 스탠다드로 성능 순간 확장 가능 인스턴스 시작 \(p. 155\)](#)
- [Auto Scaling 그룹을 사용하여 성능 순간 확장 가능 인스턴스를 무제한으로 시작 \(p. 155\)](#)
- [성능 순간 확장 가능 인스턴스의 크레딧 사양 보기 \(p. 157\)](#)
- [성능 순간 확장 가능 인스턴스의 크레딧 사양 수정 \(p. 158\)](#)
- [계정의 기본 크레딧 사양 설정 \(p. 159\)](#)
- [기본 크레딧 사양 보기 \(p. 159\)](#)

무제한 또는 스탠다드로 성능 순간 확장 가능 인스턴스 시작

Amazon EC2 콘솔, AWS SDK, 명령줄 도구 또는 Auto Scaling 그룹을 사용하여 인스턴스를 `unlimited` 또는 `standard`로 시작할 수 있습니다. 자세한 정보는 [Auto Scaling 그룹을 사용하여 성능 순간 확장 가능 인스턴스를 무제한으로 시작 \(p. 155\)](#) 단원을 참조하십시오.

무제한 또는 스탠다드로 성능 순간 확장 가능 인스턴스를 시작하려면(콘솔)

1. [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#)의 절차를 따르십시오.
2. Choose an Instance Type(인스턴스 유형 선택) 페이지에서 인스턴스 유형을 선택하고 Next: Configure Instance Details(다음: 인스턴스 정보 구성)를 선택합니다.
3. 다음과 같이 크레딧 사양을 선택합니다.
 - a. T3 이상 인스턴스를 `standard`로 시작하려면 무제한(Unlimited)의 선택을 취소합니다.
 - b. T2 인스턴스를 `unlimited`로 시작하려면 무제한(Unlimited)을 선택합니다.
4. 마법사에 표시되는 지침에 따라 계속합니다. 인스턴스 시작 검토 페이지에서 옵션 검토를 마쳤으면 시작을 선택합니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오.

무제한 또는 스탠다드로 성능 순간 확장 가능 인스턴스를 시작하려면(AWS CLI)

`run-instances` 명령을 사용하여 인스턴스를 시작합니다. `--credit-specification CpuCredits=` 파라미터를 사용하여 크레딧 사양을 지정합니다. 유효한 크레딧 사양은 `unlimited` 및 `standard`입니다.

- T3 이상의 경우 `--credit-specification` 파라미터를 포함하지 않으면 인스턴스가 `unlimited`로 시작되도록 기본 설정되어 있습니다.
- T2의 경우 `--credit-specification` 파라미터를 포함하지 않으면 인스턴스가 `standard`로 시작되도록 기본 설정되어 있습니다.

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t3.micro --key-name MyKeyPair --credit-specification "CpuCredits=unlimited"
```

Auto Scaling 그룹을 사용하여 성능 순간 확장 가능 인스턴스를 무제한으로 시작

성능 확장 가능 인스턴스가 시작되면 CPU 크레딧 없이도 좋은 부트스트래핑 환경을 경험할 수 있습니다. Auto Scaling 그룹을 사용하여 인스턴스를 시작하는 경우 인스턴스를 `unlimited`로 구성합니다. 그러한 경

우 인스턴스는 Auto Scaling 그룹에서 자동으로 시작 또는 재시작될 때 잉여 크레딧을 사용합니다. 잉여 크레딧을 사용하면 성능 제한을 막을 수 있습니다.

시작 템플릿 생성

Auto Scaling 그룹에서 인스턴스를 `unlimited`로 시작하는 데 시작 템플릿을 사용해야 합니다. 시작 구성에서는 인스턴스를 `unlimited`로 시작하는 것은 지원하지 않습니다.

인스턴스를 무제한으로 시작하는 시작 템플릿을 생성하려면(콘솔)

1. [Auto Scaling 그룹에 대한 시작 템플릿 생성](#) 절차를 수행하십시오.
2. 시작 템플릿 콘텐츠(Launch template contents)의 인스턴스 유형(Instance type)에서 인스턴스 크기를 선택합니다.
3. Auto Scaling 그룹에서 인스턴스를 `unlimited`으로 시작하려면 고급 세부 정보(Advanced details) 아래의 크레딧 사양(Credit specification)에서 무제한(Unlimited)을 선택합니다.
4. 시작 템플릿 파라미터 정의를 완료한 경우 시작 템플릿 생성을 선택합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [Auto Scaling 그룹에 대한 시작 템플릿 생성](#)을 참조하십시오.

인스턴스를 무제한으로 시작하는 시작 템플릿을 생성하려면(AWS CLI)

`create-launch-template` 명령을 사용하고 `unlimited`를 크레딧 사양으로 지정합니다.

- T3 이상의 경우 `CreditSpecification={CpuCredits=unlimited}` 값을 포함하지 않으면 인스턴스가 `unlimited`로 시작되도록 기본 설정되어 있습니다.
- T2의 경우 `CreditSpecification={CpuCredits=unlimited}` 값을 포함하지 않으면 인스턴스가 `standard`로 시작되도록 기본 설정되어 있습니다.

```
aws ec2 create-launch-template --launch-template-name MyLaunchTemplate
--version-description FirstVersion --launch-template-data
ImageId=ami-8c1be5f6, InstanceType=t3.medium, CreditSpecification={CpuCredits=unlimited}
```

Auto Scaling 그룹을 시작 템플릿에 연결

Auto Scaling 그룹에 시작 템플릿을 연결하려면 시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하거나 기존 Auto Scaling 그룹에 시작 템플릿을 추가합니다.

시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 시작 템플릿을 만들 때 사용한 리전과 동일한 리전을 선택합니다.
3. 탐색 창에서 Auto Scaling 그룹을 선택하고 Auto Scaling 그룹 생성을 선택합니다.
4. 시작 템플릿을 선택하고 시작 템플릿을 선택한 후 다음 단계를 선택합니다.
5. Auto Scaling 그룹 관련 필드를 작성합니다. 검토 페이지에서 구성 설정 검토를 마쳤으면 Auto Scaling 그룹 생성을 선택합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [시작 템플릿을 사용한 Auto Scaling 그룹 생성](#)을 참조하십시오.

시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하려면(AWS CLI)

`create-auto-scaling-group` AWS CLI 명령을 사용하여 `--launch-template` 파라미터를 지정합니다.

기존 Auto Scaling 그룹에 시작 템플릿을 추가하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 화면 상단의 탐색 모음에서 시작 템플릿을 만들 때 사용한 리전과 동일한 리전을 선택합니다.
3. 탐색 창에서 Auto Scaling 그룹을 선택합니다.
4. Auto Scaling 그룹 목록에서 Auto Scaling 그룹을 선택하고 작업, 편집을 선택합니다.
5. 세부 정보 탭의 시작 템플릿에서 시작 템플릿을 선택한 다음, 저장을 선택합니다.

기존 Auto Scaling 그룹에 시작 템플릿을 추가하려면(AWS CLI)

`update-auto-scaling-group` AWS CLI 명령을 사용하여 `--launch-template` 파라미터를 지정합니다.

성능 순간 확장 가능 인스턴스의 크레딧 사양 보기

실행 중이거나 중지된 인스턴스의 크레딧 사양(unlimited 또는 standard)을 확인할 수 있습니다.

새로운 콘솔

버스트 가능 인스턴스의 크레딧 사양을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다.
4. 세부 정보(Details)를 선택하고 크레딧 사양(Credit specification) 필드를 확인합니다. 이때 값은 `unlimited` 또는 `standard`입니다.

기존 콘솔

버스트 가능 인스턴스의 크레딧 사양을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다.
4. 설명을 선택하고 T2/T3 무제한(T2/T3 Unlimited) 필드를 확인합니다.
 - 값이 `Enabled`이면 인스턴스가 `unlimited`로 구성됩니다.
 - 값이 `Disabled`이면 인스턴스가 `standard`로 구성됩니다.

성능 순간 확장 가능 인스턴스의 크레딧 사양을 보려면(AWS CLI)

`describe-instance-credit-specifications` 명령을 사용합니다. 하나 이상의 인스턴스 ID를 지정하지 않은 경우 크레딧 사양이 `unlimited`인 모든 인스턴스가 반환되고 이전에 `unlimited` 크레딧 사양으로 구성된 인스턴스 또한 반환됩니다. 예를 들어 T3 인스턴스가 `unlimited`로 구성된 가운데 이를 M4 인스턴스로 크기 조정하는 경우 Amazon EC2에서 M4 인스턴스를 반환합니다.

Example

```
aws ec2 describe-instance-credit-specifications --instance-id i-1234567890abcdef0
```

다음은 예제 출력입니다.

```
{
    "InstanceCreditSpecifications": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "CpuCredits": "unlimited"
        }
    ]
}
```

```
]  
}
```

성능 순간 확장 가능 인스턴스의 크레딧 사양 수정

실행 중이거나 종지된 인스턴스의 크레딧 사양을 `unlimited`와 `standard` 간에 언제든지 전환할 수 있습니다.

새로운 콘솔

버스트 가능한 성능 인스턴스의 크레딧 사양을 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 몇 가지 인스턴스에 대한 크레딧 사양을 한 번에 수정하려면 해당되는 인스턴스를 모두 선택합니다.
4. 작업(Actions), 인스턴스 설정(Instance settings), 크레딧 사양 변경(Change credit specification)을 선택합니다. 이 옵션은 버스트 가능한 성능 인스턴스를 선택한 경우에만 활성화됩니다.
5. 크레딧 사양을 `unlimited`으로 변경하려면 인스턴스 ID 옆에 있는 확인란을 선택합니다. 크레딧 사양을 `standard`으로 변경하려면 인스턴스 ID 옆에 있는 확인란의 선택을 취소합니다.

기존 콘솔

버스트 가능한 성능 인스턴스의 크레딧 사양을 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 몇 가지 인스턴스에 대한 크레딧 사양을 한 번에 수정하려면 해당되는 인스턴스를 모두 선택합니다.
4. 작업, 인스턴스 설정, T2/T3 무제한 변경(Change T2/T3 Unlimited)을 선택합니다. 이 옵션은 버스트 가능한 성능 인스턴스를 선택한 경우에만 활성화됩니다.
5. 현재 크레딧 사양은 인스턴스 ID 뒤의 괄호에 표시됩니다. 크레딧 사양을 `unlimited`로 변경하려면 활성화를 선택합니다. 크레딧 사양을 `standard`로 변경하려면 비활성화를 선택합니다.

버스트 가능한 성능 인스턴스의 크레딧 사양을 수정하려면(AWS CLI)

`modify-instance-credit-specification` 명령을 사용합니다. `--instance-credit-specification` 파라미터를 사용하여 인스턴스 및 크레딧 사양을 지정합니다. 유효한 크레딧 사양은 `unlimited` 및 `standard`입니다.

Example

```
aws ec2 modify-instance-credit-specification --region us-east-1 --instance-credit-specification "InstanceId=i-1234567890abcdef0,CpuCredits=unlimited"
```

다음은 예제 출력입니다.

```
{  
    "SuccessfulInstanceCreditSpecifications": [  
        {  
            "InstanceId": "i- 1234567890abcdef0"  
        }  
    ],  
    "UnsuccessfulInstanceCreditSpecifications": []  
}
```

}

계정의 기본 크레딧 사양 설정

AWS 리전별 계정 수준에서 기본 크레딧 사양을 설정할 수 있습니다. 인스턴스 패밀리별로 기본 크레딧 사양을 지정합니다(예: T2 또는 T3).

AWS Management 콘솔에서 인스턴스 시작 마법사를 사용하여 인스턴스를 시작하면 크레딧 사양에 대해 선택한 값이 계정 수준 기본 크레딧 사양을 재정의합니다. AWS CLI를 사용하여 인스턴스를 시작하면 계정에 속한 새 성능 순간 확장 가능 인스턴스가 모두 기본 크레딧 옵션을 사용하여 시작됩니다. 실행 중이거나 중지된 기존 인스턴스의 크레딧 사양은 영향을 받지 않습니다.

`modify-default-credit-specification` API는 AWS 리전 수준에서 작동하며 각 가용 영역의 크레딧 옵션을 수정하는 비동기 작업입니다. 5분 이내에 리전의 모든 영역이 업데이트됩니다. 하지만 이 작업 중에 인스턴스가 시작되면 영역이 업데이트될 때까지 인스턴스가 새 크레딧 옵션을 가져오지 못할 수도 있습니다. 업데이트가 발생했는지 확인하려면 `get-default-credit-specification`을 호출하고 기본 크레딧 사양에서 업데이트를 확인하면 됩니다. 자세한 내용은 [기본 크레딧 사양 보기 \(p. 159\)](#) 단원을 참조하십시오.

Note

인스턴스 패밀리의 기본 크레딧 사양은 5분 동안 한 번만 수정할 수 있으며, 24시간 동안 4회까지 수정할 수 있습니다.

계정 수준에서 기본 크레딧 사양을 설정하려면(AWS CLI)

`modify-default-credit-specification` 명령을 사용하십시오. `--cpu-credits` 파라미터를 사용하여 AWS 리전, 인스턴스 패밀리 및 기본 크레딧 사양을 지정합니다. 유효한 기본 크레딧 사양은 `unlimited` 및 `standard`입니다.

```
aws ec2 modify-default-credit-specification --region us-east-1 --instance-family t2 --cpu-credits unlimited
```

기본 크레딧 사양 보기

AWS 리전별로 계정 수준에서 성능 순간 확장 가능 인스턴스 패밀리의 기본 크레딧 사양을 볼 수 있습니다.

계정 수준에서 기본 크레딧 사양을 보려면(AWS CLI)

`get-default-credit-specification` 명령을 사용하십시오. AWS 리전과 인스턴스 패밀리를 지정합니다.

```
aws ec2 get-default-credit-specification --region us-east-1 --instance-family t2
```

CPU 크레딧 모니터링

각 인스턴스의 크레딧 밸런스를 CloudWatch 콘솔의 Amazon EC2 인스턴스별 측정치로 확인할 수 있습니다.

독차

- 성능 순간 확장 가능 인스턴스에 대한 추가 CloudWatch 측정치 (p. 159)
- CPU 크레딧 사용량 계산 (p. 161)

성능 순간 확장 가능 인스턴스에 대한 추가 CloudWatch 측정치

버스트 가능한 성능 인스턴스에는 다음과 같은 추가 CloudWatch 지표가 있으며 5분마다 업데이트됩니다.

- CPUCreditUsage – 측정 기간 중에 소비한 CPU 크레딧 수.

- **CPUCreditBalance** – 한 인스턴스에서 발생한 CPU 크레딧 수입니다. CPU에서 버스트가 발생하고 CPU 크레딧이 획득 속도보다 빠르게 소비될 때 크레딧 밸런스가 고갈됩니다.
- **CPUSurplusCreditBalance** – CPUCreditBalance 값이 0일 때 CPU 사용률을 유지하기 위해 소비되는 잉여 CPU 크레딧 수.
- **CPUSurplusCreditsCharged** – 24시간 동안 획득할 수 있는 최대 CPU 크레딧 수 (p. 131)를 초과하여 추가 요금을 유발하는 잉여 CPU 크레딧 수.

마지막 두 측정치는 **unlimited**로 구성된 인스턴스에만 적용됩니다.

다음 표에서는 성능 순간 확장 가능 인스턴스에 대한 CloudWatch 측정치를 설명합니다. 자세한 내용은 [인스턴스에 사용 가능한 CloudWatch 지표 나열 \(p. 669\)](#) 단원을 참조하십시오.

지표	설명
CPUCreditUsage	<p>CPU 사용률을 위해 인스턴스에서 소비되는 CPU 크레딧의 수입니다. CPU 크레딧 하나는 1분 동안 100%의 사용률로 실행되는 vCPU 1개 또는 이와 동등한 vCPU, 사용률 및 시간의 조합과 동일합니다(예를 들어 2분 동안 50%의 사용률로 실행되는 vCPU 1개 또는 2분 동안 25%의 사용률로 실행되는 vCPU 2개).</p> <p>CPU 크레딧 측정치는 5분 간격으로만 제공됩니다. 5분 이상의 시간을 지정할 경우 Sum 통계 대신 Average 통계를 사용하십시오.</p> <p>단위: 크레딧 (vCPU-분)</p>
CPUCreditBalance	<p>시작 이후 인스턴스가 누적한 획득 CPU 크레딧 수입니다. T2 스탠다드의 경우 CPUCreditBalance에 누적된 시작 크레딧 수도 포함됩니다.</p> <p>크레딧은 획득 이후에 크레딧 밸런스에 누적되고, 소비 시 크레딧 밸런스에서 소멸됩니다. 크레딧 밸런스는 최대 한도(인스턴스 크기에 따라 결정)가 있습니다. 한도에 도달하면 새로 획득한 크레딧이 모두 삭제됩니다. T2 스탠다드의 경우 시작 크레딧은 한도에 포함되지 않습니다.</p> <p>CPUCreditBalance의 크레딧은 인스턴스가 기준 CPU 사용률 이상으로 버스터를 하는 데 소비할 수 있습니다.</p> <p>인스턴스가 실행 중인 동안 CPUCreditBalance의 크레딧은 만료되지 않습니다. T3 이상 인스턴스가 종지되면 CPUCreditBalance 값은 7일 동안 지속됩니다. 그 이후에는 누적된 크레딧이 모두 삭제됩니다. T2 인스턴스가 종지되면 CPUCreditBalance 값은 지속되지 않고 누적된 크레딧이 모두 삭제됩니다.</p> <p>CPU 크레딧 측정치는 5분 간격으로만 제공됩니다.</p> <p>단위: 크레딧 (vCPU-분)</p>
CPUSurplusCreditBalance	<p>CPUCreditBalance 값이 0일 때 unlimited 인스턴스에서 소비된 잉여 크레딧의 수입니다.</p> <p>획득한 CPU 크레딧에 따라 CPUSurplusCreditBalance 값이 청산됩니다. 잉여 크레딧의 수가 인스턴스가 24시간 동안 획득할 수 있는 최대 크레딧 수를 초과한 경우 최대 값 이상으로 소비된 잉여 크레딧은 추가 요금으로 부과됩니다.</p> <p>단위: 크레딧 (vCPU-분)</p>

지표	설명
CPUSurplusCreditsCharged	<p>획득한 CPU 크레딧으로 청산되지 않는 소비 잉여 크레딧의 수로, 추가 요금으로 부과됩니다.</p> <p>소비된 잉여 크레딧은 다음이 발생할 때 요금이 부과됩니다.</p> <ul style="list-style-type: none"> 소비한 잉여 크레딧이 인스턴스가 24시간 동안 획득할 수 있는 최대 크레딧 수를 초과하는 경우. 해당 시간이 끝날 때 최대 값 이상으로 소비한 잉여 크레딧에 요금이 부과됩니다. 인스턴스가 중지 또는 종료된 경우. 인스턴스가 <code>unlimited</code>에서 <code>standard</code>로 전환됩니다. <p>단위: 크레딧 (vCPU-분)</p>

CPU 크레딧 사용량 계산

인스턴스의 CPU 크레딧 사용량은 앞 표에 설명되어 있는 인스턴스 CloudWatch 측정치를 사용해 계산됩니다.

Amazon EC2는 5분마다 CloudWatch에 지표를 전송합니다. 어떤 시점에서든 지표의 이전 값을 참조하여 5분 전에 전송된 지표의 이전 값을 알 수 있습니다.

스탠다드에서 CPU 크레딧 사용량 계산

- CPU 사용률이 기준 미만이고 소비된 크레딧이 5분 전에 획득한 크레딧보다 적을 때 CPU 크레딧 밸런스가 증가합니다.
- CPU 사용률이 기준 이상이고 소비된 크레딧이 5분 전에 획득한 크레딧보다 많을 때 CPU 크레딧 밸런스가 감소합니다.

수학적으로 다음 수식을 통해 이를 확인할 수 있습니다:

Example

```
CPUCreditBalance = prior CPUCreditBalance + [Credits earned per hour * (5/60) - CPUCreditUsage]
```

인스턴스 크기에 따라 인스턴스가 시간당 획득할 수 있는 크레딧 수와 크레딧 밸런스에 누적할 수 있는 획득 크레딧의 수가 결정됩니다. 시간당 획득 크레딧 수, 각 인스턴스 크기에 대한 크레딧 밸런스 한도에 대한 자세한 내용은 [크레딧 표 \(p. 131\)](#)를 참조하십시오.

예

이 예제에서는 `t3.nano` 인스턴스를 사용합니다. 인스턴스의 `CPUCreditBalance` 값을 계산하려면 앞의 수식을 사용하여 다음과 같이 합니다.

- `CPUCreditBalance` – 계산하려는 현재 크레딧 밸런스입니다.
- `prior CPUCreditBalance` – 5분 전의 크레딧 밸런스입니다. 이 예제에서는 인스턴스가 2개의 크레딧을 획득했습니다.
- `Credits earned per hour` – `t3.nano` 인스턴스는 시간당 6개의 크레딧을 획득합니다.
- `5/60` – CloudWatch 지표가 게시되는 5분 간격을 나타냅니다. 시간당 획득한 크레딧에 $5/60(5\text{분})$ 을 곱해 인스턴스가 이전 5분 동안 획득한 크레딧 수를 계산합니다. `t3.nano` 인스턴스는 5분마다 0.5개 크레딧을 획득합니다.
- `CPUCreditUsage` – 이전 5분 동안 소비된 인스턴스의 크레딧 수입니다. 이 예제에서는 인스턴스가 이전 5분 동안 크레딧 1개를 소비했습니다.

이러한 값을 사용하여 CPUCreditBalance 값을 계산할 수 있습니다.

Example

```
CPUCreditBalance = 2 + [0.5 - 1] = 1.5
```

무제한 인스턴스에 대한 CPU 크레딧 사용량 계산

버스트 가능한 성능 인스턴스가 기준 성능 이상으로 버스트해야 할 때는 잉여 크레딧을 소비하기 전에 항상 누적 크레딧을 소비합니다. 획득한 CPU 크레딧 밸런스가 감소하면 필요한 시간만큼 잉여 크레딧을 소비하여 CPU를 버스트할 수 있습니다. 인스턴스의 CPU 사용률이 기준 미만으로 떨어지면 인스턴스가 크레딧을 획득하기 전에 잉여 크레딧이 항상 먼저 청산됩니다.

5분 간격으로 발생하는 활동을 반영하기 위해 다음 수식에서 Adjusted balance라는 용어를 사용하고 있습니다. CPUCreditBalance 및 CPUSurplusCreditBalance CloudWatch 지표의 값에 도달하기 위해 이 값을 사용합니다.

Example

```
Adjusted balance = [prior CPUCreditBalance - prior CPUSurplusCreditBalance] + [Credits earned per hour * (5/60) - CPUCreditUsage]
```

0의 값이 Adjusted balance이면 인스턴스는 버스트에 획득한 모든 크레딧을 소비했으며 잉여 크레딧은 소비되지 않았다는 뜻입니다. 그 결과 CPUCreditBalance와 CPUSurplusCreditBalance가 모두 0으로 설정됩니다.

Adjusted balance 값이 양수이면 인스턴스가 크레딧을 획득했고 이전의 잉여 크레딧(존재할 경우)이 청산되었다는 뜻입니다. 그 결과 Adjusted balance 값이 CPUCreditBalance로 지정되고 CPUSurplusCreditBalance가 0으로 설정됩니다. 누적할 수 있는 [최대 크레딧 수 \(p. 131\)](#)는 인스턴스 크기에 따라 결정됩니다.

Example

```
CPUCreditBalance = min [max earned credit balance, Adjusted balance]  
CPUSurplusCreditBalance = 0
```

Adjusted balance 값이 음수면 인스턴스가 누적한 모든 크레딧을 소비했고 버스트에 잉여 크레딧이 소비되었다는 뜻입니다. 그 결과 Adjusted balance 값이 CPUSurplusCreditBalance로 지정되고 CPUCreditBalance가 0으로 설정됩니다. 즉 누적할 수 있는 [최대 크레딧 수 \(p. 131\)](#)는 인스턴스 크기에 따라 결정됩니다.

Example

```
CPUSurplusCreditBalance = min [max earned credit balance, -Adjusted balance]  
CPUCreditBalance = 0
```

소비된 잉여 크레딧이 인스턴스가 누적할 수 있는 최대 크레딧을 초과하면 이전 수식에서와 같이 잉여 크레딧 밸런스가 최대 값으로 설정됩니다. 나머지 잉여 크레딧은 CPUSurplusCreditsCharged 측정치로 표현되어 요금이 부과됩니다.

Example

```
CPUSurplusCreditsCharged = max [-Adjusted balance - max earned credit balance, 0]
```

마지막으로 인스턴스가 종료하면 CPUSurplusCreditBalance로 추적된 모든 잉여 크레딧에 요금이 부과됩니다. 인스턴스가 unlimited에서 standard로 전환되면 나머지 모든 CPUSurplusCreditBalance에도 요금이 부과됩니다.

컴퓨팅 최적화 인스턴스

컴퓨팅 최적화 인스턴스는 고성능 프로세서의 이점을 활용하는 컴퓨팅 집약적 애플리케이션에 적합합니다.

C5 및 C5n 인스턴스

이 인스턴스는 다음과 같은 경우에 적합합니다.

- 일괄 처리 작업
- 미디어 트랜스코딩
- 고성능 웹 서버
- 고성능 컴퓨팅(HPC)
- 과학 모델링
- 전용 게임 서버, 광고 서비스 엔진
- 기계 학습 추론 및 기타 컴퓨팅 집약적 애플리케이션

베어 메탈 인스턴스(예: c5.meta1)에서는 애플리케이션이 프로세서, 메모리 등 호스트 서버의 물리적 리소스에 직접 액세스할 수 있습니다.

자세한 내용은 [Amazon EC2 C5 인스턴스](#)를 참조하십시오.

목차

- 하드웨어 사양 (p. 163)
- 인스턴스 성능 (p. 165)
- 네트워크 성능 (p. 165)
- SSD I/O 성능 (p. 166)
- 인스턴스 기능 (p. 167)
- 릴리스 정보 (p. 167)

하드웨어 사양

다음은 컴퓨팅 최적화 인스턴스용 하드웨어 사양을 요약한 것입니다.

인스턴스 유형	기본 vCPU	메모리(GiB)
c4.large	2	3.75
c4.xlarge	4	7.5
c4.2xlarge	8	15
c4.4xlarge	16	30
c4.8xlarge	36	60
c5.large	2	4
c5.xlarge	4	8
c5.2xlarge	8	16
c5.4xlarge	16	32
c5.9xlarge	36	72

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
컴퓨팅 최적화

인스턴스 유형	기본 vCPU	메모리(GiB)
c5.12xlarge	48	96
c5.18xlarge	72	144
c5.24xlarge	96	192
c5.metal	96	192
c5a.large	2	4
c5a.xlarge	4	8
c5a.2xlarge	8	16
c5a.4xlarge	16	32
c5a.8xlarge	32	64
c5a.12xlarge	48	96
c5a.16xlarge	64	128
c5a.24xlarge	96	192
c5ad.large	2	4
c5ad.xlarge	4	8
c5ad.2xlarge	8	16
c5ad.4xlarge	16	32
c5ad.8xlarge	32	64
c5ad.12xlarge	48	96
c5ad.16xlarge	64	128
c5ad.24xlarge	96	192
c5d.large	2	4
c5d.xlarge	4	8
c5d.2xlarge	8	16
c5d.4xlarge	16	32
c5d.9xlarge	36	72
c5d.12xlarge	48	96
c5d.18xlarge	72	144
c5d.24xlarge	96	192
c5d.metal	96	192
c5n.large	2	5.25
c5n.xlarge	4	10.5

인스턴스 유형	기본 vCPU	메모리(GiB)
c5n.2xlarge	8	21
c5n.4xlarge	16	42
c5n.9xlarge	36	96
c5n.18xlarge	72	192
c5n.metal	72	192

Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

CPU 옵션 지정에 대한 자세한 내용은 [CPU 옵션 최적화 \(p. 541\)](#) 단원을 참조하십시오.

인스턴스 성능

EBS에 최적화된 인스턴스를 사용하면 Amazon EBS I/O와 인스턴스의 다른 네트워크 간의 경합을 제거하여 EBS 볼륨에 대해 일관되게 우수한 성능을 제공할 수 있습니다. 일부 컴퓨팅 최적화 인스턴스는 추가 비용 없이도 EBS에 최적화할 수 있게 기본 설정되어 있습니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

네트워크 성능

지원되는 인스턴스 유형에 대해 향상된 네트워킹을 활성화하면 지역 시간을 줄이고 네트워크 지터를 낮추며 PPS(Packet Per Second) 성능을 높일 수 있습니다. 대부분의 애플리케이션은 항상 높은 수준의 네트워크 성능을 필요로 하지 않지만, 데이터를 주고 받을 때 증가된 대역폭에 액세스할 수 있을 경우 유익할 수 있습니다. 자세한 내용은 [Windows에서 향상된 네트워킹 \(p. 751\)](#) 단원을 참조하십시오.

다음은 향상된 네트워킹을 지원하는 컴퓨팅 최적화 인스턴스용 네트워크 성능을 요약한 것입니다.

인스턴스 유형	네트워크 성능	향상된 네트워킹
c5.4xlarge 이하 c5d.4xlarge 이하	최대 10Gbps †	ENA (p. 751)
c5.9xlarge c5d.9xlarge	10Gbps	ENA (p. 751)
c5.12xlarge c5d.12xlarge	12Gbps	ENA (p. 751)
c5n.4xlarge 이하	최대 25Gbps †	ENA (p. 751)
c5.18xlarge c5.24xlarge c5.metal c5d.18xlarge c5d.24xlarge c5d.metal	25Gbps	ENA (p. 751)
c5n.9xlarge	50Gbps	ENA (p. 751)
c5n.18xlarge c5n.metal	100Gbps	ENA (p. 751)
c4.large	보통	intel 82599 VF (p. 758)
c4.xlarge c4.2xlarge c4.4xlarge	높음	intel 82599 VF (p. 758)
c4.8xlarge	10Gbps	Intel 82599 VF (p. 758)

† 이러한 인스턴스는 네트워크 I/O 크레딧 메커니즘을 이용해 평균 대역폭 활용도를 기준으로 인스턴스에 네트워크 대역폭을 할당합니다. 해당 대역폭이 기준 대역폭 미만으로 떨어지면 크레딧이 발생하는데, 이 크레딧은 네트워크 데이터를 전송할 때 사용할 수 있습니다. 자세한 내용은 지원 사례를 개설하여 관심 있는 특정 인스턴스 유형의 기준 대역폭에 대해 문의하십시오.

SSD I/O 성능

를 사용하고 인스턴스에서 사용 가능한 모든 SSD 기반의 인스턴스 스토어 볼륨을 활용하는 경우, 다음 표와 같은 IOPS(블록 크기 4,096바이트) 성능을 얻을 수 있습니다(대기열 깊이 포화 상태에서). 그렇지 않으면 IOPS 성능이 더 낮아집니다.

인스턴스 크기	100% 임의 읽기 IOPS	IOPS 쓰기
c5ad.large	16,283	7,105
c5ad.xlarge	32,566	14,211
c5ad.2xlarge	65,132	28,421
c5ad.4xlarge	130,263	56,842
c5ad.8xlarge	260,526	113,684
c5ad.12xlarge	412,500	180,000
c5ad.16xlarge	521,053	227,368
c5ad.24xlarge	825,000	360,000
c5d.large *	20,000건	9,000
c5d.xlarge *	40,000	18,000
c5d.2xlarge *	80,000	37,000
c5d.4xlarge *	175,000	75,000
c5d.9xlarge	350,000	170,000
c5d.12xlarge	700,000	340,000
c5d.18xlarge	700,000	340,000
c5d.24xlarge	1,400,000	680,000
c5d.metal	1,400,000	680,000

* 이러한 인스턴스의 경우, 지정된 최대 성능을 얻을 수 있습니다.

인스턴스에 대한 SSD 기반 인스턴스 스토어 볼륨에 데이터가 있는 경우, 달성 가능한 쓰기 IOPS의 수는 감소합니다. 이는 SSD 컨트롤러가 가용 공간을 찾고 기존 데이터를 다시 쓰고 미사용 공간을 삭제하여 다시 쓸 수 있는 공간을 마련하기 위해 추가적인 작업을 해야 하기 때문입니다. 이러한 폐영역 회수 과정은 SSD에 대한 내부 쓰기 작업이 증폭되는 결과를 낳게 되며, 이런 결과는 사용자 쓰기 작업에 대한 SSD 쓰기 작업의 비로 표현됩니다. 이러한 성능 감소는 쓰기 작업이 4096바이트의 배수들 또는 4096바이트 경계에 정렬되지 않은 상태로 수행되는 경우에 더 심해질 수 있습니다. 정렬되지 않은 바이트를 소량으로 쓰기 작업하는 경우, SSD 컨트롤러는 쓰려는 부분의 주변 데이터를 읽고 그 결과도 새 위치에 저장해야 합니다. 이런 패턴으로 인해 쓰기 작업이 크게 증폭되고 지연 시간 증가와 I/O 성능의 급격한 감소를 초래합니다.

SSD 컨트롤러는 여러 전략을 사용해서 쓰기 작업 증폭의 영향을 감쇄할 수 있습니다. 그 중 하나의 전략은 SSD 인스턴스 스토리지에 예약 공간을 마련해서 SSD 컨트롤러가 쓰기 작업에 사용 가능한 공간을 보다 효

을적으로 관리할 수 있게 하는 것입니다. 이를 오버-프로비저닝이라고 합니다. 인스턴스에 제공된 SSD 기반 인스턴스 스토어 볼륨은 오버-프로비저닝을 위한 예약 공간을 가지고 있지 않습니다. 쓰기 작업 증폭의 영향 감쇄를 위해 최소한 볼륨의 10%를 파티션 처리되지 않은 상태로 두어서 SSD 컨트롤러가 이를 오버-프로비저닝에 사용할 수 있도록 하는 것이 좋습니다. 그러면 사용할 수 있는 스토리지는 줄어들지만, 디스크를 전체 용량에 가깝게 사용하더라도 성능은 향상됩니다.

TRIM을 지원하는 인스턴스 스토어 볼륨의 경우, TRIM 명령을 사용하여 작성한 데이터가 더 이상 필요하지 않음을 SSD 컨트롤러에 알릴 수 있습니다. 이를 통해 컨트롤러에 더 많은 여유 공간이 제공되어 쓰기 증폭이 줄어들고 성능이 향상될 수 있습니다. 자세한 내용은 [인스턴스 스토어 볼륨 TRIM 지원 \(p. 1102\)](#) 단원을 참조하십시오.

인스턴스 기능

컴퓨팅 최적화 인스턴스에 대한 기능은 다음과 같이 간략히 설명할 수 있습니다.

	EBS 전용	NVMe EBS	인스턴스 스토어	배치 그룹
C4	예	아니요	아니요	예
C5	예	예	아니요	예
C5a	예	예	아니요	예
C5ad	아니요	예	NVMe *	예
C5d	아니요	예	NVMe *	예
C5n	예	예	아니요	예

* 루트 디바이스 볼륨은 Amazon EBS 볼륨이어야 합니다.

자세한 내용은 다음 단원을 참조하십시오.

- [Windows 인스턴스의 Amazon EBS 및 NVMe \(p. 1048\)](#)
- [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#)
- [배치 그룹 \(p. 761\)](#)

릴리스 정보

- C5 및 C5d 인스턴스는 1세대(Skylake-SP) 또는 2세대(Cascade Lake)의 3.1 GHz Intel Xeon Platinum 8000 시리즈 프로세서를 사용합니다.
- C5a 및 C5ad 인스턴스는 3.3GHz의 높은 주파수로 실행되는 2세대 AMD EPYC 프로세서(Rome)를 사용합니다.
- C4 인스턴스 및 [Nitro 시스템 \(p. 118\)](#) 기반 인스턴스에는 64비트 EBS 지원 HVM AMIs가 필요합니다. 이들은 고용량 메모리를 보유하는데, 이 용량을 활용하기 위해서는 64비트 운영 체제가 필요합니다. HVM AMI는 고용량 메모리 인스턴스 유형의 반가상화(PV) AMI보다 우수한 성능을 제공합니다. 또한 확장 네트워킹을 활용하려면 HVM AMI를 사용해야 합니다.
- Nitro 시스템에 구축된 인스턴스에는 다음과 같은 요구 사항이 있습니다.
 - [NVMe 드라이버 \(p. 1048\)](#)가 설치되어 있어야 합니다.
 - [ENA\(Elastic Network Adapter\) \(p. 751\)](#) 드라이버가 설치되어 있어야 합니다.

현재 [AWS Windows AMI \(p. 26\)](#)는 이러한 요구 사항을 충족합니다.

- Nitro 시스템 인스턴스에 구축된 인스턴스는 네트워크 인터페이스, EBS 볼륨 및 NVMe 인스턴스 스토어 볼륨을 포함해 최대 28개의 연결을 지원합니다. 자세한 내용은 [Nitro 시스템 볼륨 제한 \(p. 1105\)](#) 단원을 참조하십시오.

- 베어 메탈 인스턴스를 시작하면 기본 서버가 부팅되는데, 이때 모든 하드웨어 및 펌웨어 구성 요소를 확인합니다. 즉, 인스턴스가 실행 상태가 되어 네트워크를 통해 사용할 수 있게 될 때까지 20분이 걸릴 수 있습니다.
- 베어 메탈 인스턴스에서 EBS 볼륨 또는 보조 네트워크 인터페이스를 연결 또는 분리하려면 PCIe 기본 핫 플러그 지원이 필요합니다.
- 베어 메탈 인스턴스에서는 I/O 포트 기반 직렬 디바이스가 아닌 PCI 기반 직렬 디바이스를 사용합니다. 업스트림 Linux 커널 및 최신 Amazon Linux AMI에서는 이 디바이스를 지원합니다. 베어 메탈 인스턴스도 시스템에서 PCI 기반 직렬 디바이스를 자동으로 사용할 수 있게 해주는 ACPI SPCR 테이블을 제공합니다. 최신 Windows AMI에서는 PCI 기반 직렬 디바이스를 자동으로 사용합니다.
- 한 리전에서 시작할 수 있는 총 인스턴스 수에는 제한이 있으며, 일부 인스턴스 유형에는 또 다른 제한이 있습니다. 자세한 내용은 Amazon EC2 FAQ의 [Amazon EC2에서 실행할 수 있는 인스턴스 수는 몇 개입니까?](#)를 참조하십시오.

메모리 최적화 인스턴스

메모리 최적화 인스턴스는 메모리에서 대규모 데이터를 처리하는 워크로드에 대해 빠른 성능을 제공하도록 설계되었습니다.

R5, R5a 및 R5n 인스턴스

이 인스턴스는 다음과 같은 경우에 적합합니다.

- 고성능, 관계형(MySQL) 및 NoSQL(MongoDB, Cassandra) 데이터베이스.
- 키-값 유형 데이터의 인 메모리 캐싱을 제공하는 분산된 웹 규모 캐시 저장소(Memcached 및 Redis).
- 비즈니스 intel리전스를 위해 최적화된 데이터 스토리지 형식과 분석을 사용하는 인 메모리 데이터베이스 (예: SAP HANA).
- 대용량 비정형 데이터를 실시간으로 처리하는 애플리케이션(금융 서비스, Hadoop/Spark 클러스터).
- HPC(고성능 컴퓨팅) 및 EDA(전자 설계 자동화) 애플리케이션.

베어 메탈 인스턴스(예: r5.meta1)에서는 애플리케이션이 프로세서, 메모리 등 호스트 서버의 물리적 리소스에 직접 액세스할 수 있습니다. 이 인스턴스는 다음과 같은 경우에 적합합니다.

- 가상화된 환경에서 사용할 수 없거나 완전히 지원되지 않는 저수준 하드웨어 기능(예: intel VT)에 액세스해야 하는 워크로드
- 라이선스 또는 지원을 위해 가상화되지 않은 환경이 필요한 애플리케이션

자세한 내용은 [Amazon EC2 R5 인스턴스](#)를 참조하십시오.

고용량 메모리 인스턴스

고용량 메모리 인스턴스(u-6tb1.meta1, u-9tb1.meta1, u-12tb1.meta1, u-18tb1.meta1 및 u-24tb1.meta1)는 인스턴스당 6TiB, 9TiB, 12TiB, 18TiB, 24TiB의 메모리를 제공합니다. 이러한 인스턴스는 SAP HANA 인 메모리 데이터베이스의 프로덕션 배포를 비롯하여 클라우드에서 대규모 인 메모리 데이터베이스를 실행하도록 설계되었습니다. 베어 메탈 성능에 호스트 하드웨어에 대한 직접 액세스를 제공합니다.

자세한 내용은 [Amazon EC2 고용량 메모리 인스턴스](#) 및 [SAP HANA용 스토리지 구성](#)을 참조하십시오.

X1 인스턴스

이 인스턴스는 다음과 같은 경우에 적합합니다.

- SAP HANA와 같은 인 메모리 데이터베이스[Business Suite S/4HANA, Business Suite on HANA(SoH), Business Warehouse on HANA(BW) 및 Data Mart Solutions on HANA에 대한 SAP 인증 지원 포함]. 자세한 내용은 [AWS 클라우드 기반의 SAP HANA](#)를 참조하십시오.
- Apache Spark나 Presto와 같은 빅데이터 처리 엔진.

- HPC(고성능 컴퓨팅) 애플리케이션.

자세한 내용은 [Amazon EC2 X1 인스턴스](#)를 참조하십시오.

X1e 인스턴스

이 인스턴스는 다음과 같은 경우에 적합합니다.

- 고성능 데이터베이스.
- SAP HANA와 같은 인 메모리 데이터베이스. 자세한 내용은 [AWS 클라우드 기반의 SAP HANA](#)를 참조하십시오.
- 메모리 집약적인 엔터프라이즈 애플리케이션.

자세한 내용은 [Amazon EC2 X1e 인스턴스](#)를 참조하십시오.

z1d 인스턴스

이러한 인스턴스는 컴퓨팅 용량과 메모리가 대형이며 다음의 경우 적합합니다.

- EDA(전자 설계 자동화)
- 관계형 데이터베이스 워크로드

`z1d.meta1` 인스턴스에서는 애플리케이션이 프로세서, 메모리 등 호스트 서버의 물리적 리소스에 직접 액세스할 수 있습니다. 이 인스턴스는 다음과 같은 경우에 적합합니다.

- 가상화된 환경에서 사용할 수 없거나 완전히 지원되지 않는 저수준 하드웨어 기능(예: intel VT)에 액세스해야 하는 워크로드
- 라이선스 또는 지원을 위해 가상화되지 않은 환경이 필요한 애플리케이션

자세한 내용은 [Amazon EC2 z1d 인스턴스](#)를 참조하십시오.

목차

- [하드웨어 사양 \(p. 169\)](#)
- [메모리 성능 \(p. 172\)](#)
- [인스턴스 성능 \(p. 172\)](#)
- [네트워크 성능 \(p. 172\)](#)
- [SSD I/O 성능 \(p. 173\)](#)
- [인스턴스 기능 \(p. 175\)](#)
- [높은 가용성 및 안정성\(X1\) \(p. 176\)](#)
- [개의 vCPU 지원 \(p. 176\)](#)
- [릴리스 정보 \(p. 176\)](#)

하드웨어 사양

다음은 메모리 최적화 인스턴스용 하드웨어 사양의 요약 설명입니다.

인스턴스 유형	기본 vCPU	메모리(GiB)
r4.large	2	15.25
r4.xlarge	4	30.5

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
메모리 최적화

인스턴스 유형	기본 vCPU	메모리(GiB)
r4.2xlarge	8	61
r4.4xlarge	16	122
r4.8xlarge	32	244
r4.16xlarge	64	488
r5.large	2	16
r5.xlarge	4	32
r5.2xlarge	8	64
r5.4xlarge	16	128
r5.8xlarge	32	256
r5.12xlarge	48	384
r5.16xlarge	64	512
r5.24xlarge	96	768
r5.metal	96	768
r5a.large	2	16
r5a.xlarge	4	32
r5a.2xlarge	8	64
r5a.4xlarge	16	128
r5a.8xlarge	32	256
r5a.12xlarge	48	384
r5a.16xlarge	64	512
r5a.24xlarge	96	768
r5ad.large	2	16
r5ad.xlarge	4	32
r5ad.2xlarge	8	64
r5ad.4xlarge	16	128
r5ad.8xlarge	32	256
r5ad.12xlarge	48	384
r5ad.16xlarge	64	512
r5ad.24xlarge	96	768
r5d.large	2	16
r5d.xlarge	4	32

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
메모리 최적화

인스턴스 유형	기본 vCPU	메모리(GiB)
r5d.2xlarge	8	64
r5d.4xlarge	16	128
r5d.8xlarge	32	256
r5d.12xlarge	48	384
r5d.16xlarge	64	512
r5d.24xlarge	96	768
r5d.metal	96	768
r5dn.large	2	16
r5dn.xlarge	4	32
r5dn.2xlarge	8	64
r5dn.4xlarge	16	128
r5dn.8xlarge	32	256
r5dn.12xlarge	48	384
r5dn.16xlarge	64	512
r5dn.24xlarge	96	768
r5n.large	2	16
r5n.xlarge	4	32
r5n.2xlarge	8	64
r5n.4xlarge	16	128
r5n.8xlarge	32	256
r5n.12xlarge	48	384
r5n.16xlarge	64	512
r5n.24xlarge	96	768
u-6tb1.metal	448 *	6,144
u-9tb1.metal	448 *	9,216
u-12tb1.metal	448 *	12,288
u-18tb1.metal	448 *	18,432
u-24tb1.metal	448 *	24,576
x1.16xlarge	64	976
x1.32xlarge	128	1,952
x1e.xlarge	4	122

인스턴스 유형	기본 vCPU	메모리(GiB)
x1e.2xlarge	8	244
x1e.4xlarge	16	488
x1e.8xlarge	32	976
x1e.16xlarge	64	1,952
x1e.32xlarge	128	3,904
z1d.large	2	16
z1d.xlarge	4	32
z1d.2xlarge	8	64
z1d.3xlarge	12	96
z1d.6xlarge	24	192
z1d.12xlarge	48	384
z1d.metal	48	384

* 각 논리 프로세서는 224개 코더의 하이퍼스레드입니다.

Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

CPU 옵션 지정에 대한 자세한 내용은 [CPU 옵션 최적화 \(p. 541\)](#) 단원을 참조하십시오.

메모리 성능

X1 인스턴스에는 intel 확장형 메모리 버퍼가 포함되어 있어, 300GiB/s의 지속 가능 메모리 읽기 대역폭과 140GiB/s의 지속 가능 메모리 쓰기 대역폭을 제공합니다.

메모리 최적화 인스턴스에 사용할 수 있는 RAM 크기에 대한 자세한 내용은 [하드웨어 사양 \(p. 169\)](#) 단원을 참조하십시오.

메모리 최적화 인스턴스는 고용량 메모리를 보유하며, 이 용량을 활용하기 위해 64비트 HVM AMI가 필요합니다. HVM AMI는 메모리 최적화 인스턴스의 반가상화(PV) AMI보다 우수한 성능을 제공합니다. 단원을 참조하십시오.

인스턴스 성능

메모리 최적화 인스턴스는 최신 intel AES-NI 기능을 통해 암호화 성능을 끌어올릴 수 있고, intel TSX(Transactional Synchronization Extensions)를 지원하여 인 메모리 트랜잭션 데이터 처리의 성능을 강화하며, intel AVX2(Advanced Vector Extensions 2) 프로세서 지원을 지원하여 대부분의 정수 명령을 256비트로 확장합니다.

네트워크 성능

지원되는 인스턴스 유형에 대해 향상된 네트워킹을 활성화하면 지연 시간을 줄이고 네트워크 지터를 낮추며 PPS(Packet Per Second) 성능을 높일 수 있습니다. 대부분의 애플리케이션은 항상 높은 수준의 네트워크 성능을 필요로 하지 않지만, 데이터를 주고 받을 때 증가된 대역폭에 액세스할 수 있을 경우 유익할 수 있습니다. 자세한 내용은 [Windows에서 향상된 네트워킹 \(p. 751\)](#) 단원을 참조하십시오.

다음은 향상된 네트워킹을 지원하는 메모리 최적화 인스턴스용 네트워크 성능의 요약 설명입니다.

인스턴스 유형	네트워크 성능	향상된 네트워킹
r4.4xlarge 이하 r5.4xlarge 이하 r5a.8xlarge 이하 r5ad.8xlarge 이하 r5d.4xlarge 이하 x1e.8large 이하 z1d.3xlarge 이하	최대 10Gbps †	ENA (p. 751)
r4.8xlarge r5.8xlarge r5.12xlarge r5a.12xlarge r5ad.12xlarge r5d.8xlarge r5d.12xlarge x1.16xlarge x1e.16xlarge z1d.6xlarge	10Gbps	ENA (p. 751)
r5a.16xlarge r5ad.16xlarge	12Gbps	ENA (p. 751)
r5.16xlarge r5a.24xlarge r5ad.24xlarge r5d.16xlarge	20Gbps	ENA (p. 751)
r5dn.4xlarge 이하 r5n.4xlarge 이하	최대 25Gbps †	ENA (p. 751)
r4.16xlarge r5.24xlarge r5.metal r5d.24xlarge r5d.metal r5dn.8xlarge r5n.8xlarge x1.32xlarge x1e.32xlarge z1d.12xlarge z1d.metal	25Gbps	ENA (p. 751)
r5dn.12xlarge r5n.12xlarge	50Gbps	ENA (p. 751)
r5dn.16xlarge r5n.16xlarge	75Gbps	ENA (p. 751)
r5dn.24xlarge r5n.24xlarge u-6tb1.metal * u-9tb1.metal * u-12tb1.metal * u-18tb1.metal u-24tb1.metal	100Gbps	ENA (p. 751)

* 2020년 3월 12일 이후에 시작된 이 유형의 인스턴스는 100Gbps의 네트워크 성능을 제공합니다. 2020년 3월 12일 이전에 시작된 이 유형의 인스턴스는 25Gbps의 네트워크 성능만 제공할 수 있습니다. 2020년 3월 12일 이전에 시작된 인스턴스에서 100Gbps의 네트워크 성능을 제공하도록 하려면 계정 팀에 문의하여 추가 비용 없이 인스턴스를 업그레이드하십시오.

† 이러한 인스턴스는 네트워크 I/O 크레딧 메커니즘을 이용해 평균 대역폭 활용도를 기준으로 인스턴스에 네트워크 대역폭을 할당합니다. 해당 대역폭이 기준 대역폭 미만으로 떨어지면 크레딧이 발생하는데, 이 크레딧은 네트워크 데이터를 전송할 때 사용할 수 있습니다. 자세한 내용은 지원 사례를 개설하여 관심 있는 특정 인스턴스 유형의 기준 대역폭에 대해 문의하십시오.

SSD I/O 성능

를 사용하고 인스턴스에서 사용 가능한 모든 SSD 기반의 인스턴스 스토어 볼륨을 활용하는 경우, 다음 표와 같은 IOPS(블록 크기 4,096바이트) 성능을 얻을 수 있습니다(대기열 깊이 포화 상태에서). 그렇지 않으면 IOPS 성능이 더 낮아집니다.

인스턴스 크기	100% 임의 읽기 IOPS	IOPS 쓰기
r5ad.large *	30,000개	15,000
r5ad.xlarge *	59,000	29,000
r5ad.2xlarge *	117,000	57,000
r5ad.4xlarge *	234,000	114,000

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
메모리 최적화

인스턴스 크기	100% 임의 읽기 IOPS	IOPS 쓰기
r5ad.8xlarge	466,666	233,333
r5ad.12xlarge	700,000	340,000
r5ad.16xlarge	933,333	466,666
r5ad.24xlarge	1,400,000	680,000
r5d.large *	30,000개	15,000
r5d.xlarge *	59,000	29,000
r5d.2xlarge *	117,000	57,000
r5d.4xlarge *	234,000	114,000
r5d.8xlarge	466,666	233,333
r5d.12xlarge	700,000	340,000
r5d.16xlarge	933,333	466,666
r5d.24xlarge	1,400,000	680,000
r5d.metal	1,400,000	680,000
r5dn.large *	30,000개	15,000
r5dn.xlarge *	59,000	29,000
r5dn.2xlarge *	117,000	57,000
r5dn.4xlarge *	234,000	114,000
r5dn.8xlarge	466,666	233,333
r5dn.12xlarge	700,000	340,000
r5dn.16xlarge	933,333	466,666
r5dn.24xlarge	1,400,000	680,000
z1d.large *	30,000개	15,000
z1d.xlarge *	59,000	29,000
z1d.2xlarge *	117,000	57,000
z1d.3xlarge *	175,000	75,000
z1d.6xlarge	350,000	170,000
z1d.12xlarge	700,000	340,000
z1d.metal	700,000	340,000

* 이러한 인스턴스의 경우, 지정된 최대 성능을 얻을 수 있습니다.

인스턴스에 대한 SSD 기반 인스턴스 스토어 볼륨에 데이터가 있는 경우, 달성 가능한 쓰기 IOPS의 수는 감소합니다. 이는 SSD 컨트롤러가 가용 공간을 찾고 기존 데이터를 다시 쓰고 미사용 공간을 삭제하여 다시 쓸

수 있는 공간을 마련하기 위해 추가적인 작업을 해야 하기 때문입니다. 이러한 폐영역 회수 과정은 SSD에 대한 내부 쓰기 작업이 증폭되는 결과를 낳게 되며, 이런 결과는 사용자 쓰기 작업에 대한 SSD 쓰기 작업의 비로 표현됩니다. 이러한 성능 감소는 쓰기 작업이 4096바이트의 배수들 또는 4096바이트 경계에 정렬되지 않은 상태로 수행되는 경우에 더 심해질 수 있습니다. 정렬되지 않은 바이트를 소량으로 쓰기 작업하는 경우, SSD 컨트롤러는 쓰려는 부분의 주변 데이터를 읽고 그 결과도 새 위치에 저장해야 합니다. 이런 패턴으로 인해 쓰기 작업이 크게 증폭되고 지연 시간 증가와 I/O 성능의 급격한 감소를 초래합니다.

SSD 컨트롤러는 여러 전략을 사용해서 쓰기 작업 증폭의 영향을 감쇄할 수 있습니다. 그 중 하나의 전력은 SSD 인스턴스 스토리지에 예약 공간을 마련해서 SSD 컨트롤러가 쓰기 작업에 사용 가능한 공간을 보다 효율적으로 관리할 수 있게 하는 것입니다. 이를 오버-프로비저닝이라고 합니다. 인스턴스에 제공된 SSD 기반 인스턴스 스토어 볼륨은 오버-프로비저닝을 위한 예약 공간을 가지고 있지 않습니다. 쓰기 작업 증폭의 영향 감쇄를 위해 최소한 볼륨의 10%를 파티션 처리되지 않은 상태로 두어서 SSD 컨트롤러가 이를 오버-프로비저닝에 사용할 수 있도록 하는 것이 좋습니다. 그러면 사용할 수 있는 스토리지는 줄어들지만, 디스크를 전체 용량에 가깝게 사용하더라도 성능은 향상됩니다.

TRIM을 지원하는 인스턴스 스토어 볼륨의 경우, TRIM 명령을 사용하여 작성한 데이터가 더 이상 필요하지 않음을 SSD 컨트롤러에 알릴 수 있습니다. 이를 통해 컨트롤러에 더 많은 여유 공간이 제공되어 쓰기 증폭이 줄어들고 성능이 향상될 수 있습니다. 자세한 내용은 [인스턴스 스토어 볼륨 TRIM 지원 \(p. 1102\)](#) 단원을 참조하십시오.

인스턴스 기능

메모리 최적화 인스턴스에 대한 기능은 다음과 같이 간략히 설명할 수 있습니다.

	EBS 전용	NVMe EBS	인스턴스 스토어	배치 그룹
R4	예	아니요	아니요	예
R5	예	예	아니요	예
R5a	예	예	아니요	예
R5ad	아니요	예	NVME *	예
R5d	아니요	예	NVME *	예
R5dn	아니요	예	NVME *	예
R5n	예	예	아니요	예
u-6tb1.metal	예	아니요	아니요	아니요
u-9tb1.metal	예	아니요	아니요	아니요
u-12tb1.metal	예	아니요	아니요	아니요
u-18tb1.metal	예	아니요	아니요	아니요
u-24tb1.metal	예	아니요	아니요	아니요
X1	아니요	아니요	SSD	예
X1e	아니요	아니요	SSD *	예
z1d	아니요	예	NVME *	예

* 루트 디바이스 볼륨은 Amazon EBS 볼륨이어야 합니다.

자세한 내용은 다음 단원을 참조하십시오.

- Windows 인스턴스의 Amazon EBS 및 NVMe (p. 1048)
- Amazon EC2 인스턴스 스토어 (p. 1091)
- 배치 그룹 (p. 761)

높은 가용성 및 안정성(X1)

X1 인스턴스는 멀티비트 오류를 발견하여 교정하는 Single Device Data Correction(SDDC +1)을 지원합니다. SDDC +1은 오류 검사 및 교정 코드를 사용하여 장애가 있는 단일 DRAM 디바이스를 찾아내 비활성화합니다.

뿐만 아니라 고가용성(HA) 및 재해 복구(DR) 솔루션을 실행하여 [Amazon CloudFormation](#) 및 [인스턴스 복구 \(p. 460\)](#)를 최대한 활용함으로써 목표 복구 시점(RPO), 목표 복구 시간(RTO) 및 비용 요건을 충족할 수 있습니다.

SAP HANA 프로덕션 환경을 사용하는 경우, X1 인스턴스에서 HANA System Replication(HSR)을 사용하는 옵션도 있습니다. X1 인스턴스에서 HA 및 DR 솔루션을 설계하는 것에 대한 자세한 정보는 [Amazon Web Services 클라우드 기반 SAP HANA: 쿼 스타트 레퍼런스 배포](#) 문서를 참조하십시오.

개의 vCPU 지원

메모리 최적화 인스턴스는 다수의 vCPU를 지원하므로 vCPU 수가 제한된 운영 체제에서 시작 문제가 발생할 수 있습니다. 따라서 메모리 최적화 인스턴스를 시작할 때 최신 AMI를 사용하실 것을 적극 권장합니다.

다음은 메모리 최적화 인스턴스 시작을 지원하는 AMI입니다.

- Amazon Linux 2(HVM)
- Amazon Linux AMI 2016.03(HVM) 이상
- Ubuntu Server 14.04 LTS(HVM)
- Red Hat Enterprise Linux 7.1(HVM)
- SUSE Linux Enterprise Server 12 SP1(HVM)
- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 R2 64비트
- Windows Server 2008 SP2 64비트

릴리스 정보

- R4 인스턴스는 최대 64개의 vCPU를 사용할 수 있는 것이 특징이며, 인 메모리 애플리케이션의 성능을 강화하기 위해 고용량 메모리 대역폭과 대용량 L3 캐시가 특징인 E5-2686v4 기반의 AWS 맞춤형 intel Xeon 프로세서 2개로 작동됩니다.
- R5 및 R5d 인스턴스는 1세대(Skylake-SP) 또는 2세대(Cascade Lake)의 3.1 GHz Intel Xeon Platinum 8000 시리즈 프로세서를 사용합니다.
- R5a 및 R5ad 인스턴스는 2.5GHz AMD EPYC 7000 시리즈 프로세서를 사용하는 것이 특징입니다.
- 고용량 메모리 인스턴스(u-6tb1.metal, u-9tb1.metal, u-12tb1.metal)는 미션 크리티컬 엔터프라이즈 워크로드에 최적화된 intel Xeon Platinum 8176M(Skylake) 프로세서로 8소켓 플랫폼에서 작동하는 최초의 인스턴스입니다. 18TB 및 24TB 메모리가 제공되는 대용량 메모리 인스턴스(u-18tb1.metal, u-24tb1.metal)는 2세대 인텔 제온 확장형 8280L(Cascade Lake) 프로세서가 탑재된 8소켓 플랫폼에서 구동되는 최초의 Amazon 2nd 인스턴스입니다.

- X1e 및 X1 인스턴스는 최대 128개의 vCPU를 사용할 수 있는 것이 특징이며, 인메모리 애플리케이션의 성능을 강화하기 위해 높은 메모리 대역폭과 대용량 L3 캐시를 채용한 intel Xeon E7-8880 v3 프로세서 4개로 작동됩니다.
- Nitro 시스템에 구축된 인스턴스에는 다음과 같은 요구 사항이 있습니다.
 - NVMe 드라이버 (p. 1048)가 설치되어 있어야 합니다.
 - ENA(Elastic Network Adapter) (p. 751) 드라이버가 설치되어 있어야 합니다.
- 현재 [AWS Windows AMI \(p. 26\)](#)는 이러한 요구 사항을 충족합니다.
- Nitro 시스템 인스턴스에 구축된 인스턴스는 네트워크 인터페이스, EBS 볼륨 및 NVMe 인스턴스 스토어 볼륨을 포함해 최대 28개의 연결을 지원합니다. 자세한 내용은 [Nitro 시스템 볼륨 제한 \(p. 1105\)](#) 단원을 참조하십시오.
- 베어 메탈 인스턴스를 시작하면 기본 서버가 부팅되는데, 이때 모든 하드웨어 및 펌웨어 구성 요소를 확인합니다. 즉, 인스턴스가 실행 상태가 되어 네트워크를 통해 사용할 수 있게 될 때까지 20분이 걸릴 수 있습니다.
- 베어 메탈 인스턴스에서 EBS 볼륨 또는 보조 네트워크 인터페이스를 연결 또는 분리하려면 PCIe 기본 핫 플러그 지원이 필요합니다.
- 베어 메탈 인스턴스에서는 I/O 포트 기반 직렬 디바이스가 아닌 PCI 기반 직렬 디바이스를 사용합니다. 업스트림 Linux 커널 및 최신 Amazon Linux AMI에서는 이 디바이스를 지원합니다. 베어 메탈 인스턴스도 시스템에서 PCI 기반 직렬 디바이스를 자동으로 사용할 수 있게 해주는 ACPI SPCR 테이블을 제공합니다. 최신 Windows AMI에서는 PCI 기반 직렬 디바이스를 자동으로 사용합니다.
- x1.16xlarge 인스턴스를 제외하고는 Windows Server 2008 SP2 64비트 AMI를 사용하여 X1 인스턴스를 시작할 수 없습니다.
- Windows Server 2008 SP2 64비트 AMI를 사용하여 X1e 인스턴스를 시작할 수 없습니다.
- Windows Server 2008 R2 64비트 AMI의 구 버전에서는 r4.1.large 및 r4.4.xlarge 인스턴스를 시작할 수 없습니다. 이 문제가 발생하면 이 AMI의 최신 버전으로 업데이트합니다.
- 한 리전에서 시작할 수 있는 총 인스턴스 수에는 제한이 있으며, 일부 인스턴스 유형에는 또 다른 제한이 있습니다. 자세한 내용은 Amazon EC2 FAQ의 [Amazon EC2에서 실행할 수 있는 인스턴스 수는 몇 개입니까?](#)를 참조하십시오.

스토리지 최적화 인스턴스

스토리지 최적화 인스턴스는 로컬 스토리지의 초대형 데이터 세트에 대한 순차적 읽기 및 쓰기 액세스가 많이 필요한 작업에 적합하도록 설계되었습니다. 낮은 지연 시간의 임의의 IOPS(초당 I/O 작업)를 만 단위 수준으로 애플리케이션에 제공할 수 있도록 최적화되어 있습니다.

D2 인스턴스

이 인스턴스는 다음과 같은 경우에 적합합니다.

- 대량 병렬 처리(MPP) 데이터 웨어하우스
- MapReduce 및 Hadoop 분산 컴퓨팅
- 로그 또는 데이터 처리 애플리케이션

H1 인스턴스

이 인스턴스는 다음과 같은 경우에 적합합니다.

- MapReduce 및 분산형 파일 시스템 같은 데이터 집약적 워크로드
- 직접 연결 인스턴스 스토리지에서 대량 데이터에 순차적으로 액세스해야 하는 애플리케이션
- 대량의 데이터에 대해 고속 액세스가 필요한 애플리케이션

I3 및 I3en 인스턴스

이 인스턴스는 다음과 같은 경우에 적합합니다.

- 빈도가 높은 온라인 트랜잭션 처리(OLTP) 시스템
- 관계형 데이터베이스
- NoSQL 데이터베이스
- 인 메모리 데이터베이스의 캐시(예: Redis)
- 데이터 웨어하우징 애플리케이션
- 분산 파일 시스템

베어 메탈 인스턴스에서는 애플리케이션이 프로세서, 메모리 등 호스트 서버의 물리적 리소스에 직접 액세스 할 수 있습니다. 이 인스턴스는 다음과 같은 경우에 적합합니다.

- 가상화된 환경에서 사용할 수 없거나 완전히 지원되지 않는 저수준 하드웨어 기능(예: intel VT)에 액세스 해야 하는 워크로드
- 라이선스 또는 지원을 위해 가상화되지 않은 환경이 필요한 애플리케이션

자세한 내용은 [Amazon EC2 I3 인스턴스](#)를 참조하십시오.

내용

- [하드웨어 사양 \(p. 178\)](#)
- [인스턴스 성능 \(p. 179\)](#)
- [네트워크 성능 \(p. 179\)](#)
- [SSD I/O 성능 \(p. 180\)](#)
- [인스턴스 기능 \(p. 181\)](#)
- [릴리스 정보 \(p. 182\)](#)

하드웨어 사양

D2 인스턴스의 기본 데이터 스토리지는 HDD 인스턴스 스토어 볼륨입니다. I3 및 I3en 인스턴스의 기본 데이터 스토리지는 NVMe(Non-Volatile Memory Express) SSD 인스턴스 스토어 볼륨입니다.

인스턴스 스토어 볼륨은 인스턴스의 수명 기간 동안만 유지됩니다. 인스턴스가 중지되거나 종료되면 인스턴스 스토어 볼륨의 애플리케이션과 데이터는 삭제됩니다. 따라서 정기적으로 인스턴스 스토어 볼륨에 중요한 데이터를 백업 또는 복제하는 것이 좋습니다. 자세한 내용은 [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#) 및 [SSD 인스턴스 스토어 볼륨 \(p. 1101\)](#) 단원을 참조하십시오.

다음은 스토리지 최적화 인스턴스용 하드웨어 사양의 요약 설명입니다.

인스턴스 유형	기본 vCPU	메모리(GiB)
d2.xlarge	4	30.5
d2.2xlarge	8	61
d2.4xlarge	16	122
d2.8xlarge	36	244
h1.2xlarge	8	32

인스턴스 유형	기본 vCPU	메모리(GiB)
h1.4xlarge	16	64
h1.8xlarge	32	128
h1.16xlarge	64	256
i3.large	2	15.25
i3.xlarge	4	30.5
i3.2xlarge	8	61
i3.4xlarge	16	122
i3.8xlarge	32	244
i3.16xlarge	64	488
i3.metal	72	512
i3en.large	2	16
i3en.xlarge	4	32
i3en.2xlarge	8	64
i3en.3xlarge	12	96
i3en.6xlarge	24	192
i3en.12xlarge	48	384
i3en.24xlarge	96	768
i3en.metal	96	768

Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

CPU 옵션 지정에 대한 자세한 내용은 [CPU 옵션 최적화 \(p. 541\)](#) 단원을 참조하십시오.

인스턴스 성능

NVMe 인스턴스 스토어 볼륨을 포함하는 인스턴스의 경우 AWS NVMe 드라이버를 사용해야 합니다. 자세한 내용은 [Windows 인스턴스의 AWS NVMe 드라이버 \(p. 539\)](#) 단원을 참조하십시오.

EBS에 최적화된 인스턴스를 사용하면 Amazon EBS I/O와 인스턴스의 다른 네트워크 간의 경합을 제거하여 EBS 볼륨에 대해 일관되게 우수한 성능을 제공할 수 있습니다. 일부 스토리지 최적화 인스턴스는 추가 비용 없이도 EBS에 최적화할 수 있게 기본 설정되어 있습니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

네트워크 성능

지원되는 인스턴스 유형에 대해 향상된 네트워킹을 활성화하면 지연 시간을 줄이고 네트워크 지터를 낮추며 PPS(Packet Per Second) 성능을 높일 수 있습니다. 대부분의 애플리케이션은 항상 높은 수준의 네트워크 성능을 필요로 하지 않지만, 데이터를 주고 받을 때 증가된 대역폭에 액세스할 수 있을 경우 유익할 수 있습니다. 자세한 내용은 [Windows에서 향상된 네트워킹 \(p. 751\)](#) 단원을 참조하십시오.

다음은 확장 네트워킹을 지원하는 스토리지 최적화 인스턴스용 네트워크 성능의 요약 설명입니다.

인스턴스 유형	네트워크 성능	향상된 네트워킹
i3.4xlarge 이하	최대 10Gbps †	ENA (p. 751)
i3.8xlarge h1.8xlarge	10Gbps	ENA (p. 751)
i3en.3xlarge 이하	최대 25Gbps †	ENA (p. 751)
i3.16xlarge i3.metal i3en.6xlarge h1.16xlarge	25Gbps	ENA (p. 751)
i3en.12xlarge	50Gbps	ENA (p. 751)
i3en.24xlarge i3en.metal	100Gbps	ENA (p. 751)
d2.xlarge	보통	intel 82599 VF (p. 758)
d2.2xlarge d2.4xlarge	높음	intel 82599 VF (p. 758)
d2.8xlarge	10Gbps	Intel 82599 VF (p. 758)

† 이러한 인스턴스는 네트워크 I/O 크레딧 메커니즘을 이용해 평균 대역폭 활용도를 기준으로 인스턴스에 네트워크 대역폭을 할당합니다. 해당 대역폭이 기준 대역폭 미만으로 떨어지면 크레딧이 발생하는데, 이 크레딧은 네트워크 데이터를 전송할 때 사용할 수 있습니다. 자세한 내용은 지원 사례를 개설하여 관심 있는 특정 인스턴스 유형의 기준 대역폭에 대해 문의하십시오.

SSD I/O 성능

를 사용하고 인스턴스에서 사용 가능한 모든 SSD 기반의 인스턴스 스토어 볼륨을 활용하는 경우, 다음 표와 같은 IOPS(블록 크기 4,096바이트) 성능을 얻을 수 있습니다(대기열 깊이 포화 상태에서). 그렇지 않으면 IOPS 성능이 더 낮아집니다.

인스턴스 크기	100% 임의 읽기 IOPS	IOPS 쓰기
i3.large *	100,125	35,000
i3.xlarge *	206,250	70,000
i3.2xlarge	412,500	180,000
i3.4xlarge	825,000	360,000
i3.8xlarge	1.65백만	720,000
i3.16xlarge	3.3백만	1.4백만
i3.metal	3.3백만	1.4백만
i3en.large *	42,500	32,500
i3en.xlarge *	85,000	65,000
i3en.2xlarge *	170,000	130,000
i3en.3xlarge	250,000	200,000

인스턴스 크기	100% 임의 읽기 IOPS	IOPS 쓰기
i3en.6xlarge	500,000	400,000
i3en.12xlarge	1백만 개	800,000
i3en.24xlarge	2백만	160만
i3en.metal	2백만	160만

* 이러한 인스턴스의 경우, 지정된 최대 성능을 얻을 수 있습니다.

SSD 기반 인스턴스 저장 볼륨이 줄어듦에 따라 사용자가 얻을 수 있는 I/O 성능이 감소합니다. 이는 SSD 컨트롤러가 사용 공간을 찾고 기존 데이터를 다시 쓰고 미사용 공간을 삭제하여 다시 쓸 수 있는 공간을 마련하기 위해 추가적인 작업을 해야 하기 때문입니다. 이러한 폐역역 회수 과정은 SSD에 대한 내부 쓰기 작업이 증폭되는 결과를 낳게 되며, 이런 결과는 사용자 쓰기 작업에 대한 SSD 쓰기 작업의 비로 표현됩니다. 이러한 성능 감소는 쓰기 작업이 4096바이트의 배수들 또는 4096바이트 경계에 정렬되지 않은 상태로 수행되는 경우에 더 심해질 수 있습니다. 정렬되지 않은 바이트를 소량으로 쓰기 작업하는 경우, SSD 컨트롤러는 쓰려는 부분의 주변 데이터를 읽고 그 결과도 새 위치에 저장해야 합니다. 이런 패턴으로 인해 쓰기 작업이 크게 증폭되고 지연 시간 증가와 I/O 성능의 급격한 감소를 초래합니다.

SSD 컨트롤러는 여러 전략을 사용해서 쓰기 작업 증폭의 영향을 감쇄할 수 있습니다. 그 중 하나의 전력은 SSD 인스턴스 스토리지에 예약 공간을 마련해서 SSD 컨트롤러가 쓰기 작업에 사용 가능한 공간을 보다 효율적으로 관리할 수 있게 하는 것입니다. 이를 오버-프로비저닝이라고 합니다. 인스턴스에 제공된 SSD 기반 인스턴스 스토어 볼륨은 오버-프로비저닝을 위한 예약 공간을 가지고 있지 않습니다. 쓰기 작업 증폭의 영향 감쇄를 위해 최소한 볼륨의 10%를 파티션 처리되지 않은 상태로 두어서 SSD 컨트롤러가 이를 오버-프로비저닝에 사용할 수 있도록 하는 것이 좋습니다. 그러면 사용할 수 있는 스토리지는 줄어들지만, 디스크를 전체 용량에 가깝게 사용하더라도 성능은 향상됩니다.

TRIM을 지원하는 인스턴스 스토어 볼륨의 경우, TRIM 명령을 사용하여 작성한 데이터가 더 이상 필요하지 않을 때 SSD 컨트롤러에 알릴 수 있습니다. 이를 통해 컨트롤러에 더 많은 여유 공간이 제공되어 쓰기 증폭이 줄어들고 성능이 향상될 수 있습니다. 자세한 내용은 [인스턴스 스토어 볼륨 TRIM 지원 \(p. 1102\)](#) 단원을 참조하십시오.

인스턴스 기능

스토리지 최적화 인스턴스에 대한 기능은 다음과 같이 간략히 설명할 수 있습니다.

	EBS 전용	인스턴스 스토어	배치 그룹
D2	아니요	HDD	예
H1	아니요	HDD *	예
I3	아니요	NVMe *	예
i3en	아니요	NVMe *	예

* 루트 디바이스 볼륨은 Amazon EBS 볼륨이어야 합니다.

자세한 내용은 다음 단원을 참조하십시오.

- [Windows 인스턴스의 Amazon EBS 및 NVMe \(p. 1048\)](#)
- [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#)
- [배치 그룹 \(p. 761\)](#)

릴리스 정보

- HVM AMI를 사용해서 스토리지 최적화 인스턴스를 실행해야 합니다.
- Nitro 시스템 (p. 118)에 구축된 인스턴스에는 다음과 같은 요구 사항이 있습니다.
 - NVMe 드라이버 (p. 1048)가 설치되어 있어야 합니다.
 - ENA(Elastic Network Adapter) (p. 751) 드라이버가 설치되어 있어야 합니다.

현재 AWS Windows AMI (p. 26)는 이러한 요구 사항을 충족합니다.

- 베어 메탈 인스턴스를 시작하면 기본 서버가 부팅되는데, 이때 모든 하드웨어 및 펌웨어 구성 요소를 확인합니다. 즉, 인스턴스가 실행 상태가 되어 네트워크를 통해 사용할 수 있게 될 때까지 20분이 걸릴 수 있습니다.
- 베어 메탈 인스턴스에서 EBS 볼륨 또는 보조 네트워크 인터페이스를 연결 또는 분리하려면 PCIe 기본 핫 플러그 지원이 필요합니다.
- 베어 메탈 인스턴스에서는 I/O 포트 기반 직렬 디바이스가 아닌 PCI 기반 직렬 디바이스를 사용합니다. 업스트림 Linux 커널 및 최신 Amazon Linux AMI에서는 이 디바이스를 지원합니다. 베어 메탈 인스턴스도 시스템에서 PCI 기반 직렬 디바이스를 자동으로 사용할 수 있게 해주는 ACPI SPCR 테이블을 제공합니다. 최신 Windows AMI에서는 PCI 기반 직렬 디바이스를 자동으로 사용합니다.
- 한 리전에서 시작할 수 있는 총 인스턴스 수에는 제한이 있으며, 일부 인스턴스 유형에는 또 다른 제한이 있습니다. 자세한 내용은 Amazon EC2 FAQ의 [Amazon EC2에서 실행할 수 있는 인스턴스 수는 몇 개입니까?](#)를 참조하십시오.

Windows 액셀러레이티드 컴퓨팅 인스턴스

높은 처리 기능이 필요한 경우 GPU(그래픽 처리 장치)와 같은 하드웨어 기반 컴퓨팅 액셀러레이터에 대한 액세스를 제공하는 액셀러레이티드 컴퓨팅 인스턴스를 사용하면 도움이 됩니다. 액셀러레이티드 컴퓨팅 인스턴스는 컴퓨팅 집약적 워크로드에서 더 높은 처리량을 위한 병렬 처리를 활성화합니다.

GPU 기반 인스턴스는 수천 개의 컴퓨팅 코어로 NVIDIA GPU에 대한 액세스를 제공합니다. 액셀러레이티드 컴퓨팅 인스턴스로 CUDA(Compute Unified Device Architecture) 또는 OpenCL(Open Computing Language) 병렬 컴퓨팅 프레임워크를 활용하여 GPU 기반 과학, 공학 및 렌더링 애플리케이션의 속도를 높일 수 있습니다. 게임 스트리밍, 3-D 애플리케이션 스트리밍 등의 그래픽 애플리케이션 및 기타 그래픽 워크로드에 활용할 수도 있습니다.

애플리케이션에서 그래픽을 추가로 약간 가속화해야 하는데 애플리케이션이 다른 컴퓨팅, 메모리 또는 스토리지 사양의 인스턴스 유형에 더 적합하다면 Elastic Graphics 액셀러레이터를 사용하십시오. 자세한 내용은 [Amazon Elastic Graphics \(p. 636\)](#) 단원을 참조하십시오.

가속 컴퓨팅 인스턴스를 하나의 클러스터 배치 그룹으로 클러스터링할 수 있습니다. 클러스터 배치 그룹은 단일 가용 영역 내의 인스턴스 간에 낮은 지연 시간과 높은 대역폭 연결을 제공합니다. 자세한 내용은 [배치 그룹 \(p. 761\)](#) 단원을 참조하십시오.

내용

- 액셀러레이티드 컴퓨팅 인스턴스 패밀리 (p. 183)
- 하드웨어 사양 (p. 184)
- 인스턴스 성능 (p. 185)
- 네트워크 성능 (p. 185)
- 인스턴스 기능 (p. 186)
- 릴리스 정보 (p. 186)
- Windows 인스턴스에 NVIDIA 드라이버 설치 (p. 187)
- NVIDIA GRID 가상 애플리케이션 활성화 (p. 192)
- GPU 설정 최적화 (p. 192)

Linux 액셀러레이티드 컴퓨팅 인스턴스에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux 액셀러레이티드 컴퓨팅 인스턴스](#)를 참조하십시오.

액셀러레이티드 컴퓨팅 인스턴스 패밀리

액셀러레이티드 컴퓨팅 인스턴스 패밀리는 하드웨어 액셀러레이터나 코프로세서를 사용함으로써 부동 소수점 계산 및 그래픽 처리, 또는 데이터 패턴 매칭 등과 같은 일부 기능을, CPU에서 실행하는 소프트웨어에서 수행하는 것보다 더 효율적으로 수행합니다. Amazon EC2에서는 다음과 같은 액셀러레이티드 컴퓨팅 인스턴스 패밀리를 실행할 수 있습니다.

P3 인스턴스

P3 인스턴스는 NVIDIA Tesla V100 GPU를 사용하며, CUDA 또는 OpenCL 프로그래밍 모델을 사용하거나 머신 러닝 프레임워크를 통해 범용 GPU 컴퓨팅에 맞게 설계되었습니다. P3 인스턴스는 고대역 네트워킹, 강력한 반정밀도, 단정밀도 및 배정밀도 부동 소수점 기능, GPU당 최대 32GiB의 메모리를 제공하므로, 딥 러닝, 전산 유체 역학(CFD), 계산 금융(Computational Finance), 내진 해석, 분자 모델링, 유전체학, 렌더링 및 기타 서버 측 GPU 컴퓨팅 워크로드에 이상적입니다. Tesla V100 GPU는 그래픽 모드를 지원하지 않습니다.

자세한 내용은 [Amazon EC2 P3 인스턴스](#)를 참조하십시오.

P3 인스턴스는 NVIDIA NVLink 피어 투 피어 전송을 지원합니다.

시스템에 대한 토폴로지 정보를 보려면 다음 명령을 실행합니다.

```
nvidia-smi topo -m
```

자세한 내용은 [NVIDIA NVLink](#) 단원을 참조하십시오.

P2 인스턴스

P2 인스턴스는 NVIDIA Tesla K80 GPU를 사용하며, CUDA 또는 OpenCL 프로그래밍 모델을 사용하는 일반 GPU 컴퓨팅에 맞게 설계되었습니다. P2 인스턴스는 고대역 네트워킹, 강력한 단일 정밀도 및 배정밀도 부동 소수점 기능, GPU당 12GiB의 메모리를 제공하므로, 딥 러닝, 그래프 데이터베이스, 고성능 데이터베이스, 전산 유체 역학(CFD), 계산 금융(Computational Finance), 내진 해석, 분자 모델링, 유전체학, 렌더링 및 기타 서버 GPU 컴퓨팅 워크로드에 이상적입니다.

P2 인스턴스는 NVIDIA GPUDirect 피어 투 피어 전송을 지원합니다.

시스템에 대한 토폴로지 정보를 보려면 다음 명령을 실행합니다.

```
nvidia-smi topo -m
```

자세한 내용은 [NVIDIA GPUDirect](#) 단원을 참조하십시오.

G4 인스턴스

G4 인스턴스는 NVIDIA Tesla GPU를 사용하며 DirectX 또는 OpenGL을 사용하는 그래픽 애플리케이션과 함께 CUDA 또는 기계 학습 프레임워크를 사용하여 범용 GPU 컴퓨팅을 위한 경제적인 고성능 플랫폼을 제공합니다. G4 인스턴스는 INT8 및 INT4 정밀도와 함께 고대역폭 네트워킹, 강력한 반정밀도 및 단정밀도 부동 소수점 기능을 제공합니다. 각 GPU에는 16GiB의 GDDR6 메모리가 있으므로 G4 인스턴스는 기계 학습 주론, 비디오 트랜스코딩, 원격 그래픽 워크스테이션 및 클라우드에서의 게임 스트리밍과 같은 그래픽 애플리케이션에 적합합니다.

G4 인스턴스는 NVIDIA GRID 가상 워크스테이션을 지원합니다. 자세한 내용은 [NVIDIA Marketplace 제품군](#)을 참조하십시오.

G3 인스턴스

G3 인스턴스는 NVIDIA Tesla M60 GPU를 사용하며 DirectX 또는 OpenGL을 사용하는 그래픽 애플리케이션을 위한 경제적이고도 높은 성능의 플랫폼을 제공합니다. 또한 G3 인스턴스는 해상도가 최대 4096x2160인 모니터 4대와 NVIDIA GRID 가상 애플리케이션을 지원하는 등 NVIDIA GRID 가상 워크스테이션 기능도 제공합니다. G3 인스턴스는 3D 가상화, 그래픽 집약적인 원격 워크스테이션, 3D 렌더링, 비디오 인코딩, 가상 현실 및 기타 서버 측 그래픽 워크로드 같이 뛰어난 병렬 처리 성능이 필요한 애플리케이션에 적합합니다.

G3 인스턴스는 NVIDIA GRID 가상 워크스테이션 및 NVIDIA GRID 가상 애플리케이션을 지원합니다. 이러한 기능들 중 하나를 활성화하는 방법은 [NVIDIA GRID 가상 애플리케이션 활성화 \(p. 192\)](#) 단원을 참조하십시오.

G2 인스턴스

G2 인스턴스는 NVIDIA GRID K520 GPU를 사용하며 DirectX 또는 OpenGL을 사용하는 그래픽 애플리케이션을 위한 경제적이고도 높은 성능의 플랫폼을 제공합니다. 또한 NVIDIA GRID GPU는 NVIDIA의 빠른 캡처 기능을 지원하고 API 연산을 인코딩합니다. 애플리케이션의 예로는 비디오 제작 서비스, 3D 가상화, 스트리밍 그래픽 집약적 애플리케이션 및 기타 서버 측 그래픽 워크로드 등을 들 수 있습니다.

하드웨어 사양

다음은 액셀러레이티드 컴퓨팅 인스턴스용 하드웨어 사양의 요약 설명입니다.

인스턴스 유형	기본 vCPU	메모리(GiB)	액셀러레이터
p2.xlarge	4	61	1
p2.8xlarge	32	488	8
p2.16xlarge	64	732	16
p3.2xlarge	8	61	1
p3.8xlarge	32	244	4
p3.16xlarge	64	488	8
p3dn.24xlarge	96	768	8
g2.2xlarge	8	15	1
g2.8xlarge	32	60	4
g3s.xlarge	4	30.5	1
g3.4xlarge	16	122	1
g3.8xlarge	32	244	2
g3.16xlarge	64	488	4
g4dn.xlarge	4	16	1
g4dn.2xlarge	8	32	1
g4dn.4xlarge	16	64	1
g4dn.8xlarge	32	128	1
g4dn.12xlarge	48	192	4
g4dn.16xlarge	64	256	1

인스턴스 유형	기본 vCPU	메모리(GiB)	액셀러레이터
g4dn.metal	96	384	8
f1.2xlarge	8	122	1
f1.4xlarge	16	244	2
f1.16xlarge	64	976	8

Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

CPU 옵션 지정에 대한 자세한 내용은 [CPU 옵션 최적화 \(p. 541\)](#) 단원을 참조하십시오.

인스턴스 성능

EBS에 최적화된 인스턴스를 사용하면 Amazon EBS I/O와 인스턴스의 다른 네트워크 간의 경합을 제거하여 EBS 볼륨에 대해 일관되게 우수한 성능을 제공할 수 있습니다. 일부 액셀러레이티드 컴퓨팅 인스턴스는 추가 비용 없이도 EBS에 최적화할 수 있게 기본 설정되어 있습니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

네트워크 성능

지원되는 인스턴스 유형에 대해 향상된 네트워킹을 활성화하면 지연 시간을 줄이고 네트워크 지터를 낮추며 PPS(Packet Per Second) 성능을 높일 수 있습니다. 대부분의 애플리케이션은 항상 높은 수준의 네트워크 성능을 필요로 하지 않지만, 데이터를 주고 받을 때 증가된 대역폭에 액세스할 수 있을 경우 유익할 수 있습니다. 자세한 내용은 [Windows에서 향상된 네트워킹 \(p. 751\)](#) 단원을 참조하십시오.

다음은 향상된 네트워킹을 지원하는 액셀러레이티드 컴퓨팅 인스턴스를 위한 네트워크 성능의 요약 설명입니다.

인스턴스 유형	네트워크 성능	향상된 네트워킹
f1.2xlarge f1.4xlarge g3.4xlarge p3.2xlarge	최대 10Gbps †	ENA (p. 751)
g3s.xlarge g3.8xlarge p2.8xlarge p3.8xlarge	10Gbps	ENA (p. 751)
g4dn.xlarge g4dn.2xlarge g4dn.4xlarge	최대 25Gbps †	ENA (p. 751)
f1.16xlarge g3.16xlarge p2.16xlarge p3.16xlarge	25Gbps	ENA (p. 751)
g4dn.8xlarge g4dn.12xlarge g4dn.16xlarge	50Gbps	ENA (p. 751)
g4dn.metal p3dn.24xlarge	100Gbps	ENA (p. 751)

† 이러한 인스턴스는 네트워크 I/O 크레딧 메커니즘을 이용해 평균 대역폭 활용도를 기준으로 인스턴스에 네트워크 대역폭을 할당합니다. 해당 대역폭이 기준 대역폭 미만으로 떨어지면 크레딧이 발생하는데, 이 크레딧은 네트워크 데이터를 전송할 때 사용할 수 있습니다. 자세한 내용은 지원 사례를 개설하여 관심 있는 특정 인스턴스 유형의 기준 대역폭에 대해 문의하십시오.

인스턴스 기능

액셀러레이티드 컴퓨팅 인스턴스를 위한 기능의 요약 설명입니다.

	EBS 전용	NVMe EBS	인스턴스 스토어	배치 그룹
G2	아니요	아니요	SSD	예
G3	예	아니요	아니요	예
G4	아니요	예	NVMe *	예
P2	예	아니요	아니요	예
P3	p3dn.24xlarge: 아니요 기타 모든 크기: 예	p3dn.24xlarge: 예 기타 모든 크기: 아니요	p3dn.24xlarge: NVMe *	예
SharePoint 용 F1	아니요	아니요	NVMe *	예

* 루트 디바이스 볼륨은 Amazon EBS 볼륨이어야 합니다.

자세한 내용은 다음 단원을 참조하십시오.

- [Windows 인스턴스의 Amazon EBS 및 NVMe \(p. 1048\)](#)
- [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#)
- [배치 그룹 \(p. 761\)](#)

릴리스 정보

- HVM AMI를 사용해서 인스턴스를 실행해야 합니다.
- Nitro 시스템 (p. 118)에 구축된 인스턴스에는 다음과 같은 요구 사항이 있습니다.
 - NVMe 드라이버 (p. 1048)가 설치되어 있어야 합니다.
 - ENA(Elastic Network Adapter) (p. 751) 드라이버가 설치되어 있어야 합니다.

현재 [AWS Windows AMI \(p. 26\)](#)는 이러한 요구 사항을 충족합니다.

- GPU 기반 인스턴스는 NVIDIA 드라이버를 설치해야 GPU에 액세스할 수 있습니다. 자세한 내용은 [Windows 인스턴스에 NVIDIA 드라이버 설치 \(p. 187\)](#) 단원을 참조하십시오.
- 베어 메탈 인스턴스를 시작하면 기본 서버가 부팅되는데, 이때 모든 하드웨어 및 펌웨어 구성 요소를 확인합니다. 즉, 인스턴스가 실행 상태가 되어 네트워크를 통해 사용할 수 있게 될 때까지 20분이 걸릴 수 있습니다.
- 베어 메탈 인스턴스에서 EBS 볼륨 또는 보조 네트워크 인터페이스를 연결 또는 분리하려면 PCIe 기본 핫 플러그 지원이 필요합니다.
- 베어 메탈 인스턴스에서는 I/O 포트 기반 직렬 디바이스가 아닌 PCI 기반 직렬 디바이스를 사용합니다. 업스트림 Linux 커널 및 최신 Amazon Linux AMI에서는 이 디바이스를 지원합니다. 베어 메탈 인스턴스도 시스템에서 PCI 기반 직렬 디바이스를 자동으로 사용할 수 있게 해주는 ACPI SPCR 테이블을 제공합니다. 최신 Windows AMI에서는 PCI 기반 직렬 디바이스를 자동으로 사용합니다.
- 리전당 100개의 AFI 제한이 있습니다.
- 한 리전에서 시작할 수 있는 총 인스턴스 수에는 제한이 있으며, 일부 인스턴스 유형에는 또 다른 제한이 있습니다. 자세한 내용은 Amazon EC2 FAQ에서 [Amazon EC2에서 실행 가능한 인스턴스 수](#)를 참조하십시오.

- 단일 GPU 인스턴스에서 생성된 Windows AMI를 사용하여 다중 GPU 인스턴스를 실행할 경우 4개 GPU 모두에 NVIDIA 드라이버가 설치되지는 않습니다. 새 GPU 하드웨어에 대한 드라이버 설치를 승인해야 합니다. 장치 관리자에서 기타 디바이스 범주(비활성 GPU는 디스플레이 어댑터에 표시되지 않음)를 열고 이를 직접 수정할 수 있습니다. 비활성 GPU의 경우 마우스 오른쪽 버튼을 클릭하여 상황에 맞는 메뉴를 열고 드라이버 소프트웨어 업데이트를 선택한 후 기본 옵션인 자동 업데이트를 선택합니다.
- Microsoft RDP(원격 데스크톱 프로토콜)를 사용하는 경우, WDDM 드라이버 모델을 사용하는 GPU는 비가속 원격 데스크톱 디스플레이 드라이버로 대체됩니다. [Teradici Cloud Access Software](#), [NICE DCV](#)(Desktop Cloud Visualization) 또는 VNC 등 다른 원격 액세스 도구를 사용하여 GPU에 액세스하는 것이 좋습니다. AWS Marketplace의 GPU AMI 중 하나를 사용할 수도 있습니다. 이러한 AMI는 3D 가속을 지원하는 원격 액세스 도구를 제공합니다.

Windows 인스턴스에 NVIDIA 드라이버 설치

P3 또는 G4 인스턴스와 같이 연결된 GPU가 있는 인스턴스에는 적절한 NVIDIA 드라이버가 설치되어 있어야 합니다. 인스턴스 유형에 따라 퍼블릭 NVIDIA 드라이버를 다운로드하거나, AWS 고객만 사용할 수 있는 Amazon S3에서 드라이버를 다운로드하거나, 드라이버가 미리 설치되어 있는 AMI를 사용할 수 있습니다.

목차

- [NVIDIA 드라이버의 유형](#) (p. 187)
- [인스턴스 유형별로 사용 가능한 드라이버](#) (p. 188)
- [설치 옵션](#) (p. 188)
 - 옵션 1: NVIDIA 드라이버가 설치되어 있는 AMI (p. 188)
 - 옵션 2: 퍼블릭 NVIDIA 드라이버 (p. 189)
 - 옵션 3: GRID 드라이버(G3 및 G4 인스턴스) (p. 189)
 - 옵션 4: NVIDIA 게임 드라이버(G4 인스턴스) (p. 190)
- [CUDA의 추가 버전 설치](#) (p. 192)

NVIDIA 드라이버의 유형

다음은 GPU 기반 인스턴스에서 사용할 수 있는 주요 NVIDIA 드라이버 유형입니다.

Tesla 드라이버

이러한 드라이버는 주로 컴퓨팅 워크로드를 위한 것입니다. 컴퓨팅 워크로드는 기계 학습을 위한 병렬화된 부동 소수점 계산과 고성능 컴퓨팅 애플리케이션을 위한 고속 푸리에 변환과 같은 컴퓨팅 작업에 GPU를 사용합니다.

GRID 드라이버

이러한 드라이버는 3D 모델 또는 고해상도 비디오와 같은 콘텐츠를 렌더링하는 전문 시각화 애플리케이션에 최적의 성능을 제공하는 것으로 인증됩니다. 두 가지 모드를 지원하도록 GRID 드라이버를 구성할 수 있습니다. Quadro 가상 워크스테이션은 GPU당 4개의 4K 디스플레이에 대한 액세스를 제공합니다. GRID vApp은 RDSH 앱 호스팅 기능을 제공합니다.

게임 드라이버

이러한 드라이버는 게임을 위한 최적화 기능을 포함하고 있으며 성능 향상을 제공하기 위해 자주 업데이트됩니다. 이러한 드라이버는 GPU당 단일 4K 디스플레이를 지원합니다.

구성된 모드

Windows에서 Tesla 드라이버는 TCC(Tesla 컴퓨팅 클러스터) 모드에서 실행되도록 구성됩니다. GRID 및 게임 드라이버는 WDDM(Windows 디스플레이 드라이버 모델) 모드에서 실행되도록 구성됩니다. TCC 모드에서 이 카드는 컴퓨팅 워크로드에 전용됩니다. WDDM 모드에서 이 카드는 컴퓨팅 워크로드와 그래픽 워크로드를 모두 지원합니다.

NVIDIA 제어판

NVIDIA 제어판은 GRID 및 게임 드라이버에서 지원됩니다. Tesla 드라이버에서는 지원되지 않습니다.

Tesla 드라이버에 지원되는 API

- OpenCL
- NVIDIA CUDA 및 관련 라이브러리(예: cuDNN, TensorRT, nvJPEG, cuBLAS)
- 비디오 인코딩용 NVENC 및 비디오 디코딩용 NVDEC

GRID 및 게임 드라이버에 지원되는 API

- DirectX, Direct2D, DirectX 비디오 가속, DirectX Raytracing
- OpenCL, OpenGL 및 Vulkan
- NVIDIA CUDA 및 관련 라이브러리(예: cuDNN, TensorRT, nvJPEG, cuBLAS)
- 비디오 인코딩용 NVENC 및 비디오 디코딩용 NVDEC

인스턴스 유형별로 사용 가능한 드라이버

다음 표에는 각 GPU 인스턴스 유형에 지원되는 NVIDIA 드라이버가 요약되어 있습니다.

인스턴스 유형	Tesla 드라이버	GRID 드라이버	게임 드라이버
G2	아니요	예	아니요
G3	예	예	아니요
G4	예	예	예
P2	예	아니요	아니요
P3	예	예 †	아니요

† Marketplace AMI만 사용

설치 옵션

다음 옵션 중 하나를 사용하여 GPU 인스턴스에 필요한 NVIDIA 드라이버를 가져옵니다.

옵션

- [옵션 1: NVIDIA 드라이버가 설치되어 있는 AMI \(p. 188\)](#)
- [옵션 2: 퍼블릭 NVIDIA 드라이버 \(p. 189\)](#)
- [옵션 3: GRID 드라이버\(G3 및 G4 인스턴스\) \(p. 189\)](#)
- [옵션 4: NVIDIA 게임 드라이버\(G4 인스턴스\) \(p. 190\)](#)

옵션 1: NVIDIA 드라이버가 설치되어 있는 AMI

AWS와 NVIDIA는 NVIDIA 드라이버가 설치된 상태로 제공되는 다양한 Amazon 머신 이미지(AMI)를 제공합니다.

- [Tesla 드라이버와 함께 제공되는 Marketplace 상품](#)
- [GRID 드라이버와 함께 제공되는 Marketplace 상품](#)

- 게임 드라이버와 함께 제공되는 Marketplace 상품

옵션 2: 퍼블릭 NVIDIA 드라이버

AWS에서 제공하는 옵션은 드라이버에 필요한 라이선스와 함께 제공됩니다. 또는 퍼블릭 드라이버를 설치하고 기존 보유 라이선스를 사용할 수도 있습니다. 퍼블릭 드라이버를 설치하려면 여기에 설명된 대로 NVIDIA 사이트에서 다운로드합니다.

또는 퍼블릭 드라이버 대신 AWS에서 제공하는 옵션을 사용할 수 있습니다. P3 인스턴스에서 GRID 드라이버를 사용하려면 [옵션 1 \(p. 188\)](#)에 설명된 대로 AWS Marketplace AMI를 사용합니다. G3 또는 G4 인스턴스에서 GRID 드라이버를 사용하려면 옵션 1에 설명된 대로 AWS Marketplace AMI를 사용하거나 [옵션 3 \(p. 189\)](#)에 설명된 대로 AWS에서 제공하는 NVIDIA 드라이버를 설치합니다.

퍼블릭 NVIDIA 드라이버를 다운로드하려면

Windows 인스턴스에 로그인하고 <http://www.nvidia.com/Download/Find.aspx>에서 인스턴스 유형에 적합한 64비트 NVIDIA 드라이버를 다운로드할 수 있습니다. 제품 유형, 제품 시리즈 및 제품에 다음 표의 옵션을 사용합니다.

인스턴스	제품 유형	제품 시리즈	제품
G2	GRID	GRID 시리즈	GRID K520
G3	Tesla	M-Class	M60
G4 †	Tesla	T 시리즈	T4
P2	Tesla	K 시리즈	K80
P3	Tesla	V 시리즈	V100

† G4 인스턴스에는 드라이버 버전 426.00 이상이 필요합니다.

Windows에서 NVIDIA 드라이버를 설치하려면

- 드라이버를 다운로드한 폴더를 열고 설치 파일을 실행합니다. 안내에 따라 드라이버를 설치하고 필요에 따라 인스턴스를 재부팅합니다.
- 장치 관리자를 사용하여 기본 디스플레이 어댑터를 비활성화합니다. Windows 기능인 미디어 파운데이션 및 qWave(Quality Windows Audio Video Experience)를 설치합니다.
- GPU가 올바르게 작동하는지 확인하려면 장치 관리자를 확인합니다.
- GPU에서 최상의 성능을 얻으려면 [GPU 설정 최적화 \(p. 192\)](#)의 최적화 단계를 완료합니다.

옵션 3: GRID 드라이버(G3 및 G4 인스턴스)

이러한 다운로드는 AWS 고객만 사용할 수 있습니다. 드라이버를 다운로드하면 NVIDIA Tesla T4 또는 NVIDIA Tesla M60 하드웨어와 함께 사용하기 위해 AMIs를 개발하는 용도로만 다운로드한 소프트웨어를 사용한다는 것에 동의하는 것입니다. 소프트웨어를 설치하면 [NVIDIA GRID 클라우드 최종 사용자 라이선스 계약의 약관이 적용됩니다.](#)

사전 조건

- Windows 인스턴스에서 Windows PowerShell용 AWS 도구에 대한 기본 자격 증명을 구성합니다. 자세한 내용은 Windows PowerShell용 AWS 도구 사용 설명서의 [Windows PowerShell용 AWS 도구 시작하기](#)를 참조하십시오.
- IAM 사용자는 AmazonS3ReadOnlyAccess 정책에서 부여한 권한을 가지고 있어야 합니다.

Windows 인스턴스에서 NVIDIA GRID 드라이버를 설치하려면

1. Windows 인스턴스에 연결하고 PowerShell 창을 엽니다.
2. 다음 PowerShell 명령을 사용하여 Amazon S3에서 드라이버와 [NVIDIA GRID 클라우드 최종 사용자 라이선스 계약](#)을 데스크톱에 다운로드합니다.

```
$Bucket = "ec2-windows-nvidia-drivers"
$keyPrefix = "latest"
$localPath = "$home\Desktop\NVIDIA"
$objects = Get-S3Object -BucketName $Bucket -KeyPrefix $keyPrefix -Region us-east-1
foreach ($object in $objects) {
    $localFileName = $object.Key
    if ($localFileName -ne '' -and $object.Size -ne 0) {
        $localFilePath = Join-Path $localPath $localFileName
        Copy-S3Object -BucketName $Bucket -Key $object.Key -LocalFile $localFilePath -Region us-east-1
    }
}
```

여러 버전의 NVIDIA GRID 드라이버가 이 버킷에 저장되어 있습니다. `-KeyPrefix $keyPrefix` 옵션을 제거하면 버킷에서 사용 가능한 모든 버전을 다운로드할 수 있습니다.

GRID 버전 11.0부터는 latest에서 G3 및 G4 인스턴스 모두에 드라이버를 사용할 수 있습니다. 11.0 이후 버전은 g4/latest에 추가되지 않지만 G4에 해당하는 11.0 이하 버전은 g4/latest에 유지됩니다.

3. 바탕 화면으로 이동하여 설치 파일을 두 번 클릭하여 시작합니다(인스턴스 OS 버전에 해당하는 드라이버 버전 선택). 안내에 따라 드라이버를 설치하고 필요에 따라 인스턴스를 재부팅합니다. GPU가 제대로 작동하는지 확인하려면 장치 관리자를 확인합니다.
4. (선택 사항) 다음 명령으로 제어판에서 라이선싱 페이지를 비활성화하여 사용자가 실수로 제품 유형을 변경하는 것을 방지합니다(NVIDIA GRID 가상 워크스테이션은 기본적으로 활성화되어 있음). 자세한 내용은 [GRID 라이선싱 사용 설명서](#)를 참조하십시오.

```
New-ItemProperty -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global\GridLicensing" -Name "NvCplDisableManageLicensePage" -PropertyType "DWord" -Value "1"
```

5. (선택 사항) 사용 사례에 따라 다음과 같은 선택적 단계를 완료할 수 있습니다. 이 기능이 필요하지 않으면 다음 단계를 완료하지 마십시오.
 - a. 최대 4K 해상도의 디스플레이 4개를 활용하는 데 도움이 되도록 고성능 디스플레이 프로토콜인 [NICE DCV](#)를 설정합니다.
 - b. NVIDIA Quadro 가상 워크스테이션 모드는 기본적으로 활성화되어 있습니다. RDSH 애플리케이션 호스팅 기능을 위해 GRID 가상 애플리케이션을 활성화하려면 [NVIDIA GRID 가상 애플리케이션 활성화](#) (p. 192)의 GRID 가상 애플리케이션 활성화 단계를 완료하십시오.

옵션 4: NVIDIA 게임 드라이버(G4 인스턴스)

이러한 드라이버는 AWS 고객만 사용할 수 있습니다. 드라이버를 다운로드하면 NVIDIA Tesla T4 하드웨어와 함께 사용하기 위해 AMIs를 개발하는 용도로만 다운로드한 소프트웨어를 사용한다는 것에 동의하는 것입니다. 소프트웨어를 설치하면 [NVIDIA GRID 클라우드 최종 사용자 라이선스 계약](#)의 약관이 적용됩니다.

사전 조건

- Windows 인스턴스에서 Windows PowerShell용 AWS 도구에 대한 기본 자격 증명을 구성합니다. 자세한 내용은 Windows PowerShell용 AWS 도구 사용 설명서의 [Windows PowerShell용 AWS 도구 시작하기](#)를 참조하십시오.
- IAM 사용자는 AmazonS3ReadOnlyAccess 정책에서 부여한 권한을 가지고 있어야 합니다.

Windows 인스턴스에서 NVIDIA 게임 드라이버를 설치하려면

1. Windows 인스턴스에 연결하고 PowerShell 창을 엽니다.
2. 다음 PowerShell 명령을 사용하여 게임 드라이버를 다운로드하고 설치합니다.

```
$Bucket = "nvidia-gaming"
$keyPrefix = "windows/latest"
$LocalPath = "$home\Desktop\NVIDIA"
$Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $keyPrefix -Region us-east-1
foreach ($Object in $Objects) {
    $LocalFileName = $Object.Key
    if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
        $LocalFilePath = Join-Path $LocalPath $LocalFileName
        Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -Region us-east-1
    }
}
```

여러 버전의 NVIDIA GRID 드라이버가 이 S3 버킷에 저장되어 있습니다. -KeyPrefix \$keyPrefix 옵션을 제거하면 버킷에서 사용 가능한 모든 버전을 다운로드할 수 있습니다.

3. 바탕 화면으로 이동하여 설치 파일을 두 번 클릭하여 시작합니다(인스턴스 OS 버전에 해당하는 드라이버 버전 선택). 안내에 따라 드라이버를 설치하고 필요에 따라 인스턴스를 재부팅합니다. GPU가 제대로 작동하는지 확인하려면 장치 관리자를 확인합니다.
4. HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global 키에 이름이 vGamingMarketplace이고, 유형이 DWord이고, 값이 2인 레지스트리 값을 생성합니다. 다음과 같이 명령 프롬프트 창 또는 64비트 버전의 PowerShell을 사용할 수 있습니다.

```
New-ItemProperty -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global" -Name "vGamingMarketplace" -PropertyType "DWord" -Value "2"
```

- 다음 레지스트리 명령을 사용하여 이 레지스트리 값을 생성합니다. 명령 프롬프트 창 또는 64비트 버전의 PowerShell을 사용하여 실행할 수 있습니다.

```
reg add "HKLM\SOFTWARE\NVIDIA Corporation\Global" /v vGamingMarketplace /t REG_DWORD /d 2
```

5. 다음 명령을 사용하여 인증 파일을 다운로드하고 GridSwCert.txt 파일 이름을 바꾸고 파일을 시스템 드라이브의 공용 문서 폴더로 이동하십시오. 일반적으로 폴더 경로는 C:\Users\Public\Public Documents(Windows Explorer) 또는 C:\Users\Public\Documents(명령 프롬프트 창)입니다.
- 버전 445.87 이상:

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Windows_2020_04.cert" -OutFile "$Env:PUBLIC\Documents\GridSwCert.txt"
```

- 이전 버전:

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Windows_2019_09.cert" -OutFile "$Env:PUBLIC\Documents\GridSwCert.txt"
```

6. 인스턴스를 재부팅합니다.
7. 다음 명령을 사용하여 NVIDIA 게임 라이선스를 확인합니다.

```
"C:\Program Files\NVIDIA Corporation\NVSMI\nvidia-smi.exe" -q
```

다음과 같이 출력됩니다

GRID Licensed Product	
Product Name	: GRID vGaming
License Status	: Licensed

8. (선택 사항) 최대 4K 해상도의 단일 디스플레이를 활용하는 데 도움이 되도록 고성능 디스플레이 프로토콜인 [NICE DCV](#)를 설정합니다. 이 기능이 필요하지 않으면 다음 단계를 완료하지 마십시오.

CUDA의 추가 버전 설치

인스턴스에 NVIDIA 그래픽 드라이버를 설치한 후 그래픽 드라이버와 함께 번들로 제공되는 버전이 아닌 CUDA 버전을 설치할 수 있습니다. 다음 절차에서는 인스턴스에서 여러 버전의 CUDA를 구성하는 방법을 보여줍니다.

CUDA 도구 키트를 설치하려면

1. Windows 인스턴스에 연결합니다.
2. [NVIDIA 웹 사이트](#)를 열고 필요한 CUDA 버전을 선택합니다.
3. 설치 프로그램 유형에서 exe(로컬)를 선택한 다음 다운로드를 선택합니다.
4. 브라우저를 사용하여 다운로드한 설치 파일을 실행합니다. 지침에 따라 CUDA 도구 키트를 설치합니다. 인스턴스를 재부팅해야 할 수 있습니다.

NVIDIA GRID 가상 애플리케이션 활성화

G3 및 G4 인스턴스에서 GRID 가상 애플리케이션을 활성화하려면(NVIDIA GRID 가상 워크스테이션은 기본적으로 활성화되어 있음) 레지스트리에서 드라이버에 맞는 제품 유형을 정의해야 합니다.

Windows 인스턴스에서 GRID 가상 애플리케이션을 활성화하려면

1. regedit.exe를 실행하여 레지스트리 편집기를 엽니다.
2. HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global\GridLicensing으로 이동합니다.
3. 오른쪽 창에서 마우스 오른쪽 버튼을 클릭하여 컨텍스트 메뉴를 열고 새로 생성(New)과 DWORD를 차례로 선택합니다.
4. 이름에 FeatureType을 입력한 다음 Enter를 입력합니다.
5. FeatureType에서 마우스 오른쪽 버튼을 클릭하여 컨텍스트 메뉴를 열고 수정(Modify)을 선택합니다.
6. 값 데이터(Value data)에서 NVIDIA GRID 가상 애플리케이션에 대해 0을 입력하고 확인(OK)을 선택합니다.
7. 오른쪽 창에서 마우스 오른쪽 버튼을 클릭하여 컨텍스트 메뉴를 열고 새로 생성(New)과 DWORD를 차례로 선택합니다.
8. 이름(Name)에 IgnoreSP를 입력한 다음 Enter를 누릅니다.
9. IgnoreSP에서 마우스 오른쪽 버튼을 클릭하여 컨텍스트 메뉴를 열고 수정(Modify)을 선택합니다.
10. 값 데이터(Value data)에 1을 입력하고 확인(OK)을 선택합니다.
11. 레지스트리 편집기를 닫습니다.

GPU 설정 최적화

G3, G4, P2, P3, P3dn 인스턴스에서 최고의 성능을 달성하기 위해 수행할 수 있는 몇 가지 GPU 설정 최적화가 있습니다. 기본적으로 NVIDIA 드라이버는 GPU 클록 속도에 변화를 주는 자동 부스트 기능을 사용합니

다. 자동 부스트 기능을 비활성화하고 GPU 클록 속도를 최대 주파수로 설정하면 GPU 인스턴스의 성능을 최대로 유지할 수 있습니다.

GPU 설정을 최적화하려면

- PowerShell 창을 열고 NVIDIA 설치 폴더를 탐색합니다.

```
cd "C:\Program Files\NVIDIA Corporation\NVSMI"
```

- 인스턴스에 대해 모든 GPU의 자동 부스트 기능을 비활성화합니다.

```
.\nvidia-smi --auto-boost-default=0
```

Note

P3, P3dn 및 G4 인스턴스에서 GPU는 자동 부스트를 지원하지 않습니다.

- 모든 GPU 클록 속도를 최대 주파수로 설정합니다. 다음 명령에 지정된 메모리와 그래픽 클록 속도를 사용합니다.

Note

일부 버전의 NVIDIA 드라이버는 애플리케이션 클록 속도를 설정할 수 없으며 "Setting applications clocks is not supported for GPU ..." 오류가 발생합니다. 이 오류는 무시해도 됩니다.

- G3 인스턴스:

```
.\nvidia-smi -ac "2505,1177"
```

- G4 인스턴스:

```
.\nvidia-smi -ac "5001,1590"
```

- P2 인스턴스:

```
.\nvidia-smi -ac "2505,875"
```

- P3 및 P3dn 인스턴스::

```
.\nvidia-smi -ac "877,1530"
```

Amazon EC2 인스턴스 유형 찾기

인스턴스를 시작하려면 먼저 사용할 인스턴스 유형을 선택해야 합니다. 선택하는 인스턴스 유형은 시작할 인스턴스의 요구 사항에 따라 다를 수 있습니다. 예를 들어 다음 요구 사항에 따라 인스턴스 유형을 선택할 수 있습니다.

- 가용 영역 또는 리전
- 컴퓨팅
- Memory
- 네트워킹
- 요금
- 스토리지

콘솔을 사용하여 인스턴스 유형 찾기

Amazon EC2 콘솔을 사용하여 필요에 맞는 인스턴스 유형을 찾을 수 있습니다.

콘솔을 사용하여 인스턴스 유형을 찾으려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 인스턴스를 실행할 리전을 선택합니다. 현재 위치와 관계없이 사용자가 고를 수 있는 리전을 임의로 선택합니다.
3. 탐색 창에서 인스턴스 유형을 선택합니다.
4. (선택 사항) 기본 설정(기어 모양) 아이콘을 선택하여 표시할 인스턴스 유형 속성(예: 온디맨드 Linux 요금)을 선택한 다음 확인을 선택합니다. 또는 인스턴스 유형을 선택하고 세부 정보 창을 사용하여 모든 속성을 봅니다.
5. 인스턴스 유형 속성을 사용하여 표시된 인스턴스 유형 목록을 필요에 맞는 인스턴스 유형으로만 필터링 합니다. 예를 들어, vCPU가 8개를 초과하고 최대 절전 모드도 지원하는 모든 인스턴스 유형을 나열할 수 있습니다.
6. (선택 사항) 여러 인스턴스 유형을 선택하여 세부 정보 창에 있는 모든 속성을 항목별로 비교해 봅니다.
7. (선택 사항) 추가 검토를 위해 인스턴스 유형 목록을 쉼표로 구분된 값(.csv) 파일에 저장하려면 목록 CSV 다운로드를 선택합니다. 이 파일에는 사용자가 설정한 필터와 일치하는 모든 인스턴스 유형이 포함됩니다.
8. 필요에 맞는 인스턴스 유형을 찾은 후 이를 사용하여 인스턴스를 시작할 수 있습니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오.

AWS CLI를 사용하여 인스턴스 유형 찾기

Amazon EC2에 대한 AWS CLI 명령을 사용하여 필요에 맞는 인스턴스 유형을 찾을 수 있습니다.

AWS CLI를 사용하여 인스턴스 유형을 찾으려면

1. 아직 설치하지 않았다면 AWS CLI를 설치합니다. 자세한 내용은 [AWS Command Line Interface 사용 설명서](#)를 참조하십시오.
2. `describe-instance-types` 명령을 사용하여 인스턴스 속성을 기준으로 인스턴스 유형을 필터링합니다. 예를 들어, 다음 명령을 사용하여 vCPU가 48개 있는 인스턴스 유형만 표시할 수 있습니다.

```
aws ec2 describe-instance-types --filters "Name=vcpu-info.default-vcpus,Values=48"
```

3. `describe-instance-type-offerings` 명령을 사용하여 위치(리전 또는 가용 영역)별로 제공되는 인스턴스 유형을 필터링합니다. 예를 들어, 다음 명령을 사용하여 지정된 가용 영역에서 제공되는 인스턴스 유형을 표시할 수 있습니다.

```
aws ec2 describe-instance-type-offerings --location-type "availability-zone" --filters Name=location,Values=us-east-2a --region us-east-2
```

4. 필요에 맞는 인스턴스 유형을 찾은 후 인스턴스를 시작할 때 이러한 인스턴스 유형을 사용할 수 있도록 해당 인스턴스 유형을 기록해 둡니다. 자세한 내용은 AWS Command Line Interface 사용 설명서의 [AWS CLI를 사용하여 인스턴스 시작](#)을 참조하십시오.

인스턴스 유형 변경

요구 사항이 변함에 따라 인스턴스가 과도하게(인스턴스 유형 크기가 너무 작은 경우) 또는 과소하게(인스턴스 유형 크기가 너무 큰 경우) 활용되고 있는 경우가 생길 수 있습니다. 이 경우는 인스턴스의 크기를 변경할

수 있습니다. 예를 들어 t2.micro 인스턴스가 워크로드에 비해 너무 작은 경우는 이를 워크로드에 적합한 다른 인스턴스 유형으로 변경할 수 있습니다.

또는 IPv6 지원과 같은 일부 기능의 장점을 활용하기 위해 이전 세대 인스턴스 유형에서 현재 세대 인스턴스 유형으로 마이그레이션할 수도 있습니다.

인스턴스 유형을 변경하여 간단히 인스턴스의 크기를 변경할 수 있습니다. 이를 크기 조정이라고 합니다.

인스턴스의 크기를 조정할 경우 인스턴스의 구성과 호환되는 인스턴스 유형을 선택해야 합니다. 필요한 인스턴스 유형이 해당 인스턴스 구성과 호환되지 않을 경우, 원하는 인스턴스 유형의 새 인스턴스로 애플리케이션을 마이그레이션해야 합니다.

Important

인스턴스의 크기를 조정할 때 일반적으로 크기를 조정한 인스턴스는 원본 인스턴스를 시작할 때 지정한 것과 동일한 수의 인스턴스 스토어 볼륨을 갖습니다. (기본적으로 제공되는) NVMe 인스턴스 스토어 볼륨을 지원하는 인스턴스 유형의 경우, 크기 조정된 인스턴스는 AMI에 따라 추가 인스턴스 스토어 볼륨을 가질 수도 있습니다. 그렇지 않다면 애플리케이션을 새 인스턴스 유형이 있는 인스턴스로 직접 마이그레이션해, 새 인스턴스 실행 시 필요한 인스턴스 스토어 볼륨 수를 지정하면 됩니다.

목차

- [인스턴스 크기 조정을 위한 호환성 \(p. 195\)](#)
- [Amazon EBS 지원 인스턴스 크기 조정 \(p. 196\)](#)
- [새 인스턴스 구성으로 마이그레이션 \(p. 197\)](#)

인스턴스 크기 조정을 위한 호환성

인스턴스의 현재 인스턴스 유형과 새 인스턴스 유형이 호환될 경우에만 다음과 같이 인스턴스 크기를 조정할 수 있습니다.

- **아키텍처:** AMI는 프로세서의 아키텍처에 고유하기 때문에 프로세서 아키텍처가 현재 인스턴스 유형과 동일한 인스턴스를 선택해야 합니다. 예:
 - Arm 아키텍처 기반 프로세서를 사용하는 인스턴스 유형의 크기를 조정하는 경우 Arm 아키텍처 기반 프로세서를 지원하는 인스턴스 유형(예: A1 및 M6g)으로 제한됩니다.
 - 다음 인스턴스 유형은 32비트 AMIs를 지원하는 유일한 인스턴스 유형입니다. t2.nano, t2.micro, t2.small, t2.medium, c3.large, t1.micro, m1.small, m1.medium 및 c1.medium 32비트 인스턴스의 크기를 조정하는 경우는 상기 인스턴스 유형만 사용 가능합니다.
- **네트워크:** 최신 인스턴스 유형은 VPC에서 시작해야 합니다. 따라서 기본이 아닌 VPC가 아닌 한 EC2-Classic의 인스턴스를, VPC에서만 사용할 수 있는 인스턴스 유형으로 크기 조정할 수 없습니다. 인스턴스가 VPC에 있는지 확인하려면 Amazon EC2 콘솔의 인스턴스 화면에서 세부 정보 창의 VPC ID 값을 확인하십시오. 자세한 내용은 [EC2-Classic에서 VPC로 마이그레이션 \(p. 820\)](#) 단원을 참조하십시오.
- **네트워크 어댑터:** 한 네트워크 어댑터에서 다른 네트워크 어댑터로 드라이버를 전환하는 경우 운영 체제에서 새 어댑터를 만들 때 네트워크 어댑터 설정이 재설정됩니다. 설정을 다시 구성하려면 관리자 권한이 있는 로컬 계정에 액세스해야 할 수 있습니다. 다음은 한 네트워크 어댑터에서 다른 네트워크 어댑터로 전환하는 예입니다.
 - AWS PV(T2 인스턴스)에서 인텔 82599 VF(M4 인스턴스)로
 - 인텔 82599 VF(대부분의 M4 인스턴스)에서 ENA(M5 인스턴스)로
 - ENA(M5 인스턴스)에서 고대역폭 ENA(M5n 인스턴스)로
- **향상된 네트워킹:** [향상된 네트워킹 \(p. 751\)](#)을 지원하는 인스턴스 유형을 사용하려면 필요한 드라이버가 설치되어 있어야 합니다. 예를 들어 인스턴스 유형에는 ENA(Elastic Network Adapter) 드라이버가 설치된 EBS 지원 AMI가 필요합니다. 향상된 네트워킹을 지원하는 인스턴스에 맞게 기존 인스턴스의 크기를 조정하려는 경우, 먼저 해당 인스턴스에 [ENA 드라이버 \(p. 751\)](#) 또는 [ixgbevf 드라이버 \(p. 758\)](#)를 적절히 설치해야 합니다.

- NVMe: [Nitro 시스템 \(p. 118\)](#) 기반 인스턴스에서는 EBS 볼륨이 NVMe 블록 디바이스로 표시됩니다. NVMe를 지원하지 않는 인스턴스 유형에서 NVMe를 지원하는 인스턴스 유형으로 인스턴스의 크기를 조정하는 경우 먼저, 인스턴스에 [NVMe 드라이버 \(p. 1048\)](#)를 설치해야 합니다. 또한 블록 디바이스 매핑에서 지정한 디바이스에 대한 디바이스 이름은 NVMe 디바이스 이름(/dev/nvme[0-26]n1)을 이용하여 변경됩니다.
- AMI: 향상된 네트워킹 및 NVMe를 지원하는 인스턴스 유형별로 필요한 AMI에 대한 정보는 다음 설명서의 릴리스 정보를 참조하십시오.
 - [범용 인스턴스 \(p. 122\)](#)
 - [컴퓨팅 최적화 인스턴스 \(p. 163\)](#)
 - [메모리 최적화 인스턴스 \(p. 168\)](#)
 - [스토리지 최적화 인스턴스 \(p. 177\)](#)

Amazon EBS 지원 인스턴스 크기 조정

인스턴스 유형을 변경하기 전에는 Amazon EBS- 지원 인스턴스를 중단해야 합니다. 인스턴스를 중지했다가 시작할 때 다음 사항을 인식하십시오.

- 인스턴스를 새 하드웨어로 이동하지만, 인스턴스 ID는 변경되지 않습니다.
- 인스턴스에 퍼블릭 IPv4 주소가 있는 경우 주소를 해제하고 새 퍼블릭 IPv4 주소를 제공합니다. 인스턴스는 프라이빗 IPv4 주소와 모든 탄력적 IP 주소(EIP), IPv6 주소를 유지합니다.
- 인스턴스가 Auto Scaling 그룹에 있는 경우, Amazon EC2 Auto Scaling 서비스는 중단된 인스턴스를 비정상으로 간주해 이를 종료하고 대체 인스턴스를 시작합니다. 이를 방지하기 위해서는 해당 인스턴스의 크기 를 조정하는 동안 그 그룹에 대한 조정 프로세스를 일시 중지할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [조정 프로세스 일시 중단 및 재개 단원](#)을 참조하십시오.
- 인스턴스가 [클러스터 배치 그룹 \(p. 762\)](#)에 있으며 인스턴스 유형을 변경한 후 인스턴스 시작에 실패한 경우 다음 작업을 시도합니다. 클러스터 배치 그룹의 모든 인스턴스를 중지하고 영향을 받은 인스턴스의 인스턴스 유형을 변경한 다음 클러스터 배치 그룹의 모든 인스턴스를 다시 시작합니다.
- 가동 중지는 인스턴스가 중단되었을 때 계획해야 합니다. 인스턴스 중단 및 크기 조정은 몇 분이 걸릴 수 있으며, 인스턴스를 다시 시작하는 시간은 애플리케이션의 시작 스크립트에 따라 달라질 수 있습니다.

자세한 내용은 [인스턴스 중지 및 시작 \(p. 440\)](#) 단원을 참조하십시오.

다음 절차를 사용해서 AWS Management 콘솔을 통해 Amazon EBS- 지원 인스턴스의 크기를 조정합니다.

Amazon EBS- 지원 인스턴스의 크기 조정

1. (선택 사항) 새로운 인스턴스 유형이 기존 인스턴스에 드라이버가 설치되어 있어야 하는 유형인 경우, 먼저 인스턴스에 연결하여 해당 드라이버를 설치해야 합니다. 자세한 내용은 [인스턴스 크기 조정을 위한 호환성 \(p. 195\)](#) 단원을 참조하십시오.

Note

AWS PV 드라이버 패키지는 인스턴스 패밀리를 변경하기 전에 업데이트해야 합니다. 자세한 내용은 [Windows 인스턴스에서 PV 드라이버 업그레이드 \(p. 528\)](#) 단원을 참조하십시오.

2. (선택 사항) [고정 IP 주소 지정 \(p. 564\)](#)을 사용하도록 Windows 인스턴스를 구성했고, 향상된 네트워킹을 지원하지 않는 인스턴스 유형에서 향상된 네트워킹을 지원하는 인스턴스 유형으로 인스턴스 크기를 조정하는 경우, 고정 IP 주소 지정을 다시 구성하면 잠재적 IP 주소 충돌에 대한 경고를 받을 수 있습니다. 이를 방지하려면 인스턴스 유형을 변경하기 전에 해당 인스턴스의 네트워크 인터페이스에서 DHCP를 활성화합니다. 인스턴스에서 네트워크 및 공유 센터(Network and Sharing Center)를 열고, 네트워크 인터페이스의 인터넷 프로토콜 버전 4(TCP/IPv4) 속성(Internet Protocol Version 4 (TCP/IPv4) Properties)으로 이동한 다음, 자동으로 IP 주소 획득(Obtain an IP address automatically)을 선택합니다. 인스턴스 유형을 변경하고 네트워크 인터페이스에서 고정 IP 주소 지정을 다시 구성합니다.
3. Amazon EC2 콘솔을 엽니다.

- [Windows Server 2016 이후] Windows 인스턴스에 연결하고 다음 EC2Launch PowerShell 스크립트를 실행하여 크기 조정된 후의 인스턴스를 구성합니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -  
Schedule
```

- 탐색 창에서 인스턴스를 선택합니다.
- 인스턴스를 선택하고 작업, 인스턴스 상태, 중지를 선택합니다.
- 확인 대화 상자가 나타나면 Yes, Stop을 선택합니다. 인스턴스가 중지하는 데 몇 분 정도 걸릴 수 있습니다.
- 인스턴스를 선택한 상태에서 Actions, Instance Settings, Change Instance Type를 선택합니다. 인스턴스가 stopped 상태가 아닐 경우에는 이 작업을 수행할 수 없습니다.
- 인스턴스 유형 변경 대화 상자에서 다음을 수행합니다.
 - 인스턴스 유형에서 원하는 인스턴스 유형을 선택합니다. 원하는 인스턴스 유형이 목록에 없으면 해당 인스턴스의 구성과 호환되지 않는 것입니다. 예를 들어 가상화 유형 때문일 수 있습니다. 자세한 내용은 [인스턴스 크기 조정을 위한 호환성 \(p. 195\)](#) 단원을 참조하십시오.
 - (선택 사항) 선택한 인스턴스 유형이 EBS- 최적화를 지원할 경우 EBS 최적을 선택하여 EBS- 최적화를 활성화하거나, EBS 최적의 선택을 취소하여 EBS- 최적화를 비활성화합니다. 선택한 인스턴스 유형이 기본적으로 EBS에 최적화되었을 경우 EBS 최적이 선택되고 이를 선택 취소할 수 없습니다.
 - 적용을 선택하여 새로운 설정을 승인합니다.
- 중지된 인스턴스를 다시 시작하려면 인스턴스를 선택하고 Actions, Instance State, Start를 선택합니다.
- 확인 대화 상자가 나타나면 예, 시작을 선택합니다. 인스턴스가 running 상태가 되는 데 몇 분 정도 걸릴 수 있습니다.

새 인스턴스 구성으로 마이그레이션

인스턴스의 현재 구성이 새 인스턴스 유형과 호환되지 않을 경우, 인스턴스를 해당 인스턴스 유형으로 크기 조정할 수 없습니다. 대신 새 인스턴스 유형과 호환되는 구성을 가진 새 인스턴스로 애플리케이션을 마이그레이션할 수 있습니다.

애플리케이션을 호환되는 인스턴스로 마이그레이션하려면

- 영구 스토리지를 유지해야 할 인스턴스 스토어 볼륨에 데이터를 백업합니다. 유지해야 하는 EBS 볼륨에 데이터를 마이그레이션하려면 볼륨의 스냅샷을 생성하거나([Amazon EBS 스냅샷 생성 \(p. 969\)](#) 참조), 나중에 새 인스턴스에 연결할 수 있도록 인스턴스에서 볼륨을 분리합니다([Windows 인스턴스에서 Amazon EBS 볼륨 분리 \(p. 964\)](#) 참조).
- 다음을 선택하여 새 인스턴스를 시작합니다.
 - 탄력적 IP 주소를 사용할 경우 원래 인스턴스가 현재 실행 중인 VPC를 선택합니다.
 - 원래 인스턴스에서 분리하여 새 인스턴스에 연결하려는 EBS 볼륨 또는 생성한 스냅샷에 기반한 새로운 EBS 볼륨
 - 새 인스턴스로 동일한 트래픽을 허용하려는 경우 원래 인스턴스와 연결된 보안 그룹을 선택합니다.
- 애플리케이션과 기타 필요한 소프트웨어를 인스턴스에 설치합니다.
- 원래 인스턴스의 인스턴스 스토어 볼륨에서 백업한 데이터를 복원합니다.
- 탄력적 IP 주소를 사용할 경우 다음과 같이 이 주소를 새로 시작한 인스턴스에 지정합니다.
 - 탐색 창에서 탄력적 IP를 선택합니다.
 - 원래 인스턴스와 연결된 탄력적 IP 주소를 선택하고 작업, 주소 연결 해제를 차례로 선택합니다. 확인 메시지가 나타나면 주소 연결 해제를 선택합니다.
 - 탄력적 IP 주소를 선택한 상태에서 작업, 주소 연결을 차례로 선택합니다.

- d. 인스턴스에서 새 인스턴스를 선택한 후 연결을 선택합니다.
6. (선택 사항) 원래 인스턴스가 더 이상 필요하지 않을 경우 이를 종료할 수 있습니다. 인스턴스를 선택하고 새 인스턴스가 아닌 원래 인스턴스를 종료하고 있는지 확인합니다. 예를 들어 이름이나 시작 시간을 확인합니다. 작업, 인스턴스 상태, 종료를 차례로 선택합니다.

인스턴스 유형에 대한 권장 사항 가져오기

AWS Compute Optimizer는 성능 향상, 비용 절감 또는 두 가지 모두를 위한 Amazon EC2 인스턴스 권장 사항을 제공합니다. 이러한 권장 사항을 사용하여 새 인스턴스 유형으로 이동할지 여부를 결정할 수 있습니다.

권장 사항을 만들기 위해 Compute Optimizer는 기존 인스턴스 사양과 사용률 지표를 분석합니다. 그런 다음 컴파일된 데이터를 사용하여 기존 워크로드를 처리하는 데 가장 적합한 Amazon EC2 인스턴스 유형을 권장합니다. 권장 사항은 시간당 인스턴스 요금과 함께 반환됩니다.

이 주제에서는 Amazon EC2 콘솔을 통해 권장 사항을 보는 방법에 대해 간략하게 설명합니다. 자세한 내용은 [AWS Compute Optimizer 사용 설명서](#) 단원을 참조하십시오.

Note

Compute Optimizer에서 권장 사항을 받으려면 먼저 Compute Optimizer를 옵트인해야 합니다. 자세한 내용은 AWS Compute Optimizer 사용 설명서의 [AWS Compute Optimizer 시작하기](#)를 참조하십시오.

목차

- [제한 사항 \(p. 198\)](#)
- [결과 \(p. 198\)](#)
- [권장 사항 보기 \(p. 199\)](#)
- [권장 사항 평가를 위한 고려 사항 \(p. 200\)](#)

제한 사항

Compute Optimizer는 현재 M, C, R, T 및 X 인스턴스 유형에 대한 권장 사항을 생성합니다. 다른 인스턴스 유형은 Compute Optimizer의 대상이 아닙니다. 다른 인스턴스 유형을 사용하는 경우, Compute Optimizer 권장 사항 보기에서 나타나지 않습니다. 이 인스턴스 유형 및 다른 인스턴스 유형에 대한 자세한 내용은 [인스턴스 유형 \(p. 114\)](#) 단원을 참조하십시오.

결과

Compute Optimizer는 EC2 인스턴스에 대한 결과를 다음과 같이 분류합니다.

- Under-provisioned(프로비저닝 부족) – 적어도 하나의 인스턴스 사양(CPU, 메모리, 네트워크 등)이 워크로드의 성능 요구 사항을 충족하지 않을 때 EC2 인스턴스가 부족하게 프로비저닝되었다고 봅니다. 프로비저닝 부족 EC2 인스턴스는 애플리케이션 성능 저하를 불러올 수 있습니다.
- Over-provisioned(프로비저닝 과다) – 워크로드 성능 요구 사항을 충족하면서 적어도 하나의 인스턴스 사양(CPU, 메모리, 네트워크 등)을 줄일 수 있고, 부족하게 프로비저닝된 사양이 없을 때 EC2 인스턴스가 과하게 프로비저닝되었다고 봅니다. 프로비저닝 과다 EC2 인스턴스는 불필요한 인프라 비용을 유발할 수 있습니다.
- Optimized(최적화) – CPU, 메모리, 네트워크 등 모든 인스턴스 사양이 워크로드의 성능 요구 사항을 충족하고 인스턴스가 과다 프로비저닝되지 않았을 때 EC2 인스턴스가 최적화된 것으로 봅니다. EC2 인스턴스가 최적화되면 최적의 성능과 인프라 비용으로 워크로드가 실행됩니다. 최적화된 인스턴스의 경우 Compute Optimizer가 차세대 인스턴스 유형을 권장하기도 합니다.
- 없음 – 이 인스턴스의 권장 사항이 없습니다. Compute Optimizer를 옵트인한 지 12시간이 지나지 않았거나, 인스턴스가 실행된 지 30시간이 지나지 않았거나, Compute Optimizer에서 지원하지 않는 인스턴스 유

형인 경우 이 문제가 생길 수 있습니다. 자세한 내용은 이전 섹션의 [제한 사항 \(p. 198\)](#) 단원을 참조하십시오.

권장 사항 보기

Compute Optimizer를 옵트인하면 EC2 콘솔에서 Compute Optimizer가 EC2 인스턴스에 대해 생성한 결과를 볼 수 있습니다. 그런 다음 Compute Optimizer 콘솔에 액세스하여 권장 사항을 볼 수 있습니다. 최근에 옵트인한 경우 검색 결과가 EC2 콘솔에 최대 12시간 동안 반영되지 않을 수 있습니다.

EC2 콘솔을 통해 EC2 인스턴스의 권장 사항을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 설명 탭에서 Finding(결과) 필드를 살펴봅니다. 세부 정보 보기를 선택합니다.

Compute Optimizer에서 인스턴스가 열리고 여기서 현재 인스턴스로 레이블이 지정됩니다. 서로 다른 인스턴스 유형 권장 사항이 Option 1(옵션 1), Option 2(옵션 2), Option 3(옵션 3)이라는 레이블로 3개까지 제공됩니다. 창 아래쪽에 현재 인스턴스의 최근 CloudWatch 지표 데이터인 CPU 사용률, 메모리 사용률, 네트워크 입력, 네트워크 출력력이 표시됩니다.

4. (선택 사항) Compute Optimizer 콘솔에서 설정() 아이콘을 선택하여 테이블에 표시된 열을 변경하거나 현재 및 권장 인스턴스 유형의 여러 가지 구매 옵션에 대해 공개된 요금 정보를 볼 수 있습니다.

Note

예약 인스턴스를 구매한 경우 온디맨드 인스턴스 요금이 예약 인스턴스로 청구될 수 있습니다. 현재 인스턴스 유형을 변경하기 전에 먼저 예약 인스턴스 사용률 및 적용 범위에 미치는 영향을 평가합니다.

권장 사항 중 하나를 사용할지 여부를 결정합니다. 성능 향상, 비용 절감 또는 이 두 가지를 조합하여 최적화 할 것인지 결정합니다. 자세한 내용은 AWS Compute Optimizer 사용 설명서의 [리소스 권장 사항 보기](#)를 참조하십시오.

Compute Optimizer 콘솔을 통해 모든 리전의 모든 EC2 인스턴스에 대한 권장 사항을 보려면

1. <https://console.aws.amazon.com/compute-optimizer/>에서 Compute Optimizer 콘솔을 엽니다.
 2. View recommendations for all EC2 instances(모든 EC2 인스턴스에 대한 권장 사항 보기)를 선택합니다.
 3. 권장 사항 페이지에서 다음 작업을 수행할 수 있습니다.
 - a. 하나 이상의 AWS 리전으로 권장 사항을 필터링하려면 Filter by one or more Regions(하나 이상의 리전별 필터링) 텍스트 상자에 리전 이름을 입력하거나, 표시되는 드롭다운 목록에서 리전을 하나 이상 선택합니다.
 - b. 다른 계정의 리소스에 대한 권장 사항을 보려면 계정을 선택한 다음 다른 계정 ID를 선택합니다.
- 이 옵션은 조직의 마스터 계정에 로그인하고 조직 내의 모든 구성원 계정을 옵트인한 경우에만 사용 할 수 있습니다.
- c. 선택한 필터를 지우려면 Clear filters(필터 지우기)를 선택합니다.
 - d. 현재 및 권장 인스턴스 유형에 대해 표시되는 구매 옵션을 변경하려면 설정() 아이콘을 선택한 다음 On-Demand Instances(온디맨드 인스턴스), Reserved Instances, standard 1-year no upfront(예약 인스턴스, 표준 1년 선결제 없음) 또는 Reserved Instances, standard 3-year no upfront(예약 인스턴스, 표준 3년 선결제 없음)를 선택합니다.
 - e. 추가 권장 사항 및 사용률 비교와 같은 세부 정보를 보려면 원하는 인스턴스 옆에 나열된 결과(Under-provisioned(프로비저닝 부족), Over-provisioned(프로비저닝 과다) 또는 Optimized(최적화))

를 선택합니다. 자세한 내용은 AWS Compute Optimizer 사용 설명서의 [리소스 세부 정보 보기](#)를 참조하십시오.

권장 사항 평가를 위한 고려 사항

인스턴스 유형을 변경하기 전에 다음 사항을 고려하십시오.

- 권장 사항은 사용량을 예측하지 않습니다. 권장 사항은 최근 14일 기간 동안의 사용량을 기준으로 합니다. 향후 리소스 요구 사항을 충족할 것으로 예상되는 인스턴스 유형을 선택해야 합니다.
- 그래프로 표시된 지표를 집중적으로 살펴보고 실제 사용량이 인스턴스 용량보다 낮은지 확인합니다. 또한 CloudWatch에서 지표 데이터(평균, 피크, 백분위수)를 보고 EC2 인스턴스 권장 사항을 추가로 평가할 수 있습니다. 예를 들어 CPU 백분율 지표가 하루 동안 어떻게 변화하고 수용해야 하는 피크가 있는지 확인합니다. 자세한 내용은 Amazon CloudWatch 사용 설명서의 [사용 가능한 지표 보기](#)를 참조하십시오.
- Compute Optimizer는 T3, T3a, T2 인스턴스 등 성능 버스트가 가능한 인스턴스에 대한 권장 사항을 제공할 수 있습니다. 기준 이상으로 주기적으로 버스트하는 경우 새 인스턴스 유형의 vCPU에 따라 계속 버스트할 수 있어야 합니다. 자세한 정보는 [성능 순간 확장 가능 인스턴스에 대한 CPU 크레딧 및 기준 사용률](#) (p. 130) 단원을 참조하십시오.
- 예약 인스턴스를 구매한 경우 온디맨드 인스턴스 요금이 예약 인스턴스로 청구될 수 있습니다. 현재 인스턴스 유형을 변경하기 전에 먼저 예약 인스턴스 사용률 및 적용 범위에 미치는 영향을 평가합니다.
- 가능한 경우 최신 세대 인스턴스로의 변환을 고려합니다.
- 다른 인스턴스 패밀리로 마이그레이션할 때 현재 인스턴스 유형과 새 인스턴스 유형이 가상화, 아키텍처 또는 네트워크 유형 측면에서 호환되어야 합니다. 자세한 내용은 [인스턴스 크기 조정을 위한 호환성](#) (p. 195) 단원을 참조하십시오.
- 마지막으로 각 권장 사항에 대해 제공되는 성능 위험 등급을 고려합니다. 성능 위험은 권장 인스턴스 유형이 워크로드의 성능 요구 사항을 충족하는지 여부를 검증하기 위해 얼마나 많은 노력을 기울여야 하는지를 나타냅니다. 또한 변경 전후에 엄격한 로드 및 성능 테스트를 수행하는 것이 좋습니다.

EC2 인스턴스 크기를 조정할 때 고려할 다른 내용도 있습니다. 자세한 내용은 [인스턴스 유형 변경](#) (p. 194) 단원을 참조하십시오.

추가 리소스

- [인스턴스 유형](#) (p. 114)
- [AWS Compute Optimizer 사용 설명서](#)

인스턴스 구입 옵션

Amazon EC2는 사용자가 요구 사항에 따라 비용을 최적화할 수 있도록 다음과 같은 구입 옵션을 제공합니다.

- 온디맨드 인스턴스 - 시작하는 인스턴스에 대한 비용을 시간 단위로 지불합니다.
- Savings Plans - 1년 또는 3년 기간 동안 시간당 USD로 일관된 사용량을 약정하여 Amazon EC2 비용을 절감할 수 있습니다.
- 예약 인스턴스 - 1년 또는 3년 기간 동안 인스턴스 유형 또는 지역을 포함해 일관된 인스턴스 구성의 Amazon EC2 비용을 절감할 수 있습니다.
- 정기 예약 인스턴스 - 1년 동안 항상 사용할 수 있는 인스턴스를 지정된 되풀이 일정으로 구입합니다.
- 스팟 인스턴스 - 미사용 EC2 인스턴스를 요청하여 Amazon EC2 비용을 대폭 줄일 수 있습니다.
- 전용 호스트 - 인스턴스 실행을 전담하는 실제 호스트 비용을 지불하며, 기존의 소켓, 코어 또는 VM 소프트웨어별 라이선스를 가져와 비용을 절감합니다.
- 전용 인스턴스 - 단일 테넌트 하드웨어에서 실행되는 인스턴스 비용을 시간 단위로 지불합니다.

- 용량 예약 – 원하는 기간 동안 특정 가용 영역의 EC2 인스턴스에 대해 용량을 예약합니다.

용량을 예약해야 할 경우 특정 가용 영역에 대해 예약 인스턴스 또는 용량 예약을 구입하거나 정기 예약 인스턴스를 구입하십시오. 스팟 인스턴스는 애플리케이션이 실행할 때와 중단될 수 있는 때를 유연하게 조정할 수 있는 경우 선택 가능한 비용 효율적인 방법입니다. 전용 호스트 또는 전용 인스턴스를 사용하면 기존 서버 별 소프트웨어 라이선스를 사용하여 규정 준수 요구 사항을 해결하고 비용을 줄일 수 있습니다. 자세한 내용은 [Amazon EC2 요금](#)을 참조하십시오.

Savings Plans에 대한 자세한 내용은 [AWS Savings Plans 사용 설명서](#)를 참조하십시오.

목차

- [인스턴스 수명 주기 결정 \(p. 201\)](#)
- [온디맨드 인스턴스 \(p. 202\)](#)
- [예약 인스턴스 \(p. 205\)](#)
- [정기 예약 인스턴스 \(p. 236\)](#)
- [스팟 인스턴스 \(p. 239\)](#)
- [전용 호스트 \(p. 320\)](#)
- [전용 인스턴스 \(p. 349\)](#)
- [온디맨드 용량 예약 \(p. 354\)](#)

인스턴스 수명 주기 결정

인스턴스의 수명 주기는 인스턴스가 시작되어 종료될 때까지입니다. 선택한 구매 옵션이 인스턴스의 수명 주기에 영향을 미칩니다. 예를 들어 온디맨드 인스턴스는 사용자가 인스턴스를 시작하면 실행되고 종료하면 끝납니다. 스팟 인스턴스는 용량이 가용 상태이고, 최고 가격이 스팟 가격보다 더 높은 조건 하에서만 실행됩니다. 예약된 시간 동안에 정기 예약 인스턴스를 시작할 수 있습니다. 예를 들어 Amazon EC2는 인스턴스를 시작하고 나서 예약 시간이 종료되기 3분 전에 종료됩니다.

다음 절차를 사용하여 인스턴스의 수명 주기를 결정합니다.

콘솔을 사용하여 인스턴스 수명 주기를 결정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다.
4. 설명 탭에서 테넌시를 찾습니다. 값이 host인 경우 그 인스턴스는 전용 호스트에서 실행 중인 것입니다. 값이 dedicated인 경우 인스턴스는 전용 인스턴스입니다.
5. 설명 탭에서 수명 주기를 찾습니다. 값이 spot인 경우 인스턴스는 스팟 인스턴스입니다. 값이 scheduled인 경우 그 인스턴스는 정기 예약 인스턴스입니다. 값이 normal인 경우 인스턴스는 온디맨드 인스턴스 또는 예약 인스턴스입니다.
6. (선택 사항) 예약 인스턴스를 구매했는데 적용되었는지 확인하고 싶은 경우 Amazon EC2에 대한 사용 보고서를 참고할 수 있습니다. 자세한 내용은 [Amazon EC2 사용 보고서 \(p. 1156\)](#) 단원을 참조하십시오.

AWS CLI를 사용하여 인스턴스 수명 주기를 결정하려면

아래와 같이 `describe-instances` 명령을 사용합니다.

```
aws ec2 describe-instances --instance-ids i-1234567890abcdef0
```

인스턴스가 전용 호스트에서 실행 중인 경우 출력에는 다음 정보가 포함됩니다.

```
"Tenancy": "host"
```

인스턴스가 전용 인스턴스인 경우 출력에는 다음 정보가 포함됩니다.

```
"Tenancy": "dedicated"
```

인스턴스가 스팟 인스턴스인 경우 출력에는 다음 정보가 포함됩니다.

```
"InstanceLifecycle": "spot"
```

인스턴스가 정기 예약 인스턴스인 경우, 출력에는 다음 정보가 포함됩니다.

```
"InstanceLifecycle": "scheduled"
```

그 외 경우에는 출력에 `InstanceLifecycle`이 포함되지 않습니다.

온디맨드 인스턴스

온디맨드 인스턴스를 사용하면 장기 약정 없이 시간 단위로 컴퓨팅 용량을 구입할 수 있습니다. 수명 주기를 완전하게 제어할 수 있습니다. 즉 시작, 중지, 수면, 사용 시작 또는 종료 시기를 결정할 수 있습니다.

온디맨드 인스턴스를 구매할 때 장기 약정은 필요 없습니다. 온디맨드 인스턴스가 `running` 상태인 시간에 대해서만 지불하면 됩니다. 실행 중인 온디맨드 인스턴스에 대한 시간당 요금은 고정 요금이며, [Amazon EC2 요금](#), [온디맨드 요금 페이지](#)에서 확인할 수 있습니다.

중단할 수 없는 불규칙한 단기 워크로드가 있는 애플리케이션의 경우 온디맨드 인스턴스를 사용하는 것이 좋습니다.

온디맨드 인스턴스를 통해 비용을 대폭 절감하려면 [AWS Savings Plans](#), [스팟 인스턴스 \(p. 239\)](#) 또는 [예약 인스턴스 \(p. 205\)](#)를 사용합니다.

목차

- [온디맨드 인스턴스 작업 \(p. 202\)](#)
- [온디맨드 인스턴스 제한 \(p. 203\)](#)
 - [필요한 vCPU 수 계산 \(p. 203\)](#)
 - [제한 증가 요청 \(p. 205\)](#)
 - [온디맨드 인스턴스 제한 및 사용량 모니터링 \(p. 205\)](#)
- [AWS 서비스 요금 쿼리 \(p. 205\)](#)

온디맨드 인스턴스 작업

다음과 같은 방식으로 온디맨드 인스턴스 작업을 수행할 수 있습니다.

- [인스턴스 시작 \(p. 375\)](#)
- [Windows 인스턴스에 연결 \(p. 436\)](#)
- [인스턴스 중지 및 시작 \(p. 440\)](#)
- [Windows 인스턴스를 최대 절전 모드로 실행 \(p. 443\)](#)

- [인스턴스 재부팅 \(p. 452\)](#)
- [인스턴스 만료 \(p. 452\)](#)
- [인스턴스 종료 \(p. 454\)](#)
- [인스턴스 복구 \(p. 460\)](#)
- [Windows 인스턴스 구성 \(p. 461\)](#)
- [EC2 Windows 인스턴스 식별 \(p. 634\)](#)

Amazon EC2를 처음 사용한다면 먼저 [Amazon EC2 시작 방법 \(p. 1\)](#) 단원을 참조하십시오.

온디맨드 인스턴스 제한

리전별로 AWS 계정당 온디맨드 인스턴스 실행 수에는 제한이 있습니다. 온디맨드 인스턴스 제한은 인스턴스 유형과 상관없이 실행 중인 온디맨드 인스턴스가 사용하는 가상 중앙 처리 장치(vCPU) 수를 기준으로 관리됩니다.

다음 표와 같은 다섯 가지 온디맨드 인스턴스 제한이 있습니다. 각 제한은 하나 이상의 인스턴스 패밀리에 vCPU 제한을 지정합니다. 다른 인스턴스 패밀리에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#) 단원을 참조하십시오.

온디맨드 인스턴스 제한 이름	기본 vCPU 제한
온디맨드 표준(A, C, D, H, I, M, R, T, Z) 인스턴스 실행	vCPU 1152개
온디맨드 F 인스턴스 실행	vCPU 128개
온디맨드 G 인스턴스 실행	vCPU 128개
온디맨드 Inf 인스턴스 실행	vCPU 128개
온디맨드 P 인스턴스 실행	vCPU 128개
온디맨드 X 인스턴스 실행	vCPU 128개

Note

새 AWS 계정의 경우 여기에 설명된 것보다 더 낮은 수로 제한하여 시작할 수 있습니다.

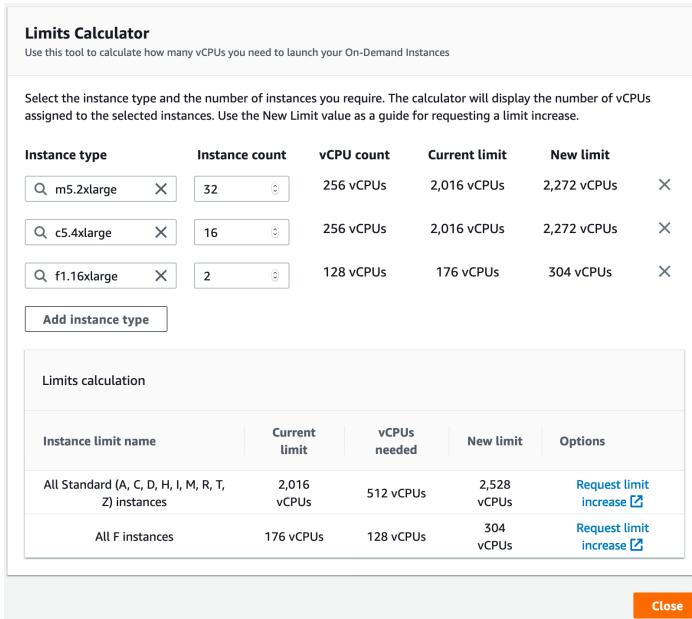
vCPU 제한을 사용하면 변화하는 애플리케이션 요구 사항을 충족하는 모든 인스턴스 유형 조합을 시작하는데 필요한 vCPU 수에 따라 제한을 사용할 수 있습니다. 예를 들어 표준 인스턴스 제한이 vCPU 256개인 경우 32개의 m5.2xlarge 인스턴스(32x8 vCPU), 16개의 c5.4xlarge 인스턴스(16x16 vCPU) 또는 표준 인스턴스 유형 및 크기를 총 256개의 vCPU로 맞춘 조합을 시작할 수 있습니다. 자세한 내용은 [EC2 온디맨드 인스턴스 제한](#) 단원을 참조하십시오.

필요한 vCPU 수 계산

vCPU 제한 계산기를 사용하여 애플리케이션 요구에 맞게 필요한 vCPU 수를 결정할 수 있습니다.

계산기를 사용할 때는 다음 사항에 주의하십시오. 계산기는 현재 제한에 도달했다고 가정합니다. 인스턴스 수(Instance count)에 입력하는 값은 현재 제한에서 허용되는 인스턴스에 추가하여 시작해야 하는 인스턴스 수입니다. 계산기는 현재 제한을 인스턴스 수(Instance count)에 추가하여 새로운 제한에 도달합니다.

다음 스크린샷은 vCPU 제한 계산기를 보여줍니다.



다음 컨트롤과 정보를 확인하고 사용할 수 있습니다.

- 인스턴스 유형 – vCPU 제한 계산기에 추가하는 인스턴스 유형입니다.
- 인스턴스 수 – 선택한 유형에 필요한 인스턴스 수입니다.
- vCPU 수 – 인스턴스 수에 해당하는 vCPU 수입니다.
- 현재 제한 – 인스턴스 유형이 속한 제한 유형의 현재 제한입니다. 제한은 제한 유형이 동일한 모든 인스턴스 유형에 적용됩니다. 예를 들어 위의 스크린샷에서 m5.2xlarge 및 c5.4xlarge에 대한 현재 제한은 vCPU 1,920개이며, 모든 표준 인스턴스 제한에 속한 모든 인스턴스 유형에 대한 제한이 됩니다.
- 새 제한 – vCPU 수와 현재 제한을 추가하여 계산되는 새로운 제한(vCPU 수)입니다.
- X – 행을 제거하려면 X를 선택합니다.
- Add instance type(인스턴스 유형 추가) – 계산기에 다른 인스턴스 유형을 추가하려면 Add instance type(인스턴스 유형 추가)을 선택합니다.
- Limits calculation(제한 계산) – 현재 제한, 필요한 vCPU 수 및 제한 유형에 대한 새 제한을 표시합니다.
 - Instance limit name(인스턴스 제한 이름) – 선택한 인스턴스 유형에 대한 제한 유형입니다.
 - 현재 제한 – 제한 유형에 대한 현재 제한입니다.
 - 필요한 vCPU 수 – 인스턴스 수에서 지정한 인스턴스 수에 해당하는 vCPU 수입니다. 모든 표준 인스턴스 제한에 필요한 vCPU 수는 이 제한 유형의 모든 인스턴스 유형에 대한 vCPU count(vCPU 수) 값을 더해 계산됩니다.
 - 새 제한 – 새로운 제한은 현재 제한과 필요한 vCPU 수를 추가하여 계산됩니다.
 - 옵션 – 해당하는 제한 유형에 대한 제한 증가를 요청하려면 사용량 제한 증가 요청을 선택합니다.

필수 vCPU 수를 계산하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 리전을 선택합니다.
3. 왼쪽 탐색기에서 제한을 선택합니다.
4. Calculate vCPU limit(vCPU 제한 계산)을 선택합니다.
5. Add instance type(인스턴스 유형 추가)을 선택하고, 필수 인스턴스 유형을 선택하고, 필수 인스턴스 수를 지정합니다. 인스턴스 유형을 더 추가하려면 Add instance type(인스턴스 유형 추가)를 다시 선택합니다.

6. 필수 새 제한에 대한 Limits calculation(제한 계산)을 확인합니다.
7. 계산기를 사용하여 종료한 경우 닫기를 선택합니다.

제한 증가 요청

[Limits page\(제한 페이지\)](#) 또는 Amazon EC2 콘솔의 vCPU 제한 계산기에서 각 온디맨드 인스턴스 제한 유형에 대한 제한 증가를 요청할 수 있습니다. 사용 사례를 활용하여 AWS 지원 센터의 [제한 증가 양식](#)에서 필수 필드를 입력하면 됩니다. Primary Instance Type(기본 인스턴스 유형)에서 vCPU 제한 계산기의 Instance limit name(인스턴스 제한 이름)에 상응하는 제한 유형을 선택합니다. 새 제한 값에는 vCPU 제한 계산기의 New limit(새 제한) 열에 표시되는 값을 사용합니다. 제한 증가 요청에 대한 자세한 내용은 [Amazon EC2 서비스 할당량 \(p. 1154\)](#) 단원을 참조하십시오.

온디맨드 인스턴스 제한 및 사용량 모니터링

다음을 사용하여 온디맨드 인스턴스 제한을 보고 관리할 수 있습니다.

- Amazon EC2 콘솔의 [제한 페이지](#)
- 서비스 할당량 콘솔의 [Amazon EC2 서비스 할당량 페이지](#)
- `get-service-quota` AWS CLI
- AWS Trusted Advisor 콘솔의 [서비스 제한 페이지](#)

자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2 서비스 할당량 \(p. 1154\)](#), 서비스 할당량 사용 설명서의 [서비스 할당량 확인](#), [AWS Trusted Advisor](#)를 참조하십시오.

Amazon CloudWatch 지표를 통해 제한에 대한 EC2 사용량을 확인할 수 있습니다. 제한 도달에 대해 경고를 받도록 경보를 구성할 수도 있습니다. 자세한 내용은 서비스 할당량 사용 설명서의 [Amazon CloudWatch 경보 사용](#)을 참조하십시오.

AWS 서비스 요금 쿼리

Price List Service API 또는 AWS Price List API를 사용하여 온디맨드 인스턴스 요금을 쿼리할 수 있습니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 [AWS Price List API 사용](#)을 참조하십시오.

예약 인스턴스

예약 인스턴스를 이용하면 온디맨드 인스턴스 요금에 비해 Amazon EC2 비용을 상당히 절감할 수 있습니다. 예약 인스턴스는 물리적 인스턴스가 아니며 계정에서 온디맨드 인스턴스를 사용할 때 적용되는 결제 할인에 가깝습니다. 이러한 온디맨드 인스턴스의 경우 결제 할인 혜택을 받으려면 인스턴스 유형 및 지역과 같은 특정 속성에 부합해야 합니다.

또한 Savings Plans는 온디맨드 인스턴스 요금에 비해 상당한 Amazon EC2 비용 절감을 제공합니다. Savings Plans를 이용해 시간당 USD로 측정되는 일관된 사용량을 약정할 수 있습니다. Savings Plans는 특정 인스턴스 구성을 약정하는 것이 아니라 사용자의 요구 사항에 가장 적합하고 지속적으로 비용을 절약해 주는 인스턴스 구성을 사용할 수 있는 유연성을 제공합니다. 자세한 내용은 [AWS Savings Plans 사용 설명서](#)를 참조하십시오.

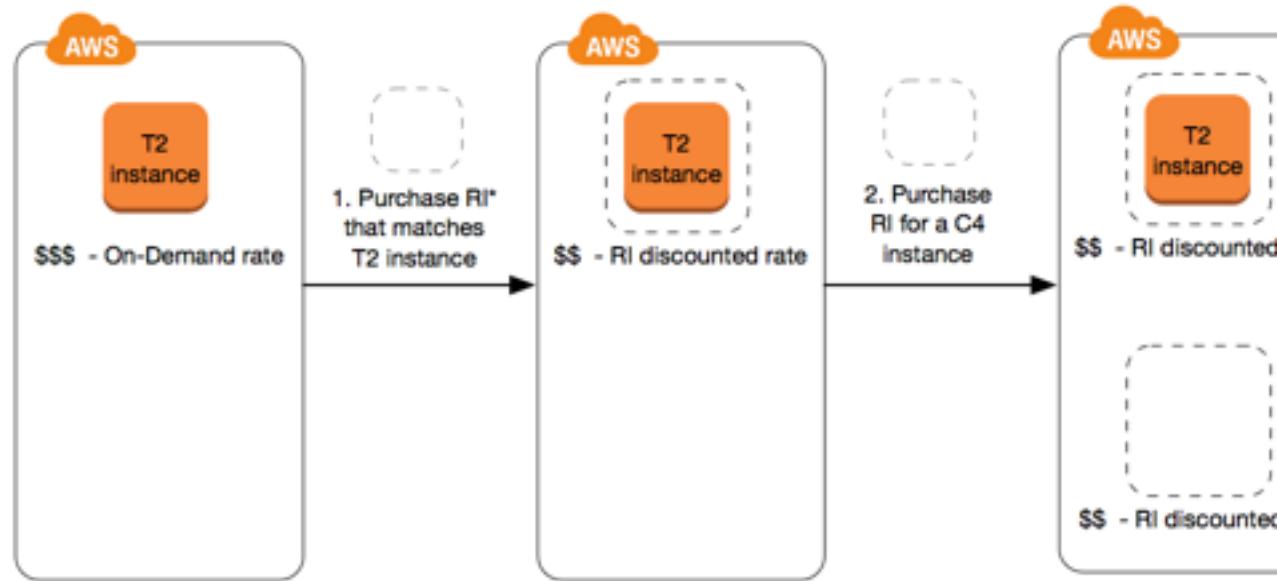
예약 인스턴스 주제

- [예약 인스턴스 개요 \(p. 206\)](#)
- [예약 인스턴스 요금을 결정하는 주요 변수 \(p. 206\)](#)
- [예약 인스턴스 제한 \(p. 207\)](#)
- [리전 및 영역 예약 인스턴스\(범위\) \(p. 208\)](#)
- [예약 인스턴스 유형\(제공 클래스\) \(p. 209\)](#)

- 예약 인스턴스 적용 방식 (p. 209)
- 요금 부과 방법 (p. 214)
- 예약 인스턴스 구입 (p. 217)
- 예약 인스턴스 마켓플레이스 (p. 223)
- 예약 인스턴스 수정 (p. 229)
- 전환형 예약 인스턴스 교환 (p. 232)

예약 인스턴스 개요

다음 다이어그램에는 예약 인스턴스 구입 및 사용에 대한 기본 개요가 나와 있습니다.



*RI = Reserved Instance

이 시나리오에서, 현재 온디맨드 요금으로 지불하고 있는 온디맨드 인스턴스(T2)가 사용자의 계정에서 실행되고 있습니다. 사용자는 실행되고 있는 인스턴스의 속성과 일치하는 예약 인스턴스를 구입하며, 결제 혜택이 즉시 적용됩니다. 그런 다음 C4 인스턴스에 대한 예약 인스턴스를 구입합니다. 이 예약 인스턴스의 속성과 일치하는 인스턴스가 계정에서 실행되고 있지 않습니다;. 최종 단계에서 사용자는 C4 예약 인스턴스의 속성과 일치하는 인스턴스를 시작하며, 결제 혜택이 즉시 적용됩니다.

예약 인스턴스 요금을 결정하는 주요 변수

예약 인스턴스 요금은 다음과 같은 주요 변수에 의해 결정됩니다.

인스턴스 속성

예약 인스턴스에는 요금을 결정하는 4개의 인스턴스 속성이 있습니다.

- 인스턴스 유형: 예를 들어, m4.large입니다. 이는 인스턴스 패밀리(예: m4)와 인스턴스 크기(예: large)로 구성됩니다.
- 리전: 예약 인스턴스를 구매한 리전입니다.
- 테넌시: 인스턴스가 공유된 하드웨어(기본)에서 실행되는지 단일 테넌트(전용) 하드웨어에서 실행되는지 여부입니다. 자세한 내용은 [전용 인스턴스 \(p. 349\)](#) 단원을 참조하십시오.

- **플랫폼:** Windows 또는 Linux/Unix와 같은 운영 체제입니다. 자세한 내용은 [플랫폼 선택 \(p. 218\)](#) 단원을 참조하십시오.

기간 약정

1년 약정 또는 더 큰 할인을 제공하는 3년 약정으로 예약 인스턴스를 구입할 수 있습니다.

- 1년: 1년은 31536000초(365일)로 정의됩니다.
- 3년: 3년은 94608000초(1095일)로 정의됩니다.

예약 인스턴스는 자동으로 갱신되지 않으므로 만료될 경우 중단 없이 EC2 인스턴스를 계속 사용할 수 있지만 온디맨드 요금이 부과됩니다. 위 예에서 T2 및 C4 인스턴스에 적용되는 예약 인스턴스가 만료되면 사용자가 인스턴스를 종료하거나 인스턴스 속성과 일치하는 새 예약 인스턴스를 구입할 때까지 온디맨드 요금 결제로 돌아갑니다.

결제 옵션

예약 인스턴스에 사용할 수 있는 결제 옵션은 다음과 같습니다.

- **전체 선결제:** 기간이 시작되는 시점에서 모든 금액을 결제하고 사용 기간 동안 기타 비용이나 추가 시간당 요금 없이 무제한으로 사용할 수 있습니다.
- **부분 선결제:** 비용 중 일부를 먼저 결제해야 하며, 결제하지 않은 시간에 대해서는 예약 인스턴스가 사용되는지 여부와 상관없이 할인된 시간당 요금이 청구됩니다.
- **선결제 없음:** 예약 인스턴스가 사용되는지 여부와 상관없이 사용 기간 동안 매시간마다 할인된 시간당 요금이 청구됩니다. 선결제 금액이 필요하지 않습니다.

Note

선결제가 없는 예약 인스턴스는 전체 예약 기간 동안 매월 결제해야 하는 계약 조건입니다. 따라서 선결제가 없는 예약 인스턴스를 구입할 수 있으려면 결제 기록에 미납액이 없어야 합니다.

일반적으로 예약 인스턴스에 대한 선결제 금액이 높을수록 요금 절약 혜택이 커집니다. 또한 예약 인스턴스 마켓플레이스에서 타사 판매업체가 제공하는 저렴하고 기간이 짧은 예약 인스턴스를 찾을 수도 있습니다. 자세한 정보는 [예약 인스턴스 마켓플레이스 \(p. 223\)](#) 단원을 참조하십시오.

제공 클래스

컴퓨팅 요건이 변경되면 제공 클래스에 따라 예약 인스턴스를 수정 또는 교환할 수 있습니다.

- **표준:** 가장 큰 할인 혜택을 제공하지만 수정만 가능합니다.
- **컨버터블:** 표준 예약 인스턴스보다 낮은 할인 혜택을 제공하지만 다른 인스턴스 속성을 포함하는 다른 컨버터블 예약 인스턴스와 교환 가능합니다. 컨버터블 예약 인스턴스는 수정도 가능합니다.

자세한 내용은 [예약 인스턴스 유형\(제공 클래스\) \(p. 209\)](#)을 참조하십시오.

예약 인스턴스를 구입한 이후에는 구입을 취소할 수 없습니다. 그러나 변경이 필요한 경우 예약 인스턴스를 수정 ([p. 229](#)), 교환 ([p. 232](#)) 또는 판매 ([p. 223](#)) 할 수 있습니다.

자세한 내용은 [Amazon EC2 예약 인스턴스 요금 페이지](#)를 참조하세요.

예약 인스턴스 제한

조직에서 보유할 수 있는 예약 인스턴스 수는 제한되어 있습니다. 각 리전에 대해 20개의 [리전 \(p. 209\)](#) 예약 인스턴스와, 각 가용 영역에 대해 달마다 추가로 20개의 [영역 \(p. 209\)](#) 예약 인스턴스를 구입할 수 있습니다.

예를 들어 가용 영역이 3개인 리전에서 한도는 월당 예약 인스턴스 80개입니다. 즉, 리전에 대해 리전 예약 인스턴스 20개와, 각 가용 영역 3개에 대해 영역 예약 인스턴스 20개($20 \times 3 = 60$)입니다.

리전 예약 인스턴스는 실행 중인 온디맨드 인스턴스에 할인을 적용합니다. 기본 온디맨드 인스턴스 제한은 20개입니다. 리전 예약 인스턴스 구매로 실행 중인 온디맨드 인스턴스 제한을 초과할 수는 없습니다. 예를 들어 이미 20개의 온디맨드 인스턴스를 실행 중이고 20개의 리전 예약 인스턴스를 구매한 경우 20개의 리전 예약 인스턴스는 20개의 실행 중인 온디맨드 인스턴스에 할인을 적용합니다. 리전 예약 인스턴스를 추가로 구매한 경우 온디맨드 인스턴스 제한에 도달했기 때문에 더 많은 인스턴스를 시작할 수는 없습니다.

영역별 예약 인스턴스를 구입하기 전에 온디맨드 인스턴스 제한이 소유하려는 리전별 예약 인스턴스의 수와 일치하는지 아니면 초과하는지 확인합니다. 필요한 경우 추가 리전별 예약 인스턴스를 구입하기 전에 온디맨드 인스턴스 제한 증가를 요청해야 합니다.

예약 인스턴스 특정 가용 영역에 대해 구매한 영역 예약 인스턴스는 할인은 물론 용량 예약을 제공합니다. 영역 예약 인스턴스 구매를 통해 실행 중인 온디맨드 인스턴스 제한을 초과할 수 있습니다. 예를 들어 이미 20개의 온디맨드 인스턴스를 실행 중이고 20개의 영역 예약 인스턴스를 구매한 경우 영역 예약 인스턴스의 사용과 일치하는 20개의 온디맨드 인스턴스를 추가로 시작할 수 있어, 총 40개의 인스턴스를 실행할 수 있습니다.

Amazon EC2 콘솔은 한도 정보를 제공합니다. 자세한 내용은 [현재 제한 조회 \(p. 1154\)](#) 단원을 참조하십시오.

리전 및 영역 예약 인스턴스(범위)

예약 인스턴스를 구입할 때 예약 인스턴스의 범위를 결정합니다. 범위는 리전 또는 영역입니다.

- 리전: 리전에 대해 예약 인스턴스를 구입하는 경우 이를 리전 예약 인스턴스라고 합니다.
- 영역: 특정 가용 영역에 대해 예약 인스턴스를 구입하는 경우 이를 영역 예약 인스턴스라고 합니다.

리전 및 영역 예약 인스턴스의 차이점

다음 표에서는 리전 예약 인스턴스와 영역 예약 인스턴스의 주요 차이점 중 일부를 요약하여 설명합니다.

	리전 예약 인스턴스	영역 예약 인스턴스
가용 영역 유연성	지정된 리전에 있는 모든 가용 영역의 인스턴스 사용량에 예약 인스턴스 할인이 적용됩니다.	가용 영역 유연성 없음 — 지정된 가용 영역의 인스턴스 사용량에만 예약 인스턴스 할인이 적용됩니다.
용량 예약	용량 예약 없음 — 리전 예약 인스턴스에서 용량 예약을 제공하지 않습니다.	영역 예약 인스턴스에서는 지정된 가용 영역의 용량 예약을 제공합니다.
인스턴스 크기 유연성	크기에 상관없이 인스턴스 패밀리 내 인스턴스 사용량에 예약 인스턴스 할인이 적용됩니다. 기본 테넌시가 있는 Amazon Linux/Unix 예약 인스턴스에 대해서만 지원됩니다. 자세한 내용은 정규화 인자에 의해 결정되는 인스턴스 크기 유연성 (p. 210) 를 참조하십시오.	인스턴스 크기 유연성 없음 — 지정된 인스턴스 유형 및 크기의 인스턴스 사용량에만 예약 인스턴스 할인이 적용됩니다.

자세한 정보와 지침은 [예약 인스턴스 적용 방식 \(p. 209\)](#) 단원을 참조하십시오.

예약 인스턴스 유형(제공 클래스)

예약 인스턴스를 구입할 경우 표준 또는 전환형 제공 클래스 중에서 선택할 수 있습니다. 예약 인스턴스는 해당 기간 동안 단일 인스턴스 유형, 플랫폼, 범위 및 테넌시에 적용됩니다. 컴퓨팅 요건이 변경되면 제공 클래스에 따라 예약 인스턴스를 수정 또는 교환할 수 있습니다. 제공 클래스에도 추가 제한이 있을 수 있습니다.

표준 및 전환형 제공 클래스 간의 차이점은 다음과 같습니다.

표준 예약 인스턴스	전환형 예약 인스턴스
해당 기간 동안 인스턴스 크기와 같은 일부 속성을 수정할 수 있지만 인스턴스 패밀리는 수정할 수 없습니다. 표준 예약 인스턴스는 교환할 수 없으며 수정만 가능합니다. 자세한 내용은 예약 인스턴스 수정 (p. 229) 단원을 참조하십시오.	기간 동안 인스턴스 패밀리, 인스턴스 유형, 플랫폼, 범위 또는 테넌시를 비롯한 새 속성이 있는 다른 전환형 예약 인스턴스와 교환할 수 있습니다. 자세한 내용은 전환형 예약 인스턴스 교환 (p. 232) 단원을 참조하십시오. 전환형 예약 인스턴스의 일부 속성을 수정할 수도 있습니다. 자세한 내용은 예약 인스턴스 수정 (p. 229) 단원을 참조하십시오.
예약 인스턴스 마켓플레이스에서 판매할 수 있습니다.	예약 인스턴스 마켓플레이스에서 판매할 수 없습니다.

표준 및 전환형 예약 인스턴스는 특정 가용 영역의 인스턴스(영역 예약 인스턴스) 또는 리전의 인스턴스(리전 예약 인스턴스)에 적용하기 위해 구입할 수 있습니다. 자세한 정보와 지침은 [예약 인스턴스 적용 방식 \(p. 209\)](#) 단원을 참조하십시오.

지정된 시작 시간과 기간에 따라 매일, 매주 또는 매월 반복적으로 용량 예약을 구매하고 싶다면 정기 예약 인스턴스가 필요 사항을 충족할 수 있습니다. 자세한 내용은 [정기 예약 인스턴스 \(p. 236\)](#) 단원을 참조하십시오.

예약 인스턴스 적용 방식

예약 인스턴스를 구입했으며 예약 인스턴스의 사양과 일치하는 인스턴스가 이미 실행 중인 경우 결제 혜택이 즉시 적용됩니다. 인스턴스를 따로 재시작할 필요가 없습니다. 실행 중인 적용 대상 인스턴스가 없는 경우, 인스턴스를 시작하고 예약 인스턴스에 대해 지정한 동일한 조건과 일치하는지 확인합니다;. 자세한 내용은 [예약 인스턴스 사용 \(p. 223\)](#) 단원을 참조하십시오.

예약 인스턴스는 제공 유형(표준 또는 전환형)과 상관없이 동일한 방식으로 사용량에 적용되므로 속성이 일치하는 실행 중인 온디맨드 인스턴스에 자동으로 적용됩니다.

영역 예약 인스턴스 적용 방식

예약 인스턴스를 특정 가용 영역에 할당하면 해당 가용 영역에서 일치하는 인스턴스 사용량에 대해서는 예약 인스턴스 할인 혜택이 제공됩니다. 예를 들어 가용 영역 us-east-1a에서 c4.xlarge 기본 테넌시 Linux/Unix 표준 예약 인스턴스 2개를 구매하면 가용 영역 us-east-1a에서 실행하는 c4.xlarge 기본 테넌시 Linux/Unix 인스턴스에 최대 2개까지 예약 인스턴스의 할인 혜택을 적용할 수 있습니다. 단, 실행할 인스턴스의 속성(테넌시, 플랫폼, 가용 영역, 인스턴스 유형 및 인스턴스 크기)이 예약 인스턴스의 속성과 일치해야 합니다.

리전 예약 인스턴스 적용 방식

리전 예약 인스턴스는 리전용으로 구입되며 가용 영역 유연성을 제공합니다. 해당 리전에 있는 모든 가용 영역의 인스턴스 사용량에 예약 인스턴스 할인이 적용됩니다.

리전 예약 인스턴스도 크기에 상관없이 인스턴스 패밀리 내 인스턴스 사용량에 예약 인스턴스 할인이 적용되는 인스턴스 크기 유연성을 제공합니다.

인스턴스 크기 유연성에 대한 제한

다음 예약 인스턴스에는 인스턴스 크기 유연성이 적용되지 않습니다.

- 특정 가용 영역(영역 예약 인스턴스)용으로 구입한 예약 인스턴스
- 전용 테넌시를 포함하는 예약 인스턴스
- 예약 인스턴스 for Windows Server, Windows Server with SQL Standard, Windows Server with SQL Server Enterprise, Windows Server with SQL Server Web, RHEL, SUSE Linux Enterprise Server
- G4 인스턴스는 예약 인스턴스 단원을 참조하십시오.

정규화 인자에 의해 결정되는 인스턴스 크기 유연성

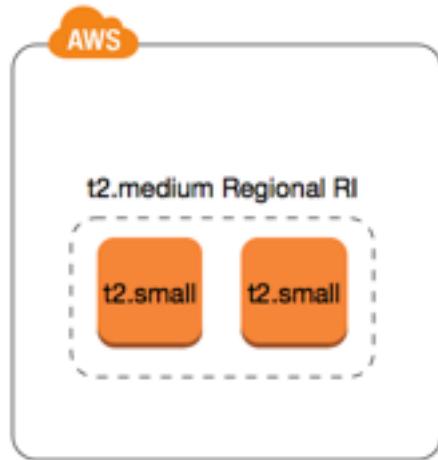
인스턴스 크기 유연성은 인스턴스 크기의 정규화 인자에 의해 결정됩니다. 리전의 모든 가용 영역에서 예약의 인스턴스 크기에 따라 모든 할인 또는 일부 할인이 동일한 인스턴스 패밀리의 실행 중인 인스턴스에 적용됩니다. 속성 중 인스턴스 패밀리, 테넌시 및 플랫폼만 일치하면 됩니다.

인스턴스 크기 유연성은 정규화 인자에 따라 인스턴스 패밀리 내 가장 작은 인스턴스 크기에서 가장 큰 인스턴스 크기에 이르기까지 두루 적용됩니다.

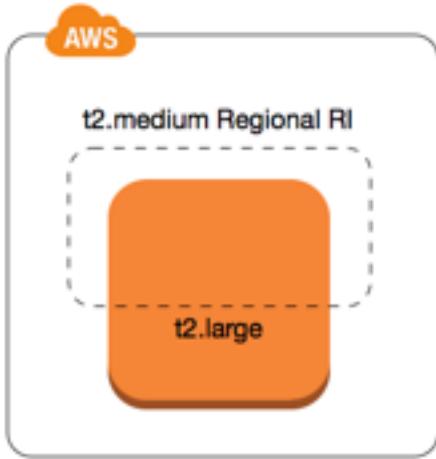
다음 표는 인스턴스 패밀리 내 서로 다른 크기 및 그에 따른 시간당 정규화 인자를 나열한 것입니다. 이 배율은 예약 인스턴스의 할인 요금을 정규화된 인스턴스 패밀리 사용량에 적용하는 데 사용됩니다.

인스턴스 크기	정규화 인자
nano	0.25
micro	0.5
small	1
medium	2
large	4
xlarge	8
2xlarge	16
3xlarge	24
4xlarge	32
6xlarge	48
8xlarge	64
9xlarge	72
10xlarge	80
12xlarge	96
16xlarge	128
18xlarge	144
24xlarge	192
32xlarge	256

예를 들어, t2.medium 인스턴스의 정규화 인자는 2입니다. 미국 동부(버지니아 북부)에서 t2.medium 기본 테넌시 Amazon Linux/Unix 예약 인스턴스를 구입하고, 해당 리전의 계정에 t2.small 인스턴스 2개가 실행 중인 경우 결제 혜택이 두 인스턴스에 전체적으로 적용됩니다.



또는 미국 동부(버지니아 북부) 리전의 계정에 실행 중인 t2.large 인스턴스 1개가 있는 경우 결제 혜택은 인스턴스 사용량의 50%에 적용됩니다.



예약 인스턴스를 수정하면 정규화 인자 역시 적용됩니다. 자세한 정보는 [예약 인스턴스 수정 \(p. 229\)](#) 단원을 참조하십시오.

베어 메탈 인스턴스에 대한 정규화 인자

인스턴스 크기 유연성은 인스턴스 패밀리 내 베어 메탈 인스턴스에도 적용됩니다. 베어 메탈 인스턴스에서 공유 테넌시를 포함하는 리전 Amazon Linux/Unix 예약 인스턴스가 있는 경우 동일한 인스턴스 패밀리 내에서 예약 인스턴스 절감 혜택을 얻을 수 있습니다. 반대의 경우도 마찬가지입니다. 베어 메탈 인스턴스와 동일한 패밀리의 인스턴스에서 공유 테넌시를 포함하는 리전 Amazon Linux/Unix 예약 인스턴스가 있는 경우 베어 메탈 인스턴스에서 예약 인스턴스 절감 혜택을 얻을 수 있습니다.

베어 메탈 인스턴스의 크기는 동일한 인스턴스 패밀리 내의 가장 큰 인스턴스의 크기와 동일합니다. 예를 들어 i3.metal 및 i3.16xlarge의 크기가 서로 동일하므로 해당 정규화 인자도 동일합니다.

Note

.metal 인스턴스 크기에는 단일 정규화 인자가 없습니다. 이 크기는 인스턴스 패밀리에 따라 달라집니다.

베어 메탈 인스턴스 크기	정규화 인자
c5.metal	192
c5d.metal	192
c5n.metal	144
g4dn.metal	128
i3.metal	128
i3en.metal	192
m5.metal	192
m5d.metal	192
r5.metal	192
r5d.metal	192
z1d.metal	96

예를 들어 i3.metal 인스턴스의 정규화 인자는 128입니다. 미국 동부(버지니아 북부)에서 i3.metal 기본 테넌시 Amazon Linux/Unix 예약 인스턴스를 구입하면 다음과 같은 요금 혜택이 적용될 수 있습니다.

- 리전에서 사용자의 계정에 실행 중인 i3.16xlarge가 하나인 경우 i3.16xlarge 인스턴스 (i3.16xlarge 정규화 인자 = 128)에 모든 요금 혜택이 적용됩니다.
- 또는 리전에서 사용자의 계정에 실행 중인 i3.8xlarge 인스턴스가 두 개인 경우 두 i3.8xlarge 인스턴스(i3.8xlarge 정규화 인자 = 64)에 모든 요금 혜택이 적용됩니다.
- 또는 리전에서 사용자의 계정에 실행 중인 i3.4xlarge 인스턴스가 4개인 경우 모든 4개의 i3.4xlarge 인스턴스(i3.4xlarge 정규화 인자 = 32)에 모든 요금 혜택이 적용됩니다.

반대의 경우도 마찬가지입니다. 예를 들어 미국 동부(버지니아 북부)에서 두 개의 i3.8xlarge 기본 테넌시 Amazon Linux/Unix 예약 인스턴스를 구입하고 리전에서 한 개의 i3.metal 인스턴스를 실행 중인 경우 i3.metal 인스턴스에 모든 요금 혜택이 적용됩니다.

예약 인스턴스 적용의 예

다음 시나리오에서 예약 인스턴스가 적용되는 방식을 알 수 있습니다.

Example 시나리오 1: 단일 계정의 예약 인스턴스

계정 A에서 다음 온디맨드 인스턴스를 실행 중입니다.

- 4 x m3.1large Linux, 가용 영역 us-east-1a의 기본 테넌시 인스턴스
- 2 x m4.xlarge Amazon Linux, 가용 영역 us-east-1b의 기본 테넌시 인스턴스
- 1 x c4.xlarge Amazon Linux, 가용 영역 us-east-1c의 기본 테넌시 인스턴스

계정 A에서 다음 예약 인스턴스를 구입합니다.

- 4 x m3.1large Linux, 가용 영역 us-east-1a의 기본 테넌시 예약 인스턴스(용량 예약됨)
- 4 x m4.1large Amazon Linux, us-east-1 리전의 기본 테넌시 예약 인스턴스
- 1 x c4.1large Amazon Linux, us-east-1 리전의 기본 테넌시 예약 인스턴스

예약 인스턴스의 혜택은 다음과 같이 적용됩니다.

- 인스턴스 간에 속성(인스턴스 크기, 리전, 플랫폼, 테넌시)이 서로 일치하므로 m3.large 영역 예약 인스턴스 4개의 할인 및 용량 예약이 m3.large 인스턴스 4개에 적용됩니다.
- m4.large 리전 예약 인스턴스에서는 기본 테넌시가 포함된 리전 단위의 Amazon Linux 예약 인스턴스이므로 가용 영역 및 인스턴스 크기 유연성을 제공합니다.

m4.large는 시간당 정규화 유닛 4개와 같습니다.

m4.large 리전 예약 인스턴스를 4개 구입하였으며, 이에 따라 시간당 정규화 유닛은 총 16개(4x4)입니다. 현재 계정 A에는 실행 중인 m4.xlarge 인스턴스가 2개이며, 이에 따라 시간당 정규화 유닛은 16개(2x8)와 같습니다. 이 경우 m4.large 리전 예약 인스턴스 4개에서는 m4.xlarge 인스턴스 2개의 전체 시간 사용량에 결제 혜택을 제공합니다.

- us-east-1의 c4.large 리전 예약 인스턴스는 기존 테넌시가 포함된 지역 Amazon Linux 예약 인스턴스이므로 가용 영역 및 인스턴스 크기 유연성을 c4.xlarge 인스턴스에 적용합니다. c4.large 인스턴스는 시간당 정규화 유닛 4개와 같고, c4.xlarge는 시간당 정규화 유닛 8개와 같습니다.

이 경우에는 c4.large 리전 예약 인스턴스가 c4.xlarge 사용량에 부분적 혜택을 제공합니다. 이는 c4.large 예약 인스턴스가 사용량의 시간당 정규화 유닛이 4개와 같지만 c4.xlarge 인스턴스는 시간당 정규화 유닛이 8개가 필요하기 때문입니다. 따라서 c4.large 예약 인스턴스의 결제 할인이 c4.xlarge 사용량의 50%에 적용됩니다. 나머지 c4.xlarge 사용량은 온디맨드 요금이 부과됩니다.

Example 시나리오 2: 연결된 계정의 리전 예약 인스턴스

예약 인스턴스가 구입 계정 내 사용량에 먼저 적용된 후 조직의 다른 계정에서 해당하는 사용량에 적용됩니다. 자세한 내용은 [예약 인스턴스 및 통합 결제 \(p. 215\)](#) 단원을 참조하십시오. 인스턴스 크기 유연성을 제공하는 리전 단위 예약 인스턴스의 경우 혜택은 인스턴스 패밀리 내 가장 작은 인스턴스 크기에서 가장 큰 인스턴스 크기에 이르기까지 두루 적용됩니다.

계정 A(구입 계정)에서 다음과 같은 온디맨드 인스턴스를 실행 중입니다.

- 2 x m4.xlarge Linux, 가용 영역 us-east-1a의 기본 테넌시 인스턴스
- 1 x m4.2xlarge Linux, 가용 영역 us-east-1b의 기본 테넌시 인스턴스
- 2 x c4.xlarge Linux, 가용 영역 us-east-1a의 기본 테넌시 인스턴스
- 1 x c4.2xlarge Linux, 가용 영역 us-east-1b의 기본 테넌시 인스턴스

다른 고객이 연결 계정인 계정 B에서 다음과 같은 온디맨드 인스턴스를 실행 중입니다.

- 2 x m4.xlarge Linux, 가용 영역 us-east-1a의 기본 테넌시 인스턴스

계정 A에서 다음과 같은 리전 단위 예약 인스턴스를 구입합니다.

- 4 x m4.xlarge Linux, us-east-1 리전의 기본 테넌시 예약 인스턴스
- 2 x c4.xlarge Linux, us-east-1 리전의 기본 테넌시 예약 인스턴스

지역 예약 인스턴스의 혜택은 다음과 같이 적용됩니다.

- 4개의 m4.xlarge 예약 인스턴스 할인은 계정 A(구입 계정)의 두 개의 m4.xlarge 인스턴스 및 하나의 m4.2xlarge 인스턴스에서 사용됩니다. 세 개의 인스턴스 모두 속성(인스턴스 패밀리, 리전, 패밀리, 테넌시)과 일치합니다. 계정 B(연결된 계정)에 예약 인스턴스와도 일치하는 두 개의 m4.xlarge가 있어도 먼저 구입 계정(계정 A)의 인스턴스에 할인이 적용됩니다. 예약 인스턴스가 리전 예약 인스턴스이므로 용량 예약이 없습니다.

- 2개의 c4.xlarge 예약 인스턴스 할인이 2개의 c4.xlarge 인스턴스에 적용되는데, 이는 c4.2xlarge 인스턴스보다 인스턴스 크기가 작기 때문입니다. 예약 인스턴스가 리전 예약 인스턴스이므로 용량 예약이 없습니다.

Example 시나리오 3: 연결된 계정의 영역 예약 인스턴스

일반적으로 계정에 속한 예약 인스턴스가 해당 계정의 사용량에 먼저 적용됩니다. 하지만 조직 내 다른 계정에 특정 가용 영역에 대해 자격을 갖추었지만 사용하지 않은 예약 인스턴스(영역 예약 인스턴스)가 있다면 계정에 속한 리전 예약 인스턴스에 앞서 이 인스턴스가 계정에 적용됩니다. 이는 예약 인스턴스의 활용도를 극대화하면서 결제 비용을 낮추기 위한 것입니다. 결제의 편의를 위해 조직 내 모든 계정은 하나의 계정으로 취급됩니다. 다음 예제를 참조하십시오.

계정 A(구입 계정)에서 다음과 같은 온디맨드 인스턴스를 실행하고 있습니다.

- 1 x m4.xlarge Linux, 가용 영역 us-east-1a의 기본 테넌시 인스턴스

고객이 연결된 계정 B에서 다음과 같은 온디맨드 인스턴스를 실행하고 있습니다.

- 1 x m4.xlarge Linux, 가용 영역 us-east-1b의 기본 테넌시 인스턴스

계정 A에서 다음과 같은 리전 단위 예약 인스턴스를 구입합니다.

- 1 x m4.xlarge Linux, us-east-1 리전의 기본 테넌시 예약 인스턴스

고객은 또한 연결된 계정 C에서 다음과 같은 영역 예약 인스턴스를 구입합니다.

- 1 x m4.xlarge Linux, 가용 영역 us-east-1a의 기본 테넌시 예약 인스턴스

예약 인스턴스의 혜택은 다음과 같이 적용됩니다.

- 계정 C에 속한 m4.xlarge 영역 예약 인스턴스의 할인은 계정 A의 m4.xlarge 사용량에 적용됩니다.
- 계정 A에 속한 m4.xlarge 리전 예약 인스턴스의 할인은 계정 B의 m4.xlarge 사용량에 적용됩니다.
- 계정 A에 속한 리전 예약 인스턴스가 계정 A의 사용량에 먼저 적용된 경우에는 계정 C에 속한 리전 예약 인스턴스가 미사용 상태로 남게 되고 계정 B의 사용량은 온디맨드 요금으로 부과됩니다.

자세한 내용은 [Billing and Cost Management 보고서의 예약 인스턴스](#) 단원을 참조하십시오.

요금 부과 방법

모든 예약 인스턴스에서는 온디맨드 요금에 비해 할인된 요금을 제공합니다. 예약 인스턴스를 사용하면 실제 사용에 상관없이 전체 기간에 대해 요금을 지불합니다. 예약 인스턴스에 대해 지정한 결제 옵션 ([p. 207](#))에 따라 예약 인스턴스에 대해 선결제, 부분 선결제 또는 월별 결제를 선택할 수 있습니다.

예약 인스턴스가 만료되면 EC2 인스턴스 사용량에 대해 온디맨드 요금이 청구됩니다. 최대 3년 전에 예약 인스턴스 구매를 대기할 수 있습니다. 이를 통해 중단 없는 보장을 받을 수 있습니다. 자세한 내용은 [구매 대기](#) ([p. 218](#)) 단원을 참조하십시오.

AWS 프리 티어는 신규 AWS 계정에 제공됩니다. AWS 프리 티어를 사용하여 Amazon EC2 인스턴스를 실행하며 예약 인스턴스를 구입하는 경우 표준 요금 정책에 따라 요금이 부과됩니다. 자세한 내용은 [AWS 프리 티어](#)를 참조하십시오.

목차

- [사용량 결제](#) ([p. 215](#))
- [청구서 보기](#) ([p. 215](#))

- [예약 인스턴스 및 통합 결제 \(p. 215\)](#)
- [예약 인스턴스 할인 요금 티어 \(p. 215\)](#)

사용량 결제

예약 인스턴스는 선택한 기간 동안 인스턴스 실행 여부와 상관없이 매 시간 청구됩니다. 각 시간은 표준 24시간 시계의 정각(0분 0초)에 시작합니다. 예를 들어 1:00:00부터 1:59:59까지가 1시간입니다. 인스턴스 상태에 대한 자세한 내용은 [인스턴스 수명 주기 \(p. 371\)](#) 단원을 참조하십시오.

예약 인스턴스 청구 혜택은 매 시간마다 한 인스턴스 시간에만 적용됩니다. 인스턴스 시간은 인스턴스가 시작될 때 시작되어 60분간 계속되거나 인스턴스 종지 또는 종료— 중 어느 것이든 먼저 발생할 때까지 계속됩니다.

한 인스턴스가 60분간 계속 실행되었거나 인스턴스가 종지된 다음에 시작된 경우 새로운 인스턴스 시간이 시작됩니다. 인스턴스를 재부팅해도 실행 중인 인스턴스 시간은 재설정되지 않습니다.

예를 들어 어떤 시계상의 시간 중에 한 인스턴스가 종지된 후 다시 시작되어 2시간 동안 더 계속 실행 중인 경우, 첫 번째 인스턴스 시간(다시 시작하기 전)에 대해서는 할인된 예약 인스턴스 요금이 청구됩니다. 다음 인스턴스 시간(다시 시작한 후)에 대해서는 온디맨드 요금이 청구되고, 그 다음 2시간의 인스턴스 시간에 대해서는 할인된 예약 인스턴스 요금이 청구됩니다.

[Billing and Cost Management](#) 콘솔의 Cost Explorer를 사용하면 실행 중인 온디맨드 인스턴스에 대해 절약된 금액을 분석할 수 있습니다. [예약 인스턴스 FAQ](#)에는 정가 계산의 예가 나와 있습니다.

AWS 계정을 닫은 경우 리소스에 대한 온디맨드 결제가 종지됩니다. 그러나 계정에 예약 인스턴스가 있는 경우 해당 인스턴스가 만료될 때까지 이에 대한 청구서를 계속 받게 됩니다.

청구서 보기

계정으로 청구되는 요금과 비용은 [AWS Billing and Cost Management](#) 콘솔에서 확인할 수 있습니다.

- 대시보드에는 계정에 대한 소비 요약이 표시됩니다.
- 청구서 페이지의 세부 정보에서 Elastic Compute Cloud 섹션과 리전을 확장하여 예약 인스턴스에 대한 결제 정보를 가져옵니다.

요금을 온라인으로 확인하거나 CSV 파일을 다운로드할 수 있습니다.

AWS 비용 및 사용 보고서를 통해 예약 인스턴스 사용률을 추적할 수도 있습니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 비용 및 사용 보고서에서 [예약 인스턴스](#)를 참조하십시오.

예약 인스턴스 및 통합 결제

구입 계정이 단일 통합 결제 지급인 계정으로 과금되는 일련의 계정 중 하나인 경우, 예약 인스턴스 요금 혜택이 공유됩니다. 모든 멤버 계정에서 발생한 인스턴스 사용량은 매월 지급인 계정으로 합산됩니다. 이 방식은 일반적으로 직무가 서로 다른 팀이나 그룹이 있는 회사에서 유용하며, 정상적인 예약 인스턴스 규칙에 따라 요금이 계산됩니다. 자세한 내용은 AWS Organizations 사용 설명서의 [통합 결제 및 AWS Organizations](#) 단원을 참조하십시오.

예약 인스턴스를 구매한 계정을 닫으면 예약 인스턴스가 만료되거나 닫힌 계정이 영구적으로 삭제될 때까지 지급인 계정에 예약 인스턴스 요금이 계속 청구됩니다. 닫힌 계정은 90일 후에 영구적으로 삭제됩니다. 삭제 후에는, 해당 멤버 계정에 대한 예약 인스턴스 결제 할인의 혜택이 종지됩니다. 계정 닫기에 대한 자세한 내용은 AWS Organizations 사용 설명서의 [AWS 계정 닫기](#)를 참조하십시오.

예약 인스턴스 할인 요금 티어

할인 요금 티어의 사용 자격에 해당되는 계정은 적용 시점부터 구입한 예약 인스턴스 종 해당 티어에 속하는 예약 인스턴스의 선결제 금액과 인스턴스 사용비가 자동으로 할인됩니다. 할인은 해당 리전 내 예약 인스턴스의 정가 총합이 500,000 USD 이상인 경우만 해당됩니다.

다음 규칙이 적용됩니다.

- 요금 티어 및 이와 관련된 할인은 Amazon EC2 스탠다드 예약 인스턴스 구입 시에만 적용됩니다.
- SQL Server Standard, SQL Server Web 및 SQL Server Enterprise 포함 Windows에는 예약 인스턴스 요금 티어가 적용되지 않습니다.
- SQL Server Standard, SQL Server Web 및 SQL Server Enterprise 포함 Linux에는 예약 인스턴스 요금 티어가 적용되지 않습니다.
- 요금 티어의 할인 혜택은 AWS를 통한 구매에만 적용됩니다. 타사 예약 인스턴스를 구입할 때는 이 할인 혜택이 적용되지 않습니다.
- 할인 요금 티어는 현재 전환형 예약 인스턴스 구입에는 적용되지 않습니다.

주제

- [예약 인스턴스 요금 할인 계산 \(p. 216\)](#)
- [구매 시 할인 티어 적용 \(p. 216\)](#)
- [요금 티어 교차 \(p. 217\)](#)
- [요금 티어 통합 결제 \(p. 217\)](#)

예약 인스턴스 요금 할인 계산

리전의 모든 예약 인스턴스에 대한 정가를 계산하여 계정에 대한 요금 티어를 확인할 수 있습니다. 각 예약의 시간당 부과 요금(hourly recurring price)에 약정 기간의 총 시간을 곱한 다음, 구매 시 할인이 적용되지 않은 선결제 금액(fixed price: 고정 가격이라고도 함)을 더합니다. 정가는 할인이 적용되지 않은 요금 또는 (공개) 요금을 기준으로 하기 때문에 볼륨 할인을 적용받는 경우나 예약 인스턴스 구입 후 가격이 내려가는 경우 정 가에는 영향을 주지 않습니다.

```
List value = fixed price + (undiscounted recurring hourly price * hours in term)
```

예를 들어 1년 부분 선결제 t2.small 예약 인스턴스의 경우 선결제 가격이 60.00 USD이고 시간당 요금이 0.007 USD라고 가정해 봅니다. 이렇게 하면 정가는 121.32 USD입니다.

```
121.32 = 60.00 + (0.007 * 8760)
```

Amazon EC2 콘솔을 사용하여 예약 인스턴스의 고정 가격을 확인하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 예약 인스턴스를 선택합니다.
3. 화면 상단 오른쪽의 열 표시/숨기기(기어 모양 아이콘)를 선택하여 선결제 가격 열을 표시합니다.

명령줄을 사용하여 예약 인스턴스의 고정 가격을 확인하려면

- [describe-reserved-instances\(AWS CLI\)](#)
- [Get-EC2ReservedInstance\(Windows PowerShell용 AWS 도구\)](#)
- [DescribeReservedInstances\(Amazon EC2 API\)](#)

구매 시 할인 티어 적용

사용자가 예약 인스턴스를 구입하면 Amazon EC2에서는 할인 요금 티어에 해당되는 구입에 대해 자동으로 그에 맞는 할인을 적용합니다. 추가 작업 없이 어떤 Amazon EC2 도구에서나 예약 인스턴스 구매가 가능합니다. 자세한 내용은 [예약 인스턴스 구입 \(p. 217\)](#) 단원을 참조하십시오.

한 리전에서 사용 중인 예약 인스턴스의 정가 총액이 할인 요금 티어 기준에 도달하면 다음에 같은 리전에서 예약 인스턴스를 구입할 때 할인이 적용됩니다. 어떤 리전에서 예약 인스턴스를 하나 구입했는데 그에 따른 합계가 할인 요금 티어 기준 금액을 초과하는 경우 기준을 초과한 금액에 대해 할인이 적용됩니다. 구매 과정에서 생성되는 임시 예약 인스턴스 ID에 대한 자세한 내용은 [요금 티어 교차 \(p. 217\)](#) 단원을 참조하십시오.

정가 총액이 예약 인스턴스 만료 등의 이유로 이용 종이던 할인 요금 티어 기준 이하로 변경되면 그 다음에 해당 리전에서 예약 인스턴스를 구입할 때는 할인이 적용되지 않습니다. 단, 구입 시 할인 요금 티어 범위에 해당되었던 기존의 예약 인스턴스에 대해서는 계속 할인을 받을 수 있습니다.

예약 인스턴스 구입 상황은 다음 네 가지 중 한 경우입니다.

- 미할인 - 같은 리전에서 구매한 합계가 할인 기준 금액보다 아직 적은 경우입니다.
- 부분 할인 - 같은 리전에서 구매하면서 최하 등급의 할인 티어 기준 금액에 도달한 경우입니다. 미할인이 하나 이상의 예약에 적용되고 할인 요금이 나머지 예약에 적용됩니다.
- 전체 할인 - 한 리전 내의 전체 구매가 동일한 할인 티어에 해당되고 적절히 할인됩니다.
- 이중 할인 - 같은 리전에서 구매하면서 할인 티어 등급이 기준보다 더 높아진 경우입니다. 이 경우 두 가지 요금이 차등 적용됩니다. 합산 가격을 기준으로 하나 또는 그 이상의 예약 인스턴스에는 기준 티어 할인이, 나머지 인스턴스에는 상위 티어 할인이 적용됩니다.

요금 티어 교차

구매 시점에서 합산 금액이 어떤 할인 요금 티어 기준을 도달하게 되면 함께 구매하는 인스턴스 중 일부는 정상적인 예약 인스턴스 가격이 적용되고 티어 기준을 초과하는 인스턴스는 티어에 따른 할인이 적용됩니다.

함께 구매한 인스턴스에 미할인 티어(정상 가격), 하나 이상의 할인 티어가 차등 적용되므로, 예약 인스턴스 서비스에서는 여러 개의 예약 인스턴스 ID를 생성합니다. ID는 같은 티어의 인스턴스를 묶어 티어당 하나씩 부여됩니다. 따라서 CLI 명령이나 API 작업으로 구입했을 때 부여되는 ID는 새로 구입한 예약 인스턴스의 실제 ID와는 다릅니다.

요금 티어 통합 결제

통합 결제 계정은 한 리전 내 회원 계정의 정가를 합산합니다. 통합 결제 계정에 속하는 사용 중인 모든 예약 인스턴스의 정가 총액이 할인 요금 티어의 기준 금액에 도달하면 통합 결제 계정의 모든 구성원 계정에서 구입한 예약 인스턴스에 대해 할인을 받을 수 있습니다(해당 통합 결제 계정의 정가가 할인 요금 티어의 기준 금액 이상으로 유지되는 동안 계속 적용). 자세한 내용은 [예약 인스턴스 및 통합 결제 \(p. 215\)](#) 단원을 참조하십시오.

예약 인스턴스 구입

예약 인스턴스를 구입하려면 AWS 및 타사 판매업자로부터 예약 인스턴스 상품을 검색하고, 찾고 있는 인스턴스와 정확히 일치하는 인스턴스를 찾을 때까지 검색 파라미터를 조정합니다.

구입할 예약 인스턴스를 검색하면 반환된 상품의 비용에 대한 견적을 받게 됩니다. 구입을 진행하면 AWS에서 구입 가격에 제한 가격을 자동으로 설정합니다. 그러면 구입하는 예약 인스턴스의 총 가격이 제시된 견적 가를 초과하지 않게 됩니다.

여하한 이유로 가격이 오르거나 변경되면 구입이 완료되지 않습니다. 구매 당시 선택한 조건과 비슷한데 가격은 더 낮은 상품이 있을 경우 AWS는 더 저렴한 상품을 판매합니다.

구입을 확정하기 전에 구매하기로 결정한 예약 인스턴스의 세부 정보를 검토하고 모든 파라미터가 정확한지 확인하십시오. 예약 인스턴스는 한 번 구매하고 나면 구매를 취소할 수 없습니다(예약 인스턴스 마켓플레이스의 타사에서 구매한 경우와 AWS에서 구매한 경우 모두 동일).

Note

예약 인스턴스를 구매하고 수정하려면 가용 영역을 설명할 수 있는 권한과 같은 적절한 권한이 IAM 사용자 계정에 있는지 확인하십시오. 자세한 내용은 [AWS CLI 또는 AWS SDK 작업을 위한 예제 정책 및 Amazon EC2 콘솔 작업을 위한 예제 정책](#)을 참조하십시오.

작업

- [플랫폼 선택 \(p. 218\)](#)
- [구매 대기 \(p. 218\)](#)
- [표준 예약 인스턴스 구입 \(p. 219\)](#)
- [전환형 예약 인스턴스 구입 \(p. 220\)](#)
- [예약 인스턴스 보기 \(p. 222\)](#)
- [대기 중인 구매 취소 \(p. 222\)](#)
- [예약 인스턴스 갱신 \(p. 222\)](#)
- [예약 인스턴스 사용 \(p. 223\)](#)

플랫폼 선택

Amazon EC2는 예약 인스턴스에 다음과 같은 Windows 플랫폼을 지원합니다.

- Windows
- SQL Server Standard가 설치된 Windows
- SQL Server Web이 설치된 Windows
- SQL Server Enterprise가 설치된 Windows

예약 인스턴스를 구입할 경우 해당 인스턴스의 운영 체제를 나타내는 플랫폼용 서비스를 선택해야 합니다.

- Windows with SQL Standard, Windows with SQL Server Enterprise, Windows with SQL Server Web의 경우, 해당 플랫폼용 특정 서비스를 선택해야 합니다.
- 그 외 모든 Windows 버전에서는 Windows 플랫폼용 서비스를 선택합니다.

Important

- 결제 코드로 AMI에서 시작된 온디맨드 인스턴스에 적용하기 위해 예약 인스턴스를 구매하는 경우 일치하는 청구 제품 코드가 예약 인스턴스에 있는지 확인합니다. 일치하는 청구 제품 코드 없이 예약 인스턴스를 구매하는 경우 예약 인스턴스는 온디맨드 인스턴스에 적용되지 않습니다. AMI 결제 코드를 얻는 방법에 대한 자세한 내용은 [결제 정보 가져오기 \(p. 108\)](#) 단원을 참조하십시오.
- 예약 인스턴스를 구입하여 AWS Marketplace AMI에서 시작된 온디맨드 인스턴스에 적용하려는 경우 먼저 AMI의 PlatformDetails 필드를 확인합니다. PlatformDetails 필드는 구입할 예약 인스턴스 항목을 나타냅니다. AMI의 플랫폼 세부 정보는 예약 인스턴스의 플랫폼과 일치해야 합니다. 그렇지 않으면 예약 인스턴스가 온디맨드 인스턴스에 적용되지 않습니다. AMI의 플랫폼 세부 정보를 보는 방법에 대한 자세한 내용은 [결제 정보 가져오기 \(p. 108\)](#) 단원을 참조하십시오.

Linux에 지원되는 플랫폼에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [플랫폼 선택](#)을 참조하십시오.

구매 대기

기본적으로 예약 인스턴스를 구매하면 즉시 실행됩니다. 또는 향후 날짜와 시간으로 구매를 대기시킬 수 있습니다. 예를 들어, 기존 예약 인스턴스 만료 즈음에 구매를 대기시킬 수 있습니다. 이를 통해 중단 없는 보장을 받을 수 있습니다.

다른 판매자로부터의 지역 예약 인스턴스(영역 예약 인스턴스나 예약 인스턴스 아님) 구매를 대기할 수 있습니다. 최대 3년 전에 구매를 대기할 수 있습니다. 예약된 날짜 및 시간에 기본 결제 방법을 사용하여 구매가 실행됩니다. 결제가 완료되면 결제 혜택이 적용됩니다.

대기 중인 구매 내역을 Amazon EC2 콘솔에서 볼 수 있습니다. 대기 중인 구매의 상태는 대기 중입니다. 예약된 시간 전에 언제든지 대기 중인 구매를 취소할 수 있습니다. 세부 정보는 [대기 중인 구매 취소 \(p. 222\)](#) 단원을 참조하십시오.

표준 예약 인스턴스 구입

특정 가용 영역에서 표준 예약 인스턴스를 구입하고 용량을 예약할 수 있습니다. 또는 용량 예약을 포기하고 리전 단위의 표준 예약 인스턴스를 구입할 수 있습니다;.

콘솔을 사용하여 표준 예약 인스턴스를 구매하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 예약 인스턴스를 선택한 다음 예약 인스턴스 구입을 선택합니다.
3. 제공 클래스에서 표준을 선택하여 표준 예약 인스턴스를 표시합니다.
4. 용량 예약을 구입하려면 구입 화면의 상단 오른쪽 모서리 부분에서 용량이 예약된 제공만 표시를 선택합니다. 리전 단위의 예약 인스턴스를 구입하려면 상자를 선택하지 않은 채로 둡니다.
5. 필요에 따라 다른 구성을 선택하고 검색을 선택합니다.

예약 인스턴스 마켓플레이스에서 표준 예약 인스턴스를 구입하려면 검색 결과의 판매자 열에서 타사를 찾습니다. 기간 열에 비 표준 약정이 표시됩니다.

6. 구입할 예약 인스턴스를 선택하고 수량을 입력한 후 장바구니에 추가를 선택합니다.
 7. 선택한 예약 인스턴스의 요약을 보려면 장바구니 보기를 선택합니다.
 8. Order On(주문 시각)이 Now(지금)이면 즉시 구매가 완료됩니다. 구매를 대기시키려면 지금을 선택하고 날짜를 선택하십시오. 장바구니에서 적합한 각 상품에 대해 다른 날짜를 선택할 수 있습니다. 구매는 브라우저 시간대에서 선택한 날짜에 00:00까지 대기합니다.
 9. 주문을 완료하려면 주문을 선택합니다.
- 주문 당시 선택한 조건과 비슷하지만 가격이 더 낮은 상품이 있는 경우 AWS는 더 저렴한 상품을 판매합니다.
10. 주문 상태가 상태 열에 나열됩니다. 주문이 완료되면 상태 값이 payment-pending에서 active로 바뀝니다. 예약 인스턴스가 active인 경우 사용할 준비가 됩니다.

Note

상태가 `retired`로 바뀌면 AWS에서 결제를 받을 수 없습니다.

AWS CLI를 사용하여 표준 예약 인스턴스를 구매하려면

1. `describe-reserved-instances-offerings` 명령을 사용하여 사용 가능한 예약 인스턴스를 찾습니다. --offering-class 파라미터에 대해 `standard`을 지정하여 표준 예약 인스턴스만 반환합니다. 추가 파라미터를 적용하여 결과를 좁힐 수 있습니다. 예를 들어 1년 동안만 Linux/UNIX에 대해 기본 테넌시가 포함된 리전 단위의 `t2.large` 예약 인스턴스를 구매하려는 경우 다음과 같이 하십시오.

```
aws ec2 describe-reserved-instances-offerings \
--instance-type t2.large \
--offering-class standard \
--product-description "Linux/UNIX" \
--instance-tenancy default \
--filters Name=duration,Values=31536000 Name=scope,Values=Region
```

예약 인스턴스 마켓플레이스에서만 예약 인스턴스를 찾으려면 `marketplace` 필터를 사용하고 기간이 1년 또는 3년 기간보다 짧을 수 있으므로 요청에 기간을 지정하지 않습니다.

```
aws ec2 describe-reserved-instances-offerings \
--instance-type t2.large \
```

```
--offering-class standard \
--product-description "Linux/UNIX" \
--instance-tenancy default \
--filters Name=marketplace,Values=true
```

요구 사항에 맞는 예약 인스턴스를 찾은 경우 상품 ID를 기록합니다. 예:

```
"ReservedInstancesOfferingId": "bec624df-a8cc-4aad-a72f-4f8abc34caf2"
```

- purchase-reserved-instances-offering 명령을 사용하여 예약 인스턴스를 구매합니다. 이전 단계에서 얻은 예약 인스턴스 상품 ID를 지정하고 예약을 위한 인스턴스 수를 지정해야 합니다.

```
aws ec2 purchase-reserved-instances-offering \
--reserved-instances-offering-id bec624df-a8cc-4aad-a72f-4f8abc34caf2 \
--instance-count 1
```

기본적으로 구매는 즉시 완료됩니다. 또는 구매를 대기시키려면 다음 매개 변수를 이전 호출에 추가하십시오.

```
--purchase-time "2020-12-01T00:00:00Z"
```

- describe-reserved-instances 명령을 사용하여 예약 인스턴스의 상태를 가져옵니다.

```
aws ec2 describe-reserved-instances
```

또는 다음 Windows PowerShell용 AWS 도구 명령을 사용합니다.

- [Get-EC2ReservedInstancesOffering](#)
- [New-EC2ReservedInstance](#)
- [Get-EC2ReservedInstance](#)

구매를 완료했으며 예약 인스턴스의 사양과 일치하는 인스턴스가 이미 실행 중인 경우 결제 혜택이 즉시 적용됩니다. 인스턴스를 따로 재시작할 필요가 없습니다. 실행 중인 적합 인스턴스가 없는 경우, 인스턴스를 시작하고 예약 인스턴스에 대해 지정한 동일한 조건과 일치하는지 확인합니다. 자세한 내용은 [예약 인스턴스 사용 \(p. 223\)](#) 단원을 참조하십시오.

예를 들어 실행 중인 인스턴스에 예약 인스턴스를 적용하는 방법은 [예약 인스턴스 적용 방식 \(p. 209\)](#) 단원을 참조하십시오.

전환형 예약 인스턴스 구입

특정 가용 영역에서 전환형 예약 인스턴스를 구입하고 용량을 예약할 수 있습니다. 또는 용량 예약을 포기하고 리전 단위의 전환형 예약 인스턴스를 구입할 수 있습니다.

콘솔을 사용하여 전환형 예약 인스턴스를 구입하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 예약 인스턴스를 선택한 다음 예약 인스턴스 구입을 선택합니다.
- 제공 클래스에 대해 컨버터블을 선택하여 전환형 예약 인스턴스를 표시합니다.
- 용량 예약을 구입하려면 구입 화면의 상단 오른쪽 모서리 부분에서 용량이 예약된 제공만 표시를 선택합니다. 리전 단위의 예약 인스턴스를 구입하려면 상자를 선택하지 않은 채로 둡니다.
- 필요에 따라 다른 구성을 선택하고 검색을 선택합니다.
- 구입할 전환형 예약 인스턴스를 선택하고 수량을 입력한 후 장바구니에 추가를 선택합니다.

7. 선택한 내역을 보려면 장바구니 보기를 선택합니다.
 8. Order On(주문 시각)이 Now(지금)이면 즉시 구매가 완료됩니다. 구매를 대기시키려면 지금을 선택하고 날짜를 선택하십시오. 장바구니에서 적합한 각 상품에 대해 다른 날짜를 선택할 수 있습니다. 구매는 브라우저 시간대에서 선택한 날짜에 00:00까지 대기합니다.
 9. 주문을 완료하려면 주문을 선택합니다.
- 주문 당시 선택한 조건과 비슷하지만 가격이 더 낮은 상품이 있는 경우 AWS는 더 저렴한 상품을 판매합니다.
10. 주문 상태가 상태 열에 나열됩니다. 주문이 완료되면 상태 값이 payment-pending에서 active로 바뀝니다. 예약 인스턴스가 active인 경우 사용할 준비가 됩니다.

Note

상태가 `retired`로 바뀌면 AWS에서 결제를 받을 수 없습니다.

AWS CLI를 사용하여 전환형 예약 인스턴스를 구입하려면

1. `describe-reserved-instances-offerings` 명령을 사용하여 사용 가능한 예약 인스턴스를 찾습니다. `--offering-class` 파라미터에 대해 `convertible`를 지정하여 전환형 예약 인스턴스만 반환합니다. 예를 들어 Linux/UNIX에 대해 기본 테넌시가 포함된 리전 단위의 `t2.large` 예약 인스턴스를 구입하려 할 경우와 같이 결과를 좁히기 위해 추가 파라미터를 적용할 수 있습니다.

```
aws ec2 describe-reserved-instances-offerings \
--instance-type t2.large \
--offering-class convertible \
--product-description "Linux/UNIX" \
--instance-tenancy default \
--filters Name=scope,Values=Region
```

요구 사항에 맞는 예약 인스턴스를 찾은 경우 상품 ID를 기록합니다. 예:

```
"ReservedInstancesOfferingId": "bec624df-a8cc-4aad-a72f-4f8abc34caf2"
```

2. `purchase-reserved-instances-offering` 명령을 사용하여 예약 인스턴스를 구매합니다. 이전 단계에서 얻은 예약 인스턴스 상품 ID를 지정하고 예약을 위한 인스턴스 수를 지정해야 합니다.

```
aws ec2 purchase-reserved-instances-offering \
--reserved-instances-offering-id bec624df-a8cc-4aad-a72f-4f8abc34caf2 \
--instance-count 1
```

기본적으로 구매는 즉시 완료됩니다. 또는 구매를 대기시키려면 다음 매개 변수를 이전 호출에 추가하십시오.

```
--purchase-time "2020-12-01T00:00:00Z"
```

3. `describe-reserved-instances` 명령을 사용하여 예약 인스턴스의 상태를 가져옵니다.

```
aws ec2 describe-reserved-instances
```

또는 다음 Windows PowerShell용 AWS 도구 명령을 사용합니다.

- [Get-EC2ReservedInstancesOffering](#)
- [New-EC2ReservedInstance](#)
- [Get-EC2ReservedInstance](#)

예약 인스턴스의 사양과 일치하는 인스턴스가 이미 실행 중인 경우 결제 혜택이 즉시 적용됩니다. 인스턴스를 따로 재시작할 필요가 없습니다. 실행 중인 적합 인스턴스가 없는 경우, 인스턴스를 시작하고 예약 인스턴스에 대해 지정한 동일한 조건과 일치하는지 확인합니다. 자세한 내용은 [예약 인스턴스 사용 \(p. 223\)](#) 단원을 참조하십시오.

예를 들어 실행 중인 인스턴스에 예약 인스턴스를 적용하는 방법은 [예약 인스턴스 적용 방식 \(p. 209\)](#) 단원을 참조하십시오.

예약 인스턴스 보기

Amazon EC2 콘솔 또는 명령줄 도구를 사용하여 구입한 예약 인스턴스를 볼 수 있습니다.

콘솔에서 예약 인스턴스를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 예약 인스턴스를 선택합니다.
3. 활성 및 수명종료 예약 인스턴스가 나열됩니다. 상태 열에 상태가 표시됩니다.
4. 예약 인스턴스 마켓플레이스의 판매자인 경우 [예약 인스턴스 마켓플레이스 \(p. 223\)](#)에 나열된 예약 상태가 내 항목 탭에 표시됩니다. 자세한 내용은 [예약 인스턴스 항목 상태 \(p. 227\)](#) 단원을 참조하십시오.

명령줄을 사용하여 예약 인스턴스를 보려면

- [describe-reserved-instances\(AWS CLI\)](#)
- [Get-EC2ReservedInstance\(Windows PowerShell용 도구\)](#)

대기 중인 구매 취소

최대 3년 전에 구매를 대기할 수 있습니다. 예약된 시간 전에 언제든지 대기 중인 구매를 취소할 수 있습니다.

대기 중인 구매 취소

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 예약 인스턴스를 선택합니다.
3. 하나 이상의 예약 인스턴스를 선택합니다.
4. 조치, 대기 중인 예약 인스턴스 삭제를 선택하십시오.
5. 확인 메시지가 나타나면 예, 삭제합니다를 선택합니다.

명령줄을 사용하여 대기 중인 구매를 취소하려면

- [delete-queued-reserved-instances\(AWS CLI\)](#)
- [Remove-EC2QueuedReservedInstance\(Windows PowerShell용 도구\)](#)

예약 인스턴스 갱신

만료되기 전에 예약 인스턴스를 갱신할 수 있습니다. 예약 인스턴스 대기열을 갱신하면 현재 예약 인스턴스가 만료될 때까지 동일한 구성으로 예약 인스턴스를 구매할 수 있습니다.

대기 중인 구매를 사용하여 예약 인스턴스를 갱신하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 예약 인스턴스를 선택합니다.
3. 하나 이상의 예약 인스턴스를 선택합니다.
4. 작업, Renew Reserved Instances(예약 인스턴스 갱신)를 선택합니다.

5. 주문을 완료하려면 주문을 선택합니다.

예약 인스턴스 사용

사양이 일치할 경우 실행 중인 온디맨드 인스턴스에 예약 인스턴스가 자동으로 적용됩니다. 예약 인스턴스의 사양과 일치하는 온디맨드 인스턴스가 실행되고 있지 않은 경우 필수 사양이 포함된 인스턴스를 시작할 때까지 예약 인스턴스가 사용되지 않습니다.

예약 인스턴스의 결제 혜택을 활용하기 위해 인스턴스를 시작할 경우 시작 시 다음 정보를 지정해야 합니다.

- **플랫폼:** 예약 인스턴스의 플랫폼(제품 설명)과 일치하는 Amazon 머신 이미지(AMI)를 선택해야 합니다.; 예를 들어, Linux/UNIX를 지정한 경우 Amazon Linux AMI 또는 Ubuntu AMI에서 인스턴스를 시작할 수 있습니다.
- **인스턴스 유형:** t2.large와 같이 예약 인스턴스와 동일한 인스턴스 유형을 지정합니다.
- **가용 영역:** 특정 가용 영역에 대해 예약 인스턴스를 구입한 경우 동일한 가용 영역으로 해당 인스턴스를 시작해야 합니다. 리전 단위의 예약 인스턴스를 구입한 경우 모든 가용 영역으로 인스턴스를 시작할 수 있습니다.
- **테넌시:** 인스턴스의 테넌시는 예약 인스턴스의 테넌시와 일치해야 합니다(예: dedicated 또는 shared). 자세한 내용은 [전용 인스턴스 \(p. 349\)](#) 단원을 참조하십시오.

자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오. 예를 들어 실행 중인 인스턴스에 예약 인스턴스를 적용하는 방법은 [예약 인스턴스 적용 방식 \(p. 209\)](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling 또는 다른 AWS 서비스를 사용하여 예약 인스턴스의 혜택이 적용되는 온디맨드 인스턴스를 시작할 수 있습니다. 자세한 내용은 [Amazon EC2 Auto Scaling 사용 설명서](#)를 참조하십시오.

예약 인스턴스 마켓플레이스

예약 인스턴스 마켓플레이스은 타사 및 AWS 고객의 미사용 표준 예약 인스턴스 판매를 지원하는 플랫폼으로, 사용 기간 및 요금 옵션이 다양합니다. 예를 들어 인스턴스를 새로운 AWS 리전으로 이동할 경우, 새 인스턴스 유형으로 변경한 후, 약정이 만료되기 전에 프로젝트가 종료될 경우, 비즈니스에서 변경이 필요한 경우 또는 불필요한 용량이 있는 경우 예약 인스턴스를 판매할 수 있습니다.

예약 인스턴스 마켓플레이스에서 미사용 예약 인스턴스를 판매하려 할 경우 특정 자격 기준을 충족해야 합니다.

내용

- [예약 인스턴스 마켓플레이스에서 판매 \(p. 223\)](#)
- [예약 인스턴스 Marketplace에서 구입 \(p. 229\)](#)

예약 인스턴스 마켓플레이스에서 판매

예약 인스턴스 마켓플레이스에 예약 인스턴스를 등록하자마자 잠재적 구매자들에게 노출되어 판매가 가능합니다. 모든 예약 인스턴스는 남은 약정 기간 및 시간당 요금에 따라 분류됩니다.

구매자의 요청을 처리할 때 AWS는 특정 그룹에서 선결제 금액이 가장 낮은 예약 인스턴스부터 판매합니다. 그런 다음 구매자의 주문이 모두 처리될 때까지 낮은 가격부터 순차적으로 예약 인스턴스를 판매합니다. 그런 다음 AWS는 이 거래를 처리하고 해당 예약 인스턴스의 소유권을 구매자에게 이전합니다.

예약 인스턴스가 판매되기 전까지는 판매자에게 소유권이 있습니다. 판매 후에는 용량 예약과 할인 기본 요금이 구매자에게 양도됩니다. 인스턴스를 계속 사용할 경우 AWS는 해당 예약 인스턴스가 판매된 시점부터 온디맨드 요금을 적용합니다.

목차

- 규제 및 제한 (p. 224)
- 판매자 등록 (p. 224)
- 지급금 은행 계좌 (p. 225)
- 세금 정보 (p. 225)
- 예약 인스턴스 요금 (p. 226)
- 예약 인스턴스 나열 (p. 226)
- 예약 인스턴스 항목 상태 (p. 227)
- 항목 기간 (p. 227)
- 예약 인스턴스 판매 후 절차 (p. 228)
- 판매 대금 정산 (p. 228)
- 구매자와의 정보 공유 (p. 228)

규제 및 제한

예약 인스턴스 마켓플레이스에서 사용하지 않은 예약을 판매하기 전에 판매자로 등록해야 합니다. 자세한 내용은 [판매자 등록 \(p. 224\)](#) 단원을 참조하십시오.

예약 인스턴스 판매 시 다음과 같은 제한 및 제약이 적용됩니다.

- 예약 인스턴스 마켓플레이스에서 Amazon EC2 표준 예약 인스턴스만 판매할 수 있습니다. Amazon EC2 전환형 예약 인스턴스는 판매할 수 없습니다. Amazon RDS 및 Amazon ElastiCache와 같은 기타 AWS 서비스의 예약 인스턴스는 판매할 수 없습니다.
- 표준 예약 인스턴스의 남은 사용 기간이 한 달 이상이어야 합니다.
- 기본적으로 비활성화된 리전에서는 표준 예약 인스턴스를 판매할 수 없습니다.
- 예약 인스턴스 마켓플레이스에서 허용되는 최소 허용 판매가는 0.00 USD입니다.
- 예약 인스턴스 마켓플레이스에서는 선결제 없음, 부분 선결제, 혹은 전체 선결제 예약 인스턴스를 판매할 수 있습니다. 예약 인스턴스에 선결제가 있으면 AWS에 선결제가 완료된 이후, 그리고 예약 활성 기간(소유 기간)이 30일 이상인 경우에만 판매할 수 있습니다.
- 등록된 항목을 예약 인스턴스 마켓플레이스에서 바로 변경하는 것은 불가능합니다. 하지만 판매 등록을 취소하고 새 파라미터를 지정한 다음 다시 등록하는 방식으로 변경하는 것은 가능합니다. 자세한 내용은 [예약 인스턴스 요금 \(p. 226\)](#) 단원을 참조하십시오. 판매 등록하기 전에 예약 인스턴스를 수정할 수도 있습니다. 자세한 내용은 [예약 인스턴스 수정 \(p. 229\)](#) 단원을 참조하십시오.
- AWS는 예약 인스턴스 마켓플레이스에서 판매하는 각 표준 예약 인스턴스에 대해 총 선결제 금액의 12%를 서비스 수수료로 청구합니다. 선결제 금액은 판매자가 판매 등록한 표준 예약 인스턴스에 책정한 가격입니다;
- 판매자로 등록하는 경우 지정한 은행에 미국 주소가 있어야 합니다. 자세한 내용은 AWS Marketplace 판매자 설명서의 [유료 제품에 대한 추가 판매자 요구 사항](#)을 참조하십시오.
- Amazon Internet Services Private Limited(AISPL) 고객은 미국 은행 계좌를 가지고 있더라도 예약 인스턴스 마켓플레이스에서 예약 인스턴스를 판매할 수 없습니다. 자세한 내용은 [AWS 계정과 AISPL 계정의 차이점은 무엇입니까?](#)를 참조하십시오.

판매자 등록

Note

AWS 계정 루트 사용자만 계정을 판매자로 등록할 수 있습니다.

예약 인스턴스 마켓플레이스에서 판매하려면 먼저 판매자로 등록해야 합니다. 등록 과정에서 다음 정보를 제공해야 합니다.

- 은행 정보 - AWS에서 예약 인스턴스를 판매할 경우 판매 대금을 지급하기 위해 사용자의 은행 정보가 필요합니다. 이때 미국 소재지가 있는 은행을 선택해야 합니다. 자세한 내용은 [지급금 은행 계좌 \(p. 225\)](#) 단원을 참조하십시오.

- 세금 정보 - 모든 판매자는 세금 신고 의무를 결정하기 위해서 세금 신고서를 작성해야 합니다. 자세한 내용은 [세금 정보 \(p. 225\)](#) 단원을 참조하십시오.

AWS에서 판매자 등록에 필요한 과정을 모두 마치면 등록 확인과 함께 예약 인스턴스 마켓플레이스에서 판매를 시작할 수 있음을 알리는 이메일이 발송됩니다.

지급금 은행 계좌

AWS에서 예약 인스턴스의 판매 대금을 지불하기 위해서는 사용자의 은행 정보가 필요합니다. 이때 미국 소재지가 있는 은행을 선택해야 합니다. 자세한 내용은 AWS Marketplace 판매자 설명서의 [유료 제품에 대한 추가 판매자 요구 사항](#)을 참조하십시오.

지급금을 받을 기본 은행 계좌를 등록하려면

- [예약 인스턴스 마켓플레이스 판매자 등록](#) 페이지를 열고 AWS 자격 증명을 사용하여 로그인합니다.
- 은행 계좌 관리(Manage Bank Account) 페이지에서 판매 대금을 지급 받을 은행의 다음 정보를 입력합니다.
 - 은행 계좌 소유자 이름
 - 송금 번호
 - 계좌 번호
 - 은행 계좌 유형

Note

법인 계좌를 사용할 경우 은행 계좌를 팩스(1-206-765-3424)로 보내라는 메시지가 표시됩니다.

등록되면 이 은행 계좌가 기본 계좌로 설정되고 은행 확인은 보류 상태가 됩니다. 새로운 은행 계좌를 확인하려면 최대 2주 정도 걸리며 이 기간 동안에는 입금을 받을 수 없습니다. 검증된 계좌는 대금 입금이 완료되는 데 보통 2일 정도 걸립니다.

지급금을 받을 기본 은행 계좌를 변경하려면

- [예약 인스턴스 마켓플레이스 판매자 등록](#) 페이지에서 등록 시 사용한 계정으로 로그인합니다.
- 은행 계좌 관리(Manage Bank Account) 페이지에서 필요에 따라 새로운 은행 계좌를 추가하거나 기본 은행 계좌를 수정합니다.

세금 정보

예약 인스턴스를 판매할 때 판매세나 부가가치세 등 거래세가 발생할 수 있습니다. 거래세의 적용 여부는 회사 내부의 세금, 법무, 회계 부서 등 관련 부서에 문의하여 확인하십시오. 거래에 관련된 세금을 정산하고 관련 부처에 납부할 책임은 사용자에게 있습니다.

판매자 등록 과정에서는 [판매자 등록 포털](#)에서 세금 신고서를 작성해야 합니다. 인터뷰어가 세금 정보를 받아서 세금 신고 의무를 결정하기 위한 IRS form W-9, W-8BEN, 혹은 W-8BEN-E를 추가합니다.

세금 신고서 작성 시 입력하는 세금 정보는 개인인지 아니면 기업인지 혹은 미국 법인인지 아니면 미국 외 법인인지에 따라 다릅니다. 세금 신고서를 작성할 때는 다음을 참고하십시오.

- 이 주제를 비롯해 AWS에서 제공하는 정보는 세금과 법률 그 외 분야에 대한 전문 조언이 아닙니다. IRS 세금 신고 규정이 기업에 미칠 수 있는 영향이나 다른 의문점은 세금, 법률, 기타 분야의 전문가에게 상담하십시오.
- IRS 세금 신고 규정을 가장 효율적으로 준수할 수 있는 방법은 인터뷰에 나오는 모든 질문에 답변하고 요청된 모든 정보를 제공하는 것입니다.

- 답변을 확인하십시오. 오타나 사업자 등록 번호가 잘못 기재되지 않도록 유의해야 합니다. 이에 따라 세금 신고서를 다시 작성해야 할 수 있습니다.

세금 신고 및 IRS 신고 기준을 바탕으로 Amazon은 Form 1099-K를 생성합니다. Amazon은 세금 계정이 기준선을 초과한 년도의 다음 해 1월 31일 또는 그 이전에 Form 1099-K 복사본을 우편으로 보냅니다. 예를 들어 세금 계정이 2018년에 한계에 도달하면 Form 1099-K는 2019년 1월 31일 또는 그 이전에 우편으로 보내집니다.

IRS 세금 신고 규정과 Form 1099-K에 대한 자세한 내용은 [IRS 웹 사이트](#) 단원을 참조하십시오.

예약 인스턴스 요금

판매자는 판매할 예약 인스턴스에 대한 선결제 금액만 책정할 수 있습니다. 선결제 금액은 구매자가 예약 인스턴스를 구매할 때 지불하는 일회성 요금입니다.;

다음은 알아 두어야 할 중요한 제한 사항입니다.

- 최대 50,000 USD의 예약 인스턴스를 판매할 수 있습니다. 이 제한을 늘리려면 [EC2 예약 인스턴스 판매 양식](#)을 작성하십시오.
- 최대 5,000개의 예약 인스턴스를 판매할 수 있습니다. 이 제한을 늘리려면 [EC2 예약 인스턴스 판매 양식](#)을 작성하십시오.
- 최소 요금은 0 USD입니다. 예약 인스턴스 마켓플레이스에서 허용되는 최소 허용 판매가는 0.00 USD입니다.

판매 등록을 직접 변경할 수는 없습니다. 하지만 판매 등록을 취소하고 새 파라미터를 지정한 다음 다시 등록하는 방식으로 변경하는 것은 가능합니다.

현재 active(활성) 상태가 아닌 항목에 대해 언제든지 판매 등록을 취소할 수 있습니다. 구매자의 검색 결과에 일치하는 항목으로 선정되어 이미 판매 처리 중인 항목은 취소할 수 없습니다. 판매 등록을 취소한 시점에서 이 등록에 속하는 일부 예약 인스턴스가 이미 판매 선정되었다면, 선정된 인스턴스를 제외한 인스턴스만 판매 등록이 취소됩니다.

기본적으로 예약 인스턴스의 가격은 시간이 지날수록 떨어지므로 AWS는 매달 일정 금액씩 가격이 내려가도록 가격을 설정할 수 있습니다. 하지만 판매자는 예약 판매 시점을 기준으로 선결제 가격을 다르게 설정할 수 있습니다.

예를 들어 사용 기간이 9개월 남은 예약 인스턴스를 판매하는 경우, 9개월이라는 기간이 남아 있는 동안 이 예약 인스턴스를 구매하는 구매자에게 받을 금액을 설정할 수 있습니다. 남은 기간이 5개월인 시점과 1개월인 시점에서의 판매 가격을 각각 책정할 수 있습니다.

예약 인스턴스 나열

등록된 판매자는 예약 인스턴스를 한 개 이상 판매하기로 선택할 수 있습니다. 한 번의 판매 등록으로 모두 판매하거나 부분적으로 판매하기로 선택할 수 있습니다. 뿐만 아니라 인스턴스 유형, 플랫폼 및 범위의 구성으로 예약 인스턴스를 판매 등록할 수 있습니다.

콘솔에서 제안 가격이 결정됩니다. 콘솔은 예약 인스턴스와 일치하는 제품을 점검하고 해당 제품을 최저 가격과 일치시킵니다. 그렇지 않으면 남은 시간 동안 예약 인스턴스 비용을 기반으로 제안 가격을 계산합니다. 계산된 값이 \$1.01보다 낮은 경우 제안 가격은 \$1.01입니다.

판매 등록을 취소할 때 인스턴스 중 일부가 이미 판매되었다면, 이미 판매된 인스턴스에 대해서는 취소가 적용되지 않습니다. 아직 판매되지 않은 인스턴스만 예약 인스턴스 마켓플레이스 판매 목록에서 삭제됩니다.

AWS Management 콘솔을 사용하여 예약 인스턴스 마켓플레이스에서 예약 인스턴스를 판매 등록하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 예약 인스턴스를 선택합니다.
3. 판매 등록할 예약 인스턴스를 선택하고 예약 인스턴스 판매를 선택합니다.
4. 예약 예약 인스턴스 구성 페이지에서 판매할 인스턴스의 수와, 남은 사용 기간에 대한 선결제 금액을 해당 열에 설정합니다. 남은 개월 수 열 옆의 화살표를 선택하여 남은 사용 기간에 따라 예약 가격이 어떻게 변경되는지 확인해 보십시오.
5. 절차에 익숙한 고급 사용자가 따로 가격 책정을 원하는 경우, 개월 수에 따라 각각 다른 금액을 설정할 수 있습니다. 일정 금액씩 하락되는 기본 설정으로 돌아가려면 재설정을 선택합니다.
6. 판매 등록 구성을 마쳤으면 계속을 선택합니다.
7. 예약 예약 인스턴스 확인 페이지에 표시된 세부 정보를 확인하고 그대로 진행하려면 예약 인스턴스 리스트를 선택합니다.

콘솔에서 등록 상품을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 예약 인스턴스를 선택합니다.
3. 판매 등록한 예약 인스턴스를 선택하고 내 항목을 선택합니다.

AWS CLI를 사용하여 예약 인스턴스 마켓플레이스에서 예약 인스턴스를 관리하려면

1. `describe-reserved-instances` 명령을 사용하여 예약 인스턴스 목록을 가져옵니다.
2. 판매 등록할 예약 인스턴스의 ID를 기록하고 `create-reserved-instances-listing`을 호출합니다. 예약 인스턴스의 ID, 인스턴스의 수 및 가격표를 지정해야 합니다.
3. 판매 등록을 보려면 `describe-reserved-instances-listings` 명령을 사용합니다.
4. 판매 등록을 취소하려면 `cancel-reserved-instances-listings` 명령을 사용합니다.

예약 인스턴스 항목 상태

다음과 같이 예약 인스턴스 페이지의 내 항목 탭에 있는 항목 상태에 판매 등록의 현재 상태가 표시됩니다.

항목 상태에서 표시되는 정보는 예약 인스턴스 마켓플레이스에 판매 등록된 항목의 상태입니다. 이 상태 정보는 예약 인스턴스 페이지의 상태 열에 표시되는 상태 정보와는 다릅니다. 이 상태 정보는 보유한 예약의 상태입니다.

- active - 구매 가능한 항목입니다.
- cancelled - 판매 등록이 취소되어 예약 인스턴스 마켓플레이스에서 구매할 수 없습니다.
- closed - 판매 등록되지 않은 예약 인스턴스입니다. 항목 판매가 완료된 예약 인스턴스의 경우에도 상태가 closed로 표시됩니다.

항목 기간

등록된 항목의 모든 인스턴스가 판매 완료된 경우, 내 항목 탭의 전체 인스턴스 수(Total instance count)의 값이 품질 항목의 값과 동일합니다. 또한 사용 가능 인스턴스가 더 이상 존재하지 않는 것을 확인할 수 있습니다. 상태 항목은 closed로 표시됩니다.

항목 중 일부 인스턴스만 판매된 경우 AWS에서 이 등록 항목에서 판매된 예약 인스턴스를 빼고 판매되지 않은 예약 인스턴스와 동일한 개수의 새 예약 인스턴스를 생성합니다. 따라서 판매 등록 ID와 해당 판매 등록은 활성 상태로 유지되지만, 남은 예약 인스턴스 수는 줄어듭니다.

이후 이 등록 항목에서 예약 인스턴스가 판매될 때마다 이같은 절차가 반복됩니다. 등록 항목에 있는 모든 예약 인스턴스가 판매되면 AWS에서 해당 등록 항목을 closed로 표시합니다.

예를 들어 예약 인스턴스 listing ID 5ec28771-05ff-4b9b-aa31-9e57dexample 항목으로 5개의 인스턴스를 판매 등록했다고 가정해 보겠습니다.

이때 콘솔의 예약 인스턴스 페이지를 열었을 때 내 항목 탭에 다음 정보가 표시됩니다.

예약 인스턴스 listing ID 5ec28771-05ff-4b9b-aa31-9e57dexample

- 전체 예약 인스턴스 개수 = 5
- Sold = 0
- Available = 5
- Status = active

구매자가 예약 중 2개를 구입한 경우 이제 판매 가능한 예약의 수는 3개가 됩니다. AWS에서는 이 부분 판매에 따라 인스턴스 개수가 세 개인 새로운 예약을 생성하며, 이 인스턴스 개수는 아직 판매 중인 인스턴스를 의미합니다.

새롭게 변경된 정보는 내 항목 탭에 다음과 같이 나타납니다.

예약 인스턴스 listing ID 5ec28771-05ff-4b9b-aa31-9e57dexample

- 전체 예약 인스턴스 개수 = 5
- Sold = 2
- Available = 3
- Status = active

판매 등록을 취소할 때 인스턴스 중 일부가 이미 판매되었다면, 이미 판매된 인스턴스에 대해서는 취소가 적용되지 않습니다. 아직 판매되지 않은 인스턴스만 예약 인스턴스 마켓플레이스 판매 목록에서 삭제됩니다.

예약 인스턴스 판매 후 절차

예약 인스턴스가 판매되면 AWS에서 이메일로 이를 알립니다. 어떤 활동이 발생하면 당일에 발생한 모든 활동 내역이 이메일로 발송됩니다. 판매를 등록하거나, 등록 상품을 판매하거나, AWS에서 대금 송금하는 활동 등이 있습니다.

콘솔에서 판매 등록된 예약 인스턴스 목록의 상태를 조회하려면 예약 인스턴스, 내 항목을 선택합니다. 내 항목 탭에는 항목 상태 값이 표시됩니다. 또한 사용 기간, 판매 가격, 등록 항목에서 Available(판매 가능), Pending(보류), Sold(판매), Cancelled(취소) 상태의 인스턴스 개수 정보도 제공됩니다. 또한 [describe-reserved-instances-listings](#) 명령을 통해 필터를 사용하여 판매 등록에 대한 정보를 알아볼 수도 있습니다.

판매 대금 정산

AWS는 구매자가 결제를 완료하자마자 판매된 해당 예약 인스턴스의 소유자로 등록된 계정 이메일 주소로 메시지를 보내 이를 알립니다.

AWS는 ACH(자동 결제) 시스템을 통해 지정된 은행 계좌로 송금합니다. 일반적인 송금 시기는 예약 인스턴스가 판매된 후 1일에서 3일 사이입니다. 지불은 매일 한 번 실시됩니다. 대금이 지급된 후 지급금 보고서가 포함된 이메일이 전송됩니다. AWS에서 은행으로부터 계좌를 확인받기 전에는 대금이 지불되지 않으므로 이 점에 유의하십시오. 이 절차는 최대 2주가 소요됩니다.

사용자가 판매한 예약 인스턴스는 사용자가 예약 인스턴스를 설명할 때 계속 표시됩니다.

예약 인스턴스를 판매한 대금은 현금으로 지급되며 판매자 명의의 은행 계좌로 직접 송금됩니다. AWS는 예약 인스턴스 마켓플레이스에서 판매하는 각 표준 예약 인스턴스에 대해 총 선결제 금액의 12%를 서비스 수수료로 청구합니다.

구매자와의 정보 공유

예약 인스턴스 마켓플레이스에서 판매할 경우 AWS는 미국 규정에 따라 구매자 명세서에 판매자의 상호명을 기재하여 제공합니다. 또한 구매자가 인보이스 또는 다른 세금 관련 이유에 대해 문의하기 위해 AWS

Support에 요청한 경우, AWS에서 구매자가 직접 연락을 취할 수 있도록 판매자의 이메일 주소를 제공해야 할 수 있습니다.

이와 비슷한 이유로 판매자의 지불 내역서에는 구매자의 지역번호(우편번호)와 국가 정보가 제공됩니다. 이 정보는 판매자 측에서 거래에 따라 정부에 납부해야 하는 세금(예: 매출세, 부가가치세)이 발생하는 경우, 이런 세금을 정산하는 데 필요합니다.

AWS에서는 세금에 대해 조언하지 않습니다. 단, 회사의 세금 전문 담당자가 특정 정보를 추가로 요청한 경우에는 [AWS Support에 문의](#)하십시오.

예약 인스턴스 Marketplace에서 구입

예약 인스턴스 마켓플레이스에서 더 이상 필요하지 않은 예약 인스턴스를 소유한 타사 판매업자로부터 예약 인스턴스를 구입할 수 있습니다. Amazon EC2 콘솔 또는 명령줄 도구를 사용하여 이 작업을 수행할 수 있습니다. 이 프로세스는 AWS에서 예약 인스턴스를 구입하는 것과 비슷합니다. 자세한 내용은 [예약 인스턴스 구입 \(p. 217\)](#) 단원을 참조하십시오.

예약 인스턴스 마켓플레이스에서 구입하는 예약 인스턴스와 AWS에서 직접 구입하는 예약 인스턴스 간에는 다음과 같은 몇 가지 차이점이 있습니다.

- 기간 - 타사에서 구입하는 예약 인스턴스는 남은 기간이 표준 약정 기간보다 짧습니다. AWS의 표준 약정 기간은 1년 또는 3년입니다.
- 선결제 요금 - 타사 예약 인스턴스는 다양한 선결제 요금으로 판매될 수 있습니다. 사용 요금이나 기본 요금은 AWS에서 예약 인스턴스를 처음에 구입했을 때 설정된 요금과 동일하게 유지됩니다.
- 예약 인스턴스의 유형 - 예약 인스턴스 마켓플레이스에서 Amazon EC2 표준 예약 인스턴스만 구입할 수 있습니다. 전환형 예약 인스턴스, Amazon RDS 및 Amazon ElastiCache 예약 인스턴스는 예약 인스턴스 마켓플레이스에서 구입할 수 없습니다.

귀하에 대한 기본 정보(우편번호 및 국가 정보 등)는 판매자와 공유됩니다.

이 정보는 판매자가 정부에 납부해야 하는 거래세(판매세, 부가가치세 등)을 계산하는 데 필요하며, 지급 내역서 형태로 제공됩니다. 드문 경우지만 판매자가 거래와 관련하여 문의할 수 있도록(세금 관련 질문 등) AWS에서 판매자에게 구매자의 이메일 주소를 제공할 수 있습니다.

또한 AWS에서 구매자에게 제공하는 구매 인보이스에는 판매자의 법인 이름이 표기됩니다. 세금이나 관련 이유로 인해 판매자에 대한 추가 정보가 필요할 경우 [AWS Support](#)로 문의하십시오.

예약 인스턴스 수정

변화가 생긴 경우 표준 또는 전환형 예약 인스턴스를 변경함으로써 요금 혜택에 따른 이점을 계속 유지할 수 있습니다. 가용 영역 및 예약 인스턴스의 범위와 같은 속성을 수정할 수 있습니다.

Note

전환형 예약 인스턴스를 구성이 다른 전환형 예약 인스턴스와 교환할 수도 있습니다. 자세한 내용은 [전환형 예약 인스턴스 교환 \(p. 232\)](#) 단원을 참조하십시오.

수정 후 예약 인스턴스의 혜택은 새로운 파라미터와 일치하는 인스턴스에만 적용됩니다. 예를 들어, 예약의 가용 영역을 변경할 경우 용량 예약 및 요금 혜택이 새로운 가용 영역의 인스턴스 사용에 자동으로 적용됩니다. 새 파라미터와 일치하지 않는 인스턴스는 계정의 다른 예약 내역 할인이 적용되지 않는 한 온디맨드 요금이 부과됩니다.

변경 요청이 성공한 경우:

- 변경된 예약이 즉시 적용되고 변경 요청 시점을 기준으로 새 인스턴스에 요금 혜택이 적용됩니다. 예를 들어, 예약 변경이 성공적으로 완료된 시간이 오후 9시 15분이라면, 요금 혜택은 오후 9시부터 새 인스턴스에 적용됩니다. 변경된 예약 인스턴스의 유효 날짜는 [describe-reserved-instances](#) 명령을 사용하여 확인할 수 있습니다.

- 본래 예약이 종료됩니다. 이 예약의 종료일은 새로운 예약의 시작일이 되며, 새 예약의 종료일은 본래 예약 인스턴스의 종료일과 동일합니다; 3년 약정 예약 중 16개월 남은 시점에서 변경했다면, 변경된 예약은 16 개월 동안 사용이 가능하며 본래 예약의 종료일과 같은 날짜에 사용 기간이 만료됩니다.
- 변경된 예약의 고정 가격은 본래 예약의 고정 가격이 아닌 \$0로 표시됩니다.
- 변경된 예약의 고정 가격은 계정에 적용되는 할인 요금 티어에는 영향을 주지 않습니다. 할인 요금 티어는 본래 예약의 고정 가격을 기준으로 하기 때문입니다.

수정 요청이 실패할 경우 예약 인스턴스는 원래의 구성을 유지하며 다른 수정 요청이 즉시 가능합니다.

수정 비용이 없기 때문에 새로운 청구서나 인보이스를 수신하지 않습니다.

원하는 만큼 예약을 수정할 수 있지만 제출한 후에는 보류 중인 수정 요청을 변경하거나 취소할 수 없습니다. 수정이 성공적으로 처리된 후에는 필요한 경우 변경 전 상태로 되돌리기 위해 또 다른 변경 요청을 제출할 수 있습니다.

목차

- [수정 요건 및 제한 사항 \(p. 230\)](#)
- [수정 요청 제출 \(p. 231\)](#)
- [수정 요청 문제 해결 \(p. 232\)](#)

수정 요건 및 제한 사항

이러한 속성을 다음과 같이 수정할 수 있습니다.

수정 가능한 속성	지원되는 플랫폼	제한 사항
같은 리전 내에서 가용 영역 변경	Linux 및 Windows	-
가용 영역에서 리전으로 범위 변경(반대 방향도 마찬가지)	Linux 및 Windows	범위를 가용 영역에서 리전으로 변경할 경우 용량 예약 혜택을 받을 수 없습니다. 범위를 리전에서 가용 영역으로 변경하면 가용 영역 유연성과 인스턴스 크기 유연성이 사라집니다(있는 경우). 자세한 내용은 예약 인스턴스 적용 방식 (p. 209) 을 참조하십시오.
동일한 인스턴스 패밀리 내에서 인스턴스 크기를 변경합니다.	Linux/UNIX 전용 SQL Server Standard가 설치된 Linux, SQL Server Web이 설치된 Linux, SQL Server Enterprise가 설치된 Linux, Red Hat Enterprise Linux, SUSE Linux, Windows, SQL Standard가 설치된 Windows, SQL Server Enterprise가 설치된 Windows, SQL Server Web이 설치된 Windows를 비롯한 다른 플랫폼에서는 예약 인스턴스에 인스턴스 크기 유연성이 제공되지 않습니다.	예약은 기본 테넌시를 사용해야 합니다. 사용 가능한 다른 크기가 없으므로 일부 인스턴스 패밀리는 지원되지 않습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 인스턴스 크기 수정을 위한 지원 단원 을 참조하십시오.

수정 가능한 속성	지원되는 플랫폼	제한 사항
EC2-Classic에서 Amazon VPC로 네트워크 변경(그 반대도 가능)	Linux 및 Windows	네트워크 플랫폼은 AWS 계정에서 사용할 수 있어야 합니다. 2013년 12월 4일 이후 AWS 계정을 생성한 경우에는 EC2-Classic을 지원하지 않습니다.

요구 사항

Amazon EC2에서는 대상 구성에 사용할 수 있는 용량이 충분히 남아 있고(해당되는 경우) 다음 조건을 충족하는 경우 수정 요청을 처리합니다.

- 구입 전이나 구입 당시에는 예약 인스턴스를 수정할 수 없습니다.
- 예약 인스턴스는 활성 상태여야 합니다.
- 보류 중인 수정 요청이 있을 수 없습니다.
- 예약 인스턴스가 예약 인스턴스 마켓플레이스에 등록되어 있지 않습니다.
- 입력 예약 인스턴스는 유형 혼합 없이 모두 표준 예약 인스턴스이거나 모두 전환형 예약 인스턴스입니다.
- 입력 예약 인스턴스가 표준 예약 인스턴스인 경우 동일한 시간 내에 만료되어야 합니다.
- 예약 인스턴스는 G4 인스턴스가 아닙니다.

수정 요청 제출

예약 인스턴스를 수정하기 전에 해당하는 [제한 사항 \(p. 230\)](#)을 읽어야 합니다.

AWS Management 콘솔을 사용하여 예약 인스턴스를 수정하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 예약 인스턴스 페이지에서 수정할 예약 인스턴스를 하나 이상 선택하고 작업, 예약 인스턴스 수정을 선택합니다.

Note

예약 인스턴스가 활성 상태가 아니거나 수정이 불가능한 경우 예약 인스턴스 수정이 비활성화 됩니다.

- 수정 표의 첫 항목은 선택한 예약 인스턴스의 속성을 표시하고 그 아래에는 하나 이상의 대상 구성이 표시됩니다. 단위 열에는 총 인스턴스 공간 크기가 표시됩니다. 추가할 새 구성 각각에 대해 추가를 선택합니다. 각 구성에 대해 필요에 따라 속성을 수정한 다음 계속을 선택합니다.
 - 범위: 구성이 적용 영역에 적용되는지 아니면 전체 리전에 적용되는지 선택합니다.
 - 가용 영역: 필요한 가용 영역을 선택합니다. 리전 단위의 예약 인스턴스에는 적용되지 않습니다.
 - 개수: 인스턴스 수를 지정합니다. 예약 인스턴스를 여러 구성으로 분할하려면 개수를 줄이고 추가를 선택한 후 추가 구성의 개수를 지정합니다. 예를 들어 개수가 10인 단일 구성이 있으면 개수를 6으로 변경하고 개수가 4인 구성을 추가할 수 있습니다. 이 프로세스에서는 새 예약 인스턴스가 활성화되면 원래의 예약 인스턴스를 중지합니다.
- 원하는 대로 구성을 지정하고 변경 사항을 확인하려면 수정 사항 제출을 선택합니다.
- 예약 인스턴스 화면의 상태 열을 확인하여 수정 요청의 상태를 알 수 있습니다. 가능한 상태 표시는 다음과 같습니다.
 - active(수정 보류 중) - 기존 예약 인스턴스의 전환 상태
 - retired(수정 보류 중) - 새 예약 인스턴스가 생성되는 동안 기존 예약 인스턴스의 전환 상태
 - retired - 예약 인스턴스가 수정되어 교체되었습니다.

- active - 다음 중 하나입니다.
 - 수정 요청이 성공한 경우 생성된 새 예약 인스턴스의 상태입니다.
 - 수정 요청이 실패한 후 원래 예약 인스턴스의 상태입니다.

명령줄을 사용하여 예약 인스턴스를 수정하는 방법

1. 예약 인스턴스를 수정하려면 다음 명령 중 하나를 사용할 수 있습니다.
 - [modify-reserved-instances](#)(AWS CLI)
 - [Edit-EC2ReservedInstance](#)(Windows PowerShell용 AWS 도구)
2. 수정 요청 상태(`processing`, `fulfilled` 또는 `failed`)를 확인하려면 다음 명령 중 하나를 사용하십시오.
 - [describe-reserved-instances-modifications](#)(AWS CLI)
 - [Get-EC2ReservedInstancesModification](#)(Windows PowerShell용 AWS 도구)

수정 요청 문제 해결

요청한 변경 항목이 중복되지 않는 고유한 설정이라면 요청을 처리 중이라는 메시지가 표시됩니다. 이 시점에서는 Amazon EC2에서 변경 요청의 파라미터가 유효함을 확인한 상태입니다. 처리 과정에서 용량이 부족해 변경 요청이 실패할 가능성은 여전히 존재합니다.

일부의 경우, 확인 메시지 대신 완료 실패나 변경 실패 메시지가 표시될 수 있습니다. 메시지에 표시된 정보는 변경 요청을 다시 신청하는 데 참고 기준으로 사용하면 도움이 됩니다. 요청을 제출하기 전에 해당하는 [제한 사항](#) (p. 230)을 읽어 보십시오.

선택한 예약 인스턴스 모두를 변경할 수 있도록 처리할 수 있는 것은 아닙니다.

Amazon EC2에서는 변경할 수 없는 예약 인스턴스를 식별하여 표시합니다. 이 메시지가 표시되었다면 Amazon EC2 콘솔의 예약 인스턴스 페이지로 이동하여 예약 인스턴스에 대한 정보를 확인하십시오.

변경 요청을 처리하는 동안 오류가 발생했습니다

하나 이상의 예약 인스턴스의 변경을 요청한 후 이 중 어떤 요청도 처리할 수 없을 때 표시되는 메시지입니다. 변경을 시도한 예약의 개수에 따라 다른 버전의 메시지가 표시될 수 있습니다.

Amazon EC2에서 요청을 처리할 수 없는 이유를 표시합니다. 예를 들어 수정하려는 예약 인스턴스의 하위 집합 중 하나 이상에 대해 동일한 대상 구성(가용 영역과 플랫폼 조합)을 지정했을 수 있습니다. 예약의 인스턴스 세부 정보가 일치하는지와 예약의 모든 하위 그룹에 대해 요청한 변경 사항이 서로 겹치지 않는지를 확인한 다음, 변경 요청을 다시 시도해 봅니다.

전환형 예약 인스턴스 교환

한 개 이상의 전환형 예약 인스턴스를 인스턴스 패밀리와 운영 체제, 테넌트를 비롯하여 구성이 다른 전환형 예약 인스턴스와 교환할 수 있습니다. 교환 횟수에 제한은 없습니다. 단, 해당 전환형 예약 인스턴스가 교환하려는 전환형 예약 인스턴스보다 가격이 높거나 같아야 합니다.

전환형 예약 인스턴스를 교환하면 현재 예약에 대한 인스턴스의 수가 대상 전환형 예약 인스턴스의 구성 값 보다 크거나 같은 여러 인스턴스로 교환됩니다. Amazon EC2는 교환한 결과로 받을 수 있는 예약 인스턴스의 수를 계산합니다.

목차

- [전환형 예약 인스턴스 교환 요건](#) (p. 233)
- [전환형 예약 인스턴스 교환 계산](#) (p. 234)
- [전환형 예약 인스턴스 병합](#) (p. 234)

- [전환형 예약 인스턴스의 일부분 교환 \(p. 235\)](#)
- [교환 요청 제출 \(p. 235\)](#)

전환형 예약 인스턴스 교환 요건

Amazon EC2에서는 다음 조건이 충족될 경우 교환 요청을 처리합니다. 전환형 예약 인스턴스가 다음 조건을 충족해야 합니다.

- 활성 상태
- 이전 교환 요청이 보류 중이지 않음

다음 규칙이 적용됩니다.

- 전환형 예약 인스턴스는 AWS에서 현재 제공하는 다른 전환형 예약 인스턴스하고만 교환할 수 있습니다.
- 전환형 예약 인스턴스는 예약 기간 동안 고정된 특정 리전과 연결됩니다. 전환형 예약 인스턴스를 다른 리전의 전환형 예약 인스턴스와 교환할 수 없습니다.
- 한 번에 전환형 예약 인스턴스 하나만 한 개 이상의 전환형 예약 인스턴스로 교환할 수 있습니다.
- 전환형 예약 인스턴스의 일부분을 교환하려면 둘 이상의 예약으로 수정한 다음 예약의 한 개 이상을 새 전환형 예약 인스턴스로 교환하면 됩니다. 자세한 내용은 [전환형 예약 인스턴스의 일부분 교환 \(p. 235\)](#) 단원을 참조하십시오. 예약 인스턴스 변경에 대한 자세한 내용은 [예약 인스턴스 수정 \(p. 229\)](#) 단원을 참조하십시오.
- 전체 선결제 전환형 예약 인스턴스를 부분 선결제 전환형 예약 인스턴스로 교환할 수 있으며 그 반대로도 교환할 수 있습니다.

Note

교환에 필요한 총 선결제 금액(트루업 요금)이 \$0.00 미만인 경우 AWS는 트루업 요금이 \$0.00 이상이 되도록 전환형 예약 인스턴스에 인스턴스 수량을 자동으로 제공합니다.

Note

새로운 전환형 예약 인스턴스의 총 금액(선결제 금액 + 시간당 요금 * 남은 시간 수)이 교환한 전환형 예약 인스턴스의 총 금액보다 낮은 경우 AWS는 총 금액이 교환한 전환형 예약 인스턴스의 금액과 같거나 그 이상이 되도록 전환형 예약 인스턴스에 인스턴스 수량을 자동으로 제공합니다.

- 요금 혜택을 더 받기 위해 선결제 없음 전환형 예약 인스턴스를 전체 선결제 또는 부분 선결제 전환형 예약 인스턴스와 교환할 수 있습니다.
- 전체 선결제 또는 부분 선결제 전환형 예약 인스턴스를 선결제 없음 전환형 예약 인스턴스와 교환할 수는 없습니다.
- 새로운 전환형 예약 인스턴스의 시간당 가격이 교환한 전환형 예약 인스턴스의 시간당 가격과 같거나 그 이상인 경우에만 선결제 없음 전환형 예약 인스턴스를 다른 선결제 없음 전환형 예약 인스턴스와 교환할 수 있습니다.

Note

새로운 전환형 예약 인스턴스의 총 금액(시간당 요금 * 남은 시간 수)이 교환한 전환형 예약 인스턴스의 총 금액보다 낮은 경우 AWS는 총 금액이 교환한 전환형 예약 인스턴스의 금액과 같거나 그 이상이 되도록 전환형 예약 인스턴스에 인스턴스 수량을 자동으로 제공합니다.

- 만료 날짜가 서로 다른 여러 전환형 예약 인스턴스를 교환하는 경우 새 전환형 예약 인스턴스의 만료 날짜는 더 나중에 오는 날짜입니다.
- 단일 전환형 예약 인스턴스를 교환하는 경우 새 전환형 예약 인스턴스와 기간이 동일해야 합니다(1년 또는 3년). 기간 길이가 다른 여러 전환형 예약 인스턴스를 병합하는 경우 새 전환형 예약 인스턴스의 기간은 3년입니다. 자세한 내용은 [전환형 예약 인스턴스 병합 \(p. 234\)](#) 단원을 참조하십시오.
- 전환형 예약 인스턴스 교환 후 원래 예약은 종료됩니다. 원래 예약의 종료일은 새로운 예약의 시작일이 되며, 새 예약의 종료일은 원래 전환형 예약 인스턴스의 종료일과 동일합니다. 예를 들어, 3년 약정 예약 중

16개월 남은 시점에서 변경했다면, 변경된 예약은 16개월 동안 사용이 가능하며 본래 예약의 종료일과 같은 날짜에 사용 기간이 만료됩니다.

전환형 예약 인스턴스 교환 계산

전환형 예약 인스턴스 교환은 무료입니다. 하지만 트루업(true-up) 비용을 지불해야 할 수 있습니다. 이 비용은 소유했던 전환형 예약 인스턴스와 교환을 통해 받는 전환형 예약 인스턴스 간의 차이를 비례 할당으로 계산한 선결제 비용입니다.

각 전환형 예약 인스턴스에는 정가가 있습니다. 교환의 결과로 받을 수 있는 인스턴스 예약의 수를 결정하기 위해 이 정가를, 원하는 전환형 예약 인스턴스의 정가와 비교합니다.

정가가 \$35인 전환형 예약 인스턴스 1개를 정가가 \$10인 새 인스턴스 유형과 교환하려는 경우를 예로 들어 보겠습니다.

\$35 / \$10 = 3.5

전환형 예약 인스턴스를 10 USD 전환형 예약 인스턴스 세 개와 교환할 수는 없습니다. 절반의 동일 시작 인스턴스를 구입할 수는 없으므로 전환형 예약 인스턴스를 추가로 구입하여 나머지를 채워야 합니다.

3.5 = 3 whole #### ## ##### + 1 additional #### ## #####.

4번째 전환형 예약 인스턴스는 다른 3개와 종료 날짜가 동일합니다. 부분 선결제 또는 전체 선결제 전환형 예약 인스턴스를 교환할 경우 4번째 동일 시작 인스턴스에 대해 트루업 비용을 지불하게 됩니다. 전환형 예약 인스턴스의 나머지 선결제 비용이 500 USD이고 대상 동일 시작 인스턴스가 비례 할당 계산 기준으로 600 USD일 경우 100 USD가 청구됩니다.

\$600 prorated upfront cost of new reservations - \$500 remaining upfront cost of original reservations = \$100 difference.

전환형 예약 인스턴스 병합

둘 이상의 전환형 예약 인스턴스를 병합하는 경우 새 전환형 예약 인스턴스의 기간은 원래 전환형 예약 인스턴스와 동일하거나 원래 전환형 예약 인스턴스보다 길어야 합니다. 새 전환형 예약 인스턴스의 만료 날짜는 더 나중에 오는 만료 날짜입니다.

예를 들어 계정에 다음과 같은 전환형 예약 인스턴스가 있다고 가정합시다.

예약 인스턴스 ID	기간	만료 날짜
aaaa1111	1년	2018-12-31
bbbb2222	1년	2018-07-31
cccc3333	3년	2018-06-30
dddd4444	3년	2019-12-31

- aaaa1111과 bbbb2222를 병합하여 1년 전환형 예약 인스턴스로 변경할 수 있습니다. 3년 전환형 예약 인스턴스로는 변경할 수 없습니다. 새 전환형 예약 인스턴스의 만료 날짜는 2018-12-31입니다.
- bbbb2222과 cccc3333를 병합하여 3년 전환형 예약 인스턴스로 변경할 수 있습니다. 1년 전환형 예약 인스턴스로는 변경할 수 없습니다. 새 전환형 예약 인스턴스의 만료 날짜는 2018-07-31입니다.

- cccc3333과 dddd4444를 병합하여 3년 전환형 예약 인스턴스로 변경할 수 있습니다. 1년 전환형 예약 인스턴스로는 변경할 수 없습니다. 새 전환형 예약 인스턴스의 만료 날짜는 2019-12-31입니다.

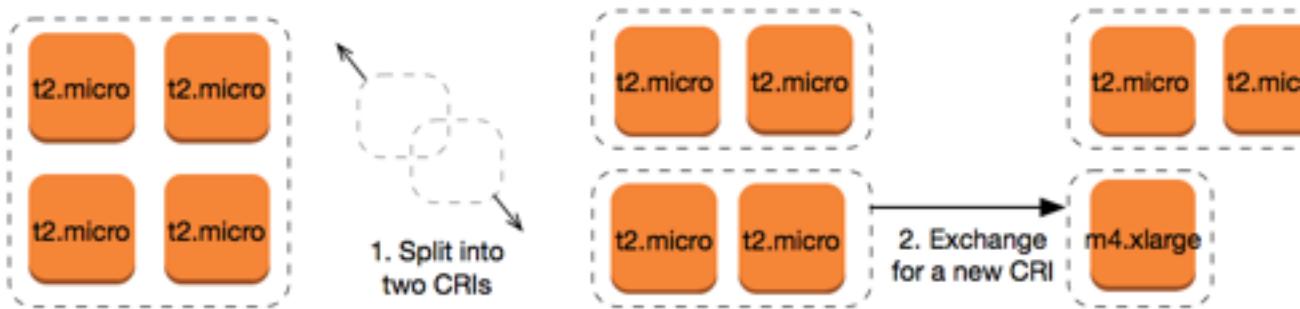
전환형 예약 인스턴스의 일부분 교환

전환형 예약 인스턴스를 작은 예약으로 분리하는 수정 과정을 이용한 다음 새 예약의 한 개 이상을 새 전환형 예약 인스턴스로 교환하면 됩니다. 다음은 그 방법을 설명하는 예제입니다.

Example 예: 인스턴스가 여럿인 전환형 예약 인스턴스

이 예에는 예약에 네 인스턴스가 있는 t2.micro 전환형 예약 인스턴스가 있습니다. 두 t2.micro 인스턴스를 m4.xlarge 인스턴스 한 개로 교환하려면,

1. t2.micro 전환형 예약 인스턴스를 각각 두 인스턴스를 가진 두 t2.micro 전환형 예약 인스턴스로 분할하여 수정합니다.
2. 새 t2.micro 전환형 예약 인스턴스 중 하나를 m4.xlarge 전환형 예약 인스턴스 하나와 교환합니다.



교환 요청 제출

Amazon EC2 콘솔 또는 명령줄 도구를 사용하여 전환형 예약 인스턴스를 교환할 수 있습니다.

콘솔을 사용한 전환형 예약 인스턴스 교환

전환형 예약 인스턴스 상품을 검색하고 제공된 선택지에서 새로운 구성을 선택할 수 있습니다.

Amazon EC2 콘솔을 사용해 전환형 예약 인스턴스를 교환하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 예약 인스턴스를 선택하여 교환할 전환형 예약 인스턴스를 선택한 후 작업, 예약 인스턴스 교환을 선택합니다.
3. 드롭다운 메뉴를 사용하여 원하는 구성의 속성을 선택하고 제품 찾기를 선택합니다.
4. 새 전환형 예약 인스턴스를 선택합니다. 인스턴스 수 옆에는 교환에 따라 받은 예약 인스턴스의 수가 표시됩니다. 요구 사항에 맞는 전환형 예약 인스턴스를 선택한 경우 교환을 선택합니다.

교환된 예약 인스턴스는 만료되고 새로운 Amazon EC2가 예약 인스턴스 콘솔에 표시됩니다. 이 프로세스는 완료하는 데 몇 분 정도 걸릴 수 있습니다.

명령줄 인터페이스를 사용한 전환형 예약 인스턴스 교환

전환형 예약 인스턴스를 교환하려면 먼저 요구 사항에 맞는 대상 전환형 예약 인스턴스를 찾습니다.

- `describe-reserved-instances-offerings`(AWS CLI)

- [Get-EC2ReservedInstancesOffering](#)(Windows PowerShell용 도구)

교환에서 가져오는 예약 인스턴스의 수 및 교환에 대한 트루업 비용이 포함된 교환 견적서를 가져옵니다.

- [get-reserved-instances-exchange-quote](#)(AWS CLI)
- [GetEC2-ReservedInstancesExchangeQuote](#)(Windows PowerShell용 도구)

마지막으로 교환을 수행합니다.

- [accept-reserved-instances-exchange-quote](#)(AWS CLI)
- [Confirm-EC2ReservedInstancesExchangeQuote](#)(Windows PowerShell용 도구)

정기 예약 인스턴스

정기 예약 인스턴스(예약된 인스턴스)를 사용하여 1년 동안 지정된 시작 시간 및 기간에 따라 매일, 매주 또는 매월 반복적으로 용량 예약을 구입할 수 있습니다. 필요할 때 사용할 수 있도록 용량을 미리 예약합니다. 인스턴스를 사용하지 않더라도 인스턴스가 예약된 시간에 대한 비용을 지불합니다.

정기 인스턴스는 지속적으로 실행되지 않지만, 정기적으로 실행되고 정해진 시간에 완료되는 워크로드에 적합한 옵션입니다. 예를 들어, 업무 시간 중에 실행되는 애플리케이션 또는 주말에 실행되는 일괄 처리에 대해 정기 인스턴스를 사용할 수 있습니다.

지속적으로 용량 예약이 필요한 경우 예약 인스턴스를 사용하면 필요한 용량을 충족함과 동시에 비용을 절감할 수 있습니다. 자세한 내용은 [예약 인스턴스 \(p. 205\)](#) 단원을 참조하십시오. 인스턴스를 실행하는 시간이 유동적인 경우 스팟 인스턴스를 사용하면 필요한 용량을 충족함과 동시에 비용을 절감할 수 있습니다. 자세한 내용은 [스팟 인스턴스 \(p. 239\)](#) 단원을 참조하십시오.

내용

- 정기 예약 인스턴스의 작동 방식 (p. 236)
- 정기 예약 인스턴스에 대한 서비스 연결 역할 (p. 237)
- 예약된 인스턴스 구매 (p. 237)
- 예약된 인스턴스 시작 (p. 238)
- 정기 예약 인스턴스 제한 (p. 239)

정기 예약 인스턴스의 작동 방식

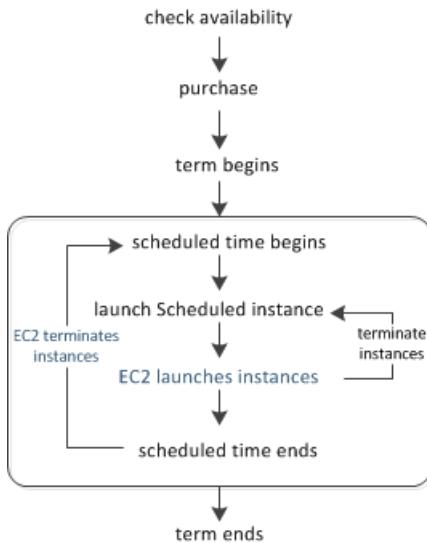
Amazon EC2는 정기 인스턴스로 사용하기 위해 각 가용 영역에서 EC2 인스턴스 풀을 무효화합니다. 각 풀은 인스턴스 유형, 운영 체제 및 네트워크의 특정 조합을 지원합니다.

시작하려면 사용 가능한 일정을 검색해야 합니다. 여러 풀 또는 단일 풀을 검색할 수 있습니다. 적합한 일정을 찾은 다음 해당 일정을 구매합니다.

인스턴스 유형, 가용 영역, 네트워크 및 플랫폼과 같이 구입한 일정의 속성에 일치하는 시작 구성은 사용하여 지정 기간 중에 예약된 인스턴스를 실행해야 합니다. 그러면 Amazon EC2에서는 지정된 시작 사양에 따라 사용자를 대신하여 EC2 인스턴스를 시작합니다. Amazon EC2는 현재 지정 기간이 끝날 때까지 EC2 인스턴스가 종료되도록 함으로써 예약된 다른 정기 인스턴스들에 대한 가용 용량을 확보해야 합니다. 따라서 현재 지정 기간이 끝나기 전에 Amazon EC2에서 EC2 인스턴스를 종료합니다.

정기 인스턴스를 중지하거나 재부팅할 수 없지만, 필요한 경우 수동으로 종료할 수 있습니다. 현재 지정 기간이 종료되기 전에 정기 인스턴스를 종료하는 경우, 몇 분 후에 다시 시작할 수 있습니다. 그렇지 않으면 다음 예약된 시간까지 기다려야 합니다.

다음 그림은 정기 인스턴스의 수명 주기를 보여줍니다.



정기 예약 인스턴스에 대한 서비스 연결 역할

정기 인스턴스를 구입하면 Amazon EC2에서 서비스 연결 역할을 만듭니다. 서비스 연결 역할은 Amazon EC2에서 다른 AWS 서비스를 자동으로 호출하기 위해 필요한 모든 권한을 포함합니다. 자세한 내용은 IAM 사용 설명서의 [서비스 연결 역할 사용](#)을 참조하십시오.

Amazon EC2는 AWSServiceRoleForEC2ScheduledInstances라는 이름의 서비스 연결 역할을 사용하여 다음 작업을 완료합니다.

- `ec2:TerminateInstances` - 일정 완료 후 정기 인스턴스 종료
- `ec2:CreateTags` - 정기 인스턴스에 시스템 태그 추가

Amazon EC2가 이 서비스 연결 역할을 지원하기 시작한 2017년 10월 이전에 정기 인스턴스를 구입한 경우, Amazon EC2에서 사용자 AWS 계정에 AWSServiceRoleForEC2ScheduledInstances 역할을 이미 생성했습니다. 자세한 내용은 IAM 사용 설명서의 [내 계정에 표시되는 새 역할](#)을 참조하십시오.

정기 인스턴스가 더 이상 필요 없는 경우에는 AWSServiceRoleForEC2ScheduledInstances 역할을 삭제할 것을 권합니다. 계정에서 이 역할을 삭제한 후 정기 인스턴스를 구입하면 Amazon EC2에서 다시 해당 역할을 만듭니다.

예약된 인스턴스 구매

정기 인스턴스를 구입하려면 정기 예약 인스턴스 예약 마법사를 사용할 수 있습니다.

Warning

정기 인스턴스를 구입한 이후에는 구입을 취소하거나, 수정하거나, 재판매할 수 없습니다.

정기 인스턴스를 구입하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 인스턴스에서 정기 인스턴스를 선택합니다. 현재 선택한 리전에서 정기 인스턴스를 지원하지 않는 경우 페이지를 사용할 수 없습니다. [자세히 알아보기 \(p. 239\)](#)
3. 정기 인스턴스 구입(Purchase Scheduled Instances)을 선택합니다.

4. 사용 가능한 일정(Find available schedules) 페이지에서 다음을 수행하십시오.

- a. 일정 생성(Create a schedule)의 시작(Starting on)에서 시작 날짜를 선택하고, 반복(Recurring)에서 예약 반복(매일, 매주 또는 매월)을 선택하고, 기간(for duration)에서 최소 기간을 선택합니다. 콘솔에서는 정기 인스턴스에 필요한 최소 사용률을 충족하는 최소 기간 값을 지정하는지 확인합니다(연간 1,200시간).

Create a schedule

The screenshot shows the 'Create a schedule' interface. It includes fields for 'Starting on' (a date), 'for duration' (set to 4 hours), and 'Recurring' (set to Daily). There is also a checkbox for '+/- 2 hours'.

- b. 인스턴스 세부 정보의 플랫폼에서 운영 체제와 네트워크를 선택합니다. 결과 범위를 좁히려면 인스턴스 유형에서 하나 이상의 인스턴스 유형을 선택하거나 가용 영역에서 하나 이상의 가용 영역을 선택합니다.

Instance details

The screenshot shows the 'Instance details' section with dropdown menus for 'Platform' (Linux/UNIX (Amazon VPC)), 'Instance type' (Any), and 'Availability Zone' (Any).

- c. 일정 찾기(Find schedules)를 선택합니다.
d. 사용 가능한 일정(Available schedules)에서 하나 이상의 일정을 선택합니다. 선택하는 각 일정에 대해 인스턴스의 수량을 설정한 다음 장바구니에 추가를 선택합니다.
e. 장바구니가 페이지의 아래쪽에 표시됩니다. 장바구니에서 일정 추가 및 제거를 마쳤으면 검토 및 구입(Review and purchase)을 선택합니다.
5. 검토 및 구입(Review and purchase) 페이지에서 선택 항목을 확인하고 필요한 경우 편집합니다. 작업을 마쳤으면 구입을 선택합니다.

정기 인스턴스를 구입하려면(AWS CLI)

`describe-scheduled-instance-availability` 명령을 사용해 요구 사항을 충족하는 일정 목록을 표시한 다음, `purchase-scheduled-instances` 명령을 사용해 구입을 완료합니다.

예약된 인스턴스 시작

정기 인스턴스를 구입한 후 예약 기간 동안 인스턴스를 시작할 수 있습니다.

정기 인스턴스를 시작하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 인스턴스에서 정기 인스턴스를 선택합니다. 현재 선택한 리전에서 정기 인스턴스를 지원하지 않는 경우 페이지를 사용할 수 없습니다. [자세히 알아보기 \(p. 239\)](#)
3. 정기 인스턴스를 선택하고 정기 인스턴스 시작을 선택합니다.
4. 구성 페이지에서 정기 인스턴스의 시작 사양을 완료한 다음 검토를 선택합니다.

Important

시작 사양이 구입한 일정의 인스턴스 유형, 가용 영역, 네트워크 및 플랫폼에 일치해야 합니다.

5. 검토 페이지에서 시작 구성을 확인하고 필요한 경우 수정합니다. 작업을 마쳤으면 시작을 선택합니다.

정기 인스턴스를 시작하려면(AWS CLI)

`describe-scheduled-instances` 명령을 사용해 정기 인스턴스 목록을 표시한 다음, 예약 기간 동안 `run-scheduled-instances` 명령을 사용해 각 정기 인스턴스를 시작합니다.

정기 예약 인스턴스 제한

정기 인스턴스에는 다음 제한이 적용됩니다.

- C3, C4, M4 및 R3 인스턴스 유형만 지원됩니다.
- 필수 기간은 365일(1년)입니다.
- 최소 필수 사용률은 연간 1,200시간입니다.
- 정기 인스턴스를 최대 3개월 전에 미리 구입할 수 있습니다.
- 이러한 인스턴스는 현재 다음 리전에서 이용 가능합니다. 미국 동부(버지니아 북부), 미국 서부(오리건) 및 유럽(아일랜드).

스팟 인스턴스

스팟 인스턴스는 온디맨드 가격보다 저렴한 비용으로 사용할 수 있는 미사용 EC2 인스턴스입니다. 스팟 인스턴스는 큰 할인율로 미사용 EC2 인스턴스를 요청할 수 있게 해주므로 사용자는 Amazon EC2 비용을 대폭 낮출 수 있습니다. 스팟 인스턴스의 시간당 가격을 스팟 가격이라고 합니다. 각 가용 영역 내 인스턴스 유형 별 스팟 가격은 Amazon EC2에서 설정하며, 스팟 인스턴스의 장기적 공급 및 수요에 따라 점진적으로 조정됩니다. 스팟 인스턴스는 용량이 가용 상태이고 요청에 대한 시간당 최고 가격이 스팟 가격보다 더 높을 때마다 실행됩니다.

스팟 인스턴스는 애플리케이션이 실행되는 시간을 유연하게 조정할 수 있고 애플리케이션을 중단할 수 있는 경우에 선택하는 비용 효율적인 방법입니다. 예를 들어 스팟 인스턴스는 데이터 분석, 배치 작업, 백그라운드 프로세싱 및 선택적 작업에 적합합니다. 자세한 내용은 [Amazon EC2 스팟 인스턴스](#) 단원을 참조하세요.

주제

- [개념 \(p. 239\)](#)
- [시작하는 방법 \(p. 241\)](#)
- [관련 서비스 \(p. 241\)](#)
- [요금 및 비용 절감 \(p. 242\)](#)

개념

스팟 인스턴스를 시작하기 전에 다음 개념을 익혀야 합니다.

- **스팟 인스턴스 풀** – 동일한 인스턴스 유형(예: `m5.large`), 운영 체제, 가용 영역 및 네트워크 플랫폼을 가리는 미사용 EC2 인스턴스의 세트입니다.
- **스팟 가격** – 스팟 인스턴스의 시간당 현재 가격입니다.
- **스팟 인스턴스 요청** – 스팟 인스턴스를 요청합니다. 스팟 인스턴스에 대해 지불하고자 하는 시간당 최고 가격을 제공합니다. 최고 가격을 지정하지 않는 경우, 기본 최고 가격은 온디맨드 가격입니다. 요청에 대한 시간당 최고 가격이 스팟 가격을 초과할 경우, Amazon EC2는 용량이 가용 상태가 되면 요청을 이행합니다. 스팟 인스턴스 요청은 일회성이거나 영구적입니다. Amazon EC2는 요청과 연결된 스팟 인스턴스가 종료된 후 자동으로 영구 스팟 인스턴스 요청을 다시 제출합니다. 스팟 인스턴스 요청은 스팟 인스턴스에 대해 지속 시간을 선택적으로 지정할 수 있는 옵션이 있습니다.

- 스팟 집합 – 사용자가 지정한 기준을 바탕으로 시작되는 스팟 인스턴스 세트입니다. 스팟 집합에서는 사용자의 요구를 충족하는 스팟 인스턴스 풀을 선택하고 플릿에 대한 목표 용량을 충족하는 스팟 인스턴스를 시작합니다. 기본적으로 스팟 집합은 플릿에서 스팟 인스턴스가 종료된 후 교체 인스턴스를 시작하여 목표 용량을 유지하도록 설정되어 있습니다. 스팟 집합을 인스턴스가 종료된 후에는 유지되지 않는 일회성 요청으로 제출할 수도 있습니다. 스팟 집합 요청에 온디맨드 인스턴스 요청을 포함할 수 있습니다.
- 스팟 인스턴스 중단 – 스팟 가격이 요청에 대한 최고 가격을 초과하거나 사용할 수 있는 용량이 더 이상 없는 경우 Amazon EC2는 스팟 인스턴스를 종료 또는 중지시키거나 최대 절전 모드로 전환합니다. Amazon EC2는 스팟 인스턴스 중단 공지를 통해 중지 2분 전에 이를 인스턴스에 경고합니다.

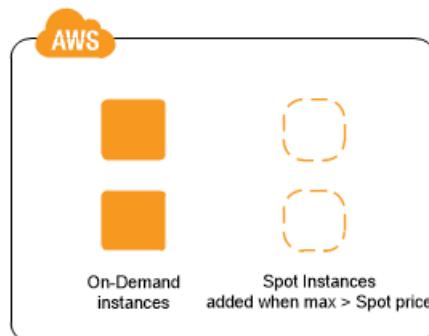
스팟 인스턴스와 온디맨드 인스턴스의 주요 차이점

다음 표에는 스팟 인스턴스와 온디맨드 인스턴스의 주요 차이점이 나열되어 있습니다.

	스팟 인스턴스	온디맨드 인스턴스
시작 시간	스팟 요청이 활성 상태이고 용량이 가용 상태인 경우 즉시 시작할 수 있습니다.	수동 시작을 요청했고 용량이 가용 상태인 경우에만 즉시 시작할 수 있습니다.
가용 용량	용량이 가용 상태가 아닌 경우 용량이 가용 상태가 될 때까지 스팟 요청이 계속해서 자동으로 시작 요청을 합니다.	시작 요청을 할 때 용량이 가용 상태가 아닌 경우 용량 부족 오류(ICE)가 발생합니다.
시간당 가격	스팟 인스턴스의 시간당 가격은 온디맨드 기준 가격입니다.	온디맨드 인스턴스의 시간당 가격은 고정된 가격입니다.
인스턴스 중단	Amazon EBS 지원 스팟 인스턴스를 중지하고 시작할 수 있습니다. 또한 용량을 더 이상 사용할 수 없거나 스팟 가격이 최고 가격을 초과하거나 스팟 인스턴스에 대한 수요가 증가하는 경우 Amazon EC2 스팟 서비스에서 개별 스팟 인스턴스를 중단 (p. 310)할 수 있습니다.	온디맨드 인스턴스가 중단(중지, 최대 절전 또는 종료)되는 시간을 결정합니다.

스팟 인스턴스 사용 전략

애플리케이션에 대해 보장된 컴퓨팅 리소스를 최소 수준으로 유지하기 위한 한 가지 전략은 온디맨드 인스턴스의 코어 그룹을 시작하고 기회가 생기면 스팟 인스턴스로 이를 보완하는 것입니다.



또 다른 전략은 지정된 지속 시간(스팟 블록이라고도 함)을 갖춘 스팟 인스턴스를 시작하는 것인데, 이것은 선택한 지속 시간 동안에는 종단되지 않으며 계속 실행됩니다. 드문 경우 Amazon EC2 용량 때문에 스팟 블록이 종단될 수 있습니다. 이 경우에는 인스턴스를 종료하기 2분 전에 경고를 보내며, 종료된 인스턴스를 사용한 경우에도 해당 인스턴스에 대해 요금이 부과되지 않습니다. 자세한 내용은 [스팟 인스턴스의 지속 시간 정의 \(p. 256\)](#) 단원을 참조하십시오.

시작하는 방법

Amazon EC2 사용에 앞서 가장 먼저 설정이 필요합니다. 스팟 인스턴스를 시작하기 전에 온디맨드 인스턴스를 시작해 보는 것도 도움이 될 수 있습니다.

실행 안내

- [Amazon EC2 설정 \(p. 12\)](#)
- [자습서: Amazon EC2 Windows 인스턴스 시작하기 \(p. 16\)](#)

스팟 기본 사항

- [스팟 인스턴스의 작동 방식 \(p. 244\)](#)
- [스팟 집합 작동 방식 \(p. 246\)](#)

스팟 인스턴스 작업

- [중단 준비 \(p. 313\)](#)
- [스팟 인스턴스 요청 생성 \(p. 258\)](#)
- [요청 상태 정보 가져오기 \(p. 308\)](#)

스팟 집합 작업

- [스팟 집합 권한 \(p. 273\)](#)
- [스팟 집합 요청 생성 \(p. 277\)](#)

관련 서비스

Amazon EC2를 사용하여 스팟 인스턴스를 직접 프로비저닝할 수 있습니다. 또한 AWS의 다른 서비스를 사용하여 스팟 인스턴스를 프로비저닝할 수 있습니다. 자세한 내용은 다음 설명서를 참조하십시오.

Amazon EC2 Auto Scaling 및 스팟 인스턴스

Amazon EC2 Auto Scaling에서 스팟 인스턴스를 시작할 수 있도록 지불하고자 하는 최고 가격으로 시작 템플릿이나 구성을 생성할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹에서 스팟 인스턴스 시작](#)과 [Amazon EC2 Auto Scaling 사용 설명서의 여러 인스턴스 유형 및 구매 옵션 사용](#)을 참조하십시오.

Amazon EMR 및 스팟 인스턴스

Amazon EMR 클러스터에서 스팟 인스턴스를 실행하는 것이 유용할 수 있는 시나리오가 있습니다. 자세한 내용은 Amazon EMR 관리 안내서의 [스팟 인스턴스 및 스팟 인스턴스를 언제 사용해야 합니까? 단원](#)을 참조하십시오.

AWS CloudFormation 템플릿

AWS CloudFormation에서는 JSON 형식의 템플릿을 사용하여 AWS 리소스 컬렉션을 생성하고 관리할 수 있습니다. AWS CloudFormation 템플릿은 지불하고자 하는 최고 가격을 포함할 수 있습니다. 자세한 내용은 [EC2 스팟 인스턴스 Updates - Auto Scaling and CloudFormation Integration](#)을 참조하십시오.

AWS SDK for Java

Java 프로그래밍 언어를 사용하여 스팟 인스턴스를 관리할 수 있습니다. 자세한 내용은 [자습서: Amazon EC2 스팟 인스턴스 및 자습서: 고급 Amazon EC2 스팟 요청 관리](#)를 참조하십시오.

.NET용 AWS SDK

.NET 프로그래밍 환경을 사용하여 스팟 인스턴스를 관리할 수 있습니다. 자세한 내용은 [자습서: Amazon EC2 스팟 인스턴스 단원](#)을 참조하십시오.

요금 및 비용 절감

스팟 인스턴스에 대해 스팟 가격을 지불합니다. 이 가격은 Amazon EC2에서 설정되며 스팟 인스턴스의 장기적 수요 및 공급에 따라 점진적으로 조정됩니다. 요청에 대한 최고 가격이 현재 스팟 가격을 초과하는 경우 Amazon EC2는 용량이 가용 상태가 될 때 요청을 이행합니다. 스팟 인스턴스는 사용자가 직접 종료를 하거나 용량이 더 이상 가용 상태가 아니거나 스팟 가격이 최고 가격을 초과하거나 Amazon EC2 Auto Scaling 그룹이 **축소** 도중 종료할 때까지 실행됩니다.

미리 정의된 지속 시간이 있는 스팟 인스턴스는 실행 중에 스팟 인스턴스에 대해 여전히 유효한 시간당 고정 가격을 사용합니다.

사용자 또는 Amazon EC2가 실행 중인 스팟 인스턴스를 중단하는 경우, 사용되는 운영 체제와 누가 스팟 인스턴스를 중단했는지에 따라 사용된 시간(초) 또는 전체 시간에 대한 요금이 부과되거나 요금이 무료일 수 있습니다. 자세한 내용은 [중단된 스팟 인스턴스에 대한 청구 \(p. 316\)](#) 단원을 참조하십시오.

가격 보기

AWS 지역 및 인스턴스 유형당 현재(5분마다 업데이트됨) 최저 스팟 가격을 보려면 [스팟 인스턴스 요금](#) 페이지를 참조하십시오.

지난 3개월 동안의 스팟 가격 기록을 보려면 Amazon EC2 콘솔 또는 `describe-spot-price-history` 명령(AWS CLI)을 사용하십시오. 자세한 내용은 [스팟 인스턴스 요금 기록 \(p. 252\)](#) 단원을 참조하십시오.

각 AWS 계정의 코드에 가용 영역을 독립적으로 매핑합니다. 따라서 서로 다른 계정 간에 동일한 가용 영역 코드(예: us-west-2a)에 대한 결과가 다를 수 있습니다.

비용 절감액 보기

단일 스팟 집합 또는 모든 스팟 인스턴스에 대해 스팟 인스턴스 사용에서 얻는 절감을 볼 수 있습니다. 지난 1시간 또는 지난 3일 동안 실현된 절감액을 볼 수 있으며, vCPU 시간당 평균 비용 및 메모리(GiB) 단위 시간당 평균 비용을 볼 수 있습니다. 절감액은 추정치이며 사용량에 대한 청구 조정이 제외되어 있기 때문에 실제 절감액과 다를 수 있습니다. 비용 절감액 보기에 대한 자세한 내용은 [스팟 인스턴스 구입으로 절감되는 비용 \(p. 253\)](#) 단원을 참조하십시오.

결제 보기

청구 요금을 검토하려면 [AWS 계정 활동 페이지](#)를 참조하십시오. 청구서에는 요금 내역을 자세하게 확인할 수 있는 사용 보고서 링크가 포함됩니다. 자세한 내용은 [AWS Account Billing](#)을 참조하십시오.

AWS 결제, 계정 및 이벤트에 관련된 질문은 [AWS Support](#)에 문의하십시오.

EC2 스팟 모범 사례

Amazon EC2 스팟 인스턴스는 AWS 클라우드의 예비 EC2 컴퓨팅 용량으로 온디맨드 요금에 비해 최대 90% 할인된 가격으로 사용할 수 있습니다. 온디맨드 인스턴스와 스팟 인스턴스 간의 유일한 차이점은 Amazon EC2에서 다시 용량이 필요할 때 Amazon EC2가 2분 전 알림을 통해 스팟 인스턴스를 중단할 수 있다는 것입니다.

스팟 인스턴스는 유연한 상태 비저장, 내결합성 애플리케이션에 권장됩니다. 예를 들어 스팟 인스턴스는 빅 데이터, 컨테이너화된 워크로드, CI/CD, 상태 비저장 웹 서버, 고성능 컴퓨팅(HPC), 렌더링 워크로드에 적합합니다.

스팟 인스턴스는 실행 중인 동안에는 온디맨드 인스턴스와 정확히 동일합니다. 그러나 스팟은 실행 중인 인스턴스를 워크로드를 완료할 수 있을 만큼 충분히 오래 유지할 수 있다고 보장하지 않습니다. 또한 스팟은 찾고 있는 인스턴스의 즉각적인 가용성을 보장하거나 요청한 총 용량을 항상 확보할 수 있다고 보장하지 않습니다. 또한 스팟 인스턴스 가용성은 수요와 공급에 따라 달라지기 때문에 스팟 인스턴스 종단 및 용량은 시간이 지남에 따라 변할 수 있으며 과거의 성능이 미래의 결과를 보장하지 않습니다.

스팟 인스턴스는 유연성이 없거나 상태 저장이거나 내결함성이 없거나 인스턴스 노드 간에 밀접하게 연결된 워크로드에 적합하지 않습니다. 또한 수시로 목표 용량을 완전히 사용할 수 없는 경우를 허용하지 않는 워크로드에는 권장되지 않습니다. 이러한 워크로드에는 스팟 인스턴스를 사용하지 말고 중단을 처리하기 위해 온디맨드 인스턴스로 장애 조치를 시도하지 않도록 강력히 경고합니다.

숙련된 스팟 사용자인지 또는 스팟 인스턴스를 처음 사용하는지 관계없이 현재 스팟 인스턴스 중단 또는 가용성에 문제가 발생하는 경우 다음 모범 사례를 따라 스팟 서비스를 사용하는 최상의 환경을 제공하는 것이 좋습니다.

스팟 모범 사례

- [개별 인스턴스에서 중단 대비 \(p. 243\)](#)
- [인스턴스 유형 및 가용 영역에 대한 유연성 유지 \(p. 243\)](#)
- [EC2 Auto Scaling 그룹 또는 스팟 집합을 사용하여 총 용량 관리 \(p. 243\)](#)
- [용량 최적화 할당 전략 사용 \(p. 244\)](#)
- [통합 AWS 서비스를 사용하여 스팟 인스턴스 관리 \(p. 244\)](#)

개별 인스턴스에서 중단 대비

스팟 인스턴스 중단을 정상적으로 처리하는 가장 좋은 방법은 내결함성이 있도록 애플리케이션을 설계하는 것입니다. 이를 위해 스팟 인스턴스 중단 공지를 활용할 수 있습니다. 스팟 인스턴스 중단 공지는 Amazon EC2에서 스팟 인스턴스를 중단하기 2분 전에 생성되는 경고입니다. [Amazon EventBridge](#)에 중단 알림을 캡처하는 규칙을 만든 다음 워크로드 진행에 대한 검사점을 트리거하거나 중단을 정상적으로 처리하는 것이 좋습니다. 이벤트 규칙을 생성하고 사용하는 방법을 안내하는 자세한 예제는 [Amazon EC2 스팟 인스턴스 중단 공지 활용](#)을 참조하십시오.

워크로드가 “시간 유연성”인 경우 중단될 때 스팟 인스턴스가 중지되거나 최대 절전 모드로 전환되도록 구성할 수도 있습니다. Amazon EC2는 중단 시 자동으로 스팟 인스턴스를 중지하거나 최대 절전 모드로 전환하며 가용 용량이 있으면 인스턴스를 자동으로 재개합니다.

자세한 내용은 [스팟 인스턴스 중단 \(p. 310\)](#) 단원을 참조하십시오.

인스턴스 유형 및 가용 영역에 대한 유연성 유지

스팟 인스턴스 풀은 인스턴스 유형(예: m5.large) 및 가용 영역(예: us-east-1a)이 동일한 미사용 EC2 인스턴스의 집합입니다. 요청하는 인스턴스 유형과 워크로드를 배포할 수 있는 가용 영역에 대한 유연성이 있어야 합니다. 그러면 스팟에서 필요한 컴퓨팅 용량을 찾고 할당할 가능성이 높아집니다. 예를 들어 c4, m5 및 m4 제품군의 large를 사용해도 무방하면 c5.large를 요청하지 마십시오.

특정 요구 사항에 따라 컴퓨팅 요구 사항을 충족하기 위해 유연하게 선택할 수 있는 인스턴스 유형을 평가할 수 있습니다. 워크로드를 수직으로 확장할 수 있는 경우 요청에 더 큰 인스턴스 유형(vCPU 및 메모리 추가)을 포함해야 합니다. 수평으로만 확장할 수 있는 경우 이전 세대 인스턴스 유형을 포함해야 합니다. 이러한 인스턴스는 온디맨드 고객의 수요가 적기 때문입니다.

일반적으로 각 워크로드에 대해 최소 10개의 인스턴스 유형을 유연하게 선택할 수 있어야 합니다. 또한 모든 가용 영역이 사용자의 VPC에서 사용하도록 구성되고 워크로드에 맞게 선택되어야 합니다.

EC2 Auto Scaling 그룹 또는 스팟 집합을 사용하여 총 용량 관리

스팟을 사용하면 개별 인스턴스가 아니라 vCPU, 메모리, 스토리지, 네트워크 처리량 등의 단위로 총 용량을 고려할 수 있습니다. Auto Scaling 그룹 및 스팟 집합을 사용하여 목표 용량을 시작 및 유지 관리할 수 있으며, 중단되거나 수동으로 종료된 모든 항목을 대체할 리소스를 자동으로 요청할 수 있습니다. Auto Scaling 그룹 또는 스팟 집합을 구성할 때는 애플리케이션 요구 사항에 따라 인스턴스 유형과 목표 용량만 지정하면 됩니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [Auto Scaling 그룹](#)과 이 사용 설명서의 [스팟 집합 요청 생성 \(p. 277\)](#) 단원을 참조하십시오.

용량 최적화 할당 전략 사용

Auto Scaling 그룹의 할당 전략은 예비 용량이 있는 스팟 인스턴스 풀을 수동으로 찾을 필요 없이 목표 용량을 프로비저닝하는데 도움이 됩니다. 이 **capacity optimized** 전략은 가장 사용 가능한 스팟 인스턴스 풀에서 인스턴스를 자동으로 프로비저닝하므로 이 전략을 사용하는 것이 좋습니다. 스팟 집합에서 **capacity optimized** 할당 전략을 활용할 수도 있습니다. 스팟 인스턴스 용량이 최적의 용량을 가진 풀에서 소싱되기 때문에 스팟 인스턴스가 회수될 가능성이 줄어듭니다. 할당 전략에 대한 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [스팟 인스턴스](#)와 이 사용 설명서의 [용량 최적화를 위한 스팟 집합 구성](#) (p. 247) 단원을 참조하십시오.

통합 AWS 서비스를 사용하여 스팟 인스턴스 관리

다른 AWS 서비스가 스팟과 통합되어 개별 인스턴스 또는 폴릿을 관리할 필요 없이 전체 컴퓨팅 비용을 절감할 수 있습니다. 적용 가능한 워크로드에 대해 Amazon EMR, Amazon ECS, AWS Batch, Amazon EKS, SageMaker, AWS Elastic Beanstalk 및 Amazon GameLift 솔루션을 고려하는 것이 좋습니다. 이러한 서비스의 스팟 모범 사례에 대한 자세한 내용은 [Amazon EC2 스팟 인스턴스 워크샵 웹 사이트](#)를 참조하십시오.

스팟 인스턴스의 작동 방식

스팟 인스턴스를 시작하려면 스팟 인스턴스 요청을 생성합니다. 그렇지 않으면 Amazon EC2에서 사용자를 대신하여 스팟 인스턴스 요청을 생성합니다. 스팟 인스턴스 요청이 이행되면 스팟 인스턴스가 시작됩니다.

다음과 같은 방법으로 스팟 인스턴스를 시작할 수 있습니다.

- 스팟 인스턴스 요청을 생성할 수 있습니다. 자세한 내용은 [스팟 인스턴스 요청 생성](#) (p. 258) 단원을 참조하세요.
- 원하는 수의 스팟 인스턴스를 지정하는 스팟 집합 요청을 생성할 수 있습니다. Amazon EC2는 스팟 집합 요청에 지정된 모든 스팟 인스턴스 요청에 대해 사용자를 대신하여 스팟 인스턴스 요청을 생성합니다. 자세한 내용은 [스팟 집합 요청 생성](#) (p. 277) 단원을 참조하세요.
- 원하는 수의 스팟 인스턴스를 지정하는 EC2 집합을 생성할 수 있습니다. Amazon EC2는 EC2 집합에 지정된 모든 스팟 인스턴스에 대해 사용자를 대신하여 스팟 인스턴스 요청을 생성합니다. 자세한 내용은 [EC2 집합 생성](#) (p. 418) 단원을 참조하세요.

스팟 인스턴스 요청에는 인스턴스별 시간당 지불하려는 최고 가격이 포함되어야 합니다. 가격을 지정하지 않으면 기본적으로 온디맨드 가격으로 설정됩니다. 요청에는 인스턴스 유형 및 가용 영역과 같은 다른 제약 조건이 포함될 수 있습니다.

지불하려는 최고 가격이 스팟 가격을 초과하고 사용 가능한 용량이 있는 경우 스팟 인스턴스가 시작됩니다. 지불하려는 최고 가격이 스팟 가격보다 낮으면 인스턴스가 시작되지 않습니다. 그러나 Amazon EC2는 스팟 인스턴스의 장기 공급 및 수요를 기반으로 스팟 가격을 점진적으로 조정하므로 지불하려는 최고 가격이 결국 스팟 가격을 초과할 수 있으며, 이 경우 인스턴스가 시작됩니다.

스팟 인스턴스는 중지하거나 종료 할 때까지 또는 Amazon EC2가 이를 중단 할 때까지 실행됩니다(스팟 인스턴스 중단이라고 함).

스팟 인스턴스를 사용할 때는 중단에 대비해야 합니다. Amazon EC2는 스팟 가격이 최고 가격을 초과하거나 스팟 인스턴스에 대한 수요가 증가하거나 스팟 인스턴스의 공급이 감소할 때 스팟 인스턴스를 중단할 수 있습니다. Amazon EC2는 스팟 인스턴스를 중단할 때 스팟 인스턴스 중단 공지를 보내서 Amazon EC2가 중지하기 2분 전에 이를 인스턴스에 경고합니다. 스팟 인스턴스에 대한 종료 방지 기능은 활성화할 수 없습니다. 자세한 내용은 [스팟 인스턴스 중단](#) (p. 310) 단원을 참조하십시오.

Amazon EBS 지원 스팟 인스턴스를 중지, 시작, 재부팅 또는 종료할 수 있습니다. 스팟 서비스는 인스턴스를 중단할 때 스팟 인스턴스를 중지, 종료 또는 최대 절전 모드로 설정할 수 있습니다.

목차

- 시작 그룹에서 스팟 인스턴스 시작 (p. 245)
- 가용 영역 그룹에서 스팟 인스턴스 시작 (p. 245)
- VPC에서 스팟 인스턴스 시작 (p. 245)

시작 그룹에서 스팟 인스턴스 시작

스팟 인스턴스 요청에서 시작 그룹을 지정하여 해당 인스턴스를 모두 시작할 수 있는 경우에만 스팟 인스턴스 집합을 시작하도록 Amazon EC2에 알립니다. 또한 스팟 서비스가 시작 그룹에 있는 인스턴스 중 하나를 종료해야 하는 경우(예를 들어, 스팟 가격이 최고 가격을 초과하는 경우) 모든 인스턴스를 종료해야 합니다. 그러나 사용자가 시작 그룹에 있는 인스턴스를 하나 이상 종료하는 경우 Amazon EC2는 시작 그룹에 있는 나머지 인스턴스를 종료하지 않습니다.

이 옵션이 유용할 수 있지만 이러한 제약 조건을 추가하면 스팟 인스턴스 요청이 이행될 가능성은 낮아지고 스팟 인스턴스가 종료될 가능성은 높아질 수 있습니다. 예를 들어, 시작 그룹에 다중 가용 영역의 인스턴스가 포함되어 있습니다. 이러한 가용 영역 중 하나에서 용량이 감소되어 더는 사용할 수 없는 상태인 경우 Amazon EC2에서는 이 시작 그룹에 대해 모든 인스턴스를 종료합니다.

이전의 성공적인 요청과 동일한(기존) 시작 그룹을 지정하는 다른 성공적인 스팟 인스턴스 요청을 생성하면 새로운 인스턴스가 시작 그룹에 추가됩니다. 이후 이 시작 그룹의 인스턴스가 종료되면 첫 번째 및 두 번째 요청에서 시작된 인스턴스를 포함하여 시작 그룹의 모든 인스턴스가 종료됩니다.

가용 영역 그룹에서 스팟 인스턴스 시작

스팟 인스턴스 요청에서 가용 영역 그룹을 지정하여 동일한 가용 영역에 있는 스팟 인스턴스 세트를 시작하도록 스팟 서비스에 알립니다. Amazon EC2가 가용 영역 그룹에 있는 모든 인스턴스를 동시에 중단할 필요는 없습니다. Amazon EC2가 가용 영역 그룹의 인스턴스를 하나 중단해야 하는 경우 다른 인스턴스는 실행 중인 상태로 유지됩니다.

이 옵션이 유용할 수 있지만 이러한 제약 조건을 추가하면 스팟 인스턴스 요청이 이행될 가능성이 낮아질 수 있습니다.

가용 영역 그룹을 지정하지만 스팟 인스턴스 요청에서 가용 영역을 지정하지 않는 경우 결과는 무엇을 지정했는지에 따라 다릅니다.

기본 VPC

Amazon EC2는 지정된 서브넷에 대한 가용 영역을 사용합니다. 서브넷을 지정하지 않으면 가용 영역 및 해당 가용 영역의 기본 서브넷이 자동으로 선택되지만 최저 요금 영역은 선택되지 않을 수 있습니다. 가용 영역에 대한 기본 서브넷을 삭제한 경우 다른 서브넷을 지정해야 합니다.

기본이 아닌 VPC

Amazon EC2는 지정된 서브넷에 대한 가용 영역을 사용합니다.

VPC에서 스팟 인스턴스 시작

스팟 인스턴스에 대해 서브넷을 지정하는 것과 동일한 방법으로 온디맨드 인스턴스에 대해 서브넷을 지정합니다.

- VPC에서는 기본 최고 가격(온디맨드 가격)을 사용하거나 스팟 인스턴스의 스팟 가격 기록을 토대로 최고 가격을 정해야 합니다.
- [기본 VPC] 낮은 가격의 특정 가용 영역에서 스팟 인스턴스가 시작되도록 하려면 스팟 인스턴스 요청에서 해당 서브넷을 지정해야 합니다. 서브넷을 지정하지 않으면 Amazon EC2에서 서브넷이 자동으로 선택되며, 이 서브넷에 대한 가용 영역에는 최저 스팟 가격이 없을 수 있습니다.
- [기본이 아닌 VPC] 스팟 인스턴스에 대해 서브넷을 지정해야 합니다.

스팟 집합 작동 방식

스팟 집합은 스팟 인스턴스 및 온디맨드 인스턴스의 모음 또는 플릿입니다.

스팟 집합은 사용자가 스팟 집합 요청에서 지정한 목표 용량을 충족하는 데 필요한 수만큼 스팟 인스턴스 및 온디맨드 인스턴스를 시작하려고 합니다. 스팟 인스턴스에 대한 요청은 요청에 지정된 최대 가격이 현재 스팟 가격을 초과하고 사용 가능한 용량이 있으면 수행됩니다. 스팟 인스턴스가 종단될 경우 스팟 집합은 대상 용량 플릿을 유지하려고 시도합니다.

또한 플릿에 대해 지불할 시간당 최대 금액을 설정할 수 있으며, 스팟 집합은 최대 금액에 도달할 때까지 인스턴스를 실행합니다. 지불하려는 최대 금액에 도달하면 플릿은 목표 용량을 충족하지 않은 경우에도 인스턴스 실행을 중지합니다.

스팟 인스턴스 풀은 동일한 인스턴스 유형(예: m5.large), 운영 체제, 가용 영역 및 네트워크 플랫폼을 가지는 미사용 EC2 인스턴스의 세트입니다. 스팟 집합 요청을 할 때 인스턴스 유형, AMI, 가용 영역 또는 서브넷에 따라 바뀌는 여러 시작 사양을 포함할 수 있습니다. 스팟 집합은 요청에 포함된 시작 사양과 스팟 집합 요청의 구성을 기반으로 스팟 집합 요청을 이행하는데 사용되는 스팟 인스턴스 풀을 선택합니다. 스팟 인스턴스는 선택한 풀에서 가져옵니다.

내용

- [스팟 집합의 온디맨드](#) (p. 246)
- [스팟 인스턴스의 할당 전략](#) (p. 246)
- [스팟 가격 재정의](#) (p. 248)
- [지출 제어](#) (p. 248)
- [스팟 집합 인스턴스 가중치 부여](#) (p. 249)
- [연습: 인스턴스 가중치를 부여한 스팟 집합 사용](#) (p. 250)

스팟 집합의 온디맨드

항상 인스턴스 용량을 사용할 수 있도록 스팟 집합 요청에 온디맨드 용량에 대한 요청을 포함할 수 있습니다. 스팟 집합 요청에 원하는 목표 용량 및 해당 용량의 몇 %가 온디맨드 용량이어야 하는지를 지정합니다. 잔고는 스팟 용량으로 구성됩니다. 스팟 용량은 가용 Amazon EC2 용량이 있고 용량이 가용 상태일 경우 시작됩니다. 예를 들어 스팟 집합 요청에 목표 용량을 10으로 지정하고 온디맨드 용량을 8로 지정하면 Amazon EC2는 8개의 용량 단위를 온디맨드로 시작하고 2개($10-8=2$)의 용량 단위를 스팟으로 시작합니다.

온디맨드 용량에 대한 인스턴스 유형 우선순위 지정

스팟 집합가 온디맨드 용량을 채우려고 시도하는 경우 기본적으로 최저 가격의 인스턴스 유형을 먼저 시작합니다. OnDemandAllocationStrategy가 prioritized로 설정된 경우 스팟 집합가, 우선 순위를 통해, 온디맨드 용량을 채우기 위해 먼저 사용할 인스턴스 유형을 결정합니다. 시작 템플릿 재정의에 우선 순위를 할당하고 우선 순위가 가장 높은 것을 먼저 시작합니다.

예를 들어 서로 다른 인스턴스 유형인 c3.large, c4.large, c5.large를 각각 지닌 3개의 시작 템플릿 재정의를 구성했다고 가정해 보겠습니다. c5.large에 대한 온디맨드 가격은 c4.large에 대한 온디맨드 가격보다 낮습니다. c3.large가 가장 낮습니다. 우선 순위를 사용해 순서를 결정하지 않는 경우 플릿이 c3.large로 시작하여 온디맨드 용량을 채운 후 c5.large를 사용합니다. 종종 c4.large에 대한 미사용 예약 인스턴스가 있게 되므로 c4.large, c3.large, c5.large의 순서이도록 시작 템플릿 재정의 우선 순위를 설정할 수 있습니다.

스팟 인스턴스의 할당 전략

스팟 집합의 스팟 인스턴스 할당 전략에 따라 시작 사양으로 표시되는 가능한 스팟 인스턴스 풀에서 스팟 집합에 대한 요청을 이행하는 방법이 결정됩니다. 다음은 스팟 집합 요청에서 지정할 수 있는 할당 전략입니다.

lowestPrice

스팟 인스턴스는 최저 가격의 풀에서 가져옵니다. 이는 기본 전략입니다.

diversified

스팟 인스턴스는 모든 풀에 두루 분산됩니다.

capacityOptimized

스팟 인스턴스는 시작하는 인스턴스의 수에 대한 용량이 최적화된 풀에서 가져옵니다.

InstancePoolsToUseCount

스팟 인스턴스는 지정한 스팟 풀 수에 걸쳐 배포됩니다. 이 파라미터는 `lowestPrice`와 함께 사용하는 경우에만 유효합니다.

목표 용량 유지

스팟 가격 또는 스팟 인스턴스 풀의 가용 용량 변화로 인해 스팟 인스턴스가 종료된 후에는 `maintain` 유형의 스팟 집합이 대체 스팟 인스턴스를 시작합니다. 할당 전략이 `lowestPrice`인 경우, 플릿은 현재 스팟 가격이 가장 낮은 풀에서 대체 인스턴스를 시작합니다. 할당 전략이 `diversified`인 경우 플릿은 나머지 풀에 대체 스팟 인스턴스를 배포합니다. 할당 전략으로써 `InstancePoolsToUseCount`와 함께 `lowestPrice`를 사용하는 경우 플릿이 최저 가격의 스팟 풀을 선택하여 지정한 스팟 풀 수에 걸쳐 스팟 인스턴스를 시작합니다.

비용 최적화를 위한 스팟 집합 구성

스팟 인스턴스 사용 비용을 최적화하려면 스팟 집합이 현재 스팟 가격을 기반으로 인스턴스 유형과 가용 영역의 가장 저렴한 조합을 자동으로 배포하도록 `lowestPrice` 할당 전략을 지정합니다.

스팟 집합은 항상 퍼블릭 온디맨드 가격을 기반으로 최저 비용의 인스턴스 유형을 온디맨드 인스턴스 목표 용량에 대해 선택하며, 스팟 인스턴스에 대해서는 계속해서 할당 전략(`lowestPrice`, `capacityOptimized` 또는 `diversified`)을 따릅니다.

비용 최적화 및 다각화를 위한 스팟 집합 구성

저렴함과 다각화를 모두 충족하는 스팟 인스턴스 플릿을 생성하려면 `InstancePoolsToUseCount`과 함께 `lowestPrice` 할당 전략을 사용하십시오. 스팟 집합이 지정된 스팟 풀 수에 걸쳐 현재 스팟 가격을 기반으로 인스턴스 유형과 가용 영역의 가장 저렴한 조합을 자동으로 배포합니다. 이 조합을 통해 가장 비싼 스팟 인스턴스를 피할 수 있습니다.

용량 최적화를 위한 스팟 집합 구성

스팟 인스턴스에서 요금은 시간이 지나면서 수요 및 공급의 장기 추세에 따라 서서히 변화하지만 용량은 실시간으로 변동합니다. `capacityOptimized` 전략은 실시간 용량 데이터를 기준으로 가장 가용성이 높은 풀을 예측하여 자동으로 스팟 인스턴스를 가장 가용성이 높은 풀로 시작합니다. 이 전략은 빅 데이터 및 분석, 이미지 및 미디어 렌더링, 기계 학습, 고성능 컴퓨팅과 같이 작업 재시작 및 체크포인트와 관련된 중단으로 인한 비용이 더 높을 수 있는 워크로드에 유용합니다. `capacityOptimized` 전략은 중단을 줄일 수 있는 가능성을 제공함으로써 전체 워크로드 비용을 낮출 수 있습니다.

적합한 할당 전략 선택

사용 사례를 바탕으로 스팟 집합을 최적화할 수 있습니다.

플릿이 작거나 짧은 시간 동안 실행될 경우 모든 인스턴스가 단일 스팟 인스턴스 풀에 있더라도 스팟 인스턴스가 중단될 확률은 낮습니다. 따라서 `lowestPrice` 전략이 요구를 충족시키는 동시에 최저 가격을 제공할 가능성이 높습니다.

플릿이 크거나 장시간 실행될 경우 스팟 인스턴스를 여러 풀로 분산하여 플릿의 가용성을 높일 수 있습니다. 예를 들어 스팟 집합 요청이 풀 10개와 인스턴스 100개의 목표 용량을 지정하면 플릿이 각 풀에서 스팟 인스턴스 10개를 시작합니다. 풀에서 스팟 가격이 최고 가격을 초과하는 경우, 플릿 중 10%만 영향을 받습니다. 이 전략을 사용하면 플릿이 시간이 지나면서 어느 한 풀에서 발생하는 스팟 가격의 상승에 덜 민감해집니다.

diversified 전략 사용 시 스팟 집합은 [온디맨드 가격](#)보다 높거나 이 가격과 동일한 스팟 가격의 풀로 스팟 인스턴스를 시작하지 않습니다.

저렴하고 다각화된 플릿을 생성하려면 `lowestPrice`과 함께 `InstancePoolsToUseCount` 전략을 사용하십시오. 스팟 인스턴스를 할당할 스팟 풀 수로써 낮은 수 또는 높은 수를 사용할 수 있습니다. 예를 들어 일괄 처리를 실행하는 경우 대기열이 비용 절감 효과를 극대화하는 동시에 컴퓨팅 파워를 항상 확보할 수 있도록 스팟 풀 수를 낮게(예: `InstancePoolsToUseCount=2`) 지정하는 것이 좋습니다. 웹 서비스를 실행하는 경우 스팟 인스턴스 풀을 일시적으로 사용할 수 없게 되었을 때 그 충격을 최소화할 수 있도록 스팟 풀 수를 낮게(예: `InstancePoolsToUseCount=10`) 지정하는 것이 좋습니다.

플릿이 작업 재시작 및 검사와 연관된 중단 비용이 높을 수 있는 워크로드를 실행하는 경우 `capacityOptimized` 전략을 사용하십시오. 이 전략은 중단을 줄일 수 있는 가능성을 제공함으로써 전체 워크로드 비용을 낮출 수 있습니다.

스팟 가격 재정의

각 스팟 집합 요청은 글로벌 최고 가격을 포함하거나 기본 가격(온디맨드 가격)을 사용할 수 있습니다. 스팟 집합은 각각의 시작 사양에 대해 기본 최고 가격으로 이 가격을 사용합니다.

하나 이상의 시작 사양에서 최고 가격을 선택적으로 지정할 수 있습니다. 이 가격은 시작 사양에 특정한 것입니다. 시작 사양에 특정 가격이 포함되는 경우 스팟 집합은 글로벌 최고 가격 대신 이 최고 가격을 사용합니다. 특정 최고 가격을 포함하지 않는 다른 시작 사양은 글로벌 최고 가격을 계속해서 사용합니다.

지출 제어

스팟 집합은 대상 용량 또는 지불할 최대 금액에 도달하면 인스턴스 실행을 중지합니다. 플릿에 대해 시간 당 지불하는 금액을 관리하라면 스팟 인스턴스의 경우 `SpotMaxTotalPrice`, 온디맨드 인스턴스의 경우 `OnDemandMaxTotalPrice`를 지정할 수 있습니다. 최대 총 가격에 도달하면 스팟 집합은 목표 용량을 충족하지 않은 경우에도 인스턴스 실행을 중지합니다.

다음 예제와 같이 이 작업을 두 가지 시나리오로 수행할 수 있습니다. 첫 번째 시나리오에서 스팟 집합은 대상 용량을 충족했을 때 인스턴스 실행을 중지합니다. 두 번째 시나리오에서 스팟 집합은 지불할 최대 금액에 도달하면 인스턴스 실행을 중지합니다.

예: 대상 용량에 도달할 때 인스턴스 실행 중지

다음과 같은 `m4.large` 온디맨드 인스턴스 요청 시:

- 온디맨드 가격: 시간당 0.10 USD
- `OnDemandTargetCapacity`: 10
- `OnDemandMaxTotalPrice`: 1.50 USD

스팟 집합은 최대 1.00 USD(10개 인스턴스 x 0.10 USD)가 `OnDemandMaxTotalPrice` 1.50 USD를 초과하지 않기 때문에 10개의 온디맨드 인스턴스를 시작합니다.

예: 최대 총 가격에 도달할 때 인스턴스 실행 중지

다음과 같은 `m4.large` 온디맨드 인스턴스 요청 시:

- 온디맨드 가격: 시간당 0.10 USD
- `OnDemandTargetCapacity`: 10
- `OnDemandMaxTotalPrice`: 0.80 USD

스팟 집합이 온디맨드 대상 용량(온디맨드 인스턴스 10개)을 시작하면 시간당 총 비용은 1.00 USD입니다. `OnDemandMaxTotalPrice`에 대해 지정된 금액(0.80 USD) 보다 높습니다. 지불할 금액보다 더 많은 지출을 방지하기 위해 스팟 집합은 8개의 온디맨드 인스턴스(온디맨드 대상 용량 미만)만 실행합니다. 더 많이 실행하면 `OnDemandMaxTotalPrice`를 초과할 수 있기 때문입니다.

스팟 집합 인스턴스 가중치 부여

스팟 인스턴스의 플릿을 요청할 때 각 인스턴스 유형이 애플리케이션의 성능에 기여하는 용량 단위를 정의하고 인스턴스 가중치를 사용하여 적절히 각 스팟 인스턴스 풀에 대한 최고 가격을 조정할 수 있습니다.

기본적으로, 사용자가 지정하는 가격은 인스턴스 시간당 가격입니다. 인스턴스 가중치 기능을 사용할 때, 사용자가 지정하는 가격은 단위 시간당 가격입니다. 단위 시간당 가격은 인스턴스 유형에 따른 가격을 인스턴스가 나타내는 단위 수로 나누어 계산합니다. 스팟 집합은 목표 용량을 인스턴스 가중치로 나누어 시작할 스팟 인스턴스 수를 계산합니다. 결과가 정수가 아닌 경우 스팟 집합은 결과를 다음 정수로 올림하므로 풀의 크기가 목표 용량을 밀들지는 않습니다. 시작된 인스턴스의 용량이 요청된 목표 용량을 초과하더라도, 스팟 집합은 시작 사양에서 지정한 어떤 풀이든 선택할 수 있습니다.

다음 표에는 목표 용량이 10인 스팟 집합 요청에서 단위당 가격을 결정하기 위한 계산 예제를 제공합니다.

인스턴스 유형	인스턴스 가중치	인스턴스 시간당 가격	단위 시간당 가격	시작된 인스턴스의 수
r3.xlarge	2	0.05 USD	.025 (0.05를 2로 나눈 값)	5 (10을 2로 나눈 값)

인스턴스 유형	인스턴스 가중치	인스턴스 시간당 가격	단위 시간당 가격	시작된 인스턴스의 수
r3.8xlarge	8	0.10 USD	.0125 (0.10를 8로 나눈 값)	2 (10을 8로 나눈 후 올림한 결과)

스팟 집합 인스턴스 가중치를 사용해 다음과 같이 원하는 목표 용량을 이행 시점에 단위당 최저 가격으로 풀에서 프로비저닝합니다.

- 스팟 집합에 대한 목표 용량을 인스턴스(기본값) 또는 선택한 단위(예: 가상 CPU 수, 메모리, 스토리지 또는 처리량)로 설정합니다.
- 단위당 가격을 설정합니다.
- 각 시작 구성을 위해, 목표 용량으로 접근하는 방향으로 인스턴스 유형이 나타내는 단위 수를 의미하는 가중치를 지정합니다.

인스턴스 가중치 부여의 예

다음과 같은 구성의 스팟 집합 요청을 고려하십시오.

- 목표 용량은 24
- 인스턴스 유형이 r3.2xlarge이고 가중치가 6인 시작 사양
- 인스턴스 유형이 c3.xlarge이고 가중치가 5인 시작 사양

가중치는 목표 용량에 대하여 인스턴스 유형이 나타내는 단위 수를 의미합니다. 첫 번째 시작 사양에서 단위당 최저 가격(인스턴스 시간당 r3.2xlarge에 대한 가격을 6으로 나눈 값)을 제공하는 경우, 스팟 플릿은 이들 인스턴스 중 4개(24를 6으로 나눈 값)를 시작합니다.

두 번째 시작 사양에서 단위당 최저 가격(인스턴스 시간당 c3.xlarge에 대한 가격을 5로 나눈 값)을 제공하는 경우 스팟 집합은 이들 인스턴스 중 5개(24를 5로 나눈 결과를 올림한 값)를 시작합니다.

인스턴스 가중치 부여 및 할당 전략

다음과 같은 구성의 스팟 집합 요청을 고려하십시오.

- 목표 용량은 30
- 인스턴스 유형이 `c3.2xlarge`이고 가중치가 8인 시작 사양
- 인스턴스 유형이 `m3.xlarge`이고 가중치가 8인 시작 사양
- 인스턴스 유형이 `r3.xlarge`이고 가중치가 8인 시작 사양

스팟 집합이 4개의 인스턴스(30을 8로 나눈 결과를 올림한 값)를 시작합니다. `lowestPrice` 전략 사용 시, 4개의 인스턴스는 전부 단위당 최저 가격을 제공하는 풀에서 가져옵니다. `diversified` 전략 사용 시 스팟 집합은 3개의 풀 각각에서 1개의 인스턴스를 시작하고 어떤 풀에 있는 것이든 4번째 인스턴스가 단위당 최저 가격을 제공합니다.

연습: 인스턴스 가중치를 부여한 스팟 집합 사용

이 연습에서는 Example Corp이라는 가상의 회사를 통해 인스턴스 가중치를 사용하여 스팟 집합을 요청하는 프로세스를 설명합니다.

목표

제약 회사인 Example Corp은 암 퇴치 효과가 있는 화합물을 검출하는 데 Amazon EC2의 컴퓨팅 파워를 사용하려고 합니다.

계획

Example Corp은 먼저 [스팟 모범 사례](#)를 살펴봅니다. 그런 다음 스팟 집합에 대한 다음 요건을 결정합니다.

인스턴스 유형

Example Corp은 최소 60GB 메모리와 8개의 가상 CPU(vCPU)로 최적의 성능을 자랑하는 컴퓨팅 및 메모리 집약적 애플리케이션을 사용하고 있습니다. 하지만 최저 가격으로 이러한 애플리케이션 리소스를 극대화하는 것이 목표입니다. 그 결과 다음 EC2 인스턴스 유형 중 하나가 이러한 요건에 적합할 것이라는 결정을 내립니다.

인스턴스 유형	메모리(GiB)	vCPUs
<code>r3.2xlarge</code>	61	8
<code>r3.4xlarge</code>	122	16
<code>r3.8xlarge</code>	244	32

목표 용량 단위

인스턴스 가중치를 부여했을 때 목표 용량은 인스턴스 수(기본값) 또는 코어(vCPU), 메모리(GiB) 및 스토리지(GB)와 같은 요소의 조합과 동일할 수 있습니다. 그래서 Example Corp는 단위 1개당 애플리케이션의 기본 용량(60GB 메모리, vCPU 8개)을 고려하여 기본 용량의 20배면 요구에 부응할 것이라고 결정을 내립니다. 그래서 스팟 집합 요청의 목표 용량을 20으로 설정합니다.

인스턴스 가중치

목표 용량이 결정되자 이제는 인스턴스 가중치를 계산합니다. 각 인스턴스 유형에 대한 인스턴스 가중치를 계산하기 위해, 다음과 같이 목표 용량에 이르기 위해 필요한 각 인스턴스 유형의 단위를 결정합니다.

- `r3.2xlarge`(61.0GB, 8 vCPU) = 단위 20개 중 1개

- r3.4xlarge(122.0GB, 16 vCPU) = 단위 20개 중 2개
- r3.8xlarge(122.0GB, 32 vCPU) = 단위 20개 중 4개

따라서 Example Corp은 스팟 집합 요청 시 1, 2 및 4의 인스턴스 가중치를 각 시작 구성에 할당합니다.

단위 시간당 가격

Example Corp은 인스턴스 시간당 [온디맨드 가격](#)을 시작 가격으로 사용합니다. 그 밖에 최근 스팟 가격을 사용하거나, 둘을 조합할 수도 있습니다. 단위 시간당 가격을 계산하려면 인스턴스 시간당 시작 가격을 가중치로 나눕니다. 다음 예를 참조하십시오.

인스턴스 유형	온디맨드 가격	인스턴스 가중치	단위 시간당 가격
r3.2xLarge	\$0.7	1	\$0.7
r3.4xLarge	\$1.4	2	\$0.7
r3.8xLarge	\$2.8	4	\$0.7

Example Corp은 단위 시간당 글로벌 가격으로 0.7 USD를 사용하기 때문에 세 가지 인스턴스 유형 모두에서 경쟁력이 있습니다. 또한 r3.8xlarge 시작 사양에서 단위 시간당 글로벌 가격으로 0.7 USD를, 단위 시간당 특정 가격으로 0.9 USD를 사용할 수도 있습니다.

권한 검증

Example Corp은 스팟 집합 요청을 생성하기 전에 필요한 권한을 가진 IAM 역할이 있는지 검증합니다. 자세한 내용은 [스팟 집합 권한 \(p. 273\)](#) 단원을 참조하십시오.

요청 생성

Example Corp은 스팟 집합 요청에 대해 다음 구성으로 config.json 파일을 생성합니다.

```
{  
    "SpotPrice": "0.70",  
    "TargetCapacity": 20,  
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",  
    "LaunchSpecifications": [  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "r3.2xlarge",  
            "SubnetId": "subnet-482e4972",  
            "WeightedCapacity": 1  
        },  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "r3.4xlarge",  
            "SubnetId": "subnet-482e4972",  
            "WeightedCapacity": 2  
        },  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "r3.8xlarge",  
            "SubnetId": "subnet-482e4972",  
            "SpotPrice": "0.90",  
            "WeightedCapacity": 4  
        }  
    ]  
}
```

Example Corp은 다음 [request-spot-fleet](#) 명령을 사용하여 스팟 집합 요청을 생성합니다.

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

자세한 내용은 [스팟 집합요청 건 \(p. 271\)](#) 단원을 참조하십시오.

이행

할당 전략에서는 스팟 인스턴스가 어느 스팟 인스턴스 풀에서 온 것인지 확인합니다.

`lowestPrice` 전략(기본 전략) 사용 시, 스팟 인스턴스는 이행 시점에 단위당 최저 가격의 풀에서 온 것입니다. 20단위의 용량을 제공하기 위해 스팟 집합이 `r3.2xlarge` 인스턴스 20개(20을 1로 나눈 값), `r3.4xlarge` 인스턴스 10개(20을 2로 나눈 값) 또는 `r3.8xlarge` 인스턴스 5개(20을 4로 나눈 값)를 시작합니다.

Example Corp에서 `diversified` 전략을 사용한 경우에는 스팟 인스턴스가 3개의 풀 전부에서 옵니다. 스팟 집합은 총 20개의 단위에 대해 `r3.2xlarge` 인스턴스 6개(6개 단위 제공), `r3.4xlarge` 인스턴스 3개(6개 단위 제공), `r3.8xlarge` 인스턴스 2개(8개 단위 제공)를 시작합니다.

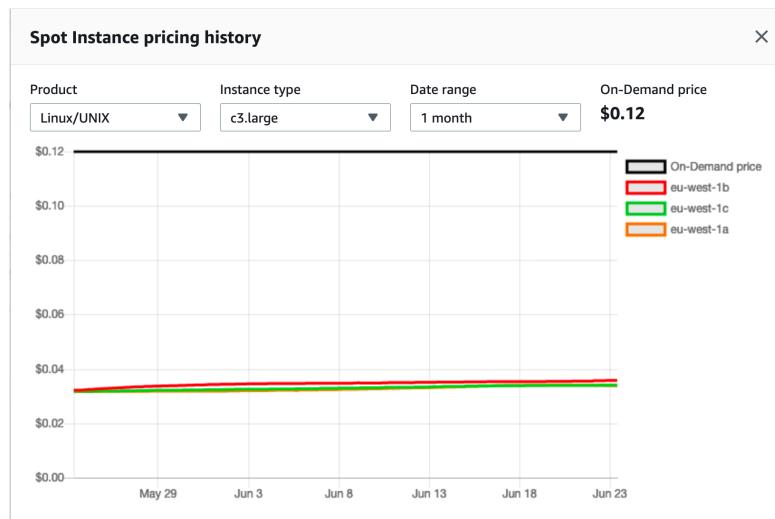
스팟 인스턴스 요금 기록

스팟 인스턴스를 요청할 때는 기본 최고 가격(온디맨드 가격)을 사용하는 것이 좋습니다. 최고 가격을 지정하고 싶은 경우에는 설정에 앞서 스팟 가격 기록을 검토하는 것이 좋습니다. 인스턴스 유형, 운영 체제 및 가용 영역을 기준으로 필터링하여 지난 90일 동안의 스팟 가격 기록을 볼 수 있습니다.

스팟 인스턴스 가격은 Amazon EC2에서 정하고, 스팟 인스턴스 용량의 장기적인 수요 공급 추세에 따라 점진적으로 조정됩니다. 현재 스팟 인스턴스 가격은 [Amazon EC2 스팟 인스턴스 가격](#)을 참조하십시오.

스팟 가격 기록을 보려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다.
3. 스팟 인스턴스를 처음 사용하는 경우 시작 페이지가 표시됩니다. 시작하기를 선택하고 화면 아래로 스크롤한 후 취소를 선택합니다.
4. 요금 내역을 선택합니다.
5. 가격 내역을 조회할 운영 체제(제품), 인스턴스 유형, 및 날짜 범위를 선택합니다. 마우스를 그래프 위에 놓으면 선택한 날짜 범위에서 특정 시간의 가격이 표시됩니다.



6. (선택 사항) 특정 가용 영역에 대한 스팟 요금 내역을 검토하려는 경우 그레프에서 가용 영역을 제거하여 가용 영역을 필터링할 수 있습니다. 그레프에서 가용 영역을 제거하면 제거할 가용 영역을 선택합니다. 다른 제품, 인스턴스 유형 또는 날짜 범위도 선택할 수 있습니다.

명령줄을 사용하여 스팟 가격 기록을 보려면

다음 명령 중 하나를 사용할 수 있습니다. 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-spot-price-history](#)(AWS CLI)
- [Get-EC2SpotPriceHistory](#)(Windows PowerShell용 AWS 도구)

스팟 인스턴스 구입으로 절감되는 비용

플랫폼 수준에서 스팟 인스턴스에 대해 또는 실행 중인 모든 스팟 인스턴스에 대해 사용량 및 절감액 정보를 확인할 수 있습니다. 플랫폼 수준에서 확인할 수 있는 사용량 및 절감액 정보에는 해당 플랫폼에서 시작 및 종료한 모든 인스턴스가 포함됩니다. 지난 1시간 또는 지난 3일에 대해 이러한 정보를 확인할 수 있습니다.

스팟 요청 페이지의 다음 스크린샷에는 스팟 집합에 대한 스팟 사용 및 비용 절감 정보가 표시됩니다.

Spot usage and savings					
4	266	700	\$9.55	\$2.99	69% Savings
Spot Instances	vCPU-hours	Mem(GiB)-hours	On-Demand total	Spot total	
				\$0.0112	\$0.0043
				Average cost per vCPU-hour	Average cost per mem(GiB)-hour
Details					
t3.medium (1)	2 vCPU hours	4 mem(GiB)-hours	\$0.01 total	70% savings	
m4.large (1)	144 vCPU hours	576 mem(GiB)-hours	\$2.52 total	68% savings	
t2.micro (2)	120 vCPU hours	120 mem(GiB)-hours	\$0.46 total	70% savings	

다음 사용 및 비용 절감 정보를 볼 수 있습니다.

- 스팟 인스턴스 – 스팟 집합에서 시작 및 종료한 스팟 인스턴스 수. 비용 절감 요약을 볼 때 이 숫자는 실행 중인 모든 스팟 인스턴스를 나타냅니다.
- vCPU-hours(vCPU-시간) – 선택한 기간 중 모든 스팟 인스턴스에서 사용된 vCPU 시간 수
- Mem(GiB)-hours(메모리(GiB)-시간) – 선택한 기간 중 모든 스팟 인스턴스에서 사용된 GiB 시간 수.
- On-Demand total(온디맨드 합계) – 인스턴스를 온디맨드 인스턴스로 시작한 경우 선택한 기간 중 결제한 총 금액
- Spot total(스팟 합계) – 선택한 기간 중 결제한 총 금액.
- Savings(절감) – 온디맨드 가격을 결제하지 않아 절감한 비율.
- Average cost per vCPU-hour(vCPU-시간당 평균 비용) – 선택한 기간 중 모든 스팟 인스턴스에서 vCPU 사용의 시간당 평균 비용으로, 다음과 같이 계산합니다. vCPU-시간당 평균 비용 = 스팟 합계 / vCPU-시간.
- Average cost per mem(GiB)-hour(메모리(GiB)-시간당 평균 비용) – 선택한 기간 중 모든 스팟 인스턴스에서 GiB 사용의 시간당 평균 비용으로, 다음과 같이 계산합니다. 메모리(GiB)-시간당 평균 비용 = 스팟 합계 / 메모리(GiB)-시간.
- 세부 정보 표 – 스팟 집합을 구성하는 여러 인스턴스 유형. 인스턴스 유형당 인스턴스 수는 괄호 안에 표시되어 있습니다. 비용 절감 요약을 볼 때 이러한 숫자는 실행 중인 모든 스팟 인스턴스를 나타냅니다.

절감 정보는 Amazon EC2 콘솔에서만 확인할 수 있습니다.

스팟 집합에 대한 비용 절감 정보를 보려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다.
3. 스팟 집합 요청을 선택하고 절감을 선택합니다.
4. 기본적으로 이 페이지에는 지난 3일에 대한 사용 및 절감 정보가 표시됩니다. last hour(지난 시간) 또는 last three days(지난 3일)를 선택합니다. 시작한지 한 시간이 되지 않는 스팟 집합의 경우 페이지에는 한 시간에 대한 절감 추정치가 표시됩니다.

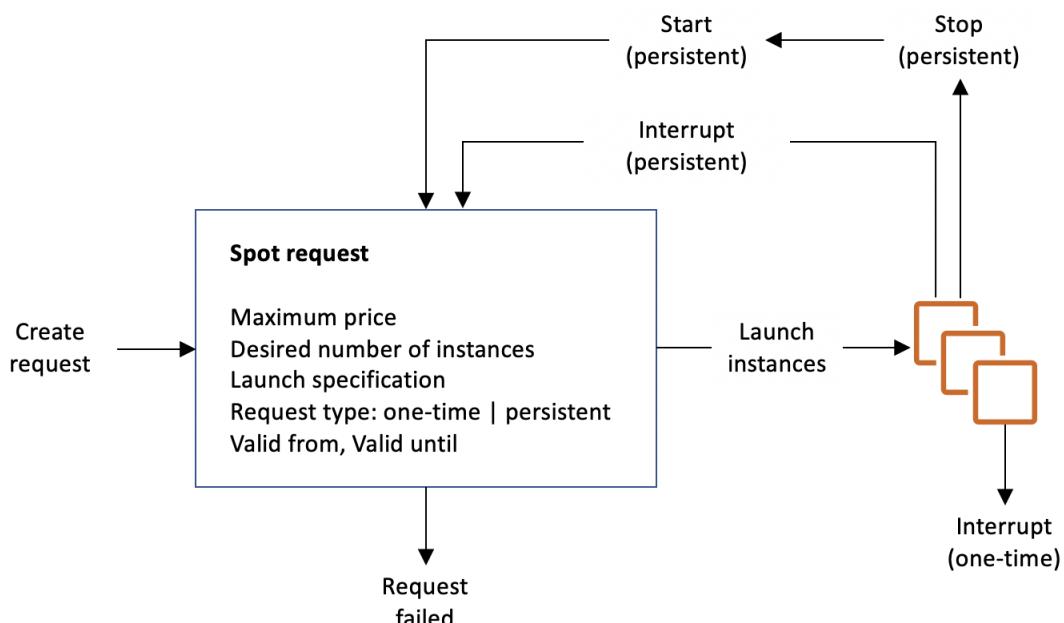
실행 중인 모든 스팟 인스턴스에 대한 비용 절감 정보를 보려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다.
3. Savings Summary(비용 절감 요약)를 선택합니다.

스팟 인스턴스요청 건

스팟 인스턴스를 사용하려면 원하는 인스턴스 수, 인스턴스 유형, 가용 영역 및 인스턴스 시간당 지불하고자 하는 최고 가격이 포함된 스팟 인스턴스 요청을 생성합니다. 최고 가격이 현재 스팟 가격을 초과하는 경우 Amazon EC2는 용량이 가용 상태가 되는 즉시 요청을 이행합니다. 그렇지 않으면 요청이 이행될 수 있을 때 까지 또는 사용자가 요청을 취소할 때까지 Amazon EC2가 대기합니다.

다음 그림에서는 스팟 요청이 작동하는 방식을 보여 줍니다. 요청 유형(일회성 또는 영구)에 따라 Amazon EC2가 스팟 인스턴스를 중단할 때 또는 사용자가 스팟 인스턴스를 중지하는 경우 요청이 다시 열리는지 여부가 결정됩니다. 요청이 영구적인 경우 스팟 인스턴스가 중단된 후 요청이 다시 열립니다. 요청이 영구적이고 사용자가 스팟 인스턴스를 중지하는 경우 스팟 인스턴스를 시작한 후에만 요청이 열립니다.



목차

- [스팟 인스턴스 요청 상태 \(p. 255\)](#)

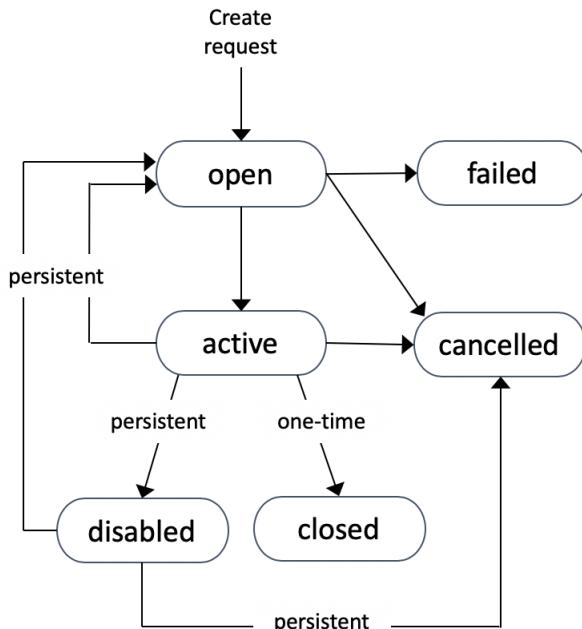
- 스팟 인스턴스의 지속 시간 정의 (p. 256)
- 스팟 인스턴스의 태넌시 지정 (p. 256)
- 스팟 인스턴스 요청에 대한 서비스 연결 역할 (p. 257)
- 스팟 인스턴스 요청 생성 (p. 258)
- 실행 중인 스팟 인스턴스 찾기 (p. 261)
- 스팟 인스턴스 요청 태그 지정 (p. 262)
- 스팟 인스턴스 요청 취소 (p. 267)
- 스팟 인스턴스 종지 (p. 267)
- 스팟 인스턴스 시작 (p. 268)
- 스팟 인스턴스를 종료하는 중 (p. 269)
- 스팟 인스턴스 요청 예시 시작 사양 (p. 269)

스팟 인스턴스 요청 상태

스팟 인스턴스 요청은 다음 상태 중 하나일 수 있습니다.

- open – 요청이 이행될 때까지 대기 중입니다.
- active – 요청이 이행되며 연결된 스팟 인스턴스가 있습니다.
- failed – 요청에 하나 이상의 잘못된 파라미터가 있습니다.
- closed – 스팟 인스턴스가 종단되거나 종료되었습니다.
- disabled – 사용자가 스팟 인스턴스를 종지했습니다.
- cancelled – 사용자가 요청을 취소했거나 요청이 만료되었습니다.

다음 그림은 요청 상태 간의 전환을 나타냅니다. 전환은 요청 유형(일회 또는 영구)에 따라 다릅니다.



일회성 스팟 인스턴스 요청은 Amazon EC2가 스팟 인스턴스를 시작하거나, 요청이 만료되거나, 사용자가 요청을 취소할 때까지 활성 상태로 유지됩니다. 스팟 가격이 최고 가격을 초과하거나 용량이 가용 상태가 아니면 스팟 인스턴스가 종료되고 스팟 인스턴스 요청이 종결됩니다.

영구적 스팟 인스턴스 요청은 요청이 이행되더라도 요청이 만료되거나 사용자가 요청을 취소할 때까지 활성 상태로 유지됩니다. 스팟 가격이 최고 가격을 초과하거나 용량이 가용 상태가 아니면 스팟 인스턴스가 중단됩니다. 인스턴스가 중단된 후에 최고 가격이 스팟 가격을 초과하거나 용량이 다시 가용 상태가 되면 스팟 인스턴스가 중지된 경우에는 시작되거나, 최대 절전 모드인 경우에는 다시 시작됩니다. 용량을 사용할 수 있고 최고 가격이 현재 스팟 가격을 초과하는 경우 스팟 인스턴스를 중지하고 다시 시작할 수 있습니다. 스팟 인스턴스가 종료되는 경우(스팟 인스턴스가 중지 상태 또는 실행 중 상태인지 여부와 상관없이) 스팟 인스턴스 요청이 다시 열리고 Amazon EC2가 새 스팟 인스턴스를 시작합니다. 자세한 내용은 [스팟 인스턴스 중지 \(p. 267\)](#), [스팟 인스턴스 시작 \(p. 268\)](#), [스팟 인스턴스를 종료하는 중 \(p. 269\)](#) 단원을 참조하십시오.

이 상태를 통해 스팟 인스턴스 요청의 상태뿐 아니라 시작된 스팟 인스턴스의 상태도 추적할 수 있습니다. 자세한 내용은 [스팟 요청 상태 \(p. 304\)](#) 단원을 참조하세요.

스팟 인스턴스의 지속 시간 정의

지속 시간(스팟 블록이라고도 함)이 정의된 스팟 인스턴스는 선택한 지속 시간 동안 중단되지 않고 계속 실행되도록 설계되었습니다. 이로 인해 배치성 프로세스, 인코딩 및 렌더링, 모델링 및 분석, 지속적 통합 작업처럼 완료하는 데 한정된 시간이 소요되는 작업에 이상적입니다.

1, 2, 3, 4, 5 또는 6시간의 지속 시간을 사용할 수 있습니다. 지불하는 요금은 지정된 지속 시간에 따라 변합니다. 1시간 또는 6시간의 지속 시간에 대한 현행 요금을 보려면 [스팟 인스턴스 요금](#)을 참조하십시오. 이 요금표를 이용해 2, 3, 4 및 5시간의 지속 시간에 대한 비용을 주산할 수 있습니다. 지속 시간으로 요청이 이행되면 스팟 인스턴스에 대한 가격이 고정되고 이 가격은 인스턴스가 종료될 때까지 유효합니다. 인스턴스를 실행하는 시간별로, 혹은 부분 시간에 대해 이 가격으로 요금이 청구됩니다. 부분 인스턴스 시간은 전체 시간으로 계산됩니다.

스팟 요청에서 지속 시간을 정의하면 각 스팟 인스턴스에 대한 지속 시간은 인스턴스 ID를 받자마자 시작됩니다. 스팟 인스턴스는 사용자가 종료할 때까지 또는 지속 시간이 끝날 때까지 실행됩니다. 지속 시간이 끝나는 시점에 Amazon EC2는 스팟 인스턴스에 종료를 표시하고 스팟 인스턴스 중단 공지를 통해 종지 2분 전에 이를 인스턴스에 경고합니다. 드문 경우 Amazon EC2 용량 때문에 스팟 블록이 중단될 수 있습니다. 이 경우에는 인스턴스를 종료하기 2분 전에 경고를 보내며, 종료된 인스턴스를 사용한 경우에도 해당 인스턴스에 대해 요금이 부과되지 않습니다.

정의된 지속 시간으로 스팟 인스턴스를 시작하려면(콘솔)

[스팟 집합 요청 생성 \(p. 277\)](#)의 절차를 따르십시오. 정의된 기간으로 스팟 인스턴스를 시작하려면 필요한 애플리케이션 또는 작업 알려주기에서 정의된 기간 워크로드를 선택합니다.

정의된 지속 시간으로 스팟 인스턴스를 시작하려면(AWS CLI)

스팟 인스턴스에 대해 지속 시간을 지정하려면 `request-spot-instances` 명령으로 `--block-duration-minutes` 옵션을 포함시키십시오. 예를 들어 다음 명령을 통해 2시간 동안 실행되는 스팟 인스턴스를 시작하는 스팟 요청이 생성됩니다.

```
aws ec2 request-spot-instances \
--instance-count 5 \
--block-duration-minutes 120 \
--type "one-time" \
--launch-specification file://specification.json
```

지정된 지속 시간으로 스팟 인스턴스 비용을 검색하려면(AWS CLI)

`describe-spot-instance-requests` 명령을 사용하여 지정된 지속 시간으로 스팟 인스턴스의 고정 비용을 검색합니다. 해당 정보는 `actualBlockHourlyPrice` 필드에 있습니다.

스팟 인스턴스의 테넌시 지정

스팟 인스턴스를 단일 테넌트 하드웨어에서 실행할 수 있습니다. 전용 스팟 인스턴스는 다른 AWS 계정에 속하는 인스턴스로부터 물리적으로 격리됩니다. 자세한 내용은 [전용 인스턴스 \(p. 349\)](#) 및 [Amazon EC2 전용 인스턴스](#) 제품 페이지를 참조하십시오.

전용 스팟 인스턴스를 실행하려면 다음 중 하나를 수행하십시오.

- 스팟 인스턴스 요청을 생성할 경우 [dedicated의 테넌시를 지정합니다](#). 자세한 내용은 [스팟 인스턴스 요청 생성 \(p. 258\)](#) 단원을 참조하십시오.
- [dedicated의 인스턴스 테넌시로 VPC에 스팟 인스턴스를 요청합니다](#). 자세한 내용은 [전용 인스턴스 테넌시의 VPC 생성하기 \(p. 352\)](#) 단원을 참조하십시오. VPC에서 [default의 인스턴스 테넌시로 스팟 인스턴스를 요청한 경우 dedicated의 테넌시로 요청할 수 없습니다](#).

다음 인스턴스 유형에서는 전용 스팟 인스턴스를 지원합니다.

현재 세대

- c4.8xlarge
- d2.8xlarge
- i3.16xlarge
- m4.10xlarge
- m4.16xlarge
- p2.16xlarge
- r4.16xlarge
- x1.32xlarge

이전 세대

- c3.8xlarge
- cc2.8xlarge
- cr1.8xlarge
- g2.8xlarge
- i2.8xlarge
- r3.8xlarge

스팟 인스턴스 요청에 대한 서비스 연결 역할

Amazon EC2는 다른 AWS 서비스를 자동으로 호출하는 데 필요한 권한에 서비스 연결 역할을 사용합니다. 서비스 연결 역할은 AWS 서비스에 직접 연결된 고유한 유형의 IAM 역할입니다. 연결된 서비스만 서비스 연결 역할을 담당할 수 있으므로 서비스 연결 역할은 AWS 서비스로 권한을 위임하는 안전한 방법을 제공합니다. 자세한 내용은 IAM 사용 설명서의 [서비스 연결 역할 사용](#)을 참조하십시오.

Amazon EC2는 [AWS Service Role for EC2 Spot](#)이라는 이름의 서비스 연결 역할을 사용하여 사용자 대신 스팟 인스턴스를 시작하고 관리합니다.

AWS Service Role for EC2 Spot에서 부여된 권한

Amazon EC2는 [AWS Service Role for EC2 Spot](#)을 사용하여 다음 작업을 완료합니다.

- [ec2:DescribeInstances](#) – 스팟 인스턴스 설명
- [ec2:StopInstances](#) – 스팟 인스턴스 중지
- [ec2:StartInstances](#) – 스팟 인스턴스 시작

서비스 연결 역할 생성

대부분의 경우 서비스 연결 역할을 수동으로 생성할 필요가 없습니다. Amazon EC2는 처음으로 콘솔을 사용하여 스팟 인스턴스를 요청할 때 [AWS Service Role for EC2 Spot](#) 서비스 연결 역할을 생성합니다.

Amazon EC2가 이 서비스 연결 역할을 지원하기 시작한 2017년 10월 이전에 활성 스팟 인스턴스를 요청한 경우 Amazon EC2에서 사용자 AWS 계정에 AWSServiceRoleForEC2Spot 역할을 이미 생성했습니다. 자세한 내용은 IAM 사용 설명서의 [내 계정에 표시되는 새 역할](#)을 참조하십시오.

AWS CLI 또는 API를 사용하여 스팟 인스턴스를 요청하려면 먼저 이 역할이 있어야 합니다. 역할을 생성하려면 다음과 같이 IAM 콘솔을 사용하십시오.

AWSServiceRoleForEC2Spot 서비스 연결 역할을 수동으로 생성하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 역할을 선택합니다.
3. 역할 생성을 선택합니다.
4. 신뢰할 수 있는 유형의 엔터티 선택 페이지의 EC2에서 EC2 - 스팟 인스턴스를 선택한 후 다음: 권한을 선택합니다.
5. 다음 페이지에서 다음:검토(Next:Review)를 선택합니다.
6. 검토 페이지에서 역할 만들기를 선택합니다.

스팟 인스턴스가 더 이상 필요 없으면 AWSServiceRoleForEC2Spot 역할을 삭제하는 것이 좋습니다. 계정에서 이 역할이 삭제된 후 스팟 인스턴스를 요청하면 Amazon EC2에서 다시 해당 역할을 생성합니다.

암호화된 AMI 및 EBS 스냅샷과 함께 사용할 CMK에 대한 액세스 권한 부여

스팟 인스턴스 요청에서 [암호화된 AMI \(p. 98\)](#) 또는 [암호화된 Amazon EBS 스냅샷 \(p. 1033\)](#)을 지정하고 암호화용 고객 관리형 고객 마스터 키(CMK)를 사용하는 경우 AWSServiceRoleForEC2Spot 역할에 CMK를 사용할 수 있는 권한을 부여해야 Amazon EC2가 사용자를 대신하여 스팟 인스턴스를 시작할 수 있습니다. 이렇게 하려면 다음 절차에 표시된 바와 같이 CMK에 권한을 추가해야 합니다.

권한을 제공할 때 권한 부여는 키 정책을 대체합니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [권한 부여 사용 및 AWS KMS에서 키 정책 사용](#) 단원을 참조하십시오.

CMK를 사용할 수 있도록 AWSServiceRoleForEC2Spot 역할 권한을 부여하려면

- `create-grant` 명령을 사용하여 CMK에 권한을 추가하고 허용된 작업을 수행할 수 있는 권한이 부여된 보안 주체(AWSServiceRoleForEC2Spot 서비스 연결 역할)를 지정합니다. CMK는 CMK의 `key-id` 파라미터 및 ARN에 의해 지정됩니다. 보안 주체는 AWSServiceRoleForEC2Spot 서비스 연결 역할의 `grantee-principal` 파라미터 및 ARN에 의해 지정됩니다.

```
aws kms create-grant \
  --region us-east-1 \
  --key-id arn:aws:kms:us-
east-1:44445556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
  --grantee-principal arn:aws:iam::111122223333:role/AWSServiceRoleForEC2Spot \
  --operations "Decrypt" "Encrypt" "GenerateDataKey"
"GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
"ReEncryptTo"
```

스팟 인스턴스 요청 생성

스팟 인스턴스를 요청하는 절차는 온디맨드 인스턴스를 시작하는 절차와 유사합니다. 다음과 같은 방법으로 스팟 인스턴스를 요청할 수 있습니다.

- 콘솔을 사용하여 스팟 인스턴스를 요청하려면 인스턴스 시작 마법사를 사용합니다. 자세한 내용은 [스팟 인스턴스 요청을 생성하려면\(콘솔\) \(p. 259\)](#) 단원을 참조하십시오.
- CLI를 사용하여 스팟 인스턴스를 요청하려면 `request-spot-instances` 명령 또는 `run-instances` 명령을 사용합니다. 자세한 내용은 `request-spot-instances`를 사용하여 스팟 인스턴스 요청을 생성하려면(AWS CLI) 및 `run-instances`를 사용하여 스팟 인스턴스 요청을 생성하려면(AWS CLI) 단원을 참조하십시오.

- 콘솔을 사용하여 정의된 기간으로 스팟 인스턴스를 요청하려면 [스팟 집합 요청 생성 \(p. 277\)](#) 절차를 따르십시오. 필요한 애플리케이션 또는 작업 알려주기에서 정의된 기간 워크로드를 선택합니다. 자세한 내용은 [스팟 인스턴스의 지속 시간 정의 \(p. 256\)](#) 단원을 참조하십시오.
- CLI를 사용하여 정의된 기간으로 스팟 인스턴스를 요청하려면 `request-spot-instances` 명령을 사용하고 `--block-duration-minutes` 파라미터를 지정합니다. 자세한 내용은 [스팟 인스턴스의 지속 시간 정의 \(p. 256\)](#) 단원을 참조하십시오.

스팟 인스턴스 요청을 제출한 후에는 요청의 파라미터를 변경할 수 없습니다. 즉, 지불하려는 최고 가격을 변경할 수 없습니다.

여러 스팟 인스턴스를 한 번에 요청하는 경우 각 요청 상태를 개별적으로 추적할 수 있도록 Amazon EC2에서 개별 스팟 인스턴스 요청을 생성합니다. 스팟 인스턴스 요청 추적에 대한 자세한 내용은 [스팟 요청 상태 \(p. 304\)](#) 단원을 참조하세요.

스팟 인스턴스 및 온디맨드 인스턴스가 포함된 플릿을 시작하려면 [스팟 집합 요청 생성 \(p. 277\)](#) 단원을 참조하십시오.

Note

인스턴스 시작 마법사 또는 `run-instances` 명령을 사용하여 동일한 호출에서 스팟 인스턴스 및 온디맨드 인스턴스를 시작할 수 없습니다.

사전 조건

시작하기 전에 최고 가격, 원하는 스팟 인스턴스 수, 사용할 인스턴스 유형을 결정합니다. 스팟 가격 추세를 검토하려면 [스팟 인스턴스 요금 기록 \(p. 252\)](#) 단원을 참조하십시오.

스팟 인스턴스 요청을 생성하려면(콘솔)

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 화면 상단의 탐색 모음에서 리전을 선택합니다.
- Amazon EC2 콘솔 대시보드에서 인스턴스 시작을 선택합니다.
- Amazon Machine Image(AMI) 선택 페이지에서 AMI를 선택합니다. 자세한 내용은 [1단계: Amazon 머신 이미지\(AMI\) 선택 \(p. 377\)](#) 단원을 참조하십시오.
- 인스턴스 유형 선택 페이지에서 시작할 하드웨어 구성 및 인스턴스 크기를 선택하고 다음: 인스턴스 세부 정보 구성을 선택합니다. 자세한 내용은 [2단계: 인스턴스 유형 선택 \(p. 377\)](#) 단원을 참조하십시오.
- 인스턴스 세부 정보 구성 페이지에서 다음과 같이 스팟 인스턴스 요청을 구성합니다.
 - 인스턴스 개수: 시작할 인스턴스의 수를 입력합니다.

Note

Amazon EC2는 각 스팟 인스턴스에 대해 별도의 요청을 생성합니다.

- (선택 사항) 애플리케이션 수요를 처리할 인스턴스의 수를 올바르게 유지하는데 도움을 주기 위해 Auto Scaling 그룹 시작을 선택해 시작 구성 및 Auto Scaling 그룹을 생성할 수 있습니다. Auto Scaling는 사양에 따라 그룹의 인스턴스 수를 조정합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 사용 설명서](#) 단원을 참조하십시오.
- 구입 옵션: 스팟 인스턴스를 시작하려면 스팟 인스턴스 요청을 선택합니다. 이 옵션을 선택하면 다음 필드가 나타납니다.
 - 현재 가격: 선택한 인스턴스 유형에 대해 각 가용 영역의 현재 스팟 가격이 표시됩니다.
 - (선택 사항) 최고 가격: 필드를 비워 두거나 지불할 최대 금액을 지정할 수 있습니다.
 - 이 필드를 비워두면 최고 가격이 현재 온디맨드 가격으로 기본 설정됩니다. 스팟 인스턴스가 온디맨드 가격 이하의 현재 스팟 가격으로 시작됩니다.
 - 현재 스팟 가격보다 높은 최고 가격을 지정하면 스팟 인스턴스가 현재 스팟 가격으로 시작되고 현재 스팟 가격으로 요금이 부과됩니다.

- 스팟 가격보다 낮은 최고 가격을 지정하면 스팟 인스턴스가 시작되지 않습니다.
- 연속 요청: 스팟 인스턴스가 종단될 경우 연속 요청을 선택하여 스팟 인스턴스 요청을 다시 제출합니다.
- 인터럽트 방식: 기본적으로 스팟 서비스는 스팟 인스턴스가 종단될 경우 이를 종료합니다. 연속 요청을 선택한 경우 스팟 인스턴스 종단 시 스팟 서비스가 이를 중지하거나 최대 절전 모드로 전환하도록 지정할 수 있습니다. 자세한 내용은 [종단 동작 \(p. 311\)](#) 단원을 참조하십시오.
- (선택 사항) 요청 유효 종료 시간: 편집을 선택하여 스팟 인스턴스 요청이 만료되는 시기를 지정합니다.

스팟 인스턴스 구성에 대한 자세한 내용은 [3단계: 인스턴스 세부 정보 구성 \(p. 378\)](#) 단원을 참조하십시오.

7. 선택한 AMI에는 루트 디바이스 볼륨을 포함한 하나 이상의 스토리지 볼륨이 있습니다. 스토리지 추가 페이지에서 새 볼륨 추가를 선택하여 인스턴스에 연결할 추가 볼륨을 지정할 수 있습니다. 자세한 내용은 [4단계: 스토리지 추가 \(p. 380\)](#) 단원을 참조하십시오.
8. 태그 추가 페이지에서 키와 값의 조합을 제공하여 [태그 \(p. 1143\)](#)를 지정합니다. 자세한 내용은 [5단계: 태그 추가 \(p. 380\)](#) 단원을 참조하십시오.
9. 보안 그룹 구성 페이지에서 기존 보안 그룹을 사용하여 인스턴스의 방화벽 규칙을 정의할 수 있습니다. 이 규칙은 인스턴스에 전달되는 수신 네트워크 트래픽을 정의합니다. 다른 모든 트래픽은 무시됩니다. (보안 그룹에 대한 자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#)을 참조하십시오.) 그룹을 선택하거나 새로 생성하고 검토 및 시작을 선택합니다. 자세한 내용은 [6단계: 보안 그룹 구성 \(p. 381\)](#) 단원을 참조하십시오.
10. 인스턴스 시작 검토 페이지에서 인스턴스 세부 정보를 확인한 다음, 해당되는 편집 링크를 선택하여 필요한 사항을 변경합니다. 준비가 완료되면 시작을 선택합니다. 자세한 내용은 [7단계: 인스턴스 시작 검토 및 키 페어 선택 \(p. 381\)](#) 단원을 참조하십시오.
11. 기존 키 페어 선택 또는 새 키 페어 생성(Select an existing key pair or create a new key pair) 대화 상자에서 기존 키 페어를 선택하거나 새 키 페어를 만들 수 있습니다. 예를 들어, 기존 키 페어 선택하고 초기 설정에서 생성한 키 페어를 선택합니다. 자세한 내용은 [Amazon EC2 키 페어 및 Windows 인스턴스 \(p. 900\)](#) 단원을 참조하십시오.

Important

키 페어 없이 진행(Proceed without key pair) 옵션을 선택할 경우 사용자가 다른 방법으로 로그인할 수 있도록 구성된 AMI를 선택해야만 인스턴스에 연결할 수 있습니다.

12. 인스턴스를 시작하려면 승인 확인란을 선택한 후 인스턴스 시작을 선택합니다.

인스턴스가 시작하지 않거나 상태가 terminated이 아닌 running로 변경되는 경우, [인스턴스 시작 문제 해결 \(p. 1178\)](#) 단원을 참고하십시오.

`request-spot-instances`를 사용하여 스팟 인스턴스 요청을 생성하려면(AWS CLI)

`request-spot-instances` 명령을 사용하여 일회성 요청을 생성합니다.

```
aws ec2 request-spot-instances \
--instance-count 5 \
--type "one-time" \
--launch-specification file://specification.json
```

`request-spot-instances` 명령을 사용하여 영구 요청을 생성합니다.

```
aws ec2 request-spot-instances \
--instance-count 5 \
--type "persistent" \
--launch-specification file://specification.json
```

이러한 명령과 함께 사용할 시작 사양 파일에 대한 예시는 [스팟 인스턴스 요청 예시 시작 사양 \(p. 269\)](#) 단원을 참조하십시오. 콘솔에서 시작 사양 파일을 다운로드하는 경우 대신 `request-spot-fleet` 명령을 사용해야 합니다(콘솔은 스팟 집합을 사용하여 스팟 요청을 지정함).

`run-instances`를 사용하여 스팟 인스턴스 요청을 생성하려면(AWS CLI)

`run-instances` 명령을 사용하고 `--instance-market-options` 파라미터에 스팟 인스턴스 옵션을 지정합니다.

```
aws ec2 run-instances \
--image-id ami-0abcdef1234567890 \
--instance-type t2.micro \
--instance-count 5 \
--subnet-id subnet-08fc749671b2d077c \
--key-name MyKeyPair \
--security-group-ids sg-0b0384b66d7d692f9 \
--instance-market-options file://spot-options.json
```

다음은 `--instance-market-options`에 대해 JSON 파일에 지정할 데이터 구조입니다.

`BlockDurationMinutes`, `ValidUntil` 및 `InstanceInterruptionBehavior`도 지정할 수 있습니다. 데이터 구조에서 필드를 지정하지 않으면 기본값이 사용됩니다. 이 예제에서는 `one-time` 요청을 생성하고 스팟 인스턴스에 지불하고자 하는 최고 가격으로 0.02를 지정합니다.

```
{
  "MarketType": "spot",
  "SpotOptions": {
    "MaxPrice": "0.02",
    "SpotInstanceType": "one-time"
  }
}
```

실행 중인 스팟 인스턴스 찾기

Amazon EC2는 최고 가격이 스팟 가격을 초과하고 용량이 가용 상태일 때 스팟 인스턴스를 시작합니다. 스팟 인스턴스는 중단되거나 사용자가 직접 종료할 때까지 실행됩니다. 최고 가격이 스팟 가격과 정확히 같은 경우에는 수요에 따라 스팟 인스턴스가 계속 실행됩니다.

실행 중인 스팟 인스턴스를 찾으려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다. 스팟 인스턴스 요청과 스팟 집합 요청을 모두 볼 수 있습니다. 스팟 인스턴스 요청이 이행된 경우 용량은 스팟 인스턴스의 ID입니다. 스팟 집합의 경우 용량은 요청된 용량 중 충족된 용량을 나타냅니다. 스팟 집합에서 인스턴스의 ID를 보려면 확장 화살표를 선택하거나 플릿을 선택한 후 인스턴스를 선택합니다.

Note

스팟 집합에서 생성된 스팟 인스턴스 요청의 경우, 요청이 속해 있는 스팟 집합을 나타내는 시스템 태그가 요청에 즉시 지정되지 않으며 일정 기간 동안 요청이 스팟 집합 요청과 별도로 나타날 수 있습니다.

또는 탐색 창에서 인스턴스를 선택합니다. 오른쪽 상단 모서리에서 표시/숨기기 아이콘을 선택하고 인스턴스 속성에서 수명 주기를 선택합니다. 각 인스턴스에 대해 수명 주기는 `normal`, `spot` 또는 `scheduled`입니다.

실행 중인 스팟 인스턴스를 찾으려면(AWS CLI)

스팟 인스턴스를 나열하려면 `--query` 옵션과 함께 `describe-spot-instance-requests` 명령을 사용합니다.

```
aws ec2 describe-spot-instance-requests \
--query "SpotInstanceRequests[*].{ID:InstanceId}"
```

다음은 예제 출력입니다.

```
[  
  {  
    "ID": "i-1234567890abcdef0"  
  },  
  {  
    "ID": "i-0598c7d356eba48d7"  
  }  
]
```

또는 --filters 옵션과 함께 [describe-instance](#) 명령을 사용하여 스팟 인스턴스를 나열할 수도 있습니다.

```
aws ec2 describe-instances \
--filters "Name=instance-lifecycle,Values=spot"
```

단일 스팟 인스턴스 인스턴스를 설명하려면 [describe-spot-instance-requests](#) 명령을 --spot-instance-request-ids 옵션과 함께 사용합니다.

```
aws ec2 describe-spot-instance-requests \
--spot-instance-request-ids sir-08b93456
```

스팟 인스턴스 요청 태그 지정

스팟 인스턴스 요청을 쉽게 분류하고 관리할 수 있도록 사용자 지정 메타데이터로 이 요청에 태그를 지정할 수 있습니다. 스팟 인스턴스 요청을 생성할 때 또는 생성한 후 요청에 태그를 지정할 수 있습니다. Amazon EC2 콘솔이나 명령줄 도구를 사용하여 태그를 지정할 수 있습니다.

스팟 인스턴스 요청에 태그를 지정하면 스팟 인스턴스 요청에서 시작된 인스턴스 및 볼륨은 자동으로 태그가 지정되지 않습니다. 스팟 인스턴스 요청에서 시작된 인스턴스 및 볼륨에 명시적으로 태그를 지정해야 합니다. 시작 중이나 이후에 스팟 인스턴스 및 볼륨에 태그를 지정할 수 있습니다.

태그 작동 방식에 대한 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

목차

- [사전 조건 \(p. 262\)](#)
- [새 스팟 인스턴스 요청에 태그 지정 \(p. 264\)](#)
- [기존 스팟 인스턴스 요청에 태그 지정 \(p. 265\)](#)
- [스팟 인스턴스 요청 태그 보기 \(p. 265\)](#)

사전 조건

IAM 사용자에게 리소스에 태그를 지정할 수 있는 권한을 부여합니다. IAM 정책 및 예제 정책에 대한 자세한 내용은 [예제: 리소스에 태그 지정 \(p. 874\)](#) 단원을 참조하십시오.

생성하는 IAM 정책은 스팟 인스턴스 요청 생성에 사용하는 방법에 따라 결정됩니다.

- 인스턴스 시작 마법사 또는 [run-instances](#)를 사용하여 스팟 인스턴스를 요청하는 경우 [인스턴스 시작 마법사 또는 run-instances를 사용할 때 IAM 사용자에게 리소스에 태그를 지정할 수 있는 권한을 부여하려면](#) 단원을 참조하십시오.
- 스팟 콘솔을 사용하여 정의된 기간으로 스팟 인스턴스를 요청하거나 [request-spot-instances](#) 명령을 사용하여 스팟 인스턴스를 요청하는 경우 [request-spot-instances를 사용할 때 IAM 사용자에게 리소스에 태그를 지정할 수 있는 권한을 부여하려면](#) 단원을 참조하십시오.

인스턴스 시작 마법사 또는 run-instances를 사용할 때 IAM 사용자에게 리소스에 태그를 지정할 수 있는 권한을 부여하려면

다음을 포함하는 IAM 정책을 만듭니다.

- ec2:RunInstances 작업입니다. IAM 사용자에게 인스턴스 시작 권한이 부여됩니다.
- Resource의 경우 spot-instances-request를 지정합니다. 이를 통해 사용자가 스팟 인스턴스를 요청하는 스팟 인스턴스 요청을 생성할 수 있습니다.
- ec2:CreateTags 작업. IAM 사용자에게 태그 생성 권한이 부여됩니다.
- Resource의 경우 *를 지정합니다. 이를 통해 사용자가 인스턴스 시작 중에 생성된 모든 리소스에 태그를 지정할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "AllowLaunchInstances",  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances"  
            ],  
            "Resource": [  
                "arn:aws:ec2:us-east-1::image/*",  
                "arn:aws:ec2:us-east-1::subnet/*",  
                "arn:aws:ec2:us-east-1::network-interface/*",  
                "arn:aws:ec2:us-east-1::security-group/*",  
                "arn:aws:ec2:us-east-1::key-pair/*",  
                "arn:aws:ec2:us-east-1::volume/*",  
                "arn:aws:ec2:us-east-1::instance/*",  
                "arn:aws:ec2:us-east-1::spot-instances-request/*"  
            ]  
        },  
        {  
            "Sid": "TagSpotInstanceRequests",  
            "Effect": "Allow",  
            "Action": "ec2:CreateTags",  
            "Resource": "*"  
        }  
    ]  
}
```

Note

RunInstances 작업을 사용하여 스팟 인스턴스 요청을 만들고 생성 시 스팟 인스턴스 요청에 태그를 지정하는 경우 Amazon EC2가 RunInstances 문에서 spot-instances-request 리소스를 평가하는 방법을 알고 있어야 합니다.

spot-instances-request 리소스는 다음과 같이 IAM 정책에서 평가됩니다.

- 생성 시 스팟 인스턴스 요청에 태그를 지정하지 않으면 Amazon EC2는 RunInstances 문에서 spot-instances-request 리소스를 평가하지 않습니다.
- 생성 시 스팟 인스턴스 요청에 태그를 지정하면 Amazon EC2는 RunInstances 문에서 spot-instances-request 리소스를 평가합니다.

따라서 spot-instances-request 리소스의 경우 IAM 정책에 다음 규칙이 적용됩니다.

- RunInstances를 사용하여 스팟 인스턴스 요청을 만들고 생성 시 스팟 인스턴스 요청에 태그를 지정하지 않으려는 경우 spot-instances-request 리소스를 명시적으로 허용할 필요가 없습니다. 호출이 성공합니다.

- RunInstances를 사용하여 스팟 인스턴스 요청을 만들고 생성 시 스팟 인스턴스 요청에 태그를 지정하려는 경우 RunInstances allow 문에 spot-instances-request 리소스를 포함해야 합니다. 그렇지 않으면 호출이 실패합니다.
- RunInstances를 사용하여 스팟 인스턴스 요청을 만들고 생성 시 스팟 인스턴스 요청에 태그를 지정하려는 경우 CreateTags allow 문에서 spot-instances-request 리소스를 지정하거나 * 와일드카드를 포함해야 합니다. 그렇지 않으면 호출이 실패합니다.

스팟 인스턴스 요청에 지원되지 않는 정책을 포함한 예제 IAM 정책은 [스팟 인스턴스 작업 \(p. 869\)](#) 단원을 참조하십시오.

request-spot-instances를 사용할 때 IAM 사용자에게 리소스에 태그를 지정할 수 있는 권한을 부여하려면 다음을 포함하는 IAM 정책을 만듭니다.

- ec2:RequestSpotInstances 작업. IAM 사용자에게 스팟 인스턴스 요청 생성 권한이 부여됩니다.
- ec2:CreateTags 작업. IAM 사용자에게 태그 생성 권한이 부여됩니다.
- Resource의 경우 spot-instances-request를 지정합니다. 이를 통해 사용자가 스팟 인스턴스 요청에만 태그를 지정할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "TagSpotInstanceRequest",  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RequestSpotInstances",  
                "ec2:CreateTags"  
            ],  
            "Resource": "arn:aws:ec2:us-east-1:111122223333:spot-instances-request/*"  
        }  
    ]  
}
```

새 스팟 인스턴스 요청에 태그 지정

콘솔을 사용하여 새 스팟 인스턴스 요청에 태그를 지정하려면

- [스팟 인스턴스 요청 생성 \(p. 258\)](#)의 절차를 따르십시오.
- 태그를 추가하려면 태그 추가 페이지에서 태그 추가를 선택하고 해당 태그에 대한 키와 값을 입력합니다. 각 추가 태그에 다른 태그 추가를 선택합니다.
각 태그에 대해 동일한 태그로 스팟 인스턴스 요청, 스팟 인스턴스 및 볼륨에 태그를 지정할 수 있습니다. 세 가지 모두에 태그를 지정하려면 인스턴스, 볼륨, 및 스팟 인스턴스 요청이 선택되어 있는지 확인합니다. 한 두 개만 태그를 지정하려면 태그를 지정할 리소스가 선택되어 있고 나머지 리소스는 선택 취소되어 있는지 확인합니다.
- 필수 필드를 입력하여 스팟 인스턴스 요청을 생성한 다음 시작을 선택합니다. 자세한 내용은 [스팟 인스턴스 요청 생성 \(p. 258\)](#) 단원을 참조하십시오.

AWS CLI를 사용하여 새 스팟 인스턴스 요청에 태그를 지정하려면

스팟 인스턴스 요청을 생성할 때 태그를 지정하려면 다음과 같이 스팟 인스턴스 요청 구성을 구성합니다.

- tag-specification 파라미터를 사용하여 스팟 인스턴스 요청에 대한 태그를 지정합니다.
- ResourceType의 경우 spot-instances-request를 지정합니다. 다른 값을 지정하면 스팟 인스턴스 요청이 실패합니다.
- Tags에 대해 키-값 페어를 지정합니다. 둘 이상의 키-값 페어를 지정할 수 있습니다.

다음 예에서 스팟 인스턴스 요청에는 Key=Environment 및 Value=Production, Key=Cost-Center 및 Value=123 두 개의 태그가 지정됩니다.

```
aws ec2 request-spot-instances \
--instance-count 5 \
--type "one-time" \
--launch-specification file://specification.json \
--tag-specification 'ResourceType=spot-instances-
request,Tags=[{Key=Environment,Value=Production},{Key=Cost-Center,Value=123}]'
```

기존 스팟 인스턴스 요청에 태그 지정

콘솔을 사용하여 기존 스팟 인스턴스 요청에 태그를 지정하려면

스팟 인스턴스 요청을 생성한 후 콘솔을 사용하여 스팟 인스턴스 요청에 태그를 추가할 수 있습니다.

1. <https://console.aws.amazon.com/ec2spot>에서 스팟 콘솔을 엽니다.
2. 스팟 인스턴스 요청을 선택합니다.
3. 태그 탭을 선택하고 태그 생성을 선택합니다.

콘솔을 사용하여 기존 스팟 인스턴스에 태그를 지정하려면

스팟 인스턴스 요청이 스팟 인스턴스를 시작한 후 콘솔을 사용하여 인스턴스에 태그를 추가할 수 있습니다.
자세한 내용은 [개별 리소스에 대한 태그 추가 및 삭제 \(p. 1149\)](#) 단원을 참조하십시오.

AWS CLI를 사용하여 기존 스팟 인스턴스 요청 또는 스팟 인스턴스에 태그를 지정하려면

`create-tags` 명령을 사용하여 기존 리소스에 태그를 지정합니다. 다음 예에서, 기존 스팟 인스턴스 요청 및 스팟 인스턴스는 Key=purpose 및 Value=test로 태그가 지정되어 있습니다.

```
aws ec2 create-tags \
--resources sir-08b93456 i-1234567890abcdef0 \
--tags Key=purpose,Value=test
```

스팟 인스턴스 요청 태그 보기

콘솔을 사용하여 스팟 인스턴스 요청 태그를 보려면

1. <https://console.aws.amazon.com/ec2spot>에서 스팟 콘솔을 엽니다.
2. 스팟 인스턴스 요청을 선택하고 태그 탭을 선택합니다.

스팟 인스턴스 요청 태그를 설명하려면

`describe-tags` 명령을 사용하여 지정된 리소스에 대한 태그를 표시합니다. 다음 예제에서는 지정된 요청에 대한 태그를 설명합니다.

```
aws ec2 describe-tags \
--filters "Name=resource-id,Values=sir-11112222-3333-4444-5555-66666EXAMPLE"
```

```
{
  "Tags": [
    {
      "Key": "Environment",
      "ResourceId": "sir-11112222-3333-4444-5555-66666EXAMPLE",
      "ResourceType": "spot-instances-request",
```

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
스팟 인스턴스

```
        "Value": "Production"
    },
    {
        "Key": "Another key",
        "ResourceId": "sir-11112222-3333-4444-5555-66666EXAMPLE",
        "ResourceType": "spot-instances-request",
        "Value": "Another value"
    }
]
```

스팟 인스턴스 요청을 설명하여 스팟 인스턴스 요청의 태그를 볼 수도 있습니다.

`describe-spot-instance-requests` 명령을 사용하여 지정된 스팟 인스턴스 요청의 구성을 볼 수 있습니다. 여기에는 요청에 대해 지정된 태그가 모두 포함되어 있습니다.

```
aws ec2 describe-spot-instance-requests \
--spot-instance-request-ids sir-11112222-3333-4444-5555-66666EXAMPLE
```

```
{
    "SpotInstanceRequests": [
        {
            "CreateTime": "2020-06-24T14:22:11+00:00",
            "InstanceId": "i-1234567890EXAMPLE",
            "LaunchSpecification": {
                "SecurityGroups": [
                    {
                        "GroupName": "launch-wizard-6",
                        "GroupId": "sg-1234567890EXAMPLE"
                    }
                ],
                "BlockDeviceMappings": [
                    {
                        "DeviceName": "/dev/xvda",
                        "Ebs": {
                            "DeleteOnTermination": true,
                            "VolumeSize": 8,
                            "VolumeType": "gp2"
                        }
                    }
                ],
                "ImageId": "ami-1234567890EXAMPLE",
                "InstanceType": "t2.micro",
                "KeyName": "my-key-pair",
                "NetworkInterfaces": [
                    {
                        "DeleteOnTermination": true,
                        "DeviceIndex": 0,
                        "SubnetId": "subnet-11122233"
                    }
                ],
                "Placement": {
                    "AvailabilityZone": "eu-west-1c",
                    "Tenancy": "default"
                },
                "Monitoring": {
                    "Enabled": false
                }
            },
            "LaunchedAvailabilityZone": "eu-west-1c",
            "ProductDescription": "Linux/UNIX",
            "SpotInstanceRequestId": "sir-1234567890EXAMPLE",
            "SpotPrice": "0.012600",
            "State": "open"
        }
    ]
}
```

```
"State": "active",
"Status": {
    "Code": "fulfilled",
    "Message": "Your spot request is fulfilled.",
    "UpdateTime": "2020-06-25T18:30:21+00:00"
},
"Tags": [
    {
        "Key": "Environment",
        "Value": "Production"
    },
    {
        "Key": "Another key",
        "Value": "Another value"
    }
],
>Type": "one-time",
"InstanceInterruptionBehavior": "terminate"
}
]
```

스팟 인스턴스 요청 취소

스팟 인스턴스 요청이 더 이상 필요 없는 경우 해당 요청을 취소할 수 있습니다. `open`, `active`, 또는 `disabled`인 스팟 인스턴스 요청만 취소할 수 있습니다.

- 요청이 아직 이행되지 않았고 인스턴스가 시작되지 않았으면 스팟 인스턴스 요청은 `open`입니다.
- 요청이 이행되었고 그 결과 스팟 인스턴스가 시작되었으면 스팟 인스턴스 요청은 `active`입니다.
- 스팟 인스턴스를 종지하면 스팟 인스턴스 요청은 `disabled`입니다.

스팟 인스턴스 요청이 `active` 상태이고 실행 중인 스팟 인스턴스가 연결되어 있을 때 요청을 취소하면 인스턴스가 종료되지 않습니다. 스팟 인스턴스 종료에 대한 자세한 내용은 [스팟 인스턴스를 종료하는 중 \(p. 269\)](#) 단원을 참조하십시오.

스팟 인스턴스 요청을 취소하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택한 다음 스팟 요청을 선택합니다.
3. 작업을 선택한 후, 요청 취소를 선택합니다.
4. (선택 사항) 연결된 스팟 인스턴스에 대한 작업을 완료했으면 종료할 수 있습니다. 스팟 요청 취소 대화 상자에서 인스턴스 종료를 선택한 다음 확인을 선택합니다.

스팟 인스턴스 요청을 취소하려면(AWS CLI)

- `cancel-spot-instance-requests` 명령을 사용하여 지정된 스팟 요청을 취소합니다.

```
aws ec2 cancel-spot-instance-requests --spot-instance-request-ids sir-08b93456
```

스팟 인스턴스 종지

지금 스팟 인스턴스가 필요하지 않지만 나중에 Amazon EBS 볼륨에 영구적으로 있는 데이터를 잃어버리지 않고 인스턴스를 다시 시작하려면 인스턴스를 종지할 수 있습니다. 스팟 인스턴스를 종지하는 단계는 온디맨드 인스턴스를 종지하는 단계와 비슷합니다. 스팟 인스턴스가 `persistent` 스팟 인스턴스 요청에서 시작된 경우에만 스팟 인스턴스를 종지할 수 있습니다.

Note

스팟 인스턴스가 중지되었을 때 일부 인스턴스 속성을 수정할 수 있지만 인스턴스 유형은 수정할 수 없습니다.

중지된 스팟 인스턴스에 대한 사용 요금이나 데이터 전송 요금이 부과되지 않지만, 모든 Amazon EBS 볼륨에 대한 스토리지 요금이 부과됩니다.

제한 사항

- 플랫 또는 시작 그룹, 가용 영역 그룹, 스팟 블록의 일부인 경우 스팟 인스턴스를 중지할 수 없습니다.

스팟 인스턴스를 중지하려면(콘솔)

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 인스턴스를 선택하고 스팟 인스턴스를 선택합니다.
- [Actions], [Instance State], [Stop]을 차례로 선택합니다.

스팟 인스턴스를 중지하려면(AWS CLI)

- `stop-instances` 명령을 사용하여 하나 이상의 스팟 인스턴스를 수동으로 중지합니다.

```
aws ec2 stop-instances --instance-ids i-1234567890abcdef0
```

스팟 인스턴스 시작

이전에 중지한 스팟 인스턴스를 시작할 수 있습니다. 스팟 인스턴스를 시작하는 단계는 온디맨드 인스턴스를 시작하는 단계와 비슷합니다.

사전 조건

다음과 같은 경우에만 스팟 인스턴스를 시작할 수 있습니다.

- 스팟 인스턴스를 수동으로 중지했습니다.
- 스팟 인스턴스가 EBS 지원 인스턴스입니다.
- 스팟 인스턴스 용량을 사용할 수 있습니다.
- 스팟 가격이 최고 가격보다 낮습니다.

제한 사항

- 플랫 또는 시작 그룹, 가용 영역 그룹, 스팟 블록의 일부인 경우 스팟 인스턴스를 시작할 수 없습니다.

스팟 인스턴스를 시작하려면(콘솔)

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 인스턴스를 선택하고 스팟 인스턴스를 선택합니다.
- 작업, 인스턴스 상태, 시작을 차례로 선택합니다.

스팟 인스턴스를 시작하려면(AWS CLI)

- `start-instance` 명령을 사용하여 하나 이상의 스팟 인스턴스를 수동으로 시작합니다.

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

스팟 인스턴스를 종료하는 증

스팟 인스턴스 요청이 active이고 연결된 실행 중인 스팟 인스턴스가 있거나, 스팟 인스턴스 요청이 disabled이고 연결된 종지된 스팟 인스턴스가 있는 경우, 요청을 취소해도 인스턴스가 종료되지 않습니다. 실행 중인 스팟 인스턴스를 수동으로 종료해야 합니다. 영구 스팟 요청에서 시작된 실행 중이거나 종지된 스팟 인스턴스를 종료하면 새 스팟 인스턴스를 시작할 수 있도록 스팟 요청이 open 상태로 돌아갑니다. 영구 스팟 요청을 취소하고 스팟 인스턴스를 종료하려면 먼저 스팟 요청을 취소한 다음 스팟 인스턴스를 종료해야 합니다. 그렇지 않으면 영구 스팟 요청이 새로운 인스턴스를 시작할 수 있습니다. 스팟 인스턴스 요청 취소에 대한 자세한 내용은 [스팟 인스턴스 요청 취소 \(p. 267\)](#) 단원을 참조하십시오.

스팟 인스턴스를 수동으로 종료하려면(AWS CLI)

- `terminate-instances` 명령을 사용하여 스팟 인스턴스를 수동으로 종료합니다.

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0 i-0598c7d356eba48d7
```

스팟 인스턴스 요청 예시 시작 사양

다음 예에서는 `request-spot-fleet` 명령과 함께 사용하여 스팟 인스턴스 요청을 생성할 수 있는 시작 구성을 보여줍니다. 자세한 내용은 [스팟 인스턴스 요청 생성 \(p. 258\)](#) 단원을 참조하십시오.

1. [스팟 인스턴스 시작 \(p. 269\)](#)
2. [지정된 가용 영역에서 스팟 인스턴스 시작 \(p. 269\)](#)
3. [지정된 서브넷에서 스팟 인스턴스 시작 \(p. 270\)](#)
4. [전용 스팟 인스턴스 시작 \(p. 270\)](#)

예 1: 스팟 인스턴스 시작

다음 예는 가용 영역 또는 서브넷을 포함하지 않습니다. Amazon EC2는 사용자를 위한 가용 영역을 선택합니다. Amazon EC2는 선택된 가용 영역의 기본 서브넷에 있는 인스턴스를 시작합니다.

```
{  
    "ImageId": "ami-1a2b3c4d",  
    "KeyName": "my-key-pair",  
    "SecurityGroupIds": [ "sg-1a2b3c4d" ],  
    "InstanceType": "m3.medium",  
    "IamInstanceProfile": {  
        "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"  
    }  
}
```

예 2: 지정된 가용 영역에서 스팟 인스턴스 시작

다음 예는 가용 영역을 포함합니다. Amazon EC2는 지정된 가용 영역의 기본 서브넷에 있는 인스턴스를 시작합니다.

```
{  
    "ImageId": "ami-1a2b3c4d",  
    "KeyName": "my-key-pair",  
    "SecurityGroupIds": [ "sg-1a2b3c4d" ],  
    "InstanceType": "m3.medium",  
    "Placement": {  
        "AvailabilityZone": "us-west-2a"  
    }  
}
```

```
"Placement": {  
    "AvailabilityZone": "us-west-2a"  
},  
"IamInstanceProfile": {  
    "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"  
}  
}
```

예 3: 지정된 서브넷에서 스팟 인스턴스 시작

다음 예는 서브넷을 포함합니다. Amazon EC2는 지정된 서브넷에서 인스턴스를 시작합니다. VPC가 기본이 아닌 VPC인 경우, 인스턴스는 기본적으로 퍼블릭 IPv4 주소를 받지 않습니다.

```
{  
    "ImageId": "ami-1a2b3c4d",  
    "SecurityGroupIds": [ "sg-1a2b3c4d" ],  
    "InstanceType": "m3.medium",  
    "SubnetId": "subnet-1a2b3c4d",  
    "IamInstanceProfile": {  
        "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"  
    }  
}
```

기본이 아닌 VPC에서 인스턴스에 퍼블릭 IPv4 주소를 할당하려면 다음 예시와 같이 AssociatePublicIpAddress 필드를 지정하십시오. 네트워크 인터페이스를 지정할 때는 예 3과 같은 SubnetId 및 SecurityGroupIds 필드를 사용하는 대신 네트워크 인터페이스를 사용해 서브넷 ID 및 보안 그룹 ID를 반드시 포함시켜야 합니다.

```
{  
    "ImageId": "ami-1a2b3c4d",  
    "KeyName": "my-key-pair",  
    "InstanceType": "m3.medium",  
    "NetworkInterfaces": [  
        {  
            "DeviceIndex": 0,  
            "SubnetId": "subnet-1a2b3c4d",  
            "Groups": [ "sg-1a2b3c4d" ],  
            "AssociatePublicIpAddress": true  
        }  
    ],  
    "IamInstanceProfile": {  
        "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"  
    }  
}
```

예 4: 전용 스팟 인스턴스 시작

다음 예에서는 dedicated의 테넌시를 사용하여 스팟 인스턴스를 요청합니다. 전용 스팟 인스턴스는 VPC에서 시작해야 합니다.

```
{  
    "ImageId": "ami-1a2b3c4d",  
    "KeyName": "my-key-pair",  
    "SecurityGroupIds": [ "sg-1a2b3c4d" ],  
    "InstanceType": "c3.8xlarge",  
    "SubnetId": "subnet-1a2b3c4d",  
    "Placement": {  
        "Tenancy": "dedicated"  
    }  
}
```

스팟 집합 요청 건

스팟 집합을 사용하려면 목표 용량, 옵션 온디맨드 부분, 인스턴스에 대한 하나 이상의 시작 사양, 지불하고자 하는 최고 가격을 포함한 스팟 집합 요청을 생성합니다. Amazon EC2가 스팟 가격 변경에 따라 스팟 집합의 목표 용량을 유지하도록 시도합니다. 자세한 내용은 [스팟 집합 작동 방식 \(p. 246\)](#) 단원을 참조하십시오.

스팟 집합 요청에는 두 가지 유형인 `request`와 `maintain`이 있습니다. 스팟 집합을 생성하여 원하는 용량을 위한 일회성 요청을 제출하거나, 일정 시간 동안 목표 용량을 유지하도록 요구할 수 있습니다. 두 가지 유형의 요청 모두 스팟 집합의 할당 전략에 따른 이익을 볼 수 있습니다.

일회성 요청을 생성하는 경우 스팟 집합에서는 필요한 요청을 수행하되, 용량이 줄어도 스팟 인스턴스를 보충하려고 하지는 않습니다. 용량이 가용 상태가 아닌 경우 스팟 집합은 대체 스팟 풀에서 요청을 제출하지 않습니다.

목표 용량 유지를 위해 스팟 집합은 이 목표 용량을 충족하도록 요청을 수행하고 중단된 모든 인스턴스를 자동으로 보충합니다.

일회성 요청이 일단 제출되고 나면 이 요청의 목표 용량을 수정할 수 없습니다. 목표 용량을 변경하려면 요청을 취소하고 새 요청을 제출합니다.

스팟 집합 요청은 요청이 만료되거나 사용자가 요청을 취소할 때까지 활성 상태로 유지됩니다. 스팟 집합 요청을 취소할 경우에는 스팟 집합 요청을 취소하면 스팟 집합에 속한 스팟 인스턴스가 종료되는지 여부를 지정할 수 있습니다.

각 시작 사양에는 AMI, 인스턴스 유형, 서브넷 또는 가용 영역, 하나 이상의 보안 그룹과 같이 Amazon EC2가 인스턴스를 시작하는 데 필요로 하는 정보가 포함됩니다.

목차

- [스팟 집합 요청 상태 \(p. 271\)](#)
- [스팟 집합 상태 확인 \(p. 272\)](#)
- [스팟 집합 요청 계획 \(p. 272\)](#)
- [스팟 집합 권한 \(p. 273\)](#)
- [스팟 집합 요청 생성 \(p. 277\)](#)
- [스팟 집합 태그 지정 \(p. 280\)](#)
- [스팟 집합 모니터링 \(p. 286\)](#)
- [스팟 집합 요청 수정 \(p. 287\)](#)
- [스팟 집합 요청 취소 \(p. 288\)](#)
- [스팟 집합 구성의 예 \(p. 289\)](#)

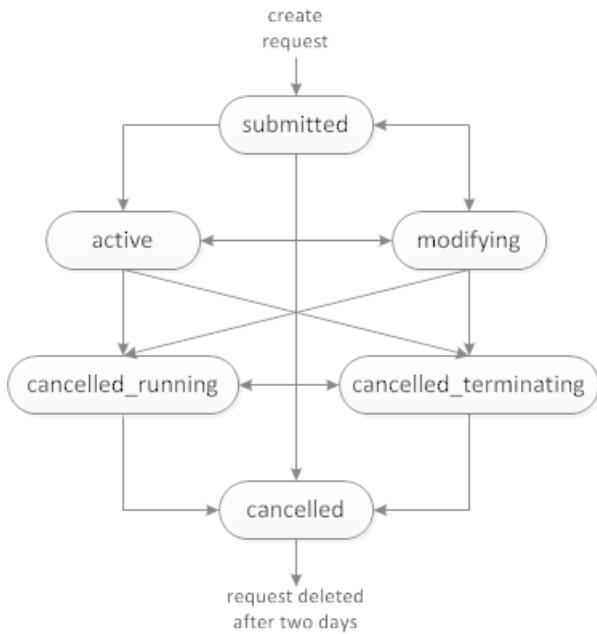
스팟 집합 요청 상태

스팟 집합 요청은 다음 상태 중 하나일 수 있습니다.

- `submitted` – 스팟 집합 요청을 평가 종이며 Amazon EC2가 목표 개수만큼 인스턴스를 시작하기 위해 준비 중입니다.
- `active` – 스팟 집합 요청이 확인되었으며 Amazon EC2에서는 실행 중인 스팟 인스턴스를 목표 개수만큼 유지하려 하고 있습니다. 그 요청은 수정 또는 취소될 때까지 계속 이 상태로 유지됩니다.
- `modifying` – 스팟 집합 요청을 수정하고 있습니다. 그 요청은 수정이 완전히 처리될 때까지 또는 스팟 집합이 취소될 때까지 계속 이 상태로 유지됩니다. 일회성 `request`는 수정할 수 없으며, 이 상태가 이런 스팟 요청에 적용되지 않습니다.
- `cancelled_running` – 스팟 집합이 취소되고 추가 스팟 인스턴스를 시작하지 않습니다. 중단되거나 종료될 때까지 기존 스팟 인스턴스가 계속 실행됩니다. 그 요청은 모든 인스턴스가 중단 또는 종료될 때까지 계속 이 상태로 유지됩니다.
- `cancelled_terminating` – 스팟 집합이 취소되고 해당 스팟 인스턴스가 종료 중입니다. 그 요청은 모든 인스턴스가 종료될 때까지 계속 이 상태로 유지됩니다.

- **cancelled** – 스팟 집합이 취소되고 실행 중인 스팟 인스턴스가 없습니다. 스팟 집합 요청은 인스턴스 종료 2일 후에 삭제됩니다.

다음 그림은 요청 상태 간의 전환을 나타냅니다. 스팟 집합 한계를 초과하면 즉시 요청이 취소됩니다.



스팟 집합 상태 확인

스팟 집합은 2분마다 폴릿에 있는 스팟 인스턴스의 상태를 확인합니다. 인스턴스의 상태는 `healthy` 또는 `unhealthy`입니다. 스팟 집합은 Amazon EC2에서 제공하는 상태 확인을 사용하여 인스턴스의 상태를 판단합니다. 세 번의 연속 상태 확인에서 인스턴스 상태 또는 시스템 상태가 `impaired`이면, 해당 인스턴스의 상태는 `unhealthy`입니다. 그렇지 않으면 상태는 `healthy`입니다. 자세한 내용은 [인스턴스 상태 확인 \(p. 651\)](#) 단원을 참조하십시오.

비정상 인스턴스를 교체하도록 스팟 집합을 구성할 수 있습니다. 상태 확인 교체를 활성화하면 상태가 `unhealthy`로 보고된 인스턴스가 교체됩니다. 스팟 집합은 비정상 인스턴스가 교체되는 동안 최대 몇 분간 목표 용량보다 적어질 수 있습니다.

요구 사항

- 상태 확인 교체는 목표 용량을 유지하는 스팟 집합에서만 지원되고 1회용 스팟 집합에서는 지원되지 않습니다.
- 비정상 인스턴스를 생성할 경우에만 이를 교체하도록 스팟 집합을 구성할 수 있습니다.
- IAM 사용자는 `ec2:DescribeInstanceStatus` 작업을 호출할 권한이 있는 경우에만 상태 확인 교체를 사용할 수 있습니다.

스팟 집합 요청 계획

스팟 집합 요청을 생성하려면 그 전에 먼저 [스팟 모범 사례](#)를 살펴보는 것이 좋습니다. 특히 스팟 집합 요청을 계획하여 원하는 인스턴스 유형을 최저 가격으로 프로비저닝하려면 이러한 모범 사례가 필요합니다. 또한, 다음을 수행하는 것이 좋습니다.

- 원하는 목표 용량을 위한 일회성 요청을 제출하는 스팟 집합을 생성할지, 아니면 시간 경과에 따라 목표 용량을 유지하는 스팟 폴릿을 생성할지 여부를 결정합니다.
- 인스턴스 유형을 결정하고 애플리케이션 요구를 만족합니다.

- 스팟 집합 요청의 목표 용량을 결정합니다. 인스턴스 또는 사용자 지정 단위에서 목표 용량을 설정할 수 있습니다. 자세한 내용은 [스팟 집합 인스턴스 가중치 부여 \(p. 249\)](#) 단원을 참조하십시오.
- 스팟 집합 목표 용량 중에서 온디맨드 용량이어야 하는 부분을 결정합니다. 온디맨드 용량으로 0을 지정할 수 있습니다.
- 인스턴스 가중치를 사용하는 경우에는 단위당 가격을 결정합니다. 단위당 가격을 계산하려면 인스턴스 시간당 가격을 이 인스턴스가 나타내는 단위 수(또는 가중치)로 나눕니다. 인스턴스 가중치를 사용하지 않는 경우 단위당 기본 가격은 인스턴스 시간당 가격입니다.
- 스팟 집합 요청에 대해 가능한 옵션을 살펴봅니다. 자세한 내용은 AWS CLI Command Reference의 [request-spot-fleet](#) 명령을 참조하십시오. 추가 예제는 다음([스팟 집합 구성의 예 \(p. 289\)](#))을 참조하십시오.

스팟 집합 권한

IAM 사용자가 스팟 집합을 생성하거나 관리하는 경우 필요한 권한을 부여해야 합니다.

Amazon EC2 콘솔을 사용하여 스팟 집합을 생성하는 경우 AWSServiceRoleForEC2SpotFleet라는 서비스 연결 역할과 스팟 집합에게 사용자 대신 리소스를 요청, 시작, 종료 및 태그 지정할 수 있는 권한을 부여하는 aws-ec2-spot-fleet-tagging-role라는 역할이 생성됩니다. AWS CLI 또는 API를 사용하는 경우 이러한 역할이 존재하는지 확인해야 합니다.

다음 지침에 따라 필요한 권한을 부여하고 역할을 생성합니다.

사용 권한 및 역할

- [IAM 사용자에게 스팟 집합에 대한 권한 부여 \(p. 273\)](#)
- [스팟 집합의 서비스 연결 역할 \(p. 275\)](#)
- [스팟 집합에 대한 IAM 역할 \(p. 276\)](#)

IAM 사용자에게 스팟 집합에 대한 권한 부여

스팟 집합을 생성하거나 관리하는 IAM 사용자에게는 다음과 같이 필요한 권한을 부여해야 합니다.

스팟 집합의 IAM 사용자 권한을 부여하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 정책을 선택한 후 정책 생성을 선택합니다.
3. 정책 생성 페이지에서 JSON을 선택하고 텍스트를 다음과 같이 바꿉니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances",  
                "ec2:CreateTags",  
                "ec2:RequestSpotFleet",  
                "ec2:ModifySpotFleetRequest",  
                "ec2:CancelSpotFleetRequests",  
                "ec2:DescribeSpotFleetRequests",  
                "ec2:DescribeSpotFleetInstances",  
                "ec2:DescribeSpotFleetRequestHistory"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": "iam:PassRole",  
            "Resource": "arn:aws:iam::aws:policy/AmazonEC2SPotFleetRole"  
        }  
    ]  
}
```

```
        "Resource": "arn:aws:iam::*:role/aws-ec2-spot-fleet-tagging-role"
    },
{
    "Effect": "Allow",
    "Action": [
        "iam:CreateServiceLinkedRole",
        "iam>ListRoles",
        "iam>ListInstanceProfiles"
    ],
    "Resource": "*"
}
]
```

위의 예제 정책은 IAM 사용자에게 대부분의 스팟 집합 사용 사례에 필요한 권한을 부여합니다. 사용자를 특정 API 작업으로 제한하려면 대신 해당 작업만 지정하십시오.

필수 EC2 및 IAM API

다음 API가 정책에 포함되어야 합니다.

- `ec2:RunInstances` – 스팟 집합에서 인스턴스를 시작하는 데 필요합니다.
- `ec2>CreateTags` – 스팟 집합 요청, 인스턴스 또는 볼륨에 태그를 지정하는 데 필요합니다.
- `iam:PassRole` – 스팟 집합 역할을 지정하는 데 필요합니다.
- `iam>CreateServiceLinkedRole` – 서비스 연결 역할을 생성하는 데 필요합니다.
- `iam>ListRoles` – 기존 IAM 역할을 열거하는 데 필요합니다.
- `iam>ListInstanceProfiles` – 기존 인스턴스 프로파일을 열거하는 데 필요합니다.

Important

시작 사양 또는 시작 템플릿에서 IAM 인스턴스 프로파일에 대한 역할을 지정하는 경우, IAM 사용자에게 역할을 서비스에 전달할 수 있는 권한을 부여해야 합니다. 이를 위해 IAM 정책에 `iam:PassRole` 작업을 위한 `"arn:aws:iam::*:role/IamInstanceProfile-role"` 리소스를 포함시키십시오. 자세한 내용은 IAM 사용 설명서의 [사용자에게 AWS 서비스에 역할을 전달할 수 있는 권한 부여](#) 단원을 참조하십시오.

스팟 집합 API

필요에 따라 다음 스팟 집합 API 작업을 정책에 추가합니다.

- `ec2:RequestSpotFleet`
- `ec2:ModifySpotFleetRequest`
- `ec2:CancelSpotFleetRequests`
- `ec2:DescribeSpotFleetRequests`
- `ec2:DescribeSpotFleetInstances`
- `ec2:DescribeSpotFleetRequestHistory`

선택적 IAM API

(선택 사항) IAM 사용자가 IAM 콘솔을 사용하여 역할 또는 인스턴스 프로파일을 생성할 수 있도록 하려면 정책에 다음 작업을 추가해야 합니다.

- `iam>AddRoleToInstanceProfile`
- `iam:AttachRolePolicy`
- `iam>CreateInstanceProfile`

- iam:CreateRole
 - iam:GetRole
 - iam>ListPolicies
4. [Review policy]를 선택합니다.
5. 정책 검토 페이지에 정책 이름과 설명을 입력한 다음 정책 생성을 선택합니다.
6. 탐색 창에서 사용자를 선택하고 사용자를 선택합니다.
7. 권한을 선택하고, Add permissions(권한 추가)를 선택합니다.
8. Attach existing policies directly(기존 정책 직접 연결)를 선택합니다. 앞에서 만든 정책을 선택하고 다음: 검토(Next: Review)를 선택합니다.
9. 권한 추가를 선택합니다.

스팟 집합의 서비스 연결 역할

Amazon EC2는 다른 AWS 서비스를 자동으로 호출하는 데 필요한 권한에 서비스 연결 역할을 사용합니다. 서비스 연결 역할은 AWS 서비스에 직접 연결된 고유한 유형의 IAM 역할입니다. 연결된 서비스만 서비스 연결 역할을 담당할 수 있으므로 서비스 연결 역할은 AWS 서비스로 권한을 위임하는 안전한 방법을 제공합니다. 자세한 내용은 IAM 사용 설명서의 [서비스 연결 역할 사용](#)을 참조하십시오.

Amazon EC2는 AWSServiceRoleForEC2Spot이라는 이름의 서비스 연결 역할을 사용하여 사용자 대신 인스턴스를 시작하고 관리합니다.

Important

스팟 집합에서 [암호화된 AMI \(p. 98\)](#) 또는 [암호화된 Amazon EBS 스냅샷 \(p. 1033\)](#)을 지정하는 경우 Amazon EC2에서 사용자를 대신하여 인스턴스를 시작할 수 있도록 AWSServiceRoleForEC2SpotFleet 역할에 CMK를 사용할 권한을 부여해야 합니다. 자세한 내용은 [암호화된 AMI 및 EBS 스냅샷과 함께 사용할 CMK에 대한 액세스 권한 부여 \(p. 276\)](#) 단원을 참조하십시오.

AWSServiceRoleForEC2SpotFleet에서 부여된 권한

Amazon EC2는 AWSServiceRoleForEC2SpotFleet을 사용하여 다음 작업을 완료합니다.

- ec2:RequestSpotInstances - 스팟 인스턴스 요청
- ec2:RunInstances - 인스턴스 시작
- ec2:TerminateInstances - 인스턴스 종료
- ec2:DescribeImages - 인스턴스에 대한 Amazon 머신 이미지(AMI) 설명
- ec2:DescribeInstanceStatus - 인스턴스 상태 설명
- ec2:DescribeSubnets - 인스턴스의 서브넷 설명
- ec2>CreateTags - 스팟 집합 요청, 인스턴스 및 볼륨에 태그 추가
- elasticloadbalancing:RegisterInstancesWithLoadBalancer - 지정된 로드 밸런서에 지정된 인스턴스 추가
- elasticloadbalancing:RegisterTargets - 지정된 대상 그룹에 지정된 대상 등록

서비스 연결 역할 생성

부분의 경우 서비스 연결 역할을 수동으로 생성할 필요가 없습니다. Amazon EC2는 처음으로 콘솔을 사용하여 스팟 집합을 생성할 때 AWSServiceRoleForEC2SpotFleet 서비스 연결 역할을 생성합니다.

AWS CLI 또는 API를 사용할 경우 이 역할이 존재하는지 확인해야 합니다.

Amazon EC2가 이 서비스 연결 역할을 지원하기 시작한 2017년 10월 이전에 활성 스팟 집합을 요청한 경우 Amazon EC2에서 사용자 AWS 계정에 AWSServiceRoleForEC2SpotFleet 역할을 이미 생성했습니다. 자세한 내용은 IAM 사용 설명서의 [내 AWS 계정에 표시되는 새 역할](#)을 참조하십시오.

AWS CLI 또는 API를 사용하여 스팟 집합을 만들려면 먼저 이 역할이 있어야 합니다. 역할을 생성하려면 다음과 같이 IAM 콘솔을 사용하십시오.

AWSServiceRoleForEC2SpotFleet 서비스 연결 역할을 수동으로 생성하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 역할을 선택합니다.
3. [Create role]을 선택합니다.
4. Select type of trusted entity(신뢰할 수 있는 엔터티 유형 선택)에서 AWS service(AWS 서비스)를 선택합니다.
5. 서비스 목록에서 EC2를 선택합니다.
6. Select your use case(사용 사례 선택) 섹션에서 EC2 - Spot Fleet(EC2 - 스팟 플릿)을 선택합니다.
7. Next: Permissions(다음: 권한)를 선택합니다.
8. 다음 페이지에서 다음:검토(Next:Review)를 선택합니다.
9. 검토 페이지에서 역할 만들기를 선택합니다.

스팟 집합이 더 이상 필요 없는 경우에는 AWSServiceRoleForEC2SpotFleet 역할을 삭제할 것을 권합니다. 계정에서 이 역할이 삭제된 후 콘솔을 사용하여 스팟 집합을 요청하면 Amazon EC2에서 다시 해당 역할을 생성합니다. 자세한 내용은 IAM 사용 설명서의 [서비스 연결 역할 삭제](#)를 참조하십시오.

암호화된 AMI 및 EBS 스냅샷과 함께 사용할 CMK에 대한 액세스 권한 부여

스팟 집합 요청에서 [암호화된 AMI \(p. 98\)](#) 또는 [암호화된 Amazon EBS 스냅샷 \(p. 1033\)](#)을 지정하고 암호화 용 고객 관리형 고객 마스터 키(CMK)를 사용하는 경우 AWSServiceRoleForEC2SpotFleet 역할에 CMK를 사용할 수 있는 권한을 부여해야 Amazon EC2가 사용자를 대신하여 인스턴스를 시작할 수 있습니다. 이렇게 하려면 다음 절차에 표시된 바와 같이 CMK에 권한을 추가해야 합니다.

권한을 제공할 때 권한 부여는 키 정책을 대체합니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [권한 부여 사용 및 AWS KMS에서 키 정책 사용](#) 단원을 참조하십시오.

CMK를 사용할 수 있도록 AWSServiceRoleForEC2SpotFleet 역할 권한을 부여하려면

- `create-grant` 명령을 사용하여 CMK에 권한을 추가하고 허용된 작업을 수행할 수 있는 권한이 부여된 보안 주체(AWSServiceRoleForEC2SpotFleet 서비스 연결 역할)를 지정합니다. CMK는 CMK의 key-id 파라미터 및 ARN에 의해 지정됩니다. 보안 주체는 AWSServiceRoleForEC2SpotFleet 서비스 연결 역할의 grantee-principal 파라미터 및 ARN에 의해 지정됩니다.

```
aws kms create-grant \
  --region us-east-1 \
  --key-id arn:aws:kms:us-
  east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
  --grantee-principal arn:aws:iam::111122223333:role/AWSServiceRoleForEC2SpotFleet \
  --operations "Decrypt" "Encrypt" "GenerateDataKey"
  "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
  "ReEncryptTo"
```

스팟 집합에 대한 IAM 역할

`aws-ec2-spot-fleet-tagging-role` IAM 역할은 스팟 집합 요청, 인스턴스 및 볼륨에 태그를 지정할 권한을 부여합니다. 자세한 내용은 [스팟 집합 태그 지정 \(p. 280\)](#) 단원을 참조하십시오.

Important

플릿에서 인스턴스에 태그를 지정하고 목표 용량을 유지하기로 선택하면(스팟 집합 요청이 `maintain` 유형) IAM 사용자와 IamFleetRole의 권한 차이로 인해 플릿에서 인스턴스의 태그 지정 동작이 일치하지 않을 수 있습니다. IamFleetRole에 CreateTags 권한이 없으면 플릿에서 시작한 일부 인스턴스에 태그가 지정되지 않을 수 있습니다. 이러한 불일치를 수정하는 동

안 플릿에서 시작한 모든 인스턴스에 태그가 지정되도록 하려면 `IamFleetRole`에 `aws-ec2-spot-fleet-tagging-role` 역할을 사용하는 것이 좋습니다. 또는 기존 역할을 사용하려면 `AmazonEC2SpotFleetTaggingRole` AWS 관리형 정책을 기존 역할에 연결합니다. 그렇지 않으면 기존 정책에 `CreateTags` 권한을 수동으로 추가해야 합니다.

IAM 태그 지정에 대한 스팟 집합 역할을 만들려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 역할을 선택합니다.
3. 신뢰할 수 있는 유형의 엔터티 선택 페이지에서 AWS 서비스, EC2, EC2 - 스팟 플릿 태그 지정, 다음: 권한을 차례로 선택합니다.
4. 연결된 권한 정책(Attached permissions policy) 페이지에서 다음: 검토(Next: Review)를 선택합니다.
5. 검토 페이지에서 역할의 이름(예: `aws-ec2-spot-fleet-tagging-role`)을 입력하고 역할 생성을 선택합니다.

스팟 집합 요청 생성

AWS Management 콘솔을 사용하면 필요한 애플리케이션 또는 작업과 최소 계산 사양을 선택해 스팟 집합 요청을 빠르게 생성할 수 있습니다. Amazon EC2에서는 필요에 가장 잘 맞는 플릿을 구성하고 스팟 모범 사례를 따릅니다. 자세한 내용은 [빠르게 스팟 집합 요청 생성\(콘솔\) \(p. 277\)](#) 단원을 참조하십시오. 그렇지 않으면 어떠한 기본 설정이든 수정할 수 있습니다. 자세한 내용은 [정의된 파라미터를 사용하여 스팟 집합 요청 생성\(콘솔\) \(p. 277\)](#) 단원을 참조하십시오.

빠르게 스팟 집합 요청 생성(콘솔)

다음 단계에 따라 빠르게 스팟 집합 요청을 생성합니다.

권장되는 설정을 사용하여 스팟 집합 요청을 생성하려면(콘솔)

1. <https://console.aws.amazon.com/ec2spot>에서 스팟 콘솔을 엽니다.
2. 스팟을 처음 사용하는 경우 시작 페이지가 표시되면 시작하기를 선택합니다. 그렇지 않다면 스팟 인스턴스 요청을 선택합니다.
3. 필요한 애플리케이션 또는 작업 알려주기에서 로드 밸런싱 워크로드, 유연한 워크로드, 빅 데이터 워크로드 또는 정의된 기간 워크로드를 선택합니다.
4. 인스턴스 구성에서 Minimum compute unit(최소 컴퓨팅 단위)에 대해 애플리케이션 또는 작업에 필요한 최소 하드웨어 사양(vCPU, 메모리, 및 스토리지)으로 as specs(사양으로) 또는 as an instance type(인스턴스 유형으로)을 선택합니다.
 - as specs(사양으로)에 대해 필요한 vCPU 수와 메모리 양을 지정합니다.
 - as an instance type(인스턴스 유형으로)에 대해 기본 인스턴스 유형을 수락하거나 Change instance type(인스턴스 유형 변경)을 선택하여 다른 인스턴스 유형을 선택합니다.
5. Tell us how much capacity you need(필요한 용량 알리기)에서 Total target capacity(총 목표 용량)에 대해 목표 용량으로 요청할 단위 수를 지정합니다. 인스턴스 또는 vCPU를 선택할 수 있습니다.
6. 선택한 애플리케이션 또는 작업을 바탕으로 권장되는 Fleet request settings(플릿 요청 설정)를 살펴보고 시작을 선택합니다.

정의된 파라미터를 사용하여 스팟 집합 요청 생성(콘솔)

정의하는 파라미터를 사용하여 스팟 집합을 생성할 수 있습니다.

정의된 파라미터를 사용하여 스팟 집합 요청을 생성하려면(콘솔)

1. <https://console.aws.amazon.com/ec2spot>에서 스팟 콘솔을 엽니다.
2. 스팟을 처음 사용하는 경우 시작 페이지가 표시되면 시작하기를 선택합니다. 그렇지 않다면 스팟 인스턴스 요청을 선택합니다.

3. 필요한 애플리케이션 또는 작업 알려주기에서 로드 밸런싱 워크로드, 유연한 워크로드, 빅 데이터 워크로드 또는 정의된 기간 워크로드를 선택합니다.
4. 인스턴스 구성에 대해 다음을 수행합니다.
 - a. (선택 사항) 시작 템플릿에서 시작 템플릿을 선택합니다. 시작 템플릿이 Amazon 머신 이미지(AMI)를 지정해야 합니다. 사용자가 시작 템플릿을 지정할 경우 스팟 집합을 사용하여 AMI를 재정의할 수 없기 때문입니다.

Important

Optional On-Demand portion(옵션 온디맨드 부분)을 지정할 경우 시작 템플릿을 선택해야 합니다.

- b. AMI에 대해서는 AWS가 제공하는 기본 AMI 중 하나를 선택하거나 AMI 검색을 선택하여 사용자 커뮤니티의 AMI, AWS Marketplace의 AMI 또는 자체 AMI를 사용합니다.
- c. Minimum compute unit(최소 컴퓨팅 단위)에 대해 애플리케이션 또는 작업에 필요한 최소 하드웨어 사양(vCPU, 메모리, 및 스토리지)으로 as specs(사양으로) 또는 as an instance type(인스턴스 유형으로)을 선택합니다.
 - as specs(사양으로)에 대해 필요한 vCPU 수와 메모리 양을 지정합니다.
 - as an instance type(인스턴스 유형으로)에 대해 기본 인스턴스 유형을 수락하거나 Change instance type(인스턴스 유형 변경)을 선택하여 다른 인스턴스 유형을 선택합니다.
- d. 네트워크에서 기존 VPC를 선택하거나 새로 생성합니다.

[기존 VPC] VPC를 선택합니다.

[새 VPC] Amazon VPC 콘솔로 이동하려면 새 VPC 생성을 선택합니다. 마친 후에 마법사로 돌아와 목록을 새로 고칩니다.

- e. (선택 사항) 가용 영역의 경우 AWS에서 스팟 인스턴스에 대한 가용 영역을 자동으로 선택하도록 하거나 가용 영역을 하나 이상 지정합니다.
- f. (선택 사항) 키 페어 이름에서 기존 키 페어를 선택하거나 새로 생성합니다.

[기존 키 페어] 키 페어를 선택합니다.

[새 키 페어] 새 키 페어 생성을 선택해 Amazon VPC 콘솔로 이동합니다. 마친 후에 마법사로 돌아와 목록을 새로 고칩니다.

5. (선택 사항) 추가 구성에서 다음을 수행합니다.
 - a. (선택 사항) 스토리지를 추가하려면 인스턴스 유형에 따라 추가 인스턴스 스토어 볼륨이나 Amazon EBS 볼륨을 지정합니다.
 - b. (선택 사항) Amazon EBS 최적화를 활성화하려면 EBS-optimized(EBS 최적화)에서 EBS 최적 인스턴스 시작을 선택합니다.
 - c. (선택 사항) 인스턴스에 대해 임시 볼록 수준 스토리지를 추가하려면 인스턴스 스토어에 대해 시작 시 연결을 선택합니다.
 - d. (선택 사항) 기본적으로 인스턴스에 대해 기본 모니터링 기능이 활성화됩니다. 세부 모니터링을 활성화하려면 모니터링에 대해 CloudWatch 세부 모니터링 활성화를 선택합니다.
 - e. (선택 사항) 비정상 인스턴스를 대체하려면 상태 검사(Health check)에서 비정상 인스턴스 교체(Replace unhealthy instances)를 선택합니다. 이 옵션을 활성화하려면 먼저 대상 용량 유지를 선택해야 합니다.
 - f. (선택 사항) 전용 스팟 인스턴스를 실행하려면 테넌시에 대해 전용 - 전용 인스턴스로 실행을 선택합니다.
- g. (선택 사항) 보안 그룹에 대해 하나 이상의 보안 그룹을 선택하거나 보안 그룹을 생성합니다.

[기준 보안 그룹] 하나 이상의 보안 그룹을 선택합니다.

[새 보안 그룹] 새 보안 그룹 생성을 선택하여 Amazon VPC 콘솔로 이동합니다. 마친 후에 마법사로 돌아와 목록을 새로 고칩니다.

- h. (선택 사항) 인터넷에서 인스턴스에 연결할 수 있도록 하려면 IPv4 퍼블릭 IP 자동 할당에 대해 활성화를 선택합니다.
- i. (선택 사항) IAM 역할로 스팟 인스턴스를 시작하려면 IAM 인스턴스 프로파일에서 역할을 선택합니다.
- j. (선택 사항) 시작 스크립트를 실행하려면 해당 스크립트를 사용자 데이터에 복사합니다.
- k. (선택 사항) 태그를 추가하려면 새로운 태그 추가를 선택하고 해당 태그에 대한 키와 값을 입력합니다. 각 태그에 대해 반복합니다.

각 태그에서 인스턴스와 스팟 집합 요청에 같은 태그를 사용하여 태그를 지정하려면 Instance tags(인스턴스 태그)와 Fleet tags(플릿 태그)가 모두 선택되어 있는지 확인합니다. 플릿에서 시작한 인스턴스에만 태그를 지정하려면 Fleet tags(플릿 태그) 선택을 취소합니다. 스팟 집합 요청에만 태그를 지정하려면 인스턴스 태그의 선택을 취소합니다.

6. Tell us how much capacity you need(필요한 용량 알리기)에 대해 다음을 수행합니다.

- a. Total target capacity(총 목표 용량)에 목표 용량으로 요청할 단위 수를 지정합니다. 인스턴스 또는 vCPU를 선택할 수 있습니다. 나중에 용량을 추가할 수 있도록 목표 용량을 0으로 지정하려면 Maintain target capacity(목표 용량 유지)를 선택합니다.
- b. (선택 사항) Optional On-Demand portion(옵션 온디맨드 부분)에 요청할 온디맨드 단위 수를 지정합니다. 이 값은 Total target capacity(총 목표 용량)보다 작아야 합니다. Amazon EC2가 차이를 계산하고 스팟 단위에 요청할 차이를 할당합니다.

Important

옵션 온디맨드 부분을 지정하려면 먼저, 시작 템플릿을 선택해야 합니다.

- c. (선택 사항) 스팟 서비스는 중단된 스팟 인스턴스를 종료하도록 기본 설정되어 있습니다. 목표 용량을 유지하려면 Maintain target capacity(목표 용량 유지)를 선택합니다. 그런 다음 스팟 서비스가 중단된 스팟 인스턴스를 종료, 중지 또는 최대 절전 모드로 전환하도록 지정할 수 있습니다. 이를 위해 인터럽트 방식에서 해당 옵션을 선택합니다.

7. Fleet request settings(플릿 요청 설정)에서 다음을 수행합니다.

- a. 선택한 애플리케이션 또는 작업을 바탕으로 플릿 요청 및 플릿 할당 전략을 살펴봅니다. 인스턴스 유형 또는 할당 전략을 변경하려면 Apply recommendations(권장 사항 적용)를 선택 취소합니다.
- b. (선택 사항) 인스턴스 유형을 제거하려면 Fleet request(플릿 요청)에 대해 제거를 선택합니다. 인스턴스 유형을 추가하려면 인스턴스 유형 선택을 선택합니다.
- c. (선택 사항) Fleet allocation strategy(플릿 할당 전략)으로 필요에 맞는 전략을 선택합니다. 자세한 내용은 [스팟 인스턴스의 할당 전략 \(p. 246\)](#) 단원을 참조하십시오.

8. (선택 사항) Additional request details(추가 요청 세부 정보)에서 다음을 수행합니다.

- a. 추가 요청 세부 정보를 검토합니다. 변경하려면 Apply defaults(기본값 적용)의 선택을 취소합니다.
- b. (선택 사항) IAM fleet role(IAM 플릿 역할)에서 기본 역할을 사용하거나 다른 역할을 선택할 수 있습니다. 역할을 변경한 후 기본 역할을 사용하려면 기본 역할 사용을 선택합니다.
- c. (선택 사항) 최고 가격에서는 기본 최고 가격(온디맨드 가격)을 사용하거나 지불하고자 하는 최고 가격을 지정합니다. 최고 가격이 선택한 인스턴스 유형에 대한 스팟 가격보다 낮으면 스팟 인스턴스가 시작되지 않습니다.
- d. (선택 사항) 특정 기간 동안만 유효한 요청을 생성하려면 요청 유효 시작 시간 및 요청 유효 종료 시간(Request valid until)을 편집합니다.
- e. (선택 사항) 기본적으로 요청 만료 시 스팟 인스턴스를 종료합니다. 요청 만료 후에도 계속 실행하려면 Terminate the instances when the request expires(요청 만료 시 인스턴스 종료)를 선택 취소합니다.

- f. (선택 사항) 로드 밸런서에 스팟 인스턴스를 등록하려면 하나 이상의 로드 밸런서에서 트래픽 수신을 선택하고 하나 이상의 Classic Load Balancer나 대상 그룹을 선택합니다.
9. (선택 사항) AWS CLI용 시작 구성의 복사본을 다운로드하려면 JSON 구성을 선택합니다.
10. 시작을 선택합니다.

스팟 집합 요청 형식은 `fleet`입니다. 요청이 이행되면 `instance` 유형의 요청이 추가되며, 그 상태는 `active`이고 상황은 `fulfilled`입니다.

AWS CLI를 사용하여 스팟 집합 요청을 생성하려면

- `request-spot-fleet` 명령을 사용하여 스팟 집합 요청을 생성합니다.

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

구성 파일에 대한 예시는 [스팟 집합 구성의 예 \(p. 289\)](#) 단원을 참조하십시오.

다음은 예제 출력입니다.

```
{  
    "SpotFleetRequestId": "sfr-73fbcd2ce-aa30-494c-8788-1cee4EXAMPLE"  
}
```

스팟 집합 태그 지정

스팟 집합 요청을 쉽게 분류하고 관리할 수 있도록 사용자 지정 메타데이터로 이 요청에 태그를 지정할 수 있습니다. 스팟 집합 요청을 만들 때 또는 만든 후 요청에 태그를 지정할 수 있습니다. Amazon EC2 콘솔이나 명령줄 도구를 사용하여 태그를 지정할 수 있습니다.

스팟 집합 요청에 태그를 지정하면 스팟 집합에서 시작한 인스턴스 및 볼륨은 자동으로 태그가 지정되지 않습니다. 스팟 집합에서 시작한 인스턴스 및 볼륨에 명시적으로 태그를 지정해야 합니다. 스팟 집합 요청에만, 플랫에서 시작한 인스턴스에만, 플랫에서 시작한 인스턴스에 연결된 볼륨에만, 또는 세 가지 모두에 태그를 할당하도록 선택할 수 있습니다.

Note

볼륨 태그는 온디맨드 인스턴스에 연결된 볼륨에 대해서만 지원됩니다. 스팟 인스턴스에 연결된 볼륨에는 태그를 지정할 수 없습니다.

태그 작동 방식에 대한 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

목차

- [사전 조건 \(p. 280\)](#)
- [새 스팟 집합에 태그 지정 \(p. 281\)](#)
- [새 스팟 집합 및 해당 플랫이 시작하는 인스턴스 및 볼륨에 태그 지정 \(p. 282\)](#)
- [기존 스팟 집합에 태그 지정 \(p. 284\)](#)
- [스팟 집합 요청 태그 보기 \(p. 285\)](#)

사전 조건

IAM 사용자에게 리소스에 태그를 지정할 수 있는 권한을 부여합니다. 자세한 내용은 [예제: 리소스에 태그 지정 \(p. 874\)](#) 단원을 참조하십시오.

IAM 사용자에게 리소스에 태그를 지정할 수 있는 권한을 부여하려면

다음을 포함하는 IAM 정책을 만듭니다.

- `ec2:CreateTags` 작업. IAM 사용자에게 태그 생성 권한이 부여됩니다.
- `ec2:RequestSpotFleet` 작업. IAM 사용자에게 스팟 집합 요청 생성 권한이 부여됩니다.
- `Resource`의 경우 "*"를 지정해야 합니다. 이를 통해 사용자는 모든 리소스 유형에 태그를 지정할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "TagSpotFleetRequest",  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateTags",  
                "ec2:RequestSpotFleet"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

Important

`spot-fleet-request` 리소스에 대한 리소스 수준 권한은 현재 지원되지 않습니다. `spot-fleet-request`를 리소스로 지정하면 플릿에 태그를 지정하려고 할 때 승인되지 않은 예외가 발생합니다. 다음 예에서는 정책을 설정하지 않는 방법을 보여 줍니다.

```
{  
    "Effect": "Allow",  
    "Action": [  
        "ec2:CreateTags",  
        "ec2:RequestSpotFleet"  
    ],  
    "Resource": "arn:aws:ec2:us-east-1:111122223333:spot-fleet-request/*"  
}
```

새 스팟 집합에 태그 지정

콘솔을 사용하여 새 스팟 집합 요청에 태그를 지정하려면

1. [정의된 파라미터를 사용하여 스팟 집합 요청 생성\(콘솔\) \(p. 277\)](#)의 절차를 따르십시오.
2. 태그를 추가하려면 추가 구성을 확장하고 새로운 태그 추가를 선택한 다음 태그의 키와 값을 입력합니다. 각 태그에 대해 반복합니다.

각 태그에 대해 동일한 태그로 스팟 집합 요청과 인스턴스에 태그를 지정할 수 있습니다. 인스턴스와 요청에 모두 태그를 지정하려면 Instance tags(인스턴스 태그)와 Fleet tags(플릿 태그)가 모두 선택되어 있는지 확인합니다. 스팟 집합 요청에만 태그를 지정하려면 인스턴스 태그의 선택을 취소합니다. 플릿에서 시작한 인스턴스에만 태그를 지정하려면 Fleet tags(플릿 태그) 선택을 취소합니다.

3. 필수 필드를 입력하여 스팟 집합 요청을 생성한 다음 시작을 선택합니다. 자세한 내용은 [정의된 파라미터를 사용하여 스팟 집합 요청 생성\(콘솔\) \(p. 277\)](#) 단원을 참조하십시오.

AWS CLI를 사용하여 새 스팟 집합 요청에 태그를 지정하려면

스팟 집합 요청을 생성할 때 태그를 지정하려면 다음과 같이 스팟 집합 요청 구성을 구성합니다.

- `SpotFleetRequestConfig`에서 스팟 집합 요청에 대한 태그를 지정합니다.

- ResourceType에 대해 spot-fleet-request를 지정합니다. 다른 값을 지정하면 플릿 요청이 실패합니다.
- Tags에 대해 키-값 페어를 지정합니다. 둘 이상의 키-값 페어를 지정할 수 있습니다.

다음 예에서 스팟 집합 요청에는 Key=Environment 및 Value=Production, Key=Cost-Center 및 Value=123 두 개의 태그가 지정됩니다.

```
{  
    "SpotFleetRequestConfig": {  
        "AllocationStrategy": "lowestPrice",  
        "ExcessCapacityTerminationPolicy": "default",  
        "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-role",  
        "LaunchSpecifications": [  
            {  
                "ImageId": "ami-0123456789EXAMPLE",  
                "InstanceType": "c4.large"  
            }  
        ],  
        "SpotPrice": "5",  
        "TargetCapacity": 2,  
        "TerminateInstancesWithExpiration": true,  
        "Type": "maintain",  
        "ReplaceUnhealthyInstances": true,  
        "InstanceInterruptionBehavior": "terminate",  
        "InstancePoolsToUseCount": 1,  
        "TagSpecifications": [  
            {  
                "ResourceType": "spot-fleet-request",  
                "Tags": [  
                    {  
                        "Key": "Environment",  
                        "Value": "Production"  
                    },  
                    {  
                        "Key": "Cost-Center",  
                        "Value": "123"  
                    }  
                ]  
            }  
        ]  
    }  
}
```

새 스팟 집합 및 해당 플릿이 시작하는 인스턴스 및 볼륨에 태그 지정

AWS CLI를 사용하여 새 스팟 집합 요청 및 해당 플릿이 시작하는 인스턴스 및 볼륨에 태그를 지정하려면 스팟 집합 요청을 생성할 때 태그를 지정하고 해당 플릿에 의해 시작할 때 인스턴스 및 볼륨에 태그를 지정하려면 스팟 집합 요청 구성을 다음과 같이 구성합니다.

스팟 집합 요청 태그:

- SpotFleetRequestConfig에서 스팟 집합 요청에 대한 태그를 지정합니다.
- ResourceType에 대해 spot-fleet-request를 지정합니다. 다른 값을 지정하면 플릿 요청이 실패합니다.
- Tags에 대해 키-값 페어를 지정합니다. 둘 이상의 키-값 페어를 지정할 수 있습니다.

인스턴스 태그:

- LaunchSpecifications의 인스턴스에 대한 태그를 지정합니다.

- ResourceType에 대해 instance를 지정합니다. 다른 값을 지정하면 플릿 요청이 실패합니다.
- Tags에 대해 키-값 페어를 지정합니다. 둘 이상의 키-값 페어를 지정할 수 있습니다.

또는 스팟 집합 요청에서 참조되는 [시작 템플릿 \(p. 383\)](#)에서 인스턴스에 대한 태그를 지정할 수 있습니다.

볼륨 태그:

- 스팟 집합 요청에서 참조되는 [시작 템플릿 \(p. 383\)](#)의 볼륨에 대한 태그를 지정합니다.
LaunchSpecifications에서의 볼륨 태그 지정은 지원되지 않습니다.

다음 예에서 스팟 집합 요청에는 Key=Environment 및 Value=Production, Key=Cost-Center 및 Value=123 두 개의 태그가 지정됩니다. 플릿에서 시작한 인스턴스에는 스팟 집합 요청의 태그 중 하나와 동일한 Key=Cost-Center 및 Value=123 태그 하나만 지정됩니다.

```
{  
    "SpotFleetRequestConfig": {  
        "AllocationStrategy": "lowestPrice",  
        "ExcessCapacityTerminationPolicy": "default",  
        "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-role",  
        "LaunchSpecifications": [  
            {  
                "ImageId": "ami-0123456789EXAMPLE",  
                "InstanceType": "c4.large",  
                "TagSpecifications": [  
                    {  
                        "ResourceType": "instance",  
                        "Tags": [  
                            {  
                                "Key": "Cost-Center",  
                                "Value": "123"  
                            }  
                        ]  
                    }  
                ]  
            }  
        ],  
        "SpotPrice": "5",  
        "TargetCapacity": 2,  
        "TerminateInstancesWithExpiration": true,  
        "Type": "maintain",  
        "ReplaceUnhealthyInstances": true,  
        "InstanceInterruptionBehavior": "terminate",  
        "InstancePoolsToUseCount": 1,  
        "TagSpecifications": [  
            {  
                "ResourceType": "spot-fleet-request",  
                "Tags": [  
                    {  
                        "Key": "Environment",  
                        "Value": "Production"  
                    },  
                    {  
                        "Key": "Cost-Center",  
                        "Value": "123"  
                    }  
                ]  
            }  
        ]  
    }  
}
```

AWS CLI를 사용하여 스팟 집합에서 시작한 인스턴스에 태그를 지정하려면

플릿에서 시작될 때 인스턴스에 태그를 지정하려면 스팟 집합 요청에서 참조되는 [시작 템플릿 \(p. 383\)](#)에서 태그를 지정하거나 다음과 같이 스팟 집합 요청 구성에서 태그를 지정할 수 있습니다.

- `LaunchSpecifications`의 인스턴스에 대한 태그를 지정합니다.
- `ResourceType`에 대해 `instance`를 지정합니다. 다른 값을 지정하면 플릿 요청이 실패합니다.
- `Tags`에 대해 키-값 페어를 지정합니다. 둘 이상의 키-값 페어를 지정할 수 있습니다.

다음 예에서는 플릿에 의해 시작되는 인스턴스에 `Key=Cost-Center` 및 `Value=123` 태그가 지정되어 있습니다.

```
{  
    "SpotFleetRequestConfig": {  
        "AllocationStrategy": "lowestPrice",  
        "ExcessCapacityTerminationPolicy": "default",  
        "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-role",  
        "LaunchSpecifications": [  
            {  
                "ImageId": "ami-0123456789EXAMPLE",  
                "InstanceType": "c4.large",  
                "TagSpecifications": [  
                    {  
                        "ResourceType": "instance",  
                        "Tags": [  
                            {  
                                "Key": "Cost-Center",  
                                "Value": "123"  
                            }  
                        ]  
                    }  
                ]  
            }  
        ],  
        "SpotPrice": "5",  
        "TargetCapacity": 2,  
        "TerminateInstancesWithExpiration": true,  
        "Type": "maintain",  
        "ReplaceUnhealthyInstances": true,  
        "InstanceInterruptionBehavior": "terminate",  
        "InstancePoolsToUseCount": 1  
    }  
}
```

AWS CLI를 사용하여 스팟 집합이 시작한 온디맨드 인스턴스에 연결된 볼륨에 태그를 지정하려면

플릿이 생성하는 볼륨에 태그를 지정하려면 스팟 집합 요청에서 참조되는 [시작 템플릿 \(p. 383\)](#)에서 태그를 지정하십시오.

Note

볼륨 태그는 온디맨드 인스턴스에 연결된 볼륨에 대해서만 지원됩니다. 스팟 인스턴스에 연결된 볼륨에는 태그를 지정할 수 없습니다.

`LaunchSpecifications`에서의 볼륨 태그 지정은 지원되지 않습니다.

기존 스팟 집합에 태그 지정

콘솔을 사용하여 기존 스팟 집합 요청에 태그를 지정하려면

스팟 집합 요청을 생성한 후 콘솔을 사용하여 플릿 요청에 태그를 추가할 수 있습니다.

1. <https://console.aws.amazon.com/ec2spot>에서 스팟 콘솔을 엽니다.
2. 스팟 집합 요청을 선택합니다.
3. 태그 탭을 선택하고 태그 생성을 선택합니다.

AWS CLI를 사용하여 기존 스팟 집합 요청에 태그를 지정하려면

`create-tags` 명령을 사용해 기존 리소스에 태그를 지정할 수 있습니다. 다음 예에서, 기존 스팟 집합 요청은 Key=purpose 및 Value=test로 태그가 지정되어 있습니다.

```
aws ec2 create-tags \
--resources sfr-11112222-3333-4444-5555-66666EXAMPLE \
--tags Key=purpose,Value=test
```

스팟 집합 요청 태그 보기

콘솔을 사용하여 스팟 집합 요청 태그를 보려면

1. <https://console.aws.amazon.com/ec2spot>에서 스팟 콘솔을 엽니다.
2. 스팟 집합 요청을 선택하고 태그 탭을 선택합니다.

스팟 집합 요청 태그를 설명하려면

`describe-tags` 명령을 사용하여 지정된 리소스에 대한 태그를 표시합니다. 다음 예제에서는 지정된 스팟 집합 요청에 대한 태그를 설명합니다.

```
aws ec2 describe-tags \
--filters "Name=resource-id,Values=sfr-11112222-3333-4444-5555-66666EXAMPLE"
```

```
{
  "Tags": [
    {
      "Key": "Environment",
      "ResourceId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
      "ResourceType": "spot-fleet-request",
      "Value": "Production"
    },
    {
      "Key": "Another key",
      "ResourceId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
      "ResourceType": "spot-fleet-request",
      "Value": "Another value"
    }
  ]
}
```

스팟 집합 요청을 설명하여 스팟 집합 요청의 태그를 볼 수도 있습니다.

`describe-spot-fleet-requests` 명령을 사용하여 지정된 스팟 집합 요청의 구성과 태그를 볼 수 있습니다. 여기에는 플릿 요청에 대해 지정된 태그가 모두 포함되어 있습니다.

```
aws ec2 describe-spot-fleet-requests \
--spot-fleet-request-ids sfr-11112222-3333-4444-5555-66666EXAMPLE
```

```
{
  "SpotFleetRequestConfigs": [
```

```
{  
    "ActivityStatus": "fulfilled",  
    "CreateTime": "2020-02-13T02:49:19.709Z",  
    "SpotFleetRequestConfig": {  
        "AllocationStrategy": "capacityOptimized",  
        "OnDemandAllocationStrategy": "lowestPrice",  
        "ExcessCapacityTerminationPolicy": "Default",  
        "FulfilledCapacity": 2.0,  
        "OnDemandFulfilledCapacity": 0.0,  
        "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-role",  
        "LaunchSpecifications": [  
            {  
                "ImageId": "ami-0123456789EXAMPLE",  
                "InstanceType": "c4.large"  
            }  
        ],  
        "TargetCapacity": 2,  
        "OnDemandTargetCapacity": 0,  
        "Type": "maintain",  
        "ReplaceUnhealthyInstances": false,  
        "InstanceInterruptionBehavior": "terminate"  
    },  
    "SpotFleetRequestId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",  
    "SpotFleetRequestState": "active",  
    "Tags": [  
        {  
            "Key": "Environment",  
            "Value": "Production"  
        },  
        {  
            "Key": "Another key",  
            "Value": "Another value"  
        }  
    ]  
}
```

스팟 집합 모니터링

스팟 집합은 최고 가격이 스팟 가격을 초과하고 용량이 가용 상태일 때 스팟 인스턴스를 시작합니다. 스팟 인스턴스는 중단되거나 사용자가 직접 종료할 때까지 실행됩니다.

스팟 집합을 모니터링하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다.
3. 스팟 집합 요청을 선택합니다. 구성 세부 정보를 보려면 설명을 선택합니다.
4. 스팟 집합에 대한 스팟 인스턴스를 나열하려면 인스턴스를 선택합니다.
5. 스팟 집합에 대한 기록을 보려면 기록을 선택합니다.

스팟 집합을 모니터링하려면(AWS CLI)

`describe-spot-fleet-requests` 명령을 사용하여 스팟 집합 요청을 설명합니다.

```
aws ec2 describe-spot-fleet-requests
```

`describe-spot-fleet-instances` 명령을 사용하여 지정한 스팟 집합에 대한 스팟 인스턴스를 설명합니다.

```
aws ec2 describe-spot-fleet-instances \
--spot-fleet-request-id sfr-73fb2ce-aa30-494c-8788-1cee4EXAMPLE
```

[describe-spot-fleet-request-history](#) 명령을 사용하여 지정한 스팟 집합 요청에 대한 기록을 설명합니다.

```
aws ec2 describe-spot-fleet-request-history \
--spot-fleet-request-id sfr-73fb2ce-aa30-494c-8788-1cee4EXAMPLE \
--start-time 2015-05-18T00:00:00Z
```

스팟 집합 요청 수정

다음 작업을 완료하기 위해 활성 스팟 집합 요청을 수정할 수 있습니다.

- 목표 용량 및 온디맨드 부분을 늘립니다.
- 목표 용량 및 온디맨드 부분을 줄입니다.

Note

일회용 스팟 집합 요청은 수정할 수 없습니다. 스팟 집합 요청을 만들 때 대상 용량 유지를 선택한 경우에만 스팟 집합 요청을 수정할 수 있습니다.

목표 용량을 늘리면 스팟 집합에서 추가 스팟 인스턴스를 시작합니다. 온디맨드 부분을 늘리면 스팟 집합에서 추가 온디맨드 인스턴스를 시작합니다.

목표 용량을 늘리면 스팟 집합이 스팟 집합 요청에 대한 할당 전략에 따라 추가 스팟 인스턴스를 시작합니다. 할당 전략이 `lowestPrice`이면 스팟 집합이 스팟 집합 요청에 있는 최저 가격의 스팟 인스턴스 풀에서 인스턴스를 시작합니다. 할당 전략이 `diversified`인 경우 스팟 집합은 스팟 집합 요청의 풀 전체에 인스턴스를 배포합니다.

목표 용량을 줄이면 스팟 집합이 새 목표 용량을 초과하는 모든 열린 요청을 취소합니다. 플릿의 크기가 새 목표 용량에 도달할 때까지 스팟 집합이 스팟 인스턴스를 종료하도록 요청할 수 있습니다. 할당 전략이 `lowestPrice`인 경우 스팟 집합은 단위당 최고 가격이 있는 인스턴스를 종료합니다. 할당 전략이 `diversified`인 경우 스팟 집합은 풀 전체의 인스턴스를 종료합니다. 또는 스팟 집합이 플릿을 현재 크기로 유지하도록 요청할 수 있습니다. 그러나 중단되거나 수동으로 종료한 스팟 인스턴스는 교체하지 마십시오.

목표 용량이 줄어 스팟 집합이 인스턴스를 종료하면 해당 인스턴스는 스팟 인스턴스 중단 공지를 받습니다.

스팟 집합 요청을 수정하려면(콘솔)

- <https://console.aws.amazon.com/ec2spot/home/fleet>에서 스팟 콘솔을 엽니다.
- 스팟 집합 요청을 선택합니다.
- 작업을 선택한 다음, `Modify target capacity`(목표 용량 수정)을 선택합니다.
- 목표 용량 수정(Modify target capacity)에서 다음 작업을 수행하십시오.
 - 새 목표 용량 및 온디맨드 부분을 입력합니다.
 - (선택 사양) 목표 용량을 줄이지만 집합은 현재 크기로 유지하고자 한다면, 인스턴스 종료 선택을 취소합니다.
 - 제출을 선택합니다.

AWS CLI를 사용하여 스팟 집합 요청을 수정하려면

[modify-spot-fleet-request](#) 명령을 사용하여 지정된 스팟 집합 요청의 목표 용량을 업데이트합니다.

```
aws ec2 modify-spot-fleet-request \
```

```
--spot-fleet-request-id sfr-73fb2ce-aa30-494c-8788-1cee4EXAMPLE \
--target-capacity 20
```

다음과 같이 이전 명령을 수정하여 결과적으로 어떤 스팟 인스턴스도 종료하지 않고 지정된 스팟 집합의 목표 용량을 출일 수 있습니다.

```
aws ec2 modify-spot-fleet-request \
--spot-fleet-request-id sfr-73fb2ce-aa30-494c-8788-1cee4EXAMPLE \
--target-capacity 10 \
--excess-capacity-termination-policy NoTermination
```

스팟 집합 요청 취소

스팟 집합 사용을 마쳤으면 스팟 집합 요청을 취소할 수 있습니다. 이렇게 하면 스팟 집합과 연결된 스팟 요청이 모두 취소되므로 스팟 집합에 대해 시작되는 새 스팟 인스턴스가 없습니다. 스팟 집합이 스팟 인스턴스를 종료할지 여부를 지정해야 합니다. 인스턴스를 종료하면 스팟 집합 요청은 `cancelled_terminating` 상태가 됩니다. 인스턴스를 종료하지 않으면, 스팟 플릿 요청은 `cancelled_running` 상태가 되고 인스턴스는 중단되거나 사용자가 수동으로 종료하지 않는 한 계속 실행됩니다.

스팟 집합 요청을 취소하려면(콘솔)

1. <https://console.aws.amazon.com/ec2spot/home/fleet>에서 스팟 콘솔을 엽니다.
2. 스팟 집합 요청을 선택합니다.
3. 작업을 클릭하고 Cancel spot request(스팟 요청 취소)를 선택합니다.
4. 스팟 요청 취소(Cancel spot request)에서 스팟 집합을 취소하겠다는 것을 확인합니다. 집합을 현재 크기로 유지하려면 인스턴스 종료를 선택 취소합니다. 준비가 완료되면 확인을 선택합니다.

AWS CLI를 사용하여 스팟 집합 요청을 취소하려면

`cancel-spot-fleet-requests` 명령을 사용하여 지정한 스팟 집합 요청을 취소하고 인스턴스를 종료합니다.

```
aws ec2 cancel-spot-fleet-requests \
--spot-fleet-request-ids sfr-73fb2ce-aa30-494c-8788-1cee4EXAMPLE \
--terminate-instances
```

다음은 예제 출력입니다.

```
{
  "SuccessfulFleetRequests": [
    {
      "SpotFleetRequestId": "sfr-73fb2ce-aa30-494c-8788-1cee4EXAMPLE",
      "CurrentSpotFleetRequestState": "cancelled_terminating",
      "PreviousSpotFleetRequestState": "active"
    }
  ],
  "UnsuccessfulFleetRequests": []
}
```

다음과 같이 이전 명령을 수정하여 인스턴스를 종료하지 않고 지정된 스팟 집합 요청을 취소할 수 있습니다.

```
aws ec2 cancel-spot-fleet-requests \
--spot-fleet-request-ids sfr-73fb2ce-aa30-494c-8788-1cee4EXAMPLE \
--no-terminate-instances
```

다음은 예제 출력입니다.

```
{  
    "SuccessfulFleetRequests": [  
        {  
            "SpotFleetRequestId": "sfr-73fbcd2ce-aa30-494c-8788-1cee4EXAMPLE",  
            "CurrentSpotFleetRequestState": "cancelled_running",  
            "PreviousSpotFleetRequestState": "active"  
        }  
    ],  
    "UnsuccessfulFleetRequests": []  
}
```

스팟 집합 구성의 예

다음 예에서는 [request-spot-fleet](#) 명령과 함께 사용하여 스팟 집합 요청을 생성할 수 있는 시작 구성을 보여 줍니다. 자세한 내용은 [스팟 집합 요청 생성 \(p. 277\)](#) 단원을 참조하십시오.

Note

스팟 집합의 경우 시작 사양에 네트워크 인터페이스 ID를 지정할 수 없습니다. 시작 사양에서 `NetworkInterfaceID` 파라미터를 생략하십시오.

1. 리전에서 최저 가격의 가용 영역 또는 서브넷을 사용하여 스팟 인스턴스 시작 (p. 289)
2. 지정된 목록에서 최저 가격의 가용 영역 또는 서브넷을 사용하여 스팟 인스턴스 시작 (p. 289)
3. 지정된 목록에서 최저 가격의 인스턴스 유형을 사용하여 스팟 인스턴스 시작 (p. 291)
4. 요청에 대한 가격 재정의 (p. 292)
5. 다각화된 할당 전략을 사용하여 스팟 집합 시작 (p. 293)
6. 인스턴스 가중치를 사용하여 스팟 집합 시작 (p. 295)
7. 온디맨드 용량으로 스팟 집합 시작 (p. 296)

예 1: 리전에서 최저 가격의 가용 영역 또는 서브넷을 사용하여 스팟 인스턴스 시작

다음 예는 가용 영역이나 서브넷이 없는 단일 시작 사양을 지정합니다. 스팟 집합은 기본 서브넷을 보유한 최저 가격의 가용 영역에 있는 인스턴스를 시작합니다. 지불하는 가격이 온디맨드 가격을 초과하지 않습니다.

```
{  
    "TargetCapacity": 20,  
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",  
    "LaunchSpecifications": [  
        {  

```

예 2: 지정된 목록에서 최저 가격의 가용 영역 또는 서브넷을 사용하여 스팟 인스턴스 시작

다음 예는 가용 영역이나 서브넷은 다르지만 인스턴스 유형과 AMI는 같은 두 개의 시작 사양을 지정합니다.

가용 영역

스팟 집합은 지정한 최저 가격의 가용 영역에 있는 기본 서브넷에서 인스턴스를 시작합니다.

```
{  
    "TargetCapacity": 20,  
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",  
    "LaunchSpecifications": [  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "KeyName": "my-key-pair",  
            "SecurityGroups": [  
                {  
                    "GroupId": "sg-1a2b3c4d"  
                }  
            ],  
            "InstanceType": "m3.medium",  
            "Placement": {  
                "AvailabilityZone": "us-west-2a, us-west-2b"  
            },  
            "IamInstanceProfile": {  
                "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"  
            }  
        }  
    ]  
}
```

서브넷

기본 서브넷이나 기본이 아닌 서브넷을 지정할 수 있으며, 기본이 아닌 서브넷은 기본 VPC 또는 기본이 아닌 VPC의 서브넷일 수 있습니다. 스팟 서비스는 최저 가격의 가용 영역에 있는 서브넷에서 인스턴스를 시작합니다.

스팟 집합 요청에서 동일한 가용 영역의 서로 다른 서브넷을 지정할 수 없습니다.

```
{  
    "TargetCapacity": 20,  
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",  
    "LaunchSpecifications": [  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "KeyName": "my-key-pair",  
            "SecurityGroups": [  
                {  
                    "GroupId": "sg-1a2b3c4d"  
                }  
            ],  
            "InstanceType": "m3.medium",  
            "SubnetId": "subnet-a61dafcf, subnet-65ea5f08",  
            "IamInstanceProfile": {  
                "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"  
            }  
        }  
    ]  
}
```

인스턴스가 기본 VPC로 시작되는 경우, 인스턴스는 기본적으로 퍼블릭 IPv4 주소를 받습니다. 인스턴스가 기본이 아닌 VPC로 시작되는 경우, 인스턴스는 기본적으로 퍼블릭 IPv4 주소를 받지 않습니다. 시작 사양에서 네트워크 인터페이스를 사용하여 기본이 아닌 VPC에서 시작되는 인스턴스에 퍼블릭 IPv4 주소를 할당하십시오. 네트워크 인터페이스를 지정할 때는 네트워크 인터페이스를 사용해 서브넷 ID 및 보안 그룹 ID를 반드시 포함시켜야 합니다.

```
...
{
    "ImageId": "ami-1a2b3c4d",
    "KeyName": "my-key-pair",
    "InstanceType": "m3.medium",
    "NetworkInterfaces": [
        {
            "DeviceIndex": 0,
            "SubnetId": "subnet-1a2b3c4d",
            "Groups": [ "sg-1a2b3c4d" ],
            "AssociatePublicIpAddress": true
        }
    ],
    "IamInstanceProfile": {
        "Arn": "arn:aws:iam::880185128111:instance-profile/my-iam-role"
    }
}
...
```

예 3: 지정된 목록에서 최저 가격의 인스턴스 유형을 사용하여 스팟 인스턴스 시작

다음 예는 인스턴스 유형은 다르지만 AMI와 가용 영역 또는 서브넷은 같은 두 개의 시작 구성이 지정합니다. 스팟 집합이 최저 가격으로 지정된 인스턴스 유형을 사용하여 인스턴스를 시작합니다.

가용 영역

```
{
    "TargetCapacity": 20,
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
    "LaunchSpecifications": [
        {
            "ImageId": "ami-1a2b3c4d",
            "SecurityGroups": [
                {
                    "GroupId": "sg-1a2b3c4d"
                }
            ],
            "InstanceType": "cc2.8xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2b"
            }
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "SecurityGroups": [
                {
                    "GroupId": "sg-1a2b3c4d"
                }
            ],
            "InstanceType": "r3.8xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2b"
            }
        }
    ]
}
```

서브넷

```
{
    "TargetCapacity": 20,
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
```

```
"LaunchSpecifications": [
    {
        "ImageId": "ami-1a2b3c4d",
        "SecurityGroups": [
            {
                "GroupId": "sg-1a2b3c4d"
            }
        ],
        "InstanceType": "cc2.8xlarge",
        "SubnetId": "subnet-1a2b3c4d"
    },
    {
        "ImageId": "ami-1a2b3c4d",
        "SecurityGroups": [
            {
                "GroupId": "sg-1a2b3c4d"
            }
        ],
        "InstanceType": "r3.8xlarge",
        "SubnetId": "subnet-1a2b3c4d"
    }
]
```

예 4. 요청에 대한 가격 재정의

기본 최고 가격인 온디맨드 가격을 사용하는 것이 좋습니다. 원할 경우 플릿 요청에 대한 최고 가격과 개별 시작 사양에 대한 최고 가격을 지정할 수 있습니다.

다음 예제에서는 플릿 요청에 대한 최고 가격과 세 가지 시작 사양 중 두 개에 대한 최고 가격을 지정합니다. 플릿 요청 대한 최고 가격은 최고 가격을 지정하지 않은 모든 시작 사양에 사용됩니다. 스팟 집합이 최저 가격으로 인스턴스 유형을 사용하여 인스턴스를 시작합니다.

가용 영역

```
{
    "SpotPrice": "1.00",
    "TargetCapacity": 30,
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
    "LaunchSpecifications": [
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c3.2xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2b"
            },
            "SpotPrice": "0.10"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c3.4xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2b"
            },
            "SpotPrice": "0.20"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c3.8xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2b"
            }
        }
    ]
}
```

}

서브넷

```
{  
    "SpotPrice": "1.00",  
    "TargetCapacity": 30,  
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",  
    "LaunchSpecifications": [  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "c3.2xlarge",  
            "SubnetId": "subnet-1a2b3c4d",  
            "SpotPrice": "0.10"  
        },  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "c3.4xlarge",  
            "SubnetId": "subnet-1a2b3c4d",  
            "SpotPrice": "0.20"  
        },  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "c3.8xlarge",  
            "SubnetId": "subnet-1a2b3c4d"  
        }  
    ]  
}
```

예 5: 다각화된 할당 전략을 사용하여 스팟 집합 시작

다음 예제에서는 diversified 할당 전략을 사용합니다. 시작 사양의 인스턴스 유형은 다르지만 AMI와 가용 영역 또는 서브넷은 같습니다. 스팟 집합이 3개의 시작 사양에 각 유형의 인스턴스가 10개씩 있도록 30개의 인스턴스를 분산합니다. 자세한 내용은 [스팟 인스턴스의 할당 전략 \(p. 246\)](#) 단원을 참조하십시오.

가용 영역

```
{  
    "SpotPrice": "0.70",  
    "TargetCapacity": 30,  
    "AllocationStrategy": "diversified",  
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",  
    "LaunchSpecifications": [  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "c4.2xlarge",  
            "Placement": {  
                "AvailabilityZone": "us-west-2b"  
            }  
        },  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "m3.2xlarge",  
            "Placement": {  
                "AvailabilityZone": "us-west-2b"  
            }  
        },  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "r3.2xlarge",  
            "Placement": {  
                "AvailabilityZone": "us-west-2b"  
            }  
        }  
    ]  
}
```

```
        }
    ]
}
```

서브넷

```
{
    "SpotPrice": "0.70",
    "TargetCapacity": 30,
    "AllocationStrategy": "diversified",
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
    "LaunchSpecifications": [
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c4.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "m3.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "r3.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        }
    ]
}
```

특정 가용 영역 중단 시 EC2 용량이 스팟 요청을 수행할 확률을 높리는 가장 좋은 방법은, 영역 간에 분산하는 것입니다. 이 시나리오에서는 사용할 수 있는 각 가용 영역을 시작 사양에 포함합니다. 그리고 매번 같은 서브넷을 사용하는 대신 (각각 다른 영역에 매핑되는) 3개의 고유 서브넷을 사용합니다.

가용 영역

```
{
    "SpotPrice": "0.70",
    "TargetCapacity": 30,
    "AllocationStrategy": "diversified",
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
    "LaunchSpecifications": [
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c4.2xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2a"
            }
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "m3.2xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2b"
            }
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "r3.2xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2c"
            }
        }
    ]
}
```

```
        }  
    ]  
}
```

서브넷

```
{  
    "SpotPrice": "0.70",  
    "TargetCapacity": 30,  
    "AllocationStrategy": "diversified",  
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",  
    "LaunchSpecifications": [  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "c4.2xlarge",  
            "SubnetId": "subnet-1a2b3c4d"  
        },  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "m3.2xlarge",  
            "SubnetId": "subnet-2a2b3c4d"  
        },  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "r3.2xlarge",  
            "SubnetId": "subnet-3a2b3c4d"  
        }  
    ]  
}
```

예 6: 인스턴스 가중치를 사용하여 스팟 집합 시작

다음 예제에서는 인스턴스 가중치를 사용하는데, 이는 곧 가격이 인스턴스 시간당이 아니라 단위 시간당 가격이라는 의미입니다. 각 시작 구성마다 다른 인스턴스 유형과 다른 가중치가 나열됩니다. 스팟 집합은 단위 시간당 최저 가격의 인스턴스 유형을 선택합니다. 스팟 집합은 목표 용량을 인스턴스 가중치로 나누어 시작 할 스팟 인스턴스의 수를 계산합니다. 결과가 정수가 아닌 경우 스팟 집합은 결과를 다음 정수로 올림하므로 플릿의 크기가 목표 용량을 밀들지는 않습니다.

r3.2xlarge 요청이 성공하면 스팟이 이들 인스턴스 중 4개를 프로비저닝합니다. 20을 6으로 나누면 총 3.33개의 인스턴스가 되는데, 이를 올림 처리하여 4개의 인스턴스가 됩니다.

c3.xlarge 요청이 성공하면 스팟이 이들 인스턴스 7개를 프로비저닝합니다. 20을 3으로 나누면 총 6.66개의 인스턴스가 되는데, 이를 올림 처리하여 7개의 인스턴스가 됩니다.

자세한 내용은 [스팟 집합 인스턴스 가중치 부여 \(p. 249\)](#) 단원을 참조하십시오.

가용 영역

```
{  
    "SpotPrice": "0.70",  
    "TargetCapacity": 20,  
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",  
    "LaunchSpecifications": [  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "r3.2xlarge",  
            "Placement": {  
                "AvailabilityZone": "us-west-2b"  
            },  
            "WeightedCapacity": 6  
        },  
    ]  
}
```

```
{  
    "ImageId": "ami-1a2b3c4d",  
    "InstanceType": "c3.xlarge",  
    "Placement": {  
        "AvailabilityZone": "us-west-2b"  
    },  
    "WeightedCapacity": 3  
}  
]  
}
```

서브넷

```
{  
    "SpotPrice": "0.70",  
    "TargetCapacity": 20,  
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",  
    "LaunchSpecifications": [  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "r3.2xlarge",  
            "SubnetId": "subnet-1a2b3c4d",  
            "WeightedCapacity": 6  
        },  
        {  
            "ImageId": "ami-1a2b3c4d",  
            "InstanceType": "c3.xlarge",  
            "SubnetId": "subnet-1a2b3c4d",  
            "WeightedCapacity": 3  
        }  
    ]  
}
```

예 7: 온디맨드 용량으로 스팟 집합 시작

항상 인스턴스 용량을 사용할 수 있도록 스팟 집합 요청에 온디맨드 용량에 대한 요청을 포함할 수 있습니다. 용량이 있는 경우 온디맨드 요청이 항상 이행됩니다. 용량이 있고 가용 상태일 경우 목표 용량의 잔고는 스팟으로 이행됩니다.

다음 예에서는 원하는 목표 용량을 10으로 지정합니다. 이 중 5는 온디맨드 용량이어야 합니다. 스팟 용량은 지정하지 않습니다. 스팟 용량은 목표 용량에서 온디맨드 용량을 뺀 나머지 용량으로 간주됩니다. Amazon EC2는 사용 가능한 Amazon EC2 용량이 있는 경우 5개의 용량 단위를 온디맨드로 시작하고 5개($10-5=5$)의 용량 단위를 스팟으로 시작합니다.

자세한 내용은 [스팟 집합의 온디맨드 \(p. 246\)](#) 단원을 참조하십시오.

```
{  
    "IamFleetRole": "arn:aws:iam::781603563322:role/aws-ec2-spot-fleet-tagging-role",  
    "AllocationStrategy": "lowestPrice",  
    "TargetCapacity": 10,  
    "SpotPrice": null,  
    "ValidFrom": "2018-04-04T15:58:13Z",  
    "ValidUntil": "2019-04-04T15:58:13Z",  
    "TerminateInstancesWithExpiration": true,  
    "LaunchSpecifications": [],  
    "Type": "maintain",  
    "OnDemandTargetCapacity": 5,  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-0dbb04d4a6cca5ad1",  
                "Version": "2"  
            }  
        }  
    ]  
}
```

```
        },
        "Overrides": [
            {
                "InstanceType": "t2.medium",
                "WeightedCapacity": 1,
                "SubnetId": "subnet-d0dc51fb"
            }
        ]
    }
}
```

스팟 집합에 대한 CloudWatch 지표

Amazon EC2는 스팟 집합을 모니터링하는 데 사용할 수 있는 Amazon CloudWatch 지표를 제공합니다.

Important

정확성을 보장하기 위해, 이 측정치를 사용할 때는 세부 모니터링을 활성화하는 것이 좋습니다. 자세한 내용은 [인스턴스에 대한 세부 모니터링 활성화 또는 비활성화 \(p. 667\)](#) 단원을 참조하십시오.

CloudWatch가 제공하는 Amazon EC2 측정치에 대한 자세한 내용은 [CloudWatch를 사용해 인스턴스 모니터링 \(p. 667\)](#) 단원을 참조하십시오.

스팟 집합 지표

AWS/EC2Spot 네임스페이스에는 다음과 같은 측정치와 플릿의 스팟 인스턴스에 대한 CloudWatch 지표가 포함되어 있습니다. 자세한 정보는 [인스턴스 지표 \(p. 669\)](#) 단원을 참조하십시오.

지표	설명
AvailableInstancePoolsCount	스팟 집합 요청에 지정된 스팟 인스턴스 풀. 단위: 개수
BidsSubmittedForCapacity	Amazon EC2가 스팟 집합 요청을 제출한 용량. 단위: 개수
EligibleInstancePoolCount	Amazon EC2에서 요청을 이행할 수 있는 경우 스팟 집합 요청에 지정된 스팟 인스턴스 풀입니다. Amazon EC2는 스팟 인스턴스에 대해 지불할 의향이 있는 최대 가격이 스팟 가격보다 낮거나 스팟 가격이 온디맨드 인스턴스의 가격보다 높은 경우 풀에서 요청을 이행하지 않습니다. 단위: 개수
FulfilledCapacity	Amazon EC2가 달성한 용량. 단위: 개수
MaxPercentCapacityAllocation	스팟 집합 요청에 지정된 모든 스팟 집합 풀에 걸친 PercentCapacityAllocation의 최대값. 단위: 백분율
PendingCapacity	TargetCapacity와 FulfilledCapacity의 차이점. 단위: 개수

지표	설명
PercentCapacityAllocation	지정된 차원의 스팟 인스턴스 풀에 할당된 용량. 모든 스팟 인스턴스 풀에 기록된 최대값을 얻으려면 MaxPercentCapacityAllocation을 사용하십시오. 단위: 백분율
TargetCapacity	스팟 집합 요청의 목표 용량. 단위: 개수
TerminatingCapacity	프로비저닝된 용량이 목표 용량보다 커서 종료되는 용량입니다. 단위: 개수

수치 측정 단위가 Count(수)인 경우, 가장 유용한 통계는 Average(평균)입니다.

스팟 집합 차원

스팟 집합에 대한 데이터를 필터링하려면 다음 차원을 사용하십시오.

차원	설명
AvailabilityZone	가용 영역별로 데이터를 필터링합니다.
FleetRequestId	스팟 집합 요청별로 데이터를 필터링합니다.
InstanceType	인스턴스 유형별로 데이터를 필터링합니다.

스팟 집합에 대한 CloudWatch 지표 보기

Amazon CloudWatch 콘솔을 사용하여 스팟 집합에 대한 CloudWatch 지표를 볼 수 있습니다. 이 측정치들은 모니터링 그래프로 표시됩니다. 이 그래프는 스팟 집합이 활성화되면 데이터 포인트를 표시합니다.

측정치는 먼저 네임스페이스별로 그룹화된 다음, 각 네임스페이스 내에서 다양한 차원 조합별로 그룹화됩니다. 예를 들어, 모든 스팟 집합 지표를 볼 수 있거나 스팟 집합 요청 ID, 인스턴스 유형 또는 가용 영역별로 그룹화된 스팟 집합 지표를 볼 수 있습니다.

스팟 집합 측정치를 보려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 [Metrics]를 선택합니다.
3. EC2 스팟 네임스페이스를 선택합니다.

Note

EC2 스팟 네임스페이스가 표시되지 않는 경우 두 가지 이유가 있습니다. 스팟 집합을 아직 사용하지 않았습니다. 사용 중인 AWS 서비스만 Amazon CloudWatch에 지표를 보냅니다. 또는 지난 2주 동안 스팟 집합을 사용하지 않은 경우 네임스페이스가 나타나지 않습니다.

4. (선택 사항) 측정치를 차원으로 필터링하려면 다음 중 하나를 선택하십시오.

- 플랫 요청 지표 – 스팟 집합 요청별 그룹
- 가용 영역별 – 스팟 집합 요청 및 가용 영역별 그룹
- 인스턴스 유형별 – 스팟 집합 요청 및 인스턴스 유형별 그룹

- 가용 영역/인스턴스 유형별 – 스팟 집합 요청, 가용 영역 및 인스턴스 유형별 그룹
5. 측정치에 대한 데이터를 보려면 측정치 옆의 확인란을 선택합니다.

FleetRequestId	Metric Name
sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	AvailableInstancePoolsCount
sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	BidsSubmittedForCapacity
<input checked="" type="checkbox"/> sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	CPUUtilization
sfr-4a707781-8fac-459b-a5ae-4701fcee47d7	DiskReadBytes

스팟 집합의 자동 조정

자동 조정은 수요에 따라 스팟 집합의 목표 용량을 자동으로 늘리거나 줄이는 기능입니다. 선택 범위 내에서 하나 이상의 조정 정책에 대한 응답으로 스팟 집합이 인스턴스 시작(스케일 아웃) 또는 인스턴스 종료(스케일 인)를 수행할 수 있습니다.

스팟 집합은 다음과 같은 자동 조정을 지원합니다.

- **대상 추적 조정 (p. 301)** – 특정 측정치의 목표 값은 기준으로 플릿의 현재 용량을 늘리거나 줄입니다. 이 작업은 온도 조절기가 집안 온도를 유지하는 방식과 비슷합니다. — 사용자가 온도만 선택하면 나머지는 온도 조절기가 알아서 합니다.
- **단계 조정 (p. 301)** – 일련의 조정 조절값(즉, 경보 위반의 크기에 따라 달라지는 단계 조절값)에 따라 플릿의 현재 용량을 늘리거나 줄입니다.
- **예약된 조정 (p. 303)** – 날짜 및 시간을 기준으로 플릿의 현재 용량을 늘리거나 줄입니다.

인스턴스 가중치 (p. 249)를 사용하는 경우 스팟 집합에서 필요에 따라 목표 용량을 초과할 수 있다는 점에 주의하십시오. 이행된 용량이 부동 소수점 숫자일 수 있으나 목표 용량은 정수여야 하므로 스팟 집합은 결과를 다음 정수로 올립니다. 경보가 트리거되면 조정 정책의 결과를 확인할 때 이러한 동작을 고려해야 합니다. 예를 들어 목표 용량이 30, 이행된 용량이 30.1이고 조정 정책이 1을 빼다고 가정하십시오. 경보가 트리거되면 자동 조정 프로세스가 30.1에서 1을 빼 29.1을 도출한 후 30으로 올리므로 조정 작업이 수행되지 않습니다. 또 다른 예로, 선택한 인스턴스의 가중치가 2, 4, 8이고 목표 용량이 10이지만 가중치 2인 인스턴스를 사용할 수 없었기 때문에 스팟 집합이 가중치 4와 8인 인스턴스를 프로비저닝하여 이행된 용량이 12가 되었다고 가정합니다. 조정 정책이 목표 용량을 20% 줄이고 경보가 트리거되면 자동 조정 프로세스가 12에서 12 * 0.2를 빼 9.6을 도출한 후 10으로 올리므로 조정 작업이 수행되지 않습니다.

스팟 집합을 위해 생성한 조정 정책은 휴지 기간을 지원합니다. 이 기간은 이전 트리거 관련 조정 활동이 향후 조정 이벤트에 영향을 줄 수 있는 경우 조정 활동이 완료된 후의 시간(초)입니다. 확장 정책의 경우, 휴지 기간이 진행되는 동안 휴지하기 시작한 이전 확장 이벤트에 의해 추가된 용량은 다음 확장에 대해 원하는 용량의 일부로 계산됩니다. 지속적이지만 과도하지는 않게 확장하기 위한 목적입니다. 축소 정책의 경우, 휴지 기간은 만료될 때까지 후속 축소 요청을 차단하기 위해 사용됩니다. 보수적으로 축소하여 애플리케이션의 가용성을 보호하기 위한 목적입니다. 그러나 축소 후 휴지 기간 동안 다른 경보가 확장 정책을 트리거하면 자동 조정은 확장 가능한 대상을 즉시 확장합니다.

인스턴스 측정치를 1분 주기로 조정하는 것이 좋습니다. 이렇게 하면 사용량 변동에 따른 응답 속도를 더욱 높일 수 있기 때문입니다. 주기를 5분으로 하면 응답 시간이 느려질 뿐만 아니라 오랜 시간이 지난 측정치 데이터를 기준으로 조정하게 됩니다. 인스턴스에 대한 측정치 데이터를 CloudWatch에 1분 동안 전송하려면 특

이 세부 모니터링을 활성화해야 합니다. 자세한 내용은 [인스턴스에 대한 세부 모니터링 활성화 또는 비활성화 \(p. 667\)](#) 및 [정의된 파라미터를 사용하여 스팟 집합 요청 생성\(콘솔\) \(p. 277\)](#) 단원을 참조하십시오.

스팟 집합 조정 구성에 대한 자세한 내용은 다음 리소스를 참조하십시오.

- AWS CLI Command Reference의 [application-autoscaling](#) 단원
- [Application Auto Scaling API 참조](#)
- [Application Auto Scaling 사용 설명서](#)

스팟 집합 자동 조정에 필요한 IAM 권한

Amazon EC2, Amazon CloudWatch, Application Auto Scaling API의 조합으로 스팟 집합에 대한 자동 조정이 가능합니다. Amazon EC2로 스팟 집합 요청이 생성되고, CloudWatch로 경보가 생성되고, Application Auto Scaling로 스케일링 정책이 생성됩니다.

[스팟 집합에 대한 IAM 권한 \(p. 273\)](#) 및 Amazon EC2외에도 플릿 조정 설정에 액세스하는 IAM 사용자는 동적 조정을 지원하는 서비스에 대한 적절한 권한이 있어야 합니다. IAM 사용자는 다음 예제 정책에 표시된 조치를 사용할 수 있는 권한이 있어야 합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "application-autoscaling:*",  
                "ec2:DescribeSpotFleetRequests",  
                "ec2:ModifySpotFleetRequest",  
                "cloudwatch:DeleteAlarms",  
                "cloudwatch:DescribeAlarmHistory",  
                "cloudwatch:DescribeAlarms",  
                "cloudwatch:DescribeAlarmsForMetric",  
                "cloudwatch:GetMetricStatistics",  
                "cloudwatch>ListMetrics",  
                "cloudwatch:PutMetricAlarm",  
                "cloudwatch:DisableAlarmActions",  
                "cloudwatch:EnableAlarmActions",  
                "iam>CreateServiceLinkedRole",  
                "sns>CreateTopic",  
                "sns:Subscribe",  
                "sns:Get*",  
                "sns>List*"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

Application Auto Scaling API 호출에 대한 보다 세분화된 권한을 허용하는 고유한 IAM 정책을 만들 수도 있습니다. 자세한 정보는 Application Auto Scaling 사용 설명서의 [인증 및 액세스 제어](#)를 참조하십시오.

Application Auto Scaling 서비스에는 스팟 집합 및 CloudWatch 경보를 설명할 수 있는 권한과 사용자를 대신하여 스팟 집합 대상 용량을 수정할 수 있는 권한이 필요합니다. 스팟 집합에 자동 조정을 활성화하면 AWS Service Role for Application Auto Scaling (`EC2-SpotFleetRequest`)라는 서비스 연결 역할이 생성됩니다. 이 서비스 연결 역할은 정책에 대한 경보를 설명하고, 플릿의 현재 용량을 모니터링하고, 플릿의 용량을 수정할 수 있는 Application Auto Scaling 권한을 부여합니다. Application Auto Scaling에 대한 관리되는 스팟 집합의 원래 역할은 `aws-ec2-spot-fleet-autoscale-role`이지만 더 이상 필요하지 않습니다. 서비스 연결 역할은 Application Auto Scaling의 기본 역할입니다. 자세한 내용은 Application Auto Scaling 사용 설명서의 [서비스 연결 역할](#)을 참조하십시오.

대상 추적 정책을 사용하여 스팟 집합 조정

대상 추적 정책에 따라 측정치를 선택하고 대상 값을 설정합니다. 스팟 집합은 조정 정책을 트리거하는 CloudWatch 경보를 생성 및 관리하면서 측정치와 목표 값을 기준으로 조정 조절값을 계산합니다. 조정 정책은 필요에 따라 용량을 추가하거나 제거하여 측정치를 지정한 목표 값으로, 혹은 목표 값에 가깝게 유지합니다. 대상 추적 조정 정책은 측정치를 목표 값에 가깝게 유지하는 것 외에도 로드 패턴의 변화로 인한 측정치 변동에 따라 반응하여 플릿의 용량이 갑작스럽게 바뀌는 것을 최소화합니다.

각각 다른 측정치를 사용한다는 전제 하에 스팟 집합에 대해 다수의 대상 추적 정책을 생성할 수 있습니다. 스팟 플릿은 최대 플릿 용량을 제공하는 정책에 따라 조정됩니다. 따라서 다양한 시나리오를 포괄하고 애플리케이션 워크로드를 처리하기에 충분한 용량을 항상 확보할 수 있습니다.

애플리케이션 가용성을 보장하기 위해 스팟 플릿은 측정치에 비례하여 가능한 신속하게 확장되지만, 축소는 점진적으로 이루어집니다.

목표 용량이 줄어 스팟 집합이 인스턴스를 종료하면 해당 인스턴스는 스팟 인스턴스 중단 공지를 받습니다.

스팟 집합이 대상 추적 정책에서 관리하는 CloudWatch 경보는 편집하거나 삭제하지 마십시오. 대상 추적 조정 정책을 삭제하면 스팟 집합에서 경보가 자동으로 삭제됩니다.

제한

- 스팟 집합 요청에 `maintain` 요청 유형이 있어야 합니다. 1회 요청 또는 스팟 블록에는 자동 조정이 지원되지 않습니다.

대상 추적 정책을 구성하려면(콘솔)

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 스팟 요청을 선택합니다.
- 스팟 집합 요청을 선택한 후 Auto Scaling을 선택합니다.
- 자동 조정이 구성되어 있지 않으면 구성을 선택합니다.
- 다음 사이로 용량 조정(Scale capacity between)을 사용하여 플릿에 대한 최소 및 최대 용량을 설정합니다. 자동 조정에서 최소 용량 미만이거나 최대 용량을 초과하는 플릿을 조정하지 않습니다.
- 정책 이름에서 정책의 이름을 입력합니다.
- 대상 지표를 선택합니다.
- 측정치에 대한 대상 값을 입력합니다.
- (선택 사항) 기본 휴지 기간을 수정하려면 휴지 기간(Cooldown period)을 설정합니다.
- (선택 사항) 현재 구성에 따라 축소 정책 생성을 생략하려면 축소 비활성화(Disable scale-in)을 선택합니다. 다른 구성을 사용하여 축소 정책을 생성할 수 있습니다.
- 저장을 선택합니다.

AWS CLI를 사용하여 대상 추적 정책을 구성하려면

- `register-scalable-target` 명령을 사용하여 스팟 집합 요청을 확장 가능 대상으로 등록합니다.
- `put-scaling-policy` 명령을 사용하여 조정 정책을 생성합니다.

단계 조정 정책을 사용하여 스팟 집합 조정

단계 조정 정책을 사용하여 조정 프로세스를 트리거하도록 CloudWatch 경보를 지정합니다. 예를 들어, CPU 사용률이 특정 레벨에 도달하면 확장하려는 경우 Amazon EC2에서 제공하는 `CPUUtilization` 측정치를 사용하여 경보를 생성합니다.

단계 조정 정책을 생성할 때 다음과 같은 조정 조절 유형 중 하나를 지정해야 합니다.

- 추가 – 지정된 수의 용량 유닛 또는 지정된 현재 용량의 퍼센트까지 플릿의 대상 용량을 늘립니다.
- 제거 – 지정된 수의 용량 단위 또는 지정된 현재 용량의 퍼센트까지 플릿의 대상 용량을 줄입니다.
- 설정 – 플릿의 대상 용량을 지정된 수의 용량 단위로 설정합니다.

경보가 트리거되면 자동 조정 프로세스가 이행된 용량과 조정 정책을 사용하여 새로운 목표 용량을 계산한 후 그에 따라 목표 용량을 업데이트합니다. 예를 들어 목표 용량과 이행된 용량이 10이고 조정 정책이 1을 추가한다고 가정하십시오. 경보가 트리거되면 자동 조정 프로세스가 10에 1을 더해 11이 되므로 스팟 집합이 1 인스턴스를 시작합니다.

목표 용량이 줄어 스팟 집합이 인스턴스를 종료하면 해당 인스턴스는 스팟 인스턴스 중단 공지를 받습니다.

제한

- 스팟 집합 요청에 `maintain` 요청 유형이 있어야 합니다. 1회 요청 또는 스팟 블록에는 자동 조정이 지원되지 않습니다.

사전 조건

- 어떤 CloudWatch 지표가 애플리케이션에 중요한지 생각하십시오. AWS에서 제공하는 측정치 또는 사용자 지정 측정치를 기반으로 CloudWatch 경보를 생성할 수 있습니다.
- 조정 정책에 사용할 AWS 측정치에 대해 측정치를 제공하는 서비스에서 기본적으로 활성화하지 않는 경우 CloudWatch 측정치 수집을 활성화합니다.

CloudWatch 경보를 생성하려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 Alarms를 선택합니다.
3. 경보 생성을 선택합니다.
4. [Specify metric and conditions] 페이지에서 Select metric을 선택합니다.
5. EC2 Spot, Fleet Request Metrics을 선택하고, CPUUtilization과 같은 메트릭을 선택하고 Select metric을 선택합니다.

선택한 지표에 대한 그래프와 기타 정보가 표시된 Specify metric and conditions(지표 및 조건 지정) 페이지가 나타납니다.

6. 기간에 대해 예를 들어 1분과 같은 경보에 대한 평가 기간을 선택합니다. 경보를 평가할 때 각 기간이 하나의 데이터 포인트로 집계됩니다.

Note

기간이 짧을수록 경보가 더 민감해집니다.

7. 조건에서 임계 조건을 정의하여 경보를 정의합니다. 예를 들어, 지표 값이 80% 이상일 때마다 경보를 트리거하는 임계값을 정의할 수 있습니다.
8. 추가 구성에서 경고할 데이터포인트에 대해 알람을 트리거하는 ALARM 상태에 있어야 하는 데이터포인트(평가 기간)의 수를 지정합니다. 예를 들어 1개의 평가 기간 또는 3개의 평가 기간 중 2개입니다. 그러면 다수의 연속 기간이 위반되면 ALARM 상태가 되는 경보가 생성됩니다. 자세한 내용은 Amazon CloudWatch 사용 설명서의 [경보 평가](#)를 참조하십시오.
9. Missing data treatment(누락된 데이터 처리)에서 옵션 중 하나를 선택합니다(또는 기본값인 Treat missing data as missing(누락된 데이터를 누락으로 처리)를 그대로 사용). 자세한 내용은 Amazon CloudWatch 사용 설명서의 [CloudWatch 경보가 누락된 데이터를 처리하는 방법 구성](#) 단원을 참조하십시오.
10. [Next]를 선택합니다.
11. (선택 사항) 조정 이벤트 알림을 수신하려면 알림에 대해 알림을 받는 Amazon SNS 주제를 선택하거나 작성할 수 있습니다. 또는 지금 알림을 삭제하고 필요에 따라 나중에 추가할 수 있습니다.

12. [Next]를 선택합니다.
13. 설명 추가에서 경보의 이름과 설명을 입력하고 다음을 선택하십시오.
14. 경보 생성을 선택합니다.

스팟 집합에 대한 단계 조정 정책을 구성하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다.
3. 스팟 집합 요청을 선택한 후 Auto Scaling을 선택합니다.
4. 자동 조정이 구성되어 있지 않으면 구성을 선택합니다.
5. 다음 사이로 용량 조정(Scale capacity between)을 사용하여 플랫에 대한 최소 및 최대 용량을 설정합니다. 자동 조정에서 최소 용량 미만이거나 최대 용량을 초과하는 플랫을 조정하지 않습니다.
6. 처음에 조정 정책(Scaling policies)에 ScaleUp 및 ScaleDown이라는 정책이 포함됩니다. 이러한 정책을 완료하거나 정책 제거(Remove policy)를 선택하여 삭제할 수 있습니다. 또한 정책 추가를 선택하여 정책을 추가할 수도 있습니다.
7. 정책을 정의하려면 다음을 수행합니다.
 - a. 정책 이름에서 정책의 이름을 입력합니다.
 - b. 정책 트리거(Policy trigger)에서 기존 경보를 선택하거나 Create new alarm(새 경보 생성)을 선택하여 Amazon CloudWatch 콘솔을 열고 경보를 생성합니다.
 - c. 용량 수정(Modify capacity)에서 조정 조절 유형을 선택하고, 숫자를 선택한 후 단위를 선택합니다.
 - d. (선택 사항) 단계 조정을 수행하려면 단계 정의(Define steps)를 선택합니다. 기본적으로 추가 정책에 하한값으로 -infinity 값이, 상한값으로 경보 임계치가 적용됩니다. 또한 제거 정책에 하한값으로 경보 임계치 및 상한값으로 +infinity 값이 기본적으로 적용됩니다. 다른 단계를 추가하려면 단계 추가/Add step)를 선택합니다.
 - e. (선택 사항) 휴지 기간의 기본값을 수정하려면 휴지 기간(Cooldown period)에서 숫자를 선택합니다.
8. 저장을 선택합니다.

AWS CLI를 사용하여 스팟 집합에 대한 단계 조정 정책을 구성하려면

1. `register-scalable-target` 명령을 사용하여 스팟 집합 요청을 확장 가능 대상으로 등록합니다.
2. `put-scaling-policy` 명령을 사용하여 조정 정책을 생성합니다.
3. `put-metric-alarm` 명령을 사용하여 조정 정책을 트리거하는 경보를 생성합니다.

예약된 조정을 사용하여 스팟 집합 조정

일정을 기반으로 조정을 수행하면 수요에 따른 로드 변경에 맞게 애플리케이션을 조정할 수 있습니다. 예약된 조정을 사용하려면 스팟 집합이 특정 시간에 조정 작업을 수행하도록 하는 예약된 작업을 생성할 수 있습니다. 예약된 작업을 생성할 때 스팟 집합, 규모 조정 활동이 발생해야 할 시점, 최소 용량 및 최대 용량을 지정할 수 있습니다. 규모를 한 번만 조정하거나 반복되는 일정으로 조정하도록 예약된 작업을 생성할 수 있습니다.

제한

- 스팟 집합 요청에 `maintain` 요청 유형이 있어야 합니다. 1회 요청 또는 스팟 블록에는 자동 조정이 지원되지 않습니다.

1회성 예약된 작업을 만들려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다.

3. 스팟 집합 요청을 선택한 후 예약 조정을 선택합니다.
4. 예약 작업 만들기(Create Scheduled Action)를 선택합니다.
5. 이름에서 예약된 작업의 이름을 지정합니다.
6. 최소 용량, 최대 용량 또는 두 가지 모두 입력합니다.
7. 반복에서 1회(Once)를 선택합니다.
8. (선택 사항) 시작 시간, 종료 시간 또는 두 가지 모두에 대해 날짜와 시간을 선택합니다.
9. 제출을 선택합니다.

반복되는 일정으로 조정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다.
3. 스팟 집합 요청을 선택한 후 예약 조정을 선택합니다.
4. 반복에서 사전 정의된 일정(예: 매일(Every day)) 중 하나를 선택하거나 사용자 지정을 선택하고 Cron 표현식을 입력합니다. 예약된 조정에서 지원하는 Cron 표현식에 대한 자세한 내용은 Amazon CloudWatch Events 사용 설명서의 [Cron 표현식](#) 단원을 참조하십시오.
5. (선택 사항) 시작 시간, 종료 시간 또는 두 가지 모두에 대해 날짜와 시간을 선택합니다.
6. 제출을 선택합니다.

예약된 작업을 편집하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다.
3. 스팟 집합 요청을 선택한 후 예약 조정을 선택합니다.
4. 예약된 작업을 선택한 다음, 작업, 편집을 선택합니다.
5. 필요한 변경을 수행하고 제출을 선택합니다.

예약된 작업을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택합니다.
3. 스팟 집합 요청을 선택한 후 예약 조정을 선택합니다.
4. 예약된 작업을 선택한 다음, 작업, 삭제를 선택합니다.
5. 확인 메시지가 나타나면 Delete를 선택합니다.

AWS CLI를 사용하여 예약된 조정을 관리하려면

다음 명령을 사용합니다.

- [put-scheduled-action](#)
- [describe-scheduled-actions](#)
- [delete-scheduled-action](#)

스팟 요청 상태

스팟 인스턴스 요청을 추적하고 스팟 인스턴스 사용 계획을 세우는데 도움이 되도록 Amazon EC2에서 제공하는 요청 상태를 사용합니다. 예를 들어, 요청 상태는 스팟 요청이 아직 이행되지 않는 이유를 알려주거나, 스팟 요청을 이행할 수 없는 제약 조건을 나열할 수 있습니다.

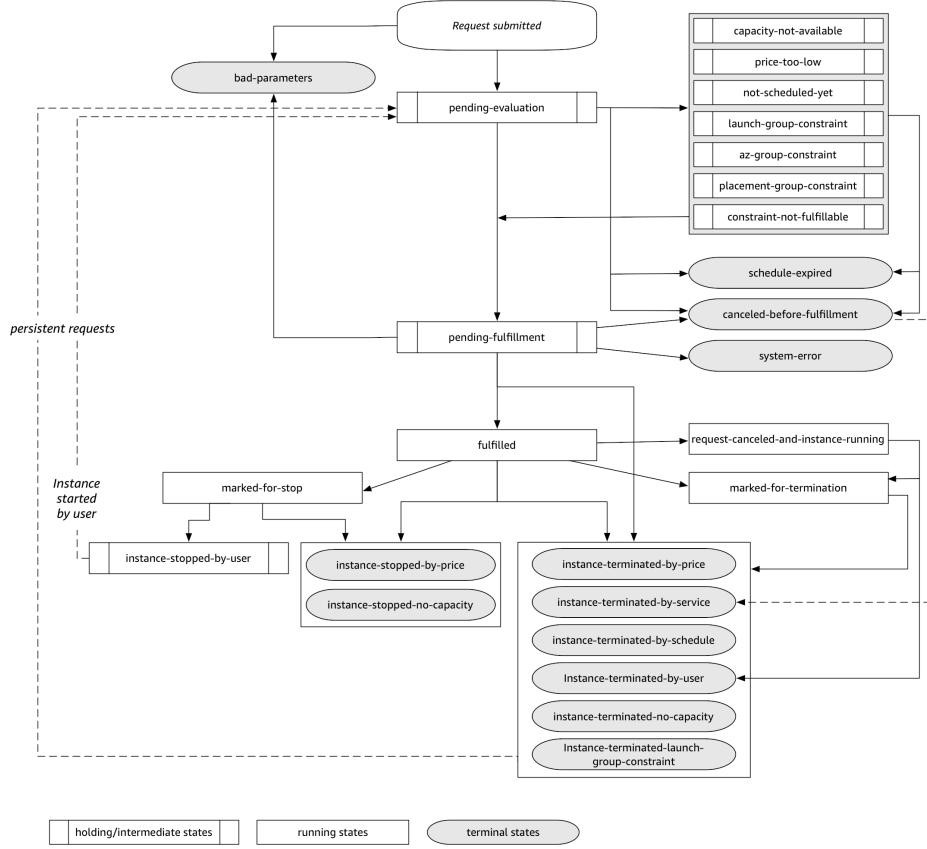
프로세스의 각 단계(스팟 요청 수명 주기라고도 함)에서 특정 이벤트에 따라 연속 요청 상태가 결정됩니다.

목차

- [스팟 요청의 수명 주기 \(p. 305\)](#)
- [요청 상태 정보 가져오기 \(p. 308\)](#)
- [스팟 요청 상태 코드 \(p. 308\)](#)

스팟 요청의 수명 주기

다음 디어그램에서는 제출부터 종료까지 전체 수명 주기 동안 스팟 요청이 따를 수 있는 경로를 보여 줍니다. 각 단계는 노드로 표시되며 각 노드의 상태 코드는 스팟 요청 및 스팟 인스턴스의 상태를 설명합니다.



평가 보류

하나 이상의 요청 파라미터가 유효하지 않은(bad-parameters) 경우를 제외하고, 스팟 인스턴스 요청을 생성하는 즉시 요청은 pending-evaluation 상태로 전환됩니다.

상태 코드	요청 상태	인스턴스 상태
pending-evaluation	open	해당 사항 없음
bad-parameters	closed	해당 사항 없음

보류

하나 이상의 요청 제약 조건이 적용되지만 아직 충족될 수 없는 경우 또는 용량이 부족한 경우 요청은 제약 조건이 충족될 때까지 대기하는 보류 상태로 전환됩니다. 요청 옵션은 요청이 이행될 가능성에 영향을 미칩니다.

니다. 예를 들어, 최고 가격을 현재 스팟 가격보다 낮게 지정할 경우 스팟 가격이 최고 가격 아래로 떨어질 때 까지 요청은 보류 상태로 유지됩니다. 가용 영역 그룹을 지정할 경우 가용 영역 제약 조건이 충족될 때까지 요청은 보류 상태로 유지됩니다.

특정 가용 영역 중단 시, 다른 가용 영역에서의 스팟 인스턴스 요청에 사용할 수 있는 예비 EC2 용량이 영향 받을 수 있습니다.

상태 코드	요청 상태	인스턴스 상태
capacity-not-available	open	해당 사항 없음
price-too-low	open	해당 사항 없음
not-scheduled-yet	open	해당 사항 없음
launch-group-constraint	open	해당 사항 없음
az-group-constraint	open	해당 사항 없음
placement-group-constraint	open	해당 사항 없음
constraint-not-fulfillable	open	해당 사항 없음

평가/이행 보류-끝

특정 기간 동안에만 유효한 요청을 생성하는 경우 요청이 이행 보류 단계에 도달하기 전에 이 기간이 만료되면 스팟 인스턴스 요청은 terminal 상태로 전환될 수 있습니다. 요청을 취소하거나 시스템 오류가 발생하는 경우에도 이와 같이 될 수 있습니다.

상태 코드	요청 상태	인스턴스 상태
schedule-expired	cancelled	해당 사항 없음
canceled-before-fulfillment*	cancelled	해당 사항 없음
bad-parameters	failed	해당 사항 없음
system-error	closed	해당 사항 없음

* 사용자가 요청을 취소하는 경우.

이행 보류

지정한 제약 조건(있는 경우)이 충족되고 최고 가격이 현재 스팟 가격보다 높거나 같은 경우 스팟 요청은 pending-fulfillment 상태가 됩니다.

이 시점에 Amazon EC2는 요청한 인스턴스를 프로비저닝할 준비를 합니다. 프로세스가 이 시점에 종지되는 경우 스팟 인스턴스가 시작되기 전에 사용자가 프로세스를 취소했기 때문일 수 있습니다. 예기치 않은 시스템 오류가 원인일 수도 있습니다.

상태 코드	요청 상태	인스턴스 상태
pending-fulfillment	open	해당 사항 없음

이행됨

스팟 인스턴스의 모든 사양이 충족되면 스팟 요청이 이행됩니다. Amazon EC2에서 스팟 인스턴스를 시작합니다. 이 작업은 몇 분 정도 걸릴 수 있습니다. 중단된 스팟 인스턴스가 최대 절전 모드로 전환되거나 중지되는 경우 요청을 다시 이행할 수 있거나 요청이 취소될 때까지 인스턴스는 이 상태를 유지합니다.

상태 코드	요청 상태	인스턴스 상태
fulfilled	active	pending → running
fulfilled	active	stopped → running

스팟 인스턴스를 중지하면 스팟 인스턴스를 다시 시작할 수 있거나 요청을 취소할 때까지 스팟 인스턴스는 `marked-for-stop` 또는 `instance-stopped-by-user` 상태로 전환됩니다.

상태 코드	요청 상태	인스턴스 상태
<code>marked-for-stop</code>	active	stopping
<code>instance-stopped-by-user</code> *	disabled 또는 cancelled**	stopped

* 인스턴스를 중지하거나 인스턴스에서 종료 명령을 실행하면 스팟 인스턴스는 `instance-stopped-by-user` 상태로 전환됩니다. 인스턴스를 중지한 후 다시 시작할 수 있습니다. 다시 시작하면 스팟 인스턴스 요청은 `pending-evaluation` 상태로 돌아온 다음, 제약 조건이 충족되면 Amazon EC2가 새스팟 인스턴스를 시작합니다.

** 스팟 인스턴스를 중지하지만 요청을 취소하지 않는 경우 스팟 요청 상태는 `disabled`입니다. 스팟 인스턴스가 중지되고 요청이 만료되는 경우 요청 상태는 `cancelled`입니다.

이행됨-끝

최고 가격이 스팟 가격보다 높거나 같고 인스턴스 유형에 대한 가용 용량이 있으며 사용자가 인스턴스를 종료하지 않는 한 스팟 인스턴스는 계속 실행됩니다. 스팟 가격 또는 가용 용량을 변경하기 위해 Amazon EC2에서 스팟 인스턴스를 종료해야 하는 경우 스팟 요청이 종료 상태로 전환됩니다. 사용자가 스팟 요청을 취소하거나 스팟 인스턴스를 종료하는 경우에도 요청이 종료 상태로 전환됩니다.

상태 코드	요청 상태	인스턴스 상태
<code>request-canceled-and-instance-running</code>	<code>cancelled</code>	<code>running</code>
<code>marked-for-stop</code>	active	<code>running</code>
<code>marked-for-termination</code>	active	<code>running</code>
<code>instance-stopped-by-price</code>	<code>disabled</code>	<code>stopped</code>
<code>instance-stopped-by-user</code>	<code>disabled</code>	<code>stopped</code>
<code>instance-stopped-no-capacity</code>	<code>disabled</code>	<code>stopped</code>
<code>instance-terminated-by-price</code>	<code>closed(1회), open(영구)</code>	<code>terminated</code>

상태 코드	요청 상태	인스턴스 상태
instance-terminated-by-schedule	closed	terminated
instance-terminated-by-service	cancelled	terminated
instance-terminated-by-user	closed 또는 cancelled*	terminated
instance-terminated-no-capacity	closed(일회), open(영구)	terminated
instance-terminated-launch-group-constraint	closed(일회), open(영구)	terminated

* 인스턴스를 종료하되 입찰을 취소하지 않는 경우 요청 상태는 closed입니다. 인스턴스를 종료하고 요청을 취소하는 경우 요청 상태는 cancelled입니다. 스팟 요청을 취소하기 전에 스팟 인스턴스를 종료해도 Amazon EC2에서 스팟 인스턴스가 종료되었음을 감지하기 전까지 지연이 발생할 수 있습니다. 이 경우 요청 상태는 closed 또는 cancelled일 수 있습니다.

영구 요청

스팟 인스턴스가 종료될 때(사용자가 종료하거나 Amazon EC2에서 종료) 스팟 요청이 영구 요청인 경우 pending-evaluation 상태로 복귀한 후 제약 조건이 충족되면 Amazon EC2에서 새로운 스팟 인스턴스를 시작할 수 있습니다.

요청 상태 정보 가져오기

AWS Management 콘솔 또는 명령줄 도구를 사용하여 요청 상태 정보를 가져올 수 있습니다.

요청 상태 정보를 가져오려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스팟 요청을 선택한 다음 스팟 요청을 선택합니다.
3. 상태를 확인하려면 설명 탭에서 상태 필드를 선택합니다.

명령줄을 사용하여 요청 상태 정보를 가져오려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-spot-instance-requests\(AWS CLI\)](#)
- [Get-EC2SpotInstanceRequest\(Windows PowerShell용 AWS 도구\)](#)

스팟 요청 상태 코드

스팟 요청 상태 정보는 상태 코드, 업데이트 시간 및 상태 메시지로 구성됩니다. 이러한 정보를 하나로 모으면 스팟 요청 배치를 결정하는 데 도움이 됩니다.

다음은 스팟 요청 상태 코드입니다.

`az-group-constraint`

Amazon EC2가 동일한 가용 영역에 요청한 모든 인스턴스를 시작할 수 있는 것은 아닙니다.

bad-parameters

스팟 요청에 대한 파라미터 하나 이상이 올바르지 않습니다(예를 들어, 지정한 AMI가 존재하지 않음). 상태 메시지는 어떤 파라미터가 유효하지 않은지를 나타냅니다.

canceled-before-fulfillment

요청이 이행되기 전에 사용자가 스팟 요청을 취소했습니다.

capacity-not-available

요청한 인스턴스에 사용 가능한 용량이 부족합니다.

constraint-not-fulfillable

하나 이상의 제약 조건이 올바르지 않기 때문에(예: 가용 영역이 존재하지 않음) 스팟 요청을 이행할 수 없습니다. 상태 메시지는 어떤 제약 조건이 유효하지 않은지를 나타냅니다.

fulfilled

상태 요청이 `active` 상태이며 Amazon EC2가 스팟 인스턴스를 시작하고 있습니다.

instance-stopped-by-price

스팟 가격이 최고 가격을 초과하여 인스턴스가 종지됩니다.

instance-stopped-by-user

사용자가 인스턴스를 종지했거나 인스턴스에서 종료 명령을 실행했기 때문에 인스턴스가 종지되었습니다.

instance-stopped-no-capacity

인스턴스에 사용 가능한 스팟 용량이 부족하여 인스턴스가 종지됩니다.

instance-terminated-by-price

스팟 가격이 최고 가격을 초과하여 인스턴스가 종료됩니다. 요청이 영구적일 경우 프로세스가 다시 시작되므로 요청은 평가 보류 상태입니다.

instance-terminated-by-schedule

스팟 인스턴스가 예약된 유효 기간의 만료로 종료되었습니다.

instance-terminated-by-service

인스턴스가 종지된 상태에서 종료되었습니다.

instance-terminated-by-user 또는 spot-instance-terminated-by-user

이행된 스팟 인스턴스를 종료했으므로 요청 상태는 `closed`(영구 요청이 아닌 경우)이고 인스턴스 상태는 `terminated`입니다.

instance-terminated-launch-group-constraint

시작 그룹에 있는 하나 이상의 인스턴스가 종료되었으므로 시작 그룹 제약 조건이 더 이상 충족되지 않습니다.

instance-terminated-no-capacity

인스턴스에 사용 가능한 스팟 용량이 부족하여 인스턴스가 종료됩니다.

launch-group-constraint

Amazon EC2가 동일한 시간에 요청한 모든 인스턴스를 시작할 수 있는 것은 아닙니다. 시작 그룹에 있는 모든 인스턴스가 함께 시작되고 종료됩니다.

limit-exceeded

EBS 볼륨 또는 전체 볼륨 스토리지 수 제한을 초과했습니다. 이러한 제한값 및 증가 요청 방법에 대한 자세한 내용은 Amazon Web Services 일반 참조에서 [Amazon EBS 제한](#)을 참조하십시오.

marked-for-stop

스팟 인스턴스가 중지할 대상으로 표시되어 있습니다.

marked-for-termination

스팟 인스턴스가 종료할 대상으로 표시되어 있습니다.

not-scheduled-yet

예정된 날짜까지 스팟 요청이 평가되지 않습니다.

pending-evaluation

스팟 인스턴스 요청을 수행한 후 시스템에서 요청 파라미터를 평가하는 동안 요청이 `pending-evaluation` 상태로 전환됩니다.

pending-fulfillment

Amazon EC2에서 스팟 인스턴스를 프로비저닝하려고 합니다.

placement-group-constraint

이 시점에는 스팟 인스턴스를 배치 그룹에 추가할 수 없으므로 아직 스팟 요청을 이행할 수 없습니다.

price-too-low

최고 가격이 스팟 가격보다 낮기 때문에 요청을 이행할 수 없습니다. 이 경우 인스턴스가 시작되지 않으며 요청이 `open` 상태로 유지됩니다.

request-canceled-and-instance-running

스팟 인스턴스가 아직 실행되고 있는 중에 사용자가 스팟 요청을 취소했습니다. 요청은 `cancelled` 상태지만 인스턴스는 여전히 `running` 상태입니다.

schedule-expired

지정된 날짜 이전에 요청이 이행되지 않았기 때문에 스팟 요청이 만료되었습니다.

system-error

예상치 않은 시스템 오류입니다. 이 문제가 반복되면 AWS Support에 문의하십시오.

스팟 인스턴스 중단

스팟 인스턴스에 대한 수요는 매 순간 상당히 다를 수 있으며 스팟 인스턴스의 가용성도 사용 가능한 미사용 EC2 인스턴스의 양에 따라 상당히 달라질 수 있습니다. 스팟 인스턴스가 중단될 가능성은 항상 있습니다. 따라서 스팟 인스턴스 중단에 대비하여 애플리케이션을 준비해야 합니다.

EC2 집합 또는 스팟 집합에 지정된 온디맨드 인스턴스는 중단할 수 없습니다.

목차

- [중단 이유 \(p. 311\)](#)
- [중단 동작 \(p. 311\)](#)
- [중단 동작 지정 \(p. 313\)](#)
- [중단 준비 \(p. 313\)](#)
- [인스턴스 최대 절전 모드 준비 \(p. 314\)](#)
- [스팟 인스턴스 중단 공지 \(p. 314\)](#)
- [중단된 스팟 인스턴스 찾기 \(p. 316\)](#)
- [Amazon EC2가 스팟 인스턴스를 중단했는지 여부 확인 \(p. 316\)](#)
- [중단된 스팟 인스턴스에 대한 청구 \(p. 316\)](#)

중단 이유

Amazon EC2에서 스팟 인스턴스를 중단시킬 수 있는 이유는 다음과 같습니다.

- 가격 – 스팟 가격이 최고 가격보다 큩니다.
- 용량 – 미사용 EC2 인스턴스가 스팟 인스턴스 수요를 충족할 만큼 충분하지 않으면 Amazon EC2에서 스팟 인스턴스를 중단합니다. 인스턴스가 중단되는 순서는 Amazon EC2에서 결정됩니다.
- 제약 조건 – 요청에 시작 그룹 또는 가용 영역 그룹과 같은 제약 조건이 포함되는 경우 제약 조건을 더 이상 충족할 수 없으면 이러한 스팟 인스턴스가 그룹으로 종료됩니다.

중단 동작

스팟 인스턴스를 중단할 때 Amazon EC2가 다음 중 하나를 수행하도록 지정할 수 있습니다.

- 스팟 인스턴스 종지
- 스팟 인스턴스를 최대 절전 모드로 전환
- 스팟 인스턴스 종료

기본값은 중단된 스팟 인스턴스를 종료하는 것입니다. 중단 동작을 변경하려면 [중단 동작 지정 \(p. 313\)](#) 단원을 참조하십시오.

중단된 스팟 인스턴스 종지

다음 요구 사항이 충족되는 경우 Amazon EC2가 중단된 스팟 인스턴스를 종지하도록 중단 동작을 지정할 수 있습니다.

요구 사항

- 스팟 인스턴스 요청의 유형은 `persistent`이어야 합니다. 스팟 인스턴스 요청에서 시작 그룹을 지정할 수 없습니다.
- EC2 집합 또는 스팟 집합 요청의 유형은 `maintain`이어야 합니다.
- 루트 볼륨은 인스턴스 스토어 볼륨이 아니라 EBS 볼륨이어야 합니다.

스팟 인스턴스가 Spot 서비스에 의해 중지된 후에는 Spot 서비스만 스팟 인스턴스를 다시 시작할 수 있으며 동일한 시작 사양을 사용해야 합니다.

`persistent` 스팟 인스턴스 요청에 의해 시작된 스팟 인스턴스의 경우 스팟 서비스는 용량이 동일한 가용 영역에서 사용 가능한 경우 중지된 인스턴스와 동일한 인스턴스 유형에 대해 중지된 인스턴스를 다시 시작합니다.

EC2 집합 또는 스팟 집합의 인스턴스가 중지되고 폴릿이 `maintain` 유형인 경우 스팟 서비스는 대체 인스턴스를 시작하여 대상 용량을 유지합니다. Spot 서비스는 지정된 할당 전략(`lowestPrice`, `diversified` 또는 `InstancePoolsToUseCount`)을 기반으로 최상의 풀을 찾습니다. 이전에 중지된 인스턴스로 풀의 우선 순위를 지정하지 않습니다. 나중에 할당 전략으로 이전에 중지된 인스턴스가 풀에 포함되면 스팟 서비스는 중지된 인스턴스를 다시 시작하여 대상 용량을 충족합니다.

예를 들어 `lowestPrice` 할당 전략을 사용하여 스팟 집합을 고려하십시오. 초기 시작 시, `c3.large` 풀은 시작 사양의 `lowestPrice` 기준을 충족합니다. 나중에 `c3.large` 인스턴스가 중단되면 스팟 서비스는 인스턴스를 중지하고 `lowestPrice` 전략에 맞는 다른 풀에서 용량을 보충합니다. 이번에는 풀이 `c4.large` 풀로, 스팟 서비스는 `c4.large` 인스턴스를 시작하여 대상 용량을 충족합니다. 마찬가지로 스팟 집합은 다음에 `c5.large` 풀로 이동할 수 있습니다. 이러한 각각의 전환에서 스팟 서비스는 이전에 인스턴스가 중지된 풀에 우선 순위를 지정하지 않고 지정된 할당 전략에 따라 우선 순위를 지정합니다. `lowestPrice` 전략은 이전에 인스턴스가 중지된 풀로 되돌아갈 수 있습니다. 예를 들어 인스턴스가 `c5.large` 풀에서 중단되고 `lowestPrice` 전략이 다시 `c3.large` 또는 `c4.large` 풀로 연결되면 이전에 중지 된 인스턴스가 다시 시작되어 대상 용량을 충족합니다.

스팟 인스턴스가 중지되었을 때 일부 인스턴스 속성을 수정할 수 있지만 인스턴스 유형은 수정할 수 없습니다. EBS 볼륨을 분리하거나 삭제한 경우 스팟 인스턴스를 시작해도 연결되지 않습니다. 루트 볼륨을 분리했는데 스팟 서비스가 해당 스팟 인스턴스를 시작하려고 시도하면 인스턴스 시작은 실패하고 스팟 서비스는 중지된 인스턴스를 종료합니다.

중지된 상태의 스팟 인스턴스를 종료할 수 있습니다. 스팟 요청이나 EC2 집합 또는 스팟 집합을 취소하면 스팟 서비스에서는 중지된 상태의 연결된 스팟 인스턴스를 모두 종료합니다.

스팟 인스턴스가 중지 상태인 동안에는 유지 중인 EBS 볼륨에 대한 요금만 부과됩니다. EC2 집합 및 스팟 집합을 사용하는 경우 중지된 인스턴스가 많으면 해당 계정의 EBS 볼륨 수 제한을 초과할 수 있습니다.

중단된 스팟 인스턴스를 최대 절전 모드로 전환

다음 요구 사항이 충족되는 경우 Amazon EC2가 중단된 스팟 인스턴스를 최대 절전 모드로 전환하도록 중단 동작을 지정할 수 있습니다.

요구 사항

- 스팟 인스턴스 요청의 유형은 `persistent`이어야 합니다. 스팟 인스턴스 요청에서 시작 그룹을 지정할 수 없습니다.
- EC2 집합 또는 스팟 집합 요청의 유형은 `maintain`이어야 합니다.
- 루트 볼륨은 인스턴스 스토어 볼륨이 아니라 EBS 볼륨이어야 하고 최대 절전 모드에서 인스턴스 메모리 (RAM)를 저장하기에 충분한 크기여야 합니다.
- 메모리 용량이 100GB 미만인 C3, C4, C5, M4, M5, R3, R4 인스턴스가 지원됩니다.
- Amazon Linux 2, Amazon Linux AMI, 4.4.0-1041 이상의 AWS-tuned Ubuntu 커널(linux-aws)이 포함된 Ubuntu, Windows Server 2008 R2 이상 등의 운영 체제가 지원됩니다.
- 지원되는 운영 체제에서 최대 절전 에이전트를 설치하거나 다음과 같이 이미 에이전트를 포함하고 있는 AMI 중 하나를 사용하십시오.
 - Amazon Linux 2
 - Amazon Linux AMI 2017.09.1 이상
 - Ubuntu Xenial 16.04 20171121 이상
 - Windows Server 2008 R2 AMI 2017.11.19 이상
 - Windows Server 2012 또는 Windows Server 2012 R2 AMI 2017.11.19 이상
 - Windows Server 2016 R2 AMI 2017.11.19 이상
 - Windows Server 2019
- 에이전트를 시작합니다. 인스턴스 시작 시 에이전트를 시작하도록 사용자 데이터를 사용하는 것이 좋습니다. 또는 에이전트를 수동으로 시작할 수도 있습니다.

권장 사항

- 최대 절전 모드에서 인스턴스 메모리가 루트 볼륨에 저장되므로 암호화된 Amazon EBS 볼륨을 루트 볼륨으로 사용해야 합니다. 이렇게 해야 데이터가 볼륨에 저장되어 있고 인스턴스와 볼륨 간을 이동 중일 때 메모리(RAM)의 콘텐츠가 암호화됩니다. 루트 볼륨이 암호화된 Amazon EBS 볼륨인지 확인하려면 다음 세 가지 옵션 중 하나를 사용합니다.
 - EBS ‘단일 단계’ 암호화: 단일 실행 인스턴스 API 호출의 경우 암호화되지 않은 AMI에서 암호화된 EBS 지원 EC2 인스턴스를 시작할 수 있습니다. 자세한 내용은 [EBS 지원 AMI를 통한 암호화 사용 \(p. 98\)](#) 단원을 참조하십시오.
 - 기본적인 EBS 암호화: 기본적으로 EBS 암호화를 활성화하여 AWS 계정에서 생성된 모든 새 EBS 볼륨이 암호화되도록 할 수 있습니다. 자세한 내용은 [암호화 기본 제공 \(p. 1036\)](#) 단원을 참조하십시오.
 - 암호화된 AMI: 암호화된 AMI를 사용하여 인스턴스를 시작하는 방식으로 EBS 암호화를 활성화할 수 있습니다. AMI에 암호화된 루트 스냅샷이 없을 경우, 이를 새로운 AMI 및 요청 암호화에 복사할 수 있습니다. 자세한 내용은 [복사 중에 암호화되지 않은 이미지 암호화 \(p. 102\)](#) 및 [AMI 복사 \(p. 107\)](#) 단원을 참조하십시오.

스팟 인스턴스가 스팟 서비스에 의해 최대 절전 모드로 전환되어도 루트 볼륨에서 EBS 볼륨과 인스턴스 메모리(RAM)이 유지됩니다. 인스턴스의 프라이빗 IP 주소 역시 유지됩니다. 인스턴스 스토리지 볼륨과 다른 탄력적 IP 주소가 아닌 퍼블릭 IP 주소는 유지되지 않습니다. 인스턴스가 최대 절전 모드에 있는 동안에는 EBS 볼륨에 대한 요금만 부과됩니다. EC2 집합 및 스팟 집합을 사용하는 경우 최대 절전 모드로 전환된 인스턴스가 많으면 해당 계정의 EBS 볼륨 수 제한을 초과할 수 있습니다.

인스턴스가 스팟 서비스에서 신호를 수신할 때 에이전트는 운영 체제에 최대 절전 모드로 전환하라는 메시지를 보냅니다. 에이전트가 설치되어 있지 않거나 기반 운영 체제가 최대 절전 모드를 지원하지 않거나 인스턴스 메모리를 저장하기 충분한 볼륨 공간이 없는 경우에는 최대 절전 모드로의 전환이 실패하고 대신에 스팟 서비스가 인스턴스를 중지합니다.

스팟 서비스가 스팟 인스턴스를 최대 절전 모드로 전환하면 스팟 인스턴스가 종단되기 2분 전에 종지 공지를 수신합니다. 최대 절전 모드가 즉시 시작됩니다. 인스턴스가 최대 절전 모드를 활성화하는 동안에는 인스턴스 상태 확인이 실패할 수 있습니다. 최대 절전 모드 활성화가 완료되면 인스턴스의 상태는 `stopped`가 됩니다.

최대 절전 모드 스팟 인스턴스 다시 시작

스팟 서비스에 의해 최대 절전 모드로 전환된 스팟 인스턴스는 해당 스팟 서비스에 의해서만 재개될 수 있습니다. 지정된 최고 가격보다 낮은 스팟 가격에서 용량이 가용 상태가 되면 스팟 서비스가 인스턴스를 재개합니다.

자세한 내용은 [인스턴스 최대 절전 모드 준비 \(p. 314\)](#) 단원을 참조하십시오.

온디맨드 인스턴스 최대 절전 모드 전환에 대한 자세한 내용은 [Windows 인스턴스를 최대 절전 모드로 실행 \(p. 443\)](#) 단원을 참조하십시오.

종단 동작 지정

종단 동작을 지정하지 않으면 스팟 인스턴스가 종단되면 기본적으로 종료됩니다. 스팟 요청을 생성할 때 종단 동작을 지정할 수 있습니다. 종단 동작을 지정하는 방법은 스팟 인스턴스를 요청하는 방법에 따라 다릅니다.

[인스턴스 시작 마법사 \(p. 376\)](#)를 사용하여 스팟 인스턴스를 요청하는 경우 종단 동작을 다음과 같이 지정할 수 있습니다. 영구 요청 확인란을 선택한 다음 종단 동작에서 종단 동작을 선택합니다.

[스팟 콘솔 \(p. 277\)](#)을 사용하여 스팟 인스턴스를 요청하는 경우 종단 동작을 다음과 같이 지정할 수 있습니다. 목표 용량 유지 확인란을 선택한 다음 종단 동작에서 종단 동작을 선택합니다.

[시작 템플릿 \(p. 383\)](#)에서 스팟 인스턴스를 구성하는 경우 종단 동작을 다음과 같이 지정할 수 있습니다. 시작 템플릿에서 고급 세부 정보를 확장하고 스팟 인스턴스 요청 확인란을 선택합니다. 사용자 지정을 선택한 다음 종단 동작에서 종단 동작을 선택합니다.

`request-spot-fleet` CLI를 사용할 때 시작 구성에서 스팟 인스턴스를 구성하는 경우 종단 동작을 다음과 같이 지정할 수 있습니다. `InstanceInterruptionBehavior`에 종단 동작을 지정합니다.

`request-spot-instances` CLI를 사용하여 스팟 인스턴스를 구성하는 경우 종단 동작을 다음과 같이 지정할 수 있습니다. `--instance-interruption-behavior`에 종단 동작을 지정합니다.

종단 준비

스팟 인스턴스 사용 시 따라야 할 몇 가지 모범 사례는 다음과 같습니다.

- 기본 최고 가격인 온디맨드 가격을 사용하십시오.
- 필수 소프트웨어 구성이 포함된 Amazon 머신 이미지(AMI)를 사용하여 요청이 이행되는 즉시 인스턴스를 실행할 준비가 되었는지 확인합니다. 시작 시 사용자 데이터를 사용하여 명령을 실행할 수도 있습니다.
- 스팟 인스턴스가 종료되어도 영향을 받지 않을 장소에 중요한 데이터를 정기적으로 저장하십시오. 예를 들어, Amazon S3, Amazon EBS 또는 DynamoDB를 사용할 수 있습니다.

- 작업을 작은 부분으로 분리하거나(눈금, 하둡 또는 대기열 기반 아키텍처 사용), 작업을 자주 저장할 수 있도록 검사점을 사용합니다.
- 스팟 인스턴스 중단 공지를 사용하여 스팟 인스턴스의 상태를 모니터링합니다.
- 이 경고를 즉시 제공하기 위해 모든 노력을 기울이고 있지만 경고를 보내기 전에 스팟 인스턴스가 종료될 수도 있습니다. 따라서 중단 공지를 테스트할 때도 애플리케이션을 테스트하여 예상치 않은 인스턴스 종료가 정상적으로 처리되는지 확인하십시오. 이렇게 하려면 온디맨드 인스턴스를 사용하여 애플리케이션을 실행한 다음 온디맨드 인스턴스를 직접 종료합니다.

인스턴스 최대 절전 모드 준비

이미 에이전트를 포함하고 있는 AMI를 사용하지 않는 한, 인스턴스에 최대 절전 모드 에이전트를 설치해야 합니다. 에이전트가 AMI에 포함되었거나 직접 설치했는지 여부와 관계없이 인스턴스 시작 시 에이전트를 실행해야 합니다.

다음 절차를 따르면 Windows 인스턴스를 준비하는 데 도움이 됩니다. Linux 인스턴스를 준비하는 방법은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스 최대 절전 모드 준비](#)를 참조하십시오.

Windows 인스턴스를 준비하려면

- AMI에 에이전트가 포함되어 있지 않은 경우에는 Windows 인스턴스의 C:\Program Files\Amazon\Hibernate 폴더로 아래 파일을 다운로드합니다.
 - [EC2HibernateAgent.exe](#)
 - [EC2HibernateAgent.ps1](#)
 - [LICENSE.txt](#)
- 사용자 데이터에 다음 명령을 추가합니다.

```
<powershell>."C:\Program Files\Amazon\Hibernate\EC2HibernateAgent.exe"</powershell>
```

스팟 인스턴스 중단 공지

스팟 인스턴스 중단을 정상적으로 처리하는 가장 좋은 방법은 내결합성이 있도록 애플리케이션을 설계하는 것입니다. 이를 위해 스팟 인스턴스 중단 공지를 활용할 수 있습니다. 스팟 인스턴스 중단 공지는 Amazon EC2가 스팟 인스턴스를 종지 또는 종료하기 2분 전에 생성되는 경고입니다. 최대 절전을 종지 행동으로 지정할 경우 종지 공지를 수신하지만, 최대 절전 과정은 즉시 시작되므로 2분 경고를 받지 않습니다.

5초마다 이러한 중단 공지를 확인하는 것이 좋습니다.

이 중단 공지를 CloudWatch 이벤트 및 스팟 인스턴스의 [인스턴스 메타데이터 \(p. 576\)](#) 항목으로 사용할 수 있습니다.

EC2 스팟 인스턴스 interruption notice

Amazon EC2에서 스팟 인스턴스를 종단할 때 실제 종단 2분 전에 이벤트를 전송합니다(최대 절전 모드는 즉시 시작하기 때문에 종단 공지를 2분 전에 받지 않는 최대 절전 모드는 제외). 이 이벤트는 Amazon CloudWatch Events를 통해 감지될 수 있습니다. CloudWatch 이벤트에 대한 자세한 내용은 [Amazon CloudWatch Events 사용 설명서](#) 단원을 참조하십시오. 이벤트 규칙을 생성하고 사용하는 방법을 안내하는 자세한 예제는 [Amazon EC2 스팟 인스턴스 종단 공지 활용](#)을 참조하십시오.

다음은 스팟 인스턴스 종단에 대한 이벤트의 예제입니다. 가능한 instance-action 값은 hibernate, stop 및 terminate입니다.

```
{  
    "version": "0",  
    "id": "12345678-1234-1234-1234-123456789012",
```

```
"detail-type": "EC2 Spot Instance Interruption Warning",
"source": "aws.ec2",
"account": "123456789012",
"time": "yyyy-mm-ddThh:mm:ssZ",
"region": "us-east-2",
"resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-1234567890abcdef0"],
"detail": {
    "instance-id": "i-1234567890abcdef0",
    "instance-action": "action"
}
```

instance-action

스팟 인스턴스가 스팟 서비스에서 중지하거나 종료하도록 표시된 경우 [인스턴스 메타데이터 \(p. 576\)](#)에 instance-action 항목이 있습니다. 표시하지 않은 경우에는 이 항목이 없습니다. 다음과 같이 instance-action을 검색할 수 있습니다.

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/spot/instance-action
```

instance-action 항목은 해당 작업과 작업이 이루어지는 대략적 시간(UTC 기준)을 지정합니다.

다음 예제에서는 이 인스턴스가 중지될 시간을 알려줍니다.

```
{"action": "stop", "time": "2017-09-18T08:22:00Z"}
```

다음 예제에서는 이 인스턴스가 종료될 시간을 알려줍니다.

```
{"action": "terminate", "time": "2017-09-18T08:22:00Z"}
```

Amazon EC2가 인스턴스를 중지 또는 종료할 준비가 되지 않거나 사용자가 인스턴스를 직접 종료한 경우 instance-action 항목이 나타나지 않으며 사용자가 이를 검색하려 하면 HTTP 404 오류를 수신하게 됩니다.

termination-time

이 항목은 이전 버전과의 호환성을 위해 보존되며, 그 대신 instance-action을 사용해야 합니다.

스팟 서비스에서 종료할 스팟 인스턴스를 표시한 경우 인스턴스 메타데이터에 termination-time 항목이 있습니다. 표시하지 않은 경우에는 이 항목이 없습니다. 다음과 같이 termination-time을 검색할 수 있습니다.

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/spot/termination-time
```

termination-time 항목은 인스턴스가 종료 신호를 받게 될 대략의 시간(UTC 기준)을 지정합니다. 다음 예를 참조하십시오.

```
2015-01-05T18:02:00Z
```

Amazon EC2가 인스턴스를 종료할 준비가 되지 않거나 사용자가 스팟 인스턴스를 직접 종료한 경우 termination-time 항목이 나타나지 않거나(HTTP 404 오류 수신) 이 항목에 시간 값이 아닌 값이 포함됩니다.

Amazon EC2에서 인스턴스를 종료하지 않으면 요청 상태가 fulfilled로 설정됩니다. termination-time 값은 과거 시점인 원래 예상 시간과 함께 인스턴스 메타데이터에 남습니다.

중단된 스팟 인스턴스 찾기

콘솔의 인스턴스 창에는 스팟 인스턴스를 비롯한 모든 인스턴스가 표시됩니다. 수명 주기 열의 spot 값에 서 스팟 인스턴스를 식별할 수 있습니다. 인스턴스 상태 열은 인스턴스가 pending, running, stopping, stopped, shutting-down 또는 terminated인지 여부를 나타냅니다. 최대 절전 모드의 스팟 인스턴스의 인스턴스 상태는 stopped입니다.

중단된 스팟 인스턴스를 찾으려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다. 오른쪽 상단 모서리에서 열 표시/숨기기 아이콘을 선택하고 인스턴스 속성에서 수명 주기를 선택합니다. 스팟 인스턴스의 경우 수명 주기는 spot입니다.
또는 탐색 창에서 스팟 요청을 선택합니다. 스팟 인스턴스 요청과 스팟 집합 요청을 모두 볼 수 있습니다. 인스턴스의 ID를 보려면 스팟 인스턴스 요청 또는 스팟 집합 요청을 선택하고 인스턴스 탭을 선택합니다. 인스턴스 ID를 선택하여 인스턴스 창에 인스턴스를 표시합니다.
3. 각 스팟 인스턴스에 대해 인스턴스 상태 열에서 상태를 볼 수 있습니다.

중단된 스팟 인스턴스를 찾으려면(AWS CLI)

--filters 파라미터와 함께 [describe-instance](#) 명령을 사용하여 중단된 스팟 인스턴스의 목록을 나열할 수 있습니다. 출력에 인스턴스 ID만 나열하려면 --query 파라미터를 추가합니다.

```
aws ec2 describe-instances \
  --filters Name=instance-lifecycle,Values=spot Name=instance-state-
  name,Values=terminated,stopped \
  --query Reservations[*].Instances[*].InstanceId
```

Amazon EC2가 스팟 인스턴스를 중단했는지 여부 확인

스팟 인스턴스가 중지되었거나, 최대 절전 모드로 전환되었거나 종료된 경우 CloudTrail을 사용하여 Amazon EC2가 스팟 인스턴스를 중단했는지 여부를 확인할 수 있습니다. CloudTrail에서 이벤트 이름 BidEvictedEvent는 Amazon EC2가 스팟 인스턴스를 중단했음을 나타냅니다. CloudTrail 사용에 관한 자세한 내용은 [AWS CloudTrail을 사용하여 Amazon EC2 및 Amazon EBS API 호출 로깅 \(p. 697\)](#) 단원을 참조하십시오.

중단된 스팟 인스턴스에 대한 청구

스팟 인스턴스(스팟 블록에 있지 않음)가 중단되면 다음과 같이 요금이 부과됩니다.

스팟 인스턴스를 중단하는 사람	운영 체제	첫 번째 시간에 중단됨	첫 번째 시간 후 어떤 시간에든 중단됨
사용자가 스팟 인스턴스를 중지하거나 종료하는 경우	Linux(RHEL 및 SUSE 제외) Windows, RHEL, SUSE	사용된 시간(초)에 대해 부과 일부 시간만 사용한 경우 에도 전체 시간에 대해 부과	사용된 시간(초)에 대해 부과 사용한 전체 시간에 대해 부과되며, 중단된 일부 시간에 대해 전체 시간 요금 부과
Amazon EC2가 스팟 인스턴스를 중단하는 경우	Linux(RHEL 및 SUSE 제외) Windows, RHEL, SUSE	무료	사용된 시간(초)에 대해 부과 사용한 전체 시간에 대해 부과되지만, 중단된 일부 시간에 대한 요금은 무료

스팟 블록에 있는 스팟 인스턴스가 중단되면 다음과 같이 요금이 부과됩니다.

스팟 인스턴스를 중단하는 사람	운영 체제	첫 번째 시간에 중단됨	첫 번째 시간 후 어떤 시간에든 중단됨
사용자가 스팟 인스턴스를 중지하거나 종료하는 경우	Linux(RHEL 및 SUSE 제외) Windows, RHEL, SUSE	사용된 시간(초)에 대해 부과 일부 시간만 사용한 경우에도 전체 시간에 대해 부과	사용된 시간(초)에 대해 부과 사용한 전체 시간에 대해 부과되며, 중단된 일부 시간에 대해 전체 시간 요금 부과
Amazon EC2가 스팟 인스턴스를 중단하는 경우	Linux(RHEL 및 SUSE 제외) Windows, RHEL, SUSE	무료 무료	무료 무료

스팟 인스턴스 데이터 피드

스팟 인스턴스 요금을 쉽게 이해할 수 있도록 Amazon EC2에서는 스팟 인스턴스 사용 및 요금을 설명하는 데이터 피드를 제공합니다. 이 데이터 피드는 데이터 피드를 구독할 때 지정하는 Amazon S3 버킷으로 전송됩니다.

일반적으로 데이터 피드 파일은 한 시간에 한 번씩 버킷에 도착하며, 각 사용 시간이 단일 데이터 파일로 설명됩니다. 이 파일은 압축(gzip)된 후 버킷으로 전송됩니다. 파일이 매우 큰 경우 Amazon EC2는 지정된 사용 시간에 대해 여러 개의 파일을 작성할 수 있습니다(예: 압축 전 해당 시간의 파일 콘텐츠가 50MB를 초과하는 경우).

Note

특정 시간 동안 스팟 인스턴스가 없는 경우 해당 시간에 대한 데이터 피드 파일이 수신되지 않습니다.

목차

- 데이터 피드 파일 이름 및 형식 (p. 317)
- Amazon S3 버킷 요구 사항 (p. 318)
- 스팟 인스턴스 데이터 피드 구독 (p. 318)
- 스팟 인스턴스 데이터 피드 삭제 (p. 319)

데이터 피드 파일 이름 및 형식

스팟 인스턴스 데이터 피드 파일 이름은 다음 형식을 사용합니다(UTC 기준 날짜 및 시간).

`bucket-name.s3.amazonaws.com/optional-prefix/aws-account-id.YYYY-MM-DD-HH.n.unique-id.gz`

예를 들어, 버킷 이름이 `my-bucket-name`이고 접두사가 `my-prefix`인 경우 파일 이름은 다음과 같습니다.

`my-bucket-name.s3.amazonaws.com/my-prefix/111122223333.2019-03-17-20.001.pwBdGTJG.gz`

버킷 이름에 대한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 [버킷 이름 지정 규칙](#)을 참조하세요.

스팟 인스턴스 데이터 피드 파일은 탭으로 구분됩니다. 데이터 파일의 각 줄은 1 인스턴스 시간에 해당하며 다음 표에 나열된 필드를 포함합니다.

필드	설명
Timestamp	이 인스턴스 사용량에 대해 청구된 가격을 결정하는 데 사용되는 타임스탬프입니다.
UsageType	청구되는 사용 유형 및 인스턴스 유형입니다. m1.small 스팟 인스턴스의 경우 이 필드는 SpotUsage로 설정됩니다. 다른 모든 인스턴스 유형의 경우 이 필드는 SpotUsage:{instance-type}으로 설정됩니다. 예: SpotUsage:c1.medium.
Operation	청구되는 제품입니다. Linux 스팟 인스턴스의 경우 이 필드는 RunInstances로 설정됩니다. Windows 스팟 인스턴스의 경우 이 필드는 RunInstances:0002로 설정됩니다. 스팟 사용은 가용 영역에 따라 그룹화됩니다.
InstanceID	이 인스턴스 사용량을 생성한 스팟 인스턴스의 ID입니다.
MyBidID	이 인스턴스 사용량을 생성한 스팟 인스턴스 요청의 ID입니다.
MyMaxPrice	이 스팟 인스턴스 요청에 대해 지정된 최고 가격입니다.
MarketPrice	Timestamp 필드에 지정된 시간의 스팟 가격입니다.
Charge	이 인스턴스 사용량에 대해 청구된 가격입니다.
Version	이 레코드에 대해 데이터 피드 파일 이름에 포함된 버전입니다.

Amazon S3 버킷 요구 사항

데이터 피드를 구독하면 데이터 피드 파일을 저장하기 위한 Amazon S3 버킷을 지정해야 합니다. 데이터 피드에 대한 Amazon S3 버킷을 선택하기 전에 다음 사항을 고려하십시오.

- `FULL_CONTROL` 및 `s3:GetBucketAcl` 작업에 대한 권한을 포함하여 버킷에 대한 `s3:PutBucketAcl` 권한이 있어야 합니다.
버킷 소유자인 경우 기본적으로 이 권한이 있습니다. 그렇지 않으면 버킷 소유자가 AWS 계정에 이 권한을 부여해야 합니다.
- 데이터 피드를 구독할 때 이러한 권한으로 버킷 ACL을 업데이트하여 AWS 데이터 피드 계정에 `FULL_CONTROL` 권한을 부여합니다. AWS 데이터 피드 계정은 버킷에 데이터 피드 파일을 쓸 수 있습니다. 필요한 권한이 계정에 없을 경우 데이터 피드 파일을 버킷에 쓸 수 없습니다.

Note

ACL을 업데이트하고 AWS 데이터 피드 계정에 대한 권한을 제거할 경우 데이터 피드 파일을 버킷에 쓸 수 없습니다. 데이터 피드 파일을 수신하려면 데이터 피드를 다시 구독해야 합니다.

- 각 데이터 피드 파일에는 고유의 ACL(버킷용 ACL과는 별도)이 있습니다. 버킷 소유자는 데이터 파일에 대한 `FULL_CONTROL` 권한을 가지고 있습니다. AWS 데이터 피드 계정은 읽기 및 쓰기 권한이 있습니다.
- 데이터 피드 구독을 삭제해도 Amazon EC2에서 버킷 또는 데이터 파일에 대한 AWS 데이터 피드 계정의 읽기 및 쓰기 권한이 제거되지 않습니다. 이러한 권한을 직접 제거해야 합니다.

스팟 인스턴스 데이터 피드 구독

데이터 피드를 구독하려면 다음 `create-spot-datafeed-subscription` 명령을 사용합니다.

```
aws ec2 create-spot-datafeed-subscription \
--bucket my-bucket-name \
[--prefix my-prefix]
```

다음은 예제 출력입니다.

```
{  
    "SpotDatafeedSubscription": {  
        "OwnerId": "111122223333",  
        "Bucket": "my-bucket-name",  
        "Prefix": "my-prefix",  
        "State": "Active"  
    }  
}
```

스팟 인스턴스 데이터 피드 삭제

데이터 피드를 삭제하려면 다음 `delete-spot-datafeed-subscription` 명령을 사용합니다.

```
aws ec2 delete-spot-datafeed-subscription
```

스팟 인스턴스 고려 사항

스팟 인스턴스 요청에는 다음 제한이 적용됩니다.

고려 사항

- [스팟 인스턴스 요청 제한 \(p. 319\)](#)
- [스팟 집합 제한 \(p. 319\)](#)
- [성능 순간 확장 가능 인스턴스 \(p. 320\)](#)

스팟 인스턴스 요청 제한

스팟 인스턴스 요청에 대한 제한은 동적입니다. 새 계정인 경우 리전별 스팟 인스턴스 요청에 대한 초기 기본 제한이 있습니다. 이 제한은 시간이 지남에 따라 증가할 수 있습니다. 스팟 인스턴스 요청을 제출하고 `Max spot instance count exceeded` 오류가 발생하면 스팟 인스턴스 요청에 대한 제한 증가를 요청할 수 있습니다. 현재 제한을 보고 제한 증가를 요청하는 방법은 [Amazon EC2 서비스 할당량 \(p. 1154\)](#) 단원을 참조하십시오.

스팟 인스턴스를 종료하고 스팟 인스턴스 요청을 취소하지 않으면 Amazon EC2가 스팟 인스턴스 종료를 감지하고 요청을 닫을 때까지 해당 요청이 동적 스팟 인스턴스 요청 제한 계산에 반영됩니다.

스팟 집합 제한

스팟 요청 가격 제한, 인스턴스 제한 및 볼륨 제한과 같이 스팟 집합 또는 EC2 집합에서 시작된 인스턴스에 일반적인 Amazon EC2 제한이 적용됩니다. 또한 다음과 같은 제한이 적용됩니다.

- 리전당 활성 스팟 집합 및 EC2 집합의 수: 1,000개*
- 스팟 인스턴스 풀 수(인스턴스 유형과 서브넷의 고유한 조합): 300*
- 시작 사양의 사용자 데이터 크기: 16KB*
- 스팟 집합 또는 EC2 집합당 대상 용량: 10,000
- 리전 내 모든 스팟 집합 및 EC2 집합의 대상 용량: 100,000
- 스팟 집합 요청 또는 EC2 집합 요청으로 리전을 확장할 수 없습니다.
- 스팟 집합 요청 또는 EC2 집합 요청으로 동일한 가용 영역의 서로 다른 서브넷을 확장할 수 없습니다.

대상 용량의 기본 한도 이상이 필요한 경우, AWS 지원 센터 [사례 생성](#) 서식을 작성하여 한도 증가를 요청하십시오. 제한 유형에서 EC2 Fleet(EC2 플릿)을 선택하고 리전을 선택한 다음 Target Fleet Capacity per Fleet (in units)(플릿당 대상 플릿 용량(유닛)) 또는 Target Fleet Capacity per Region (in units)(리전당 대상 플릿 용량(유닛))을 선택하거나 둘 다 선택합니다.

* 이 숫자는 하드 제한입니다. 이러한 제한에 대한 한도 상승을 요청할 수 없습니다.

성능 순간 확장 가능 인스턴스

스팟 인스턴스를 CPU 크레딧 발생에 대한 유휴 시간 없이 즉시 짧은 기간 동안 사용할 계획인 경우 스팟 인스턴스를 [표준 모드 \(p. 141\)](#)로 시작하여 높은 비용 지불을 방지하는 것이 좋습니다.

스팟 인스턴스를 [무제한 모드 \(p. 133\)](#)로 시작하고 CPU를 즉시 버스트하는 경우 버스팅에 대한 잉여 크레딧을 소모하게 됩니다. 인스턴스를 짧은 기간 동안 사용하는 경우 인스턴스에서 잉여 크레딧을 지불할 정도의 CPU 크레딧이 발생할 시간이 없습니다. 인스턴스를 종료할 때 잉여 크레딧에 대한 요금이 청구됩니다.

무제한 모드는 버스팅에 대한 CPU 크레딧이 발생할 정도로 인스턴스 실행이 긴 경우에만 스팟 인스턴스에 적합합니다. 그렇지 않으면 잉여 크레딧 비용을 지불하면 버스트 가능 성능 스팟 인스턴스가 다른 인스턴스를 사용하는 것보다 비용이 많이 듭니다. 자세한 내용은 [무제한 모드 대 고정 CPU 사용 시기 \(p. 134\)](#) 단원을 참조하세요.

시작 크레딧은 효율적인 컴퓨팅 리소스를 제공하여 인스턴스를 구성함으로써 T2 인스턴스에 대한 생산적인 최초 시작 환경을 제공하는 것을 목적으로 합니다. 새 시작 크레딧에 액세스하기 위한 T2 인스턴스의 반복된 시작은 허용되지 않습니다. 지속적인 CPU가 필요한 경우 (일정 기간 동안 유휴 상태로 둠으로써) 크레딧을 얻고, T2 스팟 인스턴스에 [무제한 모드 \(p. 133\)](#)를 사용하거나 전용 CPU를 포함한 인스턴스 유형을 사용할 수 있습니다.

전용 호스트

Amazon EC2 전용 호스트는 완전히 고객 전용인 EC2 인스턴스 용량을 갖춘 물리적 서버입니다. 전용 호스트에서는 Windows Server, Microsoft SQL Server, SUSE 및 Linux Enterprise Server를 포함한 기존 소켓당, 코어당 또는 VM당 소프트웨어 라이선스를 사용할 수 있습니다.

전용 호스트에서 지원되는 구성에 대한 자세한 내용은 [전용 호스트 구성표](#)를 참조하십시오.

목차

- [전용 호스트과 전용 인스턴스의 차이점 \(p. 320\)](#)
- [기존 보유 라이선스 사용 \(p. 321\)](#)
- [전용 호스트 인스턴스 용량 \(p. 321\)](#)
- [전용 호스트 제한 사항 \(p. 322\)](#)
- [요금 및 결제 \(p. 322\)](#)
- [전용 호스트 작업 \(p. 323\)](#)
- [공유된 전용 호스트 작업 \(p. 340\)](#)
- [호스트 복구 \(p. 344\)](#)
- [구성 변경 추적 \(p. 348\)](#)

전용 호스트과 전용 인스턴스의 차이점

전용 호스트와 전용 인스턴스는 모두 사용자 전용 물리적 서버로 Amazon EC2 인스턴스를 시작하는 데 사용할 수 있습니다.

전용 호스트의 인스턴스와 전용 인스턴스는 성능이나 보안상의 차이나 물리적 차이는 없습니다. 그러나 둘 사이에는 약간의 차이가 있습니다. 다음 표에는 전용 호스트와 전용 인스턴스의 주요 차이점 중 일부를 요약하여 설명합니다.

	전용 호스트	전용 인스턴스
결제	호스트 단위 결제	인스턴스 단위 결제
소켓, 코어 및 호스트 ID 표시 여부	소켓 및 물리 코어 수 표시 여부 제공	표시 여부 없음

	전용 호스트	전용 인스턴스
호스트 및 인스턴스 선호도	시간에 따라 지속적으로 동일한 물리 서버에 인스턴스 배포 허용	지원되지 않음
대상 지정 인스턴스 배치	물리 서버 내 인스턴스 배치 방법에 대한 추가 가시성 및 제어 제공	지원되지 않음
자동 인스턴스 복구	지원 자세한 내용은 호스트 복구 (p. 344) 단원을 참조하십시오.	지원
Bring Your Own License(BYOL)	지원	지원되지 않음

기준 보유 라이선스 사용

전용 호스트를 통해 기준 소켓당, 코어당 또는 VM당 소프트웨어 라이선스를 사용할 수 있습니다. 기준 보유 라이선스를 사용하는 경우 자체 라이선스 관리에 대한 책임은 고객에게 있습니다. 하지만 Amazon EC2에는 인스턴스 선호도 및 대상 지정 배치와 같은 라이선스 규정 준수를 유지하는 데 도움이 되는 기능이 있습니다.

다음은 Amazon EC2에서 기준 볼륨 라이선스 머신 이미지를 사용하려면 수행해야 할 일반 단계입니다.

1. 머신 이미지 사용을 제어하는 라이선스 조건이 가상 클라우드 환경에서 사용을 허용하는지 확인합니다. Microsoft 라이선싱에 대한 자세한 정보는 [Amazon Web Services and Microsoft Licensing](#) 단원을 참조하십시오.
2. 머신 이미지를 Amazon EC2 내에서 사용할 수 있음을 확인한 이후 VM Import/Export를 사용하여 가져옵니다. 머신 이미지를 가져오는 방법에 대한 자세한 내용은 [VM Import/Export 사용 설명서](#)를 참조하십시오.
3. 머신 이미지를 가져온 후 이 머신 이미지에서 계정에 있는 활성 전용 호스트로 인스턴스를 시작할 수 있습니다.
4. 이러한 인스턴스를 실행할 때 운영 체제에 따라 자체 KMS 서버(예: Windows Server 또는 Windows SQL Server)에 대해 해당 인스턴스를 활성화해야 할 수 있습니다. 가져온 Windows AMI는 Amazon Windows KMS 서버에 대해 활성화할 수 없습니다.

Note

AWS에서 이미지가 어떻게 사용되는지 추적하려면 AWS Config에서 호스트 기록을 활성화합니다. AWS Config를 사용하여 전용 호스트에 대한 구성 변경을 기록하고 출력을 라이선스 보고용 데이터 스스로 사용할 수 있습니다. 자세한 내용은 [구성 변경 추적 \(p. 348\)](#) 단원을 참조하십시오.

전용 호스트 인스턴스 용량

c5, m5, r5, c5n, r5n 및 m5n 인스턴스 패밀리의 경우 동일한 전용 호스트에서 여러 인스턴스 유형을 지원할 수 있습니다. 예를 들어, r5 전용 호스트를 할당하면 소켓 2개와 물리적 코어 48개가 있는 호스트를 사용할 수 있으며 이 호스트에서 r5.2xlarge 및 r5.4xlarge와 같은 다양한 인스턴스 유형을 실행할 수 있습니다. 호스트와 연결된 코어 용량의 한도 내에서 인스턴스를 몇 개든지 실행할 수 있습니다. 예를 들어, 아래 표는 전용 호스트에서 실행할 수 있는 다양한 인스턴스 유형 조합을 보여 줍니다.

인스턴스 패밀리	인스턴스 유형 조합 예
R5	<ul style="list-style-type: none"> 예 1: 4 x r5.4xlarge + 4 x r5.2xlarge 예 2: 1 x r5.12xlarge + 1 x r5.4xlarge + 1 x r5.2xlarge + 5 x r5.xlarge + 2 x r5.large

인스턴스 패밀리	인스턴스 유형 조합 예
C5	<ul style="list-style-type: none">예 1: 1 x c5.9xlarge + 2 x c5.4xlarge + 1 x c5.xlarge예 2: 4 x c5.4xlarge + 1 x c5.xlarge + 2 x c5.large
M5	<ul style="list-style-type: none">예 1: 4 x m5.4xlarge + 4 x m5.2xlarge예 2: 1 x m5.12xlarge + 1 x m5.4xlarge + 1 x m5.2xlarge + 5 x m5.xlarge + 2 x m5.large

다른 인스턴스 패밀리는 동일한 전용 호스트에서 단일 인스턴스 유형만 지원합니다. 전용 호스트에서 지원되는 인스턴스 패밀리 및 인스턴스 유형에 대한 자세한 내용은 [Amazon EC2 전용 호스트 요금](#)을 참조하십시오.

전용 호스트 제한 사항

전용 호스트를 할당하기 전에 다음 제한 및 제약에 유의하십시오.

- 전용 호스트에서 RHEL, SUSE Linux 및 SQL Server를 실행하려면 자체 AMI를 가져와야 합니다. AWS에서 제공하거나 AWS Marketplace에서 사용할 수 있는 RHEL, SUSE Linux 및 SQL Server AMI는 전용 호스트와 함께 사용할 수 없습니다. 자체 AMI를 만드는 방법에 대한 자세한 내용은 [기존 보유 라이선스 사용 \(p. 321\)](#) 단원을 참조하십시오.
- 리전당 인스턴스 패밀리당 최대 2개의 온디맨드 전용 호스트를 할당할 수 있습니다. [Amazon EC2 전용 호스트 할당 한도 상향 요청](#)에서 한도 상향을 요청할 수 있습니다.
- 전용 호스트에서 실행되는 인스턴스는 VPC에서만 시작할 수 있습니다.
- 호스트 리소스 그룹을 지정하는 시작 템플릿을 사용하는 경우 Auto Scaling 그룹이 지원됩니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [Auto Scaling 그룹을 위한 시작 템플릿 만들기](#)를 참조하십시오.
- Amazon RDS 인스턴스는 지원되지 않습니다.
- AWS 프리 티어는 전용 호스트에서 제공되지 않습니다.
- 인스턴스 배치 제어는 전용 호스트에서 인스턴스 시작을 관리하는 것을 말합니다. 배치 그룹에서는 전용 호스트를 시작할 수 없습니다.

요금 및 결제

전용 호스트의 요금은 결제 옵션에 따라 다릅니다.

결제 옵션

- [온디맨드 전용 호스트 \(p. 322\)](#)
- [전용 호스트 예약 \(p. 323\)](#)
- [Savings Plans \(p. 323\)](#)
- [전용 호스트의 Windows Server에 대한 요금 \(p. 323\)](#)

온디맨드 전용 호스트

계정에 전용 호스트를 할당하면 온디맨드 결제가 자동으로 활성화됩니다.

전용 호스트에 대한 온디맨드 요금은 인스턴스 패밀리 및 리전에 따라 다릅니다. 인스턴스를 시작하기로 선택한 인스턴스의 수량이나 크기에 관계없이 활성 전용 호스트에 대해 초당(최소 60초) 비용을 지불합니다. 온디맨드 요금에 대한 자세한 내용은 [Amazon EC2 전용 호스트 온디맨드 요금](#)을 참조하십시오.

언제든 온디맨드 전용 호스트를 해제하여 이에 대한 요금 청구를 중단시킬 수 있습니다. 전용 호스트 해제에 대한 자세한 내용은 [전용 호스트 해제 \(p. 336\)](#) 단원을 참조하십시오.

전용 호스트 예약

전용 호스트 예약은 실행 중인 온디맨드 전용 호스트와 비교해 청구 할인을 제공합니다. 다음과 같은 세 가지 결제 방식을 통해 예약이 가능합니다.

- **선결제 없음** - 선결제가 없는 예약은 사용 기간 동안 전용 호스트 사용에 대해 할인을 제공하고 선결제가 필요하지 않습니다. 사용 기간이 1년인 경우에만 가능합니다.
- **부분 선결제** - 예약의 일부를 선결제하고, 사용 기간 내 나머지 시간에 대해서는 할인 요금이 청구됩니다. 사용 기간이 1년 및 3년인 경우에 가능합니다.
- **전체 선결제** - 최저 실효 가격을 제공합니다. 사용 기간이 1년 및 3년인 경우에 사용 가능하며, 향후 추가 요금 없이 사용 기간 전체 비용을 커버합니다.

계정에 활성 전용 호스트가 있어야 예약을 구매할 수 있습니다. 각 예약은 계정에서 1개의 특정 전용 호스트에 해당됩니다. 예약은 인스턴스 크기가 아닌 호스트의 인스턴스 패밀리에 적용됩니다. 인스턴스 크기가 서로 다른 세 가지 전용 호스트(m4.xlarge, m4.medium, m4.large)가 있는 경우 단일 m4 예약을 이 모든 전용 호스트와 연결할 수 있습니다. 예약의 인스턴스 패밀리 및 리전은 연결하고자 하는 전용 호스트의 인스턴스 패밀리 및 리전과 일치해야 합니다.

하나의 예약이 전용 호스트와 연결되면 예약 기간이 끝날 때까지 전용 호스트를 해제할 수 없습니다.

예약 요금에 대한 자세한 정보는 [Amazon EC2 전용 호스트 요금](#) 단원을 참조하십시오.

Savings Plans

Savings Plans는 온디맨드 인스턴스를 통해 상당한 비용 절감을 제공하는 유연한 요금 모델입니다. Savings Plans에서는 1년 또는 3년 기간 동안 시간당 USD로 일관적인 사용량을 약정합니다. Savings Plans는 특정 전용 호스트를 약정하는 것이 아니라 사용자의 요구 사항에 가장 적합하고 지속적으로 비용을 절약해 주는 전용 호스트를 사용할 수 있는 유연성을 제공합니다. 자세한 내용은 [AWS Savings Plans 사용 설명서](#)를 참조하십시오.

전용 호스트의 Windows Server에 대한 요금

Microsoft 라이선스 약관에 따라 기존 Windows Server 및 SQL Server 라이선스를 전용 호스트에서 사용할 수 있습니다. 기존 보유 라이선스를 사용하는 경우 소프트웨어 사용에 따른 추가 요금이 없습니다.

또한 Amazon에서 제공하는 Windows Server AMI를 사용하여 전용 호스트에서 최신 Windows Server 버전도 실행할 수 있습니다. 이는 전용 호스트에서 실행할 수 있는 기존 SQL Server 라이선스가 있지만 SQL Server 워크로드를 실행하려면 Windows Server가 필요한 시나리오에서 일반적입니다. Amazon에서 제공하는 Windows Server AMI는 [최신 세대의 인스턴스 유형 \(p. 115\)](#)에서만 지원됩니다. 자세한 내용은 [Amazon EC2 전용 호스트 요금](#)을 참조하십시오.

전용 호스트 작업

전용 호스트를 사용하려면 먼저 계정에서 사용할 호스트를 할당해야 합니다. 그런 다음, 인스턴스에 대해 호스트 테넌시를 지정하여 호스트에서 인스턴스를 시작합니다. 인스턴스를 시작할 특정 호스트를 선택해야 합니다. 또는 자동 배치가 활성화되었고 인스턴스 유형이 일치하는 모든 호스트에서 시작하도록 허용할 수 있습니다. 인스턴스를 중지했다 다시 시작하는 경우 호스트 선호도 설정이 해당 인스턴스를 동일한 또는 다른 호스트에서 다시 시작할지 여부를 결정합니다.

온디맨드 호스트가 더 이상 필요하지 않을 경우 해당 호스트에서 실행 중인 인스턴스를 중지하고 다른 호스트에서 시작하도록 지시한 후 호스트를 해제합니다.

전용 호스트는 AWS License Manager와도 통합됩니다. License Manager를 사용하면 단일 개체로 관리되는 전용 호스트 모음인 호스트 리소스 그룹을 만들 수 있습니다. 호스트 리소스 그룹을 만들 때 auto-allocate,

auto-release와 같은 호스트 관리 기본 설정을 전용 호스트에 지정합니다. 그러면 수동으로 호스트를 할당하고 관리할 필요 없이 전용 호스트에서 인스턴스를 시작할 수 있습니다. 자세한 내용은 AWS License Manager 사용 설명서의 [호스트 리소스 그룹](#)을 참조하십시오.

목차

- [전용 호스트 할당 \(p. 324\)](#)
- [전용 호스트에서 인스턴스 시작 \(p. 326\)](#)
- [호스트 리소스 그룹에서 인스턴스 시작 \(p. 328\)](#)
- [자동 배치 및 선호도의 이해 \(p. 329\)](#)
- [전용 호스트 자동 배치 수정 \(p. 329\)](#)
- [지원되는 인스턴스 유형 수정 \(p. 330\)](#)
- [인스턴스 테넌시 및 선호도 수정 \(p. 332\)](#)
- [전용 호스트 보기 \(p. 333\)](#)
- [전용 호스트 태그 지정 \(p. 334\)](#)
- [전용 호스트 모니터링 \(p. 335\)](#)
- [전용 호스트 해제 \(p. 336\)](#)
- [전용 호스트 예약 구입 \(p. 337\)](#)
- [전용 호스트 예약 보기 \(p. 339\)](#)
- [전용 호스트 예약 태그 지정 \(p. 339\)](#)

전용 호스트 할당

전용 호스트 사용을 시작하려면 Amazon EC2 콘솔 또는 명령줄 도구를 사용하여 계정에서 전용 호스트를 할당해야 합니다. 전용 호스트를 할당한 후에는 전용 호스트 용량을 계정에서 즉시 사용할 수 있으며 전용 호스트에서 인스턴스를 시작할 수 있습니다.

c5, m5, r5, c5n, r5n 및 m5n 인스턴스 패밀리의 경우 동일한 전용 호스트에서 여러 인스턴스 유형을 지원할 수 있습니다. 다른 인스턴스 패밀리는 동일한 전용 호스트에서 단일 인스턴스 유형만 지원합니다.

다음 방법을 사용하여 전용 호스트를 할당할 수 있습니다.

새로운 콘솔

전용 호스트를 할당하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택한 후 전용 호스트 할당을 선택합니다.
3. 인스턴스 패밀리에서 전용 호스트에 대한 인스턴스 패밀리를 선택합니다.
4. 전용 호스트가 선택한 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하는지 또는 특정 인스턴스 유형만 지원하는지를 지정합니다. 다음 중 하나를 수행하십시오.
 - 선택한 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하도록 전용 호스트를 구성하려면 Support multiple instance types(여러 인스턴스 유형 지원)에서 활성화를 선택합니다. 이 항목을 활성화하면 동일한 인스턴스 패밀리의 다양한 인스턴스 유형을 전용 호스트에서 시작할 수 있습니다. 예를 들어 m5 인스턴스 패밀리를 선택하고 이 옵션을 선택하는 경우 m5.xlarge 및 m5.4xlarge 인스턴스를 전용 호스트에서 시작할 수 있습니다.
 - 선택한 인스턴스 패밀리 내의 단일 인스턴스 유형을 지원하도록 전용 호스트를 구성하려면 여러 인스턴스 유형 지원의 선택을 취소한 다음 인스턴스 유형에서 지원할 인스턴스 유형을 선택합니다. 이렇게 하면 전용 호스트에서 단일 인스턴스 유형을 시작할 수 있습니다. 예를 들어 이 옵션을 선택하고 m5.4xlarge를 지원되는 인스턴스 유형으로 선택하는 경우 m5.4xlarge 인스턴스만 전용 호스트에서 시작할 수 있습니다.
5. 가용 영역에서 전용 호스트를 할당할 가용 영역을 선택합니다.

6. 전용 호스트가 인스턴스 유형과 일치하는 대상이 지정되지 않은 인스턴스 시작을 허용하게 하려면 인스턴스 자동 배치에서 활성화를 선택합니다. 자동 배치에 대한 자세한 정보는 [자동 배치 및 선호도의 이해 \(p. 329\)](#) 단원을 참조하십시오.
7. 전용 호스트의 호스트 복구를 사용하려면 Host recovery(호스트 복구)에서 활성화를 선택합니다. 자세한 내용은 [호스트 복구 \(p. 344\)](#) 단원을 참조하십시오.
8. 수량에 할당할 전용 호스트 수를 입력합니다.
9. (선택 사항) 태그 추가를 선택하고 태그 키와 태그 값을 입력합니다.
10. [Allocate]를 선택합니다.

기존 콘솔

전용 호스트를 할당하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트, 전용 호스트 할당을 선택합니다.
3. 인스턴스 패밀리에서 전용 호스트에 대한 인스턴스 패밀리를 선택합니다.
4. 전용 호스트가 선택한 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하는지 또는 특정 인스턴스 유형만 지원하는지를 지정합니다. 다음 중 하나를 수행하십시오.
 - 선택한 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하도록 전용 호스트를 구성하려면 여러 인스턴스 유형 지원을 선택합니다. 이 항목을 활성화하면 동일한 인스턴스 패밀리의 다양한 인스턴스 유형을 전용 호스트에서 시작할 수 있습니다. 예를 들어 m5 인스턴스 패밀리를 선택하고 이 옵션을 선택하는 경우 m5.xlarge 및 m5.4xlarge 인스턴스를 전용 호스트에서 시작할 수 있습니다. 인스턴스 패밀리는 Nitro 시스템으로 구동되어야 합니다.
 - 선택한 인스턴스 패밀리 내의 단일 인스턴스 유형을 지원하도록 전용 호스트를 구성하려면 여러 인스턴스 유형 지원의 선택을 취소한 다음 인스턴스 유형에서 지원할 인스턴스 유형을 선택합니다. 이렇게 하면 전용 호스트에서 단일 인스턴스 유형을 시작할 수 있습니다. 예를 들어 이 옵션을 선택하고 m5.4xlarge를 지원되는 인스턴스 유형으로 선택하는 경우 m5.4xlarge 인스턴스만 전용 호스트에서 시작할 수 있습니다.
5. 가용 영역에서 전용 호스트를 할당할 가용 영역을 선택합니다.
6. 전용 호스트가 인스턴스 유형과 일치하는 대상이 지정되지 않은 인스턴스 시작을 허용하게 하려면 인스턴스 자동 배치에서 활성화를 선택합니다. 자동 배치에 대한 자세한 정보는 [자동 배치 및 선호도의 이해 \(p. 329\)](#) 단원을 참조하십시오.
7. 전용 호스트의 호스트 복구를 사용하려면 Host recovery(호스트 복구)에서 Enable(활성화)을 선택합니다. 자세한 내용은 [호스트 복구 \(p. 344\)](#) 단원을 참조하십시오.
8. 수량에 할당할 전용 호스트 수를 입력합니다.
9. (선택 사항) 태그 추가를 선택하고 태그 키와 태그 값을 입력합니다.
10. 호스트 할당을 선택합니다.

AWS CLI

전용 호스트를 할당하려면

`allocate-hosts` AWS CLI 명령을 사용합니다. 다음 명령은 us-east-1a 가용 영역에서 m5 인스턴스 패밀리의 여러 인스턴스 유형을 지원하는 전용 호스트를 할당합니다. 또한 호스트에서 호스트 복구가 활성화되고 자동 배치가 비활성화됩니다.

```
aws ec2 allocate-hosts --instance-family "m5" --availability-zone "us-east-1a" --auto-placement "off" --host-recovery "on" --quantity 1
```

다음 명령은 eu-west-1a 가용 영역에서 대상 지정되지 않은 m4.large 인스턴스를 지원하는 전용 호스트를 할당하고 호스트 복구를 활성화하며 키 purpose 및 값 production과 함께 태그를 적용합니다.

```
aws ec2 allocate-hosts --instance-type "m4.large" --availability-zone "eu-west-1a"  
--auto-placement "on" --host-recovery "on" --quantity 1 --tag-specifications  
'ResourceType=dedicated-host',Tags=[{Key=purpose,Value=production}]'
```

PowerShell

전용 호스트를 할당하려면

New-EC2Host Windows PowerShell용 AWS 도구 명령을 사용합니다. 다음 명령은 us-east-1a 가용 영역에서 m5 인스턴스 패밀리의 여러 인스턴스 유형을 지원하는 전용 호스트를 할당합니다. 또한 호스트에서 호스트 복구가 활성화되고 자동 배치가 비활성화됩니다.

```
PS C:\> New-EC2Host -InstanceFamily m5 -AvailabilityZone us-east-1a -AutoPlacement Off  
-HostRecovery On -Quantity 1
```

다음 명령은 eu-west-1a 가용 영역에서 대상 지정되지 않은 m4.large 인스턴스를 지원하는 전용 호스트를 할당하고 호스트 복구를 활성화하며 키 purpose 및 값 production과 함께 태그를 적용합니다.

생성 시 전용 호스트를 태그 지정하는 데 사용된 TagSpecification 파라미터를 사용하려면 태그 지정될 리소스 유형, 태그 키 및 태그 값을 지정하는 객체가 필요합니다. 다음 명령은 필요 객체를 생성합니다.

```
PS C:\> $tag = @{ Key="purpose"; Value="production" }  
PS C:\> $tagspec = new-object Amazon.EC2.Model.TagSpecification  
PS C:\> $tagspec.ResourceType = "dedicated-host"  
PS C:\> $tagspec.Tags.Add($tag)
```

다음 명령은 전용 호스트를 할당하고 \$tagspec 객체에 지정된 태그를 적용합니다.

```
PS C:\> New-EC2Host -InstanceType m4.large -AvailabilityZone eu-west-1a -  
AutoPlacement On -HostRecovery On -Quantity 1 -TagSpecification $tagspec
```

전용 호스트에서 인스턴스 시작

전용 호스트를 할당한 후 여기에서 인스턴스를 시작할 수 있습니다. 시작하려는 인스턴스 유형에 사용 가능한 충분한 용량을 갖춘 활성 전용 호스트가 없는 경우 host 테넌시의 인스턴스를 시작할 수 없습니다.

Note

전용 호스트에서 시작되는 인스턴스는 VPC에서만 시작할 수 있습니다. 자세한 정보는 [VPC 소개](#) 단원을 참조하십시오.

인스턴스를 시작하기 전에 제한 사항에 유의하십시오. 자세한 내용은 [전용 호스트 제한 사항 \(p. 322\)](#) 단원을 참조하십시오.

다음 방법을 사용하여 전용 호스트에서 인스턴스를 시작할 수 있습니다.

Console

전용 호스트 페이지의 특정 전용 호스트에서 인스턴스를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.

3. 전용 호스트 페이지에서 호스트를 선택하고 작업, 호스트에서 인스턴스 시작을 선택합니다.
4. 목록에서 AMI를 선택합니다. SQL Server, SUSE 및 Amazon EC2에서 제공하는 RHEL AMI는 전용 호스트와 함께 사용할 수 없습니다.
5. 인스턴스 유형 선택 페이지에서 시작할 인스턴스 유형을 선택하고 다음: 인스턴스 세부 정보 구성을 선택합니다.

전용 호스트가 단일 인스턴스 유형만 지원하는 경우 지원되는 인스턴스 유형이 기본적으로 선택되고 변경할 수 없습니다.

전용 호스트가 여러 인스턴스 유형을 지원하는 경우 전용 호스트의 가용 인스턴스 용량을 기반으로 지원되는 인스턴스 패밀리 내에서 인스턴스 유형을 선택해야 합니다. 크기가 큰 인스턴스부터 시작한 다음 필요에 따라 더 작은 인스턴스로 나머지 인스턴스 용량을 채우는 것이 좋습니다.

6. 인스턴스 세부 정보 구성 페이지에서 필요에 따라 인스턴스 설정을 구성하고 선호도에서 다음 옵션 중 하나를 선택합니다.
 - 비활성 - 인스턴스가 지정된 호스트에서 시작하지만, 종지될 경우 반드시 동일한 전용 호스트에서 다시 시작하지는 않습니다.
 - 호스트 - 종지된 경우 인스턴스가 항상 특정 호스트에서 다시 시작합니다.

선호도에 대한 자세한 내용은 [자동 배치 및 선호도의 이해 \(p. 329\)](#) 단원을 참조하십시오.

테넌시 및 호스트 옵션은 선택한 호스트에 따라 사전 구성됩니다.

7. 검토 및 시작을 선택합니다.
8. 인스턴스 시작 검토 페이지에서 시작을 선택합니다.
9. 메시지가 표시될 때 기존 키 페어를 선택하거나 새 키 페어를 생성한 후 인스턴스 시작을 선택합니다.

시작 인스턴스 마법사를 사용하여 전용 호스트에서 인스턴스를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스, 인스턴스 시작을 선택합니다.
3. 목록에서 AMI를 선택합니다. SQL Server, SUSE 및 Amazon EC2에서 제공하는 RHEL AMI는 전용 호스트와 함께 사용할 수 없습니다.
4. 시작할 인스턴스 유형을 선택하고 Next: Configure Instance Details(다음: 인스턴스 세부 정보 구성)을 클릭합니다.
5. Configure Instance Details(인스턴스 세부 정보 구성) 페이지에서 필요에 따라 인스턴스 설정을 구성한 후 다음과 같은 전용 호스트 설정을 구성합니다.
 - 테넌시 - 전용 호스트 - 전용 호스트에서 이 인스턴스 시작을 선택합니다.
 - 호스트 - 자동 배치 사용을 선택하여 자동 배치가 활성화된 전용 호스트에서 인스턴스를 시작하거나, 목록에서 특정 전용 호스트를 선택합니다. 이 목록에는 선택한 인스턴스 유형을 지원하는 전용 호스트만 표시됩니다.
 - 선호도 - 다음 옵션 중 하나를 선택합니다.
 - 비활성 - 인스턴스가 지정된 호스트에서 시작하지만, 종지될 경우 반드시 그 호스트에서 다시 시작하지는 않습니다.
 - 호스트 - 종지된 경우 인스턴스가 항상 지정된 호스트에서 다시 시작합니다.

자세한 내용은 [자동 배치 및 선호도의 이해 \(p. 329\)](#) 단원을 참조하십시오.

이러한 설정이 보이지 않으면 네트워크 메뉴에서 VPC를 선택했는지 확인하십시오.

6. 검토 및 시작을 선택합니다.
7. 인스턴스 시작 검토 페이지에서 시작을 선택합니다.

8. 메시지가 표시될 때 기존 키 페어를 선택하거나 새 키 페어를 생성한 후 인스턴스 시작을 선택합니다.

AWS CLI

전용 호스트에서 인스턴스를 시작하려면

`run-instances` AWS CLI 명령을 사용하고 `Placement` 요청 파라미터 내에서 인스턴스 선호도, 테넌시 및 호스트를 지정합니다.

PowerShell

전용 호스트에서 인스턴스를 시작하려면

`New-EC2Instance` Windows PowerShell용 AWS 도구 명령을 사용하고 `Placement` 요청 파라미터 내에서 인스턴스 선호도, 테넌시 및 호스트를 지정합니다.

호스트 리소스 그룹에서 인스턴스 시작

가용 인스턴스 용량이 있는 전용 호스트가 포함된 호스트 리소스 그룹에서 인스턴스를 시작하면 Amazon EC2가 그 호스트에서 인스턴스를 시작합니다. 가용 인스턴스 용량이 있는 호스트가 호스트 리소스 그룹에 없으면 Amazon EC2가 호스트 리소스 그룹에 새 호스트를 자동으로 할당하고 그 호스트에서 인스턴스를 시작합니다. 자세한 내용은 AWS License Manager 사용 설명서의 [호스트 리소스 그룹](#)을 참조하십시오.

요구 사항 및 제한

- 코어 또는 소켓 기반 라이선스 구성의 AMI와 연결해야 합니다.
- Amazon EC2에서 제공하는 SQL Server, SUSE 및 RHEL AMI는 전용 호스트와 함께 사용할 수 없습니다.
- 호스트 ID를 선택하여 특정 호스트를 대상으로 지정할 수 없으며, 호스트 리소스 그룹에서 인스턴스를 시작할 때 인스턴스 선호도를 활성화할 수 없습니다.

다음 방법을 사용하여 호스트 리소스 그룹에서 인스턴스를 시작할 수 있습니다.

Console

호스트 리소스 그룹에서 인스턴스를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스, 인스턴스 시작을 선택합니다.
3. AMI를 선택합니다.
4. 시작할 인스턴스 유형을 선택하고 Next: Configure Instance Details(다음: 인스턴스 세부 정보 구성)을 클릭합니다.
5. 인스턴스 세부 정보 구성 페이지에서 필요에 따라 인스턴스 설정을 구성한 후 다음을 수행합니다.
 - a. 테넌시에서 전용 호스트를 설정합니다.
 - b. Host resource group(호스트 리소스 그룹)에서 Launch instance into a host resource group(호스트 리소스 그룹에서 인스턴스 시작)을 선택합니다.
 - c. Host resource group name(호스트 리소스 그룹 이름)에서 인스턴스를 시작할 호스트 리소스 그룹을 선택합니다.
6. 검토 및 시작을 선택합니다.
7. 인스턴스 시작 검토 페이지에서 시작을 선택합니다.
8. 메시지가 표시될 때 기존 키 페어를 선택하거나 새 키 페어를 생성한 후 인스턴스 시작을 선택합니다.

AWS CLI

호스트 리소스 그룹에서 인스턴스를 시작하려면

`run-instances` AWS CLI 명령을 사용하고 `Placement` 요청 파라미터 내에서 테넌시 옵션은 생략하고 호스트 리소스 그룹 ARN을 지정합니다.

PowerShell

호스트 리소스 그룹에서 인스턴스를 시작하려면

`New-EC2Instance` Windows PowerShell용 AWS 도구 명령을 사용하고 `Placement` 요청 파라미터 내에서 테넌시 옵션은 생략하고 호스트 리소스 그룹 ARN을 지정합니다.

자동 배치 및 선호도의 이해

전용 호스트에 대한 배치 제어는 인스턴스 수준과 호스트 수준에서 모두 수행됩니다.

자동 배치

자동 배치는 호스트 수준에서 구성됩니다. 자동 배치에서는 시작하는 인스턴스가 특정 호스트에서 시작되는지 또는 일치하는 구성이 있는 모든 가용 호스트에서 시작되는지를 선택할 수 있습니다.

전용 호스트의 자동 배치가 비활성화된 경우에는 고유한 호스트 ID를 지정한 호스트 테넌시 인스턴스 시작만 허용합니다. 이것이 새로운 전용 호스트에 대한 기본 설정입니다.

전용 호스트의 자동 배치가 활성화된 경우에는, 인스턴스 유형 구성이 일치하며 대상 지정되지 않은 인스턴스 시작을 허용합니다.

인스턴스를 시작할 때 테넌시를 구성해야 합니다. 특정 `HostId`를 입력하지 않고 전용 호스트에서 인스턴스를 시작하면 자동 배치가 활성화되고 인스턴스 유형이 일치하는 모든 전용 호스트에서 시작할 수 있습니다.

호스트 선호도

호스트 선호도는 인스턴스 수준에서 구성되어야 합니다. 인스턴스와 전용 호스트 간에 시작 관계를 설정합니다.

선호도를 `Host`로 설정하면 특정 호스트에서 시작한 인스턴스가 종단된 경우 항상 동일한 호스트에서 다시 시작합니다. 대상 지정 및 대상 미지정 시작에 모두 적용됩니다.

선호도가 `off`로 설정된 상태에서 인스턴스를 중지했다 다시 시작하는 경우 모든 사용 가능한 호스트에서 다시 시작할 수 있습니다. 하지만 인스턴스는 마지막으로 실행되던 전용 호스트에서 다시 시작하려고 시도합니다(최대한 노력).

전용 호스트 자동 배치 수정

AWS 계정에 할당한 후 다음 방법 중 하나를 사용하여 전용 호스트의 자동 배치 설정을 수정할 수 있습니다.

새로운 콘솔

전용 호스트의 자동 배치를 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 호스트를 선택하고 작업, `Modify host(호스트 수정)`를 선택합니다.
4. `Instance auto-placement`(인스턴스 자동 배치)에서 활성화를 선택하여 자동 배치를 활성화하거나 활성화를 선택 취소하여 자동 배치를 비활성화합니다. 자세한 내용은 [자동 배치 및 선호도의 이해 \(p. 329\)](#) 단원을 참조하십시오.

- 저장을 선택합니다.

기존 콘솔

전용 호스트의 자동 배치를 수정하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 전용 호스트를 선택합니다.
- 전용 호스트 페이지에서 호스트를 선택하고 작업을 선택한 후 자동 배치 수정을 선택합니다.
- 자동 배치 수정 창의 인스턴스 자동 배치 허용에서 예를 선택하면 자동 배치가 활성화되고, 아니요를 선택하면 자동 배치가 비활성화됩니다. 자세한 내용은 [자동 배치 및 선호도의 이해 \(p. 329\)](#) 단원을 참조하십시오.
- 저장을 선택합니다.

AWS CLI

전용 호스트의 자동 배치를 수정하려면

`modify-hosts` AWS CLI 명령을 사용합니다. 다음 예는 지정된 전용 호스트에 대한 자동 배치를 활성화합니다.

```
aws ec2 modify-hosts --auto-placement on --host-ids h-012a3456b7890cdef
```

PowerShell

전용 호스트의 자동 배치를 수정하려면

`Edit-EC2Host` Windows PowerShell용 AWS 도구 명령을 사용합니다. 다음 예는 지정된 전용 호스트에 대한 자동 배치를 활성화합니다.

```
PS C:\> Edit-EC2Host --AutoPlacement 1 --HostId h-012a3456b7890cdef
```

지원되는 인스턴스 유형 수정

c5, m5, r5, c5n, r5n 및 m5n 인스턴스 패밀리의 경우 동일한 전용 호스트에서 여러 인스턴스 유형을 지원할 수 있습니다. 다른 인스턴스 패밀리는 동일한 전용 호스트에서 단일 인스턴스 유형만 지원합니다.

다음 방법을 사용하여 전용 호스트를 할당할 수 있습니다.

전용 호스트를 수정하여 지원되는 인스턴스 유형을 변경할 수 있습니다. 현재 단일 인스턴스 유형을 지원하는 경우 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하도록 수정할 수 있습니다. 마찬가지로, 현재 여러 인스턴스 유형을 지원하는 경우 특정 인스턴스 유형만 지원하도록 수정할 수 있습니다.

여러 인스턴스 유형을 지원하도록 전용 호스트를 수정하려면 먼저 호스트에서 실행 중인 모든 인스턴스를 종지해야 합니다. 수정을 완료하려면 10분 정도 걸립니다. 수정이 진행 중인 동안 전용 호스트는 pending 상태로 전환됩니다. pending 상태에 있는 동안에는 전용 호스트에서 종지된 인스턴스를 시작하거나 새 인스턴스를 시작할 수 없습니다.

여러 인스턴스 유형을 지원하는 전용 호스트를 단일 인스턴스 유형만 지원하도록 수정하려면 호스트에 실행 중인 인스턴스가 없거나 실행 중인 인스턴스가 호스트에서 지원하려는 인스턴스 유형이어야 합니다. 예를 들어, m5 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하는 호스트를 m5.large 인스턴스만 지원하도록 수정하려면 전용 호스트에 실행 중인 인스턴스가 없거나 m5.large 인스턴스만 실행 중이어야 합니다.

다음 방법 중 하나를 사용하여 지원되는 인스턴스 유형을 수정할 수 있습니다.

새로운 콘솔

전용 호스트에 지원되는 인스턴스 유형을 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
 2. 탐색 창에서 전용 호스트를 선택합니다.
 3. 수정할 전용 호스트를 선택하고 작업, Modify host(호스트 수정)을 선택합니다.
 4. 전용 호스트의 현재 구성에 따라 다음 중 하나를 수행합니다.
 - 전용 호스트가 현재 특정 인스턴스 유형을 지원하는 경우 Support multiple instance types(여러 인스턴스 유형 지원)이 활성화되지 않으며 인스턴스 유형에 지원되는 인스턴스 유형이 나열됩니다. 현재 인스턴스 패밀리 내의 여러 유형을 지원하도록 호스트를 수정하려면 Support multiple instance types(여러 인스턴스 유형 지원)에서 활성화를 선택합니다.
여러 인스턴스 유형을 지원하도록 호스트를 수정하려면 먼저 호스트에서 실행 중인 모든 인스턴스를 중지해야 합니다.
 - 전용 호스트가 현재 인스턴스 패밀리의 여러 인스턴스 유형을 지원하는 경우 Support multiple instance types(여러 인스턴스 유형 지원)에서 활성화가 선택됩니다. 특정 인스턴스 유형을 지원하도록 호스트를 수정하려면 Support multiple instance types(여러 인스턴스 유형 지원)에서 활성화를 선택 취소한 다음 인스턴스 유형에서 지원할 특정 인스턴스 유형을 선택합니다.
- 전용 호스트에서 지원되는 인스턴스 패밀리는 변경할 수 없습니다.
5. Save를 선택합니다.

기존 콘솔

전용 호스트에 지원되는 인스턴스 유형을 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
 2. 탐색 창에서 전용 호스트를 선택합니다.
 3. 수정할 전용 호스트를 선택하고 작업, 지원되는 인스턴스 유형 수정을 선택합니다.
 4. 전용 호스트의 현재 구성에 따라 다음 중 하나를 수행합니다.
 - 전용 호스트가 현재 특정 인스턴스 유형을 지원하는 경우 여러 인스턴스 유형 지원에서 아니요가 선택됩니다. 현재 인스턴스 패밀리 내의 여러 유형을 지원하도록 호스트를 수정하려면 여러 인스턴스 유형 지원에서 예를 선택합니다.
여러 인스턴스 유형을 지원하도록 호스트를 수정하려면 먼저 호스트에서 실행 중인 모든 인스턴스를 중지해야 합니다.
 - 전용 호스트가 현재 인스턴스 패밀리의 여러 인스턴스 유형을 지원하는 경우 여러 인스턴스 유형 지원에서 예가 선택되고 인스턴스 패밀리에 지원되는 인스턴스 패밀리가 표시됩니다. 특정 인스턴스 유형을 지원하도록 호스트를 수정하려면 여러 인스턴스 유형 지원에서 아니요를 선택한 다음 인스턴스 유형에서 지원할 특정 인스턴스 유형을 선택합니다.
- 전용 호스트에서 지원되는 인스턴스 패밀리는 변경할 수 없습니다.
5. Save를 선택합니다.

AWS CLI

전용 호스트에 지원되는 인스턴스 유형을 수정하려면

`modify-hosts` AWS CLI 명령을 사용합니다.

다음 명령은 m5 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하도록 전용 호스트를 수정합니다.

```
aws ec2 modify-hosts --instance-family m5 --host-ids h-012a3456b7890cdef
```

다음 명령은 m5.xlarge 인스턴스만 지원하도록 전용 호스트를 수정합니다.

```
aws ec2 modify-hosts --instance-type m5.xlarge --instance-family --host-ids h-012a3456b7890cdef
```

PowerShell

전용 호스트에 지원되는 인스턴스 유형을 수정하려면

[Edit-EC2Host](#) Windows PowerShell용 AWS 도구 명령을 사용합니다.

다음 명령은 m5 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하도록 전용 호스트를 수정합니다.

```
PS C:\> Edit-EC2Host --InstanceFamily m5 --HostId h-012a3456b7890cdef
```

다음 명령은 m5.xlarge 인스턴스만 지원하도록 전용 호스트를 수정합니다.

```
PS C:\> Edit-EC2Host --InstanceType m5.xlarge --HostId h-012a3456b7890cdef
```

인스턴스 테넌시 및 선호도 설정

시작한 이후에는 인스턴스의 테넌시를 `dedicated`에서 `host`로, 또는 `host`에서 `dedicated`로 변경할 수 있습니다. 또한 인스턴스와 호스트 간의 선호도를 수정할 수 있습니다. 인스턴스 테넌시 또는 선호도를 수정하려면 인스턴스가 `stopped` 상태여야 합니다.

다음 방법을 사용하여 인스턴스의 테넌시와 선호도를 수정할 수 있습니다.

Console

인스턴스 테넌시 또는 선호도를 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 인스턴스를 선택한 후 수정할 인스턴스를 선택합니다.
3. 작업, 인스턴스 상태, 중지를 차례로 선택합니다.
4. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고 인스턴스 설정, 인스턴스 배치 수정을 차례로 선택합니다.
5. 인스턴스 배치 수정 페이지에서 다음을 구성합니다.
 - 테넌시 - 다음 중 하나를 선택합니다.
 - 전용 하드웨어 인스턴스를 실행 - 인스턴스를 전용 인스턴스로 시작합니다. 자세한 내용은 [전용 인스턴스 \(p. 349\)](#) 단원을 참조하십시오.
 - 전용 호스트에서 인스턴스 시작 - 구성 가능한 선호도가 있는 전용 호스트에서 인스턴스를 시작합니다.
 - 선호도—다음 중 하나를 선택합니다.
 - 내 호스트 중 하나에서 이 인스턴스를 시작할 수 있음—인스턴스 유형을 지원하는 계정의 모든 가용한 전용 호스트에서 인스턴스를 시작합니다.
 - 선택한 호스트에서만 이 인스턴스를 실행할 수 있음—대상 호스트로 선택된 전용 호스트에서만 인스턴스를 실행할 수 있습니다.
 - 대상 호스트—인스턴스를 실행해야 하는 전용 호스트를 선택합니다. 대상 호스트 목록이 표시되지 않는 경우 계정에 사용 가능하며 호환되는 전용 호스트가 없을 수 있습니다.

자세한 내용은 [자동 배치 및 선호도의 이해 \(p. 329\)](#) 단원을 참조하십시오.

6. Save를 선택합니다.

AWS CLI

인스턴스 테넌시 또는 선호도를 수정하려면

[modify-instance-placement](#) AWS CLI 명령을 사용합니다. 다음 예는 지정된 인스턴스의 선호도를 default에서 host로 변경하고 인스턴스가 선호도를 갖는 전용 호스트를 지정합니다.

```
aws ec2 modify-instance-placement --instance-id i-1234567890abcdef0 --affinity host --  
host-id h-012a3456b7890cdef
```

PowerShell

인스턴스 테넌시 또는 선호도를 수정하려면

[Edit-EC2InstancePlacement](#) Windows PowerShell용 AWS 도구 명령을 사용합니다. 다음 예는 지정된 인스턴스의 선호도를 default에서 host로 변경하고 인스턴스가 선호도를 갖는 전용 호스트를 지정합니다.

```
PS C:\> Edit-EC2InstancePlacement -InstanceId i-1234567890abcdef0 -Affinity host -  
HostId h-012a3456b7890cdef
```

전용 호스트 보기

다음 방법을 사용하여 전용 호스트 및 개별 인스턴스의 세부 정보를 볼 수 있습니다.

새로운 콘솔

전용 호스트의 세부 정보를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 전용 호스트페이지에서 호스트를 선택합니다.
4. 호스트 정보를 보려면 세부 정보를 선택합니다.

사용 가능한 vCPU는 새 인스턴스를 시작하기 위해 전용 호스트에서 사용할 수 있는 vCPU를 나타냅니다. 예를 들어, c5 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하고 실행 중인 인스턴스가 없는 전용 호스트에는 사용 가능한 vCPU가 72개 있습니다. 따라서 사용 가능한 vCPU 72개를 활용하여 다양한 조합의 인스턴스 유형을 전용 호스트에서 시작할 수 있습니다.

호스트에서 실행 중인 인스턴스의 정보를 보려면 실행 중인 인스턴스를 선택합니다.

기존 콘솔

전용 호스트의 세부 정보를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 전용 호스트페이지에서 호스트를 선택합니다.
4. 호스트 정보를 보려면 설명을 선택합니다. 사용 가능한 vCPU는 새 인스턴스를 시작하기 위해 전용 호스트에서 사용할 수 있는 vCPU를 나타냅니다. 예를 들어, c5 인스턴스 패밀리 내의 여러 인스턴

스 유형을 지원하고 실행 중인 인스턴스가 없는 전용 호스트에는 사용 가능한 vCPU가 72개 있습니다. 따라서 사용 가능한 vCPU 72개를 활용하여 다양한 조합의 인스턴스 유형을 전용 호스트에서 시작할 수 있습니다.

호스트에서 실행 중인 인스턴스의 정보를 보려면 인스턴스를 선택합니다.

AWS CLI

전용 호스트의 용량을 보려면

[describe-hosts](#) AWS CLI 명령을 사용합니다.

다음 예제에서는 [describe-hosts](#)(AWS CLI) 명령을 사용하여 c5 인스턴스 패밀리 내의 여러 인스턴스 유형을 지원하는 전용 호스트에 사용 가능한 인스턴스 용량을 봅니다. 전용 호스트에는 이미 실행 중인 c5.4xlarge 인스턴스 2개와 c5.2xlarge 인스턴스 4개가 있습니다.

```
C:\> aws ec2 describe-hosts --host-id h-012a3456b7890cdef
```

```
"AvailableInstanceCapacity": [  
    { "AvailableCapacity": 2,  
      "InstanceType": "c5.xlarge",  
      "TotalCapacity": 18 },  
    { "AvailableCapacity": 4,  
      "InstanceType": "c5.large",  
      "TotalCapacity": 36 }  
],  
"AvailableVCpus": 8
```

PowerShell

전용 호스트의 인스턴스 용량을 보려면

[Get-EC2Host](#) Windows PowerShell용 AWS 도구 명령을 사용합니다.

```
PS C:\> Get-EC2Host -HostId h-012a3456b7890cdef
```

전용 호스트 태그 지정

기존 전용 호스트에 사용자 지정 태그를 할당하여 용도, 소유자, 환경 등 다양한 방식으로 주소를 분류할 수 있습니다. 그러면 할당한 사용자 지정 태그를 기반으로 특정 전용 호스트를 빠르게 찾을 수 있습니다. 전용 호스트 태그는 비용 할당 추적에만 사용할 수 있습니다.

또한 생성 시 전용 호스트 볼륨에 태그를 적용할 수도 있습니다. 자세한 내용은 [전용 호스트 할당 \(p. 324\)](#) 단원을 참조하십시오.

다음 방법을 사용하여 전용 호스트에 태그를 지정할 수 있습니다.

새로운 콘솔

전용 호스트에 태그를 지정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 태그 지정할 전용 호스트를 선택하고 작업, 태그 관리를 선택합니다.
4. 태그 관리 화면에서 태그 추가를 선택한 후 태그에 키와 값을 지정합니다.
5. (선택 사항) 태그 추가를 선택하여 전용 호스트에 태그를 추가합니다.
6. [Save changes]를 선택합니다.

기존 콘솔

전용 호스트에 태그를 지정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 태그 지정할 전용 호스트를 선택하고 태그를 선택합니다.
4. 태그 추가/편집을 선택합니다.
5. 태그 추가/편집 대화 상자에서 태그 생성을 선택하고 태그에 대한 키와 값을 지정합니다.
6. (선택 사항) 추가 태그를 전용 호스트에 추가하려면 태그 생성을 선택합니다.
7. 저장을 선택합니다.

AWS CLI

전용 호스트에 태그를 지정하려면

`create-tags` AWS CLI 명령을 사용합니다.

다음 명령은 지정된 전용 호스트에 `Owner=TeamA` 태그를 지정합니다.

```
aws ec2 create-tags --resources h-abc12345678909876 --tags Key=Owner,Value=TeamA
```

PowerShell

전용 호스트에 태그를 지정하려면

`New-EC2Tag` Windows PowerShell용 AWS 도구 명령을 사용합니다.

`New-EC2Tag` 명령에는 전용 호스트 태그에 사용할 키-값 페어를 지정하는 `Tag` 객체가 필요합니다. 다음 명령은 키와 값이 각각 `Tag`, `$tag`인 `Owner`라는 이름의 `TeamA` 객체를 만듭니다.

```
PS C:\> $tag = New-Object Amazon.EC2.Model.Tag
PS C:\> $tag.Key = "Owner"
PS C:\> $tag.Value = "TeamA"
```

다음 명령은 지정된 전용 호스트에 `$tag` 객체를 지정합니다.

```
PS C:\> New-EC2Tag -Resource h-abc12345678909876 -Tag $tag
```

전용 호스트 모니터링

Amazon EC2는 전용 호스트의 상태를 지속적으로 모니터링합니다. 업데이트는 Amazon EC2 콘솔에서 전달됩니다. 다음 방법을 사용하여 전용 호스트에 대한 정보를 볼 수 있습니다.

Console

전용 호스트의 상태를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 목록에서 전용 호스트를 찾아서 상태 열의 값을 검토합니다.

AWS CLI

전용 호스트의 상태를 보려면

[describe-hosts](#) AWS CLI 명령을 사용하고 hostSet 응답 요소의 state 속성을 검토합니다.

```
aws ec2 describe-hosts --host-id h-012a3456b7890cdef
```

PowerShell

전용 호스트의 상태를 보려면

[Get-EC2Host](#) Windows PowerShell용 AWS 도구 명령을 사용하고 hostSet 응답 요소의 state 속성을 검토합니다.

```
PS C:\> Get-EC2Host -HostId h-012a3456b7890cdef
```

다음 표는 가능한 전용 호스트 상태를 설명합니다.

상태	설명
available	AWS가 전용 호스트에 대한 문제를 감지하지 못했습니다. 유지 관리 또는 수리가 예정되어 있지 않습니다. 이 전용 호스트에서 인스턴스를 시작할 수 있습니다.
released	전용 호스트가 해제되었습니다. 더 이상 이 호스트 ID가 사용되지 않습니다. 해제된 호스트는 다시 사용할 수 없습니다.
under-assessment	AWS가 전용 호스트에 있을 수 있는 문제를 탐색 중입니다. 작업이 필요할 경우 AWS Management 콘솔 또는 이메일을 통해 통보됩니다. 이 상태에서는 전용 호스트에서 인스턴스를 시작할 수 없습니다.
pending	새 인스턴스를 시작하기 위해 전용 호스트를 사용할 수 없습니다. 이 호스트는 여러 인스턴스 유형을 지원하도록 수정 (p. 330)되고 있거나, 호스트 복구 (p. 344)가 진행 중입니다.
permanent-failure	복구할 수 없는 오류가 감지되었습니다. 인스턴스 및 이메일을 통해 제거 알림이 제공됩니다. 인스턴스는 계속 실행할 수 있습니다. 이 상태의 전용 호스트에서 모든 인스턴스를 중지 또는 종료할 경우 AWS가 해당 호스트를 사용 중지합니다. AWS는 이 상태에 있는 인스턴스는 다시 시작하지 않습니다. 이 상태에서는 전용 호스트에서 인스턴스를 시작할 수 없습니다.
released-permanent-failure	AWS에서는 오류가 발생한 전용 호스트를 영구 해제하여 더 이상 인스턴스가 실행되지 못하도록 합니다. 전용 호스트 ID를 더 이상 사용할 수 없습니다.

전용 호스트 해제

전용 호스트에서 실행되는 모든 인스턴스를 중지해야 해당 호스트를 해제할 수 있습니다. 이 인스턴스들을 계정의 다른 전용 호스트로 마이그레이션하여 계속 사용할 수 있습니다. 이 단계들은 온디맨드 전용 호스트에만 적용됩니다.

다음 방법을 사용하여 전용 호스트를 해제할 수 있습니다.

새로운 콘솔

전용 호스트를 해제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.

3. 전용 호스트 페이지에서 해제할 전용 호스트를 선택합니다.
4. 작업, Release host(호스트 릴리스)를 선택합니다.
5. 릴리스를 선택하여 확인합니다.

기존 콘솔

전용 호스트를 해제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 전용 호스트 페이지에서 해제할 전용 호스트를 선택합니다.
4. 작업, 호스트 릴리스를 선택합니다.
5. 릴리스를 선택하여 확인합니다.

AWS CLI

전용 호스트를 해제하려면

`release-hosts` AWS CLI 명령을 사용합니다.

```
aws ec2 release-hosts --host-ids h-012a3456b7890cdef
```

PowerShell

전용 호스트를 해제하려면

`Remove-EC2Hosts` Windows PowerShell용 AWS 도구 명령을 사용합니다.

```
PS C:\> Remove-EC2Hosts -HostId h-012a3456b7890cdef
```

전용 호스트를 해제한 후에는 동일한 호스트 또는 호스트 ID를 다시 사용할 수 없으며, 더 이상 해당 호스트에 대해 온디맨드 결제 요금이 부과되지 않습니다. 전용 호스트의 상태가 `released`로 변경되고 이 호스트에서 인스턴스를 시작할 수 없게 됩니다.

Note

최근에 전용 호스트를 해제한 경우 제한 계산에서 제외될 때까지 시간이 약간 걸릴 수 있습니다. 이 시간 동안 새로운 전용 호스트 할당을 시도할 경우 `LimitExceeded` 오류가 발생할 수 있습니다. 이런 경우 몇 분 후에 새 호스트를 할당해 보십시오.

중지된 인스턴스는 계속 사용할 수 있으며 인스턴스 페이지에 나열됩니다. 또한 `host` 테넌시 설정을 유지합니다.

전용 호스트 예약 구입

다음과 같은 방법으로 예약을 구매할 수 있습니다.

Console

예약을 구매하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 전용 호스트, 전용 호스트 예약, 전용 호스트 예약 구매를 선택합니다.

3. 전용 호스트 예약 구매 화면에서 기본 설정을 사용하여 이용 가능한 상품을 검색하거나 다음에 대한 사용자 지정 값을 지정할 수 있습니다.
 - 호스트 인스턴스 패밀리—나열되는 옵션은 계정에서 예약에 할당되지 않은 전용 호스트에 해당합니다.
 - 가용 영역—계정에서 예약에 할당되지 않은 전용 호스트의 가용 영역입니다.
 - 결제 옵션—상품에 대한 결제 방식입니다.
 - 기간—1년 또는 3년이 될 수 있는 예약 기간입니다.
4. 상품 찾기를 선택하고 요건에 맞는 상품을 선택합니다.
5. 예약에 연결할 전용 호스트를 선택하고 검토를 선택합니다.
6. 주문을 검토한 후 주문을 선택합니다.

AWS CLI

예약을 구매하려면

1. **describe-host-reservation-offerings** AWS CLI 명령을 사용하여 요구 사항에 맞는 사용 가능한 상품 목록을 나열합니다. 다음 예는 m4 인스턴스 패밀리의 인스턴스를 지원하고 사용 기간이 1년인 상품 목록을 나열합니다.

Note

기간은 초 단위로 지정됩니다. 1년 기간에는 31,536,000초가 포함되고, 3년 기간에는 94,608,000초가 포함됩니다.

```
aws ec2 describe-host-reservation-offerings --filter Name=instance-family,Values=m4  
--max-duration 31536000
```

이 명령은 조건과 일치하는 상품 목록을 반환합니다. 구입할 상품의 `offeringId`를 기록하십시오.

2. **purchase-host-reservation** AWS CLI 명령을 사용하여 상품을 구입하고 이전 단계에서 기록한 `offeringId`를 입력합니다. 다음 예제에서는 지정된 예약을 구매하고 이미 AWS 계정에 할당된 특정 전용 호스트와 연결하며, 키가 `purpose`이고 값이 `production`인 태그를 적용합니다.

```
aws ec2 purchase-host-reservation --offering-id hro-03f707bf363b6b324 --  
host-id-set h-013abcd2a00cbd123 --tag-specifications 'ResourceType=host-  
reservation,Tags=[{Key=purpose,Value=production}]'
```

PowerShell

예약을 구매하려면

1. **Get-EC2HostReservationOffering** Windows PowerShell용 AWS 도구 명령을 사용하여 요구 사항에 맞는 사용 가능한 상품 목록을 나열합니다. 다음 예는 m4 인스턴스 패밀리의 인스턴스를 지원하고 사용 기간이 1년인 상품 목록을 나열합니다.

Note

기간은 초 단위로 지정됩니다. 1년 기간에는 31,536,000초가 포함되고, 3년 기간에는 94,608,000초가 포함됩니다.

```
PS C:\> $filter = @{Name="instance-family"; Value="m4"}
```

```
PS C:\> Get-EC2HostReservationOffering -filter $filter -MaxDuration 31536000
```

이 명령은 조건과 일치하는 상품 목록을 반환합니다. 구입할 상품의 `OfferingId`를 기록하십시오.

2. [New-EC2HostReservation](#) Windows PowerShell용 AWS 도구 명령을 사용하여 상품을 구입하고 이 전 단계에서 기록한 `OfferingId`를 입력합니다. 다음 예는 특정 예약을 구입하고 이미 AWS 계정에 할당된 특정 전용 호스트와 연결합니다.

```
PS C:\> New-EC2HostReservation -OfferingId hro-03f707bf363b6b324 -  
HostIdSet h-013abcd2a00cbd123
```

전용 호스트 예약 보기

다음과 같은 예약과 관련된 전용 호스트 정보를 볼 수 있습니다.

- 예약 기간
- 결제 옵션
- 시작 및 종료 날짜

다음과 같은 방법으로 전용 호스트 예약에 대한 세부 정보를 볼 수 있습니다.

Console

전용 호스트 예약의 세부 정보를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 전용 호스트 페이지에서 전용 호스트 예약을 선택하고 제공된 목록에서 해당 예약을 선택합니다.
4. 예약에 대한 정보를 보려면 세부 정보를 선택합니다.
5. 예약이 연결되어 있는 전용 호스트에 대한 정보를 보려면 호스트를 선택합니다.

AWS CLI

전용 호스트 예약의 세부 정보를 보려면

[describe-host-reservations](#) AWS CLI 명령을 사용합니다.

```
aws ec2 describe-host-reservations
```

PowerShell

전용 호스트 예약의 세부 정보를 보려면

[Get-EC2HostReservation](#) Windows PowerShell용 AWS 도구 명령을 사용합니다.

```
PS C:\> Get-EC2HostReservation
```

전용 호스트 예약 태그 지정

전용 호스트 예약에 사용자 지정 태그를 할당하여 예약을 용도, 소유자, 환경 등 다양한 방식으로 분류할 수 있습니다. 그러면 할당한 사용자 지정 태그를 기반으로 특정 전용 호스트 예약을 빠르게 찾을 수 있습니다.

명령줄 도구만 사용하여 전용 호스트 예약에 태그를 지정할 수 있습니다.

AWS CLI

전용 호스트 예약에 태그를 지정하려면

[create-tags](#) AWS CLI 명령을 사용합니다.

```
aws ec2 create-tags --resources hr-1234563a4ffc669ae --tags Key=Owner,Value=TeamA
```

PowerShell

전용 호스트 예약에 태그를 지정하려면

[New-EC2Tag](#) Windows PowerShell용 AWS 도구 명령을 사용합니다.

New-EC2Tag 명령에는 전용 호스트 예약 태그에 사용할 키-값 페어를 지정하는 Tag 파라미터가 필요합니다. 다음 명령은 Tag 파라미터를 생성합니다.

```
PS C:\> $tag = New-Object Amazon.EC2.Model.Tag
PS C:\> $tag.Key = "Owner"
PS C:\> $tag.Value = "TeamA"
```

```
PS C:\> New-EC2Tag -Resource hr-1234563a4ffc669ae -Tag $tag
```

공유된 전용 호스트 작업

전용 호스트 공유를 통해 전용 호스트 소유자는 전용 호스트를 다른 AWS 계정과 공유하거나 AWS 조직 내에서 공유할 수 있습니다. 이를 통해 전용 호스트를 중앙에서 생성 및 관리하고 전용 호스트를 여러 AWS 계정 또는 AWS 조직 내에서 공유할 수 있습니다.

이 모델에서 전용 호스트를 소유한 AWS 계정(소유자)이 다른 AWS 계정(소비자)과 전용 호스트를 공유합니다. 소비자는 자신의 계정에서 할당한 전용 호스트에서 인스턴스를 시작할 때와 동일한 방식으로 공유된 전용 호스트에서 인스턴스를 시작할 수 있습니다. 소유자는 전용 호스트 및 거기서 시작한 인스턴스를 관리할 책임이 있습니다. 소유자는 소비자가 공유된 전용 호스트에서 시작하는 인스턴스를 수정할 수 없습니다. 소비자는 공유된 전용 호스트에서 시작한 인스턴스를 관리할 책임이 있습니다. 소비자는 다른 소비자나 전용 호스트 소유자가 소유한 인스턴스를 보거나 수정할 수 없으며 공유된 전용 호스트를 수정할 수 없습니다.

전용 호스트 소유자는 다음 상대와 전용 호스트를 공유할 수 있습니다.

- AWS 조직 내부 또는 외부의 특정 AWS 계정
- AWS 조직 내부의 조직 단위
- 전체 AWS 조직

목차

- [전용 호스트 공유를 위한 사전 조건 \(p. 341\)](#)
- [전용 호스트 공유 제한 사항 \(p. 341\)](#)
- [관련 서비스 \(p. 341\)](#)
- [여러 가용 영역 공유 \(p. 341\)](#)
- [전용 호스트 공유 \(p. 341\)](#)
- [공유된 전용 호스트의 공유 해제 \(p. 342\)](#)
- [공유된 전용 호스트 식별 \(p. 343\)](#)
- [공유된 전용 호스트에서 실행되는 인스턴스 보기 \(p. 343\)](#)
- [공유된 전용 호스트 권한 \(p. 344\)](#)
- [결제 및 측정 \(p. 344\)](#)

- [전용 호스트 제한 \(p. 344\)](#)
- [호스트 복구 및 전용 호스트 공유 \(p. 344\)](#)

전용 호스트 공유를 위한 사전 조건

- 전용 호스트를 공유하려면 전용 호스트를 AWS 계정에 소유하고 있어야 합니다. 자신과 공유된 전용 호스 트는 공유할 수 없습니다.
- AWS 조직 또는 AWS 조직의 조직 단위와 전용 호스트를 공유하려면 AWS Organizations로 공유를 활성화 해야 합니다. 자세한 내용은 AWS RAM 사용 설명서의 [AWS Organizations를 사용하여 공유 사용 설정을 참조하십시오.](#)

전용 호스트 공유 제한 사항

`u-6tb1.metal`, `u-9tb1.metal`, `u-12tb1.metal`, `u-18tb1.metal` 및 `u-24tb1.metal` 인스턴스 유형에 할당된 전용 호스트는 공유할 수 없습니다.

관련 서비스

AWS 리소스 액세스 관리자

전용 호스트 공유는 AWS 리소스 액세스 관리자(AWS RAM)와 통합됩니다. AWS RAM은 어떤 AWS 계정이나 AWS Organizations를 통해 AWS 리소스를 공유하도록 해주는 서비스입니다. AWS RAM을 사용하여 리소스 공유로 생성한 사용자 소유 리소스를 공유할 수 있습니다. 리소스 공유는 공유할 리소스와 공유 대상 소비자를 지정합니다. 소비자는 개인 AWS 계정 또는 조직 단위 또는 AWS Organizations의 전체 조직일 수 있습니다.

AWS RAM에 대한 자세한 내용은 [AWS RAM 사용 설명서](#)를 참조하십시오.

여러 가용 영역 공유

리전의 가용 영역에 걸쳐 리소스가 배포될 수 있도록 AWS는 각 계정의 이름에 가용 영역을 독립적으로 매핑합니다. 이로 인해 계정 전체에서 가용 영역 이름의 차이가 발생할 수 있습니다. 예를 들어 AWS AWS 계정의 `us-east-1a` 가용 영역은 다른 AWS 계정에 대한 `us-east-1a` 가용 영역과 위치가 동일하지 않을 수 있습니다.

계정과 관련된 전용 호스트의 위치를 확인하려면 가용 영역 ID(AZ ID)를 사용해야 합니다. 가용 영역 ID는 모든 AWS 계정에서 가용 영역의 고유하고 일관된 식별자입니다. 예를 들어 `use1-az1`은 `us-east-1` 리전의 가용 영역 ID이고 모든 AWS 계정에서 동일한 위치입니다.

계정에 속한 가용 영역의 가용 영역 ID를 보려면

1. <https://console.aws.amazon.com/ram>에서 AWS RAM 콘솔을 엽니다.
2. 현재 리전의 가용 영역 ID는 화면 오른쪽에 있는 Your AZ ID(AZ ID) 패널에 표시됩니다.

전용 호스트 공유

소유자가 전용 호스트를 공유하면 소비자가 호스트에서 인스턴스를 시작할 수 있게 됩니다. 소비자는 가용 용량이 허용하는 만큼 많은 인스턴스를 공유된 호스트에서 시작할 수 있습니다.

Important

전용 호스트에서 BYOL 라이선스를 공유할 수 있는 적절한 라이선스 권리가 있는지 확인할 책임은 사용자에게 있습니다.

자동 배치가 활성화된 상태로 전용 호스트를 공유하는 경우 의도하지 않은 전용 호스트 사용으로 이어질 수 있으므로 다음 내용에 유의하십시오.

- 소비자가 전용 호스트 테넌시로 인스턴스를 시작하고 계정에서 소유한 전용 호스트에 용량이 없으면 인스턴스는 공유된 전용 호스트에서 자동으로 시작됩니다.

전용 호스트를 공유하려면 리소스 공유에 추가해야 합니다. AWS RAM 리소스 공유는 여러 AWS 계정에서 리소스를 공유할 수 있게 해주는 리소스입니다. 리소스 공유는 공유할 리소스와 공유 대상 소비자를 지정합니다. 전용 호스트를 기존 리소스에 추가하거나 새 리소스 공유에 추가할 수 있습니다.

AWS Organizations의 조직에 속해 있고 조직 내의 공유가 활성화되어 있으면 조직의 소비자는 공유된 전용 호스트에 대한 액세스 권한이 자동으로 부여됩니다. 그렇지 않으면 소비자는 리소스 공유에 가입하라는 초대장을 받고 초대를 수락한 후 공유된 전용 호스트의 액세스 권한을 받습니다.

Note

전용 호스트를 공유한 후 몇 분이 지나면 소비자가 그에 대한 액세스 권한을 갖게 됩니다.

다음 방법 중 하나를 사용하여 소유하고 있는 전용 호스트를 공유할 수 있습니다.

Amazon EC2 console

Amazon EC2 콘솔을 사용하여 소유하고 있는 전용 호스트를 공유하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 공유할 전용 호스트를 선택하고 작업, 호스트 공유를 선택합니다.
4. 전용 호스트를 추가할 리소스 공유를 선택하고 호스트 공유를 선택합니다.

소비자가 공유 호스트에 액세스하려면 몇 분이 걸릴 수 있습니다.

AWS RAM console

AWS RAM 콘솔을 사용하여 소유하고 있는 전용 호스트를 공유하려면

AWS RAM 사용 설명서에서 [리소스 공유 만들기](#)를 참조하십시오.

AWS CLI

AWS CLI를 사용하여 소유하고 있는 전용 호스트를 공유하려면

[create-resource-share](#) 명령을 사용합니다.

공유된 전용 호스트의 공유 해제

전용 호스트 소유자는 언제든지 공유된 전용 호스트를 공유 해제할 수 있습니다. 공유된 전용 호스트를 공유 해제할 때 다음 규칙이 적용됩니다.

- 전용 호스트를 공유한 소비자는 더 이상 새 인스턴스를 이 호스트에서 시작할 수 없습니다.
- 공유를 해제할 때 전용 호스트에서 실행되던 소비자 소유의 인스턴스는 계속 실행되지만 [만료되도록 예약](#)됩니다. 소비자는 인스턴스 만료 알림을 받고 2주가 지나면 알림에 대한 조치가 취해집니다. 하지만 전용 호스트가 만료 통지 기간 안에 소비자와 다시 공유되면 인스턴스 만료가 취소됩니다.

소유하고 있는 공유된 전용 호스트를 공유 해제하려면 리소스 공유에서 제거해야 합니다. 이를 위해 다음 방법 중 하나를 사용할 수 있습니다.

Amazon EC2 console

Amazon EC2 콘솔을 사용하여 소유하고 있는 공유 전용 호스트를 공유 해제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 전용 호스트를 선택합니다.
3. 공유를 해제할 전용 호스트를 선택하고 공유 탭을 선택합니다.
4. 공유 탭에는 전용 호스트가 추가된 리소스 공유가 나열됩니다. 전용 호스트를 제거할 리소스 공유를 선택하고 리소스 공유에서 호스트 제거를 선택합니다.

AWS RAM console

AWS RAM 콘솔을 사용하여 소유하고 있는 공유 전용 호스트를 공유 해제하려면

AWS RAM 사용 설명서에서 [리소스 공유 업데이트](#)를 참조하십시오.

Command line

AWS CLI를 사용하여 소유하고 있는 공유 전용 호스트를 공유 해제하려면

`disassociate-resource-share` 명령을 사용합니다.

공유된 전용 호스트 식별

소유자와 소비자는 다음 방법 중 하나를 사용하여 공유 전용 호스트를 식별할 수 있습니다.

Amazon EC2 console

Amazon EC2 콘솔을 사용하여 공유된 전용 호스트를 식별하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다. 소유하고 있는 전용 호스트 및 자신과 공유된 전용 호스트가 화면에 나열됩니다. 소유자 열에 전용 호스트 소유자의 AWS 계정 ID가 표시됩니다.

Command line

AWS CLI를 사용하여 공유된 전용 호스트를 식별하려면

`describe-hosts` 명령을 사용합니다. 이 명령은 소유하고 있는 전용 호스트 및 자신과 공유된 전용 호스트를 반환합니다.

공유된 전용 호스트에서 실행되는 인스턴스 보기

소유자와 소비자는 다음 방법 중 하나를 사용하여 언제든 공유 전용 호스트에서 실행 중인 인스턴스를 볼 수 있습니다.

Amazon EC2 console

Amazon EC2 콘솔을 사용하여 공유된 전용 호스트에서 실행되는 인스턴스를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 인스턴스를 볼 전용 호스트를 선택하고 인스턴스를 선택합니다. 호스트에서 실행 중인 인스턴스가 탭에 나열됩니다. 소유자는 소비자가 시작한 인스턴스를 포함하여 호스트에서 실행되는 인스턴스를 모두 볼 수 있습니다. 소비자는 자신이 호스트에서 시작하여 실행 중인 인스턴스만 볼 수 있습니다. 소유자 열에 인스턴스를 시작한 계정의 AWS 계정 ID가 표시됩니다.

Command line

AWS CLI를 사용하여 공유된 전용 호스트에서 실행되는 인스턴스를 보려면

[describe-hosts](#) 명령을 사용합니다. 이 명령은 각 전용 호스트에서 실행되는 인스턴스를 반환합니다. 소유자는 호스트에서 실행되는 인스턴스를 모두 볼 수 있습니다. 소비자는 공유된 호스트에서 자신이 시작하여 실행 중인 인스턴스만 볼 수 있습니다. `InstanceOwnerId`에 인스턴스 소유자의 AWS 계정 ID가 표시됩니다.

공유된 전용 호스트 권한

소유자에 대한 권한

소유자는 공유된 전용 호스트 및 거기서 시작한 인스턴스를 관리할 책임이 있습니다. 소유자는 소비자가 시작한 인스턴스를 포함하여 공유된 전용 호스트에서 실행되는 인스턴스를 모두 볼 수 있습니다. 하지만 소유자는 소비자가 시작하여 실행 중인 인스턴스에 대해 아무 작업도 할 수 없습니다.

소비자에 대한 권한

소비자는 공유된 전용 호스트에서 시작한 인스턴스를 관리할 책임이 있습니다. 소비자는 어떤 식으로든 공유된 전용 호스트를 수정할 수 없으며 다른 소비자 또는 전용 호스트 소유자가 시작한 인스턴스를 보거나 수정할 수 없습니다.

결제 및 측정

전용 호스트 공유에 대한 추가 비용은 없습니다.

공유하는 전용 호스트 비용이 소유자에게 청구됩니다. 공유된 전용 호스트에서 소비자가 시작한 인스턴스 비용을 소비자에게 청구하지 않습니다.

전용 호스트 예약은 공유된 전용 호스트의 결제 할인을 계속 제공합니다. 전용 호스트 소유자만 자신이 소유한 공유 전용 호스트의 전용 호스트 예약을 구매할 수 있습니다.

전용 호스트 제한

공유된 전용 호스트 수는 소유자의 전용 호스트 제한에만 포함됩니다. 소비자의 전용 호스트 제한은 자신과 공유된 전용 호스트의 영향을 받지 않습니다. 마찬가지로 공유된 전용 호스트에서 소비자가 시작하는 인스턴스도 그 인스턴스 제한에 포함되지 않습니다.

호스트 복구 및 전용 호스트 공유

호스트 복구는 전용 호스트 소유자 및 이 호스트가 공유된 소비자가 시작한 인스턴스를 복구합니다. 대체 전용 호스트는 소유자 계정에 할당되며, 원래의 전용 호스트와 같은 리소스 공유에 추가되고 같은 소비자와 공유됩니다.

자세한 내용은 [호스트 복구 \(p. 344\)](#) 단원을 참조하십시오.

호스트 복구

호스트 복구는 전용 호스트에서 오류가 감지되면 새 대체 호스트로 인스턴스를 자동으로 다시 시작합니다. 호스트 복구는 수동 개입의 필요성을 줄이고 예기치 않은 전용 호스트 장애가 발생하는 경우 운영 부담을 줄입니다.

또한 내장된 AWS License Manager 통합 기능은 호스트 복구가 발생하는 경우 라이선스 추적 및 관리를 자동화합니다.

Note

AWS License Manager 통합은 AWS License Manager가 제공되는 지역에서만 지원됩니다.

목차

- [호스트 복구 기본 사항 \(p. 345\)](#)
- [지원되는 인스턴스 유형 \(p. 346\)](#)
- [호스트 복구 구성 \(p. 346\)](#)
- [호스트 복구 상태 \(p. 347\)](#)
- [지원되지 않는 인스턴스 수동 복구 \(p. 347\)](#)
- [관련 서비스 \(p. 348\)](#)
- [요금 \(p. 348\)](#)

호스트 복구 기본 사항

호스트 복구는 호스트 수준 상태 확인을 사용하여 전용 호스트 가용성을 평가하고 기본 시스템 오류를 감지합니다. 시스템 상태 확인이 실패할 수 있는 문제의 예를 들면 다음과 같습니다.

- 네트워크 연결 끊김
- 시스템 전원 중단
- 물리적 호스트의 하드웨어 또는 소프트웨어 문제

전용 호스트에서 시스템 장애가 감지되면 호스트 복구가 시작되고 Amazon EC2가 자동으로 전용 호스트 대체를 할당합니다. 전용 호스트 대체는 새로운 호스트 ID를 받지만 다음과 같이 원본 전용 호스트와 동일한 속성을 유지합니다.

- 가용 영역
- 인스턴스 유형
- 태그
- 자동 배치 설정

대체 전용 호스트가 할당된 후 인스턴스는 대체 전용 호스트로 복구됩니다. 복구 된 인스턴스는 다음과 같이 원본 인스턴스와 동일한 속성을 유지합니다.

- 인스턴스 ID
- 프라이빗 IP 주소
- 탄력적 IP 주소
- EBS 볼륨 장치
- 모든 인스턴스 메타데이터

인스턴스가 손상된 전용 호스트와 호스트 선호도 관계를 갖는 경우 복구된 인스턴스는 대체 전용 호스트와 호스트 선호도를 수립합니다.

모든 인스턴스가 대체 전용 호스트로 복구되면 손상된 전용 호스트가 해제되고 대체 전용 호스트를 사용할 수 있게 됩니다.

호스트 복구가 시작되면 AWS 계정 소유자에게 이메일 및 AWS Personal Health Dashboard 이벤트로 알림이 전송됩니다. 호스트 복구가 성공적으로 완료되면 두 번째 알림이 전송됩니다.

중지된 인스턴스는 대체 전용 호스트로 복구되지 않습니다. 손상된 전용 호스트를 대상으로 하는 중지된 인스턴스를 시작하려고 하면 인스턴스가 시작되지 않습니다. 중지된 인스턴스를 수정하여 다른 전용 호스트를 대상으로하거나 일치하는 구성 및 자동 배치가 활성화된 사용 가능한 전용 호스트에서 시작하는 것이 좋습니다.

인스턴스 스토리지가 포함된 인스턴스는 대체 전용 호스트로 복구되지 않습니다. 해결 수단으로, 손상된 전용 호스트는 만료 표시되고 호스트 복구가 완료된 후 만료 알림이 전송됩니다. 손상된 전용 호스트의 나머지 인스턴스를 수동으로 복구하려면 지정된 기간 내에 만료 알림에 설명된 해결 단계를 수행합니다.

AWS License Manager를 사용하여 라이선스를 추적하는 경우 AWS License Manager는 라이선스 구성 제한에 따라 대체 전용 호스트에 대한 새 라이선스를 할당합니다. 라이선스 구성에 호스트 복구의 결과로 위반되는 하드 제한이 있는 경우 복구 프로세스가 허용되지 않으며 Amazon SNS 알림을 통해 호스트 복구 실패 알림이 전송됩니다. 라이선스 구성에 호스트 복구의 결과로 위반되는 소프트 제한이 있는 경우 복구가 계속 허용되며 Amazon SNS 알림을 통해 제한 위반 알림이 전송됩니다. 자세한 내용은 AWS License Manager 사용 설명서의 [라이선스 구성 사용 단원](#)을 참조하십시오.

지원되는 인스턴스 유형

호스트 복구는 A1, C3, C4, C5, C5n, M3, M4, M5, M5n, P3, R3, R4, R5, R5n, X1, X1e, u-6tb1, u-9tb1, u-12tb1, u-18tb1 및 u-24tb1의 인스턴스 패밀리에 대해 지원됩니다.

지원되지 않는 인스턴스를 복구하려면 [지원되지 않는 인스턴스 수동 복구 \(p. 347\)](#) 단원을 참조하십시오.

호스트 복구 구성

전용 호스트 할당 시 또는 Amazon EC2 콘솔 또는 AWS Command Line Interface(CLI)를 사용하여 할당한 후 호스트 복구를 구성할 수 있습니다.

목차

- [호스트 복구 활성화 \(p. 346\)](#)
- [호스트 복구 비활성화 \(p. 346\)](#)
- [호스트 복구 구성 보기 \(p. 347\)](#)

호스트 복구 활성화

전용 호스트 할당 시 또는 할당 후 호스트 복구를 활성화할 수 있습니다.

전용 호스트 할당 시 호스트 복구를 활성화하는 방법에 대한 자세한 내용은 [전용 호스트 할당 \(p. 324\)](#) 단원을 참조하십시오.

콘솔을 사용하여 할당 후 호스트 복구를 활성화하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 호스트 복구를 활성화할 전용 호스트를 선택한 다음 작업, Modify Host Recovery(호스트 복구 수정)를 선택합니다.
4. Host recovery(호스트 복구)에서 활성화를 선택한 다음 저장을 선택합니다.

AWS CLI를 사용하여 할당 후 호스트 복구를 활성화하려면

`modify-hosts` 명령을 사용하여 `host-recovery` 파라미터를 지정합니다.

```
$ aws ec2 modify-hosts --host-recovery on --host-ids h-012a3456b7890cdef
```

호스트 복구 비활성화

전용 호스트가 할당된 후 언제든 호스트 복구를 비활성화할 수 있습니다.

콘솔을 사용하여 할당 후 호스트 복구를 비활성화하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 전용 호스트를 선택합니다.
3. 호스트 복구를 비활성화할 전용 호스트를 선택한 다음 작업, Modify Host Recovery(호스트 복구 수정)를 선택합니다.

- Host recovery(호스트 복구)에서 비활성화를 선택한 다음 저장을 선택합니다.

AWS CLI를 사용하여 할당 후 호스트 복구를 비활성화하려면

`modify-hosts` 명령을 사용하여 `host-recovery` 파라미터를 지정합니다.

```
$ aws ec2 modify-hosts --host-recovery off --host-ids h-012a3456b7890cdef
```

호스트 복구 구성 보기

언제든지 전용 호스트에 대한 호스트 복구 구성을 볼 수 있습니다.

콘솔을 사용하여 전용 호스트에 대한 호스트 복구 구성을 보려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 전용 호스트를 선택합니다.
- 전용 호스트를 선택하고 설명 탭에서 Host Recovery(호스트 복구) 필드를 검토합니다.

AWS CLI를 사용하여 전용 호스트에 대한 호스트 복구 구성을 보려면

`describe-hosts` 명령을 사용합니다.

```
$ aws ec2 describe-hosts --host-ids h-012a3456b7890cdef
```

HostRecovery 응답 요소는 호스트 복구가 활성화 또는 비활성화됐는지 여부를 나타냅니다.

호스트 복구 상태

전용 호스트 장애가 감지되면 손상된 전용 호스트는 `under-assessment` 상태가 되고 모든 인스턴스는 `impaired` 상태가 됩니다. `under-assessment` 상태에 있는 동안에는 손상된 전용 호스트에서 인스턴스를 시작할 수 없습니다.

대체 전용 호스트가 할당된 후에는 `pending` 상태가 됩니다. 호스트 복구 프로세스가 완료될 때까지 이 상태로 유지됩니다. `pending` 상태에 있는 동안에는 대체 전용 호스트에서 인스턴스를 시작할 수 없습니다. 대체 전용 호스트에서 복구된 인스턴스는 복구 프로세스 중에 `impaired` 상태로 유지됩니다.

호스트 복구가 완료되면 대체 전용 호스트가 `available` 상태가 되고 복구된 인스턴스는 `running` 상태로 돌아갑니다. `available` 상태가 된 후 대체 전용 호스트로 인스턴스를 시작할 수 있습니다. 손상된 원본 전용 호스트는 영구적으로 해제되고 `released-permanent-failure` 상태가 됩니다.

손상된 전용 호스트에 인스턴스 스토어 볼륨을 가진 인스턴스 등 호스트 복구를 지원하지 않는 인스턴스가 있는 경우 전용 호스트가 해제되지 않습니다. 대신, 만료 표시되고 `permanent-failure` 상태가 됩니다.

지원되지 않는 인스턴스 수동 복구

호스트 복구는 인스턴스 스토어 볼륨을 사용하는 인스턴스 복구를 지원하지 않습니다. 자동으로 복구할 수 없는 인스턴스를 수동으로 복구하려면 아래 지침을 따릅니다.

Warning

인스턴스가 중지되거나 종료되면 인스턴스 스토어 볼륨의 데이터는 삭제됩니다. 여기에는 루트 디바이스가 EBS 볼륨인 인스턴스에 연결된 인스턴스 스토어 볼륨도 포함됩니다. 인스턴스 스토어 볼륨에서 데이터를 유지하려면 인스턴스가 중지되거나 종료되기 전에 영구 스토리지에 백업하십시오.

EBS 지원 인스턴스 수동 복구

자동으로 복구할 수 없는 EBS 지원 인스턴스의 경우 인스턴스를 수동으로 중지했다가 시작하여 새 전용 호스트로 복구하는 것이 좋습니다. 인스턴스 중단과 중단 후 인스턴스 구성에 발생하는 변경 사항에 대한 자세한 내용은 [인스턴스 중지 및 시작 \(p. 440\)](#) 단원을 참조하십시오.

인스턴스 스토어 지원 인스턴스 수동 복구

예를 들어 자동으로 복구할 수 없는 스토어 백업 인스턴스의 경우 다음작업을 수행하는 것이 좋습니다.

1. 최신 전용 호스트의 새로운 AMI에서 대체 인스턴스를 시작합니다.
2. 필요한 모든 데이터를 대체 인스턴스로 마이그레이션합니다.
3. 손상된 전용 호스트의 원본 인스턴스를 종료합니다.

관련 서비스

전용 호스트는 다음과 같은 AWS 서비스와 통합됩니다.

- AWS License Manager—Amazon EC2 전용 호스트에서 라이선스를 추적합니다(AWS License Manager를 사용할 수 있는 리전에서만 지원됨). 자세한 내용은 [AWS License Manager 사용 설명서](#)를 참조하십시오.

요금

호스트 복구 사용에 대한 추가 요금은 없지만 일반적으로 전용 호스트 요금이 적용됩니다. 자세한 내용은 [Amazon EC2 전용 호스트 요금](#)을 참조하십시오.

호스트 복구가 시작된 직후 손상된 전용 호스트에 대해서는 더 이상 요금이 청구되지 않습니다. 대체 전용 호스트에 대한 요금 청구는 available 상태가 된 후에 시작됩니다.

손상된 전용 호스트가 온디맨드 요금을 사용하여 청구된 경우 대체 전용 호스트도 온디맨드 요금을 사용하여 청구됩니다. 손상된 전용 호스트에 활성 전용 호스트 예약이 있는 경우, 대체 전용 호스트로 전송됩니다.

구성 변경 추적

AWS Config를 사용하여 전용 호스트의 구성 변경 내용과 이 전용 호스트에서 시작, 중지 또는 종료된 인스턴스의 구성 변경 내용을 기록할 수 있습니다. 그런 다음 AWS Config가 캡처한 정보를 라이선스 보고용 데이터 소스로 사용할 수 있습니다.

AWS Config는 전용 호스트 및 인스턴스의 구성 정보를 개별적으로 기록하고 관계를 통해 이 정보를 페어링 합니다. 보고 조건은 세 가지가 있습니다.

- AWS Config 레코딩 상태—활성이라면 AWS Config에서 하나 이상의 AWS 리소스 유형을 녹화 중입니다. 이러한 리소스 유형에는 전용 호스트 및 전용 인스턴스가 포함될 수 있습니다. 라이선스 보고에 필요한 정보를 캡처하려면 다음 필드에서 호스트 및 인스턴스가 기록되는지 확인합니다.
- 호스트 레코딩 상태 - 활성으로 설정 시 전용 호스트 구성 정보가 기록됩니다.
- 인스턴스 레코딩 상태 - 활성이라면 전용 인스턴스 구성 정보가 레코딩됩니다.

세 조건 중 하나라도 비활성화되면 Config 레코딩 편집 버튼의 아이콘이 빨간색으로 표시됩니다. 이 도구의 이점을 최대한 활용하려면 세 기록 방법을 모두 활성화하십시오. 세 방법이 모두 활성화되면 아이콘이 녹색으로 표시됩니다. 설정을 편집하려면 Config 레코딩 편집을 선택합니다. 그러면 AWS Config 콘솔의 AWS Config 설정 페이지로 이동하며, 여기서 AWS Config를 설정하고 호스트, 인스턴스 및 기타 지원되는 리소스 유형에 대한 기록을 시작할 수 있습니다. 자세한 내용은 AWS Config 개발자 안내서의 [콘솔을 사용하여 AWS Config 설정](#) 단원을 참조하십시오.

Note

AWS Config가 리소스를 발견하여 기록을 시작합니다. 이 과정은 몇 분이 걸릴 수 있습니다.

AWS Config가 호스트 및 인스턴스 구성 변경을 기록하기 시작한 후, 설정 또는 해제한 호스트와 시작, 중지 또는 종료한 인스턴스의 구성 내역을 확인할 수 있습니다. 예를 들어 전용 호스트 구성 내역의 특정 시점에서 호스트에서 몇 개의 인스턴스가 시작되었는지 여부를 호스트의 소켓 및 코어 수와 함께 확인할 수 있습니다. 이러한 인스턴스 각각에 대해 Amazon 머신 이미지(AMI)의 ID를 조회할 수도 있습니다. 이 정보를 이용하여 소켓당 또는 코어당 라이선스된 서버 한정 소프트웨어에 대한 라이선스를 보고할 수 있습니다.

다음 방법 중 하나를 사용하여 구성 내역을 볼 수 있습니다:

- AWS Config 콘솔 사용. 기록된 리소스 각각에 대해 구성 세부 정보의 내역을 제공하는 타임라인 페이지를 볼 수 있습니다. 이 페이지를 보려면 전용 호스트 페이지의 Config 타임라인 열에서 회색 아이콘을 선택합니다. 보다 자세한 내용은 AWS Config 개발자 안내서의 [AWS Config 콘솔에서 구성 세부 사항 보기](#)를 참조하십시오.
- AWS CLI 명령 실행 먼저 `list-discovered-resources` 명령을 사용하여 모든 호스트 및 인스턴스의 목록을 가져올 수 있습니다. 그런 다음 `get-resource-config-history` 명령을 사용하여 특정 기간에 대해 특정 호스트 또는 인스턴스의 구성 세부 정보를 가져올 수 있습니다. 보다 자세한 내용은 AWS Config 개발자 안내서의 [CLI를 사용하여 구성 세부 정보 보기](#) 단원을 참조하십시오.
- 애플리케이션에서 AWS Config API 사용. 먼저 `ListDiscoveredResources` 작업을 사용하여 모든 호스트 및 인스턴스의 목록을 가져올 수 있습니다. 그런 다음 `GetResourceConfigHistory` 작업을 사용하여 특정 기간에 대해 특정 호스트 또는 인스턴스의 구성 세부 정보를 가져올 수 있습니다.

예를 들어 AWS Config에서 모든 전용 호스트의 목록을 가져오려면 다음과 같은 CLI 명령을 실행합니다.

```
aws configservice list-discovered-resources --resource-type AWS::EC2::Host
```

AWS Config에서 특정 전용 호스트의 구성 내역을 가져오려면 다음과 같은 CLI 명령을 실행합니다.

```
aws configservice get-resource-config-history --resource-type AWS::EC2::Instance --  
resource-id i-1234567890abcdef0
```

콘솔을 사용하여 AWS Config 설정을 관리하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 전용 호스트 페이지에서 Config 레코딩 편집을 선택합니다.
3. AWS Config 콘솔에서 제공되는 단계를 수행하여 기록을 캡니다. 자세한 내용은 [Setting up AWS Config using the Console](#) 단원을 참조하십시오.

자세한 내용은 [AWS Config 콘솔에서 구성 세부 정보 보기](#) 단원을 참조하십시오.

명령줄 또는 API를 사용하여 AWS Config를 활성화하려면

- AWS CLI: AWS Config 개발자 안내서의 [구성 세부 정보 보기\(AWS CLI\)](#).
- Amazon EC2 API: [GetResourceConfigHistory](#).

전용 인스턴스

전용 인스턴스는 단일 고객 전용 하드웨어의 Virtual Private Cloud(VPC)에서 실행되는 Amazon EC2 인스턴스입니다. 서로 다른 AWS 계정에 속한 전용 인스턴스는 그러한 계정이 단일 지급인 계정에 연결되어있더라도 하드웨어 수준에서 물리적으로 격리됩니다. 하지만 전용 인스턴스는 전용 인스턴스가 아닌 동일한 AWS 계정의 다른 인스턴스와 하드웨어를 공유할 수 있습니다.

Note

전용 호스트 역시 고객 전용의 물리적 서버입니다. 를 사용하여 서버에서 인스턴스의 배치 방법을 확인 및 제어할 수 있습니다. 자세한 내용은 [전용 호스트 \(p. 320\)](#) 단원을 참조하십시오.

전용 인스턴스 기본 사항

VPC에서 실행하는 각 인스턴스는 테넌시 속성으로 실행됩니다. 이 속성에는 다음과 같은 값이 있습니다.

테넌시 값	설명
default	인스턴스가 공유된 하드웨어에서 실행됩니다.
dedicated	인스턴스가 단일 테넌트 하드웨어에서 실행됩니다.
host	인스턴스는 구성을 제어할 수 있는 격리된 서버인 전용 호스트에서 실행됩니다.

인스턴스를 시작한 이후에는 테넌시를 변경하는 데 몇 가지 제한이 있습니다.

- 시작한 이후에는 인스턴스의 테넌시를 default에서 dedicated 또는 host로 변경할 수 없습니다.
- 시작한 이후에는 인스턴스의 테넌시를 dedicated 또는 host에서 default로 변경할 수 없습니다.

시작한 이후에는 인스턴스의 테넌시를 dedicated에서 host로, 또는 host에서 dedicated로 변경할 수 있습니다. 자세한 내용은 [인스턴스의 테넌시 변경 \(p. 354\)](#) 단원을 참조하십시오.

각 인스턴스에는 관련 인스턴스 테넌시 속성이 있습니다. 이 속성에는 다음과 같은 값이 있습니다.

테넌시 값	설명
default	인스턴스 시작 중 다른 테넌트를 명시적으로 지정하지 않는 한, VPC로 시작된 인스턴스는 기본적으로 공유 하드웨어에서 실행됩니다.
dedicated	인스턴스 시작 중 host의 테넌시를 명시적으로 지정하지 않는 한, VPC로 시작된 인스턴스는 기본적으로 전용 인스턴스입니다. 인스턴스 시작 중 default의 테넌트를 지정할 수 없습니다.

VPC의 인스턴스 테넌시는 생성한 후에 dedicated에서 default로 변경할 수 있습니다. VPC의 인스턴스 테넌시는 dedicated로 변경할 수 없습니다.

전용 인스턴스를 생성하려면 다음 작업을 수행합니다.

- dedicated로 설정된 인스턴스 테넌시로 VPC를 생성합니다. 이 VPC로 시작된 모든 인스턴스는 전용 인스턴스입니다.
- default로 설정된 인스턴스 테넌시로 VPC를 생성하고 시작할 때 모든 인스턴스에 대해 dedicated 테넌시를 지정합니다.

전용 인스턴스 제한 사항

일부 AWS 서비스나 그 기능은 인스턴스 테넌시가 dedicated로 설정된 VPC에서 작동하지 않습니다. 서비스 설명서에서 이에 관한 제한 사항이 있는지 확인하십시오.

일부 인스턴스 유형은 인스턴스 테넌시가 dedicated로 설정된 VPC에서 시작할 수 없습니다. 지원되는 인스턴스 유형에 대한 자세한 내용은 [Amazon EC2 전용 인스턴스](#)를 참조하십시오.

Amazon EBS다음으로 바꿉니다.전용 인스턴스

Amazon EBS 지원 전용 인스턴스를 시작할 경우 EBS 볼륨은 단일 테넌트 하드웨어에서 실행되지 않습니다.

전용 테넌시를 포함하는 예약 인스턴스

전용 인스턴스를 시작하기 위한 용량을 충분히 확보하려면 예약 인스턴스를 구입하면 됩니다. 자세한 내용은 [예약 인스턴스 \(p. 205\)](#)를 참조하십시오.

전용 예약 인스턴스를 구입하면 대폭 할인된 사용 요금으로 전용 인스턴스를 VPC에서 시작할 수 있는 용량이 제공됩니다. 사용 요금 인하는 전용 테넌시 인스턴스를 시작할 경우에만 적용됩니다. 기본 테넌시가 있는 예약 인스턴스를 구입하면 `default` 테넌시가 있는 실행 중인 인스턴스에만 적용되고 `dedicated` 테넌시가 있는 실행 중인 인스턴스에는 적용되지 않습니다.

또한 예약 인스턴스를 구입한 후에는 수정 프로세스를 사용하여 테넌시를 변경할 수 없습니다. 그러나 전환 형 예약 인스턴스를 테넌시가 다른 새 전환형 예약 인스턴스와 교환할 수 있습니다.

전용 인스턴스 자동 조정

Amazon EC2 Auto Scaling을 사용하여 전용 인스턴스를 시작할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [VPC에서 Auto Scaling 인스턴스 시작](#)을 참조하십시오.

전용 인스턴스 자동 복구

기본 하드웨어 장애나 복구하는 데 AWS 개입이 필요한 문제로 인해 전용 인스턴스가 손상된 경우 이에 대해 자동 복구를 구성할 수 있습니다. 자세한 내용은 [인스턴스 복구 \(p. 460\)](#) 단원을 참조하십시오.

전용 스팟 인스턴스

스팟 인스턴스 요청을 생성할 때 `dedicated`의 테넌시를 지정하여 전용 스팟 인스턴스를 실행할 수 있습니다. 자세한 내용은 [스팟 인스턴스의 테넌시 지정 \(p. 256\)](#) 단원을 참조하십시오.

전용 인스턴스 가격

전용 인스턴스 요금은 온디맨드 인스턴스 요금과 다릅니다. 자세한 내용은 [Amazon EC2 전용 인스턴스 제품 페이지](#)를 참조하십시오.

전용 인스턴스를 사용한 버스트 가능한 성능 인스턴스

[the section called “성능 순간 확장 가능 인스턴스” \(p. 129\)](#)를 사용하여 전용 테넌시 하드웨어에서 실행하는 이점을 활용할 수 있습니다. T3 전용 인스턴스는 기본적으로 무제한 모드로 시작되며, 기본 수준의 CPU 성능 외에 버스트 기능이 있어 워크로드에 필요한 만큼 성능을 높일 수 있습니다. T3 기본 성능과 버스트 기능은 CPU 크레딧에 의해 좌우됩니다. T3 인스턴스 유형의 버스트 가능한 특성상, 최상의 성능을 위해 T3 인스턴스가 전용 하드웨어의 CPU 리소스를 어떻게 사용하는지 모니터링하는 것이 좋습니다. T3 전용 인스턴스는 임의의 CPU 동작을 나타내지만 평균 CPU 사용량이 기본 사용량 이하인 다양한 워크로드를 사용하는 고객을 대상으로 합니다. 자세한 내용은 [the section called “CPU 크레딧 및 기준 사용률” \(p. 130\)](#) 단원을 참조하십시오.

Amazon EC2에는 성능 변동성을 식별하고 수정할 수 있는 시스템이 있습니다. 그러나 상관 관계가 있는 CPU 사용 패턴을 가진 T3 전용 인스턴스를 여러 개 실행하면 단기적인 변동성이 발생할 수 있습니다. 이처럼 까다롭거나 상관 관계가 있는 워크로드의 경우 T3 전용 인스턴스 대신 M5 또는 M5a 전용 인스턴스를 사용하는 것이 좋습니다.

전용 인스턴스 작업

`dedicated` 인스턴스 테넌시로 VPC를 생성하여 해당 VPC로 시작되는 모든 인스턴스가 전용 인스턴스가 되게 합니다. 또는 시작되는 동안 인스턴스의 테넌시를 지정할 수 있습니다.

주제

- 전용 인스턴스 테넌시의 VPC 생성하기 (p. 352)
- VPC에서 전용 인스턴스 시작 (p. 352)
- 테넌시 정보 조회 (p. 353)
- 인스턴스의 테넌시 변경 (p. 354)
- VPC의 테넌시 변경 (p. 354)

전용 인스턴스 테넌시의 VPC 생성하기

VPC를 생성할 경우 VPC의 인스턴스 테넌시를 지정하는 옵션이 제공됩니다. Amazon VPC 콘솔을 사용하는 경우 VPC 마법사 또는 VPC 페이지를 사용하여 VPC를 생성할 수 있습니다.

전용 인스턴스 테넌시로 VPC를 생성하려면(VPC 마법사)

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 대시보드에서 Start VPC Wizard(VPC 마법사 시작)를 선택합니다.
3. VPC 구성을 선택한 후 선택을 선택합니다.
4. 마법사 다음 페이지의 하드웨어 테넌시 목록에서 전용을 선택합니다.
5. VPC 만들기를 선택합니다.

전용 인스턴스 테넌시로 VPC를 생성하려면(Create VPC 대화 상자)

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 VPC와 VPC 만들기를 차례로 선택합니다.
3. 테넌시에서 전용을 선택합니다. CIDR 블록을 지정하고 예, 생성을 선택합니다.

명령줄을 사용하여 VPC를 생성할 때 테넌시 옵션을 설정하려면

- `create-vpc`(AWS CLI)
- `New-EC2Vpc`(Windows PowerShell용 AWS 도구)

`dedicated` 인스턴스 테넌시가 있는 VPC에서 인스턴스를 시작하면 인스턴스 테넌시와 상관없이 인스턴스가 자동으로 전용 인스턴스가 됩니다.

VPC에서 전용 인스턴스 시작

Amazon EC2 인스턴스 시작 마법사를 사용하여 전용 인스턴스를 시작할 수 있습니다.

콘솔을 사용하여 기본 테넌시 VPC에서 전용 인스턴스를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 인스턴스 시작을 선택합니다.
3. Amazon 머신 이미지(AMI) 선택 페이지에서 AMI를 선택한 후 선택을 선택합니다.
4. 인스턴스 유형 선택 페이지에서 인스턴스 유형을 선택하고 Next: Configure Instance Details(다음: 인스턴스 세부 정보 구성)를 차례대로 선택합니다.

Note

전용 인스턴스를 지원하는 인스턴스 유형을 선택해야 합니다. 자세한 내용은 [Amazon EC2 전용 인스턴스](#)를 참조하십시오.

5. Configure Instance Details(인스턴스 세부 정보 구성) 페이지에서 VPC와 서브넷을 선택합니다. 테넌시 목록에서 Dedicated - Run a dedicated instance(전용 - 전용 인스턴스 실행)와 Next: Add Storage(다음: 스토리지 추가)를 차례로 선택합니다.

6. 마법사에 표시되는 지침에 따라 계속합니다. 인스턴스 시작 검토 페이지에서 옵션을 모두 검토했으면 시작을 선택하여 키 페어를 선택하고 전용 인스턴스를 시작합니다.

host 테넌시를 사용하여 인스턴스를 시작하는 자세한 내용은 [전용 호스트에서 인스턴스 시작 \(p. 326\)](#) 단원을 참조하십시오.

명령줄을 사용하여 시작 중에 인스턴스의 테넌시 옵션을 설정하려면

- [run-instances](#)(AWS CLI)
- [New-EC2Instance](#)(Windows PowerShell용 AWS 도구)

테넌시 정보 조회

콘솔을 사용하여 VPC의 테넌시 정보를 조회하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 VPC를 선택합니다.
3. 테넌시 열에서 해당 VPC의 인스턴스 테넌시를 확인합니다.
4. 테넌시 열이 표시되지 않는 경우 테이블 열 편집(기어 모양 아이콘)을 선택하고 열 표시/숨기기 대화 상자에서 테넌시를 선택한 후 닫기를 선택합니다.

콘솔을 사용하여 인스턴스의 테넌시 정보를 조회하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 테넌시 열에서 해당 인스턴스의 테넌시를 확인합니다.
4. 테넌시 열이 표시되지 않으면 다음 중 하나를 수행하십시오.
 - 열 표시/숨기기](기어 모양 아이콘)를 선택하고 열 표시/숨기기 대화 상자에서 테넌시를 선택한 후 닫기를 선택합니다.
 - 인스턴스를 선택합니다. 세부 정보 창의 설명 탭에 인스턴스의 정보와 그 테넌시가 표시됩니다.

명령줄을 사용하여 VPC의 테넌시를 나타내려면

- [describe-vpcs](#)(AWS CLI)
- [Get-EC2Vpc](#)(Windows PowerShell용 AWS 도구)

명령줄을 사용하여 인스턴스의 테넌시를 나타내려면

- [describe-instances](#) (AWS CLI)
- [Get-EC2Instance](#)(Windows PowerShell용 AWS 도구)

명령줄을 사용하여 예약 인스턴스의 테넌시 값을 나타내려면

- [describe-reserved-instances](#)(AWS CLI)
- [Get-EC2ReservedInstance](#)(Windows PowerShell용 AWS 도구)

명령줄을 사용하여 예약 인스턴스 상품의 테넌시 값을 나타내려면

- [describe-reserved-instances-offerings](#)(AWS CLI)

- [Get-EC2ReservedInstancesOffering](#)(Windows PowerShell용 AWS 도구)

인스턴스의 테넌시 변경

인스턴스 유형과 플랫폼에 따라 시작된 이후에 중지된 전용 인스턴스의 테넌시를 host로 변경할 수 있습니다. 다음에 인스턴스를 시작하면 계정에 할당된 에서 시작됩니다. 전용 호스트를 할당 및 사용하는 방법과 전용 호스트에서 사용할 수 있는 인스턴스 유형에 대한 자세한 내용은 [전용 호스트 작업 \(p. 323\)](#) 단원을 참조하십시오. 마찬가지로 시작된 이후에 중지된 전용 호스트 인스턴스의 테넌시를 dedicated로 변경할 수 있습니다. 인스턴스를 다음에 시작하면 Amazon에서 제어하는 단일 테넌트 하드웨어에서 시작됩니다.

콘솔을 사용하여 인스턴스의 테넌시를 변경하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
3. 작업, 인스턴스 상태, 중지를 차례로 선택합니다.
4. 작업, 인스턴스 설정, 인스턴스 배치 수정을 차례대로 선택합니다.
5. 테넌시 목록에서 인스턴스를 전용 하드웨어에서 실행할지 전용 호스트에서 실행할지를 선택합니다. 저장을 선택합니다.

명령줄을 사용하여 인스턴스의 테넌시 값을 수정하려면

- [modify-instance-placement](#)(AWS CLI)
- [Edit-EC2InstancePlacement](#) (Windows PowerShell용 AWS 도구)

VPC의 테넌시 변경

VPC의 인스턴스 테넌시 속성은 dedicated에서 default로 변경할 수 있습니다. VPC의 인스턴스 테넌시를 수정해도 VPC에 있는 기존 인스턴스의 테넌시에는 영향을 미치지 않습니다. 다음에 VPC에서 인스턴스를 시작할 때 시작 시 다르게 지정하지 않는 한 default 테넌시가 유지됩니다.

VPC의 인스턴스 테넌시 속성은 dedicated로 변경할 수 없습니다.

VPC의 인스턴스 테넌시 속성은 AWS CLI, AWS SDK 또는 Amazon EC2 API만 사용하여 수정할 수 있습니다.

AWS CLI를 사용하여 VPC의 인스턴스 테넌시 속성을 수정하려면

- [modify-vpc-tenancy](#) 명령을 사용하여 VPC의 ID와 인스턴스 테넌시 값을 지정합니다. 지원되는 유일한 값은 default입니다.

```
aws ec2 modify-vpc-tenancy --vpc-id vpc-1a2b3c4d --instance-tenancy default
```

온디맨드 용량 예약

온디맨드 용량 예약을 사용하면 특정 가용 영역의 Amazon EC2 인스턴스에 대해 원하는 기간만큼 용량을 예약할 수 있습니다. 따라서 Savings Plans 또는 리전 예약 인스턴스에서 제공하는 결제 할인과는 별도로 용량 예약을 생성 및 관리할 수 있습니다. 용량 예약을 생성하면 필요할 때 언제든지 필요한 만큼 EC2 용량에 액세스할 수 있으며 유지할 수 있습니다. 1년 또는 3년 기간의 약정에 가입하지 않고도 언제든지 용량 예약을 생성할 수 있으며 용량은 즉시 이용할 수 있습니다. 더 이상 필요하지 않으면 용량 예약을 취소하여 요금 발생을 중지합니다.

용량 예약을 생성할 때 다음을 지정합니다.

- 용량을 예약할 가용 영역입니다.

- 용량을 예약할 인스턴스 수입니다.
- 인스턴스 유형, 테넌시, platform/OS 등을 포함한 인스턴스 속성

용량 예약은 해당 속성과 일치하는 인스턴스에서만 사용할 수 있습니다. 기본적으로 속성과 일치하는 인스턴스를 실행하면 용량 예약이 자동으로 사용됩니다. 용량 예약의 속성과 일치하는 인스턴스를 실행하고 있지 않으면 속성과 일치하는 인스턴스를 시작할 때까지 사용되지 않은 상태로 유지됩니다.

또한 Savings Plans 및 리전 예약 인스턴스를 용량 예약과 함께 사용하여 결제 할인 혜택을 받을 수 있습니다. 용량 예약의 속성이 Savings Plan 또는 리전 예약 인스턴스의 속성과 일치하면 AWS에서 할인을 자동으로 적용합니다. 자세한 내용은 [결제 할인 \(p. 357\)](#) 단원을 참조하세요.

목차

- [용량 예약, 예약 인스턴스 및 Savings Plans의 차이점 \(p. 355\)](#)
- [지원되는 플랫폼 \(p. 356\)](#)
- [용량 예약 제한 \(p. 356\)](#)
- [용량 예약 제한 및 제약 \(p. 356\)](#)
- [용량 예약 요금 및 결제 \(p. 356\)](#)
- [용량 예약로 작업 \(p. 357\)](#)
- [공유된 용량 예약에 대한 작업 \(p. 366\)](#)
- [온디맨드 용량 예약에 대한 CloudWatch 지표 \(p. 369\)](#)

용량 예약, 예약 인스턴스 및 Savings Plans의 차이점

다음 표에 용량 예약, 예약 인스턴스 및 Savings Plans 간의 주요 차이점이 설명되어 있습니다.

	용량 예약	영역 예약 인스턴스	리전 예약 인스턴스	Savings Plans
기간	약정이 필요하지 않습니다. 필요할 때마다 생성하거나 취소할 수 있습니다.	고정 1년 또는 3년 약정 필요		
용량 혜택	특정 가용 영역에서 예약된 용량입니다.		가용 영역에서 용량을 예약하지 않습니다.	
결제 할인	결제 할인이 제공되지 않습니다. 용량 예약으로 시작된 인스턴스는 스탠다드 온디맨드 요율로 요금이 부과됩니다. 한편, Savings Plans 또는 리전 예약 인스턴스를 용량 예약과 함께 사용하면 결제 할인 혜택을 받을 수 있습니다. 영역 예약 인스턴스는 용량 예약에 적용되지 않습니다..	결제 할인 제공		
인스턴스 제한	리전당 온디맨드 인스턴스 제한으로 제한됩니다.	가용 영역당 20으로 제한됩니다. 제한 증가를 요청할 수 있습니다.	리전당 20으로 제한됩니다. 제한 증가를 요청할 수 있습니다.	제한이 없습니다.

자세한 내용은 다음 자료를 참조하십시오.

- [예약 인스턴스 \(p. 205\)](#)
- [Savings Plans 사용 설명서](#)

지원되는 플랫폼

인스턴스와 일치하는지 확인하려면 올바른 플랫폼으로 용량 예약을 생성해야 합니다. 용량 예약은 다음 플랫폼을 지원합니다.

- Windows
- SQL Server가 설치된 Windows
- SQL Server Web이 설치된 Windows
- SQL Server Standard가 설치된 Windows
- SQL Server Enterprise가 설치된 Windows

지원되는 Linux 플랫폼에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [지원되는 플랫폼](#)을 참조하십시오.

용량 예약 제한

용량 예약이 가능한 인스턴스 수는 계정의 온디맨드 인스턴스 제한을 기반으로 합니다. 이 제한이 허용하는 인스턴스 수에서 이미 실행 중인 인스턴스 수를 차감한 인스턴스 수에 대해 용량을 예약할 수 있습니다.

용량 예약 제한 및 제약

용량 예약을 생성하기 전에 다음 제한 및 제약에 유의하십시오.

- 활성 및 미사용 용량 예약 수는 온디맨드 인스턴스 제한에 포함됩니다.
- 용량 예약은 한 AWS 계정에서 다른 AWS 계정으로 이전할 수 없습니다. 그러나 용량 예약을 다른 AWS 계정과 공유할 수 있습니다. 자세한 내용은 [공유된 용량 예약에 대한 작업 \(p. 366\)](#) 단원을 참조하십시오.
- 영역 예약 인스턴스 결제 할인이 용량 예약에 적용되지 않습니다.
- 배치 그룹에서는 용량 예약을 생성할 수 없습니다.
- 용량 예약은 전용 호스트와 사용할 수 없습니다.
- 용량 예약은 로컬 영역과 사용할 수 없습니다.

용량 예약 요금 및 결제

용량 예약의 요금은 결제 옵션에 따라 다릅니다.

요금

용량 예약이 활성 상태인 경우 인스턴스 실행 여부에 따라 상응하는 온디맨드 요율로 요금이 부과됩니다. 예약을 사용하지 않는 경우 EC2 청구서에 사용되지 않은 예약으로 표시됩니다. 예약의 특성과 일치하는 인스턴스를 실행하는 경우 인스턴스 요금만 지불하고 예약 요금은 지불하지 않습니다. 선불금 또는 추가 요금은 없습니다.

예를 들어, 20개 `m4.1large` Linux 인스턴스에 대해 용량 예약을 생성하고 동일한 가용 영역에서 15개 `m4.1large` Linux 인스턴스를 실행할 경우 예약의 15개 활성 인스턴스와 사용되지 않은 5개 인스턴스에 대해 요금이 부과됩니다.

Savings Plans 및 리전 예약 인스턴스에 대한 결제 할인이 용량 예약에 적용됩니다. 자세한 내용은 [결제 할인 \(p. 357\)](#) 단원을 참조하십시오.

자세한 내용은 [Amazon EC2 요금을](#) 참조하십시오.

결제

용량 예약은 초당 요금으로 청구됩니다. 다시 말해서 사용 시간이 한 시간 미만이라도 요금이 부과됩니다. 예를 들어 예약이 24시간 15분 동안 계정에서 예약이 활성 상태로 유지되는 경우 24.25시간이 예약 시간으로 청구됩니다.

다음 예는 용량 예약 요금 청구 방식을 보여줍니다. 용량 예약이 m4.large Linux 인스턴스 하나에 대해 생성되었으며, 사용 시간당 \$0.10의 온디맨드 요율이 적용됩니다. 이 예에서는 용량 예약이 이 계정에 대해 다섯 시간 동안 활성 상태입니다. 처음 한 시간 동안 용량 예약이 사용되지 않았으므로, 사용되지 않은 한 시간에 대해서는 m4.large 인스턴스 유형의 스탠다드 온디맨드 요율로 요금이 청구됩니다. 2~5시간 동안은 m4.large 인스턴스에서 용량 예약이 사용됩니다. 이 시간 동안은 용량 예약에 대해 요금이 청구되지 않으며, 대신에 용량 예약을 사용하는 m4.large 인스턴스에 대해 이 계정에 요금이 청구됩니다. 여섯 번째 시간 동안은 용량 예약이 취소되었으므로 m4.large 인스턴스가 예약 용량 외부에서 일반적으로 실행됩니다. 해당 시간 동안은 m4.large 인스턴스 유형의 온디맨드 요율로 요금이 청구됩니다.

Hour	1	2	3	
Unused Capacity Reservation	\$0.10	\$0.00	\$0.00	\$
On-demand Instance Usage	\$0.00	\$0.10	\$0.10	\$
Hourly cost	\$0.10	\$0.10	\$0.10	\$

결제 할인

Savings Plans 및 리전 예약 인스턴스에 대한 결제 할인이 용량 예약에 적용됩니다. AWS는 일치하는 속성이 있는 용량 예약에 이러한 할인을 자동으로 적용합니다. 용량 예약이 인스턴스에 의해 사용되는 경우 해당 인스턴스에 할인이 적용됩니다. 할인은 사용되지 않은 용량 예약을 적용하기 전에 인스턴스 사용량에 우선적으로 적용됩니다.

영역 예약 인스턴스에 대한 결제 할인은 용량 예약에 적용되지 않습니다.

자세한 내용은 다음 자료를 참조하십시오.

- [예약 인스턴스 \(p. 205\)](#)
- [Savings Plans 사용 설명서](#)

청구서 보기

계정으로 청구되는 요금과 비용은 AWS Billing and Cost Management 콘솔에서 검토할 수 있습니다.

- 대시보드에는 계정에 대한 소비 요약이 표시됩니다.
- 청구서 페이지의 세부 정보에서 Elastic Compute Cloud 섹션과 리전을 확장하여 용량 예약에 대한 결제 정보를 가져옵니다.

요금을 온라인으로 확인하거나 CSV 파일을 다운로드할 수 있습니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 [용량 예약 항목](#)을 참조하십시오.

용량 예약으로 작업

용량 예약 사용을 시작하려면 필요 가용 영역에서 용량 예약을 생성합니다. 그런 다음, 인스턴스를 예약 용량으로 시작하거나 용량 사용률을 실시간으로 확인할 수 있으며, 필요 시 용량을 늘리거나 줄일 수 있습니다.

기본적으로 용량 예약은 일치하는 속성(인스턴스 유형, 플랫폼 및 가용 영역)이 있는 새 인스턴스 및 실행 중인 인스턴스와 자동으로 맞춰집니다. 즉, 일치하는 속성을 가진 인스턴스는 용량 예약에서 자동으로 실행됩니다. 하지만 용량 예약을 특정 워크로드에 지정할 수도 있습니다. 이렇게 하면 예약 용량으로 실행할 수 있는 인스턴스를 명시적으로 제어할 수 있습니다.

예약이 종료되는 방법을 지정할 수 있습니다. 용량 예약을 수동으로 취소하거나, 지정된 시간에 자동으로 예약을 종료하도록 선택할 수 있습니다. 종료 시간을 지정하면 지정된 시간부터 1시간 내에 용량 예약이 취소됩니다. 예를 들어 2019년 5월 31일 13:30:55를 지정하는 경우 용량 예약은 2019년 5월 31일 13:30:55 ~ 14:30:55에 종료됩니다. 예약이 종료된 후에는 더 이상 인스턴스를 용량 예약으로 지정할 수 없습니다. 예약 용량으로 실행 중인 인스턴스가 중단되지 않은 상태로 계속 실행됩니다. 용량 예약으로 지정된 인스턴스를 중지하는 경우 용량 예약 지정 기본 설정을 제거하거나 다른 용량 예약으로 지정되도록 구성할 때까지 해당 인스턴스를 다시 시작할 수 없습니다.

목차

- [용량 예약 생성 \(p. 358\)](#)
- [용량 예약 그룹 작업 \(p. 359\)](#)
- [인스턴스를 기존 용량 예약으로 시작 \(p. 362\)](#)
- [용량 예약 수정 \(p. 363\)](#)
- [인스턴스의 용량 예약 설정 수정 \(p. 364\)](#)
- [용량 예약 보기 \(p. 365\)](#)
- [용량 예약 취소 \(p. 365\)](#)

용량 예약 생성

용량 예약을 생성하고 나면 이 용량을 즉시 사용할 수 있습니다. 이 용량은 용량 예약이 활성 상태인 동안은 예약 상태로 유지되며 언제든지 인스턴스를 이 용량으로 시작할 수 있습니다. 용량 예약이 열려 있으면 일치하는 속성이 있는 새 인스턴스 및 기존 인스턴스는 용량 예약의 용량으로 자동 실행됩니다. 용량 예약이 *targeted*이면 인스턴스를 예약된 용량으로 실행하도록 지정해야 합니다.

다음 중 하나에 해당하는 경우 용량 예약 생성 요청이 실패할 수 있습니다.

- Amazon EC2에 용량이 충분하지 않아서 요청을 이행할 수 없습니다. 나중에 다시 시도하거나, 다른 가용 영역을 사용하거나, 용량을 줄여서 시도하십시오. 애플리케이션이 인스턴스 유형 및 크기 면에서 가변적인 경우 다른 인스턴스 속성으로 생성해 봅니다.
- 요청한 수량이 선택한 인스턴스 패밀리에 대한 온디맨드 인스턴스 제한을 초과합니다. 인스턴스 패밀리에 대한 온디맨드 인스턴스 제한을 늘리고 다시 시도하십시오. 자세한 내용은 [온디맨드 인스턴스 제한 \(p. 203\)](#) 단원을 참조하십시오.

콘솔을 사용하여 용량 예약을 생성하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 용량 예약을 선택한 후 용량 예약 생성을 선택합니다.
3. 용량 예약 생성 페이지의 인스턴스 세부 정보 섹션에서 다음 설정을 구성합니다. 시작하는 인스턴스 유형, 플랫폼, 인스턴스의 가용 영역은 여기서 지정한 인스턴스 유형, 플랫폼, 가용 영역과 일치해야 합니다. 그렇지 않으면 용량 예약이 적용되지 않습니다. 예를 들어 열려 있는 용량 예약이 일치하지 않으면 명시적으로 용량 예약을 대상으로 하는 인스턴스 시작이 실패합니다.
 - a. 인스턴스 유형 - 예약된 용량으로 시작할 인스턴스 유형입니다.
 - b. EBS 최적 인스턴스 시작 - EBS 최적 인스턴스 용량을 예약할지 여부를 지정합니다. 이 옵션은 특정 인스턴스 유형에 대해 기본적으로 선택됩니다. EBS 최적화 인스턴스에 대한 자세한 내용은 [Amazon Elastic Block Store \(p. 928\)](#) 단원을 참조하십시오.
 - c. 시작 시 인스턴스 스토어 연결 - 인스턴스를 용량 예약으로 시작된 인스턴스에서 임시 볼륨 수준 저장소를 사용할지 여부를 지정합니다. 인스턴스 스토어 볼륨의 데이터는 관련 인스턴스의 수명 기간 동안만 지속됩니다.

- d. 플랫폼 - 사용자 인스턴스에 사용할 운영 체제입니다. 자세한 내용은 [지원되는 플랫폼 \(p. 356\)](#) 단원을 참조하세요. 지원되는 Linux 플랫폼에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [지원되는 플랫폼](#)을 참조하십시오.
 - e. 가용 영역 - 용량을 예약할 가용 영역입니다.
 - f. 테넌시 - 공유 하드웨어(기본)를 실행할지 전용 인스턴스를 실행할지 지정합니다.
 - g. 수량—용량을 예약할 인스턴스 수입니다. 선택한 인스턴스 유형에 대해 남은 온디맨드 인스턴스 제한을 초과하는 수량을 지정하는 경우 이 요청이 거부됩니다.
4. 예약 세부 정보 섹션에서 다음 설정을 구성합니다.
- a. 예약 종료 - 다음 옵션 중 하나를 선택합니다.
 - 수동 - 명시적으로 취소할 때까지 용량을 예약합니다.
 - 지정 시간—지정된 날짜 및 시간에 용량 예약을 자동으로 취소합니다.
 - b. 인스턴스 자격 - 다음 옵션 중 하나를 선택합니다.
 - open - (기본값) 용량 예약이 일치하는 속성(인스턴스 유형, 플랫폼 및 가용 영역)이 있는 인스턴스를 검색합니다. 일치하는 속성이 있는 인스턴스를 시작할 경우 예약 용량으로 자동 배치됩니다.
 - targeted—용량 예약에서, 일치하는 속성(인스턴스 유형, 플랫폼 및 가용 영역)이 있고 예약을 명시적으로 지정하는 인스턴스만 허용합니다.
5. 예약 요청을 선택합니다.

AWS CLI를 사용하여 용량 예약을 생성하는 방법

`create-capacity-reservation` 명령을 사용합니다. 자세한 내용은 [지원되는 플랫폼 \(p. 356\)](#) 단원을 참조하세요. 지원되는 Linux 플랫폼에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [지원되는 플랫폼](#)을 참조하십시오.

예를 들어, 다음 명령은 us-east-1a 가용 영역에서 SQL Server AMI가 설치된 Windows를 실행하는 세 개의 m5.2xlarge 인스턴스에 대해 용량을 예약하는 용량 예약을 생성합니다.

```
aws ec2 create-capacity-reservation --instance-type m5.2xlarge --instance-platform Windows  
with SQL Server --availability-zone us-east-1a --instance-count 3
```

용량 예약 그룹 작업

AWS 리소스 그룹을 사용하여 리소스 그룹이라고 하는 용량 예약의 논리적 모음을 생성할 수 있습니다. 리소스 그룹은 모두 동일한 AWS 리전에 있는 AWS 리소스의 논리적 그룹입니다. 속성(인스턴스 유형, 플랫폼 및 가용 영역)이 서로 다른 여러 용량 예약을 단일 리소스 그룹에 포함할 수 있습니다.

용량 예약에 대한 리소스 그룹을 생성할 때 인스턴스를 개별 용량 예약 대신 용량 예약의 그룹으로 지정할 수 있습니다. 용량 예약의 그룹을 지정하는 인스턴스는 일치하는 속성(인스턴스 유형, 플랫폼 및 가용 영역) 및 가용 용량이 있는 그룹에서 용량 예약과 일치합니다. 그룹에 일치하는 속성 및 가용 용량이 있는 용량 예약이 없는 경우 인스턴스는 온디맨드 용량을 사용하여 실행됩니다. 이후 단계에서 대상 지정 그룹에 일치하는 용량 예약이 추가되면 인스턴스가 예약 용량과 자동으로 일치되고 해당 예약 용량으로 이동합니다.

그룹에서 용량 예약의 의도하지 않은 사용을 방지하려면 명시적으로 용량 예약을 지정하는 인스턴스만 허용하도록 그룹에서 용량 예약을 구성합니다. 이렇게 하려면 Amazon EC2 콘솔을 사용하여 용량 예약을 생성할 때 인스턴스 자격을 대상 지정(이전 콘솔)으로 설정하거나 이 예약을 지정하는 인스턴스만(새 콘솔)으로 설정합니다. AWS CLI를 사용하는 경우 용량 예약을 생성할 때 `--instance-match-criteria targeted`를 지정합니다. 이렇게 하면 그룹 또는 그룹의 용량 예약을 명시적으로 지정하는 인스턴스만 그룹에서 실행할 수 있습니다.

실행 중인 인스턴스가 있을 때 그룹의 용량 예약이 취소되거나 만료되는 경우 인스턴스는 일치하는 속성 및 가용 용량이 있는 그룹에서 다른 용량 예약 인스턴스로 자동으로 이동됩니다. 일치하는 속성 및 가용 용량이 있는 그룹에 남은 용량 예약이 없는 경우 인스턴스는 온디맨드 용량으로 실행됩니다. 이후 단계에서 대상 지정 그룹에 일치하는 용량 예약이 추가되면 인스턴스가 예약 용량으로 자동으로 이동합니다.

용량 예약에 대한 그룹을 생성하려면

[create-group](#) AWS CLI 명령을 사용합니다. name의 경우 설명이 포함된 그룹 이름을 제공하고 configuration에 대해 두 개의 Type 요청 파라미터를 지정합니다.

- AWS::EC2::CapacityReservationPool: 리소스 그룹이 인스턴스 시작 대상으로 지정될 수 있도록 합니다.
- allowed-resource-types가 AWS::EC2::CapacityReservation으로 설정된 AWS::ResourceGroups::Generic: 리소스 그룹이 용량 예약만 허용하도록 합니다.

예를 들어, 다음 명령은 이름이 MyCRGroup인 그룹을 생성합니다.

```
C:\> aws resource-groups create-group --name MyCRGroup --configuration
'{"Type":"AWS::EC2::CapacityReservationPool"}' '{"Type":"AWS::ResourceGroups::Generic",
"Parameters": [{"Name": "allowed-resource-types", "Values":
["AWS::EC2::CapacityReservation"]}]}'
```

다음은 출력의 예입니다.

```
{
  "GroupConfiguration": {
    "Status": "UPDATE_COMPLETE",
    "Configuration": [
      {
        "Type": "AWS::EC2::CapacityReservationPool"
      },
      {
        "Type": "AWS::ResourceGroups::Generic",
        "Parameters": [
          {
            "Values": [
              "AWS::EC2::CapacityReservation"
            ],
            "Name": "allowed-resource-types"
          }
        ]
      }
    ],
    "Group": {
      "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup",
      "Name": "MyCRGroup"
    }
  }
}
```

그룹에 용량 예약을 추가하려면

[group-resources](#) AWS CLI 명령을 사용합니다. group의 경우 용량 예약을 추가할 그룹의 이름을 지정하고 resources의 경우 추가할 용량 예약의 ARN을 지정합니다. 용량 예약을 여러 개 추가하려면 ARN을 공백으로 구분합니다.

예를 들어, 다음 명령은 이름이 MyCRGroup인 그룹에 두 개의 용량 예약을 추가합니다.

```
C:\> aws resource-groups group-resources --group MyCRGroup --resource-arns arn:aws:ec2:sa-
east-1:123456789012:capacity-reservation/cr-1234567890abcdef1 arn:aws:ec2:sa-
east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

다음은 출력의 예입니다.

```
{
```

```
"Failed": [],
"Succeeded": [
    "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1",
    "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
]
}
```

특정 그룹의 용량 예약을 보려면

[list-group-resources](#) AWS CLI 명령을 사용합니다. group의 경우 그룹 이름을 지정합니다.

예를 들어, 다음 명령은 이름이 MyCRGroup인 그룹에 용량 예약을 나열합니다.

```
C:\> aws resource-groups list-group-resources --group MyCRGroup
```

다음은 출력의 예입니다.

```
{
    "QueryErrors": [],
    "ResourceIdentifiers": [
        {
            "ResourceType": "AWS::EC2::CapacityReservation",
            "ResourceArn": "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1"
        },
        {
            "ResourceType": "AWS::EC2::CapacityReservation",
            "ResourceArn": "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
        }
    ]
}
```

특정 용량 예약이 추가된 그룹을 보려면(AWS CLI)

[get-groups-for-capacity-reservation](#) AWS CLI 명령을 사용합니다.

예를 들어, 다음 명령은 용량 예약 cr-1234567890abcdef1이 추가된 그룹을 나열합니다.

```
C:\> aws ec2 get-groups-for-capacity-reservation --capacity-reservation-id cr-1234567890abcdef1
```

다음은 출력의 예입니다.

```
{
    "CapacityReservationGroups": [
        {
            "OwnerId": "123456789012",
            "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup"
        }
    ]
}
```

특정 용량 예약이 추가된 그룹을 보려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 용량 예약을 선택하고 보려는 용량 예약을 선택한 다음 보기 선택합니다.

용량 예약이 추가된 그룹이 그룹 카드에 나열됩니다.

그룹에서 용량 예약을 제거하려면

[ungroup-resources](#) AWS CLI 명령을 사용합니다. group의 경우 용량 예약을 제거할 그룹의 ARN을 지정하고 resources의 경우 제거할 용량 예약의 ARN을 지정합니다. 용량 예약을 여러 개 제거하려면 ARN을 공백으로 구분합니다.

다음 예제에서는 이름이 용량 예약인 그룹에서 두 개의 MyCRGroup을 제거합니다.

```
C:\> aws resource-groups ungroup-resources --group MyCRGroup --  
resource-arns arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/  
cr-0e154d26a16094dd arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/  
cr-54321abcdef567890
```

다음은 출력의 예입니다.

```
{  
    "Failed": [],  
    "Succeeded": [  
        "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-0e154d26a16094dd",  
        "arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"  
    ]  
}
```

그룹을 삭제하려면

[delete-group](#) AWS CLI 명령을 사용합니다. group에 대해 삭제할 그룹의 이름을 제공합니다.

예를 들어, 다음 명령은 이름이 MyCRGroup인 그룹을 삭제합니다.

```
C:\> aws resource-groups delete-group --group MyCRGroup
```

다음은 출력의 예입니다.

```
{  
    "Group": {  
        "GroupArn": "arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup",  
        "Name": "MyCRGroup"  
    }  
}
```

인스턴스를 기준 용량 예약으로 시작

인스턴스를 시작할 때 인스턴스를 임의 open 용량 예약, 특정 용량 예약 또는 용량 예약 그룹으로 시작할지 여부를 지정할 수 있습니다. 인스턴스는 일치하는 속성(인스턴스 유형, 플랫폼 및 가용 영역) 및 충분한 용량이 있는 용량 예약로만 시작할 수 있습니다. 또는 일치하는 속성 및 가용 용량이 있는 open 용량 예약이 있다라도 용량 예약에서 실행되지 않도록 인스턴스를 구성할 수 있습니다.

인스턴스를 용량 예약으로 시작하면 시작된 인스턴스 수만큼 가용 용량이 감소됩니다. 예를 들어, 인스턴스 세 개를 시작할 경우 용량 예약의 가용 용량이 3만큼 감소됩니다.

콘솔을 사용하여 기준 용량 예약으로 인스턴스를 시작하는 방법

1. 대시보드 또는 인스턴스에서 인스턴스 시작을 선택하여 인스턴스 시작 마법사를 엽니다.
2. Amazon 머신 이미지(AMI)와 인스턴스 유형을 선택합니다.
3. 인스턴스 세부 정보 구성을 설정합니다. 용량 예약에서 다음 옵션 중 하나를 선택합니다.
 - 없음 - 인스턴스가 용량 예약으로 시작되지 않도록 합니다. 인스턴스는 온디맨드 용량으로 실행됩니다.

- 열기 - 일치하는 속성과, 선택한 인스턴스 수에 맞는 용량이 있는 용량 예약으로 인스턴스를 시작합니다. 충분한 용량이 있는 적절한 용량 예약이 없는 경우 인스턴스는 온디맨드 용량을 사용합니다.
 - ID별 대상 지정 — 선택한 용량 예약로 인스턴스를 시작합니다. 선택한 용량 예약의 용량이 선택한 인스턴스 수에 맞게 충분하지 않으면 인스턴스가 시작되지 않습니다.
 - 그룹별 대상 지정 — 선택한 용량 예약 그룹에서 일치하는 속성 및 가용 용량이 있는 용량 예약로 인스턴스를 시작합니다. 선택한 그룹에 일치하는 속성 및 가용 용량이 있는 용량 예약이 없는 경우 인스턴스는 온디맨드 용량으로 시작됩니다.
4. 나머지 단계를 완료하여 인스턴스를 시작합니다.

AWS CLI를 사용하여 기존 용량 예약으로 인스턴스를 시작하는 방법

`run-instances` 명령을 사용하여 `--capacity-reservation-specification` 파라미터를 지정합니다.

다음 예제에서는 일치하는 속성과 가용 용량이 있는 열려 있는 용량 예약으로 `t2.micro` 인스턴스가 시작됩니다.

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t2.micro --key-name MyKeyPair --subnet-id subnet-1234567890abcdef1 --capacity-reservation-specification CapacityReservationPreference=open
```

다음 예제에서는 `t2.micro` 인스턴스가 `targeted` 용량 예약으로 시작됩니다.

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t2.micro --key-name MyKeyPair --subnet-id subnet-1234567890abcdef1 --capacity-reservation-specification CapacityReservationTarget={CapacityReservationId=cr-a1234567}
```

다음 예제에서는 `t2.micro` 인스턴스가 용량 예약 그룹으로 시작됩니다.

```
aws ec2 run-instances --image-id ami-abc12345 --count 1 --instance-type t2.micro --key-name MyKeyPair --subnet-id subnet-1234567890abcdef1 --capacity-reservation-specification CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-groups:us-west-1:123456789012:group/my-cr-group}
```

용량 예약 설정

생성한 후 활성 용량 예약의 속성을 변경할 수 있습니다. 만료되었거나 명시적으로 취소한 후에는 용량 예약을 수정할 수 없습니다.

용량 예약을 수정하는 경우, 수량 늘리기/줄이기 또는 해제 방식 변경만 가능합니다. 용량 예약의 인스턴스 유형, EBS 최적화, 인스턴스 스토어 설정, 플랫폼, 가용 영역 또는 인스턴스 자격은 변경할 수 없습니다. 이러한 속성을 수정해야 하는 경우에는 예약을 취소하고 나서 필요한 속성으로 예약을 다시 생성하는 것이 좋습니다.

선택한 인스턴스 유형에 대해 남은 온디맨드 인스턴스 제한을 초과하는 새 수량을 지정하는 경우 업데이트가 실패합니다.

콘솔을 사용하여 용량 예약을 수정하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 용량 예약을 선택한 후, 수정할 용량 예약을 선택하고 나서 편집을 선택합니다.
3. 필요에 따라 수량 또는 예약 종료 옵션을 수정하고 변경 사항 저장을 선택합니다.

AWS CLI를 사용하여 용량 예약을 수정하는 방법

`modify-capacity-reservations` 명령을 사용합니다.

예를 들어, 다음 명령은 용량 예약을 수정하여 8개의 인스턴스에 대한 용량을 예약합니다.

```
aws ec2 modify-capacity-reservation --capacity-reservation-id cr-1234567890abcdef0 --  
instance-count 8
```

인스턴스의 용량 예약 설정 수정

중지된 인스턴트에 대한 다음 용량 예약 설정은 다음과 같이 언제든지 수정할 수 있습니다.

- 일치하는 속성(인스턴스 유형, 플랫폼 및 가용 영역) 및 가용 용량이 있는 용량 예약에서 시작합니다.
- 특정 용량 예약에서 인스턴스를 시작합니다.
- 용량 예약 그룹에 일치하는 속성 및 가용 용량이 있는 용량 예약에서 시작합니다.
- 인스턴스가 용량 예약에서 시작되지 않도록 합니다.

콘솔을 사용하여 인스턴스의 용량 예약 설정을 수정하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 인스턴스를 선택한 후 수정할 인스턴스를 선택합니다. 인스턴스가 아직 중지되지 않은 경우 해당 인스턴스를 중지합니다.
3. 작업, 용량 예약 설정 수정을 선택합니다.
4. 용량 예약에서 다음 옵션 중 하나를 선택합니다.
 - 열기 - 일치하는 속성과, 선택한 인스턴스 수에 맞는 용량이 있는 용량 예약으로 인스턴스를 시작합니다. 충분한 용량이 있는 적절한 용량 예약이 없는 경우 인스턴스는 온디맨드 용량을 사용합니다.
 - 없음 - 인스턴스가 용량 예약으로 시작되지 않도록 합니다. 인스턴스는 온디맨드 용량으로 실행됩니다.
 - 용량 예약 지정 — 선택한 용량 예약로 인스턴스를 시작합니다. 선택한 용량 예약의 용량이 선택한 인스턴스 수에 맞게 충분하지 않으면 인스턴스가 시작되지 않습니다.
 - 용량 예약 그룹 지정 — 선택한 용량 예약 그룹에 일치하는 속성 및 가용 용량이 있는 용량 예약에서 인스턴스를 시작합니다. 선택한 그룹에 일치하는 속성 및 가용 용량이 있는 용량 예약이 없는 경우 인스턴스는 온디맨드 용량으로 시작됩니다.

AWS CLI를 사용하여 인스턴스의 용량 예약 설정을 수정하는 방법

`modify-instance-capacity-reservation-attributes` 명령을 사용합니다.

예를 들어, 다음 명령은 인스턴스의 용량 예약 설정을 open 또는 none으로 변경합니다.

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-id i-1234567890abcdef0  
--capacity-reservation-specification CapacityReservationPreference=none|open
```

예를 들어, 다음 명령은 인스턴스를 수정하여 특정 용량 예약을 지정합니다.

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-  
id i-1234567890abcdef0 --capacity-reservation-specification  
CapacityReservationTarget={CapacityReservationId=cr-1234567890abcdef0}
```

예를 들어, 다음 명령은 인스턴스를 수정하여 특정 용량 예약 그룹을 지정합니다.

```
aws ec2 modify-instance-capacity-reservation-attributes --instance-  
id i-1234567890abcdef0 --capacity-reservation-specification  
CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-groups:us-  
west-1:123456789012:group/my-cr-group}
```

용량 예약 보기

용량 예약에는 다음과 같은 잠재적인 상태가 있습니다.

- **active** - 용량을 사용할 수 있습니다.
- **expired** - 용량 예약이 예약 요청 시 지정한 날짜 및 시간에 자동으로 만료됩니다. 예약 용량을 더 이상 사용할 수 없습니다.
- **cancelled** - 용량 예약이 수동으로 취소되었습니다. 예약 용량을 더 이상 사용할 수 없습니다.
- **pending** - 용량 예약 요청에 성공했지만 용량 프로비저닝이 여전히 대기 중입니다.
- **failed** - 용량 예약 요청에 실패했습니다. 잘못된 요청 파라미터, 용량 제약 조건 또는 인스턴스 제한 제약 조건이 요청 실패의 원인일 수 있습니다. 60분 동안 실패한 요청을 볼 수 있습니다.

콘솔을 사용하여 용량 예약을 보는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 용량 예약을 선택하고 확인할 용량 예약을 선택합니다.
3. 이 예약에 대하여 시작된 인스턴스 보기를 선택합니다.

AWS CLI를 사용하여 용량 예약을 보는 방법

`describe-capacity-reservations` 명령을 사용합니다.

예를 들어, 다음 명령은 모두 용량 예약을 설명합니다.

```
aws ec2 describe-capacity-reservations
```

용량 예약 취소

예약된 용량이 더 이상 필요하지 않을 경우 언제라도 용량 예약을 취소할 수 있습니다. 용량 예약을 취소할 경우 해당 용량이 즉시 해지되고 더 이상 사용 용량으로 예약되지 않습니다.

인스턴스를 실행 중인 용량 예약과 빈 용량 예약을 취소할 수 있습니다. 인스턴스를 실행 중인 용량 예약을 취소할 경우 인스턴스가 일반적으로 표준 온디맨드 인스턴스 요율이나 할인된 요율(일치하는 Savings Plan 또는 리전 예약 인스턴스가 있는 경우)으로 용량 예약 외부에서 계속 정상적으로 실행됩니다.

용량 예약을 취소한 후에는 해당 용량 예약으로 지정된 인스턴스를 더 이상 시작할 수 없습니다. 다른 용량 예약으로 지정되거나, 일치하는 속성 및 충분한 용량이 있는 '열린' 용량 예약으로 시작하거나, 용량 예약으로 시작하지 않도록 이러한 인스턴스를 수정합니다. 자세한 내용은 [인스턴스의 용량 예약 설정 수정 \(p. 364\)](#) 단원을 참조하십시오.

콘솔을 사용하여 용량 예약을 취소하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 용량 예약을 선택하고 취소할 용량 예약을 선택합니다.
3. 예약 취소, 예약 취소를 선택합니다.

AWS CLI를 사용하여 용량 예약을 취소하는 방법

`cancel-capacity-reservation` 명령을 사용합니다.

예를 들어, 다음 명령은 ID가 cr-1234567890abcdef0인 용량 예약을 취소합니다.

```
aws ec2 cancel-capacity-reservation --capacity-reservation-id cr-1234567890abcdef0
```

공유된 용량 예약에 대한 작업

용량 예약 공유를 통해 용량 예약 소유자는 예약된 용량을 다른 AWS 계정이나 AWS 조직 내에서 공유할 수 있습니다. 이를 통해 용량 예약을 중앙 집중식으로 생성 및 관리하고 예약된 용량을 여러 AWS 계정 또는 AWS 조직 내에서 공유할 수 있습니다.

이 모델에서 용량 예약(소유자)를 소유한 AWS 계정은 다른 AWS 계정(소비자)과 이를 공유합니다. 소비자는 자신의 계정에서 자신이 소유한 용량 예약에서 인스턴스를 시작하는 동일한 방식으로 인스턴스를 용량 예약에서 시작할 수 있습니다. 용량 예약 소유자는 용량 예약 및 해당 위치에서 시작한 인스턴스를 관리해야 합니다. 소유자는 공유한 용량 예약로 소비자가 시작한 인스턴스를 수정할 수 없습니다. 소비자는 공유된 용량 예약 위치에서 시작한 인스턴스를 관리해야 합니다. 소비자는 다른 소비자 또는 용량 예약 소유자가 소유한 인스턴스를 보거나 수정할 수 없습니다.

용량 예약 소유자는 다음과 용량 예약을 공유할 수 있습니다.

- AWS 조직 내부 또는 외부의 특정 AWS 계정
- AWS 조직 내부의 조직 단위
- 전체 AWS 조직

목차

- [용량 예약 공유를 위한 사전 조건 \(p. 366\)](#)
- [관련 서비스 \(p. 366\)](#)
- [여러 가용 영역 공유 \(p. 367\)](#)
- [용량 예약 공유 \(p. 367\)](#)
- [용량 예약 공유 중지 \(p. 368\)](#)
- [공유된 용량 예약 식별 \(p. 368\)](#)
- [공유 용량 예약 사용량 보기 \(p. 368\)](#)
- [공유된 용량 예약 권한 \(p. 369\)](#)
- [결제 및 측정 \(p. 369\)](#)
- [인스턴스 제한 \(p. 369\)](#)

용량 예약 공유를 위한 사전 조건

- 용량 예약을 공유하려면 AWS 계정에서 소유하고 있어야 합니다. 나와 공유된 용량 예약을 공유할 수 없습니다.
- 공유 테넌시 인스턴스에 대해서만 용량 예약을 공유할 수 있습니다. 전용 테넌시 인스턴스에 대해 용량 예약을 공유할 수 없습니다.
- 결제 내역이 제한된 새 AWS 계정 또는 AWS 계정에서는 용량 예약 공유를 사용할 수 없습니다. 유자격 마스터(지급인) 계정에 링크되어 있거나 AWS 조직을 통해 연결된 새 계정은 이 제한에서 제외됩니다.
- AWS 조직 또는 AWS 조직의 조직 단위와 용량 예약을 공유하려면 AWS Organizations를 사용하여 공유를 사용하도록 설정해야 합니다. 자세한 내용은 AWS RAM 사용 설명서의 [AWS Organizations를 사용하여 공유 사용 설정](#)을 참조하십시오.

관련 서비스

용량 예약 공유는 AWS 리소스 액세스 관리자(AWS RAM)와 통합됩니다. AWS RAM은 어떤 AWS 계정이나 AWS Organizations를 통해 나의 AWS 리소스를 공유할 수 있게 해주는 서비스입니다. AWS RAM을 사용하여 리소스 공유로 생성한 사용자 소유 리소스를 공유할 수 있습니다. 리소스 공유는 공유할 리소스와 공유 대상 소비자를 지정합니다. 소비자는 개인 AWS 계정 또는 조직 단위 또는 AWS Organizations의 전체 조직일 수 있습니다.

AWS RAM에 대한 자세한 내용은 [AWS RAM 사용 설명서](#)를 참조하십시오.

여러 가용 영역 공유

리전의 가용 영역에 걸쳐 리소스가 배포될 수 있도록 AWS는 각 계정의 이름에 가용 영역을 독립적으로 매핑합니다. 이로 인해 계정 전체에서 가용 영역 이름의 차이가 발생할 수 있습니다. 예를 들어 AWS AWS 계정의 us-east-1a 가용 영역은 다른 AWS 계정에 대한 us-east-1a 가용 영역과 위치가 동일하지 않을 수 있습니다.

계정과 관련된 용량 예약 상대의 위치를 확인하려면 가용 영역 ID(AZ ID)를 사용해야 합니다. AZ ID는 모든 AWS 계정의 가용 영역에 대한 고유하고 일관된 식별자입니다. 예를 들어, use1-az1은 us-east-1 리전의 AZ ID이고, 모든 AWS 계정에서 위치가 동일합니다.

계정의 가용 영역에 대한 AZ ID 보는 방법

1. <https://console.aws.amazon.com/ram>에서 AWS RAM 콘솔을 엽니다.
2. 현재 지역의 AZ ID는 화면의 오른쪽에 있는 사용자 AZ ID 패널에 표시됩니다.

용량 예약 공유

자신이 소유한 용량 예약을 다른 AWS 계정과 공유하면 예약된 용량으로 인스턴스를 시작할 수 있습니다. 열린 용량 예약을 공유하는 경우 의도하지 않은 용량 예약 사용으로 이어질 수 있으므로 다음 사항을 명심하십시오.

- 소비자가 용량 예약의 속성과 일치하는 인스턴스를 실행하고 CapacityReservationPreference 파라미터를 open으로 설정하고 예약된 용량에서 아직 실행하지 않는 경우, 공유된 용량 예약을 자동으로 사용합니다.
- 소비자가 일치하는 속성(인스턴스 유형, 플랫폼 및 가용 영역)이 있는 인스턴스를 실행하고 CapacityReservationPreference 매개 변수를 open으로 설정하면 자동으로 공유된 용량 예약로 실행됩니다.

용량 예약을 공유하려면 리소스 공유에 추가해야 합니다. AWS RAM 리소스 공유는 여러 AWS 계정에서 리소스를 공유할 수 있게 해주는 리소스입니다. 리소스 공유는 공유할 리소스와 공유 대상 소비자를 지정합니다. Amazon EC2 콘솔을 사용하여 용량 예약을 공유하면 기존 리소스 공유에 추가합니다. 새 리소스 공유에 용량 예약을 추가하려면 [AWS RAM 콘솔](#)을 사용하여 리소스 공유를 만들어야 합니다.

AWS Organizations의 조직에 속해 있고 조직 내의 공유가 활성화되어 있으면 조직의 소비자는 공유된 용량 예약에 대한 액세스 권한이 자동으로 부여됩니다. 그렇지 않으면 소비자는 리소스 공유에 가입하라는 초대장을 받고 초대를 수락한 후 공유 용량 예약에 대한 액세스 권한이 부여됩니다.

Amazon EC2 콘솔, AWS RAM 콘솔 또는 AWS CLI를 사용하여 소유하고 있는 용량 예약을 공유할 수 있습니다.

Amazon EC2 콘솔을 사용하여 소유하고 있는 용량 예약을 공유하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 용량 예약을 선택합니다.
3. 공유할 용량 예약을 선택하고 작업, 공유 예약을 선택하십시오.
4. 용량 예약을 추가할 리소스 공유를 선택하고 용량 예약 공유를 선택하십시오.

소비자가 공유 용량 예약에 액세스하려면 몇 분이 걸릴 수 있습니다.

AWS RAM 콘솔을 사용하여 소유하고 있는 용량 예약을 공유하려면

AWS RAM 사용 설명서에서 [리소스 공유 만들기](#)를 참조하십시오.

AWS CLI를 사용하여 소유하고 있는 용량 예약을 공유하려면

[create-resource-share](#) 명령을 사용합니다.

용량 예약 공유 중지

용량 예약 소유자는 언제든지 용량 예약의 공유를 중지할 수 있습니다. 다음 규칙이 적용됩니다.

- 공유가 중지될 때 공유 용량으로 실행 중인 소비자 소유의 인스턴스는 예약된 용량 밖에서 정상적으로 계속 실행되며 용량은 Amazon EC2 용량 가용성에 따라 용량 예약로 복원됩니다.
- 용량 예약이 공유된 사용자는 더 이상 예약된 용량에 새로운 인스턴스를 시작할 수 없습니다.

소유하고 있는 용량 예약의 공유를 중지하려면 리소스 공유에서 제거해야 합니다. 이를 위해 Amazon EC2 콘솔, AWS RAM 콘솔 또는 AWS CLI를 사용할 수 있습니다.

Amazon EC2 콘솔을 사용하여 소유하고 있는 용량 예약의 공유를 중지하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 용량 예약을 선택합니다.
3. 용량 예약을 선택하고 공유(Sharing) 탭을 선택합니다.
4. 공유 탭에는 용량 예약이 추가된 리소스 공유가 나열됩니다. 용량 예약을 제거할 리소스 공유를 선택하고 리소스 공유에서 제거를 선택하십시오.

AWS RAM 콘솔을 사용하여 소유하고 있는 용량 예약의 공유를 중지하려면

AWS RAM 사용 설명서에서 [리소스 공유 업데이트](#)를 참조하십시오.

AWS CLI를 사용하여 소유하고 있는 용량 예약의 공유를 중지하려면

[disassociate-resource-share](#) 명령을 사용합니다.

공유된 용량 예약 식별

소유자와 소비자는 Amazon EC2 콘솔 및 AWS CLI를 사용하여 용량 예약 공유를 식별할 수 있으며

Amazon EC2 콘솔을 사용하여 용량 예약 공유를 식별하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 용량 예약을 선택합니다. 화면에는 내가 소유하고 있는 용량 예약과 나와 공유된 용량 예약이 나열됩니다. 소유자 옆에는 용량 예약 소유자의 AWS 계정 ID가 표시됩니다. AWS 계정 ID 옆의 (me)는 사용자가 소유자임을 나타냅니다.

AWS CLI를 사용하여 용량 예약 공유를 식별하려면

[describe-capacity-reservations](#) 명령을 사용합니다. 이 명령은 내가 소유한 용량 예약과 나와 공유된 용량 예약을 반환합니다. `OwnerId`는 용량 예약 소유자의 AWS 계정 ID를 보여줍니다.

공유 용량 예약 사용량 보기

공유 용량 예약의 소유자는 Amazon EC2 콘솔과 AWS CLI를 사용하여 언제든지 사용량을 볼 수 있습니다.

Amazon EC2 콘솔을 사용하여 용량 예약 사용량을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 용량 예약을 선택합니다.
3. 사용량을 볼 수 있는 용량 예약 항목을 선택하고 사용량 탭을 선택하십시오.

AWS 계정 ID 열에는 현재 용량 예약을 사용하는 소비자의 계정 ID가 표시됩니다. 시작된 인스턴스 열에는 현재 각 사용자가 예약된 용량으로 실행 중인 인스턴스 수가 표시됩니다.

AWS CLI를 사용하여 용량 예약 사용량을 보려면

[get-capacity-reservation-usage](#) 명령을 사용하십시오. `AccountId`는 용량 예약을 사용하는 계정의 계정 ID를 보여줍니다. `UsedInstanceCount`는 현재 예약된 용량에서 실행 중인 인스턴스의 수를 보여줍니다.

공유된 용량 예약 권한

소유자에 대한 권한

소유자는 공유 용량 예약을 관리하고 취소하는 일을 담당합니다. 소유자는 다른 계정이 소유한 공유 용량 예약에서 실행 중인 인스턴스를 수정할 수 없습니다. 소유자는 용량 예약 공유로 실행되는 인스턴스를 관리해야 합니다.

소비자에 대한 권한

소비자는 공유 용량 예약을 실행하는 인스턴스를 관리해야 합니다. 소비자는 어떤 식으로든 공유 용량 예약을 수정할 수 없으며 다른 소비자 또는 용량 예약 소유자가 소유한 인스턴스를 보거나 수정할 수 없습니다.

결제 및 측정

용량 예약 공유에 대한 추가 비용은 없습니다.

용량 예약 소유자에게는 용량 예약 내부에서 실행하는 인스턴스 및 사용되지 않은 예약 용량에 대해 요금이 청구됩니다. 소비자는 용량 예약 공유 내에서 실행되는 인스턴스에 대해 요금이 청구됩니다.

인스턴스 제한

모든 용량 예약 사용량은 용량 예약 소유자의 온디맨드 인스턴스 한도에 포함됩니다. 여기에는 다음이 포함됩니다.

- 미사용 예약 용량
- 용량 예약 소유자가 소유한 인스턴스 별 사용량
- 소비자가 소유한 인스턴스 별 사용량

소비자가 공유하는 용량으로 시작된 인스턴스는 용량 예약 소유자의 온디맨드 인스턴스 한도에 포함됩니다. 소비자의 인스턴스 제한은 자신의 온디맨드 인스턴스 제한과 액세스할 수 있는 공유 용량 예약에서 사용 가능한 용량의 합계입니다.

온디맨드 용량 예약에 대한 CloudWatch 지표

CloudWatch 지표를 사용하면 사용량 임계값이 충족될 때 알림을 보내도록 CloudWatch 경보를 설정하여 용량 예약을 효율적으로 모니터링하고 사용하지 않은 용량을 식별할 수 있습니다. 이를 통해 일정한 용량 예약 볼륨을 유지하고 더 높은 수준의 사용률을 달성할 수 있습니다.

온디맨드 용량 예약은 5분마다 CloudWatch로 지표를 전송합니다. 5분 미만 동안 활성 상태인 용량 예약에는 지표가 지원되지 않습니다.

CloudWatch 콘솔에서 지표를 보는 방법에 대한 자세한 내용은 [Amazon CloudWatch 지표 사용](#)을 참조하십시오. 경보 생성에 대한 자세한 내용은 [Amazon CloudWatch 경보 생성](#)을 참조하십시오.

목차

- [용량 예약 사용량 지표 \(p. 370\)](#)
- [용량 예약 지표 차원 \(p. 370\)](#)

- 용량 예약에 대한 CloudWatch 지표 보기 (p. 370)

용량 예약 사용량 지표

AWS/EC2CapacityReservations 네임스페이스에는 예약에 대해 지정한 임계값 내에서 온디맨드 용량을 모니터링하고 유지 관리하는데 사용할 수 있는 다음과 같은 사용량 지표가 포함됩니다.

지표	설명
UsedInstanceCount	현재 사용 중인 인스턴스 수입니다. 단위: 수
AvailableInstanceCount	사용 가능한 인스턴스 수입니다. 단위: 수
TotalInstanceCount	예약한 총 인스턴스 수입니다. 단위: 수
InstanceUtilization	현재 사용 중인 예약 용량 인스턴스의 비율입니다. 단위: 백분율

용량 예약 지표 차원

다음 차원을 사용하여 이전 표에 나열된 지표를 구체화할 수 있습니다.

차원	설명
CapacityReservationId	이 전역적으로 고유한 차원은 식별된 용량 예약에 대해 요청한 데이터만 필터링합니다.

용량 예약에 대한 CloudWatch 지표 보기

지표는 먼저 서비스 네임스페이스별로 그룹화된 다음, 각 네임스페이스 내에서 지원되는 차원별로 그룹화됩니다. 다음 절차에 따라 용량 예약에 대한 지표를 볼 수 있습니다.

CloudWatch 콘솔을 사용하여 용량 예약 지표를 보려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 필요한 경우 리전을 변경합니다. 탐색 모음에서 용량 예약이 상주하는 리전을 선택합니다. 자세한 정보는 [리전 및 엔드포인트](#)를 참조하십시오.
3. 탐색 창에서 지표를 선택합니다.
4. 모든 지표에 대해 EC2 용량 예약을 선택합니다.
5. 용량 예약 기준 지표 차원을 선택합니다. 지표는 CapacityReservationId를 기준으로 그룹화됩니다.
6. 지표를 정렬하려면 열 머리글을 사용합니다. 측정치를 그래프로 표시하려면 측정치 옆에 있는 확인란을 선택합니다.

용량 예약 지표를 보려면(AWS CLI)

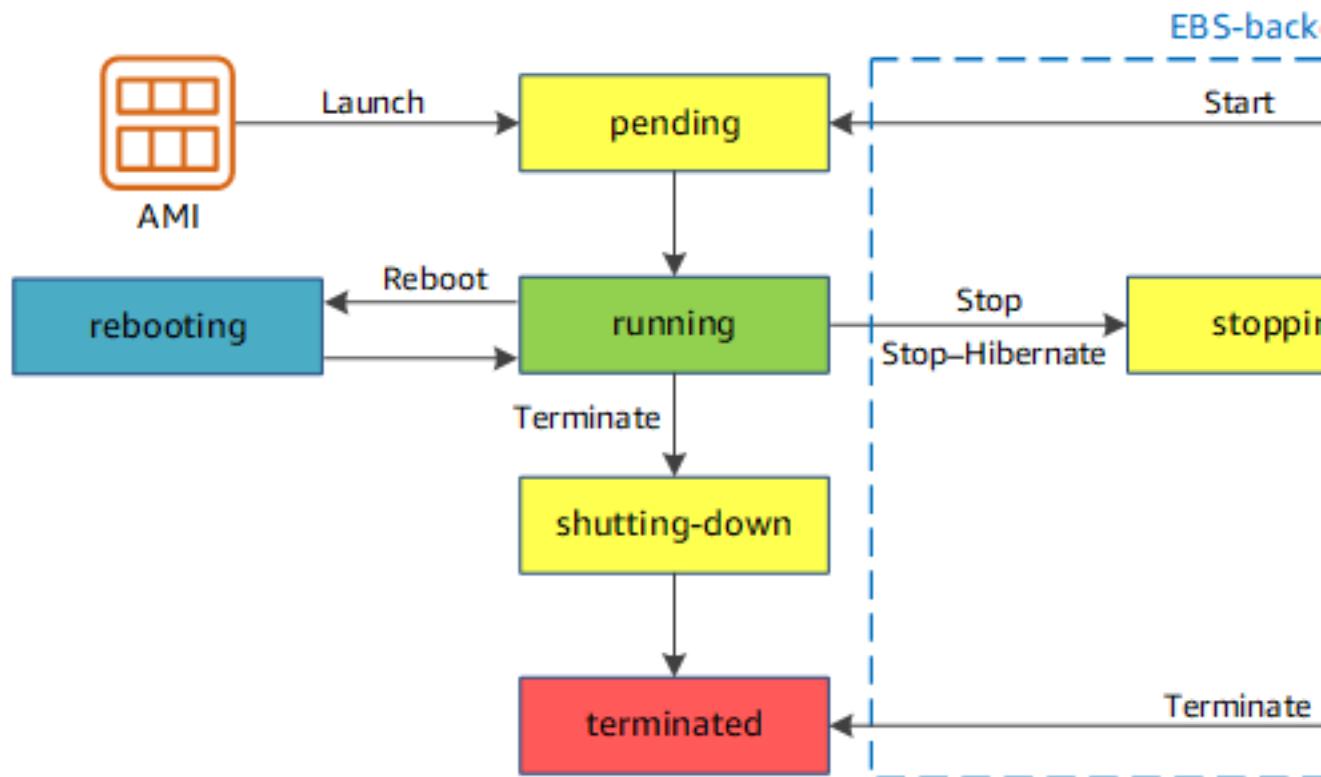
다음 [list-metrics](#) 명령을 사용합니다.

```
aws cloudwatch list-metrics --namespace "AWS/EC2CapacityReservations"
```

인스턴스 수명 주기

Amazon EC2 인스턴스는 시작한 순간부터 종료될 때까지 다양한 상태로 전환됩니다.

다음 그림은 인스턴스 상태 간 전환을 나타냅니다.



다음 표는 각 인스턴스 상태에 대한 간략한 정보와 청구 여부를 설명합니다.

Note

이 표는 인스턴스 사용에 관련된 청구만 보여줍니다. Amazon EBS 볼륨 및 탄력적 IP 주소와 같은 일부 AWS 리소스는 인스턴스의 상태와 상관없이 요금을 부과합니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서에서 [예기치 않은 비용 방지](#) 단원을 참조하십시오.

인스턴스 상태	설명	인스턴스 사용 요금
pending	인스턴스는 running 상태로 될 준비를 하고 있습니다. 인스턴스를 처음 시작하거나 stopped 상태의 인스턴스를 다시 시작하면 pending 상태가 됩니다.	미청구
running	인스턴스를 실행하고 사용할 준비가 되었습니다.	청구

인스턴스 상태	설명	인스턴스 사용 요금
stopping	인스턴스가 중지 또는 종지-최대 절전 모드로 전환할 준비를 하고 있습니다.	중지 준비 중인 경우 미청구 최대 절전 모드로 전환 준비 중인 경우 청구
stopped	인스턴스가 종료되고 사용이 불가합니다. 언제든지 인스턴스를 다시 시작할 수 있습니다.	미청구
shutting down	인스턴스가 종료할 준비를 하고 있습니다.	미청구
terminated	인스턴스가 영구적으로 삭제되었으며 시작할 수 없습니다.	미청구 Note 종료된 인스턴스에 적용되는 예약 인스턴스는 결제 옵션에 따라 기간이 종료될 때까지 요금이 청구됩니다. 자세한 내용은 예약 인스턴스 (p. 205) 단원을 참조하십시오.

Note

인스턴스를 재부팅해도 `running` 상태로 남아있기 때문에 새 인스턴스 청구 기간이 시작되지 않습니다.

인스턴스 시작

인스턴스를 시작하면 인스턴스가 `pending` 상태로 전환됩니다. 시작 시 지정한 인스턴스 유형에 따라 인스턴스에 사용되는 호스트 컴퓨터의 하드웨어가 결정됩니다. 인스턴스는 시작 시 사용자가 지정한 Amazon 머신 이미지(AMI)를 사용하여 부팅됩니다. 인스턴스 사용이 준비되고 나면 인스턴스가 `running` 상태로 전환됩니다. 실행 중인 인스턴스에 연결하여 바로 앞에 있는 컴퓨터를 사용하는 것처럼 인스턴스를 사용할 수 있습니다.

인스턴스가 `running` 상태로 전환되는 즉시 인스턴스 실행이 지속된 각 시간 또는 부분 시간에 대해 비용이 청구됩니다. 인스턴스가 유휴 상태이고 인스턴스에 연결하지 않더라도 마찬가지입니다.

자세한 내용은 [인스턴스 시작 \(p. 375\)](#) 및 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.

인스턴스 중지 및 시작(Amazon EBS 지원 인스턴스에만 해당)

인스턴스가 상태 확인을 통과하지 못하거나 애플리케이션이 예상대로 실행되고 있지 않은 경우 또는 인스턴스의 루트 볼륨이 Amazon EBS 볼륨인 경우 인스턴스를 중지했다가 시작하여 문제를 해결해 볼 수 있습니다.

인스턴스를 중지하면 `stopping` 상태로 전환되고 나서 `stopped` 상태로 전환됩니다. 인스턴스를 중지하고 나면 인스턴스에 대해 시간당 사용 요금이나 데이터 전송 요금이 부과되지는 않지만 모든 Amazon EBS 볼륨에 대한 스토리지 요금은 부과됩니다. 인스턴스가 `stopped` 상태인 경우 인스턴스 유형을 비롯하여 인스턴스의 특정 속성을 수정할 수 있습니다.

인스턴스를 시작하면 인스턴스가 `pending` 상태로 전환되며, 인스턴스가 새 호스트 컴퓨터로 이동됩니다(경우에 따라 현재 호스트에 남아 있음). 인스턴스를 중지했다가 시작하면 이전 호스트 컴퓨터의 인스턴스 스토어 볼륨에 있는 데이터가 모두 손실됩니다.

인스턴스에서 프라이빗 IPv4 주소가 유지됩니다. 즉, 프라이빗 IPv4 주소 또는 네트워크 인터페이스와 연결된 탄력적 IP 주소가 여전히 인스턴스와 연결되어 있다는 의미입니다. 인스턴스에 IPv6 주소가 있는 경우 해당 IPv6 주소를 유지합니다.

stopped에서 running으로 인스턴스를 전환할 때마다 1시간의 인스턴스 시간 전체에 요금을 청구하며, 여러 번의 전환이 한 시간 안에 이루어지더라도 마찬가지입니다.

자세한 내용은 [인스턴스 중지 및 시작 \(p. 440\)](#) 단원을 참조하십시오.

인스턴스 최대 절전 모드(Amazon EBS 지원 인스턴스에만 해당)

인스턴스를 최대 절전 모드로 전환하면 운영 체제에 최대 절전 모드(suspend-to-disk)를 수행하도록 알립니다. 그러면 인스턴스 메모리(RAM)의 콘텐츠를 Amazon EBS 루트 볼륨에 저장합니다. 인스턴스의 Amazon EBS 루트 볼륨과 연결된 모든 Amazon EBS 데이터 볼륨을 유지합니다. 인스턴스를 시작하면 Amazon EBS 루트 볼륨이 이전 상태로 복원되고, RAM 콘텐츠가 다시 로드됩니다. 이전에 연결된 데이터 볼륨이 다시 연결되고, 인스턴스는 해당 인스턴스 ID를 유지합니다.

인스턴스를 최대 절전 모드로 전환하면 stopping 상태로 전환되고 나서 stopped 상태로 전환됩니다. 최대 절전 모드로 전환하지 않고 [인스턴스를 중지 \(p. 372\)](#)한 경우와 달리 최대 절전 모드 인스턴스가 stopped 상태이면 해당 인스턴스에 대해서는 시간당 사용 요금을 청구할 수 없지만 stopping 상태일 때 비용이 청구됩니다. 데이터 전송에 대해 사용 요금이 부과되지는 않지만 RAM 데이터에 대한 스토리지를 포함해 모든 Amazon EBS 볼륨에 대한 스토리지 요금은 부과됩니다.

최대 절전 모드의 인스턴스를 시작하면 인스턴스가 pending 상태로 전환되며, 인스턴스가 새 호스트 컴퓨터로 이동됩니다(경우에 따라 현재 호스트에 남아 있음).

인스턴스에서 프라이빗 IPv4 주소가 유지됩니다. 즉, 프라이빗 IPv4 주소 또는 네트워크 인터페이스와 연결된 탄력적 IP 주소가 여전히 인스턴스와 연결되어 있다는 의미입니다. 인스턴스에 IPv6 주소가 있는 경우 해당 IPv6 주소를 유지합니다.

자세한 내용은 [Windows 인스턴스를 최대 절전 모드로 실행 \(p. 443\)](#) 단원을 참조하십시오.

인스턴스 재부팅

Amazon EC2 콘솔, 명령줄 도구 및 Amazon EC2 API를 사용하여 인스턴스를 재부팅할 수 있습니다. Amazon EC2를 사용하여 인스턴스에서 운영 체제 재부팅 명령을 실행하는 대신 인스턴스를 재부팅하는 것이 좋습니다.

인스턴스 재부팅은 운영 체제 재부팅과 같습니다. 인스턴스가 동일한 호스트 컴퓨터에 남아 있고, 퍼블릭 DNS 이름, 프라이빗 IP 주소 및 인스턴스 스토어 볼륨의 모든 데이터가 유지됩니다. 일반적으로 재부팅을 완료하는 데 몇 분 정도 소요되지만, 재부팅 소요 시간은 인스턴스 구성에 따라 달라집니다.

인스턴스를 재부팅하면 인스턴스 청구 시간.

자세한 내용은 [인스턴스 재부팅 \(p. 452\)](#) 단원을 참조하십시오.

인스턴스 만료

AWS에서 인스턴스를 호스팅하는 기본 하드웨어의 복구 불가능한 장애가 검색되는 경우 인스턴스가 만료 대상으로 예약됩니다. 예약된 만료 날짜에 도달하면 인스턴스가 AWS에 의해 종지되거나 종료됩니다. 인스턴스 루트 디바이스가 Amazon EBS 볼륨인 경우 인스턴스가 종지되며 언제든지 이 인스턴스를 다시 시작할 수 있습니다. 인스턴스 루트 디바이스가 인스턴스 스토어 볼륨인 경우 인스턴스가 종료되어 다시 사용할 수 없습니다.

자세한 내용은 [인스턴스 만료 \(p. 452\)](#) 단원을 참조하십시오.

인스턴스 종료

더 이상 인스턴스가 필요하지 않다고 판단되면 인스턴스를 종료할 수 있습니다. 인스턴스 상태가 *shutting-down* 또는 *terminated*로 변경되는 즉시 해당 인스턴스에 대한 요금 부과가 중지됩니다.

종료 방지 기능을 사용하는 경우 콘솔, CLI 또는 API를 사용하여 인스턴스를 종료할 수 없습니다.

인스턴스는 종료한 후에도 잠시 동안 콘솔에 표시되며 그 이후 항목이 자동으로 삭제됩니다. 또한 CLI 및 API를 사용하여 종료된 인스턴스를 설명할 수도 있습니다. 리소스(예: 태그)는 종료된 인스턴스에서 점차 연결 해제되므로 잠시 후 종료된 인스턴스에서 더 이상 보이지 않을 수 있습니다. You can't connect to or recover a terminated instance.

각각의 Amazon EBS 기반 인스턴스는 *InstanceInitiatedShutdownBehavior* 속성을 지원하는데, 이러한 속성은 인스턴스 자체 내에서 종료를 시작할 때 인스턴스가 중지되는지, 종료되는지를 제어합니다. 기본 동작은 인스턴스를 중지하는 것입니다. 인스턴스가 실행 중이거나 중단된 상태에 있을 때 이 속성을 수정할 수 있습니다.

각각의 Amazon EBS 볼륨은 *DeleteOnTermination* 속성을 지원하는데, 이 속성은 연결된 인스턴스를 종료할 때 볼륨이 삭제되는지, 유지되는지를 제어합니다. 기본값은 루트 디바이스 볼륨을 삭제하고 다른 EBS 볼륨을 유지하는 것입니다.

자세한 내용은 [인스턴스 종료 \(p. 454\)](#) 단원을 참조하십시오.

재부팅, 중지, 최대 절전 모드 및 종료의 차이

다음 표에는 인스턴스 재부팅, 중지, 최대 절전 모드 및 종료의 주요 차이점이 요약되어 있습니다.

특성	재부팅	중지/시작(Amazon EBS 기반 인스턴스에만 해당)	최대 절전 모드(Amazon EBS 지원 인스턴스에만 해당)	Terminate
호스트 컴퓨터	인스턴스가 동일 호스트 컴퓨터에서 유지됩니다.	인스턴스가 새 호스트 컴퓨터로 이동됩니다 (경우에 따라 현재 호스트에 남아 있음).	인스턴스가 새 호스트 컴퓨터로 이동됩니다 (경우에 따라 현재 호스트에 남아 있음).	없음
프라이빗 및 퍼블릭 IPv4 주소	이러한 주소는 동일하게 유지됩니다.	인스턴스가 관련 프라이빗 IPv4 주소를 유지합니다. 중지/시작 중에 변경되지 않는 탄력적 IP 주소가 지정되지 않는 한, 인스턴스가 새 퍼블릭 IPv4 주소를 가져옵니다.	인스턴스가 관련 프라이빗 IPv4 주소를 유지합니다. 중지/시작 중에 변경되지 않는 탄력적 IP 주소가 지정되지 않는 한, 인스턴스가 새 퍼블릭 IPv4 주소를 가져옵니다.	없음
탄력적 IP 주소(IPv4)	탄력적 IP 주소가 인스턴스와 연결된 상태로 유지됩니다.	탄력적 IP 주소가 인스턴스와 연결된 상태로 유지됩니다.	탄력적 IP 주소가 인스턴스와 연결된 상태로 유지됩니다.	인스턴스로부터 탄력적 IP 주소 연결이 끊깁니다.
IPv6 주소	주소가 동일하게 유지됩니다.	인스턴스가 관련 IPv6 주소를 유지합니다.	인스턴스가 관련 IPv6 주소를 유지합니다.	없음
인스턴스 스토어 볼륨	데이터가 유지됩니다.	데이터가 지워집니다.	데이터가 지워집니다.	데이터가 지워집니다.
루트 디바이스 볼륨	볼륨이 유지됩니다.	볼륨이 유지됩니다.	볼륨이 유지됩니다.	볼륨이 기본적으로 삭제됩니다.

특성	재부팅	종지/시작(Amazon EBS 기반 인스턴스에만 해당)	최대 절전 모드(Amazon EBS 지원 인스턴스에만 해당)	Terminate
RAM(메모리의 콘텐츠)	RAM이 지워집니다.	RAM이 지워집니다.	RAM은 루트 볼륨의 파일에 저장됩니다.	RAM이 지워집니다.
결제	인스턴스 결제 시간이 변경되지 않습니다.	상태가 <code>stopping</code> 으로 변경되는 즉시 인스턴스에 대한 요금 발생이 중지됩니다. 인스턴스 상태가 <code>stopped</code> 에서 <code>running</code> 으로 전환될 때마다 새로운 인스턴스 결제 시간.	인스턴스가 <code>stopping</code> 상태이면 비용이 발생하지만 <code>stopped</code> 상태일 때는 비용이 발생하지 않습니다. 인스턴스 상태가 <code>stopped</code> 에서 <code>running</code> 으로 전환될 때마다 새로운 인스턴스 결제 시간.	상태가 <code>shutting-down</code> 으로 변경되는 즉시 인스턴스에 대한 요금 발생이 중지됩니다.

운영 체제 종료 명령을 실행하면 항상 인스턴스 스토어 기반 인스턴스가 종료됩니다. 운영 체제 종료 명령으로 Amazon EBS 기반 인스턴스를 중지할지, 종료할지를 제어할 수 있습니다. 자세한 내용은 [인스턴스가 개시하는 종료 동작 변경 \(p. 457\)](#) 단원을 참조하십시오.

인스턴스 시작

인스턴스는 AWS 클라우드의 가상 서버입니다. 인스턴스는 Amazon Machine Image(AMI)에서 시작됩니다. AMI는 운영 체제와 애플리케이션 서버, 그리고 인스턴스 사용을 위한 애플리케이션을 제공합니다.

AWS 가입 시 무상으로 Amazon EC2를 시작할 수 있는 [AWS 프리 티어](#)를 제공합니다. 프리 티어를 사용하여 12개월 동안 무료로 t2.micro 인스턴스를 시작하고 사용할 수 있습니다(t2.micro를 사용할 수 없는 리전에서는 프리 티어에서 t3.micro 인스턴스를 사용할 수 있음). 프리 티어 외의 인스턴스를 시작하는 경우에는 인스턴스에 대하여 표준 Amazon EC2 사용 요금이 청구됩니다. 자세한 내용은 [Amazon EC2 요금](#)을 참조하십시오.

다음 방법을 사용하여 인스턴스를 시작할 수 있습니다.

방법	설명서
[Amazon EC2 콘솔] 시작 인스턴스 마법사를 사용하여 시작 파라미터를 지정	인스턴스 시작 마법사를 사용하여 인스턴스 시작 (p. 376)
[Amazon EC2 콘솔] 시작 템플릿을 생성하고 이 시작 템플릿에서 인스턴스를 시작	시작 템플릿에서 인스턴스 시작 (p. 381)
[Amazon EC2 콘솔] 기존 인스턴스를 기본 템플릿으로 사용	기존 인스턴스의 파라미터를 사용하여 인스턴스 시작 (p. 397)
[Amazon EC2 콘솔] AWS Marketplace에서 구매한 AMI 사용	AWS Marketplace 인스턴스 시작 (p. 398)
[AWS CLI] 선택한 AMI 사용	AWS CLI를 통해 Amazon EC2를 사용하는 방법
[Windows PowerShell용 AWS 도구] 선택한 AMI 사용	Amazon EC2(Windows PowerShell용 AWS 도구 사용)
[AWS CLI] EC2 플랫폼을 사용하여 여러 EC2 인스턴스 유형, 가용 용역 및 온디맨드 인스턴스, 예약 인스턴스 및 스팟 인스턴스 구매 모델에 걸쳐 용량을 프로비저닝합니다.	EC2 집합을 사용하여 인스턴스 시작 (p. 400)

인스턴스를 시작할 때 다음 리소스 중 하나에 연결된 서브넷에서 인스턴스를 시작할 수 있습니다.

- 가용 영역 - 이 옵션이 기본값입니다.
- 로컬 영역 - 로컬 영역에서 인스턴스를 시작하려면 로컬 영역을 옵트인한 다음 로컬 영역에 서브넷을 만들어야 합니다. 자세한 내용은 [로컬 영역](#)을 참조하세요.
- Wavelength Zone - Wavelength Zone에서 인스턴스를 시작하려면 Wavelength Zone을 옵트인한 다음 Wavelength Zone에 서브넷을 만들어야 합니다. Wavelength Zone에서 인스턴스를 시작하는 방법에 대한 자세한 내용은 AWS Wavelength 개발자 안내서의 [AWS Wavelength 시작하기](#)를 참조하세요.
- Outpost - Outpost에서 인스턴스를 시작하려면 Outpost를 만들어야 합니다. Outpost를 만드는 방법에 대한 자세한 내용은 AWS Outposts 사용 설명서의 [AWS Outposts 시작하기](#)를 참조하십시오.

인스턴스 시작한 다음 인스턴스를 연결하여 사용할 수 있습니다. 인스턴스는 pending 상태로 시작됩니다. 인스턴스 부팅이 시작되면 인스턴스의 상태가 running로 변경됩니다. 인스턴스 연결이 가능해 질 때까지 약간의 시간이 걸릴 수 있습니다. 베어 메탈 인스턴스 유형을 시작하는 데 더 오래 걸릴 수 있습니다. 베어 메탈 인스턴스에 대한 자세한 내용은 [Nitro 시스템에 구축된 인스턴스 \(p. 118\)](#) 단원을 참조하십시오.

인스턴스에서 수신하는 퍼블릭 DNS 이름은 사용자가 인터넷 상에서 해당 인스턴스에 접속할 때 사용됩니다. 인스턴스에서 수신하는 프라이빗 DNS 이름은 동일한 네트워크(EC2-Classic 또는 EC2-VPC) 내 다른 인스턴스에서 해당 인스턴스에 접속할 때 사용됩니다. 인스턴스 연결에 대한 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.

인스턴스 작업을 완료한 후에는 반드시 인스턴스를 삭제하십시오. 자세한 내용은 [인스턴스 종료 \(p. 454\)](#) 단원을 참조하십시오.

인스턴스 시작 마법사를 사용하여 인스턴스 시작

인스턴스 시작 마법사를 사용하여 인스턴스를 시작할 수 있습니다. 인스턴스 시작 마법사는 인스턴스를 시작하는 데 필요한 모든 시작 파라미터를 지정합니다. 인스턴스 시작 마법사에서 기본값을 제공하는 경우 기본 값을 그대로 사용하거나 고유한 값을 지정할 수 있습니다. 인스턴스를 시작하려면 최소한 AMI와 키 페어를 선택해야 합니다.

인스턴스를 시작하기 전에 설정을 확인합니다. 자세한 내용은 [Amazon EC2 설정 \(p. 12\)](#) 단원을 참조하십시오.

Important

시작하는 인스턴스가 [AWS 프리 티어](#)에 해당되지 않는 경우, 유휴 상태를 포함해 인스턴스가 실행된 시간에 대하여 과금이 청구됩니다.

인스턴스를 시작하는 단계는 다음과 같습니다.

- [인스턴스 시작 개시 \(p. 376\)](#)
- [1단계: Amazon 머신 이미지\(AMI\) 선택 \(p. 377\)](#)
- [2단계: 인스턴스 유형 선택 \(p. 377\)](#)
- [3단계: 인스턴스 세부 정보 구성 \(p. 378\)](#)
- [4단계: 스토리지 추가 \(p. 380\)](#)
- [5단계: 태그 추가 \(p. 380\)](#)
- [6단계: 보안 그룹 구성 \(p. 381\)](#)
- [7단계: 인스턴스 시작 검토 및 키 페어 선택 \(p. 381\)](#)

인스턴스 시작 개시

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에는 현재 리전이 표시됩니다(예: 미국 동부(오하이오)). 요구에 맞는 인스턴스의 리전을 선택합니다. 일부 Amazon EC2 리소스는 리전 간에 공유될 수 있지만 그렇지 않은 리소스도 있으므로 잘 선택해야 합니다. 자세한 내용은 [리소스 위치 \(p. 1132\)](#) 단원을 참조하십시오.

3. Amazon EC2 콘솔 대시보드에서 인스턴스 시작을 선택합니다.

1단계: Amazon 머신 이미지(AMI) 선택

인스턴스를 시작할 때 구성을 선택해야 하며, 이것을 Amazon 머신 이미지(AMI)이라고 합니다. AMI는 새 인스턴스를 생성하는 데 필요한 정보를 담고 있습니다. 예를 들어, AMI에는 웹 서버 역할을 수행하는 데 필요한 소프트웨어가 포함될 수 있습니다(Windows, Apache, 사용자의 웹 사이트 등).

인스턴스를 시작할 때 목록에서 AMI를 선택하거나 AMI ID를 가리키는 시스템 관리자 파라미터를 선택할 수 있습니다. 자세한 내용은 [시스템 관리자 파라미터를 사용하여 AMI 찾기 \(p. 70\)](#) 단원을 참조하십시오.

Amazon 머신 이미지(AMI) 선택 페이지에서 두 가지 옵션 중 하나를 사용하여 AMI를 선택합니다. [AMI 목록 검색 \(p. 377\)](#) 또는 [시스템 관리자 파라미터로 검색 \(p. 377\)](#)을 사용할 수 있습니다.

AMI 목록 검색

1. 왼쪽 창에서 사용할 AMI의 유형을 선택합니다:

빠른 시작

빠른 시작을 도와주는 인기 AMI를 선별하여 보여줍니다. 프리 티어로 이용할 수 있는 AMI만 선택하려면 왼쪽 창에서 프리 티어만을 선택합니다. 이러한 AMI는 프리 티어 사용 가능으로 표시됩니다.

My AMIs

사용자가 소유한 프라이빗 AMI 또는 공유된 프라이빗 AMI입니다. 사용자와 공유되는 AMI를 보려면 왼쪽 창에서 나와 공유됨을 선택합니다.

AWS Marketplace

AMI를 비롯하여 AWS에서 실행되는 소프트웨어를 구입할 수 있는 온라인 상점입니다. AWS Marketplace에서 인스턴스를 시작하는 방법에 대한 자세한 내용은 [AWS Marketplace 인스턴스 시작 \(p. 398\)](#)을 참조하십시오.

Community AMIs

AWS 커뮤니티 멤버가 다른 사람의 사용을 허용하여 게시한 AMI입니다. 운영 체제에 따라 AMI 목록을 필터링하려면 운영 체제 아래의 확인란을 선택하십시오. 이 외에도 아키텍처나 루트 디바이스 타입에 따라 필터링할 수 있습니다.

2. 각 AMI의 지원 가상화 유형 목록을 확인합니다. `hvm` 또는 `paravirtual` 중 필요한 유형의 AMI를 확인하십시오. 예를 들어 일부 인스턴스 유형은 HVM이 필요합니다.
3. 용도에 적합한 AMI를 선택하고 선택 버튼을 선택합니다.

시스템 관리자 파라미터

1. 오른쪽 상단에 있는 Search by 시스템 관리자 parameter(Systems Manager 파라미터로 검색)을 선택합니다.
2. 시스템 관리자 parameter(Systems Manager 파라미터)에서 파라미터를 선택합니다. 해당 AMI ID가 Currently resolves to(현재 확인된 값) 옆에 나타납니다.
3. 검색을 선택합니다. AMI ID와 일치하는 AMI가 목록에 나타납니다.
4. 목록에서 해당 AMI를 선택하고 선택을 선택합니다.

2단계: 인스턴스 유형 선택

인스턴스 유형 선택 페이지에서 시작할 인스턴스의 하드웨어 구성 및 크기를 선택합니다. 대형 인스턴스는 CPU와 메모리가 더 높습니다. 자세한 정보는 [인스턴스 유형 \(p. 114\)](#) 단원을 참조하십시오.

프리 티어 자격을 유지하려면 t2.micro 인스턴스 유형(또는 t2.micro를 사용할 수 없는 리전에서는 t3.micro 인스턴스 유형)을 선택합니다. 자세한 내용은 [성능 순간 확장 가능 인스턴스 \(p. 129\)](#) 단원을 참조하십시오.

기본 설정에서 마법사는 현 세대의 인스턴스 유형을 표시하고 사용자가 선택한 AMI를 기반으로 하여 첫 번째로 사용 가능한 유형을 선택합니다. 필터 목록에서 모든 세대를 선택하면 이전 세대의 인스턴스 유형을 볼 수 있습니다.

Note

테스트 목적으로 인스턴스를 빠르게 설정하려는 경우, 검토 및 시작(Review and Launch)을 선택하여 기본 구성 설정을 적용하고 인스턴스를 시작합니다. 그렇지 않은 경우 다음: 인스턴스 세부 정보 구성(Next: Configure Instance Details)을 선택해 인스턴스를 세부 구성할 수 있습니다.

3단계: 인스턴스 세부 정보 구성

인스턴스 세부 정보 구성(Configure Instance Details) 페이지에서 필요에 맞게 다음 설정을 변경하고(모든 설정 항목을 확장 표시하려면 고급 세부 정보 클릭), 다음: 스토리지 추가(Next: Add Storage)를 선택합니다.

- 인스턴스 개수: 시작할 인스턴스의 수를 입력합니다.

Tip

인스턴스가 빨리 시작되도록 하려면 큰 요청을 여러 개의 작은 배치로 나눕니다. 예를 들어 인스턴스 500개에 대해 시작 요청을 한 개 생성하는 대신, 인스턴스 100개에 대해 한 개씩 총 5개의 시작 요청을 생성합니다.

- (선택 사항) 애플리케이션 수요를 처리할 인스턴스의 수를 올바르게 유지하는 데 도움을 주기 위해 Auto Scaling 그룹 시작을 선택해 시작 구성 및 Auto Scaling 그룹을 생성할 수 있습니다. Auto Scaling은 사양에 따라 그룹의 인스턴스 수를 조정합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 사용 설명서](#) 단원을 참조하십시오.

Note

Amazon EC2 Auto Scaling에서 상태가 비정상인 것으로 표시한 Auto Scaling 그룹 내 인스턴스는 자동으로 교체 예정되어 종료되고 다른 인스턴스가 시작되어 원래 인스턴스의 데이터가 손실됩니다. 사용자가 인스턴스를 중지 또는 재부팅하거나 다른 이벤트에서 인스턴스의 상태를 비정상으로 표시하는 경우 인스턴스가 비정상으로 표시됩니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [Auto Scaling 인스턴스에 대한 상태 점검](#)을 참조하세요.

- 구입 옵션: 스팟 인스턴스를 시작하려면 스팟 인스턴스 요청을 선택합니다. 이렇게 하여 이 페이지에 선택 사항을 추가하거나 제거합니다. 최대 가격을 설정하고, 필요에 따라 요청 유형과 중단 동작 및 요청 유효성을 업데이트합니다. 자세한 내용은 [스팟 인스턴스 요청 생성 \(p. 258\)](#) 단원을 참조하십시오.
- 네트워크 VPC를 선택하거나 새 VPC 생성을 선택하여 Amazon VPC 콘솔로 이동해 새 VPC를 생성합니다. 마친 후에 마법사로 돌아와 새로 고침을 선택하면 해당 VPC가 목록에 로딩됩니다.
- 서브넷: 가용 영역, 로컬 영역, Wavelength Zone 또는 Outpost와 연결된 서브넷에서 인스턴스를 시작할 수 있습니다.

가용 영역에서 인스턴스를 시작하려면 인스턴스를 시작할 서브넷을 선택하십시오. 기본 설정 없음을 선택하여 AWS에서 임의의 가용 영역 내 기본 서브넷을 선택할 수 있습니다. 새 서브넷을 생성하려면 새 서브넷 생성을 선택하여 Amazon VPC 콘솔로 이동합니다. 마친 후에 마법사로 돌아와 새로 고침을 선택하면 해당 서브넷이 목록에 로딩됩니다.

로컬 영역에서 인스턴스를 시작하려면 로컬 영역에서 만든 서브넷을 선택하십시오.

Outpost에서 인스턴스를 시작하려면 Outpost와 연결된 VPC의 서브넷을 선택하십시오.

- 퍼블릭 IP 자동 할당: 인스턴스의 퍼블릭 IPv4 주소 수신 여부를 지정합니다. 기본 설정 사용 시, 기본 서브넷을 사용하는 인스턴스는 퍼블릭 IPv4 주소를 수신하고 기본이 아닌 서브넷의 인스턴스는 수신하지 않습니다. 활성화 또는 비활성화를 선택하여 서브넷의 기본 설정을 재정의할 수 있습니다. 자세한 내용은 [퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름 \(p. 703\)](#) 단원을 참조하십시오.

- 자동 할당 IPv6 IP: 인스턴스가 서브넷 범위 내에서 IPv6 주소를 수신할지 지정합니다. 활성화 또는 비활성화를 선택하여 서브넷의 기본 설정을 재정의합니다. 이 옵션은 IPv6 CIDR 블록에 VPC와 서브넷을 연결한 경우에만 사용할 수 있습니다. 자세한 내용은 Amazon VPC 사용 설명서의 [VPC 및 서브넷](#)을 참조하십시오.
- 도메인 조인 디렉터리(Domain join directory): 시작 후 Windows 인스턴스가 조인된 AWS Directory Service 디렉터리(도메인)를 선택합니다. 도메인을 선택하는 경우, 필요한 권한이 있는 IAM 역할을 선택해야 합니다. 자세한 내용은 [Windows EC2 인스턴스를 원활하게 조인](#)을 참조하세요.
- 배치 그룹: 배치 그룹은 인스턴스의 배치 전략을 결정합니다. 기존의 배치 그룹을 선택하거나 새로 만들 수 있습니다. 이 옵션은 배치 그룹을 지원하는 인스턴스 유형을 선택한 경우에만 사용할 수 있습니다. 자세한 내용은 [배치 그룹 \(p. 761\)](#) 단원을 참조하십시오.
- 용량 예약: 인스턴스를 공유 용량, 임의 open 용량 예약, 특정 용량 예약 또는 용량 예약 그룹으로 시작할지 여부를 지정합니다. 자세한 정보는 [인스턴스를 기준 용량 예약으로 시작 \(p. 362\)](#) 단원을 참조하십시오.
- IAM 역할: 인스턴스와 연결할 AWS Identity and Access Management(IAM) 역할을 선택합니다. 자세한 내용은 [Amazon EC2의 IAM 역할 \(p. 890\)](#) 단원을 참조하십시오.
- CPU options(CPU 옵션): Specify CPU options(CPU 옵션 지정)를 선택해 시작 중 vCPU의 수를 사용자 지정할 수 있습니다. CPU 코어 수와 코어당 스레드 수를 설정합니다. 자세한 내용은 [CPU 옵션 최적화 \(p. 541\)](#) 단원을 참조하십시오.
- 종료 동작: 인스턴스 종료 시 적용할 인스턴스 상태(종지 또는 종료)를 선택합니다. 자세한 정보는 [인스턴스가 개시하는 종료 동작 변경 \(p. 457\)](#) 단원을 참조하십시오.
- 종지 - 최대 절전 모드 동작: 최대 절전 모드를 활성화하려면 이 확인란을 선택하십시오. 이 옵션은 인스턴스가 최대 절전 모드 필수 조건을 충족하는 경우에만 사용할 수 있습니다. 자세한 내용은 [Windows 인스턴스를 최대 절전 모드로 실행 \(p. 443\)](#) 단원을 참조하십시오.
- 종료 방지 기능 활성화: 실수로 인스턴스를 종료하는 일을 방지하려면 이 확인란을 선택합니다. 자세한 내용은 [종료 방지 기능 활성화 \(p. 456\)](#) 단원을 참조하십시오.
- 모니터링: 이 확인란을 선택하면 Amazon CloudWatch 사용을 통한 인스턴스 세부 모니터링 기능이 활성화됩니다. 추가 요금이 발생합니다. 자세한 정보는 [CloudWatch를 사용해 인스턴스 모니터링 \(p. 667\)](#) 단원을 참조하십시오.
- EBS 최적화 인스턴스: Amazon EBS 최적화 인스턴스는 최적화된 구성 스택을 사용하며 Amazon EBS I/O를 위한 추가 전용 용량을 제공합니다. 인스턴스 유형이 이 기능을 지원하는 경우 이 확인란을 선택하여 활성화합니다. 추가 요금이 발생합니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.
- 테넌시: VPC로 인스턴스를 시작하는 경우 격리된 전용 하드웨어(전용) 또는 전용 호스트(전용 호스트)에서 인스턴스를 실행하도록 선택할 수 있습니다. 추가 요금이 적용될 수 있습니다. 자세한 내용은 [전용 인스턴스 \(p. 349\)](#) 및 [전용 호스트 \(p. 320\)](#) 단원을 참조하십시오.
- T2/T3 무제한: 애플리케이션이 필요한 시간만큼 기준 이상으로 버스트를 할 수 있도록 하려면 이 확인란을 선택합니다. 추가 요금이 적용될 수 있습니다. 자세한 내용은 [성능 순간 확장 가능 인스턴스 \(p. 129\)](#) 단원을 참조하십시오.
- 파일 시스템: 하나 이상의 Amazon EFS 파일 시스템을 인스턴스에 마운트하려면 파일 시스템 추가를 선택합니다. 자세한 내용은 [Amazon EC2와 함께 Amazon EFS 사용 \(p. 1104\)](#) 단원을 참조하십시오.
- 네트워크 인터페이스: 특정 서브넷을 선택한 경우, 인스턴스에 대해 네트워크 인터페이스를 최대 2개까지 지정할 수 있습니다.
 - 네트워크 인터페이스의 경우, AWS에서 새로운 인터페이스가 자동으로 생성되도록 새 네트워크 인터페이스를 선택하거나 사용 가능한 기준 네트워크 인터페이스를 선택합니다.
 - 기본 IP의 경우, 서브넷 범위에서 프라이빗 IPv4 주소를 입력하거나 AWS에서 프라이빗 IPv4 주소가 자동으로 선택되도록 자동 할당을 그대로 듭니다.
 - 보조 IP 주소에서 IP 추가를 선택하면 선택한 네트워크 인터페이스에 프라이빗 IPv4 주소를 두 개 이상 할당할 수 있습니다.
 - (IPv6 전용) IPv6 IP에서 IP 추가를 선택하고 서브넷 범위에서 IPv6 주소를 입력하거나 AWS가 자동으로 선택하도록 자동 할당을 그대로 듭니다.
 - 디바이스 추가를 선택하여 보조 네트워크 인터페이스를 추가합니다. 보조 네트워크 인터페이스는 인스턴스와 동일한 가용 영역에 있는 경우 VPC의 다른 서브넷에 상주할 수 있습니다.

자세한 내용은 [탄력적 네트워크 인터페이스 \(p. 729\)](#) 단원을 참조하십시오. 네트워크 인터페이스를 두 개 이상 지정하면 인스턴스가 퍼블릭 IPv4 주소를 수신할 수 없습니다. 또한 eth0에 대해 기존 네트워크 인터페이스를 지정하면 퍼블릭 IP 자동 할당을 사용하여 서브넷의 퍼블릭 IPv4 설정을 재정의할 수 없습니다. 자세한 내용은 [인스턴스 시작 시 퍼블릭 IPv4 주소 할당 \(p. 706\)](#) 단원을 참조하십시오.

- 커널 ID: (반가상화(PV) AMIs만 해당) 특정 커널을 사용하려는 경우가 아니라면 기본값 사용을 선택합니다.
- RAM 디스크 ID: (반가상화(PV) AMIs만 해당) 특정 RAM 디스크를 사용하려는 경우가 아니라면 기본값 사용을 선택합니다. 커널을 선택해 사용할 때는 해당 커널을 지원하는 드라이버가 설치된 RAM 디스크 지정이 필요할 수 있습니다.
- 액세스 가능한 메타데이터: 인스턴스 메타데이터에 대한 액세스를 활성화하거나 비활성화할 수 있습니다. 자세한 내용은 [인스턴스 메타데이터 서비스 구성 \(p. 577\)](#) 단원을 참조하십시오.
- 메타데이터 버전: 인스턴스 메타데이터에 대한 액세스를 활성화한 경우 인스턴스 메타데이터를 요청할 때 인스턴스 메타데이터 서비스 버전 2의 사용을 요구하도록 선택할 수 있습니다. 자세한 내용은 [새 인스턴스에 대한 인스턴스 메타데이터 옵션 구성 \(p. 580\)](#) 단원을 참조하십시오.
- 메타데이터 토큰 응답 흡수 제한: 인스턴스 메타데이터를 활성화하는 경우 메타데이터 토큰에 허용되는 네트워크 흡수 수를 설정할 수 있습니다. 자세한 내용은 [인스턴스 메타데이터 서비스 구성 \(p. 577\)](#) 단원을 참조하십시오.
- 사용자 데이터: 시작 과정에서 인스턴스를 구성하거나 구성 스크립트를 실행할 때 사용할 사용자 데이터를 지정할 수 있습니다. 파일을 첨부하려면 파일 옵션을 선택하여 첨부할 파일을 선택하십시오.

4단계: 스토리지 추가

선택한 AMI에는 루트 디바이스 볼륨을 포함한 하나 이상의 스토리지 볼륨이 있습니다. [스토리지 추가](#) 페이지에서 새 볼륨 추가를 선택하여 인스턴스에 연결할 추가 볼륨을 지정할 수 있습니다. 각 볼륨을 다음과 같이 구성한 후, 다음: 태그 추가를 선택합니다.

- 유형: 인스턴스에 연결할 인스턴스 스토어 또는 Amazon EBS 볼륨을 선택합니다. 목록에 표시되는 볼륨 유형은 선택한 인스턴스 유형에 따라 달라집니다. 자세한 내용은 [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#) 및 [Amazon EBS 볼륨 \(p. 929\)](#) 단원을 참조하십시오.
- 디바이스: 볼륨에서 사용할 디바이스 이름을 목록에서 선택합니다.
- 스냅샷: 볼륨 복원에 사용할 스냅샷의 이름이나 ID를 입력합니다. 또는 스냅샷 필드에 텍스트를 입력하여 사용 가능한 공유 및 퍼블릭 스냅샷을 검색할 수 있습니다. 스냅샷 정보는 대/소문자를 구분합니다.
- 크기: EBS 볼륨의 경우, 스토리지 크기를 지정할 수 있습니다. 선택한 AMI와 인스턴스가 프리 티어에 해당 되는 경우에도 프리 티어 한도를 유지하려면 총 스토리지 크기를 30GiB 미만으로 유지해야 합니다. 자세한 내용은 [EBS 볼륨의 크기 및 구성에 대한 제약 \(p. 945\)](#) 단원을 참조하십시오.
- 볼륨 유형: EBS 볼륨에 대한 볼륨 유형을 선택합니다. 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.
- IOPS: 프로비저닝된 IOPS SSD 볼륨 유형을 선택한 경우, 볼륨에서 지원되는 초당 I/O(IOPS) 수를 입력할 수 있습니다.
- 종료 시 삭제 여부: Amazon EBS 볼륨에 적용되는 기능으로, 확인란을 선택하면 인스턴스 종료 시 볼륨을 삭제합니다. 자세한 정보는 [인스턴스 종료 시 Amazon EBS 볼륨 보존 \(p. 458\)](#) 단원을 참조하십시오.
- 암호화: 인스턴스 유형이 EBS 암호화를 지원하는 경우, 볼륨의 암호화 상태를 지정할 수 있습니다. 이 리전에서 기본적으로 암호화를 활성화한 경우 사용자에 대해 기본 CMK가 선택됩니다. 다른 키를 선택하거나 암호화를 비활성화할 수 있습니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

5단계: 태그 추가

태그 추가 페이지에서 키와 값의 조합을 제공하여 [태그 \(p. 1143\)](#)를 지정합니다. 인스턴스 또는 볼륨 또는 이를 모두에 태그를 지정할 수 있습니다. 스팟 인스턴스의 경우 스팟 인스턴스 요청에만 태그를 지정할 수 있습니다. 리소스에 2개 이상의 태그를 추가하려면 다른 태그 추가를 선택합니다. 모두 마쳤으면 다음: 보안 그룹 구성(Next: Configure Security Group)을 선택합니다.

6단계: 보안 그룹 구성

보안 그룹 구성 페이지에서 기존 보안 그룹을 사용하여 인스턴스의 방화벽 규칙을 정의할 수 있습니다. 이 규칙은 인스턴스에 전달되는 수신 네트워크 트래픽을 정의합니다. 다른 모든 트래픽은 무시됩니다. (보안 그룹에 대한 자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#)을 참조하십시오.) 다음 과정에 따라 그룹을 선택하거나 새로 생성하고 검토 후 시작(Review and Launch)을 선택합니다.

- 기존 보안 그룹을 선택하려면 Select an existing security group(기존 보안 그룹 선택)을 선택하고 원하는 보안 그룹을 선택합니다. 기존의 보안 그룹 규칙은 수정할 수 없으며, 대신 새로 복사를 선택하여 새 보안 그룹으로 규칙을 복사할 수 있습니다. 다음 단계의 설명에 따라 규칙을 추가할 수 있습니다.
- 새 보안 그룹을 만들려면 새 보안 그룹 생성을 선택합니다. 마법사에서 자동으로 launch-wizard-x 보안 그룹을 정의하고, RDP(3389번 포트)를 통한 인스턴스 연결을 허용하는 인바운드 규칙을 생성합니다.
- 규칙은 필요에 따라 추가할 수 있습니다. 예를 들어 웹 서버인 인스턴스는 80번 포트(HTTP)와 443번 포트(HTTPS)를 개방해 인터넷 트래픽을 허용할 수 있습니다.

규칙을 추가하려면 규칙 추가를 선택한 다음 네트워크 트래픽의 개방 프로토콜을 선택하고 소스를 지정합니다. 소스 목록에서 내 IP를 선택하면 마법사에서 사용자 컴퓨터의 퍼블릭 IP 주소가 자동으로 추가됩니다. 하지만 고정 IP 주소 없이 방화벽 뒤에서 또는 ISP를 통해 연결하는 경우에는 클라이언트 컴퓨터가 사용하는 IP 주소의 범위를 찾아야 합니다.

Warning

모든 IP 주소(0.0.0.0/0)가 SSH 또는 RDP를 통해 인스턴스에 액세스할 수 있도록 허용하는 규칙은 이 짧은 예제에서만 사용하고, 프로덕션 환경에서는 위험하니 사용하지 마십시오. 특정 주소나 IP 주소 범위에서만 인스턴스 액세스를 허용하도록 설정해야 합니다.

7단계: 인스턴스 시작 검토 및 키 페어 선택

인스턴스 시작 검토 페이지에서 인스턴스 세부 정보를 확인한 다음, 해당되는 편집 링크를 선택하여 필요한 사항을 변경합니다.

준비가 완료되면 시작을 선택합니다.

기존 키 페어 선택 또는 새 키 페어 생성>Select an existing key pair or create a new key pair) 대화 상자에서 기존 키 페어를 선택하거나 새 키 페어를 만들 수 있습니다. 예를 들어, 기존 키 페어 선택을 선택하고 초기 설정에서 생성한 키 페어를 선택합니다. 자세한 내용은 [Amazon EC2 키 페어 및 Windows 인스턴스 \(p. 900\)](#) 단원을 참조하십시오.

Important

키 페어 없이 진행(Proceed without key pair) 옵션을 선택할 경우 사용자가 다른 방법으로 로그인할 수 있도록 구성된 AMI를 선택해야만 인스턴스에 연결할 수 있습니다.

인스턴스를 시작하려면 승인 확인란을 선택한 후 인스턴스 시작을 선택합니다.

(선택 사항) 인스턴스의 상태 확인 정보를 생성할 수 있습니다(추가 비용 적용 가능). (지금 결정하지 않아도 언제든지 나중에 추가할 수 있습니다.) 확인 화면에서 상태 검사 정보 생성을 선택하여 지침에 따릅니다. 자세한 내용은 [상태 확인 정보 생성 및 수정 \(p. 654\)](#) 단원을 참조하십시오.

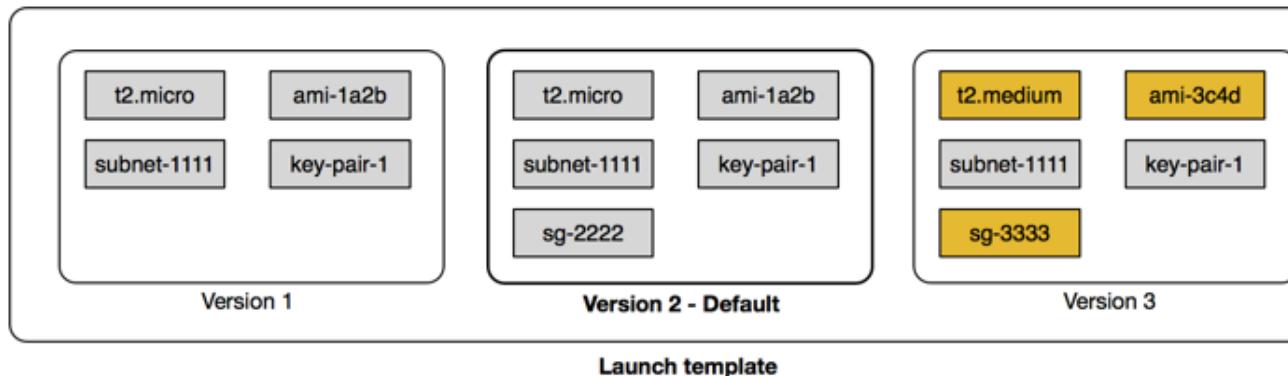
인스턴스가 시작하지 않거나 상태가 terminated이 아닌 running로 변경되는 경우, [인스턴스 시작 문제 해결 \(p. 1178\)](#) 단원을 참고하십시오.

시작 템플릿에서 인스턴스 시작

인스턴스 시작에 필요한 구성 정보가 포함된 시작 템플릿을 생성할 수 있습니다. 시작 템플릿은 인스턴스를 시작할 때마다 이를 지정할 필요가 없도록 시작 파라미터를 저장할 수 있게 해줍니다. 예를 들어 시작 템플릿에는 AMI ID, 인스턴스 유형, 인스턴스 시작에 일반적으로 사용되는 네트워크 설정이 포함될 수 있습니다. Amazon EC2 콘솔, AWS SDK 또는 명령줄 도구를 사용하여 인스턴스를 시작할 때 사용할 시작 템플릿을 지정할 수 있습니다.

각 시작 템플릿에서 번호가 지정된 시작 템플릿 버전을 하나 이상 생성할 수 있습니다. 버전마다 시작 파라미터가 다를 수 있습니다. 시작 템플릿에서 인스턴스를 시작할 때 어떤 시작 템플릿 버전이든 사용할 수 있습니다. 버전을 지정하지 않으면 기본 버전이 사용됩니다. 어떤 시작 템플릿 버전이든 기본 버전으로 설정이 가능하며, 기본적으로 시작 템플릿의 최초 버전이 설정되어 있습니다.

다음 그림은 세 가지 버전으로 시작 템플릿을 보여줍니다. 첫 번째 버전은 인스턴스를 시작하는 데 사용할 인스턴스 유형, AMI ID, 서브넷 및 키 페어를 지정합니다. 두 번째 버전은 첫 번째 버전을 토대로 하되, 인스턴스의 보안 그룹도 지정합니다. 세 번째 버전은 일부 파라미터에서 서로 다른 값을 사용합니다. 버전 2가 기본 버전으로 설정되어 있습니다. 이 시작 템플릿에서 인스턴스를 시작한 경우 다른 버전이 지정되어 있지 않으면 버전 2의 시작 파라미터가 사용됩니다.



목차

- [시작 템플릿 제한 \(p. 382\)](#)
- [시작 템플릿을 사용하여 시작 파라미터 제어 \(p. 383\)](#)
- [시작 템플릿 사용 제어 \(p. 383\)](#)
- [시작 템플릿 생성 \(p. 383\)](#)
- [시작 템플릿 버전 관리 \(p. 391\)](#)
- [시작 템플릿에서 인스턴스 시작 \(p. 394\)](#)
- [Amazon EC2 Auto Scaling에서 시작 템플릿 사용 \(p. 396\)](#)
- [EC2 집합에서 시작 템플릿 사용 \(p. 396\)](#)
- [스팟 집합에서 시작 템플릿 사용 \(p. 397\)](#)
- [시작 템플릿 삭제 \(p. 397\)](#)

시작 템플릿 제한

다음 규칙은 시작 템플릿과 시작 템플릿 버전에 적용됩니다.

- 리전당 5,000개의 시작 템플릿과 시작 템플릿당 10,000개의 버전으로 생성이 제한됩니다.
- 시작 템플릿 파라미터는 선택 사항입니다. 그러나 인스턴스 시작 요청에 필요한 모든 파라미터가 포함되도록 해야 합니다. 예를 들어 시작 템플릿에 AMI ID가 포함되어 있지 않으면 인스턴스를 시작할 때 시작 템플릿과 AMI ID를 모두 지정해야 합니다.
- 시작 템플릿을 생성할 때 시작 템플릿 파라미터가 완전히 확인되지 않습니다. 파라미터에 잘못된 값을 지정하거나 지원되는 파라미터 조합을 사용하지 않으면 이 시작 템플릿을 사용하여 인스턴스를 시작할 수 없습니다. 해당 파라미터에 대해 올바른 값을 지정하고 지원되는 파라미터 조합을 사용하는지 확인합니다. 예를 들어 배치 그룹에서 인스턴스를 시작하려면 지원되는 인스턴스 유형을 지정해야 합니다.
- 시작 템플릿에 태그를 지정할 수 있지만, 시작 템플릿 버전에는 태그 지정이 불가능합니다.
- 시작 템플릿 버전은 생성한 순서대로 번호가 지정됩니다. 시작 템플릿 버전을 생성할 때 버전 번호를 자체적으로 지정할 수 없습니다.

시작 템플릿을 사용하여 시작 파라미터 제어

시작 인스턴스에는 인스턴스 시작에 필요한 전체 또는 일부 파라미터가 포함될 수 있습니다. 시작 템플릿을 사용해 인스턴스를 실행할 때 시작 템플릿에 지정된 파라미터를 재정의할 수 있습니다. 또는 시작 템플릿에 없는 추가 파라미터를 지정할 수 있습니다.

Note

시작 작업 동안 시작 템플릿 파라미터를 제거할 수 없습니다(예를 들어 파라미터에 대해 null 값을 지정할 수 없음). 파라미터를 제거하려면 파라미터 없이 새로운 버전의 시작 템플릿을 생성하고 이 버전을 사용하여 인스턴스를 시작합니다.

인스턴스를 시작하려면 IAM 사용자는 `ec2:RunInstances` 작업을 사용할 권한이 있어야 합니다. 또한 해당 인스턴스로 생성하거나 해당 인스턴스와 연관된 리소스를 생성 또는 사용할 권한이 있어야 합니다. `ec2:RunInstances` 작업에 대한 리소스 수준 권한을 사용하여 사용자가 지정할 수 있는 시작 파라미터를 제어할 수 있습니다. 또는 사용자에게 시작 템플릿을 사용하여 인스턴스를 시작할 권한을 부여할 수 있습니다. 이렇게 하면 IAM 정책이 아닌 시작 템플릿에서 시작 파라미터를 관리하고 시작 템플릿을 사용하여 인스턴스 시작을 위한 권한 부여 방법으로 시작 템플릿을 사용할 수 있습니다. 예를 들어 사용자가 시작 템플릿을 사용하여 인스턴스를 시작만 할 수 있고 특정한 시작 템플릿만 사용하도록 지정할 수 있습니다. 또한 사용자가 시작 템플릿에서 재정의할 수 있는 시작 파라미터를 제어할 수도 있습니다. 예제 정책은 [시작 템플릿](#) (p. 867) 단원을 참조하십시오.

시작 템플릿 사용 제어

기본적으로 IAM 사용자에게는 시작 템플릿 사용 권한이 없습니다. 사용자에게 시작 템플릿과 시작 템플릿 버전을 생성, 수정, 설명 및 삭제할 수 있는 권한을 부여하는 IAM 사용자 정책을 생성할 수 있습니다. 일부 시작 템플릿 작업에 리소스 수준 권한을 적용하여 이러한 작업에서 특정 리소스를 사용할 수 있는 권한을 제어할 수도 있습니다. 자세한 내용은 [예제: 시작 템플릿 사용](#) (p. 878) 정책 예제를 참조하십시오.

`ec2:CreateLaunchTemplate` 및 `ec2:CreateLaunchTemplateVersion` 작업을 사용할 수 있는 권한을 부여할 때는 신중해야 합니다. 리소스 수준 권한을 사용하여 사용자가 시작 템플릿에서 지정할 수 있는 리소스를 제어할 수 없습니다. 인스턴스를 시작하는 데 사용되는 리소스를 제한하려면 시작 템플릿과 시작 템플릿 버전을 생성할 수 있는 권한을 해당 관리자에게만 부여해야 합니다.

시작 템플릿 생성

정의한 파라미터를 사용하여 새로운 시작 템플릿을 생성하거나 기존 시작 템플릿 또는 인스턴스를 새 시작 템플릿 생성을 위한 기준으로 사용합니다.

작업

- [정의한 파라미터를 사용하여 새 시작 템플릿 생성](#) (p. 383)
- [기존 시작 템플릿에서 시작 템플릿 생성](#) (p. 389)
- [인스턴스에서 시작 템플릿 생성](#) (p. 391)

정의한 파라미터를 사용하여 새 시작 템플릿 생성

새로운 콘솔

콘솔을 사용하여 정의된 파라미터를 사용하여 새 시작 템플릿을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택한 다음 시작 템플릿 생성을 선택합니다.
3. Launch template name에 대해 실행 템플릿의 설명이 포함된 이름을 입력하십시오.
4. 템플릿 버전 설명에서 시작 템플릿 버전에 대한 간단한 설명을 입력합니다.
5. 생성 시 시작 템플릿에 태그를 지정하려면 템플릿 태그를 확장하고 태그 추가를 선택한 다음 태그 키 및 값 페어를 입력합니다.

6. 시작 템플릿 내용에 다음 정보를 입력합니다.

- AMI: 인스턴스를 시작할 AMI입니다. 사용 가능한 모든 AMI를 검색하려면 AMI 검색을 선택합니다. 일반적으로 사용되는 AMI를 선택하려면 빠른 시작을 선택합니다. 또는 AWS Marketplace나 커뮤니티 AMI를 선택합니다. 소유하고 있는 AMI를 사용하거나 [적절한 AMI를 검색 \(p. 68\)](#)할 수 있습니다.
 - 인스턴스 유형: 인스턴스 유형이 지정한 AMI와 호환되는지 확인합니다. 자세한 내용은 [인스턴스 유형 \(p. 114\)](#) 단원을 참조하십시오.
 - 키 페어 이름: 인스턴스에 대한 키 페어입니다. 자세한 정보는 [Amazon EC2 키 페어 및 Windows 인스턴스 \(p. 900\)](#) 단원을 참조하십시오.
 - 네트워크 플랫폼: 해당하는 경우 VPC 또는 EC2-Classic으로 인스턴스를 시작할지 여부입니다. VPC를 선택한 경우에는 네트워크 인터페이스 섹션에서 서브넷을 지정합니다. Classic을 선택한 경우에는 EC2-Classic에서 지정된 인스턴스 유형이 지원되는지 확인하고 인스턴스에 대한 가용 영역을 지정합니다.
 - 보안 그룹: 인스턴스에 연결할 하나 이상의 보안 그룹입니다. 시작 템플릿에 네트워크 인터페이스를 추가하는 경우 이 설정을 생략하고 네트워크 인터페이스 사양의 일부로 보안 그룹을 지정합니다. 보안 그룹 및 네트워크 인터페이스를 지정하는 시작 템플릿에서는 인스턴스를 시작할 수 없습니다. 자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#) 단원을 참조하십시오.
7. 스토리지(볼륨)에서 AMI에서 지정된 볼륨 외에 인스턴스에 연결할 볼륨을 지정합니다(볼륨 1(AMI 루트)). 새 볼륨을 추가하려면 새 볼륨 추가를 선택합니다.
- 볼륨 유형: 인스턴스에 연결할 인스턴스 스토어나 Amazon EBS 볼륨입니다. 볼륨 유형은 선택한 인스턴스 유형에 따라 달라집니다. 자세한 내용은 [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#) 및 [Amazon EBS 볼륨 \(p. 929\)](#) 단원을 참조하십시오.
 - 디바이스 이름: 볼륨의 디바이스 이름입니다.
 - 스냅샷: 볼륨이 생성되는 스냅샷의 ID입니다.
 - 크기: Amazon EBS 볼륨의 경우 스토리지 크기입니다.
 - 볼륨 유형: Amazon EBS 볼륨에 대한 볼륨 유형을 선택합니다. 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.
 - IOPS: 프로비저닝된 IOPS SSD 볼륨 유형의 경우 해당 볼륨이 지원할 수 있는 초당 입력/출력 작업 수(IOPS)입니다.
 - 종료 시 삭제: Amazon EBS 볼륨의 경우 연결된 인스턴스가 종료되면 볼륨을 삭제할지 여부입니다. 자세한 정보는 [인스턴스 종료 시 Amazon EBS 볼륨 보존 \(p. 458\)](#) 단원을 참조하십시오.
 - 암호화: 인스턴스 유형이 EBS 암호화를 지원하는 경우, 볼륨의 암호화를 활성화할 수 있습니다. 이 리전에서 기본적으로 암호화를 활성화한 경우, 사용자에 대해 암호화가 활성화됩니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.
 - 키: EBS 암호화에 사용할 CMK입니다. AWS Key Management Service을 사용하여 생성한 고객 마스터 키(CMK)의 ARN을 지정할 수 있습니다. CMK를 지정하는 경우에도 암호화를 사용하여 암호화를 활성화해야 합니다.
8. 리소스 태그에서 키와 값의 조합을 제공하여 [태그 \(p. 1143\)](#)를 지정합니다. 인스턴스, 볼륨, 스팟 인스턴스 요청 또는 이 세 가지 모두에 태그를 지정할 수 있습니다.
9. 네트워크 인터페이스에서 인스턴스에 대한 [네트워크 인터페이스 \(p. 729\)](#)를 최대 두 개까지 지정할 수 있습니다.
- 디바이스 인덱스: 네트워크 인터페이스의 디바이스 번호입니다(예: 기본 네트워크 인터페이스의 경우 eth0). 이 필드를 비워두면 AWS가 기본 네트워크 인터페이스를 생성합니다.
 - 네트워크 인터페이스: 네트워크 인터페이스의 ID입니다. AWS가 새로운 네트워크 인터페이스를 생성하도록 하려면 비워둡니다.
 - 설명: (선택 사항) 새로운 네트워크 인터페이스의 설명입니다.
 - 서브넷: 새로운 네트워크 인터페이스를 생성할 서브넷입니다. 기본 네트워크 인터페이스(eth0)에서 이는 인스턴스가 시작되는 서브넷입니다. eth0에서 기존 네트워크 인터페이스를 입력한 경우에는 네트워크 인터페이스가 위치하는 서브넷에서 인스턴스가 시작됩니다.

- 자동 할당 퍼블릭 IP: 디바이스 인덱스가 eth0인 네트워크 인터페이스에 퍼블릭 IP 주소를 자동으로 할당할 것인지 여부입니다. 이 설정은 새로운 단일 네트워크 인터페이스에서만 활성화될 수 있습니다.
 - 기본 IP: 서브넷 범위 중 프라이빗 IPv4 주소입니다. AWS가 자동으로 프라이빗 IPv4 주소를 선택하도록 하려면 비워 둡니다.
 - 보조 IP: 서브넷 범위 중 보조 프라이빗 IPv4 주소입니다. AWS가 자동으로 선택하도록 하려면 비워 둡니다.
 - (IPv6에만 해당) IPv6 IPs: 서브넷 범위 중 IPv6 주소입니다.
 - 보안 그룹: 네트워크 인터페이스를 연결할 VPC의 하나 이상의 보안 그룹입니다.
 - 종료 시 삭제: 인스턴스가 삭제될 때 네트워크 인터페이스도 삭제되도록 할 것인지 여부입니다.
10. 고급 세부 정보에서 필드를 볼 수 있도록 섹션을 확장하고 인스턴스를 위한 추가 파라미터를 지정합니다.
- 구입 옵션: 구입 모델입니다. 온디맨드 가격으로 제한된 스팟 가격에서 스팟 인스턴스를 요청하려면 스팟 인스턴스 요청을 선택하고 기본 스팟 인스턴스 설정을 변경하려면 사용자 지정을 선택합니다. 스팟 인스턴스를 요청하지 않으면 EC2는 기본적으로 온디맨드 인스턴스를 시작합니다. 자세한 내용은 [스팟 인스턴스 \(p. 239\)](#) 단원을 참조하십시오.
 - IAM 인스턴스 프로파일: 인스턴스에 연결할 AWS Identity and Access Management(IAM) 인스턴스 프로파일입니다. 자세한 내용은 [Amazon EC2의 IAM 역할 \(p. 890\)](#) 단원을 참조하십시오.
 - 종료 방식: 종료 시 인스턴스를 중지 또는 종료해야 할지 여부입니다. 자세한 내용은 [인스턴스가 개시하는 종료 동작 변경 \(p. 457\)](#) 단원을 참조하십시오.
 - Stop - Hibernate behavior(중지 - 최대 절전 모드 동작): 인스턴스에 대해 최대 절전 모드가 활성화되어 있는지 여부입니다. 이 필드는 최대 절전 모드 사전 조건을 충족하는 인스턴스에만 유효합니다. 자세한 내용은 [Windows 인스턴스를 최대 절전 모드로 실행 \(p. 443\)](#) 단원을 참조하십시오.
 - 종료 방지: 실수로 인한 종료를 방지할 것인지 여부입니다. 자세한 정보는 [종료 방지 기능 활성화 \(p. 456\)](#) 단원을 참조하십시오.
 - 세부 CloudWatch 모니터링: Amazon CloudWatch를 사용하여 인스턴스에 대한 세부 모니터링 기능을 활성화할 것인지 여부입니다. 추가 요금이 발생합니다. 자세한 정보는 [CloudWatch를 사용해 인스턴스 모니터링 \(p. 667\)](#) 단원을 참조하십시오.
 - 탄력적 GPU: 인스턴스에 연결할 Elastic Graphics 액셀러레이터입니다. 일부 인스턴스 유형은 Elastic Graphics를 지원하지 않습니다. 자세한 내용은 [Amazon Elastic Graphics \(p. 636\)](#) 단원을 참조하십시오.
 - 탄력적 추론: EC2 CPU 인스턴스에 연결할 탄력적 추론 액셀러레이터입니다. 자세한 내용은 [Amazon 탄력적 추론 작업](#)을 참조하십시오.
 - T2/T3 무제한: 애플리케이션이 필요한 시간만큼 기준 이상으로 버스트를 할 수 있는지 여부입니다. 이 필드는 T2, T3 및 T3a 인스턴스에만 유효합니다. 추가 요금이 적용될 수 있습니다. 자세한 내용은 [성능 순간 확장 가능 인스턴스 \(p. 129\)](#) 단원을 참조하십시오.
 - 배치 그룹 이름: 인스턴스를 시작할 배치 그룹을 지정합니다. 하나의 배치 그룹에서 모든 인스턴스 유형을 시작할 수 있는 것은 아닙니다. 자세한 내용은 [배치 그룹 \(p. 761\)](#) 단원을 참조하십시오.
 - EBS 최적화 인스턴스: Amazon EBS I/O에 대한 추가, 전용 용량을 제공합니다. 모든 인스턴스 유형이 이 기능을 지원하는 것은 아니며, 추가 요금이 적용됩니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.
 - 용량 예약: 인스턴스를 공유 용량, 임의 open 용량 예약, 특정 용량 예약 또는 용량 예약 그룹으로 시작할지 여부를 지정합니다. 자세한 내용은 [인스턴스를 기준 용량 예약으로 시작 \(p. 362\)](#) 단원을 참조하십시오.
 - 테넌시: 인스턴스를 공유 하드웨어(공유), 격리된 전용 하드웨어(전용) 또는 전용 호스트(전용 호스트)에서 실행할지 선택합니다. 전용 호스트에서 인스턴스를 시작하도록 선택하면 인스턴스를 호스트 리소스 그룹에서 시작할지 여부를 지정하거나 특정 전용 호스트를 대상으로 지정할 수 있습니다. 추가 요금이 적용될 수 있습니다. 자세한 내용은 [전용 인스턴스 \(p. 349\)](#) 및 [전용 호스트 \(p. 320\)](#) 단원을 참조하십시오.

- RAM 디스크 ID: (반가상화(PV) AMIs에만 유효) 인스턴스의 RAM 디스크입니다. 커널을 지정해 사용할 때는 해당 커널을 지원하는 드라이버가 설치된 RAM 디스크 지정이 필요할 수 있습니다.
- 커널 ID: (반가상화(PV) AMIs에만 유효) 인스턴스의 커널입니다.
- 라이선스 구성: 지정된 라이선스 구성에 대해 인스턴스를 시작하여 라이선스 사용을 추적할 수 있습니다. 자세한 내용은 AWS License Manager 사용 설명서의 [라이선스 구성 생성](#)을 참조하십시오.
- 액세스 가능한 메타데이터: 인스턴스 메타데이터에 대한 액세스를 활성화할지 또는 비활성화할지 여부입니다. 자세한 내용은 [인스턴스 메타데이터 서비스 구성 \(p. 577\)](#) 단원을 참조하십시오.
- 메타데이터 버전: 인스턴스 메타데이터에 대한 액세스를 활성화한 경우 인스턴스 메타데이터를 요청할 때 인스턴스 메타데이터 서비스 버전 2의 사용을 요구하도록 선택할 수 있습니다. 자세한 내용은 [새 인스턴스에 대한 인스턴스 메타데이터 옵션 구성 \(p. 580\)](#) 단원을 참조하십시오.
- 메타데이터 응답 흡 제한: 인스턴스 메타데이터를 활성화하는 경우 메타데이터 토큰에 허용되는 네트워크 흡 수를 설정할 수 있습니다. 자세한 내용은 [인스턴스 메타데이터 서비스 구성 \(p. 577\)](#) 단원을 참조하십시오.
- 사용자 데이터: 시작 과정에서 인스턴스를 구성하거나 구성 스크립트를 실행할 때 사용할 사용자 데이터를 지정할 수 있습니다. 자세한 내용은 다음을 참조하십시오. [시작 시 Windows 인스턴스에서 명령 실행 \(p. 568\)](#).

11. 시작 템플릿 생성을 선택합니다.

기존 콘솔

콘솔을 사용하여 정의된 파라미터를 사용하여 새 시작 템플릿을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택한 다음 시작 템플릿 생성을 선택합니다.
3. Launch template name에 대해 실행 템플릿의 설명이 포함된 이름을 입력하십시오. 작성 시 실행 템플릿에 태그를 지정하려면 Show Tags, Add Tag를 선택한 다음 태그 키 및 값 페어를 입력하십시오.
4. 템플릿 버전 설명에서 시작 템플릿 버전에 대한 간단한 설명을 입력합니다.
5. 시작 템플릿 내용에 다음 정보를 입력합니다.
 - AMI ID: 인스턴스를 시작할 AMI를 지정합니다. 사용 가능한 모든 AMI를 검색하려면 AMI 검색을 선택합니다. 일반적으로 사용되는 AMI를 선택하려면 빠른 시작을 선택합니다. 또는 AWS Marketplace나 커뮤니티 AMI를 선택합니다. 소유하고 있는 AMI를 사용하거나 [적절한 AMI를 검색 \(p. 68\)](#)할 수 있습니다.
 - 인스턴스 유형: 인스턴스 유형이 지정한 AMI와 호환되는지 확인합니다. 자세한 내용은 [인스턴스 유형 \(p. 114\)](#) 단원을 참조하십시오.
 - 키 페어 이름: 인스턴스에 대한 키 페어입니다. 자세한 내용은 [Amazon EC2 키 페어 및 Windows 인스턴스 \(p. 900\)](#) 단원을 참조하십시오.
 - 네트워크 유형: 해당하는 경우 VPC 또는 EC2-Classic으로 인스턴스를 시작할 것인지 여부입니다. VPC를 선택한 경우에는 네트워크 인터페이스 섹션에서 서브넷을 지정합니다. Classic을 선택한 경우에는 EC2-Classic에서 지정된 인스턴스 유형이 지원되는지 확인하고 인스턴스에 대한 가용 영역을 지정합니다.
 - 보안 그룹: 인스턴스에 연결할 하나 이상의 보안 그룹입니다. 자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#) 단원을 참조하십시오.
6. 네트워크 인터페이스에서 인스턴스에 대한 [네트워크 인터페이스 \(p. 729\)](#)를 최대 두 개까지 지정할 수 있습니다.
 - 디바이스: 네트워크 인터페이스의 디바이스 번호입니다(예: 기본 네트워크 인터페이스의 경우 eth0). 이 필드를 비워두면 AWS가 기본 네트워크 인터페이스를 생성합니다.
 - 네트워크 인터페이스: 네트워크 인터페이스의 ID입니다. AWS가 새로운 네트워크 인터페이스를 생성하도록 하려면 비워둡니다.

- 설명: (선택 사항) 새로운 네트워크 인터페이스의 설명입니다.
 - 서브넷: 새로운 네트워크 인터페이스를 생성할 서브넷입니다. 기본 네트워크 인터페이스(eth0)에서 이는 인스턴스가 시작되는 서브넷입니다. eth0에서 기존 네트워크 인터페이스를 입력한 경우에는 네트워크 인터페이스가 위치하는 서브넷에서 인스턴스가 시작됩니다.
 - 자동 할당 퍼블릭 IP: 디바이스 인덱스가 eth0인 네트워크 인터페이스에 퍼블릭 IP 주소를 자동으로 할당할 것인지 여부입니다. 이 설정은 새로운 단일 네트워크 인터페이스에서만 활성화될 수 있습니다.
 - 기본 IP: 서브넷 범위 중 프라이빗 IPv4 주소입니다. AWS가 자동으로 프라이빗 IPv4 주소를 선택하도록 하려면 비워 둡니다.
 - 보조 IP: 서브넷 범위 중 보조 프라이빗 IPv4 주소입니다. AWS가 자동으로 선택하도록 하려면 비워 둡니다.
 - (IPv6에만 해당) IPv6 IPs: 서브넷 범위 중 IPv6 주소입니다.
 - 보안 그룹 ID: 네트워크 인터페이스를 연결할 VPC의 보안 그룹 ID입니다.
 - 종료 시 삭제: 인스턴스가 삭제될 때 네트워크 인터페이스도 삭제되도록 할 것인지 여부입니다.
7. 스토리지(볼륨)에서 AMI에서 지정된 볼륨 외에 인스턴스에 연결할 볼륨을 지정합니다.
- 볼륨 유형: 인스턴스에 연결할 인스턴스 스토어나 Amazon EBS 볼륨입니다. 볼륨 유형은 선택한 인스턴스 유형에 따라 달라집니다. 자세한 내용은 [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#) 및 [Amazon EBS 볼륨 \(p. 929\)](#) 단원을 참조하십시오.
 - 디바이스 이름: 볼륨의 디바이스 이름입니다.
 - 스냅샷: 볼륨이 생성되는 스냅샷의 ID입니다.
 - 크기: Amazon EBS 볼륨의 경우 스토리지 크기입니다.
 - 볼륨 유형: Amazon EBS 볼륨에 대한 볼륨 유형을 선택합니다. 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.
 - IOPS: 프로비저닝된 IOPS SSD 볼륨 유형의 경우 해당 볼륨이 지원할 수 있는 초당 입력/출력 작업 수(IOPS)입니다.
 - 종료 시 삭제: Amazon EBS 볼륨의 경우 연결된 인스턴스가 종료되면 볼륨을 삭제할지 여부입니다. 자세한 정보는 [인스턴스 종료 시 Amazon EBS 볼륨 보존 \(p. 458\)](#) 단원을 참조하십시오.
 - 암호화: 인스턴스 유형이 EBS 암호화를 지원하는 경우, 볼륨의 암호화를 활성화할 수 있습니다. 이 리전에서 기본적으로 암호화를 활성화한 경우, 사용자에 대해 암호화가 활성화됩니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.
 - 키: EBS 암호화에 사용할 CMK입니다. AWS Key Management Service을 사용하여 생성한 고객 마스터 키(CMK)의 ARN을 지정할 수 있습니다. CMK를 지정하는 경우에도 암호화를 사용하여 암호화를 활성화해야 합니다.
8. 인스턴스 태그에서 키와 값의 조합을 제공하여 [태그 \(p. 1143\)](#)를 지정합니다. 인스턴스 또는 볼륨 또는 이 둘 모두에 태그를 지정할 수 있습니다.
9. 고급 세부 정보에서 필드를 볼 수 있도록 섹션을 확장하고 인스턴스를 위한 추가 파라미터를 지정합니다.
- 구입 옵션: 구입 모델입니다. 온디맨드 가격으로 제한된 스팟 가격에서 스팟 인스턴스를 요청하려면 스팟 인스턴스 요청을 선택하고 스팟 인스턴스 설정을 변경하려면 스팟 파라미터 사용자 지정을 선택합니다. 스팟 인스턴스를 요청하지 않으면 EC2는 기본적으로 온디맨드 인스턴스를 시작합니다. 자세한 내용은 [스팟 인스턴스 \(p. 239\)](#) 단원을 참조하십시오.
 - IAM 인스턴스 프로파일: 인스턴스에 연결할 AWS Identity and Access Management(IAM) 인스턴스 프로파일입니다. 자세한 내용은 [Amazon EC2의 IAM 역할 \(p. 890\)](#) 단원을 참조하십시오.
 - 종료 방식: 종료 시 인스턴스를 종지 또는 종료해야 할지 여부입니다. 자세한 내용은 [인스턴스가 개시하는 종료 동작 변경 \(p. 457\)](#) 단원을 참조하십시오.
 - Stop - Hibernate behavior(종지 - 최대 절전 모드 동작): 인스턴스에 대해 최대 절전 모드가 활성화되어 있는지 여부입니다. 이 필드는 최대 절전 모드 사전 조건을 충족하는 인스턴스에만 유효합니다. 자세한 내용은 [Windows 인스턴스를 최대 절전 모드로 실행 \(p. 443\)](#) 단원을 참조하십시오.

- 종료 방지: 실수로 인한 종료를 방지할 것인지 여부입니다. 자세한 내용은 [종료 방지 기능 활성화 \(p. 456\)](#) 단원을 참조하십시오.
- 모니터링: Amazon CloudWatch를 사용하여 인스턴스에 대한 세부 모니터링 기능을 활성화할 것인지 여부입니다. 추가 요금이 발생합니다. 자세한 내용은 [CloudWatch를 사용해 인스턴스 모니터링 \(p. 667\)](#) 단원을 참조하십시오.
- Elastic Graphics: 인스턴스에 연결할 Elastic Graphics 액셀러레이터입니다. 일부 인스턴스 유형은 Elastic Graphics를 지원하지 않습니다. 자세한 내용은 [Amazon Elastic Graphics \(p. 636\)](#) 단원을 참조하십시오.
- T2/T3 무제한: 애플리케이션이 필요한 시간만큼 기준 이상으로 버스트를 할 수 있는지 여부입니다. 이 필드는 T2 및 T3 인스턴스에만 유효합니다. 추가 요금이 적용될 수 있습니다. 자세한 내용은 [성능 순간 확장 가능 인스턴스 \(p. 129\)](#) 단원을 참조하십시오.
- 배치 그룹 이름: 인스턴스를 시작할 배치 그룹을 지정합니다. 하나의 배치 그룹에서 모든 인스턴스 유형을 시작할 수 있는 것은 아닙니다. 자세한 내용은 [배치 그룹 \(p. 761\)](#) 단원을 참조하십시오.
- EBS 최적화 인스턴스: Amazon EBS I/O에 대한 추가, 전용 용량을 제공합니다. 모든 인스턴스 유형이 이 기능을 지원하는 것은 아니며, 추가 요금이 적용됩니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.
- 테넌시: 인스턴스를 공유 하드웨어(공유), 격리된 전용 하드웨어(전용) 또는 전용 호스트(전용 호스트)에서 실행할지 선택합니다. 전용 호스트에서 인스턴스를 시작하도록 선택하면 인스턴스를 호스트 리소스 그룹에서 시작할지 여부를 지정하거나 특정 전용 호스트를 대상으로 지정할 수 있습니다. 추가 요금이 적용될 수 있습니다. 자세한 내용은 [전용 인스턴스 \(p. 349\)](#) 및 [전용 호스트 \(p. 320\)](#) 단원을 참조하십시오.
- RAM 디스크 ID: 인스턴스를 위한 RAM 디스크입니다. 커널을 지정해 사용할 때는 해당 커널을 지원하는 드라이버가 설치된 RAM 디스크 지정이 필요할 수 있습니다. 반가상화(PV) AMIs에서만 유효합니다.
- 커널 ID: 인스턴스를 위한 커널입니다. 반가상화(PV) AMIs에서만 유효합니다.
- 사용자 데이터: 시작 과정에서 인스턴스를 구성하거나 구성 스크립트를 실행할 때 사용할 사용자 데이터를 지정할 수 있습니다. 자세한 내용은 다음을 참조하십시오. [시작 시 Windows 인스턴스에서 명령 실행 \(p. 568\)](#).

10. 시작 템플릿 생성을 선택합니다.

AWS CLI

AWS CLI를 사용하여 시작 템플릿을 생성하려면

- [create-launch-template](#) 명령을 사용합니다. 아래 예제에서는 다음을 지정하는 시작 템플릿을 생성합니다.
 - 시작 템플릿의 이름입니다(purpose=production)
 - 시작할 인스턴스 유형(r4.4xlarge) 및 AMI(ami-8c1be5f6)
 - 총 8개의 vCPU에 대한 코어 수(4) 및 코어당 스레드 수(2)(코어 4개 x 스레드 2개)
 - 인스턴스를 시작할 서브넷입니다(subnet-7b16de0c).

이 템플릿은 인스턴스에 퍼블릭 IP 주소와 IPv6 주소를 할당하고 인스턴스를 위한 태그를 생성합니다(Name=webserver).

```
aws ec2 create-launch-template \
--launch-template-name TemplateForWebServer \
--version-description WebVersion1 \
--tag-specifications 'ResourceType=launch-
template,Tags=[{Key=purpose,Value=production}]' \
--launch-template-data file://template-data.json
```

다음은 예제 template-data.json 파일입니다.

```
{  
    "NetworkInterfaces": [ {  
        "AssociatePublicIpAddress": true,  
        "DeviceIndex": 0,  
        "Ipv6AddressCount": 1,  
        "SubnetId": "subnet-7b16de0c"  
    } ],  
    "ImageId": "ami-8c1be5f6",  
    "InstanceType": "r4.4xlarge",  
    "TagSpecifications": [ {  
        "ResourceType": "instance",  
        "Tags": [ {  
            "Key": "Name",  
            "Value": "webserver"  
        } ]  
    } ],  
    "CpuOptions": {  
        "CoreCount": 4,  
        "ThreadsPerCore": 2  
    }  
}
```

다음은 예제 출력입니다.

```
{  
    "LaunchTemplate": {  
        "LatestVersionNumber": 1,  
        "LaunchTemplateId": "lt-01238c059e3466abc",  
        "LaunchTemplateName": "TemplateForWebServer",  
        "DefaultVersionNumber": 1,  
        "CreatedBy": "arn:aws:iam::123456789012:root",  
        "CreateTime": "2017-11-27T09:13:24.000Z"  
    }  
}
```

기존 시작 템플릿에서 시작 템플릿 생성

새로운 콘솔

콘솔을 사용하여 기존의 시작 템플릿에서 시작 템플릿을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택한 다음 시작 템플릿 생성을 선택합니다.
3. Launch template name에 대해 실행 템플릿의 설명이 포함된 이름을 입력하십시오.
4. 템플릿 버전 설명에서 시작 템플릿 버전에 대한 간단한 설명을 입력합니다.
5. 생성 시 시작 템플릿에 태그를 지정하려면 템플릿 태그를 확장하고 태그 추가를 선택한 다음 태그 키 및 값 페어를 입력합니다.
6. 소스 템플릿을 확장하고 시작 템플릿 이름에 대해 새 시작 템플릿의 기준으로 사용할 시작 템플릿을 선택합니다.
7. 소스 템플릿 버전에서 새로운 시작 템플릿의 토대가 되는 시작 템플릿 버전을 선택합니다.
8. 필요에 따라 시작 파라미터를 조정하고 시작 템플릿 생성을 선택합니다.

기존 콘솔

콘솔을 사용하여 기존의 시작 템플릿에서 시작 템플릿을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿 생성을 선택합니다. 시작 템플릿의 이름, 설명, 태그를 제공합니다.
4. 소스 템플릿에서 새로운 시작 템플릿의 토대가 되는 시작 템플릿을 선택합니다.
5. 소스 템플릿 버전에서 새로운 시작 템플릿의 토대가 되는 시작 템플릿 버전을 선택합니다.
6. 필요에 따라 시작 파라미터를 조정하고 시작 템플릿 생성을 선택합니다.

AWS CLI

AWS CLI를 사용하여 인스턴스에서 시작 템플릿 데이터를 가져오려면

- `get-launch-template-data` 명령을 사용하여 인스턴스 ID를 지정합니다. 출력을 새로운 시작 템플릿이나 시작 템플릿 버전을 생성하기 위한 기본 템플릿으로 사용할 수 있습니다. 기본적으로 출력에는 시작 템플릿 데이터에서 지정할 수 없는 최상위 `LaunchTemplateData` 객체가 포함되어 있습니다. 이 객체를 제외하려면 `--query` 옵션을 사용합니다.

```
aws ec2 get-launch-template-data \
--instance-id i-0123d646e8048babc \
--query "LaunchTemplateData"
```

다음은 예제 출력입니다.

```
{
    "Monitoring": {},
    "ImageId": "ami-8c1be5f6",
    "BlockDeviceMappings": [
        {
            "DeviceName": "/dev/xvda",
            "Ebs": {
                "DeleteOnTermination": true
            }
        }
    ],
    "EbsOptimized": false,
    "Placement": {
        "Tenancy": "default",
        "GroupName": "",
        "AvailabilityZone": "us-east-1a"
    },
    "InstanceType": "t2.micro",
    "NetworkInterfaces": [
        {
            "Description": "",
            "NetworkInterfaceId": "eni-35306abc",
            "PrivateIpAddresses": [
                {
                    "Primary": true,
                    "PrivateIpAddress": "10.0.0.72"
                }
            ],
            "SubnetId": "subnet-7b16de0c",
            "Groups": [
                "sg-7c227019"
            ],
            "Ipv6Addresses": [

```

```
{  
    "Ipv6Address": "2001:db8:1234:1a00::123"  
}  
]  
,"PrivateIpAddress": "10.0.0.72"  
}  
}
```

예를 들면 파일에 직접 출력을 기록할 수 있습니다.

```
aws ec2 get-launch-template-data \  
--instance-id i-0123d646e8048babc \  
--query "LaunchTemplateData" >> instance-data.json
```

인스턴스에서 시작 템플릿 생성

콘솔을 사용하여 인스턴스에서 시작 템플릿을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업 및 Create template from instance(인스턴스에서 템플릿 만들기)를 선택합니다.
4. 이름, 설명 및 태그를 입력하고 필요에 따라 시작 파라미터를 조정합니다.

Note

인스턴스에서 시작 템플릿을 생성할 때 인스턴스의 네트워크 인터페이스 ID 및 IP 주소는 이 템플릿에 포함되지 않습니다.

5. Create Template From Instance(인스턴스에서 템플릿 만들기)를 선택합니다.

시작 템플릿 버전 관리

특정 시작 템플릿에 대한 시작 템플릿 버전을 생성하고 기본 버전을 설정하며 더 이상 필요하지 않은 버전을 삭제할 수 있습니다.

작업

- [시작 템플릿 버전 생성 \(p. 391\)](#)
- [기본 시작 템플릿 버전 설정 \(p. 392\)](#)
- [시작 템플릿 버전 설명 \(p. 393\)](#)
- [시작 템플릿 버전 삭제 \(p. 394\)](#)

시작 템플릿 버전 생성

시작 템플릿 버전을 생성할 때 새로운 시작 파라미터를 지정하거나 기존 버전을 새 버전을 위한 기본 템플릿으로 사용할 수 있습니다. 시작 파라미터에 대한 자세한 정보는 [시작 템플릿 생성 \(p. 383\)](#) 단원을 참조하십시오.

새로운 콘솔

콘솔을 사용하여 시작 템플릿 버전을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿을 선택한 다음 작업, 템플릿 수정(새 버전 생성)을 선택합니다.
4. 템플릿 버전 설명에서 시작 템플릿 버전에 대한 설명을 입력합니다.
5. (선택 사항) 소스 템플릿을 확장하고 새 시작 템플릿 버전의 기반으로 사용할 시작 템플릿 버전을 선택합니다. 새 시작 템플릿 버전은 이 시작 템플릿 버전으로부터 시작 파라미터를 상속합니다.
6. 필요에 따라 시작 파라미터를 수정하고 시작 템플릿 생성을 선택합니다.

기존 콘솔

콘솔을 사용하여 시작 템플릿 버전을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿 생성을 선택합니다.
4. 어떤 작업을 하시겠습니까?에서 새 템플릿 버전 생성을 선택합니다.
5. 시작 템플릿 이름 목록에서 기존 시작 템플릿의 이름을 선택합니다.
6. 템플릿 버전 설명에서 시작 템플릿 버전에 대한 설명을 입력합니다.
7. (선택 사항) 해당하는 시작 템플릿의 버전 또는 새 시작 템플릿 버전의 토대로 사용할 다른 시작 템플릿의 버전을 선택합니다. 새 시작 템플릿 버전은 이 시작 템플릿 버전으로부터 시작 파라미터를 상속합니다.
8. 필요에 따라 시작 파라미터를 수정하고 시작 템플릿 생성을 선택합니다.

AWS CLI

AWS CLI를 사용하여 시작 템플릿 버전을 생성하려면

- `create-launch-template-version` 명령을 사용합니다. 새 버전의 토대가 될 소스 버전을 지정할 수 있습니다. 새 버전은 이 버전에서 시작 파라미터를 상속하며, `--launch-template-data`를 사용하여 파라미터를 재정의할 수 있습니다. 아래 예제에서는 시작 템플릿 버전 1을 토대로 새 버전을 생성하고 다른 AMI ID를 지정합니다.

```
aws ec2 create-launch-template-version \
--launch-template-id lt-0abcd290751193123 \
--version-description WebVersion2 \
--source-version 1 \
--launch-template-data "ImageId=ami-c998b6b2"
```

기본 시작 템플릿 버전 설정

시작 템플릿의 기본 버전을 설정할 수 있습니다. 시작 템플릿에서 인스턴스를 시작하고 버전을 지정하지 않으면 기본 버전의 파라미터를 사용하여 인스턴스가 시작됩니다.

새로운 콘솔

콘솔을 사용하여 기본 시작 템플릿 버전을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿을 선택하고 작업, 기본 버전 설정을 선택합니다.
4. 템플릿 버전의 경우 기본 버전으로 설정할 버전 번호를 선택하고 기본 버전으로 설정을 선택합니다.

기존 콘솔

콘솔을 사용하여 기본 시작 템플릿 버전을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿을 선택하고 작업, 기본 버전 설정을 선택합니다.
4. 기본 버전에서 버전 번호를 선택하고 기본 버전으로 설정하십시오를 선택합니다.

AWS CLI

AWS CLI를 사용하여 기본 시작 템플릿 버전을 설정하려면

- `modify-launch-template` 명령을 사용하여 기본으로 설정할 버전을 지정합니다.

```
aws ec2 modify-launch-template \
--launch-template-id lt-0abcd290751193123 \
--default-version 2
```

시작 템플릿 버전 설명

콘솔을 사용하여 선택한 시작 템플릿의 모든 버전을 보거나 최신 또는 기본 버전이 특정 버전 번호와 일치하는 시작 템플릿 목록을 가져올 수 있습니다. AWS CLI를 사용하여 지정된 시작 템플릿의 모든 버전, 개별 버전 또는 버전 범위를 설명할 수 있습니다. 계정에 있는 모든 시작 템플릿의 모든 최신 버전 또는 모든 기본 버전을 설명할 수도 있습니다.

새로운 콘솔

콘솔을 사용하여 시작 템플릿 버전을 설명하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 특정 시작 템플릿의 버전을 보거나 최신 또는 기본 버전이 특정 버전 번호와 일치하는 시작 템플릿 목록을 가져올 수 있습니다.
 - 시작 템플릿의 버전을 보려면: 시작 템플릿을 선택합니다. 버전 탭의 버전에서 세부 정보를 볼 버전을 선택합니다.
 - 최신 버전이 특정 버전 번호와 일치하는 모든 시작 템플릿 목록을 가져오려면: 검색 창에서 최신 버전을 선택한 다음 버전 번호를 선택합니다.
 - 기본 버전이 특정 버전 번호와 일치하는 모든 시작 템플릿 목록을 가져오려면: 검색 창에서 기본 버전을 선택한 다음 버전 번호를 선택합니다.

AWS CLI

AWS CLI를 사용하여 시작 템플릿 버전을 설명하려면

- `describe-launch-template-versions` 명령을 사용하고 버전 번호를 지정합니다. 다음 예에서는 버전 1과 3이 지정됩니다.

```
aws ec2 describe-launch-template-versions \
--launch-template-id lt-0abcd290751193123 \
--versions 1 3
```

AWS CLI를 사용하여 계정의 모든 최신 및 기본 시작 템플릿 버전을 설명하려면

- [describe-launch-template-versions](#) 명령을 사용하고 `$Latest`, `$Default` 또는 둘 모두를 지정합니다. 호출에서 시작 템플릿 ID와 이름을 생략해야 합니다. 버전 번호는 지정할 수 없습니다.

```
aws ec2 describe-launch-template-versions \
--versions "$Latest,$Default"
```

시작 템플릿 버전 삭제

시작 템플릿 버전이 더 이상 필요하지 않으면 이를 삭제할 수 있습니다. 삭제한 후에는 버전 번호를 바꿀 수 없습니다. 시작 템플릿의 기본 버전은 삭제가 불가능하며, 기본으로 다른 버전을 먼저 할당해야 합니다.

새로운 콘솔

콘솔을 사용하여 시작 템플릿 버전을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿을 선택하고 작업, 템플릿 버전 삭제를 선택합니다.
4. 삭제할 버전을 선택하고 삭제를 선택합니다.

기존 콘솔

콘솔을 사용하여 시작 템플릿 버전을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿을 선택하고 작업, 템플릿 버전 삭제를 선택합니다.
4. 삭제할 버전을 선택하고 시작 템플릿 버전 삭제를 선택합니다.

AWS CLI

AWS CLI를 사용하여 시작 템플릿 버전을 삭제하려면

- [delete-launch-template-versions](#) 명령을 사용하여 삭제할 버전 번호를 지정합니다.

```
aws ec2 delete-launch-template-versions \
--launch-template-id lt-0abcd290751193123 \
--versions 1
```

시작 템플릿에서 인스턴스 시작

시작 템플릿에 포함된 파라미터를 사용하여 인스턴스를 시작할 수 있습니다. 인스턴스를 시작하기 전에 시작 파라미터를 재정의 또는 추가하는 옵션이 제공됩니다.

시작 템플릿을 사용해 시작되는 인스턴스에는 `aws:ec2launchtemplate:id` 및 `aws:ec2launchtemplate:version` 등 두 개의 키를 통해 두 개의 태그가 자동 할당됩니다. 이러한 태그는 제거 또는 편집이 불가능합니다.

새로운 콘솔

콘솔을 사용하여 시작 템플릿에서 인스턴스를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿을 선택하고 작업, 템플릿에서 인스턴스 시작을 선택합니다.
4. 소스 템플릿 버전의 경우 사용할 시작 템플릿 버전을 선택합니다.
5. 인스턴스 수에 대해 시작할 인스턴스 수를 지정합니다.
6. (선택 사항) 인스턴스 세부 정보 섹션에서 파라미터를 변경 및 추가하여 시작 템플릿 파라미터를 재정의하거나 추가할 수 있습니다.
7. 템플릿으로 인스턴스 시작을 선택합니다.

기존 콘솔

콘솔을 사용하여 시작 템플릿에서 인스턴스를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿을 선택하고 작업, 템플릿에서 인스턴스 시작을 선택합니다.
4. 사용할 시작 템플릿 버전을 선택합니다.
5. (선택 사항) 인스턴스 세부 정보 섹션에서 파라미터를 변경 및 추가하여 시작 템플릿 파라미터를 재정의하거나 추가할 수 있습니다.
6. 템플릿으로 인스턴스 시작을 선택합니다.

AWS CLI

AWS CLI를 사용하여 시작 템플릿에서 인스턴스를 시작하려면

- `run-instances` 명령을 사용하여 `--launch-template` 파라미터를 지정합니다. 선택에 따라 사용할 시작 템플릿 버전을 지정합니다. 버전을 지정하지 않으면 기본 버전이 사용됩니다.

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123,Version=1
```

- 시작 템플릿 파라미터를 재정의하려면 `run-instances` 명령에서 파라미터를 지정합니다. 아래 예제는 시작 템플릿(존재하는 경우)에 지정된 인스턴스 유형을 재정의합니다.

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123 \
    --instance-type t2.small
```

- 복합 구조의 일부인 중첩 파라미터를 지정하면 시작 템플릿에 지정된 복합 구조를 비롯하여 지정된 추가 중첩 파라미터를 사용하여 인스턴스가 시작됩니다.

아래 예제에서는 `Owner=TeamA` 태그를 비롯해 시작 템플릿에 지정된 기타 태그를 통해 인스턴스가 시작됩니다. 시작 템플릿이 `Owner` 키와 함께 기존 태그를 가지고 있는 경우, 이 값이 `TeamA`로 바뀝니다.

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123 \
    --tag-specifications "ResourceType=instance,Tags=[{Key=Owner,Value=TeamA}]"
```

아래 예제에서는 디바이스 이름 /dev/xvdb를 비롯해 시작 템플릿에 지정된 기타 블록 디바이스 맵핑을 통해 볼륨에서 인스턴스가 시작됩니다. 시작 템플릿이 /dev/xvdb에 정의된 기존 볼륨을 가지고 있는 경우, 이 값이 지정된 값으로 바뀝니다.

```
aws ec2 run-instances \
--launch-template LaunchTemplateId=lt-0abcd290751193123 \
--block-device-mappings "DeviceName=/dev/xvdb,Ebs={VolumeSize=20,VolumeType=gp2}"
```

인스턴스가 시작하지 않거나 상태가 terminated이 아닌 running로 변경되는 경우, [인스턴스 시작 문제 해결 \(p. 1178\)](#) 단원을 참고하십시오.

Amazon EC2 Auto Scaling에서 시작 템플릿 사용

Auto Scaling 그룹을 생성하고 그룹에 사용할 시작 템플릿을 지정할 수 있습니다. Amazon EC2 Auto Scaling은 Auto Scaling 그룹에서 인스턴스를 시작할 때 연결된 시작 템플릿에 정의된 시작 파라미터를 사용합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [시작 템플릿을 사용한 Auto Scaling 그룹 생성](#) 을 참조하십시오.

시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하려면 먼저 AMI의 ID 등 Auto Scaling 그룹의 인스턴스를 시작하는 데 필요한 파라미터를 포함하는 시작 템플릿을 생성해야 합니다. 새 콘솔에서는 Auto Scaling에서 사용할 수 있는 템플릿을 생성하는 데 도움이 되는 지침을 제공합니다.

콘솔을 사용하여 Auto Scaling에서 사용할 시작 템플릿을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택한 다음 시작 템플릿 생성을 선택합니다.
3. Launch template name에 대해 실행 템플릿의 설명이 포함된 이름을 입력하십시오.
4. 템플릿 버전 설명에서 시작 템플릿 버전에 대한 간단한 설명을 입력합니다.
5. Auto Scaling 지침에서 확인란을 선택하여 Amazon EC2에서 Auto Scaling과 함께 사용할 템플릿을 생성하는 데 도움이 되는 지침을 제공하도록 합니다.
6. 필요에 따라 시작 파라미터를 수정합니다. Auto Scaling 지침을 선택했기 때문에 일부 필드는 필수이고 일부 필드는 사용할 수 없습니다. 시작 템플릿을 생성할 때 고려해야 할 사항 및 Auto Scaling에 대한 시작 파라미터를 구성하는 방법에 대한 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [Auto Scaling 그룹에 대한 시작 템플릿 생성](#)을 참조하십시오.
7. 시작 템플릿 생성을 선택합니다.
8. (선택 사항) 이 시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하려면 다음 단계 페이지에서 Auto Scaling 그룹 생성을 선택합니다.

AWS CLI를 사용하여 시작 템플릿으로 Amazon EC2 Auto Scaling 그룹을 생성 또는 업데이트하려면

- `create-auto-scaling-group` 또는 `update-auto-scaling-group` 명령을 사용하여 `--launch-template` 파라미터를 지정합니다.

EC2 집합에서 시작 템플릿 사용

인스턴스 구성에서 EC2 집합 요청을 생성하고 시작 템플릿을 지정할 수 있습니다. Amazon EC2는 EC2 집합 요청을 이행할 때 연결된 시작 템플릿에 정의된 시작 파라미터를 사용합니다. 시작 템플릿에 지정된 일부 파라미터는 재정의가 가능합니다.

자세한 내용은 [EC2 집합 생성 \(p. 418\)](#) 단원을 참조하십시오.

AWS CLI를 사용하여 시작 템플릿으로 EC2 집합을 생성하려면

- [create-fleet](#) 명령을 사용합니다. --launch-template-configs 파라미터를 사용하여 시작 템플릿과 시작 템플릿에 대한 모든 재구성을 지정합니다.

스팟 집합에서 시작 템플릿 사용

인스턴스 구성에서 스팟 집합 요청을 생성하고 시작 템플릿을 지정할 수 있습니다. Amazon EC2는 스팟 집합 요청을 이행할 때 연결된 시작 템플릿에 정의된 시작 파라미터를 사용합니다. 시작 템플릿에 지정된 일부 파라미터는 재정의가 가능합니다.

자세한 내용은 [스팟 집합요청 건 \(p. 271\)](#) 단원을 참조하십시오.

AWS CLI를 사용하여 시작 템플릿으로 스팟 집합 요청을 생성하려면

- [request-spot-fleet](#) 명령을 사용합니다. LaunchTemplateConfigs 파라미터를 사용하여 시작 템플릿과 시작 템플릿에 대한 모든 재구성을 지정합니다.

시작 템플릿 삭제

시작 템플릿이 더 이상 필요하지 않으면 이를 삭제할 수 있습니다. 시작 템플릿을 삭제하면 모든 버전이 삭제됩니다.

시작 템플릿을 삭제하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 시작 템플릿을 선택하고 작업, 템플릿 삭제를 선택합니다.
4. 시작 템플릿 삭제를 선택합니다.

시작 템플릿을 삭제하려면(AWS CLI)

- [delete-launch-template](#)(AWS CLI) 명령을 사용하여 시작 템플릿을 지정합니다.

```
aws ec2 delete-launch-template --launch-template-id lt-01238c059e3466abc
```

기존 인스턴스의 파라미터를 사용하여 인스턴스 시작

Amazon EC2 콘솔은 기존 인스턴스를 기반으로 시작 마법사 옵션을 제공하여 현재 인스턴스를 기본 템플릿으로 사용하여 다른 인스턴스를 시작할 수 있도록 합니다. 이 옵션을 사용하면 Amazon EC2 시작 마법사에서 선택한 인스턴스의 세부적인 구성 정보가 자동으로 입력됩니다.

Note

기존 인스턴스를 기반으로 시작 마법사 옵션은 선택한 인스턴스를 복제하는 것이 아니라 일부 구성 정보만 복제합니다. 인스턴스의 사본을 만드려면 해당 인스턴스에서 AMI를 생성한 후 AMI에서 추가 인스턴스를 시작하십시오.

또는 인스턴스의 시작 파라미터를 저장하도록 [시작 템플릿 \(p. 381\)](#)을 생성합니다.

선택한 인스턴스에서 시작 마법사로 복제되는 구성 정보:

- AMI ID
- 인스턴스 유형

- 선택 인스턴스가 위치한 가용 영역 또는 VPC, 서브넷
- 퍼블릭 IPv4 주소. 선택한 인스턴스에 현재 할당된 퍼블릭 IPv4 주소가 있다면 이 인스턴스의 기본 퍼블릭 IPv4 주소 설정에 상관 없이 새 인스턴스에서도 퍼블릭 IPv4 주소를 수신합니다. 퍼블릭 IPv4 주소에 대한 자세한 내용은 [퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름 \(p. 703\)](#) 단원을 참조하십시오.
- 배치 그룹(해당되는 경우)
- 인스턴스에 연결된 IAM 규칙(해당되는 경우)
- 종료 동작 설정(종지 또는 종료)
- 종료 보호 설정(True 또는 False)
- CloudWatch 모니터링(활성화 또는 비활성화)
- Amazon EBS 최적화 설정(True/False 설정)
- 테넌시 설정(VPC에서 시작하는 경우, 공유 또는 전용)
- 커널 ID 및 RAM 디스크 ID(해당되는 경우)
- 사용자 데이터(지정된 경우)
- 인스턴스에 연결된 태그(해당되는 경우)
- 인스턴스에 연결된 보안 그룹
- 연결 정보. 선택한 인스턴스에 구성 파일이 연결된 경우, 새 인스턴스도 동일한 파일에 자동으로 연결됩니다. 해당 구성 파일에 조인된 도메인 구성이 포함되었다면 새 인스턴스가 동일한 도메인에 조인됩니다. 도메인 조인에 대한 자세한 내용은 [인스턴스\(Simple AD 및 Microsoft AD\) 시작](#)을 참조하십시오.

선택한 인스턴스에서 마법사로 복제되지 않고 마법사에서 자체 기본 설정을 적용하는 구성 정보:

- 네트워크 인터페이스 수(기본값은 기본 네트워크 인터페이스(eth0)인 네트워크 인터페이스 1개)
- Storage: AMI와 인스턴스 유형에 따라 기본 스토리지 구성이 결정됩니다.

현재 인스턴스를 템플릿으로 사용하는 방법

- 인스턴스 페이지에서 사용할 인스턴스를 선택합니다.
- 작업을 선택한 다음 기존 인스턴스를 기반으로 시작을 선택합니다.
- 인스턴스 시작 검토 페이지에서 시작 마법사가 열립니다. 인스턴스 세부 정보를 확인한 다음, 해당되는 편집 링크를 클릭해 필요한 사항을 변경할 수 있습니다.
준비되면 시작을 선택한 다음 키 페어를 선택하고 인스턴스를 시작합니다.
- 인스턴스가 시작하지 않거나 상태가 terminated이 아닌 running로 변경되는 경우, [인스턴스 시작 문제 해결 \(p. 1178\)](#) 단원을 참고하십시오.

AWS Marketplace 인스턴스 시작

AWS Marketplace 제품을 구독하고 Amazon EC2 시작 마법사를 사용하여 제품의 AMI에서 인스턴스를 시작할 수 있습니다. 유료 AMI에 대한 자세한 내용은 [유료 AMI \(p. 80\)](#) 단원을 참조하십시오. 시작한 이후에 구독을 취소하려면 먼저 해당 구독에서 실행 중인 모든 인스턴스를 종료해야 합니다. 자세한 내용은 [AWS Marketplace 구독 관리 \(p. 83\)](#) 단원을 참조하십시오.

시작 마법사를 사용하여 AWS Marketplace에서 인스턴스를 시작하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- Amazon EC2 대시보드에서 인스턴스 시작을 선택합니다.
- Amazon 머신 이미지(AMI) 선택 페이지에서 왼쪽에 있는 AWS Marketplace 범주를 선택합니다. 범주를 검색하거나 검색 기능을 사용하여 적합한 AMI를 찾습니다. 선택을 선택하여 제품을 선택합니다.
- 대화 상자에 선택한 제품에 대한 개요가 표시됩니다. 요금 정보와 공급업체에서 제공한 기타 정보를 조회할 수 있습니다. 준비가 되면 계속을 선택합니다.

Note

AMI를 사용하여 인스턴스를 시작하기 전에는 제품 사용 요금이 부과되지 않습니다. 마법사의 다음 페이지에서 인스턴스 유형을 선택하라는 메시지가 표시되므로 지원되는 각 인스턴스 유형의 요금을 기록해 둡니다. 제품에 추가 세금이 적용될 수도 있습니다.

5. 인스턴스 유형 선택 페이지에서 시작할 인스턴스의 하드웨어 구성 및 크기를 선택합니다. 완료되면 Next: Configure Instance Details(다음: 인스턴스 세부 정보 구성)을 선택합니다.
6. 마법사의 다음 페이지에서 인스턴스를 구성하고, 스토리지 및 태그를 추가할 수 있습니다. 구성 가능한 다른 옵션에 대한 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오. 보안 그룹 구성 페이지가 나타날 때까지 다음을 선택합니다.

이 마법사에서는 제품에 대한 공급업체의 사양에 따라 새 보안 그룹을 생성합니다. 보안 그룹은 Linux의 SSH(포트 22) 또는 Windows의 RDP(포트 3389)에 모든 IPv4 주소(0.0.0.0/0) 액세스를 허용하는 규칙을 포함할 수 있습니다. 특정 주소 또는 주소 범위에만 해당 포트를 통한 인스턴스 액세스를 허용하도록 규칙을 조정하는 것이 좋습니다.

준비가 되면 Review and Launch(검토 후 시작)를 선택합니다.

7. Review Instance Launch(인스턴스 시작 검토) 페이지에서 인스턴스를 시작할 AMI에 대한 세부 정보와 마법사에서 설정한 기타 구성 정보를 확인합니다. 준비되면 시작을 선택하여 키 페어를 선택하거나 생성하고 인스턴스를 시작합니다.
8. 구독한 제품에 따라 인스턴스를 시작하는 데 몇 분 또는 그 이상 걸릴 수 있습니다. 인스턴스를 시작하려면 먼저 제품을 구독해야 합니다. 신용 카드 정보에 문제가 있는 경우 계정 세부 정보를 업데이트하라는 메시지가 나타납니다. 시작 확인 페이지가 표시되면 인스턴스 보기 선택하여 인스턴스 페이지로 이동합니다.

Note

유류 상태를 포함해 인스턴스가 실행 중인 동안 구독 요금이 청구됩니다. 인스턴스를 중지하더라도 스토리지에 대해 요금이 부과될 수 있습니다.

9. 인스턴스가 running 상태일 때 인스턴스에 연결할 수 있습니다. 이렇게 하려면 목록에서 인스턴스를 선택하고 연결을 선택합니다. 대화 상자의 지침을 따릅니다. 인스턴스 연결에 대한 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.

Important

인스턴스에 로그인하는 데 특정 사용자 이름을 사용해야 할 수도 있으므로 공급업체의 사용 지침을 주의해서 확인하십시오. 구독 세부 정보 액세스에 대한 자세한 내용은 [AWS Marketplace 구독 관리 \(p. 83\)](#) 단원을 참조하십시오.

10. 인스턴스가 시작하지 않거나 상태가 terminated이 아닌 running로 변경되는 경우, [인스턴스 시작 문제 해결 \(p. 1178\)](#) 단원을 참고하십시오.

API 및 CLI를 사용하여 AWS Marketplace AMI 인스턴스 시작

API 또는 명령줄 도구를 사용하여 AWS Marketplace 제품에서 인스턴스를 시작하려면 먼저 제품을 구독해야 합니다. 다음 방법을 사용하여 제품의 AMI ID로 인스턴스를 시작할 수 있습니다.

방법	설명서
AWS CLI	<code>run-instances</code> 명령을 사용합니다. 자세한 정보는 인스턴스 시작 단원을 참조하십시오.
Windows PowerShell용 AWS 도구	<code>New-EC2Instance</code> 명령을 사용합니다. 자세한 내용은 Windows PowerShell을 사용하여 Amazon EC2 시작 을 참조하십시오.
Query API	<code>RunInstances</code> 요청을 사용합니다.

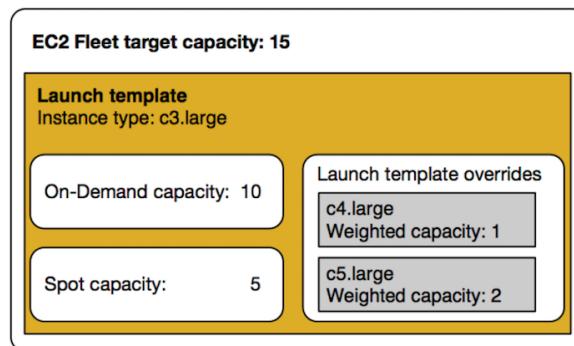
EC2 집합을 사용하여 인스턴스 시작

EC2 집합에는 인스턴스의 플릿 또는 그룹을 시작하기 위한 구성 정보가 있습니다. 한 번의 API 호출에서 플릿은 온디맨드 인스턴스, 예약 인스턴스 및 스팟 인스턴스 구매 옵션을 함께 사용하여 여러 가용 영역에서 여러 인스턴스 유형을 시작할 수 있습니다. EC2 집합을 사용하여 다음 작업을 수행할 수 있습니다.

- 별도의 온디맨드 및 스팟 용량 목표 및 시간당 지불하려는 최대 금액 정의
- 애플리케이션에 가장 적합하게 작동하는 인스턴스 유형 지정
- 각 구매 옵션 내에서 Amazon EC2가 플릿 용량을 배포해야 하는 방법 지정

또한 플릿에 대해 지불할 시간당 최대 금액을 설정할 수 있으며, EC2 집합은 최대 금액에 도달할 때까지 인스턴스를 실행합니다. 지불하려는 최대 금액에 도달하면 플릿은 목표 용량을 충족하지 않은 경우에도 인스턴스 실행을 중지합니다.

EC2 집합은 요청에 지정된 목표 용량을 충족하는 데 필요한 수만큼 인스턴스를 시작하려고 시도합니다. 시간당 총 최대 요금을 지정하면 지불하려는 최대 금액에 도달할 때까지 용량을 충족합니다. 스팟 인스턴스가 중단될 경우 플릿은 대상 스팟 용량을 유지하려고 시도할 수 있습니다. 자세한 정보는 [스팟 인스턴스의 작동 방식 \(p. 244\)](#) 단원을 참조하십시오.



EC2 집합마다 인스턴스 유형 수를 무제한 지정할 수 있습니다. 온디맨드 및 스팟 구매 옵션을 둘 다 사용하여 이 인스턴스 유형을 프로비저닝할 수 있습니다. 여러 가용 영역을 지정하고, 인스턴스마다 서로 다른 최대 스팟 가격을 지정하며, 플릿마다 추가 스팟 옵션을 선택할 수도 있습니다. Amazon EC2에서는 지정된 옵션을 사용하여 플릿이 시작될 때 용량을 프로비저닝합니다.

플릿이 실행되는 동안 Amazon EC2에서 가격 증가나 인스턴스 오류로 인해 스팟 인스턴스를 회수할 경우 EC2 집합이 해당 인스턴스를 사용자가 지정하는 인스턴스 유형으로 대체하려고 시도할 수 있습니다. 따라서 스팟 가격이 급증하는 동안 용량을 더 쉽게 다시 획득할 수 있습니다. 플릿마다 유연하고 탄력적인 리소싱 전략을 개발할 수 있습니다. 예를 들어 특정한 플릿에서 기본 용량은 저렴한 스팟 용량(사용 가능한 경우)으로 보충된 온디맨드일 수 있습니다.

예약 인스턴스가 있고 플릿에 온디맨드 인스턴스를 지정하면 EC2 집합에서는 예약 인스턴스를 사용합니다. 예를 들어 플릿에서 온디맨드 인스턴스를 c4.large로 지정하고 예약 인스턴스의 c4.large가 있으면 예약 인스턴스 요금을 수신합니다.

EC2 집합을 사용해도 추가 요금이 부과되지 않으며 플릿이 시작되는 EC2 인스턴스에 대해서만 비용을 지불합니다.

목차

- [EC2 집합 제한 사항 \(p. 401\)](#)
- [EC2 집합 제한 \(p. 401\)](#)
- [EC2 집합 구성 전략 \(p. 402\)](#)
- [EC2 집합 관리 \(p. 410\)](#)

EC2 집합 제한 사항

EC2 집합에는 다음과 같은 제한이 적용됩니다.

- API 또는 AWS CLI를 통해서만 EC2 집합을 사용할 수 있습니다.
- EC2 집합 요청으로 AWS 리전을 확장할 수 없습니다. 리전마다 따로 EC2 집합을 생성해야 합니다.
- EC2 집합 요청으로 동일한 가용 영역의 서로 다른 서브넷을 확장할 수 없습니다.

EC2 집합 제한

스팟 요청 가격 제한, 인스턴스 제한 및 볼륨 제한과 같이 EC2 집합에서 시작된 인스턴스에 일반적인 Amazon EC2 제한이 적용됩니다. 또한 다음과 같은 제한이 적용됩니다.

- AWS 리전당 활성 EC2 집합 수: 1,000 *†
- 스팟 인스턴스 풀 수(인스턴스 유형과 서브넷의 고유한 조합): 300* ‡
- 시작 사양의 사용자 데이터 크기: 16KB†
- EC2 집합당 대상 용량: 10,000
- 리전 내 모든 EC2 집합의 대상 용량: 100,000*
- EC2 집합 요청으로 리전을 확장할 수 없습니다.
- EC2 집합 요청으로 동일한 가용 영역의 서로 다른 서브넷을 확장할 수 없습니다.

대상 용량의 기본 한도 이상이 필요한 경우, AWS 지원 센터 [사례 생성](#) 서식을 작성하여 한도 증가를 요청하십시오. 제한 유형에서 EC2 Fleet(EC2 플릿)을 선택하고 리전을 선택한 다음 Target Fleet Capacity per Fleet (in units)(플릿당 대상 플릿 용량(유닛)) 또는 Target Fleet Capacity per Region (in units)(리전당 대상 플릿 용량(유닛))을 선택하거나 둘 다 선택합니다.

*이러한 제한은 사용자의 EC2 집합 및 스팟 집합 모두에 적용됩니다.

† 이 숫자는 하드 제한입니다. 이러한 제한에 대한 한도 상승을 요청할 수 없습니다.

‡ 이 제한은 `request` 또는 `maintain` 유형의 플릿에만 적용됩니다. 이 제한은 `instant` 플릿에는 적용되지 않습니다.

T3 스팟 인스턴스

T3 스팟 인스턴스를 CPU 크레딧 발생에 대한 유류 시간 없이 즉시 짧은 기간 동안 사용할 계획인 경우 T3 스팟 인스턴스를 [standard](#) (p. 141) 모드로 시작하여 높은 비용 지불을 방지하는 것이 좋습니다.

T3 스팟 인스턴스를 [unlimited](#) (p. 133) 모드를 시작하고 CPU를 즉시 버스트하는 경우 버스팅에 대한 잉여 크레딧을 소모하게 됩니다. 인스턴스를 짧은 기간 동안 사용하는 경우 인스턴스에서 잉여 크레딧을 지불할 정도의 CPU 크레딧이 발생할 시간이 없습니다. 인스턴스를 종료할 때 잉여 크레딧에 대한 요금이 청구됩니다.

T3 스팟 인스턴스가 [Unlimited](#) 모드인 경우는 버스팅에 대한 CPU 크레딧이 발생할 정도로 인스턴스 실행이 긴 경우에만 적합합니다. 그렇지 않은 경우 잉여 크레딧에 대한 비용을 지불하면 T3 스팟 인스턴스가 M5 또는 C5 인스턴스에 비해 비싸집니다. 자세한 내용은 [무제한 모드 대 고정 CPU 사용 시기](#) (p. 134) 단원을 참조하십시오.

T2 스팟 인스턴스

시작 크레딧은 효율적인 컴퓨팅 리소스를 제공하여 인스턴스를 구성함으로써 T2 인스턴스에 대한 생산적인 최초 시작 환경을 제공하는 것을 목적으로 합니다. 새 시작 크레딧에 액세스하기 위한 T2 인스턴스의 반복된 시작은 허용되지 않습니다. 지속적인 CPU가 필요한 경우 (일정 기간 동안 유류 상태로 둠으로써) 크레딧을 얻고 [T2 무제한](#) (p. 133)을 사용하거나 전용 CPU(예: `c4.large`)를 포함한 인스턴스 유형을 사용할 수 있습니다.

EC2 집합 구성 전략

EC2 집합은 온디맨드 인스턴스 및 스팟 인스턴스로 구성된 그룹입니다.

EC2 집합은 집합 요청에서 지정한 목표 용량을 충족하는 데 필요한 인스턴스 수만큼 시작하려고 시도합니다. 집합은 온디맨드 인스턴스 또는 스팟 인스턴스만으로 구성되거나 온디맨드 인스턴스 및 스팟 인스턴스의 조합으로 구성될 수 있습니다. 스팟 인스턴스에 대한 요청은 요청의 시간당 최대 가격이 현재 스팟 가격을 초과하고 사용 가능한 용량이 있으면 수행됩니다. 스팟 인스턴스가 중단될 경우 플릿은 대상 용량을 유지하려고 시도합니다.

또한 플릿에 대해 지불할 시간당 최대 금액을 설정할 수 있으며, EC2 집합은 최대 금액에 도달할 때까지 인스턴스를 실행합니다. 지불하려는 최대 금액에 도달하면 플릿은 목표 용량을 충족하지 않은 경우에도 인스턴스 실행을 중지합니다.

스팟 인스턴스 풀은 동일한 인스턴스 유형, 운영 체제, 가용 영역 및 네트워크 플랫폼을 가지는 미사용 EC2 인스턴스의 세트입니다. EC2 집합을 생성할 때 인스턴스 유형, 가용 영역, 서브넷 및 최고 가격에 따라 달라지는 여러 시작 사양을 포함할 수 있습니다. 플릿은 요청에 포함된 시작 사양과 요청의 구성을 기반으로 요청을 이행하는 데 사용되는 스팟 인스턴스 풀을 선택합니다. 스팟 인스턴스는 선택한 풀에서 가져옵니다.

EC2 집합을 사용하면 코어나 인스턴스 수 또는 메모리 양을 기반으로 애플리케이션에 맞는 대량의 EC2 용량을 프로비저닝할 수 있습니다. 예를 들어 EC2 집합이 인스턴스 200개의 목표 용량을 시작하도록 지정할 수 있습니다. 이 가운데 130개는 온디맨드 인스턴스이고 나머지는 스팟 인스턴스입니다.

적절한 구성 전략을 사용하여 요구에 맞는 EC2 집합을 생성하십시오.

내용

- [EC2 집합 계획 \(p. 402\)](#)
- [EC2 집합 요청 유형 \(p. 403\)](#)
- [스팟 인스턴스를 위한 할당 전략 \(p. 403\)](#)
- [온디맨드 백업을 위한 EC2 집합 구성 \(p. 404\)](#)
- [최고 가격 재정의 \(p. 405\)](#)
- [지출 제어 \(p. 405\)](#)
- [EC2 집합 인스턴스 가중치 부여 \(p. 406\)](#)
- [자습서: 인스턴스 가중치를 부여한 EC2 집합 사용 \(p. 407\)](#)
- [자습서: 기본 용량이 온디맨드인 EC2 집합 사용 \(p. 409\)](#)

EC2 집합 계획

EC2 집합을 계획할 때 다음을 수행하는 것이 좋습니다.

- 원하는 목표 용량의 동기 또는 비동기식 일회성 요청을 제출하는 EC2 집합을 생성할지 아니면 시간 경과에 따라 목표 용량을 유지하는 플릿을 생성할지 결정합니다. 자세한 내용은 [EC2 집합 요청 유형 \(p. 403\)](#) 단원을 참조하십시오.
- 인스턴스 유형을 결정하고 애플리케이션 요구를 만족합니다.
- 스팟 인스턴스를 EC2 집합에 포함하려면 플릿을 생성하기 전에 [스팟 모범 사례](#)를 살펴보십시오. 가능한 한 최저 가격으로 인스턴스를 프로비저닝할 수 있도록 플릿을 계획할 때 이 모범 사례를 사용하십시오.
- EC2 집합의 목표 용량을 결정합니다. 인스턴스 또는 사용자 지정 단위에서 목표 용량을 설정할 수 있습니다. 자세한 내용은 [EC2 집합 인스턴스 가중치 부여 \(p. 406\)](#) 단원을 참조하십시오.
- EC2 집합 목표 용량 중에서 온디맨드 용량 및 스팟 용량이어야 하는 부분을 결정합니다. 온디맨드 용량이나 스팟 용량 또는 둘 다 0을 지정할 수 있습니다.
- 인스턴스 가중치를 사용하는 경우에는 단위당 가격을 결정합니다. 단위당 가격을 계산하려면 인스턴스 시간당 가격을 이 인스턴스가 나타내는 단위 수(또는 가중치)로 나눕니다. 인스턴스 가중치를 사용하지 않는 경우 단위당 기본 가격은 인스턴스 시간당 가격입니다.

- 플릿에 대해 지불할 최대 시간당 금액을 결정하십시오. 자세한 내용은 [지출 제어 \(p. 405\)](#) 단원을 참조하십시오.
- EC2 집합의 가능한 옵션을 살펴봅니다. 자세한 내용은 [EC2 집합 JSON 구성 파일 참조 \(p. 416\)](#) 단원을 참조하십시오. EC2 집합 구성 예제는 [EC2 집합 구성의 예 \(p. 426\)](#) 단원을 참조하십시오.

EC2 집합 요청 유형

EC2 집합에는 다음 세 가지 유형의 요청이 있습니다.

instant

요청 유형을 `instant`로 구성하면 EC2 집합이 원하는 용량을 얻기 위한 동기식 일회성 요청을 합니다. API 응답에서 시작할 수 없는 인스턴스에 대한 오류와 함께 시작된 인스턴스를 반환합니다.

request

요청 유형을 `request`로 구성하면 EC2 집합이 원하는 용량을 얻기 위한 비동기식 일회성 요청을 합니다. 그 뒤에 스팟 종단으로 인해 용량이 감소할 경우 플릿은 스팟 인스턴스를 보충하려고 하지 않으며 용량을 사용할 수 없는 경우 대체 스팟 인스턴스에서 요청을 제출하지 않습니다.

maintain

(기본값) 요청 유형을 `maintain`으로 구성하면 EC2 집합은 원하는 용량을 얻기 위한 비동기식 요청을 하고 종단된 모든 스팟 인스턴스를 자동으로 보충해 용량을 유지합니다.

세 가지 유형의 요청 모두에 할당 전략이 유익합니다. 자세한 내용은 [스팟 인스턴스를 위한 할당 전략 \(p. 403\)](#) 단원을 참조하십시오.

스팟 인스턴스를 위한 할당 전략

EC2 집합의 할당 전략에 따라, 시작 사양으로 표시되는 가능한 스팟 인스턴스 풀에서 스팟 인스턴스에 대한 요청을 이행하는 방법이 결정됩니다. 플릿에 지정할 수 있는 할당 전략은 다음과 같습니다.

lowest-price

스팟 인스턴스는 최저 가격의 풀에서 가져옵니다. 이는 기본 전략입니다.

diversified

스팟 인스턴스는 모든 풀에 두루 분산됩니다.

capacity-optimized

스팟 인스턴스는 시작하는 인스턴스의 수에 대한 용량이 최적화된 풀에서 가져옵니다.

InstancePoolsToUseCount

스팟 인스턴스는 지정한 스팟 풀 수에 걸쳐 배포됩니다. 이 파라미터는 `lowest-price`와 함께 사용하는 경우에만 유효합니다.

목표 용량 유지

스팟 가격 또는 스팟 인스턴스 풀의 가용 용량 변화로 인해 스팟 인스턴스가 종료된 후에는 `maintain` 유형의 EC2 집합이 대체 스팟 인스턴스를 시작합니다. 할당 전략이 `lowest-price`인 경우, 플릿은 현재 스팟 가격이 가장 낮은 풀에서 대체 인스턴스를 시작합니다. 할당 전략으로써 `InstancePoolsToUseCount`와 함께 `lowest-price`를 사용하는 경우 플릿이 최저 가격의 스팟 풀을 선택하여 지정한 스팟 풀 수에 걸쳐 스팟 인스턴스를 시작합니다. 할당 전략이 `capacity-optimized`인 경우, 플릿은 가용성이 뛰어난 스팟 인스턴스 용량이 있는 풀에서 대체 인스턴스를 시작합니다. 할당 전략이 `diversified`인 경우 플릿은 나머지 풀에 대체 스팟 인스턴스를 배포합니다.

비용 최적화를 위한 EC2 집합 구성

스팟 인스턴스 사용 비용을 최적화하려면 EC2 집합이 현재 스팟 가격을 기반으로 인스턴스 유형과 가용 영역의 가장 저렴한 조합을 자동으로 배포하도록 `lowest-price` 할당 전략을 지정합니다.

EC2 집합은 항상 퍼블릭 온디맨드 가격을 기반으로 가장 저렴한 인스턴스 유형을 온디맨드 인스턴스 목표 용량에 대해 선택하며, 스팟 인스턴스에 대해서는 계속해서 할당 전략(`lowest-price`, `capacity-optimized` 또는 `diversified`)을 따릅니다.

비용 최적화 및 다각화를 위한 EC2 집합 구성

저렴함과 다각화를 모두 충족하는 스팟 인스턴스 플릿을 생성하려면 `InstancePoolsToUseCount`와 함께 `lowest-price` 할당 전략을 사용하십시오. EC2 집합이 지정된 스팟 풀 수에 걸쳐 현재 스팟 가격을 기반으로 인스턴스 유형과 가용 영역의 가장 저렴한 조합을 자동으로 배포합니다. 이 조합을 통해 가장 비싼 스팟 인스턴스를 피할 수 있습니다.

용량 최적화를 위한 EC2 집합 구성

스팟 인스턴스에서 요금은 시간이 지나면서 수요 및 공급의 장기 추세에 따라 서서히 변화하지만 용량은 실시간으로 변동합니다. `capacity-optimized` 전략은 실시간 용량 데이터를 기준으로 가장 가용성이 높은 풀을 예측하여 자동으로 스팟 인스턴스를 가장 가용성이 높은 풀로 시작합니다. 이 전략은 빅 데이터 및 분석, 이미지 및 미디어 렌더링, 기계 학습, 고성능 컴퓨팅과 같이 작업 재시작 및 체크포인트와 관련된 중단으로 인한 비용이 더 높을 수 있는 워크로드에 유용합니다. `capacity-optimized` 전략은 중단을 줄일 수 있는 가능성을 제공함으로써 전체 워크로드 비용을 낮출 수 있습니다.

적합한 할당 전략 선택

사용 사례를 바탕으로 플릿을 최적화할 수 있습니다.

플릿이 작거나 짧은 시간 동안 실행될 경우 모든 인스턴스가 단일 스팟 인스턴스 풀에 있더라도 스팟 인스턴스가 중단될 확률은 낮습니다. 따라서 `lowest-price` 전략이 요구를 충족시키는 동시에 최저 가격을 제공할 가능성이 높습니다.

플릿이 크거나 장시간 실행될 경우 스팟 인스턴스를 여러 풀로 분산하여 플릿의 가용성을 높일 수 있습니다. 예를 들어 EC2 집합이 풀 10개와 인스턴스 100개의 목표 용량을 지정하면 플릿이 각 풀에서 스팟 인스턴스 10개를 시작합니다. 풀에서 스팟 가격이 최고 가격을 초과하는 경우, 풀 중 10%만 영향을 받습니다. 이 전략을 사용하면 플릿이 시간이 지나면서 어느 한 풀에서 발생하는 스팟 가격의 상승에 덜 민감해집니다.

`diversified` 전략 사용 시 EC2 집합은 [온디맨드 가격](#)보다 높거나 이 가격과 동일한 스팟 가격의 풀로 스팟 인스턴스를 시작하지 않습니다.

저렴하고 다각화된 플릿을 생성하려면 `lowest-price`과 함께 `InstancePoolsToUseCount` 전략을 사용하십시오. 스팟 인스턴스를 할당 할 스팟 풀 수로써 낮은 수 또는 높은 수를 사용할 수 있습니다. 예를 들어 일괄 처리를 실행하는 경우 대기열이 비용 절감 효과를 극대화하는 동시에 컴퓨팅 파워를 항상 확보할 수 있도록 스팟 풀 수를 낮게(예: `InstancePoolsToUseCount=2`) 지정하는 것이 좋습니다. 웹 서비스를 실행하는 경우 스팟 인스턴스 풀을 일시적으로 사용할 수 없게 되었을 때 그 충격을 최소화할 수 있도록 스팟 풀 수를 낮게(예: `InstancePoolsToUseCount=10`) 지정하는 것이 좋습니다.

플릿이 작업 재시작 및 검사와 연관된 중단 비용이 높을 수 있는 워크로드를 실행하는 경우 `capacity-optimized` 전략을 사용하십시오. 이 전략은 중단을 줄일 수 있는 가능성을 제공함으로써 전체 워크로드 비용을 낮출 수 있습니다.

온디맨드 백업을 위한 EC2 집합 구성

주요 뉴스 이벤트나 게임 출시 중에 조정이 필요한 뉴스 웹 사이트와 같이 예측할 수 없는 조정 요구가 있을 경우 선호하는 옵션에 충분한 가용 용량이 없으면 온디맨드 인스턴스의 대체 인스턴스 유형을 지정하는 것이 좋습니다. 예를 들어 `c5.2xlarge` 온디맨드 인스턴스를 선호하지만 가용 용량이 부족하다면 피크 로드 중에 `c4.2xlarge` 인스턴스를 몇 개 사용하고 싶은 경우가 있을 수 있습니다. 이 경우 EC2 집합은 `c5.2xlarge` 인스턴스를 사용하여 목표 용량을 모두 충족하려고 하지만 용량이 부족하면 자동으로 `c4.2xlarge` 인스턴스를 시작하여 목표 용량을 충족합니다.

온디맨드 용량에 대한 인스턴스 유형 우선순위 지정

EC2 집합가 온디맨드 용량을 채우려고 시도하는 경우 기본적으로 최저 가격의 인스턴스 유형을 먼저 시작합니다. `AllocationStrategy`가 `prioritized`로 설정된 경우 EC2 집합가, 우선 순위를 통해, 온디맨드 용량을 채우기 위해 먼저 사용할 인스턴스 유형을 결정합니다. 시작 템플릿 재정의에 우선 순위를 할당하고 우선 순위가 가장 높은 것을 먼저 시작합니다.

예를 들어 서로 다른 인스턴스 유형인 `c3.large`, `c4.large`, `c5.large`를 각각 지닌 3개의 시작 템플릿 재정의를 구성했다고 가정해 보겠습니다. `c5.large`에 대한 온디맨드 가격은 `c4.large`에 대한 온디맨드 가격보다 낮습니다. `c3.large`가 가장 낮습니다. 우선 순위를 사용해 순서를 결정하지 않는 경우 플릿이 `c3.large`로 시작하여 온디맨드 용량을 채운 후 `c5.large`를 사용합니다. 종종 `c4.large`에 대한 미사용 예약 인스턴스가 있게 되므로 `c4.large`, `c3.large`, `c5.large`의 순서이도록 시작 템플릿 재정의 우선 순위를 설정할 수 있습니다.

온디맨드 인스턴스에 대해 용량 예약 사용

용량 예약을 위한 사용 전략을 `use-capacity-reservations-first`로 설정하여 온디맨드 인스턴스를 시작할 때 온디맨드 용량 예약부터 사용하도록 플릿을 구성할 수 있습니다. 이 설정을 온디맨드 인스턴스에 대한 할당 전략(`lowest-price` 또는 `prioritized`)과 함께 사용할 수 있습니다.

미사용 용량 예약을 사용하여 온디맨드 용량을 이행할 경우:

- 플릿은 미사용 용량 예약을 사용하여 온디맨드 용량을 목표 온디맨드 용량까지 이행합니다.
- 여러 인스턴스 폴에 미사용 용량 예약이 있는 경우, 온디맨드 할당 전략(`lowest-price` 또는 `prioritized`)이 적용됩니다.
- 미사용 용량 예약 수가 온디맨드 목표 용량보다 적을 경우, 온디맨드 할당 전략(`lowest-price` 또는 `prioritized`)에 따라 나머지 온디맨드 목표 용량이 시작됩니다.

미사용 온디맨드 용량 예약은 `instant` 유형의 플릿에만 사용할 수 있습니다.

온디맨드 용량을 이행하기 위해 용량 예약을 사용하도록 플릿을 구성하는 방법의 예는 [EC2 집합 구성의 예](#) (p. 426) 단원을 참조하십시오. 자세한 내용은 [온디맨드 용량 예약](#) (p. 354) 및 [온디맨드 용량 예약 FAQ](#)를 참조하십시오.

최고 가격 재정의

각 EC2 집합은 글로벌 최고 가격을 포함하거나 기본 가격(온디맨드 가격)을 사용할 수 있습니다. 플릿은 각 시작 사양의 기본 최고 가격으로 이 가격을 사용합니다.

하나 이상의 시작 사양에서 최고 가격을 선택적으로 지정할 수 있습니다. 이 가격은 시작 사양에 특정한 것입니다. 시작 사양에 특정 가격이 포함되는 경우 EC2 집합은 글로벌 최고 가격 대신 이 최고 가격을 사용합니다. 특정 최고 가격을 포함하지 않는 다른 시작 사양은 글로벌 최고 가격을 계속해서 사용합니다.

지출 제어

EC2 집합은 `TotalTargetCapacity` 또는 `MaxTotalPrice`(지불할 최대 금액) 파라미터 중 하나를 충족하면 인스턴스 실행을 중지합니다. 플릿에 대해 시간당 지불하는 금액을 관리하려면 `MaxTotalPrice`를 지정할 수 있습니다. 최대 총 가격에 도달하면 EC2 집합은 목표 용량을 충족하지 않은 경우에도 인스턴스 실행을 중지합니다.

다음 예제와 같이 이 작업을 두 가지 시나리오로 수행할 수 있습니다. 첫 번째 시나리오에서 EC2 집합은 대상 용량을 충족했을 때 인스턴스 실행을 중지합니다. 두 번째 시나리오에서 EC2 집합은 지불할 최대 금액에 도달하면 인스턴스 실행을 중지합니다(`MaxTotalPrice`).

예: 대상 용량에 도달할 때 인스턴스 실행 중지

다음과 같은 `m4.large` 온디맨드 인스턴스 요청 시:

- 온디맨드 가격: 시간당 0.10 USD

- **OnDemandTargetCapacity:** 10
- **MaxTotalPrice:** 1.50 USD

EC2 집합은 온디맨드 인스턴스의 경우 최대 1.00 USD(10개 인스턴스 x 0.10 USD)가 **MaxTotalPrice** 1.50 USD를 초과하지 않기 때문에 10개의 온디맨드 인스턴스를 시작합니다.

예: 최대 총 가격에 도달할 때 인스턴스 실행 중지

다음과 같은 m4.large 온디맨드 인스턴스 요청 시:

- 온디맨드 가격: 시간당 0.10 USD
- **OnDemandTargetCapacity:** 10
- **MaxTotalPrice:** 0.80 USD

EC2 집합이 온디맨드 대상 용량(온디맨드 인스턴스 10개)을 시작하면 시간당 총 비용은 1.00 USD입니다. 온디맨드 인스턴스의 **MaxTotalPrice**에 대해 지정된 금액(0.80 USD) 보다 높습니다. 지불할 금액보다 더 많은 지출을 방지하기 위해 EC2 집합은 8개의 온디맨드 인스턴스(온디맨드 대상 용량 미만) 만 실행합니다. 더 많이 실행하면 온디맨드 인스턴스의 **MaxTotalPrice**를 초과할 수 있기 때문입니다.

EC2 집합 인스턴스 가중치 부여

EC2 집합을 생성할 때 각 인스턴스 유형이 애플리케이션의 성능에 기여할 수 있는 용량 단위를 정의할 수 있습니다. 인스턴스 가중치를 사용하여 각 시작 사양의 최대 가격을 조정할 수 있습니다.

기본적으로, 사용자가 지정하는 가격은 인스턴스 시간당 가격입니다. 인스턴스 가중치 기능을 사용할 때, 사용자가 지정하는 가격은 단위 시간당 가격입니다. 단위 시간당 가격은 인스턴스 유형에 따른 가격을 인스턴스가 나타내는 단위 수로 나누어 계산합니다. EC2 집합은 목표 용량을 인스턴스 가중치로 나누어 시작할 인스턴스 수를 계산합니다. 결과가 정수가 아닌 경우 플릿은 결과를 다음 정수로 반올림하므로 플릿 크기가 목표 용량을 밀들지는 않습니다. 시작된 인스턴스의 용량이 요청된 목표 용량을 초과하더라도 플릿은 시작 사양에 지정한 어떤 풀이든 선택할 수 있습니다.

다음 표에는 목표 용량이 10인 EC2 집합의 단위당 가격을 결정하기 위한 계산의 예가 있습니다.

인스턴스 유형	인스턴스 가중치	목표 용량	시작된 인스턴스의 수	인스턴스 시간당 가격	단위 시간당 가격
r3.xlarge	2	10	5 (10을 2로 나눈 값)	0.05 USD	0.025 USD (0.05를 2로 나눈 값)
r3.8xlarge	8	10	2 (10을 8로 나눈 후 올림한 결과)	0.10 USD	0.0125 USD (0.10를 8로 나눈 값)

EC2 집합 인스턴스 가중치를 사용하여 다음과 같이 원하는 목표 용량을 이행 시점의 단위당 최저 가격으로 풀에서 프로비저닝합니다.

1. EC2 집합의 목표 용량을 인스턴스(기본값) 또는 선택한 단위(예: 가상 CPU 수, 메모리, 스토리지 또는 처리량)로 설정합니다.
2. 단위당 가격을 설정합니다.
3. 목표 용량에 대해 인스턴스 유형이 나타내는 단위 수를 의미하는 가중치를 시작 사양마다 지정합니다.

인스턴스 가중치 부여의 예

다음과 같은 구성의 EC2 집합 요청을 고려하십시오.

- 목표 용량은 24
- 인스턴스 유형이 r3.2xlarge이고 가중치가 6인 시작 사양
- 인스턴스 유형이 c3.xlarge이고 가중치가 5인 시작 사양

가중치는 목표 용량에 대하여 인스턴스 유형이 나타내는 단위 수를 의미합니다. 첫 번째 시작 사양에서 단위당 최저 가격(인스턴스 시간당 r3.2xlarge의 가격을 6으로 나눈 값)을 제공하는 경우 EC2 집합은 이 인스턴스 중 4개(24를 6으로 나눈 값)를 시작합니다.

두 번째 시작 사양에서 단위당 최저 가격(인스턴스 시간당 c3.xlarge에 대한 가격을 5로 나눈 값)을 제공하는 경우 EC2 집합은 이들 인스턴스 중 5개(24를 5로 나눈 결과를 올림한 값)를 시작합니다.

인스턴스 가중치 부여 및 할당 전략

다음과 같은 구성의 EC2 집합 요청을 고려하십시오.

- 목표 용량 스팟 인스턴스 30개
- 인스턴스 유형이 c3.2xlarge이고 가중치가 8인 시작 사양
- 인스턴스 유형이 m3.xlarge이고 가중치가 8인 시작 사양
- 인스턴스 유형이 r3.xlarge이고 가중치가 8인 시작 사양

EC2 집합이 4개의 인스턴스(30을 8로 나눈 결과를 올림한 값)를 시작합니다. `lowest-price` 전략 사용 시, 4개의 인스턴스는 전부 단위당 최저 가격을 제공하는 풀에서 가져옵니다. `diversified` 전략 사용 시 풀릿은 풀 3개에서 각각 인스턴스 1개를 시작하고 풀 3개 중 어디에 있든 네 번째 인스턴스가 단위당 최저 가격을 제공합니다.

자습서: 인스턴스 가중치를 부여한 EC2 집합 사용

이 자습서에서는 Example Corp이라는 가상의 회사를 통해 인스턴스 가중치를 사용하여 EC2 집합을 요청하는 프로세스를 설명합니다.

목표

제약 회사인 Example Corp은 암 퇴치 효과가 있는 화합물을 검출하는 데 Amazon EC2의 컴퓨팅 능력을 사용하려고 합니다.

계획

Example Corp은 먼저 [스팟 모범 사례](#)를 살펴봅니다. 그런 다음 Example Corp이 EC2 집합에 대해 다음과 같은 요구 사항을 결정합니다.

인스턴스 유형

Example Corp은 최소 60GB 메모리와 8개의 가상 CPU(vCPU)로 최적의 성능을 자랑하는 컴퓨팅 및 메모리 집약적 애플리케이션을 사용하고 있습니다. 하지만 최저 가격으로 이러한 애플리케이션 리소스를 극대화하는 것이 목표입니다. 그 결과 다음 EC2 인스턴스 유형 중 하나가 이러한 요건에 적합할 것이라는 결정을 내립니다.

인스턴스 유형	메모리(GiB)	vCPUs
r3.2xlarge	61	8
r3.4xlarge	122	16

인스턴스 유형	메모리(GiB)	vCPUs
r3.8xlarge	244	32

목표 용량 단위

인스턴스 가중치를 부여했을 때 목표 용량은 인스턴스 수(기본값) 또는 코어(vCPU), 메모리(GiB) 및 스토리지(GB)와 같은 요소의 조합과 동일할 수 있습니다. 그래서 Example Corp은 단위 1개당 애플리케이션의 기본 용량(60GB 메모리, vCPU 8개)을 고려하여 기본 용량의 20배가 요구에 적합하겠다고 결정하고 그래서 EC2 집합 요청의 목표 용량을 20으로 설정합니다.

인스턴스 가중치

목표 용량이 결정되자 이제는 인스턴스 가중치를 계산합니다. 각 인스턴스 유형에 대한 인스턴스 가중치를 계산하기 위해, 다음과 같이 목표 용량에 이르기 위해 필요한 각 인스턴스 유형의 단위를 결정합니다.

- r3.2xlarge(61.0GB, 8 vCPU) = 단위 20개 중 1개
- r3.4xlarge(122.0GB, 16 vCPU) = 단위 20개 중 2개
- r3.8xlarge(122.0GB, 32 vCPU) = 단위 20개 중 4개

따라서 Example Corp은 EC2 집합 요청 시 1, 2 및 4의 인스턴스 가중치를 각 시작 구성에 할당합니다.

단위 시간당 가격

Example Corp은 인스턴스 시간당 [온디맨드 가격](#)을 시작 가격으로 사용합니다. 그 밖에 최근 스팟 가격을 사용하거나, 둘을 조합할 수도 있습니다. 단위 시간당 가격을 계산하려면 인스턴스 시간당 시작 가격을 가중치로 나눕니다. 예:

인스턴스 유형	온디맨드 가격	인스턴스 가중치	단위 시간당 가격
r3.2xLarge	\$0.7	1	\$0.7
r3.4xLarge	\$1.4	2	\$0.7
r3.8xLarge	\$2.8	4	\$0.7

Example Corp은 단위 시간당 글로벌 가격으로 0.7 USD를 사용하기 때문에 세 가지 인스턴스 유형 모두에서 경쟁력이 있습니다. 또한 r3.8xlarge 시작 사양에서 단위 시간당 글로벌 가격으로 0.7 USD를, 단위 시간당 특정 가격으로 0.9 USD를 사용할 수도 있습니다.

권한 검증

Example Corp은 EC2 집합을 생성하기 전에 필요한 권한을 가진 IAM 역할이 있는지 확인합니다. 자세한 내용은 [EC2 집합 사전 조건 \(p. 412\)](#) 단원을 참조하십시오.

EC2 집합 생성

Example Corp은 EC2 집합에 대해 다음 구성으로 config.json 파일을 생성합니다.

```
{  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-07b3bc7625cdab851",  
                "Version": "1"  
            }  
        }  
    ]  
}
```

```
        },
        "Overrides": [
            {
                "InstanceType": "r3.2xlarge",
                "SubnetId": "subnet-482e4972",
                "WeightedCapacity": 1
            },
            {
                "InstanceType": "r3.4xlarge",
                "SubnetId": "subnet-482e4972",
                "WeightedCapacity": 2
            },
            {
                "InstanceType": "r3.8xlarge",
                "MaxPrice": "0.90",
                "SubnetId": "subnet-482e4972",
                "WeightedCapacity": 4
            }
        ]
    },
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    }
}
```

Example Corp은 다음 [create-fleet](#) 명령을 사용하여 EC2 집합을 생성합니다.

```
aws ec2 create-fleet \
    --cli-input-json file://config.json
```

자세한 내용은 [EC2 집합 생성 \(p. 418\)](#) 단원을 참조하십시오.

이행

할당 전략에서는 스팟 인스턴스가 어느 스팟 인스턴스 풀에서 온 것인지 확인합니다.

`lowest-price` 전략(기본 전략)을 사용할 경우 스팟 인스턴스는 이행 시점에 단위당 최저 가격의 풀에서 옵니다. 20단위의 용량을 제공하기 위해 EC2 집합이 `r3.2xlarge` 인스턴스 20개(20을 1로 나눈 값), `r3.4xlarge` 인스턴스 10개(20을 2로 나눈 값) 또는 `r3.8xlarge` 인스턴스 5개(20을 4로 나눈 값)를 시작합니다.

Example Corp에서 `diversified` 전략을 사용한 경우에는 스팟 인스턴스가 3개의 풀 전부에서 옵니다. EC2 집합은 총 20개의 단위에 대해 `r3.2xlarge` 인스턴스 6개(6개 단위 제공), `r3.4xlarge` 인스턴스 3개(6개 단위 제공), `r3.8xlarge` 인스턴스 2개(8개 단위 제공)를 시작합니다.

자습서: 기본 용량이 온디맨드인 EC2 집합 사용

이 자습서에서는 ABC Online이라는 가상의 회사를 통해 기본 용량인 온디맨드와 스팟 용량(사용할 수 있는 경우)이 있는 EC2 집합을 요청하는 프로세스를 설명합니다.

목표

식당 배달 회사인 ABC Online은 EC2 인스턴스 유형 및 구매 옵션에 Amazon EC2 용량을 프로비저닝하여 원하는 규모, 성능 및 비용을 달성하려고 합니다.

계획

ABC Online은 피크 시간의 운영을 위해 고정 용량이 필요하지만 저렴한 가격으로 더 큰 용량을 이용하고자 합니다. ABC Online은 EC2 집합에 대해 다음 요구 사항을 결정합니다.

- 온디맨드 인스턴스 용량 – ABC Online은 온디맨드 인스턴스 15개가 있어야 피크 시간의 트래픽을 수용할 수 있습니다.
- 스팟 인스턴스 용량 – ABC Online은 저렴한 비용으로 스팟 인스턴스 5개를 프로비저닝하여 성능을 개선하려고 합니다.

권한 검증

ABC Online은 EC2 집합을 생성하기 전에 필요한 권한을 가진 IAM 역할이 있는지 확인합니다. 자세한 내용은 [EC2 집합 사전 조건 \(p. 412\)](#) 단원을 참조하십시오.

EC2 집합 생성

ABC Online은 EC2 집합에 대해 다음 구성으로 config.json 파일을 생성합니다.

```
{  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-07b3bc7625cdab851",  
                "Version": "2"  
            }  
        }  
    ],  
    "TargetCapacitySpecification": {  
        "TotalTargetCapacity": 20,  
        "OnDemandTargetCapacity": 15,  
        "DefaultTargetCapacityType": "spot"  
    }  
}
```

ABC Online은 다음 [create-fleet](#) 명령을 사용하여 EC2 집합을 생성합니다.

```
aws ec2 create-fleet \  
  --cli-input-json file://config.json
```

자세한 내용은 [EC2 집합 생성 \(p. 418\)](#) 단원을 참조하십시오.

이행

할당 전략에 따라 온디맨드 용량이 항상 충족되지만 사용 가능한 용량이 있는 경우 남아 있는 목표 용량이 스팟으로 충족되도록 결정됩니다.

EC2 집합 관리

EC2 집합을 사용하려면 총 목표 용량, 온디맨드 용량, 스팟 용량, 인스턴스에 대한 하나 이상의 시작 사양, 지불하려는 최고 가격을 포함하는 요청을 생성합니다. AMI, 인스턴스 유형, 서브넷 또는 가용 영역 및 하나 이상의 보안 그룹과 같이 인스턴스를 시작하기 위해 필릿에 필요한 정보를 정의하는 시작 템플릿을 플릿 요청에 포함해야 합니다. 인스턴스 유형, 서브넷, 가용 영역 및 지불하려는 최고 가격의 시작 사양 재정의를 지정할 수 있으며 가중치가 적용된 용량을 각 시작 사양 재정의에 할당할 수 있습니다.

플릿에 스팟 인스턴스가 포함되어 있으면 Amazon EC2에서 스팟 가격의 변화에 따라 플릿 목표 용량을 유지하려고 할 수 있습니다.

EC2 집합 요청은 요청이 만료되거나 삭제될 때까지 활성 상태로 유지됩니다. 플릿을 삭제할 때 삭제로 인해 해당 플릿의 인스턴스가 종료될지 여부를 지정할 수 있습니다.

목차

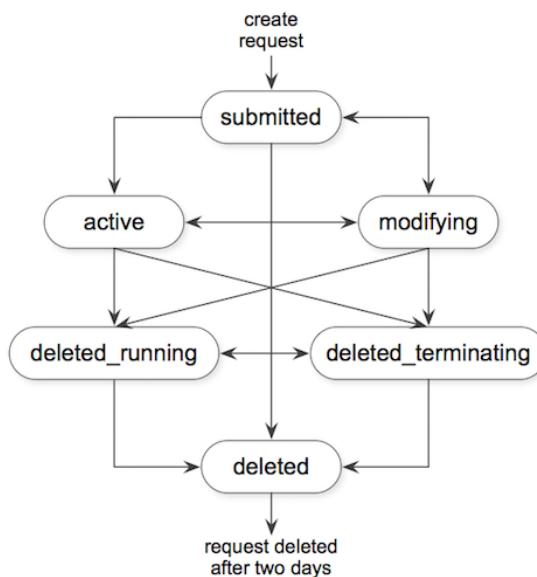
- EC2 집합 요청 상태 (p. 411)
- EC2 집합 사전 조건 (p. 412)
- EC2 집합 상태 확인 (p. 414)
- EC2 집합 JSON 구성 파일 생성 (p. 414)
- EC2 집합 생성 (p. 418)
- EC2 집합 태그 지정 (p. 421)
- EC2 집합 모니터링 (p. 423)
- EC2 집합 수정 (p. 424)
- EC2 집합 삭제 (p. 425)
- EC2 집합 구성의 예 (p. 426)

EC2 집합 요청 상태

EC2 집합 요청은 다음 상태 중 하나일 수 있습니다.

- **submitted** – EC2 집합 요청을 평가 종이며 Amazon EC2가 목표 개수만큼 인스턴스를 시작할 준비 중입니다(온디맨드 인스턴스, 스팟 인스턴스, 또는 둘 다 포함할 수 있음).
- **active** – EC2 집합 요청이 확인되었으며 Amazon EC2가 실행 중인 인스턴스를 목표 개수만큼 유지하려고 시도하고 있습니다. 요청은 수정하거나 삭제할 때까지 이 상태로 유지됩니다.
- **modifying** – EC2 집합 요청을 수정하고 있습니다. 수정이 완전히 처리되거나 요청이 삭제될 때까지 요청이 이 상태로 유지됩니다. **maintain** 요청 유형만 수정할 수 있습니다. 이 상태는 다른 요청 유형에는 적용되지 않습니다.
- **deleted_running** – EC2 집합 요청이 삭제되고 추가 인스턴스를 시작하지 않습니다. 종단되거나 종료될 때까지 기존 인스턴스가 계속 실행됩니다. 그 요청은 모든 인스턴스가 종단 또는 종료될 때까지 계속 이 상태로 유지됩니다.
- **deleted_terminating** – EC2 집합 요청이 삭제되고 해당 인스턴스가 종료됩니다. 그 요청은 모든 인스턴스가 종료될 때까지 계속 이 상태로 유지됩니다.
- **deleted** – EC2 집합이 삭제되고 실행 중인 인스턴스가 없습니다. 인스턴스가 종료되고 2일 후 요청이 삭제됩니다.

다음 그림은 EC2 집합 요청 상태의 전환을 나타냅니다. 플릿 제한을 초과하면 즉시 요청이 삭제됩니다.



EC2 집합 사전 조건

EC2 집합을 생성하려면 다음 사전 조건이 갖춰져야 합니다.

시작 템플릿

시작 템플릿에는 인스턴스 유형, 가용 영역, 지불하려는 최고 가격 등 시작할 인스턴스에 대한 정보가 포함됩니다. 자세한 내용은 [시작 템플릿에서 인스턴스 시작 \(p. 381\)](#) 단원을 참조하십시오.

EC2 집합의 서비스 연결 역할

AWS Service Role for EC2 Fleet 역할은 사용자를 대신해 인스턴스를 요청, 시작, 종료 및 태그 지정할 권한을 EC2 집합에 부여합니다. Amazon EC2는 이 서비스 연결 역할을 사용하여 다음 작업을 완료합니다.

- `ec2:RunInstances` – 인스턴스를 시작합니다.
- `ec2:RequestSpotInstances` – 스팟 인스턴스 요청.
- `ec2:TerminateInstances` – 인스턴스를 종료합니다.
- `ec2:DescribeImages` – 스팟 인스턴스용 Amazon Machine Image(AMI)를 설명합니다.
- `ec2:DescribeInstanceStatus` – 스팟 인스턴스의 상태를 설명합니다.
- `ec2:DescribeSubnets` – 스팟 인스턴스용 서브넷에 대해 설명합니다.
- `ec2:CreateTags` – EC2 집합, 인스턴스 및 볼륨에 태그를 추가합니다.

AWS CLI 또는 API를 사용하여 EC2 집합을 만들려면 먼저 이 역할이 있어야 합니다.

Note

instant EC2 집합에는 이 역할이 필요하지 않습니다.

역할을 생성하려면 다음과 같이 IAM 콘솔을 사용하십시오.

EC2 집합에 대한 AWS Service Role for EC2 Fleet 역할 생성

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 역할을 선택한 후 역할 생성을 선택합니다.
3. 신뢰할 수 있는 유형의 엔터티 선택에서 AWS 서비스를 선택합니다.
4. Choose the service that will use this role(이 역할을 사용할 서비스 선택)에서 EC2 - Fleet(EC2 - 플랫폼)을 선택한 다음 Next: Permissions(다음: 권한), Next: Tags(다음: 태그), 및 Next: Review(다음: 검토)를 선택합니다.
5. 검토 페이지에서 역할 만들기를 선택합니다.

EC2 집합이 더 이상 필요 없으면 AWS Service Role for EC2 Fleet 역할을 삭제하는 것이 좋습니다. 계정에서 이 역할을 삭제한 후 다른 플랫폼을 생성하면 다시 역할을 만들 수 있습니다.

자세한 내용은 IAM 사용 설명서의 [서비스 연결 역할 사용](#)을 참조하십시오.

암호화된 AMI 및 EBS 스냅샷과 함께 사용할 CMK에 대한 액세스 권한 부여

EC2 집합 요청에서 [암호화된 AMI \(p. 98\)](#) 또는 [암호화된 Amazon EBS 스냅샷 \(p. 1033\)](#)을 지정하고 암호화 용 고객 관리형 고객 마스터 키(CMK)를 사용하는 경우 AWS Service Role for EC2 Fleet 역할에 CMK를 사용할 수 있는 권한을 부여해야 Amazon EC2가 사용자를 대신하여 스팟 인스턴스를 시작할 수 있습니다. 이렇게 하려면 다음 절차에 표시된 바와 같이 CMK에 권한을 추가해야 합니다.

권한을 제공할 때 권한 부여는 키 정책을 대체합니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [권한 부여 사용](#) 및 [AWS KMS에서 키 정책 사용](#) 단원을 참조하십시오.

CMK를 사용할 수 있도록 AWSServiceRoleForEC2Fleet 역할 권한을 부여하려면

- `create-grant` 명령을 사용하여 CMK에 권한을 추가하고 허용된 작업을 수행할 수 있는 권한이 부여된 보안 주체(AWSServiceRoleForEC2Fleet 서비스 연결 역할)를 지정합니다. CMK는 CMK의 key-id 파라미터 및 ARN에 의해 지정됩니다. 보안 주체는 AWSServiceRoleForEC2Fleet 서비스 연결 역할의 grantee-principal 파라미터 및 ARN에 의해 지정됩니다.

```
aws kms create-grant \
--region us-east-1 \
--key-id arn:aws:kms:us-
east-1:44445556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
--grantee-principal arn:aws:iam::111122223333:role/AWSServiceRoleForEC2Fleet \
--operations "Decrypt" "Encrypt" "GenerateDataKey"
"GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
"ReEncryptTo"
```

EC2 집합 및 IAM 사용자

EC2 집합을 생성하거나 관리하는 IAM 사용자에게는 다음과 같이 필요한 권한을 부여해야 합니다.

EC2 집합의 IAM 사용자 권한을 부여하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 정책을 선택합니다.
3. 정책 생성을 선택합니다.
4. 정책 생성 페이지에서 JSON 탭을 선택한 다음, 텍스트를 다음과 같이 바꾸고 정책 검토를 선택합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam>ListRoles",
                "iam:PassRole",
                "iam>ListInstanceProfiles"
            ],
            "Resource": "*"
        }
    ]
}
```

`ec2:*`는 모든 Amazon EC2 API 작업을 호출할 수 있는 IAM 사용자 권한을 부여합니다. 사용자를 특정 Amazon EC2 API 작업으로 제한하려면 해당 작업을 대신 지정하십시오.

IAM 사용자에는 기존 IAM 역할을 열거하는 `iam>ListRoles` 작업, EC2 집합 역할을 지정하는 `iam:PassRole` 작업 및 기존 인스턴스 프로파일을 열거하는 `iam>ListInstanceProfiles` 작업을 호출할 수 있는 권한이 있어야 합니다.

(선택 사항) IAM 사용자가 IAM 콘솔을 사용하여 역할 또는 인스턴스 프로파일을 생성할 수 있도록 하려면 정책에 다음 작업도 추가해야 합니다.

- iam:AddRoleToInstanceProfile
 - iam:AttachRolePolicy
 - iam>CreateInstanceProfile
 - iam>CreateRole
 - iam:GetRole
 - iam>ListPolicies
5. 정책 검토 페이지에 정책 이름과 설명을 입력한 다음 정책 생성을 선택합니다.
6. 탐색 창에서 사용자를 선택하고 사용자를 선택합니다.
7. 권한 탭에서 권한 추가를 선택합니다.
8. Attach existing policies directly(기존 정책 직접 연결)를 선택합니다. 앞에서 만든 정책을 선택하고 다음: 검토(Next: Review)를 선택합니다.
9. 권한 추가를 선택합니다.

EC2 집합 상태 확인

EC2 집합은 분마다 플릿의 인스턴스 상태를 확인합니다. 인스턴스의 상태는 healthy 또는 unhealthy입니다. 플릿은 Amazon EC2에서 제공하는 상태 확인을 사용하여 인스턴스 상태를 판단합니다. 세 번의 연속 상태 확인에서 인스턴스 상태 또는 시스템 상태가 impaired이면, 해당 인스턴스의 상태는 unhealthy입니다. 그렇지 않으면 상태는 healthy입니다. 자세한 내용은 [인스턴스 상태 확인 \(p. 651\)](#) 단원을 참조하십시오.

비정상 인스턴스를 교체하도록 EC2 집합을 구성할 수 있습니다. 상태 확인 교체를 활성화하면 상태가 unhealthy로 보고된 인스턴스가 교체됩니다. 플릿은 비정상 인스턴스가 교체되는 동안 최대 몇 분간 목표 용량을 밀들 수 있습니다.

요구 사항

- 상태 확인 교체는 목표 용량을 유지하는 EC2 집합에서만 지원되고 일회성 플릿에서는 지원되지 않습니다.
- 비정상 인스턴스를 생성할 경우에만 이를 교체하도록 EC2 집합을 구성할 수 있습니다.
- IAM 사용자는 ec2:DescribeInstanceStatus 작업을 호출할 권리 있는 경우에만 상태 확인 교체를 사용할 수 있습니다.

EC2 집합 JSON 구성 파일 생성

EC2 집합을 생성하려면 시작 템플릿, 총 목표 용량 및 기본 구매 옵션이 온디マン드인지 스팟인지 여부를 지정해야 합니다. 이 파라미터를 지정하지 않으면 플릿에 기본값이 사용됩니다. 플릿 구성 파라미터의 전체 목록을 보려면 다음과 같이 JSON 파일을 생성할 수 있습니다.

명령줄을 사용하여 가능한 모든 EC2 집합 파라미터가 포함된 JSON 파일을 생성하려면

- [create-fleet\(AWS CLI\)](#) 명령 및 --generate-cli-skeleton 파라미터를 사용하여 EC2 집합 JSON 파일을 생성합니다.

```
aws ec2 create-fleet \
--generate-cli-skeleton
```

사용할 수 있는 EC2 집합 파라미터는 다음과 같습니다.

```
{
  "DryRun": true,
  "ClientToken": "",
```

```
"SpotOptions": {
    "AllocationStrategy": "lowest-price",
    "InstanceInterruptionBehavior": "hibernate",
    "InstancePoolsToUseCount": 0,
    "SingleInstanceType": true,
    "SingleAvailabilityZone": true,
    "MaxTotalPrice": 0,
    "MinTargetCapacity": 0
},
"OnDemandOptions": {
    "AllocationStrategy": "prioritized",
    "SingleInstanceType": true,
    "SingleAvailabilityZone": true,
    "MaxTotalPrice": 0,
    "MinTargetCapacity": 0
},
"ExcessCapacityTerminationPolicy": "termination",
"LaunchTemplateConfigs": [
    {
        "LaunchTemplateSpecification": {
            "LaunchTemplateId": "",
            "LaunchTemplateName": "",
            "Version": ""
        },
        "Overrides": [
            {
                "InstanceType": "t2.micro",
                "MaxPrice": "",
                "SubnetId": "",
                "AvailabilityZone": "",
                "WeightedCapacity": null,
                "Priority": null,
                "Placement": {
                    "AvailabilityZone": "",
                    "Affinity": "",
                    "GroupName": "",
                    "PartitionNumber": 0,
                    "HostId": "",
                    "Tenancy": "dedicated",
                    "SpreadDomain": ""
                }
            }
        ]
    }
],
"TargetCapacitySpecification": {
    "TotalTargetCapacity": 0,
    "OnDemandTargetCapacity": 0,
    "SpotTargetCapacity": 0,
    "DefaultTargetCapacityType": "spot"
},
"TerminateInstancesWithExpiration": true,
"Type": "maintain",
"ValidFrom": "1970-01-01T00:00:00",
"ValidUntil": "1970-01-01T00:00:00",
"ReplaceUnhealthyInstances": true,
"TagSpecifications": [
    {
        "ResourceType": "fleet",
        "Tags": [
            {
                "Key": "",
                "Value": ""
            }
        ]
    }
]
```

}

EC2 집합 JSON 구성 파일 참조

Note

모든 파라미터 값에 소문자를 사용하십시오. 그렇지 않으면 Amazon EC2에서 JSON 파일을 사용하여 EC2 집합을 시작할 때 오류가 발생합니다.

AllocationStrategy(SpotOptions를 위한)

(선택 사항) EC2 집합에 지정된 스팟 인스턴스 폴에 스팟 인스턴스 목표 용량을 할당하는 방법을 나타냅니다. 유효 값은 `lowest-price`, `diversified` 및 `capacity-optimized`입니다. 기본값은 `lowest-price`입니다. 필요에 맞는 할당 전략을 지정합니다. 자세한 내용은 [스팟 인스턴스를 위한 할당 전략 \(p. 403\)](#) 단원을 참조하십시오.

InstanceInterruptionBehavior

(선택) 스팟 인스턴스가 중단될 경우 동작입니다. 유효 값은 `hibernate`, `stop` 및 `terminate`입니다. 스팟 서비스는 중단된 스팟 인스턴스를 종료하도록 기본 설정되어 있습니다. 플릿 유형이 `maintain`인 경우 스팟 서비스가 중단된 스팟 인스턴스를 최대 절전 모드로 전환하거나 중지하도록 지정할 수 있습니다.

InstancePoolsToUseCount

대상 스팟 용량을 할당할 스팟 폴 수입니다. 스팟 AllocationStrategy가 `lowest-price`로 설정된 경우에만 유효합니다. EC2 집합은 가장 저렴한 스팟 폴을 선택하고 지정한 스팟 폴 수에 걸쳐 대상 스팟 용량을 균등하게 할당합니다.

SingleInstanceType

플릿이 단일 인스턴스 유형을 사용하여 플릿에 있는 모든 스팟 인스턴스를 시작하는 것을 나타냅니다.

SingleAvailabilityZone

플릿이 단일 가용 영역으로 모든 스팟 인스턴스를 시작하는 것을 나타냅니다.

MaxTotalPrice

스팟 인스턴스에 대해 지불할 최대 시간당 금액입니다.

MinTargetCapacity

플릿의 스팟 인스턴스에 대한 최소 대상 용량입니다. 최소 대상 용량에 도달하지 못하면 플릿은 아무 인스턴스도 시작하지 않습니다.

AllocationStrategy(OnDemandOptions를 위한)

온디맨드 용량을 채우기 위해 사용할 시작 템플릿 재정의 순서입니다. `lowest-price`를 지정하면 EC2 집합이 가격을 통해 순서를 결정하여 최저 가격을 먼저 시작합니다. 우선 순위를 지정하면 EC2 집합이 각 시작 템플릿 재정의에 할당된 우선 순위를 사용하여 최우선 순위를 먼저 시작합니다. 값을 지정하지 않으면 EC2 집합이 `lowest-price`를 기본값으로 사용합니다.

SingleInstanceType

플릿이 단일 인스턴스 유형을 사용하여 플릿에 있는 모든 온디맨드 인스턴스를 시작하는 것을 나타냅니다.

SingleAvailabilityZone

플릿이 단일 가용 영역으로 모든 온디맨드 인스턴스를 시작하는 것을 나타냅니다.

MaxTotalPrice

온디맨드 인스턴스에 대해 지불할 최대 시간당 금액입니다.

MinTargetCapacity

플릿의 온디맨드 인스턴스에 대한 최소 대상 용량입니다. 최소 대상 용량에 도달하지 못하면 플릿은 아무 인스턴스도 시작하지 않습니다.

ExcessCapacityTerminationPolicy

(선택 사항) EC2 집합의 총 목표 용량이 EC2 집합의 현재 크기보다 작아질 경우 실행 중인 인스턴스를 종료할지 여부를 나타냅니다. 유효한 값은 `no-termination` 및 `termination`입니다.

LaunchTemplateId

사용할 시작 템플릿의 ID입니다. 시작 템플릿 ID 또는 시작 템플릿 이름을 지정해야 합니다. 시작 템플릿은 Amazon 머신 이미지(AMI)를 지정해야 합니다. 시작 템플릿 생성에 대한 자세한 내용은 [시작 템플릿에서 인스턴스 시작 \(p. 381\)](#) 단원을 참조하십시오.

LaunchTemplateName

사용할 시작 템플릿의 이름입니다. 시작 템플릿 ID 또는 시작 템플릿 이름을 지정해야 합니다. 시작 템플릿은 Amazon 머신 이미지(AMI)를 지정해야 합니다. 자세한 내용은 [시작 템플릿에서 인스턴스 시작 \(p. 381\)](#) 단원을 참조하십시오.

버전

시작 템플릿 버전 번호(`$Latest` 또는 `$Default`)입니다. 값을 지정해야 하며, 그렇지 않으면 요청이 실패합니다. 값이 `$Latest`인 경우 Amazon EC2는 시작 템플릿의 최신 버전을 사용합니다. 값이 `$Default`인 경우 Amazon EC2는 시작 템플릿의 기본 버전을 사용합니다. 자세한 내용은 [시작 템플릿 버전 관리 \(p. 391\)](#) 단원을 참조하십시오.

InstanceType

(선택) 인스턴스 유형입니다. 입력하면 이 값이 시작 템플릿을 재정의합니다. 인스턴스 유형은 필요한 최소 하드웨어 사양(vCPU, 메모리 또는 스토리지)을 갖춰야 합니다.

MaxPrice

(선택) 스팟 인스턴스에 대해 지불하려는 단위 시간당 최고 가격입니다. 입력하면 이 값이 시작 템플릿을 재정의합니다. 기본 최고 가격(온디맨드 가격)을 사용하거나 지불하려는 최고 가격을 지정할 수 있습니다. 최고 가격이 지정한 인스턴스 유형의 스팟 가격보다 낮으면 스팟 인스턴스가 시작되지 않습니다.

SubnetId

(선택) 인스턴스를 시작할 서브넷 ID입니다. 입력하면 이 값이 시작 템플릿을 재정의합니다.

새 VPC를 생성하려면 Amazon VPC 콘솔로 이동합니다. 마친 후 JSON 파일로 돌아와 새 서브넷 ID를 입력합니다.

AvailabilityZone

(선택) 인스턴스를 시작할 사용 영역입니다. AWS가 스팟 인스턴스를 위한 영역을 선택하도록 하는 것이 기본값입니다. 원한다면 특정 영역을 지정할 수 있습니다. 입력하면 이 값이 시작 템플릿을 재정의합니다.

하나 이상의 사용 영역을 지정합니다. 영역에 서브넷이 2개 이상 있으면 알맞은 서브넷을 지정하십시오. 서브넷을 추가하려면 Amazon VPC 콘솔로 이동합니다. 마친 후 JSON 파일로 돌아와 새 서브넷 ID를 입력합니다.

WeightedCapacity

(선택) 지정된 인스턴스 유형에서 제공하는 단위 수입니다. 입력하면 이 값이 시작 템플릿을 재정의합니다.

우선 순위

시작 템플릿 재정의에 대한 우선 순위입니다. `AllocationStrategy`를 `prioritized`로 설정한 경우, EC2 집합이 우선 순위를 사용하여 온디맨드 용량을 채우기 위해 어떤 시작 템플릿 재정의를 먼저 사용할지

결정합니다. 가장 높은 우선 순위를 먼저 시작합니다. 0부터 시작하는 모든 숫자가 유효한 값입니다. 숫자가 작을수록 우선 순위가 높아집니다. 숫자를 설정하지 않으면 재정의가 가장 낮은 우선 순위를 갖습니다.

TotalTargetCapacity

시작할 인스턴스 수입니다. vCPU, 메모리, 스토리지 같이 애플리케이션 워크로드에 중요한 인스턴스 또는 성능 특성을 선택할 수 있습니다. 요청 유형이 `maintain`일 경우, 목표 용량을 0으로 설정하고 나중에 용량을 추가할 수 있습니다.

OnDemandTargetCapacity

(선택 사항) 시작할 온디맨드 인스턴스 수입니다. 이 수는 `TotalTargetCapacity` 미만이어야 합니다.

SpotTargetCapacity

(선택 사항) 시작할 스팟 인스턴스 수입니다. 이 수는 `TotalTargetCapacity` 미만이어야 합니다.

DefaultTargetCapacityType

`TotalTargetCapacity`의 값이 `OnDemandTargetCapacity` 및 `SpotTargetCapacity`의 조합된 값보다 크면 여기에 지정된 인스턴스 구매 옵션에 따라 두 값의 차이가 시작됩니다. 유효한 값은 `on-demand` 또는 `spot`입니다.

TerminateInstancesWithExpiration

(선택 사항) 기본적으로 EC2 집합 요청이 만료되면 Amazon EC2에서 인스턴스를 종료합니다. 기본값은 `true`입니다. 요청이 만료된 후에도 계속 실행하려면 이 파라미터의 값을 입력하지 마십시오.

Type

(선택 사항) EC2 집합이 원하는 용량을 얻기 위해 동기식 일회성 요청(`instant`) 또는 비동기식 일회성 요청(`request`)을 제출했으나 용량을 유지하려는 시도를 하지 않았거나, 용량을 사용할 수 없는 경우 대체 용량 풀에서 요청을 제출했거나, 원하는 용량을 얻기 위해 비동기식 요청을 제출하고 중단된 스팟 인스턴스를 보충해 원하는 용량을 계속해서 유지할지(`maintain`) 여부를 나타냅니다. 유효 값은 `instant`, `request` 및 `maintain`입니다. 기본 값은 `maintain`입니다. 자세한 내용은 [EC2 집합 요청 유형 \(p. 403\)](#) 단원을 참조하십시오.

ValidFrom

(선택) 특정 기간에만 유효한 요청을 생성하려면 시작 날짜를 입력합니다.

ValidUntil

(선택) 특정 기간에만 유효한 요청을 생성하려면 종료 날짜를 입력합니다.

ReplaceUnhealthyInstances

(선택 사항) 플릿에 대해 `maintain` 작업을 수행하도록 구성된 EC2 집합의 비정상 인스턴스를 교체하려면 `true`를 입력합니다. 그렇지 않은 경우 이 파라미터를 비워두십시오.

TagSpecifications

(선택 사항) 생성 시 EC2 집합 요청에 태그를 지정하기 위한 키-값 페어입니다. `ResourceType`의 값은 `fleet`이어야 합니다. 그렇지 않으면 집합 요청이 실패합니다. 시작 시 인스턴스에 태그를 지정하려면 [시작 템플릿 \(p. 383\)](#)에서 태그를 지정합니다. 시작 후 태그 지정에 대한 자세한 내용은 [리소스에 태그 지정 \(p. 1144\)](#) 단원을 참조하십시오.

EC2 집합 생성

EC2 집합을 생성할 때 인스턴스 유형, 가용 영역, 지불하려는 최고 가격 등 시작할 인스턴스에 대한 정보를 포함하는 시작 템플릿을 지정해야 합니다.

시작 템플릿을 재정의하는 여러 시작 사양이 포함된 EC2 집합을 생성할 수 있습니다. 시작 사양은 인스턴스 유형, 가용 영역, 서브넷, 최고 가격에 따라 달라질 수 있으며 다른 가중치 용량을 포함할 수 있습니다.

EC2 집합을 생성할 때 JSON 파일을 사용하여 시작할 인스턴스에 대한 정보를 지정하십시오. 자세한 내용은 [EC2 집합 JSON 구성 파일 참조 \(p. 416\)](#) 단원을 참조하십시오.

EC2 집합은 AWS CLI API로만 생성할 수 있습니다.

EC2 집합을 생성하려면(AWS CLI)

- [create-fleet](#)(AWS CLI) 명령을 사용하여 EC2 집합을 생성합니다.

```
aws ec2 create-fleet \
--cli-input-json file://file_name.json
```

구성 파일에 대한 예시는 [EC2 집합 구성의 예 \(p. 426\)](#) 단원을 참조하십시오.

다음은 request 또는 maintain 유형의 플릿에 대한 예제 출력입니다.

```
{
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

다음은 목표 용량을 시작한 instant 유형의 플릿에 대한 예제 출력입니다.

```
{
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
    "Errors": [],
    "Instances": [
        {
            "LaunchTemplateAndOverrides": {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
                    "Version": "1"
                },
                "Overrides": {
                    "InstanceType": "c5.large",
                    "AvailabilityZone": "us-east-1a"
                }
            },
            "Lifecycle": "on-demand",
            "InstanceIds": [
                "i-1234567890abcdef0",
                "i-9876543210abcdef9"
            ],
            "InstanceType": "c5.large",
            "Platform": null
        },
        {
            "LaunchTemplateAndOverrides": {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
                    "Version": "1"
                },
                "Overrides": {
                    "InstanceType": "c4.large",
                    "AvailabilityZone": "us-east-1a"
                }
            },
            "Lifecycle": "on-demand",
            "InstanceIds": [
                "i-5678901234abcdef0",
                "i-5432109876abcdef9"
            ]
        }
    ]
}
```

```
],
  "InstanceType": "c4.large",
  "Platform": null
},
]
}
```

다음은 시작되지 않은 인스턴스에 대한 오류와 함께 목표 용량의 일부를 시작한 instant 유형의 플릿에 대한 예제 출력입니다.

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
  "Errors": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c4.xlarge",
          "AvailabilityZone": "us-east-1a",
        }
      },
      "Lifecycle": "on-demand",
      "ErrorCode": "InsufficientInstanceCapacity",
      "ErrorMessage": "",
      "InstanceType": "c4.xlarge",
      "Platform": null
    },
  ],
  "Instances": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c5.large",
          "AvailabilityZone": "us-east-1a"
        }
      },
      "Lifecycle": "on-demand",
      "InstanceIds": [
        "i-1234567890abcdef0",
        "i-9876543210abcdef9"
      ],
      "InstanceType": "c5.large",
      "Platform": null
    },
  ]
}
```

어떤 인스턴스도 시작하지 않은 instant 유형의 플릿에 대한 예제 출력입니다.

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
  "Errors": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        }
      }
    }
  ]
}
```

```
        "Version": "1"
    },
    "Overrides": {
        "InstanceType": "c4.xlarge",
        "AvailabilityZone": "us-east-1a",
    }
},
"Lifecycle": "on-demand",
"ErrorCode": "InsufficientCapacity",
"ErrorMessage": "",
"InstanceType": "c4.xlarge",
"Platform": null
},
{
    "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
            "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
            "Version": "1"
        },
        "Overrides": {
            "InstanceType": "c5.large",
            "AvailabilityZone": "us-east-1a",
        }
    },
    "Lifecycle": "on-demand",
    "ErrorCode": "InsufficientCapacity",
    "ErrorMessage": "",
    "InstanceType": "c5.large",
    "Platform": null
    },
],
"Instances": []
}
```

EC2 집합 태그 지정

EC2 집합 요청을 쉽게 분류하고 관리할 수 있도록 사용자 지정 메타데이터로 이 요청에 태그를 지정할 수 있습니다. EC2 집합 요청을 만들 때 또는 만든 후 요청에 태그를 지정할 수 있습니다.

플릿 요청에 태그를 지정할 때 플릿에서 시작한 인스턴스 및 볼륨에는 태그가 자동으로 지정되지 않습니다. 플릿에서 시작한 인스턴스 및 볼륨에 명시적으로 태그를 지정해야 합니다. 플릿 요청에만, 플릿에서 시작한 인스턴스에만, 플릿에서 시작한 인스턴스에 연결된 볼륨에만, 또는 세 가지 모두에 태그를 할당하도록 선택할 수 있습니다.

Note

instant 플릿 유형의 경우 온디맨드 인스턴스 및 스팟 인스턴스에 연결된 볼륨에 태그를 지정할 수 있습니다. request 또는 maintain 플릿 유형의 경우 온디맨드 인스턴스에 연결된 볼륨에만 태그를 지정할 수 있습니다.

태그 작동 방식에 대한 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

사전 조건

IAM 사용자에게 리소스에 태그를 지정할 수 있는 권한을 부여합니다. 자세한 내용은 [예제: 리소스에 태그 지정 \(p. 874\)](#) 단원을 참조하십시오.

IAM 사용자에게 리소스에 태그를 지정할 수 있는 권한을 부여하려면

다음을 포함하는 IAM 정책을 만듭니다.

- `ec2:CreateTags` 작업. IAM 사용자에게 태그 생성 권한이 부여됩니다.

- `ec2:CreateFleet` 작업. IAM 사용자에게 EC2 집합 요청 생성 권한이 부여됩니다.
- `Resource`의 경우 "*"를 지정하는 것이 좋습니다. 이를 통해 사용자는 모든 리소스 유형에 태그를 지정할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "TagEC2FleetRequest",  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateTags",  
                "ec2:CreateFleet"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

Important

`create-fleet` 리소스에 대한 리소스 수준 권한은 현재 지원되지 않습니다. `create-fleet`을 리소스로 지정하면 플릿에 태그를 지정하려고 할 때 승인되지 않은 예외가 발생합니다. 다음 예에서는 정책을 설정하지 않는 방법을 보여 줍니다.

```
{  
    "Effect": "Allow",  
    "Action": [  
        "ec2:CreateTags",  
        "ec2:CreateFleet"  
    ],  
    "Resource": "arn:aws:ec2:us-east-1:111122223333:create-fleet/*"  
}
```

새 EC2 집합 요청에 태그를 지정하려면

EC2 집합 요청 생성 시 요청에 태그를 지정하려면 플릿을 생성하는 데 사용되는 [JSON 파일 \(p. 414\)](#)에 키-값 페어를 지정하십시오. `ResourceType`의 값은 `fleet`이어야 합니다. 다른 값을 지정하면 집합 요청이 실패합니다.

EC2 집합에서 시작한 인스턴스 및 볼륨에 태그를 지정하려면

플릿이 시작하는 해당 인스턴스 및 볼륨에 태그를 지정하려면 EC2 집합 요청에서 참조되는 [시작 템플릿 \(p. 383\)](#)에서 태그를 지정하십시오.

Note

`request` 또는 `maintain` 플릿 유형에서 시작한 스팟 인스턴스에 연결된 볼륨에는 태그를 지정할 수 없습니다.

기존 EC2 집합 요청, 인스턴스 및 볼륨에 태그를 지정하려면(AWS CLI)

`create-tags` 명령을 사용하여 기존 리소스에 태그를 지정합니다.

```
aws ec2 create-tags \  
    --resources fleet-12a34b55-67cd-8ef9-  
ba9b-9208dEXAMPLE i-1234567890abcdef0 vol-1234567890EXAMPLE \  
    --tags Key=purpose,Value=test
```

EC2 집합 모니터링

EC2 집합은 가용 용량이 있을 때 온디맨드 인스턴스를 시작하며, 최고 가격이 스팟 가격을 초과하고 가용 용량이 있을 때 스팟 인스턴스를 시작합니다. 온디맨드 인스턴스는 사용자가 종료할 때까지 실행되고 스팟 인스턴스는 중단되거나 사용자가 종료할 때까지 실행됩니다.

반환되는 실행 중 인스턴스 목록은 주기적으로 새로 고쳐지며 최신 상태가 아닐 수도 있습니다.

EC2 집합을 모니터링하려면(AWS CLI)

[describe-fleets](#) 명령을 사용하여 EC2 집합을 설명합니다.

```
aws ec2 describe-fleets
```

다음은 예제 출력입니다.

```
{  
    "Fleets": [  
        {  
            "Type": "maintain",  
            "FulfilledCapacity": 2.0,  
            "LaunchTemplateConfigs": [  
                {  
                    "LaunchTemplateSpecification": {  
                        "Version": "2",  
                        "LaunchTemplateId": "lt-07b3bc7625cdab851"  
                    }  
                }  
            ],  
            "TerminateInstancesWithExpiration": false,  
            "TargetCapacitySpecification": {  
                "OnDemandTargetCapacity": 0,  
                "SpotTargetCapacity": 2,  
                "TotalTargetCapacity": 2,  
                "DefaultTargetCapacityType": "spot"  
            },  
            "FulfilledOnDemandCapacity": 0.0,  
            "ActivityStatus": "fulfilled",  
            "FleetId": "fleet-76e13e99-01ef-4bd6-ba9b-9208de883e7f",  
            "ReplaceUnhealthyInstances": false,  
            "SpotOptions": {  
                "InstanceInterruptionBehavior": "terminate",  
                "InstancePoolsToUseCount": 1,  
                "AllocationStrategy": "lowest-price"  
            },  
            "FleetState": "active",  
            "ExcessCapacityTerminationPolicy": "termination",  
            "CreateTime": "2018-04-10T16:46:03.000Z"  
        }  
    ]  
}
```

[describe-fleet-instances](#) 명령을 사용하여 지정한 EC2 집합의 인스턴스를 설명합니다.

```
aws ec2 describe-fleet-instances \  
  --fleet-id fleet-73fb2ce-aa30-494c-8788-1cee4EXAMPLE
```

```
{  
    "ActiveInstances": [
```

```
{  
    "InstanceId": "i-09cd595998cb3765e",  
    "InstanceHealth": "healthy",  
    "InstanceType": "m4.large",  
    "SpotInstanceRequestId": "sir-86k84j6p"  
},  
{  
    "InstanceId": "i-09cf95167ca219f17",  
    "InstanceHealth": "healthy",  
    "InstanceType": "m4.large",  
    "SpotInstanceRequestId": "sir-dvxi7fsm"  
}  
,  
"FleetId": "fleet-73fb2ce-aa30-494c-8788-1cee4EXAMPLE"  
}
```

[describe-fleet-history](#) 명령을 사용하여 지정한 시간 동안 지정한 EC2 집합의 기록을 설명합니다.

```
aws ec2 describe-fleet-history --fleet-request-id fleet-73fb2ce-aa30-494c-8788-1cee4EXAMPLE --start-time 2018-04-10T00:00:00Z
```

```
{  
    "HistoryRecords": [],  
    "FleetId": "fleet-73fb2ce-aa30-494c-8788-1cee4EXAMPLE",  
    "LastEvaluatedTime": "1970-01-01T00:00:00.000Z",  
    "StartTime": "2018-04-09T23:53:20.000Z"  
}
```

EC2 집합 수정

submitted 또는 active 상태인 EC2 집합을 수정할 수 있습니다. 플릿을 수정할 때 플릿은 modifying 상태가 됩니다.

유형이 maintain인 EC2 집합만 수정할 수 있습니다. 유형이 request 또는 instant인 EC2 집합은 수정 할 수 없습니다.

EC2 집합의 다음 파라미터를 수정할 수 있습니다.

- target-capacity-specification – TotalTargetCapacity, OnDemandTargetCapacity 및 SpotTargetCapacity의 목표 용량을 늘리거나 줄입니다.
- excess-capacity-termination-policy – EC2 집합의 총 목표 용량이 플릿의 현재 크기보다 작아지 면 실행 중인 인스턴스를 종료할지 여부입니다. 유효한 값은 no-termination 및 termination입니다.

목표 용량을 늘리면 EC2 집합이 DefaultTargetCapacityType에 지정한 인스턴스 구입 옵션(온디맨드 인스턴스 또는 스팟 인스턴스)에 따라 추가 인스턴스를 시작합니다.

DefaultTargetCapacityType이 spot이면 EC2 집합이 할당 전략에 따라 추가 스팟 인스턴스를 시작합니다. 할당 전략이 lowest-price이면 플릿이 요청에 있는 최저 가격의 스팟 인스턴스 풀에서 인스턴스를 시작합니다. 할당 전략이 diversified이면 플릿이 요청에 있는 풀에 인스턴스를 배포합니다.

목표 용량을 줄이면 EC2 집합이 새 목표 용량을 초과하는 모든 열린 요청을 삭제합니다. 플릿 크기가 새 목표 용량에 도달할 때까지 플릿이 인스턴스를 종료하도록 요청할 수 있습니다. 할당 전략이 lowest-price이면 플릿이 단위당 최고 가격의 인스턴스를 종료합니다. 할당 전략이 diversified이면 플릿이 풀 전체의 인스턴스를 종료합니다. 또는 EC2 집합이 플릿을 현재 크기로 유지하되 중단된 스팟 인스턴스나 사용자가 수동으로 종료하는 인스턴스를 교체하지 않도록 요청할 수 있습니다.

목표 용량이 줄어 EC2 집합이 스팟 인스턴스를 종료할 때 인스턴스는 스팟 인스턴스 중단 공지를 받습니다.

EC2 집합을 수정하려면(AWS CLI)

`modify-fleet` 명령을 사용하여 지정된 EC2 집합의 목표 용량을 업데이트합니다.

```
aws ec2 modify-fleet \
--fleet-id fleet-73fdbd2ce-aa30-494c-8788-1cee4EXAMPLE \
--target-capacity-specification TotalTargetCapacity=20
```

목표 용량을 줄이고 플릿은 현재 크기로 유지하려는 경우 다음과 같이 이전의 명령을 수정할 수 있습니다.

```
aws ec2 modify-fleet \
--fleet-id fleet-73fdbd2ce-aa30-494c-8788-1cee4EXAMPLE \
--target-capacity-specification TotalTargetCapacity=10 \
--excess-capacity-termination-policy no-termination
```

EC2 집합 삭제

EC2 집합이 더 이상 필요 없으면 삭제할 수 있습니다. 플릿을 삭제한 후에는 새로운 인스턴스를 시작하지 않습니다.

EC2 집합이 인스턴스를 종료할지 여부를 지정해야 합니다. 플릿이 삭제되면 인스턴스가 종료되도록 지정할 경우 `deleted_terminating` 상태가 되고, 그렇지 않으면 `deleted_running` 상태가 되어 인스턴스가 종단되거나 수동으로 종료될 때까지 계속 실행됩니다.

유형이 `request` 및 `maintain`인 플릿만 삭제할 수 있습니다. `instant` EC2 집합은 삭제할 수 없습니다.

EC2 집합을 삭제하려면(AWS CLI)

`delete-fleets` 명령과 `--terminate-instances` 파라미터를 사용하여 지정된 EC2 집합을 삭제하고 인스턴스를 종료합니다.

```
aws ec2 delete-fleets \
--fleet-ids fleet-73fdbd2ce-aa30-494c-8788-1cee4EXAMPLE \
--terminate-instances
```

다음은 예제 출력입니다.

```
{
    "UnsuccessfulFleetDeletions": [],
    "SuccessfulFleetDeletions": [
        {
            "CurrentFleetState": "deleted_terminating",
            "PreviousFleetState": "active",
            "FleetId": "fleet-73fdbd2ce-aa30-494c-8788-1cee4EXAMPLE"
        }
    ]
}
```

`--no-terminate-instances` 파라미터를 사용해 이전의 명령을 수정하여 인스턴스를 수정하지 않고 지정된 EC2 집합을 삭제할 수 있습니다.

```
aws ec2 delete-fleets \
--fleet-ids fleet-73fdbd2ce-aa30-494c-8788-1cee4EXAMPLE \
--no-terminate-instances
```

다음은 예제 출력입니다.

```
{  
    "UnsuccessfulFleetDeletions": [],  
    "SuccessfulFleetDeletions": [  
        {  
            "CurrentFleetState": "deleted_running",  
            "PreviousFleetState": "active",  
            "FleetId": "fleet-4b8aaae8-dfb5-436d-a4c6-3dafa4c6b7dcEXAMPLE"  
        }  
    ]  
}
```

삭제가 실패하는 이유

EC2 집합이 삭제에 실패하는 경우 UnsuccessfulFleetDeletions에서 EC2 집합의 ID, 오류 코드 및 오류 메시지를 반환합니다. 오류 코드는 fleetIdDoesNotExist, fleetIdMalformed, fleetNotInDeletableState 및 unexpectedError입니다.

EC2 집합 구성의 예

다음 예제에서는 [create-fleet](#) 명령에 사용하여 EC2 집합을 생성할 수 있는 시작 구성을 보여줍니다. [create-fleet](#) 파라미터에 대한 자세한 내용은 [EC2 집합 JSON 구성 파일 참조 \(p. 416\)](#) 단원을 참고하십시오.

예

- 예 1: 스팟 인스턴스를 기본 구입 옵션으로 시작 (p. 426)
- 예 2: 온디맨드 인스턴스를 기본 구입 옵션으로 시작 (p. 427)
- 예 3: 온디맨드 인스턴스를 기본 용량으로 시작 (p. 427)
- 예 4: lowest-price 할당 전략을 사용하여 스팟 인스턴스 시작 (p. 427)
- 예 5: 용량 예약 및 prioritized 할당 전략을 사용하여 온디맨드 인스턴스 시작 (p. 428)
- 예 6: 총 목표 용량이 미사용 용량 예약 수보다 많은 경우 용량 예약 및 prioritized 할당 전략을 사용하여 온디맨드 인스턴스 시작 (p. 430)
- 예 7: 용량 예약 및 lowest-price 할당 전략을 사용하여 온디맨드 인스턴스 시작 (p. 432)
- 예 8: 총 목표 용량이 미사용 용량 예약 수보다 많은 경우 용량 예약 및 lowest-price 할당 전략을 사용하여 온디맨드 인스턴스 시작 (p. 434)

예 1: 스팟 인스턴스를 기본 구입 옵션으로 시작

다음 예제에서는 EC2 집합에 필요한 최소한의 파라미터, 즉 시작 템플릿, 목표 용량 및 기본 구매 옵션을 지정합니다. 시작 템플릿은 시작 템플릿 ID와 버전 번호로 식별됩니다. 템플릿의 목표 용량은 인스턴스 2개이고 기본 구입 옵션은 spot이므로 템플릿이 스팟 인스턴스 2개를 시작합니다.

```
{  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-0e8c754449b27161c",  
                "Version": "1"  
            }  
        }  
    ],  
    "TargetCapacitySpecification": {  
        "TotalTargetCapacity": 2,  
        "DefaultTargetCapacityType": "spot"  
    }  
}
```

예 2: 온디맨드 인스턴스를 기본 구입 옵션으로 시작

다음 예제에서는 EC2 집합에 필요한 최소한의 파라미터, 즉 시작 템플릿, 목표 용량 및 기본 구매 옵션을 지정합니다. 시작 템플릿은 시작 템플릿 ID와 버전 번호로 식별됩니다. 템플릿의 목표 용량은 인스턴스 2개이고 기본 구입 옵션은 on-demand이므로 템플릿이 온디맨드 인스턴스 2개를 시작합니다.

```
{  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-0e8c754449b27161c",  
                "Version": "1"  
            }  
        },  
        {  
            "TargetCapacitySpecification": {  
                "TotalTargetCapacity": 2,  
                "DefaultTargetCapacityType": "on-demand"  
            }  
        }  
    ]  
}
```

예 3: 온디맨드 인스턴스를 기본 용량으로 시작

다음 예제에서는 총 목표 용량인 인스턴스 2개를 템플릿에 지정하고 목표 용량은 온디맨드 인스턴스 1개로 지정합니다. 기본 구매 옵션은 spot입니다. 지정한 대로 템플릿은 온디맨드 인스턴스 1개를 시작하지만 총 목표 용량을 충족하려면 인스턴스를 하나 더 시작해야 합니다. 차이에 대한 구매 옵션이 TotalTargetCapacity - OnDemandTargetCapacity = DefaultTargetCapacityType으로 계산되므로 템플릿에서 스팟 인스턴스 1개를 시작합니다.

```
{  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-0e8c754449b27161c",  
                "Version": "1"  
            }  
        },  
        {  
            "TargetCapacitySpecification": {  
                "TotalTargetCapacity": 2,  
                "OnDemandTargetCapacity": 1,  
                "DefaultTargetCapacityType": "spot"  
            }  
        }  
    ]  
}
```

예 4: lowest-price 할당 전략을 사용하여 스팟 인스턴스 시작

스팟 인스턴스의 할당 전략이 지정되어 있지 않으면 기본 할당 전략인 lowest-price가 사용됩니다. 다음 예제에서는 lowest-price 할당 전략을 사용합니다. 시작 템플릿을 재정의하고 서로 인스턴스 유형은 다르지만 가중치 용량과 서브넷이 동일한 시작 사양 3개가 있습니다. 총 목표 용량은 인스턴스 2개이고 기본 구매 옵션은 spot입니다. EC2 집합은 최저 가격이 지정된 시작 사양의 인스턴스 유형을 사용하여 스팟 인스턴스 2개를 시작합니다.

```
{  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-0e8c754449b27161c",  
                "Version": "1"  
            }  
        },  
        {  
            "TargetCapacitySpecification": {  
                "TotalTargetCapacity": 2,  
                "OnDemandTargetCapacity": 1,  
                "DefaultTargetCapacityType": "spot"  
            }  
        }  
    ]  
}
```

```
        "Version": "1"
    }
}
"Overrides": [
{
    "InstanceType": "c4.large",
    "WeightedCapacity": 1,
    "SubnetId": "subnet-a4f6c5d3"
},
{
    "InstanceType": "c3.large",
    "WeightedCapacity": 1,
    "SubnetId": "subnet-a4f6c5d3"
},
{
    "InstanceType": "c5.large",
    "WeightedCapacity": 1,
    "SubnetId": "subnet-a4f6c5d3"
}
]
},
],
"TargetCapacitySpecification": {
    "TotalTargetCapacity": 2,
    "DefaultTargetCapacityType": "spot"
}
}
```

예 5: 용량 예약 및 prioritized 할당 전략을 사용하여 온디맨드 인스턴스 시작

용량 예약에 대한 사용 전략을 `use-capacity-reservations-first`로 설정하여 온디맨드 인스턴스를 시작할 때 온디맨드 용량 예약부터 사용하도록 플릿을 구성할 수 있습니다. 그리고 여러 인스턴스 풀에 미사용 용량 예약가 있는 경우, 선택한 온디맨드 할당 전략이 적용됩니다. 이 예의 온디맨드 할당 전략은 `prioritized`입니다.

이 예에서는 사용할 수 있는 미사용 용량 예약가 15개 있습니다. 이것은 이 플릿의 목표 온디맨드 용량인 온디맨드 인스턴스 12개보다 많습니다.

이 계정은 3개의 풀에 다음과 같은 미사용 용량 예약 15개를 가지고 있습니다. 각 풀의 용량 예약 수는 `AvailableInstanceCount`로 표시됩니다.

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "c4.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "c3.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-333",
```

```
"InstanceType": "c5.large",
"InstancePlatform": "Linux/UNIX",
"AvailabilityZone": "us-east-1a",
"AvailableInstanceCount": 5,
"InstanceMatchCriteria": "open",
"State": "active"
}
```

다음 플릿 구성에서는 이 예제와 관련된 구성만 보여 줍니다. 온디맨드 할당 전략은 prioritized이고, 용량 예약의 사용 전략은 use-capacity-reservations-first입니다. 총 목표 용량은 12이고, 기본 목표 용량 유형은 on-demand입니다.

Note

플릿 유형은 instant여야 합니다. 다른 플릿 유형에는 용량 예약 가 지원되지 않습니다.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-1234567890abcdefg",
                "Version": "1"
            }
        },
        "Overrides": [
            {
                "InstanceType": "c4.large",
                "AvailabilityZone": "us-east-1a",
                "WeightedCapacity": 1,
                "Priority": 1.0
            },
            {
                "InstanceType": "c3.large",
                "AvailabilityZone": "us-east-1a",
                "WeightedCapacity": 1,
                "Priority": 2.0
            },
            {
                "InstanceType": "c5.large",
                "AvailabilityZone": "us-east-1a",
                "WeightedCapacity": 1,
                "Priority": 3.0
            }
        ]
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 12,
        "DefaultTargetCapacityType": "on-demand"
    },
    "OnDemandOptions": {
        "AllocationStrategy": "prioritized"
    },
    "Type": "instant",
}
```

이전 구성을 사용하여 instant 플릿을 생성하면 목표 용량을 충족하기 위해 다음 12개의 인스턴스가 시작됩니다.

- us-east-1a의 c4.large 온디맨드 인스턴스 5개 – us-east-1a의 c4.large가 첫 번째 우선 순위이고 사용 가능한 미사용 c4.large 용량 예약은 5개입니다.

- us-east-1a의 c3.large 온디맨드 인스턴스 5개 – us-east-1a의 c3.large가 두 번째 우선 순위이며 사용 가능한 미사용 c3.large 용량 예약 5개
- us-east-1a의 c5.large 온디맨드 인스턴스 2개 – us-east-1a의 c5.large가 세 번째 우선 순위이고 사용 가능한 미사용 c5.large 용량 예약은 5개, 그중 2개만 있으면 목표 용량 충족 가능

플릿이 시작된 후, [describe-capacity-reservations](#)를 실행하여 미사용 용량 예약이 몇 개나 남아 있는지 확인할 수 있습니다. 이 예에서는 c4.large 및 c3.large 용량 예약이 모두 사용되었고 c5.large 예약 3개는 미사용 상태로 남아 있음을 보여주는 다음과 같은 응답이 나타납니다.

```
{  
    "CapacityReservationId": "cr-111",  
    "InstanceType": "c4.large",  
    "AvailableInstanceCount": 0  
}  
  
{  
    "CapacityReservationId": "cr-222",  
    "InstanceType": "c3.large",  
    "AvailableInstanceCount": 0  
}  
  
{  
    "CapacityReservationId": "cr-333",  
    "InstanceType": "c5.large",  
    "AvailableInstanceCount": 3  
}
```

예 6: 총 목표 용량이 미사용 용량 예약 수보다 많은 경우 용량 예약 및 prioritized 할당 전략을 사용하여 온디맨드 인스턴스 시작

용량 예약에 대한 사용 전략을 `use-capacity-reservations-first`로 설정하여 온디맨드 인스턴스를 시작할 때 온디맨드 용량 예약부터 사용하도록 플릿을 구성할 수 있습니다. 미사용 용량 예약 수가 온디맨드 목표 용량보다 적으면 선택한 온디맨드 할당 전략에 따라 나머지 온디맨드 목표 용량이 시작됩니다. 이 예의 온디맨드 할당 전략은 `prioritized`입니다.

이 예에서는 사용할 수 있는 미사용 용량 예약이 15개 있습니다. 이는 플릿의 온디맨드 목표 용량인 온디맨드 인스턴스 16개보다 적습니다.

이 계정은 3개의 풀에 다음과 같은 미사용 용량 예약 15개를 가지고 있습니다. 각 풀의 용량 예약 수는 `AvailableInstanceCount`로 표시됩니다.

```
{  
    "CapacityReservationId": "cr-111",  
    "InstanceType": "c4.large",  
    "InstancePlatform": "Linux/UNIX",  
    "AvailabilityZone": "us-east-1a",  
    "AvailableInstanceCount": 5,  
    "InstanceMatchCriteria": "open",  
    "State": "active"  
}  
  
{  
    "CapacityReservationId": "cr-111",  
    "InstanceType": "c3.large",  
    "InstancePlatform": "Linux/UNIX",  
    "AvailabilityZone": "us-east-1a",  
    "AvailableInstanceCount": 5,  
    "InstanceMatchCriteria": "open",  
    "State": "active"  
}
```

```
{  
    "CapacityReservationId": "cr-111",  
    "InstanceType": "c5.large",  
    "InstancePlatform": "Linux/UNIX",  
    "AvailabilityZone": "us-east-1a",  
    "AvailableInstanceCount": 5,  
    "InstanceMatchCriteria": "open",  
    "State": "active"  
}
```

다음 플릿 구성에서는 이 예제와 관련된 구성만 보여 줍니다. 온디맨드 할당 전략은 prioritized이고, 용량 예약의 사용 전략은 use-capacity-reservations-first입니다. 총 목표 용량은 16이고 기본 목표 용량 유형은 on-demand입니다.

Note

플릿 유형은 instant여야 합니다. 다른 플릿 유형에는 용량 예약 가 지원되지 않습니다.

```
{  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-0e8c754449b27161c",  
                "Version": "1"  
            },  
            "Overrides": [  
                {  
                    "InstanceType": "c4.large",  
                    "AvailabilityZone": "us-east-1a",  
                    "WeightedCapacity": 1,  
                    "Priority": 1.0  
                },  
                {  
                    "InstanceType": "c3.large",  
                    "AvailabilityZone": "us-east-1a",  
                    "WeightedCapacity": 1,  
                    "Priority": 2.0  
                },  
                {  
                    "InstanceType": "c5.large",  
                    "AvailabilityZone": "us-east-1a",  
                    "WeightedCapacity": 1,  
                    "Priority": 3.0  
                }  
            ]  
        }  
    ],  
    "TargetCapacitySpecification": {  
        "TotalTargetCapacity": 16,  
        "DefaultTargetCapacityType": "on-demand"  
    },  
    "OnDemandOptions": {  
        "AllocationStrategy": "prioritized"  
        "CapacityReservationOptions": {  
            "UsageStrategy": "use-capacity-reservations-first"  
        }  
    },  
    "Type": "instant",  
}
```

이전 구성을 사용하여 instant 플릿을 생성한 후, 목표 용량을 충족하기 위해 다음 16개의 인스턴스가 시작됩니다.

- us-east-1a의 c4.large 온디맨드 인스턴스 6개 – us-east-1a의 c4.large가 첫 번째 우선 순위이고 사용 가능한 미사용 c4.large 용량 예약은 5개입니다. 먼저 용량 예약을 사용하여 온디맨드 인스턴스 5개를 시작하고, 온디맨드 할당 전략(이 예에서는 prioritized)에 따라 온디맨드 인스턴스를 추가로 시작합니다.
- us-east-1a의 c3.large 온디맨드 인스턴스 5개 – us-east-1a의 c3.large가 두 번째 우선 순위이며 사용 가능한 미사용 c3.large 용량 예약 5개
- us-east-1a의 c5.large 온디맨드 인스턴스 5개 – us-east-1a의 c5.large가 세 번째 우선 순위이며 사용 가능한 미사용 c5.large 용량 예약 5개

플릿이 시작된 후, [describe-capacity-reservations](#)를 실행하여 미사용 용량 예약가 몇 개나 남아 있는지 확인할 수 있습니다. 이 예에서는 모든 풀의 용량 예약가 모두 사용되었음을 보여 주는 다음 응답이 나타납니다.

```
{  
    "CapacityReservationId": "cr-111",  
    "InstanceType": "c4.large",  
    "AvailableInstanceCount": 0  
}  
  
{  
    "CapacityReservationId": "cr-222",  
    "InstanceType": "c3.large",  
    "AvailableInstanceCount": 0  
}  
  
{  
    "CapacityReservationId": "cr-333",  
    "InstanceType": "c5.large",  
    "AvailableInstanceCount": 0  
}
```

예 7: 용량 예약 및 lowest-price 할당 전략을 사용하여 온디맨드 인스턴스 시작

용량 예약에 대한 사용 전략을 `use-capacity-reservations-first`로 설정하여 온디맨드 인스턴스를 시작할 때 온디맨드 용량 예약부터 사용하도록 플릿을 구성할 수 있습니다. 그리고 여러 인스턴스 풀에 미사용 용량 예약가 있는 경우, 선택한 온디맨드 할당 전략이 적용됩니다. 이 예의 온디맨드 할당 전략은 `lowest-price`입니다.

이 예에서는 사용할 수 있는 미사용 용량 예약가 15개 있습니다. 이것은 이 플릿의 목표 온디맨드 용량인 온디맨드 인스턴스 12개보다 많습니다.

이 계정은 3개의 풀에 다음과 같은 미사용 용량 예약 15개를 가지고 있습니다. 각 풀의 용량 예약 수는 `AvailableInstanceCount`로 표시됩니다.

```
{  
    "CapacityReservationId": "cr-111",  
    "InstanceType": "m5.large",  
    "InstancePlatform": "Linux/UNIX",  
    "AvailabilityZone": "us-east-1a",  
    "AvailableInstanceCount": 5,  
    "InstanceMatchCriteria": "open",  
    "State": "active"  
}  
  
{  
    "CapacityReservationId": "cr-222",  
    "InstanceType": "m4.xlarge",  
    "InstancePlatform": "Linux/UNIX",  
    "AvailabilityZone": "us-east-1a",  
    "AvailableInstanceCount": 5,  
    "InstanceMatchCriteria": "open",  
    "State": "active"  
}
```

```
{  
    "CapacityReservationId": "cr-333",  
    "InstanceType": "m4.2xlarge",  
    "InstancePlatform": "Linux/UNIX",  
    "AvailabilityZone": "us-east-1a",  
    "AvailableInstanceCount": 5,  
    "InstanceMatchCriteria": "open",  
    "State": "active"  
}
```

다음 플릿 구성에서는 이 예제와 관련된 구성만 보여 줍니다. 온디맨드 할당 전략은 `lowest-price`이고, 용량 예약의 사용 전략은 `use-capacity-reservations-first`입니다. 총 목표 용량은 12이고, 기본 목표 용량 유형은 `on-demand`입니다.

이 예에서 온디맨드 인스턴스 가격은 다음과 같습니다.

- m5.large – 시간당 0.096 USD
- m4.xlarge – 시간당 0.20 USD
- m4.2xlarge – 시간당 0.40 USD

Note

플릿 유형은 `instant`여야 합니다. 다른 플릿 유형에는 용량 예약 가 지원되지 않습니다.

```
{  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-0e8c754449b27161c",  
                "Version": "1"  
            }  
            "Overrides": [  
                {  
                    "InstanceType": "m5.large",  
                    "AvailabilityZone": "us-east-1a",  
                    "WeightedCapacity": 1  
                },  
                {  
                    "InstanceType": "m4.xlarge",  
                    "AvailabilityZone": "us-east-1a",  
                    "WeightedCapacity": 1  
                },  
                {  
                    "InstanceType": "m4.2xlarge",  
                    "AvailabilityZone": "us-east-1a",  
                    "WeightedCapacity": 1  
                }  
            ]  
        }  
    ],  
    "TargetCapacitySpecification": {  
        "TotalTargetCapacity": 12,  
        "DefaultTargetCapacityType": "on-demand"  
    },  
    "OnDemandOptions": {  
        "AllocationStrategy": "lowest-price"  
        "CapacityReservationOptions": {  
            "UsageStrategy": "use-capacity-reservations-first"  
        }  
    },  
},
```

```
    "Type": "instant",  
}
```

이전 구성은 사용하여 instant 플릿을 생성하면 목표 용량을 충족하기 위해 다음 12개의 인스턴스가 시작됩니다.

- us-east-1a의 m5.large 온디맨드 인스턴스 5개 – us-east-1a의 m5.large가 최저 가격이고 사용 가능한 미사용 m5.large 용량 예약 5개.
- us-east-1a의 m4.xlarge 온디맨드 인스턴스 5개 – us-east-1a의 m4.xlarge가 두 번째 최저 가격이며 사용 가능한 미사용 m4.xlarge 용량 예약 5개
- us-east-1a의 m4.2xlarge 온디맨드 인스턴스 2개 – us-east-1a의 m4.2xlarge가 세 번째 최저 가격이고 사용 가능한 미사용 m4.2xlarge 용량 예약은 5개이며, 그중 2개만 있으면 목표 용량 충족 가능

플릿이 시작된 후, [describe-capacity-reservations](#)를 실행하여 미사용 용량 예약가 몇 개나 남아 있는지 확인할 수 있습니다. 이 예에서는 m5.large 및 m4.xlarge 용량 예약이 모두 사용되었고 m4.2xlarge 용량 예약 3개는 사용되지 않은 상태로 남아 있음을 보여주는 다음 응답이 나타납니다.

```
{  
    "CapacityReservationId": "cr-111",  
    "InstanceType": "m5.large",  
    "AvailableInstanceCount": 0  
}  
  
{  
    "CapacityReservationId": "cr-222",  
    "InstanceType": "m4.xlarge",  
    "AvailableInstanceCount": 0  
}  
  
{  
    "CapacityReservationId": "cr-333",  
    "InstanceType": "m4.2xlarge",  
    "AvailableInstanceCount": 3  
}
```

예 8: 총 목표 용량이 미사용 용량 예약 수보다 많은 경우 용량 예약 및 lowest-price 할당 전략을 사용하여 온디맨드 인스턴스 시작

용량 예약에 대한 사용 전략을 `use-capacity-reservations-first`로 설정하여 온디맨드 인스턴스를 시작할 때 온디맨드 용량 예약부터 사용하도록 플릿을 구성할 수 있습니다. 미사용 용량 예약 수가 온디맨드 목표 용량보다 적으면 선택한 온디맨드 할당 전략에 따라 나머지 온디맨드 목표 용량이 시작됩니다. 이 예의 온디맨드 할당 전략은 `lowest-price`입니다.

이 예에서는 사용할 수 있는 미사용 용량 예약가 15개 있습니다. 이는 플릿의 온디맨드 목표 용량인 온디맨드 인스턴스 16개보다 적습니다.

이 계정은 3개의 풀에 다음과 같은 미사용 용량 예약 15개를 가지고 있습니다. 각 풀의 용량 예약 수는 `AvailableInstanceCount`로 표시됩니다.

```
{  
    "CapacityReservationId": "cr-111",  
    "InstanceType": "m5.large",  
    "InstancePlatform": "Linux/UNIX",  
    "AvailabilityZone": "us-east-1a",  
    "AvailableInstanceCount": 5,  
    "InstanceMatchCriteria": "open",  
    "State": "active"  
}
```

```
{  
    "CapacityReservationId": "cr-222",  
    "InstanceType": "m4.xlarge",  
    "InstancePlatform": "Linux/UNIX",  
    "AvailabilityZone": "us-east-1a",  
    "AvailableInstanceCount": 5,  
    "InstanceMatchCriteria": "open",  
    "State": "active"  
}  
  
{  
    "CapacityReservationId": "cr-333",  
    "InstanceType": "m4.2xlarge",  
    "InstancePlatform": "Linux/UNIX",  
    "AvailabilityZone": "us-east-1a",  
    "AvailableInstanceCount": 5,  
    "InstanceMatchCriteria": "open",  
    "State": "active"  
}
```

다음 플릿 구성에서는 이 예제와 관련된 구성만 보여 줍니다. 온디맨드 할당 전략은 `lowest-price`이고, 용량 예약의 사용 전략은 `use-capacity-reservations-first`입니다. 총 목표 용량은 16이고 기본 목표 용량 유형은 `on-demand`입니다.

이 예에서 온디맨드 인스턴스 가격은 다음과 같습니다.

- m5.large – 시간당 0.096 USD
- m4.xlarge – 시간당 0.20 USD
- m4.2xlarge – 시간당 0.40 USD

Note

플릿 유형은 `instant`어야 합니다. 다른 플릿 유형에는 용량 예약 지원되지 않습니다.

```
{  
    "LaunchTemplateConfigs": [  
        {  
            "LaunchTemplateSpecification": {  
                "LaunchTemplateId": "lt-0e8c754449b27161c",  
                "Version": "1"  
            }  
            "Overrides": [  
                {  
                    "InstanceType": "m5.large",  
                    "AvailabilityZone": "us-east-1a",  
                    "WeightedCapacity": 1  
                },  
                {  
                    "InstanceType": "m4.xlarge",  
                    "AvailabilityZone": "us-east-1a",  
                    "WeightedCapacity": 1  
                },  
                {  
                    "InstanceType": "m4.2xlarge",  
                    "AvailabilityZone": "us-east-1a",  
                    "WeightedCapacity": 1  
                }  
            ]  
        }  
    ],  
    "TargetCapacitySpecification": {
```

```
        "TotalTargetCapacity": 16,
        "DefaultTargetCapacityType": "on-demand"
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price"
        "CapacityReservationOptions": {
            "UsageStrategy": "use-capacity-reservations-first"
        }
    },
    "Type": "instant",
}
```

이전 구성은 사용하여 instant 플랫을 생성한 후, 목표 용량을 충족하기 위해 다음 16개의 인스턴스가 시작됩니다.

- us-east-1a의 m5.large 온디맨드 인스턴스 6개 – us-east-1a의 m5.large가 최저 가격이고 사용 가능한 미사용 m5.large 용량 예약 5개. 먼저 용량 예약을 사용하여 온디맨드 인스턴스 5개를 시작하고, 온디맨드 할당 전략(이 예에서는 lowest-price)에 따라 온디맨드 인스턴스를 추가로 시작합니다.
- us-east-1a의 m4.xlarge 온디맨드 인스턴스 5개 – us-east-1a의 m4.xlarge가 두 번째 최저 가격이며 사용 가능한 미사용 m4.xlarge 용량 예약 5개
- us-east-1a의 m4.2xlarge 온디맨드 인스턴스 5개 – us-east-1a의 m4.2xlarge가 세 번째 최저 가격이며 사용 가능한 미사용 m4.2xlarge 용량 예약 5개

플랫이 시작된 후, [describe-capacity-reservations](#)를 실행하여 미사용 용량 예약이 몇 개나 남아 있는지 확인할 수 있습니다. 이 예에서는 모든 풀의 용량 예약이 모두 사용되었음을 보여 주는 다음 응답이 나타납니다.

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "m4.xlarge",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-333",
    "InstanceType": "m4.2xlarge",
    "AvailableInstanceCount": 0
}
```

Windows 인스턴스에 연결

원격 데스크톱을 사용하면 대부분의 Windows Amazon 머신 이미지(AMIs)에서 생성되는 Amazon EC2 인스턴스에 연결할 수 있습니다. 원격 데스크톱은 RDP(Remote Desktop Protocol)를 사용하며 바로 앞에 있는 컴퓨터를 사용하는 것처럼 인스턴스를 연결하여 사용할 수 있도록 해줍니다. 대부분의 Windows 버전과 Mac OS에서 사용할 수 있습니다.

Linux 인스턴스 연결 방법에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux 인스턴스에 연결](#)을 참조하십시오.

내용

- [사전 조건 \(p. 437\)](#)
- [Windows 인스턴스 연결 \(p. 437\)](#)
- [IPv6 주소를 사용하여 Windows 인스턴스에 연결 \(p. 439\)](#)

- 세션 관리자를 사용하여 Windows 인스턴스에 연결 (p. 439)
- Windows 인스턴스로 파일 전송 (p. 440)

사전 조건

- RDP 클라이언트 설치
 - [Windows] Windows에는 기본적으로 RDP 클라이언트가 포함되어 있습니다. 확인하려면 명령 프롬프트 창에 mstsc를 입력합니다. 컴퓨터에서 이 명령이 인식되지 않으면 [Windows 헤더 페이지](#)를 참조하여 Microsoft 원격 데스크톱 앱 다운로드를 검색합니다.
 - [Mac OS X] Mac 앱 스토어에서 Microsoft 원격 데스크톱 앱을 다운로드합니다.
 - [Linux] [Remmina](#)를 사용합니다.
- 인스턴스의 ID 보기.

Amazon EC2 콘솔을 사용하여(인스턴스 ID 열에서) 인스턴스의 ID를 가져올 수 있습니다. 원하는 경우 [describe-instances](#)(AWS CLI) 또는 [Get-EC2Instance](#)(Windows PowerShell용 AWS 도구) 명령을 사용할 수 있습니다.
- 인스턴스의 퍼블릭 DNS 이름을 가져옵니다.

Amazon EC2 콘솔을 사용하여 인스턴스의 퍼블릭 DNS를 가져올 수 있습니다. 퍼블릭 DNS(IPv4) 열을 확인합니다. 이 열이 숨겨져 있는 경우 화면 상단 오른쪽에 있는 열 표시/숨기기 아이콘을 클릭하고 퍼블릭 DNS(Public DNS)(IPv4)를 선택합니다. 원하는 경우 [describe-instances](#)(AWS CLI) 또는 [Get-EC2Instance](#)(Windows PowerShell용 AWS 도구) 명령을 사용할 수 있습니다.
- (IPv6 전용) 인스턴스의 IPv6 주소를 가져옵니다.

인스턴스에 IPv6 주소를 할당했다면 퍼블릭 IPv4 주소나 퍼블릭 IPv4 DNS 호스트 이름 대신 IPv6 주소를 사용하여 인스턴스에 연결할 수도 있습니다. 로컬 컴퓨터에 IPv6 주소가 있고 IPv6를 사용하도록 컴퓨터를 구성해야 합니다. Amazon EC2 콘솔을 사용하여 인스턴스의 IPv6 주소를 가져올 수 있습니다. IPv6 IP 필드를 확인합니다. 원하는 경우 [describe-instances](#)(AWS CLI) 또는 [Get-EC2Instance](#)(Windows PowerShell 용 AWS 도구) 명령을 사용할 수 있습니다. IPv6에 대한 자세한 내용은 [IPv6 주소 \(p. 704\)](#) 단원을 참조하십시오.
- 프라이빗 키 찾기

인스턴스를 시작할 때 지정한 키 페어를 찾기 위해 .pem 파일의 컴퓨터 상 위치에 대한 정규화된 경로를 염습니다. 키 페어를 생성하는 방법에 대한 자세한 내용은 [Amazon EC2를 사용하여 키 페어 생성](#)을 참조하십시오.
- IP 주소에서 인스턴스로의 인바운드 RDP 트래픽 활성화

인스턴스와 연관된 보안 그룹이 IP 주소로부터 들어오는 RDP 트래픽(포트 3389)을 허용하는지 확인하십시오. 기본 보안 그룹은 기본적으로 들어오는 RDP 트래픽을 허용하지 않습니다. 자세한 내용은 [Windows 인스턴스의 인바운드 트래픽 권한 부여 \(p. 898\)](#) 단원을 참조하십시오.
- 최상의 Internet Explorer 사용 환경을 구축하려면 최신 버전을 실행합니다.

Windows 인스턴스 연결

Windows 인스턴스에 연결하려면 최초 관리자 암호를 검색한 다음(아래 2단계 참조) 원격 데스크톱을 사용하여 인스턴스에 연결할 때 이 암호를 지정해야 합니다.

관리자 계정의 이름은 운영 체제의 언어에 따라 다릅니다. 예를 들어 영어는 Administrator, 프랑스어는 Administrateur, 포르투갈어는 Administrador입니다. 자세한 내용은 Microsoft TechNet Wiki의 [Localized Names for Administrator Account in Windows](#)를 참조하십시오.

인스턴스를 도메인에 조인한 경우 AWS Directory Service에서 정의한 도메인 자격 증명을 사용하여 인스턴스에 연결할 수 있습니다. 원격 데스크톱 로그인 화면에서 로컬 컴퓨터 이름과 생성된 암호를 사용하는 대신 관리자의 정규화된 사용자 이름(예: **corp.example.com\Admin**)과 이 계정의 암호를 사용합니다.

Windows Server 운영 체제(OS) 라이선스는 관리 목적으로 두 개의 동시 원격 연결을 허용합니다. Windows 인스턴스 가격에는 Windows Server 라이선스가 포함됩니다. 2개를 초과하는 동시 원격 연결이 필요할 경우, 원격 데스크톱 서비스(RDS) 라이선스를 구매해야 합니다. 제3의 연결을 시도하면 오류가 발생합니다. 자세한 정보는 [Configure the Number of Simultaneous Remote Connections Allowed for a Connection](#)을 참조하십시오.

RDP 클라이언트를 사용하여 Windows 인스턴스에 연결하려면

1. Amazon EC2 콘솔에서 인스턴스를 선택한 다음, [Connect]를 선택합니다.
2. 인스턴스에 연결 대화 상자에서 암호 가져오기를 선택합니다(인스턴스가 시작된 후 몇 분 정도 지나야 암호를 사용할 수 있음).
3. Browse(찾아보기)를 선택하고 인스턴스를 시작할 때 생성한 프라이빗 키 파일을 탐색합니다. 파일을 선택하고 열기를 클릭하여 파일의 전체 내용을 콘텐츠 필드로 복사합니다.
4. [Decrypt Password]를 선택합니다. 콘솔에서는 [Connect To Your Instance] 대화 상자에 해당 인스턴스에 대한 기본 관리자 암호가 표시되어 이전에 표시된 [Get Password]에 대한 링크가 실제 암호로 바뀝니다.
5. 기본 관리자 암호를 기록하거나 클립보드로 복사합니다. 이 암호는 인스턴스에 연결하는 데 필요합니다.
6. [Download Remote Desktop File]을 선택합니다. 브라우저에서 .rdp 파일을 열거나 저장하라는 메시지가 표시됩니다. 어떤 옵션이든 좋습니다. 마쳤으면 [Close]를 선택하여 [Connect To Your Instance] 대화 상자를 닫습니다.
 - .rdp 파일을 연 경우에는 원격 데스크톱 연결 대화 상자가 나타납니다.
 - .rdp 파일을 저장한 경우에는, 다운로드 디렉터리로 이동해 .rdp 파일을 열면 대화 상자가 표시됩니다.
7. 원격 연결 게시자를 알 수 없다는 경고를 받을 수도 있습니다. 계속해서 인스턴스에 연결할 수 있습니다.
8. 관련 메시지가 표시되면 운영 체제 관리자 계정과 이전에 기록하거나 복사한 암호를 사용하여 인스턴스에 로그인합니다. 원격 데스크톱 연결에 관리자 계정이 이미 설정되어 있는 경우에는 다른 계정 사용 옵션을 선택해 사용자 이름과 암호를 수동으로 입력해야 할 수도 있습니다.

Note

때로는 콘텐츠를 복사하고 붙여 넣으면 데이터가 손상될 수 있습니다. 로그인할 때 "Password Failed" 오류가 발생하면 암호를 수동으로 입력해 보십시오.

9. 자체 서명된 인증서의 특성으로 인해, 보안 인증서를 인증할 수 없다는 경고 메시지가 나타날 수도 있습니다. 다음 단계에 따라 원격 컴퓨터의 자격 증명을 확인하거나, 인증서를 신뢰할 경우에는 단순히 [Yes] 또는 [Continue]를 선택하여 계속 진행합니다.
 - a. Windows PC에서 [Remote Desktop Connection]을 사용 중이라면 [View certificate]를 선택합니다. Mac에서 [Microsoft Remote Desktop]을 사용 중이라면 [Show Certificate]를 선택합니다.
 - b. 세부 정보 탭을 선택하고 Windows PC에서는 지문 항목, Mac에서는 SHA1 지문 항목이 나타날 때 까지 아래로 스크롤합니다. 이것은 원격 컴퓨터의 보안 인증서에 대한 고유한 식별자입니다.
 - c. Amazon EC2 콘솔에서 인스턴스를 선택하고 [Actions]를 선택한 다음, [Get System Log]를 선택합니다.
 - d. 시스템 로그 출력에서 RDPCERTIFICATE-THUMBPRINT라는 항목을 확인합니다. 이 값이 인증서의 지문과 일치한다면 원격 컴퓨터의 자격 증명을 확인한 것입니다.
 - e. Windows PC에서 Remote Desktop Connection을 사용 중이라면 [Certificate] 대화 상자로 돌아가서 [OK]를 선택합니다. Mac에서 [Microsoft Remote Desktop]을 사용 중이라면 [Verify Certificate]으로 돌아가서 [Continue]를 선택합니다.
 - f. [Windows] 원격 데스크톱 연결 창에서 예를 선택하여 인스턴스에 연결합니다.
- [Mac OS] 메시지가 표시되면 이전에 기록했거나 복사한 기본 관리자 계정과 기본 관리자 암호를 사용하여 로그인합니다. 로그인 화면을 보려면 스페이스를 전환해야 할 수도 있습니다. 스페이스에 대한 자세한 정보는 support.apple.com/en-us/HT204100을 참조하십시오.
- g. 인스턴스에 연결을 시도하는 동안 오류가 발생한 경우 [원격 데스크톱으로 원격 컴퓨터에 연결할 수 없음 \(p. 1181\)](#) 단원을 참조하십시오.

연결한 후에는 다음 작업을 수행하는 것이 좋습니다.

- 기본값으로 제공된 관리자 암호를 변경합니다. 다른 Windows Server처럼 인스턴스 자체에 로그온한 상태에서 암호를 변경할 수 있습니다.
- 인스턴스에서 관리자 권한을 사용하여 또 다른 사용자 계정을 만듭니다. 관리자 권한이 있는 또 다른 계정은 관리자 암호를 분실했거나 관리자 계정에 문제가 있는 경우에 사용할 수 있는 보호 수단입니다. 사용자 계정은 인스턴스에 원격으로 액세스하기 위한 권한을 보유해야 합니다. Windows 바탕 화면이나 파일 탐색기에서 이 PC 아이콘을 마우스 오른쪽 단추로 클릭하고 속성을 선택하여 시스템 속성을 엽니다. 원격 설정을 선택하고 사용자 선택을 선택하여 사용자를 원격 데스크톱 사용자 그룹에 추가합니다.

IPv6 주소를 사용하여 Windows 인스턴스에 연결

IPv6용 VPC를 활성화하고 IPv6 주소를 Windows 인스턴스에 할당했다면, 퍼블릭 IPv4 주소나 퍼블릭 DNS 호스트 이름 대신 RDP 클라이언트의 IPv6 주소를 사용해 인스턴스에 연결할 수 있습니다. 자세한 내용은 [IPv6 주소 \(p. 704\)](#) 단원을 참조하십시오.

IPv6 주소를 사용하여 Windows 인스턴스에 연결하려면

- Amazon EC2 콘솔에서 인스턴스를 선택한 다음 연결을 선택합니다.
- 인스턴스에 연결 대화 상자에서 암호 가져오기를 선택합니다(인스턴스가 시작된 후 몇 분 정도 지나야 암호를 사용할 수 있음).
- Browse(찾아보기)를 선택하고 인스턴스를 시작할 때 생성한 브라이빗 키 파일을 탐색합니다. 파일을 선택하고 열기를 클릭하여 파일의 전체 내용을 콘텐츠 필드로 복사합니다.
- 암호 해독을 선택합니다.
- 기본 관리자 암호를 복사합니다. 이 암호는 인스턴스에 연결하는 데 필요합니다.
- 컴퓨터에서 RDP 클라이언트를 엽니다.
- [Windows] Windows 컴퓨터의 RDP 클라이언트에 대해 옵션 표시를 선택하고 다음을 수행합니다.
 - 컴퓨터에 Windows 인스턴스의 IPv6 주소(예: 2001:db8:1234:1a00:9691:9503:25ad:1761)를 입력합니다.
 - 사용자 이름에 관리자를 입력합니다.
 - 연결을 선택합니다.

[Mac OS X] Microsoft 원격 데스크톱 앱에서 새로 만들기를 선택하고 다음을 수행합니다.

- PC 이름에 Windows 인스턴스의 IPv6 주소(예: 2001:db8:1234:1a00:9691:9503:25ad:1761)를 입력합니다.
- 사용자 이름에 관리자를 입력합니다.
- 대화 상자를 닫습니다. My Desktops(내 데스크톱)에서 연결을 선택하고 시작을 선택합니다.
- 자체 서명된 인증서의 특성으로 인해, 보안 인증서를 인증할 수 없다는 경고 메시지가 나타날 수도 있습니다. 다음 단계에 따라 원격 컴퓨터의 자격 증명을 확인하거나, 인증서를 신뢰할 경우에는 단순히 예 또는 계속을 선택하여 계속 진행합니다.
- 메시지가 나타나면 이전에 기록했거나 복사한 암호를 입력합니다.

세션 관리자를 사용하여 Windows 인스턴스에 연결

세션 관리자는 대화형 원클릭 브라우저 기반 셀이나 AWS CLI를 통해 Amazon EC2 인스턴스를 관리할 수 있는 완전 관리형 AWS 시스템 관리자 기능입니다. 세션 관리자를 사용하여 계정 내 인스턴스와의 세션을 시작할 수 있습니다. 세션이 시작된 후에는 다른 연결 유형과 마찬가지로 Powershell 명령을 실행할 수 있습니다. 세션 관리자에 대한 자세한 내용은 AWS 시스템 관리자 사용자 안내서의 [AWS 시스템 관리자 세션 관리자](#)를 참조하십시오.

세션 관리자를 사용하여 인스턴스에 연결하기 전에 필요한 설정 단계가 완료되었는지 확인합니다. 자세한 내용은 [세션 관리자 시작하기](#)를 참조하십시오.

Amazon EC2 콘솔에서 세션 관리자를 사용하여 Windows 인스턴스에 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택한 다음 연결을 선택합니다.
4. 연결 방법으로 세션 관리자를 선택합니다.
5. [Connect]를 선택합니다.

Note

하나 이상의 시스템 관리자 작업(ssm:[command-name](#))을 수행할 권한이 없다는 오류가 표시되면 Amazon EC2 콘솔에서 세션을 시작할 수 있도록 정책을 업데이트해야 합니다. 자세한 내용과 지침은 AWS 시스템 관리자 사용 설명서의 [세션 관리자용 빠른 시작 기본 IAM 정책](#)을 참조하십시오.

Windows 인스턴스로 파일 전송

다른 Windows 서버를 사용할 때와 똑같이 Windows 인스턴스를 사용할 수 있습니다. 예를 들어, Microsoft 원격 데스크톱 연결 소프트웨어의 로컬 파일 공유 기능을 사용하여 Windows 인스턴스와 로컬 컴퓨터 간에 파일을 전송할 수 있습니다. 이 옵션을 활성화하는 경우 Windows 인스턴스에서 로컬 파일에 액세스할 수 있습니다. 하드 디스크 드라이브, DVD 드라이브, 휴대용 미디어 드라이브 및 매핑된 네트워크 드라이브에 있는 로컬 파일에 액세스할 수 있습니다.

로컬 디바이스와 리소스를 Windows의 원격 세션에서 이용할 수 있게 하려면 원격 세션 드라이브를 로컬 드라이브에 매핑해야 합니다.

원격 세션 드라이브를 로컬 드라이브에 매핑하려면

1. 원격 데스크톱 연결 클라이언트를 엽니다.
2. [Show Options]를 선택합니다.
3. 로컬 리소스 탭을 선택합니다.
4. 로컬 디바이스 및 리소스에서 더 보기...를 선택합니다.
5. 드라이브를 열고 Windows 인스턴스에 매핑할 로컬 드라이브를 선택합니다.
6. 확인을 선택합니다.
7. 연결을 선택하여 Windows 인스턴스에 연결합니다.

Mac 컴퓨터의 원격 세션에서 로컬 디바이스를 사용할 수 있도록 하는 방법에 대한 자세한 내용은 [Mac에서 원격 데스크톱 시작하기](#)를 참조하십시오.

인스턴스 중지 및 시작

Amazon EBS 볼륨을 루트 디바이스로 사용하는 인스턴스를 중지했다가 다시 시작할 수 있습니다. 인스턴스 ID는 유지되지만 [개요 \(p. 441\)](#) 단원의 설명처럼 인스턴스는 변경될 수 있습니다.

인스턴스를 중지하면 인스턴스가 종료됩니다. 중지된 인스턴스에 대해 사용 요금이나 데이터 전송 요금이 부과되지는 않지만 모든 Amazon EBS 볼륨에 대한 스토리지 요금은 부과됩니다. 중지한 인스턴스를 시작할 때 마다 전체 인스턴스 시간 요금이 부과되며, 1시간 내에 여러 번 전환한 경우에도 동일하게 부과됩니다.

인스턴스가 중지되어 있는 동안 해당 루트 볼륨을 다른 볼륨과 마찬가지로 처리하고 수정할 수 있습니다. 예를 들어, 파일 시스템 문제를 복구하거나 소프트웨어를 업데이트할 수 있습니다. 볼륨을 중지된 인스턴스에서 분리하고 실행 중인 인스턴스에 연결하고 변경한 후 실행 중인 인스턴스에서 분리하고 중지된 인스턴스에

다시 연결하면 됩니다. 볼륨을 다시 연결할 때 인스턴스에 대한 블록 디바이스 매핑에 루트 디바이스로 지정된 스토리지 디바이스 이름을 사용해야 합니다.

더 이상 필요 없는 인스턴스는 종료할 수 있습니다. 인스턴스의 상태가 `shutting-down`이나 `terminated`로 변경되는 즉시 해당 인스턴스에 대한 요금 발생이 중지됩니다. 자세한 내용은 [인스턴스 종료 \(p. 454\)](#) 단원을 참조하십시오. 인스턴스를 최대 절전 모드로 전환하려는 경우 [Windows 인스턴스를 최대 절전 모드로 실행 \(p. 443\)](#) 단원을 참조하십시오. 자세한 내용은 [재부팅, 중지, 최대 절전 모드 및 종료의 차이 \(p. 374\)](#) 단원을 참조하십시오.

목차

- [개요 \(p. 441\)](#)
- [인스턴스 중지 시 발생하는 상황 \(p. 442\)](#)
- [인스턴스 중지 및 시작 \(p. 442\)](#)
- [중지된 인스턴스 수정 \(p. 443\)](#)
- [문제 해결 \(p. 443\)](#)

개요

실행 중인 인스턴스를 중지하면 다음과 같이 진행됩니다.

- 인스턴스가 일반적인 종료 과정을 수행하고 실행을 중지하며, 인스턴스의 상태가 `stopping`으로 바뀌었다가 `stopped`로 바집니다.
- 모든 Amazon EBS 볼륨이 인스턴스에 연결된 상태로 유지되고 해당 데이터도 남습니다.
- 호스트 컴퓨터의 RAM이나 인스턴스 스토어 볼륨에 저장된 모든 데이터가 손실됩니다.
- 대부분의 경우 인스턴스가 시작되면 새로운 기본 호스트 컴퓨터로 마이그레이션됩니다(경우에 따라 현재 호스트에 남아 있음).
- 인스턴스를 중지했다가 다시 시작해도 프라이빗 IPv4 주소와 모든 IPv6 주소는 유지됩니다. AWS에서 퍼블릭 IPv4 주소를 해제했다가 인스턴스가 시작되면 새 주소를 할당합니다.
- 인스턴스가 연결된 탄력적 IP 주소를 유지합니다. 중지된 인스턴스와 연결된 모든 탄력적 IP 주소에 대한 요금이 부과됩니다. EC2-Classic을 사용하면 중단 시 탄력적 IP 주소가 인스턴스와 연결 해제됩니다. 자세한 내용은 [EC2-Classic \(p. 802\)](#) 단원을 참조하십시오.
- Windows 인스턴스를 중지 및 시작할 때 EC2Config 서비스가 연결된 Amazon EBS 볼륨의 드라이브 문자를 변경하는 등 인스턴스에 대한 작업을 수행합니다. 이러한 기본값에 대한 자세한 내용과 기본값을 변경하는 방법은 [EC2Config 서비스를 사용한 Windows 인스턴스 구성 \(p. 496\)](#)을 참조하십시오.
- 인스턴스가 Auto Scaling 그룹에 있는 경우, Amazon EC2 Auto Scaling 서비스는 종단된 인스턴스를 비정상으로 간주해 이를 종료하고 대체 인스턴스를 시작합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [Auto Scaling 인스턴스에 대한 상태 점검](#) 단원을 참조하십시오.
- ClassicLink 인스턴스를 중지하면 연결되었던 VPC와의 연결이 해제됩니다. 인스턴스를 시작한 후 VPC에 다시 연결해야 합니다. ClassicLink에 대한 자세한 내용은 [ClassicLink \(p. 809\)](#) 단원을 참조하십시오.

자세한 내용은 [재부팅, 중지, 최대 절전 모드 및 종료의 차이 \(p. 374\)](#) 단원을 참조하십시오.

다음은 인스턴스가 중지되었을 때만 수정할 수 있는 인스턴스의 속성입니다.

- 인스턴스 유형
- 사용자 데이터
- 커널
- RAM 디스크

인스턴스가 실행되고 있을 때 이러한 속성을 수정하려고 하면 Amazon EC2에서 `IncorrectInstanceState` 오류를 반환합니다.

인스턴스 중지 시 발생하는 상황

`stop-instances` 명령을 사용하여 EC2 인스턴스를 중지하면 OS 수준에서 다음 항목이 등록됩니다.

- API 요청은 버튼 누름 이벤트를 게스트로 전송합니다.
- 다양한 시스템 서비스는 버튼 누름 이벤트의 결과로 종지됩니다. `systemd`는 시스템의 정상 종료를 처리합니다. 정상 종료는 하이퍼바이저에서 ACPI 종료 버튼 누름 이벤트에 의해 트리거됩니다.
- ACPI 종료가 시작됩니다.
- 정상 종료 프로세스가 종료되면 인스턴스가 종료됩니다. 구성 가능한 OS 종료 시간은 없습니다.
- 인스턴스 OS가 몇 분 이내에 완전히 종료되지 않으면 하드 종료가 수행됩니다.

인스턴스 중지 및 시작

콘솔이나 명령줄을 사용하여 Amazon EBS 기반 인스턴스를 중지했다가 시작할 수 있습니다.

기본적으로 `shutdown` 또는 `poweroff` 명령을 사용하여 Amazon EBS 지원 인스턴스에서 종료를 시작하면 인스턴스가 종료됩니다. 인스턴스가 종지되지 않고 종료되도록 이 동작을 변경할 수 있습니다. 자세한 내용은 [인스턴스가 개시하는 종료 동작 변경 \(p. 457\)](#) 단원을 참조하십시오.

Important

인스턴스에서 `halt` 명령을 사용해도 종료가 시작되지 않습니다. 인스턴스를 사용하면 인스턴스가 종료되지 않고 대신 CPU를 `HLT`에 배치하면 인스턴스가 계속 실행됩니다.

콘솔을 사용하여 Amazon EBS 기반 인스턴스를 중지하고 시작하려면

1. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
2. [Actions], [Instance State], [Stop]을 차례로 선택합니다. 종지가 비활성화되어 있으면 해당 인스턴스가 이미 종지되었거나 루트 디바이스가 인스턴스 스토어 볼륨인 것입니다.

Warning

인스턴스를 종지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

3. 확인 대화 상자가 나타나면 예, 종지를 선택합니다. 인스턴스가 종지하는 데 몇 분 정도 걸릴 수 있습니다.
4. 인스턴스가 종지되어 있는 동안 특정 인스턴스 속성을 수정할 수 있습니다. 자세한 내용은 [종지된 인스턴스 수정 \(p. 443\)](#) 단원을 참조하십시오.
5. 종지된 인스턴스를 시작하려면 그 인스턴스를 선택하고 작업, 인스턴스 상태, 시작을 선택합니다.
6. 확인 대화 상자가 나타나면 예, 시작을 선택합니다. 인스턴스가 `running` 상태가 되는 데 몇 분 정도 걸릴 수 있습니다.

명령줄을 사용하여 Amazon EBS 기반 인스턴스를 중지하고 시작하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `stop-instances` 및 `start-instances`(AWS CLI)
- `Stop-EC2Instance` 및 `Start-EC2Instance`(Windows PowerShell용 AWS 도구)

중지된 인스턴스 설정

AWS Management 콘솔 또는 명령줄 인터페이스를 사용하여 중지된 인스턴스의 인스턴스 유형, 사용자 데이터 및 EBS 최적화 속성을 변경할 수 있습니다. AWS Management 콘솔을 사용하여 DeleteOnTermination, 커널 또는 RAM 디스크 속성을 수정할 수 없습니다.

인스턴스 속성을 수정하려면

- 인스턴스 유형을 변경하려면 [인스턴스 유형 변경 \(p. 194\)](#)을 참조하십시오.
- 인스턴스의 사용자 데이터를 변경하려면 [인스턴스 사용자 데이터로 작업 \(p. 589\)](#)을 참조하십시오.
- 인스턴스의 EBS-최적화를 설정 또는 해제하려면 [EBS-최적화 수정 \(p. 1062\)](#)을 참조하십시오.
- 인스턴스의 루트 볼륨의 DeleteOnTermination 속성을 변경하려면 다음([실행 중인 인스턴스의 블록 디바이스 매핑 업데이트 \(p. 1114\)](#))을 참조하십시오. 이 속성을 변경하기 위해 인스턴스를 종지할 필요는 없습니다.

명령줄을 사용하여 인스턴스 속성을 수정하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [modify-instance-attribute\(AWS CLI\)](#)
- [Edit-EC2InstanceAttribute\(Windows PowerShell용 AWS 도구\)](#)

문제 해결

중지한 Amazon EBS 기반 인스턴스가 stopping 상태에서 "멈춘" 것으로 나타나는 경우 해당 인스턴스를 강제로 종지할 수 있습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스 종지 문제 해결](#) 단원을 참조하십시오.

Windows 인스턴스를 최대 절전 모드로 실행

인스턴스를 최대 절전 모드로 전환하면 Amazon EC2에서 최대 절전 모드(디스크 일시 종단)를 수행하도록 운영 체제에 신호를 보냅니다. 최대 절전 모드는 인스턴스 메모리(RAM)의 콘텐츠를 Amazon EBS 루트 볼륨에 저장합니다. Amazon EC2는 인스턴스의 Amazon EBS 루트 볼륨 및 연결된 모든 Amazon EBS 데이터 볼륨을 유지합니다. 인스턴스를 시작할 때:

- Amazon EBS 루트 볼륨이 이전 상태로 복원됩니다
- RAM 내용이 다시 로드됩니다
- 이전에 인스턴스에서 실행되었던 프로세스가 재개됩니다.
- 이전에 연결된 데이터 볼륨이 다시 연결되고, 인스턴스는 해당 인스턴스 ID를 유지합니다.

인스턴스에 대해 [최대 절전 모드가 활성화 \(p. 446\)](#)되어 있고 [최대 절전 모드 사전 조건 \(p. 445\)](#)을 충족하는 경우 인스턴스를 최대 절전 모드로 전환할 수 있습니다.

인스턴스 또는 애플리케이션에서 최적의 생산성을 내기 위해 메모리 공간을 부트스트랩 및 빌드하는 데 시간이 오래 걸리는 경우 최대 절전 모드를 사용해 인스턴스를 사전 워밍힐 수 있습니다. 인스턴스를 사전 예열하려면 다음을 수행하십시오.

- 최대 절전 모드가 활성화된 상태에서 시작하십시오.
- 원하는 상태로 만듭니다.
- 최대 절전 모드로 전환하여 필요에 따라 동일한 상태로 다시 시작할 수 있습니다.

stopped 상태가 되면 최대 절전 모드로 설정된 인스턴스의 인스턴스 사용량에 대해서는 요금이 청구되지 않습니다. RAM의 콘텐츠를 Amazon EBS 루트 볼륨으로 전송한 경우 인스턴스가 stopping 상태이면 인스턴스 사용에 대한 비용이 부과되지 않습니다. (이 기능은 최대 절전 모드를 실행하지 않고 [인스턴스를 중지](#) (p. 440)할 때와 다릅니다.) 데이터 전송 요금은 청구되지 않습니다. 그러나 RAM 콘텐츠에 대한 스토리지를 포함하여 모든 Amazon EBS 볼륨의 스토리지에 대해서는 요금이 부과됩니다.

인스턴스가 더 이상 필요하지 않을 경우 stopped(최대 절전 모드) 상태인 경우를 포함해 언제든지 인스턴스를 종료할 수 있습니다. 자세한 내용은 [인스턴스 종료](#) (p. 454) 단원을 참조하십시오.

Note

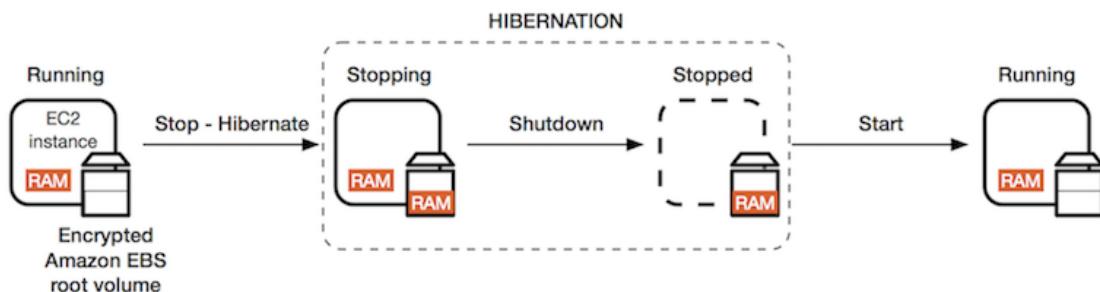
Linux 인스턴스에서 최대 절전 모드 실행은 Linux 인스턴스용 Amazon EC2 사용 설명서 단원의 [Linux 인스턴스 최대 절전 모드 실행](#)을 참조합니다.

목차

- [최대 절전 모드 개요](#) (p. 444)
- [최대 절전 모드 사전 조건](#) (p. 445)
- [제한 사항](#) (p. 446)
- [인스턴스에 대한 최대 절전 모드 활성화](#) (p. 446)
- [인스턴스를 최대 절전 모드로 전환](#) (p. 449)
- [최대 절전 모드의 인스턴스 시작](#) (p. 450)
- [최대 절전 모드 문제 해결](#) (p. 451)

최대 절전 모드 개요

다음 그림은 최대 절전 모드 프로세스 기본 개요를 보여줍니다.



실행 중인 인스턴스를 최대 절전 모드로 전환하면 다음과 같이 진행됩니다.

- 최대 절전 모드를 시작하면 인스턴스가 stopping 상태로 전환됩니다. Amazon EC2에서 최대 절전 모드 (디스크 일시 중단)를 수행하도록 운영 체제에 신호를 보냅니다. 최대 절전 모드를 실행하면 모든 프로세스가 동결되고, RAM의 콘텐츠를 Amazon EBS 루트 볼륨에 저장한 다음 일상적인 종료를 수행합니다.
- 종료가 완료되면 인스턴스가 stopped 상태가 됩니다.
- 모든 Amazon EBS 볼륨이 인스턴스에 연결된 상태로 유지되고 저장된 RAM 콘텐츠를 포함해 볼륨의 데이터도 유지됩니다.
- 대부분의 경우 인스턴스가 시작되면 새로운 기본 호스트 컴퓨터로 마이그레이션됩니다. 인스턴스를 중지 했다가 시작할 때도 이렇게 됩니다.
- 인스턴스를 시작하면 인스턴스가 부팅되고, 운영 체제가 Amazon EBS 루트 볼륨에서 RAM의 콘텐츠를 읽은 다음 프로세스가 동결 해제되어 상태를 다시 시작합니다.
- 인스턴스는 프라이빗 IPv4 주소와 모든 IPv6 주소를 유지합니다. 인스턴스를 시작할 때 인스턴스는 프라이빗 IPv4 주소와 모든 IPv6 주소를 계속 유지합니다.
- Amazon EC2는 퍼블릭 IPv4 주소를 해제합니다. 인스턴스를 시작하면 Amazon EC2는 인스턴스에 새 퍼블릭 IPv4 주소를 할당합니다.

- 인스턴스가 연결된 탄력적 IP 주소를 유지합니다. 최대 절전 모드 인스턴스와 연결된 모든 엘라스틱 IP 주소에 대한 요금이 부과됩니다. EC2-Classic을 사용하면 최대 절전 모드 시 탄력적 IP 주소가 인스턴스와 연결 해제됩니다. 자세한 내용은 [EC2-Classic \(p. 802\)](#) 단원을 참조하십시오.
- ClassicLink 인스턴스를 최대 절전 모드로 전환하면 연결되었던 VPC와의 연결이 해제됩니다. 인스턴스를 시작한 후 VPC에 다시 연결해야 합니다. 자세한 정보는 [ClassicLink \(p. 809\)](#) 단원을 참조하십시오.

최대 절전 모드와 재부팅, 중지 및 종료 간의 차이점은 [재부팅, 중지, 최대 절전 모드 및 종료의 차이 \(p. 374\)](#) 단원을 참조하십시오.

최대 절전 모드 사전 조건

인스턴스를 최대 절전 모드로 전환하려면 다음 사전 조건을 충족해야 합니다.

- 지원되는 인스턴스 패밀리 - C3, C4, C5, M3, M4, M5, M5a, R3, R4, R5, R5a 및 T2.
- 인스턴스 RAM 크기 - 이어야 합니다. 최대 16GB.
- 인스턴스 크기 - 베어 메탈 인스턴스에서는 지원되지 않습니다.
- 지원되는 AMI(최대 절전 모드를 지원하는 HVM AMI여야 함):
 - 2019년 9월 11일 릴리스된 Windows Server 2012 AMI 이상
 - 2019년 9월 11일 릴리스된 Windows Server 2012 R2 AMI 이상
 - 2019년 9월 11일 릴리스된 Windows Server 2016 AMI 이상
 - 2019년 9월 11일 릴리스된 Windows Server 2019 AMI 이상

Linux에 대해 지원되는 AMI에 대한 정보는, Linux 인스턴스용 Amazon EC2 사용 설명서 단원의 [최대 절전 모드 요구 사항](#)을 참조합니다.

- 루트 볼륨 유형 - 인스턴스 스토어 볼륨이 아닌 Amazon EBS 볼륨이어야 합니다.
- 지원되는 Amazon EBS 볼륨 유형 - 범용 SSD(gp2) 또는 프로비저닝된 IOPS SSD(io1 또는 io2). 프로비저닝된 IOPS SSD(io1 또는 io2) 볼륨 유형을 선택한 경우 최대 절전 모드에서 최적의 성능을 얻으려면 적절한 IOPS로 EBS 볼륨을 프로비저닝해야 합니다. 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.
- Amazon EBS 루트 볼륨 크기 - RAM 콘텐츠를 저장하고 예상한 사용량(예: OS 또는 애플리케이션)을 수용 할 수 있을 정도로 커야 합니다. 최대 절전 모드를 활성화하면 RAM 저장을 시작할 수 있도록 루트 볼륨에 공간이 할당됩니다.
- Amazon EBS 루트 볼륨 암호화 - 최대 절전 모드를 사용하려면 최대 절전 모드 시 메모리에 있는 중요한 콘텐트를 보호할 수 있도록 루트 볼륨을 암호화해야 합니다. RAM 데이터가 Amazon EBS 루트 볼륨으로 이전하면 항상 암호화됩니다. 루트 볼륨 암호화는 인스턴스 시작 시 적용됩니다. 루트 볼륨이 암호화된 Amazon EBS 볼륨인지 확인하려면 다음 세 가지 옵션 중 하나를 사용합니다.
 - EBS ‘단일 단계’ 암호화: 암호화되지 않은 AMI에서 암호화된 EBS 지원 EC2 인스턴스를 시작하고, 그 와 동시에 최대 절전 모드를 활성화할 수도 있습니다. 자세한 내용은 [EBS 지원 AMI를 통한 암호화 사용 \(p. 98\)](#) 단원을 참조하십시오.
 - 기본적인 EBS 암호화: 기본적으로 EBS 암호화를 활성화하여 AWS 계정에서 생성된 모든 새 EBS 볼륨이 암호화되도록 할 수 있습니다. 이러한 방식으로 인스턴스 실행 시 암호화 의도를 지정하지 않고 인스턴스에 대한 하이버네이션을 활성화할 수 있습니다. 자세한 내용은 [암호화 기본 제공 \(p. 1036\)](#) 단원을 참조하십시오.
 - 암호화된 AMI: 암호화된 AMI를 사용하여 인스턴스를 시작하는 방식으로 EBS 암호화를 활성화할 수 있습니다. AMI에 암호화된 루트 스냅샷이 없을 경우, 이를 새로운 AMI 및 요청 암호화에 복사할 수 있습니다. 자세한 내용은 [복사 중에 암호화되지 않은 이미지 암호화 \(p. 102\)](#) 및 [AMI 복사 \(p. 107\)](#) 단원을 참조하십시오.
- 시작 시 최대 절전 모드 활성화 - 기존 인스턴스(실행 또는 종지됨)에 대해서는 최대 절전 모드를 활성화할 수 없습니다. 자세한 내용은 [인스턴스에 대한 최대 절전 모드 활성화 \(p. 446\)](#) 단원을 참조하십시오.
- 구입 옵션 - 이 기능은 온디マン드 인스턴스 및 예약 인스턴스에 사용할 수 있습니다. 따라서 스팟 인스턴스를 지원하지 않습니다. 자세한 내용은 [중단된 스팟 인스턴스를 최대 절전 모드로 전환 \(p. 312\)](#) 단원을 참조하십시오.

제한 사항

- 최대 절전 모드에서는 다음 작업을 수행할 수 없습니다.
 - 최대 절전 모드 인스턴스의 유형 또는 크기 변경
 - 최대 절전 모드가 활성화된 인스턴스에서 스냅샷 또는 AMI 생성
 - 최대 절전 모드 인스턴스에서 스냅샷 또는 AMI 생성
- 인스턴스 스토어 지원 인스턴스는 중지하거나 최대 절전 모드로 전환할 수 없습니다.*
- 16GB 이상의 RAM이 있는 인스턴스를 최대 절전 모드로 전환할 수 없습니다.
- Auto Scaling 그룹에 속하거나 Amazon ECS에서 사용하는 인스턴스는 최대 절전 모드로 전환할 수 없습니다. 인스턴스가 Auto Scaling 그룹에 있으며 최대 절전 모드로 전환하려고 하면, Amazon EC2 Auto Scaling 서비스가 중지된 인스턴스를 비정상으로 간주해 이를 종료하고 대체 인스턴스를 시작할 수 있습니다. 자세한 정보는 Amazon EC2 Auto Scaling 사용 설명서의 [Auto Scaling 인스턴스에 대한 상태 점검 단원](#)을 참조하십시오.
- 인스턴스는 60일까지만 최대 절전 모드로 유지할 수 있습니다. 인스턴스를 60일 이상 유지하려면 최대 절전 모드의 인스턴스를 시작하고, 중지하고, 다시 시작해야 합니다.
- 업그레이드 및 보안 패치를 사용해 플랫폼을 지속적으로 업데이트하는 과정에서 기존의 최대 절전 모드 인스턴스와 충돌할 수 있습니다. 필요한 업그레이드 및 보안 패치를 적용하기 위해 종료 또는 재부팅하고자 최대 절전 모드의 인스턴스를 다시 시작해야 하는 중요 업데이트에 대해서는 AWS에서 알려 드립니다.

*최대 절전 모드가 활성화된 C3 및 R3 인스턴스의 경우 인스턴스 스토어 볼륨을 사용하지 마십시오.

인스턴스에 대한 최대 절전 모드 활성화

인스턴스를 최대 절전 모드로 전환하려면 먼저, 인스턴스에 대해 최대 절전 모드를 활성화해야 합니다. 최대 절전 모드를 활성화하려면 인스턴스를 시작하는 동안 수행해야 합니다.

Important

인스턴스를 시작한 후에는 인스턴스에 대해 최대 절전 모드를 활성화하거나 비활성화할 수 없습니다.

Console

콘솔을 사용하여 최대 절전 모드를 활성화하려면

- [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#)의 절차를 따르십시오.
- Amazon 머신 이미지 선택 페이지에서 최대 절전 모드를 지원하는 AMI를 선택합니다. 지원되는 AMI에 대한 자세한 내용은 [최대 절전 모드 사전 조건 \(p. 445\)](#) 단원을 참조하십시오.
- Choose an Instance Type(인스턴스 유형 선택) 페이지에서 지원되는 인스턴스 유형을 선택하고 Next: Configure Instance Details(다음: 인스턴스 정보 구성)를 선택합니다. 지원되는 인스턴스 유형에 대한 자세한 내용은 [최대 절전 모드 사전 조건 \(p. 445\)](#) 단원을 참조하십시오.
- Configure Instance Details(인스턴스 정보 구성) 페이지에서 Stop - Hibernate Behavior(중지 - 최대 절전 모드 동작)에 대해 Enable hibernation as an additional stop behavior(추가 중지 동작으로 최대 절전 모드 활성화) 확인란을 선택합니다.
- Add Storage(스토리지 추가) 페이지에서 루트 볼륨에 대해 다음 정보를 지정합니다.
 - 크기(GiB)에 Amazon EBS 루트 볼륨 크기를 입력합니다. 볼륨은 RAM 내용을 저장하고 예상 사용량을 수용할 수 있을 정도로 커야 합니다.
 - 볼륨 유형에서 지원되는 Amazon EBS 볼륨 유형(범용 SSD(gp2) 또는 프로비저닝된 IOPS SSD(io1 또는 io2))를 선택합니다.
 - 암호화에서 볼륨의 암호화 키를 선택합니다. 이 AWS 리전에서 기본적으로 암호화를 활성화한 경우 기본 암호화 키가 선택됩니다.

루트 볼륨의 사전 조건에 대한 자세한 내용은 [최대 절전 모드 사전 조건 \(p. 445\)](#) 단원을 참조하십시오.

6. 마법사에 표시되는 지침에 따라 계속합니다. 인스턴스 시작 검토 페이지에서 옵션 검토를 마쳤으면 시작을 선택합니다. 자세한 정보는 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오.

AWS CLI

AWS CLI를 사용하여 최대 절전 모드를 활성화하려면

`run-instances` 명령을 사용하여 인스턴스를 시작합니다. `--block-device-mappings file://mapping.json` 파라미터를 사용하여 EBS 루트 볼륨 파라미터를 지정하고, `--hibernation-options Configured=true` 파라미터를 사용하여 최대 절전 모드를 활성화합니다.

```
aws ec2 run-instances \
--image-id ami-0abcdef1234567890 \
--instance-type m5.large \
--block-device-mappings file://mapping.json \
--hibernation-options Configured=true \
--count 1 \
--key-name MyKeyPair
```

`mapping.json`에서 다음을 지정합니다.

```
[{"DeviceName": "/dev/xvda", "Ebs": {"VolumeSize": 30, "VolumeType": "gp2", "Encrypted": true}}]
```

Note

`DeviceName`의 값은 AMI와 연결된 루트 디바이스 이름과 일치해야 합니다. 루트 디바이스 이름을 찾으려면 다음과 같이 `describe-images` 명령을 사용합니다.

```
aws ec2 describe-images --image-id ami-0abcdef1234567890
```

이 AWS 리전에서 기본적으로 암호화를 활성화한 경우 `"Encrypted": true`를 생략할 수 있습니다.

Windows PowerShell용 AWS 도구

Windows PowerShell용 AWS 도구를 사용하여 최대 절전 모드를 활성화하려면

`New-EC2Instance` 명령을 사용하여 인스턴스를 시작합니다. 먼저 블록 디바이스 매핑을 정의한 다음 `-BlockDeviceMappings` 파라미터를 사용하여 명령에 추가하여 EBS 루트 볼륨을 지정합니다. `-HibernationOptions_Configured $true` 파라미터를 사용하여 최대 절전 모드를 활성화합니다.

```
PS C:\> $ebs_encrypt = New-Object Amazon.EC2.Model.BlockDeviceMapping
PS C:\> $ebs_encrypt.DeviceName = "/dev/xvda"
PS C:\> $ebs_encrypt.Ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
PS C:\> $ebs_encrypt.Ebs.VolumeSize = 30
PS C:\> $ebs_encrypt.Ebs.VolumeType = "gp2"
```

```
PS C:\> $ebs_encrypt.Ebs.Encrypted = $true

PS C:\> New-EC2Instance `

    -ImageId ami-0abcdef1234567890 `

    -InstanceType m5.large `

    -BlockDeviceMappings $ebs_encrypt `

    -HibernationOptions_Configured $true `

    -MinCount 1 `

    -MaxCount 1 `

    -KeyName MyKeyPair
```

Note

DeviceName의 값은 AMI와 연결된 루트 디바이스 이름과 일치해야 합니다. 루트 디바이스 이름을 찾으려면 다음과 같이 [Get-EC2Image](#) 명령을 사용합니다.

```
Get-EC2Image -ImageId ami-0abcdef1234567890
```

이 AWS 리전에서 기본적으로 암호화를 활성화한 경우 블록 디바이스 매핑에서 Encrypted = \$true를 생략할 수 있습니다.

Console

콘솔을 사용하여 인스턴스에 대해 최대 절전 모드가 활성화되어 있는지 확인하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 템색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 세부 정보 창에서 Stop - Hibernation behavior(중지 - 최대 절전 모드 동작)를 살펴봅니다. 활성은 인스턴스에 대해 최대 절전 모드가 활성화되어 있음을 나타냅니다.

AWS CLI

AWS CLI를 사용하여 인스턴스에 대해 최대 절전 모드가 활성화되어 있는지 확인하려면

`describe-instances` 명령을 사용해 최대 절전 모드가 활성화된 인스턴스를 필터링하도록 `--filters "Name=hibernation-options.configured,Values=true"` 파라미터를 지정합니다.

```
aws ec2 describe-instances \
  --filters "Name=hibernation-options.configured,Values=true"
```

출력의 다음 필드는 인스턴스에 대해 최대 절전 모드가 활성화되었음을 나타냅니다.

```
"HibernationOptions": {
  "Configured": true
}
```

Windows PowerShell용 AWS 도구

Windows PowerShell용 AWS 도구를 사용하여 인스턴스에 대해 최대 절전 모드가 활성화되어 있는지 확인하려면

`Get-EC2Instance` 명령을 사용해 최대 절전 모드가 활성화된 인스턴스를 필터링하도록 `-Filter @{ Name="hibernation-options.configured"; Value="true" }` 파라미터를 지정합니다.

```
Get-EC2Instance `
```

```
-Filter @{ Name="hibernation-options.configured"; Value="true" }
```

출력에는 최대 절전 모드로 활성화된 EC2 인스턴스가 나열됩니다.

인스턴스를 최대 절전 모드로 전환

인스턴스에 대해 [최대 절전 모드가 활성화 \(p. 446\)](#)되어 있고 [최대 절전 모드 사전 조건 \(p. 445\)](#)을 충족하는 경우 인스턴스를 최대 절전 모드로 전환할 수 있습니다. 인스턴스를 최대 절전 모드로 전환할 수 없는 경우 정상 종료가 진행됩니다.

Console

콘솔을 사용하여 Amazon EBS 지원 인스턴스를 최대 절전 모드로 전환하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, Instance State(인스턴스 상태), Stop - Hibernate(중지 - 최대 절전 모드)를 선택합니다. Stop - Hibernate(중지 - 최대 절전 모드)가 비활성화되어 있으면 해당 인스턴스가 이미 최대 절전 모드로 전환 또는 중지되었거나 최대 절전 모드로 전환할 수 없는 것입니다. 자세한 내용은 [최대 절전 모드 사전 조건 \(p. 445\)](#) 단원을 참조하십시오.
4. 확인 대화 상자가 나타나면 Yes, Stop - Hibernate(예, 중지 - 최대 절전 모드)를 선택합니다. 인스턴스가 최대 절전 모드로 전환하는 데 몇 분 정도 걸릴 수 있습니다. 인스턴스가 최대 절전 모드로 전환 중일 때는 Instance State(인스턴스 상태)가 중지 중으로 변경되고, 최대 절전 모드가 되면 중지 됨으로 바뀝니다.

AWS CLI

AWS CLI를 사용하여 Amazon EBS 지원 인스턴스를 최대 절전 모드로 전환하려면

`stop-instances` 명령을 사용하여 `--hibernate` 파라미터를 지정합니다.

```
aws ec2 stop-instances \
--instance-ids i-1234567890abcdef0 \
--hibernate
```

Windows PowerShell용 AWS 도구

Windows PowerShell용 AWS 도구를 사용하여 Amazon EBS 지원 인스턴스를 최대 절전 모드로 전환하려면

`Stop-EC2Instance` 명령을 사용하여 `-Hibernate $true` 파라미터를 지정합니다.

```
Stop-EC2Instance ^
-InstanceId i-1234567890abcdef0 ^
-Hibernate $true
```

Console

콘솔을 사용하여 인스턴스에 대해 최대 절전 모드가 시작되었는지 확인하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.

3. 인스턴스를 선택하고 세부 정보 창에서 상태 전환 이유 메시지를 살펴봅니다.
Client.UserInitiatedHibernate: User initiated hibernate 메시지는 인스턴스에 대해 최대 절전 모드가 시작되었음을 나타냅니다.

AWS CLI

AWS CLI를 사용하여 인스턴스에 대해 최대 절전 모드가 시작되었는지 확인하려면

`describe-instances` 명령을 사용해 최대 절전 모드가 시작된 인스턴스를 보려면 `state-reason-code` 필터를 지정합니다.

```
aws ec2 describe-instances \
--filters "Name=state-reason-code,Values=Client.UserInitiatedHibernate"
```

출력의 다음 필드는 인스턴스에 대해 최대 절전 모드가 시작되었음을 나타냅니다.

```
"StateReason": {
    "Code": "Client.UserInitiatedHibernate"
}
```

Windows PowerShell용 AWS 도구

Windows PowerShell용 AWS 도구를 사용하여 인스턴스에 대해 최대 절전 모드가 시작되었는지 확인하려면

`Get-EC2Instance` 명령을 사용해 최대 절전 모드가 시작된 인스턴스를 보려면 `state-reason-code` 필터를 지정합니다.

```
Get-EC2Instance ` 
-Filter @{Name="state-reason-code";Value="Client.UserInitiatedHibernate"}
```

출력에는 최대 절전 모드가 시작된 EC2 인스턴스가 나열됩니다.

최대 절전 모드의 인스턴스 시작

중지된 인스턴스를 다시 시작하는 것과 같은 방법으로 최대 절전 모드 인스턴스를 시작합니다.

Console

콘솔을 사용하여 최대 절전 모드 인스턴스를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 최대 절전 모드 인스턴스를 선택하고 작업, Instance State(인스턴스 상태), 시작을 선택합니다. 인스턴스가 running 상태가 되는 데 몇 분 정도 걸릴 수 있습니다. 이 과정에서 인스턴스 상태 확인(p. 652)에는 그 인스턴스가 시작될 때까지 실패 상태가 표시됩니다.

AWS CLI

AWS CLI를 사용하여 최대 절전 모드 인스턴스를 시작하려면

아래와 같이 `start-instances` 명령을 사용합니다.

```
aws ec2 start-instances \
--instance-ids i-1234567890abcdef0
```

Windows PowerShell용 AWS 도구

Windows PowerShell용 AWS 도구를 사용하여 최대 절전 모드 인스턴스를 시작하려면

`Start-EC2Instance` 명령을 사용합니다.

```
Start-EC2Instance  
-InstanceId i-1234567890abcdef0
```

최대 절전 모드 문제 해결

이 정보를 사용하여 인스턴스를 최대 절전 모드로 전환할 때 발생할 수 있는 문제를 진단 및 수정합니다.

시작 직후 최대 절전 모드로 전환할 수 없음

인스턴스를 시작한 후 너무 빨리 인스턴스를 최대 절전 모드로 전환하려고 하면 오류 메시지가 표시됩니다.

시작 후 최대 절전 모드로 전환하려면 약 ~5분 가량 기다려야 합니다.

stopping에서 stopped로 전환하는 데 너무 오래 걸리고 시작 후 메모리 상태가 복원되지 않음

최대 절전 모드 중인 인스턴스가 stopping 상태에서 stopped 상태로 전환되는데 너무 오래 걸리고 시작 후 메모리 상태가 복원되지 않는 경우 최대 절전 모드가 적절하게 구성되지 않았을 수 있습니다.

Windows Server 2016 이상

EC2 시작 로그를 확인하고 최대 절전 모드와 관련된 메시지를 살펴보십시오. EC2 시작 로그에 액세스하려면, 인스턴스에 [연결](#) (p. 436)하고 텍스트 편집기에서 C:\ProgramData\Amazon\EC2-Windows\Launch\Log\Ec2Launch.log 파일을 엽니다.

Note

기본적으로 Windows는 파일과 폴더를 C:\ProgramData 아래에 숨깁니다. EC2 디렉터리와 파일을 보려면 Windows 탐색기에 경로를 입력하거나 숨겨진 파일과 폴더를 표시하도록 폴더 속성을 변경합니다.

최대 절전 모드에 대한 로그 줄을 찾습니다. 긴 줄에 실패라고 표시되거나 로그 줄이 없는 경우 시작 시 최대 절전 모드 구성에 실패했을 가능성이 큽니다.

예를 들어 다음 메시지는 최대 절전 모드 구성에 실패했음을 나타냅니다: `Message: Failed to enable hibernation.`

의로그 줄이 `HibernationEnabled: true`를 포함하면 최대 절전 모드가 성공적으로 구성된 것입니다.

Windows Server 2012 R2 및 이전

EC2 구성 로그를 확인하고 최대 절전 모드와 관련된 메시지를 살펴보십시오. EC2 구성 로그에 액세스하려면, 인스턴스에 [연결](#) (p. 436)하고 텍스트 편집기에서 C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2ConfigLog.txt 파일을 엽니다. `SetHibernateOnSleep`에 대한 로그 줄을 찾습니다. 긴 줄에 실패라고 표시되거나 로그 줄이 없는 경우 시작 시 최대 절전 모드 구성에 실패했을 가능성이 큽니다.

예를 들어, 다음 메시지는 인스턴스 루트 볼륨이 충분히 크지 않음을 나타냅니다: `SetHibernateOnSleep: Failed to enable hibernation: Hibernation failed with the following error: There is not enough space on the disk.`

의 로그 줄이 SetHibernateOnSleep: HibernationEnabled: true이면 최대 절전 모드가 성공적으로 구성된 것입니다.

이러한 프로세스에서 어떠한 로그도 볼 수 없는 경우 AMI가 최대 절전 모드를 지원하지 않을 수 있습니다. 지원되는 AMI에 대한 내용은 [최대 절전 모드 사전 조건 \(p. 445\)](#) 단원을 참조하십시오.

인스턴스가 중지 상태에 멈춰 있음

인스턴스를 최대 절전 모드로 전환했는데 stopping 상태에 "멈춰" 있으면 강제로 중지할 수 있습니다. 자세한 내용은 [인스턴스 중지 문제 해결 \(p. 1204\)](#) 단원을 참조하십시오.

인스턴스 재부팅

인스턴스 재부팅은 운영 체제 재부팅과 같습니다. 대부분의 경우 인스턴스를 재부팅하는 데는 몇 분 밖에 걸리지 않습니다. 인스턴스를 재부팅하는 경우 퍼블릭 DNS 이름(IPv4), 프라이빗 IPv4 주소, IPv6 주소(해당되는 경우) 및 인스턴스 스토어 볼륨의 모든 데이터가 유지됩니다.

인스턴스를 중지했다가 다시 시작할 때와는 달리, 인스턴스를 재부팅해도 인스턴스 청구 시간이 새로 시작되지 않습니다.

재부팅이 필요한 업데이트를 적용해야 하는 경우와 같이 필수 유지 관리를 위해 인스턴스 재부팅을 예약해야 합니다. 사용자의 별도 작업은 필요하지 않습니다. 예약된 시간 내에 재부팅될 때까지 기다리는 것이 좋습니다. 자세한 내용은 [예약된 인스턴스 이벤트 \(p. 655\)](#) 단원을 참조하십시오.

Amazon EC2 콘솔, 명령줄 도구 또는 Amazon EC2 API를 사용하여 인스턴스에서 운영 체제 재부팅 명령을 실행하는 대신 인스턴스를 재부팅하는 것이 좋습니다. Amazon EC2 콘솔, 명령줄 도구 또는 Amazon EC2 API를 사용하여 인스턴스를 재부팅하는 경우 해당 인스턴스가 몇 분 이내에 완전히 종료되지 않으면 하드 재부팅을 수행합니다. AWS CloudTrail을 사용하는 경우 Amazon EC2를 사용하여 인스턴스를 재부팅해도 인스턴스가 재부팅되는 시점의 API 레코드가 생성됩니다.

Windows가 인스턴스에 업데이트를 설치할 경우 모든 업데이트가 설치될 때까지는 Amazon EC2 콘솔 또는 명령줄을 사용하여 인스턴스를 재부팅하거나 종료하지 않는 것이 좋습니다. 콘솔 또는 명령줄을 사용하여 인스턴스를 재부팅하거나 종료할 경우 인스턴스가 하드 재부팅할 위험이 있습니다. 업데이트가 설치되는 도중의 하드 재부팅은 인스턴스 상태를 불안정하게 만들 수 있습니다.

콘솔을 사용하여 인스턴스를 재부팅하려면

1. Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, 인스턴스 상태, 재부팅을 차례로 선택합니다.
4. 확인 메시지가 표시되면 예, 재부팅을 선택합니다. 인스턴스는 “실행 중” 상태로 유지됩니다.

명령줄을 사용하여 인스턴스를 재부팅하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [reboot-instances\(AWS CLI\)](#)
- [Restart-EC2Instance\(Windows PowerShell용 AWS 도구\)](#)

인스턴스 만료

AWS에서 인스턴스를 호스팅하는 기본 하드웨어의 복구 불가능한 장애가 검색되는 경우 인스턴스가 만료 대상으로 예약됩니다. 예약된 만료 날짜에 도달하면 인스턴스가 AWS에 의해 중지되거나 됩니다. 인스턴스 루트 디바이스가 Amazon EBS 볼륨인 경우 인스턴스가 중지되며 언제든지 이 인스턴스를 다시 시작할 수 있습니다. 중지된 인스턴스를 시작하면 새 하드웨어로 마이그레이션됩니다.

인스턴스 이벤트 유형에 대한 자세한 내용은 [예약된 인스턴스 이벤트 \(p. 655\)](#) 단원을 참조하십시오.

목차

- [만료 예약된 인스턴스 식별 \(p. 453\)](#)
- [만료 예약된 인스턴스에 대해 수행할 작업 \(p. 454\)](#)

만료 예약된 인스턴스 식별

인스턴스에 대한 만료가 예약되어 있는 경우 만료 이벤트가 발생하기 전에 인스턴스 ID와 만료 날짜가 포함된 이메일이 수신됩니다. Amazon EC2 콘솔이나 명령줄을 사용하여 만료 예약된 인스턴스를 확인할 수도 있습니다.

Important

인스턴스에 대한 만료가 예약되어 있는 경우 인스턴스에 연결할 수 없을 수 있으므로 가능한 한 빨리 조치를 취하는 것이 좋습니다. (수신한 이메일 알림에는 “이 성능 저하로 인해 이미 인스턴스에 연결할 수 없을 수 있습니다.”라는 메시지가 표시됩니다.) 수행해야 하는 권장 작업에 대한 자세한 내용은 [인스턴스에 연결할 수 있는지 확인](#) 단원을 참조하십시오.

만료 예약된 인스턴스를 식별하는 방법

- [이메일 알림 \(p. 453\)](#)
- [콘솔 식별 \(p. 453\)](#)

이메일 알림

인스턴스에 대한 만료가 예약되어 있는 경우 만료 이벤트가 발생하기 전에 인스턴스 ID와 만료 날짜가 포함된 이메일이 수신됩니다.

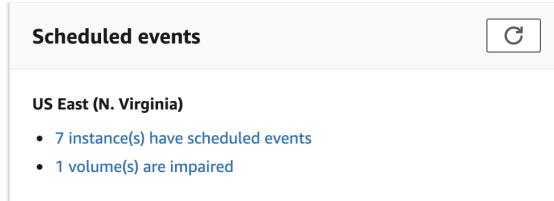
이 이메일은 계정과 연결된 주소로 전송됩니다. AWS Management 콘솔에 로그인하는 데 사용하는 이메일 주소와 동일합니다. 계정의 연락처 정보를 업데이트하려면 [계정 설정](#) 페이지로 이동합니다.

콘솔 식별

인스턴스 만료 알림을 정기적으로 확인하지 않는 이메일 계정을 사용하는 경우 Amazon EC2 콘솔이나 명령줄을 사용하여 인스턴스 중 하나에 대한 만료가 예약되어 있는지 여부를 확인할 수 있습니다.

콘솔을 사용하여 만료 예약된 인스턴스를 식별하려면

1. Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 EC2 대시보드를 선택합니다. 예약된 이벤트에서 리전별로 구성되어 있는 Amazon EC2 인스턴스 및 볼륨과 연결된 이벤트를 확인할 수 있습니다.



3. 예약된 이벤트가 나열되어 있는 인스턴스가 있는 경우 리전 이름 아래에 있는 링크를 선택하여 이벤트 페이지로 이동합니다.
4. 이벤트 페이지에는 이벤트가 연결되어 있는 모든 리소스가 나열됩니다. 만료가 예약되어 있는 인스턴스를 보려면 첫 번째 필터 목록에서 인스턴스 리소스를 선택하고 두 번째 필터 목록에서 인스턴스 종지 또는 만료를 선택합니다.

- 필터 결과에 인스턴스에 대한 만료가 예약되어 있는 것으로 나타나면 해당 인스턴스를 선택하고 세부 정보 창의 시작 시간 필드에 표시된 날짜와 시간을 기록해 둡니다. 이 날짜가 인스턴스 만료 날짜입니다.

명령줄을 사용하여 만료 예약된 인스턴스를 식별하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-instance-status\(AWS CLI\)](#)
- [Get-EC2InstanceState\(Windows PowerShell용 AWS 도구\)](#)

만료 예약된 인스턴스에 대해 수행할 작업

만료되는 인스턴스의 데이터를 보존하려면 다음 작업 중 하나를 수행할 수 있습니다. 예기치 않은 종단 시간 및 데이터 손실을 방지하려면 인스턴스 만료 날짜 전에 이 작업을 수행해야 합니다.

인스턴스에 연결할 수 있는지 확인

인스턴스에 대한 만료가 예약되어 있다는 알림을 받으면 가능한 한 빨리 다음 작업을 수행하는 것이 좋습니다.

- 인스턴스에 [??? \(p. 436\)](#) 인스턴스에 대해 ping을 실행하여 인스턴스에 연결할 수 있는지 확인합니다.
- 인스턴스에 연결할 수 있는 경우 예약된 만료 날짜 이전에 영향이 가장 적은 적합한 시간에 인스턴스를 종지/시작하도록 계획해야 합니다. 인스턴스 종지 및 시작과 인스턴스 종지 시 발생하는 결과(예: 인스턴스 와 연결된 퍼블릭, 프라이빗 및 탄력적 IP 주소에 대한 영향)에 대한 자세한 내용은 [인스턴스 종지 및 시작 \(p. 440\)](#) 단원을 참조하십시오. 인스턴스를 종지하고 시작하면 인스턴스 스토어 볼륨의 데이터가 손실 됩니다.
- 인스턴스에 연결할 수 없는 경우 즉시 작업을 수행하고 [종지/시작 \(p. 440\)](#)을 수행하여 인스턴스를 복구 해야 합니다.
- 또는 인스턴스를 [종료 \(p. 454\)](#)하려는 경우 가능한 한 빨리 종료하여 인스턴스에 대한 요금 발생을 종지 할 수 있습니다.

인스턴스의 백업 생성

인스턴스에서 EBS 지원 AMI를 생성하면 백업이 생깁니다. 데이터 무결성을 보장하려면 AMI를 생성하기 전에 인스턴스를 종지합니다. 예약된 만료 날짜까지 기다리거나(인스턴스가 종지되는 경우), 만료 날짜 전에 인스턴스를 종지합니다. 언제든지 인스턴스를 시작할 수 있습니다. 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오.

대체 인스턴스 시작

인스턴스에서 AMI를 생성한 후 AMI를 사용하여 대체 인스턴스를 시작할 수 있습니다. Amazon EC2 콘솔에서 새 AMI를 선택한 다음 작업(Actions), 시작(Launch)을 선택합니다. 마법사가 안내하는 대로 인스턴스를 시작합니다. 마법사의 각 단계에 대한 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하세요.

인스턴스 종료

더 이상 필요하지 않은 인스턴스는 삭제할 수 있습니다. 이를 인스턴스 종료라고 합니다. 인스턴스 상태가 `shutting-down` 또는 `terminated`로 변경되는 즉시 해당 인스턴스에 대한 반복적인 요금 부과가 종단됩니다.

인스턴스를 종료한 후에는 그 인스턴스에 다시 연결하거나 재시작할 수 없습니다. 하지만 동일한 AMI를 사용해서 추가 인스턴스를 실행할 수 있습니다. 인스턴스를 종지했다가 다시 시작하거나 최대 절전 모드로 전환하면 [인스턴스 종지 및 시작 \(p. 440\)](#) 또는 [Windows 인스턴스를 최대 절전 모드로 실행 \(p. 443\)](#) 단원을 참조하십시오. 자세한 내용은 [재부팅, 종지, 최대 절전 모드 및 종료의 차이 \(p. 374\)](#) 단원을 참조하십시오.

목차

- [인스턴스 종료 \(p. 455\)](#)
- [인스턴스 종료 시 발생하는 상황 \(p. 455\)](#)
- [인스턴스 종료 \(p. 455\)](#)
- [종료 방지 기능 활성화 \(p. 456\)](#)
- [인스턴스가 개시하는 종료 동작 변경 \(p. 457\)](#)
- [인스턴스 종료 시 Amazon EBS 볼륨 보존 \(p. 458\)](#)

인스턴스 종료

인스턴스는 종료한 후에도 잠시 동안 콘솔에 표시되며 그 이후 항목이 자동으로 삭제됩니다. 종료된 인스턴스 항목을 사용자가 직접 삭제할 수는 없습니다. 인스턴스가 종료되면 태그 및 볼륨과 같은 리소스가 해당 인스턴스에서 점차 연결 해제되며, 조금 지나면 종료된 인스턴스에서 더 이상 보이지 않을 수 있습니다.

인스턴스가 종료하면 해당 인스턴스와 관련된 모든 인스턴스 스토어 볼륨의 데이터는 삭제됩니다.

기본적으로 Amazon EBS 루트 디바이스 볼륨은 인스턴스 종료 시 자동으로 삭제됩니다. 하지만 시작 시 연결하는 추가 EBS 볼륨 또는 기존 인스턴스에 연결하는 EBS 볼륨은 인스턴스가 종료된 후에도 기본적으로 유지됩니다. 이런 동작은 해당 볼륨의 `DeleteOnTermination` 속성에 의해 제어되며, 이러한 속성은 사용자가 변경할 수 있습니다. 자세한 정보는 [인스턴스 종료 시 Amazon EBS 볼륨 보존 \(p. 458\)](#) 단원을 참조하십시오.

AWS Management 콘솔, CLI, API를 사용하는 타인의 실수로 인스턴스가 종료되는 것을 방지할 수 있습니다. 이 기능은 Amazon EC2 인스턴스 스토어 지원 및 Amazon EBS-지원 인스턴스에 대해 제공됩니다. 각 인스턴스는 `DisableApiTermination` 속성을 가지고 있으며 그 기본 값은 `false`로 설정되어 있습니다(해당 인스턴스는 Amazon EC2를 통해 종료할 수 있음). 인스턴스가 실행 중이거나 중단된 상태에 있을 때 이 인스턴스를 변경할 수 있습니다(Amazon EBS 지원 인스턴스의 경우). 자세한 내용은 [종료 방지 기능 활성화 \(p. 456\)](#) 단원을 참조하십시오.

시스템 종료에 대한 운영 체제 명령을 사용해서 인스턴스에서 종료를 개시한 경우, 인스턴스가 중단 또는 종료되는 것을 사용자가 제어할 수 있습니다. 자세한 내용은 [인스턴스가 개시하는 종료 동작 변경 \(p. 457\)](#) 단원을 참조하십시오.

인스턴스 종료에 대한 스크립트를 사용하는 경우 것을 보장할 방법이 없기 때문에 인스턴스가 비정상적으로 종료될 종료 스크립트를 실행합니다. Amazon EC2 가 인스턴스를 안전하게 종료하고 시스템 종료 스크립트를 실행하도록 작동하지만, 하드웨어 장애 등 특정 이벤트가 이런 시스템 종료 스크립트 실행을 방해할 수 있습니다.

인스턴스 종료 시 발생하는 상황

`terminate-instances` 명령을 사용하여 EC2 인스턴스를 종료하면 OS 수준에서 다음 항목이 등록됩니다.

- API 요청은 버튼 누름 이벤트를 게스트로 전송합니다.
- 다양한 시스템 서비스는 버튼 누름 이벤트의 결과로 종지됩니다. `systemd`는 시스템의 정상 종료를 처리합니다. 정상 종료는 하이퍼바이저에서 ACPI 종료 버튼 누름 이벤트에 의해 트리거됩니다.
- ACPI 종료가 시작됩니다.
- 정상 종료 프로세스가 종료되면 인스턴스가 종료됩니다. 구성 가능한 OS 종료 시간은 없습니다.

인스턴스 종료

AWS Management 콘솔 또는 명령줄을 사용해서 인스턴스를 종료할 수 있습니다.

콘솔을 사용한 인스턴스 종료 방법

1. 인스턴스를 종료하기 전에, Amazon EBS 볼륨이 종료 시 삭제되지 않는지 그리고 인스턴스 스토어 볼륨에서 Amazon EBS 또는 Amazon S3으로 필요한 데이터를 복사했는지를 확인해서 데이터 손실이 일어나지 않도록 합니다.
2. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
3. 탐색 창에서 인스턴스를 선택합니다.
4. 인스턴스를 선택하고 작업, 인스턴스 상태, 종료를 선택합니다.
5. 확인 메시지가 나타나면 예, 종료를 선택합니다.

명령줄을 사용한 인스턴스 종료 방법

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `terminate-instances(AWS CLI)`
- `Stop-EC2Instance(Windows PowerShell용 AWS 도구)`

종료 방지 기능 활성화

기본 설정상 Amazon EC2, 콘솔, 명령줄, API를 사용해서 인스턴스를 종료할 수 있습니다. Amazon EC2를 사용할 때 인스턴스가 실수로 종료되지 않도록 하기 위해 해당 인스턴스에 대한 종료 방지 기능을 활성화 할 수 있습니다. `DisableApiTermination` 속성은 콘솔, CLI, API를 사용해서 인스턴스가 종료될 수 있는지를 제어합니다. 기본 설정상 인스턴스에 대한 종료 보호 기능은 비활성화되어 있습니다. 인스턴스를 실행할 때 또는 인스턴스가 실행 중이거나 인스턴스가 중지되어 있을 때, 이 속성의 값을 설정할 수 있습니다 (Amazon EBS 지원 인스턴스의 경우).

`DisableApiTermination` 속성은 `InstanceInitiatedShutdownBehavior` 속성이 설정된 때에는 시스템 종료에 대한 운영 체제 명령을 사용해서 인스턴스에서 종료를 개시한 경우의 인스턴스 종료를 방지하지 않습니다. 자세한 내용은 [인스턴스가 개시하는 종료 동작 변경 \(p. 457\)](#) 단원을 참조하십시오.

제한 사항

스팟 인스턴스의 종료 방지 기능은 활성화할 수 없습니다. — 스팟 가격이 스팟 인스턴스에 대해 지불할 금액을 넘어서면 스팟 인스턴스가 종료됩니다. 하지만 스팟 인스턴스 중단을 처리할 수 있도록 애플리케이션을 준비하는 것은 가능합니다. 자세한 내용은 [스팟 인스턴스 중단 \(p. 310\)](#) 단원을 참조하십시오.

`DisableApiTermination` 속성으로는 Amazon EC2 Auto Scaling의 인스턴스 종료를 방지할 수 없습니다. Auto Scaling 그룹에 있는 인스턴스의 경우 Amazon EC2 종료 보호 대신 다음의 Amazon EC2 Auto Scaling 기능을 사용합니다.

- 인스턴스 보호 기능을 사용하면 확장 시 Auto Scaling 그룹에 속한 인스턴스가 종료되지 않습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [인스턴스 보호](#)를 참조하십시오.
- Amazon EC2 Auto Scaling의 비정상 인스턴스 종료를 방지하려면 `ReplaceUnhealthy` 프로세스를 일시 중단하십시오. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [조정 프로세스 일시 중단 및 재개](#) 단원을 참조하십시오.
- 종료 정책을 선택하여 Amazon EC2 Auto Scaling가 어떤 인스턴스를 먼저 종료해야 할지 지정하십시오. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [종료 정책 사용자 지정](#)을 참조하십시오.

실행 시에 인스턴스에 대한 종료 방지 기능 활성화 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 대시보드에서 인스턴스 시작을 선택하고 마법사의 지시를 따릅니다.

3. Configure Instance Details(인스턴스 세부 정보 구성) 페이지에서 종료 방지 기능 활성화 확인란을 선택합니다.

실행 중인 또는 중단된 인스턴스에 대한 종료 방지 기능 활성화 방법

1. 인스턴스를 선택하고 작업, 인스턴스 설정, 종료 방지 기능 변경을 선택합니다.
2. 예, 활성화를 선택합니다.

실행 중인 또는 중단된 인스턴스에 대한 종료 방지 기능 비활성화 방법

1. 인스턴스를 선택하고 작업, 인스턴스 설정, 종료 방지 기능 변경을 선택합니다.
2. 예, 비활성화를 선택합니다.

명령줄을 사용한 종료 방지 기능의 활성화 또는 비활성화 방법

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [modify-instance-attribute\(AWS CLI\)](#)
- [Edit-EC2InstanceAttribute\(Windows PowerShell용 AWS 도구\)](#)

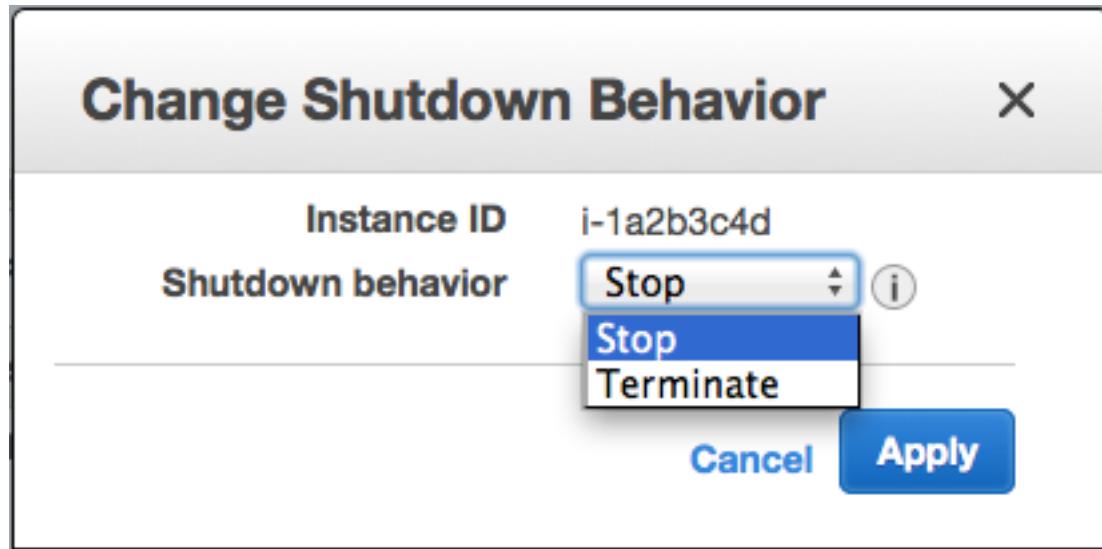
인스턴스가 개시하는 종료 동작 변경

기본적으로 (shutdown 또는 poweroff 등과 같은 명령을 사용하여) Amazon EBS 지원 인스턴스에서 종료를 시작하면 해당 인스턴스가 종지합니다. halt는 poweroff 명령을 실행하지 않는데, 이 명령을 사용하면 인스턴스가 종료되지 않고 대신 CPU를 HLT에 배치해 인스턴스가 실행 중인 상태로 남아 있습니다. 인스턴스에 대한 `InstanceInitiatedShutdownBehavior` 속성을 사용해서 이런 동작을 변경해서 인스턴스가 종단되지 않고 종료되도록 할 수 있습니다. 인스턴스가 실행 중이거나 중단된 상태에 있을 때 이 속성을 업데이트할 수 있습니다.

Amazon EC2 또는 명령줄을 사용하여 `InstanceInitiatedShutdownBehavior` 속성을 업데이트할 수 있습니다. `InstanceInitiatedShutdownBehavior` 속성은 인스턴스 자체의 운영 체제에서 종료하는 경우에만 적용되며, `StopInstances` API 또는 Amazon EC2 콘솔을 사용하여 인스턴스를 종지하는 경우에는 적용되지 않습니다.

콘솔을 사용한 인스턴스의 종료 동작 변경 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, 인스턴스 설정, 종료 동작 변경을 선택합니다. 현재 동작은 이미 선택된 상태입니다.
4. 동작을 변경하려면 종료 동작 목록에서 옵션을 선택하고 적용을 선택합니다.



명령줄을 사용한 인스턴스의 종료 동작 변경 방법

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [modify-instance-attribute\(AWS CLI\)](#)
- [Edit-EC2InstanceAttribute\(Windows PowerShell용 AWS 도구\)](#)

인스턴스 종료 시 Amazon EBS 볼륨 보존

인스턴스가 종료되면 Amazon EC2가 연결된 각 Amazon EBS 볼륨의 `DeleteOnTermination` 속성 값을 사용하여 볼륨 유지 또는 삭제 여부를 결정합니다.

`DeleteOnTermination` 속성의 기본값은 볼륨이 인스턴스의 루트 볼륨인지 아니면 인스턴스에 연결된 루트 외 볼륨인지에 따라 다릅니다.

루트 볼륨

기본적으로 인스턴스의 루트 볼륨의 `DeleteOnTermination` 속성은 `true`로 설정됩니다. 따라서 기본 값은 인스턴스가 종료될 때 인스턴스의 루트 볼륨을 삭제하는 것입니다. AMI를 생성한 사람과 인스턴스를 시작한 사람이 `DeleteOnTermination` 속성을 설정할 수 있습니다. AMI를 생성한 사람 또는 인스턴스를 시작한 사람이 속성을 변경하면 새로운 설정이 원래 AMI 기본 설정을 재정의합니다. AMI를 사용하여 인스턴스를 시작한 후에는 `DeleteOnTermination` 속성에 대한 기본 설정을 확인하는 것이 좋습니다.

루트 외 볼륨

기본적으로 [루트 외 EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#)하면 그 `DeleteOnTermination` 속성이 `false`로 설정됩니다. 따라서 기본값은 이러한 볼륨을 유지하는 것입니다. 인스턴스가 종료된 후에 유지된 볼륨의 스냅샷을 만들거나 다른 인스턴스에 연결할 수 있습니다. 추가 비용이 청구되지 않도록 하려면 볼륨을 삭제해야 합니다. 자세한 내용은 [Amazon EBS 볼륨 삭제 \(p. 966\)](#) 단원을 참조하십시오.

사용 중인 EBS 볼륨의 `DeleteOnTermination` 속성 값을 확인하려면 해당 인스턴스의 블록 디바이스 맵핑을 검색합니다. 자세한 내용은 [인스턴스 블록 디바이스 맵핑에서 EBS 볼륨 보기 \(p. 1114\)](#) 단원을 참조하십시오.

인스턴스를 시작할 때 또는 인스턴스 실행 중에 볼륨의 DeleteOnTermination 속성 값을 변경할 수 있습니다.

예제

- 콘솔을 사용하여 실행 시 유지할 루트 볼륨 변경 (p. 459)
- 명령줄을 사용하여 실행 시 유지할 루트 볼륨 변경 (p. 459)
- 명령줄을 사용하여 실행 중인 인스턴스의 루트 볼륨이 유지되도록 변경 (p. 460)

콘솔을 사용하여 실행 시 유지할 루트 볼륨 변경

콘솔을 사용하면 인스턴스를 시작할 때 DeleteOnTermination 속성을 변경할 수 있습니다. 실행 중인 인스턴스의 속성을 변경하려면 명령줄을 사용해야 합니다.

콘솔을 사용해서 실행 시에 인스턴스의 루트 볼륨이 유지되도록 변경하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 콘솔 대시보드에서 인스턴스 시작을 선택합니다.
3. Amazon 머신 이미지(AMI) 선택 페이지에서 AMI를 선택한 후 선택을 선택합니다.
4. 마법사 안내에 따라 인스턴스 유형 선택 및 인스턴스 세부 정보 구성 설정을 완료합니다.
5. 스토리지 추가 페이지에서 루트 볼륨에 대한 종료 시 삭제 확인란 선택을 해제합니다.
6. 나머지 마법사 페이지를 완료한 후 시작을 선택합니다.

인스턴스의 세부 정보 창에서 루트 디바이스 볼륨의 세부 정보를 조회하여 설정을 확인할 수 있습니다. 블록 디바이스 옆의 루트 디바이스 볼륨 항목을 선택합니다. 종료 시 삭제의 기본 설정은 True입니다. 기본 설정을 변경하면 종료 시 삭제의 설정 값이 False가 됩니다.

명령줄을 사용하여 실행 시 유지할 루트 볼륨 변경

EBS 지원 인스턴스를 시작할 때 다음 명령 중 하나를 사용해서 루트 디바이스 볼륨이 유지되도록 변경할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [run-instances\(AWS CLI\)](#)
- [New-EC2Instance\(Windows PowerShell용 AWS 도구\)](#)

예를 들어, 다음 옵션을 run-instances 명령에 추가합니다.

```
--block-device-mappings file://mapping.json
```

mapping.json에서 다음을 지정합니다.

```
[  
  {  
    "DeviceName": "/dev/sda1",  
    "Ebs": {  
      "DeleteOnTermination": false,  
      "SnapshotId": "snap-1234567890abcdef0",  
      "VolumeType": "gp2"  
    }  
  }  
]
```

명령줄을 사용하여 실행 중인 인스턴스의 루트 볼륨이 유지되도록 변경

다음 명령 중 하나를 사용하여 실행 중인 EBS 지원 인스턴스의 루트 디바이스 볼륨이 유지되도록 변경할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [modify-instance-attribute\(AWS CLI\)](#)
- [Edit-EC2InstanceAttribute\(Windows PowerShell용 AWS 도구\)](#)

예를 들어, 다음 명령을 사용합니다.

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --block-device-mappings file://mapping.json
```

mapping.json에서 다음을 지정합니다.

```
[  
  {  
    "DeviceName": "/dev/sda1",  
    "Ebs": {  
      "DeleteOnTermination": false  
    }  
  }  
]
```

인스턴스 복구

사용자는 Amazon EC2 인스턴스를 모니터링하고 기본 하드웨어 장애나 복구에 AWS 개입이 필요한 문제로 인해 인스턴스가 손상된 경우 인스턴스를 자동으로 복구하는 Amazon CloudWatch 정보를 만들 수 있습니다. 종료한 인스턴스는 복구할 수 없습니다. 복구된 인스턴스는 인스턴스 ID, 프라이빗 IP 주소, 탄력적 IP 주소 및 모든 인스턴스 메타데이터를 포함하여 원본 인스턴스와 동일합니다. 손상된 인스턴스가 배치 그룹에 있다면, 복구된 인스턴스는 배치 그룹에서 실행됩니다. 인스턴스 복구를 위한 Amazon CloudWatch 정보 사용에 대한 자세한 내용은 [Amazon CloudWatch 경보에 복구 작업 추가 \(p. 691\)](#) 단원을 참조하십시오. 인스턴스 복구 실패로 인한 문제를 해결하려면 [인스턴스 복구 실패 문제 해결 \(p. 461\)](#) 단원을 참조하십시오.

`StatusCheckFailed_System` 경보가 트리거되고 복구 작업이 시작되는 경우 경보를 생성하고 복구 작업을 연결할 때 선택한 Amazon SNS 주제로 통지됩니다. 인스턴스 복구 중에 인스턴스를 재부팅할 때 인스턴스가 마이그레이션되고 모든 인 메모리 데이터가 손실됩니다. 프로세스가 완료되면 해당 경보를 위해 구성해둔 SNS 주제로 정보가 게시됩니다. 이 SNS 주제에 가입되어 있는 사람은 누구나 복구 시도 상태와 세부 지침이 포함된 이메일 알림을 받게 됩니다. 복구된 인스턴스에서 인스턴스를 재부팅하라는 메시지가 나타납니다.

시스템 상태 확인이 실패하게 되는 문제의 예를 들면 다음과 같습니다.

- 네트워크 연결 끊김
- 시스템 전원 중단
- 물리적 호스트의 소프트웨어 문제
- 네트워크 연결성에 영향을 주는 물리적 호스트의 하드웨어 문제

인스턴스에 퍼블릭 IPv4 주소가 있는 경우 복구 후에도 해당 퍼블릭 IPv4 주소를 유지합니다.

요구 사항

복구 작업은 다음 특성을 지닌 인스턴스에만 지원됩니다.

- 다음 인스턴스 유형 중 하나를 사용합니다. C3, C4, C5, C5a, C5n, M3, M4, M5, M5a, M5n, P3, R3, R4, R5, R5a, R5n, T2, T3, T3a, X1 또는 X1e

- Virtual Private Cloud(VPC)에서 실행
- default 또는 dedicated 인스턴스 테넌시 사용
- EBS 볼륨만 있음(인스턴스 스토어 볼륨을 구성하지 않음)

인스턴스 복구 실패 문제 해결

다음 문제로 인해 인스턴스의 자동 복구가 실패할 수 있습니다.

- 대체 하드웨어의 일시적인 용량 부족
- 인스턴스에 인스턴스 스토어 스토리지가 연결되었으나, 자동 인스턴스 복구용으로 지원되지 않는 구성입니다.
- 진행 중인 서비스 상태 대시보드 이벤트가 있어서 복구 프로세스가 성공적으로 실행되지 못했습니다. 최신 서비스 가용성 정보는 <http://status.aws.amazon.com>을 참조하십시오.
- 인스턴스 복구 시도가 하루 최대 허용 횟수인 3회에 도달했습니다.

자동 복구 프로세스는 매일 최대 3회의 개별 실패에 대해서만 인스턴스 복구를 시도합니다. 인스턴스 시스템 상태 확인 실패가 계속되는 경우 인스턴스를 수동으로 중지 및 시작하는 것이 좋습니다. 자세한 내용은 [인스턴스 중지 및 시작 \(p. 440\)](#) 단원을 참조하십시오.

자동 복구가 실패하고 원래 시스템 상태 확인 실패의 근본 원인이 하드웨어 성능 저하로 확인되는 경우 이후에 인스턴스가 사용 중지될 수 있습니다.

Windows 인스턴스 구성

Windows 인스턴스는 클라우드에서 Windows Server를 실행하는 가상 서버입니다.

인스턴스를 시작하여 로그인한 후 인스턴스를 변경하여 특정 애플리케이션의 요구 사항에 맞게 구성할 수 있습니다. 다음은 시작하는 데 도움이 되는 몇 가지 일반적인 작업입니다.

내용

- [EC2Launch v2를 사용한 Windows 인스턴스 구성 \(p. 461\)](#)
- [EC2Launch를 사용한 Windows 인스턴스 구성 \(p. 489\)](#)
- [EC2Config 서비스를 사용한 Windows 인스턴스 구성 \(p. 496\)](#)
- [Windows 인스턴스의 반가상화\(PV\) 드라이버 \(p. 523\)](#)
- [Windows 인스턴스의 AWS NVMe 드라이버 \(p. 539\)](#)
- [CPU 옵션 최적화 \(p. 541\)](#)
- [Windows 인스턴스에 대한 시간 설정 \(p. 556\)](#)
- [Windows 인스턴스 암호 설정 \(p. 560\)](#)
- [설치 미디어를 사용하여 Windows 구성 요소 추가 \(p. 560\)](#)
- [Windows 인스턴스의 보조 프라이빗 IPv4 주소 구성 \(p. 564\)](#)
- [시작 시 Windows 인스턴스에서 명령 실행 \(p. 568\)](#)
- [인스턴스 메타데이터 및 사용자 데이터 \(p. 576\)](#)
- [EC2의 SQL Server 클러스터링에 대한 모범 사례 및 권장 사항 \(p. 605\)](#)

EC2Launch v2를 사용한 Windows 인스턴스 구성

Windows Server를 실행하는 지원되는 모든 Amazon EC2 인스턴스에는 EC2Launch v2 서비스 (`EC2Launch.exe`)가 포함됩니다. EC2Launch v2는 인스턴스 시작 중에 작업을 수행하며 인스턴스가 종지된 후 나중에 시작되거나 재시작된 경우 실행됩니다. EC2Launch v2는 온디マン드로 작업을 수행할 수도 있습니다.

니다. 이러한 작업 중 일부는 자동으로 활성화되고, 나머지는 수동으로 활성화해야 합니다. EC2Launch v2 서비스는 모든 EC2Config 및 EC2Launch 기능을 지원합니다.

이 서비스는 구성 파일을 사용하여 작업을 제어합니다. 그래픽 도구를 사용하거나 단일 .yml 파일(`agent-config.yml`)로 직접 편집하여 구성 파일을 업데이트할 수 있습니다. 서비스 바이너리는 `%ProgramFiles%\Amazon\EC2Launch` 디렉터리에 있습니다.

EC2Launch v2에서는 오류 문제를 해결하고 트리거를 설정하는 데 도움이 되는 Windows 이벤트 로그를 게시합니다. 자세한 내용은 [Windows 이벤트 로그 \(p. 484\)](#) 단원을 참조하십시오.

지원되는 운영 체제

- Windows Server 2019(장기 서비스 채널 및 연 2회 채널)
- Windows Server 2016
- Windows Server 2012 및 2012 R2
- Windows Server 2008 SP2 및 2008 R2

EC2Launch v2 단원 내용

- [EC2Launch v2 개요 \(p. 462\)](#)
- [EC2Launch v2의 최신 버전을 설치하려면 \(p. 465\)](#)
- [EC2Launch v2으로 마이그레이션 \(p. 465\)](#)
- [EC2Launch v2 종지, 재시작, 삭제 또는 제거 \(p. 466\)](#)
- [EC2Launch v2 버전 확인 \(p. 467\)](#)
- [EC2Launch v2 서비스 알림 구독 \(p. 467\)](#)
- [EC2Launch v2 설정 \(p. 467\)](#)
- [EC2Launch v2 문제 해결 \(p. 483\)](#)
- [EC2Launch v2 버전 기록 \(p. 489\)](#)

EC2Launch v2 개요

EC2Launch v2는 인스턴스 시작 시 작업을 수행하며 인스턴스가 중지된 후 나중에 시작되거나 재시작된 경우 실행되는 서비스입니다.

개요 주제

- [Amazon EC2 시작 서비스 비교 \(p. 462\)](#)
- [EC2Launch v2 개념 \(p. 463\)](#)
- [EC2Launch v2 작업 \(p. 464\)](#)

Amazon EC2 시작 서비스 비교

다음 표에서는 EC2Config, EC2Launch v1, EC2Launch v2 간의 주요 기능 차이점을 보여줍니다.

기능	EC2Config	EC2Launch v1	EC2Launch v2
다음으로 실행	Windows 서비스	PowerShell 스크립트	Windows 서비스
지원	Windows 2003 Windows 2008 Windows 2008 R2	Windows 2016 Windows 2019(LTSC 및 SAC)	Windows 2008 Windows 2008 R2 Windows 2012

기능	EC2Config	EC2Launch v1	EC2Launch v2
	Windows 2012 Windows 2012 R2		Windows 2012 R2 Windows 2016 Windows 2019(LTSC 및 SAC)
구성 파일	XML	XML	YAML
관리자 사용자 이름 설정	아니요	아니요	예
사용자 데이터 크기	16KB	16KB	60KB(압축)
AMI에서 베이크된 로컬 사용자 데이터	아니요	아니요	예, 구성 가능
사용자 데이터의 작업 구성	아니요	아니요	예
구성 가능한 월페이퍼 (wallpaper)	아니요	아니요	예
작업 실행 순서 사용자 지정	아니요	아니요	예
구성 가능한 작업 수	15	9	20(시작 시)
Windows 이벤트 뷰어 지원	예	아니요	예
이벤트 뷰어 이벤트 유형 수	2	0	30

EC2Launch v2 개념

EC2Launch v2를 고려하는 경우 다음 개념을 이해하는 것이 좋습니다.

작업

작업을 호출하여 인스턴스에 대한 작업을 수행할 수 있습니다. EC2Launch v2에서 사용 가능한 작업의 전체 목록은 [EC2Launch v2 작업 \(p. 464\)](#)을 참조하십시오. 각 작업에는 실행할 수 있는 일련의 스테이지, 정의된 빈도 및 입력이 포함됩니다. 작업은 agent-config 파일에서 또는 user-data를 통해 구성할 수 있습니다.

Stages

스테이지는 서비스에서 실행되는 작업으로 이루어진 논리적 그룹입니다. 일부 작업은 특정 스테이지에서만 실행할 수 있고, 일부 작업은 여러 스테이지로 실행할 수 있습니다. 로컬 데이터를 사용할 때는 작업이 실행될 스테이지를 지정해야 합니다. 사용자 데이터를 사용할 때는 스테이지가 내재되어 있습니다.

다음 목록에는 스테이지가 실행 순서대로 나와 있습니다.

1. 부팅
2. 네트워크
3. 사전 준비(PreReady)
4. 사후 준비(PostReady)
5. UserData

주파수

작업 빈도는 부팅 컨텍스트에 따라 작업 실행 일정을 설정하는 데 사용됩니다.

다음 빈도를 지정할 수 있습니다.

- 한 번 — AMI가 처음 부팅될 때 작업이 한 번 실행됩니다(Sysprep이 완료됨).
- 항상 — 첫 부팅뿐 아니라 AMI가 부팅될 때마다 작업이 실행됩니다.

agent-config

agent-config는 EC2Launch v2의 구성 폴더에 있는 파일로, 부팅, 네트워크, 사전 준비 및 사후 준비 스텝 이지에 대한 구성이 포함되어 있습니다. 이 파일은 AMI가 처음 부팅되거나 이후에 부팅될 때 실행되어야 하는 작업의 인스턴스 구성을 지정하는 데 사용됩니다.

기본적으로 EC2Launch v2 설치 시 표준 Amazon Windows AMI에서 사용되는 권장 구성은 포함한 agent-config 파일이 설치됩니다. 구성 파일을 업데이트하여 EC2Launch v2에서 지정되는 AMI의 기본 부팅 환경을 변경할 수 있습니다.

사용자 데이터

사용자 데이터는 인스턴스를 시작할 때 구성 가능한 데이터입니다. 사용자 데이터를 업데이트하여 사용자 지정 AMI 또는 퀵 스타트 AMI 구성 방식을 동적으로 변경할 수 있습니다. EC2Launch v2는 60kB의 사용자 데이터 입력 길이를 지원합니다. 사용자 데이터에는 사용자 데이터 스테이지만 포함되므로 agent-config 파일 이후에 실행됩니다.

EC2Launch v2 작업

EC2Launch v2에서는 부팅 때마다 다음 작업을 수행할 수 있습니다.

- 인스턴스에 대한 정보를 렌더링하는 새로운/선택적으로 사용자 지정된 월페이퍼(wallpaper)를 설정합니다.
- 로컬 시스템에 생성된 관리자 계정의 속성을 설정합니다.
- 검색 접미사 목록에 DNS 접미사를 추가합니다. 아직 존재하지 않는 접미사만 목록에 추가됩니다.
- 추가 볼륨에 대해 드라이브 문자를 설정하고 사용 가능한 공간을 사용하도록 확장합니다.
- 인터넷이나 구성의 파일을 디스크에 씁니다. 구성의 콘텐츠는 base64로 디코딩되거나 인코딩될 수 있습니다. 인터넷의 콘텐츠는 압축이 풀릴 수 있습니다.
- 인터넷이나 구성의 스크립트를 실행합니다. 구성의 스크립트는 base64로 디코딩될 수 있습니다. 인터넷의 스크립트는 압축이 풀릴 수 있습니다.
- 주어진 인수를 사용해 프로그램을 실행합니다.
- 컴퓨터 이름을 설정합니다.
- 인스턴스 정보를 Amazon EC2 콘솔에 전송합니다.
- RDP 인증서 지문을 EC2 콘솔에 전송합니다.
- 파티션 처리되지 않은 공간을 포함시키기 위한 운영 시스템 파티션을 동적으로 확장.
- 사용자 데이터를 실행합니다. 사용자 데이터 지정에 대한 자세한 내용은 [EC2Launch v2 작업 구성 \(p. 476\)](#) 단원을 참조하십시오.
- 영구 정적 경로를 설정하여 메타데이터 서비스 및 KMS 서버에 도달합니다.
- 부팅 파티션이 아닌 파티션을 MBR 또는 GPT로 설정합니다.
- Sysprep 이후에 Systems Manager(SSM) 서비스를 시작합니다.
- ENA 설정을 최적화합니다.
- 최신 Windows 버전을 위해 OpenSSH를 활성화합니다.
- 점보 프레임을 활성화합니다.
- EC2Launch v2에서 Sysprep이 실행되도록 설정합니다.
- Windows 이벤트 로그를 게시합니다.

EC2Launch v2의 최신 버전을 설치하려면

EC2Launch v2는 현재 다운로드하거나, SSM Distributor에서 설치하거나, 지원되는 모든 Windows AMI에서 사용할 수 있습니다.

다운로드

EC2Launch v2의 최신 버전을 설치하려면 다음 위치에서 서비스를 다운로드합니다.

Note

AmazonEC2Launch.msi는 EC2Launch(v1) 또는 EC2Config와 같은 EC2 시작 서비스의 이전 버전을 제거하지 않습니다. 이전 시작 서비스 버전에서 EC2Launch v2로 업그레이드하려면 [EC2Launch v2으로 마이그레이션 \(p. 465\)](#) 단원을 참조하세요.

- 64비트 — <https://s3.amazonaws.com/amazon-ec2launch-v2/windows/amd64/latest/AmazonEC2Launch.msi>
- 32비트 — <https://s3.amazonaws.com/amazon-ec2launch-v2/windows/386/latest/AmazonEC2Launch.msi>

AWS SSM Distributor에서 설치

AWS SSM Distributor에서 AWSEC2Launch-Agent 패키지를 설치할 수 있습니다. SSM Distributor에서 패키지를 설치하는 방법에 대한 자침은 AWS SSM 사용 설명서의 [패키지 설치 또는 업데이트](#)를 참조하십시오.

EC2Launch v2가 사전 설치된 AMI 사용(비프로덕션 워크로드)

EC2Launch v2는 다음 AMI에 사전 설치되어 있습니다. 이러한 AMI는 EC2Launch v2 서비스가 기존 프로세스 및 워크로드에서 제대로 작동하는지 확인하기 위한 목적으로 제공되므로 프로덕션 워크로드에는 사용하지 마십시오. Amazon EC2 콘솔에서 이러한 AMI를 찾거나 EC2 CLI를 사용하여 접두사 `EC2LaunchV2_Preview-Windows_Server-`로 검색할 수 있습니다.

- `EC2LaunchV2_Preview-Windows_Server-2004-English-Core-Base`
- `EC2LaunchV2_Preview-Windows_Server-2019-English-Full-Base`
- `EC2LaunchV2_Preview-Windows_Server-2019-English-Core-Base`
- `EC2LaunchV2_Preview-Windows_Server-2016-English-Full-Base`
- `EC2LaunchV2_Preview-Windows_Server-2016-English-Core-Base`
- `EC2LaunchV2_Preview-Windows_Server-2012_R2_RTM-English-Full-Base`
- `EC2LaunchV2_Preview-Windows_Server-2012_R2_RTM-English-Core`
- `EC2LaunchV2_Preview-Windows_Server-2012_RTM-English-Full-Base`
- `EC2LaunchV2_Preview-Windows_Server-2019-English-Full-SQL_2019_Express`
- `EC2LaunchV2_Preview-Windows_Server-2016-English-Full-SQL_2017_Express`

EC2Launch v2으로 마이그레이션

EC2Launch 마이그레이션 도구에는 이전 버전의 서비스를 제거하고 EC2Launch v2를 설치하는 업그레이드 옵션이 포함되어 있습니다. 이 옵션을 선택하면 이전 시작 서비스의 해당되는 모든 구성이 자동으로 새 서비스로 마이그레이션됩니다. EC2Config에서 마이그레이션하는 경우 EC2Config가 제거되면 [AWS Systems Manager\(SSM\) 에이전트](#)가 제거됩니다. EC2Launch v2 업그레이드는 SSM 에이전트의 안정적인 버전을 다시 설치합니다. 기본 변경 제어 프로세스를 사용하여 SSM 에이전트를 최신 버전으로 업데이트하는 것이 좋습니다.

마이그레이션 도구를 다운로드하거나 SSM RunCommand 문서를 사용하여 설치할 수 있습니다.

다음 위치에서 도구를 다운로드할 수 있습니다.

Note

관리자 권한으로 EC2Launch v2 마이그레이션 도구를 실행해야 합니다.

- 64비트 — <https://s3.amazonaws.com/amazon-ec2launch-v2-utils/MigrationTool/windows/amd64/latest/EC2LaunchMigrationTool.zip>
- 32비트 — <https://s3.amazonaws.com/amazon-ec2launch-v2-utils/MigrationTool/windows/386/latest/EC2LaunchMigrationTool.zip>

[AWS EC2Launch – Run Migration](#) SSM 문서를 사용해 SSM Run Command로 최신 EC2Launch 버전으로 마이그레이션합니다. 이 문서를 사용하는 데는 파라미터가 필요 없습니다. SSM Run Command 사용에 대한 자세한 내용은 [AWS Systems Manager Run Command](#)을 참조하십시오.

EC2Launch v2 중지, 재시작, 삭제 또는 제거

EC2Launch v2 서비스는 다른 Windows 서비스와 마찬가지 방식으로 관리할 수 있습니다.

EC2Launch v2는 부팅 시 한 번 실행되며 구성된 모든 작업을 실행합니다. 작업을 실행한 후 서비스는 중지 상태가 됩니다. 서비스를 다시 시작하면 서비스가 구성된 모든 작업을 다시 실행하고 중지된 상태로 돌아갑니다.

인스턴스에 업데이트된 설정을 적용하려면 서비스를 중단한 후에 재시작해야 합니다. EC2Launch v2를 수동으로 설치하는 경우에는 서비스를 먼저 중지해야 합니다.

EC2Launch v2 서비스를 중지하려면

1. 실행을 시작해서 Windows 인스턴스에 연결합니다.
2. 시작 메뉴에서 관리 도구를 선택한 다음에 서비스를 엽니다.
3. 서비스 목록에서 Amazon EC2Launch를 오른쪽 클릭하고 중지를 선택합니다.

EC2Launch v2 서비스를 다시 시작하려면

1. 실행을 시작해서 Windows 인스턴스에 연결합니다.
2. 시작 메뉴에서 관리 도구를 선택한 다음에 서비스를 엽니다.
3. 서비스 목록에서 Amazon EC2Launch를 오른쪽 클릭하고 다시 시작을 선택합니다.

구성 설정을 업데이트하거나, 고유 AMI를 생성하거나, AWS Systems Manager를 사용할 필요가 없는 경우에는 서비스를 삭제하고 제거할 수 있습니다. 서비스를 삭제하면 등록 서브키도 제거됩니다. 서비스를 제거하면 파일, 등록 서브키, 서비스 바로가기도 제거됩니다.

EC2Launch v2 서비스를 삭제하려면

1. 명령 프롬프트 창을 시작합니다.
2. 다음 명령을 실행합니다.

```
sc delete EC2Launch
```

EC2Launch v2를 제거하려면

1. 실행을 시작해서 Windows 인스턴스에 연결합니다.
2. 시작 메뉴에서 제어판을 선택합니다.
3. Programs and Features(프로그램 및 기능)을 엽니다.
4. 프로그램 목록에서 Amazon EC2Launch v2를 선택하고 제거를 선택합니다.

EC2Launch v2 버전 확인

Windows AMI에 포함된 EC2Launch v2 버전에 대한 자세한 내용은 [관리형 AWS Windows AMI \(p. 26\)](#) 단원을 참조하십시오.

EC2Launch v2의 최신 버전은 [EC2Launch v2 버전 기록 \(p. 489\)](#) 단원을 참조하십시오.

EC2Launch v2 마이그레이션 도구의 최신 버전은 [EC2Launch v2 마이그레이션 도구 버전 기록 \(p. 489\)](#) 단원을 참조하십시오.

새로운 EC2Launch v2 서비스 버전이 릴리스되면 알림을 받을 수 있습니다. 자세한 내용은 [EC2Launch v2 서비스 알림 구독 \(p. 467\)](#) 단원을 참조하십시오.

EC2Launch v2 서비스 알림 구독

새로운 EC2Launch v2 서비스 버전이 릴리스되면 이를 알리도록 Amazon SNS를 설정할 수 있습니다. 알림을 받으려면 다음 절차를 수행합니다.

EC2Launch v2 알림 구독

1. AWS Management 콘솔에 로그인한 다음 <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 필요한 경우 탐색 모음에서 리전을 미국 동부(버지니아 북부)로 변경합니다. 구독하는 SNS 알림이 이 리전에 생성되었기 때문에 이 리전을 선택해야 합니다.
3. 탐색 창에서 구독을 선택합니다.
4. 구독 생성을 선택합니다.
5. 구독 생성 대화 상자에서 다음 작업을 수행합니다.
 - a. 주제 ARN에서 다음 Amazon 리소스 이름(ARN)을 사용합니다. arn:aws:sns:us-east-1:309726204594:amazon-ec2launch-v2.
 - b. 프로토콜에서 이메일을 선택합니다.
 - c. 엔드포인트에는 알림을 받는 데 사용할 수 있는 이메일 주소를 입력합니다.
 - d. Create subscription을 선택합니다.
6. 구독을 확인하도록 요청하는 이메일이 전송되면 이메일을 열고 지침에 따라 구독을 완료합니다.

새 EC2Launch v2 서비스 버전이 릴리스될 때마다 구독자에게 알림이 전송됩니다. 이런 알림을 더 이상 받지 않기를 원하는 경우, 다음 절차를 수행해서 구독을 해제하십시오.

1. Amazon SNS 콘솔을 엽니다.
2. 탐색 창에서 구독을 선택합니다.
3. 구독을 선택한 후 작업, 구독 삭제를 선택합니다. 확인 메시지가 나타나면 삭제를 선택합니다.

EC2Launch v2 설정

이 단원에서는 EC2Launch v2에 대한 설정을 구성하는 방법을 설명합니다.

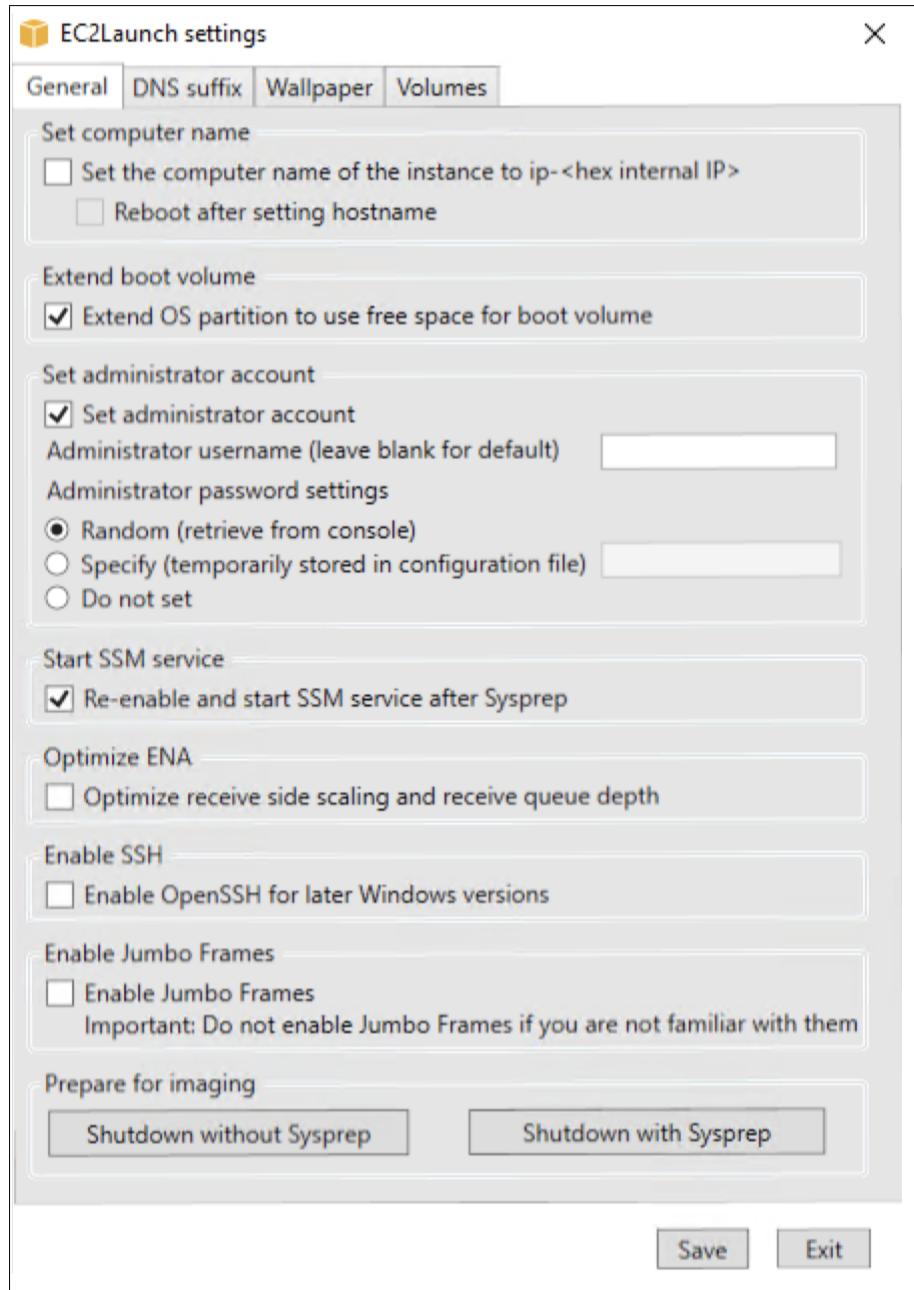
포함된 주제:

- [EC2Launch v2 설정 대화 상자를 사용하여 설정 변경 \(p. 468\)](#)
- [EC2Launch v2 디렉터리 구조 \(p. 472\)](#)
- [CLI를 사용하여 EC2Launch v2 구성 \(p. 473\)](#)
- [EC2Launch v2 작업 구성 \(p. 476\)](#)
- [EC2Launch v2 및 Sysprep \(p. 474\)](#)

EC2Launch v2 설정 대화 상자를 사용하여 설정 변경

다음 절차는 EC2Launch v2 설정 대화 상자를 사용해서 설정을 활성화 또는 비활성화하는 방법을 설명합니다.

1. 실행을 시작해서 Windows 인스턴스에 연결합니다.
2. 시작 메뉴에서 모든 프로그램을 선택한 다음 EC2Launch 설정으로 이동합니다.



3. EC2Launch 설정 대화 상자의 일반 탭에서 다음 설정을 활성화 또는 비활성화할 수 있습니다.

a. 컴퓨터 이름 설정

이 설정을 활성화하면(기본적으로 비활성화됨) 부팅 때마다 호스트 이름이 현재 내부 IP 주소와 비교됩니다. 호스트 이름과 내부 IP 주소가 일치하지 않으면 호스트 이름이 내부 IP 주소를 포함하도록 재설정된 후 시스템이 재부팅되어 새 호스트 이름이 선택됩니다. 자신의 호스트 이름을 설정하거나 기존 호스트 이름이 변경되는 것을 방지하려면 이 설정을 활성화하지 마십시오.

b. 부트 볼륨 확장

이 설정은 파티션 처리되지 않은 공간을 모두 포함하도록 등적으로 Disk 0/Volume 0을 확장합니다. 이는 인스턴스가 사용자 설정 크기를 가진 루트 디바이스 볼륨에서 부팅될 때 유용할 수 있습니다.

c. 관리자 계정 설정

이 기능이 활성화되면 로컬 시스템에 생성된 관리자 계정의 사용자 이름 및 암호 속성을 설정할 수 있습니다. 이 기능이 활성화되지 않으면 Sysprep 이후에 시스템에서 관리자 계정이 생성되지 않습니다. adminPasswordType이 Specify인 경우에만 adminPassword에 암호를 입력합니다.

암호 유형은 다음과 같이 정의됩니다.

i. Random

EC2Launch는 암호를 생성하고 사용자의 키를 사용하여 암호를 암호화합니다. 인스턴스가 재부팅 또는 중지되었다가 시작된 경우 이 암호가 그대로 유지되도록 시스템은 인스턴스가 시작된 후 이 설정을 비활성화합니다.

ii. Specify

adminPassword에 지정한 암호가 EC2Launch에 사용됩니다. 암호가 시스템 요구 사항에 맞지 않으면 EC2Launch에서 임의의 암호를 대신 생성합니다. 암호는 agent-config.yml에 일반 텍스트로 저장되며 Sysprep에서 관리자 암호를 설정한 후에 삭제됩니다. EC2Launch는 사용자의 키를 사용하여 암호를 암호화합니다.

iii. DoNothing

unattend.xml 파일에 지정한 암호가 EC2Launch에 사용됩니다. unattend.xml에 암호를 지정하지 않으면 관리자 계정이 비활성화됩니다.

d. SSM 서비스 시작

이 옵션을 선택하면 Sysprep 이후에 Systems Manager 서비스가 시작되도록 설정됩니다. EC2Launch v2는 [앞에서 \(p. 464\)](#) 설명한 모든 작업을 수행하며 SSM 에이전트는 Run Command 및 State Manager와 같은 Systems Manager 기능에 대한 요청을 처리합니다.

Run Command를 사용하여 기존 인스턴스가 최신 버전의 EC2Launch v2 서비스와 SSM 에이전트를 사용하도록 업그레이드할 수 있습니다. 자세한 내용은 AWS Systems Manager 사용 설명서의 [Run Command를 사용하여 SSM 에이전트 업데이트](#)를 참조하십시오.

e. ENA 최적화

이 옵션을 선택하면 ENA 수신측 조정 및 수신 대기열 깊이 설정이 AWS에 맞게 최적화되도록 ENA 설정이 구성됩니다. 자세한 내용은 [RSS CPU 선호도 구성 \(p. 758\)](#) 단원을 참조하십시오.

f. SSH 활성화

이 설정을 사용하면 최신 Windows 버전에서 원격 시스템 관리가 가능하도록 OpenSSH가 활성화됩니다.

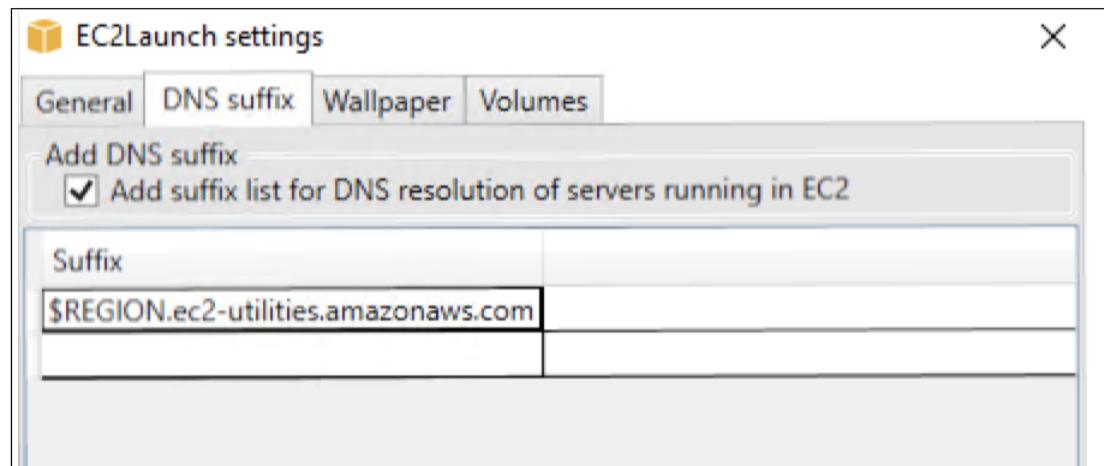
g. 점보 프레임 활성화

점보 프레임을 활성화하려면 선택합니다. 점보 프레임은 네트워크 통신에 의도하지 않은 영향을 줄 수 있으므로 점보 프레임을 활성화하기 전에 시스템에 미칠 수 있는 영향을 파악해야 합니다. 점보 프레임에 대한 자세한 내용은 [점보 프레임\(9001 MTU\) \(p. 773\)](#) 단원을 참조하십시오.

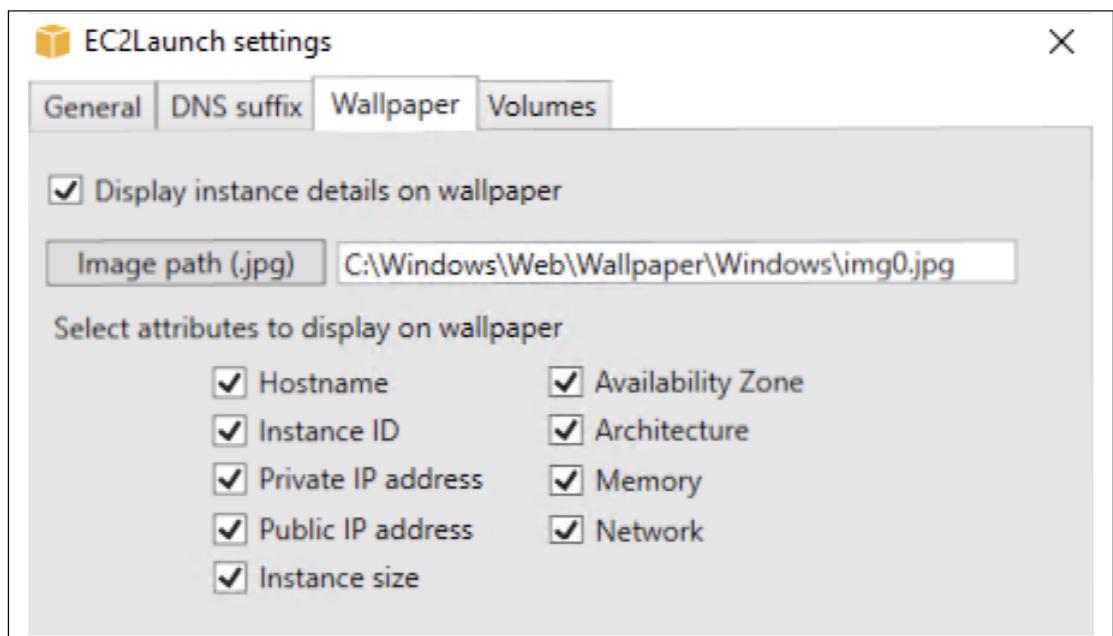
h. 이미징 준비

EC2 인스턴스를 종료할 때 Sysprep을 실행할지 아니면 실행하지 않을지를 선택합니다.
EC2Launch v2에서 Sysprep을 실행하려면 Sysprep을 실행하여 종료를 선택합니다.

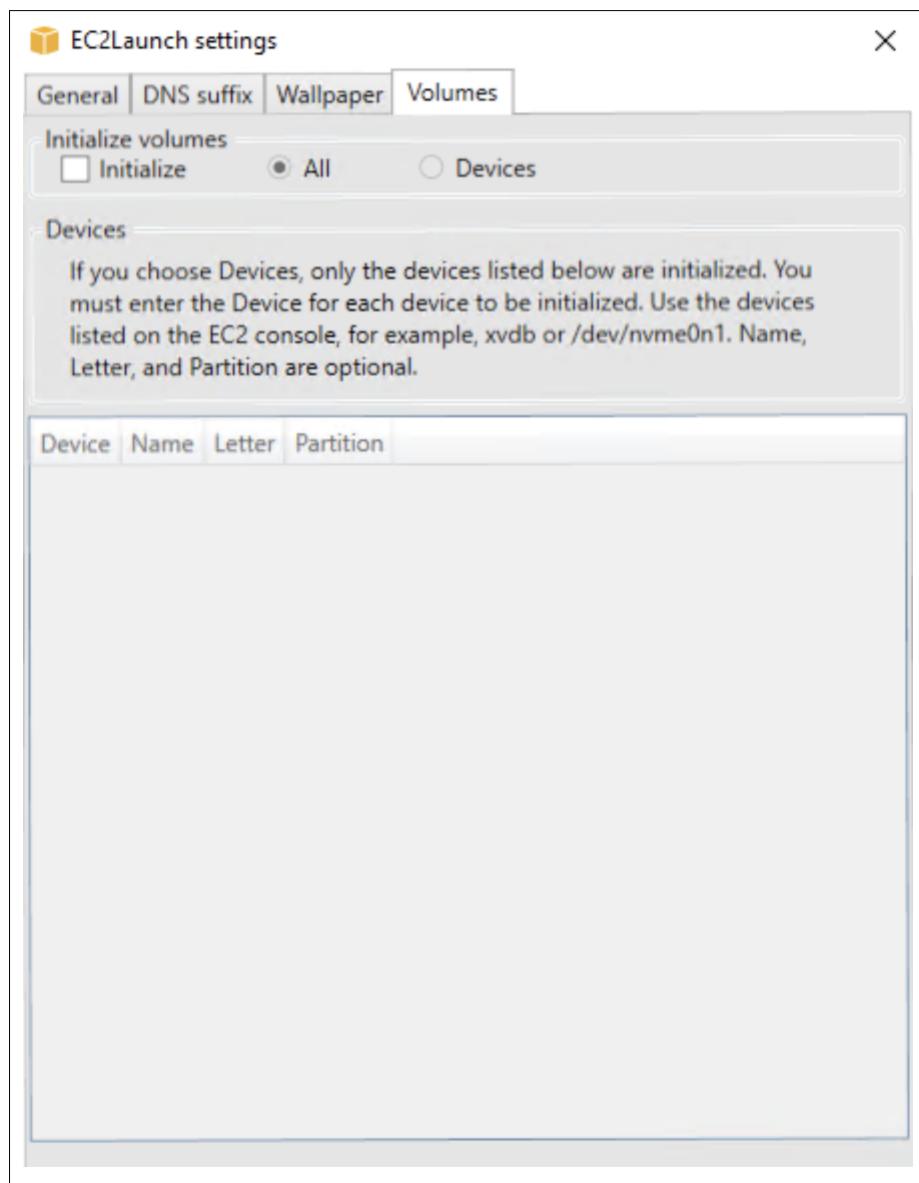
4. DNS 접미사 탭에서, 정규화된 도메인 이름을 제공하지 않고 EC2에서 실행되는 서버의 DNS를 확인 할 수 있도록 DNS 접미사 목록의 추가 여부를 선택할 수 있습니다. DNS 접미사는 변수 \$REGION 및 \$AZ를 포함할 수 있습니다. 아직 존재하지 않는 접미사만 목록에 추가됩니다.



5. 월페이퍼 탭에서 선택한 인스턴스 세부 정보를 월페이퍼에 표시하도록 설정할 수 있습니다. 사용자 지정 이미지를 선택할 수도 있습니다. 세부 정보는 로그인할 때마다 생성됩니다. 월페이퍼에서 인스턴스 세부 정보를 제거하려면 확인란의 선택을 취소합니다.



6. 볼륨 탭에서 인스턴스에 연결된 볼륨을 초기화할지 여부를 선택합니다. 활성화하면 추가 볼륨의 드라이브 문자가 설정되고 사용 가능한 공간을 사용하도록 확장됩니다. 모두를 선택하면 모든 스토리지 볼륨이 초기화됩니다. 디바이스를 선택하면 목록에 지정된 디바이스만 초기화됩니다. 초기화할 각 디바이스에 대해 디바이스를 입력해야 합니다. EC2 콘솔에 나열된 디바이스(예: xvdb 또는 /dev/nvme0n1)를 사용합니다. 이름, 문자 및 파티션은 선택적인 필드입니다. 파티션에 값을 지정하지 않으면 2TB보다 큰 스토리지 볼륨은 GPT 파티션 유형으로 초기화되고 2TB보다 작은 스토리지 볼륨은 MBR 파티션 유형으로 초기화됩니다. 디바이스가 구성되고 NTFS 이외의 디바이스에 파티션 테이블이 포함되어 있거나 디스크의 처음 4KB에 데이터가 포함된 경우 디스크를 건너뛰고 작업이 기록됩니다.



다음은 EC2Launch 대화 상자에 입력한 설정에서 생성된 구성 YAML 파일의 예입니다.

```
version: 1.0
```

```
config:
  - stage: boot
    tasks:
      - task: extendRootPartition
  - stage: preReady
    tasks:
      - task: activateWindows
        inputs:
          activation:
            type: amazon
      - task: setDnsSuffix
        inputs:
          suffixes:
            - $REGION.ec2-utilities.amazonaws.com
      - task: setAdminAccount
        inputs:
          password:
            type: random
      - task: setWallpaper
        inputs:
          path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
          attributes:
            - hostName
            - instanceId
            - privateIpAddress
            - publicIpAddress
            - instanceSize
            - availabilityZone
            - architecture
            - memory
            - network
  - stage: postReady
    tasks:
      - task: startSsm
```

EC2Launch v2 디렉터리 구조

EC2Launch v2는 다음 디렉터리에 설치해야 합니다.

- 서비스 바이너리: %ProgramFiles%\Amazon\EC2Launch
- 서비스 데이터(설정, 로그 파일 및 상태 파일): %ProgramData%\Amazon\EC2Launch

Note

기본적으로 Windows는 파일과 폴더를 C:\ProgramData 아래에 숨깁니다. EC2Launch v2 디렉터리와 파일을 보려면 Windows 탐색기에 경로를 입력하거나 숨겨진 파일과 폴더를 표시하도록 폴더 속성을 변경해야 합니다.

%ProgramFiles%\Amazon\EC2Launch 디렉터리에는 바이너리와 지원 라이브러리가 들어 있습니다. 여기에는 다음과 같은 하위 디렉터리가 포함됩니다.

- settings
 - EC2LaunchSettingsUI.exe - agent-config.yml 파일을 수정하기 위한 사용자 인터페이스
 - YamlDotNet.dll - 사용자 인터페이스에서 일부 작업을 지원하기 위한 DLL
- tools
 - ebsnvme-id.exe - 인스턴스에서 EBS 볼륨의 메타데이터를 검사하기 위한 도구
 - AWSAcpISpcrReader.exe - 사용할 올바른 COM 포트를 결정하기 위한 도구
 - EC2LaunchEventMessage.dll - EC2Launch에 대한 Windows 이벤트 로깅을 지원하기 위한 DLL
- EC2Launch.exe - 주요 EC2Launch 실행 파일

- `EC2LaunchAgentAttribution.txt` - EC2Launch 내에서 사용되는 코드의 저작권 표시

`%ProgramData%\Amazon\EC2Launch` 디렉터리에는 다음 하위 디렉터리가 포함됩니다. 로그, 구성 및 상태를 포함하여 서비스에서 생성된 모든 데이터가 이 디렉터리에 저장됩니다.

- `config` - 구성

서비스 구성 파일은 이 디렉터리에 `agent-config.yml`로 저장됩니다. 기본적으로 서비스에서 실행되는 작업을 수정, 추가 또는 제거하도록 이 파일을 업데이트할 수 있습니다.

- `log` - 인스턴스 로그

서비스(`agent.log`), 콘솔(`console.log`), 성능(`bench.log`) 및 오류(`error.log`)에 대한 로그가 이 디렉터리에 저장됩니다. 서비스의 후속 실행 시 로그 파일이 추가됩니다.

- `state` - 서비스 상태 데이터

서비스가 실행할 작업을 결정하는 데 사용하는 상태가 여기에 저장됩니다. Sysprep 이후에 서비스가 이미 실행되었는지 여부를 나타내는 `.run-once` 파일이 있습니다(따라서 빈도가 한 번인 작업은 다음 실행 시 건너뜀). 이 하위 디렉터리에는 각 작업의 상태를 추적하는 `state.json` 및 `previous-state.json`이 포함되어 있습니다.

- `sysprep` - Sysprep

이 디렉터리에는 재사용할 수 있는 사용자 지정 Windows AMI를 만들 때 Sysprep에서 수행할 작업을 결정하는 데 사용되는 파일이 포함되어 있습니다.

CLI를 사용하여 EC2Launch v2 구성

CLI(명령줄 인터페이스)를 사용하여 EC2Launch 설정을 구성하고 서비스를 관리할 수 있습니다. 다음 단원에서는 EC2Launch v2를 관리하는 데 사용할 수 있는 CLI 명령에 대한 설명과 사용법 정보를 제공합니다.

명령

- [collect-logs \(p. 473\)](#)
- [reset \(p. 474\)](#)
- [run \(p. 474\)](#)
- [sysprep \(p. 474\)](#)
- [validate \(p. 475\)](#)
- [version \(p. 475\)](#)
- [wallpaper \(p. 475\)](#)

collect-logs

EC2Launch에 대한 로그 파일을 수집하고, 파일을 압축하여 지정된 디렉터리에 배치합니다.

예

```
ec2launch collect-logs -o C:\Mylogs.zip
```

사용량

```
ec2launch collect-logs [flags]
```

Flags

`-h, --help`

collect-logs에 대한 도움말

-o, --output string

압축된 출력 로그 파일 경로

reset

한 번 실행하도록 지정된 작업이 다음 실행 시 실행되도록 .runonce 파일을 삭제합니다. 선택적으로 서비스 및 sysprep 로그도 삭제할 수 있습니다.

예

```
ec2launch reset -c
```

사용량

ec2launch reset [flags]

Flags

-c, --clean

reset 전에 인스턴스 로그 정리

-h, --help

reset에 대한 도움말

run

EC2Launch v2를 실행합니다.

예

```
ec2launch run
```

사용량

ec2launch run [flags]

Flags

-h, --help

reset에 대한 도움말

sysprep

서비스 상태를 재설정하고, unattend.xml을 업데이트하고, RDP를 비활성화하고, Sysprep을 실행합니다.

예:

```
ec2launch sysprep
```

사용량

ec2launch sysprep [flags]

Flags

-c,--clean

Sysprep 전에 인스턴스 로그 정리

-h,--help

Sysprep에 대한 도움말

-s,--shutdown

Sysprep 이후에 인스턴스 종료(기본값 true)

validate

agent-config 파일 C:\ProgramData\Amazon\EC2LaunchAgent\config\agent-config.yml을
검증합니다

예

```
ec2launch validate
```

사용량

ec2launch validate [flags]

Flags

-h , --help

validate에 대한 도움말

version

실행 가능한 버전을 가져옵니다.

예

```
ec2launch version
```

사용량

ec2launch version [flags]

Flags

-h, --help

version에 대한 도움말

wallpaper

제공된 월페이퍼 경로(.jpg 파일)로 새 월페이퍼를 설정하고 선택한 인스턴스 세부 정보를 표시합니다.

예

```
ec2launch wallpaper --path="C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg" --  
attributes=hostName,instanceId,privateIpAddress,publicIpAddress,instanceSize,availabilityZone,architect
```

사용량

```
ec2launch wallpaper [flags]
```

Flags

```
--attributes strings
```

월페이퍼 속성

```
-h, --help
```

```
wallpaper에 대한 도움말
```

```
-p, --path string
```

월페이퍼 파일 경로

EC2Launch v2 작업 구성

이 단원에는 `agent-config.yml` 및 `user-data.yml` 파일에 대한 구성 작업, 세부 정보 및 예시가 나와 있습니다.

작업 및 예시

- [activateWindows \(p. 476\)](#)
- [enableJumboFrames \(p. 477\)](#)
- [enableOpenSsh \(p. 477\)](#)
- [executeProgram \(p. 477\)](#)
- [executeScript \(p. 478\)](#)
- [extendRootPartition \(p. 479\)](#)
- [initializeVolume \(p. 479\)](#)
- [optimizeEna \(p. 480\)](#)
- [setAdminAccount \(p. 480\)](#)
- [setDnsSuffix \(p. 480\)](#)
- [setHostName \(p. 481\)](#)
- [setWallpaper \(p. 481\)](#)
- [startSsm \(p. 482\)](#)
- [writeFile \(p. 482\)](#)
- 예: `agent-config.yml` (p. 482)
- 예: `user-data.yml` (p. 483)

activateWindows

KMS 서버 집합에 대해 Windows를 활성화합니다.

Frequency - once

AllowedStages - [PreReady]

Inputs -

activation: (맵)

type: (문자열) 사용할 활성화 유형, amazon으로 설정

예

```
task: activateWindows
inputs:
  activation:
    type: amazon
```

enableJumboFrames

네트워크 어댑터의 최대 전송 단위(MTU)를 늘리는 점보 프레임을 활성화합니다. 자세한 내용은 [점보 프레임 \(9001 MTU\) \(p. 773\)](#) 단원을 참조하십시오.

Frequency - always

AllowedStages - [PostReady, UserData]

Inputs - none

예

```
task: enableJumboFrames
```

enableOpenSsh

Windows OpenSSH를 활성화하고 인스턴스의 공개 키를 인증된 키 폴더에 추가합니다.

Frequency - once

AllowedStages - [PreReady, UserData]

Inputs - none

예

다음 예에서는 인스턴스에서 OpenSSH를 활성화하고 인스턴스의 퍼블릭 키를 인증된 키 폴더에 추가하는 방법을 보여줍니다. 이 구성은 Windows Server 2019를 실행하는 인스턴스에서만 작동합니다.

```
task: enableOpenSsh
```

executeProgram

선택적 인수와 지정된 빈도를 사용하여 프로그램을 실행합니다.

Frequency - Inputs 참조

AllowedStages - [PostReady, UserData]

Inputs -

frequency: (문자열) once 또는 always 중 하나

path: (문자열) 실행 파일 경로

arguments: (문자열 목록) 실행 파일에 전달할 문자열 인수 목록

runAs: (문자열) localSystem으로 설정해야 함

예

다음 예에서는 이미 인스턴스에 있는 실행 파일을 실행하는 방법을 보여줍니다.

```
task: executeProgram
inputs:
- frequency: always
  path: C:\Users\Administrator\Desktop\setup.exe
  arguments: ['-quiet']
```

예제 2

다음 예에서는 이미 인스턴스에 있는 실행 파일을 실행하는 방법을 보여줍니다. 이 구성은 인스턴스의 c: 드라이브에 있는 VLC .exe 파일을 설치합니다. /L=1033 및 /S는 VLC .exe 파일과 함께 문자열 목록으로 전달되는 VLC 인수입니다.

```
task: executeProgram
inputs:
- frequency: always
  path: C:\vlc-3.0.11-win64.exe
  arguments: ['/L=1033', '/S']
  runAs: localSystem
```

executeScript

선택적 인수와 지정된 빈도를 사용하여 스크립트를 실행합니다.

Frequency - Inputs 참조

AllowedStages - [PostReady, UserData]

Inputs -

frequency: (문자열) once 또는 always 중 하나
type: (문자열) batch 또는 powershell 중 하나
arguments: (문자열 목록) 셸에 전달할 문자열 인수 목록
content: (문자열) 스크립트의 콘텐츠
runAs: (문자열) admin 또는 localSystem 중 하나

예

```
task: executeScript
inputs:
- frequency: always
  type: powershell
  content: |
    Get-Process | Out-File -FilePath .\Process.txt
  runAs: localSystem
```

예제 2

다음 예에서는 EC2 인스턴스에서 PowerShell 스크립트를 실행하는 방법을 보여줍니다. 이 구성은 c: 드라이브에 텍스트 파일을 만듭니다.

```
task: executeScript
inputs:
- frequency: always
  type: powershell
  content: |
    New-Item -Path 'C:\PowerShellTest.txt' -ItemType
```

File

다음 형식은 이 서비스의 이전 버전과 호환됩니다.

```
<powershell>
    $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

[extendRootPartition](#)

루트 볼륨을 확장하여 디스크에서 사용 가능한 공간을 모두 사용합니다.

Frequency - once

AllowedStages - [Boot]

Inputs - none

예

```
task: extendRootParitition
```

[initializeVolume](#)

인스턴스에 연결된 볼륨을 초기화하여 활성화 및 분할합니다. 비어 있지 않은 것으로 감지된 볼륨은 초기화되지 않습니다.

Frequency - always

AllowedStages - [PostReady, UserData]

Inputs -

initialize: (문자열) 사용할 초기화 전략 유형. all 또는 devices 중 하나

devices: (맵 목록)

device: 인스턴스를 생성할 때 사용되는 디바이스 식별자(예: xvdb, xvdf 또는 /dev/nvme0n1)

name: (문자열) 할당할 드라이브 이름

letter: (문자열) 할당할 드라이브 문자

partition: (문자열) 사용할 파티션 유형. mbr 또는 gpt 중 하나

예 1

다음 예에서는 선택한 볼륨이 초기화되도록 설정하는 InitializeVolume 작업에 대한 입력을 보여줍니다.

```
task: initializeVolume
inputs:
  initialize: devices
  devices:
    - device: xvdb
      name: MyVolumeOne
      letter: D
      partition: mbr
```

```
- device: /dev/nvme0n1
  name: MyVolumeTwo
  letter: E
  partition: gpt
```

예제 2

다음 예에서는 인스턴스에 연결된 EBS 볼륨을 초기화하는 방법을 보여줍니다. 이 구성은 인스턴스에 연결된 모든 빈 EBS 볼륨을 초기화합니다. 볼륨이 비어 있지 않으면 초기화되지 않습니다.

```
task: initializeVolume
inputs:
  initialize: all
```

optimizeEna

현재 인스턴스 유형에 따라 ENA 설정을 최적화합니다. 인스턴스를 재부팅할 수 있습니다.

Frequency - always

AllowedStages - [PostReady, UserData]

Inputs - none

예

```
task: optimizeEna
```

setAdminAccount

로컬 시스템에 생성된 기본 관리자 계정의 속성을 설정합니다.

Frequency - once

AllowedStages - [PreReady]

Inputs -

name: (문자열) 관리자 계정의 이름

password: (맵)

type: (문자열) 암호 설정 전략. static, random 또는 doNothing

doNothing: (문자열) 필드 유형에 따라 데이터 저장

data

예

```
task: setAdminAccount
inputs:
  name: Administrator
  password:
    type: random
```

setDnsSuffix

검색 접미사 목록에 DNS 접미사를 추가합니다. 아직 존재하지 않는 접미사만 목록에 추가됩니다.

Frequency - always

AllowedStages - [PreReady]

Inputs -

suffixes: (문자열 목록) 하나 이상의 유효한 DNS 접미사 목록. 유효한 대체 변수는 \$REGION 및 \$AZ

예

```
task: setDnsSuffix
inputs:
  suffixes:
    - $REGION.ec2-utilities.amazonaws.com
```

setHostName

컴퓨터의 호스트 이름을 프라이빗 IPv4 주소로 설정합니다.

Frequency - always

AllowedStages - [PostReady, UserData]

Inputs -

reboot: (부울) 호스트 이름이 변경될 때 재부팅이 허용되는지 여부를 나타냅니다.

예

```
task: setHostName
inputs:
  reboot: true
```

setWallpaper

인스턴스 속성을 표시하는 사용자 지정 월페이퍼를 사용해 인스턴스를 설정합니다.

Frequency - always

AllowedStages - [PreReady, UserData]

Inputs -

path: (문자열) 월페이퍼 이미지로 사용할 로컬 .jpg 파일 경로

attributes: (문자열 목록) 월페이퍼에 추가할 속성 목록. hostName, instanceId, privateIpAddress, publicIpAddress, instanceSize, availabilityZone, architecture, memory 또는 network 중 하나

예

```
task: setWallpaper
inputs:
  path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
  attributes:
    - hostName
    - instanceId
    - privateIpAddress
    - publicIpAddress
```

startSsm

Sysprep 이후에 Systems Manager(SSM) 서비스를 시작합니다.

Frequency - always

AllowedStages - [PostReady, UserData]

Inputs - none

예

```
task: startSsm
```

writeFile

대상에 파일을 씁니다.

Frequency - Inputs 참조

AllowedStages - [PostReady, UserData]

Inputs -

frequency: (문자열) once 또는 always 중 하나

destination: (문자열) 콘텐츠를 쓸 경로

content: (문자열) 대상에 쓸 텍스트

예

```
task: writeFile
inputs:
- frequency: once
  destination: C:\Users\Administrator\Desktop\booted.txt
  content: Windows Has Booted
```

예: agent-config.yml

다음 예에서는 agent-config.yml 구성 파일에 대한 설정을 보여줍니다.

```
version: 1.0
config:
- stage: boot
  tasks:
  - task: extendRootPartition
- stage: preReady
  tasks:
  - task: activateWindows
    inputs:
      activation:
        type: amazon
  - task: setDnsSuffix
    inputs:
      suffixes:
      - $REGION.ec2-utilities.amazonaws.com
- task: setAdminAccount
  inputs:
    password:
      type: random
- task: setWallpaper
```

```
inputs:
  path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
  attributes:
    - hostName
    - instanceId
    - privateIpAddress
    - publicIpAddress
    - instanceSize
    - availabilityZone
    - architecture
    - memory
    - network
  - stage: postReady
  tasks:
    - task: startSsm
```

예: user-data.yml

다음 예에서는 user-data.yml 구성 파일에 대한 설정을 보여줍니다.

```
version: 1.0
tasks:
- task: executeScript
  inputs:
    - frequency: always
      type: powershell
      runAs: localSystem
    content: |
      New-Item -Path 'C:\PowerShellTest.txt' -ItemType File
```

다음 형식은 이 서비스의 이전 버전과 호환됩니다.

```
<powershell>
  $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
  New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

EC2Launch v2 및 Sysprep

EC2Launch v2 서비스에서는 재사용할 수 있는 사용자 지정된 Windows AMI를 생성하는 데 사용할 수 있는 Microsoft 도구인 Sysprep을 실행합니다. EC2Launch v2는 Sysprep을 호출할 때 %ProgramData%\Amazon\EC2Launch의 파일을 사용하여 어느 작업을 수행할지 결정합니다. 이러한 파일은 EC2Launch 설정 대화 상자를 사용하여 간접적으로 편집하거나, YAML 편집기 또는 텍스트 편집기를 사용하여 직접 편집할 수 있습니다. 그러나 EC2Launch 설정 대화 상자에서 사용할 수 없는 몇 가지 고급 설정이 있으며, 이러한 항목은 직접 편집해야 합니다.

설정을 업데이트한 후에 인스턴스에서 AMI를 생성하면 새로운 설정은 그 새로운 AMI에서 실행하는 모든 인스턴스에 적용됩니다. AMI 생성에 대한 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조 하십시오.

EC2Launch v2 문제 해결

이 단원에서는 EC2Launch v2에 대한 일반적인 문제 해결 시나리오, Windows 이벤트 로그 보기에 대한 정보 및 콘솔 로그 출력과 메시지를 보여줍니다.

주제 문제 해결

- [일반적인 문제 해결 시나리오 \(p. 484\)](#)
- [Windows 이벤트 로그 \(p. 484\)](#)

- [EC2Launch v2 콘솔 로그 출력 \(p. 487\)](#)

일반적인 문제 해결 시나리오

이 단원에서는 일반적인 문제 해결 시나리오와 해결 단계를 보여 줍니다.

시나리오

- [서비스에서 월페이퍼를 설정하지 못함 \(p. 484\)](#)
- [서비스에서 사용자 데이터를 실행하지 못함 \(p. 484\)](#)
- [서비스에서 작업을 한 번만 실행함 \(p. 484\)](#)
- [서비스에서 작업을 실행하지 못함 \(p. 484\)](#)

서비스에서 월페이퍼를 설정하지 못함

해결

1. %AppData%\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\setwallpaper.lnk가 있는지 확인합니다.
2. %ProgramData%\Amazon\EC2Launch\log\agent.log를 검토해 오류가 발생했는지 확인합니다.

서비스에서 사용자 데이터를 실행하지 못함

가능한 원인: 사용자 데이터를 실행하기 전에 서비스가 실패했을 수 있습니다.

해결

1. %ProgramData%\Amazon\EC2Launch\state\previous-state.json을 검토합니다.
2. boot, network, preReady 및 postReadyLocalData가 모두 성공으로 표시되었는지 확인합니다.
3. 스테이지 중 하나가 실패한 경우 %ProgramData%\Amazon\EC2Launch\log\agent.log에서 구체적인 오류를 확인합니다.

서비스에서 작업을 한 번만 실행함

해결

1. 작업 빈도를 확인합니다.
2. 서비스가 Sysprep 이후에 이미 실행되었고 작업 빈도가 once로 설정된 경우 작업이 다시 실행되지 않습니다.
3. EC2Launch v2 실행 때마다 작업을 실행하려는 경우 작업 빈도를 always로 설정합니다.

서비스에서 작업을 실행하지 못함

해결

1. %ProgramData%\Amazon\EC2Launch\log\agent.log에서 최근 항목을 확인합니다.
2. 오류가 발생하지 않은 경우 "%ProgramFiles%\Amazon\EC2Launch\EC2Launch.exe" run에서 수동으로 서비스를 실행하여 작업이 성공했는지 확인합니다.

Windows 이벤트 로그

EC2Launch v2에서는 서비스 시작, Windows 준비, 작업 성공 및 실패와 같은 중요한 이벤트에 대한 Windows 이벤트 로그를 게시합니다. 이벤트 식별자는 특정 이벤트를 고유하게 식별합니다. 각 이벤트에는

스테이지, 작업 및 레벨 정보와 설명이 포함됩니다. 이벤트 식별자를 사용하여 특정 이벤트에 대한 트리거를 설정할 수 있습니다.

주제

- [이벤트 ID 형식 \(p. 485\)](#)
- [이벤트 ID 예 \(p. 485\)](#)
- [Windows 이벤트 로그 스키마 \(p. 486\)](#)

이벤트 ID 형식

다음 표에서는 EC2Launch v2 이벤트 식별자의 형식을 보여줍니다.

3	2 1	0
S	T	L

표의 문자와 숫자는 다음과 같은 이벤트 유형 및 정의를 나타냅니다.

이벤트 유형	정의
S(스테이지)	0 - 서비스 수준 메시지 1 - 부팅 2 - 네트워크 3 - 사전 준비 5 - Windows 준비 6 - 사후 준비 7 - 사용자 데이터
T(작업)	해당 두 값으로 표시되는 작업은 각 스테이지마다 다릅니다. 전체 이벤트 목록을 보려면 Windows 이벤트 로그 스키마 (p. 486) 를 참조하십시오.
L(이벤트 수준)	0 - 성공 1 - 정보 2 - 경고 3 - 오류

이벤트 ID 예

다음은 이벤트 ID의 예입니다.

- 5000 - Windows를 사용할 준비가 됨
- 3010 - 사전 준비 스테이지에서 Windows 활성화 작업 성공
- 6013 - 사후 준비 로컬 데이터 스테이지에서 월페이퍼 설정 작업에 오류 발생

Windows 이벤트 로그 스키마

메시지 ID/이벤트 ID	이벤트 메시지
... .0	Success
... .1	Informational
... .2	Warning
... .3	Error
x	EC2Launch service-level logs
0	EC2Launch Service exited successfully
1	EC2Launch Service starting
2	Error stopping EC2Launch service
10	Replace state.json with previous-state.json
100	Serial Port
200	Sysprep
300	PrimaryNic
400	Metadata
x000	Stage (1 digit), Task (2 digits), Status (1 digit)
1000	Boot
1010	Boot - extend_root_partition
2000	Network
2010	Network - add_routes
3000	PreReady
3010	PreReady - activate_windows
3020	PreReady - install_egpu_manager
3030	PreReady - set_monitor_on
3040	PreReady - set_hibernation
3050	PreReady - set_admin_account
3060	PreReady - set_dns_suffix
3070	PreReady - set_wallpaper
3080	PreReady - set_update_schedule
3090	PreReady - output_log
3100	PreReady - enable_open_ssh

메시지 ID/이벤트 ID	이벤트 메시지
5000	Windows is Ready to use
6000	PostReadyLocalData
7000	PostReadyUserData
6010/7010	PostReadyLocal/UserData - set_wallpaper
6020/7020	PostReadyLocal/UserData - set_update_schedule
6030/7030	PostReadyLocal/UserData - set_hostname
6040/7040	PostReadyLocal/UserData - execute_program
6050/7050	PostReadyLocal/UserData - execute_script
6060/7060	PostReadyLocal/UserData - manage_package
6070/7070	PostReadyLocal/UserData - initialize_volume
6080/7080	PostReadyLocal/UserData - write_file
6090/7090	PostReadyLocal/UserData - start_ssm
7100	PostReadyUserData - enable_open_ssh
6110/7110	PostReadyLocal/UserData - enable_jumbo_frames

EC2Launch v2 콘솔 로그 출력

이 단원에는 EC2Launch v2에 대한 샘플 콘솔 로그 출력이 포함되어 있으며, 문제를 해결하는 데 도움이 되는 모든 EC2Launch v2 콘솔 로그 오류 메시지가 나열됩니다.

출력

- [EC2Launch v2 콘솔 로그 출력 \(p. 487\)](#)
- [EC2Launch v2 콘솔 로그 메시지 \(p. 488\)](#)

EC2Launch v2 콘솔 로그 출력

다음은 EC2Launch v2에 대한 샘플 콘솔 로그 출력입니다.

```
2020/08/13 17:25:12Z: Windows is being configured. SysprepState=IMAGE_STATE_UNDEPLOYABLE
2020/08/13 17:27:44Z: Windows is being configured. SysprepState=IMAGE_STATE_UNDEPLOYABLE
2020/08/13 17:28:02Z: Windows sysprep configuration complete.
2020/08/13 17:28:03Z: Message: Waiting for meta-data accessibility...
2020/08/13 17:28:03Z: Message: Meta-data is now available.
2020/08/13 17:28:03Z: AMI Origin Version: 2020.07.15
2020/08/13 17:28:03Z: AMI Origin Name: EC2LaunchV2_Preview-Windows_Server-2012_R2_RTM-
English-Full-Base
2020/08/13 17:28:03Z: OS: Microsoft Windows NT 6.3.9600
```

```
2020/08/13 17:28:03Z: OsVersion: 6.3
2020/08/13 17:28:03Z: OsProductName: Windows Server 2012 R2 Standard
2020/08/13 17:28:03Z: OsBuildLabEx: 9600.19761.amd64fre.winblue_ltsb.200610-0600
2020/08/13 17:28:03Z: OsCurrentBuild: 9600
2020/08/13 17:28:03Z: Language: en-US
2020/08/13 17:28:03Z: TimeZone: GMT
2020/08/13 17:28:03Z: Offset: UTC +0000
2020/08/13 17:28:03Z: Launch: EC2 Launch v2.0.0
2020/08/13 17:28:03Z: AMI-ID: ami-1a2b3c4d
2020/08/13 17:28:03Z: Instance-ID: i-1234567890abcdef0
2020/08/13 17:28:03Z: Instance Type: t2.nano
2020/08/13 17:28:07Z: Driver: AWS PV Driver Package v8.3.3
2020/08/13 17:28:07Z: RDPCERTIFICATE-SUBJECTNAME: EC2AMAZ-A1B2C3D
2020/08/13 17:28:07Z: RDPCERTIFICATE-THUMBPRINT: A1B2C3D4E5
2020/08/13 17:28:12Z: SSM: Amazon SSM Agent v2.3.842.0
2020/08/13 17:28:13Z: Username: Administrator
2020/08/13 17:28:13Z: Password: <Password>
A1B2C3D4E5F6G7H8I9J10K11L12M13N14O15P16Q17
</Password>
2020/08/13 17:28:13Z: Message: Windows is Ready to use
```

EC2Launch v2 콘솔 로그 메시지

다음은 모든 EC2Launch v2 콘솔 로그 메시지 목록입니다.

```
Message: Error EC2Launch service is stopping. {error message}
Error setting up EC2Launch agent folders
See instance logs for detail
Error stopping service
Error initializing service
Message: Windows sysprep configuration complete
Message: Invalid administrator username: {invalid username}
Message: Invalid administrator password
Username: {username}
Password: <Password>{encrypted password}</Password>
AMI Origin Version: {amiVersion}
AMI Origin Name: {amiName}
Microsoft Windows NT {currentVersion}.{currentBuildNumber}
OsVersion: {currentVersion}
OsProductName: {productName}
OsBuildLabEx: {buildLabEx}
OsCurrentBuild: {currentBuild}
OsReleaseId: {releaseId}
Language: {language}
Timezone: {timeZone}
Offset: UTC {offset}
Launch agent: EC2Launch {BuildVersion}
AMI-ID: {amiId}
Instance-ID: {instanceId}
Instance Type: {instanceType}
RDPCERTIFICATE-SUBJECTNAME: {certificate subject name}
RDPCERTIFICATE-THUMBPRINT: {thumbprint hash}
SqlServerBilling: {sql billing}
SqlServerInstall: {sql patch leve, edition type}
Driver: AWS NVMe Driver {version}
Driver: Inbox NVMe Driver {version}
Driver: AWS PV Driver Package {version}
Microsoft-Hyper-V is installed.
Unable to get service status for vmms
Microsoft-Hyper-V is {status}
SSM: Amazon SSM Agent {version}
AWS VSS Version: {version}
Message: Windows sysprep configuration complete
Message: Windows is being configured. SysprepState is {state}
Windows is still being configured. SysprepState is {state}
```

```
Message: Windows is Ready to use
Message: Waiting for meta-data accessibility...
Message: Meta-data is now available.
Message: Still waiting for meta-data accessibility...
Message: Failed to find primary network interface...retrying...
```

EC2Launch v2 버전 기록

버전 기록

- [EC2Launch v2 버전 기록 \(p. 489\)](#)
- [EC2Launch v2 마이그레이션 도구 버전 기록 \(p. 489\)](#)

EC2Launch v2 버전 기록

다음 표에서는 EC2Launch v2의 릴리스 버전에 대해 설명합니다.

버전	세부 정보	릴리스 날짜
2.0.124	<ul style="list-style-type: none">• 배경 화면에 OS 버전을 표시하는 옵션을 추가합니다.• 암호화된 EBS 볼륨을 초기화합니다.• 로컬 DNS 이름이 없는 VPC에 대한 경로를 추가합니다.	2020년 9월 10일
2.0.104	<ul style="list-style-type: none">• DNS 접미사 검색 목록이 존재하지 않으면 만듭니다.• 요청받지 않으면 최대 절전 모드를 건너뜁니다.	2020년 8월 12일
2.0.0	최초 릴리스.	2020년 6월 30일

EC2Launch v2 마이그레이션 도구 버전 기록

다음 표에서는 EC2Launch v2 마이그레이션 도구의 릴리스 버전에 대해 설명합니다.

버전	세부 정보	릴리스 날짜
1.0.60	EC2Launch 에이전트의 버전 번호를 2.0.124로 증가시킵니다.	2020년 9월 10일
1.0.54	<ul style="list-style-type: none">• 에이전트가 설치되지 않은 경우 EC2Launch v2를 설치합니다.• EC2Launch 에이전트의 버전 번호를 2.0.104로 증가시킵니다.• SSM 에이전트를 분리합니다.	2020년 8월 12일
1.0.50	NuGet 종속성을 제거합니다.	2020년 8월 10일
1.0.0	최초 릴리스.	2020년 6월 30일

EC2Launch를 사용한 Windows 인스턴스 구성

EC2Launch는 Windows Server 2016 이후 AMI에서 EC2Config 서비스를 대체한 Windows PowerShell 스크립트 세트입니다. 지원되는 모든 Windows Server 버전에 대한 최신 시작 서비스는 EC2Config와 EC2Launch를 모두 대체하는 [EC2Launch v2 \(p. 461\)](#)입니다.

목차

- [EC2Launch 작업 \(p. 490\)](#)
- [최신 버전의 EC2Launch 설치 \(p. 490\)](#)
- [EC2Launch 버전 확인 \(p. 491\)](#)
- [EC2Launch 디렉터리 구조 \(p. 491\)](#)
- [EC2Launch 구성 \(p. 491\)](#)
- [EC2Launch 버전 기록 \(p. 494\)](#)

EC2Launch 작업

EC2Launch는 초기 인스턴스 부팅 중에 기본적으로 다음 작업을 수행합니다.

- 인스턴스에 대한 정보를 렌더링하는 새로운 월페이퍼(wallpaper)를 설정합니다.
- 컴퓨터 이름을 설정합니다.
- 인스턴스 정보를 Amazon EC2 콘솔에 전송합니다.
- RDP 인증서 지문을 EC2 콘솔에 전송합니다.
- 관리자 계정에 대한 무작위 암호를 설정합니다.
- DNS 접미사를 추가합니다.
- 분할되지 않은 공간을 포함하도록 운영 체제 파티션을 동적으로 확장합니다.
- 사용자 데이터를 실행합니다(지정된 경우). 사용자 데이터 지정에 대한 자세한 내용은 [인스턴스 사용자 데이터로 작업 \(p. 589\)](#) 단원을 참조하십시오.
- 영구 정적 경로를 설정하여 메타데이터 서비스 및 KMS 서버에 도달합니다.

Important

사용자 지정 AMI를 이 인스턴스에서 생성한 경우 이 경로는 OS 구성의 일부로서 캡처되며 AMI에서 시작한 새로운 인스턴스는 서브넷 배치와 상관없이 동일한 경로를 유지합니다. 경로를 업데이트하려면 [사용자 지정 AMI 시작 시 Server 2016 이후에 대한 메타데이터/KMS 경로 업데이트 \(p. 94\)](#) 단원을 참조하십시오.

다음 작업은 EC2Config 서비스와 역방향 호환성을 유지하는 데 도움이 됩니다. 스타트업 중에 이러한 작업을 수행하도록 EC2Launch를 구성할 수도 있습니다.

- 둘째 EBS 볼륨을 초기화합니다.
- Windows Event 로그를 EC2 콘솔 로그에 전송합니다.
- Windows is ready to use 메시지를 EC2 콘솔에 전송합니다.

Windows Server 2019에 대한 자세한 내용은 Microsoft.com에서 [Windows Server 버전별 기능 비교](#)를 참조하십시오.

최신 버전의 EC2Launch 설치

다음 절차를 이용해 인스턴스에 최신 버전의 EC2Launch를 다운로드하여 설치합니다.

최신 버전의 EC2Launch 다운로드하여 설치하기

1. 이미 인스턴스에 EC2Launch를 설치하여 구성한 경우 EC2Launch 구성 파일의 백업을 만듭니다. 설치 프로세스는 이 파일에 변경 사항을 보존하지 않습니다. 기본적으로 c:\ProgramData\Amazon\EC2-Windows\Launch\Config 디렉터리에 파일이 위치합니다.

2. 인스턴스의 디렉터리로 [EC2-Windows-Launch.zip](#) 파일을 다운로드합니다.
3. EC2-Windows-Launch.zip 파일을 다운로드한 동일한 디렉터리에 [install.ps1](#)을 다운로드합니다.
4. 실행 [install.ps1](#)
5. EC2Launch 구성 파일의 백업을 만든 경우 C:\ProgramData\Amazon\EC2-Windows\Launch\Config 디렉터리에 복사합니다.

EC2Launch 버전 확인

다음 Windows PowerShell 명령을 사용하여 설치된 EC2Launch 버전을 확인합니다.

```
PS C:\> Test-ModuleManifest -Path "C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Ec2Launch.ps1" | Select Version
```

EC2Launch 디렉터리 구조

EC2Launch는 기본적으로 루트 디렉터리 C:\ProgramData\Amazon\EC2-Windows\Launch의 Windows Server 2016 이후 AMI에 설치됩니다.

Note

기본적으로 Windows는 파일과 폴더를 C:\ProgramData 아래에 숨깁니다. EC2Launch 디렉터리와 파일을 보려면 Windows 탐색기에 경로를 입력하거나 숨겨진 파일과 폴더를 표시하도록 폴더 속성을 변경해야 합니다.

Launch 디렉터리에는 다음 하위 디렉터리가 포함됩니다.

- Scripts - EC2Launch를 구성하는 PowerShell 스크립트가 포함됩니다.
- Module - Amazon EC2와 관련된 스크립트를 빌드하기 위한 모듈이 포함됩니다.
- Config - 사용자가 사용자 지정할 수 있는 스크립트 구성 파일이 포함됩니다.
- Sysprep - Sysprep 리소스가 포함됩니다.
- Settings - Sysprep 그래픽 사용자 인터페이스용 애플리케이션이 포함됩니다.
- Logs - 스크립트가 생성하는 로그 파일이 포함됩니다.

EC2Launch 구성

인스턴스가 처음으로 초기화된 후에는 다시 실행하고 다른 스타트업 작업을 수행하도록 EC2Launch를 구성할 수 있습니다.

작업

- [초기화 작업 구성 \(p. 491\)](#)
- [매 부팅마다 실행하도록 EC2Launch 예약 \(p. 492\)](#)
- [드라이브 및 맵 드라이브 문자 초기화 \(p. 493\)](#)
- [EC2 콘솔에 Windows 이벤트 로그 전송 \(p. 494\)](#)
- [부팅 성공 후 Windows is ready 메시지 전송 \(p. 494\)](#)

초기화 작업 구성

LaunchConfig.json 구성 파일에서 설정을 지정하여 다음과 같은 초기화 작업을 수행 혹은 해제합니다.

- 컴퓨터 이름을 설정합니다.
- 새로운 월페이퍼(wallpaper)를 설정합니다.
- DNS 접미사 목록을 추가합니다.
- 부팅 볼륨 크기를 확장합니다.
- 관리자 암호를 설정합니다.

초기화 설정을 구성하려면

1. 구성할 인스턴스에서 텍스트 편집기에 C:\ProgramData\Amazon\EC2-Windows\Launch\Config\LaunchConfig.json 파일을 엽니다.
2. 다음 설정을 필요에 따라 업데이트하고 변경 내용을 저장합니다. adminPasswordType이 Specify인 경우에만 adminPassword에 암호를 입력합니다.

```
{  
    "setComputerName": false,  
    "setWallpaper": true,  
    "addDnsSuffixList": true,  
    "extendBootVolumeSize": true,  
    "handleUserData": true,  
    "adminPasswordType": "Random | Specify | DoNothing",  
    "adminPassword": "password that adheres to your security policy (optional)"  
}
```

암호 유형은 다음과 같이 정의됩니다.

Random

EC2Launch는 암호를 생성하고 사용자의 키를 사용하여 암호를 암호화합니다. 인스턴스가 재부팅 또는 중지되었다가 시작된 경우 이 암호가 그대로 유지되도록 시스템은 인스턴스가 시작된 후 이 설정을 비활성화합니다.

Specify

adminPassword에 지정한 암호가 EC2Launch에 사용됩니다. 암호가 시스템 요구 사항에 맞지 않으면 EC2Launch에서 임의의 암호를 대신 생성합니다. 암호는 LaunchConfig.json에 일반 텍스트로 저장되며 Sysprep에서 관리자 암호를 설정한 후에 삭제됩니다. EC2Launch는 사용자의 키를 사용하여 암호를 암호화합니다.

DoNothing

unattend.xml 파일에 지정한 암호가 EC2Launch에 사용됩니다. unattend.xml에 암호를 지정하지 않으면 관리자 계정이 비활성화됩니다.

3. Windows PowerShell에서 다음 명령을 실행하여 스크립트가 Windows 예약된 작업으로 실행되도록 예약합니다. 스크립트는 다음 부팅 중에 이 작업을 한 번 실행한 다음 이 작업이 다시 실행되지 않도록 비활성화합니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -  
Schedule
```

매 부팅마다 실행하도록 EC2Launch 예약

처음 부팅할 때만이 아니라 매 부팅마다 실행하도록 EC2Launch를 예약할 수 있습니다.

매 부팅마다 실행하도록 EC2Launch를 활성화하는 방법:

1. Windows PowerShell을 열어 다음 명령을 실행합니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -  
SchedulePerBoot
```

2. 또는 다음 명령을 사용하여 실행 파일을 실행합니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Settings\Ec2LaunchSettings.exe
```

그런 다음 Run EC2Launch on every boot를 선택합니다. EC2 인스턴스 Shutdown without Sysprep 또는 Shutdown with Sysprep을 지정할 수 있습니다.

Note

매 부팅마다 실행하도록 EC2Launch를 활성화할 경우 다음 번에 EC2Launch를 실행할 때 LaunchConfig.json이 다음과 같이 변경됩니다.

- 매 부팅마다 암호가 변경되지 않도록 AdminPasswordType이 DoNothing으로 다시 설정됩니다.
- 사용자 데이터에서 persist를 true로 설정한 경우가 아니면 HandleUserData가 false로 다시 설정됩니다. 사용자 데이터 스크립트에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [사용자 데이터 스크립트](#)를 참조하십시오.

이와 마찬가지로, 다음 부팅 시 암호를 리셋하지 않으려면 재부팅 전에 AdminPasswordType을 DoNothing으로 설정해야 합니다.

드라이브 및 맵 드라이브 문자 초기화

DriveLetterMappingConfig.json 파일에서 설정을 지정하여 드라이브를 초기화 및 포맷하고 드라이브 문자를 EC2 인스턴스의 볼륨에 매핑합니다. 이 스크립트는 드라이브가 아직 초기화 및 분할되지 않은 경우에 이 작업을 수행합니다.

드라이브 문자를 볼륨에 매핑하려면

1. 텍스트 에디터에서 C:\ProgramData\Amazon\EC2-Windows\Launch\Config\DriveLetterMappingConfig.json 파일을 엽니다.
2. 다음 볼륨 설정을 지정하고 변경 내용을 저장합니다.

```
{  
  "driveLetterMapping": [  
    {  
      "volumeName": "sample volume",  
      "driveLetter": "H"  
    }  
  ]  
}
```

3. 디스크를 초기화할 EC2Launch 스크립트를 수행하기 위해서 Windows PowerShell을 열고 다음 명령을 실행합니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1
```

인스턴스를 부팅할 때마다 디스크를 초기화하려면 다음과 같이 -Schedule 플래그를 추가하십시오.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1 -Schedule
```

EC2 콘솔에 Windows 이벤트 로그 전송

EventLogConfig.json 구성 파일에서 설정을 지정하여 Windows Event 로그를 EC2 콘솔 로그에 전송합니다.

Windows Event 로그를 전송하도록 설정을 구성하려면

1. 인스턴스의 텍스트 편집기에서 C:\ProgramData\Amazon\EC2-Windows\Launch\Config\EventLogConfig.json 파일을 엽니다.
2. 다음 로그 설정을 구성하고 변경 내용을 저장합니다.

```
{  
  "events": [  
    {  
      "logName": "System",  
      "source": "An event source (optional)",  
      "level": "Error | Warning | Information",  
      "numEntries": 3  
    }  
  ]  
}
```

3. Windows PowerShell에서 다음 명령을 실행합니다. 그러면 시스템에서 인스턴스가 부팅될 때마다 스크립트가 Windows 예약된 작업으로 실행되도록 예약됩니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\SendEventLogs.ps1 -Schedule
```

로그가 EC2 콘솔 로그에 나타나려면 3분 이상 걸릴 수 있습니다.

부팅 성공 후 Windows is ready 메시지 전송

EC2Config 서비스는 매번 부팅 후에 "Windows is ready" 메시지를 EC2 콘솔에 전송했습니다. EC2Launch는 초기 부팅 후에만 이 메시지를 전송합니다. EC2Config 서비스와 역방향 호환성을 위해 매번 부팅한 후에 이 메시지를 전송하도록 EC2Launch를 예약할 수 있습니다. 인스턴스에서 Windows PowerShell을 열고 다음 명령을 실행합니다. 시스템에서 이 스크립트는 Windows 예약된 작업으로 실행되도록 예약됩니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\SendWindowsIsReady.ps1 -Schedule
```

EC2Launch 버전 기록

Windows Server 2016으로 시작하는 Windows AMI에는 EC2Launch라는 Windows Powershell 스크립트 세트가 포함되어 있습니다. EC2Launch는 초기 인스턴스 부팅 중에 작업을 수행합니다. Windows AMI에 포함된 EC2Launch 버전에 대한 자세한 내용은 [관리형 AWS Windows AMI \(p. 26\)](#) 단원을 참조하십시오.

최신 버전의 EC2Launch를 다운로드하여 설치하려면 최신 버전의 [EC2Launch 설치 \(p. 490\)](#) 단원을 참조하십시오.

다음 표에서는 EC2Launch의 릴리스 버전에 대해 설명합니다. 버전 형식은 버전 1.3.610 이후에 변경되었습니다.

버전	세부 정보	릴리스 날짜
1.3.2003155	인스턴스 유형 정보가 업데이트되었습니다.	2020년 8월 25일
1.3.2003150	OsCurrentBuild 및 OsReleaseId가 콘솔 출력에 추가되었습니다.	2020년 4월 22일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EC2Launch

버전	세부 정보	릴리스 날짜
1.3.2003040	IMDS 버전 1 대체 로직이 수정했습니다.	2020년 4월 7일
1.3.2002730	IMDS V2에 대한 지원이 추가되었습니다.	2020년 3월 3일
1.3.2002240	사소한 문제를 수정했습니다.	2019년 10월 31일
1.3.2001660	처음 Sysprep을 실행한 후 암호가 없는 사용자의 자동 로그인 문제를 수정했습니다.	2019년 7월 2일
1.3.2001360	사소한 문제를 수정했습니다.	2019년 3월 27일
1.3.2001220	모든 PowerShell 스크립트는 서명이 된 것입니다.	2019년 2월 28일
1.3.2001200	Microsoft Windows Server Failover Cluster이 노드 상에서 스크립트를 실행하는 InitializeDisks.ps1을 통해 문제를 수정하면 드라이버 문자가 로컬 드라이버 문자와 일치하는 원격 노드 상에서 드라이버가 포맷됩니다.	2019년 2월 27일
1.3.2001160	Windows 2019에서 월페이퍼(wallpaper) 누락 문제를 수정했습니다.	2019년 2월 22일
1.3.2001040	<ul style="list-style-type: none"> ACPI 문제 해결을 위해 모니터가 절대로 꺼지지 않게 설정하는 플러그인을 추가했습니다. SQL Server 에디션과 버전을 콘솔에 기록했습니다. 	2019년 1월 21일
1.3.2000930	ipv6 사용 ENI에서 메타데이터로 가는 경로를 추가하는 수정사항이 적용되었습니다.	2019년 1월 2일
1.3.2000760	<ul style="list-style-type: none"> RSS에 대한 기본 구성과 ENA 디바이스에 대한 수신 대기열 설정을 추가했습니다. Sysprep 도중 최대 절전 모드를 비활성화했습니다. 	2018년 12월 5일
1.3.2000630	<ul style="list-style-type: none"> DNS 서버에 대한 라우트 169.254.169.253/32를 추가했습니다. 관리자 사용자 설정 필터를 추가했습니다. 인스턴스 최대 절전 모드를 개선했습니다. 매 부팅마다 실행하도록 EC2Launch를 예약하는 옵션을 추가했습니다. 	2018년 11월 9일
1.3.2000430.0	<ul style="list-style-type: none"> AMZN 시간 서비스에 라우트 169.254.169.123/32 추가. GRID 라이선스 서비스에 라우트 169.254.169.249/32 추가. 시스템 관리자 시작 시도 시 25초의 시간 제한 추가. 	2018년 9월 19일
1.3.200039.0	<ul style="list-style-type: none"> EBS NVME 볼륨의 부적절한 드라이브 문자 수정. NVME 드라이버 버전에 대한 추가 로깅 추가. 	2018년 8월 15일
1.3.2000080	사소한 문제를 수정했습니다.	
1.3.610	사용자 데이터에서 파일로 출력 및 오류 리디렉션 문제를 수정했습니다.	
1.3.590	<ul style="list-style-type: none"> 월페이퍼(wallpaper)에 누락된 인스턴스 유형을 추가했습니다. 드라이브 문자 매핑 및 디스크 설치 문제를 수정했습니다. 	

버전	세부 정보	릴리스 날짜
1.3.580	<ul style="list-style-type: none"> 웹 요청에 기본 시스템 프록시 설정을 사용하도록 Get-Metadata를 수정했습니다. 디스크 설치에서 NVMe 특수 사례를 추가했습니다. 사소한 문제를 수정했습니다. 	
1.3.550	종료 없이 Sysprep을 활성화하는 -NoShutdown 옵션을 추가했습니다.	
1.3.540	사소한 문제를 수정했습니다.	
1.3.530	사소한 문제를 수정했습니다.	
1.3.521	사소한 문제를 수정했습니다.	
1.3.0	<ul style="list-style-type: none"> 컴퓨터 이름 변경에 발생한 16진수 길이 문제를 수정했습니다. 컴퓨터 이름 변경에 발생 가능한 재부팅 루프를 수정했습니다. 월페이퍼(wallpaper) 설정 문제를 수정했습니다. 	
1.2.0	<ul style="list-style-type: none"> 설치된 운영 체제(OS)에 대한 정보가 EC2 시스템 로그에 표시되도록 업데이트합니다. EC2 시스템 로그에 EC2Launch 및 SSM 에이전트 버전이 표시되도록 업데이트합니다. 사소한 문제를 수정했습니다. 	
1.1.2	<ul style="list-style-type: none"> EC2 시스템 로그에 ENA 드라이버 정보가 표시되도록 업데이트합니다. 기본 NIC 필터 로직에서 Hyper-V를 제외하도록 업데이트합니다. KMS 정품 인증을 위한 레지스트리 키에 KMS 서버와 포트를 추가했습니다. 여러 사용자에 대한 월페이퍼(wallpaper)를 개선했습니다. 영구 저장소에서 경로를 지우도록 업데이트합니다. DNS 접미사 목록의 가용 영역에서 z를 제거하도록 업데이트합니다. 사용자 데이터에서 <runAsLocalSystem> 태그로 문제를 해결하도록 업데이트합니다. 	
1.1.1	최초 릴리스.	

EC2Config 서비스를 사용한 Windows 인스턴스 구성

지원되는 모든 Windows Server 버전에 대한 최신 시작 서비스는 EC2Config와 EC2Launch를 모두 대체하는 [EC2Launch v2 \(p. 461\)](#)입니다.

Windows Server 2012 R2 및 그 이전 Windows AMI는 EC2Config 서비스(EC2Config.exe)라는 선택적 서비스를 포함합니다. 시작 중에 인스턴스가 부팅되고 작업을 수행할 때와 인스턴스를 중지하거나 시작할 때마다 EC2Config가 시작됩니다. 또한 EC2Config는 요청 시 작업을 수행할 수 있습니다. 이러한 작업 중 일부는 자동으로 활성화되고, 나머지는 수동으로 활성화해야 합니다. 선택 사항이긴 하지만, 이 서비스는 다른 방식으로는 사용할 수 없는 고급 기능에 대한 액세스를 제공합니다. 이 서비스는 LocalSystem 계정에서 실행됩니다.

Note

EC2Launch는 Windows Server 2016 이상의 Windows AMI에 있는 EC2Config를 대체합니다. 자세한 내용은 [EC2Launch를 사용한 Windows 인스턴스 구성 \(p. 489\)](#) 단원을 참조하십시오. 지원되는 모든 Windows Server 버전에 대한 최신 시작 서비스는 EC2Config와 EC2Launch를 모두 대체하는 [EC2Launch v2 \(p. 461\)](#)입니다.

EC2Config에서 설정 파일을 사용하여 해당 작업을 제어합니다. 이러한 설정 파일은 그래픽 도구를 사용하거나 XML 파일을 직접 편집하여 업데이트할 수 있습니다. 서비스 바이너리 및 추가 파일은 %ProgramFiles%\Amazon\EC2ConfigService 디렉터리에 저장됩니다.

내용

- [EC2Config 작업 \(p. 497\)](#)
- [최신 EC2Config 설치 \(p. 498\)](#)
- [EC2Config 중단, 재시작, 삭제, 설치 제거 \(p. 499\)](#)
- [EC2Config 및 AWS 시스템 관리자 \(p. 500\)](#)
- [EC2Config 및 Sysprep \(p. 500\)](#)
- [EC2 서비스 속성 \(p. 500\)](#)
- [EC2Config 설정 파일 \(p. 504\)](#)
- [EC2Config 서비스의 프록시 설정 구성 \(p. 508\)](#)
- [EC2Config 버전 기록 \(p. 509\)](#)
- [EC2Config 서비스와 관련된 문제 해결 \(p. 521\)](#)

EC2Config 작업

EC2Config는 인스턴스가 처음 시작할 때 초기 시작 작업을 실행하고 이후에 그 작업을 비활성화합니다. 이런 작업을 다시 실행하기 위해서는 인스턴스를 종료하기 전에 이를 명시적으로 활성화하거나 Sysprep을 수동으로 실행하는 방법을 사용해야 합니다. 이 작업은 다음과 같습니다.

- 관리자 계정에 대한 무작위의 암호화된 암호 설정.
- Remote Desktop Connection(원격 데스크톱 연결)에 사용되는 호스트 인증서의 생성 및 설치.
- 파티션 처리되지 않은 공간을 포함시키기 위한 운영 시스템 파티션을 동적으로 확장.
- 지정된 사용자 데이터 실행(설치된 경우는 Cloud-Init도 실행). 사용자 데이터 지정에 대한 자세한 내용은 [인스턴스 사용자 데이터로 작업 \(p. 589\)](#) 단원을 참조하십시오.

EC2Config는 다음 작업을 인스턴스가 시작할 때마다 실행합니다.

- 16진수 표기법으로 프라이빗 IP 주소와 일치하도록 호스트 이름 변경(이 작업은 기본적으로 비활성화되어 있으며 인스턴스가 시작할 때 실행하려면 활성화해야 함)
- AWS KMS(핵심 관리 서버) 구성, Windows 정품 인증 상태 확인, 필요한 경우 Windows 정품 인증을 활성화.
- 모든 Amazon EBS 볼륨 및 인스턴스 스토어 볼륨을 마운트하고 볼륨 이름을 드라이브 문자로 매핑합니다.
- 문제 해결을 돋기 위해 이벤트 로그 항목을 콘솔에 기록(이 작업은 기본적으로 비활성화되어 있으며 인스턴스가 시작할 때 실행될 수 있도록 활성화되어야 함)
- Windows가 준비된 상태를 콘솔에 기록.
- 복수 NIC가 연결될 때 다음 IP 주소(169.254.169.250, 169.254.169.251, 169.254.169.254)를 활성화하기 위해 기본 네트워크 어댑터에 사용자 지정 라우팅 추가. 이 주소들은 Windows 정품 인증을 할 때와 인스턴스 메타데이터에 액세스할 때 사용됩니다.

EC2Config는 다음 작업을 사용자가 로그인할 때마다 실행합니다.

- 바탕 화면 배경에 월페이퍼(wallpaper) 정보 표시

인스턴스가 실행 중일 때 사용자는 EC2Config가 다음 작업을 필요할 때 수행하도록 요청할 수 있습니다.

- AMI를 인스턴스에서 생성할 수 있도록 Sysprep을 실행하고 인스턴스를 종료. 자세한 내용은 [Sysprep을 이용한 표준 Amazon 머신 이미지 생성 \(p. 87\)](#) 단원을 참조하십시오.

최신 EC2Config 설치

기본적으로 EC2Config 서비스는 Windows Server 2016 이전의 AMI에 포함되어 있습니다. EC2Config 서비스가 업데이트되면 새 AWS Windows AMI에 최신 버전의 서비스가 포함됩니다. 그러나 사용자 자신의 Windows AMI 및 인스턴스는 별도로 최신 버전의 EC2Config로 업데이트해야 합니다.

Note

EC2Launch는 Windows Server 2016 이상 AMI의 EC2Config를 대체합니다. 자세한 내용은 [EC2Launch를 사용한 Windows 인스턴스 구성 \(p. 489\)](#) 단원을 참조하십시오. 지원되는 모든 Windows Server 버전에 대한 최신 시작 서비스는 EC2Config와 EC2Launch를 모두 대체하는 [EC2Launch v2 \(p. 461\)](#)입니다.

EC2Config 업데이트 알림을 받는 방법에 대한 자세한 내용은 [EC2Config 서비스 알림 받기 \(p. 521\)](#) 단원을 참조하십시오. 각 버전의 변경 사항에 대한 자세한 내용은 [EC2Config 버전 기록 \(p. 509\)](#) 단원을 참조하십시오.

시작하기 전에

- .NET framework 3.5 SP1 이상이 설치되어 있는지 확인합니다.
- 기본적으로 설치는 사용자의 설정 파일을 기본 설정 파일로 교체하고 설치가 완료되면 EC2Config 서비스를 재시작합니다. EC2Config 서비스 설정을 변경한 경우 config.xml 디렉터리에서 %Program Files%\Amazon\Ec2ConfigService\Settings 파일을 복사합니다. EC2Config 서비스를 업데이트한 후 이 파일을 복원하여 구성 변경을 유지할 수 있습니다.
- EC2Config 버전이 2.1.19 이전 버전이고 2.2.12 이전 버전을 설치하는 경우 먼저 2.1.19 버전을 설치해야 합니다. 2.1.19 버전을 설치하려면 [EC2Install_2.1.19.zip](#) 파일을 다운로드하고 압축을 해제한 후 EC2Install.exe 파일을 실행합니다.

Note

EC2Config 버전이 2.1.19 이전 버전이고 2.3.313 이후 버전을 설치하는 경우 먼저 버전 2.1.19를 설치하지 않고 2.3.313 이후 버전을 직접 설치할 수 있습니다.

EC2Config 버전 확인

다음 절차를 사용하여 인스턴스에 설치된 EC2Config의 버전을 확인합니다.

설치된 EC2Config 버전을 확인하려면

1. AMI에서 인스턴스를 실행해서 여기에 연결합니다.
2. 제어판에서 프로그램 및 기능을 선택합니다.
3. 설치된 프로그램 목록에서 Ec2ConfigService를 찾습니다. 버전 번호가 버전 열에 표시됩니다.

EC2Config 업데이트

다음 절차를 사용하여 인스턴스에 최신 버전의 EC2Config를 다운로드하고 설치합니다.

최신 버전의 EC2Config를 다운로드하고 설치하려면

1. [EC2Config 설치 관리자](#)를 다운로드하고 압축을 풁니다.
2. 실행 `EC2Install.exe`. 전체 옵션 목록을 보려면 `EC2Install` 옵션을 포함해 `/?` 파일을 실행합니다. 기본적으로 설치는 프롬프트를 표시합니다. 프롬프트 없이 명령을 실행하려면 `/quiet` 옵션을 사용합니다.

Important

저장한 `config.xml` 파일의 사용자 지정 설정을 그대로 유지하려면 `EC2Install` 옵션으로 `/norestart` 파일을 실행하고 사용자의 설정을 복원한 다음에 EC2Config 서비스를 수동으로 재시작합니다.

3. EC2Config 버전 4.0 이상을 실행하는 경우 Microsoft Services 스냅인 인스턴스에서 SSM 에이전트를 재시작해야 합니다.

Note

인스턴스를 재부팅하거나 중지하고 시작할 때까지 업데이트된 EC2Config 버전 정보는 인스턴스 시스템 로그 또는 Trusted Advisor 검사에 나타나지 않습니다.

EC2Config 중단, 재시작, 삭제, 설치 제거

EC2Config 서비스는 다른 서비스와 마찬가지 방식으로 관리할 수 있습니다.

인스턴스에 업데이트된 설정을 적용하려면 서비스를 중단한 후에 재시작해야 합니다. EC2Config를 수동으로 설치하는 경우는 서비스를 먼저 중단해야 합니다.

EC2Config 서비스 중단 방법

1. 실행을 시작해서 Windows 인스턴스에 연결합니다.
2. 시작 메뉴에서 관리 도구를 가리킨 다음에 서비스를 클릭합니다.
3. 서비스 목록에서 EC2Config를 오른쪽 클릭하고 중지를 선택합니다.

EC2Config 서비스 재시작 방법

1. 실행을 시작해서 Windows 인스턴스에 연결합니다.
2. 시작 메뉴에서 관리 도구를 가리킨 다음에 서비스를 클릭합니다.
3. 서비스 목록에서 EC2Config를 오른쪽 클릭하고 재시작을 클릭합니다.

구성 설정을 업데이트하거나, 자체 AMI를 생성하거나, AWS 시스템 관리자를 사용할 필요가 없는 경우에는 서비스를 삭제하고 제거할 수 있습니다. 서비스를 삭제하면 등록 서브키도 제거됩니다. 서비스를 설치 제거하면 파일, 등록 서브키, 서비스 바로가기도 제거됩니다.

EC2Config 서비스 삭제 방법

1. 명령 프롬프트 창을 시작합니다.
2. 다음 명령을 실행합니다.

```
sc delete ec2config
```

EC2Config 설치 제거 방법

1. 실행을 시작해서 Windows 인스턴스에 연결합니다.

2. 시작 메뉴에서 제어판을 클릭합니다.
3. 프로그램 및 기능을 두 번 클릭합니다.
4. 프로그램 목록에서 EC2ConfigService를 선택하고 설치 제거를 클릭합니다.

EC2Config 및 AWS 시스템 관리자

EC2Config 서비스는 2016년 11월 이전에 발표된 Windows Server 2016 이전 버전의 Windows Server용 AMI에서 생성된 인스턴스에 대한 시스템 관리자 요청을 처리합니다.

2016년 11월 이후에 발표된 Windows Server 2016 이전 버전의 Windows Server용 AMI에서 생성된 인스턴스에는 EC2Config 서비스 및 SSM 에이전트가 포함되어 있습니다. EC2Config는 앞서 설명한 모든 작업을 수행하고 SSM 에이전트는 명령 실행 및 상태 관리자 같은 시스템 관리자 기능에 대한 요청을 처리합니다.

Run Command를 사용하여 기존 인스턴스가 최신 버전의 EC2Config 서비스와 SSM 에이전트를 사용하도록 업그레이드할 수 있습니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서에서 [Run Command를 사용하여 SSM 에이전트 업데이트](#)를 참조하십시오.

EC2Config 및 Sysprep

EC2Config 서비스에서는 재사용할 수 있는 사용자 정의된 Windows AMI를 생성하는 데 사용할 수 있는 Microsoft 도구인 Sysprep을 실행합니다. EC2Config가 Sysprep을 호출하면 Sysprep은 %ProgramFiles%\Amazon\EC2ConfigService\Settings 안의 파일을 사용하여 어느 작업을 수행할지 결정합니다. 이러한 파일은 Ec2 서비스 속성(Ec2 Service Properties) 대화 상자를 사용하여 간접적으로 편집하거나, XML 편집기 또는 텍스트 편집기를 사용하여 직접 편집할 수 있습니다. 그러나 Ec2 서비스 속성(Ec2 Service Properties) 대화 상자에서 사용할 수 없는 몇 가지 고급 설정이 있으며, 이러한 항목은 직접 편집해야 합니다.

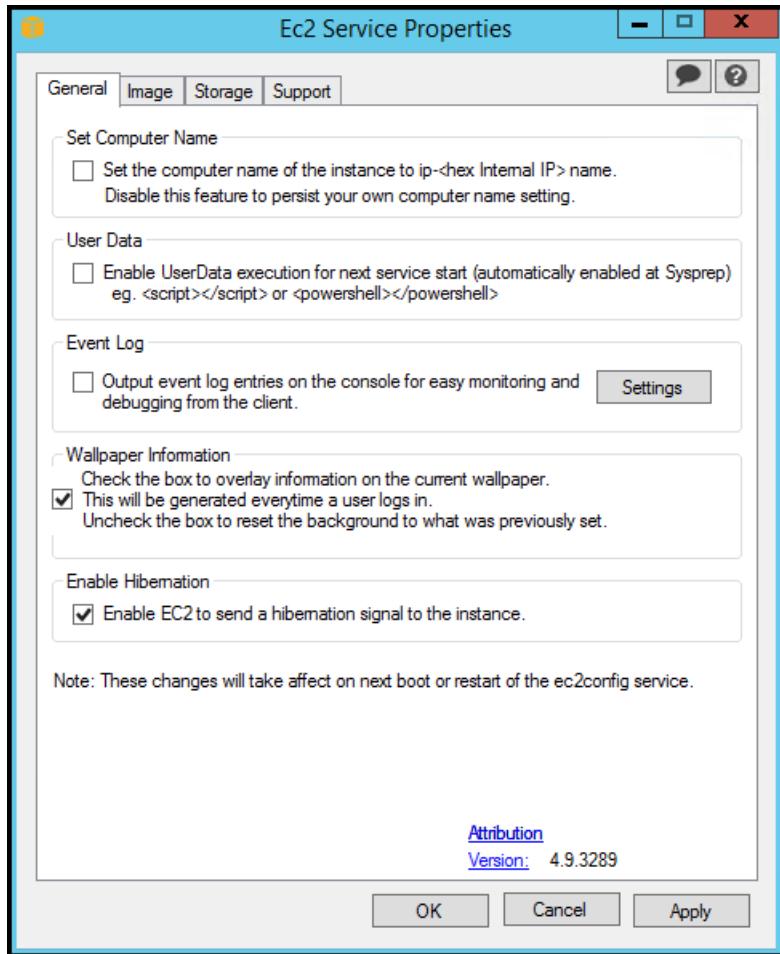
설정을 업데이트한 후에 인스턴스에서 AMI를 생성하면 새로운 설정은 그 새로운 AMI에서 실행하는 모든 인스턴스에 적용됩니다. AMI 생성에 대한 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오.

EC2 서비스 속성

다음 절차는 Ec2 서비스 속성(Ec2 Service Properties) 대화 상자를 사용해서 설정을 활성화 또는 비활성화하는 방법을 설명합니다.

Ec2 서비스 속성(Ec2 Service Properties) 대화 상자를 사용한 설정 변경 방법

1. 실행을 시작해서 Windows 인스턴스에 연결합니다.
2. 시작 메뉴에서 모든 프로그램을 클릭하고 EC2ConfigService Settings를 클릭합니다.



3. Ec2 서비스 속성(Ec2 Service Properties) 대화 상자의 일반(General) 탭에서 다음 설정을 활성화 또는 비활성화 할 수 있습니다.

컴퓨터 이름 설정

이 설정이 활성화된 경우(기본 설정상 비활성화되어 있음), 매 부팅마다 호스트 이름은 현재 IP 주소와 비교됩니다. 호스트 이름과 IP 주소가 일치하지 않는 경우, 호스트 이름은 내부 IP 주소를 포함하도록 재설정되며 이후 시스템은 재부팅되어 새로운 호스트 이름을 가지게 됩니다. 자신의 호스트 이름을 설정하거나 기존 호스트 이름이 변경되는 것을 방지하려면 이 설정을 활성화하지 마십시오.

사용자 데이터

사용자 데이터 실행을 통해 인스턴스 메타데이터에서 스크립트를 지정할 수 있습니다. 기본적으로 이러한 스크립트는 초기 실행 중에 실행됩니다. 또한 다음 번에 인스턴스를 재부팅하거나 시작할 때 또는 매번 인스턴스를 재부팅하거나 시작할 때마다 실행하도록 구성할 수도 있습니다.

크기가 큰 스크립트가 있는 경우, 사용자 데이터를 사용해서 스크립트를 다운로드하고 이를 실행하는 것이 권장됩니다.

자세한 내용은 [사용자 데이터 실행 \(p. 570\)](#) 단원을 참조하십시오.

Event Log(이벤트 로그)

이 설정을 사용해서 모니터링 및 디버깅을 손쉽게 할 수 있도록 부팅 동안 콘솔의 이벤트 로그 항목을 표시합니다.

설정을 클릭해서 콘솔에 전송되는 로그 항목에 대한 필터를 지정합니다. 기본 필터는 시스템 이벤트 로그에서 콘솔로 가장 최근의 오류 항목 3개를 전송합니다.

월페이퍼 정보

이 설정을 사용해서 바탕 화면 배경에 시스템 정보를 표시합니다. 다음은 바탕 화면 배경에 표시되는 정보의 예입니다.

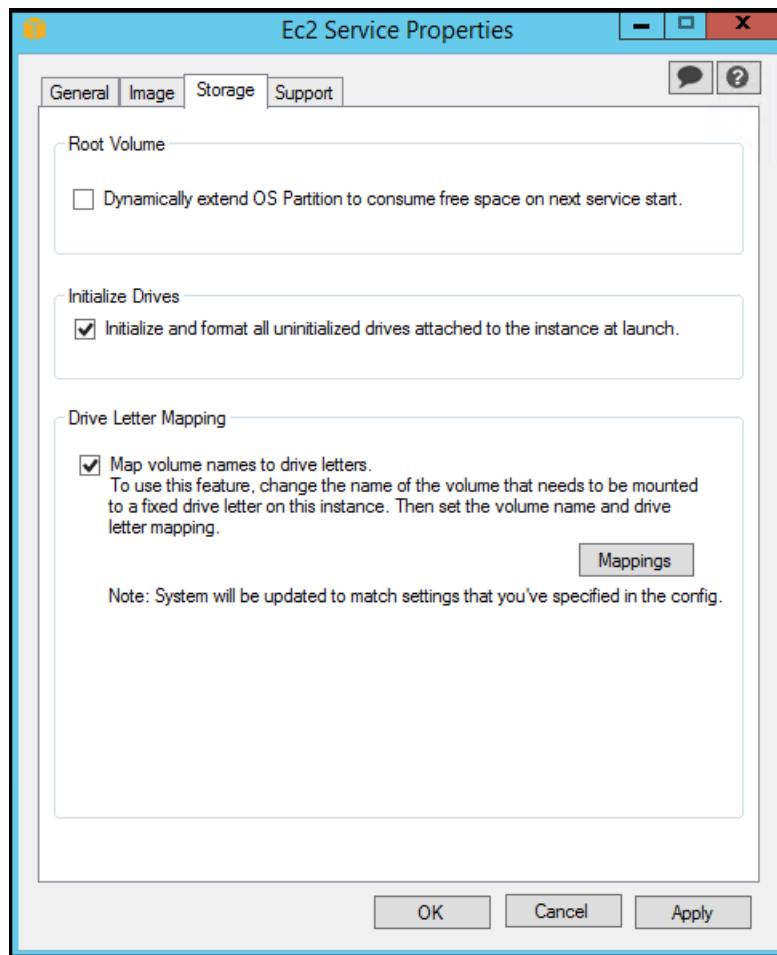
```
Hostname      : WIN-U0RFOJCTPUU
Instance ID   : i-d583f76a
Public IP Address : 54.208.43.227
Private IP Address : 172.31.42.195
Availability Zone : us-east-1b
Instance Size   : t2.micro
Architecture    : AMD64
```

바탕 화면 배경에 표시되는 정보는 설정 파일 EC2ConfigService\Settings\WallpaperSettings.xml에 의해 제어됩니다.

최대 절전 모드 활성화

EC2에서 최대 절전 모드를 수행하도록 운영 체제에 신호를 보낼 수 있도록 하려면 이 설정을 사용 합니다.

4. 스토리지 탭을 클릭합니다. 다음 설정을 활성화 또는 비활성화할 수 있습니다.



Root Volume(루트 볼륨)

이 설정은 파티션 처리되지 않은 공간을 모두 포함하도록 동적으로 디스크 0/볼륨 0을 확장합니다.
이는 인스턴스가 사용자 설정 크기를 가진 루트 디바이스 볼륨에서 부팅될 때 유용할 수 있습니다.

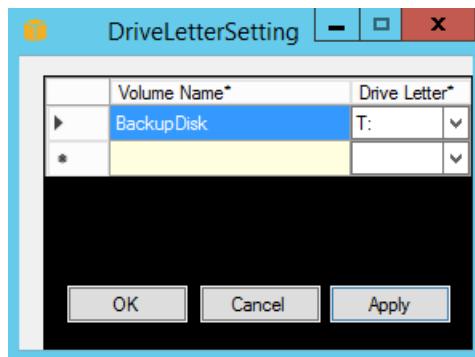
드라이브 초기화

이 설정은 시작 동안 인스턴스에 연결된 모든 볼륨을 포맷하고 마운트합니다.

드라이브 문자 매핑

시스템은 인스턴스에 연결된 볼륨을 드라이브 문자로 매핑합니다. Amazon EBS 볼륨의 경우, 기본 설정은 D:부터 Z:까지 드라이브 문자를 배정합니다. 인스턴스 스토어 볼륨의 경우, 기본 설정은 드라이버에 따라 달라집니다. Citrix PV 드라이버는 인스턴스 스토어 볼륨에 D:부터 Z:까지 드라이브 문자를 배정합니다. Red Hat 드라이버는 인스턴스 스토어 볼륨에 D:부터 Z:까지 드라이브 문자를 배정합니다.

볼륨에 대한 드라이브 문자를 선택하려면 매핑(Mappings)을 클릭합니다. DriveLetterSetting 대화 상자에서 각 볼륨에 대한 볼륨 이름(Volume Name)과 드라이브 문자(Drive Letter) 값을 지정하고 적용을 클릭한 후 확인을 클릭합니다. 사용할 가능성 있는 드라이브 문자와 충돌하지 않는 드라이브 문자(예: 알파벳 중간 위치 정도에 있는 문자)를 선택하는 것이 권장됩니다.



드라이브 문자 매핑을 지정하고 지정한 볼륨 이름과 동일한 라벨에 볼륨을 연결합니다. 그러나 드라이브 문자가 이미 사용 중인 경우 드라이브 문자 매핑은 실패하게 됩니다. 드라이브 문자 매핑을 지정한 경우 EC2Config는 이미 마운트된 볼륨의 드라이브 문자를 변경하지는 않는다는 점을 알아두시기 바랍니다.

- 설정을 저장하고 이후에 작업을 계속하려면 확인을 클릭해서 Ec2 서비스 속성(Ec2 Service Properties) 대화 상자를 닫습니다. 인스턴스의 사용자 지정을 완료하고 그 인스턴스에서 AMI를 생성하고 싶다면 [Sysprep을 이용한 표준 Amazon 머신 이미지 생성 \(p. 87\)](#) 단원을 참조하십시오.

EC2Config 설정 파일

본 설정 파일은 EC2Config 서비스의 작동을 제어합니다. 이 파일은 C:\Program Files\Amazon\Ec2ConfigService\Settings 디렉터리에 위치합니다.

- ActivationSettings.xml - KMS(핵심 관리 서버)를 사용해서 제품의 정품 인증을 제어합니다.
- AWS.EC2.Windows.CloudWatch.json - CloudWatch에 전송할 성능 카운터 및 CloudWatch Logs에 전송할 로그가 무엇인지를 제어합니다.
- BundleConfig.xml - EC2Config가 AMI 생성을 위한 인스턴스 스토어 지원 인스턴스의 준비 방법을 제어합니다.
- Config.xml - 기본 설정을 제어합니다.
- DriveLetterConfig.xml - 드라이브 문자 매핑을 제어합니다.
- EventLogConfig.xml - 인스턴스가 부팅 중인 동안 콘솔에 표시되는 이벤트 로그 정보를 제어합니다.
- WallpaperSettings.xml - 바탕 화면 배경에 표시되는 정보를 제어합니다.

ActivationSettings.xml

이 파일은 제품 정품 인증을 제어하는 설정을 포함하고 있습니다. Windows가 부팅될 때 EC2Config 서비스는 Windows가 이미 정품 인증되었는지를 검사합니다. Windows가 아직 정품 인증되지 않은 경우, EC2Config 서비스는 지정된 KMS 서버를 검색해서 Windows 정품 인증을 시도합니다.

- SetAutodiscover - KMS를 자동으로 감지할지 여부를 지정합니다.
- TargetKMSServer - KMS의 프라이빗 IP 주소를 저장합니다. KMS는 인스턴스와 동일한 리전에 위치해야 합니다.
- DiscoverFromZone - 지정된 DNS 영역에서 KMS 서버를 발견합니다.
- ReadFromUserData - UserData에서 KMS 서버를 가져옵니다.
- LegacySearchZones - 지정된 DNS 영역에서 KMS 서버를 발견합니다.
- DoActivate - 해당 섹션의 지정된 설정을 사용해서 정품 인증을 시도합니다. 이 값은 true 또는 false일 수 있습니다.
- LogResultToConsole - 결과를 콘솔에 표시합니다.

BundleConfig.xml

이 파일은 EC2Config가 AMI 생성을 위한 인스턴스 스토어 지원 인스턴스의 준비 방법을 제어하는 설정을 포함하고 있습니다.

- AutoSysprep - Sysprep을 자동적으로 사용할지 여부를 지정합니다. Yes로 값을 변경해서 Sysprep을 사용하도록 합니다.
- SetRDPCertificate - 자체 서명된 인증서를 원격 데스크톱 서버로 설정합니다. 이를 통해 안전하게 RDP를 인스턴스에 연결시킬 수 있습니다. 새 인스턴스가 인증서를 보유해야 하는 경우는 이 값을 Yes로 변경합니다.

Windows Server 2008 또는 Windows Server 2012 인스턴스는 자신 고유의 인증서를 생성할 수 있기 때문에 이 설정은 상기 인스턴스에서는 사용되지 않습니다.

- SetPasswordAfterSysprep - 새로 실행된 인스턴스에 무작위 암호를 설정하고 이를 사용자 실행 키로 암호화하고 암호화된 암호를 콘솔에 출력합니다. 새 인스턴스를 무작위의 암호화된 암호로 설정하지 않아야 하는 경우는 설정의 값을 No로 변경합니다.

Config.xml

플러그인

- Ec2SetPassword - 인스턴스를 실행할 때마다 무작위의 암호화된 암호를 생성합니다. 이 기능은 첫 실행 후 기본적으로 비활성화되어 해당 인스턴스의 재부팅으로 인해 사용자가 설정한 암호가 변경되지 않도록 합니다. 이 설정을 Enabled로 변경하면 인스턴스를 실행할 때마다 계속 새로운 암호를 생성합니다.

이 설정은 인스턴스에서 AMI를 생성하려는 경우 필요할 수 있습니다.

- Ec2SetComputerName - 인스턴스의 호스트 이름을 인스턴스의 IP 주소에 기반한 고유의 이름으로 변경하고 인스턴스를 재부팅합니다. 자신의 호스트 이름을 설정하거나 기존 호스트 이름이 변경되는 것을 방지하려면 이 설정을 비활성화해야 합니다.
- Ec2InitializeDrives - 시작 시에 모든 볼륨을 초기화하고 포맷합니다. 이 기능은 기본적으로 활성화되어 있습니다.
- Ec2EventLog - 콘솔에서 이벤트 로그 항목을 표시합니다. 기본적으로 시스템 이벤트 로그에서 가장 최근의 오류 항목 3개가 표시됩니다. 표시할 이벤트 로그 항목을 지정하려면 EventLogConfig.xml 디렉터리에 있는 EC2ConfigService\Settings 파일을 편집합니다. 아 파일의 설정에 대한 자세한 내용은 MSDN 라이브러리의 [Eventlog Key](#) 단원을 참조하십시오.
- Ec2ConfigureRDP - 인스턴스에 자체 서명된 인증서를 설정해서 사용자가 원격 데스크톱을 사용하여 안전하게 인스턴스에 액세스할 수 있도록 합니다. Windows Server 2008 및 Windows Server 2012 인스턴스는 자신 고유의 인증서를 생성할 수 있기 때문에 이 설정은 상기 인스턴스에서는 비활성화되어 있습니다.
- Ec2OutputRDPCert - 원격 데스크톱 인증서 정보를 사용자가 그 지문에 대해 검증할 수 있도록 상기 정보를 콘솔로 표시합니다.
- Ec2SetDriveLetter - 사용자가 지정한 설정에 기반하는 마운트된 볼륨의 드라이브 문자를 설정합니다. 기본적으로 Amazon EBS 볼륨이 인스턴스에 연결되면 인스턴스의 드라이브 문자를 사용하여 상기 볼륨을 마운트할 수 있습니다. 드라이브 문자 매핑을 지정하려면 DriveLetterConfig.xml에 있는 EC2ConfigService\Settings 파일을 편집합니다.
- Ec2WindowsActivate - 플러그인이 Windows 정품 인증을 처리합니다. Windows가 정품 인증되었는지 확인합니다. 인증되지 않은 경우 KMS 클라이언트 설정을 업데이트한 후 Windows를 정품 인증합니다.

KMS 설정을 변경하려면 ActivationSettings.xml 디렉터리에 위치한 EC2ConfigService\Settings 파일을 편집합니다.

- Ec2DynamicBootVolumeSize - 파티션 처리되지 않은 모든 공간을 포함하도록 디스크 0/볼륨 0을 확장합니다.
- Ec2HandleUserData - Sysprep이 실행된 후에 인스턴스를 처음 실행했을 때 사용자가 생성한 스크립트를 생성하여 실행시킵니다. 스크립트 태그에 둘러싸인 명령은 배치 파일로 저장되고, PowerShell 태그로 둘러싸인 명령은 .ps1 파일로 저장됩니다. 이는 Ec2 Service Properties(Ec2 서비스 속성) 대화 상자의 사용자 데이터 확인란에 해당되는 기능입니다.

- **Ec2ElasticGpuSetup** - 인스턴스가 탄력적 GPU와 연결되어 있으면 탄력적 GPU 소프트웨어 패키지를 설치합니다.
- **Ec2FeatureLogging** - Windows 기능 설치 및 이에 상응하는 서비스 상태를 콘솔로 전송합니다. Microsoft Hyper-V 기능 및 이에 상응하는 vmms 서비스에 대해서만 지원됩니다.

글로벌 설정

- **ManageShutdown** - 인스턴스 스토어 지원 AMI에서 실행한 인스턴스가 Sysprep 실행 중에 종료되지 않도록 합니다.
- **SetDnsSuffixList** - Amazon EC2에 대한 네트워크 어댑터의 DNS 접미사를 설정합니다. 이를 통해 정규화된 도메인 이름을 제공하지 않고도 Amazon EC2에서 실행 중인 서버의 DNS 확인을 할 수 있습니다.
- **WaitForMetaDataAvailable** - 부팅이 시작되기 전에 메타데이터의 액세스와 네트워크의 사용이 가능해질 때까지 EC2Config 서비스가 대기하도록 설정합니다. 이를 통해 EC2Config가 정품 인증 및 기타 플러그인에 대한 메타데이터를 얻을 수 있는지를 검사할 수 있습니다.
- **ShouldAddRoutes** - 복수 NIC가 연결될 때 다음 IP 주소(169.254.169.250, 169.254.169.251, 169.254.169.254)를 활성화하기 위해 기본 네트워크 어댑터에 사용자 지정 라우팅 추가. 이 주소들은 Windows 정품 인증을 할 때와 인스턴스 메타데이터에 액세스할 때 사용됩니다.
- **RemoveCredentialsFromSysprepOnStartup** - 다음에 서비스가 시작할 때 `Sysprep.xml`에서 관리자 암호를 제거합니다. 암호가 유지되도록 하려면 이 설정을 편집합니다.

DriveLetterConfig.xml

이 파일은 드라이브 문자 매핑을 제어하는 설정을 포함하고 있습니다. 기본적으로 볼륨은 사용 가능한 모든 드라이브 문자에 매핑할 수 있습니다. 다음과 같은 특정 드라이브에 볼륨을 마운트할 수 있습니다.

```
<?xml version="1.0" standalone="yes"?>
<DriveLetterMapping>
  <Mapping>
    <VolumeName></VolumeName>
    <DriveLetter></DriveLetter>
  </Mapping>
  . .
  <Mapping>
    <VolumeName></VolumeName>
    <DriveLetter></DriveLetter>
  </Mapping>
</DriveLetterMapping>
```

- **VolumeName** - 볼륨 레이블. 예: **My Volume**. 인스턴스 스토리지 볼륨에 대한 매핑을 지정하려면 라벨 **Temporary Storage x**를 사용합니다(x는 0-25 범위 내 숫자).
- **DriveLetter** - 드라이브 문자. 예: **M:**. 그러나 드라이브 문자가 이미 사용 중인 경우 드라이브 문자 매핑은 실패하게 됩니다.

EventLogConfig.xml

이 파일은 인스턴스가 부팅 중인 동안 콘솔에 표시되는 이벤트 로그 정보를 제어하는 설정을 포함하고 있습니다. 기본적으로 AWS는 시스템 이벤트 로그에서 가장 최근의 오류 항목 3개를 표시합니다.

- **Category** - 모니터링할 이벤트 로그 키.
- **ErrorType** - 이벤트 유형(예: Error, Warning, Information)
- **NumEntries** - 해당 카테고리에 대해 저장된 이벤트 수.
- **LastMessageTime** - 동일한 메시지가 반복적으로 푸시되는 것을 방지하기 위해 서비스는 이 값을 메시지를 푸시할 때마다 업데이트합니다.
- **AppName** - 이벤트를 기록한 이벤트 소스 또는 애플리케이션.

WallpaperSettings.xml

이 파일은 바탕 화면 배경에 표시되는 정보를 제어하는 설정을 포함하고 있습니다. 기본적으로 다음 정보가 표시됩니다.

- Hostname - 컴퓨터 이름을 표시합니다.
- Instance ID - 인스턴스의 ID를 표시합니다.
- Public IP Address - 인스턴스의 퍼블릭 IP 주소를 표시합니다.
- Private IP Address - 인스턴스의 프라이빗 IP 주소를 표시합니다.
- Availability Zone - 인스턴스가 실행 중인 가용 영역을 표시합니다.
- Instance Size - 인스턴스의 유형을 표시합니다.
- Architecture - PROCESSOR_ARCHITECTURE 환경 변수의 설정을 표시합니다.

기본적으로 표시되는 정보를 해당 항목을 삭제하여 제거할 수 있습니다. 다음과 같이 표시할 추가 인스턴스 메타데이터를 추가할 수 있습니다.

```
<WallpaperInformation>
  <name>display_name</name>
  <source>metadata</source>
  <identifier>meta-data/<path></identifier>
</WallpaperInformation>
```

다음과 같이 표시할 시스템 환경 변수를 추가할 수 있습니다.

```
<WallpaperInformation>
  <name>display_name</name>
  <source>EnvironmentVariable</source>
  <identifier>variable-name</identifier>
</WallpaperInformation>
```

InitializeDrivesSettings.xml

이 파일에는 EC2Config의 드라이브 초기화 방식을 제어하는 설정이 포함되어 있습니다.

기본적으로 EC2Config는 운영 체제를 통해 온라인 상태가 되지 않은 드라이브를 초기화합니다. 다음과 같이 플러그인을 사용자 지정할 수 있습니다.

```
<InitializeDrivesSettings>
  <SettingsGroup>setting</SettingsGroup>
</InitializeDrivesSettings>
```

설정 그룹을 사용하여 드라이브 초기화 방식을 지정합니다:

FormatWithTRIM

드라이브 포맷 시 TRIM 명령을 활성화합니다. 드라이브가 포맷되고 초기화된 후 시스템이 TRIM 구성을 복원합니다.

EC2Config 버전 3.18부터 기본적으로 디스크 포맷 작업 중에 TRIM 명령이 비활성화됩니다. 이를 통해 포맷 시간이 단축되었습니다. 이 설정을 사용하여 EC2Config 버전 3.18 이상에서 디스크 포맷 작업 중에 TRIM을 활성화합니다.

FormatWithoutTRIM

드라이브 포맷 시 TRIM 명령을 비활성화하고 Windows 포맷 시간을 단축합니다. 드라이브가 포맷되고 초기화된 후 시스템이 TRIM 구성을 복원합니다.

DisableInitializeDrives

새 드라이브의 포맷을 비활성화합니다. 이 설정을 사용하여 드라이브를 수동으로 초기화합니다.

EC2Config 서비스의 프록시 설정 구성

.NET용 AWS SDK, system.net 요소 또는 Microsoft Group Policy 및 Internet Explorer를 사용하여 프록시를 통해 통신하도록 EC2Config 서비스를 구성할 수 있습니다. AWS SDK for .NET을 사용하면 사용자 이름과 암호를 지정할 수 있기 때문에 좋습니다.

메소드

- AWS SDK for .NET을 사용하여 프록시 설정 구성(권장) (p. 508)
- system.net 요소를 사용하여 프록시 설정 구성 (p. 508)
- Microsoft Group Policy 및 Microsoft Internet Explorer를 사용하여 프록시 설정 구성 (p. 509)

AWS SDK for .NET을 사용하여 프록시 설정 구성(권장)

proxy 파일에 Ec2Config.exe.config 요소를 지정하여 EC2Config 서비스에 대한 프록시 설정을 구성할 수 있습니다. 자세한 내용은 [.NET용 AWS SDK에 대한 구성 파일 참조](#)를 참조하세요.

Ec2Config.exe.config에서 프록시 요소를 지정하는 방법

1. EC2Config 서비스가 프록시를 통해 통신하게 하려는 인스턴스에서 Ec2Config.exe.config 파일을 편집합니다. 기본적으로 이 파일은 %ProgramFiles%\Amazon\Ec2ConfigService 디렉터리에 위치합니다.
2. 다음 aws 요소를 configSections에 추가합니다. 기존의 sectionGroups에 추가하면 안 됩니다.

EC2Config 버전 3.17 이하의 경우

```
<configSections>
    <section name="aws" type="Amazon.AWSSection, AWSSDK"/>
</configSections>
```

EC2Config 버전 3.18 이상의 경우

```
<configSections>
    <section name="aws" type="Amazon.AWSSection, AWSSDK.Core"/>
</configSections>
```

3. 다음 aws 요소를 Ec2Config.exe.config 파일에 추가합니다.

```
<aws>
    <proxy
        host="string value"
        port="string value"
        username="string value"
        password="string value" />
</aws>
```

4. 변경 내용을 저장합니다.

system.net 요소를 사용하여 프록시 설정 구성

system.net 파일의 Ec2Config.exe.config 요소에 프록시 설정을 지정할 수 있습니다. 자세한 내용은 MSDN의 [defaultProxy Element](#) 단원을 참조하십시오.

Ec2Config.exe.config에서 system.net 요소를 지정하는 방법

1. EC2Config 서비스가 프록시를 통해 통신하게 하려는 인스턴스에서 Ec2Config.exe.config 파일을 편집합니다. 기본적으로 이 파일은 %ProgramFiles%\Amazon\Ec2ConfigService 디렉터리에 위치합니다.
2. defaultProxy 항목을 system.net에 추가합니다. 자세한 내용은 MSDN의 [defaultProxy Element](#) 단원을 참조하십시오.

예를 들어, 다음과 같이 구성하면 프록시를 우회하는 메타데이터 및 라이선스 트래픽을 제외하고 모든 트래픽이 현재 Internet Explorer에 구성된 프록시를 사용하도록 라우팅됩니다.

```
<defaultProxy>
    <proxy usesystemdefault="true" />
    <bypasslist>
        <add address="169.254.169.250" />
        <add address="169.254.169.251" />
        <add address="169.254.169.254" />
    </bypasslist>
</defaultProxy>
```

3. 변경 내용을 저장합니다.

Microsoft Group Policy 및 Microsoft Internet Explorer를 사용하여 프록시 설정 구성

EC2Config 서비스는 로컬 시스템 사용자 계정으로 실행됩니다. 인스턴스에서 Group Policy 설정을 변경한 후 Internet Explorer에서 이 계정에 대한 인스턴스 전역 프록시 설정을 지정할 수 있습니다.

Group Policy 및 Internet Explorer를 사용하여 프록시 설정을 구성하려면

1. EC2Config 서비스를 프록시를 통해 통신하게 하려는 인스턴스에서 관리자 자격으로 명령 프롬프트를 열고 **gpedit.msc**를 입력한 후 Enter를 누릅니다.
2. 로컬 그룹 정책 편집기의 로컬 컴퓨터 정책에서 컴퓨터 구성, 관리 템플릿, Windows 구성 요소, Internet Explorer를 선택합니다.
3. 오른쪽 창에서 사용자 단위보다는 컴퓨터 단위로 프록시 설정 만들기를 선택한 후 정책 설정 편집을 선택합니다.
4. 사용을 선택한 후 적용을 선택합니다.
5. Internet Explorer를 열고 도구 버튼을 선택합니다.
6. 인터넷 옵션을 선택한 후 연결 탭을 선택합니다.
7. LAN 설정을 선택합니다.
8. 프록시 서버에서 LAN에 프록시 서버 사용 옵션을 선택합니다.
9. 주소와 포트 정보를 지정한 후 확인을 선택합니다.

EC2Config 버전 기록

Windows Server 2016 이전의 Windows AMI는 Config 서비스(EC2Config.exe/EC2)라는 선택적 서비스를 포함합니다. 시작 중에 인스턴스가 부팅되고 작업을 수행할 때와 인스턴스를 종지하거나 시작할 때마다 EC2Config가 시작됩니다. Windows AMI에 포함된 EC2Config 버전에 대한 자세한 내용은 [관리형 AWS Windows AMI \(p. 26\)](#) 단원을 참조하십시오.

새로운 EC2Config 서비스 버전이 릴리스되면 알림을 받을 수 있습니다. 자세한 내용은 [EC2Config 서비스 알림 받기 \(p. 521\)](#) 단원을 참조하십시오.

다음 표에서는 EC2Config의 릴리스 버전에 대해 설명합니다. SSM 에이전트 업데이트에 대한 자세한 내용은 [시스템 관리자 SSM 에이전트 출시 정보](#)를 참조하십시오.

버전	세부 정보	릴리스 날짜
4.9.4222	<ul style="list-style-type: none"> IMDS 버전 1 대체 로직을 수정했습니다. 새로운 SSM 에이전트 버전 2.3.842.0 	2020년 4월 7일
4.9.4122	<ul style="list-style-type: none"> IMDS v2에 대한 지원을 추가했습니다. 새로운 SSM 에이전트 버전 2.3.814.0 	2020년 3월 4일
4.9.3865	<ul style="list-style-type: none"> 메탈 인스턴스에서 Windows Server 2008 R2에 대한 COM 포트 감지 문제를 해결했습니다. 새로운 SSM 에이전트 버전 2.3.722.0 	2019년 10월 31일
4.9.3519	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전 2.3.634.0 	2019년 6월 18일
4.9.3429	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전 2.3.542.0 	2019년 4월 25일
4.9.3289	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전 2.3.444.0 	2019년 2월 11일
4.9.3270	<ul style="list-style-type: none"> ACPI 문제 해결을 위해 모니터가 절대로 꺼지지 않게 설정하는 플러그인을 추가했습니다. SQL Server 에디션과 버전을 콘솔에 기록했습니다. 새로운 SSM 에이전트 버전 2.3.415.0 	2019년 1월 22일
4.9.3230	<ul style="list-style-type: none"> 기능에 좀 더 부합하도록 드라이브 문자 매핑 설명을 업데이트했습니다. 새로운 SSM 에이전트 버전 2.3.372.0 	2019년 1월 10일
4.9.3160	<ul style="list-style-type: none"> 기본 NIC의 대기 시간이 증가했습니다. RSS에 대한 기본 구성과 ENA 디바이스에 대한 수신 대기열 설정을 추가했습니다. Sysprep 도중 최대 절전 모드를 비활성화했습니다. 새로운 SSM 에이전트 버전 2.3.344.0 AWS SDK를 1.11.333으로 업그레이드했습니다. 	2018년 12월 15일
4.9.3067	<ul style="list-style-type: none"> 인스턴스 최대 절전 모드를 개선했습니다. 새로운 SSM 에이전트 버전 2.3.235.0 	2018년 11월 8일
4.9.3034	<ul style="list-style-type: none"> DNS 서버에 대한 라우트 169.254.169.253/32를 추가했습니다. 새로운 SSM 에이전트 버전 2.3.193.0 	2018년 10월 24일
4.9.2986	<ul style="list-style-type: none"> 모든 EC2Config 관련에 대한 서명을 추가했습니다. 새로운 SSM 에이전트 버전 2.3.136.0 	2018년 10월 11일
4.9.2953	새로운 SSM 에이전트 버전(2.3.117.0)	2018년 10월 2일
4.9.2926	새로운 SSM 에이전트 버전(2.3.68.0)	2018년 9월 18일
4.9.2905	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전(2.3.50.0) AMZN 시간 서비스에 라우트 169.254.169.123/32 추가. GRID 라이선스 서비스에 라우트 169.254.169.249/32 추가. 	2018년 9월 17일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EC2Config 서비스

버전	세부 정보	릴리스 날짜
	• EBS NVMe 볼륨이 임시로 표시되는 문제 해결.	
4.9.2854	새로운 SSM 에이전트 버전(2.3.13.0)	2018년 8월 17일
4.9.2831	새로운 SSM 에이전트 버전(2.2.916.0)	2018년 8월 7일
4.9.2818	새로운 SSM 에이전트 버전(2.2.902.0)	2018년 7월 31일
4.9.2756	새로운 SSM 에이전트 버전(2.2.800.0)	2018년 6월 27일
4.9.2688	새로운 SSM 에이전트 버전(2.2.607.0)	2018년 5월 25일
4.9.2660	새로운 SSM 에이전트 버전(2.2.546.0)	2018년 5월 11일
4.9.2644	새로운 SSM 에이전트 버전(2.2.493.0)	2018년 4월 26일
4.9.2586	새로운 SSM 에이전트 버전(2.2.392.0)	2018년 3월 28일
4.9.2565	• 새로운 SSM 에이전트 버전(2.2.355.0) • M5 및 C5 인스턴스의 문제 수정(PV 드라이버 검색 불가) • 인스턴스 유형, 최신 PV 드라이버, NVMe 드라이버에 대한 콘솔 로깅 추가	2018년 3월 13일
4.9.2549	새로운 SSM 에이전트 버전(2.2.325.0)	2018년 3월 8일
4.9.2461	새로운 SSM 에이전트 버전(2.2.257.0)	2018년 2월 15일
4.9.2439	새로운 SSM 에이전트 버전(2.2.191.0)	2018년 2월 6일
4.9.2400	새로운 SSM 에이전트 버전(2.2.160.0)	2018년 1월 16일
4.9.2327	• 새로운 SSM 에이전트 버전(2.2.120.0) • Amazon EC2 베어 메탈 인스턴스에 COM 포트 검색을 추가했습니다. • Amazon EC2 베어 메탈 인스턴스에 Hyper-V 상태 로깅을 추가했습니다.	2018년 1월 2일
4.9.2294	새로운 SSM 에이전트 버전(2.2.103.0)	2017년 12월 4일
4.9.2262	새로운 SSM 에이전트 버전(2.2.93.0)	2017년 11월 15일
4.9.2246	새로운 SSM 에이전트 버전(2.2.82.0)	2017년 11월 11일
4.9.2218	새로운 SSM 에이전트 버전(2.2.64.0)	2017년 10월 29일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EC2Config 서비스

버전	세부 정보	릴리스 날짜
4.9.2212	새로운 SSM 에이전트 버전(2.2.58.0)	2017년 10월 23일
4.9.2203	새로운 SSM 에이전트 버전(2.2.45.0)	2017년 10월 19일
4.9.2188	새로운 SSM 에이전트 버전(2.2.30.0)	2017년 10월 10일
4.9.2180	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전(2.2.24.0) GPU 인스턴스용 탄력적 GPU 플러그인을 추가했습니다. 	2017년 10월 5일
4.9.2143	새로운 SSM 에이전트 버전(2.2.16.0)	2017년 10월 1일
4.9.2140	새로운 SSM 에이전트 버전(2.1.10.0)	
4.9.2130	새로운 SSM 에이전트 버전(2.1.4.0)	
4.9.2106	새로운 SSM 에이전트 버전(2.0.952.0)	
4.9.2061	새로운 SSM 에이전트 버전(2.0.922.0)	
4.9.2047	새로운 SSM 에이전트 버전(2.0.913.0)	
4.9.2031	새로운 SSM 에이전트 버전(2.0.902.0)	
4.9.2016	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전(2.0.879.0) Windows Server 2003용 CloudWatch Logs 디렉터리 경로를 수정했습니다. 	
4.9.1981	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전(2.0.847.0) EBS 볼륨에서 생성되는 <code>important.txt</code>와 관련된 문제를 해결했습니다. 	
4.9.1964	새로운 SSM 에이전트 버전(2.0.842.0)	
4.9.1951	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전(2.0.834.0) 후발성 드라이브에 대해 Z:에서 매핑되지 않는 드라이브 문자 문제를 해결했습니다. 	
4.9.1925	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전(2.0.822.0) [버그] 이 버전은 SSM Agent v4.9.1775에서 업데이트할 수 있는 대상 버전이 아닙니다. 	
4.9.1900	새로운 SSM 에이전트 버전(2.0.805.0)	
4.9.1876	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전(2.0.796.0) 관리 사용자 데이터 실행에 대한 출력/오류 리디렉션 문제를 해결했습니다. 	
4.9.1863	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전(2.0.790.0) 여러 EBS 볼륨을 Amazon EC2 인스턴스에 연결할 때 발생하는 문제를 해결했습니다. 이전 버전과의 호환성을 위해 구성 경로를 가져오도록 CloudWatch 를 개선했습니다. 	

버전	세부 정보	릴리스 날짜
4.9.1791	새로운 SSM 에이전트 버전(2.0.767.0)	
4.9.1775	새로운 SSM 에이전트 버전(2.0.761.0)	
4.9.1752	새로운 SSM 에이전트 버전(2.0.755.0)	
4.9.1711	새로운 SSM 에이전트 버전(2.0.730.0)	
4.8.1676	새로운 SSM 에이전트 버전(2.0.716.0)	
4.7.1631	새로운 SSM 에이전트 버전(2.0.682.0)	
4.6.1579	<ul style="list-style-type: none"> 새로운 SSM 에이전트 버전(2.0.672.0) v4.3, v4.4, v4.5의 에이전트 업데이트 문제를 수정했습니다. 	
4.5.1534	새로운 SSM 에이전트 버전(2.0.645.1)	
4.4.1503	새로운 SSM 에이전트 버전(2.0.633.0)	
4.3.1472	새로운 SSM 에이전트 버전(2.0.617.1)	
4.2.1442	새로운 SSM 에이전트 버전(2.0.599.0)	
4.1.1378	새로운 SSM 에이전트 버전(2.0.558.0)	
4.0.1343	<ul style="list-style-type: none"> Run Command, State Manager, CloudWatch 에이전트, 도메인 조인 지원이 SSM 에이전트라고 하는 다른 에이전트로 이전되었습니다. SSM 에이전트는 EC2Config 업그레이드의 일부로 설치됩니다. 자세한 내용은 EC2Config 및 AWS 시스템 관리자 (p. 500) 단원을 참조하십시오. EC2Config에 프록시를 설정해 놓은 경우 업그레이드하기 전에 SSM 에이전트의 프록시 설정을 업데이트해야 합니다. 프록시 설정을 업데이트하지 않으면 Run Command를 사용하여 인스턴스를 관리할 수 없습니다. 이러한 문제를 방지하려면 새로운 버전으로 업데이트하기 전에 AWS 시스템 관리자 사용 설명서의 Installing and Configuring SSM Agent on Windows Instances의 정보를 참조하십시오. 이전에 로컬 구성 파일(AWS.EC2.Windows.CloudWatch.json)을 사용하여 인스턴스에서 CloudWatch 통합을 활성화한 경우 SSM 에이전트로 작업하도록 파일을 구성해야 합니다. 	
3.19.1153	<ul style="list-style-type: none"> 이전 KMS 구성을 사용하여 인스턴스에 대한 활성화 플러그인을 다시 활성화했습니다. 디스크 포맷 작업 중에 비활성화할 기본 TRIM 행동을 변경하고, 사용자 데이터로 InitializeDisks 플러그인을 재정의하기 위해 FormatWithTRIM을 추가했습니다. 	
3.18.1118	<ul style="list-style-type: none"> 기본 네트워크 어댑터에 대한 경로를 안정적으로 추가하도록 수정했습니다. AWS 서비스 지원 향상을 위한 업데이트. 	
3.17.1032	<ul style="list-style-type: none"> 필터를 동일한 범주로 설정한 경우에 표시되는 중복 시스템 로그를 수정했습니다. 디스크 초기화 중에 중지되는 문제를 수정했습니다. 	
3.16.930	시작할 때 Windows 이벤트 로그에 "Window is Ready to use" 이벤트를 기록하도록 지원을 추가했습니다.	

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EC2Config 서비스

버전	세부 정보	릴리스 날짜
3.15.880	'!' 문자를 포함하는 S3 버킷 이름에 대한 시스템 관리자 Run Command 출력을 업로드할 수 있도록 수정했습니다.	
3.14.786	InitializeDisks 플러그인 설정을 재정의하도록 지원을 추가했습니다. 예: SSD 디스크 초기화 속도를 높이려면 사용자 데이터에서 다음을 지정하여 TRIM을 일시적으로 비활성화 할 수 있습니다. <InitializeDrivesSettings><SettingsGroup>FormatWithoutTRIM</SettingsGroup></InitializeDrivesSettings>	
3.13.727	시스템 관리자 Run Command - Windows가 재부팅된 후 명령을 안정적으로 처리하도록 수정했습니다.	
3.12.649	<ul style="list-style-type: none"> 명령/스크립트를 실행할 때 재부팅을 정상적으로 처리하도록 수정했습니다. 실행 중인 명령을 안정적으로 취소할 수 있도록 수정했습니다. 시스템 관리자 Run Command를 통해 애플리케이션을 설치할 때 MSI 로그를 S3에 업로드(선택 사항)하기 위한 지원을 추가했습니다. 	
3.11.521	<ul style="list-style-type: none"> Windows Server 2003에 대한 RDP 지문 생성을 활성화하도록 수정했습니다. EC2Config 로그 줄에 시간대와 UTC 오프셋을 포함하도록 수정했습니다. 시스템 관리자은 병렬 Run Command 실행을 지원합니다. 분할된 디스크를 온라인으로 전환하기 위해 이전 변경 사항을 룰백 합니다. 	
3.10.442	<ul style="list-style-type: none"> MSI 애플리케이션을 설치할 때 발생하는 시스템 관리자 구성 장애를 해결했습니다. 스토리지 디스크를 온라인으로 안정적으로 전환하도록 수정했습니다. AWS 서비스 지원 향상을 위한 업데이트. 	
3.9.359	<ul style="list-style-type: none"> Sysprep 이후 스크립트에서 Windows 업데이트 구성을 기본 상태로 유지하도록 수정했습니다. GPO 암호 정책 설정을 가져올 때 안정성을 향상하기 위해 암호 생성 플러그인을 수정했습니다. EC2Config/SSM 로그 폴더 권한을 로컬 Administrators 그룹으로 제한합니다. AWS 서비스 지원 향상을 위한 업데이트. 	
3.8.294	<ul style="list-style-type: none"> 기본 드라이브에 있지 않은 경우 로그가 업로드되지 않는 CloudWatch 문제를 해결했습니다. 재시도 로직을 추가하여 디스크 초기화 프로세스를 개선했습니다. AMI 생성 중에 SetPassword 플러그인이 실패할 경우에 발생하는 오류 처리 성능을 개선했습니다. AWS 서비스 지원 향상을 위한 업데이트. 	

버전	세부 정보	릴리스 날짜
3.7.308	<ul style="list-style-type: none"> 인스턴스 내에서 구성을 테스트하고 문제를 해결하기 위해 ec2config-cli 유ти리티를 개선했습니다. OpenVPN 어댑터에서 KMS 및 메타 데이터 서비스에 대한 정적 경로 추가를 방지합니다. 사용자 데이터 실행 중에 "persist" 태그를 인식하지 못하는 문제를 해결했습니다. EC2 콘솔 로깅을 사용할 수 없을 때의 오류 처리 성능을 개선했습니다. AWS 서비스 지원 향상을 위한 업데이트. 	
3.6.269	<ul style="list-style-type: none"> KMS를 통해 Windows를 정품 인증할 때 링크 로컬 주소 169.254.0.250/251을 먼저 사용하도록 Windows 정품 인증 안정성을 수정했습니다. 시스템 관리자, Windows 정품 인증 및 도메인 조인 시나리오에 대한 프록시 처리 성능을 개선했습니다. Sysprep 응답 파일에 사용자 계정 줄이 중복으로 추가되는 문제를 해결했습니다. 	
3.5.228	<ul style="list-style-type: none"> CloudWatch 플러그인이 Windows 이벤트 로그를 읽을 때 CPU와 메모리를 과도하게 사용하는 문제를 해결했습니다. EC2Config 설정 UI에 CloudWatch 구성 문서에 대한 링크를 추가했습니다. 	
3.4.212	<ul style="list-style-type: none"> VM 가져오기와 함께 사용될 때 발생하는 EC2Config 문제를 해결했습니다. WiX 설치 관리자의 서비스 이름 지정 문제를 해결했습니다. 	
3.3.174	<ul style="list-style-type: none"> 시스템 관리자 및 도메인 조인 장애에 대한 예외 처리 성능을 개선했습니다. 시스템 관리자 SSM 스키마 버전 관리를 지원하도록 변경했습니다. Win2K3의 휘발성 디스크 포맷을 수정했습니다. 2TB보다 큰 디스크 크기 구성을 지원하도록 변경했습니다. GC 모드를 기본값으로 설정하여 가상 메모리 사용을 축소했습니다. aws:psModule 및 aws:application 플러그인의 UNC 경로에서 아티팩트 다운로드를 지원합니다. Windows 정품 인증 플러그인에 대한 로깅을 개선했습니다. 	
3.2.97	<ul style="list-style-type: none"> 시스템 관리자 SSM 어셈블리를 지역 로드하여 성능을 개선했습니다. 형식이 잘못된 sysprep2008.xml에 대한 예외 처리를 개선했습니다. 시스템 관리자 "Apply" 구성에 대한 명령줄 지원을 추가했습니다. 컴퓨터 이름 바꾸기가 보류 중일 때 도메인 조인을 지원하도록 변경했습니다. aws:applications 플러그인에서 선택적 파라미터를 지원합니다. aws:psModule 플러그인에서 명령 어레이를 지원합니다. 	

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EC2Config 서비스

버전	세부 정보	릴리스 날짜
3.0.54	<ul style="list-style-type: none"> 시스템 관리자 지원 활성화 시스템 관리자를 통해 EC2 Windows 인스턴스를 AWS 딜렉터리에 자동으로 도메인 조인합니다. 시스템 관리자를 통해 CloudWatch 로그/지표를 구성하고 업로드합니다. 시스템 관리자를 통해 PowerShell 모듈을 설치합니다. 시스템 관리자를 통해 MSI 애플리케이션을 설치합니다. 	
2.4.233	<ul style="list-style-type: none"> 서비스 시작 실패로부터 EC2Config를 복구하도록 예약된 작업을 추가했습니다. 콘솔 로그 오류 메시지를 개선했습니다. AWS 서비스 지원 향상을 위한 업데이트. 	
2.3.313	<ul style="list-style-type: none"> CloudWatch 로그 기능을 활성화할 때 많은 메모리가 사용되는 문제를 해결했습니다. 업그레이드 버그를 수정하여 이제 ec2config 2.1.19 이하 버전에서 최신 버전으로 업그레이드할 수 있습니다. 로그에서 COM 포트 열기 예외를 보다 친숙하고 유용하게 업데이트했습니다. Ec2configServiceSettings UI 크기 조정을 비활성화하고 UI의 특성 및 버전 표시 위치를 수정했습니다. 	
2.2.12	<ul style="list-style-type: none"> 종종 null을 반환하는 Windows Sysprep 상태를 결정하기 위해 레지스트리 키를 쿼리할 때 발생하는 NullPointerException 예외를 해결했습니다. 마지막 블록에서 관리되지 않는 리소스를 확보했습니다. 	
2.2.11	CloudWatch 플러그인에서 빈 로그 줄 처리 문제를 해결했습니다.	
2.2.10	<ul style="list-style-type: none"> UI를 통한 CloudWatch 로그 설정構성을 제거했습니다. 사용자가 항후 개선을 허용하도록 %ProgramFiles%\Amazon\Ec2ConfigService\Settings\AWS.EC2.Windows.CloudWatch.json 파일에서 CloudWatch 로그 설정을 정의할 수 있습니다. 	
2.2.9	처리되지 않는 예외를 해결하고 로깅을 추가했습니다.	
2.2.8	<ul style="list-style-type: none"> Windows Server 2003 SP1 이상을 지원하도록 EC2Config 설치 관리자에서 Windows OS 버전 검사를 수정했습니다. Sysprep 구성 파일과 관련된 레지스트리 키를 읽을 때 null 값 처리를 수정했습니다. 	
2.2.7	<ul style="list-style-type: none"> Windows 2008 이상에 대해 Sysprep을 실행하는 중에 EC2Config를 실행하도록 지원을 추가했습니다. 향상된 진단을 위해 예외 처리 및 로깅을 개선했습니다. 	
2.2.6	<ul style="list-style-type: none"> 로그 이벤트를 업로드할 때 인스턴스 및 CloudWatch 로그에 대한 부하를 축소했습니다. CloudWatch 로그 플러그인이 항상 활성화된 상태로 유지되지 않는 업그레이드 문제를 해결했습니다. 	

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EC2Config 서비스

버전	세부 정보	릴리스 날짜
2.2.5	<ul style="list-style-type: none"> CloudWatch 로그 서비스에 대한 로그 업로드 지원을 추가했습니다. Ec2OutputRDPCert 플러그인에서 경합 문제를 해결했습니다. TakeNoAction에서 재시작하도록 EC2Config 서비스 복구 옵션을 변경했습니다. EC2Config가 충돌하는 경우에 더 많은 예외 정보를 추가했습니다. 	
2.2.4	<ul style="list-style-type: none"> PostSysprep.cmd에서 오타를 수정했습니다. EC2Config가 OS2012+의 시작 메뉴에 고정되지 않는 버그를 수정했습니다. 	
2.2.3	<ul style="list-style-type: none"> 설치 시 서비스를 즉시 시작하지 않고 EC2Config를 설치하기 위한 옵션을 추가했습니다. 사용하려면 명령 프롬프트에서 'Ec2Install.exe start=false'를 실행합니다. 월페이퍼(wallpaper) 플러그인에 월페이퍼 추가/제거를 제어하기 위한 파라미터를 추가했습니다. 사용하려면 명령 프롬프트에서 'Ec2WallpaperInfo.exe set' 또는 'Ec2WallpaperInfo.exe revert'를 실행합니다. RealTimelsUniversal 레지스트리 키의 잘못된 설정을 콘솔에 출력하도록 RealTimelsUniversal 키 검사를 추가했습니다. Windows temp 폴더에 대한 EC2Config 종속성을 제거했습니다. .Net 3.5에 대한 사용자 데이터 실행 종속성을 제거했습니다. 	
2.2.2	<ul style="list-style-type: none"> 리소스가 릴리스 중인지 확인하도록 서비스 중지 동작에 대한 검사 기능을 추가했습니다. 도메인에 조인될 때 실행 시간이 오래 걸리는 문제를 해결했습니다. 	
2.2.1	<ul style="list-style-type: none"> 이전 버전에서의 업그레이드를 허용하도록 설치 관리자를 업데이트했습니다. .Net4.5 전용 환경에서 Ec2WallpaperInfo 버그를 해결했습니다. 간헐적인 드라이버 감지 버그를 해결했습니다. 자동 설치 옵션을 추가했습니다. '-q' 옵션을 사용하여 Ec2Install.exe 실행(예: 'Ec2Install.exe -q') 	
2.2.0	<ul style="list-style-type: none"> .Net4 및 .Net4.5 전용 환경에 대한 지원을 추가했습니다. 설치 관리자를 업데이트했습니다. 	
2.1.19	<ul style="list-style-type: none"> intel 네트워크 드라이버(예: C3 인스턴스 유형)를 사용하는 경우의 휨발성 디스크 레이블 지정 자세한 내용은 Windows에서 향상된 네트워킹 (p. 751) 단원을 참조하십시오. 콘솔 출력에 AMI 원본 버전 및 AMI 원본 이름 지원을 추가했습니다. 일관된 서식/구문 분석을 위해 콘솔 출력을 변경했습니다. 도움말 파일을 업데이트했습니다. 	
2.1.18	<ul style="list-style-type: none"> 완료 알림에 대한 EC2Config WMI 객체(-Namespace root\Amazon -Class EC2_ConfigService)를 추가했습니다. 대용량 이벤트 로그를 사용하여 초기 실행 중에 오래 동안 높은 CPU를 사용하는 시작 WMI 쿼리 성능을 개선했습니다. 	

버전	세부 정보	릴리스 날짜
2.1.17	<ul style="list-style-type: none"> 표준 출력 및 표준 오류 버퍼 채우기를 통해 사용자 데이터 실행 문제를 해결했습니다. w2k8 이상 OS에 대한 콘솔 출력에 나타나는 잘못된 RDP 지문을 수정했습니다. 이제 Windows 2008 이상 버전의 콘솔 출력에는 머신 이름 값과 포함하는 'RDPCERTIFICATE-SubjectName:'이 포함되어 있습니다. [D:\ to Drive Letter Mapping] 드롭다운을 추가했습니다. 도움말 버튼을 오른쪽 위로 이동하고 모양/느낌을 변경했습니다. Feedback Survey 링크를 오른쪽 위에 추가했습니다. 	
2.1.16	<ul style="list-style-type: none"> [General] 탭에 새 버전의 EC2Config 다운로드 페이지에 대한 링크가 포함되어 있습니다. MyDoc 리디렉션을 지원하기 위해 이제 바탕 화면 월페이퍼 (Wallpaper) 오버레이가 [내 문서] 대신 Users Local Appdata 폴더에 저장됩니다. Sysprep 이후 스크립트에서 MSSQLServer 이름이 시스템과 동기화되었습니다(2008 이상). 애플리케이션 폴더 순서를 변경했습니다(파일을 Plugin 디렉터리로 이동하고 종복 파일 제거). 시스템 로그 출력(콘솔)을 변경했습니다. *쉬운 구문 분석을 위해 날짜, 이름, 값 형식으로 전환했습니다. 새 형식에 대한 종속성 마이그레이션을 시작하십시오. *"Ec2SetPassword" 플러그인 상태를 추가했습니다. *Sysprep 시작 및 종료 시간을 추가했습니다. 영어 이외 언어로 된 운영 체제에서 휘발성 디스크가 'Temporary Storage'로 레이블링되지 않는 문제를 해결했습니다. Sysprep을 실행한 이후의 EC2Config 제거 오류를 해결했습니다. 	
2.1.15	<ul style="list-style-type: none"> 메타데이터 서비스에 대한 요청을 최적화했습니다. 메타데이터는 이제 프록시 설정을 우회합니다. 휘발성 디스크가 'Temporary Storage'로 레이블 지정되고 Important.txt가 볼륨에 배치됩니다(있는 경우)(Citrix PV 드라이버에만 해당). 자세한 내용은 Windows 인스턴스에서 PV 드라이버 업그레이드 (p. 528) 단원을 참조하십시오. 휘발성 디스크에는 A부터 Z 사이의 드라이브 문자가 할당됩니다 (Citrix PV 드라이버에만 해당). 드라이브 문자 매핑 플러그인을 사용하여 볼륨 레이블 'Temporary Storage X'로 할당을 덮어쓸 수 있습니다. 여기서 x는 0-25 사이의 숫자입니다. 사용자 데이터는 이제 'Windows is Ready' 상태 전환 후 바로 실행됩니다. 	
2.1.14	바탕 화면 월페이퍼(wallpaper)를 수정했습니다.	
2.1.13	<ul style="list-style-type: none"> 바탕 화면 월페이퍼(wallpaper)에 기본적으로 호스트 이름이 표시됩니다. Windows 시간 서비스에 대한 종속성을 제거했습니다. 단일 인터페이스에 여러 IP가 할당되는 경우의 경로를 추가했습니다. 	

버전	세부 정보	릴리스 날짜
2.1.11	<ul style="list-style-type: none"> Ec2Activation 플러그인을 변경했습니다. -정품 인증 상태를 30일마다 확인합니다. -유예 기간이 180일 중에서 90일이 남으면 정품 인증을 다시 시도합니다. 	
2.1.10	<ul style="list-style-type: none"> Sysprep 또는 [Shutdown without Sysprep] 시 바탕 화면 월페이퍼 (wallpaper) 오버레이가 더 이상 지속되지 않습니다. 모든 서비스에 대해 실행할 사용자 데이터 옵션은 <persist>true</persist>로 시작 /DisableWinUpdate.cmd의 위치와 이름을 /Scripts/PostSysprep.cmd로 변경했습니다. /Scripts/PostSysprep.cmd에서 관리자 암호를 기본적으로 만료되지 않도록 설정했습니다. 설치 제거하면 EC2Config PostSysprep 스크립트가 c:\windows\setup\script\CommandComplete.cmd에서 제거됩니다. [Add Route]에서 사용자 지정 인터페이스 지표를 지원합니다. 	
2.1.9	사용자 데이터 실행이 더 이상 3851자로 제한되지 않습니다.	
2.1.7	<ul style="list-style-type: none"> OS 버전과 언어 식별자가 콘솔에 기록됩니다. EC2Config 버전이 콘솔에 기록됩니다. PV 드라이버 버전이 콘솔에 기록됩니다. 버그를 검사한 후 버그가 있을 경우 다음에 부팅할 때 콘솔에 출력합니다. Sysprep 자격 증명을 유지하도록 config.xml에 옵션을 추가했습니다. 시작 시 ENI를 사용할 수 없을 경우 경로 재시도 로직을 추가했습니다. 사용자 데이터 실행 PID를 콘솔에 기록했습니다. 최소 생성 암호 길이를 GPO에서 검색합니다. 3회 재시도하도록 서비스 시작을 설정했습니다. S3_DownloadFile.ps1 및 S3_Upload file.ps1 예를 /Scripts 폴더에 추가했습니다. 	

버전	세부 정보	릴리스 날짜
2.1.6	<ul style="list-style-type: none"> 버전 정보를 [General] 탭에 추가했습니다. [Bundle] 탭의 이름을 [Image]로 변경했습니다. 암호 지정 프로세스를 간소화하고 암호 관련 UI를 [General] 탭에서 [Image] 탭으로 이동했습니다. [Disk Settings] 탭의 이름을 [Storage]로 변경했습니다. 문제 해결을 위한 일반 도구가 들어 있는 [Support] 탭을 추가했습니다. OS 파티션을 기본적으로 확장하도록 Windows Server 2003 sysprep.ini를 설정했습니다. 월페이퍼(wallpaper)에 대한 프라이빗 IP 주소를 추가했습니다. 프라이빗 IP 주소가 월페이퍼(wallpaper)에 표시됩니다. 콘솔 출력에 대한 재시도 로직을 추가했습니다. 메타데이터 액세스 가능성에 대한 COM 포트 예외(콘솔 출력이 표시되기 이전에 EC2Config가 종료되는 문제)를 수정했습니다. 부팅할 때마다 정품 인증 상태를 확인하여 필요 시 정품 인증을 수행합니다. 시작 폴더에서 월페이퍼(wallpaper) 바로 가기를 수동으로 실행할 때 Administrator/logs를 가리키는 상대 경로 문제를 해결했습니다. Windows Server 2003 사용자(관리자 이외)에 대한 기본 배경색을 수정했습니다. 	
2.1.2	<ul style="list-style-type: none"> UTC(Zulu) 기반 콘솔 타임스탬프 [Sysprep] 탭에서 하이퍼링크 모양을 제거했습니다. Windows 2008 이상을 처음으로 부팅할 때 루트 볼륨을 등적으로 확장하는 기능을 추가했습니다. [Set-Password]를 활성화하면 이제 암호 설정을 위해 EC2Config가 자동으로 활성화됩니다. EC2Config는 Sysprep을 실행하기 이전에 정품 인증 상태를 확인합니다(정품 인증되지 않은 경우 경고 표시). 이제 Windows Server 2003 sysprep.xml은 기본적으로 태평양 시간대 대신 UTC 시간대로 설정됩니다. 임의 정품 인증 서버 [Drive Mapping] 탭의 이름을 [Disk Settings]로 변경했습니다. [Initialize Drives] UI 항목을 [General] 탭에서 [Disk Settings] 탭으로 이동했습니다. 도움말 버튼이 이제 HTML 도움말 파일을 가리킵니다. 변경 사항으로 HTML 도움말 파일을 업데이트했습니다. 드라이브 문자 매핑에 대한 'Note' 텍스트를 업데이트했습니다. Sysprep 이전에 패치 및 정리를 자동화하기 위해 InstallUpdates.ps1을 /Scripts 폴더에 추가했습니다. 	
2.1.0	<ul style="list-style-type: none"> 처음으로 로그온할 때(연결을 끊었다가 다시 연결할 때 아님) 바탕 화면 월페이퍼(wallpaper)에 기본적으로 인스턴스 정보가 표시됩니다. 코드를 <powershell></powershell>로 뷰어서 사용자 데이터에서 PowerShell을 실행할 수 있습니다. 	

EC2Config 서비스 알림 받기

새로운 EC2Config 서비스 버전이 릴리스되면 이를 알리도록 Amazon SNS를 설정할 수 있습니다. 알림을 받으려면 다음 절차를 수행합니다.

EC2Config 알림을 구독하려면

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 필요한 경우 탐색 모음에서 리전을 미국 동부(버지니아 북부)로 변경하십시오. 구독하는 SNS 알림이 이 리전에 생성되었기 때문에 이 리전을 선택해야 합니다.
3. 탐색 창에서 구독을 선택합니다.
4. 구독 생성을 선택합니다.
5. 구독 생성 대화 상자에서 다음 작업을 수행합니다.
 - a. 주제 ARN에 다음 Amazon 리소스 이름(ARN)을 사용합니다.

```
arn:aws:sns:us-east-1:801119661308:ec2-windows-ec2config
```

- b. 프로토콜에서 Email을 선택합니다.
- c. 앤드포인트에서 알림을 받을 이메일 주소를 입력합니다.
- d. Create subscription을 선택합니다.

6. 구독을 확인하도록 요청하는 전자 메일이 전송되면 이메일을 열고 지침에 따라 구독을 완료합니다.

새 EC2 Config 서비스 버전이 릴리스될 때마다 구독자에게 알림이 전송됩니다. 이런 알림을 더 이상 받지 않기를 원하는 경우, 다음 절차를 수행해서 구독을 해제하십시오.

EC2Config 알림을 구독 해제하려면

1. Amazon SNS 콘솔을 엽니다.
2. 탐색 창에서 구독을 선택합니다.
3. 구독을 선택한 후 작업, 구독 삭제를 선택합니다. 확인 메시지가 나타나면 삭제를 선택합니다.

EC2Config 서비스와 관련된 문제 해결

다음 정보는 EC2Config 서비스와 관련된 문제를 해결하는 데 도움이 될 수 있습니다.

접속 불가 인스턴스에서 EC2Config 업데이트

원격 데스크톱을 사용하여 액세스할 수 없는 Windows Server 인스턴스에서 EC2Config 서비스를 업데이트 하려면 다음 절차를 사용합니다.

연결할 수 없는 Amazon EBS 지원 Windows 인스턴스에서 EC2Config 업데이트 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 문제가 발생한 인스턴스를 찾습니다. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고, 인스턴스 상태를 선택한 다음 중지를 선택합니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

4. 인스턴스 시작을 선택하고 문제가 발생한 인스턴스와 동일한 가용 영역에 임시 t2.micro 인스턴스를 생성합니다. 문제가 발생한 인스턴스를 시작하는 데 사용한 것과 다른 AMI를 사용합니다.

Important

문제가 발생한 인스턴스와 동일한 가용 영역에서 인스턴스를 생성하지 않는 경우에는 문제가 발생한 인스턴스의 루트 볼륨을 새 인스턴스에 연결할 수 없습니다.

5. EC2 콘솔에서 볼륨을 선택합니다.
6. 문제가 발생한 인스턴스의 루트 볼륨을 찾습니다. [볼륨을 분리 \(p. 964\)](#)하고 이전에 생성한 임시 인스턴스에 [볼륨을 연결 \(p. 950\)](#)합니다. 기본 디바이스 이름(xvdf)으로 연결합니다.
7. 원격 드라이브를 사용하여 임시 인스턴스에 연결한 후 디스크 관리 유ти리티를 사용하여 [볼륨을 사용할 수 있도록 지정 \(p. 951\)](#)합니다.
8. 최신 버전의 EC2Config 서비스를 [다운로드](#)합니다. 연결한 드라이브의 .zip 디렉터리에 Temp 파일의 압축을 풁니다.
9. 임시 인스턴스에서 실행 대화 상자를 열고 **regedit**를 입력한 다음 Enter 키를 누릅니다.
10. HKEY_LOCAL_MACHINE을 선택합니다. 파일 메뉴에서 Hive 로드(Load Hive)를 선택합니다. 드라이브를 선택한 다음 해당 드라이브로 이동하여 Windows\System32\config\SOFTWARE 파일을 엽니다. 메시지가 나타나면 키 이름을 지정합니다.
11. 방금 로드한 키를 선택하고 Microsoft\Windows\CurrentVersion 경로로 이동합니다. RunOnce 키를 선택합니다. 이 키가 없는 경우 컨텍스트(오른쪽 클릭) 메뉴에서 CurrentVersion을 선택하고 새로 생성을 선택한 다음 키를 선택합니다. 키 이름을 RunOnce로 지정합니다.
12. 컨텍스트(오른쪽 클릭) 메뉴에서 RunOnce 키를 선택하고 새로 생성을 선택한 다음 문자열 값(String Value)을 선택합니다. Ec2Install을 이름으로, C:\Temp\Ec2Install.exe /quiet를 데이터로 입력합니다.
13. HKEY_LOCAL_MACHINE\specified key name\Microsoft\Windows NT\CurrentVersion \Winlogon 키를 선택합니다. 컨텍스트(오른쪽 클릭) 메뉴에서 새로 생성을 선택한 다음 문자열 값(String Value)을 선택합니다. AutoAdminLogon을 이름으로, 1를 값 데이터로 입력합니다.
14. HKEY_LOCAL_MACHINE\specified key name\Microsoft\Windows NT\CurrentVersion \Winlogon> 키를 선택합니다. 컨텍스트(오른쪽 클릭) 메뉴에서 새로 생성을 선택한 다음 문자열 값(String Value)을 선택합니다. DefaultUserName을 이름으로, Administrator를 값 데이터로 입력합니다.
15. HKEY_LOCAL_MACHINE\specified key name\Microsoft\Windows NT\CurrentVersion \Winlogon 키를 선택합니다. 컨텍스트(오른쪽 클릭) 메뉴에서 새로 생성을 선택한 다음 문자열 값(String Value)을 선택합니다. DefaultPassword를 이름으로 입력하고 값 데이터에 암호를 입력합니다.
16. 레지스트리 편집기 탐색 창에서 처음 레지스트리 편집기를 열 때 생성한 임시 키를 선택합니다.
17. 파일(File) 메뉴에서 Hive 언로드(Unload Hive)를 선택합니다.
18. 디스크 관리 유ти리티에서 이전에 연결한 드라이브를 선택하고 마우스 오른쪽 버튼을 클릭하여 컨텍스트 메뉴를 열고 오프라인을 선택합니다.
19. Amazon EC2 콘솔에서 임시 인스턴스로부터 문제가 발생한 볼륨을 분리하고 디바이스 이름 /dev/sda1을 사용하여 인스턴스에 이를 다시 연결합니다. 볼륨을 루트 볼륨으로 지정하려면 이 디바이스 이름을 지정해야 합니다.
20. [인스턴스 중지 및 시작 \(p. 440\)](#) 인스턴스.
21. 인스턴스 시작 후에 시스템 로그를 검사하여 Windows is ready to use 메시지를 확인합니다.
22. 레지스트리 편집기를 열고 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT \CurrentVersion\Winlogon을 선택합니다. 앞에서 만든 문자열 값 키 AutoAdminLogon, DefaultUserName 및 DefaultPassword를 삭제합니다.
23. 앞에서 임시로 생성한 인스턴스는 이 절차에서 삭제하거나 종단합니다.

Windows 인스턴스의 반가상화(PV) 드라이버

Windows AMI는 가상화 하드웨어에 대한 액세스를 허용하는 드라이버 세트를 포함하고 있습니다. 이 드라이버는 Amazon EC2에 의해 사용되어 인스턴스 스토어 및 Amazon EBS 볼륨을 해당 디바이스로 매핑합니다. 다음 표는 드라이버 간의 주요 차이점을 보여줍니다.

	RedHat PV	Citrix PV	AWS PV
인스턴스 유형	모든 인스턴스 유형에서 지원되는 것은 아님. 사용자가 지원되지 않는 인스턴스를 지정한 경우 인스턴스가 손상됩니다.	모든 인스턴스 유형에서 지원됨.	모든 인스턴스 유형에서 지원됨.
연결된 볼륨	최대 16개 볼륨 연결 지원.	16개 이상 볼륨 연결 지원.	16개 이상 볼륨 연결 지원.
네트워크	이 드라이버에는 부하가 높은 경우 (예: 빠른 FTP 파일 전송) 네트워크 연결이 초기화되는 알려진 문제가 있습니다.		호환되는 인스턴스 유형인 경우 드라이버는 네트워크 어댑터에서 점보 프레임을 자동으로 구성합니다. 인스턴스가 클러스터 배치 그룹 (p. 761)에 속하게 되면 이 클러스터 배치 그룹 내 인스턴스 간의 네트워크 성능이 향상됩니다.

다음 목록에서는 Amazon EC2의 각 Windows Server 버전에서 어떤 PV 드라이버를 실행해야 하는지를 보여줍니다.

- Windows Server 2019: AWS PV
- Windows Server 2016: AWS PV
- Windows Server 2012 및 2012 R2: AWS PV
- Windows Server 2008 R2: AWS PV
- Windows Server 2008: Citrix PV 5.9

내용

- [AWS PV 드라이버 \(p. 524\)](#)
- [Citrix PV 드라이버 \(p. 526\)](#)
- [RedHat PV 드라이버 \(p. 527\)](#)
- [알림 구독 \(p. 527\)](#)
- [Windows 인스턴스에서 PV 드라이버 업그레이드 \(p. 528\)](#)
- [PV 드라이버 문제 해결 \(p. 534\)](#)

AWS PV 드라이버

AWS PV 드라이버는 %ProgramFiles%\Amazon\Xentools 디렉터리에 저장됩니다. 이 디렉터리에는 퍼블릭 기호 및 xenstore_client.exe 명령줄 도구가 포함되어 사용자는 XenStore의 항목에 액세스할 수 있습니다. 예를 들어, 다음 PowerShell 명령은 하이퍼바이저에서 현재 시간을 반환합니다.

```
PS C:\> [DateTime]::FromFileTimeUTC((gwmi -n root\wmi -cl AWSXenStoreBase).XenTime).ToString("hh:mm:ss")
11:17:00
```

AWS PV 드라이버 구성 요소는 Windows 레지스트리의 HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services 아래에 나열됩니다. 이러한 드라이버 구성 요소로는 xenbus, xeniface, xennet, xenvbd 및 xenvif 등이 있습니다.

또한 AWS PV 드라이버에는 사용자 모드에서 실행되는 LiteAgent라는 Windows 서비스가 있습니다. 이 서비스는 Xen 세대 인스턴스의 AWS API에서 이벤트 종료 및 재시작과 같은 작업을 수행합니다. 사용자는 명령 줄에서 Services.msc를 실행하여 서비스에 액세스하고 관리할 수 있습니다. Nitro 세대 인스턴스에서 실행할 때는 AWS PV 드라이버가 사용되지 않고 LiteAgent 서비스가 드라이버 버전 8.2.4에서 시작을 자체적으로 중지합니다. 또한 최신 AWS PV 드라이버로 업데이트하면 LiteAgent가 업데이트되고 모든 인스턴스 세대에서 신뢰성을 높일 수 있습니다.

최신 AWS PV 드라이버 설치

Amazon Windows AMI는 가상화 하드웨어에 대한 액세스를 허용하는 드라이버 세트를 포함하고 있습니다. 이 드라이버는 Amazon EC2에 의해 사용되어 인스턴스 스토어 및 Amazon EBS 볼륨을 해당 디바이스로 매핑합니다. EC2 Windows 인스턴스의 안정성과 성능을 향상하려면 최신 드라이버를 설치하는 것이 좋습니다.

설치 옵션

- AWS 시스템 관리자를 사용하여 PV 드라이버를 자동으로 업데이트할 수 있습니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [연습: EC2 Windows 인스턴스에서 PV 드라이버 자동 업데이트\(콘솔\)](#)를 참조하십시오.
- 설치 패키지를 [다운로드](#)한 후 설치 프로그램을 수동으로 실행할 수 있습니다. AWS PV 드라이버 다운로드 및 설치에 대한 자세한 내용 또는 도메인 컨트롤러를 업그레이드 할 경우 [Windows Server 인스턴스 업그레이드\(AWS PV 업그레이드\)](#) (p. 528) 단원을 참조하십시오.

AWS PV 드라이버 패키지 내역

다음 표에서는 각 드라이버 릴리스에 대한 AWS PV 드라이버의 변경 사항을 보여 줍니다.

패키지 버전	세부 정보	릴리스 날짜
8.3.4	네트워크 디바이스 연결의 안정성이 향상되었습니다.	2020년 8월 4일
8.3.3	<ul style="list-style-type: none">오류 처리 경로 중 버그 검사를 방지하기 위해 XenStore 지향 구성 요소로 업데이트합니다.잘못된 SRB가 제출될 때 충돌을 방지하기 위해 스토리지 구성 요소로 업데이트합니다.	2020년 2월 4일

Note

Windows Server 2008 R2 인스턴스에서 이 드라이버를 업데이트하려면 먼저 Microsoft 보안 공지 사항 (<https://docs.microsoft.com/en-us/security-updates/>)

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
PV 드라이버

패키지 버전	세부 정보	릴리스 날짜
	<p>SecurityAdvisories/2014/2949927)을 해결하기 위해 적절한 패치가 설치되어 있는지 확인해야 합니다.</p>	
8.3.2	네트워킹 구성 요소 안정성이 향상되었습니다.	2019년 7월 30일
8.3.1	스토리지 구성 요소의 성능과 견고성을 개선했습니다.	2019년 6월 12일
8.2.7	최신 세대 인스턴스 유형으로의 마이그레이션을 지원하는 향상된 효율성.	2019년 5월 20일
8.2.6	충돌 덤프 경로의 효율성을 개선했습니다.	2019년 1월 15일
8.2.5	추가 보안 개선사항 이제 패키지에서 PowerShell 설치 관리자를 이용할 수 있습니다.	2018년 12월 12일
8.2.4	안정성 개선.	2018년 10월 2일
8.2.3	버그 수정 및 성능 향상. EBS 볼륨 ID를 EBS 볼륨의 디스크 일련 번호로 보고합니다. 이렇게 하면 S2D 같은 클러스터 시나리오가 활성화됩니다.	2018년 5월 29일
8.2.1	네트워크 및 스토리지 성능 개선 및 향상을 위한 다양한 수정. 이 버전이 설치되어 있는지 확인하려면 Windows 레지스트리 값 HKLM \Software\Amazon\PVDriver\Version 8.2.1을 참조하십시오.	2018년 3월 8일
7.4.6	AWS PV 드라이버의 복원력을 높여주는 안정성 수정.	2017년 4월 26일
7.4.3	Windows Server 2016에 대한 지원 추가됨. 지원되는 모든 Windows OS 버전에 대한 안정성 수정. *AWS PV 드라이버 버전 7.4.3의 서명은 2019년 3월 29일 만료됩니다. 최신 AWS PV 드라이버 업데이트를 권장합니다.	2016년 11월 18일
7.4.2	X1 인스턴스 유형의 지원에 대한 안정성 수정.	2016년 8월 2일
7.4.1	<ul style="list-style-type: none"> AWS PV 스토리지 드라이버의 성능 향상. AWS PV 스토리지 드라이버의 안정성 수정: 인스턴스가 버그 검사 코드 0x0000DEAD의 시스템 충돌을 일으키는 문제 수정됨. AWS PV 네트워크 드라이버의 안정성 수정. Windows Server 2008R2에 대한 지원 추가됨. 	2016년 7월 12일
7.3.2	<ul style="list-style-type: none"> 로깅 및 진단 개선됨. AWS PV 스토리지 드라이버의 안정성 수정. 경우에 따라 인스턴스에 디스크를 다시 연결한 후 디스크가 Windows에서 표시되지 않을 수 있습니다. Windows Server 2012에 대한 지원 추가됨. 	2015년 6월 24일

패키지 버전	세부 정보	릴리스 날짜
7.3.1	TRIM 업데이트: TRIM 요청과 관련하여 수정이 이루어졌습니다. 이 업데이트는 많은 수의 TRIM 요청을 관리할 때 인스턴스를 안정화하고 인스턴스 성능을 높입니다.	
7.3.0	TRIM 지원: 이제 AWS PV 드라이버가 TRIM 요청을 하이퍼바이저에 전송합니다. 기본 스토리지에서 TRIM(SSD)을 지원할 경우 휘발성 디스크가 TRIM 요청을 제대로 처리합니다. 2015년 3월을 기준으로 EBS 기반 스토리지에서 TRIM을 지원하지 않습니다.	
7.2.5	<ul style="list-style-type: none"> AWS PV 스토리지 드라이버의 안정성 수정: 경우에 따라 AWS PV 드라이버가 유효하지 않은 메모리를 역참조하고 시스템 오류를 유발할 수 있습니다. 충돌 덤프를 생성하는 중 안정성 수정: 경우에 따라 AWS PV 드라이버가 충돌 덤프를 작성할 때 경합 상태로 멈출 수 있습니다. 이 릴리스 전에는 드라이버를 강제로 중지하고 다시 시작하여 문제를 해결 할 수 있었지만 메모리 덤프가 손실되었습니다. 	
7.2.4	디바이스 ID 지속성: 이 드라이버 수정은 플랫폼 PCI 디바이스 ID를 숨기고 인스턴스가 이동된 경우에도 시스템이 항상 동일한 디바이스 ID를 표시하도록 강제 적용합니다. 이러한 수정 사항은 대체로 하이퍼바이저가 가상 디바이스를 표시하는 방법에 영향을 미치며, AWS PV 드라이버의 공동 설치 관리자에 대한 수정 사항도 포함하므로 시스템이 매핑된 가상 디바이스를 유지합니다.	
7.2.2	<ul style="list-style-type: none"> DSRM(디렉터리 서비스 복원 모드) 모드에서 AWS PV 드라이버 로드: 디렉터리 서비스 복원 모드는 Windows 서버 도메인 컨트롤러에 사용할 수 있는 안전 모드 부팅 옵션입니다. 네트워크 어댑터 디바이스가 다시 연결된 경우 영구 디바이스 ID: 이 수정 사항은 시스템이 MAC 주소 매핑을 확인하고 디바이스 ID를 유지하도록 강제 적용합니다. 또한 이를 통해 어댑터가 다시 연결된 경우 해당 정적 설정을 유지할 수 있습니다. 	
7.2.1	<ul style="list-style-type: none"> 안전 모드에서 실행: 드라이버가 안전 모드에서 로드되지 않는 문제를 해결했습니다. 이전에는 AWS PV 드라이버가 정상 실행 중인 시스템에서만 인스턴스화했습니다. Microsoft Windows 스토리지 폴에 디스크 추가: 이전에는 페이지 83 쿼리를 합성했으며, 수정 사항으로 페이지 83 지원이 비활성화됐습니다. 이는 PV 디스크가 유효한 클러스터 디스크가 아니므로 클러스터 환경에서 사용되는 스토리지 폴에 영향을 미치지 않습니다. 	
7.2.0	기본: AWS PV 기본 버전입니다.	

Citrix PV 드라이버

Citrix PV 드라이버는 %ProgramFiles%\Citrix\XenTools(32비트 인스턴트의 경우) 또는 %ProgramFiles(x86)%\Citrix\XenTools(64비트 인스턴스의 경우) 디렉터리에 저장됩니다.

Citrix PV 드라이버 구성 요소는 Windows 레지스트리의 HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services 아래에 나열됩니다. 이러한 드라이버 구성요소로는 xenevtchn, xeniface, xennet, Xennet6, xensvc, xenkbd 및 xenvif 등이 있습니다.

또한, Citrix에는 Windows 서비스를 구동하는 드라이버 구성요소인 XenGuestAgent가 있습니다. LiteAgent는 API 이벤트 종료 및 재시작과 같은 작업을 수행합니다. 사용자는 명령줄에서 Services.msc를 실행하여 서비스에 액세스하고 관리할 수 있습니다.

특정 워크로드 실행 시 네트워크 오류가 발생한 경우 Citrix PV 드라이버에서 TCP 오프로딩 기능을 비활성화해야 합니다. 자세한 내용은 [TCP 오프로드 \(p. 538\)](#) 단원을 참조하십시오.

RedHat PV 드라이버

RedHat 드라이버는 레거시 인스턴스에 사용할 수 있도록 지원되지만, 드라이버 제한 사항으로 인해 RAM이 12GB 이상인 새로운 인스턴스에서는 사용하지 않는 것이 좋습니다. RedHat 드라이버를 실행 중인 RAM이 12GB보다 큰 인스턴스는 부팅에 실패하고 액세스할 수 없는 상태가 될 수 있습니다. RedHat 드라이버를 Citrix PV 드라이버로 업그레이드한 다음 Citrix PV 드라이버를 AWS PV 드라이버로 업그레이드하는 것이 좋습니다.

RedHat 드라이버의 소스 파일은 %ProgramFiles%\RedHat(32비트 인스턴스의 경우) 또는 %ProgramFiles(x86)%\RedHat(64비트 인스턴스의 경우) 디렉터리에 저장됩니다. 드라이버로는 RedHat 반가상화 네트워크 드라이버인 `rhelnet`과 RedHat SCSI 미니포트 드라이버인 `rhe1scsi`의 두 가지가 있습니다.

알림 구독

새로운 EC2 Windows Driver 버전이 릴리스되면 이를 알리도록 Amazon SNS를 설정할 수 있습니다. 알림을 받으려면 다음 절차를 수행합니다.

콘솔에서 EC2 알림을 받으려면

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 필요한 경우 탐색 모음에서 리전을 미국 동부(버지니아 북부)로 변경하십시오. 구독을 신청하는 SNS 알림이 이 지역에 있기 때문에 이 지역을 선택해야 합니다.
3. 탐색 창에서 구독을 선택합니다.
4. 구독 생성을 선택합니다.
5. 구독 생성 대화 상자에서 다음 작업을 수행합니다.
 - a. TopicARN의 경우, 다음 Amazon 리소스 이름(ARN)을 복사합니다.
`arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers`
 - b. 프로토콜에서 `Email`을 선택합니다.
 - c. 엔드포인트에서 알림을 받을 이메일 주소를 입력합니다.
 - d. `Create subscription`을 선택합니다.
6. 확인 이메일이 발송됩니다. 이메일을 열고 지침에 따라 구독을 완료합니다.

새 EC2 Windows 드라이버가 릴리스될 때마다 구독자에게 알림이 전송됩니다. 이런 알림을 더 이상 받지 않기를 원하는 경우, 다음 절차를 수행해서 구독을 해제하십시오.

Amazon EC2 Windows 드라이버 알림을 구독 해제하려면

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 탐색 창에서 구독을 선택합니다.
3. 구독 확인란을 선택한 후 작업, 구독 삭제를 선택합니다. 확인 메시지가 나타나면 삭제를 선택합니다.

AWS CLI를 사용하여 EC2 알림을 받으려면

AWS CLI를 사용하여 EC2 알림을 받으려면 다음 명령을 사용합니다.

```
aws sns subscribe --topic-arn arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers --  
protocol email --notification-endpoint YourUserName@YourDomainName.ext
```

Windows PowerShell용 AWS Tools를 사용하여 EC2 알림을 받으려면

Windows PowerShell용 AWS Tools를 사용하여 EC2 알림을 받으려면 다음 명령을 사용합니다.

```
Connect-SNSNotification -TopicArn 'arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers'  
-Protocol email -Region us-east-1 -Endpoint 'YourUserName@YourDomainName.ext'
```

Windows 인스턴스에서 PV 드라이버 업그레이드

Windows 인스턴스에서 사용 중인 드라이버를 확인하려면 제어판에서 네트워크 연결(Network Connections)을 열고 로컬 영역 연결(Local Area Connection)을 확인합니다. 드라이버가 다음 중 하나에 해당하는지 확인합니다.

- AWS PV 네트워크 디바이스
- Citrix PV 이더넷 어댑터
- RedHat PV NIC 드라이버

아니면 pnputil -e 명령의 출력을 통해서도 확인이 가능합니다.

내용

- Windows Server 인스턴스 업그레이드(AWS PV 업그레이드) (p. 528)
- 도메인 컨트롤러 업그레이드(AWS PV 업그레이드) (p. 529)
- Windows Server 2008 및 2008 R2 인스턴스 업그레이드(Redhat에서 Citrix PV로 업그레이드) (p. 531)
- Citrix Xen 게스트 에이전트 서비스 업그레이드 (p. 533)

Windows Server 인스턴스 업그레이드(AWS PV 업그레이드)

다음 절차에 따라 AWS PV 드라이버의 현재 위치 업그레이드를 수행하거나, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016 또는 Windows Server 2019의 Citrix PV 드라이버에서 AWS PV 드라이버로 업그레이드할 수 있습니다. 이 업그레이드는 RedHat 드라이버 또는 다른 버전의 Windows Server에서는 제공되지 않습니다.

Important

인스턴스가 도메인 컨트롤러인 경우 [도메인 컨트롤러 업그레이드\(AWS PV 업그레이드\) \(p. 529\)](#) 단원을 참조하십시오. 도메인 컨트롤러 인스턴스의 업그레이드 프로세스는 Windows의 표준 버전과 다릅니다.

AWS PV 드라이버를 업그레이드하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 드라이버 업그레이드가 필요한 인스턴스를 선택하고 컨텍스트(오른쪽 클릭) 메뉴를 열고 인스턴스 상태를 선택한 후 중지를 선택합니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

4. 인스턴스가 중지되면 백업을 생성합니다. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고 이미지를 선택한 다음 이미지 생성을 선택합니다.
5. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴에서 인스턴스 상태를 선택한 다음 시작을 선택합니다.
6. 원격 데스크톱을 사용하여 인스턴스에 연결하고 인스턴스를 업그레이드할 준비를 합니다. 이 업그레이드를 수행하기 전에 모든 비 시스템 디스크를 오프라인으로 하고 모든 드라이브 문자 매핑을 디스크 관리의 보조 디스크에 기록하는 것이 좋습니다. AWS PV 드라이버의 현재 위치 업그레이드를 수행할 경우에는 이 단계가 필요하지 않습니다. 또한 서비스 콘솔에서 필수적이지 않은 서비스를 수동 시작으로 설정하는 것이 좋습니다.

7. 최신 드라이버 패키지를 인스턴스로 [다운로드](#)합니다.

또는 다음 PowerShell 명령을 실행합니다.

```
PS C:\>invoke-webrequest https://s3.amazonaws.com/ec2-windows-drivers-downloads/AWSPV/Latest/AWSPVDriver.zip -outfile $env:USERPROFILE\pv_driver.zip  
expand-archive $env:UserProfile\pv_driver.zip -DestinationPath  
$env:UserProfile\pv_drivers
```

8. 폴더의 내용 압축을 풀고 AWSPVDriverSetup.msi를 실행합니다.

MSI를 실행하면 인스턴스가 자동으로 재부팅되고 드라이버를 업그레이드합니다. 최대 15분 동안 인스턴스를 사용할 수 없습니다. 업그레이드를 완료하고 인스턴스가 Amazon EC2 콘솔에서 두 상태 확인을 모두 통과하면 원격 데스크톱을 사용하여 인스턴스에 연결한 다음, PowerShell 명령을 실행하여 새 드라이버가 설치되었는지 확인할 수 있습니다.

```
Get-ItemProperty HKLM:\SOFTWARE\Amazon\PVDriver
```

드라이버 버전이 드라이버 버전 기록 표에 나열된 최신 버전과 동일한지 확인합니다. 자세한 내용은 [AWS PV 드라이버 패키지 내역 \(p. 524\)](#) 개방형 디스크 관리(Open Disk Management)를 참조하여 모든 오프라인 보조 볼륨을 검토하고 6단계에서 기록한 드라이버 문자에 따라 이를 온라인으로 전환합니다.

Citrix PV 드라이버용 Netsh를 사용하여 이전에 [TCP 오프로드 \(p. 538\)](#)을 비활성화한 경우 AWS PV 드라이버로 업그레이드한 후 이 기능을 다시 활성화하는 것이 좋습니다. Citrix 드라이버의 TCP 오프로딩 문제가 AWS PV 드라이버에는 없습니다. 따라서 TCP 오프로딩은 AWS PV 드라이버를 사용할 때 더 우수한 성능을 제공합니다.

이전에 네트워크 인터페이스에 정적 IP 주소 또는 DNS 구성은 적용한 경우 AWS PV 드라이버 업그레이드 이후 네트워크 인터페이스에 정적 IP 주소 또는 DNS 구성은 다시 적용해야 합니다.

도메인 컨트롤러 업그레이드(AWS PV 업그레이드)

도메인 컨트롤러에서 다음 절차를 사용하여 AWS PV 드라이버의 현재 위치 업그레이드를 수행하거나 Citrix PV 드라이버를 AWS PV 드라이버로 업그레이드합니다.

도메인 컨트롤러를 업그레이드하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 드라이버 업그레이드가 필요한 인스턴스를 선택하고 컨텍스트(오른쪽 클릭) 메뉴를 열고 인스턴스 상태를 선택한 후 중지를 선택합니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

4. 인스턴스가 중지되면 백업을 생성합니다. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고 이미지를 선택한 다음 이미지 생성을 선택합니다.
5. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴에서 인스턴스 상태를 선택한 다음 시작을 선택합니다.
6. 다음 명령을 실행하여 Windows를 DSRM(Directory Services Restore Mode)으로 부팅하도록 구성합니다.

```
bcdedit /set {default} safeboot dsrepair
```

PowerShell:

```
PS C:\> bcdedit /set "{default}" safeboot dsrepair
```

Warning

이 명령을 실행하기 전에 DSRM 암호를 알고 있는지 확인합니다. 업그레이드가 완료되고 인스턴스가 자동으로 재부팅된 후 인스턴스에 로그인할 수 있도록 이 정보가 필요합니다.

시스템을 DSRM으로 부팅해야 합니다. 업그레이드 유ти리티가 AWS PV 드라이버를 설치하기 위해 Citrix PV 스토리지 드라이버를 제거하기 때문입니다. 따라서 모든 드라이버 문자 및 폴더 매핑을 디스크 관리의 보조 디스크에 기록하는 것이 좋습니다. Citrix PV 스토리지 드라이버가 존재하지 않으면 보조 드라이브가 검색되지 않습니다. 보조 디스크가 검색되지 않으므로 보조 드라이브의 NTDS 폴더를 사용하는 도메인 컨트롤러가 부팅되지 않습니다.

Warning

이 명령을 실행한 후 시스템을 수동으로 재부팅하지 마십시오. Citrix PV 드라이버는 DSRM을 지원하지 않으므로 시스템에 접속할 수 없습니다.

7. 다음 명령을 실행하여 **DisableDCCheck**를 레지스트리에 추가합니다.

```
reg add HKLM\SOFTWARE\Wow6432Node\Amazon\AWSPVDriverSetup /v DisableDCCheck /t REG_SZ /d true
```

8. 최신 드라이버 패키지를 인스턴스로 [다운로드](#)합니다.
9. 폴더의 내용 압축을 풀고 AWSPVDriverSetup.msi를 실행합니다.

MSI를 실행하면 인스턴스가 자동으로 재부팅되고 드라이버를 업그레이드합니다. 최대 15분 동안 인스턴스를 사용할 수 없습니다.

10. 업그레이드를 완료하고 인스턴스가 Amazon EC2 콘솔에서 두 상태 확인을 모두 통과하면 원격 데스크톱을 사용하여 인스턴스에 연결합니다. 개방형 디스크 관리에서 모든 오프라인 보조 볼륨을 검토하고 6 단계에서 기록한 드라이버 문자와 폴더 매핑에 따라 이를 온라인으로 전환합니다.

Important

사용자 이름을 hostname\administrator 형식으로 지정하여 인스턴스와 연결해야 합니다. 형식 (예: Win2k12TestBox\administrator)으로 지정하여 인스턴스에 연결해야 합니다.

11. 다음 명령을 실행하여 DSRM 부팅 구성을 제거합니다.

```
bcdedit /deletevalue safeboot
```

12. 인스턴스를 재부팅합니다.
13. 업그레이드 프로세스를 완료하려면 새 드라이버가 설치되었는지 확인합니다. 디바이스 관리자(Device Manager)의 스토리지 컨트롤러(Storage Controllers) 아래에서 AWS PV Storage Host Adapter를 찾습니다. 드라이버 버전이 드라이버 버전 표에 나열된 최신 버전과 동일한지 확인합니다. 자세한 내용은 [AWS PV 드라이버 패키지 내역 \(p. 524\)](#) 단원을 참조하십시오.

14. 다음 명령을 실행하여 레지스트리에서 **DisableDCCheck**를 삭제합니다.

```
reg delete HKLM\SOFTWARE\Wow6432Node\Amazon\AWSPVDriverSetup /v DisableDCCheck
```

Note

Citrix PV 드라이버용 Netsh를 사용하여 이전에 [TCP 오프로드 \(p. 538\)](#)을 비활성화한 경우 AWS PV 드라이버로 업그레이드한 후 이 기능을 다시 활성화하는 것이 좋습니다. Citrix 드라이버의 TCP 오프로딩 문제가 AWS PV 드라이버에는 없습니다. 따라서 TCP 오프로딩은 AWS PV 드라이버를 사용할 때 더 우수한 성능을 제공합니다.

Windows Server 2008 및 2008 R2 인스턴스 업그레이드(Redhat에서 Citrix PV로 업그레이드)

RedHat 드라이버를 Citrix PV 드라이버로 업그레이드하기 전에 다음을 수행해야 합니다.

- 최신 버전의 EC2Config 서비스를 설치합니다. 자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 단원을 참조하십시오.
- Windows PowerShell 2.0이 설치되었는지 확인합니다. 설치된 버전을 확인하려면 PowerShell 창에서 다음 명령을 실행합니다.

```
PS C:\> $PSVersionTable.PSVersion
```

2.0 버전을 설치해야 하는 경우 Microsoft 지원의 [Windows Management Framework Core Package \(Windows PowerShell 2.0 and WinRM\)](#)를 참조하십시오.

- 중요한 정보를 인스턴스에 백업하거나 인스턴스에서 AMI를 생성합니다. AMI 생성에 대한 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오. AMI를 생성하는 경우 다음을 수행해야 합니다.
 - 암호를 입력합니다.
 - Sysprep 도구를 직접 실행하거나 EC2Config 서비스를 사용하지 마십시오.
 - 이더넷 어댑터를 설정하여 DHCP를 통해 IP 주소를 자동으로 할당받습니다. 자세한 내용은 Microsoft TechNet Library의 [TCP/IP 설정 구성](#)을 참조하십시오.

Redhat 드라이버를 업그레이드하려면

- 인스턴스 연결 후 로컬 관리자로 로그인합니다. 인스턴스 연결에 대한 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 주제를 참조하십시오.
- 인스턴스에서 Citrix PV 업그레이드 패키지를 [다운로드](#)합니다.
- 원하는 위치에서 업그레이드 패키지 콘텐츠의 압축을 풁니다.
- Upgrade.bat 파일을 두 번 클릭합니다. 보안 경고가 나타나면 실행(Run)을 선택합니다.
- 드라이버 업그레이드(Upgrade Drivers) 대화 상자에서 정보를 확인한 다음 업그레이드를 시작하려면 예(Yes)를 선택합니다.
- Red Hat Paravirtualized Xen Drivers for Windows 설치 제거 프로그램(Red Hat Paravirtualized Xen Drivers for Windows uninstaller) 대화 상자에서 예(Yes)를 선택하여 RedHat 소프트웨어를 제거합니다. 그러면 인스턴스가 재부팅됩니다.

Note

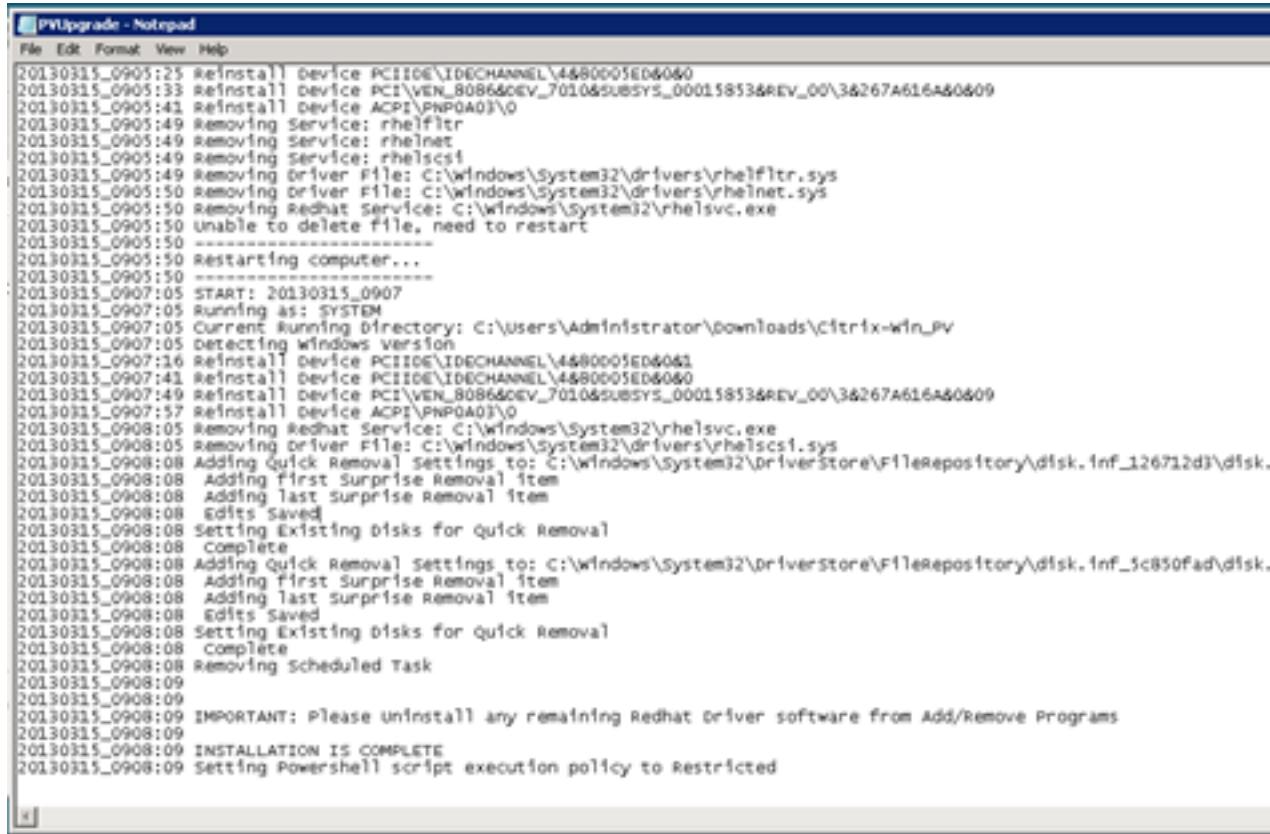
제거 대화 상자가 나타나지 않으면 Windows 작업 표시줄에서 Red Hat Paravirtualize를 선택합니다.



- 인스턴스가 재부팅되고 사용할 준비가 되었는지 확인합니다.
 - <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
 - 인스턴스(Instances) 페이지에서 인스턴스를 마우스 오른쪽 단추로 클릭한 다음 시스템 로그 가져오기(Get System Log)를 선택합니다.
 - 그러면 업그레이드 작업이 수행되며 서버가 3회 또는 4회 재시작됩니다. 이 작업은 로그 파일에 표시되는 횟수 Windows is Ready to use로 확인할 수 있습니다.

```
Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
Ec2Config service v2.1.9.0
RedHat PV NIC Driver v1.3.10.0
2013/03/15 17:11:01Z: Waiting for meta-data accessibility...
2013/03/15 17:11:02Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BF64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
<Username>Administrator</Username>
<Password>
L79ThJPF8LyIL38I2ht0FBrjet3vnT2csTiU/XGVMRCH7kQtBnznAnXrKdlsirXlx19BwVMsd9b38jFJqv01IUpgNNJRZoCDc7IbUw
</Password>
2013/03/15 17:11:30Z: Product activation was successful.
2013/03/15 17:11:32Z: Message: Windows is Ready to use
Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
Ec2Config service v2.1.9.0
2013/03/15 21:04:24Z: There was an exception writing driver information to console: System.Exception: 
at Ec2Config.Service1.Go()
2013/03/15 21:04:35Z: Waiting for meta-data accessibility...
2013/03/15 21:04:40Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BF64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
2013/03/15 21:05:08Z: Product activation was successful.
2013/03/15 21:05:09Z: Message: Windows is Ready to use
Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
Ec2Config service v2.1.9.0
Citrix PV Ethernet Adapter v5.9.960.49119
2013/03/15 21:07:20Z: Waiting for meta-data accessibility...
2013/03/15 21:07:21Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BF64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
2013/03/15 21:07:27Z: Message: Windows is Ready to use
```

8. 인스턴스 연결 후 로컬 관리자로 로그인합니다.
9. Red Hat Paravirtualized Xen Drivers for Windows 설치 제거 프로그램(Red Hat Paravirtualized Xen Drivers for Windows uninstaller) 대화 상자를 닫습니다.
10. 설치가 완료되었는지 확인합니다. 이전에 압축을 푼 Citrix-WIN_PV 폴더로 이동한 다음 PVUpgrade.log 파일을 열고 INSTALLATION IS COMPLETE라는 텍스트가 있는지 확인합니다.



```
PVUpgrade - Notepad
File Edit Format View Help
20130315_0905:25 reinstall device PCIIDE\IDECHANNEL\4&80005ED&0
20130315_0905:33 reinstall device PCI\VEN_8086&DEV_7010&SUBSYS_00015853&REV_00\3&267A616A&0&0
20130315_0905:41 reinstall device ACPI\PNP0403\0
20130315_0905:49 removing Service: rhelflt
20130315_0905:49 removing Service: rhelnet
20130315_0905:49 removing Service: rhelscsf
20130315_0905:49 Removing Driver File: c:\windows\System32\drivers\rhelflt.sys
20130315_0905:50 Removing Driver File: C:\windows\System32\drivers\rhelnet.sys
20130315_0905:50 Removing Redhat Service: C:\windows\System32\rhelsvc.exe
20130315_0905:50 Unable to delete file, need to restart
20130315_0905:50 -----
20130315_0905:50 Restarting computer...
20130315_0905:50 -----
20130315_0907:05 START: 20130315_0907
20130315_0907:05 Running as: SYSTEM
20130315_0907:05 Current Running Directory: C:\Users\Administrator\Downloads\cfrtrix-wfn_PV
20130315_0907:05 Detecting Windows version
20130315_0907:16 reinstall device PCIIDE\IDECHANNEL\4&80005ED&0
20130315_0907:141 reinstall device PCIIDE\IDECHANNEL\4&80005ED&0
20130315_0907:49 reinstall device PCI\VEN_8086&DEV_7010&SUBSYS_00015853&REV_00\3&267A616A&0&0
20130315_0907:57 reinstall device ACPI\PNP0403\0
20130315_0908:05 Removing Redhat Service: C:\windows\System32\rhelsvc.exe
20130315_0908:05 Removing Driver File: C:\windows\System32\drivers\rhelscsf.sys
20130315_0908:08 Adding Quick Removal Settings to: C:\windows\System32\DriverStore\FileRepository\disk.inf_126712d3\disk
20130315_0908:08 Adding First Surprise Removal Item
20130315_0908:08 Adding Last Surprise Removal Item
20130315_0908:08 Edits Saved
20130315_0908:08 Setting Existing Disks for Quick Removal
20130315_0908:08 Complete
20130315_0908:08 Adding Quick Removal Settings to: C:\windows\System32\DriverStore\FileRepository\disk.inf_5c850fad\disk
20130315_0908:08 Adding First Surprise Removal Item
20130315_0908:08 Adding Last Surprise Removal Item
20130315_0908:08 Edits Saved
20130315_0908:08 Setting Existing Disks for Quick Removal
20130315_0908:08 Complete
20130315_0908:08 Removing Scheduled Task
20130315_0908:09
20130315_0908:09 -----
20130315_0908:09 IMPORTANT: Please uninstall any remaining Redhat driver software from Add/Remove Programs
20130315_0908:09
20130315_0908:09 INSTALLATION IS COMPLETE
20130315_0908:09 Setting Powershell script execution policy to Restricted
```

Citrix Xen 게스트 에이전트 서비스 업그레이드

Windows Server에서 Citrix PV 드라이버를 사용하는 경우 Citrix Xen 게스트 에이전트 서비스를 업그레이드 할 수 있습니다. 이 Windows 서비스는 API의 종료 및 재시작 이벤트와 같은 작업을 처리합니다. 인스턴스에서 Citrix PV 드라이버를 실행하는 경우 Windows Server의 모든 버전에서 이 업그레이드 패키지를 실행할 수 있습니다.

Important

Windows Server 2008 R2 이상의 경우 게스트 에이전트 업데이트가 포함된 AWS PV 드라이버로 업그레이드하는 것이 좋습니다.

드라이버 업그레이드를 시작하기 전, 중요한 정보를 인스턴스에 백업하거나 인스턴스에서 AMI를 생성합니다. AMI 생성에 대한 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오. AMI를 생성하는 경우 다음을 수행해야 합니다.

- EC2Config 서비스에서 Sysprep 도구를 활성화하지 마십시오.
- 암호를 입력합니다.
- 이더넷 어댑터를 DHCP로 설정합니다.

Citrix Xen 게스트 에이전트 서비스를 업그레이드하려면

1. 인스턴스 연결 후 로컬 관리자로 로그인합니다. 인스턴스 연결에 대한 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 주제를 참조하십시오.
2. 인스턴스에서 Citrix 업그레이드 패키지를 [다운로드](#)합니다.
3. 원하는 위치에서 업그레이드 패키지 콘텐츠의 압축을 풉니다.

4. Upgrade.bat 파일을 두 번 클릭합니다. 보안 경고가 나타나면 실행(Run)을 선택합니다.
5. 드라이버 업그레이드(Upgrade Drivers) 대화 상자에서 정보를 확인한 다음 업그레이드를 시작하려면 예(Yes)를 선택합니다.
6. 업그레이드가 완료되면 PVupgrade.log라는 텍스트가 있는 UPGRADE IS COMPLETE 파일이 열립니다.
7. 인스턴스를 재부팅합니다.

PV 드라이버 문제 해결

다음은 기존 Amazon EC2 이미지 및 PV 드라이버에 발생할 수 있는 문제에 대한 해결 방법입니다.

목차

- [인스턴스를 재부팅한 후 Windows Server 2012 R2에서 네트워크와 스토리지 연결이 끊김 \(p. 534\)](#)
- [TCP 오프로드 \(p. 538\)](#)
- [시간 동기화 \(p. 539\)](#)

인스턴스를 재부팅한 후 Windows Server 2012 R2에서 네트워크와 스토리지 연결이 끊김

Important

이 문제는 2014년 9월 이전에 제공된 AMI에만 발생합니다

2014년 9월 10일 이전에 제공된 Windows Server 2012 R2 Amazon 머신 이미지(AMI)에서는 인스턴스 재부팅 후 네트워크와 스토리지 연결이 끊길 수 있습니다. AWS Management 콘솔 시스템 로그의 오류는 "콘솔 출력에 대한 PV 드라이버 세부 정보를 감지하는 중 장애 발생"으로 표시됩니다. 플러그-앤플레이 클린업 기능으로 인해 연결이 끊깁니다. 이 기능은 30일마다 비활성 시스템 디바이스를 검색하고 비활성화합니다. 기능이 EC2 네트워크 디바이스를 비활성 상태로 잘못 식별하고 시스템에서 제거합니다. 이러한 경우 인스턴스가 재부팅 후 네트워크 연결이 끊깁니다.

주의 대상 시스템은 이 문제에 의해 영향을 받을 수 있으며, 현재 위치 드라이버 업그레이드를 다운로드하고 실행할 수 있습니다. 현재 위치 드라이버 업그레이드를 수행할 수 없는 경우에는 헬퍼 스크립트를 실행할 수 있습니다. 스크립트가 인스턴스가 영향을 받는지 여부를 판별합니다. 영향을 받고 있고 Amazon EC2 네트워크 디바이스가 제거되지 않은 경우 스크립트가 플러그-앤플레이 클린업 스캔을 비활성화합니다. 네트워크 디바이스가 제거되지 않은 경우 스크립트는 디바이스를 복구하고, 플러그-앤플레이 클린업 스캔을 비활성화하며, 네트워크 연결이 활성화된 상태로 인스턴스가 재부팅되도록 허용합니다.

목차

- [문제 해결 방법의 선택 \(p. 534\)](#)
- [방법 1 - 향상된 네트워킹 기능 \(p. 535\)](#)
- [방법 2 - 레지스트리 구성 \(p. 535\)](#)
- [수정 스크립트 실행 \(p. 537\)](#)

문제 해결 방법의 선택

문제가 발생한 인스턴스에 대한 네트워크 및 스토리지 연결을 복구하는 방법은 두 가지가 있습니다. 다음 방법 중 한 가지를 선택하십시오.

방법	사전 조건	절차 개요
방법 1 - 향상된 네트워킹 기능	향상된 네트워킹 기능은 C3 인스턴스 유형을 사용하는 Virtual Private Cloud(VPC)에서만 가능	서버 인스턴스 유형을 C3 인스턴스로 변경합니다. 그러면 향상된 네트워킹 기능을 통해 문제가 발

방법	사전 조건	절차 개요
	합니다. 서버에서 현재 사용하는 유형이 C3 인스턴스가 아닐 때는 잠시 C3 인스턴스로 변경해야 합니다.	생한 인스턴스에 연결하여 문제를 해결할 수 있습니다. 문제를 해결 한 후에는 원래 인스턴스 유형으로 다시 바꿀 수 있습니다. 일반적으로 이 방법은 방법 2보다 빠를 뿐만 아니라 사용자 오류의 가능성이 낮습니다. C3 인스턴스가 실행 중일 때는 변경 사항이 추가로 발생하기 마련입니다.
방법 2 - 레지스트리 구성	보조 서버 생성 및 액세스 기능. 레지스트리 설정 변경 기능.	문제가 발생한 인스턴스에서 루트 볼륨을 분리하여 다른 인스턴스에 연결한 다음 레지스트리 설정을 변경합니다. 보조 서버가 실행 중 일 때는 변경 사항이 추가로 발생하기 마련입니다. 이 방법은 방법 1보다 느리지만 방법 1로 문제를 해결하지 못하는 상황에서 효과가 있었습니다.

방법 1 - 향상된 네트워킹 기능

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 문제가 발생한 인스턴스를 찾습니다. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고, 인스턴스 상태를 선택한 다음 중지를 선택합니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

4. 인스턴스가 중지되면 백업을 생성합니다. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고 이미지를 선택한 다음 이미지 생성을 선택합니다.
5. 인스턴스 유형을 C3 인스턴스로 **변경합니다**.
6. 인스턴스를 **시작** 합니다.
7. 원격 데스크톱을 사용하여 인스턴스에 연결한 다음 AWS PV 드라이버 업그레이드 패키지를 인스턴스에 **다운로드** 합니다.
8. 폴더의 내용 압축을 풀고 **AWS PV Driver Setup.msi**를 실행합니다.

MSI를 실행하면 인스턴스가 자동으로 재부팅되고 드라이버를 업그레이드합니다. 최대 15분 동안 인스턴스를 사용할 수 없습니다.

9. 업그레이드를 완료하고 인스턴스가 Amazon EC2 콘솔에서 두 상태 확인을 모두 통과하면 원격 데스크톱을 사용하여 인스턴스에 연결하고 새 드라이버가 설치되었는지 확인합니다. 디바이스 관리자의 스토리지 컨트롤러(Storage Controllers)에서 AWS PV Storage Host Adapter를 찾습니다. 드라이버 버전이 드라이버 버전 기록 표에 나열된 최신 버전과 동일한지 확인합니다. 자세한 내용은 [AWS PV 드라이버 패키지 내역 \(p. 524\)](#) 단원을 참조하십시오.
10. 인스턴스를 중단하고 원래 인스턴스 유형으로 다시 바꿉니다.
11. 원래 인스턴스를 시작하여 정상적인 사용을 재개합니다.

방법 2 - 레지스트리 구성

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 인스턴스를 선택합니다.
3. 문제가 발생한 인스턴스를 찾습니다. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고, 인스턴스 상태를 선택한 다음 중지를 선택합니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

4. 인스턴스 시작을 선택하고 문제가 발생한 인스턴스와 동일한 가용 영역에 임시 Windows Server 2008 또는 Windows Server 2012 인스턴스를 생성합니다. Windows Server 2012 R2 인스턴스를 생성하지 마십시오.

Important

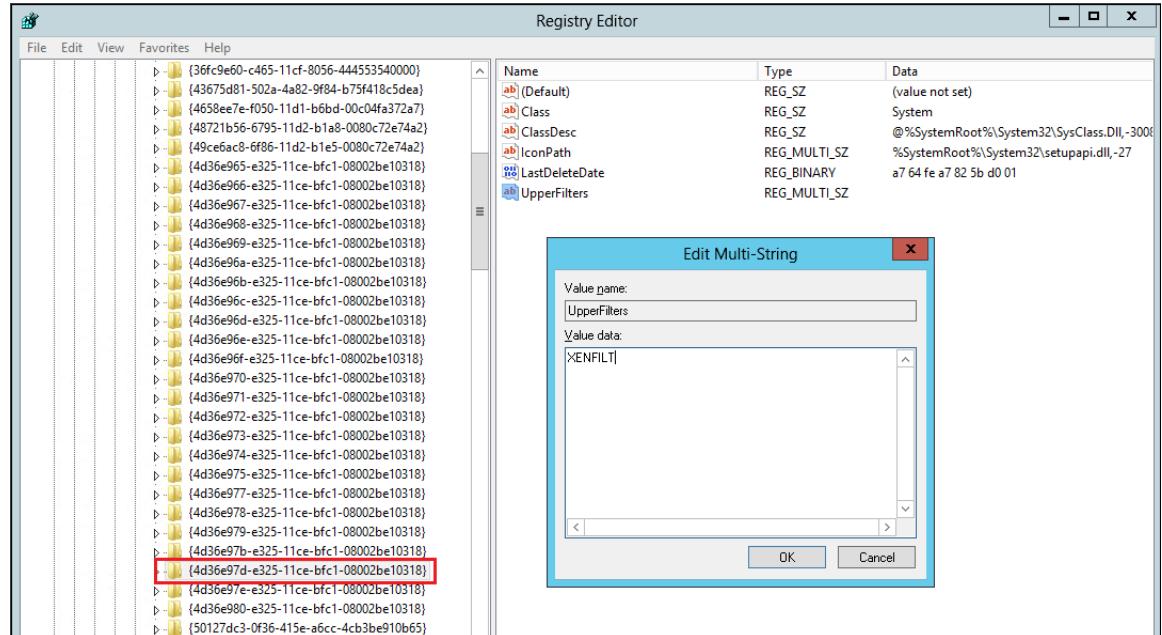
문제가 발생한 인스턴스와 동일한 가용 영역에서 인스턴스를 생성하지 않는 경우에는 문제가 발생한 인스턴스의 루트 볼륨을 새 인스턴스에 연결할 수 없습니다.

5. 탐색 창에서 볼륨을 선택합니다.
6. 문제가 발생한 인스턴스의 루트 볼륨을 찾습니다. [볼륨을 분리 \(p. 964\)](#)하고 이전에 생성한 임시 인스턴스에 [볼륨을 연결 \(p. 950\)](#)합니다. 기본 디바이스 이름(xvdf)으로 연결합니다.
7. 원격 데스크톱을 사용하여 임시 인스턴스에 연결한 후 디스크 관리 유ти리티를 사용하여 [볼륨을 사용할 수 있도록 지정 \(p. 951\)](#)합니다.
8. 임시 인스턴스에서 실행 대화 상자를 열고 **regedit**를 입력한 다음 Enter 키를 누릅니다.
9. 레지스트리 편집기 탐색 창에서 HKEY_Local_Machine을 선택한 후 파일(File) 메뉴에서 Hive 로드(Load Hive)를 선택합니다.
10. Hive 로드(Load Hive) 대화 상자에서 Affected Volume\Windows\System32\config\System을 탐색하고 키 이름(Key Name) 대화 상자에 임시 이름을 입력합니다. 예를 들어, OldSys를 입력합니다.
11. 레지스트리 편집기의 탐색 창에서 다음 키를 찾습니다.

HKEY_LOCAL_MACHINE\your_temporary_key_name\ControlSet001\Control\Class\4d36e97d-e325-11ce-bfc1-08002be10318

HKEY_LOCAL_MACHINE\your_temporary_key_name\ControlSet001\Control\Class\4d36e96a-e325-11ce-bfc1-08002be10318

12. 각각의 키에 대해 UpperFilters를 두 번 클릭하고 XENFILT 값을 입력한 후 확인(OK)을 선택합니다.



13. 다음 키를 찾습니다.

HKEY_LOCAL_MACHINE\your_temporary_key_name\ControlSet001\Services\XENBUS
\Parameters

14. ActiveDevice 이름 및 다음 값을 사용하여 새 문자열(REG_SZ)을 생성합니다.

PCI\VEN_5853&DEV_0001&SUBSYS_00015853&REV_01

15. 다음 키를 찾습니다.

HKEY_LOCAL_MACHINE\your_temporary_key_name\ControlSet001\Services\XENBUS

16. 수(Count)를 0에서 1로 변경합니다.

17. 다음 키를 찾아 삭제합니다.

HKEY_LOCAL_MACHINE\your_temporary_key_name\ControlSet001\Services\xenvbd
\StartOverride

HKEY_LOCAL_MACHINE\your_temporary_key_name\ControlSet001\Services\xenfilt
\StartOverride

18. 레지스트리 편집기 탐색 창에서 처음 레지스트리 편집기를 열 때 생성한 임시 키를 선택합니다.

19. 파일(File) 메뉴에서 Hive 언로드(Unload Hive)를 선택합니다.

20. 디스크 관리(Disk Management) 유ти리티에서 이전에 연결한 드라이브를 선택하고 컨텍스트(오른쪽 클릭) 메뉴를 열고 오프라인(Offline)을 선택합니다.

21. Amazon EC2 콘솔에서 임시 인스턴스로부터 문제가 발생한 볼륨을 분리하고 디바이스 이름 /dev/sda1 을 사용하여 Windows Server 2012 R2 인스턴스에 다시 연결합니다. 볼륨을 루트 볼륨으로 지정하려면 이 디바이스 이름을 지정해야 합니다.

22. 인스턴스를 시작합니다.

23. 원격 데스크톱을 사용하여 인스턴스에 연결한 다음 AWS PV 드라이버 업그레이드 패키지를 인스턴스에 다운로드합니다.

24. 폴더의 내용 압축을 풀고 AWSPVDriverSetup.msi를 실행합니다.

MSI를 실행하면 인스턴스가 자동으로 재부팅되고 드라이버를 업그레이드합니다. 최대 15분 동안 인스턴스를 사용할 수 없습니다.

25. 업그레이드를 완료하고 인스턴스가 Amazon EC2 콘솔에서 두 상태 확인을 모두 통과하면 원격 데스크톱을 사용하여 인스턴스에 연결하고 새 드라이버가 설치되었는지 확인합니다. 디바이스 관리자의 스토리지 컨트롤러(Storage Controllers)에서 AWS PV Storage Host Adapter를 찾습니다. 드라이버 버전이 드라이버 버전 기록 표에 나열된 최신 버전과 동일한지 확인합니다. 자세한 내용은 [AWS PV 드라이버 패키지 내역 \(p. 524\)](#) 단원을 참조하십시오.

26. 앞에서 임시로 생성한 인스턴스는 이 절차에서 삭제하거나 중단합니다.

수정 스크립트 실행

현재 위치 드라이버 업그레이드를 수행하거나 새로운 인스턴스를 마이그레이션할 수 없는 경우 수정 스크립트를 실행하여 플러그-앤플레이 클린업 작업에서 발생한 문제를 수정할 수 있습니다.

수정 스크립트를 실행하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 수정 스크립트를 실행할 인스턴스를 선택합니다. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고, 인스턴스 상태를 선택한 다음 중지를 선택합니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

4. 인스턴스가 중지되면 백업을 생성합니다. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고 이미지를 선택한 다음 이미지 생성을 선택합니다.
5. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고, 인스턴스 상태를 선택한 다음 시작을 선택합니다.
6. 원격 데스크톱을 사용하여 인스턴스에 연결하고 RemediateDriverIssue.zip 폴더를 인스턴스에 [다운로드](#)합니다.
7. 폴더 내용을 추출합니다.
8. Readme.txt 파일의 지침에 따라 수정 스크립트를 실행합니다. 파일은 RemediateDriverIssue.zip을 추출한 폴더에 있습니다.

TCP 오프로드

Important

AWS PV 또는 Intel 네트워크 드라이버를 실행하는 인스턴스에는 이 문제가 적용되지 않습니다.

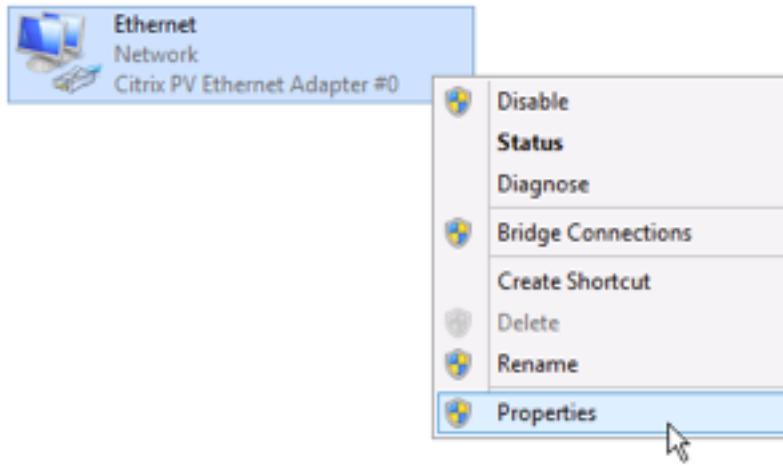
기본적으로, Windows AMI의 Citrix PV 드라이버에서는 TCP 오프로드가 활성화됩니다. 예를 들어, 특정 SQL 워크로드 등 전송 수준 오류 또는 패킷 전송 오류(Windows 성능 모니터에서 확인 가능)가 발생한 경우 이 기능을 비활성화해야 합니다.

Warning

TCP 오프로드를 비활성화하면 인스턴스의 네트워크 성능이 감소합니다.

Windows Server 2012 및 2008에서 TCP 오프로드를 비활성화하려면,

1. 인스턴스 연결 후 로컬 관리자로 로그인합니다.
2. Windows Server 2012의 경우 Ctrl+Esc를 눌러 시작 화면에 액세스한 다음 제어판을 선택합니다. Windows Server 2008의 경우 시작 화면에 액세스한 다음 제어판을 선택합니다.
3. 네트워크 및 인터넷, 그 다음 네트워크 및 공유 센터를 선택합니다.
4. 어댑터 설정 변경을 선택합니다.
5. Citrix PV Ethernet Adapter #0를 마우스 오른쪽 단추로 클릭한 다음 속성을 선택합니다.



6. 로컬 영역 연결 속성 대화 상자에서 구성을 선택하여 Citrix PV Ethernet Adapter #0 속성 대화 상자를 엽니다.
7. 고급 탭에서 TCP/UDP 체크섬 값 수정 외에 각 속성을 해제합니다. 속성을 해제하려면 속성 중에서 선택하여 값에서 비활성화를 선택합니다.
8. 확인을 선택합니다.
9. 명령 프롬프트 창에서 다음 명령을 실행합니다.

```
netsh int ip set global taskoffload=disabled
netsh int tcp set global chimney=disabled
netsh int tcp set global rss=disabled
netsh int tcp set global netdma=disabled
```

10. 인스턴스를 재부팅합니다.

시간 동기화

2013.02.13 Windows AMI 릴리스 이전에는 Citrix Xen 게스트 에이전트에서 시스템 시간이 올바르지 않게 설정될 수 있었습니다. 이로 인해 DHCP 임대 만료가 발생했습니다. 인스턴스 연결에 문제가 있으면 이 에이전트를 업데이트하십시오.

Citrix Xen 게스트 에이전트의 업데이트 여부를 확인하려면 C:\Program Files\Citrix\xenGuestAgent.exe 파일의 날짜가 2013년 3월인지 확인합니다. 이 파일의 날짜가 그 이전인 경우 Citrix Xen 게스트 에이전트 서비스를 업데이트합니다. 자세한 내용은 [Citrix Xen 게스트 에이전트 서비스 업그레이드 \(p. 533\)](#) 단원을 참조하십시오.

Windows 인스턴스의 AWS NVMe 드라이버

EBS 볼륨 및 인스턴스 스토어 볼륨은 [Nitro 기반 인스턴스 \(p. 118\)](#)에서 NVMe 블록 디바이스로 표시됩니다. NVMe 블록 디바이스를 사용하려면 AWS NVMe 드라이버가 설치되어 있어야 합니다. Windows Server 2008 R2 이상용 최신 AWS Windows AMI에는 필수 AWS NVMe 드라이버가 포함되어 있습니다.

EBS 및 NVMe에 대한 자세한 내용은 [Windows 인스턴스의 Amazon EBS 및 NVMe \(p. 1048\)](#) 단원을 참조하십시오. SSD 인스턴스 스토어 및 NVMe에 대한 자세한 내용은 [SSD 인스턴스 스토어 볼륨 \(p. 1101\)](#) 단원을 참조하십시오.

AWS NVMe 드라이버 설치 또는 업그레이드

Amazon이 제공하는 최신 AWS Windows AMI를 사용하고 있지 않은 경우에는 다음 절차를 이용하여 최신 AWS NVMe 드라이버를 설치합니다. 인스턴스를 재부팅하기 편리한 시간에 이 업데이트를 수행해야 합니다. 설치 스크립트에 따라 인스턴스가 재부팅되거나 최종 단계로 인스턴스를 재부팅해야 합니다.

사전 요구사항

PowerShell 3.0 이상

최신 AWS NVMe 드라이버를 다운로드하고 설치하려면

1. 인스턴스 연결 후 로컬 관리자로 로그인합니다.
2. 다음 옵션 중 하나를 사용하여 드라이버를 다운로드하고 압축을 풉니다.
 - 브라우저 사용:
 - a. 최신 드라이버 패키지를 인스턴스로 [다운로드](#)합니다.
 - b. ZIP 아카이브를 추출합니다.
 - PowerShell 사용:

```
invoke-webrequest https://s3.amazonaws.com/ec2-windows-drivers-downloads/NVMe/
Latest/AWSNVMe.zip -outfile $env:USERPROFILE\nvme_driver.zip
expand-archive $env:UserProfile\nvme_driver.zip -DestinationPath $env:UserProfile
\nvme_driver
```
3. install.ps1 PowerShell 스크립트를 실행하여 드라이버를 설치합니다. 오류가 발생하면 PowerShell 3.0 이상을 사용하고 있는지 확인합니다.
4. 설치 관리자에서 인스턴스가 재부팅되지 않는 경우 인스턴스를 재부팅합니다.

AWS NVMe 드라이버 버전 내역

다음 표에서는 AWS NVMe 드라이버의 릴리스 버전에 대해 설명합니다.

드라이버 버전	세부 정보	릴리스 날짜
1.3.2	IO를 처리하는 EBS 볼륨 수정과 관련된 문제가 수정되어 데이터가 손상될 수 있습니다. 온라인 EBS 볼륨을 수정(예: 크기 조정 또는 유형 변경)하지 않는 고객은 영향을 받지 않습니다.	2019년 9월 10일
1.3.1	안정성 개선	2019년 5월 21일
1.3.0	디바이스 최적화 개선 사항	2018년 8월 31일
1.2.0	베어 메탈 인스턴스를 포함하여 지원되는 모든 인스턴스의 AWS NVMe 장치에 대한 성능과 안정성을 개선했습니다.	2013년 6월 13일
1.0.0	Windows Server를 실행하는 지원 인스턴스 유형에 대한 AWS NVMe 드라이버	2018년 2월 12일

알림 구독

새로운 EC2 Windows Driver 버전이 릴리스되면 이를 알리도록 Amazon SNS를 설정할 수 있습니다. 알림을 받으려면 다음 절차를 수행합니다.

콘솔에서 EC2 알림을 받으려면

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 필요한 경우 탐색 모음에서 리전을 미국 동부(버지니아 북부)로 변경하십시오. 구독을 신청하는 SNS 알림이 이 지역에 있기 때문에 이 지역을 선택해야 합니다.
3. 탐색 창에서 구독을 선택합니다.
4. 구독 생성을 선택합니다.
5. 구독 생성 대화 상자에서 다음 작업을 수행합니다.
 - a. TopicARN의 경우, 다음 Amazon 리소스 이름(ARN)을 복사합니다.
`arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers`
 - b. 프로토콜에서 Email을 선택합니다.
 - c. 엔드포인트에서 알림을 받을 이메일 주소를 입력합니다.
 - d. Create subscription을 선택합니다.
6. 확인 이메일이 발송됩니다. 이메일을 열고 지침에 따라 구독을 완료합니다.

새 EC2 Windows 드라이버가 릴리스될 때마다 구독자에게 알림이 전송됩니다. 이런 알림을 더 이상 받지 않기를 원하는 경우, 다음 절차를 수행해서 구독을 해제하십시오.

Amazon EC2 Windows 드라이버 알림을 구독 해제하려면

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 탐색 창에서 구독을 선택합니다.
3. 구독 확인란을 선택한 후 작업, 구독 삭제를 선택합니다. 확인 메시지가 나타나면 삭제를 선택합니다.

AWS CLI를 사용하여 EC2 알림을 받으려면

AWS CLI를 사용하여 EC2 알림을 받으려면 다음 명령을 사용합니다.

```
aws sns subscribe --topic-arn arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers --  
protocol email --notification-endpoint YourUserName@YourDomainName.ext
```

Windows PowerShell용 AWS Tools를 사용하여 EC2 알림을 받으려면

Windows PowerShell용 AWS Tools를 사용하여 EC2 알림을 받으려면 다음 명령을 사용합니다.

```
Connect-SNSNotification -TopicArn 'arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers'  
-Protocol email -Region us-east-1 -Endpoint 'YourUserName@YourDomainName.ext'
```

CPU 옵션 최적화

Amazon EC2 인스턴스는 여러 개의 스레드를 하나의 CPU 코어에서 동시에 실행할 수 있도록 하는 멀티스레딩을 지원합니다. 각 스레드는 인스턴스에서 가상 CPU(vCPU)로 표현됩니다. 인스턴스에는 인스턴스 유형에 따라 달라지는 기본 CPU 코어 수가 있습니다. 예를 들어 m5.xlarge 인스턴스 유형에는 기본적으로 2개의 CPU 코어와 코어당 2개의 스레드가 있어 vCPU는 총 4개입니다.—

Note

각 vCPU는 T2 인스턴스 및 AWS Graviton2 프로세서에서 구동되는 인스턴스를 제외하고 CPU 코어의 스레드입니다.

대부분의 경우 워크로드에 적합하도록 메모리와 vCPU 수를 결합한 Amazon EC2 인스턴스가 있습니다. 그러나 특정 워크로드 또는 비즈니스 필요를 위해 인스턴스를 최적화하는 다음 CPU 옵션을 지정할 수 있습니다.

- CPU 코어 수: 인스턴스에 대한 CPU 코어 수를 사용자 지정할 수 있습니다. 이를 통해 메모리 집약 워크로드용 RAM이 충분한 반면 CPU 코어를 적게 사용하는 인스턴스의 소프트웨어 라이선스 비용을 잠재적으로 최적화할 수 있습니다.
- 코어당 스레드: CPU 코어당 단일 스레드를 지정하여 멀티스레딩을 비활성화할 수 있습니다. HPC(고성능 컴퓨팅) 워크로드와 같은 특정 워크로드에 대해 이를 수행할 수 있습니다.

인스턴스 시작 중 이러한 CPU 옵션을 지정할 수 있습니다. CPU 옵션 지정에 따른 추가 요금이나 비용 경감은 없습니다. 기존 CPU 옵션으로 시작한 인스턴스와 동일하게 청구됩니다.

내용

- [CPU 옵션 지정 규칙 \(p. 541\)](#)
- [인스턴스 유형별/CPU당 CPU 코어 및 스레드 \(p. 542\)](#)
- [인스턴스의 CPU 옵션 지정 \(p. 554\)](#)
- [인스턴스의 CPU 옵션 보기 \(p. 555\)](#)

CPU 옵션 지정 규칙

인스턴스의 CPU 옵션을 지정하려면 다음 규칙을 알아야 합니다.

- CPU 옵션은 인스턴스 시작 중에만 지정할 수 있으며 시작 후에는 수정할 수 없습니다.
- 인스턴스를 시작할 경우 요청에서 CPU 코어 수와 코어당 스레드를 모두 지정해야 합니다. 예제 요청은 [인스턴스의 CPU 옵션 지정 \(p. 554\)](#) 단원을 참조하십시오.
- 인스턴스의 vCPU 수는 코어당 스레드를 곱한 CPU 코어 수입니다. 사용자 지정 vCPU 수를 지정하려면 인스턴스 유형에 대해 유효한 CPU 코어 수와 코어당 스레드를 지정해야 합니다. 인스턴스의 기본 vCPU 수

를 초과할 수 없습니다. 자세한 내용은 [인스턴스 유형별/CPU당 CPU 코어 및 스레드 \(p. 542\)](#) 단원을 참조하십시오.

- 멀티스레딩을 비활성화하려면 코어당 하나의 스레드를 지정하십시오.
- 기존 인스턴스의 [인스턴스 유형을 변경 \(p. 194\)](#)하면 CPU 옵션이 자동으로 새 인스턴스 유형의 기본 CPU 옵션으로 변경됩니다.
- 지정한 CPU 옵션은 인스턴스를 중지, 시작 또는 재부팅한 후에도 유지됩니다.

인스턴스 유형별/CPU당 CPU 코어 및 스레드

다음 표는 CPU 옵션 지정을 지원하는 인스턴스 유형을 나열합니다. 각 유형에 대해 표는 기본 및 지원되는 CPU 코어 수와 코어당 스레드를 표시합니다.

액셀러레이티드 컴퓨팅 인스턴스

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
f1.2xlarge	8	4	2	1, 2, 3, 4	1, 2
f1.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
f1.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
g3.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
g3.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
g3.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
g3s.xlarge	4	2	2	1, 2	1, 2
g4dn.xlarge	4	2	2	1, 2	1, 2
g4dn.2xlarge	8	4	2	1, 2, 3, 4	1, 2
g4dn.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
g4dn.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
g4dn.12xlarge	48	24	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
g4dn.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
p2.xlarge	4	2	2	1, 2	1, 2
p2.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
p2.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
p3.2xlarge	8	4	2	1, 2, 3, 4	1, 2
p3.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
p3.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
p3dn.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2

컴퓨팅 최적화 인스턴스

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
c4.large	2	1	2	1	1, 2
c4.xlarge	4	2	2	1, 2	1, 2
c4.2xlarge	8	4	2	1, 2, 3, 4	1, 2
c4.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
c4.8xlarge	36	18	2	2, 4, 6, 8, 10, 12, 14, 16, 18	1, 2
c5.large	2	1	2	1	1, 2
c5.xlarge	4	2	2	2	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
c5.2xlarge	8	4	2	2, 4	1, 2
c5.4xlarge	16	8	2	2, 4, 6, 8	1, 2
c5.9xlarge	36	18	2	2, 4, 6, 8, 10, 12, 14, 16, 18	1, 2
c5.12xlarge	48	24	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
c5.18xlarge	72	36	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36	1, 2
c5.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
c5a.large	2	1	2	1	1, 2
c5a.xlarge	4	2	2	1, 2	1, 2
c5a.2xlarge	8	4	2	1, 2, 3, 4	1, 2
c5a.4xlarge	16	8	2	1, 2, 3, 4, 8	1, 2
c5a.8xlarge	32	16	2	1, 2, 3, 4, 8, 12, 16	1, 2
c5a.12xlarge	48	24	2	1, 2, 3, 4, 8, 12, 16, 20, 24	1, 2
c5a.16xlarge	64	32	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1, 2
c5a.24xlarge	96	48	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48	1, 2
c5ad.large	2	1	2	1	1, 2
c5ad.xlarge	4	2	2	1, 2	1, 2
c5ad.2xlarge	8	4	2	1, 2, 3, 4	1, 2
c5ad.4xlarge	16	8	2	1, 2, 3, 4, 8	1, 2
c5ad.8xlarge	32	16	2	1, 2, 3, 4, 8, 12, 16	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
c5ad.12xlarge	48	24	2	1, 2, 3, 4, 8, 12, 16, 20, 24	1, 2
c5ad.16xlarge	64	32	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32	1, 2
c5ad.24xlarge	96	48	2	1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48	1, 2
c5d.large	2	1	2	1	1, 2
c5d.xlarge	4	2	2	2	1, 2
c5d.2xlarge	8	4	2	2, 4	1, 2
c5d.4xlarge	16	8	2	2, 4, 6, 8	1, 2
c5d.9xlarge	36	18	2	2, 4, 6, 8, 10, 12, 14, 16, 18	1, 2
c5d.12xlarge	48	24	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
c5d.18xlarge	72	36	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36	1, 2
c5d.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
c5n.large	2	1	2	1	1, 2
c5n.xlarge	4	2	2	2	1, 2
c5n.2xlarge	8	4	2	2, 4	1, 2
c5n.4xlarge	16	8	2	2, 4, 6, 8	1, 2
c5n.9xlarge	36	18	2	2, 4, 6, 8, 10, 12, 14, 16, 18	1, 2
c5n.18xlarge	72	36	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36	1, 2

범용 인스턴스

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
m5.large	2	1	2	1	1, 2
m5.xlarge	4	2	2	2	1, 2
m5.2xlarge	8	4	2	2, 4	1, 2
m5.4xlarge	16	8	2	2, 4, 6, 8	1, 2
m5.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
m5.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
m5.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
m5.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
m5a.large	2	1	2	1	1, 2
m5a.xlarge	4	2	2	2	1, 2
m5a.2xlarge	8	4	2	2, 4	1, 2
m5a.4xlarge	16	8	2	2, 4, 6, 8	1, 2
m5a.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
m5a.12xlarge	48	24	2	6, 12, 18, 24	1, 2
m5a.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
m5a.24xlarge	96	48	2	12, 18, 24, 36, 48	1, 2
m5ad.large	2	1	2	1	1, 2
m5ad.xlarge	4	2	2	2	1, 2
m5ad.2xlarge	8	4	2	2, 4	1, 2
m5ad.4xlarge	16	8	2	2, 4, 6, 8	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
m5ad.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
m5ad.12xlarge	48	24	2	6, 12, 18, 24	1, 2
m5ad.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
m5ad.24xlarge	96	48	2	12, 18, 24, 36, 48	1, 2
m5d.large	2	1	2	1	1, 2
m5d.xlarge	4	2	2	2	1, 2
m5d.2xlarge	8	4	2	2, 4	1, 2
m5d.4xlarge	16	8	2	2, 4, 6, 8	1, 2
m5d.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
m5d.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
m5d.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
m5d.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
m5dn.large	2	1	2	1	1, 2
m5dn.xlarge	4	2	2	2	1, 2
m5dn.2xlarge	8	4	2	2, 4	1, 2
m5dn.4xlarge	16	8	2	2, 4, 6, 8	1, 2
m5dn.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
m5dn.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
m5dn.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
m5dn.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
m5n.large	2	1	2	1	1, 2
m5n.xlarge	4	2	2	2	1, 2
m5n.2xlarge	8	4	2	2, 4	1, 2
m5n.4xlarge	16	8	2	2, 4, 6, 8	1, 2
m5n.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
m5n.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
m5n.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
m5n.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
t3.nano	2	1	2	1	1, 2
t3.micro	2	1	2	1	1, 2
t3.small	2	1	2	1	1, 2
t3.medium	2	1	2	1	1, 2
t3.large	2	1	2	1	1, 2
t3.xlarge	4	2	2	2	1, 2
t3.2xlarge	8	4	2	2, 4	1, 2
t3a.nano	2	1	2	1	1, 2
t3a.micro	2	1	2	1	1, 2
t3a.small	2	1	2	1	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
t3a.medium	2	1	2	1	1, 2
t3a.large	2	1	2	1	1, 2
t3a.xlarge	4	2	2	2	1, 2
t3a.2xlarge	8	4	2	2, 4	1, 2

메모리 최적화 인스턴스

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
r4.large	2	1	2	1	1, 2
r4.xlarge	4	2	2	1, 2	1, 2
r4.2xlarge	8	4	2	1, 2, 3, 4	1, 2
r4.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
r4.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
r4.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
r5.large	2	1	2	1	1, 2
r5.xlarge	4	2	2	2	1, 2
r5.2xlarge	8	4	2	2, 4	1, 2
r5.4xlarge	16	8	2	2, 4, 6, 8	1, 2
r5.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
r5.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
r5.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
r5.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36,	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
				38, 40, 42, 44, 46, 48	
r5a.large	2	1	2	1	1, 2
r5a.xlarge	4	2	2	2	1, 2
r5a.2xlarge	8	4	2	2, 4	1, 2
r5a.4xlarge	16	8	2	2, 4, 6, 8	1, 2
r5a.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
r5a.12xlarge	48	24	2	6, 12, 18, 24	1, 2
r5a.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
r5a.24xlarge	96	48	2	12, 18, 24, 36, 48	1, 2
r5ad.large	2	1	2	1	1, 2
r5ad.xlarge	4	2	2	2	1, 2
r5ad.2xlarge	8	4	2	2, 4	1, 2
r5ad.4xlarge	16	8	2	2, 4, 6, 8	1, 2
r5ad.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
r5ad.12xlarge	48	24	2	6, 12, 18, 24	1, 2
r5ad.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
r5ad.24xlarge	96	48	2	12, 18, 24, 36, 48	1, 2
r5d.large	2	1	2	1	1, 2
r5d.xlarge	4	2	2	2	1, 2
r5d.2xlarge	8	4	2	2, 4	1, 2
r5d.4xlarge	16	8	2	2, 4, 6, 8	1, 2
r5d.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
r5d.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
r5d.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
r5d.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
r5dn.large	2	1	2	1	1, 2
r5dn.xlarge	4	2	2	2	1, 2
r5dn.2xlarge	8	4	2	2, 4	1, 2
r5dn.4xlarge	16	8	2	2, 4, 6, 8	1, 2
r5dn.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
r5dn.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
r5dn.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
r5dn.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
r5n.large	2	1	2	1	1, 2
r5n.xlarge	4	2	2	2	1, 2
r5n.2xlarge	8	4	2	2, 4	1, 2
r5n.4xlarge	16	8	2	2, 4, 6, 8	1, 2
r5n.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
r5n.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
r5n.16xlarge	64	32	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
r5n.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
x1.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
x1.32xlarge	128	64	2	4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64	1, 2
x1e.xlarge	4	2	2	1, 2	1, 2
x1e.2xlarge	8	4	2	1, 2, 3, 4	1, 2
x1e.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
x1e.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
x1e.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
x1e.32xlarge	128	64	2	4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64	1, 2
z1d.large	2	1	2	1	1, 2
z1d.xlarge	4	2	2	2	1, 2
z1d.2xlarge	8	4	2	2, 4	1, 2
z1d.3xlarge	12	6	2	2, 4, 6	1, 2
z1d.6xlarge	24	12	2	2, 4, 6, 8, 10, 12	1, 2

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
CPU 옵션 최적화

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
z1d.12xlarge	48	24	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2

스토리지 최적화 인스턴스

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
d2.xlarge	4	2	2	1, 2	1, 2
d2.2xlarge	8	4	2	1, 2, 3, 4	1, 2
d2.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
d2.8xlarge	36	18	2	2, 4, 6, 8, 10, 12, 14, 16, 18	1, 2
h1.2xlarge	8	4	2	1, 2, 3, 4	1, 2
h1.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
h1.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
h1.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
i3.large	2	1	2	1	1, 2
i3.xlarge	4	2	2	1, 2	1, 2
i3.2xlarge	8	4	2	1, 2, 3, 4	1, 2
i3.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
i3.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
i3.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
i3en.large	2	1	2	1	1, 2
i3en.xlarge	4	2	2	2	1, 2

인스턴스 유형	기본 vCPU	기본 CPU 코어	코어당 기본 스레드	유효한 CPU 코어 수	코어당 유효한 스레드 수
i3en.2xlarge	8	4	2	2, 4	1, 2
i3en.3xlarge	12	6	2	2, 4, 6	1, 2
i3en.6xlarge	24	12	2	2, 4, 6, 8, 10, 12	1, 2
i3en.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
i3en.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2

인스턴스의 CPU 옵션 지정

인스턴스 시작 중 CPU 옵션을 지정할 수 있습니다. 다음 예는 r4.4xlarge 인스턴스 유형을 위한 것이며 다음과 같은 [기본값](#) (p. 549)을 포함합니다.

- 기본 CPU 코어: 8
- 코어당 기본 스레드: 2
- 기본 vCPU: 16(8 * 2)
- 유효한 CPU 코어 수: 1, 2, 3, 4, 5, 6, 7, 8
- 코어당 유효한 스레드 수: 1, 2

멀티스레딩 비활성화

멀티스레딩을 비활성화하려면 코어당 하나의 스레드를 지정하십시오.

인스턴스 시작 중 멀티스레딩을 비활성화하려면(콘솔)

1. [인스턴스 시작 마법사를 사용하여 인스턴스 시작](#) (p. 376)의 절차를 따르십시오.
2. Configure Instance Details(인스턴스 정보 구성) 페이지에서 CPU options(CPU 옵션)에 대해 CPU 옵션 지정을 선택합니다.
3. 코어 수에 대해 필요한 CPU 코어 수를 선택합니다. 이 예에서 r4.4xlarge 인스턴스에 필요한 기본 CPU 코어 개수를 지정하려면 8을 선택합니다.
4. 멀티스레딩을 비활성화하려면 코어당 스레드로 1을 선택하십시오.
5. 마법사에 표시되는 지침에 따라 계속합니다. 인스턴스 시작 검토 페이지에서 옵션 검토를 마쳤으면 시작을 선택합니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작](#) (p. 376)를 참조하십시오.

인스턴스 시작 중 멀티스레딩을 비활성화하려면(AWS CLI)

`run-instances` AWS CLI 명령을 사용하여 `--cpu-options` 파라미터의 `ThreadsPerCore`에 값을 1로 지정합니다. `CoreCount`에 대해 CPU 코어 수를 지정합니다. 이 예에서 r4.4xlarge 인스턴스에 필요한 기본 CPU 코어 개수를 지정하려면 값을 8로 지정합니다.

```
aws ec2 run-instances --image-id ami-1a2b3c4d --instance-type r4.4xlarge --cpu-options "CoreCount=8,ThreadsPerCore=1" --key-name MyKeyPair
```

vCPU 수 사용자 지정

인스턴스의 코어당 CPU 코어와 스레드 수를 사용자 지정할 수 있습니다.

인스턴스 시작 중 사용자 지정 vCPU 수를 지정하려면(콘솔)

다음 예는 6개의 vCPU로 r4.4xlarge 인스턴스를 시작합니다.

1. [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#)의 절차를 따르십시오.
2. Configure Instance Details(인스턴스 정보 구성) 페이지에서 CPU options(CPU 옵션)에 대해 CPU 옵션 지정을 선택합니다.
3. 6개의 vCPU를 얻기 위해 다음과 같이 3개의 CPU 코어와 코어당 2개의 스레드를 지정합니다.
 - 코어 수로 3을 선택합니다.
 - 코어당 스레드로 2를 선택합니다.
4. 마법사에 표시되는 지침에 따라 계속합니다. 인스턴스 시작 검토 페이지에서 옵션 검토를 마쳤으면 시작을 선택합니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오.

인스턴스 시작 중 vCPU 수를 사용자 지정하려면(AWS CLI)

다음 예는 6개의 vCPU로 r4.4xlarge 인스턴스를 시작합니다.

`run-instances` AWS CLI 명령을 사용하여 `--cpu-options` 파라미터에서 CPU 코어 수와 스레드 수를 지정합니다. 6개의 vCPU를 얻기 위해 3개의 CPU 코어와 코어당 2개의 스레드를 지정할 수 있습니다.

```
aws ec2 run-instances --image-id ami-1a2b3c4d --instance-type r4.4xlarge --cpu-options "CoreCount=3,ThreadsPerCore=2" --key-name MyKeyPair
```

또는 6개의 vCPU를 얻기 위해 6개의 CPU 코어와 코어당 하나의 스레드(멀티스레딩 비활성화)를 지정합니다.

```
aws ec2 run-instances --image-id ami-1a2b3c4d --instance-type r4.4xlarge --cpu-options "CoreCount=6,ThreadsPerCore=1" --key-name MyKeyPair
```

인스턴스의 CPU 옵션 보기

Amazon EC2 콘솔에서 또는 AWS CLI를 사용해 인스턴스를 설명하여 기존 인스턴스의 CPU 옵션을 볼 수 있습니다.

인스턴스의 CPU 옵션을 보려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
3. 설명을 선택하고 vCPU 수 필드를 확인합니다.
4. 코어 수와 코어당 스레드 수를 확인하려면 vCPU 수 필드 값을 선택합니다.

인스턴스의 CPU 옵션을 보려면(AWS CLI)

아래와 같이 `describe-instances` 명령을 사용합니다.

```
aws ec2 describe-instances --instance-ids i-123456789abcde123
```

```
...
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": "ec2-198-51-100-5.eu-central-1.compute.amazonaws.com",
            "State": {
                "Code": 16,
                "Name": "running"
            },
            "EbsOptimized": false,
            "LaunchTime": "2018-05-08T13:40:33.000Z",
            "PublicIpAddress": "198.51.100.5",
            "PrivateIpAddress": "172.31.2.206",
            "ProductCodes": [],
            "VpcId": "vpc-1a2b3c4d",
            "CpuOptions": {
                "CoreCount": 34,
                "ThreadsPerCore": 1
            },
            "StateTransitionReason": "",
            ...
        }
    ]
...
}
```

반환된 출력에서 CoreCount 필드는 인스턴스의 코어 수를 나타냅니다. ThreadsPerCore 필드는 코어당 스레드 수를 나타냅니다.

또는 인스턴스에 연결하고 작업 관리자를 사용하여 인스턴스에 대한 CPU 정보를 봅니다.

AWS Config를 사용하여 종료된 인스턴스를 포함한 인스턴스의 구성 변경을 기록, 액세스, 감사 및 평가할 수 있습니다. 자세한 내용은 AWS Config Developer Guide의 [AWS Config 시작하기](#)를 참조하십시오.

Windows 인스턴스에 대한 시간 설정

많은 서버 작업과 프로세스에서 일관되고 정확한 시간 참조가 중요합니다. 대부분의 시스템 로그에는 문제가 발생한 시간과 이벤트가 발생한 순서를 파악하는 데 사용할 수 있는 타임스탬프가 포함되어 있습니다. AWS CLI 또는 AWS SDK를 사용하여 인스턴스에서 요청하는 경우 이러한 도구가 사용자를 대신하여 요청에 서명합니다. 인스턴스의 날짜와 시간이 잘못 설정되어 있으면 서명 날짜가 요청 날짜와 일치하지 않아 AWS가 해당 요청을 거부할 수 있습니다. Windows 인스턴스의 경우 UTC(협정 세계시)를 사용하는 것이 좋습니다. 원하는 경우 다른 표준 시간대를 사용할 수도 있습니다.

목차

- [표준 시간대 변경 \(p. 556\)](#)
- [NTP\(Network Time Protocol\) 구성 \(p. 557\)](#)
- [Amazon Windows AMI에 대한 기본 NTP\(Network Time Protocol\) 설정 \(p. 558\)](#)
- [Windows Server 2008 이상에 대한 시간 설정 구성 \(p. 559\)](#)
- [관련 리소스 \(p. 560\)](#)

표준 시간대 변경

Windows 인스턴스는 기본적으로 UTC 표준 시간대로 설정됩니다. 현지 표준 시간대 또는 네트워크의 다른 부분에 대한 표준 시간대와 일치하도록 시간을 변경할 수 있습니다.

인스턴스에 대한 표준 시간대를 변경하려면

1. 인스턴스에서 명령 프롬프트 창을 엽니다.
2. 인스턴스에서 사용할 표준 시간대를 식별합니다. 표준 시간대 목록을 가져오려면 tzutil /l 명령을 사용합니다. 이 명령은 사용 가능한 모든 표준 시간대의 목록을 다음 형식으로 반환합니다.

```
display name  
time zone ID
```

3. 인스턴스에 배정할 표준 시간대 ID를 찾습니다.
4. 다음 명령을 사용하여 인스턴스에 표준 시간대를 배정합니다.

```
tzutil /s "Pacific Standard Time"
```

새 표준 시간대는 즉시 적용됩니다.

NTP(Network Time Protocol) 구성

Amazon은 Amazon Time Sync Service를 제공합니다. 이 서비스는 모든 EC2 인스턴스에서 액세스할 수 있고 다른 AWS 서비스에 의해서도 사용됩니다. Amazon Time Sync Service를 사용하여 인스턴스를 구성하는 것이 좋습니다. 이 서비스는 각 AWS 리전의 위성 연결 및 원자 기준 시계 집합을 사용하여 협정 세계시(UTC) 세계 표준의 정확한 현재 시간 판독을 제공합니다. Amazon Time Sync Service는 UTC에 추가된 윤초를 자동으로 평활화합니다. 이 서비스는 VPC에서 실행 중인 모든 인스턴스의 169.254.169.123 IP 주소에서 사용할 수 있으며, 인스턴스는 인터넷 액세스 없이도 이 서비스를 사용할 수 있습니다. Windows AMI는 2018년 8월 릴리스부터 기본적으로 Amazon Time Sync Service를 사용합니다.

NTP 구성 확인하려면

1. 인스턴스에서 명령 프롬프트 창을 엽니다.
2. 다음 명령을 입력하여 현재 NTP 구성 설정을 가져옵니다.

```
w32tm /query /configuration
```

이 명령은 Windows 인스턴스에 대한 현재 구성 설정을 반환합니다.

3. (선택 사항) 다음 명령을 입력하여 현재 구성의 상태를 가져옵니다.

```
w32tm /query /status
```

이 명령은 인스턴스를 NTP 서버와 마지막으로 동기화한 시간, 폴링 간격 등의 정보를 반환합니다.

Amazon Time Sync Service를 사용하도록 NTP 서버를 변경하려면

1. 명령 프롬프트 창에서 다음 명령을 실행합니다.

```
w32tm /config /manualpeerlist:169.254.169.123 /syncfromflags:manual /update
```

2. 다음 명령을 사용하여 새 설정을 확인합니다.

```
w32tm /query /configuration
```

반환된 출력에서 NtpServer에 169.254.169.123 IP 주소가 표시되는지 확인합니다.

필요한 경우 다른 NTP 서버 집합을 사용하도록 인스턴스를 변경할 수 있습니다. 예를 들어, 인터넷에 액세스 할 수 없는 Windows 인스턴스가 있는 경우 프라이빗 네트워크에 있는 NTP 서버를 사용하도록 인스턴스를 구성할 수 있습니다. 인스턴스가 도메인 내에 있는 경우 도메인 컨트롤러를 시간 소스로 사용하여 시간 불일치를 방지하도록 설정을 변경해야 합니다. 포트 123(NTP)의 아웃바운드 UDP 트래픽을 허용하도록 인스턴스의 보안 그룹을 구성해야 합니다.

NTP 서버를 변경하려면

1. 명령 프롬프트 창에서 다음 명령을 실행합니다.

```
w32tm /config /manualpeerlist:comma-delimited list of NTP servers /  
syncfromflags:manual /update
```

여기에서 *comma-delimited list of NTP servers*는 인스턴스에서 사용할 NTP 서버의 목록입니다.

2. 다음 명령을 사용하여 새 설정을 확인합니다.

```
w32tm /query /configuration
```

Amazon Windows AMI에 대한 기본 NTP(Network Time Protocol) 설정

일반적으로 Amazon 머신 이미지(AMI)는 EC2 인프라에서 함수 변경이 필요한 경우를 제외하고는 내장된 기본 설정을 그대로 유지합니다. 다음 설정은 가상 환경에서 잘 작동할 뿐 아니라 클럭 드리프트를 1초 이내의 정확도로 유지하는 것으로 판명되었습니다.

- 업데이트 간격 – 시간 서비스가 시스템 시간을 정확하게 조정하는 빈도를 관리합니다. AWS는 2분마다 업데이트가 이루어지도록 업데이트 간격을 구성합니다.
- NTP 서버 – AMI는 2018년 8월 릴리스부터 기본적으로 Amazon Time Sync Service를 사용합니다. 이 시 간 서비스는 169.254.169.123 엔드포인트에 있는 모든 EC2 리전에서 액세스할 수 있습니다. 뿐만 아니라 0x9 플래그는 시간 서비스가 클라이언트 역할을 하면서 SpecialPollInterval을 사용해 구성된 시간 서버에 체크인하는 빈도를 파악함을 나타냅니다.
- 유형 – “NTP”은 서비스가 도메인 일부의 역할을 수행하는 대신 독립 실행형 NTP 클라이언트의 역할을 수행할 것임을 뜻합니다.
- 활성화 및 InputProvider – 시간 서비스가 활성화되어 운영 체제에 시간을 제공합니다.
- 특수 폴링 간격 – 구성된 NTP 서버를 900초 또는 15분마다 점검합니다.

레지스트리 경로	키 이름	테스트
HKLM:\System\CurrentControlSet\services\w32time\Config	UpdateInterval	120
HKLM:\System\CurrentControlSet\services\w32time\Parameters	NtpServer	169.254.169.123,0x9
HKLM:\System\CurrentControlSet\services\w32time\Parameters	유형	NTP
HKLM:\System\CurrentControlSet\services	활성	1

레지스트리 경로	키 이름	테스트
\w32time\TimeProviders \NtpClient		
HKLM:\System \CurrentControlSet\services \w32time\TimeProviders \NtpClient	InputProvider	1
HKLM:\System \CurrentControlSet\services \w32time\TimeProviders \NtpClient	SpecialPollInterval	900

Windows Server 2008 이상에 대한 시간 설정 구성

Windows 인스턴스의 시간을 변경할 경우 시스템을 다시 시작해도 시간이 계속 유지되는지 확인해야 합니다. 그렇지 않으면 인스턴스가 다시 시작될 때 UTC 시간으로 다시 돌아갑니다. Windows Server 2008 이상에서는 RealTimelsUniversal 레지스트리 키를 추가하여 시간 설정을 계속 유지할 수 있습니다. 이 키는 모든 현재 세대 인스턴스에서 기본적으로 설정됩니다. RealTimelsUniversal 레지스트리 키가 설정되어 있는지 확인하려면 다음 절차의 4단계를 참조하세요. 키가 설정되지 않은 경우 처음부터 다음 단계를 수행하세요.

RealTimelsUniversal 레지스트리 키를 설정하려면

- 인스턴스에서 명령 프롬프트 창을 엽니다.
- 다음 명령을 사용하여 레지스트리 키를 추가합니다.

```
reg add "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation" /v
RealTimelsUniversal /d 1 /t REG_DWORD /f
```

- 2013년 2월 22일 이전에 생성된 Windows Server 2008 AMI(Windows Server 2008 R2가 아님)를 사용하고 있는 경우에는 최신 AWS Windows AMI로 업데이트하는 것이 좋습니다. Windows Server 2008 R2를 실행하는 AMI(Windows Server 2008 아님)를 사용하는 경우, Microsoft 핫픽스[KB2922223](#)가 설치되어 있는지 확인해야 합니다. 이 핫픽스가 설치되지 않은 경우 최신 AWS Windows AMI로 업데이트하는 것이 좋습니다.
- (선택 사항) 다음 명령을 사용하여 인스턴스에서 키를 성공적으로 저장했는지 확인합니다.

```
reg query "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation" /s
```

이 명령은 TimeZoneInformation 레지스트리 키에 대한 하위 키를 반환합니다. 목록의 맨 아래에 RealTimelsUniversal 키가 다음과 비슷하게 표시되어야 합니다.

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation		
Bias	REG_DWORD	0x1e0
DaylightBias	REG_DWORD	0xffffffffc4
DaylightName	REG_SZ	@tzres.dll,-211
DaylightStart	REG_BINARY	00000300020002000000000000000000
StandardBias	REG_DWORD	0x0
StandardName	REG_SZ	@tzres.dll,-212
StandardStart	REG_BINARY	00000B00010002000000000000000000
TimeZoneKeyName	REG_SZ	Pacific Standard Time
DynamicDaylightTimeDisabled	REG_DWORD	0x0
ActiveTimeBias	REG_DWORD	0x1a4
RealTimeIsUniversal	REG_DWORD	0x1

관련 리소스

윤초(leap second)의 추가를 포함해 Windows 운영 체제의 시간 조정 및 관리 방법에 대한 자세한 내용은 다음 설명서를 참조하십시오.

- [Windows 시간 서비스의 원리](#)(Microsoft)
- [W32tm](#)(Microsoft)
- [Windows 시간 서비스의 윤초 처리 방법](#)(Microsoft)
- [윤초와 Windows에 대한 이야기: Y2K와의 차이](#)(Microsoft)

Windows 인스턴스 암호 설정

Windows 인스턴스에 연결할 경우 인스턴스에 대한 액세스 권한이 있는 사용자 계정과 암호를 지정해야 합니다. 인스턴스에 처음 연결할 때 관리자 계정과 기본 암호를 지정하라는 메시지가 나옵니다.

Windows Server 2012 R2 이하용 AWS Window AMI를 사용하는 경우 [EC2Config 서비스 \(p. 496\)](#)에서 기본 암호가 생성됩니다. Windows Server 2016 이상용 AWS Windows AMI를 사용한다면 [EC2Launch \(p. 489\)](#)에서 기본 암호를 생성합니다.

Note

Windows Server 2016 이상을 사용하는 경우 로컬 관리자에 대해 `Password never expires`가 비활성화됩니다. Windows Server 2012 R2 이하를 사용하는 경우 로컬 관리자에 대해 `Password never expires`가 활성화됩니다.

연결 후 관리자 암호 변경

인스턴스에 처음 연결했을 때 관리자 암호를 기본값과 다르게 변경하는 것이 좋습니다. 다음 절차에 따라 Windows 인스턴스에 대한 관리자 암호를 변경합니다.

Important

새 암호를 안전한 위치에 저장합니다. Amazon EC2 콘솔을 사용하여 새 암호를 검색할 수 없습니다. 콘솔에서는 기본 암호만 검색할 수 있습니다. 암호를 변경한 이후에 기본 암호를 사용하여 인스턴스에 연결할 경우 "Your credentials did not work."라는 오류가 발생합니다.

로컬 관리자 암호를 변경하려면

1. 인스턴스에 연결하고 명령 프롬프트를 엽니다.
2. 다음 명령을 실행합니다. 새 암호에 특수 문자가 포함된 경우 암호를 겹따옴표로 둑습니다.

```
net user Administrator "new_password"
```

3. 새 암호를 안전한 위치에 저장합니다.

분실 또는 만료된 암호 변경

암호가 기억나지 않거나 만료된 경우 새 암호를 생성할 수 있습니다. 암호 재설정 절차는 [기억나지 않거나 만료된 Windows 관리자 암호 재설정 \(p. 1196\)](#)을 참조하십시오.

설치 미디어를 사용하여 Windows 구성 요소 추가

Windows Server 운영 체제에는 여러 선택 구성 요소가 포함됩니다. 각 Amazon EC2 Windows Server AMI에 모든 선택 구성 요소를 포함시키는 것은 유용하지 않습니다. 대신 설치 미디어 EBS 스냅샷을 제공하는데 여기에는 Windows 인스턴스에 구성 요소를 구성 또는 설치하는 데 필요한 파일이 있습니다.

선택 구성 요소에 액세스하고 설치하려면 현재 Windows Server 버전의 올바른 EBS 스냅샷을 찾고 스냅샷에서 볼륨을 생성하고 인스턴스에 볼륨을 연결해야 합니다.

시작하기 전에

AWS Management 콘솔 또는 명령줄 도구를 사용하여 인스턴스의 인스턴스 ID 및 가용 영역을 가져올 수 있습니다. 인스턴스와 동일한 가용 영역에서 EBS 볼륨을 생성해야 합니다.

콘솔을 사용하여 Windows 구성 요소 추가

다음 절차에 따라 AWS Management 콘솔을 사용하여 인스턴스에 Windows 구성 요소를 추가할 수 있습니다.

콘솔을 사용하여 인스턴스에 Windows 구성 요소를 추가하려면

1. <https://console.aws.amazon.com/ec2>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스냅샷을 선택합니다.
3. 필터 막대에서 퍼블릭 스냅샷을 선택합니다.
4. 소유자 필터를 추가하고 Amazon 이미지를 선택합니다.
5. 설명 필터를 추가하고 **Windows**를 입력합니다.
6. Enter를 누릅니다
7. 시스템 아키텍처 및 언어 기본 설정과 일치하는 스냅샷을 선택합니다. 예를 들어 현재 인스턴스가 Windows Server 2016을 실행한다면 Windows 2019 English Installation Media를 선택합니다.
8. 작업, 볼륨 생성을 선택합니다.
9. 볼륨 생성 대화 상자에서 Windows 인스턴스와 일치하는 가용 영역을 선택하고 생성을 선택합니다.
10. 성공적으로 볼륨 생성 완료 메시지에서 방금 만든 볼륨을 선택합니다.
11. 작업, 볼륨 연결을 선택합니다.
12. 볼륨 연결 대화 상자에서 연결할 인스턴스 ID와 디바이스 이름을 입력하고 연결을 선택합니다. 디바이스 이름과 관련하여 도움이 필요하면 [디바이스 명명](#)을 참조하십시오.
13. 인스턴스에 연결하고 볼륨을 사용 가능하도록 만듭니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오.

Important

볼륨을 초기화하지 마십시오.

14. 제어판을 열고 프로그램 및 기능을 엽니다. Windows 기능 사용/사용 안 함을 선택합니다. 설치할 매체를 선택하라는 메시지가 뜨면 EBS 볼륨을 지정합니다.

Windows PowerShell용 도구를 사용하여 Windows 구성 요소 추가

다음 절차에 따라 Windows PowerShell용 도구를 사용하여 인스턴스에 Windows 구성 요소를 추가할 수 있습니다.

Windows PowerShell용 도구를 사용하여 인스턴스에 Windows 구성 요소를 추가하려면

1. `Get-EC2Snapshot` cmdlet와 `Owner` 및 `description` 필터를 사용하여 사용 가능한 설치 미디어 스냅샷의 목록을 가져옵니다.

```
PS C:\> Get-EC2Snapshot -Owner amazon -Filter @{ Name="description"; Values="Windows*" }
```

2. 출력에서 시스템 아키텍처 및 언어 기본 설정과 일치하는 스냅샷의 ID를 적어둡니다. 다음 예를 참조하십시오.

```
...  
DataEncryptionKeyId :  
Description : Windows 2019 English Installation Media  
Encrypted : False  
KmsKeyId :  
OwnerAlias : amazon  
OwnerId : 123456789012  
Progress : 100%  
SnapshotId : snap-22da283e  
StartTime : 10/25/2019 8:00:47 PM  
State : completed  
StateMessage :  
Tags : {}  
VolumeId : vol-be5eafcb  
VolumeSize : 6  
...
```

3. [New-EC2Volume cmdlet](#)을 사용하여 스냅샷에서 볼륨을 만듭니다. 인스턴스와 동일한 가용 영역을 지정합니다.

```
PS C:\> New-EC2Volume -AvailabilityZone us-east-1a -VolumeType gp2 -  
SnapshotId snap-22da283e
```

4. 볼륨 ID를 출력에 기록해 둡니다.

```
Attachments : {}  
AvailabilityZone : us-east-1a  
CreateTime : 4/18/2017 10:50:25 AM  
Encrypted : False  
Iops : 100  
KmsKeyId :  
Size : 6  
SnapshotId : snap-22da283e  
State : creating  
Tags : {}  
VolumeId : vol-06aa9e1fbf8b82ed1  
VolumeType : gp2
```

5. [Add-EC2Volume cmdlet](#)을 사용하여 인스턴스에 볼륨을 연결합니다.

```
PS C:\> Add-EC2Volume -InstanceId i-087711ddaf98f9489 -VolumeId vol-06aa9e1fbf8b82ed1 -  
Device xvdh
```

6. 인스턴스에 연결하고 볼륨을 사용 가능하도록 만듭니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오.

Important

볼륨을 초기화하지 마십시오.

7. 제어판을 열고 프로그램 및 기능을 엽니다. Windows 기능 사용/사용 안 함을 선택합니다. 설치할 매체를 선택하라는 메시지가 뜨면 EBS 볼륨을 지정합니다.

AWS CLI를 사용하여 Windows 구성 요소 추가

다음 절차에 따라 AWS CLI를 사용하여 인스턴스에 Windows 구성 요소를 추가할 수 있습니다.

AWS CLI를 사용하여 인스턴스에 Windows 구성 요소를 추가하려면

1. [describe-snapshots](#) 명령을 사용하고 owner-ids 파라미터 및 description 필터를 적용하여 사용 가능한 설치 미디어 스냅샷의 목록을 가져옵니다.

```
aws ec2 describe-snapshots --owner-ids amazon --filters
    Name=description,Values=Windows*
```

2. 출력에서 시스템 아키텍처 및 언어 기본 설정과 일치하는 스냅샷의 ID를 적어둡니다. 다음 예를 참조하십시오.

```
{
    "Snapshots": [
        ...
        {
            "OwnerAlias": "amazon",
            "Description": "Windows 2019 English Installation Media",
            "Encrypted": false,
            "VolumeId": "vol-be5eafcb",
            "State": "completed",
            "VolumeSize": 6,
            "Progress": "100%",
            "StartTime": "2019-10-25T20:00:47.000Z",
            "SnapshotId": "snap-22da283e",
            "OwnerId": "123456789012"
        },
        ...
    ]
}
```

3. [create-volume](#) 명령을 사용하여 스냅샷에서 볼륨을 만듭니다. 인스턴스와 동일한 가용 영역을 지정합니다.

```
aws ec2 create-volume --snapshot-id snap-22da283e --volume-type gp2 --availability-
zone us-east-1a
```

4. 볼륨 ID를 출력에 기록해둡니다.

```
{
    "AvailabilityZone": "us-east-1a",
    "Encrypted": false,
    "VolumeType": "gp2",
    "VolumeId": "vol-0c98b37f30bcbe290",
    "State": "creating",
    "Iops": 100,
    "SnapshotId": "snap-22da283e",
    "CreateTime": "2017-04-18T10:33:10.940Z",
    "Size": 6
}
```

5. [attach-volume](#) 명령을 사용하여 이 볼륨을 인스턴스에 연결합니다.

```
aws ec2 attach-volume --volume-id vol-0c98b37f30bcbe290 --instance-
id i-01474ef662b89480 --device xvdf
```

6. 인스턴스에 연결하고 볼륨을 사용 가능하도록 만듭니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오.

Important

볼륨을 초기화하지 마십시오.

- 제어판을 열고 프로그램 및 기능을 엽니다. Windows 기능 사용/사용 안 함을 선택합니다. 설치할 매체를 선택하라는 메시지가 뜨면 EBS 볼륨을 지정합니다.

Windows 인스턴스의 보조 프라이빗 IPv4 주소 구성

인스턴스에서 다중 프라이빗 IPv4 주소를 지정할 수 있습니다. 인스턴스에 보조 프라이빗 IPv4 주소를 배정한 후에는 보조 프라이빗 IPv4 주소를 인식하도록 인스턴스의 운영 체제를 구성해야 합니다.

보조 프라이빗 IPv4 주소를 인식하도록 Windows 인스턴스의 운영 체제를 구성하려면 다음을 수행해야 합니다.

- 1단계: 인스턴스에서 고정 IP 주소 지정 구성 (p. 564)
- 2단계: 인스턴스에 대한 보조 프라이빗 IP 주소 구성 (p. 566)
- 3단계: 보조 프라이빗 IP 주소를 사용하도록 애플리케이션 구성 (p. 567)

Note

이러한 지침은 Windows Server 2008 R2에 기초합니다. 이러한 단계의 구현은 Windows 인스턴스의 운영 체제에 따라 다를 수 있습니다.

사전 조건

시작하기 전에 다음 요구 사항을 충족하는지 확인하십시오.

- 모범 사례로서 최신 AMI를 사용하여 Windows 인스턴스를 시작합니다. 구형 Windows AMI를 사용하는 경우 <http://support.microsoft.com/kb/2582281>에 나와 있는 Microsoft 핫픽스를 적용했는지 확인하십시오.
- VPC에서 인스턴스를 시작한 후 부 프라이빗 IP 주소를 추가합니다. 자세한 내용은 [보조 프라이빗 IPv4 주소 할당 \(p. 710\)](#) 단원을 참조하십시오.
- 이러한 단계의 작업을 완료한 후 해당 웹 사이트에 대한 인터넷 요청을 허용하려면 탄력적 IP 주소를 구성하고 이 주소를 부 프라이빗 IP 주소와 연결해야 합니다. 자세한 내용은 [탄력적 IP 주소와 보조 프라이빗 IPv4 주소 연결 \(p. 712\)](#) 단원을 참조하십시오.

1단계: 인스턴스에서 고정 IP 주소 지정 구성

Windows 인스턴스에서 여러 IP 주소를 사용할 수 있도록 하려면 DHCP 보다는 고정 IP 주소 지정 방식을 사용하도록 인스턴스를 구성해야 합니다.

Important

인스턴스에서 고정 IP 주소 지정을 구성하는 경우 IP 주소가 콘솔, CLI 또는 API에 표시된 것과 정확히 일치해야 합니다. 이러한 IP 주소를 잘못 입력할 경우 인스턴스가 연결하지 못할 수 있습니다.

Windows 인스턴스에서 고정 IP 주소 지정을 구성하려면

- 인스턴스에 연결합니다.
- 먼저 다음 단계를 수행하여 인스턴스에 대한 IP 주소, 서브넷 마스크 및 기본 게이트웨이 주소를 찾습니다.
 - 명령 프롬프트 창에서 다음 명령을 실행합니다.

```
ipconfig /all
```

출력에서 다음 섹션을 살펴보고 네트워크 인터페이스에 대한 IPv4 주소, 서브넷 마스크, 기본 게이트웨이 및 DNS 서버 값을 메모해 둡니다.

```
Ethernet adapter Local Area Connection:  
  
Connection-specific DNS Suffix . :  
Description . . . . . :  
Physical Address . . . . . :  
DHCP Enabled. . . . . :  
Autoconfiguration Enabled . . . . :  
IPv4 Address. . . . . : 10.0.0.131  
Subnet Mask . . . . . : 255.255.255.0  
Default Gateway . . . . . : 10.0.0.1  
DNS Servers . . . . . : 10.1.1.10  
10.1.1.20
```

3. 다음 명령을 실행하여 네트워킹 및 공유 센터를 엽니다.

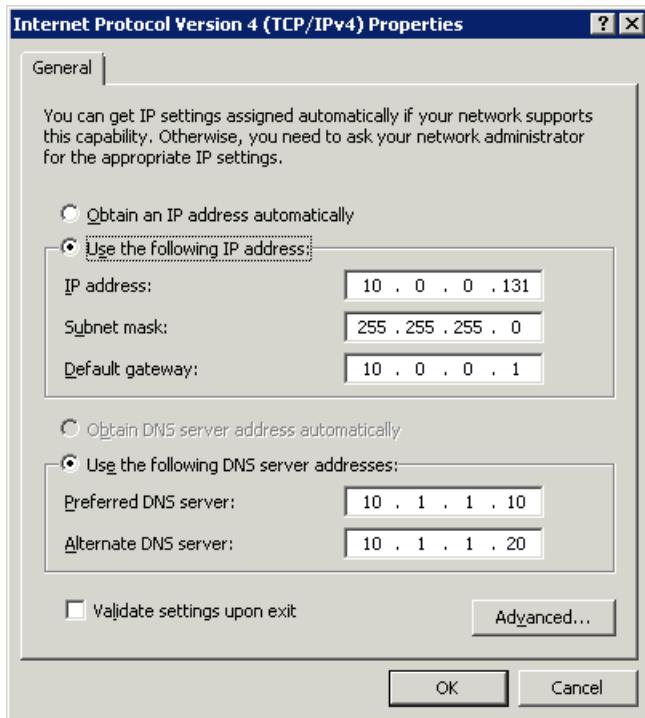
```
%SystemRoot%\system32\control.exe ncpa.cpl
```

4. 네트워크 인터페이스(로컬 영역 연결)에 대한 컨텍스트 메뉴를 열고(마우스 오른쪽 버튼 클릭) 속성을 선택합니다.
5. 인터넷 프로토콜 버전 4(TCP/IPV4)를 선택하고 속성을 클릭합니다.
6. 인터넷 프로토콜 버전 4(TCP/IPV4) 속성 대화 상자에서 다음 IP 주소 사용을 선택하고 다음 값을 입력한 다음 확인을 클릭합니다.

필드	값
IP 주소	위의 2단계에서 얻은 IPv4 주소입니다.
서브넷 마스크	위의 2단계에서 얻은 서브넷 마스크입니다.
기본 게이트웨이	위의 2단계에서 얻은 기본 게이트웨이 주소입니다.
기본 설정 DNS 서버	위의 2단계에서 얻은 DNS 서버입니다.
보조 DNS 서버	위의 2단계에서 얻은 보조 DNS 서버입니다. 보조 DNS 서버가 나열되지 않은 경우 이 필드를 비워 둡니다.

Important

IP 주소를 현재 IP 주소가 아닌 다른 값으로 설정할 경우 인스턴스와의 연결이 끊어집니다.



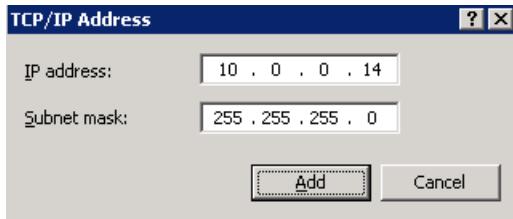
인스턴스가 DHCP 사용을 고정 주소 지정으로 변환하는 몇 초 동안은 Windows 인스턴스와 RDP 간의 연결이 끊어집니다. 인스턴스가 전과 동일한 IP 주소 정보를 유지하지만, 지금은 이 정보가 고정 정보이며 DHCP에 의해 관리되지 않습니다.

2단계: 인스턴스에 대한 보조 프라이빗 IP 주소 구성

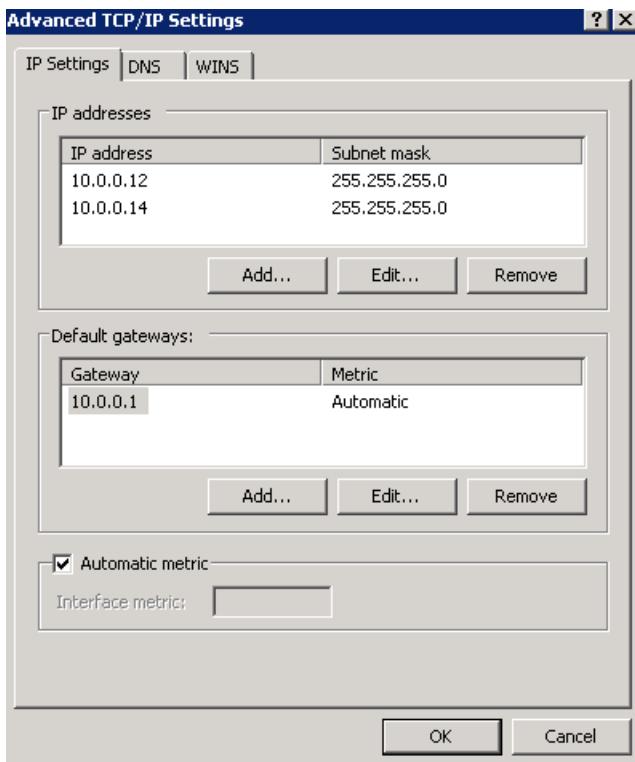
Windows 인스턴스에서 고정 IP 주소 지정을 설정하고 나면 두 번째 프라이빗 IP 주소를 준비할 수 있습니다.

보조 IP 주소를 구성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
3. 설명 탭에 보조 IP 주소를 적습니다.
4. 인스턴스에 연결합니다.
5. Windows 인스턴스에서 시작, 제어판을 선택합니다.
6. 네트워크 및 인터넷, 네트워크 및 공유 센터를 선택합니다.
7. 네트워크 인터페이스(로컬 영역 연결)를 선택한 다음, 속성을 선택합니다.
8. 로컬 영역 연결 속성 페이지에서 인터넷 프로토콜 버전 4(TCP/IPv4), 속성, 고급을 선택합니다.
9. 추가를 선택합니다.
10. TCP/IP 주소 대화 상자의 IP 주소에 보조 프라이빗 IP 주소를 입력합니다. 서브넷 마스크 필드에 [1단계: 인스턴스에서 고정 IP 주소 지정 구성 \(p. 564\)](#)의 주 프라이빗 IP 주소로 입력한 것과 동일한 서브넷 마스크를 입력하고 추가를 선택합니다.



11. IP 주소 설정을 확인하고 확인을 선택합니다.



12. 확인, 닫기를 선택합니다.
13. 보조 IP 주소가 운영 체제에 추가되었는지 확인하려면 명령 프롬프트에서 ipconfig /all 명령을 실행합니다.

3단계: 보조 프라이빗 IP 주소를 사용하도록 애플리케이션 구성

부 프라이빗 IP 주소를 사용하도록 애플리케이션을 구성할 수 있습니다. 예를 들어, 인스턴스에서 IIS의 웹 사이트를 실행 중인 경우 부 프라이빗 IP 주소를 사용하도록 IIS를 구성할 수 있습니다.

부 프라이빗 IP 주소를 사용하도록 IIS를 구성하려면

1. 인스턴스에 연결합니다.
2. IIS(인터넷 정보 서비스) 관리자를 엽니다.
3. 연결 창에서 사이트를 확장합니다.
4. 웹 사이트에 대한 컨텍스트 메뉴를 열고(마우스 오른쪽 버튼 클릭) 바인딩 편집을 선택합니다.
5. 사이트 바인딩 대화 상자의 유형에서 http, 편집을 선택합니다.
6. 사이트 바인딩 편집 대화 상자의 IP 주소에서 보조 프라이빗 IP 주소를 선택합니다. 기본적으로 각 웹 사이트는 모든 IP 주소의 HTTP 요청을 허용합니다.



7. 확인, 닫기를 선택합니다.

보조 네트워크 인터페이스 구성

두 번째 탄력적 네트워크 인터페이스를 인스턴스에 연결할 수 있습니다.

보조 네트워크 인터페이스를 구성하려면

- 1단계: [인스턴스에서 고정 IP 주소 지정 구성 \(p. 564\)](#)의 위 절차에 따라 주 탄력적 네트워크 인터페이스를 위한 고정 IP 주소 지정을 구성합니다.
- 동일한 절차에 따라 보조 탄력적 네트워크 인터페이스를 위한 고정 IP 주소 지정을 구성합니다.

시작 시 Windows 인스턴스에서 명령 실행

Amazon EC2에서 Windows 인스턴스를 시작할 때 사용자 데이터를 인스턴스에 전달하여 자동화된 구성 작업을 수행하는 데 사용할 수 있고, 인스턴스가 시작된 후에 스크립트를 실행하는 데 사용할 수도 있습니다. 인스턴스 사용자 데이터는 불분명한 데이터로 처리되므로, 해석은 인스턴스에 따라 다릅니다. 사용자 데이터는 EC2Launch v2([지원되는 미리 보기 AMI 및 다운로드 \(p. 465\)](#)), Windows Server 2016 이상에서 [EC2Launch \(p. 489\)](#), Windows Server 2012 R2 이하에서 [EC2Config \(p. 496\)](#)를 통해 처리됩니다.

AWS CloudFormation 템플릿의 `UserData` 속성 어셈블리 예제는 [Base64 인코딩 UserData 속성 및 AccessKey 및 SecretKey를 사용하는 Base64 인코딩 UserData 속성을 참조하세요.](#)

시작 시 Linux 인스턴스에서 명령을 실행하는 방법에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [시작 시 Linux 인스턴스에서 명령 실행](#)을 참조하십시오.

목차

- [사용자 데이터 스크립트 \(p. 568\)](#)
- [사용자 데이터 실행 \(p. 570\)](#)
- [사용자 데이터 및 콘솔 \(p. 572\)](#)
- [사용자 데이터 및 Windows PowerShell용 도구 \(p. 574\)](#)

사용자 데이터 스크립트

EC2Config 또는 EC2Launch에서 스크립트를 실행하려면 사용자 데이터에 추가할 때 해당 스크립트를 특수 태그 중 하나 안에 포함시켜야 합니다. 사용하는 태그는 명령 프롬프트 창에서 명령이 실행되는지(배치 명령) 아니면 Windows PowerShell을 사용하는지에 따라 달라집니다.

배치 스크립트와 Windows PowerShell 스크립트를 둘 다 지정할 경우, 인스턴스 사용자 데이터에 나타나는 순서와 관계 없이, 배치 스크립트가 먼저 실행되고 Windows PowerShell 스크립트가 다음에 실행됩니다.

사용자 데이터 스크립트에서 AWS CLI를 포함한 AWS API를 사용하는 경우 인스턴스를 시작할 때 인스턴스 프로파일을 사용해야 합니다. 인스턴스 프로필은 사용자 데이터 스크립트에서 API 호출을 실행하는데 필요한 적절한 AWS 자격 증명을 제공합니다. 자세한 내용은 [인스턴스 프로파일 \(p. 891\)](#) 단원을 참조하십시오. IAM 역할에 할당하는 권한은 API를 사용하여 호출하는 서비스에 따라 다릅니다. 자세한 내용은 [Amazon EC2의 IAM 역할](#)을 참조하세요.

스크립트 유형

- [배치 스크립트 구문 \(p. 569\)](#)
- [Windows PowerShell 스크립트 구문 \(p. 569\)](#)
- [YAML 구성 스크립트 구문 \(p. 570\)](#)
- [Base64 인코딩 \(p. 570\)](#)

배치 스크립트 구문

script 태그를 사용하여 배치 스크립트를 지정합니다. 줄 바꿈을 사용하여 명령을 구분합니다. 다음 예를 참조하십시오.

```
<script>
echo Current date and time >> %SystemRoot%\Temp\test.log
echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
```

기본적으로 사용자 데이터 스크립트는 인스턴스를 시작할 때 한 번만 실행됩니다. 인스턴스를 재부팅하거나 시작할 때마다 사용자 데이터 스크립트를 실행하려면 `<persist>true</persist>`를 사용자 데이터에 추가합니다.

```
<script>
echo Current date and time >> %SystemRoot%\Temp\test.log
echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
<persist>true</persist>
```

Windows PowerShell 스크립트 구문

AWS Windows AMI에는 [Windows PowerShell용 AWS 도구](#)가 포함되어 있으므로 사용자 데이터에서 이러한 cmdlet을 지정할 수 있습니다. IAM 역할을 인스턴스와 연결하는 경우, 인스턴스에서 실행되는 애플리케이션이 역할의 자격 증명을 사용하여 AWS 리소스(예: Amazon S3 버킷 등)에 액세스할 수 있기 때문에 cmdlet에 대한 자격 증명을 지정할 필요가 없습니다.

powershell 태그를 사용하여 Windows PowerShell 스크립트를 지정합니다. 줄 바꿈을 사용하여 명령을 구분합니다. 다음 예를 참조하십시오.

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
```

기본적으로 사용자 데이터 스크립트는 인스턴스를 시작할 때 한 번만 실행됩니다. 인스턴스를 재부팅하거나 시작할 때마다 사용자 데이터 스크립트를 실행하려면 `<persist>true</persist>`를 사용자 데이터에 추가합니다.

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
```

```
<persist>true</persist>
```

YAML 구성 스크립트 구문

EC2Launch v2를 사용하여 스크립트를 실행하는 경우 YAML 형식을 사용할 수 있습니다. EC2Launch v2에 대한 구성 작업, 세부 정보 및 예제를 보려면 [EC2Launch v2 작업 구성 \(p. 476\)](#) 단원을 참조하세요.

`executeScript` 작업을 통해 YAML 스크립트를 지정합니다.

PowerShell 스크립트를 실행하는 YAML 구문 예제

```
version: 1.0
tasks:
- task: executeScript
  inputs:
  - frequency: always
    type: powershell
    runAs: localSystem
  content: |-
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
```

배치 스크립트를 실행하는 YAML 구문 예제

```
version: 1.0
tasks:
- task: executeScript
  inputs:
  - frequency: always
    type: batch
    runAs: localSystem
  content: |-
    echo Current date and time >> %SystemRoot%\Temp\test.log
    echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
```

Base64 인코딩

Amazon EC2 API 또는 사용자 데이터의 base64 인코딩을 수행하지 않는 도구를 사용하는 경우, 직접 사용자 데이터를 인코딩해야 합니다. 그렇지 않을 경우, 실행할 `script` 또는 `powershell` 태그를 찾을 수 없다는 오류가 기록됩니다. 다음은 Windows PowerShell을 사용하여 인코딩하는 예제입니다.

```
$UserData =
[System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Script))
```

다음은 PowerShell을 사용하여 디코딩하는 예제입니다.

```
$Script =
[System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($UserData))
```

base64 인코딩에 대한 자세한 내용은 <http://tools.ietf.org/html/rfc4648>을 참조하십시오.

사용자 데이터 실행

기본적으로 모든 AWS Windows AMI는 초기 실행에 대해 활성화되는 사용자 데이터 실행 기능을 포함하고 있습니다. 다음 번에 인스턴스가 재부팅하거나 재시작할 때 사용자 데이터 스크립트가 실행되도록 지정할 수 있습니다. 또는 인스턴스가 재부팅하거나 재시작할 때마다 사용자 데이터 스크립트가 실행되도록 지정할 수도 있습니다.

사용자 데이터 스크립트는 임의 암호가 생성될 때 로컬 관리자 계정에서 실행됩니다. 그렇지 않으면 사용자 데이터 스크립트가 시스템 계정에서 실행됩니다.

인스턴스 시작

인스턴스 사용자 데이터의 스크립트는 인스턴스 초기 시작 중에 실행됩니다. `persist` 태그가 있는 경우 사용자 데이터 실행이 후속 재부팅이나 시작에 대해 활성화됩니다. EC2Launch v2, EC2Launch 및 EC2Config 용 로그 파일에는 표준 출력 및 표준 오류 스트림의 출력이 들어 있습니다.

EC2Launch v2

EC2Launch v2의 로그 파일은 `C:\ProgramData\Amazon\EC2Launch\log\agent.log`입니다.

Note

`C:\ProgramData` 폴더를 습길 수 있습니다. 폴더를 보려면 숨겨진 파일과 폴더를 표시해야 합니다.

다음 정보는 사용자 데이터가 실행될 때 기록됩니다.

- Info: Converting user-data to yaml format – 사용자 데이터가 XML 형식으로 제공된 경우
- Info: Initializing user-data state – 사용자 데이터 실행의 시작
- Info: Frequency is: always – 부팅할 때마다 사용자 데이터 작업이 실행되는 경우
- Info: Frequency is: once – 사용자 데이터 작업이 한 번만 실행되는 경우
- Stage: postReadyUserData execution completed – 사용자 데이터 실행의 끝

EC2Launch

EC2Launch에 대한 로그 파일은 `C:\ProgramData\Amazon\EC2-Windows\Launch\Log\userdataExecution.log`입니다.

Note

`C:\ProgramData` 폴더를 습길 수 있습니다. 폴더를 보려면 숨겨진 파일과 폴더를 표시해야 합니다.

다음 정보는 사용자 데이터가 실행될 때 기록됩니다.

- Userdata execution begins – 사용자 데이터 실행의 시작
- <persist> tag was provided: true – persist 태그가 있는 경우
- Running userdata on every boot – persist 태그가 있는 경우
- <powershell> tag was provided.. running powershell content – powershell 태그가 있는 경우
- <script> tag was provided.. running script content – script 태그가 있는 경우
- Message: The output from user scripts – 사용자 데이터 스크립트가 실행되면 출력은 로깅됩니다.

EC2Config

EC2Config의 로그 파일은 `C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2Config.log`입니다. 다음 정보는 사용자 데이터가 실행될 때 기록됩니다.

- Ec2HandleUserData: Message: Start running user scripts – 사용자 데이터 실행의 시작
- Ec2HandleUserData: Message: Re-enabled userdata execution – persist 태그가 있는 경우

- **Ec2HandleUserData:** Message: Could not find <persist> and </persist>-persist 태그가 없는 경우
- **Ec2HandleUserData:** Message: The output from user scripts - 사용자 데이터 스크립트가 실행되면 출력은 로깅됩니다.

이후 재부팅 또는 시작

인스턴스 사용자 데이터를 업데이트하는 경우 인스턴스를 재부팅하거나 시작할 때 사용자 데이터 스크립트가 자동으로 실행되지 않습니다. 하지만 인스턴스를 재부팅하거나 시작할 때 한 번만 또는 인스턴스를 재부팅하거나 시작할 때마다 사용자 데이터 스크립트가 실행되도록 사용자 데이터 실행을 활성화할 수 있습니다.

Shutdown with Sysprep(Sysprep으로 종료) 옵션을 선택하는 경우 이후 재부팅이나 시작에 대해 사용자 데이터 실행을 활성화하지 않았더라도 다음에 인스턴스가 재부팅되거나 시작될 때 사용자 데이터 스크립트가 실행됩니다. 사용자 데이터 스크립트는 이후 재부팅 또는 시작 시 실행되지 않습니다.

EC2Launch v2를 사용하여 사용자 데이터 실행을 활성화하려면(미리 보기 AMI)

- 처음 부팅할 때 사용자 데이터에서 작업을 실행하려면 frequency를 once로 설정합니다.
- 부팅할 때마다 사용자 데이터에서 작업을 실행하려면 frequency를 always로 설정합니다.

EC2Launch를 사용하여 사용자 데이터 실행을 활성화하려면(Windows Server 2016 이상)

1. Windows 인스턴스에 연결합니다.
2. PowerShell 명령 창을 열고 다음 명령을 실행합니다.

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -Schedule
```

3. Windows 인스턴스에서 연결을 해제합니다. 다음 번에 인스턴스가 시작될 때 업데이트된 스크립트를 실행하려면 인스턴스를 중지하고 사용자 데이터를 업데이트합니다. 자세한 내용은 [인스턴스 사용자 데이터 보기 및 업데이트 \(p. 573\)](#) 단원을 참조하십시오.

EC2Config를 사용하여 사용자 데이터 실행을 활성화하려면(Windows Server 2012 R2 이하)

1. Windows 인스턴스에 연결합니다.
2. Open C:\Program Files\Amazon\Ec2ConfigService\Ec2ConfigServiceSetting.exe.
3. 사용자 데이터에서 Enable UserData execution for next service start(다음 서비스 시작 시 사용자 데이터 실행 활성화)를 선택합니다.
4. Windows 인스턴스에서 연결을 해제합니다. 다음 번에 인스턴스가 시작될 때 업데이트된 스크립트를 실행하려면 인스턴스를 중지하고 사용자 데이터를 업데이트합니다. 자세한 내용은 [인스턴스 사용자 데이터 보기 및 업데이트 \(p. 573\)](#) 단원을 참조하십시오.

사용자 데이터 및 콘솔

인스턴스를 시작할 때 인스턴스 사용자 데이터를 지정할 수 있습니다. 인스턴스의 루트 볼륨이 EBS 볼륨이면 인스턴스를 중지하고 사용자 데이터를 업데이트 할 수도 있습니다.

시작 시 인스턴스 사용자 데이터 지정

인스턴스를 시작할 때 인스턴스 시작 마법사의 3단계: 인스턴스 세부 정보 구성 페이지에 있는 고급 세부 정보, 사용자 데이터에서 스크립트를 지정합니다. 다음 이미지의 예제에서는 파일 이름의 현재 날짜 및 시간을 사용하여 Windows 임시 폴더에 파일을 생성합니다. <persist>true</persist>를 포함할 경우 인스턴스를 재부팅하거나 시작할 때마다 스크립트가 실행됩니다. 텍스트를 선택한 경우 Amazon EC2 콘솔에서 base64 인코딩을 자동으로 수행합니다.

▼ Advanced Details

User data i

As text As file Input

```
<powershell>
$file = $env:SystemRoot+
New-Item $file -ItemType fi
</powershell>
<persist>true</persist>
```

인스턴스 사용자 데이터 보기 및 업데이트

모든 인스턴스의 인스턴스 사용자 데이터를 볼 수 있으며, 중지된 인스턴스의 인스턴스 사용자 데이터를 업데이트할 수 있습니다.

콘솔을 사용하여 인스턴스의 사용자 데이터를 업데이트하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 또는 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, 인스턴스 상태, 중지를 선택합니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

4. 확인 메시지가 표시되면 예, 중지를 선택합니다. 인스턴스가 중지하는 데 몇 분 정도 걸릴 수 있습니다.
5. 인스턴스가 선택된 상태에서 작업을 선택하고 인스턴스 설정을 선택한 후 사용자 데이터 보기/변경을 선택합니다. 인스턴스가 실행 중일 때는 사용자 데이터를 변경할 수 없습니다. 하지만 볼 수는 있습니다.
6. 사용자 데이터 보기/변경 대화 상자에서 사용자 데이터를 업데이트하고 저장을 선택합니다. 매번 인스턴스를 재부팅하거나 시작할 때마다 사용자 데이터 스크립트를 실행하려면 다음 예제와 같이 <persist>true</persist>를 추가합니다.

View/Change User Data

Instance ID: i-08240c2f0f225277a

User Data:

```
<powershell>
$file = $env:SystemRoot+"\Temp\"+(Get-Date).ToString("MM-dd-yyyy")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

Plain text Input is already base64 encoded

7. 인스턴스를 시작합니다. 이후의 재부팅 또는 시작에 대해 사용자 데이터 실행을 활성화한 경우 인스턴스 시작 프로세스의 일부로서 업데이트된 사용자 데이터 스크립트가 실행됩니다.

사용자 데이터 및 Windows PowerShell용 도구

Windows PowerShell용 도구를 사용하여 인스턴스의 사용자 데이터를 지정, 수정 및 확인할 수 있습니다. 인스턴스 메타데이터를 사용하여 인스턴스의 사용자 데이터를 보는 방법에 대한 자세한 내용은 [인스턴스 사용자 데이터 검색 \(p. 590\)](#) 단원을 참조하십시오. 사용자 데이터 및 AWS CLI에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [사용자 데이터 및 AWS CLI](#)를 참조하십시오.

예제: 시작 시 인스턴스 사용자 데이터 지정

인스턴스 사용자 데이터를 포함하는 텍스트 파일을 생성합니다. 매번 인스턴스를 재부팅하거나 시작할 때마다 사용자 데이터 스크립트를 실행하려면 다음 예제와 같이 <persist>true</persist>를 추가합니다.

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yyyy")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

인스턴스를 시작할 때 인스턴스 사용자 데이터를 지정하려면 [New-EC2Instance](#) 명령을 사용합니다. 이 명령은 사용자 데이터의 base64 인코딩을 수행하지 않습니다. 다음 명령을 사용하여 사용자 데이터를 script.txt 이름의 텍스트 파일로 인코딩합니다.

```
PS C:\> $Script = Get-Content -Raw script.txt
```

```
PS C:\> $UserData =  
[System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Script))
```

-UserData 파라미터를 사용하여 사용자 데이터를 New-EC2Instance 명령으로 전달합니다.

```
PS C:\> New-EC2Instance -ImageId ami-abcd1234 -MinCount 1 -MaxCount 1 -  
InstanceType m3.medium \  
-KeyName my-key-pair -SubnetId subnet-12345678 -SecurityGroupIds sg-1a2b3c4d \  
-UserData $UserData
```

예제: 중지된 인스턴스에 대한 인스턴스 사용자 데이터 업데이트

Edit-EC2InstanceAttribute 명령을 사용하여 중지된 인스턴스의 사용자 데이터를 수정할 수 있습니다.

새 스크립트를 포함하는 텍스트 파일을 생성합니다. 다음 명령을 사용하여 사용자 데이터를 new-script.txt 이름의 텍스트 파일로 인코딩합니다.

```
PS C:\> $NewScript = Get-Content -Raw new-script.txt  
PS C:\> $NewUserData =  
[System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($NewScript))
```

-UserData 및 -Value 파라미터를 사용하여 사용자 데이터를 지정합니다.

```
PS C:\> Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -Attribute userData -  
Value $NewUserData
```

예제: 인스턴스 사용자 데이터 보기

인스턴스의 사용자 데이터를 가져오려면 Get-EC2InstanceAttribute 명령을 사용합니다.

```
PS C:\> (Get-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -Attribute  
userData).UserData
```

다음은 예제 출력입니다. 사용자 데이터가 인코딩됩니다.

```
PHBvd2Vyc2h1bGw  
+DQpSZW5hbWUtQ29tcHV0ZXIgLU5ld05hbWUgdXNlcilkYXRhLXRlc3QNCjwvcG93ZXJzaGVsbD4=
```

다음 명령을 사용하여 인코딩된 사용자 데이터를 변수에 저장한 다음 디코딩합니다.

```
PS C:\> $UserData_encoded = (Get-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -  
Attribute userData).UserData  
PS C:  
\> [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($UserData_encoded))
```

다음은 예제 출력입니다.

```
<powershell>  
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")  
New-Item $file -ItemType file  
</powershell>  
<persist>true</persist>
```

예제: 태그 값과 일치하도록 인스턴스의 이름 바꾸기

태그 값을 읽으려면 첫 부팅 시 태그 값과 일치하도록 인스턴스의 이름을 바꾸고, 재부팅하고, Get-EC2Tag 명령을 사용합니다. 이 명령을 실행하려면 ec2:DescribeTags 권한이 있는 역할이 있어야 합니다. 태그 정

보를 메타데이터에서 사용할 수 없으므로 API 호출로 검색해야 하기 때문입니다. 인스턴스에 정책을 연결하는 방법에 대한 자세한 내용은 [IAM 역할을 인스턴스에 연결](#)을 참조하십시오

Note

이 스크립트는 2008 이전의 Windows Server 버전에서 실패합니다.

```
<powershell>
$instanceId = (invoke-webrequest http://169.254.169.254/latest/meta-data/instance-id - UseBasicParsing).content
$nameValue = (get-ec2tag -filter @{$Name="resource-id";$Value=$instanceid}, @{$Name="key";$Value="Name"}).Value
$pattern = "^(?!([0-9]{1,15})[a-zA-Z0-9-]{1,15}$"
##Verify Name Value satisfies best practices for Windows hostnames
If ($nameValue -match $pattern)
{
    Try
        {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
    Catch
        {$ErrorMessage = $_.Exception.Message
        Write-Output "Rename failed: $ErrorMessage"}
}
Else
{
    Throw "Provided name not a valid hostname. Please ensure Name value is between 1 and 15 characters in length and contains only alphanumeric or hyphen characters"
</powershell>
```

인스턴스 메타데이터 및 사용자 데이터

인스턴스 메타데이터는 실행 중인 인스턴스를 구성 또는 관리하는 데 사용될 수 있는 인스턴스 관련 데이터입니다. 인스턴스 메타데이터는 예를 들어 호스트 이름, 이벤트 및 보안 그룹과 같은 범주로 분류됩니다.

인스턴스 메타데이터를 사용하여 인스턴스를 시작할 때 지정한 사용자 데이터에도 액세스할 수 있습니다. 예를 들어, 인스턴스를 구성하기 위한 파라미터를 지정하거나 단순 스크립트를 포함시킬 수 있습니다. 일반 AMI를 빌드하고 사용자 데이터를 사용하여 시작 시간에 제공되는 구성 파일을 수정할 수 있습니다. 예를 들어, 다양한 소규모 비즈니스용으로 웹 서버를 운영하는 경우 모두 동일한 일반 AMI를 사용하고 시작 시 사용자 데이터에서 지정하는 Amazon S3 버킷에서 콘텐츠를 가져올 수 있습니다. 언제든지 새 고객을 추가하려면 고객에 대한 버킷을 생성하고, 콘텐츠를 추가한 다음, 사용자 데이터에서 코드에 제공된 고유의 버킷 이름으로 AMI를 시작합니다. 1개 이상의 인스턴스를 동시에 시작하는 경우 해당 동일 시작 인스턴스의 모든 인스턴스에서 사용자 데이터를 이용할 수 있습니다. 동일한 예약에 속한 각 인스턴스에는 수행할 작업을 제어하는 코드를 작성할 수 있도록 고유의 ami-launch-index 번호가 있습니다. 예를 들어, 첫 번째 호스트는 클러스터의 원래 노드로 자체 선택될 수 있습니다.

또한, EC2 인스턴스에는 인스턴스가 시작되었을 때 생성되는 인스턴스 자격 증명 문서와 같은 동적 데이터가 포함됩니다. 자세한 내용은 [동적 데이터 카테고리 \(p. 597\)](#) 단원을 참조하십시오.

Important

사용자는 인스턴스 자체 내에서 인스턴스 메타데이터 및 사용자 데이터에만 액세스할 수 있지만, 데이터는 인증 또는 암호화 방법으로 보호되지 않습니다. 인스턴스에 직접 액세스할 수 있는 모든 사람과 인스턴스에서 실행 중인 모든 소프트웨어는 메타데이터를 볼 수 있습니다. 따라서 암호 또는 수명이 긴 암호화 키와 같은 민감한 데이터를 사용자 데이터로 저장해서는 안 됩니다.

목차

- [인스턴스 메타데이터 서비스 구성 \(p. 577\)](#)
- [인스턴스 메타데이터 검색 \(p. 582\)](#)
- [인스턴스 사용자 데이터로 작업 \(p. 589\)](#)
- [동적 데이터 검색 \(p. 591\)](#)
- [인스턴스 메타데이터 카테고리 \(p. 591\)](#)
- [인스턴스 자격 증명 문서 \(p. 597\)](#)

인스턴스 메타데이터 서비스 구성

다음 방법 중 하나를 사용하여 실행 중인 인스턴스에서 인스턴스 메타데이터에 액세스할 수 있습니다.

- 인스턴스 메타데이터 서비스 버전 1(IMDSv1) – 요청/응답 방법
- 인스턴스 메타데이터 서비스 버전 2(IMDSv2) – 세션 지향 방법

기본적으로 IMDSv1 또는 IMDSv2를 사용하거나 둘 다 사용할 수 있습니다. 인스턴스 메타데이터 서비스는 특정 요청에 대해 IMDSv2에 고유한 PUT 또는 GET 헤더가 해당 요청에 있는지 여부에 따라 IMDSv1 및 IMDSv2 요청 간을 구별합니다.

로컬 코드 또는 사용자가 IMDSv2를 사용해야 하도록 각 인스턴스에서 인스턴스 메타데이터 서비스를 구성할 수 있습니다. IMDSv2를 사용해야 하도록 지정하면 IMDSv1은 더 이상 작동하지 않습니다. 자세한 내용은 [인스턴스 메타데이터 옵션 구성 \(p. 580\)](#) 단원을 참조하십시오.

인스턴스 메타데이터 서비스 버전 2 작동 방식

IMDSv2는 세션 지향 요청을 사용합니다. 세션 지향 요청을 사용하여 세션 기간을 정의하는 세션 토큰을 생성합니다. 세션 기간은 최소 1초에서 최대 6시간일 수 있습니다. 지정된 기간 중에는 후속 요청에 동일한 세션 토큰을 사용할 수 있습니다. 지정된 기간이 만료된 후에는 향후 요청에 사용할 새로운 세션 토큰을 생성할 수 있습니다.

다음 예제에서는 PowerShell 셸 스크립트와 IMDSv2를 사용하여 최상위 인스턴스 메타데이터 항목을 가져옵니다. 예제 명령:

- PUT 요청을 사용하여 6시간(21,600초) 동안 지속되는 세션 토큰을 생성합니다.
- `token`이라는 변수에 세션 토큰 헤더를 저장합니다.
- 토큰을 사용하여 최상위 메타데이터 항목을 요청합니다.

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/
```

토큰을 생성한 후에는 만료될 때까지 토큰을 재사용할 수 있습니다. 다음 예제 명령에서는 인스턴스를 시작하는 데 사용한 AMI의 ID를 가져오고 이전 예제에서 `$token`에 저장한 토큰을 재사용합니다.

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -uri http://169.254.169.254/latest/meta-data/ami-id
```

IMDSv2를 사용하여 인스턴스 메타데이터를 요청하는 경우 요청에는 다음이 포함되어야 합니다.

1. PUT 요청을 사용하여 인스턴스 메타데이터 서비스의 세션을 초기화합니다. PUT 요청은 후속 GET 요청에 포함되어야 하는 토큰을 인스턴스 메타데이터 서비스에 반환합니다. 토큰은 IMDSv2를 사용하여 메타데이터에 액세스하는 데 필요합니다.
2. 인스턴스 메타데이터 서비스에 대한 모든 GET 요청에 토큰을 포함시킵니다. 토큰 사용이 `required`로 설정되면 유효한 토큰이 없거나 만료된 토큰이 있는 요청은 401 – Unauthorized HTTP 오류 코드를 수신합니다. 토큰 사용 요구 사항을 변경하는 방법에 대한 자세한 내용은 AWS CLI Command Reference의 [modify-instance-metadata-options](#)를 참조하십시오.
 - 토큰은 인스턴스에 특정한 키입니다. 토큰은 다른 EC2 인스턴스에서 유효하지 않으며 해당 토큰이 생성된 인스턴스 외부에서 사용하려고 시도하면 거부됩니다.

- PUT 요청에는 토큰의 TTL(Time to Live)를 최대 6시간(21,600초)까지 초 단위로 지정하는 헤더가 포함되어야 합니다. 토큰은 논리 세션을 나타냅니다. TTL은 토큰이 유효한 시간 길이를 지정하며 따라서 세션 기간을 지정합니다.
- 토큰이 만료된 후 인스턴스 메타데이터에 계속 액세스하려면 다른 PUT를 사용하여 새 세션을 생성해야 합니다.
- 토큰을 재사용하거나 모든 요청에 새 토큰을 생성하도록 선택할 수 있습니다. 요청 수가 적은 경우 인스턴스 메타데이터 서비스에 액세스해야 할 때마다 토큰을 생성하고 즉시 사용하는 것이 더 간편할 수 있습니다. 하지만 효율성을 향상하려면 인스턴스 메타데이터를 요청해야 할 때마다 PUT 요청을 작성하는 대신 토큰에 더 긴 기간을 지정하고 토큰을 재사용할 수 있습니다. 각각 고유의 세션을 나타내는 동시 토큰 수에는 실질적인 제한이 없습니다. 하지만 IMDSv2는 표준 인스턴스 메타데이터 서비스 연결 및 조절 제한에 따라 여전히 제약됩니다. 자세한 내용은 [Throttling \(p. 588\)](#) 단원을 참조하십시오.

IMDSv2 인스턴스 메타데이터 요청에서는 HTTP GET 및 HEAD 메서드가 허용됩니다. PUT 요청은 X-Forwarded-For 헤더가 포함될 경우 거부됩니다.

기본적으로 PUT 요청에 대한 응답에는 IP 프로토콜 수준에서 1의 응답 흡 제한(TTL(Time to Live))이 있습니다. 흡 제한을 더 크게 만들어야 하는 경우 `modify-instance-metadata-options` 명령을 사용하여 흡 제한을 조정할 수 있습니다. 예를 들어, 인스턴스에서 실행 중인 컨테이너 서비스가 있는 경우 이전 버전과의 호환성을 위해 더 큰 흡 제한이 필요할 수 있습니다. 자세한 내용은 AWS CLI Command Reference의 `modify-instance-metadata-options`를 참조하십시오.

인스턴스 메타데이터 서비스 버전 2 사용으로 전환

인스턴스 메타데이터 서비스 버전 2(IMDSv2) 사용은 선택적입니다. 인스턴스 메타데이터 서비스 버전 1(IMDSv1)은 계속해서 무기한 지원됩니다. IMDSv2 사용으로 마이그레이션하도록 선택하는 경우 다음과 같은 도구와 전환 경로를 사용하는 것이 좋습니다.

IMDSv2로 전환하는 데 도움이 되는 도구

소프트웨어가 IMDSv1을 사용하는 경우 다음 도구를 사용하면 IMDSv2를 사용하도록 소프트웨어를 재구성하는 데 도움이 됩니다.

- AWS 소프트웨어: 최신 버전의 AWS SDK 및 CLI는 IMDSv2를 지원합니다. IMDSv2를 사용하려면 EC2 인스턴스에 최신 버전의 AWS SDK 및 CLI가 있는지 확인합니다. CLI 업데이트에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의 [최신 버전의 AWS CLI로 업그레이드](#)를 참조하십시오.
- CloudWatch: IMDSv2는 토큰 지원 세션을 사용하지만 IMDSv1은 사용하지 않습니다. `MetadataNoToken` CloudWatch 지표는 IMDSv1을 사용하는 인스턴스 메타데이터 서비스에 대한 호출 수를 추적합니다. 이 지표를 0까지 추적하면 모든 소프트웨어가 IMDSv2를 사용하도록 업그레이드되었는지 여부와 업그레이드된 시간을 확인할 수 있습니다. 자세한 내용은 [인스턴스 지표 \(p. 669\)](#) 단원을 참조하십시오.
- EC2 API 및 CLI로 업데이트: 기존 인스턴스의 경우 `modify-instance-metadata-options` CLI 명령(또는 `ModifyInstanceMetadataOptions` API)을 사용하여 IMDSv2를 사용해야 하도록 설정할 수 있습니다. 새 인스턴스의 경우 `run-instances` CLI 명령(또는 `RunInstances` API)과 `metadata-options` 파라미터를 사용하여 IMDSv2를 사용해야 하는 새 인스턴스를 시작할 수 있습니다.

Auto Scaling 그룹에서 시작한 모든 새 인스턴스에서 IMDSv2를 사용해야 하는 경우 Auto Scaling 그룹에서 시작 템플릿 또는 시작 구성을 사용할 수 있습니다. [시작 템플릿을 생성하거나 시작 구성을 생성할 때 IMDSv2를 반드시 사용하도록 MetadataOptions 파라미터를 구성해야 합니다.](#) 시작 템플릿 또는 시작 구성을 구성한 후 Auto Scaling 그룹은 새 시작 템플릿 또는 시작 구성을 사용하여 새 인스턴스를 시작하지만 기존 인스턴스는 영향을 받지 않습니다.

`modify-instance-metadata-options` CLI 명령(또는 `ModifyInstanceMetadataOptions` API)을 사용하여 기존 인스턴스에서 IMDSv2를 사용하도록 하거나 인스턴스를 종료하면 Auto Scaling 그룹이 시작 템플릿 또는 시작 구성을 정의된 인스턴스 메타데이터 옵션 설정으로 새 교체 인스턴스를 시작합니다.

시작 구성을 사용하는 Auto Scaling 그룹의 경우 [시작 구성을 시작 템플릿으로 바꿀 수 있습니다.](#)

- IAM 정책 및 SCP: IAM 조건을 통해 IAM 사용자가 IMDSv2를 사용하지 않는 인스턴스를 시작하지 못하도록 설정할 수 있습니다. 또한 IAM 조건을 사용하여 IAM 사용자가 IMDSv1을 다시 활성화하기 위해 실행 중

인 인스턴스를 수정할 수 없도록 설정할 수 있으며, 인스턴스에서 인스턴스 메타데이터 서비스를 사용할 수 있도록 설정할 수 있습니다.

`ec2:MetadataHttpTokens`, `ec2:MetadataHttpPutResponseHopLimit` 및 `ec2:MetadataHttpEndpoint` IAM 조건 키를 사용하여 [RunInstances](#) 및 [ModifyInstanceMetadataOptions](#) API와 해당 CLI 사용을 제어할 수 있습니다. 정책이 생성되고 API 호출의 파라미터가 조건 키를 사용하는 정책에 지정된 상태와 일치하지 않으면 API 또는 CLI 호출이 `UnauthorizedOperation` 응답과 함께 실패합니다. 이러한 조건 키는 IAM 정책 또는 AWS Organizations 서비스 제어 정책(SCP)에서 사용할 수 있습니다.

추가로, IMDSv1에서 IMDSv2로 변경을 시행하기 위한 추가 보호 계층을 선택할 수 있습니다. EC2 역할 자격 증명을 통해 호출되는 API에 관한 액세스 관리 계층에서는 IAM 정책 또는 AWS Organizations 서비스 제어 정책(SCP)에서 새 조건 키를 사용할 수 있습니다. 특히, IAM 정책에서 값이 2.0인 정책 조건 키 `ec2:RoleDelivery`를 사용하여 IMDSv1에서 얻은 EC2 역할 자격 증명으로 API 호출을 수행하면 `UnauthorizedOperation` 응답이 수신됩니다. SCP에 따라 필요한 조건을 사용하여 동일한 작업을 더 광범위하게 수행할 수 있습니다. 이렇게 하면 지정된 조건에 맞지 않게 API를 호출할 경우 `UnauthorizedOperation` 오류가 수신되기 때문에 실제로 IMDSv1을 통해 제공된 자격 증명을 사용하여 API를 호출할 수 없습니다. 예제 IAM 정책은 [인스턴스 메타데이터 작업 \(p. 879\)](#) 단원을 참조하십시오. 자세한 내용은 AWS Organizations 사용 설명서의 [서비스 제어 정책](#)을 참조하십시오.

IMDSv2에 액세스해야 하도록 설정하는 권장 경로

위의 도구를 사용하여 이 경로를 따라 IMDSv2로 전환하는 것이 좋습니다.

1단계: 시작 시

EC2 인스턴스에서 역할 자격 증명을 사용하는 SDK, CLI 및 소프트웨어를 IMDSv2 호환 버전으로 업데이트합니다. CLI 업데이트에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의 [최신 버전의 AWS CLI로 업그레이드](#)를 참조하십시오.

그런 다음, IMDSv2 요청을 사용하여 인스턴스 메타데이터에 직접 액세스하는(다시 말해서, SDK를 사용하지 않는) 소프트웨어를 변경합니다.

2단계: 전환 도중

CloudWatch 지표 `MetadataNoToken`을 사용하여 전환 진행률을 추적합니다. 이 지표는 인스턴스에서 IMDSv1을 사용하는 인스턴스 메타데이터 서비스에 대한 호출 수를 표시합니다. 자세한 내용은 [인스턴스 지표 \(p. 669\)](#) 단원을 참조하십시오.

3단계: 모든 인스턴스에서 모든 준비가 완료될 때

CloudWatch 지표 `MetadataNoToken`이 IMDSv1 사용량 0을 기록하면 모든 인스턴스에서 모든 준비가 완료됩니다. 이 단계에서 다음 작업을 수행할 수 있습니다.

- 기존 인스턴스의 경우: [modify-instance-metadata-options](#) 명령을 통해 IMDSv2를 사용해야 하도록 설정할 수 있습니다. 실행 중인 인스턴스에서 이렇게 변경할 수 있으며, 인스턴스를 다시 시작할 필요는 없습니다.
- 새 인스턴스의 경우: 새 인스턴스를 시작할 때 [run-instances](#) 명령을 사용하여 IMDSv2만 사용하도록 지정할 수 있습니다.

API 또는 AWS CLI를 통해서만 인스턴스 메타데이터 옵션을 지정할 수 있으며, 현재는 AWS Management 콘솔을 통해 이 옵션을 지정할 수 없습니다. 자세한 내용은 [인스턴스 메타데이터 옵션 구성 \(p. 580\)](#) 단원을 참조하십시오.

4단계: 모든 인스턴스가 IMDSv2로 전환될 때

`ec2:MetadataHttpTokens`, `ec2:MetadataHttpPutResponseHopLimit` 및 `ec2:MetadataHttpEndpoint` IAM 조건 키를 사용하여 [RunInstances](#) 및 [ModifyInstanceMetadataOptions](#)

API와 해당 CLI 사용을 제어할 수 있습니다. 정책이 생성되고 API 호출의 파라미터가 조건 키를 사용하는 정책에 지정된 상태와 일치하지 않으면 API 또는 CLI 호출이 UnauthorizedOperation 응답과 함께 실패합니다. 예제 IAM 정책은 [인스턴스 메타데이터 작업 \(p. 879\)](#) 단원을 참조하십시오.

인스턴스 메타데이터 옵션 구성

인스턴스 메타데이터 옵션을 통해 신규 또는 기존 인스턴스를 구성하여 다음과 같이 설정할 수 있습니다.

- 인스턴스 메타데이터를 요청할 때 IMDSv2를 사용해야 하도록 설정
- PUT 응답 흡수 제한 지정
- 인스턴스 메타데이터에 대한 액세스 비활성화

또한 IAM 정책 또는 SCP에서 IAM 조건 키를 사용하여 다음과 같이 설정할 수 있습니다.

- IMDSv2를 사용해야 하도록 구성된 경우에만 인스턴스를 시작하도록 허용
- 허용된 흡수 제한
- 인스턴스 메타데이터에 대한 액세스 비활성화

신규 또는 기존 인스턴스에서 인스턴스 메타데이터 옵션을 구성하려면 AWS SDK 또는 CLI를 사용합니다. 자세한 내용은 AWS CLI Command Reference의 [run-instances](#) 및 [modify-instance-metadata-options](#)를 참조하십시오.

Note

작업을 조심스럽게 진행해야 하며 무엇이든 변경하기 전에 세심하게 테스트해야 합니다. 다음에 유의하십시오.

- IMDSv2를 사용해야 하도록 설정하면 인스턴스 메타데이터 액세스에 IMDSv1를 사용하는 애플리케이션이나 에이전트는 중단됩니다.
- 인스턴스 메타데이터에 대한 모든 액세스를 고면 인스턴스 메타데이터 액세스를 이용하여 작동하는 애플리케이션이나 에이전트는 중단됩니다.

주제

- [새 인스턴스에 대한 인스턴스 메타데이터 옵션 구성 \(p. 580\)](#)
- [기존 인스턴스에 대한 인스턴스 메타데이터 옵션 구성 \(p. 581\)](#)

새 인스턴스에 대한 인스턴스 메타데이터 옵션 구성

인스턴스를 시작할 때 인스턴스에서 IMDSv2를 사용해야 하도록 설정할 수 있습니다. 또한 새 인스턴스에서 IMDSv2를 사용해야 하도록 설정되지 않은 경우 사용자가 새 인스턴스를 시작하지 못하게 하는 IAM 정책을 생성할 수 있습니다.

새 인스턴스에서 IMDSv2를 사용해야 하도록 설정하려면

다음 [run-instances](#) 예에서는 `--metadata-options`을 `HttpTokens=required`로 설정하여 `c3.large` 인스턴스를 시작합니다. 또한 `HttpTokens`의 값을 지정할 때 `HttpEndpoint`을 `enabled`로 설정해야 합니다. 메타데이터 검색 요청에 대해 보안 토큰 헤더가 `required`로 설정되어 있으므로 인스턴스 메타데이터를 요청할 때 인스턴스가 IMDSv2를 사용해야 합니다.

```
aws ec2 run-instances
  --image-id ami-0abcdef1234567890
  --instance-type c3.large
  ...
  --metadata-options "HttpEndpoint=enabled,HttpTokens=required"
```

모든 새 인스턴스에서 IMDSv2를 사용해야 하도록 설정하려면

IAM 사용자가 인스턴스 메타데이터 요청 시 IMDSv2를 사용해야 하는 인스턴스만 시작할 수 있도록 하려면 IMDSv2를 사용해야 한다는 조건을 충족한 후에만 인스턴스를 시작할 수 있도록 지정할 수 있습니다. IAM 정책 예제는 [인스턴스 메타데이터 작업 \(p. 879\)](#) 단원을 참조하십시오.

인스턴스 메타데이터에 대한 액세스를 끄려면

인스턴스 메타데이터에 대한 액세스를 끄려면 사용 중인 인스턴스 메타데이터 서비스의 버전에 관계없이 --metadata-options를 `HttpEndpoint=disabled`로 설정하여 인스턴스를 시작합니다. 나중에 `modify-instance-metadata-options` 명령을 사용하여 액세스를 켤 수 있습니다.

```
aws ec2 run-instances
  --image-id ami-0abcdef1234567890
  --instance-type c3.large
  ...
  --metadata-options "HttpEndpoint=disabled"
```

기존 인스턴스에 대한 인스턴스 메타데이터 옵션 구성

기존 인스턴스에서 IMDSv2를 사용해야 하도록 설정할 수 있습니다. 또한 PUT 응답 흌 제한을 변경하고 기존 인스턴스의 인스턴스 메타데이터에 대한 액세스를 비활성화할 수 있으며, 사용자가 기존 인스턴스의 인스턴스 메타데이터 옵션을 수정하지 못하도록 하는 IAM 정책을 생성할 수도 있습니다.

IMDSv2의 사용하도록 요구하려면

인스턴스 메타데이터 요청 시 IMDSv2를 사용해야 하도록 선택할 수 있습니다. `modify-instance-metadata-options` CLI 명령을 사용하고 `http-tokens` 파라미터를 `required`로 설정합니다. 또한 `http-tokens`의 값을 지정할 때 `http-endpoint`를 `enabled`로 설정해야 합니다.

```
aws ec2 modify-instance-metadata-options \
  --instance-id i-1234567898abcdef0 \
  --http-tokens required \
  --http-endpoint enabled
```

PUT 응답 흌 제한을 변경하려면

기존 인스턴스의 경우 PUT 응답 흌 제한에 대한 설정을 변경할 수 있습니다. `modify-instance-metadata-options` CLI 명령을 사용하고 `http-put-response-hop-limit` 파라미터를 필요한 흌 수로 설정합니다. 다음 예제에서는 옵 제한이 3으로 설정됩니다. 또한 `http-put-response-hop-limit`의 값을 지정할 때 `http-endpoint`를 `enabled`로 설정해야 합니다.

```
aws ec2 modify-instance-metadata-options \
  --instance-id i-1234567898abcdef0 \
  --http-put-response-hop-limit 3 \
  --http-endpoint enabled
```

IMDSv2를 사용하여 인스턴스에서 IMDSv1의 사용을 복원하려면

인스턴스 메타데이터를 요청할 때 `http-tokens`가 `optional`로 설정된 `modify-instance-metadata-options` CLI를 사용하여 IMDSv1의 사용을 복원할 수 있습니다.

```
aws ec2 modify-instance-metadata-options \
  --instance-id i-1234567898abcdef0 \
  --http-tokens optional \
  --http-endpoint enabled
```

인스턴스 메타데이터에 대한 액세스를 끄려면

사용 중인 인스턴스 메타데이터 서비스의 버전과 상관없이, 인스턴스 메타데이터 서비스의 HTTP 엔드포인트를 비활성화하여 인스턴스 메타데이터에 대한 액세스를 끌 수 있습니다. HTTP 엔드포인트를 활성화하여 언제든지 이 변경을 되돌릴 수 있습니다. [modify-instance-metadata-options](#) CLI 명령을 사용하고 `http-endpoint` 파라미터를 `disabled`로 설정합니다.

```
aws ec2 modify-instance-metadata-options \
--instance-id i-1234567898abcdef0 \
--http-endpoint disabled
```

`modify-instance-metadata-options`의 사용을 제어하려면

인스턴스 메타데이터 옵션을 수정할 수 있는 IAM 사용자를 제어하려면 지정된 역할을 가진 사용자 이외의 모든 사용자가 [ModifyInstanceMetadataOptions](#) API를 사용하지 못하도록 하는 정책을 지정합니다. IAM 정책 예제는 [인스턴스 메타데이터 작업 \(p. 879\)](#) 단원을 참조하십시오.

인스턴스 메타데이터 검색

실행 중인 인스턴스에서 인스턴스 메타데이터를 사용할 수 있기 때문에 Amazon EC2 콘솔 또는 AWS CLI를 사용할 필요가 없습니다. 이는 인스턴스에서 실행할 스크립트를 작성할 때 유용합니다. 예를 들어, 사용자는 인스턴스 메타데이터에서 인스턴스의 로컬 IP 주소에 액세스하여 외부 애플리케이션과의 연결을 관리할 수 있습니다.

인스턴스 메타데이터는 몇 가지 범주로 분류될 수 있습니다. 각 인스턴스 메타데이터 범주에 대한 설명은 [인스턴스 메타데이터 카테고리 \(p. 591\)](#) 단원을 참조하십시오.

실행 중인 인스턴스 내에 있는 인스턴스 메타데이터의 모든 범주를 보려면 다음 URI를 사용하십시오.

```
http://169.254.169.254/latest/meta-data/
```

IP 주소 169.254.169.254는 링크-로컬 주소이며 인스턴스에서만 유효합니다. 자세한 내용은 Wikipedia에서 [Link-local address](#)를 참조하십시오.

인스턴스 메타데이터 및 사용자 데이터를 가져오기 위해 사용되는 HTTP 요청 비용은 청구되지 않습니다.

명령 형식은 IMDSv1을 사용하는지 또는 IMDSv2를 사용하는지에 따라 다릅니다. 기본적으로 두 가지 인스턴스 메타데이터 서비스를 모두 사용할 수 있습니다. IMDSv2를 사용해야 하도록 설정하려면 [인스턴스 메타데이터 서비스 구성 \(p. 577\)](#) 단원을 참조하십시오.

PowerShell cmdlet을 사용하여 URI를 검색할 수 있습니다. 예를 들어, PowerShell 버전 3.0 이상을 실행 중인 경우 다음 cmdlet을 사용합니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET - Uri http://169.254.169.254/latest/meta-data/
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/
```

PowerShell을 사용하지 않으려면 GNU Wget 또는 cURL과 같은 타사 도구를 설치할 수 있습니다.

Important

Windows 인스턴스에 타사 도구를 설치한 경우 HTTP 호출 방법 및 출력 형식이 이 문서와 다를 수 있으므로 부속 문서를 주의 깊게 정독해야 합니다.

응답 및 오류 메시지

모든 인스턴스 메타데이터는 텍스트(HTTP 콘텐츠 유형 `text/plain`)로 반환됩니다.

특정 메타데이터 리소스를 요청하면 적절한 값이 반환되거나 소스를 이용할 수 없는 경우 `404 - Not Found` HTTP 오류 코드가 반환됩니다.

일반 메타데이터 리소스(/로 끝나는 URI)를 요청한 경우 이용 가능한 리소스 목록이 반환되거나 해당 리소스가 없는 경우 `404 - Not Found` HTTP 오류 코드가 반환됩니다. 목록 항목은 개별 라인에 표시되고 줄바꿈(ASCII 10)으로 끝납니다.

인스턴스 메타데이터 서비스 버전 2를 사용하여 수행한 요청의 경우 다음 HTTP 오류 코드가 반환될 수 있습니다.

- `400 - Missing or Invalid Parameters` – PUT 요청이 유효하지 않습니다.
- `401 - Unauthorized` – GET 요청이 유효하지 않은 토큰을 사용합니다. 권장되는 작업은 새 토큰을 생성하는 것입니다.
- `403 - Forbidden` – 요청이 허용되지 않거나 인스턴스 메타데이터 서비스가 꺼져 있습니다.

인스턴스 메타데이터 검색 예제

예제

- [인스턴스 메타데이터의 사용 가능한 버전 가져오기](#) (p. 583)
- [최고 수준 메타데이터 항목 가져오기](#) (p. 584)
- [사용 가능한 퍼블릭 키 목록 가져오기](#) (p. 586)
- [퍼블릭 키 0을 이용할 수 있는 형식 표시](#) (p. 586)
- [퍼블릭 키 0\(OpenSSH 키 형식\) 가져오기](#) (p. 587)
- [인스턴스에 대한 서브넷 ID 가져오기](#) (p. 588)

인스턴스 메타데이터의 사용 가능한 버전 가져오기

이 예제를 통해 이용 가능한 인스턴스 메타데이터 버전을 가져올 수 있습니다. 이 버전과 Amazon EC2 API 버전은 서로 상관이 없습니다. 이전 버전의 구조 및 정보를 사용하는 스크립트인 경우 이전 버전을 사용할 수 있습니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{ "X-aws-ec2-metadata-token" = $token } -Method GET - Uri http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
```

```
2008-09-01  
2009-04-04  
2011-01-01  
2011-05-01  
2012-01-12  
2014-02-25  
2014-11-05  
2015-10-20  
2016-04-19  
2016-06-30  
2016-09-02  
latest
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/  
1.0  
2007-01-19  
2007-03-01  
2007-08-29  
2007-10-10  
2007-12-15  
2008-02-01  
2008-09-01  
2009-04-04  
2011-01-01  
2011-05-01  
2012-01-12  
2014-02-25  
2014-11-05  
2015-10-20  
2016-04-19  
2016-06-30  
2016-09-02  
latest
```

최고 수준 메타데이터 항목 가져오기

이 예제는 최고 수준 메타데이터 항목을 가져옵니다. 자세한 내용은 [인스턴스 메타데이터 카테고리 \(p. 591\)](#) 단원을 참조하십시오.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" =  
"21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -  
Uri http://169.254.169.254/latest/meta-data/  
ami-id  
ami-launch-index  
ami-manifest-path  
block-device-mapping/  
hostname  
iam/  
instance-action  
instance-id  
instance-life-cycle  
instance-type  
local-hostname  
local-ipv4  
mac
```

```
metrics/  
network/  
placement/  
profile  
public-hostname  
public-ipv4  
public-keys/  
reservation-id  
security-groups  
services/
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/  
ami-id  
ami-launch-index  
ami-manifest-path  
block-device-mapping/  
hostname  
iam/  
instance-action  
instance-id  
instance-type  
local-hostname  
local-ipv4  
mac  
metrics/  
network/  
placement/  
profile  
public-hostname  
public-ipv4  
public-keys/  
reservation-id  
security-groups  
services/
```

다음 예제에서는 이전 예제에서 얻은 몇 가지 최상위 메타데이터 항목의 값을 가져옵니다. IMDSv2 요청에서는 토큰이 만료되지 않았다고 가정하고 이전 예제 명령에서 생성한 저장된 토큰을 사용합니다.

IMDSv2

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -  
Uri http://169.254.169.254/latest/meta-data/ami-id  
ami-0abcdef1234567890
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/ami-id  
ami-0abcdef1234567890
```

IMDSv2

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -  
Uri http://169.254.169.254/latest/meta-data/reservation-id  
r-0efghijk987654321
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/reservation-id  
r-0efghijk987654321
```

IMDSv2

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -  
Uri http://169.254.169.254/latest/meta-data/local-hostname  
ip-10-251-50-12.ec2.internal
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/local-hostname  
ip-10-251-50-12.ec2.internal
```

IMDSv2

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -  
Uri http://169.254.169.254/latest/meta-data/public-hostname  
ec2-203-0-113-25.compute-1.amazonaws.com
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-hostname  
ec2-203-0-113-25.compute-1.amazonaws.com
```

사용 가능한 퍼블릭 키 목록 가져오기

이 예제를 통해 이용 가능한 퍼블릭 키 목록을 획득할 수 있습니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" =  
"21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -  
Uri http://169.254.169.254/latest/meta-data/public-keys/  
0=my-public-key
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-  
keys/ 0=my-public-key
```

퍼블릭 키 0을 이용할 수 있는 형식 표시

이 예제는 퍼블릭 키 0을 이용할 수 있는 형식을 보여줍니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

퍼블릭 키 0(OpenSSH 키 형식) 가져오기

이 예제에서는 퍼블릭 키 0(OpenSSH 키 형식)을 획득합니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
ssh-rsa MIICiTCCAFICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMCVVMx
CzAJBgNVBAgTAlDbMRRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBASTC01BTSBDB25zb2x1MRIwEAYDVQQDEwluZXN0Q21sYWMxHzAd
BgkqhkiG9w0BCQEWEGB5vb25lQGftYXpbvi5jb20wHhcNMTEwNDI1MjAONTIxWhcN
MTIwNDI0MjAONTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAlDbMRRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC01BTSBDB25z
b2x1MRIwEAYDVQQDEwluZXN0Q21sYWMxHzAdBgkqhkiG9w0BCQEWEGB5vb25lQGft
YXpbvi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzzswY6786m86gpE
Ibb3OhjZnzcvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNY5f6GuoEDmFJ10ZxBHjJnyp3780D8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

```
ssh-rsa MIICiTCCAFICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMCVVMx
CzAJBgNVBAgTAlDbMRRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBASTC01BTSBDB25zb2x1MRIwEAYDVQQDEwluZXN0Q21sYWMxHzAd
BgkqhkiG9w0BCQEWEGB5vb25lQGftYXpbvi5jb20wHhcNMTEwNDI1MjAONTIxWhcN
MTIwNDI0MjAONTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAlDbMRRAwDgYD
VQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC01BTSBDB25z
b2x1MRIwEAYDVQQDEwluZXN0Q21sYWMxHzAdBgkqhkiG9w0BCQEWEGB5vb25lQGft
YXpbvi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzzswY6786m86gpE
Ibb3OhjZnzcvQAaRHhd1QWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJI1J00zbhNY5f6GuoEDmFJ10ZxBHjJnyp3780D8uTs7fLvjx79LjSTb
```

```
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

인스턴스에 대한 서브넷 ID 가져오기

이 예제에서는 인스턴스에 대한 서브넷 ID를 가져옵니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id subnet-be9b61d7
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id subnet-be9b61d7
```

Throttling

쿼리는 인스턴스당 인스턴스 메타데이터 서비스로 스로틀링되고, 한 인스턴스에서 인스턴스 메타데이터 서비스로의 동시 연결 수에도 제한이 있습니다.

인스턴스 메타데이터 서비스를 사용하여 AWS 보안 자격 증명을 가져올 경우, 모든 트랜잭션 중에 또는 많은 스레드나 프로세스에서 동시에 자격 증명을 쿼리하지 마십시오. 이렇게 하면 조절이 발생할 수 있습니다. 자격 증명 만료일이 다가오기 전까지는 자격 증명을 캐시에 저장하는 것이 좋습니다.

인스턴스 메타데이터 서비스를 액세스하는 동안 조절이 발생하면 지수 백오프 전략으로 쿼리를 다시 시도하십시오.

인스턴스 메타데이터 서비스 액세스 제한

로컬 방화벽 규칙을 사용하여 인스턴스 메타데이터 서비스에 대한 일부 또는 모든 프로세스의 액세스를 비활성화할 수 있습니다.

Windows 방화벽을 사용하여 액세스 제한

다음 PowerShell 예제에서는 기본 제공 Windows 방화벽을 사용하여 Internet Information Server 웹 서버(기본 설치 사용자 ID NT AUTHORITY\IUSR 기준)가 169.254.169.254에 액세스할 수 없도록 설정합니다. 이 예제에서는 거부 규칙을 사용하여 해당 사용자로 실행하는 모든 프로세스의 모든 인스턴스 메타데이터 요청(IMDSv1이든 IMDSv2든 상관없이)을 거부합니다.

```
PS C:\> $blockPrincipal = New-Object -TypeName System.Security.Principal.NTAccount ("NT AUTHORITY\IUSR")
PS C:\> $BlockPrincipalSID =
$blockPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $BlockPrincipalSDDL = "D:(A;;CC;;;$BlockPrincipalSID)"
PS C:\> New-NetFirewallRule -DisplayName "Block metadata service from IIS" -Action block -Direction out -Protocol TCP -RemoteAddress 169.254.169.254 -LocalUser $BlockPrincipalSDDL
```

또는 허용 규칙을 사용하여 특정 사용자 또는 그룹에 대한 액세스만 허용할 수도 있습니다. 허용 규칙을 사용하면 어떤 소프트웨어가 인스턴스 메타데이터에 액세스해야 하는지를 결정해야 하므로, 허용 규칙은 보안 관

점에서 더 관리하기 쉬울 수 있습니다. 허용 규칙을 사용하면 인스턴스의 소프트웨어나 구성을 나중에 변경하는 경우 소프트웨어가 메타데이터 서비스(액세스할 의도가 없는 서비스)에 액세스하도록 우발적으로 허용할 가능성이 낮습니다. 방화벽 규칙을 변경할 필요 없이 허용 그룹에서 사용자를 추가하고 제거할 수 있도록, 그룹 사용을 허용 규칙과 결합할 수도 있습니다.

다음 예제에서는 exceptionPrincipal에서 지정한 프로세스(이 예제에서는 trustworthy-users라는 그룹)를 제외하고, 변수 blockPrincipal에서 지정한 OS 그룹으로 실행하는 모든 프로세스(이 예제에서는 Windows 그룹 Everyone)가 인스턴스 메타데이터에 액세스할 수 없도록 설정합니다. Linux iptables의 ! --uid-owner trustworthy-user 규칙과 달리 Windows 방화벽은 다른 모든 보안 주체를 거부하여 특정 보안 주체만 허용하는 바로 가기 메커니즘을 제공하지 않으므로, 보안 주체 거부 및 허용을 모두 지정해야 합니다.

```
PS C:\> $blockPrincipal = New-Object -TypeName System.Security.Principal.NTAccount
          ("Everyone")
PS C:\> $BlockPrincipalSID =
          $blockPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $exceptionPrincipal = New-Object -TypeName System.Security.Principal.NTAccount
          ("trustworthy-users")
PS C:\> $ExceptionPrincipalSID =
          $exceptionPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $PrincipalSDDL = "O:LSD:(D;;CC;;;$ExceptionPrincipalSID)(A;;CC;;;
          $BlockPrincipalSID)"
PS C:\> New-NetFirewallRule -DisplayName "Block metadata service for
          $($blockPrincipal.Value), exception: $($exceptionPrincipal.Value)" -Action block -
          Direction out `

          -Protocol TCP -RemoteAddress 169.254.169.254 -LocalUser $PrincipalSDDL
```

Note

로컬 방화벽 규칙을 사용하려면 이전 예제 명령을 필요에 맞게 조정해야 합니다.

netsh 규칙을 사용하여 액세스 제한

netsh 규칙을 사용하여 모든 소프트웨어를 차단할 수 있지만, 이러한 규칙은 훨씬 더 유연합니다.

```
C:\> netsh advfirewall firewall add rule name="Block metadata service altogether" dir=out
      protocol=TCP remoteip=169.254.169.254 action=block
```

Note

- 로컬 방화벽 규칙을 사용하려면 이전 예제 명령을 필요에 맞게 조정해야 합니다.
- netsh 규칙은 승격된 명령 프롬프트에서 설정해야 하며, 특정 보안 주체를 거부하거나 허용하도록 설정할 수 없습니다.

인스턴스 사용자 데이터로 작업

인스턴스 사용자 데이터로 작업 시 다음 사항에 유의하십시오.

- 사용자 데이터는 base64로 인코딩해야 합니다. Amazon EC2 콘솔은 base64 인코딩을 수행하거나 base64로 인코딩된 입력을 수락할 수 있습니다.
- 사용자 데이터는 base64로 인코딩되기 전에 원시 16KB 형식으로 제한됩니다. base64 인코딩 이후 n 길이의 문자열 크기는 ceil(n/3)*4입니다.
- 사용자 데이터는 가져올 때 base64로 디코딩해야 합니다. 인스턴스 메타데이터 또는 콘솔을 사용하여 데이터를 가져오는 경우 데이터는 자동으로 디코딩됩니다.
- 사용자 데이터는 블루명 데이터로 취급됨: 제공한 것만을 살펴볼 수 있습니다. 해석 가능성은 인스턴스에 따라 다릅니다.

- 인스턴스를 중지하고 사용자 데이터를 수정한 다음 인스턴스를 시작할 경우 인스턴스를 시작할 때 업데이트된 사용자 데이터가 자동으로 실행되지 않습니다. 하지만 인스턴스를 할 때 한 번만 또는 매번 인스턴스를 재부팅하거나 시작할 때마다 업데이트된 사용자 데이터 스크립트가 실행되도록 설정을 구성할 수 있습니다.

시작 시 인스턴스 사용자 데이터 지정

인스턴스를 시작할 때 사용자 데이터를 지정할 수 있습니다. 사용자 데이터가 실행 시 한 번만 실행되도록 지정하거나 매번 인스턴스를 재부팅하거나 시작할 때마다 실행되도록 지정할 수 있습니다. 자세한 내용은 [시작 시 Windows 인스턴스에서 명령 실행 \(p. 568\)](#) 단원을 참조하십시오.

인스턴스 사용자 데이터 설정

루트 볼륨이 EBS 볼륨인 경우 인스턴스의 사용자 데이터를 중지된 상태에서 수정할 수 있습니다. 자세한 내용은 [인스턴스 사용자 데이터 보기 및 업데이트 \(p. 573\)](#) 단원을 참조하십시오.

인스턴스 사용자 데이터 검색

실행 중인 인스턴스 내에서 사용자 데이터를 가져오려면 다음 URI를 사용합니다.

```
http://169.254.169.254/latest/user-data
```

사용자 데이터를 요청하면 데이터 자체(콘텐츠 유형 application/octet-stream)가 반환됩니다.

이 예제는 쉼표로 구분된 텍스트로 제공된 사용자 데이터를 반환합니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

IMDSv1

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = Invoke-RestMethod -Headers @ {"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token} -Method GET -uri http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

이 예제는 스크립트로 제공된 사용자 데이터를 반환합니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @ {"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @ {"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/user-data
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
```

```
</powershell>
<persist>true</persist>
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/user-data
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

고유의 컴퓨터에서 인스턴스용 사용자 데이터를 가져오려면 [사용자 데이터 및 Windows PowerShell용 도구 \(p. 574\)](#) 단원을 참조하십시오.

동적 데이터 검색

실행 중인 인스턴스 내에서 동적 데이터를 가져오려면 다음 URI를 사용합니다.

```
http://169.254.169.254/latest/dynamic/
```

이 예제는 높은 수준의 인스턴스 자격 증명 범주를 가져오는 방법을 보여 줍니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" =
"21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token

PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -
Uri http://169.254.169.254/latest/dynamic/instance-identity/
document
rsa2048
pkcs7
signature
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/dynamic/instance-identity/
document
rsa2048
pkcs7
signature
```

동적 데이터 및 가져오기 방법의 예제에 대한 자세한 내용은 [인스턴스 자격 증명 문서 \(p. 597\)](#) 단원을 참조하십시오.

인스턴스 메타데이터 카테고리

다음 표는 인스턴스 메타데이터의 카테고리를 목록으로 표시합니다.

Note

Amazon EC2에서 새 인스턴스 메타데이터 범주를 릴리스하면 새 범주에 대한 인스턴스 메타데이터를 기존 인스턴스에서 사용하지 못할 수 있습니다. 기존 인스턴스에서 인스턴스 메타데이터를 사용할 수 있도록 하려면 인스턴스를 [종지했다가 시작해야 \(p. 440\)](#) 합니다.

Important

다음 표의 범주 이름 중 일부는 해당 인스턴스에만 있는 데이터 자리 표시자입니다. 예를 들어, **mac**은 네트워크 인터페이스의 MAC 주소를 나타냅니다. 이 자리 표시자를 실제 값으로 바꿔야 합니다.

테스트	설명	릴리스 날짜
ami-id	인스턴스를 시작하기 위해 사용된 AMI ID.	버전 1.0
ami-launch-index	1개 이상의 인스턴스를 동시에 시작하는 경우 이 값은 인스턴스가 시작된 순서를 나타냅니다. 첫 번째 인스턴스의 값은 0입니다.	버전 1.0
ami-manifest-path	Amazon S3에 위치한 AMI 매니페스트 파일 경로. Amazon EBS 지원 AMI를 사용하여 인스턴스를 시작한 경우 반환되는 결과는 unknown입니다.	버전 1.0
ancestor-ami-ids	이 AMI를 생성하기 위해 다시 번들링 된 모든 인스턴스의 AMI ID. 이 값은 AMI 매니페스트 파일에 <code>ancestor-ami-ids</code> 키가 있는 경우에만 존재합니다.	2007-10-10
block-device-mapping/ami	루트/부트 파일 시스템을 포함하는 가상 디바이스.	2007-12-15
block-device-mapping/ebs N	Amazon EBS 볼륨과 연결된 가상 디바이스. Amazon EBS 볼륨은 시작 시 존재하는 경우 또는 인스턴스를 마지막으로 시작한 시점에만 메타데이터에서 사용할 수 있습니다. N은 Amazon EBS 볼륨의 색인을 나타냅니다(ebs1 또는 ebs2 등).	2007-12-15
block-device-mapping/eph emeral N	모든 비 NVMe 인스턴스 스토어 볼륨의 가상 디바이스입니다. N은 각 볼륨의 인덱스를 나타냅니다. 블록 디바이스 매핑에 있는 인스턴스 스토어 볼륨 수는 인스턴스에 대한 실제 인스턴스 스토어 볼륨 수와 일치하지 않을 수도 있습니다. 인스턴스 유형은 인스턴스에 사용 가능한 인스턴스 스토어 볼륨 수를 결정합니다. 블록 디바이스 매핑에 있는 인스턴스 스토어 볼륨 수가 인스턴스에 사용 가능한 수를 초과한 경우 추가 인스턴스 스토어 볼륨이 무시됩니다.	2007-12-15
block-device-mapping/root	루트 디바이스 또는 루트(/ 또는 C:) 파일 시스템이 특정 인스턴스와 연결된 경우 가상 디바이스의 파티션과 연결된 가상 디바이스 또는 파티션입니다.	2007-12-15
block-device-mapping/swap	swap와 연결된 가상 디바이스. 항상 존재하는 것은 아님.	2007-12-15

테스트	설명	릴리스 날짜
<code>elastic-gpus/ associations/<i>elastic-gpu-id</i></code>	인스턴스에 연결된 탄력적 GPU가 있는 경우 탄력적 GPU에 대한 정보(해당 ID 및 연결 정보 포함)를 비롯한 JSON 문자열을 포함합니다.	2016-11-30
<code>elastic-inference/ associations/<i>eia-id</i></code>	인스턴스에 연결된 탄력적 추론 액셀러레이터가 있는 경우, 탄력적 추론 액셀러레이터에 대한 정보(해당 ID 및 유형 포함)를 비롯한 JSON 문자열을 포함합니다.	2018-11-29
<code>events/maintenance/history</code>	인스턴스에 대해 완료되거나 취소한 유지 관리 이벤트가 있다면, 이벤트에 관한 정보가 있는 JSON 문자열이 포함됩니다. 자세한 내용은 완료되거나 취소된 이벤트에 대한 이벤트 기록 보기 (p. 660) 를 참조하십시오.	2018-08-17
<code>events/maintenance/sched uled</code>	인스턴스에 대해 활성화된 유지 관리 이벤트가 있다면, 이벤트에 관한 정보가 있는 JSON 문자열이 포함됩니다. 자세한 내용은 예약된 이벤트 보기 (p. 656) 단원을 참조하십시오.	2018-08-17
<code>hostname</code>	인스턴스의 프라이빗 IPv4 DNS 호스트 이름. 다중 네트워크 인터페이스가 존재하는 경우 eth0 디바이스를 의미함(디바이스 번호가 0인 디바이스).	버전 1.0
<code>iam/info</code>	인스턴스 시작 시 IAM 역할이 연결되어 있을 경우, 인스턴스의 LastUpdated date, InstanceProfileArn 및 InstanceProfileId 등 마지막으로 인스턴스 프로파일이 업데이트된 시간 관련 정보를 포함합니다. 그렇지 않을 경우 제공되지 않습니다.	2012-01-12
<code>iam/security-credentials/ role-name</code>	인스턴스 시작 시 IAM 역할이 연결되어 있을 경우 <i>role-name</i> 은 역할 이름이고 <i>role-name</i> 에는 이 역할과 연결된 임시 보안 자격 증명이 들어 있습니다(자세한 내용은 인스턴스 메타데이터에서 보안 자격 증명 검색 (p. 891) 참조). 그렇지 않을 경우 제공되지 않습니다.	2012-01-12
<code>identity-credentials/ec2/ info</code>	[내부 전용으로 예약됨] AWS에서 나머지 Amazon EC2 인프라에 대해 인스턴스를 식별하는데 사용하는 자격 증명 관련 정보.	2018-05-23
<code>identity-credentials/ec2/ security-credentials/ec2- instance</code>	[내부 전용으로 예약됨] AWS에서 나머지 Amazon EC2 인프라에 대해 인스턴스를 식별하는데 사용하는 자격 증명.	2018-05-23

테스트	설명	릴리스 날짜
<code>instance-action</code>	번들링을 준비하기 위해 재부팅되어야 함을 인스턴스에 통지합니다. 유효한 값: none shutdown bundle-pending.	2008-09-01
<code>instance-id</code>	이 인스턴스의 ID.	버전 1.0
<code>instance-life-cycle</code>	이 인스턴스의 구매 옵션입니다. 자세한 내용은 인스턴스 구입 옵션 (p. 200) 단원을 참조하십시오.	2019년 10월 1일
<code>instance-type</code>	인스턴스 유형. 자세한 내용은 인스턴스 유형 (p. 114) 단원을 참조하십시오.	2007-08-29
<code>kernel-id</code>	이 인스턴스와 함께 시작한 커널 ID(해당하는 경우).	2008-02-01
<code>local-hostname</code>	인스턴스의 프라이빗 IPv4 DNS 호스트 이름. 다중 네트워크 인터페이스가 존재하는 경우 eth0 디바이스를 의미함(디바이스 번호가 0인 디바이스).	2007-01-19
<code>local-ipv4</code>	인스턴스의 프라이빗 IPv4 주소. 다중 네트워크 인터페이스가 존재하는 경우 eth0 디바이스를 의미함(디바이스 번호가 0인 디바이스).	버전 1.0
<code>mac</code>	인스턴스의 미디어 액세스 제어(MAC) 주소. 다중 네트워크 인터페이스가 존재하는 경우 eth0 디바이스를 의미함(디바이스 번호가 0인 디바이스).	2011-01-01
<code>metrics/vhostmd</code>	더 이상 사용할 수 없습니다.	2011-05-01
<code>network/interfaces/macs/mac/device-number</code>	해당 인터페이스와 연결된 고유한 디바이스 번호. 이 디바이스 번호는 디바이스 이름과 부합됩니다. 예를 들어 device-number 2는 eth2 디바이스의 번호입니다. 이 범주는 AWS CLI용 Amazon EC2 API 및 EC2 명령에서 사용하는 DeviceIndex 및 device-index 필드와 부합됩니다.	2011-01-01
<code>network/interfaces/macs/mac/interface-id</code>	네트워크 인터페이스의 ID입니다.	2011-01-01
<code>network/interfaces/macs/mac/ipv4-associations/public-ip</code>	각 퍼블릭 IP 주소와 연결되고 해당 인터페이스에 할당된 프라이빗 IPv4 주소.	2011-01-01
<code>network/interfaces/macs/mac/ipv6s</code>	IPv6 주소는 인터페이스와 연결됩니다. VPC에서 시작된 인스턴스인 경우에만 반환됩니다.	2016-06-30
<code>network/interfaces/macs/mac/local-hostname</code>	인터페이스의 로컬 호스트 이름.	2011-01-01

테스트	설명	릴리스 날짜
network/interfaces/macs/mac/local-ipv4s	프라이빗 IPv4 주소는 인터페이스와 연결됩니다.	2011-01-01
network/interfaces/macs/mac/mac	인스턴스의 MAC 주소.	2011-01-01
network/interfaces/macs/mac/owner-id	네트워크 인터페이스 소유자 ID. 다중 인터페이스 환경에서 인터페이스는 Elastic Load Balancing 등 타사 제품이 연결될 수 있습니다. 인터페이스 상의 트래픽은 항상 인터페이스 소유자에게 청구됩니다.	2011-01-01
network/interfaces/macs/mac/public-hostname	인터넷 퍼블릭 DNS(IPv4). 이 범주는 enableDnsHostnames 속성이 true로 설정된 경우에만 반환됩니다. 자세한 내용은 VPC에서 DNS 사용하기 단원을 참조하십시오.	2011-01-01
network/interfaces/macs/mac/public-ipv4s	인터넷 퍼블릭 IP 주소 또는 탄력적 IP 주소입니다. 인스턴스에는 다중 IPv4 주소가 있을 수 있습니다.	2011-01-01
network/interfaces/macs/mac/security-groups	네트워크 인터페이스에 속한 보안 그룹.	2011-01-01
network/interfaces/macs/mac/security-group-ids	네트워크 인터페이스에 속한 보안 그룹의 ID.	2011-01-01
network/interfaces/macs/mac/subnet-id	인터넷 퍼블릭 IP 주소가 위치하는 서브넷 ID.	2011-01-01
network/interfaces/macs/mac/subnet-ipv4-cidr-block	인터넷 퍼블릭 IP 주소가 위치하는 서브넷의 IPv4 CIDR 블록.	2011-01-01
network/interfaces/macs/mac/subnet-ipv6-cidr-blocks	인터넷 퍼블릭 IP 주소가 위치하는 서브넷의 IPv6 CIDR 블록.	2016-06-30
network/interfaces/macs/mac/vpc-id	인터넷 퍼블릭 IP 주소가 위치하는 VPC의 ID.	2011-01-01
network/interfaces/macs/mac/vpc-ipv4-cidr-block	VPC의 기본 IPv4 CIDR 블록.	2011-01-01
network/interfaces/macs/mac/vpc-ipv4-cidr-blocks	VPC에 대한 IPv4 CIDR 블록.	2016-06-30
network/interfaces/macs/mac/vpc-ipv6-cidr-blocks	인터넷 퍼블릭 IP 주소가 위치하는 VPC의 IPv6 CIDR 블록.	2016-06-30
placement/availability-zone	인스턴스가 시작된 가용 영역.	2008-02-01
placement/availability-zone-id	인스턴스가 시작된 정적 가용 영역 ID입니다. 가용 영역 ID는 계정 간에 일관성이 있습니다. 그러나 가용 영역과는 다를 수 있으며, 가용 영역은 계정에 따라 다를 수 있습니다.	2020-08-24

테스트	설명	릴리스 날짜
placement/group-name	인스턴스가 시작된 배치 그룹의 이름입니다.	2020-08-24
placement/host-id	인스턴스가 시작된 호스트의 ID입니다. 전용 호스트에만 해당됩니다.	2020-08-24
placement/partition-number	인스턴스가 시작된 파티션의 번호입니다.	2020-08-24
placement/region	인스턴스가 시작된 AWS 리전입니다.	2020-08-24
product-codes	인스턴스와 연결된 AWS Marketplace 제품 코드입니다(있는 경우).	2007-03-01
public-hostname	인스턴스의 퍼블릭 DNS. 이 범주는 enableDnsHostnames 속성이 true로 설정된 경우에만 반환됩니다. 자세한 내용은 Amazon VPC 사용 설명서의 VPC에서 DNS 사용 을 참조하십시오.	2007-01-19
public-ipv4	퍼블릭 IPv4 주소. 인스턴스와 탄력적 IP 주소가 연결된 경우 반환된 값은 탄력적 IP 주소입니다.	2007-01-19
public-keys/0/openssh-key	퍼블릭 키. 시작 시에 인스턴스가 제공된 경우에만 사용할 수 있습니다.	버전 1.0
ramdisk-id	시작 시에 지정된 RAM의 ID(해당하는 경우).	2007-10-10
reservation-id	예약 ID:	버전 1.0
security-groups	인스턴스에 적용된 보안 그룹의 이름. 시작 이후 인스턴스의 보안 그룹을 변경할 수 있습니다. 해당 변경은 여기 및 network/interfaces/macs/ <i>mac</i> /security-groups에 반영됩니다.	버전 1.0
services/domain	리전의 AWS 리소스에 대한 도메인입니다.	2014-02-25
services/partition	리소스가 있는 파티션. 표준 AWS 리전에서 파티션은 aws입니다. 리소스가 다른 파티션에 있는 경우 파티션은 aws- <i>partitionname</i> 입니다. 예를 들어 중국(베이징) 리전에 있는 리소스의 파티션은 aws-cn입니다.	2015-10-20
spot/instance-action	항목이 발생할 때 작업(최대 절전 모드, 중지 또는 종료)과 작업이 이루어지는 대략의 시간(UTC)입니다. 이 항목은 스팟 인스턴스가 최대 절전 모드, 중지, 종료로 표시된 경우에만 존재합니다. 자세한 내용은 instance-action (p. 315) 단원을 참조하십시오.	2016-11-15

테스트	설명	릴리스 날짜
spot/termination-time	<p>스팟 인스턴스의 운영 체제가 종료 신호를 수신하는 UTC 기준 예상 시간. Amazon EC2가 스팟 인스턴스에 종료 표시를 한 경우에만 이 항목이 존재하고 시간 값(예: 2015-01-05T18:02:00Z)이 포함됩니다. 사용자가 스팟 인스턴스를 직접 종료한 경우 종료 시간 항목에 시간이 설정되지 않습니다. 자세한 내용은 termination-time (p. 315) 단원을 참조하십시오.</p>	2014-11-05

동적 데이터 카테고리

다음 표는 동적 데이터의 카테고리를 목록으로 표시합니다.

테스트	설명	릴리스 날짜
fws/instance-monitoring	고객이 CloudWatch에서 1분 세부 모니터링을 설정했는지 보여주는 값. 유효한 값: enabled disabled	2009-04-04
instance-identity/document	인스턴스 ID, 프라이빗 IP 주소 등 인스턴스 속성을 포함하는 JSON. 인스턴스 자격 증명 문서 (p. 597) 단원을 참조하십시오.	2009-04-04
instance-identity/pkcs7	문서의 신뢰성 및 서명 내용을 검증하는 데 사용됩니다. 인스턴스 자격 증명 문서 (p. 597) 단원을 참조하십시오.	2009-04-04
instance-identity/signature	출처 및 신뢰성을 검증하기 위해 다른 사용자가 사용할 수 있는 데이터. 인스턴스 자격 증명 문서 (p. 597) 단원을 참조하십시오.	2009-04-04

인스턴스 자격 증명 문서

시작하는 각 인스턴스에는 인스턴스 자체에 대한 정보를 제공하는 인스턴스 자격 증명 문서가 있습니다. 인스턴스 자격 증명 문서를 사용하여 인스턴스의 속성을 확인할 수 있습니다.

인스턴스가 시작되고 인스턴스 메타데이터 서비스를 통해 일반 텍스트 JSON 형식으로 노출될 때 인스턴스 자격 증명 문서가 생성됩니다. IP 주소 169.254.169.254는 링크-로컬 주소이며 인스턴스에서만 유효합니다. 자세한 내용은 Wikipedia에서 [Link-local address](#)를 참조하십시오.

언제든지 실행 중인 인스턴스에서 인스턴스 자격 증명 문서를 검색할 수 있습니다. 인스턴스 자격 증명 문서에는 다음 정보가 포함됩니다.

테스트	설명
devpayProductCodes	사용되지 않음.
marketplaceProductCodes	인스턴스를 시작하는 데 사용된 AMI의 AWS Marketplace 제품 코드.
availabilityZone	인스턴스가 실행 중인 가용 영역.
privateIp	인스턴스의 프라이빗 IPv4 주소.

테스트	설명
version	인스턴스 자격 증명 문서 형식의 버전.
instanceId	인스턴스의 ID
billingProducts	인스턴스를 시작하는 데 사용된 AMI의 결제 제품 코드.
instanceType	인스턴스의 인스턴스 유형입니다.
accountId	인스턴스를 시작한 AWS 계정의 ID.
imageId	인스턴스를 시작하는 데 사용된 AMI의 ID.
pendingTime	인스턴스가 시작된 날짜와 시간.
architecture	인스턴스를 시작하는 데 사용된 AMI의 아키텍처(i386 x86_64 arm64).
kernelId	인스턴스에 연결된 커널의 ID(해당되는 경우).
ramdiskId	인스턴스에 연결된 RAM 디스크의 ID(해당되는 경우).
region	인스턴스가 실행 중인 리전.

일반 텍스트 인스턴스 자격 증명 문서 검색

일반 텍스트 인스턴스 자격 증명 문서를 검색하려면

인스턴스에 연결하고 인스턴스에 사용되는 IMDS(인스턴스 메타데이터 서비스) 버전에 따라 다음 명령 중 하나를 실행합니다.

IMDSv2

```
PS C:\> $Token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/token).Content
PS C:\> (Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/document).Content
```

IMDSv1

```
PS C:\> (Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/document).Content
```

다음은 예제 출력입니다.

```
{
  "devpayProductCodes" : null,
  "marketplaceProductCodes" : [ "1abc2defghijklmnopqrstuvwxyz" ],
  "availabilityZone" : "us-west-2b",
  "privateIp" : "10.158.112.84",
  "version" : "2017-09-30",
  "instanceId" : "i-1234567890abcdef0",
  "billingProducts" : null,
  "instanceType" : "t2.micro",
  "accountId" : "123456789012",
  "imageId" : "ami-5fb8c835",
  "pendingTime" : "2016-11-19T16:32:11Z",
```

```
"architecture" : "x86_64",
"kernelId" : null,
"ramdiskId" : null,
"region" : "us-west-2"
}
```

인스턴스 자격 증명 문서 확인

중요한 목적으로 인스턴스 자격 증명 문서의 내용을 사용하려는 경우 사용 전에 해당 내용과 신뢰성을 확인해야 합니다.

일반 텍스트 인스턴스 자격 증명 문서에는 해시되고 암호화된 3개의 서명이 함께 제공됩니다. 이러한 서명을 사용하여 인스턴스 자격 증명 문서의 출처 및 신뢰성과 포함된 정보를 확인할 수 있습니다. 다음과 같은 서명이 제공됩니다.

- Base64 인코딩 서명—RSA 키 페어를 사용하여 암호화된 인스턴스 자격 증명 문서의 Base64로 인코딩된 SHA256 해시입니다.
- PKCS7 서명—DSA 키 페어를 사용하여 암호화된 인스턴스 자격 증명 문서의 SHA1 해시입니다.
- RSA-2048 서명—RSA-2048 키 페어를 사용하여 암호화된 인스턴스 자격 증명 문서의 SHA256 해시입니다.

각 서명은 인스턴스 메타데이터의 서로 다른 엔드포인트에서 사용할 수 있습니다. 해시 및 암호화 요구 사항에 따라 이러한 서명 중 하나를 사용할 수 있습니다. 서명을 확인하려면 해당 AWS 퍼블릭 인증서를 사용해야 합니다.

Important

base64 인코딩 서명 또는 RSA2048 서명을 사용하여 인스턴스 자격 증명 문서를 확인하려면 [AWS Support](#)에서 해당 AWS 퍼블릭 인증서를 요청해야 합니다.

다음 주제에서는 각 서명을 사용하여 인스턴스 자격 증명 문서를 확인하는 자세한 단계를 설명합니다.

- [PKCS7 서명을 사용하여 인스턴스 자격 증명 문서 확인 \(p. 599\)](#)
- [base64 인코딩 서명을 사용하여 인스턴스 자격 증명 문서 확인 \(p. 603\)](#)
- [RSA-2048 서명을 사용하여 인스턴스 자격 증명 문서 확인 \(p. 604\)](#)

PKCS7 서명을 사용하여 인스턴스 자격 증명 문서 확인

이 주제에서는 PKCS7 서명 및 AWS DSA 퍼블릭 인증서를 사용하여 인스턴스 자격 증명 문서를 확인하는 방법을 설명합니다.

사전 조건

이 절차를 수행하려면 `System.Security` Microsoft .NET Core 클래스가 필요합니다. PowerShell 세션에 클래스를 추가하려면 다음 명령을 실행합니다.

```
PS C:\> Add-Type -AssemblyName System.Security
```

Note

이 명령은 현재 PowerShell 세션에만 클래스를 추가합니다. 새 세션을 시작하는 경우 명령을 다시 실행해야 합니다.

PKCS7 서명 및 AWS DSA 퍼블릭 인증서를 사용하여 인스턴스 자격 증명 문서를 확인하려면

1. 인스턴스에 연결합니다.

2. 인스턴스 메타데이터에서 PKCS7 서명을 검색하여 바이트 배열로 변환한 다음 \$Signature라는 변수에 추가합니다. 인스턴스에 사용되는 IMDS 버전에 따라 다음 명령 중 하나를 사용합니다.

IMDSv2

```
PS C:\> $Token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/token).Content
```

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/pkcs7).Content)
```

IMDSv1

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/pkcs7).Content)
```

3. 인스턴스 메타데이터에서 일반 텍스트 인스턴스 자격 증명 문서를 검색하여 바이트 배열로 변환한 다음 \$Document라는 변수에 추가합니다. 인스턴스에 사용되는 IMDS 버전에 따라 다음 명령 중 하나를 사용합니다.

IMDSv2

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

IMDSv1

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

4. certificate.pem이라는 이름의 새 파일을 생성하고 해당 리전에 따라 다음 AWS DSA 퍼블릭 인증서 중 하나를 추가합니다.

Important

해당 리전의 AWS DSA 퍼블릭 certificate.pem이 아래에 나열되어 있지 않은 경우 [AWS Support](#)에 문의하십시오.

Other AWS Regions

다음 AWS 퍼블릭 인증서는 홍콩, 바레인, 중국 및 GovCloud를 제외한 모든 AWS 리전에 적용됩니다.

```
-----BEGIN CERTIFICATE-----  
MIIC7TCCAg0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTA1VTMRkw  
FWYDVQQIExBYXXN0aW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD  
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExmQzAeFw0xMjAxMDUxMjU2MTJaFw0z  
ODAxMDUxMjU2MTJaMFwxCzAJBgcNVBAYTA1VTMRkwFwYDVQQIExBYXXN0aW5ndG9u  
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdhbWF6b24gV2ViIFNl  
cnZpY2VzIExmQzCCAbcwggEsbgcqhkjOOAQBMIBHwKBgQCjkvcS2bb1VQ4yt/5e  
ih5O06kK/n1Lz1lr7D8ZwtQP8fOEpp5E2ng+D6UD1Z1gYipr58Kj3nssSNpI6bx3  
VyIQzK7wLcInd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUp3QbyYXJdmVMegN6P  
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j  
k+tKqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJu9Qxw3rAotXau8Qe+MBcJ1/U  
hy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCchFnIzbdlx1E9rpUp7bnF  
lRa2v1ntMX3caRVDbtPEWmdxSCYSYFDk4mzrOLBA4GEAAKBgEbmeve5f8LIE/Gf  
MNmP9CM5eovQOGx5ho8WqD+aTeb+s+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
```

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
인스턴스 메타데이터 및 사용자 데이터

```
MXrs3IgIb6+hUIB+S8dz8/mm00bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHiziqQYMAkGBYqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

Hong Kong Region

총 콩 리전을 위한 AWS 퍼블릭 인증서는 다음과 같습니다.

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq4CCQCO7MJe5Y3VLjAJBgcqhkjOOAQDMFwxCzAJBgnVBAYTA1VTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnPzY2VzIExMQzAeFw0xOTAYMDMwMjIxMjFaFw00
NTAyMDMwMjIxMjFaMFwxCzAJBgnVBAYTA1VTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIExMQzCCAbgwggEsBgcqhkjOOAQBMIIIBhWBgQDvQ9RzVvf4MAwGbqfx
b1CvCoVb99570kLGn/04CowHXJ+vTBR7eyIa6AoXltsQXBOMrJswToFKkxT4gbuw
jK7s9Q0Q4CmTRwC Eg02RxtZSVjOhsUQmH+yf7Ht4OVL97LwnNfGsX2cwjcRWHYgI
71vnubNBzLQhdSEwMNq0Bk76PwIVAMan6XIEEPnw4e6u/RNnWBGKd9FAoGBAOOG
eSNnxpW4QFu4pI1Aykm6EnTzKKHT87gdXkAkfoc5AfOxxhnE2HezzHpp9Ap2tMV5
8bWNv0PHvoKCQgwfm+OUB1AxC/3vqoVkJL2mG1KgUH9+hrtptkW03REnKe7I50
x9qDimJpOihL4I0dYvy9xUoz+DzFAW8+y1LWVypA4GFAAKBqQDbnBAKSxWr9QHY
6Dt+EFdGz61AZLedeBKpaP53Z1DT034J0C55YbJTwBTFGqPtOLxnUVd1GiD6GbmC
80f3jvogPR1mSmGsydbNbZnbUEVWrRhe+y5zJ3g9qs/DWmDW0deEFvkhWVnLJkFJ
9pdOu/ibRPH11E2nz6pK7GbQ0tLyHTAJBgcqhkjOOAQDAzAAAMC0CFQCoJlwGtJQC
cLoM4p/jtVF0j26xbgiUUS4pDKyHaG/eaygLttFpFJqzWHc=
-----END CERTIFICATE-----
```

Bahrain Region

바레인 리전을 위한 AWS 퍼블릭 인증서는 다음과 같습니다.

```
-----BEGIN CERTIFICATE-----
MIIC7jCCAq4CCQCVWIGSmP8RhTAJBgcqhkjOOAQDMFwxCzAJBgnVBAYTA1VTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnPzY2VzIExMQzAeFw0xOTAYMDUxMzA2MjFaFw00
NTAyMDUxMzA2MjFaMFwxCzAJBgnVBAYTA1VTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIExMQzCCAbgwggEsBgcqhkjOOAQBMIIIBhWBgQDcwojQfgWdV1Qli0OB
8n6cLZ38VE7ZmrjZ9OQV//Gst6S1h7euhC23YppKXi1zovefSDwFU54zi3/oJ++q
PH1P1WGL8IZ34BUGRTtG4TVolvp0smjkMvyRu5hIdKtzjV93Ccx15gVgyk+o1IEG
fZ2Kbw/Dd8JfoPS7KaSCmJKXQIVAIzbIaDFRGA2qcMkW2HWASyND17bAoGBAnTz
IdhfMq+l2I51ofY2oj3HI21Kj3LtZrWEg3W+/4rvhL31TmONne1rl9yGujrjQwy5
Zp9V4A/w9w2010Lx4K6hj34Efey/aQnZwNdhv/FQP7Azofju+Y16L13OOHqrL0z
Q+9cF7zEosekEnBQx3v6psNknKgD3Shgx+GO/LpCA4GFAAKBqOCVS7m77nuNALZ8
wvUqcooxXMPKxJF154NxAsAul9KP9KN4svm003Zrb7t2F0tXRM8zU3TqMpryqlo5
mpMPsZDg6RXo9BF7HnDoZ6PJTamkFA6md+NyTJWJKvXC7iJ8fGDBJqTciUHuCKr
12AztQ8bfWsrTgTzPE3p6U5ckcgV1TAJBgcqhkjOOAQDAy8AMCwCFB2NZGwm5ED1
86ayV3c1PEDukgQIAhQow38rQkN/VwHVeSW9DqEshXHjuQ==
-----END CERTIFICATE-----
```

GovCloud Regions

AWS GovCloud 리전을 위한 AWS 퍼블릭 인증서는 다음과 같습니다.

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgnVBAYTA1VTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnPzY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgnVBAYTA1VTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXR1MRAwDgYDVQQHEwdTZWF0dGx1MSAwHgYDVQQKExdBbWF6b24gV2ViIFN1
cnZpY2VzIExMQzCCAbgwggEsBgcqhkjOOAQBMIIIBhWBgQcjkvcS2bb1VQ4yt/5e
-----END CERTIFICATE-----
```

```

ih5006kK/n1Lz1lr7D8ZwtQP8fOEpp5E2ng+D6UD1Z1gYipr58Kj3nssSNpI6bX3
Vy1QzK7wLcInd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nY14hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBCJ1/U
hhy1KHVpCG19fueQ2s6IL0Ca0/buycU1CiYQk40KNHCChfNiZbdlx1E9rpUp7bnF
1Ra2v1ntMX3caRVDbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIB6+hUIB+S8dz8/mm0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIZiqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6R0k0k9K
-----END CERTIFICATE-----

```

China Regions

중국(베이징) 및 중국(닝샤) 리전을 위한 AWS 퍼블릭 인증서는 다음과 같습니다.

```

-----BEGIN CERTIFICATE-----
MIIDNjCCAh4CCQD3yZ1w1AVkTzANBgkqhkiG9w0BAQsFADBcMQswCQYDVQQGEwJV
UzEZMBcGA1UECBMQUEV2FzaGluZ3Rvb1BTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEg
MB4GA1UEChMXQW1hem9uIFd1YiBTZXJ2aWNlcyclBMTEMwIBcNMTUwNTEzMDk1OTE1
WhgPMjE5NDEwMTYwOTU5MTVaMFwxCzAJBgNVBAYTALVTMRkwFwYDVQQIExBXYXNo
aW5ndG9uIFN0YXR1MRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24g
V2ViIFNlcnPzY2VzIEzMqzCCASiWdQYJKoZIhvCNQEBBQADggEPADCCAQoCggEB
AMWk9vyppSmDU3AxZ2Cy2bvKeF3F1UqNpMuveriizi+NTs8tQqtNloaQcqhto/1
gsw9+QSnEJeYWnmivJWOBdn9CyDpN7cpHVmeGgNJL2fvImWyWe2f2Kq/BL917N7C
P2ZT52/sH9orlck1n2z08xPi7MTtgPHQwu30xsGQsAdWucdxjHGtdchulpoluJ31
jSTAPKZ3p1/sxPXBBAgBMatPHhRBqhwHO/Twm4J3GmTLWN7oVDDs4W3bPKQfnw3r
vtBj/SM4/IgQ3xJslFc190TzbQbgxiI88R/gWTbs7GsyT2PzstU30yLdjhKfdZKz
/aIzraHvoDTWFa0dy0+OoECAwEAATANBgkqhkiG9w0BAQsFAAOCAQEAdszN2+0E
V1BfR3DPWJHWrf1b7z1+1X/ZseW2hYE5r6YxrLv+1VPf/L5I6kB7GETghZUqteY7
zAeoLrVu/70ynRyfQetJVGichaaxLNm3lcr6kcxOowb+WQO84cwrB3keykH4gRX
KHB2rlWSxta+2panSEO1JX2q5jhcFP90rDOTZjlpYv57N/Z9iQ+dvQPJnChdq3BK
5pZlnIDnVVxqRike7BFy8tKyPj7HzoPEF5mh9Ffn1YoSVu+61lMVv/qRjnyKfS9
c96nE98sYFj0ZBzXw8Sq4Gh8FiVmFHbOp1peGC19idOUqxPxWsasWxQX00azYsP
9RyWLHKxH1dMuA==
-----END CERTIFICATE-----

```

- 인증서 파일에서 인증서를 추출하여 \$Store라는 변수에 저장합니다.

```
PS C:\> $Store =
[Security.Cryptography.X509Certificates.X509Certificate2Collection]::new([Security.Cryptography.X509Certificates.X509Certificate]::FromFile("certificate.pem"))
```

- 서명을 확인합니다.

```
PS C:\> $SignatureDocument = [Security.Cryptography.Pkcs.SignedCms]::new()
```

```
PS C:\> $SignatureDocument.Decode($Signature)
```

```
PS C:\> $SignatureDocument.CheckSignature($Store, $true)
```

서명이 유효하면 명령에서 출력이 반환되지 않으며, 서명을 확인할 수 없으면 명령에서 Exception calling "CheckSignature" with "2" argument(s): "Cannot find the original signer"가 반환됩니다. 서명을 확인할 수 없는 경우 AWS Support에 문의하십시오.

- 인스턴스 자격 증명 문서의 내용을 확인합니다.

```
PS C:
\> [Linq.Enumerable]::SequenceEqual($SignatureDocument.ContentInfo.Content, $Document)
```

인스턴스 자격 증명 문서의 내용이 유효하면 명령에서 `True`가 반환됩니다. 인스턴스 자격 증명 문서를 확인할 수 없는 경우 AWS Support에 문의하십시오.

base64 인코딩 서명을 사용하여 인스턴스 자격 증명 문서 확인

이 주제에서는 base64 인코딩 서명 및 AWS RSA 퍼블릭 인증서를 사용하여 인스턴스 자격 증명 문서를 확인하는 방법을 설명합니다.

Important

base64 인코딩 서명을 사용하여 인스턴스 자격 증명 문서를 확인하려면 [AWS Support](#)에서 해당 AWS RSA 퍼블릭 인증서를 요청해야 합니다.

PKCS7 서명 및 AWS DSA 퍼블릭 인증서를 사용하여 인스턴스 자격 증명 문서를 확인하려면

1. 인스턴스에 연결합니다.
2. 인스턴스 메타데이터에서 base64 인코딩 서명을 검색하여 바이트 배열로 변환한 다음 `$Signature`라는 변수에 추가합니다. 인스턴스에 사용되는 IMDS 버전에 따라 다음 명령 중 하나를 사용합니다.

IMDSv2

```
PS C:\> $Token = (Invoke-WebRequest -Method Put -Headers @{ 'X-aws-ec2-metadata-token-ttl-seconds' = '21600' } http://169.254.169.254/latest/api/token).Content
```

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers @{ 'X-aws-ec2-metadata-token' = $Token } http://169.254.169.254/latest/dynamic/instance-identity/signature).Content)
```

IMDSv1

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/signature).Content)
```

3. 인스턴스 메타데이터에서 일반 텍스트 인스턴스 자격 증명 문서를 검색하여 바이트 배열로 변환한 다음 `$Document`라는 변수에 추가합니다. 인스턴스에 사용되는 IMDS 버전에 따라 다음 명령 중 하나를 사용합니다.

IMDSv2

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers @{ 'X-aws-ec2-metadata-token' = $Token } http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

IMDSv1

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

4. AWS Support에서 받은 AWS RSA 퍼블릭 인증서를 `certificate.pem`이라는 파일에 추가합니다.
5. 인스턴스 자격 증명 문서를 확인합니다.

```
PS C:\> [Security.Cryptography.X509Certificates.X509Certificate2]::new((Resolve-Path certificate.pem)).PublicKey.Key.VerifyData($Document, 'SHA256', $Signature)
```

서명이 유효하면 명령에서 `true`가 반환됩니다. 서명을 확인할 수 없는 경우 AWS Support에 문의하십시오.

RSA-2048 서명을 사용하여 인스턴스 자격 증명 문서 확인

이 주제에서는 RSA-2048 서명 및 AWS RSA-2048 퍼블릭 인증서를 사용하여 인스턴스 자격 증명 문서를 확인하는 방법을 설명합니다.

Important

RSA-2048 서명을 사용하여 인스턴스 자격 증명 문서를 확인하려면 [AWS Support](#)에서 AWS RSA-2048 퍼블릭 인증서를 요청해야 합니다.

사전 조건

이 절차를 수행하려면 `System.Security` Microsoft .NET Core 클래스가 필요합니다. PowerShell 세션에 클래스를 추가하려면 다음 명령을 실행합니다.

```
PS C:\> Add-Type -AssemblyName System.Security
```

Note

이 명령은 현재 PowerShell 세션에만 클래스를 추가합니다. 새 세션을 시작하는 경우 명령을 다시 실행해야 합니다.

RSA-2048 서명 및 AWS RSA-2048 퍼블릭 인증서를 사용하여 인스턴스 자격 증명 문서를 확인하려면

1. 인스턴스에 연결합니다.
2. 인스턴스 메타데이터에서 RSA-2048 서명을 검색하여 바이트 배열로 변환한 다음 `$Signature`라는 변수에 추가합니다. 인스턴스에 사용되는 IMDS 버전에 따라 다음 명령 중 하나를 사용합니다.

IMDSv2

```
PS C:\> $Token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/token).Content
```

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/rsa2048).Content)
```

IMDSv1

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/rsa2048).Content)
```

3. 인스턴스 메타데이터에서 일반 텍스트 인스턴스 자격 증명 문서를 검색하여 바이트 배열로 변환한 다음 `$Document`라는 변수에 추가합니다. 인스턴스에 사용되는 IMDS 버전에 따라 다음 명령 중 하나를 사용합니다.

IMDSv2

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

IMDSv1

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

4. AWS Support에서 받은 AWS RSA-2048 퍼블릭 인증서를 certificate.pem이라는 새 파일에 추가합니다.
5. 인증서 파일에서 인증서를 추출하여 \$Store라는 변수에 저장합니다.

```
PS C:\> $Store = [Security.Cryptography.X509Certificates.X509Certificate2Collection]::new([Security.Cryptography.X509Certificates.X509Certificate2]::FromFile("certificate.pem"))
```

6. 서명을 확인합니다.

```
PS C:\> $SignatureDocument = [Security.Cryptography.Pkcs.SignedCms]::new()
```

```
PS C:\> $SignatureDocument.Decode($Signature)
```

```
PS C:\> $SignatureDocument.CheckSignature($Store, $true)
```

서명이 유효하면 명령에서 출력이 반환되지 않으며, 서명을 확인할 수 없으면 명령에서 Exception calling "CheckSignature" with "2" argument(s): "Cannot find the original signer가 반환됩니다. 서명을 확인할 수 없는 경우 AWS Support에 문의하십시오.

7. 인스턴스 자격 증명 문서의 내용을 확인합니다.

```
PS C:\> [Linq.Enumerable]::SequenceEqual($SignatureDocument.ContentInfo.Content, $Document)
```

인스턴스 자격 증명 문서의 내용이 유효하면 명령에서 True가 반환됩니다. 인스턴스 자격 증명 문서를 확인할 수 없는 경우 AWS Support에 문의하십시오.

EC2의 SQL Server 클러스터링에 대한 모범 사례 및 권장 사항

SQL Always On 클러스터링은 공유 스토리지에 대한 요구 사항이 없는 고가용성을 제공합니다. 이 주제의 사례 목록은 [Prerequisites, Restrictions, and Recommendations for Always On availability groups](#)에 나열된 사전 조건 외에도 AWS의 SQL Server Always On 클러스터를 운영할 때 최상의 결과를 얻는 데 도움이 될 수 있습니다. 이 주제에 나열된 사례는 로그를 수집하는 방법을 제공합니다.

Note

노드가 서로 다른 가용 영역 또는 동일한 영역 내의 다른 서브넷에 배포되는 경우 다중 서브넷 클러스터로 처리해야 합니다. 모범 사례를 적용하고 실현 가능한 장애 시나리오를 해결할 때 이 점을 기억하십시오.

목차

- IP 주소 할당 (p. 606)
- 클러스터 속성 (p. 606)
- 다중 사이트 클러스터에서 클러스터 퀄럼 투표 및 50/50 분할 (p. 607)
- DNS 등록 (p. 607)

- [Elastic Network Adapter\(ENA\) \(p. 607\)](#)
- [다중 사이트 클러스터 및 EC2 인스턴스 배치 \(p. 608\)](#)
- [인스턴스 유형 선택 \(p. 608\)](#)
- [인스턴스에 탄력적 네트워크 인터페이스 및 IP 할당 \(p. 608\)](#)
- [하트비트 네트워크 \(p. 608\)](#)
- [OS에서 네트워크 어댑터 구성 \(p. 608\)](#)
- [IPv6 \(p. 608\)](#)
- [SQL 가용성 그룹 리스너에 대한 호스트 레코드 TTL \(p. 608\)](#)
- [로깅 \(p. 609\)](#)
- [TCP를 통한 NetBIOS \(p. 609\)](#)
- [NetFT Virtual Adapter \(p. 609\)](#)
- [잠재적인 소유자\(Possible Owner\) 설정 \(p. 609\)](#)
- [장애 조치 임계값 튜닝 \(p. 610\)](#)
- [감시 중요성 및 동적 쿼럼 아키텍처 \(p. 611\)](#)
- [문제 해결 \(p. 611\)](#)

IP 주소 할당

각 클러스터 노드에는 서브넷의 세 가지 프라이빗 IP 주소(기본 IP 주소, 클러스터 IP 주소 및 가용성 그룹 IP 주소)가 포함된 탄력적 네트워크 인터페이스가 하나씩 할당되어야 합니다. 운영 체제(OS)에는 DHCP용으로 구성된 NIC가 있어야 합니다. 클러스터 IP 및 가용성 그룹의 IP 주소는 Failover Cluster Manager에서 가상으로 처리되므로 고정 IP 주소로 설정하면 안 됩니다. eth0의 기본 IP만 사용하도록 구성된 경우 NIC는 고정 IP에 대해 구성될 수 있습니다. 다른 IP가 NIC에 할당되면 장애 조치 이벤트 중에 인스턴스에 대한 네트워크 드롭이 발생할 수 있습니다.

IP가 잘못 할당되었거나 장애 조치 이벤트 또는 네트워크 장애가 발생하여 네트워크 드롭이 발생하는 경우 일반적으로 장애 발생 시 다음 이벤트 로그 항목이 표시됩니다.

```
Isatap interface isatap.{9468661C-0AEB-41BD-BB8C-1F85981D5482} is no longer active.
```

```
Isatap interface isatap.{9468661C-0AEB-41BD-BB8C-1F85981D5482} with address fe80::5efe:169.254.1.105 has been brought up.
```

이러한 메시지는 네트워크 문제를 설명하는 것처럼 보이므로 정전이나 장애의 원인을 네트워크 오류로 착각하기 쉽습니다. 하지만 이러한 오류는 장애의 원인보다는 증상에 대해 설명합니다. ISATAP는 IPv4를 통해 IPv6를 사용하는 터널링 기술입니다. IPv4 연결이 실패하면 ISATAP 어댑터도 실패합니다. 네트워크 문제가 해결되면 이러한 항목은 더 이상 이벤트 로그에 표시되지 않습니다. 또는 다음 명령을 사용하여 ISATAP를 안전하게 비활성화하여 네트워크 오류를 제거할 수 있습니다.

```
netsh int ipv6 isatap set state disabled
```

이 명령을 실행하면 어댑터가 Device Manager에서 제거됩니다. 이 명령은 모든 노드에서 실행해야 합니다. 이는 클러스터의 기능이 작동하는 데 영향을 주지 않습니다. 대신에 명령이 실행될 때 ISATAP가 더 이상 사용되지 않습니다. 그러나 이 명령은 ISATAP를 활용하는 다른 애플리케이션에 알려지지 않은 영향을 줄 수 있으므로 테스트해야 합니다.

클러스터 속성

전체 클러스터 구성을 보려면 다음 PowerShell 명령을 실행합니다.

```
Get-Cluster | Format-List -Property *
```

다중 사이트 클러스터에서 클러스터 쿼럼 투표 및 50/50 분할

클러스터 쿼럼의 작동 방식 및 장애 발생 시 예상되는 사항에 대한 자세한 내용은 [Understanding Cluster and Pool Quorum](#)을 참조하십시오.

DNS 등록

Windows Server 2012에서 장애 조치 클러스터링은 기본적으로 각 DNS 노드를 클러스터 이름으로 등록하려고 시도합니다. 이는 SQL 대상이 다중 사이트용으로 구성되어 있는 것을 인식하는 애플리케이션에서 허용됩니다. 하지만 클라이언트가 이러한 방식으로 구성되지 않는 경우 각 개별 노드에 대한 연결 시도 및 비활성 노드의 장애로 인해 시간 초과, 지연 및 애플리케이션 오류가 발생할 수 있습니다. 이러한 문제를 방지하려면 클러스터 리소스 파라미터 RegisterAllProvidersIP를 0으로 변경해야 합니다. 자세한 내용은 [RegisterAllProvidersIP Setting](#) 및 [Multi-subnet Clustered SQL + RegisterAllProvidersIP + SharePoint 2013](#)을 참조하십시오.

RegisterAllProvidersIP은 다음 PowerShell 스크립트로 수정할 수 있습니다.

```
Import-Module FailoverClusters
$cluster = (Get-ClusterResource | where {($_.ResourceType -eq "Network Name") -and
($_.OwnerGroup -ne "Cluster Group")}).Name
Get-ClusterResource $cluster | Set-ClusterParameter RegisterAllProvidersIP 0
Get-ClusterResource $cluster | Set-ClusterParameter HostRecordTTL 300
Stop-ClusterResource $cluster
Start-ClusterResource $cluster
```

클러스터 리소스 파라미터를 0으로 설정하는 것 외에도 클러스터에 클러스터 이름에 대한 DNS 항목을 수정할 수 있는 권한이 있는지 확인해야 합니다.

1. 도메인의 도메인 컨트롤러(DC)에 로그인하거나 도메인의 정방향 조회 영역을 호스팅하는 서버에 로그인합니다.
2. DNS 관리 콘솔을 시작하고 클러스터에 대한 A 레코드를 찾습니다.
3. A 레코드를 마우스 오른쪽 버튼으로 클릭하고 Properties(속성)를 선택합니다.
4. [Security]를 선택합니다.
5. Add(추가)를 선택합니다.
6. Object Types...(객체 유형...)를 선택하고 Computers(컴퓨터)를 선택한 다음, OK(확인)를 선택합니다.
7. 클러스터 리소스 객체의 이름을 입력하고 Check name(이름 확인) 및 OK if resolve(확인 가능한 경우 확인)를 선택합니다.
8. Full Control(전체 제어) 확인란을 선택합니다.
9. 확인을 선택합니다.

Elastic Network Adapter(ENA)

AWS는 ENA 드라이버 버전 1.2.3에서 실행되는 일부 클러스터링 워크로드 문제를 확인했습니다. 버전 1.5.0 이상으로 업그레이드하고 OS의 NIC 설정을 조정하는 것을 권장합니다. 최신 버전의 경우 [Amazon ENA 드라이버 버전](#)을 참조하십시오. 모든 시스템에 적용되는 첫 번째 설정은 수신 버퍼를 높입니다. 이는 다음 예제 PowerShell 명령을 사용하여 수행될 수 있습니다.

```
Set-NetAdapterAdvancedProperty -Name (Get-NetAdapter | Where-Object
{$_.InterfaceDescription -like '*Elastic*'}).Name -DisplayName "Receive Buffers" -
DisplayValue 8192
```

16개보다 많은 vCPU가 있는 인스턴스의 경우 CPU 0에서 RSS가 실행되지 않도록 하는 것이 좋습니다.

다음 명령을 실행합니다.

```
Set-NetAdapterRss -name (Get-NetAdapter | Where-Object {$_.InterfaceDescription -like '*Elastic*'}).Name -BaseProcessorGroup 0 -BaseProcessorNumber 1
```

다중 사이트 클러스터 및 EC2 인스턴스 배치

각 클러스터는 [다중 사이트 클러스터](#)로 간주됩니다. EC2 서비스는 가상으로 IP 주소를 공유하지 않습니다. 각 노드는 고유한 [서브넷](#)에 있어야 합니다. 필수는 아니지만 각 노드가 고유한 [가용 영역 \(p. 5\)](#)에도 있는 것 이 좋습니다.

인스턴스 유형 선택

Windows Server 장애 조치 클러스터링에 권장되는 인스턴스 유형은 워크로드에 따라 다릅니다. 프로덕션 작업 부하의 경우 [EBS 최적화](#) 및 [향상된 네트워킹](#)을 지원하는 인스턴스가 권장됩니다.

인스턴스에 탄력적 네트워크 인터페이스 및 IP 할당

EC2 클러스터의 각 노드에는 탄력적 네트워크 인터페이스가 하나만 연결되어 있어야 합니다. 네트워크 인터페이스에는 최소 두 개의 프라이빗 IP 주소가 할당되어야 합니다. 하지만 SQL Always On과 같이 가용성 그룹을 사용하는 워크로드의 경우 각 가용성 그룹에 대해 추가 IP 주소를 포함해야 합니다. 기본 IP 주소는 서버 액세스 및 관리에 사용되고 보조 IP 주소는 클러스터 IP 주소로 사용되며 각 추가 IP 주소는 필요에 따라 가용성 그룹에 할당됩니다.

하트비트 네트워크

일부 Microsoft 문서에서는 전용 [하트비트 네트워크](#)를 사용하도록 권장합니다. 그러나 이 권장 사항은 EC2에는 적용되지 않습니다. EC2의 경우 하트비트 네트워크에 두 번째 탄력적 네트워크 인터페이스를 할당하고 사용할 수 있지만 동일한 인프라를 사용하고 기본 네트워크 인터페이스와 대역폭을 공유합니다. 따라서 인프라 내의 트래픽을 우선순위로 지정할 수 없으며 전용 네트워크 인터페이스의 이점을 활용할 수 없습니다.

OS에서 네트워크 어댑터 구성

DHCP 옵션 세트에서 검색되는 DNS 서버가 노드에 대해 서로 분석하도록 허용하는 경우 OS의 NIC가 DHCP를 계속 사용할 수 있습니다. NIC를 정적으로 구성하도록 설정할 수 있습니다. 완료되면 탄력적 네트워크 인터페이스의 기본 IP 주소만 수동으로 구성합니다. 장애 조치 클러스터링은 필요에 따라 추가 IP 주소를 관리하고 할당합니다.

모든 인스턴스 유형에 대해 네트워크 어댑터의 최대 전송 단위(MTU)를 9001로 높여서 [점보 프레임](#)을 지원할 수 있습니다. 이 구성은 점보 프레임이 지원되는 위치에서 패킷의 조각화를 줄입니다. 다음 예에서는 PowerShell을 사용하여 Elastic Network Adapter에 대한 점보 프레임을 구성하는 방법을 보여 줍니다.

```
Get-NetAdapter | Set-NetAdapterAdvancedProperty -DisplayName "MTU" -DisplayValue 9001
```

IPv6

Microsoft는 Windows 클러스터에서 IPv6을 비활성화하는 것을 권장하지 않습니다. 장애 조치 클러스터링은 IPv4 전용 환경에서 작동하는 한편, Microsoft는 IPv6이 활성화된 클러스터를 테스트합니다. 자세한 내용은 [Failover Clustering and IPv6 in Windows Server 2012 R2](#)를 참조하십시오.

SQL 가용성 그룹 리스너에 대한 호스트 레코드 TTL

호스트 레코드 TTL을 기본 20분(1200초) 대신 300초로 설정합니다. 레거시 클라이언트 비교성의 경우 SQL 가용성 그룹 리스너에 대해 RegisterAllProvidersIP를 0으로 설정합니다. 모든 환경에서 필수는 아

닙니다. 일부 레거시 클라이언트 애플리케이션이 연결 문자열에 MultiSubnetFailover를 사용할 수 없기 때문에 이러한 설정이 중요합니다. 자세한 내용은 [HostRecordTTL 설정](#)을 참조하십시오. 이 설정을 변경하면 클러스터 리소스를 다시 시작해야 합니다. 클러스터 리소스가 재시작되면 리스너에 대한 클러스터 그룹이 중지되므로 클러스터 그룹을 시작해야 합니다. 클러스터 그룹을 시작하지 않으면 가용성 그룹이 오프라인 RESOLVING 상태로 유지됩니다. 다음은 TTL 및 RegisterAllProvidersIP 설정을 변경하기 위한 PowerShell 스크립트의 예입니다.

```
Get-ClusterResource yourListenerName | Set-ClusterParameter RegisterAllProvidersIP 0
```

```
Get-ClusterResource yourListenerName | Set-ClusterParameter HostRecordTTL 300
```

```
Stop-ClusterResource yourListenerName
```

```
Start-ClusterResource yourListenerName
```

```
Start-ClusterGroup yourListenerGroupName
```

로깅

클러스터 로그의 기본 로깅 수준은 3입니다. 로그 정보의 세부 수준을 높이려면 로깅 수준을 5로 설정합니다. PowerShell cmdlet에 대한 자세한 내용은 [Set-ClusterLog](#)를 참조하십시오.

```
Set-ClusterLog -Level 5
```

TCP를 통한 NetBIOS

Windows Server 2012 R2에서 TCP를 통한 NetBIOS를 비활성화하여 장애 조치 프로세스의 속도를 높일 수 있습니다. 이 기능은 Windows Server 2016에서 제거되었습니다. 사용자 환경에서 이전 운영 체제를 사용하는 경우 이 절차를 테스트해야 합니다. 자세한 내용은 [Speeding Up Failover Tips-n-Tricks](#)를 참조하십시오. 다음은 TCP를 통한 NetBIOS를 비활성화하는 PowerShell 명령의 예입니다.

```
Get-ClusterResource "Cluster IP Address" | Set-ClusterParameter EnableNetBIOS 0
```

NetFT Virtual Adapter

2016 이전의 Windows Server 버전 및 비 Hyper-V 워크로드의 경우 Microsoft는 OS의 어댑터에서 NetFT Virtual Adapter 성능 필터를 활성화하도록 권장합니다. NetFT Virtual Adapter를 활성화하면 내부 클러스터 트래픽이 NetFT Virtual Adapter로 직접 라우팅됩니다. 자세한 내용은 [NetFT Virtual Adapter Performance Filter](#)를 참조하십시오. NIC 속성의 확인란을 선택하거나 다음 PowerShell 명령을 사용하여 NetFT Virtual Adapter를 활성화할 수 있습니다.

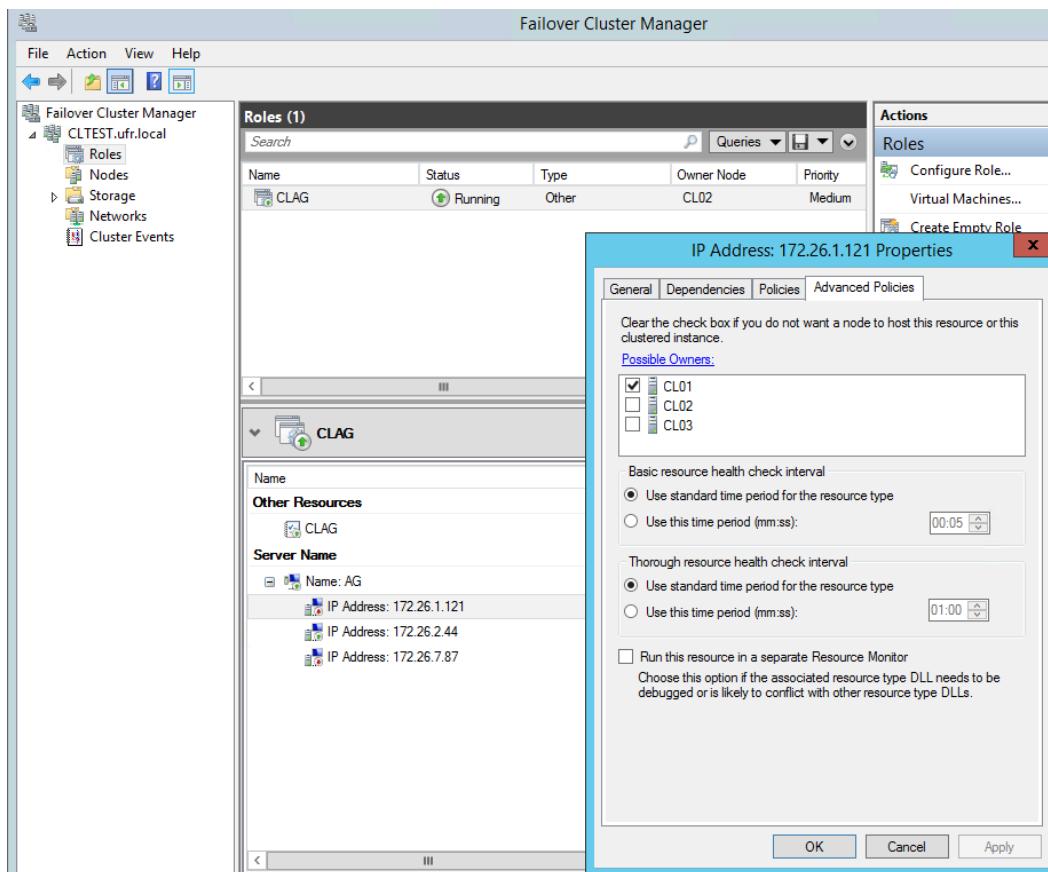
```
Get-NetAdapter | Set-NetAdapterBinding -ComponentID ms_netftfl -Enable $true
```

잠재적인 소유자(Possible Owner) 설정

Failover Cluster Manager는 클러스터 코어 리소스 및 가용성 그룹 리소스에 지정된 각 IP 주소가 해당 IP가 속한 노드에서만 온라인 상태가 될 수 있도록 구성될 수 있습니다. Failover Cluster Manager가 이렇게 구성되지 않고 장애가 발생하는 경우 클러스터가 주소를 인식하지 못하는 노드에서 IP를 가져오려고 시도할 때

장애 조치에 약간의 지연이 발생합니다. 자세한 내용은 [SQL Server Manages Preferred and Possible Owner Properties for AlwaysOn Availability Group/Role](#)을 참조하십시오.

클러스터의 각 리소스에는 잠재적인 소유자에 대한 설정이 있습니다. 이 설정은 클러스터에게 리소스를 '온라인' 상태로 허용할 수 있는 노드를 알려 줍니다. 각 노드는 VPC의 고유한 서브넷에서 실행됩니다. EC2는 인스턴스 간에 IP를 공유할 수 없기 때문에 클러스터의 IP 리소스를 특정 노드에서만 온라인 상태로 만들 수 있습니다. 기본적으로 클러스터에 리소스로 추가된 각 IP 주소에는 모든 노드가 잠재적인 소유자로 나열됩니다. 이로 인해 장애가 발생하지는 않습니다. 하지만 예상되는 오류 및 예상치 못한 장애가 발생하면 로그에서 IP 충돌과 IP의 온라인 상태 실패에 대한 오류가 표시될 수 있습니다. 이러한 오류는 무시할 수 있습니다. 잠재적인 소유자 속성을 설정하면 서비스가 다른 노드로 이동하는 동안 이러한 오류를 완전히 제거할 수 있으며 가동 중지 시간도 방지할 수 있습니다.



장애 조치 임계값 튜닝

Server 2012 R2에서 장애 조치 하트비트 네트워크 임계값은 높은 값으로 기본 설정됩니다. 자세한 내용은 [Tuning Failover Cluster Network Thresholds](#)를 참조하십시오. 잠재적으로 신뢰할 수 없는 이 구성(클러스터 간 거리가 서로 먼 경우)은 하트비트 수의 증가와 함께 Server 2016에서 해결되었습니다. 매우 짧은 임시 네트워크 문제로 인해 클러스터가 장애 조치되는 것으로 확인되었습니다. 하트비트 네트워크는 일반적으로 TCP보다 신뢰성이 낮고 불완전한 통신을 하는 경향이 있는 UDP 3343으로 유지됩니다. AWS 가용 영역 간 연결의 지연 시간은 짧지만, 리소스를 분리하는 많은 "흡"을 통해 여전히 지리적으로 구분되어 있습니다. 가용 영역 내에서 고객이 배치 그룹 또는 전용 호스트를 사용하지 않는 경우 클러스터 간에 거리가 있을 수 있습니다. 결과적으로 UDP 기반의 하트비트 장애 가능성은 TCP 기반 하트비트보다 높습니다.

클러스터가 장애 조치를 수행하는 유일한 시간은 전송 중에 손실된 일부 UDP 패킷과 달리 하드 장애 조치를 경험하는 서비스 또는 노드와 같은 합법적인 종단이 있는 경우입니다. 합법적인 종단을 확인하려면 [Tuning Failover Cluster Network Thresholds](#)에 나열된 Server 2016의 설정과 일치하거나 조과하도록 임계값을 조정하는 것이 좋습니다. 다음 PowerShell 명령을 사용하여 설정을 변경할 수 있습니다.

```
(get-cluster).SameSubnetThreshold = 10
```

```
(get-cluster).CrossSubnetThreshold = 20
```

이러한 값을 설정할 때 예기치 않은 장애 조치가 크게 줄어들어야 합니다. 하트비트 사이의 지연을 높여 이러한 설정을 세밀하게 튜닝할 수 있습니다. 하지만 더 높은 임계값으로 하트비트를 더 자주 전송하는 것이 좋습니다. 이러한 임계값을 더 높게 설정하면 장애 조치 전에 더욱 긴 시간의 지연이 있는 상태로 하드 장애 조치 시나리오에서만 장애 조치가 발생하게 됩니다. 애플리케이션에 대해 어느 정도의 가동 중지가 허용될 수 있는지 결정해야 합니다.

SameSubnetThreshold 또는 CrossSubnetThreshold를 높인 후에 RouteHistoryLength를 높여 두 값 중 더 높은 값을 두 배로 늘리는 것이 좋습니다. 이렇게 하면 문제 해결을 위한 충분한 로깅이 보장됩니다. 다음 PowerShell 명령을 사용하여 RouteHistoryLength를 설정할 수 있습니다.

```
(Get-Cluster).RouteHistoryLength = 20
```

감시 중요성 및 동적 쿼럼 아키텍처

디스크 감시와 파일 공유 감시 간에는 차이가 있습니다. 디스크 감시는 파일 공유 감시가 수행되지 않는 동안 클러스터 데이터베이스 백업을 유지합니다. 둘 다 [클러스터에 투표 \(p. 607\)](#)를 추가합니다. iSCSI 기반 스토리지를 사용하는 경우, 디스크 감시를 사용할 수 있습니다. 감시 옵션에 대한 자세한 내용은 [File Share witness vs Disk witness for local clusters](#)를 참조하십시오.

문제 해결

예기치 않은 장애 조치가 발생하는 경우 먼저 네트워킹, 서비스 및 인프라 문제가 발생하지 않도록 해야 합니다.

1. 노드에 네트워크 관련 문제가 없는지 확인합니다.
2. 드라이버 업데이트를 확인합니다. 인스턴스에서 오래된 드라이버를 사용하고 있는 경우 드라이버를 업데이트해야 합니다. 드라이버를 업데이트하면 현재 설치된 버전에 나타날 수 있는 버그 및 안정성 문제가 해결될 수 있습니다.
3. CPU 및 디스크 I/O와 같이 인스턴스가 응답하지 않을 수 있는 잠재적인 리소스 병목 현상이 없는지 확인합니다. 노드가 요청을 처리할 수 없는 경우 클러스터 서비스에 의해 중지된 것처럼 보일 수 있습니다.

Amazon EC2 Windows 인스턴스를 새 버전의 Windows Server로 업그레이드

인스턴스에서 실행하는 Windows Server의 이전 버전을 새 버전으로 업그레이드하는 방법에는 인 플레이스 (in-place) 업그레이드와 마이그레이션(단계별 업그레이드라고도 함)이 있습니다. 인 플레이스 업그레이드는 운영 체제 파일은 업그레이드하는 반면 개인 설정과 파일에는 영향을 주지 않습니다. 마이그레이션은 설정, 구성 및 데이터를 캡처하고 이를 새로운 Amazon EC2 인스턴스의 최신 운영 체제로 이식합니다.

Microsoft는 전통적으로 Windows Server를 업그레이드하는 대신 새 버전으로 마이그레이션하도록 권장해왔습니다. 마이그레이션은 업그레이드 오류 또는 문제가 줄어들 수 있지만, 새 인스턴스를 프로비저닝하고 애플리케이션을 계획 및 포팅하고 새 인스턴스에서 구성 설정을 조정해야 하므로 현재 위치 업그레이드보다 시간 더 걸릴 수 있습니다. 현재 위치 업그레이드가 더 빠를 수 있지만 소프트웨어 호환성 문제 때문에 오류가 발생할 수도 있습니다.

목차

- [인 플레이스 업그레이드 수행 \(p. 612\)](#)
- [자동 업그레이드 수행 \(p. 616\)](#)
- [최신 세대 인스턴스 유형으로 마이그레이션 \(p. 622\)](#)
- [Microsoft SQL Server 데이터베이스용 Windows에서 Linux로의 리플랫포밍 어시스턴트 \(p. 627\)](#)
- [업그레이드 문제 해결 \(p. 634\)](#)

인 플레이스 업그레이드 수행

현재 위치 업그레이드를 수행하기 전에 인스턴스에서 어느 네트워크 드라이버가 실행되고 있는지 확인해야 합니다. PV 네트워크 드라이버는 사용자가 원격 데스크톱을 사용하여 인스턴스에 액세스할 수 있게 해줍니다. Windows Server 2008 R2부터 인스턴스는 AWS PV, intel Network Adapter 또는 Enhanced Networking 드라이버를 사용합니다. Windows Server 2003 및 Windows Server 2008의 인스턴스는 Citrix PV 드라이버를 사용합니다. 자세한 내용은 [Windows 인스턴스의 반가상화\(PV\) 드라이버 \(p. 523\)](#) 단원을 참조하십시오.

자동 업그레이드

AWS 시스템 관리자를 사용하여 자동으로 Windows Server 2008 R2를 Server 2012 R2로 업그레이드하거나 Windows Server 2012 R2의 SQL Server 2008 R2를 SQL Server 2016으로 업그레이드하는 방법에 대한 단계는 [AWS에서 지원 종료된 Microsoft 2008 워크로드를 간편한 업그레이드를 참조하십시오.](#)

인플레이스(In-Place) 업그레이드를 시작하기 전에

다음 작업을 완료하고 인 플레이스 업그레이드를 시작하기 전에 다음과 같은 중요 세부 정보를 기록합니다.

- 업그레이드 요구 사항, 알려진 문제점 및 제약 조건을 파악할 수 있도록 Microsoft 설명서를 읽습니다. 공식적인 업그레이드 지침도 검토해야 합니다.
 - [Windows Server 2008로 업그레이드](#)
 - [Windows Server 2008 R2로 업그레이드](#)
 - [Windows Server 2012용 업그레이드 옵션](#)
 - [Windows Server 2012 R2용 업그레이드 옵션](#)
 - [Windows Server 2016 업그레이드 및 전환 옵션](#)
 - [Windows Server 2019 업그레이드 및 전환 옵션](#)
 - [Windows Server 업그레이드 센터](#)
- 2개 이상의 vCPU와 4GB 이상의 RAM을 이용하는 인스턴스에서 운영 체제 업그레이드를 수행하는 것이 좋습니다. 필요하다면 인스턴스를 같은 유형의 더 큰 인스턴스로 변경하고(예: t2.small에서 t2.large로), 업그레이드를 수행한 다음 원래 크기로 다시 변경할 수도 있습니다. 인스턴스 크기를 유지해야 한다면, [인스턴스 콘솔 스크린샷 \(p. 1188\)](#)을 사용하여 진행 상황을 모니터링할 수 있습니다. 자세한 내용은 [인스턴스 유형 변경 \(p. 194\)](#) 단원을 참조하십시오.
- Windows 인스턴스의 루트 볼륨에 사용 가능한 디스크 공간이 충분히 있는지 확인합니다. Windows 설치 프로세스에서 디스크 공간 부족에 대한 경고 메시지를 표시하지 않을 수도 있습니다. 특정 운영 체제를 업그레이드하는 데 필요한 디스크 공간에 대한 정보는 Microsoft 설명서를 참조하십시오. 볼륨에 공간이 부족한 경우 확장할 수 있습니다. 자세한 내용은 [Amazon EBS 탄력적 볼륨 \(p. 1023\)](#) 단원을 참조하십시오.
- 업그레이드 경로를 확인합니다. 운영 체제를 동일한 아키텍처로 업그레이드해야 합니다. 예를 들어, 32비트 시스템을 32비트 시스템으로 업그레이드해야 합니다. Windows Server 2008 R2 이상은 64비트 전용입니다.
- 안티바이러스와 안티스파이웨어 소프트웨어 및 방화벽을 비활성화합니다. 이러한 유형의 소프트웨어는 업그레이드 프로세스와 충돌할 수 있습니다. 업그레이드를 마친 후에는 안티바이러스와 안티스파이웨어 소프트웨어 및 방화벽을 다시 활성화합니다.
- [최신 세대 인스턴스 유형으로 마이그레이션 \(p. 622\)](#)에 나온 방법에 따라 최신 드라이버로 업데이트합니다.

- 업그레이드 헬퍼 서비스는 Citrix PV 드라이버를 실행하는 인스턴스만 지원합니다. 인스턴스가 Red Hat 드라이버를 실행하는 경우에는 먼저 수동으로 [이러한 드라이버를 업그레이드 \(p. 528\)](#)해야 합니다.

AWS PV, 인텔 Network Adapter 또는 향상된 네트워킹 드라이버를 사용하여 인스턴스 인 플레이스 업그레이드

다음 절차를 통해 AWS PV, intel Network Adapter 또는 Enhanced Networking 네트워크 드라이버를 사용하여 Windows Server 인스턴스를 업그레이드합니다.

인 플레이스 업그레이드를 수행하려면

- 백업 또는 테스트를 위해 업그레이드할 시스템의 AMI를 생성합니다. 그런 다음 사본에서 업그레이드를 수행하여 테스트 환경을 시뮬레이션합니다. 업그레이드가 완료되면 거의 가동 중단 없이 트래픽을 이 인스턴스로 전환할 수 있습니다. 업그레이드에 실패할 경우에는 백업으로 되돌릴 수 있습니다. 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오.
- Windows Server 인스턴스가 최신 네트워크 드라이버를 사용하고 있는지 확인합니다. AWS PV 드라이버 업그레이드에 대한 자세한 내용은 [Windows 인스턴스에서 PV 드라이버 업그레이드 \(p. 528\)](#) 단원을 참조하십시오.
- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 인스턴스를 선택합니다. 인스턴스를 찾습니다. 인스턴스 ID 및 인스턴스의 Availability Zone 을 기록해둡니다. 이 정보는 이 절차의 뒷부분에서 필요합니다.
- Windows Server 2012 또는 2012 R2에서 Windows Server 2016 또는 2019로 업그레이드할 경우 계속하기 전에 인스턴스에서 다음을 수행하십시오.
 - EC2Config 서비스를 제거합니다. 자세한 내용은 [EC2Config 종단, 재시작, 삭제, 설치 제거 \(p. 499\)](#) 단원을 참조하십시오.
 - EC2Launch 서비스를 설치합니다. 자세한 내용은 [최신 버전의 EC2Launch 설치 \(p. 490\)](#) 단원을 참조하십시오.
 - AWS 시스템 관리자 SSM 에이전트 설치 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [SSM 에이전트 작업](#)을 참조하십시오.
- Windows Server 설치 미디어 스냅샷에서 새 볼륨을 생성합니다.
 - 탐색 창에서 스냅샷, 퍼블릭 스냅샷을 선택합니다.
 - 소유자 필터를 추가하고 Amazon 이미지를 선택합니다.
 - 설명 필터를 추가하고 **windows**를 입력합니다. Enter를 누릅니다.
 - 업그레이드하려는 시스템 아키텍처 및 언어 기본 설정과 일치하는 스냅샷을 선택합니다. 예를 들어 Windows Server 2019로 업그레이드하려면 Windows 2019 English Installation Media를 선택합니다.
 - 작업, 볼륨 생성을 선택합니다.
 - 볼륨 생성 대화 상자에서 Windows 인스턴스와 일치하는 가용 영역을 선택하고 생성을 선택합니다.
- 성공적으로 볼륨 생성 완료 메시지에서 방금 만든 볼륨을 선택합니다.
- 작업, 볼륨 연결을 선택합니다.
- 볼륨 연결 대화 상자에서 인스턴스 ID를 입력하고 연결을 선택합니다.
- Windows PowerShell을 사용하여 업그레이드를 시작하고 인스턴스에 연결한 설치 미디어 볼륨을 엽니다.
 - Windows Server 2016 이상으로 업그레이드한다면 다음을 실행합니다.

```
./setup.exe /auto upgrade
```

Windows Server의 이전 버전으로 업그레이드하는 경우 다음을 실행합니다.

Sources/setup.exe

- b. 설치하고자 하는 운영 체제 선택(Select the operating system you want to install)에서 Windows Server 인스턴스의 전체 설치 SKU를 선택한 후 다음을 선택합니다.
- c. 어떤 설치 유형으로 하시겠습니까?(Which type of installation do you want?)에서 업그레이드를 선택합니다.
- d. 마법사를 완료합니다.

Windows Server 설치 프로그램이 파일을 복사하고 처리합니다. 몇 분 후 원격 데스크톱 세션이 닫힙니다. 업그레이드하는 데 걸리는 시간은 Windows Server 인스턴스에서 실행하는 애플리케이션 및 서버 역할의 수에 따라 달라집니다. 업그레이드 프로세스는 최소한 40분 또는 몇 시간이 걸릴 수도 있습니다. 업그레이드 프로세스 중에는 인스턴스가 두 상태 확인 중 하나에 실패합니다. 업그레이드가 완료되면 두 상태 확인이 모두 통과됩니다. 시스템 로그에서 콘솔 출력을 확인하거나 디스크 또는 CPU 활동에 대한 Amazon CloudWatch 측정치를 사용하여 업그레이드가 진행되지 않는지 확인할 수 있습니다.

Note

Windows Server 2019로 업그레이드할 경우, 업그레이드가 완료된 후 필요하면 바탕 화면 배경을 직접 변경하여 이전 운영 체제 이름을 제거할 수 있습니다.

몇 시간 후에도 인스턴스가 두 상태 확인을 모두 통과하지 못한 경우에는 [업그레이드 문제 해결 \(p. 634\)](#) 단원을 참조하십시오.

Citrix PV 드라이버를 사용하여 인스턴스 인 플레이스 업그레이드

Citrix PV 드라이버는 Windows Server 2003 및 2008에서 사용됩니다. 업그레이드 과정에서 알려진 문제로, Windows 설치 프로그램이 사용자가 원격 데스크톱을 사용하여 인스턴스에 연결할 수 있도록 해주는 반가상화(PV) 드라이버의 일부를 제거하는 문제가 있습니다. 이 문제를 방지하기 위해, 다음 절차에서 현재 위치 업그레이드 시 업그레이드 헬퍼 서비스를 사용하는 방법을 설명합니다.

업그레이드 헬퍼 서비스 사용

업그레이드를 시작하기 전에 먼저 업그레이드 헬퍼 서비스를 실행해야 합니다. 이를 실행하면 유틸리티가 업그레이드 이후 단계에서 실행되는 Windows 서비스를 생성하여 드라이버 상태를 수정합니다. 실행 파일은 C#로 작성되고 .NET Framework 버전 2.0부터 4.0에서 실행할 수 있습니다.

업그레이드하기 전에 시스템에서 업그레이드 헬퍼 서비스를 실행하면 다음 작업이 수행됩니다.

- UpgradeHelperService라는 새 Windows 서비스를 생성합니다.
- Citrix PV 드라이버가 설치되어 있는지 확인합니다.
- 서명되지 않은 부팅 필수 드라이버가 있는지 확인하고, 발견될 경우 경고를 표시합니다. 서명되지 않은 부팅 필수 드라이버가 있으면, 드라이버가 새 Windows Server 버전과 호환되지 않을 경우 업그레이드 후 시스템 오류가 발생할 수 있습니다.

업그레이드한 후에 시스템에서 업그레이드 헬퍼 서비스를 실행하면 다음 작업이 수행됩니다.

- 올바른 시간 동기화를 위해 RealTimeIsUniversal 레지스트리 키를 활성화합니다.
- 다음 명령을 실행하여 누락된 PV 드라이버를 복원합니다.

```
pnputil -i -a "C:\Program Files (x86)\Citrix\XenTools\*.inf"
```

- 다음 명령을 실행하여 누락된 디바이스를 설치합니다.

```
C:\Temp\EC2DriverUtils.exe install "C:\Program Files (x86)\Citrix\XenTools\xevtchn.inf"
ROOT\XENEVTCHN
```

- 완료되면 자동으로 UpgradeHelperService를 제거합니다.

Citrix PV 드라이브를 실행하는 인스턴스에서 업그레이드 수행

업그레이드를 완료하려면 EC2 인스턴스에 설치 미디어 볼륨을 연결하고 UpgradeHelperService.exe를 사용해야 합니다.

Citrix PV 드라이버를 실행하는 Windows Server 인스턴스를 업그레이드하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 인스턴스를 찾습니다. 인스턴스 ID 및 인스턴스의 Availability Zone을 기록해둡니다. 이 정보는 이 절차의 뒷부분에서 필요합니다.
3. Windows Server 설치 미디어 스냅샷에서 새 볼륨을 생성합니다.
 - a. 탐색 창에서 스냅샷, 퍼블릭 스냅샷을 선택합니다.
 - b. 소유자 필터를 추가하고 Amazon 이미지를 선택합니다.
 - c. 설명 필터를 추가하고 windows를 입력합니다. Enter를 누릅니다.
 - d. 인스턴스의 시스템 아키텍처와 일치하는 스냅샷을 선택합니다. 예를 들면 Windows 2008 64-bit Installation Media가 있습니다.
 - e. 작업, 볼륨 생성을 선택합니다.
 - f. 볼륨 생성 대화 상자에서 Windows 인스턴스와 일치하는 가용 영역을 선택한 후 생성을 선택합니다.
4. 성공적으로 볼륨 생성 완료 대화 상자에서 방금 만든 볼륨을 선택합니다.
5. 작업, 볼륨 연결을 선택합니다.
6. 볼륨 연결 대화 상자에서 인스턴스 ID를 입력하고 연결을 선택합니다.
7. Windows 인스턴스의 경우 c:\ 드라이브에 temp라는 이름의 폴더를 생성합니다.

Important

이 폴더는 업그레이드 후 동일한 위치에서 사용할 수 있어야 합니다. Windows 시스템 폴더 또는 사용자 프로필 폴더(예: 바탕 화면)에 폴더를 생성하면 업그레이드에 실패할 수 있습니다.

8. OSUpgrade.zip을 다운로드한 후 C:\temp 폴더로 압축을 풁니다.
9. C:\temp\UpgradeHelperService.exe를 실행하고 C:\temp\Log.txt 파일에서 경고가 있는지 확인합니다.
10. Microsoft의 [기술 자료 문서 950376](#)을 사용하여 Windows 2003 인스턴스에서 PowerShell을 제거합니다.
11. Windows 탐색기를 이용해 업그레이드를 시작하여 인스턴스에 연결한 설치 미디어 볼륨을 엽니다.
12. Sources\Setup.exe 파일을 실행합니다.
13. 설치하고자 하는 운영 체제 선택>Select the operating system you want to install)에서 Windows Server 인스턴스의 전체 설치 SKU를 선택한 다음 다음을 선택합니다.
14. 어떤 설치 유형으로 하시겠습니까?(Which type of installation do you want?)에서 업그레이드를 선택합니다.
15. 마법사를 완료합니다.

Windows Server 설치 프로그램이 파일을 복사하고 처리합니다. 몇 분 후 원격 데스크톱 세션이 닫힙니다. 업그레이드하는 데 걸리는 시간은 Windows Server 인스턴스에서 실행하는 애플리케이션 및 서버 역할의 수에 따라 달라집니다. 업그레이드 프로세스는 최소한 40분 또는 몇 시간이 걸릴 수도 있습니다. 업그레이드 프로세스 중에는 인스턴스가 두 상태 확인 중 하나에 실패합니다. 업그레이드가 완료되면 두 상태 확인이 모두 통과됩니다. 시스템 로그에서 콘솔 출력을 확인하거나 디스크 또는 CPU 활동에 대한 Amazon CloudWatch 측정치를 사용하여 업그레이드가 진행되지 않는지 확인할 수 있습니다.

업그레이드 이후 작업

1. 인스턴스에 로그인하여 .NET Framework의 업그레이드를 시작하고 메시지가 나타나면 시스템을 재부팅합니다.
2. 최신 버전의 EC2Config 서비스(Windows 2012 R2 이전 버전) 또는 EC2Launch(Windows 2016 이상)를 설치합니다. 자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 또는 [최신 버전의 EC2Launch 설치 \(p. 490\)](#) 단원을 참조하십시오.
3. Microsoft 핫픽스 [KB2800213](#)을 설치합니다.
4. Microsoft 핫픽스 [KB2922223](#)을 설치합니다.
5. Windows Server 2012 R2로 업그레이드한 경우에는 PV 드라이버를 AWS PV 드라이버로 업그레이드하는 것이 좋습니다. Nitro 기반 인스턴스로 업그레이드한 경우 NVME 및 ENA 드라이버를 설치하거나 업그레이드하는 것이 좋습니다. 자세한 내용은 [Windows Server 2012 R2, AWS NVMe 드라이버 설치 또는 업그레이드 \(p. 539\)](#) 또는 [Windows에서 향상된 네트워킹 기능 활성화](#)를 참조하십시오.
6. 안티바이러스와 안티스파이웨어 소프트웨어 및 방화벽을 다시 활성화합니다.

자동 업그레이드 수행

AWS 시스템 관리자 자동화 문서를 사용하여 AWS의 Windows Server 2008 R2, Windows Server 2012 R2, Windows Server 2016 및 SQL Server 2008 R2(서비스 팩 3 인스턴스 포함)에서 자동 업그레이드를 수행할 수 있습니다.

시스템 관리자 자동화 문서는 두 가지 업그레이드 경로를 소개합니다.

- Windows Server 2008 R2, 2012 R2 또는 2016에서 Windows Server 2012 R2, 2016 또는 2019로(이름이 [AWSEC2-CloneInstanceAndUpgradeWindows](#)인 자동화용 시스템 관리자 문서 사용)
- Windows Server 2012 R2dml SQL Server 2008 R2에서 SQL Server 2016으로(이름이 [AWSEC2-CloneInstanceAndUpgradeSQLServer](#)인 자동화용 시스템 관리자 문서 사용)

목차

- [관련 서비스 \(p. 616\)](#)
- [사전 조건 \(p. 617\)](#)
- [업그레이드 경로 \(p. 618\)](#)
- [자동 업그레이드 수행 단계 \(p. 619\)](#)

관련 서비스

다음 AWS 서비스는 자동 업그레이드 프로세스에 사용됩니다.

- AWS 시스템 관리자. AWS 시스템 관리자는 강력한 통합 인터페이스로서 AWS 리소스를 중앙 집중식으로 관리합니다. 자세한 내용은 [AWS 시스템 관리자 사용 설명서](#)를 참조하십시오.
- AWS 시스템 관리자 에이전트(SSM 에이전트)는 Amazon EC2 인스턴스, 온프레미스 서버 또는 가상 머신(VM)에 설치 및 구성할 수 있는 Amazon 소프트웨어입니다. SSM 에이전트를 사용하면 시스템 관리자가 이러한 리소스를 업데이트, 관리 및 구성할 수 있습니다. 에이전트는 AWS 클라우드에서 시스템 관리자 서비스의 요청을 처리하고 요청에 지정된 대로 실행합니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [SSM 에이전트 작업](#)을 참조하십시오.
- AWS 시스템 관리자 SSM 문서. SSM 문서는 시스템 관리자가 관리형 인스턴스에서 실행하는 작업을 정의합니다. SSM 문서는 JavaScript Object Notation(JSON) 또는 YAML을 사용하며 사용자가 지정하는 단계와 파라미터를 포함합니다. 이 주제에서는 두 자동화용 시스템 관리자 SSM 문서를 사용합니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 문서](#)를 참조합니다.

사전 조건

AWS 시스템 관리자 자동화 문서로 업그레이드를 자동화하려면 다음 작업을 수행해야 합니다.

- [지정한 IAM 정책으로 IAM 규칙 만들기 \(p. 617\)](#)를 하여 시스템 관리자가 Amazon EC2 인스턴스에서 자동화 작업을 수행하고 사용자가 필수적인 시스템 관리자 사용 조건에 부합하는지 확인합니다.
- [원하는 자동화 실행 방법과 관련해 옵션을 선택합니다 \(p. 617\)](#). 실행 옵션으로는 Simple execution(간편 실행), Rate control(속도 제어), Multi-account and Region(다중 계정 및 리전), Manual execution(수동 실행)이 있습니다.

지정된 권한으로 IAM 역할 생성

AWS 시스템 관리자가 사용자를 대신해 리소스에 액세스할 수 있도록 IAM 역할을 생성하는 방법은 IAM 사용 설명서의 [AWS 서비스에 권한을 위임할 역할 생성](#)에서 단계를 확인하십시오. 또한 이 주제에서는 사용자의 계정이 필수적인 시스템 관리자 사용 조건에 부합하는지 확인하는 방법에 대한 정보도 소개합니다.

실행 옵션 선택

시스템 관리자 콘솔에서 자동화를 선택할 때 실행을 선택합니다. 자동화 문서를 선택한 후 자동화 실행 옵션을 선택하라는 메시지가 표시됩니다. 다음 옵션 중 하나를 선택합니다. 이 주제에서 소개하는 경로에서는 각 단계에 간편 실행 옵션을 사용합니다.

간편 실행

단일 인스턴스를 업데이트하지만 각각의 자동화 단계를 거쳐 결과를 감사하지는 않으려면 이 옵션을 선택합니다. 이 옵션은 이어지는 업그레이드 단계에서 자세히 설명됩니다.

속도 제어

둘 이상의 인스턴스에 업그레이드를 적용하려면 이 옵션을 선택합니다. 다음 설정을 정의합니다.

- **파라미터**

이 설정은 다중 계정 및 리전 설정에도 적용되어 있는데, 자동화의 브랜칭 아웃 방법을 정의합니다.

- **대상**

자동화를 적용하려는 대상을 선택합니다. 또한 이 설정은 다중 계정 및 리전 설정에도 적용됩니다.

- **파라미터 값**

자동화 문서 파라미터에 정의된 값을 사용합니다.

- **리소스 그룹**

AWS에서 리소스란 사용할 수 있는 개체를 뜻합니다. 그 예로는 Amazon EC2 인스턴스, AWS CloudFormation 스택, 또는 Amazon S3 버킷이 포함됩니다. 다수의 리소스를 사용할 경우, 작업을 할 때마다 AWS 서비스 사이를 이동하는 것보다는 해당 리소스를 그룹으로 관리하는 것이 더 유용할 수 있습니다. 몇몇 경우에는 하나의 애플리케이션 계층을 구성하는 EC2 인스턴스 등의 관련 리소스를 다수로 관리해야 할 수 있습니다. 이런 경우에는 해당 리소스에 대한 일괄 작업을 한 번에 수행해야 할 수 있습니다.

- **태그**

태그를 사용하면 AWS 리소스를 용도별, 소유자별 또는 환경별 등 다양한 방식으로 분류할 수 있습니다. 이처럼 분류를 해 놓으면 동일한 유형의 리소스가 많을 때 유용합니다. 할당된 태그를 사용하여 특정 리소스를 신속하게 식별할 수 있습니다.

- **속도 제어**

또한 속도 제어는 다중 계정 및 리전 설정에도 적용됩니다. 속도 제어 파라미터를 설정하면 대상 개수 또는 백분율로 자동화를 적용하려는 플랫의 수를 정의합니다.

다중 계정 및 리전

속도 제어에서 지정되고 다중 계정 및 리전 설정에도 사용되는 파라미터에 더해 추가로 살펴볼 설정이 두 가지 더 있습니다.

- 계정 및 OU(조직 단위)

자동화를 실행할 여러 계정을 지정합니다.

- AWS 리전

자동화를 실행할 여러 AWS 리전을 지정합니다.

수동 실행

이 옵션은 간편 실행과 비슷하지만 각 자동화 단계를 거치고 결과를 감사할 수 있다는 데서 차이를 보입니다.

업그레이드 경로

업그레이드 경로는 두 가지로, 서로 다른 두 개의 AWS 시스템 관리자 Automation 문서를 사용합니다.

- [AWSEC2-CloneInstanceAndUpgradeWindows](#)를 선택하십시오. 이 스크립트는 사용자 계정에 있는 Windows Server 2008 R2, 2012 R2 또는 2016 인스턴스에서 Amazon Machine Image(AMI)를 생성하고 이 AMI를 선택한 지원되는 버전(Windows Server 2012 R2, 2016 또는 2019)으로 업그레이드합니다. 이 단계 프로세스는 완료까지 최대 2시간이 소요됩니다.

Windows Server 2008 R2 인스턴스를 Windows Server 2016 또는 2019로 업그레이드하기 위해서는 인 플레이스 업그레이드가 두 번 수행됩니다. 먼저 Windows Server 2008 R2에서 Windows Server 2012 R2로 업그레이드한 다음 Windows Server 2012 R2에서 Windows Server 2016 또는 2019로 업그레이드합니다. Windows Server 2008 R2를 Windows Server 2016 또는 2019로 직접 업그레이드하는 것은 지원되지 않습니다.

이 워크플로우에서는 자동화를 통해 인스턴스에서 AMI를 생성한 다음 사용자가 제공하는 서브넷에서 새 AMI를 실행합니다. 자동화 워크플로는 Windows Server 2008 R2, 2012 R2 또는 2016에서 선택한 버전(Windows Server 2012 R2, 2016 또는 2019)으로 적절한 업그레이드를 수행합니다. 또한 이 워크플로는 업그레이드된 인스턴스에 필요한 AWS 드라이버를 업데이트하거나 설치합니다. 이 워크플로는 업그레이드 완료 후 새 AMI를 생성하고 업그레이드된 인스턴스를 종료합니다. Windows Server 2008 R2에서 Windows Server 2016 또는 2019로 업그레이드하는 경우 인 플레이스 업그레이드가 두 번 수행되므로 자동화에서 두 개의 AMI를 만듭니다.

- [AWSEC2-CloneInstanceAndUpgradeSQLServer](#)를 선택하십시오. 이 스크립트는 계정 내에서 SQL Server 2008 R2 SP3를 실행하는 Amazon EC2 인스턴스에서 AMI를 생성한 다음 해당 AMI를 SQL Server 2016 SP2로 업그레이드합니다. 이 단계 프로세스는 완료까지 최대 2시간이 소요됩니다.

이 워크플로우에서는 자동화를 통해 인스턴스에서 AMI를 생성한 다음 사용자가 제공하는 서브넷에서 새 AMI를 실행합니다. 그런 다음 이 자동화는 SQL Server 2008 R2에서 SQL Server 2016 SP2로 현재 위치 업그레이드를 수행합니다. 이 자동화는 업그레이드 완료 후 업그레이드된 인스턴스를 종료하기 전에 새 AMI를 생성합니다.

자동 업그레이드 프로세스에는 두 개의 AMI가 포함됩니다.

- **현재 실행 중인 인스턴스:** 첫 번째 AMI는 현재 실행 중인 인스턴스로서 업그레이드되지 않았습니다. 이 AMI는 다른 인스턴스를 시작하여 현재 위치 업그레이드를 실행하는 데 쓰입니다. 프로세스가 완료되면 이 AMI는 원본 인스턴스를 유지하도록 특별히 요청하지 않는 한 계정에서 삭제됩니다. 이 설정은 파라미터 `KeepPreUpgradeImageBackup`에서 처리합니다(기본값은 `false`이며, 즉 AMI가 기본적으로 삭제됨).
- **업그레이드된 AMI:** 이 AMI는 자동화 프로세스의 결과물입니다. 두 번째 AMI에는 SQL Server 2008 R2 가 아닌 SQL Server 2016 SP2가 포함됩니다.

최종 결과물은 하나의 AMI로서, 업그레이드된 AMI 인스턴스입니다.

업그레이드가 완료되면 VPC에서 새 AMI를 시작하여 애플리케이션 기능을 테스트할 수 있습니다. 테스트 이후와 다른 업그레이드를 수행하기 전에는 업그레이드된 인스턴스로 완전히 전환하기에 앞서 애플리케이션 중단 시간을 예약하십시오.

자동 업그레이드 수행 단계

업그레이드 경로

- Windows Server 2008 R2, 2012 R2 또는 2016을 Windows Server 2012 R2, 2016 또는 2019로 업그레이드 (p. 619)
- SQL Server 2008 R2에서 SQL Server 2016으로 업그레이드 (p. 620)

Windows Server 2008 R2, 2012 R2 또는 2016을 Windows Server 2012 R2, 2016 또는 2019로 업그레이드

이 업그레이드 경로에는 작업 완료를 위한 사전 조건이 추가로 필요합니다. 해당 사전 조건은 AWS 시스템 관리자 사용 설명서의 [AWSEC2-CloneInstanceAndUpgradeWindows](#) 자동화 문서 세부 정보에서 확인할 수 있습니다.

추가적 사전 조건 작업을 확인한 후에는 다음의 단계를 따라 AWS 시스템 관리자의 자동화 문서를 사용하여 Windows 2008 R2 인스턴스를 Windows 2012 R2로 업그레이드합니다.

1. AWS Management Console에서 시스템 관리자를 엽니다.
2. 왼쪽 탐색 창에서 자동화를 선택합니다.
3. 자동화 실행(Execute automation)을 선택합니다.
4. AWSEC2-CloneInstanceAndUpgradeWindows라는 이름의 자동화 문서를 찾습니다.
5. 문서 이름이 보이면 선택합니다. 해당 이름을 선택하면 문서의 세부 정보가 표시됩니다.
6. 다음을 선택하여 이 문서의 파라미터를 입력합니다. 페이지 상단에서 Simple execution(간편 실행)이 선택된 채로 듭니다.
7. 요청된 파라미터를 다음 지침을 따라 입력합니다.

- InstanceID

유형: 문자열

(필수) SSM 에이전트가 설치된 상태에서 Windows Server 2008 R2, 2012 R2 또는 2016을 실행하는 인스턴스입니다.

- InstanceProfile를 선택하십시오.

유형: 문자열

(필수) IAM 인스턴스 프로파일입니다. Amazon EC2 인스턴스 및 AWS AMI에 대해 시스템 관리자 자동화를 수행하는데 사용되는 IAM 규칙입니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서에서 시스템 관리자에 대한 [IAM 인스턴스 프로파일 생성](#)을 참조하십시오.

- TargetWindowsVersion

유형: 문자열

(필수) 대상 Windows 버전을 선택합니다.

- SubnetId

유형: 문자열

(필수) 업그레이드 프로세스를 위한 서브넷으로, 소스 EC2 인스턴스가 상주합니다. 서브넷에서 (패치를 다운로드할) AWS 서비스(Amazon S3 포함) 및 Microsoft로의 아웃바운드 연결이 설정되었는지 확인합니다.

- `KeepPreUpgradedBackUp`

유형: 문자열

(선택 사항) 이 파라미터가 `true`로 설정된 경우, 인스턴스에서 생성된 이미지는 자동화 과정에서 그대로 유지됩니다. 기본 설정은 `false`입니다.

- `RebootInstanceBeforeTakingImage`

유형: 문자열

(선택 사항) 기본값은 `false`입니다(재부팅 안 함). 이 파라미터가 `true`로 설정된 경우, 시스템 관리자에서는 업그레이드를 위해 AMI를 만들기 전에 인스턴스를 재부팅합니다.

8. 파라미터를 입력한 다음 실행을 선택합니다. 자동화가 시작되면 실행 진척도를 모니터링할 수 있습니다.
9. 자동화가 완료되면 AMI ID를 볼 수 있습니다. AMI를 시작하여 Windows OS의 업그레이드 여부를 확인할 수 있습니다.

Note

모든 단계를 자동화로 실행할 필요는 없습니다. 해당 단계는 자동화와 인스턴스의 동작에 따라 달라질 수 있습니다. 시스템 관리자는 필수가 아닌 단계는 건너뛸 수 있습니다.

또한 일부 단계는 시간이 초과될 수 있습니다. 시스템 관리자는 모든 최신 패치의 업그레이드와 설치를 시도합니다. 그러나 때에 따라서 특정 단계의 정의 가능 제한 시간 설정에 따라 패치의 시간이 초과됩니다. 이 경우에는 내부 OS가 대상 Windows Server 버전으로 업그레이드될 수 있도록 시스템 관리자 자동화가 다음 단계까지 계속됩니다.

10. 자동화가 완료되고 나면 AMI ID로 Amazon EC2 인스턴스를 실행하여 업그레이드를 검토할 수 있습니다. AWS AMI에서 Amazon EC2 인스턴스를 만드는 방법에 대한 자세한 내용은 [사용자 지정 AMI\(Amazon 머신 이미지\)에서 EC2 인스턴스를 시작하려면 어떻게 해야 합니까?](#)를 참조하십시오.

SQL Server 2008 R2에서 SQL Server 2016으로 업그레이드

이 업그레이드 경로에는 작업 완료를 위한 사전 조건이 추가로 필요합니다. 해당 사전 조건은 AWS 시스템 관리자 사용 설명서의 [AWSEC2-CloneInstanceAndUpgradeSQLServer](#) 자동화 문서 세부 정보에서 확인할 수 있습니다.

추가적 사전 조건 작업을 확인한 후에는 다음의 단계를 따라 AWS 시스템 관리자의 자동화 문서를 사용하여 SQL Server 2008 R2 데이터베이스 앤진을 SQL Server 2016으로 업그레이드합니다.

1. 아직 확인이 되지 않은 경우에는 SQL Server 2016을 .iso 파일로 다운로드하고 이를 소스 서버에 마운트 합니다.
2. .iso 파일 마운팅 후에는 구성 요소 파일을 모두 복사하고 원하는 볼륨에 배치합니다.
3. 볼륨의 EBS 스냅샷을 만들고 스냅샷의 ID를 이후에 사용할 클립보드에 복사합니다. EBS 스냅샷 생성에 대한 자세한 내용은 Amazon Elastic Compute Cloud 사용 설명서의 [EBS 스냅샷 만들기](#)를 참조하십시오.
4. 인스턴스 프로파일을 EC2 소스 인스턴스에 연결합니다. 그러면 시스템 관리자가 EC2 인스턴스와 통신하고 해당 인스턴스가 AWS 시스템 관리자 서비스에 추가된 후에 명령을 실행할 수 있습니다. 이 예에서는 역할을 `SSM-EC2-Profile-Role`로 명명했으며 해당 역할에는 `AmazonSSMManagedInstanceCore` 정책이 연결되었습니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서에서 시스템 관리자에 대한 [IAM 인스턴스 프로파일 생성](#)을 참조하십시오.
5. AWS 시스템 관리자 콘솔의 왼쪽 탐색 창에서 `Managed Instances`(관리형 인스턴스)를 선택합니다. EC2 인스턴스가 관리형 인스턴스 목록에 있는지 확인합니다. 몇 분 후에도 인스턴스가 표시되지 않으면 AWS 시스템 관리자 사용 설명서의 [제 인스턴스는 어디에 있나요?](#)를 참조하십시오.

6. 왼쪽 탐색 창에서 Automation(자동화)을 선택합니다.
7. Execute automation(자동화 실행)을 선택합니다.
8. AWSEC2-CloneInstanceAndUpgradeSQLServer SSM 문서 옆에 있는 버튼을 선택하고 Next(다음)를 선택합니다.
9. Simple execution(간편 실행) 옵션이 선택되었는지 확인합니다.
10. 요청된 파라미터를 다음 지침을 따라 입력합니다.

- `InstanceId`

유형: 문자열

(필수) SQL Server 2008 R2(또는 이후 버전) 실행 인스턴스입니다.

- `IamInstanceProfile`

유형: 문자열

(필수) IAM 인스턴스 프로파일입니다.

- `SnapshotId`

유형: 문자열

(필수) SQL Server 2016 설치 미디어의 스냅샷 ID입니다.

- `SubnetId`

유형: 문자열

(필수) 업그레이드 프로세스를 위한 서브넷으로, 소스 EC2 인스턴스가 상주합니다. 서브넷에서 (패치를 다운로드할) AWS 서비스(Amazon S3 포함) 및 Microsoft로의 아웃바운드 연결이 설정되었는지 확인합니다.

- `KeepPreUpgradedBackUp`

유형: 문자열

(선택 사항) 이 파라미터가 `true`로 설정된 경우, 인스턴스에서 생성된 이미지는 자동화 과정에서 그대로 유지됩니다. 기본 설정은 `false`입니다.

- `RebootInstanceBeforeTakingImage`

유형: 문자열

(선택 사항) 기본값은 `false`입니다(재부팅 안 함). 이 파라미터가 `true`로 설정된 경우, 시스템 관리자에서는 업그레이드를 위해 AMI를 만들기 전에 인스턴스를 재부팅합니다.

11. 파라미터를 입력한 다음 Execute(실행)를 선택합니다. 자동화가 시작되면 실행 진척도를 모니터링할 수 있습니다.
12. Execution Status(실행 상태)가 Success(성공)로 표시되면 Outputs(출력)를 선택하여 AMI 정보를 봅니다. AMI ID를 사용하여 원하는 VPC에 대해 SQL Server 2016 인스턴스를 시작할 수 있습니다.
13. EC2 콘솔을 엽니다. 왼쪽 탐색 창에서 AMI를 선택합니다. 그러면 새 AMI가 보일 것입니다.
14. SQL Server 2016이 성공적으로 설치되었는지 확인하려면 새 AMI를 선택하고 시작을 선택합니다.
15. 배포하려는 AMI, VPC 및 서브넷, 그리고 사용하려는 스토리지에 대해 인스턴스 유형을 선택합니다. AMI에서 새 인스턴스를 시작하므로 해당 볼륨은 사용자가 시작하는 EC2 인스턴스 내에 포함할 옵션으로 표시됩니다. 이 볼륨은 제거할 수 있으며, 반대로 추가할 수도 있습니다.
16. 인스턴스 식별에 도움이 되도록 태그를 추가합니다.
17. 보안 그룹을 인스턴스에 추가합니다.
18. 인스턴스 시작을 선택합니다.
19. 인스턴스의 이름을 선택하고 작업 드롭다운 메뉴 아래에서 연결을 선택합니다.

20. SQL Server 2016이 새 인스턴스의 신규 데이터베이스 엔진인지 확인합니다.

최신 세대 인스턴스 유형으로 마이그레이션

AWS Windows AMI는 Microsoft 설치 미디어에서 사용하는 기본 설정 및 일부 사용자 지정으로 구성됩니다. 사용자 지정에는 최신 세대 인스턴스 유형을 지원하는 드라이버 및 구성이 포함됩니다. 그러나 다음과 같은 경우에는 베어 메탈 인스턴스를 포함하여 최신 세대 EC2 인스턴스와 Nitro 인스턴스로 마이그레이션할 때 이 주제의 단계를 따르는 것이 좋습니다.

- 사용자 지정 Windows AMI에서 인스턴스를 시작하는 경우
- 2018년 8월 이전에 생성된, Amazon에서 제공한 Windows AMI에서 인스턴스를 시작하는 경우

자세한 내용은 [Amazon EC2 업데이트 - 추가 인스턴스 유형, Nitro 시스템 및 CPU 옵션](#)을 참조하십시오.

목차

- [1부: AWS PV 드라이버 설치 및 업그레이드 \(p. 623\)](#)
- [2부: ENA 설치 및 업그레이드 \(p. 623\)](#)
- [3부: AWS NVMe 드라이버 업그레이드 \(p. 624\)](#)
- [4부: EC2Config와 EC2Launch 업데이트 \(p. 624\)](#)
- [5부: 베어 메탈 인스턴스를 위한 직렬 포트 설치 \(p. 625\)](#)
- [6부: 전원 관리 설정 업데이트 \(p. 625\)](#)
- [7부: 새 인스턴스 유형에 대한 인텔 칩셋 드라이버 업데이트 \(p. 625\)](#)
- [\(대안\) AWS 시스템 관리자를 사용하여 AWS PV, ENA 및 NVMe 드라이버 업그레이드 \(p. 626\)](#)

Note

또는 [AWS Support - Upgrade Windows AWS Drivers](#) 자동화 문서를 사용하여 1부, 2부 및 3부에서 설명한 절차를 자동화할 수 있습니다. 자동화된 절차 사용을 선택한 경우 다음([\(대안\) AWS 시스템 관리자를 사용하여 AWS PV, ENA 및 NVMe 드라이버 업그레이드 \(p. 626\)](#))을 참조하고 4부와 5부로 계속 진행합니다.

시작하기 전에

이 절차에서는 전 세대 Xen 기반 인스턴스 유형(M4 또는 C4 등)에서 현재 실행 중이며, M5 또는 C5 등의 최신 세대 인스턴스 유형으로 마이그레이션하는 경우를 전제로 설명합니다.

업그레이드를 성공적으로 수행하려면 PowerShell 버전 3.0 이상을 사용해야 합니다.

Note

최대 세대 인스턴스로 마이그레이션하는 경우 해당 인스턴스가 새 향상된 네트워킹 어댑터 디바이스로 기본 설정되므로 기존 ENI에 대한 정적 IP 또는 사용자 지정 DNS 네트워크 설정이 손실될 수 있습니다.

이 절차의 단계를 수행하기 전에 인스턴스를 백업해 놓는 것이 좋습니다. [EC2 콘솔](#)에서 마이그레이션 필요인 인스턴스를 선택하고, 바로 가기(마우스 오른쪽 버튼 클릭) 메뉴를 열고 인스턴스 상태를 선택한 후 종지를 선택합니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 보존하기 위해 영구 스토리지에 데이터를 백업해야 합니다.

[EC2 콘솔](#)에서 인스턴스의 바로 가기(마우스 오른쪽 버튼 클릭) 메뉴를 열고 이미지를 선택한 후 이미지 생성을 선택합니다.

Note

이 지침의 4부와 5부는 인스턴스 유형을 최신 세대(M5 또는 C5 등)로 마이그레이션하거나 변경한 후 수행할 수 있습니다. 하지만 EC2 Bare Metal 인스턴스 유형으로 마이그레이션하는 경우에는 마이그레이션 전에 완료하는 것이 좋습니다.

1부: AWS PV 드라이버 설치 및 업그레이드

Nitro 시스템에서 AWS PV 드라이버를 사용하지 않더라도 이전 버전의 Citrix PV 또는 AWS PV를 사용하는 경우 업그레이드해야 합니다. 최신 AWS PV 드라이버는 이전 버전 드라이버에서 Nitro 시스템을 사용하거나 Xen 기반 인스턴스로 다시 마이그레이션해야 할 경우 발생할 수 있는 버그를 해결했습니다. 항상 AWS의 Windows 인스턴스용 최신 드라이버로 업데이트하는 것이 가장 좋습니다.

다음 절차에 따라 AWS PV 드라이버의 현재 위치 업그레이드를 수행하거나, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016 또는 Windows Server 2019의 Citrix PV 드라이버에서 AWS PV 드라이버로 업그레이드할 수 있습니다. 자세한 내용은 [Windows 인스턴스에서 PV 드라이버 업그레이드 \(p. 528\)](#) 단원을 참조하십시오.

도메인 컨트롤러를 업그레이드하려면 [도메인 컨트롤러 업그레이드\(AWS PV 업그레이드\) \(p. 529\)](#) 단원을 참조하십시오.

AWS PV 드라이버의 업그레이드를 수행하려면

- 원격 데스크톱을 사용하여 인스턴스에 연결하고 인스턴스를 업그레이드할 준비를 합니다. 업그레이드를 수행하기 전에 시스템 디스크가 아닌 모든 디스크를 오프라인으로 전환합니다. AWS PV 드라이버의 현재 위치 업그레이드를 수행할 경우에는 이 단계가 필요하지 않습니다. 또한 서비스 콘솔에서 필수적이지 않은 서비스를 수동 시작으로 설정합니다.
- 최신 드라이버 패키지를 인스턴스로 [다운로드](#)합니다.
- 폴더의 내용 압축을 풀고 `AWSPVDriverSetup.msi`를 실행합니다.

MSI를 실행하면 인스턴스가 자동으로 재부팅되고 드라이버를 업그레이드합니다. 최대 15분 동안 인스턴스를 사용할 수 없습니다.

업그레이드를 완료하고 인스턴스가 Amazon EC2 콘솔에서 두 상태 확인을 모두 통과하면 원격 데스크톱을 사용하여 인스턴스에 연결하고 새 드라이버가 설치되었는지 확인합니다. 디바이스 관리자(Device Manager)의 스토리지 컨트롤러(Storage Controllers) 아래에서 AWS PV Storage Host Adapter를 찾습니다. 드라이버 버전이 드라이버 버전 기록 표에 나열된 최신 버전과 동일한지 확인합니다. 자세한 내용은 [AWS PV 드라이버 패키지 내역 \(p. 524\)](#) 단원을 참조하십시오.

2부: ENA 설치 및 업그레이드

모든 네트워크 기능을 지원할 수 있도록 최신 Elastic Network Adapter 드라이버로 업그레이드합니다. 인스턴스를 시작했는데 향상된 네트워킹 기능이 활성화되어 있지 않은 경우에는 인스턴스에 필요한 네트워크 어댑터 드라이버를 다운로드하고 설치해야 합니다. enaSupport 인스턴스 속성을 설정하여 향상된 네트워킹을 활성화합니다. 이 속성은 지원되는 인스턴스 유형 및 ENA 드라이버가 설치된 경우에만 활성화할 수 있습니다. 자세한 내용은 [Windows 인스턴스에서 ENA\(Elastic Network Adapter\)를 사용하여 향상된 네트워킹 활성화 \(p. 751\)](#) 단원을 참조하십시오.

- 최신 드라이버를 인스턴스로 [다운로드](#)합니다.
- ZIP 아카이브를 추출합니다.
- 압축 파일을 폰 폴더에서 `install.ps1` PowerShell 스크립트를 실행하여 드라이버를 설치합니다.

Note

설치 오류를 피하려면 `install.ps1` 스크립트를 관리자로 실행하십시오.

4. AMI에 `enaSupport`가 활성화되었는지 확인합니다. 그렇지 않은 경우 [Windows 인스턴스에서 ENA\(Elastic Network Adapter\)를 사용하여 향상된 네트워킹 활성화 \(p. 751\)](#)의 설명서를 따라 계속하십시오.

3부: AWS NVMe 드라이버 업그레이드

AWS NVMe 드라이버는 성능 개선을 위해 Nitro 시스템에 NVMe 블록 디바이스로 표시되는 Amazon EBS 및 SSD 인스턴스 스토어 볼륨과 상호 작용하는 데 사용됩니다.

Important

다음 지침은 인스턴스를 최신 세대 인스턴스 유형으로 마이그레이션하기 위해 이전 세대 인스턴스에 AWS NVMe를 설치하거나 업그레이드할 경우 내용이 수정됩니다.

1. 최신 드라이버 패키지를 인스턴스로 [다운로드](#)합니다.
2. ZIP 아카이브를 추출합니다.
3. `dpinst.exe`를 실행하여 드라이버를 설치합니다.
4. PowerShell 세션을 열고 다음 명령을 실행합니다.

```
start rundll32.exe sppnp.dll,Sysprep_Generalize_Pnp -wait
```

Note

명령을 적용하려면 PowerShell 세션을 관리자로 실행해야 합니다.

이 명령은 드라이버 디바이스에서만 sysprep을 실행합니다. 전체 sysprep 준비를 실행하지는 않습니다.

5. Windows Server 2008 R2 및 Windows Server 2012의 경우 인스턴스를 종료하고, 인스턴스 유형을 최신 세대 인스턴스로 변경하고 시작한 후 4부로 계속 진행합니다. 최신 세대 인스턴스 유형으로 마이그레이션하기 전에 전 세대 인스턴스 유형에서 다시 인스턴스를 시작하면 부팅되지 않습니다. 지원되는 다른 Windows AMI를 위해, 디바이스에서 sysprep을 수행한 후 언제든지 인스턴스 유형을 변경할 수 있습니다.

4부: EC2Config와 EC2Launch 업데이트

Windows 인스턴스의 경우, 최신 EC2Config와 EC2Launch 유ти리티는 EC2 Bare Metal을 비롯한 Nitro 시스템에서 실행할 때 추가 기능과 정보를 제공합니다. 기본적으로 EC2Config 서비스는 Windows Server 2016 이전의 AMI에 포함되어 있습니다. EC2Launch는 Windows Server 2016 이상 AMI의 EC2Config를 대체합니다.

EC2Config 서비스와 EC2Launch 서비스가 업데이트되면 새 AWS Windows AMI에 최신 버전의 서비스가 포함됩니다. 그러나 자체 Windows AMI 및 인스턴스는 별도로 최신 버전의 EC2Config 및 EC2Launch로 업데이트해야 합니다.

EC2Config를 설치 또는 업데이트하려면

1. [EC2Config 설치 관리자](#)를 다운로드하고 압축을 풉니다.
2. 실행 `EC2Install.exe`. 전체 옵션 목록을 보려면 `EC2Install` 옵션을 포함해 `/?` 파일을 실행합니다. 기본적으로 설치는 프롬프트를 표시합니다. 프롬프트 없이 명령을 실행하려면 `/quiet` 옵션을 사용합니다.

자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 단원을 참조하십시오.

EC2Launch를 설치 또는 업데이트하려면

1. 이미 인스턴스에 EC2Launch를 설치하여 구성한 경우 EC2Launch 구성 파일의 백업을 만듭니다. 설치 프로세스는 이 파일에 변경 사항을 보존하지 않습니다. 기본적으로 c:\ProgramData\Amazon\EC2-Windows\Launch\Config 디렉터리에 파일이 위치합니다.
2. 인스턴스의 디렉터리로 [EC2-Windows-Launch.zip](#) 파일을 다운로드합니다.
3. EC2-Windows-Launch.zip 파일을 다운로드한 동일한 디렉터리에 [install.ps1](#)을 다운로드합니다.
4. 실행 `install.ps1`.

Note

설치 오류를 피하려면 `install.ps1` 스크립트를 관리자로 실행하십시오.

5. EC2Launch 구성 파일의 백업을 만든 경우 C:\ProgramData\Amazon\EC2-Windows\Launch\Config 디렉터리에 복사합니다.

자세한 내용은 [EC2Launch를 사용한 Windows 인스턴스 구성 \(p. 489\)](#) 단원을 참조하십시오.

5부: 베어 메탈 인스턴스를 위한 직렬 포트 설치

i3.metal 인스턴스 유형은 I/O 포트 기반 직렬 디바이스가 아닌 PCI 기반 직렬 디바이스를 사용합니다. 최신 Windows AMI는 PCI 기반 직렬 디바이스를 자동으로 사용하며 직렬 포트 드라이버가 설치되어 있습니다. Amazon에서 제공한 2018.04.11일자 또는 그 이전 Windows AMI에서 시작한 인스턴스를 사용하지 않는 경우, 직렬 포트 드라이버를 설치하여 직렬 디바이스에서 EC2 기능(암호 생성 및 콘솔 출력 등)을 사용할 수 있도록 해야 합니다. 또한 최신 EC2Config와 EC2Launch 유ти리티는 i3.metal을 지원하고 추가 기능을 제공합니다. 아직 수행하지 않은 경우 4부의 단계를 수행할 것을 권장합니다.

직렬 시리얼 포트 드라이버를 설치하려면

1. 직렬 드라이버 패키지를 인스턴스로 [다운로드](#)합니다.
2. 폴더 내용의 압축을 풀고 aws_ser.INF의 컨텍스트 메뉴를 열고(마우스 오른쪽 버튼으로 클릭) 설치를 선택합니다.
3. 확인을 선택합니다.

6부: 전원 관리 설정 업데이트

전원 관리 설정에 대한 다음 업데이트는 디스플레이가 꺼지지 않도록 설정해 Nitro 시스템에서 OS를 정상적으로 종료할 수 있도록 합니다. Amazon에서 2018년 11월 28일 이후 제공한 모든 Windows AMI에는 이러한 기본 구성이 이미 적용되어 있습니다.

1. 명령 프롬프트 또는 PowerShell 세션을 엽니다.
2. 다음 명령을 실행합니다:

```
powercfg /setacvalueindex 381b4222-f694-41f0-9685-ff5bb260df2e 7516b95f-f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbc2b7e 0
powercfg /setacvalueindex 8c5e7fda-e8bf-4a96-9a85-a6e23a8c635c 7516b95f-f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbc2b7e 0
powercfg /setacvalueindex a1841308-3541-4fab-bc81-f71556f20b4a 7516b95f-f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbc2b7e 0
```

7부: 새 인스턴스 유형에 대한 인텔 칩셋 드라이버 업데이트

u-6tb1.metal, u-9tb1.metal 및 u-12tb1.metal 인스턴스 유형은 이전에 Windows AMI에 설치되지 않은 칩셋 드라이버가 필요한 하드웨어를 사용합니다. Amazon에서 제공한 2018.11.19일자 또는 그 이전

Windows AMI에서 시작한 인스턴스를 사용하지 않는 경우, 인텔 Chipset INF 유ти리티를 사용하여 드라이버를 설치해야 합니다.

칩셋 드라이버를 설치하려면

1. 인스턴스로 [칩셋 유ти리티를 다운로드합니다.](#)
2. 파일의 압축을 풁니다.
3. 실행 `SetupChipset.exe`.
4. 인텔 소프트웨어 라이선스 계약에 동의하고 칩셋 드라이버를 설치합니다.
5. 인스턴스를 재부팅합니다.

(대안) AWS 시스템 관리자를 사용하여 AWS PV, ENA 및 NVMe 드라이버 업그레이드

AWS Support - Upgrade Windows AWS Drivers 자동화 문서는 1부, 2부 및 3부에서 설명한 단계를 자동화합니다. 이 방법은 또한 드라이버 업그레이드 실패 시 인스턴스를 복구할 수 있습니다.

AWS Support - Upgrade Windows AWS Drivers 자동화 문서는 지정된 EC2 인스턴스의 스토리지 및 네트워크 AWS 드라이버를 업그레이드 또는 복구합니다. 문서는 AWS 시스템 관리자 에이전트(SSM 에이전트)를 호출하여 온라인으로 AWS 드라이버의 최신 버전 설치를 시도합니다. SSM 에이전트에 접속할 수 없다면 문서는 명시적 요청된 경우 AWS 드라이버의 오프라인 설치를 수행할 수 있습니다.

Note

이 절차는 도메인 컨트롤러에는 수행하지 못합니다. 도메인 컨트롤러의 드라이버를 업데이트하려면 [도메인 컨트롤러 업그레이드\(AWS PV 업그레이드\)](#) (p. 529)을 참조하십시오.

AWS 시스템 관리자 사용을 통해 AWS PV, ENA 및 NVMe 드라이버를 자동으로 업그레이드하려면

1. <https://console.aws.amazon.com/systems-manager>에서 시스템 관리자 콘솔을 엽니다.
2. 자동화를 선택한 다음, 자동화 실행(Execute automation)을 선택합니다.
3. AWS Support - Upgrade Windows AWS Drivers 자동화 문서를 선택한 다음 입력 파라미터 섹션에서 다음 옵션을 구성합니다.

인스턴스 ID

업그레이드할 인스턴스의 고유 ID를 입력합니다.

AllowOffline

(선택 사항) 다음 옵션 중 하나를 선택합니다.

- `True` - 이 옵션을 선택하여 오프라인 설치를 수행합니다. 업그레이드 프로세스 도중에는 인스턴스가 중지 및 재시작됩니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 보존하기 위해 영구 스토리지에 데이터를 백업해야 합니다.

- `False` - (기본) 온라인 설치를 수행하려면 이 옵션을 선택한 채로 둡니다. 업그레이드 프로세스 도중에는 인스턴스가 재시작됩니다.

Important

온라인 및 오프라인 업그레이드에서 업그레이드 작업 전에 AMI가 생성됩니다. AMI는 자동화 완료 이후에도 지속됩니다. AMI에 대한 액세스에 보안 조치를 취하거나 더 이상 필요하지 않은 경우 삭제합니다.

SubnetId

(선택 사항) 다음 값 중 하나를 입력합니다.

- `SelectedInstanceSubnet` — (기본) 업그레이드 프로세스에서 도우미 인스턴스를 업그레이드해야 하는 인스턴스로 동일한 서브넷에서 시작합니다. 서브넷은 시스템 관리자 앤드포인트 (`ssm.*`)와의 통신을 허용해야 합니다.
- `CreateNewVPC` — 업그레이드 프로세스가 새 VPC에 도우미 인스턴스를 시작합니다. 대상 인스턴스의 서브넷이 `ssm.*` 앤드포인트와의 통신을 허용하는지 불확실한 경우 이 옵션을 사용합니다. IAM 사용자에게 VPC 생성 권한이 있어야 합니다.
- 특정 서브넷 ID — 도우미 인스턴스를 시작할 특정 서브넷의 ID를 지정합니다. 인스턴스를 업그레이드해야 하므로 서브넷은 동일한 가용 영역에 있어야 하고, `ssm.*` 앤드포인트와의 통신을 허용해야 합니다.

4. Execute automation(자동화 실행)을 선택합니다.
5. 업그레이드 완료를 허용합니다. 온라인 업그레이드를 완료하려면 최대 10분이 걸리고, 오프라인 업그레이드를 완료하려면 최대 25분이 걸릴 수 있습니다.

Microsoft SQL Server 데이터베이스용 Windows에서 Linux로의 리플랫포밍 어시스턴트

Microsoft SQL Server 데이터베이스 서비스를 위한 Windows에서 Linux로의 리플랫포밍 어시스턴트는 스크립팅 도구입니다. 기존 Microsoft SQL Server 워크로드를 Windows에서 Linux 운영 체제로 이전하는 데 도움이 됩니다. 이 리플랫포밍 어시스턴트는 클라우드 또는 온프레미스 환경에 호스팅되어 Microsoft SQL Server 2009 이상을 실행하는 모든 Windows Server 가상 머신(VM)에서 사용할 수 있습니다. 이 도구는 일반적인 비호환성을 검사하고, Windows VM에서 데이터베이스를 내보내고, Ubuntu 16.04에서 Microsoft SQL Server 2017를 실행하는 EC2 인스턴스로 가져옵니다. 자동화된 프로세스를 통해 실험 및 테스트에 사용할 수 있는, 선택한 SQL Server 데이터베이스로 구성된 즉시 사용할 수 있는 Linux VM이 생성됩니다.

목차

- [개념 \(p. 627\)](#)
- [관련 서비스 \(p. 628\)](#)
- [Microsoft SQL Server용 Windows에서 Linux로의 리플랫포밍 어시스턴트 작동 방식 \(p. 628\)](#)
- [구성 요소 \(p. 628\)](#)
- [설정 \(p. 628\)](#)
- [시작하기 \(p. 630\)](#)

개념

다음 용어 및 개념은 Microsoft SQL Server 데이터베이스 용 Windows에서 Linux로의 리플랫포밍 어시스턴트를 이해하고 사용하는 데 있어 핵심입니다.

백업

Microsoft SQL Server 백업은 데이터 또는 로그 레코드를 Microsoft SQL Server 데이터베이스 또는 트랜잭션 로그에서 디스크와 같은 백업 장치로 복사합니다. 자세한 내용은 [백업 개요\(Microsoft SQL Server\)](#)를 참조하십시오.

복원

일련의 Microsoft SQL Server 백업을 복원하기 위한 논리적이고 의미 있는 순서. 자세한 내용은 [복원 및 복구 개요\(Microsoft SQL Server\)](#)를 참조하십시오.

리플랫포밍

Microsoft SQL Server 데이터베이스는 EC2 Windows 인스턴스에서 Microsoft SQL Server를 실행하는 EC2 Linux 인스턴스로 리플랫폼될 수 있습니다. 또한 AWS에서 Microsoft SQL Server Linux를 실행하는 VMware Cloud로 리플랫폼될 수 있습니다.

관련 서비스

[AWS 시스템 관리자\(시스템 관리자\)](#)는 AWS 인프라에 대한 가시성과 제어를 제공합니다. Microsoft SQL Server 데이터베이스용 Windows에서 Linux로의 리플랫폼 어시스턴트는 EC2 Linux에서 Microsoft SQL 데이터베이스를 Microsoft SQL Server로 이전하는 데 시스템 관리자를 사용합니다. 시스템 관리자에 대한 자세한 내용은 [AWS 시스템 관리자 사용 설명서](#)를 참조하십시오.

Microsoft SQL Server용 Windows에서 Linux로의 리플랫포밍 어시스턴트 작동 방식

Microsoft SQL Server 데이터베이스를 위한 Windows에서 Linux로의 리플랫폼 어시스턴트 기능을 사용하면 온 프레미스 환경 또는 EC2 Windows 인스턴스에서 Microsoft SQL Server 데이터베이스를 백업 및 복원을 사용하여 EC2 Linux의 Microsoft SQL Server 2017로 마이그레이션할 수 있습니다. 대상 EC2 Linux 인스턴스의 경우, EC2 인스턴스 ID 또는 EC2 인스턴스 유형에 서브넷 ID 및 EC2 키 쌍을 제공합니다.

원본 Microsoft SQL Server 데이터베이스의 Microsoft SQL Server 데이터베이스를 위한 Windows에서 Linux로의 리플랫폼 어시스턴트용 PowerShell 스크립트를 실행하면 Windows 인스턴스는 암호화된 [Amazon Simple Storage Service\(S3\)](#) 스토리지 버킷에 데이터베이스를 백업합니다. 그런 다음 EC2 Linux 인스턴스에서 기존 Microsoft SQL Server로 백업을 복원하거나 EC2 Linux 인스턴스에서 새 Microsoft SQL Server를 시작하고 백업을 새로 생성한 인스턴스로 복원합니다. 이 프로세스는 엔터프라이즈 애플리케이션을 실행하는 티어 데이터베이스를 다시 구성하는 데 사용할 수 있습니다. 또한 원본 Microsoft SQL Server가 온라인 상태인 채로 Linux를 Microsoft SQL Server로 복제하여 애플리케이션을 테스트할 수 있습니다. 테스트가 끝나면 애플리케이션 중단 시간을 예약하고 최종 전환 기간 동안 PowerShell 백업 스크립트를 다시 실행할 수 있습니다.

전체 리플랫포밍 프로세스는 자동화하고 무인으로 실행할 수도 있습니다. PowerShell 백업 스크립트를 사용하지 않고 시스템 관리자 SSM 문서 [AWSEC2-SQLServerDBRestore](#)를 실행하여 기존 데이터베이스 백업 파일을 EC2 Linux의 Microsoft SQL Server로 가져올 수 있습니다.

구성 요소

Microsoft SQL Server 데이터베이스 스크립트를 위한 Windows에서 Linux로의 리플랫포밍 어시스턴트는 두 가지 주요 구성 요소로 이루어져 있습니다.

1. 온프레미스 Microsoft SQL Server 데이터베이스를 Amazon S3 스토리지 버킷에 백업하는 [AWS 서명 PowerShell 백업 스크립트](#). 그런 다음 SSM 자동화 문서 [AWSEC2-SQLServerDBRestore](#)를 호출하여 EC2 Linux 인스턴스에서 Microsoft SQL Server로 백업을 복원합니다.
2. AWSEC2-SQLServerDBRestore라는 SSM 자동화 문서는 EC2 Linux의 Microsoft SQL Server로 데이터베이스 백업을 복원합니다. 이 자동화는 Amazon S3에 저장된 Microsoft SQL Server 데이터베이스 백업을 EC2 Linux 인스턴스에서 실행되는 Microsoft SQL Server 2017로 복원합니다. Microsoft SQL Server 2017 Linux를 실행하는 EC2 인스턴스를 직접 제공하거나 자동화를 통해 Ubuntu 16.04에서 Microsoft SQL Server 2017을 사용하는 새 EC2 인스턴스를 시작 및 구성할 수 있습니다. 자동화는 전체, 차등 및 트랜잭션 로그 백업의 복원을 지원하고 여러 데이터베이스 백업 파일을 허용합니다. 자동화는 제공된 파일에서 각 데이터베이스의 최신 유효 백업을 자동으로 복원합니다. 자세한 내용은 [AWSEC2-SQLServerDBRestore](#)를 참조합니다.

설정

이 단원에서는 Windows에서 Linux로의 리플랫폼 스크립트를 실행하는 데 필수적인 단계에 대해 설명합니다.

목차

- [사전 조건 \(p. 629\)](#)
- [기존 EC2 인스턴스로의 리플랫포밍 전제 조건 \(p. 630\)](#)

사전 조건

Microsoft SQL Server Databases 스크립트를 위해 Windows에서 Linux로의 리플랫포밍 어시스턴트를 실행 하려면 다음을 수행해야 합니다.

1. AWS PowerShell 모듈 설치

AWS PowerShell 모듈을 설치하려면 [Windows 기반 컴퓨터에서 PowerShell 용 AWS 도구 설정](#)에 나열된 단계를 따르십시오. 백업 스크립트가 제대로 작동하려면 PowerShell 3.0 이상을 사용하는 것이 좋습니다.

2. Windows에서 Linux로의 리플랫포밍 어시스턴트 PowerShell 백업 스크립트 설치

Windows에서 Linux로의 플랫폼 변경 어시스턴트를 실행하려면 PowerShell 백업 스크립트 ([MigrateSQLServerToEC2Linux.ps1](#))를 다운로드하십시오.

3. AWS SDK 스토어에 AWS 사용자 프로필 추가

AWS 사용자 프로파일을 추가 및 구성하려면 AWS Tools for PowerShell 사용 설명서의 [프로필 관리](#)에 나열된 단계를 참조하십시오. 사용자 프로필에 대해 다음 [IAM 정책](#)을 설정합니다. 이러한 권한을 IAM 콘솔을 사용하여 AWS 사용자 계정 하위의 인라인 정책으로 추가할 수도 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RebootInstances",  
                "ec2:DescribeInstanceStatus",  
                "ec2:DescribeInstances",  
                "ec2>CreateTags",  
                "ec2:RunInstances",  
                "ec2:DescribeImages",  
                "iam:PassRole",  
                "ssm:StartAutomationExecution",  
                "ssm:DescribeInstanceInformation",  
                "ssm>ListCommandInvocations",  
                "ssm>ListCommands",  
                "ssm:SendCommand",  
                "ssm:GetAutomationExecution",  
                "ssm:GetCommandInvocation",  
                "s3:PutEncryptionConfiguration",  
                "s3>CreateBucket",  
                "s3>ListBucket",  
                "s3:PutObject",  
                "s3:GetObject",  
                "s3>DeleteObject",  
                "s3>DeleteBucket"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

4. IAM 인스턴스 프로파일 역할 생성

EC2 Linux에서 시스템 관리자를 실행하기 위해 IAM 인스턴스 프로파일 역할을 만들려면 AWS 시스템 관리자 사용 설명서에서 [시스템 관리자용 인스턴스 프로파일 만들기](#)에 나열된 단계를 참조하십시오.

기존 EC2 인스턴스로의 리플랫포밍 전제 조건

Linux에서 Microsoft SQL Server 2017을 실행하는 기존 인스턴스로 리플랫폼하려면 다음을 수행해야 합니다.

1. AWS Identity and Access Management(IAM) 인스턴스 프로파일을 사용하여 EC2 인스턴스를 구성하고 AmazonSSMManagedInstanceCore 관리되는 정책을 첨부하십시오.
시스템 관리자용 IAM 인스턴스 프로파일을 생성하고 인스턴스에 연결하는 방법에 대한 자세한 내용은 AWS 시스템 관리자 사용 설명서의 다음 주제를 참조하십시오.
 - [시스템 관리자에 대한 인스턴스 프로파일 생성](#)
 - [Amazon EC2 인스턴스에 IAM 인스턴스 프로파일 연결](#)
2. EC2 인스턴스에 SSM Agent가 설치되었는지 확인합니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [Windows 인스턴스에서 SSM 에이전트 설치 및 구성](#)을 참조하십시오.
3. EC2 인스턴스에 Microsoft SQL Server 백업을 다운로드하고 복원할 충분한 여유 디스크 공간이 있는지 확인하십시오.

시작하기

이 단원에는 PowerShell 매개 변수 정의와 데이터베이스를 리플랫폼하기 위한 스크립트가 포함되어 있습니다. PowerShell 스크립트를 사용하는 방법에 대한 자세한 내용은 [PowerShell](#)을 참조하십시오.

주제

- [Microsoft SQL Server용 Windows에서 Linux로의 리플랫포밍 스크립트 실행](#) (p. 630)
- [파라미터](#) (p. 631)

Microsoft SQL Server용 Windows에서 Linux로의 리플랫포밍 스크립트 실행

다음의 일반적인 시나리오 및 예제 PowerShell 스크립트는 Microsoft SQL Server 데이터베이스용 Windows에서 Linux로의 리플랫포밍 어시스턴트를 사용하여 Microsoft SQL Server 데이터베이스를 리플랫포밍하는 방법을 보여줍니다.

Important

Microsoft SQL Server 데이터베이스용 Windows에서 Linux로의 리플랫포밍 어시스턴트는 실행될 때마다 대상 인스턴스에서 SQL Server 서버 관리자(SA) 사용자 암호를 다시 설정합니다. 리플랫포밍 프로세스가 완료되면 대상 SQL Server 인스턴스에 연결하기 전에 고유한 SA 사용자 암호를 설정해야 합니다.

구문

Microsoft SQL Server 데이터베이스용 Windows에서 Linux로의 리플랫포밍 어시스턴트 스크립트는 다음 예제와 같은 구문을 따릅니다.

```
PS C:\> C:\MigrateSqlServerToEC2Linux.ps1 [[-SqlServerInstanceName] <String>] [[-  
DBNames]<Object[]>] [-  
MigrateAllDBs] [PathForBackup] <String> [-SetSourceDBModeReadOnly] [-  
IamInstanceProfileName] <String>[-  
AWSRegion] <String> [[-EC2InstanceId] <String>] [[-EC2InstanceType] <String>] [[-  
EC2KeyPair] <String>] [[-  
SubnetId] <String>] [[-AWSProfileName] <String>] [[-AWSProfileLocation] <String>] [-  
GeneratePresignedUrls]  
[<CommonParameters>]
```

예제 1: 데이터베이스를 EC2 인스턴스로 이동

다음 예제는 AdventureDB라는 이름의 데이터베이스를 MSSQLSERVER라는 이름의 Microsoft SQL Server 인스턴스에서 인스턴스 ID가 i-024689abcdef인 Linux 인스턴스의 EC2 Microsoft SQL Server로 이전하는 방법을 보여줍니다. 사용할 백업 디렉터리는 D:\\Backup이며 AWS 지역은 us-east-2입니다.

```
PS C:\> ./MigrateSQLServerToEC2Linux.ps1 - SQLServerInstanceName MSSQLSERVER -  
EC2InstanceId i-024689abcdef -DBNames AdventureDB -PathForBackup D:\\Backup -AWSRegion us-east-2 -  
AWSProfileName DBMigration -IamInstanceProfileName AmazonSSMManagedInstanceCore
```

예제 2: AWS 자격 증명 프로파일을 사용하여 데이터베이스를 EC2 인스턴스로 이동

다음 예제에서는 AWS 자격 증명 프로파일을 사용하여 예제 1의 데이터베이스를 이동하는 방법을 보여줍니다: DBMigration.

```
PS C:\> ./MigrateSQLServerToEC2Linux.ps1 - SQLServerInstanceName MSSQLSERVER -  
EC2InstanceId i-024689abcdef -DBNames AdventureDB -PathForBackup D:\\Backup -AWSRegion us-east-2 -  
AWSProfileName DBMigration -IamInstanceProfileName AmazonSSMManagedInstanceCore
```

예제 3: 데이터베이스를 새로운 m5.large 유형 인스턴스로 이동

다음 예제는 customer-ec2-keypair 키 쌍을 사용하여 subnet-abc127에서 m5.large 유형 EC2 Linux 인스턴스를 작성한 다음 AdventureDB, TestDB를 예제 1 및 2에서 사용된 데이터베이스에서 새 인스턴스로 이동하는 방법을 보여줍니다.

```
PS C:\> ./MigrateSQLServerToEC2Linux.ps1 -EC2InstanceType m5.large -SubnetId subnet-abc127  
-EC2KeyPair customer-ec2-keypair -DBNames AdventureDB,TestDB -PathForBackup D:\\Backup -AWSRegion us-east-2 -  
AWSProfileName DBMigration -IamInstanceProfileName AmazonSSMManagedInstanceCore
```

예제 4: 모든 데이터베이스를 새로운 m5.large 유형 인스턴스로 이동

다음 예제는 customer-ec2-keypair 키 쌍을 사용하여 subnet-abc127에서 m5.large 유형 EC2 Linux 인스턴스를 작성한 다음 모든 데이터베이스를 예제 1 및 2에서 사용된 데이터베이스에서 새 인스턴스로 이동하는 방법을 보여줍니다.

```
PS C:\> ./MigrateSQLServerToEC2Linux.ps1 -EC2InstanceType m5.large -SubnetId subnet-abc127  
-EC2KeyPair customer-ec2-keypair -MigrateAllDBs -PathForBackup D:\\Backup -AWSRegion us-east-2 -  
AWSProfileName DBMigration -IamInstanceProfileName AmazonSSMManagedInstanceCore
```

파라미터

다음 파라미터는 PowerShell 스크립트가 Microsoft SQL Server 데이터베이스를 리플랫폼하는 데 사용됩니다.

-SqlServerInstanceName

백업할 Microsoft SQL Server 인스턴스의 이름. SqlServerInstanceName 값이 제공되지 않으면 기본적으로 \$env:ComputerName이 사용됩니다.

형식: 문자열

필수 항목 여부: 아니요

-DBNames

백업 및 복원할 데이터베이스의 이름. 쉼표로 구분된 목록(예: adventureDB, universityDB)으로 데이터베이스의 이름을 지정하십시오. DBNames 또는 MigrateAllDBs 파라미터가 필요합니다.

유형: 객체

필수 항목 여부: 아니요

-MigrateAllDBs

이 스위치는 기본적으로 비활성화되어 있습니다. 이 스위치를 활성화하면 자동화가 시스템 데이터베이스(master, msdb, tempdb)를 제외한 모든 데이터베이스를 마이그레이션합니다. DBNames 또는 MigrateAllDBs 파라미터가 필요합니다.

유형: SwitchParameter

필수 여부: 아니요

-PathForBackup

전체 백업이 저장되는 경로입니다.

형식: 문자열

필수 항목 여부: 예

-SetSourceDBModeReadOnly

이 스위치는 기본적으로 비활성화되어 있습니다. 이 스위치를 활성화하면 마이그레이션 중에 데이터베이스는 읽기 전용 상태가 됩니다.

유형: SwitchParameter

필수 여부: 아니요

-IamInstanceProfileName

사용자를 대신하여 시스템 관리자 자동화를 실행할 권한이 있는 AWS IAM 인스턴스 역할을 입력하십시오. AWS 시스템 관리자 사용 설명서에서 [자동화 시작하기](#)를 참조합니다.

형식: 문자열

필수 항목 여부: 예

-AWSRegion

Amazon S3 버킷이 생성되어 데이터베이스 백업을 저장할 AWS 지역을 입력하십시오.

형식: 문자열

필수 항목 여부: 예

-EC2InstanceId

Microsoft SQL Server 데이터베이스를 Microsoft SQL Server Linux를 실행하는 기존 EC2 인스턴스로 복원하려면 인스턴스의 인스턴스 ID를 입력하십시오. EC2 인스턴스에 이미 AWS 시스템 관리자 SSM 에이전트가 설치되어 실행 중인지 확인하십시오.

형식: 문자열

필수 항목 여부: 아니요

-EC2InstanceType

Microsoft SQL Server 데이터베이스를 새 EC2 Linux 인스턴스로 복원하려면 시작될 인스턴스의 인스턴스 유형을 입력하십시오.

형식: 문자열

필수 항목 여부: 아니요

-EC2KeyPair

Microsoft SQL Server 데이터베이스를 새 EC2 Linux 인스턴스로 복원하려면 인스턴스에 액세스하는 데 사용할 EC2 키 쌍의 이름을 입력하십시오. 이 파라미터는 새 EC2 Linux 인스턴스를 작성하는 경우 권장됩니다.

형식: 문자열

필수 항목 여부: 아니요

-SubnetId

이 파라미터는 새 EC2 Linux 인스턴스를 작성하는 경우 필수입니다. 새 EC2 Linux 인스턴스를 작성할 때 SubnetId를 제공하지 않으면 AWS 사용자 기본 서브넷을 사용하여 EC2 Linux 인스턴스를 시작합니다.

형식: 문자열

필수 항목 여부: 아니요

-AWSProfileName

AWS 서비스에 연결할 때 자동화가 사용하는 AWS 프로파일의 이름입니다. 필요한 IAM 사용자 권한에 대한 자세한 내용은 AWS 시스템 관리자 사용 설명서에서 [자동화 시작하기](#)를 참조하십시오. 프로파일을 입력하지 않으면 자동화는 기본 AWS 프로파일을 사용합니다.

형식: 문자열

필수 항목 여부: 아니요

-AWSProfileLocation

AWS 프로파일이 기본 위치에 저장되지 않은 경우 AWS 프로파일의 위치입니다.

형식: 문자열

필수 항목 여부: 아니요

-GeneratePresignedUrls

이 파라미터는 비EC2 인스턴스(예: AWS의 VMware Cloud 또는 온프레미스 VM)로 리플랫포밍할 때만 사용됩니다.

유형: SwitchParameter

필수 여부: 아니요

<CommonParameters>

이 cmdlet은 일반 파라미터를 지원합니다: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, OutVariable. 자세한 내용은 Microsoft PowerShell 설명서의 [공통 파라미터 정보](#)를 참조하십시오.

필수 항목 여부: 아니요

업그레이드 문제 해결

AWS는 Citrix PV 드라이버가 관련된 현재 위치 업그레이드를 수행하는 사용자를 돋는 AWS 유틸리티인 업그레이드 헬퍼 서비스를 통해 업그레이드 문제에 대한 지원을 제공합니다.

업그레이드 후 인스턴스에서 .NET 런타임 최적화 서비스가 .NET 프레임워크를 최적화하는 동안 일시적으로 CPU 사용률이 평균보다 높게 나타날 수 있습니다. 이는 예상된 동작입니다.

몇 시간 후에도 인스턴스가 두 상태 확인을 모두 통과하지 못한 경우에는 다음을 점검하십시오.

- Windows Server 2008로 업그레이드한 후 몇 시간이 지나도 두 상태 확인에 실패하는 경우 업그레이드되지 않고 확인 클릭(Click OK)을 수행하여 룰백을 확인하라는 메시지가 표시될 수 있습니다. 이 상태에서는 콘솔에 액세스할 수 없기 때문에 버튼을 클릭할 수 있는 방법이 없습니다. 이 문제를 해결하려면 Amazon EC2 콘솔 또는 API를 통해 재부팅을 수행하십시오. 재부팅은 시작하는 데 10분 이상 걸리고, 25분이 지나면 인스턴스를 사용할 수 있습니다.
- 서버에서 애플리케이션 또는 서버 역할을 제거하고 다시 시도합니다.

서버에서 애플리케이션 또는 서버 역할을 제거한 후에도 인스턴스가 두 상태 확인을 통과하지 못하는 경우 다음을 수행하십시오.

- 인스턴스를 중지하고 루트 볼륨을 다른 인스턴스에 연결합니다. 자세한 내용은 "[Waiting for the metadata service](#)" (p. 1223)에서 루트 볼륨을 중지하고 다른 인스턴스에 연결하는 방법에 대한 설명을 참조하십시오.
- Windows 설치 로그 파일 및 장애에 대한 이벤트 로그를 분석합니다.

운영 체제 업그레이드 또는 마이그레이션과 관련된 기타 모든 문제에 대해서는 [인플레이스\(In-Place\) 업그레이드를 시작하기 전에](#) (p. 612)에 수록된 도움말을 확인하는 것이 좋습니다.

EC2 Windows 인스턴스 식별

애플리케이션이 EC2 인스턴스에서 실행 중인지 여부를 확인해야 할 수 있습니다.

Linux 인스턴스 식별에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [EC2 Linux 인스턴스 식별](#) 단원을 참조하십시오.

인스턴스 자격 증명 문서 검사

EC2 인스턴스를 식별하는 명확하고 암호화된 방법은 서명이 포함된 인스턴스 자격 증명 문서를 확인하십시오. 이러한 문서는 라우팅할 수 없는 로컬 주소 <http://169.254.169.254/latest/dynamic/instance-identity/>에서 모든 EC2 인스턴스에 대해 제공됩니다. 자세한 정보는 [인스턴스 자격 증명 문서](#) (p. 597) 단원을 참조하십시오.

시스템 UUID 검사

시스템 UUID를 가져와 UUID의 시작 8진수에 문자 "EC2"가 있는지 찾을 수 있습니다. 시스템이 EC2 인스턴스인지 확인하는 이 방법은 빠르지만 EC2 인스턴스가 아닌 시스템에도 이러한 문자로 시작하는 UUID가 있을 가능성이 약간이라도 있기 때문에 부정확할 수 있습니다. 또한 SMBIOS 2.4를 사용하는 EC2 인스턴스은 리틀 엔디안 형식으로 UUID를 나타낼 수 있으므로 "EC2" 문자는 EC2 시작 부분에 나타나지 않습니다.

Example : WMI 또는 Windows PowerShell을 사용하여 UUID 가져오기

다음과 같이 Windows Management Instrumentation 명령줄(WMIC)을 사용합니다.

```
wmic path win32_computersystemproduct get uuid
```

또는 Windows PowerShell을 사용하는 경우 다음과 같이 Get-WmiObject cmdlet을 사용합니다.

```
PS C:\> Get-WmiObject -query "select uuid from Win32_ComputerSystemProduct" | Select UUID
```

다음 출력 예에서 UUID가 "EC2"로 시작합니다. 이는 시스템이 아마도 EC2 인스턴스라는 것을 나타냅니다.

```
EC2AE145-D1DC-13B2-94ED-01234ABCDEF
```

SMBIOS 2.4를 사용하는 인스턴스의 경우 UUID를 리틀 엔디안 형식으로 나타낼 수 있습니다. 예를 들면 다음과 같습니다.

```
45E12AEC-DCD1-B213-94ED-01234ABCDEF
```

Amazon Elastic Graphics

Amazon Elastic Graphics는 Windows 인스턴스를 위한 유연하고 비용이 저렴한 고성능 그래픽 가속화 기능을 제공합니다. Elastic Graphics 액셀러레이터는 다양한 크기로 제공되며 GPU 그래픽 인스턴스 유형(예: G2 및 G3) 사용에 대한 저가형 대안입니다. 애플리케이션의 컴퓨팅, 메모리 및 스토리지 요구를 충족하는 인스턴스 유형을 선택할 수 있는 유연성을 얻을 수 있습니다. 그런 다음 워크로드의 그래픽 요구 사항을 충족하는 인스턴스용 액셀러레이터를 선택합니다.

Elastic Graphics는 그래픽 성능을 높이기 위해 소량의 GPU나 비정기적으로 GPU가 추가로 필요하며, OpenGL 그래픽 지원을 사용하는 애플리케이션에 적합합니다. 직접 연결된 Full GPU에 액세스해야 하거나, DirectX, CUDA, OpenCL(Open Computing Language)을 사용해야 할 경우 액셀러레이티드 컴퓨팅 인스턴스 유형 인스턴스를 대신 사용하십시오. 자세한 내용은 [Windows 액셀러레이티드 컴퓨팅 인스턴스 \(p. 182\)](#) 단원을 참조하십시오.

내용

- [Elastic Graphics 기본 사항 \(p. 636\)](#)
- [Elastic Graphics 가격 \(p. 638\)](#)
- [Elastic Graphics 제한 사항 \(p. 638\)](#)
- [Elastic Graphics 작업 \(p. 638\)](#)
- [CloudWatch 지표를 사용하여 Elastic Graphics 모니터링 \(p. 643\)](#)
- [문제 해결 \(p. 645\)](#)

Elastic Graphics 기본 사항

Elastic Graphics를 사용하려면 Windows 인스턴스를 시작하고 시작 중에 인스턴스에 대한 액셀러레이터 유형을 지정합니다. AWS는 사용 Elastic Graphics 용량을 찾고 인스턴스와 Elastic Graphics 액셀러레이터 간의 네트워크 연결을 설정합니다.

Note

베어 메탈 인스턴스는 지원되지 않습니다.

Elastic Graphics 액셀러레이터는 us-east-1, us-east-2, us-west-2, ap-northeast-1, ap-southeast-1, ap-southeast-2, eu-central-1, eu-west-1의 AWS 리전에서 사용할 수 있습니다.

다음 인스턴스 유형에서는 Elastic Graphics 액셀러레이터를 지원합니다.

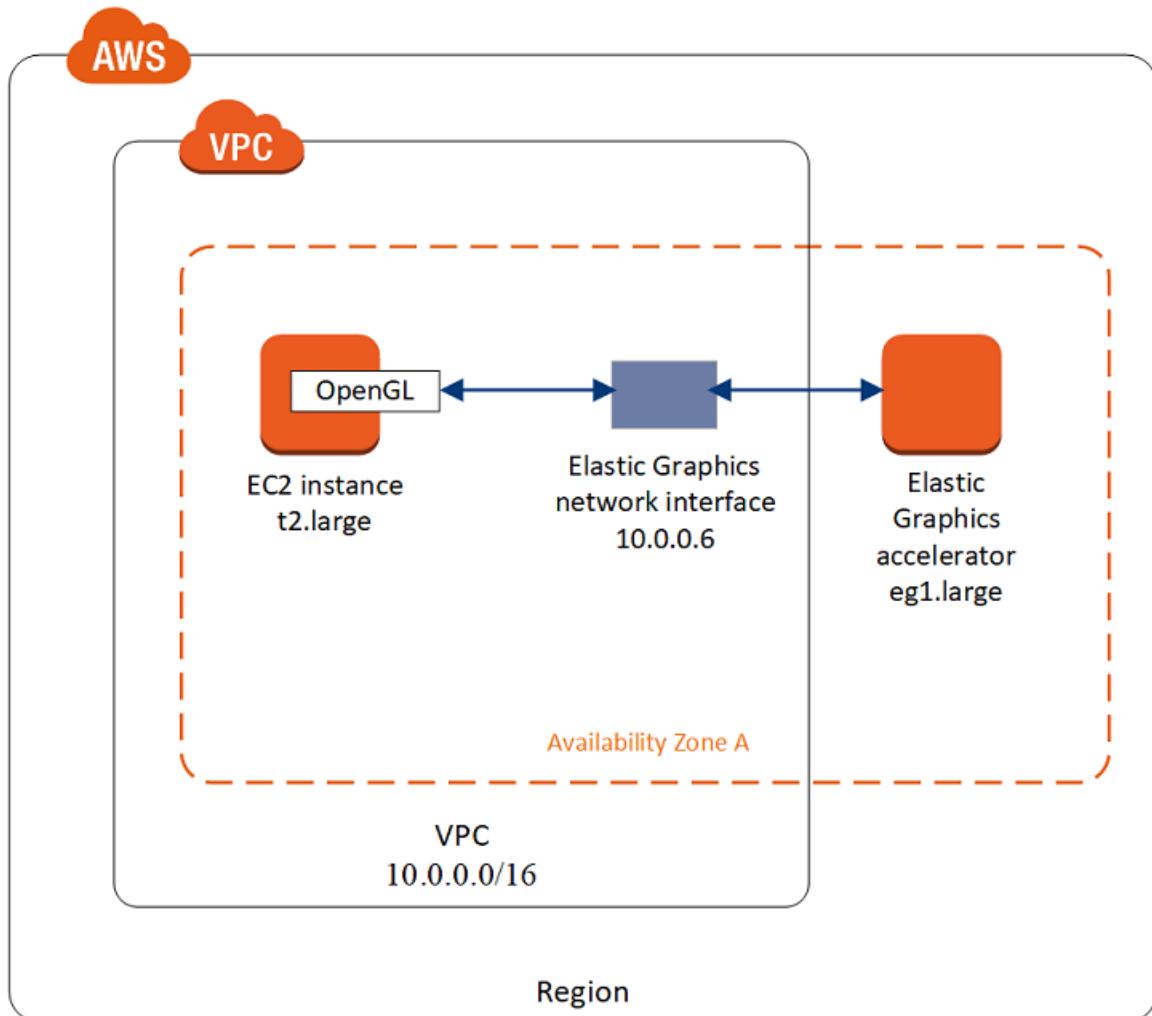
- C3 | C4 | C5 | C5a | C5d | C5n
- D2
- H1
- I3 | I3en
- M3 | M4 | M5 | M5d | M5dn | M5n
- P2 | P3 | P3dn
- R3 | R4 | R5 | R5d | R5dn | R5n
- t2.medium 이상 | t3.medium 이상
- X1 | X1e
- z1d

다음 Elastic Graphics 액셀러레이터를 사용할 수 있습니다. Elastic Graphics 액셀러레이터를 지원되는 인스턴스 유형에 연결할 수 있습니다.

Elastic Graphics 액셀러레이터	그래픽 메모리(GB)
eg1.medium	1
eg1.large	2
eg1.xlarge	4
eg1.2xlarge	8

Elastic Graphics 액셀러레이터는 인스턴스 하드웨어에 포함되지 않습니다. 대신에 Elastic Graphics 네트워크 인터페이스라고 하는 네트워크 인터페이스를 통해 네트워크에 연결됩니다. 그래픽 가속화 기능으로 인스턴스를 시작할 경우 Elastic Graphics 네트워크 인터페이스가 VPC에서 자동으로 생성됩니다.

Elastic Graphics 네트워크 인터페이스는 인스턴스와 동일한 VPC와 서브넷에 생성되며, 해당 서브넷의 프라이빗 IPv4 주소를 할당받습니다. Amazon EC2 인스턴스에 연결된 액셀러레이터는 인스턴스와 동일한 가용 영역의 사용 가능한 액셀러레이터 풀에서 할당됩니다.



Elastic Graphics 액셀러레이터는 OpenGL 4.3 이하 API의 API 표준을 지원합니다. 이러한 표준은 배치 애플리케이션 및 3D 그래픽 가속화용으로 사용할 수 있습니다. 인스턴스에 대한 Amazon 최적화 OpenGL 라이브러리는 연결된 액셀러레이터를 감지합니다. 그리고 인스턴스의 OpenGL API 호출을 액셀러레이터로 보냅니다.

후 요청을 처리하고 결과를 반환합니다. 인스턴스와 액셀러레이터 간의 트래픽은 인스턴스 네트워크 트래픽과 동일한 대역폭을 사용하므로 사용 가능한 네트워크 대역폭이 충분하도록 해야 합니다. OpenGL 보안 및 규정 준수와 버전은 소프트웨어 공급업체에 문의하십시오.

기본적으로 VPC의 기본 보안 그룹은 Elastic Graphics 네트워크 인터페이스와 연결됩니다. Elastic Graphics 네트워크 트래픽은 TCP 프로토콜과 포트 2007을 사용합니다. 인스턴스의 보안 그룹이 이를 허용하는지 확인합니다. 자세한 내용은 [보안 그룹 구성 \(p. 639\)](#) 단원을 참조하십시오.

Elastic Graphics 가격

Elastic Graphics 액셀러레이터가 `ok` 상태일 때 액셀러레이터가 `running` 상태인 인스턴스에 연결되어 있는 매초마다 요금이 부과됩니다. 상태가 `pending`, `stopping`, `stopped`, `shutting-down` 또는 `terminated`인 인스턴스에 연결된 액셀러레이터에 대해서는 요금이 부과되지 않습니다. 액셀러레이터가 `Unknown` 또는 `Impaired` 상태일 때도 요금이 부과되지 않습니다.

액셀러레이터의 요금은 온디맨드 요율로만 가능합니다. 예약 인스턴스, 예약된 인스턴스 또는 스팟 인스턴스에 액셀러레이터를 연결할 수 있습니다. 액셀러레이터에 대한 온디맨드 요금이 항상 적용됩니다.

자세한 내용은 [Amazon Elastic Graphics 요금](#)을 참조하십시오.

Elastic Graphics 제한 사항

Elastic Graphics 액셀러레이터 사용을 시작하기 전에 다음과 같은 제한에 유의하십시오.

- Microsoft Windows Server 2012 R2 이상이 설치된 Windows 인스턴스에만 액셀러레이터를 연결할 수 있습니다. 현재 Linux 인스턴스는 지원되지 않습니다.
- 한 번에 액셀러레이터 하나를 인스턴스에 연결할 수 있습니다.
- 인스턴스 시작 시에만 액셀러레이터를 연결할 수 있습니다. 기존 인스턴스에 액셀러레이터를 연결할 수 없습니다.
- 인스턴스 간에 액셀러레이터를 공유할 수 없습니다.
- 액셀러레이터를 인스턴스로부터 분리하거나 다른 인스턴스로 이전할 수 없습니다. 액셀러레이터가 더 이상 필요하지 않은 경우 해당 인스턴스를 종료해야 합니다. 액셀러레이터 유형을 변경하려면 인스턴스에서 AMI를 생성하여 인스턴스를 종료한 후 다른 액셀러레이터를 지정하여 새로운 인스턴스를 시작합니다.
- OpenGL API 4.3 이하 버전만 지원됩니다. DirectX, CUDA, OpenCL은 지원되지 않습니다.
- Elastic Graphics 액셀러레이터는 인스턴스의 디바이스 관리자를 통해 보거나 액세스할 수 없습니다.
- 액셀러레이터 용량을 예약하거나 계획할 수 없습니다.
- 액셀러레이터를 EC2-Classic의 인스턴스에는 연결할 수 없습니다.
- 인스턴스 메타데이터 서비스 v2(IMDSv2)를 사용하도록 구성된 인스턴스에는 액셀러레이터를 연결할 수 없습니다.

Elastic Graphics 작업

인스턴스를 시작하고, 시작 시 이 인스턴스를 Elastic Graphics 액셀러레이터와 연결할 수 있습니다. 그런 다음 액셀러레이터와 통신하는 데 필요한 라이브러리를 인스턴스에 직접 설치해야 합니다. 제한 사항은 [Elastic Graphics 제한 사항 \(p. 638\)](#) 단원을 참조하십시오.

작업

- [보안 그룹 구성 \(p. 639\)](#)
- [Elastic Graphics 액셀러레이터로 인스턴스 시작 \(p. 639\)](#)

- [Elastic Graphics용 필요 소프트웨어 설치 \(p. 640\)](#)
- [인스턴스에서 Elastic Graphics 기능 확인 \(p. 640\)](#)
- [Elastic Graphics 정보 보기 \(p. 642\)](#)
- [피드백 제출 \(p. 643\)](#)

보안 그룹 구성

Amazon EC2 콘솔을 사용하여 Elastic Graphics 액셀러레이터로 인스턴스를 시작하고 보안 그룹을 자동으로 생성할 경우 콘솔에서 Elastic Graphics 포트에 대한 트래픽을 허용하는 데 필요한 인바운드 및 아웃바운드 규칙을 추가합니다. AWS CLI 또는 SDK를 사용하여 인스턴스를 시작하려는 경우 보안 그룹에서 Elastic Graphics 포트에 대한 트래픽을 허용하는지 확인해야 합니다.

Elastic Graphics에 대한 보안 그룹을 생성하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹과 보안 그룹 생성을 차례로 선택합니다.
3. "Elastic Graphics 보안 그룹"과 같은 보안 그룹 이름과, 보안 그룹에 대한 설명을 지정합니다. Elastic Graphics 액셀러레이터로 인스턴스를 시작하는 데 사용할 VPC를 선택합니다.
4. 다음과 같이 인바운드 보안 그룹 규칙을 생성합니다.
 - a. 인바운드 탭에서 규칙 추가를 선택합니다.
 - b. 유형으로 Elastic Graphics를 선택합니다. 소스로 사용자 지정을 선택하고 보안 그룹의 ID를 입력합니다.
5. 다음과 같이 아웃바운드 보안 그룹 규칙을 생성합니다.
 - a. 아웃바운드 탭에서 규칙 추가를 선택합니다.
 - b. 유형으로 모든 TCP를 선택합니다. 대상으로 사용자 지정을 선택하고 보안 그룹의 ID를 입력합니다.
6. Create를 선택합니다.

자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#) 단원을 참조하십시오.

Elastic Graphics 액셀러레이터로 인스턴스 시작

시작 시 Elastic Graphics 액셀러레이터를 인스턴스에 연결할 수 있습니다. 시작이 실패하면 다음 이유 중 하나 때문일 수 있습니다.

- Elastic Graphics 액셀러레이터 용량 부족
- 리전의 Elastic Graphics 액셀러레이터 제한 초과
- VPC의 프라이빗 IPv4 주소가 액셀러레이터에 대한 네트워크 인터페이스를 생성하기에 부족함

자세한 내용은 [Elastic Graphics 제한 사항 \(p. 638\)](#) 단원을 참조하십시오.

인스턴스 시작 시 Elastic Graphics 액셀러레이터를 연결하는 방법(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 대시보드에서 인스턴스 시작을 선택합니다.
3. Windows AMI와 지원되는 인스턴스 유형을 선택합니다. 자세한 내용은 [Elastic Graphics 기본 사항 \(p. 636\)](#) 단원을 참조하십시오.
4. 인스턴스 세부 정보 구성 페이지에서 인스턴스를 시작할 VPC와 서브넷을 선택합니다.
5. 그래픽 가속화 추가를 선택하고 Elastic Graphics 액셀러레이터 유형을 선택합니다.

6. (선택 사항) 스토리지 추가와 태그 추가 페이지에서 필요에 따라 볼륨 및 태그를 추가합니다.
7. 보안 그룹 구성 페이지에서 콘솔을 통해 필요한 인바운드 및 아웃바운드 규칙을 사용하여 보안 그룹이 자동으로 생성되도록 하거나 [보안 그룹 구성 \(p. 639\)](#)에서 수동으로 생성한 보안 그룹을 사용할 수 있습니다. 필요에 따라 보안 그룹을 더 추가합니다.
8. 검토 및 시작을 선택하여 인스턴스 옵션을 검토하고 나서 시작을 선택합니다.

인스턴스 시작 시 Elastic Graphics 액셀러레이터를 연결하는 방법(AWS CLI)

이 경우 [run-instances](#) AWS CLI 명령을 다음 파라미터와 함께 사용할 수 있습니다.

```
--elastic-gpu-specification Type=eg1.medium
```

--security-group-ids 파라미터의 경우 필요한 인바운드 및 아웃바운드 규칙이 있는 보안 그룹을 포함해야 합니다. 자세한 내용은 [보안 그룹 구성 \(p. 639\)](#) 단원을 참조하십시오.

인스턴스 시작 시 Elastic Graphics 액셀러레이터를 연결하는 방법(Windows PowerShell용 도구)

[New-EC2Instance](#) Windows PowerShell용 도구 명령을 사용합니다.

Elastic Graphics용 필요 소프트웨어 설치

현재 AWS Windows AMI를 사용하여 인스턴스를 시작한 경우 처음 부팅하는 동안 필요 소프트웨어가 자동으로 설치됩니다. 필요 소프트웨어를 자동으로 설치하지 않는 Windows AMI를 사용하여 인스턴스를 시작한 경우 먼저 인스턴스에 필요 소프트웨어를 설치해야 합니다.

Elastic Graphics용 필요 소프트웨어를 설치하는 방법(필요 시)

1. 인스턴스에 연결합니다.
2. [Elastic Graphics 설치 관리자를 다운로드하여 업니다.](#) 설치 관리자가 Elastic Graphics 엔드포인트에 연결하여 필요한 소프트웨어의 최신 버전을 다운로드합니다.
3. 인스턴스를 재부팅하여 확인합니다.

인스턴스에서 Elastic Graphics 기능 확인

인스턴스의 Elastic Graphics 패키지에는 액셀러레이터의 상태를 보고, 액셀러레이터에 대한 인스턴스에서 OpenGL 명령이 작동하는지 확인하는 데 사용할 수 있는 도구가 포함되어 있습니다.

Elastic Graphics 패키지가 설치되어 있지 않은 AMI로 인스턴스를 시작한 경우 직접 다운로드하여 설치하면 됩니다. 자세한 내용은 [Elastic Graphics용 필요 소프트웨어 설치 \(p. 640\)](#) 단원을 참조하십시오.

내용

- [Elastic Graphics 상태 모니터 사용 \(p. 640\)](#)
- [Elastic Graphics 명령줄 도구 사용 \(p. 641\)](#)

Elastic Graphics 상태 모니터 사용

상태 모니터 도구를 사용하여 연결된 Elastic Graphics 액셀러레이터의 상태에 대한 정보를 볼 수 있습니다. 기본적으로 이 도구는 Windows 인스턴스의 작업 표시줄에 있는 알림 영역에서 사용할 수 있으며, 그래픽 액셀러레이터의 상태를 표시합니다. 사용 가능한 값은 다음과 같습니다.

Healthy

Elastic Graphics 액셀러레이터가 활성화되어 정상적으로 작동하고 있습니다.

Updating

Elastic Graphics 액셀러레이터의 상태를 현재 업데이트 중입니다. 상태를 표시하는 몇 분이 걸릴 수도 있습니다.

Out of service

Elastic Graphics 액셀러레이터가 서비스 불가능 상태입니다. 오류에 대한 자세한 정보를 확인하려면 더 보기를 선택합니다.

Elastic Graphics 명령줄 도구 사용

Elastic Graphics 명령줄 도구인 egcli.exe를 사용하여 액셀러레이터의 상태를 확인할 수 있습니다. 액셀러레이터에 문제가 있는 경우 오류 메시지가 반환됩니다.

도구를 시작하려면 인스턴스 내에서 명령 프롬프트를 열고 다음 명령을 실행합니다.

```
C:\Program Files\Amazon\EC2ElasticGPUs\manager\egcli.exe
```

이 도구는 다음 파라미터도 지원합니다.

--json, -j

JSON 메시지 표시 여부를 나타냅니다. 가능한 값은 true과 false입니다. 기본값은 true입니다.

--imds, -i

인스턴스 메타데이터의 액셀러레이터 가용성 확인 여부를 나타냅니다. 가능한 값은 true과 false입니다. 기본값은 true입니다.

다음은 예제 출력입니다. OK 상태는 액셀러레이터가 활성화되었고 정상 상태임을 나타냅니다.

```
EG Infrastructure is available.  
Instance ID egpu-f6d94dfa66df4883b284e96db7397ee6  
Instance Type eg1.large  
EG Version 1.0.0.885 (Manager) / 1.0.0.95 (OpenGL Library) / 1.0.0.69 (OpenGL Redirector)  
EG Status: Healthy  
JSON Message:  
{  
    "version": "2016-11-30",  
    "status": "OK"  
}
```

사용 가능한 status 값은 다음과 같습니다.

OK

Elastic Graphics 액셀러레이터가 활성화되어 정상적으로 작동하고 있습니다.

UPDATING

Elastic Graphics 드라이버를 업데이트 중입니다.

NEEDS_REBOOT

Elastic Graphics 드라이버를 업데이트했으며 Amazon EC2 인스턴스를 재부팅해야 합니다.

LOADING_DRIVER

Elastic Graphics 드라이버를 로드 중입니다.

CONNECTING_EGPU

Elastic Graphics 드라이버에서 Elastic Graphics 액셀러레이터와의 연결을 확인하는 중입니다.

ERROR_UPDATE_RETRY

Elastic Graphics 드라이버를 업데이트하는 동안 오류가 발생했으며, 업데이트가 곧 다시 시도됩니다.

ERROR_UPDATE

Elastic Graphics 드라이버를 업데이트하는 동안 복구할 수 없는 오류가 발생했습니다.

ERROR_LOAD_DRIVER

Elastic Graphics 드라이버를 로드하는 동안 오류가 발생했습니다.

ERROR_EGPU_CONNECTIVITY

Elastic Graphics 액셀러레이터가 연결할 수 없습니다.

Elastic Graphics 정보 보기

인스턴스에 연결된 Elastic Graphics 액셀러레이터에 대한 정보를 볼 수 있습니다.

Elastic Graphics 액셀러레이터에 대한 정보를 보는 방법(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
3. 설명 탭에서 Elastic Graphics ID를 찾습니다. Elastic Graphics 액셀러레이터에 대한 다음 정보를 확인할 ID를 선택합니다.
 - 연결 상태
 - 유형
 - 상태 확인

Elastic Graphics 액셀러레이터에 대한 정보를 보는 방법(AWS CLI)

[describe-elastic-gpus](#) AWS CLI 명령을 사용합니다.

```
aws ec2 describe-elastic-gpus
```

[describe-network-interfaces](#) AWS CLI 명령을 사용하고 소유자 ID로 필터링하여 Elastic Graphics 네트워크 인터페이스에 대한 정보를 볼 수 있습니다.

```
aws ec2 describe-network-interfaces --filters "Name=attachment.instance-owner-id,Values=amazon-elasticgpus"
```

Elastic Graphics 액셀러레이터에 대한 정보를 보는 방법(Windows PowerShell용 도구)

다음 명령을 사용합니다.

- [Get-EC2ElasticGpu](#)
- [Get-EC2NetworkInterface](#)

인스턴스 메타데이터를 사용하여 Elastic Graphics 액셀러레이터에 대한 정보를 보는 방법

1. Elastic Graphics 액셀러레이터를 사용하는 Windows 인스턴스에 연결합니다.
2. 다음 중 하나를 수행하십시오.
 - PowerShell에서 다음 cmdlet을 사용합니다.

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/elastic-gpus/associations/egpu-f6d94dfa66df4883b284e96db7397ee6
```

- 웹 브라우저에서 다음 URL을 주소 필드에 붙여 넣습니다.

```
http://169.254.169.254/latest/meta-data/elastic-gpus/associations/egpu-f6d94dfa66df4883b284e96db7397ee6
```

피드백 제출

다음 단계를 통해 Elastic Graphics 사용에 대한 피드백 제출하여 서비스 개선에 참여하실 수 있습니다.

Elastic Graphics 상태 모니터를 사용하여 피드백을 제출하는 방법

1. Windows 인스턴스 작업 표시줄의 알림 영역에서 Elastic Graphics 상태 모니터를 엽니다.
2. 하단 왼쪽 모서리 부분에서 피드백을 선택합니다.
3. 의견을 입력하고 제출을 선택합니다.

CloudWatch 지표를 사용하여 Elastic Graphics 모니터링

액셀러레이터 성능에 대한 지표를 수집하는 Amazon CloudWatch를 사용하여 Elastic Graphics 액셀러레이터를 모니터링할 수 있습니다. 이러한 통계는 2주간 기록되므로 기록 정보를 보고 웹 애플리케이션이나 서비스가 어떻게 실행되고 있는지 전체적으로 더 잘 파악할 수 있습니다.

기본적으로 Elastic Graphics 액셀러레이터는 5분 동안 지표 데이터를 CloudWatch로 전송합니다.

Amazon CloudWatch에 대한 자세한 내용은 [Amazon CloudWatch 사용 설명서](#)를 참조하십시오.

Elastic Graphics 지표

AWS/ElasticGPUs 네임스페이스에는 Elastic Graphics에 대한 다음 지표가 포함됩니다.

지표	설명
GPUConnectivityCheckFailed	Elastic Graphics 액셀러레이터와의 연결이 활성 상태인지 실패했는지 여부를 보고합니다. 값이 0이면 연결이 유효함을 나타냅니다. 값이 1이면 연결이 실패했음을 나타냅니다. 단위: 개수
GPUHealthCheckFailed	Elastic Graphics 액셀러레이터에서 마지막 1분 동안 상태 확인을 통과했는지 보고합니다. 값이 0이면 상태 확인을 통과한 것이고, 값이 1이면 상태 확인이 실패했음을 나타냅니다. 단위: 개수
GPUMemoryUtilization	사용된 GPU 메모리. 단위: MiB

Elastic Graphics 차원

다음 차원을 사용하여 Elastic Graphics 액셀러레이터의 지표 데이터를 필터링할 수 있습니다.

차원	설명
EGPUID	Elastic Graphics 액셀러레이터를 기준으로 데이터를 필터링합니다.
InstanceId	Elastic Graphics 액셀러레이터가 연결된 인스턴스를 기준으로 데이터를 필터링합니다.

Elastic Graphics에 대한 CloudWatch 지표 보기

지표는 먼저 서비스 네임스페이스별로 그룹화된 다음, 각 네임스페이스 내에서 지원되는 차원별로 그룹화됩니다. 다음 절차를 사용하여 Elastic Graphics 액셀러레이터에 대한 지표를 볼 수 있습니다.

CloudWatch 콘솔을 사용하여 Elastic Graphics 지표를 보는 방법

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 필요한 경우 리전을 변경합니다. 탐색 모음에서 Elastic Graphics 액셀러레이터가 상주하는 리전을 선택합니다. 자세한 정보는 [리전 및 엔드포인트](#)를 참조하십시오.
3. 탐색 창에서 지표를 선택합니다.
4. 모든 지표에서 Elastic Graphics, Elastic Graphics 지표를 선택합니다.

Elastic Graphics 지표(AWS CLI)를 보는 방법

다음 `list-metrics` 명령을 사용합니다.

```
aws cloudwatch list-metrics --namespace "AWS/ElasticGPUs"
```

Elastic Graphics 모니터링을 위한 CloudWatch 경보 생성

경보가 상태를 변경하면 Amazon SNS 메시지를 보내는 CloudWatch 경보를 생성할 수 있습니다. 경보는 지정한 기간에 단일 측정치를 감시하고, 여러 기간에 지정된 임계값에 대한 측정치 값들 기준으로 Amazon SNS 주제에 알림을 보냅니다.

예를 들어 Elastic Graphics 액셀러레이터의 상태를 모니터링하는 경보를 생성하여 그래픽 액셀러레이터가 5분 기간 상태 확인을 연속 3회 실패할 경우 알림을 보낼 수 있습니다.

Elastic Graphics 액셀러레이터 상태에 대한 경보를 생성하는 방법

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 경보, 경보 생성을 선택합니다.
3. 지표 선택, Elastic Graphics, Elastic Graphics 지표를 선택합니다.,
4. GPUHealthCheckFailed 지표를 선택하고 지표 선택을 선택합니다.
5. 다음과 같이 경보를 구성합니다.
 - a. 경보 세부 정보에서 경보의 이름 및 설명을 입력합니다. 다음 경우 항상에서 \geq 를 선택하고 1을 입력합니다.
 - b. 작업에서 기존 알림 목록을 선택하거나 새 목록을 선택합니다.

- c. 경보 생성을 선택합니다.

문제 해결

다음은 일반적인 오류 및 문제 해결 단계입니다.

목차

- 애플리케이션 성능 문제 조사 (p. 645)
 - OpenGL 렌더링 성능 문제 (p. 645)
 - 원격 액세스 성능 문제 (p. 646)
- 비정상 상태 문제 해결 (p. 646)
 - 인스턴스 중지 및 시작 (p. 646)
 - 설치된 구성 요소 확인 (p. 647)
 - Elastic Graphics 로그 확인 (p. 647)

애플리케이션 성능 문제 조사

Elastic Graphics에서는 인스턴스 네트워크를 사용하여 OpenGL 명령을 원격으로 연결된 그래픽 카드에 전송합니다. 또한 Elastic Graphics 액셀러레이터를 사용하여 OpenGL 애플리케이션을 실행하는 데스크톱은 일반적으로 원격 액세스 기술을 사용하여 액세스됩니다. OpenGL 렌더링 관련된 성능 문제와 데스크톱 원격 액세스 기술 관련 성능 문제를 구별해야 합니다.

OpenGL 렌더링 성능 문제

OpenGL 렌더링 성능은 OpenGL 명령 수와 원격 인스턴스에서 생성된 프레임 수에 따라 결정됩니다.

렌더링 성능은 다음 요소에 따라 다를 수 있습니다.

- Elastic Graphics 액셀러레이터 성능
- 네트워크 성능
- CPU 성능
- 렌더링 모델, 시나리오 복잡성
- OpenGL 애플리케이션 동작

성능을 평가하는 쉬운 방법은 원격 인스턴스에 대해 렌더링된 프레임 수를 표시하는 것입니다. Elastic Graphics 액셀러레이터는 네트워크 사용을 최소화하면서 최상의 품질을 내기 위해 원격 인스턴스에 대해 최대 25FPS를 표시합니다.

생성된 프레임 수를 표시하려면

1. 텍스트 편집기에서 다음 파일을 엽니다. 파일이 없으면 새로 만듭니다.

```
C:\Program Files\Amazon\EC2ElasticGPUs\conf\eg.conf
```

2. [Application] 섹션을 확인하여 다음 구성 파라미터를 추가합니다. 합니다. 이 섹션이 없으면 추가합니다.

```
[Application]
show_fps=1
```

3. 애플리케이션을 다시 시작하고 FPS를 다시 확인합니다.

렌더링된 장면을 업데이트할 때 FPS가 15-25FPS에 도달하면 Elastic Graphics 액셀러레이터가 최고 성능을 내고 있는 것입니다. 발생한 다른 성능 문제는 인스턴스 데스크톱에 대한 원격 액세스와 관련이 있을 가능성 이 높습니다. 이 경우 원격 액세스 성능 문제 단원을 참조하십시오.

FPS가 15 미만이면 다음과 같이 해볼 수 있습니다.

- 보다 강력한 그래픽 액셀러레이터 유형을 선택하여 Elastic Graphics 액셀러레이터 성능을 높입니다.
- 다음 팁을 사용하여 전반적인 네트워크 성능을 개선합니다.
- Elastic Graphics 액셀러레이터 엔드포인트에 대한 수신 및 발신 대역폭의 양을 확인합니다. Elastic Graphics 액셀러레이터 엔드포인트는 다음과 같은 PowerShell 명령을 사용하여 가져올 수 있습니다.

```
PS C:\> (Invoke-WebRequest http://169.254.169.254/latest/meta-data/elastic-gpus/associations/[ELASTICGPU_ID]).content
```

- 인스턴스에서 Elastic Graphics 액셀러레이터 엔드포인트로의 네트워크 트래픽은 OpenGL 애플리케이션에서 생성 중인 명령의 양과 관련이 있습니다.
- Elastic Graphics 액셀러레이터 엔드포인트에서 인스턴스로의 네트워크 트래픽은 그래픽 액셀러레이터가 생성한 프레임의 수와 관련이 있습니다.
- 네트워크 사용량이 인스턴스 최대 네트워크 처리량에 도달한 경우 네트워크 처리 허용량이 더 높은 인스턴스를 사용해 보십시오.
- CPU 성능 개선:
 - 애플리케이션에 많은 CPU 리소스가 필요할 수 있으며 Elastic Graphics 액셀러레이터에도 리소스가 필요할 수 있습니다. Windows 작업 관리자에서 CPU 사용량이 높다고 보고할 경우 더 높은 CPU 사양으로 인스턴스를 사용해 보십시오.

원격 액세스 성능 문제

Elastic Graphics 액셀러레이터가 연결된 인스턴스는 다양한 원격 액세스 기술을 사용하여 액세스할 수 있습니다. 성능 및 품질은 다음 요인에 따라 다를 수 있습니다.

- 원격 액세스 기술
- 인스턴스 성능
- 클라이언트 성능
- 클라이언트와 인스턴스 간의 네트워크 지연 시간 및 대역폭

원격 액세스 프로토콜에 대해 다음을 선택할 수 있습니다.

- Microsoft Remote Desktop Connection
- NICE DCV
- VNC

최적화에 대한 자세한 내용은 특정 프로토콜을 참조하십시오.

비정상 상태 문제 해결

Elastic Graphics 액셀러레이터 상태가 비정상인 경우 다음 문제 해결 절차를 사용하여 문제를 해결합니다.

인스턴스 중지 및 시작

Elastic Graphics 액셀러레이터 상태가 비정상인 경우 가장 간단한 방법은 인스턴스를 중지한 후 다시 시작하는 것입니다. 자세한 내용은 [인스턴스 중지 및 시작 \(p. 442\)](#) 단원을 참조하십시오.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

설치된 구성 요소 확인

Windows 제어판을 열고 다음 구성 요소가 설치되어 있는지 확인합니다.

- Amazon Elastic Graphics Manager
- Amazon Elastic Graphics OpenGL Library
- Amazon EC2 Elastic GPUs OpenGL Redirector

이러한 항목 중 하나라도 없으면 수동으로 설치해야 합니다. 자세한 내용은 [Elastic Graphics용 필요 소프트웨어 설치 \(p. 640\)](#) 단원을 참조하십시오.

Elastic Graphics 로그 확인

Windows 이벤트 뷰어를 열고 응용 프로그램 및 서비스 로그 섹션을 확장한 후, 다음 이벤트 로그에서 오류를 검색합니다.

- EC2ElasticGPUs
- EC2ElasticGPUs GUI

Amazon EC2 모니터링

모니터링은 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스 및 AWS 솔루션의 안정성, 가용성 및 성능을 유지하는 데 있어서 중요한 부분입니다. 발생하는 각종 지점 실패를 보다 쉽게 디버깅할 수 있도록 AWS 솔루션의 모든 부분으로부터 모니터링 데이터를 수집해야 합니다. 그러나 Amazon EC2 모니터링을 시작하려면 먼저 다음을 포함하는 모니터링 계획을 생성해야 합니다.

- 모니터링의 목표
- 모니터링할 리소스
- 이러한 리소스를 모니터링하는 빈도
- 사용할 모니터링 도구
- 모니터링 작업을 수행할 사람
- 문제 발생 시 알려야 할 대상

모니터링 목표를 정의하고 모니터링 계획을 생성했으면, 다음 단계는 환경에서 Amazon EC2 성능의 기준선을 설정하는 것입니다. 다양한 시간과 다양한 부하 조건에서 Amazon EC2 성능을 측정해야 합니다. Amazon EC2를 모니터링할 때 수집한 모니터링 데이터의 기록을 저장해야 합니다. 현재 Amazon EC2 성능을 이 기록 데이터와 비교하면 일반적인 성능 패턴과 성능 이상을 식별하고 이를 해결할 방법을 고안할 수 있습니다. 예를 들어, EC2 인스턴스에 대해 CPU 사용률, 디스크 I/O 및 네트워크 사용률을 모니터링할 수 있습니다. 설정한 기준 이하로 성능이 떨어지면 인스턴스를 재구성하거나 최적화하여 CPU 사용률을 줄이거나 디스크 I/O를 개선하거나 네트워크 트래픽을 줄일 수 있습니다.

기준선을 설정하려면 최소한 다음 항목을 모니터링해야 합니다.

모니터링할 항목	Amazon EC2 지표	모니터링 에이전트/CloudWatch Logs
CPU 사용률	CPUUtilization (p. 669)	
네트워크 사용률	NetworkIn (p. 669) NetworkOut (p. 669)	
디스크 성능	DiskReadOps (p. 669) DiskWriteOps (p. 669)	
디스크 읽기/쓰기	DiskReadBytes (p. 669) DiskWriteBytes (p. 669)	
메모리 사용률, 디스크 스왑 사용률, 디스크 공간 사용률, 페이지 파일 사용률, 로그 수집		[Linux 및 Windows Server 인스턴스] CloudWatch 에이전트를 사용하여 Amazon EC2 인스턴스 및 온프레미스 서버로부터 지표 및 로그 수집 [Windows Server 인스턴스의 이전 CloudWatch Logs 에이전트에서 마이그레이션] Windows 서버 인스턴스 로그 수집을

모니터링할 항목	Amazon EC2 지표	모니터링 에이전트/CloudWatch Logs
		CloudWatch 에이전트로 마이그레이션

자동 및 수동 모니터링

AWS는 Amazon EC2를 모니터링하는 데 사용할 수 있는 다양한 도구를 제공합니다. 이들 도구 중에는 모니터링을 자동으로 수행하도록 구성할 수 있는 도구도 있지만, 수동 작업이 필요한 도구도 있습니다.

모니터링 도구

- [자동 모니터링 도구 \(p. 649\)](#)
- [수동 모니터링 도구 \(p. 650\)](#)

자동 모니터링 도구

다음과 같은 자동 모니터링 도구를 사용하여 Amazon EC2를 관찰하고 문제 발생 시 보고를 받을 수 있습니다.

- 시스템 상태 검사 – 인스턴스를 사용하는 데 필요한 AWS 시스템을 모니터링하여 올바르게 작동 중인지 확인합니다. 이러한 확인에서는 복구 시 AWS 개입이 필요한 인스턴스 관련 문제를 찾아냅니다. 시스템 상태 확인이 실패하는 경우, AWS에서 문제를 해결할 때까지 기다리거나, 인스턴스를 중지했다가 다시 시작하거나 종료하고 교체하는 등의 방법으로 사용자가 문제를 직접 해결할 수도 있습니다. 시스템 상태 확인이 실패하게 되는 문제의 예를 들면 다음과 같습니다.
 - 네트워크 연결 끊김
 - 시스템 전원 중단
 - 물리적 호스트의 소프트웨어 문제
 - 네트워크 연결성에 영향을 주는 물리적 호스트의 하드웨어 문제

자세한 내용은 [인스턴스 상태 확인 \(p. 651\)](#) 단원을 참조하십시오.

- 인스턴스 상태 검사 – 개별 인스턴스에 대한 소프트웨어 및 네트워크 구성은 모니터링합니다. 이러한 확인에서는 복구 시 사용자의 개입이 필요한 문제를 찾아냅니다. 인스턴스 상태 확인이 실패할 경우 일반적으로 사용자는 인스턴스를 재부팅하거나 운영 체제를 수정하는 등의 방법으로 문제를 직접 해결해야 합니다. 인스턴스 상태 확인이 실패하게 되는 문제의 예를 들면 다음과 같습니다.
 - 시스템 상태 확인 실패
 - 네트워크 구성 또는 시작 구성이 잘못됨
 - 메모리가 모두 사용됨
 - 파일 시스템 손상
 - 호환되지 않는 커널

자세한 내용은 [인스턴스 상태 확인 \(p. 651\)](#) 단원을 참조하십시오.

- Amazon CloudWatch 경보 – 지정하는 기간 동안 단일 지표를 관찰하고 특정 기간 동안 지정된 임계값을 기준으로 지표의 값에 따라 하나 이상의 작업을 수행합니다. 이 작업은 Amazon Simple Notification Service(Amazon SNS) 주제나 Amazon EC2 Auto Scaling 정책으로 전송되는 알림입니다. 경보는 지속적인 상태 변경에 대해서만 작업을 호출합니다. CloudWatch 경보는 특정 상태가 되었다고 해서 작업을 호출하지는 않습니다. 이러한 상태가 변경되어 지정한 기간 동안 유지되어야 합니다. 자세한 내용은 [CloudWatch를 사용해 인스턴스 모니터링 \(p. 667\)](#) 단원을 참조하십시오.
- Amazon CloudWatch Events – AWS 서비스를 자동화하여 시스템 이벤트에 자동으로 응답합니다. AWS 서비스 이벤트는 거의 실시간으로 CloudWatch 이벤트에 전송되며, 전송된 이벤트가 사용자가 정의한 규

최과 일치할 경우 실행할 자동 작업을 지정할 수 있습니다. 자세한 내용은 [Amazon CloudWatch Events이란 무엇입니까?](#) 단원을 참조하십시오.

- Amazon CloudWatch Logs – Amazon EC2 인스턴스, AWS CloudTrail 또는 기타 소스의 로그 파일을 모니터링, 저장 및 액세스합니다. 자세한 내용은 [Amazon CloudWatch Logs User Guide](#) 섹션을 참조하십시오.
- CloudWatch 에이전트 – EC2 인스턴스와 온프레미스 서버의 호스트 및 게스트 모두에서 로그와 시스템 수준 지표를 수집합니다. 자세한 내용은 Amazon CloudWatch 사용 설명서의 [CloudWatch 에이전트를 사용하여 Amazon EC2 인스턴스 및 온프레미스 서버로부터 지표 및 로그 수집](#)을 참조하십시오.
- System Center Operations Manager용 AWS 관리 팩 – Amazon EC2 인스턴스 및 인스턴스 내부에서 실행 중인 Microsoft Windows 또는 Linux 운영 체제를 연결합니다. AWS 관리 팩은 Microsoft System Center Operations Manager의 확장 기능입니다. 데이터 센터의 지정된 컴퓨터(감시자 노드)와 Amazon Web Services API를 사용하여 AWS 리소스에 대한 정보를 원격으로 검색하고 수집합니다. 자세한 정보는 [AWS Management Pack for Microsoft System Center \(p. 1244\)](#) 단원을 참조하십시오.

수동 모니터링 도구

Amazon EC2 모니터링의 또 한 가지 중요한 부분은 모니터링 스크립트, 상태 확인 및 CloudWatch 경보에 포함되지 않는 항목을 수동으로 모니터링해야 한다는 점입니다. Amazon EC2 및 CloudWatch 콘솔 대시보드에서는 Amazon EC2 환경을 한 눈에 파악할 수 있습니다.

- Amazon EC2 대시보드는 다음 정보를 표시합니다.
 - 리전별 서비스 상태 및 예약된 이벤트
 - 인스턴스 상태
 - 상태 확인
 - 경보 상태
 - 인스턴스 측정치 세부 정보(탐색 창에서 인스턴스를 선택하고, 인스턴스를 선택한 다음 모니터링 탭 선택)
 - 볼륨 지표 정보(탐색 창에서 볼륨을 선택하고 볼륨을 선택한 다음 모니터링 탭 선택)
- Amazon CloudWatch 대시보드는 다음 정보를 표시합니다.
 - 현재 경보 및 상태
 - 경보 및 리소스 그래프
 - 서비스 상태

또한 CloudWatch를 사용하여 다음 작업을 수행할 수도 있습니다.

- Amazon EC2 모니터링 데이터를 그래프로 작성하여 문제를 해결하고 추세 파악
- 모든 AWS 리소스 지표 검색 및 찾아보기
- 문제에 대해 알려주는 경보 생성 및 편집
- 경보 및 AWS 리소스를 한 눈에 파악할 수 있는 개요 정보 보기

모니터링 모범 사례

다음과 같은 모니터링 모범 사례를 이용하면 Amazon EC2 모니터링 작업을 보다 효과적으로 수행할 수 있습니다.

- 큰 문제로 확대되기 전에 작은 문제를 미리 방지하도록 모니터링 우선 순위를 지정하십시오.
- 발생하는 경우 다중 지점 실패를 보다 쉽게 디버깅할 수 있도록 AWS 솔루션의 모든 부분에서 모니터링 데이터를 수집하는 모니터링 계획을 생성하고 구현하십시오. 모니터링 계획은 최소한 다음 질문 사항에 대한 해답을 제공해야 합니다.
 - 모니터링의 목표
 - 모니터링 할 리소스

- 이러한 리소스를 모니터링하는 빈도
- 사용할 모니터링 도구
- 모니터링 작업을 수행할 사람
- 문제 발생 시 알려야 할 대상
- 모니터링 작업을 최대한 자동화하십시오.
- EC2 인스턴스에서 로그 파일을 확인하십시오.

인스턴스 상태 모니터링

인스턴스의 상태 확인과 예약된 이벤트 정보를 확인하면 인스턴스의 상태를 모니터링할 수 있습니다.

상태 확인은 Amazon EC2에서 실시하는 자동 확인 작업을 통해 정보를 제공합니다. 이러한 자동 검사는 인스턴스에 영향을 미치는 특정 문제가 있을 때 이를 감지합니다. 상태 확인 정보는 Amazon CloudWatch에서 제공되는 데이터와 함께 각 인스턴스에 대한 세부적인 운영 정보를 시각적으로 제공합니다.

인스턴스에서 예약된 특정 이벤트의 상태 또한 확인이 가능합니다. 이벤트 상태는 재부팅이나 만료 등 인스턴스에 대해 설정된 예정 활동에 대한 정보를 제공합니다. 또한 각 이벤트의 예약된 시작 시간과 종료 시간도 제공합니다.

목차

- [인스턴스 상태 확인 \(p. 651\)](#)
- [예약된 인스턴스 이벤트 \(p. 655\)](#)

인스턴스 상태 확인

인스턴스 상태 모니터링 작업은 Amazon EC2에서 인스턴스의 애플리케이션 실행에 지장을 줄 수 있는 문제를 발견했을 때 빠르게 확인할 수 있는 방법입니다. Amazon EC2에서는 실행 중인 모든 EC2 인스턴스에 대하여 자동 검사를 실행하여 하드웨어 및 소프트웨어 문제를 확인합니다. 이러한 상태 확인 결과를 토대로 식별 가능한 특정 문제를 확인할 수 있습니다. 이벤트 상태 데이터는 Amazon EC2가 이미 각 인스턴스 상태 (pending, running, stopping 등)에 대해 제공하는 정보와 Amazon CloudWatch가 모니터링하는 사용 지표(CPU 사용량, 네트워크 트래픽, 디스크 활동)를 보완합니다.

상태 확인은 1분마다 실행되며 통과 또는 실패 상태를 반환합니다. 모든 검사 결과가 통과인 경우 인스턴스의 전체 상태는 정상으로 표시됩니다. 하나 이상의 검사 결과가 실패인 경우에는 인스턴스의 전체 상태가 손상됨으로 표시됩니다. 상태 확인은 Amazon EC2에 내장된 기능으로 비활성화하거나 삭제할 수 없습니다.

상태 확인이 실패하면 상태 확인에 대한 해당 CloudWatch 지표가 증가합니다. 자세한 내용은 [상태 확인 지표 \(p. 674\)](#) 단원을 참조하십시오. 이러한 지표를 사용하여 상태 확인 결과를 기준으로 트리거되는 CloudWatch 경보를 생성할 수 있습니다. 예를 들어 특정 인스턴스의 상태 확인에서 실패 항목이 있을 때 알리는 경보를 생성할 수 있습니다. 자세한 내용은 [상태 확인 경보 생성 및 수정 \(p. 654\)](#) 단원을 참조하십시오.

Amazon EC2 인스턴스를 모니터링하고 기본 문제로 인해 인스턴스가 손상된 경우 인스턴스를 자동으로 복구하는 Amazon CloudWatch 경보를 생성할 수도 있습니다. 자세한 내용은 [인스턴스 복구 \(p. 460\)](#) 단원을 참조하십시오.

목차

- [상태 확인 유형 \(p. 652\)](#)
- [상태 확인 결과 확인 \(p. 652\)](#)
- [인스턴스 상태 보고 \(p. 653\)](#)

- 상태 확인 경보 생성 및 수정 (p. 654)

상태 확인 유형

상태 확인은 시스템 상태 확인과 인스턴스 상태 확인, 두 가지로 제공됩니다.

시스템 상태 확인

시스템 상태 확인은 인스턴스가 실행되는 AWS 시스템을 모니터링합니다. 이러한 확인에서는 복구 시 AWS 개입이 필요한 인스턴스와 관련된 근본적인 문제를 찾아냅니다. 시스템 상태 확인이 실패한 경우, AWS에서 문제를 해결할 때까지 기다리거나 문제를 직접 해결할 수 있습니다. Amazon EBS가 지원하는 인스턴스의 경우, 직접 인스턴스를 중지한 후 시작할 수 있으며 대부분의 경우 이 인스턴스를 새 호스트로 마이그레이션합니다. 인스턴스 스토어가 지원하는 인스턴스의 경우, 인스턴스를 종료하고 교체할 수 있습니다.

다음은 시스템 상태 확인의 실패 원인이 되는 몇 가지 문제의 예입니다.

- 네트워크 연결 끊김
- 시스템 전원 중단
- 물리적 호스트의 소프트웨어 문제
- 네트워크 연결성에 영향을 주는 물리적 호스트의 하드웨어 문제

인스턴스 상태 확인

인스턴스 상태 확인은 개별 인스턴스에 대한 소프트웨어 및 네트워크 구성의 모니터링입니다. Amazon EC2에서는 네트워크 인터페이스(NIC)로 주소 확인 프로토콜(ARP)을 전송하여 인스턴스의 상태를 점검합니다. 이러한 확인에서는 복구 시 사용자의 개입이 필요한 문제를 찾아냅니다. 인스턴스 상태 확인이 실패할 경우에는 일반적으로 사용자가 인스턴스를 재부팅하거나 인스턴스 구성의 변경을 통해 문제를 직접 해결해야 합니다.

다음은 인스턴스 상태 확인의 실패 원인이 되는 몇 가지 문제의 예입니다.

- 시스템 상태 확인 실패
- 잘못된 네트워킹 또는 스트리밍 구성
- 메모리가 모두 사용됨
- 파일 시스템 손상
- 인스턴스 재부팅이나 Windows 인스턴스 스토어 지원 인스턴스 번들링 중 인스턴스가 다시 available 상태로 변경될 때까지 인스턴스 상태 확인이 실패를 보고합니다.

상태 확인 결과 확인

Amazon EC2는 상태 확인 결과를 확인하고 대응할 수 있는 몇 가지 방법이 있습니다.

콘솔을 사용하여 상태 확인

AWS Management 콘솔을 사용해 상태 확인 결과를 확인할 수 있습니다.

상태 확인을 보려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스 페이지의 상태 검사 열에는 각 인스턴스의 운영 상태가 목록으로 표시됩니다.
4. 특정 인스턴스의 상태를 보려면 인스턴스를 선택하고 상태 검사 탭을 선택합니다.

The screenshot shows the 'Status Checks' section of the EC2 instance details. It includes tabs for 'Description', 'Status Checks' (which is active), 'Monitoring', and 'Tags'. A 'Create Status Check Alarm' button is present. The 'System Status Checks' section indicates a 'System reachability check passed'. The 'Instance Status Checks' section shows a failure with the message 'Instance reachability check failed at October 7, 2018'.

상태 확인 실패로 표시된 인스턴스가 있고 이 인스턴스가 확인 불가 상태로 20분 넘게 유지될 경우, AWS Support를 선택하여 지원을 요청합니다.

5. 상태 확인에 대한 CloudWatch 지표를 검토하려면 인스턴스를 선택한 다음 모니터링 탭을 선택합니다. 다음 지표에 대한 그래프가 보일 때까지 스크롤합니다.
 - 상태 확인 실패(해당되는 경우)
 - 상태 확인 실패(인스턴스)
 - 상태 확인 실패(시스템)

명령줄을 사용하여 상태 보기

`describe-instance-status`(AWS CLI) 명령을 사용해 실행 중인 인스턴스의 상태 확인 결과를 확인할 수 있습니다.

모든 인스턴스 상태를 확인하려면 다음 명령을 사용합니다.

```
aws ec2 describe-instance-status
```

impaired로 표시된 인스턴스의 상태를 모두 확인하려면 다음 명령을 사용합니다.

```
aws ec2 describe-instance-status \
--filters Name=instance-status.status,Values=impaired
```

단일 인스턴스의 상태를 확인하려면 다음 명령을 사용합니다.

```
aws ec2 describe-instance-status \
--instance-ids i-1234567890abcdef0
```

또는 다음 명령을 사용합니다.

- `Get-EC2InstanceState`(Windows PowerShell용 AWS 도구)
- `DescribeInstanceStatus`(Amazon EC2 Query API)

인스턴스 상태 보고

상태가 impaired로 표시되지 않은 인스턴스임에도 불구하고 문제가 발생하는 경우 또는 impaired 상태의 인스턴스와 관련해 AWS에 문제에 대해 추가 정보를 보내고 싶은 경우 피드백을 제공할 수 있습니다.

전달받은 피드백은 다수의 고객이 경험하는 문제를 식별하는 데 사용되며, 개별적인 계정 문제에 대해 따로 응답을 제공하지는 않습니다. 피드백을 제공해도 해당 인스턴스의 상태 확인 결과는 변동되지 않습니다.

콘솔을 사용하여 상태 피드백 보고

인스턴스 상태를 보고하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고, Status Checks(상태 확인) 탭을 선택한 후 피드백 제출을 선택합니다.
4. 인스턴스 상태 보고 양식을 작성한 후 제출을 선택합니다.

명령줄을 사용하여 상태 피드백 보고

[report-instance-status\(AWS CLI\)](#) 명령을 사용해 손상된 상태의 인스턴스에 대한 피드백을 전송합니다.

```
aws ec2 report-instance-status \
--instances i-1234567890abcdef0 \
--status impaired \
--reason-codes code
```

또는 다음 명령을 사용합니다.

- [Send-EC2InstanceState](#)(Windows PowerShell용 AWS 도구)
- [ReportInstanceState](#)(Amazon EC2 Query API)

상태 확인 경보 생성 및 수정

[상태 확인 지표 \(p. 674\)](#)를 사용하여 인스턴스에 실패한 상태 확인이 있을 때 알리는 CloudWatch 경보를 생성할 수 있습니다.

콘솔을 사용하여 상태 확인 경보 생성

다음 절차에 따라 상태 확인에 실패할 때 이메일을 통해 알림을 전송하거나 인스턴스를 중지, 종료 또는 복구하는 경보를 구성합니다.

상태 확인 경보 생성하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고, Status Checks(상태 확인) 탭을 선택한 후 상태 확인 경보 생성을 선택합니다.
4. 알림 보내기를 선택합니다. 기존 SNS 주제를 선택하거나, 주제 생성을 선택하여 새로운 주제를 생성합니다. 새로운 주제를 생성할 때는 수신자에 자신의 이메일 주소와 추가 수신자의 주소를 콤마로 구분하여 입력합니다.
5. (선택 사항) Take the action(작업 수행)을 선택한 후 원하는 작업을 선택합니다.
6. 다음 경우 항상에서 알고 싶은 상태 확인을 선택합니다.

이전 단계에서 인스턴스 복구를 선택한 경우에는 상태 검사 실패(시스템)를 선택합니다.

7. 최소 발생 기간에서 원하는 평가 주기의 개수를 설정하고 연속 기간에서 경보가 실행되고 이메일 전송이 이루어지기 전에 적용할 평가 주기의 시간 단위를 설정합니다.
8. (선택 사항) 경보 이름에서 경보의 기본 이름을 다른 이름으로 변경합니다.

- 경보 생성을 선택합니다.

Important

수신자 목록에 이메일 주소를 추가했거나 새 주제를 만든 경우 Amazon SNS에서는 각각의 새 주소로 가입 확인 이메일을 보냅니다. 모든 수신자는 각각 이메일에 포함된 링크를 선택하여 가입 여부를 확인해야 합니다. 경고 알림은 확인된 주소로만 전송됩니다.

필요한 경우 인스턴스 상태 경보를 수정할 수 있습니다.

콘솔을 사용해 상태 확인 경보를 편집하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 인스턴스를 선택합니다.
- 인스턴스를 선택하고 작업을 선택한 후 CloudWatch 모니터링을 선택하고 경보 추가/편집을 선택합니다.
- 경보 세부 정보(Alarm Details) 대화 상자에서 경보 이름을 선택합니다.
- 경보 편집 대화 상자에서 필요한 설정을 변경한 후 저장을 선택합니다.

AWS CLI를 사용하여 상태 확인 경보 생성

다음은 인스턴스에서 연속으로 2주기 이상 인스턴스 검사 또는 시스템 상태 확인이 중단되면서 경보가 발생하여 SNS 주제인 `arn:aws:sns:us-west-2:111122223333:my-sns-topic`에 대한 알림 메시지를 게시하는 예제입니다. 사용된 CloudWatch 지표는 `StatusCheckFailed`입니다.

AWS CLI를 사용해 상태 확인 경보를 생성하려면

- 기존의 SNS 주제를 선택하거나 새로운 주제를 생성합니다. 자세한 내용은 AWS Command Line Interface 사용 설명서에서 [Amazon SNS와 함께 AWS CLI 사용하기](#) 단원을 참조하십시오.
- 아래와 같이 `list-metrics` 명령을 사용하여 Amazon EC2에 유효한 Amazon CloudWatch 지표를 확인합니다.

```
aws cloudwatch list-metrics --namespace AWS/EC2
```

- 아래와 같이 `put-metric-alarm` 명령을 사용하여 경보를 생성합니다.

```
aws cloudwatch put-metric-alarm --alarm-name StatusCheckFailed-Alarm-for-i-1234567890abcdef0 --metric-name StatusCheckFailed --namespace AWS/EC2 --statistic Maximum --dimensions Name=InstanceId,Value=i-1234567890abcdef0 --unit Count --period 300 --evaluation-periods 2 --threshold 1 --comparison-operator GreaterThanOrEqualToThreshold --alarm-actions arn:aws:sns:us-west-2:111122223333:my-sns-topic
```

기간은 Amazon CloudWatch 지표가 수집되는 시간 프레임(초)입니다. 이 예제에서는 60초와 5분을 곱셈하여 300초를 사용합니다. 평균 기간은 지표 값을 임계값과 비교해야 하는 연속 기간의 수입니다. 이 예제에서는 2를 사용합니다. 경보 작업은 경보가 트리거될 때 수행할 작업입니다. 이 예제에서는 Amazon SNS를 사용해 이메일을 보낼 수 있도록 경보를 구성합니다.

예약된 인스턴스 이벤트

AWS는 재부팅, 종단/시작 또는 만료 등 여러 가지 인스턴스 이벤트를 예약할 수 있습니다. 이러한 이벤트들은 자주 발생하지 않습니다. 예약된 이벤트의 영향을 받는 인스턴스가 존재하는 경우 AWS가 해당 이벤트가 발생하기 전에 AWS 계정에 연동되어 있는 이메일 주소로 이메일을 전송합니다. 이메일은 시작일과 종료일 등 이벤트에 대한 세부 정보를 제공합니다. 이벤트 기간을 제어할 수 있는 작업은 이벤트에 따라 다릅니다.

예약된 이벤트는 AWS에서 관리되므로 인스턴스에 대한 이벤트를 예약할 수 없습니다. AWS에 의해 예약된 이벤트를 보고, 이메일 알림에서 태그를 포함하거나 제거하도록 예약된 이벤트 알림을 사용자 지정하고, 인스턴스가 재부팅, 만료 또는 중지되도록 예약된 경우 작업을 수행할 수 있습니다.

예약된 이벤트에 대한 세부 정보를 알 수 있도록 계정의 연락처 정보를 업데이트하려면 [계정 설정](#) 페이지로 이동합니다.

목차

- [예약된 이벤트 유형 \(p. 656\)](#)
- [예약된 이벤트 보기 \(p. 656\)](#)
- [예약된 이벤트 알림 사용자 지정 \(p. 660\)](#)
- [중지 또는 만료 예약된 인스턴스 관련 작업 \(p. 663\)](#)
- [재부팅 예약된 인스턴스 관련 작업 \(p. 663\)](#)
- [유지 관리 예약된 인스턴스 관련 작업 \(p. 664\)](#)
- [예약된 이벤트 다시 예약 \(p. 665\)](#)

예약된 이벤트 유형

Amazon EC2에서는 예약된 시간에 이벤트가 발생하는 인스턴스에 대해 다음과 같은 유형의 이벤트를 생성할 수 있습니다.

- Instance stop(인스턴스 중지): 예약된 시간에 인스턴스가 중지됩니다. 인스턴스를 다시 시작하면 새 호스트로 마이그레이션됩니다. 이러한 유형은 Amazon EBS가 지원하는 인스턴스에만 적용됩니다.
- Instance retirement(인스턴스 만료): 예약된 시간에 인스턴스가 Amazon EBS에서 지원되는 경우 중지되거나 인스턴스 스토어에서 지원되는 경우 종료됩니다.
- Instance reboot(인스턴스 재부팅): 예약된 시간에 인스턴스가 재부팅됩니다.
- System reboot(시스템 재부팅): 예약된 시간에 인스턴스의 호스트가 재부팅됩니다.
- System maintenance(시스템 유지 관리): 예약된 시간에 네트워크 또는 전력 유지 관리로 인스턴스가 일시적인 영향을 받을 수 있습니다.

예약된 이벤트 보기

예약된 이벤트에 대한 알림 메시지를 이메일로 받는 것 외에도, 다음 방법 중 하나를 이용해 예약된 이벤트를 확인할 수 있습니다.

새로운 콘솔

콘솔을 사용해 인스턴스에 예약된 이벤트를 확인하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 열니다.
2. 다음 화면에서 예약된 이벤트를 볼 수 있습니다.
 - 탐색 창에서 이벤트를 선택합니다. 연결된 이벤트가 있는 모든 리소스가 표시됩니다. 리소스 ID, 리소스 유형, 가용 영역, 이벤트 상태 또는 이벤트 유형을 기준으로 필터링할 수 있습니다.

The screenshot shows the 'Events' section of the Amazon EC2 console. At the top, there are three filter buttons: 'Resource type: instance', 'Event status: Scheduled', and 'Event type: instance-stop'. Below the filters is a table header with columns: Resource ID, Event status, and Event type. A single row is visible in the table, showing 'i-02c48fffbba61cd16f' under Resource ID, 'Scheduled' under Event status, and 'instance-stop' under Event type. To the left of the table, there's a sidebar titled 'Scheduled events' which lists 'US East (N. Virginia)' with '7 instance(s) have scheduled events' and '1 volume(s) are impaired'. A yellow callout box on the right side of the table area contains the text 'Retiring: This instance is scheduled for retirement after February 12, 2020 at 12:00:00 AM UTC+2.' followed by an info icon.

- 또는 탐색 창에서 EC2 대시보드를 선택합니다. 연결된 이벤트가 있는 모든 리소스가 예약된 이벤트 아래에 표시됩니다.
- 일부 이벤트가 영향을 받는 리소스에 대해서도 표시됩니다. 예를 들어 탐색 창에서 인스턴스를 선택한 후 인스턴스를 하나 선택합니다. 인스턴스에 인스턴스 중지 또는 인스턴스 만료 이벤트가 연동되어 있는 경우 해당 이벤트가 하단 창에 표시됩니다.

기존 콘솔

콘솔을 사용해 인스턴스에 예약된 이벤트를 확인하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 다음 화면에서 예약된 이벤트를 볼 수 있습니다.
 - 탐색 창에서 이벤트를 선택합니다. 연결된 이벤트가 있는 모든 리소스가 표시됩니다. 표시 방식은 리소스 유형이나 특정 이벤트 유형으로 필터링할 수 있습니다. 또한 리소스를 선택하여 세부 정보를 확인할 수도 있습니다.

Filter: All resource types ▾ All event types ▾ Ongoing and scheduled ▾

Resource Name	Resource Type	Resource Id	Event Type
my-instance	instance	i-c3870335	instance-stop

Event: i-c3870335

Availability Zone: us-west-2a
Event type: instance-stop
Event status: Scheduled
Description: The instance is running on degraded hardware
Start time: May 22, 2015 at 5:00:00 PM UTC-7
End time:

- 또는 탐색 창에서 EC2 대시보드를 선택합니다. 연결된 이벤트가 있는 모든 리소스가 예약된 이벤트 아래에 표시됩니다.

Scheduled Events

US West (Oregon):

1 instances have scheduled events

- 일부 이벤트가 영향을 받는 리소스에 대해서도 표시됩니다. 예를 들어 탐색 창에서 인스턴스를 선택한 후 인스턴스를 하나 선택합니다. 인스턴스에 인스턴스 중지 또는 인스턴스 만료 이벤트가 연동되어 있는 경우 해당 이벤트가 하단 창에 표시됩니다.



Retiring: This instance is scheduled for retirement after May 22, 2015 at 5:00:00 PM UTC-7.



AWS CLI

AWS CLI를 사용해 인스턴스에 예약된 이벤트를 확인하는 방법

describe-instance-status 명령을 사용합니다.

```
aws ec2 describe-instance-status \
--instance-id i-1234567890abcdef0 \
--query "InstanceStatuses[ ].Events"
```

다음 예제 출력은 재부팅 이벤트를 보여줍니다.

```
[{"Events": [
{
    "InstanceEventId": "instance-event-0d59937288b749b32",
    "Code": "system-reboot",
    "Description": "The instance is scheduled for a reboot",
    "NotAfter": "2019-03-15T22:00:00.000Z",
    "NotBefore": "2019-03-14T20:00:00.000Z",
    "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"
}]}
```

```
]  
]
```

다음 예제 출력은 인스턴스 만료 이벤트를 표시합니다.

```
[  
    "Events": [  
        {  
            "InstanceEventId": "instance-event-0e439355b779n26",  
  
            "Code": "instance-stop",  
            "Description": "The instance is running on degraded hardware",  
            "NotBefore": "2015-05-23T00:00:00.000Z"  
        }  
    ]  
]
```

PowerShell

Windows PowerShell용 AWS 도구를 사용해 인스턴스에 예약된 이벤트를 확인하는 방법

다음 [Get-EC2InstanceState](#) 명령을 사용합니다.

```
PS C:\> (Get-EC2InstanceState -InstanceId i-1234567890abcdef0).Events
```

다음 예제 출력은 인스턴스 만료 이벤트를 표시합니다.

```
Code      : instance-stop  
Description : The instance is running on degraded hardware  
NotBefore : 5/23/2015 12:00:00 AM
```

Instance metadata

인스턴스 메타데이터를 사용해 인스턴스에 예약된 이벤트를 확인하는 방법

인스턴스 메타데이터 서비스 버전 2 또는 인스턴스 메타데이터 서비스 버전 1을 사용하여 [인스턴스 메타데이터 \(p. 576\)](#)에서 인스턴스에 대해 활성화된 유지 관리 이벤트 정보를 검색할 수 있습니다.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \  
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/events/maintenance/scheduled
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/maintenance/scheduled
```

다음은 예약된 시스템 재부팅 이벤트에 관한 정보를 JSON 형식으로 표시하는 예제 출력입니다.

```
[  
    {  
        "NotBefore" : "21 Jan 2019 09:00:43 GMT",  
        "Code" : "system-reboot",  
        "Description" : "scheduled reboot",  
        "EventId" : "instance-event-0d59937288b749b32",  
        "NotAfter" : "21 Jan 2019 09:17:23 GMT",  
    }  
]
```

```
        "State" : "active"  
    }  
]
```

인스턴스 메타데이터를 사용하여 인스턴스에 대해 완료되거나 취소된 이벤트 관련 이벤트 기록을 확인하는 방법

인스턴스 메타데이터 서비스 버전 2 또는 인스턴스 메타데이터 서비스 버전 1을 사용하여 [인스턴스 메타데이터 \(p. 576\)](#)에서 인스턴스에 대해 완료되거나 취소된 이벤트 관련 정보를 검색할 수 있습니다.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`  
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/events/maintenance/history
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/maintenance/history
```

다음은 취소된 시스템 재부팅 이벤트와 완료된 시스템 재부팅 이벤트 관련 정보를 JSON 형식으로 표현하는 예제 출력입니다.

```
[  
 {  
     "NotBefore" : "21 Jan 2019 09:00:43 GMT",  
     "Code" : "system-reboot",  
     "Description" : "[Canceled] scheduled reboot",  
     "EventId" : "instance-event-0d59937288b749b32",  
     "NotAfter" : "21 Jan 2019 09:17:23 GMT",  
     "State" : "canceled"  
 },  
 {  
     "NotBefore" : "29 Jan 2019 09:00:43 GMT",  
     "Code" : "system-reboot",  
     "Description" : "[Completed] scheduled reboot",  
     "EventId" : "instance-event-0d59937288b749b32",  
     "NotAfter" : "29 Jan 2019 09:17:23 GMT",  
     "State" : "completed"  
 }  
 ]
```

예약된 이벤트 알림 사용자 지정

이메일 알림에 태그를 포함하도록 예약된 이벤트 알림을 사용자 지정할 수 있습니다. 이렇게 하면 영향을 받는 리소스(인스턴스 또는 전용 호스트)를 더 쉽게 식별하고 다가오는 이벤트에 대한 작업의 우선 순위를 지정할 수 있습니다.

태그를 포함하도록 이벤트 알림을 사용자 지정할 때 다음을 포함하도록 선택할 수 있습니다.

- 영향을 받는 리소스와 연결된 모든 태그
- 영향을 받는 리소스와 연결된 특정 태그만

예를 들어 모든 인스턴스에 application, costcenter, project 및 owner 태그를 할당하는 경우, 이벤트 알림에 이러한 모든 태그를 포함하도록 선택하거나 이벤트 알림에 owner 및 project 태그만 표시하기 위해 이러한 태그만 포함하도록 선택할 수 있습니다.

포함할 태그를 선택하면 이벤트 알림에 영향을 받는 리소스와 연결된 리소스 ID(인스턴스 전용 호스트 ID 또는 ID)와 태그 키 및 값 페어가 포함됩니다.

주제

- [이벤트 알림에 태그 포함 \(p. 661\)](#)
- [이벤트 알림에서 태그 제거 \(p. 661\)](#)
- [이벤트 알림에 포함할 태그 보기 \(p. 662\)](#)

이벤트 알림에 태그 포함

포함하도록 선택한 태그는 선택한 리전의 모든 리소스(인스턴스 및 전용 호스트)에 적용됩니다. 다른 지역에서 이벤트 알림을 사용자 지정하려면 먼저 필요한 리전을 선택한 후 다음 단계를 수행합니다.

다음 방법 중 하나를 사용하여 이벤트 알림에 태그를 포함할 수 있습니다.

새로운 콘솔

이벤트 알림에 태그를 포함하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 템색 창에서 [Events]를 선택합니다.
3. 작업, Manage event notifications(이벤트 알림 관리)를 선택합니다.
4. Include resource tags in event notifications(이벤트 알림에 리소스 태그 포함)를 선택합니다.
5. 이벤트 알림에 포함할 태그에 따라 다음 중 하나를 수행합니다.
 - 영향을 받는 인스턴스 또는 전용 호스트와 연결된 모든 태그를 포함하려면 Include all resource tags(모든 리소스 태그 포함)를 선택합니다.
 - 포함할 태그를 수동으로 선택하려면 Choose the tags to include(포함할 태그 선택)를 선택한 다음 Choose the tags to include(포함할 태그 선택)에서 태그 키를 입력하고 Enter를 누릅니다.
6. 저장을 선택합니다.

AWS CLI

이벤트 알림에 모든 태그를 포함하려면

`register-instance-event-notification-attributes` AWS CLI 명령을 사용하고 `IncludeAllTagsOfInstance` 파라미터를 `true`로 설정합니다.

```
aws ec2 register-instance-event-notification-attributes --instance-tag-attribute "IncludeAllTagsOfInstance=true"
```

이벤트 알림에 특정 태그를 포함하려면

`register-instance-event-notification-attributes` AWS CLI 명령을 사용하고 `InstanceTagKeys` 파라미터를 사용하여 포함할 태그를 지정합니다.

```
aws ec2 register-instance-event-notification-attributes --instance-tag-attribute 'InstanceTagKeys=["tag_key_1", "tag_key_2", "tag_key_3"]'
```

이벤트 알림에서 태그 제거

다음 방법 중 하나를 사용하여 이벤트 알림에서 태그를 제거할 수 있습니다.

새로운 콘솔

이벤트 알림에서 태그를 제거하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Events]를 선택합니다.
3. 작업, Manage event notifications(이벤트 알림 관리)를 선택합니다.
4. 이벤트 알림에서 제거할 태그에 따라 다음 중 하나를 수행합니다.
 - 이벤트 알림에서 모든 태그를 제거하려면 Include resource tags in event notifications(이벤트 알림에 리소스 태그 포함)의 선택을 취소합니다.
 - 이벤트 알림에서 특정 태그를 제거하려면 Choose the tags to include(포함할 태그 선택) 필드 아래에 나열된 태그에 대해 제거(X)를 선택합니다.
5. 저장을 선택합니다.

AWS CLI

이벤트 알림에서 모든 태그를 제거하려면

`deregister-instance-event-notification-attributes` AWS CLI 명령을 사용하고 `IncludeAllTagsOfInstance` 파라미터를 `false`로 설정합니다.

```
aws ec2 deregister-instance-event-notification-attributes --instance-tag-attribute "IncludeAllTagsOfInstance=false"
```

이벤트 알림에서 특정 태그를 제거하려면

`deregister-instance-event-notification-attributes` AWS CLI 명령을 사용하고 `InstanceTagKeys` 파라미터를 사용하여 제거할 태그를 지정합니다.

```
aws ec2 deregister-instance-event-notification-attributes --instance-tag-attribute 'InstanceTagKeys=["tag_key_1", "tag_key_2", "tag_key_3"]'
```

이벤트 알림에 포함할 태그 보기

다음 방법 중 하나를 사용하여 이벤트 알림에 포함할 태그를 볼 수 있습니다.

새로운 콘솔

이벤트 알림에 포함할 태그를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Events]를 선택합니다.
3. 작업, Manage event notifications(이벤트 알림 관리)를 선택합니다.

AWS CLI

이벤트 알림에 포함할 태그를 보려면

`describe-instance-event-notification-attributes` AWS CLI 명령을 사용합니다.

```
aws ec2 describe-instance-event-notification-attributes
```

중지 또는 만료 예약된 인스턴스 관련 작업

AWS가 인스턴스의 기본 호스트에서 복구 불가능한 결함을 감지하면 인스턴스의 루트 디바이스 유형에 따라 인스턴스의 중지 또는 종료를 예약합니다. 루트 디바이스가 EBS 볼륨이면 인스턴스 중단이 예약됩니다. 그렇지 않고 루트 디바이스가 인스턴스 스토어 볼륨이면 인스턴스 종료가 예약됩니다. 자세한 내용은 [인스턴스 만료 \(p. 452\)](#) 단원을 참조하십시오.

Important

인스턴스가 중지되거나 종료되면 인스턴스 스토어 볼륨에 저장되었던 데이터는 모두 삭제됩니다. 여기에는 루트 디바이스가 EBS 볼륨인 인스턴스에 연결된 인스턴스 스토어 볼륨도 포함됩니다. 따라서 인스턴스 스토어 볼륨에서 나중에 필요한 데이터는 인스턴스가 중단 또는 종료되기 전에 반드시 저장하십시오.

Amazon EBS에서 지원되는 인스턴스 작업

인스턴스가 예약 시간에 중단될 때까지 기다릴 수 있습니다. 또는 직접 인스턴스를 중지한 후 시작하여 새 호스트로 마이그레이션하는 것도 가능합니다. 인스턴스 중단과 중단 후 인스턴스 구성 변경 방법에 대한 자세한 내용은 [인스턴스 중지 및 시작 \(p. 440\)](#)을 참조하십시오.

예약된 인스턴스 중지 이벤트에 대한 응답으로 즉시 중지 및 시작을 자동화할 수 있습니다. 자세한 내용은 AWS Health 사용 설명서의 [EC2 인스턴스에 대한 작업 자동화](#)를 참조하십시오.

인스턴스 스토어에서 지원되는 인스턴스 작업

인스턴스 종료 예약 시간 이전에 가장 최신 AMI에서 생성된 인스턴스로 대체하고 필요한 모든 정보를 대체 인스턴스로 마이그레이션하는 것이 권장됩니다. 작업 후에는 원본 인스턴스를 종료하거나 예약 시간에 종료될 때까지 기다리면 됩니다.

재부팅 예약된 인스턴스 관련 작업

AWS에 업데이트 설치나 기본 호스트 유지 관리 등의 작업이 필요할 때는 인스턴스 또는 인스턴스의 기본 호스트가 재부팅되도록 예약할 수 있습니다. 적합한 특정 날짜 및 시간에 인스턴스가 재부팅되도록 [대부분의 재부팅 이벤트를 다시 예약 \(p. 665\)](#) 할 수 있습니다.

링크된 [EC2-Classic 인스턴스 \(p. 812\)](#)를 중지하면 자동으로 VPC에서 링크가 해제되고 VPC 보안 그룹이 인스턴스와 더 이상 연결되지 않습니다. 인스턴스를 다시 시작한 후 VPC에 다시 연결할 수 있습니다.

재부팅 이벤트 유형 보기

다음 방법 중 하나를 사용하여 재부팅 이벤트가 인스턴스 재부팅인지 아니면 시스템 재부팅인지 확인할 수 있습니다.

새로운 콘솔

콘솔에서 예약된 재부팅 이벤트의 유형을 확인하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Events]를 선택합니다.
3. 필터 목록에서 리소스 유형: 인스턴스를 선택합니다.
4. 각 인스턴스의 이벤트 유형 열에서 값을 확인합니다. 값은 system-reboot 또는 instance-reboot입니다.

기존 콘솔

콘솔에서 예약된 재부팅 이벤트의 유형을 확인하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 이벤트를 선택합니다.
3. 필터 목록에서 인스턴스 리소스를 선택합니다.
4. 각 인스턴스의 이벤트 유형 열에서 값을 확인합니다. 값은 system-reboot 또는 instance-reboot입니다.

AWS CLI

AWS CLI를 사용하여 예약된 재부팅 이벤트의 유형을 확인하는 방법

[describe-instance-status](#) 명령을 사용합니다.

```
aws ec2 describe-instance-status --instance-id i-1234567890abcdef0
```

예약된 재부팅 이벤트의 경우 Code의 값은 system-reboot 또는 instance-reboot입니다. 다음 예제 출력은 system-reboot 이벤트를 보여줍니다.

```
[  
    "Events": [  
        {  
            "InstanceEventId": "instance-event-0d59937288b749b32",  
            "Code": "system-reboot",  
            "Description": "The instance is scheduled for a reboot",  
            "NotAfter": "2019-03-14T22:00:00.000Z",  
            "NotBefore": "2019-03-14T20:00:00.000Z",  
            "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"  
        }  
    ]  
]
```

인스턴스 재부팅 작업

예약된 유지 관리 기간 내에 인스턴스 재부팅이 실행될 때까지 기다리거나, 적합한 특정 날짜 및 시간으로 인스턴스 재부팅을 [다시 예약 \(p. 665\)](#)하거나, 편리한 시간에 직접 인스턴스를 [재부팅 \(p. 452\)](#)할 수 있습니다.

인스턴스가 재부팅된 후 예약된 이벤트가 삭제되고 이벤트 설명이 업데이트됩니다. 기본 호스트에서 보류되었던 점검이 완료되면 부팅이 완전히 끝난 이후에 인스턴스를 다시 사용할 수 있습니다.

시스템 재부팅 작업

시스템은 직접 재부팅할 수 없습니다. 예약된 유지 관리 기간 내에 시스템이 재부팅될 때까지 기다리거나, 적합한 날짜 및 시간으로 시스템 재부팅을 [다시 예약 \(p. 665\)](#)할 수 있습니다. 시스템 재부팅은 보통 분 단위로 완료됩니다. 시스템 재부팅이 발생한 후 인스턴스는 해당 IP 주소 및 DNS 이름을 그대로 유지하고 로컬 인스턴스 스토어 볼륨의 데이터가 보존됩니다. 시스템 재부팅이 완료되면 인스턴스에 예약된 이벤트가 삭제되며, 인스턴스 소프트웨어가 예상대로 실행되는지 확인할 수 있습니다.

또는 인스턴스를 다른 시간에 유지 관리해야 하며 시스템 재부팅을 다시 예약할 수 없는 경우 Amazon EBS 지원 인스턴스를 중지한 후 시작하여 새 호스트로 마이그레이션하는 것이 가능합니다. 그러나 로컬 인스턴스 스토어 볼륨에 저장된 데이터가 손실됩니다. 또한 예약된 시스템 재부팅 이벤트에 대한 응답으로 즉시 인스턴스 중지 및 시작을 자동화할 수 있습니다. 자세한 내용은 AWS Health 사용 설명서의 [EC2 인스턴스에 대한 작업 자동화](#)를 참조하십시오. 인스턴스 스토어 지원 인스턴스의 경우 시스템 재부팅을 다시 예약할 수 없는 경우 가장 최근 AMI에서 교체 인스턴스를 시작하고 예약된 유지 관리 기간 이전에 필요한 데이터를 모두 교체 인스턴스로 마이그레이션한 다음 원본 인스턴스를 종료할 수 있습니다.

유지 관리 예약된 인스턴스 관련 작업

AWS에서 인스턴스의 기본 호스트를 유지 관리해야 하는 경우 인스턴스의 유지 관리가 예약됩니다. 유지 관리 유형은 네트워크 유지 관리와 전력 유지 관리, 두 가지입니다.

네트워크 유지 관리 시에는 예약된 인스턴스의 네트워크 연결이 잠시 동안 끊어집니다. 유지 관리가 완료되면 인스턴스의 네트워크 연결이 평소처럼 복구됩니다.

전력 유지 관리 시에는 예약된 인스턴스가 잠시 동안 오프라인 상태로 전환되다가 재부팅됩니다. 재부팅 이후에도 인스턴스의 모든 구성 설정은 그대로 유지됩니다.

약 몇 분 후에 인스턴스가 재부팅되면 애플리케이션이 정상적으로 작동하는지 확인하도록 합니다. 이때 인스턴스에 더 이상 예약된 이벤트가 없거나, 있는 경우 예약된 이벤트가 [완료]로 표시됩니다. 인스턴스 상태 설명을 새로 고치는 데 최대 1시간이 걸리는 경우도 있습니다. 완료된 유지 관리 이벤트는 Amazon EC2 콘솔 대시보드에 일주일까지 표시됩니다.

Amazon EBS에서 지원되는 인스턴스 작업

예약 시간에 유지 관리가 실행될 때까지 기다릴 수 있습니다. 또는 인스턴스를 중지한 후 시작하여 새 호스트로 마이그레이션하는 것도 가능합니다. 인스턴스 중단과 중단 후 인스턴스 구성을 변경하는 방법에 대한 자세한 내용은 [인스턴스 중지 및 시작 \(p. 440\)](#)을 참조하십시오.

예약된 유지 관리 이벤트에 대한 응답으로 즉시 중지 및 시작을 자동화할 수 있습니다. 자세한 내용은 AWS Health 사용 설명서의 [EC2 인스턴스에 대한 작업 자동화](#)를 참조하십시오.

인스턴스 스토어에서 지원되는 인스턴스 작업

예약 시간에 유지 관리가 실행될 때까지 기다릴 수 있습니다. 그 밖에 유지 관리 예약 기간에도 정상적인 작업을 지속해야 할 경우에는 가장 최근 AMI에서 대체 인스턴스를 실행한 다음 예약 기간 이전에 필요한 데이터를 모두 대체 인스턴스로 마이그레이션하고 원본 인스턴스를 종료할 수도 있습니다.

예약된 이벤트 다시 예약

이벤트가 적합한 특정 날짜와 시간에 발생하도록 다시 예약할 수 있습니다. 기한 날짜가 있는 이벤트만 다시 예약할 수 있습니다. [이벤트를 다시 예약할 때 제한 사항 \(p. 666\)](#)이 더 있습니다.

다음 방법 중 하나를 사용하여 이벤트를 다시 예약할 수 있습니다.

새로운 콘솔

콘솔을 사용하여 이벤트를 다시 예약하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Events]를 선택합니다.
3. 필터 목록에서 리소스 유형: 인스턴스를 선택합니다.
4. 하나 이상의 인스턴스를 선택한 다음 작업, Schedule event(이벤트 예약)를 선택합니다.

Deadline(기한) 값으로 표시되는 이벤트 기한이 있는 이벤트만 다시 예약할 수 있습니다. 선택한 이벤트 중 하나에 기한 날짜가 없으면 작업, Schedule event(이벤트 예약)가 비활성화됩니다.

5. New start time(새 시작 시간)에 이벤트에 대한 새 날짜 및 시간을 입력합니다. 새 날짜 및 시간은 Event deadline(이벤트 기한) 이전이어야 합니다.
6. 저장을 선택합니다.

업데이트된 이벤트 시작 시간이 콘솔에 반영되려면 1-2분의 시간이 걸릴 수 있습니다.

기존 콘솔

콘솔을 사용하여 이벤트를 다시 예약하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 이벤트를 선택합니다.

3. 필터 목록에서 인스턴스 리소스를 선택합니다.
4. 하나 이상의 인스턴스를 선택한 다음 작업, Schedule Event(이벤트 예약)를 선택합니다.

Event Deadline(이벤트 기한) 값으로 표시되는 이벤트 기한이 있는 이벤트만 다시 예약할 수 있습니다.

5. Event start time(이벤트 시작 시간)에 이벤트에 대한 새 날짜 및 시간을 입력합니다. 새 날짜 및 시간은 Event Deadline(이벤트 기한) 이전이어야 합니다.
6. Schedule Event(이벤트 예약)를 선택합니다.

업데이트된 이벤트 시작 시간이 콘솔에 반영되려면 1-2분의 시간이 걸릴 수 있습니다.

AWS CLI

AWS CLI를 사용하여 이벤트를 다시 예약하려면

1. NotBeforeDeadline 값으로 표시되는 이벤트 기한이 있는 이벤트만 다시 예약할 수 있습니다. `describe-instance-status` 명령을 사용하여 NotBeforeDeadline 파라미터 값을 확인합니다.

```
aws ec2 describe-instance-status --instance-id i-1234567890abcdef0
```

다음 예제 출력에서는 NotBeforeDeadline에 값이 포함되므로 다시 예약할 수 있는 system-reboot 이벤트를 보여줍니다.

```
[  
    "Events": [  
        {  
            "InstanceEventId": "instance-event-0d59937288b749b32",  
            "Code": "system-reboot",  
            "Description": "The instance is scheduled for a reboot",  
            "NotAfter": "2019-03-14T22:00:00.000Z",  
            "NotBefore": "2019-03-14T20:00:00.000Z",  
            "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"  
        }  
    ]  
]
```

2. 이벤트를 다시 예약하려면 `modify-instance-event-start-time` 명령을 사용합니다. `not-before` 파라미터를 사용하여 새 이벤트 시작 시간을 지정합니다. 새 이벤트 시작 시간은 NotBeforeDeadline 이전이어야 합니다.

```
aws ec2 modify-instance-event-start-time --instance-id i-1234567890abcdef0  
--instance-event-id instance-event-0d59937288b749b32 --not-before 2019-03-25T10:00:00.000
```

`describe-instance-status` 명령에서 업데이트된 `not-before` 파라미터 값을 반환하는 데 1-2분의 시간이 걸릴 수 있습니다.

제한 사항

- 이벤트 기한이 있는 이벤트만 다시 예약할 수 있습니다. 이 이벤트는 이벤트 기한까지 다시 예약될 수 있습니다. 콘솔의 Deadline(기한) 일 및 AWS CLI의 NotBeforeDeadline 필드는 이벤트에 기한이 있음을 나타냅니다.
- 아직 시작하지 않은 이벤트만 다시 예약할 수 있습니다. 콘솔의 시작 시간 일 및 AWS CLI의 NotBefore 필드는 이벤트 시작 시간을 나타냅니다. 다음 5분 내에 시작하도록 예약된 이벤트는 다시 예약할 수 없습니다.

- 새 이벤트 시작 시간은 현재 시간에서 최소 60분 내여야 합니다.
- 콘솔을 사용하여 여러 이벤트를 다시 예약하는 경우 이벤트 기한은 가장 이른 이벤트 기한의 이벤트에 의해 결정됩니다.

CloudWatch를 사용해 인스턴스 모니터링

Amazon EC2에서 원시 데이터를 수집하여 읽기 가능하며 실시간에 가까운 측정치로 처리하는 Amazon CloudWatch를 사용해 인스턴스를 모니터링할 수 있습니다. 이러한 통계는 15개월간 기록되므로 기록 정보를 보고 웹 애플리케이션이나 서비스가 어떻게 실행되고 있는지 전체적으로 더 잘 파악할 수 있습니다.

Amazon EC2는 기본적으로 측정치 데이터를 5분 동안 CloudWatch에 전송합니다. 인스턴스에 대한 측정치 데이터를 CloudWatch에 1분 동안 전송하기 위해 해당 인스턴스에 대한 세부 모니터링을 활성화할 수 있습니다. 자세한 내용은 [인스턴스에 대한 세부 모니터링 활성화 또는 비활성화 \(p. 667\)](#) 단원을 참조하십시오.

Amazon EC2 콘솔에는 Amazon CloudWatch의 원시 데이터를 기초로 하는 일련의 그래프가 표시됩니다. 필요에 따라 콘솔의 그래프 대신에 Amazon CloudWatch에서 인스턴스 데이터를 얻는 것을 선호할 수도 있습니다.

Amazon CloudWatch에 대한 자세한 내용은 [Amazon CloudWatch 사용 설명서](#)를 참조하십시오.

목차

- [인스턴스에 대한 세부 모니터링 활성화 또는 비활성화 \(p. 667\)](#)
- [인스턴스에 사용 가능한 CloudWatch 지표 나열 \(p. 669\)](#)
- [인스턴스에 대한 지표 통계 가져오기 \(p. 679\)](#)
- [인스턴스에 대한 그래프 지표 \(p. 687\)](#)
- [인스턴스에 대해 CloudWatch 경보 만들기 \(p. 687\)](#)
- [인스턴스를 중지, 종료, 재부팅 또는 복구하는 경보 만들기 \(p. 688\)](#)

인스턴스에 대한 세부 모니터링 활성화 또는 비활성화

기본 설정상 인스턴스는 기본 모니터링 기능이 활성화되어 있습니다. 세부 모니터링 활성화를 선택할 수 있습니다. 세부 모니터링을 활성화하면 Amazon EC2 콘솔에 인스턴스에 대한 1분 모니터링 그래프가 표시됩니다.

다음은 인스턴스에 대한 기본 및 세부 모니터링의 데이터 간격 및 요금에 대해 설명합니다.

기본 모니터링

자동으로 5분 기간 동안 데이터를 무료로 사용할 수 있습니다.

세부 모니터링

추가 비용을 지불하면 데이터를 1분 기간으로 사용할 수 있습니다.

이러한 데이터 수준을 얻으려면 인스턴스에 대해 해당 수준을 사용하도록 설정해야 합니다. 세부 모니터링을 활성화한 인스턴스의 경우 유사한 인스턴스 그룹 간에 집계된 데이터를 얻을 수도 있습니다.

세부 모니터링 요금

세부 모니터링을 사용하도록 설정하면 CloudWatch로 전송되는 지표별로 요금이 청구됩니다. 데이터 스토리지에는 요금이 부과되지 않습니다. 세부 모니터링 요금에 대한 자세한 내용은 [Amazon CloudWatch 요금 페이지](#)의 유료 티어를 참조하십시오. 요금 예는 [Amazon CloudWatch 요금 페이지](#)의 예 1 - EC2 세부 모니터링을 참조하십시오.

세부 모니터링 활성화

인스턴스를 시작할 때 또는 인스턴스가 실행 중이거나 종지된 후에 인스턴스에 대한 세부 모니터링을 활성화 할 수 있습니다. 인스턴스에 대한 세부 모니터링 기능을 활성화해도 인스턴스에 연결된 EBS 볼륨 모니터링에는 영향을 주지 않습니다. 자세한 내용은 [Amazon EBS에 대한 Amazon CloudWatch 지표 \(p. 1076\)](#) 단원을 참조하십시오.

기존 인스턴스에 대한 세부 모니터링을 활성화하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, CloudWatch 모니터링, 세부 모니터링 활성화를 차례로 선택합니다.
4. 세부 모니터링 활성화 대화 상자에서 예, 활성화를 선택합니다.
5. 닫기를 선택합니다.

인스턴스 시작 시 세부 모니터링을 활성화하려면(콘솔)

AWS Management 콘솔을 사용해 인스턴스를 시작할 때 Configure Instance Details(인스턴스 세부 정보 구성) 페이지에서 모니터링에 있는 확인란을 선택합니다.

기존 인스턴스에 대한 세부 모니터링을 활성화하려면(AWS CLI)

다음 `monitor-instances` 명령을 사용하여 지정된 인스턴스에 대한 세부 모니터링을 활성화합니다.

```
aws ec2 monitor-instances --instance-ids i-1234567890abcdef0
```

인스턴스 시작 시 세부 모니터링을 활성화하려면(AWS CLI)

--monitoring 플래그와 함께 `run-instances` 명령을 사용하여 세부 모니터링을 활성화합니다.

```
aws ec2 run-instances --image-id ami-09092360 --monitoring Enabled=true...
```

세부 모니터링 비활성화

인스턴스를 시작할 때 또는 인스턴스가 실행 중이거나 종지된 후에 인스턴스에 대한 세부 모니터링을 비활성화할 수 있습니다.

세부 모니터링을 비활성화하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, CloudWatch 모니터링, 세부 모니터링 비활성화를 차례로 선택합니다.
4. 세부 모니터링 비활성화 대화 상자에서 예, 비활성화를 선택합니다.
5. 닫기를 선택합니다.

세부 모니터링을 비활성화하려면(AWS CLI)

다음 `unmonitor-instances` 명령을 사용하여 지정된 인스턴스에 대한 세부 모니터링을 비활성화합니다.

```
aws ec2 unmonitor-instances --instance-ids i-1234567890abcdef0
```

인스턴스에 사용 가능한 CloudWatch 지표 나열

Amazon EC2는 측정치를 Amazon CloudWatch로 전송합니다. AWS Management 콘솔, AWS CLI 또는 API를 사용하여 Amazon EC2가 CloudWatch에 전송하는 측정치를 나열할 수 있습니다. 기본적으로 각 데이터 요소는 인스턴스의 시작 시간 이후 5분간 활동을 다룹니다. 세부 모니터링을 활성화한 경우 각 데이터 요소는 시작 시간부터 1분간 활동을 다룹니다.

이 측정치에 대한 통계를 얻는 방법에 대한 자세한 내용은 [인스턴스에 대한 지표 통계 가져오기 \(p. 679\)](#) 단원을 참조하십시오.

목차

- [인스턴스 지표 \(p. 669\)](#)
- [CPU 크레딧 지표 \(p. 671\)](#)
- [Nitro 기반 인스턴스용 Amazon EBS 지표 \(p. 672\)](#)
- [상태 확인 지표 \(p. 674\)](#)
- [트래픽 미러링 지표 \(p. 674\)](#)
- [Amazon EC2 지표 차원 \(p. 674\)](#)
- [Amazon EC2 사용량 지표 \(p. 675\)](#)
- [콘솔을 사용해 지표 나열 \(p. 676\)](#)
- [AWS CLI를 사용해 지표 나열 \(p. 678\)](#)

인스턴스 지표

AWS/EC2 네임스페이스에는 다음 인스턴스 지표가 포함되어 있습니다.

지표	설명
CPUUtilization	<p>인스턴스에서 현재 사용 중인 할당된 EC2 컴퓨팅 유닛(ECU)의 비율(%)입니다. 이 지표는 선택한 인스턴스에서 애플리케이션을 실행하는데 필요한 처리 능력을 식별합니다.</p> <p>인스턴스 유형에 따라, 인스턴스에 전체 프로세스 코어가 할당되지 않았을 때 운영 체제의 도구에서 비율이 CloudWatch보다 낮게 표시할 수 있습니다.</p> <p>단위: 백분율</p>
DiskReadOps	<p>지정된 시간 내에 인스턴스에 사용할 수 있는 모든 인스턴스 스토어 볼륨에서 읽기 작업 완료.</p> <p>기간의 평균 IOPS(초당 I/O 작업 수)를 계산하려면 기간의 총 작업 수를 해당 기간의 초 수로 나누십시오.</p> <p>인스턴스 스토어 볼륨이 없으면 값이 0이거나 측정치가 보고되지 않습니다.</p> <p>단위: 개수</p>
DiskWriteOps	<p>지정된 시간 내에 인스턴스에 사용할 수 있는 모든 인스턴스 스토어 볼륨에 대한 쓰기 작업 완료.</p> <p>기간의 평균 IOPS(초당 I/O 작업 수)를 계산하려면 기간의 총 작업 수를 해당 기간의 초 수로 나누십시오.</p>

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 사용 가능한 지표 나열

지표	설명
	<p>인스턴스 스토어 볼륨이 없으면 값이 0이거나 측정치가 보고되지 않습니다.</p> <p>단위: 개수</p>
DiskReadBytes	<p>인스턴스에 사용할 수 있는 모든 인스턴스 스토어 볼륨에서 읽은 바이트 수.</p> <p>이 지표는 애플리케이션이 인스턴스의 하드 디스크에서 읽는 데이터 볼륨을 결정하는 데 사용됩니다. 이를 사용하여 애플리케이션의 속도를 결정할 수 있습니다.</p> <p>보고된 숫자는 해당 기간에 수신된 바이트 수입니다. 기본(5분) 모니터링을 사용하는 경우, 이 숫자를 300으로 나누어 바이트/초를 찾을 수 있습니다. 세부(1분) 모니터링의 경우 60으로 나눕니다.</p> <p>인스턴스 스토어 볼륨이 없으면 값이 0이거나 측정치가 보고되지 않습니다.</p> <p>단위: 바이트</p>
DiskWriteBytes	<p>인스턴스에 사용할 수 있는 모든 인스턴스 스토어 볼륨에 쓴 바이트 수.</p> <p>이 지표는 애플리케이션이 인스턴스의 하드 디스크에 쓰는 데이터 볼륨을 결정하는 데 사용됩니다. 이를 사용하여 애플리케이션의 속도를 결정할 수 있습니다.</p> <p>보고된 숫자는 해당 기간에 수신된 바이트 수입니다. 기본(5분) 모니터링을 사용하는 경우, 이 숫자를 300으로 나누어 바이트/초를 찾을 수 있습니다. 세부(1분) 모니터링의 경우 60으로 나눕니다.</p> <p>인스턴스 스토어 볼륨이 없으면 값이 0이거나 측정치가 보고되지 않습니다.</p> <p>단위: 바이트</p>
NetworkIn	<p>인스턴스가 모든 네트워크 인터페이스에서 받은 바이트 수입니다. 이 측정치는 단일 인스턴스로 들어오는 네트워크 트래픽의 볼륨을 식별합니다.</p> <p>보고된 숫자는 해당 기간에 수신된 바이트 수입니다. 기본(5분) 모니터링을 사용하는 경우, 이 숫자를 300으로 나누어 바이트/초를 찾을 수 있습니다. 세부(1분) 모니터링의 경우 60으로 나눕니다.</p> <p>단위: 바이트</p>
NetworkOut	<p>인스턴스가 모든 네트워크 인터페이스에서 보낸 바이트 수입니다. 이 측정치는 단일 인스턴스에서 나가는 네트워크 트래픽의 볼륨을 식별합니다.</p> <p>보고된 숫자는 해당 기간에 전송된 바이트 수입니다. 기본(5분) 모니터링을 사용하는 경우, 이 숫자를 300으로 나누어 바이트/초를 찾을 수 있습니다. 세부(1분) 모니터링의 경우 60으로 나눕니다.</p> <p>단위: 바이트</p>

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
사용 가능한 지표 나열

지표	설명
NetworkPacketsIn	<p>인스턴스가 모든 네트워크 인터페이스에서 받은 패킷 수입니다. 이 지표는 단일 인스턴스에서 수신 트래픽의 볼륨을 패킷 수 기준으로 측정합니다. 기본 모니터링에서만 이 지표를 사용할 수 있습니다.</p> <p>단위: 개수</p> <p>Statistics: Minimum, Maximum, Average</p>
NetworkPacketsOut	<p>인스턴스가 모든 네트워크 인터페이스에서 보낸 패킷 수입니다. 이 지표는 단일 인스턴스에서 발신 트래픽의 볼륨을 패킷 수 기준으로 측정합니다. 기본 모니터링에서만 이 지표를 사용할 수 있습니다.</p> <p>단위: 개수</p> <p>Statistics: Minimum, Maximum, Average</p>
MetadataNoToken	<p>ток론을 사용하지 않는 방법으로 인스턴스 메타데이터 서비스에 성공적으로 액세스한 횟수입니다.</p> <p>이 메트릭은 인스턴스 메타데이터 서비스 버전 1을 사용 중인 인스턴스 메타데이터에 토큰을 사용하지 않고 액세스하는 프로세스가 있는지 확인하는데 사용됩니다. 모든 요청이 토큰 지원 세션을 사용하는 경우(예: 인스턴스 메타데이터 서비스 버전 2) 값은 0입니다. 자세한 내용은 인스턴스 메타데이터 서비스 버전 2 사용으로 전환 (p. 578) 단원을 참조하십시오.</p> <p>단위: 개수</p>

CPU 크레딧 지표

AWS/EC2 네임스페이스에는 [성능 순간 확장 가능 인스턴스 \(p. 129\)](#)에 대한 다음 CPU 크레딧 지표가 포함되어 있습니다.

지표	설명
CPUCreditUsage	<p>CPU 사용률을 위해 인스턴스에서 소비되는 CPU 크레딧의 수입니다. CPU 크레딧 하나는 1분 동안 100%의 사용률로 실행되는 vCPU 1개 또는 이와 동등한 vCPU, 사용률 및 시간의 조합과 동일합니다(예를 들어 2분 동안 50%의 사용률로 실행되는 vCPU 1개 또는 2분 동안 25%의 사용률로 실행되는 vCPU 2개).</p> <p>CPU 크레딧 측정치는 5분 간격으로만 제공됩니다. 5분 이상의 시간을 지정할 경우 Sum 통계 대신 Average 통계를 사용하십시오.</p> <p>단위: 크레딧 (vCPU-분)</p>
CPUCreditBalance	<p>시작 이후 인스턴스가 누적한 획득 CPU 크레딧 수입니다. T2 스탠다드의 경우 CPUCreditBalance에 누적된 시작 크레딧 수도 포함됩니다.</p> <p>크레딧은 획득 이후에 크레딧 밸런스에 누적되고, 소비 시 크레딧 밸런스에서 소멸됩니다. 크레딧 밸런스는 최대 한도(인스턴스 크기에 따라 결정)가 있습니다. 한도에 도달하면 새로 획득한 크레딧이 모두 삭제됩니다. T2 스탠다드의 경우 시작 크레딧은 한도에 포함되지 않습니다.</p>

지표	설명
	<p>CPUCreditBalance의 크레딧은 인스턴스가 기준 CPU 사용률 이상으로 버스터를 하는 데 소비할 수 있습니다.</p> <p>인스턴스가 실행 중인 동안 CPUCreditBalance의 크레딧은 만료되지 않습니다. T3 또는 T3a 인스턴스가 종지되면 CPUCreditBalance 값은 7일 동안 지속됩니다. 그 이후에는 누적된 크레딧이 모두 삭제됩니다. T2 인스턴스가 종지되면 CPUCreditBalance 값은 지속되지 않고 누적된 크레딧이 모두 삭제됩니다.</p> <p>CPU 크레딧 측정치는 5분 간격으로만 제공됩니다.</p> <p>단위: 크레딧 (vCPU-분)</p>
CPUSurplusCreditBalance	<p>CPUCreditBalance 값이 0일 때 unlimited 인스턴스에서 소비된 잉여 크레딧의 수입니다.</p> <p>획득한 CPU 크레딧에 따라 CPUSurplusCreditBalance 값이 청산됩니다. 잉여 크레딧의 수가 인스턴스가 24시간 동안 획득할 수 있는 최대 크레딧 수를 초과한 경우 최대 값 이상으로 소비된 잉여 크레딧은 추가 요금으로 부과됩니다.</p> <p>CPU 크레딧 측정치는 5분 간격으로만 제공됩니다.</p> <p>단위: 크레딧 (vCPU-분)</p>
CPUSurplusCreditsCharged	<p>획득한 CPU 크레딧으로 청산되지 않는 소비 잉여 크레딧의 수로, 추가 요금으로 부과됩니다.</p> <p>소비된 잉여 크레딧은 다음이 발생할 때 요금이 부과됩니다.</p> <ul style="list-style-type: none"> 소비한 잉여 크레딧이 인스턴스가 24시간 동안 획득할 수 있는 최대 크레딧 수를 초과하는 경우. 해당 시간이 끝날 때 최대 값 이상으로 소비한 잉여 크레딧에 요금이 부과됩니다. 인스턴스가 종지 또는 종료된 경우. 인스턴스가 unlimited에서 standard로 전환됩니다. <p>CPU 크레딧 측정치는 5분 간격으로만 제공됩니다.</p> <p>단위: 크레딧 (vCPU-분)</p>

Nitro 기반 인스턴스용 Amazon EBS 지표

AWS/EC2 네임스페이스에는 베어 메탈 인스턴스가 아닌 Nitro 기반 인스턴스에 대한 다음 Amazon EBS 측정치가 포함되어 있습니다. Nitro 기반 인스턴스 유형의 목록은 [Nitro 시스템에 구축된 인스턴스 \(p. 118\)](#) 단원을 참조하십시오.

Nitro 기반 인스턴스에 대한 측정치 값은 항상 정수인 반면에 Xen 기반 인스턴스의 값은 소수를 지원합니다. 따라서 Nitro 기반 인스턴스의 낮은 인스턴스 CPU 사용률은 반내림되어 0으로 나타날 수 있습니다.

지표	설명
EBSReadOps	지정된 기간 내에 인스턴스에 연결된 모든 Amazon EBS 볼륨에서 완료된 읽기 작업입니다.

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
사용 가능한 지표 나열

지표	설명
	<p>해당 기간의 초당 평균 읽기 I/O 작업 수(읽기 IOPS)를 계산하려면 해당 기간의 총 작업 수를 해당 기간의 초 수로 나누십시오. 기본(5분) 모니터링을 사용하는 경우, 이 숫자를 300으로 나누어 읽기 IOPS를 계산할 수 있습니다. 세부(1분) 모니터링의 경우 60으로 나눕니다.</p> <p>단위: 수</p>
EBSWriteOps	<p>지정된 기간 내에 인스턴스에 연결된 모든 EBS 볼륨으로의 완료된 쓰기 작업입니다.</p> <p>해당 기간의 초당 평균 쓰기 I/O 작업 수(쓰기 IOPS)를 계산하려면 해당 기간의 총 작업 수를 해당 기간의 초 수로 나누십시오. 기본(5분) 모니터링을 사용하는 경우, 이 숫자를 300으로 나누어 쓰기 IOPS를 계산할 수 있습니다. 세부(1분) 모니터링의 경우 60으로 나눕니다.</p> <p>단위: 수</p>
EBSReadBytes	<p>지정된 기간 내에 인스턴스에 연결된 모든 EBS 볼륨에서의 바이트 읽기 작업입니다.</p> <p>보고된 숫자는 해당 기간에 읽은 바이트 수입니다. 기본(5분) 모니터링을 사용하는 경우, 이 숫자를 300으로 나누어 읽기 바이트/초를 찾을 수 있습니다. 세부(1분) 모니터링의 경우 60으로 나눕니다.</p> <p>단위: 바이트</p>
EBSWriteBytes	<p>지정된 기간 내에 인스턴스에 연결된 모든 EBS 볼륨으로의 바이트 쓰기 작업입니다.</p> <p>보고된 숫자는 해당 기간에 써진 바이트 수입니다. 기본(5분) 모니터링을 사용하는 경우, 이 숫자를 300으로 나누어 쓰기 바이트/초를 찾을 수 있습니다. 세부(1분) 모니터링의 경우 60으로 나눕니다.</p> <p>단위: 바이트</p>
EBSIOBalance%	<p>더 작은 인스턴스 크기에서만 사용할 수 있습니다. 버스트 버킷에 남아 있는 I/O 크레딧의 비율에 대한 정보를 제공합니다. 기본 모니터링에서만 이 지표를 사용할 수 있습니다.</p> <p>Sum 통계는 이 지표에 적용할 수 없습니다.</p> <p>단위: 백분율</p>
EBSByteBalance%	<p>더 작은 인스턴스 크기에서만 사용할 수 있습니다. 버스트 버킷에 남아 있는 처리량 크레딧의 비율에 대한 정보를 제공합니다. 기본 모니터링에서만 이 지표를 사용할 수 있습니다.</p> <p>Sum 통계는 이 지표에 적용할 수 없습니다.</p> <p>단위: 백분율</p>

EBS 볼륨에 제공되는 측정치에 대한 자세한 내용은 [Amazon EBS 지표 \(p. 1076\)](#) 단원을 참조하십시오. 스팟 집합에 제공되는 측정치에 대한 자세한 내용은 [스팟 집합에 대한 CloudWatch 지표 \(p. 297\)](#) 단원을 참조하십시오.

상태 확인 지표

AWS/EC2 네임스페이스에는 다음과 같은 상태 확인 지표가 포함되어 있습니다. 기본적으로 시스템 상태 지표는 1분 주기로 무료로 사용할 수 있습니다. 새로 시작된 인스턴스의 경우, 인스턴스에서 초기화 상태를 완료해야 상태 확인 지표 데이터를 얻을 수 있습니다(인스턴스가 실행 상태로 시작되는 몇 분 내). EC2 상태 확인에 대한 자세한 내용은 [인스턴스 상태 확인 \(p. 651\)](#) 단원을 참조하십시오.

지표	설명
StatusCheckFailed	<p>인스턴스가 마지막으로 인스턴스 상태 확인 및 시스템 상태 확인을 통과했는지 여부를 보고합니다.</p> <p>이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다.</p> <p>기본적으로 이 지표는 1분 주기로 무료로 사용할 수 있습니다.</p> <p>단위: 개수</p>
StatusCheckFailed_Instance	<p>인스턴스가 마지막으로 인스턴스 상태 확인을 통과했는지 여부를 보고합니다.</p> <p>이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다.</p> <p>기본적으로 이 지표는 1분 주기로 무료로 사용할 수 있습니다.</p> <p>단위: 개수</p>
StatusCheckFailed_System	<p>인스턴스가 마지막으로 시스템 상태 확인을 통과했는지 여부를 보고합니다.</p> <p>이 지표는 0(통과) 또는 1(실패)이 될 수 있습니다.</p> <p>기본적으로 이 지표는 1분 주기로 무료로 사용할 수 있습니다.</p> <p>단위: 개수</p>

트래픽 미러링 지표

AWS/EC2 네임스페이스에는 미러링된 트래픽에 대한 지표가 포함됩니다. 자세한 내용은 Amazon VPC 트래픽 미러링 가이드의 [Amazon CloudWatch를 사용한 미러링된 트래픽 모니터링](#)을 참조하십시오.

Amazon EC2 지표 차원

다음 차원을 사용하여 이전 표에 나열된 지표를 구체화할 수 있습니다.

차원	설명
AutoScalingGroupName	<p>이 차원은 사용자가 지정된 용량 그룹의 모든 인스턴스에 대해 요청하는 데이터를 필터링합니다. Auto Scaling 그룹은 Auto Scaling을 사용할 경우 사용자가 정의하는 인스턴스 모음입니다. 이 차원은 인스턴스가 이러한 Auto Scaling 그룹에 있을 때 Amazon EC2 측정치에만 사용할 수 있습니다. 세부 또는 기본 모니터링이 설정된 인스턴스에 사용할 수 있습니다.</p>

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
사용 가능한 지표 나열

차원	설명
ImageId	이 차원은 사용자가 이 Amazon EC2 Amazon 마신 이미지(AMI)를 실행하는 모든 인스턴스에 대해 요청하는 데이터를 필터링합니다. 세부 모니터링이 설정된 인스턴스에 사용할 수 있습니다.
InstanceId	이 차원은 사용자가 식별된 인스턴스에 대해 요청하는 데이터만 필터링합니다. 이는 데이터를 모니터링할 정확한 인스턴스를 정확히 식별하는 데 도움이 됩니다.
InstanceType	이 차원은 사용자가 지정된 이 인스턴스 유형으로 실행되는 모든 인스턴스에 대해 요청하는 데이터를 필터링합니다. 이는 실행 중인 인스턴스 유형별로 데이터를 범주화하는 데 도움이 됩니다. 예를 들어, m1.small 인스턴스와 m1.large 인스턴스의 데이터를 비교하여 애플리케이션에 대해 더 높은 비즈니스 가치를 가진 인스턴스를 결정할 수 있습니다. 세부 모니터링이 설정된 인스턴스에 사용할 수 있습니다.

Amazon EC2 사용량 지표

CloudWatch 사용량 지표를 사용하여 계정의 리소스 사용량을 확인할 수 있습니다. 이러한 지표를 사용하여 CloudWatch 그래프 및 대시보드에서 현재 서비스 사용량을 시각화합니다.

Amazon EC2 사용량 지표는 AWS 서비스 할당량에 해당합니다. 사용량이 서비스 할당량에 가까워지면 경고하는 경보를 구성할 수 있습니다. CloudWatch와 서비스 할당량의 통합에 대한 자세한 내용은 [서비스 할당량 통합 및 사용량 지표](#)를 참조하십시오.

Amazon EC2는 AWS/Usage 네임스페이스에 다음 지표를 게시합니다.

지표	설명
ResourceCount	계정에서 실행 중인 지정된 리소스의 수입니다. 리소스는 지표와 연결된 차원에 의해 정의됩니다. 이 지표에 대한 가장 유용한 통계는 1분 동안 사용되는 최대 리소스 수를 나타내는 MAXIMUM입니다.

다음 차원은 Amazon EC2에 의해 게시되는 사용량 지표를 구체화하는 데 사용됩니다.

차원	설명
Service	리소스가 포함된 AWS 서비스의 이름 Amazon EC2 사용량 지표의 경우 이 차원 값은 EC2입니다.
Type	보고되는 엔터티의 유형입니다. 현재 Amazon EC2 사용량 지표에 대한 유일한 유효 값은 Resource입니다.
Resource	실행 중인 리소스의 유형입니다. 현대 Amazon EC2 사용량 지표에 대한 유일한 유효 값은 실행 중인 인스턴스에 대한 정보를 반환하는 vCPU입니다.
Class	추적 중인 리소스의 클래스. Resource 차원의 값이 vCPU인 Amazon EC2 사용량 지표의 경우 유효한 값은 Standard/OnDemand, F/OnDemand, G/OnDemand, Inf/OnDemand, P/OnDemand 및 X/OnDemand입니다.

차원	설명
	이 차원의 값은 지표에서 보고하는 인스턴스 유형의 첫 글자를 정의합니다. 예를 들어, Standard/OnDemand는 유형이 A, C, D, H, I, M, R, T 및 Z로 시작하는 모든 실행 중인 인스턴스에 대한 정보를 반환하며 G/OnDemand는 유형이 G로 시작하는 모든 실행 중인 인스턴스에 대한 정보를 반환합니다.

콘솔을 사용해 지표 나열

지표는 먼저 네임스페이스별로 그룹화된 다음, 각 네임스페이스 내에서 다양한 차원 조합별로 그룹화됩니다. 예를 들어, Amazon EC2에 의해 제공되는 모든 측정치나 인스턴스 ID, 인스턴스 유형, 이미지(AMI) ID 또는 Auto Scaling 그룹별로 제공되는 측정치를 볼 수 있습니다.

범주별로 사용 가능한 측정치를 보려면(콘솔)

1. <https://console.aws.amazon.com/cloudwatch>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 지표를 선택합니다.
3. EC2 측정치 네임스페이스를 선택합니다.

The screenshot shows the CloudWatch Metrics search interface. At the top, there are three tabs: 'All metrics' (highlighted in orange), 'Graphed metrics' (selected), and 'Graph options'. Below the tabs is a search bar with the placeholder text 'Search for any metric, dimension or resource id'. The main area displays a grid of metrics categorized by service. The categories and their counts are: EBS (117 Metrics), EC2 (316 Metrics), EFS (7 Metrics), ELB (210 Metrics), ElasticBeanstalk (8 Metrics), RDS (60 Metrics), and S3 (4 Metrics).

4. 지표 차원(예: 인스턴스별 지표)을 선택합니다.

The screenshot shows the 'All metrics' tab selected in the top navigation bar. Below it, the breadcrumb navigation shows 'All > EC2'. A search bar is present with the placeholder 'Search for any metric, dimension or resource id'. The main content area displays '103 Metrics' and lists them under several categories:

- By Auto Scaling Group**: 28 Metrics
- By Image (AMI) Id**: 7 Metrics
- Per-Instance Metrics**: 54 Metrics
- Aggregated by Instance Type**: 7 Metrics
- Across All Instances**: 7 Metrics

5. 측정치를 정렬하려면 열 머리글을 사용합니다. 측정치를 그래프로 표시하려면 측정치 옆에 있는 확인란을 선택합니다. 리소스로 필터링하려면 리소스 ID를 선택한 후 검색에 추가를 선택합니다. 지표로 필터링하려면 지표 이름을 선택한 후 검색에 추가를 선택합니다.

	Instance Name (192)	InstanceId	Metric Name
<input type="checkbox"/>	my-instance	i-abbc12a7	CPUUtilization
<input type="checkbox"/>	my-instance		DiskReadBytes
<input type="checkbox"/>	my-instance		DiskReadOps
<input type="checkbox"/>	my-instance		DiskWriteBytes
<input type="checkbox"/>	my-instance		DiskWriteOps
<input type="checkbox"/>	my-instance		NetworkIn
<input type="checkbox"/>	my-instance		NetworkOut
<input type="checkbox"/>	my-instance	i-abbc12a7	NetworkPacketsIn
<input type="checkbox"/>	my-instance	i-abbc12a7	NetworkPacketsOut

AWS CLI를 사용해 지표 나열

`list-metrics` 명령을 사용하여 인스턴스에 대한 CloudWatch 측정치를 나열합니다.

Amazon EC2의 모든 측정치를 표시하려면(AWS CLI)

다음 예제는 Amazon EC2에 대한 모든 지표를 볼 수 있도록 AWS/EC2 네임스페이스를 지정합니다.

```
aws cloudwatch list-metrics --namespace AWS/EC2
```

다음은 예제 출력입니다.

```
{
  "Metrics": [
    {
      "Namespace": "AWS/EC2",
      "Dimensions": [
        {
          "Name": "InstanceId",
          "Value": "i-1234567890abcdef0"
        }
      ],
      "MetricName": "NetworkOut"
    },
    {
      "Namespace": "AWS/EC2",
      "Dimensions": [
        {
          "Name": "InstanceId",
          "Value": "i-1234567890abcdef0"
        }
      ],
      "MetricName": "CPUUtilization"
    }
  ]
}
```

```
},
{
  "Namespace": "AWS/EC2",
  "Dimensions": [
    {
      "Name": "InstanceId",
      "Value": "i-1234567890abcdef0"
    }
  ],
  "MetricName": "NetworkIn"
},
...
]
```

인스턴스에 대한 모든 측정치를 표시하려면(AWS CLI)

다음 예제는 지정한 인스턴스의 결과만 보도록 AWS/EC2 네임스페이스와 InstanceId 차원을 지정합니다.

```
aws cloudwatch list-metrics --namespace AWS/EC2 --dimensions
  Name=InstanceId,Value=i-1234567890abcdef0
```

모든 인스턴스에 대한 측정치를 나열하려면(AWS CLI)

다음 예제는 지정한 지표의 결과만 보도록 AWS/EC2 네임스페이스와 지표 이름을 지정합니다.

```
aws cloudwatch list-metrics --namespace AWS/EC2 --metric-name CPUUtilization
```

인스턴스에 대한 지표 통계 가져오기

인스턴스에 대한 CloudWatch 측정치 통계를 볼 수 있습니다.

목차

- 통계 개요 (p. 679)
- 특정 인스턴스에 대한 통계 가져오기 (p. 680)
- 여러 인스턴스의 통계 집계 (p. 683)
- Auto Scaling 그룹별 통계 집계 (p. 685)
- AMI별 집계 통계 (p. 686)

통계 개요

통계는 지정한 기간에 걸친 지표 데이터 집계입니다. CloudWatch에서는 사용자 지정 데이터를 통해 제공되었거나 다른 AWS 서비스에서 CloudWatch에 제공한 지표 데이터 요소를 기반으로 통계를 제공합니다. 집계는 네임스페이스, 지표 이름, 차원 및 데이터 요소 측정 단위를 사용하여 지정한 기간에 대해 수행됩니다. 다음 표에서는 사용 가능한 통계에 대해 설명합니다.

통계	설명
Minimum	지정된 기간 중 관찰된 가장 낮은 값입니다. 이 값을 사용하여 애플리케이션에 대한 낮은 볼륨의 활동을 확인할 수 있습니다.
Maximum	지정된 기간 중 관찰된 가장 높은 값입니다. 이 값을 사용하여 애플리케이션에 대한 높은 볼륨의 활동을 확인할 수 있습니다.
Sum	일치하는 지표에 대해 제출된 모든 값이 서로 더해진 값입니다. 이 통계는 지표의 총 볼륨을 확인할 때 유용할 수 있습니다.

통계	설명
Average	지정된 기간 중 Sum/SampleCount의 값입니다. 이 통계를 Minimum 및 Maximum과 비교하면 지표의 전체 범위와 평균 사용량이 Minimum 및 Maximum에 얼마나 근접했는지 확인할 수 있습니다. 이와 같은 비교를 통해 필요에 따라 리소스를 늘리거나 줄여야 하는 시점을 파악할 수 있습니다.
SampleCount	통계 계산에 사용된 데이터 요소의 수(숫자)입니다.
pNN.NN	지정된 백분위 수의 값. 소수점 두 자리까지 사용하여 백분위 수를 지정할 수 있습니다 (예: p95.45).

특정 인스턴스에 대한 통계 가져오기

다음 예제는 AWS Management 콘솔 또는 AWS CLI 명령을 사용하여 특정 EC2 인스턴스의 최대 CPU 사용률을 확인하는 방법을 보여 줍니다.

요구 사항

- 인스턴스의 ID가 필요합니다. 인스턴스 ID는 AWS Management 콘솔이나 [describe-instances](#) 명령을 사용하여 확인할 수 있습니다.
- 기본적으로 기본 모니터링이 사용되지만 세부 모니터링을 사용하도록 설정할 수 있습니다. 자세한 내용은 [인스턴스에 대한 세부 모니터링 활성화 또는 비활성화 \(p. 667\)](#) 단원을 참조하십시오.

특정 인스턴스에 대한 CPU 사용률을 표시하려면(콘솔)

- <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
- 탐색 창에서 지표를 선택합니다.
- EC2 측정치 네임스페이스를 선택합니다.

The screenshot shows the AWS CloudWatch Metrics console interface. At the top, there are three tabs: "All metrics" (highlighted in orange), "Graphed metrics" (selected), and "Graph options". Below the tabs is a search bar with the placeholder text "Search for any metric, dimension or resource id". The main area displays a grid of six categories, each with a count of metrics:

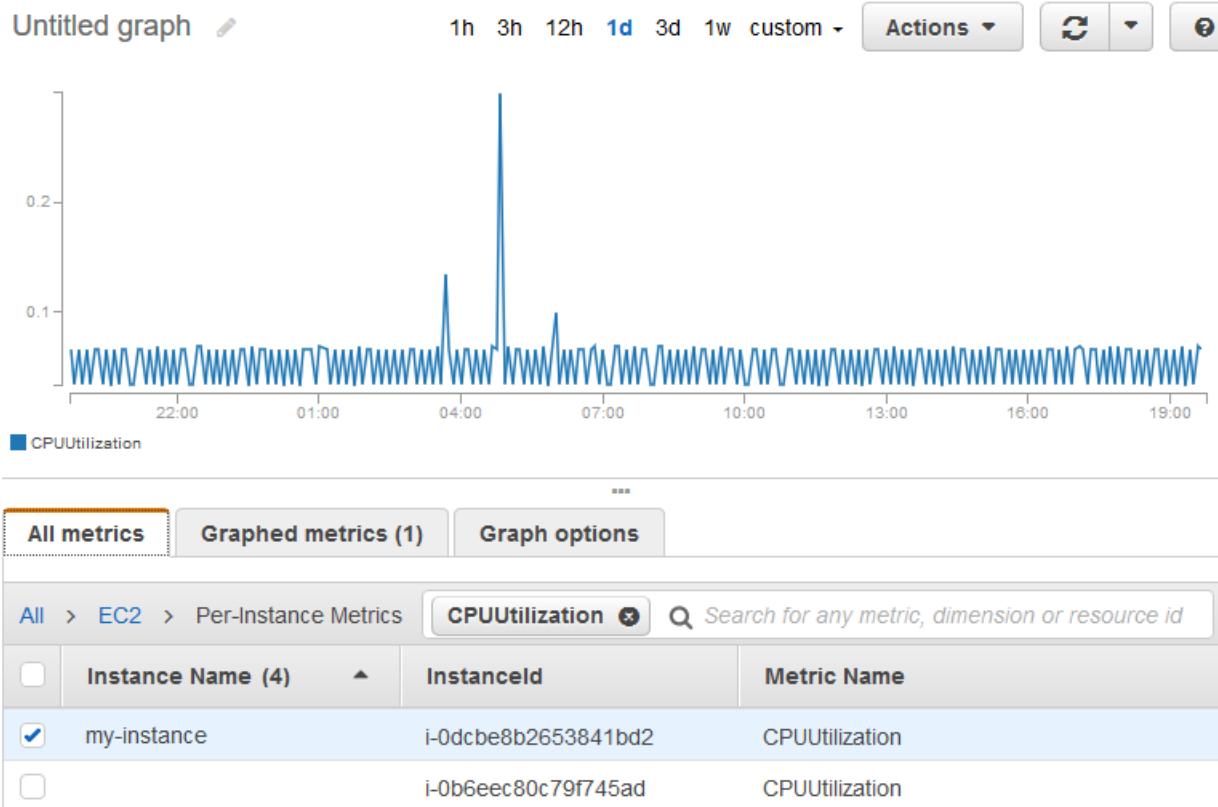
- EBS: 117 Metrics
- EC2: 316 Metrics
- EFS: 7 Metrics
- ELB: 210 Metrics
- ElasticBeanstalk: 8 Metrics
- RDS: 60 Metrics
- S3: 4 Metrics

- 인스턴스별 지표 차원을 선택합니다.

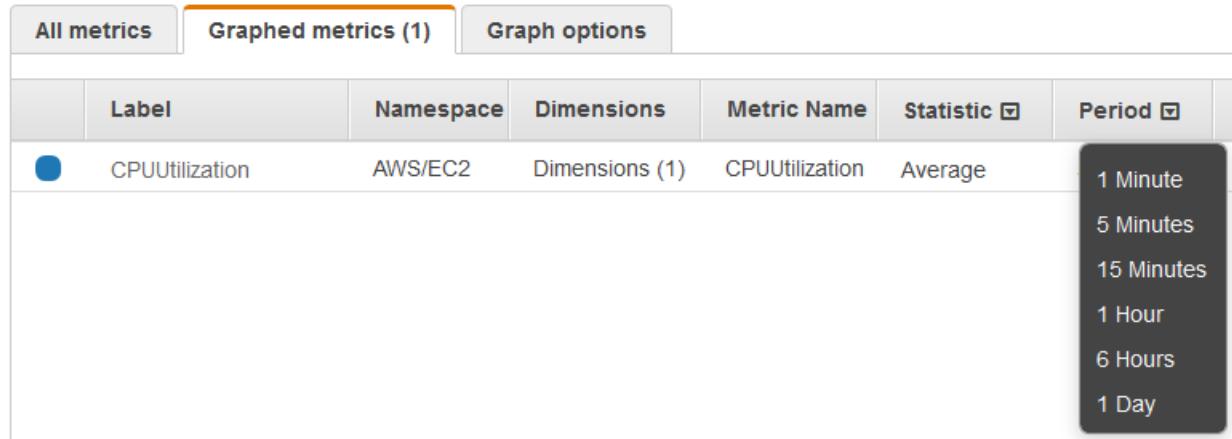
The screenshot shows the Amazon CloudWatch Metrics console interface. At the top, there are three tabs: "All metrics" (selected), "Graphed metrics", and "Graph options". Below the tabs, a breadcrumb navigation shows "All > EC2" and a search bar with the placeholder "Search for any metric, dimension or resource id". The main content area displays "103 Metrics" and is organized into five categories:

- By Auto Scaling Group**: 28 Metrics
- By Image (AMI) Id**: 7 Metrics
- Per-Instance Metrics**: 54 Metrics
- Aggregated by Instance Type**: 7 Metrics
- Across All Instances**: 7 Metrics

5. 검색 필드에 **CPUutilization**을 입력하고 Enter를 누릅니다. 특정 인스턴스의 행을 선택합니다. 그러면 해당 인스턴스의 CPUUtilization 측정치 그라프가 표시됩니다. 그래프 이름을 지정하려면 연필 아이콘을 선택합니다. 시간 범위를 변경하려면 제공되는 값 중 하나를 선택하거나 사용자 지정을 선택합니다.



6. 측정치에 대한 통계 또는 기간을 변경하려면 그래프로 표시된 지표 탭을 선택합니다. 열 머리글이나 개별 값을 선택한 후 다른 값을 선택합니다.



특정 인스턴스에 대한 CPU 사용률을 확인하려면(AWS CLI)

다음 `get-metric-statistics` 명령을 사용하여, 지정된 기간 및 시간 간격을 사용하는 지정된 인스턴스의 CPUUtilization 측정치를 확인합니다.

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name CPUUtilization --
period 3600 \
--statistics Maximum --dimensions Name=InstanceId,Value=i-1234567890abcdef0 \
--start-time 2016-10-18T23:18:00 --end-time 2016-10-19T23:18:00
```

다음은 예제 출력입니다. 각 값은 단일 EC2 인스턴스에 대한 최대 CPU 사용률을 나타냅니다.

```
{  
    "Datapoints": [  
        {  
            "Timestamp": "2016-10-19T00:18:00Z",  
            "Maximum": 0.3300000000000002,  
            "Unit": "Percent"  
        },  
        {  
            "Timestamp": "2016-10-19T03:18:00Z",  
            "Maximum": 99.67000000000002,  
            "Unit": "Percent"  
        },  
        {  
            "Timestamp": "2016-10-19T07:18:00Z",  
            "Maximum": 0.3400000000000002,  
            "Unit": "Percent"  
        },  
        {  
            "Timestamp": "2016-10-19T12:18:00Z",  
            "Maximum": 0.3400000000000002,  
            "Unit": "Percent"  
        },  
        ...  
    ],  
    "Label": "CPUUtilization"  
}
```

여러 인스턴스의 통계 집계

세부 모니터링이 활성화된 인스턴스에 대해서만 통계를 집계할 수 있습니다. 기본 모니터링을 사용하는 인스턴스는 집계에 포함되지 않습니다. 또한 Amazon CloudWatch에서는 리전 간 데이터는 집계하지 않습니다. 따라서 지표는 리전 간에 완전히 개별적입니다. 인스턴스 간에 집계된 통계를 얻으려면 1분 기간의 데이터를 제공하는 세부 모니터링(추가 비용 발생)을 활성화해야 합니다.

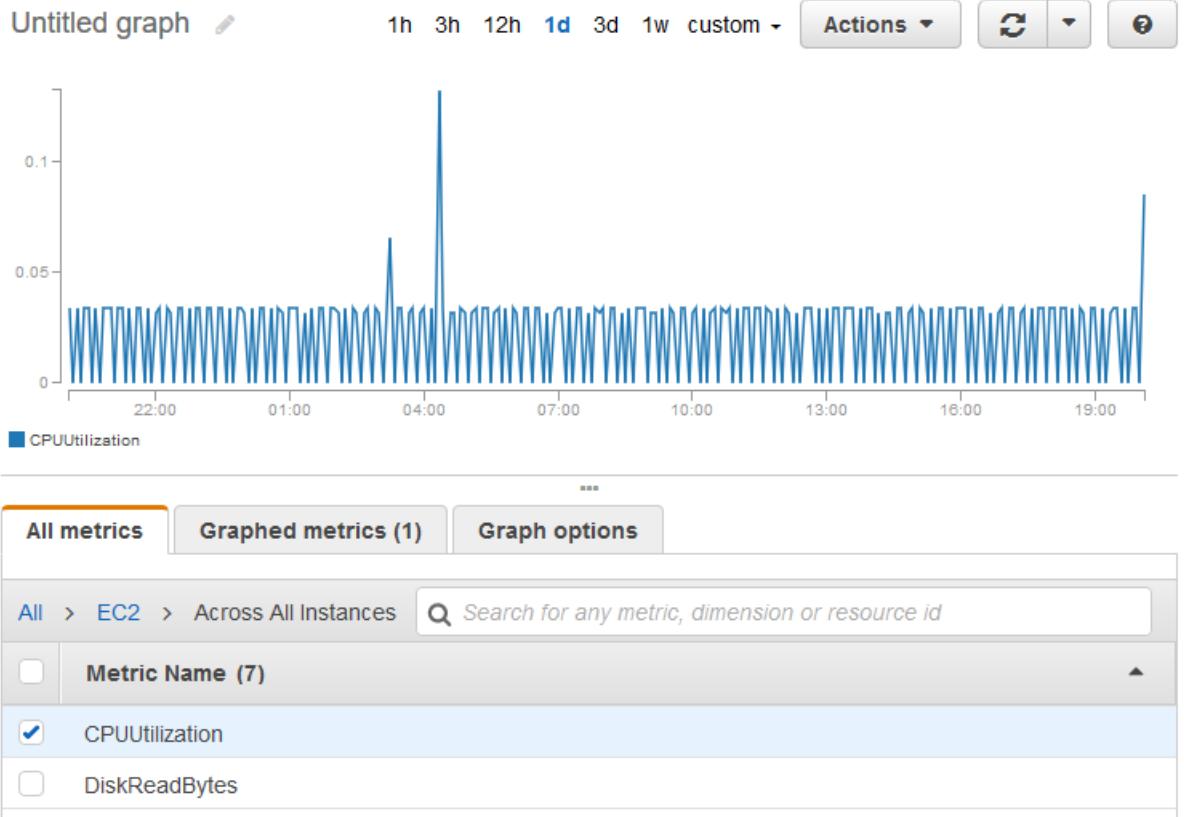
이 예제는 세부 모니터링을 사용하여 EC2 인스턴스의 평균 CPU 사용량을 확인하는 방법을 보여 줍니다. 지정된 차원이 없으므로 CloudWatch에서는 AWS/EC2 네임스페이스의 모든 차원에 대한 통계를 반환합니다.

Important

AWS 네임스페이스에서 모든 차원을 검색하는 기능은 Amazon CloudWatch에 게시한 사용자 지정 네임스페이스에 대해서는 작동하지 않습니다. 사용자 지정 네임스페이스를 사용하는 경우 데이터 요소가 포함된 통계를 검색하려면 특정 데이터 요소와 연결된 전체 차원 세트를 지정해야 합니다.

인스턴스 전반에 걸친 평균 CPU 사용률을 표시하려면(콘솔)

1. <https://console.aws.amazon.com/cloudwatch>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 지표를 선택합니다.
3. EC2 네임스페이스를 선택한 후 전체 인스턴스를 선택합니다.
4. CPUUtilization을 포함하는 행을 선택합니다. 그러면 모든 EC2 인스턴스에 대한 지표 그래프가 표시됩니다. 그래프 이름을 지정하려면 연필 아이콘을 선택합니다. 시간 범위를 변경하려면 제공되는 값 중 하나를 선택하거나 사용자 지정을 선택합니다.



5. 측정치에 대한 통계 또는 기간을 변경하려면 그래프로 표시된 지표 탭을 선택합니다. 열 머리글이나 개별 값을 선택한 후 다른 값을 선택합니다.

인스턴스 간 평균 CPU 사용률을 얻으려면(AWS CLI)

다음과 같이 `get-metric-statistics` 명령을 사용하여 인스턴스에 대한 평균 CPUUtilization 측정치를 확인합니다.

```
aws cloudwatch get-metric-statistics \
--namespace AWS/EC2 \
--metric-name CPUUtilization \
--period 3600 --statistics "Average" "SampleCount" \
--start-time 2016-10-11T23:18:00 \
--end-time 2016-10-12T23:18:00
```

다음은 예제 출력입니다.

```
{
  "Datapoints": [
    {
      "SampleCount": 238.0,
      "Timestamp": "2016-10-12T07:18:00Z",
      "Average": 0.038235294117647062,
      "Unit": "Percent"
    },
    {
      "SampleCount": 240.0,
      "Timestamp": "2016-10-12T09:18:00Z",
      "Average": 0.16670833333333332,
      "Unit": "Percent"
    }
  ]
}
```

```
},
{
  "SampleCount": 238.0,
  "Timestamp": "2016-10-11T23:18:00Z",
  "Average": 0.041596638655462197,
  "Unit": "Percent"
},
...
],
"Label": "CPUUtilization"
}
```

Auto Scaling 그룹별 통계 집계

EC2 인스턴스에 대한 통계를 하나의 Auto Scaling 그룹에 집계할 수 있습니다. Amazon CloudWatch는 리전 전체의 데이터는 집계할 수 없습니다. 지표는 리전별로 개별적입니다.

이 예제는 하나의 Auto Scaling 그룹에 대해 디스크에 기록되는 총 바이트 수를 확인하는 방법을 보여 줍니다. 이 값은 지정한 Auto Scaling 그룹의 모든 EC2 인스턴스에 대해 24시간 간격으로 1분 기간에 대해 계산됩니다.

Auto Scaling 그룹의 인스턴스에 대한 DiskWriteBytes를 보려면(콘솔)

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 지표를 선택합니다.
3. EC2 네임스페이스를 선택한 후 Auto Scaling 그룹별을 선택합니다.
4. DiskWriteBytes 측정치의 행과 특정 Auto Scaling 그룹을 선택합니다. 그러면 해당 Auto Scaling 그룹의 인스턴스에 대한 측정치 그래프가 표시됩니다. 그래프 이름을 지정하려면 연필 아이콘을 선택합니다. 시간 범위를 변경하려면 제공되는 값 중 하나를 선택하거나 사용자 지정을 선택합니다.
5. 측정치에 대한 통계 또는 기간을 변경하려면 그래프로 표시된 지표 탭을 선택합니다. 열 머리글이나 개별 값을 선택한 후 다른 값을 선택합니다.

Auto Scaling 그룹의 인스턴스에 대한 DiskWriteBytes를 보려면(AWS CLI)

다음과 같이 `get-metric-statistics` 명령을 사용합니다.

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name DiskWriteBytes --
period 360 \
--statistics "Sum" "SampleCount" --dimensions Name=AutoScalingGroupName,Value=my-asg --
start-time 2016-10-16T23:18:00 --end-time 2016-10-18T23:18:00
```

다음은 예제 출력입니다.

```
{
  "Datapoints": [
    {
      "SampleCount": 18.0,
      "Timestamp": "2016-10-19T21:36:00Z",
      "Sum": 0.0,
      "Unit": "Bytes"
    },
    {
      "SampleCount": 5.0,
      "Timestamp": "2016-10-19T21:42:00Z",
      "Sum": 0.0,
      "Unit": "Bytes"
    }
  ]
}
```

```
        ],
        "Label": "DiskWriteBytes"
    }
```

AMI별 집계 통계

세부 모니터링이 활성화된 인스턴스에 대해 통계를 집계할 수 있습니다. 기본 모니터링을 사용하는 인스턴스는 포함되지 않습니다. Amazon CloudWatch는 리전 전체의 데이터는 집계할 수 없습니다. 지표는 리전별로 개별적입니다.

인스턴스 간에 집계된 통계를 얻으려면 1분 기간의 데이터를 제공하는 세부 모니터링(추가 비용 발생)을 활성화해야 합니다. 자세한 내용은 [인스턴스에 대한 세부 모니터링 활성화 또는 비활성화 \(p. 667\)](#) 단원을 참조하십시오.

이 예제는 특정 Amazon 머신 이미지(AMI)를 사용하는 모든 인스턴스의 평균 CPU 사용률을 확인하는 방법을 보여 줍니다. 평균은 1일 기간의 60초 시간 간격에 대한 평균입니다.

AMI의 평균 CPU 사용률을 보려면(콘솔)

1. <https://console.aws.amazon.com/cloudwatch>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 지표를 선택합니다.
3. EC2 네임스페이스를 선택한 후 이미지(AMI) ID별을 선택합니다.
4. CPUUtilization 측정치 행과 특정 AMI를 선택합니다. 그러면 지정한 AMI의 그래프가 표시됩니다. 그래프 이름을 지정하려면 연필 아이콘을 선택합니다. 시간 범위를 변경하려면 제공되는 값 중 하나를 선택하거나 사용자 지정을 선택합니다.
5. 측정치에 대한 통계 또는 기간을 변경하려면 그래프로 표시된 지표 탭을 선택합니다. 열 머리글이나 개별 값을 선택한 후 다른 값을 선택합니다.

이미지 ID에 대한 평균 CPU 사용률을 얻으려면(AWS CLI)

다음과 같이 `get-metric-statistics` 명령을 사용합니다.

```
aws cloudwatch get-metric-statistics --namespace AWS/ECC2 --metric-name CPUUtilization --period 3600 \
--statistics Average --dimensions Name=ImageId,Value=ami-3c47a355 --start-time 2016-10-10T00:00:00 --end-time 2016-10-11T00:00:00
```

다음은 예제 출력입니다. 각 값은 지정한 AMI를 실행 중인 EC2 인스턴스의 평균 CPU 사용률을 나타냅니다.

```
{
    "Datapoints": [
        {
            "Timestamp": "2016-10-10T07:00:00Z",
            "Average": 0.04100000000000009,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2016-10-10T14:00:00Z",
            "Average": 0.079579831932773085,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2016-10-10T06:00:00Z",
            "Average": 0.03600000000000011,
            "Unit": "Percent"
        },
        ...
    ]
}
```

```
],
  "Label": "CPUUtilization"
}
```

인스턴스에 대한 그래프 지표

인스턴스를 시작한 후 Amazon EC2 콘솔을 열고 모니터링 탭에서 인스턴스에 대한 모니터링 그래프를 볼 수 있습니다. 각 그래프는 사용 가능한 Amazon EC2 측정치 중 하나를 기반으로 합니다.

다음과 같은 그래프를 사용할 수 있습니다.

- Average CPU Utilization (Percent)
- Average Disk Reads (Bytes)
- Average Disk Writes (Bytes)
- Maximum Network In (Bytes)
- Maximum Network Out (Bytes)
- Summary Disk Read Operations (Count)
- Summary Disk Write Operations (Count)
- Summary Status (Any)
- Summary Status Instance (Count)
- Summary Status System (Count)

측정치와 이러한 측정치가 그래프에 제공하는 데이터에 대한 자세한 내용은 [인스턴스에 사용 가능한 CloudWatch 지표 나열 \(p. 669\)](#) 단원을 참조하십시오.

CloudWatch 콘솔을 사용한 측정치 그래프

CloudWatch 콘솔을 사용하여 Amazon EC2 및 기타 AWS 서비스에서 생성한 측정치 데이터의 그래프를 생성할 수도 있습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서에서 [측정치 그래프](#)를 참조하십시오.

인스턴스에 대해 CloudWatch 경보 만들기

CloudWatch 지표를 모니터링하는 CloudWatch 경보를 생성할 수 있습니다. 지표가 지정된 임계값에 도달하면 CloudWatch에서 자동으로 알림을 보냅니다. Amazon EC2 콘솔이나 CloudWatch 콘솔에 제공된 고급 옵션을 사용하여 CloudWatch 경보를 만들 수 있습니다.

CloudWatch 콘솔을 이용하여 경보 생성하기

구체적인 예시는 Amazon CloudWatch 사용 설명서의 [Amazon CloudWatch 경보 생성](#)을 참조하십시오.

Amazon EC2 콘솔을 이용하여 경보 생성하기

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다.
4. 페이지 하단에 있는 모니터링 탭에서 경보 생성을 선택합니다. 또는 작업 드롭다운에서 CloudWatch 모니터링, 경보 추가/편집을 선택합니다.
5. 경보 생성 대화 상자에서 다음 작업을 수행하십시오.
 - a. 주제 생성을 선택합니다. Send a notification to(알림 보내기)에 SNS 주제의 이름을 입력합니다. 수신자들에게 알림을 수신할 하나 이상의 이메일 주소를 입력합니다.

- b. 정책에 대한 지표와 기준을 지정합니다. 예를 들어 다음 경우 항상(CPU 평균 사용률)를 기본 설정으로 유지할 수 있습니다. ls에서 >=을 선택하고 80%를 입력합니다. 최소 다음의 경우에서 연속 기간으로 1(5 Minutes)을 지정합니다.
- c. 경보 생성을 선택합니다.

Create Alarm

You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you define. To edit an alarm, first choose whom to notify and then define when the notification should be sent.

Send a notification to: my-topic [cancel](#)

With these recipients: me@mycompany.com

Take the action: Recover this instance [i](#)
 Stop this instance [i](#)
 Terminate this instance [i](#)
 Reboot this instance [i](#)

Whenever: Average [▼](#) of CPU Utilization [▼](#)

Is: [>=](#) Percent

For at least: consecutive period(s) of [5 Minutes](#) [▼](#)

Name of alarm: CPU-Utilization

[Cancel](#) [Create Alarm](#)

인스턴스를 중지, 종료, 재부팅 또는 복구하는 경보 만들기

Amazon CloudWatch 경보 작업을 사용하면 인스턴스를 자동으로 중지, 종료, 재부팅 또는 복구하는 경보를 만들 수 있습니다. 인스턴스를 더 이상 실행할 필요가 없을 때 중지 또는 종료 작업을 사용하여 비용을 절약할 수 있습니다. 재부팅 및 복구 작업을 사용하면 시스템 장애가 발생할 경우 인스턴스를 자동으로 재부팅하거나 새로운 하드웨어로 인스턴스를 복구할 수 있습니다.

AWS는 AWSServiceRoleForCloudWatchEvents 서비스 연결 역할을 통해 사용자를 대신하여 경보 작업을 수행할 수 있습니다. AWS Management 콘솔에서 처음으로 경보 생성 시 IAM CLI, 혹은 IAM API, CloudWatch는 사용자를 대신해 서비스 연결 역할을 생성합니다.

인스턴스를 자동으로 중지하거나 종료해야 하는 경우는 매우 다양합니다. 예를 들어 일정 기간 동안 실행한 다음 작업을 완료하는 일괄 급여 처리 작업 또는 과학적 컴퓨팅 작업 전용 인스턴스가 있을 수 있습니다. 이러한 인스턴스를 유지 상태로 유지하여 비용이 발생하도록 하는 대신 중지하거나 종료하면 비용을 절감할 수 있습니다. 경보 작업 중지와 종료 간의 주요 차이는 나중에 다시 실행해야 하는 경우 중지된 인스턴스는 쉽게 다시 시작할 수 있고 동일한 인스턴스 ID 및 루트 볼륨을 유지할 수 있다는 점입니다. 그러나 종료된 인스턴스를 다시 시작할 수는 없습니다. 대신, 새 인스턴스를 시작해야 합니다.

Amazon CloudWatch에서 제공하는 기본 및 세부 모니터링 지표(AWS/Ec2 네임스페이스)를 비롯한 인스턴스 측정치당 Amazon EC2 및 InstanceId 값이 실행 중인 유효한 Amazon EC2 인스턴스를 참조하는 경우 차원을 포함하는 모든 사용자 지정 지표에 대해 설정된 경보에 중지, 종료, 재부팅 또는 복구 작업을 추가할 수 있습니다.

콘솔 지원

Amazon EC2 콘솔 또는 CloudWatch 콘솔을 사용하여 경보를 만들 수 있습니다. 이 문서의 절차는 Amazon EC2 콘솔을 사용합니다. CloudWatch 콘솔을 사용하는 절차는 Amazon CloudWatch 사용 설명서의 [인스턴스를 중지, 종료, 재부팅 또는 복구하는 경보 생성](#)을 참조하십시오.

권한

AWS Identity and Access Management(IAM) 사용자인 경우 경보를 만들거나 수정하려면 다음과 같은 권한이 있어야 합니다.

- `iam:CreateServiceLinkedRole`, `iam:GetPolicy`, `iam:GetPolicyVersion`, 및 `iam:GetRole` - Amazon EC2 작업을 수반하는 모든 경보
- `ec2:DescribeInstanceStatus` 및 `ec2:DescribeInstances` - Amazon EC2 인스턴스 상태 지표에 대한 모든 경보 -
- `ec2:StopInstances` - 중지 작업을 수반하는 경보 -
- `ec2:TerminateInstances` - 종료 작업을 수반하는 경보 -
- 복구 작업을 수반하는 경보는 권한 제한이 없습니다.

읽기/쓰기 권한이 Amazon CloudWatch에 대해서는 있지만 Amazon EC2에 대해서는 없는 경우 경보를 만들 수는 있지만 Amazon EC2 인스턴스에 대해 중지 또는 종료 작업이 수행되지 않습니다. 그러나 이후에 연결된 Amazon EC2 API를 사용하도록 권한을 부여 받은 경우 앞서 만든 경보 작업이 수행됩니다. IAM 권한에 대한 자세한 내용은 IAM 사용 설명서의 [정책 및 권한](#)을 참조하십시오.

목차

- [Amazon CloudWatch 경보에 중지 작업 추가](#) (p. 689)
- [Amazon CloudWatch 경보에 종료 작업 추가](#) (p. 690)
- [Amazon CloudWatch 경보에 재부팅 작업 추가](#) (p. 691)
- [Amazon CloudWatch 경보에 복구 작업 추가](#) (p. 691)
- [Amazon CloudWatch 콘솔을 사용하여 경보 및 작업 기록 보기](#) (p. 693)
- [Amazon CloudWatch 경보 작업 시나리오](#) (p. 693)

Amazon CloudWatch 경보에 중지 작업 추가

특정 임계값에 도달한 경우 Amazon EC2 인스턴스를 중지하는 경보를 만들 수 있습니다. 예를 들어 개발 또는 테스트 인스턴스를 실행한 후 종료하는 것을 잊을 수 있습니다. 24시간 동안 평균 CPU 사용률이 10% 아래로 떨어지는 경우 즉, 유휴 상태로 더 이상 사용되지 않는 경우 트리거되는 경보를 만들 수 있습니다. 필요에 맞춰 임계값 및 기간을 조정할 수 있습니다. 또한 경보가 트리거되면 이메일을 받을 수 있도록 Amazon Simple Notification Service(Amazon SNS) 알림을 추가할 수 있습니다.

Amazon EBS 볼륨을 루트 디바이스로 사용하는 인스턴스는 중지하거나 종료할 수 있지만, 인스턴스 스토어를 루트 디바이스로 사용하는 인스턴스는 종료만 할 수 있습니다.

유휴 인스턴스를 중지하는 경보를 생성하려면(Amazon EC2 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 모니터링 탭에서 경보 생성을 선택합니다.
4. 경보 생성 대화 상자에서 다음 작업을 수행하십시오.

- a. 경보가 트리거될 때 이메일을 받으려면 알림 보내기에서 기존 Amazon SNS 주제를 선택하거나, 주제 생성을 선택하여 새 주제를 만듭니다.

새 주제를 만들려면 Send a notification to(알림 보내기)에 주제 이름을 입력한 다음 수신자들에게 수신자의 이메일 주소를 쉼표로 구분하여 입력합니다. 경보를 만든 후에는 이 주제에 대한 알림을 받으려면 먼저 수락해야 하는 구독 확인 이메일이 전송됩니다.

- b. 다음 작업을 수행할 선택하고 이 인스턴스를 중지합니다를 선택합니다.
c. 다음 경우 항상에서 사용하려는 통계를 선택한 다음 지표를 선택합니다. 이 예에서는 평균 및 CPU 사용률을 선택합니다.
d. 결과 값에 지표 임계값을 지정합니다. 이 예에서는 10%를 입력합니다.
e. 최소 다음의 경우에 경보의 평가 기간을 지정합니다. 이 예에서는 24 연속 기간(1시간)을 입력합니다.
f. 경보 이름을 변경하려면 경보 이름에 새 이름을 입력합니다. 경보 이름은 ASCII 문자만 포함해야 합니다.

경보 이름을 입력하지 않으면 Amazon CloudWatch에서 이름이 자동으로 지정됩니다.

Note

경보 구성은 경보를 만들기 전에 요구사항에 따라 조정하거나 나중에 편집할 수 있습니다. 이러한 구성에는 메트릭, 임계값, 기간, 작업 및 알림 설정이 있습니다. 그러나 경보를 만든 후에는 경보 이름은 편집할 수 없습니다.

- g. 경보 생성을 선택합니다.

Amazon CloudWatch 경보에 종료 작업 추가

인스턴스에 대해 종료 보호가 비활성화되어 있는 경우에 한해서 특정 임계값에 도달한 경우 EC2 인스턴스를 자동으로 종료하는 경보를 만들 수 있습니다. 예를 들어 인스턴스의 작업 완료 후 해당 인스턴스가 다시 필요 없는 경우 인스턴스를 종료하려고 할 수 있습니다. 나중에 인스턴스를 사용하려는 경우에는 종료하지 말고 중지해야 합니다. 인스턴스 종료 방지 기능의 활성화/비활성화에 대한 자세한 내용은 [종료 방지 기능 활성화 \(p. 456\)](#) 단원을 참조하십시오.

유형 인스턴스를 종료하는 경보를 생성하려면(Amazon EC2 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 모니터링 탭에서 경보 생성을 선택합니다.
4. 경보 생성 대화 상자에서 다음 작업을 수행하십시오.

- a. 경보가 트리거될 때 이메일을 받으려면 알림 보내기에서 기존 Amazon SNS 주제를 선택하거나, 주제 생성을 선택하여 새 주제를 만듭니다.

새 주제를 만들려면 Send a notification to(알림 보내기)에 주제 이름을 입력한 다음 수신자들에게 수신자의 이메일 주소를 쉼표로 구분하여 입력합니다. 경보를 만든 후에는 이 주제에 대한 알림을 받으려면 먼저 수락해야 하는 구독 확인 이메일이 전송됩니다.

- b. 다음 작업을 수행할 선택하고 이 인스턴스를 종료합니다를 선택합니다.
c. 다음 경우 항상에서 사용하려는 통계를 선택한 다음 지표를 선택합니다. 이 예에서는 평균 및 CPU 사용률을 선택합니다.
d. 결과 값에 지표 임계값을 지정합니다. 이 예에서는 10%를 입력합니다.
e. 최소 다음의 경우에 경보의 평가 기간을 지정합니다. 이 예에서는 24 연속 기간(1시간)을 입력합니다.
f. 경보 이름을 변경하려면 경보 이름에 새 이름을 입력합니다. 경보 이름은 ASCII 문자만 포함해야 합니다.

경보 이름을 입력하지 않으면 Amazon CloudWatch에서 이름이 자동으로 지정됩니다.

Note

경보 구성은 경보를 만들기 전에 요구사항에 따라 조정하거나 나중에 편집할 수 있습니다. 이러한 구성에는 메트릭, 임계값, 기간, 작업 및 알림 설정이 있습니다. 그러나 경보를 만든 후에는 경보 이름은 편집할 수 없습니다.

- g. 경보 생성을 선택합니다.

Amazon CloudWatch 경보에 재부팅 작업 추가

Amazon EC2 인스턴스를 모니터링하고 인스턴스를 자동으로 재부팅하는 Amazon CloudWatch 경보를 만들 수 있습니다. 재부팅 경보 작업은 인스턴스 상태 확인 오류(복구 경보 작업은 시스템 상태 확인 오류에 적합)에 권장됩니다. 인스턴스 재부팅은 운영 체제 재부팅과 같습니다. 대부분의 경우 인스턴스를 재부팅하는데는 몇 분 밖에 걸리지 않습니다. 인스턴스를 재부팅하는 경우 동일한 물리적 호스트에 남아 있으므로 퍼블릭 DNS 이름, 프라이빗 IP 주소 및 인스턴스 스토어 볼륨의 모든 데이터가 유지됩니다.

인스턴스를 재부팅해도 인스턴스를 중지했다가 다시 시작할 때와는 달리 인스턴스 청구 시간이 새로 시작되지 않습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [인스턴스 재부팅](#)을 참조하십시오.

Important

재부팅과 복원 작업 간에 경합 상태가 발생하지 않도록 하려면 재부팅 경보와 복원 경보에 동일한 평가 기간 값을 설정하지 마십시오. 재부팅 경보를 각각 1분의 평가 기간 3회로 설정하는 것이 좋습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서의 [경보 평가](#)를 참조하십시오.

인스턴스를 재부팅하는 경보를 생성하려면(Amazon EC2 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 모니터링 탭에서 경보 생성을 선택합니다.
4. 경보 생성 대화 상자에서 다음 작업을 수행하십시오.
 - a. 경보가 트리거될 때 이메일을 받으려면 알림 보내기에서 기존 Amazon SNS 주제를 선택하거나, 주제 생성을 선택하여 새 주제를 만듭니다.

새 주제를 만들려면 Send a notification to(알림 보내기)에 주제 이름을 입력한 다음 수신자들에게에 수신자의 이메일 주소를 쉼표로 구분하여 입력합니다. 경보를 만든 후에는 이 주제에 대한 알림을 받으려면 먼저 수락해야 하는 구독 확인 이메일이 전송됩니다.

 - b. 다음 작업을 선택하고 이 인스턴스를 재부팅합니다를 선택합니다.
 - c. 다음 경우 항상에서 상태 확인 실패(인스턴스)를 선택합니다.
 - d. 최소 다음의 경우에 경보의 평가 기간을 지정합니다. 이 예에서는 3 연속 기간(1분)을 입력합니다.
 - e. 경보 이름을 변경하려면 경보 이름에 새 이름을 입력합니다. 경보 이름은 ASCII 문자만 포함해야 합니다.

경보 이름을 입력하지 않으면 Amazon CloudWatch에서 이름이 자동으로 지정됩니다.
 - f. 경보 생성을 선택합니다.

Amazon CloudWatch 경보에 복구 작업 추가

Amazon EC2 인스턴스를 모니터링하는 Amazon CloudWatch 경보를 만들 수 있습니다. 기본 하드웨어 장애나 복구에 AWS 개입이 필요한 문제로 인해 인스턴스가 손상된 경우 해당 인스턴스를 자동으로 복구할 수 있

습니다. 종료한 인스턴스는 복구할 수 없습니다. 복구된 인스턴스는 인스턴스 ID, 프라이빗 IP 주소, 탄력적 IP 주소 및 모든 인스턴스 메타데이터를 포함하여 원본 인스턴스와 동일합니다.

CloudWatch은 복구 작업을 지원하지 않는 인스턴스에 대한 경보에 복구 작업을 추가할 수 없게 합니다.

`StatusCheckFailed_System` 경보가 트리거되고 복구 작업이 시작되는 경우 경보를 만들고 복구 작업을 연결할 때 선택한 Amazon SNS 주제별로 통지됩니다. 인스턴스 복구 중에 인스턴스를 재부팅할 때 인스턴스가 마이그레이션되고 모든 인 메모리 데이터가 손실됩니다. 프로세스가 완료되면 해당 경보를 위해 구성해둔 SNS 주제로 정보가 게시됩니다. 이 SNS 주제에 가입되어 있는 사람은 누구나 복구 시도 상태와 세부 지침이 포함된 이메일 알림을 받게 됩니다. 복구된 인스턴스에서 인스턴스를 재부팅하라는 메시지가 나타납니다.

복구 작업은 `StatusCheckFailed_Instance`가 아닌 `StatusCheckFailed_System`을 통해서만 사용할 수 있습니다.

다음과 같은 문제가 있을 경우 시스템 상태 확인이 실패할 수 있습니다.

- 네트워크 연결 끊김
- 시스템 전원 중단
- 물리적 호스트의 소프트웨어 문제
- 네트워크 연결성에 영향을 주는 물리적 호스트의 하드웨어 문제

복구 작업은 다음 특성을 지닌 인스턴스에만 지원됩니다.

- 게이트웨이 유형에 대한 권장 인스턴스 유형 C3, C4, C5, C5a, C5n, M3, M4, M5, M5a, M5n, P3, R3, R4, R5, R5a, R5n, T2, T3, T3a, X1 또는 X1e 중 하나를 사용합니다.
- `default` 또는 `dedicated` 인스턴스 테넌시 사용
- EBS 볼륨(인스턴스 스토어 볼륨을 구성하지 않음)만 사용합니다. 자세한 내용은 '[이 인스턴스를 복구합니다'가 비활성화됨 단원](#)을 참조하십시오.

인스턴스에 퍼블릭 IP 주소가 있는 경우 복구 후에도 해당 퍼블릭 IP 주소를 유지합니다.

Important

재부팅과 복원 작업 간에 경합 상태가 발생하지 않도록 하려면 재부팅 경보와 복원 경보에 동일한 평가 기간 값을 설정하지 마십시오. 복구 경보는 각각 1분의 평가 기간 2회로 설정하는 것이 좋습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서의 [경보 평가](#)를 참조하십시오.

인스턴스를 복구하는 경보를 생성하려면(Amazon EC2 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 모니터링 탭에서 경보 생성을 선택합니다.
4. 경보 생성 대화 상자에서 다음 작업을 수행하십시오.
 - a. 경보가 트리거될 때 이메일을 받으려면 알림 보내기에서 기존 Amazon SNS 주제를 선택하거나, 주제 생성을 선택하여 새 주제를 만듭니다.

새 주제를 만들려면 Send a notification to(알림 보내기)에 주제 이름을 입력한 다음 수신자들에게 수신자의 이메일 주소를 쉼표로 구분하여 입력합니다. 경보를 만든 후에는 이 주제에 대한 이메일을 받기 전에 수락해야 하는 구독 확인 이메일이 전송됩니다.

Note

- 경보가 트리거될 때 이메일 알림을 수신하려면 사용자는 지정된 SNS 주제를 구독해야 합니다.

- AWS 계정의 루트 사용자는 SNS 주제가 지정되어 있지 않더라도 자동 인스턴스 복구 작업이 발생하면 항상 이메일 알림을 받습니다.
 - AWS 계정의 루트 사용자는 지정된 SNS 주제를 구독하지 않더라도 자동 인스턴스 복구 작업이 발생하면 항상 이메일 알림을 받습니다.
- b. 다음 작업을 수행을 선택하고 이 인스턴스를 복구합니다를 선택합니다.
 - c. 다음 경우 항상에서 상태 확인 실패(시스템)을 선택합니다.
 - d. 최소 다음의 경우에 경보의 평가 기간을 지정합니다. 이 예에서는 2 연속 기간(1분)을 입력합니다.
 - e. 경보 이름을 변경하려면 경보 이름에 새 이름을 입력합니다. 경보 이름은 ASCII 문자만 포함해야 합니다.
- 경보 이름을 입력하지 않으면 Amazon CloudWatch에서 이름이 자동으로 지정됩니다.
- f. 경보 생성을 선택합니다.

Amazon CloudWatch 콘솔을 사용하여 경보 및 작업 기록 보기

Amazon CloudWatch 콘솔에서 경보 및 작업 기록을 볼 수 있습니다. Amazon CloudWatch에서는 지난 2주 간의 경보 및 작업 기록을 보관합니다.

트리거된 경보 및 작업 기록을 보려면(CloudWatch 콘솔)

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 Alarms를 선택합니다.
3. 경보를 선택합니다.
4. 세부 정보 탭에 최근 상태 변화가 시간 및 지표 값과 함께 표시됩니다.
5. 최근 기록 항목을 보려면 기록 탭을 선택합니다.

Amazon CloudWatch 경보 작업 시나리오

Amazon EC2 콘솔을 사용하여 특정 조건이 충족되면 Amazon EC2 인스턴스를 중지하거나 종료하는 경보 작업을 만들 수 있습니다. 경보 작업을 설정하는 콘솔 페이지의 다음 화면 캡처에서는 설정에 번호가 표시되어 있습니다. 또한 적절한 작업을 만드는 데 도움이 되도록 시나리오의 설정에도 번호를 표시했습니다.

Create Alarm

You can use CloudWatch alarms to be notified automatically whenever metric data reaches a level you specify. To edit an alarm, first choose whom to notify and then define when the notification should be sent.

① Send a notification to: [create topic](#)

② Take the action: Recover this instance [i](#)
 Stop this instance [i](#)
 Terminate this instance [i](#)
 Reboot this instance [i](#)

Whenever: of
Is: Percent

For at least: consecutive period(s) of

Name of alarm:

시나리오 1: 유휴 개발 및 테스트 인스턴스 중지

소프트웨어 개발 및 테스트에 사용된 인스턴스가 한 시간 이상 유휴 상태인 경우 중지하는 경보를 만듭니다.

설정	값
1	Stop
2	Maximum
3	CPUUtilization
4	<=
5	10%
6	60분
7	1

시나리오 2: 유휴 인스턴스 중지

인스턴스가 24시간 동안 유휴 상태인 경우 인스턴스를 중지하고 이메일을 보내는 경보를 만듭니다.

설정	값
1	Stop and email
2	Average
3	CPUUtilization
4	<=
5	5%
6	60분
7	24

시나리오 3: 트래픽이 비정상적으로 높은 웹 서버에 대해 이메일 보내기

인스턴스가 일일 아웃바운드 네트워크 트래픽인 10GB를 초과하는 경우 이메일을 보내는 경보를 만듭니다.

설정	값
1	이메일
2	Sum
3	NetworkOut
4	>
5	10GB
6	1 day
7	1

시나리오 4: 트래픽이 비정상적으로 높은 웹 서버 중지

아웃바운드 트래픽이 시간당 1GB를 초과하는 경우 인스턴스를 중지하고 문자 메시지(SMS)를 보내는 경보를 만듭니다.

설정	값
1	Stop and send SMS
2	Sum
3	NetworkOut
4	>
5	1GB

설정	값
6	1시간
7	1

시나리오 5: 메모리 누수가 발생하는 인스턴스 중지

메모리 사용률이 90%에 도달했거나 90%를 초과한 경우 인스턴스를 중지하는 경보를 만듭니다. 그러면 문제 해결을 위해 애플리케이션 로그를 검색할 수 있습니다.

Note

MemoryUtilization 지표는 사용자 지정 지표입니다. MemoryUtilization 지표를 사용 하려면 Linux 인스턴스용 Perl 스크립트를 설치해야 합니다. 자세한 내용은 [Amazon EC2 Linux 인스턴스에 대한 메모리 및 디스크 지표 모니터링](#)을 참조하십시오.

설정	값
1	Stop
2	Maximum
3	MemoryUtilization
4	>=
5	90%
6	1분
7	1

시나리오 6: 손상된 인스턴스 중지

5분 간격으로 수행된 연속 3회의 상태 확인에 실패한 인스턴스를 중지하는 경보를 만듭니다.

설정	값
1	Stop
2	Average
3	StatusCheckFailed_System
4	>=
5	1
6	15 minutes
7	1

시나리오 7: 배치 처리 작업이 완료되면 인스턴스 종료

결과 데이터를 더 이상 보내지 않는 경우 일괄 작업을 실행하는 인스턴스를 종료하는 경보를 만듭니다.

설정	값
1	Terminate
2	Maximum
3	NetworkOut
4	<=
5	100,000 bytes
6	5 minutes
7	1

CloudWatch 이벤트를 사용한 Amazon EC2 자동화

Amazon CloudWatch Events를 사용하여 AWS 서비스를 자동화하고 애플리케이션 가용성 문제나 리소스 변경 같은 시스템 이벤트에 자동으로 응답합니다. AWS 서비스 이벤트는 거의 실시간으로 CloudWatch 이벤트로 전송됩니다. 원하는 이벤트만 표시하도록 간단한 규칙을 정의한 후 규칙과 일치하는 이벤트 발생 시 실행할 자동 작업을 지정할 수 있습니다. 자동으로 트리거할 수 있는 작업은 다음과 같습니다.

- AWS Lambda 함수 호출
- Amazon EC2 Run Command 호출
- Amazon Kinesis Data Streams로 이벤트 릴레이
- AWS Step Functions 상태 머신 활성화
- Amazon SNS 주제 또는 Amazon SQS 대기열 알림

다음은 CloudWatch 이벤트를 Amazon EC2에 사용하는 몇 가지 예입니다.

- 새로운 Amazon EC2 인스턴스를 시작할 때마다 Lambda 함수를 활성화합니다.
- Amazon EBS 볼륨을 생성하거나 수정할 때 Amazon SNS 주제를 알립니다.
- 다른 AWS 서비스에서 특정 이벤트 발생 시 Amazon EC2 Run Command를 사용하여 명령을 하나 이상의 Amazon EC2 인스턴스에 전송합니다.

자세한 내용은 [Amazon CloudWatch Events 사용 설명서](#)를 참조하십시오.

AWS CloudTrail을 사용하여 Amazon EC2 및 Amazon EBS API 호출 로깅

Amazon EC2 및 Amazon EBS는 Amazon EC2 및 Amazon EBS의 사용자, 역할 또는 AWS 서비스가 수행한 작업에 대한 레코드를 제공하는 서비스인 AWS CloudTrail과 통합됩니다. CloudTrail은 콘솔의 호출 및 API 코드 호출 등 Amazon EC2 및 Amazon EBS에 대한 모든 API 호출을 이벤트로 캡처합니다. 추적을 생성하면, Amazon EC2 및 Amazon EBS 이벤트를 비롯하여 CloudTrail 이벤트를 Amazon S3 버킷으로 지속적으로 배포할 수 있습니다. 추적을 구성하지 않은 경우 이벤트 기록에서 CloudTrail 콘솔의 최신 이벤트를 볼 수도 있습니다. CloudTrail에서 수집하는 정보를 사용하여 Amazon EC2 및 Amazon EBS에 수행된 요청, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다.

CloudTrail에 대한 자세한 내용은 [AWS CloudTrail User Guide](#)을 참조하십시오.

CloudTrail의 Amazon EC2 및 Amazon EBS 정보

CloudTrail은 계정 생성 시 AWS 계정에서 활성화됩니다. Amazon EC2 및 Amazon EBS에서 활동이 수행되면 해당 활동은 이벤트 기록에서 다른 AWS 서비스 이벤트와 함께 CloudTrail 이벤트에 기록됩니다. AWS 계정에서 최신 이벤트를 확인, 검색 및 다운로드할 수 있습니다. 자세한 내용은 [CloudTrail 이벤트 기록에서 이벤트 보기](#)를 참조하십시오.

Amazon EC2 및 Amazon EBS 이벤트를 비롯하여 AWS 계정의 이벤트 기록을 보유하려면 추적을 생성하십시오. 추적은 CloudTrail이 Amazon S3 버킷으로 로그 파일을 전송할 수 있도록 합니다. 콘솔에서 추적을 생성하면 기본적으로 모든 리전에 추적이 적용됩니다. 추적은 AWS 파티션에 있는 모든 리전의 이벤트를 로깅하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 또는 CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음을 참조하십시오.

- [추적 생성 개요](#)
- [CloudTrail 지원 서비스 및 통합](#)
- [CloudTrail에 대한 Amazon SNS 알림 구성](#)
- [여러 리전에서 CloudTrail 로그 파일 받기 및 여러 계정에서 CloudTrail 로그 파일 받기](#)

모든 Amazon EC2 작업과 Amazon EBS 관리 작업은 CloudTrail에서 로깅되고 [Amazon EC2 API Reference](#)에 문서화됩니다. 예를 들어 [RunInstances](#), [DescribeInstances](#) 또는 [CreateImage](#) 작업을 호출하면 CloudTrail 로그 파일에 항목이 생성됩니다.

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 자격 증명 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트로 했는지 아니면 IAM 사용자 자격 증명으로 했는지 여부
- 역할 또는 연합된 사용자에 대한 임시 보안 자격 증명을 사용하여 요청이 생성되었는지 여부.
- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 [CloudTrail userIdentity 요소](#)를 참조하십시오.

Amazon EC2 및 Amazon EBS 로그 파일 항목 이해

추적은 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 제공할 수 있도록 해 주는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함됩니다. 이벤트는 어떤 소스로부터의 단일 요청을 나타내며 요청된 작업, 작업 날짜와 시간, 요청 파라미터 등에 대한 정보가 들어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 호출의 순서가 지정된 스택 추적이 아니기 때문에 특정 순서로 표시되지 않습니다.

다음 로그 파일 레코드는 사용자가 인스턴스를 종료했음을 보여 줍니다.

```
{  
    "Records": [  
        {  
            "eventVersion": "1.03",  
            "userIdentity": {  
                "type": "Root",  
                "principalId": "123456789012",  
                "arn": "arn:aws:iam::123456789012:root",  
                "accountId": "123456789012",  
                "accessKeyId": "AKIAIOSFODNN7EXAMPLE",  
                "userName": "user"  
            },  
            "eventTime": "2016-05-20T08:27:45Z",  
            "eventSource": "ec2.amazonaws.com",  
            "eventName": "TerminateInstances",  
            "awsRegion": "us-west-2",  
            "version": "1"  
        }  
    ]  
}
```

```
"sourceIPAddress":"198.51.100.1",
"userAgent":"aws-cli/1.10.10 Python/2.7.9 Windows/7botocore/1.4.1",
"requestParameters":{
    "instancesSet": {
        "items": [
            "instanceId":"i-1a2b3c4d"
        ]
    }
},
"responseElements": {
    "instancesSet": {
        "items": [
            "instanceId":"i-1a2b3c4d",
            "currentState": {
                "code":32,
                "name":"shutting-down"
            },
            "previousState": {
                "code":16,
                "name":"running"
            }
        ]
    }
},
"requestID":"be112233-1ba5-4ae0-8e2b-1c302EXAMPLE",
"eventID":"6e12345-2a4e-417c-aa78-7594fEXAMPLE",
"eventType":"AwsApiCall",
"recipientAccountId":"123456789012"
}
]
```

AWS CloudTrail을 사용하여 EC2 Instance Connect를 통해 연결하는 사용자 감사

AWS CloudTrail을 사용하여 EC2 Instance Connect를 통해 인스턴스에 연결하는 사용자를 감사합니다.

AWS CloudTrail 콘솔을 사용하여 EC2 Instance Connect를 통해 SSH 활동을 감사하려면

1. <https://console.aws.amazon.com/cloudtrail/>에서 AWS CloudTrail 콘솔을 엽니다.
2. 올바른 리전에 있는지 확인합니다.
3. 탐색 창에서 Event history(이벤트 내역)를 선택합니다.
4. Filter(필터)에서 Event source(이벤트 소스), ec2-instance-connect.amazonaws.com을 선택합니다.
5. (선택 사항) Time range(시간 범위)에서 시간 범위를 선택합니다.
6. Refresh events(새로 고침 이벤트) 아이콘을 선택합니다.
7. 이 페이지에는 `SendSSHPublicKey` API 호출에 해당하는 이벤트가 표시됩니다. 화살표를 사용하여 이벤트를 확장하면 SSH 연결을 만드는 데 사용된 사용자 이름, AWS 액세스 키 및 소스 IP 주소와 같은 추가 세부 정보를 볼 수 있습니다.
8. 전체 이벤트 정보를 JSON 형식으로 표시하려면 View event(이벤트 보기)를 선택합니다.
requestParameters 필드에는 SSH 연결을 만드는 데 사용된 대상 인스턴스 ID, OS 사용자 이름 및 퍼블릭 키가 포함되어 있습니다.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDEFGGNOMOOOCB6XYTQEXAMPLE",
        "arn": "arn:aws:iam::1234567890120:user/IAM-friendly-name",
    }
}
```

```
"accountId": "123456789012",
"accessKeyId": "ABCDEFGUZHNAW4OSN2AEXAMPLE",
"userName": "IAM-friendly-name",
"sessionContext": {
    "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-09-21T21:37:58Z"
    }
},
"eventTime": "2018-09-21T21:38:00Z",
"eventSource": "ec2-instance-connect.amazonaws.com",
"eventName": "SendSSHPublicKey",
"awsRegion": "us-west-2",
"sourceIPAddress": "123.456.789.012",
"userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
"requestParameters": {
    "instanceId": "i-0123456789EXAMPLE",
    "osUser": "ec2-user",
    "SSHKey": {
        "publicKey": "ssh-rsa ABCDEFGHIJKLMNOP01234567890EXAMPLE"
    }
},
"responseElements": null,
"requestID": "1a2s3d4f-bde6-11e8-a892-f7ec64543add",
"eventID": "1a2w3d4r5-a88f-4e28-b3bf-30161f75be34",
"eventType": "AwsApiCall",
"recipientAccountId": "0987654321"
}
```

S3 버킷에서 CloudTrail 이벤트를 수집하도록 AWS 계정을 구성한 경우 프로그래밍 방식으로 정보를 다운로드하고 감사할 수 있습니다. 자세한 내용은 AWS CloudTrail User Guide의 [CloudTrail 로그 파일 가져오기 및 보기](#)를 참조하십시오.

CloudWatch Application Insights를 사용하여 .NET 및 SQL Server 애플리케이션 모니터링

.NET 및 SQL Server용 CloudWatch Application Insights를 사용하면 다른 [AWS 애플리케이션 리소스](#)와 함께 Amazon EC2 인스턴스를 사용하는 .NET 및 SQL Server 애플리케이션을 모니터링할 수 있습니다. 이 기능은 애플리케이션 리소스 및 기술 스택(예: Microsoft SQL Server 데이터베이스, 웹(IIS) 및 애플리케이션 서버, OS, 로드 밸런서, 대기열 등) 전반에서 주요 지표, 로그 및 경보를 파악하고 설정합니다. 지표 및 로그를 지속적으로 모니터링하여 이상 및 오류를 감지하고 연결합니다. 오류 및 이상이 감지되면 Application Insights에서 알림을 설정하고 작업을 수행할 수 있는 [CloudWatch 이벤트](#)를 생성합니다. 문제 해결을 돋기 위해 감지한 문제에 대한 자동 대시보드를 생성함으로써 상관관계가 있는 지표 이상 항목 및 로그 오류와 함께 잠재적인 근본 원인을 알려주는 추가 통찰력을 제공합니다. 자동화된 대시보드를 사용하면 애플리케이션의 상태를 정상으로 유지하고 애플리케이션의 최종 사용자에게 미치는 영향을 방지하기 위해 신속한 조치를 취할 수 있습니다.

지원되는 로그 및 지표의 전체 목록을 보려면 [.NET 및 SQL Server용 Amazon CloudWatch Application Insights에서 지원되는 로그 및 지표](#)를 참조하십시오.

감지된 문제에 대해 다음과 같은 정보가 제공됩니다.

- 문제의 간략한 요약
- 문제의 시작 시간 및 날짜
- 문제의 심각도: 높음/중간/낮음
- 감지된 문제의 상태: 진행 중/해결됨
- 통찰력: 감지된 문제 및 가능한 근본 원인에 대해 자동으로 생성된 통찰력

- 통찰력에 대한 피드백: .NET 및 SQL Server용 CloudWatch Application Insights에서 생성한 통찰력의 유용성에 대한 피드백
- 관련 관찰: 다양한 애플리케이션 구성 요소에서 발생한 문제와 관련된 로그의 오류 조각 및 지표 이상의 상세 보기

피드백

감지된 문제에 대해 자동으로 생성된 통찰력에 대한 유용성을 평가하여 피드백을 제공할 수 있습니다. 통찰력에 대한 피드백은 애플리케이션 진단(지표 이상 및 로그 예외)과 함께 향후 유사한 문제의 탐지를 개선하는데 사용됩니다.

자세한 내용은 Amazon CloudWatch 사용 설명서의 [.NET 및 SQL Server용 CloudWatch Application Insights 설명서](#)를 참조하십시오.

Amazon EC2의 네트워킹

Amazon EC2는 다음과 같은 네트워킹 기능을 제공합니다.

기능

- Amazon EC2 인스턴스 IP 주소 지정 (p. 702)
- Amazon EC2의 고유 IP 주소 가져오기(BYOIP) (p. 716)
- 탄력적인 IP 주소 (p. 721)
- 탄력적 네트워크 인터페이스 (p. 729)
- Windows에서 향상된 네트워킹 (p. 751)
- 배치 그룹 (p. 761)
- EC2 인스턴스에 대한 네트워크 MTU(최대 전송 단위) (p. 772)
- 가상 사설 클라우드 (p. 776)
- Windows Amazon 머신 이미지(AMI)용 포트 및 프로토콜 (p. 777)
- EC2-Classic (p. 802)

Amazon EC2 인스턴스 IP 주소 지정

Amazon EC2와 Amazon VPC는 IPv4 및 IPv6 주소 지정 프로토콜을 모두 지원합니다. Amazon EC2와 Amazon VPC는 IPv4 주소 지정 프로토콜을 사용하도록 기본 설정되어 있으며 이 동작은 비활성화할 수 없습니다. VPC를 생성할 때 VPC에 IPv4 CIDR 블록(프라이빗 IPv4 주소)을 지정해야 합니다. IPv6 CIDR 블록을 VPC와 서브넷에 할당하고 그 블록에 속한 IPv6 주소를 서브넷의 인스턴스에 할당할 수도 있습니다. IPv6 주소는 인터넷으로 접속할 수 있습니다. IPv6에 대한 자세한 내용은 Amazon VPC 사용 설명서의 [VPC에서 IP 주소 지정](#) 단원을 참조하십시오.

목차

- [프라이빗 IPv4 주소 및 내부 DNS 호스트 이름](#) (p. 702)
- [퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름](#) (p. 703)
- [탄력적 IP 주소\(IPv4\)](#) (p. 704)
- [Amazon DNS 서비스](#) (p. 704)
- [IPv6 주소](#) (p. 704)
- [인스턴스에 대한 IP 주소 작업](#) (p. 705)
- [다중 IP 주소](#) (p. 709)

프라이빗 IPv4 주소 및 내부 DNS 호스트 이름

프라이빗 IPv4 주소는 인터넷을 통해 연결할 수 없는 IP 주소입니다. 프라이빗 IPv4 주소는 동일 VPC에서 인스턴스 간의 통신을 위해 사용될 수 있습니다. 프라이빗 IPv4 주소의 표준 및 사양에 대한 자세한 내용은 [RFC 1918](#)을 참조하십시오. DHCP를 사용하여 개인 IPv4 주소를 인스턴스에 할당합니다.

Note

RFC 1918에 지정된 프라이빗 IPv4 주소 범위에 속하지 않는 공개적으로 라우팅 가능한 CIDR 블록을 사용하여 VPC를 생성할 수 있습니다. 하지만 이 설명서에서 프라이빗 IPv4 주소(또는 프라이빗 IP 주소)는 VPC의 IPv4 CIDR 범위 내에 있는 IP 주소를 말합니다.

인스턴스를 시작할 때 인스턴스에 기본 프라이빗 IPv4 주소가 할당됩니다. 또한 각 인스턴스에는 `ip-10-251-50-12.ec2.internal`과 같이 기본 프라이빗 IP 주소를 확인하는 내부 DNS 호스트 이름이

할당됩니다. 동일한 VPC에 있는 인스턴스 간의 통신에 내부 DNS 호스트 이름을 사용할 수 있지만, VPC 외부에 있는 내부 DNS 호스트 이름을 확인할 수 없습니다.

인스턴스에는 서브넷 IPv4 주소 범위 내의 기본 프라이빗 IP 주소가 할당됩니다. 자세한 내용은 Amazon VPC 사용 설명서의 [VPC 및 서브넷 크기 조정](#)을 참조하십시오. 인스턴스 시작 시 사용자가 기본 프라이빗 IP 주소를 지정하지 않으면 사용자 서브넷 IPv4 범위 내의 IP 주소가 할당됩니다. 각 인스턴스는 기본 프라이빗 IPv4 주소가 할당된 기본 네트워크 인터페이스(eth0)를 갖습니다. 또한, 사용자는 보조 프라이빗 IPv4 주소라는 추가 프라이빗 IPv4 주소를 지정할 수 있습니다. 기본 프라이빗 IP 주소와 달리, 보조 프라이빗 IP 주소는 한 인스턴스에서 다른 인스턴스로 재할당될 수 있습니다. 자세한 내용은 [다중 IP 주소 \(p. 709\)](#) 단원을 참조하십시오.

프라이빗 IPv4 주소는 기본 주소인지 보조 주소인지 관계없이 인스턴스가 중지되었다가 시작될 때 또는 최대 절전 모드로 전환되었다가 시작될 때 네트워크 인터페이스와 연결이 유지되고 인스턴스가 종료되면 릴리스됩니다.

퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름

퍼블릭 IP 주소는 인터넷을 통해 연결할 수 있는 IPv4 주소입니다. 퍼블릭 주소는 인스턴스와 인터넷의 상호 통신을 위해 사용될 수 있습니다.

또한, 퍼블릭 IP 주소가 할당된 각 인스턴스에는 외부 DNS 호스트 이름이 할당됩니다. 예: `ec2-203-0-113-25.compute-1.amazonaws.com`. AWS에서는 외부 DNS 호스트 이름을 VPC 외부 인스턴스의 퍼블릭 IP 주소 및 VPC 내부 인스턴스의 프라이빗 IPv4 주소로 확인합니다. 퍼블릭 IP 주소는 네트워크 주소 변환(NAT)을 통해 기본 프라이빗 IP 주소로 매핑됩니다. 자세한 내용은 [RFC 1631: IP 네트워크 주소 변환기\(NAT\)](#)를 참조하십시오.

기본 VPC에서 인스턴스를 시작할 때 기본적으로 퍼블릭 IP 주소가 할당됩니다. 기본 VPC가 아닌 VPC로 인스턴스를 시작하는 경우 서브넷은 이 서브넷으로 시작되는 인스턴스가 퍼블릭 IPv4 주소 풀로부터 퍼블릭 IP 주소를 부여받는지 여부를 결정하는 속성을 갖습니다. 기본적으로 기본 서브넷이 아닌 서브넷에서 시작된 인스턴스에 퍼블릭 IP 주소가 할당되지 않습니다.

다음과 같이 인스턴스가 퍼블릭 IP 주소를 수신할지 여부를 제어할 수 있습니다.

- 서브넷의 퍼블릭 IP 주소 지정 속성 수정. 자세한 내용은 Amazon VPC 사용 설명서의 [서브넷의 퍼블릭 IPv4 주소 지정 속성 수정](#) 단원을 참조하십시오.
- 시작 시 퍼블릭 IP 주소 지정 기능을 활성화 또는 비활성화(서브넷의 퍼블릭 IP 주소 지정 속성 재정의). 자세한 내용은 [인스턴스 시작 시 퍼블릭 IPv4 주소 할당 \(p. 706\)](#) 단원을 참조하십시오.

퍼블릭 IP 주소는 Amazon의 퍼블릭 IPv4 주소 풀에서 사용자 인스턴스로 지정되고 AWS 계정과는 관련이 없습니다. 인스턴스와 퍼블릭 IP 주소의 연결이 해제되면 해당 퍼블릭 IP 주소는 퍼블릭 IPv4 주소 풀로 해제되지만 사용자가 해당 주소를 다시 사용할 수 없습니다.

사용자는 인스턴스에서 퍼블릭 IP 주소를 수동으로 연결 또는 해제할 수 없습니다. 어떤 경우에는 Amazon에서 귀하의 인스턴스로부터 퍼블릭 IP 주소를 해제하거나 새 인스턴스에 할당합니다:

- 인스턴스가 중지되거나 최대 절전 모드로 전환되거나 종료되면 인스턴스의 퍼블릭 IP 주소는 릴리스됩니다. 중지되거나 최대 절전 모드로 전환된 인스턴스가 시작되면 새 퍼블릭 IP 주소가 할당됩니다.
- 탄력적 IP 주소를 인스턴스와 연결하는 경우 인스턴스의 퍼블릭 IP 주소가 릴리스됩니다. 사용자가 인스턴스에서 탄력적 IP 주소의 연결을 해제하면 새 퍼블릭 IP 주소가 할당됩니다.
- VPC 인스턴스의 퍼블릭 IP 주소가 해제되고 인스턴스에 1개 이상의 네트워크 인터페이스가 연결된 경우 새 퍼블릭 IP 주소가 할당되지 않습니다.
- 인스턴스의 퍼블릭 IP 주소가 릴리스된 가운데 탄력적 IP 주소와 연결된 보조 프라이빗 IP 주소를 보유한 경우 인스턴스는 새 퍼블릭 IP 주소를 수신하지 않습니다.

필요에 따라 인스턴스 간에 연결할 수 있는 영구 퍼블릭 IP 주소가 필요한 경우 탄력적 IP 주소를 대신하여 사용합니다.

동적 DNS를 사용하여 새 인스턴스의 퍼블릭 IP 주소에 기존 DNS 이름을 연결하는 경우 IP 주소가 인터넷을 통해 전해지는 데 24시간까지 걸릴 수 있습니다. 따라서 종료된 인스턴스가 요청을 계속 받는 동안 새 인스턴스가 트래픽을 받지 못할 수 있습니다. 이 문제를 해결하려면 탄력적 IP 주소를 사용합니다. 사용자는 고유 탄력적 IP 주소를 할당하고 인스턴스와 연결할 수 있습니다. 자세한 내용은 [탄력적인 IP 주소 \(p. 721\)](#) 단원을 참조하십시오.

탄력적 IP 주소가 인스턴스에 할당된 경우 IPv4 DNS 호스트 이름이 활성화되어 있으면 인스턴스에 DNS 호스트 이름이 할당됩니다. 자세한 내용은 Amazon VPC 사용 설명서의 [VPC에서 DNS 사용](#)을 참조하십시오.

Note

인스턴스가 동일 리전에 존재하는지 여부에 따라 퍼블릭 NAT IP 주소를 통해 다른 인스턴스에 액세스하는 인스턴스에는 리전별 또는 인터넷 데이터 전송 비용이 청구됩니다.

탄력적 IP 주소(IPv4)

탄력적 IP 주소는 사용자가 계정에 연결할 수 있는 퍼블릭 IPv4 주소입니다. 사용자는 필요에 따라 인스턴스 간에 연결할 수 있고 해제되기 전까지는 사용자 계정에 할당됩니다. 엘라스틱 IP 주소 및 사용 방법에 대한 자세한 내용은 [탄력적인 IP 주소 \(p. 721\)](#) 단원을 참조하십시오.

IPv6에 대한 탄력적 IP 주소는 지원하지 않습니다.

Amazon DNS 서버

Amazon은 Amazon이 제공한 IPv4 DNS 호스트 이름을 IPv4 주소로 변환하는 DNS 서버를 제공합니다. Amazon DNS 서버는 사용자 VPC 네트워크 범위 +2의 범위에 위치합니다. 자세한 내용은 Amazon VPC 사용 설명서의 [Amazon DNS 서버](#) 단원을 참조하십시오.

IPv6 주소

IPv6 CIDR 블록과 VPC를 연결하고 IPv6 CIDR 블록과 서브넷을 연결할 수도 있습니다. VPC에 대한 IPv6 CIDR 블록은 Amazon의 IPv6 주소 풀에서 자동으로 할당되므로 범위를 직접 선택할 수 없습니다. 자세한 내용은 Amazon VPC 사용 설명서에서 다음 주제를 참조하십시오.

- [IPv6의 경우, VPC 및 서브넷 크기 조정](#)
- [IPv6 CIDR 블록을 VPC와 연결](#)
- [IPv6 CIDR 블록을 서브넷에 연결](#)

IPv6 주소는 전역적으로 고유하므로 인터넷으로 접속할 수 있습니다. IPv6 CIDR 블록이 VPC와 서브넷에 연결되어 있고 다음 중 하나가 true이면 인스턴스는 IPv6 주소를 받습니다.

- 서브넷은 시작 중인 인스턴스에 IPv6 주소를 자동으로 할당하도록 구성됩니다. 자세한 내용은 [서브넷의 IPv6 주소 지정 속성 수정](#) 단원을 참조하십시오.
- 시작하는 동안 인스턴스에 IPv6 주소를 할당합니다.
- 시작 후 인스턴스의 기본 네트워크 인터페이스에 IPv6 주소를 할당합니다.
- 동일 서브넷에서 네트워크 인터페이스에 IPv6 주소를 할당하고 시작을 완료한 후에 인스턴스에 네트워크 인터페이스를 연결합니다.

시작하는 과정에서 인스턴스가 IPv6 주소를 받는 경우, 해당 주소는 인스턴스의 기본 네트워크 인터페이스 (eth0)와 연결됩니다. 네트워크 인터페이스에서 IPv6 주소 연결을 해제할 수 있습니다. 인스턴스에 대해서는 IPv6 DNS 호스트 이름을 지원하지 않습니다.

인스턴스를 중지했다가 시작할 때 또는 최대 절전 모드로 전환했다가 시작할 때에는 IPv6 주소가 지속되다가 인스턴스를 종료하면 릴리스됩니다. IPv6 주소는 다른 네트워크 인터페이스에 할당되는 동안에는 재할당할 수 없으므로, 먼저 할당을 해제해야 합니다.—

인스턴스에 연결된 네트워크 인터페이스에 IPv6 주소를 할당함으로써 인스턴스에 추가 IPv6 주소를 할당할 수 있습니다. 네트워크 인터페이스에 할당할 수 있는 IPv6 주소의 개수, 그리고 인스턴스에 연결할 수 있는 네트워크 인터페이스의 개수는 인스턴스 유형에 따라 달라집니다. 자세한 내용은 [인스턴스 유형별 네트워크 인터페이스당 IP 주소 \(p. 730\)](#) 단원을 참조하십시오.

인스턴스에 대한 IP 주소 작업

인스턴스에 할당된 IP 주소를 확인하고, 시작 중에 퍼블릭 IPv4 주소를 인스턴스에 할당하며, 시작 중에 IPv6 주소를 인스턴스에 할당할 수 있습니다.

목차

- [퍼블릭, 프라이빗, 탄력적 IP 주소 결정 \(p. 705\)](#)
- [IPv6 주소 결정 \(p. 706\)](#)
- [인스턴스 시작 시 퍼블릭 IPv4 주소 할당 \(p. 706\)](#)
- [인스턴스에 IPv6 주소 할당 \(p. 707\)](#)
- [인스턴스에 할당된 IPv6 주소 해제 \(p. 708\)](#)

퍼블릭, 프라이빗, 탄력적 IP 주소 결정

사용자는 Amazon EC2 콘솔을 사용하여 인스턴스의 프라이빗 IPv4 주소, 퍼블릭 IPv4 주소 및 탄력적 IP 주소를 결정할 수 있습니다. 또한, 사용자는 인스턴스 메타데이터를 사용하여 인스턴스 내에서 인스턴스의 퍼블릭 IPv4 및 프라이빗 IPv4 주소를 결정할 수 있습니다. 자세한 내용은 [인스턴스 메타데이터 및 사용자 데이터 \(p. 576\)](#) 단원을 참조하십시오.

콘솔을 이용하여 인스턴스의 프라이빗 IPv4 주소를 결정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 세부 정보 창의 프라이빗 IP 필드에서 프라이빗 IPv4 주소를 획득하고 프라이빗 DNS 필드에서 내부 DNS 호스트 이름을 획득합니다.
4. 인스턴스에 연결된 네트워크 인터페이스에 1개 이상의 보조 프라이빗 IPv4 주소가 할당된 경우 해당 IP 주소는 보조 프라이빗 IP 필드에서 획득할 수 있습니다.
5. 아니면, 탐색 창에서 네트워크 인터페이스를 선택한 후 인스턴스에 연결된 네트워크 인터페이스를 선택합니다.
6. 기본 프라이빗 IPv4 IP 필드에서 기본 프라이빗 IP 주소를 획득하고 프라이빗 DNS(IPv4) 필드에서 내부 DNS 호스트 이름을 획득합니다.
7. 네트워크 인터페이스에 보조 프라이빗 IP 주소가 할당된 경우 해당 IP 주소는 보조 프라이빗 IPv4 IP에서 획득할 수 있습니다.

콘솔을 이용하여 인스턴스의 퍼블릭 IPv4 주소를 결정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 세부 정보 창의 IPv4 퍼블릭 IP 필드에서 퍼블릭 IP 주소를 획득하고 퍼블릭 DNS(IPv4) 필드에서 외부 DNS 호스트 이름을 획득합니다.
4. 탄력적 IP 주소가 하나 이상 인스턴스에 연결되어 있는 경우에는 탄력적 IP 필드에서 탄력적 IP 주소를 획득합니다.

Note

인스턴스가 퍼블릭 IPv4 주소를 가지고 있지 않지만 사용자가 인스턴스에서 탄력적 IP 주소를 네트워크 인터페이스에 연결한 경우에는 IPv4 퍼블릭 IP 필드에 탄력적 IP 주소가 표시됩니다.

5. 아니면, 탐색 창에서 네트워크 인터페이스를 선택한 후 인스턴스에 연결된 네트워크 인터페이스를 선택합니다.
6. IPv4 퍼블릭 IP 필드에서 퍼블릭 IP 주소를 획득합니다. 별표시(*)는 기본 프라이빗 IPv4 주소와 매핑된 퍼블릭 IPv4 주소 또는 탄력적 IP 주소를 나타냅니다.

Note

퍼블릭 IPv4 주소는 콘솔에서 네트워크 인터페이스의 속성으로 표시되지만 NAT를 통해 기본 프라이빗 IPv4 주소와 매핑됩니다. 그러므로, 예를 들어 ifconfig(Linux) 또는 ipconfig(Windows)를 통해 인스턴스 네트워크 카드의 속성을 확인하는 경우 퍼블릭 IPv4 주소는 표시되지 않습니다. 인스턴스의 퍼블릭 IPv4 주소를 인스턴스 내에서 결정하려면 인스턴스 메타데이터를 사용할 수 있습니다.

인스턴스 메타데이터를 이용하여 인스턴스의 IPv4 주소를 결정하려면

1. 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
2. 다음 명령을 사용하여 프라이빗 IP 주소에 액세스합니다.

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/local-ipv4
```

3. 다음 명령을 사용하여 퍼블릭 IP 주소에 액세스합니다.

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/public-ipv4
```

인스턴스와 탄력적 IP 주소가 연결된 경우 반환된 값은 탄력적 IP 주소입니다.

IPv6 주소 결정

Amazon EC2 콘솔을 사용하여 인스턴스의 IPv6 주소를 결정할 수 있습니다.

콘솔을 이용하여 인스턴스의 IPv6 주소를 결정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 세부 정보 창의 IPv6 IP에서 IPv6 주소를 가져옵니다.

인스턴스 메타데이터를 이용하여 인스턴스의 IPv6 주소를 결정하려면

1. 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
2. 다음 명령을 사용하여 IPv6 주소를 봅니다(<http://169.254.169.254/latest/meta-data/network/interfaces/macs/>에서 MAC 주소를 가져올 수 있음).

```
PS C:\> Invoke-RestMethod http://169.254.169.254/latest/meta-data/network/interfaces/macs/mac-address/ipv6s
```

인스턴스 시작 시 퍼블릭 IPv4 주소 할당

각 서브넷은 퍼블릭 IP 주소가 할당되는 서브넷에서 인스턴스를 시작할지 여부를 결정하는 속성을 갖습니다. 기본적으로 기본이 아닌 서브넷의 이 속성은 false로 설정되고 기본 서브넷의 속성 값은 true입니다. 인스턴스를 시작할 때 퍼블릭 IPv4 주소 지정 기능을 사용하여 인스턴스에 퍼블릭 IPv4 주소가 할당되는지 여부를 제어할 수도 있습니다. 서브넷의 IP 주소 지정 속성의 기본 동작을 재정의할 수 있습니다. 퍼블릭 IPv4 주소

는 Amazon의 퍼블릭 IPv4 주소 풀에서 할당되고 디바이스 색인이 eth0인 네트워크 인터페이스에 할당됩니다. 이 기능은 인스턴스 시작 시점의 특정 조건에 따라 달라집니다.

Important

사용자는 인스턴스가 시작된 이후에는 퍼블릭 IP 주소를 수동으로 해제할 수 없습니다. 대신, 특정 조건에 자동으로 해제되고 그 이후에 사용자는 해당 주소를 다시 사용할 수 없습니다. 자세한 내용은 [퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름 \(p. 703\)](#) 단원을 참조하십시오. 연결 또는 해제할 수 있는 영구 퍼블릭 IP 주소가 필요한 경우 시작 후에 인스턴스에 탄력적 IP 주소를 할당합니다. 자세한 내용은 [탄력적인 IP 주소 \(p. 721\)](#) 단원을 참조하십시오.

인스턴스 시작 시 퍼블릭 IP 주소 지정 기능에 액세스하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 인스턴스 시작을 선택합니다.
3. AMI와 인스턴스 유형을 선택한 후 다음: 인스턴스 세부 정보 구성(Next: Configure Instance Details)을 선택합니다.
4. 인스턴스 세부 정보 구성 페이지의 네트워크에서 VPC를 선택합니다. 퍼블릭 IP 자동 할당 목록이 표시됩니다. 활성화 또는 비활성화를 선택하여 서브넷의 기본 설정을 재정의합니다.

Important

네트워크 인터페이스를 두 개 이상 지정하면 퍼블릭 IP 주소를 자동 할당할 수 없습니다. 또한 eth0에 대해 기존 네트워크 인터페이스를 지정하면 퍼블릭 IP 자동 할당 기능을 사용하여 서브넷 설정을 재정의할 수 없습니다.

5. 마법사의 다음 페이지에서 제공되는 단계를 따라 인스턴스 설정을 완료합니다. 마법사 구성 옵션에 대한 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오. 마지막 인스턴스 시작 검토 페이지에서는 설정을 검토한 후 시작을 선택하여 키 페어를 선택하고 인스턴스를 시작합니다.
6. 인스턴스 페이지에서 새 인스턴스를 선택한 다음 세부 정보 창의 IPv4 퍼블릭 IP 필드에서 퍼블릭 IP 주소를 확인합니다.

퍼블릭 IP 주소 지정 기능은 시작 동안에만 사용 가능합니다. 그러나 시작 도중에 퍼블릭 IP 주소가 인스턴스에 할당되는지의 여부와는 관계없이 시작 후에는 인스턴스와 탄력적 IP 주소를 연결할 수 있습니다. 자세한 내용은 [탄력적인 IP 주소 \(p. 721\)](#) 단원을 참조하십시오. 또한, 사용자는 서브넷의 퍼블릭 IPv4 주소 지정 동작을 변경할 수 있습니다. 자세한 내용은 [서브넷의 퍼블릭 IPv4 주소 지정 속성 수정](#) 단원을 참조하십시오.

명령줄을 사용한 퍼블릭 IP 주소 지정 기능의 활성화 또는 비활성화 방법

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `run-instances` 명령(AWS CLI)에서 `--associate-public-ip-address` 또는 `--no-associate-public-ip-address` 옵션을 사용합니다.
- `New-EC2Instance` 명령(Windows PowerShell용 AWS 도구)에서 `-AssociatePublicIp` 파라미터를 사용합니다.

인스턴스에 IPv6 주소 할당

VPC와 서브넷에 연결된 IPv6 CIDR 블록이 있는 경우, 시작 중 또는 시작 후 인스턴스에 IPv6 주소를 할당할 수 있습니다. IPv6 주소는 서브넷의 IPv6 주소 범위에서 할당되고 디바이스 색인이 eth0인 네트워크 인터페이스에 할당됩니다.

IPv6는 모든 현재 세대 인스턴스 유형과 C3, R3 및 I2 이전 세대 인스턴스 유형에서 지원됩니다.

시작하는 과정에서 인스턴스에 IPv6 주소를 할당하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. IPv6을 지원하는 AMI와 인스턴스 유형을 선택하고 다음: 인스턴스 세부 정보 구성(Next: Configure Instance Details)을 선택합니다.
3. 인스턴스 세부 정보 구성 페이지의 네트워크에서 VPC를 선택하고 서브넷에서 서브넷을 선택합니다. IPv6 IP 자동 할당에 대해 활성화를 선택합니다.
4. 마법사의 나머지 단계를 수행하여 인스턴스를 시작합니다.

또는 시작을 완료한 후 인스턴스에 IPv6 주소를 할당할 수 있습니다.

시작 후 인스턴스에 IPv6 주소를 할당하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, 네트워킹, IP 주소 관리를 선택합니다.
4. IPv6 주소에서 새 IP 할당을 선택합니다. 서브넷 범위에 속한 IPv6 주소를 지정하거나, Amazon이 IPv6 주소를 자동으로 선택하도록 자동 할당 값을 그대로 둡니다.
5. 저장을 선택합니다.

Note

Amazon Linux 2016.09.0 이상 버전 또는 Windows Server 2008 R2 이상 버전을 사용하여 인스턴스를 시작한 경우, 인스턴스는 IPv6에 맞게 구성되어 있으므로 IPv6 주소가 인스턴스에서 인식되는지 추가적으로 확인할 필요가 없습니다. 이전 AMI에서 인스턴스를 시작한 경우 인스턴스를 수동으로 구성해야 할 수 있습니다. 자세한 내용은 Amazon VPC 사용 설명서의 [인스턴스에서 IPv6 구성하기](#) 단원을 참조하십시오.

명령줄을 사용하여 IPv6 주소를 할당하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `run-instances`(AWS CLI) 명령에서 `--ipv6-addresses` 옵션을 사용합니다.
- `New-EC2Instance`(Windows PowerShell용 AWS 도구) 명령에서 `-NetworkInterface`에 대한 `Ipv6Addresses` 속성을 사용합니다.
- `assign-ipv6-addresses`(AWS CLI)
- `Register-EC2Ipv6AddressList`(Windows PowerShell용 AWS 도구)

인스턴스에 할당된 IPv6 주소 해제

Amazon EC2 콘솔을 사용하여 인스턴스에서 IPv6 주소 할당을 해제할 수 있습니다.

인스턴스에서 IPv6 주소 할당 해제

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, 네트워킹, IP 주소 관리를 선택합니다.
4. IPv6 주소에서 할당을 해제할 IPv6 주소에 대해 할당 해제를 선택합니다.
5. 예, 업데이트를 선택합니다.

명령줄을 사용하여 IPv6 주소 할당을 해제하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `unassign-ipv6-addresses`(AWS CLI)
- `Unregister-EC2Ipv6AddressList`(Windows PowerShell용 AWS 도구).

다중 IP 주소

인스턴스에 다중 프라이빗 IPv4 및 IPv6 주소를 지정할 수 있습니다. 인스턴스에 지정할 수 있는 네트워크 인터페이스 및 프라이빗 IPv4 및 IPv6 주소의 수는 인스턴스 유형에 의해 결정됩니다. 자세한 내용은 [인스턴스 유형별 네트워크 인터페이스당 IP 주소 \(p. 730\)](#) 단원을 참조하십시오.

다음을 수행하여 VPC 인스턴스에 다중 IP 주소를 할당할 수 있습니다.

- 단일 서버에서 다중 SSL 인증서를 사용하거나 특정 IP 주소에 각 인증서를 연결하여 단일 서버에 다중 웹 사이트 호스팅.
- 각 네트워크 인터페이스에 다중 IP 주소가 있는 네트워크 어플라이언스(방화벽 또는 로드 밸런서 등) 운영.
- 대기 중인 인스턴스에 보조 IP 주소를 할당하여 인스턴스에서 오류가 발생한 경우 대기 인스턴스로 내부 트래픽 리디렉션.

목차

- [다중 IP 주소 동작 방법 \(p. 709\)](#)
- [다중 IPv4 주소 관련 작업 \(p. 710\)](#)
- [다중 IPv6 주소 관련 작업 \(p. 713\)](#)

다중 IP 주소 동작 방법

다음 목록은 다중 IP 주소를 갖는 네트워크 인터페이스의 동작 방법을 설명합니다.

- 사용자는 모든 네트워크 인터페이스에 보조 프라이빗 IPv4 주소를 할당할 수 있습니다. 네트워크 인터페이스는 인스턴스에 연결하지 않아도 됩니다.
- 연결된 IPv6 CIDR 블록이 있는 서브넷의 네트워크 인터페이스에 다중 IPv6 주소를 할당할 수 있습니다.
- 네트워크 인터페이스의 서브넷 IPv4 주소 CIDR 블록 범위 내에서 보조 IPv4를 선택해야 합니다.
- 네트워크 인터페이스의 서브넷 IPv6 CIDR 블록 범위 내에서 IPv6 주소를 선택해야 합니다.
- 보안 그룹을 개별 IP 주소가 아니라 네트워크 인터페이스와 연결합니다. 따라서 네트워크 인터페이스에 지정한 각 IP 주소가 네트워크 인터페이스의 보안 그룹에 종속됩니다.
- 다중 IP 주소는 실행 중 또는 중지된 인스턴스에 연결된 네트워크 인터페이스에 할당되거나 할당되지 않을 수 있습니다.
- 네트워크 인터페이스에 할당된 보조 프라이빗 IPv4 주소는 사용자가 명시적으로 허용한 경우 다른 네트워크 인터페이스로 재할당될 수 있습니다.
- IPv6 주소는 다른 네트워크 인터페이스에 재할당될 수 없습니다. 우선 기준 네트워크 인터페이스에서 IPv6 주소의 할당을 해제해야 합니다.
- 명령줄 도구 또는 API를 이용하여 네트워크 인터페이스에 IP 주소를 여러 개 할당하는 경우 IP 주소 중 하나를 할당할 수 없으면 전체 작업이 실패하게 됩니다.
- 인스턴스에서 분리되거나 인스턴스에 연결되어도 기본 프라이빗 IPv4 주소, 보조 프라이빗 IPv4 주소, 탄력적 IP 주소 및 IPv6 주소는 보조 네트워크 인터페이스에 연결 상태를 유지합니다.
- 기본 네트워크 인터페이스는 인스턴스에서 분리할 수 없지만 기본 네트워크 인터페이스의 보조 프라이빗 IPv4 주소는 다른 네트워크 인터페이스로 재할당이 가능합니다.

다음 목록은 다중 IP 주소를 갖는 탄력적 IP 주소의 동작 방법을 설명합니다(IPv4만 해당).

- 각 프라이빗 IPv4 주소는 단일 탄력적 IP 주소로 연결될 수 있고 그 반대도 가능합니다.
- 보조 프라이빗 IPv4 주소가 다른 인터페이스로 재할당된 경우 보조 프라이빗 IPv4 주소와 탄력적 IP 주소는 연결 상태를 유지합니다.
- 보조 프라이빗 IPv4 주소가 인터페이스에서 할당이 해제된 경우 연결된 탄력적 IP 주소는 보조 프라이빗 IPv4 주소에서 자동으로 할당이 해제됩니다.

다중 IPv4 주소 관련 작업

보조 프라이빗 IPv4 주소를 인스턴스에 할당하고 탄력적 IPv4 주소와 보조 프라이빗 IPv4 주소를 연결하며, 보조 프라이빗 IPv4 주소의 할당을 해제할 수 있습니다.

목차

- [보조 프라이빗 IPv4 주소 할당 \(p. 710\)](#)
- [보조 프라이빗 IPv4 주소를 인식하도록 인스턴스 운영 체제 구성 \(p. 712\)](#)
- [탄력적 IP 주소와 보조 프라이빗 IPv4 주소 연결 \(p. 712\)](#)
- [보조 프라이빗 IPv4 주소 확인 \(p. 712\)](#)
- [보조 프라이빗 IPv4 주소 할당 해제 \(p. 712\)](#)

보조 프라이빗 IPv4 주소 할당

사용자는 인스턴스 시작 시 또는 인스턴스가 실행된 다음 인스턴스의 네트워크 인터페이스에 보조 프라이빗 IPv4 주소를 할당할 수 있습니다. 이 섹션에는 다음 절차가 포함됩니다.

- [인스턴스를 시작할 때 보조 프라이빗 IPv4 주소를 할당하려면 \(p. 710\)](#)
- [명령줄을 이용하여 시작 중에 보조 IPv4 주소를 할당하려면 \(p. 711\)](#)
- [네트워크 인터페이스에 보조 프라이빗 IPv4 주소를 할당하려면 \(p. 711\)](#)
- [명령줄을 이용하여 기존 인스턴스에 보조 프라이빗 IPv4 주소를 할당하려면 \(p. 711\)](#)

인스턴스를 시작할 때 보조 프라이빗 IPv4 주소를 할당하려면

- [https://console.aws.amazon.com/ec2/에서 Amazon EC2 콘솔을 엽니다.](#)
- 인스턴스 시작을 선택합니다.
- AMI를 선택하고 인스턴스 유형을 선택한 후 다음: 인스턴스 세부 정보 구성(Next: Configure Instance Details)을 선택합니다.
- 인스턴스 세부 정보 구성 페이지의 네트워크에서 VPC를 선택하고 서브넷에서 서브넷을 선택합니다.
- 네트워크 인터페이스 섹션에서 다음을 수행하고 다음: 스토리지 추가(Next: Add Storage)를 선택합니다.
 - 다른 네트워크 인터페이스를 추가하려면 디바이스 추가를 선택합니다. 콘솔을 사용하여 인스턴스 시작 시 네트워크 인터페이스를 최대 두 개 지정할 수 있습니다. 인스턴스를 시작한 후 탐색 창에서 네트워크 인터페이스를 선택하여 네트워크 인터페이스를 추가합니다. 연결 가능한 총 네트워크 인터페이스의 수는 인스턴스 유형에 따라 다릅니다. 자세한 내용은 [인스턴스 유형별 네트워크 인터페이스당 IP 주소 \(p. 730\)](#) 단원을 참조하십시오.

Important

두 번째 네트워크 인터페이스를 추가하면 시스템에서 더 이상 퍼블릭 IPv4 주소를 자동 할당할 수 없습니다. 기본 네트워크 인터페이스(eth0)에 탄력적 IP 주소를 할당하지 않는 이상 IPv4를 통해 인스턴스에 연결할 수 없습니다. 시작 마법사를 완료한 후에는 탄력적 IP 주소를 할당할 수 있습니다. 자세한 내용은 [탄력적 IP 주소 작업 \(p. 722\)](#) 단원을 참조하십시오.

- 각 네트워크 인터페이스에 대해 보조 IP 주소에서 IP 추가를 선택한 후 서브넷 범위 내의 프라이빗 IP 주소를 입력하거나, 기본 설정인 Auto-assign을 수락하여 Amazon의 주소 선택을 허용합니다.
- 다음 스토리지 추가 페이지에서 사용자는 볼륨을 지정하여 AMI에 의해 지정된 볼륨 옆에 인스턴스(루트 디바이스 볼륨 등)를 연결한 다음 다음: 태그 추가(Next: Add Tags)를 선택합니다.
 - 태그 추가 페이지에서 인스턴스에 태그(예: 사용자에게 친숙한 이름)를 지정한 후 다음: 보안 그룹 구성(Next: Configure Security Group)을 선택합니다.
 - 보안 그룹 구성 페이지에서 기존 보안 그룹을 선택하거나 새 보안 그룹을 생성합니다. 검토 및 시작을 선택합니다.
 - 인스턴스 시작 검토 페이지에서 설정을 검토한 후 시작을 선택하여 키 페어를 선택하고 인스턴스를 시작합니다. Amazon EC2를 처음 사용하며 아직 키 페어를 생성하지 않은 경우 키 페어를 생성하라는 메시지가 마법사에 표시됩니다.

Important

네트워크 인터페이스에 보조 프라이빗 IP 주소를 추가한 이후에는 인스턴스에 연결하고 인스턴스 자체에 보조 프라이빗 IP 주소를 구성해야 합니다. 자세한 내용은 [보조 프라이빗 IPv4 주소를 인식하도록 인스턴스 운영 체제 구성 \(p. 712\)](#) 단원을 참조하십시오.

명령줄을 이용하여 시작 중에 보조 IPv4 주소를 할당하려면

- 다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.
 - `run-instances` 명령에서 `--secondary-private-ip-addresses` 옵션(AWS CLI)
 - `-NetworkInterface`를 정의하고 `New-EC2Instance` 명령(Windows PowerShell용 AWS 도구)과 함께 `PrivateIpAddresses` 파라미터를 지정합니다.

네트워크 인터페이스에 보조 프라이빗 IPv4 주소를 할당하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 네트워크 인터페이스를 선택한 후 인스턴스에 연결된 네트워크 인터페이스를 선택합니다.
- 작업, IP 주소 관리를 선택합니다.
- IPv4 주소에서 새 IP 할당을 선택합니다.
- 인스턴스의 서브넷 범위 내에 있는 특정 IPv4 주소를 입력합니다. 또는 필드를 공란으로 남기면 Amazon에서 IP 주소를 자동으로 선택합니다.
- (선택 사항) 재할당 허용을 선택하면 다른 네트워크 인터페이스가 이미 할당된 경우 보조 프라이빗 IP 주소가 재할당됩니다.
- 예, 업데이트를 선택합니다.

대안으로, 사용자는 인스턴스에 보조 프라이빗 IPv4 주소를 할당할 수 있습니다. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택한 후 작업, 네트워킹, IP 주소 관리를 차례로 선택합니다. 위의 단계와 마찬가지로 동일한 정보를 구성할 수 있습니다. IP 주소는 인스턴스에 대한 기본 네트워크 인터페이스(eth0)에 할당됩니다.

명령줄을 이용하여 기존 인스턴스에 보조 프라이빗 IPv4 주소를 할당하려면

- 다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.
 - `assign-private-ip-addresses`(AWS CLI)
 - `Register-EC2PrivateIpAddress`(Windows PowerShell용 AWS 도구)

보조 프라이빗 IPv4 주소를 인식하도록 인스턴스 운영 체제 구성

인스턴스에 보조 프라이빗 IPv4 주소를 할당한 이후에는 인스턴스에 운영 체제를 구성하여 보조 프라이빗 IP 주소가 인식되어야 합니다.

Windows 인스턴스에 대한 자세한 내용은 [Windows 인스턴스의 보조 프라이빗 IPv4 주소 구성 \(p. 564\)](#)을 참조합니다.

탄력적 IP 주소와 보조 프라이빗 IPv4 주소 연결

탄력적 IP 주소와 보조 프라이빗 IPv4 주소를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 탄력적 IP를 선택합니다.
3. 작업을 선택한 후 주소 연결을 선택합니다.
4. 네트워크 인터페이스에서 네트워크 인터페이스를 선택한 다음 프라이빗 IP 목록에서 보조 IP 주소를 선택합니다.
5. 연결을 선택합니다.

명령줄을 이용하여 탄력적 IP 주소와 보조 프라이빗 IPv4 주소를 연결하려면

- 다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.
 - `associate-address`(AWS CLI)
 - `Register-EC2Address`(Windows PowerShell용 AWS 도구)

보조 프라이빗 IPv4 주소 확인

네트워크 인터페이스에 할당된 프라이빗 IPv4 주소를 확인하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 확인할 프라이빗 IP 주소를 갖는 네트워크 인터페이스를 선택합니다.
4. 세부 정보 창의 세부 정보 탭에서 네트워크 인터페이스에 할당된 기본 프라이빗 IPv4 주소 및 모든 보조 프라이빗 IPv4 주소의 기본 프라이빗 IPv4 IP 및 보조 프라이빗 IPv4 IP 필드 확인란을 선택합니다.

인스턴스에 할당된 프라이빗 IPv4 주소를 확인하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 확인할 프라이빗 IPv4 주소를 갖는 인스턴스를 선택합니다.
4. 세부 정보 창의 설명 탭에서 네트워크 인터페이스를 통해 할당된 기본 프라이빗 IPv4 주소 및 모든 보조 프라이빗 IPv4 주소의 프라이빗 IP 및 보조 프라이빗 IP 필드 확인란을 선택합니다.

보조 프라이빗 IPv4 주소 할당 해제

보조 프라이빗 IPv4 주소가 더 이상 필요하지 않은 경우 인스턴스 또는 네트워크 인터페이스에서 해당 주소를 할당 해제할 수 있습니다. 보조 프라이빗 IPv4 주소가 네트워크 인터페이스에서 할당이 해제된 경우 탄력적 IP 주소(존재하는 경우)도 또한 연결이 해제됩니다.

인스턴스에서 보조 프라이빗 IPv4 주소의 할당을 해제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, 네트워킹, IP 주소 관리를 선택합니다.
4. IPv4 주소에서 할당을 해제할 IPv4 주소에 대해 할당 해제를 선택합니다.
5. 예, 업데이트를 선택합니다.

네트워크 인터페이스에서 보조 프라이빗 IPv4 주소의 할당을 해제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 작업, IP 주소 관리를 선택합니다.
4. IPv4 주소에서 할당을 해제할 IPv4 주소에 대해 할당 해제를 선택합니다.
5. 예, 업데이트를 선택합니다.

명령줄을 이용하여 보조 프라이빗 IPv4 주소의 할당을 해제하려면

- 다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.
 - [unassign-private-ip-addresses](#)(AWS CLI)
 - [Unregister-EC2PrivateIpAddress](#)(Windows PowerShell용 AWS 도구)

다중 IPv6 주소 관련 작업

다중 IPv6 주소를 인스턴스에 할당하고, 인스턴스에 할당된 IPv6 주소를 확인하며, 인스턴스에서 IPv6 주소 할당을 해제할 수 있습니다.

목차

- [다중 IPv6 주소 할당 \(p. 713\)](#)
- [IPv6 주소 확인 \(p. 715\)](#)
- [IPv6 주소 할당 해제 \(p. 715\)](#)

다중 IPv6 주소 할당

시작 중 또는 시작 후 인스턴스에 하나 이상의 IPv6 주소를 할당할 수 있습니다. 인스턴스에 IPv6 주소를 할당하려면 인스턴스를 시작하는 VPC와 서브넷에 연결된 IPv6 CIDR 블록이 있어야 합니다. 자세한 내용은 [Amazon VPC 사용 설명서의 VPC 및 서브넷](#)을 참조하십시오.

시작 중에 다중 IPv6 주소 할당

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 대시보드에서 인스턴스 시작을 선택합니다.
3. AMI를 선택하고 인스턴스 유형을 선택한 후 다음: 인스턴스 세부 정보 구성(Next: Configure Instance Details)을 선택합니다. IPv6를 지원하는 인스턴스 유형을 선택해야 합니다. 자세한 내용은 [인스턴스 유형 \(p. 114\)](#) 단원을 참조하십시오.
4. 인스턴스 세부 정보 구성 페이지의 네트워크 목록에서 VPC를 선택한 다음 서브넷 목록에서 서브넷을 선택합니다.
5. 네트워크 인터페이스 섹션에서 다음을 수행하고 다음: 스토리지 추가(Next: Add Storage)를 선택합니다.

- 기본 네트워크 인터페이스(eth0)에 단일 IPv6 주소를 할당하려면 IPv6 IP에서 IP 추가를 선택합니다. 보조 IPv6 주소를 추가하려면 IP 추가를 다시 선택합니다. 서브넷 범위에 속한 IPv6 주소를 직접 입력하거나, Amazon이 서브넷에 속한 IPv6 주소를 자동으로 선택하도록 기본 자동 할당 값을 그대로 둘 수 있습니다.
 - 다른 네트워크 인터페이스를 추가하려면 디바이스 추가를 선택하고, 하나 이상의 IPv6 주소를 네트워크 인터페이스에 추가하려면 단계를 반복합니다. 콘솔을 사용하여 인스턴스 시작 시 네트워크 인터페이스를 최대 두 개 지정할 수 있습니다. 인스턴스를 시작한 후 탐색 창에서 네트워크 인터페이스를 선택하여 네트워크 인터페이스를 추가합니다. 연결 가능한 총 네트워크 인터페이스의 수는 인스턴스 유형에 따라 다릅니다. 자세한 내용은 [인스턴스 유형별 네트워크 인터페이스당 IP 주소 \(p. 730\)](#) 단원을 참조하십시오.
- 마법사의 다음 단계를 수행하여 볼륨을 연결하고 인스턴스에 태그를 지정합니다.
 - 보안 그룹 구성 페이지에서 기존 보안 그룹을 선택하거나 새 보안 그룹을 생성합니다. IPv6를 통해 인스턴스에 연결할 수 있으려면 보안 그룹에 IPv6 주소로부터 액세스하도록 허용하는 규칙이 있어야 합니다. 자세한 내용은 [보안 그룹 규칙 참조 \(p. 919\)](#) 단원을 참조하십시오. 검토 및 시작을 선택합니다.
 - 인스턴스 시작 검토 페이지에서 설정을 검토한 후 시작을 선택하여 키 페어를 선택하고 인스턴스를 시작합니다. Amazon EC2를 처음 사용하며 아직 키 페어를 생성하지 않은 경우 키 페어를 생성하라는 메시지가 마법사에 표시됩니다.

인스턴스 화면 Amazon EC2 콘솔을 사용하여 기존 인스턴스에 다중 IPv6 주소를 할당할 수 있습니다. 그러면 인스턴스의 기본 네트워크 인터페이스(eth0)에 IPv6 주소가 할당됩니다. 인스턴스에 특정 IPv6 주소를 할당하려면 IPv6 주소에 이미 다른 인스턴스나 네트워크 인터페이스가 할당되어 있어서는 안 됩니다.

기존 인스턴스에 다중 IPv6 주소를 할당하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 인스턴스를 선택합니다.
- 인스턴스를 선택하고 작업, 네트워킹, IP 주소 관리를 선택합니다.
- IPv6 주소에서 추가할 IPv6 주소에 대해 새 IP 할당을 선택합니다. 서브넷 범위에 속한 IPv6 주소를 지정하거나, Amazon이 IPv6 주소를 자동으로 선택하도록 자동 할당 값을 그대로 둡니다.
- 예, 업데이트를 선택합니다.

또는 기존 네트워크 인터페이스에 다중 IPv6 주소를 할당할 수도 있습니다. 네트워크 인터페이스는 연결된 IPv6 CIDR 블록이 있는 서브넷에서 생성되어야 합니다. 네트워크 인터페이스에 특정 IPv6 주소를 할당하려면 IPv6 주소에 이미 다른 네트워크 인터페이스가 할당되어 있어서는 안 됩니다.

네트워크 인터페이스에 다중 IPv6 주소를 할당하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 네트워크 인터페이스를 선택합니다.
- 네트워크 인터페이스를 선택하고 작업, IP 주소 관리를 선택합니다.
- IPv6 주소에서 추가할 IPv6 주소에 대해 새 IP 할당을 선택합니다. 서브넷 범위에 속한 IPv6 주소를 지정하거나, Amazon이 IPv6 주소를 자동으로 선택하도록 자동 할당 값을 그대로 둡니다.
- 예, 업데이트를 선택합니다.

CLI 개요

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- 시작 중에 IPv6 주소 할당:
 - `run-instances` 명령과 함께 `--ipv6-addresses` 또는 `--ipv6-address-count` 옵션을 사용합니다. (AWS CLI)

- `-NetworkInterface`를 정의하고 [New-EC2Instance](#) 명령과 함께 `Ipv6Addresses` 또는 `Ipv6AddressCount` 파라미터를 지정합니다. (Windows PowerShell용 AWS 도구).
- 네트워크 인터페이스에 IPv6 주소 할당:
 - [assign-ipv6-addresses](#)(AWS CLI)
 - [Register-EC2Ipv6AddressList](#)(Windows PowerShell용 AWS 도구)

IPv6 주소 확인

인스턴스 또는 네트워크 인터페이스에 대한 IPv6 주소를 확인할 수 있습니다.

인스턴스에 할당된 IPv6 주소를 확인하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택합니다. 세부 정보 창에서 IPv6 IP 필드를 검토합니다.

네트워크 인터페이스에 할당된 IPv6 주소를 확인하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택합니다. 세부 정보 창에서 IPv6 IP 필드를 검토합니다.

CLI 개요

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- 인스턴스에 대한 IPv6 주소 확인:
 - [describe-instances](#) (AWS CLI)
 - [Get-EC2Instance](#)(Windows PowerShell용 AWS 도구).
- 네트워크 인터페이스에 대한 IPv6 주소 확인:
 - [describe-network-interfaces](#)(AWS CLI)
 - [Get-EC2NetworkInterface](#)(Windows PowerShell용 AWS 도구)

IPv6 주소 할당 해제

인스턴스의 기본 네트워크 인터페이스에서 IPv6 주소 할당을 해제하거나 네트워크 인터페이스에서 IPv6 주소 할당을 해제할 수 있습니다.

인스턴스에서 IPv6 주소 할당 해제

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, 네트워킹, IP 주소 관리를 선택합니다.
4. IPv6 주소에서 할당을 해제할 IPv6 주소에 대해 할당 해제를 선택합니다.
5. 예, 업데이트를 선택합니다.

네트워크 인터페이스에 할당된 IPv6 주소를 해제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 작업, IP 주소 관리를 선택합니다.
4. IPv6 주소에서 할당을 해제할 IPv6 주소에 대해 할당 해제를 선택합니다.
5. 저장을 선택합니다.

CLI 개요

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [unassign-ipv6-addresses\(AWS CLI\)](#)
- [Unregister-EC2Ipv6AddressList\(Windows PowerShell용 AWS 도구\).](#)

Amazon EC2의 고유 IP 주소 가져오기(BYOIP)

온프레미스 네트워크에서 AWS 계정으로 모든 퍼블릭 IPv4 주소 범위 또는 IPv6 주소 범위의 부분 또는 전체를 가져올 수 있습니다. 주소 범위를 계속해서 소유할 수 있지만 AWS에서는 기본적으로 인터넷에 이러한 주소 범위를 알립니다. 주소 범위를 AWS로 가져오고 나면 이러한 주소가 계정에 주소 풀로 나타납니다.

BYOIP는 일부 리전에서는 사용할 수 없습니다. 지원되는 리전 목록은 [고유 IP 주소 가져오기에 대한 FAQ](#)를 참조하십시오.

Note

다음 단계에서는 Amazon EC2에서만 사용할 IP 주소 범위를 가져오는 방법에 대해 설명합니다. AWS Global Accelerator에서 사용할 고유 IP 주소 범위를 가져오는 단계는 AWS Global Accelerator 개발자 안내서의 [고유 IP 주소 가져오기\(BYOIP\)](#)를 참조하십시오.

주제

- [요구 사항 \(p. 716\)](#)
- [AWS 계정으로 주소 범위를 가져오도록 준비하기 \(p. 717\)](#)
- [AWS에서 사용할 수 있도록 주소 범위 프로비저닝 \(p. 718\)](#)
- [AWS를 통해 주소 범위 알리기 \(p. 719\)](#)
- [주소 범위 관련 작업 \(p. 720\)](#)
- [주소 범위 프로비저닝 취소 \(p. 721\)](#)

요구 사항

- 주소 범위를 ARIN(미국 인터넷 번호 등록 협회), RIPE(Réseaux IP Européens Network Coordination Centre) 또는 APNIC(아시아 태평양 지역 네트워크 정보 센터)와 같은 RIR(지역 인터넷 등록 기관)에 등록해야 합니다. 주소 범위를 기업 또는 기관에 등록해야 하며 개인에게는 등록할 수 없습니다.
- 가져올 수 있는 가장 구체적인 IPv4 주소 범위는 /24입니다.
- 가져올 수 있는 가장 구체적인 IPv6 주소 범위는 공개적으로 알려지는 CIDR의 경우 /48, [공개적으로 알려지지 않는 \(p. 719\)](#) CIDR의 경우 /56입니다.
- 각 주소 범위를 한 번에 하나의 리전으로 가져올 수 있습니다.
- 리전당 총 5개의 IPv4 및 IPv6 주소 범위를 AWS 계정으로 가져올 수 있습니다.
- IP 주소 범위의 주소에는 명확한 기록이 있어야 합니다. 당사는 IP 주소 범위의 평판을 조사할 수 있으며, 좋지 않은 평판이 있거나 악의적인 동작과 연관된 IP 주소가 포함될 경우 IP 주소 범위를 거부할 권리보유합니다.
- 사용하는 IP 주소를 소유해야 합니다. 즉, 다음 항목만 지원됩니다.

- ARIN - "직접 할당" 및 "직접 지정" 네트워크 유형
- RIPE - "ALLOCATED PA", "LEGACY", "ASSIGNED PI", "ALLOCATED-BY-RIR" 할당 상태
- APNIC – "ALLOCATED PORTABLE" 및 "ASSIGNED PORTABLE" 할당 상태

AWS 계정으로 주소 범위를 가져오도록 준비하기

자신만 주소 범위를 AWS 계정으로 가져올 수 있도록 하려면 주소 범위를 알릴 수 있는 권한을 Amazon에 부여해야 합니다. 또한 서명된 권한 부여 메시지를 통해 주소 범위를 소유하고 있다는 증명을 제공해야 합니다.

ROA(라우팅 오리진 권한 부여)는 RIR를 통해 생성할 수 있는 라우팅 공지에 대한 암호화 설명입니다. 이 문서에는 주소 범위, 주소 범위를 알리도록 허용된 자율 시스템 번호(ASN)와 만료 날짜가 포함되어 있습니다. ROA는 Amazon에 특정 AS 번호로 주소 범위를 알릴 수 있는 권한을 부여합니다. 그러나 사용자의 AWS 계정에는 AWS에 주소 범위를 가져올 수 있는 권한을 부여하지 않습니다. AWS 계정에 AWS로 주소 범위를 가져올 수 있는 권한을 부여하려면 주소 범위에 대한 등록 데이터 액세스 프로토콜(RDAP) 설명에 자체 서명된 X509 인증서를 게시해야 합니다. 이 인증서에는 제공한 권한 부여-컨텍스트 서명을 확인하기 위해 AWS가 사용하는 퍼블릭 키가 포함되어 있습니다. 프라이빗 키는 안전하게 보관하고, 권한 부여-컨텍스트 메시지에 서명하는 데 사용하십시오.

이 작업의 명령은 Linux에서 지원됩니다. Windows에서는 [Windows Subsystem for Linux](#)를 사용하여 Linux 명령을 실행할 수 있습니다.

작업

- ROA 객체 생성 (p. 717)
- 자체 서명된 X509 인증서 생성 (p. 717)
- 서명된 권한 부여 메시지 생성 (p. 718)

ROA 객체 생성

ROA 객체를 생성하여 Amazon ASN 16509 및 14618이 주소 범위를 공급하도록 권한을 부여하고 현재 주소 범위의 공급 권한이 있는 ASN이 주소 범위를 공급하도록 권한을 부여합니다. 가져오려는 가장 작은 접두사 크기에 대한 최대 길이를 설정해야 합니다(예: /24). Amazon이 ROA를 사용할 수 있을 때까지 최대 24시간이 걸릴 수 있습니다. 자세한 내용은 다음 단원을 참조하십시오.

- ARIN - [ROA 요청](#)
- RIPE - [ROA 관리](#)
- APNIC — 라우팅 관리

자체 서명된 X509 인증서 생성

다음 절차에 따라 자체 서명된 X509 인증서를 생성하여 RIR의 RDAP 레코드에 추가하십시오. `openssl` 명령에는 OpenSSL 버전 1.0.2 이상이 필요합니다.

아래 명령을 복사하고 자리 표시자 값(색상이 있는 기울임꼴 텍스트)만 바꿉니다.

자체 서명된 X509 인증서를 생성하여 RIR 레코드에 추가하려면

1. 다음과 같이 RSA 2048비트 키 페어를 생성합니다.

```
openssl genrsa -out private.key 2048
```

2. 다음 명령을 사용하여 키 페어에서 퍼블릭 X509 인증서를 생성합니다. 이 예제에서 인증서는 365일이 지나면 만료되므로, 이 기간이 지난 후에는 신뢰할 수 없습니다. 따라서 만료 날짜를 적절하게 설정해야 합니다. 정보를 입력하라는 메시지가 표시되면 기본값을 수락할 수 있습니다.

```
openssl req -new -x509 -key private.key -days 365 | tr -d "\n" > publickey.cer
```

- X509 인증서로 RIR에 대한 RDAP 레코드를 업데이트합니다. 인증서에서 -----BEGIN CERTIFICATE----- 및 -----END CERTIFICATE-----를 복사해야 합니다. 이전 단계에서 tr -d "\n" 명령을 사용해 아직 제거하지 않은 경우에는 줄바꿈 문자를 제거해야 합니다. 인증서를 보려면 다음 명령을 실행합니다.

```
cat publickey.cer
```

ARIN의 경우 주소 범위의 "공개 주석" 섹션에 인증서를 추가합니다. 조직의 주석 섹션에 추가하지 마십시오.

RIPE의 경우 주소 범위의 새 "설명" 필드에 인증서를 추가합니다. 조직의 주석 섹션에 추가하지 마십시오.

APNIC의 경우 이메일을 통해 퍼블릭 키를 helpdesk@apnic.net로 전송하여 주소 범위의 "설명" 필드에 수동으로 추가합니다. IP 주소의 APNIC 공인 연락처를 사용하여 이메일을 전송합니다.

서명된 권한 부여 메시지 생성

서명된 권한 부여 메시지의 형식은 다음과 같으며, 여기서 날짜는 메시지의 만료 날짜입니다.

```
1|aws|account|cidr|YYYYMMDD|SHA256|RSAPSS
```

서명된 권한 부여 메시지를 만들려면

- 일반 텍스트 권한 부여 메시지를 만들고 다음 예제와 같이 `text_message`라는 변수에 저장합니다. 다음 예제를 복사하고 예제 계정 번호, 주소 범위 및 만료 날짜만 해당 값으로 바꿉니다.

```
text_message="1|aws|123456789012|198.51.100.0/24|20191201|SHA256|RSAPSS"
```

- 생성한 키 페어를 사용하여 `text_message`의 권한 부여 메시지에 서명하고 `signed_message`라는 변수에 저장합니다.

```
signed_message=$(echo $text_message | tr -d "\n" | openssl dgst -sha256 -sigopt rsa_padding_mode:pss -sigopt rsa_pss_saltlen:-1 -sign private.key -keyform PEM | openssl base64 | tr -- '+=' '-'_-' | tr -d "\n")
```

Important

이 명령을 복사하여 붙여넣는 것이 좋습니다. 값을 수정하거나 바꾸지 마십시오.

AWS에서 사용할 수 있도록 주소 범위 프로비저닝

AWS에서 사용할 수 있도록 주소 범위를 프로비저닝하는 경우 주소 범위를 소유하고 있는지 확인하고 Amazon에 해당 주소 범위를 알릴 수 있는 권한을 부여하고 있는 것입니다. 또한 서명된 권한 부여 메시지를 통해 주소 범위를 소유하고 있음을 확인합니다. 이 메시지는 X509 인증서로 RDAP 레코드를 업데이트할 때 사용한 자체 서명된 X509 키 페어로 서명됩니다.

주소 범위를 프로비저닝하려면 다음 `provision-byoip-cidr` 명령을 사용합니다. 예제 주소 범위를 해당 주소 범위로 바꿉니다. `--cidr-authorization-context` 옵션은 ROA 메시지가 아니라 이전에 생성한 변수를 사용합니다.

```
aws ec2 provision-byoip-cidr --cidr address-range --cidr-authorization-context  
Message="$text_message",Signature="$signed_message"
```

주소 범위 프로비저닝은 비동기 작업이므로, 호출이 즉시 반환되지만 주소 범위는 상태가 `pending-provision`에서 `provisioned`로 변경되어야 사용할 준비가 된 것입니다. 프로비저닝 프로세스를 완료 하려면 최대 5주까지 걸릴 수 있습니다. 프로비저닝한 주소 범위의 상태를 모니터링하려면 다음 `describe-byoip-cidrs` 명령을 사용합니다.

```
aws ec2 describe-byoip-cidrs --max-results 5
```

프로비저닝 중에 문제가 발생하고 상태가 `failed-provision`으로 전환되면 문제가 해결된 후 `provision-byoip-cidr` 명령을 다시 실행해야 합니다.

공개적으로 알려지지 않는 IPv6 주소 범위 프로비저닝

기본적으로 주소 범위는 인터넷에 공개적으로 알려지도록 프로비저닝됩니다. 공개적으로 알려지지 않는 IPv6 주소 범위를 프로비저닝할 수 있습니다. 비공개 주소 범위의 IPv6 CIDR 블록을 VPC와 연결하면 IPv6 CIDR은 AWS Direct Connect 연결을 통해서만 액세스할 수 있습니다.

ROA는 비공개 주소 범위를 프로비저닝하는 데 필요하지 않습니다.

공개적으로 알려지지 않는 IPv6 주소 범위를 프로비저닝하려면 다음 `provision-byoip-cidr` 명령을 사용합니다.

```
aws ec2 provision-byoip-cidr --cidr address-range --cidr-authorization-context  
Message="$text_message",Signature="$signed_message" --no-publicly-advertisable
```

Important

프로비저닝 중에만 `publicly-advertisable` 또는 `no-publicly-advertisable` 플래그를 설정할 수 있습니다. 주소 범위의 알리기 상태를 나중에 변경할 수는 없습니다.

AWS를 통해 주소 범위 알리기

주소 범위가 프로비저닝되면 알릴 준비가 된 것입니다. 프로비저닝한 정확한 주소 범위를 알려야 합니다. 프로비저닝한 주소 범위의 일부만 알릴 수 없습니다.

공개적으로 알려지지 않는 IPv6 주소 범위를 프로비저닝한 경우 이 단계를 완료할 필요가 없습니다.

AWS를 통해 알리기 전에 다른 위치에서 주소 범위 알리기를 중지하는 것이 좋습니다. 다른 위치에서 IP 주소 범위 알리기를 계속하면 당사가 해당 주소 범위를 안정적으로 지원하지 못하거나 문제를 제대로 해결하지 못할 수 있습니다. 특히, 해당 주소 범위로의 트래픽이 당사 네트워크로 들어오는지 보장할 수 없습니다.

가동 중지 시간을 최소화하기 위해 주소를 알리기 전에 주소 풀의 주소를 사용하도록 AWS 리소스를 구성하고, 동시에 현재 위치에서 주소 알리기를 중지한 다음 AWS를 통해 주소 알리기를 시작할 수 있습니다. 주소 풀의 탄력적 IP 주소 할당에 대한 자세한 내용은 [탄력적 IP 주소 할당 \(p. 722\)](#) 단원을 참조하십시오.

주소 범위를 알리려면 다음 `provision-byoip-cidr` 명령을 사용합니다.

```
aws ec2 advertise-byoip-cidr --cidr address-range
```

Important

매번 다른 주소 범위를 지정하더라도 최소 10초마다 `advertise-byoip-cidr` 명령을 실행할 수 있습니다.

주소 범위 알리기를 중지하려면 다음 [withdraw-byoip-cidr](#) 명령을 사용합니다.

```
aws ec2 withdraw-byoip-cidr --cidr address-range
```

Important

매번 다른 주소 범위를 지정하더라도 최소 10초마다 withdraw-byoip-cidr 명령을 실행할 수 있습니다.

주소 범위 관련 작업

계정에서 프로비저닝한 IPv4 및 IPv6 주소 범위를 보고 작업할 수 있습니다.

IPv4 주소 범위

IPv4 주소 풀에서 탄력적 IP 주소를 생성하여 AWS 리소스(예: EC2 인스턴스, NAT 게이트웨이, Network Load Balancer)와 함께 사용할 수 있습니다.

계정에서 프로비저닝한 IPv4 주소 풀에 대한 정보를 보려면 다음 [describe-public-ipv4-pools](#) 명령을 사용합니다.

```
aws ec2 describe-public-ipv4-pools
```

IPv4 주소 풀에서 탄력적 IP 주소를 생성하려면 [assign-address](#) 명령을 사용합니다. [--public-ipv4-pool](#) 옵션을 사용하여 [describe-byoip-cidrs](#)에서 반환된 주소 풀의 ID를 지정할 수 있습니다. 또는 [--address](#) 옵션을 사용하여 프로비저닝한 주소 범위에서 주소를 지정할 수 있습니다.

IPv6 주소 범위

계정에서 프로비저닝한 IPv6 주소 풀에 대한 정보를 보려면 다음 [describe-ipv6-pools](#) 명령을 사용합니다.

```
aws ec2 describe-ipv6-pools
```

VPC를 생성하고 IPv6 주소 풀에서 IPv6 CIDR를 지정하려면 다음 [create-vpc](#) 명령을 사용합니다. Amazon이 IPv6 주소 풀에서 IPv6 CIDR을 선택하도록 하려면 [--ipv6-cidr-block](#) 옵션을 생략합니다.

```
aws ec2 create-vpc --cidr-block 10.0.0.0/16 --ipv6-cidr-block ipv6-cidr --ipv6-pool pool-id
```

IPv6 주소 풀의 IPv6 CIDR 블록을 VPC와 연결하려면 다음 [associate-vpc-cidr-block](#) 명령을 사용합니다. Amazon이 IPv6 주소 풀에서 IPv6 CIDR을 선택하도록 하려면 [--ipv6-cidr-block](#) 옵션을 생략합니다.

```
aws ec2 associate-vpc-cidr-block --vpc-id vpc-123456789abc123ab --ipv6-cidr-block ipv6-cidr --ipv6-pool pool-id
```

VPC 및 연결된 IPv6 주소 풀에 대한 정보를 보려면 [describe-vpcs](#) 명령을 사용합니다. 특정 IPv6 주소 풀에서 연결된 IPv6 CIDR 블록에 대한 정보를 보려면 다음 [get-associated-ipv6-pool-cidrs](#) 명령을 사용합니다.

```
aws ec2 get-associated-ipv6-pool-cidrs --pool-id pool-id
```

VPC에서 IPv6 CIDR 블록의 연결을 해제하면 해당 블록이 IPv6 주소 풀로 다시 해제됩니다.

VPC 콘솔에서 IPv6 CIDR 블록 관련 작업을 수행하는 방법에 대한 자세한 내용은 Amazon VPC 사용 설명서의 [VPC 및 서브넷 관련 작업](#)을 참조하십시오.

주소 범위 프로비저닝 취소

AWS에서 주소 범위 사용을 중지하려면 먼저 탄력적 IP 주소를 해제하고 여전히 주소 풀에서 할당된 IPv6 CIDR 블록의 연결을 해제합니다. 그런 다음 주소 범위 알리기를 중지하고 마지막으로 주소 범위 프로비저닝을 취소하십시오.

주소 범위의 일부에 대해 프로비저닝을 취소할 수는 없습니다. AWS에서 보다 구체적인 주소 범위를 사용하려면 전체 주소 범위의 프로비저닝을 취소하고 보다 구체적인 주소 범위를 프로비저닝하십시오.

(IPv4) 각 탄력적 IP 주소를 해제하려면 다음 [release-address](#) 명령을 사용합니다.

```
aws ec2 release-address --allocation-id eipalloc-12345678abcbabc
```

(IPv6) IPv6 CIDR 블록의 연결을 해제하려면 다음 [disassociate-vpc-cidr-block](#) 명령을 사용합니다.

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-12345abcd1234abc1
```

주소 범위 알리기를 중지하려면 다음 [withdraw-byoip-cidr](#) 명령을 사용합니다.

```
aws ec2 withdraw-byoip-cidr --cidr address-range
```

주소 범위 프로비저닝을 취소하려면 다음 [deprovision-byoip-cidr](#) 명령을 사용합니다.

```
aws ec2 deprovision-byoip-cidr --cidr address-range
```

주소 범위의 프로비저닝을 해제하는 데 최대 하루가 걸릴 수 있습니다.

탄력적인 IP 주소

탄력적 IP 주소는 동적 클라우드 컴퓨팅을 위해 고안된 정적 IPv4 주소입니다. 탄력적 IP 주소는 AWS 계정과 연결됩니다. 탄력적 IP 주소를 사용하면 주소를 계정의 다른 인스턴스에 신속하게 다시 매핑하여 인스턴스나 소프트웨어의 오류를 마스킹할 수 있습니다.

탄력적 IP 주소는 인터넷에서 연결 가능한 퍼블릭 IPv4 주소입니다. 인스턴스에 퍼블릭 IPv4 주소가 없는 경우 탄력적 IP 주소를 인스턴스에 연결하여 인터넷 통신을 활성화할 수 있습니다. 예를 들어 로컬 컴퓨터에서 인스턴스에 연결할 수 있습니다.

현재는 IPv6에 대한 탄력적 IP 주소를 지원하지 않습니다.

목차

- [탄력적 IP 주소 기본 사항 \(p. 721\)](#)
- [탄력적 IP 주소 작업 \(p. 722\)](#)
- [이메일 애플리케이션에 역방향 DNS 사용 \(p. 728\)](#)
- [탄력적 IP 주소 제한 \(p. 728\)](#)

탄력적 IP 주소 기본 사항

탄력적 IP 주소의 기본 특성은 다음과 같습니다.

- 탄력적 IP 주소를 사용하려면 먼저 계정에 주소를 할당한 후 인스턴스 또는 네트워크 인터페이스와 연결합니다.
- 탄력적 IP 주소를 인스턴스와 연결하면 해당 인스턴스의 기본 네트워크 인터페이스와도 연결됩니다. 탄력적 IP 주소를 인스턴스에 연결된 네트워크 인터페이스와 연결하면 해당 인스턴스와도 연결됩니다.

- 탄력적 IP 주소를 인스턴스 또는 기본 네트워크 인터페이스와 연결하면, 인스턴스의 퍼블릭 IPv4 주소(인 경우)는 Amazon의 퍼블릭 IPv4 주소 폴로 다시 릴리스됩니다. 퍼블릭 IPv4 주소를 재사용할 수 없으며, 퍼블릭 IPv4 주소를 탄력적 IP 주소로 변환할 수 없습니다. 자세한 내용은 [퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름 \(p. 703\)](#) 단원을 참조하십시오.
- 탄력적 IP 주소는 리소스에서 연결 해제했다가 다른 리소스와 다시 연결할 수 있습니다. 예기치 않은 동작을 방지하려면 변경하기 전에 기존 연결에 이름이 지정된 리소스에 대한 모든 활성 연결이 닫혀 있는지 확인합니다. 탄력적 IP 주소를 다른 리소스에 연결한 후 새로 연결된 리소스에 대한 연결을 다시 열 수 있습니다.
- 연결 해제한 탄력적 IP 주소는 명시적으로 릴리스할 때까지 계정에 할당되어 있습니다.
- 탄력적 IP 주소의 효율적인 사용을 위해 탄력적 IP 주소가 실행 중인 인스턴스와 연결되어 있지 않거나 종지된 인스턴스 또는 연결되지 않은 네트워크 인터페이스와 연결된 경우 소액의 시간당 요금이 부과됩니다. 인스턴스가 실행 중인 동안에는 이와 연결된 탄력적 IP 주소 하나에 대해서는 요금이 부과되지 않지만 해당 인스턴스와 연결된 추가 탄력적 IP 주소에 대해서는 요금이 부과됩니다. 자세한 내용은 [Amazon EC2 요금, 온디맨드 요금 페이지](#)에서 탄력적 IP 주소 섹션을 참조하세요.
- 탄력적 IP 주소는 특정 네트워크 경계 그룹에서만 사용할 수 있습니다.
- 탄력적 IP 주소를 이전에 퍼블릭 IPv4 주소가 있던 인스턴스와 연결하면 인스턴스의 퍼블릭 DNS 호스트 이름이 탄력적 IP 주소에 맞게 변경됩니다.
- Amazon은 퍼블릭 DNS 호스트 이름을 인스턴스 네트워크 외부에서는 인스턴스의 퍼블릭 IPv4 주소 또는 탄력적 IP 주소로 변환하고, 인스턴스 네트워크 내부에서는 인스턴스의 프라이빗 IPv4 주소로 변환합니다.
- AWS 계정으로 가져온 IP 주소 폴에서 탄력적 IP 주소를 할당하는 경우 해당 IP 주소는 탄력적 IP 주소 한도에 포함되지 않습니다. 자세한 내용은 [탄력적 IP 주소 제한 \(p. 728\)](#) 단원을 참조하십시오.
- 탄력적 IP 주소를 할당할 때 탄력적 IP 주소를 네트워크 경계 그룹에 연결할 수 있습니다. Amazon은 여기서 CIDR 블록을 알립니다. 네트워크 경계 그룹을 설정하면 CIDR 블록이 이 그룹으로 제한됩니다. 네트워크 경계 그룹을 지정하지 않으면 리전(예: us-west-2)의 모든 가용 영역을 포함하는 경계 그룹이 설정됩니다.

탄력적 IP 주소 작업

다음 섹션에서는 탄력적 IP 주소를 이용한 작업 방법에 대해 살펴보겠습니다.

작업

- [탄력적 IP 주소 할당 \(p. 722\)](#)
- [탄력적 IP 주소 설명 \(p. 723\)](#)
- [탄력적 IP 주소 태그 지정 \(p. 724\)](#)
- [실행 중인 인스턴스 또는 네트워크 인터페이스와 탄력적 IP 주소 연결 \(p. 725\)](#)
- [탄력적 IP 주소 연결 해제 \(p. 726\)](#)
- [탄력적 IP 주소 해제 \(p. 727\)](#)
- [탄력적 IP 주소 복구 \(p. 727\)](#)

탄력적 IP 주소 할당

퍼블릭 IPv4 주소의 Amazon 폴 또는 AWS 계정으로 가져온 사용자 지정 IP 주소 폴에서 탄력적 IP 주소를 할당할 수 있습니다. AWS 계정으로 고유한 IP 주소 범위 가져오기에 대한 자세한 내용은 [Amazon EC2의 고유 IP 주소 가져오기\(BYOIP\) \(p. 716\)](#) 단원을 참조하십시오.

다음 방법 중 하나를 사용하여 탄력적 IP 주소를 할당할 수 있습니다.

새로운 콘솔

탄력적 IP 주소를 할당하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. Allocate Elastic IP address(탄력적 IP 주소 할당)를 선택합니다.
4. 사용할 범위에 따라 범위에서 VPC 또는 EC2-Classic을 선택합니다.
5. (VPC 범위에만 해당)Public IPv4 address pool(퍼블릭 IPv4 주소 풀)의 경우 다음 중 하나를 선택합니다.
 - Amazon의 IP 주소 풀 — IPv4 주소를 Amazon의 IP 주소 풀에서 할당하려는 경우.
 - My pool of public IPv4 addresses(내 퍼블릭 IPv4 주소 풀) — AWS 계정으로 가져온 IP 주소 풀에서 IPv4 주소를 할당하려는 경우. IP 주소 풀이 없는 경우에는 이 옵션을 사용할 수 없습니다.
 - Customer owned pool of IPv4 addresses(고객 소유 IPv4 주소 풀) — AWS Outpost에서 사용하기 위해 온프레미스 네트워크에서 만든 풀에서 IPv4 주소를 할당하려는 경우. AWS Outpost가 없는 경우 이 옵션이 비활성화됩니다.
6. [Allocate]를 선택합니다.

기존 콘솔

탄력적 IP 주소를 할당하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. Allocate new address를 선택합니다.
4. IPv4 주소 풀로 Amazon 풀을 선택합니다.
5. 할당을 선택하고 확인 화면을 닫습니다.

AWS CLI

탄력적 IP 주소를 할당하려면

`allocate-address` AWS CLI 명령을 사용합니다.

PowerShell

탄력적 IP 주소를 할당하려면

`New-EC2Address` Windows PowerShell용 AWS 도구 명령을 사용합니다.

탄력적 IP 주소 설명

다음 방법 중 하나를 사용하여 탄력적 IP 주소를 설명할 수 있습니다.

새로운 콘솔

탄력적 IP 주소를 설명하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 탄력적 IP를 선택합니다.
3. 보려는 탄력적 IP 주소를 선택하고 작업, 세부 정보 보기를 선택합니다.

기존 콘솔

탄력적 IP 주소를 설명하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 탄력적 IP를 선택합니다.

- 리소스 속성 목록에서 필터를 선택하여 검색을 시작합니다. 단일 검색에 여러 필터를 사용할 수 있습니다.

AWS CLI

탄력적 IP 주소를 설명하려면

[describe-addresses](#) AWS CLI 명령을 사용합니다.

PowerShell

탄력적 IP 주소를 설명하려면

[Get-EC2Address](#) Windows PowerShell용 AWS 도구 명령을 사용합니다.

탄력적 IP 주소 태그 지정

탄력적 IP 주소에 사용자 지정 태그를 할당하여 용도, 소유자, 환경 등 다양한 방식으로 주소를 분류할 수 있습니다. 그러면 할당한 사용자 지정 태그를 기반으로 특정 탄력적 IP 주소를 빠르게 찾을 수 있습니다.

VPC 범위에 있는 탄력적 IP 주소에만 태그를 지정할 수 있습니다.

Note

탄력적 IP 주소 태그를 사용한 비용 할당 추적은 지원되지 않습니다.

다음 방법 중 하나를 사용하여 탄력적 IP 주소에 태그를 지정할 수 있습니다.

새로운 콘솔

탄력적 IP 주소를 태그하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 [Elastic IPs]를 선택합니다.
- 태그를 지정할 탄력적 IP 주소를 선택하고 작업, 세부 정보 보기를 선택합니다.
- 태그 섹션에서 태그 관리를 선택합니다.
- 태그 키 및 값 페어를 지정합니다.
- (선택 사항) 태그 추가를 선택하여 태그를 추가합니다.
- 저장을 선택합니다.

기존 콘솔

탄력적 IP 주소를 태그하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 [Elastic IPs]를 선택합니다.
- 태그를 지정할 탄력적 IP 주소를 선택하고 태그를 선택합니다.
- [Add/Edit Tags]를 선택합니다.
- 태그 추가/편집 대화 상자에서 태그 생성을 선택하고 태그에 대한 키와 값을 지정합니다.
- (선택 사항) 태그 생성을 선택하여 탄력적 IP 주소에 추가 태그를 추가합니다.
- 저장을 선택합니다.

AWS CLI

탄력적 IP 주소를 태그하려면

create-tags AWS CLI 명령을 사용합니다.

```
aws ec2 create-tags --resources eipalloc-12345678 --tags Key=Owner,Value=TeamA
```

PowerShell

탄력적 IP 주소를 태그하려면

New-EC2Tag Windows PowerShell용 AWS 도구 명령을 사용합니다.

New-EC2Tag 명령에는 탄력적 IP 주소 태그에 사용할 키-값 페어를 지정하는 Tag 파라미터가 필요합니다. 다음 명령은 Tag 파라미터를 생성합니다.

```
PS C:\> $tag = New-Object Amazon.EC2.Model.Tag
PS C:\> $tag.Key = "Owner"
PS C:\> $tag.Value = "TeamA"
```

```
PS C:\> New-EC2Tag -Resource eipalloc-12345678 -Tag $tag
```

실행 중인 인스턴스 또는 네트워크 인터페이스와 탄력적 IP 주소 연결

인스턴스와 탄력적 IP 주소를 연결하여 인터넷과 통신을 활성화하는 경우 인스턴스가 퍼블릭 서브넷에 위치해야 합니다. 자세한 내용은 [Amazon VPC 사용 설명서](#)의 인터넷 게이트웨이 단원을 참조하십시오.

다음 방법 중 하나를 사용하여 탄력적 IP 주소를 인스턴스 또는 네트워크 인터페이스에 연결할 수 있습니다.

새로운 콘솔

인스턴스와 엘라스틱 IP 주소를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 템색 창에서 탄력적 IP를 선택합니다.
3. 연결할 탄력적 IP 주소를 선택하고 작업, 탄력적 IP 주소 연결을 선택합니다.
4. 리소스 유형에서 인스턴스를 선택합니다.
5. 예를 들어 탄력적 IP 주소를 연결할 인스턴스를 선택합니다. 텍스트를 입력하여 특정 인스턴스를 검색할 수도 있습니다.
6. (선택 사항) 프라이빗 IP 주소에 탄력적 IP 주소를 연결할 프라이빗 IP 주소를 지정합니다.
7. 연결을 선택합니다.

네트워크 인터페이스와 탄력적 IP 주소를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 템색 창에서 탄력적 IP를 선택합니다.
3. 연결할 탄력적 IP 주소를 선택하고 작업, 탄력적 IP 주소 연결을 선택합니다.
4. [Resource type(리소스 유형)]에서 Network interface(네트워크 인터페이스)을 선택합니다.
5. 네트워크 인터페이스에서 탄력적 IP 주소를 연결할 네트워크 인터페이스를 선택합니다. 텍스트를 입력하여 특정 네트워크 인터페이스를 검색할 수도 있습니다.
6. (선택 사항) 프라이빗 IP 주소에 탄력적 IP 주소를 연결할 프라이빗 IP 주소를 지정합니다.
7. 연결을 선택합니다.

기존 콘솔

인스턴스와 엘라스틱 IP 주소를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 탄력적 IP를 선택합니다.
3. 탄력적 IP 주소를 선택하고 작업, 주소 연결을 선택합니다.
4. 인스턴스에서 인스턴스를 선택한 후 연결을 선택합니다.

AWS CLI

탄력적 IP 주소를 연결하려면

[associate-address](#) AWS CLI 명령을 사용합니다.

PowerShell

탄력적 IP 주소를 연결하려면

[Register-EC2Address](#) Windows PowerShell용 AWS 도구 명령을 사용합니다.

탄력적 IP 주소 연결 해제

언제든지 인스턴스 또는 네트워크 인터페이스에서 탄력적 IP 주소의 연결을 해제할 수 있습니다. 탄력적 IP 주소의 연결을 해제한 후 다른 리소스와 다시 연결할 수 있습니다.

다음 방법 중 하나를 사용하여 탄력적 IP 주소의 연결을 해제할 수 있습니다.

새로운 콘솔

탄력적 IP 주소의 연결을 해제하고 다시 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 탄력적 IP를 선택합니다.
3. 연결을 해제할 탄력적 IP 주소를 선택하고 작업, 탄력적 IP 주소 연결 해제를 선택합니다.
4. 연결 해제를 선택합니다.

기존 콘솔

탄력적 IP 주소의 연결을 해제하고 다시 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 탄력적 IP를 선택합니다.
3. 탄력적 IP 주소를 선택하고 작업을 선택한 후 주소 연결 해제를 선택합니다.
4. 주소 연결 해제를 선택합니다.

AWS CLI

엘라스틱 IP 주소를 연결 해제하려면

[disassociate-address](#) AWS CLI 명령을 사용합니다.

PowerShell

엘라스틱 IP 주소를 연결 해제하려면

[Unregister-EC2Address](#) Windows PowerShell용 AWS 도구 명령을 사용합니다.

탄력적 IP 주소 해제

탄력적 IP 주소가 더 이상 필요하지 않으면 다음 방법 중 하나를 사용하여 해제하는 것이 좋습니다. 해제할 주소는 현재 EC2 인스턴스, NAT 게이트웨이 또는 Network Load Balancer와 같은 AWS 리소스와 연결되어 있지 않아야 합니다.

새로운 콘솔

엘라스틱 IP 주소를 해제합니다

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. 해제할 탄력적 IP 주소를 선택하고 작업, 탄력적 IP 주소 해제를 선택합니다.
4. 릴리스를 선택합니다.

기존 콘솔

엘라스틱 IP 주소를 해제합니다

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. 탄력적 IP 주소를 선택하고 작업을 선택한 후 주소 릴리스를 선택합니다. 메시지가 나타나면 릴리스를 선택합니다.

AWS CLI

엘라스틱 IP 주소를 해제합니다

`release-address` AWS CLI 명령을 사용합니다.

PowerShell

엘라스틱 IP 주소를 해제합니다

`Remove-EC2Address` Windows PowerShell용 AWS 도구 명령을 사용합니다.

탄력적 IP 주소 복구

탄력적 IP 주소를 릴리스한 경우 해당 주소를 복구할 수 있습니다. 다음 규칙이 적용됩니다.

- 탄력적 IP 주소가 다른 AWS 계정에 할당되었거나, 탄력적 IP 주소 한도를 초과하는 경우에는 탄력적 IP 주소를 복구할 수 없습니다.
- 탄력적 IP 주소와 연결된 태그는 복구할 수 있습니다.
- Amazon EC2 API 또는 명령줄 도구만을 사용하여 탄력적 IP 주소를 복구할 수 있습니다.

AWS CLI

탄력적 IP 주소를 복구하려면

다음과 같이 `allocate-address` AWS CLI 명령을 사용하고, `--address` 파라미터를 사용하여 IP 주소를 지정합니다.

```
aws ec2 allocate-address --domain vpc --address 203.0.113.3
```

PowerShell

탄력적 IP 주소를 복구하려면

다음과 같이 **New-EC2Address** Windows PowerShell용 AWS 도구 명령을 사용하고 -Address 파라미터를 사용하여 IP 주소를 지정합니다.

```
PS C:\> New-EC2Address -Address 203.0.113.3 -Domain vpc -Region us-east-1
```

이메일 애플리케이션에 역방향 DNS 사용

인스턴스에서 타사에 이메일을 보내려는 경우 탄력적 IP 주소를 하나 이상 프로비저닝하여 AWS에 제공하는 것이 좋습니다. AWS는 ISP 및 인터넷 스팸 방지 기관과의 공동 작업을 통해 이러한 주소에서 보내는 이메일에 스팸으로 플래그가 지정될 가능성을 줄입니다.

또한 이메일을 보내는 데 사용되는 탄력적 IP 주소에 고정 역방향 DNS 레코드를 지정하면 스팸 방지 기관에서 이메일에 스팸으로 플래그를 지정하는 것을 방지할 수 있습니다. 탄력적 IP 주소를 가리키는 정방향 DNS 레코드(레코드 유형 A)가 있어야 역방향 DNS 레코드를 만들 수 있습니다.

역방향 DNS 레코드가 탄력적 IP 주소와 연결되어 있는 경우 탄력적 IP 주소는 사용자 계정에 고정됩니다. 따라서 계정에서 탄력적 IP 주소를 해제하려면 해당 레코드를 제거해야 합니다.

이메일 전송 제한을 제거하거나 탄력적 IP 주소와 역방향 DNS 레코드를 제공하려면 [이메일 전송 제한 제거 요청](#) 페이지로 이동하십시오.

탄력적 IP 주소 제한

퍼블릭(IPv4) 인터넷 주소는 훈치 않은 퍼블릭 리소스이기 때문에 기본적으로 모든 AWS 계정은 리전당 5개의 탄력적 IP 주소로 제한됩니다. 인스턴스 장애 시 주소를 다른 인스턴스로 다시 매핑하는 기능이 필요할 때는 탄력적 IP 주소를 주로 사용하고, 다른 모든 노드 간 통신에는 DNS 호스트 이름을 사용하는 것이 좋습니다.

사용 중인 탄력적 IP 주소 수를 확인하려면

Amazon EC2에서 <https://console.aws.amazon.com/ec2/> 콘솔을 열고 탐색 창에서 탄력적 IP를 선택합니다.

탄력적 IP 주소에 대한 현재 계정 제한을 확인하려면

Amazon EC2 콘솔 또는 Service Quotas 콘솔에서 제한을 확인할 수 있습니다. 다음 중 하나를 수행하십시오.

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

탐색 창에서 제한을 선택한 다음 검색 필드에 **IP**를 입력합니다. 제한은 EC2-VPC 탄력적 IP입니다. EC2-Classic에 액세스할 수 있는 경우 EC2-Classic 탄력적 IP라는 추가 제한이 있습니다.

- <https://console.aws.amazon.com/servicequotas/>에서 Service Quotas 콘솔을 엽니다.

대시보드에서 Amazon Elastic Compute Cloud(Amazon EC2)를 선택합니다. Amazon Elastic Compute Cloud(Amazon EC2)가 대시보드에 나열되지 않은 경우 AWS 서비스를 선택하고 검색 필드에 **EC2**를 입력한 다음 Amazon Elastic Compute Cloud(Amazon EC2)를 선택합니다.

Amazon EC2 서비스 할당량 페이지의 검색 필드에 **IP**를 입력합니다. 제한은 EC2-VPC 탄력적 IP입니다. EC2-Classic에 액세스할 수 있는 경우 EC2-Classic 탄력적 IP라는 추가 제한이 있습니다. 자세한 내용을 보려면 제한을 선택하십시오.

아키텍처에서 추가 탄력적 IP 주소를 보증한다고 생각되면 Service Quotas 콘솔에서 직접 할당량 증가를 요청할 수 있습니다.

탄력적 네트워크 인터페이스

탄력적 네트워크 인터페이스는 VPC에서 가상 네트워크 카드를 나타내는 논리적 네트워킹 구성 요소입니다. 여기에는 다음 속성이 포함될 수 있습니다.

- VPC의 IPv4 주소 범위 중 기본 프라이빗 IPv4 주소
- VPC의 IPv4 주소 범위 중 하나 이상의 보조 프라이빗 IPv4 주소
- 프라이빗 IPv4 주소당 한 개의 탄력적 IP 주소(IPv4)
- 한 개의 퍼블릭 IPv4 주소
- 한 개 이상의 IPv6 주소
- 하나 이상의 보안 그룹
- MAC 주소
- 원본/대상 확인 플래그
- 설명

계정에서 네트워크 인터페이스를 생성 및 구성하고 VPC의 인스턴스에 연결할 수 있습니다. 계정에는 사용자가 다른 리소스 및 서비스를 사용할 수 있도록 AWS 서비스가 생성하고 관리하는 요청자 관리 네트워크 인터페이스가 있을 수도 있습니다. 이러한 네트워크 인터페이스는 사용자가 직접 관리할 수 없습니다. 자세한 내용은 [요청자 관리 네트워크 인터페이스 \(p. 750\)](#) 단원을 참조하십시오.

이 AWS 리소스를 AWS Management 콘솔 및 Amazon EC2 API에서 네트워크 인터페이스라고 합니다. 따라서 이 설명서에서는 “탄력적 네트워크 인터페이스” 대신 “네트워크 인터페이스”를 사용합니다. 이 설명서에서 “네트워크 인터페이스”라는 용어는 항상 “탄력적 네트워크 인터페이스”를 의미합니다.

목차

- [네트워크 인터페이스 기본 사항 \(p. 729\)](#)
- [인스턴스 유형별 네트워크 인터페이스당 IP 주소 \(p. 730\)](#)
- [네트워크 인터페이스 시나리오 \(p. 739\)](#)
- [네트워크 인터페이스 구성 모범 사례 \(p. 741\)](#)
- [네트워크 인터페이스 관련 작업 \(p. 741\)](#)
- [요청자 관리 네트워크 인터페이스 \(p. 750\)](#)

네트워크 인터페이스 기본 사항

네트워크 인터페이스를 만들고, 인스턴스에 연결하고, 인스턴스에서 분리한 후 다른 인스턴스에 연결할 수 있습니다. 인스턴스에 연결하거나 분리한 후 다른 인스턴스에 다시 연결하면 네트워크 인터페이스의 속성이 해당 네트워크 인터페이스를 따릅니다. 네트워크 인터페이스를 인스턴스 간에 이동하면 네트워크 트래픽이 새 인스턴스로 리디렉션됩니다.

사용자가 네트워크 인터페이스의 속성을 수정할 수도 있습니다. 예를 들어 보안 그룹을 변경하고 IP 주소를 관리할 수 있습니다.

VPC의 모든 인스턴스는 기본 네트워크 인터페이스라는 기본 네트워크 인터페이스를 갖습니다. 주 네트워크 인터페이스는 인스턴스에서 분리할 수 없습니다. 추가 네트워크 인터페이스를 만들고 연결할 수 있습니다. 사용 가능한 최대 네트워크 인터페이스 수는 인스턴스 유형에 따라 다릅니다. 자세한 내용은 [인스턴스 유형별 네트워크 인터페이스당 IP 주소 \(p. 730\)](#) 단원을 참조하십시오.

네트워크 인터페이스용 퍼블릭 IPv4 주소

VPC에서 모든 서브넷은 해당 서브넷에서 생성된(따라서 인스턴스가 그 서브넷으로 시작됨) 네트워크 인터페이스가 퍼블릭 IPv4 주소에 할당될 것인지 결정하는 수정 가능한 속성을 갖습니다. 자세한 내용은 Amazon VPC 사용 설명서의 [서브넷에 대한 IP 주소 지정 동작](#)을 참조하십시오. 퍼블릭 IPv4 주소는 Amazon의 퍼블

릭 IPv4 주소 풀에서 할당됩니다. 인스턴스를 시작하면 생성된 기본 네트워크 인터페이스에 IP 주소가 할당됩니다.

네트워크 인터페이스를 생성할 때 네트워크 인터페이스는 서브넷에서 퍼블릭 IPv4 주소 지정 속성을 상속합니다. 이후에 서브넷의 퍼블릭 IPv4 주소 지정 속성을 수정하면 네트워크 인터페이스는 처음 생성될 때 적용된 설정을 그대로 유지합니다. 인스턴스를 시작하고 기존 네트워크 인터페이스를 기본 네트워크 인터페이스로 지정하는 경우 퍼블릭 IPv4 주소 속성은 이 네트워크 인터페이스에 의해 결정됩니다.

자세한 내용은 [퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름 \(p. 703\)](#) 단원을 참조하십시오.

네트워크 인터페이스용 IPv6 주소

IPv6 CIDR 블록을 VPC 및 서브넷에 연결하고 서브넷 범위에 속하는 하나 이상의 IPv6 주소를 네트워크 인터페이스에 할당할 수 있습니다.

모든 서브넷은 해당 서브넷에서 생성된(따라서 인스턴스가 그 서브넷으로 시작된) 네트워크 인터페이스가 서브넷 범위에 속하는 IPv6 주소에 자동으로 할당될 것인지 결정하는 수정 가능한 속성을 갖습니다. 자세한 내용은 Amazon VPC 사용 설명서의 [서브넷에 대한 IP 주소 지정 동작](#)을 참조하십시오. 인스턴스를 시작하면 생성된 기본 네트워크 인터페이스에 IPv6 주소가 할당됩니다.

자세한 내용은 [IPv6 주소 \(p. 704\)](#) 단원을 참조하십시오.

IP 트래픽 모니터링

네트워크 인터페이스에서 VPC 흐름 로그를 활성화하여 네트워크 인터페이스로 주고 받는 IP 트래픽에 대한 정보를 캡처합니다. 흐름 로그를 생성하고 난 다음 Amazon CloudWatch Logs의 데이터를 확인하고 가져올 수 있습니다. 자세한 내용은 Amazon VPC 사용 설명서의 [VPC 흐름 로그](#)를 참조하십시오.

인스턴스 유형별 네트워크 인터페이스당 IP 주소

다음 표에는 인스턴스 유형별 최대 네트워크 인터페이스 수와 네트워크 인터페이스당 최대 프라이빗 IPv4 주소 및 IPv6 주소 수가 나열되어 있습니다. IPv6 주소 제한은 네트워크 인터페이스당 프라이빗 IPv4 주소 제한과 별개입니다. 모든 인스턴스 유형이 IPv6 주소 지정을 지원하는 것은 아닙니다. 네트워크 인터페이스, 여러 프라이빗 IPv4 주소, IPv6 주소는 VPC에서 실행 중인 인스턴스에만 사용할 수 있습니다. IPv6 주소는 퍼블릭이며 인터넷으로는 접속할 수 없습니다. 자세한 내용은 [다중 IP 주소 \(p. 709\)](#) 단원을 참조하십시오. VPC에서 IPv6에 대한 자세한 내용은 Amazon VPC 사용 설명서의 [VPC에서 IP 주소 지정](#) 단원을 참조하십시오.

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
c1.medium	2	6	IPv6는 지원되지 않습니다
c1.xlarge	4	15	IPv6는 지원되지 않습니다
c3.large	3	10	10
c3.xlarge	4	15	15
c3.2xlarge	4	15	15
c3.4xlarge	8	30	30
c3.8xlarge	8	30	30
c4.large	3	10	10
c4.xlarge	4	15	15
c4.2xlarge	4	15	15
c4.4xlarge	8	30	30

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 인스턴스 유형별 네트워크 인터페이스당 IP 주소

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
c4.8xlarge	8	30	30
c5.large	3	10	10
c5.xlarge	4	15	15
c5.2xlarge	4	15	15
c5.4xlarge	8	30	30
c5.9xlarge	8	30	30
c5.12xlarge	8	30	30
c5.18xlarge	15	50	50
c5.24xlarge	15	50	50
c5.metal	15	50	50
c5a.large	3	10	10
c5a.xlarge	4	15	15
c5a.2xlarge	4	15	15
c5a.4xlarge	8	30	30
c5a.8xlarge	8	30	30
c5a.12xlarge	8	30	30
c5a.16xlarge	15	50	50
c5a.24xlarge	15	50	50
c5ad.large	3	10	10
c5ad.xlarge	4	15	15
c5ad.2xlarge	4	15	15
c5ad.4xlarge	8	30	30
c5ad.8xlarge	8	30	30
c5ad.12xlarge	8	30	30
c5ad.16xlarge	15	50	50
c5ad.24xlarge	15	50	50
c5d.large	3	10	10
c5d.xlarge	4	15	15
c5d.2xlarge	4	15	15
c5d.4xlarge	8	30	30

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 인스턴스 유형별 네트워크 인터페이스당 IP 주소

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
c5d.9xlarge	8	30	30
c5d.12xlarge	8	30	30
c5d.18xlarge	15	50	50
c5d.24xlarge	15	50	50
c5d.metal	15	50	50
c5n.large	3	10	10
c5n.xlarge	4	15	15
c5n.2xlarge	4	15	15
c5n.4xlarge	8	30	30
c5n.9xlarge	8	30	30
c5n.18xlarge	15	50	50
c5n.metal	15	50	50
cc2.8xlarge	8	30	IPv6는 지원되지 않습니다
cr1.8xlarge	8	30	IPv6는 지원되지 않습니다
d2.xlarge	4	15	15
d2.2xlarge	4	15	15
d2.4xlarge	8	30	30
d2.8xlarge	8	30	30
f1.2xlarge	4	15	15
f1.4xlarge	8	30	30
f1.16xlarge	8	50	50
g2.2xlarge	4	15	IPv6는 지원되지 않습니다
g2.8xlarge	8	30	IPv6는 지원되지 않습니다
g3s.xlarge	4	15	15
g3.4xlarge	8	30	30
g3.8xlarge	8	30	30
g3.16xlarge	15	50	50
g4dn.xlarge	3	10	10
g4dn.2xlarge	3	10	10
g4dn.4xlarge	3	10	10

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 인스턴스 유형별 네트워크 인터페이스당 IP 주소

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
g4dn.8xlarge	4	15	15
g4dn.12xlarge	8	30	30
g4dn.16xlarge	4	15	15
g4dn.metal	15	50	50
h1.2xlarge	4	15	15
h1.4xlarge	8	30	30
h1.8xlarge	8	30	30
h1.16xlarge	15	50	50
hs1.8xlarge	8	30	IPv6는 지원되지 않습니다
i2.xlarge	4	15	15
i2.2xlarge	4	15	15
i2.4xlarge	8	30	30
i2.8xlarge	8	30	30
i3.large	3	10	10
i3.xlarge	4	15	15
i3.2xlarge	4	15	15
i3.4xlarge	8	30	30
i3.8xlarge	8	30	30
i3.16xlarge	15	50	50
i3.metal	15	50	50
i3en.large	3	10	10
i3en.xlarge	4	15	15
i3en.2xlarge	4	15	15
i3en.3xlarge	4	15	15
i3en.6xlarge	8	30	30
i3en.12xlarge	8	30	30
i3en.24xlarge	15	50	50
i3en.metal	15	50	50
m1.small	2	4	IPv6는 지원되지 않습니다
m1.medium	2	6	IPv6는 지원되지 않습니다

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 인스턴스 유형별 네트워크 인터페이스당 IP 주소

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
m1.large	3	10	IPv6는 지원되지 않습니다
m1.xlarge	4	15	IPv6는 지원되지 않습니다
m2.xlarge	4	15	IPv6는 지원되지 않습니다
m2.2xlarge	4	30	IPv6는 지원되지 않습니다
m2.4xlarge	8	30	IPv6는 지원되지 않습니다
m3.medium	2	6	IPv6는 지원되지 않습니다
m3.large	3	10	IPv6는 지원되지 않습니다
m3.xlarge	4	15	IPv6는 지원되지 않습니다
m3.2xlarge	4	30	IPv6는 지원되지 않습니다
m4.large	2	10	10
m4.xlarge	4	15	15
m4.2xlarge	4	15	15
m4.4xlarge	8	30	30
m4.10xlarge	8	30	30
m4.16xlarge	8	30	30
m5.large	3	10	10
m5.xlarge	4	15	15
m5.2xlarge	4	15	15
m5.4xlarge	8	30	30
m5.8xlarge	8	30	30
m5.12xlarge	8	30	30
m5.16xlarge	15	50	50
m5.24xlarge	15	50	50
m5.metal	15	50	50
m5a.large	3	10	10
m5a.xlarge	4	15	15
m5a.2xlarge	4	15	15
m5a.4xlarge	8	30	30
m5a.8xlarge	8	30	30
m5a.12xlarge	8	30	30

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 인스턴스 유형별 네트워크 인터페이스당 IP 주소

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
m5a.16xlarge	15	50	50
m5a.24xlarge	15	50	50
m5ad.large	3	10	10
m5ad.xlarge	4	15	15
m5ad.2xlarge	4	15	15
m5ad.4xlarge	8	30	30
m5ad.8xlarge	8	30	30
m5ad.12xlarge	8	30	30
m5ad.16xlarge	15	50	50
m5ad.24xlarge	15	50	50
m5d.large	3	10	10
m5d.xlarge	4	15	15
m5d.2xlarge	4	15	15
m5d.4xlarge	8	30	30
m5d.8xlarge	8	30	30
m5d.12xlarge	8	30	30
m5d.16xlarge	15	50	50
m5d.24xlarge	15	50	50
m5d.metal	15	50	50
m5dn.large	3	10	10
m5dn.xlarge	4	15	15
m5dn.2xlarge	4	15	15
m5dn.4xlarge	8	30	30
m5dn.8xlarge	8	30	30
m5dn.12xlarge	8	30	30
m5dn.16xlarge	15	50	50
m5dn.24xlarge	15	50	50
m5n.large	3	10	10
m5n.xlarge	4	15	15
m5n.2xlarge	4	15	15

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 인스턴스 유형별 네트워크 인터페이스당 IP 주소

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
m5n.4xlarge	8	30	30
m5n.8xlarge	8	30	30
m5n.12xlarge	8	30	30
m5n.16xlarge	15	50	50
m5n.24xlarge	15	50	50
p2.xlarge	4	15	15
p2.8xlarge	8	30	30
p2.16xlarge	8	30	30
p3.2xlarge	4	15	15
p3.8xlarge	8	30	30
p3.16xlarge	8	30	30
p3dn.24xlarge	15	50	50
r3.large	3	10	10
r3.xlarge	4	15	15
r3.2xlarge	4	15	15
r3.4xlarge	8	30	30
r3.8xlarge	8	30	30
r4.large	3	10	10
r4.xlarge	4	15	15
r4.2xlarge	4	15	15
r4.4xlarge	8	30	30
r4.8xlarge	8	30	30
r4.16xlarge	15	50	50
r5.large	3	10	10
r5.xlarge	4	15	15
r5.2xlarge	4	15	15
r5.4xlarge	8	30	30
r5.8xlarge	8	30	30
r5.12xlarge	8	30	30
r5.16xlarge	15	50	50

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 인스턴스 유형별 네트워크 인터페이스당 IP 주소

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
r5.24xlarge	15	50	50
r5.metal	15	50	50
r5a.large	3	10	10
r5a.xlarge	4	15	15
r5a.2xlarge	4	15	15
r5a.4xlarge	8	30	30
r5a.8xlarge	8	30	30
r5a.12xlarge	8	30	30
r5a.16xlarge	15	50	50
r5a.24xlarge	15	50	50
r5ad.large	3	10	10
r5ad.xlarge	4	15	15
r5ad.2xlarge	4	15	15
r5ad.4xlarge	8	30	30
r5ad.8xlarge	8	30	30
r5ad.12xlarge	8	30	30
r5ad.16xlarge	15	50	50
r5ad.24xlarge	15	50	50
r5d.large	3	10	10
r5d.xlarge	4	15	15
r5d.2xlarge	4	15	15
r5d.4xlarge	8	30	30
r5d.8xlarge	8	30	30
r5d.12xlarge	8	30	30
r5d.16xlarge	15	50	50
r5d.24xlarge	15	50	50
r5d.metal	15	50	50
r5dn.large	3	10	10
r5dn.xlarge	4	15	15
r5dn.2xlarge	4	15	15

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 인스턴스 유형별 네트워크 인터페이스당 IP 주소

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
r5dn.4xlarge	8	30	30
r5dn.8xlarge	8	30	30
r5dn.12xlarge	8	30	30
r5dn.16xlarge	15	50	50
r5dn.24xlarge	15	50	50
r5n.large	3	10	10
r5n.xlarge	4	15	15
r5n.2xlarge	4	15	15
r5n.4xlarge	8	30	30
r5n.8xlarge	8	30	30
r5n.12xlarge	8	30	30
r5n.16xlarge	15	50	50
r5n.24xlarge	15	50	50
t1.micro	2	2	IPv6는 지원되지 않습니다
t2.nano	2	2	2
t2.micro	2	2	2
t2.small	3	4	4
t2.medium	3	6	6
t2.large	3	12	12
t2.xlarge	3	15	15
t2.2xlarge	3	15	15
t3.nano	2	2	2
t3.micro	2	2	2
t3.small	3	4	4
t3.medium	3	6	6
t3.large	3	12	12
t3.xlarge	4	15	15
t3.2xlarge	4	15	15
t3a.nano	2	2	2
t3a.micro	2	2	2

인스턴스 유형	최대 네트워크 인터페이스 수	인터페이스당 프라이빗 IPv4 주소 수	인터페이스당 IPv6 주소 수
t3a.small	2	4	4
t3a.medium	3	6	6
t3a.large	3	12	12
t3a.xlarge	4	15	15
t3a.2xlarge	4	15	15
u-6tb1.metal	5	30	30
u-9tb1.metal	5	30	30
u-12tb1.metal	5	30	30
u-18tb1.metal	15	50	50
u-24tb1.metal	15	50	50
x1.16xlarge	8	30	30
x1.32xlarge	8	30	30
x1e.xlarge	3	10	10
x1e.2xlarge	4	15	15
x1e.4xlarge	4	15	15
x1e.8xlarge	4	15	15
x1e.16xlarge	8	30	30
x1e.32xlarge	8	30	30
z1d.large	3	10	10
z1d.xlarge	4	15	15
z1d.2xlarge	4	15	15
z1d.3xlarge	8	30	30
z1d.6xlarge	8	30	30
z1d.12xlarge	15	50	50
z1d.metal	15	50	50

네트워크 인터페이스 시나리오

다음을 수행하려는 경우 여러 네트워크 인터페이스를 하나의 인스턴스에 연결하면 유용합니다.

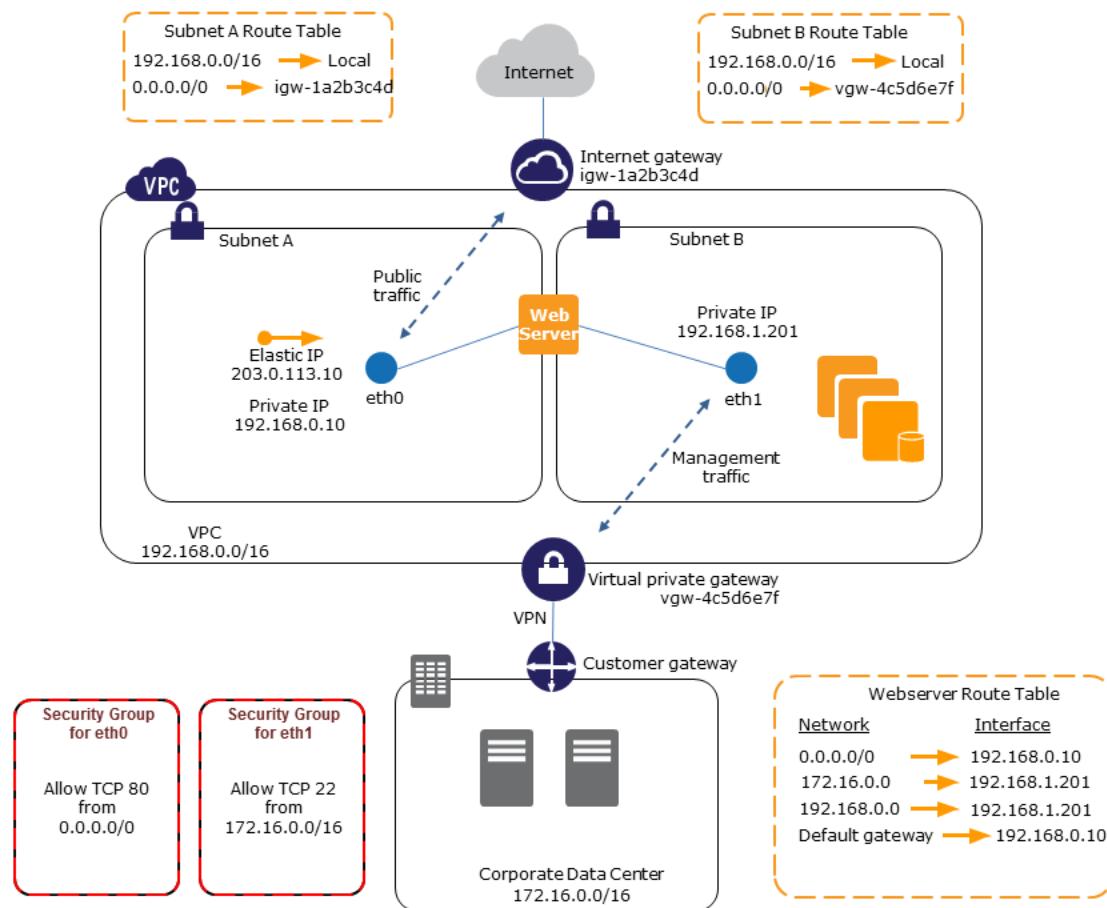
- 관리 네트워크 생성
- VPC에서 네트워크 및 보안 어플라이언스 사용
- 별도의 서브넷에 워크로드/역할이 있는 이중 흐름 인스턴스 생성

- 저예산 고가용성 솔루션 생성

관리 네트워크 생성

네트워크 인터페이스를 사용하여 관리 네트워크를 생성할 수 있습니다. 이 시나리오에서는 인스턴스의 주 네트워크 인터페이스(eth0)가 퍼블릭 트래픽을 처리하고 보조 네트워크 인터페이스(eth1)가 백엔드 관리 트래픽을 처리합니다. 또한 보조 네트워크 인터페이스는 VPC에서 액세스 제어가 더욱 제한적인 별도의 서브넷에 연결됩니다. 로드 밸런서를 지원하거나 지원하지 않을 수 있는 퍼블릭 인터페이스에 인터넷에서 서버에 액세스할 수 있도록 허용하는(예: 로드 밸런서 또는 0.0.0.0/0의 TCP 포트 80 및 443) 보안 그룹이 연결되는 반면, 프라이빗 인터페이스에는 VPC, 인터넷 또는 가상 프라이빗 게이트웨이 내 프라이빗 서브넷 안에 속하는 허용 IP 주소 범위에서만 RDP 액세스를 허용하는 보안 그룹이 연결됩니다.

장애 조치 기능을 유지하려면 네트워크 인터페이스에서 유입 트래픽에 대해 보조 프라이빗 IPv4 사용을 고려해 보십시오. 인스턴스 장애 발생 시 인터페이스 및/또는 보조 프라이빗 IPv4 주소를 스탠바이 인스턴스로 이동할 수 있습니다.



VPC에서 네트워크 및 보안 어플라이언스 사용

로드 밸런서, 네트워크 주소 변환(NAT) 서버 및 프록시 서버와 같은 일부 네트워크 및 보안 어플라이언스는 여러 네트워크 인터페이스로 구성하는 것이 좋습니다. 이러한 유형의 애플리케이션을 실행하는 부 네트워크 인터페이스를 VPC에서 생성 및 연결한 후 이 추가 인터페이스를 고유의 퍼블릭 및 프라이빗 IP 주소, 보안 그룹 및 원본/대상 확인으로 구성할 수 있습니다.

워크로드/역할이 개별 서브넷에 지정된 이중 흄 인스턴스 생성

애플리케이션 서버가 있는 중간 티어 네트워크에 연결되는 각각의 웹 서버에 네트워크 인터페이스를 배치할 수 있습니다. 애플리케이션 서버를 데이터베이스 서버가 있는 백엔드 네트워크(서브넷)에 이중 흄 상태로 연결할 수 있습니다. 이중 흄 인스턴스를 통한 라우팅 네트워크 패킷 대신 각 이중 흄 인스턴스가 프런트 엔드에서 요청을 수신 및 처리하고, 백엔드에 대한 연결을 초기화한 다음 백엔드 네트워크의 서버에 요청을 보냅니다.

저예산 고가용성 솔루션 생성

특정 기능을 제공하는 인스턴스 중 하나에 장애가 발생할 경우 서비스를 신속하게 복구하기 위해 관련 네트워크 인터페이스를 동일한 역할로 미리 구성된 대체 또는 핫 스탠바이 인스턴스에 연결할 수 있습니다. 예를 들어, 데이터베이스 인스턴스 또는 NAT 인스턴스와 같은 중요한 서비스에 대한 기본 또는 보조 네트워크 인터페이스로 네트워크 인터페이스를 사용할 수 있습니다. 인스턴스가 작동하지 않는 경우 사용자 또는 사용자를 대신하는 실행 중인 코드는 네트워크 인터페이스를 핫 스탠바이 인스턴스에 연결할 수 있습니다. 인터페이스에서 프라이빗 IP 주소, 탄력적 IP 주소 및 MAC 주소를 관리하므로 네트워크 인터페이스를 대체 인스턴스에 연결하자마자 네트워크 트래픽이 스탠바이 인스턴스로 전달되기 시작합니다. 인스턴스에 장애가 발생한 시간과 네트워크 인터페이스가 대기 인스턴스에 연결되는 시간 사이에 잠시 연결이 끊기지만 VPC 라우팅 테이블 또는 DNS 서버에 대해 어떠한 변경도 수행할 필요가 없습니다.

네트워크 인터페이스 구성 모범 사례

- 실행 중 상태(핫 연결), 중지 상태(웜 연결) 또는 시작 중 상태(콜드 연결)의 인터페이스에 네트워크 인터페이스를 연결할 수 있습니다.
- 인스턴스가 실행 중이거나 중지된 경우 보조 네트워크 인터페이스를 분리할 수 있습니다. 하지만 기본 네트워크 인터페이스는 분리할 수 없습니다.
- 인스턴스가 동일한 가용 영역과 VPC에 있지만 서로 다른 서브넷에 있는 경우 네트워크 인터페이스를 한 인스턴스에서 다른 인스턴스로 이동할 수 있습니다.
- CLI, API 또는 SDK에서 인스턴스를 시작할 때 기본 네트워크 인터페이스 및 추가 네트워크 인터페이스를 지정할 수 있습니다.
- 여러 네트워크 인터페이스를 포함하는 Amazon Linux 또는 Windows Server 인스턴스를 시작하면 인스턴스의 운영 체제에서 인터페이스, 프라이빗 IPv4 주소 및 라우팅 테이블이 자동으로 구성됩니다.
- 추가 네트워크 인터페이스의 웜 연결 또는 핫 연결을 수행하려면 수동으로 두 번째 인터페이스를 가동하고 프라이빗 IPv4 주소를 구성하며 라우팅 테이블을 그에 맞게 수정해야 합니다. Amazon Linux 또는 Windows Server를 실행하는 인스턴스는 웜 또는 핫 연결을 자동으로 인식하여 자체적으로 구성됩니다.
- 인스턴스에 다른 네트워크 인터페이스를 연결(예: NIC 팀 구성)하는 방법으로 이중 흄 인스턴스로 송/수신되는 네트워크 대역폭을 높이거나 두 배로 늘릴 수 없습니다.
- 동일한 서브넷에서 2개 이상의 네트워크 인터페이스를 인스턴스에 연결하면 비대칭 라우팅과 같은 네트워킹 문제가 발생할 수 있습니다. 가능한 한 기본 네트워크 인터페이스에서 보조 프라이빗 IPv4 주소를 대신 사용하십시오. 자세한 내용은 [보조 프라이빗 IPv4 주소 할당 \(p. 710\)](#) 단원을 참조하십시오. 여러 개의 네트워크 인터페이스를 사용해야 하는 경우에는 네트워크 인터페이스를 정적 라우팅을 사용할 수 있도록 구성해야 합니다. 자세한 내용은 [보조 네트워크 인터페이스 구성 \(p. 568\)](#) 단원을 참조하십시오.

네트워크 인터페이스 관련 작업

Amazon EC2 콘솔 또는 명령줄을 사용하여 네트워크 인터페이스 작업을 수행할 수 있습니다.

목차

- [네트워크 인터페이스 생성 \(p. 742\)](#)
- [네트워크 인터페이스 삭제 \(p. 742\)](#)
- [네트워크 인터페이스 세부 정보 보기 \(p. 743\)](#)
- [인스턴스 시작 시 네트워크 인터페이스 연결 \(p. 743\)](#)

- 중지되었거나 실행 중인 인스턴스에 네트워크 인터페이스 연결 (p. 744)
- 인스턴스에서 네트워크 인터페이스 분리 (p. 745)
- 보안 그룹 변경 (p. 746)
- 원본 또는 대상 확인 변경 (p. 746)
- 탄력적 IP 주소(IPv4) 연결 (p. 747)
- 탄력적 IP 주소(IPv4) 연결 해제 (p. 747)
- IPv6 주소 할당 (p. 748)
- IPv6 주소 할당 해제 (p. 748)
- 종료 동작 변경 (p. 748)
- 설명 추가 또는 편집 (p. 749)
- 태그 추가 또는 편집 (p. 749)

네트워크 인터페이스 생성

서브넷에서 네트워크 인터페이스를 생성할 수 있습니다. 네트워크 인터페이스는 일단 생성되고 나면 다른 서브넷으로 옮길 수 없으며 동일 가용 영역의 인스턴스에만 네트워크 인터페이스를 연결할 수 있습니다.

콘솔을 사용하여 네트워크 인터페이스를 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스 생성을 선택합니다.
4. 설명에 설명 이름을 입력합니다.
5. 서브넷에서 서브넷을 선택합니다.
6. 프라이빗 IP(또는 IPv4 프라이빗 IP)에 기본 프라이빗 IPv4 주소를 입력합니다. IPv4 주소를 지정하지 않는 경우 선택한 서브넷 내에서 사용 가능한 프라이빗 IPv4 주소가 선택됩니다.
7. (IPv6 전용) 연결된 IPv6 CIDR 블록이 있는 서브넷을 선택한 경우, 옵션으로 IPv6 IP 필드에서 IPv6 주소를 지정할 수 있습니다.
8. Elastic Fabric Adapter(EFA)를 생성하려면 Elastic Fabric Adapter(EFA)를 선택합니다.
9. 보안 그룹에서 하나 이상의 보안 그룹을 선택합니다.
10. (선택 사항) 태그 추가를 선택하고 태그 키와 태그 값을 입력합니다.
11. 예, 생성을 선택합니다.

명령줄을 사용하여 네트워크 인터페이스를 생성하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [create-network-interface](#)(AWS CLI)
- [New-EC2NetworkInterface](#)(Windows PowerShell용 AWS 도구)

네트워크 인터페이스 삭제

인스턴스를 삭제하려면 먼저 네트워크 인터페이스를 분리해야 합니다. 네트워크 인터페이스를 삭제하면 인터페이스와 연결된 모든 속성이 해제되고 다른 인스턴스에서 사용할 수 있도록 프라이빗 IP 주소나 탄력적 IP 주소가 해제됩니다.

콘솔을 사용하여 네트워크 인터페이스를 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 삭제를 선택합니다.
4. 네트워크 인터페이스 삭제 대화 상자에서 예, 삭제를 선택합니다.

명령줄을 사용하여 네트워크 인터페이스를 삭제하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [delete-network-interface](#)(AWS CLI)
- [Remove-EC2NetworkInterface](#)(Windows PowerShell용 AWS 도구)

네트워크 인터페이스 세부 정보 보기

계정에서 모든 네트워크 인스턴스를 볼 수 있습니다.

콘솔을 사용하여 네트워크 인터페이스를 설명하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택합니다.
4. 세부 정보를 확인하려면 세부 정보를 선택합니다.

명령줄을 사용하여 네트워크 인터페이스를 설명하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-network-interfaces](#)(AWS CLI)
- [Get-EC2NetworkInterface](#)(Windows PowerShell용 AWS 도구)

명령줄을 사용하여 네트워크 인터페이스 속성을 설명하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-network-interface-attribute](#)(AWS CLI)
- [Get-EC2NetworkInterfaceAttribute](#)(Windows PowerShell용 AWS 도구)

인스턴스 시작 시 네트워크 인터페이스 연결

인스턴스를 시작할 때 기존 네트워크 인터페이스를 지정하거나 네트워크 인터페이스를 추가적으로 연결할 수 있습니다.

Note

인스턴스에 네트워크 인터페이스를 연결할 때 오류가 발생하는 경우 이로 인해 인스턴스가 시작되지 않습니다.

콘솔을 사용하여 인스턴스 시작 시 네트워크 인터페이스를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 인스턴스 시작을 선택합니다.
3. AMI와 인스턴스 유형을 선택하고 다음: 인스턴스 세부 정보 구성(Next: Configure Instance Details)을 선택합니다.
4. 인스턴스 세부 정보 구성 페이지의 네트워크에서 VPC를, 서브넷에서 서브넷을 선택합니다.
5. 콘솔을 사용하면 네트워크 인터페이스 섹션에서 인스턴스 시작 시 네트워크 인터페이스(신규, 기존 또는 결합)를 두 개까지 지정할 수 있습니다. 새 인터페이스에 대해 하나의 기본 IPv4 주소와 하나 이상의 보조 IPv4 주소를 입력할 수도 있습니다.

시작한 후에는 추가로 다른 네트워크 인터페이스를 인스턴스에 추가할 수 있습니다. 연결 가능한 총 네트워크 인터페이스의 수는 인스턴스 유형에 따라 다릅니다. 자세한 내용은 [인스턴스 유형별 네트워크 인터페이스당 IP 주소 \(p. 730\)](#) 단원을 참조하십시오.

Note

네트워크 인터페이스를 두 개 이상 지정하면 퍼블릭 IPv4 주소를 인스턴스에 자동 할당할 수 없습니다.

6. (IPv6 전용) 연결된 IPv6 CIDR 블록이 있는 서브넷으로 인스턴스를 시작하는 경우, 연결한 모든 네트워크 인터페이스에 IPv6 주소를 지정할 수 있습니다. IPv6 IP에서 IP 추가를 선택합니다. 보조 IPv6 주소를 추가하려면 IP 추가를 다시 선택합니다. 서브넷 범위에 속한 IPv6 주소를 직접 입력하거나, Amazon이 서브넷에 속한 IPv6 주소를 자동으로 선택하도록 기본 자동 할당 값을 그대로 둘 수 있습니다.
7. 다음: 스토리지 추가를 선택합니다.
8. 스토리지 추가 페이지에서 사용자는 볼륨을 지정하여 AMI에 의해 지정된 볼륨 옆에 인스턴스(루트 디바이스 볼륨 등)를 연결한 다음 다음: 태그 추가를 선택합니다.
9. 태그 추가 페이지에서 인스턴스에 태그(예: 사용자에게 친숙한 이름)를 지정한 후 다음: 보안 그룹 구성(Next: Configure Security Group)을 선택합니다.
10. 보안 그룹 구성 페이지에서 보안 그룹을 선택하거나 새 보안 그룹을 만들 수 있습니다. 검토 및 시작을 선택합니다.

Note

5단계에서 기존 네트워크 인터페이스를 지정한 경우, 이 단계에서 어떤 옵션을 선택하든 상관 없이 인스턴스는 그 네트워크 인터페이스에 대한 보안 그룹과 연결됩니다.

11. 인스턴스 시작 검토 페이지에 기본 및 추가 네트워크 인터페이스에 대한 세부 정보가 표시됩니다. 설정을 검토한 다음 시작을 선택하여 키 페어를 선택하고 인스턴스를 시작합니다. Amazon EC2를 처음으로 사용하는 것이고 키 페어를 생성하지 않은 경우 새로운 키 페어를 생성하는 메시지가 마법사에 표시됩니다.

명령줄 사용하여 인스턴스 시작 시 네트워크 인터페이스를 연결하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [run-instances\(AWS CLI\)](#)
- [New-EC2Instance\(Windows PowerShell용 AWS 도구\)](#)

중지되었거나 실행 중인 인스턴스에 네트워크 인터페이스 연결

Amazon EC2 콘솔의 인스턴스 또는 네트워크 인터페이스 페이지를 사용하여 VPC에서 중지되었거나 실행 중인 인스턴스 중 하나에 네트워크 인터페이스를 연결할 수 있습니다.

Note

VPC 인스턴스의 퍼블릭 IP 주소가 해제되는 경우 인스턴스에 두 개 이상의 네트워크 인터페이스가 연결되어 있으면 새 퍼블릭 IPv4 주소를 받을 수 없습니다. 퍼블릭 IPv4 주소의 동작에 대한 자세한 내용은 [퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름 \(p. 703\)](#) 단원을 참조하십시오.

Instances 페이지를 사용하여 인스턴스에 네트워크 인터페이스를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 작업, 네트워킹, 네트워크 인터페이스 연결을 선택합니다.
4. 네트워크 인터페이스 연결 대화 상자에서 네트워크 인터페이스를 선택한 다음 연결을 선택합니다.

Network Interfaces 페이지를 사용하여 네트워크 인터페이스를 인스턴스에 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 연결을 선택합니다.
4. 네트워크 인터페이스 연결 대화 상자에서 인스턴스를 선택한 다음 연결을 선택합니다.

명령줄 사용하여 인스턴스에 네트워크 인터페이스를 연결하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [attach-network-interface](#)(AWS CLI)
- [Add-EC2NetworkInterface](#)(Windows PowerShell용 AWS 도구)

인스턴스에서 네트워크 인터페이스 분리

Amazon EC2 콘솔의 인스턴스 또는 네트워크 인터페이스 페이지를 사용하면 언제든지 EC2 인스턴스에 연결된 보조 네트워크 인터페이스를 분리할 수 있습니다.

Instances 페이지를 사용하여 인스턴스에서 네트워크 인터페이스를 분리하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 작업, 네트워킹, 네트워크 인터페이스 분리를 선택합니다.
4. 네트워크 인터페이스 분리 대화 상자에서 네트워크 인터페이스를 선택하고 분리를 선택합니다.

Amazon EC2 콘솔을 사용하면 Elastic Load Balancing 로드 밸런서, Lambda 함수, WorkSpace 또는 NAT 게이트웨이와 같은 다른 서비스에서 리소스에 연결된 네트워크 인터페이스를 분리할 수 없습니다. 리소스가 삭제되면 해당 리소스의 네트워크 인터페이스가 삭제됩니다.

네트워크 인터페이스 페이지를 사용하여 인스턴스에서 네트워크 인터페이스를 분리하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 설명을 확인하여 네트워크 인터페이스가 다른 유형의 리소스가 아닌 인스턴스에 연결되어 있는지 확인합니다. 리소스가 EC2 인스턴스인 경우 분리를 선택합니다.

네트워크 인터페이스가 인스턴스의 기본 네트워크 인터페이스인 경우 분리 버튼이 비활성화됩니다.

4. 확인 메시지가 나타나면 예, 분리를 선택합니다.
5. 네트워크 인터페이스가 인스턴스에서 분리되지 않으면 강제 분리를 선택하고 다시 시도합니다. 이 옵션은 최후의 수단으로만 사용하는 것이 좋습니다. 강제 분리하면 인스턴스를 다시 시작할 때까지 동일한 인덱스에 다른 네트워크 인터페이스가 연결되지 않을 수 있습니다. 또한 인스턴스를 다시 시작할 때까지 네트워크 인터페이스가 분리되었음을 인스턴스 메타데이터에 반영되지 않게 할 수 있습니다.

명령줄을 사용하여 네트워크 인터페이스를 분리하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [detach-network-interface\(AWS CLI\)](#)
- [Dismount-EC2NetworkInterface\(Windows PowerShell용 AWS 도구\)](#)

보안 그룹 변경

네트워크 인터페이스와 연결된 보안 그룹을 변경할 수 있습니다. 보안 그룹을 생성할 때는 네트워크 인터페이스의 서브넷과 동일한 VPC를 지정해야 합니다.

Note

다른 서비스에서 소유하는 인터페이스에 대한 보안 그룹 멤버십(예: Elastic Load Balancing)을 변경하려면 해당 서비스에 대한 콘솔이나 명령줄 인터페이스를 사용하십시오.

콘솔을 사용하여 네트워크 인터페이스의 보안 그룹을 변경하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 작업, 보안 그룹 변경을 선택합니다.
4. 보안 그룹 변경 대화 상자에서 사용할 보안 그룹을 선택하고 저장을 선택합니다.

명령줄을 사용하여 네트워크 인터페이스의 보안 그룹을 변경하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [modify-network-interface-attribute\(AWS CLI\)](#)
- [Edit-EC2NetworkInterfaceAttribute\(Windows PowerShell용 AWS 도구\)](#)

원본 또는 대상 확인 변경

Source/Destination Check 속성은 인스턴스에서 원본/대상 확인이 활성화/비활성화되었는지를 제어합니다. 이 속성을 비활성화하면 인스턴스에서 대상이 특별히 해당 인스턴스로 지정되지 않은 네트워크 트래픽을 처리할 수 있습니다. 예를 들어, 네트워크 주소 변환, 라우팅, 방화벽 등의 서비스를 실행 중인 인스턴스는 이 값을 disabled를 설정해야 합니다. 기본 값은 enabled입니다.

콘솔을 사용하여 네트워크 인터페이스에 대한 원본/대상 확인을 변경하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 작업, 소스/대상 확인을 선택합니다.
4. 대화 상자에서 활성화(활성화된 경우) 또는 비활성화(비활성화된 경우)를 선택하고 저장을 선택합니다.

명령줄을 사용하여 네트워크 인터페이스에 대한 원본/대상 확인을 변경하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [modify-network-interface-attribute](#)(AWS CLI)
- [Edit-EC2NetworkInterfaceAttribute](#)(Windows PowerShell용 AWS 도구)

탄력적 IP 주소(IPv4) 연결

탄력적 IP 주소(IPv4)가 있는 경우 이 주소를 네트워크 인터페이스에 대한 프라이빗 IPv4 주소 중 하나와 연결할 수 있습니다. 한 탄력적 IP 주소를 각 프라이빗 IPv4 주소와 연결할 수 있습니다.

Amazon EC2 콘솔이나 명령줄을 사용하여 탄력적 IP 주소를 연결할 수 있습니다.

콘솔을 사용하여 엘라스틱 IP 주소를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 작업, 주소 연결을 선택합니다.
4. 탄력적 IP 주소 연결 대화 상자의 주소 목록에서 탄력적 IP 주소를 선택합니다.
5. 프라이빗 IP 주소에 연결에서 탄력적 IP 주소와 연결할 프라이빗 IPv4 주소를 선택합니다.
6. 재연결 허용을 선택하여 탄력적 IP 주소가 현재 다른 인스턴스나 네트워크 인터페이스와 연결되어 있는 경우 지정된 네트워크 인터페이스와 연결될 수 있도록 한 다음 주소 연결을 선택합니다.

명령줄을 사용하여 탄력적 IP 주소를 연결하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [associate-address](#)(AWS CLI)
- [Register-EC2Address](#)(Windows PowerShell용 AWS 도구)

탄력적 IP 주소(IPv4) 연결 해제

네트워크 인터페이스에 연결된 탄력적 IP 주소(IPv4)가 있는 경우 해당 주소를 분리하고 다른 네트워크 인터페이스와 연결하거나 해제하여 주소 풀로 반환합니다. 네트워크 인터페이스는 특정 서브넷에서 고유하므로 네트워크 인터페이스를 사용하여 다른 서브넷이나 VPC의 인스턴스와 탄력적 IP 주소를 연결하는 방법은 이 방법뿐입니다.

Amazon EC2 콘솔이나 명령줄을 사용하여 탄력적 IP 주소를 분리할 수 있습니다.

콘솔을 사용하여 엘라스틱 IP 주소 연결을 끊으려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 작업, 주소 연결 해제를 선택합니다.
4. IP 주소 연결 해제 대화 상자에서 예, 연결 해제를 선택합니다.

명령줄을 사용하여 탄력적 IP 주소 연결을 끊으려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [disassociate-address](#)(AWS CLI)
- [Unregister-EC2Address](#)(Windows PowerShell용 AWS 도구)

IPv6 주소 할당

사용자는 네트워크 인터페이스에 하나 이상의 IPv6 주소를 할당할 수 있습니다. 네트워크 인터페이스는 연결된 IPv6 CIDR 블록이 있는 서브넷에 속해야 합니다. 네트워크 인터페이스에 특정 IPv6 주소를 할당하려면 IPv6 주소에 이미 다른 네트워크 인터페이스가 할당되어 있어서는 안 됩니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택한 후 네트워크 인터페이스를 선택합니다.
3. 작업, IP 주소 관리를 선택합니다.
4. IPv6 주소에서 새 IP 할당을 선택합니다. 서브넷의 주소 범위에 속한 IPv6 주소를 지정합니다. AWS가 자동으로 주소를 선택하도록 하려면 자동 할당 값을 그대로 둡니다.
5. 예, 업데이트를 선택합니다.

명령줄을 사용하여 네트워크 인터페이스에 IPv6 주소를 할당하려면

- 다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.
 - `assign-ipv6-addresses`(AWS CLI)
 - `Register-EC2Ipv6AddressList`(Windows PowerShell용 AWS 도구)

IPv6 주소 할당 해제

Amazon EC2 콘솔을 사용하여 네트워크 인터페이스에서 IPv6 주소 할당을 해제할 수 있습니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택한 후 네트워크 인터페이스를 선택합니다.
3. 작업, IP 주소 관리를 선택합니다.
4. IPv6 주소에서 제거할 IPv6 주소에 대해 할당 해제를 선택합니다.
5. 예, 업데이트를 선택합니다.

명령줄을 사용하여 네트워크 인터페이스에서 IPv6 주소 할당을 해제하려면

- 다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.
 - `unassign-ipv6-addresses`(AWS CLI)
 - `Unregister-EC2Ipv6AddressList`(Windows PowerShell용 AWS 도구).

종료 동작 변경

인스턴스에 연결된 네트워크 인터페이스의 종료 동작을 설정할 수 있습니다. 연결된 인스턴스를 종료할 때 네트워크 인터페이스를 자동으로 삭제할지를 지정할 수 있습니다.

Amazon EC2 콘솔 또는 명령줄을 사용하여 네트워크 인터페이스에 대한 종료 동작을 변경할 수 있습니다.

콘솔을 사용하여 네트워크 인터페이스에 대한 종료 동작을 변경하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 작업, 종료 방식 변경을 선택합니다.

4. 인스턴스를 종료할 때 네트워크 인터페이스를 삭제하려면 종료 방식 변경 대화 상자에서 종료 시 삭제 확인란을 선택합니다.

명령줄을 사용하여 네트워크 인터페이스에 대한 종료 동작을 변경하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [modify-network-interface-attribute](#)(AWS CLI)
- [Edit-EC2NetworkInterfaceAttribute](#)(Windows PowerShell용 AWS 도구)

설명 추가 또는 편집

Amazon EC2 콘솔 또는 명령줄을 사용하여 네트워크 인터페이스에 대한 설명을 변경할 수 있습니다.

콘솔을 사용하여 네트워크 인터페이스에 대한 설명을 변경하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 작업, 설명 변경을 선택합니다.
4. 설명 변경 대화 상자에서 네트워크 인터페이스에 대한 설명을 입력하고 저장을 선택합니다.

명령줄을 사용하여 네트워크 인터페이스에 대한 설명을 변경하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [modify-network-interface-attribute](#)(AWS CLI)
- [Edit-EC2NetworkInterfaceAttribute](#)(Windows PowerShell용 AWS 도구)

태그 추가 또는 편집

태그는 네트워크 인터페이스에 추가할 수 있는 메타데이터입니다. 태그는 개인적인 정보이므로 해당 계정에만 표시됩니다. 각 태그는 키와 값(선택 사항)으로 구성됩니다. 태그에 대한 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

콘솔을 사용하여 네트워크 인터페이스에 대한 태그를 추가하거나 편집하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택합니다.
4. 세부 정보 창에서 태그, 태그 추가/편집을 선택합니다.
5. 태그 추가/편집 대화 상자에서 생성할 각 태그에 대해 태그 생성을 선택하고 키와 값(선택 사항)을 입력합니다. 완료되면 저장을 선택합니다.

명령줄을 사용하여 네트워크 인터페이스에 대한 태그를 추가 또는 편집하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [create-tags](#)(AWS CLI)

- [New-EC2Tag](#)(Windows PowerShell용 AWS 도구)

요청자 관리 네트워크 인터페이스

요청자 관리 네트워크 인터페이스는 AWS 서비스가 사용자의 VPC에 생성하는 네트워크 인터페이스입니다. 이 네트워크 인터페이스는 Amazon RDS 인스턴스와 같은 다른 서비스의 인스턴스를 나타낼 수도 있고, 사용자가 AWS PrivateLink 서비스, Amazon ECS 작업 등 다른 서비스 또는 리소스에 액세스하도록 해줄 수도 있습니다.

요청자 관리 네트워크 인터페이스는 수정 또는 분리할 수 없습니다. 네트워크 인터페이스가 나타내는 리소스를 삭제할 경우 AWS 서비스가 자동으로 네트워크 인터페이스를 분리하고 삭제합니다. 요청자 관리 네트워크 인터페이스의 보안 그룹을 변경하려면 해당 서비스에 대한 콘솔 또는 명령줄 도구를 사용해야 할 수 있습니다. 자세한 내용은 서비스별 설명서를 참조하십시오.

요청자 관리 네트워크 인터페이스에 태그를 지정할 수 있습니다. 자세한 내용은 [태그 추가 또는 편집 \(p. 749\)](#) 단원을 참조하십시오.

본인의 계정에 포함된 요청자 관리 네트워크 인터페이스를 볼 수 있습니다.

콘솔을 사용하여 요청자 관리 네트워크 인터페이스를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 네트워크 인터페이스를 선택하고 세부 정보 창에서 다음 정보를 확인합니다.
 - 연결 소유자: 사용자가 네트워크 인터페이스를 생성한 경우 이 필드는 사용자의 AWS 계정 ID를 표시합니다. 그렇지 않은 경우, 네트워크 인터페이스를 생성한 보안 주체 또는 서비스의 별칭 또는 ID가 여기에 표시됩니다.
 - 설명: 네트워크 인터페이스의 용도에 관한 정보를 제공합니다. 예: "VPC 엔드포인트 인터페이스".

명령줄을 사용하여 요청자 관리 네트워크 인터페이스를 보려면

1. [describe-network-interfaces](#) AWS CLI 명령을 사용하여 계정의 네트워크 인터페이스를 설명합니다.

```
aws ec2 describe-network-interfaces
```

2. 다른 AWS 서비스가 네트워크 인터페이스를 관리하는 경우, 출력에서 RequesterManaged 필드가 true로 표시됩니다.

```
{  
    "Status": "in-use",  
    ...  
    "Description": "VPC Endpoint Interface vpce-089f2123488812123",  
    "NetworkInterfaceId": "eni-c8fbc27e",  
    "VpcId": "vpc-1a2b3c4d",  
    "PrivateIpAddresses": [  
        {  
            "PrivateDnsName": "ip-10-0-2-227.ec2.internal",  
            "Primary": true,  
            "PrivateIpAddress": "10.0.2.227"  
        }  
    ],  
    "RequesterManaged": true,  
    ...  
}
```

또는 [Get-EC2NetworkInterface](#) Windows PowerShell용 도구 명령을 사용합니다.

Windows에서 향상된 네트워킹

향상된 네트워킹에서는 [지원되는 인스턴스 유형 \(p. 751\)](#)에서 단일 루트 I/O 가상화(SR-IOV)를 사용하여 고성능 네트워킹 기능을 제공합니다. SR-IOV는 기존 가상 네트워크 인터페이스에 비해 높은 I/O 성능 및 낮은 CPU 사용률을 제공하는 디바이스 가상화 방법입니다. 향상된 네트워킹을 통해 대역폭과 PPS(Packet Per Second) 성능이 높아지고, 인스턴스 간 지연 시간이 지속적으로 낮아집니다. 향상된 네트워킹 사용에 따른 주가 요금은 없습니다.

목차

- [향상된 네트워킹 유형 \(p. 751\)](#)
- [인스턴스에서 향상된 네트워킹 기능 활성화 \(p. 751\)](#)
- [Windows 인스턴스에서 ENA\(Elastic Network Adapter\)를 사용하여 향상된 네트워킹 활성화 \(p. 751\)](#)
- [Windows 인스턴스에서 intel 82599 VF 인터페이스를 사용하여 향상된 네트워킹 활성화 \(p. 758\)](#)

향상된 네트워킹 유형

인스턴스 유형에 따라 다음 중 한 가지 메커니즘을 사용하여 향상된 네트워킹을 활성화 할 수 있습니다.

ENA(Elastic Network Adapter)

탄력적 네트워크 어댑터(ENA)는 지원되는 인스턴스 유형에 대해 최대 100Gbps의 네트워크 속도를 지원합니다.

[현재 세대 \(p. 115\)](#) 인스턴스는 m4.16xlarge보다 작은 M4 인스턴스를 제외하고 향상된 네트워킹을 위해 ENA를 지원합니다.

intel 82599 Virtual Function(VF) 인터페이스

intel 82599 Virtual Function 인터페이스는 지원되는 인스턴스 유형에 대해 최대 10Gbps의 네트워크 속도를 지원합니다.

C3, C4, D2, I2, M4(m4.16xlarge 제외) 및 R3 인스턴스 유형은 향상된 네트워킹을 위해 Intel 82599 VF 인터페이스를 사용합니다.

각 인스턴스 유형에 대해 지원되는 네트워크 속도에 대한 자세한 내용은 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

인스턴스에서 향상된 네트워킹 기능 활성화

인스턴스 유형에서 향상된 네트워킹을 위해 ENA를 지원하는 경우 [Windows 인스턴스에서 ENA\(Elastic Network Adapter\)를 사용하여 향상된 네트워킹 활성화 \(p. 751\)](#)의 절차를 따르십시오.

인스턴스 유형에서 향상된 네트워킹을 위해 intel 82599 VF 인터페이스를 지원하는 경우 [Windows 인스턴스에서 intel 82599 VF 인터페이스를 사용하여 향상된 네트워킹 활성화 \(p. 758\)](#)의 절차를 따르십시오.

Windows 인스턴스에서 ENA(Elastic Network Adapter)를 사용하여 향상된 네트워킹 활성화

Amazon EC2는 ENA(Elastic Network Adapter)를 통해 향상된 네트워킹을 제공합니다. 향상된 네트워킹을 사용하려면 필수 ENA 모듈을 설치하고 ENA 지원을 활성화해야 합니다.

내용

- [요구 사항 \(p. 752\)](#)

- 향상된 네트워킹 성능 (p. 752)
- 향상된 네트워킹 기능 활성화 여부 확인 (p. 752)
- Windows에서 향상된 네트워킹 기능 활성화 (p. 753)
- Amazon ENA 드라이버 버전 (p. 754)
- 알림 구독 (p. 527)
- 운영 체제 최적화 (p. 757)

요구 사항

ENA를 사용하여 향상된 네트워킹을 준비하려면 인스턴스를 다음과 같이 설정하십시오.

- m4.16xlarge보다 작은 M4 인스턴스를 제외하고 현재 세대 (p. 115) 인스턴스 유형을 사용하여 인스턴스를 시작합니다.
- 인스턴스가 인터넷에 연결되어 있는지 확인합니다.
- AWS CLI 또는 Windows PowerShell용 AWS 도구를 자신이 선택한 컴퓨터에 설치하고 구성합니다(로컬 데스크톱/노트북 권장). 자세한 내용은 Amazon EC2에 액세스 (p. 3) 단원을 참조하십시오. Amazon EC2 콘솔에서는 향상된 네트워킹을 관리할 수 없습니다.
- 인스턴스에 보존해야 할 중요한 데이터가 있는 경우 인스턴스에서 AMI를 만들어 데이터를 백업해야 합니다. 커널 및 커널 모듈 업데이트 외에도 enaSupport 속성을 활성화하면 호환되지 않는 인스턴스나 운영 체제에 접속할 수 없게 됩니다. 최신 백업을 확보하면 이 경우에도 데이터를 보존할 수 있습니다.

향상된 네트워킹 성능

다음 설명서에는 ENA 향상된 네트워킹을 지원하는 인스턴스 유형의 네트워크 성능이 요약되어 있습니다.

- 액셀러레이티드 컴퓨팅 인스턴스의 네트워크 성능 (p. 185)
- 컴퓨팅 최적화 인스턴스의 네트워크 성능 (p. 165)
- 범용 인스턴스의 네트워크 성능 (p. 125)
- 메모리 최적화 인스턴스의 네트워크 성능 (p. 172)
- 스토리지 최적화 인스턴스의 네트워크 성능 (p. 179)

향상된 네트워킹 기능 활성화 여부 확인

향상된 네트워킹 기능이 활성화되었는지를 알아보려면 인스턴스에 드라이버가 설치되어 있는지, 그리고 enaSupport 속성이 설정되어 있는지를 확인해야 합니다.

인스턴스 속성(enaSupport)

다음 명령 중 하나를 사용하여 인스턴스에 향상된 네트워킹 enaSupport 속성 세트가 있는지 확인할 수 있습니다. 속성이 설정되었으면 true가 반환됩니다.

- [describe-instances](#) (AWS CLI)

```
aws ec2 describe-instances --instance-ids instance_id --query  
"Reservations[].[Instances[]].EnaSupport"
```

- [Get-EC2Instance](#) Windows PowerShell용 도구

```
(Get-EC2Instance -InstanceId instance-id).Instances.EnaSupport
```

이미지 속성(enaSupport)

다음 명령 중 하나를 사용하여 AMI에 향상된 네트워킹 enaSupport 속성이 설정되어 있는지 확인할 수 있습니다. 속성이 설정되었으면 true가 반환됩니다.

- [describe-images](#)(AWS CLI)

```
aws ec2 describe-images --image-id ami_id --query "Images[ ].EnaSupport"
```

- [Get-EC2Image](#) Windows PowerShell용 도구

```
(Get-EC2Image -ImageId ami_id).EnaSupport
```

Windows에서 향상된 네트워킹 기능 활성화

확장 네트워크를 설정하지 않은 상태로 인스턴스를 시작한 경우에는 인스턴스에 필요한 네트워크 어댑터 드라이버를 다운로드하여 설치한 다음 enaSupport 인스턴스 속성을 설정하여 확장 네트워크를 활성화해야 합니다. 이 속성은 지원되는 인스턴스 유형 및 ENA 드라이버가 설치된 경우에만 활성화할 수 있습니다. 자세한 내용은 [향상된 네트워킹 유형 \(p. 751\)](#) 단원을 참조하십시오.

향상된 네트워킹을 활성화하려면

1. 인스턴스 연결 후 로컬 관리자로 로그인합니다.
2. [Windows Server 2016 이상] 다음 EC2Launch PowerShell 스크립트를 실행하여 드라이버가 설치된 후의 인스턴스를 구성합니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -  
Schedule
```

3. 다음과 같이 인스턴스 상에서 드라이버를 설치합니다.

- a. 최신 드라이버를 인스턴스로 [다운로드](#)합니다.
- b. ZIP 아카이브를 추출합니다.
- c. install.ps1 PowerShell 스크립트를 실행하여 드라이버를 설치합니다.

Note

실행 정책 오류가 발생하면 정책을 Unrestricted(기본값으로 Restricted 또는 RemoteSigned로 설정되어 있음)로 설정합니다. 명령 줄에서 Set-ExecutionPolicy -ExecutionPolicy Unrestricted를 실행한 다음 install.ps1 PowerShell 스크립트를 다시 실행하십시오.

4. 로컬 컴퓨터를 사용하는 경우, Amazon EC2 콘솔을 사용하거나 다음 명령 중 하나를 사용하여 인스턴스를 중지하십시오. [stop-instances](#)(AWS CLI), [Stop-EC2Instance](#)(Windows PowerShell용 AWS 도구). 인스턴스를 AWS OpsWorks에서 관리할 경우 AWS OpsWorks 콘솔에서 인스턴스를 중지해야 인스턴스 상태가 동기화됩니다.
5. 다음과 같이 인스턴스에서 ENA 지원을 활성화합니다.
 - a. 로컬 컴퓨터에서 다음 명령 중 하나를 실행하여 해당 인스턴스의 EC2 인스턴스 ENA 지원 속성을 확인합니다. 이 속성이 활성 상태가 아니면 "[]" 또는 공백이 출력됩니다. 기본적으로 EnaSupport는 false로 설정됩니다.

- [describe-instances](#) (AWS CLI)

```
aws ec2 describe-instances --instance-ids instance_id --query  
"Reservations[ ].Instances[ ].EnaSupport"
```

- [Get-EC2Instance](#) Windows PowerShell용 도구

```
(Get-EC2Instance -InstanceId instance-id).Instances.EnaSupport
```

- b. ENA 지원을 활성화하려면 다음 명령 중 하나를 실행합니다.

- [modify-instance-attribute\(AWS CLI\)](#)

```
aws ec2 modify-instance-attribute --instance-id instance_id --ena-support
```

- [Edit-EC2InstanceAttribute\(Windows PowerShell용 AWS 도구\)](#)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -EnaSupport $true
```

인스턴스를 재시작할 때 문제가 발생하는 경우 다음 명령 중 하나를 사용하여 ENA 지원을 비활성화할 수도 있습니다.

- [modify-instance-attribute\(AWS CLI\)](#)

```
aws ec2 modify-instance-attribute --instance-id instance_id --no-ena-support
```

- [Edit-EC2InstanceAttribute\(Windows PowerShell용 AWS 도구\)](#)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -EnaSupport $false
```

- c. 이전 설명과 같이 `describe-instances` 또는 `Get-EC2Instance`를 사용하여 속성이 `true`로 설정되어 있는지 확인합니다. 이제 다음 결과가 표시됩니다.

```
[  
    true  
]
```

6. 로컬 컴퓨터를 사용하는 경우, Amazon EC2 콘솔을 사용하거나 다음 명령 중 하나를 사용하여 인스턴스를 시작하십시오. [start-instances\(AWS CLI\)](#), [Start-EC2Instance\(Windows PowerShell용 AWS 도구\)](#). 인스턴스를 AWS OpsWorks에서 관리할 경우 AWS OpsWorks 콘솔을 사용하여 인스턴스를 시작해야 인스턴스 상태가 동기화됩니다.
7. 인스턴스에서 다음과 같이 ENA 드라이버가 설치되고 활성화되어 있는지 확인합니다.
- 네트워크 아이콘을 마우스 오른쪽 버튼으로 클릭하고 네트워크 및 공유 센터 열기(Open Network and Sharing Center)를 선택합니다.
 - 이더넷 어댑터(예: Ethernet 2)를 선택합니다.
 - 세부 정보를 선택합니다. 네트워크 연결 세부 정보(Network Connection Details)에서 설명>Description>이 Amazon Elastic Network Adapter인지 확인합니다.
8. (선택 사항) 인스턴스에서 AMI를 만듭니다. 생성된 AMI는 인스턴스의 `enaSupport` 속성을 상속합니다. 따라서 이 AMI를 사용하여 기본적으로 ENA가 활성화된 상태로 다른 인스턴스를 시작할 수 있습니다. 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오.

Amazon ENA 드라이버 버전

Windows AMI에는 향상된 네트워킹을 활성화하기 위한 Amazon ENA 드라이버가 포함되어 있습니다. 다음 표에는 각 릴리스에 대한 변경 사항이 요약되어 있습니다.

드라이버 버전	세부 정보	릴리스 날짜
2.2.0	새로운 기능	2020년 8월 12일

드라이버 버전	세부 정보	릴리스 날짜
	<ul style="list-style-type: none"> 차세대 하드웨어 유형에 대한 지원을 추가합니다. 중지-최대 절전 모드에서 재개한 후 인스턴스 시작 시간을 개선하고 오탑지 ENA 오류 메시지를 제거합니다. <p>성능 최적화</p> <ul style="list-style-type: none"> 인바운드 트래픽의 처리를 최적화합니다. 낮은 리소스 환경에서 공유 메모리 관리를 개선합니다. <p>버그 수정</p> <ul style="list-style-type: none"> 드문 경우이지만 드라이버를 재설정하지 못하는 경우 ENA 디바이스 제거 시 시스템 충돌을 방지합니다. 	
2.1.5	버그 수정	2020년 6월 23일
	<ul style="list-style-type: none"> 베어 메탈 인스턴스에서 가끔 발생하는 네트워크 어댑터 초기화 실패를 해결합니다. 	
2.1.4	버그 수정	2019년 11월 25일
	<ul style="list-style-type: none"> 네트워크 스택에서 도착하는 손상된 LSO 패킷 메타데이터로 인해 발생하는 연결 문제를 방지합니다. 이미 릴리스된 패킷 메모리에 액세스하게 되는 드문 교착 상태로 인해 발생하는 시스템 충돌을 방지합니다. 	
2.1.2	<p>새로운 기능</p> <ul style="list-style-type: none"> OS에서 MAC 기반 UUID를 생성할 수 있도록 공급업체 ID 보고서에 대한 지원이 추가되었습니다. <p>버그 수정</p> <ul style="list-style-type: none"> 초기화 시 DHCP 네트워크 구성 성능이 향상되었습니다. 최대 전송 단위(MTU)가 4K를 초과하는 경우 인바운드 IPv6 트래픽에서 L4 체크섬을 적절히 계산합니다. 드라이버 안정성 및 사소한 버그 수정에 대한 전반적인 개선 사항입니다. 	2019년 11월 4일
2.1.1	버그 수정	2019년 9월 16일
	<ul style="list-style-type: none"> 운영 체제에서 매우 조각화된 TCP LSO 패킷이 떨어지는 것을 방지합니다. IPv6 네트워크의 IPSec 내에서 캡슐화 보안 페이로드(ESP) 프로토콜을 적절히 처리합니다. 	

드라이버 버전	세부 정보	릴리스 날짜
2.1.0	<p>ENA Windows 드라이버 v2.1은 새로운 ENA 디바이스 기능을 도입하고 성능 향상을 제공하며 새로운 기능을 추가하고 여러 안정성 개선 기능을 포함합니다.</p> <ul style="list-style-type: none"> 새로운 기능 <ul style="list-style-type: none"> 점보 프레임 구성에 표준화된 Windows 레지스트리 키를 사용합니다. ENA 드라이버 속성 GUI를 통한 VLAN ID 설정을 허용합니다. 복구 흐름이 개선되었습니다. <ul style="list-style-type: none"> 결함 식별 메커니즘이 개선되었습니다. 튜닝 가능한 복구 파라미터에 대한 지원이 추가되었습니다. vCPU가 8개 이상인 최신 EC2 인스턴스의 경우 최대 32개의 I/O 대기열을 지원합니다. 드라이버 메모리 공간 90%까지 절감 성능 최적화 <ul style="list-style-type: none"> 전송 경로 지연 시간 감소 수신 체크섬 오프로드를 지원합니다. 과다 로드된 시스템(잠금 메커니즘의 사용 최적화)을 위한 성능 최적화 CPU 사용률을 줄이고 로드 시 시스템 응답 속도를 개선하는 추가 향상 기능 버그 수정 <ul style="list-style-type: none"> 불연속 Tx 헤더의 유효하지 않은 구문 분석으로 인한 충돌을 수정했습니다. Bare Metal 인스턴스에서 ENI 분리 중 드라이버 v1.5 충돌 문제를 수정했습니다. IPv6에 대한 LSO 의사 헤더 체크섬 계산 오류를 수정했습니다. 초기화 실패 시 잠재적인 메모리 리소스 유출을 수정했습니다. IPv4 조각에 대한 TCP/UDP 체크섬 오프로드를 비활성화했습니다. VLAN 구성에 대해 수정했습니다. VLAN 우선 순위만 비활성화해야 한 경우 VLAN이 잘못 비활성화되었습니다. 이벤트 뷰어로 사용자 지정 드라이버 메시지의 올바른 구문 분석을 활성화했습니다. 잘못된 타임스탬프 처리로 인한 드라이버 초기화 실패 문제를 수정했습니다. 데이터 처리 및 ENA 디바이스 비활성화 사이의 교착 상태를 수정했습니다. 	2019년 7월 1일
1.5.0	<ul style="list-style-type: none"> 안정성 및 성능 수정 사항이 개선되었습니다. 이제, ENA NIC의 고급 속성에서 수신 버퍼를 최대 8192의 값으로 구성할 수 있습니다. 기본 수신 버퍼는 1K입니다. 	2018년 10월 4일

드라이버 버전	세부 정보	릴리스 날짜
1.2.3	안정성 수정 사항이 포함되고, Windows Server 2008 R2부터 Windows Server 2016에 이르는 지원을 통합합니다.	2018년 2월 13일
1.0.9	일부 안정성 수정 사항을 포함합니다. Windows Server 2008 R2에만 적용됩니다. 다른 버전의 Windows Server에는 권장하지 않습니다.	2016년 12월
1.0.8	최초 릴리스입니다. Windows Server 2008 R2, Windows Server 2012 RTM, Windows Server 2012 R2 및 Windows Server 2016용 AMI에 포함됩니다.	2016년 7월

알림 구독

새로운 EC2 Windows Driver 버전이 릴리스되면 이를 알리도록 Amazon SNS를 설정할 수 있습니다. 알림을 받으려면 다음 절차를 수행합니다.

EC2 알림을 받으려면

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 필요한 경우 탐색 모음에서 리전을 미국 동부(버지니아 북부)로 변경하십시오. 구독을 신청하는 SNS 알림이 이 지역에 있기 때문에 이 지역을 선택해야 합니다.
3. 탐색 창에서 구독을 선택합니다.
4. 구독 생성을 선택합니다.
5. 구독 생성 대화 상자에서 다음 작업을 수행합니다.
 - a. TopicARN의 경우, 다음 Amazon 리소스 이름(ARN)을 복사합니다.
`arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers`
 - b. 프로토콜에서 `Email`을 선택합니다.
 - c. 앤드포인트에는 알림을 받는 데 사용할 수 있는 이메일 주소를 입력합니다.
 - d. `Create subscription`을 선택합니다.
6. 확인 이메일이 발송됩니다. 이메일을 열고 지침에 따라 구독을 완료합니다.

새 EC2 Windows 드라이버가 릴리스될 때마다 구독자에게 알림이 전송됩니다. 이런 알림을 더 이상 받지 않기를 원하는 경우, 다음 절차를 수행해서 구독을 해제하십시오.

Amazon EC2 Windows 드라이버 알림을 구독 해제하려면

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 탐색 창에서 구독을 선택합니다.
3. 구독 확인란을 선택한 후 작업, 구독 삭제를 선택합니다. 확인 메시지가 나타나면 `Delete`를 선택합니다.

운영 체제 최적화

향상된 네트워킹을 지원하는 인스턴스에서 네트워크 성능을 최대화하려는 경우 기본 운영 체제 구성은 수정해야 할 수도 있습니다. 높은 수준의 네트워크 성능이 필요한 애플리케이션에는 다음 구성 변경을 권장합니다. 그 밖의 최적화(예: 체크섬 오프로드 사용, RSS 활성화)는 공식 Windows AMI에 이미 준비되어 있습니다.

Note

대부분 용도에서는 TCP Chimney 오프로드를 비활성화해야 하며, Windows Server 2016부터는 더 이상 사용되지 않았습니다.

이러한 운영 체제 최적화 외에도, 네트워크 트래픽의 최대 전송 단위(MTU)를 고려하여 워크로드 및 네트워크 아키텍처에 맞게 조정해야 합니다. 자세한 내용은 [EC2 인스턴스에 대한 네트워크 MTU\(최대 전송 단위\) \(p. 772\)](#) 단원을 참조하십시오.

AWS는 클러스터 배치 그룹에서 시작된 인스턴스 간의 평균 왕복 지연 시간(50us)과 99.9 백분위 템포 지연 시간(200us)을 정기적으로 측정합니다. 애플리케이션에서 일관되게 낮은 지연 시간을 요구하는 경우 고정 성능 Nitro 기반 인스턴스에 최신 버전의 ENA 드라이버를 사용하는 것이 좋습니다.

RSS CPU 선호도 구성

네트워크 트래픽 CPU 로드를 여러 프로세서로 분산하는 데 수신 측 조정(RSS)이 사용됩니다. 기본적으로 공식 Amazon Windows AMI는 RSS가 활성화된 상태로 구성됩니다. ENA ENI는 최대 8개의 RSS 대기열을 제공합니다. RSS 대기열 및 다른 시스템 프로세스에 대한 CPU 선호도를 정의하면, 다중 코어 시스템으로 CPU 로드를 분산해 더 많은 네트워크 트래픽을 처리할 수 있습니다. vCPU를 16개 이상 사용하는 인스턴스 유형의 경우, `Set-NetAdapterRSS` PowerShell cmdlt(Windows Server 2012 이상에서 사용 가능) 사용을 권장합니다. 다양한 시스템 구성요소의 경합을 방지하기 위해 모든 ENI의 RSS 구성에서 부트 프로세서(하이퍼 스레딩 활성화 시 논리 프로세서 0 및 1)를 수동으로 제외합니다.

Windows는 하이퍼 스레드를 인식하며, 단일 NIC의 RSS 대기열이 언제나 다른 물리적 코어에 위치하게 합니다. 따라서 다른 NIC와의 경합 방지를 위해 하이퍼 스레딩을 비활성화하지 않는다면, 각 NIC의 RSS 구성은 16개의 논리적 프로세서에 분산합니다. `Set-NetAdapterRSS cmdlt`를 이용하면 `BaseProcessorGroup`, `BaseProcessorNumber`, `MaxProcessingGroup`, `MaxProcessorNumber`와 `NumaNode`(선택 사항) 값을 정의해, 유효한 논리적 프로세서의 NIC 범위를 정의할 수 있습니다. 물리적 코어가 부족해 NIC간 경합을 완전히 제거할 수 없다면, 중첩되는 범위를 최소화하거나 ENI의 예상 워크로드에 따라 ENI 범위에 있는 논리적 프로세서의 수를 줄여보십시오(적은 볼륨의 관리자 네트워크 ENI는 활성화된 RSS 대기열보다 적은 대기열이 필요할 수 있습니다). 또한 위에서 언급했듯이 다양한 구성요소를 CPU 0에서 실행해야 하며, 따라서 충분한 vCPU를 사용할 수 있다면 모든 RSS 구성에서 이를 제외하는 것이 좋습니다.

예를 들어 하이퍼 스레딩을 활성화한 NUMA 노드 2개가 있는 72 vCPU 인스턴스에 ENI 3개가 있다면, 다음 명령은 중첩되거나 코어 0 사용을 완전히 방지하지 않고도 네트워크 로드를 두 CPU에 분산합니다.

```
Set-NetAdapterRSS -Name NIC1 -BaseProcessorGroup 0 -BaseProcessorNumber 2 -  
MaxProcessorNumber 16  
Set-NetAdapterRSS -Name NIC2 -BaseProcessorGroup 1 -BaseProcessorNumber 0 -  
MaxProcessorNumber 14  
Set-NetAdapterRSS -Name NIC3 -BaseProcessorGroup 1 -BaseProcessorNumber 16 -  
MaxProcessorNumber 30
```

이러한 설정은 각 네트워크 어댑터에 영구적으로 적용됩니다. 인스턴스가 vCPU 숫자가 다른 인스턴스로 크기 조정된다면, 활성화된 각 ENI에 대해 RSS 구성은 재평가해야 합니다. `Set-NetAdapterRSS cmdlt`에 대한 Microsoft 설명서 전문은 <https://docs.microsoft.com/en-us/powershell/module/netadapter/set-netadapterrss>에서 확인할 수 있습니다.

SQL 워크로드 전용 참고사항: 같은 CPU에 대한 IO와 네트워크 경합을 최소화할 수 있도록, IO 스레드 선호도 설정과 ENI RSS 구성은 검토해 보십시오. 자세한 내용은 [선호도 마스크 서버 구성 옵션](#)을 확인하십시오.

Windows 인스턴스에서 intel 82599 VF 인터페이스를 사용하여 향상된 네트워킹 활성화

Amazon EC2에서는 Intel ixgbevf 드라이버를 사용하는 Intel 82599 VF 인터페이스를 통해 향상된 네트워킹 기능을 제공합니다.

내용

- [요구 사항 \(p. 759\)](#)
- [향상된 네트워킹 기능 활성화 여부 확인 \(p. 759\)](#)
- [Windows에서 향상된 네트워킹 활성화 \(p. 760\)](#)

요구 사항

intel 82599 VF 인터페이스를 사용하여 향상된 네트워킹을 준비하려면 인스턴스를 다음과 같이 설정하십시오.

- C3, C4, D2, I2, M4(m4.16xlarge 제외) 및 R3의 지원되는 인스턴스 유형에서 선택합니다.
- 64비트 HVM AMI에서 인스턴스를 시작합니다. Windows Server 2008과 Windows Server 2003에서는 향상된 네트워킹을 활성화할 수 없습니다. Windows Server 2012 R2와 Windows Server 2016 이상 AMI에서는 확장 네트워킹이 이미 활성화되어 있습니다. Windows Server 2012 R2에는 intel 드라이버 1.0.15.30이 포함되어 있으므로 Pnputil.exe 유ти리티를 사용하여 해당 드라이버를 최신 버전으로 업그레이드하는 것이 좋습니다.
- 인스턴스가 인터넷에 연결되어 있는지 확인합니다.
- [AWS CLI](#) 또는 [Windows PowerShell용 AWS 도구](#)를 자신이 선택한 컴퓨터에 설치하고 구성합니다(로컬 데스크톱/노트북 권장). 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오. Amazon EC2 콘솔에서는 향상된 네트워킹을 관리할 수 없습니다.
- 인스턴스에 보존해야 할 중요한 데이터가 있는 경우 인스턴스에서 AMI를 만들어 데이터를 백업해야 합니다. 커널 및 커널 모듈 업데이트 외에도 sriovNetSupport 속성을 활성화하면 호환되지 않는 인스턴스나 운영 체제에 접속할 수 없게 됩니다. 최신 백업을 확보하면 이 경우에도 데이터를 보존할 수 있습니다.

향상된 네트워킹 기능 활성화 여부 확인

드라이버가 인스턴스에 설치되어 있고 sriovNetSupport 속성이 설정된 경우 intel 82599 VF 인터페이스를 사용하는 향상된 네트워킹이 활성화됩니다.

드라이버

드라이버의 설치 여부를 확인하려면 인스턴스에 연결한 뒤 Device Manager를 실행합니다. 네트워크 어댑터 (Network adapters) 아래에 "intel(R) 82599 Virtual Function"이 표시되면 정상입니다.

인스턴스 속성(sriovNetSupport)

다음 명령 중 하나를 사용하여 인스턴스에 향상된 네트워킹 sriovNetSupport 속성 세트가 있는지 확인할 수 있습니다.

- [describe-instance-attribute\(AWS CLI\)](#)

```
aws ec2 describe-instance-attribute --instance-id instance_id --attribute sriovNetSupport
```

- [Get-EC2InstanceAttribute\(Windows PowerShell용 AWS 도구\)](#)

```
Get-EC2InstanceAttribute -InstanceId instance_id -Attribute sriovNetSupport
```

속성이 설정되지 않으면 SriosNetSupport는 비어 있습니다. 속성이 설정된 경우 다음 예제 출력과 같이 값이 간단합니다.

```
"SriosNetSupport": {  
    "Value": "simple"  
},
```

이미지 속성(sriovNetSupport)

다음 명령 중 하나를 사용하여 AMI에 향상된 네트워킹 sriovNetSupport 속성 세트가 있는지 확인할 수 있습니다.

- [describe-images\(AWS CLI\)](#)

```
aws ec2 describe-images --image-id ami_id --query "Images[ ].SriovNetSupport"
```

- [Get-EC2Image](#) (Windows PowerShell용 AWS 도구)

```
(Get-EC2Image -ImageId ami_id).SriovNetSupport
```

속성이 설정되지 않으면 `SriovNetSupport`는 비어 있습니다. 속성이 설정된 경우 값은 간단합니다.

Windows에서 향상된 네트워킹 활성화

확장 네트워크를 설정하지 않은 상태로 인스턴스를 시작한 경우에는 인스턴스에 필요한 네트워크 어댑터 드라이버를 다운로드하여 설치한 다음 `sriovNetSupport` 인스턴스 속성을 설정하여 확장 네트워크를 활성화해야 합니다. 이 속성 또는 지원되는 인스턴스 유형만 사용할 수 있습니다. 자세한 내용은 [향상된 네트워킹 유형 \(p. 751\)](#) 단원을 참조하십시오.

Important

Windows AMI에서 intel 드라이버의 최신 버전을 보려면 [AWS Windows AMI 버전에 관한 세부 정보 \(p. 30\)](#)를 참조하십시오.

Warning

향상된 네트워킹 속성을 활성화한 다음에는 다시 비활성화할 수 없습니다.

향상된 네트워킹을 활성화하려면

1. 인스턴스 연결 후 로컬 관리자로 로그인합니다.
2. [Windows Server 2016 이상] 다음 EC2Launch PowerShell 스크립트를 실행하여 드라이버가 설치된 후의 인스턴스를 구성합니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 - Schedule
```

Important

초기화 인스턴스 EC2Launch 스크립트를 활성화하면 관리자 암호가 재설정됩니다. 관리자 암호 재설정은 구성 파일을 수정하여 비활성화할 수 있는데 초기화 작업에 대한 설정에서 이를 지정하면 됩니다. 암호 재설정을 비활성화하는 절차는 [초기화 작업 구성 \(p. 491\)](#) 단원을 참조하십시오.

3. 다음과 같이 인스턴스 상에서 드라이버를 설치합니다.

- a. 운영 체제에 맞는 intel 네트워크 어댑터 드라이버를 다운로드합니다:

- [Windows Server 2008 R2](#)
- [Windows Server 2012](#)
- [Windows Server 2012 R2](#)
- [Windows Server 2016](#)(Server 버전 1803 및 이전 버전을 대상*)
- [Windows Server 2019](#)(Server 버전 1809 및 이후 버전*을 대상)

*Server 버전 1803 및 이전 버전과 1809 및 이후 버전은 Intel 드라이버 및 소프트웨어 페이지에 구체적으로 설명되지 않았습니다.

- b. Download 폴더에서 `PROWinx64.exe` 파일을 찾습니다. 이 파일의 이름을 `PROWinx64.zip`으로 바꿉니다.

- c. PROWinx64.zip에서 컨텍스트(오른쪽 클릭) 메뉴를 열고 전체 압축 해제(Extract All)를 선택합니다. 경로를 지정하고 압축 해제(Extract)를 선택합니다.
- d. 명령 프롬프트 창을 열고, 압축을 푼 파일이 있는 폴더로 이동한 다음, pnputil 유ти리티를 사용하여 드라이버 스토어에 INF 파일을 추가하고 설치합니다.

Windows Server 2019

```
pnputil -i -a PROXGB\Winx64\NDIS68\vxn68x64.inf
```

Windows Server 2016

```
pnputil -i -a PROXGB\Winx64\NDIS65\vxn65x64.inf
```

Windows Server 2012 R2

```
pnputil -i -a PROXGB\Winx64\NDIS64\vxn64x64.inf
```

Windows Server 2012

```
pnputil -i -a PROXGB\Winx64\NDIS63\vxn63x64.inf
```

Windows Server 2008 R2

```
pnputil -a PROXGB\Winx64\NDIS62\vxn62x64.inf
```

4. 로컬 컴퓨터에서 Amazon EC2 콘솔 또는 [stop-instances](#) (AWS CLI), [Stop-EC2Instance](#) (Windows PowerShell용 AWS 도구) 명령 중 하나를 사용하여 인스턴스를 중지하십시오. 인스턴스를 AWS OpsWorks에서 관리할 경우 AWS OpsWorks 콘솔에서 인스턴스를 중지해야 인스턴스 상태가 동기화됩니다.
5. 사용자의 로컬 컴퓨터에서 다음 명령 중 하나를 사용하여 확장 네트워크 속성을 활성화합니다.

- [modify-instance-attribute](#)(AWS CLI)

```
aws ec2 modify-instance-attribute --instance-id instance_id --srivnet-support simple
```

- [Edit-EC2InstanceAttribute](#)(Windows PowerShell용 AWS 도구)

```
Edit-EC2InstanceAttribute -InstanceId instance_id -SriovNetSupport "simple"
```

6. (선택 사항) [사용자 지정 Windows AMI 생성 \(p. 84\)](#)의 설명에 따라 인스턴스에서 AMI를 생성합니다. 생성된 AMI는 인스턴스의 확장 네트워크 속성을 상속합니다. 따라서 이 AMI를 사용하여 기본적으로 향상된 네트워킹 기능이 활성화된 상태로 다른 인스턴스를 시작할 수 있습니다.
7. 로컬 컴퓨터를 사용하는 경우, Amazon EC2 콘솔을 사용하거나 다음 명령 중 하나를 사용하여 인스턴스를 시작하십시오. [start-instances](#)(AWS CLI), [Start-EC2Instance](#)(Windows PowerShell용 AWS 도구). 인스턴스를 AWS OpsWorks에서 관리할 경우 AWS OpsWorks 콘솔에서 인스턴스를 시작해야 인스턴스 상태가 동기화됩니다.

배치 그룹

새 EC2 인스턴스를 시작하면 EC2 서비스는 모든 인스턴스가 기본 하드웨어 전반에 분산되도록 하여 상호 관련 오류의 위험을 줄입니다. 그러나 배치 그룹을 사용하면 워크로드의 요구 사항을 충족하도록 하는 독립

적 인스턴스의 그룹의 배치에 영향을 줄 수 있습니다. 워크로드의 유형에 따라 다음 배치 전략 중 하나를 사용하여 배치 그룹을 생성할 수 있습니다.

- 클러스터 – 인스턴스를 가용 영역 안에 서로 근접하게 패킹합니다. 이 전략은 워크로드가 HPC 애플리케이션에서 일반적인 긴밀히 결합된 노드 간 통신에 필요한 낮은 지연 시간의 네트워크 성능을 달성할 수 있습니다.
- 파티션 – 인스턴스를 논리적 파티션에 분산해, 한 파티션에 있는 인스턴스 그룹이 다른 파티션의 인스턴스 그룹과 기본 하드웨어를 공유하지 않게 합니다. 이 전략은 일반적으로 Hadoop, Cassandra, Kafka 등 대규모의 분산 및 복제된 워크로드에 필요합니다.
- 분산 – 소규모의 인스턴스 그룹을 다른 기본 하드웨어로 분산하여 상호 관련 오류를 줄입니다.

배치 그룹 생성은 무료입니다.

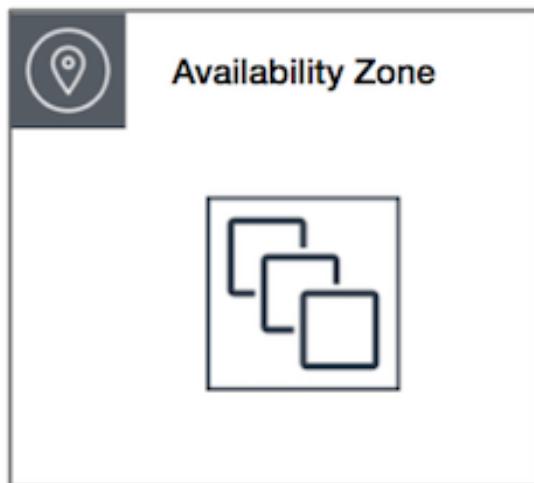
목차

- [클러스터 배치 그룹 \(p. 762\)](#)
- [파티션 배치 그룹 \(p. 763\)](#)
- [분산형 배치 그룹 \(p. 764\)](#)
- [배치 그룹 규칙 및 제한 사항 \(p. 764\)](#)
- [배치 그룹 생성 \(p. 765\)](#)
- [배치 그룹 태그 지정 \(p. 766\)](#)
- [배치 그룹으로 인스턴스 시작 \(p. 768\)](#)
- [배치 그룹의 인스턴스 설명 \(p. 769\)](#)
- [인스턴스의 배치 그룹 변경 \(p. 770\)](#)
- [배치 그룹 삭제 \(p. 771\)](#)

클러스터 배치 그룹

클러스터 배치 그룹은 단일 가용 영역 내에 있는 인스턴스의 논리적 그룹입니다. 클러스터 배치 그룹은 동일한 리전의 피어링된 VPC에 걸쳐 적용될 수 있습니다. 동일한 클러스터 배치 그룹의 인스턴스는 TCP/IP 트래픽에 더 높은 흐름당 처리량 제한을 제공하며 네트워크의 동일한 높은 양방향 대역폭 세그먼트에 배치됩니다.

다음 이미지는 클러스터 배치 그룹에 배치되는 인스턴스를 보여줍니다.



클러스터 배치 그룹은 짧은 네트워크 지연 시간, 높은 네트워크 처리량 또는 둘 다의 이점을 활용할 수 있는 애플리케이션에 권장됩니다. 또한 대부분의 네트워크 트래픽이 그룹 내 인스턴스 간에 전송되는 경우에도 권장됩니다. 배치 그룹에 가장 짧은 지연 시간과 가장 높은 초당 패킷 네트워크 성능을 제공하려면 향상된 네트워킹을 지원하는 인스턴스 유형을 선택하십시오. 자세한 내용은 [향상된 네트워킹 \(p. 751\)](#)을 참조하십시오.

다음과 같은 방법으로 인스턴스를 시작하는 것이 좋습니다.

- 단일 시작 요청을 사용하여 배치 그룹에 필요한 수의 인스턴스를 시작합니다.
- 배치 그룹의 모든 인스턴스에 동일한 인스턴스 유형을 사용합니다.

나중에 배치 그룹에 인스턴스를 더 추가하거나 배치 그룹에서 두 가지 이상의 인스턴스 유형을 시작하려고 하면 용량 부족 오류가 발생할 가능성이 커집니다.

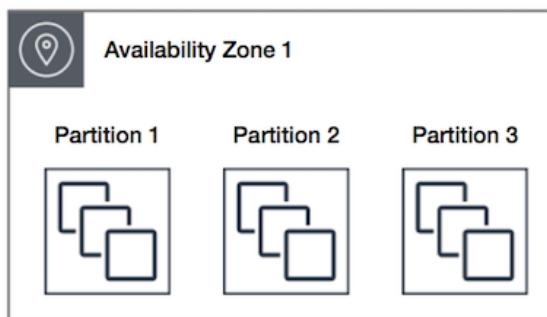
배치 그룹의 인스턴스를 중지한 후 다시 시작하면 인스턴스가 계속 배치 그룹에서 실행됩니다. 그러나 인스턴스에 대해 용량이 부족한 경우 시작에 실패합니다.

이미 인스턴스를 실행한 배치 그룹의 인스턴스를 시작할 때 용량 오류가 발생하는 경우, 배치 그룹의 모든 인스턴스를 중지하고 시작한 후 다시 실행해 보십시오. 인스턴스를 시작하면 요청한 모든 인스턴스를 수용할 용량이 있는 하드웨어로 인스턴스가 마이그레이션될 수 있습니다.

파티션 배치 그룹

파티션 배치 그룹은 애플리케이션에 대한 상관 관계가 있는 하드웨어 장애 가능성 줄이는데 도움이 됩니다. 파티션 배치 그룹을 사용할 때 Amazon EC2는 각 그룹을 파티션이라고 하는 논리 세그먼트로 나눕니다. Amazon EC2는 배치 그룹 내 각 파티션에 자체 랙 세트가 있는지 확인합니다. 각 랙은 자체 네트워크 및 전원이 있습니다. 배치 그룹 내 두 파티션이 동일한 랙을 공유하지 않으므로 애플리케이션 내 하드웨어 장애의 영향을 격리시킬 수 있습니다.

다음 이미지는 단일 가용 영역에 있는 파티션 배치 그룹을 시각적으로 간단하게 표현한 것입니다. 여기서는 세 개의 파티션이 파티션 1, 파티션 2 및 파티션 3이 있는 파티션 배치 그룹에 배치된 인스턴스를 보여줍니다. 각 파티션은 여러 인스턴스로 구성됩니다. 각 파티션에 있는 인스턴스는 다른 파티션에 있는 인스턴스와 랙을 공유하지 않기 때문에 단일 하드웨어 장애의 영향을 관련 파티션으로만 국한할 수 있습니다.



파티션 배치 그룹은 HDFS, HBase, Cassandra 같은 대규모 분산 및 복제 워크로드를 별개의 랙으로 분산해 배포하는 데 사용될 수 있습니다. 인스턴스를 파티션 배치 그룹으로 시작하면 Amazon EC2는 사용자가 지정한 수의 파티션에 인스턴스를 균일하게 배포합니다. 인스턴스를 특정 파티션으로 시작하면 인스턴스가 배치되는 위치에 대한 제어를 강화할 수도 있습니다.

파티션 배치 그룹은 동일한 리전의 여러 가용 영역에서 파티션을 가질 수 있습니다. 파티션 배치 그룹은 가용 영역당 파티션을 최대 7개까지 가질 수 있습니다. 파티션 배치 그룹에서 실행할 수 있는 인스턴스 숫자는 계정 제한의 적용을 받습니다.

또한 파티션 배치 그룹은 파티션 확인 기능도 제공합니다. 어떤 인스턴스가 어떤 파티션에 있는지 확인할 수 있습니다. 이 정보를 HDFS, HBase, Cassandra와 같은 토플로지 인식 애플리케이션과 공유할 수 있습니다.

이러한 애플리케이션은 이 정보를 이용하여 데이터 가용성 및 내구성을 높이기 위한 데이터 복제 결정을 지능적으로 수립합니다.

하나의 파티션 배치 그룹에서 하나의 인스턴스를 시작하고 요청을 이행하기 위한 고유의 하드웨어가 충분하지 않은 경우에는 요청이 실패합니다. Amazon EC2에서는 시간이 지나면서 더 많은 고유 하드웨어를 사용할 수 있기 때문에 나중에 다시 요청을 시도할 수 있습니다.

분산형 배치 그룹

분산형 배치 그룹은 각각 고유한 랙에 배치된 인스턴스 그룹이며 랙마다 자체 네트워크 및 전원이 있습니다.

다음 이미지는 분산형 배치 그룹에 배치되는 단일 가용 영역에 있는 인스턴스 7개를 보여줍니다. 7개의 인스턴스는 7개의 서로 다른 랙에 배치됩니다.



서로 떨어져 있어야 하는 중요 인스턴스의 수가 적은 애플리케이션에서는 분산형 배치 그룹이 권장됩니다. 분산형 배치 그룹에서 인스턴스를 시작하면 인스턴스가 동일한 랙을 공유할 때 장애가 동시에 발생할 수 있는 위험이 줄어듭니다. 분산형 배치 그룹은 별개의 랙에 대한 액세스를 제공하기 때문에 시간에 따라 인스턴스를 시작하거나 인스턴스 유형을 혼합할 때 적합합니다.

분산형 배치 그룹은 동일한 리전의 여러 가용 영역에 적용될 수 있습니다. 그룹당 가용 영역별로 최대 7개의 실행 중인 인스턴스를 가질 수 있습니다.

하나의 분산형 배치 그룹에서 하나의 인스턴스를 시작하고 요청을 이행하기 위한 고유의 하드웨어가 충분하지 않은 경우에는 요청이 실패합니다. Amazon EC2에서는 시간이 지나면서 더 많은 고유 하드웨어를 사용할 수 있기 때문에 나중에 다시 요청을 시도할 수 있습니다.

배치 그룹 규칙 및 제한 사항

일반 규칙 및 제한 사항

배치 그룹을 사용하기 전에 다음 규칙에 유의해야 합니다.

- 배치 그룹에 지정하는 이름은 해당 리전의 AWS 계정 내에서 고유해야 합니다.
- 여러 배치 그룹을 병합할 수는 없습니다.
- 인스턴스는 한 번에 하나의 배치 그룹에서 시작될 수 있습니다. 여러 배치 그룹으로 확장될 수 없습니다.
- [온디맨드 용량 예약 \(p. 356\)](#) 및 [영역 예약 인스턴스 \(p. 208\)](#)에서는 특정 가용 영역의 EC2 인스턴스에 용량 예약을 제공합니다. 용량 예약은 배치 그룹 내 인스턴스에서 사용이 가능합니다. 하지만 배치 그룹에 대한 용량을 명시적으로 예약할 수는 없습니다.
- 배치 그룹에서는 전용 호스트를 시작할 수 없습니다.

클러스터 배치 그룹 규칙 및 제한 사항

클러스터 배치 그룹에는 다음 규칙이 적용됩니다.

- 클러스터 배치 그룹의 인스턴스는 다음과 같은 지원되는 인스턴스 유형을 사용해야 합니다.
 - [버스트 가능한 성능 \(p. 129\)](#) 인스턴스(예: T2)를 제외한 [현재 세대 \(p. 115\)](#) 인스턴스 유형.
 - [이전 세대 \(p. 117\)](#) 인스턴스: C3, cc2.8xlarge, cr1.8xlarge, G2, hs1.8xlarge, I2, R3.
- 클러스터 배치 그룹은 여러 가용 영역을 포괄할 수 없습니다.

- 두 인스턴스의 속도가 느려지면 한 클러스터 배치 그룹에 있는 두 인스턴스 간에 트래픽의 최대 네트워크 처리 속도도 느려집니다. 많은 양을 처리해야 하는 애플리케이션의 경우, 요구 사항을 충족하는 네트워크 연결을 지원하는 인스턴스 유형을 선택하십시오.
- 향상된 네트워킹을 지원하는 인스턴스에는 다음 규칙이 적용됩니다.
 - 클러스터 배치 그룹 내부의 인스턴스는 단일 흐름 트래픽에 최대 10Gbps를 사용할 수 있습니다. 클러스터 배치 그룹 외부의 인스턴스는 단일 흐름 트래픽에 최대 5Gbps를 사용할 수 있습니다.
 - 동일한 리전 내에서 퍼블릭 IP 주소 공간이나 VPC 엔드포인트를 통해 Amazon S3 버킷과 주고받는 트래픽은 사용 가능한 인스턴스 집계 대역폭을 전부 사용할 수 있습니다.
- 하나의 클러스터 배치 그룹으로 여러 인스턴스 유형을 시작할 수 있습니다. 그러나 이렇게 하면 시작에 성공하는 데 필요한 용량이 원활하게 제공될 가능성이 낮아집니다. 클러스터 배치 그룹의 모든 인스턴스에 동일한 인스턴스 유형을 사용하는 것이 좋습니다.
- 인터넷으로 가는 네트워크 트래픽과 AWS Direct Connect 연결을 통해 온프레미스 리소스로 가는 네트워크 트래픽은 5Gbps로 제한됩니다.

파티션 배치 그룹 규칙 및 제한 사항

파티션 배치 그룹에는 다음 규칙이 적용됩니다.

- 파티션 배치 그룹은 사용 영역당 파티션을 최대 7개까지 지원합니다. 파티션 배치 그룹에서 실행할 수 있는 인스턴스 숫자는 계정 제한의 적용을 받습니다.
- 인스턴스를 파티션 배치 그룹으로 시작하면 Amazon EC2는 인스턴스를 모든 파티션에 균일하게 배포합니다. Amazon EC2는 모든 파티션에 대한 균일한 인스턴스 배포를 보장하지 않습니다.
- 전용 인스턴스가 있는 파티션 배치 그룹은 파티션을 최대 2개까지 가질 수 있습니다.

분산형 배치 그룹 규칙 및 제한 사항

분산형 배치 그룹에는 다음 규칙이 적용됩니다.

- 분산형 배치 그룹은 각 그룹의 사용 영역당 실행 인스턴스를 최대 7개까지 지원합니다. 예를 들어 사용 영역이 세 개인 리전에서는 그룹에서 총 21개의 실행 인스턴스를 실행할 수 있습니다(영역당 7개). 동일한 사용 영역과 동일한 분산 배치 그룹에서 여덟 번째 인스턴스를 시작하면 그 인스턴스는 시작되지 않습니다. 사용 영역에 인스턴스가 여덟 개 이상 있어야 하며, 분산 배치 그룹을 여러 개 사용할 것을 권장합니다. 여러 분산 배치 그룹을 사용해도 인스턴스가 그룹 간에 분산된다고 보장할 수는 없지만, 각 그룹에 분산함으로써 특정 종류의 실패로 인한 영향을 제한할 수는 있습니다.
- 분산형 배치 그룹은 전용 인스턴스에서 지원되지 않습니다.

배치 그룹 생성

다음 방법 중 하나를 사용하여 배치 그룹을 생성할 수 있습니다.

Note

배치 그룹을 생성할 때 명령줄 도구만 사용하여 배치 그룹에 태그를 지정할 수 있습니다.

새로운 콘솔

콘솔을 사용하여 배치 그룹을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 배치 그룹과 배치 그룹 생성을 차례로 선택합니다.
3. 그룹의 이름을 지정합니다.
4. 그룹의 배치 전략을 선택합니다. 파티션을 선택하는 경우 그룹 내 파티션 수를 선택합니다.

5. 그룹 생성을 선택합니다.

기존 콘솔

콘솔을 사용하여 배치 그룹을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 배치 그룹과 배치 그룹 생성을 차례로 선택합니다.
3. 그룹의 이름을 지정합니다.
4. 그룹의 배치 전략을 선택합니다. 파티션을 선택하는 경우 그룹 내 파티션 수를 지정합니다.
5. Create를 선택합니다.

AWS CLI

AWS CLI를 사용하여 배치 그룹을 생성하려면

`create-placement-group` 명령을 사용합니다. 다음 예제에서는 `cluster` 배치 전략을 사용하는 `my-cluster`라는 배치 그룹을 생성한 후 키가 `purpose`이고 값이 `production`인 태그를 적용합니다.

```
aws ec2 create-placement-group --group-name my-cluster --strategy cluster --tag-specifications 'ResourceType=placement-group,Tags={Key=purpose,Value=production}'
```

AWS CLI를 사용하여 파티션 배치 그룹을 생성하려면

`create-placement-group` 명령을 사용합니다. `partition` 값을 사용하여 `--strategy` 파라미터를 지정하고, 원하는 파티션 수를 사용하여 `--partition-count` 파라미터를 지정합니다. 이 예제에서 파티션 배치 그룹은 이름이 `HDFS-Group-A`이며 파티션 5개로 생성됩니다.

```
aws ec2 create-placement-group --group-name HDFS-Group-A --strategy partition --partition-count 5
```

PowerShell

Windows PowerShell용 AWS 도구를 사용하여 배치 그룹을 생성하려면

`New-EC2PlacementGroup` 명령을 사용합니다.

배치 그룹 태그 지정

기존 배치 그룹을 분류하고 관리할 수 있도록 사용자 지정 메타데이터로 태그를 지정할 수 있습니다. 태그 작성 방식에 대한 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

배치 그룹에 태그를 지정하면 배치 그룹으로 시작되는 인스턴스에는 자동으로 태그가 지정되지 않습니다. 배치 그룹으로 시작되는 인스턴스에 명시적으로 태그를 지정해야 합니다. 자세한 내용은 [인스턴스 시작 시 태그 추가 \(p. 1150\)](#) 단원을 참조하십시오.

새 콘솔과 명령줄 도구를 사용하여 태그를 보고, 추가하고, 삭제할 수 있습니다.

새 콘솔

기존 배치 그룹의 태그를 보거나 추가 또는 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 배치 그룹을 선택합니다.
3. 배치 그룹을 선택한 다음 작업, 태그 관리를 선택합니다.

4. 태그 관리 섹션에는 배치 그룹에 할당된 모든 태그가 표시됩니다. 태그를 추가하거나 제거하려면 다음을 수행합니다.
 - 태그를 추가하려면 태그 추가를 선택한 다음 태그 키와 값을 입력합니다. 배치 그룹당 최대 50개의 태그를 추가할 수 있습니다. 자세한 내용은 [태그 제한 \(p. 1147\)](#) 단원을 참조하십시오.
 - 태그를 삭제하려면 삭제할 태그 옆에 있는 제거를 선택합니다.
5. [Save changes]를 선택합니다.

AWS CLI

배치 그룹 태그를 보려면

`describe-tags` 명령을 사용하여 지정된 리소스에 대한 태그를 표시합니다. 다음 예제에서는 모든 배치 그룹의 태그를 설명합니다.

```
aws ec2 describe-tags \
--filters Name=resource-type,Values=placement-group
```

```
{
    "Tags": [
        {
            "Key": "Environment",
            "ResourceId": "pg-0123456789EXAMPLE",
            "ResourceType": "placement-group",
            "Value": "Production"
        },
        {
            "Key": "Environment",
            "ResourceId": "pg-9876543210EXAMPLE",
            "ResourceType": "placement-group",
            "Value": "Production"
        }
    ]
}
```

`describe-tags` 명령을 사용하면 해당 ID를 지정하여 배치 그룹의 태그를 볼 수도 있습니다. 다음 예제에서는 pg-0123456789EXAMPLE에 대한 태그를 설명합니다.

```
aws ec2 describe-tags \
--filters Name=resource-id,Values=pg-0123456789EXAMPLE
```

```
{
    "Tags": [
        {
            "Key": "Environment",
            "ResourceId": "pg-0123456789EXAMPLE",
            "ResourceType": "placement-group",
            "Value": "Production"
        }
    ]
}
```

배치 그룹을 설명하여 배치 그룹의 태그를 볼 수도 있습니다.

`describe-placement-groups` 명령을 사용하여 배치 그룹에 대해 지정된 모든 태그를 포함하는 지정된 배치 그룹의 구성을 표시합니다.

```
aws ec2 describe-placement-groups \
```

```
--group-name my-cluster
```

```
{  
    "PlacementGroups": [  
        {  
            "GroupName": "my-cluster",  
            "State": "available",  
            "Strategy": "cluster",  
            "GroupId": "pg-0123456789EXAMPLE",  
            "Tags": [  
                {  
                    "Key": "Environment",  
                    "Value": "Production"  
                }  
            ]  
        }  
    ]  
}
```

AWS CLI를 사용하여 기존 배치 그룹에 태그를 지정하려면

[create-tags](#) 명령을 사용해 기존 리소스에 태그를 지정할 수 있습니다. 다음 예제에서는 기존 배치 그룹에 Key=Cost-Center 및 Value=CC-123으로 태그가 지정됩니다.

```
aws ec2 create-tags \  
    --resources pg-0123456789EXAMPLE \  
    --tags Key=Cost-Center,Value=CC-123
```

AWS CLI를 사용하여 배치 그룹에서 태그를 삭제하려면

[delete-tags](#) 명령을 사용하여 기존 리소스에서 태그를 삭제할 수 있습니다. 예제를 보려면 AWS CLI Command Reference의 [예제](#)를 참조하십시오.

PowerShell

배치 그룹 태그를 보려면

[Get-EC2Tag](#) 명령을 사용합니다.

특정 배치 그룹의 태그를 설명하려면

[Get-EC2PlacementGroup](#) 명령을 사용합니다.

기존의 배치 그룹에 태그를 지정하려면

[New-EC2Tag](#) 명령을 사용합니다.

배치 그룹에서 태그를 삭제하려면

[Remove-EC2Tag](#) 명령을 사용합니다.

배치 그룹으로 인스턴스 시작

배치 그룹 규칙 및 제한 사항이 총족 (p. 764)되는 경우 다음 방법 중 하나를 사용하여 인스턴스를 배치 그룹으로 시작할 수 있습니다.

Console

콘솔을 사용하여 인스턴스를 배치 그룹으로 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 인스턴스를 선택합니다.
3. [Launch Instance]를 선택합니다. 마법사의 안내에 따라 다음 작업을 주의하여 수행하십시오.
 - 인스턴스 유형 선택 페이지에서 배치 그룹으로 실행할 인스턴스 유형을 선택합니다.
 - 인스턴스 세부 정보 구성 페이지에서 배치 그룹에 다음 필드를 적용할 수 있습니다.
 - 나중에 배치 그룹에 인스턴스를 추가하지 못할 수 있으므로 인스턴스 개수에 이 배치 그룹에서 필요한 총 인스턴스 개수를 입력합니다.
 - 배치 그룹에서 배치 그룹에 인스턴스를 추가합니다. 확인란을 선택합니다. 이 페이지에 배치 그룹이 표시되지 않으면 배치 그룹으로 시작할 수 있는 인스턴스 유형을 선택했는지 확인합니다. 그렇지 않으면 이 옵션을 사용할 수 없습니다.
 - 배치 그룹 이름에서 기존 배치 그룹에 인스턴스를 추가하거나 생성한 새 배치 그룹에 인스턴스를 추가하도록 선택할 수 있습니다.
 - 배치 그룹 전략에서 적합한 전략을 선택합니다. 파티션을 선택하면 Target partition(대상 파티션)에서 Auto distribution(자동 분산)을 선택하여 Amazon EC2에서 인스턴스를 그룹 내 모든 파티션에 골고루 배포하도록 하거나 인스턴스를 시작할 파티션을 지정합니다.

AWS CLI

AWS CLI를 사용하여 인스턴스를 배치 그룹으로 시작하려면

`run-instances` 명령을 사용하고 `--placement "GroupName = my-cluster"` 파라미터를 사용해 배치 그룹 이름을 지정합니다. 이 예제에서 배치 그룹의 이름은 `my-cluster`입니다.

```
aws ec2 run-instances --placement "GroupName = my-cluster"
```

AWS CLI를 사용하여 인스턴스를 파티션 배치 그룹의 특정 파티션으로 시작하려면

`run-instances` 명령어를 이용하고 `--placement "GroupName = HDFS-Group-A, PartitionNumber = 3"` 파라미터를 이용해 배치 그룹 이름과 파티션을 지정하십시오. 이 예제에서 파티션 배치 그룹은 이름이 `HDFS-Group-A`이며 파티션 숫자는 3개입니다.

```
aws ec2 run-instances --placement "GroupName = HDFS-Group-A, PartitionNumber = 3"
```

PowerShell

Windows PowerShell용 AWS 도구를 사용하여 인스턴스를 배치 그룹으로 시작하려면

`New-EC2Instance` 명령을 사용하고 `-Placement_GroupName` 파라미터를 사용해 배치 그룹 이름을 지정합니다.

배치 그룹의 인스턴스 설명

다음 방법 중 하나를 사용하여 인스턴스의 배치 정보를 볼 수 있습니다. AWS CLI를 사용하여 파티션 번호별로 파티션 배치 그룹을 필터링할 수도 있습니다.

Console

콘솔을 사용하여 인스턴스의 배치 그룹 및 파티션 번호를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 세부 정보 창에서 배치 그룹을 살펴봅니다. 인스턴스가 배치 그룹에 없다면, 필드는 빈칸으로 표시됩니다. 그렇지 않다면 배치 그룹 이름이 표시됩니다. 배치 그룹이 파티션 배치 그룹이면 인스턴스의 파티션 번호에 대한 파티션 번호를 검사합니다.

AWS CLI

AWS CLI를 사용하여 파티션 배치 그룹 내 인스턴스의 파티션 번호를 보려면

`describe-instances` 명령어를 이용하고 `--instance-id` 파라미터를 지정합니다.

```
aws ec2 describe-instances --instance-id i-0123a456700123456
```

응답에는 배치 정보가 포함되며, 이 정보는 인스턴스의 배치 그룹 이름과 파티션 숫자를 포함합니다.

```
"Placement": {  
    "AvailabilityZone": "us-east-1c",  
    "GroupName": "HDFS-Group-A",  
    "PartitionNumber": 3,  
    "Tenancy": "default"  
}
```

AWS CLI를 사용하여 특정 파티션 배치 그룹 및 파티션 번호로 인스턴스를 필터링하려면

`describe-instances` 명령어를 사용하고 `placement-group-name` 및 `placement-partition-number` 필터를 이용해 `--filters` 파라미터를 지정합니다. 이 예제에서 파티션 배치 그룹은 이름이 HDFS-Group-A이며 파티션 숫자는 7개입니다.

```
aws ec2 describe-instances --filters "Name = placement-group-name, Values = HDFS-Group-A" "Name = placement-partition-number, Values = 7"
```

응답은 지정된 배치 그룹에 있는 지정된 파티션의 인스턴스를 모두 나열합니다. 다음은 반환된 인스턴스에 대한 인스턴스 ID, 인스턴스 유형과 배치 정보만 표시하는 출력 예시입니다.

```
"Instances": [  
    {  
        "InstanceId": "i-0a1bc23d4567e8f90",  
        "InstanceType": "r4.large",  
    },  
  
    {"Placement": {  
        "AvailabilityZone": "us-east-1c",  
        "GroupName": "HDFS-Group-A",  
        "PartitionNumber": 7,  
        "Tenancy": "default"  
    }  
  
    {  
        "InstanceId": "i-0a9b876cd5d4ef321",  
        "InstanceType": "r4.large",  
    },  
  
    {"Placement": {  
        "AvailabilityZone": "us-east-1c",  
        "GroupName": "HDFS-Group-A",  
        "PartitionNumber": 7,  
        "Tenancy": "default"  
    }  
],
```

인스턴스의 배치 그룹 변경

다음과 같은 방법으로 인스턴스의 배치 그룹을 변경할 수 있습니다.

- 기존 인스턴스를 배치 그룹으로 이동
- 한 배치 그룹에서 다른 배치 그룹으로 인스턴스 이동
- 배치 그룹에서 인스턴스 제거

인스턴스를 이동하거나 제거하기 전에 인스턴스가 stopped 상태여야 합니다. AWS CLI 또는 AWS SDK를 사용하여 인스턴스를 이동하거나 제거할 수 있습니다.

AWS CLI

AWS CLI를 사용하여 인스턴스를 배치 그룹으로 이동하려면

1. `stop-instances` 명령을 사용하여 인스턴스를 중지합니다.
2. `modify-instance-placement` 명령을 사용하고 인스턴스가 이동할 배치 그룹의 이름을 지정합니다.

```
aws ec2 modify-instance-placement --instance-id i-0123a456700123456 --group-name MySpreadGroup
```

3. `start-instances` 명령을 사용하여 인스턴스를 시작합니다.

PowerShell

Windows PowerShell용 AWS 도구를 사용하여 인스턴스를 배치 그룹으로 이동하려면

1. `Stop-EC2Instance` 명령을 사용하여 인스턴스를 중지합니다.
2. `Edit-EC2InstancePlacement` 명령을 사용하고 인스턴스가 이동할 배치 그룹의 이름을 지정합니다.
3. `Start-EC2Instance` 명령을 사용하여 인스턴스를 중지합니다.

AWS CLI

AWS CLI를 사용하여 배치 그룹에서 인스턴스를 제거하려면

1. `stop-instances` 명령을 사용하여 인스턴스를 중지합니다.
2. `modify-instance-placement` 명령을 사용하고 배치 그룹 이름에 대해 빈 문자열을 지정합니다.

```
aws ec2 modify-instance-placement --instance-id i-0123a456700123456 --group-name ""
```

3. `start-instances` 명령을 사용하여 인스턴스를 시작합니다.

PowerShell

Windows PowerShell용 AWS 도구를 사용하여 배치 그룹에서 인스턴스를 제거하려면

1. `Stop-EC2Instance` 명령을 사용하여 인스턴스를 중지합니다.
2. `Edit-EC2InstancePlacement` 명령을 사용하고 배치 그룹 이름에 대해 빈 문자열을 지정합니다.
3. `Start-EC2Instance` 명령을 사용하여 인스턴스를 중지합니다.

배치 그룹 삭제

대체해야 하거나 더 이상 필요하지 않은 배치 그룹을 삭제할 수 있습니다. 다음 방법 중 하나를 사용하여 배치 그룹을 삭제할 수 있습니다.

Important

배치 그룹을 삭제하려면 배치 그룹에 인스턴스가 없어야 합니다. 배치 그룹으로 시작한 모든 인스턴스를 종료 (p. 455)하거나, 인스턴스를 다른 배치 그룹으로 이동 (p. 771)하거나, 인스턴스를 배치 그룹에서 제거 (p. 771)해야 합니다. 인스턴스 화면에서 인스턴스를 선택하고 세부 정보 창에서 배치 그룹의 값을 확인하면 인스턴스를 종료 또는 이동하기 전에 인스턴스가 해당 배치 그룹에 있는지 확인할 수 있습니다.

새로운 콘솔

콘솔을 사용하여 배치 그룹을 삭제하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 배치 그룹을 선택합니다.
3. 배치 그룹을 선택하고 삭제를 선택합니다.
4. 확인 메시지가 나타나면 **Delete**를 입력한 다음 삭제를 선택합니다.

기존 콘솔

콘솔을 사용하여 배치 그룹을 삭제하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 배치 그룹을 선택합니다.
3. 배치 그룹을 선택하고 배치 그룹 삭제를 선택합니다.
4. 확인 메시지가 나타나면 삭제를 선택합니다.

AWS CLI

AWS CLI를 사용하여 배치 그룹을 삭제하려면 다음을 수행합니다.

`delete-placement-group` 명령을 사용하고 배치 그룹 이름을 지정하여 배치 그룹을 삭제합니다. 이 예제에서 배치 그룹의 이름은 `my-cluster`입니다.

```
aws ec2 delete-placement-group --group-name my-cluster
```

PowerShell

Windows PowerShell용 AWS 도구를 사용하여 배치 그룹을 삭제하려면 다음을 수행합니다.

`Remove-EC2PlacementGroup` 명령을 사용하여 배치 그룹을 삭제합니다.

EC2 인스턴스에 대한 네트워크 MTU(최대 전송 단위)

네트워크 연결의 MTU(최대 전송 단위)는 연결을 통해 전달할 수 있는 허용되는 최대 크기의 패킷 크기(바이트)입니다. 연결의 MTU가 클수록 하나의 패킷으로 전달할 수 있는 데이터의 양이 늘어납니다. 이더넷 패킷은 프레임 또는 전송 중인 실제 데이터와 이를 둘러싼 네트워크 오버헤드 정보로 구성됩니다.

이더넷 프레임은 여러 가지 형식으로 제공될 수 있으며, 가장 일반적인 형식은 표준 이더넷 v2 프레임 형식입니다. 대부분의 인터넷에서 지원되는 최대 이더넷 패킷 크기인 1500MTU를 지원합니다. 인스턴스의 지원되는 최대 MTU는 인스턴스 유형에 따라 다릅니다. 모든 Amazon EC2 인스턴스 유형은 1500MTU를 지원하며, 현재 다수의 인스턴스 크기가 9001MTU 또는 점보 프레임을 지원합니다.

인스턴스가 Wavelength Zone에서 실행되는 경우 최대 MTU 값은 1300입니다.

목차

- [점보 프레임\(9001 MTU\) \(p. 773\)](#)
- [경로 MTU 검색 \(p. 773\)](#)
- [두 호스트 간 경로 MTU 확인 \(p. 774\)](#)
- [Windows 인스턴스에서 MTU 확인 및 설정 \(p. 774\)](#)
- [문제 해결 \(p. 776\)](#)

점보 프레임(9001 MTU)

점보 프레임에서는 패킷당 페이로드 크기를 늘려 1500바이트 이상의 데이터가 허용됩니다. 그 결과, 패킷 오버헤드에 해당하지 않는 패킷의 비율이 늘어납니다. 같은 양의 사용 가능한 데이터를 보내더라도 더 적은 수의 패킷만 있으면 됩니다. 그러나 지정된 AWS 리전(EC2-Classic), 단일 VPC 또는 VPC 피어링 연결 외부에서는 1500MTU의 최대 경로를 경험할 수 있습니다. VPN 연결 및 인터넷 게이트웨이를 통해 전송되는 트래픽은 1500 MTU로 제한됩니다. 패킷이 1500바이트인 경우, 단편화되거나 IP 헤더에 `Don't Fragment` 플래그가 설정된 경우 삭제됩니다.

인터넷 트래픽이나 VPC를 벗어나는 트래픽에 점보 프레임을 사용할 때는 주의가 필요합니다. 중간 시스템에서 패킷이 단편화되면서 트래픽이 느려지기 때문입니다. VPC 내에서 점보 프레임을 사용하고 VPC 외부의 느린 트래픽에는 사용하지 않으려면 라우팅을 기준으로 MTU 크기를 구성하거나, MTU 크기와 라우팅을 달리하여 다수의 탄력적 네트워크 인터페이스를 사용할 수도 있습니다.

그러나 클러스터 배치 그룹 내부에 함께 배치된 인스턴스의 경우, 점보 프레임이 최고의 네트워크 처리 속도를 달성하는 데 도움을 주므로 사용이 권장됩니다. 자세한 내용은 [배치 그룹 \(p. 761\)](#) 단원을 참조하십시오.

AWS Direct Connect를 통한 VPC와 온프레미스 네트워크 간의 트래픽에 점보 프레임을 사용할 수 있습니다. 자세한 내용과 점보 프레임 기능 확인 방법은 AWS Direct Connect 사용 설명서의 [네트워크 MTU 설정](#)을 참조하십시오.

모든 [현재 세대 인스턴스 \(p. 120\)](#)는 점보 프레임을 지원합니다. 다음의 이전 세대 인스턴스는 점보 프레임을 지원합니다: C3, G2, I2, M3, 및 R3.

전송 게이트웨이에 대해 지원되는 MTU 크기에 대한 자세한 내용은 Amazon VPC 전송 게이트웨이의 [MTU](#)를 참조하십시오.

경로 MTU 검색

경로 MTU 검색(PMTUD)은 네트워크 경로의 최대 전송 단위(MTU)를 확인하는 데 사용됩니다. 경로 MTU는 원래 호스트와 수신 호스트 간의 최대 패킷 크기입니다. 호스트가 수신 호스트의 MTU 또는 경로를 따라 디바이스의 MTU보다 큰 패킷을 전송하는 경우 수신 호스트 또는 디바이스가 Destination Unreachable: Fragmentation Needed and Don't Fragment was Set(유형 3, 코드 4)과 같은 ICMP 메시지를 반환합니다. 이는 패킷을 전송할 수 있을 때까지 MTU를 조정하도록 원본 호스트에 지시합니다.

기본적으로 보안 그룹은 인바운드 ICMP 트래픽을 허용하지 않습니다. 그러나 보안 그룹은 상태가 저장되므로 보안 그룹 규칙에 관계없이 아웃바운드 요청에 대한 ICMP 응답은 인바운드 흐름이 허용될 수 있습니다. 따라서 인스턴스가 ICMP 메시지 응답을 받을 수 있도록 인바운드 ICMP 규칙을 명시적으로 추가할 필요가 없습니다. 네트워크 ACL에서 ICMP 규칙을 구성하는 방법에 대한 자세한 내용은 Amazon VPC 사용 설명서의 [경로 MTU 검색](#)을 참조하십시오.

Important

경로 MTU 검색에서는 일부 라우터에서 점보 프레임이 삭제되지 않도록 보장하지 않습니다. VPC의 인터넷 게이트웨이는 패킷을 최대 1,500바이트까지만 전송합니다. 인터넷 트래픽에는 1,500MTU 패킷이 권장됩니다.

두 호스트 간 경로 MTU 확인

mturoute.exe 명령(<http://www.elifulkerson.com/projects/mturoute.php>)에서 다운로드하고 설치할 수 있는 Amazon Linux 포함)을 사용하여 두 호스트 사이에 경로 MTU를 확인할 수 있습니다.

mturoute.exe를 사용하여 경로 MTU를 확인하려면

1. <http://www.elifulkerson.com/projects/mturoute.php>에서 mturoute.exe를 다운로드합니다.
2. 명령 프롬프트 창을 열고 mturoute.exe를 다운로드한 디렉터리로 변경합니다.
3. 다음 명령을 사용하여 EC2 인스턴스와 다른 호스트 간의 경로 MTU를 확인합니다. DNS 이름 또는 IP 주소를 대상으로 사용할 수 있습니다. 대상이 다른 EC2 인스턴스인 경우에는 보안 그룹에서 인바운드 UDP 트래픽을 허용하는지 확인합니다. 이 예제에서는 EC2 인스턴스와 www.elifulkerson.com 간의 경로 MTU를 확인합니다.

```
.\mturoute.exe www.elifulkerson.com
* ICMP Fragmentation is not permitted. *
* Speed optimization is enabled. *
* Maximum payload is 10000 bytes. *
+ ICMP payload of 1472 bytes succeeded.
- ICMP payload of 1473 bytes is too big.
Path MTU: 1500 bytes.
```

이 예제에서 경로 MTU는 1500입니다.

Windows 인스턴스에서 MTU 확인 및 설정

일부 드라이버는 점보 프레임을 사용하도록 구성되어 있는 반면, 표준 프레임 크기를 사용하도록 구성된 드라이버도 있습니다. VPC 내의 네트워크 트래픽에 점보 프레임을 사용하거나 인터넷 트래픽에 표준 프레임을 사용할 수 있습니다. 어떤 사용 사례든 인스턴스가 예상한대로 동작하는지 확인하는 것이 좋습니다.

인스턴스가 Wavelength Zone에서 실행되는 경우 최대 MTU 값은 1300입니다.

ENA 드라이버

드라이버 버전 1.5 이하

디바이스 관리자(Device Manager) 또는 Set-NetAdapterAdvancedProperty 명령을 사용하여 MTU 설정을 변경할 수 있습니다.

Get-NetAdapterAdvancedProperty 명령을 사용하여 현재 MTU 설정을 가져오려면 다음 명령을 사용합니다. 인터페이스 이름 MTU에 대한 항목을 확인합니다. 값 9001은 점보 프레임이 활성화되었음을 나타냅니다. 점보 프레임은 기본적으로 비활성화되어 있습니다.

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

다음과 같이 점보 프레임을 활성화합니다.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "MTU" -RegistryValue 9001
```

다음과 같이 점보 프레임을 비활성화합니다.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "MTU" -RegistryValue 1500
```

드라이버 버전 2.1.0 이상

디바이스 관리자(Device Manager) 또는 Set-NetAdapterAdvancedProperty 명령을 사용하여 MTU 설정을 변경할 수 있습니다.

Get-NetAdapterAdvancedProperty 명령을 사용하여 현재 MTU 설정을 가져오려면 다음 명령을 사용합니다. 인터페이스 이름 *JumboPacket에 대한 항목을 확인합니다. 9015 값은 점보 프레임이 활성화되어 있음을 나타냅니다. 점보 프레임은 기본적으로 비활성화되어 있습니다.

Get-NetAdapterAdvancedProperty를 실행하거나 와일드카드(별표)를 사용하여 해당하는 모든 이더넷 이름을 검색합니다.

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

다음 명령을 실행하고 쿼리할 이더넷 이름을 포함합니다.

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

다음과 같이 점보 프레임을 활성화합니다.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -  
RegistryValue 9015
```

다음과 같이 점보 프레임을 비활성화합니다.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -  
RegistryValue 1514
```

Intel SRIOV 82599 드라이버

디바이스 관리자(Device Manager) 또는 Set-NetAdapterAdvancedProperty 명령을 사용하여 MTU 설정을 변경할 수 있습니다.

Get-NetAdapterAdvancedProperty 명령을 사용하여 현재 MTU 설정을 가져오려면 다음 명령을 사용합니다. 인터페이스 이름 *JumboPacket에 대한 항목을 확인합니다. 값 9014는 점보 프레임이 활성화되었음을 나타냅니다. (MTU 크기에는 헤더와 페이로드가 포함되어 있습니다.) 점보 프레임은 기본적으로 비활성화되어 있습니다.

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

다음과 같이 점보 프레임을 활성화합니다.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -  
RegistryValue 9014
```

다음과 같이 점보 프레임을 비활성화합니다.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -  
RegistryValue 1514
```

AWS PV 드라이버

디바이스 관리자(Device Manager)를 사용하여 MTU 설정을 변경할 수 없지만 netsh 명령을 사용하여 변경할 수 있습니다.

다음 명령을 사용하여 현재 MTU 설정을 가져옵니다. 인터페이스 이름은 다를 수 있습니다. 출력에서 "Ethernet," "Ethernet 2" 또는 "Local Area Connection"이라는 이름의 항목을 찾습니다. 점보 프레임을 활성화 또는 비활성화하려면 인터페이스 이름이 필요합니다. 값 9001은 점보 프레임이 활성화되었음을 나타냅니다.

```
netsh interface ipv4 show subinterface
```

다음과 같이 점보 프레임을 활성화합니다.

```
netsh interface ipv4 set subinterface "Ethernet" mtu=9001
```

다음과 같이 점보 프레임을 비활성화합니다.

```
netsh interface ipv4 set subinterface "Ethernet" mtu=1500
```

문제 해결

점보 프레임을 사용할 때 EC2 인스턴스와 Amazon Redshift 클러스터 사이에 연결 문제가 발생할 경우 Amazon Redshift Cluster Management Guide의 [쿼리가 반응이 없는 것으로 나타남](#)을 참조하십시오.

가상 사설 클라우드

Amazon Virtual Private Cloud(Amazon VPC)를 사용하면 Virtual Private Cloud(VPC)로 알려져 있는 AWS 클라우드에서 논리적으로 독립된 고유 영역에 가상 네트워크를 정의할 수 있습니다. 인스턴스와 같은 Amazon EC2 리소스를 VPC의 서브넷으로 실행할 수 있습니다. VPC는 고객의 자체 데이터 센터에서 운영하는 기존 네트워크와 매우 유사하지만 AWS의 확장 가능한 인프라를 사용한다는 이점을 제공합니다. 해당 IP 주소 범위를 선택하고, 서브넷을 만든 후 라우팅 테이블, 네트워크 게이트웨이 및 보안 설정을 구성하여 VPC를 구성할 수 있습니다. VPC의 인스턴스를 인터넷 또는 자체 데이터 센터에 연결합니다.

AWS 계정이 생성되면 각 리전에서 기본 VPC가 생성됩니다. 기본 VPC는 이미 구성되어 즉시 사용할 수 있는 VPC입니다. 기본 VPC로 인스턴스를 즉시 시작할 수 있습니다. 또는 기본이 아닌 VPC를 직접 생성할 수 있으며 필요에 따라 구성할 수 있습니다.

2013년 12월 4일 이전에 AWS 계정을 생성한 경우 일부 리전에서 EC2-Classic 플랫폼 지원을 받을 수 있습니다. 2013년 12월 4일 이후에 AWS 계정을 생성한 경우 EC2-Classic 지원이 되지 않으므로 VPC에서 리소스를 시작해야 합니다. 자세한 내용은 [EC2-Classic \(p. 802\)](#) 단원을 참조하십시오.

Amazon VPC 설명서

Amazon VPC에 대한 자세한 내용은 다음 문서를 참조하십시오.

가이드	설명
Amazon VPC 사용 설명서	Amazon VPC의 주요 개념을 설명하고 기능 사용에 대한 지침을 제공합니다.
Amazon VPC Peering Guide	VPC 피어링 연결을 설명하고 기능 사용에 대한 지침을 제공합니다.
Amazon VPC 전송 게이트웨이	전송 게이트웨이에 대해 설명하고 이를 구성 및 사용하기 위한 지침을 제공합니다.

가이드	설명
AWS Site-to-Site VPN 사용 설명서	Site-to-Site VPN 연결에 대해 설명하고 이를 구성 및 사용하기 위한 지침을 제공합니다.

Windows Amazon 머신 이미지(AMI)용 포트 및 프로토콜

다음 표에는 Windows Amazon 머신 이미지의 워크로드에 따른 포트, 프로토콜 및 방향이 나와 있습니다.

목차

- [AllJoyn 라우터 \(p. 777\)](#)
- [디바이스로 캐스팅 \(p. 778\)](#)
- [코어 네트워킹 \(p. 780\)](#)
- [배달 최적화 \(p. 793\)](#)
- [진단 추적 \(p. 794\)](#)
- [DIAL 프로토콜 서버 \(p. 794\)](#)
- [DFS\(분산 파일 시스템\) 관리 \(p. 794\)](#)
- [파일 및 프린터 공유 \(p. 795\)](#)
- [파일 서버 원격 관리 \(p. 798\)](#)
- [ICMP v4 모두 \(p. 798\)](#)
- [멀티캐스트 \(p. 798\)](#)
- [원격 데스크톱 \(p. 799\)](#)
- [Windows 장치 관리 \(p. 801\)](#)
- [Windows 방화벽 원격 관리 \(p. 801\)](#)
- [Windows 원격 관리 \(p. 802\)](#)

AllJoyn 라우터

OS	규칙	설명	포트	프로토콜	방향
Windows Server 2016 Windows Server 2019	AllJoyn 라우터 (TCP-In)	AllJoyn 라우터 트래픽의 인바운드 규칙입니다. [TCP]	로컬: 9955 원격: 모두	TCP	In
	AllJoyn 라우터 (TCP-Out)	AllJoyn 라우터 트래픽의 아웃바운드 규칙입니다. [TCP]	로컬: 모두 원격: 모두	TCP	Out
	AllJoyn 라우터 (UDP-In)	AllJoyn 라우터 트래픽의 인바운드 규칙입니다. [UDP]	로컬: 모두 원격: 모두	UDP	In
	AllJoyn 라우터 (UDP-Out)	AllJoyn 라우터 트래픽의 아웃바운드 규칙입니다. [UDP]	로컬: 모두	UDP	Out

OS	규칙	설명	포트	프로토콜	방향
		바운드 규칙입니다. [UDP]	원격: 모두		

디바이스로 캐스팅

OS	규칙	설명	포트	프로토콜	방향
Windows Server 2016 Windows Server 2019	디바이스로 캐스팅 기능 (qWave-TCP-In)	Quality Windows Audio Video Experience Service 사용을 허용하는 디바이스로 캐스팅 기능의 인바운드 규칙입니다. [TCP 2177]	로컬: 2177 원격: 모두	TCP	In
	디바이스로 캐스팅 기능 (qWave-TCP-Out)	Quality Windows Audio Video Experience Service 사용을 허용하는 디바이스로 캐스팅 기능의 아웃바운드 규칙입니다. [TCP 2177]	로컬: 모두 원격: 2177	TCP	Out
	디바이스로 캐스팅 기능 (qWave-UDP-In)	Quality Windows Audio Video Experience Service 사용을 허용하는 디바이스로 캐스팅 기능의 인바운드 규칙입니다. [UDP 2177]	로컬: 2177 원격: 모두	UDP	In
	디바이스로 캐스팅 기능 (qWave-UDP-Out)	Quality Windows Audio Video Experience Service 사용을 허용하는 디바이스로 캐스팅 기능의 아웃바운드 규칙입니다. [UDP 2177]	로컬: 모두 원격: 2177	UDP	Out
	디바이스로 캐스팅 SSDP 검색(UDP-In)	SSDP를 사용하여 디바이스로 캐스팅 대상의	로컬: Ply2Disc 원격: 모두	UDP	In

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
디바이스로 캐스팅

OS	규칙	설명	포트	프로토콜	방향
		검색을 허용하는 인바운드 규칙입니다.			
	디바이스로 캐스팅 스트리밍 서버(HTTP-Streaming-In)	HTTP를 사용하는 스트리밍을 허용하는 디바이스로 캐스팅 서버의 인바운드 규칙입니다. [TCP 10246]	로컬: 10246 원격: 모두	TCP	In
	디바이스로 캐스팅 스트리밍 서버(RTCP-Streaming-In)	RTSP 및 RTP를 사용하는 스트리밍을 허용하는 디바이스로 캐스팅 서버의 인바운드 규칙입니다. [UDP]	로컬: 모두 원격: 모두	UDP	In
	디바이스로 캐스팅 스트리밍 서버(RTP-Streaming-In)	RTSP 및 RTP를 사용하는 스트리밍을 허용하는 디바이스로 캐스팅 서버의 아웃바운드 규칙입니다. [UDP]	로컬: 모두 원격: 모두	UDP	Out
	디바이스로 캐스팅 스트리밍 서버(RTSP-Streaming-In)	RTSP 및 RTP를 사용하는 스트리밍을 허용하는 디바이스로 캐스팅 서버의 인바운드 규칙입니다. [TCP 23554, 23555, 23556]	로컬: 235, 542, 355, 523, 556 원격: 모두	TCP	In
	디바이스로 캐스팅 UPnP 이벤트(TCP-In)	디바이스로 캐스팅 대상에서 UPnP 이벤트 수신을 허용하는 인바운드 규칙입니다.	로컬: 2869 원격: 모두	TCP	In

코어 네트워킹

Windows Server 2012, 2012 R2, 2016, and 2019

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2012	대상에 연결할 수 없음 (ICMPv6-In)	패킷이 통과하는 임의의 노드에서 대상에 연결할 수 없음 오류 메시지를 보냅니다. 정체를 제외한 어떤 이유로든 패킷을 전달할 수 없습니다.	로컬: 68 원격: 67	ICMPv6	In
Windows Server 2012 R2					
Windows Server 2016					
Windows Server 2019	대상에 연결할 수 없음 조각화 필요(ICMPv4-In)	패킷이 통과하는 임의의 노드에서 대상에 연결할 수 없음 조각화 필요 오류 메시지를 보냅니다. 조각화가 필요한데 조각화 금지 비트가 설정되었기 때문에 패킷을 전달할 수 없습니다.	로컬: 68 원격: 67	ICMPv4	In
	코어 네트워킹 - DNS(UDP-Out)	DNS 요청을 허용하는 아웃바운드 규칙입니다. 이 규칙에 맞는 요청에 따른 DNS 응답이 소스 주소에 관계없이 허용됩니다. 이 동작을 느슨한 소스 매핑으로 분류합니다.	로컬: 모두 원격: 53	UDP	Out
	Dynamic Host Configuration Protocol(DHCP-In)	상태 저장 자동 구성 위해 DHCP(Dynamic Host Configuration Protocol) 메시지를 허용합니다.	로컬: 68 원격: 67	UDP	In
	Dynamic Host Configuration Protocol(DHCP-Out)	상태 저장 자동 구성 위해 DHCP(Dynamic Host	로컬: 68 원격: 67	UDP	Out

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
		Configuration Protocol) 메시지를 허용합니다.			
	IPv6용 Dynamic Host Configuration Protocol(DHCPV6 In)	상태 저장 및 상태 비저장 구성을 위해 DHCPV6(IPv6 용 Dynamic Host Configuration Protocol) 메시지를 허용합니다.	로컬: 546 원격: 547	UDP	In
	IPv6용 Dynamic Host Configuration Protocol(DHCPV6 Out)	상태 저장 및 상태 비저장 구성을 위해 DHCPV6(IPv6 용 Dynamic Host Configuration Protocol) 메시지를 허용합니다.	로컬: 546 원격: 547	UDP	Out
	코어 네트워킹 - 그룹 정책 (LSASS-Out)	그룹 정책 업데이트를 위해 원격 LSASS 트래픽을 허용하는 아웃바운드 규칙입니다.	로컬: 모두 원격: 모두	TCP	Out
	코어 네트워킹 - 그룹 정책 (NP-Out)	코어 네트워킹 - 그룹 정책 (NP-Out)	로컬: 모두 원격: 445	TCP	Out
	코어 네트워킹 - 그룹 정책 (TCP-Out)	그룹 정책 업데이트를 위해 원격 RPC 트래픽을 허용하는 아웃바운드 규칙입니다.	로컬: 모두 원격: 모두	TCP	Out
	Internet Group Management Protocol(IGMP-In)	멀티캐스트 그룹을 생성, 조인, 분리하기 위해 노드 간에 IGMP 메시지를 주고 받습니다.	로컬: 68 원격: 67	2	In

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	코어 네트워킹 - Internet Group Management Protocol(IGMP-Out)	멀티캐스트 그룹을 생성, 조인, 분리하기 위해 노드 간에 IGMP 메시지를 주고 받습니다.	로컬: 68 원격: 67	2	Out
	코어 네트워킹 - IPHTTPS(TCP-In)	HTTP 프록시 및 방화벽을 통한 연결을 제공하기 위해 IPHTTPS 터널링 기술을 허용하는 인바운드 TCP 규칙입니다.	로컬: IPHTTPS 원격: 모두	TCP	In
	코어 네트워킹 - IPHTTPS(TCP-Out)	HTTP 프록시 및 방화벽을 통한 연결을 제공하기 위해 IPHTTPS 터널링 기술을 허용하는 아웃바운드 TCP 규칙입니다.	로컬: 모두 원격: IPHTTPS	TCP	Out
	IPv6(IPv6-In)	ISATAP(Intra-Site Automatic Tunnel Addressing Protocol) 및 6to4 터널링 서비스에 IPv6 트래픽을 허용하는 데 필요한 인바운드 규칙입니다.	로컬: 모두 원격: 445	41	In
	IPv6(IPv6-Out)	ISATAP(Intra-Site Automatic Tunnel Addressing Protocol) 및 6to4 터널링 서비스에 IPv6 트래픽을 허용하는 데 필요한 아웃바운드 규칙입니다.	로컬: 모두 원격: 445	41	Out

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	멀티캐스트 리스너 완료 (ICMPv6-In)	멀티캐스트 리스너 완료 메시지는 서브넷에 특정 멀티캐스트 주소에 대해 남아 있는 구성 원이 없음을 로컬 라우터에 알립니다.	로컬: 68 원격: 67	ICMPv6	In
	멀티캐스트 리스너 완료 (ICMPv6-Out)	멀티캐스트 리스너 완료 메시지는 서브넷에 특정 멀티캐스트 주소에 대해 남아 있는 구성 원이 없음을 로컬 라우터에 알립니다.	로컬: 68 원격: 67	ICMPv6	Out
	멀티캐스트 리스너 쿼리 (ICMPv6-In)	IPv6 멀티캐스트 가능 라우터는 멀티캐스트 리스너 쿼리 메시지를 사용하여 멀티캐스트 그룹 등록에 대한 링크를 쿼리합니다.	로컬: 68 원격: 67	ICMPv6	In
	멀티캐스트 리스너 쿼리 (ICMPv6-Out)	IPv6 멀티캐스트 가능 라우터는 멀티캐스트 리스너 쿼리 메시지를 사용하여 멀티캐스트 그룹 등록에 대한 링크를 쿼리합니다.	로컬: 68 원격: 67	ICMPv6	Out
	멀티캐스트 리스너 보고 (ICMPv6-In)	멀티캐스트 리스너 보고 메시지는 수신 노드가 특정 멀티캐스트 주소에서 또는 멀티캐스트 리스너 쿼리에 대한 응답으로 멀티캐스트 트래픽 수신에 대한 관심을 즉시 보고하는 데 사용됩니다.	로컬: 68 원격: 67	ICMPv6	In

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	멀티캐스트 리스너 보고 (ICMPv6-Out)	멀티캐스트 리스너 보고 메시지는 수신 노드가 특정 멀티캐스트 주소에서 또는 멀티캐스트 리스너 쿼리에 대한 응답으로 멀티캐스트 트래픽 수신에 대한 관심을 즉시 보고하는 데 사용됩니다.	로컬: 68 원격: 67	ICMPv6	Out
	멀티캐스트 리스너 보고 v2(ICMPv6-In)	멀티캐스트 리스너 보고 v2 메시지는 수신 노드가 특정 멀티캐스트 주소에서 또는 멀티캐스트 리스너 쿼리에 대한 응답으로 멀티캐스트 트래픽 수신에 대한 관심을 즉시 보고하는 데 사용됩니다.	로컬: 68 원격: 67	ICMPv6	In
	멀티캐스트 리스너 보고 v2(ICMPv6-Out)	멀티캐스트 리스너 보고 v2 메시지는 수신 노드가 특정 멀티캐스트 주소에서 또는 멀티캐스트 리스너 쿼리에 대한 응답으로 멀티캐스트 트래픽 수신에 대한 관심을 즉시 보고하는 데 사용됩니다.	로컬: 68 원격: 67	ICMPv6	Out
	네트워크 환경 검색 알림 (ICMPv6-In)	노드에서 다른 노드에 링크 계층 주소 변경 내용을 알리거나 네트워크 환경 검색 요청에 대한 응답으로 네트워크 환경 검색 알림 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	네트워크 환경 검색 공고 (ICMPv6-Out)	노드에서 다른 노드에 링크 계층 주소 변경 내용을 알리거나 네트워크 환경 검색 요청에 대한 응답으로 네트워크 환경 검색 알림 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	네트워크 환경 검색 요청 (ICMPv6-In)	노드에서 연결되어 있는 다른 IPv6 노드의 링크 계층 주소를 검색하기 위해 네트워크 환경 검색 요청을 보냅니다.	로컬: 68 원격: 67	ICMPv6	In
	네트워크 환경 검색 요청 (ICMPv6-Out)	노드에서 연결되어 있는 다른 IPv6 노드의 링크 계층 주소를 검색하기 위해 네트워크 환경 검색 요청을 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	패킷이 너무 큼 (ICMPv6-In)	패킷이 통과하는 노드에서 다음 링크에 대해 패킷이 너무 커서 패킷을 전달할 수 없을 때 패킷이 너무 큼 오류 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In
	패킷이 너무 큼 (ICMPv6-Out)	패킷이 통과하는 노드에서 다음 링크에 대해 패킷이 너무 커서 패킷을 전달할 수 없을 때 패킷이 너무 큼 오류 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	파라미터 문제 (ICMPv6-In)	패킷이 잘못 생성되면 노드에서 파라미터 문제 오류 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	파라미터 문제 (ICMPv6-Out)	패킷이 잘못 생성되면 노드에서 파라미터 문제 오류 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	라우터 공고 (ICMPv6-In)	라우터는 상태 비저장 자동 구성을 위해 다른 노드로 라우터 공고 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In
	라우터 공고 (ICMPv6-Out)	라우터는 상태 비저장 자동 구성을 위해 다른 노드로 라우터 공고 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	라우터 요청 (ICMPv6-In)	노드에서 상태 비저장 자동 구성은 제공하는 라우터를 찾을 때 라우터 요청 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In
	라우터 요청 (ICMPv6-Out)	노드에서 상태 비저장 자동 구성은 제공하는 라우터를 찾을 때 라우터 요청 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	코어 네트워킹 - Teredo(UDP-In)	Teredo 엣지 통과를 허용하는 인바운드 UDP 규칙입니다. 이 기술은 IPv6/I Pv4 호스트가 IPv4 네트워크 주소 변환기 뒤에 있는 경우 유니캐스트 IPv6 트래픽에 주소 할당 및 자동 터널링을 제공합니다.	로컬: Teredo 원격: 모두	UDP	In

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	코어 네트워킹 - Teredo(UDP-Out)	Teredo 엣지 통과를 허용하는 아웃바운드 UDP 규칙입니다. 이 기술은 IPv6/IPv4 호스트가 IPv4 네트워크 주소 변환기 뒤에 있는 경우 유니캐스트 IPv6 트래픽에 주소 할당 및 자동 터널링을 제공합니다.	로컬: 모두 원격: 모두	UDP	Out
	시간 초과 (ICMPv6-In)	패킷이 통과하는 노드에서 출제한 값이 경로의 어떤 지점에서 0으로 감소되는 경우 시간초과 오류 메시지가 생성됩니다.	로컬: 68 원격: 67	ICMPv6	In
	시간 초과 (ICMPv6-Out)	패킷이 통과하는 노드에서 출제한 값이 경로의 어떤 지점에서 0으로 감소되는 경우 시간초과 오류 메시지가 생성됩니다.	로컬: 68 원격: 67	ICMPv6	Out

Windows Server 2008 R2 and SP2

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2008 R2	대상에 연결할 수 없음 (ICMPv6-In)	패킷이 통과하는 임의의 노드에서 대상에 연결할 수 없음 오류 메시지를 보냅니다. 정체를 제외한 어떤 이유로든 패킷을 전달할 수 없습니다.	로컬: 68 원격: 67	ICMPv6	In
Windows Server 2008 SP2	대상에 연결할 수 없음 조각화 필요(ICMPv4-In)	패킷이 통과하는 임의의 노드에서 대상에 연결할 수 없음	로컬: 68 원격: 67	ICMPv4	In

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
		조각화 필요 오류 메시지를 보냅니다. 조각화가 필요한데 조각화 금지 비트가 설정되었기 때문에 패킷을 전달할 수 없습니다.			
	Dynamic Host Configuration Protocol(DHCP-In)	상태 저장 자동 구성을 위해 DHCP(Dynamic Host Configuration Protocol) 메시지를 허용합니다.	로컬: 68 원격: 67	UDP	In
	Dynamic Host Configuration Protocol(DHCP-Out)	상태 저장 자동 구성을 위해 DHCP(Dynamic Host Configuration Protocol) 메시지를 허용합니다.	로컬: 68 원격: 67	UDP	Out
	IPv6용 Dynamic Host Configuration Protocol(DHCPV6 In)	상태 저장 및 상태 비저장 구성을 위해 DHCPV6(IPv6 용 Dynamic Host Configuration Protocol) 메시지를 허용합니다.	로컬: 546 원격: 547	UDP	In
	IPv6용 Dynamic Host Configuration Protocol(DHCPV6 Out)	상태 저장 및 상태 비저장 구성을 위해 DHCPV6(IPv6 용 Dynamic Host Configuration Protocol) 메시지를 허용합니다.	로컬: 546 원격: 547	UDP	Out

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	Internet Group Management Protocol(IGMP-In)	멀티캐스트 그룹을 생성, 조인, 분리하기 위해 노드 간에 IGMP 메시지를 주고 받습니다.	로컬: 68 원격: 67	2	In
	IPv6(IPv6-In)	ISATAP(Intra-Site Automatic Tunnel Addressing Protocol) 및 6to4 터널링 서비스에 IPv6 트래픽을 허용하는 데 필요한 인바운드 규칙입니다.	로컬: 모두 원격: 445	41	In
	IPv6(IPv6-Out)	ISATAP(Intra-Site Automatic Tunnel Addressing Protocol) 및 6to4 터널링 서비스에 IPv6 트래픽을 허용하는 데 필요한 아웃바운드 규칙입니다.	로컬: 모두 원격: 445	41	Out
	멀티캐스트 리스너 완료 (ICMPv6-In)	멀티캐스트 리스너 완료 메시지는 서브넷에 특정 멀티캐스트 주소에 대해 남아 있는 구성 원이 없음을 로컬 라우터에 알립니다.	로컬: 68 원격: 67	ICMPv6	In
	멀티캐스트 리스너 완료 (ICMPv6-Out)	멀티캐스트 리스너 완료 메시지는 서브넷에 특정 멀티캐스트 주소에 대해 남아 있는 구성 원이 없음을 로컬 라우터에 알립니다.	로컬: 68 원격: 67	ICMPv6	Out

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	멀티캐스트 리스너 쿼리 (ICMPv6-In)	IPv6 멀티캐스트 가능 라우터는 멀티캐스트 리스너 쿼리 메시지를 사용하여 멀티캐스트 그룹 등록에 대한 링크를 쿼리합니다.	로컬: 68 원격: 67	ICMPv6	In
	멀티캐스트 리스너 쿼리 (ICMPv6-Out)	IPv6 멀티캐스트 가능 라우터는 멀티캐스트 리스너 쿼리 메시지를 사용하여 멀티캐스트 그룹 등록에 대한 링크를 쿼리합니다.	로컬: 68 원격: 67	ICMPv6	Out
	멀티캐스트 리스너 보고 (ICMPv6-In)	멀티캐스트 리스너 보고 메시지는 수신 노드가 특정 멀티캐스트 주소에서 또는 멀티캐스트 리스너 쿼리에 대한 응답으로 멀티캐스트 트래픽 수신에 대한 관심을 즉시 보고하는 데 사용됩니다.	로컬: 68 원격: 67	ICMPv6	In
	멀티캐스트 리스너 보고 (ICMPv6-Out)	멀티캐스트 리스너 보고 메시지는 수신 노드가 특정 멀티캐스트 주소에서 또는 멀티캐스트 리스너 쿼리에 대한 응답으로 멀티캐스트 트래픽 수신에 대한 관심을 즉시 보고하는 데 사용됩니다.	로컬: 68 원격: 67	ICMPv6	Out

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	멀티캐스트 리스너 보고 v2(ICMPv6-In)	멀티캐스트 리스너 보고서 v2 메시지는 수신 노드가 특정 멀티캐스트 주소에서 또는 멀티캐스트 리스너 쿼리에 대한 응답으로 멀티캐스트 트래픽 수신에 대한 관심을 즉시 보고하는 데 사용됩니다.	로컬: 68 원격: 67	ICMPv6	In
	멀티캐스트 리스너 보고 v2(ICMPv6-Out)	멀티캐스트 리스너 보고서 v2 메시지는 수신 노드가 특정 멀티캐스트 주소에서 또는 멀티캐스트 리스너 쿼리에 대한 응답으로 멀티캐스트 트래픽 수신에 대한 관심을 즉시 보고하는 데 사용됩니다.	로컬: 68 원격: 67	ICMPv6	Out
	네트워크 환경 검색 알림 (ICMPv6-In)	노드에서 다른 노드에 링크 계층 주소 변경 내용을 알리거나 네트워크 환경 검색 요청에 대한 응답으로 네트워크 환경 검색 알림 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In
	네트워크 환경 검색 공고 (ICMPv6-Out)	노드에서 다른 노드에 링크 계층 주소 변경 내용을 알리거나 네트워크 환경 검색 요청에 대한 응답으로 네트워크 환경 검색 알림 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
코어 네트워킹

OS	규칙	정의	포트	프로토콜	방향
	네트워크 환경 검색 요청 (ICMPv6-In)	노드에서 연결되어 있는 다른 IPv6 노드의 링크 계층 주소를 검색하기 위해 네트워크 환경 검색 요청을 보냅니다.	로컬: 68 원격: 67	ICMPv6	In
	네트워크 환경 검색 요청 (ICMPv6-Out)	노드에서 연결되어 있는 다른 IPv6 노드의 링크 계층 주소를 검색하기 위해 네트워크 환경 검색 요청을 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	패킷이 너무 큼 (ICMPv6-In)	패킷이 통과하는 노드에서 다음 링크에 대해 패킷이 너무 커서 패킷을 전달할 수 없을 때 패킷이 너무 큼 오류 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In
	패킷이 너무 큼 (ICMPv6-Out)	패킷이 통과하는 노드에서 다음 링크에 대해 패킷이 너무 커서 패킷을 전달할 수 없을 때 패킷이 너무 큼 오류 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	파라미터 문제 (ICMPv6-In)	패킷이 잘못 생성되면 노드에서 파라미터 문제 오류 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In
	파라미터 문제 (ICMPv6-Out)	패킷이 잘못 생성되면 노드에서 파라미터 문제 오류 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	라우터 공고 (ICMPv6-In)	라우터는 상태 비저장 자동 구성 을 위해 다른 노드로 라우터 공고 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
배달 최적화

OS	규칙	정의	포트	프로토콜	방향
	라우터 공고 (ICMPv6-Out)	라우터는 상태 비저장 자동 구성을 위해 다른 노드로 라우터 공고 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	라우터 요청 (ICMPv6-In)	노드에서 상태 비저장 자동 구성은 제공하는 라우터를 찾을 때 라우터 요청 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	In
	라우터 요청 (ICMPv6-Out)	노드에서 상태 비저장 자동 구성은 제공하는 라우터를 찾을 때 라우터 요청 메시지를 보냅니다.	로컬: 68 원격: 67	ICMPv6	Out
	시간 초과 (ICMPv6-In)	패킷이 통과하는 노드에서 흡제한 값이 경로의 어떤 지점에서 0으로 감소되는 경우 시간 초과 오류 메시지가 생성됩니다.	로컬: 68 원격: 67	ICMPv6	In
	시간 초과 (ICMPv6-Out)	패킷이 통과하는 노드에서 흡제한 값이 경로의 어떤 지점에서 0으로 감소되는 경우 시간 초과 오류 메시지가 생성됩니다.	로컬: 68 원격: 67	ICMPv6	Out

배달 최적화

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2019	배달 최적화-TCP-In	배달 최적화 기능이 원격 앤드 포인트에 연결하도록 허용하는 인바운드 규칙입니다.	로컬: 7680 원격: 모두	TCP	In

OS	규칙	정의	포트	프로토콜	방향
	배달 최적화- UDP-In	배달 최적화 기능이 원격 엔드 포인트에 연결하도록 허용하는 인바운드 규칙입니다.	로컬: 7680 원격: 모두	UDP	In

진단 추적

Windows Server 2019

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2019	연결된 사용자 환경 및 원격 측정	통합 원격 측정 클라이언트 아웃바운드 트래픽	로컬: 모두 원격: 443	TCP	Out

Windows Server 2016

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2016	연결된 사용자 환경 및 원격 측정	통합 원격 측정 클라이언트 아웃바운드 트래픽	로컬: 모두 원격: 모두	TCP	Out

DIAL 프로토콜 서버

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2016 Windows Server 2019	DIAL 프로토콜 서버(HTTP-In)	HTTP를 사용하는 앱의 원격 제어를 허용하는 DIAL 프로토콜 서버에 대한 인바운드 규칙입니다.	로컬: 10247 원격: 모두	TCP	In

DFS(분산 파일 시스템) 관리

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2008 R2	DFS 관리 (SMB-In)	SMB 트래픽이 파일 서비스 역	로컬: 445 원격: 모두	TCP	In

OS	규칙	정의	포트	프로토콜	방향
		할을 관리하도 록 허용하는 인 바운드 규칙입 니다.			
	DFS 관리(WMI-In)	WMI 트래픽이 파일 서비스 역 할을 관리하도 록 허용하는 인 바운드 규칙입 니다.	로컬: RPC 원격: 모두	TCP	In
	DFS 관리(DCOM-In)	DCOM 트래픽 이 파일 서비스 역할을 관리하 도록 허용하는 인바운드 규칙 입니다.	로컬: 135 원격: 모두	TCP	In
	DFS 관리(TCP-In)	TCP 트래픽이 파일 서비스 역 할을 관리하도 록 허용하는 인 바운드 규칙입 니다.	로컬: RPC 원격: 모두	TCP	In

파일 및 프린터 공유

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2008 R2 Windows Server 2008 SP2	파일 및 프린터 공유(에코 요청 - ICMPv4-In)	다른 노드에 대한 ping 요청으로 에코 요청 메시지를 보냅니다.	로컬: 5355 원격: 모두	ICMPv4	In
Windows Server 2012 Windows Server 2012 R2	파일 및 프린터 공유(에코 요청 - ICMPv4-Out)	다른 노드에 대한 ping 요청으로 에코 요청 메시지를 보냅니다.	로컬: 5355 원격: 모두	ICMPv4	Out
	파일 및 프린터 공유(에코 요청 - ICMPv6-In)	다른 노드에 대한 ping 요청으로 에코 요청 메시지를 보냅니다.	로컬: 5355 원격: 모두	ICMPv6	In
	파일 및 프린터 공유(에코 요청 - ICMPv6-Out)	다른 노드에 대한 ping 요청으로 에코 요청 메시지를 보냅니다.	로컬: 5355 원격: 모두	ICMPv6	Out

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
파일 및 프린터 공유

OS	규칙	정의	포트	프로토콜	방향
	파일 및 프린터 공유(LLMNR-UDP-In)	파일 및 프린터 공유가 LLMNR(Link Local Multicast Name Resolution)을 허용하기 위한 인바운드 규칙입니다.	로컬: 5355 원격: 모두	UDP	In
	파일 및 프린터 공유(LLMNR-UDP-Out)	파일 및 프린터 공유가 LLMNR(Link Local Multicast Name Resolution)을 허용하기 위한 아웃바운드 규칙입니다.	로컬: 모두 원격: 5355	UDP	Out
	파일 및 프린터 공유(NB-Datagram-In)	파일 및 프린터 공유가 NetBIOS 데이터그램 전송 및 수신을 허용하기 위한 인바운드 규칙입니다.	로컬: 138 원격: 모두	UDP	In
	파일 및 프린터 공유(NB-Datagram-Out)	파일 및 프린터 공유가 NetBIOS 데이터그램 전송 및 수신을 허용하기 위한 아웃바운드 규칙입니다.	로컬: 모두 원격: 138	UDP	Out
	파일 및 프린터 공유(NB-Name-In)	파일 및 프린터 공유가 NetBIOS 이름 확인을 허용하기 위한 인바운드 규칙입니다.	로컬: 137 원격: 모두	UDP	In
	파일 및 프린터 공유(NB-Name-Out)	파일 및 프린터 공유가 NetBIOS 이름 확인을 허용하기 위한 아웃바운드 규칙입니다.	로컬: 모두 원격: 137	UDP	Out

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
파일 및 프린터 공유

OS	규칙	정의	포트	프로토콜	방향
	파일 및 프린터 공유(NB-Session-In)	파일 및 프린터 공유가 NetBIOS 세션 서비스 연결을 허용하기 위한 인바운드 규칙입니다.	로컬: 139 원격: 모두	TCP	In
	파일 및 프린터 공유(NB-Session-Out)	파일 및 프린터 공유가 NetBIOS 세션 서비스 연결을 허용하기 위한 아웃바운드 규칙입니다.	로컬: 모두 원격: 139	TCP	Out
	파일 및 프린터 공유(SMB-In)	파일 및 프린터 공유가 명명된 파일을 통해 서버 메시지 블록 전송 및 수신을 허용하기 위한 인바운드 규칙입니다.	로컬: 445 원격: 모두	TCP	In
	파일 및 프린터 공유(SMB-Out)	파일 및 프린터 공유가 명명된 파일을 통해 서버 메시지 블록 전송 및 수신을 허용하기 위한 아웃바운드 규칙입니다.	로컬: 모두 원격: 445	TCP	Out
	파일 및 프린터 공유(스플러 서비스 - RPC)	파일 및 프린터 공유에서 TCP/RPC를 통해 인쇄 스플러 서비스의 통신을 허용하는 인바운드 규칙입니다.	로컬: RPC 원격: 모두	TCP	In
	파일 및 프린터 공유(스플러 서비스 - RPC-EPMAP)	RPCSS 서비스가 스플러 서비스에 RPC/TCP 트래픽을 허용하기 위한 인바운드 규칙입니다.	로컬: RPC-EPMAP 원격: 모두	TCP	In

파일 서버 원격 관리

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2008 SP2 Windows Server 2012 Windows Server 2012 R2	파일 서버 원격 관리(DCOM-In)	DCOM 트래픽이 파일 서비스 역할을 관리하도록 허용하는 인바운드 규칙입니다.	로컬: 135 원격: 모두	TCP	In
	파일 서버 원격 관리(SMB-In)	SMB 트래픽이 파일 서비스 역할을 관리하도록 허용하는 인바운드 규칙입니다.	로컬: 445 원격: 모두	TCP	In
	WMI-In	WMI 트래픽이 파일 서비스 역할을 관리하도록 허용하는 인바운드 규칙입니다.	로컬: RPC 원격: 모두	TCP	In

ICMP v4 모두

OS	규칙	포트	프로토콜	방향
Windows Server 2012 Windows Server 2012 R2	모든 ICMP v4	로컬: 139 원격: 모두	ICMPv4	In

멀티캐스트

Windows Server 2019

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2019	mDNS(UDP-In)	mDNS 트래픽의 인바운드 규칙입니다.	로컬: 5353 원격: 모두	UDP	In
	mDNS(UDP-Out)	mDNS 트래픽의 아웃바운드 규칙입니다.	로컬: 모두 원격: 5353	UDP	Out

Windows Server 2016

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2016	mDNS(UDP-In)	mDNS 트래픽의 인바운드 규칙입니다.	로컬: mDNS 원격: 모두	UDP	In
	mDNS(UDP-Out)	mDNS 트래픽의 아웃바운드 규칙입니다.	로컬: 5353 원격: 모두	UDP	Out

원격 데스크톱

Windows Server 2012 R2, 2016, and 2019

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2012 R2 Windows Server 2016 Windows Server 2019	원격 데스크톱 - 새도우(TCP-In)	원격 데스크톱 서비스가 기존 원격 데스크톱 세션의 새도입을 허용하기 위한 인바운드 규칙입니다.	로컬: 모두 원격: 모두	TCP	In
	원격 데스크톱 - 사용자 모드 (TCP-In)	원격 데스크톱 서비스에서 RDP 트래픽을 허용하는 인바운드 규칙입니다.	로컬: 3389 원격: 모두	TCP	In
	원격 데스크톱 - 사용자 모드 (UDP-In)	원격 데스크톱 서비스에서 RDP 트래픽을 허용하는 인바운드 규칙입니다.	로컬: 3389 원격: 모두	UDP	In

Windows Server 2012

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2012	원격 데스크톱 - 사용자 모드 (TCP-In)	원격 데스크톱 서비스에서 RDP 트래픽을 허용하는 인바운드 규칙입니다.	로컬: 3389 원격: 모두	TCP	In
	원격 데스크톱 - 사용자 모드 (UDP-In)	원격 데스크톱 서비스에서 RDP 트래픽을 허용하는 인바운드 규칙입니다.	로컬: 3389 원격: 모두	UDP	In

Amazon Elastic Compute Cloud
 Windows Instances용 사용 설명서
 원격 데스크톱

OS	규칙	정의	포트	프로토콜	방향
		허용하는 인바운드 규칙입니다.			

Windows Server 2008 SP2

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2008 SP2	원격 데스크톱 - 샤도우(TCP-In)	원격 데스크톱 서비스가 기존 원격 데스크톱 세션의 샤도잉을 허용하기 위한 인바운드 규칙입니다.	로컬: 모두 원격: 모두	TCP	In
	원격 데스크톱 - 사용자 모드 (TCP-In)	원격 데스크톱 서비스에서 RDP 트래픽을 허용하는 인바운드 규칙입니다.	로컬: 3389 원격: 모두	TCP	In
	원격 데스크톱 - 사용자 모드 (UDP-In)	원격 데스크톱 서비스에서 RDP 트래픽을 허용하는 인바운드 규칙입니다.	로컬: 3389 원격: 모두	UDP	In

Windows Server 2008 R2

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2008 R2	RemoteFX(TCP-In)	원격 데스크톱 서비스에서 RDP 트래픽을 허용하는 인바운드 규칙입니다.	로컬: 3389 원격: 모두	TCP	In
	TCP-In	원격 데스크톱 서비스에서 RDP 트래픽을 허용하는 인바운드 규칙입니다.	로컬: 3389 원격: 모두	TCP	In

Windows 장치 관리

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2019	Windows 장치 관리 인증서 설치 관리자(TCP out)	Windows 장치 관리 인증서 설치 관리자의 아웃바운드 TCP 트래픽을 허용합니다.	로컬: 모두 원격: 모두	TCP	Out
	Windows 장치 관리 등록 서비스(TCP out)	Windows 장치 관리 등록 서비스의 아웃바운드 TCP 트래픽을 허용합니다.	로컬: 모두 원격: 모두	TCP	Out
	Windows 장치 관리 동기화 클라이언트(TCP out)	Windows 장치 관리 동기화 클라이언트의 아웃바운드 TCP 트래픽을 허용합니다.	로컬: 모두 원격: 모두	TCP	Out
	Windows 등록 WinRT(TCP Out)	Windows 등록 WinRT의 아웃바운드 TCP 트래픽을 허용합니다.	로컬: 모두 원격: 모두	TCP	Out

Windows 방화벽 원격 관리

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2008 SP2 Windows Server 2012 R2	Windows 방화벽 원격 관리 (RPC)	RPC/TCP를 통해 Windows 방화벽을 원격으로 관리하도록 허용하는 인바운드 규칙입니다.	로컬: RPC 원격: 모두	TCP	In
	Windows 방화벽 원격 관리 (RPC-EPMAP)	RPCSS 서비스가 Windows 방화벽에 RPC/TCP 트래픽을 허용하기 위한 인바운드 규칙입니다.	로컬: RPC-EPMAP 원격: 모두	TCP	In

Windows 원격 관리

OS	규칙	정의	포트	프로토콜	방향
Windows Server 2008 R2	Windows 원격 관리(HTTP-In)	WS-Management을 통한 Windows 원격 관리의 인바운드 규칙입니다.	로컬: 5985 원격: 모두	TCP	In
Windows Server 2008 SP2					
Windows Server 2012					
Windows Server 2012 R2					
Windows Server 2016					
Windows Server 2019					

Amazon EC2 보안 그룹에 대한 자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹](#)을 참조하십시오.

EC2-Classic

EC2-Classic 사용을 통해 다른 고객과 공유하는 단일 일반 네트워크에서 인스턴스가 실행됩니다. Amazon VPC 사용을 통해 AWS 계정에 속하도록 논리적으로 독립된 Virtual Private Cloud(VPC)에서 인스턴스가 실행됩니다.

Amazon EC2의 최초 출시 당시 EC2-Classic 플랫폼이 도입되었습니다. 2013년 12월 4일 이후에 AWS 계정을 생성한 경우 EC2-Classic을 지원하지 않으니 VPC에서 Amazon EC2 인스턴스를 시작해야 합니다.

계정이 EC2-Classic을 지원하지 않는 경우 기본 VPC가 생성됩니다. 기본적으로 인스턴스를 시작할 때 이를 기본 VPC에 시작합니다. 또는 기본 VPC가 아닌 VPC를 생성하고 인스턴스를 시작할 때 이를 지정할 수 있습니다.

지원되는 플랫폼 감지

Amazon EC2 콘솔은 사용자가 선택한 리전에서 인스턴스를 시작할 수 있는 플랫폼 및 해당 리전에 기본 VPC가 있는지 여부를 표시합니다.

탐색 모음에서 사용할 리전이 선택되어 있는지 확인합니다. Amazon EC2 콘솔 대시보드의 계정 속성에서 지원 가능한 플랫폼을 찾습니다.

EC2-Classic을 지원하는 계정

대시보드에서 계정 속성 아래에 다음과 같은 정보가 표시되어 계정에서 리전의 EC2-Classic 플랫폼과 VPC 모두 지원하지만 리전에 기본 VPC가 없음을 나타냅니다.

Account Attributes



Supported Platforms

EC2

VPC

[describe-account-attributes](#) 명령의 출력에는 supported-platforms 속성의 EC2 및 VPC 값이 포함됩니다.

```
aws ec2 describe-account-attributes --attribute-names supported-platforms
{
    "AccountAttributes": [
        {
            "AttributeName": "supported-platforms",
            "AttributeValues": [
                {
                    "AttributeValue": "EC2"
                },
                {
                    "AttributeValue": "VPC"
                }
            ]
        }
    ]
}
```

VPC를 필요로 하는 계정

대시보드에서 계정 속성 아래에 다음과 같은 정보가 표시되어 이 리전에서 인스턴스를 시작하려면 계정에 VPC가 필요하고, 이 리전에서 EC2-Classic 플랫폼을 지원하지 않으며, 리전에 식별자가 vpc-1a2b3c4d인 기본 VPC가 있음을 나타냅니다.

Account Attributes



Supported Platforms

VPC

Default VPC

vpc-1a2b3c4d

지정된 리전에 대한 [describe-account-attributes](#) 명령의 출력에는 supported-platforms 속성의 VPC 값만 포함됩니다.

```
aws ec2 describe-account-attributes --attribute-names supported-platforms --region us-east-2
{
    "AccountAttributes": [
        {
            "AttributeValues": [
                {
                    "AttributeValue": "VPC"
                }
            ],
            "AttributeName": "supported-platforms",
        }
    ]
}
```

EC2-Classic에서 사용 가능한 인스턴스 유형

최신 인스턴스 유형 대부분은 VPC를 필요로 합니다. 다음은 EC2-Classic에서 지원되는 인스턴스 유형입니다.

- 범용: M1, M3 및 T1
- 컴퓨팅 최적화: C1, C3 및 CC2
- 메모리 최적화: CR1, M2 및 R3
- 스토리지 최적화: D2, HS1 및 I2
- 액셀러레이티드 컴퓨팅: G2

계정에서 EC2-Classic을 지원하며 기본이 아닌 VPC를 생성하지 않은 경우 다음 중 하나를 수행하여 VPC를 필요로 하는 인스턴스를 시작할 수 있습니다.

- 기본이 아닌 VPC를 생성하고 요청에 서브넷 ID 또는 네트워크 인터페이스 ID를 지정하여 VPC 전용 인스턴스를 시작하십시오. 기본 VPC가 없고 AWS CLI, Amazon EC2 API 또는 AWS SDK를 사용하여 VPC 전용 인스턴스를 시작하는 경우 기본이 아닌 VPC를 생성해야 합니다.
- Amazon EC2 콘솔 대시보드를 사용하여 VPC 전용 인스턴스를 시작합니다. Amazon EC2 콘솔은 계정에서 기본이 아닌 VPC를 생성하고 첫 가용 영역의 서브넷에서 인스턴스를 시작합니다. 콘솔은 다음 속성이 있는 VPC를 생성합니다.
 - 가용 영역마다 하나의 서브넷에서 퍼블릭 IPv4 주소 속성이 `true`로 설정되므로 인스턴스가 퍼블릭 IPv4 주소를 받습니다. 자세한 내용은 Amazon VPC 사용 설명서의 [VPC에서 IP 주소 지정](#)을 참조하십시오.
 - 인터넷 게이트웨이 및 VPC의 트래픽을 인터넷 게이트웨이로 라우팅하는 기본 라우팅 테이블입니다. 이를 통해 VPC에서 시작하는 인스턴스가 인터넷을 통해 통신할 수 있습니다. 자세한 내용은 [Amazon VPC 사용 설명서](#)의 인터넷 게이트웨이 단원을 참조하십시오.
 - VPC의 기본 보안 그룹 및 각 서브넷과 연결된 기본 네트워크 ACL입니다. 자세한 정보는 Amazon VPC 사용 설명서의 [VPC의 보안 그룹](#) 단원을 참조하십시오.

EC2-Classic에 다른 리소스가 있는 경우 단계를 밟아 VPC로 마이그레이션할 수 있습니다. 자세한 내용은 [EC2-Classic에서 VPC로 마이그레이션 \(p. 820\)](#) 단원을 참조하십시오.

EC2-Classic 인스턴스와 VPC 간 차이점

다음 표에서는 EC2-Classic, 기본 VPC에서 시작되는 인스턴스, 기본이 아닌 VPC에서 시작되는 인스턴스의 차이점을 요약합니다.

특성	EC2-Classic	기본 VPC	기본이 아닌 VPC
퍼블릭 IPv4 주소(Amazon 퍼블릭 IP 주소 플랫폼에서 제공)	인스턴스에서 EC2-Classic 퍼블릭 IPv4 주소 폴로부터 퍼블릭 IPv4 주소를 수신합니다.	실행 시 따로 설정하거나 서브넷의 공인 IPv4 주소 속성을 변경하지 않은 경우 기본 서브넷 실행 인스턴스에 퍼블릭 IPv4 주소가 기본으로 할당됩니다.	시작 시 따로 지정하거나 서브넷의 퍼블릭 IPv4 주소 속성을 변경하지 않는 한 퍼블릭 IPv4 주소는 인스턴스에 기본으로 할당되지 않습니다.
프라이빗 IPv4 주소	인스턴스를 시작할 때마다 EC2-Classic 범위 내의 프라이빗 IPv4 주소가 할당됩니다.	인스턴스를 시작할 때마다 기본 VPC 주소 범위 내의 고정 프라이빗 IPv4 주소가 할당됩니다.	인스턴스를 시작할 때마다 VPC 주소 범위 내의 고정 프라이빗 IPv4 주소가 할당됩니다.

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EC2-Classic 인스턴스와 VPC 간 차이점

특성	EC2-Classic	기본 VPC	기본이 아닌 VPC
다중 프라이빗 IPv4 주소	인스턴스별로 하나의 프라이빗 IP 주소가 할당되며 다른 IP 주소는 지원되지 않습니다.	인스턴스에 다중 프라이빗 IPv4 주소를 할당할 수 있습니다.	인스턴스에 다중 프라이빗 IPv4 주소를 할당할 수 있습니다.
탄력적 IP 주소 (IPv4)	인스턴스를 종지하면 탄력적 IP는 인스턴스에서 연결해제됩니다.	인스턴스를 종지하면 탄력적 IP는 인스턴스와 연결된 상태를 유지합니다.	인스턴스를 종지하면 탄력적 IP는 인스턴스와 연결된 상태를 유지합니다.
탄력적 IP 주소 연결	탄력적 IP 주소를 인스턴스와 연결할 수 있습니다.	탄력적 IP 주소는 네트워크 인터페이스의 속성입니다. 인스턴스에 연결된 네트워크 인터페이스를 업데이트하여 탄력적 IP 주소를 인스턴스와 연결합니다.	탄력적 IP 주소는 네트워크 인터페이스의 속성입니다. 인스턴스에 연결된 네트워크 인터페이스를 업데이트하여 탄력적 IP 주소를 인스턴스와 연결합니다.
탄력적 IP 주소 재연결	탄력적 IP 주소가 이미 다른 인스턴스와 이미 연결된 경우 주소가 자동으로 새 인스턴스와 연결됩니다.	탄력적 IP 주소가 이미 다른 인스턴스와 이미 연결된 경우 주소가 자동으로 새 인스턴스와 연결됩니다.	탄력적 IP 주소가 다른 인스턴스와 이미 연결된 경우 재연결을 허용한 경우에만 성공합니다.
탄력적 IP 주소 태그 지정	탄력적 IP 주소에 태그를 적용할 수 없습니다.	탄력적 IP 주소에 태그를 적용할 수 있습니다.	탄력적 IP 주소에 태그를 적용할 수 있습니다.
DNS 호스트 이름	기본적으로 DNS 호스트 이름을 사용하도록 되어 있습니다.	기본적으로 DNS 호스트 이름을 사용하도록 되어 있습니다.	기본적으로 DNS 호스트 이름을 사용하지 않도록 되어 있습니다.
보안 그룹	보안 그룹에서 다른 AWS 계정에 속한 보안 그룹을 참조할 수 있습니다.	보안 그룹에서 VPC 또는 VPC 피어링 연결의 피어 VPC에서 보안 그룹을 참조할 수 있습니다.	보안 그룹에서는 사용자의 VPC 내 보안 그룹만 참조할 수 있습니다.
보안 그룹 연결	실행 중인 인스턴스의 보안 그룹은 변경할 수 없습니다. 할당된 보안 그룹의 규칙을 변경하시거나, 인스턴스를 새로운 것으로 교체하시면 됩니다(인스턴스에서 AMI 생성 -> 해당 AMI의 인스턴스를 원하는 보안 그룹에 연결하여 실행 -> 기존 인스턴스에 할당했던 EIP 주소 연결을 해제하고 해당 주소를 새 인스턴스에 할당 -> 기존 인스턴스 종료).	인스턴스당 최대 5개의 보안 그룹을 할당할 수 있습니다. 인스턴스 실행 시, 그리고 실행 중에도 보안 그룹을 할당할 수 있습니다.	인스턴스당 최대 5개의 보안 그룹을 할당할 수 있습니다. 인스턴스 실행 시, 그리고 실행 중에도 보안 그룹을 할당할 수 있습니다.
보안 그룹 규칙	인바운드 트래픽에만 규칙을 추가할 수 있습니다.	인바운드 및 아웃바운드 모두에 규칙을 지정 할 수 있습니다.	인바운드 및 아웃바운드 모두에 규칙을 지정 할 수 있습니다.
테넌시	인스턴스가 공유된 하드웨어에서 실행됩니다.	공유된 하드웨어나 단일 테넌트 하드웨어에서 인스턴스를 실행할 수 있습니다.	공유된 하드웨어나 단일 테넌트 하드웨어에서 인스턴스를 실행할 수 있습니다.

특성	EC2-Classic	기본 VPC	기본이 아닌 VPC
인터넷 액세스	인스턴스에서 인터넷에 액세스할 수 있습니다. 인스턴스가 퍼블릭 IP 주소를 자동으로 수신하고 AWS 네트워크 엣지를 통해 직접 인터넷에 액세스할 수 있습니다.	기본적으로 인스턴스가 인터넷에 액세스할 수 있습니다. 인스턴스에 퍼블릭 IP 주소가 기본으로 할당됩니다. 인터넷 게이트웨이가 기본 VPC에 연결되고 기본 서브넷에 인터넷 게이트웨이로 연결되는 경로가 있습니다.	기본적으로 인스턴스가 인터넷에 액세스할 수 없습니다. 인스턴스에 퍼블릭 IP 주소가 기본으로 할당되지 않습니다. 생성된 방법에 따라 VPC에 인터넷 게이트웨이가 있을 수 있습니다.
IPv6 주소 지정	IPv6 주소 지정은 지원되지 않습니다. 인스턴스에 IPv6 주소를 할당할 수 없습니다.	IPv6 CIDR 블록을 VPC에 연결하고 IPv6 주소를 VPC의 인스턴스에 할당할 수도 있습니다.	IPv6 CIDR 블록을 VPC에 연결하고 IPv6 주소를 VPC의 인스턴스에 할당할 수도 있습니다.

EC2-Classic의 보안 그룹

EC2-Classic을 사용하는 경우 EC2-Classic용으로 특별히 생성된 보안 그룹을 사용해야 합니다. EC2-Classic에서 인스턴스를 시작할 경우 인스턴스와 동일한 리전에서 보안 그룹을 지정해야 합니다. EC2-Classic에서 인스턴스를 시작할 경우 VPC용으로 생성된 보안 그룹을 지정할 수 없습니다.

EC2-Classic에서 인스턴스를 시작한 후에는 해당 보안 그룹을 변경할 수 없습니다. 그러나 보안 그룹에 규칙을 추가하거나 보안 그룹에서 규칙을 제거할 수 있으며, 잠시 후 보안 그룹과 연결된 모든 인스턴스에 해당 변경 내용이 자동으로 적용됩니다.

AWS 계정에서 자동으로 EC2-Classic 리전별 기본 보안 그룹을 보유합니다. 기본 보안 그룹을 삭제하려는 경우 다음과 같은 오류가 표시됩니다. Client.InvalidGroup.Reserved: The security group 'default' is reserved.

사용자 지정 보안 그룹을 생성할 수 있습니다. 보안 그룹 이름은 해당 리전의 사용자 계정 내에서 고유해야 합니다. EC2-Classic에서 사용할 보안 그룹을 생성하려면 VPC에 대해 VPC 없음을 선택하십시오.

기본 및 사용자 지정 보안 그룹에 인바운드 규칙을 추가할 수 있습니다. EC2-Classic 보안 그룹에 대한 아웃바운드 규칙을 변경할 수 없습니다. 보안 그룹을 생성할 때 동일한 리전에 있는 EC2-Classic에 대해 다른 보안 그룹을 소스 또는 대상으로 사용할 수 있습니다. 다른 AWS 계정에 대한 보안 그룹을 지정하려면 AWS 계정 ID를 접두사로 추가합니다. 예: 111122223333/sg-edcd9784.

EC2-Classic에서는 계정별로 각 리전에 최대 500개의 보안 그룹이 있을 수 있습니다. 보안 그룹당 최대 100개의 규칙을 추가할 수 있습니다. 인스턴스당 최대 800개의 보안 그룹 규칙을 지정할 수 있습니다. 이 값은 보안 그룹당 규칙과 인스턴스당 보안 그룹의 배수로 계산됩니다. 보안 그룹 규칙에서 다른 보안 그룹을 참조하는 경우 22자 이하의 보안 그룹 이름을 사용하는 것이 좋습니다.

IP 주소 지정 및 DNS

Amazon은 Amazon이 제공한 IPv4 DNS 호스트 이름을 IPv4 주소로 변환하는 DNS 서버를 제공합니다. EC2-Classic의 경우 Amazon DNS 서버는 172.16.0.23에 위치합니다.

EC2-Classic에서 사용자 지정 방화벽 구성을 생성하는 경우 Amazon DNS 서버의 주소에서 한 시적인 범위의 대상 포트와 함께 포트 53(DNS) —으로부터의 인바운드 트래픽을 허용하는 규칙을 방화벽에 생성해야 합니다. 그렇지 않으면 인스턴스에서 —내부 DNS 확인이 실패합니다. 방화벽에서 DNS 쿼리 응답이 자동 허용되지 않는 경우 Amazon DNS 서버의 IP 주소에서 전송되는 트래픽을 허용하도록 설정해야 합니다. Amazon DNS 서버의 IP 주소를 획득하려면 사용자 인스턴스에서 다음 명령을 사용합니다.

```
ipconfig /all | findstr /c:"DNS Servers"
```

탄력적 IP 주소

계정이 EC2-Classic을 지원하는 경우 EC2-Classic 플랫폼에 사용할 수 있는 탄력적 IP 주소 풀과 VPC에 사용할 수 있는 탄력적 IP 주소 풀이 하나씩 있습니다. VPC에 사용하도록 할당한 탄력적 IP 주소를 EC2-Classic의 인스턴스와 연결할 수 없으며 그 반대의 경우도 마찬가지입니다. 하지만 EC2-Classic 플랫폼에서 사용할 목적으로 할당한 탄력적 IP 주소를 VPC 플랫폼에서 사용하도록 마이그레이션할 수 있습니다. 탄력적 IP 주소는 다른 리전으로 마이그레이션할 수 없습니다.

콘솔을 사용하여 EC2-Classic에서 사용할 탄력적 IP 주소를 할당하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 탄력적 IP를 선택합니다.
3. Allocate new address를 선택합니다.
4. Classic을 선택한 후 할당을 선택합니다. 확인 화면을 닫습니다.

EC2-Classic에서 탄력적 IP 주소 마이그레이션

계정이 EC2-Classic을 지원하는 경우 EC2-Classic 플랫폼에서 사용할 목적으로 할당한 탄력적 IP 주소를 동일 리전 내 VPC로 마이그레이션할 수 있습니다. EC2-Classic에서 VPC로 리소스를 마이그레이션할 수 있는 이유도 바로 여기에 있습니다. 예를 들어, VPC에서 새로운 웹 서버를 실행한 다음 EC2-Classic의 웹 서버에서 사용한 탄력적 IP 주소를 새로운 VPC 웹 서버에도 똑같이 사용할 수 있습니다.

VPC로 탄력적 IP 주소를 마이그레이션한 이후 EC2-Classic에서 이를 사용할 수 없습니다. 하지만 필요한 경우 이를 EC2-Classic에 복원할 수 있습니다. 처음부터 VPC에서 사용할 목적으로 할당한 탄력적 IP 주소는 EC2-Classic으로 마이그레이션할 수 없습니다.

탄력적 IP 주소를 마이그레이션하려면 연결되어 있는 인스턴스가 없어야 합니다. 탄력적 IP 주소를 인스턴스에서 연결 해제하기 위한 자세한 내용은 [탄력적 IP 주소 연결 해제 \(p. 726\)](#) 단원을 참조하십시오.

EC2-Classic 탄력적 IP 주소는 계정에 가질 수 있는 만큼 마이그레이션할 수 있습니다. 하지만 탄력적 IP 주소를 마이그레이션하는 경우 VPC에 대한 탄력적 IP 주소 한도에서 차감됩니다. 따라서 한도를 초과하는 경우에는 탄력적 IP 주소를 마이그레이션할 수 없습니다. 마찬가지로 탄력적 IP 주소를 EC2-Classic로 복구할 때도 EC2-Classic의 탄력적 IP 주소의 최대 수에서 차감됩니다. 자세한 내용은 [탄력적 IP 주소 제한 \(p. 728\)](#) 단원을 참조하십시오.

계정에 할당되고 24시간이 지나지 않은 탄력적 IP 주소는 마이그레이션할 수 없습니다.

Amazon EC2 콘솔 또는 Amazon VPC 콘솔을 사용하여 EC2-Classic에서 탄력적 IP 주소를 마이그레이션할 수 있습니다. 이 옵션은 계정이 EC2-Classic을 지원하는 경우에만 사용 가능합니다.

Amazon EC2 콘솔을 사용하여 탄력적 IP 주소를 이동하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 탄력적 IP를 선택합니다.
3. 탄력적 IP 주소를 선택하고 작업, VPC 범위로 이동을 선택합니다.
4. 확인 대화 상자가 나타나면 탄력적 IP 이동을 선택합니다.

Amazon EC2 콘솔 또는 Amazon VPC 콘솔을 사용하여 탄력적 IP 주소를 EC2-Classic으로 복원할 수 있습니다.

Amazon EC2 콘솔을 사용하여 탄력적 IP 주소를 EC2-Classic으로 복원하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 탄력적 IP를 선택합니다.
3. 탄력적 IP 주소를 선택하고 작업, EC2 범위로 복원을 선택합니다.
4. 확인 대화 상자가 나타나면 복원을 선택합니다.

탄력적 IP 주소를 마이그레이션하거나 복구하는 데 명령을 사용한 경우에는 마이그레이션 프로세스가 몇 분 걸릴 수 있습니다. 이때는 [describe-moving-addresses](#) 명령을 사용하면 탄력적 IP 주소가 아직 마이그레이션 중인지, 혹은 마이그레이션이 완료되었는지 확인할 수 있습니다.

탄력적 IP 주소를 이동한 후 할당 ID 필드의 탄력적 IP 페이지에서 할당 ID를 확인할 수 있습니다.

탄력적 IP 주소를 이동할 때 5분이 지나도 완료되지 않으면 [Premium Support](#)에 문의하십시오.

명령줄을 사용하여 탄력적 IP 주소를 이동하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [move-address-to-vpc](#)(AWS CLI)
- [Move-EC2AddressToVpc](#)(Windows PowerShell용 AWS 도구)

명령줄을 사용하여 탄력적 IP 주소를 EC2-Classic으로 복원하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [restore-address-to-classic](#)(AWS CLI)
- [Restore-EC2AddressToClassic](#)(Windows PowerShell용 AWS 도구)

명령줄을 사용하여 주소 이동 상태를 설명하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-moving-addresses](#)(AWS CLI)
- [Get-EC2Address](#)(Windows PowerShell용 AWS 도구)

EC2-Classic과 VPC 간 리소스 공유 및 액세스

AWS 계정의 일부 리소스와 기능은 EC2-Classic 및 VPC 간에서 공유 또는 액세스될 수 있습니다(예: [ClassicLink](#)). 자세한 내용은 [ClassicLink \(p. 809\)](#) 단원을 참조하십시오.

EC2-Classic 지원 계정을 사용하는 경우 EC2-Classic에서 사용할 리소스를 설정했을 수 있습니다. EC2-Classic에서 VPC로 마이그레이션하기 위해서는 해당 리소스를 VPC에서 다시 만들어야 합니다. EC2-Classic에서 VPC로의 마이그레이션에 대한 자세한 내용은 [EC2-Classic에서 VPC로 마이그레이션 \(p. 820\)](#)을(를) 참조하십시오.

다음은 EC2-Classic와 VPC 간에 공유나 액세스가 가능한 리소스입니다.

리소스	참고
AMI	
번들 작업	
EBS 볼륨	
탄력적 IP 주소(IPv4)	탄력적 IP 주소는 EC2-Classic에서 VPC로 마이그레이션할 수 있습니다. 처음부터 EC2-VPC에서 사용할 목적으로 할당한 탄력적 IP 주소는 EC2-Classic으로 마이그레이션하지 못합니다. 자세한 내

리소스	참고
	용은 EC2-Classic에서 탄력적 IP 주소 마이그레이션 (p. 807) 단원을 참조하십시오.
인스턴스	<p>EC2-Classic 인스턴스는 퍼블릭 IPv4 주소를 할당 받은 VPC의 인스턴스와 통신할 수 있으며, 프라이빗 IPv4 주소를 가지고 있는 인스턴스와의 통신을 원하는 경우 ClassicLink를 사용할 수 있습니다.</p> <p>EC2-Classic에서 VCP로 인스턴스를 마이그레이션 할 수 없습니다. 그러나 EC2-Classic 내 인스턴스에서 VPC 내 인스턴스로는 애플리케이션을 마이그레이션할 수 있습니다. 자세한 내용은 EC2-Classic에서 VPC로 마이그레이션 (p. 820) 단원을 참조하십시오.</p>
키 페어	
로드 밸런서	<p>ClassicLink를 사용하는 경우 VPC 내 링크한 EC2-Classic 인스턴스에 로드 밸런서를 등록할 수 있습니다. 이때 VPC가 인스턴스처럼 동일한 가용 영역에 서브넷을 보유해야 합니다.</p> <p>EC2-Classic에서 VCP로 로드 밸런서를 마이그레이션 할 수 없습니다. VPC 내 EC2-Classic에 인스턴스를 등록할 수 없습니다.</p>
배치 그룹	
예약 인스턴스	예약 인스턴스에 대한 네트워크 플랫폼을 EC2-Classic에서 VPC로 변경할 수 있습니다. 자세한 내용은 예약 인스턴스 수정 (p. 229) 단원을 참조하십시오.
보안 그룹	<p>링크한 EC2-Classic 인스턴스는 ClassicLink을(를) 통해서 VPC 보안 그룹을 사용하여 VPC로 그리고 VPC로부터 트래픽을 제어할 수 있습니다. VPC 인스턴스에서 EC2-Classic 보안 그룹을 사용할 수 없습니다.</p> <p>EC2-Classic에서 VCP로 보안 그룹을 마이그레이션 할 수 없습니다. EC2-Classic의 보안 그룹에서 VPC의 보안 그룹으로 규칙을 복사할 수 있습니다. 자세한 내용은 보안 그룹 생성 (p. 912) 단원을 참조하십시오.</p>
스냅샷	

다음은 EC2-Classic과 VPC 간에 공유나 이동이 불가능한 리소스입니다.

- **스팟** 인스턴스

ClassicLink

ClassicLink를 사용하면 같은 리전 내에 있는 해당 계정의 VPC에 EC2-Classic 인스턴스를 연결할 수 있습니다. VPC 보안 그룹과 EC2-Classic 인스턴스를 연결하는 경우 이를 통해 EC2-Classic 인스턴스와 프라이빗

IPv4 주소를 사용하는 VPC 내 인스턴스와 통신할 수 있습니다. ClassicLink를 사용하면 이러한 플랫폼의 인스턴스 간 통신을 위해 퍼블릭 IPv4 주소 또는 탄력적 IP 주소를 사용할 필요가 없습니다.

ClassicLink는 EC2-Classic 플랫폼을 지원하는 계정을 갖는 모든 사용자에게 제공되며 모든 EC2-Classic 인스턴스에 사용할 수 있습니다. 리소스를 VPC로 마이그레이션한 방법에 대한 자세한 내용은 [EC2-Classic에서 VPC로 마이그레이션 \(p. 820\)](#) 단원을 참조하십시오.

ClassicLink 사용에 따르는 추가 요금은 없습니다. 데이터 전송 및 인스턴스 사용에 대한 표준 요금이 그대로 적용됩니다.

목차

- [ClassicLink 기본 사항 \(p. 810\)](#)
- [ClassicLink 제한 사항 \(p. 812\)](#)
- [ClassicLink 작업 \(p. 813\)](#)
- [ClassicLink에 대한 예제 IAM 정책 \(p. 816\)](#)
- [예: 3tier 웹 애플리케이션의 ClassicLink 보안 그룹 구성 \(p. 818\)](#)

ClassicLink 기본 사항

두 단계를 통해 EC2-Classic를 사용하여 ClassicLink 인스턴스를 VPC에 링크할 수 있습니다. 우선 VPC에서 ClassicLink를 활성화해야 합니다. 기본적으로는 격리 상태를 유지하기 위해 계정의 모든 VPC에서 ClassicLink가 비활성화됩니다. VPC에서 ClassicLink를 활성화하면 계정의 같은 리전에서 실행 중인 모든 EC2-Classic 인스턴스를 해당 VPC에 링크할 수 있습니다. 인스턴스를 링크할 때는 EC2-Classic 인스턴스에 연결할 보안 그룹을 VPC에서 선택합니다. 링크된 인스턴스는 VPC 보안 그룹에서 허용하는 경우 프라이빗 IP 주소를 사용하여 VPC의 인스턴스와 통신할 수 있습니다. EC2-Classic 인스턴스의 프라이빗 IP 주소는 VPC에 링크되어도 그대로 유지됩니다.

Note

인스턴스를 VPC에 링크하는 작업을 인스턴스 연결이라고도 합니다.

링크된 EC2-Classic 인스턴스는 VPC의 인스턴스와 통신할 수 있지만 VPC에 속하지 않습니다. 예를 들어 `DescribeInstances` API 요청 또는 Amazon EC2 콘솔의 인스턴스 화면을 사용하여 인스턴스를 나열하고 VPC로 필터링하면 VPC에 링크된 EC2-Classic 인스턴스는 결과로 반환되지 않습니다. 링크된 EC2-Classic 인스턴스를 확인하는 방법에 대한 자세한 내용은 [ClassicLink 가능 VPC 및 링크된 인스턴스 보기 \(p. 815\)](#) 단원을 참조하십시오.

기본적으로 퍼블릭 DNS 호스트 이름을 사용하여 연결된 EC2-Classic 인스턴스에서 VPC의 인스턴스를 처리하는 경우, 호스트 이름은 해당 인스턴스의 퍼블릭 IP 주소로 확인됩니다. 퍼블릭 DNS 호스트 이름을 사용하여 VPC의 인스턴스에서 연결된 EC2-Classic 인스턴스를 처리하는 경우에도 동일합니다. 퍼블릭 DNS 호스트 이름이 프라이빗 IP 주소가 되도록 하려면 VPC에 대해 ClassicLink DNS 지원을 활성화하면 됩니다. 자세한 내용은 [ClassicLink DNS 지원 활성화 \(p. 815\)](#) 단원을 참조하십시오.

인스턴스와 VPC 간에 ClassicLink 연결이 더 이상 필요하지 않은 경우 VPC에서 EC2-Classic 인스턴스의 링크를 해제할 수 있습니다. 이렇게 하면 VPC 보안 그룹이 EC2-Classic 인스턴스에서 분리됩니다. 링크된 EC2-Classic 인스턴스를 종지하면 자동으로 VPC와 링크가 해제됩니다. VPC에서 링크된 모든 EC2-Classic 인스턴스의 링크를 해제한 후 VPC에서 ClassicLink를 비활성화할 수 있습니다.

VPC의 다른 AWS 서비스와 함께 ClassicLink 사용

링크된 EC2-Classic 인스턴스는 VPC의 AWS, Amazon Redshift, Amazon ElastiCache 및 Elastic Load Balancing Amazon RDS 서비스에 액세스할 수 있습니다. 그러나 VPC의 인스턴스는 AWS를 사용하여 EC2-Classic 플랫폼이 프로비저닝한 ClassicLink 서비스에 액세스할 수 없습니다.

Elastic Load Balancing을 사용하는 경우 연결된 EC2-Classic 인스턴스를 로드 밸런서에 등록할 수 있습니다. ClassicLink가 활성화된 VPC에서 로드 밸런서를 생성하고 인스턴스가 실행되는 가용 영역을 활성화해야 합니다. 링크된 EC2-Classic 인스턴스를 종료하면 로드 밸런서가 인스턴스의 등록을 해제합니다.

Amazon EC2 Auto Scaling 사용 시에는 시작될 때 지정한 Amazon EC2 Auto Scaling 가능 VPC에 자동으로 링크되는 인스턴스가 포함된 ClassicLink 그룹을 생성할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서에서 [VPC에 EC2-Classic 인스턴스 링크](#)를 참조하십시오.

VPC에서 Amazon RDS 인스턴스 또는 Amazon Redshift 클러스터를 사용하고 공개적으로 액세스 가능(인터넷에서 액세스 가능)한 경우, 기본적으로 연결된 EC2-Classic 인스턴스에서 이러한 리소스를 처리하는 데 사용하는 엔드포인트가 퍼블릭 IP 주소로 확인됩니다. 이러한 리소스에 공개적으로 액세스할 수 없는 경우에는 엔드포인트가 프라이빗 IP 주소로 확인됩니다. ClassicLink를 사용하여 프라이빗 IP를 통해 공개적으로 액세스 가능한 RDS 인스턴스 또는 Redshift 클러스터를 처리하려면 해당 프라이빗 IP 주소 또는 프라이빗 DNS 호스트 이름을 사용하거나 VPC에 대해 ClassicLink DNS 지원을 활성화해야 합니다.

프라이빗 DNS 호스트 이름 또는 프라이빗 IP 주소를 사용하여 RDS 인스턴스를 처리하는 경우 링크된 EC2-Classic 인스턴스에서 다른 AZ 배포에 장애 조치 지원을 사용할 수 없습니다.

Amazon EC2 콘솔을 사용하여 Amazon Redshift, Amazon ElastiCache 또는 Amazon RDS 리소스의 프라이빗 IP 주소를 확인할 수 있습니다.

VPC에서 AWS 리소스의 프라이빗 IP 주소를 확인하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스를 선택합니다.
3. 설명 열에서 네트워크 인터페이스의 설명을 확인합니다. Amazon Redshift, Amazon ElastiCache 또는 Amazon RDS에 사용된 네트워크 인터페이스의 설명에는 서비스의 이름이 명시됩니다. 예를 들어 Amazon RDS 인스턴스에 연결된 네트워크 인터페이스의 설명은 `RDSNetworkInterface`입니다.
4. 필요한 네트워크 인터페이스를 선택합니다.
5. 세부 정보 창의 기본 프라이빗 IPv4 IP 필드에서 프라이빗 IP 주소를 확인합니다.

ClassicLink 사용 제어

기본적으로 IAM 사용자에게는 ClassicLink 사용 권한이 없습니다. 사용자에게 VPC의 IAM을 활성화/비활성화하고, ClassicLink 가능 VPC에 인스턴스를 링크하거나 링크를 해제하고, ClassicLink 가능 VPC 및 링크된 ClassicLink 인스턴스를 조회하는 권한을 부여하는 EC2-Classic 정책을 생성할 수 있습니다. IAM의 Amazon EC2 정책에 대한 자세한 내용은 [Amazon EC2에 대한 IAM 정책 \(p. 838\)](#) 단원을 참조하십시오.

ClassicLink 작업 관련 정책에 대한 자세한 내용은 [ClassicLink에 대한 예제 IAM 정책 \(p. 816\)](#)의 예제를 참조하십시오.

ClassicLink의 보안 그룹

EC2-Classic 인스턴스를 VPC에 링크해도 EC2-Classic 보안 그룹에는 영향이 없습니다. 보안 그룹은 인스턴스가 송수신하는 모든 트래픽을 계속 제어합니다. VPC의 인스턴스가 송수신하는 트래픽은 여기에서 제외되며, EC2-Classic 인스턴스에 연결한 VPC 보안 그룹에 의해 제어됩니다. 같은 VPC에 링크된 여러 EC2-Classic 인스턴스는 같은 VPC 보안 그룹에 연결되어 있는지 여부에 관계없이 VPC를 통해 서로 통신할 수 있습니다. EC2-Classic 인스턴스 간의 통신은 이러한 인스턴스에 연결된 EC2-Classic 보안 그룹에 의해 제어됩니다. 보안 그룹 구성의 예는 [예: 3티어 웹 애플리케이션의 ClassicLink 보안 그룹 구성 \(p. 818\)](#) 단원을 참조하십시오.

인스턴스를 VPC에 링크한 후에는 인스턴스에 연결된 VPC 보안 그룹을 변경할 수 없습니다. 인스턴스에 다른 보안 그룹을 연결하려면 우선 인스턴스의 링크를 해제한 후 VPC에 다시 링크하면서 필요한 보안 그룹을 선택해야 합니다.

ClassicLink의 라우팅

VPC에서 ClassicLink를 활성화하면 모든 VPC 라우팅 테이블에 대상이 10.0.0.0/8, 타겟이 `local`인 고정 라우팅이 추가됩니다. 따라서 VPC의 인스턴스와 VPC에 링크된 EC2-Classic 인스턴스 간에 통신이 가능합니다. ClassicLink 가능 VPC에 사용자 지정 라우팅 테이블을 추가하면 대상이 10.0.0.0/8, 타겟이

local인 고정 라우팅이 자동으로 추가됩니다. VPC에서 ClassicLink를 비활성화하면 이 라우팅이 모든 VPC 라우팅 테이블에서 자동으로 삭제됩니다.

10.0.0.0/16 및 10.1.0.0/16 IP 주소 범위에 속하는 VPC에서 ClassicLink를 활성화하려면 라우팅 테이블에 10.0.0.0/8 IP 주소 범위에 속하는 기존 고정 라우팅이 없어야 합니다. VPC 생성 시 자동으로 추가된 로컬 라우팅은 여기에서 제외됩니다. 마찬가지로 VPC에서 ClassicLink를 활성화한 경우 10.0.0.0/8 IP 주소 범위에 속하는 특정 라우팅을 라우팅 테이블에 추가할 수 없습니다.

Important

VPC의 CIDR 블록이 공개적으로 라우팅 가능한 IP 주소 범위인 경우, VPC에 EC2-Classic 인스턴스를 링크하기 전에 보안 문제를 고려해야 합니다. 예를 들어 링크된 EC2-Classic 인스턴스가 VPC의 IP 주소 범위에 속하는 소수 IP 주소로부터 DoS(Denial of Service) 요청 포화 공격을 받는 경우 응답 트래픽이 VPC로 전송됩니다. [RFC 1918](#) 규격에 따라 프라이빗 IP 주소 범위를 사용하여 VPC를 생성하는 것이 좋습니다.

라우팅 테이블 및 VPC의 라우팅에 대한 자세한 내용은 Amazon VPC 사용 설명서에서 [라우팅 테이블](#)을 참조하십시오.

ClassicLink에 대한 VPC 피어링 연결 활성화

두 VPC 간에 VPC 피어링 연결이 있고 ClassicLink를 통해 이 두 VPC 중 하나 또는 둘 다에 연결된 하나 이상의 EC2-Classic 인스턴스가 있는 경우, EC2-Classic 인스턴스와 VPC 피어링 연결의 다른 쪽에 있는 VPC의 인스턴스 간 통신이 활성화되도록 VPC 피어링 연결을 확장할 수 있습니다. 이렇게 하면 EC2-Classic 인스턴스와 VPC의 인스턴스가 프라이빗 IP 주소를 사용하여 통신할 수 있습니다. 이를 위해 로컬 VPC가 피어 VPC의 연결된 EC2-Classic 인스턴스와 통신하도록 하거나, 로컬 연결된 EC2-Classic 인스턴스가 피어 VPC의 인스턴스와 통신하도록 할 수 있습니다.

로컬 VPC가 피어 VPC의 연결된 EC2-Classic 인스턴스와 통신하도록 하면, 목적지가 10.0.0.0/8이고 대상이 local인 라우팅 테이블에 정적 경로가 자동으로 추가됩니다.

자세한 내용과 예시는 Amazon VPC Peering Guide의 [ClassicLink로 구성](#) 단원을 참조하십시오.

ClassicLink 제한 사항

ClassicLink 기능을 사용하려면 다음과 같은 제한 사항을 숙지해야 합니다.

- 한 번에 하나의 VPC에만 EC2-Classic 인스턴스를 링크할 수 있습니다.
- 링크된 EC2-Classic 인스턴스를 종지하면 자동으로 VPC에서 링크가 해제되고 VPC 보안 그룹이 인스턴스에 더 이상 연결되지 않습니다. 인스턴스를 다시 시작한 후 VPC에 다시 연결할 수 있습니다.
- 다른 리전이나 다른 AWS 계정에 속하는 VPC에는 EC2-Classic 인스턴스를 링크할 수 없습니다.
- ClassicLink를 사용하여 VPC 인스턴스를 다른 VPC 또는 EC2-Classic 리소스에 링크할 수 없습니다. VPC 간에 프라이빗 연결을 설정하려면 VPC 피어링 연결을 사용합니다. 자세한 내용은 [Amazon VPC Peering Guide](#)의 내용을 참조하십시오.
- 링크된 EC2-Classic 인스턴스에는 VPC 탄력적 IP 주소를 연결할 수 없습니다.
- IPv6 통신에 대해 EC2-Classic 인스턴스를 활성화할 수 없습니다. IPv6 CIDR 블록을 VPC와 연결하고 IPv6 주소를 VPC의 리소스에 할당할 수 있지만 ClassicLinked 인스턴스와 VPC의 리소스 간 통신은 IPv4를 통해서만 이루어집니다.
- EC2-Classic 프라이빗 IP 주소 범위 10/8과 충돌하는 라우팅이 있는 VPC에서는 ClassicLink를 활성화할 수 없습니다. 라우팅 테이블에 이미 로컬 라우팅이 있는 10.0.0.0/16 및 10.1.0.0/16 IP 주소 범위의 VPC는 여기에 포함되지 않습니다. 자세한 내용은 [ClassicLink의 라우팅 \(p. 811\)](#) 단원을 참조하십시오.
- 전용 하드웨어 테넌시로 구성된 VPC에서는 ClassicLink를 활성화할 수 없습니다. 전용 테넌시 VPC에서 ClassicLink를 활성화하려는 경우 AWS Support에 문의하십시오.

Important

EC2-Classic 인스턴스는 공유된 하드웨어에서 실행됩니다. 규제 또는 보안 요구 사항으로 인해 VPC의 테넌시를 dedicated로 설정한 경우 VPC에 EC2-Classic 인스턴스를 연결하면 공유된

테넌시 리소스가 프라이빗 IP 주소를 사용하여 격리된 리소스를 직접 처리할 수 있으므로 이러한 요구 사항을 준수하지 않을 수 있습니다. ClassicLink용 전용 VPC를 활성화하려는 경우 AWS 지원 요청에 대한 자세한 이유를 기재하십시오.

- EC2-Classic 인스턴스를 172.16.0.0/16 범위의 VPC에 연결하고 해당 VPC 내의 172.16.0.23/32 IP 주소로 DNS 서버를 실행하는 경우 연결된 EC2-Classic 인스턴스는 VPC DNS 서버에 액세스할 수 없습니다. 이 문제를 해결하려면 DNS 서버를 VPC 내에서 다른 IP 주소로 실행합니다.
- ClassicLink는 VPC 외부의 전이 관계를 지원하지 않습니다. 연결된 EC2-Classic 인스턴스는 VPC에 연결된 VPN 연결, VPC 게이트웨이 엔드포인트, NAT 게이트웨이 또는 인터넷 게이트웨이에 액세스할 수 없습니다. 마찬가지로 VPN 연결의 반대편에 있는 리소스 또는 인터넷 게이트웨이는 연결된 EC2-Classic 인스턴스에 액세스할 수 없습니다.

ClassicLink 작업

Amazon EC2 및 Amazon VPC 콘솔을 사용하여 ClassicLink 관련 작업을 수행할 수 있습니다. VPC에서 ClassicLink를 활성화 또는 비활성화하고 VPC에 EC2-Classic 인스턴스를 링크하거나 링크를 해제할 수 있습니다.

Note

ClassicLink 기능은 EC2-Classic을 지원하는 계정 및 리전의 콘솔에만 표시됩니다.

작업

- [VPC에서 ClassicLink 활성화 \(p. 813\)](#)
- [ClassicLink가 활성화된 VPC 생성 \(p. 813\)](#)
- [VPC에 인스턴스 링크 \(p. 814\)](#)
- [인스턴스 시작 시 VPC에 링크 \(p. 814\)](#)
- [ClassicLink 가능 VPC 및 링크된 인스턴스 보기 \(p. 815\)](#)
- [ClassicLink DNS 지원 활성화 \(p. 815\)](#)
- [ClassicLink DNS 지원 비활성화 \(p. 815\)](#)
- [VPC에서 인스턴스 링크 해제 \(p. 816\)](#)
- [VPC에 대해 ClassicLink 비활성화 \(p. 816\)](#)

VPC에서 ClassicLink 활성화

VPC에 EC2-Classic 인스턴스를 링크하려면 우선 VPC에서 ClassicLink를 활성화해야 합니다. VPC에 ClassicLink 프라이빗 IP 주소 범위와 충돌하는 라우팅이 있는 경우 VPC에서 EC2-Classic을 활성화할 수 없습니다. 자세한 내용은 [ClassicLink의 라우팅 \(p. 811\)](#) 단원을 참조하십시오.

VPC에서 ClassicLink를 활성화하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 VPC를 선택합니다.
3. VPC를 선택한 다음 작업, ClassicLink 활성화를 선택합니다.
4. 확인 대화 상자에서 예, 활성화를 선택합니다.
5. (선택 사항) 퍼블릭 DNS 호스트 이름이 프라이빗 IP 주소로 확인되도록 하려면 인스턴스를 연결하기 전에 VPC에 대해 ClassicLink DNS 지원을 활성화합니다. 자세한 내용은 [ClassicLink DNS 지원 활성화 \(p. 815\)](#) 단원을 참조하십시오.

ClassicLink가 활성화된 VPC 생성

ClassicLink 콘솔에서 VPC 마법사를 사용하면 새 VPC를 생성할 때 Amazon VPC를 즉시 활성화할 수 있습니다.

ClassicLink가 활성화된 VPC를 생성하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. Amazon VPC 대시보드에서 VPC 마법사 시작을 선택합니다.
3. VPC 구성 옵션 중 하나를 선택하고 선택을 선택합니다.
4. 마법사 다음 페이지의 ClassicLink 활성화에서 예를 선택합니다. 마법사의 나머지 단계를 완료하여 VPC를 생성합니다. VPC 마법사 사용에 대한 자세한 정보는 Amazon VPC 사용 설명서에서 [Amazon VPC용 시나리오](#) 단원을 참조하십시오.
5. (선택 사항) 퍼블릭 DNS 호스트 이름이 프라이빗 IP 주소로 확인되도록 하려면 인스턴스를 연결하기 전에 VPC에 대해 ClassicLink DNS 지원을 활성화합니다. 자세한 내용은 [ClassicLink DNS 지원 활성화 \(p. 815\)](#) 단원을 참조하십시오.

VPC에 인스턴스 링크

VPC에서 ClassicLink를 활성화한 후 VPC에 EC2-Classic 인스턴스를 링크할 수 있습니다.

Note

실행 중인 EC2-Classic 인스턴스만 VPC에 링크할 수 있습니다. stopped 상태인 인스턴스는 링크 할 수 없습니다.

퍼블릭 DNS 호스트 이름이 프라이빗 IP 주소로 확인되도록 하려면 인스턴스를 연결하기 전에 VPC에 대해 ClassicLink DNS 지원을 활성화합니다. 자세한 내용은 [ClassicLink DNS 지원 활성화 \(p. 815\)](#) 단원을 참조하십시오.

VPC에 인스턴스를 링크하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 실행 중인 EC2-Classic 인스턴스를 선택한 다음, 작업, ClassicLink, VPC에 연결을 선택합니다. 둘 이상의 인스턴스를 선택하여 동일한 VPC에 링크할 수 있습니다.
4. 표시되는 대화 상자의 목록에서 VPC를 선택합니다. ClassicLink가 활성화된 VPC만 표시됩니다.
5. VPC의 보안 그룹을 하나 이상 선택하여 인스턴스와 연결합니다. 완료되면 VPC에 연결을 선택합니다.

인스턴스 시작 시 VPC에 링크

Amazon EC2 콘솔에서 시작 마법사를 사용하면 EC2-Classic 인스턴스를 시작할 때 ClassicLink 가능 VPC에 즉시 링크할 수 있습니다.

인스턴스 시작 시 VPC에 링크하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. Amazon EC2 대시보드에서 인스턴스 시작을 선택합니다.
3. AMI를 선택하고 인스턴스 유형을 선택합니다. 인스턴스 세부 정보 구성 페이지의 네트워크 목록에서 EC2-Classic으로 시작을 선택합니다.

Note

T2 인스턴스 유형 등의 일부 인스턴스 유형은 VPC로만 시작할 수 있습니다. EC2-Classic으로 시작할 수 있는 인스턴스 유형을 선택해야 합니다.

4. VPC 링크(ClassicLink) 섹션의 VPC에 연결에서 VPC를 선택합니다. ClassicLink 가능 VPC만 표시됩니다. VPC에서 인스턴스에 연결할 보안 그룹을 선택합니다. 페이지의 다른 구성 옵션을 완료한 후 마법사의 나머지 단계를 완료하여 인스턴스를 시작합니다. 시작 마법사 사용에 대한 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오.

ClassicLink 가능 VPC 및 링크된 인스턴스 보기

ClassicLink 콘솔에서 모든 Amazon VPC 가능 VPC를, EC2-Classic 콘솔에서 링크된 Amazon EC2 인스턴스를 확인할 수 있습니다.

ClassicLink 가능 VPC를 확인하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 VPC를 선택합니다.
3. VPC를 선택하고 요약 탭에서 ClassicLink 필드를 찾습니다. 값이 활성화이면 VPC에서 ClassicLink가 활성화된 것입니다.
4. 또는 ClassicLink 열을 찾고 각 VPC에 표시된 값(활성화 또는 비활성화)을 확인합니다. 해당 열이 보이지 않는 경우 테이블 열 편집(기어 모양 아이콘)을 선택하고 ClassicLink 속성을 선택한 다음 닫기를 선택합니다.

링크된 EC2-Classic 인스턴스를 확인하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. EC2-Classic 인스턴스를 선택하고 설명 탭에서 ClassicLink 필드를 찾습니다. 인스턴스가 VPC에 링크된 경우 필드에 인스턴스가 링크된 VPC의 ID가 표시됩니다. 인스턴스가 VPC에 링크되지 않은 경우 필드에 연결 해제됨이 표시됩니다.
4. 또는 인스턴스를 필터링하여 특정 VPC 또는 보안 그룹에 링크된 EC2-Classic 인스턴스만 표시할 수도 있습니다. 검색 창에서 classicLink를 입력하고 관련 ClassicLink 리소스 속성을 선택한 후 보안 그룹 ID 또는 VPC ID를 선택합니다.

ClassicLink DNS 지원 활성화

연결된 EC2-Classic 인스턴스와 VPC의 인스턴스 사이에서 처리되는 DNS 호스트 이름이 퍼블릭 IP 주소가 아니라 프라이빗 IP 주소로 확인되도록 VPC에 대해 ClassicLink DNS 지원을 비활성화할 수 있습니다. 이 기능을 사용하려면 DNS 호스트 이름과 DNS 확인에 대해 VPC가 활성화되어 있어야 합니다.

Note

VPC에 대해 ClassicLink DNS 지원을 활성화하면, 연결된 EC2-Classic 인스턴스는 VPC에 연결된 어떤 프라이빗 호스팅 영역에도 액세스할 수 있습니다. 자세한 내용은 Amazon Route 53 개발자 안내서에서 [프라이빗 호스팅 영역 작업 단원](#)을 참조하십시오.

ClassicLink DNS 지원을 활성화하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 VPC를 선택합니다.
3. VPC를 선택한 다음 작업, ClassicLink DNS 지원 편집을 선택합니다.
4. ClassicLink DNS 지원에서 활성화를 선택한 다음 저장을 선택합니다.

ClassicLink DNS 지원 비활성화

연결된 EC2-Classic 인스턴스와 VPC의 인스턴스 사이에서 처리되는 DNS 호스트 이름이 프라이빗 IP 주소가 아니라 퍼블릭 IP 주소로 확인되도록 VPC에 대해 ClassicLink DNS 지원을 비활성화할 수 있습니다.

ClassicLink DNS 지원을 비활성화하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.

2. 탐색 창에서 VPC를 선택합니다.
3. VPC를 선택한 다음 작업, ClassicLink DNS 지원 편집을 선택합니다.
4. ClassicLink DNS 지원에서 활성화 확인란을 선택 취소한 다음 저장을 선택합니다.

VPC에서 인스턴스 링크 해제

ClassicLink 인스턴스와 VPC 간에 EC2-Classic 연결이 더 이상 필요하지 않은 경우 VPC에서 인스턴스의 링크를 해제할 수 있습니다. 인스턴스의 링크를 해제하면 VPC 보안 그룹이 인스턴스에서 분리됩니다.

Note

인스턴스를 중지하면 자동으로 VPC와 링크가 해제됩니다.

VPC에서 인스턴스의 링크를 해제하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
3. 작업 목록에서 ClassicLink, 인스턴스 링크 해제를 선택합니다. 둘 이상의 인스턴스를 선택하여 동일한 VPC에서 링크를 해제할 수 있습니다.
4. 확인 대화 상자에서 예를 선택합니다.

VPC에 대해 ClassicLink 비활성화

EC2-Classic 인스턴스와 VPC 간에 연결이 더 이상 필요하지 않은 경우 VPC에서 ClassicLink를 비활성화할 수 있습니다. 우선 VPC에 링크된 모든 EC2-Classic 인스턴스의 링크를 해제해야 합니다.

VPC에서 ClassicLink를 비활성화하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 VPC를 선택합니다.
3. VPC를 선택한 다음, 작업, ClassicLink 비활성화를 선택합니다.
4. 확인 대화 상자에서 예, 비활성화를 선택합니다.

ClassicLink에 대한 예제 IAM 정책

VPC에서 ClassicLink를 활성화한 후 VPC에 EC2-Classic 인스턴스를 링크할 수 있습니다. 또한 ClassicLink 가능 VPC 및 VPC에 링크된 모든 EC2-Classic 인스턴스를 확인할 수 있습니다. `ec2:EnableVpcClassicLink`, `ec2:DisableVpcClassicLink`, `ec2:AttachClassicLinkVpc` 및 `ec2:DetachClassicLinkVpc` 작업에 대한 리소스 수준 권한을 포함하는 정책을 생성하여 사용자가 해당 작업을 사용할 수 있는지 여부를 제어할 수 있습니다. `ec2:Describe*` 작업에는 리소스 수준 권한이 지원되지 않습니다.

예제:

- ClassicLink 작업 관련 전체 권한 (p. 816)
- VPC에서 ClassicLink 활성화 및 비활성화 (p. 817)
- 인스턴스 링크 (p. 817)
- 인스턴스 링크 해제 (p. 818)

ClassicLink 작업 관련 전체 권한

다음 정책은 ClassicLink 가능 VPC 및 링크된 EC2-Classic 인스턴스를 확인하고, VPC에서 ClassicLink를 활성화 및 비활성화하고, ClassicLink 가능 VPC에서 인스턴스를 링크 및 링크 해제할 권한을 부여합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:DescribeClassicLinkInstances", "ec2:DescribeVpcClassicLink",  
                "ec2:EnableVpcClassicLink", "ec2:DisableVpcClassicLink",  
                "ec2:AttachClassicLinkVpc", "ec2:DetachClassicLinkVpc"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

VPC에서 ClassicLink 활성화 및 비활성화

다음 정책은 'ClassicLink' 태그가 있는 VPC에서 purpose=classiclink를 활성화 및 비활성화하도록 허용합니다. 다른 VPC에서는 ClassicLink를 활성화하거나 비활성화할 수 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:*VpcClassicLink",  
            "Resource": "arn:aws:ec2:region:account:vpc/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:ResourceTag/purpose": "classiclink"  
                }  
            }  
        }  
    ]  
}
```

인스턴스 링크

다음 정책은 인스턴스가 m3.large 유형일 때만 VPC에 인스턴스를 링크할 권한을 부여합니다. 두 번째 명령문에서는 인스턴스를 VPC에 링크하는 데 필요한 VPC 및 보안 그룹 리소스 사용을 허용합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:AttachClassicLinkVpc",  
            "Resource": "arn:aws:ec2:region:account:instance/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:InstanceType": "m3.large"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:AttachClassicLinkVpc",  
            "Resource": [  
                "arn:aws:ec2:region:account:vpc/*",  
                "arn:aws:ec2:region:account:security-group/*"  
            ]  
        }  
    ]  
}
```

```
    ]  
}
```

다음 정책은 인스턴스를 특정 VPC(vpc-1a2b3c4d)에만 링크하고 VPC의 특정 보안 그룹(sg-1122aabb 및 sg-aabb2233)만 인스턴스에 연결할 권한을 부여합니다. 사용자는 다른 VPC에는 인스턴스를 링크할 수 없고, 요청 시 인스턴스에 연결할 다른 VPC 보안 그룹을 지정할 수 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:AttachClassicLinkVpc",  
            "Resource": [  
                "arn:aws:ec2:region:account:vpc/vpc-1a2b3c4d",  
                "arn:aws:ec2:region:account:instance/*",  
                "arn:aws:ec2:region:account:security-group/sg-1122aabb",  
                "arn:aws:ec2:region:account:security-group/sg-aabb2233"  
            ]  
        }  
    ]  
}
```

인스턴스 링크 해제

다음 정책은 인스턴스에 "EC2-Classic" 태그가 있을 때만 VPC에서 링크된 `unlink=true` 인스턴스의 링크를 해제할 권한을 부여합니다. 두 번째 명령문에서는 VPC에서 인스턴스의 링크를 해제하는 데 필요한 VPC 리소스를 사용할 권한을 부여합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:DetachClassicLinkVpc",  
            "Resource": [  
                "arn:aws:ec2:region:account:instance/*"  
            ],  
            "Condition": {  
                "StringEquals": {  
                    "ec2:ResourceTag/unlink": "true"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:DetachClassicLinkVpc",  
            "Resource": [  
                "arn:aws:ec2:region:account:vpc/*"  
            ]  
        }  
    ]  
}
```

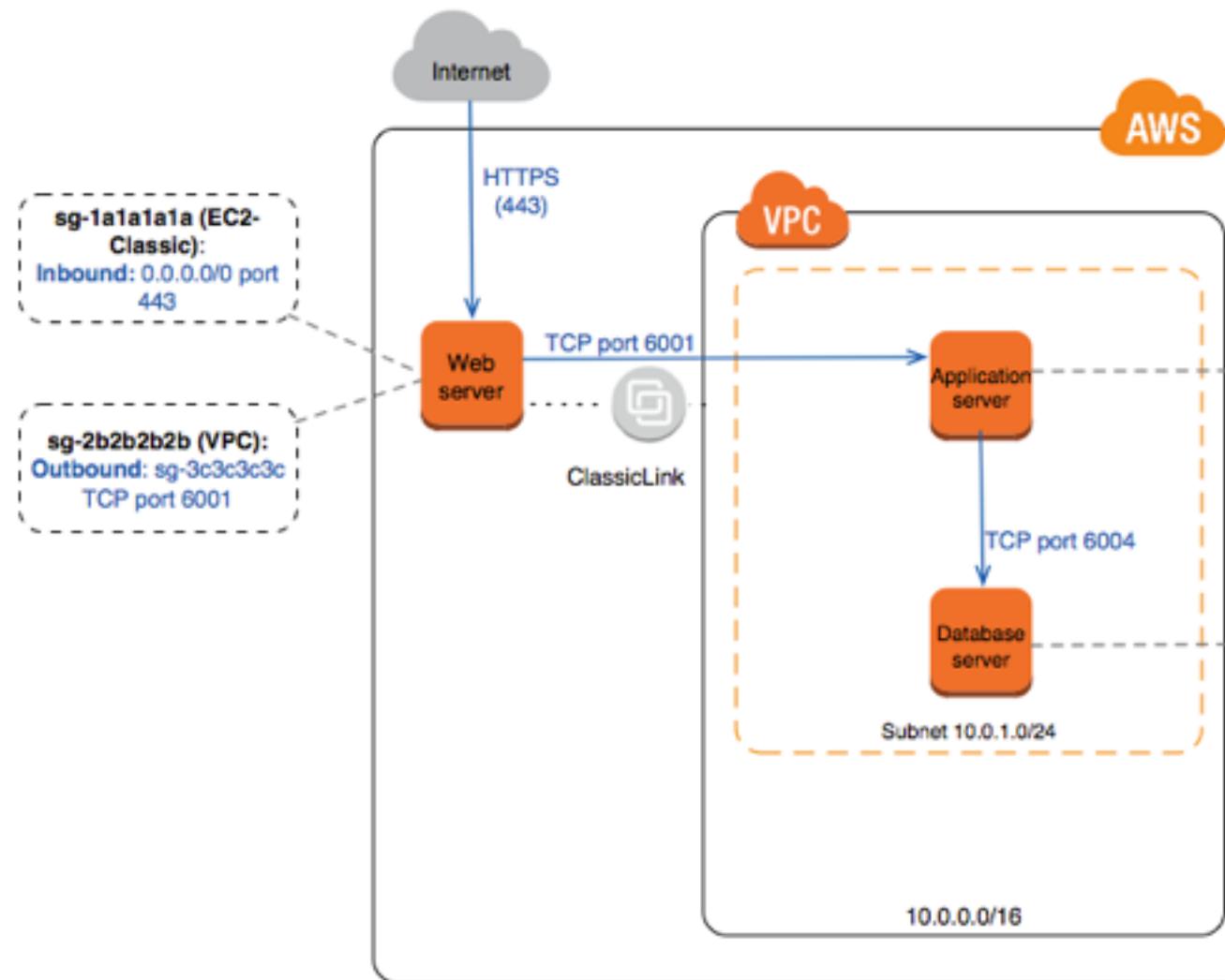
예: 3티어 웹 애플리케이션의 ClassicLink 보안 그룹 구성

이 예에서 애플리케이션은 세 가지 인스턴스, 즉 퍼블릭 웹 서버, 애플리케이션 서버, 그리고 데이터베이스 서버로 구성됩니다. 웹 서버는 인터넷의 HTTPS 트래픽을 수신한 후 TCP 포트 6001을 통해 애플리케이션 서버와 통신합니다. 그런 다음 애플리케이션 서버는 TCP 포트 6004를 통해 데이터베이스 서버와 통신합니다. 현재 사용자는 전체 애플리케이션을 계정 내 VPC로 마이그레이션하는 중입니다. 애플리케이션

서버와 데이터베이스 서버는 이미 VPC로 마이그레이션하였습니다. 웹 서버는 아직 EC2-Classic에 있고 ClassicLink를 통해 VPC로 링크된 상태입니다.

사용자는 이 세 가지 인스턴스에서만 트래픽을 주고받을 수 있도록 보안 그룹을 구성하려고 합니다. 보안 그룹은 웹 서버용 2개(sg-1a1a1a1a, sg-2b2b2b2b), 애플리케이션 서버용 1개(sg-3c3c3c3c), 그리고 데이터베이스 서버용 1개(sg-4d4d4d4d)까지 총 4개입니다.

다음은 인스턴스 아키텍처와 보안 그룹 구성을 나타낸 다이어그램입니다.



웹 서버용 보안 그룹(sg-1a1a1a1a, sg-2b2b2b2b)

보안 그룹 하나는 EC2-Classic에, 그리고 나머지 하나는 VPC에 있습니다. 그리고 ClassicLink를 통해 인스턴스를 VPC로 링크했을 때 VPC 보안 그룹을 웹 서버 인스턴스와 연동시켰습니다. 이제 웹 서버에서 애플리케이션 서버로 보내지는 아웃바운드 트래픽을 VPC 보안 그룹에서 제어할 수 있습니다.

다음은 EC2-Classic 보안 그룹(sg-1a1a1a1a)에 적용되는 보안 그룹 규칙입니다.

Inbound			
Source	Type	Port Range	Comments

0.0.0.0/0	HTTPS	443	인터넷 트래픽의 웹 서버 전송을 허용합니다.
-----------	-------	-----	--------------------------

다음은 VPC 보안 그룹(sg-2b2b2b2b)에 적용되는 보안 그룹 규칙입니다.

Outbound			
Destination	Type	Port Range	Comments
sg-3c3c3c3c	TCP	6001	웹 서버에서 VPC의 애플리케이션 서버(또는 sg-3c3c3c3c와 연동된 다른 인스턴스)로 전송되는 아웃바운드 트래픽을 허용합니다.

애플리케이션 서버용 보안 그룹(**sg-3c3c3c3c**)

다음은 애플리케이션 서버와 연동되어 있는 VPC 보안 그룹의 보안 그룹 규칙입니다.

Inbound			
Source	Type	Port Range	Comments
sg-2b2b2b2b	TCP	6001	웹 서버(또는 sg-2b2b2b2b와 연동되어 있는 기타 인스턴스)에서 애플리케이션 서버로 특정 유형의 트래픽을 전송할 수 있도록 허용합니다.
Outbound			
Destination	Type	Port Range	Comments
sg-4d4d4d4d	TCP	6004	애플리케이션 서버에서 데이터베이스 서버(또는 sg-4d4d4d4d와 연동되어 있는 기타 인스턴스)로 전송되는 아웃바운드 트래픽을 허용합니다.

데이터베이스 서버용 보안 그룹(**sg-4d4d4d4d**)

다음은 데이터베이스 서버와 연동되어 있는 VPC 보안 그룹의 보안 그룹 규칙입니다.

Inbound			
Source	Type	Port Range	Comments
sg-3c3c3c3c	TCP	6004	애플리케이션 서버(또는 sg-3c3c3c3c와 연동되어 있는 기타 인스턴스)에서 데이터베이스 서버로 특정 유형의 트래픽을 전송할 수 있도록 허용합니다.

EC2-Classic에서 VPC로 마이그레이션

2013년 12월 4일 이전에 AWS 계정을 생성한 경우 일부 AWS 리전에서 EC2-Classic 지원을 받을 수 있습니다. 향상된 네트워킹 및 새로운 인스턴스 유형과 같은 일부 Amazon EC2 리소스 및 기능에는 Virtual Private

Cloud(VPC)가 필요합니다. 일부 리소스는 EC2-Classic 및 VPC 간에 공유될 수 있으며 일부는 공유될 수 없습니다. 자세한 내용은 [EC2-Classic과 VPC 간 리소스 공유 및 액세스 \(p. 808\)](#) 단원을 참조하십시오. VPC 전용 기능을 활용하려면 VPC로 마이그레이션하는 것이 좋습니다.

EC2-Classic에서 VPC로 마이그레이션하려면 VPC에서 EC2-Classic 리소스를 마이그레이션하거나 다시 생성해야 합니다. 전체 리소스를 마이그레이션하고 다시 생성하거나 ClassicLink를 사용하여 시간이 지남에 따라 종분식 마이그레이션을 수행할 수 있습니다.

목차

- [기본 VPC를 가져오기 위한 옵션 \(p. 821\)](#)
- [리소스를 VPC로 마이그레이션 \(p. 822\)](#)
- [종분식 마이그레이션에 ClassicLink 사용 \(p. 825\)](#)
- [예제: 간단한 웹 애플리케이션 마이그레이션 \(p. 827\)](#)

기본 VPC를 가져오기 위한 옵션

기본 VPC는 구성되고 바로 사용할 수 있는 VPC이며, VPC 전용 리전에서만 사용할 수 있습니다. EC2-Classic을 지원하는 리전의 경우 기본이 아닌 VPC를 생성하여 리소스를 설정할 수 있습니다. 하지만 VPC를 직접 설정하지 않으려는 경우 또는 VPC 구성에 대한 특정 요구 사항이 없는 경우 기본 VPC를 사용할 수 있습니다. 기본 VPC에 대한 자세한 내용은 Amazon VPC 사용 설명서의 [기본 VPC 및 서브넷](#)을 참조하십시오.

다음은 EC2-Classic을 지원하는 AWS 계정이 있는 경우 기본 VPC를 사용하기 위한 옵션입니다.

옵션

- [VPC 전용 리전으로 전환 \(p. 821\)](#)
- [새 AWS 계정 생성 \(p. 821\)](#)
- [기존 AWS 계정을 VPC 전용으로 변환 \(p. 821\)](#)

VPC 전용 리전으로 전환

기존 계정을 사용하여 기본 VPC에서 리소스를 설정하고 특정 리전을 사용할 필요가 없는 경우 이 옵션을 사용합니다. 기본 VPC가 있는 리전을 찾으려면 [지원되는 플랫폼 감지 \(p. 802\)](#) 단원을 참조하십시오.

새 AWS 계정 생성

새 AWS 계정은 VPC만 지원합니다. 모든 리전에 기본 VPC가 있는 계정을 원하는 경우 이 옵션을 사용합니다.

기존 AWS 계정을 VPC 전용으로 변환

기존 계정의 모든 리전에 기본 VPC를 원하는 경우 이 옵션을 사용합니다. 계정을 변환하려면 먼저 모든 EC2-Classic 리소스를 삭제해야 합니다. 일부 리소스를 VPC로 마이그레이션할 수도 있습니다. 자세한 내용은 [리소스를 VPC로 마이그레이션 \(p. 822\)](#) 단원을 참조하십시오.

EC2-Classic 계정을 변환하려면

1. EC2-Classic에서 사용하기 위해 생성한 리소스를 삭제하거나 마이그레이션(해당하는 경우)합니다. 다음을 포함합니다.
 - Amazon EC2 인스턴스
 - EC2-Classic 보안 그룹(사용자 자신을 삭제할 수 없는 기본 보안 그룹 제외)
 - EC2-Classic 탄력적 IP 주소
 - Classic Load Balancer

- Amazon RDS 리소스
 - Amazon ElastiCache 리소스
 - Amazon Redshift 리소스
 - AWS Elastic Beanstalk 리소스
 - AWS Data Pipeline 리소스
 - Amazon EMR 리소스
 - AWS OpsWorks 리소스
2. console.aws.amazon.com/support에서 AWS 지원 센터로 이동합니다.
 3. Create case(사례 생성)을 선택합니다.
 4. 계정 및 결제 지원을 선택합니다.
 5. 유형에 대해 계정을 선택합니다. 카테고리에서 EC2 Classic을 VPC로 변환을 선택합니다.
 6. 필요에 따라 다른 세부 정보를 입력하고 제출을 선택합니다. 요청을 검토하고 다음 단계를 안내하기 위해 연락을 드릴 것입니다.

리소스를 VPC로 마이그레이션

일부 리소스를 VPC로 마이그레이션하거나 이동할 수 있습니다. 일부 리소스는 EC2-Classic에서 동일한 리전 및 동일한 AWS 계정에 있는 VPC로만 마이그레이션할 수 있습니다. 리소스를 마이그레이션할 수 없는 경우 VPC에서 사용할 새 리소스를 생성해야 합니다.

사전 조건

시작하기 전에 VPC가 있어야 합니다. 기본 VPC가 없는 경우 다음 방법 중 하나를 사용하여 기본이 아닌 VPC를 생성할 수 있습니다.

- Amazon VPC 콘솔에서 VPC 마법사를 사용하여 새 VPC를 생성합니다. 자세한 내용은 [Amazon VPC 콘솔 마법사 구성](#)을 참조하십시오. 사용 가능한 구성 옵션 중 하나를 사용하여 VPC를 빠르게 설정하려면 이 옵션을 사용합니다.
- Amazon VPC 콘솔에서 요구 사항에 따라 VPC의 구성 요소를 설정합니다. 자세한 내용은 [VPC 및 서브넷](#)을 참조하십시오. 특정 서브넷 수와 같이 VPC에 대한 특정 요구 사항이 있는 경우 이 옵션을 사용합니다.

주제

- [보안 그룹 \(p. 822\)](#)
- [탄력적 IP 주소 \(p. 823\)](#)
- [AMI 및 인스턴스 \(p. 823\)](#)
- [Amazon RDS DB 인스턴스 \(p. 825\)](#)

보안 그룹

VPC의 인스턴스가 EC2-Classic 인스턴스와 동일한 보안 그룹 규칙을 갖도록 하려는 경우 Amazon EC2 콘솔을 사용하여 기존 EC2-Classic 보안 그룹 규칙을 새 VPC 보안 그룹에 복사할 수 있습니다.

동일한 리전의 동일한 AWS 계정에서만 보안 그룹 규칙을 새 보안 그룹에 복사할 수 있습니다. 다른 리전 또는 다른 AWS 계정을 사용하는 경우 새 보안 그룹을 생성하고 규칙을 직접 추가해야 합니다. 자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#) 단원을 참조하십시오.

새 보안 그룹에 보안 그룹 규칙을 복사하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.

- EC2-Classic 인스턴스와 연결된 보안 그룹을 선택한 다음 작업과 새로 복사를 차례대로 선택합니다.

Note

EC2-Classic 보안 그룹을 식별하려면 VPC ID 열을 확인합니다. 각 EC2-Classic 보안 그룹에 대해 열의 값은 비어 있거나 – 기호입니다.

- 보안 그룹 생성 대화 상자에서 새 보안 그룹의 이름과 설명을 지정합니다. VPC 목록에서 해당 VPC를 선택합니다.
- 인바운드 탭이 EC2-Classic 보안 그룹의 규칙으로 채워집니다. 필요에 따라 규칙을 수정할 수 있습니다. 아웃바운드 탭에는 모든 아웃바운드 트래픽을 허용하는 규칙이 자동으로 생성되어 있습니다. 보안 그룹 규칙 수정에 대한 자세한 내용은 [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#) 단원을 참조하십시오.

Note

EC2-Classic 보안 그룹에서 다른 보안 그룹을 참조하는 규칙을 정의한 경우 VPC 보안 그룹에서는 동일한 규칙을 사용할 수 없습니다. 동일한 VPC의 보안 그룹을 참조하도록 규칙을 수정하십시오.

- Create를 선택합니다.

탄력적 IP 주소

EC2-Classic에서 사용하도록 할당한 탄력적 IP 주소를 VPC에서 사용하도록 마이그레이션할 수 있습니다. 탄력적 IP 주소는 다른 리전 또는 AWS 계정으로 마이그레이션할 수 없습니다. 자세한 내용은 [EC2-Classic에서 탄력적 IP 주소 마이그레이션 \(p. 807\)](#) 단원을 참조하십시오.

EC2-Classic에서 사용하도록 할당된 탄력적 IP 주소를 식별하려면

Amazon EC2 콘솔의 탐색 창에서 탄력적 IP를 선택합니다. 범위 열에서 값은 표준입니다.

또는 다음 `describe-addresses` 명령을 사용합니다.

```
aws ec2 describe-addresses --filters Name=domain,Values=standard
```

AMI 및 인스턴스

AMI는 Amazon EC2 인스턴스를 시작하기 위한 템플릿입니다. 기존 EC2-Classic 인스턴스를 기반으로 고유의 AMI를 생성한 다음 해당 AMI를 사용하여 인스턴스를 VPC로 시작할 수 있습니다.

주제

- [EC2-Classic 인스턴스 식별 \(p. 823\)](#)
- [AMI 생성 \(p. 824\)](#)
- [\(선택 사항\) AMI 공유 또는 복사 \(p. 824\)](#)
- [\(선택 사항\) Amazon EBS 볼륨에 데이터 저장 \(p. 824\)](#)
- [VPC로 인스턴스 시작 \(p. 825\)](#)

EC2-Classic 인스턴스 식별

EC2-Classic 및 VPC 모두에서 실행 중인 인스턴스가 있는 경우 EC2-Classic 인스턴스를 식별할 수 있습니다. Amazon EC2 콘솔의 탐색 창에서 인스턴스를 선택합니다. VPC ID 열에서 각 EC2-Classic 인스턴스의 값은 비어 있거나 – 기호입니다.

또는 다음 `describe-instances` AWS CLI 명령을 사용하고 `--query` 파라미터를 사용하여 `vpcId`의 값이 `null`인 인스턴스만 표시합니다.

```
aws ec2 describe-instances --query 'Reservations[*].Instances[?VpcId==`null`]'
```

AMI 생성

EC2-Classic 인스턴스를 식별한 후 해당 인스턴스에서 AMI를 생성할 수 있습니다.

Windows AMI를 생성하려면

자세한 내용은 [사용자 지정 Windows AMI 생성](#)을 참조하십시오.

Linux AMI를 생성하려면

Linux AMI를 생성하기 위해 사용하는 방법은 인스턴스의 루트 디바이스 유형과 인스턴스가 실행되는 운영 체제 플랫폼에 따라 다릅니다. 인스턴스의 루트 디바이스 유형을 알아내려면 인스턴스 페이지로 이동하고 인스턴스를 선택한 다음 설명 탭의 루트 디바이스 유형 필드에서 정보를 봅니다. 값이 ebs인 경우 EBS 기반 인스턴스이고, 값이 instance-store인 경우 인스턴스 스토어 기반 인스턴스입니다. [describe-instances](#) AWS CLI 명령을 사용하여 루트 디바이스 유형을 알아낼 수도 있습니다.

다음 표에서는 인스턴스의 루트 디바이스 유형과 소프트웨어 플랫폼을 기반으로 Linux AMI를 생성하는 옵션을 제공합니다.

Important

PV 및 HVM 가상화를 모두 지원하는 인스턴스 유형도 있지만, 둘 중 하나만 지원하는 유형도 있습니다. AMI를 사용하여 현재 인스턴스 유형과 다른 인스턴스 유형을 시작하려는 경우 인스턴스 유형이 AMI에서 제공하는 가상화 유형을 지원하는지 확인하십시오. AMI에서 PV 가상화를 지원하는 경우 HVM 가상화를 지원하는 인스턴스 유형을 사용하려면 기본 HVM AMI에 소프트웨어를 다시 설치해야 할 수 있습니다. PV 및 HVM 가상화에 대한 자세한 내용은 [Linux AMI 가상화 유형](#)을 참조하십시오.

인스턴스 루트 디바이스 유형	작업
EBS	인스턴스에서 EBS 기반 AMI를 생성합니다. 자세한 내용은 Amazon EBS 지원 Linux AMI 생성 을 참조하십시오.
인스턴스 스토어	AMI 도구를 사용하여 인스턴스에서 인스턴스 스토어 기반 AMI를 생성합니다. 자세한 내용은 인스턴스 스토어 지원 Linux AMI 생성 을 참조하십시오.
인스턴스 스토어	인스턴스 스토어 지원 인스턴스를 EBS 지원 인스턴스로 변환합니다. 자세한 내용은 인스턴스 스토어 지원 AMI를 Amazon EBS 지원 AMI로 변환 을 참조하십시오.

(선택 사항) AMI 공유 또는 복사

AMI를 사용하여 새 AWS 계정에서 인스턴스를 시작하려면 먼저 AMI를 새 계정과 공유해야 합니다. 자세한 내용은 [지정한 AWS 계정과 AMI 공유 \(p. 76\)](#) 단원을 참조하십시오.

AMI를 사용하여 다른 리전의 VPC에서 인스턴스를 시작하려면 먼저 AMI를 해당 리전에 복사해야 합니다. 자세한 내용은 [AMI 복사 \(p. 103\)](#) 단원을 참조하십시오.

(선택 사항) Amazon EBS 볼륨에 데이터 저장

Amazon EBS 볼륨을 생성하고 이 볼륨을 사용하여 물리적 하드 드라이브를 사용할 때와 같이 데이터를 백업하고 인스턴스에 저장할 수 있습니다. 동일한 가용 영역의 모든 인스턴스에서 —Amazon EBS 볼륨을 연결하고 분리할 수 있습니다. EC2-Classic의 인스턴스에서 볼륨을 분리하고, 동일한 가용 영역의 VPC로 시작하는 새 인스턴스에 연결할 수 있습니다.

Amazon EBS 볼륨에 대한 자세한 내용은 다음 단원을 참조하십시오.

- [Amazon EBS 볼륨 \(p. 929\)](#)
- [Amazon EBS 볼륨 생성 \(p. 948\)](#)
- [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#)

Amazon EBS 볼륨의 데이터를 백업하려면 볼륨의 정기적 스냅샷을 만듭니다. 자세한 내용은 [Amazon EBS 스냅샷 생성 \(p. 969\)](#) 단원을 참조하십시오. 필요한 경우 스냅샷에서 Amazon EBS 볼륨을 복원할 수 있습니다. 자세한 내용은 [스냅샷에서 볼륨 생성 \(p. 949\)](#) 단원을 참조하십시오.

VPC로 인스턴스 시작

AMI를 생성한 후 Amazon EC2 시작 마법사를 사용하여 VPC에서 인스턴스를 시작할 수 있습니다. 인스턴스는 기존 EC2-Classic 인스턴스와 동일한 데이터 및 구성을 사용합니다.

Note

이 기회를 사용하여 [현재 세대 인스턴스 유형으로 업그레이드](#)할 수 있습니다.

VPC로 인스턴스를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 대시보드에서 인스턴스 시작을 선택합니다.
3. Amazon Machine Image(AMI) 선택 페이지에서 나의 AMI 범주를 선택하고 생성한 AMI를 선택합니다. 또는 다른 계정에서 AMI를 공유한 경우 소유권 필터 목록에서 나와 공유를 선택합니다. EC2-Classic 계정에서 공유한 AMI를 선택합니다.
4. 인스턴스 유형 선택 페이지에서 인스턴스 유형을 선택하고 다음: 인스턴스 세부 정보 구성(Next: Configure Instance Details)을 선택합니다.
5. 인스턴스 세부 정보 구성 페이지의 네트워크 목록에서 VPC를 선택합니다. 서브넷 목록에서 필요한 서브넷을 선택합니다. 기타 필요한 세부 정보를 구성한 다음 보안 그룹 구성 페이지에 도달할 때까지 마법사의 다음 페이지로 이동합니다.
6. 기존 그룹 선택을 선택하고 VPC에 대해 생성한 보안 그룹을 선택합니다. 검토 및 시작을 선택합니다.
7. 인스턴스 정보를 검토한 다음 시작을 선택하여 키 페어를 지정하고 인스턴스를 시작합니다.

마법사의 각 단계에서 구성할 수 있는 파라미터에 대한 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 단원을 참조하십시오.

Amazon RDS DB 인스턴스

EC2-Classic DB 인스턴스를 동일한 리전의 동일한 계정에 있는 VPC로 이동할 수 있습니다. 자세한 내용은 Amazon RDS 사용 설명서의 [DB 인스턴스에 대한 VPC 업데이트](#)를 참조하십시오.

증분식 마이그레이션에 ClassicLink 사용

ClassicLink 기능을 사용하면 VPC로 증분식 마이그레이션 작업을 더 쉽게 관리할 수 있습니다. ClassicLink에서는 새 VPC 리소스가 프라이빗 IPv4 주소를 사용하여 EC2-Classic 인스턴스와 통신할 수 있도록 EC2-Classic 인스턴스를 동일 리전의 계정에 있는 VPC에 연결할 수 있습니다. 그런 다음 애플리케이션이 VPC에서 완전히 실행될 때까지 한 번에 구성 요소 하나씩 기능을 마이그레이션할 수 있습니다.

マイグレーション 중에 다운타임을 감당할 수 없는 경우(예: 프로세스를 중단할 수 없는 멀티 티어 애플리케이션이 있는 경우) 이 옵션을 사용합니다.

For more information about ClassicLink, see [ClassicLink \(p. 809\)](#).

작업

- 1단계: 마이그레이션 시퀀스 준비 (p. 826)
- 2단계: ClassicLink에 대해 VPC 활성화 (p. 826)
- 3단계: VPC에 EC2-Classic 인스턴스 연결 (p. 826)
- 4단계: VPC 마이그레이션 완료 (p. 826)

1단계: 마이그레이션 시퀀스 준비

ClassicLink를 효과적으로 사용하려면 먼저 VPC로 마이그레이션해야 하는 애플리케이션 구성 요소를 식별한 다음 해당 기능을 마이그레이션하는 순서를 확인해야 합니다.

예를 들어, 프레젠테이션 웹 서버, 백 엔드 데이터베이스 서버 및 거래용 인증 로직을 이용하는 애플리케이션이 있는 경우 인증 로직으로 마이그레이션 프로세스를 시작한 다음 데이터베이스 서버를 마이그레이션하고 마지막으로 웹 서버를 마이그레이션할 수 있습니다.

그런 다음 리소스를 마이그레이션하거나 다시 생성할 수 있습니다. 자세한 내용은 [리소스를 VPC로 마이그레이션 \(p. 822\)](#) 단원을 참조하십시오.

2단계: ClassicLink에 대해 VPC 활성화

새 VPC 인스턴스를 구성하고 애플리케이션 기능을 VPC에서 사용할 수 있게 만든 후에는 ClassicLink를 사용하여 새로운 VPC 인스턴스와 EC2-Classic 인스턴스 간에 프라이빗 IP 통신을 활성화할 수 있습니다. 먼저 ClassicLink에 대해 VPC를 활성화해야 합니다.

VPC에서 ClassicLink를 활성화하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 VPC를 선택합니다.
3. VPC를 선택한 다음 작업, ClassicLink 활성화를 선택합니다.
4. 예, 활성화를 선택합니다.

3단계: VPC에 EC2-Classic 인스턴스 연결

VPC에서 ClassicLink를 활성화한 후 EC2-Classic 인스턴스를 VPC에 연결할 수 있습니다.

VPC에 인스턴스를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. EC2-Classic 인스턴스를 선택한 다음 작업, ClassicLink 및 VPC에 연결을 차례대로 선택합니다.

Note

인스턴스가 running 상태인지 확인합니다.

4. ClassicLink 지원 VPC를 선택합니다(CclassicLink에 대해 활성화된 VPC만 표시됨).
5. VPC의 보안 그룹을 하나 이상 선택하여 인스턴스와 연결합니다. 완료되면 VPC에 연결을 선택합니다.

4단계: VPC 마이그레이션 완료

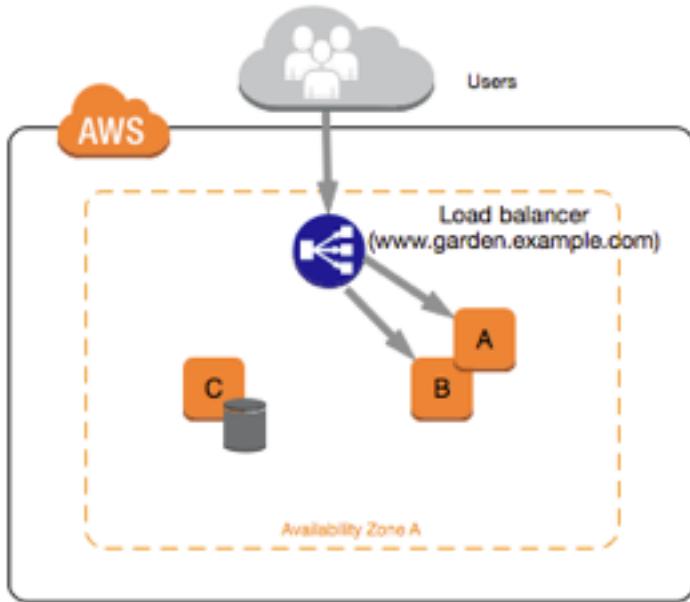
애플리케이션의 크기와 마이그레이션해야 할 기능에 따라 이전 단계를 반복하여 애플리케이션의 모든 구성 요소를 EC2-Classic에서 VPC로 이동합니다.

EC2-Classic 및 VPC 인스턴스 간에 내부 통신을 활성화한 경우 EC2-Classic 플랫폼의 서비스 대신 VPC에 있는 마이그레이션된 서비스를 가리키도록 애플리케이션을 업데이트해야 합니다. 이 작업을 위한 정확한 단계는 애플리케이션 설계에 따라 다릅니다. 일반적으로 이 작업에는 EC2-Classic 인스턴스 대신 VPC 인스턴스의 IP 주소를 가리키도록 대상 IP 주소를 업데이트하는 단계가 포함됩니다.

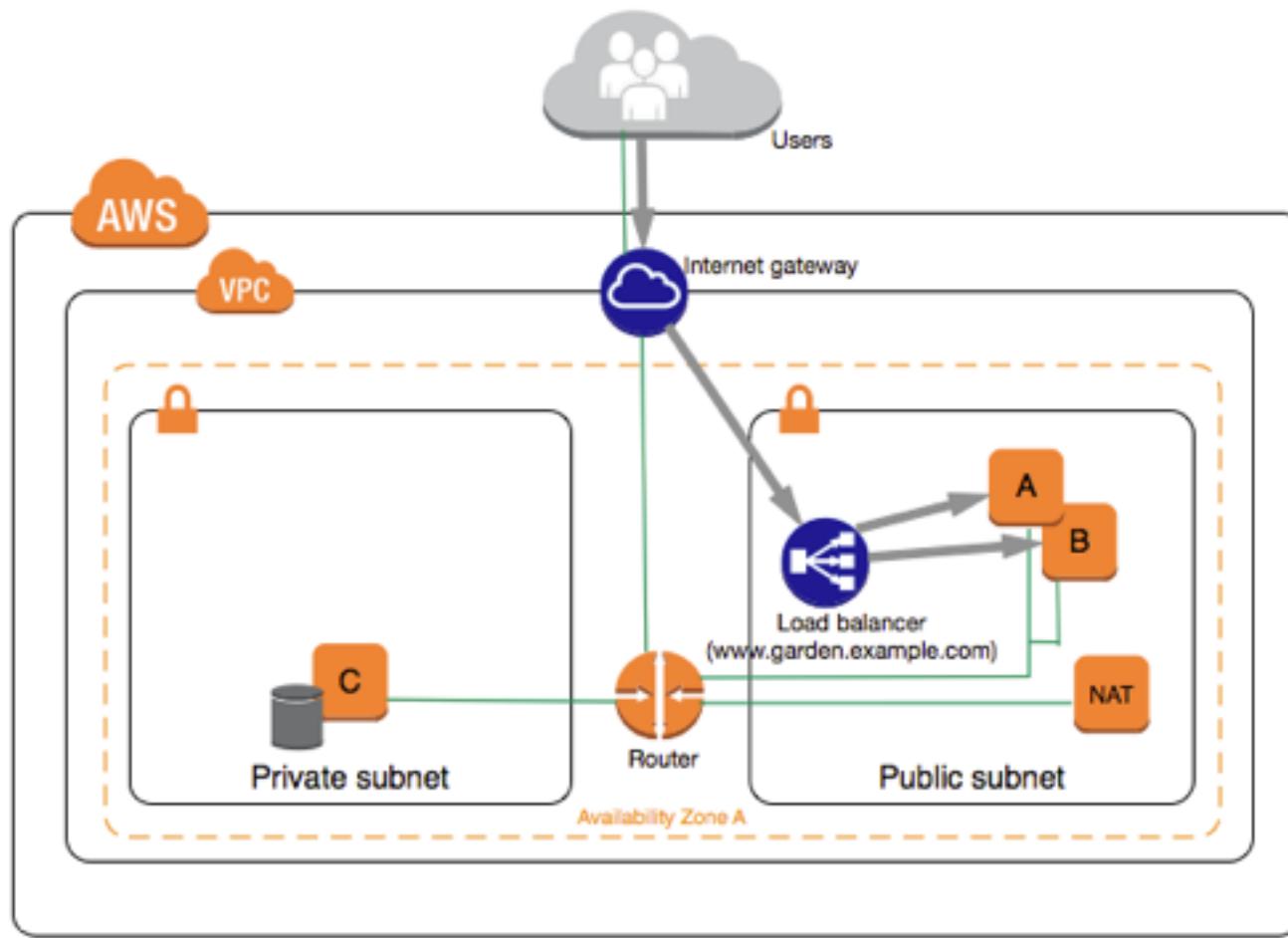
이 단계를 완료하고 애플리케이션이 VPC에서 작동하는지 테스트한 후 EC2-Classic 인스턴스를 종료하고 VPC에 대해 ClassicLink를 비활성화할 수 있습니다. 또한 더 이상 필요하지 않은 EC2-Classic 리소스를 정리하여 요금이 부과되지 않도록 할 수 있습니다. 예를 들어 탄력적 IP 주소를 해제하고 EC2-Classic 인스턴스와 연결된 볼륨을 삭제할 수 있습니다.

예제: 간단한 웹 애플리케이션 마이그레이션

이 예제에서는 AWS를 사용하여 원예 웹 사이트를 호스팅합니다. 웹 사이트를 관리하기 위해 EC2-Classic에서 세 개의 인스턴스를 실행하고 있습니다. 인스턴스 A와 B는 퍼블릭 웹 애플리케이션을 호스팅하며 Elastic Load Balancing을 사용하여 이 두 인스턴스 간 트래픽의 로드 밸런스를 유지합니다. 탄력적 IP 주소를 인스턴스 A와 B에 배정하여 해당 인스턴스에 구성 및 관리 작업을 위한 고정 IP 주소를 만들었습니다. 인스턴스 C에는 웹 사이트를 위한 MySQL 데이터베이스가 저장되어 있습니다. 도메인 이름 www.garden.example.com을 등록하고 Route 53를 사용하여 로드 밸런서의 DNS 이름과 연결된 별칭 레코드 세트를 포함하는 호스팅 영역을 생성했습니다.



VPC로 마이그레이션하는 첫 단계는 어떤 종류의 VPC 아키텍처가 요구 사항에 부합하는지 결정하는 것입니다. 이 경우 웹 서버용 퍼블릭 서브넷 하나와 데이터베이스 서버용 프라이빗 서브넷 하나를 결정했습니다. 웹 사이트가 커지면 더 많은 웹 서버와 데이터베이스 서버를 서브넷에 추가할 수 있습니다. 기본적으로 프라이빗 서브넷의 인스턴스는 인터넷에 액세스할 수 없지만, 퍼블릭 서브넷의 NAT(Network Address Translation) 디바이스를 통해 인터넷 액세스를 활성화할 수 있습니다. 인터넷에서 데이터베이스 서버에 대한 정기 업데이트 및 패치를 지원하도록 NAT 디바이스를 설정해야 할 수 있습니다. 탄력적 IP 주소를 VPC로 마이그레이션하고 퍼블릭 서브넷에서 로드 밸런서를 생성하여 웹 서버 간 트래픽의 로드 밸런스를 유지합니다.



VPC로 웹 애플리케이션을 마이그레이션하려면 다음 단계를 따릅니다.

- VPC 생성: 이 경우 Amazon VPC 콘솔의 VPC 마법사를 사용하여 VPC와 서브넷을 생성할 수 있습니다. 두 번째 마법사 구성은 프라이빗 서브넷 하나와 퍼블릭 서브넷 하나가 있는 VPC를 생성하고, 퍼블릭 서브넷에서 NAT 디바이스를 시작하고 구성합니다. 자세한 내용은 Amazon VPC 사용 설명서의 [퍼블릭 및 프라이빗 서브넷이 있는 VPC\(NAT\)](#)를 참조하십시오.
- 보안 그룹 구성: EC2-Classic 환경에서는 웹 서버에 대한 보안 그룹이 하나 있고 데이터베이스 서버에 대한 다른 보안 그룹이 하나 있습니다. Amazon EC2 콘솔을 사용하여 각 보안 그룹에서 VPC의 새 보안 그룹으로 규칙을 복사할 수 있습니다. 자세한 내용은 [보안 그룹 \(p. 822\)](#) 단원을 참조하십시오.

Tip

다른 보안 그룹에서 참조되는 보안 그룹을 먼저 생성하십시오.

- AMI 생성 및 새 인스턴스 시작: 웹 서버 중 하나에서 AMI를 생성하고 데이터베이스 서버에서 두 번째 AMI를 생성합니다. 그런 다음 퍼블릭 서브넷으로 대체 웹 서버를 시작하고, 프라이빗 서브넷으로 대체 데이터베이스 서버를 시작합니다. 자세한 내용은 [AMI 생성 \(p. 824\)](#) 단원을 참조하십시오.
- NAT 디바이스 구성: NAT 인스턴스를 사용 중인 경우 프라이빗 서브넷에서 오는 HTTP 및 HTTPS 트래픽을 허용하는 보안 그룹을 생성해야 합니다. 자세한 내용은 [NAT 인스턴스](#)를 참조하십시오. NAT 게이트웨이를 사용 중인 경우 프라이빗 서브넷에서 오는 트래픽이 자동으로 허용됩니다.
- 데이터베이스 구성: EC2-Classic의 데이터베이스 서버에서 AMI를 생성한 경우 해당 인스턴스에 저장된 모든 구성 정보가 AMI로 복사됩니다. 새 데이터베이스 서버에 연결하고 구성 세부 정보를 업데이트해야 할 수 있습니다. 예를 들어, EC2-Classic에서 웹 서버에 전체 읽기, 쓰기 및 수정 권한을 부여하도록 데이터베이스

이스를 구성한 경우 새로운 VPC 웹 서버에 동일한 권한을 대신 부여하도록 구성 파일을 업데이트해야 합니다.

- 웹 서버 구성: 웹 서버는 EC2-Classic 인스턴스와 동일한 구성 설정을 사용합니다. 예를 들어, EC2-Classic에서 데이터베이스를 사용하도록 웹 서버를 구성한 경우 새로운 데이터베이스 인스턴스를 가리키도록 웹 서버의 구성 설정을 업데이트합니다.

Note

시작 시 다르게 지정하지 않는 한, 기본이 아닌 서브넷으로 시작한 인스턴스에는 퍼블릭 IP 주소가 기본적으로 배정되지 않습니다. 새 데이터베이스 서버에는 퍼블릭 IP 주소가 없을 수 있습니다. 이 경우 새로운 데이터베이스 서버의 프라이빗 DNS 이름을 사용하도록 웹 서버의 구성 파일을 업데이트할 수 있습니다. 동일한 VPC에 있는 인스턴스는 프라이빗 IP 주소를 통해 서로 통신 할 수 있습니다.

- 탄력적 IP 주소 마이그레이션: 탄력적 IP 주소를 EC2-Classic의 웹 서버에서 해제한 후 VPC로 마이그레이션합니다. 마이그레이션이 완료되면 탄력적 IP 주소를 VPC의 새로운 웹 서버와 연동시킵니다. 자세한 내용은 [EC2-Classic에서 탄력적 IP 주소 마이그레이션 \(p. 807\)](#) 단원을 참조하십시오.
- 새로운 로드 밸런서 생성: 계속 Elastic Load Balancing을 사용하여 인스턴스에 대한 트래픽의 로드 밸런스를 유지하려면 VPC에서 로드 밸런서를 구성할 수 있는 다양한 방법을 이해해야 합니다. 자세한 내용은 [Elastic Load Balancing 사용 설명서](#) 단원을 참조하십시오.
- DNS 레코드 업데이트: 퍼블릭 서브넷에서 로드 밸런서를 설정한 후에는 www.garden.example.com 도메인이 새로운 로드 밸런서를 가리키는지 확인해야 합니다. 이렇게 하려면 Route 53에서 DNS 레코드와 별칭 레코드 세트를 업데이트합니다. Route 53 사용에 대한 자세한 내용은 [Route 53 시작하기](#)를 참조하십시오.
- EC2-Classic 리소스 종료: 웹 애플리케이션이 VPC 아키텍처 내에서 작동하고 있는지 확인한 후 EC2-Classic 리소스를 종료하여 해당 요금이 발생하지 않도록 할 수 있습니다.

Amazon EC2의 보안

AWS에서 클라우드 보안을 가장 중요하게 생각합니다. AWS 고객은 보안에 가장 보안에 민감한 조직의 요구 사항에 부합하도록 구축된 데이터 센터 및 네트워크 아키텍처의 혜택을 누릴 수 있습니다.

보안은 AWS와 귀하의 공동 책임입니다. [공동 책임 모델](#)은 이 사항을 클라우드의 보안 및 클라우드 내 보안으로 설명합니다.

- **클라우드의 보안** – AWS는 AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호합니다. AWS는 또한 안전하게 사용할 수 있는 서비스를 제공합니다. 타사 감사자는 [AWS 규정 준수 프로그램](#)의 일환으로 보안 효과를 정기적으로 테스트하고 검증합니다. Amazon EC2에 적용되는 규정 준수 프로그램에 대해 자세히 알아보려면 [규정 준수 프로그램 제공 AWS 범위 내 서비스](#)를 참조하십시오.
- **클라우드 내 보안** – 귀하의 책임은 귀하가 사용하는 AWS 서비스에 의해 결정됩니다. 또한 귀하는 데이터의 민감도, 회사 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 Amazon EC2를 사용할 때 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 보안 및 규정 준수 목표에 맞게 Amazon EC2를 구성하는 방법을 보여줍니다. 또한 Amazon EC2 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법을 알아봅니다.

Windows Server를 실행하는 Amazon EC2의 보안 모범 사례는 [Amazon EC2 Windows 모범 사례 \(p. 21\)](#)의 보안 및 네트워크를 참조하십시오.

목차

- [Amazon EC2의 인프라 보안 \(p. 830\)](#)
- [Amazon EC2 및 인터페이스 VPC 엔드포인트 \(p. 832\)](#)
- [Amazon EC2의 복원성 \(p. 834\)](#)
- [Amazon EC2의 데이터 보호 \(p. 834\)](#)
- [Amazon EC2의 자격 증명 및 액세스 관리 \(p. 836\)](#)
- [Amazon EC2 키 페어 및 Windows 인스턴스 \(p. 900\)](#)
- [Windows 인스턴스에 대한 Amazon EC2 보안 그룹 \(p. 908\)](#)
- [Amazon EC2의 구성 관리 \(p. 924\)](#)
- [Amazon EC2의 업데이트 관리 \(p. 925\)](#)
- [Amazon EC2의 변경 관리 \(p. 925\)](#)
- [Amazon EC2의 규정 준수 확인 \(p. 925\)](#)
- [Amazon EC2의 감사 및 책임 \(p. 926\)](#)

Amazon EC2의 인프라 보안

관리형 서비스인 Amazon EC2는 [Amazon Web Services: 보안 프로세스 개요](#) 백서에 설명된 AWS 글로벌 네트워크 보안 절차로 보호됩니다.

AWS에서 개시한 API 호출을 사용하여 네트워크를 통해 Amazon EC2에 액세스합니다. 클라이언트가 TLS(전송 계층 보안) 1.0 이상을 지원해야 합니다. TLS 1.2 이상을 권장합니다. 클라이언트는 Ephemeral Diffie-Hellman(DHE) 또는 Elliptic Curve Ephemeral Diffie-Hellman(ECDHE)과 같은 PFS(전달 완전 보안, Perfect Forward Secrecy)가 포함된 암호 제품군도 지원해야 합니다. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 요청은 액세스 키 ID 및 IAM 주체와 연결된 보안 액세스 키를 사용해 서명해야 합니다. 또는 [AWS Security Token Service\(AWS STS\)](#)를 사용하여 임시 보안 자격 증명을 생성하여 요청에 서명할 수 있습니다.

네트워크 격리

Virtual Private Cloud(VPC)는 AWS 클라우드에서 논리적으로 격리된 고유한 영역의 가상 네트워크입니다. 별도의 VPC를 사용하여 워크로드별 또는 조직체별로 인프라를 격리합니다.

서브넷은 VPC의 IP 주소 범위입니다. 인스턴스를 시작할 때 VPC의 서브넷에서 인스턴스를 시작합니다. 서브넷을 사용하여 단일 VPC 내의 애플리케이션 티어(예: 웹, 애플리케이션 및 데이터베이스)를 격리합니다. 인터넷에서 직접 액세스하면 안 되는 경우 프라이빗 서브넷을 인스턴스에 사용합니다.

퍼블릭 인터넷을 통해 트래픽을 보내지 않고 VPC에서 Amazon EC2 API를 호출하려면 AWS PrivateLink를 사용합니다.

물리적 호스트에서 격리

동일한 물리적 호스트에 있는 다양한 EC2 인스턴스는 개별 물리적 호스트에 있는 것처럼 서로 격리됩니다. 하이퍼바이저는 CPU와 메모리를 격리하며, 인스턴스에는 원시 디스크 디바이스에 대한 액세스 대신 가상화된 디스크가 제공됩니다.

인스턴스를 종지하거나 종료하면 인스턴스에 할당된 메모리는 새 인스턴스에 할당되기 전에 하이퍼바이저에서 스크러빙(0으로 설정)되며 스토리지의 모든 블록은 재설정됩니다. 이 방법으로 데이터가 의도하지 않게 다른 인스턴스에 노출되지 않도록 보호합니다.

네트워크 MAC 주소는 AWS 네트워크 인프라에서 인스턴스에 동적으로 할당됩니다. IP 주소는 AWS 네트워크 인프라에서 인스턴스에 동적으로 할당되거나, 인증된 API 요청을 통해 EC2 관리자가 할당합니다. AWS 네트워크에서 인스턴스는 할당된 MAC 및 IP 주소에서만 트래픽을 전송할 수 있습니다. 그렇지 않으면 트래픽이 끊어집니다.

기본적으로 인스턴스는 특정하게 주소 지정되지 않은 트래픽을 수신할 수 없습니다. 인스턴스에서 NAT(Network Address Translation), 라우팅 또는 방화벽 서비스를 실행해야 하는 경우 네트워크 인터페이스에 대한 소스/대상 확인을 비활성화할 수 있습니다.

네트워크 트래픽 제어

EC2 인스턴스의 네트워크 트래픽을 제어하기 위해 다음 옵션을 고려해 보십시오.

- [보안 그룹 \(p. 908\)](#)을 사용하여 인스턴스에 대한 액세스를 제한합니다. Amazon EC2 인스턴스에 필요한 최소 네트워크 트래픽을 허용하고 정의된 위치, 예상 위치 및 승인된 위치에서만 액세스를 허용하도록 Amazon EC2 인스턴스 보안 그룹을 구성합니다. 예를 들어 Amazon EC2 인스턴스가 IIS 웹 서버인 경우 인바운드 HTTP/HTTPS, Windows 관리 트래픽 및 최소 아웃바운드 연결만 허용하도록 보안 그룹을 구성합니다.
- 보안 그룹을 Amazon EC2 인스턴스에 대한 네트워크 액세스를 제어하는 기본 메커니즘으로 활용합니다. 필요한 경우 네트워크 ACL을 거의 사용하지 않고 상태 비저장, 거친 네트워크 제어를 제공합니다. 보안 그룹은 상태 저장 패킷 필터링을 수행하고 다른 보안 그룹을 참조하는 규칙을 만들 수 있기 때문에 네트워크 ACL보다 다재다능합니다. 그러나 네트워크 ACL은 특정한 트래픽 하위 집합을 거부하거나 상위 수준의 서브넷 가드레일을 제공하는 보조적 제어 장치로 효과를 발휘할 수 있습니다. 또한 네트워크 ACL은 전체 서브넷에 적용되므로 인스턴스가 올바른 보안 그룹 없이 실수로 시작될 경우 이를 심층 방어 기능으로 사용할 수 있습니다.
- Windows 방화벽 설정을 GPO(그룹 정책 개체)로 중앙에서 관리하여 네트워크 제어를 더욱 강화합니다. 고객들은 종종 네트워크 트래픽을 더 잘 파악하고 보안 그룹 필터를 보완하기 위해 Windows 방화벽을 사용하며, 고급 규칙을 만들어 특정 애플리케이션이 네트워크에 액세스하지 못하도록 차단하거나 하위 집합 IP 주소에서 오는 트래픽을 필터링하기도 합니다. 예를 들어 Windows 방화벽은 화이트리스트에 있는 특정 사용자 또는 애플리케이션만 EC2 메타데이터 서비스 IP 주소에 액세스할 수 있도록 제한할 수 있습니다.

또는 공용 서비스에서 보안 그룹을 사용하여 트래픽을 특정 포트로 제한하고 Windows 방화벽을 사용하여 명시적으로 차단된 IP 주소의 블랙리스트를 유지 관리할 수 있습니다.

- Windows 인스턴스를 관리할 때는 잘 정의된 중앙 집중식 관리 서버 또는 접속 호스트에만 액세스하도록 제한하여 환경의 공격 대상 영역을 줄이십시오. 또한 SSL/TLS를 통한 RDP 캡슐화와 같은 보안 관리 프로토콜을 사용하십시오. 원격 데스크톱 게이트웨이 빠른 시작은 SSL/TLS를 사용하도록 RDP를 구성하는 등 원격 데스크톱 게이트웨이 배포의 모범 사례를 제공합니다.
- Active Directory 또는 AWS Directory Service를 사용하여 Windows 인스턴스에 대한 사용자 및 그룹의 대화식 액세스를 중앙 집중식으로 엄격하게 제어 및 모니터링하고 로컬 사용자 권한을 피할 수 있습니다. 또한 도메인 관리자를 사용하지 않고 보다 세분화된 애플리케이션별 역할 기반 계정을 만들 수 있습니다. JEA(충분한 관리)를 사용하면 대화식 액세스 또는 관리자 액세스 없이도 Windows 인스턴스의 변경 사항을 관리할 수 있습니다. 또한 JEA에서는 인스턴스 관리에 필요한 Windows PowerShell 명령의 하위 집합에 대한 관리 액세스를 잠글 수 있습니다. 자세한 내용은 [AWS 보안 모범 사례](#) 백서의 “Amazon EC2에 대한 OS 수준 액세스 관리” 단원을 참조하세요.
- 시스템 관리자는 액세스 권한이 제한된 Windows 계정을 사용하여 일상적인 작업을 수행하고, 특정한 구성 변경을 수행하는 데 필요한 경우에만 액세스 권한을 상승시켜야 합니다. 또한 절대적으로 필요한 경우에만 Windows 인스턴스에 직접 액세스합니다. 그 대신 EC2 실행 명령, SCCM(시스템 센터 구성 관리자), Windows PowerShell DSC 또는 SSM(Amazon EC2 Systems Manager)과 같은 중앙 구성 관리 시스템을 활용하여 Windows 서버에 변경 사항을 적용합니다.
- 필요한 최소한의 네트워크 경로로 Amazon VPC 서브넷 라우팅 테이블을 구성합니다. 예를 들어 인터넷 게이트웨이로 가는 경로가 있는 서브넷에는 인터넷에 직접 액세스해야 하는 Amazon EC2 인스턴스만 배치하고, 가상 프라이빗 게이트웨이로 가는 경로가 있는 서브넷에는 내부 네트워크에 직접 액세스해야 하는 Amazon EC2 인스턴스만 배치합니다.
- 보안 그룹 또는 ENI를 추가로 사용하여 일반 애플리케이션 트래픽과 별도로 Amazon EC2 인스턴스 관리 트래픽을 제어하고 감사하는 방법을 고려해 보십시오. 이렇게 하면 변경 제어를 위한 특별한 IAM 정책을 고객이 구현할 수 있으므로 보안 그룹 규칙 또는 자동화된 규칙 확인 스크립트의 변경 사항을 감사하기가 쉬워집니다. 또한 ENI가 여럿이면 호스트 기반 라우팅 정책을 생성하거나 ENI의 할당된 서브넷에 따라 다양한 VPC 서브넷 라우팅 규칙을 활용하는 기능 등 네트워크 트래픽 제어의 옵션이 늘어납니다.
- AWS Virtual Private Network 또는 AWS Direct Connect를 사용하여 원격 네트워크에서 VPC로 프라이빗 연결을 설정합니다. 자세한 내용은 [네트워크-AWS VPC 연결 옵션](#)을 참조하십시오.
- [VPC 흐름 로그](#)를 사용하여 인스턴스에 도달하는 트래픽을 모니터링합니다.
- [AWS Security Hub](#)를 사용하여 인스턴스에서 의도하지 않게 네트워크에 액세스할 수 있는지 확인합니다.
- [AWS 시스템 관리자 Session Manager](#)를 사용하여 인바운드 RDP 포트를 관리하는 대신 원격으로 인스턴스에 액세스합니다.
- [AWS 시스템 관리자 Run Command](#)를 사용하여 인바운드 RDP 포트를 관리하는 대신 공통 관리 작업을 자동화합니다.
- Windows OS 역할 및 Microsoft 비즈니스 애플리케이션은 대부분 IIS 내 IP 주소 범위 제한, Microsoft SQL Server의 TCP/IP 필터링 정책 및 Microsoft Exchange의 연결 필터 정책과 같은 향상된 기능을 제공합니다. 애플리케이션 계층 내 네트워크 제한 기능은 중요한 비즈니스 애플리케이션 서버에 대한 추가 방어 계층을 제공할 수 있습니다.

Amazon VPC에서는 각 Amazon EC2 인스턴스에 대한 네트워크 액세스를 제한하는 것 외에도 인라인 게이트웨이, 프록시 서버 및 다양한 네트워크 모니터링 옵션과 같은 네트워크 보안 제어를 추가로 구현할 수 있습니다.

자세한 내용은 [AWS 보안 모범 사례](#) 백서를 참조하십시오.

Amazon EC2 및 인터페이스 VPC 엔드포인트

인터페이스 VPC 엔드포인트를 사용하도록 Amazon EC2를 구성하여 VPC의 보안 상태를 향상 시킬 수 있습니다. 인터페이스 엔드포인트는 VPC와 Amazon EC2 사이의 모든 네트워크 트래픽을 Amazon 네트워크로 제한하여 Amazon EC2 API에 비공개로 액세스할 수 있는 AWS PrivateLink 기술로 구동됩니다. 인터페이스 엔드포인트를 사용하면 인터넷 게이트웨이, NAT 디바이스 또는 가상 프라이빗 게이트웨이가 필요 없습니다.

AWS PrivateLink를 구성하는 것이 필수는 아니지만 구성하는 것이 좋습니다. AWS PrivateLink 및 VPC 엔드포인트에 대한 자세한 내용은 [인터페이스 VPC 엔드포인트\(AWS PrivateLink\)](#)를 참조하십시오.

주제

- [인터페이스 VPC 엔드포인트 생성 \(p. 833\)](#)
- [인터페이스 VPC 엔드포인트 정책 생성 \(p. 833\)](#)

인터페이스 VPC 엔드포인트 생성

다음 서비스 이름을 사용하여 Amazon EC2에 대한 엔드포인트를 생성합니다.

- **com.amazonaws.region.ec2** — Amazon EC2 API 작업에 대한 엔드포인트를 생성합니다.

자세한 내용은 Amazon VPC 사용 설명서의 [인터페이스 엔드포인트 생성](#)을 참조하십시오.

인터페이스 VPC 엔드포인트 정책 생성

VPC 엔드포인트에 정책을 연결하여 Amazon EC2 API에 대한 액세스를 제어할 수 있습니다. 이 정책은 다음을 지정합니다.

- 작업을 수행할 수 있는 보안 주체.
- 수행할 수 있는 작업.
- 작업을 수행할 수 있는 리소스

Important

Amazon EC2에 대한 인터페이스 VPC 엔드포인트에 기본값이 아닌 정책이 적용되면 RequestLimitExceeded 실패와 같은 특정 API 요청 실패가 AWS CloudTrail 또는 Amazon CloudWatch에 기록되지 않을 수 있습니다.

자세한 내용은 Amazon VPC 사용 설명서의 [VPC 엔드포인트를 통해 서비스에 대한 액세스 제어](#)를 참조하십시오.

다음 예는 암호화되지 않은 볼륨을 생성하거나 암호화되지 않은 볼륨으로 인스턴스를 시작할 수 있는 권한을 거부하는 VPC 엔드포인트 정책을 보여 줍니다. 또한 이 정책 예에서는 다른 모든 Amazon EC2 작업을 수행할 수 있는 권한을 부여합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Action": "ec2:*",  
            "Effect": "Allow",  
            "Resource": "*",  
            "Principal": "*"  
        },  
        {  
            "Action": [  
                "ec2>CreateVolume"  
            ],  
            "Effect": "Deny",  
            "Resource": "*",  
            "Principal": "*",  
            "Condition": {  
                "Bool": {  
                    "ec2:Encrypted": "false"  
                }  
            }  
        }  
    ]  
}
```

```
        }
    },
{
  "Action": [
    "ec2:RunInstances"
  ],
  "Effect": "Deny",
  "Resource": "*",
  "Principal": "*",
  "Condition": {
    "Bool": {
      "ec2:Encrypted": "false"
    }
  }
}]}
```

Amazon EC2의 복원성

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다. 리전은 물리적으로 분리되고 격리된 다수의 가용 영역을 제공하며 이러한 가용 영역은 짧은 지연 시간, 높은 처리량 및 높은 중복성을 갖춘 네트워크를 통해 연결되어 있습니다. 가용 영역을 사용하면 중단 없이 영역 간에 자동으로 장애 조치가 이루어지 는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

더 먼 지역적 거리를 두고 데이터 또는 애플리케이션을 복제해야 하는 경우 AWS 로컬 영역을 사용하세요. AWS 로컬 영역은 사용자와 지리적으로 근접한 AWS 리전의 확장입니다. 로컬 영역은 인터넷에 대한 자체 연결을 가지고 있으며 AWS Direct Connect를 지원합니다. 모든 AWS 리전과 마찬가지로 AWS 로컬 영역은 다른 AWS 영역과 완벽히 격리되어 있습니다.

AWS 로컬 영역에서 데이터나 애플리케이션을 복제해야 하는 경우 AWS에서는 다음 영역 중 하나를 장애조치 영역으로 사용하는 것이 좋습니다.

- 다른 로컬 영역
- 상위 영역이 아닌 리전의 가용 영역입니다. [describe-availability-zones](#) 명령을 사용하여 상위 영역을 볼 수 있습니다.

AWS 리전 및 가용 영역에 대한 자세한 내용은 [AWS 글로벌 인프라](#)를 참조하십시오.

AWS 글로벌 인프라 외에도 Amazon EC2는 데이터 복원성을 지원하기 위해 다음과 같은 기능을 제공합니다.

- 리전 간 AMI 복사
- 리전 간 EBS 스냅샷 복사
- Amazon 데이터 수명 주기 관리자를 사용하여 EBS 스냅샷 자동화
- Amazon EC2 Auto Scaling을 사용하여 플랫의 상태 및 가용성 유지 관리
- Elastic Load Balancing을 사용하여 단일 가용 영역 또는 여러 가용 영역의 여러 인스턴스 간에 수신 트래픽 분산

Amazon EC2의 데이터 보호

Amazon Elastic Compute Cloud(Amazon EC2)는 AWS [공동 책임 모델](#)을 준수하며, 여기에는 데이터 보호에 대한 규정과 지침이 포함됩니다. AWS는 모든 AWS 서비스를 실행하는 글로벌 인프라를 보호할 책임이 있습니다.

니다. AWS는 고객 콘텐츠 및 개인 데이터를 처리하기 위한 보안 구성 제어를 포함하여 이 인프라에서 호스팅되는 데이터에 대한 제어를 유지합니다. 데이터 컨트롤러 또는 데이터 프로세서의 역할을 담당하는 AWS 고객과 APN 파트너는 AWS 클라우드에 입력한 모든 개인 데이터에 대해 책임이 있습니다.

데이터 보호를 위해 AWS 계정 자격 증명을 보호하고 AWS Identity and Access Management (IAM)을 사용하여 개별 사용자 계정을 설정하여 각 사용자에게 직무를 수행하는 데 필요한 권한만 부여하는 것이 좋습니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정마다 멀티 팩터 인증(MFA)을 사용합니다.
- TLS를 사용하여 AWS 리소스와 통신합니다.
- AWS CloudTrail로 API 및 사용자 활동 로깅을 설정합니다.
- AWS 암호화 솔루션을 AWS 서비스 내의 모든 기본 보안 컨트롤과 함께 사용합니다.
- Amazon S3에 저장된 개인 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용합니다.

함수 이름 및 태그와 같은 메타데이터나 자유 형식 필드에 고객 계정 번호 등의 중요 식별 정보를 절대 입력하지 마십시오. 메타데이터에 입력하는 모든 데이터는 진단 로그에 포함하기 위해 선택될 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 자격 증명 정보를 URL에 포함시키지 마십시오.

데이터 보호에 대한 자세한 내용은 AWS 보안 블로그에서 [AWS 공동 책임 모델 및 GDPR](#) 블로그 게시물을 참조하십시오.

저장된 암호화

Amazon EBS 암호화는 EBS 볼륨과 스냅샷에 대한 암호화 솔루션입니다. 이 암호화는 AWS Key Management Service(AWS KMS) 고객 마스터 키(CMK)를 사용합니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

또한 폴더 및 파일 수준 암호화에 Microsoft EFS 및 NTFS 권한을 사용할 수도 있습니다.

인스턴스의 하드웨어 모듈에 구현된 XTS-AES-256 암호를 사용하여 NVMe 인스턴스 스토어 볼륨의 데이터를 암호화합니다. 하드웨어 모듈을 사용하여 암호화 키를 생성하며, 암호화 키는 각 NVMe 인스턴스 스토리지 디바이스에 고유합니다. 인스턴스가 중지되거나 종료되면 모든 암호화 키가 손상되어 복구가 불가능해집니다. 이 암호화를 비활성화할 수 없으며, 사용자 자신의 암호화 키를 제공할 수 없습니다.

전송 중 데이터 암호화

RDP는 Windows 인스턴스에 대한 원격 액세스를 위해 보안 통신 채널을 제공합니다. AWS 시스템 관리자 Session Manager 및 Run Command를 사용하는 인스턴스에 대한 원격 액세스는 TLS 1.2를 사용하여 암호화되며, 연결을 생성하기 위한 요청은 SigV4를 사용하여 서명됩니다.

TLS(전송 계층 보안)와 같은 암호화 프로토콜을 사용하여 클라이언트와 인스턴스 간에 전송 중인 민감한 데이터를 암호화합니다.

EC2 인스턴스와 AWS API 엔드포인트 또는 기타 중요한 원격 네트워크 서비스 간에는 암호화된 연결만 허용해야 합니다. 아웃바운드 보안 그룹 또는 [Windows 방화벽](#) 규칙을 통해 이를 적용할 수 있습니다.

AWS는 모든 유형의 EC2 인스턴스 간 보안 프라이빗 연결을 제공합니다. 또한 일부 인스턴스 유형은 기본 하드웨어의 오프로드 기능을 사용하여 256비트 암호화 AEAD 알고리즘으로 인스턴스 간 전송 중 트래픽을 자동으로 암호화합니다. 네트워크 성능에는 영향을 미치지 않습니다. 추가 전송 중 트래픽 암호화를 보장하려면 다음 요구 사항을 충족해야 합니다.

- 인스턴스는 C5a, C5ad, C5n, G4, I3en, M5dn, M5n, P3dn, R5dn 및 R5n 인스턴스 유형을 사용합니다.
- 인스턴스가 동일한 리전에 있습니다.

- 인스턴스가 동일한 VPC 또는 피어링된 VPC에 있으며, 트래픽이 로드 밸런서나 전송 게이트웨이 같은 가상 네트워크 디바이스를 통과하지 않습니다.

Amazon EC2의 자격 증명 및 액세스 관리

보안 자격 증명은 AWS의 서비스에서 사용자를 식별하고 Amazon EC2 리소스와 같은 AWS 리소스의 무제한 사용을 허가하는 데 사용됩니다. Amazon EC2 및 AWS Identity and Access Management(IAM)의 기능을 사용하면 보안 자격 증명을 공유하지 않고도 다른 사용자, 서비스 및 애플리케이션에 Amazon EC2 리소스 사용을 허가할 수 있습니다. IAM을 사용하여 다른 사용자가 AWS 계정의 리소스를 사용하는 방법을 제어하고 보안 그룹을 사용하여 Amazon EC2 인스턴스에 대한 액세스를 제어할 수 있습니다. Amazon EC2 리소스의 전체 사용 또는 제한 사용을 허가할 수 있습니다.

목차

- [인스턴스에 대한 네트워크 액세스 \(p. 836\)](#)
- [Amazon EC2 권한 속성 \(p. 836\)](#)
- [IAM 및 Amazon EC2 \(p. 836\)](#)
- [Amazon EC2에 대한 IAM 정책 \(p. 838\)](#)
- [Amazon EC2의 IAM 역할 \(p. 890\)](#)
- [Windows 인스턴스의 인바운드 트래픽 권한 부여 \(p. 898\)](#)

인스턴스에 대한 네트워크 액세스

보안 그룹은 하나 이상의 인스턴스에 도달하도록 허용되는 트래픽을 제어하는 방화벽 역할을 합니다. 인스턴스를 시작할 때 하나 이상의 보안 그룹을 할당합니다. 각 보안 그룹에는 인스턴스의 트래픽을 제어하는 규칙을 추가합니다. 언제든지 보안 그룹에 대한 규칙을 수정할 수 있습니다. 새 규칙은 보안 그룹이 할당된 모든 인스턴스에 자동으로 적용됩니다.

자세한 내용은 [Windows 인스턴스의 인바운드 트래픽 권한 부여 \(p. 898\)](#) 단원을 참조하십시오.

Amazon EC2 권한 속성

조직에는 여러 AWS 계정이 있을 수 있습니다. Amazon EC2에서는 Amazon 머신 이미지(AMI) 및 Amazon EBS 스냅샷을 사용할 수 있는 추가 AWS 계정을 지정할 수 있습니다. 이러한 권한은 AWS 계정 수준으로만 적용되며, 지정된 AWS 계정에 속한 특정 사용자의 권한을 제한할 수는 없습니다. 지정한 AWS 계정의 모든 사용자가 AMI 또는 스냅샷을 사용할 수 있습니다.

각 AMI에는 AMI에 액세스할 수 있는 AWS 계정을 제어하는 `LaunchPermission` 속성이 있습니다. 자세한 내용은 [퍼블릭 AMI 설정 \(p. 75\)](#) 단원을 참조하십시오.

각 Amazon EBS 스냅샷에는 스냅샷을 사용할 수 있는 AWS 계정을 제어하는 `createVolumePermission` 속성이 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

IAM 및 Amazon EC2

IAM을 사용하여 다음을 수행할 수 있습니다.

- AWS 계정에 속하는 사용자 및 그룹 생성
- AWS 계정 사용자 각각에 고유한 보안 자격 증명 할당
- 작업 수행 시 각 사용자의 AWS 리소스 사용 권한 제어

- 다른 AWS 계정의 사용자와 AWS 리소스 공유
- AWS 계정에 적용할 규칙 생성 및 규칙을 관리할 사용자나 서비스 규정
- 엔터프라이즈의 기존 자격 증명을 사용해 AWS 리소스를 사용하는 작업 권한 허용

IAM과 Amazon EC2 함께 사용하면 조직 내 사용자별로 특정 Amazon EC2 API 작업을 사용하는 작업 수행과 특정 AWS 리소스의 사용 권한을 제어할 수 있습니다.

이 항목에서는 다음과 같은 의문 사항을 해결해 줍니다.

- IAM에서 그룹과 사용자를 생성하려면 어떻게 해야 하나요?
- 정책을 생성하려면 어떻게 해야 하나요?
- IAM에서 작업을 수행하려면 어떠한 Amazon EC2 정책이 필요한가요?
- Amazon EC2에서 작업을 수행할 수 있는 권한을 부여하려면 어떻게 해야 하나요?
- Amazon EC2의 특정 리소스에 대해 작업을 수행할 수 있는 권한을 부여하려면 어떻게 해야 하나요?

IAM 그룹 및 사용자 생성

IAM 그룹을 생성하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 그룹을 선택한 다음, 새 그룹 생성을 선택합니다.
3. 그룹 이름에서 그룹 이름을 입력한 다음, 다음 단계를 선택합니다.
4. 정책 연결 페이지에서 AWS 관리형 정책을 선택한 후 다음 단계를 선택합니다. 예를 들어 Amazon EC2의 경우 다음 AWS 관리형 정책 중 하나가 적합할 수 있습니다.
 - PowerUserAccess
 - ReadOnlyAccess
 - AmazonEC2FullAccess
 - AmazonEC2ReadOnlyAccess
5. 그룹 생성을 선택합니다.

그룹 이름 아래에 새 그룹이 나열됩니다.

IAM 사용자를 생성하고, 그룹에 사용자를 추가하고, 사용자의 암호를 생성하려면 다음을 수행합니다.

1. 탐색 창에서 사용자를 선택한 다음 사용자 추가를 선택합니다.
2. User name에 사용자 이름을 입력합니다.
3. 액세스 유형(Access type)에서 프로그래밍 방식 액세스(Programmatic access) 및 AWS Management 콘솔 액세스를 모두 선택합니다.
4. 콘솔 암호(Console password)의 경우 다음 중 하나를 선택합니다.
 - 자동 생성된 암호(Autogenerated password). 각 사용자는 현재 유효한 암호 정책(있는 경우)에 따라 임의로 생성되는 암호를 받습니다. Final(최종) 페이지에 이르면 암호를 보거나 다운로드할 수 있습니다.
 - Custom password(사용자 지정 비밀 번호). 입력란에 입력하는 암호가 각 사용자에게 할당됩니다.
5. Next: Permissions(다음: 권한)를 선택합니다.
6. 권한 설정 페이지에서 그룹에 사용자 추가를 선택합니다. 이전에 생성한 그룹 옆에 있는 확인란을 선택하고 다음: 검토(Next: Review)를 선택합니다.

7. Create user를 선택합니다.
8. 사용자의 액세스 키(액세스 키 ID와 보안 액세스 키)를 보려면 보고자 하는 각 암호와 보안 액세스 키 옆에 있는 표시를 선택합니다. 액세스 키를 저장하려면 .csv 다운로드(Download .csv)를 선택한 후 안전한 위치에 파일을 저장합니다.

Important

이 단계를 완료한 후에는 보안 액세스 키를 검색할 수 없으며, 키를 분실한 경우 새로 생성해야 합니다.

9. 닫기를 선택합니다.
10. 각 사용자에게 자격 증명(액세스 키와 암호)을 제공하여 IAM 그룹에 지정한 권한에 따라 서비스를 사용할 수 있도록 허용합니다.

관련 항목

IAM에 대한 자세한 내용은 다음을 참조하십시오.

- [Amazon EC2에 대한 IAM 정책 \(p. 838\)](#)
- [Amazon EC2의 IAM 역할 \(p. 890\)](#)
- [AWS Identity and Access Management \(IAM\)](#)
- [IAM 사용 설명서](#)

Amazon EC2에 대한 IAM 정책

기본적으로 IAM 사용자에게는 Amazon EC2 리소스를 생성 또는 수정하거나 Amazon EC2 API를 사용하여 작업을 수행할 권리가 없습니다. Amazon EC2 콘솔이나 CLI를 사용하더라도 마찬가지입니다. IAM 사용자에게 리소스 생성 또는 수정 및 작업 수행을 허용하려면 IAM 사용자에게 필요한 특정 리소스 및 API 작업을 사용할 권한을 부여하는 IAM 정책을 생성하고, 해당 권한을 필요로 하는 IAM 사용자 또는 그룹에게 정책을 연결해야 합니다.

사용자 또는 사용자 그룹에 정책을 연결하면 지정된 리소스에 대해 지정된 작업을 수행할 권리가 허용되거나 거부됩니다. IAM 정책에 대한 자세한 내용은 IAM 사용 설명서에서 [권한 및 정책](#)을 참조하십시오. 사용자 지정 IAM 정책 관리 및 생성에 대한 자세한 내용은 [IAM 정책 관리](#) 단원을 참조하십시오.

시작하기

IAM 정책은 하나 이상의 Amazon EC2 작업을 사용할 권한을 허용하거나 거부해야 합니다. 또한 작업에 사용할 수 있는 리소스를 지정해야 합니다. 모든 리소스일 수도 있고, 경우에 따라서는 특정 리소스일 수도 있습니다. 또한 정책은 리소스에 적용할 조건을 포함할 수 있습니다.

Amazon EC2에서는 리소스 수준 권한을 부분적으로 지원합니다. 즉, 일부 EC2 API 작업의 경우에는 사용자가 해당 작업에 사용할 수 있는 리소스를 별도로 지정할 수 없습니다. 대신 사용자에게 해당 작업에 대해 모든 리소스 작업을 할 수 있도록 허용해야 합니다.

작업	주제
정책의 기본적인 구조 이해	정책 구문 (p. 839)
정책의 작업 정의	Amazon EC2 작업 (p. 840)
정책의 특정 리소스 정의	Amazon EC2의 Amazon 리소스 이름 (ARN) (p. 840)

작업	주제
리소스 사용에 조건 적용	Amazon EC2의 조건 키 (p. 841)
Amazon EC2에서 사용 가능한 리소스 수준 권한 작업	Amazon EC2에 사용되는 작업, 리소스 및 조건 키(IAM 사용 설명서)
정책 테스트	사용자에게 필요한 권한이 있는지 확인 (p. 842)
CLI 또는 SDK용 예제 정책	AWS CLI 또는 AWS SDK 작업을 위한 정책 예제 (p. 845)
Amazon EC2 콘솔용 예제 정책	Amazon EC2 콘솔 작업을 위한 예제 정책 (p. 882)

정책 구조

다음 항목에서는 IAM 정책의 구조에 대해 설명합니다.

목차

- [정책 구문 \(p. 839\)](#)
- [Amazon EC2 작업 \(p. 840\)](#)
- [Amazon EC2 API 작업에 지원되는 리소스 수준 권한 \(p. 840\)](#)
- [Amazon EC2의 Amazon 리소스 이름\(ARN\) \(p. 840\)](#)
- [Amazon EC2의 조건 키 \(p. 841\)](#)
- [사용자에게 필요한 권한이 있는지 확인 \(p. 842\)](#)

정책 구문

IAM 정책은 하나 이상의 명령문으로 구성된 JSON 문서입니다. 각 명령문의 구조는 다음과 같습니다.

```
{  
    "Statement": [  
        {  
            "Effect": "effect",  
            "Action": "action",  
            "Resource": "arn",  
            "Condition": {  
                "condition": {  
                    "key": "value"  
                }  
            }  
        }  
    ]  
}
```

명령문을 이루는 요소는 다양합니다.

- **효과:** effect는 Allow 또는 Deny일 수 있습니다. 기본적으로 IAM 사용자에게는 리소스 및 API 작업을 사용할 권한이 없으므로 모든 요청이 거부됩니다. 명시적 허용은 기본 설정을 무시합니다. 명시적 거부는 모든 허용을 무시합니다.
- **Action:** action은 권한을 부여하거나 거부할 특정 API 작업입니다. 작업을 지정하는 방법에 대한 자세한 내용은 [Amazon EC2 작업 \(p. 840\)](#) 단원을 참조하십시오.
- **리소스:** 작업의 영향을 받는 리소스입니다. 일부 Amazon EC2 API 작업의 경우 작업이 생성하거나 수정 할 수 있는 리소스를 정책에 구체적으로 포함할 수 있습니다. Amazon 리소스 이름(ARN)을 사용하거나 명령문이 모든 리소스에 적용됨을 표시하는 와일드카드(*)를 사용하여 리소스를 지정합니다. 자세한 내용은 [Amazon EC2 API 작업에 지원되는 리소스 수준 권한 \(p. 840\)](#) 단원을 참조하십시오.

- Condition: Condition은 선택 사항으로서 정책이 적용되는 시점을 제어하는 데 사용할 수 있습니다. Amazon EC2에 조건을 지정하는 방법에 대한 자세한 내용은 [Amazon EC2의 조건 키 \(p. 841\)](#) 단원을 참조하십시오.

IAM용 예제 Amazon EC2 정책 명령문에 대한 자세한 내용은 [AWS CLI 또는 AWS SDK 작업을 위한 정책 예제 \(p. 845\)](#) 단원을 참조하십시오.

Amazon EC2 작업

IAM 정책 명령문에는 IAM을 지원하는 모든 서비스의 모든 API 작업을 지정할 수 있습니다. Amazon EC2의 경우 ec2: 접두사와 함께 API 작업 이름을 사용합니다. 예를 들어 ec2:RunInstances 및 ec2:CreateImage 등입니다.

명령문 하나에 여러 작업을 지정하려면 다음과 같이 쉼표로 구분합니다.

```
"Action": ["ec2:action1", "ec2:action2"]
```

와일드카드를 사용하여 여러 작업을 지정할 수도 있습니다. 예를 들어 다음과 같이 이름이 "Describe"로 시작되는 모든 작업을 지정할 수 있습니다.

```
"Action": "ec2:Describe*"
```

모든 Amazon EC2 API 작업을 지정하려면 다음과 같이 * 와일드카드를 사용합니다.

```
"Action": "ec2:/*"
```

Amazon EC2 작업의 목록은 Amazon EC2 API Reference에서 [작업](#)을 참조하십시오.

Amazon EC2 API 작업에 지원되는 리소스 수준 권한

리소스 수준 권한은 사용자가 작업을 수행할 수 있는 리소스를 지정할 수 있는 기능입니다. Amazon EC2는 리소스 수준 권한을 부분적으로 지원합니다. 즉, 필요 조건을 지정하거나 사용 가능한 특정 리소스를 지정하여 사용자가 특정 Amazon EC2 작업을 사용할 수 있는지 여부를 제어할 수 있습니다. 예를 들어 사용자에게 인스턴스 시작 권한을 부여하면서 특정 유형 또는 특정 AMI만 사용하도록 제한할 수 있습니다.

IAM 정책 설명에서 리소스를 지정하려면 Amazon 리소스 이름(ARN)을 사용합니다. ARN 값을 지정하는 방법에 대한 자세한 내용은 [Amazon EC2의 Amazon 리소스 이름\(ARN\) \(p. 840\)](#) 단원을 참조하십시오. API 작업이 개별 ARN을 지원하지 않는 경우 와일드카드(*)를 사용하여 모든 리소스가 작업의 영향을 받을 수 있도록 지정해야 합니다.

리소스 수준 권한을 지원하는 Amazon EC2 API 작업과, 정책에서 사용할 수 있는 ARN 및 조건 키를 식별하는 테이블을 보려면 IAM 사용 설명서의 [Amazon EC2에 사용되는 작업, 리소스 및 조건 키](#)를 참조하십시오. Amazon EC2에 대한 조건 키는 이후 단원에서 자세히 설명합니다.

Amazon EC2 API 작업에 사용하는 IAM 정책에는 태그 기반의 리소스 수준 권한을 적용할 수 있습니다. 이를 통해 사용자가 생성, 수정 또는 사용할 수 있는 리소스를 더욱 정확하게 제어할 수 있습니다. 자세한 내용은 [생성 시 리소스 태그 지정에 대한 권한 부여 \(p. 843\)](#) 단원을 참조하십시오.

Amazon EC2의 Amazon 리소스 이름(ARN)

각 IAM 정책 명령문은 ARN을 사용하여 지정한 리소스에 적용됩니다.

ARN의 일반적인 구문은 다음과 같습니다.

```
arn:aws:[service]:[region]:[account]:resourceType/resourcePath
```

service

서비스(예: ec2)입니다.

region

리소스의 리전(예: us-east-1)입니다.

account

AWS 계정 ID이며 하이픈은 제외합니다(예: 123456789012).

resourceType

리소스의 유형(예: instance)입니다.

resourcePath

리소스를 식별하는 경로입니다. 경로에 * 와일드카드를 사용할 수 있습니다.

예를 들어 명령문에서 다음과 같이 ARN을 사용하여 특정 인스턴스(i-1234567890abcdef0)를 나타낼 수 있습니다.

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0"
```

다음과 같이 * 와일드카드를 사용하여 특정 계정에 속하는 모든 인스턴스를 지정할 수도 있습니다.

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:instance/*"
```

다음과 같이 * 와일드카드를 사용하여 특정 계정에 속하는 모든 Amazon EC2 리소스를 지정할 수도 있습니다.

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:*
```

모든 리소스를 지정해야 하거나 특정 API 작업이 ARN을 지원하지 않는 경우 다음과 같이 Resource 요소에 * 와일드카드를 사용합니다.

```
"Resource": "*"
```

다양한 Amazon EC2 API 작업에는 여러 리소스가 관여합니다. 예를 들어 AttachVolume은 Amazon EBS 볼륨을 인스턴스에 연결하므로 IAM 사용자에게 볼륨 사용 권한과 인스턴스 사용 권한이 있어야 합니다. 단일 명령문에서 여러 리소스를 지정하려면 다음과 같이 각 ARN을 쉼표로 구분합니다.

```
"Resource": ["arn1", "arn2"]
```

Amazon EC2 리소스에 대한 ARN 목록은 IAM 사용 설명서의 [Amazon EC2에서 정의한 리소스 유형](#)을 참조하십시오.

Amazon EC2의 조건 키

정책 명령문에서 정책이 적용되는 시점을 제어하는 조건을 지정할 수 있습니다. 각 조건에는 하나 이상의 키-값 페어가 포함됩니다. 조건 키는 대/소문자를 구분하지 않습니다. AWS 전체 범위 조건 키 및 추가적인 서비스별 조건 키가 정의되어 있습니다.

Amazon EC2에 대한 서비스별 조건 키 목록은 IAM 사용 설명서에서 [Amazon EC2의 조건 키](#)를 참조하십시오. Amazon EC2는 AWS 전체 범위 조건 키도 구현합니다. 자세한 내용은 IAM 사용 설명서의 [모든 요청에서 사용 가능한 정보](#)를 참조하십시오.

IAM 정책에 조건 키를 사용하려면 **Condition** 명령문을 사용합니다. 예를 들어 다음 정책은 사용자에게 임의의 보안 그룹에 대한 인바운드 및 아웃바운드 규칙을 추가하고 제거하는 권한을 부여합니다. **ec2:vpc** 조건 키를 사용하여 특정 VPC의 보안 그룹에서만 이러한 작업을 수행할 수 있도록 지정합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [ {  
        "Effect": "Allow",  
        "Action": [  
            "ec2:AuthorizeSecurityGroupIngress",  
            "ec2:AuthorizeSecurityGroupEgress",  
            "ec2:RevokeSecurityGroupIngress",  
            "ec2:RevokeSecurityGroupEgress"],  
        "Resource": "arn:aws:ec2:region:account:security-group/*",  
        "Condition": {  
            "StringEquals": {  
                "ec2:Vpc": "arn:aws:ec2:region:account:vpc/vpc-11223344556677889"  
            }  
        }  
    }]  
}
```

여러 조건을 지정하거나 조건 하나에 여러 키를 지정하는 경우 논리적 AND 연산을 적용하여 평가합니다. 조건 하나에서 키 하나에 여러 값을 지정하면 논리적 OR 연산자를 적용하여 조건을 평가합니다. 모든 조건이 충족되어야 권한이 부여됩니다.

조건을 지정할 때 자리표 시자를 사용할 수도 있습니다. 예를 들어 IAM 사용자 이름을 지정하는 태그가 포함된 리소스를 사용할 IAM 사용자 권한을 부여할 수 있습니다. 자세한 내용은 IAM 사용 설명서에서 **정책 변수**를 참조하십시오.

Important

여러 조건 키들이 하나의 리소스에 달려 있고, 일부 API 작업은 다수의 리소스를 사용합니다. 조건 키로 정책을 작성하는 경우에는 설명의 **Resource** 요소를 이용해 조건 키가 적용되는 리소스를 지정하십시오. 그렇게 하지 않으면, 조건 키가 해당되지 않는 리소스에 대해서는 조건 검사가 실패하여 정책이 사용자로 하여금 작업을 전혀 수행하지 못하게 막을 수도 있습니다. 리소스를 지정하고 싶지 않거나 다수의 API 작업을 포함하도록 정책의 **Action** 요소를 작성했다면, 반드시 ...**IfExists** 조건 유형을 이용해 조건 키가 그것을 사용하지 않는 리소스에 대해서는 무시되도록 해야 합니다. 자세한 내용은 IAM 사용 설명서의 **...IfExists 조건** 단원을 참조하십시오.

모든 Amazon EC2 작업은 **aws:RequestedRegion** 및 **ec2:Region** 조건 키를 지원합니다. 자세한 내용은 [예제: 특정 리전으로 액세스 제한 \(p. 846\)](#) 단원을 참조하십시오.

ec2:SourceInstanceARN 키는 요청이 이루어진 인스턴스의 ARN을 지정하는 조건에 사용할 수 있습니다. 이 조건 키는 AWS 전체 범위에서 사용 가능하고 서비스에 특정하지 않습니다. 정책 예제는 [EC2 인스턴스가 볼륨을 연결 또는 분리하도록 허용](#) 및 [예제: 특정 인스턴스가 기타 AWS 서비스에서 리소스를 확인하도록 허용 \(p. 878\)](#)을 참조하십시오. **ec2:SourceInstanceARN** 키는 명령문에서 **Resource** 요소에 대한 ARN을 채우는 변수로 사용할 수 없습니다.

Amazon EC2용 예제 정책 명령문은 [AWS CLI 또는 AWS SDK 작업을 위한 정책 예제 \(p. 845\)](#) 단원을 참조하십시오.

사용자에게 필요한 권한이 있는지 확인

IAM 정책을 생성한 후에는 사용자에게 필요한 특정 API 작업 및 리소스를 사용할 권한이 제대로 부여되는지를 확인한 후에 정책을 실무에 적용하는 것이 좋습니다.

우선 테스트용으로 IAM 사용자를 생성하고 앞서 생성한 IAM 정책을 연결하여 사용자를 테스트합니다. 그런 다음 테스트 사용자 자격으로 요청을 수행합니다.

리소스를 생성하거나 수정하는 Amazon EC2 작업을 테스트하는 경우 DryRun 파라미터를 사용하여 요청하거나 --dry-run 옵션과 함께 AWS CLI 명령을 실행해야 합니다. 이렇게 하면 호출 시 권한 부여 확인은 완료되지만 작업은 완료되지 않습니다. 예를 들어 인스턴스를 실제로 종료하지 않고 사용자가 특정 인스턴스를 종료할 수 있는지 여부를 확인할 수 있습니다. 테스트 사용자에게 필요한 권한이 있는 경우 요청 시 DryRunOperation이 반환되고, 그렇지 않은 경우 UnauthorizedOperation이 반환됩니다.

정책이 사용자에게 정상적으로 권한을 부여하지 못하거나 권한을 과도하게 부여하는 경우, 원하는 결과가 나올 때까지 정책을 조정하고 다시 테스트할 수 있습니다.

Important

변경된 정책이 전파되어 효력을 발휘하려면 몇 분이 걸릴 수 있습니다. 따라서 정책을 업데이트한 경우 5분간 기다린 후에 테스트하는 것이 좋습니다.

요청 시 권한 부여 확인에 실패하면 진단 정보가 포함된 인코딩 메시지가 반환됩니다. DecodeAuthorizationMessage 작업을 사용하여 메시지를 디코딩할 수 있습니다. 자세한 내용은 AWS Security Token Service API Reference의 [DecodeAuthorizationMessage](#) 및 AWS CLI Command Reference의 [decode-authorization-message](#) 단원을 참조하십시오.

생성 시 리소스 태그 지정에 대한 권한 부여

일부 리소스 생성 Amazon EC2 API 작업에서는 리소스를 생성할 때 태그를 지정할 수 있습니다. 자세한 내용은 [리소스에 태그 지정 \(p. 1144\)](#) 단원을 참조하십시오.

사용자가 생성 시 리소스에 태그를 지정할 수 있으려면 리소스를 생성하는 작업을 사용할 권한이 있어야 합니다(예: ec2:RunInstances 또는 ec2:CreateVolume). 리소스 생성 작업에서 태그가 지정되면 Amazon은 ec2:CreateTags 작업에서 추가 권한 부여를 수행해 사용자에게 태그를 생성할 권한이 있는지 확인합니다. 따라서 사용자는 ec2:CreateTags 작업을 사용할 명시적 권한도 가지고 있어야 합니다.

ec2:CreateTags 작업에 대한 IAM 정책 정의에서 ec2:CreateAction 조건 키와 함께 Condition 요소를 사용하여 리소스를 만드는 작업에 태그 지정 권한을 부여합니다.

다음 예제의 정책은 사용자가 인스턴스를 시작하고 시작 도중 인스턴스와 볼륨에 임의의 태그를 적용하는 것을 허용합니다. 사용자는 기존 리소스에 태그를 지정할 수 없습니다(ec2:CreateTags 작업을 직접 호출할 수 없습니다).

```
{  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateTags"  
            ],  
            "Resource": "arn:aws:ec2:region:account:/*/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:CreateAction" : "RunInstances"  
                }  
            }  
        }  
    ]  
}
```

마찬가지로 다음 정책은 사용자가 볼륨을 생성하고 볼륨 생성 도중 볼륨에 임의의 태그를 적용하는 것을 허용합니다. 사용자는 기존 리소스에 태그를 지정할 수 없습니다(ec2:CreateTags 작업을 직접 호출할 수 없습니다).

```
{  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateVolume"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateTags"  
            ],  
            "Resource": "arn:aws:ec2:region:account:*//*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:CreateAction" : "CreateVolume"  
                }  
            }  
        }  
    ]  
}
```

ec2:CreateTags 작업은 리소스 생성 작업 도중 태그가 적용되는 경우에만 평가됩니다. 따라서 리소스를 생성할 권한이 있는 사용자(태그 지정 조건은 없다고 가정)는 요청에서 태그가 지정되지 않은 경우, ec2:CreateTags 작업을 사용할 권한이 필요하지 않습니다. 하지만 사용자가 태그를 사용하여 리소스 생성을 시도하는 경우, 사용자에게 ec2:CreateTags 작업을 사용할 권한이 없다면 요청은 실패합니다.

시작 템플릿에 태그가 제공되는 경우 ec2:CreateTags 작업도 평가됩니다. 정책에 대한 예는 [시작 템플릿의 태그 \(p. 866\)](#)을 참조하십시오.

특정 태그에 대한 액세스 제어

IAM 정책의 Condition 요소에 추가 조건을 사용하여 리소스에 적용할 수 있는 태그 키와 값을 제어할 수 있습니다.

앞 단원의 예제에 다음 조건 키를 사용할 수 있습니다.

- aws:RequestTag: 특정 태그 키 또는 태그 키 및 값이 요청에 존재해야 함을 표시. 요청에서 다른 태그도 지정할 수 있습니다.
- 특정한 태그와 키 및 가치의 조합을 적용하려면(예를 들어 태그 StringEquals=cost-center:를 적용하려면) cc123 조건 연산자와 함께 사용하십시오.

```
"StringEquals": { "aws:RequestTag/cost-center": "cc123" }
```

- 요청에서 특정 태그 키를 적용하려면(예를 들어 태그 키 StringLike:를 적용하려면) purpose 조건 연산자와 함께 사용하십시오.

```
"StringLike": { "aws:RequestTag/purpose": "*" }
```

- aws:TagKeys: 요청에서 사용되는 태그 키를 적용.
- 요청 시 지정하려면 ForAllValues 변경자와 함께 특정 태그 키를 적용하십시오(요청에서 태그가 지정되면 특정 태그 키만 허용되고 다른 태그는 허용되지 않습니다). 예를 들어 태그 키 environment 또는 cost-center가 허용됩니다.

```
"ForAllValues:StringEquals": { "aws:TagKeys": ["environment", "cost-center"] }
```

- 요청에서 지정된 태그 키 중 최소한 1개의 존재를 적용하려면 `ForAnyValue` 변경자와 함께 사용하십시오. 예를 들어 요청에 태그 키 `environment` 또는 `webserver` 중 최소한 1개가 존재해야 합니다.

```
"ForAnyValue:StringEquals": { "aws:TagKeys": ["environment", "webserver"] }
```

이들 조건 키는 `ec2:CreateTags` 및 `ec2:DeleteTags` 작업뿐 아니라 태그 지정을 지원하는 리소스 생성 작업에 적용될 수 있습니다. Amazon EC2 API 작업에서 태그 지정을 지원하는지 알아보려면 IAM 사용 설명서의 [Amazon EC2에 사용되는 작업, 리소스 및 조건 키](#)를 참조하십시오.

사용자가 리소스를 생성할 때 강제로 태그를 지정하도록 하려면 리소스 생성 작업에서 `aws:RequestTag` 조건 키 또는 `aws:TagKeys` 조건 키를 `ForAnyValue` 변경자와 함께 사용해야 합니다. 이때 사용자가 리소스 생성 시 태그를 지정하지 않으면 `ec2:CreateTags` 작업이 평가되지 않습니다.

조건의 경우 조건 키는 대소문자를 구분하지 않고 조건 값은 대소문자를 구분합니다. 따라서 태그 키의 대소 문자 구별을 설정하려면 태그 키가 조건의 값으로 지정된 `aws:TagKeys` 조건 키를 사용합니다.

예제 IAM 정책은 [AWS CLI 또는 AWS SDK 작업을 위한 정책 예제 \(p. 845\)](#) 단원을 참조하십시오. 다중 값 조건에 대한 자세한 내용은 IAM 사용 설명서의 [다중 키 값을 테스트하는 조건 생성](#) 단원을 참조하십시오.

리소스 태그를 사용하여 EC2 리소스에 대한 액세스 제어

IAM 사용자에게 EC2 리소스 사용 권한을 부여하는 IAM 정책을 생성할 때 정책의 `Condition` 요소에 태그 정보를 포함시키면 태그를 기반으로 액세스를 제어할 수 있습니다. 이를 통해 사용자가 수정, 사용 또는 삭제 할 수 있는 EC2 리소스를 더욱 정확하게 제어할 수 있습니다.

예를 들어 사용자가 인스턴스를 종료할 수 있도록 허용하지만 인스턴스에 `environment=production` 태그가 있는 경우 작업을 거부하는 정책을 만들 수 있습니다. 이렇게 하려면 `ec2:ResourceTag` 조건 키를 사용하여 리소스에 연결된 태그를 기반으로 리소스에 대한 액세스를 허용하거나 거부합니다.

```
"StringEquals": { "ec2:ResourceTag/environment": "production" }
```

Amazon EC2 API 작업에서 `ec2:ResourceTag` 조건 키를 사용한 액세스 제어를 지원하는지 알아보려면 IAM 사용 설명서의 [Amazon EC2에 사용되는 작업, 리소스 및 조건 키](#)를 참조하십시오. `Describe` 작업은 리소스 수준 권한을 지원하지 않기 때문에 조건 없이 별도의 명령문에 지정해야 합니다.

예제 IAM 정책은 [AWS CLI 또는 AWS SDK 작업을 위한 정책 예제 \(p. 845\)](#) 단원을 참조하십시오.

Note

태그를 기준으로 리소스에 대한 사용자 액세스를 허용 또는 거부하는 경우 동일한 리소스에서 태그를 추가 또는 제거할 수 있도록 사용자를 명시적으로 거부할 것을 고려해야 합니다. 그렇지 않으면 사용자가 제한을 피해 태그를 수정하여 리소스에 대한 액세스 권한을 얻을 수 있습니다.

AWS CLI 또는 AWS SDK 작업을 위한 정책 예제

다음 예제는 IAM 사용자가 갖는 Amazon EC2 관련 권한을 제어하는 데 사용할 수 있는 정책 명령문을 보여 줍니다. 이러한 정책은 AWS CLI 또는 AWS SDK를 통한 요청에 맞게 설계되었습니다. Amazon EC2 콘솔 작업과 관련된 예제 정책은 [Amazon EC2 콘솔 작업을 위한 예제 정책 \(p. 882\)](#) 단원을 참조하십시오. Amazon VPC와 관련된 IAM 정책의 예는 [Amazon VPC에 대한 Identity and Access Management](#)을 참조하십시오.

예제:

- 예: 읽기 전용 액세스 (p. 846)
- 예제: 특정 리전으로 액세스 제한 (p. 846)
- 인스턴스 작업 (p. 847)
- 볼륨 작업 (p. 848)
- 스냅샷 작업 (p. 851)
- 인스턴스 시작(RunInstances) (p. 858)
- 스팟 인스턴스 작업 (p. 869)
- 예제: 예약 인스턴스 작업 (p. 874)
- 예제: 리소스에 태그 지정 (p. 874)
- 예제: IAM 역할 작업 (p. 876)
- 예제: 라우팅 테이블 작업 (p. 877)
- 예제: 특정 인스턴스가 기타 AWS 서비스에서 리소스를 확인하도록 허용 (p. 878)
- 예제: 시작 템플릿 사용 (p. 878)
- 인스턴스 메타데이터 작업 (p. 879)

예: 읽기 전용 액세스

다음 정책은 이름이 `Describe`로 시작되는 모든 Amazon EC2 API 작업을 사용할 권한을 부여합니다. Resource 요소에 와일드카드가 사용되었으므로 사용자가 이러한 API 작업에 모든 리소스를 지정할 수 있습니다. API 작업이 리소스 수준 권한을 지원하지 않는 경우에도 * 와일드카드가 필요합니다. Amazon EC2 API 작업에 사용할 수 있는 ARN에 대한 자세한 내용은 IAM 사용 설명서의 [Amazon EC2에 사용되는 작업, 리소스 및 조건 키](#)를 참조하십시오.

다른 명령문으로 해당 권한을 부여하지 않는 경우 리소스에 대해 작업을 수행할 권한은 부여되지 않습니다. 해당 API 작업을 사용할 권한은 기본적으로 거부됩니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:Describe*",  
            "Resource": "*"  
        }  
    ]  
}
```

예제: 특정 리전으로 액세스 제한

다음 정책은 유럽(프랑크푸르트) 리전이 아닌 경우 모든 Amazon EC2 API 작업을 사용할 수 있는 사용자 권한을 거부합니다. 모든 Amazon EC2 API 작업에서 지원하는 전역 조건 키 `aws:RequestedRegion`을 사용합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Deny",  
            "Action": "ec2:*",  
            "Resource": "*",  
            "Condition": {  
                "StringNotEquals": {  
                    "aws:RequestedRegion": "eu-central-1"  
                }  
            }  
        }  
    ]  
}
```

```
        }
    ]
}
```

또는 Amazon EC2에 고유하고 모든 Amazon EC2 API 작업에서 지원하는 조건 키 `ec2:Region`을 사용할 수 있습니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "ec2:*",
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {
          "ec2:Region": "eu-central-1"
        }
      }
    }
  ]
}
```

인스턴스 작업

예제:

- 예제: 모든 인스턴스를 실행, 중지, 시작 및 종료 (p. 847)
- 예제: 모든 인스턴스를 설명할 수 있지만 특정 인스턴스만 중지, 시작 및 종료 (p. 848)

예제: 모든 인스턴스를 실행, 중지, 시작 및 종료

다음 정책은 사용자에게 Action 요소에 지정된 API 작업을 사용할 권한을 부여합니다. Resource 요소에 * 와일드카드가 사용되었으므로 사용자가 이러한 API 작업에 모든 리소스를 지정할 수 있습니다. API 작업이 리소스 수준 권한을 지원하지 않는 경우에도 * 와일드카드가 필요합니다. Amazon EC2 API 작업에 사용할 수 있는 ARN에 대한 자세한 내용은 IAM 사용 설명서의 [Amazon EC2에 사용되는 작업, 리소스 및 조건 키](#)를 참조하십시오.

다른 명령문으로 해당 권한을 부여하지 않는 경우 다른 API 작업을 사용할 권한은 부여되지 않습니다. 해당 API 작업을 사용할 권한은 기본적으로 거부됩니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeInstances",
        "ec2:DescribeImages",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeAvailabilityZones",
        "ec2:RunInstances",
        "ec2:TerminateInstances",
        "ec2:StopInstances",
        "ec2:StartInstances"
      ],
      "Resource": "*"
    }
  ]
}
```

}

예제: 모든 인스턴스를 설명할 수 있지만 특정 인스턴스만 중지, 시작 및 종료

다음 정책은 모든 인스턴스를 설명하고, 인스턴스 i-1234567890abcdef0 및 i-0598c7d356eba48d7만 시작 및 종지하고, 리소스 태그가 "미국 동부(버지니아 북부) 지역"인 us-east-1의 인스턴스(purpose=test)만 종료하도록 허용합니다.

첫 번째 명령문의 Resource 요소에 * 와일드카드가 사용되었으므로 사용자가 작업에 모든 리소스를 지정할 수 있습니다. 여기에서는 모든 인스턴스를 나열할 수 있습니다. API 작업(여기에서는 ec2:DescribeInstances)이 리소스 수준 권한을 지원하지 않는 경우에도 * 와일드카드가 필요합니다. Amazon EC2 API 작업에 사용할 수 있는 ARN에 대한 자세한 내용은 IAM 사용 설명서의 [Amazon EC2에 사용되는 작업, 리소스 및 조건 키](#)를 참조하십시오.

두 번째 명령문의 StopInstances 및 StartInstances 작업에는 리소스 수준 권한이 사용되었습니다. Resource 요소의 ARN에 의해 특정 인스턴스가 지정되었습니다.

세 번째 명령문은 사용자가 지정된 AWS 계정에 속하며 미국 동부(버지니아 북부) 지역 태그를 갖는 us-east-1의 모든 인스턴스("purpose=test")를 종료하도록 허용합니다. Condition 요소는 정책 명령문 적용 시에 평가됩니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:DescribeInstances",  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:StopInstances",  
                "ec2:StartInstances"  
            ],  
            "Resource": [  
                "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0",  
                "arn:aws:ec2:us-east-1:123456789012:instance/i-0598c7d356eba48d7"  
            ]  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:TerminateInstances",  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:instance/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:ResourceTag/purpose": "test"  
                }  
            }  
        }  
    ]  
}
```

볼륨 작업

예제:

- 예제: 볼륨 연결 및 분리 (p. 849)
- 예제: 볼륨 생성 (p. 849)
- 예제: 태그를 사용하여 볼륨 생성 (p. 850)

예제: 볼륨 연결 및 분리

API 작업의 호출자가 여러 리소스를 지정해야 하는 경우 사용자가 필요한 모든 리소스에 액세스하도록 허용하는 정책 명령문을 생성해야 합니다. 이러한 리소스가 하나 이상 포함된 `Condition` 요소를 사용해야 하는 경우 이 예제와 같이 여러 명령문을 생성해야 합니다.

다음 정책은 "volume_user=iam-user-name" 태그가 있는 볼륨을 "department=dev" 태그가 있는 인스턴스에 연결하고 해당 볼륨을 해당 인스턴스에서 분리하도록 허용합니다. IAM 그룹에 이 정책을 연결하면 `aws:username` 정책 변수가 그룹의 각 IAM 사용자에게 자신의 `volume_user` 사용자 이름을 값으로 하는 IAM라는 태그가 있는 인스턴스에 볼륨을 연결하거나 분리할 권한을 부여합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:AttachVolume",  
                "ec2:DetachVolume"  
            ],  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:instance/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:ResourceTag/department": "dev"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:AttachVolume",  
                "ec2:DetachVolume"  
            ],  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:volume/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:ResourceTag/volume_user": "${aws:username}"  
                }  
            }  
        }  
    ]  
}
```

예제: 볼륨 생성

다음 정책은 사용자가 [CreateVolume](#) API 작업을 사용하는 것을 허용합니다. 사용자는 볼륨이 암호화되고 볼륨 크기가 20GB 미만인 경우에만 볼륨을 생성할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2>CreateVolume"  
            ],  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:volume/*",  
            "Condition": {  
                "NumericLessThan": {  
                    "ec2:VolumeSize" : "20"  
                },  
                "Bool": {  
                    "ec2:Encrypted" : "true"  
                }  
            }  
        }  
    ]  
}
```

```
        }
    }
}
```

예제: 태그를 사용하여 볼륨 생성

다음 정책에는 사용자가 태그 aws:RequestTag 및 costcenter=115를 사용하여 생성하는 볼륨에 태그를 지정해야 하는 stack=prod 조건 키가 포함됩니다. aws:TagKeys 조건 키는 ForAllValues 변경자를 사용하여 요청에서 키 costcenter 및 stack만 허용됨을 표시합니다(다른 어떤 태그도 지정할 수 없습니다). 사용자가 이 특정 키들을 전달하지 않거나 태그를 전혀 지정하지 않으면 요청은 실패합니다.

태그를 적용하는 리소스 생성 작업의 경우, 사용자가 createTags 작업을 사용할 권한도 가지고 있어야 합니다. 두 번째 문은 ec2:CreateAction 조건 키를 사용하여 사용자가 CreateVolume의 컨텍스트에서만 태그를 생성하도록 허용합니다. 사용자는 기존의 볼륨이나 다른 어떤 리소스에도 태그를 지정할 수 없습니다. 자세한 내용은 [생성 시 리소스 태그 지정에 대한 권한 부여 \(p. 843\)](#) 단원을 참조하십시오.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCreateTaggedVolumes",
      "Effect": "Allow",
      "Action": "ec2:CreateVolume",
      "Resource": "arn:aws:ec2:us-east-1:123456789012:volume/*",
      "Condition": {
        "StringEquals": {
          "aws:RequestTag/costcenter": "115",
          "aws:RequestTag/stack": "prod"
        },
        "ForAllValues:StringEquals": {
          "aws:TagKeys": ["costcenter", "stack"]
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:123456789012:volume/*",
      "Condition": {
        "StringEquals": {
          "ec2:CreateAction" : "CreateVolume"
        }
      }
    }
  ]
}
```

다음 정책은 사용자가 태그를 지정하지 않고 볼륨을 생성하는 것을 허용합니다. CreateTags 작업은 CreateVolume 요청에서 태그가 지정되는 경우에만 평가됩니다. 사용자가 태그를 지정하는 경우, 태그는 purpose=test여야 합니다. 다른 어떤 태그도 요청에서 허용되지 않습니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:CreateVolume",
      "Resource": "*"
    }
  ]
}
```

```
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:us-east-1:1234567890:volume/*",
    "Condition": {
        "StringEquals": {
            "aws:RequestTag/purpose": "test",
            "ec2:CreateAction" : "CreateVolume"
        },
        "ForAllValues:StringEquals": {
            "aws:TagKeys": "purpose"
        }
    }
}
]
```

스냅샷 작업

다음은 CreateSnapshot(EBS 볼륨의 특정 시점 스냅샷)과 CreateSnapshots(다중 볼륨 스냅샷)에 대한 정책 예제입니다.

예제:

- 예제: 스냅샷 생성 (p. 851)
- 예제: 스냅샷 생성 (p. 852)
- 예제: 태그를 사용하여 스냅샷 생성 (p. 852)
- 예제: 태그를 사용하여 스냅샷 생성 (p. 853)
- 예제: 스냅샷 권한 설정 수정 (p. 857)

예제: 스냅샷 생성

다음 정책은 고객이 CreateSnapshot API 작업을 사용하는 것을 허용합니다. 고객은 볼륨이 암호화되고 볼륨 크기가 20GiB 미만인 경우에만 스냅샷을 생성할 수 있습니다.

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect":"Allow",
            "Action":"ec2:CreateSnapshot",
            "Resource":"arn:aws:ec2:us-east-1::snapshot/*"
        },
        {
            "Effect":"Allow",
            "Action":"ec2:CreateSnapshot",
            "Resource":"arn:aws:ec2:us-east-1:123456789012:volume/*",
            "Condition":{
                "NumericLessThan":{
                    "ec2:VolumeSize":"20"
                },
                "Bool":{
                    "ec2:Encrypted":"true"
                }
            }
        }
    ]
}
```

예제: 스냅샷 생성

다음 정책은 고객이 [CreateSnapshots](#) API 작업을 사용하는 것을 허용합니다. 고객은 인스턴스의 모든 볼륨이 GP2 유형인 경우에만 스냅샷을 생성할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateSnapshots",  
            "Resource": [  
                "arn:aws:ec2:us-east-1::snapshot/*",  
                "arn:aws:ec2:*::instance/*"  
            ]  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateSnapshots",  
            "Resource": "arn:aws:ec2:us-east-1::*:volume/*",  
            "Condition": {  
                "StringLikeIfExists": {  
                    "ec2:VolumeType": "gp2"  
                }  
            }  
        }  
    ]  
}
```

예제: 태그를 사용하여 스냅샷 생성

다음 정책에는 고객이 태그 `aws:RequestTag` 및 `costcenter=115`를 모든 새로운 스냅샷에 적용해야 하는 `stack=prod` 조건 키가 포함됩니다. `aws:TagKeys` 조건 키는 `ForAllValues` 변경자를 사용하여 요청에서 키 `costcenter` 및 `stack`만 지정될 수 있음을 표시합니다. 조건 중 하나라도 충족하지 않으면 요청은 실패합니다.

태그를 적용하는 리소스 생성 작업의 경우, 고객이 `CreateTags` 작업을 사용할 권한도 가지고 있어야 합니다. 세 번째 문은 `ec2:CreateAction` 조건 키를 사용하여 고객이 `CreateSnapshot`의 컨텍스트에서만 태그를 생성하도록 허용합니다. 고객은 기존의 볼륨이나 다른 어떤 리소스에도 태그를 지정할 수 없습니다. 자세한 내용은 [생성 시 리소스 태그 지정에 대한 권한 부여](#) (p. 843) 단원을 참조하십시오.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateSnapshot",  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:volume/*"  
        },  
        {  
            "Sid": "AllowCreateTaggedSnapshots",  
            "Effect": "Allow",  
            "Action": "ec2:CreateSnapshot",  
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:RequestTag/costcenter": "115",  
                    "aws:RequestTag/stack": "prod"  
                },  
                "ForAllValues:StringEquals": {  
                    "aws:TagKeys": [  
                        "costcenter",  
                        "stack"  
                    ]  
                }  
            }  
        }  
    ]  
}
```

```
        "stack"
    ],
}
},
{
    "Effect": "Allow",
    "Action": "ec2:CreateTags",
    "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
    "Condition": {
        "StringEquals": {
            "ec2:CreateAction": "CreateSnapshot"
        }
    }
}
]
```

예제: 태그를 사용하여 스냅샷 생성

다음 정책에는 고객이 태그 aws:RequestTag 및 costcenter=115를 모든 새로운 스냅샷에 적용해야 하는 stack=prod 조건 키가 포함됩니다. aws:TagKeys 조건 키는 ForAllValues 변경자를 사용하여 요청에서 키 costcenter 및 stack만 지정될 수 있음을 표시합니다. 조건 중 하나라도 충족하지 않으면 요청은 실패합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshots",
            "Resource": [
                "arn:aws:ec2:us-east-1::snapshot/*",
                "arn:aws:ec2:*:*:instance/*",
                "arn:aws:ec2:*:*:volume/*"
            ]
        },
        {
            "Sid": "AllowCreateTaggedSnapshots",
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshots",
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/costcenter": "115",
                    "aws:RequestTag/stack": "prod"
                },
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": [
                        "costcenter",
                        "stack"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": "CreateSnapshots"
                }
            }
        }
    ]
}
```

```
        }
    ]
}
```

다음 정책은 고객이 태그를 지정하지 않고 스냅샷을 생성하는 것을 허용합니다. CreateTags 작업은 CreateSnapshot 또는 CreateSnapshots 요청에서 태그가 지정되는 경우에만 평가됩니다. 태그가 지정된 경우 태그는 purpose=test여야 합니다. 다른 어떤 태그도 요청에서 허용되지 않습니다.

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect":"Allow",
            "Action":"ec2:CreateSnapshot",
            "Resource":"*"
        },
        {
            "Effect":"Allow",
            "Action":"ec2:CreateTags",
            "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
            "Condition":{
                "StringEquals":{
                    "aws:RequestTag/purpose":"test",
                    "ec2:CreateAction":"CreateSnapshot"
                },
                "ForAllValues:StringEquals":{
                    "aws:TagKeys":"purpose"
                }
            }
        }
    ]
}
```

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect":"Allow",
            "Action":"ec2:CreateSnapshots",
            "Resource":"*"
        },
        {
            "Effect":"Allow",
            "Action":"ec2:CreateTags",
            "Resource":"arn:aws:ec2:us-east-1::snapshot/*",
            "Condition":{
                "StringEquals":{
                    "aws:RequestTag/purpose":"test",
                    "ec2:CreateAction":"CreateSnapshots"
                },
                "ForAllValues:StringEquals":{
                    "aws:TagKeys":"purpose"
                }
            }
        }
    ]
}
```

다음 정책은 소스 볼륨이 고객에 대해 User:username으로 태그 지정된 경우, 그리고 스냅샷 자체가 Environment:Dev 및 User:username으로 태그 지정된 경우에만 스냅샷 생성을 허용합니다. 고객은 스냅샷에 추가 태그를 추가할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateSnapshot",  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:volume/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:ResourceTag/User": "${aws:username}"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateSnapshot",  
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:RequestTag/Environment": "Dev",  
                    "aws:RequestTag/User": "${aws:username}"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateTags",  
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*"  
        }  
    ]  
}
```

CreateSnapshots에 대한 다음 정책은 고객에 대해 User:username으로 태그 지정된 경우, 그리고 스냅샷 자체가 Environment:Dev 및 User:username으로 태그 지정된 경우에만 스냅샷 생성을 허용합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateSchedules",  
            "Resource": "arn:aws:ec2:us-east-1::*:instance/*",  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateSchedules",  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:volume/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:ResourceTag/User": "${aws:username}"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateSchedules",  
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:RequestTag/Environment": "Dev",  
                    "aws:RequestTag/User": "${aws:username}"  
                }  
            }  
        }  
    ]  
}
```

```
{  
    "Effect": "Allow",  
    "Action": "ec2:CreateTags",  
    "Resource": "arn:aws:ec2:us-east-1::snapshot/*"  
}  
]  
}
```

다음 정책은 스냅샷이 고객에 대해 User:username으로 태그 지정된 경우에만 스냅샷 삭제를 허용합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:DeleteSnapshot",  
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:ResourceTag/User": "${aws:username}"  
                }  
            }  
        }  
    ]  
}
```

다음 정책은 고객의 스냅샷 생성을 허용하지만 생성되는 스냅샷이 태그 키 value=stack을 보유한 경우에
는 작업을 거부합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateSnapshot",  
                "ec2:CreateTags"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Deny",  
            "Action": "ec2:CreateSnapshot",  
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",  
            "Condition": {  
                "ForAnyValue:StringEquals": {  
                    "aws:TagKeys": "stack"  
                }  
            }  
        }  
    ]  
}
```

다음 정책은 고객의 스냅샷 생성을 허용하지만 생성되는 스냅샷이 태그 키 value=stack을 보유한 경우에
는 작업을 거부합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateSnapshot",  
                "ec2:CreateTags"  
            ]  
        }  
    ]  
}
```

```
        "ec2:CreateSnapshots",
        "ec2:CreateTags"
    ],
    "Resource": "*"
},
{
    "Effect": "Deny",
    "Action": "ec2:CreateSnapshots",
    "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
    "Condition": {
        "ForAnyValue:StringEquals": {
            "aws:TagKeys": "stack"
        }
    }
}
]
```

다음 정책은 여러 작업을 정책 하나로 결합하도록 허용합니다. 스냅샷이 us-east-1 리전에서 생성된 경우에만 (CreateSnapshots의 컨텍스트에서) 스냅샷을 생성할 수 있습니다. 스냅샷이 us-east-1 리전에서 생성되고 인스턴스 유형이 t2*인 경우에만 (CreateSnapshots의 컨텍스트에서) 스냅샷을 생성할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateSnapshots",
                "ec2:CreateSnapshot",
                "ec2:CreateTags"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*",
                "arn:aws:ec2:*:*:snapshot/*",
                "arn:aws:ec2:*:*:volume/*"
            ],
            "Condition": {
                "StringEqualsIgnoreCase": {
                    "ec2:Region": "us-east-1"
                },
                "StringLikeIfExists": {
                    "ec2:InstanceType": ["t2.*"]
                }
            }
        }
    ]
}
```

예제: 스냅샷 권한 설정 수정

다음 정책에서는 스냅샷에 **User:username** 태그가 지정된 경우에만 스냅샷 수정을 허용합니다. 여기서 **username**은 고객의 AWS 계정 사용자 이름입니다. 이 조건이 충족되지 않으면 요청은 실패합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2: ModifySnapshotAttribute",
            "Resource": "arn:aws:ec2:us-east-1::snapshot/*",
            "Condition": {

```

```
        "StringEquals":{  
            "ec2:ResourceTag/user-name":"${aws:username}"  
        }  
    }  
}  
]
```

인스턴스 시작(RunInstances)

[RunInstances API](#) 작업은 하나 이상의 온디맨드 인스턴스 또는 하나 이상의 스팟 인스턴스를 시작합니다. [RunInstances](#)는 AMI를 필요로 하며 인스턴스를 생성합니다. 사용자는 요청에 키 페어와 보안 그룹을 지정할 수 있습니다. VPC로 시작하는 경우 서브넷을 입력 받아 네트워크 인터페이스를 생성합니다. Amazon EBS 지원 AMI에서 시작하면 볼륨이 생성됩니다. 따라서 사용자가 이러한 Amazon EC2 리소스를 사용할 권한을 가지고 있어야 합니다. 사용자가 [RunInstances](#)에서 선택적 파라미터를 지정하도록 요구하거나 사용자가 파라미터에 특정 값을 사용하도록 제한하는 정책 명령문을 생성할 수 있습니다.

인스턴스를 시작하는 데 필요한 리소스 수준 권한에 대한 자세한 내용은 IAM 사용 설명서의 [Amazon EC2에 사용되는 작업, 리소스 및 조건 키](#)를 참조하십시오.

기본적으로는 사용자에게 생성된 인스턴스를 설명, 시작, 중지 또는 종료할 권한이 없습니다. 사용자에게 결과 인스턴스를 관리할 권한을 부여하는 방법 중 하나는 각 인스턴스에 대한 특정 태그를 생성하고 해당 태그를 갖는 인스턴스를 관리하도록 허용하는 명령문을 생성하는 것입니다. 자세한 내용은 [인스턴스 작업 \(p. 847\)](#) 단원을 참조하십시오.

리소스

- [AMI \(p. 858\)](#)
- [인스턴스 유형 \(p. 859\)](#)
- [서브넷 \(p. 860\)](#)
- [EBS 볼륨 \(p. 861\)](#)
- [Tags \(p. 862\)](#)
- [시작 템플릿의 태그 \(p. 866\)](#)
- [탄력적 GPU \(p. 866\)](#)
- [시작 템플릿 \(p. 867\)](#)

AMI

다음 정책은 사용자가 지정된 AMI(ami-9e1670f7 및 ami-45cf5c3c)만 사용하여 인스턴스를 시작하도록 허용합니다. 다른 명령문에서 해당 권한을 부여하지 않는 경우 다른 AMI를 사용하여 인스턴스를 시작할 수 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "Resource": [  
                "arn:aws:ec2:region::image/ami-9e1670f7",  
                "arn:aws:ec2:region::image/ami-45cf5c3c",  
                "arn:aws:ec2:region:account:instance/*",  
                "arn:aws:ec2:region:account:volume/*",  
                "arn:aws:ec2:region:account:key-pair/*",  
                "arn:aws:ec2:region:account:security-group/*",  
                "arn:aws:ec2:region:account:subnet/*",  
                "arn:aws:ec2:region:account:network-interface/*"  
            ]  
        }  
    ]  
}
```

```
        ]  
    }
```

또는 다음 정책으로 Amazon이 소유한 모든 AMI에서 인스턴스를 시작하도록 허용할 수 있습니다. 첫 번째 명령문의 Condition 요소는 ec2:Owner가 amazon인지 여부를 테스트합니다. 다른 명령문에서 해당 권한을 부여하지 않는 경우 다른 AMI를 사용하여 인스턴스를 시작할 수 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "Resource": [  
                "arn:aws:ec2:region::image/ami-*"  
            ],  
            "Condition": {  
                "StringEquals": {  
                    "ec2:Owner": "amazon"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "Resource": [  
                "arn:aws:ec2:region:account:instance/*",  
                "arn:aws:ec2:region:account:subnet/*",  
                "arn:aws:ec2:region:account:volume/*",  
                "arn:aws:ec2:region:account:network-interface/*",  
                "arn:aws:ec2:region:account:key-pair/*",  
                "arn:aws:ec2:region:account:security-group/*"  
            ]  
        }  
    ]  
}
```

인스턴스 유형

다음 정책은 사용자가 t2.micro 및 t2.small 인스턴스 유형만 사용하여 인스턴스를 시작하도록 허용하므로 비용 통제에 도움이 됩니다. 첫 번째 명령문의 Condition 요소에서 ec2:InstanceType이 t2.micro 또는 t2.small인지 여부를 테스트하므로 더욱 큰 인스턴스는 시작할 수 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "Resource": [  
                "arn:aws:ec2:region:account:instance/*"  
            ],  
            "Condition": {  
                "StringEquals": {  
                    "ec2:InstanceType": ["t2.micro", "t2.small"]  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "Resource": [  
                "arn:aws:ec2:region::image/ami-*",  
            ]  
        }  
    ]  
}
```

```
        "arn:aws:ec2:region:account:subnet/*",
        "arn:aws:ec2:region:account:network-interface/*",
        "arn:aws:ec2:region:account:volume/*",
        "arn:aws:ec2:region:account:key-pair/*",
        "arn:aws:ec2:region:account:security-group/*"
    ]
}
]
```

또는 t2.micro 및 t2.small 인스턴스 유형을 제외한 모든 인스턴스를 시작할 수 있는 사용자 권한을 거부하는 정책을 생성할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:region:account:instance/*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ec2:InstanceType": ["t2.micro", "t2.small"]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:region::image/ami-*",
                "arn:aws:ec2:region:account:network-interface/*",
                "arn:aws:ec2:region:account:instance/*",
                "arn:aws:ec2:region:account:subnet/*",
                "arn:aws:ec2:region:account:volume/*",
                "arn:aws:ec2:region:account:key-pair/*",
                "arn:aws:ec2:region:account:security-group/*"
            ]
        }
    ]
}
```

서브넷

다음 정책은 사용자가 지정된 서브넷(subnet-12345678)만 사용하여 인스턴스를 시작하도록 허용합니다. 다른 명령문에서 해당 권한을 부여하지 않는 경우 그룹에서 다른 서브넷으로 인스턴스를 시작할 수 없습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:region:account:subnet/subnet-12345678",
                "arn:aws:ec2:region:account:network-interface/*",
                "arn:aws:ec2:region:account:instance/*",
                "arn:aws:ec2:region:account:volume/*",
                "arn:aws:ec2:region::image/ami-*",
                "arn:aws:ec2:region:account:key-pair/*",
            ]
        }
    ]
}
```

```
        "arn:aws:ec2:region:account:security-group/*"
    ]
}
}
```

또는 다른 서브넷으로 인스턴스를 시작할 수 있는 사용자 권한을 거부하는 정책을 생성할 수 있습니다. 명령문에서 subnet-12345678 서브넷이 지정된 경우를 제외하고 네트워크 인터페이스를 생성할 권한을 거부하면 됩니다. 이러한 거부는 다른 서브넷으로 인스턴스를 시작하도록 허용할 목적으로 생성된 다른 정책을 모두 무시합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:region:account:network-interface/*"
            ],
            "Condition": {
                "ArnNotEquals": {
                    "ec2:Subnet": "arn:aws:ec2:region:account:subnet/subnet-12345678"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:region::image/ami-*",
                "arn:aws:ec2:region:account:network-interface/*",
                "arn:aws:ec2:region:account:instance/*",
                "arn:aws:ec2:region:account:subnet/*",
                "arn:aws:ec2:region:account:volume/*",
                "arn:aws:ec2:region:account:key-pair/*",
                "arn:aws:ec2:region:account:security-group/*"
            ]
        }
    ]
}
```

EBS 볼륨

다음 정책은 인스턴스의 EBS 볼륨이 암호화된 경우에만 사용자가 인스턴스를 시작하는 것을 허용합니다. 사용자는 암호화된 스냅샷을 사용하여 생성된 AMI에서 인스턴스를 시작하여 루트 볼륨이 암호화되도록 해야 합니다. 시작 도중 사용자가 인스턴스에 연결하는 추가적 볼륨도 암호화되어야 합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:region:volume/*"
            ],
            "Condition": {
                "Bool": {
                    "ec2:Encrypted": "true"
                }
            }
        },
    ]
}
```

```
{  
    "Effect": "Allow",  
    "Action": "ec2:RunInstances",  
    "Resource": [  
        "arn:aws:ec2::::image/ami-*",  
        "arn:aws:ec2::::network-interface/*",  
        "arn:aws:ec2::::instance/*",  
        "arn:aws:ec2::::subnet/*",  
        "arn:aws:ec2::::key-pair/*",  
        "arn:aws:ec2::::security-group/*"  
    ]  
}  
}  
]
```

Tags

생성 시 인스턴스에 태그 지정

다음 정책은 사용자가 인스턴스를 시작하고 생성 중에 인스턴스에 태그를 지정하는 것을 허용합니다. 태그를 적용하는 리소스 생성 작업의 경우, 사용자가 CreateTags 작업을 사용할 권한을 가지고 있어야 합니다. 두 번째 문은 ec2:CreateAction 조건 키를 사용하여 사용자가 RunInstances의 컨텍스트에 대해 인스턴스의 태그만을 생성하는 것을 허용합니다. 사용자는 기존의 리소스에 태그를 지정할 수 없으며, RunInstances 요청을 사용하여 볼륨에 태그를 지정할 수 없습니다.

자세한 내용은 [생성 시 리소스 태그 지정에 대한 권한 부여 \(p. 843\)](#) 단원을 참조하십시오.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateTags"  
            ],  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:instance/*",  
            "Condition": {  
                "StringEquals": {  
                    "ec2:CreateAction" : "RunInstances"  
                }  
            }  
        }  
    ]  
}
```

생성 시 인스턴스 및 볼륨에 특정 태그를 사용하여 태그 지정

다음 정책에는 태그 aws:RequestTag 및 RunInstances를 사용하여 environment=production에 의해 생성되는 인스턴스와 볼륨에 사용자가 태그를 지정해야 하는 purpose=webserver 조건 키가 포함됩니다. aws:TagKeys 조건 키는 ForAllValues 변경자를 사용하여 요청에서 키 environment 및 purpose만 허용됨을 표시합니다(다른 어떤 태그도 지정할 수 없습니다). 요청에서 태그가 지정되지 않으면 요청이 실패합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances"  
            ],  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:instance/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:RequestTag/environment" : "environment=production",  
                    "aws:RequestTag/purpose" : "purpose=webserver",  
                    "aws:TagKeys" : "environment,purpose"  
                }  
            }  
        }  
    ]  
}
```

```
{
    "Effect": "Allow",
    "Action": [
        "ec2:RunInstances"
    ],
    "Resource": [
        "arn:aws:ec2:region::image/*",
        "arn:aws:ec2:region:account:subnet/*",
        "arn:aws:ec2:region:account:network-interface/*",
        "arn:aws:ec2:region:account:security-group/*",
        "arn:aws:ec2:region:account:key-pair/*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:RunInstances"
    ],
    "Resource": [
        "arn:aws:ec2:region:account:volume/*",
        "arn:aws:ec2:region:account:instance/*"
    ],
    "Condition": {
        "StringEquals": {
            "aws:RequestTag/environment": "production" ,
            "aws:RequestTag/purpose": "webserver"
        },
        "ForAllValues:StringEquals": {
            "aws:TagKeys": ["environment", "purpose"]
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:region:account:/*",
    "Condition": {
        "StringEquals": {
            "ec2:CreateAction" : "RunInstances"
        }
    }
}
]
```

생성 시 인스턴스 및 볼륨에 하나 이상의 특정 태그를 사용하여 태그 지정

다음 정책은 `ForAnyValue` 조건에서 `aws:TagKeys` 변경자를 사용하여 요청에서 적어도 하나의 태그가 지정되어야 하고 태그에 키 `environment` 또는 `webserver`가 포함되어야 함을 표시합니다. 태그는 인스턴스와 볼륨에 모두 적용되어야 합니다. 요청에서 어떤 태그 값도 지정할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:region::image/*",
                "arn:aws:ec2:region:account:subnet/*",
                "arn:aws:ec2:region:account:volume/*",
                "arn:aws:ec2:region:account:instance/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:TagKeys": ["environment", "purpose"]
                }
            }
        }
    ]
}
```

```
"arn:aws:ec2:region:account:network-interface/*",
"arn:aws:ec2:region:account:security-group/*",
"arn:aws:ec2:region:account:key-pair/*"
],
},
{
"Effect": "Allow",
"Action": [
    "ec2:RunInstances"
],
"Resource": [
    "arn:aws:ec2:region:account:volume/*",
    "arn:aws:ec2:region:account:instance/*"
],
"Condition": {
    "ForAnyValue:StringEquals": {
        "aws:TagKeys": ["environment", "webserver"]
    }
},
{
"Effect": "Allow",
"Action": [
    "ec2:CreateTags"
],
"Resource": "arn:aws:ec2:region:account:*/",
"Condition": {
    "StringEquals": {
        "ec2:CreateAction" : "RunInstances"
    }
}
}
]
```

인스턴스를 생성할 때 태그를 지정하는 경우 특정 태그를 사용해야 함

다음 정책에서는 요청에서 태그를 지정할 필요가 없지만 지정하는 경우, 태그는 purpose=test여야 합니다. 다른 어떤 태그도 허용되지 않습니다. 사용자는 RunInstances 요청에서 태그 지정 가능한 어떤 리소스에도 태그를 적용할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:region:account:*/",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/purpose": "test",
                    "ec2:CreateAction" : "RunInstances"
                },
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": "purpose"
                }
            }
        }
    ]
}
```

```
        }
    }
}
```

RunInstances의 경우 생성 시 호출자가 태그를 지정할 수 없도록 하려면

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1::subnet/*",
                "arn:aws:ec2:us-east-1::network-interface/*",
                "arn:aws:ec2:us-east-1::security-group/*",
                "arn:aws:ec2:us-east-1::key-pair/*",
                "arn:aws:ec2:us-east-1::volume/*",
                "arn:aws:ec2:us-east-1::instance/*",
                "arn:aws:ec2:us-east-1::spot-instances-request/*"
            ]
        },
        {
            "Sid": "VisualEditor0",
            "Effect": "Deny",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

spot-instances-request에 대해서만 특정 태그를 허용합니다. 여기서도 예외적인 비일관성이 적용됩니다. 일반적인 상황에서는 태그를 지정하지 않으면 인증되지 않습니다. spot-instances-request의 경우 spot-instances-request 태그가 없으면 이 정책이 평가되지 않으므로 태그가 지정되지 않은 Spot on Run 요청이 성공합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1::subnet/*",
                "arn:aws:ec2:us-east-1::network-interface/*",
                "arn:aws:ec2:us-east-1::security-group/*",
                "arn:aws:ec2:us-east-1::key-pair/*",
                "arn:aws:ec2:us-east-1::volume/*",
                "arn:aws:ec2:us-east-1::instance/*",
                "arn:aws:ec2:us-east-1::spot-instances-request/*"
            ]
        },
        {
            "Sid": "VisualEditor0",

```

```
        "Effect": "Allow",
        "Action": "ec2:RunInstances",
        "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",
        "Condition": {
            "StringEquals": {
                "aws:RequestTag/environment": "production"
            }
        }
    ]
}
```

시작 템플릿의 태그

아래 예제에서 사용자는 특정 시작 템플릿(1t-09477bcd97b0d310e)을 사용하는 경우에만 인스턴스를 시작할 수 있습니다. `ec2:IsLaunchTemplateResource` 조건 키는 사용자가 시작 템플릿에 지정된 모든 리소스를 재정의할 수 없도록 합니다. 이 명령문의 두 번째 부분은 사용자가 생성 시 인스턴스에 태그를 지정하도록 허용합니다. 이 부분은 시작 템플릿의 인스턴스에 대해 태그가 지정되어 있는 경우에 반드시 필요합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": "*",
            "Condition": {
                "ArnLike": {
                    "ec2:LaunchTemplate": "arn:aws:ec2:region:account:launch-template/
1t-09477bcd97b0d310e"
                },
                "Bool": {
                    "ec2:IsLaunchTemplateResource": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2>CreateTags"
            ],
            "Resource": "arn:aws:ec2:region:account:instance/*",
            "Condition": {
                "StringEquals": {
                    "ec2>CreateAction" : "RunInstances"
                }
            }
        }
    ]
}
```

탄력적 GPU

다음 정책에서 사용자는 인스턴스를 시작하고 탄력적 GPU를 지정하여 인스턴스에 연결합니다. 사용자는 모든 지역에서 인스턴스를 시작할 수 있지만 `us-east-2` 지역에서 시작 작업 동안 탄력적 GPU만 연결할 수 있습니다.

`ec2:ElasticGpuType` 조건 키는 `ForAnyValue` 한정자를 사용하여 요청에서 엘라스틱 GPU 유형 `eg1.medium` 및 `eg1.large`만 허용됨을 표시합니다.

```
{
    "Version": "2012-10-17",
```

```

"Statement": [
    {
        "Effect": "Allow",
        "Action": [
            "ec2:RunInstances"
        ],
        "Resource": [
            "arn:aws:ec2:*:account:elastic-gpu/*"
        ],
        "Condition": {
            "StringEquals": {
                "ec2:Region": "us-east-2"
            },
            "ForAnyValue:StringLike": {
                "ec2:ElasticGpuType": [
                    "eg1.medium",
                    "eg1.large"
                ]
            }
        }
    },
    {
        "Effect": "Allow",
        "Action": "ec2:RunInstances",
        "Resource": [
            "arn:aws:ec2:*::image/ami-*",
            "arn:aws:ec2:*:account:network-interface/*",
            "arn:aws:ec2:*:account:instance/*",
            "arn:aws:ec2:*:account:subnet/*",
            "arn:aws:ec2:*:account:volume/*",
            "arn:aws:ec2:*:account:key-pair/*",
            "arn:aws:ec2:*:account:security-group/*"
        ]
    }
]
}

```

시작 템플릿

아래 예제에서 사용자는 특정 시작 템플릿(1t-09477bcd97b0d310e)을 사용하는 경우에만 인스턴스를 시작할 수 있습니다. 사용자는 RunInstances 작업에서 파라미터를 지정하여 시작 템플릿의 모든 파라미터를 재정의할 수 있습니다.

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": "*",
            "Condition": {
                "ArnLike": {
                    "ec2:LaunchTemplate": "arn:aws:ec2:region:account:launch-template/
1t-09477bcd97b0d310e"
                }
            }
        }
    ]
}

```

아래 예제에서 사용자는 시작 템플릿을 사용하는 경우에만 인스턴스를 시작할 수 있습니다. 정책은 ec2:IsLaunchTemplateResource 조건 키를 사용하여 사용자가 시작 템플릿의 기존 ARN을 재정의하지 못하도록 합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "Resource": "*",  
            "Condition": {  
                "ArnLike": {  
                    "ec2:LaunchTemplate": "arn:aws:ec2:region:account:launch-template/*"  
                },  
                "Bool": {  
                    "ec2:IsLaunchTemplateResource": "true"  
                }  
            }  
        }  
    ]  
}
```

아래 예제는 사용자가 시작 템플릿을 사용하는 경우에만 인스턴스를 시작하도록 허용하는 정책입니다. 사용자는 요청 시 서브넷 및 네트워크 인터페이스 파라미터를 재정의할 수 없으며, 시작 템플릿에서만 이러한 파라미터들을 지정할 수 있습니다. 이 명령문의 첫 번째 부분은 [NotResource](#) 요소를 사용하여 서브넷 및 네트워크 인터페이스를 제외한 다른 모든 리소스를 허용합니다. 이 명령문의 두 번째 부분은 시작 템플릿에서 나온 경우에만 서브넷 및 네트워크 인터페이스 리소스를 허용합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "NotResource": [ "arn:aws:ec2:region:account:subnet/*",  
                            "arn:aws:ec2:region:account:network-interface/*" ],  
            "Condition": {  
                "ArnLike": {  
                    "ec2:LaunchTemplate": "arn:aws:ec2:region:account:launch-template/*"  
                }  
            }  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "Resource": [ "arn:aws:ec2:region:account:subnet/*",  
                         "arn:aws:ec2:region:account:network-interface/*" ],  
            "Condition": {  
                "ArnLike": {  
                    "ec2:LaunchTemplate": "arn:aws:ec2:region:account:launch-template/*"  
                },  
                "Bool": {  
                    "ec2:IsLaunchTemplateResource": "true"  
                }  
            }  
        }  
    ]  
}
```

아래 예제는 시작 템플릿을 사용하고 있고 시작 템플릿에 Purpose=Webservers 태그가 있는 경우에만 인스턴스를 시작할 수 있도록 허용합니다. 사용자는 RunInstances 작업의 어떤 시작 템플릿 파라미터도 재정의할 수 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "Resource": "arn:aws:ec2:region:account:instance/*",  
            "Condition": {  
                "StringLike": {  
                    "ec2:Tags[*].Value": "Purpose=Webservers"  
                }  
            }  
        }  
    ]  
}
```

```
"Statement": [
    {
        "Effect": "Allow",
        "Action": "ec2:RunInstances",
        "NotResource": "arn:aws:ec2:region:account:launch-template/*",
        "Condition": {
            "ArnLike": {
                "ec2:LaunchTemplate": "arn:aws:ec2:region:account:launch-template/*"
            },
            "Bool": {
                "ec2:IsLaunchTemplateResource": "true"
            }
        }
    },
    {
        "Effect": "Allow",
        "Action": "ec2:RunInstances",
        "Resource": "arn:aws:ec2:region:account:launch-template/*",
        "Condition": {
            "StringEquals": {
                "ec2:ResourceTag/Purpose": "Webservers"
            }
        }
    }
]
```

스팟 인스턴스 작업

RunInstances 작업을 사용하여 스팟 인스턴스 요청을 만들고 생성 시 스팟 인스턴스 요청에 태그를 지정할 수 있습니다. RunInstances에 대해 지정할 리소스는 spot-instances-request입니다.

spot-instances-request 리소스는 다음과 같이 IAM 정책에서 평가됩니다.

- 생성 시 스팟 인스턴스 요청에 태그를 지정하지 않으면 Amazon EC2는 RunInstances 문에서 spot-instances-request 리소스를 평가하지 않습니다.
- 생성 시 스팟 인스턴스 요청에 태그를 지정하면 Amazon EC2는 RunInstances 문에서 spot-instances-request 리소스를 평가합니다.

따라서 spot-instances-request 리소스의 경우 IAM 정책에 다음 규칙이 적용됩니다.

- RunInstances를 사용하여 스팟 인스턴스 요청을 만들고 생성 시 스팟 인스턴스 요청에 태그를 지정하지 않으려는 경우 spot-instances-request 리소스를 명시적으로 허용할 필요가 없습니다. 호출이 성공 합니다.
- RunInstances를 사용하여 스팟 인스턴스 요청을 만들고 생성 시 스팟 인스턴스 요청에 태그를 지정하려는 경우 RunInstances allow 문에 spot-instances-request 리소스를 포함해야 합니다. 그렇지 않으면 호출이 실패합니다.
- RunInstances를 사용하여 스팟 인스턴스 요청을 만들고 생성 시 스팟 인스턴스 요청에 태그를 지정하려는 경우 CreateTags allow 문에서 spot-instances-request 리소스 또는 * 와일드카드를 지정해야 합니다. 그렇지 않으면 호출이 실패합니다.

RunInstances 또는 RequestSpotInstances를 사용하여 스팟 인스턴스를 요청할 수 있습니다. 다음 예제 IAM 정책은 RunInstances를 사용하여 스팟 인스턴스를 요청할 때만 적용됩니다.

예: RunInstances를 사용한 스팟 인스턴스 요청

다음 정책은 사용자가 RunInstances 작업을 사용하여 스팟 인스턴스를 요청할 수 있도록 허용합니다. RunInstances에서 생성되는 spot-instances-request 리소스는 스팟 인스턴스를 요청합니다.

Note

RunInstances를 사용하여 스팟 인스턴스 요청을 만들 때 생성 시 스팟 인스턴스 요청에 태그를 지정하지 않으려는 경우 Resource 목록에서 spot-instances-request를 생략할 수 있습니다. 생성 시 스팟 인스턴스 요청에 태그를 지정하지 않으면 Amazon EC2는 RunInstances 문에서 spot-instances-request 리소스를 평가하지 않기 때문입니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "AllowRun",  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances"  
            ],  
            "Resource": [  
                "arn:aws:ec2:us-east-1::image/*",  
                "arn:aws:ec2:us-east-1:*:subnet/*",  
                "arn:aws:ec2:us-east-1:*:network-interface/*",  
                "arn:aws:ec2:us-east-1:*:security-group/*",  
                "arn:aws:ec2:us-east-1:*:key-pair/*",  
                "arn:aws:ec2:us-east-1:*:volume/*",  
                "arn:aws:ec2:us-east-1:*:instance/*",  
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"  
            ]  
        }  
    ]  
}
```

Warning

지원되지 않음 – 예: 사용자에게 RunInstances를 사용하여 스팟 인스턴스를 요청할 수 있는 권한 거부

spot-instances-request 리소스에는 다음 정책이 지원되지 않습니다.

다음 정책은 사용자에게 온디맨드 인스턴스를 시작할 수 있는 권한을 부여하지만 사용자에게 스팟 인스턴스를 요청할 수 있는 권한은 거부합니다. RunInstances에서 생성되는 spot-instances-request 리소스는 스팟 인스턴스를 요청하는 리소스입니다. 두 번째 문은 spot-instances-request 리소스에 대한 RunInstances 작업을 거부합니다. 그러나 생성 시 스팟 인스턴스 요청에 태그를 지정하지 않으면 Amazon EC2는 RunInstances 문에서 spot-instances-request 리소스를 평가하지 않으므로 이 조건은 지원되지 않습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "AllowRun",  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances"  
            ],  
            "Resource": [  
                "arn:aws:ec2:us-east-1::image/*",  
                "arn:aws:ec2:us-east-1:*:subnet/*",  
                "arn:aws:ec2:us-east-1:*:network-interface/*",  
                "arn:aws:ec2:us-east-1:*:security-group/*",  
                "arn:aws:ec2:us-east-1:*:key-pair/*",  
                "arn:aws:ec2:us-east-1:*:volume/*",  
                "arn:aws:ec2:us-east-1:*:instance/*"  
            ]  
        },  
        {  
            "Sid": "DenyRunInstances",  
            "Effect": "Deny",  
            "Action": ["ec2:RunInstances"],  
            "Resource": [  
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"  
            ]  
        }  
    ]  
}
```

```
        "Sid": "DenySpotInstancesRequests - NOT SUPPORTED - DO NOT USE!",
        "Effect": "Deny",
        "Action": "ec2:RunInstances",
        "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
    }
]
```

예: 생성 시 스팟 인스턴스 요청에 태그 지정

다음 정책은 사용자가 인스턴스 시작 중에 생성된 모든 리소스에 태그를 지정할 수 있도록 허용합니다. 첫 번째 문은 RunInstances에서 나열된 리소스를 생성할 수 있도록 허용합니다. RunInstances에서 생성되는 spot-instances-request 리소스는 스팟 인스턴스를 요청하는 리소스입니다. 두 번째 문은 * 와일드카드를 지정하여 인스턴스 시작 시 리소스가 생성될 때 모든 리소스에 태그를 지정하는 것을 허용합니다.

Note

생성 시 스팟 인스턴스 요청에 태그를 지정하면 Amazon EC2는 RunInstances 문에서 spot-instances-request 리소스를 평가합니다. 따라서 RunInstances 작업의 경우 spot-instances-request 리소스를 명시적으로 허용해야 합니다. 그렇지 않으면 호출이 실패합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1::*:subnet/*",
                "arn:aws:ec2:us-east-1::*:network-interface/*",
                "arn:aws:ec2:us-east-1::*:security-group/*",
                "arn:aws:ec2:us-east-1::*:key-pair/*",
                "arn:aws:ec2:us-east-1::*:volume/*",
                "arn:aws:ec2:us-east-1::*:instance/*",
                "arn:aws:ec2:us-east-1::*:spot-instances-request/*"
            ]
        },
        {
            "Sid": "TagResources",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

예: 생성 시 스팟 인스턴스 요청에 태그 지정 거부

다음 정책은 사용자에게 인스턴스 시작 중에 생성된 리소스에 태그를 지정할 수 있는 권한을 거부합니다.

첫 번째 문은 RunInstances에서 나열된 리소스를 생성할 수 있도록 허용합니다. RunInstances에서 생성되는 spot-instances-request 리소스는 스팟 인스턴스를 요청하는 리소스입니다. 두 번째 문은 * 와일드카드를 제공하여 인스턴스 시작 시 리소스가 생성될 때 모든 리소스에 태그를 지정하는 것을 거부합니다. 생성 시 spot-instances-request 또는 다른 리소스에 태그를 지정하면 RunInstances 호출이 실패합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
{  
    "Sid": "AllowRun",  
    "Effect": "Allow",  
    "Action": [  
        "ec2:RunInstances"  
    ],  
    "Resource": [  
        "arn:aws:ec2:us-east-1::image/*",  
        "arn:aws:ec2:us-east-1::subnet/*",  
        "arn:aws:ec2:us-east-1::network-interface/*",  
        "arn:aws:ec2:us-east-1::security-group/*",  
        "arn:aws:ec2:us-east-1::key-pair/*",  
        "arn:aws:ec2:us-east-1::volume/*",  
        "arn:aws:ec2:us-east-1::instance/*",  
        "arn:aws:ec2:us-east-1::spot-instances-request/*"  
    ]  
},  
{  
    "Sid": "DenyTagResources",  
    "Effect": "Deny",  
    "Action": "ec2:CreateTags",  
    "Resource": "*"  
}  
]  
}
```

Warning

지원되지 않음 – 예: 특정 태그가 지정된 경우에만 스팟 인스턴스 요청을 생성하는 것을 허용 spot-instances-request 리소스에는 다음 정책이 지원되지 않습니다.

다음 정책은 요청에 특정 태그가 지정된 경우에만 RunInstances에 스팟 인스턴스 요청을 생성할 수 있는 권한을 부여합니다.

첫 번째 문은 RunInstances에서 나열된 리소스를 생성할 수 있도록 허용합니다.

두 번째 문은 요청에 environment=production 태그가 있는 경우에만 사용자에게 스팟 인스턴스 요청을 생성할 수 있는 권한을 부여합니다. 이 조건이 RunInstances에서 생성된 다른 리소스에 적용되는 경우 태그를 지정하지 않으면 Unauthenticated 오류가 발생합니다. 그러나 스팟 인스턴스 요청에 태그가 지정되지 않은 경우 Amazon EC2는 RunInstances 문에서 spot-instances-request 리소스를 평가하지 않으므로 RunInstances에서 태그가 지정되지 않은 스팟 인스턴스 요청이 생성됩니다.

사용자가 스팟 인스턴스 요청에 태그를 지정하는 경우 Amazon EC2는 RunInstances 문에서 spot-instances-request 리소스를 평가하기 때문에 environment=production 이외의 다른 태그를 지정하면 Unauthenticated 오류가 발생합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "AllowRun",  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances"  
            ],  
            "Resource": [  
                "arn:aws:ec2:us-east-1::image/*",  
                "arn:aws:ec2:us-east-1::subnet/*",  
                "arn:aws:ec2:us-east-1::network-interface/*",  
                "arn:aws:ec2:us-east-1::security-group/*",  
                "arn:aws:ec2:us-east-1::key-pair/*",  
                "arn:aws:ec2:us-east-1::volume/*",  
                "arn:aws:ec2:us-east-1::instance/*"  
            ]  
        },  
    ]  
},
```

```
{  
    "Sid": "RequestSpotInstancesOnlyIfTagIs_environment=production - NOT  
SUPPORTED - DO NOT USE!",  
    "Effect": "Allow",  
    "Action": "ec2:RunInstances",  
    "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",  
    "Condition": {  
        "StringEquals": {  
            "aws:RequestTag/environment": "production"  
        }  
    }  
},  
{  
    "Sid": "TagResources",  
    "Effect": "Allow",  
    "Action": "ec2:CreateTags",  
    "Resource": "*"  
}  
]  
}
```

예: 특정 태그가 지정된 경우 스팟 인스턴스 요청을 생성하는 것을 거부

다음 정책은 요청에 environment=production 태그가 지정된 경우 RunInstances에 스팟 인스턴스 요청을 생성할 수 있는 권한을 거부합니다.

첫 번째 문은 RunInstances에서 나열된 리소스를 생성할 수 있도록 허용합니다.

두 번째 문은 요청에 environment=production 태그가 있는 경우 사용자에게 스팟 인스턴스 요청을 생성할 수 있는 권한을 거부합니다. environment=production을 태그로 지정하면 Unauthenticated 오류가 발생합니다. 다른 태그를 지정하거나 태그를 지정하지 않으면 스팟 인스턴스 요청이 생성됩니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "AllowRun",  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances"  
            ],  
            "Resource": [  
                "arn:aws:ec2:us-east-1::image/*",  
                "arn:aws:ec2:us-east-1::*:subnet/*",  
                "arn:aws:ec2:us-east-1::*:network-interface/*",  
                "arn:aws:ec2:us-east-1::*:security-group/*",  
                "arn:aws:ec2:us-east-1::*:key-pair/*",  
                "arn:aws:ec2:us-east-1::*:volume/*",  
                "arn:aws:ec2:us-east-1::*:instance/*",  
                "arn:aws:ec2:us-east-1::*:spot-instances-request/*"  
            ]  
        },  
        {  
            "Sid": "DenySpotInstancesRequests",  
            "Effect": "Deny",  
            "Action": "ec2:RunInstances",  
            "Resource": "arn:aws:ec2:us-east-1::*:spot-instances-request/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:RequestTag/environment": "production"  
                }  
            }  
        },  
    ]  
}
```

```
{  
    "Sid": "TagResources",  
    "Effect": "Allow",  
    "Action": "ec2:CreateTags",  
    "Resource": "*"  
}  
]  
}
```

예제: 예약 인스턴스 작업

다음 정책에서는 계정에서 예약 인스턴스를 확인, 수정 및 구입할 수 있는 권한을 사용자에게 부여합니다.

개별 예약 인스턴스에 대해서는 리소스 수준 권한을 설정할 수 없습니다. 이 정책은 사용자들이 계정의 모든 예약 인스턴스에 액세스할 수 있음을 뜻합니다.

Resource 요소에서는 와일드카드(*)를 사용하여 사용자가 해당 작업에서 모든 리소스를 지정할 수 있음을 나타냅니다. 이 경우 사용자는 계정의 모든 예약 인스턴스를 나열하고 수정할 수 있습니다. 계정 자격 증명을 사용해 예약 인스턴스를 구매할 수도 있습니다. API 작업이 리소스 수준 권한을 지원하지 않는 경우에도 * 와일드카드가 필요합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:DescribeReservedInstances",  
                "ec2:ModifyReservedInstances",  
                "ec2:PurchaseReservedInstancesOffering",  
                "ec2:DescribeAvailabilityZones",  
                "ec2:DescribeReservedInstancesOfferings"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

사용자에게 계정의 예약 인스턴스를 확인 및 수정하도록 허용하되 새 예약 인스턴스를 구매할 수는 없게 하려면

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:DescribeReservedInstances",  
                "ec2:ModifyReservedInstances",  
                "ec2:DescribeAvailabilityZones"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

예제: 리소스에 태그 지정

다음 정책은 태그에 키 CreateTags 및 값 environment이 포함된 경우에만 사용자가 production 작업을 사용하여 인스턴스에 태그를 적용하는 것을 허용합니다. ForAllValues 변경자는 aws:TagKeys 조건

키와 함께 사용되어 요청에서 키 environment만 허용됨을 표시합니다(다른 어떤 태그도 허용되지 않습니다). 사용자는 다른 어떤 리소스 유형에도 태그를 지정할 수 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateTags"  
            ],  
            "Resource": "arn:aws:ec2:region:account:instance/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:RequestTag/environment": "production"  
                },  
                "ForAllValues:StringEquals": {  
                    "aws:TagKeys": [  
                        "environment"  
                    ]  
                }  
            }  
        }  
    ]  
}
```

다음 정책은 키가 owner이고 값이 IAM username인 태그를 이미 가진 태그 지정 가능한 리소스에 태그를 지정하는 것을 허용합니다. 또한 사용자는 요청에서 키가 anycompany:environment-type이고 값이 test 또는 prod인 태그를 지정해야 합니다. 사용자는 요청에서 추가 태그를 지정할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CreateTags"  
            ],  
            "Resource": "arn:aws:ec2:region:account:/*/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:RequestTag/anycompany:environment-type": ["test", "prod"],  
                    "ec2:ResourceTag/owner": "${aws:username}"  
                }  
            }  
        }  
    ]  
}
```

사용자가 리소스의 특정 태그를 삭제하는 것을 허용하는 IAM 정책을 만들 수 있습니다. 예를 들어 요청에서 지정된 태그 키가 environment 또는 cost-center인 경우, 다음 정책은 사용자가 볼륨의 태그를 삭제하는 것을 허용합니다. 태그에는 어떤 값도 지정할 수 있지만 태그 키는 지정된 키 중 하나와 일치해야 합니다.

Note

리소스를 삭제하면 리소스에 지정되어 있는 모든 태그도 함께 삭제됩니다. 사용자는 ec2:DeleteTags 작업을 사용할 권한이 없어도 태그가 지정된 리소스를 삭제할 수 있습니다. 삭제 작업을 수행할 권한만 있으면 됩니다.

```
{
```

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:DeleteTags",  
            "Resource": "arn:aws:ec2:us-east-1:123456789012:volume/*",  
            "Condition": {  
                "ForAllValues:StringEquals": {  
                    "aws:TagKeys": ["environment", "cost-center"]  
                }  
            }  
        }  
    ]  
}
```

이 정책은 키가 owner이고 값이 IAM username인 키로 리소스에 태그가 지정된 경우에 한해 어떤 리소스에 서든 environment=prod 태그만을 삭제하는 것을 허용합니다. 사용자는 리소스의 다른 어떤 태그도 삭제할 수 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:DeleteTags"  
            ],  
            "Resource": "arn:aws:ec2:region:account:/*/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:RequestTag/environment": "prod",  
                    "ec2:ResourceTag/owner": "${aws:username}"  
                },  
                "ForAllValues:StringEquals": {  
                    "aws:TagKeys": ["environment"]  
                }  
            }  
        }  
    ]  
}
```

예제: IAM 역할 작업

다음 정책을 통해 사용자는 department=test 태그가 있는 인스턴스에 IAM 역할을 연결, 교체 및 분리할 수 있습니다. IAM 역할을 교체하거나 분리하려면 연결 ID가 필요하기 때문에 정책은 사용자에게 ec2:DescribeIamInstanceProfileAssociations 작업을 사용할 수 있는 권한도 부여합니다.

IAM 사용자가 인스턴스에 역할을 전달하기 위해서는 iam:PassRole 작업을 사용할 수 있는 권한이 있어야 합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:AssociateIamInstanceProfile",  
                "ec2:ReplaceIamInstanceProfileAssociation",  
                "ec2:DisassociateIamInstanceProfile"  
            ],  
            "Resource": "arn:aws:ec2:region:account:instance/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:TagKeys": ["department"]  
                }  
            }  
        }  
    ]  
}
```

```
        "StringEquals": {
            "ec2:ResourceTag/department": "test"
        }
    },
{
    "Effect": "Allow",
    "Action": "ec2:DescribeIamInstanceProfileAssociations",
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "*"
}
]
```

다음 정책을 통해 사용자는 모든 인스턴스에 IAM 역할을 연결하거나 교체할 수 있습니다. 사용자는 이름이 TestRole-로 시작하는 IAM 역할만 연결하거나 교체할 수 있습니다. iam:PassRole 작업의 경우, 인스턴스 프로파일이 아닌 IAM 역할의 이름을 지정하십시오(이름이 다른 경우). 자세한 정보는 [인스턴스 프로파일 \(p. 891\)](#) 단원을 참조하십시오.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AssociateIamInstanceProfile",
                "ec2:ReplaceIamInstanceProfileAssociation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeIamInstanceProfileAssociations",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::account:role/TestRole-*"
        }
    ]
}
```

예제: 라우팅 테이블 작업

다음 정책은 VPC vpc-ec43eb89에만 연결된 라우팅 테이블의 경로에 대해 사용자가 추가, 제거 및 바꾸기 작업을 수행할 수 있도록 허용합니다. ec2:Vpc 조건 키에 대한 VPC를 지정하려면 VPC의 전체 ARN을 지정해야 합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DeleteRoute",
                "ec2:CreateRoute",
                "ec2:ReplaceRoute"
            ]
        }
    ]
}
```

```
        ],
        "Resource": [
            "arn:aws:ec2:region:account:route-table/*"
        ],
        "Condition": {
            "StringEquals": {
                "ec2:Vpc": "arn:aws:ec2:region:account:vpc/vpc-ec43eb89"
            }
        }
    ]
}
```

예제: 특정 인스턴스가 기타 AWS 서비스에서 리소스를 확인하도록 허용

다음은 IAM 역할에 연결할 수 있는 정책의 예제입니다. 이 정책은 인스턴스가 다양한 AWS 서비스에서 리소스를 확인하도록 허용합니다. `ec2:SourceInstanceARN` 조건 키를 사용하여 요청이 이루어진 인스턴스가 `i-093452212644b0dd6` 인스턴스가 되도록 지정합니다. 동일한 IAM 역할이 다른 인스턴스와 연결된 경우에는 다른 인스턴스에서 이러한 작업을 수행할 수 없습니다.

`ec2:SourceInstanceARN` 키는 AWS 전체 범위 조건 키이므로 Amazon EC2뿐 아니라 다른 서비스 작업에 사용할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeVolumes",
                "s3>ListAllMyBuckets",
                "dynamodb>ListTables",
                "rds:DescribeDBInstances"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "ArnEquals": {
                    "ec2:SourceInstanceARN": "arn:aws:ec2:region:account:instance/i-093452212644b0dd6"
                }
            }
        }
    ]
}
```

예제: 시작 템플릿 사용

아래 정책은 특정 시작 템플릿(`lt-09477bcd97b0d3abc`)에서만 시작 템플릿 버전을 생성하고 시작 템플릿을 수정할 수 있도록 허용합니다. 사용자는 다른 시작 템플릿은 사용할 수 없습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "ec2>CreateLaunchTemplateVersion",
                "ec2:ModifyLaunchTemplate"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:ec2:region:account:launchTemplate/lt-09477bcd97b0d3abc"
            ]
        }
    ]
}
```

```
        "Resource": "arn:aws:ec2:region:account:launch-template/lt-09477bcd97b0d3abc"
    }
]
```

아래 정책은 시작 템플릿에 Purpose=Testing 태그가 지정되어 있는 경우에 모든 시작 템플릿 및 시작 템플릿 버전을 삭제할 수 있도록 허용합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:DeleteLaunchTemplate",
        "ec2:DeleteLaunchTemplateVersions"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:ec2:region:account:launch-template/*",
      "Condition": {
        "StringEquals": {
          "ec2:ResourceTag/Purpose": "Testing"
        }
      }
    ]
  }
}
```

인스턴스 메타데이터 작업

다음 정책은 사용자가 인스턴스 메타데이터 서비스 버전 2(IMDSv2)를 사용하여 [인스턴스 메타데이터 \(p. 576\)](#)만 검색할 수 있도록 합니다. 다음 정책 4개를 결합하여 명령문 4개를 가진 정책 하나로 만들 수 있습니다. 정책이 하나로 결합되면 이 정책을 서비스 제어 정책(SCP)으로 사용할 수 있습니다. 이 정책은 기존 IAM 정책에 적용한 deny 정책(기존 권한 취소 및 제한)이나 계정, 조직 단위(OU) 또는 전체 조직에서 전역적으로 적용되는 SCP와 똑같이 원활하게 적용됩니다.

Note

보안 주체에 RunInstances로 인스턴스를 시작할 수 있는 권한을 제공하는 정책과 함께 다음 RunInstances 메타데이터 옵션 정책을 사용해야 합니다. 또한 보안 주체에 RunInstances 권한이 없으면 인스턴스를 시작할 수 없습니다. 자세한 내용은 [인스턴스 작업 \(p. 847\)](#) 및 [인스턴스 시작 \(RunInstances\) \(p. 858\)](#)의 정책을 참조하십시오.

Important

Auto Scaling 그룹을 사용할 때 모든 새 인스턴스에서 IMDSv2를 사용해야 하는 경우 Auto Scaling 그룹에서 시작 템플릿을 사용해야 합니다.

Auto Scaling 그룹이 시작 템플릿을 사용하는 경우 새 Auto Scaling 그룹이 생성될 때 IAM 보안 주체의 ec2:RunInstances 권한이 확인됩니다. 또한 새 시작 템플릿이나 새 버전의 시작 템플릿을 사용하도록 기존 Auto Scaling 그룹이 업데이트될 때도 확인됩니다.

RunInstances에 대한 IAM 보안 주체의 IMDSv1 사용에 대한 제한은 시작 템플릿을 사용하는 Auto Scaling 그룹이 생성 또는 업데이트될 때에만 확인됩니다. Latest 또는 Default 시작 템플릿을 사용하도록 구성된 Auto Scaling 그룹의 경우 새 버전의 시작 템플릿을 생성할 때 권한이 확인되지 않습니다. 권한을 확인하려면 특정 버전의 시작 템플릿을 사용하도록 Auto Scaling 그룹을 구성해야 합니다.

Auto Scaling 그룹에서 시작한 인스턴스에 대해 IMDSv2를 사용하도록 하려면 다음과 같은 추가 단계가 필요합니다.

1. 생성되는 새 보안 주체에 대해 IAM 권한 경계 또는 서비스 제어 정책(SCP)을 사용하여 조직의 모든 계정에 대해 시작 구성의 사용을 비활성화합니다. Auto Scaling 그룹 권한이 있는 기존 IAM 보

안 주체의 경우 이 조건 키로 연결된 정책을 업데이트합니다. 시작 구성의 사용을 비활성화하려면 값이 `null`로 지정된 "autoscaling:LaunchConfigurationName" 조건 키를 사용하여 관련 SCP, 권한 경계 또는 IAM 정책을 생성하거나 수정합니다.

2. 새 시작 템플릿의 경우 시작 템플릿에서 인스턴스 메타데이터 옵션을 구성합니다. 기존 시작 템플릿의 경우 새 버전의 시작 템플릿을 만들고 새 버전에서 인스턴스 메타데이터 옵션을 구성합니다.
3. 보안 주체에게 시작 템플릿을 사용할 권한을 부여하는 정책에서 "`autoscaling:LaunchTemplateVersionSpecified`": "true"를 지정하여 `$latest` 및 `$default`의 연결을 제한합니다. 특정 버전의 시작 템플릿으로 사용을 제한하면 인스턴스 메타데이터 옵션이 구성된 버전을 사용하여 새 인스턴스가 시작되도록 할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling API 참조의 [LaunchTemplateSpecification](#), 특히 `Version` 파라미터를 참조하십시오.
4. 시작 구성을 사용하는 Auto Scaling 그룹의 경우 시작 구성을 시작 템플릿으로 바꿉니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 [시작 구성의 시작 템플릿으로 바꾸기](#)를 참조하십시오.
5. 시작 템플릿을 사용하는 Auto Scaling 그룹의 경우 인스턴스 메타데이터 옵션이 구성된 새 시작 템플릿을 사용하거나 인스턴스 메타데이터 옵션이 구성된 현재 시작 템플릿의 새 버전을 사용해야 합니다. 자세한 내용은 AWS CLI Command Reference의 [update-auto-scaling-group](#)을 참조하십시오.

예제:

- [IMDSv2의 사용 요구 \(p. 880\)](#)
- [최대 흡 제한 지정 \(p. 880\)](#)
- [인스턴스 메타데이터 옵션을 수정할 수 있는 사용자 제한 \(p. 881\)](#)
- [IMDSv2에서 역할 자격 증명을 검색하도록 요구 \(p. 881\)](#)

IMDSv2의 사용 요구

다음 정책에서는 IMDSv2("ec2:MetadataHttpTokens": "required"로 표시) 사용을 요구하도록 인스턴스도 옵트인되지 않으면 RunInstances API를 호출할 수 없도록 지정합니다. 인스턴스가 IMDSv2를 요구하도록 지정하지 않으면 RunInstances API를 호출할 때 UnauthorizedOperation 오류가 발생합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "RequireImdsV2",  
            "Effect": "Deny",  
            "Action": "ec2:RunInstances",  
            "Resource": "arn:aws:ec2:*:instance/*",  
            "Condition": {  
                "StringNotEquals": {  
                    "ec2:MetadataHttpTokens": "required"  
                }  
            }  
        }  
    ]  
}
```

최대 흡 제한 지정

다음 정책은 흡 제한도 지정하지 않으면(흡 제한은 3을 초과할 수 없음) RunInstances API를 호출할 수 없도록 지정합니다. 그렇게 하지 않으면 RunInstances API를 호출할 때 UnauthorizedOperation 오류가 발생합니다.

Note

다음 정책과 앞에 나온 정책이 SCP를 통해 계정에 적용되면 EC2 콘솔을 사용하여 인스턴스를 시작할 수 없습니다. 콘솔에서 아직 `MetadataHttpTokens` 파라미터와 `MetadataHttpPutResponseHopLimit` 파라미터를 지원하지 않기 때문입니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "MaxImdsHopLimit",  
            "Effect": "Deny",  
            "Action": "ec2:RunInstances",  
            "Resource": "arn:aws:ec2::*:instance/*",  
            "Condition": {  
                "NumericGreaterThan": {  
                    "ec2:MetadataHttpPutResponseHopLimit": "3"  
                }  
            }  
        }  
    ]  
}
```

인스턴스 메타데이터 옵션을 수정할 수 있는 사용자 제한

다음 정책은 일반 관리자들이 인스턴스 메타데이터 옵션을 수정할 수 있는 기능을 제거하고 `ec2-imds-admins` 역할이 있는 사용자만 변경할 수 있도록 허용합니다. `ec2-imds-admins` 역할이 아닌 보안 주체가 `ModifyInstanceMetadataOptions` API를 호출하려고 하면 `UnauthorizedOperation` 오류가 발생합니다. 이 명령문을 사용하여 `ModifyInstanceMetadataOptions` API 사용을 제어할 수 있습니다. 지금은 `ModifyInstanceMetadataOptions` API의 세부적인 액세스 제어(조건)가 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "AllowOnlyImdsAdminsToModifySettings",  
            "Effect": "Deny",  
            "Action": "ec2:ModifyInstanceMetadataOptions",  
            "Resource": "*",  
            "Condition": {  
                "StringNotLike": {  
                    "aws:PrincipalARN": "arn:aws:iam::*:role/ec2-imds-admins"  
                }  
            }  
        }  
    ]  
}
```

IMDSv2에서 역할 자격 증명을 검색하도록 요구

다음 정책은 이 정책이 역할에 적용되고, EC2 서비스가 이 역할을 맡으면, 그 결과로 생긴 자격 증명이 요청에 서명하는 데 사용되며, IMDSv2에서 검색한 EC2 역할 자격 증명으로 요청에 서명해야 한다고 지정합니다. 그렇게 하지 않으면 모든 API 호출에서 `UnauthorizedOperation` 오류가 발생합니다. 이 명령문/정책은 일반적으로 적용됩니다. EC2 역할 자격 증명으로 요청에 서명하지 않으면 요청은 아무 효과가 없습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "RequireAllEc2RolesToUseV2",  
            "Effect": "Deny",  
            "Action": "sts:AssumeRole",  
            "Resource": "*"  
        }  
    ]  
}
```

```
"Effect": "Deny",
"Action": "*",
"Resource": "*",
"Condition": {
    "NumericLessThan": {
        "ec2:RoleDelivery": "2.0"
    }
}
}
```

Amazon EC2 콘솔 작업을 위한 예제 정책

IAM 콘솔에서 Amazon EC2 정책을 사용하여 특정 리소스를 조회하고 관련 작업을 수행할 권한을 부여할 수 있습니다. 이전 섹션의 예제 정책을 사용할 수 있지만 해당 정책은 AWS CLI 또는 AWS SDK를 통한 요청에 맞게 설계되었습니다. 콘솔에서는 추가적인 API 작업을 통해 해당 기능을 구현하므로 이러한 정책이 예상과 다르게 작동할 수 있습니다. 예를 들어 `DescribeVolumes` API 작업만 사용할 권한을 갖는 경우 콘솔에서 볼륨을 조회하려고 하면 오류가 발생합니다. 이 섹션에서는 콘솔의 특정 부분을 사용하도록 허용하는 정책을 보여 줍니다.

Tip

콘솔에서 작업을 수행하는 데 필요한 API 작업을 파악하려는 경우 AWS CloudTrail 등의 서비스를 사용할 수 있습니다. 자세한 내용은 [AWS CloudTrail User Guide](#)를 참조하십시오. 정책에서 특정 리소스를 생성하거나 수정할 권한을 부여하지 않는 경우 콘솔에 진단 정보가 포함된 인코딩 메시지가 표시됩니다. AWS STS의 `DecodeAuthorizationMessage` API 작업이나 AWS CLI의 `decode-authorization-message` 명령을 사용하여 메시지를 디코딩할 수 있습니다.

예제

- 예: 읽기 전용 액세스 (p. 882)
- 예제: EC2 시작 마법사 사용 (p. 883)
- 예제: 볼륨 작업 (p. 886)
- 예제: 보안 그룹 작업 (p. 887)
- 예제: 탄력적 IP 주소 작업 (p. 889)
- 예제: 예약 인스턴스 작업 (p. 889)

Amazon EC2 콘솔용 정책을 생성하는 방법에 대한 자세한 내용은 AWS 보안 블로그 게시물 [Granting Users Permission to Work in the Amazon EC2 Console](#)을 참조하십시오.

예: 읽기 전용 액세스

사용자가 Amazon EC2 콘솔에서 모든 리소스를 조회하도록 허용하려면 다음 예제와 같은 정책을 사용합니다. 예: 읽기 전용 액세스 (p. 846). 다른 명령문에서 해당 권한을 부여하지 않는 경우 이러한 리소스에 대해 작업을 수행하거나 새 리소스를 생성할 수는 없습니다.

인스턴스, AMI 및 스냅샷 조회

리소스 중 일부에 대한 읽기 전용 액세스를 제공할 수도 있습니다. 이렇게 하려면 `ec2:Describe` API 작업에서 *와 일드카드를 구체적인 리소스별 `ec2:Describe` 작업으로 대체합니다. 다음 정책은 사용자가 Amazon EC2 콘솔에서 모든 인스턴스, AMI 및 스냅샷을 조회하도록 허용합니다. `ec2:DescribeTags` 작업에서는 사용자가 퍼블릭 AMI를 조회할 수 있습니다. 콘솔에 퍼블릭 AMI를 표시하려면 태그 지정 정보가 필요하지만 사용자가 프라이빗 AMI만 조회하도록 하려면 이 작업을 제거할 수도 있습니다.

```
{
    "Version": "2012-10-17",
```

```
"Statement": [{}  
    "Effect": "Allow",  
    "Action": [  
        "ec2:DescribeInstances",  
        "ec2:DescribeImages",  
        "ec2:DescribeTags",  
        "ec2:DescribeSnapshots"  
    ],  
    "Resource": "*"  
}  
]  
}
```

Note

Amazon EC2 ec2:Describe* API 작업은 리소스별 권한을 지원하지 않으므로 사용자가 콘솔에서 조회할 수 있는 리소스를 개별적으로 제어할 수는 없습니다. 따라서 위 명령문의 Resource 요소에 * 와일드카드가 필요합니다. Amazon EC2 API 작업에 사용할 수 있는 ARN에 대한 자세한 내용은 IAM 사용 설명서의 [Amazon EC2에 사용되는 작업, 리소스 및 조건 키](#)를 참조하십시오.

인스턴스 및 CloudWatch 측정치 조회

다음 정책은 사용자로 하여금 인스턴스 페이지의 모니터링 탭에 있는 CloudWatch 경보 및 지표뿐만 아니라 Amazon EC2 콘솔의 인스턴스까지도 조회할 수 있도록 허용합니다. Amazon EC2 콘솔은 CloudWatch API를 이용해 경보 및 측정치를 표시하므로, 반드시 사용자에게 cloudwatch:DescribeAlarms 및 cloudwatch:GetMetricStatistics 작업을 사용할 수 있는 권한을 부여해야 합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [{}  
        "Effect": "Allow",  
        "Action": [  
            "ec2:DescribeInstances",  
            "cloudwatch:DescribeAlarms",  
            "cloudwatch:GetMetricStatistics"  
        ],  
        "Resource": "*"  
    ]  
}
```

예제: EC2 시작 마법사 사용

Amazon EC2 시작 마법사는 인스턴스 구성 및 시작 옵션을 포함하는 일련의 화면으로 구성됩니다. 사용자가 마법사의 옵션을 사용할 수 있도록 정책에 API 작업 사용 권한이 포함되어야 합니다. 해당 작업 사용 권한이 정책에 포함되지 않으면 마법사의 일부 항목이 제대로 로드되지 않고 사용자가 시작을 완료할 수 없습니다.

기본 시작 마법사에 액세스

성공적으로 시작을 완료하려면 사용자에게 ec2:RunInstances API 작업 및 최소한 다음과 같은 API 작업 사용 권한을 부여해야 합니다.

- ec2:DescribeImages: AMI를 조회하고 선택합니다.
- ec2:DescribeVpcs: 사용 가능한 네트워크 옵션을 조회합니다.
- ec2:DescribeSubnets: 선택한 VPC에 대한 모든 사용 가능한 서브넷을 조회합니다.
- ec2:DescribeSecurityGroups 또는 ec2>CreateSecurityGroup: 기존 보안 그룹을 조회하고 선택하거나 새로 생성합니다.
- ec2:DescribeKeyPairs 또는 ec2>CreateKeyValuePair: 기존 키 페어를 선택하거나 새로 생성합니다.

- **ec2:AuthorizeSecurityGroupIngress:** 인바운드 규칙을 추가합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:DescribeInstances",  
                "ec2:DescribeImages",  
                "ec2:DescribeKeyPairs",  
                "ec2:DescribeVpcs",  
                "ec2:DescribeSubnets",  
                "ec2:DescribeSecurityGroups",  
                "ec2:CreateSecurityGroup",  
                "ec2:AuthorizeSecurityGroupIngress",  
                "ec2:CreateKeyPair"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": "ec2:RunInstances",  
            "Resource": "*"  
        }  
    ]  
}
```

정책에 API 작업을 추가하여 다음과 같이 사용자에게 더 많은 옵션을 제공할 수 있습니다.

- **ec2:DescribeAvailabilityZones:** 특정 가용 영역을 조회하고 선택합니다.
- **ec2:DescribeNetworkInterfaces:** 선택한 서브넷의 기존 네트워크 인터페이스를 조회하고 선택합니다.
- VPC 보안 그룹에 아웃바운드 규칙을 추가하려면 **ec2:AuthorizeSecurityGroupEgress** API 작업 사용 권한이 부여되어야 합니다. 기존 규칙을 수정 또는 삭제하려면 관련 **ec2:RevokeSecurityGroup*** API 작업 사용 권한이 부여되어야 합니다.
- **ec2:CreateTags:** **RunInstances**에 의해 생성된 리소스에 태그 지정. 자세한 내용은 [생성 시 리소스 태그 지정에 대한 권한 부여 \(p. 843\)](#) 단원을 참조하십시오. 이 작업을 사용할 권한이 없는 사용자가 시작 마법사의 태그 지정 페이지에서 태그를 지정하려 시도하는 경우, 시작은 실패합니다.

Important

사용자에게 **ec2:CreateTags** 작업을 사용할 수 있는 권한을 부여하면 **ec2:ResourceTag** 조건 키를 사용하여 다른 리소스의 사용을 제한하지 못할 수 있으므로 주의해야 합니다. 사용자가 **ec2:CreateTags** 작업을 사용할 수 있는 권한을 부여받으면 리소스의 태그를 변경하여 제한을 우회할 수 있습니다. 자세한 내용은 [리소스 태그를 사용하여 EC2 리소스에 대한 액세스 제어 \(p. 845\)](#) 단원을 참조하십시오.

- AMI를 선택하는 데 시스템 관리자 파라미터를 사용하려면 정책에 **ssm:DescribeParameters** 및 **ssm:GetParameters**를 추가해야 합니다. **ssm:DescribeParameters**는 IAM 사용자에게 시스템 관리자 파라미터를 보고 선택할 수 있는 권한을 부여하며, **ssm:GetParameters**는 IAM 사용자에게 시스템 관리자 파라미터의 값을 가져올 수 있는 권한을 부여합니다. 특정 시스템 관리자 파라미터에 대한 액세스를 제한할 수도 있습니다. 자세한 내용은 이 단원의 뒷부분에 있는 특정 시스템 관리자 파라미터에 대한 액세스 제한을 참조하십시오.

현재 Amazon EC2 **Describe*** API 작업은 리소스별 권한을 지원하지 않으므로 사용자가 시작 마법사에서 조회할 수 있는 리소스를 개별적으로 제한할 수는 없습니다. 그러나 **ec2:RunInstances** API 작업에 리소스별 권한을 적용하여 사용자가 인스턴스를 시작하는 데 사용 가능한 리소스를 제한할 수 있습니다. 사용자가 사용 권한이 없는 옵션을 선택하면 시작에 실패합니다.

특정 인스턴스 유형, 서브넷 및 리전에 대한 액세스 제한

다음 정책은 Amazon이 소유한 AMI를 사용하여 t2.micro 인스턴스를 시작하되 특정 서브넷 (`subnet-1a2b3c4d`)으로만 시작하도록 허용합니다. 사용자는 sa-east-1 지역에서만 시작할 수 있습니다. 사용자가 다른 지역을 선택하거나 시작 마법사에서 다른 인스턴스 유형, AMI 또는 서브넷을 선택하면 시작에 실패합니다.

첫 번째 명령문은 위 예제에 설명된 대로 사용자가 시작 마법사에서 옵션을 조회할 권한을 부여합니다. 두 번째 명령문은 `ec2:RunInstances` 작업에서 네트워크 인터페이스, 볼륨, 키 페어, 보안 그룹 및 서브넷 리소스를 사용할 권한을 부여합니다. 이 권한은 인스턴스를 VPC로 시작하는 데 필요합니다. `ec2:RunInstances` 작업 사용에 대한 자세한 내용은 [인스턴스 시작\(RunInstances\) \(p. 858\)](#) 단원을 참조하십시오. 세 번째, 네 번째 명령문은 각각 인스턴스 및 AMI 리소스 사용 권한을 부여하지만 인스턴스가 t2.micro 인스턴스이고 AMI를 Amazon이 소유한 경우로 한정합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [{  
        "Effect": "Allow",  
        "Action": [  
            "ec2:DescribeInstances",  
            "ec2:DescribeImages",  
            "ec2:DescribeKeyPairs",  
            "ec2:CreateKeyPair",  
            "ec2:DescribeVpcs",  
            "ec2:DescribeSubnets",  
            "ec2:DescribeSecurityGroups",  
            "ec2:CreateSecurityGroup",  
            "ec2:AuthorizeSecurityGroupIngress"  
        ],  
        "Resource": "*"  
    },  
    {  
        "Effect": "Allow",  
        "Action": "ec2:RunInstances",  
        "Resource": [  
            "arn:aws:ec2:sa-east-1:111122223333:network-interface/*",  
            "arn:aws:ec2:sa-east-1:111122223333:volume/*",  
            "arn:aws:ec2:sa-east-1:111122223333:key-pair/*",  
            "arn:aws:ec2:sa-east-1:111122223333:security-group/*",  
            "arn:aws:ec2:sa-east-1:111122223333:subnet/subnet-1a2b3c4d"  
        ]  
    },  
    {  
        "Effect": "Allow",  
        "Action": "ec2:RunInstances",  
        "Resource": [  
            "arn:aws:ec2:sa-east-1:111122223333:instance/*"  
        ],  
        "Condition": {  
            "StringEquals": {  
                "ec2:InstanceType": "t2.micro"  
            }  
        }  
    },  
    {  
        "Effect": "Allow",  
        "Action": "ec2:RunInstances",  
        "Resource": [  
            "arn:aws:ec2:sa-east-1::image/ami-*"  
        ],  
        "Condition": {  
            "StringEquals": {  
                "ec2:Owner": "amazon"  
            }  
        }  
    }  
}
```

```
        }
    ]
}
```

특정 시스템 관리자 파라미터에 대한 액세스 제한

다음 정책은 특정 이름의 시스템 관리자 파라미터를 사용할 수 있는 액세스 권한을 부여합니다.

첫 번째 문은 시작 마법사에서 AMI를 선택할 때 시스템 관리자 파라미터를 볼 수 있는 권한을 사용자에게 부여하며, 두 번째 문은 사용자에게 prod-*라는 이름이 지정된 파라미터만 사용할 수 있는 권한을 부여합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeParameters"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameters"
            ],
            "Resource": "arn:aws:ssm:us-east-2:123456123:parameter/prod-*"
        }
    ]
}
```

예제: 볼륨 작업

다음 정책은 볼륨을 조회 및 생성하고 특정 인스턴스에 볼륨을 연결 및 분리할 권한을 부여합니다.

사용자는 "purpose=test" 태그가 있는 인스턴스에 볼륨을 연결하고 해당 인스턴스에서 볼륨을 분리할 수 있습니다. Amazon EC2 콘솔을 사용하여 볼륨을 연결하려는 경우 사용자에게 ec2:DescribeInstances 작업 사용 권한을 부여하는 것이 좋습니다. 이렇게 하면 볼륨 연결 대화 상자의 미리 구성된 목록에서 인스턴스를 선택할 수 있습니다. 그러나 이렇게 하면 사용자가 콘솔의 인스턴스 페이지에서 모든 인스턴스를 조회할 수 있으므로 이 작업을 생략할 수 있습니다.

첫 번째 명령문에서 ec2:DescribeAvailabilityZones 작업은 볼륨을 생성할 때 사용자가 가용 영역을 선택할 수 있도록 하는 데 필요합니다.

사용자는 볼륨 생성 중 또는 생성 후 본인이 작성한 볼륨에 태그를 지정할 수 없습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeVolumes",
                "ec2:DescribeAvailabilityZones",
                "ec2>CreateVolume",
                "ec2:DescribeInstances"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AttachVolume"
            ],
            "Resource": "arn:aws:ec2:us-east-2:123456123:volume/*"
        }
    ]
}
```

```
"Action": [
    "ec2:AttachVolume",
    "ec2:DetachVolume"
],
"Resource": "arn:aws:ec2:region:111122223333:instance/*",
"Condition": {
    "StringEquals": {
        "ec2:ResourceTag/purpose": "test"
    }
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:AttachVolume",
        "ec2:DetachVolume"
    ],
    "Resource": "arn:aws:ec2:region:111122223333:volume/*"
}
]
```

예제: 보안 그룹 작업

보안 그룹 조회와 규칙의 추가 및 삭제

다음 정책은 사용자가 Amazon EC2 콘솔에서 보안 그룹을 조회하고 인바운드 및 아웃바운드 규칙을 추가 및 제거하며 `Department=Test` 태그가 있는 기준 보안 그룹에 대한 규칙 설명을 수정할 권한을 부여합니다.

첫 번째 명령문에서 `ec2:DescribeTags` 작업은 사용자가 콘솔에서 태그를 조회하도록 허용하므로 사용자가 수정 가능한 보안 그룹을 쉽게 식별할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeTags"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AuthorizeSecurityGroupIngress",
                "ec2:RevokeSecurityGroupIngress",
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:RevokeSecurityGroupEgress",
                "ec2:UpdateSecurityGroupRuleDescriptionsIngress",
                "ec2:UpdateSecurityGroupRuleDescriptionsEgress"
            ],
            "Resource": [
                "arn:aws:ec2:region:111122223333:security-group/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/Department": "Test"
                }
            }
        }
    ]
}
```

보안 그룹 생성 대화 상자 작업

사용자가 Amazon EC2 콘솔에서 보안 그룹 생성 대화 상자를 사용하도록 허용하는 정책을 생성할 수 있습니다. 이 대화 상자를 사용하려면 최소한 다음과 같은 API 작업 사용 권한을 부여해야 합니다.

- `ec2:CreateSecurityGroup`: 새 보안 그룹을 생성합니다.
- `ec2:DescribeVpcs`: VPC 목록에서 기존 VPC의 목록을 조회합니다.

이 권한이 있으면 사용자가 새 보안 그룹을 생성할 수 있지만 규칙을 추가할 수는 없습니다. 보안 그룹 생성 대화 상자에서 규칙 관련 작업을 수행하려면 정책에 다음 API 작업을 추가합니다.

- `ec2:AuthorizeSecurityGroupIngress`: 인바운드 규칙을 추가합니다.
- `ec2:AuthorizeSecurityGroupEgress`: VPC 보안 그룹에 아웃바운드 규칙을 추가합니다.
- `ec2:RevokeSecurityGroupIngress`: 기존 인바운드 규칙을 수정하거나 삭제합니다. 이 권한은 사용자가 콘솔에서 새로 복사 기능을 사용하도록 허용하려는 경우 유용합니다. 이 기능은 보안 그룹 생성 대화 상자를 열고 선택한 보안 그룹과 같은 규칙을 미리 입력합니다.
- `ec2:RevokeSecurityGroupEgress`: VPC 보안 그룹의 아웃바운드 규칙을 수정하거나 삭제합니다. 이 권한은 모든 아웃바운드 트래픽을 허용하는 기본 아웃바운드 규칙을 사용자가 수정 또는 삭제하도록 허용하는 데 유용합니다.
- `ec2>DeleteSecurityGroup`: 잘못된 규칙을 저장할 수 없도록 합니다. 콘솔에서 먼저 보안 그룹을 만든 후 지정된 규칙을 추가합니다. 규칙이 잘못된 경우 작업이 실패하고 콘솔이 보안 그룹을 삭제하려고 합니다. 사용자는 보안 그룹 생성 대화 상자에 남아 있기 때문에 잘못된 규칙을 수정한 후 보안 그룹을 다시 생성해 볼 수 있습니다. 이 API 작업은 필수적이지는 않지만 해당 사용 권한을 부여하지 않으면 사용자가 잘못된 규칙이 포함된 보안 그룹을 생성하려고 할 때 규칙이 없는 보안 그룹이 생성되며, 사용자가 이후에 규칙을 추가해야 합니다.
- `ec2:UpdateSecurityGroupRuleDescriptionsIngress`: 수신(인바운드) 보안 그룹 규칙에 대한 설명을 추가하거나 업데이트합니다.
- `ec2:UpdateSecurityGroupRuleDescriptionsEgress`: 발신(아웃바운드) 보안 그룹 규칙에 대한 설명을 추가하거나 업데이트합니다.

현재 `ec2:CreateSecurityGroup` API 작업은 리소스 수준 권한을 지원하지 않지만 `ec2:AuthorizeSecurityGroupIngress` 및 `ec2:AuthorizeSecurityGroupEgress` 작업에 리소스 수준 권한을 적용하여 사용자의 규칙 생성 방법을 제어할 수 있습니다.

다음 정책은 보안 그룹 생성 대화 상자를 사용하고 특정 VPC(`vpc-1a2b3c4d`)에 연결된 보안 그룹에 인바운드 및 아웃바운드 규칙을 생성할 권한을 부여합니다. 사용자는 EC2-Classic 또는 다른 VPC의 보안 그룹을 생성할 수 있지만 규칙을 추가할 수는 없습니다. 마찬가지로 VPC `vpc-1a2b3c4d`에 연결되지 않은 기존 보안 그룹에는 규칙을 추가할 수는 없습니다. 또한 콘솔에서 모든 보안 그룹을 조회할 권리가 부여됩니다. 따라서 사용자가 인바운드 규칙을 추가할 수 있는 보안 그룹을 쉽게 식별할 수 있습니다. 또한 이 정책은 VPC `vpc-1a2b3c4d`에 연결된 보안 그룹을 삭제할 권한을 부여합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:DescribeSecurityGroups",  
                "ec2:CreateSecurityGroup",  
                "ec2:DescribeVpcs"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:DeleteSecurityGroup",  
                "ec2:RevokeSecurityGroupIngress",  
                "ec2:RevokeSecurityGroupEgress"  
            ],  
            "Resource": "vpc-1a2b3c4d"  
        }  
    ]  
}
```

```
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:AuthorizeSecurityGroupEgress"
    ],
    "Resource": "arn:aws:ec2:region:111122223333:security-group/*",
    "Condition": {
        "ArnEquals": {
            "ec2:Vpc": "arn:aws:ec2:region:111122223333:vpc/vpc-1a2b3c4d"
        }
    }
}
```

예제: 탄력적 IP 주소 작업

사용자가 Amazon EC2 콘솔에서 탄력적 IP 주소를 볼 수 있도록 하려면 사용자에게 `ec2:DescribeAddresses` 작업을 사용할 수 있는 권한을 부여해야 합니다.

사용자에게 탄력적 IP 주소 관련 작업을 허용하려면 정책에 다음 작업을 추가합니다.

- `ec2:AllocateAddress`: 탄력적 IP 주소를 할당합니다.
- `ec2:ReleaseAddress`: 탄력적 IP 주소를 해제합니다.
- `ec2:AssociateAddress`: 인스턴스 또는 네트워크 인터페이스에 탄력적 IP 주소를 연결합니다.
- `ec2:DescribeNetworkInterfaces` 및 `ec2:DescribeInstances`: 주소 연결 화면에서 작업합니다.
탄력적 IP 주소를 연결할 수 있는 네트워크 인터페이스나 가용 인스턴스가 화면에 표시됩니다.
- `ec2:DisassociateAddress`: 인스턴스 또는 네트워크 인터페이스에서 탄력적 IP 주소를 분리합니다.

다음 정책을 통해 사용자는 탄력적 IP 주소를 확인하고 인스턴스에 할당, 연결할 수 있습니다. 사용자는 탄력적 IP 주소를 네트워크 인터페이스에 연결하거나 탄력적 IP 주소 연결을 끊거나 릴리스할 수 없습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAddresses",
                "ec2:AllocateAddress",
                "ec2:DescribeInstances",
                "ec2:AssociateAddress"
            ],
            "Resource": "*"
        }
    ]
}
```

예제: 예약 인스턴스 작업

다음 정책을 IAM 사용자에 연결할 수 있습니다. 그렇게 하면 사용자가 계정의 예약 인스턴스를 보고 수정할 수 있을 뿐만 아니라 AWS Management 콘솔에서 새 예약 인스턴스를 구매할 수 있는 액세스 권한을 갖게 됩니다.

이 정책을 통해 사용자는 계정의 모든 예약 인스턴스 및 온디マン드 인스턴스를 볼 수 있습니다. 개별 예약 인스턴스에 대해서는 리소스 수준 권한을 설정할 수 없습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [{


```

```
"Effect": "Allow",
"Action": [
    "ec2:DescribeReservedInstances",
    "ec2:ModifyReservedInstances",
    "ec2:PurchaseReservedInstancesOffering",
    "ec2:DescribeInstances",
    "ec2:DescribeInstanceTypes",
    "ec2:DescribeAvailabilityZones",
    "ec2:DescribeReservedInstancesOfferings"
],
"Resource": "*"
}
]
```

ec2:DescribeAvailabilityZones 작업은 Amazon EC2 콘솔이 예약 인스턴스를 구매할 수 있는 가용 영역에 대한 정보를 표시하도록 하는 데 필수적입니다. ec2:DescribeInstances 작업은 필수적이지는 않지만 사용자가 계정에서 인스턴스를 보고, 정확한 사양에 맞추기 위해 예약을 구매할 수 있도록 해줍니다.

API 작업을 조정해 사용자 액세스를 제한할 수 있습니다. 예를 들어 ec2:DescribeInstances와 ec2:DescribeAvailabilityZones를 제거하면 사용자가 읽기 전용 액세스 권한만을 갖게 됩니다.

Amazon EC2의 IAM 역할

애플리케이션은 AWS 자격 증명으로 API 요청에 서명해야 합니다. 따라서 애플리케이션 개발자는 EC2 인스턴스에서 실행되는 인스턴스의 자격 증명을 관리할 전략을 수립해야 합니다. 예를 들어 AWS 자격 증명을 인스턴스에 안전하게 배포하여 다른 사용자로부터 보호하는 한편 해당 인스턴스의 애플리케이션이 자격 증명을 사용하여 요청에 서명하도록 할 수 있습니다. 그러나 각 인스턴스에 자격 증명을 안전하게 배포하기란 쉽지 않으며, 스팟 인스턴스와 같이 AWS에서 자동으로 생성하는 인스턴스 또는 Auto Scaling 그룹의 인스턴스에 대해서는 특히 어렵습니다. 또한 AWS 자격 증명을 교체할 때 각 인스턴스의 자격 증명을 업데이트할 수 있어야 합니다.

애플리케이션이 사용하는 보안 자격 증명을 직접 관리할 필요 없이 인스턴스의 애플리케이션에서 안전하게 API 요청을 전송할 수 있도록 IAM 역할을 설계했습니다. AWS 자격 증명을 생성하고 배포하는 대신 다음과 같이 IAM 역할을 사용하여 API 요청 전송 권한을 위임할 수 있습니다.

1. IAM 역할을 생성합니다.
2. 역할을 수행할 수 있는 계정 또는 AWS 서비스를 정의합니다.
3. 역할을 수행하면서 애플리케이션이 사용할 수 있는 API 작업 및 리소스를 정의합니다.
4. 인스턴스를 시작할 때 역할을 지정하거나, 기존 인스턴스에 역할을 연결합니다.
5. 애플리케이션에서 임시 자격 증명 세트를 검색하여 사용하도록 합니다.

예를 들어 IAM 역할을 사용하여 인스턴스에서 실행되며 Amazon S3의 버킷을 사용해야 하는 애플리케이션에 해당 권한을 부여할 수 있습니다. JSON 형식으로 정책을 생성하여 IAM 역할에 권한을 지정할 수 있습니다. 이 방법은 IAM 사용자를 대상으로 정책을 생성할 때와 비슷합니다. 역할을 변경하면 모든 인스턴스에 변경 내용이 전파됩니다.

단일 인스턴스에 여러 IAM 역할을 연결할 수 없지만 여러 인스턴스에 단일 IAM 역할을 연결할 수 있습니다. IAM 역할 생성 및 사용에 대한 자세한 내용은 IAM 사용 설명서에서 [역할](#)을 참조하십시오.

IAM 정책에 리소스 수준 권한을 적용하여 사용자가 인스턴스에 IAM 역할을 연결, 교체 또는 분리할 수 있는 권한을 제어할 수 있습니다. 자세한 내용은 [Amazon EC2 API 작업에 지원되는 리소스 수준 권한 \(p. 840\)](#) 및 다음 예제: [예제: IAM 역할 작업 \(p. 876\)](#) 단원을 참조하십시오.

목차

- [인스턴스 프로파일 \(p. 891\)](#)
- [인스턴스 메타데이터에서 보안 자격 증명 검색 \(p. 891\)](#)

- IAM 사용자에게 IAM 역할을 인스턴스에 전달할 수 있는 권한 부여 (p. 892)
- IAM 역할 작업 (p. 892)

인스턴스 프로파일

Amazon EC2에서는 인스턴스 프로파일을 IAM 역할의 컨테이너로 사용합니다. IAM 콘솔을 사용하여 IAM 역할을 생성하면 인스턴스 프로파일이 자동으로 생성되고 해당 역할과 동일한 이름이 지정됩니다. Amazon EC2 콘솔을 사용하여 IAM 역할로 인스턴스를 시작하거나 인스턴스에 IAM 역할을 연결하는 경우 인스턴스 프로파일 이름 목록을 기반으로 역할을 선택합니다.

AWS CLI, API 또는 AWS SDK를 사용하여 역할을 생성하면 역할과 인스턴스 프로파일이 별개의 작업으로 생성되며 이름은 각각 다를 수 있습니다. AWS CLI, API 또는 AWS SDK를 사용하여 IAM 역할로 인스턴스를 시작하거나 인스턴스에 IAM 역할을 연결하는 경우 인스턴스 프로파일 이름을 지정합니다.

인스턴스 프로파일은 하나의 IAM 역할만 포함할 수 있습니다. 이 한도는 늘릴 수 없습니다.

자세한 내용은 IAM 사용 설명서에서 [인스턴스 프로파일](#)을 참조하십시오.

인스턴스 메타데이터에서 보안 자격 증명 검색

인스턴스의 애플리케이션은 인스턴스 메타데이터 항목 `iam/security-credentials/role-name`에서 역할이 제공하는 보안 자격 증명을 검색합니다. 역할에 연결된 보안 자격 증명을 통해 역할에 정의한 작업 및 리소스에 대한 권한이 애플리케이션에 부여됩니다. 이러한 보안 자격 증명은 임시로 발급되며 자동으로 교체됩니다. 이전 자격 증명이 만료되기 최소 5분 전에 새 자격 증명이 제공됩니다.

Warning

IAM 역할과 함께 인스턴스 메타데이터를 사용하는 서비스를 사용하는 경우 서비스에서 사용자 대신 HTTP 호출을 수행할 때 자격 증명이 노출되지 않도록 주의하십시오. 자격 증명이 노출될 수 있는 서비스 유형은 HTTP 프록시, HTML/CSS 검증 서비스, XML 포함을 지원하는 XML 프로세서 등입니다.

다음 명령은 IAM라는 s3access 역할의 보안 자격 증명을 검색합니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

다음은 예제 출력입니다.

```
{  
    "Code" : "Success",  
    "LastUpdated" : "2012-04-26T16:39:16Z",  
    "Type" : "AWS-HMAC",  
    "AccessKeyId" : "ASIAIOSFODNN7EXAMPLE",  
    "SecretAccessKey" : "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",  
}
```

```
{  
    "Token" : "token",  
    "Expiration" : "2017-05-17T15:09:54Z"  
}
```

인스턴스에서 실행되는 애플리케이션, AWS CLI 및 Windows PowerShell용 도구 명령의 경우, 임시 보안 자격 증명을 명시적으로 받지 않아도 됩니다. AWS SDK, AWS CLI, Windows PowerShell용 도구이 EC2 인스턴스 메타데이터 서비스에서 자동으로 자격 증명을 받아 사용하기 때문입니다. 임시 보안 자격 증명을 사용하여 인스턴스 외부로 호출하려면(예: IAM 정책 테스트) 액세스 키, 보안 키 및 세션 토큰을 제공해야 합니다. 자세한 내용은 IAM 사용 설명서의 [임시 보안 자격 증명을 사용해 AWS 리소스에 대한 액세스 요청하기](#)를 참조하십시오.

인스턴스 메타데이터에 대한 자세한 내용은 [인스턴스 메타데이터 및 사용자 데이터 \(p. 576\)](#) 단원을 참조하십시오. 인스턴스 메타데이터 IP 주소에 대한 자세한 내용은 [인스턴스 메타데이터 검색 \(p. 582\)](#) 단원을 참조하세요.

IAM 사용자에게 IAM 역할을 인스턴스에 전달할 수 있는 권한 부여

IAM 사용자가 IAM 역할로 인스턴스를 시작하거나 기존 인스턴스에 IAM 역할을 연결하거나 교체할 수 있도록 하려면 인스턴스에 역할을 전달할 권한을 부여해야 합니다.

다음 IAM 정책은 사용자에게 IAM 역할로 인스턴스(ec2:RunInstances)를 시작하거나 기존 인스턴스(ec2:AssociateIamInstanceProfile 및 ec2:ReplaceIamInstanceProfileAssociation)에 IAM 역할을 연결하거나 교체할 수 있는 권한을 부여합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:RunInstances",  
                "ec2:AssociateIamInstanceProfile",  
                "ec2:ReplaceIamInstanceProfileAssociation"  
            ],  
            "Resource": "*"  
        },  
        {  
            "Effect": "Allow",  
            "Action": "iam:PassRole",  
            "Resource": "*"  
        }  
    ]  
}
```

이 정책은 리소스를 "*"로 지정하여 IAM 사용자에게 모든 역할에 대한 액세스 권한을 부여합니다. 그러나 이 경우 해당 역할(기존 역할 또는 이후에 생성할 역할)로 인스턴스를 시작하는 사용자에게 불필요한 권한까지 과도하게 부여해도 무방할지를 고려해야 합니다.

IAM 역할 작업

IAM 역할을 만들고 시작 도중 또는 후에 인스턴스에 연결할 수 있습니다. 인스턴스에 대한 IAM 역할을 교체하거나 분리할 수도 있습니다.

목차

- [IAM 역할 생성 \(p. 893\)](#)
- [IAM 역할로 인스턴스 시작 \(p. 894\)](#)
- [IAM 역할을 인스턴스에 연결 \(p. 895\)](#)
- [IAM 역할 교체 \(p. 896\)](#)

- IAM 역할 분리 (p. 897)

IAM 역할 생성

특정 역할로 인스턴스를 시작하거나 인스턴스에 연결하려면 우선 IAM 역할을 생성해야 합니다.

IAM 콘솔을 사용하여 IAM 역할을 생성하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 역할, 역할 생성을 선택합니다.
3. 역할 유형 선택 페이지에서 EC2와 EC2 사용 사례를 선택합니다. 다음: 권한을 선택합니다.
4. 권한 정책 연결(Attach permissions policy) 페이지에서 인스턴스가 필요로 하는 리소스에 대한 액세스 권한을 부여하는 AWS 관리형 정책을 선택합니다.
5. 검토 페이지에서 역할의 이름을 입력하고 역할 생성을 선택합니다.

또는 AWS CLI를 사용하여 IAM 역할을 만들 수 있습니다. 다음 예제에서는 IAM 역할이 Amazon S3 버킷을 사용하도록 허용하는 정책을 사용하여 이 역할을 만듭니다.

IAM 역할과 인스턴스 프로파일을 생성하려면(AWS CLI)

1. 다음 트러스트 정책을 생성하고 `ec2-role-trust-policy.json`이라는 텍스트 파일로 저장합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Principal": { "Service": "ec2.amazonaws.com"},  
            "Action": "sts:AssumeRole"  
        }  
    ]  
}
```

2. `s3access` 역할을 만들고 `create-role` 명령을 사용하여 생성한 신뢰 정책을 지정합니다.

```
aws iam create-role --role-name s3access --assume-role-policy-document file://ec2-role-trust-policy.json  
{  
    "Role": {  
        "AssumeRolePolicyDocument": {  
            "Version": "2012-10-17",  
            "Statement": [  
                {  
                    "Action": "sts:AssumeRole",  
                    "Effect": "Allow",  
                    "Principal": {  
                        "Service": "ec2.amazonaws.com"  
                    }  
                }  
            ]  
        },  
        "RoleId": "AROAIIZKPBKS2LEXAMPLE",  
        "CreateDate": "2013-12-12T23:46:37.247Z",  
        "RoleName": "s3access",  
        "Path": "/",  
        "Arn": "arn:aws:iam::123456789012:role/s3access"  
    }  
}
```

3. 액세스 정책을 생성하고 `ec2-role-access-policy.json`이라는 텍스트 파일로 저장합니다. 예를 들어 이 정책은 인스턴스에서 실행되는 애플리케이션에 Amazon S3 관리 권한을 부여합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": ["s3:*"],  
            "Resource": ["*"]  
        }  
    ]  
}
```

4. `put-role-policy` 명령을 사용하여 액세스 정책을 역할에 연결합니다.

```
aws iam put-role-policy --role-name s3access --policy-name S3-Permissions --policy-document file://ec2-role-access-policy.json
```

5. `create-instance-profile` 명령을 사용하여 `s3access-profile`라는 인스턴스 프로파일을 만듭니다.

```
aws iam create-instance-profile --instance-profile-name s3access-profile  
{  
    "InstanceProfile": {  
        "InstanceProfileId": "AIPAJTLBPJLEGREXAMPLE",  
        "Roles": [],  
        "CreateDate": "2013-12-12T23:53:34.093Z",  
        "InstanceProfileName": "s3access-profile",  
        "Path": "/",  
        "Arn": "arn:aws:iam::123456789012:instance-profile/s3access-profile"  
    }  
}
```

6. `s3access` 인스턴스 프로파일에 `s3access-profile` 역할을 추가합니다.

```
aws iam add-role-to-instance-profile --instance-profile-name s3access-profile --role-name s3access
```

또는 다음 Windows PowerShell용 AWS 도구 명령을 사용합니다.

- [New-IAMRole](#)
- [Register-IAMRolePolicy](#)
- [New-IAMInstanceProfile](#)

IAM 역할로 인스턴스 시작

IAM 역할을 생성한 후 인스턴스를 시작하면서 해당 역할을 연결할 수 있습니다.

Important

IAM 역할을 생성한 후 권한이 전파되기까지 몇 초가 걸릴 수 있습니다. 특정 역할로 인스턴스를 처음 시작하려는 시도가 실패할 경우 몇 초간 기다린 후에 다시 시도해 보십시오. 자세한 내용은 IAM 사용 설명서에서 [역할 작업 문제 해결](#) 단원을 참조하십시오.

IAM 역할로 인스턴스를 시작 하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 대시보드에서 인스턴스 시작을 선택합니다.

3. AMI와 인스턴스 유형을 선택한 후 다음: 인스턴스 세부 정보 구성(Next: Configure Instance Details)을 선택합니다.
4. 인스턴스 세부 정보 구성(Configure Instance Details) 페이지의 IAM 역할에서 생성한 IAM 역할을 선택합니다.

Note

IAM 역할 생성 시 생성된 인스턴스 프로파일 이름이 IAM 역할 목록에 표시됩니다. 콘솔을 사용하여 IAM 역할을 생성한 경우 인스턴스 프로파일이 자동으로 생성되어 역할과 동일한 이름이 지정된 상태입니다. IAM, API 또는 AWS CLI SDK를 사용하여 AWS 역할을 생성한 경우 인스턴스 프로파일의 이름이 다를 수 있습니다.

5. 기타 세부 정보를 구성하고 지침에 따라 마법사의 나머지 절차를 완료하거나, 검토 후 시작을 선택하여 기본 설정을 수락하고 인스턴스 시작 검토 페이지로 바로 이동합니다.
6. 설정을 검토한 다음 시작을 선택하여 키 페어를 선택하고 인스턴스를 시작합니다.
7. 애플리케이션에서 Amazon EC2 API 작업을 사용하는 경우 인스턴스에 제공된 AWS 보안 자격 증명을 검색하고 이를 사용하여 요청에 서명합니다. AWS SDK는 이 작업을 자동으로 수행합니다.

IMDSv2

```
PS C:\> $token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

또는 AWS CLI를 사용하여 시작 중에 인스턴스와 역할을 연결할 수 있습니다. 명령에 인스턴스 프로파일을 지정해야 합니다.

IAM 역할로 인스턴스를 시작하려면(AWS CLI)

1. `run-instances` 명령을 사용하여 인스턴스 프로파일을 사용하는 인스턴스를 시작합니다. 다음 예제에서는 인스턴스 프로파일과 함께 인스턴스를 시작하는 방법을 보여 줍니다.

```
aws ec2 run-instances --image-id ami-11aa22bb --iam-instance-profile Name="s3access-profile" --key-name my-key-pair --security-groups my-security-group --subnet-id subnet-1a2b3c4d
```

또는 `New-EC2Instance` Windows PowerShell용 도구 명령을 사용합니다.

2. 애플리케이션에서 Amazon EC2 API 작업을 사용하는 경우 인스턴스에 제공된 AWS 보안 자격 증명을 검색하고 이를 사용하여 요청에 서명합니다. AWS SDK는 이 작업을 자동으로 수행합니다.

```
curl http://169.254.169.254/latest/meta-data/iam/security-credentials/role_name
```

IAM 역할을 인스턴스에 연결

IAM 역할을 역할이 없는 인스턴스에 연결하려면, 인스턴스가 `stopped` 또는 `running` 상태에 있어야 합니다.

IAM 역할을 인스턴스에 연결하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 나서 작업, 인스턴스 설정, IAM 역할 연결/바꾸기를 차례대로 선택합니다.
4. 인스턴스에 연결할 IAM 역할을 선택한 후 적용을 선택합니다.

IAM 역할을 인스턴스에 연결하려면(AWS CLI)

1. 필요한 경우 인스턴스를 설명하여 역할을 연결할 인스턴스의 ID를 가져옵니다.

```
aws ec2 describe-instances
```

2. [associate-iam-instance-profile](#) 명령을 사용하여 인스턴스 프로파일을 지정하여 인스턴스에 IAM 역할을 연결합니다. 인스턴스 프로파일의 Amazon 리소스 이름(ARN) 또는 이름을 사용할 수 있습니다.

```
aws ec2 associate-iam-instance-profile --instance-id i-1234567890abcdef0 --iam-instance-profile Name="TestRole-1"

{
    "IamInstanceProfileAssociation": {
        "InstanceId": "i-1234567890abcdef0",
        "State": "associating",
        "AssociationId": "iip-assoc-0dbd8529a48294120",
        "IamInstanceProfile": {
            "Id": "AIPAJLNLDX3AMYZNWWYAY",
            "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-1"
        }
    }
}
```

또는 다음 Windows PowerShell용 도구 명령을 사용합니다.

- [Get-EC2Instance](#)
- [Register-EC2IamInstanceProfile](#)

IAM 역할 교체

IAM 역할이 이미 연결된 인스턴스의 IAM 역할을 교체하려면, 인스턴스는 *running* 상태에 있어야 합니다. 기존 인스턴스를 먼저 분리하지 않고 인스턴스에 대한 IAM 역할을 변경하려는 경우 이 작업을 수행할 수 있습니다. 예를 들어 인스턴스에서 실행 중인 애플리케이션에서 수행된 API 작업이 중단되지 않도록 하기 위해 이 작업을 수행할 수 있습니다.

인스턴스에서 IAM 역할을 대체하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 나서 작업, 인스턴스 설정, IAM 역할 연결/바꾸기를 차례대로 선택합니다.
4. 인스턴스에 연결할 IAM 역할을 선택한 후 적용을 선택합니다.

인스턴스에서 IAM 역할을 대체하려면(AWS CLI)

1. 필요한 경우 IAM 인스턴스 프로파일 연결을 설명하여 교체할 IAM 인스턴스 프로파일의 연결 ID를 가져옵니다.

```
aws ec2 describe-iam-instance-profile-associations
```

2. [replace-iam-instance-profile-association](#) 명령을 사용하여 기존 인스턴스 프로파일에 대한 연결 ID와 교체할 인스턴스 프로파일의 ARN 또는 이름을 지정하여 IAM 인스턴스 프로파일을 교체합니다.

```
aws ec2 replace-iam-instance-profile-association --association-id iip-assoc-0044d817db6c0a4ba --iam-instance-profile Name="TestRole-2"

{
    "IamInstanceProfileAssociation": {
        "InstanceId": "i-087711ddaf98f9489",
        "State": "associating",
        "AssociationId": "iip-assoc-09654be48e33b91e0",
        "IamInstanceProfile": {
            "Id": "AIPAJCJEDKX7QYHWYK7GS",
            "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
        }
    }
}
```

또는 다음 Windows PowerShell용 도구 명령을 사용합니다.

- [Get-EC2IamInstanceProfileAssociation](#)
- [Set-EC2IamInstanceProfileAssociation](#)

IAM 역할 분리

실행 중이거나 중지된 인스턴스에서 IAM 역할을 분리할 수 있습니다.

인스턴스에서 IAM 역할 분리하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 나서 작업, 인스턴스 설정, IAM 역할 연결/바꾸기를 차례대로 선택합니다.
4. IAM 역할에서 역할 없음을 선택합니다. 적용을 선택합니다.
5. 확인 대화 상자에서 예, 분리를 선택합니다.

인스턴스에서 IAM 역할을 분리하려면(AWS CLI)

1. 필요한 경우 [describe-iam-instance-profile-associations](#)를 사용하여 IAM 인스턴스 프로파일 연결을 설명하고 분리할 IAM 인스턴스 프로파일의 연결 ID를 가져옵니다.

```
aws ec2 describe-iam-instance-profile-associations

{
    "IamInstanceProfileAssociations": [
        {
            "InstanceId": "i-088ce778fbfeb4361",
            "State": "associated",
            "AssociationId": "iip-assoc-0044d817db6c0a4ba",
            "IamInstanceProfile": {
                "Id": "AIPAJEDNCAA64SSD265D6",
                "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"
            }
        }
    ]
}
```

```
        ]  
    }
```

2. [disassociate-iam-instance-profile](#) 명령을 사용하여 연결 ID를 사용 중인 IAM 인스턴스 프로파일을 분리합니다.

```
aws ec2 disassociate-iam-instance-profile --association-id iip-assoc-0044d817db6c0a4ba  
{  
    "IamInstanceProfileAssociation": {  
        "InstanceId": "i-087711ddaf98f9489",  
        "State": "disassociating",  
        "AssociationId": "iip-assoc-0044d817db6c0a4ba",  
        "IamInstanceProfile": {  
            "Id": "AIPAJEDNCAA64SSD265D6",  
            "Arn": "arn:aws:iam::123456789012:instance-profile/TestRole-2"  
        }  
    }  
}
```

또는 다음 Windows PowerShell용 도구 명령을 사용합니다.

- [Get-EC2IamInstanceProfileAssociation](#)
- [Unregister-EC2IamInstanceProfile](#)

Windows 인스턴스의 인바운드 트래픽 권한 부여

보안 그룹을 통해 인스턴스로 들어오는 트래픽 및 인스턴스에 도달할 수 있는 트래픽의 유형을 제어할 수 있습니다. 예를 들어 험 네트워크에 속하는 컴퓨터만 RDP를 사용하여 인스턴스에 액세스하도록 허용할 수 있습니다. 인스턴스가 웹 서버인 경우 외부 사용자가 웹 서버의 콘텐츠를 탐색할 수 있도록 모든 IP 주소가 HTTP 또는 HTTPS를 사용하여 인스턴스에 액세스하도록 허용할 수 있습니다.

기본 보안 그룹과 새로 생성한 보안 그룹에는 인터넷에서 해당 인스턴스에 대한 액세스를 허용하지 않는 기본 규칙이 포함되어 있습니다. 자세한 내용은 [기본 보안 그룹 \(p. 911\)](#) 및 [사용자 지정 보안 그룹 \(p. 911\)](#) 단원을 참조하십시오. 인스턴스에 대한 네트워크 액세스를 가능하게 하려면 인스턴스의 인바운드 트래픽을 허용해야 합니다. 인바운드 트래픽을 위한 포트를 열려면 인스턴스 시작 시 연결한 보안 그룹에 규칙을 추가합니다.

인스턴스에 연결하려면 컴퓨터의 퍼블릭 IPv4 주소에서 비롯되는 RDP 트래픽을 인증하는 규칙을 설정해야 합니다. 추가 IP 주소 범위에서 비롯되는 RDP 트래픽을 허용하려면 인증할 각 범위에 대해 다른 규칙을 추가합니다.

IPv6용 VPC를 활성화하고 IPv6 주소로 인스턴스를 시작했다면, 퍼블릭 IPv4 주소 대신 IPv6 주소를 사용해 인스턴스에 연결할 수 있습니다. 로컬 컴퓨터에 IPv6 주소가 있고 IPv6를 사용하도록 컴퓨터를 구성해야 합니다.

Linux 인스턴스에 대한 네트워크 액세스를 가능하게 하려면 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [Linux 인스턴스의 인바운드 트래픽 권한 부여](#) 단원을 참조하십시오.

시작하기 전에

인스턴스에 액세스해야 하는 대상(예: 단일 호스트 또는 신뢰할 수 있는 특정 네트워크, 로컬 컴퓨터의 퍼블릭 IPv4 주소)을 결정합니다. Amazon EC2 콘솔의 보안 그룹 편집기는 로컬 컴퓨터의 퍼블릭 IPv4 주소를 자동으로 검색할 수 있습니다. 또는 인터넷 브라우저에서 "내 IP 주소"와 같은 검색 구문을 사용하거나 [Check IP](#) 서비스를 사용할 수도 있습니다. 고정 IP 주소 없이 ISP 또는 방화벽을 경유하여 연결하는 경우에는 클라우드 컴퓨터가 사용하는 IP 주소의 범위를 알아내야 합니다.

Warning

0.0.0.0/0을 사용하면 모든 IPv4 주소에서 RDP를 사용하여 인스턴스에 액세스할 수 있습니다.
::/0을 사용하면 모든 IPv6 주소를 사용해 인스턴스에 액세스할 수 있습니다. 테스트 환경에서 잠시 사용하는 것은 괜찮지만 프로덕션 환경에서는 안전하지 않습니다. 프로덕션에서는 특정 IP 주소나 주소 범위만 인스턴스에 액세스하도록 허용하십시오.

Windows 방화벽은 수신 트래픽도 차단할 수 있습니다. 인스턴스에 대한 액세스를 설정하는 데 문제가 있는 경우 Windows 방화벽을 비활성화할 수 있습니다. 자세한 내용은 [원격 데스크톱으로 원격 컴퓨터에 연결할 수 없음 \(p. 1181\)](#) 단원을 참조하십시오.

Windows 인스턴스의 인바운드 RDP 트래픽에 대한 규칙 추가

보안 그룹은 연결된 인스턴스에 대한 방화벽 역할을 하여 인스턴스 수준에서 인바운드 트래픽과 아웃바운드 트래픽을 모두 제어합니다. Windows를 사용하여 IP 주소에서 RDP 인스턴스에 연결할 수 있게 하는 규칙을 보안 그룹에 추가해야 합니다.

보안 그룹에 IPv4를 통한 인바운드 RDP 트래픽에 대한 규칙을 추가하려면(콘솔)

1. Amazon EC2 콘솔의 탐색 창에서 인스턴스를 선택합니다. 인스턴스를 선택하고 설명 탭을 확인합니다. 인스턴스에 연결된 보안 그룹이 보안 그룹에 나열됩니다. 인바운드 규칙 보기 를 선택하여 인스턴스에 적용되고 있는 규칙 목록을 표시합니다.
2. 탐색 창에서 보안 그룹을 선택합니다. 인스턴스에 연결된 보안 그룹 중 하나를 선택합니다.
3. 세부 정보 창의 인바운드 탭에서 편집을 선택합니다. 대화 상자에서 규칙 추가를 선택하고 유형 목록에서 RDP를 선택합니다.
4. 필드를 로컬 컴퓨터의 퍼블릭 IPv4 주소로 자동으로 채우려면 소스 필드에서 내 IP를 선택하면 됩니다. 또는 사용자 지정을 선택하고 컴퓨터 또는 네트워크의 퍼블릭 IPv4 주소를 CIDR 표기법으로 지정해도 됩니다. 예를 들어, IPv4 주소가 203.0.113.25인 경우 이 단일 IPv4 주소를 CIDR 표기법으로 나열하려면 203.0.113.25/32를 지정합니다. 회사에서 주소를 범위로 할당하는 경우 전체 범위(예: 203.0.113.0/24)를 지정합니다.

IP 주소 확인에 대한 자세한 내용은 [시작하기 전에 \(p. 898\)](#) 단원을 참조하십시오.

5. 저장을 선택합니다.

IPv6 주소로 인스턴스를 시작했는데 IPv6 주소를 사용해 인스턴스에 연결하려면 RDP를 통한 인바운드 IPv6 트래픽을 허용하는 규칙을 추가해야 합니다.

보안 그룹에 IPv6를 통한 인바운드 RDP 트래픽에 대한 규칙을 추가하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다. 인스턴스에 대한 보안 그룹을 선택합니다.
3. 인바운드, 편집, 규칙 추가 를 선택합니다.
4. 유형의 경우 RDP를 선택합니다.
5. 소스 필드에 컴퓨터의 IPv6 주소를 CIDR 표기법으로 지정합니다. 예를 들어, IPv6 주소가 2001:db8:1234:1a00:9691:9503:25ad:1761인 경우 단일 IP 주소를 CIDR 표기법으로 나열하려면 2001:db8:1234:1a00:9691:9503:25ad:1761/128을 지정합니다. 회사에서 주소를 범위로 할당하는 경우 전체 범위(예: 2001:db8:1234:1a00::/64)를 지정합니다.
6. 저장을 선택합니다.

Note

인스턴스 자체가 아닌 로컬 시스템에서 다음 명령을 실행해야 합니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

명령줄을 사용하여 보안 그룹에 규칙을 추가하려면 다음을 수행합니다.

1. 다음 명령 중 하나를 사용하여 인스턴스와 관련된 보안 그룹을 찾습니다.

- [describe-instance-attribute\(AWS CLI\)](#)

```
aws ec2 describe-instance-attribute --instance-id instance_id --attribute groupSet
```

- [Get-EC2InstanceAttribute\(Windows PowerShell용 AWS 도구\)](#)

```
PS C:\> (Get-EC2InstanceAttribute -InstanceId instance_id -Attribute groupSet).Groups
```

두 명령 모두 다음 단계에서 사용하는 보안 그룹 ID를 반환합니다.

2. 다음 명령 중 하나를 사용하여 보안 그룹에 규칙을 추가하십시오.

- [authorize-security-group-ingress\(AWS CLI\)](#)

```
aws ec2 authorize-security-group-ingress --group-id security_group_id --protocol tcp  
--port 3389 --cidr cidr_ip_range
```

- [Grant-EC2SecurityGroupIngress\(Windows PowerShell용 AWS 도구\)](#)

Grant-EC2SecurityGroupIngress 명령에는 보안 그룹 규칙에 사용할 프로토콜, 포트 범위 및 IP 주소 범위를 설명하는 IpPermission 파라미터가 필요합니다. 다음 명령은 IpPermission 파라미터를 생성합니다.

```
PS C:\> $ip1 = @{ IpProtocol="tcp"; FromPort="3389"; ToPort="3389";  
IpRanges="cidr_ip_range" }
```

```
PS C:\> Grant-EC2SecurityGroupIngress -GroupId security_group_id -IpPermission  
@($ip1)
```

인스턴스에 보안 그룹 할당

인스턴스를 시작할 때 인스턴스에 보안 그룹을 할당할 수 있습니다. 규칙을 추가하거나 제거하면 해당 보안 그룹을 할당한 모든 인스턴스에 변경 내용이 자동으로 적용됩니다.

인스턴스를 시작한 이후에는 해당 보안 그룹을 변경할 수 없습니다. 자세한 내용은 Amazon VPC 사용 설명서의 [인스턴스의 보안 그룹 변경](#) 단원을 참조하십시오.

Amazon EC2 키 페어 및 Windows 인스턴스

프라이빗 키와 퍼블릭 키로 구성된 키 페어는 인스턴스에 연결할 때 자격 증명 입증에 사용하는 보안 자격 증명 집합입니다. Amazon EC2는 퍼블릭 키를 저장하고 사용자는 프라이빗 키를 저장합니다. 프라이빗 키를 사용하여 인스턴스에 안전하게 액세스할 수 있습니다. 프라이빗 키를 소유하는 사람은 누구나 인스턴스에 연결할 수 있으므로 보안된 위치에 프라이빗 키를 저장해 두는 것이 중요합니다.

인스턴스를 시작할 때 키 페어를 입력하라는 메시지가 표시됩니다 (p. 381). RDP를 사용하여 인스턴스에 연결하려는 경우 키 페어를 지정해야 합니다. 기존 키 페어를 선택하거나 새로 만들 수 있습니다. Windows 인스턴스에서는 프라이빗 키를 사용하여 관리자 암호를 가져오고 RDP를 사용하여 로그인합니다. 인스턴스 연결에 대한 자세한 내용은 [Windows 인스턴스에 연결](#) (p. 436) 주제를 참조하십시오. 키 페어 및 Linux 인스턴스에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2 키 페어 및 Linux 인스턴스](#)를 참조하십시오.

Amazon EC2에는 프라이빗 키의 사본이 보관되지 않으므로, 프라이빗 키를 분실하면 이를 복구할 방법이 전혀 없습니다. 그러나 분실된 프라이빗 키를 사용하는 인스턴스에 연결하는 방법은 여전히 있을 수 있습니다. 자세한 내용은 [프라이빗 키를 분실한 경우 Windows 인스턴스에 연결하는 방법 \(p. 907\)](#) 단원을 참조하십시오.

Amazon EC2에서 사용되는 키는 2048비트 SSH-2 RSA 키입니다. 리전당 최대 5,000개의 키 페어를 가질 수 있습니다.

목차

- [키 페어 생성 또는 가져오기 \(p. 901\)](#)
- [키 페어 태그 지정 \(p. 904\)](#)
- [키 페어에 맞는 퍼블릭 키 검색 \(p. 905\)](#)
- [인스턴스 메타데이터를 통해 키 페어의 퍼블릭 키 검색 \(p. 905\)](#)
- [시작 시 지정된 키 페어 식별 \(p. 906\)](#)
- [\(선택 사항\) 키 페어의 지문 확인 \(p. 906\)](#)
- [프라이빗 키를 분실한 경우 Windows 인스턴스에 연결하는 방법 \(p. 907\)](#)
- [키 페어 삭제 \(p. 907\)](#)

키 페어 생성 또는 가져오기

Amazon EC2를 사용하여 새 키 페어를 생성하거나 기존 키 페어를 가져올 수 있습니다.

옵션

- [옵션 1: Amazon EC2를 사용하여 키 페어 생성 \(p. 901\)](#)
- [옵션 2: Amazon EC2로 사용자의 퍼블릭 키 가져오기 \(p. 902\)](#)

옵션 1: Amazon EC2를 사용하여 키 페어 생성

다음 방법 중 하나를 사용하여 키 페어를 만들 수 있습니다.

새로운 콘솔

키 페어를 만들려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 NETWORK & SECURITY에서 Key Pairs를 선택합니다.
3. Create key pair(키 페어 생성)를 선택합니다.
4. 이름에 설명이 포함된 키 페어 이름을 입력합니다. Amazon EC2는 키 이름으로 지정한 이름과 퍼블릭 키를 연결합니다. 키 이름에는 최대 255자이며 ASCII 문자를 포함할 수 있습니다. 선행 또는 후행 공백을 포함할 수 없습니다.
5. 파일 형식에서 프라이빗 키를 저장할 형식을 선택합니다. OpenSSH에서 사용할 수 있는 형식으로 프라이빗 키를 저장하려면 pem을 선택합니다. PuTTY에서 사용할 수 있는 형식으로 프라이빗 키를 저장하려면 ppk를 선택합니다.
6. Create key pair(키 페어 생성)를 선택합니다.
7. 브라우저에서 프라이빗 키 파일이 자동으로 다운로드됩니다. 기본 파일 이름은 키 페어의 이름으로 지정한 이름이며, 파일 이름 확장명은 선택한 파일 형식에 따라 결정됩니다. 안전한 장소에 프라이빗 키 파일을 저장합니다.

Important

이때가 사용자가 프라이빗 키 파일을 저장할 수 있는 유일한 기회입니다.

기존 콘솔

키 페어를 만들려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 NETWORK & SECURITY에서 Key Pairs를 선택합니다.
3. Create Key Pair를 선택합니다.
4. 키 페어 이름에서 키 페어에 대한 설명이 포함된 이름을 입력한 다음 생성을 선택합니다. 키 이름에는 최대 255자의 ASCII 문자를 포함할 수 있습니다. 선행 또는 후행 공백을 포함할 수 없습니다.
5. 브라우저에서 프라이빗 키 파일이 자동으로 다운로드됩니다. 기본 파일 이름은 키 페어의 이름으로 지정된 이름이며, 파일 이름 확장명은 .pem입니다. 안전한 장소에 프라이빗 키 파일을 저장합니다.

Important

이때가 사용자가 프라이빗 키 파일을 저장할 수 있는 유일한 기회입니다.

AWS CLI

키 페어를 만들려면

- 다음과 같이 `create-key-pair` AWS CLI 명령을 사용하여 키를 생성하고 .pem 파일에 저장합니다.

```
aws ec2 create-key-pair --key-name my-key-pair --query 'KeyMaterial' --output text  
> my-key-pair.pem
```

PowerShell

키 페어를 만들려면

다음과 같이 `New-EC2KeyPair` Windows PowerShell용 AWS 도구 명령을 사용하여 키를 생성하고 .pem 파일에 저장합니다.

```
PS C:\> (New-EC2KeyPair -KeyName "my-key-pair").KeyMaterial | Out-File -Encoding ascii  
-FilePath C:\path\my-key-pair.pem
```

옵션 2: Amazon EC2로 사용자의 퍼블릭 키 가져오기

Amazon EC2를 사용하지 않고 키 페어를 만들었다면 타사 도구를 사용하여 RSA 키 페어를 만든 후 Amazon EC2로 퍼블릭 키를 가져올 수 있습니다. 예를 들면 ssh-keygen(표준 OpenSSH 설치 시 제공되는 도구)을 사용하여 키 페어를 만들 수 있습니다. 또는 Java, Ruby, Python 등 각종 프로그래밍 언어에서 제공하는 표준 라이브러리를 사용하여 RSA 키 페어를 만들어도 됩니다.

요구 사항

- 지원되는 형식은 다음과 같습니다.
 - OpenSSH 퍼블릭 키 형식
 - Base64 인코딩 DER 형식
 - [RFC4716](#)에 지정된 SSH 퍼블릭 키 파일 형식
 - SSH 프라이빗 키 파일 형식은 PEM이어야 합니다(예를 들어 ssh-keygen -m PEM을 사용하여 OpenSSH 키를 PEM 형식으로 변환).
- RSA 키를 생성하십시오. Amazon EC2에서 DSA 키는 허용되지 않습니다.
- 지원되는 길이는 1024, 2048, 4096입니다.

타사 도구를 이용한 키 페어 만들기

1. 타사 도구로 원하는 키 페어를 생성합니다.
2. 퍼블릭 키는 로컬 파일에 저장합니다. 예: C:\keys\my-key-pair.pub. 이 파일의 파일 이름 확장자는 중요하지 않습니다.
3. 프라이빗 키를 확장자가 .pem인 다른 로컬 파일에 저장합니다. 예: C:\keys\my-key-pair.pem. 프라이빗 키 파일은 안전한 장소에 저장합니다. 인스턴스를 시작할 때 키 페어의 이름을 제공하고, 인스턴스에 연결할 때마다 해당 프라이빗 키를 제공해야 합니다.

키 페어를 만든 후 다음 방법 중 하나를 사용하여 키 페어를 Amazon EC2로 가져옵니다.

새로운 콘솔

퍼블릭 키 가져오기

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Key Pairs]를 선택합니다.
3. 키 페어 가져오기를 선택합니다.
4. 이름에 키 페어를 설명하는 이름을 입력합니다. 이름에는 최대 255자의 ASCII 문자를 포함할 수 있습니다. 선행 또는 후행 공백을 포함할 수 없습니다.
5. 찾아보기를 선택하여 퍼블릭 키를 탐색하고 선택하거나 퍼블릭 키의 내용을 퍼블릭 키 내용 필드에 붙여 넣습니다.
6. 키 페어 가져오기를 선택합니다.
7. 가져온 키 페어가 키 페어 목록에 나타나는지 확인합니다.

기존 콘솔

퍼블릭 키 가져오기

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 NETWORK & SECURITY에서 Key Pairs를 선택합니다.
3. 키 페어 가져오기를 선택합니다.
4. 키 페어 가져오기 대화 상자에서 찾아보기를 선택하고 이전에 저장한 퍼블릭 키 파일을 선택합니다. 키 페어 이름 필드에 키 페어의 이름을 입력하고 가져오기를 선택합니다. 이름에는 최대 255자의 ASCII 문자를 포함할 수 있습니다. 선행 또는 후행 공백을 포함할 수 없습니다.
5. 가져온 키 페어가 키 페어 목록에 나타나는지 확인합니다.

AWS CLI

퍼블릭 키 가져오기

`import-key-pair` AWS CLI 명령을 사용합니다.

키 페어를 성공적으로 가져왔는지 확인하려면

`describe-key-pairs` AWS CLI 명령을 사용합니다.

PowerShell

퍼블릭 키 가져오기

`Import-EC2KeyPair` Windows PowerShell용 AWS 도구 명령을 사용합니다.

키 페어를 성공적으로 가져왔는지 확인하려면

Get-EC2KeyPair Windows PowerShell용 AWS 도구 명령을 사용합니다.

키 페어 태그 지정

기존 키 페어를 분류하고 관리할 수 있도록 사용자 지정 메타데이터로 태그를 지정할 수 있습니다. 태그 작동 방식에 대한 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

새 콘솔과 명령줄 도구를 사용하여 태그를 보고, 추가하고, 삭제할 수 있습니다.

새로운 콘솔

기존 키 페어의 태그를 보거나 추가 또는 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Key Pairs]를 선택합니다.
3. 키 페어를 선택한 다음 작업, 태그 관리를 선택합니다.
4. 태그 관리 섹션에는 키 페어에 할당된 모든 태그가 표시됩니다.
 - 태그를 추가하려면 태그 추가를 선택한 다음 태그 키와 값을 입력합니다. 페어당 최대 50개의 태그를 추가할 수 있습니다. 자세한 내용은 [태그 제한 \(p. 1147\)](#) 단원을 참조하십시오.
 - 태그를 삭제하려면 삭제할 태그 옆에 있는 제거를 선택합니다.
5. [Save changes]를 선택합니다.

AWS CLI

키 페어 태그를 보려면

[describe-tags](#) AWS CLI 명령을 사용합니다. 다음 예제에서는 모든 키 페어의 태그를 설명합니다.

```
C:\> aws ec2 describe-tags --filters "Name=resource-type,Values=key-pair"
```

```
{  
    "Tags": [  
        {  
            "Key": "Environment",  
            "ResourceId": "key-0123456789EXAMPLE",  
            "ResourceType": "key-pair",  
            "Value": "Production"  
        },  
        {  
            "Key": "Environment",  
            "ResourceId": "key-9876543210EXAMPLE",  
            "ResourceType": "key-pair",  
            "Value": "Production"  
        }]  
}
```

특정 키 페어의 태그를 설명하려면

[describe-key-pairs](#) AWS CLI 명령을 사용합니다.

```
C:\> aws ec2 describe-key-pairs --key-pair-ids key-0123456789EXAMPLE
```

```
{
```

```
"KeyPairs": [
{
    "KeyName": "MyKeyPair",
    "KeyFingerprint":
"1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
    "KeyId": "key-0123456789EXAMPLE",
    "Tags": [
{
        "Key": "Environment",
        "Value": "Production"
    }
]
}
```

기존 키 페어에 태그를 지정하려면

[create-tags](#) AWS CLI 명령을 사용합니다. 다음 예제에서는 기존 키 쌍에 Key=Cost-Center 및 Value=CC-123 태그가 지정됩니다.

```
C:\> aws ec2 create-tags --resources key-0123456789EXAMPLE --tags Key=Cost-Center,Value=CC-123
```

키 쌍에서 태그를 삭제하려면

[delete-tags](#) AWS CLI 명령을 사용합니다. 예제를 보려면 AWS CLI Command Reference의 [예제](#)를 참조하십시오.

PowerShell

키 페어 태그를 보려면

[Get-EC2Tag](#) 명령을 사용합니다.

특정 키 페어의 태그를 설명하려면

[Get-EC2KeyPair](#) 명령을 사용합니다.

기존 키 페어에 태그를 지정하려면

[New-EC2Tag](#) 명령을 사용합니다.

키 쌍에서 태그를 삭제하려면

[Remove-EC2Tag](#) 명령을 사용합니다.

키 페어에 맞는 퍼블릭 키 검색

로컬 Windows 컴퓨터에서 PuTTYgen을 사용하여 키 페어의 퍼블릭 키를 가져올 수 있습니다.

PuTTYgen을 시작하고 로드를 선택합니다. .ppk 및 .pem 파일을 선택합니다. PuTTYgen의 OpenSSH authorized_keys 파일에 넣기 위한 퍼블릭 키 아래에 퍼블릭 키가 표시됩니다. 또한 퍼블릭 키 저장을 선택하고, 파일 이름을 지정한 다음, 파일을 저장하고, 파일을 열어서 퍼블릭 키를 볼 수도 있습니다.

인스턴스 메타데이터를 통해 키 페어의 퍼블릭 키 검색

또한, 인스턴스를 시작할 때 지정한 퍼블릭 키는 해당 인스턴스 메타데이터를 통해 확인할 수 있습니다. 인스턴스를 시작할 때 지정한 퍼블릭 키를 보려면 인스턴스에서 다음 명령을 사용하면 됩니다.

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

다음은 예제 출력입니다.

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS706Vhz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzoOWbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJOI0iBXrlsLnBItnckij7FbtxJMXLvvwJryDUilBMTjYtwB+QhYXUMOzce5Pjz5/i8SeJtjnV3iAoG/cQk+OFzzqaeJAAHco+CY+5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3RBQoQzd8v7yeb7Oz1PnWOyN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE my-key-pair
```

인스턴스에 연결하는 데 사용하는 키 페어를 변경하면 새 퍼블릭 키가 표시되도록 인스턴스 메타데이터가 업데이트되지 않습니다. 대신 인스턴스 메타데이터는 인스턴스를 시작할 때 지정한 키 페어의 퍼블릭 키를 계속 표시합니다. 자세한 내용은 [인스턴스 메타데이터 검색 \(p. 582\)](#) 단원을 참조하십시오.

시작 시 지정된 키 페어 식별

인스턴스를 시작할 때 키 페어를 입력하라는 메시지가 표시됩니다 (p. 381). RDP를 사용하여 인스턴스에 연결하려는 경우 키 페어를 지정해야 합니다.

시작 시 지정된 키 페어를 식별하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택한 다음 인스턴스를 선택합니다.
3. 설명 탭의 키 페어 이름 필드에는 인스턴스를 시작할 때 지정한 키 페어의 이름이 표시됩니다. 인스턴스의 퍼블릭 키를 변경하거나 키 페어를 추가하더라도 키 페어 이름의 값은 변경되지 않습니다.

(선택 사항) 키 페어의 지문 확인

Amazon EC2 콘솔에서 키 페어 페이지를 보면 지문 열에 키 페어에서 생성된 지문이 표시됩니다. AWS에 의한 지문 계산 값은 키 페어가 AWS와 타사 도구 중 어디서 생성되었는지에 따라 달라집니다. AWS를 사용하여 키 페어를 만든 경우 지문은 SHA-1 해시 함수를 통해 산출됩니다. 타사 도구로 키 페어를 만들고 AWS에 퍼블릭 키를 업로드한 경우이거나 기존 AWS에서 만든 프라이빗 키에서 새 퍼블릭 키를 생성하여 AWS에 업로드한 경우, 지문은 MD5 해시 함수를 통해 산출됩니다.

키 페어 페이지에 표시된 SSH2 지문을 사용하여 로컬 컴퓨터에 있는 프라이빗 키가 AWS에 저장된 퍼블릭 키와 일치하는지 확인할 수 있습니다. 프라이빗 키 파일을 다운로드한 컴퓨터에서 프라이빗 키 파일에서 SSH2 지문을 생성합니다. 이 경우 출력이 콘솔에 표시된 지문과 일치해야 합니다.

AWS를 사용하여 키 페어를 생성한 경우 다음 예제와 같이 OpenSSL 도구를 사용하여 지문을 생성할 수 있습니다.

```
C:\> openssl pkcs8 -in path_to_private_key -inform PEM -outform DER -topk8 -nocrypt | openssl sha1 -c
```

타사 도구를 사용하여 키 페어를 생성하고 AWS에 퍼블릭 키를 업로드한 경우, OpenSSL 도구를 사용하여 다음 예제와 같이 지문을 생성할 수 있습니다.

```
C:\> openssl rsa -in path_to_private_key -pubout -outform DER | openssl md5 -c
```

OpenSSH 7.8 이상을 사용하여 OpenSSH 키 페어를 생성하고 AWS에 퍼블릭 키를 업로드한 경우 ssh-keygen을 사용하여 다음 예제와 같이 지문을 생성할 수 있습니다.

```
C:\> ssh-keygen -ef path_to_private_key -m PEM | openssl rsa -RSAPublicKey_in -outform DER  
| openssl md5 -c
```

프라이빗 키를 분실한 경우 Windows 인스턴스에 연결하는 방법

새로 시작된 Windows 인스턴스에 연결하는 경우 인스턴스를 시작할 때 지정한 키 페어의 프라이빗 키를 사용하여 관리자 계정에 대한 암호를 해독합니다.

관리자 암호를 분실하여 더 이상 프라이빗 키를 가지고 있지 않은 경우 암호를 재설정하거나 새 인스턴스를 생성해야 합니다. 자세한 내용은 [기억나지 않거나 만료된 Windows 관리자 암호 재설정 \(p. 1196\)](#) 단원을 참조하십시오. AWS Systems Manager 문서를 사용하여 비밀번호를 재설정하는 단계는 AWS 시스템 관리자 사용 설명서의 [Amazon EC2 인스턴스에서 암호 및 SSH 키 재설정](#)을 참조하십시오.

키 페어 삭제

키 페어 삭제는 곧 Amazon EC2의 퍼블릭 키 사본만 삭제하는 것을 의미합니다. 키 페어를 삭제하더라도 컴퓨터에 있는 프라이빗 키나 해당 키 페어를 사용하여 이전에 시작한 임의의 인스턴스에 대한 퍼블릭 키에는 영향이 미치지 않습니다. 삭제된 키 페어를 사용하여 새 인스턴스를 시작할 수는 없지만, 프라이빗 키(.pem) 파일을 계속 보유하고 있다면 삭제된 키 페어를 사용하여 시작한 임의의 인스턴스에 계속해서 연결할 수 있습니다.

Auto Scaling 그룹을 사용 중인 경우(예: Elastic Beanstalk 환경에서 사용), 삭제하려는 키 페어가 시작 구성에서 지정되지 않았는지 확인하십시오. 비정상 인스턴스가 발견될 경우 Amazon EC2 Auto Scaling에서 대체 인스턴스를 시작하지만, 키 페어를 찾을 수 없으면 인스턴스 시작에 실패합니다.

다음 방법 중 하나를 사용하여 키 페어를 삭제할 수 있습니다.

새로운 콘솔

키 페어를 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Key Pairs]를 선택합니다.
3. 삭제할 키 페어를 선택하고 삭제를 선택합니다.
4. 확인 필드에 **Delete**를 입력한 다음 삭제를 선택합니다.

기존 콘솔

키 페어를 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 NETWORK & SECURITY에서 Key Pairs를 선택합니다.
3. 키 페어를 선택하고 삭제를 선택합니다.
4. 메시지가 나타나면 예를 선택합니다.

AWS CLI

키 페어를 삭제하려면

delete-key-pair AWS CLI 명령을 사용합니다.

PowerShell

키 페어를 삭제하려면

[Remove-EC2KeyPair](#) Windows PowerShell용 AWS 도구 명령을 사용합니다.

Windows 인스턴스에 대한 Amazon EC2 보안 그룹

보안 그룹은 EC2 인스턴스에 대한 수신 및 발신 트래픽을 제어하는 가상 방화벽 역할을 합니다. 인바운드 규칙은 인스턴스로 들어오는 트래픽을 제어하고 아웃바운드 규칙은 인스턴스에서 나가는 트래픽을 제어합니다. 인스턴스를 시작할 때 하나 이상의 보안 그룹을 지정할 수 있습니다. 보안 그룹을 지정하지 않을 경우 Amazon EC2에서 기본 보안 그룹이 사용됩니다. 연결된 인스턴스에서 트래픽을 주고 받을 수 있도록 하는 규칙을 각 보안 그룹에 추가할 수 있습니다. 언제든지 보안 그룹에 대한 규칙을 수정할 수 있습니다. 새 규칙 및 수정된 규칙은 보안 그룹에 연결된 모든 인스턴스에 자동으로 적용됩니다. Amazon EC2는 트래픽이 인스턴스에 도달하도록 허용할지 여부를 결정할 때 인스턴스와 연결된 모든 보안 그룹에서 모든 규칙을 평가합니다.

VPC에서 인스턴스를 시작할 때 해당 VPC에 대해 생성된 보안 그룹을 지정해야 합니다. 인스턴스를 시작한 이후에는 해당 보안 그룹을 변경할 수 없습니다. 보안 그룹은 네트워크 인터페이스와 연결됩니다. 인스턴스의 보안 그룹을 변경하면 기본 네트워크 인터페이스(eth0)와 연결된 보안 그룹이 변경됩니다. 자세한 내용은 Amazon VPC 사용 설명서의 [인스턴스의 보안 그룹 변경](#) 단원을 참조하십시오. 다른 네트워크 인터페이스와 연결된 보안 그룹을 변경할 수도 있습니다. 자세한 내용은 [보안 그룹 변경 \(p. 746\)](#) 단원을 참조하십시오.

보안 그룹으로 완전히 충족되지 않는 요구 사항이 있는 경우 보안 그룹을 사용하면서 인스턴스에 대한 자체 방화벽을 유지합니다.

Linux 인스턴스에 대한 트래픽을 허용하려는 경우 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux 인스턴스에 대한 Amazon EC2 보안 그룹](#)을 참조하십시오.

목차

- [보안 그룹 규칙 \(p. 909\)](#)
 - [연결 추적 \(p. 910\)](#)
- [기본 보안 그룹 \(p. 911\)](#)
- [사용자 지정 보안 그룹 \(p. 911\)](#)
- [보안 그룹 작업 \(p. 912\)](#)
 - [보안 그룹 생성 \(p. 912\)](#)
 - [보안 그룹 복사 \(p. 913\)](#)
 - [보안 그룹 보기 \(p. 913\)](#)
 - [보안 그룹에 규칙 추가 \(p. 914\)](#)
 - [보안 그룹 규칙 업데이트 \(p. 917\)](#)
 - [보안 그룹에서 규칙 삭제 \(p. 918\)](#)
 - [보안 그룹 삭제 \(p. 918\)](#)
- [보안 그룹 규칙 참조 \(p. 919\)](#)
 - [웹 서버 규칙 \(p. 919\)](#)
 - [데이터베이스 서버 규칙 \(p. 920\)](#)
 - [컴퓨터에서 인스턴스 연결에 대한 규칙 \(p. 921\)](#)
 - [보안 그룹이 동일한 인스턴스에서 인스턴스에 대한 연결 규칙 \(p. 922\)](#)
 - [Ping/ICMP 규칙 \(p. 922\)](#)
 - [DNS 서버 규칙 \(p. 922\)](#)
 - [Amazon EFS 규칙 \(p. 923\)](#)
 - [Elastic Load Balancing 규칙 \(p. 923\)](#)

- VPC 피어링 규칙 (p. 924)

보안 그룹 규칙

보안 그룹의 규칙은 보안 그룹과 연결된 인스턴스에 도달할 수 있는 인바운드 트래픽과 인스턴스에서 나갈 수 있는 아웃바운드 트래픽을 제어합니다.

다음은 보안 그룹 규칙의 특징입니다.

- 기본적으로 보안 그룹은 모든 아웃바운드 트래픽을 허용합니다.
- 보안 그룹 규칙은 항상 허용적입니다. 따라서 액세스를 거부하는 규칙을 생성할 수 없습니다.
- 보안 그룹은 상태가 저장됩니다. 사용자가 인스턴스에서 요청을 전송하면 해당 요청의 응답 트래픽은 인바운드 보안 그룹 규칙에 관계없이 인바운드 흐름이 허용됩니다. VPC 보안 그룹의 경우, 허용된 인바운드 트래픽에 대한 응답은 아웃바운드 규칙에 관계없이 아웃바운드 흐름이 허용됩니다. 자세한 내용은 [연결 추적 \(p. 910\)](#) 단원을 참조하십시오.
- 언제든지 규칙을 추가하고 제거할 수 있습니다. 변경 내용은 보안 그룹과 연결된 인스턴스에 자동으로 적용됩니다.

일부 규칙 변경 사항이 미치는 효과는 트래픽의 추적 방법에 따라 다를 수 있습니다. 자세한 내용은 [연결 추적 \(p. 910\)](#) 단원을 참조하십시오.

- 여러 보안 그룹을 한 인스턴스와 연결할 경우 각 보안 그룹의 규칙이 유효하게 결합된 단일 규칙 세트가 생성됩니다. Amazon EC2는 이 규칙 세트를 사용하여 액세스를 허용할지 여부를 결정합니다.

인스턴스에 여러 보안 그룹을 할당할 수 있습니다. 따라서 한 인스턴스에 수백 개의 규칙이 적용될 수 있습니다. 이로 인해 인스턴스에 액세스할 때 문제가 발생할 수 있습니다. 규칙을 최대한 간략하게 만드는 것이 좋습니다.

각 규칙에 대해 다음을 지정합니다.

- 이름: 보안 그룹의 이름입니다(예: my-security-group).

이름의 최대 길이는 255자입니다. 허용되는 문자는 a-z, A-Z, 0-9, 공백 및 _-:/()#,@[]+=;{}!\$*. 이름에 후행 공백이 포함되어 있으면 이름을 저장할 때 공백을 자릅니다. 예를 들어 이름에 “테스트 보안 그룹”을 입력하면 “테스트 보안 그룹”으로 저장됩니다.
- 프로토콜: 허용할 프로토콜. 가장 일반적인 프로토콜은 6(TCP), 17(UDP) 및 1(ICMP)입니다.
- 포트 범위: TCP, UDP 또는 사용자 지정 프로토콜의 경우 허용할 포트의 범위. 단일 포트 번호(예: 22) 또는 포트 번호의 범위(예: 7000-8000)를 지정할 수 있습니다.
- ICMP 유형 및 코드: ICMP의 경우, ICMP 유형과 코드.
- 소스 또는 대상: 트래픽에 대한 소스(인바운드 규칙) 또는 대상(아웃바운드 규칙). 다음 옵션 중 하나를 지정합니다.
 - 개별 IPv4 주소. /32 접두사 길이를 사용해야 합니다(예: 203.0.113.1/32).
 - 개별 IPv6 주소. /128 접두사 길이를 사용해야 합니다(예: 2001:db8:1234:1a00::123/128).
 - CIDR 블록 표기법으로 표시된 IPv4 주소의 범위(예: 203.0.113.0/24).
 - CIDR 블록 표기법으로 표시된 IPv6 주소의 범위(예: 2001:db8:1234:1a00::/64).
 - 접두사 목록 ID(예: p1-1234abc1234abc123). 자세한 내용은 Amazon VPC 사용 설명서의 [접두사 목록](#)을 참조하십시오.
- 다른 보안 그룹. 이 옵션을 사용하면 지정된 보안 그룹과 연결된 인스턴스가 이 보안 그룹과 연결된 인스턴스에 액세스할 수 있습니다. 하지만 이 보안 그룹에 소스 보안 그룹의 규칙이 추가되지는 않습니다. 다음 보안 그룹 중 하나를 지정할 수 있습니다.
 - 현재 보안 그룹
 - 동일한 VPC에 대한 다른 보안 그룹

- VPC 피어링 연결에서 동일한 VPC 또는 피어 VPC에 대한 다른 보안 그룹.
- (선택 사항) 설명: 나중에 쉽게 식별할 수 있도록 규칙에 대한 설명을 입력할 수 있습니다. 설명 길이는 최대 255자입니다. 허용되는 문자는 a-z, A-Z, 0-9, 공백 및 _-:/()#,@[]+=;{}!\$*.

보안 그룹을 규칙의 소스 또는 대상으로 지정할 경우 규칙은 보안 그룹과 연결된 모든 인스턴스에 영향을 줍니다. 유입 트래픽은 퍼블릭 IP 주소 또는 탄력적 IP 주소가 아닌 원본 보안 그룹과 연결된 인스턴스의 프라이빗 IP 주소를 기반으로 허용됩니다. IP 주소에 대한 자세한 내용은 [Amazon EC2 인스턴스 IP 주소 지정 \(p. 702\)](#) 단원을 참조하십시오. 보안 그룹 규칙이 피어 VPC의 보안 그룹을 참조하고 참조된 보안 그룹 또는 VPC 피어링 연결이 삭제된 경우, 규칙은 무효로 표시됩니다. 자세한 내용은 Amazon VPC Peering Guide의 [무효 보안 그룹 규칙으로 작업](#) 단원을 참조하십시오.

특정 포트에 대한 규칙이 여러 개 있는 경우 Amazon EC2는 최대 허용 규칙을 적용합니다. 예를 들어, IP 주소 203.0.113.1의 TCP 포트 3389(RDP) 액세스를 허용하는 규칙과 모든 사용자의 TCP 포트 3389 액세스를 허용하는 규칙이 있는 경우 모든 사용자가 TCP 포트 3389에 액세스할 수 있습니다.

연결 주제

보안 그룹은 연결 주제를 사용해 인스턴스가 송수신하는 트래픽에 대한 정보를 추적합니다. 규칙은 트래픽의 연결 상태를 기반으로 적용되어 해당 트래픽을 허용 또는 거부할지 결정합니다. 이를 통해 보안 그룹은 상태가 저장될 수 있습니다. 인바운드 트래픽에 대한 응답은 아웃바운드 보안 그룹 규칙에 관계없이 인스턴스에서 나가도록 허용되며 반대의 경우도 마찬가지입니다. 예를 들어 인바운드 보안 규칙이 ICMP 트래픽을 허용하는 경우 사용자가 자택 컴퓨터에서 인스턴스로 ICMP ping 명령을 시작하면 연결에 대한 정보(포트 정보 포함)가 추적됩니다. ping 명령에 대한 인스턴스의 응답 트래픽은 새로운 요청이 아니라 설정된 연결로 추적되며, 아웃바운드 보안 그룹 규칙이 아웃바운드 ICMP 트래픽을 제한하더라도 인스턴스에서 나가도록 허용됩니다.

모든 트래픽 흐름이 추적되지는 않습니다. 보안 그룹 규칙이 모든 트래픽(0.0.0.0/0)에 대해 TCP 또는 UDP를 허용하고, 다른 방향에서 모든 포트(0-65535)에 대해 모든 응답 트래픽(0.0.0.0/0)을 허용하는 해당 규칙이 있을 경우 해당 트래픽 흐름은 추적되지 않습니다. 그러므로 응답 트래픽이 추적 정보가 아니라 응답 트래픽을 허용하는 인바운드 또는 아웃바운드를 기반으로 흐름이 허용됩니다.

다음 예제의 보안 그룹에는 TCP 및 ICMP 트래픽에 대한 특정 인바운드 규칙과 모든 아웃 바운드 트래픽을 허용하는 아웃 바운드 규칙이 있습니다.

인바운드 규칙		
프로토콜 유형	포트 번호	소스 IP
TCP	22(SSH)	203.0.113.1/32
TCP	80(HTTP)	0.0.0.0/0
ICMP	모두	0.0.0.0/0

아웃바운드 규칙		
프로토콜 유형	포트 번호	목적지 IP
모두	모두	0.0.0.0/0

인바운드 규칙은 203.0.113.1/32의 트래픽만 허용하며 모든 IP 주소(0.0.0.0/0)를 허용하는 것은 아니기 때문에 인스턴스에서 양방향으로 이동하는 포트 22(SSH)의 TCP 트래픽이 추적됩니다. 인바운드 및 아웃바운드 규칙이 모든 TCP 트래픽(0.0.0.0/0)을 허용하기 때문에 인스턴스에서 양방향으로 이동하는 포트 80(HTTP)의 TCP 트래픽이 추적되지 않습니다. 규칙에 관계없이 ICMP 트래픽은 항상 추적됩니다. 보안 그룹에서 아웃바운드 규칙을 제거할 경우 포트 80을 통해 전송되는 트래픽(HTTP)을 비롯하여 해당 인스턴스로/에서 송/수신되는 모든 트래픽이 추적됩니다.

흐름을 허용하는 규칙이 제거 또는 수정될 경우 추적되지 않는 트래픽 흐름이 즉시 중단됩니다. 예를 들어 열린 (0.0.0.0/0) 아웃바운드 규칙이 있고 인스턴스에 대한 모든 (0.0.0.0/0) 인바운드 SSH(TCP 포트 22) 트래픽을 허용하는 규칙을 제거하거나 연결이 더 이상 허용되지 않도록 수정할 경우 인스턴스에 대한 기존 SSH 연결이 즉시 삭제됩니다. 이전에 연결이 추적되지 않았으므로 변경으로 인해 연결이 끊어집니다. 반면에 처음에는 SSH 연결을 허용하는(연결이 추적되었음을 의미) 보다 좁은 범위의 인바운드 규칙이 있지만 현재 SSH 클라이언트의 주소에서 새 연결을 더 이상 허용하지 않도록 해당 규칙을 변경할 경우 규칙 변경으로 인해 기존 연결이 끊어지지 않습니다.

TCP, UDP 또는 ICMP 이외의 프로토콜에 대해서는 IP 주소와 프로토콜 번호만 추적됩니다. 인스턴스가 다른 호스트(호스트 B)로 트래픽을 보내고 호스트 B가 원래 요청 또는 응답으로부터 600초 이내에 별도의 요청으로 사용자의 인스턴스에 대해 동일한 유형의 트래픽을 시작할 경우 인스턴스는 인바운드 보안 그룹 규칙에 관계없이 해당 트래픽을 수락합니다. 인스턴스가 응답 트래픽으로 간주되기 때문에 트래픽을 수락하는 것입니다.

보안 그룹 규칙을 제거하는 즉시 트래픽이 중단되도록 하거나 모든 인바운드 트래픽이 방화벽 규칙에 따르도록 하려면 서브넷의 네트워크 ACL을 사용할 수 있습니다. 네트워크 ACL은 상태 비저장이므로 자동으로 응답 트래픽을 허용하지 않습니다. 자세한 내용은 Amazon VPC 사용 설명서의 [네트워크 ACL](#)을 참조하십시오.

기본 보안 그룹

AWS 계정에 각 지역의 기본 VPC에 대한 기본 보안 그룹이 자동으로 생성됩니다. 인스턴스를 시작할 때 보안 그룹을 지정하지 않을 경우 VPC에 대해 인스턴스는 기본 보안 그룹과 자동으로 연결됩니다.

기본 보안 그룹의 이름은 `default`고 AWS에 의해 ID가 배정됩니다. 각 기본 보안 그룹에 대한 기본 규칙은 다음과 같습니다.

- 기본 보안 그룹과 연결된 다른 인스턴스의 모든 인바운드 트래픽 허용 보안 그룹은 인바운드 규칙에서 소스 보안 그룹으로 지정됩니다.
- 인스턴스의 모든 아웃바운드 트래픽을 허용합니다.

기본 보안 그룹에 대한 인바운드 및 아웃바운드 규칙을 추가하거나 제거할 수 있습니다.

기본 보안 그룹을 삭제할 수 없습니다. 기본 보안 그룹을 삭제하려고 하면 `client.CannotDelete: the specified group: "sg-51530134" name: "default" cannot be deleted by a user`라는 오류가 표시됩니다.

사용자 지정 보안 그룹

인스턴스에서 기본 보안 그룹을 사용하지 않도록 하려면 고유한 보안 그룹을 생성하고 인스턴스를 시작할 때 해당 보안 그룹을 지정합니다. 인스턴스가 수행하는 다양한 역할(예: 웹 서버, 데이터베이스 서버)을 반영하는 여러 보안 그룹을 생성할 수 있습니다.

보안 그룹을 생성할 때 이름과 설명을 제공해야 합니다. 보안 그룹의 이름과 설명은 최대 255자이며 다음과 같은 문자로 제한됩니다.

a-z, A-Z, 0-9, 공백 및 _:-/()#@[]+=&;!\$*

보안 그룹 이름은 `sg-`로 시작할 수 없습니다. 보안 그룹 이름은 VPC 내에서 고유해야 합니다.

다음은 생성하는 보안 그룹의 기본 규칙입니다.

- 인바운드 트래픽을 허용 안 함
- 모든 아웃바운드 트래픽을 허용합니다

보안 그룹을 생성한 후 연결된 인스턴스에 도달할 인바운드 트래픽의 유형을 반영하도록 인바운드 규칙을 변경할 수 있습니다. 아웃바운드 규칙도 변경할 수 있습니다.

보안 그룹에 추가할 수 있는 규칙에 대한 자세한 내용은 [보안 그룹 규칙 참조 \(p. 919\)](#) 단원을 참조하십시오.

보안 그룹 작업

인스턴스를 시작할 때 인스턴스에 보안 그룹을 할당할 수 있습니다. 규칙을 추가하거나 제거하면 해당 보안 그룹을 할당한 모든 인스턴스에 변경 내용이 자동으로 적용됩니다.

인스턴스를 시작한 이후에는 해당 보안 그룹을 변경할 수 없습니다. 자세한 내용은 [Amazon VPC 사용 설명서](#)의 인스턴스의 보안 그룹 변경 단원을 참조하십시오.

Amazon EC2 콘솔 및 명령줄 도구를 사용하여 보안 그룹과 보안 그룹 규칙을 생성하고 보고 업데이트하고 삭제할 수 있습니다.

작업

- [보안 그룹 생성 \(p. 912\)](#)
- [보안 그룹 복사 \(p. 913\)](#)
- [보안 그룹 보기 \(p. 913\)](#)
- [보안 그룹에 규칙 추가 \(p. 914\)](#)
- [보안 그룹 규칙 업데이트 \(p. 917\)](#)
- [보안 그룹에서 규칙 삭제 \(p. 918\)](#)
- [보안 그룹 삭제 \(p. 918\)](#)

보안 그룹 생성

다음 방법 중 하나를 사용하여 사용자 지정 보안 그룹을 생성할 수 있습니다. 보안 그룹을 생성할 VPC를 지정해야 합니다.

새로운 콘솔

보안 그룹을 생성하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 열니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 보안 그룹 생성을 선택합니다.
4. Basic details(기본 세부 정보) 섹션에서 다음을 수행합니다.
 - a. 보안 그룹의 설명이 포함된 이름과 간단한 설명을 입력합니다. 이름과 설명은 최대 255자까지 입력할 수 있으며 a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{ }!\$*을 포함할 수 있습니다.
 - b. VPC의에서 보안 그룹을 생성할 VPC를 선택합니다. 보안 그룹은 생성된 VPC에서만 사용할 수 있습니다.
5. 보안 그룹 규칙을 지금 추가하거나 보안 그룹을 생성한 후 언제든지 추가할 수 있습니다. 보안 그룹 규칙 추가에 대한 자세한 내용은 [보안 그룹에 규칙 추가 \(p. 914\)](#) 단원을 참조하십시오.
6. Create를 선택합니다.

기존 콘솔

보안 그룹을 생성하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 열니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.

3. 보안 그룹 생성을 선택합니다.
4. 보안 그룹의 이름과 설명을 지정합니다.
5. VPC에서 VPC의 ID를 선택합니다.
6. 규칙을 추가할 수 있습니다. 또는 생성을 선택하여 지금 보안 그룹을 생성하고 나중에 규칙을 추가할 수 있습니다. 규칙 추가에 대한 자세한 내용은 [보안 그룹에 규칙 추가 \(p. 914\)](#) 단원을 참조하십시오.

Command line

보안 그룹을 생성하는 방법

다음 명령 중 하나를 사용합니다.

- [create-security-group\(AWS CLI\)](#)
- [New-EC2SecurityGroup\(Windows PowerShell용 AWS 도구\)](#)

보안 그룹 복사

기존 보안 그룹의 복사본을 생성하여 새 보안 그룹을 만들 수 있습니다. 보안 그룹을 복사하면 원래 보안 그룹과 동일한 인바운드 및 아웃바운드 규칙을 사용하여 복사본이 생성됩니다. 원래 보안 그룹이 VPC에 있는 경우 다른 보안 그룹을 지정하지 않는 한 동일한 VPC에 복사본이 생성됩니다.

복사본은 새 고유 보안 그룹 ID를 받게 되므로 별도로 이름을 지정해야 합니다. 설명을 추가할 수도 있습니다.

보안 그룹을 한 리전에서 다른 리전으로 복사할 수 없습니다.

다음 방법 중 하나를 사용하여 보안 그룹의 복사본을 생성할 수 있습니다.

새로운 콘솔

보안 그룹을 복사하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 복사할 보안 그룹을 선택하고 작업, Copy to new security group(새 보안 그룹에 복사)을 선택합니다.
4. 이름과 설명(선택 사항)을 지정하고 필요한 경우 VPC 및 보안 그룹 규칙을 변경합니다.
5. Create를 선택합니다.

기존 콘솔

보안 그룹을 복사하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 복사할 보안 그룹을 선택하고 작업과 새로 복사를 차례로 선택합니다.
4. 보안 그룹 생성 대화 상자가 열리고 기존 보안 그룹의 규칙으로 채워집니다. 새 보안 그룹의 이름과 설명을 지정합니다. VPC에서 VPC의 ID를 선택합니다. 완료했으면 생성을 선택합니다.

보안 그룹 보기

다음 방법 중 하나를 사용하여 보안 그룹에 대한 정보를 볼 수 있습니다.

새로운 콘솔

보안 그룹을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 보안 그룹이 나열됩니다. 인바운드 및 아웃바운드 규칙을 포함하여 특정 보안 그룹에 대한 세부 정보를 보려면 보안 그룹 ID 열에서 해당 ID를 선택합니다.

기존 콘솔

보안 그룹을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. (선택 사항) 필터 목록에서 VPC ID를 선택한 다음 VPC의 ID를 선택합니다.
4. 보안 그룹을 선택합니다. 설명 탭에 일반 정보가 표시되고, 인바운드 탭에 인바운드 규칙이 표시되고, 아웃바운드 탭에 아웃바운드 규칙이 표시되고, 태그 탭에 태그가 표시됩니다.

Command line

보안 그룹을 보려면

다음 명령 중 하나를 사용합니다.

- [describe-security-groups](#)(AWS CLI)
- [Get-EC2SecurityGroup](#)(Windows PowerShell용 AWS 도구)

보안 그룹에 규칙 추가

보안 그룹에 규칙을 추가할 경우 보안 그룹과 연결된 인스턴스에 새 규칙이 자동으로 적용됩니다. 규칙이 적용되기 전에 약간의 지연이 있을 수 있습니다. 특정 유형의 액세스를 위한 보안 그룹 규칙 선택에 대한 자세한 내용은 [보안 그룹 규칙 참조 \(p. 919\)](#) 단원을 참조하십시오. 보안 그룹 규칙 할당량에 대한 자세한 내용은 Amazon VPC 사용 설명서의 [Amazon VPC 할당량](#)을 참조하세요.

다음 방법 중 하나를 사용하여 보안 그룹에 규칙을 추가할 수 있습니다.

새로운 콘솔

보안 그룹에 인바운드 규칙을 추가하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 목록에서 보안 그룹을 선택하고 작업, 인바운드 규칙 편집을 선택합니다.
4. 규칙 추가를 선택하고 다음을 수행합니다.
 - a. 유형에서 허용할 프로토콜 유형을 선택합니다.
 - 사용자 지정 TCP 또는 UDP 프로토콜을 선택하는 경우 허용할 포트 범위를 수동으로 입력해야 합니다.
 - 사용자 지정 ICMP 프로토콜을 선택하는 경우 프로토콜에서 ICMP 유형 이름을 선택하고, 해당되는 경우 포트 범위에서 코드 이름을 선택해야 합니다.
 - 다른 유형을 선택하면 프로토콜과 포트 범위가 자동으로 구성됩니다.

b. 소스에서 다음 중 하나를 수행합니다.

- 사용자 지정을 선택한 다음 CIDR 표기법의 IP 주소, CIDR 블록, 다른 보안 그룹 또는 인바운드 트래픽을 허용할 접두사 목록을 입력합니다.
- 지정된 프로토콜의 모든 인바운드 트래픽이 인스턴스에 도달하도록 허용하려면 위치 무관을 선택합니다. 이 옵션은 자동으로 0.0.0.0/0 IPv4 CIDR 블록을 허용된 소스로 추가합니다. 테스트 환경에서 잠시 사용하는 것은 괜찮지만 프로덕션 환경에서는 안전하지 않습니다. 프로덕션에서는 특정 IP 주소나 주소 범위만 인스턴스에 액세스하도록 허용하십시오.

보안 그룹이 IPv6를 사용하도록 설정된 VPC에 있는 경우 이 옵션은 자동으로 IPv6 트래픽에 대한 두 번째 규칙(: :: /0)을 추가합니다.

- 로컬 컴퓨터의 퍼블릭 IPv4 주소에서 들어오는 인바운드 트래픽만 허용하려면 내 IP를 선택합니다.

c. 필요한 경우 설명에 규칙에 대한 간단한 설명을 지정합니다.

5. 변경 사항 미리 보기, 규칙 저장을 선택합니다.

보안 그룹에 아웃바운드 규칙을 추가하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 [Security Groups]를 선택합니다.
- 목록에서 보안 그룹을 선택하고 작업, 아웃바운드 규칙 편집을 선택합니다.
- 규칙 추가를 선택하고 다음을 수행합니다.

a. 유형에서 허용할 프로토콜 유형을 선택합니다.

- 사용자 지정 TCP 또는 UDP 프로토콜을 선택하는 경우 허용할 포트 범위를 수동으로 입력해야 합니다.
- 사용자 지정 ICMP 프로토콜을 선택하는 경우 프로토콜에서 ICMP 유형 이름을 선택하고, 해당되는 경우 포트 범위에서 코드 이름을 선택해야 합니다.
- 다른 유형을 선택하면 프로토콜과 포트 범위가 자동으로 구성됩니다.

b. 대상에서 다음 중 하나를 수행합니다.

- 사용자 지정을 선택한 다음 CIDR 표기법의 IP 주소, CIDR 블록, 다른 보안 그룹 또는 아웃바운드 트래픽을 허용할 접두사 목록을 입력합니다.
- 모든 IP 주소에 대한 아웃바운드 트래픽을 허용하려면 위치 무관을 선택합니다. 이 옵션은 자동으로 0.0.0.0/0 IPv4 CIDR 블록을 허용된 소스로 추가합니다.

보안 그룹이 IPv6를 사용하도록 설정된 VPC에 있는 경우 이 옵션은 자동으로 IPv6 트래픽에 대한 두 번째 규칙(: :: /0)을 추가합니다.

- 로컬 컴퓨터의 퍼블릭 IPv4 주소로 나가는 아웃바운드 트래픽만 허용하려면 내 IP를 선택합니다.

c. 필요한 경우 설명에 규칙에 대한 간단한 설명을 지정합니다.

5. Preview changes(변경 사항 미리 보기), 확인을 선택합니다.

기존 콘솔

보안 그룹에 규칙을 추가하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 보안 그룹을 선택하고 보안 그룹을 선택합니다.
- 인바운드 탭에서 편집을 선택합니다.
- 대화 상자에서 규칙 추가를 선택하고 다음과 같이 실행합니다.

- 유형에는 프로토콜을 선택합니다.
- 사용자 지정 TCP 또는 UDP 프로토콜을 선택하는 경우 포트 범위에 포트 범위를 지정합니다.
- 사용자 지정 ICMP 프로토콜을 선택하는 경우 프로토콜에서 ICMP 유형 이름을 선택하고, 해당되는 경우 포트 범위에서 코드 이름을 선택합니다.
- 소스의 경우 다음 중 하나를 선택합니다.
 - 사용자 지정: 제공되는 필드에 IP 주소(CIDR 표기법), CIDR 블록 또는 다른 보안 그룹을 지정해야 합니다.
 - 위치 무관: 0.0.0.0/0 IPv4 CIDR 블록을 자동으로 추가합니다. 이 옵션으로 지정된 유형의 모든 트래픽이 인스턴스에 도착하도록 할 수 있습니다. 테스트 환경에서 잠시 사용하는 것은 괜찮지만 프로덕션 환경에서는 안전하지 않습니다. 프로덕션에서는 특정 IP 주소나 주소 범위만 인스턴스에 액세스하도록 허용하십시오.

보안 그룹이 IPv6용으로 사용되는 VPC에 있는 경우, 위치 무관 옵션을 선택하면 IPv4 트래픽에 대해 한 개(0.0.0.0/0), IPv6 트래픽에 대해 한 개(:/:0), 총 2개의 규칙이 생성됩니다.

- 내 IP: 로컬 컴퓨터의 퍼블릭 IPv4 주소를 자동으로 추가합니다.
- 필요한 경우 설명에 규칙에 대한 설명을 지정합니다.

추가할 수 있는 규칙의 유형에 대한 자세한 내용은 [보안 그룹 규칙 참조 \(p. 919\)](#) 단원을 참조하십시오.

5. 저장을 선택합니다.
6. 아웃바운드 규칙을 지정할 수도 있습니다. 아웃바운드 탭에서 편집, 규칙 추가를 선택하고 다음 작업을 수행합니다.
 - 유형에는 프로토콜을 선택합니다.
 - 사용자 지정 TCP 또는 UDP 프로토콜을 선택하는 경우 포트 범위에 포트 범위를 지정합니다.
 - 사용자 지정 ICMP 프로토콜을 선택하는 경우 프로토콜에서 ICMP 유형 이름을 선택하고, 해당되는 경우 포트 범위에서 코드 이름을 선택합니다.
 - 대상의 경우 다음 중 하나를 선택합니다.
 - 사용자 지정: 제공되는 필드에 IP 주소(CIDR 표기법), CIDR 블록 또는 다른 보안 그룹을 지정해야 합니다.
 - 위치 무관: 0.0.0.0/0 IPv4 CIDR 블록을 자동으로 추가합니다. 이 옵션을 선택하면 모든 IP 주소로 아웃바운드 트래픽이 전송됩니다.
7. 저장을 선택합니다.

Command line

보안 그룹에 규칙을 추가하려면

다음 명령 중 하나를 사용합니다.

- [authorize-security-group-ingress](#)(AWS CLI)
- [Grant-EC2SecurityGroupIngress](#)(Windows PowerShell용 AWS 도구)

보안 그룹에 하나 이상의 송신 규칙을 추가하려면

다음 명령 중 하나를 사용합니다.

- [authorize-security-group-egress\(AWS CLI\)](#)
- [Grant-EC2SecurityGroupEgress\(Windows PowerShell용 AWS 도구\)](#)

보안 그룹 규칙 업데이트

다음 방법 중 하나를 사용하여 보안 그룹 규칙을 업데이트할 수 있습니다.

새로운 콘솔

콘솔을 사용하여 기존 보안 그룹의 프로토콜, 포트 범위 또는 소스/목적지를 수정하면 콘솔은 기존 규칙을 삭제하고 새 규칙을 추가합니다.

보안 그룹 규칙을 업데이트하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 업데이트할 보안 그룹을 선택하고 작업을 선택한 다음, 인바운드 트래픽에 대한 규칙을 업데이트하려면 인바운드 규칙 편집을, 아웃바운드 트래픽에 대한 규칙을 업데이트하려면 아웃바운드 규칙 편집을 선택합니다.
4. 필요에 따라 규칙을 업데이트한 후 Preview changes(변경 사항 미리 보기), 확인을 선택합니다.

기존 콘솔

콘솔을 사용하여 기존 보안 그룹의 프로토콜, 포트 범위 또는 소스/목적지를 수정하면 콘솔은 기존 규칙을 삭제하고 새 규칙을 추가합니다.

보안 그룹 규칙을 업데이트하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 업데이트할 보안 그룹을 선택하고 인바운드 트래픽에 대한 규칙을 업데이트하려면 인바운드 탭을, 아웃바운드 트래픽에 대한 규칙을 업데이트하려면 아웃바운드 탭을 선택합니다.
4. [Edit]를 선택합니다.
5. 필요에 따라 규칙 항목을 수정한 다음, 저장을 선택합니다.

Command line

Amazon EC2 API 또는 명령줄 도구를 사용하여 기존 규칙의 프로토콜, 포트 범위, 소스 및 대상을 수정할 수 있습니다. 대신 기존 규칙을 삭제하고 새 규칙을 추가해야 합니다. 그러나 기존 규칙에 대한 설명을 업데이트할 수 있습니다.

기존 인바운드 규칙에 대한 설명을 업데이트하려면

다음 명령 중 하나를 사용합니다.

- [update-security-group-rule-descriptions-ingress\(AWS CLI\)](#)
- [Update-EC2SecurityGroupRuleIngressDescription\(Windows PowerShell용 AWS 도구\)](#)

기존 아웃바운드 규칙에 대한 설명을 업데이트하려면

다음 명령 중 하나를 사용합니다.

- [update-security-group-rule-descriptions-egress\(AWS CLI\)](#)

- [Update-EC2SecurityGroupRuleEgressDescription](#)(Windows PowerShell용 AWS 도구)

보안 그룹에서 규칙 삭제

보안 그룹에서 규칙을 삭제할 경우 보안 그룹과 연결된 인스턴스에 해당 변경 내용이 자동으로 적용됩니다.

다음 방법 중 하나를 사용하여 보안 그룹에서 규칙을 삭제할 수 있습니다.

새로운 콘솔

보안 그룹 규칙을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 업데이트 할 보안 그룹을 선택하고 작업을 선택한 다음, 인바운드 규칙을 제거하려면 인바운드 규칙 편집을, 아웃바운드 규칙을 제거하려면 아웃바운드 규칙 편집을 선택합니다.
4. 삭제할 규칙 오른쪽에 있는 제거 버튼을 선택합니다.
5. Preview changes(변경 사항 미리 보기), 확인을 선택합니다.

기존 콘솔

보안 그룹 규칙을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹을 선택합니다.
4. 인바운드 탭(인바운드 규칙) 또는 아웃바운드 탭(아웃바운드 규칙)에서 편집을 선택합니다. 삭제할 각 규칙 옆의 삭제(x 아이콘)를 선택합니다.
5. 저장을 선택합니다.

Command line

보안 그룹에서 하나 이상의 수신 규칙을 제거하려면

다음 명령 중 하나를 사용합니다.

- [revoke-security-group-ingress](#)(AWS CLI)
- [Revoke-EC2SecurityGroupIngress](#)(Windows PowerShell용 AWS 도구)

보안 그룹에서 하나 이상의 송신 규칙을 제거하려면

다음 명령 중 하나를 사용합니다.

- [revoke-security-group-egress](#)(AWS CLI)
- [Revoke-EC2SecurityGroupEgress](#)(Windows PowerShell용 AWS 도구)

보안 그룹 삭제

인스턴스와 연결된 보안 그룹과 기본 보안 그룹은 삭제할 수 없습니다. 같은 VPC에 있는 다른 보안 그룹의 규칙에서 참조하는 보안 그룹도 삭제할 수 없습니다. 자체 규칙 중 하나에서 보안 그룹이 참조하는 경우 보안 그룹을 삭제하려면 해당 규칙을 삭제해야 합니다.

새로운 콘솔

보안 그룹을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 삭제할 보안 그룹을 선택하고 작업, 보안 그룹 삭제, 삭제를 선택합니다.

기존 콘솔

보안 그룹을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹을 선택한 다음 작업, 보안 그룹 삭제를 선택합니다.
4. 예, 삭제를 선택합니다.

Command line

보안 그룹을 삭제하려면

다음 명령 중 하나를 사용합니다.

- [delete-security-group](#)(AWS CLI)
- [Remove-EC2SecurityGroup](#)(Windows PowerShell용 AWS 도구)

보안 그룹 규칙 참조

보안 그룹을 생성하고 보안 그룹과 연결된 인스턴스의 역할을 반영하는 규칙을 추가합니다. 예를 들어 웹 서버로 구성된 인스턴스에는 인바운드 HTTP 및 HTTPS 액세스를 허용하는 보안 그룹 규칙이 필요합니다. 마찬가지로 데이터베이스 인스턴스에는 데이터베이스 유형에 대한 액세스(예: MySQL의 경우 포트 3306을 통한 액세스)를 허용하는 규칙이 필요합니다.

다음은 특정한 종류의 액세스에 대해 보안 그룹에 추가할 수 있는 규칙의 종류를 예로 든 것입니다.

예제:

- 웹 서버 규칙 (p. 919)
- 데이터베이스 서버 규칙 (p. 920)
- 컴퓨터에서 인스턴스 연결에 대한 규칙 (p. 921)
- 보안 그룹이 동일한 인스턴스에서 인스턴스에 대한 연결 규칙 (p. 922)
- Ping/ICMP 규칙 (p. 922)
- DNS 서버 규칙 (p. 922)
- Amazon EFS 규칙 (p. 923)
- Elastic Load Balancing 규칙 (p. 923)
- VPC 피어링 규칙 (p. 924)

웹 서버 규칙

다음 인바운드 규칙에서는 임의의 IP 주소로부터 HTTP 및 HTTPS 액세스를 허용합니다. VPC가 IPv6용으로 활성화되면 IPv6 주소에서 인바운드 HTTP 및 HTTPS 트래픽을 제어하기 위한 규칙을 추가할 수 있습니다.

프로토콜 유형	프로토콜 번호	포트	소스 IP	참고
TCP	6	80(HTTP)	0.0.0.0/0	임의의 IPv4 주소에서 인바운드 HTTP 액세스를 허용함
TCP	6	443(HTTPS)	0.0.0.0/0	임의의 IPv4 주소에서 인바운드 HTTPS 액세스를 허용함
TCP	6	80(HTTP)	::/0	임의의 IPv6 주소에서 인바운드 HTTP 액세스를 허용함
TCP	6	443(HTTPS)	::/0	임의의 IPv6 주소에서 인바운드 HTTPS 액세스를 허용함

데이터베이스 서버 규칙

다음의 인바운드 규칙은 인스턴스에서 실행 중인 데이터베이스의 유형에 따라 데이터베이스 액세스를 위해 추가할 수 있는 규칙을 예로 든 것입니다. Amazon RDS 인스턴스에 대한 자세한 내용은 [Amazon RDS 사용 설명서](#)를 참조하십시오.

원본 IP의 경우 다음 중 하나를 지정합니다.

- 로컬 네트워크의 특정 IP 주소 또는 IP 주소 범위(CIDR 블록 표기법)
- 데이터베이스에 액세스하는 인스턴스 그룹의 보안 그룹 ID

프로토콜 유형	프로토콜 번호	포트	참고
TCP	6	1433(MS SQL)	Microsoft SQL Server 데이터베이스 액세스를 위한 기본 포트(예: Amazon RDS 인스턴스에서)
TCP	6	3306(MYSQL/Aurora)	MySQL 또는 Aurora 데이터베이스 액세스를 위한 기본 포트(예: Amazon RDS 인스턴스에서)
TCP	6	5439(Redshift)	Amazon Redshift 클러스터 데이터베이스 액세스를 위한 기본 포트.
TCP	6	5432(PostgreSQL)	PostgreSQL 데이터베이스 액세스를 위한 기본 포트(예: Amazon RDS 인스턴스에서)
TCP	6	1521(Oracle)	Oracle 데이터베이스 액세스를 위한 기본 포트

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
보안 그룹 규칙 참조

프로토콜 유형	프로토콜 번호	포트	참고
			(예: Amazon RDS 인스턴스에서)

선택적으로 데이터베이스 서버의 아웃바운드 트래픽을 제한할 수 있습니다. 예를 들어 소프트웨어 업데이트를 위해 인터넷 액세스를 허용하지만 다른 모든 종류의 트래픽은 제한할 수 있습니다. 먼저 모든 아웃바운드 트래픽을 허용하는 기본 아웃바운드 규칙을 제거해야 합니다.

프로토콜 유형	프로토콜 번호	포트	목적지 IP	참고
TCP	6	80(HTTP)	0.0.0.0/0	임의의 IPv4 주소에 대한 아웃바운드 HTTP 액세스를 허용함
TCP	6	443(HTTPS)	0.0.0.0/0	임의의 IPv4 주소에 대한 아웃바운드 HTTPS 액세스를 허용함
TCP	6	80(HTTP)	::/0	(IPv6 사용 VPC만 해당) 임의의 IPv6 주소에 대한 아웃바운드 HTTP 액세스를 허용함
TCP	6	443(HTTPS)	::/0	(IPv6 사용 VPC만 해당) 임의의 IPv6 주소에 대한 아웃바운드 HTTPS 액세스를 허용함

컴퓨터에서 인스턴스 연결에 대한 규칙

인스턴스에 연결하려면 보안 그룹에 SSH 액세스(Linux 인스턴스) 또는 RDP 액세스(Windows 인스턴스)를 허용하는 인바운드 규칙이 있어야 합니다.

프로토콜 유형	프로토콜 번호	포트	소스 IP
TCP	6	22(SSH)	컴퓨터의 퍼블릭 IPv4 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법). IPv6를 위해 VPC가 활성화되어 있고 인스턴스에 IPv6 주소가 있는 경우 IPv6 주소 또는 범위를 입력할 수 있습니다.
TCP	6	3389(RDP)	컴퓨터의 퍼블릭 IPv4 주소 또는 로컬 네트워크의 IP 주소 범위(CIDR 블록 표기법). IPv6를 위해 VPC가 활성화되어 있고

프로토콜 유형	프로토콜 번호	포트	소스 IP
			인스턴스에 IPv6 주소가 있는 경우 IPv6 주소 또는 범위를 입력할 수 있습니다.

보안 그룹이 동일한 인스턴스에서 인스턴스에 대한 연결 규칙

같은 보안 그룹과 연결된 여러 인스턴스가 서로 통신할 수 있게 하려면 이에 대한 규칙을 명시적으로 추가해야 합니다.

다음 표에서는 연결된 인스턴스가 서로 통신할 수 있도록 하기 위한 보안 그룹의 인바운드 규칙을 설명합니다. 이 규칙에서는 모든 유형의 트래픽을 허용합니다.

프로토콜 유형	프로토콜 번호	포트	소스 IP
-1(모두)	-1(모두)	-1(모두)	보안 그룹의 ID

Ping/ICMP 규칙

ping 명령은 ICMP 트래픽의 한 유형입니다. 인스턴스를 ping하려면 다음 인바운드 ICMP 규칙을 추가해야 합니다.

프로토콜 유형	프로토콜 번호	ICMP 유형	ICMP 코드	소스 IP
ICMP	1	8(에코)	해당 사항 없음	컴퓨터의 퍼블릭 IPv4 주소 또는 로컬 네트워크의 IPv4 주소 범위(CIDR 블록 표기법).

ping6 명령을 사용하여 인스턴스에 대한 IPv6 주소를 ping하려면 다음 인바운드 ICMPv6 규칙을 추가해야 합니다.

프로토콜 유형	프로토콜 번호	ICMP 유형	ICMP 코드	소스 IP
ICMPv6	58	128(에코)	0	컴퓨터의 IPv6 주소 또는 로컬 네트워크의 IPv6 주소 범위(CIDR 블록 표기법).

DNS 서버 규칙

EC2 인스턴스를 DNS 서버로 설정한 경우 TCP 및 UDP 트래픽이 포트 53을 통해 DNS 서버에 연결할 수 있는지 확인해야 합니다.

원본 IP의 경우 다음 중 하나를 지정합니다.

- 네트워크의 IP 주소 또는 IP 주소 범위(CIDR 블록 표기법)

- 네트워크에서 DNS 서버로의 액세스를 필요로 하는 인스턴스 세트에 대한 보안 그룹의 ID

프로토콜 유형	프로토콜 번호	포트
TCP	6	53
UDP	17	53

Amazon EFS 규칙

Amazon EC2 인스턴스에서 Amazon EFS 파일 시스템을 사용하려면 Amazon EFS 마운트 대상과 연결되는 보안 그룹이 NFS 프로토콜을 통한 트래픽 전송을 허용해야 합니다.

프로토콜 유형	프로토콜 번호	포트	소스 IP	참고
TCP	6	2049(NFS)	보안 그룹의 ID.	이 보안 그룹과 연결된 리소스(탑재 대상 포함)에서 인바운드 NFS 액세스를 허용합니다.

Amazon EFS 파일 시스템을 Amazon EC2 인스턴스에 마운트하려면 인스턴스에 연결해야 합니다. 따라서 인스턴스와 연결되는 보안 그룹은 로컬 컴퓨터 또는 로컬 네트워크의 인바운드 SSH 트래픽을 허용하는 규칙이 필요합니다.

프로토콜 유형	프로토콜 번호	포트	소스 IP	참고
TCP	6	22(SSH)	로컬 컴퓨터의 IP 주소 범위 또는 네트워크의 IP 주소 범위(CIDR 블록 표기법).	로컬 컴퓨터로부터의 인바운드 SSH 액세스를 허용합니다.

Elastic Load Balancing 규칙

로드 밸런서를 사용하고 있는 경우 로드 밸런서에 연결된 보안 그룹은 인스턴스 또는 대상과 통신을 허용하는 규칙을 보유해야 합니다.

인바운드				
프로토콜 유형	프로토콜 번호	포트	소스 IP	참고
TCP	6	리스너 포트	인터넷 경계 로드 밸런서의 경우: 0.0.0.0/0(모든 IPv4 주소) 내부 로드 밸런서의 경우: VPC의 IPv4 CIDR 블록	로드 밸런서 리스너 포트의 인바운드 트래픽을 허용합니다.

아웃바운드

프로토콜 유형	프로토콜 번호	포트	목적지 IP	참고
TCP	6	인스턴스 리스너 포트	인스턴스 보안 그룹의 ID	인스턴스 리스너 포트의 인스턴스로 아웃바운드 트래픽을 허용합니다.
TCP	6	상태 확인 포트	인스턴스 보안 그룹의 ID	상태 확인 포트의 인스턴스로 아웃바운드 트래픽을 허용합니다.

인스턴스에 대한 보안 그룹 규칙은 로드 밸런서가 리스너 포트 및 상태 확인 포트에서 인스턴스와 통신할 수 있도록 허용해야 합니다.

인바운드

프로토콜 유형	프로토콜 번호	포트	소스 IP	참고
TCP	6	인스턴스 리스너 포트	로드 밸런서 보안 그룹의 ID	인스턴스 리스너 포트의 로드 밸런서에서 트래픽을 허용합니다.
TCP	6	상태 확인 포트	로드 밸런서 보안 그룹의 ID	상태 확인 포트의 로드 밸런서에서 트래픽을 허용합니다.

자세한 내용은 Classic Load Balancer 사용 설명서의 [Classic Load Balancer에 대한 보안 그룹 구성](#)과 Application Load Balancer 사용 설명서의 [Application Load Balancer에 대한 보안 그룹을 참조하십시오.](#)

VPC 피어링 규칙

피어링된 VPC의 보안 그룹을 참조하도록 VPC 보안 그룹의 인바운드 또는 아웃바운드 규칙을 업데이트할 수 있습니다. 그렇게 하면 피어링된 VPC의 참조 보안 그룹과 연결된 인스턴스 간에 트래픽을 주고받을 수 있습니다. VPC 피어링을 위한 보안 그룹을 구성하는 방법에 대한 자세한 내용은 [피어 VPC 그룹을 참조하도록 보안 그룹 업데이트](#)를 참조하십시오.

Amazon EC2의 구성 관리

Amazon 머신 이미지(AMI)는 Amazon EC2 인스턴스에 대한 초기 구성을 제공합니다. 여기에는 Windows OS와 애플리케이션 및 보안 제어 같은 고객별 사용자 지정 옵션 등이 포함됩니다. 사용자 지정된 보안 구성 기준이 들어 있는 AMI 카탈로그를 생성하여 모든 Windows 인스턴스가 표준 보안 제어로 시작되도록 합니다. 보안 기준을 AMI에 베이킹하거나, EC2 인스턴스가 시작될 때 동적으로 부트스트래핑하거나, 하나의 제품으로 패키징하여 AWS 서비스 카탈로그 포트폴리오를 통해 균일하게 배포할 수 있습니다. AMI 보안 유지에 대한 자세한 내용은 [AMI 작성 모범 사례](#)를 참조하십시오.

각 Amazon EC2 인스턴스는 조직의 보안 표준을 준수해야 합니다. 필요 없는 Windows 역할 및 기능을 설치하지 말고, 악성 코드로부터 보호하고(바이러스 백신, 맬웨어 방지, 악용 완화) 호스트 무결성을 모니터링하고 침입 탐지를 수행하는 소프트웨어를 설치하십시오. OS 보안 설정을 모니터링 및 유지 관리하고, 중요한 OS 파일의 무결성을 보호하며, 보안 기준에서 벗어나면 경보를 보내도록 보안 소프트웨어를 구성합니다.

Microsoft, CIS(인터넷 보안 센터) 또는 NIST(미국 국립 표준 기술 연구소)에서 게시한 권장 보안 구성 벤치 마크를 구현하는 것이 좋습니다. [SQL Server용 모범 사례 분석기](#) 등 특정 애플리케이션 서버를 위한 다른 Microsoft 도구를 사용하는 방법을 고려하십시오.

AWS 고객은 또한 Amazon Inspector 평가를 실행하여 Amazon EC2 인스턴스에 배포된 애플리케이션의 보안 및 규정 준수를 향상시킬 수 있습니다. Amazon Inspector는 애플리케이션의 취약성 또는 모범 사례와의 편차가 있는지 자동으로 평가하며, 공통의 보안 규정 준수 표준(예: PCI DSS)과 취약성 정의에 매핑된 수백 가지 규칙 지식 기반을 포함합니다. 원격 루트 로그인이 활성화되어 있는지 또는 취약한 소프트웨어 버전이 설치되어 있는지 확인하는 것이 대표적인 기본 제공 규칙입니다. 이러한 규칙은 AWS AWS 보안 연구원이 정기적으로 업데이트합니다.

Amazon EC2의 업데이트 관리

EC2 인스턴스에서 운영 체제와 애플리케이션을 정기적으로 패치, 업데이트 및 보호하는 것이 좋습니다. [AWS 시스템 관리자 Patch Manager](#)를 사용하여 운영 체제와 애플리케이션 모두에 대해 보안 관련 업데이트를 설치하는 프로세스를 자동화할 수 있습니다. 또는 자동 업데이트 서비스를 사용하거나, 애플리케이션 공급업체에서 제공하는 업데이트를 설치하기 위한 권장 프로세스를 사용할 수 있습니다.

Windows Server를 실행하는 Amazon EC2 인스턴스에서 Windows 업데이트를 구성해야 합니다. 기본적으로, AWS가 제공한 AMI에 대한 Windows 업데이트는 고객에게 전달되지 않습니다. Windows Server를 실행하는 최신 Amazon EC2 AMI 목록은 [AWS Windows AMI 버전 세부 정보](#)를 참조하십시오.

Amazon EC2의 변경 관리

시작 시 Amazon EC2 인스턴스에 초기 보안 기준이 되고, 그 이후에는 진행 중인 Amazon EC2 변경 사항을 제어하여 가상 시스템의 보안을 유지합니다. AWS 리소스(예: 보안 그룹, 라우팅 테이블, 네트워크 ACL)뿐 아니라 OS 및 애플리케이션 구성(예: Windows 또는 애플리케이션 패치 적용, 소프트웨어 업그레이드 또는 구성 파일 업데이트)에 대한 변경 사항을 승인하고 통합하는 변경 관리 프로세스를 설정합니다.

AWS는 AWS CloudTrail, AWS Config, AWS CloudFormation, AWS Elastic Beanstalk, AWS OpsWorks 등 의 AWS 리소스 변경 사항 관리를 위한 여러 가지 도구와 시스템 센터 운영 관리자 및 시스템 센터 가상 컴퓨터 관리자를 위한 관리 팩을 제공합니다. Microsoft는 매주 화요일(때로는 매일) Windows 패치를 릴리스하고, AWS는 Microsoft가 패치를 릴리스한 후 5일 안에 AWS가 관리하는 모든 Windows AMI를 업데이트합니다. 따라서 모든 기본 AMI를 지속적으로 패치하고, AWS CloudFormation 템플릿과 Auto Scaling 그룹 구성을 최신 AMI ID로 업데이트하고, 실행 중인 인스턴스 패치 관리를 자동화하는 도구를 구현해야 합니다.

Microsoft는 Windows OS 및 애플리케이션 변경 사항을 관리하기 위한 몇 가지 옵션을 제공합니다. 예를 들어, SCCM은 수명 주기 범위 전체에 걸쳐 환경을 수정할 수 있습니다. 비즈니스 요구 사항을 해결하고 변경 사항이 애플리케이션 SLA, 용량, 보안 및 재해 복구 절차에 미치는 영향을 제어하는 도구를 선택하십시오. 수동 변경을 방지하고, 그 대신 자동화된 구성 관리 소프트웨어나 EC2 Run Command 또는 Windows PowerShell과 같은 명령줄 도구를 활용하여 반복 가능한 스크립트형 변경 프로세스를 구현하십시오. 이러한 요구 사항을 지원하려면 Windows 인스턴스와의 모든 상호 작용에 향상된 로그 기능의 접속 호스트를 사용하여 모든 이벤트와 작업이 자동으로 기록되도록 하십시오.

Amazon EC2의 규정 준수 확인

타사 감사자는 여러 AWS 규정 준수 프로그램의 일환으로 Amazon EC2의 보안 및 규정 준수를 평가합니다. 여기에는 SOC, PCI, FedRAMP, HIPAA 등이 포함됩니다.

특정 규정 준수 프로그램의 범위 내에 있는 AWS 서비스 목록은 [규정 준수 프로그램 제공 범위 내 AWS 서비스](#)를 참조하십시오. 일반적인 내용은 [AWS 규정 준수 프로그램](#)을 참조하십시오.

AWS Artifact를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 [AWS Artifact에서 보고서 다운로드](#)를 참조하십시오.

Amazon EC2는 보안 기술 구현 가이드(STIG)의 규정 준수 표준을 충족하는데 도움이 되는 Microsoft Windows Server용 Amazon Machine Images(AMI)를 제공합니다. 이러한 AMI는 STIG 규정 준수 표준을 충족하면서 배포를 시작할 수 있도록 다수의 STIG 표준으로 미리 구성됩니다. 자세한 내용은 [STIG 규정을 준수하는 Amazon EC2 Windows Server AMI \(p. 65\)](#) 단원을 참조하십시오.

Amazon EC2 사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표, 적용되는 법률과 규정에 따라 결정됩니다. AWS에서는 규정 준수를 지원하기 위해 다음과 같은 리소스를 제공합니다.

- **보안 및 규정 준수 빠른 시작 안내서** – 이 배포 가이드에서는 아키텍처 고려 사항에 대해 설명하고 보안 및 규정 준수에 중점을 둔 기본 AWS 환경을 배포하기 위한 단계를 제공합니다.
- **HIPAA 보안 및 규정 준수 기술 백서 설계** – 이 백서는 기업에서 AWS를 사용하여 HIPAA를 준수하는 애플리케이션을 만드는 방법을 설명합니다.
- **AWS 규정 준수 리소스** – 이 워크북 및 안내서 모음은 귀사가 속한 업계 및 위치에 적용될 수 있습니다.
- AWS Config Developer Guide의 [규칙을 사용하여 리소스 평가](#) – AWS Config는 리소스 구성이 내부 사례, 업계 지침, 규정을 얼마나 잘 준수하는지 평가합니다.
- **AWS Security Hub** – 이 AWS 제품으로 보안 업계 표준 및 모범 사례 규정 준수 여부를 확인하는 데 도움이 되는 AWS 내 보안 상태에 대한 포괄적인 관점을 제공합니다.

Amazon EC2의 감사 및 책임

AWS CloudTrail, AWS Config, AWS Config 규칙은 AWS 리소스 변경 감사를 위한 감사 및 변경 추적 기능을 제공합니다. 로컬 로그 파일을 중앙 로그 관리 시스템으로 전송하도록 Windows 이벤트 로그를 구성하여 보안 및 운영 동작 분석을 위해 로그 데이터를 보존합니다. Microsoft SCOM(시스템 센터 운영 관리자)은 Windows 인스턴스에 배포된 Microsoft 애플리케이션에 대한 정보를 집계하고 애플리케이션 역할 및 서비스를 기반으로 사전 구성된 규칙 및 사용자 지정 규칙 세트를 적용합니다. 시스템 센터 관리 팩은 SCOM을 기반으로 애플리케이션별 모니터링 및 구성 지침을 제공합니다. 이러한 [관리 팩](#)은 Windows Server Active Directory, SharePoint Server 2013, Exchange Server 2013, Lync Server 2013, SQL Server 2014 등 다양한 서버와 기술을 지원합니다. Microsoft SCOM(System Center Operations Manager)용 AWS 관리 팩과 Microsoft SCVMM(System Center Virtual Machine Manager)용 AWS 시스템 관리자는 Microsoft 시스템 센터와 통합되어 온 프레미스 및 AWS 환경을 함께 모니터링하고 관리할 수 있습니다.

고객은 Microsoft 시스템 관리 도구 외에도 Amazon CloudWatch를 사용하여 인스턴스 CPU 사용률, 디스크 성능, 네트워크 I/O를 모니터링하고 호스트 및 인스턴스 상태 확인을 수행할 수 있습니다. EC2Config 및 EC2Launch 서비스를 통해 Windows 인스턴스에 대한 고급 기능을 추가로 이용할 수 있습니다. 예를 들어 Windows 시스템, 보안, 애플리케이션 및 IIS(인터넷 정보 서비스) 로그를 CloudWatch Logs로 내보내고 이를 Amazon CloudWatch 메트릭 및 경보와 통합할 수 있습니다. 고객은 Windows 성능 카운터를 Amazon CloudWatch 사용자 지정 메트릭으로 내보내는 스크립트를 만들 수도 있습니다.

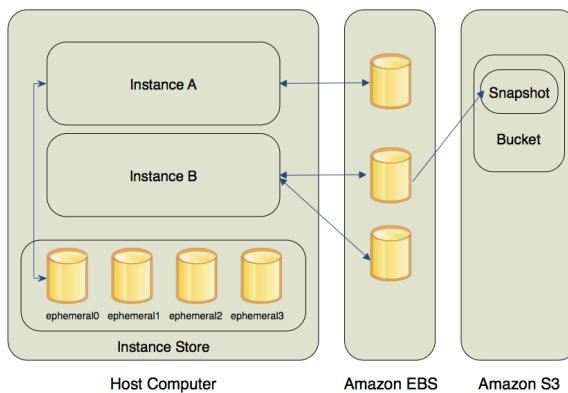
스토리지

Amazon EC2는 고객의 상황에 맞춰 유연하고 비용대비 효율적이며 사용이 쉬운 데이터 스토리지 옵션을 제공합니다. 각 옵션은 성능과 내구성이 조합되어 고유하게 구성됩니다. 이러한 스토리지 옵션은 독립적으로 또는 요구 사항에 맞춰 조합하여 사용할 수 있습니다.

이 섹션을 확인한 후, Amazon EC2가 제공하는 데이터 스토리지 옵션을 활용하여 사용자의 특정 요구 사항을 충족시킬 수 있는 방법에 대해 확실하게 이해할 수 있습니다. 제공되는 스토리지 옵션:

- [Amazon Elastic Block Store \(p. 928\)](#)
- [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#)
- [Amazon EC2와 함께 Amazon S3 사용 \(p. 1102\)](#)

다음 그림은 이러한 스토리지 옵션과 인스턴스 간의 관계를 보여줍니다.



Amazon EBS

Amazon EBS는 내구성이 있는 블록 수준 스토리지 볼륨을 제공하여 실행 중인 인스턴스에 연결하는 것이 가능합니다. 세분화된 업데이트를 자주 수행하는 데이터의 경우 기본 스토리지 디바이스로 Amazon EBS를 사용할 수 있습니다. 예를 들어, Amazon EBS는 인스턴스에서 데이터베이스를 실행할 때 권장되는 스토리지 옵션입니다.

EBS 볼륨은 단일 인스턴스에 연결할 수 있고 형식이 지정되지 않은 외부 블록 원시 디바이스와 같은 방식으로 동작합니다. 볼륨은 인스턴스의 실행 수명과 독립적으로 유지됩니다. 일단 EBS 볼륨이 인스턴스에 연결되면, 다른 물리적 하드 드라이브처럼 사용할 수 있습니다. 이전 그림에서 설명된 것과 같이 여러 볼륨을 단일 인스턴스에 연결할 수 있습니다. 한 인스턴스에서 EBS 볼륨을 분리한 다음 다른 인스턴스에 연결하는 것도 가능합니다. 인스턴스에 연결된 볼륨의 구성을 동적으로 변경할 수 있습니다. 또한 Amazon EBS 암호화 기능을 사용하여 EBS 볼륨을 암호화된 볼륨으로 생성할 수도 있습니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

EBS 볼륨의 스냅샷을 생성하여 Amazon S3에 저장하면 데이터의 백업 사본을 유지할 수 있습니다. 스냅샷에서 새 EBS 볼륨을 만든 후 다른 인스턴스에 연결할 수 있습니다. 자세한 내용은 [Amazon Elastic Block Store \(p. 928\)](#) 단원을 참조하십시오.

Amazon EC2 인스턴스 스토어

여러 인스턴스는 호스트 컴퓨터에 물리적으로 연결된 디스크의 스토리지에 액세스할 수 있습니다. 이러한 디스크 스토리지를 인스턴스 스토어라고 합니다. 인스턴스 스토어는 인스턴스에 블록 수준의 임시 스토리지를 제공합니다. 인스턴스 스토어에 저장된 데이터는 연관 인스턴스의 수명 기간 동안에만 유지되고, 해당 인스

터스를 중지하거나 종료하면 인스턴스 스토어 볼륨의 데이터가 손실됩니다. 자세한 내용은 [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#) 단원을 참조하십시오.

Amazon S3

Amazon S3를 활용하면 저렴하지만 신뢰성이 있는 데이터 스토리지 인프라에 액세스할 수 있습니다. S3은 언제든지 Amazon EC2 내 또는 웹의 어디서나 원하는 데이터의 양을 저장하고 가져올 수 있게 해주어 웹 규모의 컴퓨팅 작업을 쉽게 수행할 수 있도록 설계되었습니다. 예를 들어, Amazon S3를 활용하면 데이터 및 애플리케이션의 백업 사본을 저장할 수 있습니다. Amazon EC2는 Amazon S3를 사용하여 EBS 스냅샷과 인스턴스 스토어 지원 AMI를 저장합니다. 자세한 내용은 [Amazon EC2와 함께 Amazon S3 사용 \(p. 1102\)](#) 단원을 참조하십시오.

스토리지 추가

AMI에서 인스턴스를 실행할 때마다 해당 인스턴스에 대한 루트 스토리지 디바이스가 생성됩니다. 루트 스토리지 디바이스에는 인스턴스를 부팅하기 위해 필요한 모든 정보가 포함됩니다. 블록 디바이스 매핑을 사용하면 AMI를 생성하거나 인스턴스를 실행할 때 루트 디바이스 볼륨과 스토리지 볼륨을 지정할 수 있습니다. 자세한 내용은 [블록 디바이스 매핑 \(p. 1107\)](#) 단원을 참조하십시오.

실행 중인 인스턴스에 EBS 볼륨을 연결할 수도 있습니다. 자세한 내용은 [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#) 단원을 참조하십시오.

스토리지 요금

스토리지 요금에 대한 자세한 내용을 보려면 [AWS 요금](#)을 열고 서비스 요금 아래로 스크롤하여 스토리지를 선택한 다음 스토리지 옵션을 선택하여 해당 스토리지 옵션의 요금 페이지를 여십시오. 스토리지 비용 추정에 대한 자세한 내용은 [AWS Pricing Calculator](#)를 참조하십시오.

Amazon Elastic Block Store (Amazon EBS)

Amazon Elastic Block Store(Amazon EBS)는 EC2 인스턴스에 사용할 수 있는 블록 수준 스토리지 볼륨을 제공합니다. EBS 볼륨은 형식이 지정되지 않은 원시 블록 디바이스처럼 동작합니다. 이러한 볼륨을 인스턴스에 디바이스로 마운트할 수 있습니다. 인스턴스에 연결된 EBS 볼륨은 스토리지 볼륨으로 표시되며, 인스턴스 수명에 관계없이 지속됩니다. 이러한 볼륨 위에 파일 시스템을 생성하거나 하드 드라이브와 같은 블록 디바이스를 사용하는 것처럼 볼륨을 사용할 수 있습니다. 인스턴스에 연결된 볼륨의 구성을 동적으로 변경할 수 있습니다.

데이터에 빠르게 액세스하고 장기적으로 지속해야 하는 경우 Amazon EBS를 사용하는 것이 좋습니다. EBS 볼륨은 세분화된 업데이트가 필요하고 형식이 지정되지 않은 블록 수준의 원시 스토리지에 액세스해야 하는 파일 시스템, 데이터베이스 또는 애플리케이션의 기본 스토리지로 사용하기에 특히 적합합니다. Amazon EBS는 입의 읽기 및 쓰기에 의존하는 데이터베이스 스타일의 애플리케이션과 장시간의 지속적인 읽기 및 쓰기를 수행하여 처리량이 큰 애플리케이션에 모두 적합합니다.

Amazon EBS에서는 사용한 만큼만 지불하면 됩니다. Amazon Elastic Block Store 요금에 대한 자세한 정보는 [Amazon EBS 페이지](#)의 비용 예측 단원을 참조하십시오.

내용

- [Amazon EBS의 기능 \(p. 929\)](#)
- [Amazon EBS 볼륨 \(p. 929\)](#)
- [Amazon EBS 스냅샷 \(p. 966\)](#)
- [Amazon EBS 데이터 서비스 \(p. 1023\)](#)
- [Windows 인스턴스의 Amazon EBS 및 NVMe \(p. 1048\)](#)
- [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#)
- [Amazon EBS 인스턴스의 Windows 볼륨 성능 \(p. 1063\)](#)

- [Amazon EBS에 대한 Amazon CloudWatch 지표 \(p. 1076\)](#)
- [Amazon EBS용 Amazon CloudWatch Events \(p. 1081\)](#)

Amazon EBS의 기능

- EBS 볼륨은 특정 가용 영역에서 생성한 후 동일한 가용 영역에 있는 아무 인스턴스에나 연결할 수 있습니다. 가용 영역 외부에 볼륨을 제공하기 위해 스냅샷을 생성하고 해당 지역 어디서나 새 볼륨으로 복원할 수 있습니다. 스냅샷을 다른 지역에 복사한 다음 새 볼륨에 복원하면 지리적 확장, 데이터 센터 마이그레이션 및 재해 복구를 위해 여러 AWS 지역을 쉽게 활용할 수 있습니다.
- Amazon EBS는 범용 SSD(gp2), 프로비저닝된 IOPS SSD(io1 및 io2), 처리량에 최적화된 HDD(st1) 및 Cold HDD(sc1) 볼륨 유형을 제공합니다. 다음은 각 볼륨 유형에 대한 성능 및 사용 사례의 요약입니다.
 - 범용 SSD 볼륨에서는 3 IOPS/GiB를 기본 성능으로 제공하며, 시간을 연장할 경우 최대 3,000 IOPS까지 버스트할 수 있습니다. 이러한 볼륨은 부트 볼륨, 중소 규모 데이터베이스, 개발 및 테스트 환경 등의 광범위한 사용 사례에 적합합니다. 자세한 내용은 [범용 SSD\(gp2\) 볼륨 \(p. 934\)](#) 단원을 참조하십시오.
 - 프로비저닝된 IOPS SSD 볼륨은 최대 64,000 IOPS 및 1,000MiB/s의 처리량을 지원합니다. 따라서 예측 가능한 방식으로 EC2 인스턴스당 수만 IOPS까지 확장할 수 있습니다. 자세한 정보는 [프로비저닝된 IOPS SSD\(io1 및 io2\) 볼륨 \(p. 937\)](#) 단원을 참조하십시오.
 - 처리량에 최적화된 HDD 볼륨은 IOPS가 아닌 처리량으로 성능을 정의하는 저비용 마그네틱 스토리지를 제공합니다. 이러한 볼륨은 Amazon EMR, ETL, 데이터 웨어하우스, 로그 처리 같은 대용량 순차 워크로드에 적합합니다. 자세한 내용은 [처리량에 최적화된 HDD\(st1\) 볼륨 \(p. 938\)](#) 단원을 참조하십시오.
 - Cold HDD 볼륨은 IOPS가 아닌 처리량으로 성능을 정의하는 저비용 마그네틱 스토리지를 제공합니다. 이 볼륨은 순차적인 대용량 콜드 데이터 워크로드에 적합합니다. 데이터에 자주 액세스할 필요가 없고 비용을 절약해야 한다면 저렴한 블록 스토리지로 이러한 볼륨이 적합합니다. 자세한 내용은 [Cold HDD\(sc1\) 볼륨 \(p. 941\)](#) 단원을 참조하십시오.
- 암호화된 볼륨으로 EBS 볼륨을 생성하여 규제/감사 데이터 및 애플리케이션에 대한 다양한 저장 시 데이터 암호화 요구 사항을 충족할 수 있습니다. 암호화된 EBS 볼륨을 생성하여 지원되는 인스턴스 유형에 연결하면 볼륨에 저장된 데이터, 디스크 I/O 및 볼륨에서 생성된 스냅샷이 모두 암호화됩니다. 암호화는 EC2 인스턴스를 호스팅하는 서버에서 수행되므로 EC2 인스턴스에서 EBS 스토리지로 전송되는 데이터가 암호화됩니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.
- Amazon S3까지 지속되는 EBS 볼륨의 지정 시간 스냅샷을 생성할 수 있습니다. 이러한 스냅샷은 데이터를 장기적으로 안전하게 보호하며 새로운 EBS 볼륨의 시작점으로 사용할 수도 있습니다. 또한, 스냅샷을 사용하여 원하는 수만큼 볼륨을 인스턴스화할 수 있습니다. 이러한 스냅샷을 AWS 지역에서 복사할 수 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 \(p. 966\)](#) 단원을 참조하십시오.
- 대역폭, 처리량, 지연 시간, 평균 대기열 길이 등의 성능 지표가 AWS Management 콘솔을 통해 제공됩니다. Amazon CloudWatch에 의해 제공되는 이러한 지표를 통해 볼륨의 성능을 모니터링하면 필요 없는 리소스를 구입하지 않고도 애플리케이션에 충분한 성능을 제공할 수 있습니다. 자세한 내용은 [Amazon EBS 인스턴스의 Windows 볼륨 성능 \(p. 1063\)](#) 단원을 참조하십시오.

Amazon EBS 볼륨

Amazon EBS 볼륨은 내구성이 있는 블록 수준 스토리지 디바이스이며 인스턴스를 연결하는 것이 가능합니다. 볼륨을 인스턴스에 연결하면 물리적 하드 드라이브처럼 사용할 수 있습니다. EBS 볼륨은 유연합니다. 현재 세대 인스턴스 유형에 연결된 현재 세대 볼륨의 경우 크기를 동적으로 늘리고 프로비저닝된 IOPS 용량을 수정하며 라이브 프로덕션 볼륨의 볼륨 유형을 변경할 수 있습니다.

인스턴스의 시스템 드라이브 또는 데이터베이스 애플리케이션용 스토리지 등 자주 업데이트해야 하는 데이터의 경우 EBS 볼륨을 기본 스토리지로 사용할 수 있습니다. 연속으로 디스크 스캔을 수행하는 처리량 집약적 애플리케이션에도 해당 볼륨을 사용할 수 있습니다. EBS 볼륨은 EC2 인스턴스의 실행 주기와는 독립적으로 유지됩니다.

여러 EBS 볼륨을 단일 인스턴스에 연결할 수 있습니다. 볼륨 및 인스턴스는 동일 가용 영역에 위치해야 합니다.

Amazon EBS는 범용 SSD(gp2), 프로비저닝된 IOPS SSD(io1 및 io2), 처리량에 최적화된 HDD(st1), Cold HDD(sc1) 및 Magnetic(standard, 이전 세대 유형) 같은 볼륨 유형을 제공합니다. 이 두 유형은 성능 특성과 가격이 다르므로 애플리케이션의 필요에 맞게 스토리지 성능과 비용을 조정할 수 있습니다. 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.

계정당 사용할 수 있는 EBS 볼륨 수와 사용 가능한 총 스토리지는 제한됩니다. 이러한 제한값 및 제한값 증가 요청 방법에 대한 자세한 내용은 [Amazon EC2 서비스 할당량 \(p. 1154\)](#) 단원을 참조하십시오.

요금에 대한 자세한 내용은 [Amazon EBS 요금](#)을 참조하십시오.

목차

- [EBS 볼륨 사용의 이점 \(p. 930\)](#)
- [Amazon EBS 볼륨 유형 \(p. 932\)](#)
- [EBS 볼륨의 크기 및 구성에 대한 제약 \(p. 945\)](#)
- [Amazon EBS 볼륨 생성 \(p. 948\)](#)
- [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#)
- [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#)
- [Amazon EBS 볼륨에 대한 정보 보기 \(p. 955\)](#)
- [이전 스냅샷을 사용하여 Amazon EBS 볼륨 바꾸기 \(p. 957\)](#)
- [볼륨 상태 모니터링 \(p. 957\)](#)
- [Windows 인스턴스에서 Amazon EBS 볼륨 분리 \(p. 964\)](#)
- [Amazon EBS 볼륨 삭제 \(p. 966\)](#)

EBS 볼륨 사용의 이점

EBS 볼륨은 인스턴스 스토어 볼륨과 차별화된 이점을 제공합니다.

데이터 가용성

EBS 볼륨을 생성하면 단일 하드웨어 구성 요소의 장애로 인한 데이터 손실을 방지하기 위해 해당 가용 영역 내에서 자동으로 복제됩니다. 동일한 가용 영역에 있는 EC2 인스턴스에 EBS 볼륨을 연결할 수 있습니다. 볼륨을 연결한 후에 인스턴스는 하드 드라이브 또는 기타 물리 드라이브와 같은 원시 블록 디바이스처럼 보입니다. 이 시점에 인스턴스는 로컬 드라이브와 동일한 방식으로 볼륨과 상호 작용할 수 있습니다. 인스턴스에 연결하고 NTFS 등의 파일 시스템으로 EBS 볼륨을 포맷한 다음 애플리케이션을 설치할 수 있습니다.

사용자가 명명한 디바이스에 다중 볼륨이 연결된 경우 사용자는 I/O 및 처리 성능을 향상하기 위해 전체 볼륨에서 데이터를 스트라이프할 수 있습니다.

추가 비용 없이 EBS 볼륨 및 EBS 기반 인스턴스의 루트 디바이스 볼륨의 데이터를 모니터링할 수 있습니다. 측정치의 모니터링에 대한 자세한 내용은 [Amazon EBS에 대한 Amazon CloudWatch 지표 \(p. 1076\)](#) 단원을 참조하십시오. 볼륨 상태 추적에 대한 자세한 내용은 [Amazon EBS용 Amazon CloudWatch Events \(p. 1081\)](#) 단원을 참조하십시오.

데이터 지속성

EBS 볼륨은 인스턴스의 수명에 관계없이 유지되는 오프 인스턴스 스토리지입니다. 사용자는 데이터가 유지되는 동안 볼륨 사용량에 대한 비용을 계속해서 지불합니다.

실행 중인 인스턴스에 연결된 EBS 볼륨은 사용자가 EC2 콘솔에서 해당 인스턴스에 대한 EBS 볼륨을 구성할 때 종료 시 삭제 확인란을 선택하지 않은 경우 인스턴스가 종료될 때 해당 데이터가 원래 상태를 유지한 채로 해당 인스턴스에서 자동으로 분리될 수 있습니다. 그러면 해당 볼륨은 새 인스턴스로 재연결되어 빠른 복구가 가능합니다. 종료 시 삭제 확인란이 선택된 경우에는 EC2 인스턴스 종료 시 볼륨이 삭제됩니다. EBS

기반 인스턴스를 사용하는 경우 연결된 볼륨에 저장된 데이터에 영향을 주지 않고 해당 인스턴스를 중지하고 다시 시작할 수 있습니다. 해당 볼륨은 정지-시작 주기 동안 연결 상태를 유지합니다. 이를 통해 사용자는 필요할 때 처리 및 스토리지 리소스만을 사용하여 볼륨에서 데이터를 무기한으로 처리 및 저장할 수 있습니다. 데이터는 볼륨이 완전히 삭제될 때까지 볼륨에서 유지됩니다. 삭제된 EBS 볼륨에서 사용된 물리 볼륨 스토리지는 다른 계정에 할당되기 전까지 0으로 덮어쓰기됩니다. 민감한 데이터를 사용하는 경우 데이터를 직접 암호화하거나 Amazon EBS 암호화으로 보호되는 볼륨에 데이터를 저장해야 합니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

기본적으로, 실행 시 생성되어 인스턴스에 연결된 루트 EBS 볼륨은 해당 인스턴스가 종료되면 삭제됩니다. 사용자는 인스턴스 시작 시 플래그 값을 `DeleteOnTermination`에서 `false`로 변경하여 해당 동작을 수정할 수 있습니다. 값이 수정되면 인스턴스가 종료된 후에도 볼륨이 유지되어 해당 볼륨에 다른 인스턴스를 연결할 수 있습니다.

기본적으로, 실행 시 생성되어 인스턴스에 연결된 추가 EBS 볼륨은 해당 인스턴스가 종료되면 삭제되지 않습니다. 사용자는 인스턴스 시작 시 플래그 값을 `DeleteOnTermination`에서 `true`로 변경하여 해당 동작을 수정할 수 있습니다. 이 수정된 값으로 인해 인스턴스가 종료될 때 볼륨이 삭제됩니다.

데이터 암호화

단순 데이터 암호화의 경우 Amazon EBS 암호화 기능으로 암호화된 EBS 볼륨을 생성할 수 있습니다. 모든 EBS 볼륨 유형은 암호화를 지원합니다. 암호화된 EBS 볼륨을 사용하여 규제/감사 데이터 및 애플리케이션에 대한 다양한 유형 데이터 암호화 요구 사항을 충족할 수 있습니다. Amazon EBS 암호화는 256비트 고급 암호화 표준 알고리즘(AES-256) 및 Amazon이 관리하는 키 인프라를 사용합니다. 암호화는 EC2 인스턴스를 호스트하는 서버에서 수행되므로 EC2 인스턴스에서 Amazon EBS 스토리지로 전송되는 데이터가 암호화됩니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

Amazon EBS 암호화는 AWS Key Management Service(AWS KMS) 마스터 키를 사용하여 암호화된 볼륨을 생성하고 암호화된 볼륨에서 모든 스냅샷을 생성합니다. 리전에서 암호화된 EBS 볼륨을 처음 생성할 때 기본 마스터 키가 자동으로 생성됩니다. Amazon EBS 암호화를 사용하여 별도로 생성된 CMK(고객 마스터 키)를 선택하는 경우를 제외하고 이 키가 AWS KMS에 사용됩니다. CMK를 직접 생성하면 액세스 제어를 생성, 교체, 비활성화, 정의하고 데이터를 보호하는 데 사용된 암호화 키를 감사하는 등 보다 폭넓은 작업이 가능합니다. 자세한 내용은 [AWS Key Management Service Developer Guide](#) 단원을 참조하십시오.

스냅샷

Amazon EBS를 사용하면 모든 EBS 볼륨의 스냅샷(백업)을 생성하고 볼륨 내 데이터 사본을 다중 가용 영역에 중복 저장이 가능한 Amazon S3에 작성할 수 있습니다. 볼륨이 실행 중인 인스턴스에 연결되어 있지 않아도 스냅샷을 만드는 데는 문제가 없습니다. 볼륨에 데이터를 계속해서 작성하면 새 볼륨의 기준으로 사용될 볼륨 스냅샷을 주기적으로 생성할 수 있습니다. 이 스냅샷을 사용하여 새로운 EBS 볼륨을 여러 개 생성하거나 가용 영역 간에 볼륨을 이동할 수 있습니다. 암호화된 EBS 볼륨의 스냅샷은 자동으로 암호화됩니다.

스냅샷에서 새로운 볼륨을 생성하는 경우 새로 생성된 스냅샷이 생성될 시점의 원본 볼륨 사본과 정확히 일치합니다. 암호화된 스냅샷에서 생성된 EBS 볼륨은 자동으로 암호화됩니다. 다양한 가용 영역을 지정하는 옵션이 있습니다. 이러한 기능을 사용하여 이 영역에 복제 볼륨을 생성할 수 있습니다. 스냅샷은 특정 AWS 계정과 공유하거나 퍼블릭으로 설정할 수 있습니다. 스냅샷을 생성하는 경우 볼륨의 총 크기에 따라 Amazon S3에서 비용이 발생하게 됩니다. 볼륨 스냅샷을 연속으로 생성하면 볼륨의 원본 크기와 대비하여 추가된 데이터에 해당하는 비용만이 청구됩니다.

;스냅샷은 마지막 스냅샷 이후 변경된 볼륨의 블록만 저장되는 충분식 백업입니다. 100GiB 데이터를 가진 볼륨이 있지만 마지막 스냅샷 이후 5GiB만이 변경된 경우 변경된 5GiB만이 Amazon S3에 작성됩니다. 스냅샷은 충분식으로 저장되지만, 스냅샷 삭제 프로세스는 가장 최근의 스냅샷만 유지하도록 설계되어 있습니다.

볼륨 및 스냅샷을 쉽게 범주화하고 관리할 수 있도록 사용자는 원하는 메타데이터로 볼륨 및 스냅샷에 태그를 사용할 수 있습니다. 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

볼륨을 자동으로 백업하려는 경우 [Amazon 데이터 수명 주기 관리자 \(p. 1012\)](#) 또는 [AWS Backup](#)를 사용할 수 있습니다.

유연성

EBS 볼륨은 프로덕션 중에 라이브 구성 변경을 지원합니다. 서비스 중단 없이 볼륨 유형, 볼륨 크기, IOPS 용량을 수정할 수 있습니다. 자세한 내용은 [Amazon EBS 탄력적 볼륨 \(p. 1023\)](#) 단원을 참조하십시오.

Amazon EBS 볼륨 유형

Amazon EBS는 다음의 볼륨 유형을 제공하고 이러한 볼륨 유형은 성능 특성과 가격이 다르므로 애플리케이션의 필요에 맞게 스토리지 성능과 비용을 조정할 수 있습니다. 볼륨 유형은 다음 두 가지 범주로 나뉩니다.

- SSD 지원 볼륨: 작은 I/O 크기의 읽기/쓰기 작업을 자주 처리하는 트랜잭션 워크로드에 최적화되어 있으며, 기준 성능 측정은 IOPS
- HDD 지원 볼륨: 대용량 스트리밍 워크로드에 최적화되어 있으며, IOPS보다는 처리량(MiB/s로 측정)이 더 정확한 성능 측정 기준

인스턴스 구성, I/O 특성 및 워크로드 요구량 등 여러 가지 요인이 EBS 볼륨의 성능에 영향을 미칠 수 있습니다. EBS 볼륨을 최대한 이용하는 방법에 대한 자세한 내용은 [Amazon EBS 인스턴스의 Windows 볼륨 성능 \(p. 1063\)](#)을 참조하십시오.

요금에 대한 자세한 내용은 [Amazon EBS 요금](#)을 참조하십시오.

볼륨 특성

다음 표는 각 볼륨 유형에 대한 사용 사례 및 성능 특성을 설명합니다. 기본 볼륨 유형은 범용 SSD(gp2)입니다.

볼륨 유형	SSD(Solid-State Drive)			HDD(Hard disk Drive)	
	범용 SSD (gp2)	프로비저닝된 IOPS SSD		처리량에 최적화된 HDD (st1)	Cold HDD (sc1)
설명		io2	io1		
설명	다양한 워크로드에 사용할 수 있으며 가격 대비 성능이 우수한 범용 SSD 볼륨	지연 시간이 짧거나 처리량이 많은 미션 크리티컬 워크로드에 적합한 고성능 SSD 볼륨		자주 액세스하는 처리량 집약적 워크로드에 적합한 저비용 HDD 볼륨	자주 액세스하지 않는 워크로드에 적합한 최저 비용 HDD 볼륨
내구성	99.8%~99.9% 내구성(연간 장애율 0.1%~0.2%)	99.999% 내구성(연간 장애율 0.001%)	99.8%~99.9% 내구성(연간 장애율 0.1%~0.2%)	99.8%~99.9% 내구성(연간 장애율 0.1%~0.2%)	99.8%~99.9% 내구성(연간 장애율 0.1%~0.2%)
사용 사례	<ul style="list-style-type: none">대부분의 워크로드에 추천시스템 부트 볼륨가상 데스크톱지연 시간이 짧은 대화형 앱	<ul style="list-style-type: none">지속적인 IOPS 성능이나 16,000 IOPS 또는 250MiB/s 이상의 볼륨당 처리량을 필요로 하는 중요한 비즈니스 애플리케이션라지 데이터베이스 워크로드. 예:<ul style="list-style-type: none">MongoDBCassandraMicrosoft SQL ServerMySQLPostgreSQL	<ul style="list-style-type: none">저비용으로 일관되고 높은 처리량을 요구하는 스트리밍 워크로드빅 데이터데이터 웨어하우스로그 처리	<ul style="list-style-type: none">자주 액세스하지 않는 대용량 데이터를 위한 처리량 중심의 스토리지스토리지 비용이 최대한 낮아야 하는 시나리오부트 볼륨이 될 수 없음	

	SSD(Solid-State Drive)			HDD(Hard disk Drive)	
	• 개발 및 테스트 환경	• Oracle		• 부트 볼륨이 될 수 없음	
Amazon EBS 다중 연결	지원되지 않음	지원되지 않음	지원	지원되지 않음	지원되지 않음
API 이름	gp2	io2	io1	st1	sc1
볼륨 크기	1GiB - 16TiB	4GiB - 16TiB		500GiB - 16TiB	500GiB - 16TiB
주요 성능 특성	IOPS	IOPS		MiB/s	MiB/s
볼륨당 최대 IOPS	16,000(16KiB I/O)*	64,000(16KiB I/O)†		500(1MiB I/O)	250(1MiB I/O)
볼륨당 최대 처리량	250MiB/s*	1,000MiB/s†		500MiB/s	250MiB/s
인스턴스당 최대 IOPS ††	160,000				
인스턴스당 최대 처리량 ††	4,750MB/s				

* 처리량 한도는 볼륨 크기에 따라 128MiB/s~250 MiB/s입니다. 170GiB 이하의 볼륨은 최대 128MiB/s의 처리량을 제공합니다. 170GiB에서 334GiB까지의 볼륨은 버스트 크레딧을 사용할 수 있는 경우 최대 250MiB/s의 처리량을 제공합니다. 334 GiB 이상의 볼륨은 버스트 크레딧에 상관없이 250MiB/s를 제공합니다. 이전 gp2 볼륨은 볼륨을 수정하지 않는 한 전체 성능에 도달할 수 없습니다. 자세한 내용은 [Amazon EBS 탄력적 볼륨 \(p. 1023\)](#) 단원을 참조하십시오.

† 최대 IOPS 및 처리량은 32,000 IOPS 이상으로 프로비저닝된 [Nitro 시스템에 구축된 인스턴스 \(p. 118\)](#)에서만 보장됩니다. 다른 인스턴스는 최대 32,000 IOPS 및 500 MiB/s의 성능을 보장합니다. 이전 io1 볼륨은 볼륨을 수정하지 않는 한 전체 성능에 도달할 수 없습니다. 자세한 내용은 [Amazon EBS 탄력적 볼륨 \(p. 1023\)](#) 단원을 참조하십시오.

†† 이 처리량을 달성하려면 [EBS 최적화 \(p. 1049\)](#)를 지원하는 인스턴스가 있어야 합니다.

이전 세대 볼륨 유형

다음 표에서는 이전 세대 EBS 볼륨 유형을 설명합니다. 이전 세대 볼륨보다 우수한 성능 또는 성능 일관성이 필요한 경우, 범용 SSD(gp2) 또는 기타 현재 볼륨 유형 사용을 고려할 것을 권장합니다. 자세한 내용은 [이전 세대 볼륨](#)을 참조하십시오.

HDD(Hard disk Drive)	
볼륨 유형	Magnetic
사용 사례	데이터에 자주 액세스하지 않는 워크로드
API 이름	standard
볼륨 크기	1GiB - 1TiB
볼륨당 최대 IOPS	40~200

HDD(Hard disk Drive)	
볼륨당 최대 처리량	40-90MiB/s
인스턴스당 최대 IOPS	80,000
인스턴스당 최대 처리량	1,750MB/s
주요 성능 특성	IOPS

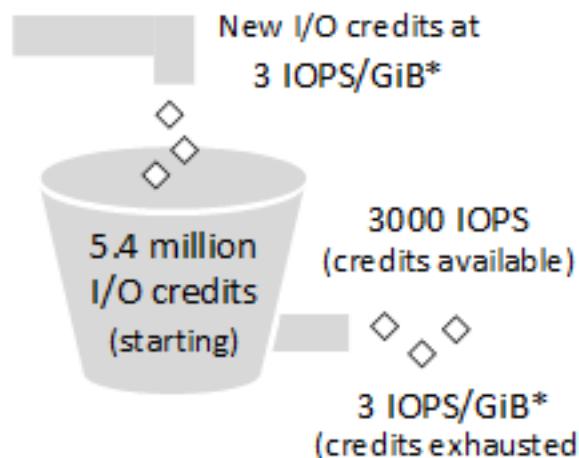
범용 SSD(gp2) 볼륨

범용 SSD(gp2) 볼륨은 광범위한 작업에서 이상적으로 사용될 수 있는 비용 효과적인 스토리지를 제공합니다. 이러한 볼륨은 시간을 연장할 경우 3,000IOPS의 버스트 기능까지 지원되어 지연 시간이 한 자릿수 밀리초에 불과합니다. 최소 100 IOPS(33.33GiB 이하)와 최대 16,000 IOPS(5,334GiB 이상) 사이에서, 기준 성능은 볼륨 크기의 GiB당 3 IOPS로 일정하게 확장됩니다. AWS는 프로비저닝된 성능의 99%의 시간에 제공하기 위해 gp2 볼륨을 설계합니다. gp2 볼륨 크기는 1GiB~16TiB입니다.

I/O 크레딧 및 버스트 성능

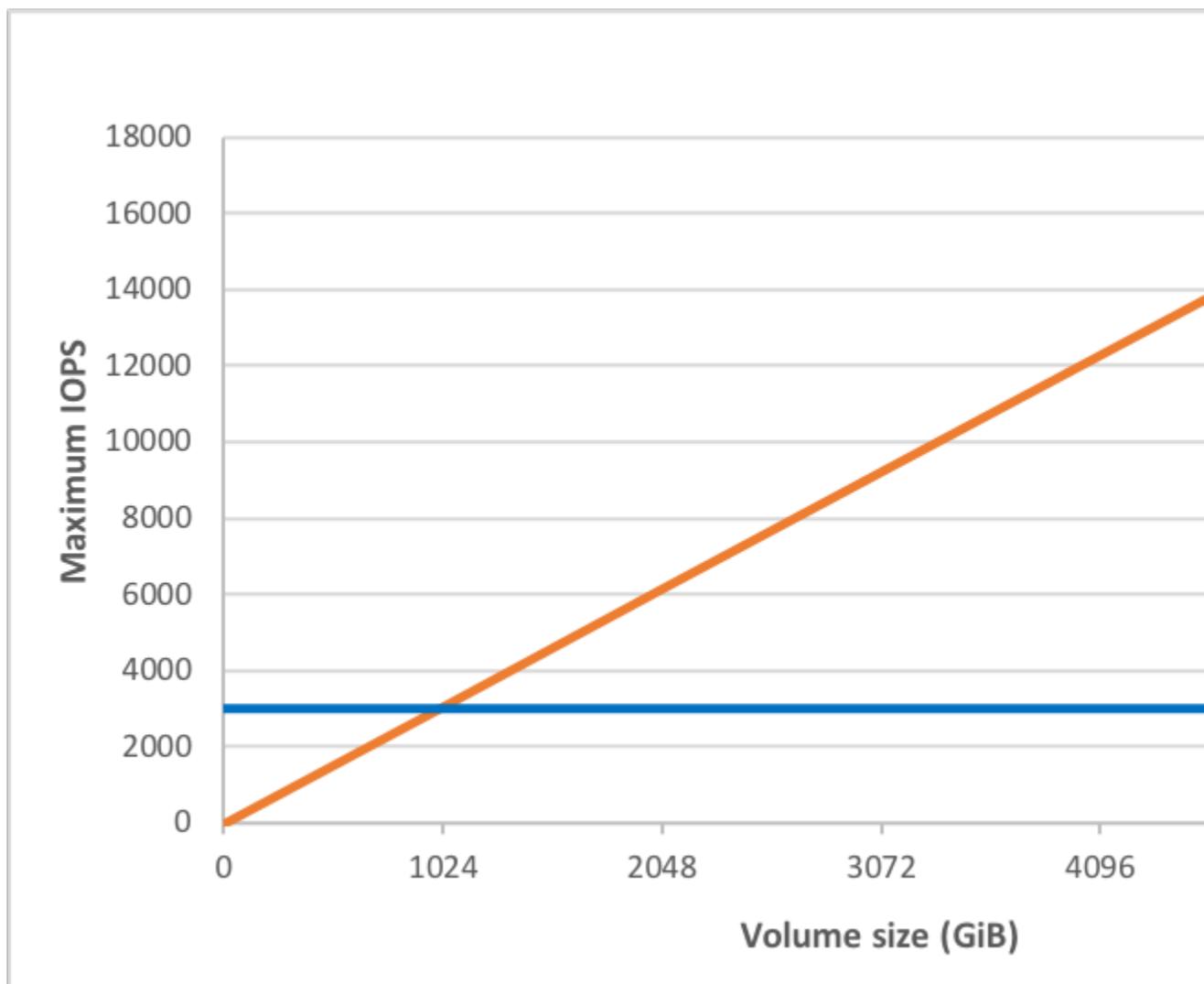
gp2 볼륨의 성능은 볼륨 크기에 따라 정해지고 볼륨의 기준 성능 수준 및 I/O 크레딧이 얼마나 빨리 누적되는지를 결정합니다. 볼륨이 클수록 기준 성능 수준이 크고 I/O 크레딧이 빨리 누적됩니다. I/O 크레딧이란 기준 성능 이상이 필요한 경우 대규모 I/O를 버스트하도록 gp2 볼륨이 사용할 수 있는 가용 대역폭입니다. 볼륨에 I/O 크레딧이 많을수록 더 오랜 기간 동안 볼륨이 기준 성능 수준 이상을 버스트할 수 있고 더 큰 성능이 필요할 때 더 좋은 성능을 발휘할 수 있습니다. 다음 다이어그램은 gp2의 버스트 버킷 동작을 보여줍니다.

GP2 burst bucket



* Scaling linearly between minimum 100 IOPS and maximum 16,000 IOPS

각 볼륨의 초기 I/O 크레딧 밸런스는 540만 I/O 크레딧이고 이것은 30분 동안 3,000 IOPS의 최대 버스트 성능을 유지할 수 있는 수준입니다. 이러한 초기 크레딧 밸런스는 부트 볼륨에 빠른 초기 부팅 주기를 제공하고 기타 애플리케이션에 좋은 부트스트래핑 환경을 제공하도록 설계되었습니다. 볼륨은 볼륨 크기의 GiB 당 3 IOPS의 기준 성능 비율로 I/O 크레딧을 획득합니다. 예를 들어, 100 GiB gp2 볼륨은 300 IOPS의 기준 성능을 갖습니다.



볼륨에 기준 성능 I/O 수준 이상이 필요한 경우 크레딧 밸런스에서 I/O 크레딧을 사용하여 최대 3,000 IOPS 까지 필요한 성능 수준을 버스트할 수 있습니다. 볼륨이 초당 획득한 I/O 크레딧 이하를 사용하는 경우 미사용 I/O 크레딧은 I/O 크레딧 밸런스에 가산됩니다. 볼륨의 최대 I/O 크레딧은 초기 크레딧 밸런스(540만 I/O 크레딧)와 동일합니다.

볼륨의 기준 성능이 최대 버스트 성능보다 높으면 I/O 크레딧이 사용되지 않습니다. 볼륨이 [Nitro 시스템](#) (p. 118) 기반 인스턴스에 연결된 경우 버스트 밸런스가 보고되지 않습니다. 다른 인스턴스의 경우 보고되는 버스트 밸런스는 100%입니다.

볼륨의 버스트 구간은 볼륨의 크기, 필요한 버스트 IOPS 및 버스트가 시작되는 크레딧 밸런스에 의해 결정됩니다. 방법은 다음 수식과 같습니다.

$$\text{Burst duration} = \frac{(\text{Credit balance})}{(\text{Burst IOPS}) - 3(\text{Volume size in GiB})}$$

다음 표는 여러 볼륨 크기 및 관련 볼륨 기준 성능(I/O 크레딧 누적 비율), 최대 3,000 IOPS에서의 버스트 기간(전체 크레딧 밸런스에서 시작 시) 및 볼륨이 빈 크레딧 밸런스를 다시 채우는 데 걸리는 초 단위 시간을 보여줍니다.

볼륨 크기(GiB)	기준 성능(IOPS)	지속적으로 3,000 IOPS(초) 구동 시 버스트 지속 시간	IO 구동이 없을 시 빈 크레딧 밸런스를 채우는데 소요되는 시간(초)
1	100	1,802	54,000
100	300	2,000	18,000
250	750	2,400	7,200
334(최대 처리량에 대한 최소 크기)	1,002	2,703	5,389
500	1,500	3,600	3,600
750	2,250	7,200	2,400
1,000	3,000	해당 사항 없음*	해당 사항 없음*
5,334(최대 IOPS에 대한 최소 크기)	16,000	해당 사항 없음*	해당 사항 없음*
16,384(16TiB, 최대 볼륨 크기)	16,000	해당 사항 없음*	해당 사항 없음*

* 볼륨의 기준 성능이 최대 버스트 성능을 초과합니다.

I/O 크레딧 밸런스가 0이 되면 어떻게 되나요?

gp2 볼륨에서 I/O 크레딧 잔고 전부가 소진되면 볼륨의 최대 IOPS 성능이 기준 IOPS 성능 수준(볼륨이 크레딧을 획득하는 속도)으로 유지되고 볼륨의 최대 처리량은 최대 I/O 크기를 곱한 기준 IOPS로 줄어듭니다. 처리량은 250MiB/s를 초과할 수 있습니다. I/O 요구가 기준 성능 수준 이하로 떨어지고 미사용 크레딧이 I/O 크레딧 밸런스에 추가되면, 볼륨의 최대 IOPS 성능이 다시 기준 수준을 초과합니다. 예를 들어, 크레딧 잔고가 0인 100GiB gp2 볼륨은 300 IOPS의 기준 성능 및 75MiB/s의 처리 한도를 갖습니다(초당 300I/O 작업 * I/O 작업당 256KiB = 75MiB/s). 볼륨이 커지면 기준 성능도 높아지고 크레딧 밸런스가 더 빨리 보충됩니다. IPOS 측정 방법에 대한 자세한 내용은 [I/O 특성 및 모니터링 \(p. 1064\)](#) 단원을 참조하십시오.

볼륨 성능이 자주 기준 수준 한도에 도달하는 경우(빈 I/O 크레딧 잔고로 인해) 더 큰 gp2 볼륨(기준 성능 수준이 향상)을 사용하거나, IOPS 성능을 16,000 IOPS 이상으로 유지해야 하는 워크로드인 경우 io1 또는 io2 볼륨으로 전환하는 것을 고려해야 합니다.

CloudWatch 지표 및 경보를 사용하여 버스트 버킷 잔고를 모니터링하는 방법은 [gp2, st1, sc1 볼륨에 대한 버스트 버킷 밸런스 모니터링 \(p. 945\)](#) 단원을 참조하십시오.

처리량 성능

gp2 볼륨의 처리량은 다음 공식을 사용하여 계산할 수 있습니다. 최대 처리량 한도는 250MiB/s입니다.

$$\text{Throughput in MiB/s} = ((\text{Volume size in GiB}) \times (\text{IOPS per GiB}) \times (\text{I/O size in KiB}))$$

V = 볼륨 크기, I = I/O 크기, R = I/O 속도, T = 처리량이라고 가정하면 이 공식을 다음과 같이 축약할 수 있습니다.

$$T = V \times I \times R$$

최대 처리량을 달성하는 최소 볼륨 크기는 다음과 같이 계산됩니다.

```


$$V = \frac{T}{I R}$$


$$= \frac{250 \text{ MiB/s}}{(256 \text{ KiB})(3 \text{ IOPS/GiB})}$$


$$= \frac{[(250)(2^{20})(\text{Bytes})]/\text{s}}{(256)(2^{10})(\text{Bytes})([3 \text{ IOP/s}]/[(2^{30})(\text{Bytes})])}$$


$$= \frac{(250)(2^{20})(2^{30})(\text{Bytes})}{(256)(2^{10})(3)}$$


$$= 357,913,941,333 \text{ Bytes}$$


$$= 333\# \text{ GiB (334 GiB in practice because volumes are provisioned in whole gibibytes)}$$


```

프로비저닝된 IOPS SSD(io1 및 io2) 볼륨

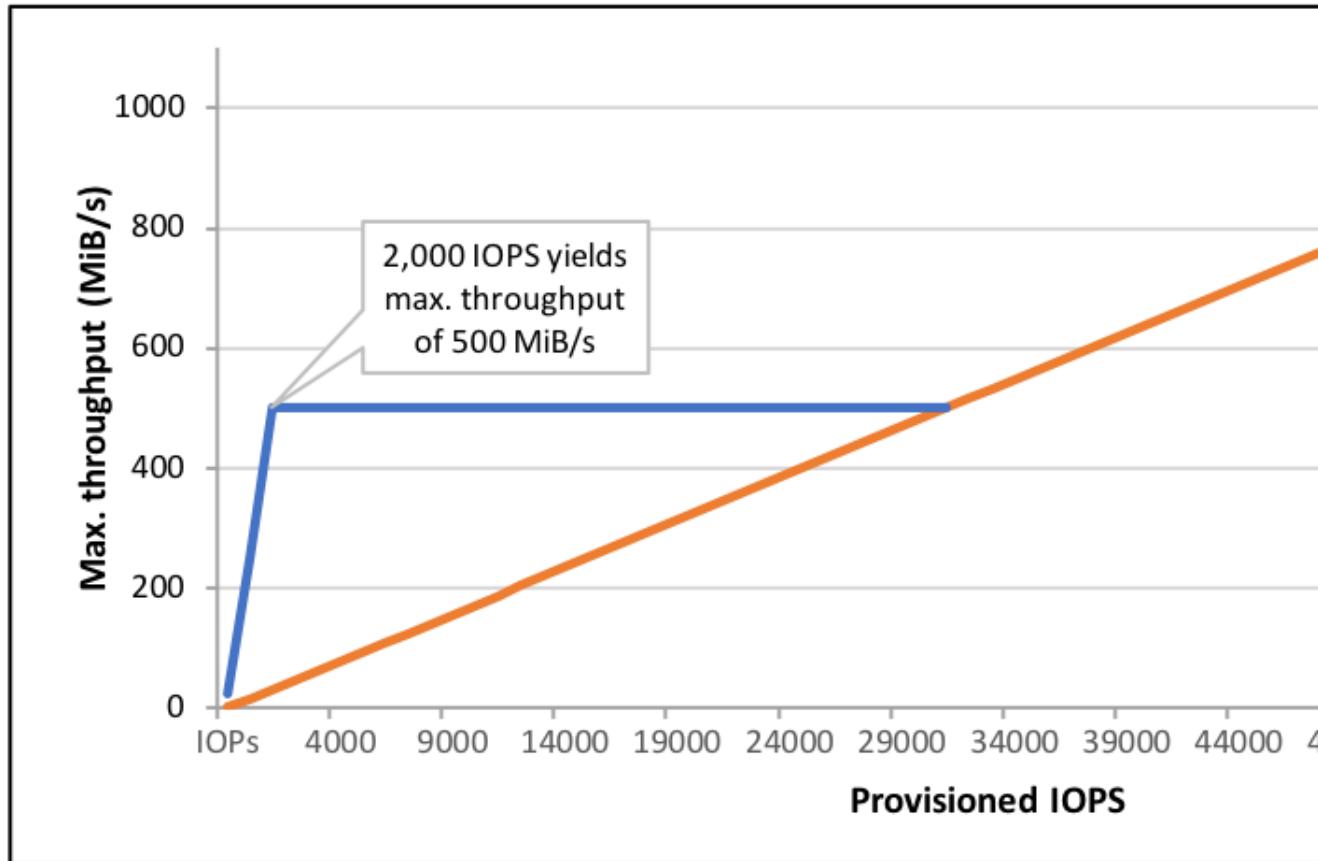
프로비저닝된 IOPS SSD(io1 및 io2) 볼륨은 스토리지 성능과 일관성에 민감한 I/O 집약적 워크로드, 특히 데이터베이스 워크로드의 요구 사항을 충족하도록 설계되었습니다. 버킷과 크레딧 모델을 사용해 성능을 계산하는 gp2와 달리 io1 및 io2 볼륨을 사용하면 볼륨을 생성할 때 일정한 IOPS 속도를 지정할 수 있으며, Amazon EBS는 해당 기간 중 프로비저닝된 성능의 99.9%까지 제공합니다.

io1 볼륨은 99.8~99.9%의 볼륨 내구성을 제공하도록 설계되었으며 연간 장애율(AFR)은 0.2% 이하입니다. 즉, 1년 동안 실행 중인 볼륨 1,000개당 최대 2개의 볼륨 장애가 발생합니다. io2 볼륨은 99.999%의 볼륨 내구성을 제공하도록 설계되었으며 AFR은 0.001% 이하입니다. 즉, 1년 동안 실행 중인 볼륨 100,000개당 1개의 볼륨 장애가 발생합니다.

io1 및 io2 볼륨 크기는 4GiB~16TiB입니다. 100 IOPS부터 [Nitro 시스템에 구축된 인스턴스 \(p. 118\)](#)에서 는 볼륨당 최대 64,000 IOPS, 그리고 다른 인스턴스에서는 최대 32,000까지 프로비저닝할 수 있습니다. 요청된 볼륨 크기(GiB)에 대한 프로비저닝된 IOPS의 최대 비율은 io1 볼륨의 경우 50:1이고 io2 볼륨의 경우 500:1입니다. 예를 들어 100GiB io1 볼륨은 최대 5,000 IOPS로 프로비저닝할 수 있고 100GiB io2 볼륨은 최대 50,000 IOPS로 프로비저닝할 수 있습니다. 지원되는 인스턴스 유형에서 다음 볼륨 크기로 최대 64,000 IOPS까지 프로비저닝할 수 있습니다.

- 크기가 1,280GiB 이상인 io1 볼륨($50 \times 1,280\text{GiB} = 64,000 \text{ IOPS}$)
- 크기가 128GiB 이상인 io2 볼륨($500 \times 128\text{GiB} = 64,000 \text{ IOPS}$)

최대 32,000 IOPS로 프로비저닝된 io1 및 io2 볼륨은 최대 256KiB의 I/O 크기를 지원하고 최대 500MiB/s의 처리량을 제공합니다. I/O 크기가 최대일 때 2,000 IOPS에서 피크 처리량에 도달합니다. 32,000 IOPS 이상(상한 64,000 IOPS)으로 프로비저닝된 볼륨은 최대 16KiB의 I/O 크기를 지원하고 최대 1,000MiB/s의 처리량을 제공합니다. 다음 그래프에 이러한 성능 특성이 예시되어 있습니다.



I/O당 지연 시간 환경은 프로비저닝된 IOPS 및 워크로드 프로파일에 따라 다릅니다. 최상의 I/O 지연 시간 환경을 위해 워크로드의 I/O 프로파일을 충족하도록 IOPS를 프로비저닝해야 합니다.

Note

2012년 이전에 만들어진 일부 AWS 계정은 프로비저닝된 IOPS SSD(io1) 볼륨을 지원하지 않는 us-west-1 또는 ap-northeast-1의 가용 영역에 대한 액세스 권한이 있을 수도 있습니다. 이런 리전 중 하나에 io1 볼륨을 만들거나 블록 디바이스 매핑에서 io1 볼륨이 있는 인스턴스를 시작할 수 없는 경우, 해당 리전에서 다른 가용 영역을 사용해 보십시오. 가용 영역에 4GiB의 io1 볼륨을 만들어 그 영역에서 io1 볼륨을 지원하는지 확인할 수 있습니다.

처리량에 최적화된 HDD(st1) 볼륨

처리량에 최적화된 HDD(st1) 볼륨은 IOPS가 아닌 처리량으로 성능을 정의하는 저비용 마그네틱 스토리지 를 제공합니다. 이 볼륨 유형은 Amazon EMR, ETL, 데이터 웨어하우스, 로그 처리 같은 대용량 순차 워크로드에 적합합니다. 부팅 가능한 st1 볼륨은 지원되지 않습니다.

처리량에 최적화된 HDD(st1) 볼륨은 Cold HDD(sc1) 볼륨과 비슷하지만 자주 액세스하는 데이터를 지원하도록 설계되었습니다.

이 볼륨 유형은 대용량 순차 I/O와 관련된 워크로드에 최적화되어 있으며, 소량의 랜덤 I/O 워크로드를 처리하는 고객에게는 gp2 사용을 권장합니다. 자세한 내용은 [HDD 기반 소량 읽기/쓰기의 비효율성 \(p. 945\)](#) 단원을 참조하십시오.

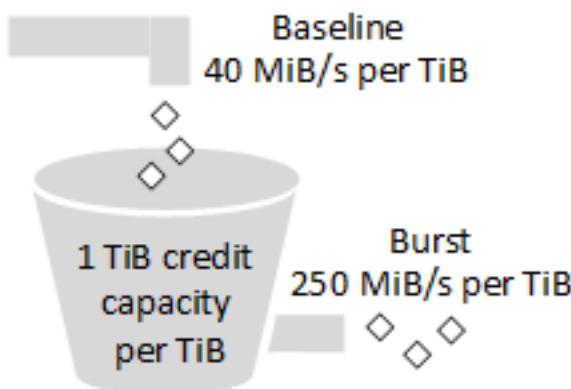
처리량 크레딧 및 버스트 성능

gp2처럼 st1 역시 성능 측정에 버스트 버킷 모델을 사용합니다. 볼륨 크기에 따라 볼륨의 기준 처리량, 즉 볼륨이 처리량 크레딧을 누적하는 속도가 결정됩니다. 볼륨 크기는 볼륨의 버스트 처리량, 즉 사용 가능한 크레

딜을 소비할 수 있는 속도도 결정합니다. 볼륨이 클수록 기본 및 버스트 처리량이 높습니다. 볼륨에 크레딧이 많을수록 버스트 수준에서 더 오랫동안 I/O를 구동할 수 있습니다.

다음 다이어그램은 st1의 버스트 버킷 동작을 보여줍니다.

ST1 burst bucket



처리량 및 처리량 크레딧 한도가 적용되는 st1 볼륨의 사용 가능 처리량은 다음 수식으로 표현됩니다.

$$(Volume\ size) \times (Credit\ accumulation\ rate\ per\ TiB) = Throughput$$

1TiB st1 볼륨의 경우 버스트 처리량은 250MiB/s로 제한되고, 버킷의 크레딧은 40MiB/s 속도로 채워지며, 최대 1TiB에 해당하는 크레딧을 보유할 수 있습니다.

최대 처리량 한도인 500MiB/s 내에서, 볼륨 크기에 비례하여 이러한 제한이 확장됩니다. 버킷이 고갈된 후 처리량은 TiB당 40MiB/s의 기준 속도로 제한됩니다.

0.5TiB~16TiB 범위의 볼륨 크기를 기준으로 기준 처리량은 20MiB/s~500MiB/s(한도)이며, 다음과 같이 12.5TiB에서 한도에 도달합니다.

$$12.5 \text{ TiB} \times \frac{40 \text{ MiB/s}}{1 \text{ TiB}} = 500 \text{ MiB/s}$$

버스트 처리량은 125MiB/s~500MiB/s(한도)이며, 다음과 같이 2TiB에서 한도에 도달합니다.

$$2 \text{ TiB} \times \frac{250 \text{ MiB/s}}{1 \text{ TiB}} = 500 \text{ MiB/s}$$

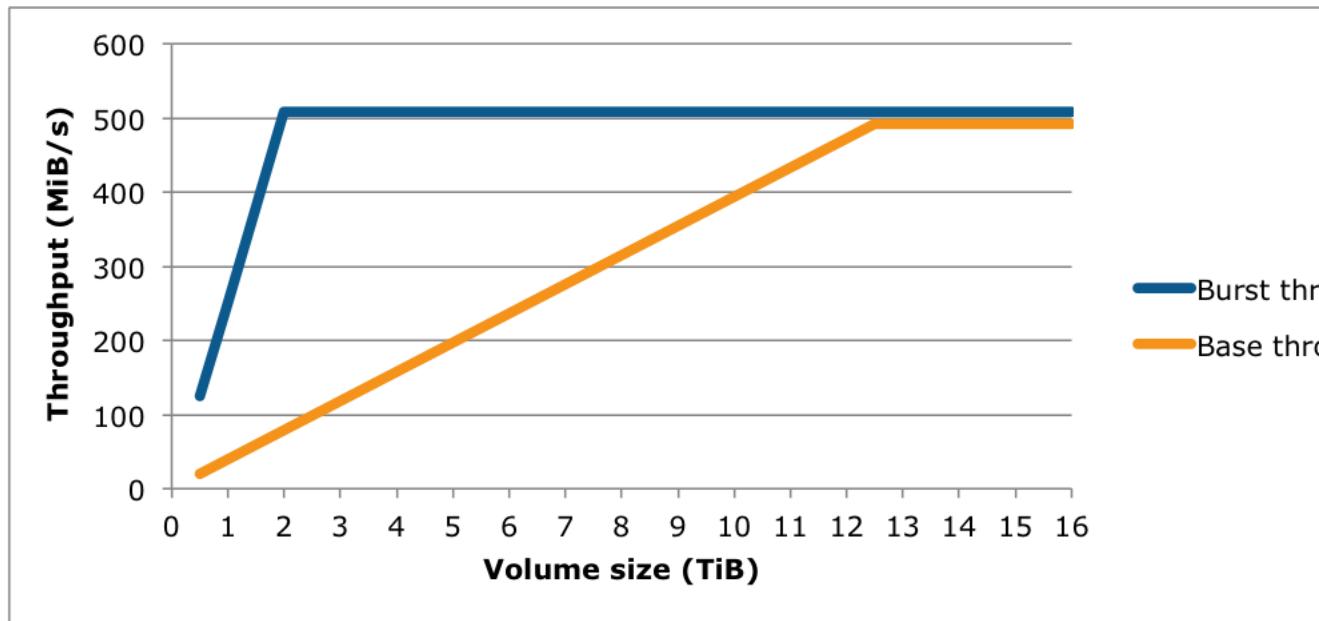
다음 표는 st1의 기준 및 버스트 처리량 값 전체를 보여줍니다.

볼륨 크기(TiB)	ST1 기준 처리량(MiB/s)	ST1 버스트 처리량(MiB/s)
0.5	20	125
1	40	250
2	80	500
3	120	500

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 볼륨

볼륨 크기(TiB)	ST1 기준 처리량(MiB/s)	ST1 버스트 처리량(MiB/s)
4	160	500
5	200	500
6	240	500
7	280	500
8	320	500
9	360	500
10	400	500
11	440	500
12	480	500
12.5	500	500
13	500	500
14	500	500
15	500	500
16	500	500

다음 다이어그램은 표의 값을 도식화한 것입니다.



Note

처리량에 최적화된 HDD(st1) 볼륨의 스냅샷을 생성하는 경우, 스냅샷이 진행되는 동안 성능이 볼륨의 기준 값까지 떨어질 수 있습니다.

CloudWatch 지표 및 경보를 사용하여 버스트 버킷 잔고를 모니터링하는 방법은 [gp2, st1, sc1 볼륨에 대한 버스트 버킷 밸런스 모니터링 \(p. 945\)](#) 단원을 참조하십시오.

Cold HDD(sc1) 볼륨

Cold HDD(sc1) 볼륨은 IOPS가 아닌 처리량으로 성능을 정의하는 저비용 마그네틱 스토리지를 제공합니다. 처리량 제한이 st1보다 낮은 sc1은 대용량 순차 콜드 데이터 워크로드에 적합합니다. 데이터에 자주 액세스 할 필요가 없고 비용을 절약해야 한다면 저렴한 블록 스토리지로 sc1이 적합합니다. 부팅 가능한 sc1 볼륨은 지원되지 않습니다.

Cold HDD(sc1) 볼륨은 처리량에 최적화된 HDD(st1) 볼륨과 비슷하지만 자주 액세스하지 않는 데이터를 지원하도록 설계되었습니다.

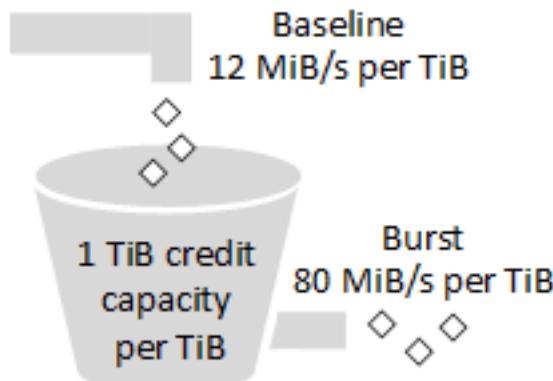
Note

이 볼륨 유형은 대용량 순차 I/O와 관련된 워크로드에 최적화되어 있으며, 소량의 랜덤 I/O 워크로드를 처리하는 고객에게는 gp2 사용을 권장합니다. 자세한 내용은 [HDD 기반 소량 읽기/쓰기의 비효율성 \(p. 945\)](#) 단원을 참조하십시오.

처리량 크레딧 및 버스트 성능

gp2처럼 sc1 역시 성능 측정에 버스트 버킷 모델을 사용합니다. 볼륨 크기에 따라 볼륨의 기준 처리량, 즉 볼륨이 처리량 크레딧을 누적하는 속도가 결정됩니다. 볼륨 크기는 볼륨의 버스트 처리량, 즉 사용 가능한 크레딧을 소비할 수 있는 속도도 결정합니다. 볼륨이 클수록 기본 및 버스트 처리량이 높습니다. 볼륨에 크레딧이 많을수록 버스트 수준에서 더 오랫동안 I/O를 구동할 수 있습니다.

SC1 burst bucket



처리량 및 처리량 크레딧 한도가 적용되는 sc1 볼륨의 사용 가능 처리량은 다음 수식으로 표현됩니다.

$$(\text{Volume size}) \times (\text{Credit accumulation rate per TiB}) = \text{Throughput}$$

1TiB sc1 볼륨의 경우 버스트 처리량은 80MiB/s로 제한되고, 버킷의 크레딧은 12MiB/s 속도로 채워지며, 최대 1TiB에 해당하는 크레딧을 보유할 수 있습니다.

최대 처리량 한도인 250MiB/s 내에서, 볼륨 크기에 비례하여 이러한 제한이 확장됩니다. 버킷이 고갈된 후 처리량은 TiB당 12MiB/s의 기준 속도로 제한됩니다.

0.5TiB~16TiB 범위의 볼륨 크기를 기준으로 기준 처리량은 6MiB/s~192MiB/s(최대)이며, 다음과 같이 16TiB에서 한도에 도달합니다.

$$\frac{12 \text{ MiB/s}}{16 \text{ TiB}} \times \frac{\text{-----}}{1 \text{ TiB}} = 192 \text{ MiB/s}$$

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 볼륨

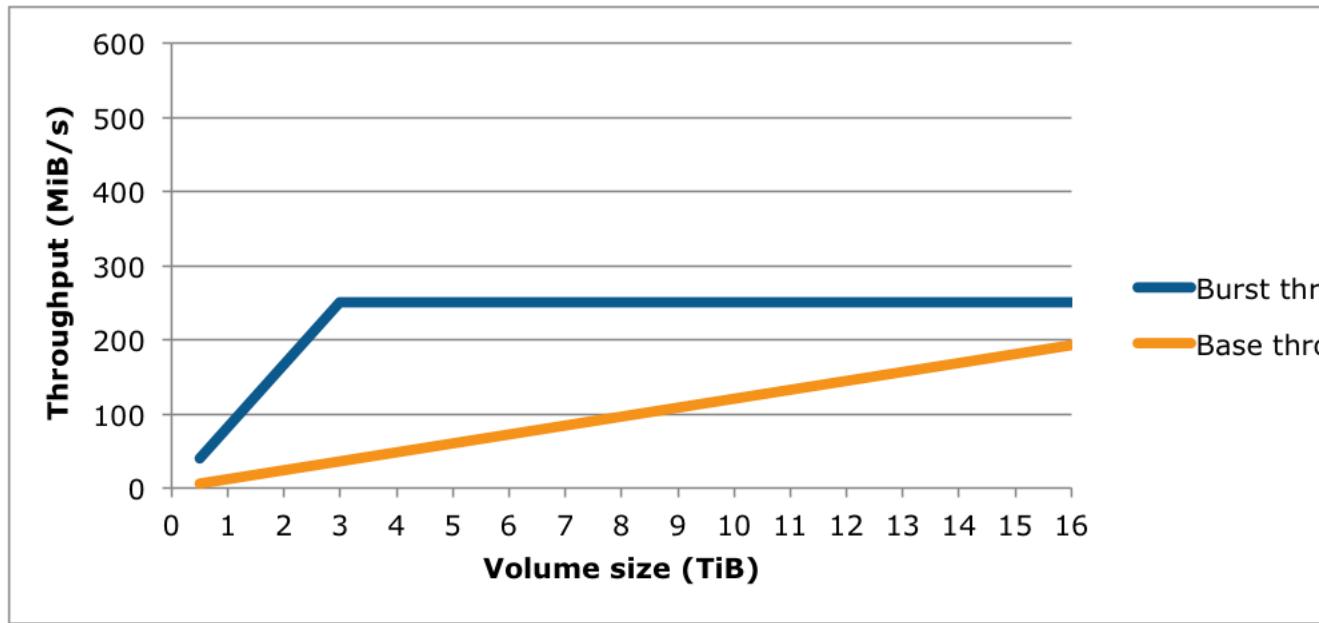
버스트 처리량은 40MiB/s~250MiB/s(한도)이며, 다음과 같이 3.125TiB에서 한도에 도달합니다.

$$80 \text{ MiB/s} \\ 3.125 \text{ TiB} \times \frac{80 \text{ MiB/s}}{1 \text{ TiB}} = 250 \text{ MiB/s}$$

다음 표는 sc1의 기준 및 버스트 처리량 값 전체를 보여줍니다.

볼륨 크기(TiB)	SC1 기준 처리량(MiB/s)	SC1 버스트 처리량(MiB/s)
0.5	6	40
1	12	80
2	24	160
3	36	240
3.125	37.5	250
4	48	250
5	60	250
6	72	250
7	84	250
8	96	250
9	108	250
10	120	250
11	132	250
12	144	250
13	156	250
14	168	250
15	180	250
16	192	250

다음 다이어그램은 표의 값을 도식화한 것입니다.



Note

Cold HDD(sc1) 볼륨의 스냅샷을 생성하는 경우, 스냅샷이 진행되는 동안 성능이 볼륨의 기준 값까지 떨어질 수 있습니다.

CloudWatch 지표 및 경보를 사용하여 버스트 버킷 잔고를 모니터링하는 방법은 [gp2, st1, sc1 볼륨에 대한 버스트 버킷 밸런스 모니터링 \(p. 945\)](#) 단원을 참조하십시오.

Magnetic (standard)

Magnetic 볼륨은 마그네틱 드라이브로 구성되어 있으며, 데이터 액세스가 드문 워크로드, 작은 볼륨 크기에 맞는 저비용 스토리지가 중요한 시나리오에 적합합니다. Magnetic 볼륨의 평균 IOPS는 약 100 정도이며, 버스팅 시 몇백 수준으로 증가합니다. 크기는 1GiB에서 1TiB까지입니다.

Note

Magnetic은 이전 세대 볼륨 유형입니다. 새로운 애플리케이션에는 새로운 볼륨 유형 중에서 선택해서 사용하는 것이 좋습니다. 자세한 내용은 [이전 세대 볼륨을 참조하십시오.](#)

CloudWatch 지표 및 경보를 사용하여 버스트 버킷 잔고를 모니터링하는 방법은 [gp2, st1, sc1 볼륨에 대한 버스트 버킷 밸런스 모니터링 \(p. 945\)](#) 단원을 참조하십시오.

HDD 볼륨 사용 시 성능 고려사항

HDD 볼륨 사용 시 최적의 처리량을 달성하려면 다음 사항을 염두에 두고 워크로드를 계획하십시오.

처리량에 최적화된 HDD 대 Cold HDD 비교

st1 및 sc1의 버킷 크기는 볼륨 크기에 따라 다르며, 최대 버킷에는 최대 볼륨 스캔에 충분한 토큰이 포함되어 있습니다. 그러나 st1 및 sc1 볼륨이 더 큰 경우 인스턴스당, 볼륨당 처리량 제한 때문에 볼륨 스캔을 완료하는 시간이 더 오래 걸립니다. 작은 인스턴스에 연결된 볼륨은 st1 또는 sc1 처리량이 아닌 인스턴스당 처리량에 따라 제한됩니다.

st1 및 sc1은 모두 99%의 기간 동안 90%의 버스트 처리량에 성능 일관성을 제공하도록 설계되었습니다. 매 시간 총 처리량 목표 99%를 달성하기 위해, 준수하지 않는 기간은 대략적으로 균등하게 분산됩니다.

다음 표는 최대 버킷과 충분한 인스턴스 처리량을 가정할 때 다양한 크기의 볼륨에 이상적인 스캔 시간을 보여줍니다.

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 볼륨

일반적으로 스캔 시간은 이 수식으로 표현됩니다.

```
Volume size
----- = Scan time
Throughput
```

예를 들어 성능 일관성 보장과 기타 최적화를 고려할 때, 5TiB 볼륨을 사용 중인 st1 고객이 전체 볼륨 스캔을 완료하는 데 걸리는 시간은 2.91~3.27시간으로 예상할 수 있습니다.

```
5 TiB      5 TiB
----- = ----- = 10,486 s = 2.91 hours (optimal)
500 MiB/s   0.00047684 TiB/s

2.91 hours
2.91 hours + ----- = 3.27 hours (minimum expected)
(0.90)(0.99) <-- From expected performance of 90% of burst 99% of the time
```

마찬가지로, 5TiB 볼륨을 사용 중인 sc1 고객이 전체 볼륨 스캔을 완료하는 데 걸리는 시간은 5.83~6.54시간으로 예상됩니다.

```
5 TiB
----- = 20972 s = 5.83 hours (optimal)
0.000238418 TiB/s

5.83 hours
----- = 6.54 hours (minimum expected)
(0.90)(0.99)
```

볼륨 크기(TiB)	버스팅 시 ST1 스캔 시간(단위: 시간)*	버스팅 시 SC1 스캔 시간(단위: 시간)*
1	1.17	3.64
2	1.17	3.64
3	1.75	3.64
4	2.33	4.66
5	2.91	5.83
6	3.50	6.99
7	4.08	8.16
8	4.66	9.32
9	5.24	10.49
10	5.83	11.65
11	6.41	12.82
12	6.99	13.98
13	7.57	15.15
14	8.16	16.31

볼륨 크기(TiB)	버스팅 시 ST1 스캔 시간(단위: 시간)*	버스팅 시 SC1 스캔 시간(단위: 시간)*
15	8.74	17.48
16	9.32	18.64

* 이 스캔 시간은 1MiB의 순차 I/O를 수행할 때 4 이상의 평균 대기열 깊이(가장 가까운 정수로 반올림)를 가정합니다.

따라서 빠르게 스캔을 완료해야 하거나(최대 500MiB/s) 하루 안에 여러 건의 전체 볼륨 스캔이 필요한 처리량 중심의 워크로드를 가지고 있는 경우 st1을 사용하십시오. 비용을 최적화해야 하고 데이터에 그다지 자주 액세스하지 않으며 250MiB/s 이상의 스캔 성능이 필요하지 않다면 sc1을 사용하십시오.

HDD 기반 소량 읽기/쓰기의 비효율성

st1 및 sc1 볼륨의 성능 모델은 순차 I/O, 높은 처리량의 워크로드 사용, 혼합 IOPS 및 처리량의 워크로드에 허용되는 성능 제공, 소용량 랜덤 I/O 회피에 최적화되어 있습니다.

예를 들어 1MiB 이하의 I/O 요청은 1MiB I/O 크레딧으로 간주됩니다. 그러나 순차 I/O는 1MiB I/O 블록으로 병합되고 1MiB I/O 크레딧으로 간주됩니다.

인스턴스당 처리량에 대한 제한

st1 및 sc1 볼륨의 처리량은 항상 다음 중 작은 값에 따라 결정됩니다.

- 볼륨의 처리량 제한
- 인스턴스의 처리량 제한

모든 Amazon EBS 볼륨에서와 같이, 네트워크 병목 현상을 피하려면 적절한 EBS 최적화 EC2 인스턴스를 선택하는 것이 좋습니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

gp2, st1, sc1 볼륨에 대한 버스트 버킷 밸런스 모니터링

gp2, st1, sc1 볼륨에 대해 Amazon CloudWatch에서 제공하는 EBS BurstBalance 지표를 사용하여 버스트 버킷 수준을 모니터링할 수 있습니다. 이 측정치는 버스트 버킷에 남아 있는 I/O 크레딧(gp2의 경우)의 비율 또는 처리량 크레딧의 비율(st1 및 sc1의 경우)을 보여줍니다. BurstBalance 지표 및 I/O와 관련된 기타 지표에 대한 자세한 내용은 [I/O 특성 및 모니터링 \(p. 1064\)](#) 단원을 참조하십시오. CloudWatch에서는 BurstBalance 값이 특정 수준 밑으로 떨어질 경우 이를 알리도록 경보를 설정할 수도 있습니다. 자세한 내용은 [Amazon CloudWatch 경보 생성](#)을 참조하십시오.

EBS 볼륨의 크기 및 구성에 대한 제약

Amazon EBS 볼륨의 크기는 블록 데이터 스토리지의 물리 및 산술뿐 아니라 운영 체제(OS) 및 파일 시스템 디자이너의 구현 결정에 따라서도 제약을 받습니다. AWS는 서비스의 안정성을 보호하기 위해 볼륨 크기에 추가 제한을 부과합니다.

다음 섹션에서는 EBS 볼륨의 사용 가능한 크기를 제한하고 EBS 볼륨을 구성하기 위한 권장 사항을 제공하는 가장 중요한 요소에 대해 설명합니다.

목차

- [스토리지 용량 \(p. 946\)](#)
- [서비스 제한 \(p. 946\)](#)
- [파티셔닝 체계 \(p. 946\)](#)
- [데이터 블록 크기 \(p. 947\)](#)

스토리지 용량

다음 표에는 4,096바이트 블록 크기를 가정할 때 Amazon EBS 가장 일반적으로 사용되는 파일 시스템의 이론적 스토리지 용량과 구현된 스토리지 용량이 요약되어 있습니다.

파티셔닝 체계	최대 주소 지정 가능한 블록	이론적 최대 크기(블록 x 블록 크기)	Ext4에서 구현되는 최대 크기*	XFS에서 구현되는 최대 크기**	NTFS에서 구현되는 최대 크기	EBS에서 지원되는 최대 크기
MBR	2^{32}	2TiB	2TiB	2TiB	2TiB	2TiB
GPT	2^{64}	64ZiB	$1\text{EiB} = 1024^2\text{TiB}$ (RHEL7에서 인증된 50TiB)	500TiB (RHEL7에서 인증됨)	256TiB	16TiB

* https://ext4.wiki.kernel.org/index.php/Ext4_Howto 및 <https://access.redhat.com/solutions/1532>

** <https://access.redhat.com/solutions/1532>

서비스 제한

Amazon EBS는 데이터 센터에서 대량으로 분산되는 스토리지를 가상 하드 디스크 드라이브로 추상화합니다. EC2 인스턴스에 설치된 운영 체제에서 연결된 EBS 볼륨은 512바이트 디스크 섹터가 포함된 물리적 하드 디스크 드라이브로 나타납니다. OS는 스토리지 관리 유틸리티를 통해 해당 가상 섹터에 데이터 블록(또는 클러스터)을 할당하는 작업을 관리합니다. 할당은 마스터 부트 레코드(MBR) 또는 GUID 파티션 테이블(GPT)과 같은 볼륨 파티셔닝 체계에 따라 수행되며 설치된 파일 시스템(ext4, NTFS 등)의 기능 내에서 수행됩니다.

EBS는 가상 디스크 섹터에 포함된 데이터를 인식하지 않으며, 섹터의 무결성을 보장할 뿐입니다. 따라서 AWS 작업과 OS 작업은 서로 독립적입니다. 볼륨 크기를 선택할 때는 다음과 같은 경우의 기능과 한계를 알아 두십시오.

- EBS는 현재 16TiB의 최대 볼륨 크기를 지원합니다. 즉, EBS 볼륨의 크기를 16TiB까지 만들 수 있지만, OS가 해당 용량을 모두 인식하는지 여부는 자체적인 설계 특성 및 볼륨 파티셔닝 방법에 따라 결정됩니다.
- Amazon EC2를 사용하려면 Windows 부트 볼륨에 MBR 파티셔닝을 사용해야 합니다. [파티셔닝 체계 \(p. 946\)](#) 단원에서 설명한 바와 같이, 이 제한으로 인해 부트 볼륨이 2TiB보다 클 수 없습니다. Windows 데이터 볼륨에는 이 제한 사항이 적용되지 않으며 GPT 파티셔닝을 사용할 수 있습니다. 2TiB 이상의 Windows 부트 볼륨이 동적 MBR 파티션 테이블을 사용하도록 변환되는 경우 디스크 관리자에 볼륨의 오차가 표시됩니다.
- 2TiB(2,048GiB) 이상의 Windows 비부트 볼륨이 전체 볼륨에 액세스하려면 GPT 파티션 테이블을 사용해야 합니다. 크기가 2TiB를 초과하는 EBS 볼륨이 시작 시 Windows 인스턴스에 연결된 경우 GPT 파티션 테이블을 사용하여 자동으로 포맷되지 않습니다. 크기가 2TiB를 초과하는 EBS 볼륨을 시작 후 Windows 인스턴스에 연결하는 경우 GPT 테이블을 사용하여 수동으로 초기화해야 합니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오.

파티셔닝 체계

다른 영향 중에서도 특히, 파티셔닝 체계는 단일 볼륨에서 여러 논리적 데이터 블록을 고유하게 주소 지정할 수 있는 방법을 결정합니다. 자세한 내용은 [데이터 블록 크기 \(p. 947\)](#) 단원을 참조하십시오. 사용 중인 일반적인 파티셔닝 체계는 마스터 부트 레코드(MBR) 및 GUID 파티션 테이블(GPT)입니다. 이러한 체계 간의 중요한 차이점은 다음과 같이 요약할 수 있습니다.

MBR

MBR은 32비트 데이터 구조를 사용하여 블록 주소를 저장합니다. 따라서 각 데이터 블록은 2^{32} 개의 가능한 정수 중 하나와 매핑됩니다. 주소 지정 가능한 최대 볼륨 크기는 다음과 같은 방법으로 지정됩니다.

$$(2^{32} - 1) \times \text{Block size} = \text{Number of addressable blocks}$$

MBR 볼륨의 블록 크기는 관례적으로 512바이트로 제한됩니다. 따라서:

$$(2^{32} - 1) \times 512 \text{ bytes} = 2 \text{ TiB} - 512 \text{ bytes}$$

MBR 볼륨에 대한 이 2TiB 제한을 증가시키기 위한 엔지니어링 해결 방법은 업계에서 광범위하게 채택되는 방식과 일치하지 않습니다. 따라서 AWS에서 크기가 더 크게 표시되더라도 Linux와 Windows에서는 MBR 볼륨이 2TiB보다 더 큰 크기로 인식되지 않습니다.

GPT

GPT는 64비트 데이터 구조를 사용하여 블록 주소를 저장합니다. 따라서 각 데이터 블록은 2^{64} 개의 가능한 정수 중 하나와 매핑됩니다. 주소 지정 가능한 최대 볼륨 크기는 다음과 같은 방법으로 지정됩니다.

$$(2^{64} - 1) \times \text{Block size} = \text{Number of addressable blocks}$$

GPT 볼륨의 블록 크기는 일반적으로 4,096바이트입니다. 따라서:

$$\begin{aligned} (2^{64} - 1) \times 4,096 \text{ bytes} \\ = 2^{64} \times 4,096 \text{ bytes} - 1 \times 4,096 \text{ bytes} \\ = 2^{64} \times 2^{12} \text{ bytes} - 4,096 \text{ bytes} \\ = 2^{70} \times 2^6 \text{ bytes} - 4,096 \text{ bytes} \\ = 64 \text{ Zib} - 4,096 \text{ bytes} \end{aligned}$$

실제 컴퓨터 시스템은 이러한 이론적 최대 크기와 비슷한 크기를 지원하지 않습니다. 구현되는 파일 시스템 크기는 현재 ext4의 경우 50TiB 및 NTFS의 경우 256TiB로 제한되며, 이 두 가지 크기는 모두 AWS에서 부과하는 16TiB 제한을 초과합니다.

데이터 블록 크기

최신 하드 드라이브의 데이터 스토리지는 논리적 블록 주소 지정을 통해 관리됩니다. 논리적 블록 주소 지정은 운영 체제가 기본 하드웨어에 대한 많은 지식 없이 논리적 블록에서 데이터를 읽고 쓸 수 있도록 하는 추상적 계층입니다. OS는 스토리지 디바이스를 이용하여 블록을 물리적 섹터에 매핑합니다. EBS는 운영 체제에 512바이트 섹터를 공급하며, 운영 체제는 섹터 크기보다 몇 배 더 큰 데이터 블록을 사용하여 디스크에서 데이터를 읽고 씁니다.

논리 데이터 블록의 업계 기본 크기는 현재 4,096바이트(4KiB)입니다. 특정 워크로드는 더 작거나 더 큰 블록 크기에서 이점을 얻을 수 있기 때문에 파일 시스템은 포맷 중 지정할 수 없는 비 기본 블록 크기를 지원합니다. 비 기본 블록 크기를 사용해야 하는 시나리오는 이 주제의 범위를 벗어나지만, 블록 크기 선택은 볼륨의 스토리지 용량에 영향을 미칩니다. 다음 표에는 스토리지 용량이 블록 크기의 함수로 표시됩니다.

블록 크기	최대 볼륨 크기
4KiB(기본값)	16TiB
8KiB	32TiB
16KiB	64TiB
32KiB	128TiB
64KiB(최대)	256TiB

현재 EBS에서 부과하는 볼륨 크기(16TiB)에 대한 제한은 4KiB 데이터 블록에서 지원되는 최대 크기와 같습니다.

Amazon EBS 볼륨 생성

Amazon EBS 볼륨을 생성한 다음, 동일한 가용 영역에 있는 모든 EC2 인스턴스에 연결할 수 있습니다. 암호화된 EBS 볼륨을 생성하는 경우, 지원되는 인스턴스 유형에만 연결할 수 있습니다. 자세한 내용은 [지원되는 인스턴스 유형 \(p. 1034\)](#) 단원을 참조하십시오.

고성능 스토리지 시나리오용으로 볼륨을 생성하는 경우 프로비저닝된 IOPS SSD(io1 또는 io2) 볼륨을 사용하고 EBS에 최적화된 인스턴스 또는 10Gb 네트워크에 연결된 인스턴스 등 애플리케이션을 지원하기에 충분한 대역폭이 있는 인스턴스에 연결합니다. 처리량에 최적화된 HDD(st1) 및 Cold HDD(sc1) 볼륨에도 같은 원칙이 적용됩니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

빈 EBS 볼륨은 사용 가능하지만 초기화(이전에는 사전 위밍이라고 함)가 필요하지 않은 시점에 최고 성능을 발휘합니다. 하지만 스냅샷에서 생성된 볼륨의 스토리지 블록은 초기화(Amazon S3에서 가져와 볼륨에 기록) 후에만 액세스할 수 있습니다. 이 예비 작업은 시간이 걸리며, 각 블록을 처음 액세스할 때 I/O 작업의 지연 시간을 상당히 증가시킬 수 있습니다. 모든 블록을 다운로드하고 볼륨에 기록한 후에 볼륨 성능이 구현됩니다. 대부분 애플리케이션의 경우 볼륨 수명 주기 동안 이 비용을 분할 상환할 수 있습니다. 프로덕션 환경에서 이러한 초기 성능 저하를 방지하려면 전체 볼륨을 즉시 초기화하거나 빠른 스냅샷 복원을 활성화할 수 있습니다. 자세한 내용은 [Amazon EBS 볼륨 초기화 \(p. 1066\)](#) 단원을 참조하십시오.

볼륨 생성 방법

- 인스턴스 시작 시 블록 디바이스 매핑을 지정하여 EBS 볼륨을 생성하고 연결합니다. 자세한 내용은 [인스턴스 시작 마법사를 사용하여 인스턴스 시작 \(p. 376\)](#) 및 [블록 디바이스 매핑 \(p. 1107\)](#) 단원을 참조하십시오.
- 빈 EBS 볼륨을 생성하여 실행 인스턴스에 연결합니다. 자세한 내용은 아래 [빈 볼륨 생성 \(p. 948\)](#) 단원을 참조하십시오.
- 이전에 생성한 스냅샷에서 EBS 볼륨을 생성하여 실행 중인 인스턴스에 연결합니다. 자세한 내용은 아래 [스냅샷에서 볼륨 생성 \(p. 949\)](#) 단원을 참조하십시오.

빈 볼륨 생성

빈 볼륨은 사용 가능하지만 초기화가 필요하지 않은 시점에 최고 성능을 발휘합니다.

콘솔을 사용하여 빈 EBS 볼륨을 생성하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 모음에서 볼륨을 생성할 리전을 선택합니다. 일부 Amazon EC2 리소스는 리전 간에 공유될 수 있지만 그렇지 않은 리소스도 있으므로 잘 선택해야 합니다. 자세한 정보는 [리소스 위치 \(p. 1132\)](#) 단원을 참조하십시오.
- 탐색 창에서 ELASTIC BLOCK STORE, 볼륨을 선택합니다.
- 볼륨 생성을 선택합니다.
- 볼륨 유형에서 볼륨 유형을 선택합니다. 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.
- 크기(GiB)에서 볼륨의 크기를 입력합니다. 자세한 내용은 [EBS 볼륨의 크기 및 구성에 대한 제약 \(p. 945\)](#) 단원을 참조하십시오.
- 프로비저닝된 IOPS SSD 볼륨의 경우 IOPS에서 볼륨이 지원해야 하는 최대 IOPS(초당 입/출력 작업) 수를 입력합니다.
- 가용 영역에서 볼륨을 생성할 가용 영역을 선택합니다. EBS 볼륨은 동일한 가용 영역 내의 EC2 인스턴스에만 연결될 수 있습니다.
- (선택 사항) 인스턴스 유형이 EBS 암호화를 지원하고 사용자가 볼륨을 암호화하려는 경우 Encrypt this volume(이 볼륨 암호화)를 선택하고 CMK를 선택합니다. 기본적으로 암호화가 이 리전에서 활성화되는

경우 EBS 암호화가 활성화되고 EBS 암호화에 대한 기본 CMK가 선택됩니다. 마스터 키에서 다른 CMK를 선택하거나 액세스할 수 있는 키의 전체 ARN을 붙여넣을 수 있습니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

10. (선택 사항) 추가 태그 생성(Create additional tags)을 선택하여 볼륨에 태그를 추가합니다. 각 태그에 대해 태그 키와 태그 값을 제공합니다. 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.
11. 볼륨 생성을 선택합니다. 볼륨 상태가 사용 가능이면 볼륨을 사용할 준비가 된 것입니다.
12. 새 볼륨을 사용하려면 인스턴스에 연결하고 포맷한 다음 탑재합니다. 자세한 내용은 [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#) 단원을 참조하십시오.

명령줄을 사용하여 빈 EBS 볼륨을 생성하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [create-volume](#) AWS CLI
- [New-EC2Volume](#) Windows PowerShell용 AWS 도구

스냅샷에서 볼륨 생성

스냅샷에서 생성된 새 볼륨은 백그라운드에서 느리게 로드됩니다. 이는 Amazon S3에서 EBS 볼륨으로 모든 데이터가 전송될 때까지 기다리지 않아도 인스턴스에서 연결된 볼륨과 모든 데이터에 액세스할 수 있음을 의미합니다. 인스턴스가 아직 로드되지 않은 데이터에 액세스하는 경우, 볼륨은 요청한 데이터를 Amazon S3에서 즉시 다운로드한 후, 백그라운드에서 볼륨 데이터의 나머지 로드를 진행합니다. 모든 블록을 다운로드하고 볼륨에 기록한 후에 볼륨 성능이 구현됩니다. 프로덕션 환경에서 초기 성능 저하를 방지하려면 [Amazon EBS 볼륨 초기화 \(p. 1066\)](#) 단원을 참조하십시오.

암호화된 스냅샷에서 생성된 새 EBS 볼륨은 자동으로 암호화됩니다. 암호화되지 않은 스냅샷에서 복원하는 동안 즉석에서 볼륨을 암호화할 수도 있습니다. 암호화된 볼륨은 EBS 암호화를 지원하는 인스턴스 유형에만 연결할 수 있습니다. 자세한 내용은 [지원되는 인스턴스 유형 \(p. 1034\)](#) 단원을 참조하십시오.

스냅샷에서 볼륨을 생성하려면 다음 절차에 따르십시오.

콘솔을 사용하여 스냅샷에서 EBS 볼륨을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 스냅샷이 있는 리전을 선택합니다.

스냅샷을 사용하여 다른 리전에서 볼륨을 생성하려면 스냅샷을 새 리전에 복사한 다음, 이를 사용하여 해당 리전에서 볼륨을 생성할 수 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 복사 \(p. 985\)](#) 단원을 참조하십시오.

3. 탐색 창에서 ELASTIC BLOCK STORE, 볼륨을 선택합니다.
4. 볼륨 생성을 선택합니다.
5. 볼륨 유형에서 볼륨 유형을 선택합니다. 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.
6. 스냅샷 ID에서 볼륨을 복원할 스냅샷의 ID 또는 설명을 입력한 다음 제안 옵션 목록에서 선택합니다.
7. (선택 사항) 이 볼륨 암호화를 선택하면 볼륨의 암호화 상태를 변경할 수 있습니다. [암호화 기본 제공 \(p. 1036\)](#)이 활성화되어 있는 경우 선택 사항입니다. 마스터 키에서 CMK를 선택하여 EBS 암호화를 위한 기본 CMK 이외의 CMK를 지정합니다.
8. 크기(GiB)에서 볼륨의 크기를 입력하거나 스냅샷의 기본 크기가 적절한지 확인합니다.

볼륨 크기와 스냅샷을 모두 지정한 경우 크기는 스냅샷 크기보다 크거나 같아야 합니다. 볼륨 유형과 스냅샷을 선택하면 볼륨의 최소 및 최대 크기가 크기 옆에 표시됩니다. 자세한 내용은 [EBS 볼륨의 크기 및 구성에 대한 제약 \(p. 945\)](#) 단원을 참조하십시오.

9. 프로비저닝된 IOPS SSD 볼륨의 경우 IOPS에서 볼륨이 지원해야 하는 최대 IOPS(초당 입/출력 작업 수)를 입력합니다.
10. 가용 영역에서 볼륨을 생성할 가용 영역을 선택합니다. 동일한 가용 영역의 EC2 인스턴스에만 EBS 볼륨을 연결할 수 있습니다.
11. (선택 사항) 추가 태그 생성(Create additional tags)을 선택하여 볼륨에 태그를 추가합니다. 각 태그에 대해 태그 키와 태그 값을 제공합니다.
12. 볼륨 생성을 선택합니다.
13. 새 볼륨을 사용하려면 인스턴스에 연결하고 탑재합니다. 자세한 내용은 [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#) 단원을 참조하십시오.
14. 스냅샷보다 큰 볼륨을 생성한 경우, 볼륨의 파일 시스템을 확장하여 추가 공간을 활용할 수 있어야 합니다. 자세한 내용은 [Amazon EBS 탄력적 볼륨 \(p. 1023\)](#) 단원을 참조하십시오.

명령줄을 사용하여 스냅샷에서 EBS 볼륨을 생성하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `create-volume` AWS CLI
- `New-EC2Volume` Windows PowerShell용 AWS 도구

Amazon EBS 볼륨을 인스턴스에 연결

사용 가능한 EBS 볼륨을 해당 볼륨과 동일한 가용 영역에 있는 하나 이상의 인스턴스에 연결할 수 있습니다.

사전 조건

- 인스턴스에 연결할 수 있는 볼륨 수를 확인합니다. 자세한 내용은 [인스턴스 볼륨 제한 \(p. 1104\)](#) 단원을 참조하십시오.
- 볼륨이 암호화된 경우에는 Amazon EBS 암호화를 지원하는 인스턴스에만 연결할 수 있습니다. 자세한 내용은 [지원되는 인스턴스 유형 \(p. 1034\)](#) 단원을 참조하십시오.
- 볼륨에 AWS Marketplace 제품 코드가 있는 경우:
 - 해당 볼륨은 중지된 인스턴스에만 연결될 수 있습니다.
 - 이 경우 볼륨에 있는 AWS Marketplace 코드에 구독되어 있어야 합니다.
 - 인스턴스 구성(인스턴스 유형, 운영 체제)에서 해당 AWS Marketplace 코드를 지원해야 합니다. 예를 들어, Windows 인스턴스의 볼륨을 Linux 인스턴스로 연결할 수 없습니다.
 - AWS 제품 코드는 볼륨에서 인스턴스로 복사됩니다.

콘솔을 사용하여 EBS 볼륨을 인스턴스에 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 ELASTIC BLOCK STORE, 볼륨을 선택합니다.
3. 사용 가능한 볼륨을 선택한 후 작업, 볼륨 연결을 선택합니다.
4. 인스턴스의 경우, 인스턴스 이름 혹은 ID를 입력하십시오. 옵션 목록에서 인스턴스를 선택하십시오.(보기 볼륨과 동일한 가용 영역의 인스턴스만 표시됩니다.)
5. 디바이스의 경우, 제안된 디바이스 이름을 사용하거나 지원되는 다른 디바이스 이름을 입력할 수 있습니다. 자세한 내용은 [Windows 인스턴스의 디바이스 명명 \(p. 1106\)](#) 단원을 참조하십시오.
6. 연결을 선택합니다.
7. 인스턴스에 연결하고 볼륨을 탑재합니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오.

명령줄을 사용하여 EBS 볼륨을 인스턴스에 연결하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [attach-volume](#) AWS CLI
- [Add-EC2Volume](#) (Windows PowerShell용 AWS 도구)

Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기

인스턴스에 Amazon EBS 볼륨을 연결하면 이 볼륨은 블록 디바이스로 표시되고 Windows에서는 이동식 디스크로 표시됩니다. 볼륨을 원하는 파일 시스템으로 포맷한 다음 마운트합니다. EBS 볼륨을 사용할 수 있게 만들면 다른 볼륨과 동일한 방식으로 액세스할 수 있습니다. 이 파일 시스템에 작성된 모든 데이터가 EBS 볼륨에 작성되고 해당 디바이스를 사용하는 애플리케이션도 그대로 적용됩니다.

다른 볼륨을 생성할 때 기준으로 사용하거나 백업을 목적으로 EBS 볼륨의 스냅샷을 생성할 수 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 \(p. 966\)](#) 단원을 참조하십시오.

Linux 인스턴스에서의 볼륨에 대한 지침은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux에서 볼륨을 사용 가능하게 만들기](#) 단원을 참조하십시오.

디스크 관리 유ти리티와 DiskPart 명령줄 도구를 사용하여 사용 가능한 EBS 볼륨을 만들 수 있습니다.

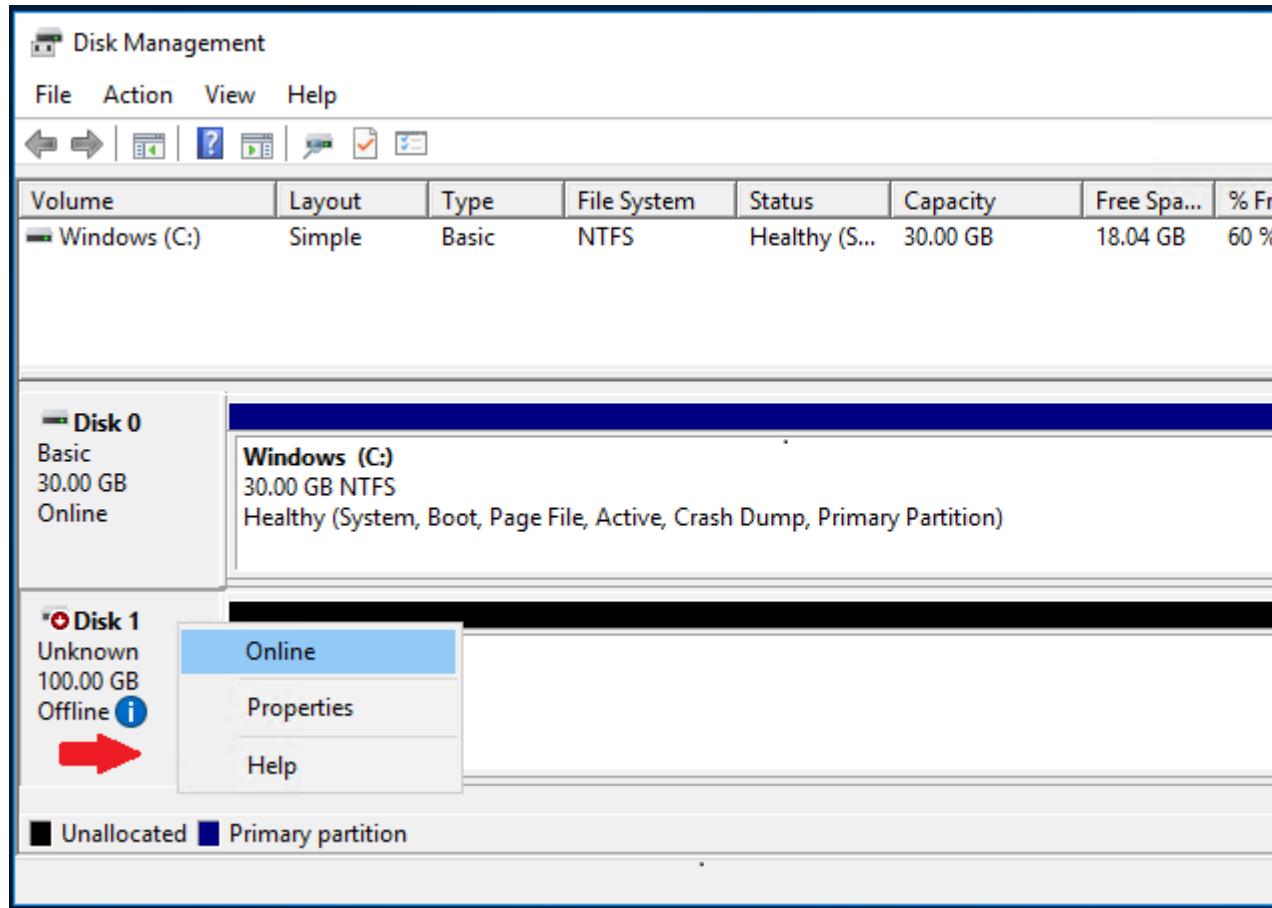
디스크 관리 유ти리티를 통해 사용 가능한 EBS 볼륨을 만들려면

1. 원격 데스크톱을 사용하여 Windows 인스턴스에 로그인합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
2. 디스크 관리 유ти리티를 시작합니다. 작업 표시줄에서 마우스 오른쪽 버튼을 클릭하여 Windows 로고에 대한 컨텍스트 메뉴를 열고 디스크 관리(Disk Management)를 선택합니다.

Note

Windows Server 2008에서는 시작(Start), 관리 도구(Administrative Tools), 컴퓨터 관리(Computer Management), 디스크 관리(Disk Management)를 선택합니다.

3. 볼륨을 온라인 상태로 전환합니다. 아래쪽의 왼쪽 창에서 마우스 오른쪽 버튼을 클릭하여 EBS 볼륨용 디스크에 대한 컨텍스트 메뉴를 엽니다. 온라인을 선택합니다.

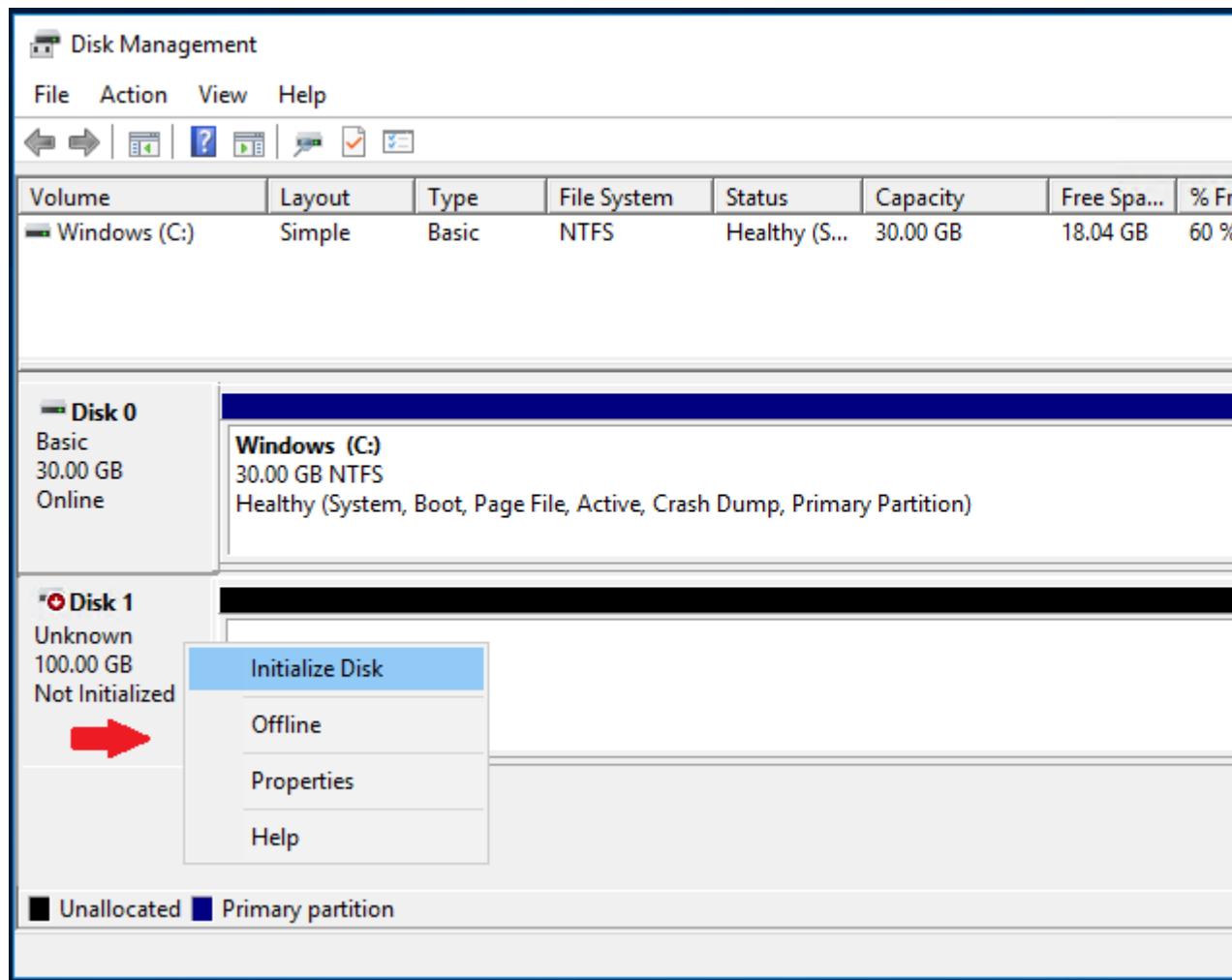


4. (조건 사항) 사용에 앞서 먼저 디스크를 초기화해야 합니다.

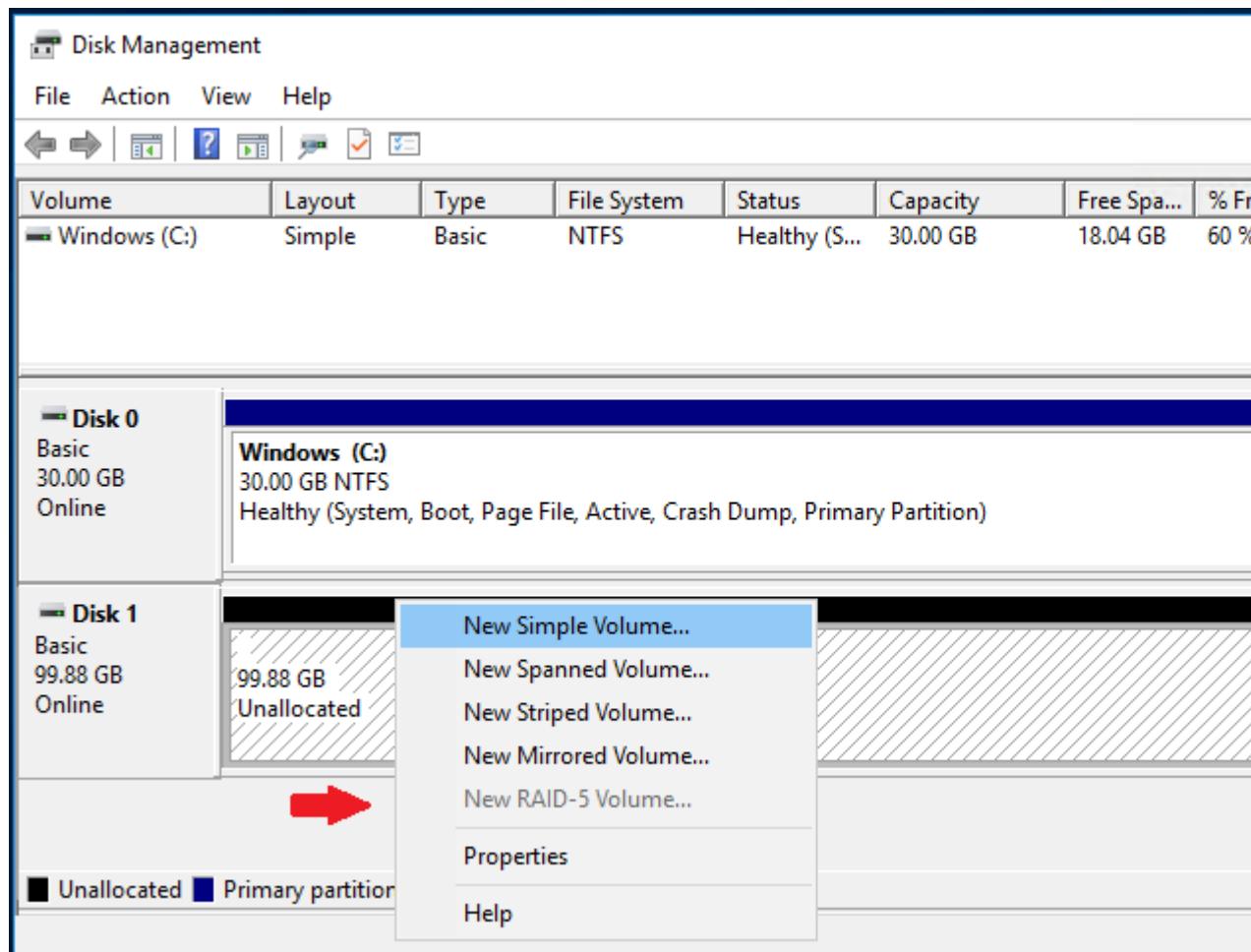
Warning

이미 데이터가 있는 볼륨을 마운트하는 경우(예: 퍼블릭 데이터 세트 또는 스냅샷에서 생성된 볼륨), 볼륨을 다시 포맷하지 말고 기존 데이터를 삭제하십시오.

디스크가 초기화되어 있지 않으면 다음과 같이 초기화를 수행하십시오. 왼쪽 창에서 마우스 오른쪽 버튼을 클릭하여 디스크에 대한 컨텍스트 메뉴를 열고 디스크 초기화(Initialize Disk)를 선택합니다. 디스크 초기화(Initialize Disk) 대화 상자에서 파티션 스타일을 선택하고 확인(OK)을 선택합니다.



5. 오른쪽 창에서 마우스 오른쪽 버튼을 클릭하여 디스크에 대한 컨텍스트 메뉴를 열고 새 단순 볼륨(New Simple Volume)을 선택합니다. 마법사를 완료합니다.



DiskPart 명령줄 도구를 통해 사용 가능한 EBS 볼륨을 만들려면

- 원격 데스크톱을 사용하여 Windows 인스턴스에 로그인합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
- diskpart.txt라는 이름의 새로운 스크립트 파일을 만듭니다.
- 스크립트 파일에 다음 명령을 추가하고 볼륨 레벨 및 드라이브 문자를 지정합니다. 이 스크립트는 볼륨이 마스터 부트 레코드(MBR) 파티션 구조를 사용하도록 구성하고 볼륨 형식을 NTFS 볼륨으로 지정하고 볼륨 레벨을 설정하고 볼륨에 드라이브 문자를 할당합니다.

Warning

이미 데이터가 있는 볼륨을 마운트하는 경우 볼륨을 재포맷하지 않아야 기존 데이터가 삭제되지 않습니다.

```
select disk 1
attributes disk clear readonly
online disk
convert mbr
create partition primary
format quick fs=ntfs label="volume_label"
assign letter="drive_letter"
```

자세한 내용은 [DiskPart 구문 및 파라미터](#)를 참조하십시오.

4. 스크립트가 있는 폴더로 이동하여 다음 명령을 실행합니다.

```
C:\> diskpart /s diskpart.txt
```

Amazon EBS 볼륨에 대한 정보 보기

EBS 볼륨에 대한 설명이 포함된 정보를 볼 수 있습니다. 예를 들어 특정 리전에 있는 모든 볼륨에 대한 정보를 보거나 크기, 볼륨 유형, 볼륨 암호화 여부, 볼륨 암호화에 사용된 마스터 키, 볼륨이 연결된 특정 인스턴스 등 단일 볼륨에 대한 자세한 정보를 볼 수 있습니다.

사용 가능한 디스크 공간 등 EBS 볼륨에 대한 추가 정보를 인스턴스의 운영 체제에서 가져올 수 있습니다.

볼륨 정보 보기

콘솔을 사용하여 EBS 볼륨에 대한 정보를 확인하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 볼륨을 선택합니다.
3. (선택 사항) 검색 창의 필터 옵션을 사용하여 원하는 볼륨만 표시합니다. 예를 들어 볼륨을 보려는 인스턴스 ID를 알고 있는 경우 검색 창으로 이동하여 필터 메뉴에서 인스턴스 ID를 선택한 다음 제공된 목록에서 원하는 인스턴스 ID를 선택합니다. 필터를 제거하려면 필터를 다시 선택합니다.
4. 볼륨에 대한 자세한 정보를 확인하려면 선택합니다.
5. 세부 정보 창에서 볼륨에 대해 제공된 정보를 검사할 수 있습니다. 연결 정보에는 이 볼륨이 연결된 인스턴스 ID와 해당 볼륨이 연결된 디바이스 이름이 표시됩니다.
6. (선택 사항) 인스턴스 세부 정보를 보려면 연결 정보 링크를 선택합니다.

콘솔을 사용하여 인스턴스에 연결된 EBS 볼륨을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스에 대한 자세한 정보를 확인하려면 선택합니다.
4. 세부 정보 창에서 루트 및 블록 디바이스에 대해 제공된 정보를 검사할 수 있습니다. 디바이스 이름 링크를 선택하여 디바이스 이름 아래에 연결된 볼륨에 대한 정보를 표시합니다.
5. (선택 사항) 이전 단계에서 선택한 디바이스 이름으로 이 인스턴스에 연결된 볼륨의 세부 정보를 보려면 EBS ID 링크를 선택합니다.

명령줄을 사용하여 EBS 볼륨에 대한 정보를 확인하려면

다음 명령 중 하나를 사용하여 볼륨 속성을 볼 수 있습니다. 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-volumes\(AWS CLI\)](#)
- [Get-EC2Volume \(Windows PowerShell용 AWS 도구\)](#)

볼륨 상태

볼륨 상태는 Amazon EBS 볼륨의 가용성을 설명합니다. 볼륨 상태는 콘솔의 볼륨 페이지에 있는 상태 열에서 확인하거나 [describe-volumes AWS CLI](#) 명령을 사용하여 확인할 수 있습니다.

가능한 볼륨 상태는 다음과 같습니다.

생성

볼륨이 생성되고 있습니다.

사용 가능

볼륨이 인스턴스에 연결되어 있지 않습니다.

사용 중

볼륨이 인스턴스에 연결되어 있습니다.

삭제

볼륨이 삭제 중입니다.

삭제됨

볼륨이 삭제되었습니다.

오류

EBS 볼륨과 관련된 기본 하드웨어에 장애가 발생하여 볼륨과 연결된 데이터를 복구할 수 없습니다. 볼륨을 복원하거나 볼륨의 데이터를 복구하는 방법에 대한 자세한 내용은 [내 EBS 볼륨이 “오류” 상태입니다](#)를 참조하십시오.

볼륨 지표 보기

Amazon CloudWatch에서 EBS 볼륨에 대한 추가 정보를 얻을 수 있습니다. 자세한 내용은 [Amazon EBS에 대한 Amazon CloudWatch 지표 \(p. 1076\)](#) 단원을 참조하십시오.

여유 디스크 공간 보기

사용 가능한 디스크 공간 등 EBS 볼륨에 대한 추가 정보를 인스턴스의 Windows 운영 체제에서 가져올 수 있습니다. 예를 들어 파일 탐색기를 열어 이 PC(This PC)를 선택하면 여유 디스크 공간을 볼 수 있습니다.

다음과 같이 dir 명령을 사용하여 출력된 내용의 마지막 행에서 여유 디스크 공간을 확인할 수도 있습니다.

```
C:\> dir C:  
Volume in drive C has no label.  
Volume Serial Number is 68C3-8081  
  
Directory of C:\  
  
03/25/2018  02:10 AM    <DIR>      .  
03/25/2018  02:10 AM    <DIR>      ..  
03/25/2018  03:47 AM    <DIR>      Contacts  
03/25/2018  03:47 AM    <DIR>      Desktop  
03/25/2018  03:47 AM    <DIR>      Documents  
03/25/2018  03:47 AM    <DIR>      Downloads  
03/25/2018  03:47 AM    <DIR>      Favorites  
03/25/2018  03:47 AM    <DIR>      Links  
03/25/2018  03:47 AM    <DIR>      Music  
03/25/2018  03:47 AM    <DIR>      Pictures  
03/25/2018  03:47 AM    <DIR>      Saved Games  
03/25/2018  03:47 AM    <DIR>      Searches  
03/25/2018  03:47 AM    <DIR>      Videos  
          0 File(s)           0 bytes  
       13 Dir(s)  18,113,662,976 bytes free
```

다음과 같이 `fsutil` 명령을 사용해 여유 디스크 공간을 확인할 수도 있습니다.

```
C:\> fsutil volume diskfree C:  
Total # of free bytes : 18113204224  
Total # of bytes : 32210153472  
Total # of avail free bytes : 18113204224
```

이전 스냅샷을 사용하여 Amazon EBS 볼륨 바꾸기

Amazon EBS 스냅샷은 속도, 편리성 및 비용으로 인해 Amazon EC2에서 선호하는 백업 도구입니다. 스냅샷에서 볼륨을 생성할 경우 모든 데이터를 그대로 유지한 상태로 과거의 특정 시점에서 해당 상태를 재생성합니다. 스냅샷에서 생성된 볼륨을 인스턴스에 연결하면 여러 리전에서 데이터를 복제하고 테스트 환경을 생성하며 손상된 프로덕션 볼륨 전체를 바꾸거나 특정 파일 및 디렉터리를 검색하여 연결된 다른 볼륨으로 전송할 수 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 \(p. 966\)](#) 단원을 참조하십시오.

다음 절차를 사용하여, EBS 볼륨을 해당 볼륨의 이전 스냅샷에서 생성된 다른 볼륨으로 바꿀 수 있습니다. 현재 볼륨을 분리한 다음 새 볼륨을 연결해야 합니다.

동일한 가용 영역의 EC2 인스턴스에만 EBS 볼륨을 연결할 수 있습니다.

볼륨을 교체하려면

1. 스냅샷에서 볼륨을 생성하고 새 볼륨의 ID를 기록합니다. 자세한 내용은 [스냅샷에서 볼륨 생성 \(p. 949\)](#) 단원을 참조하십시오.
2. 볼륨 페이지에서 교체할 볼륨의 확인란을 선택합니다. 설명 탭에서 연결 정보를 찾아 볼륨의 디바이스 이름(예: /dev/sda1이나 /dev/xvda(루트 볼륨의 경우), /dev/sdb 또는 xvdb)과 인스턴스의 ID를 기록합니다.
3. (선택 사항) 인스턴스의 루트 볼륨을 분리하려면 먼저 인스턴스를 중지해야 합니다. 루트 볼륨을 교체하지 않을 경우 인스턴스를 중단하지 않고 다음 단계를 계속할 수 있습니다. 인스턴스를 중지하려면 연결 정보에서 인스턴스 ID 위에 마우스를 올려 놓고 마우스 오른쪽 버튼을 클릭한 다음 새 브라우저 탭에서 인스턴스를 엽니다. [Actions], [Instance State], [Stop]을 차례로 선택합니다. 인스턴스 페이지가 열려 있는 탭을 그대로 두고 볼륨 페이지가 있는 브라우저 탭으로 돌아갑니다.
4. 해당 볼륨을 선택한 상태에서 작업, 볼륨 분리를 선택합니다. 확인 메시지가 나타나면 예, 분리를 선택합니다. 이 볼륨의 확인란을 선택 취소합니다.
5. 1단계에서 만든 새 볼륨의 확인란을 선택합니다. 작업, 볼륨 연결을 선택합니다. 2단계에서 적어 둔 인스턴스 ID 및 디바이스 이름을 입력한 다음 연결을 선택합니다.
6. (선택 사항) 인스턴스를 중지한 경우 인스턴스를 다시 시작해야 합니다. 인스턴스 페이지가 있는 브라우저 탭으로 돌아가서 작업, 인스턴스 상태, 시작을 선택합니다.
7. 인스턴스에 연결하고 볼륨을 탑재합니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오.

볼륨 상태 모니터링

Amazon Web Services(AWS)에서는 Amazon Elastic Block Store(Amazon EBS) 볼륨을 모니터링하는 데 사용할 수 있는 데이터를 자동으로 제공합니다.

목차

- [EBS 볼륨 상태 확인 \(p. 958\)](#)
- [EBS 볼륨 이벤트 \(p. 959\)](#)
- [손상된 볼륨 작업 \(p. 961\)](#)
- [자동 활성화된 IO 볼륨 속성 작업 \(p. 963\)](#)

모니터링에 대한 자세한 내용은 [Amazon EBS에 대한 Amazon CloudWatch 지표 \(p. 1076\)](#) 및 [Amazon EBS 용 Amazon CloudWatch Events \(p. 1081\)](#) 단원을 참조하십시오.

EBS 볼륨 상태 확인

볼륨 상태 확인을 사용하여 Amazon EBS 볼륨에 있는 데이터의 잠재적 불일치를 더 잘 파악, 추적 및 관리할 수 있습니다. 볼륨 상태 확인은 Amazon EBS 볼륨이 손상되었는지 여부를 확인하는 데 필요한 정보를 제공하며, 잠재적으로 일치하지 않는 볼륨을 처리하는 방법을 제어하는 데 도움이 됩니다.

볼륨 상태 확인은 5분마다 테스트를 자동으로 실행하여 통과 또는 실패 상태를 반환합니다. 모든 확인을 통과한 경우 볼륨의 상태는 `ok`이고, 확인에 실패한 경우 볼륨의 상태는 `impaired`입니다. 상태가 `insufficient-data`인 경우 볼륨에 대한 확인이 아직 진행 중일 수 있습니다. 볼륨 상태 확인의 결과를 보고 손상된 볼륨을 식별하고 필요한 조치를 취할 수 있습니다.

Amazon EBS에서 볼륨의 데이터가 잠재적으로 일치하지 않는 것으로 확인하면 데이터 손상을 방지하기 위해 기본적으로 연결된 EC2 인스턴스에서 볼륨으로의 I/O가 비활성화됩니다. I/O가 비활성화되면 다음 볼륨 상태 확인에 실패하고 볼륨 상태는 `impaired`가 됩니다. 또한 I/O가 비활성화되었으며 볼륨에 대한 I/O를 활성화하여 볼륨의 손상된 상태를 해결할 수 있다고 알려주는 이벤트가 표시됩니다. I/O를 활성화한 다음 인스턴스에서 볼륨을 계속 사용할지 아니면 `chkdsk`와 같은 명령을 사용하여 일관성 확인을 실행한 다음 볼륨을 사용할지 여부를 결정할 수 있습니다.

Note

볼륨 상태는 볼륨 상태 검사 결과를 기준으로 한 것으로, 볼륨 상태를 직접 반영하는 것은 아닙니다. 따라서 볼륨 상태가 `error` 상태의 볼륨을 나타내는 것은 아닙니다(예: 볼륨이 I/O를 허용할 수 없을 때). 볼륨 상태에 대한 내용은 [볼륨 상태 \(p. 955\)](#) 단원을 참조하십시오.

특정 볼륨의 일관성은 문제가 아니고, 볼륨이 손상된 경우 볼륨을 즉시 사용할 수 있게 하려면 I/O를 자동으로 활성화하도록 볼륨을 구성하여 기본 동작을 무시할 수 있습니다. IO 자동 활성화 볼륨 속성(API의 `autoEnableIO`)을 활성화하면 볼륨 상태 검사를 계속 통과합니다. 또한 볼륨이 잠재적으로 일치하지 않는 것으로 확인되었지만 I/O가 자동으로 활성화되었다고 알려주는 이벤트가 표시됩니다. 그러면 볼륨의 일관성을 확인하거나 나중에 볼륨을 교체할 수 있습니다.

I/O 성능 상태 확인은 실제 볼륨 성능을 볼륨의 예상 성능과 비교하고 볼륨 성능이 예상보다 낮은 경우 알림을 표시합니다. 이 상태 확인은 인스턴스에 연결된 프로비저닝된 IOPS SSD(`io1` 및 `io2`) 볼륨에 대해서만 사용할 수 있습니다. 범용 SSD(`gp2`), 처리량에 최적화된 HDD(`st1`), Cold HDD(`sc1`) 또는 Magnetic(`standard`) 볼륨에는 유효하지 않습니다. I/O 성능 상태 확인은 1분마다 수행되고 CloudWatch에서 이 데이터를 5분마다 수집하므로 `io1` 또는 `io2` 볼륨을 인스턴스에 연결한 후 이 확인에서 I/O 성능 상태를 보고하는 데 최대 5분 정도 걸릴 수 있습니다.

Important

스냅샷에서 복원한 `io1` 및 `io2` 볼륨을 초기화할 경우 볼륨의 성능이 예상 수준보다 50퍼센트 이하로 떨어질 수 있으며, 이로 인해 볼륨에서 I/O 성능 상태 확인에 대해 `warning` 상태를 표시할 수 있습니다. 이는 원래 그런 것이므로 초기화 중에는 `io1` 및 `io2` 볼륨에 대한 `warning` 상태를 무시해도 됩니다. 자세한 내용은 [Amazon EBS 볼륨 초기화 \(p. 1066\)](#) 단원을 참조하십시오.

다음 표에는 Amazon EBS 볼륨에 대한 상태가 나와 있습니다.

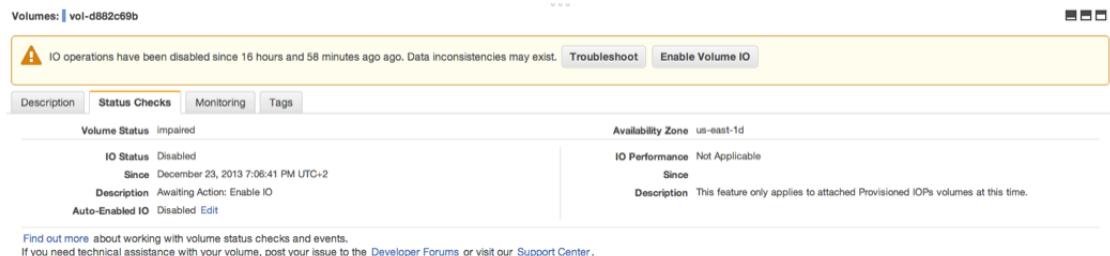
볼륨 상태	I/O 활성화 상태	I/O 성능 상태(프로비저닝된 IOPS 볼륨에만 사용 가능)
<code>ok</code>	활성화됨(I/O 활성화 또는 I/O 자동 활성화)	정상(볼륨 성능이 예상대로임)
<code>warning</code>	활성화됨(I/O 활성화 또는 I/O 자동 활성화)	성능 저하(볼륨 성능이 예상보다 낮음)

볼륨 상태	I/O 활성화 상태	I/O 성능 상태(프로비저닝된 IOPS 볼륨에만 사용 가능)
		심각한 성능 저하(볼륨 성능이 예상보다 훨씬 낮음)
impaired	활성화됨(I/O 활성화 또는 I/O 자동 활성화) 비활성화됨(볼륨이 오프라인이고 복구 보류 중이거나 사용자가 I/O를 활성화하기를 기다리는 중)	중단됨(볼륨 성능이 저하됨) 사용할 수 없음(I/O가 비활성화되어 I/O 성능을 확인할 수 없음)
insufficient-data	활성화됨(I/O 활성화 또는 I/O 자동 활성화) 데이터 부족	데이터 부족

상태 확인을 보면서 작업하려면 Amazon EC2 콘솔, API 또는 명령줄 인터페이스를 사용합니다.

콘솔에서 상태 확인을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 볼륨을 선택합니다. 볼륨 상태 열에 각 볼륨의 작업 상태가 나열됩니다.
3. 볼륨의 상태 세부 정보를 보려면 볼륨을 선택하고 상태 검사를 선택합니다.



4. 상태 검사에 실패한 볼륨이 있는 경우(상태가 손상됨) [손상된 볼륨 작업 \(p. 961\)](#) 단원을 참조하십시오.

또는 탐색기에서 이벤트 창을 선택하여 인스턴스와 볼륨에 대한 모든 이벤트를 볼 수 있습니다. 자세한 내용은 [EBS 볼륨 이벤트 \(p. 959\)](#) 단원을 참조하십시오.

명령줄로 볼륨 상태 정보를 보려면

다음 명령 중 하나를 사용하여 Amazon EBS 볼륨의 상태를 볼 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-volume-status\(AWS CLI\)](#)
- [Get-EC2VolumeStatus\(Windows PowerShell용 AWS 도구\)](#)

EBS 볼륨 이벤트

Amazon EBS에서 볼륨의 데이터가 잠재적으로 일치하지 않는 것으로 확인하면 기본적으로 연결된 EC2 인스턴스에서 볼륨으로의 I/O가 비활성화됩니다. 그러면 볼륨 상태 확인에 실패하고 실패의 원인을 나타내는 볼륨 상태 이벤트가 생성됩니다.

데이터가 잠재적으로 일치하지 않는 볼륨에서 I/O를 자동으로 활성화하려면 IO 자동 활성화 볼륨 속성(API의 `autoEnableIO`)의 설정을 변경합니다. 이 속성 변경에 대한 자세한 내용은 [손상된 볼륨 작업 \(p. 961\)](#) 단원을 참조하십시오.

각 이벤트에는 이벤트가 발생한 시간을 나타내는 시작 시간과 볼륨에 대한 I/O가 비활성화된 시간을 나타내는 기간이 포함됩니다. 볼륨에 대한 I/O가 활성화되면 이벤트에 종료 시간이 추가됩니다.

볼륨 상태 이벤트는 다음 설명 중 하나를 포함합니다.

Awaiting Action: Enable IO

볼륨 데이터가 잠재적으로 일치하지 않습니다. 사용자가 명시적으로 활성화할 때까지 볼륨에 대해 I/O가 비활성화됩니다. I/O를 명시적으로 활성화하면 이벤트 설명이 IO Enabled로 변경됩니다.

IO Enabled

이 볼륨에 대해 I/O 작업이 명시적으로 활성화되었습니다.

IO Auto-Enabled

이벤트가 발생한 후 이 볼륨에서 I/O 작업이 자동으로 활성화되었습니다. 데이터를 계속 사용하려면 먼저 데이터 불일치를 확인하는 것이 좋습니다.

보통

io1 및 io2 볼륨 전용입니다. 볼륨 성능이 예상대로입니다.

성능 저하

io1 및 io2 볼륨 전용입니다. 볼륨 성능이 예상보다 낮습니다.

Severely Degraded

io1 및 io2 볼륨 전용입니다. 볼륨 성능이 예상보다 훨씬 낮습니다.

Stalled

io1 및 io2 볼륨 전용입니다. 볼륨 성능이 저하되었습니다.

Amazon EC2 콘솔, API 또는 명령줄 인터페이스를 사용하여 볼륨에 대한 이벤트를 볼 수 있습니다.

콘솔에서 볼륨에 대한 이벤트를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 이벤트를 선택합니다. 이벤트가 있는 모든 인스턴스와 볼륨이 나열됩니다.
3. 볼륨을 기준으로 필터링하여 볼륨 상태만 볼 수 있습니다. 특정 상태 유형을 기준으로 필터링할 수도 있습니다.
4. 특정 이벤트를 보려는 볼륨을 선택합니다.

Resource Name	Resource Type	Resource Id	Availability Zone	Event Type	Event Description	Event Status	Start Time	Duration	Event Progress
volume	volume	vol-0381c540	us-east-1d	potential-data-i...	Awaiting Action...	⚠️ Awaiting A...	December 23, ...	30 days, 15 ho...	IO Disabled
volume	volume	vol-3682c675	us-east-1d	potential-data-i...	Awaiting Action...	⚠️ Awaiting A...	December 23, ...	30 days, 15 ho...	IO Disabled

Event: vol-3682c675

⚠️ IO operations have been disabled since 30 days, 15 hours and 22 minutes ago. Data inconsistencies may exist. [Enable Volume IO](#)

Availability Zone	us-east-1d
Event Type	potential-data-inconsistency
Event Status	Awaiting Action: Enable IO
IO status	IO Disabled
Attached to	i-93aae4ea
Start Time	December 23, 2013 7:09:20 PM UTC+2
End time	

Find out more about [monitoring volume events](#).

I/O 가 비활성화된 볼륨이 있는 경우 [손상된 볼륨 작업 \(p. 961\)](#) 단원을 참조하십시오. I/O 성능이 정상보다 낮은 볼륨이 있는 경우 수행한 작업(예: 피크 사용 동안 볼륨 스냅샷 생성, 필요한 I/O 대역폭을 지원할 수 없는 인스턴스에서 볼륨 실행, 볼륨의 데이터에 처음 액세스 등)으로 인한 일시적인 현상일 수 있습니다.

명령줄로 볼륨에 대한 이벤트를 보려면

다음 명령 중 하나를 사용하여 Amazon EBS 볼륨에 대한 이벤트 정보를 볼 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-volume-status\(AWS CLI\)](#)
- [Get-EC2VolumeStatus\(Windows PowerShell용 AWS 도구\)](#)

손상된 볼륨 작업

볼륨의 데이터가 잠재적으로 일치하지 않아서 볼륨이 손상된 경우 다음 옵션을 사용합니다.

옵션

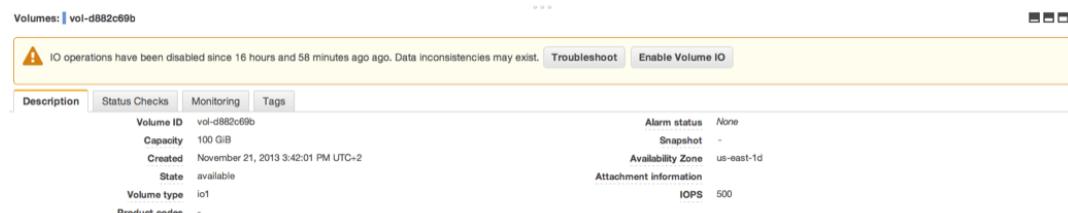
- [옵션 1: 인스턴스에 연결된 볼륨에 대한 일관성 확인 수행 \(p. 961\)](#)
- [옵션 2: 다른 인스턴스를 사용하여 볼륨에 대한 일관성 확인 수행 \(p. 962\)](#)
- [옵션 3: 볼륨이 더 이상 필요하지 않은 경우 볼륨 삭제 \(p. 962\)](#)

옵션 1: 인스턴스에 연결된 볼륨에 대한 일관성 확인 수행

가장 간단한 옵션은 볼륨이 Amazon EC2 인스턴스에 연결된 상태에서 I/O를 활성화한 다음 볼륨에 대한 데이터 일관성 확인을 수행하는 것입니다.

연결된 볼륨에 대해 일관성 확인을 수행하려면

1. 모든 애플리케이션의 볼륨 사용을 중지합니다.
2. 볼륨에서 I/O를 활성화합니다.
 - a. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
 - b. 탐색 창에서 볼륨을 선택합니다.
 - c. I/O 작업을 활성화할 볼륨을 선택합니다.
 - d. 세부 정보 창에서 볼륨 IO 활성화를 선택한 다음, 예, 활성화를 선택합니다.



3. 볼륨의 데이터를 확인합니다.

- a. chkdsk 명령을 실행합니다.
- b. (선택 사항) 애플리케이션 또는 시스템 로그에 관련 오류 메시지가 있는지 검토합니다.
- c. 볼륨 손상 상태가 20분 이상 지속된 경우 AWS 지원 센터에 문의할 수 있습니다. 문제 해결을 선택한 다음 상태 검사 문제 해결 대화 상자에서 고객 지원을 선택하여 지원 사례를 제출합니다.

명령줄로 볼륨에 대한 I/O를 활성화하려면

다음 명령 중 하나를 사용하여 Amazon EBS 볼륨에 대한 이벤트 정보를 볼 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [enable-volume-io\(AWS CLI\)](#)
- [Enable-EC2VolumeIO\(Windows PowerShell용 AWS 도구\)](#)

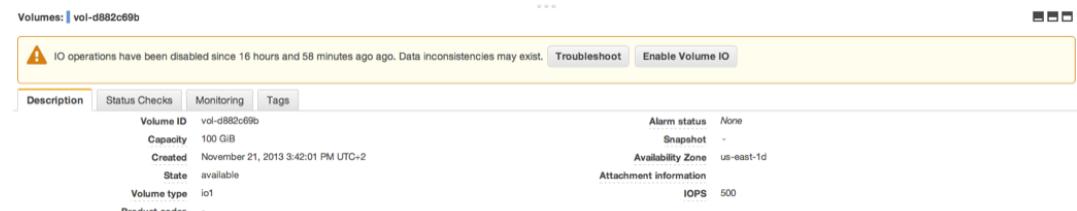
옵션 2: 다른 인스턴스를 사용하여 볼륨에 대한 일관성 확인 수행

다음 절차에 따라 프로덕션 환경 외부의 볼륨을 확인합니다.

Important

이 절차를 수행하면 볼륨 I/O가 비활성화된 상태에서 일시 중지된 쓰기 I/O가 손실될 수 있습니다.

격리 중인 볼륨에 대한 일관성 확인을 수행하려면

1. 모든 애플리케이션의 볼륨 사용을 중지합니다.
 2. 인스턴스에서 볼륨을 분리합니다.
 - a. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
 - b. 탐색 창에서 볼륨을 선택합니다.
 - c. 분리할 볼륨을 선택합니다.
 - d. 작업, 볼륨 강제 분리를 선택합니다. 확인 메시지가 나타납니다.
 3. 볼륨에서 I/O를 활성화합니다.
 - a. 탐색 창에서 볼륨을 선택합니다.
 - b. 이전 단계에서 분리한 볼륨을 선택합니다.
 - c. 세부 정보 창에서 볼륨 IO 활성화를 선택한 다음, 예, 활성화를 선택합니다.
- 
- The screenshot shows the 'Volumes' section of the EC2 console. A specific volume, 'vol-d882c69b', is selected. A prominent yellow warning box at the top states: 'IO operations have been disabled since 16 hours and 58 minutes ago ago. Data inconsistencies may exist.' It includes three buttons: 'Troubleshoot', 'Enable Volume IO', and a close button. Below the warning, there are tabs for 'Description', 'Status Checks', 'Monitoring', and 'Tags'. The 'Description' tab is active, showing details like Volume ID, Capacity (100 GiB), Created (November 21, 2013 3:42:01 PM UTC-2), State (available), Volume type (io1), and Product codes. To the right, there are sections for Alarm status (None), Snapshot (-), Availability Zone (us-east-1d), Attachment information (IOPS 500), and a 'Delete' button.
4. 볼륨을 다른 인스턴스에 연결합니다. 자세한 내용은 [인스턴스 시작 \(p. 375\)](#) 및 [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#) 단원을 참조하십시오.
 5. 볼륨의 데이터를 확인합니다.
 - a. chkdsk 명령을 실행합니다.
 - b. (선택 사항) 애플리케이션 또는 시스템 로그에 관련 오류 메시지가 있는지 검토합니다.
 - c. 볼륨 손상 상태가 20분 이상 지속된 경우 AWS 지원 센터에 문의할 수 있습니다. 문제 해결을 선택하고 문제 해결 대화 상자에서 고객 지원을 선택하여 지원 사례를 제출합니다.

명령줄로 볼륨에 대한 I/O를 활성화하려면

다음 명령 중 하나를 사용하여 Amazon EBS 볼륨에 대한 이벤트 정보를 볼 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [enable-volume-io\(AWS CLI\)](#)
- [Enable-EC2VolumeIO\(Windows PowerShell용 AWS 도구\)](#)

옵션 3: 볼륨이 더 이상 필요하지 않은 경우 볼륨 삭제

환경에서 볼륨을 제거하려면 볼륨을 삭제하면 됩니다. 볼륨 삭제에 대한 자세한 내용은 [Amazon EBS 볼륨 삭제 \(p. 966\)](#) 단원을 참조하십시오.

볼륨의 데이터를 백업하는 최근 스냅샷이 있는 경우 해당 스냅샷에서 새 볼륨을 생성할 수 있습니다. 자세한 내용은 [스냅샷에서 볼륨 생성 \(p. 949\)](#) 단원을 참조하십시오.

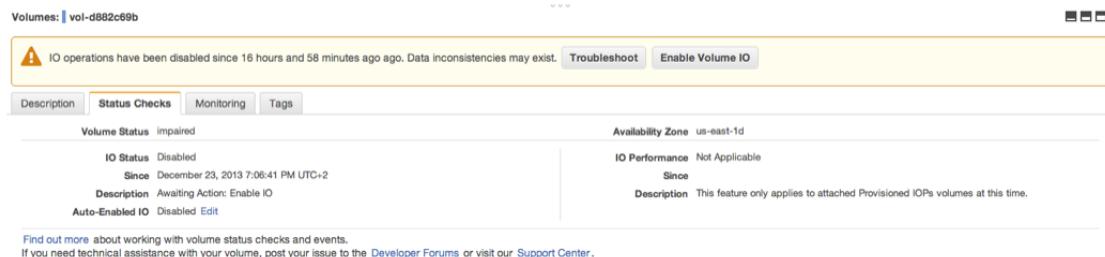
자동 활성화된 IO 볼륨 속성 작업

Amazon EBS에서 볼륨의 데이터가 잠재적으로 일치하지 않는 것으로 확인하면 기본적으로 연결된 EC2 인스턴스에서 볼륨으로의 I/O가 비활성화됩니다. 그러면 볼륨 상태 확인에 실패하고 실패의 원인을 나타내는 볼륨 상태 이벤트가 생성됩니다. 특정 볼륨의 일관성은 문제가 아니고, 볼륨이 손상된 상태인 경우 볼륨을 즉시 사용할 수 있게 하려면 I/O를 자동으로 활성화하도록 볼륨을 구성하여 기본 동작을 재정의할 수 있습니다. 자동 활성화된 IO 볼륨 속성(API의 `autoEnableIO`)을 활성화하면 볼륨과 인스턴스 사이의 I/O가 자동으로 다시 활성화되고 볼륨 상태 확인을 통과합니다. 또한 볼륨이 잠재적으로 일치하지 않는 상태인 것으로 결정되었지만 I/O가 자동으로 활성화되었다고 알려주는 이벤트가 표시됩니다. 이 이벤트가 발생하면 볼륨의 일관성을 확인하고 필요한 경우 볼륨을 교체해야 합니다. 자세한 내용은 [EBS 볼륨 이벤트 \(p. 959\)](#) 단원을 참조하십시오.

이 절차에서는 볼륨의 자동 활성화된 IO 속성을 보고 수정하는 방법을 설명합니다.

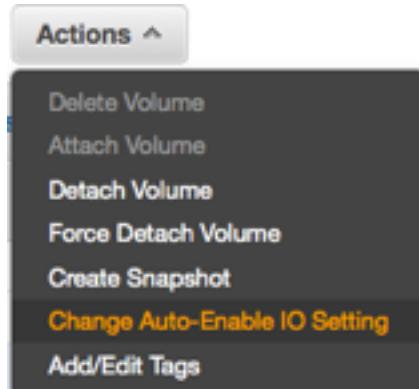
콘솔에서 볼륨의 자동 활성화된 IO 속성을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 볼륨을 선택합니다.
3. 볼륨을 선택하고 상태 검사를 선택합니다. 자동 활성화된 IO는 볼륨에 대한 현재의 설정(활성화 또는 비활성화)을 표시합니다.

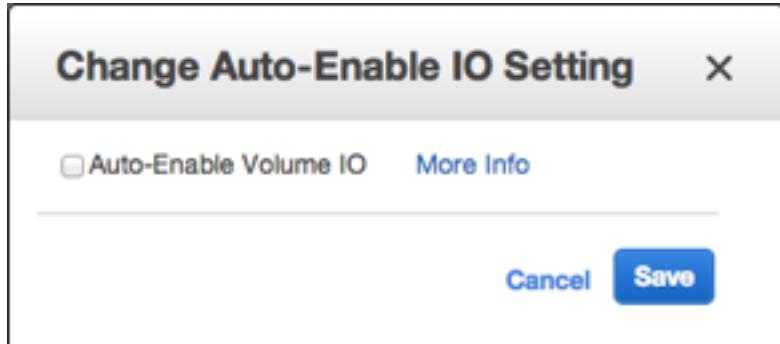


콘솔에서 볼륨의 자동 활성화된 IO 속성을 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 볼륨을 선택합니다.
3. 볼륨을 선택하고 작업, IO 자동 활성화 설정 변경을 선택합니다. 또는 상태 검사 탭을 선택하고 자동 활성화된 IO에서 편집을 선택합니다.



4. 볼륨 IO 자동 활성화 확인란을 선택하여 손상된 볼륨에 대한 I/O를 자동으로 활성화합니다. 이 기능을 비활성화하려면 확인란의 선택을 취소합니다.



5. Save를 선택합니다.

명령줄로 볼륨의 AutoEnableIO 속성을 보거나 수정하려면

다음 명령 중 하나를 사용하여 Amazon EBS 볼륨의 autoEnableIO 속성을 볼 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-volume-attribute\(AWS CLI\)](#)
- [Get-EC2VolumeAttribute\(Windows PowerShell용 AWS 도구\)](#)

볼륨의 autoEnableIO 속성을 수정하려면 다음 명령 중 하나를 사용합니다.

- [modify-volume-attribute\(AWS CLI\)](#)
- [Edit-EC2VolumeAttribute\(Windows PowerShell용 AWS 도구\)](#)

Windows 인스턴스에서 Amazon EBS 볼륨 분리

인스턴스에서 Amazon EBS 볼륨을 분리하거나 인스턴스를 종료하는 것이 가능합니다. 그러나 인스턴스가 실행 중인 경우 인스턴스에서 먼저 해당 볼륨의 마운트를 해제해야 합니다.

EBS 볼륨이 인스턴스의 루트 디바이스인 경우에는 볼륨을 분리하기 전에 인스턴스를 중지해야 합니다.

AWS Marketplace 제품 코드가 있는 볼륨을 인스턴스에서 분리하면 제품 코드는 더 이상 인스턴스와 연결되어 있지 않습니다.

Important

볼륨을 해제한 이후에도 스토리지 용량이 AWS 프리 티어 한도를 초과할 경우 볼륨 스토리지에 대해 비용이 계속해서 청구됩니다. 추가 비용이 청구되지 않도록 하려면 볼륨을 삭제해야 합니다. 자세한 내용은 [Amazon EBS 볼륨 삭제 \(p. 966\)](#) 단원을 참조하십시오.

이 예제에서는 볼륨의 마운트를 해제한 다음 인스턴스에서 명시적으로 분리합니다. 인스턴스를 종료하거나 볼륨을 다른 인스턴스에 연결하려고 할 때 이 예제가 유용하게 사용될 수 있습니다. 볼륨이 인스턴스에 더 이상 연결되어 있지 않은지를 확인하려면 [Amazon EBS 볼륨에 대한 정보 보기 \(p. 955\)](#) 단원을 참조하십시오.

분리된(탑재를 해제하지 않고) 볼륨을 다시 연결할 수 있지만 동일 탑재 지점을 가져올 수는 없습니다. 분리된 상태에서 진행 중인 볼륨 쓰기 작업이 있으면 볼륨의 데이터가 동기화되지 않을 수 있습니다.

Linux 인스턴스의 볼륨에 대한 지침은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux 인스턴스에서 볼륨 분리](#)를 참조하십시오.

콘솔을 이용하여 EBS 볼륨을 분리하려면

1. Windows 인스턴스에서 볼륨을 마운트 해제하십시오.

- a. 원격 데스크톱을 사용하여 Windows 인스턴스에 로그인합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
- b. 디스크 관리 유ти리티를 시작합니다.

Windows Server 2012 이상의 경우 작업 표시줄에서 Windows 로고를 마우스 오른쪽 단추를 클릭한 다음 Disk Management(디스크 관리)를 선택합니다. Windows Server 2008에서는 시작(Start), 관리 도구(Administrative Tools), 컴퓨터 관리(Computer Management), 디스크 관리(Disk Management)를 선택합니다.

- c. 디스크를 마우스 오른쪽 단추로 클릭하고(예: 디스크 1을 마우스 오른쪽 단추로 클릭) 오프라인을 선택하십시오. Amazon EC2 콘솔을 열기 전에 디스크 상태가 오프라인으로 변경될 때까지 기다리십시오.

2. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
3. 탐색 창에서 볼륨을 선택합니다.
4. 볼륨을 선택한 후 작업, 볼륨 분리를 선택합니다.
5. 확인 대화 상자에서 예, 분리를 선택합니다.

명령줄을 사용하여 인스턴스에서 EBS 볼륨을 분리하려면

볼륨의 탑재를 해제한 후 다음 명령 중 하나를 사용하여 볼륨을 분리할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [detach-volume](#)(AWS CLI)
- [Dismount-EC2Volume](#)(Windows PowerShell용 AWS 도구)

문제 해결

다음에서는 볼륨을 분리할 때 발생할 수 있는 일반적인 문제와 해결 방법에 대해 설명합니다.

Note

데이터 손실에 대비하여 볼륨을 해제하기 전 볼륨 스냅샷을 만들어 두십시오. 고착된 볼륨을 강제로 분리할 경우 파일 시스템 또는 여기에 포함된 데이터가 손상되거나 인스턴스를 재부팅 하지 않는 이상 동일한 디바이스 이름으로 새 볼륨을 연결할 수 없게 될 수 있습니다.

- Amazon EC2 콘솔을 통해 볼륨을 분리하는 동안 문제가 발생할 경우 `describe-volumes` CLI 명령을 사용하여 문제를 진단하는 것이 좋습니다. 자세한 내용은 [describe-volumes](#)를 참조하십시오.
- 볼륨이 `detaching` 상태를 유지하는 경우 강제 분리를 선택하여 강제 분리할 수 있습니다. 이 옵션은 오류가 발생한 인스턴스에서 볼륨 분리 또는 삭제할 목적으로 볼륨을 분리하는 경우에만 최후의 수단으로 사용하십시오. 인스턴스는 파일 시스템 캐시 또는 파일 시스템 메타데이터를 폴러시하지 않습니다. 이 옵션을 사용하는 경우 파일 시스템 확인 및 복구 절차를 수행해야 합니다.
- 몇 분 동안 강제 볼륨 분리를 수차례 시도하였지만 `detaching` 상태가 계속해서 유지되는 경우 [Amazon EC2 forum](#)에 도움을 요청하십시오. 해결 방법을 신속히 찾아내려면 볼륨 ID를 기재하고 어떤 단계를 수행했는지에 대해 설명하십시오.
- 아직 마운트되어 있는 볼륨을 분리하려는 경우 분리 시도 중에 볼륨이 `busy` 상태로 고착될 수 있습니다. 다음의 `describe-volumes` 출력 화면은 이 조건을 보여주는 예입니다.

```
aws ec2 describe-volumes --region us-west-2 --volume-ids vol-1234abcd
{
    "Volumes": [
        {
            "AvailabilityZone": "us-west-2b",
            "Attachments": [
                {
                    "AttachTime": "2016-07-21T23:44:52.000Z",
                    "InstanceId": "i-fedc9876",
                    "VolumeId": "vol-1234abcd"
                }
            ],
            "CreateTime": "2016-07-21T23:44:52.000Z",
            "Encrypted": false,
            "Iops": 1000,
            "KmsKeyId": null,
            "Lun": 1,
            "MountPoint": "/mnt/test",
            "SnapshotId": "snap-12345678",
            "Status": "in-use",
            "Type": "gp2"
        }
    ]
}
```

```
"VolumeId": "vol-1234abcd",
"State": "busy",
"DeleteOnTermination": false,
"Device": "/dev/sdf"
}
....
```

이 상태가 발생하면 볼륨의 마운트를 해제하거나 강제 분리하거나 인스턴스를 재부팅하거나 세 가지 조치를 모두 실행하기 전까지 분리가 무한히 지연될 수 있습니다.

Amazon EBS 볼륨 삭제

Amazon EBS 볼륨이 더 이상 필요하지 않으면 삭제할 수 있습니다. 볼륨을 삭제한 후에는 데이터가 사라지므로 해당 볼륨을 인스턴스에 연결할 수 없습니다. 그러나 삭제하기 전에 볼륨의 스냅샷을 저장하면 이 스냅샷을 사용하여 나중에 볼륨을 재생성할 수 있습니다.

Note

인스턴스에 연결된 볼륨은 삭제할 수 없습니다. 볼륨을 삭제하려면 먼저 볼륨을 분리해야 합니다. 자세한 내용은 [Windows 인스턴스에서 Amazon EBS 볼륨 분리 \(p. 964\)](#) 단원을 참조하십시오. 볼륨이 인스턴스에 연결되어 있는지 확인할 수 있습니다. 콘솔의 볼륨 페이지에서 볼륨의 상태를 볼 수 있습니다.

- 볼륨이 인스턴스에 연결되면 `in-use` 상태가 됩니다.
- 볼륨이 인스턴스에서 분리되면 `available` 상태가 됩니다. 이 볼륨을 삭제할 수 있습니다.

콘솔을 사용하여 EBS 볼륨을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 볼륨을 선택합니다.
3. 볼륨을 선택한 후 작업, 볼륨 삭제를 선택합니다. 볼륨 삭제가 회색으로 표시되면 볼륨이 인스턴스에 연결됩니다.
4. 확인 대화 상자에서 예, 삭제를 선택합니다.

명령줄을 사용하여 EBS 볼륨을 삭제하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `delete-volume`(AWS CLI)
- `Remove-EC2Volume`(Windows PowerShell용 AWS 도구)

Amazon EBS 스냅샷

지정 시간 스냅샷을 만들어 Amazon S3에 Amazon EBS 볼륨의 데이터를 백업할 수 있습니다. 스냅샷은 종분식 백업이어서 마지막 스냅샷 이후 변경된 디바이스의 블록만이 저장됩니다. 그러면 스냅샷을 만드는 데 필요한 시간이 최소화되며 데이터를 복제하지 않으므로 스토리지 비용이 절약됩니다. 각 스냅샷에는 (스냅샷을 만든 시점의) 데이터를 새 EBS 볼륨에 복원하는 데 필요한 모든 정보가 들어 있습니다.

스냅샷을 기반으로 EBS 볼륨을 생성하는 경우, 새 볼륨은 해당 스냅샷을 생성하는 데 사용된 원본 볼륨과 정확히 일치합니다. 복제된 볼륨은 사용자가 즉시 사용할 수 있도록 백그라운드에서 데이터를 로드합니다. 아직 로드되지 않은 데이터에 액세스하는 경우, 볼륨은 요청한 데이터를 Amazon S3에서 즉시 다운로드한 후 백그라운드에서 볼륨의 나머지 데이터를 계속해서 로드합니다. 자세한 내용은 [Amazon EBS 스냅샷 생성 \(p. 969\)](#) 단원을 참조하십시오.

스냅샷을 삭제하면 해당 스냅샷에 고유한 데이터만 제거됩니다. 자세한 내용은 [Amazon EBS 스냅샷 삭제 \(p. 982\)](#) 단원을 참조하십시오.

스냅샷 이벤트

CloudWatch 이벤트를 통해 EBS 스냅샷의 상태를 추적할 수 있습니다. 자세한 내용은 [EBS 스냅샷 이벤트 \(p. 1085\)](#) 단원을 참조하십시오.

애플리케이션 일치 스냅샷

시스템 관리자 명령 실행을 사용하여 Amazon EC2 Windows 인스턴스에 연결된 모든 EBS 볼륨에 대해 애플리케이션이 일치하는 스냅샷을 찍을 수 있습니다. 스냅샷 프로세스에서는 Windows VSS(Volume Shadow Copy Service)를 사용하여 이 애플리케이션과 디스크 사이에 대기 중인 트랜잭션에서 전송되는 데이터를 비롯해 VSS 인식 애플리케이션에 대해 이미지 수준 백업을 받습니다. 연결된 볼륨을 모두 백업하는 경우 인스턴스를 종료하거나 연결을 해제할 필요가 없습니다. 자세한 내용은 [VSS 애플리케이션이 일치하는 스냅샷 생성](#)을 참조하세요.

다중 볼륨 스냅샷

스냅샷을 사용하여 여러 EBS 볼륨에 걸쳐 있는 파일 시스템 또는 대규모 데이터베이스 등의 중요한 워크로드 백업을 생성할 수 있습니다. 다중 볼륨 스냅샷을 통해 EC2 인스턴스에 연결된 여러 EBS 볼륨에서 정확한 특정 시점, 데이터 조정 및 충돌 일치 스냅샷을 생성할 수 있습니다. 스냅샷은 여러 EBS 볼륨에서 자동으로 생성되기 때문에 더 이상 인스턴스를 중지하거나, 중단 일관성을 유지하기 위해 볼륨 간을 조정할 필요가 없습니다. 자세한 내용은 [Amazon EBS 스냅샷 생성 \(p. 969\)](#) 아래에서 다중 볼륨 EBS 스냅샷을 생성하기 위한 단계를 참조하세요.

스냅샷 요금

스냅샷에 대한 요금은 저장된 데이터 양에 따라 결정됩니다. 스냅샷은 종분이므로 스냅샷을 삭제하면 데이터 스토리지 비용이 줄어들지 않을 수 있습니다. 스냅샷에서 독점적으로 참조하는 데이터는 해당 스냅샷이 삭제될 때 제거되지만 다른 스냅샷에서 참조하는 데이터는 보존됩니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 [Amazon Elastic Block Store 볼륨 및 스냅샷](#)을 참조하십시오.

목차

- [종분 스냅샷의 작동 방식 \(p. 967\)](#)
- [스냅샷 복사 및 공유 \(p. 969\)](#)
- [스냅샷에 대한 암호화 지원 \(p. 969\)](#)
- [Amazon EBS 스냅샷 생성 \(p. 969\)](#)
- [VSS 애플리케이션이 일치하는 스냅샷 생성 \(p. 972\)](#)
- [Amazon EBS 스냅샷 삭제 \(p. 982\)](#)
- [Amazon EBS 스냅샷 복사 \(p. 985\)](#)
- [Amazon EBS 스냅샷 정보 보기 \(p. 989\)](#)
- [Amazon EBS 스냅샷 공유 \(p. 990\)](#)
- [EBS 스냅샷 콘텐츠 액세스 \(p. 993\)](#)
- [Amazon EBS 스냅샷 수명 주기 자동화 \(p. 1012\)](#)

종분 스냅샷의 작동 방식

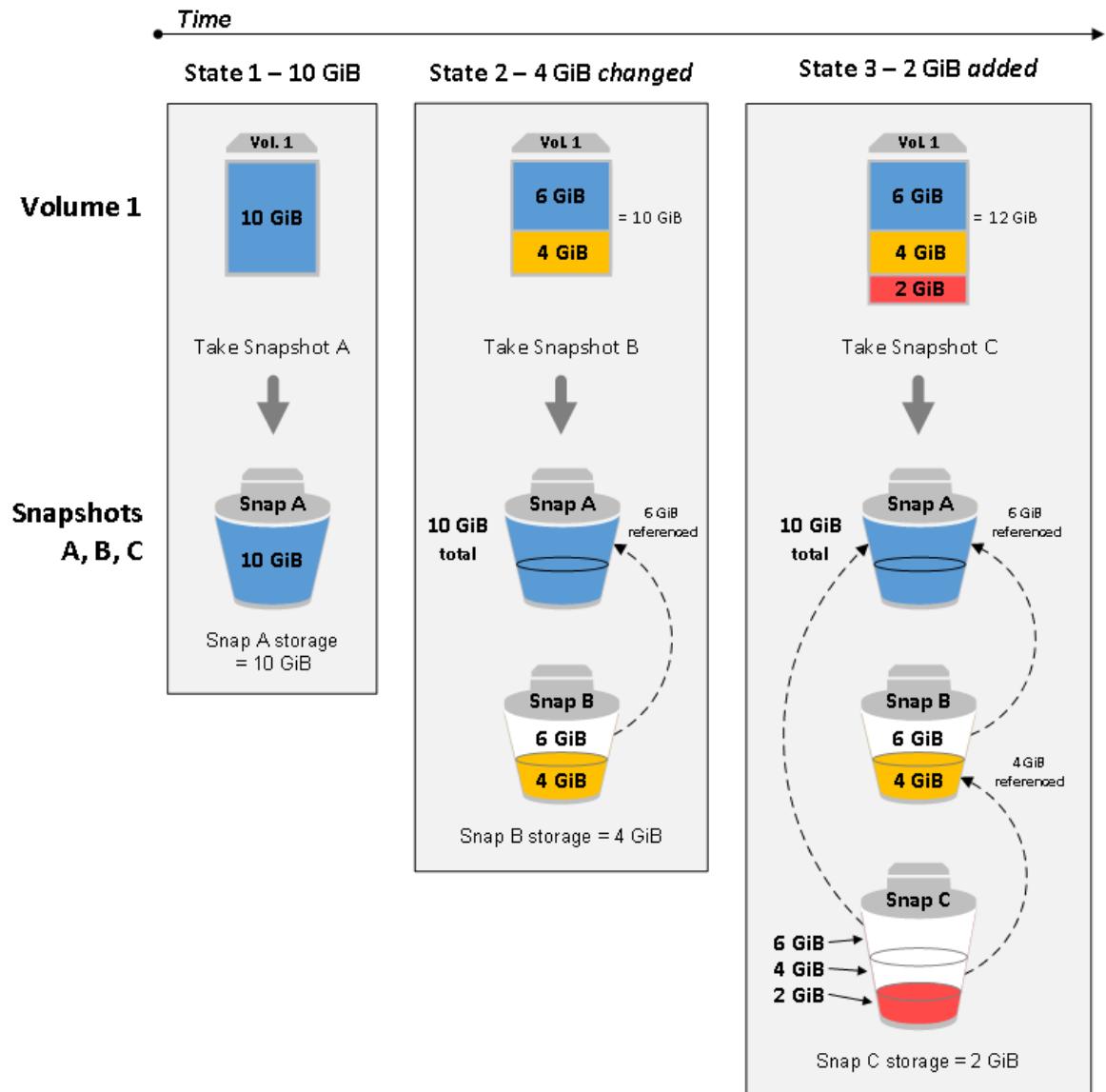
이 단원에서는 EBS 스냅샷이 특정 시점의 볼륨 상태를 캡처하는 방법과 변화하는 볼륨의 연속 스냅샷에 이러한 변경 기록이 표시되는 방법에 대해 설명합니다.

아래 다이어그램에서 볼륨 1은 세 가지 시점에 표시됩니다. 이 각 세 가지 볼륨 상태에 대한 스냅샷이 만들어집니다.

- 상태 1의 볼륨에는 10GiB의 데이터가 있습니다. 스냅 A는 이 볼륨의 첫 번째 스냅샷이므로 10GiB 데이터 전체를 복사해야 합니다.

- 상태 2의 볼륨에는 여전히 10GiB의 데이터가 있지만 4GiB가 변경되었습니다. 스냅 B는 스냅 A를 만든 후 변경된 4GiB만 복사하고 저장해야 합니다. 스냅 A에 이미 복사되어 저장된 변경되지 않은 나머지 6GiB 데이터는 (다시) 복사되는 것이 아니라 스냅 B에서 참조됩니다. 이는 파선 모양 화살표로 표시됩니다.
- 상태 3에서는 2GiB의 데이터가 볼륨에 추가되어 총 12GiB가 되었습니다. 스냅 C는 스냅 B를 만든 후 추가된 2GiB를 복사해야 합니다. 파선 모양 화살표로 표시되었듯이 스냅 C는 스냅 B에 저장된 4GiB의 데이터 및 스냅 A에 저장된 6GiB의 데이터를 참조합니다.
- 이 세 스냅샷에 필요한 총 스토리지는 16GiB입니다.

볼륨의 여러 스냅샷 간의 관계



Note

새 CMK로 스냅샷을 복사 및 암호화하면 항상 전체(증분 아님) 복사가 생성되고 추가 지연 및 스토리지 비용이 발생합니다.

스냅샷을 삭제할 때 데이터가 관리되는 방법에 대한 자세한 내용은 [Amazon EBS 스냅샷 삭제 \(p. 982\)](#) 단원을 참조하십시오.

스냅샷 복사 및 공유

액세스 권한을 수정하여 AWS 계정 간에 스냅샷을 공유할 수 있습니다. 자체 스냅샷뿐 아니라 공유된 스냅샷의 복사본을 만들 수 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

스냅샷은 해당 스냅샷이 생성된 AWS 리전으로 제한됩니다. EBS 볼륨의 스냅샷을 생성한 후, 해당 스냅샷을 사용하여 동일한 리전에 새로운 볼륨을 생성할 수 있습니다. 자세한 내용은 [스냅샷에서 볼륨 생성 \(p. 949\)](#) 단원을 참조하십시오. 또한 리전 전반에 스냅샷을 복사하면 지리적 확장, 데이터 센터 마이그레이션 및 재해 복구를 위해 여러 리전을 사용할 수 있습니다. 스냅샷의 복사 및 액세스는 스냅샷이 completed 상태인 경우에 가능합니다. 자세한 내용은 [Amazon EBS 스냅샷 복사 \(p. 985\)](#) 단원을 참조하십시오.

스냅샷에 대한 암호화 지원

EBS 스냅샷은 EBS 암호화를 전적으로 지원합니다.

- 암호화된 볼륨의 스냅샷은 자동으로 암호화됩니다.
- 암호화된 스냅샷에서 생성되는 볼륨은 자동으로 암호화됩니다.
- 사용자가 소유하거나 액세스 권한이 있는 암호화되지 않은 스냅샷에서 생성한 볼륨은 즉시 암호화할 수 있습니다.
- 자신이 소유한 암호화되지 않은 스냅샷을 복사할 때 복사 프로세스 중에 해당 스냅샷을 암호화할 수 있습니다.
- 자신이 소유하거나 액세스 권한이 있는 암호화된 스냅샷을 복사할 때는 복사 프로세스 중에 다른 키로 해당 스냅샷을 다시 암호화할 수 있습니다.
- 암호화되지 않은 스냅샷에서 생성된 암호화된 볼륨의 첫 번째 스냅샷은 항상 전체 스냅샷입니다.
- 소스 스냅샷과 비교하여 다른 CMK를 가진 재암호화된 볼륨의 첫 번째 스냅샷은 항상 전체 스냅샷입니다.

Note

새 CMK로 스냅샷을 복사 및 암호화하면 항상 전체(증분 아님) 복사가 생성되고 추가 지역 및 스토리지 비용이 발생합니다.

가능한 스냅샷 암호화 시나리오에 대한 전체 문서는 [Amazon EBS 스냅샷 생성 \(p. 969\)](#) 및 [Amazon EBS 스냅샷 복사 \(p. 985\)](#)에서 제공됩니다.

자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

Amazon EBS 스냅샷 생성

애플리케이션에 일관되는 스냅샷 생성은 [VSS 애플리케이션이 일치하는 스냅샷 생성 \(p. 972\)](#) 단원을 참조합니다.

EBS 볼륨의 특정 시점 스냅샷을 생성하여 새 볼륨이나 데이터 백업의 기준으로 사용할 수 있습니다. 볼륨의 스냅샷이 주기적으로 생성되는 경우 스냅샷은 증분식이어서 새 스냅샷은 마지막 스냅샷 이후 변경된 블록만 저장합니다.

스냅샷은 비동기적으로 생성됩니다. 특정 시점 스냅샷은 즉시 생성되지만 스냅샷이 완료될 때까지, 즉 수정된 블록이 pending로 모두 이동할 때까지 스냅샷 상태는 Amazon S3입니다. 따라서 크기가 큰 최초의 스냅샷이나 변경된 블록이 많은 후속 스냅샷의 경우 몇 시간씩 시간이 걸릴 수 있습니다. 완료하는 동안 진행 중인 스냅샷은 볼륨에 대한 지속적인 읽기 및 쓰기의 영향을 받지 않습니다.

연결되어 사용 중인 볼륨의 스냅샷을 만들 수 있습니다. 하지만 스냅샷은 snapshot 명령을 실행할 때 Amazon EBS 볼륨에 기록된 데이터만 캡처합니다. 이때 애플리케이션이나 운영 체제에 의해 캐시된 데이터가 제외될 수 있습니다. 스냅샷을 만들기에 충분한 시간 동안 볼륨에 대한 파일 쓰기 작업을 일시 종지할 수 있는 경우 스냅샷이 완전해야 합니다. 하지만 볼륨에 대한 모든 파일 쓰기를 일시 종지할 수는 없는 경우에는 인스턴스 내에서 볼륨을 분리하고 snapshot 명령을 실행한 다음, 볼륨을 다시 마운트하여 일관되고 완전한 스냅샷이 되도록 해야 합니다. 스냅샷 상태가 pending인 상태에서 볼륨을 다시 마운트하고 사용할 수 있습니다.

스냅샷을 보다 쉽게 관리하기 위해 생성 중에 스냅샷에 태그를 지정하거나 나중에 태그를 추가할 수도 있습니다. 예를 들어, 스냅샷이 생성된 원래 볼륨 또는 원래 볼륨을 인스턴스에 연결하는 데 사용된 디바이스 이름을 설명하는 태그를 적용할 수 있습니다. 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

스냅샷 암호화

암호화된 볼륨으로 생성한 스냅샷은 자동으로 암호화됩니다. 암호화된 스냅샷으로 생성한 볼륨도 자동으로 암호화됩니다. 암호화된 볼륨 및 모든 연관 스냅샷의 데이터는 사용되지 않을 때와 사용될 때 모두 보호됩니다. 자세한 내용은 [Amazon EBS 암호화 \(p. 1033\)](#) 단원을 참조하십시오.

기본적으로 사용자는 자기 소유의 스냅샷에서만 볼륨을 생성할 수 있습니다. 하지만 암호화되지 않은 스냅샷은 특정 AWS 계정과 공유하거나 공개 상태로 지정하여 전체 AWS 커뮤니티에서 공유할 수 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

암호화된 스냅샷은 특정 AWS 계정과만 공유할 수 있습니다. 다른 계정의 사용자가 암호화된 공유 스냅샷을 사용할 수 있도록 하려면 해당 스냅샷을 암호화할 때 사용했던 CMK 키도 공유해야 합니다. 암호화된 스냅샷에 대한 액세스 권한이 있는 사용자는 개인적으로 자체 복사본을 생성하고 이를 사용해야 합니다. 암호화된 공유 스냅샷의 복사본을 다른 키로 다시 암호화할 수도 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

Note

새 CMK로 스냅샷을 복사 및 암호화하면 항상 전체(증분 아님) 복사가 생성되고 추가 지역 및 스토리지 비용이 발생합니다.

다중 볼륨 스냅샷

EC2 인스턴스에 연결된 모든 EBS 볼륨의 특정 시점 스냅샷인 다중 볼륨 스냅샷을 생성할 수 있습니다. 또한 다중 볼륨 스냅샷의 생성 및 보존을 자동화하는 수명 주기 정책을 생성할 수도 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 수명 주기 자동화 \(p. 1012\)](#) 단원을 참조하십시오.

스냅샷이 생성된 후 각 스냅샷은 개별 스냅샷으로 처리됩니다. 단일 볼륨 스냅샷과 마찬가지로, 복원, 삭제, 리전 또는 계정 간 복사 등의 모든 스냅샷 작업을 수행할 수 있습니다. 또한 단일 볼륨 스냅샷과 마찬가지로 다중 볼륨 스냅샷에도 태그를 지정할 수 있습니다. 복원, 복사 또는 보존 중에 한꺼번에 관리할 수 있도록 다중 볼륨 스냅샷에 태그를 지정하는 것이 좋습니다.

다중 볼륨, 충돌 일치 스냅샷은 일반적으로 세트로 복원됩니다. 인스턴스 ID, 이름 또는 기타 관련 세부 정보로 세트에 태그를 지정하면 충돌 일치 세트에 있는 스냅샷을 식별하는 데 도움이 됩니다. 소스 볼륨에서 해당 스냅샷으로 태그를 자동 복사하도록 선택할 수 있습니다. 그러면 액세스 정책, 연결 정보, 비용 할당 등의 스냅샷 메타데이터를 소스 볼륨과 일치하도록 설정하는 데 도움이 됩니다.

다중 볼륨 스냅샷은 생성 후 다른 모든 스냅샷과 마찬가지로 동작합니다. 리전 및 계정에서 복원, 삭제, 복사 등의 모든 작업을 수행할 수 있습니다. 스냅샷에 태그를 지정할 수도 있습니다. 복원, 복사 또는 보존 중에 한꺼번에 관리할 수 있도록 다중 볼륨 스냅샷에 태그를 지정하는 것이 좋습니다.

스냅샷을 생성하면 정확한 특정 시점에 생성된 스냅샷이 EC2 콘솔에 표시됩니다. 스냅샷은 한꺼번에 관리되므로 볼륨 세트의 스냅샷 하나가 실패하면 다른 모든 스냅샷도 오류 상태를 표시합니다.

Amazon 데이터 수명 주기 관리자

스냅샷을 수동으로 생성, 보존 및 삭제하거나 Amazon 데이터 수명 주기 관리자를 사용하여 스냅샷을 관리할 수 있습니다. 자세한 내용은 [스냅샷 자동화 \(p. 1012\)](#) 단원을 참조하십시오.

고려 사항

다음은 스냅샷을 생성할 때 고려할 사항입니다.

- 루트 디바이스 역할을 하는 EBS 볼륨의 스냅샷을 생성할 때는 인스턴스를 중지한 후 스냅샷을 생성해야 합니다.
- 최대 절전 모드가 활성화된 인스턴스에서는 스냅샷을 생성할 수 없습니다.

- 최대 절전 모드 인스턴스에서는 스냅샷을 생성할 수 없습니다.
- 볼륨의 이전 스냅샷이 pending 상태일 때에도 볼륨의 스냅샷을 생성할 수는 있지만 볼륨의 pending 스냅샷을 여러 개 생성하면 스냅샷이 완료될 때까지 볼륨 성능이 저하될 수 있습니다.
- 단일 gp2, io1, io2 또는 Magnetic 볼륨에 대해 다섯 개의 pending 스냅샷, 또는 단일 st1 또는 sc1 볼륨에 대해 한 개의 pending 스냅샷으로 제한됩니다. 동일 볼륨에 대해 여러 개의 동일 스냅샷을 생성하려 할 때 ConcurrentSnapshotLimitExceeded 오류가 표시되면 한 개 이상의 pending 스냅샷이 완료될 때까지 기다린 후 해당 볼륨의 다른 스냅샷을 생성하십시오.
- AWS Marketplace 제품 코드를 가진 볼륨에서 스냅샷을 만들면 해당 제품 코드가 스냅샷으로 전파됩니다.

스냅샷 생성

지정된 볼륨에서 스냅샷을 생성하려면 다음 절차에 따르십시오.

콘솔을 이용하여 스냅샷을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Elastic Block Store에서 스냅샷을 선택합니다.
3. Create Snapshot을 클릭합니다.
4. 리소스 유형 선택에서 볼륨을 선택합니다.
5. 볼륨에서 볼륨을 선택합니다.
6. (선택 사항) 스냅샷에 대한 설명을 입력합니다.
7. (선택 사항) 태그 추가를 선택하여 스냅샷에 태그를 추가합니다. 각 태그에 대해 태그 키와 태그 값을 제공합니다.
8. Create Snapshot을 클릭합니다.

명령줄을 이용하여 스냅샷을 생성하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [create-snapshot](#) (AWS CLI)
- [New-EC2Snapshot](#)(Windows PowerShell용 AWS 도구)

다중 볼륨 스냅샷 생성

인스턴스의 볼륨에서 스냅샷을 생성하려면 다음 절차에 따르십시오.

콘솔을 사용하여 다중 볼륨 스냅샷을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Elastic Block Store에서 스냅샷을 선택합니다.
3. Create Snapshot을 클릭합니다.
4. 리소스 유형 선택에서 인스턴스를 선택합니다.
5. 연결된 모든 EBS 볼륨의 동시 백업을 생성할 인스턴스 ID를 선택합니다. 다중 볼륨 스냅샷은 인스턴스 당 최대 40개의 EBS 볼륨을 지원합니다.
6. (선택 사항) Exclude root volume(루트 볼륨 제외)을 설정합니다.
7. (선택 사항) Copy tags from volume(볼륨에서 태그 복사) 플래그를 설정하여 태그를 소스 볼륨에서 해당 스냅샷으로 자동으로 복사합니다. 그러면 액세스 정책, 연결 정보, 비용 할당 등의 스냅샷 메타데이터가 소스 볼륨과 일치하도록 설정됩니다.
8. (선택 사항) 태그 추가를 선택하여 스냅샷에 태그를 추가합니다. 각 태그에 대해 태그 키와 태그 값을 제공합니다.

9. Create Snapshot을 클릭합니다.

스냅샷 생성 중에 스냅샷이 함께 관리됩니다. 볼륨 세트의 스냅샷 중 하나가 실패하면 볼륨 세트의 다른 스냅샷이 오류 상태로 이동합니다. [CloudWatch Events](#)를 사용하여 스냅샷의 진행 상황을 모니터링할 수 있습니다. 스냅샷 생성 프로세스가 완료되면 CloudWatch는 해당 인스턴스의 상태 및 모든 관련 스냅샷 세부 정보가 포함된 이벤트를 생성합니다.

명령줄을 사용하여 다중 볼륨 스냅샷을 생성하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [create-snapshots\(AWS CLI\)](#)
- [New-EC2SnapshotBatch\(Windows PowerShell용 AWS 도구\)](#)

시스템 관리자 Run Command를 사용하여 애플리케이션 일치 스냅샷을 생성하려면

시스템 관리자 Run Command를 사용하여 Amazon EC2 Windows 인스턴스에 연결된 모든 EBS 볼륨의 애플리케이션 일치 스냅샷을 생성할 수 있습니다. 스냅샷 프로세스에서는 Windows [VSS\(Volume Shadow Copy Service\)](#)를 사용하여 이 애플리케이션과 디스크 사이에 대기 중인 트랜잭션에서 전송되는 데이터를 비롯해 VSS 인식 애플리케이션에 대해 이미지 수준 백업을 받습니다. 연결된 볼륨을 모두 백업하는 경우 인스턴스를 종료하거나 연결을 해제할 필요가 없습니다. 자세한 내용은 [VSS 애플리케이션이 일치하는 스냅샷 생성 \(p. 972\)](#) 단원을 참조하십시오.

EBS 스냅샷 작업

스냅샷을 복사하고, 스냅샷을 공유하고, 스냅샷에서 볼륨을 생성할 수 있습니다. 자세한 정보는 다음을 참조하십시오.

- [Amazon EBS 스냅샷 복사 \(p. 985\)](#)
- [Amazon EBS 스냅샷 공유 \(p. 990\)](#)
- [스냅샷에서 볼륨 생성 \(p. 949\)](#)

VSS 애플리케이션이 일치하는 스냅샷 생성

AWS 시스템 관리자Run Command를 사용하여 Amazon EC2 인스턴스의 Windows에 연결된 모든 [Amazon Elastic Block Store\(Amazon EBS\)](#) 볼륨에 대해 애플리케이션이 일치하는 스냅샷을 찍을 수 있습니다. 스냅샷 프로세스에서는 Windows [VSS\(Volume Shadow Copy Service\)](#)를 사용하여 VSS 인식 애플리케이션에 대해 이미지 수준 백업을 받습니다. 스냅샷에는 이러한 애플리케이션과 디스크 간에 대기 중인 트랜잭션의 데이터가 포함됩니다. 연결된 볼륨을 모두 백업해야 하는 경우 인스턴스를 종료하거나 연결을 해제할 필요가 없습니다.

VSS를 이용하는 EBS 스냅샷 사용에 따르는 추가 요금은 없습니다. 백업 프로세스에서 생성한 EBS 스냅샷에 대해서만 요금을 지불합니다. 자세한 정보는 [EBS 스냅샷 요금 계산 방식](#)을 참조하십시오.

목차

- [작동 방식 \(p. 973\)](#)
- [시작하기 전 \(p. 973\)](#)
- [시작하기 \(p. 973\)](#)
- [AWS CLI, Windows PowerShell용 AWS 도구 또는 AWSEC2-ManageVssIO SSM 문서를 사용하여 VSS 애플리케이션이 일치하는 스냅샷 생성 \(p. 978\)](#)
- [VSS를 이용하는 EBS 스냅샷에서 볼륨 복구 \(p. 981\)](#)
- [AWS VSS 구성 요소 패키지 버전 기록 \(p. 982\)](#)

작동 방식

다음 단계는 애플리케이션이 일치하는 VSS 이용 EBS 스냅샷을 찍는 프로세스의 작동 방식입니다.

1. 시스템 관리자 사전 조건 완료
2. AWSEC2-CreateVssSnapshot SSM 문서에 대해 파라미터를 입력하고 Run Command를 사용하여 이 문서를 실행합니다. 특정 볼륨의 경우에는 VSS를 이용하는 EBS 스냅샷을 생성할 수 없습니다. 그러나 백업 프로세스에서 부팅 볼륨을 제외할 파라미터를 지정할 수 있습니다.
3. 해당 인스턴스에 있는 VSS 에이전트는 애플리케이션 실행을 위해 모든 지속적인 I/O 작업을 조정합니다.
4. 이 시스템은 모든 I/O 버퍼를 플러시하고 모든 I/O 작업을 일시적으로 중지합니다. 중지 시간은 최대 10초입니다.
5. 중지된 시간 동안 시스템은 인스턴스에 연결된 모든 볼륨의 스냅샷을 생성합니다.
6. 중지가 해제되면 I/O가 작업을 재개합니다.
7. 시스템은 새로 생성된 모든 스냅샷을 EBS 스냅샷 목록에 추가합니다. 시스템은 이 프로세스에 따라 성공적으로 생성된 모든 VSS 이용 EBS 스냅샷에 AppConsistent:true라는 태그를 붙입니다. 이 태그는 이 프로세스에 따라 생성된 스냅샷을 다른 프로세스에 의해 생성된 것과 구별하는 데 도움이 됩니다. 시스템에 오류가 발생하면 이 프로세스에 따라 생성된 스냅샷에는 AppConsistent:true 태그가 포함되지 않습니다.
8. 스냅샷에서 복원해야 하는 경우 스냅샷에서 볼륨을 생성하는 표준 EBS 프로세스를 사용하거나 이 단원 후반부에 기술된 대로 샘플 스크립트를 사용하여 인스턴스에 모든 볼륨을 복원할 수 있습니다.

시작하기 전

Run Command를 사용하여 VSS를 이용하는 EBS 스냅샷을 생성하기 전에 다음 요구 사항 및 제한 사항을 검토하고 필요한 작업을 완료합니다.

Amazon EC2 Windows 인스턴스의 요구 사항

VSS를 이용하는 EBS 스냅샷은 Windows Server 2008 R2 이후 버전을 실행하는 인스턴스에 대해 지원됩니다. (Windows Server 2008 R2 Core는 현재 지원되지 않습니다.) 해당 인스턴스가 Amazon EC2 Windows에 대한 모든 요구 사항을 충족하는지 확인합니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 설정](#)을 참조하십시오.

SSM Agent 버전

인스턴스에서 SSM Agent 2.2.58.0 이후 버전을 사용할 수 있도록 업데이트합니다. 이전 버전의 SSM Agent을 사용하는 경우 Run Command를 사용하여 업데이트할 수 있습니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [Run Command를 사용한 SSM Agent 업데이트](#)를 참조하십시오.

Windows PowerShell용 AWS 도구 버전

인스턴스에서 Windows PowerShell용 AWS 도구 보존 3.3.48.0 이상을 실행 중인지 확인합니다. 버전 번호를 확인하려면 인스턴스에서 다음 명령을 실행합니다.

```
Get-AWSPowerShellVersion
```

인스턴스에서 Windows PowerShell용 도구의 버전을 업데이트해야 하는 경우 Windows PowerShell용 AWS 도구 사용 설명서의 [Windows 기반 컴퓨터에서 Windows PowerShell용 AWS 도구 설정](#)을 참조하십시오.

시작하기

이 지침에서는 VSS 구성 요소를 설치하고 EC2 Windows 인스턴스에 연결된 EBS 볼륨에 대한 애플리케이션이 일치하는 스냅샷을 수행하는 방법에 대한 지침을 제공합니다. 자세한 내용은 [Amazon EC2 Windows 인스턴스 시작하기](#)를 참조하십시오.

목차

- VSS를 이용하는 스냅샷에 대한 IAM 역할 생성 (p. 974)
- VSS 구성 요소를 EC2 인스턴스의 Windows로 다운로드하여 설치합니다. (p. 975)
- 콘솔을 사용하여 VSS 애플리케이션이 일치하는 스냅샷 생성 (p. 976)

VSS를 이용하는 스냅샷에 대한 IAM 역할 생성

다음 절차에서는 IAM 정책 및 IAM 역할을 사용하여 작업하는 방법에 대해 설명합니다. 그 정책을 통해 시스템 관리자는 스냅샷을 생성하고, 스냅샷을 태깅하며, 디바이스 ID와 같은 메타데이터를 시스템이 생성하는 기본 스냅샷 태그에 연결할 수 있습니다.

VSS를 이용하는 스냅샷에 대해 IAM 정책을 생성하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 정책을 선택한 후 정책 생성을 선택합니다.
3. 정책 생성 페이지에서 JSON 탭을 선택한 후, 기본 콘텐츠를 다음 JSON 정책으로 바꿉니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "ec2:CreateTags",  
            "Resource": [  
                "arn:aws:ec2:*::snapshot/*",  
                "arn:aws:ec2:*::image/*"  
            ]  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:DescribeInstances",  
                "ec2:CreateSnapshot",  
                "ec2:CreateImage",  
                "ec2:DescribeImages"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

CreateAmi 파라미터를 True로 설정하지 않으려면 첫 번째 정책 설명에서 arn:aws:ec2:*::image/*를 생략할 수 있으며 두 번째 정책 설명에서 ec2:CreateImage 및 ec2:DescribeImages를 생략할 수 있습니다.

CreateAmi 파라미터를 항상 True로 설정하려는 경우 두 번째 정책 설명에서 ec2:CreateSnapshot을 생략할 수 있습니다.

4. 정책 검토를 선택합니다.
5. 이름에 이 정책을 식별하기 위한 이름(예: **VssSnapshotRole**) 또는 선호하는 다른 이름을 입력합니다.
6. (선택 사항) 설명에 정책의 목적에 대한 설명을 입력합니다.
7. 정책 생성을 선택합니다.

다음 절차를 사용하여 VSS를 이용하는 스냅샷에 대한 IAM 역할을 생성합니다. 이 역할은 Amazon EC2 및 시스템 관리자에 대한 정책을 포함합니다.

VSS를 이용하는 EBS 스냅샷에 대해 IAM 역할을 생성하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.

2. 탐색 창에서 역할을 선택한 후 역할 생성을 선택합니다.
3. 신뢰할 수 있는 유형의 엔터티 선택 아래에서 AWS 서비스를 선택합니다.
4. 이 역할을 사용할 서비스 선택에서 EC2와 다음: 권한을 차례대로 선택합니다.
5. 사용 사례 선택에서 EC2를 선택한 후 다음: 권한을 선택합니다.
6. 정책 목록에서 AmazonSSMManagedInstanceCore 옆에 있는 상자를 선택합니다. (목록을 좁혀야 할 경우 검색 상자에 **SSM** 입력)
7. 다음: 태그를 선택합니다.
8. (선택 사항) 이 역할에 대한 액세스를 구성, 추적 또는 제어할 태그-키 값 페어를 하나 이상 추가한 후 다음: 검토를 선택합니다.
9. 역할 이름에 역할의 이름(예: **VssSnapshotRole** 또는 자신이 선호하는 다른 이름)을 입력합니다.
10. (선택 사항) 역할 설명에서 기본 텍스트를 이 역할의 목적에 대한 설명으로 바꿉니다.
11. 역할 생성을 선택합니다. 그러면 역할 페이지로 돌아갑니다.
12. 방금 생성한 역할을 선택합니다. 역할 요약 페이지가 열립니다.
13. 정책 연결을 선택합니다.
14. **VssSnapshotRole**과 같이 이전 절차에서 만든 정책 옆의 상자 또는 다른 이름을 검색하고 선택합니다.
15. 정책 연결을 선택합니다.
16. VSS를 이용하는 EBS 스냅샷을 생성하고자 하는 인스턴스에 이 역할을 연결합니다. 자세한 정보는 Amazon EC2 사용 설명서의 [인스턴스에 IAM 역할 연결](#)을 참조하십시오.

VSS 구성 요소를 EC2 인스턴스의 Windows로 다운로드하여 설치합니다.

시스템 관리자에는 해당 인스턴스에 설치될 VSS 구성 요소가 필요합니다. 다음 절차에 따라 AWSVssComponents 패키지를 사용하여 구성 요소를 설치합니다. 패키지는 두 가지 구성 요소, 즉 VSS 요청자 및 VSS 공급자를 설치합니다. 최신 AWS VSS 구성 요소 패키지를 설치하여 EC2 Windows 인스턴스에서 애플리케이션이 일치하는 스냅샷의 안정성과 성능을 개선하는 것이 좋습니다. 최신 패키지 버전을 보려면 [AWS VSS 구성 요소 패키지 버전 기록 \(p. 982\)](#) 단원을 참조하세요.

1. <https://console.aws.amazon.com/systems-manager/>에서 AWS 시스템 관리자 콘솔을 엽니다.
2. 탐색 창에서 Run Command를 선택합니다.
3. Run 명령을 선택합니다.
4. 명령 문서에서 AWS-ConfigureAWSPackage 옆의 버튼을 선택합니다.
5. 명령 파라미터에서 다음을 수행합니다.
 - a. 작업에서 설치로 설정되었는지 확인합니다.
 - b. 이름에 **AwsVssComponents**를 입력합니다.
 - c. 버전 필드는 비워두어 시스템 관리자가 최신 버전을 설치하도록 합니다.
6. 대상에서, 수동으로 태그를 지정하거나 인스턴스를 선택하여 이 작업을 실행할 인스턴스를 식별합니다.

Note

인스턴스를 수동으로 선택하려고 할 때 예상한 인스턴스가 목록에 없는 경우 AWS 시스템 관리자 사용 설명서의 [Where Are My Instances? \(내 인스턴스는 어디 있습니까?\)](#)에서 문제 해결 팁을 확인하십시오.

7. 다른 파라미터:
 - (선택 사항) 설명에 이 명령에 대한 정보를 입력합니다.
 - 제한 시간(초)에서 전체 명령 실행이 실패할 때까지 시스템이 기다리는 시간을 초 단위로 지정합니다.
8. (선택 사항) 속도 제어:
 - 동시성에서 명령을 동시에 실행할 인스턴스의 백분율 또는 개수를 지정합니다.

Note

Amazon EC2 태그를 선택하여 대상을 선택했지만 몇 개의 인스턴스가 선택된 태그를 사용할지 확실치 않다면 백분율을 지정하여 동시에 문서를 실행할 수 있는 인스턴스의 수를 제한하십시오.

- 오류 임계값에서, 명령이 인스턴스의 개수 또는 백분율에서 실패한 후 다른 인스턴스에서 해당 명령의 실행을 중지할 시간을 지정합니다. 예를 들어 세 오류를 지정하면 네 번째 오류를 받았을 때 시스템 관리자가 명령 전송을 중지합니다. 여전히 명령을 처리 중인 인스턴스도 오류를 전송할 수 있습니다.
9. (선택 사항) 출력 옵션 섹션에서 명령 출력을 파일에 저장하려면 Enable writing to an S3 bucket(S3 버킷에 쓰기 활성화) 옆의 상자를 선택합니다. 버킷과 접두사 (폴더) 이름(선택 사항)을 지정합니다.

Note

데이터를 S3 버킷에 쓰는 기능을 부여하는 S3 권한은 이 작업을 수행하는 IAM 사용자의 권한이 아닌 인스턴스에 할당된 인스턴스 프로파일의 권한입니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서에서 시스템 관리자에 대한 [IAM 인스턴스 프로파일 생성](#)을 참조하십시오.

10. (선택 사항) SNS 알림 옵션을 지정합니다.

Run Command에 대한 Amazon SNS 알림 구성에 대한 자세한 내용은 [AWS 시스템 관리자에 대한 Amazon SNS 알림 구성](#)을 참조하십시오.

11. 실행을 선택합니다.

콘솔을 사용하여 VSS 애플리케이션이 일치하는 스냅샷 생성

다음 절차에 따라 VSS를 이용하는 EBS 스냅샷을 생성합니다.

콘솔을 사용해 VSS를 이용하는 EBS 스냅샷을 생성하려면

1. <https://console.aws.amazon.com/systems-manager/>에서 AWS 시스템 관리자 콘솔을 엽니다.
2. 탐색 창에서 Run Command를 선택합니다.
3. Run command(명령 실행)를 선택합니다.
4. 명령 문서(Command document)에서 문서 이름(Document name)으로 AWSEC2-CreateVssSnapshot을 선택한 다음 문서 버전(Document version)으로 Latest version at runtime을 선택합니다.
5. 대상에서, 수동으로 태그를 지정하거나 인스턴스를 선택하여 이 작업을 실행할 인스턴스를 식별합니다.

Note

인스턴스를 수동으로 선택하려고 할 때 예상한 인스턴스가 목록에 없는 경우 [Where Are My Instances?\(내 인스턴스는 어디 있습니까?\)](#)에서 문제 해결 팁을 확인하십시오.

6. 명령 파라미터에서 다음을 수행합니다.
- a. Exclude Boot Volume(부트 볼륨 제외) 목록에서 옵션을 선택합니다. 이 파라미터를 사용하여 백업 프로세스에서 부팅 볼륨을 제외할 수 있습니다.
 - b. (선택 사항) 설명 필드에 설명을 입력합니다. 이 설명은 이 프로세스를 통해 생성되는 모든 스냅샷에 적용됩니다.
 - c. (선택 사항) 태그에 이 프로세스에서 생성된 모든 스냅샷에 적용하고 싶은 태그의 키와 값을 입력합니다. 태그를 통해 스냅샷 목록에서 볼륨을 검색, 관리, 복구할 수 있습니다. 시스템은 기본적으로 태그 파라미터에 Name 키를 입력합니다. 이 키의 값에 이 프로세스에서 생성된 스냅샷에 적용하고 싶은 이름을 지정합니다. 태그를 추가로 지정하고 싶다면 세미콜론을 사용해 태그를 구분합니다. 예, Key=*Environment*, Value=*Test*;Key=*User*, Value=*TestUser1*.

스냅샷에 태그를 지정하는 것이 좋습니다. 시스템은 기본적으로 디바이스 ID와 AppConsistent(성공적이고 애플리케이션과 일치하는 VSS 이용 EBS 스냅샷을 나타내는 데 사용)가 있는 스냅샷을 태깅합니다.

- d. Copy Only(복사만)에서 True를 선택해 복사만 백업 작업을 수행합니다. 이 옵션은 기본적으로 False으로 설정되어 있으므로 시스템에서는 전체 백업 작업을 수행합니다. 전체 백업 작업은 시스템에서 백업 수행 시 SQL Server에서 차등 백업 체인을 끊을 수 없도록 합니다.

Note

이 옵션을 사용하려면 AWS VSS 공급자 버전 1.2.00 이상이 설치되어 있어야 합니다.

- e. 스냅샷 프로세스에서 애플리케이션 VSS 쓰기를 제외하려면 No Writers(쓰기 없음)에서 True를 선택합니다. 이렇게 하면 타사 VSS 백업 구성 요소와의 갈등을 해결하는 데 도움이 될 수 있습니다. 이 옵션의 기본값은 False입니다.

Note

이 옵션을 사용하려면 AWS VSS 공급자 버전 1.2.00 이상이 설치되어 있어야 합니다.

- f. CreateAMI에서 EBS 스냅샷 대신 VSS가 활성화된 Amazon 머신 이미지(AMI) 백업을 생성하려면 True를 선택합니다. 이 옵션의 기본값은 False입니다. AMI 생성에 대한 자세한 내용은 [실행 중인 인스턴스에서 Windows AMI 생성](#)을 참조하세요.
- g. (선택 사항) Aminame에서 생성된 AMI의 이름을 지정합니다. 이 옵션은 CreateAmi 옵션이 True로 설정된 경우에만 적용됩니다.

7. 다른 파라미터:

- 설명에는 명령에 대한 정보를 입력합니다.
- 제한 시간(초)에서 전체 명령 실행이 실패할 때까지 시스템이 기다리는 시간을 초 단위로 지정합니다.

8. (선택 사항) 속도 제어:

- 동시성에서 명령을 동시에 실행할 인스턴스의 백분율 또는 개수를 지정합니다.

Note

Amazon EC2 태그를 선택하여 대상을 선택했지만 몇 개의 인스턴스가 선택된 태그를 사용할지 확실치 않다면 백분율을 지정하여 동시에 문서를 실행할 수 있는 인스턴스의 수를 제한하십시오.

- 오류 임계값에서, 명령이 인스턴스의 개수 또는 백분율에서 실패한 후 다른 인스턴스에서 해당 명령의 실행을 중지할 시간을 지정합니다. 예를 들어 세 오류를 지정하면 네 번째 오류를 받았을 때 시스템 관리자가 명령 전송을 중지합니다. 여전히 명령을 처리 중인 인스턴스도 오류를 전송할 수 있습니다.
9. (선택 사항) 출력 옵션에서 명령 출력을 파일에 저장하려면 Enable writing to an S3 bucket(S3 버킷에 쓰기 활성화) 옆의 상자를 선택합니다. 버킷과 접두사 (폴더) 이름(선택 사항)을 지정합니다.

Note

데이터를 S3 버킷에 쓰는 기능을 부여하는 S3 권한은 이 작업을 수행하는 IAM 사용자의 권한이 아닌 인스턴스에 할당된 인스턴스 프로파일의 권한입니다. 자세한 내용은 [시스템 관리자 설정](#)을 참조하십시오.

10. (선택 사항) SNS 알림 옵션을 지정합니다.

Run Command에 대한 Amazon SNS 알림 구성에 대한 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자에 대한 Amazon SNS 알림 구성](#)을 참조하십시오.

11. 실행을 선택합니다.

성공적으로 실행되면 명령은 EBS 스냅샷 목록에 새로운 스냅샷을 입력합니다. 지정한 태그를 검색하거나 AppConsistent를 검색하여 EBS 스냅샷 목록에서 이 스냅샷을 찾을 수 있습니다. 명령 실행에 실패하면 시스템 관리자 명령 출력을 보고 명령이 실패한 이유에 대한 세부 정보를 확인합니다. 명령이 성공적으로 완료되었지만 특정 볼륨의 백업이 실패한 경우 EBS 볼륨 목록에서 문제를 해결할 수 있습니다.

명령이 실패한 경우 시스템 관리자에서 VPC 엔드포인트를 사용 중이면 com.amazonaws.**region**.ec2-엔드포인트를 구성했는지 확인하십시오. **EC2** 엔드포인트가 정의되어 있지 않으면 연결된 EBS 볼륨을

표시하는 호출이 실패하고 이에 따라 시스템 관리자 명령에 실패합니다. 시스템 관리자(를) 이용한 VPC 엔드포인트 설정에 대한 자세한 설명은 AWS 시스템 관리자 사용 설명서의 [Virtual Private Cloud 엔드포인트 생성](#) 단원을 참조하십시오.

Note

[AWSEC2-CreateVssSnapshot](#) SSM 문서를 사용하는 Maintenance Window 작업을 생성하여 백업을 자동화할 수 있습니다. 자세한 내용은 [Maintenance Windows\(콘솔\)을 사용한 작업](#)을 참조하십시오.

AWS CLI, Windows PowerShell용 AWS 도구 또는 AWSEC2-ManageVssIO SSM 문서를 사용하여 VSS 애플리케이션이 일치하는 스냅샷 생성

이 단원에는 AWS CLI 또는 Windows PowerShell용 AWS 도구(를) 사용해 VSS를 이용하는 EBS 스냅샷을 생성하는 절차가 포함되어 있습니다. 또한 AWSEC2-ManageVssIO SSM 문서를 사용해 VSS를 이용하는 스냅샷을 생성하는 고급 방법도 설명합니다.

목차

- [AWS CLI 또는 Windows PowerShell용 도구\(를\) 사용한 VSS 패키지 설치](#) (p. 978)
- [AWS CLI, Windows PowerShell용 도구 또는 AWSEC2-ManageVssIO SSM 문서를 통해 VSS를 이용하는 EBS 스냅샷을 생성하려면](#) (p. 979)

AWS CLI 또는 Windows PowerShell용 도구(를) 사용한 VSS 패키지 설치

다음 명령줄 절차 중 하나를 사용하여 VSS 구성 요소를 EC2 인스턴스의 Windows에 다운로드하여 설치합니다.

AWS CLI를 사용한 VSS 패키지 설치

다음 절차를 통해 AWS CLI에서 Run Command를 사용해 AwsVssComponents 패키지를 다운로드하여 인스턴스에 설치합니다. 패키지는 두 가지 구성 요소, 즉 VSS 요청자 및 VSS 공급자를 설치합니다. 시스템은 이를 구성 요소들 인스턴스의 디렉터리에 복사한 후 공급자 DLL을 VSS 공급자로 등록합니다.

AWS CLI를 사용해 VSS 패키지를 설치하려면

1. Install and configure the AWS CLI, if you have not already.

For information, see [Install or Upgrade and then Configure the AWS CLI](#) in the AWS 시스템 관리자 사용 설명서.

2. 다음 명령을 실행하여 시스템 관리자용 필수 VSS 구성 요소를 다운로드하여 설치합니다.

```
aws ssm send-command --document-name "AWS-ConfigureAWSPackage" --instance-ids "i-12345678" --parameters '{"action": ["Install"], "name": ["AwsVssComponents"]}'
```

Windows PowerShell용 도구를 사용한 VSS 패키지 설치

다음 절차를 통해 Windows PowerShell용 도구에서 Run Command를 사용해 AwsVssComponents 패키지를 다운로드하여 인스턴스에 설치합니다. 패키지는 두 가지 구성 요소, 즉 VSS 요청자 및 VSS 공급자를 설치합니다. 시스템은 이를 구성 요소들 인스턴스의 디렉터리에 복사한 후 공급자 DLL을 VSS 공급자로 등록합니다.

Windows용 AWS 도구 PowerShell을 사용해 VSS 패키지를 설치하려면

1. Windows PowerShell용 AWS 도구를 열고 다음 명령을 실행하여 자격 증명을 지정합니다. Amazon EC2에 관리자 권한이 있거나 IAM에서 적절한 권한을 부여 받아야 합니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 설정](#)을 참조하십시오.

```
Set-AWSCredentials -AccessKey key_name -SecretKey key_name
```

2. 다음 명령을 실행하여 PowerShell 세션의 리전을 설정합니다. 이 예에서는 us-east-2 리전을 사용합니다.

```
Set-DefaultAWSRegion -Region us-east-2
```

3. 다음 명령을 실행하여 시스템 관리자용 필수 VSS 구성 요소를 다운로드하여 설치합니다.

```
Send-SSMCommand -DocumentName AWS-ConfigureAWSPackage -InstanceId "$instance" -Parameter @{'action'='Install';'name'='AwsVssComponents'}
```

AWS CLI, Windows PowerShell용 도구 또는 AWSEC2-ManageVssIO SSM 문서를 통해 VSS를 이용하는 EBS 스냅샷을 생성하려면

다음 명령줄 절차를 실행하여 VSS를 이용하는 EBS 스냅샷을 생성합니다.

AWS CLI를 사용해 VSS를 이용하는 EBS 스냅샷 생성

다음 절차에 따라 AWS CLI를 통해 VSS를 이용하는 EBS 스냅샷을 생성합니다. 명령을 실행할 때 다음 파라미터를 지정할 수 있습니다.

- 인스턴스(필수): Amazon EC2 Windows 한 개 이상 지정합니다. 인스턴스를 수동으로 지정하거나 태그를 지정할 수 있습니다.
- 설명(선택 사항): 이 백업에 대한 세부 정보를 지정합니다.
- 태그(선택 사항): 스냅샷에 할당하고 싶은 키-값 태그 쌍을 지정합니다. 태그를 통해 스냅샷 목록에서 볼륨을 검색, 관리, 복구할 수 있습니다. 시스템은 기본적으로 태그 파라미터에 Name 키를 입력합니다. 이 키의 값에 이 프로세스에서 생성된 스냅샷에 적용하고 싶은 이름을 지정합니다. 또한 Key=*Environment*, Value=*Test*, Key=*User*, Value=*TestUser1* 형식을 사용하여 이 목록에 사용자 지정 태그를 추가할 수 있습니다.

이 파라미터는 선택 사항이지만 스냅샷을 태깅하는 것이 좋습니다. 시스템은 기본적으로 디바이스 ID와 AppConsistent(성공적이고 애플리케이션과 일치하는 VSS 이용 EBS 스냅샷을 나타내는 데 사용)가 있는 스냅샷을 태깅합니다.

- 부팅 볼륨 제외(선택 사항): 이 파라미터를 사용하여 백업 프로세스에서 부팅 볼륨을 제외할 수 있습니다.

AWS CLI를 통해 VSS를 이용하는 EBS 스냅샷을 생성하려면

1. Install and configure the AWS CLI, if you have not already.

For information, see [Install or Upgrade and then Configure the AWS CLI](#) in the AWS 시스템 관리자 사용 설명서.

2. 다음 명령을 실행하여 VSS를 이용하는 EBS 스냅샷을 생성합니다.

```
aws ssm send-command --document-name "AWSEC2-CreateVssSnapshot" --instance-ids "i-12345678" --parameters '[{"ExcludeBootVolume":["False"], "description": ["Description"], "tags": [{"Key":key_name, "Value":tag_value}]}'
```

성공적으로 실행되면 명령은 EBS 스냅샷 목록에 새로운 스냅샷을 입력합니다. 지정한 태그를 검색하거나 AppConsistent를 검색하여 EBS 스냅샷 목록에서 이 스냅샷을 찾을 수 있습니다. 명령 실행이 실패하면 명령 출력을 보고 명령이 실패한 이유에 대한 세부 정보를 확인합니다.

AWSEC2-CreateVssSnapshot SSM 문서를 사용하는 Maintenance Window 작업을 생성하여 백업을 자동화할 수 있습니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [Maintenance Windows\(콘솔\)을 사용한 작업](#)을 참조하십시오.

Windows용 AWS 도구 PowerShell을 사용해 VSS를 이용하는 EBS 스냅샷 생성

다음 절차에 따라 Windows용 AWS 도구 PowerShell을 통해 VSS를 이용하는 EBS 스냅샷을 생성합니다. 명령을 실행할 때 다음 파라미터를 지정할 수 있습니다.

- 인스턴스(필수): Amazon EC2 Windows 한 개 이상 지정합니다. 인스턴스를 수동으로 지정하거나 태그를 지정할 수 있습니다.
- 설명(선택 사항): 이 백업에 대한 세부 정보를 지정합니다.
- 태그(선택 사항): 스냅샷에 할당하고 싶은 키-값 태그 쌍을 지정합니다. 태그를 통해 스냅샷 목록에서 볼륨을 검색, 관리, 복구할 수 있습니다. 시스템은 기본적으로 태그 파라미터에 Name 키를 입력합니다. 이 키의 값에 이 프로세스에서 생성된 스냅샷에 적용하고 싶은 이름을 지정합니다. 또한 Key=*Environment*, Value=*Test*, Key=*User*, Value=*TestUser1* 형식을 사용하여 이 목록에 사용자 지정 태그를 추가할 수 있습니다.

이 파라미터는 선택 사항이지만 스냅샷을 태깅하는 것이 좋습니다. 시스템은 기본적으로 디바이스 ID와 AppConsistent(성공적이고 애플리케이션과 일치하는 VSS 이용 EBS 스냅샷을 나타내는데 사용)가 있는 스냅샷을 태깅합니다.

- 부팅 볼륨 제외(선택 사항): 이 파라미터를 사용하여 백업 프로세스에서 부팅 볼륨을 제외할 수 있습니다.

Windows용 AWS 도구 PowerShell을 통해 VSS를 이용하는 EBS 스냅샷을 생성하려면

1. Windows PowerShell용 AWS 도구를 열고 다음 명령을 실행하여 자격 증명을 지정합니다. Amazon EC2에 관리자 권한이 있거나 IAM에서 적절한 권한을 부여 받아야 합니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 설정](#)을 참조하십시오.

```
Set-AWSCredentials -AccessKey key_name -SecretKey key_name
```

2. 다음 명령을 실행하여 PowerShell 세션의 리전을 설정합니다. 이 예에서는 us-east-2 리전을 사용합니다.

```
Set-DefaultAWSRegion -Region us-east-2
```

3. 다음 명령을 실행하여 VSS를 이용하는 EBS 스냅샷을 생성합니다.

```
Send-SSMCommand -DocumentName AWSEC2-CreateVssSnapshot -InstanceId "$instance" -  
Parameter @{'ExcludeBootVolume'='False';'description'='a_description'  
;'tags'='Key=key_name,Value=tag_value'}
```

성공적으로 실행되면 명령은 EBS 스냅샷 목록에 새로운 스냅샷을 입력합니다. 지정한 태그를 검색하거나 AppConsistent를 검색하여 EBS 스냅샷 목록에서 이 스냅샷을 찾을 수 있습니다. 명령 실행이 실패하면 명령 출력을 보고 명령이 실패한 이유에 대한 세부 정보를 확인합니다. 명령이 성공적으로 완료되었지만 특정 볼륨의 백업이 실패한 경우 EBS 스냅샷 목록에서 문제를 해결할 수 있습니다.

AWSEC2-CreateVssSnapshot SSM 문서를 사용하는 Maintenance Window 작업을 생성하여 백업을 자동화할 수 있습니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [Maintenance Windows\(콘솔\)을 사용한 작업](#)을 참조하십시오.

AWSEC2-ManageVssIO SSM 문서(고급)를 통해 VSS를 이용하는 EBS 스냅샷 생성

다음 스크립트와 사전 정의된 AWSEC2-ManageVssIO SSM 문서를 사용하여 일시적으로 I/O를 중지하고, VSS를 이용하는 EBS 스냅샷을 생성하고, I/O를 다시 시작할 수 있습니다. 이 프로세스는 명령을 실행하는

사용자의 맥락에서 실행됩니다. 사용자에게 스냅샷을 생성하고 태깅할 수 있는 권한이 충분히 있는 경우 AWS 시스템 관리자는(는) 해당 인스턴스에서 추가 IAM 스냅샷 역할 없이도 VSS를 이용하는 EBS 스냅샷을 생성하고 태깅할 수 있습니다.

이와 대조적으로 AWSEC2-CreateVssSnapshot 문서에서는 EBS 스냅샷을 생성할 대상인 각 인스턴스에 IAM 스냅샷 역할을 할당해야 합니다. 정책 또는 규정 준수를 이유로 해당 인스턴스에 추가 IAM 권한을 부여하고 싶지 않다면 다음 스크립트를 사용할 수 있습니다.

시작하기 전

이 프로세스에 대한 다음과 같은 중요 세부 정보에 주의하십시오.

- 이 프로세스에서는 PowerShell 스크립트(CreateVssSnapshotAdvancedScript.ps1)를 사용하여 사용자가 지정하는 인스턴스의 모든 볼륨(루트 볼륨 제외)에 대해 스냅샷을 만듭니다. 루트 볼륨의 스냅샷을 만들어야 하는 경우 AWSEC2-CreateVssSnapshot SSM 문서를 사용해야 합니다.
- 이 스크립트는 AWSEC2-ManageVssIO 문서를 두 번 호출합니다. 첫 번째는 Action 파라미터가 Freeze로 설정되며, 인스턴스에서 모든 I/O를 중지합니다. 두 번째는 Action 파라미터가 Thaw로 설정되고, I/O를 다시 시작합니다.
- CreateVssSnapshotAdvancedScript.ps1 스크립트를 사용하지 않는 방식으로 AWSEC2-ManageVssIO 문서를 사용하려고 해서는 안 됩니다. VSS 관련 제한으로 인해 Freeze 및 Thaw 작업은 각각 10초 이상 호출할 수 없으며, 스크립트 없이 이 작업들을 수동으로 호출하면 오류가 발생할 수 있습니다.

AWSEC2-ManageVssIO SSM 문서를 통해 VSS를 이용하는 EBS 스냅샷을 생성하려면

- Windows PowerShell용 AWS 도구를 열고 다음 명령을 실행하여 자격 증명을 지정합니다. Amazon EC2에 관리자 권한이 있거나 IAM에서 적절한 권한을 부여 받아야 합니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 설정](#)을 참조하십시오.

```
Set-AWSCredentials -AccessKey key_name -SecretKey key_name
```

- 다음 명령을 실행하여 PowerShell 세션의 리전을 설정합니다. 이 예에서는 us-east-2 리전을 사용합니다.

```
Set-DefaultAWSRegion -Region us-east-2
```

- [CreateVssSnapshotAdvancedScript.zip](#) 파일을 다운로드한 후에 파일 콘텐츠의 압축을 풉니다.
- 텍스트 편집기에서 CreateVssSnapshotAdvancedScript.ps1을 열고 유효한 EC2 인스턴스 ID, 스냅샷 설명 및 원하는 태그 값을 사용하여 스크립트 하단에서 샘플 호출을 편집한 다음 PowerShell에서 스크립트를 실행합니다.

성공적으로 실행되면 명령은 EBS 스냅샷 목록에 새로운 스냅샷을 입력합니다. 지정한 태그를 검색하거나 AppConsistent를 검색하여 EBS 스냅샷 목록에서 이 스냅샷을 찾을 수 있습니다. 명령 실행이 실패하면 명령 출력을 보고 명령이 실패한 이유에 대한 세부 정보를 확인합니다. 명령이 성공적으로 완료되었지만 특정 볼륨의 백업이 실패한 경우 EBS 볼륨 목록에서 문제를 해결할 수 있습니다.

VSS를 이용하는 EBS 스냅샷에서 볼륨 복구

RestoreVssSnapshotSampleScript.ps1 스크립트를 통해 VSS를 이용하는 EBS 스냅샷에서 인스턴스의 볼륨을 복구할 수 있습니다. 이 스크립트는 다음 작업을 수행합니다.

- 인스턴스 종지
- 인스턴스에서 기존 드라이브를 모두 제거(부팅 볼륨이 제외된 경우 제외된 부팅 볼륨은 제외)
- 스냅샷에서 새 볼륨 생성
- 스냅샷의 디바이스 ID 태그를 사용하여 볼륨을 인스턴스에 연결
- 인스턴스를 다시 시작

Important

다음 스크립트는 인스턴스에 연결된 모든 볼륨을 분리한 후 스냅샷에서 새 볼륨을 만듭니다. 인스턴스를 적당히 백업했는지 확인하십시오. 이전 볼륨은 삭제되지 않습니다. 삭제하고 싶다면 이전 볼륨을 삭제하도록 스크립트를 편집할 수 있습니다.

VSS를 이용하는 EBS 스냅샷에서 볼륨을 복구하려면

1. Windows PowerShell용 AWS 도구를 열고 다음 명령을 실행하여 자격 증명을 지정합니다. Amazon EC2에 관리자 권한이 있거나 IAM에서 적절한 권한을 부여 받아야 합니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 설정](#)을 참조하십시오.

```
Set-AWSCredentials -AccessKey key_name -SecretKey key_name
```

2. 다음 명령을 실행하여 PowerShell 세션의 리전을 설정합니다. 이 예에서는 us-east-2 리전을 사용합니다.

```
Set-DefaultAWSRegion -Region us-east-2
```

3. [RestoreVssSnapshotSampleScript.zip](#) 파일을 다운로드한 후에 파일 콘텐츠의 압축을 풉니다.
4. 텍스트 에디터에서 [RestoreVssSnapshotSampleScript.zip](#)을 열고 유효한 EC2 인스턴스 ID, EBS 스냅샷 ID를 사용하여 스크립트 하단에서 샘플 호출을 편집한 다음 PowerShell에서 스크립트를 실행합니다.

AWS VSS 구성 요소 패키지 버전 기록

다음 표에서는 AWS VSS 구성 요소 패키지의 릴리스 버전에 대해 설명합니다.

버전	세부 정보	릴리스 날짜
1.3.1.0	<ul style="list-style-type: none">NTDS VSS 라이터 로깅 오류와 관련하여 도메인 컨트롤러에서 실패하는 스냅샷을 수정했습니다.버전 1.0 VSS 공급자를 제거할 때의 VSS 에이전트 오류를 해결했습니다.	2020년 2월 6일
1.3.00	<ul style="list-style-type: none">원하지 않는 세부 사항을 줄여 로깅을 개선했습니다.설치 중 지역화 문제를 해결했습니다.일부 공급자 등록 오류 조건에 대한 반환 코드를 수정했습니다.다양한 설치 문제를 해결했습니다.	2019년 3월 19일
1.2.00	<ul style="list-style-type: none">에이전트에 명령줄 파라미터 -nw(라이터 없음) 및 -copy(복사 전용)를 추가했습니다.부적절한 메모리 할당 호출로 인해 발생하는 EventLog 오류를 해결했습니다.	2018년 11월 15일
1.1	AwsVssProvider.dll이 기본 Windows 백업 및 복원 공급자로 잘못 사용되는 문제를 해결했습니다.	2017년 12월 12일
1.0	최초 릴리스.	2017년 11월 20일

Amazon EBS 스냅샷 삭제

볼륨의 Amazon EBS 스냅샷이 더 이상 필요하지 않으면 삭제할 수 있습니다. 스냅샷을 삭제해도 볼륨에는 영향을 주지 않습니다. 볼륨을 삭제해도 해당 볼륨에서 만들어진 스냅샷에는 아무런 영향을 미치지 않습니다.

증분 스냅샷 삭제

정기적으로 볼륨의 스냅샷을 만드는 경우, 스냅샷은 증분식으로 늘어납니다. 다시 말해 새 스냅샷에는 마지막 스냅샷 이후로 변경된 디바이스 블록만 저장됩니다. 스냅샷은 증분식으로 저장되지만 스냅샷 삭제 프로세스는 볼륨을 생성하기 위해 가장 최근의 스냅샷만을 유지할 수 있도록 설계됩니다. 볼륨에 존재했고, 이전 스냅샷 또는 스냅샷 시리즈에 저장되며 나중에 해당 볼륨에서 삭제되는 데이터는 여전히 이전 스냅샷의 고유한 데이터로 간주됩니다. 이 고유 데이터는 고유한 데이터를 참조하는 모든 스냅샷을 삭제하지 않는 한 스냅샷 시퀀스에서 삭제되지 않습니다.

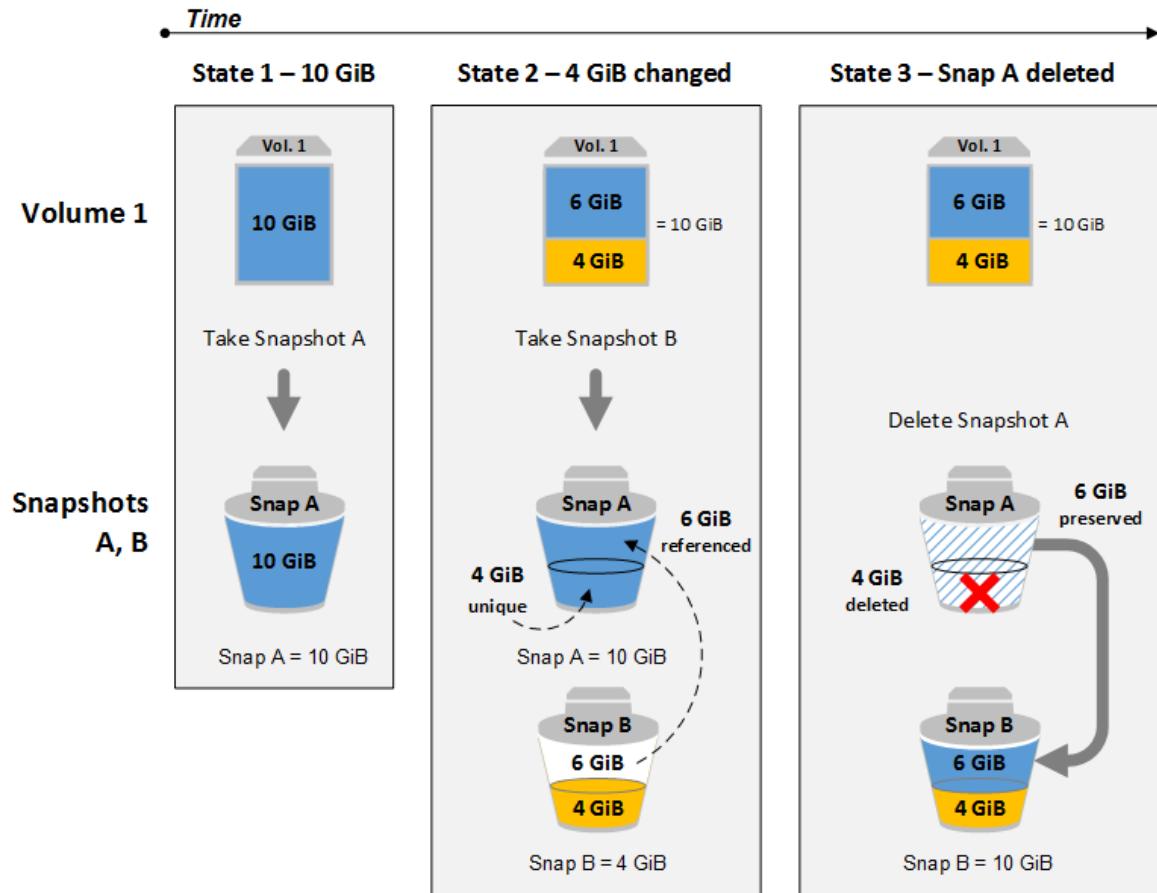
스냅샷을 삭제하면 해당 스냅샷에서만 참조하는 데이터만 제거됩니다. 고유한 데이터는 해당 데이터를 참조하는 모든 스냅샷을 삭제하지 않는 한 삭제되지 않습니다. 볼륨의 이전 스냅샷을 삭제해도 해당 볼륨의 이후 스냅샷에서 볼륨을 생성하는 기능에는 영향을 주지 않습니다.

스냅샷을 삭제해도 조직의 데이터 스토리지 비용이 줄어들지 않을 수 있습니다. 다른 스냅샷은 해당 스냅샷의 데이터를 참조할 수 있으며, 참조된 데이터는 항상 보존됩니다. 이후의 스냅샷에서 사용 중인 데이터가 포함된 스냅샷을 삭제하는 경우, 참조된 데이터와 관련된 비용이 이후의 스냅샷에 할당됩니다. 스냅샷이 데이터를 저장하는 방법에 대한 자세한 내용은 [증분 스냅샷의 작동 방식 \(p. 967\)](#) 및 다음 예를 참조하십시오.

다음 다이어그램에서 볼륨 1은 세 가지 시점에 표시됩니다. 스냅샷이 첫 두 상태를 각각 캡처했으며, 세 번째에서는 스냅샷이 삭제되었습니다.

- 상태 1의 볼륨에는 10GiB의 데이터가 있습니다. 스냅 A는 이 볼륨의 첫 번째 스냅샷으로 10GiB 데이터 전체를 복사해야 합니다.
- 상태 2의 볼륨에는 여전히 10GiB의 데이터가 있지만 4GiB가 변경되었습니다. 스냅 B는 스냅 A를 만든 후 변경된 4GiB만 복사하고 저장해야 합니다. 스냅 A에 이미 복사되어 저장된 변경되지 않은 나머지 6GiB 데이터는 (다시) 복사되는 것이 아니라 스냅 B에서 참조됩니다. 이는 파선 모양 화살표로 표시됩니다.
- 상태 3에서 볼륨은 상태 2 이후로 변경되지 않았지만 스냅샷 A가 삭제되었습니다. 스냅샷 A에 저장된 6GiB의 데이터를 스냅샷 B에서 참조했었지만 굵은 화살표로 표시되었듯이 이제 스냅샷 B로 이동했습니다. 결과적으로 스냅 A에서 보존된 변경되지 않은 6GiB의 데이터와 스냅 B에서 변경된 4GiB의 데이터를 합쳐, 여전히 10GiB의 데이터를 저장한 것이 됩니다.

다른 스냅샷에서 참조된 데이터가 포함된 스냅샷 삭제



고려 사항

다음은 스냅샷을 삭제할 때 고려할 사항입니다.

- 등록된 AMI에서 사용된 EBS 볼륨의 루트 디바이스에 대한 스냅샷을 삭제할 수 없습니다. 스냅샷을 삭제하기 전 우선 AMI를 등록해야 합니다. 자세한 내용은 [Windows AMI 등록 최소 \(p. 111\)](#) 단원을 참조하십시오.
- Amazon EC2를 사용하여 AWS Backup 서비스에서 관리하는 스냅샷은 삭제할 수 없습니다. 대신 AWS Backup을 사용하여 백업 볼트의 해당 복구 지점을 삭제합니다.
- 스냅샷을 수동으로 생성, 보존 및 삭제하거나 Amazon 데이터 수명 주기 관리자를 사용하여 스냅샷을 관리할 수 있습니다. 자세한 내용은 [스냅샷 자동화 \(p. 1012\)](#) 단원을 참조하십시오.
- 진행 중인 스냅샷을 삭제할 수는 있지만, 삭제가 적용되려면 해당 스냅샷이 완전해야 합니다. 이 작업은 시간이 오래 걸릴 수 있습니다. 동시 스냅샷 제한 상태에서 스냅샷을 추가로 만들려고 하면 `ConcurrentSnapshotLimitExceeded` 오류를 받을 수 있습니다. 자세한 내용은 Amazon Web Services 일반 참조의 Amazon EBS에 대한 [서비스 할당량](#)을 참조하세요.

스냅샷 삭제

스냅샷을 삭제하려면 다음 절차에 따르십시오.

콘솔을 이용하여 스냅샷을 삭제하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 스냅샷을 선택합니다.

3. 스냅샷을 선택한 다음 작업 목록에서 삭제를 선택합니다.
4. 예, 삭제를 선택합니다.

명령줄을 이용하여 스냅샷을 삭제하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [delete-snapshot\(AWS CLI\)](#)
- [Remove-EC2Snapshot\(Windows PowerShell용 AWS 도구\)](#)

다중 볼륨 스냅샷 삭제

다중 볼륨 스냅샷을 삭제하려면 스냅샷을 생성했을 때 그룹에 적용한 태그를 사용하여 다중 볼륨 그룹의 모든 스냅샷을 검색합니다. 그런 다음 스냅샷을 개별적으로 삭제합니다. 다중 볼륨 스냅샷 그룹의 개별 스냅샷을 삭제할 수 있습니다.

Amazon EBS 스냅샷 복사

Amazon EBS를 사용하면 볼륨의 특정 시점 스냅샷을 생성할 수 있으며 이 스냅샷은 Amazon S3에 저장됩니다. 스냅샷을 생성하고 Amazon S3로 복사를 완료한 경우(스냅샷 상태가 completed일 때) 이를 하나의 AWS 리전에서 다른 리전으로 또는 동일한 리전 내에서 복사할 수 있습니다. Amazon S3 서버 측 암호화(256비트 AES)는 복사 작업 중에 전송 중인 스냅샷 데이터를 보호합니다. 스냅샷 복사본은 원본 스냅샷의 ID와 다른 ID를 받습니다.

다중 볼륨 스냅샷을 다른 AWS 리전에 복사하려면 스냅샷을 생성했을 때 다중 볼륨 스냅샷 그룹에 적용한 태그를 사용하여 스냅샷을 검색합니다. 그런 다음 스냅샷을 다른 리전에 개별적으로 복사합니다.

Amazon RDS 스냅샷 복사에 대한 자세한 내용은 Amazon RDS 사용 설명서의 [DB 스냅샷 복사](#)를 참조하십시오.

다른 계정에서도 스냅샷을 복사할 수 있도록 하고 싶다면 해당 계정에 액세스할 수 있도록 스냅샷 권한을 변경하거나 모든 AWS 계정이 복사할 수 있도록 스냅샷을 퍼블릭으로 설정해야 합니다. 자세한 내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

AWS 리전 및 계정 간의 스냅샷 복사와 관련된 요금 정보는 [Amazon EBS 요금](#)을 참조하십시오. 단일 계정 및 리전 내에서의 스냅샷 복사 작업은 스냅샷 복사본의 암호화 상태가 변경되지 않을 경우 실제로 데이터가 복사되지 않기 때문에 아무런 비용도 발생하지 않습니다.

Note

새 지역으로 스냅샷을 복사하면 항상 전체(증분 아님) 복사가 생성되고 추가 지역 및 스토리지 비용이 발생합니다.

Note

새 CMK로 스냅샷을 복사 및 암호화하면 항상 전체(증분 아님) 복사가 생성되고 추가 지역 및 스토리지 비용이 발생합니다.

사용 사례

- **지리적 확장:** 새 AWS 리전에서 애플리케이션 시작.
- **マイ그레이션:** 새 리전으로 애플리케이션을 마이그레이션하여 가용성을 향상하고 비용을 최소화.
- **재해 복구:** 서로 다른 지리적 위치에 있는 데이터와 로그를 정기적인 시간 간격으로 백업. 재난 복구의 경우 보조 리전에 저장된 특정 시점 백업을 사용하여 애플리케이션을 복구할 수 있습니다. 이를 통해 데이터 손실 및 복구 시간이 최소화됩니다.

- 암호화: 이전에 암호화되지 않은 스냅샷을 암호화하고 스냅샷 암호화 시 사용한 키를 변경하거나, 자신과 공유된 암호화된 스냅샷의 경우에는 자기 소유의 복사본에서 볼륨을 생성하기 위해 해당 복사본을 생성합니다.
- 데이터 보존 및 감사 요구 사항: 한 AWS 계정에서 다른 AWS 계정으로 암호화된 EBS 스냅샷을 복사하여 데이터 로그나 감사 또는 데이터 보존을 위한 다른 파일을 보존합니다. 다른 계정을 사용하면 실수로 스냅샷을 삭제하는 것을 방지하고 기본 AWS 계정에 문제가 생길 경우 보호하는 데 도움이 됩니다.

사전 조건

- 공유 스냅샷 및 사용자가 생성한 스냅샷 등 completed 상태인 액세스 가능 스냅샷을 복사할 수 있습니다.
- AWS Marketplace, VM Import/Export 및 AWS Storage Gateway 스냅샷을 복사할 수 있지만 대상 리전에서 해당 스냅샷이 지원되는지 확인해야 합니다.

제한

- 각 계정에서는 단일 대상 리전으로 최대 20개의 동시 스냅샷 복사를 요청할 수 있습니다.
- 사용자 정의 태그는 원본 스냅샷에서 새로운 스냅샷으로 복사되지 않습니다. 복사 작업 도중이나 이후에 사용자 정의 태그를 추가할 수 있습니다. 자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.
- CopySnapshot 작업을 통해 생성된 스냅샷에는 어떠한 용도로도 사용할 수 없는 임의 볼륨 ID가 있습니다.

증분 스냅샷 복사

스냅샷 복사가 증분 복사인지 여부는 가장 최근 완료된 스냅샷 복사에 따라 결정됩니다. 리전 또는 계정 간에 스냅샷을 복사할 때 다음 조건이 충족되면 복사를 증분 복사입니다.

- 이전에 스냅샷이 대상 리전 또는 계정에 복사되었습니다.
- 가장 최근 스냅샷 복사는 여전히 대상 리전 또는 계정에 있습니다.
- 대상 리전 또는 계정에 있는 스냅샷의 모든 복사는 암호화되지 않거나 동일한 CMK를 사용하여 암호화되었습니다.

가장 최근 스냅샷 복사본이 삭제된 경우 다음 복사본은 전체 복사이며, 증분 복사가 아닙니다. 다른 복사를 시작할 때 복사가 보류 중인 경우 첫 번째 복사가 완료된 후에만 두 번째 복사가 시작됩니다.

대상 리전 또는 계정에서 볼륨의 가장 최근 스냅샷 복사를 추적할 수 있도록 스냅샷에 볼륨 ID와 생성 시간을 표시하는 태그를 지정하는 것이 좋습니다.

스냅샷 복사가 증분인지 확인하려면 [copySnapshot \(p. 1087\)](#) CloudWatch 이벤트를 점검하십시오.

암호화 및 스냅샷 복사

스냅샷을 복사할 때는 복사본을 암호화하거나 원본과 다른 CMK(고객 마스터 키)를 지정할 수 있습니다. 그러면 복사된 스냅샷은 새로운 CMK를 사용합니다. 하지만 복사 작업 중 스냅샷의 암호화 상태를 변경하면 증분식이 아닌 전체 복사본이 생성되어 많은 양의 데이터가 전송되고 스토리지 요금이 많이 발생할 수 있습니다.

다른 AWS 계정에서 공유한 암호화된 스냅샷을 복사하려면 해당 스냅샷의 사용 권한과 함께 스냅샷 암호화에 사용된 고객 마스터 키(CMK) 사용 권한도 필요합니다. 자신에게 공유해 준 암호화된 스냅샷을 사용할 때는 자체 CMK로 스냅샷을 복사하여 다시 암호화하는 것이 좋습니다. 이를 통해 원본 CMK가 손상되거나 소유자가 키를 취소하는 바람에 자신이 스냅샷으로 만든 어떤 암호화된 볼륨에도 액세스하지 못하게 되는 상황을 피할 수 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

Encrypted 파라미터를 true로 설정하여 EBS 스냅샷 복사본에 암호화를 적용합니다. [encryption by default\(암호화 기본 제공\) \(p. 1036\)](#)가 활성화되어 있는 경우 Encrypted 파라미터는 선택 사항입니다.

선택적으로 KmsKeyId를 사용하여 스냅샷 복사본을 암호화하는 데 사용할 사용자 지정 키를 지정할 수 있습니다. 기본 암호화가 설정되어 있어도 Encrypted 파라미터도 true로 설정해야 합니다. KmsKeyId가 지정되지 않은 경우 암호화에 사용되는 키는 소스 스냅샷 및 해당 소유권의 암호화 상태에 따라 달라집니다.

다음 표에서는 가능한 각 설정 조합에 대한 암호화 결과를 설명합니다.

암호화 결과: 스냅샷 복사

파라미터 Encrypted 설정 여부	암호화 기본 설정 여부	소스 스냅샷	기본값(KmsKeyId가 지 정되지 않음)	사용자 지정 (KmsKeyId가 지정됨)
아니요	아니요	암호화되지 않은 소유 스냅샷	암호화되지 않음	해당 사항 없음
아니요	아니요	암호화된 소유 스냅샷	동일한 키로 암호화됨	
아니요	아니요	암호화되지 않은 공유 스냅샷	암호화되지 않음	
아니요	아니요	암호화된 공유 스냅샷	기본 CMK로 암호화됨*	
예	아니요	암호화되지 않은 소유 스냅샷	기본 CMK로 암호화됨	지정된 CMK로 암호 화됨**
예	아니요	암호화된 소유 스냅샷	동일한 키로 암호화됨	
예	아니요	암호화되지 않은 공유 스냅샷	기본 CMK로 암호화됨	
예	아니요	암호화된 공유 스냅샷	기본 CMK로 암호화됨	
아니요	예	암호화되지 않은 소유 스냅샷	기본 CMK로 암호화됨	해당 사항 없음
아니요	예	암호화된 소유 스냅샷	동일한 키로 암호화됨	
아니요	예	암호화되지 않은 공유 스냅샷	기본 CMK로 암호화됨	
아니요	예	암호화된 공유 스냅샷	기본 CMK로 암호화됨	
예	예	암호화되지 않은 소유 스냅샷	기본 CMK로 암호화됨	지정된 CMK로 암호 화됨
예	예	암호화된 소유 스냅샷	동일한 키로 암호화됨	
예	예	암호화되지 않은 공유 스냅샷	기본 CMK로 암호화됨	
예	예	암호화된 공유 스냅샷	기본 CMK로 암호화됨	

* 이는 AWS 계정 및 리전의 EBS 암호화에 사용되는 기본 CMK입니다. 기본적으로 EBS에 대한 고유한 AWS 관리형 CMK이거나 고객 관리형 CMK를 지정할 수 있습니다. 자세한 내용은 [EBS 암호화에 대한 기본 키 \(p. 1035\)](#) 단원을 참조하십시오.

** 이는 복사 작업용으로 지정된 고객 관리형 CMK입니다. 이 CMK는 AWS 계정 및 리전에 대한 기본 CMK 대신 사용됩니다.

스냅샷 복사

다음 절차에 따라 Amazon EC2 콘솔을 사용하여 스냅샷을 복사합니다.

콘솔을 사용하여 스냅샷을 복사하려면,

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스냅샷을 선택합니다.
3. 복사할 스냅샷을 선택한 다음 작업 목록에서 복사를 선택합니다.
4. 스냅샷 복사 대화 상자에서 필요한 경우 다음을 업데이트합니다.
 - 대상 리전: 스냅샷 사본을 작성할 리전 선택.
 - 설명: 기본적으로, 설명에는 소스 스냅샷에 대한 정보가 포함되어 사용자는 원본과 사본을 구분할 수 있습니다. 필요한 경우 이 설명을 수정할 수 있습니다.
 - 암호화: 원본 스냅샷이 암호화되지 않은 경우에는 사본을 암호화할 수 있습니다. [암호화 기본 제공 \(p. 1036\)](#)을 활성화한 경우 암호화 옵션이 설정되어 스냅샷 콘솔로부터 설정 해제할 수 없습니다. 암호화 옵션이 설정된 경우, 아래 설명된 필드에서 하나를 선택하여 고객 관리형 CMK로 암호화하도록 선택할 수 있습니다.

암호화된 스냅샷에서 암호화를 제거할 수 없습니다.

Note

새 CMK로 스냅샷을 복사 및 암호화하면 항상 전체(증분 아님) 복사가 생성되고 추가 지연 및 스토리지 비용이 발생합니다.

- 마스터 키: 이 스냅샷을 암호화하는 데 사용할 고객 마스터 키(CMK). 계정의 기본 키가 처음에는 표시되지만 선택적으로 계정의 마스터 키를 선택하거나 다른 계정의 키 ARN을 입력/붙여넣기 할 수 있습니다. 새로운 마스터 암호화 키는 IAM 콘솔 <https://console.aws.amazon.com/iam/>에서 생성할 수 있습니다.
5. 복사를 선택합니다.
6. 스냅샷 복사 확인 대화 상자에서 스냅샷을 선택해 지정된 리전의 스냅샷 페이지로 이동하거나 닫기를 선택합니다.

복사 프로세스 진행률을 확인하려면 대상 리전으로 전환한 다음, 스냅샷 페이지를 새로 고칩니다. 복사 진행률이 페이지 상단에 표시됩니다.

오류를 확인하려면

암호화 키 사용 권한 없이 암호화된 스냅샷을 복사하려고 하면 작업이 자동으로 실패합니다. 페이지를 새로 고칠 때까지는 콘솔에 오류 상태가 표시되지 않습니다. 또한 다음 예와 같이 명령줄에서 스냅샷의 상태를 확인할 수 있습니다.

```
aws ec2 describe-snapshots --snapshot-id snap-0123abcd
```

키 권한 부족으로 복사에 실패한 경우에는 "StateMessage": "Given key ID is not accessible" 메시지가 표시됩니다.

암호화된 스냅샷을 복사하려면 기본 CMK에 대한 `DescribeKey` 권한이 있어야 합니다. 이러한 권한을 명시적으로 거부하면 복사에 실패합니다. CMK 키 관리에 대한 자세한 내용은 [고객 마스터 키에 대한 액세스 제어](#)를 참조하십시오.

명령줄을 이용하여 스냅샷을 복사하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [copy-snapshot\(AWS CLI\)](#)
- [Copy-EC2Snapshot\(Windows PowerShell용 AWS 도구\)](#)

Amazon EBS 스냅샷 정보 보기

스냅샷에 대한 세부 정보를 볼 수 있습니다.

콘솔을 사용하여 스냅샷 정보를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스냅샷을 선택합니다.
3. 목록을 줄이려면 필터 목록에서 옵션을 선택합니다. 예를 들어, 자신이 소유한 스냅샷만을 확인하려면 내 소유를 선택합니다. 태그와 스냅샷 속성을 사용하여 스냅샷을 필터링할 수도 있습니다. 검색 창을 선택하여 사용 가능한 태그와 속성을 봅니다.
4. 스냅샷에 대한 자세한 정보를 확인하려면 선택합니다.

명령줄을 사용하여 스냅샷 정보를 보려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [describe-snapshots\(AWS CLI\)](#)
- [Get-EC2Snapshot\(Windows PowerShell용 AWS 도구\)](#)

Example 예: 태그를 기준으로 필터링

다음 명령은 Stack=production 태그를 가진 스냅샷을 설명합니다.

```
aws ec2 describe-snapshots --filters Name>tag:Stack,Values=production
```

Example 예: 볼륨을 기준으로 필터링

다음 명령은 지정된 볼륨에서 생성된 스냅샷을 설명합니다.

```
aws ec2 describe-snapshots --filters Name=volume-id,Values=vol-049df61146c4d7901
```

Example 예: 스냅샷 경과 시간을 기준으로 필터링

AWS CLI를 사용하면 JMESPath를 통해 표현식을 사용하여 결과를 필터링할 수 있습니다. 예를 들어, 다음 명령은 지정된 날짜([2020-03-31](#)로 표시) 이전에 AWS 계정에서 생성된 모든 스냅샷의 ID([123456789012](#)로 표시)를 표시합니다. 소유자를 지정하지 않으면 모든 퍼블릭 스냅샷이 결과에 포함됩니다.

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query "Snapshots[?(StartTime<=`2020-03-31`)].[SnapshotId]" --output text
```

다음 명령은 지정된 날짜 범위에 생성된 모든 스냅샷의 ID를 표시합니다.

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query "Snapshots[?(StartTime>=`2019-01-01` && (StartTime<=`2019-12-31`)].[SnapshotId]" --output text
```

Amazon EBS 스냅샷 공유

스냅샷 권한을 수정하여 본인이 지정한 AWS 계정과 공유할 수 있습니다. 허가된 사용자는 원본 스냅샷에 아무런 영향을 주지 않으면서 공유된 스냅샷을 토대로 자체 EBS 볼륨을 생성할 수 있습니다.

원한다면 암호화되지 않은 스냅샷을 모든 AWS 사용자에게 공개할 수 있습니다. 암호화된 스냅샷은 공개적으로 사용 가능하게 만들 수 없습니다.

암호화된 스냅샷을 공유할 때는 해당 스냅샷을 암호화하는 데 사용된 고객 관리형 CMK도 공유해야 합니다. 생성 당시에 또는 나중에 고객 관리형 CMK에 교차 계정 권한을 적용할 수 있습니다.

Important

스냅샷을 공유하면 다른 사람들이 해당 스냅샷의 모든 데이터에 액세스할 수 있게 됩니다. 그러므로 스냅샷의 모든 데이터를 공유하려는 사용자하고만 스냅샷을 공유하십시오.

고려 사항

다음은 스냅샷을 공유할 때 고려할 사항입니다.

- 스냅샷은 생성된 리전으로 제한됩니다. 스냅샷을 다른 리전과 공유하려면 스냅샷을 해당 리전에 복사합니다. 자세한 내용은 [Amazon EBS 스냅샷 복사 \(p. 985\)](#) 단원을 참조하십시오.
- 스냅샷에서 더 긴 리소스 ID 형식을 사용하는 경우 더 긴 ID를 지원하는 다른 계정과 공유해야 합니다. 자세한 내용은 [Resource ID \(p. 1133\)](#)을 참조하십시오.
- AWS는 기본 CMK로 암호화된 스냅샷의 공유를 금지합니다. 그 대신, 고객 관리형 CMK로 공유하려는 스냅샷을 암호화해야 합니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [키 생성](#) 단원을 참조하십시오.
- 공유된 CMK 사용자가 암호화된 스냅샷에 액세스하는 경우, 키에 대해 `kms:DescribeKey`, `kms:CreateGrant`, `GenerateDataKey` 및 `kms:ReEncrypt` 작업을 수행할 수 있는 권한을 부여해야 합니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [사용자 지정 마스터 키에 대한 액세스 제어](#)를 참조하십시오.

콘솔을 사용하여 암호화되지 않은 스냅샷 공유

콘솔을 사용해 스냅샷을 공유하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 스냅샷을 선택합니다.
- 스냅샷을 선택한 다음 작업과 권한 설정을 선택합니다.
- 다음과 같이 스냅샷을 퍼블릭 스냅샷으로 만들거나 특정 AWS 계정과 공유합니다.
 - 스냅샷을 퍼블릭으로 설정하려면 퍼블릭을 선택합니다.

암호화된 스냅샷 또는 AWS Marketplace 제품 코드가 있는 스냅샷에는 이 옵션을 사용할 수 없습니다.

- 하나 이상의 AWS 계정과 스냅샷을 공유하려면 프라이빗을 선택하고 AWS 계정 번호에 AWS 계정 ID(아이픈 제외)를 입력한 다음 권한 추가를 선택합니다. 모든 추가 AWS 계정에 대해 반복합니다.

- Save를 선택합니다.

본인에게 개인적으로 공유해 준 암호화되지 않은 스냅샷을 사용하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 스냅샷을 선택합니다.

3. 프라이빗 스냅샷 필터를 선택합니다.
4. ID 또는 설명으로 해당 스냅샷을 찾습니다. 이 스냅샷도 다른 스냅샷처럼 사용할 수 있습니다. 예를 들어, 이 스냅샷으로 볼륨을 생성하거나 스냅샷을 다른 리전으로 복사할 수 있습니다.

콘솔을 사용하여 암호화된 스냅샷 공유

콘솔을 사용하여 암호화된 스냅샷을 공유하려면

1. <https://console.aws.amazon.com/kms>에서 AWS KMS 콘솔을 엽니다.
2. AWS 리전을 변경하려면 페이지의 오른쪽 위 모서리에 있는 리전 선택기를 사용합니다.
3. 탐색 창에서 Customer managed keys(고객 관리형 키)를 선택합니다.
4. 별칭 열에서 스냅샷을 암호화하는 데 사용한 고객 관리형 키의 별칭(텍스트 링크)을 선택합니다. 키 세부 정보가 새 페이지에서 열립니다.
5. Key policy(키 정책) 섹션에는 정책 보기 또는 기본 보기 표시됩니다. 정책 보기에는 주요 정책 문서가 표시됩니다. 기본 보기에는 키 관리자, 키 삭제, 키 사용 및 기타 AWS 계정에 대한 섹션에 표시됩니다. 콘솔에서 정책을 생성하고 사용자 지정하지 않은 경우 기본 보기 표시됩니다. 기본 보기 사용할 수 없는 경우 정책 보기에서 정책을 수동으로 편집해야 합니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [키 정책 보기\(콘솔\)](#)를 참조하십시오.

이용 가능한 보기에 따라 정책 보기 또는 기본 보기 사용하여 다음과 같이 하나 이상의 AWS 계정 ID를 정책에 추가합니다.

- (정책 보기) Edit(편집)를 선택합니다. "Allow use of the key" 및 "Allow attachment of persistent resources" 명령문에 하나 이상의 AWS 계정 ID를 추가합니다. [Save changes]를 선택합니다. 다음 예에서는 AWS 계정 ID 444455556666이 정책에 추가됩니다.

```
{  
    "Sid": "Allow use of the key",  
    "Effect": "Allow",  
    "Principal": {"AWS": [  
        "arn:aws:iam::1112222333:user/CMKUser",  
        "arn:aws:iam::444455556666:root"  
    ]},  
    "Action": [  
        "kms:Encrypt",  
        "kms:Decrypt",  
        "kms:ReEncrypt*",  
        "kms:GenerateDataKey*",  
        "kms:DescribeKey"  
    ],  
    "Resource": "*"  
},  
{  
    "Sid": "Allow attachment of persistent resources",  
    "Effect": "Allow",  
    "Principal": {"AWS": [  
        "arn:aws:iam::1112222333:user/CMKUser",  
        "arn:aws:iam::444455556666:root"  
    ]},  
    "Action": [  
        "kms>CreateGrant",  
        "kms>ListGrants",  
        "kms:RevokeGrant"  
    ],  
    "Resource": "*",  
    "Condition": {"Bool": {"kms:GrantIsForAWSResource": true}}  
}
```

- (기본 보기) 아래로 스크롤하여 Other AWS accounts(다른 AWS 계정)로 이동합니다. Add other AWS accounts(다른 AWS 계정 추가)를 선택하고 메시지가 표시되면 AWS 계정 ID를 입력합니다.

다른 계정을 추가하려면 Add another AWS account(다른 AWS 계정 추가)를 선택하고 AWS 계정 ID를 입력합니다. AWS 계정을 모두 추가했으면 변경 사항 저장을 선택합니다.

6. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
7. 탐색 창에서 스냅샷을 선택합니다.
8. 스냅샷을 선택한 다음 작업과 권한 설정을 선택합니다.
9. 각각의 AWS 계정에 대해 AWS 계정 번호에 AWS 계정 ID를 입력하고 권한 추가를 선택합니다. AWS 계정을 모두 추가했으면 저장을 선택합니다.

본인에게 공유해 준 암호화된 스냅샷을 사용하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 스냅샷을 선택합니다.
3. 프라이빗 스냅샷 필터를 선택합니다. 원한다면 암호화 필터도 추가합니다.
4. ID 또는 설명으로 해당 스냅샷을 찾습니다.
5. 스냅샷을 선택하고 작업 및 복사를 선택합니다.
6. (선택 사항) 대상 리전을 선택합니다.
7. 스냅샷의 사본은 마스터 키에 표시된 키로 암호화됩니다. 기본적으로 선택한 키는 계정의 기본 CMK입니다. 고객 관리형 CMK를 선택하려면 입력 상자 안을 클릭하여 사용 가능한 키 목록을 조회합니다.
8. 복사를 선택합니다.

명령줄을 사용하여 스냅샷 공유

스냅샷에 대한 권한은 스냅샷의 `createVolumePermission` 속성을 사용하여 지정됩니다. 스냅샷을 퍼블릭으로 설정하려면 그룹을 `all`로 설정합니다. 특정 AWS 계정과 스냅샷을 공유하려면 사용자를 AWS 계정 ID로 설정합니다.

명령줄을 이용하여 스냅샷 권한을 수정하려면

다음 명령 중 하나를 사용합니다.

- [modify-snapshot-attribute\(AWS CLI\)](#)
- [Edit-EC2SnapshotAttribute\(Windows PowerShell용 AWS 도구\)](#)

명령줄을 사용하여 스냅샷 권한을 보려면

다음 명령 중 하나를 사용합니다.

- [describe-snapshot-attribute\(AWS CLI\)](#)
- [Get-EC2SnapshotAttribute\(Windows PowerShell용 AWS 도구\)](#)

명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

공유 스냅샷 사용 확인

AWS CloudTrail를 사용하여 다른 사용자와 공유한 스냅샷이 복사되거나 볼륨을 생성하는데 사용되는지를 모니터링할 수 있습니다. 다음 이벤트가 CloudTrail에 기록됩니다.

- `SharedSnapshotCopyInitiated` — 공유 스냅샷이 복사되고 있습니다.
- `SharedSnapshotVolumeCreated` — 공유 스냅샷이 볼륨을 생성하는 데 사용되고 있습니다.

CloudTrail 사용에 관한 자세한 내용은 [AWS CloudTrail을 사용하여 Amazon EC2 및 Amazon EBS API 호출 로깅 \(p. 697\)](#) 단원을 참조하십시오.

EBS 스냅샷 콘텐츠 액세스

Amazon Elastic Block Store(Amazon EBS) direct API를 사용하여 EBS 스냅샷을 생성하고, 스냅샷에 직접 데이터를 쓰고, 스냅샷에서 데이터를 읽고, 두 스냅샷 간의 차이점 또는 변경 사항을 파악할 수 있습니다. Amazon EBS용 백업 서비스를 제공하는 Independent Software Vendor(ISV)는 EBS 디렉트 API를 사용하여 스냅샷을 통해 더 효율적이고 비용 효과적으로 EBS 볼륨에 대한 충분 변경 사항을 추적할 수 있습니다. 스냅샷에서 새 볼륨을 만들지 않고도 이 작업이 가능하며 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스를 사용하여 차이를 비교할 수 있습니다.

온프레미스의 데이터에서 직접 EBS 볼륨 및 클라우드로 충분 스냅샷을 생성하고 이를 사용해 신속하게 재해 복구를 수행할 수 있습니다. 스냅샷을 쓰고 읽을 수 있으므로 재해 발생 시 온프레미스 데이터를 EBS 스냅샷에 쓸 수 있습니다. 그리고 복구 후에는 스냅샷에서 AWS 또는 온프레미스로 복원할 수 있습니다. Amazon EBS로/에서 데이터를 복사하기 위해 더 이상 복잡한 메커니즘을 빌드하고 유지 관리할 필요가 없습니다.

이 사용 설명서에는 EBS 디렉트 API를 구성하는 요소의 개요와 이 요소를 효과적으로 사용하는 방법의 예가 나와 있습니다. API의 작업, 데이터 형식, 파라미터 및 오류에 대한 자세한 내용은 [EBS 디렉트 API 참조](#)를 참조하십시오. EBS 디렉트 API에 대해 지원되는 AWS 리전, 엔드포인트 및 서비스 할당량은 AWS General Reference의 [Amazon EBS 엔드포인트 및 할당량](#)을 참조하십시오.

목차

- [EBS 디렉트 API 이해 \(p. 993\)](#)
- [IAM 사용자의 권한 \(p. 996\)](#)
- [암호화 사용 \(p. 1000\)](#)
- [Signature 버전 4 서명 사용 \(p. 1000\)](#)
- [체크섬 사용 \(p. 1000\)](#)
- [API 또는 AWS SDK를 사용하여 EBS 디렉트 API 작업 \(p. 1001\)](#)
- [명령줄을 사용한 EBS 디렉트 API 작업 \(p. 1005\)](#)
- [성능 최적화 \(p. 1008\)](#)
- [FAQ \(p. 1009\)](#)
- [AWS CloudTrail을 사용하여 EBS 디렉트 API API 호출 로깅 \(p. 1010\)](#)
- [StartSnapshot API에 대한 명령 \(p. 1012\)](#)

EBS 디렉트 API 이해

EBS 디렉트 API를 시작하려면 먼저 다음과 같은 핵심 요소를 이해해야 합니다.

요금

EBS 디렉트 API를 사용하기 위해 지불하는 요금은 요청하는 내용에 따라 다릅니다. 자세한 내용은 [Amazon EBS 요금](#)을 참조하십시오.

스냅샷

스냅샷은 EBS 볼륨에서 데이터를 백업하는 기본 방법입니다. EBS 디렉트 API를 사용하면 온프레미스 디스크의 데이터를 스냅샷으로 백업할 수도 있습니다. 스토리지 비용을 절약하기 위해 이전 스냅샷 이후로 변경된 볼륨 데이터만 연속 스냅샷에 충분식으로 포함시킵니다. 자세한 내용은 [Amazon EBS 스냅샷 \(p. 966\)](#) 단원을 참조하십시오.

Note

EBS 디렉트 API에서는 퍼블릭 스냅샷이 지원되지 않습니다.

블록

블록은 스냅샷에 있는 데이터의 조각입니다. 각 스냅샷에 수천 개의 블록을 포함할 수 있습니다. 스냅샷에 있는 모든 블록은 크기가 고정되어 있습니다.

블록 인덱스

블록 인덱스는 스냅샷에 있는 블록의 오프셋 위치이며 블록을 식별하는 데 사용됩니다. 논리 볼륨에 있는 데 이터의 논리적 오프셋을 구하려면 BlockIndex 값과 BlockSize 값을 곱합니다(BlockIndex * BlockSize).

블록 토큰

블록 토큰은 스냅샷에 있는 블록의 식별 해시이며 블록 데이터를 찾는 데 사용됩니다. EBS 디렉트 API에서 반환된 블록 토큰은 임시로, 지정된 만료 타임스탬프에 변경되거나 동일한 스냅샷에 대해 다른 ListSnapshotBlocks 또는 ListChangedBlocks 요청을 실행하면 변경됩니다.

체크섬

체크섬은 전송 또는 저장 중에 발생한 오류를 탐지하기 위해 데이터 블록에서 제공되는 작은 크기의 데이터입니다. EBS 디렉트 API는 데이터 무결성을 검증하기 위해 체크섬을 사용합니다. EBS 스냅샷에서 데이터를 읽을 때는 이 서비스가 전송된 각 데이터 블록에 대해 Base64로 인코딩된 SHA256 체크섬을 제공하며, 이 체크섬을 검증에 사용할 수 있습니다. EBS 스냅샷에 데이터를 쓸 때는 사용자가 전송된 각 데이터 블록에 대해 Base64로 인코딩된 SHA256 체크섬을 제공해야 합니다. 그러면 이 서비스가 제공된 체크섬을 사용하여 수신된 데이터를 검증합니다. 자세한 내용은 이 설명서 후반부의 [체크섬 사용 \(p. 1000\)](#) 단원을 참조하십시오.

암호화

암호화는 읽을 수 없는 코드로 데이터를 변환하고 암호화에 사용되는 키에 액세스할 수 있는 사용자만 코드를 해독할 수 있도록 하여 데이터를 보호합니다. EBS 디렉트 API를 사용하여 암호화된 스냅샷을 읽고 쓸 수 있지만 몇 가지 제한 사항이 있습니다. 자세한 내용은 이 설명서 후반부의 [암호화 사용 \(p. 1000\)](#) 단원을 참조하십시오.

API 작업

EBS 디렉트 API는 6가지 작업으로 구성됩니다. 3가지 읽기 작업과 3가지 쓰기 작업이 있습니다. 읽기 작업은 ListSnapshotBlocks, ListChangedBlocks, GetSnapshotBlock이고, 쓰기 작업은 StartSnapshot, PutSnapshotBlock, CompleteSnapshot입니다. 이러한 작업에 대해서는 다음 단원에서 설명합니다.

스냅샷 블록 나열

ListSnapshotBlocks 작업은 지정된 스냅샷에 있는 블록의 블록 인덱스와 블록 토큰을 반환합니다.

변경된 블록 나열

ListChangedBlocks 작업은 볼륨/스냅샷 계보가 같은 2개의 지정된 스냅샷에서 차이가 있는 블록의 블록 인덱스와 블록 토큰을 반환합니다.

스냅샷 블록 가져오기

GetSnapshotBlock 작업은 지정된 스냅샷 ID, 블록 인덱스 및 블록 토큰의 블록 내 데이터를 반환합니다.

스냅샷 시작

StartSnapshot 작업은 기존 스냅샷의 증분 스냅샷 또는 새 스냅샷으로 스냅샷을 시작합니다. 시작된 스냅샷은 CompleteSnapshot 작업 사용을 마칠 때까지 대기 중 상태로 유지됩니다.

스냅샷 블록 추가

PutSnapshotBlock 작업은 개별 블록 형태로 시작된 스냅샷에 데이터를 추가합니다. 전송되는 데이터의 블록에 대해 Base64 인코딩 SHA256 체크섬을 지정해야 합니다. 이 서비스는 전송이 완료되고 나면 체크섬을 검증합니다. 서비스에서 계산된 체크섬이 지정한 체크섬과 일치하지 않으면 요청이 실패합니다.

스냅샷 완료

CompleteSnapshot 작업은 대기 중 상태인 시작된 스냅샷을 완료합니다. 그러면 스냅샷이 완료됨 상태로 바뀝니다.

EBS 디렉트 API를 사용하여 스냅샷 읽기

다음 단계에서는 EBS 디렉트 API를 사용하여 스냅샷을 읽는 방법을 설명합니다.

1. ListSnapshotBlocks 작업을 사용하여 스냅샷에 있는 블록의 모든 블록 인덱스와 블록 토큰을 표시합니다. 또는 ListChangedBlocks 작업을 사용하여 볼륨/스냅샷 계보가 같은 2개의 지정된 스냅샷에서 차이가 있는 블록의 블록 인덱스와 블록 토큰만 표시합니다. 이러한 작업을 통해 데이터를 가져올 블록의 블록 토큰 및 블록 인덱스를 확인할 수 있습니다.
2. GetSnapshotBlock 작업을 사용하고 데이터를 가져올 블록의 블록 인덱스 및 블록 토큰을 지정합니다.

이러한 작업을 실행하는 방법에 대한 예는 이 설명서 후반부의 [API 또는 AWS SDK를 사용하여 EBS 디렉트 API 작업 \(p. 1001\)](#) 및 [명령줄을 사용한 EBS 디렉트 API 작업 \(p. 1005\)](#) 단원을 참조하십시오.

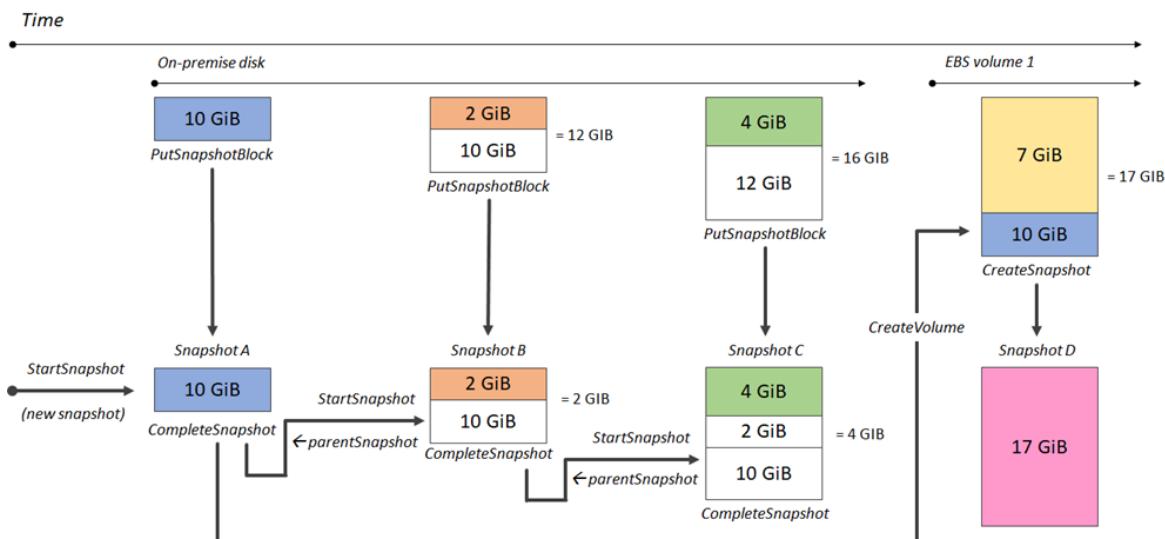
EBS 디렉트 API를 사용하여 종분 스냅샷 쓰기

다음 단계에서는 EBS 디렉트 API를 사용하여 종분 스냅샷을 쓰는 방법을 설명합니다.

1. StartSnapshot 작업을 사용하고 상위 스냅샷 ID를 지정하여 기존 스냅샷의 종분 스냅샷으로 스냅샷을 시작하거나 상위 스냅샷 ID를 생략하여 새 스냅샷을 시작합니다. 이 작업은 대기 중 상태의 새 스냅샷 ID를 반환합니다.
2. PutSnapshotBlock 작업을 사용하고 대기 중인 스냅샷의 ID를 지정하여 개별 블록 형식으로 데이터를 추가합니다. 전송되는 데이터의 블록에 대해 Base64 인코딩 SHA256 체크섬을 지정해야 합니다. 이 서비스는 수신된 데이터의 체크섬을 계산하고 지정된 체크섬과 대조하여 검증합니다. 체크섬이 일치하지 않으면 작업이 실패합니다.
3. 대기 중인 스냅샷에 데이터를 추가했으면 CompleteSnapshot 작업을 사용하여 스냅샷을 봉인하고 완료됨 상태로 전환하는 비동기 워크플로우를 시작합니다.

이 단계를 반복하여 이전에 생성한 스냅샷을 상위 스냅샷으로 사용해 새 종분 스냅샷을 생성합니다.

예를 들어 다음 디어그램에서 스냅샷 A는 처음 시작된 새 스냅샷입니다. 스냅샷 A는 스냅샷 B를 시작하는데 상위 스냅샷으로 사용됩니다. 스냅샷 B는 스냅샷 C를 시작하고 생성하는데 상위 스냅샷으로 사용됩니다. 스냅샷 A, B, C는 종분 스냅샷입니다. 스냅샷 A는 EBS 볼륨 1을 생성하는 데 사용됩니다. 스냅샷 D는 EBS 볼륨 1에서 생성됩니다. 스냅샷 D는 A의 종분 스냅샷이며 B 또는 C의 종분 스냅샷이 아닙니다.



이러한 작업을 실행하는 방법에 대한 예는 이 설명서 후반부의 [API 또는 AWS SDK를 사용하여 EBS 디렉트 API 작업 \(p. 1001\)](#) 및 [명령줄을 사용한 EBS 디렉트 API 작업 \(p. 1005\)](#) 단원을 참조하십시오.

IAM 사용자의 권한

AWS Identity and Access Management(IAM) 사용자는 다음 정책이 있어야 EBS 디렉트 API를 사용할 수 있습니다. 자세한 내용은 [IAM 사용자의 권한 변경](#)을 참조하십시오.

IAM 사용자에게 다음 정책을 할당할 때는 주의해야 합니다. 이 정책을 할당하면 CopySnapshot 또는 CreateVolume 작업과 같은 Amazon EC2 API를 통해 동일한 리소스에 액세스하는 것이 거부된 사용자에게 액세스 권한을 부여할 수 있습니다.

스냅샷 읽기 권한

다음 정책은 특정 AWS 리전의 모든 스냅샷에 읽기 EBS 디렉트 API를 사용할 수 있도록 허용합니다. 정책에서 <Region>을 스냅샷의 리전으로 바꾸십시오.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ebs>ListSnapshotBlocks",  
                "ebs>ListChangedBlocks",  
                "ebs>GetSnapshotBlock"  
            ],  
            "Resource": "arn:aws:ec2:<Region>::snapshot/*"  
        }  
    ]  
}
```

다음 정책은 특정 키-값 태그가 있는 스냅샷에 읽기 EBS 디렉트 API를 사용할 수 있도록 허용합니다. 정책에서 <Key>를 태그의 키 값으로 바꾸고 <Value>를 태그 값으로 바꾸십시오.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ebs>ListSnapshotBlocks",  
                "ebs>ListChangedBlocks",  
                "ebs>GetSnapshotBlock"  
            ],  
            "Resource": "arn:aws:ec2:*::snapshot/*",  
            "Condition": {  
                "StringEqualsIgnoreCase": {  
                    "aws:ResourceTag/<Key>": "<Value>"  
                }  
            }  
        }  
    ]  
}
```

다음 정책은 특정 시간 범위 내에서만 계정의 모든 스냅샷에 읽기 EBS 디렉트 API를 사용할 수 있도록 허용합니다. 이 정책은 aws:CurrentTime 전역 조건 키를 기반으로 EBS 디렉트 API를 사용할 수 있는 권한을 부여합니다. 정책에서 표시된 날짜 및 시간 범위를 정책의 날짜 및 시간 범위로 바꿔야 합니다.

```
{  
    "Version": "2012-10-17",
```

```
"Statement": [
    {
        "Effect": "Allow",
        "Action": [
            "ebs>ListSnapshotBlocks",
            "ebs>ListChangedBlocks",
            "ebs>GetSnapshotBlock"
        ],
        "Resource": "arn:aws:ec2::snapshot/*",
        "Condition": {
            "DateGreaterThan": {
                "aws:CurrentTime": "2018-05-29T00:00:00Z"
            },
            "DateLessThan": {
                "aws:CurrentTime": "2020-05-29T23:59:59Z"
            }
        }
    }
]
```

다음 정책은 AWS Key Management Service(AWS KMS)에서 특정 키 ID를 사용하여 암호화된 스냅샷을 해독할 수 있는 액세스 권한을 부여합니다. EBS 스냅샷의 기본 AWS KMS 키 ID를 사용하여 새 스냅샷을 암호화할 수 있는 액세스 권한을 부여합니다. 또한 계정에서 암호화가 기본적으로 활성화되어 있는지 여부를 확인할 수 있도록 합니다. 정책에서 <Region>을 AWS KMS 키의 리전으로 바꾸고, <AccountId>를 키의 AWS 계정 ID로 바꾸고, <KeyId>를 EBS 디렉트 API로 읽을 스냅샷을 암호화하는 데 사용되는 키의 ID로 바꾸십시오.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:GenerateDataKey",
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:ReEncrypt*",
                "kms>CreateGrant",
                "ec2>CreateTags",
                "kms:DescribeKey",
                "ec2:GetEbsDefaultKmsKeyId",
                "ec2:GetEbsEncryptionByDefault"
            ],
            "Resource": "arn:aws:kms:<Region>:<AccountId>:key/<KeyId>"
        }
    ]
}
```

자세한 내용은 IAM 사용 설명서의 [IAM 사용자의 권한 변경](#)을 참조하십시오.

스냅샷 쓰기 권한

다음 정책은 특정 AWS 리전의 모든 스냅샷에 쓰기 EBS 디렉트 API를 사용할 수 있도록 허용합니다. 정책에서 <Region>을 스냅샷의 리전으로 바꾸십시오.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {

```

```
        "Effect": "Allow",
        "Action": [
            "ebs:StartSnapshot",
            "ebs:PutSnapshotBlock",
            "ebs:CompleteSnapshot"
        ],
        "Resource": "arn:aws:ec2:<Region>::snapshot/*"
    }
]
```

다음 정책은 특정 키-값 태그가 있는 스냅샷에 쓰기 EBS 디렉트 API를 사용할 수 있도록 허용합니다. 정책에서 **<Key>**를 태그의 키 값으로 바꾸고 **<Value>**를 태그 값으로 바꾸십시오.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ebs:StartSnapshot",
                "ebs:PutSnapshotBlock",
                "ebs:CompleteSnapshot"
            ],
            "Resource": "arn:aws:ec2::*::snapshot/*",
            "Condition": {
                "StringEqualsIgnoreCase": {
                    "aws:ResourceTag/<Key>": "<Value>"
                }
            }
        }
    ]
}
```

다음 정책은 모든 EBS 디렉트 API를 사용할 수 있도록 허용합니다. 또한 상위 스냅샷 ID가 지정된 경우에 한해 StartSnapshot 작업을 허용합니다. 즉, 이 정책은 상위 스냅샷을 사용하지 않고는 새 스냅샷을 시작하지 못하도록 차단합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ebs:*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ebs:ParentSnapshot": "arn:aws:ec2::*::snapshot/*"
                }
            }
        }
    ]
}
```

다음 정책은 모든 EBS 디렉트 API를 사용할 수 있도록 허용합니다. 또한 새 스냅샷에 대해 user 태그 키만 생성할 수 있도록 합니다. 또한 이 정책은 사용자가 태그를 생성할 수 있는 액세스 권한을 갖도록 합니다. StartSnapshot 작업은 태그를 지정할 수 있는 유일한 작업입니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        "Effect": "Allow",
        "Action": "ebs:/*",
        "Resource": "*",
        "Condition": {
            "ForAllValues:StringEquals": [
                "aws:TagKeys": "user"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

다음 정책은 특정 시간 범위 내에서만 계정의 모든 스냅샷에 쓰기 EBS 다이렉트 API를 사용할 수 있도록 허용합니다. 이 정책은 `aws:CurrentTime` 전역 조건 키를 기반으로 EBS 다이렉트 API를 사용할 수 있는 권한을 부여합니다. 정책에서 표시된 날짜 및 시간 범위를 정책의 날짜 및 시간 범위로 바꿔야 합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ebs:StartSnapshot",  
                "ebs:PutSnapshotBlock",  
                "ebs:CompleteSnapshot"  
            ],  
            "Resource": "arn:aws:ec2:*:snapshot/*",  
            "Condition": {  
                "DateGreaterThan": {  
                    "aws:CurrentTime": "2018-05-29T00:00:00Z"  
                },  
                "DateLessThan": {  
                    "aws:CurrentTime": "2020-05-29T23:59:59Z"  
                }  
            }  
        }  
    ]  
}
```

다음 정책은 AWS Key Management Service(AWS KMS)에서 특정 키 ID를 사용하여 암호화된 스냅샷을 해독할 수 있는 액세스 권한을 부여합니다. EBS 스냅샷의 기본 AWS KMS 키 ID를 사용하여 새 스냅샷을 암호화할 수 있는 액세스 권한을 부여합니다. 또한 계정에서 암호화가 기본적으로 활성화되어 있는지 여부를 확인할 수 있도록 합니다. 정책에서 `<Region>`을 AWS KMS 키의 리전으로 바꾸고, `<AccountId>`를 키의 AWS 계정 ID로 바꾸고, `<KeyId>`를 EBS 디렉트 API로 읽을 스냅샷을 암호화하는 데 사용되는 키의 ID로 바꾸십시오.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "VisualEditor0",  
            "Effect": "Allow",  
            "Action": [  
                "kms:Encrypt",  
                "kms:Decrypt",  
                "kms:GenerateDataKey",
```

```
    "kms:GenerateDataKeyWithoutPlaintext",
    "kms:ReEncrypt*",
    "kms>CreateGrant",
    "ec2>CreateTags",
    "kms:DescribeKey",
    "ec2:GetEbsDefaultKmsKeyId",
    "ec2:GetEbsEncryptionByDefault"
],
"Resource": "arn:aws:kms:<Region>:<AccountId>:key/<KeyId>"
}
]
```

자세한 내용은 IAM 사용 설명서의 [IAM 사용자의 권한 변경](#)을 참조하십시오.

암호화 사용

AWS 계정에서 Amazon EBS 암호화가 기본적으로 활성화되어 있는 경우 암호화되지 않은 상위 스냅샷을 사용하여 새 스냅샷을 시작할 수 없습니다. 먼저 상위 스냅샷을 복사하여 암호화해야 합니다. 자세한 내용은 [Amazon EBS 스냅샷 복사 \(p. 985\)](#) 및 [암호화 기본 제공 \(p. 1036\)](#) 단원을 참조하십시오.

암호화된 스냅샷을 시작하려면 AWS KMS 키의 Amazon 리소스 이름(ARN)을 지정하거나 StartSnapshot 요청에서 암호화된 상위 스냅샷을 지정합니다. 둘 다 지정되지 않고 계정에서 Amazon EBS 암호화가 기본적으로 활성화되어 있으면 계정의 기본 CMK가 사용됩니다. 계정에 대해 기본 CMK가 지정되지 않은 경우 AWS 관리형 CMK가 사용됩니다.

Important

기본적으로 계정의 모든 보안 주체는 기본 AWS 관리형 CMK에 액세스할 수 있으며 이를 EBS 암호화 및 암호 해독 작업에 사용할 수 있습니다. 자세한 내용은 [EBS 암호화에 대한 기본 키 \(p. 1035\)](#) 단원을 참조하십시오.

암호화와 함께 EBS 디렉트 API를 사용하려면 추가 IAM 권한이 필요할 수 있습니다. 자세한 내용은 이 설명서 앞부분에 있는 [IAM 사용자의 권한 \(p. 996\)](#) 단원을 참조하십시오.

Signature 버전 4 서명 사용

Signature 버전 4는 HTTP로 전송된 AWS 요청에 인증 정보를 추가하는 프로세스입니다. 보안을 위해 대부분의 AWS 요청은 액세스 키 ID와 보안 액세스 키로 구성된 액세스 키로 서명해야 합니다. 이 두 키는 일반적으로 보안 자격 증명이라고 합니다. 계정의 자격 증명을 얻는 방법에 대한 자세한 내용은 [자격 증명 이해 및 가져오기](#)를 참조하십시오.

HTTP 요청을 수동으로 생성하려는 경우 서명하는 방법을 알아야 합니다. AWS Command Line Interface(AWS CLI) 또는 AWS SDK 중 하나를 사용하여 AWS에 요청할 경우 도구를 구성할 때 지정한 액세스 키를 사용하여 요청에 자동으로 서명됩니다. 따라서 이러한 도구를 사용할 경우 요청에 서명하는 방법을 알 필요가 없습니다.

자세한 내용은 AWS General Reference의 [서명 버전 4를 사용하여 AWS 요청에 서명](#)을 참조하세요.

체크섬 사용

GetSnapshotBlock 작업은 스냅샷 블록에 있는 데이터를 반환하고, PutSnapshotBlock 작업은 스냅샷의 블록에 데이터를 추가합니다. 전송되는 블록 데이터는 Signature 버전 4 서명 프로세스의 일부로 서명되지 않습니다. 따라서 다음과 같이 체크섬을 사용하여 데이터의 무결성을 검증됩니다.

- GetSnapshotBlock 작업을 사용하는 경우 응답은 x-amz-Checksum 헤더를 사용하여 블록 데이터에 대한 Base64로 인코딩된 SHA256 체크섬을 제공하고 x-amz-Checksum-Algorithm 헤더를 사용하여 체크섬 알고리즘을 제공합니다. 반환된 체크섬을 사용하여 데이터의 무결성을 검증하십시오. 생성한 체크섬이 Amazon EBS에서 제공된 체크섬과 일치하지 않는 경우 데이터가 유효하지 않은 것으로 간주하고 요청을 다시 시도해야 합니다.

- PutSnapshotBlock 작업을 사용하는 경우 요청에서 x-amz-Checksum 헤더를 사용하여 블록 데이터에 대한 Base64로 인코딩된 SHA256 체크섬을 제공하고 x-amz-Checksum-Algorithm 헤더를 사용하여 체크섬 알고리즘을 제공해야 합니다. 제공하는 체크섬은 Amazon EBS에서 생성된 체크섬과 대조되어 데이터의 무결성이 검증됩니다. 체크섬이 일치하지 않으면 요청이 실패합니다.
- CompleteSnapshot 작업을 사용하는 경우 요청에서 스냅샷에 추가된 전체 데이터 세트에 대해 Base64로 인코딩된 SHA256 체크섬을 선택적으로 제공할 수 있습니다. x-amz-Checksum 헤더를 사용하여 체크섬을 제공하고, x-amz-Checksum-Algorithm 헤더를 사용하여 체크섬 알고리즘을 제공하고, x-amz-Checksum-Aggregation-Method 헤더를 사용하여 체크섬 집계 방법을 제공합니다. 선형 집계 방법을 사용하여 집계된 체크섬을 생성하려면 작성된 각 블록의 체크섬을 블록 인덱스의 오름차순으로 정렬하고 이를 연결하여 단일 문자열을 형성한 다음 SHA256 알고리즘을 사용하여 전체 문자열에 대한 체크섬을 생성합니다.

이러한 작업의 체크섬은 Signature 버전 4 서명 프로세스의 일부입니다.

API 또는 AWS SDK를 사용하여 EBS 디렉트 API 작업

EBS 디렉트 API 참조에는 각 서비스의 작업과 데이터 형식에 맞는 설명과 구문이 나와 있습니다. AWS SDK 중 하나를 통해 사용 중인 프로그래밍 언어나 플랫폼에 맞는 API에 액세스할 수도 있습니다. 자세한 정보는 [AWS SDK를 참조하십시오.](#)

EBS 디렉트 API에는 AWS Signature 버전 4 서명이 필요합니다. 자세한 내용은 [Signature 버전 4 서명 사용 \(p. 1000\)](#) 단원을 참조하십시오.

API를 사용하여 스냅샷 읽기

스냅샷 블록 나열

다음 [ListChangedBlocks](#) 예제 요청은 스냅샷 snap-0acEXAMPLEcf41648에 있는 블록의 블록 인덱스 및 블록 토큰을 반환합니다. startingBlockIndex 파라미터는 결과를 1000보다 큰 블록 인덱스로 제한하고, maxResults 파라미터는 결과를 첫 번째 100 블록으로 제한합니다.

```
GET /snapshots/snap-0acEXAMPLEcf41648/blocks?maxResults=100&startingBlockIndex=0 HTTP/1.1
Host: ebs.us-east-2.amazonaws.com
Accept-Encoding: identity
User-Agent: <User agent parameter>
X-Amz-Date: 20200617T231953Z
Authorization: <Authentication parameter>
```

이전 요청에 대한 다음 예제 응답은 스냅샷의 블록 인덱스 및 블록 토큰을 나열합니다. GetSnapshotBlock 작업을 사용하고 데이터를 가져올 블록의 블록 인덱스 및 블록 토큰을 지정합니다. 블록 토큰은 기재된 만료 시간까지 유효합니다.

```
HTTP/1.1 200 OK
x-amzn-RequestId: d6e5017c-70a8-4539-8830-57f5557f3f27
Content-Type: application/json
Content-Length: 2472
Date: Wed, 17 Jun 2020 23:19:56 GMT
Connection: keep-alive

{
    "BlockSize": 524288,
    "Blocks": [
        {
            "BlockIndex": 0,
            "BlockToken": "AAUBAcuWqOCnDNuKle1ls7IIIX6jp6FYcC/q8oT93913HhvLvA+3JRrSybp/0"
        },
        {
            "BlockIndex": 1536,
            "BlockToken": "AAUBAWudwfmofcrQhGV1LwuRKm2b8ZXPiyrgoykTRC6IU1NbxEWDY1pPjvnV"
        }
    ]
}
```

```
{  
    "BlockIndex": 3072,  
    "BlockToken": "AAUBAV7p6pC5fKAC7TokoNCtAnZhqq27u6YEXZ3MwRevBkDjmMx6iuA6tsBt"  
},  
{  
    "BlockIndex": 3073,  
    "BlockToken": "AAUBAbqt9zpqBUEvtO2HINAfFaWToOwlPjbIsQOlx6JUN/0+iMQL0NtNbN4"  
},  
...  
],  
"ExpiryTime": 1.59298379649E9,  
"VolumeSize": 3  
}
```

두 스냅샷에서 차이가 있는 블록 나열

다음 [ListChangedBlocks](#) 예제 요청은 스냅샷 snap-0acEXAMPLEcf41648과 snap-0c9EXAMPLE1b30e2f에서 차이가 있는 블록의 블록 인덱스 및 블록 토큰을 반환합니다. startingBlockIndex 파라미터는 결과를 0보다 큰 블록 인덱스로 제한하고, maxResults 파라미터는 결과를 첫 번째 500 블록으로 제한합니다.

```
GET /snapshots/snap-0c9EXAMPLE1b30e2f/changedblocks?  
firstSnapshotId=snap-0acEXAMPLEcf41648&maxResults=500&startingBlockIndex=0 HTTP/1.1  
Host: ebs.us-east-2.amazonaws.com  
Accept-Encoding: identity  
User-Agent: <User agent parameter>  
X-Amz-Date: 20200617T232546Z  
Authorization: <Authentication parameter>
```

이전 요청에 대한 다음 예제 응답은 블록 인덱스 0, 3072, 6002, 6003이 두 스냅샷에서 차이가 있다는 것을 보여 줍니다. 또한 응답에 기재된 두 번째 블록 토큰이 없으므로 지정된 첫 번째 스냅샷 ID에만 블록 인덱스 6002 및 6003이 있고 두 번째 스냅샷 ID에는 없습니다.

GetSnapshotBlock 작업을 사용하고 데이터를 가져올 블록의 블록 인덱스 및 블록 토큰을 지정합니다. 블록 토큰은 기재된 만료 시간까지 유효합니다.

```
HTTP/1.1 200 OK  
x-amzn-RequestId: fb0f6743-6d81-4be8-afbe-db11a5bb8a1f  
Content-Type: application/json  
Content-Length: 1456  
Date: Wed, 17 Jun 2020 23:25:47 GMT  
Connection: keep-alive  
  
{  
    "BlockSize": 524288,  
    "ChangedBlocks": [  
        {  
            "BlockIndex": 0,  
            "FirstBlockToken": "AAUBAVaWqOCnDNuKle11s7IIX6jp6FYcC/tJuVT1GgP23AuLntwiMdJ  
+OJkL",  
            "SecondBlockToken": "AAUBASxzy0Y0b33JVRLoYm3NOresCxn5R0+HVFzXW3Y/  
RwfFaPX2Edx8QHCh"  
        },  
        {  
            "BlockIndex": 3072,  
            "FirstBlockToken": "AAUBAcHp6pC5fKAC7TokoNCtAnZhqq27u6fxRfZOLOemeXLmHBf2R/  
Yb24MaS",  
            "SecondBlockToken":  
            "AAUBARGCaufCqBRZC8tEkPYGGkSv3vqvOjJ2xKDj3ljdFiytUxBLXYgTmkid"  
        },  
        {  
            "BlockIndex": 6002,  
            "FirstBlockToken":  
            "AAUBAV7p6pC5fKAC7TokoNCtAnZhqq27u6YEXZ3MwRevBkDjmMx6iuA6tsBt"  
        }  
    ]  
}
```

```
"FirstBlockToken": "AAABASqX4/  
NWjvNceoyMULjcRd0DnwbSwNnes1UkoP62CrQXvn47BY5435aw"  
,  
{  
    "BlockIndex": 6003,  
    "FirstBlockToken":  
"AAABASmJ0O5JxAOce25rF4P1sdRtyIDsX12tFEDunnePYUKOf4PBROuICb2A"  
,  
...  
],  
"ExpiryTime": 1.592976647009E9,  
"VolumeSize": 3  
}
```

스냅샷에서 블록 데이터 가져오기

다음 [GetSnapshotBlock](#) 예제 요청은 스냅샷 snap-0c9EXAMPLE1b30e2f에서 블록 토큰이 AAUBARGCaufCqBRZC8tEkPYGGkSv3vqvOjJ2xKDi3ljdFiytUxBLXYgTmkid인 블록 인덱스 3072의 데이터를 반환합니다.

```
GET /snapshots/snap-0c9EXAMPLE1b30e2f/blocks/3072?  
blockToken=AAUBARGCaufCqBRZC8tEkPYGGkSv3vqvOjJ2xKDi3ljdFiytUxBLXYgTmkid HTTP/1.1  
Host: ebs.us-east-2.amazonaws.com  
Accept-Encoding: identity  
User-Agent: <User agent parameter>  
X-Amz-Date: 20200617T232838Z  
Authorization: <Authentication parameter>
```

이전 요청에 대한 다음 예제 응답은 반환된 데이터의 크기, 데이터를 검증하기 위한 체크섬 및 체크섬 생성에 사용되는 알고리즘을 보여줍니다. 이진 데이터는 응답 본문에 포함돼 전송되며 다음 예제에서는 **BlockData**로 표시되어 있습니다.

```
HTTP/1.1 200 OK  
x-amzn-RequestId: 2d0db2fb-bd88-474d-a137-81c4e57d7b9f  
x-amz-Data-Length: 524288  
x-amz-C checksum: Vc0yY2j3qg8bUL9I6GQui2orTudrQRBDMIhc7bdEsw=  
x-amz-C checksum-Algorithm: SHA256  
Content-Type: application/octet-stream  
Content-Length: 524288  
Date: Wed, 17 Jun 2020 23:28:38 GMT  
Connection: keep-alive  
  
BlockData
```

API를 사용하여 증분 스냅샷 쓰기

스냅샷 시작

다음 [StartSnapshot](#) 예제 요청은 스냅샷 snap-123EXAMPLE1234567을 상위 스냅샷으로 사용하여 8GiB 스냅샷을 시작합니다. 새 스냅샷은 상위 스냅샷의 증분 스냅샷이 됩니다. 지정된 60분의 제한 시간 내에 스냅샷에 대한 추가 또는 완료 요청이 없으면 스냅샷이 오류 상태로 전환됩니다. 550e8400-e29b-41d4-a716-446655440000 클라이언트 토큰은 요청에 대한 역동성을 보장합니다. 이 클라이언트 토큰이 생략되면 AWS SDK에서 자동으로 생성됩니다. 역동성에 대한 자세한 내용은 [StartSnapshot API에 대한 역동성 \(p. 1012\)](#) 단원을 참조하십시오.

```
POST /snapshots HTTP/1.1  
Host: ebs.us-east-2.amazonaws.com  
Accept-Encoding: identity  
User-Agent: <User agent parameter>  
X-Amz-Date: 20200618T040724Z
```

```
Authorization: <Authentication parameter>

{
    "VolumeSize": 8,
    "ParentSnapshot": "snap-123EXAMPLE1234567",
    "ClientToken": "550e8400-e29b-41d4-a716-446655440000",
    "Timeout": 60
}
```

이전 요청에 대한 다음 예제 응답은 스냅샷 ID, AWS 계정 ID, 상태, 볼륨 크기(GiB) 및 스냅샷의 블록 크기를 보여줍니다. 스냅샷이 대기 중 상태로 시작됩니다. 후속 PutSnapshotBlocks 요청에서 스냅샷 ID를 지정하여 스냅샷에 데이터를 씁니다.

```
HTTP/1.1 201 Created
x-amzn-RequestId: 929e6eb9-7183-405a-9502-5b7da37c1b18
Content-Type: application/json
Content-Length: 181
Date: Thu, 18 Jun 2020 04:07:29 GMT
Connection: keep-alive

{
    "BlockSize": 524288,
    "Description": null,
    "OwnerId": "138695307491",
    "Progress": null,
    "SnapshotId": "snap-052EXAMPLEc85d8dd",
    "StartTime": null,
    "Status": "pending",
    "Tags": null,
    "VolumeSize": 8
}
```

스냅샷에 데이터 추가

다음 PutSnapshot 예제 요청은 스냅샷 snap-052EXAMPLEc85d8dd의 블록 인덱스 1000에 524288바이트의 데이터를 씁니다. Base64로 인코딩된 QOD3gmEQQXATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM= 체크섬은 SHA256 알고리즘을 사용하여 생성되었습니다. 데이터는 요청 본문에 포함돼 전송되며 다음 예제에서는 BlockData로 표시되어 있습니다.

```
PUT /snapshots/snap-052EXAMPLEc85d8dd(blocks/1000 HTTP/1.1
Host: ebs.us-east-2.amazonaws.com
Accept-Encoding: identity
x-amz-Data-Length: 524288
x-amz-C checksum: QOD3gmEQQXATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM=
x-amz-C checksum-Algorithm: SHA256
User-Agent: <User agent parameter>
X-Amz-Date: 20200618T042215Z
X-Amz-Content-SHA256: UNSIGNED-PAYLOAD
Authorization: <Authentication parameter>

BlockData
```

이전 요청에 대한 다음 예제 응답은 서비스에서 수신한 데이터의 데이터 길이, 체크섬 및 체크섬 알고리즘을 확인합니다.

```
HTTP/1.1 201 Created
x-amzn-RequestId: 643ac797-7e0c-4ad0-8417-97b77b43c57b
x-amz-C checksum: QOD3gmEQQXATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM=
x-amz-C checksum-Algorithm: SHA256
Content-Type: application/json
Content-Length: 2
```

```
Date: Thu, 18 Jun 2020 04:22:12 GMT
Connection: keep-alive

{}
```

스냅샷 완료

다음 [CompleteSnapshot](#) 예제 요청은 스냅샷 snap-052EXAMPLEc85d8dd를 완료합니다. 이 명령은 5 블록을 스냅샷에 쓰도록 지정합니다. 6D3nmwi5f2F0wlh7xX8QprJBFzDX8aacdOcA3KCM3c= 체크섬은 스냅샷에 쓴 전체 데이터 세트의 체크섬을 나타냅니다.

```
POST /snapshots/completion/snap-052EXAMPLEc85d8dd HTTP/1.1
Host: ebs.us-east-2.amazonaws.com
Accept-Encoding: identity
x-amz-ChangedBlocksCount: 5
x-amz-C checksum: 6D3nmwi5f2F0wlh7xX8QprJBFzDX8aacdOcA3KCM3c=
x-amz-C checksum-Algorithm: SHA256
x-amz-C checksum-Aggregation-Method: LINEAR
User-Agent: <User agent parameter>
X-Amz-Date: 20200618T043158Z
Authorization: <Authentication parameter>
```

다음은 이전 요청에 대한 응답의 예입니다.

```
HTTP/1.1 202 Accepted
x-amzn-RequestId: 06cba5b5-b731-49de-af40-80333ac3a117
Content-Type: application/json
Content-Length: 20
Date: Thu, 18 Jun 2020 04:31:50 GMT
Connection: keep-alive

{"Status": "pending"}
```

명령줄을 사용한 EBS 디렉트 API 작업

다음 예제에서는 AWS Command Line Interface(AWS CLI)를 통해 EBS 디렉트 API를 이용하는 방법을 보여줍니다. AWS CLI 설치 및 구성에 대한 자세한 내용은 [AWS CLI 설치 및 빠르게 AWS CLI 구성](#)을 참조하십시오.

AWS CLI를 사용하여 스냅샷 읽기

스냅샷 블록 나열

다음 [list-snapshot-blocks](#) 예제 명령은 스냅샷 snap-0987654321에 있는 블록의 블록 인덱스 및 블록 토큰을 반환합니다. --starting-block-index 파라미터는 결과를 1000보다 큰 블록 인덱스로 제한하고, --max-results 파라미터는 결과를 첫 번째 100 블록으로 제한합니다.

```
aws ebs list-snapshot-blocks --snapshot-id snap-0987654321 --starting-block-index 1000 --
max-results 100
```

이전 명령에 대한 다음 예제 응답은 스냅샷의 블록 인덱스 및 블록 토큰을 나열합니다. [get-snapshot-block](#) 명령을 사용하고 데이터를 가져올 블록의 블록 인덱스 및 블록 토큰을 지정합니다. 블록 토큰은 기재된 만료 시간까지 유효합니다.

```
{
  "Blocks": [
    {
      "BlockIndex": 1001,
```

```
        "BlockToken": "AAABAV3/PNhXOynVdMYHUpPsetaSvjLB1dtIGfbJv5OJ0sX855EzGTWos4a4"
    },
    {
        "BlockIndex": 1002,
        "BlockToken": "AAABATGQIgwr0WwIuqIMjCA/Sy7e/YoQFZsHejzGNvjKauzNgzeI13YHBfQB"
    },
    {
        "BlockIndex": 1007,
        "BlockToken": "AAABAZ9CTuQtUvp/dXqRWw4d07eOgTZ3jvn6hiW30W9duM8MiMw6yQayzF2c"
    },
    {
        "BlockIndex": 1012,
        "BlockToken": "AAABAQdzxhw0rVV6PNmsfo/YRIxo9JPR85XxPf1BLjg0Hec6pygYr6laE1p0"
    },
    {
        "BlockIndex": 1030,
        "BlockToken": "AAABAAaYvPax6mv+iGWLdTUjQtFWouQ7Dqz6nSD9L+CbXnvpkswA6iIDID523d"
    },
    {
        "BlockIndex": 1031,
        "BlockToken": "AAABATgWZC0XcFwUKvTJbUXMiSPg59KVxJGL+BWBC1kw6spzCxJVqDVaTskJ"
    },
    ...
],
"ExpiryTime": 1576287332.806,
"VolumeSize": 32212254720,
"BlockSize": 524288
}
```

두 스냅샷에서 차이가 있는 블록 나열

다음 `list-changed-blocks` 예제 요청은 스냅샷 snap-1234567890과 snap-0987654321에서 차이가 있는 블록의 블록 인덱스 및 블록 토큰을 반환합니다. --starting-block-index 파라미터는 결과를 0보다 큰 블록 인덱스로 제한하고, --max-results 파라미터는 결과를 첫 번째 500 블록으로 제한합니다.

```
aws ebs list-changed-blocks --first-snapshot-id snap-1234567890 --second-snapshot-
id snap-0987654321 --starting-block-index 0 --max-results 500
```

이전 명령에 대한 다음 예제 응답은 블록 인덱스 0, 6000, 6001, 6002, 6003이 두 스냅샷에서 차이가 있다는 것을 보여 줍니다. 또한 응답에 기재된 두 번째 블록 토큰이 없으므로 지정된 첫 번째 스냅샷 ID에만 블록 인덱스 6001, 6002, 6003이 있고 두 번째 스냅샷 ID에는 없습니다.

`get-snapshot-block` 명령을 사용하고 데이터를 가져올 블록의 블록 인덱스 및 블록 토큰을 지정합니다. 블록 토큰은 기재된 만료 시간까지 유효합니다.

```
{
    "ChangedBlocks": [
        {
            "BlockIndex": 0,
            "FirstBlockToken": "AAABAVahm9SO60Dyi0ORySzn2ZjGjW/
KN3uygGlS0QOYWesbzBbDnX2dGpmC",
            "SecondBlockToken":
"AAABAf8o0o6UFi1rDbSZGIRaCEdDyBu9TlvtCQxxoKV8qrUPQP7vcM6iWGsr"
        },
        {
            "BlockIndex": 6000,
            "FirstBlockToken": "AAABAbYSiZvJ0/
R9tz8suI8dSzecLjN4kkazK8inFXVintPkdaVFLfCMQsKe",
            "SecondBlockToken":
"AAABAZnqTdzFmKRpsaMAsDxviVqEI/3jJzI2crq2eFDCgHmyNf777elD9oVR"
        },
        {
            "BlockIndex": 6001,
            "FirstBlockToken": "AAABAbYSiZvJ0/
R9tz8suI8dSzecLjN4kkazK8inFXVintPkdaVFLfCMQsKe",
            "SecondBlockToken":
"AAABAZnqTdzFmKRpsaMAsDxviVqEI/3jJzI2crq2eFDCgHmyNf777elD9oVR"
        },
        {
            "BlockIndex": 6002,
            "FirstBlockToken": "AAABAbYSiZvJ0/
R9tz8suI8dSzecLjN4kkazK8inFXVintPkdaVFLfCMQsKe",
            "SecondBlockToken":
"AAABAZnqTdzFmKRpsaMAsDxviVqEI/3jJzI2crq2eFDCgHmyNf777elD9oVR"
        },
        {
            "BlockIndex": 6003,
            "FirstBlockToken": "AAABAbYSiZvJ0/
R9tz8suI8dSzecLjN4kkazK8inFXVintPkdaVFLfCMQsKe",
            "SecondBlockToken":
"AAABAZnqTdzFmKRpsaMAsDxviVqEI/3jJzI2crq2eFDCgHmyNf777elD9oVR"
        }
    ]
}
```

```
"BlockIndex": 6001,  
"FirstBlockToken": "AAABASBpSJ2UAD3PLxJnCt6zun4/  
T4sU25Bnb8jB5Q6FRXHFqAIAqE04hJoR"  
,  
{  
    "BlockIndex": 6002,  
    "FirstBlockToken": "AAABASqX4/  
NWjvNceoyMULjcRd0DnwbSwNnes1UkoP62CrQXvn47BY5435aw"  
,  
{  
    "BlockIndex": 6003,  
    "FirstBlockToken":  
"AAABASmJ005JxAOce25rF4P1sdRtyIDsX12tFEDunnePYUKOf4PBROuiCb2A"  
,  
...  
],  
"ExpiryTime": 1576308931.973,  
"VolumeSize": 32212254720,  
"BlockSize": 524288,  
"NextToken": "AAADARqElNng/sV98CYk/bJDCXeLJmLJHnNSkHvLzVaO0zsPH/QM3Bi3zF//O6Mdi/  
BbJarBnp8h"  
}
```

스냅샷에서 블록 데이터 가져오기

다음 `get-snapshot-block` 예제 요청은 스냅샷 `snap-1234567890`에서 블록 토큰이 `AAABASBpSJ2UAD3PLxJnCt6zun4/T4sU25Bnb8jB5Q6FRXHFqAIAqE04hJoR`인 블록 인덱스 6001의 데이터를 반환합니다. Windows 컴퓨터에서 C:\Temp 디렉터리의 data 파일로 이진 데이터가 출력됩니다. Linux 또는 Unix 컴퓨터에서 명령을 실행할 때는 출력 경로를 /tmp/data로 바꾸어 /tmp 디렉터리의 data 파일로 데이터를 출력하십시오.

```
aws ebs get-snapshot-block --snapshot-id snap-1234567890 --block-index 6001 --block-  
token AAABASBpSJ2UAD3PLxJnCt6zun4/T4sU25Bnb8jB5Q6FRXHFqAIAqE04hJoR C:/Temp/data
```

이전 명령에 대한 다음 예제 응답은 반환된 데이터의 크기, 데이터를 검증하기 위한 체크섬 및 체크섬 알고리즘을 보여줍니다. 요청 명령에서 지정한 디렉터리와 파일에 이진 데이터가 자동으로 저장됩니다.

```
{  
    "DataLength": "524288",  
    "Checksum": "cf0Y6/Fn0oFa4VyjQPOa/iD0zhTf1PTKzxGv2OKowXc=",  
    "ChecksumAlgorithm": "SHA256"  
}
```

AWS CLI를 사용하여 종분 스냅샷 쓰기

스냅샷 시작

다음 `start-snapshot` 예제 요청은 스냅샷 `snap-123EXAMPLE1234567`을 상위 스냅샷으로 사용하여 8GiB 스냅샷을 시작합니다. 새 스냅샷은 상위 스냅샷의 종분 스냅샷이 됩니다. 지정된 60분의 제한 시간 내에 스냅샷에 대한 추가 또는 완료 요청이 없으면 스냅샷이 오류 상태로 전환됩니다. 550e8400-e29b-41d4-a716-446655440000 클라이언트 토큰은 요청에 대한 멱등성을 보장합니다. 이 클라이언트 토큰이 생략되면 AWS SDK에서 자동으로 생성됩니다. 멱등성에 대한 자세한 내용은 [StartSnapshot API에 대한 멱등성 \(p. 1012\)](#) 단원을 참조하십시오.

```
aws ebs start-snapshot --volume-size 8 --parent-snapshot snap-123EXAMPLE1234567 --  
timeout 60 --client-token 550e8400-e29b-41d4-a716-446655440000
```

이전 명령에 대한 다음 예제 응답은 스냅샷 ID, AWS 계정 ID, 상태, 볼륨 크기(GiB) 및 스냅샷의 블록 크기를 보여줍니다. 스냅샷이 pending 상태로 시작됩니다. 후속 `put-snapshot-block` 명령에서 스냅샷 ID

를 지정하여 스냅샷에 데이터를 쓴 다음 `complete-snapshot` 명령을 사용하여 스냅샷을 완료하고 상태를 `completed`로 변경합니다.

```
{  
    "SnapshotId": "snap-0aaEXAMPLEe306d62",  
    "OwnerId": "111122223333",  
    "Status": "pending",  
    "VolumeSize": 8,  
    "BlockSize": 524288  
}
```

스냅샷에 데이터 추가

다음 `put-snapshot-block` 예제 명령은 스냅샷 `snap-0aaEXAMPLEe306d62`의 블록 인덱스 1000에 524288바이트의 데이터를 씁니다. Base64로 인코딩된 `QOD3gmEQOXATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM=` 체크섬은 SHA256 알고리즘을 사용하여 생성되었습니다. 전송되는 데이터는 `/tmp/data` 파일에 있습니다.

```
aws ebs put-snapshot-block --snapshot-id snap-0aaEXAMPLEe306d62  
    --block-index 1 --data-length 524288 --block-data /tmp/data --  
    checksum QOD3gmEQOXATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM= --checksum-algorithm SHA256
```

이전 명령에 대한 다음 예제 응답은 서비스에서 수신한 데이터 길이, 체크섬 및 체크섬 알고리즘을 확인합니다.

```
{  
    "DataLength": "524288",  
    "Checksum": "QOD3gmEQOXATfJx2Aa34W4FU2nZGyXfqtsUuktOw8DM=",  
    "ChecksumAlgorithm": "SHA256"  
}
```

스냅샷 완료

다음 `complete-snapshot` 예제 명령은 스냅샷 `snap-0aaEXAMPLEe306d62`를 완료합니다. 이 명령은 5 블록을 스냅샷에 쓰도록 지정합니다. `6D3nmwi5f2F0wlh7xX8OprrJBFzDX8aacdOcA3KCM3c=` 체크섬은 스냅샷에 쓴 전체 데이터 세트의 체크섬을 나타냅니다. 체크섬에 대한 자세한 내용은 이 설명서 앞부분에 있는 [체크섬 사용 \(p. 1000\)](#) 단원을 참조하십시오.

```
aws ebs complete-snapshot --snapshot-id snap-0aaEXAMPLEe306d62 --changed-blocks-count 5  
    --checksum 6D3nmwi5f2F0wlh7xX8OprrJBFzDX8aacdOcA3KCM3c= --checksum-algorithm SHA256 --  
    checksum-aggregation-method LINEAR
```

다음은 이전 명령에 대한 응답의 예입니다.

```
{  
    "Status": "pending"  
}
```

성능 최적화

API 요청을 동시에 실행할 수 있습니다. `PutSnapshotBlock` 지연 시간이 100ms라고 가정하면 한 스레드는 1초에 10개의 요청을 처리할 수 있습니다. 또한 클라이언트 애플리케이션이 여러 스레드 및 연결(예: 100개의 연결)을 생성한다고 가정하면 초당 총 1000($10 * 100$)개의 요청을 할 수 있습니다. 이는 초당 약 500MB의 처리량에 해당합니다.

다음은 애플리케이션에서 고려할 몇 가지 사항입니다.

- 각 스레드가 별도의 연결을 사용합니까? 애플리케이션에서 연결이 제한되어 있으면 여러 스레드가 연결을 사용할 수 있을 때까지 대기하여 처리량이 낮아집니다.

- 두 추가 요청 사이에 애플리케이션에 대기 시간이 있습니까? 이 경우 스레드의 유효 처리량이 줄어듭니다.
- 인스턴스의 대역폭 제한 - 인스턴스의 대역폭을 다른 애플리케이션에서 공유하는 경우 PutSnapshotBlock 요청에 사용할 수 있는 처리량이 제한될 수 있습니다.

병목 현상을 방지하려면 계정에서 실행될 수 있는 다른 워크로드를 고려해야 합니다. 또한 조절, 시간 초과 및 서비스 사용 불가를 처리하려면 EBS 다이렉트 API 워크플로우에 재시도 메커니즘을 빌드해야 합니다.

EBS 다이렉트 API 서비스 할당량을 검토하여 초당 실행할 수 있는 최대 API 요청을 확인합니다. 자세한 내용은 AWS 일반 참조의 [Amazon Elastic Block Store 엔드포인트 및 할당량](#)을 참조하십시오.

FAQ

대기 중 상태일 때 EBS 다이렉트 API를 사용하여 스냅샷에 액세스할 수 있습니까?

아니요. 완료됨 상태인 경우에만 스냅샷에 액세스할 수 있습니다.

EBS 다이렉트 API에서 번호순으로 블록 인덱스를 반환합니까?

예. 반환되는 블록 인덱스는 고유하며 번호순입니다.

MaxResults 파라미터 값이 100 미만인 요청을 제출할 수 있습니까?

아니요. 사용할 수 있는 최소 MaxResult 파라미터 값은 100입니다. MaxResult 파라미터 값이 100 미만인 요청을 제출할 경우 스냅샷에 있는 블록이 100개를 초과하면 API가 최소한 100개의 결과를 반환합니다.

API 요청을 동시에 실행할 수 있습니까?

API 요청을 동시에 실행할 수 있습니다. 병목 현상을 방지하려면 계정에서 실행될 수 있는 다른 워크로드를 고려해야 합니다. 또한 조절, 시간 초과 및 서비스 사용 불가를 처리하려면 EBS 다이렉트 API 워크플로우에 재시도 메커니즘을 빌드해야 합니다. 자세한 내용은 [성능 최적화 \(p. 1008\)](#) 단원을 참조하십시오.

EBS 다이렉트 API 서비스 할당량을 검토하여 초당 실행할 수 있는 API 요청을 확인합니다. 자세한 내용은 AWS 일반 참조의 [Amazon Elastic Block Store 엔드포인트 및 할당량](#)을 참조하십시오.

ListChangedBlocks 작업을 실행할 때 스냅샷에 블록이 있어도 빈 응답을 받을 수 있습니까?

예. 스냅샷에 변경된 블록이 거의 없으면 응답이 비어 있을 수 있지만 API가 다음 페이지 토큰 값을 반환합니다. 다음 페이지 토큰 값을 사용하여 결과의 다음 페이지로 계속 진행하십시오. API가 다음 페이지 토큰 값 null을 반환하면 마지막 결과 페이지에 도달했음을 확인할 수 있습니다.

NextToken 파라미터가 StartingBlockIndex 파라미터와 함께 지정되면 둘 중 어느 것이 사용됩니까?

NextToken이 사용되며 StartingBlockIndex는 무시됩니다.

블록 토큰과 다음 토큰은 얼마나 유효합니까?

블록 토큰은 7일간 유효하고 다음 토큰은 60분간 유효합니다.

암호화된 스냅샷이 지원됩니까?

예. EBS 다이렉트 API를 사용하여 암호화된 스냅샷에 액세스할 수 있습니다.

암호화된 스냅샷에 액세스하려면 스냅샷 암호화에 사용된 키와 AWS KMS 암호 해독 작업에 대한 액세스 권한이 사용자에게 있어야 합니다. 사용자에게 할당할 AWS KMS 정책에 대해서는 이 설명서 앞부분의 [IAM 사용자의 권한 \(p. 996\)](#) 단원을 참조하십시오.

퍼블릭 스냅샷이 지원됩니까?

퍼블릭 스냅샷이 지원되지 않습니다.

독록 스냅샷 블록은 스냅샷의 모든 블록 인덱스와 블록 토큰을 반환합니까? 아니면 데이터가 쓰여진 블록 인덱스와 블록 토큰만 반환합니까?

데이터가 쓰여진 블록 인덱스와 토큰만 반환합니다.

보안 분석 및 운영 문제 해결 목적으로 내 계정에서 이루어진 모든 EBS 디렉트 API API 호출 기록을 얻을 수 있습니까?

예. 계정에서 이루어진 EBS 디렉트 API API 호출 기록을 수신하려면 AWS 관리 콘솔에서 AWS CloudTrail을 활성화합니다. 자세한 내용은 [AWS CloudTrail을 사용하여 EBS 디렉트 API API 호출 로깅 \(p. 1010\)](#) 단원을 참조하십시오.

AWS CloudTrail을 사용하여 EBS 디렉트 API API 호출 로깅

EBS 디렉트 API 서비스는 AWS CloudTrail과 통합됩니다. CloudTrail은 EBS 디렉트 API에서 사용자, 역할 또는 AWS 서비스가 수행한 작업에 대한 기록을 제공하는 서비스입니다. CloudTrail은 EBS 디렉트 API에 대한 [StartSnapshot](#) 및 [CompleteSnapshot](#) API 호출을 이벤트로 캡처합니다. 추적을 생성하면 EBS 디렉트 API에 대한 이벤트를 비롯하여 CloudTrail 이벤트를 Amazon Simple Storage Service(Amazon S3) 버킷으로 지속적으로 전송할 수 있습니다. 추적을 구성하지 않은 경우 이벤트 기록에서 CloudTrail 콘솔의 최신 이벤트를 볼 수도 있습니다. CloudTrail에서 수집하는 정보를 사용하여 EBS 디렉트 API에 수행된 요청, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다.

CloudTrail에 대한 자세한 내용은 [AWS CloudTrail User Guide](#) 단원을 참조하십시오.

CloudTrail의 EBS 디렉트 API 정보

CloudTrail은 계정 생성 시 AWS 계정에서 활성화됩니다. 지원되는 이벤트 활동이 EBS 디렉트 API에서 발생하면 해당 활동이 그 밖의 AWS 서비스 이벤트와 함께 이벤트 기록에 CloudTrail 이벤트로 기록됩니다. AWS 계정에서 최신 이벤트를 확인, 검색 및 다운로드할 수 있습니다. 자세한 내용은 [CloudTrail 이벤트 기록에서 이벤트 보기](#)를 참조하십시오.

EBS 디렉트 API에 대한 이벤트를 포함하여 AWS 계정에 이벤트를 지속적으로 기록하려면 추적을 생성합니다. 추적은 CloudTrail이 S3 버킷으로 로그 파일을 전송할 수 있도록 합니다. 콘솔에서 추적을 생성하면 기본적으로 모든 AWS 리전에 추적이 적용됩니다. 추적은 AWS 파티션에 있는 모든 리전의 이벤트를 로깅하고 지정된 S3 버킷에 로그 파일을 제공합니다. 또는 CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음 단원을 참조하세요.

- 추적 생성 개요
- CloudTrail 지원 서비스 및 통합
- CloudTrail에 대한 Amazon SNS 알림 구성
- 여러 리전에서 CloudTrail 로그 파일 받기 및 [여러 계정에서 CloudTrail 로그 파일 받기](#)

지원되는 API 작업

다음 API 작업은 CloudTrail 로그 파일의 이벤트로 로깅을 지원합니다.

- [StartSnapshot](#)
- [CompleteSnapshot](#)

자격 증명 정보

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 자격 증명 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트로 했는지 아니면 AWS Identity and Access Management(IAM) 사용자 자격 증명으로 했는지 여부
- 역할 또는 연합된 사용자에 대한 임시 보안 자격 증명을 사용하여 요청이 생성되었는지 여부.
- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 [CloudTrail userIdentityElement](#)를 참조하세요.

EBS 디렉트 API 로그 파일 항목 이해

추적은 지정한 S3 버킷에 이벤트를 로그 파일로 전달할 수 있도록 하는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함됩니다. 이벤트는 어떤 소스로부터의 단일 요청을 나타내며 요청된 작업, 작업 날짜와 시간, 요청 파라미터 등에 대한 정보가 포함되어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 호출의 주문 스택 추적이 아니므로 특정 순서로 표시되지 않습니다.

다음 예제에는 StartSnapshot 및 CompleteSnapshot 작업을 보여주는 CloudTrail 로그 항목이 나와 있습니다.

StartSnapshot 예제:

```
{  
    "eventVersion": "1.05",  
    "userIdentity": {  
        "type": "IAMUser",  
        "principalId": "123456789012",  
        "arn": "arn:aws:iam::123456789012:root",  
        "accountId": "123456789012",  
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",  
        "userName": "user"  
    },  
    "eventTime": "2020-07-03T23:27:26Z",  
    "eventSource": "ebs.amazonaws.com",  
    "eventName": "StartSnapshot",  
    "awsRegion": "eu-west-1",  
    "sourceIPAddress": "192.0.2.0",  
    "userAgent": "PostmanRuntime/7.25.0",  
    "requestParameters": {  
        "volumeSize": 8,  
        "clientToken": "token",  
        "encrypted": true  
    },  
    "responseElements": {  
        "snapshotId": "snap-123456789012",  
        "ownerId": "123456789012",  
        "status": "pending",  
        "startTime": "Jul 3, 2020 11:27:26 PM",  
        "volumeSize": 8,  
        "blockSize": 524288,  
        "kmsKeyArn": "HIDDEN_DUE_TO_SECURITY_REASONS"  
    },  
    "requestID": "be112233-1ba5-4ae0-8e2b-1c302EXAMPLE",  
    "eventId": "6e12345-2a4e-417c-aa78-7594fEXAMPLE",  
    "eventType": "AwsApiCall",  
    "recipientAccountId": "123456789012"  
}
```

CompleteSnapshot 예제:

```
{  
    "eventVersion": "1.05",  
    "userIdentity": {  
        "type": "IAMUser",  
        "principalId": "123456789012",  
        "arn": "arn:aws:iam::123456789012:root",  
        "accountId": "123456789012",  
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",  
        "userName": "user"  
    },  
    "eventTime": "2020-07-03T23:28:24Z",  
    "eventSource": "ebs.amazonaws.com",  
    "eventName": "CompleteSnapshot",  
    "responseElements": {  
        "snapshotId": "snap-123456789012",  
        "ownerId": "123456789012",  
        "status": "completed",  
        "startTime": "Jul 3, 2020 11:27:26 PM",  
        "volumeSize": 8,  
        "blockSize": 524288,  
        "kmsKeyArn": "HIDDEN_DUE_TO_SECURITY_REASONS"  
    }  
}
```

```
"awsRegion": "eu-west-1",
"sourceIPAddress": "192.0.2.0",
"userAgent": "PostmanRuntime/7.25.0",
"requestParameters": {
    "snapshotId": "snap-123456789012",
    "changedBlocksCount": 5
},
"responseElements": {
    "status": "completed"
},
"requestID": "be112233-1ba5-4ae0-8e2b-1c302EXAMPLE",
"eventID": "6e12345-2a4e-417c-aa78-7594fEXAMPLE",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}
```

StartSnapshot API에 대한 멱등성

멱등성은 API 요청이 한 번만 완료되도록 합니다. 멱등성 요청 시, 원래 요청이 성공적으로 완료되면 후속 재시도는 원래 성공한 요청의 결과를 반환하며 추가 영향이 없습니다.

[StartSnapshot](#) API는 클라이언트 토큰을 사용하여 멱등성을 지원합니다. 클라이언트 토큰은 API 요청을 할 때 지정하는 고유 문자열입니다. 요청이 성공적으로 완료된 후 동일한 클라이언트 토큰과 동일한 요청 파라미터를 사용하여 API 요청을 다시 시도하면 원래 요청의 결과가 반환됩니다. 동일한 클라이언트 토큰으로 요청을 다시 시도하지만 요청 파라미터를 하나 이상 변경하면 `ConflictException` 오류가 반환됩니다.

고유 클라이언트 토큰을 지정하지 않으면 AWS SDK가 멱등성을 위해 요청에 대한 클라이언트 토큰을 자동으로 생성합니다.

클라이언트 토큰은 최대 64자의 ASCII 문자를 포함하는 모든 문자열이 될 수 있습니다. 다른 요청에 동일한 클라이언트 토큰을 재사용해서는 안 됩니다.

API를 사용하여 고유 클라이언트 토큰으로 멱등성 StartSnapshot 요청을 하려면

`ClientToken` 요청 파라미터를 지정합니다.

```
POST /snapshots HTTP/1.1
Host: ebs.us-east-2.amazonaws.com
Accept-Encoding: identity
User-Agent: <User agent parameter>
X-Amz-Date: 20200618T040724Z
Authorization: <Authentication parameter>

{
    "VolumeSize": 8,
    "ParentSnapshot": snap-123EXAMPLE1234567,
    "ClientToken": "550e8400-e29b-41d4-a716-446655440000",
    "Timeout": 60
}
```

AWS CLI를 사용하여 고유 클라이언트 토큰으로 멱등성 StartSnapshot 요청을 하려면

`client-token` 요청 파라미터를 지정합니다.

```
C:\> aws ebs start-snapshot --region us-east-2 --volume-size 8 --parent-snapshot
snap-123EXAMPLE1234567 --timeout 60 --client-token 550e8400-e29b-41d4-a716-446655440000
```

Amazon EBS 스냅샷 수명 주기 자동화

Amazon EBS 볼륨을 백업하기 위해 만든 스냅샷의 생성, 보관, 삭제를 Amazon 데이터 수명 주기 관리자로 자동화할 수 있습니다. 스냅샷 관리를 자동화하여 다음과 같은 이점을 누려 보십시오.

- 정기적인 백업 일정을 실행하여 중요한 데이터를 보호합니다.
- 감사 기관이나 내부 규정 준수 부서에서 요구하는 백업을 보관합니다.
- 오래된 백업을 삭제하여 스토리지 비용을 절감합니다.

Amazon CloudWatch Events 및 AWS CloudTrail의 모니터링 기능을 Amazon 데이터 수명 주기 관리자와 조합하면 추가 비용 없이 EBS 볼륨의 완벽한 백업 솔루션을 얻을 수 있습니다.

목차

- [Amazon 데이터 수명 주기 관리자 작동 방식 \(p. 1013\)](#)
- [Amazon 데이터 수명 주기 관리자에 대한 고려 사항 \(p. 1014\)](#)
- [사전 조건 \(p. 1015\)](#)
- [콘솔을 사용하여 백업 관리 \(p. 1017\)](#)
- [AWS CLI를 사용하여 백업 관리 \(p. 1019\)](#)
- [API를 사용하여 백업 관리 \(p. 1021\)](#)
- [스냅샷 수명 주기 모니터링 \(p. 1022\)](#)

Amazon 데이터 수명 주기 관리자 작동 방식

다음은 Amazon 데이터 수명 주기 관리자의 핵심 요소입니다.

요소

- [스냅샷 \(p. 1013\)](#)
- [대상 리소스 태그 \(p. 1013\)](#)
- [스냅샷 태그 \(p. 1013\)](#)
- [수명 주기 정책 \(p. 1014\)](#)

스냅샷

스냅샷은 EBS 볼륨에서 데이터를 백업하는 기본 방법입니다. 스토리지 비용을 절약하기 위해 이전 스냅샷 이후로 변경된 볼륨 데이터만 연속 스냅샷에 종분식으로 포함시킵니다. 특정 볼륨의 스냅샷 시리즈에서 스냅샷 하나를 삭제하면 해당 스냅샷에 있는 데이터만 제거됩니다. 캡처된 볼륨 기록의 나머지는 보존됩니다.

자세한 내용은 [Amazon EBS 스냅샷 \(p. 966\)](#) 단원을 참조하십시오.

대상 리소스 태그

Amazon 데이터 수명 주기 관리자는 백업할 EBS 볼륨을 리소스 태그로 파악합니다. 태그란 사용자가 AWS 리소스(EBS 볼륨 및 스냅샷 포함)에 할당할 수 있는 사용자 지정 가능한 메타데이터입니다. (아래에 설명된) Amazon 데이터 수명 주기 관리자 정책은 단일 태그를 사용하여 백업용 볼륨을 대상으로 합니다. 여러 정책을 실행하려면 복수의 태그를 볼륨에 할당할 수 있습니다.

'!' 또는 '=' 문자는 태그 키에 사용할 수 없습니다.

자세한 내용은 [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#) 단원을 참조하십시오.

스냅샷 태그

Amazon 데이터 수명 주기 관리자는 정책에 따라 생성된 모든 스냅샷에 다음 태그를 적용하여 다른 방법으로 생성된 스냅샷과 구분합니다.

- `aws:dlm:lifecycle-policy-id`
- `aws:dlm:lifecycle-schedule-name`

스냅샷을 생성할 때 사용자 지정 태그가 적용되도록 지정할 수도 있습니다.

'!' 또는 '=' 문자는 태그 키에 사용할 수 없습니다.

볼륨을 정책과 연관시키기 위해 Amazon 데이터 수명 주기 관리자이 사용하는 대상 태그는 정책에 의해 생성된 스냅샷에 선택적으로 적용될 수 있습니다.

수명 주기 정책

수명 주기 정책은 다음의 핵심 설정으로 이루어집니다.

- 리소스 유형—정책에 따라 관리하는 AWS 리소스의 유형입니다. VOLUME을 사용하여 개별 볼륨의 스냅샷을 생성하거나 INSTANCE를 사용하여 인스턴스의 볼륨에서 다중 볼륨 스냅샷을 생성합니다. 자세한 내용은 [다중 볼륨 스냅샷 \(p. 970\)](#) 단원을 참조하십시오.
- 대상 태그—정책에 따라 관리할 수 있도록 EBS 볼륨 또는 EC2 인스턴스와 연결해야 하는 태그입니다.
- 일정—스냅샷을 생성하기 위한 시작 시간과 간격입니다. 지정된 시작 시간 후 한 시간 이내에 정책에 따라 첫 번째 스냅샷이 생성됩니다. 후속 스냅샷은 예약된 시간으로부터 1시간 이내에 생성됩니다.
- 보존 — 총 스냅샷 수 또는 각 스냅샷 수명에 따라 스냅샷을 보존할 수 있습니다.

예를 들어 태그 계정이 account=Finance인 모든 EBS 볼륨을 관리하는 정책을 만들고, 24시간마다 0900에 스냅샷을 생성하여 최근 스냅샷 다섯 개를 보관할 수 있습니다. 매일 늦어도 0959에는 스냅샷 생성이 시작됩니다.

Amazon 데이터 수명 주기 관리자에 대한 고려 사항

AWS 계정에는 Amazon 데이터 수명 주기 관리자와 관련하여 다음과 같은 할당량이 있습니다.

- 리전마다 최대 100개의 수명 주기 정책을 생성할 수 있습니다.
- 리소스당 최대 45개의 태그를 추가할 수 있습니다.
- 수명 주기 정책당 하나의 일정을 생성할 수 있습니다.

다음은 수명 주기 정책에서 고려할 사항입니다.

- 정책의 활성화 상태를 활성으로 설정해야 정책이 스냅샷을 생성하기 시작합니다. 생성과 동시에 활성화되도록 정책을 구성할 수 있습니다.
- 지정된 시작 시간 후 한 시간 이내에 정책에 따라 첫 번째 스냅샷이 생성됩니다. 후속 스냅샷은 예약된 시간으로부터 1시간 이내에 생성됩니다.
- 대상 태그를 변경하거나 삭제하여 정책을 수정하면 그러한 태그가 있는 EBS 볼륨에는 더 이상 해당 정책이 적용되지 않습니다.
- 정책의 일정 이름을 수정하면 예전의 일정 이름으로 생성된 스냅샷에는 더 이상 해당 정책이 적용되지 않습니다.
- 새 시간 간격을 사용하도록 시간을 기준으로 하는 보존 일정을 수정하면 새 간격은 새 스냅샷에만 사용됩니다. 새로운 일정은 이 정책에 따라 생성된 기존 스냅샷의 보존 일정에 영향을 미치지 않습니다.
- 스냅샷 수에서 각 스냅샷 수명으로 정책의 보존 일정을 변경할 수 없습니다. 이렇게 변경하려면 새 정책을 생성해야 합니다.
- 보존 일정이 각 스냅샷의 수명을 기준으로 하는 정책을 비활성화하면, 정책이 비활성화된 동안 보존 기간이 만료되는 스냅샷이 무기한 보존됩니다. 이러한 스냅샷은 수동으로 삭제해야 합니다. 정책을 다시 활성화하면 보존 기간이 만료될 경우 Amazon 데이터 수명 주기 관리자가 스냅샷 삭제를 재개합니다.
- 개수 기반 보존이 포함된 정책이 적용되는 리소스를 삭제하면 정책은 이전에 생성한 스냅샷을 더 이상 관리하지 않습니다. 더 이상 필요하지 않은 스냅샷은 수동으로 삭제해야 합니다.
- 수명 기반 보존이 포함된 정책이 적용되는 리소스를 삭제하면 정책은 정의된 일정에 따라 마지막 스냅샷까지 스냅샷을 계속 삭제합니다. 더 이상 필요하지 않은 경우 마지막 스냅샷을 수동으로 삭제해야 합니다.
- EBS 볼륨 또는 EC2 인스턴스를 백업하도록 여러 정책을 생성할 수 있습니다. EBS 볼륨에 12시간마다 스냅샷을 만드는 정책 A의 대상 태그인 태그 A와 24시간마다 스냅샷을 만드는 정책 B의 대상 태그인 태그 B,

이렇게 두 개의 태그가 있다면 Amazon 데이터 수명 주기 관리자는 양쪽 정책의 일정에 따라 스냅샷을 생성합니다.

다음은 수명 주기 정책과 빠른 스냅샷 복원 (p. 1044)에 적용되는 고려 사항입니다.

- 빠른 스냅샷 복원에 대해 활성화된 스냅샷은 수명 주기 정책을 삭제 또는 비활성화하거나, 수명 주기 정책에 대한 빠른 스냅샷 복원을 비활성화하거나, 가용 영역에 대한 빠른 스냅샷 복원을 비활성화하더라도 활성화된 상태로 유지됩니다. 이러한 스냅샷에 대한 빠른 스냅샷 복원을 수동으로 비활성화할 수 있습니다.
 - 빠른 스냅샷 복원을 활성화하고 빠른 스냅샷 복원에 대해 활성화 할 수 있는 최대 스냅샷 수를 초과하는 경우 Amazon 데이터 수명 주기 관리자는 예약된 대로 스냅샷을 생성하지만 빠른 스냅샷 복원에 대해 스냅샷을 활성화하지 않습니다. 빠른 스냅샷 복원에 대해 활성화된 스냅샷을 삭제한 후 Amazon 데이터 수명 주기 관리자가 생성하는 다음 스냅샷은 빠른 스냅샷 복원에 대해 활성화됩니다.
 - 스냅샷에 대해 빠른 스냅샷 복원을 활성화할 때 스냅샷을 최적화하려면 TiB당 60분 걸립니다. Amazon 데이터 수명 주기 관리자에서 다음 스냅샷을 생성하기 전에 각 스냅샷이 완전히 최적화되도록 일정을 생성하는 것이 좋습니다.
 - 특정 가용 영역의 스냅샷에 대해 빠른 스냅샷 복원이 활성화된 1분마다 요금이 청구됩니다. 요금은 최소 1시간으로 비례 청구됩니다. 자세한 내용은 [요금 및 결제 \(p. 1047\)](#) 단원을 참조하십시오.

Note

수명 주기 정책의 구성에 따라 빠른 스냅샷 복원에 대해 여러 스냅샷을 동시에 활성화할 수 있습니다.

사전 조건

Amazon 데이터 수명 주기 관리자에는 다음과 같은 사전 조건이 필요합니다.

사전 조건

- Amazon 데이터 수명 주기 관리자 권한 (p. 1015)
 - IAM 사용자의 권한 (p. 1016)
 - 암호화된 스냅샷에 대한 권한 (p. 1017)

Amazon 데이터 수명 주기 관리자 권한

Amazon 데이터 수명 주기 관리자는 스냅샷을 관리하는 데 필요한 권한을 얻기 위해 IAM 역할을 사용합니다. AWS Management 콘솔을 사용하여 수명 주기 정책을 처음 생성하면 Amazon 데이터 수명 주기 관리자가 `AWSDataLifecycleManagerDefaultRole` 역할을 생성합니다. 다음 `create-default-role` 명령으로 이 역할을 생성할 수도 있습니다.

卷之三十一

또는 필요한 권한이 있는 사용자 지정 IAM 역할을 만든 다음, 수명 주기 정책을 생성할 때 이를 선택하는 방법도 있습니다.

사용자 지정 IAM 역할을 생성하려면

- ### 1. 다음 권한의 역할을 만듭니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [
```

```
        "ec2:CreateSnapshot",
        "ec2:CreateSnapshots",
        "ec2>DeleteSnapshot",
        "ec2:DescribeVolumes",
        "ec2:DescribeInstances",
        "ec2:DescribeSnapshots"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*::snapshot/*"
}
]
```

자세한 정보는 IAM 사용 설명서의 [역할 생성](#) 단원을 참조하십시오.

2. 역할에 신뢰 관계를 추가합니다.
 - a. IAM 콘솔에서 역할을 선택합니다.
 - b. 생성된 역할을 선택하고 신뢰 관계를 선택합니다.
 - c. 신뢰 관계 편집을 선택하고 다음 정책을 추가한 뒤 신뢰 정책 업데이트를 선택합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "dlm.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

IAM 사용자의 권한

IAM 사용자는 다음과 같은 Amazon 데이터 수명 주기 관리자 사용 권한을 가지고 있어야 합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": ["iam:PassRole", "iam>ListRoles"],
            "Resource": "arn:aws:iam::123456789012:role/AWSDataLifecycleManagerDefaultRole"
        },
        {
            "Effect": "Allow",
            "Action": "dlm:*",
            "Resource": "*"
        }
    ]
}
```

자세한 정보는 IAM 사용 설명서에서 [IAM 사용자의 권한 변경](#)을 참조하십시오.

암호화된 스냅샷에 대한 권한

리전 간에 암호화된 스냅샷을 복사하려면 AWS Key Management Service(AWS KMS)의 소스 및 대상 고객 마스터 키(CMK)에 대한 액세스 권한이 있어야 합니다.

소스 볼륨이 암호화된 경우 AWSDataLifecycleManagerDefaultRole 역할에 볼륨 암호화에 사용되는 CMK를 사용할 수 있는 권한이 있는지 확인합니다. 교차 리전 복사를 활성화하고 복사된 스냅샷을 암호화하도록 선택한 경우 AWSDataLifecycleManagerDefaultRole 역할에 대상 리전의 스냅샷을 암호화하는 데 필요한 CMK를 사용할 수 있는 권한이 있는지 확인합니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [AWS KMS CMK에 대한 액세스 관리](#)를 참조하십시오.

콘솔을 사용하여 백업 관리

다음 예제에서는 Amazon 데이터 수명 주기 관리자에서 AWS Management 콘솔을 사용하여 EBS 볼륨의 백업을 관리하는 방법을 보여 줍니다.

작업

- [수명 주기 정책 생성 \(p. 1017\)](#)
- [수명 주기 정책 보기 \(p. 1018\)](#)
- [수명 주기 정책 수정 \(p. 1018\)](#)
- [수명 주기 정책 삭제 \(p. 1018\)](#)

수명 주기 정책 생성

다음 절차에 따라 수명 주기 정책을 생성합니다.

수명 주기 정책 생성 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Elastic Block Store, Lifecycle Manager(수명 주기 관리자)를 선택한 다음, Create snapshot lifecycle policy(스냅샷 수명 주기 정책 생성)를 선택합니다.
3. 필요에 따라 정책에 대한 다음 정보를 제공합니다.
 - 설명 – 정책 설명입니다.
 - 리소스 유형 – 백업할 리소스의 유형입니다. `Volume`을 사용하여 개별 볼륨의 스냅샷을 생성하거나 `Instance`를 사용하여 인스턴스의 볼륨에서 다중 볼륨 스냅샷을 생성합니다.
 - 이러한 태그가 있는 대상 – 백업할 볼륨 또는 인스턴스를 식별하는 리소스 태그입니다.
 - 수명 주기 정책 태그 – 수명 주기 정책에 대한 태그입니다.
 - 일정 이름 – 일정의 이름입니다.
 - 빈도– 정책 실행 간격입니다. 매일, 매주, 매월 또는 매년 일정에 따라 정책 실행을 구성할 수 있습니다. 또는 사용자 지정 cron 표현식을 선택하여 최대 1년의 간격을 지정합니다. 자세한 내용은 Amazon CloudWatch Events 사용 설명서의 [Cron 표현식](#)을 참조하십시오.
 - hh:mm UTC에 시작 – 정책 실행을 시작하도록 예약된 시간입니다. 첫 번째 정책 실행은 예약된 시간 후 한 시간 이내에 시작됩니다.
 - 보존 유형– 총 스냅샷 수 또는 각 스냅샷 수명에 따라 스냅샷을 보존할 수 있습니다. 개수를 기준으로 보존하는 경우 범위는 1-1000입니다. 최대 수에 도달한 후에는 새 스냅샷을 생성할 때 가장 오래된 스냅샷이 삭제됩니다. 수명을 기반으로 보존하는 경우 범위는 1일-100년입니다. 각 스냅샷의 보존 기간이 만료되면 스냅샷이 삭제됩니다. 보존 기간은 생성 간격보다 크거나 같아야 합니다.
 - 정보 태그 지정–소스 볼륨에 있는 모든 사용자 정의 태그를 이 정책에 따라 생성된 스냅샷으로 복사할지 여부를 선택합니다. 또한 Amazon 데이터 수명 주기 관리자에서 적용되는 태그 외에도 스냅샷의 추가 태그를 지정할 수 있습니다. 리소스 유형이 INSTANCE인 경우 `instance-id` 및 `timestamp`와 같은 변수 태그를 사용하여 자동으로 스냅샷에 태그를 지정하도록 선택할 수 있습니다. 변수 태그의 값은 태그가 추가될 때 결정됩니다.

- **빠른 스냅샷 복원** – 이 정책에 의해 생성된 모든 스냅샷에 대해 빠른 스냅샷 복원을 활성화할지 여부를 선택합니다. 빠른 스냅샷 복원을 활성화하는 경우 이를 활성화할 가용 영역을 선택해야 합니다. 특정 가용 영역의 스냅샷에 대해 빠른 스냅샷 복원이 활성화된 1분마다 요금이 청구됩니다. 요금은 최소 1시간으로 비례 청구됩니다. 빠른 스냅샷 복원에 대해 활성화할 수 있는 최대 스냅샷 수도 지정할 수 있습니다.
- **교차 리전 복사 활성화** – 최대 3개의 추가 리전으로 각 스냅샷을 복사할 수 있습니다. 리전당 동시 스냅샷 복사본 수를 초과하지 않도록 해야 합니다. 리전마다 다른 보존 정책을 선택하고 태그를 모두 복사할지 또는 복사하지 않을지 선택할 수 있습니다. 소스 스냅샷이 암호화되거나 암호화가 기본적으로 활성화된 경우 스냅샷 복사가 암호화됩니다. 소스 스냅샷이 암호화 해제된 경우 암호화를 활성화할 수 있습니다. CMK를 지정하지 않으면 스냅샷은 각 대상 리전에서 EBS 암호화의 기본 키를 사용하여 암호화됩니다. 대상 리전에 대한 CMK를 지정하는 경우 CMK에 대한 액세스 권한이 있어야 합니다.
- **IAM 역할-스냅샷을 생성, 삭제 및 설명하고 볼륨을 설명할 권한이 있는 IAM 역할입니다.** AWS가 기본 역할 AWSDataLifecycleManagerDefaultRole을 제공하거나, 고객이 사용자 지정 IAM 역할을 만듭니다.
- **생성 후 정책 상태** – 다음 예약 시간에 정책 실행을 시작하려면 정책 활성화를 선택하고, 정책을 실행하지 않으려면 정책 비활성화를 선택합니다.

4. 정책 생성을 선택합니다.

수명 주기 정책 보기

다음 절차에 따라 수명 주기 정책을 봅니다.

수명 주기 정책을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Elastic Block Store, Lifecycle Manager(수명 주기 관리자)를 선택합니다.
3. 목록에서 수명 주기 정책을 선택합니다. 세부 정보 탭에는 정책에 대한 정보가 표시됩니다.

수명 주기 정책 수정

다음 절차에 따라 수명 주기 정책을 수정합니다.

수명 주기 정책을 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Elastic Block Store, Lifecycle Manager(수명 주기 관리자)를 선택합니다.
3. 목록에서 수명 주기 정책을 선택합니다.
4. 작업, 스냅샷 수명 주기 정책 수정을 선택합니다.
5. 필요에 따라 정책 설정을 수정합니다. 예를 들어 일정을 수정하거나, 태그를 추가 또는 제거하거나, 정책을 활성화 또는 비활성화 할 수 있습니다.
6. Update policy를 선택합니다.

수명 주기 정책 삭제

다음 절차에 따라 수명 주기 정책을 삭제합니다.

수명 주기 정책을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Elastic Block Store, Lifecycle Manager(수명 주기 관리자)를 선택합니다.
3. 목록에서 수명 주기 정책을 선택합니다.
4. 작업, 스냅샷 수명 주기 정책 삭제를 선택합니다.

5. 확인 메시지가 나타나면 스냅샷 수명 주기 정책 삭제를 선택합니다.

AWS CLI를 사용하여 백업 관리

다음 예에서는 Amazon 데이터 수명 주기 관리자에서 AWS CLI를 사용하여 EBS 볼륨의 백업을 관리하는 방법을 보여 줍니다.

예제:

- 수명 주기 정책 생성 (p. 1019)
- 수명 주기 정책 표시 (p. 1020)
- 수명 주기 정책 수정 (p. 1021)
- 수명 주기 정책 삭제 (p. 1021)

수명 주기 정책 생성

`create-lifecycle-policy` 명령을 사용하여 수명 주기 정책을 생성합니다. 구문을 간소화하기 위해 이 예제에서는 정책 세부 정보가 포함된 JSON 파일(`policyDetails.json`)을 참조합니다.

이 예제에서는 리소스 유형 `VOLUME`을 사용하여 지정된 대상 태그가 있는 모든 볼륨의 스냅샷을 생성합니다. 지정된 대상 태그가 있는 모든 인스턴스에 대한 모든 볼륨의 스냅샷을 생성하려면 리소스 유형 `INSTANCE`를 대신 사용합니다.

```
aws dlm create-lifecycle-policy --description "My volume policy" --state ENABLED --  
execution-role-arn arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRole --  
policy-details file://policyDetails.json
```

다음은 `policyDetails.json` 파일의 예입니다.

```
{  
    "ResourceTypes": [  
        "VOLUME"  
    ],  
    "TargetTags": [  
        {  
            "Key": "costcenter",  
            "Value": "115"  
        }  
    ],  
    "Schedules": [  
        {  
            "Name": "DailySnapshots",  
            "TagsToAdd": [  
                {  
                    "Key": "type",  
                    "Value": "myDailySnapshot"  
                }  
            ],  
            "CreateRule": {  
                "Interval": 24,  
                "IntervalUnit": "HOURS",  
                "Times": [  
                    "03:00"  
                ]  
            },  
            "RetainRule": {  
                "Count": 5  
            },  
            "CopyTags": false  
        }  
    ]  
}
```

```
        ]  
    }  
}
```

성공할 경우, 이 명령은 새로 생성된 정책의 ID를 반환합니다. 다음은 예제 출력입니다.

```
{  
    "PolicyId": "policy-0123456789abcdef0"  
}
```

수명 주기 정책 표시

`get-lifecycle-policy` 명령을 사용하여 수명 주기 정책에 대한 정보를 표시합니다.

```
aws dlm get-lifecycle-policy --policy-id policy-0123456789abcdef0
```

다음은 예제 출력입니다. 여기에는 사용자가 지정한 정보와 AWS에서 삽입한 메타데이터가 들어 있습니다.

```
{  
    "Policy": {  
        "Description": "My first policy",  
        "DateCreated": "2018-05-15T00:16:21+0000",  
        "State": "ENABLED",  
        "ExecutionRoleArn":  
            "arn:aws:iam::210774411744:role/AWSDataLifecycleManagerDefaultRole",  
        "PolicyId": "policy-0123456789abcdef0",  
        "DateModified": "2018-05-15T00:16:22+0000",  
        "PolicyDetails": {  
            "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",  
            "ResourceTypes": [  
                "VOLUME"  
            ],  
            "TargetTags": [  
                {  
                    "Value": "115",  
                    "Key": "costcenter"  
                }  
            ],  
            "Schedules": [  
                {  
                    "TagsToAdd": [  
                        {  
                            "Value": "myDailySnapshot",  
                            "Key": "type"  
                        }  
                    ],  
                    "RetainRule": {  
                        "Count": 5  
                    },  
                    "CopyTags": false,  
                    "CreateRule": {  
                        "Interval": 24,  
                        "IntervalUnit": "HOURS",  
                        "Times": [  
                            "03:00"  
                        ]  
                    },  
                    "Name": "DailySnapshots"  
                }  
            ]  
        }  
    }  
}
```

}

수명 주기 정책 수정

[update-lifecycle-policy](#) 명령을 사용하여 수명 주기 정책의 정보를 수정합니다. 구문을 간단히 하기 위해 이 예에서는 정책의 세부 정보가 들어 있는 JSON 파일(policyDetailsUpdated.json)을 참조합니다.

```
aws dlm update-lifecycle-policy --state DISABLED --execution-role-arn arn:aws:iam::12345678910:role/AWSDataLifecycleManagerDefaultRole" --policy-details file://policyDetailsUpdated.json
```

다음은 policyDetailsUpdated.json 파일의 예입니다.

```
{
    "ResourceTypes": [
        "VOLUME"
    ],
    "TargetTags": [
        {
            "Key": "costcenter",
            "Value": "120"
        }
    ],
    "Schedules": [
        {
            "Name": "DailySnapshots",
            "TagsToAdd": [
                {
                    "Key": "type",
                    "Value": "myDailySnapshot"
                }
            ],
            "CreateRule": {
                "Interval": 12,
                "IntervalUnit": "HOURS",
                "Times": [
                    "15:00"
                ]
            },
            "RetainRule": {
                "Count": 5
            },
            "CopyTags": false
        }
    ]
}
```

업데이트된 정책을 보려면 [get-lifecycle-policy](#) 명령을 사용하십시오. 상태, 태그 값, 스냅샷 간격, 스냅샷 시작 시간이 변경된 것을 알 수 있습니다.

수명 주기 정책 삭제

[delete-lifecycle-policy](#) 명령을 사용하여 수명 주기 정책을 삭제하고, 정책에 지정된 대상 태그를 해제하여 재사용할 수 있도록 합니다.

```
aws dlm delete-lifecycle-policy --policy-id policy-0123456789abcdef0
```

API를 사용하여 백업 관리

[Amazon 데이터 수명 주기 관리자 API](#) 참조에 Amazon 데이터 수명 주기 관리자 쿼리 API의 데이터 유형과 각 작업에 대한 설명 및 구문이 나와 있습니다.

또는 AWS SDK 중 하나를 사용하여 사용 중인 프로그래밍 언어나 플랫폼에 맞게 조정된 방식으로 API에 액세스할 수도 있습니다. 자세한 정보는 [AWS SDK](#)를 참조하십시오.

스냅샷 수명 주기 모니터링

다음 기능을 사용하여 스냅샷의 수명 주기를 모니터링할 수 있습니다.

기능

- 콘솔 및 AWS CLI (p. 1022)
- CloudWatch 이벤트 (p. 1022)
- AWS CloudTrail (p. 1023)

콘솔 및 AWS CLI

Amazon EC2 콘솔 또는 AWS CLI를 사용하여 수명 주기 정책을 볼 수 있습니다. 정책에 따라 생성된 각 스냅샷에는 타임스탬프 및 해당 정책과 관련된 태그가 있습니다. 이 태그로 스냅샷을 필터링하여 백업이 의도대로 생성되고 있는지 확인할 수 있습니다. 콘솔에서 수명 주기 정책을 보는 방법에 대한 자세한 내용은 [수명 주기 정책 보기 \(p. 1018\)](#) 단원을 참조하십시오. CLI를 사용하여 수명 주기 정책에 대한 정보를 표시하는 방법은 [수명 주기 정책 표시 \(p. 1020\)](#) 단원을 참조하십시오.

CloudWatch 이벤트

Amazon EBS 및 Amazon 데이터 수명 주기 관리자는 수명 주기 정책 작업과 관련된 이벤트를 발생시킵니다. AWS Lambda 및 Amazon CloudWatch Events를 사용하여 이벤트 알림을 프로그래밍 방식으로 처리할 수 있습니다. 자세한 내용은 [Amazon CloudWatch Events 사용 설명서](#)를 참조하십시오.

다음과 같은 이벤트를 사용할 수 있습니다.

- `createSnapshot - CreateSnapshot` 작업이 성공하거나 실패할 때 생기는 Amazon EBS 이벤트입니다. 자세한 내용은 [Amazon EBS용 Amazon CloudWatch Events \(p. 1081\)](#) 단원을 참조하십시오.
- `DLM Policy State Change` - 수명 주기 정책이 오류 상태가 될 때 생기는 Amazon 데이터 수명 주기 관리자 이벤트입니다. 이 이벤트에는 오류의 원인에 대한 설명이 포함되어 있습니다. 다음은 IAM 역할에서 부여한 권한이 충분할 때 나타나는 이벤트의 예입니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-0123456789ab",  
    "detail-type": "DLM Policy State Change",  
    "source": "aws.dlm",  
    "account": "123456789012",  
    "time": "2018-05-25T13:12:22Z",  
    "region": "us-east-1",  
    "resources": [  
        "arn:aws:dml:us-east-1:123456789012:policy/policy-0123456789abcdef"  
    ],  
    "detail": {  
        "state": "ERROR",  
        "cause": "Role provided does not have sufficient permissions",  
        "policy_id": "arn:aws:dml:us-east-1:123456789012:policy/policy-0123456789abcdef"  
    }  
}
```

다음은 한도를 초과할 때 발생하는 이벤트의 예입니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-0123456789ab",  
    "detail-type": "DLM Policy State Change",  
    "source": "aws.dlm",  
    "account": "123456789012",  
    "time": "2018-05-25T13:12:22Z",  
    "region": "us-east-1",  
    "resources": [  
        "arn:aws:dml:us-east-1:123456789012:policy/policy-0123456789abcdef"  
    ],  
    "detail": {  
        "state": "OVERLIMIT",  
        "cause": "The number of snapshots for this policy has reached the limit.",  
        "policy_id": "arn:aws:dml:us-east-1:123456789012:policy/policy-0123456789abcdef"  
    }  
}
```

```
"source": "aws.dlm",
"account": "123456789012",
"time": "2018-05-25T13:12:22Z",
"region": "us-east-1",
"resources": [
    "arn:aws:dlm:us-east-1:123456789012:policy/policy-0123456789abcdef"
],
"detail": {
    "state": "ERROR",
    "cause": "Maximum allowed active snapshot limit exceeded",
    "policy_id": "arn:aws:dlm:us-east-1:123456789012:policy/policy-0123456789abcdef"
}
}
```

AWS CloudTrail

AWS CloudTrail로 사용자 활동과 API 사용을 추적하여 내부 정책 및 규제 표준에 대한 준수 사실을 입증할 수 있습니다. 자세한 정보는 [AWS CloudTrail User Guide](#)를 참조하십시오.

Amazon EBS 데이터 서비스

Amazon EBS는 다음 데이터 서비스를 제공합니다.

데이터 서비스

- [Amazon EBS 탄력적 볼륨 \(p. 1023\)](#)
- [Amazon EBS 암호화 \(p. 1033\)](#)
- [Amazon EBS 빠른 스냅샷 복원 \(p. 1044\)](#)

Amazon EBS 탄력적 볼륨

Amazon EBS 탄력적 볼륨을 통해 볼륨 크기를 늘리거나 볼륨 유형을 변경하거나 EBS 볼륨의 성능을 조정할 수 있습니다. 인스턴스가 탄력적 볼륨을 지원하는 경우에는 볼륨을 분리하거나 인스턴스를 재시작하지 않고도 이것이 가능합니다. 따라서 변경 사항이 적용되는 동안 애플리케이션을 계속 사용할 수 있습니다.

볼륨 구성 설정은 무료입니다. 볼륨 설정이 시작된 후 새 볼륨 구성에 대한 요금이 청구됩니다. 자세한 내용은 [Amazon EBS 요금](#) 페이지를 참조하십시오.

목차

- [볼륨 수정 시 요구 사항 \(p. 1023\)](#)
- [EBS 볼륨에 대한 수정 요청 \(p. 1025\)](#)
- [볼륨 수정 진행 상황 모니터링 \(p. 1027\)](#)
- [볼륨 크기 조정 후 Windows 파일 시스템 확장 \(p. 1030\)](#)

볼륨 수정 시 요구 사항

Amazon EBS 볼륨을 수정할 때 다음과 같은 요구 사항과 제한 사항이 적용됩니다. EBS 볼륨에 대한 일반 요구 사항에 대한 자세한 내용은 [EBS 볼륨의 크기 및 구성에 대한 제약 \(p. 945\)](#)을 참조하십시오.

지원되는 인스턴스 유형

탄력적 볼륨을 지원하는 인스턴스는 다음과 같습니다.

- 모든 [현재 세대 인스턴스 \(p. 115\)](#)
- 이전 세대 인스턴스: C1, C3, CC2, CR1, G2, I2, M1, M3, R3

인스턴스 유형에서 탄력적 볼륨을 지원하지 않는 경우에는 [탄력적 볼륨이 지원되지 않는 경우의 EBS 볼륨 설정 \(p. 1027\)](#) 단원을 참조하십시오.

Windows 볼륨에 대한 요구 사항

기본적으로 Windows는 마스터 부트 레코드(MBR) 파티션 테이블을 사용하여 볼륨을 초기화합니다. MBR는 2TiB(2,048GiB) 미만의 볼륨만 지원하기 때문에 Windows에서는 MBR 볼륨 크기를 이 한도를 넘는 크기로 변경할 수 없습니다. 이렇게 하면 Windows 디스크 관리 유ти리티에서 볼륨 확장 옵션이 비활성화됩니다. AWS Management 콘솔이나 AWS CLI를 사용하여 크기 제한을 초과하는 MBR 파티션 볼륨을 생성하면 Windows에서 추가 공간을 검색하거나 사용할 수 없습니다. Linux 볼륨에 적용되는 요구 사항의 경우에는 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux 볼륨에 대한 요구 사항](#)을 참조하십시오.

이 제한을 해결하려면 GUID 파티션 테이블(GPT)을 사용하여 더 큰 볼륨을 새로 만든 후 원래 MBR 볼륨의 데이터를 복사합니다.

GPT 볼륨을 생성하려면

1. EC2 인스턴스의 가용 영역에서 원하는 크기의 비어 있는 새 볼륨을 생성하고 이 볼륨을 인스턴스에 연결합니다.

Note

새 볼륨은 스냅샷에서 복원한 볼륨이 아니어야 합니다.

2. Windows 시스템에 로그인하고 디스크 관리(diskmgmt.exe)를 엽니다.
3. 새 디스크를 마우스 오른쪽 버튼으로 클릭하여 컨텍스트 메뉴를 열고 온라인을 선택합니다.
4. 디스크 초기화 창에서 새 디스크를 선택하고 GPT(GUID 파티션 테이블), 확인을 선택합니다.
5. 초기화가 완료되면 robocopy 또는 teracopy 등의 도구를 사용하여 원래 볼륨의 데이터를 새 볼륨으로 복사합니다.
6. 디스크 관리에서 드라이브 문자를 적절한 값으로 변경하고 기존 볼륨을 오프라인으로 전환합니다.
7. Amazon EC2 콘솔에서 기존 볼륨을 인스턴스에서 분리하고, 인스턴스를 재부팅하여 인스턴스가 제대로 작동하는지 확인한 다음, 기존 볼륨을 삭제합니다.

제한 사항

- 다중 연결 지원 Amazon EBS 볼륨에서는 탄력적 볼륨 작업이 지원되지 않습니다.
- 새로운 볼륨 크기는 지원되는 볼륨 용량을 초과할 수 없습니다. 자세한 내용은 [EBS 볼륨의 크기 및 구성에 대한 제약 \(p. 945\)](#) 단원을 참조하십시오.
- 볼륨이 2016년 11월 3일 23:40 UTC 이전에 연결된 경우에는 탄력적 볼륨 지원을 초기화해야 합니다. 자세한 내용은 [탄력적 볼륨 지원 초기화 \(p. 1026\)](#)를 참조하십시오.
- 지원되지 않는 이전 세대의 인스턴스 유형을 사용하거나 볼륨을 수정하려는 중에 오류가 발생한 경우, [탄력적 볼륨이 지원되지 않는 경우의 EBS 볼륨 수정 \(p. 1027\)](#) 단원을 참조하십시오.
- 루트 볼륨으로 인스턴스에 연결된 gp2 볼륨은 st1 또는 sc1 볼륨으로 수정될 수 없습니다. 분리되거나 st1 또는 sc1으로 수정되었다면, 루트 볼륨으로 인스턴스에 연결할 수 없습니다.
- 요청한 볼륨 크기가 st1과 sc1의 최소 크기 이하라면, gp2 볼륨은 st1 또는 sc1 볼륨으로 수정될 수 없습니다.
- 경우에 따라 수정을 진행하기 위해 볼륨을 분리하거나 인스턴스를 중단해야 합니다. EBS 볼륨을 수정하고 시도할 때 오류 메시지가 표시되는 경우 또는 이전 세대 인스턴스 유형에 연결된 EBS 볼륨을 수정하는 경우 다음 중 한 가지 조치를 취하십시오.
 - 루트가 아닌 볼륨의 경우, 인스턴스에서 볼륨을 분리하고 수정 사항을 적용한 다음 볼륨을 다시 연결합니다.
 - 루트(부트) 볼륨의 경우, 인스턴스를 중단하고 수정 사항을 적용한 다음 인스턴스를 다시 시작합니다.
- 기존 io1 또는 io2 볼륨에서 32,000 IOPS를 초과하여 프로비저닝하려는 경우 다음 중 하나를 수행하여 전체 성능 개선을 확인해야 할 수도 있습니다.
 - 볼륨을 분리한 후 다시 연결합니다.

- 인스턴스를 다시 시작합니다.
- EBS 볼륨 크기 축소는 지원되지 않습니다. 그러나 비슷한 볼륨을 만든 후 robocopy 같은 애플리케이션 수준 도구를 사용하여 그 볼륨으로 데이터를 마이그레이션할 수 있습니다.
- 완전히 초기화되지 않은 볼륨을 수정하는 경우 수정 시간이 늘어납니다. 자세한 내용은 [Amazon EBS 볼륨 초기화 \(p. 1066\)](#) 단원을 참조하십시오.
- 볼륨을 수정한 후 동일한 볼륨에 추가 수정 사항을 적용하려면 볼륨이 `in-use` 또는 `available` 상태가 되도록 6시간 이상 기다립니다.
- `m3.medium` 인스턴스는 볼륨 수정을 완전하게 지원하지만 `m3.large`, `m3.xlarge` 및 `m3.2xlarge` 인스턴스는 일부 볼륨 수정 기능을 지원하지 않을 수 있습니다.

EBS 볼륨에 대한 수정 요청

탄력적 볼륨을 사용하면 Amazon EBS 볼륨을 분리하지 않고도 크기, 성능 및 볼륨 크기를 동적으로 수정할 수 있습니다.

다음 절차에 따라 볼륨을 수정합니다.

1. (선택 사항) 중요한 데이터가 포함된 볼륨을 수정하려면 먼저 변경 내용을 룰백해야 할 경우를 대비하여 볼륨의 스냅샷을 생성하는 것이 바람직합니다. 자세한 내용은 [Amazon EBS 스냅샷 생성 \(p. 969\)](#) 단원을 참조하십시오.
2. 볼륨 수정을 요청합니다.
3. 볼륨 수정의 진행 상황을 모니터링합니다. 자세한 내용은 [볼륨 수정 진행 상황 모니터링 \(p. 1027\)](#) 단원을 참조하십시오.
4. 볼륨 크기가 수정된 경우 볼륨의 파일 시스템을 확장하여 스토리지 용량 증가를 활용합니다. 자세한 내용은 [볼륨 크기 조정 후 Windows 파일 시스템 확장 \(p. 1030\)](#) 단원을 참조하십시오.

목차

- [탄력적 볼륨을 사용하여 EBS 볼륨 수정\(콘솔\) \(p. 1025\)](#)
- [탄력적 볼륨을 사용하여 EBS 볼륨 수정\(AWS CLI\) \(p. 1026\)](#)
- [탄력적 볼륨 지원 초기화\(필요한 경우\) \(p. 1026\)](#)
- [탄력적 볼륨이 지원되지 않는 경우의 EBS 볼륨 수정 \(p. 1027\)](#)

탄력적 볼륨을 사용하여 EBS 볼륨 수정(콘솔)

다음 절차를 사용하여 EBS 볼륨을 수정합니다.

콘솔을 사용하여 EBS 볼륨을 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 볼륨을 선택하고, 수정할 볼륨을 선택한 다음, 작업, 볼륨 수정을 선택합니다.
3. 볼륨 수정 창에 볼륨 ID와 유형, 크기, IOPS를 포함한 볼륨의 현재 구성이 표시됩니다. 한 번의 작업으로 이 모든 설정을 변경할 수 있습니다. 다음과 같이 새로운 구성 값을 설정합니다.
 - 유형을 수정하려면 볼륨 유형의 값을 선택합니다.
 - 크기를 수정하려면 크기에 허용된 정수 값을 입력합니다.
 - 볼륨 유형으로 프로비저닝된 IOPS SSD(IO1) 또는 프로비저닝된 IOPS SSD(IO2)를 선택한 경우에는 IOPS에 허용되는 정수 값을 입력합니다.
4. 볼륨 설정 변경을 완료했으면 수정을 선택합니다. 확인 메시지가 표시되면 예를 선택합니다.
5. 볼륨 크기를 수정해도 새 스토리지 용량을 활용하기 위해 볼륨의 파일 시스템도 확장할 때까지는 실질적인 효과가 없습니다. 자세한 내용은 [볼륨 크기 조정 후 Windows 파일 시스템 확장 \(p. 1030\)](#) 단원을 참조하십시오.

- AWS NVMe 드라이버가 없는 인스턴스에서 NVMe 볼륨의 크기를 늘릴 경우 인스턴스를 재부팅하여 Windows를 활성화하고 새 볼륨의 크기를 확인해야 합니다. AWS NVMe 드라이버 설치에 대한 자세한 내용은 [Windows 인스턴스의 AWS NVMe 드라이버 \(p. 539\)](#) 단원을 참조하십시오.

탄력적 볼륨을 사용하여 EBS 볼륨 수정(AWS CLI)

`modify-volume` 명령을 사용하여 볼륨의 구성 설정을 하나 이상 수정합니다. 예를 들어 크기가 100GiB이고 유형이 gp2인 볼륨을 가지고 있는 경우, 다음 명령이 IOPS가 10,000이고 크기가 200GiB이며 유형이 io1인 볼륨에 대한 구성을 변경합니다.

```
aws ec2 modify-volume --volume-type io1 --iops 10000 --size 200 --volume-id vol-1111111111111111
```

다음은 예제 출력입니다.

```
{  
    "VolumeModification": {  
        "TargetSize": 200,  
        "TargetVolumeType": "io1",  
        "ModificationState": "modifying",  
        "VolumeId": "vol-1111111111111111",  
        "TargetIops": 10000,  
        "StartTime": "2017-01-19T22:21:02.959Z",  
        "Progress": 0,  
        "OriginalVolumeType": "gp2",  
        "OriginalIops": 300,  
        "OriginalSize": 100  
    }  
}
```

볼륨 크기를 수정해도 새 스토리지 용량을 활용하기 위해 볼륨의 파일 시스템도 확장할 때까지는 실질적인 효과가 없습니다. 자세한 내용은 [볼륨 크기 조정 후 Windows 파일 시스템 확장 \(p. 1030\)](#) 단원을 참조하십시오.

탄력적 볼륨 지원 초기화(필요한 경우)

2016년 11월 3일 23:40 UTC 이전에 인스턴스에 연결된 볼륨을 수정하기 전에 다음 중 한 가지 조치를 취하여 볼륨 수정 지원을 초기화해야 합니다.

- 볼륨을 분리한 후 다시 연결합니다.
- 인스턴스 중지 및 시작

다음 절차 중 하나를 사용하여 인스턴스가 볼륨 수정이 가능한 상태인지 확인합니다.

인스턴스가 콘솔을 사용할 준비가 되었는지를 확인하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 인스턴스를 선택합니다.
- 열 표시/숨기기(Show/Hide Columns) 아이콘(기어 모양)을 선택합니다. 시작 시간 및 블록 디바이스 속성을 선택한 다음 닫기를 선택합니다.
- 시작 시간(Launch Time) 열을 기준으로 인스턴스의 목록을 정렬합니다. 컷오프 날짜 이전에 시작된 인스턴스의 경우 디바이스가 연결된 시기를 확인합니다. 다음 예에서, 첫 번째 인스턴스가 컷오프 날짜 이전에 시작되었으며 해당 루트 볼륨이 컷오프 날짜 이전에 연결되었으므로 첫 번째 인스턴스에 대한 볼륨 수정을 초기화해야 합니다. 다른 인스턴스는 컷오프 날짜 이후에 시작되었으므로 사용 준비가 되었습니다.

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 데이터 서비스

Instance ID	Launch Time	Block Devices
i-e905622e	February 25, 2016 at 1:49:35 PM UTC-8	/dev/xvda=vol-e6b46410:attached:2016-02-25T21:49:35.000Z:true
i-719f99a8	December 8, 2016 at 2:21:51 PM UTC-8	/dev/xvda=vol-bad60e7a:attached:2016-01-15T18:36:12.000Z:true
i-006b02c1b78381e57	May 17, 2017 at 1:52:52 PM UTC-7	/dev/sda=vol-0de9250441c73024c:attached:2017-05-17T20:52:53.000Z:true, xvdb=vol-0863a86c393496d3d:attached:2017-05-17T20:52:53.000Z:false
i-e3d172ed	May 17, 2017 at 2:48:54 PM UTC-7	/dev/sda=vol-04c34d0b:attached:2015-01-21T21:19:46.000Z:true

인스턴스가 CLI를 사용할 준비가 되었는지를 확인하려면

다음 [describe-instances](#) 명령을 사용하여 2016년 11월 3일 23:40 UTC 전에 볼륨이 연결되었는지 여부를 확인합니다.

```
aws ec2 describe-instances --query "Reservations[*].Instances[*].[InstanceId,LaunchTime<='2016-11-01',BlockDeviceMappings[*][Ebs.AttachTime<='2016-11-01']]"
--output text
```

각 인스턴스의 출력 첫 줄에는 해당 ID와 컷오프 날짜 이전에 시작되었는지 여부(True 또는 False)가 표시됩니다. 첫 줄 다음에는 각 EBS 볼륨이 컷오프 날짜 이전에 연결되었는지 여부를 보여주는(True 또는 False) 줄이 하나 이상 뒤따라 표시됩니다. 다음 예제 출력에서, 첫 번째 인스턴스가 컷오프 날짜 이전에 시작되었으며 해당 루트 볼륨이 컷오프 날짜 이전에 연결되었으므로 첫 번째 인스턴스에 대한 볼륨 설정을 초기화해야 합니다. 다른 인스턴스는 컷오프 날짜 이후에 시작되었으므로 사용 준비가 되었습니다.

```
i-e905622e      True
True
i-719f99a8      False
True
i-006b02c1b78381e57  False
False
False
i-e3d172ed      False
True
```

탄력적 볼륨이 지원되지 않는 경우의 EBS 볼륨 수정

지원되는 인스턴스 유형을 사용하고 있는 경우에는 탄력적 볼륨을 이용해 Amazon EBS 볼륨을 분리하지 않고도 크기, 성능 및 볼륨 유형을 동적으로 수정할 수 있습니다.

탄력적 볼륨을 사용할 수는 없지만 루트(부트) 볼륨을 수정해야 하는 경우에는 인스턴스를 중지하고 볼륨을 수정한 후 인스턴스를 다시 시작해야 합니다.

인스턴스가 시작된 후 파일 시스템의 크기를 확인하여 인스턴스가 더 큰 볼륨 공간을 인식하는지 파악할 수 있습니다.

새로 확장된 볼륨이 크기에 반영되지 않을 경우 인스턴스에서 새 공간을 사용할 수 있도록 디바이스의 파일 시스템을 확장해야 합니다. 자세한 내용은 단원을 참조하십시오.[볼륨 크기 조정 후 Windows 파일 시스템 확장 \(p. 1030\)](#)

볼륨을 사용하려면 온라인에 접속해야 할 수 있습니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오. 볼륨을 다시 포맷할 필요는 없습니다.

볼륨 수정 진행 상황 모니터링

수정 시 EBS 볼륨은 상태 시퀀스를 통과합니다. 볼륨은 `modifying` 상태가 된 다음 `optimizing` 상태가 되고, 마지막으로 `completed` 상태가 됩니다. 그러면 볼륨을 더 수정할 준비가 완료됩니다.

Note

드물지만 일시적 AWS 결함으로 인해 `failed` 상태가 될 수 있습니다. 이는 볼륨 상태를 나타내는 것이 아니라 단지 볼륨 수정이 실패했음을 나타내는 것입니다. 이 경우 볼륨 수정을 다시 시도합니다.

볼륨이 optimizing 상태에 있는 동안 볼륨 성능은 소스 및 대상 구성 사양 사이에 있습니다. 일시적인 볼륨 성능은 소스 볼륨 성능 이상입니다. IOPS를 다운로드하면 일시적인 볼륨 성능은 대상 볼륨 성능 이상입니다.

볼륨 수정 변경 사항은 다음과 같이 적용됩니다.

- 크기를 변경하면 볼륨이 optimizing 상태가 된 후 완료되어 적용되는 데 대개 몇 초가 걸립니다.
- 성능(IOPS) 변경이 완료되는 데 몇 분에서 몇 시간이 걸릴 수 있으며, 시간은 현재 수행 중인 구성 변경에 따라 달라집니다.
- 새 구성이 적용되려면 최대 24시간이 걸리며, 볼륨이 완전히 초기화되지 않은 경우처럼 특별한 경우에는 더 걸릴 수도 있습니다. 일반적으로 완전히 사용된 1TiB 볼륨은 새 성능 구성으로 마이그레이션하는 데 약 6시간이 걸립니다.

다음 방법 중 하나를 사용하여 볼륨 수정의 진행 상황을 모니터링합니다.

목차

- [볼륨 수정의 진행 상황 모니터링\(콘솔\)](#) (p. 1028)
- [볼륨 수정의 진행 상황 모니터링\(AWS CLI\)](#) (p. 1029)
- [볼륨 수정의 진행 상황 모니터링\(CloudWatch 이벤트\)](#) (p. 1030)

볼륨 수정의 진행 상황 모니터링(콘솔)

다음 절차를 사용하여 하나 이상의 볼륨 수정 진행 상황을 모니터링합니다.

콘솔을 사용하여 수정의 진행 상황을 모니터링하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 볼륨을 선택합니다.
- 볼륨을 선택합니다.
- 세부 정보 창의 상태(State) 열과 상태(State) 필드에는 volume-state - modification-state (progress%) 형식의 정보가 들어 있습니다. 가능한 볼륨 상태는 creating, available, in-use, deleting, deleted, error입니다. 가능한 수정 상태는 modifying, optimizing, completed입니다. 볼륨 수정이 완료된 직후 수정 상태와 진행 상황을 제거하고 볼륨 상태만 남깁니다.

이 예에서는 선택한 볼륨의 수정 상태가 optimizing입니다. 다음 볼륨의 수정 상태는 modifying입니다.

The screenshot shows the AWS CloudWatch Metrics Insights interface with a query for monitoring EBS volume modification progress. The results table lists four volumes, with the second one (vol-02940f6ee433f...) currently being modified. A detailed view of this volume shows its current state as 'in-use - optimizing (1%)' and provides a 'Volume modification details' summary table.

Name	Volume ID	Size	Volume Type	IOPS	Snapshot	Created	Availability Zone	State
	vol-0dda54cd90f5...	8 GiB	gp2	100	snap-09aa45c...	January 9, 2020 at ...	eu-west-1b	in-use
	vol-02940f6ee433f...	16 GiB	gp2	100	snap-076d641...	January 9, 2020 at ...	eu-west-1c	in-use - optimizing (1%)
	Windows-ins...	8 GiB	gp2	100		October 11, 2019 at ...	eu-west-1a	available - modifying (0%)
	attach-vol-te...	100 GiB	gp2	300		January 30, 2019 at ...	eu-west-1b	available

Volumes: vol-02940f6ee433f...

Description Status Checks Monitoring Tags

Volume ID: vol-02940f6ee433f...
Size: 16 GiB
Created: January 9, 2020 at 2:08:04 PM UTC+2
State: in-use - optimizing (1%)

Attachment information:
Volume type: gp2
Product codes: -
IOPS: 100

Volume modification details:

Original Volume Type	gp2
Original Size	8
Original IOPS	100
Target Volume Type	gp2
Target Size	16
Target IOPS	100
Status message	-

Alarm status: None
Snapshot: snap-076d641...
Availability Zone: eu-west-1c
Encryption: Not Encrypted
KMS Key ID:
KMS Key Aliases:
KMS Key ARN:
Multi-Attach Enabled: No

- 이전 단계와 같이 세부 정보 창의 상태(State) 필드에서 텍스트를 선택하여 가장 최근의 수정 작업에 대한 정보를 표시합니다.

볼륨 수정의 진행 상황 모니터링(AWS CLI)

`describe-volumes-modifications` 명령을 사용하여 하나 이상의 볼륨 수정 진행 상황을 모니터링합니다. 다음 예제에서는 두 볼륨의 볼륨 수정을 설명합니다.

```
aws ec2 describe-volumes-modifications --volume-id vol-1111111111111111 vol-2222222222222222
```

다음 예제 출력에서 볼륨 수정의 여전히 `modifying` 상태입니다. 진행률은 백분율로 보고됩니다.

```
{  
    "VolumesModifications": [  
        {  
            "TargetSize": 200,  
            "TargetVolumeType": "io1",  
            "ModificationState": "modifying",  
            "VolumeId": "vol-1111111111111111",  
            "TargetIops": 10000,  
            "StartTime": "2017-01-19T22:21:02.959Z",  
            "Progress": 0,  
            "OriginalVolumeType": "gp2",  
            "OriginalIops": 300,  
            "OriginalSize": 100  
        },  
        {  
            "TargetSize": 2000,  
            "TargetVolumeType": "sc1",  
            "ModificationState": "modifying",  
            "VolumeId": "vol-2222222222222222",  
            "StartTime": "2017-01-19T22:23:22.158Z",  
            "Progress": 0,  
            "OriginalVolumeType": "gp2",  
            "OriginalIops": 300,  
            "OriginalSize": 1000  
        }  
    ]  
}
```

다음 예제에서는 수정 상태가 `optimizing` 또는 `completed`인 모든 볼륨을 설명하고, 2017년 2월 1일 이후 시작된 수정만 표시하도록 결과를 필터링 및 형식 지정합니다.

```
aws ec2 describe-volumes-modifications --filters Name=modification-state,Values="optimizing","completed" --query "VolumesModifications[?StartTime>='2017-02-01'].{ID:VolumeId,STATE:ModificationState}"
```

다음은 두 볼륨에 대한 정보가 포함된 출력 예제입니다.

```
[  
    {  
        "STATE": "optimizing",  
        "ID": "vol-06397e7a0eEXAMPLE"  
    },  
    {  
        "STATE": "completed",  
        "ID": "vol-ba74e18c2aEXAMPLE"  
    }  
]
```

볼륨 수정의 진행 상황 모니터링(CloudWatch 이벤트)

CloudWatch 이벤트를 사용하여 볼륨 수정 이벤트에 대한 알림 규칙을 생성할 수 있습니다. 규칙을 사용하여 Amazon SNS로 알림 메시지를 생성하거나 일치하는 이벤트에 대한 응답으로 Lambda 함수를 호출할 수 있습니다.

CloudWatch 이벤트를 사용하여 수정 진행 상황을 모니터링하려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 이벤트, 규칙 생성을 선택합니다.
3. 서비스별 이벤트와 일치시킬 이벤트 패턴을 작성에 대해 사용자 지정 이벤트 패턴을 선택합니다.
4. 사용자 지정 이벤트 패턴 작성의 내용을 다음과 같이 바꾸고 저장을 선택합니다.

```
{  
    "source": [  
        "aws.ec2"  
    ],  
    "detail-type": [  
        "EBS Volume Notification"  
    ],  
    "detail": {  
        "event": [  
            "modifyVolume"  
        ]  
    }  
}
```

다음은 이벤트 데이터 예제입니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-012345678901",  
    "detail-type": "EBS Volume Notification",  
    "source": "aws.ec2",  
    "account": "012345678901",  
    "time": "2017-01-12T21:09:07Z",  
    "region": "us-east-1",  
    "resources": [  
        "arn:aws:ec2:us-east-1:012345678901:volume/vol-03a55cf56513fa1b6"  
    ],  
    "detail": {  
        "result": "optimizing",  
        "cause": "",  
        "event": "modifyVolume",  
        "request-id": "01234567-0123-0123-0123-0123456789ab"  
    }  
}
```

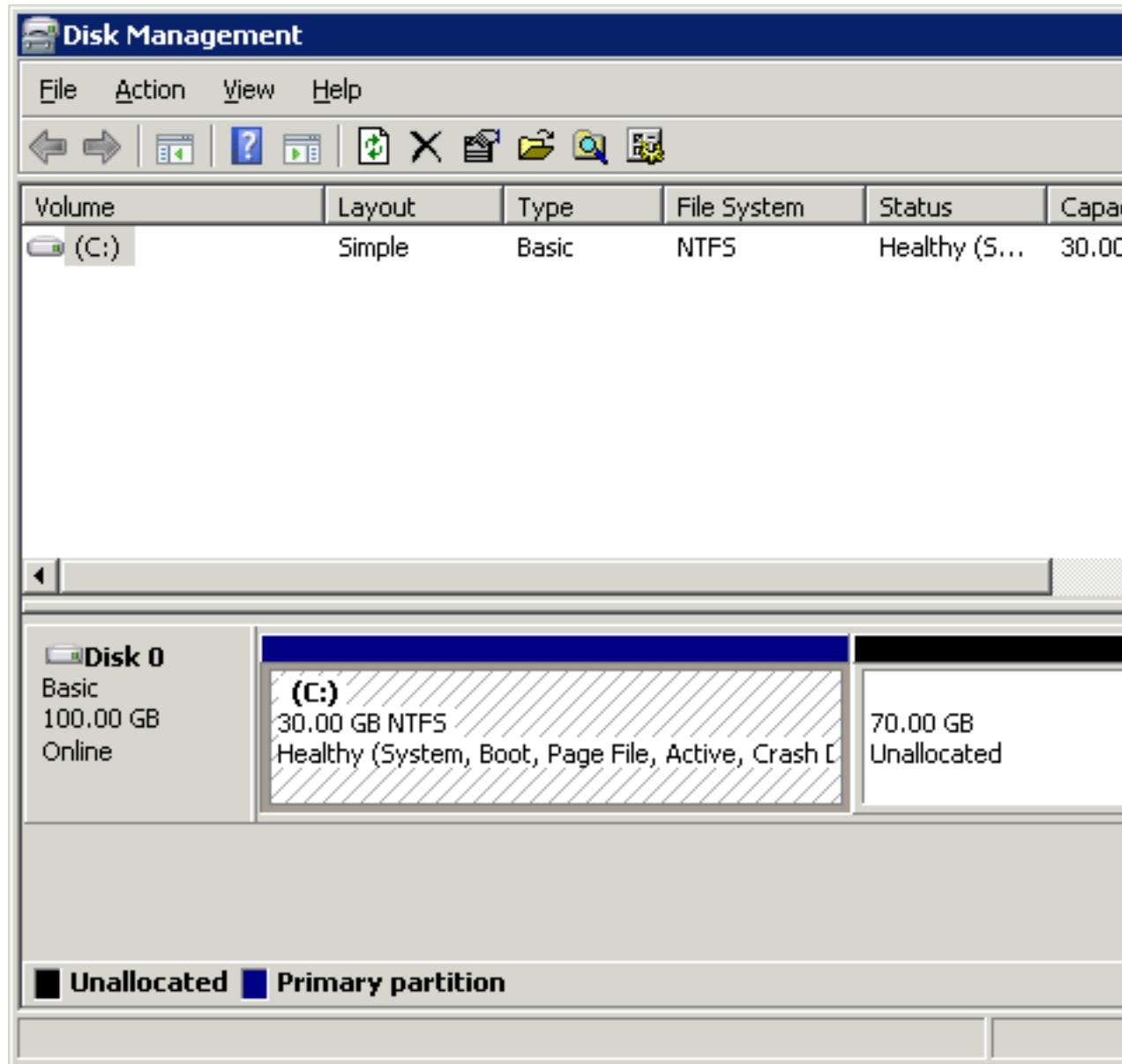
볼륨 크기 조정 후 Windows 파일 시스템 확장

EBS 볼륨 크기를 늘리고 난 후에는 Windows Disk Management 유ти리티를 사용하여 디스크 크기를 볼륨의 새 크기로 늘립니다. 볼륨이 optimizing 상태가 되자마자 파일 시스템 크기 조정을 시작할 수 있습니다. 이 유ти리티에 대한 자세한 내용은 Microsoft Docs 웹 사이트에서 [기본 볼륨 확장](#)을 참조하십시오.

Linux에서의 파일 시스템 확장에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [볼륨 크기 조정 후 Linux 파일 시스템 확장](#)을 참조하십시오.

Windows 파일 시스템을 확장하려면

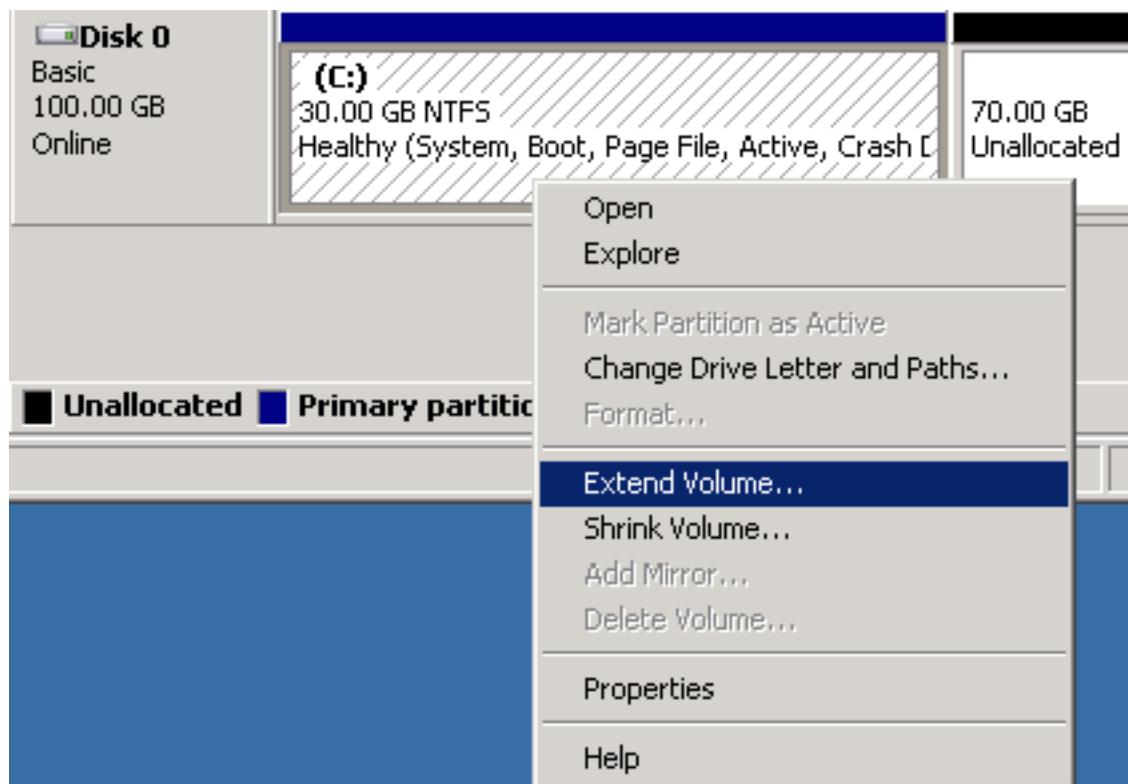
1. 중요한 데이터가 저장된 파일 시스템을 확장하려면 먼저 변경 내용을 둘백해야 할 경우를 대비하여 파일 시스템이 저장된 볼륨 스냅샷을 생성하는 것이 바람직합니다. 자세한 내용은 [Amazon EBS 스냅샷 생성 \(p. 969\)](#) 단원을 참조하십시오.
2. 원격 데스크톱을 사용하여 Windows 인스턴스에 로그인합니다.
3. 실행 대화 상자에 diskmgmt.msc를 입력하고 Enter 키를 누릅니다. 디스크 관리 유ти리티가 열립니다.



4. 디스크 관리(Disk Management) 메뉴에서 작업, 디스크 다시 스캔(Rescan Disks)을 선택합니다.
5. 확장된 드라이브를 오른쪽 클릭하여 컨텍스트 메뉴를 열고 볼륨 확장(Extend Volume)을 선택합니다.

Note

할당되지 않은 공간이 확장할 드라이브의 오른쪽에 인접해야 합니다. 확장 볼륨이 회색으로 표시되면 할당되지 않은 공간이 드라이브 옆에 있지 않을 수 있습니다.



6. Extend Volume(볼륨 확장) 마법사에서 다음을 선택합니다. MB 단위로 공간 크기 선택(Select the amount of space in MB)에 볼륨 확장에 적용할 메가바이트 수를 입력합니다. 일반적으로 최대 사용 가능한 공간을 설정합니다. 선택(Selected) 아래에 강조된 텍스트는 추가되는 공간의 양이며, 볼륨의 최종 크기가 아닙니다. 마법사를 완료합니다.

Extend Volume Wizard

Select Disks

You can use space on one or more disks to extend the volume.

You can only extend the volume to the available space shown below because your disk cannot be converted to dynamic or the volume being extended is a boot or system volume.

Available:

Add >

< Remove

< Remove All

Selected:

Disk 0	71679 MB
--------	----------

Total volume size in megabytes (MB):

102397

Maximum available space in MB:

71679

Select the amount of space in MB:

71679



< Back

Next >

Cancel

7. AWS NVMe 드라이버가 없는 인스턴스에서 NVMe 볼륨의 크기를 늘릴 경우 Windows에서 새 볼륨의 크기를 확인할 수 있도록 인스턴스를 재부팅해야 합니다. AWS NVMe 드라이버 설치에 대한 자세한 내용은 [Windows 인스턴스의 AWS NVMe 드라이버 \(p. 539\)](#) 단원을 참조하십시오.

Amazon EBS 암호화

EC2 인스턴스와 연결된 EBS 리소스를 위한 간단한 암호화 솔루션으로 Amazon EBS 암호화를 사용할 수 있습니다. Amazon EBS 암호화를 사용하면 고유 키 관리 인프라를 빌드, 유지 관리 및 보호할 필요가 없습니다. Amazon EBS 암호화는 암호화된 볼륨 및 스냅샷을 생성할 때 AWS Key Management Service(AWS KMS) 고객 마스터 키(CMK)를 사용합니다.

암호화 작업은 EC2 인스턴스를 호스팅하는 서버에서 진행되며, 인스턴스와 인스턴스에 연결된 EBS 스토리지 간 유 휴 데이터 및 전송 중 데이터의 보안을 모두 보장합니다.

인스턴스에는 암호화된 볼륨과 암호화되지 않은 볼륨을 동시에 연결할 수 있습니다.

목차

- [EBS의 암호화 방식 \(p. 1034\)](#)
- [요구 사항 \(p. 1034\)](#)

- EBS 암호화에 대한 기본 키 (p. 1035)
- 암호화 기본 제공 (p. 1036)
- EBS 리소스 암호화 (p. 1037)
- 암호화 시나리오 (p. 1038)
- API 및 CLI를 사용한 암호화 기본값 설정 (p. 1043)

EBS의 암호화 방식

EC2 인스턴스의 부팅 및 데이터 볼륨을 모두 암호화할 수 있습니다. 암호화된 EBS 볼륨을 만들고 지원되는 인스턴스 유형에 이 볼륨을 연결하면 다음 유형의 데이터가 암호화됩니다.

- 볼륨 내부에 있는 데이터
- 볼륨과 인스턴스 사이에서 이동하는 모든 데이터
- 볼륨에서 생성된 모든 스냅샷
- 그런 스냅샷에서 생성된 모든 볼륨

EBS는 산업 표준 AES-256 알고리즘을 사용하여 데이터 키로 볼륨을 암호화합니다. 데이터 키는 암호화된 데이터와 함께 디스크에 저장되지만 EBS가 고객의 CMK로 암호화하기 전에는 저장되지 않습니다. 데이터 키가 디스크에서 일반 텍스트로 보이는 일은 결코 없습니다. 볼륨의 스냅샷은 동일한 데이터 키를 공유하며 그런 스냅샷에서 생성한 후속 볼륨도 데이터 키를 공유합니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [데이터 키](#)를 참조하십시오.

Amazon EBS는 AWS KMS와 함께 작동하여 다음과 같이 EBS 볼륨을 암호화 및 해독합니다.

1. Amazon EBS는 볼륨 암호화용으로 선택된 CMK를 지정하여 AWS KMS에 [GenerateDataKeyWithoutPlaintext](#) 요청을 보냅니다.
2. AWS KMS는 새 데이터 키를 생성하고 볼륨 암호화용으로 선택된 CMK로 암호화한 후 볼륨 메타데이터와 함께 저장할 암호화된 데이터 키를 Amazon EBS로 보냅니다.
3. 암호화된 볼륨을 인스턴스에 연결하면 Amazon EC2는 암호화된 데이터 키를 지정하여 AWS KMS에 [Decrypt](#) 요청을 보냅니다.
4. Amazon EBS는 데이터 키를 해독할 수 있도록 AWS KMS에 [CreateGrant](#) 요청을 보냅니다.
5. AWS KMS는 암호화된 데이터 키를 해독하고, 해독한 데이터 키를 Amazon EC2로 보냅니다.
6. Amazon EC2는 하이퍼바이저 메모리에서 일반 텍스트 데이터 키를 사용하여 디스크 I/O를 볼륨으로 암호화합니다. 볼륨이 인스턴스에 연결되어 있는 동안에는 일반 텍스트 형태의 데이터 키가 메모리에 유지됩니다.

자세한 내용은 AWS Key Management Service Developer Guide에서 [Amazon Elastic Block Store\(Amazon EBS\)가 AWS KMS를 사용하는 방식](#) 및 [Amazon EC2 예제 2](#)를 참조하십시오.

요구 사항

시작하기 전에 다음 요구 사항을 충족하는지 확인하십시오.

지원되는 볼륨 유형

암호화는 모든 EBS 볼륨 유형에서 지원됩니다. 암호화된 볼륨에서는 지역 시간에 대한 영향을 최소화한 채 암호화되지 않은 볼륨과 동일한 IOPS 성능을 기대할 수 있습니다. 암호화되지 않은 볼륨에 액세스하는 것과 동일한 방법으로 암호화된 볼륨에 액세스할 수 있습니다. 암호화 및 암호 해독은 중단 없이 처리되므로 사용자나 사용자의 애플리케이션에서 별도로 조치할 부분은 없습니다.

지원되는 인스턴스 유형

Amazon EBS 암호화는 모든 [현재 세대](#) (p. 115) 인스턴스 유형과 C3, cr1.8xlarge, G2, I2, M3, R3와 같은 [이전 세대](#) (p. 117) 인스턴스 유형에서 사용할 수 있습니다.

IAM 사용자의 권한

CMK를 EBS 암호화의 기본 키로 구성하면, 기본 키 정책은 필수 KMS 작업에 대한 액세스 권한이 있는 IAM 사용자가 이 키를 사용하여 EBS 리소스를 암호화하거나 해독하도록 허용합니다. EBS 암호화를 사용하기 위해 다음 작업을 호출할 수 있는 권한을 IAM 사용자에게 부여해야 합니다.

- kms:CreateGrant
- kms:Decrypt
- kms:DescribeKey
- kms:GenerateDataKeyWithoutPlainText
- kms:ReEncrypt

최소 권한의 주체를 따르려면 kms:CreateGrant에 대한 모든 액세스 권한을 허용하지 마십시오. 대신에 다음 예제에 표시된 대로 AWS 서비스에서 사용자를 대신하여 승인을 생성하는 경우에만 사용자는 CMK에 대해 승인을 생성할 수 있습니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "kms>CreateGrant",  
            "Resource": [  
                "arn:aws:kms:us-east-2:123456789012:key/abcd1234-a123-456d-a12b-a123b4cd56ef"  
            ],  
            "Condition": {  
                "Bool": {  
                    "kms:GrantIsForAWSResource": true  
                }  
            }  
        }  
    ]  
}
```

자세한 내용은 AWS Key Management Service Developer Guide의 기본 키 정책 단원에 있는 [AWS 계정에 대한 액세스 허용 및 IAM 정책 활성화](#)를 참조하십시오.

EBS 암호화에 대한 기본 키

Amazon EBS는 사용자가 AWS 리소스를 저장하는 각 리전에 고유한 AWS 관리형 CMK를 자동으로 생성합니다. 이 키의 별칭은 alias/aws/ebs입니다. 기본적으로 Amazon EBS는 암호화에 이 키를 사용합니다. 또는 생성한 대칭 고객 관리형 CMK를 EBS 암호화의 기본 키로 지정할 수 있습니다. 자체 CMK를 사용하여 키 생성, 교체 및 비활성화 기능을 비롯한 다양한 작업을 수행할 수 있습니다.

Important

Amazon EBS에서는 비대칭 CMK가 지원되지 않습니다. 자세한 내용은 AWS Key Management Service 개발자 안내서의 [대칭 및 비대칭 키 사용](#)을 참조하십시오.

새로운 콘솔

리전에 대한 EBS 암호화의 기본 키를 구성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 해당 리전을 선택합니다.
3. 탐색 창에서 EC2 대시보드를 선택합니다.

4. 페이지의 오른쪽 위 모서리에서 Account Attributes(계정 속성), EBS encryption(EBS 암호화)을 선택합니다.
5. 관리를 선택합니다.
6. Default encryption key(기본 암호화 키)에서 고객 관리형 대칭 CMK를 선택합니다.
7. Update EBS encryption(EBS 암호화 업데이트)를 선택합니다.

기존 콘솔

리전에 대한 EBS 암호화의 기본 키를 구성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 해당 리전을 선택합니다.
3. 탐색 창에서 EC2 대시보드를 선택합니다.
4. 페이지의 오른쪽 위 모서리에서 계정 속성, 설정을 선택합니다.
5. Change the default key(기본 키 변경)를 선택한 후 사용 가능한 키를 선택합니다.
6. Save settings(설정 저장)를 선택합니다.

암호화 기본 제공

사용자의 AWS 계정을 구성하여 생성하는 새 EBS 볼륨 및 스냅샷 사본의 암호화를 적용할 수 있습니다. 예를 들어 Amazon EBS는 인스턴스 시작 시 생성된 EBS 볼륨과 암호화되지 않은 스냅샷에서 복사하는 스냅샷을 암호화합니다. 암호화되지 않은 EBS 리소스에서 암호화된 리소스로의 이전 예제는 [암호화되지 않은 리소스의 암호화 \(p. 1037\)](#) 단원을 참조하십시오.

기본적으로 암호화는 기존 EBS 볼륨이나 스냅샷에 영향을 미치지 않습니다.

고려 사항

- 암호화 기본 제공은 리전별 설정입니다. 특정 기능에 대해 이 기능을 활성화하면 해당 리전의 개별 볼륨 또는 스냅샷에 대해 비활성화할 수 없습니다.
- 암호화 기본 제공을 활성화하면 인스턴스 유형이 EBS 암호화를 지원할 때만 인스턴스를 시작할 수 있습니다. 자세한 내용은 [지원되는 인스턴스 유형 \(p. 1034\)](#) 단원을 참조하십시오.
- AWS Server Migration Service(SMS)을 사용하여 서버를 마이그레이션할 때 기본적으로 암호화를 켜지 마십시오. 기본적으로 암호화가 이미 설정되어 있고 델타 복제 오류가 발생하는 경우 이 기능을 해제하십시오. 대신 복제 작업을 생성할 때 AMI 암호화를 활성화하십시오.

새로운 콘솔

리전에서 암호화를 기본적으로 활성화하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 해당 리전을 선택합니다.
3. 탐색 창에서 EC2 대시보드를 선택합니다.
4. 페이지의 오른쪽 위 모서리에서 Account Attributes(계정 속성), EBS encryption(EBS 암호화)을 선택합니다.
5. 관리를 선택합니다.
6. Enable(활성화)을 선택합니다. 자동으로 생성된 별칭(alias/aws/ebs)을 가진 AWS 관리형 CMK를 기본 암호화 키로 유지하거나 고객 관리형 대칭 CMK를 선택합니다.
7. Update EBS encryption(EBS 암호화 업데이트)를 선택합니다.

기존 콘솔

리전에서 암호화를 기본적으로 활성화하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 해당 리전을 선택합니다.
3. 탐색 창에서 EC2 대시보드를 선택합니다.
4. 페이지의 오른쪽 위 모서리에서 계정 속성, 설정을 선택합니다.
5. EBS 스토리지 아래에서 Always encrypt new EBS volumes(항상 새 EBS 볼륨 암호화)를 선택합니다.
6. Save settings(설정 저장)를 선택합니다.

기존 스냅샷이나 암호화된 볼륨과 연동되어 있는 CMK는 변경할 수 없습니다. 하지만 스냅샷 복사 작업 중에 다른 CMK와 연동시킬 수는 있습니다. 복사된 스냅샷은 새로운 CMK로 암호화됩니다.

EBS 리소스 암호화

[암호화를 기본으로 \(p. 1036\)](#) 사용하여 암호화를 활성화하거나 암호화하려는 볼륨을 생성할 때 암호화를 활성화하여 EBS 볼륨을 암호화합니다.

볼륨 암호화 시 볼륨을 암호화하는 데 사용할 대칭 CMK를 지정할 수 있습니다. CMK를 지정하지 않은 경우 암호화에 사용되는 키는 소스 스냅샷 및 해당 소유권의 암호화 상태에 따라 달라집니다. 자세한 내용은 [암호화 결과표 \(p. 1042\)](#)를 참조하십시오.

기존 스냅샷이나 볼륨과 연동되어 있는 CMK는 변경할 수 없습니다. 하지만 스냅샷 복사 작업 중에 다른 CMK와 연동시킬 수는 있습니다. 복사된 스냅샷은 새로운 CMK로 암호화됩니다.

비어 있는 볼륨 생성 시 암호화

비어 있는 새 EBS 볼륨을 생성하는 경우 특정 볼륨 생성 작업에 대한 암호화를 활성화하여 암호화할 수 있습니다. EBS 암호화를 기본으로 활성화한 경우 볼륨은 자동으로 암호화됩니다. 기본적으로 볼륨은 EBS 암호화에 대한 기본 키에 암호화됩니다. 또는 특정 볼륨 생성 작업에 대해 다른 대칭 CMK를 지정할 수 있습니다. 볼륨은 최초로 사용 가능한 시점까지 암호화되므로 데이터가 항상 안전한 상태를 유지합니다. 자세한 절차는 [Amazon EBS 볼륨 생성 \(p. 948\)](#) 단원을 참조하십시오.

기본적으로 볼륨을 생성할 때 선택한 CMK가 볼륨에서 생성한 스냅샷과 암호화된 스냅샷에서 복원한 볼륨을 암호화합니다. 암호화된 볼륨 또는 스냅샷으로부터 암호화를 제거할 수 없습니다. 즉, 암호화된 스냅샷 또는 암호화된 스냅샷의 사본에서 복원된 볼륨은 항상 암호화됩니다.

암호화된 볼륨의 퍼블릭 스냅샷은 지원하지 않지만 암호화된 스냅샷을 특정 계정과 공유할 수는 있습니다. 자세한 지침은 [Amazon EBS 스냅샷 공유 \(p. 990\)](#) 단원을 참조하십시오.

암호화되지 않은 리소스의 암호화

암호화되지 않은 기존 볼륨 또는 스냅샷을 암호화는 직접적인 방법은 없지만 볼륨 또는 스냅샷을 생성하여 암호화할 수 있습니다. 암호화를 기본적으로 활성화한 경우 Amazon EBS는 EBS 암호화에 대한 사용자의 기본 키를 사용하여, 결과로 얻은 새로운 볼륨 또는 스냅샷을 암호화합니다. 암호화를 기본적으로 활성화하지 않은 경우라도 개별 볼륨 또는 스냅샷을 생성할 때 암호화를 활성화할 수 있습니다. 기본적으로 또는 개별 생성 작업에서 암호화를 활성화하는 경우 사용자는 EBS 암호화에 대한 기본 키를 재정의하고 대칭 고객 관리형 CMK를 선택할 수 있습니다. 자세한 내용은 [Amazon EBS 볼륨 생성 \(p. 948\)](#) 및 [Amazon EBS 스냅샷 복사 \(p. 985\)](#) 단원을 참조하십시오.

스냅샷 사본을 고객 관리형 CMK로 암호화하려면 [암호화되지 않은 스냅샷\(활성화되지 않은 암호화 기본 제공\) 복사 \(p. 1039\)](#)에 표시된 대로 암호화를 활성화하고 키를 명시해야 합니다.

Important

Amazon EBS에서는 비대칭 CMK가 지원되지 않습니다. 자세한 내용은 AWS Key Management Service 개발자 안내서의 [대칭 및 비대칭 키 사용](#)을 참조하십시오.

EBS 지원 AMI에서 인스턴스를 시작할 때 새 암호화 상태를 적용할 수도 있습니다. 이는 EBS 지원 AMI에, 설정된 대로 암호화할 수 있는 EBS 볼륨의 스냅샷이 포함되기 때문입니다. 자세한 내용은 [EBS 지원 AMI를 통한 암호화 사용 \(p. 98\)](#) 단원을 참조하십시오.

암호화 시나리오

암호화된 EBS 리소스를 생성하면 볼륨 생성 파라미터에서 다른 고객 관리형 CMK를 지정하거나 AMI 또는 인스턴스에 대한 블록 장치 매핑을 지정하지 않는 한, 사용자 계정의 EBS 암호화에 대한 기본 키에 의해 암호화됩니다. 자세한 내용은 [EBS 암호화에 대한 기본 키 \(p. 1035\)](#) 단원을 참조하십시오.

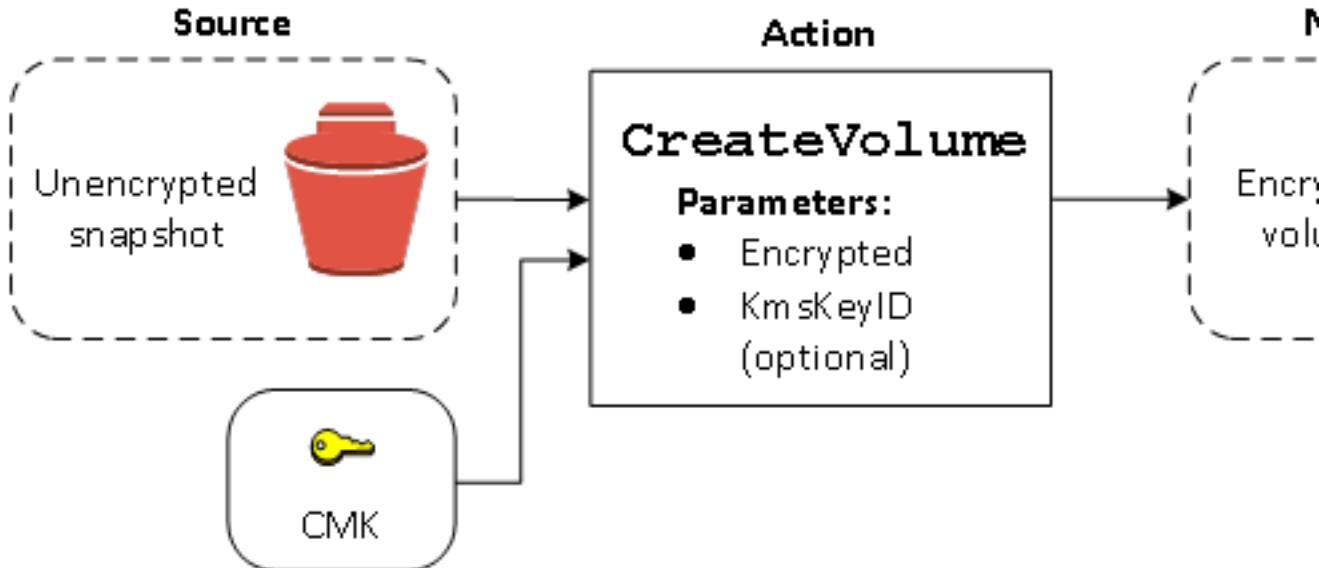
다음 예제에서는 볼륨 및 스냅샷의 암호화 상태를 관리할 수 있는 방법을 보여줍니다. 암호화 사례의 전체 목록은 [암호화 결과표 \(p. 1042\)](#)를 참조하십시오.

예제:

- [암호화되지 않은 볼륨\(활성화되지 않은 암호화 기본 제공\) 복원 \(p. 1038\)](#)
- [암호화되지 않은 볼륨\(활성화된 암호화 기본 제공\) 복원 \(p. 1039\)](#)
- [암호화되지 않은 스냅샷\(활성화되지 않은 암호화 기본 제공\) 복사 \(p. 1039\)](#)
- [암호화되지 않은 스냅샷\(활성화된 암호화 기본 제공\) 복사 \(p. 1040\)](#)
- [암호화된 볼륨의 재암호화 \(p. 1040\)](#)
- [암호화된 스냅샷의 재암호화 \(p. 1041\)](#)
- [암호화된 볼륨과 암호화되지 않은 볼륨 간 데이터 마이그레이션 \(p. 1042\)](#)
- [암호화 결과 \(p. 1042\)](#)

암호화되지 않은 볼륨(활성화되지 않은 암호화 기본 제공) 복원

암호화 기본 제공을 활성화하지 않은 상태에서는 암호화되지 않은 스냅샷에서 복원한 볼륨이 기본적으로 암호화되지 않습니다. 그러나 Encrypted 파라미터와, 선택적으로, KmsKeyId 파라미터를 설정하여, 결과로 얻은 볼륨을 암호화할 수 있습니다. 다음 디어그램에서 프로세스를 보여 줍니다.

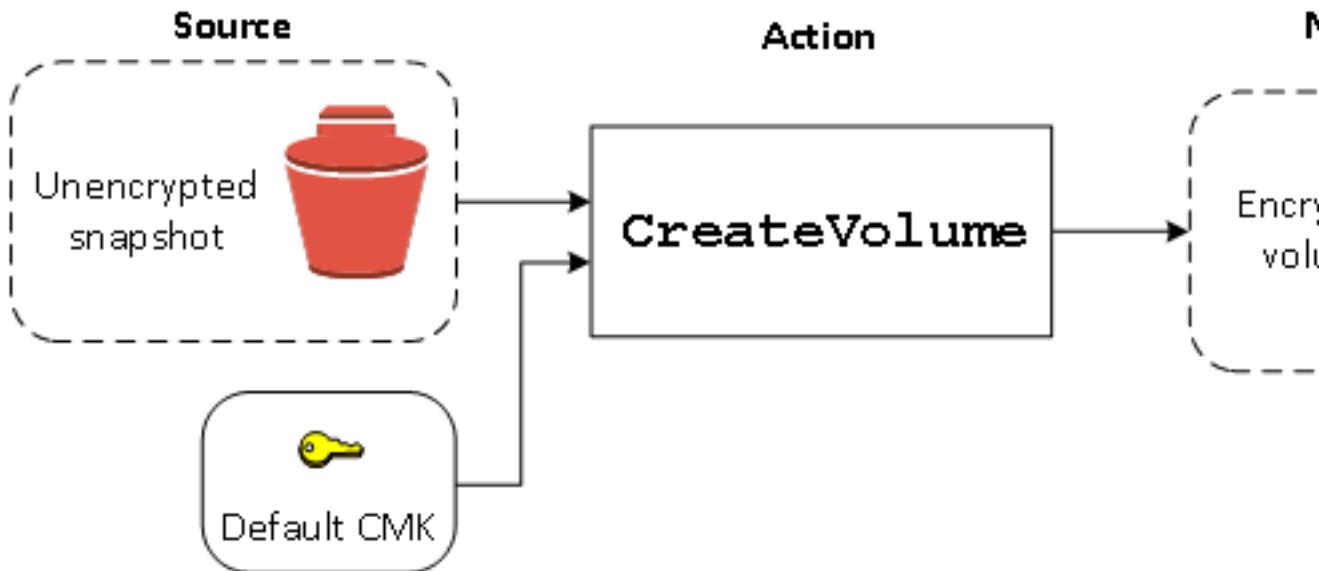


KmsKeyId 파라미터를 그대로 놓아두면 결과로 얻은 볼륨이 EBS 암호화에 대한 기본 키를 사용하여 암호화됩니다. 키 ID를 지정하여 해당 볼륨을 다른 CMK로 암호화해야 합니다.

자세한 내용은 [스냅샷에서 볼륨 생성 \(p. 949\)](#) 단원을 참조하십시오.

암호화되지 않은 볼륨(활성화된 암호화 기본 제공) 복원

암호화를 기본적으로 활성화한 경우 암호화되지 않은 스냅샷에서 복원된 볼륨에 대한 암호화가 필수이며 사용자의 기본 CMK를 사용하는 데 암호화 파라미터는 필요하지 않습니다. 다음 다이어그램은 이러한 간단한 기본 사례를 보여 줍니다.

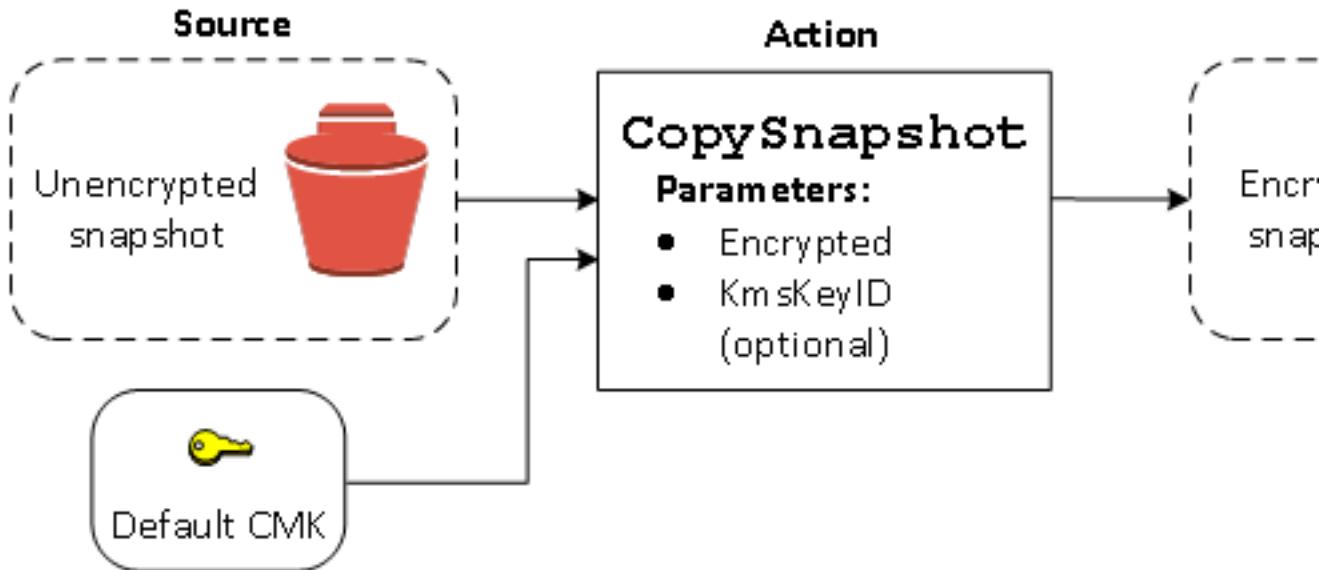


복원된 볼륨을 대칭 고객 관리형 CMK로 암호화하려면 [암호화되지 않은 볼륨\(활성화된 암호화 기본 제공\) 복원 \(p. 1038\)](#)에서처럼 Encrypted 및 KmsKeyId 파라미터를 모두 입력해야 합니다.

암호화되지 않은 스냅샷(활성화되지 않은 암호화 기본 제공) 복사

암호화 기본 제공을 활성화하지 않은 상태에서는 암호화되지 않은 스냅샷의 사본이 기본적으로 암호화되지 않습니다. 그러나 Encrypted 파라미터와, 선택적으로, KmsKeyId 파라미터를 설정하여, 결과로 얻은 볼륨을 암호화할 수 있습니다. KmsKeyId를 생략하면 결과로 얻은 스냅샷이 사용자의 기본 CMK로 암호화됩니다. 키 ID를 지정하여 해당 볼륨을 다른 대칭 CMK로 암호화해야 합니다.

다음 다이어그램에서 프로세스를 보여 줍니다.



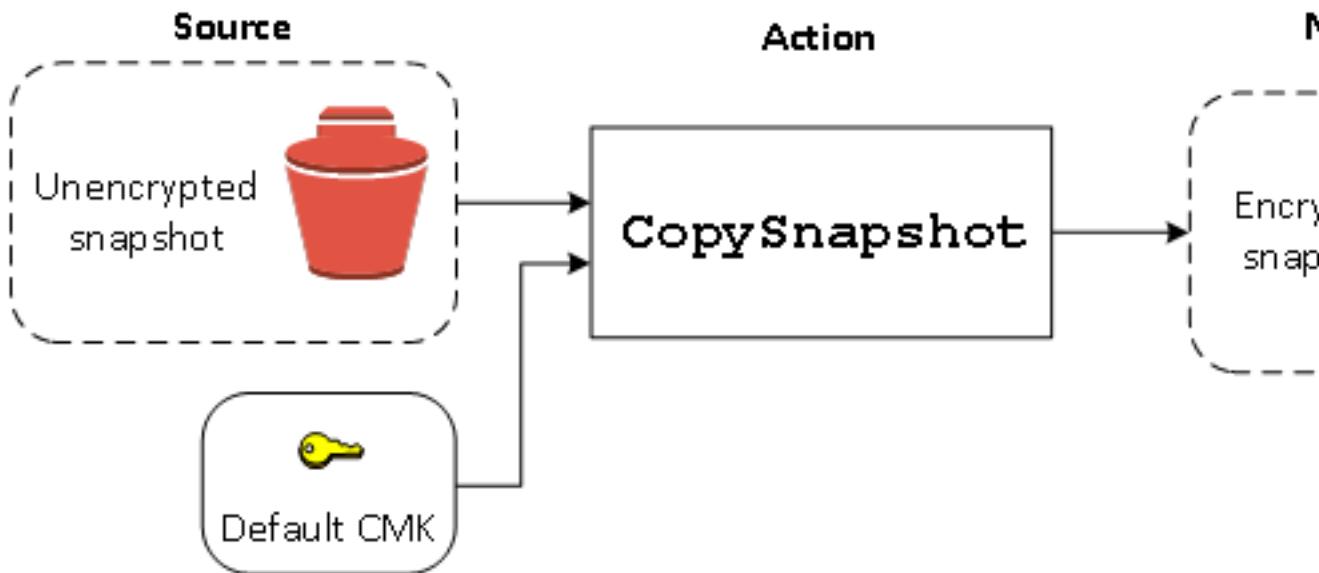
Note

새 CMK로 스냅샷을 복사 및 암호화하면 항상 전체(증분 아님) 복사가 생성되고 추가 지연 및 스토리지 비용이 발생합니다.

암호화되지 않은 스냅샷을 암호화된 스냅샷에 복사한 다음 암호화된 스냅샷에서 볼륨을 생성하여 EBS 볼륨을 암호화할 수 있습니다. 자세한 내용은 [Amazon EBS 스냅샷 복사 \(p. 985\)](#) 단원을 참조하십시오.

[암호화되지 않은 스냅샷\(활성화된 암호화 기본 제공\) 복사](#)

암호화를 기본적으로 활성화한 경우 암호화되지 않은 스냅샷의 복사가 필수이며 사용자의 기본 CMK가 사용된다면 암호화 파라미터는 필요하지 않습니다. 다음 다이어그램에서 기본 사례를 보여 줍니다.

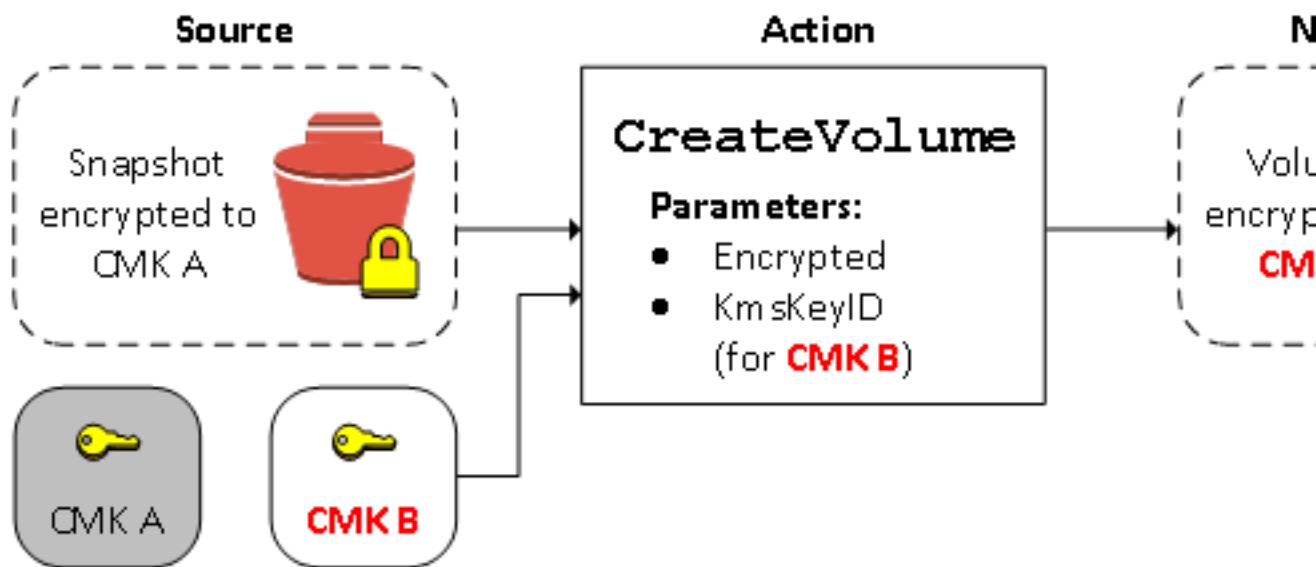


Note

새 CMK로 스냅샷을 복사 및 암호화하면 항상 전체(증분 아님) 복사가 생성되고 추가 지연 및 스토리지 비용이 발생합니다.

[암호화된 볼륨의 재암호화](#)

암호화된 스냅샷에서 `CreateVolume` 작업을 수행할 때 다른 CMK를 통해 재암호화하는 옵션이 있습니다. 다음 다이어그램에서 프로세스를 보여 줍니다. 이 예에서는 CMK A와 CMK B, 2개의 CMK가 있습니다. 원본 스냅샷은 CMK A로 암호화됩니다. 볼륨을 생성하는 동안 파라미터로 지정된 CMK B의 키 ID를 통해 원본 데이터가 자동으로 암호 해독된 다음 CMK B로 재암호화됩니다.



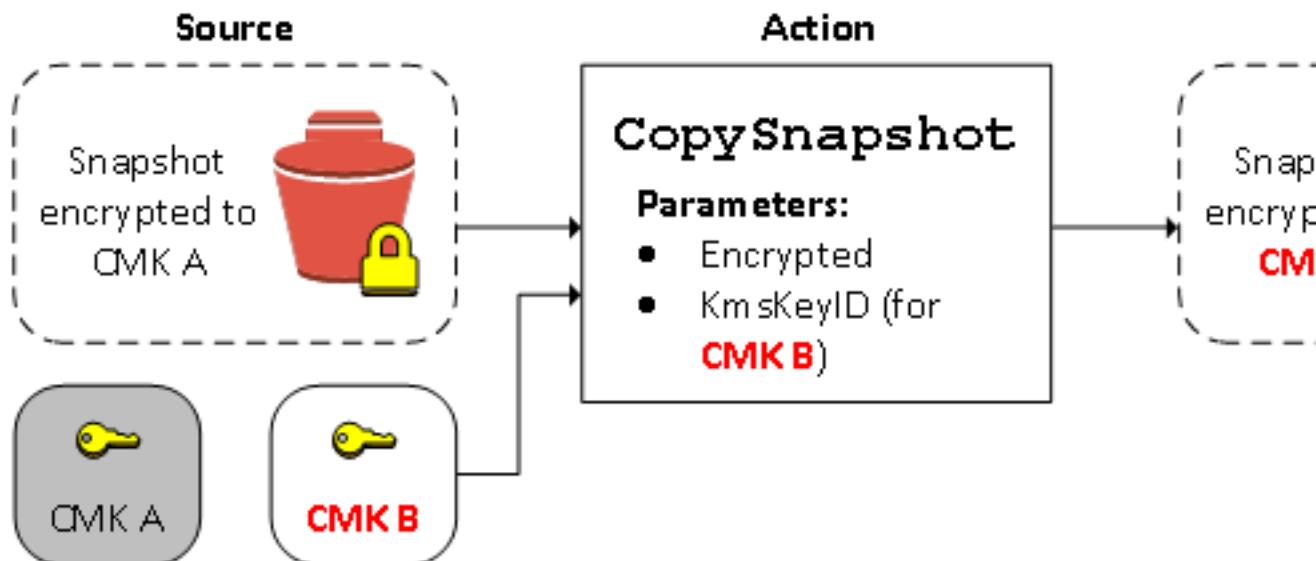
Note

새 CMK로 스냅샷을 복사 및 암호화하면 항상 전체(종분 아님) 복사가 생성되고 추가 지연 및 스토리지 비용이 발생합니다.

자세한 내용은 [스냅샷에서 볼륨 생성 \(p. 949\)](#) 단원을 참조하십시오.

암호화된 스냅샷의 재암호화

복사 중에 스냅샷을 암호화하는 기능을 사용하여 자신이 소유하고 있는 이미 암호화된 스냅샷에 새 대칭 CMK를 적용할 수 있습니다. 새 CMK를 사용하여 결과 복사본에서 복원된 볼륨에만 액세스할 수 있습니다. 다음 다이어그램에서 프로세스를 보여 줍니다. 이 예에서는 CMK A와 CMK B, 2개의 CMK가 있습니다. 원본 스냅샷은 CMK A로 암호화됩니다. 복사하는 동안 파라미터로 지정된 CMK B의 키 ID를 통해 원본 데이터가 자동으로 CMK B로 재암호화됩니다.



Note

새 CMK로 스냅샷을 복사 및 암호화하면 항상 전체(증분 아님) 복사가 생성되고 추가 지연 및 스토리지 비용이 발생합니다.

관련된 시나리오에서 자신과 공유된 스냅샷의 복사본에 새 암호화 파라미터를 적용하도록 선택할 수도 있습니다. 기본적으로 이 복사본은 스냅샷의 소유자가 공유하는 CMK로 암호화됩니다. 하지만 복사 프로세스 중에 자신이 관리하는 다른 CMK를 사용하여 공유 스냅샷의 복사본을 만드는 게 좋습니다. 이를 통해 원래 CMK가 손상되거나 소유자가 어떤 이유로든 CMK를 취소하는 경우 볼륨에 대한 액세스 권한을 보호할 수 있습니다. 자세한 내용은 [암호화 및 스냅샷 복사 \(p. 986\)](#) 단원을 참조하십시오.

암호화된 볼륨과 암호화되지 않은 볼륨 간 데이터 마이그레이션

암호화된 볼륨과 암호화되지 않은 볼륨에 모두 액세스할 수 있는 경우, 둘 사이에서 자유롭게 데이터를 전송할 수 있습니다. EC2는 암호화 및 복호화 작업을 투명하게 수행합니다.

예를 들어 robocopy 명령을 사용하여 데이터를 복사합니다. 다음 명령에서 소스 데이터는 D:\에 있고 대상 볼륨은 E:\에 마운트되어 있습니다.

```
PS C:\> robocopy D:\sourcefolder E:\destinationfolder /e /copyall /eta
```

숨겨진 폴더로 인한 잠재적 문제를 방지하기 위해 전체 볼륨을 복사하는 대신 폴더를 사용하는 것이 좋습니다.

암호화 결과

다음 표에서는 가능한 각 설정 조합에 대한 암호화 결과를 설명합니다.

암호화를 활성화합니까?	암호화를 기본 설정합니까?	볼륨의 소스	기본값(CMK가 지정되지 않음)	사용자 지정(CMK가 지정됨)
아니요	아니요	새(빈) 볼륨	암호화되지 않음	해당 사항 없음
아니요	아니요	암호화되지 않은 소유 스냅샷	암호화되지 않음	
아니요	아니요	암호화된 소유 스냅샷	동일한 키로 암호화됨	
아니요	아니요	암호화되지 않은 공유 스냅샷	암호화되지 않음	
아니요	아니요	암호화된 공유 스냅샷	기본 CMK로 암호화됨*	
예	아니요	새 볼륨	기본 CMK로 암호화됨	지정된 CMK로 암호화됨**
예	아니요	암호화되지 않은 소유 스냅샷	기본 CMK로 암호화됨	
예	아니요	암호화된 소유 스냅샷	동일한 키로 암호화됨	
예	아니요	암호화되지 않은 공유 스냅샷	기본 CMK로 암호화됨	
예	아니요	암호화된 공유 스냅샷	기본 CMK로 암호화됨	

암호화를 활성화합니까?	암호화를 기본 설정합니까?	볼륨의 소스	기본값(CMK가 지정되지 않음)	사용자 지정(CMK가 지정됨)
아니요	예	새(빈) 볼륨	기본 CMK로 암호화 됨	해당 사항 없음
아니요	예	암호화되지 않은 소유 스냅샷	기본 CMK로 암호화 됨	
아니요	예	암호화된 소유 스냅샷	동일한 키로 암호화 됨	
아니요	예	암호화되지 않은 공유 스냅샷	기본 CMK로 암호화 됨	
아니요	예	암호화된 공유 스냅샷	기본 CMK로 암호화 됨	
예	예	새 볼륨	기본 CMK로 암호화 됨	지정된 CMK로 암호화 됨
예	예	암호화되지 않은 소유 스냅샷	기본 CMK로 암호화 됨	
예	예	암호화된 소유 스냅샷	동일한 키로 암호화 됨	
예	예	암호화되지 않은 공유 스냅샷	기본 CMK로 암호화 됨	
예	예	암호화된 공유 스냅샷	기본 CMK로 암호화 됨	

* 이는 AWS 계정 및 리전의 EBS 암호화에 사용되는 기본 CMK입니다. 기본적으로 EBS에 대한 고유한 AWS 관리형 CMK이거나 고객 관리형 CMK를 지정할 수 있습니다. 자세한 내용은 [EBS 암호화에 대한 기본 키 \(p. 1035\)](#) 단원을 참조하십시오.

** 이는 실행 시 볼륨용으로 지정된 고객 관리형 CMK입니다. 이 CMK는 AWS 계정 및 리전에 대한 기본 CMK 대신 사용됩니다.

API 및 CLI를 사용한 암호화 기본값 설정

다음 API 작업 및 CLI 명령을 사용하여 암호화 기본 제공과 기본 고객 마스터 키(CMK)를 관리할 수 있습니다.

API 작업	CLI 명령	설명
DisableEbsEncryptionByDefault	<code>disable-ebs-encryption-by-default</code>	암호화를 기본적으로 비활성화합니다.
EnableEbsEncryptionByDefault	<code>enable-ebs-encryption-by-default</code>	암호화를 기본적으로 활성화합니다.
GetEbsDefaultKmsKeyId	<code>get-ebs-default-kms-key-id</code>	기본 CMK를 설명합니다.
GetEbsEncryptionByDefault	<code>get-ebs-encryption-by-default</code>	암호화 기본 제공을 활성화할지 여부를 나타냅니다.

API 작업	CLI 명령	설명
ModifyEbsDefaultKmsKeyId	modify-ebs-default-kms-key-id	EBS 볼륨을 암호화하는데 사용되는 기본 CMK를 변경합니다.
ResetEbsDefaultKmsKeyId	reset-ebs-default-kms-key-id	AWS 관리형 기본 CMK를 EBS 볼륨을 암호화하는데 사용되는 기본 CMK로 재설정합니다.

Amazon EBS 빠른 스냅샷 복원

Amazon EBS 빠른 스냅샷 복원을 사용하면 생성 시 완전히 초기화된 스냅샷에서 볼륨을 생성할 수 있습니다. 이렇게 하면 처음 액세스할 때 볼록에 대한 I/O 작업 지연 시간이 없어집니다. 빠른 스냅샷 복원을 사용하여 생성된 볼륨은 프로비저닝된 모든 성능을 즉시 제공합니다.

시작하려면 특정 가용 영역의 특정 스냅샷에 대해 빠른 스냅샷 복원을 활성화합니다. 각 스냅샷 및 가용 영역에 있는 하나의 빠른 스냅샷 복원을 나타냅니다. 활성화된 가용 영역 중 하나에 있는 이러한 스냅샷 중 하나에서 볼륨을 생성하면 빠른 스냅샷 복원을 사용하여 볼륨이 복원됩니다.

본인이 소유한 스냅샷과 본인에게 공유된 퍼블릭 및 프라이빗 스냅샷에 대해 빠른 스냅샷 복원을 활성화할 수 있습니다.

목차

- [빠른 스냅샷 복원 할당량 \(p. 1044\)](#)
- [빠른 스냅샷 복원 상태 \(p. 1044\)](#)
- [볼륨 생성 크레딧 \(p. 1045\)](#)
- [빠른 스냅샷 복원 관리 \(p. 1045\)](#)
- [빠른 스냅샷 복원이 활성화된 스냅샷 보기 \(p. 1046\)](#)
- [빠른 스냅샷 복원을 사용하여 복원된 볼륨 보기 \(p. 1047\)](#)
- [빠른 스냅샷 복원 모니터링 \(p. 1047\)](#)
- [요금 및 결제 \(p. 1047\)](#)

빠른 스냅샷 복원 할당량

리전별로 빠른 스냅샷 복원을 위해 최대 50개의 스냅샷을 활성화할 수 있습니다. 할당량은 본인이 소유한 스냅샷과 본인에게 공유된 스냅샷에 적용됩니다. 본인에게 공유된 스냅샷에 대해 빠른 스냅샷 복원을 활성화하면 빠른 스냅샷 복원 할당량에 포함됩니다. 스냅샷 소유자의 빠른 스냅샷 복원 할당량에 포함되지 않습니다.

빠른 스냅샷 복원 상태

스냅샷에 대해 빠른 스냅샷 복원을 활성화하면 스냅샷이 다음 상태 중 하나일 수 있습니다.

- **enabling** — 빠른 스냅샷 복원 활성화를 요청했습니다.
- **optimizing** — 빠른 스냅샷 복원이 활성화되고 있습니다. 스냅샷을 최적화하려면 TiB당 60분이 소요됩니다.
- **enabled** — 빠른 스냅샷 복원이 활성화되었습니다.
- **disabling** — 빠른 스냅샷 복원을 비활성화하도록 요청했거나 빠른 스냅샷 복원을 활성화하는 요청이 실패했습니다.
- **disabled** — 빠른 스냅샷 복원이 비활성화되었습니다. 필요에 따라 빠른 스냅샷 복원을 다시 활성화할 수 있습니다.

볼륨 생성 크레딧

빠른 스냅샷 복원의 최대 성능 이점을 얻는 볼륨 수는 스냅샷에 대한 볼륨 생성 크레딧에 의해 결정됩니다. 가용 영역당 스냅샷별로 하나의 크레딧 버킷이 있습니다. 빠른 스냅샷 복원이 활성화된 스냅샷에서 생성하는 각 볼륨은 크레딧 버킷에서 하나의 크레딧을 사용합니다.

본인에게 공유된 스냅샷에 대해 빠른 스냅샷 복원을 활성화하면 계정의 공유 스냅샷에 대해 별도의 크레딧 버킷이 생성됩니다. 공유 스냅샷에서 볼륨을 생성하는 경우 크레딧은 크레딧 버킷에서 소비되고 스냅샷 소유자의 크레딧 버킷에서 소비되지 않습니다.

크레딧 버킷의 크기는 스냅샷에서 생성된 볼륨의 크기가 아니라 스냅샷의 크기에 따라 다릅니다. 각 스냅샷의 크레딧 버킷 크기는 다음과 같이 계산됩니다.

```
MAX (1, MIN (10, FLOOR(1024/snapshot_size_gib)))
```

크레딧을 사용하면 크레딧 버킷은 시간 경과에 따라 재충전됩니다. 각 크레딧 버킷의 재충전 비율은 다음과 같이 계산됩니다.

```
MIN (10, 1024/snapshot_size_gib)
```

예를 들어 크기가 100GiB인 스냅샷에 대해 빠른 스냅샷 복원을 활성화하는 경우 크레딧 버킷의 최대 크기는 10 크레딧이며 재충전 비율은 시간당 10 크레딧입니다. 크레딧 버킷이 가득차면 이 스냅샷에서 10개의 초기화된 볼륨을 동시에 생성할 수 있습니다.

Cloudwatch 지표를 사용하여 크레딧 버킷의 크기와 각 버킷에서 사용할 수 있는 크레딧의 수를 모니터링할 수 있습니다. 자세한 내용은 [빠른 스냅샷 복원 지표 \(p. 1080\)](#) 단원을 참조하십시오.

빠른 스냅샷 복원이 활성화된 스냅샷에서 볼륨을 생성한 후, [describe-volumes](#)를 사용하여 볼륨을 설명하고 출력에서 `fastRestored` 필드를 확인하여 볼륨이 빠른 스냅샷 복원을 사용하여 초기화된 볼륨으로 생성되었는지 여부를 확인할 수 있습니다.

빠른 스냅샷 복원 관리

스냅샷에 대해 빠른 스냅샷 복원은 기본적으로 비활성화되어 있습니다. 본인이 소유한 스냅샷과 본인에게 공유된 스냅샷에 대해 빠른 스냅샷 복원을 활성화하거나 비활성화할 수 있습니다. 스냅샷에 대해 빠른 스냅샷 복원을 활성화하거나 비활성화하면 변경 사항이 계정에만 적용됩니다.

Note

스냅샷에 대해 빠른 스냅샷 복원을 활성화하면 특정 가용 영역에서 빠른 스냅샷 복원이 활성화된 1 분마다 계정에 요금이 청구됩니다. 요금은 최소 1시간으로 비례 청구됩니다.

본인이 소유한 스냅샷을 삭제하면 계정의 해당 스냅샷에 대해 빠른 스냅샷 복원이 자동으로 비활성화됩니다. 본인에게 공유된 스냅샷에 대해 빠른 스냅샷 복원을 활성화하고 스냅샷 소유자가 해당 스냅샷을 삭제하거나 공유 해제하는 경우 계정의 공유 스냅샷에 대해 빠른 스냅샷 복원이 자동으로 비활성화됩니다.

본인에게 공유된 스냅샷에 대해 빠른 스냅샷 복원을 활성화하고 사용자 지정 CMK를 사용하여 암호화된 경우 스냅샷 소유자가 사용자 지정 CMK에 대한 액세스를 취소해도 스냅샷에 대해 빠른 스냅샷 복원이 자동으로 비활성화되지 않습니다. 해당 스냅샷에 대해 빠른 스냅샷 복원을 수동으로 비활성화해야 합니다.

본인이 소유한 스냅샷 또는 본인에게 공유된 스냅샷에 대해 빠른 스냅샷 복원을 활성화하거나 비활성화하려면 다음 절차를 따르십시오.

빠른 스냅샷 복원을 활성화하거나 비활성화하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 탐색 창에서 Snapshots(스냅샷)을 선택합니다.
3. 스냅샷을 선택합니다.
4. Actions(작업), Manage Fast Snapshot Restore(빠른 스냅샷 관리)를 선택합니다.
5. 가용 영역을 선택하거나 선택 취소한 다음 Save(저장)를 선택합니다.
6. 활성화된 경우 빠른 스냅샷 복원의 상태를 추적하려면 Description(설명) 탭의 Fast Snapshot Restore(빠른 스냅샷 복원)를 참조하십시오.

AWS CLI를 사용하여 빠른 스냅샷 복원을 관리하려면

- [enable-fast-snapshot-restores](#)
- [disable-fast-snapshot-restores](#)
- [describe-fast-snapshot-restores](#)

빠른 스냅샷 복원이 활성화된 스냅샷 보기

본인이 소유한 스냅샷 또는 본인에게 공유된 스냅샷에 대한 빠른 스냅샷 복원 상태를 보려면 다음 절차를 따르십시오.

콘솔을 사용하여 빠른 스냅샷 복원 상태를 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Snapshots(스냅샷)을 선택합니다.
3. 스냅샷을 선택합니다.
4. Description(설명) 탭에서 Fast Snapshot Restore(빠른 스냅샷 복원)를 확인합니다. 빠른 스냅샷 복원의 상태가 나타나 있습니다. 예를 들어 “2개의 가용 영역 최적화” 또는 “2개의 가용 영역 활성화” 상태를 표시할 수 있습니다.

AWS CLI를 사용하여 빠른 스냅샷 복원이 활성화된 스냅샷을 보려면

빠른 스냅샷 복원이 활성화된 스냅샷을 확인하려면 [describe-fast-snapshot-restores](#) 명령을 사용합니다.

```
aws ec2 describe-fast-snapshot-restores --filters Name=state,Values=enabled
```

다음은 예제 출력입니다.

```
{  
    "FastSnapshotRestores": [  
        {  
            "SnapshotId": "snap-0e946653493cb0447",  
            "AvailabilityZone": "us-east-2a",  
            "State": "enabled",  
            "StateTransitionReason": "Client.UserInitiated - Lifecycle state transition",  
            "OwnerId": "123456789012",  
            "EnablingTime": "2020-01-25T23:57:49.596Z",  
            "OptimizingTime": "2020-01-25T23:58:25.573Z",  
            "EnabledTime": "2020-01-25T23:59:29.852Z"  
        },  
        {  
            "SnapshotId": "snap-0e946653493cb0447",  
            "AvailabilityZone": "us-east-2b",  
            "State": "enabled",  
            "StateTransitionReason": "Client.UserInitiated - Lifecycle state transition",  
            "OwnerId": "123456789012",  
            "EnablingTime": "2020-01-25T23:57:49.596Z",  
            "OptimizingTime": "2020-01-25T23:58:25.573Z",  
            "EnabledTime": "2020-01-25T23:59:29.852Z"  
        }  
    ]  
}
```

```
        "OptimizingTime": "2020-01-25T23:58:25.573Z",
        "EnabledTime": "2020-01-25T23:59:29.852Z"
    }
}
```

빠른 스냅샷 복원을 사용하여 복원된 볼륨 보기

볼륨의 가용 영역에서 빠른 스냅샷 복원이 활성화된 스냅샷을 기반으로 볼륨을 생성하면 빠른 스냅샷 복원을 사용하여 볼륨이 복원됩니다.

빠른 스냅샷 복원이 활성화된 스냅샷을 기반으로 생성된 볼륨을 확인하려면 [describe-volumes](#) 명령을 사용합니다.

```
aws ec2 describe-volumes --filters Name=fast-restored,Values=true
```

다음은 예제 출력입니다.

```
{
    "Volumes": [
        {
            "Attachments": [],
            "AvailabilityZone": "us-east-2a",
            "CreateTime": "2020-01-26T00:34:11.093Z",
            "Encrypted": true,
            "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/8c5b2c63-b9bc-45a3-a87a-5513e232e843",
            "Size": 20,
            "SnapshotId": "snap-0e946653493cb0447",
            "State": "available",
            "VolumeId": "vol-0d371921d4ca797b0",
            "Iops": 100,
            "VolumeType": "gp2",
            "FastRestored": true
        }
    ]
}
```

빠른 스냅샷 복원 모니터링

Amazon EBS는 스냅샷에 대한 빠른 스냅샷 복원 상태가 변경되면 Amazon CloudWatch 이벤트를 보냅니다. 자세한 내용은 [EBS 빠른 스냅샷 복원 이벤트 \(p. 1089\)](#) 단원을 참조하십시오.

요금 및 결제

특정 가용 영역의 스냅샷에 대해 빠른 스냅샷 복원이 활성화된 1분마다 요금이 청구됩니다. 요금은 최소 1시간으로 비례 청구됩니다.

예를 들어 한 달(30일) 동안 us-east-1a에서 한 스냅샷에 대해 빠른 스냅샷 복원을 활성화하면 \$540(스냅샷 1개 x AZ 1개 x 720시간 x 시간당 \$0.75)가 청구됩니다. 동일한 기간 동안 us-east-1a, us-east-1b, us-east-1c에서 두 스냅샷에 대해 빠른 스냅샷 복원을 활성화하면 \$3240(스냅샷 2개 x AZ 3개 x 720시간 x 시간당 \$0.75)가 청구됩니다.

본인에게 공유된 퍼블릭 또는 프라이빗 스냅샷에 대해 빠른 스냅샷 복원을 활성화하는 경우 계정에 요금이 청구되며 스냅샷 소유자에게는 요금이 청구되지 않습니다. 본인에게 공유된 스냅샷이 스냅샷 소유자에 의해 삭제되거나 공유 해제되면 계정의 스냅샷에 대해 빠른 스냅샷 복원이 비활성화되고 청구가 중지됩니다.

자세한 내용은 [Amazon EBS 요금](#)을 참조하십시오.

Windows 인스턴스의 Amazon EBS 및 NVMe

Nitro 시스템 (p. 118) 기반 인스턴스에서는 EBS 볼륨이 NVMe 블록 디바이스로 표시됩니다.

블록과 디바이스 간의 인터페이스가 무엇이든 간에 [Amazon EBS 제품 세부 정보](#)에 명시된 EBS 성능 보장은 유효합니다.

목차

- [NVMe 드라이버 설치 또는 업그레이드 \(p. 1048\)](#)
- [EBS 디바이스 식별 \(p. 1048\)](#)
- [NVMe EBS 볼륨 관련 작업 \(p. 1049\)](#)
- [I/O 작업 시간 제한 \(p. 1049\)](#)

NVMe 드라이버 설치 또는 업그레이드

Windows Server 2008 R2 이상용 AWS Windows AMI에는 AWS NVMe 드라이버가 포함되어 있습니다. Amazon에서 제공하는 최신 AWS Windows AMI를 사용하지 않는 경우 [AWS NVMe 드라이버 설치 또는 업그레이드 \(p. 539\)](#)을 참조하십시오.

EBS 디바이스 식별

EBS는 단일 루트 I/O 가상화(SR-IOV)를 사용하여 NVMe 사양을 사용하는 Nitro 기반 인스턴스에서 볼륨 연결을 제공합니다. 이러한 디바이스는 운영 체제의 표준 NVMe 드라이버에 의존합니다. 이러한 드라이버는 일반적으로 인스턴스 부팅 중에 PCI 버스를 스캔하여 연결된 디바이스를 검색한 다음, 블록 디바이스 맵핑에서 디바이스가 지정되는 방식이 아닌, 디바이스가 응답하는 순서에 기초하여 디바이스 노드를 생성합니다.

Windows Server 2008 R2 및 이상

또한 `ebsnvme-id` 명령을 실행하여 NVMe 디바이스 디스크 번호를 EBS 볼륨 ID와 디바이스 이름에 매핑할 수 있습니다. 기본적으로 모든 EBS NVMe 디바이스가 열거되어 있습니다. 디스크 번호를 특정 디바이스의 열거 번호로 전달할 수 있습니다. `Ebsnvme-id`는 C:\PROGRAMDATA\AMAZONTools에 있는 최신 AWS 제공 Windows Server AMI에 포함되어 있습니다.

또한 [ebsnvme-id.zip](#)을 다운로드하고 그 내용을 Amazon EC2 인스턴스에 추출하여 `ebsnvme-id.exe`에 액세스할 수 있습니다.

```
PS C:\Users\Administrator\Desktop> ebsnvme-id.exe
Disk Number: 0
Volume ID: vol-0d6d7ee9f6e471a7f
Device Name: sda1

Disk Number: 1
Volume ID: vol-03a26248ff39b57cf
Device Name: xvdd

Disk Number: 2
Volume ID: vol-038bd1c629aa125e6
Device Name: xvde

Disk Number: 3
Volume ID: vol-034f9d29ec0b64c89
Device Name: xvdb

Disk Number: 4
Volume ID: vol-03e2dbe464b66f0a1
```

```
Device Name: xvdc
PS C:\Users\Administrator\Desktop> ebsnvme-id.exe 4
Disk Number: 4
Volume ID: vol-03e2dbe464b66f0a1
Device Name: xvdc
```

NVMe EBS 볼륨 관련 작업

최신 AWS Windows AMI에는 EBS 볼륨을 NVMe 블록 디바이스로 표시하는 인스턴스 유형에 필요한 AWS NVMe 드라이버가 포함되어 있습니다. 하지만 Windows 시스템에서 루트 볼륨 크기를 조정한 경우 이 변경 사항이 인스턴스에 반영되도록 볼륨을 다시 스캔해야 합니다. 다른 AMI에서 인스턴스를 시작한 경우에는 필요한 AWS NVMe 드라이버가 없을 수 있습니다. 인스턴스에 최신 AWS NVMe 드라이브가 없는 경우 설치해야 합니다. 자세한 내용은 [Windows 인스턴스의 AWS NVMe 드라이버 \(p. 539\)](#) 단원을 참조하십시오.

I/O 작업 시간 제한

대부분의 운영 체제는 I/O 작업이 NVMe 디바이스에 제출되는 시간에 제한을 두고 있습니다. Windows 시스템에서 기본 제한 시간은 60초이며 최대 허용 시간은 255초입니다. [SCSI 미니포트 드라이버를 위한 레지스트리 항목](#)에 설명된 절차를 사용하여 `TimeoutValue` 디스크 클래스 레지스트리 설정을 수정할 수 있습니다.

Amazon EBS-최적화 인스턴스

Amazon EBS 최적화 인스턴스는 최적화된 구성 스택을 사용하며 Amazon EBS I/O를 위한 추가 전용 용량을 제공합니다. 이러한 최적화를 통해 인스턴스에서 Amazon EBS I/O와 기타 트래픽 간의 경합이 최소화되어 EBS 볼륨의 성능이 극대화됩니다.

EBS 최적화 인스턴스는 Amazon EBS에 전용 대역폭을 제공합니다. EBS 최적화 인스턴스에 연결된 경우 범용 SSD(gp2) 볼륨은 99%의 시간 동안 기본 성능 및 버스트 성능을 제공하도록 설계되며, 프로비저닝된 IOPS SSD(io1 및 io2) 볼륨은 99.9%의 시간 동안 프로비저닝된 성능을 제공하도록 설계됩니다. 처리량에 최적화된 HDD(st1) 및 Cold HDD(sc1)는 모두 99%의 기간 동안 90%의 버스트 처리량에 대해 성능 일관성을 보장합니다. 매 시간 총 처리량 목표 99%를 달성하기 위해, 준수하지 않는 기간은 대략적으로 균등하게 분산됩니다. 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.

목차

- [지원되는 인스턴스 유형 \(p. 1049\)](#)
- [최대 성능 얻기 \(p. 1061\)](#)
- [시작 시 EBS 최적화 활성화 \(p. 1062\)](#)
- [기존 인스턴스에 대해 EBS 최적화 활성화 \(p. 1062\)](#)

지원되는 인스턴스 유형

다음 표에서는 어떤 인스턴스 유형이 EBS 최적화를 지원하는지를 보여 줍니다. 여기에는 Amazon EBS에 대한 전용 대역폭, 스트리밍 읽기 워크로드 및 128KiB I/O 크기로 해당 연결에서 달성할 수 있는 일반적인 최대 집계 처리량, 16KiB I/O 크기를 사용할 경우 인스턴스가 지원할 수 있는 IOPS 최대량이 포함됩니다. 애플리케이션에 필요한 것보다 많은 전용 Amazon EBS 처리량을 제공하는 EBS에 최적화된 인스턴스를 선택해야 합니다. 그렇게 하지 않으면 Amazon EBS와 Amazon EC2 간의 연결이 성능 병목 현상으로 변할 수 있습니다.

기본적으로 EBS에 최적화됨

다음 표에는 EBS 최적화를 지원하고 EBS 최적화가 기본적으로 활성화되는 인스턴스 유형이 나열됩니다. EBS 최적화를 활성화할 필요가 없으며 EBS 최적화를 비활성화해도 효과가 없습니다.

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
c4.large	500	62.5	4,000
c4.xlarge	750	93.75	6,000
c4.2xlarge	1,000	125	8,000
c4.4xlarge	2,000건	250	16,000
c4.8xlarge	4,000	500	32,000
c5.large *	4,750	593.75	20,000건
c5.xlarge *	4,750	593.75	20,000건
c5.2xlarge *	4,750	593.75	20,000건
c5.4xlarge	4,750	593.75	20,000건
c5.9xlarge	9,500	1,187.5	40,000
c5.12xlarge	9,500	1,187.5	40,000
c5.18xlarge	19,000	2,375	80,000
c5.24xlarge	19,000	2,375	80,000
c5.metal	19,000	2,375	80,000
c5a.large *	3,170	396	13,300
c5a.xlarge *	3,170	396	13,300
c5a.2xlarge *	3,170	396	13,300
c5a.4xlarge *	3,170	396	13,300
c5a.8xlarge	3,170	396	13,300
c5a.12xlarge	4,750	594	20,000건
c5a.16xlarge	6,300	788	26,700
c5a.24xlarge	9,500	1,188	40,000
c5ad.large *	3,170	396	13,300
c5ad.xlarge *	3,170	396	13,300
c5ad.2xlarge *	3,170	396	13,300
c5ad.4xlarge *	3,170	396	13,300
c5ad.8xlarge	3,170	396	13,300
c5ad.12xlarge	4,750	594	20,000건
c5ad.16xlarge	6,300	788	26,700
c5ad.24xlarge	9,500	1,188	40,000

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
c5d.large *	4,750	593.75	20,000건
c5d.xlarge *	4,750	593.75	20,000건
c5d.2xlarge *	4,750	593.75	20,000건
c5d.4xlarge	4,750	593.75	20,000건
c5d.9xlarge	9,500	1,187.5	40,000
c5d.12xlarge	9,500	1,187.5	40,000
c5d.18xlarge	19,000	2,375	80,000
c5d.24xlarge	19,000	2,375	80,000
c5d.metal	19,000	2,375	80,000
c5n.large *	4,750	593.75	20,000건
c5n.xlarge *	4,750	593.75	20,000건
c5n.2xlarge *	4,750	593.75	20,000건
c5n.4xlarge	4,750	593.75	20,000건
c5n.9xlarge	9,500	1,187.5	40,000
c5n.18xlarge	19,000	2,375	80,000
c5n.metal	19,000	2,375	80,000
d2.xlarge	750	93.75	6,000
d2.2xlarge	1,000	125	8,000
d2.4xlarge	2,000건	250	16,000
d2.8xlarge	4,000	500	32,000
f1.2xlarge	1,700	212.5	12,000
f1.4xlarge	3,500	437.5	44,000
f1.16xlarge	14,000	1,750	75,000
g3s.xlarge	850	106.25	5,000
g3.4xlarge	3,500	437.5	20,000건
g3.8xlarge	7,000	875	40,000
g3.16xlarge	14,000	1,750	80,000
g4dn.xlarge *	3,500	437.5	20,000건
g4dn.2xlarge *	3,500	437.5	20,000건
g4dn.4xlarge	4,750	593.75	20,000건

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
g4dn.8xlarge	9,500	1,187.5	40,000
g4dn.12xlarge	9,500	1,187.5	40,000
g4dn.16xlarge	9,500	1,187.5	40,000
g4dn.metal	19,000	2,375	80,000
h1.2xlarge	1,750	218.75	12,000
h1.4xlarge	3,500	437.5	20,000건
h1.8xlarge	7,000	875	40,000
h1.16xlarge	14,000	1,750	80,000
i3.large	425	53.13	3000
i3.xlarge	850	106.25	6000
i3.2xlarge	1,700	212.5	12,000
i3.4xlarge	3,500	437.5	16,000
i3.8xlarge	7,000	875	32,500
i3.16xlarge	14,000	1,750	65,000
i3.metal	19,000	2,375	80,000
i3en.large *	4,750	593.75	20,000건
i3en.xlarge *	4,750	593.75	20,000건
i3en.2xlarge *	4,750	593.75	20,000건
i3en.3xlarge *	4,750	593.75	20,000건
i3en.6xlarge	4,750	593.75	20,000건
i3en.12xlarge	9,500	1,187.5	40,000
i3en.24xlarge	19,000	2,375	80,000
i3en.metal	19,000	2,375	80,000
m4.large	450	56.25	3,600
m4.xlarge	750	93.75	6,000
m4.2xlarge	1,000	125	8,000
m4.4xlarge	2,000건	250	16,000
m4.10xlarge	4,000	500	32,000
m4.16xlarge	10,000건	1,250	65,000
m5.large *	4,750	593.75	18,750

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
m5.xlarge *	4,750	593.75	18,750
m5.2xlarge *	4,750	593.75	18,750
m5.4xlarge	4,750	593.75	18,750
m5.8xlarge	6,800	850	30,000개
m5.12xlarge	9,500	1,187.5	40,000
m5.16xlarge	13,600	1,700	60,000
m5.24xlarge	19,000	2,375	80,000
m5.metal	19,000	2,375	80,000
m5a.large *	2,880	360	16,000
m5a.xlarge *	2,880	360	16,000
m5a.2xlarge *	2,880	360	16,000
m5a.4xlarge	2,880	360	16,000
m5a.8xlarge	4,750	593.75	20,000건
m5a.12xlarge	6,780	847.5	30,000개
m5a.16xlarge	9,500	1,187.50	40,000
m5a.24xlarge	13,570	1,696.25	60,000
m5ad.large *	2,880	360	16,000
m5ad.xlarge *	2,880	360	16,000
m5ad.2xlarge *	2,880	360	16,000
m5ad.4xlarge	2,880	360	16,000
m5ad.8xlarge	4,750	593.75	20,000건
m5ad.12xlarge	6,780	847.5	30,000개
m5ad.16xlarge	9,500	1,187.5	40,000
m5ad.24xlarge	13,570	1,696.25	60,000
m5d.large *	4,750	593.75	18,750
m5d.xlarge *	4,750	593.75	18,750
m5d.2xlarge *	4,750	593.75	18,750
m5d.4xlarge	4,750	593.75	18,750
m5d.8xlarge	6,800	850	30,000개
m5d.12xlarge	9,500	1,187.5	40,000

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
m5d.16xlarge	13,600	1,700	60,000
m5d.24xlarge	19,000	2,375	80,000
m5d.metal	19,000	2,375	80,000
m5dn.large *	4,750	593.75	18,750
m5dn.xlarge *	4,750	593.75	18,750
m5dn.2xlarge *	4,750	593.75	18,750
m5dn.4xlarge	4,750	593.75	18,750
m5dn.8xlarge	6,800	850	30,000개
m5dn.12xlarge	9,500	1,187.5	40,000
m5dn.16xlarge	13,600	1,700	60,000
m5dn.24xlarge	19,000	2,375	80,000
m5n.large *	4,750	593.75	18,750
m5n.xlarge *	4,750	593.75	18,750
m5n.2xlarge *	4,750	593.75	18,750
m5n.4xlarge	4,750	593.75	18,750
m5n.8xlarge	6,800	850	30,000개
m5n.12xlarge	9,500	1,187.5	40,000
m5n.16xlarge	13,600	1,700	60,000
m5n.24xlarge	19,000	2,375	80,000
p2.xlarge	750	93.75	6,000
p2.8xlarge	5,000	625	32,500
p2.16xlarge	10,000개	1,250	65,000
p3.2xlarge	1,750	218.75	10,000개
p3.8xlarge	7,000	875	40,000
p3.16xlarge	14,000	1,750	80,000
p3dn.24xlarge	19,000	2,375	80,000
r4.large	425	53.13	3,000
r4.xlarge	850	106.25	6,000
r4.2xlarge	1,700	212.5	12,000
r4.4xlarge	3,500	437.5	18,750

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
r4.8xlarge	7,000	875	37,500
r4.16xlarge	14,000	1,750	75,000
r5.large *	4,750	593.75	18,750
r5.xlarge *	4,750	593.75	18,750
r5.2xlarge *	4,750	593.75	18,750
r5.4xlarge	4,750	593.75	18,750
r5.8xlarge	6,800	850	30,000건
r5.12xlarge	9,500	1,187.5	40,000
r5.16xlarge	13,600	1,700	60,000
r5.24xlarge	19,000	2,375	80,000
r5.metal	19,000	2,375	80,000
r5a.large *	2,880	360	16,000
r5a.xlarge *	2,880	360	16,000
r5a.2xlarge *	2,880	360	16,000
r5a.4xlarge	2,880	360	16,000
r5a.8xlarge	4,750	593.75	20,000건
r5a.12xlarge	6,780	847.5	30,000건
r5a.16xlarge	9,500	1,187.5	40,000
r5a.24xlarge	13,570	1,696.25	60,000
r5ad.large *	2,880	360	16,000
r5ad.xlarge *	2,880	360	16,000
r5ad.2xlarge *	2,880	360	16,000
r5ad.4xlarge	2,880	360	16,000
r5ad.8xlarge	4,750	593.75	20,000건
r5ad.12xlarge	6,780	847.5	30,000건
r5ad.16xlarge	9,500	1,187.5	40,000
r5ad.24xlarge	13,570	1,696.25	60,000
r5d.large *	4,750	593.75	18,750
r5d.xlarge *	4,750	593.75	18,750
r5d.2xlarge *	4,750	593.75	18,750

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
r5d.4xlarge	4,750	593.75	18,750
r5d.8xlarge	6,800	850	30,000개
r5d.12xlarge	9,500	1,187.5	40,000
r5d.16xlarge	13,600	1,700	60,000
r5d.24xlarge	19,000	2,375	80,000
r5d.metal	19,000	2,375	80,000
r5dn.large *	4,750	593.75	18,750
r5dn.xlarge *	4,750	593.75	18,750
r5dn.2xlarge *	4,750	593.75	18,750
r5dn.4xlarge	4,750	593.75	18,750
r5dn.8xlarge	6,800	850	30,000개
r5dn.12xlarge	9,500	1,187.5	40,000
r5dn.16xlarge	13,600	1,700	60,000
r5dn.24xlarge	19,000	2,375	80,000
r5n.large *	4,750	593.75	18,750
r5n.xlarge *	4,750	593.75	18,750
r5n.2xlarge *	4,750	593.75	18,750
r5n.4xlarge	4,750	593.75	18,750
r5n.8xlarge	6,800	850	30,000개
r5n.12xlarge	9,500	1,187.5	40,000
r5n.16xlarge	13,600	1,700	60,000
r5n.24xlarge	19,000	2,375	80,000
t3.nano *	2,085	260.57	11,800
t3.micro *	2,085	260.57	11,800
t3.small *	2,085	260.57	11,800
t3.medium *	2,085	260.57	11,800
t3.large *	2,780	347.5	15,700
t3.xlarge *	2,780	347.5	15,700
t3.2xlarge *	2,780	347.5	15,700
t3a.nano *	2,085	260.57	11,800

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
t3a.micro *	2,085	260.57	11,800
t3a.small *	2,085	260.57	11,800
t3a.medium *	2,085	260.57	11,800
t3a.large *	2,780	347.5	15,700
t3a.xlarge *	2,780	347.5	15,700
t3a.2xlarge *	2,780	347.5	15,700
u-6tb1.metal	38,000	4,750	160,000
u-9tb1.metal	38,000	4,750	160,000
u-12tb1.metal	38,000	4,750	160,000
u-18tb1.metal	38,000	4,750	160,000
u-24tb1.metal	38,000	4,750	160,000
x1.16xlarge	7,000	875	40,000
x1.32xlarge	14,000	1,750	80,000
x1e.xlarge	500	62.5	3,700
x1e.2xlarge	1,000	125	7,400
x1e.4xlarge	1,750	218.75	10,000개
x1e.8xlarge	3,500	437.5	20,000건
x1e.16xlarge	7,000	875	40,000
x1e.32xlarge	14,000	1,750	80,000
z1d.large *	3,170	396.25	13,333
z1d.xlarge *	3,170	396.25	13,333
z1d.2xlarge	3,170	396.25	13,333
z1d.3xlarge	4,750	593.75	20,000건
z1d.6xlarge	9,500	1,187.5	40,000
z1d.12xlarge	19,000	2,375	80,000
z1d.metal	19,000	2,375	80,000

* 이러한 인스턴스 유형에서는 24시간마다 최소 한 번씩 30분의 최대 성능을 지원합니다. 30분 넘는 시간 동안 지속적으로 최대 성능이 필요한 워크로드가 있는 경우 다음 표의 기준 성능에 따라 인스턴스 유형을 선택합니다.

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	기준 대역폭(Mbps)	기준 처리량(MB/s, 128KiB I/O)	기준 IOPS(16 KiB I/O)
c5.large	650	81.25	4,000
c5.xlarge	1,150	143.75	6,000
c5.2xlarge	2,300	287.5	10,000개
c5a.large	200	25	800
c5a.xlarge	400	50	1,600
c5a.2xlarge	800	100	3,200
c5a.4xlarge	1,580	198	6,600
c5ad.large	200	25	800
c5ad.xlarge	400	50	1,600
c5ad.2xlarge	800	100	3,200
c5ad.4xlarge	1,580	198	6,600
c5d.large	650	81.25	4,000
c5d.xlarge	1,150	143.75	6,000
c5d.2xlarge	2,300	287.5	10,000개
c5n.large	650	81.25	4,000
c5n.xlarge	1,150	143.75	6,000
c5n.2xlarge	2,300	287.5	10,000개
g4dn.xlarge	950	118.75	3,000
g4dn.2xlarge	1,150	143.75	6,000
i3en.large	577	72.1	3,000
i3en.xlarge	1,154	144.2	6,000
i3en.2xlarge	2,307	288.39	12,000
i3en.3xlarge	3,800	475	15,000
m5.large	650	81.25	3,600
m5.xlarge	1,150	143.75	6,000
m5.2xlarge	2,300	287.5	12,000
m5a.large	650	81.25	3,600
m5a.xlarge	1,085	135.63	6,000
m5a.2xlarge	1,580	197.5	8,333
m5ad.large	650	81.25	3,600

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	기준 대역폭(Mbps)	기준 처리량(MB/s, 128KiB I/O)	기준 IOPS(16 KiB I/O)
m5ad.xlarge	1,085	135.63	6,000
m5ad.2xlarge	1,580	197.5	8,333
m5d.large	650	81.25	3,600
m5d.xlarge	1,150	143.75	6,000
m5d.2xlarge	2,300	287.5	12,000
m5dn.large	650	81.25	3,600
m5dn.xlarge	1,150	143.75	6,000
m5dn.2xlarge	2,300	287.5	12,000
m5n.large	650	81.25	3,600
m5n.xlarge	1,150	143.75	6,000
m5n.2xlarge	2,300	287.5	12,000
r5.large	650	81.25	3,600
r5.xlarge	1,150	143.75	6,000
r5.2xlarge	2,300	287.5	12,000
r5a.large	650	81.25	3,600
r5a.xlarge	1,085	135.63	6,000
r5a.2xlarge	1,580	197.5	8,333
r5ad.large	650	81.25	3,600
r5ad.xlarge	1,085	135.63	6,000
r5ad.2xlarge	1,580	197.5	8,333
r5d.large	650	81.25	3,600
r5d.xlarge	1,150	143.75	6,000
r5d.2xlarge	2,300	287.5	12,000
r5dn.large	650	81.25	3,600
r5dn.xlarge	1,150	143.75	6,000
r5dn.2xlarge	2,300	287.5	12,000
r5n.large	650	81.25	3,600
r5n.xlarge	1,150	143.75	6,000
r5n.2xlarge	2,300	287.5	12,000
t3.nano	43	5.43	250

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS 최적화

인스턴스 크기	기준 대역폭(Mbps)	기준 처리량(MB/s, 128KiB I/O)	기준 IOPS(16 KiB I/O)
t3.micro	87	10.86	500
t3.small	174	21.71	1,000
t3.medium	347	43.43	2,000건
t3.large	695	86.86	4,000
t3.xlarge	695	86.86	4,000
t3.2xlarge	695	86.86	4,000
t3a.nano	45	5.63	250
t3a.micro	90	11.25	500
t3a.small	175	21.88	1,000
t3a.medium	350	43.75	2,000건
t3a.large	695	86.86	4,000
t3a.xlarge	695	86.86	4,000
t3a.2xlarge	695	86.86	4,000
z1d.large	800	100	3,333
z1d.xlarge	1,580	197.5	6,667

EBS 최적화 지원됨

다음 표에는 EBS 최적화를 지원하지만 EBS 최적화가 기본적으로 활성화되지 않는 인스턴스 유형이 나열됩니다. 이러한 인스턴스를 시작할 때 또는 인스턴스를 실행한 후 EBS 최적화를 활성화할 수 있습니다. 설명된 수준의 성능을 달성하려면 인스턴스에서 EBS 최적화를 활성화해야 합니다. 기본적으로 EBS 최적화되지 않은 인스턴스에 EBS 최적화를 사용하도록 설정할 경우 전용 용량을 위해 소정의 시간당 추가 요금이 청구됩니다. 요금 정보는 [Amazon EC2 요금, 온디マン드 요금 페이지](#)에서 EBS 최적화 인스턴스를 참조하세요.

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
c1.xlarge	1,000	125	8,000
c3.xlarge	500	62.5	4,000
c3.2xlarge	1,000	125	8,000
c3.4xlarge	2,000건	250	16,000
g2.2xlarge	1,000	125	8,000
i2.xlarge	500	62.5	4,000
i2.2xlarge	1,000	125	8,000
i2.4xlarge	2,000건	250	16,000

인스턴스 크기	최대 대역폭(Mbps)	최대 처리량(MB/s, 128KiB I/O)	최대 IOPS(16KiB I/O)
m1.large	500	62.5	4,000
m1.xlarge	1,000	125	8,000
m2.2xlarge	500	62.5	4,000
m2.4xlarge	1,000	125	8,000
m3.xlarge	500	62.5	4,000
m3.2xlarge	1,000	125	8,000
r3.xlarge	500	62.5	4,000
r3.2xlarge	1,000	125	8,000
r3.4xlarge	2,000건	250	16,000

i2.8xlarge, c3.8xlarge, r3.8xlarge 인스턴스에는 전용 EBS 대역폭이 없으므로 EBS 최적화에 영향을 미치지 않습니다. 이러한 인스턴스에서 네트워크 트래픽과 트래픽은 동일한 10기가비트 네트워크 인터페이스를 공유합니다.

최대 성능 얻기

EBSIOBalance% 및 EBSSByteBalance% 지표를 사용하여 인스턴스 크기가 올바르게 설정되는지 여부를 확인할 수 있습니다. CloudWatch 콘솔에서 이 지표를 확인하고 사용자가 지정한 임계값에 따라 트리거될 경보를 설정할 수 있습니다. 이 지표는 백분율로 표현됩니다. 일관되게 낮은 균형 백분율을 나타내는 인스턴스는 규모를 늘리기에 적합한 대상입니다. 균형 백분율이 100% 이하로 결코 떨어지지 않는 인스턴스는 규모를 줄이기에 적합한 대상입니다. 자세한 정보는 [CloudWatch를 사용해 인스턴스 모니터링 \(p. 667\)](#) 단원을 참조하십시오.

고용량 메모리 인스턴스는 SAP HANA 인 메모리 데이터베이스의 프로덕션 배포를 비롯하여 클라우드에서 대규모 인 메모리 데이터베이스를 실행하도록 설계되었습니다. EBS 성능을 최대화하려면 동일한 프로비저닝된 성능의 io1 또는 io2 볼륨이 짹수로 포함된 고용량 메모리 인스턴스를 사용합니다. 예를 들어 IOPS가 많은 워크로드의 경우 프로비저닝된 IOPS가 40,000인 io1 또는 io2 볼륨 4개를 사용하여 최대 160,000의 인스턴스 IOPS를 얻을 수 있습니다. 마찬가지로 처리량이 많은 워크로드의 경우 프로비저닝된 IOPS가 48,000인 io1 또는 io2 볼륨 6개를 사용하여 최대 4,750MB/s 처리량을 얻을 수 있습니다. 추가 권장 사항은 [SAP HANA용 스토리지 구성](#)을 참조하십시오.

고려 사항

- 2020년 2월 26일 이후 시작된 G4, I3en, M5a, M5ad, R5a, R5ad, T3, T3a 및 Z1d 인스턴스는 위 표에 나열된 최대 성능을 제공합니다. 2020년 2월 26일 이전에 시작된 인스턴스에서 최대 성능을 얻으려면 해당 인스턴스를 중지했다가 시작합니다.
- 2019년 12월 3일 이후에 시작된 C5, C5d, C5n, M5, M5d, M5n, M5dn, R5, R5d, R5n, R5dn, 및 P3dn 인스턴스는 위 표에 나열된 최대 성능을 제공합니다. 2019년 12월 3일 이전에 시작된 인스턴스에서 최대 성능을 얻으려면 해당 인스턴스를 중지했다가 시작합니다.
- 2020년 3월 12일 이후에 시작된 u-6tb1.metal, u-9tb1.metal 및 u-12tb1.metal 인스턴스는 위 표에 나열된 성능을 제공합니다. 2020년 3월 12일 이전에 시작된 이러한 유형의 인스턴스는 성능이 더 낮을 수 있습니다. 2020년 3월 12일 이전에 시작된 인스턴스에서 최대 성능을 얻으려면 계정 팀에 문의하여 추가 비용 없이 인스턴스를 업그레이드하십시오.

시작 시 EBS 최적화 활성화

EBS 최적화에 대한 속성을 설정하여 인스턴스에 대한 최적화를 활성화할 수 있습니다.

콘솔로 인스턴스 시작 시 Amazon EBS 최적화를 활성화하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 인스턴스 시작을 선택합니다.
3. 1단계: Amazon 머신 이미지(AMI) 선택에서 AMI를 선택합니다.
4. 2단계: 인스턴스 유형 선택에서 Amazon EBS 최적화를 지원하는 인스턴스 유형을 선택합니다.
5. 3단계: 인스턴스 세부 정보 구성에서 필요한 필드 정보를 모두 입력하고 EBS 최적 인스턴스로 시작을 선택합니다. 이전 단계에서 선택한 인스턴스 유형이 최적화를 지원하지 않을 경우 이 옵션이 제공되지 않습니다. 선택한 인스턴스 유형이 기본적으로 Amazon EBS 최적화를 지원할 경우에는 이 옵션이 선택되며 선택을 취소할 수 없습니다.
6. 지시에 따라 마법사를 완료하고 인스턴스를 시작합니다.

명령줄을 사용하여 인스턴스 시작 시 EBS 최적화를 활성화하려면

해당 옵션과 함께 다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `run-instances`와 `--ebs-optimized`(AWS CLI)
- `New-EC2Instance`와 `-EbsOptimized`(Windows PowerShell용 AWS 도구)

기존 인스턴스에 대해 EBS 최적화 활성화

기존 인스턴스에 대해 Amazon EBS 최적화 인스턴스 속성을 수정하여 최적화를 활성화하거나 비활성화할 수 있습니다. 인스턴스가 실행 중인 경우 먼저 인스턴스를 중지해야 합니다.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

콘솔을 사용하여 기존 인스턴스에 대해 EBS 최적화를 활성화하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
3. 인스턴스를 중지하기 위해 작업, 인스턴스 상태, 중지를 선택합니다.
4. 확인 대화 상자가 나타나면 예, 중지를 선택합니다. 인스턴스가 중지하는 데 몇 분 정도 걸릴 수 있습니다.
5. 인스턴스를 선택한 상태에서 Actions, Instance Settings, Change Instance Type를 선택합니다.
6. 인스턴스 유형 변경 대화 상자에서 다음 중 하나를 수행하십시오.
 - 해당 인스턴스의 인스턴스 유형이 기본적으로 Amazon EBS 최적화되었을 경우 EBS 최적이 선택되고 이를 변경할 수 없습니다. 해당 인스턴스에 대해 Amazon EBS 최적화가 이미 활성화되었으므로 취소를 선택합니다.
 - 해당 인스턴스의 인스턴스 유형이 Amazon EBS 최적화를 지원할 경우 EBS 최적, 적용을 선택합니다.
 - 해당 인스턴스의 인스턴스 유형이 Amazon EBS 최적화를 지원하지 않을 경우 EBS 최적을 선택할 수 없습니다. 인스턴스 유형에서 Amazon EBS 최적화를 지원하는 인스턴스 유형을 선택하고 EBS 최적, 적용을 선택합니다.
7. 작업, 인스턴스 상태, 시작을 차례로 선택합니다.

명령줄을 사용하여 기존 인스턴스에 대해 EBS 최적화를 활성화하려면

1. 인스턴스가 실행 중인 경우 다음 명령 중 하나를 사용하여 인스턴스를 중지합니다.
 - [stop-instances\(AWS CLI\)](#)
 - [Stop-EC2Instance\(Windows PowerShell용 AWS 도구\)](#)
2. EBS 최적화를 활성화하려면 다음 명령 중 하나를 해당 옵션과 함께 사용합니다.
 - [modify-instance-attribute](#)와 [--ebs-optimized\(AWS CLI\)](#)
 - [Edit-EC2InstanceAttribute](#)와 [-EbsOptimized\(Windows PowerShell용 AWS 도구\)](#)

Amazon EBS 인스턴스의 Windows 볼륨 성능

I/O 특성, 인스턴스와 볼륨의 구성 등 여러 가지 요인이 Amazon EBS 볼륨에 영향을 끼칠 수 있습니다. Amazon EBS 및 Amazon EC2 제품 세부 정보 페이지의 지침을 따르는 고객은 대체로 처음부터 우수한 성능을 달성할 수 있습니다. 그러나 경우에 따라 플랫폼에서 피크 성능을 얻기 위해서는 약간의 튜닝이 필요합니다. 이 주제에서는 일반적인 모범 사례 및 특정한 사용 사례에만 적용되는 성능 튜닝에 대해 설명합니다. 벤치마킹 외에도 실제 워크로드의 정보에 따라 성능을 튜닝하여 최적의 구성을 결정하는 것이 좋습니다. EBS 볼륨 작업의 기초를 배운 후에는 필요한 I/O 성능과 그러한 요건에 맞게 Amazon EBS 성능을 향상하기 위한 옵션을 살펴보는 것이 좋습니다.

EBS 볼륨 유형의 성능에 대한 AWS 업데이트가 기존 볼륨에 즉시 적용되지 않을 수 있습니다. 기존 볼륨의 전체 성능을 확인하려면 먼저 볼륨에 대해 [ModifyVolume](#) 작업을 수행해야 할 수 있습니다. 자세한 정보는 [Windows에서 EBS 볼륨의 크기, IOPS 또는 유형 설정을 참조하십시오.](#)

내용

- [Amazon EBS 성능 팁 \(p. 1063\)](#)
- [I/O 특성 및 모니터링 \(p. 1064\)](#)
- [Amazon EBS 볼륨 초기화 \(p. 1066\)](#)
- [Windows의 RAID 구성 \(p. 1068\)](#)
- [EBS 볼륨 벤치마킹 \(p. 1073\)](#)

Amazon EBS 성능 팁

이러한 팁은 다양한 사용자 시나리오에서 최적의 EBS 볼륨 성능을 달성하는 방법에 대한 모범 사례를 보여 줍니다.

EBS 최적화 인스턴스 사용

EBS 최적화 처리량을 지원하지 않는 인스턴스에서는 네트워크 트래픽이 사용자의 인스턴스와 EBS 볼륨 간 트래픽과 경합할 수 있습니다. EBS 최적화 인스턴스에서는 이 두 유형의 트래픽이 분리되어 있습니다. 일부 EBS 최적화 인스턴스 구성은 추가 요금을 요구하지만(예: C3, R3, M3), 일부는 추가 요금 없이 항상 EBS에 최적화됩니다(예: M4, C4, C5, D2). 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

성능 계산 방법 이해

EBS 볼륨의 성능을 측정할 때는 관련된 측정 단위와 성능 계산 방법을 이해해야 합니다. 자세한 내용은 [I/O 특성 및 모니터링 \(p. 1064\)](#) 단원을 참조하십시오.

워크로드 이해

EBS 볼륨의 최대 성능, I/O 작업의 크기와 횟수, 각 작업을 완료하는 데 걸리는 시간은 서로 관련이 있습니다. 이러한 각 요소(성능, I/O 및 지연 시간)는 서로에게 영향을 미치며 애플리케이션마다 다른 요소에 더 민감합니다.

스냅샷에서 볼륨을 초기화하는 경우 성능 저하에 유의

스냅샷에서 생성된 새 EBS 볼륨의 각 데이터 블록에 처음 액세스할 때 지연 시간이 상당히 증가합니다. 다음 옵션 중 하나를 사용하여 이 성능 저하를 방지할 수 있습니다.

- 볼륨을 프로덕션에 투입하기 전에 각 블록에 액세스합니다. 이 프로세스를 초기화(이전에는 사전 위밍이라고 함)라고 합니다. 자세한 내용은 [Amazon EBS 볼륨 초기화 \(p. 1066\)](#) 단원을 참조하십시오.
- 스냅샷에서 빠른 스냅샷 복원을 활성화하여 스냅샷에서 생성된 EBS 볼륨이 생성 시 완전히 초기화되고 모든 프로비저닝된 성능을 즉시 제공하도록 보장합니다. 자세한 내용은 [Amazon EBS 빠른 스냅샷 복원 \(p. 1044\)](#) 단원을 참조하십시오.

HDD 성능을 저하시킬 수 있는 요인

처리량에 최적화된 HDD(st1) 또는 Cold HDD(sc1) 볼륨의 스냅샷을 생성하는 경우, 스냅샷이 진행되는 동안 성능이 볼륨의 기준 값까지 떨어질 수 있습니다. 이 동작은 이러한 볼륨 유형에만 해당합니다. 성능을 제한할 수 있는 다른 요소로는 인스턴스가 지원할 수 있는 수준 이상의 처리량을 구동하는 경우, 스냅샷에서 생성된 볼륨을 초기화하는 동안의 성능 저하, 볼륨에 소량의 랜덤 I/O가 과도하게 많은 경우 등을 들 수 있습니다. HDD 볼륨의 처리량 계산에 관한 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.

애플리케이션이 충분한 I/O 요청을 보내지 않는 경우에도 성능이 영향을 받을 수 있습니다. 볼륨의 대기열 길이와 I/O 크기를 보면 확인할 수 있습니다. 대기열 길이는 애플리케이션에서 볼륨으로 보내는 I/O 요청 중 대기 중인 요청의 수입니다. 일관성을 극대화하기 위해 HDD 지원 볼륨은 1MiB 순차 I/O를 수행하는 동안 4 이상의 대기열 길이(반올림)를 유지해야 합니다. 일정한 볼륨 성능 보장에 관한 자세한 내용은 [I/O 특성 및 모니터링 \(p. 1064\)](#) 단원을 참조하십시오.

RAID 0을 사용하여 인스턴스 리소스 활용도 극대화

일부 인스턴스 유형은 단일 EBS 볼륨에 대해 프로비저닝할 수 있는 것보다 많은 I/O 처리량을 구동할 수 있습니다. 여러 gp2, io1, io2, st1 또는 sc1 볼륨을 RAID 0 구성으로 함께 조인하여 이 인스턴스에 사용 가능한 대역폭을 사용할 수 있습니다. 자세한 내용은 [Windows의 RAID 구성 \(p. 1068\)](#) 단원을 참조하십시오.

Amazon CloudWatch를 통한 성능 추적

Amazon Web Services는 Amazon EBS를 통해 보고 분석할 수 있는 Amazon CloudWatch에 대한 성능 지표와 볼륨의 상태를 모니터링하는데 사용할 수 있는 상태 검사를 제공합니다. 자세한 내용은 [볼륨 상태 모니터링 \(p. 957\)](#) 단원을 참조하십시오.

I/O 특성 및 모니터링

지정된 볼륨 구성에서 특정 I/O 특성은 EBS 볼륨의 성능 동작을 구동합니다. SSD 지원 볼륨-범용 SSD(gp2) 및 프로비저닝된 IOPS SSD(io1 및 io2)-I/O 작업이 랜덤이든 순차든 상관 없이 일관된 성능을 제공합니다. HDD 지원 볼륨-처리량에 최적화된 HDD(st1) 및 Cold HDD(sc1)-I/O 작업이 대용량 순차인 경우에만 최적의 성능을 제공합니다. SSD 및 HDD 볼륨이 애플리케이션에서 어떻게 작동하는지 이해하려면 볼륨 수요와 가용 IOPS 수량, I/O 작업을 완료하는 데 소요되는 시간, 볼륨의 처리량 제한 사이의 관계를 아는 것이 중요합니다.

IOPS

IOPS는 초당 입력/출력 작업을 나타내는 측정 단위입니다. 작업은 KiB 단위로 측정되며, 기본 드라이브 기술에 따라 볼륨 유형이 단일 I/O로 계산하는 최대 데이터 용량이 결정됩니다. I/O 크기의 최대 한도는 SSD 볼륨이 256KiB, HDD 볼륨이 1,024KiB이며 이렇게 차이가 나는 이유는 SSD 볼륨이 HDD 볼륨보다 소용량 또는 랜덤 I/O를 훨씬 더 효과적으로 처리하기 때문입니다.

소용량 I/O 작업이 물리적으로 연속되는 경우, Amazon EBS가 최대 크기까지 단일 I/O 작업으로 병합을 시도합니다. 예를 들어, SSD 볼륨의 경우 1건의 1,024KiB I/O 작업은 4건($1,024 \div 256 = 4$)의 작업으로 계산되지만 각각 32KiB인 8건의 연속하는 I/O 작업은 1건($8 \times 32 = 256$)의 작업으로 계산됩니다. 하지만 각각 32KiB인 8건의 랜덤 불연속 I/O 작업은 8건의 작업으로 계산됩니다. 이 경우 각각 32KiB보다 작은 I/O 작업은 1건의 작업으로 계산됩니다.

마찬가지로, HDD 지원 볼륨의 경우 1,024KiB I/O 작업 1건과 128KiB 순차 작업 8건은 모두 각각 1건의 작업으로 계산됩니다. 하지만 랜덤 128KiB I/O 작업 8건은 8건의 작업으로 계산됩니다.

그러므로 3,000 IOPS를 지원하는 SSD 지원 볼륨을 생성하여(io1 또는 io2 볼륨을 3,000 IOPS에서 프로비저닝하거나 gp2 볼륨의 크기를 1,000GiB에서 조정하는 방법으로), 충분한 대역폭을 제공할 수 있는 EBS 최적화 인스턴스에 연결할 경우, 초당 최대 3,000 I/O 데이터 전송이 가능하며 처리량은 I/O 크기에 따라 결정됩니다.

볼륨 대기열 길이 및 지연 시간

볼륨 대기열 길이는 디바이스에 대해 보류 중인 I/O 요청 수입니다. 지연 시간은 I/O 작업의 실제 종단 간 클라이언트 시간입니다. 다시 말해 EBS로 I/O를 전송한 후 EBS로부터 I/O 읽기 또는 쓰기가 완료되었다는 승인을 받기까지 소요된 시간입니다. 대기열 길이를 I/O 크기 및 지연 시간에 따라 정확히 보정하여, 게스트 운영 체제나 EBS로 연결되는 네트워크 링크에 병목 현상이 발생하지 않도록 해야 합니다.

최적의 대기열 길이는 워크로드마다 다른데, IOPS 및 지연 시간에 대한 특정 애플리케이션의 민감도에 따라 결정됩니다. 워크로드가 EBS 볼륨에 대해 사용 가능한 성능을 전부 사용할 만큼 충분한 I/O 요청을 제공하지 않는 경우, 프로비저닝된 처리량이나 IOPS를 볼륨이 제공하지 못할 수 있습니다.

트랜잭션 집약적 애플리케이션은 I/O 지연 시간 증가에 민감하며, SSD 지원 io1, io2 및 gp2 볼륨에 적합합니다. 대기열 길이를 줄이고 볼륨에서 사용할 수 있는 IOPS 개수를 늘리면 높은 IOPS를 유지하는 동시에 지연 시간을 단축할 수 있습니다. 볼륨이 수용할 수 있는 수준보다 높은 IOPS를 계속 구동하면 I/O 지연 시간이 길어질 수 있습니다.

처리량 집약적인 애플리케이션은 I/O 지연 시간 증가에 덜 민감하며, HDD 지원 st1 및 sc1 볼륨에 적합합니다. 대용량 순차 I/O를 수행할 때 대기열 길이를 길게 유지하면 HDD 지원 볼륨에서 높은 처리량을 유지할 수 있습니다.

I/O 크기 및 볼륨 처리량 제한이 없음

SSD 지원 볼륨의 경우, I/O 크기가 매우 큼면 볼륨 처리량 제한에 도달하기 때문에 프로비저닝한 것보다 IOPS가 적을 수 있습니다. 예를 들어, 버스트 크레딧을 사용할 수 있는 1,000GiB 미만의 gp2 볼륨에는 3,000 IOPS 제한과 250MiB/s의 볼륨 처리량 제한이 있습니다. 256KiB I/O 크기를 사용하는 경우, 볼륨은 1000 IOPS에서 처리량 제한에 도달합니다($1000 \times 256\text{KiB} = 250\text{MiB}$). I/O 크기가 작다면(예: 16KiB), 처리량이 250MiB/s에 훨씬 못 미치기 때문에 동일한 볼륨이 3,000 IOPS를 유지할 수 있습니다. (이 예제는 볼륨의 I/O 가 인스턴스의 처리량 제한에 도달하지 않는다고 가정합니다.) 각 EBS 볼륨 유형의 처리량 제한에 대한 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오.

소용량 I/O 작업의 경우, 인스턴스 내에서 측정했을 때 프로비저닝된 IOPS 값보다 큰 값을 관찰할 수 있습니다. 인스턴스 운영 체제가 소용량 I/O 작업을 Amazon EBS로 전달하기 전 대용량 작업에 병합할 때 이런 결과가 발생합니다.

워크로드가 HDD 지원 st1 및 sc1 볼륨의 순차 I/O를 사용한다면 인스턴스 내에서 측정했을 때 예상보다 높은 IOPS를 관찰할 수 있습니다. 인스턴스 운영 체제가 순차 I/O를 병합하고 1,024KiB 크기 단위로 계산되는 경우에 이런 결과가 발생합니다. 워크로드가 소용량 또는 랜덤 I/O를 사용하는 경우 예상보다 적은 처리량을 관찰할 수 있습니다. 이는 각각의 비순차적인 랜덤 I/O를 총 IOPS 계산에 적용하기 때문이며, 이로 인해 예상보다 일찍 볼륨의 IOPS 제한에 도달할 수 있습니다.

EBS 볼륨 유형이 무엇이든 현재 구성에서 기대한 IOPS 또는 처리량을 달성하지 못할 경우에는 EC2 인스턴스 대역폭이 제한 요소가 아닌지 확인하십시오. 최적의 성능을 위해 항상 현재 세대 EBS 최적화 인스턴스(또는 10Gb/s 네트워크 연결을 포함한 인스턴스)를 사용해야 합니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오. EBS 볼륨에 충분한 I/O를 구동하고 있지 않은 경우에도 IOPS가 예상과 다를 수 있습니다.

CloudWatch를 사용하여 I/O 특성 모니터링

각 볼륨의 [CloudWatch 볼륨 지표 \(p. 1077\)](#)로 이러한 I/O 특성을 모니터링할 수 있습니다. 고려해야 할 중요한 지표는 다음과 같습니다.

- `BurstBalance`
- `VolumeReadBytes`
- `VolumeWriteBytes`
- `VolumeReadOps`
- `VolumeWriteOps`
- `VolumeQueueLength`

`BurstBalance`는 gp2, st1, sc1 볼륨에 대한 버스트 버킷 잔고를 남은 잔고에 대한 비유로 표시합니다. 버스트 버킷이 모두 사용되면 볼륨 I/O(gp2 볼륨용) 또는 볼륨 처리량(st1 및 sc1 볼륨용)이 기준 수준으로 스로틀링됩니다. `BurstBalance` 값을 확인하여 이런 이유로 볼륨이 조절되는지 판단합니다.

HDD 지원 st1 및 sc1 볼륨은 1,024KiB 최대 I/O 크기를 활용하는 워크로드에서 가장 잘 작동하도록 설계되었습니다. 볼륨의 평균 I/O 크기를 구하려면 `VolumeWriteBytes` 를 `VolumeWriteOps` 값으로 나눕니다. 읽기 작업에도 같은 계산 방법이 적용됩니다. 평균 I/O 크기는 64KiB 미만이며, st1 또는 sc1 볼륨으로 보내는 I/O 작업의 크기가 큰 경우 성능을 개선해야 합니다.

Note

평균 I/O 크기가 44KiB이거나 그에 가까운 경우에는 간접 서술자가 지원되지 않는 인스턴스 또는 커널을 사용 중일 수 있습니다. 현재 세대 인스턴스뿐만 아니라 Linux 커널 3.8 이상 버전도 모두 이 지원을 제공합니다.

I/O 지연 시간이 필요한 것보다 긴 경우 `VolumeQueueLength`을 확인하여 애플리케이션이 프로비저닝한 것보다 많은 IOPS를 구동하려고 하고 있지 않은지 확인하십시오. 볼륨이 제공할 수 있는 것보다 많은 수의 IOPS를 요구하는 애플리케이션인 경우, 기준 성능 수준이 높은 대용량 gp2 볼륨 또는 프로비저닝된 IOPS가 더 많은 io1 또는 io2 볼륨을 사용하여 지연 시간을 줄이는 것을 고려해야 합니다.

관련 리소스

Amazon EBS I/O 특성에 관한 자세한 내용은 [Amazon EBS: Designing for Performance re:Invent 발표](#)를 참조하십시오.

Amazon EBS 볼륨 초기화

빈 EBS 볼륨은 생성되었지만 초기화(이전에는 사전 위밍이라고 함)가 필요하지 않은 시점에 최고 성능을 발휘합니다.

스냅샷에서 생성된 볼륨의 경우, 스토리지 블록에 액세스하려면 먼저 스토리지 블록을 Amazon S3에서 풀다운하고 볼륨에 기록해야 합니다. 이 예비 작업에는 시간이 걸리며, 이로 인해 각 블록에 처음 액세스할 때 I/O 작업의 지연 시간이 상당히 증가할 수 있습니다. 모든 블록을 다운로드하고 볼륨에 기록한 후에 볼륨 성능이 구현됩니다.

Important

스냅샷에서 생성된 io1 및 io2 볼륨을 초기화할 경우 볼륨의 성능이 예상 수준보다 50퍼센트 이하로 떨어질 수 있으며, 이로 인해 볼륨에서 I/O 성능 상태 확인에 대해 warning 상태를 표시할 수 있습니다. 이는 원래 그런 것이므로 초기화 중에는 io1 및 io2 볼륨에 대한 warning 상태를 무시해도 됩니다. 자세한 내용은 [EBS 볼륨 상태 확인 \(p. 958\)](#) 단원을 참조하십시오.

대부분의 애플리케이션은 볼륨 수명 주기 동안 초기화 비용을 분할 상환할 수 있습니다. 프로덕션 환경에서 이 초기 성능 저하를 방지하려면 다음 옵션 중 하나를 사용할 수 있습니다.

- 전체 볼륨을 강제로 즉시 초기화합니다. 자세한 내용은 [Windows에서 Amazon EBS 볼륨 초기화 \(p. 1067\)](#) 단원을 참조하십시오.
- 스냅샷에서 빠른 스냅샷 복원을 활성화하여 스냅샷에서 생성된 EBS 볼륨이 생성 시 완전히 초기화되고 모든 프로비저닝된 성능을 즉시 제공하도록 보장합니다. 자세한 내용은 [Amazon EBS 빠른 스냅샷 복원 \(p. 1044\)](#) 단원을 참조하십시오.

Windows에서 Amazon EBS 볼륨 초기화

새 EBS 볼륨은 사용 가능하지만 초기화(이전에는 사전 위밍이라고 함)가 필요하지 않은 시점에 최고 성능을 발휘합니다. 스냅샷에서 생성된 볼륨의 경우 Windows용 dd 또는 fio를 사용하여 볼륨의 모든 블록에서 읽습니다. 볼륨의 기존 데이터는 모두 보존됩니다.

Linux에서 Amazon EBS 볼륨 초기화에 대한 자세한 내용은 [Linux에서 Amazon EBS 볼륨 초기화](#)를 참조하십시오.

어느 도구든 사용하기 전에 다음과 같이 시스템의 디스크에 관한 정보를 수집하십시오.

1. wmic 명령을 사용하여 시스템에서 사용 가능한 디스크를 나열합니다.

```
wmic diskdrive get size,deviceid
```

다음은 예제 출력입니다.

DeviceID	Size
\\.\PHYSICALDRIVE2	80517265920
\\.\PHYSICALDRIVE1	80517265920
\\.\PHYSICALDRIVE0	128849011200
\\.\PHYSICALDRIVE3	107372805120

2. dd 또는 fio를 사용하여 초기화할 디스크를 식별합니다. c: 드라이브는 \\.\PHYSICALDRIVE0에 있습니다. 어떤 드라이브 번호를 사용해야 하는지 확실하지 않은 경우 diskmgmt.msc 유ти리티를 사용하여 드라이브 문자를 디스크 드라이브 번호와 비교합니다.

dd 사용

다음 절차를 완료하여 dd를 설치하고 사용하여 볼륨을 초기화합니다.

Note

이 단계는 EC2 인스턴스 대역폭, 볼륨에 대해 프로비저닝된 IOPS 및 볼륨 크기에 따라 몇 분에서 몇 시간까지 걸릴 수 있습니다.

Windows용 dd 설치

Windows용 dd 프로그램은 Linux 및 Unix 시스템에 일반적으로 사용할 수 있는 dd 프로그램과 비슷한 환경을 제공하며, 이 프로그램을 사용하여 스냅샷에서 생성된 Amazon EBS 볼륨을 초기화할 수 있습니다. 현재 최신 베타 버전에는 스냅샷에서 생성된 볼륨을 초기화하는 데 필요한 /dev/null 가상 디바이스가 포함되어 있습니다. 프로그램에 대한 전체 설명서는 <http://www.chrysocome.net/dd>에서 제공됩니다.

1. 최신 바이너리 버전의 Windows용 dd를 <http://www.chrysocome.net/dd>에서 다운로드합니다. 볼륨을 초기화하려면 버전 0.6 베타 3 이상을 사용해야 합니다.
2. (선택 사항) C:\bin과 같이 찾기 쉽고 기억하기 쉬운 명령줄 유ти리티용 폴더를 만듭니다. 명령줄 유ти리티용으로 지정된 폴더가 이미 있는 경우 다음 단계에서 해당 폴더를 대신 사용할 수 있습니다.
3. 바이너리 패키지의 압축을 풀고 dd.exe 파일을 명령줄 유ти리티 폴더(예: C:\bin)에 복사합니다.
4. 명령줄 유ти리티 폴더를 경로 환경 변수에 추가합니다 그러면 해당 폴더에 있는 프로그램을 어디서나 실행할 수 있습니다.

Important

다음 단계를 수행해도 현재 명령 프롬프트 창에서 환경 변수가 업데이트되지 않습니다. 이 단계를 완료한 후 새로 여는 명령 프롬프트 창에 업데이트가 포함됩니다. 따라서 환경이 제대로 설정되었는지 확인하려면 새 명령 프롬프트 창을 열어야 합니다.

- a. 시작(Start)을 선택하고 컴퓨터(Computer)에서 컨텍스트(오른쪽 클릭) 메뉴를 연 후, 속성(Properties)을 선택합니다.

- b. 고급 시스템 설정(Advanced system settings), 환경 변수(Environment Variables)를 선택합니다.
- c. 시스템 변수(System Variables)에서 경로(Path) 변수를 선택하고 편집(Edit)을 선택합니다.
- d. 변수 값(Variable value)에서 세미콜론과 명령줄 유ти리티 폴더의 위치(;c:\bin\))을 기존 값 끝에 추가합니다.
- e. 확인(OK)을 선택하여 시스템 변수 편집(Edit System Variable) 창을 닫습니다.

Windows용 dd를 사용하여 볼륨을 초기화하려면

1. 다음 명령을 실행하여 지정된 디바이스에 있는 모든 블록을 읽고 출력을 /dev/null 가상 디바이스에 전송합니다. 이 명령은 기존 데이터를 안전하게 초기화합니다.

Important

dd를 잘못 사용하면 볼륨 데이터가 쉽게 삭제될 수 있습니다. 아래 예제 명령을 정확하게 따라야 합니다. 읽고 있는 디바이스의 이름에 따라 if=\\.\PHYSICALDRIVE n 파라미터만 다를 수 있습니다.

```
dd if=\\.\PHYSICALDRIVE $n$  of=/dev/null bs=1M --progress --size
```

Note

dd가 볼륨의 끝을 지나 읽기를 시도할 경우 오류가 발생할 수 있습니다. 이 오류는 무시해도 됩니다.

2. 작업이 완료되면 새 볼륨을 사용할 준비가 된 것입니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오.

사용 fio

다음 절차를 완료하여 fio를 설치하고 사용하여 볼륨을 초기화합니다.

Windows용 fio를 설치하려면

Windows용 fio 프로그램은 Linux 및 Unix 시스템에 일반적으로 사용할 수 있는 fio 프로그램과 비슷한 환경을 제공하며, 이 프로그램을 사용하여 스냅샷에서 생성된 Amazon EBS 볼륨을 초기화할 수 있습니다. 자세한 내용은 <https://github.com/axboe/fio>를 참조하십시오.

1. [fio MSI 설치 프로그램](#)(최신 x86 또는 x64 빌드를 선택한 다음 결과물 선택)을 다운로드합니다.
2. fio를 설치합니다.

Windows용 fio를 사용하여 볼륨을 초기화하려면

1. 다음과 비슷한 명령을 실행하여 볼륨을 초기화합니다.

```
fio --filename=\\.\PHYSICALDRIVE $n$  --rw=read --bs=128k --iodepth=32 --direct=1 --name=volume-initialize
```

2. 작업이 완료되면 새 볼륨을 사용할 준비가 된 것입니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오.

Windows의 RAID 구성

Amazon EBS를 사용하면 기존 운영 체제 미설치 서버에서 사용 가능한 스탠다드 RAID 구성을 사용할 수 있습니다. 단, 해당 RAID 구성이 인스턴스에 대한 운영 체제에서 지원되어야 합니다. 이는 모든 RAID가 소프트웨어 수준에서 실행되기 때문입니다. 단일 볼륨을 사용할 때보다 더 나은 I/O 성능을 얻기 위해 RAID 0은 여

러 볼륨을 함께 스트라이프할 수 있고, 온인스턴스 중복을 위해 RAID 1은 두 개의 볼륨을 함께 미러링할 수 있습니다.

Amazon EBS 볼륨 데이터는 단일 구성 요소의 고장으로 인한 데이터 손실을 방지하기 위해 가용 영역의 여러 서버에 복제됩니다. 이 복제 기능으로 인해 Amazon EBS 볼륨이 일반 상용 디스크 드라이브보다 10배 더 안정적입니다. 자세한 내용은 Amazon EBS 제품 정보 페이지의 [Amazon EBS 가용성 및 내구성 단원](#)을 참조하십시오.

Note

RAID 볼륨에서는 부팅하지 않아야 합니다. 디바이스 중 하나에 장애가 발생하면 운영 체제를 부팅하지 못할 수 있습니다.

Linux 인스턴스에서 RAID 어레이를 생성해야 하는 경우 Linux 인스턴스용 Amazon EC2 사용 설명서의 [RAID Configuration on Linux](#)를 참조하십시오.

내용

- [RAID 구성 옵션 \(p. 1069\)](#)
- [Windows에서 RAID 어레이 생성 \(p. 1070\)](#)
- [RAID 배열에 볼륨 스냅샷 생성 \(p. 1073\)](#)

RAID 구성 옵션

다음 표에서는 일반 RAID 0 옵션과 RAID 1 옵션을 비교합니다.

구성	--set-visible-to-all-users	장점	단점
RAID 0	I/O 성능이 내결함성보다 더 중요한 경우(예: 데이터 복제가 이미 별도로 설정되어 있는 사용량이 많은 데이터베이스)	I/O가 스트라이프의 볼륨에 분산됩니다. 볼륨을 추가하면 처리량 및 IOPS도 그에 따라 바로 추가됩니다.	스트라이프의 성능이 세트에서 가장 성능이 떨어지는 볼륨으로 제한됩니다. 단일 볼륨이 손실되면 어레이의 데이터가 완전히 손실됩니다.
RAID 1	내결함성이 I/O 성능보다 더 중요한 경우(예: 중요 애플리케이션)	데이터 내구성 면에서 더 안전합니다.	쓰기 성능이 향상되지 않습니다. 또한 데이터를 동시에 여러 볼륨에 쓰기 때문에 비 RAID 구성에 비해 Amazon EC2와 Amazon EBS 사이에 더 큰 대역폭이 필요합니다.

Important

RAID 5 및 RAID 6는 이 RAID 모드의 패리티 쓰기 작업에서 볼륨에 사용 가능한 IOPS의 일부를 사용하기 때문에 Amazon EBS에 권장되지 않습니다. RAID 어레이의 구성에 따라 이러한 RAID 모드에서는 RAID 0 구성보다 20-30% 더 적은 가용 IOPS를 제공합니다. 이러한 RAID 모드는 비용 증가의 한 요인이기도 합니다. 볼륨 크기와 속도가 동일할 경우 2 볼륨 RAID 0 어레이가 두 배 더 비싼 4 볼륨 RAID 6 어레이보다 더 우수한 성능을 제공합니다.

RAID 0 어레이를 생성하면 단일 Amazon EBS 볼륨에서 프로비저닝할 때보다 파일 시스템의 성능이 더 향상됩니다. RAID 1 어레이는 중복성 강화를 위해 데이터를 "미러링"합니다. 이 절차를 수행하기 전에 RAID 어레이의 크기와 프로비저닝할 IOPS 수를 결정해야 합니다.

RAID 0 어레이의 결과 크기는 어레이 내 볼륨의 크기 합계이고, 대역폭은 어레이 내 볼륨의 가용 대역폭 합계입니다. RAID 1 어레이의 결과 크기 및 대역폭은 어레이 내 볼륨의 크기 및 대역폭과 같습니다. 예를 들어, 4,000의 프로비저닝된 IOPS가 있는 500GiB Amazon EBS io1 볼륨 2개는 각각 가용 대역폭이 8,000 IOPS이고 처리량이 1,000MiB/s인 1000GiB RAID 0 어레이를 생성하거나, 가용 대역폭이 4,000 IOPS이고 처리량이 500MiB/s인 500GiB RAID 1 어레이를 생성합니다.

이 문서는 기본 RAID 설정의 예를 제공합니다. RAID 구성, 성능 및 복구에 대한 자세한 내용은 https://raid.wiki.kernel.org/index.php/Linux_Raid에서 Linux RAID Wiki를 참조하십시오.

Windows에서 RAID 어레이 생성

다음 절차에 따라 RAID 어레이를 생성합니다. Linux 인스턴스에 대한 지침은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux에서 RAID 어레이 생성](#)을 참조하십시오.

Windows에서 RAID 어레이를 생성하려면

1. 어레이에 대한 Amazon EBS 볼륨을 생성합니다. 자세한 내용은 [Amazon EBS 볼륨 생성 \(p. 948\)](#) 단원을 참조하십시오.

Important

어레이에 대해 크기 및 IOPS 성능 값이 동일한 볼륨을 생성합니다. EC2 인스턴스의 사용 대역폭을 초과하는 어레이를 생성하지 마십시오.

2. 어레이를 호스팅할 인스턴스에 Amazon EBS 볼륨을 연결합니다. 자세한 내용은 [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#) 단원을 참조하십시오.
3. Windows 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
4. 명령 프롬프트를 열고 diskpart 명령을 입력합니다.

```
diskpart
```

```
Microsoft DiskPart version 6.1.7601
Copyright (C) 1999-2008 Microsoft Corporation.
On computer: WIN-BM6QPPL51CO
```

5. DISKPART 프롬프트에서 다음 명령을 사용하여 사용 가능한 디스크를 나열합니다.

```
DISKPART> list disk
```

Disk #	Status	Size	Free	Dyn	Gpt
Disk 0	Online	30 GB	0 B		
Disk 1	Online	8 GB	0 B		
Disk 2	Online	8 GB	0 B		
Disk 3	Online	8 GB	0 B		
Disk 4	Online	8 GB	0 B		
Disk 5	Online	419 GB	0 B		
Disk 6	Online	419 GB	0 B		

어레이에서 사용할 디스크를 식별하고 해당 디스크 번호를 기록해 둍니다.

6. 어레이에서 사용할 각 디스크는 기존 볼륨을 포함하지 않는 온라인 동적 디스크어야 합니다. 다음 단계에 따라 기본 디스크를 동적 디스크로 변환하고 기존 볼륨을 삭제합니다.
 - a. 다음 명령을 사용하여 어레이에서 사용할 디스크를 선택합니다. 여기에서 **n**을 디스크 번호로 대체합니다.

```
DISKPART> select disk n
```

```
Disk n is now the selected disk.
```

- b. 선택한 디스크가 **offline**으로 나열되는 경우 **online disk** 명령을 실행하여 온라인으로 전환합니다.
- c. 선택한 디스크에서 이전 **list disk** 명령 출력의 **Dyn** 열에 별표가 없는 경우 디스크를 동적 디스크로 전환해야 합니다.

```
DISKPART> convert dynamic
```

Note

디스크가 쓰기 금지되었다는 오류가 표시되는 경우 ATTRIBUTE DISK CLEAR READONLY 명령을 사용하여 읽기 전용 플래그를 지운 다음 동적 디스크 전환을 다시 시도할 수 있습니다.

- d. detail disk 명령을 사용하여 선택한 디스크의 기존 볼륨을 확인합니다.

```
DISKPART> detail disk

XENSRV PVIDISK SCSI Disk Device
Disk ID: 2D8BF659
Type : SCSI
Status : Online
Path : 0
Target : 1
LUN ID : 0
Location Path : PCIROOT(0)#PCI(0300)#SCSI(P00T01L00)
Current Read-only State : No
Read-only : No
Boot Disk : No
Pagefile Disk : No
Hibernation File Disk : No
Crashdump Disk : No
Clustered Disk : No

Volume ### Ltr Label FS Type Size Status Info
----- -- -- -- -- -- --
Volume 2 D NEW VOLUME FAT32 Simple 8189 MB Healthy
```

디스크의 볼륨 번호를 기록해 둡니다. 이 예제에서 볼륨 번호는 2입니다. 볼륨이 없는 경우 다음 단계를 건너뛸 수 있습니다.

- e. (이전 단계에서 볼륨이 식별된 경우에만 필요함) 이전 단계에서 식별된 디스크의 기존 볼륨을 선택하여 삭제합니다.

Warning

그러면 볼륨의 기존 데이터가 소멸됩니다.

- i. 볼륨을 선택합니다. 여기에서 **n**을 볼륨 번호로 대체합니다.

```
DISKPART> select volume n
Volume n is the selected volume.
```

- ii. 볼륨을 삭제합니다.

```
DISKPART> delete volume

DiskPart successfully deleted the volume.
```

- iii. 선택한 디스크에서 삭제할 각 볼륨에 대해 이 하위 단계를 반복합니다.

- f. 어레이에서 사용할 각 디스크에 대해 Step 6 (p. 1070)을 반복합니다.

7. 사용할 디스크가 새 동적 디스크인지 확인합니다.

```
DISKPART> list disk

Disk ### Status Size Free Dyn Gpt
```

Disk	Status	Size	Allocation	File System
Disk 0	Online	30 GB	0 B	
Disk 1	Online	8 GB	0 B	*
Disk 2	Online	8 GB	0 B	*
Disk 3	Online	8 GB	0 B	*
* Disk 4	Online	8 GB	0 B	*
Disk 5	Online	419 GB	0 B	
Disk 6	Online	419 GB	0 B	

8. RAID 어레이를 생성합니다. Windows에서는 RAID 0 볼륨을 스트라이프 볼륨이라고 하고, RAID 1 볼륨을 미러링된 볼륨이라고 합니다.

(스트라이프 볼륨에만 해당) 디스크 1과 2에서 스트라이프 볼륨 어레이를 생성하려면 다음 명령을 사용합니다(어레이를 스트라이프하려면 stripe 옵션 사용).

```
DISKPART> create volume stripe disk=1,2
DiskPart successfully created the volume.
```

(미러링된 볼륨에만 해당) 디스크 3과 4에서 미러링된 볼륨 어레이를 생성하려면 다음 명령을 사용합니다(어레이를 미러링하려면 mirror 옵션 사용).

```
DISKPART> create volume mirror disk=3,4
DiskPart successfully created the volume.
```

9. 새 볼륨을 확인합니다.

```
DISKPART> list volume

Volume ### Ltr Label Fs Type Size Status Info
----- -- -- -- -- -- -- -- --
Volume 0 C NTFS Partition 29 GB Healthy System
* Volume 1 RAW Mirror 8190 MB Healthy
Volume 2 RAW Stripe 15 GB Healthy
Volume 5 Z Temporary S NTFS Partition 419 GB Healthy
Volume 6 Y Temporary S NTFS Partition 419 GB Healthy
```

이 예제에서 Type 열에 Mirror 볼륨과 Stripe 볼륨이 나열됩니다.

10. 볼륨을 사용할 수 있도록 볼륨을 선택하여 포맷합니다.

- a. 포맷할 볼륨을 선택합니다. 여기에서 **n**을 볼륨 번호로 대체합니다.

```
DISKPART> select volume n
Volume n is the selected volume.
```

- b. 볼륨을 포맷합니다.

Note

전체 포맷을 수행하려면 quick 옵션을 생략합니다.

```
DISKPART> format quick recommended label="My new volume"
100 percent completed
DiskPart successfully formatted the volume.
```

- c. 사용 가능한 드라이브 문자를 볼륨에 배정합니다.

```
DISKPART> assign letter f
DiskPart successfully assigned the drive letter or mount point.
```

이제 새 볼륨을 사용할 준비가 되었습니다.

RAID 배열에 볼륨 스냅샷 생성

스냅샷을 사용하여 RAID 배열의 EBS 볼륨에 데이터를 백업하려는 경우 스냅샷이 일관되어야 합니다. 이러한 볼륨의 스냅샷은 둑립적으로 생성되기 때문입니다. 동기화되지 않은 스냅샷을 사용하여 RAID 배열의 EBS 볼륨을 복원할 경우 배열의 무결성이 손상됩니다.

RAID 배열에 일관된 스냅샷 집합을 생성하려면 [EBS 다중 볼륨 스냅샷](#)을 사용합니다. 다중 볼륨 스냅샷을 통해 EC2 인스턴스에 연결된 여러 EBS 볼륨에서 특정 시점, 데이터 조정 및 충돌 일치 스냅샷을 생성할 수 있습니다. 스냅샷은 여러 EBS 볼륨에서 자동으로 생성되기 때문에 일관성을 유지하기 위해 인스턴스를 중지하여 볼륨 간을 조정할 필요가 없습니다. 자세한 내용은 [Amazon EBS 스냅샷 만들기](#) 아래에서 다중 볼륨 스냅샷 만들기 단계를 참조하십시오.

EBS 볼륨 벤치마킹

I/O 워크로드를 시뮬레이션하여 Amazon EBS 볼륨의 성능을 테스트할 수 있습니다. 프로세스는 다음과 같습니다.

1. EBS에 최적화된 인스턴스 시작.
2. 새 EBS 볼륨을 생성합니다.
3. EBS에 최적화된 인스턴스에 볼륨 추가.
4. 블록 디바이스를 구성하고 마운트합니다.
5. I/O 성능 벤치마크를 위한 도구 설치.
6. 볼륨의 I/O 성능 벤치마크.
7. 요금이 계속 발생하지 않도록 볼륨 삭제 및 인스턴스 종료.

Important

일부 절차를 수행할 경우 자신이 벤치마크하는 EBS 볼륨에 있는 기존 데이터가 소멸되는 결과를 낳게 됩니다. 벤치마킹 절차는 프로덕션 볼륨이 아니라 테스트 목적으로 특별히 생성된 볼륨에 적용하기 위한 것입니다.

인스턴스 설정

EBS 볼륨에서 최적의 성능을 얻으려면 EBS에 최적화된 인스턴스를 사용하는 것이 좋습니다. EBS에 최적화된 인스턴스는 인스턴스와 함께 Amazon EC2와 Amazon EBS 사이의 전용 처리량을 제공합니다. EBS에 최적화된 인스턴스는 Amazon EC2 ~ Amazon EBS 간에 전용 대역폭을 전송하며, 인스턴스 유형에 따라 지정할 수 있습니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

EBS에 최적화된 인스턴스를 생성하려면 Amazon EC2 콘솔을 사용하여 인스턴스를 시작할 때 EBS 최적화 인스턴스로 시작(Launch as an EBS-Optimized instance)을 선택하거나 명령줄을 사용할 때 --ebs-optimized를 지정합니다. 이 옵션을 지원하는 현재 세대 인스턴스를 시작해야 합니다. 자세한 내용은 [Amazon EBS-최적화 인스턴스 \(p. 1049\)](#) 단원을 참조하십시오.

프로비저닝된 IOPS SSD(io1 및 io2) 볼륨 설정

io1 또는 io2 볼륨을 생성하려면 Amazon EC2 콘솔을 사용하여 볼륨 생성 시 프로비저닝된 IOPS SSD(**io1**) 또는 프로비저닝된 IOPS SSD(**io2**)를 선택하거나 명령줄에서 --volume-type io1|io2 --iops n 을

지정합니다. 여기서 n은 100~64,000 범위의 정수입니다. EBS 볼륨 사양에 대한 자세한 내용은 [Amazon EBS 볼륨 유형 \(p. 932\)](#) 단원을 참조하십시오. EBS 볼륨 생성에 대한 자세한 내용은 [Amazon EBS 볼륨 생성 \(p. 948\)](#) 단원을 참조하십시오. 인스턴스에 볼륨 연결에 대한 자세한 내용은 [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#) 단원을 참조하십시오.

처리량에 최적화된 HDD(st1) 또는 Cold HDD(sc1) 볼륨 설정

st1 볼륨을 생성하려면 Amazon EC2 콘솔을 사용하여 볼륨을 생성할 때 처리량에 최적화된 HDD를 선택하거나 명령줄을 사용할 때 --type **st1**을 지정합니다. sc1 볼륨을 생성하려면 Amazon EC2 콘솔을 사용하여 볼륨을 생성할 때 Cold HDD를 선택하거나 명령줄을 사용할 때 --type **sc1**을 지정합니다. EBS 볼륨 생성에 대한 자세한 내용은 [Amazon EBS 볼륨 생성 \(p. 948\)](#) 단원을 참조하십시오. 인스턴스에 이러한 볼륨 연결에 대한 자세한 내용은 [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#) 단원을 참조하십시오.

벤치마크 도구 설치

다음 표에는 EBS 볼륨의 성능을 벤치마크하기 위해 사용할 수 있는 도구 중 일부가 나열되어 있습니다.

도구	설명
DiskSpd	<p>DiskSpd는 Microsoft에서 Windows, Windows Server 및 Cloud Server Infrastructure 엔지니어링 팀의 스토리지 성능 도구입니다. https://gallery.technet.microsoft.com/DiskSpd-A-Robust-Storage-6ef84e62/file/199535/2/DiskSpd-2.0.21a.zip에서 해당 도구를 다운로드할 수 있습니다.</p> <p>diskspd.exe 실행 파일을 다운로드한 후 관리 권한으로 명령 프롬프트를 연 다음 ("관리자 권한으로 실행" 선택) diskspd.exe 파일을 복사한 디렉터리로 이동합니다.</p> <p>원하는 diskspd.exe 실행 파일을 적절한 실행 폴더(amd64fre, armfre 또는 x86fre)에서 C:\DiskSpd 같이 짧고 간단한 경로로 복사합니다. 대부분의 경우 amd64fre 폴더에서 64비트 버전의 DiskSpd를 사용해야 합니다.</p> <p>DiskSpd의 소스 코드는 GitHub에서 호스팅됩니다(https://github.com/Microsoft/diskspd)。</p>
CrystalDiskMark	CrystalDiskMark는 간단한 디스크 벤치마크 소프트웨어입니다. https://crystalmark.info/en/software/crystaldiskmark/ 에서 이 소프트웨어를 다운로드할 수 있습니다.

이러한 벤치마크 도구는 다양한 테스트 파라미터를 지원합니다. 볼륨이 지원하는 작업에 근접하는 명령을 사용해야 합니다. 아래 제공된 명령은 사용자가 시작하는 데 도움이 되는 예시입니다.

볼륨 대기열 길이 선택

워크로드와 볼륨 유형에 따라 최적의 볼륨 대기열 길이를 선택합니다

SSD 지원 볼륨에서 대기열 길이

SSD 지원 볼륨의 워크로드에 대한 최적의 대기열 길이를 확인하려면 사용 가능한 모든 1000 IOPS에 대해 대기열 길이를 1로 지정하는 것이 좋습니다(gp2 볼륨의 경우 기준 및 io1 및 io2 볼륨의 경우 프로비저닝된 양). 그러면 애플리케이션 성능을 모니터링하고 애플리케이션 요구 사항을 기준으로 해당 값을 조정할 수 있습니다.

대기열 길이를 길게 하면 프로비저닝된 IOPS, 처리량 또는 최적 시스템 대기열 길이 값(현재 32로 설정)을 얻을 때까지 유용합니다. 예를 들어 프로비저닝된 IOPS가 3,000인 볼륨은 대기열 길이 3을 목표로 해야 합니다. 이 값을 높이거나 낮추면서 튜닝을 시도하여 애플리케이션에 가장 적합한 설정을 찾아야 합니다.

HDD 지원 볼륨에서 대기열 길이

HDD 지원 볼륨에서 워크로드에 가장 적합한 대기열 길이를 알아내려면 1MiB 순차 I/O를 수행하는 동시에 최소 4 이상의 대기열 길이를 목표로 하는 것이 좋습니다. 그러면 애플리케이션 성능을 모니터링하고 애플리케이션 요구 사항을 기준으로 해당 값을 조정할 수 있습니다. 예를 들어 버스트 처리량은 500MiB/s, IOPS는 500인 2TiB st1 볼륨의 경우 1,024KiB, 512KiB 또는 256KiB 순차 I/O를 수행하는 동시에 각각 4, 8 또는 16 대기열 길이를 목표로 해야 합니다. 이 값을 높이거나 낮추면서 튜닝을 시도하여 애플리케이션에 가장 적합한 설정을 찾아야 합니다.

C 상태 비활성화

벤치마킹 실행 전에 프로세서 C 상태를 비활성화해야 합니다. 일시적으로 지원 CPU 내 유형 코어가 C 상태가 되어 전력을 절감할 수 있습니다. 코어가 호출되어 처리를 재개할 때 코어가 다시 완전히 작동하기까지 특정 시간이 흐릅니다. 이 지역 시간이 프로세서 벤치마킹 루틴을 방해할 수 있습니다. C 상태 및 이를 지원하는 EC2 인스턴스 유형에 대한 자세한 내용은 [EC2 인스턴스에 대한 프로세서 상태 제어](#)를 참조하십시오.

Windows에서 C 상태 비활성화

다음과 같이 Windows에서 C 상태를 비활성화할 수 있습니다.

- PowerShell에서 현재 활성 전력 체계를 가져옵니다.

```
C:\> $current_scheme = powercfg /getactivescheme
```

- 전력 체계 GUID를 가져옵니다.

```
C:\> (Get-WmiObject -class Win32_PowerPlan -Namespace "root\cimv2\power" -Filter "ElementName='High performance'").InstanceID
```

- 전력 설정 GUID를 가져옵니다.

```
C:\> (Get-WmiObject -class Win32_PowerSetting -Namespace "root\cimv2\power" -Filter "ElementName='Processor idle disable'").InstanceID
```

- 전력 설정 하위 그룹 GUID를 가져옵니다.

```
C:\> (Get-WmiObject -class Win32_PowerSettingSubgroup -Namespace "root\cimv2\power" -Filter "ElementName='Processor power management'").InstanceID
```

- 인덱스의 값을 1로 설정하여 C 상태를 비활성화합니다. 값이 0인 경우 C 상태가 비활성화되었음을 나타냅니다.

```
C:\> powercfg /setacvalueindex <power_scheme_guid> <power_setting_subgroup_guid> <power_setting_guid> 1
```

- 활성 체계를 설정하여 설정이 저장되었는지 확인합니다.

```
C:\> powercfg /setactive <power_scheme_guid>
```

벤치마킹 수행

다음 절차에서는 다양한 EBS 볼륨 유형에 대한 벤치마킹 명령을 설명합니다.

연결된 EBS 볼륨이 있는 EBS에 최적화된 인스턴스에서 다음 명령을 실행합니다. 스냅샷에서 EBS 볼륨을 생성한 경우, 반드시 벤치마킹 전에 초기화해야 합니다. 자세한 내용은 [Amazon EBS 볼륨 초기화 \(p. 1066\)](#) 단원을 참조하십시오.

볼륨 테스트를 마치면 정리에 도움이 되는 [Amazon EBS 볼륨 삭제 \(p. 966\)](#) 및 [인스턴스 종료 \(p. 454\)](#) 주제를 참조하십시오.

io1 및 io2 볼륨 벤치마킹

생성한 볼륨에서 DiskSpd를 실행합니다.

다음 명령은 T: 드라이브에 있는 20GB 테스트 파일(25% 쓰기 및 75% 읽기 비율, 8K 블록 크기)을 사용하여 30초 임의 I/O 테스트를 실행합니다. 각각 4개의 미해결 I/O와 1GB의 쓰기 엔트로피 값 시드가 있는 8개의 작업 스레드를 사용합니다. 테스트 결과는 DiskSpeedResults.txt라는 텍스트 파일에 저장됩니다. 이러한 파라미터는 SQL Server OLTP 워크로드를 시뮬레이션합니다.

```
diskspd -b8K -d30 -o4 -t8 -h -r -w25 -L -Z1G -c20G T:\iotest.dat > DiskSpeedResults.txt
```

결과를 해석하는 방법에 대한 자세한 내용은 [Inspecting disk IO performance with DiskSPD](#) 자습서를 참조하십시오.

Amazon EBS에 대한 Amazon CloudWatch 지표

CloudWatch 지표는 볼륨의 작동 동작을 살펴보고, 분석하고, 경보를 설정하는 데 사용할 수 있는 통계 데이터입니다.

다음 표에서는 Amazon EBS 볼륨에 사용할 수 있는 모니터링 데이터 유형을 설명합니다.

유형	설명
기본	자동으로 5분 기간 동안 데이터를 무료로 사용할 수 있습니다. 여기에는 EBS 기반 인스턴스의 루트 디바이스 볼륨에 대한 데이터가 포함됩니다.
세부	프로비저닝된 IOPS SSD(io1 및 io2) 볼륨이 1분 지표를 CloudWatch에 자동으로 보냅니다.

CloudWatch에서 데이터를 가져올 때 반환되는 데이터의 세부 수준을 지정하는 `Period` 요청 파라미터를 포함할 수 있습니다. 이 파라미터는 데이터를 수집할 때 사용하는 기간(5분 기간)과 같습니다. 반환되는 데이터가 유효하도록 요청의 기간을 수집 기간보다 길거나 같게 지정하는 것이 좋습니다.

CloudWatch API 또는 Amazon EC2 콘솔을 사용하여 데이터를 가져올 수 있습니다. 이 콘솔은 CloudWatch API에서 원시 데이터를 가져오고 데이터를 기반으로 일련의 그래프를 표시합니다. 필요에 따라 API의 데이터나 콘솔의 그래프를 사용할 수 있습니다.

Amazon EBS 지표

Amazon Elastic Block Store(Amazon EBS)에서는 여러 지표에 대한 데이터 요소를 CloudWatch로 전송합니다. Amazon EBS 범용 SSD(gp2), 처리량에 최적화된 HDD(st1), 콜드 HDD(sc1) 및 Magnetic(표준) 볼륨은 5분 지표를 CloudWatch에 자동으로 전송합니다. 프로비저닝된 IOPS SSD(io1 및 io2) 볼륨은 1분 지표를 CloudWatch에 자동으로 전송합니다. 볼륨이 인스턴스에 연결될 때만 CloudWatch에 데이터가 보고됩니다.

Nitro 기반 인스턴스에서는 이러한 지표 중 일부가 다릅니다. Nitro 시스템에 기초한 인스턴스 유형 목록은 [Nitro 시스템에 구축된 인스턴스 \(p. 118\)](#) 단원을 참조하십시오.

AWS/EBS 네임스페이스에는 다음 지표가 포함되어 있습니다.

Metrics

- [볼륨 지표 \(p. 1077\)](#)
- [빠른 스냅샷 복원 지표 \(p. 1080\)](#)

볼륨 지표

AWS/EBS 네임스페이스에는 EBS 볼륨에 대한 다음 지표가 포함되어 있습니다. 인스턴스의 운영 체제에서 사용 가능한 디스크 공간에 대한 자세한 내용은 [여유 디스크 공간 보기 \(p. 956\)](#) 단원을 참조하십시오.

지표	설명
VolumeReadBytes	<p>지정된 기간의 읽기 작업에 대한 정보를 제공합니다. Sum 통계는 해당 기간 동안 전송된 총 바이트 수를 보고합니다. Average 통계는 Nitro 기반 인스턴스에 연결된 볼륨을 제외하고 해당 기간 동안 각 읽기 작업의 평균 크기를 보고합니다. 여기에서 평균은 지정된 기간 동안의 평균을 나타냅니다. SampleCount 통계를 보면 해당 기간 동안 총 읽기 작업 수(Nitro 기반 인스턴스에 연결된 볼륨의 작업은 제외)를 알 수 있습니다. 여기에서 샘플 개수는 통계 계산 시 사용된 데이터 요소의 수를 나타냅니다. Xen 인스턴스의 경우 볼륨에서 읽기 작업이 있을 때에만 데이터가 보고됩니다.</p> <p>이 지표의 Minimum 및 Maximum 통계는 Nitro 기반 인스턴스에 연결된 볼륨에서만 지원됩니다.</p> <p>단위: 바이트</p>
VolumeWriteBytes	<p>지정된 기간의 쓰기 작업에 대한 정보를 제공합니다. Sum 통계는 해당 기간 동안 전송된 총 바이트 수를 보고합니다. Average 통계는 Nitro 기반 인스턴스에 연결된 볼륨을 제외하고 해당 기간 동안 각 쓰기 작업의 평균 크기를 보고합니다. 여기에서 평균은 지정된 기간 동안의 평균을 나타냅니다. SampleCount 통계를 보면 해당 기간 동안 총 쓰기 작업 수(Nitro 기반 인스턴스에 연결된 볼륨의 작업은 제외)를 알 수 있습니다. 여기에서 샘플 개수는 통계 계산 시 사용된 데이터 요소의 수를 나타냅니다. Xen 인스턴스의 경우 볼륨에서 쓰기 작업이 있을 때에만 데이터가 보고됩니다.</p> <p>이 지표의 Minimum 및 Maximum 통계는 Nitro 기반 인스턴스에 연결된 볼륨에서만 지원됩니다.</p> <p>단위: 바이트</p>
VolumeReadOps	<p>지정된 기간의 총 읽기 작업 수입니다.</p> <p>해당 기간의 초당 평균 읽기 작업 수(읽기 IOPS)를 계산하려면 해당 기간의 총 읽기 작업 수를 해당 기간의 초 수로 나누십시오.</p> <p>이 지표의 Minimum 및 Maximum 통계는 Nitro 기반 인스턴스에 연결된 볼륨에서만 지원됩니다.</p> <p>단위: 개수</p>
VolumeWriteOps	<p>지정된 기간의 총 쓰기 작업 수입니다.</p> <p>해당 기간의 초당 평균 쓰기 작업 수(쓰기 IOPS)를 계산하려면 해당 기간의 총 쓰기 작업 수를 해당 기간의 초 수로 나누십시오.</p> <p>이 지표의 Minimum 및 Maximum 통계는 Nitro 기반 인스턴스에 연결된 볼륨에서만 지원됩니다.</p> <p>단위: 개수</p>

지표	설명
<code>VolumeTotalReadTime</code>	<p>Note</p> <p>이 지표는 다중 연결 지원 볼륨에서 지원되지 않습니다.</p> <p>지정된 기간 동안 완료된 모든 읽기 작업에서 사용한 총 시간(초)입니다. 여러 요청이 동시에 제출된 경우 이 총계가 기간 길이보다 클 수 있습니다. 예를 들어, 5분(300초) 동안 700개의 작업이 완료되고 작업당 1초가 걸린 경우 값은 700초입니다. Xen 인스턴스의 경우 볼륨에서 읽기 작업이 있을 때에만 데이터가 보고됩니다.</p> <p>이 지표에 대한 <code>Average</code> 통계는 Nitro 기반 인스턴스에 연결된 볼륨과는 관련이 없습니다.</p> <p>이 지표의 <code>Minimum</code> 및 <code>Maximum</code> 통계는 Nitro 기반 인스턴스에 연결된 볼륨에서만 지원됩니다.</p> <p>단위: 초</p>
<code>VolumeTotalWriteTime</code>	<p>Note</p> <p>이 지표는 다중 연결 지원 볼륨에서 지원되지 않습니다.</p> <p>지정된 기간 동안 완료된 모든 쓰기 작업에서 사용한 총 시간(초)입니다. 여러 요청이 동시에 제출된 경우 이 총계가 기간 길이보다 클 수 있습니다. 예를 들어, 5분(300초) 동안 700개의 작업이 완료되고 작업당 1초가 걸린 경우 값은 700초입니다. Xen 인스턴스의 경우 볼륨에서 쓰기 작업이 있을 때에만 데이터가 보고됩니다.</p> <p>이 지표에 대한 <code>Average</code> 통계는 Nitro 기반 인스턴스에 연결된 볼륨과는 관련이 없습니다.</p> <p>이 지표의 <code>Minimum</code> 및 <code>Maximum</code> 통계는 Nitro 기반 인스턴스에 연결된 볼륨에서만 지원됩니다.</p> <p>단위: 초</p>
<code>VolumeIdleTime</code>	<p>Note</p> <p>이 지표는 다중 연결 지원 볼륨에서 지원되지 않습니다.</p> <p>지정된 기간 동안 읽기 또는 쓰기 작업이 제출되지 않은 총 시간(초)입니다.</p> <p>이 지표에 대한 <code>Average</code> 통계는 Nitro 기반 인스턴스에 연결된 볼륨과는 관련이 없습니다.</p> <p>이 지표의 <code>Minimum</code> 및 <code>Maximum</code> 통계는 Nitro 기반 인스턴스에 연결된 볼륨에서만 지원됩니다.</p> <p>단위: 초</p>

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
EBS CloudWatch 지표

지표	설명
VolumeQueueLength	<p>지정된 기간 동안 완료 대기 중인 읽기 및 쓰기 작업 요청 수입니다.</p> <p>이 지표에 대한 Sum 통계는 Nitro 기반 인스턴스에 연결된 볼륨과는 관련이 없습니다.</p> <p>이 지표의 Minimum 및 Maximum 통계는 Nitro 기반 인스턴스에 연결된 볼륨에서만 지원됩니다.</p> <p>단위: 개수</p>
VolumeThroughputPercentage	<p>Note</p> <p>이 지표는 다중 연결 지원 볼륨에서 지원되지 않습니다.</p> <p>프로비저닝된 IOPS SSD 볼륨에만 사용됩니다. Amazon EBS 볼륨에 대해 프로비저닝된 총 IOPS 중 전송된 IOPS(초당 I/O 작업 수)의 비율(%)입니다. 프로비저닝된 IOPS SSD 볼륨은 99.9% 시간 동안 프로비저닝된 성능을 제공합니다.</p> <p>쓰기 중 1분 동안 보류 중인 다른 I/O 요청이 없으면 지표 값이 100%가 됩니다. 또한 사용자가 취한 조치(예: 사용량 피크 시 볼륨 스냅샷 생성, EBS에 최적화되지 않은 인스턴스에서 볼륨 실행, 볼륨 데이터에 최초로 액세스 등)로 인해 볼륨의 I/O 성능이 일시적으로 저하될 수 있습니다.</p> <p>단위: 백분율</p>
VolumeConsumedReadWriteOps	<p>프로비저닝된 IOPS SSD 볼륨에만 사용됩니다. 지정된 시간 동안 소비한 총 읽기 및 쓰기 작업량(256,000 용량 단위로 정규화됨)입니다.</p> <p>256,000보다 작은 I/O 작업은 각각 1개의 소비 IOPS로 계산되고, 256,000보다 큰 I/O 작업은 256,000 용량 단위로 계산됩니다. 예를 들어, 1,024,000 I/O는 소비 IOPS 4개로 계산됩니다.</p> <p>단위: 개수</p>
BurstBalance	<p>범용 SSD(gp2), 처리량에 최적화된 HDD(st1) 및 Cold HDD(sc1) 볼륨에만 사용됩니다. 버스트 버킷에 남아 있는 I/O 크레딧(gp2의 경우)의 비율 또는 처리량 크레딧의 비율(st1 및 sc1의 경우) 정보를 보여 줍니다. 볼륨이 활성 상태일 때만 CloudWatch에 데이터가 보고되고, 볼륨이 연결되지 않은 경우에는 데이터가 보고되지 않습니다.</p> <p>이 지표에 대한 Sum 통계는 Nitro 기반 인스턴스에 연결된 볼륨과는 관련이 없습니다.</p> <p>볼륨의 기준 성능이 최대 버스트 성능을 초과하면 크레딧은 사용되지 않습니다. 볼륨이 Nitro 기반 인스턴스에 연결된 경우 버스트 잔고가 보고되지 않습니다. 비 Nitro 기반 인스턴스의 경우 보고된 버스트 잔고는 100%입니다. 자세한 내용은 I/O 크레딧 및 버스트 성능 (p. 934) 단원을 참조하십시오.</p> <p>단위: 백분율</p>

빠른 스냅샷 복원 지표

AWS/EBS 네임스페이스에는 [빠른 스냅샷 복원 \(p. 1044\)](#)에 대한 다음 지표가 포함되어 있습니다.

지표	설명
FastSnapshotRestoreCreditsBucket	생성할 수 있는 최대 볼륨 생성 크레딧 수. 이 지표는 가용 영역당 스냅샷별로 보고됩니다. 가장 유용한 통계는 Average입니다. Minimum 및 Maximum 통계의 결과는 Average의 통계와 동일하며 대신 사용될 수 있습니다.
FastSnapshotRestoreCreditsBalance	사용 가능한 볼륨 생성 크레딧 수. 이 지표는 가용 영역당 스냅샷별로 보고됩니다. 가장 유용한 통계는 Average입니다. Minimum 및 Maximum 통계의 결과는 Average의 통계와 동일하며 대신 사용될 수 있습니다.

Amazon EBS 지표 차원

지원되는 차원은 볼륨 ID(`VolumeId`)입니다. 사용 가능한 모든 통계는 볼륨 ID로 필터링됩니다.

[볼륨 지표 \(p. 1077\)](#)의 경우 지원되는 차원은 볼륨 ID(`VolumeId`)입니다. 사용 가능한 모든 통계는 볼륨 ID로 필터링됩니다.

[빠른 스냅샷 복원 지표 \(p. 1080\)](#)의 경우 지원되는 차원은 스냅샷 ID(`SnapshotId`) 및 가용 영역(`AvailabilityZone`)입니다.

Amazon EC2 콘솔의 그래프

볼륨을 생성한 후 Amazon EC2 콘솔로 가서 볼륨의 모니터링 그래프를 볼 수 있습니다. 콘솔의 볼륨 페이지에서 볼륨을 선택하고 모니터링을 선택합니다. 다음 표에는 표시되는 그래프가 나열되어 있습니다. 오른쪽 열에는 CloudWatch API의 원시 데이터 측정치로 각 그래프가 생성되는 방법이 설명되어 있습니다. 모든 그래프의 기간은 5분입니다.

그래프	원시 지표를 사용하여 설명
Read Bandwidth (KiB/s)	<code>Sum(VolumeReadBytes) / Period / 1024</code>
Write Bandwidth (KiB/s)	<code>Sum(VolumeWriteBytes) / Period / 1024</code>
읽기 처리 속도 (IOPS)	<code>Sum(VolumeReadOps) / Period</code>
쓰기 처리 속도 (IOPS)	<code>Sum(VolumeWriteOps) / Period</code>
평균 대기열 길이 (작업)	<code>Avg(VolumeQueueLength)</code>
% Time Spent Idle	<code>Sum(VolumeIdleTime) / Period * 100</code>
평균 읽기 크기 (KiB/작업)	<code>Avg(VolumeReadBytes) / 1024</code> Nitro 기반 인스턴스의 경우, 다음 공식에서 CloudWatch 지표 수식 을 사용하여 평균 읽기 크기를 도출합니다. <code>(Sum(VolumeReadBytes) / Sum(VolumeReadOps)) / 1024</code>

그래프	원시 지표를 사용하여 설명
	VolumeReadBytes 및 VolumeReadOps 지표는 EBS CloudWatch 콘솔에서 사용할 수 있습니다.
평균 쓰기 크기 (KiB/작업)	$\text{Avg}(\text{VolumeWriteBytes}) / 1024$ <p>Nitro 기반 인스턴스의 경우, 다음 공식에서 CloudWatch 지표 수식을 사용하여 평균 쓰기 크기를 도출합니다.</p> $(\text{Sum}(\text{VolumeWriteBytes}) / \text{Sum}(\text{VolumeWriteOps})) / 1024$ <p>VolumeWriteBytes 및 VolumeWriteOps 지표는 EBS CloudWatch 콘솔에서 사용할 수 있습니다.</p>
평균 읽기 지연 시간 (ms/작업)	$\text{Avg}(\text{VolumeTotalReadTime}) \times 1000$ <p>Nitro 기반 인스턴스의 경우, 다음 공식에서 CloudWatch 지표 수식을 사용하여 평균 읽기 지연 시간을 도출합니다.</p> $(\text{Sum}(\text{VolumeTotalReadTime}) / \text{Sum}(\text{VolumeReadOps})) \times 1000$ <p>VolumeTotalReadTime 및 VolumeReadOps 지표는 EBS CloudWatch 콘솔에서 사용할 수 있습니다.</p>
평균 쓰기 지연 시간 (ms/작업)	$\text{Avg}(\text{VolumeTotalWriteTime}) \times 1000$ <p>Nitro 기반 인스턴스의 경우, 다음 공식에서 CloudWatch 지표 수식을 사용하여 평균 쓰기 지연 시간을 도출합니다.</p> $(\text{Sum}(\text{VolumeTotalWriteTime}) / \text{Sum}(\text{VolumeWriteOps})) \times 1000$ <p>VolumeTotalWriteTime 및 VolumeWriteOps 지표는 EBS CloudWatch 콘솔에서 사용할 수 있습니다.</p>

평균 지연 시간 그래프 및 평균 크기 그래프의 경우 기간 중 완료된 총 작업(그래프에 해당하는 읽기 또는 쓰기) 수를 기준으로 평균을 계산합니다.

Amazon EBS용 Amazon CloudWatch Events

Amazon EBS는 Amazon CloudWatch Events를 기반으로 다양한 볼륨, 스냅샷 및 암호화 상태 변경에 대한 알림을 보냅니다. CloudWatch 이벤트에서는 볼륨, 스냅샷 또는 암호화 키 상태 변경에 대한 응답으로 프로그래밍 방식의 작업을 트리거하는 규칙을 수립할 수 있습니다. 예를 들어 스냅샷이 생성되면 AWS Lambda 함수를 트리거하여 완료된 스냅샷을 다른 계정과 공유하거나 재해 복구를 위해 다른 리전으로 복사할 수 있습니다.

CloudWatch의 이벤트는 JSON 객체로 표현됩니다. 이 이벤트에 고유한 필드는 JSON 객체의 "세부 정보" 섹션에 포함되어 있습니다. "이벤트" 필드에는 이벤트 이름이 포함됩니다. "결과" 필드에는 이벤트를 트리거한 작업의 완료 상태가 포함됩니다. 자세한 내용은 Amazon CloudWatch Events 사용 설명서에서 [CloudWatch 이벤트의 이벤트 패턴](#)을 참조하십시오.

자세한 정보는 Amazon CloudWatch 사용 설명서의 [이벤트 사용](#)을 참조하십시오.

목차

- [EBS 볼륨 이벤트 \(p. 1082\)](#)

- EBS 스냅샷 이벤트 (p. 1085)
- EBS 볼륨 수정 이벤트 (p. 1088)
- EBS 빠른 스냅샷 복원 이벤트 (p. 1089)
- AWS Lambda를 사용하여 CloudWatch 이벤트 처리 (p. 1090)

EBS 볼륨 이벤트

Amazon EBS는 다음과 같은 볼륨 이벤트가 발생하는 경우 CloudWatch 이벤트에 이벤트를 보냅니다.

이벤트

- 볼륨 생성(createVolume) (p. 1082)
- 볼륨 삭제(deleteVolume) (p. 1083)
- 볼륨 연결 또는 다시 연결(attachVolume, reattachVolume) (p. 1084)

볼륨 생성(createVolume)

볼륨 생성 작업이 완료되면 createVolume 이벤트가 AWS 계정으로 전송됩니다. 그러나 저장, 로깅, 아카이빙은 되지 않습니다. 이 이벤트에서 available 또는 failed 결과가 발생할 수 있습니다. 잘못된 KMS 키가 제공되면 아래 예와 같이 생성이 실패합니다.

이벤트 데이터

아래 목록에 성공적인 createVolume 이벤트에 대해 EBS가 발생시키는 JSON 객체의 예를 열거했습니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-012345678901",  
    "detail-type": "EBS Volume Notification",  
    "source": "aws.ec2",  
    "account": "012345678901",  
    "time": "yyyy-mm-ddThh:mm:ssZ",  
    "region": "us-east-1",  
    "resources": [  
        "arn:aws:ec2:us-east-1:012345678901:volume/vol-01234567"  
    ],  
    "detail": {  
        "result": "available",  
        "cause": "",  
        "event": "createVolume",  
        "request-id": "01234567-0123-0123-0123-0123456789ab"  
    }  
}
```

아래 목록에 createVolume 이벤트 실패 이후 EBS가 발생시키는 JSON 객체의 예를 열거했습니다. 실패의 원인은 비활성화된 KMS 키였습니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-0123456789ab",  
    "detail-type": "EBS Volume Notification",  
    "source": "aws.ec2",  
    "account": "012345678901",  
    "time": "yyyy-mm-ddThh:mm:ssZ",  
    "region": "sa-east-1",  
    "resources": [  
        "arn:aws:ec2:sa-east-1:012345678901:volume/vol-0123456789ab"  
    ]  
}
```

```
    "arn:aws:ec2:sa-east-1:0123456789ab:volume/vol-01234567",
],
"detail": {
    "event": "createVolume",
    "result": "failed",
    "cause": "arn:aws:kms:sa-east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab
is disabled.",
    "request-id": "01234567-0123-0123-0123-0123456789ab",
}
}
```

아래에 createVolume 이벤트 실패 이후 EBS가 발생시키는 JSON 객체의 예를 제시했습니다. 실패의 원인은 보류 중인 KMS 키 가져오기였습니다.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "EBS Volume Notification",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "sa-east-1",
    "resources": [
        "arn:aws:ec2:sa-east-1:0123456789ab:volume/vol-01234567",
    ],
    "detail": {
        "event": "createVolume",
        "result": "failed",
        "cause": "arn:aws:kms:sa-east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab
is pending import.",
        "request-id": "01234567-0123-0123-0123-0123456789ab",
    }
}
```

볼륨 삭제(deleteVolume)

볼륨 삭제 작업이 완료되면 deleteVolume 이벤트가 AWS 계정으로 전송됩니다. 그러나 저장, 로깅, 아카이빙은 되지 않습니다. 이 이벤트에 deleted 결과가 있습니다. 삭제가 완료되지 않으면 이벤트가 전송되지 않습니다.

이벤트 데이터

아래 목록에 성공적인 deleteVolume 이벤트에 대해 EBS가 발생시키는 JSON 객체의 예를 열거했습니다.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-012345678901",
    "detail-type": "EBS Volume Notification",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:012345678901:volume/vol-01234567"
    ],
    "detail": {
        "result": "deleted",
        "cause": "",
        "event": "deleteVolume",
        "request-id": "01234567-0123-0123-0123-0123456789ab"
    }
}
```

}

볼륨 연결 또는 다시 연결(attachVolume, reattachVolume)

볼륨이 인스턴스에 연결하지 못하거나 다시 연결하지 못하면 AWS 계정으로 attachVolume 또는 reattachVolume 이벤트가 전송됩니다. 그러나 저장, 로깅, 암호화는 되지 않습니다. KMS 키를 사용하여 EBS 볼륨을 암호화할 경우 키가 무효해지면, EBS는 해당 키가 나중에 인스턴스에 연결하거나 다시 연결하는 데 사용될 경우 이벤트를 발송합니다(아래 예 참조).

이벤트 데이터

아래 목록에 attachVolume 이벤트 실패 이후 EBS가 발생시키는 JSON 객체의 예를 열거했습니다. 실패의 원인은 보류 중인 KMS 키의 삭제였습니다.

Note

AWS가 정기 서버 유지 관리 후 볼륨에 다시 연결을 시도할 수 있습니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-0123456789ab",  
    "detail-type": "EBS Volume Notification",  
    "source": "aws.ec2",  
    "account": "012345678901",  
    "time": "yyyy-mm-ddThh:mm:ssZ",  
    "region": "us-east-1",  
    "resources": [  
        "arn:aws:ec2:us-east-1:0123456789ab:volume/vol-01234567",  
        "arn:aws:kms:us-east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab"  
    ],  
    "detail": {  
        "event": "attachVolume",  
        "result": "failed",  
        "cause": "arn:aws:kms:us-east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab  
is pending deletion.",  
        "request-id": ""  
    }  
}
```

아래 목록에 reattachVolume 이벤트 실패 이후 EBS가 발생시키는 JSON 객체의 예를 열거했습니다. 실패의 원인은 보류 중인 KMS 키의 삭제였습니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-0123456789ab",  
    "detail-type": "EBS Volume Notification",  
    "source": "aws.ec2",  
    "account": "012345678901",  
    "time": "yyyy-mm-ddThh:mm:ssZ",  
    "region": "us-east-1",  
    "resources": [  
        "arn:aws:ec2:us-east-1:0123456789ab:volume/vol-01234567",  
        "arn:aws:kms:us-east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab"  
    ],  
    "detail": {  
        "event": "reattachVolume",  
        "result": "failed",  
        "cause": "arn:aws:kms:us-east-1:0123456789ab:key/01234567-0123-0123-0123-0123456789ab  
is pending deletion.",  
        "request-id": ""  
    }  
}
```

EBS 스냅샷 이벤트

Amazon EBS는 다음과 같은 볼륨 이벤트가 발생하는 경우 CloudWatch 이벤트에 이벤트를 보냅니다.

이벤트

- [스냅샷 생성\(createSnapshot\) \(p. 1085\)](#)
- [스냅샷 생성\(createSnapshots\) \(p. 1085\)](#)
- [스냅샷 복사\(copySnapshot\) \(p. 1087\)](#)
- [스냅샷 공유\(shareSnapshot\) \(p. 1088\)](#)

스냅샷 생성(createSnapshot)

스냅샷 생성 작업이 완료되면 `createSnapshot` 이벤트가 AWS 계정으로 전송됩니다. 그러나 저장, 로깅, 아카이빙은 되지 않습니다. 이 이벤트에서 `succeeded` 또는 `failed` 결과가 발생할 수 있습니다.

이벤트 데이터

아래 목록에 성공적인 `createSnapshot` 이벤트에 대해 EBS가 발생시키는 JSON 객체의 예를 열거했습니다. `detail` 섹션에서 `source` 필드에는 소스 볼륨의 ARN이 들어 있습니다. `startTime` 및 `endTime` 필드는 스냅샷 생성이 시작된 시점과 완료된 시점을 나타냅니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-012345678901",  
    "detail-type": "EBS Snapshot Notification",  
    "source": "aws.ec2",  
    "account": "012345678901",  
    "time": "yyyy-mm-ddThh:mm:ssZ",  
    "region": "us-east-1",  
    "resources": [  
        "arn:aws:ec2:us-west-2::snapshot/snap-01234567"  
    ],  
    "detail": {  
        "event": "createSnapshot",  
        "result": "succeeded",  
        "cause": "",  
        "request-id": "",  
        "snapshot_id": "arn:aws:ec2:us-west-2::snapshot/snap-01234567",  
        "source": "arn:aws:ec2:us-west-2::volume/vol-01234567",  
        "startTime": "yyyy-mm-ddThh:mm:ssZ",  
        "endTime": "yyyy-mm-ddThh:mm:ssZ"    }  
}
```

스냅샷 생성(createSnapshots)

다중 볼륨 스냅샷 생성 작업이 완료되면 `createSnapshots` 이벤트가 AWS 계정으로 전송됩니다. 이 이벤트에서 `succeeded` 또는 `failed` 결과가 발생할 수 있습니다.

이벤트 데이터

아래 목록에 성공적인 `createSnapshots` 이벤트에 대해 EBS가 발생시키는 JSON 객체의 예를 열거했습니다. `detail` 섹션의 `source` 필드에는 다중 볼륨 스냅샷 세트의 소스 볼륨 ARN이 포함되어 있습니다. `startTime` 및 `endTime` 필드는 스냅샷 생성이 시작된 시점과 완료된 시점을 나타냅니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-012345678901",  
    "detail-type": "EBS Multi-Volume Snapshots Completion Status",  
    "source": "arn:aws:ec2:us-west-2::volume/vol-01234567",  
    "start_time": "2015-06-10T18:00:00Z",  
    "end_time": "2015-06-10T18:01:00Z",  
    "status": "succeeded",  
    "volume_ids": ["arn:aws:ec2:us-west-2::volume/vol-01234567"]  
}
```

```
"source": "aws.ec2",
"account": "012345678901",
"time": "yyyy-mm-ddThh:mm:ssZ",
"region": "us-east-1",
"resources": [
    "arn:aws:ec2::us-east-1:snapshot/snap-01234567",
    "arn:aws:ec2::us-east-1:snapshot/snap-012345678"
],
"detail": {
    "event": "createSnapshots",
    "result": "succeeded",
    "cause": "",
    "request-id": "",
    "startTime": "yyyy-mm-ddThh:mm:ssZ",
    "endTime": "yyyy-mm-ddThh:mm:ssZ",
    "snapshots": [
        {
            "snapshot_id": "arn:aws:ec2::us-east-1:snapshot/snap-01234567",
            "source": "arn:aws:ec2::us-east-1:volume/vol-01234567",
            "status": "completed"
        },
        {
            "snapshot_id": "arn:aws:ec2::us-east-1:snapshot/snap-012345678",
            "source": "arn:aws:ec2::us-east-1:volume/vol-012345678",
            "status": "completed"
        }
    ]
}
```

아래 목록에 createSnapshots 이벤트 실패 이후 EBS가 발생시키는 JSON 객체의 예를 열거했습니다. 실패의 원인은 하나 이상의 스냅샷을 완료하기 못했기 때문입니다. snapshot_id의 값은 실패한 스냅샷의 ARN입니다. startTime 및 endTime은 스냅샷 생성 작업이 시작 및 종료된 시기를 나타냅니다.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-012345678901",
    "detail-type": "EBS Multi-Volume Snapshots Completion Status",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2::us-east-1:snapshot/snap-01234567",
        "arn:aws:ec2::us-east-1:snapshot/snap-012345678"
    ],
    "detail": {
        "event": "createSnapshots",
        "result": "failed",
        "cause": "Snapshot snap-01234567 is in status deleted",
        "request-id": "",
        "startTime": "yyyy-mm-ddThh:mm:ssZ",
        "endTime": "yyyy-mm-ddThh:mm:ssZ",
        "snapshots": [
            {
                "snapshot_id": "arn:aws:ec2::us-east-1:snapshot/snap-01234567",
                "source": "arn:aws:ec2::us-east-1:volume/vol-01234567",
                "status": "error"
            },
            {
                "snapshot_id": "arn:aws:ec2::us-east-1:snapshot/snap-012345678",
                "source": "arn:aws:ec2::us-east-1:volume/vol-012345678",
                "status": "deleted"
            }
        ]
    }
}
```

```
    ]  
}
```

스냅샷 복사(copySnapshot)

스냅샷 복사 작업이 완료되면 copySnapshot 이벤트가 AWS 계정으로 전송됩니다. 그러나 저장, 로깅, 아카이빙은 되지 않습니다. 이 이벤트에서 succeeded 또는 failed 결과가 발생할 수 있습니다.

이벤트 데이터

아래 목록에는 copySnapshot 이벤트가 성공한 후 EBS가 전송하는 JSON 객체의 예가 나와 있습니다. snapshot_id의 값은 새로 생성된 스냅샷의 ARN입니다. detail 섹션에서 source의 값은 소스 스냅샷의 ARN입니다. startTime 및 endTime은 스냅샷 복사 작업이 시작된 시점과 종료된 시점을 나타냅니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-012345678901",  
    "detail-type": "EBS Snapshot Notification",  
    "source": "aws.ec2",  
    "account": "123456789012",  
    "time": "yyyy-mm-ddThh:mm:ssZ",  
    "region": "us-east-1",  
    "resources": [  
        "arn:aws:ec2:us-west-2::snapshot/snap-01234567"  
    ],  
    "detail": {  
        "event": "copySnapshot",  
        "result": "succeeded",  
        "cause": "",  
        "request-id": "",  
        "snapshot_id": "arn:aws:ec2:us-west-2::snapshot/snap-01234567",  
        "source": "arn:aws:ec2:eu-west-1::snapshot/snap-76543210",  
        "startTime": "yyyy-mm-ddThh:mm:ssZ",  
        "endTime": "yyyy-mm-ddThh:mm:ssZ",  
        "Incremental": "True"  
    }  
}
```

아래 목록에 copySnapshot 이벤트 실패 이후 EBS가 발생시키는 JSON 객체의 예를 열거했습니다. 실패의 원인은 잘못된 소스 스냅샷 ID였습니다. snapshot_id의 값은 실패한 스냅샷의 ARN입니다. detail 섹션에서 source의 값은 소스 스냅샷의 ARN입니다. startTime 및 endTime은 스냅샷 복사 작업이 시작된 시점과 종료된 시점을 나타냅니다.

```
{  
    "version": "0",  
    "id": "01234567-0123-0123-0123-012345678901",  
    "detail-type": "EBS Snapshot Notification",  
    "source": "aws.ec2",  
    "account": "123456789012",  
    "time": "yyyy-mm-ddThh:mm:ssZ",  
    "region": "us-east-1",  
    "resources": [  
        "arn:aws:ec2:us-west-2::snapshot/snap-01234567"  
    ],  
    "detail": {  
        "event": "copySnapshot",  
        "result": "failed",  
        "cause": "Source snapshot ID is not valid",  
        "request-id": "",  
        "snapshot_id": "arn:aws:ec2:us-west-2::snapshot/snap-01234567",  
        "source": "arn:aws:ec2:eu-west-1::snapshot/snap-76543210",  
    }  
}
```

```
        "startTime": "yyyy-mm-ddThh:mm:ssZ",
        "endTime": "yyyy-mm-ddThh:mm:ssZ"
    }
}
```

스냅샷 공유(shareSnapshot)

다른 계정이 스냅샷을 공유하면 shareSnapshot 이벤트가 AWS 계정으로 전송됩니다. 그러나 저장, 로깅, 아카이빙은 되지 않습니다. 결과는 항상 succeeded입니다.

이벤트 데이터

아래에 shareSnapshot 이벤트 완료 이후 EBS가 발생시키는 JSON 객체의 예를 열거했습니다. detail 섹션에서 source의 값은 스냅샷을 공유한 사용자의 AWS 계정 번호입니다. startTime 및 endTime은 스냅샷 공유 작업이 시작된 시점과 종료된 시점을 나타냅니다. shareSnapshot 이벤트는 프라이빗 스냅샷이 다른 사용자와 공유될 때만 발생합니다. 퍼블릭 스냅샷 공유는 이벤트를 트리거하지 않습니다.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-012345678901",
    "detail-type": "EBS Snapshot Notification",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-west-2::snapshot/snap-01234567"
    ],
    "detail": {
        "event": "shareSnapshot",
        "result": "succeeded",
        "cause": "",
        "request-id": "",
        "snapshot_id": "arn:aws:ec2:us-west-2::snapshot/snap-01234567",
        "source": "012345678901",
        "startTime": "yyyy-mm-ddThh:mm:ssZ",
        "endTime": "yyyy-mm-ddThh:mm:ssZ"
    }
}
```

EBS 볼륨 수정 이벤트

Amazon EBS는 볼륨이 수정될 때 CloudWatch 이벤트에 modifyVolume 이벤트를 보냅니다. 그러나 저장, 로깅, 아카이빙은 되지 않습니다.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-012345678901",
    "detail-type": "EBS Volume Notification",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:012345678901:volume/vol-03a55cf56513fa1b6"
    ],
    "detail": {
        "result": "optimizing",
        "cause": "",
        "event": "modifyVolume",
        "request-id": "01234567-0123-0123-0123-0123456789ab"
    }
}
```

```
}
```

EBS 빠른 스냅샷 복원 이벤트

Amazon EBS는 스냅샷에 대한 빠른 스냅샷 복원 상태가 변경되면 이벤트를 CloudWatch 이벤트로 보냅니다.

다음은 이 이벤트의 예제 데이터입니다.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-012345678901",
    "detail-type": "EBS Fast Snapshot Restore State-change Notification",
    "source": "aws.ec2",
    "account": "123456789012",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1::snapshot/snap-03a55cf56513fa1b6"
    ],
    "detail": {
        "snapshot-id": "snap-1234567890abcdef0",
        "state": "optimizing",
        "zone": "us-east-1a",
        "message": "Client.UserInitiated - Lifecycle state transition",
    }
}
```

가능한 state 값은 enabling, optimizing, enabled, disabling, disabled입니다.

message에 가능한 값은 다음과 같습니다.

Client.InvalidSnapshot.InvalidState – The requested snapshot transitioned to an invalid state (Error)

빠른 스냅샷 복원을 요청하는 요청이 실패하고 상태가 disabling 또는 disabled로 전환되었습니다.

빠른 스냅샷 복원은 이 스냅샷에 대해 활성화할 수 없습니다.

Client.UserInitiated

상태가 enabling 또는 disabling으로 전환되었습니다.

Client.UserInitiated - Lifecycle state transition

상태가 optimizing, enabled 또는 disabled로 전환되었습니다.

Server.InsufficientCapacity – There was insufficient capacity available to satisfy the request

불충분한 용량으로 인해 빠른 스냅샷 복원을 요청하는 요청이 실패하고 상태가 disabling 또는 disabled로 전환되었습니다. 잠시 기다렸다가 다시 시도하십시오.

Server.InternalError – An internal error caused the operation to fail

내부 오류로 인해 빠른 스냅샷 복원을 요청하는 요청이 실패하고 상태가 disabling 또는 disabled로 전환되었습니다. 잠시 기다렸다가 다시 시도하십시오.

Client.InvalidSnapshot.InvalidState – The requested snapshot was deleted or access permissions were revoked

스냅샷이 스냅샷 소유자에 의해 삭제되거나 공유 해제되었으므로 스냅샷에 대한 빠른 스냅샷 복원 상태가 disabling 또는 disabled로 전환되었습니다. 삭제되었거나 더 이상 본인에게 공유되지 않은 스냅샷에 대해서는 빠른 스냅샷 복원을 활성화할 수 없습니다.

AWS Lambda를 사용하여 CloudWatch 이벤트 처리

Amazon EBS와 CloudWatch 이벤트를 사용하여 데이터 백업 워크플로를 자동화할 수 있습니다. 이를 위해 IAM 정책, 이벤트를 처리할 AWS Lambda 함수, 수신 이벤트와 일치하는 Amazon CloudWatch Events 규칙을 생성하고 Lambda 함수로 라우팅해야 합니다.

다음 절차에서는 재해 복구를 위해 `createSnapshot` 이벤트를 사용하여 완료된 스냅샷을 다른 리전으로 자동으로 복사합니다.

완료된 스냅샷을 다른 리전으로 복사하려면

1. 다음 예에 표시된 것과 같은 IAM 정책을 생성하여 `CopySnapshot` 작업을 실행하고 CloudWatch 이벤트 로그에 쓸 수 있는 권한을 제공합니다. CloudWatch 이벤트를 처리할 IAM 사용자에게 정책을 할당합니다.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "logs>CreateLogGroup",  
                "logs>CreateLogStream",  
                "logs:PutLogEvents"  
            ],  
            "Resource": "arn:aws:logs:*:*:  
        },  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:CopySnapshot"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

2. CloudWatch 콘솔에서 사용할 수 있는 Lambda 함수를 정의합니다. 아래 Node.js로 작성된 샘플 Lambda 함수는 일치하는 `createSnapshot` 이벤트를 Amazon EBS가 발생시킬 때 CloudWatch에 의해 호출됩니다. 이는 스냅샷이 완료되었음을 의미합니다. 호출되면 함수가 `us-east-2`에서 `us-east-1`로 스냅샷을 복사합니다.

```
// Sample Lambda function to copy an EBS snapshot to a different region  
  
var AWS = require('aws-sdk');  
var ec2 = new AWS.EC2();  
  
// define variables  
var destinationRegion = 'us-east-1';  
var sourceRegion = 'us-east-2';  
console.log ('Loading function')  
  
//main function  
exports.handler = (event, context, callback) => {  
  
    // Get the EBS snapshot ID from the CloudWatch event details  
    var snapshotArn = event.detail.snapshot_id.split('/');  
    const snapshotId = snapshotArn[1];  
    const description = `Snapshot copy from ${snapshotId} in ${sourceRegion}.`;  
    console.log ("snapshotId:", snapshotId);
```

```
// Load EC2 class and update the configuration to use destination Region to
// initiate the snapshot.
AWS.config.update({region: destinationRegion});
var ec2 = new AWS.EC2();

// Prepare variables for ec2.modifySnapshotAttribute call
const copySnapshotParams = {
    Description: description,
    DestinationRegion: destinationRegion,
    SourceRegion: sourceRegion,
    SourceSnapshotId: snapshotId
};

// Execute the copy snapshot and log any errors
ec2.copySnapshot(copySnapshotParams, (err, data) => {
    if (err) {
        const errorMessage = `Error copying snapshot ${snapshotId} to Region
${destinationRegion}.`;
        console.log(errorMessage);
        console.log(err);
        callback(errorMessage);
    } else {
        const successMessage = `Successfully started copy of snapshot ${snapshotId}
to Region ${destinationRegion}.`;
        console.log(successMessage);
        console.log(data);
        callback(null, successMessage);
    }
});
```

Lambda 함수를 CloudWatch 콘솔에서 사용하도록 보장하려면 CloudWatch 이벤트가 발생하는 리전에서 생성합니다. 자세한 내용은 [AWS Lambda 개발자 안내서](#)를 참조하십시오.

3. <https://console.aws.amazon.com/cloudwatch>에서 CloudWatch 콘솔을 엽니다.
4. 이벤트, 규칙 생성, Select event source(이벤트 소스 선택), Amazon EBS 스냅샷을 선택합니다.
5. Specific Event(s)(특정 이벤트)에서 createSnapshot(스냅샷 생성)을, Specific Result(s)(특정 결과)에서 성공을 선택합니다.
6. Rule target(대상 실행)에서 이전에 생성한 샘플 함수를 찾아 선택합니다.
7. 대상, 대상 추가를 선택합니다.
8. Lambda 함수에서 이전에 생성한 Lambda 함수를 선택하고 세부 정보 구성을 선택합니다.
9. 규칙 세부 정보 구성 페이지에서 이름 및 설명에 값을 입력합니다. 상태 확인란을 선택하여 함수를 활성화합니다(활성으로 설정).
10. 규칙 생성을 선택합니다.

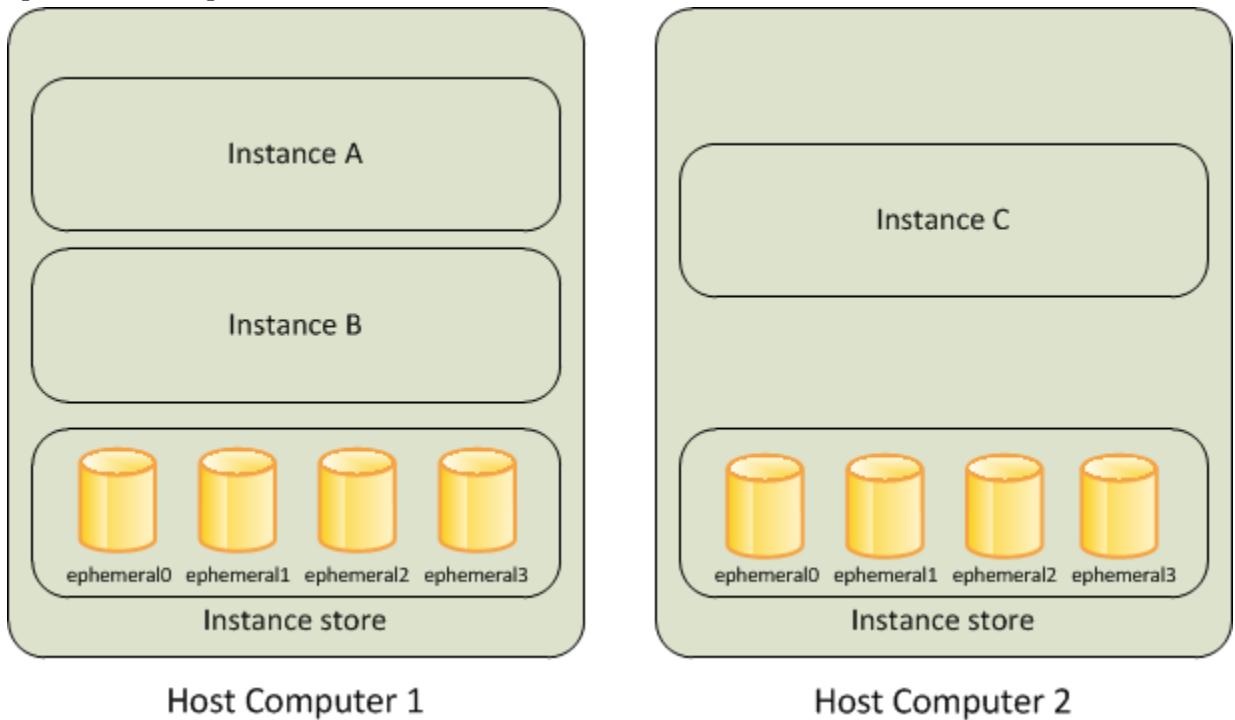
이제 규칙 탭에 규칙이 표시됩니다. 표시된 예에서 구성된 이벤트는 다음에 스냅샷을 복사할 때 EBS가 발생시킵니다.

Amazon EC2 인스턴스 스토어

인스턴스 스토어는 인스턴스에 블록 수준의 임시 스토리지를 제공합니다. 스토리지는 호스트 컴퓨터에 물리적으로 연결된 디스크에 위치합니다. 인스턴스 스토어는 버퍼, 캐시, 스크래치 데이터 및 기타 임시 콘텐츠와 같이 자주 변경되는 정보의 임시 스토리지나 로드가 분산된 웹 서버 풀과 같은 여러 인스턴스상에서 복제되는 데이터에 가장 적합합니다.

하나 이상의 인스턴스 스토어 볼륨으로 구성된 인스턴스 스토어는 블록 디바이스로 표시됩니다. 인스턴스 스토어의 크기는 물론 사용 가능한 디바이스의 수는 인스턴스 유형에 따라 다릅니다.

인스턴스 스토어 볼륨의 가상 디바이스는 `ephemeral[0-23]`입니다. 인스턴스 스토어 볼륨 1개를 지원하는 인스턴스 유형은 `ephemeral0`를 갖습니다. 인스턴스 스토어 볼륨 2개를 지원하는 인스턴스 유형은 `ephemeral0` 및 `ephemeral1` 등을 갖습니다.



목차

- [인스턴스 스토어 수명 \(p. 1092\)](#)
- [인스턴스 스토어 볼륨 \(p. 1093\)](#)
- [EC2 인스턴스에 인스턴스 스토어 볼륨 추가 \(p. 1098\)](#)
- [SSD 인스턴스 스토어 볼륨 \(p. 1101\)](#)

인스턴스 스토어 수명

실행 시에만 인스턴스에 대한 인스턴스 스토어 볼륨을 지정할 수 있습니다. 하나의 인스턴스에서 인스턴스 스토어 볼륨을 분리하고 다른 인스턴스에 연결할 수 없습니다.

인스턴스 스토리지의 데이터는 관련 인스턴스의 수명 기간 동안만 지속됩니다. 인스턴스가 재부팅(의도적 또는 의도적이지 않게)되면 인스턴스 스토어의 데이터는 유지됩니다. 그러나 다음 상황에서는 인스턴스 스토어의 데이터가 손실됩니다.

- 기본 디스크 드라이브 오류
- 인스턴스가 중지됨
- 인스턴스가 종료됨

그러므로 중요한 장기 데이터의 경우 인스턴스 스토어에 의존하지 마십시오. 오히려 Amazon S3, Amazon EBS 또는 Amazon EFS 등 내구성이 뛰어난 데이터 스토리지를 사용하는 것이 좋습니다.

인스턴스를 중지하거나 종료하면 인스턴스 스토어의 모든 스토리지 블록이 리셋됩니다. 따라서 다른 인스턴스의 인스턴스 스토어를 통해 데이터를 액세스할 수 없습니다.

인스턴스에서 AMI를 생성한 경우 해당 인스턴스 스토어 볼륨의 데이터는 보존되지 않고 이 AMI를 실행한 인스턴스용 인스턴스 스토어 볼륨에 존재하지 않습니다.

인스턴스 유형을 변경하면 인스턴스 스토어가 새 인스턴스 유형에 연결되지 않습니다. 자세한 내용은 [인스턴스 유형 변경 \(p. 194\)](#) 단원을 참조하십시오.

인스턴스 스토어 볼륨

인스턴스 유형은 사용 가능한 인스턴스 스토어의 크기와 인스턴스 스토어 볼륨에서 사용되는 하드웨어 유형을 결정합니다. 인스턴스 스토어 볼륨은 인스턴스의 사용 요금의 일부로 포함됩니다. 인스턴스를 시작할 때 사용할 인스턴스 스토어 볼륨을 지정해야 합니다(기본적으로 사용할 수 있는 NVMe 인스턴스 스토어 볼륨은 제외). 그런 다음 사용하기에 앞서 인스턴스 스토어 볼륨을 포맷하고 마운트해야 합니다. 인스턴스를 시작한 이후에는 사용 가능한 인스턴스 스토어 볼륨을 연결할 수 없습니다. 자세한 내용은 [EC2 인스턴스에 인스턴스 스토어 볼륨 추가 \(p. 1098\)](#) 단원을 참조하십시오.

일부 인스턴스 유형은 NVMe 또는 SATA 기반 SSD(Solid State Drive)를 사용하여 높은 랜덤 I/O 성능을 제공합니다. 이것은 지연시간이 매우 짧은 스토리지가 필요하지만 인스턴스가 종료될 경우에는 지속할 데이터가 필요가 없는 경우, 또는 내결함성 아키텍처를 활용할 수 있는 경우에 적합한 옵션입니다. 자세한 내용은 [SSD 인스턴스 스토어 볼륨 \(p. 1101\)](#) 단원을 참조하십시오.

다음 표에는 지원되는 각 인스턴스 유형에서 사용 가능한 인스턴스 스토어 볼륨의 수량, 크기, 유형 및 성능 최적화가 나와 있습니다. EBS 전용 유형을 포함한 전체 인스턴스 유형 목록을 보려면 [Amazon EC2 인스턴스 유형](#)을 참조하십시오.

인스턴스 유형	인스턴스 스토어 볼륨	Type	초기화 필요*	TRIM 지원**
c1.medium	1 x 350GB	HDD	✓	
c1.xlarge	4 x 420GB(1.6TB)	HDD	✓	
c3.large	2 x 16GB(32GB)	SSD	✓	
c3.xlarge	2 x 40GB(80GB)	SSD	✓	
c3.2xlarge	2 x 80GB(160GB)	SSD	✓	
c3.4xlarge	2 x 160GB(320GB)	SSD	✓	
c3.8xlarge	2 x 320GB(640GB)	SSD	✓	
c5ad.large	1 x 75GB	NVMe SSD		✓
c5ad.xlarge	1 x 150GB	NVMe SSD		✓
c5ad.2xlarge	1 x 300GB	NVMe SSD		✓
c5ad.4xlarge	2 x 300GB(600GB)	NVMe SSD		✓
c5ad.8xlarge	2 x 600GB(1.2TB)	NVMe SSD		✓
c5ad.12xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
c5ad.16xlarge	2 x 1.2TB(2.4TB)	NVMe SSD		✓
c5ad.24xlarge	2 x 1.9TB(3.8TB)	NVMe SSD		✓
c5d.large	1 x 50GB	NVMe SSD		✓
c5d.xlarge	1 x 100GB	NVMe SSD		✓
c5d.2xlarge	1 x 200GB	NVMe SSD		✓

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
인스턴스 스토어 볼륨

인스턴스 유형	인스턴스 스토어 볼륨	Type	초기화 필요*	TRIM 지원**
c5d.4xlarge	1 x 400GB	NVMe SSD		✓
c5d.9xlarge	1 x 900GB	NVMe SSD		✓
c5d.12xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
c5d.18xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
c5d.24xlarge	4 x 900GB(3.6TB)	NVMe SSD		✓
c5d.metal	4 x 900GB(3.6TB)	NVMe SSD		✓
cc2.8xlarge	4 x 840GB(3.36TB)	HDD	✓	
cr1.8xlarge	2 x 120GB(240GB)	SSD	✓	
d2.xlarge	3 x 2,000GB(6TB)	HDD		
d2.2xlarge	6 x 2,000GB(12TB)	HDD		
d2.4xlarge	12 x 2,000GB(24TB)	HDD		
d2.8xlarge	24 x 2,000GB(48TB)	HDD		
f1.2xlarge	1 x 470GB	NVMe SSD		✓
f1.4xlarge	1 x 940GB	NVMe SSD		✓
f1.16xlarge	4 x 940GB(3.76TB)	NVMe SSD		✓
g2.2xlarge	1 x 60GB	SSD	✓	
g2.8xlarge	2 x 120GB(240GB)	SSD	✓	
g4dn.xlarge	1 x 125GB	NVMe SSD		✓
g4dn.2xlarge	1 x 225GB	NVMe SSD		✓
g4dn.4xlarge	1 x 225GB	NVMe SSD		✓
g4dn.8xlarge	1 x 900GB	NVMe SSD		✓
g4dn.12xlarge	1 x 900GB	NVMe SSD		✓
g4dn.16xlarge	1 x 900GB	NVMe SSD		✓
g4dn.metal	2 x 900GB(1.8TB)	NVMe SSD		✓
h1.2xlarge	1 x 2000GB(2TB)	HDD		
h1.4xlarge	2 x 2000GB(4TB)	HDD		
h1.8xlarge	4 x 2000GB(8TB)	HDD		
h1.16xlarge	8 x 2000GB(16TB)	HDD		
hs1.8xlarge	24 x 2,000GB(48TB)	HDD	✓	
i2.xlarge	1 x 800GB	SSD		✓
i2.2xlarge	2 x 800GB(1.6TB)	SSD		✓

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
인스턴스 스토어 볼륨

인스턴스 유형	인스턴스 스토어 볼륨	Type	초기화 필요*	TRIM 지원**
i2.4xlarge	4 x 800GB(3.2TB)	SSD		✓
i2.8xlarge	8 x 800GB(6.4TB)	SSD		✓
i3.large	1 x 475GB	NVMe SSD		✓
i3.xlarge	1 x 950GB	NVMe SSD		✓
i3.2xlarge	1 x 1,900GB	NVMe SSD		✓
i3.4xlarge	2 x 1,900GB(3.8TB)	NVMe SSD		✓
i3.8xlarge	4 x 1,900GB(7.6TB)	NVMe SSD		✓
i3.16xlarge	8 x 1,900GB(15.2TB)	NVMe SSD		✓
i3.metal	8 x 1,900GB(15.2TB)	NVMe SSD		✓
i3en.large	1 x 1,250GB	NVMe SSD		✓
i3en.xlarge	1 x 2,500GB	NVMe SSD		✓
i3en.2xlarge	2 x 2,500GB(5TB)	NVMe SSD		✓
i3en.3xlarge	1 x 7,500GB	NVMe SSD		✓
i3en.6xlarge	2 x 7,500GB(15TB)	NVMe SSD		✓
i3en.12xlarge	4 x 7,500GB(30TB)	NVMe SSD		✓
i3en.24xlarge	8 x 7,500GB(60TB)	NVMe SSD		✓
i3en.metal	8 x 7,500GB(60TB)	NVMe SSD		✓
m1.small	1 x 160GB	HDD	✓	
m1.medium	1 x 410GB	HDD	✓	
m1.large	2 x 420GB(840GB)	HDD	✓	
m1.xlarge	4 x 420GB(1.6TB)	HDD	✓	
m2.xlarge	1 x 420GB	HDD	✓	
m2.2xlarge	1 x 850GB	HDD	✓	
m2.4xlarge	2 x 840GB(1.68TB)	HDD	✓	
m3.medium	1 x 4GB	SSD	✓	
m3.large	1 x 32GB	SSD	✓	
m3.xlarge	2 x 40GB(80GB)	SSD	✓	
m3.2xlarge	2 x 80GB(160GB)	SSD	✓	
m5ad.large	1 x 75GB	NVMe SSD		✓
m5ad.xlarge	1 x 150GB	NVMe SSD		✓
m5ad.2xlarge	1 x 300GB	NVMe SSD		✓

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
인스턴스 스토어 볼륨

인스턴스 유형	인스턴스 스토어 볼륨	Type	초기화 필요*	TRIM 지원**
m5ad.4xlarge	2 x 300GB(600GB)	NVMe SSD		✓
m5ad.8xlarge	2 x 600GB(1.2TB)	NVMe SSD		✓
m5ad.12xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
m5ad.16xlarge	4 x 600GB(2.4TB)	NVMe SSD		✓
m5ad.24xlarge	4 x 900GB(3.6TB)	NVMe SSD		✓
m5d.large	1 x 75GB	NVMe SSD		✓
m5d.xlarge	1 x 150GB	NVMe SSD		✓
m5d.2xlarge	1 x 300GB	NVMe SSD		✓
m5d.4xlarge	2 x 300GB(600GB)	NVMe SSD		✓
m5d.8xlarge	2 x 600GB(1.2TB)	NVMe SSD		✓
m5d.12xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
m5d.16xlarge	4 x 600GB(2.4TB)	NVMe SSD		✓
m5d.24xlarge	4 x 900GB(3.6TB)	NVMe SSD		✓
m5d.metal	4 x 900GB(3.6TB)	NVMe SSD		✓
m5dn.large	1 x 75GB	NVMe SSD		✓
m5dn.xlarge	1 x 150GB	NVMe SSD		✓
m5dn.2xlarge	1 x 300GB	NVMe SSD		✓
m5dn.4xlarge	2 x 300GB(600GB)	NVMe SSD		✓
m5dn.8xlarge	2 x 600GB(1.2TB)	NVMe SSD		✓
m5dn.12xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
m5dn.16xlarge	4 x 600GB(2.4TB)	NVMe SSD		✓
m5dn.24xlarge	4 x 900GB(3.6TB)	NVMe SSD		✓
p3dn.24xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
r3.large	1 x 32GB	SSD		✓
r3.xlarge	1 x 80GB	SSD		✓
r3.2xlarge	1 x 160GB	SSD		✓
r3.4xlarge	1 x 320GB	SSD		✓
r3.8xlarge	2 x 320GB(640GB)	SSD		✓
r5ad.large	1 x 75GB	NVMe SSD		✓
r5ad.xlarge	1 x 150GB	NVMe SSD		✓
r5ad.2xlarge	1 x 300GB	NVMe SSD		✓

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
인스턴스 스토어 볼륨

인스턴스 유형	인스턴스 스토어 볼륨	Type	초기화 필요*	TRIM 지원**
r5ad.4xlarge	2 x 300GB(600GB)	NVMe SSD		✓
r5ad.8xlarge	2 x 600GB(1.2TB)	NVMe SSD		✓
r5ad.12xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
r5ad.16xlarge	4 x 600GB(2.4TB)	NVMe SSD		✓
r5ad.24xlarge	4 x 900GB(3.6TB)	NVMe SSD		✓
r5d.large	1 x 75GB	NVMe SSD		✓
r5d.xlarge	1 x 150GB	NVMe SSD		✓
r5d.2xlarge	1 x 300GB	NVMe SSD		✓
r5d.4xlarge	2 x 300GB(600GB)	NVMe SSD		✓
r5d.8xlarge	2 x 600GB(1.2TB)	NVMe SSD		✓
r5d.12xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
r5d.16xlarge	4 x 600GB(2.4TB)	NVMe SSD		✓
r5d.24xlarge	4 x 900GB(3.6TB)	NVMe SSD		✓
r5d.metal	4 x 900GB(3.6TB)	NVMe SSD		✓
r5dn.large	1 x 75GB	NVMe SSD		✓
r5dn.xlarge	1 x 150GB	NVMe SSD		✓
r5dn.2xlarge	1 x 300GB	NVMe SSD		✓
r5dn.4xlarge	2 x 300GB(600GB)	NVMe SSD		✓
r5dn.8xlarge	2 x 600GB(1.2TB)	NVMe SSD		✓
r5dn.12xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
r5dn.16xlarge	4 x 600GB(2.4TB)	NVMe SSD		✓
r5dn.24xlarge	4 x 900GB(3.6TB)	NVMe SSD		✓
x1.16xlarge	1 x 1,920GB	SSD		
x1.32xlarge	2 x 1,920GB(3.84TB)	SSD		
x1e.xlarge	1 x 120GB	SSD		
x1e.2xlarge	1 x 240GB	SSD		
x1e.4xlarge	1 x 480GB	SSD		
x1e.8xlarge	1 x 960GB	SSD		
x1e.16xlarge	1 x 1,920GB	SSD		
x1e.32xlarge	2 x 1,920GB(3.84TB)	SSD		
z1d.large	1 x 75GB	NVMe SSD		✓

인스턴스 유형	인스턴스 스토어 볼륨	Type	초기화 필요*	TRIM 지원**
z1d.xlarge	1 x 150GB	NVMe SSD		✓
z1d.2xlarge	1 x 300GB	NVMe SSD		✓
z1d.3xlarge	1 x 450GB	NVMe SSD		✓
z1d.6xlarge	1 x 900GB	NVMe SSD		✓
z1d.12xlarge	2 x 900GB(1.8TB)	NVMe SSD		✓
z1d.metal	2 x 900GB(1.8TB)	NVMe SSD		✓

* 특정 인스턴스에 연결된 볼륨은 초기화되지 않을 경우 최초 쓰기 페널티를 받게 됩니다.

** 자세한 내용은 [인스턴스 스토어 볼륨 TRIM 지원 \(p. 1102\)](#) 단원을 참조하십시오.

EC2 인스턴스에 인스턴스 스토어 볼륨 추가

블록 디바이스 매핑을 사용하여 인스턴스에 대한 EBS 볼륨 및 인스턴스 스토어 볼륨을 지정합니다. 블록 디바이스 매핑의 각 항목은 디바이스 이름 및 매핑된 볼륨을 포함합니다. 기본 블록 디바이스 매핑은 사용하는 AMI에 의해 지정됩니다. 또는 시작 시 인스턴스에 대한 블록 디바이스 매핑을 지정할 수 있습니다.

인스턴스 유형에서 지원되는 모든 NVMe 인스턴스 스토어 볼륨이 인스턴스 시작 시 자동으로 열거되고 디바이스 이름이 할당됩니다. 따라서 AMI 또는 인스턴스에 대한 블록 디바이스 매핑에 이를 포함하는 것은 효과가 없습니다. 자세한 내용은 [블록 디바이스 매핑 \(p. 1107\)](#) 단원을 참조하십시오.

블록 디바이스 매핑은 항상 인스턴스에 대한 루트 볼륨을 지정합니다. 루트 볼륨은 자동으로 마운트됩니다. Windows 인스턴스의 경우 루트 볼륨은 Amazon EBS 볼륨이어야 합니다. 루트 볼륨에는 인스턴스 스토어가 지원되지 않습니다.

블록 디바이스 매핑을 사용하면 인스턴스를 실행할 때 인스턴스에 연결할 추가 EBS 볼륨을 지정하거나 인스턴스가 실행된 후에 추가 EBS 볼륨을 연결할 수 있습니다. 자세한 내용은 [Amazon EBS 볼륨 \(p. 929\)](#) 단원을 참조하십시오.

인스턴스 실행 시에만 인스턴스에 대한 인스턴스 스토어 볼륨을 지정할 수 있습니다. 인스턴스를 실행한 이후에는 인스턴스 스토어 볼륨을 연결할 수 없습니다.

인스턴스 유형을 변경하면 인스턴스 스토어가 새 인스턴스 유형에 연결되지 않습니다. 자세한 내용은 [인스턴스 유형 변경 \(p. 194\)](#) 단원을 참조하십시오.

이러한 볼륨의 개수 및 크기는 인스턴스 유형에 따라 다른 인스턴스에서 사용 가능한 인스턴스 스토어 볼륨을 초과하지 않아야 합니다. 일부 인스턴스 유형은 인스턴스 스토어 볼륨을 지원하지 않습니다. 블록 디바이스 매핑에 있는 인스턴스 스토어 볼륨 수가 인스턴스에 사용 가능한 인스턴스 스토어 볼륨 수를 초과한 경우 추가 볼륨이 무시됩니다. 인스턴스 유형별 인스턴스 스토어 볼륨 지원에 대한 자세한 내용은 [인스턴스 스토어 볼륨 \(p. 1093\)](#) 단원을 참조하십시오.

인스턴스에 대해 선택한 인스턴스 유형이 비 NVMe 인스턴스 스토어 볼륨을 지원하는 경우 인스턴스를 실행할 때 인스턴스에 대한 블록 디바이스 매핑에 추가해야 합니다. NVMe 인스턴스 스토어 볼륨은 기본적으로 사용 가능합니다. 인스턴스를 실행한 후에는 인스턴스에 대한 인스턴스 스토어 볼륨이 사용하기에 앞서 포맷되고 마운트되었는지 확인해야 합니다. 인스턴스 스토어 지원 인스턴스의 루트 볼륨은 기본적으로 마운트됩니다.

목차

- [AMI에 인스턴스 스토어 볼륨 추가 \(p. 1099\)](#)
- [인스턴스에 인스턴스 스토어 볼륨 추가 \(p. 1099\)](#)

- [인스턴스 스토어 볼륨을 인스턴스에서 사용 가능하도록 만들기 \(p. 1100\)](#)

AMI에 인스턴스 스토어 볼륨 추가

인스턴스 스토어 볼륨을 포함하는 블록 디바이스 매핑으로 AMI를 생성할 수 있습니다. 인스턴스 스토어 볼륨을 지원하는 인스턴스 유형과 해당 블록 디바이스 매핑에 인스턴스 스토어 볼륨을 지정하는 AMI로 인스턴스를 시작하는 경우, 인스턴스에는 이러한 인스턴스 스토어 볼륨이 포함됩니다. 블록 디바이스 매핑에 있는 인스턴스 스토어 볼륨 수가 인스턴스에 사용 가능한 인스턴스 스토어 볼륨 수를 초과한 경우 추가 인스턴스 스토어 볼륨이 무시됩니다.

고려 사항

- M3 인스턴스의 경우, AMI가 아니라 인스턴스의 블록 디바이스 매핑을 사용하여 인스턴스 스토어 볼륨을 지정합니다. Amazon EC2가 AMI의 블록 디바이스 매핑에서만 지정된 인스턴스 스토어 볼륨을 무시할 수 있습니다.
- 인스턴스를 실행할 때 AMI 블록 디바이스 매핑에서 지정된 비 NVMe 인스턴스 스토어 볼륨을 생략하거나 인스턴스 스토어 볼륨을 추가할 수 있습니다.

콘솔을 사용하여 Amazon EBS 지원 AMI에 인스턴스 스토어 볼륨을 추가하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
3. 작업, 이미지, 이미지 생성을 차례로 선택합니다.
4. 이미지 생성 대화 상자에 이미지의 이름 및 설명을 입력합니다.
5. 추가할 각 인스턴스 스토어 볼륨에서 새 볼륨 추가를 선택한 다음 볼륨 유형에서 인스턴스 스토어 볼륨을 선택하고 디바이스에서 디바이스 이름을 선택합니다. (자세한 내용은 [Windows 인스턴스의 디바이스 명명 \(p. 1106\)](#) 단원을 참조하십시오.) 사용할 수 있는 인스턴스 스토어 볼륨의 개수는 인스턴스 유형에 따라 다릅니다. NVMe 인스턴스 스토어 볼륨이 있는 인스턴스의 경우, 이러한 볼륨의 디바이스 매핑은 운영 체제가 볼륨을 열거하는 순서에 따라 다릅니다.
6. 이미지 생성을 선택합니다.

명령줄을 사용하여 AMI에 인스턴스 스토어 볼륨을 추가하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [create-image](#) 또는 [register-image](#)(AWS CLI)
- [New-EC2Image](#) 및 [Register-EC2Image](#)(Windows PowerShell용 AWS 도구)

인스턴스에 인스턴스 스토어 볼륨 추가

인스턴스를 실행할 때 기본 블록 디바이스 매핑은 지정된 AMI에 의해 제공됩니다. 추가 인스턴스 스토어 볼륨이 필요할 경우 실행할 때 인스턴스에 추가해야 합니다. AMI 블록 디바이스 매핑에서 지정된 디바이스는 생략할 수도 있습니다.

고려 사항

- M3 인스턴스의 경우, 인스턴스의 블록 디바이스 매핑에서 지정하지 않더라도 인스턴스 스토어 볼륨을 받을 수 있습니다.
- HS1 인스턴스의 경우, 사용자가 AMI의 블록 디바이스 매핑에서 몇 개의 인스턴스 스토어 볼륨을 지정하더라도 AMI으로부터 시작된 인스턴스의 블록 디바이스 매핑이 지원되는 최대 개수의 인스턴스 스토어 볼륨을 자동으로 포함합니다. 인스턴스를 시작하기 전에 해당 인스턴스의 블록 디바이스 매핑에서 원치 않는 인스턴스 스토어 볼륨을 명시적으로 제거해야 합니다.

콘솔을 사용하여 인스턴스에 대한 블록 디바이스 매핑을 업데이트하려면

1. Amazon EC2 콘솔을 엽니다.
2. 대시보드에서 인스턴스 시작을 선택합니다.
3. 1단계: Amazon 머신 이미지(AMI) 선택(Step 1: Choose an Amazon Machine Image (AMI))에서 사용할 AMI를 선택하고 선택을 선택합니다.
4. 마법사를 따라 1단계: Amazon 머신 이미지(AMI) 선택(Step 1: Choose an Amazon Machine Image (AMI)), 2단계: 인스턴스 유형 선택(Step 2: Choose an Instance Type) 및 3단계: 인스턴스 세부 정보 구성(Step 3: Configure Instance Details)을 완료합니다.
5. 4단계: 스토리지 추가(Step 4: Add Storage)에서 필요에 따라 기존 항목을 수정합니다. 추가할 각 인스턴스 스토어 볼륨에서 새 볼륨 추가를 선택한 다음 볼륨 유형에서 인스턴스 스토어 볼륨을 선택하고 디바이스에서 디바이스 이름을 선택합니다. 사용할 수 있는 인스턴스 스토어 볼륨의 개수는 인스턴스 유형에 따라 다릅니다.
6. 마법사를 완료하고 인스턴스를 시작합니다.
7. (선택 사항) 인스턴스에 사용 가능한 인스턴스 스토어 볼륨을 확인하려면 Windows 디스크 관리를 엽니다.

명령줄을 사용하여 인스턴스에 대한 블록 디바이스 매핑을 업데이트하려면

해당 명령과 함께 다음 옵션 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- `run-instances`(AWS CLI)를 사용한 `--block-device-mappings`
- `New-EC2Instance`(Windows PowerShell용 AWS 도구)를 사용한 `-BlockDeviceMapping`

인스턴스 스토어 볼륨을 인스턴스에서 사용 가능하도록 만들기

인스턴스를 시작한 후에 인스턴스 스토어 볼륨은 인스턴스에서 사용 가능하지만 마운트 이후에 액세스할 수 있습니다. Linux 인스턴스의 경우 인스턴스 유형에 따라 어느 인스턴스 스토어 볼륨이 마운트되고 어느 것을 마운트할 수 있는지가 결정됩니다. Windows 인스턴스의 경우 EC2Config 서비스가 인스턴스에서 대한 인스턴스 스토리지 볼륨을 마운트합니다. 인스턴스의 블록 디바이스 드라이버는 볼륨이 마운트될 때 실제 볼륨 이름을 할당하고 할당된 이름은 Amazon EC2 권장 이름과 다를 수 있습니다.

여러 인스턴스 스토리지 볼륨은 ext3 파일 시스템으로 사전 포맷됩니다. SSD 기반 인스턴스 스토리지 볼륨 (TRIM 명령 지원)은 어떤 파일 시스템으로도 사전 포맷되지 않습니다. 그러나 인스턴스를 시작한 후 볼륨을 원하는 파일 시스템으로 포맷할 수 있습니다. 자세한 내용은 [인스턴스 스토어 볼륨 TRIM 지원 \(p. 1102\)](#) 단원을 참조하십시오. Windows 인스턴스의 경우 EC2Config 서비스가 NTFS 파일 시스템으로 인스턴스 스토리지 볼륨을 다시 포맷합니다.

인스턴스에서 인스턴스 스토어 디바이스를 사용할 수 있는지의 여부는 인스턴스 메타데이터를 사용하여 확인할 수 있습니다. 자세한 내용은 [인스턴스 스토어 볼륨용 인스턴스 블록 디바이스 매핑 보기 \(p. 1115\)](#) 단원을 참조하십시오.

Windows 인스턴스의 경우 Windows 디스크 관리를 사용하여 인스턴스 스토어 볼륨을 볼 수도 있습니다. 자세한 내용은 [Windows 디스크 관리를 이용하여 디스크 나열 \(p. 1116\)](#)을 참조하십시오.

인스턴스 스토어 볼륨을 수동으로 탑재하려면

1. 시작을 선택하고 컴퓨터 관리를 입력한 다음 Enter 키를 누릅니다.
2. 왼쪽 패널에서 디스크 관리를 선택합니다.
3. 볼륨을 초기화하라는 메시지가 나타나면 초기화할 볼륨을 선택하고, 사용 사례에 따라 필요한 파티션 유형을 선택한 다음 확인을 선택합니다.
4. 볼륨 목록에서 탑재할 볼륨을 마우스 오른쪽 버튼으로 클릭한 다음 새 단순 볼륨을 선택합니다.

5. 마법사에서 다음을 선택합니다.
6. 볼륨 크기 지정 화면에서 다음을 선택하여 최대 볼륨 크기를 사용합니다. 또는 최소 및 최대 디스크 공간 사이의 볼륨 크기를 선택합니다.
7. 드라이브 문자 또는 경로 할당 화면에서 다음 중 하나를 수행하고 다음을 선택합니다.
 - 드라이브 문자를 사용하여 볼륨을 탑재하려면 다음 드라이브 문자 할당을 선택한 다음, 사용할 드라이브 문자를 선택합니다.
 - 볼륨을 폴더로 탑재하려면 다음 빈 NTFS 폴더에서 탑재를 선택한 다음 찾아보기를 선택하여 사용할 폴더를 생성하거나 선택합니다.
 - 드라이브 문자나 경로 없이 볼륨을 탑재하려면 드라이브 문자 또는 드라이브 경로 할당 안 함을 선택합니다.
8. 파티션 포맷 화면에서 볼륨을 포맷할지 여부를 지정합니다. 볼륨을 포맷하도록 선택한 경우 필요한 파일 시스템 및 단위 크기를 선택하고 볼륨 레이블을 지정합니다.
9. 다음, 완료를 선택합니다.

SSD 인스턴스 스토어 볼륨

다른 인스턴스 스토어 볼륨과 마찬가지로 인스턴스 시작 시 인스턴스에 대한 SSD 인스턴스 스토어 볼륨을 매핑해야 합니다. SSD 인스턴스 볼륨의 데이터는 연결된 인스턴스의 수명 기간 동안만 지속됩니다. 자세한 내용은 [EC2 인스턴스에 인스턴스 스토어 볼륨 추가 \(p. 1098\)](#) 단원을 참조하십시오.

NVMe SSD 볼륨

C5d, G4, I3, I3en, F1, M5ad, M5d, p3dn.24xlarge, R5ad, R5d 및 z1d 인스턴스는 NVMe(Non-Volatile Memory Express) SSD(Solid State Drive) 인스턴스 스토어 볼륨을 제공합니다. 다음 운영 체제용 최신 AWS Windows AMI에는 성능 향상을 위해 NVMe 블록 디바이스로 표시되는 EBS 및 SSD 인스턴스 스토어 볼륨과 상호 작용하는 데 사용되는 AWS NVMe 드라이버가 포함되어 있습니다.

- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2
- Windows Server 2012
- Windows Server 2008 R2

인스턴스에 연결한 후 디스크 관리자에서 NVMe 볼륨이 보이는지 확인할 수 있습니다. 작업 표시줄에서 마우스 오른쪽 버튼을 클릭하여 Windows 로고에 대한 컨텍스트 메뉴를 열고 디스크 관리(Disk Management)를 선택합니다. Windows Server 2008 R2에서는 시작, 관리 도구, 컴퓨터 관리, 디스크 관리를 선택합니다.

Amazon에서 제공하는 AWS Windows AMI에는 AWS NVMe 드라이버가 포함되어 있습니다. 최신 AWS Windows AMI를 사용하고 있지 않은 경우 [현재 AWS NVMe 드라이버를 설치 \(p. 539\)](#)할 수 있습니다.

인스턴스 하드웨어 모듈에 구현된 XTS-AES-256 블록 암호를 사용하여 NVMe 인스턴스 스토리지의 데이터를 암호화합니다. 하드웨어 모듈을 사용하여 암호화 키를 생성하며, 암호화 키는 각 NVMe 인스턴스 스토리지 디바이스에 고유합니다. 인스턴스가 종지되거나 종료되면 모든 암호화 키가 손상되어 복구가 불가능해집니다. 이 암호화를 비활성화할 수 없으며, 사용자 자신의 암호화 키를 제공할 수 없습니다.

비 NVMe SSD 볼륨

C3, G2, I2, M3, R3, X1 인스턴스는 비 NVMe SSD를 사용하여 높은 랜덤 I/O 성능을 제공하는 인스턴스 스토어 볼륨을 지원합니다. 인스턴스 유형별 인스턴스 스토어 볼륨 지원에 대한 자세한 내용은 [인스턴스 스토어 볼륨 \(p. 1093\)](#) 단원을 참조하십시오.

인스턴스 스토어 볼륨 TRIM 지원

다음 인스턴스는 TRIM을 통해 SSD 볼륨을 지원합니다. C5d, F1, G4, I2, I3, I3en, M5ad, M5d, p3dn.24xlarge, R3, R5ad, R5d 및 z1d.

Windows Server 2012 R2를 실행하는 인스턴스는 AWS PV 드라이버 버전 7.3.0부터 TRIM을 지원합니다. 이전 버전의 Windows Server를 실행하는 인스턴스는 TRIM을 지원하지 않습니다.

TM을 지원하는 인스턴스 스토어 볼륨은 인스턴스에 할당되기 전 완전히 트리밍(trimming)됩니다. 이러한 볼륨은 인스턴스가 실행될 때 파일 시스템으로 포맷되지 않으므로, 마운트 후 사용하기 전 사용자가 해당 볼륨을 포맷해야 합니다. 이러한 볼륨에 액세스하는 속도를 높이려면 볼륨을 포맷할 때 TRIM 작업을 건너뛰어야 합니다. Windows에서 초기 포맷 중에 TRIM 지원을 일시적으로 비활성화하려면 `fsutil behavior set DisableDeleteNotify 1` 명령을 사용합니다. 포맷이 완료되면 `fsutil behavior set DisableDeleteNotify 0`을 사용하여 TRIM 지원을 다시 활성화합니다.

TRIM을 지원하는 인스턴스 스토어 볼륨을 사용할 경우 TRIM 명령을 사용하여 작성한 데이터가 더 이상 필요하지 않음을 SSD 컨트롤러에 통지할 수 있습니다. 이를 통해 컨트롤러에 더 많은 여유 공간이 제공되므로 쓰기 작업 증폭을 줄이고 성능을 향상시킬 수 있습니다. Windows에서는 `fsutil behavior set DisableDeleteNotify 0` 명령을 사용하여 정상적인 작업 중에 TRIM 지원이 활성화되었는지 확인합니다.

파일 스토리지

클라우드 파일 스토리지는 공유 파일 시스템을 통해 서버와 애플리케이션에 데이터에 대한 액세스를 제공하도록 클라우드에 데이터를 저장하는 방법입니다. 이러한 호환성 덕분에 클라우드 파일 스토리지는 공유 파일 시스템을 사용하는 워크로드에 적합하며 코드 변경 없이 간단하게 통합할 수 있습니다.

블록 스토리지를 기본 사양으로 사용하며 확장성이 없거나 데이터 보호를 위한 중복성이 거의 없는 컴퓨팅 인스턴스에 있는 단일 노드 파일 서버에서 직접 클러스터링하는 솔루션 및 완전 관리형 솔루션에 이르는 다양한 파일 스토리지 솔루션이 존재합니다. 다음 내용에서는 Windows와 함께 사용하도록 AWS에서 제공하는 스토리지 서비스 중 일부를 소개합니다.

목차

- [Amazon EC2와 함께 Amazon S3 사용 \(p. 1102\)](#)
- [Amazon EC2와 함께 Amazon EFS 사용 \(p. 1104\)](#)
- [Amazon EC2와 함께 Amazon FSx for Windows File Server 사용 \(p. 1104\)](#)

Amazon EC2와 함께 Amazon S3 사용

Amazon S3는 인터넷 데이터의 리포지토리입니다. Amazon S3는 안정적이고 빠르며 비용이 저렴한 데이터 스토리지 인프라에 대한 액세스를 제공합니다. 언제든지 Amazon EC2 내에서 또는 웹의 어디서나 데이터를 원하는 크기만큼 저장하고 가져올 수 있게 지원함으로써 웹 규모의 컴퓨팅 작업을 쉽게 수행할 수 있도록 설계되었습니다. Amazon S3은 데이터 객체를 여러 시설에 걸쳐 다수의 디바이스에 중복으로 저장하고, 많은 개별 클라이언트 또는 애플리케이션 스레드가 이런 데이터 객체에 대해 연속적 읽기 및 쓰기 액세스를 할 수 있도록 지원합니다. Amazon S3에 저장된 중복 데이터를 사용해서 인스턴스 또는 애플리케이션 장애로부터 빠르게 복구할 수 있습니다.

Amazon EC2는 Amazon 머신 이미지(AMI)를 저장하기 위해 Amazon S3를 사용합니다. AMI를 사용해서 EC2 인스턴스를 시작합니다. 인스턴스 장애의 경우, 저장된 AMI를 사용해서 즉시 다른 인스턴스를 실행할 수 있으며 이를 통해 빠른 복구 및 비즈니스 지속성을 달성할 수 있습니다.

또한 Amazon EC2는 Amazon S3를 사용해서 데이터 볼륨의 스냅샷(백업 사본)을 저장합니다. 애플리케이션 또는 시스템 장애가 발생한 경우 스냅샷을 사용해서 빠르고 안정적으로 데이터를 복원할 수 있습니다. 또한 스냅샷을 하나의 기준으로 사용하여 다수의 새 데이터 볼륨 생성, 기존 데이터 볼륨의 크기 확장, 다수 가용 영역 간 데이터 볼륨 이동 등을 수행할 수 있으며, 이를 통해 사용할 수 있는 데이터를 높은 수준으로 확장할

수 있습니다. 데이터 볼륨 및 스냅샷 사용에 대한 자세한 내용은 [Amazon Elastic Block Store \(p. 928\)](#) 단원을 참조하십시오.

객체는 Amazon S3에 저장되는 기본 개체입니다. Amazon S3에 저장된 모든 객체는 버킷에 저장됩니다. 버킷은 Amazon S3 네임스페이스를 최상위 수준에서 구성하며 해당 스토리지를 담당하는 계정을 식별합니다. Amazon S3 버킷은 인터넷 도메인 이름과 유사합니다. 버킷에 저장된 객체는 고유의 키 값을 가지고 있으며 URL을 사용해서 검색할 수 있습니다. 예를 들어, 키 값이 /photos/mygarden.jpg인 객체는 [DOC-EXAMPLE-BUCKET1](#) 버킷에 저장되며, 다음 URL을 사용하여 주소를 지정할 수 있습니다. <https://DOC-EXAMPLE-BUCKET1.s3.amazonaws.com/photos/mygarden.jpg>

Amazon S3에 대한 자세한 내용은 [Amazon S3제품 페이지](#)를 참조하십시오.

사용 예제:

스토리지에 있어 Amazon S3의 이점을 고려하여 이 서비스를 사용해서 EC2 인스턴스에 사용할 파일 및 데이터 세트를 저장하는 경우가 있을 수 있습니다. Amazon S3 및 인스턴스 간에 데이터를 주고 받는 방법은 여러가지가 있습니다. 아래 설명한 예뿐만 아니라 여러 사람들이 작성한 다양한 도구가 있으며, 이를 사용해서 컴퓨터 또는 인스턴스에서 Amazon S3의 데이터에 액세스할 수 있습니다. 흔하게 사용하는 방법들 중 일부는 AWS 포럼에서 논의되고 있습니다.

권한을 부여받은 경우, 다음 방법 중 하나를 사용해서 Amazon S3 및 인스턴스로 또는 인스턴스로부터 파일을 복사할 수 있습니다.

Windows PowerShell용 AWS 도구

Windows 인스턴스는 Amazon S3 콘솔을 액세스하기 위해 사용할 수 있는 그래픽 브라우저로 인한 이점이 있습니다. 그러나 스크립팅의 경우 Windows 사용자는 [Windows PowerShell용 AWS 도구](#)을 사용해서도 Amazon S3와 객체를 주고받을 수 있습니다.

다음 명령을 사용해서 Amazon S3 객체를 Windows 인스턴스로 복사합니다.

```
PS C:\> Copy-S3Object -BucketName my_bucket -Key path-to-file -LocalFile my_copied_file.ext
```

AWS Command Line Interface

AWS Command Line Interface(AWS CLI)는 AWS 서비스를 관리하는 통합 도구입니다. AWS CLI를 통해 사용자는 인증을 하고 Amazon S3에서 제한되는 항목을 다운로드하고 다른 항목을 업로드할 수도 있게 됩니다. 이 도구의 설치 및 구성 등에 대한 자세한 내용은 [AWS Command Line Interface 세부 정보 페이지](#) 단원을 참조하십시오.

aws s3 cp 명령은 Unix cp 명령과 비슷합니다. Amazon S3에서 인스턴스로 파일을 복사하거나, 인스턴스에서 Amazon S3로 파일을 복사하거나, 하나의 Amazon S3 위치에서 다른 위치로 파일을 복사할 수도 있습니다.

다음 명령을 사용해서 Amazon S3에서 인스턴스로 객체를 복사합니다.

```
aws s3 cp s3://my_bucket/my_folder/my_file.ext my_copied_file.ext
```

다음 명령을 사용해서 인스턴스에서 Amazon S3로 객체를 복사합니다.

```
aws s3 cp my_copied_file.ext s3://my_bucket/my_folder/my_file.ext
```

aws s3 sync 명령은 전체 Amazon S3 버킷을 로컬 디렉터리 위치에 동기화할 수 있습니다. 이는 데이터 세트를 다운로드하고 로컬 사본을 원격 세트에 따라 최신으로 유지하는 데 도움이 될 수 있습니다. Amazon S3 버킷에서 적절한 권한을 보유한 경우, 작업이 완료되면 소스와 대상의 위치를 바꿔 입력해 명령을 실행해서 로컬 디렉터리를 클라우드로 푸시할 수 있습니다.

다음 명령을 사용해서 전체 Amazon S3 버킷을 사용자의 로컬 디렉터리로 다운로드할 수 있습니다.

```
aws s3 sync s3://remote_S3_bucket local_directory
```

Amazon S3 API

개발자라면 API를 사용해서 Amazon S3의 데이터에 액세스할 수 있습니다. 자세한 내용은 [Amazon Simple Storage Service 개발자 가이드](#)를 참조하십시오. 이런 API 및 그 예들을 사용해서 애플리케이션 개발을 지원하고 이를 boto Python 인터페이스 등 다른 API 및 SDK와 통합할 수 있습니다.

Amazon EC2와 함께 Amazon EFS 사용

Amazon EFS는 Amazon EC2에서 사용할 수 있는 확장 가능한 파일 스토리지를 제공합니다. EFS 파일 시스템을 만든 후 파일 시스템을 마운트하도록 인스턴스를 구성할 수 있습니다. 하나의 EFS 파일 시스템을 여러 인스턴스에서 실행하는 워크로드 및 애플리케이션에 대한 공통 데이터 소스로 사용할 수 있습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon Elastic File System\(Amazon EFS\)](#) 및 [Amazon Elastic File System 상품 페이지](#)를 참조하십시오.

Important

Amazon EFS는 Windows 인스턴스에서 지원되지 않습니다.

Amazon EC2와 함께 Amazon FSx for Windows File Server 사용

Amazon FSx for Windows File Server는 다양한 기능과 성능, 호환성을 바탕으로 엔터프라이즈 애플리케이션을 AWS로 쉽게 리프트 앤 시프트 할 수 있는, 기본 Windows 파일 시스템을 바탕으로 하는 완전 관리형 Windows 파일 서버를 제공합니다.

Amazon FSx는 Microsoft Windows Server에 구축된 완전 관리형 파일 스토리지를 제공하는 다양한 엔터프라이즈 Windows 워크로드 모음을 지원합니다. Amazon FSx는 Windows 파일 시스템 기능과 네트워크를 통한 파일 스토리지에 액세스하는 업계 표준 SMB(Server Message Block) 프로토콜을 기본적으로 지원합니다. Amazon FSx는 AWS Cloud의 엔터프라이즈 애플리케이션에 최적화되어 있으며, Windows 호환성과 엔터프라이즈 성능 및 기능, 지속적인 1밀리초 미만 지연 시간을 지원합니다.

Windows 개발자와 관리자가 이용 중인 Amazon FSx 상의 파일 스토리지, 코드, 애플리케이션과 도구를 이용하면 어떤 변경도 없이 작업을 계속 진행할 수 있습니다. Amazon FSx에 이상적인 Windows 애플리케이션과 워크로드에는 비즈니스 애플리케이션, 품 디렉터리, 웹 지원, 콘텐츠 관리, 데이터 분석, 소프트웨어 빌드 설정 및 미디어 처리 워크로드 등이 있습니다.

완전 관리형 서비스인 Amazon FSx for Windows File Server는 파일 서버 및 스토리지 볼륨 설정과 프로비저닝을 위한 관리 부담이 없습니다. 또한 Windows 소프트웨어를 최신 상태로 유지하고, 하드웨어 오류를 감지하고 처리하며, 백업을 수행하기도 합니다. Microsoft Active Directory용 AWS Directory Service, Amazon WorkSpaces, AWS Key Management Service, AWS CloudTrail 같은 다른 AWS 서비스와의 풍부한 통합도 제공합니다.

자세한 내용은 [Amazon FSx for Windows File Server 사용 설명서](#) 단원을 참조하십시오. 요금 정보는 [Amazon FSx for Windows File Server 요금](#)을 참조하십시오.

인스턴스 볼륨 제한

인스턴스에서 보유할 수 있는 최대 볼륨 수는 운영 체제와 인스턴스 유형에 따라 다릅니다. 인스턴스에 추가할 볼륨의 수를 고려할 때 I/O 대역폭 증가 또는 스토리지 용량 증가의 필요성 여부를 고려해야 합니다.

목차

- [Nitro 시스템 볼륨 제한 \(p. 1105\)](#)
- [Windows 볼륨 제한 \(p. 1105\)](#)

- 대역폭 및 용량 비교 (p. 1105)

Nitro 시스템 볼륨 제한

Nitro 시스템 (p. 118)에 구축된 인스턴스는 네트워크 인터페이스, EBS 볼륨 및 NVMe 인스턴스 스토어 볼륨 간에 공유되는 최대 연결 수를 지원합니다. 모든 인스턴스에는 최소 한 개의 네트워크 인터페이스 연결이 있습니다. NVMe 인스턴스 스토어 볼륨은 자동으로 연결됩니다. 자세한 내용은 [탄력적 네트워크 인터페이스](#) (p. 729) 및 [인스턴스 스토어 볼륨](#) (p. 1093) 단원을 참조하십시오.

이러한 인스턴스는 대부분 최대 28개의 연결을 지원합니다. 예를 들어, EBS 전용 인스턴스에 그 밖의 네트워크 인터페이스 연결이 없는 경우, 해당 인터페이스에 EBS 볼륨을 최대 27개까지 연결할 수 있습니다. 인스턴스에 NVMe 인스턴스 스토어 볼륨이 2개인 추가 네트워크 인터페이스가 1개 있는 경우 해당 인터페이스에 EBS 볼륨 24개를 연결할 수 있습니다.

다른 인스턴스의 경우 다음 제한이 적용됩니다.

- 대부분의 베어 메탈 인스턴스는 최대 31개의 EBS 볼륨을 지원합니다.
- u-6tb1.metal, u-9tb1.metal 및 u-12tb1.metal 인스턴스는 2020년 3월 12일 이후에 시작된 경우 최대 19개의 EBS 볼륨을 지원하고, 그렇지 않은 경우 최대 14개의 EBS 볼륨을 지원합니다. 2020년 3월 12일 이전에 시작된 인스턴스에서 14개가 넘는 EBS 볼륨을 연결하려면 계정 팀에 문의하여 추가 비용 없이 인스턴스를 업그레이드하십시오.
- u-18tb1.metal 및 u-24tb1.metal 인스턴스는 최대 19개의 EBS 볼륨을 지원합니다.

Windows 볼륨 제한

다음 표는 사용된 드라이버를 기반으로 Windows 인스턴스에 대한 볼륨 제한을 보여 줍니다. 이러한 개수에는 루트 볼륨과 함께 연결된 인스턴스 스토어 볼륨과 EBS 볼륨이 모두 포함됩니다.

Important

Windows 인스턴스에 다음 개수 이상의 볼륨을 연결할 수 있도록 최상의 노력이 제공되지만 보장되지는 않습니다.

드라이버	볼륨 제한
AWS PV	26
Citrix PV	26
Red Hat PV	17

성능 문제가 발생할 수 있으므로 AWS PV 또는 Citrix PV 드라이버를 사용하는 Windows 인스턴스에 볼륨을 26개 이상 연결하지 않는 것이 좋습니다.

인스턴스에서 사용하는 PV 드라이버를 확인하거나 Red Hat에서 Citrix PV 드라이버로 Windows 인스턴스를 업그레이드하려면 [Windows 인스턴스에서 PV 드라이버 업그레이드](#) (p. 528)를 참조하십시오.

디바이스 이름이 볼륨과 연결된 방식에 대한 자세한 내용은 [Windows 인스턴스의 볼륨에 디스크 맵핑](#) (p. 1116)을 참조하십시오.

대역폭 및 용량 비교

대역폭을 일관성 있고 예측 가능하게 사용하기 위해서는 EBS에 최적화된 인스턴스 또는 10Gb 네트워크에 연결된 인스턴스 및 범용 SSD 또는 프로비저닝된 IOPS SSD 볼륨을 사용해야 합니다. [Amazon EBS-최적화](#)

인스턴스 (p. 1049)의 지침을 준수하여 볼륨에 프로비저닝된 IPOS와 인스턴스가 최대 성능에서 사용 가능한 대역폭을 일치시킵니다. RAID 구성의 경우 볼륨이 8개 이상인 어레이에는 I/O 오버헤드가 증가하여 성능이 저하되는 결과가 여러 관리자에 의해 관찰되었습니다. 따라서 개별 애플리케이션의 성능을 테스트한 다음 필요에 따라 조정하십시오.

Windows 인스턴스의 디바이스 명명

볼륨을 인스턴스에 연결할 때 해당 볼륨에 대한 디바이스 이름을 포함합니다. 이 디바이스 이름은 Amazon EC2에서 사용합니다. 인스턴스의 블록 디바이스 드라이버는 볼륨이 마운트될 때 실제 볼륨 이름을 할당하고 할당된 이름은 Amazon EC2에서 사용하는 이름과 다를 수 있습니다.

인스턴스에 지원할 수 있는 볼륨의 수는 운영 체제에 따라 결정됩니다. 자세한 내용은 [인스턴스 볼륨 제한 \(p. 1104\)](#) 단원을 참조하십시오.

내용

- [사용 가능한 디바이스 이름 \(p. 1106\)](#)
- [디바이스 이름 고려 사항 \(p. 1107\)](#)

Linux 인스턴스의 디바이스 이름에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Linux 인스턴스에서의 디바이스 이름 지정](#) 단원을 참조하십시오.

사용 가능한 디바이스 이름

Windows AMI는 AWS PV, Citrix PV, RedHat PV 드라이버 세트 중 하나를 이용해 가상화 하드웨어에 대한 액세스를 허용합니다. 자세한 내용은 [Windows 인스턴스의 반가상화\(PV\) 드라이버 \(p. 523\)](#) 단원을 참조하십시오.

다음 표는 블록 디바이스 매핑에서 또는 EBS 볼륨에 연결 시 지정할 수 있는 사용 가능한 디바이스 이름을 나열합니다.

드라이버 유형	응시 가능	루트 전용	EBS 볼륨 추천	인스턴스 스토어 볼륨
AWS PV, Citrix PV	xvd[b-z] xvd[b-c][a-z] /dev/sda1 /dev/sd[b-e]	/dev/sda1	xvd[f-z] * **	xvdc[a-x] xvd[a-e] **
Red Hat PV	xvd[a-z] xvd[b-c][a-z] /dev/sda1 /dev/sd[b-e]	/dev/sda1	xvd[f-p]	xvdc[a-x] xvd[a-e]

* Citrix PV와 Red Hat PV의 경우 EBS 볼륨에 xvda 이름을 매핑하면, Windows는 볼륨을 인식하지 못합니다 (볼륨은 AWS PV나 AWS NVMe에서 확인할 수 있습니다).

** NVMe 인스턴스 스토어 볼륨은 자동으로 열거되고 Windows 드라이브 문자가 할당됩니다.

인스턴스 스토어 볼륨에 대한 자세한 내용은 [Amazon EC2 인스턴스 스토어 \(p. 1091\)](#) 단원을 참조하십시오. EBS 디바이스를 식별하는 방법을 포함하여 NVMe EBS 볼륨(Nitro 기반 인스턴스)에 대한 자세한 내용은 [Windows 인스턴스의 Amazon EBS 및 NVMe \(p. 1048\)](#) 단원을 참조하십시오.

디바이스 이름 고려 사항

디바이스 이름을 선택할 때는 다음 사항에 주의하십시오.

- 인스턴스 스토어 볼륨을 연결할 때 사용된 디바이스 이름을 사용하여 EBS 볼륨을 연결할 수 있지만, 이러한 경우 예기치 않은 동작이 발생할 수 있으므로 수행하지 않는 것이 좋습니다.
- 인스턴스의 NVMe 인스턴스 스토어 볼륨의 수는 인스턴스의 크기에 따라 다릅니다. NVMe 인스턴스 스토어 볼륨은 자동으로 열거되고 Windows 드라이브 문자가 할당됩니다.
- AWS Windows AMI는 처음 가동 시 인스턴스를 준비하는 추가 소프트웨어와 함께 제공됩니다. EC2Config 서비스(Windows Server 2016 이전의 Windows AMI) 또는 EC2Launch(Windows Server 2016 이상)입니다. 디바이스가 드라이브에 매핑된 후 이 서비스가 초기화되고 마운트됩니다. 루트 디바이스는 초기화된 다음 c:\로 마운트됩니다. 인스턴스에 연결된 인스턴스 스토어 볼륨은 초기화되어 z:\, y:\ 등으로 마운트됩니다. 기본적으로, EBS 볼륨이 Windows 인스턴스에 연결되면 인스턴스에서 드라이브 문자로 표시됩니다. 설정을 변경하여 사양에 따라 볼륨의 드라이브 문자를 설정할 수 있습니다. 자세한 내용은 [EC2Config 서비스를 사용한 Windows 인스턴스 구성 \(p. 496\)](#), [EC2Launch를 사용한 Windows 인스턴스 구성 \(p. 489\)](#), [Windows 인스턴스의 볼륨에 디스크 매핑 \(p. 1116\)](#) 단원을 참조하십시오.

블록 디바이스 매핑

시작한 각 인스턴스에는 연결된 루트 디바이스 볼륨(Amazon EBS 볼륨 또는 인스턴스 스토어 볼륨)이 있습니다. 블록 디바이스 매핑을 사용하면 실행될 때 인스턴스에 연결할 추가 EBS 볼륨 또는 인스턴스 스토어 볼륨을 지정할 수 있습니다. 또한, 실행 중인 인스턴스에 EBS 볼륨을 추가로 연결할 수도 있습니다. [Amazon EBS 볼륨을 인스턴스에 연결 \(p. 950\)](#)을 참조하십시오. 그러나 블록 디바이스 매핑을 사용하여 인스턴스가 시작되었을 때 볼륨을 연결하는 방식으로만 인스턴스에 인스턴스 스토어 볼륨을 연결할 수 있습니다.

루트 디바이스 볼륨에 대한 자세한 내용은 [루트 디바이스 볼륨 \(p. 7\)](#) 단원을 참조하십시오.

목차

- [블록 디바이스 매핑의 개념 \(p. 1107\)](#)
- [AMI 블록 디바이스 매핑 \(p. 1110\)](#)
- [인스턴스 블록 디바이스 매핑 \(p. 1112\)](#)

블록 디바이스 매핑의 개념

블록 디바이스는 바이트 또는 비트(블록) 단위로 순차적으로 데이터를 이동시키는 스토리지 디바이스입니다. 이러한 디바이스는 임의 액세스를 지원하고 일반적으로 버퍼 I/O를 사용합니다. 그러한 예로는 하드 디스크, CD-ROM 드라이브, 플래시 드라이브 등이 있습니다. 블록 디바이스는 컴퓨터에 물리적으로 장착될 수 있고 그렇지 않은 경우 컴퓨터에 물리적으로 장착된 것처럼 임의 액세스가 가능합니다.

Amazon EC2가 지원하는 두 가지 블록 디바이스 유형:

- 인스턴스 스토어 볼륨(기본 하드웨어가 인스턴스의 호스트 컴퓨터에 물리적으로 장착된 가상 디바이스)
- EBS 볼륨(원격 스토리지 디바이스)

블록 디바이스 매핑은 인스턴스에 연결할 블록 디바이스(인스턴스 볼륨 및 EBS 볼륨)를 정의합니다. AMI 생성 시 블록 디바이스 매핑을 지정하면 AMI에서 실행되는 모든 인스턴스가 해당 매핑을 사용할 수 있습니다.

아니면, 인스턴스 생성 시 블록 디바이스 매핑을 지정하여 이 매핑이 인스턴스가 실행된 AMI에서 지정된 매핑을 재정의하도록 할 수 있습니다. 인스턴스 유형에서 지원되는 모든 NVMe 인스턴스 스토어 볼륨이 인스턴스 시작 시 자동으로 열거되고 디바이스 이름이 할당됩니다. 따라서 블록 디바이스 매핑에 이를 포함하는 것은 효과가 없습니다.

목차

- [블록 디바이스 매핑 항목 \(p. 1108\)](#)
- [블록 디바이스 매핑 인스턴스 스토어 경고 \(p. 1108\)](#)
- [블록 디바이스 매핑 예제 \(p. 1109\)](#)
- [운영 체제에서 디바이스 사용 방법 \(p. 1109\)](#)

블록 디바이스 매핑 항목

블록 디바이스 매핑을 생성할 때 인스턴스에 연결할 각 블록 디바이스에 다음 정보를 지정합니다.

- Amazon EC2 내에서 사용되는 디바이스 이름 볼륨을 마운트할 때 인스턴스용 블록 디바이스 드라이버가 실제 볼륨 이름을 할당합니다. 할당된 이름이 Amazon EC2에서 권장하는 이름과 다를 수 있습니다. 자세한 내용은 [Windows 인스턴스의 디바이스 명명 \(p. 1106\)](#) 단원을 참조하십시오.
- [인스턴스 스토어 볼륨] 가상 디바이스: `ephemeral[0-23]` 그러나 이러한 볼륨의 개수 및 크기는 인스턴스 유형에 따라 다른 인스턴스에서 사용 가능한 인스턴스 스토어 볼륨을 초과하지 않아야 합니다.
- [NVMe 인스턴스 스토어 볼륨] 이러한 볼륨은 자동으로 열거되고 디바이스 이름이 할당되므로 블록 디바이스 매핑에 이를 포함하는 것은 효과가 없습니다.
- [EBS 볼륨] 블록 디바이스를 생성하기 위해 사용하는 스냅샷 ID(snap-xxxxxxxx) 볼륨 크기를 지정하는 경우 이 값은 선택 사항입니다.
- [EBS; 볼륨] GiB 단위의 볼륨 크기 지정된 크기는 지정된 스냅샷 크기 이상이어야 합니다.
- [EBS; 볼륨] 인스턴스 종료 시 볼륨 삭제 여부(true 또는 false) 기본값은 루트 디바이스 볼륨은 true이고 연결된 볼륨은 false입니다 AMI를 생성하면 그 블록 디바이스 매핑이 인스턴스에서 이 설정을 내려 받습니다. 인스턴스를 시작하면 AMI에서 이 설정을 내려 받습니다.
- [EBS 볼륨] 볼륨 유형은 범용 SSD 볼륨의 경우 `gp2`, 프로비저닝된 IOPS SSD 볼륨의 경우 `io1` 또는 `io2`, 처리량에 최적화된 HDD 볼륨의 경우 `st1`, Cold HDD 볼륨의 경우 `sc1`, Magnetic 볼륨의 경우 `standard`일 수 있습니다. 기본 값은 `gp2`입니다.
- [EBS 볼륨] 볼륨이 지원하는 초당 입력/출력 작업 수(IOPS) (`gp2`, `st1`, `sc1` 또는 `standard` 볼륨에서는 사용되지 않음.)

블록 디바이스 매핑 인스턴스 스토어 경고

블록 디바이스 매핑에 인스턴스 스토어 볼륨이 있는 AMIs에서 인스턴스를 시작하는 경우 고려해야 할 몇 가지 경고 사항이 있습니다.

- 일부 인스턴스 유형은 다른 인스턴스보다 인스턴스 스토어 볼륨이 더 있거나 어떤 인스턴스 유형은 인스턴스 스토어 볼륨이 아예 없을 수도 있습니다. 인스턴스 볼륨이 1개의 인스턴스 스토어 볼륨을 지원하는 데 AMI에 2개의 인스턴스 스토어 볼륨이 있는 경우 인스턴스는 1개의 인스턴스 스토어 볼륨으로 실행됩니다.
- 인스턴스 스토어 볼륨은 실행 시에만 매핑될 수 있습니다. 인스턴스 스토어 볼륨이 없는 인스턴스 (`t2.micro` 등)는 중지할 수 없으므로 해당 인스턴스를 인스턴스 스토어 볼륨을 지원하는 유형으로 변경한 다음 인스턴스 스토어 볼륨이 있는 인스턴스를 다시 시작합니다. 그러나 인스턴스에서 AMI를 생성하고 인스턴스 스토어 볼륨을 지원하는 인스턴스 유형에서 실행한 다음 그러한 인스턴스 스토어 볼륨을 인스턴스로 매핑하는 것은 가능합니다.
- 인스턴스 스토어 볼륨이 있는 매핑된 인스턴스를 실행한 다음 인스턴스를 중지하고 인스턴스 스토어 볼륨의 개수가 적은 인스턴스 유형으로 변경한 후 다시 시작한 경우 인스턴스 메타데이터에는 처음 실행된 인스턴스 스토어 볼륨 매핑이 계속해서 표시됩니다. 그러나 그러한 인스턴스에서는 해당 인스턴스 유형에서 지원되는 최대 인스턴스 스토어 볼륨 갯수만 사용할 수 있습니다.

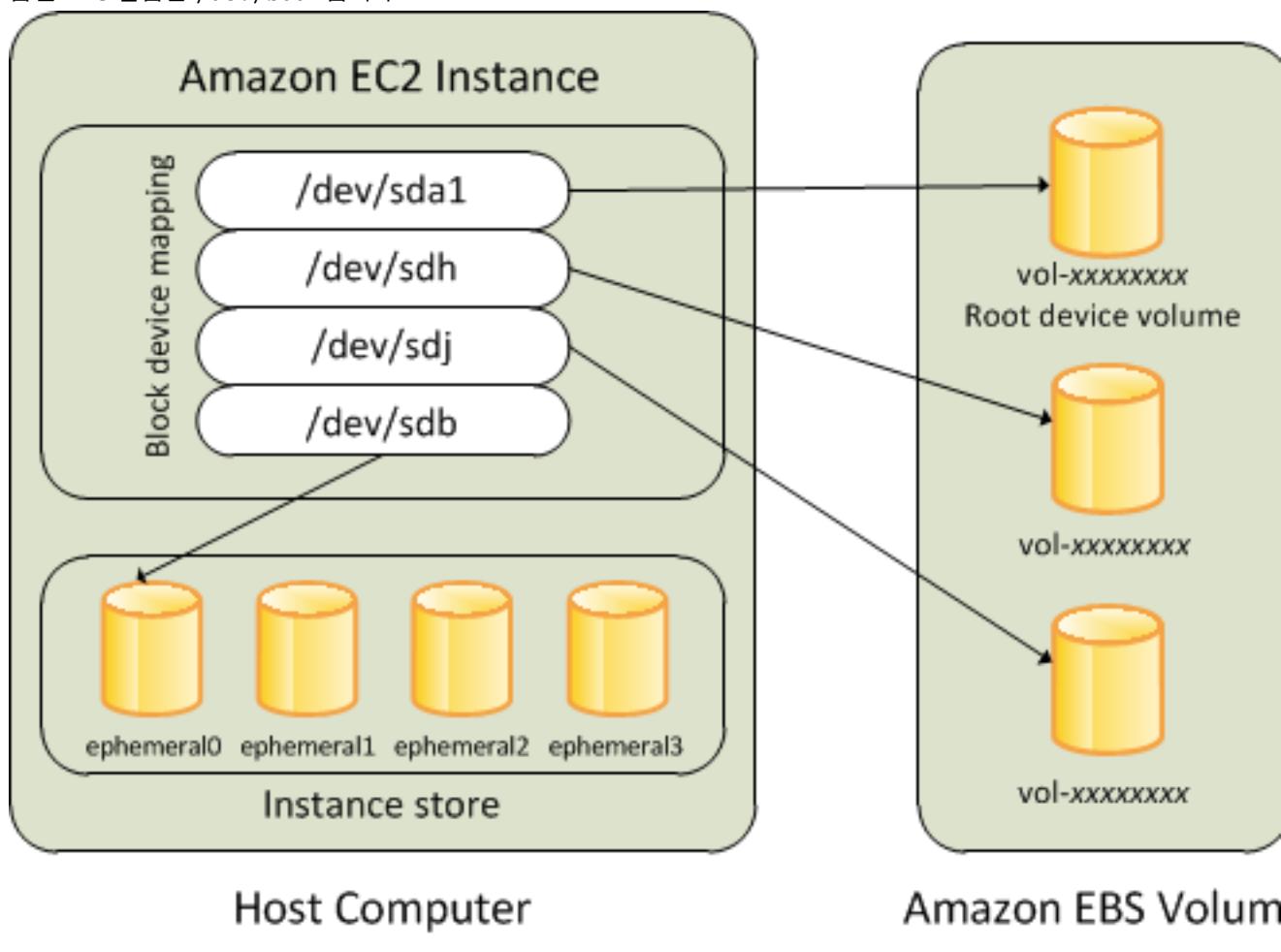
Note

인스턴스가 종지되면 인스턴스 스토어 볼륨의 모든 데이터가 손실됩니다.

- 실행 시의 인스턴스 스토어 용량에 따라 실행 시 지정되지 않는 경우 M3 인스턴스는 실행되는 AMI 인스턴스 스토어 블록 디바이스 매핑을 무시할 수 있습니다. 실행하려는 AMI에 AMI 매핑 인스턴스 스토어 볼륨이 있는 경우 실행 시 인스턴스 스토어 블록 디바이스 매핑을 지정해야 인스턴스가 실행될 때 인스턴스 스토어 볼륨을 사용할 수 있습니다.

블록 디바이스 매핑 예제

이 그림은 EBS 기반 인스턴스의 블록 디바이스 매핑 예제를 보여줍니다. `/dev/sdb`를 `ephemeral0`으로 매핑하고 두 개의 EBS 볼륨을 각각 `/dev/sdh` 및 `/dev/sdj`로 매핑합니다. 또한 여기에서 루트 디바이스 볼륨인 EBS 볼륨은 `/dev/sda1`입니다.



이 예제 블록 디바이스 매핑에서는 이 주제와 관련된 예제 명령어 및 API가 사용되었습니다. [AMI용 블록 디바이스 매핑 지정 \(p. 1110\)](#) 및 [인스턴스를 시작할 때 블록 디바이스 매핑 업데이트 \(p. 1112\)](#)에서 블록 디바이스 매핑을 생성하는 API와 예제 명령을 확인할 수 있습니다.

운영 체제에서 디바이스 사용 방법

`/dev/sdh` 및 `xvdh` 등의 디바이스 이름은 Amazon EC2에서 블록 디바이스를 나타내는 이름으로 사용됩니다. Amazon EC2에서 블록 디바이스 매핑은 EC2 인스턴스를 연결하는 블록 디바이스를 지정하는 데 사용됩니다.

니다. 블록 디바이스가 인스턴스에 연결되면 운영 체제에 마운트되어야 사용자가 해당 스토리지 디바이스에 액세스할 수 있습니다. 블록 디바이스가 인스턴스에서 분리되면 운영 체제에서 마운트가 해제되고 사용자는 더 이상 해당 스토리지 디바이스에 액세스할 수 없습니다.

Windows 인스턴스의 경우 블록 디바이스 매핑에 지정된 디바이스 이름은 인스턴스가 처음 부팅될 때 해당하는 블록 디바이스로 매핑된 후 Ec2Config 서비스가 초기화되고 드라이브가 마운트됩니다. 루트 디바이스 볼륨은 c:\로 마운트됩니다. 인스턴스 스토어 볼륨은 z:\, y:\ 등으로 마운트됩니다. EBS 볼륨이 마운트될 때는 사용 가능한 드라이브 문자를 사용하여 마운트될 수 있습니다. 그러나 사용자가 Ec2Config 서비스에서 EBS 볼륨에 드라이브 문자를 지정하는 방법을 구성할 수 있습니다. 자세한 내용은 [EC2Config 서비스를 사용한 Windows 인스턴스 구성 \(p. 496\)](#)을 참조하십시오.

AMI 블록 디바이스 매핑

각 AMI에는 AMI에서 시작될 때 인스턴스로 연결될 블록 디바이스를 지정하는 블록 디바이스 매핑이 있습니다. Amazon은 루트 디바이스가 포함된 AMI만을 지원합니다. AMI에 추가 블록 디바이스를 추가하려면 고유 AMI를 생성해야 합니다.

내용

- [AMI용 블록 디바이스 매핑 지정 \(p. 1110\)](#)
- [AMI 블록 디바이스 매핑에서 EBS 볼륨 보기 \(p. 1111\)](#)

AMI용 블록 디바이스 매핑 지정

두 가지 방법으로 AMI를 생성할 때 루트 디바이스 볼륨과 볼륨을 지정할 수 있습니다. 인스턴스에서 AMI를 생성하기 전 실행 중인 인스턴스에 볼륨을 이미 연결한 경우 AMI용 블록 디바이스 매핑에는 동일한 해당 볼륨이 포함됩니다. EBS 볼륨에서 기존 데이터는 새 스냅샷에 저장되고 블록 디바이스 매핑에 새로운 이 스냅샷이 지정됩니다. 인스턴스 스토어 볼륨의 경우 데이터는 보존되지 않습니다.

EBS 기반 AMI의 경우 블록 디바이스 매핑을 사용하여 EBS 볼륨 및 인스턴스 스토어 볼륨을 추가할 수 있습니다. 인스턴스 스토어 지원 AMI의 경우 이미지를 등록할 때 이미지 매니페스트 파일에서 블록 디바이스 매핑 항목을 수정하여 인스턴스 스토어 볼륨만 추가할 수 있습니다.

Note

M3 인스턴스의 경우 실행 시 인스턴스에 대한 블록 디바이스 매핑에 인스턴스 스토어 볼륨을 반드시 지정해야 합니다. M3 인스턴스 실행 시 인스턴스 스토어 볼륨이 인스턴스 블록 디바이스 매핑으로 지정되지 않으면 AMI용 블록 디바이스 매핑에 지정된 인스턴스 스토어 볼륨이 무시될 수 있습니다.

콘솔을 사용하여 AMI에 볼륨을 추가하려면

1. Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 인스턴스를 선택하고 작업, 이미지, 이미지 생성을 선택합니다.
4. 이미지 생성 대화 상자에서 새 볼륨 추가를 선택합니다.
5. 유형 목록에서 볼륨 유형을 선택하고 디바이스 목록에서 디바이스 이름을 선택합니다. EBS 볼륨의 경우 스냅샷, 볼륨 크기와 볼륨 유형을 선택적으로 지정할 수 있습니다.
6. 이미지 생성을 선택합니다.

명령줄을 사용하여 AMI에 볼륨을 추가하려면

[create-image](#) AWS CLI 명령을 사용하여 EBS 지원 AMI에 블록 디바이스 매핑을 지정합니다. [register-image](#) AWS CLI 명령을 사용하여 인스턴스 스토어 지원 AMI에 블록 디바이스 매핑을 지정합니다.

--block-device-mappings 파라미터를 사용하여 블록 디바이스 매핑을 지정합니다. JSON으로 인코딩된 인수는 명령 줄에서 직접 제공하거나 파일 참조로 제공할 수 있습니다.

```
--block-device-mappings [mapping, ...]  
--block-device-mappings [file://mapping.json]
```

인스턴스 스토어 볼륨을 추가하려면 다음 매핑을 사용합니다.

```
{  
    "DeviceName": "xvdb",  
    "VirtualName": "ephemeral0"  
}
```

비어 있는 100GiB gp2 볼륨을 추가하려면 다음 매핑을 사용합니다.

```
{  
    "DeviceName": "xvdg",  
    "Ebs": {  
        "VolumeSize": 100  
    }  
}
```

스냅샷 기반 EBS 볼륨을 추가하려면 다음 매핑을 사용합니다.

```
{  
    "DeviceName": "xvdh",  
    "Ebs": {  
        "SnapshotId": "snap-xxxxxxxx"  
    }  
}
```

디바이스에 대한 매핑을 생략하려면 다음 매핑을 사용합니다.

```
{  
    "DeviceName": "xvdj",  
    "NoDevice": ""  
}
```

또는 다음 명령(Windows PowerShell용 AWS 도구)과 함께 -BlockDeviceMapping 파라미터를 사용할 수 있습니다.

- [New-EC2Image](#)
- [Register-EC2Image](#)

AMI 블록 디바이스 매핑에서 EBS 볼륨 보기

AMI의 블록 디바이스 매핑에서 EBS 볼륨을 쉽게 확인할 수 있습니다.

콘솔을 사용하여 AMI용 EBS 볼륨을 확인하려면

1. Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 AMI를 선택합니다.
3. 필터 목록에서 EBS 이미지를 선택하여 EBS 지원 AMI 목록을 가져옵니다.

- 원하는 AMI를 선택한 후 세부 정보 탭을 확인합니다. 루트 디바이스에서 최소한으로 사용 가능한 정보는 다음과 같습니다.

- 루트 디바이스 유형 (ebs)
- Root Device Name(예: /dev/sda1)
- 블록 디바이스(예: /dev/sda1=snap-1234567890abcdef0:8:true)

AMI가 블록 디바이스 매핑을 사용하여 추가 EBS 볼륨으로 생성된 경우 블록 디바이스 필드에 해당 추가 볼륨에 대한 매핑도 표시됩니다. (이 화면에는 인스턴스 스토어 볼륨이 표시되지 않습니다.)

명령줄을 사용하여 AMI의 EBS 볼륨을 보려면

[describe-images](#)(AWS CLI) 명령 또는 [Get-EC2Image](#)(Windows PowerShell용 AWS 도구) 명령을 사용하여 AMI용 블록 디바이스 매핑에 EBS 볼륨을 표시합니다.

인스턴스 블록 디바이스 매핑

기본적으로, 사용자가 실행한 인스턴스에는 인스턴스가 실행된 AMI의 블록 디바이스 매핑에 지정된 모든 스토리지 디바이스가 포함됩니다. 인스턴스 실행 시 해당 인스턴스에 대한 블록 디바이스 매핑을 변경하면 해당 업데이트는 AMI의 블록 디바이스 매핑을 덮어 쓰거나 병합됩니다.

제한 사항

- 루트 볼륨에서는 다음 항목만 수정할 수 있습니다. 볼륨 크기, 볼륨 유형 및 종료 시 삭제 여부 플래그.
- EBS 볼륨을 수정할 때 크기는 줄일 수 없습니다. 그러므로 AMI의 블록 디바이스 매핑에서 지정된 스냅샷과 크기가 같거나 큰 스냅샷을 지정해야 합니다.

목차

- [인스턴스를 시작할 때 블록 디바이스 매핑 업데이트 \(p. 1112\)](#)
- [실행 중인 인스턴스의 블록 디바이스 매핑 업데이트 \(p. 1114\)](#)
- [인스턴스 블록 디바이스 매핑에서 EBS 볼륨 보기 \(p. 1114\)](#)
- [인스턴스 스토어 볼륨용 인스턴스 블록 디바이스 매핑 보기 \(p. 1115\)](#)

인스턴스를 시작할 때 블록 디바이스 매핑 업데이트

실행 시 인스턴스에 EBS 볼륨 및 인스턴스 스토어 볼륨을 추가할 수 있습니다. 인스턴스의 블록 디바이스 매핑을 업데이트해도 인스턴스가 실행된 AMI의 블록 디바이스 매핑이 영구적으로 변경되는 것은 아님에 주의하십시오.

콘솔을 사용하여 인스턴스에 볼륨을 추가하려면

- Amazon EC2 콘솔을 엽니다.
- 대시보드에서 인스턴스 시작을 선택합니다.
- Amazon 머신 이미지(AMI) 선택 페이지에서 사용할 AMI를 선택하고 선택을 선택합니다.
- 마법사 안내에 따라 인스턴스 유형 선택 및 인스턴스 세부 정보 구성 설정을 완료합니다.
- 스토리지 추가 페이지에서 루트 볼륨, EBS 볼륨 및 인스턴스 스토어 볼륨을 다음과 같이 수정할 수 있습니다.
 - 루트 볼륨 크기를 변경하려면 유형 열 아래에 있는 루트 볼륨으로 이동한 후 크기 필드를 변경합니다.
 - 인스턴스를 실행하는 데 사용된 AMI의 블록 디바이스 매핑에서 지정된 EBS 볼륨을 표시하지 않으려면 해당 볼륨을 찾아 Delete 아이콘을 클릭합니다.

- EBS 볼륨을 추가하려면 새 볼륨 추가를 선택하고 유형 목록에서 EBS를 선택한 후 필드(디바이스, 스냅샷 등)를 작성합니다.
 - 인스턴스가 실행된 AMI의 블록 디바이스 매핑에서 지정된 인스턴스 스토어 볼륨을 표시하지 않으려면 해당 볼륨으로 이동한 다음 삭제 아이콘을 선택합니다.
 - 인스턴스 스토어 볼륨을 추가하려면 새 볼륨 추가를 선택하고, 유형 목록에서 인스턴스 스토어를 선택한 후 디바이스에서 디바이스 이름을 선택합니다.
6. 나머지 마법사 페이지를 완료한 다음 시작을 선택합니다.

AWS CLI를 사용하여 인스턴스에 볼륨을 추가하려면

`run-instances` AWS CLI 명령을 `--block-device-mappings` 옵션과 함께 사용하여 시작 시 인스턴스에 블록 디바이스 매핑을 지정합니다.

예를 들어, EBS 기반 AMI가 다음 블록 디바이스 매핑을 지정한다고 가정하면,

- `xvdb=ephemeral0`
- `xvdh=snap-1234567890abcdef0`
- `xvdj=:100`

이 AMI에서 시작된 인스턴스에 `xvdj`가 연결되지 않게 하려면 다음 매핑을 사용합니다.

```
{  
    "DeviceName": "xvdj",  
    "NoDevice": ""  
}
```

`xvdh`의 크기를 300GiB로 늘리려면, 다음 매핑을 지정합니다. 디바이스 이름을 지정하면 볼륨을 식별하는 데 충분하므로 `xvdh`에 스냅샷 ID를 지정할 필요가 없음에 유의하십시오.

```
{  
    "DeviceName": "xvdh",  
    "Ebs": {  
        "VolumeSize": 300  
    }  
}
```

인스턴스 시작 시 루트 볼륨의 크기를 늘리려면 먼저 해당 AMI ID를 가진 `describe-images`를 호출하여 루트 볼륨의 디바이스 이름을 확인합니다. 예: `"RootDeviceName": "/dev/xvda"`. 루트 볼륨의 크기를 재정의하려면 AMI에서 사용하는 루트 디바이스의 디바이스 이름과 새 볼륨 크기를 지정합니다.

```
{  
    "DeviceName": "/dev/xvda",  
    "Ebs": {  
        "VolumeSize": 100  
    }  
}
```

추가 인스턴스 스토어 볼륨 `xvdc`를 연결하려면 다음 매핑을 지정합니다. 다중 인스턴스 스토어 볼륨을 지원하지 않는 인스턴스 유형의 경우 이 매핑은 영향을 미치지 않습니다. 인스턴스가 NVMe 인스턴스 스토어 볼륨을 지원하는 경우 해당 볼륨이 자동으로 열거되고 NVMe 디바이스 이름이 할당됩니다.

```
{  
    "DeviceName": "xvdc",  
    "VirtualName": "ephemeral1"
```

}

Windows PowerShell용 AWS 도구를 사용하여 인스턴스에 볼륨을 추가하려면

[New-EC2Instance](#) 명령(Windows PowerShell용 AWS 도구)에서 `-BlockDeviceMapping` 파라미터를 사용합니다.

실행 중인 인스턴스의 블록 디바이스 매핑 업데이트

[modify-instance-attribute](#) AWS CLI 명령을 사용하여 실행 중인 인스턴스의 블록 디바이스 매핑을 업데이트 할 수 있습니다. 이 속성을 변경하기 전에 인스턴스를 종지할 필요는 없습니다.

```
aws ec2 modify-instance-attribute --instance-id i-1a2b3c4d --block-device-mappings file://mapping.json
```

예를 들어, 인스턴스 종료 시 루트 볼륨을 유지하려면 `mapping.json`에서 다음을 지정합니다.

```
[  
 {  
   "DeviceName": "/dev/sda1",  
   "Ebs": {  
     "DeleteOnTermination": false  
   }  
 }  
]
```

또는 [Edit-EC2InstanceAttribute](#) 명령(Windows PowerShell용 AWS 도구)과 함께 `-BlockDeviceMapping` 파라미터를 사용할 수 있습니다.

인스턴스 블록 디바이스 매핑에서 EBS 볼륨 보기

인스턴스에 매핑된 EBS 볼륨을 쉽게 확인할 수 있습니다.

Note

2009-10-31 API 릴리스 이전에 실행된 인스턴스의 경우 AWS는 블록 디바이스 매핑을 표시할 수 없습니다. 반드시 해당 볼륨을 분리 후 다시 연결해야 블록 디바이스 매핑이 표시될 수 있습니다.

콘솔을 사용하여 인스턴스의 EBS 볼륨을 보려면

1. Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 검색 상자에 루트 디바이스 유형을 입력한 후 EBS를 선택합니다. 이렇게 하면 EBS 기반 인스턴스 목록이 표시됩니다.
4. 원하는 인스턴스를 선택한 후 설명 탭에 표시되는 세부 정보를 확인합니다. 루트 디바이스에서 최소한으로 사용 가능한 정보는 다음과 같습니다.
 - 루트 디바이스 유형 (ebs)
 - Root device(예: /dev/sda1)
 - 블록 디바이스(예: /dev/sda1, xvhd 및 xvhdj)

인스턴스가 블록 디바이스 매핑을 사용하여 추가 EBS 볼륨으로 실행된 경우 블록 디바이스 필드에 루트 디바이스와 함께 해당 추가 볼륨도 표시됩니다. (이 화면에는 인스턴스 스토어 볼륨이 표시되지 않습니다.)

Root device type	ebs
Root device	/dev/sda1
Block devices	/dev/sda1 /dev/sdf

5. 블록 디바이스에 대한 추가 정보를 표시하려면 블록 디바이스 옆의 항목을 클릭합니다. 그러면 블록 디바이스와 관련한 다음 정보가 표시됩니다.
- EBS ID(vol-xxxxxxxx)
 - 루트 디바이스 유형(ebs)
 - 연결 시간(yyyy-mmThh:mm:ss.ssTZD)
 - 블록 디바이스 상태(attach, attached, detach, detached)
 - 종료 시 삭제 여부(Yes, No)

명령줄을 사용하여 인스턴스의 EBS 볼륨을 보려면

[describe-instances](#)(AWS CLI) 명령 또는 [Get-EC2Instance](#)(Windows PowerShell용 AWS 도구) 명령을 사용하여 인스턴스용 블록 디바이스 매핑에 EBS 볼륨을 표시합니다.

인스턴스 스토어 볼륨용 인스턴스 블록 디바이스 매핑 보기

인스턴스에 대한 블록 디바이스 매핑을 볼 때 인스턴스 스토어 볼륨이 아닌 EBS 볼륨만 확인할 수 있습니다. 인스턴스 메타데이터를 사용하여 블록 디바이스 매핑에 있는 비 NVMe 인스턴스 스토어 볼륨을 쿼리할 수 있습니다. NVMe 인스턴스 스토어 볼륨은 포함되지 않습니다.

전체 인스턴스 메타데이터를 요청하기 위한 기본 URI는 <http://169.254.169.254/latest/>입니다. 자세한 내용은 [인스턴스 메타데이터 및 사용자 데이터\(p. 576\)](#) 단원을 참조하십시오.

우선, 실행 중인 인스턴스에 연결합니다. 인스턴스에서 이 쿼리를 사용하여 블록 디바이스 매핑을 가져옵니다.

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/block-device-mapping/
```

인스턴스에 대한 블록 디바이스 이름이 응답으로 제공됩니다. 예를 들어, 인스턴스 스토어 지원 m1.small 인스턴스에 대한 결과는 다음과 같습니다.

```
ami
ephemeral0
root
swap
```

인스턴스에서 보이는 것과 같이 ami 디바이스가 루트 디바이스입니다. 인스턴스 스토어 볼륨의 이름은 ephemeral[0-23]입니다. swap 디바이스는 페이지 파일용입니다. 또한, EBS 볼륨을 매핑한 경우 ebs1, ebs2 등으로 표시됩니다.

블록 디바이스 매핑 내 개별 블록 디바이스에 대한 세부 정보를 확인하려면 여기에서와 같이 이전 쿼리에 이름을 추가합니다.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
```

```
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

인스턴스 유형은 인스턴스에 사용 가능한 인스턴스 스토어 볼륨 수를 결정합니다. 블록 디바이스 매핑에 있는 인스턴스 스토어 볼륨 수가 인스턴스에 사용 가능한 인스턴스 스토어 볼륨 수를 초과한 경우 추가 볼륨이 무시됩니다. 인스턴스에 대한 인스턴스 스토어 볼륨을 확인하려면 Windows 디스크 관리를 엽니다. 각 인스턴스 유형에서 지원되는 인스턴스 스토어 볼륨 수를 알아보려면 [인스턴스 스토어 볼륨 \(p. 1093\)](#) 단원을 참조하십시오.

Windows 인스턴스의 볼륨에 디스크 매핑

Windows 인스턴스는 루트 볼륨의 기능을 하는 EBS 볼륨과 함께 제공됩니다. Windows 인스턴스가 AWS PV 또는 Citrix PV 드라이버를 사용하는 경우 최대 25개의 볼륨을 선택적으로 추가할 수 있으므로 총 볼륨은 26 개가 됩니다. 자세한 내용은 [인스턴스 볼륨 제한 \(p. 1104\)](#) 단원을 참조하십시오.

인스턴스의 인스턴스 유형에 따라 인스턴스는 0~24개의 인스턴스 스토어 볼륨을 가질 수 있습니다. 인스턴스에서 사용 가능한 인스턴스 스토어 볼륨을 사용하려면 AMI 생성 시 또는 인스턴스 시작 시에 인스턴스 스토어 볼륨을 지정해야 합니다. 또한 AMI를 생성하거나 인스턴스를 시작할 때 EBS 볼륨을 추가하거나, 인스턴스가 실행 중인 상태에서 해당 볼륨을 연결할 수 있습니다. 자세한 내용은 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기 \(p. 951\)](#) 단원을 참조하십시오.

인스턴스에 볼륨을 추가할 때 Amazon EC2에서 사용하는 디바이스 이름을 지정합니다. 자세한 내용은 [Windows 인스턴스의 디바이스 명명 \(p. 1106\)](#) 단원을 참조하십시오. AWS Windows Amazon 머신 이미지 (AMI)에는 Amazon EC2가 인스턴스 스토어 및 EBS 볼륨을 Windows 디스크 및 드라이브 문자에 매핑하기 위해 사용하는 드라이버 세트 1개가 포함됩니다. AWS PV 또는 Citrix PV 드라이버를 사용하는 Windows AMI에서 인스턴스를 시작한 경우, 이 페이지에서 설명된 관계를 활용하여 Windows 디스크를 인스턴스 스토어 및 EBS 볼륨에 매핑할 수 있습니다. Windows AMI가 Red Hat PV를 사용하는 경우 인스턴스를 업데이트하면 Citrix 드라이버를 사용할 수 있습니다. 자세한 내용은 [Windows 인스턴스에서 PV 드라이버 업그레이드 \(p. 528\)](#) 단원을 참조하십시오.

내용

- [Windows 디스크 관리를 이용하여 디스크 나열 \(p. 1116\)](#)
- [Windows PowerShell을 사용하여 디스크 나열\(Windows Server 2012 이상\) \(p. 1118\)](#)
- [디스크 디바이스와 디스크 이름 간 매핑 \(p. 1121\)](#)

Windows 디스크 관리를 이용하여 디스크 나열

Windows 디스크 관리를 사용하면 Windows 인스턴스에 있는 디스크를 검색할 수 있습니다.

Windows 인스턴스에 있는 디스크를 검색하려면

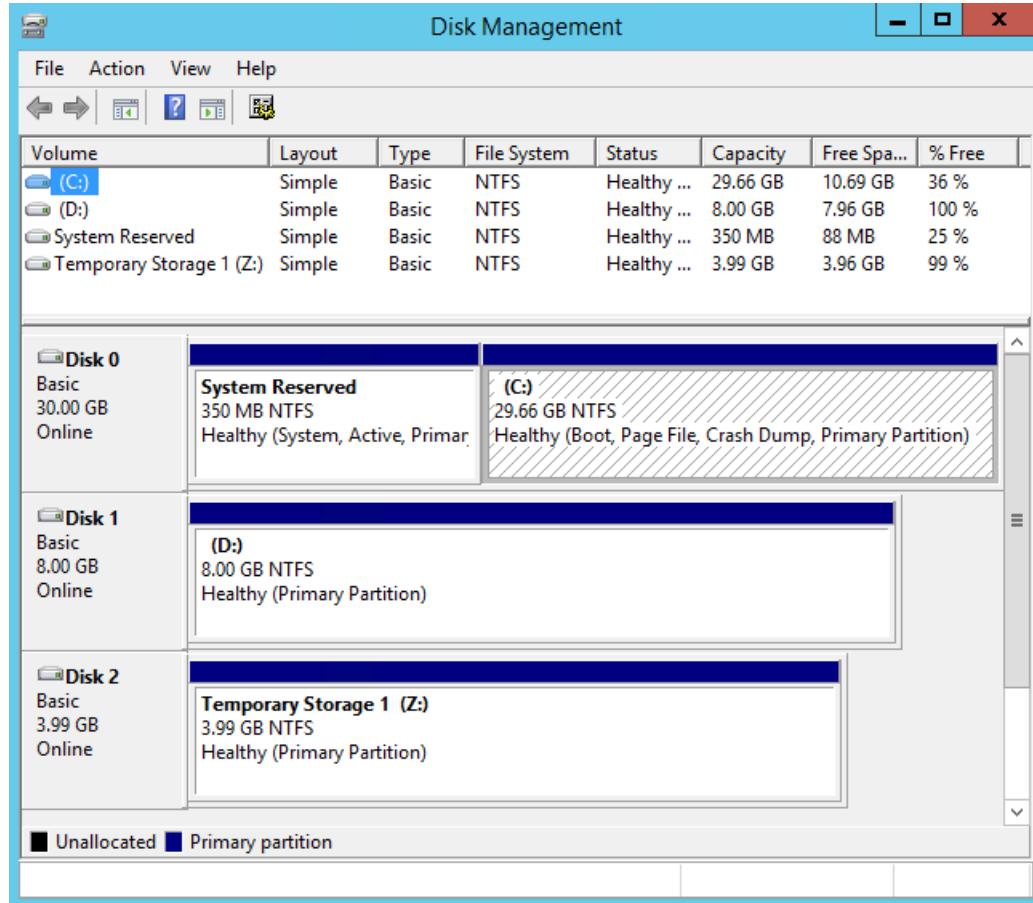
1. 원격 데스크톱을 사용하여 Windows 인스턴스에 로그인합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.

2. 디스크 관리 유틸리티를 시작합니다.

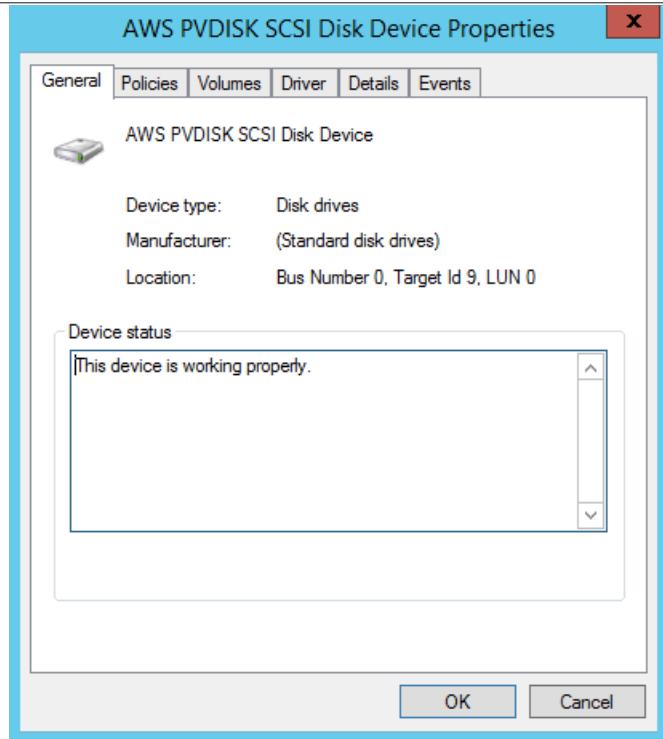
Windows Server 2012 이상의 경우 작업 표시줄에서 Windows 로고를 마우스 오른쪽 단추를 클릭한 다음 Disk Management(디스크 관리)를 선택합니다. Windows Server 2008에서는 시작(Start), 관리 도구(Administrative Tools), 컴퓨터 관리(Computer Management), 디스크 관리(Disk Management)를 선택합니다.

3. 디스크를 확인합니다. 루트 볼륨은 c:\로 마운트되는 EBS 볼륨입니다. 다른 디스크가 표시되지 않는 경우 AMI를 생성하거나 인스턴스를 시작할 때 추가 볼륨을 지정하지 않은 것입니다.

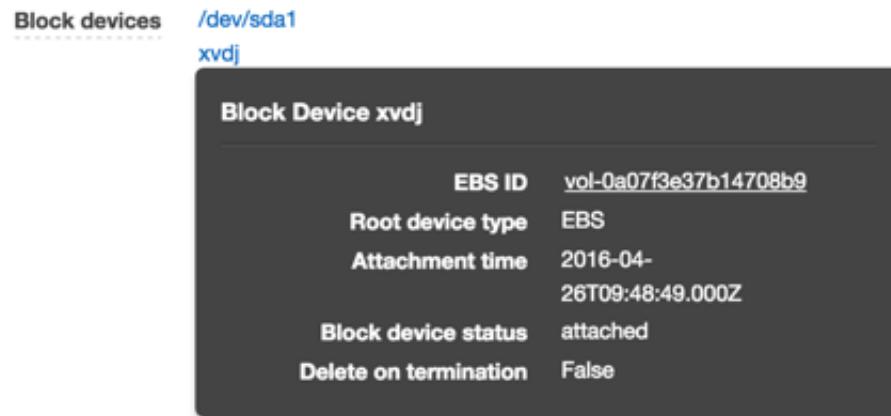
다음 예는 인스턴스 스토어 볼륨(디스크 2) 및 추가 EBS 볼륨(디스크 1)을 포함하여 m3.medium 인스턴스를 시작할 경우 사용 가능한 디스크를 보여줍니다.



4. 회색 창으로 표시된 디스크 1을 마우스 오른쪽 단추로 클릭한 후 속성을 선택합니다. 위치의 값을 [디스크 디바이스와 디스크 이름 간 매핑 \(p. 1121\)](#)의 표를 확인합니다. 예를 들어, 다음 디스크에는 버스 번호 0, 타겟 ID 9, LUN 0 위치가 있습니다. EBS 볼륨 테이블에 따르면 이 위치에 있는 디바이스 이름은 xvdf입니다.



- EBS 볼륨의 디바이스 이름을 볼륨 ID에 매핑하려면 컴퓨터에서 Amazon EC2 콘솔을 엽니다. 탐색 창에서 인스턴스를 선택한 후 인스턴스를 선택합니다. 설명 탭의 블록 디바이스에서 디바이스 이름을 클릭하고 EBS ID를 확인합니다. 이 예제에서 볼륨 ID는 vol-0a07f3e37b14708b9입니다.



Amazon EC2 콘솔에는 EBS 볼륨만 표시된다는 점을 유의하십시오.

콘솔에서 볼륨 ID를 보는 또 다른 방법은 탐색 창의 Elastic Block Store에서 볼륨을 선택하는 것입니다. 그러면 인스턴스 이름과 볼륨 크기별로 볼륨 목록이 표시됩니다. 연결 정보를 검토하거나 선택하여 볼륨이 연결된 인스턴스를 확인하십시오.

Windows PowerShell을 사용하여 디스크 나열 (Windows Server 2012 이상)

다음 PowerShell 스크립트는 각 디스크 및 해당 디바이스 이름과 볼륨을 목록으로 표시합니다.

Note

다음 스크립트는 동적 디스크에 대한 정보를 반환하지 않습니다.

```
# List the Windows disks

function Get-EC2InstanceMetadata {
    param([string]$Path)
    (Invoke-WebRequest -Uri "http://169.254.169.254/latest/$Path").Content
}

function Convert-SCSITargetIdToDeviceName {
    param([int]$SCSITargetId)
    If ($SCSITargetId -eq 0) {
        return "sda1"
    }
    $deviceName = "xvd"
    If ($SCSITargetId -gt 25) {
        $deviceName += [char](0x60 + [int]($SCSITargetId / 26))
    }
    $deviceName += [char](0x61 + $SCSITargetId % 26)
    return $deviceName
}

Try {
    $InstanceId = Get-EC2InstanceMetadata "meta-data/instance-id"
    $AZ = Get-EC2InstanceMetadata "meta-data/placement/availability-zone"
    $Region = $AZ.Remove($AZ.Length - 1)
    $BlockDeviceMappings = (Get-EC2Instance -Region $Region -Instance
    $InstanceId).Instances.BlockDeviceMappings
    $VirtualDeviceMap = @{}
    (Get-EC2InstanceMetadata "meta-data/block-device-mapping").Split("`n") | ForEach-Object {
        $VirtualDevice = $_
        $BlockDeviceName = Get-EC2InstanceMetadata "meta-data/block-device-mapping/
$VirtualDevice"
        $VirtualDeviceMap[$BlockDeviceName] = $VirtualDevice
        $VirtualDeviceMap[$VirtualDevice] = $BlockDeviceName
    }
}
Catch {
    Write-Host "Could not access the AWS API, therefore, VolumeId is not available.
Verify that you provided your access keys." -ForegroundColor Yellow
}

Get-disk | ForEach-Object {
    $DriveLetter = $null
    $VolumeName = $null

    $DiskDrive = $_
    $Disk = $_.Number
    $Partitions = $_.NumberOfPartitions
    $EbsVolumeID = $_.SerialNumber -replace "[^]*$" -replace "vol", "vol-"
    Get-Partition -DiskId $_.Path | ForEach-Object {
        if ($_.DriveLetter -ne "") {
            $DriveLetter = $_.DriveLetter
            $VolumeName = (Get-PSDrive | Where-Object {$_ .Name -eq $DriveLetter}).Description
        }
    }

    If ($DiskDrive.path -like "*PROD_PVDISK*") {
        $BlockDeviceName = Convert-SCSITargetIdToDeviceName((Get-WmiObject -
Class Win32_Diskdrive | Where-Object {$_ .DeviceID -eq ("\".\PHYSICALDRIVE" +
$DiskDrive.Number) }).SCSITargetId)
        $BlockDeviceName = "/dev/" + $BlockDeviceName
    }
}
```

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
Windows PowerShell을 사용하여 디
스크 나열(Windows Server 2012 이상)

```
$BlockDevice = $BlockDeviceMappings | Where-Object { $BlockDeviceName -like "*"+$_.DeviceName+"*" }
    $EbsVolumeID = $BlockDevice.Ebs.VolumeId
    $VirtualDevice = If ($VirtualDeviceMap.ContainsKey($BlockDeviceName))
    { $VirtualDeviceMap[$BlockDeviceName] } Else { $null }
}
ElseIf ($DiskDrive.path -like "*PROD_AMAZON_EC2_NVME*") {
    $BlockDeviceName = Get-EC2InstanceMetadata "meta-data/block-device-mapping/
ephemeral$((Get-WmiObject -Class Win32_Diskdrive | Where-Object {$_.DeviceID -eq ("\\.\\"$PhysicalDrive"+$DiskDrive.Number) }).SCSIPort - 2)"
    $BlockDevice = $null
    $VirtualDevice = If ($VirtualDeviceMap.ContainsKey($BlockDeviceName))
    { $VirtualDeviceMap[$BlockDeviceName] } Else { $null }
}
ElseIf ($DiskDrive.path -like "*PROD_AMAZON*") {
    $BlockDevice = ""
    $BlockDeviceName = ($BlockDeviceMappings | Where-Object {$_.ebs.VolumeId -eq $EbsVolumeID}).DeviceName
    $VirtualDevice = $null
}
Else {
    $BlockDeviceName = $null
    $BlockDevice = $null
    $VirtualDevice = $null
}
New-Object PSObject -Property @{
    Disk          = $Disk;
    Partitions    = $Partitions;
    DriveLetter   = If ($DriveLetter -eq $null) { "N/A" } Else { $DriveLetter };
    EbsVolumeID   = If ($EbsVolumeID -eq $null) { "N/A" } Else { $EbsVolumeID };
    Device        = If ($BlockDeviceName -eq $null) { "N/A" } Else { $BlockDeviceName };
    VirtualDevice = If ($VirtualDevice -eq $null) { "N/A" } Else { $VirtualDevice };
    VolumeName    = If ($VolumeName -eq $null) { "N/A" } Else { $VolumeName };
}
} | Sort-Object Disk | Format-Table -AutoSize -Property Disk, Partitions, DriveLetter,
EbsVolumeID, Device, VirtualDevice, VolumeName
```

Note

이 스크립트에는 PS용 AWS 도구에서 구성된 프로파일 또는 인스턴스에 연결된 IAM 역할이 필요 합니다.

이 스크립트를 실행하기 전 다음 명령을 실행하여 PowerShell 스크립트를 실행합니다.

```
Set-ExecutionPolicy RemoteSigned
```

스크립트를 복사한 다음 Windows 인스턴스에 .ps1 파일로 저장합니다. 액세스 키를 설정하지 않고 이 스크립트를 실행하면 다음과 유사한 출력이 표시됩니다.

Disk	Partitions	DriveLetter	EbsVolumeID	Device	VirtualDevice	VolumeName
0	0	N/A	N/A	xvdca	ephemeral0	N/A
1	0	N/A	N/A	xvdcb	ephemeral1	N/A
2	1	C:	vol-0064aexamplec838a	/dev/sdal	root	Windows
3	0	N/A	vol-02256example8a4a3	xvdf	ebs2	N/A

인스턴스 시작 시 Amazon EC2에 대한 액세스를 허용하는 정책을 통해 IAM 역할을 지정하거나 Windows PowerShell용 AWS 도구 사용 설명서의 [AWS 자격 증명 사용](#)에 설명된 대로 Windows 인스턴스에 자격 증명을 설정한 경우, NA 대신 EBS 볼륨의 볼륨 ID가 VolumeId 열에 표시됩니다.

디스크 디바이스와 디스크 이름 간 매핑

볼륨을 마운트할 때 인스턴스용 블록 디바이스 드라이버가 실제 볼륨 이름을 할당합니다.

매핑

- [인스턴스 스토어 볼륨 \(p. 1121\)](#)
- [EBS 볼륨 \(p. 1121\)](#)
- [NVMe EBS 볼륨 \(p. 1122\)](#)

인스턴스 스토어 볼륨

다음 표는 Citrix PV 및 AWS PV 드라이버가 NVMe가 아닌 인스턴스 스토어 볼륨을 Windows 볼륨에 매핑하는 방법을 설명합니다. 사용할 수 있는 인스턴스 스토어 볼륨의 개수는 인스턴스 유형에 의해 결정됩니다. 자세한 내용은 [인스턴스 스토어 볼륨 \(p. 1093\)](#) 단원을 참조하십시오.

위치	디바이스 이름
버스 번호 0, 타겟 ID 78, LUN 0	xvdca
버스 번호 0, 타겟 ID 79, LUN 0	xvdcb
버스 번호 0, 타겟 ID 80, LUN 0	xvdcc
버스 번호 0, 타겟 ID 81, LUN 0	xvdcd
버스 번호 0, 타겟 ID 82, LUN 0	xvdce
버스 번호 0, 타겟 ID 83, LUN 0	xvdcf
버스 번호 0, 타겟 ID 84, LUN 0	xvdcg
버스 번호 0, 타겟 ID 85, LUN 0	xvdch
버스 번호 0, 타겟 ID 86, LUN 0	xvdci
버스 번호 0, 타겟 ID 87, LUN 0	xvdcj
버스 번호 0, 타겟 ID 88, LUN 0	xvdck
버스 번호 0, 타겟 ID 89, LUN 0	xvdcl

EBS 볼륨

다음 표는 Citrix PV 및 AWS PV 드라이버가 인스턴스 스토어 볼륨을 NVME가 아닌 EBS 볼륨에 매핑하는 방법을 설명합니다.

위치	디바이스 이름
버스 번호 0, 타겟 ID 0, LUN 0	/dev/sda1
버스 번호 0, 타겟 ID 1, LUN 0	xvdb
버스 번호 0, 타겟 ID 2, LUN 0	xvdc
버스 번호 0, 타겟 ID 3, LUN 0	xvdd

위치	디바이스 이름
버스 번호 0, 타겟 ID 4, LUN 0	xvde
버스 번호 0, 타겟 ID 5, LUN 0	xvdf
버스 번호 0, 타겟 ID 6, LUN 0	xvdg
버스 번호 0, 타겟 ID 7, LUN 0	xvdh
버스 번호 0, 타겟 ID 8, LUN 0	xvdi
버스 번호 0, 타겟 ID 9, LUN 0	xvdj
버스 번호 0, 타겟 ID 10, LUN 0	xvdk
버스 번호 0, 타겟 ID 11, LUN 0	xndl
버스 번호 0, 타겟 ID 12, LUN 0	xvdm
버스 번호 0, 타겟 ID 13, LUN 0	xvdn
버스 번호 0, 타겟 ID 14, LUN 0	xvdo
버스 번호 0, 타겟 ID 15, LUN 0	xvdp
버스 번호 0, 타겟 ID 16, LUN 0	xvdq
버스 번호 0, 타겟 ID 17, LUN 0	xvdr
버스 번호 0, 타겟 ID 18, LUN 0	xvds
버스 번호 0, 타겟 ID 19, LUN 0	xvdt
버스 번호 0, 타겟 ID 20, LUN 0	xvdu
버스 번호 0, 타겟 ID 21, LUN 0	xvdv
버스 번호 0, 타겟 ID 22, LUN 0	xvdw
버스 번호 0, 타겟 ID 23, LUN 0	xvdx
버스 번호 0, 타겟 ID 24, LUN 0	xvdy
버스 번호 0, 타겟 ID 25, LUN 0	xvdz

NVMe EBS 볼륨

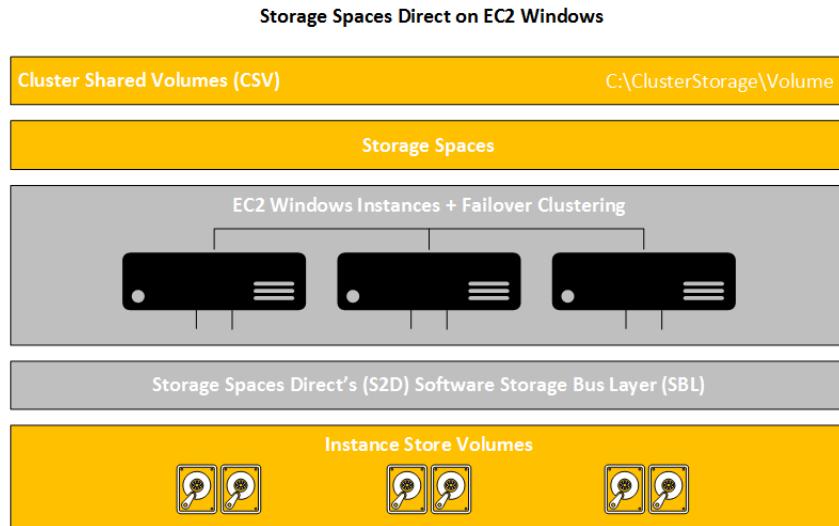
[Nitro 시스템 \(p. 118\)](#) 기반 인스턴스에서는 EBS 볼륨이 NVMe 디바이스로 표시됩니다. [Get-Disk 명령](#)을 써서 Windows 디스크 번호를 EBS 볼륨 ID로 매핑할 수 있습니다. 자세한 내용은 [EBS 디바이스 식별 \(p. 1048\)](#) 단원을 참조하십시오.

자습서: Amazon EC2에 저장소 공간 다이렉트(S2D) 배포

저장소 공간 다이렉트(S2D)는 사용자가 Windows Server 2016의 기능을 사용해 로컬 스토리지를 클러스터링할 수 있도록 하는 확장성이 매우 뛰어난 소프트웨어 정의 스토리지 아키텍처입니다. S2D는 기존의 SAN

또는 NAS 어레이를 대체합니다. 또한 내장 Windows 기능과 도구를 사용해 클러스터 내 여러 노드에 걸쳐 있는 고가용성 스토리지를 구성합니다. 자세한 내용은 Microsoft 문서의 [저장소 공간 다이렉트](#)를 참조하십시오.

다음 다이어그램은 Amazon EC2 Windows 의 S2D 아키텍처를 나타냅니다.



기술 수준

Windows Server 컴퓨팅과 VPC에서 도메인 병합 Amazon EC2 Windows 인스턴스의 생성 및 관리 방법에 대한 이해가 필요합니다. Windows PowerShell용 AWS 도구 및 Windows Failover Clustering에 대한 지식이 있으면 유용하긴 하지만 반드시 필요하지는 않습니다.

이 자습서에서 배울 내용

- 저장소 공간 다이렉트(S2D)를 사용하여 고가용성 스토리지 클러스터 프로비저닝
- 클러스터에 대해 내결함성 클러스터 공유 볼륨(CSV) 프로비저닝

시작하기 전에

- 아직 수행하지 않은 경우 <https://aws.amazon.com/>을 열고 AWS 계정을 생성합니다.
- 퍼블릭 서브넷 하나와 인스턴스의 프라이빗 서브넷 2개를 사용해 가상 사설 클라우드(VPC)를 생성합니다. 세 번째 프라이빗 서브넷은 AWS Directory Service를 위해 구성해야 합니다.
- Windows Server 2016에 맞는 최신 Amazon Machine Image(AMI) 중 하나를 선택합니다. 이 AMI를 있는 그대로 사용하거나 자체 사용자 지정 AMI에 대한 기준으로 사용할 수 있습니다. AWS에서는 최신 퍼블릭 EC2 Windows Server 2016 AMI를 사용할 것을 권장합니다.
- AWS Directory Service 디렉터리를 생성합니다. 이는 더 이상 Windows Server 2016에서 장애 조치 클러스터링 기능을 사용하기 위한 요구 사항이 아닙니다. 그러나 이 자습서에서는 사용자의 인스턴스가 EC2 또는 AWS Managed Active Directory에서 Active Directory 도메인에 병합된다고 가정합니다. 자세한 내용은 AWS Directory Service Administration Guide의 [AWS Directory Service 시작하기](#)를 참조하십시오.
- 컴퓨터에 Windows PowerShell용 AWS 도구를 설치 및 구성합니다. 자세한 내용은 [Windows PowerShell 용 AWS 도구 사용 설명서](#) 단원을 참조하십시오.

중요 고려 사항

- 데이터를 백업 또는 복제하지 않은 경우 [인스턴스 스토어 볼륨 \(p. 1091\)](#)이 있는 인스턴스를 종료하면 데이터가 손실될 수 있습니다. 인스턴스 스토리지의 데이터는 관련 인스턴스의 수명 기간 동안만 지속됩니다.

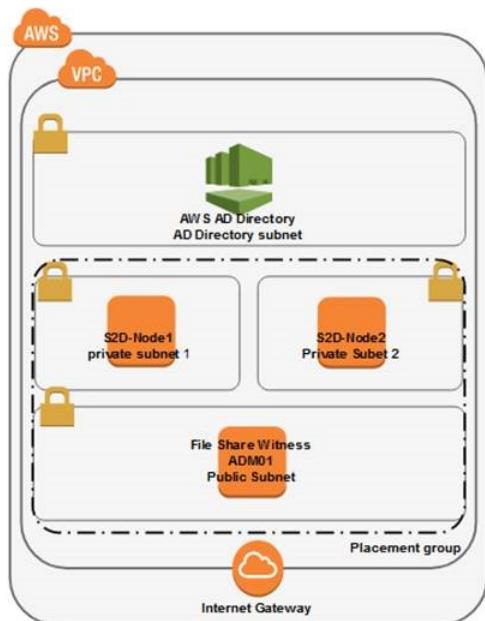
다. 인스턴스가 재부팅(의도적 또는 의도적이지 않게)되면 인스턴스 스토어의 데이터는 유지됩니다. 그러나 다음 상황에서는 인스턴스 스토어의 데이터가 손실됩니다.

- 기본 디스크 드라이브 오류
 - 인스턴스가 중지됩니다.
 - 인스턴스가 종료됩니다.
-
- 데이터를 백업 또는 복제하지 않은 경우 클러스터의 인스턴스를 너무 많이 종지하면 데이터가 손실될 수 있습니다. AWS에서 클러스터를 사용하는 것처럼 S2D를 사용하는 경우 내결함성보다 더 많은 노드를 손실하게 되면 데이터가 손실될 수 있습니다. 클러스터에서 발생할 수 있는 가장 큰 위험은 모든 노드 손실입니다. 클러스터 중복성은 단일 인스턴스에서(또는 내결함성이 클러스터 중복성을 지원하는 경우) 실패가 발생하지 않도록 방지합니다. 하지만 클러스터 내에서 실패한 디스크 드라이브가 있는 인스턴스의 수가 내결함성을 초과하면 데이터가 손실될 수 있습니다. 또한 중지 또는 종료된 인스턴스의 수가 내결함성을 초과하는 경우에도 데이터가 손실될 수 있습니다. 위험을 줄이기 위해 클러스터에서 인스턴스를 중지 또는 종료할 수 있는 사람 또는 시스템의 수를 제한하십시오. 클러스터 노드 인스턴스의 종료 위험을 줄이려면 이러한 인스턴스에서 [종료 방지 기능을 활성화](#) (p. 456)하십시오. 또한 사용자가 AWS 콘솔에서 노드를 중지하지 못하고 다시 시작만 수행하도록 [IAM 정책](#)을 구성할 수도 있습니다.
 - S2D에서는 전체 클러스터에 영향을 주는 네트워킹 또는 데이터 센터 장애를 방지하지 않습니다. 장애가 발생할 위험을 줄이려면 인스턴스를 동일한 랙에 배치하지 않도록 전용 호스트 사용을 고려하십시오.

작업

- [1단계: 시작 및 도메인 병합 인스턴스](#) (p. 1125)
- [2단계: 인스턴스 사전 조건 설치 및 구성](#) (p. 1127)
- [3단계: 장애 조치 클러스터 생성](#) (p. 1128)
- [4단계: S2D 활성화](#) (p. 1129)
- [5단계: 스토리지 프로비저닝](#) (p. 1129)
- [6단계: S2D 리소스 검토](#) (p. 1130)
- [7단계: 정리](#) (p. 1131)
- [추가 리소스](#) (p. 1131)

다음 다이어그램은 AWS에서 기존 버스천 머신에서 호스트되는 파일 공유 감시 기능을 사용하는 2노드 EC2 Windows S2D 클러스터의 아키텍처를 보여줍니다.



1단계: 시작 및 도메인 병합 인스턴스

모든 Nitro 인스턴스는 EBS 및/또는 NVMe를 사용하여 저장소 공간 다이렉트를 지원합니다. 현재 세대의 모든 Xen 기반 인스턴스는 AWS PV 드라이버 8.2.3 이상의 설치를 통해 저장소 공간 다이렉트를 지원합니다. I3 인스턴스는 로컬 인스턴스 스토어에 NVMe와 고성능 네트워크를 제공하기 때문에 이러한 인스턴스를 사용해 최적의 스토리지 성능을 얻을 수 있습니다. Amazon EC2에서 S2D를 구성하려면 최소 2개보다 많지만 16개를 넘지 않는 인스턴스로 구성된 클러스터가 필요합니다. 이러한 각 인스턴스는 노드 간 고성능 네트워크 연결을 제공하는 NVMe 디바이스가 두 개 이상 있어야 하고 Windows Server 2016을 실행해야 합니다. 자세한 내용은 Microsoft 문서([저장소 공간 다이렉트 하드웨어 요구 사항](#))를 참조하십시오.

S2D 하드웨어 요구 사항을 충족하고 가장 크고 빠른 가용 인스턴스 스토어 디바이스를 포함하고 있기 때문에 I3 인스턴스 크기가 권장됩니다. 또한 향상된 네트워킹 기능이 포함되어 있어 인스턴스당 S2D의 가용 리소스를 극대화할 수 있습니다. NVMe 디스크가 2개 이상 있는 M5D 및 R5D 인스턴스 유형을 사용할 수 있습니다. 하지만 로컬 인스턴스 스토어 디스크는 저장소 공간 다이렉트 클러스터의 캐시 디스크로 사용되고 용량 스토리지를 제공하기 위해 각 인스턴스에 EBS 볼륨을 2개 이상 추가해야 합니다.

파일 공유 감시 등과 같은 감시 기능을 구성한 경우 클러스터에서 내결합성을 유지하는 동시에 단일 노드에 대한 유지 관리를 가능하게 하는 3방향 모니터링 S2D 내결합성을 활용하기 위해 인스턴스를 3개 시작하는 것이 좋습니다. 또한 비용 절감형 솔루션으로 인스턴스 2개를 사용하는 양방향 모니터링을 사용할 수도 있지만 감시 기능이 필요하고 클러스터 노드를 유지 관리하는 동안에는 고가용성이 유지되지 않습니다.

관리 워크스테이션의 역할을 하는 기존의 배스천 머신에서 호스팅되는 파일 공유 감시를 사용해 2노드 클러스터 아키텍처를 배포합니다. 각 클러스터 노드를 다른 서브넷에 배포해야 합니다. Microsoft에서는 현재 저 저장소 공간 다이렉트에서 확장 클러스터를 지원하지 않기 때문에 이 아키텍처는 단일 가용 영역에 배포되지 않습니다. 그러나 가용 영역을 위한 당사의 지연 시간이 매우 낮은 고대역 설계를 적용한 결과, 단일 가용 영역과 다중 가용 영역 간의 성능이 거의 같습니다.

클러스터의 인스턴스를 시작하려면

1. Amazon EC2 콘솔 또는 [New-EC2Instance cmdlet](#)을 사용하여 i3.8xlarge 인스턴스 2개를 시작해 클러스터를 생성하고 t2.medium 인스턴스는 파일 공유 감시 기능을 호스팅할 관리 워크스테이션으로 시작합니다. 각 인스턴스에 다른 서브넷을 사용합니다. IP 할당을 위한 로직을 따르려는 경우 생성 시 기본

프라이빗 IP 주소를 정의합니다. 이 경우 나중에 클러스터 VIP에 보조 IP가 할당되기 때문에 각 클러스터 노드에 대해 보조 프라이빗 IP 주소를 정의해야 합니다.

PowerShell을 사용하여 각 인스턴스를 생성하려면 [New-EC2Instance](#) 명령을 사용합니다.

```
New-EC2Instance -ImageId ami-c49c0dac -MinCount 1 -MaxCount 1 -KeyName myPSKeyPair -  
SecurityGroupId mySGID -InstanceType i3.8xlarge -SubnetId mysubnetID
```

PowerShell을 사용하여 AWS AD 디렉터리를 생성하려면 [New-DSMicrosoftAD](#) 명령을 사용합니다. 또는 [AWS에서 AWS Managed Microsoft AD 디렉터리 생성](#)을 참조하십시오.

```
New-DSMicrosoftAD -Name corp.example.com -ShortName corp -Password P@ssw0rd -  
Description "AWS DS Managed" - VpcSettings_VpcId vpc-xxxxxxx -VpcSettings_SubnetId  
subnet-xxxxxxx, subnet-xxxxxxx
```

다음 S2D-node1 네트워크 인터페이스 구성은 사용합니다.

▼ Network interfaces ⓘ

Device	Network Interface	Subnet	Primary IP	Secondary IP addresses	IPv6 IPs
eth0	New network interface ▾	subnet-9850a3fe	172.16.1.199	172.16.1.200 172.16.1.201	Add IP

Note

이 클러스터에 배포된 각 역할(예: SQL Failover Cluster 인스턴스 또는 파일 서버)에는 각 노드에 추가 보조 IP 주소가 필요합니다. 예외는 액세스 지점이 필요 없는 스케일 아웃 파일 서버 역할입니다.

당사는 다음 구성은 사용합니다.

서버 NetBIOS 이름	IP 주소	서브넷
S2D-Node1	172.16.1.199(기본) 172.16.1.200(클러스터 VIP에 사용될 보조 IP) 172.16.1.201(나중에 SQL FCI 등과 같은 역할에 사용될 보조 IP)	AZ1(예: eu-west-1a) – 프라이빗 서브넷 1
S2D-Node2	172.16.3.199(기본) 172.16.3.200(클러스터 VIP에 사용될 보조 IP) 172.16.3.201(나중에 SQL FCI 등과 같은 역할에 사용될 보조 IP)	AZ1(예: eu-west-1a) – 프라이빗 서브넷 2

서버 NetBIOS 이름	IP 주소	서브넷
ADM01	지정되지 않음	AZ1(예: eu-west-1a) - 퍼블릭 서브넷

2. 생성 시 매끄러운 도메인 조인을 사용하여 인스턴스를 도메인에 결합할 수 있습니다. 인스턴스가 시작된 후 도메인에 인스턴스를 결합하려는 경우 [Add-Computer](#) 명령을 사용합니다. AWS 시스템 관리자 및 [AWS Directory Service](#)를 사용하여 [도메인에 EC2 인스턴스를 원활하게 조인](#)하는 것이 좋습니다.

이 자습서의 나머지 부분에 나오는 단계는 각 인스턴스에 대한 로컬 관리 권한을 가진 도메인 계정으로 실행해야 합니다. 구성으로 이동하기 전에 인스턴스 이름을 원하는 대로 바꿉니다. 이러한 노드에서 원격 PowerShell 연결 및 클러스터 통신이 가능하도록 보안 그룹 및 Windows 방화벽이 적절하게 구성되어 있는지 확인합니다.

2단계: 인스턴스 사전 조건 설치 및 구성

S2D를 사용하려면 파일 서비스 및 장애 조치 클러스터 Window 기능이 필요하고 10Gbps 네트워크 인터페이스가 한 개 이상 필요합니다. RSS 클라이언트 연결 수가 향상된 네트워크 어댑터의 RSS 대기열 수와 일치하는 [SMB 다중 채널](#)을 사용하도록 SMB를 구성하는 것이 좋습니다.

다음 단계는 배스천 인스턴스 ADM01에서 수행합니다.

필요한 Windows 기능을 설치하려면

- 클러스터 노드에서 관리 도구를 사용해 파일 서비스 및 장애 조치-클러스터링 Windows 기능을 설치합니다. ADM01에는 장애 조치 관리 도구만 설치합니다.

Note

두 인스턴스에 대해 설정한 컴퓨터 이름을 반영해 "S2D-Node1" 및 "S2D-Node2"를 변경합니다. 그렇지 않으면 값이 바뀌지 않습니다.

```
$nodes = "S2D-Node1", "S2D-Node2"
foreach ($node in $nodes) {
    Install-WindowsFeature -ComputerName $node -Name File-Services, Failover-Clustering
    -IncludeManagementTools
}
Install-WindowsFeature -Name RSAT-Clustering
```

네트워킹을 구성하려면

- 다중 채널을 활성화하고 RSS 연결 수를 설정합니다.

```
foreach ($node in $nodes) {
    Invoke-Command -ComputerName $node -ScriptBlock {
        [int]$RssQCount = (Get-NetAdapterAdvancedProperty | Where DisplayName -
        like "Maximum Number of RSS Queues").RegistryValue | Select -First 1
        $Params = @{
            EnableMultiChannel          = $true;
            ConnectionCountPerRssNetworkInterface = $RssQCount;
            Confirm                      = $false;
        }
        Set-SmbClientConfiguration @Params
    }
}
```

- RSS를 구성합니다.

```
foreach ($node in $nodes) {
    Invoke-Command -ComputerName $node -ScriptBlock {
        Get-WmiObject -class Win32_processor | ft systemname, Name, DeviceID,
        NumberOfCores, NumberOfLogicalProcessors
        $maxvcpu = (Get-WmiObject -class Win32_processor).NumberOfLogicalProcessors
        Get-NetAdapter | Set-NetAdapterRss -BaseProcessorNumber 2 -MaxProcessors
    $maxvcpu
    }
}
```

Note

RSS 구성을 설정하면 네트워크 어댑터가 다시 시작되므로 이 명령을 실행할 때 연결 해제 메시지가 표시됩니다.

수신측 크기 조정(RSS)은 Windows에서의 네트워킹에 매우 중요한 기술입니다. RSS는 수신되는 네트워크 트래픽이 처리를 위해 서버의 사용 프로세스 간에 분산되도록 합니다. RSS를 사용하지 않으면 네트워크 처리가 하나의 프로세스로 바인딩되어 약 4GBps로 제한됩니다. 현재, 기본적으로 모든 NIC는 RSS를 활성화하지만 구성이 최적화되어 있는지 않습니다. 모든 NIC는 기본적으로 "기본 프로세서" 0으로 구성되어 있습니다. 이는 다른 NIC와 함께 프로세서 0에서 처리를 시작함을 의미합니다. RSS를 최적의 상태로 구성하려면 프로세서 1을 시작합니다. 따라서 프로세서 0으로 기본적으로 랜딩되는 프로세스를 방해하지 않습니다.

3. 스토리지 공간 I/O 제한 시간 값을 30초로 늘립니다(게스트 클러스터로 구성하는 경우 권장).

```
foreach ($node in $nodes) {
    Invoke-Command -ComputerName $node -ScriptBlock {
        Set-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\spaceport
        \Parameters -Name HwTimeout -Value 0x00007530 -Verbose
    }
}
```

4. 모든 노드를 재부팅하여 변경 사항을 모두 적용합니다.

```
Restart-Computer -ComputerName $nodes -Wait -For Wmi -Force
```

3단계: 장애 조치 클러스터 생성

S2D는 기존 장애 조치 클러스터에서 활성화되는 기능입니다. 장애 조치 클러스터에서 S2D를 활성화하면 클러스터에 있는 각 노드의 로컬 스토리지를 제어합니다. 따라서 생성 시 스토리지 없이 클러스터를 설치한 다음 S2D를 활성화하는 것이 좋습니다.

AWS에서 클러스터를 생성하는 경우 노드가 배치된 각 서브넷에서 정적 IP 주소를 할당해야 합니다. 할당된 정적 IP 주소는 콘솔에서 각 노드에 대한 보조 프라이빗 IP 주소로 설정해야 합니다. 이 자습서에서는 각 노드 배포 시 172.16.1.200 및 172.16.3.200을 구성했습니다.

기본 제공 [Test-Cluster](#) 명령을 사용하여 클러스터 구성 확인하고 검토할 수 있습니다.

클러스터 구성 테스트 및 확인

1. Storage Spaces Direct, Inventory, Network 및 System Configuration 테스트와 함께 [Test-Cluster](#) 명령을 실행합니다.

```
$report = Test-Cluster -Node $nodes -Include 'Storage Spaces
Direct', 'Inventory', 'Network', 'System Configuration'
```

2. 테스트 결과를 검토합니다.

```
$reportFilePath = $report.FullName  
Start-Process $reportFilePath
```

3. [New-Cluster](#)를 사용하여 클러스터를 생성합니다. 가상 IP를 각 노드에 AWS 콘솔의 프라이빗 IP 주소로 할당해야 합니다.

```
$vips = "172.16.1.200", "172.16.3.200"  
New-Cluster -Name S2D -Node $nodes -StaticAddress $vips -NoStorage
```

4. 파일 공유 감시를 구성합니다.

```
New-Item -ItemType Directory -Path c:\Share\Witness  
[string]$DomainName = (Get-WmiObject win32_computersystem).domain  
New-SmbShare -Name fsw -Path c:\Share\Witness -FullAccess ($DomainName + "\Domain  
Computers")  
Set-ClusterQuorum -Cluster S2D -FileShareWitness \\$env:COMPUTERNAME\fsw
```

4단계: S2D 활성화

클러스터가 준비가 되면 다음과 같이 [Enable-ClusterS2D](#)를 사용하여 노드 중 하나에서 S2D를 활성화합니다. 현재 설정(로컬 NVMe)에 한 가지 유형의 디스크뿐이 없기 때문에 캐시 디스크 등과 같은 디스크는 사용하지 않습니다.

1. [Enable-ClusterS2D](#) 명령을 사용해 i3 인스턴스 유형에 대해 S2D를 활성화합니다.

```
Enable-ClusterS2D -PoolFriendlyName S2DPool -Confirm:$false -SkipEligibilityChecks:  
$true -CimSession $nodes[0]
```

2. NVMe 및 EBS와 함께 m5d 또는 r5d 인스턴스 유형을 사용하는 경우 NVMe 디스크를 캐시 디스크로 사용합니다. 명령은 다음과 같습니다.

```
Enable-ClusterS2D -PoolFriendlyName S2DPool -CacheDeviceModel "Amazon EC2 NVMe" -  
Confirm:$false -SkipEligibilityChecks:$true -CimSession $nodes[0]
```

5단계: 스토리지 프로비저닝

스토리지를 프로비저닝하려면 스토리지 풀을 생성한 다음 해당 풀에서 볼륨을 생성합니다. 간단하게 수행할 수 있도록 기본적으로 [Enable-ClusterS2D](#) 명령은 클러스터의 모든 가용 디스크를 사용해 풀을 생성합니다. 이 명령을 사용해 스토리지 풀 이름을 "S2D Pool"로 구성했습니다.

볼륨이 생성되면 클러스터의 모든 노드에서 이러한 볼륨에 액세스할 수 있습니다. 그런 다음 이러한 볼륨은 파일 서버 역할 등과 같은 클러스터의 특정 역할에 할당하거나 [클러스터 공유 볼륨](#)(CSV)으로 할당할 수 있습니다. CSV는 전체 클러스터에 액세스할 수 있습니다. 즉, 이 클러스터의 모든 노드를 이 볼륨에 쓰고 읽을 수 있다는 의미입니다.

성능 개선을 위해 CSV에 수정된 프로비저닝 및 ReFS 파일 시스템을 사용하는 것이 좋습니다. 섹터 크기는 클러스터에 배포되는 워크로드의 유형에 따라 달라집니다. 섹터 크기에 대한 자세한 내용은 [Cluster Size Recommendations for ReFS and NTFS](#)를 참조하십시오. 로컬 읽기 성능 개선을 위해 애플리케이션 또는 워크로드를 호스팅하는 노드에 CSV를 맞추는 것이 좋습니다. 노드 간에 여러 CSV와 애플리케이션을 분산시킬 수 있습니다.

클러스터 공유 볼륨(CSV) 생성

- [New-Volume](#) 명령을 사용하여 새 1TB CSV를 생성합니다.

```
$Params = @{
    FriendlyName      = 'CSV1';
    FileSystem        = 'CSVFS_ReFS';
    StoragePoolFriendlyName = 'S2DPool';
    Size              = 1TB;
    AllocationUnitSize = 65536;
    ProvisioningType   = 'Fixed';
    CimSession         = $nodes[0];
}
New-Volume @Params
```

6단계: S2D 리소스 검토

구성한 S2D 리소스가 Failover Cluster Manager에 표시됩니다.

CSV를 보려면

1. 서버 관리자를 엽니다.
2. 도구, Failover Cluster Manager를 선택합니다.
3. 클러스터의 이름을 확장하고 스토리지를 확장한 다음 디스크를 선택합니다.

표시 이름, 용량, CSV를 호스팅하는 노드와 기타 데이터가 표시됩니다. CSV 관리에 대한 자세한 내용은 [Use Cluster Shared Volumes in a Failover Cluster](#)를 참조하십시오.

CSV에서 로드를 합성하려면

[Diskspd 유ти리티](#) 등과 같은 도구를 사용합니다. RDP를 사용해 클러스터 노드 중 하나에 연결하고 Diskspd 도구를 사용하여 다음을 실행합니다.

```
$mycsv = (gci C:\ClusterStorage\ | select -First 1).fullname
.\diskspd.exe -d60 -b4k -o1024 -t32 -L -Sh -r -w50 -W60 -c100G $mycsv\test.dat
```

클러스터의 S2D 스토리지 성능을 보려면

[Get-StorageHealthReport](#) 명령을 사용하여 클러스터 노드 중 하나에 대한 클러스터 성능을 확인합니다.

1. 새 PowerShell 창을 열고 합성된 워크로드를 시작합니다.
2. 원래 PowerShell 창에서 [Get-StorageSubSystem *cluster*](#) | [Get-StorageHealthReport](#)를 실행해 워크로드 실행 중 스토리지 하위 시스템의 성능 결과를 확인합니다.

```
PS C:\> Get-StorageSubSystem *cluster* | Get-StorageHealthReport

CPUUsageAverage          : 60.44 %
CapacityPhysicalPooledAvailable : 9.82 GB
CapacityPhysicalPooledTotal   : 13.82 TB
CapacityPhysicalTotal       : 13.82 TB
CapacityPhysicalUnpooled     : 0 B
CapacityVolumesAvailable    : 1.89 TB
CapacityVolumesTotal        : 2 TB
```

IOLatencyAverage	:	257.56 ms
IOLatencyRead	:	255.87 ms
IOLatencyWrite	:	259.25 ms
IOOPSRead	:	64327.37 /S
IOPSTotal	:	128582.85 /S
IOPSWrite	:	64255.49 /S
IOThroughputRead	:	251.28 MB/S
IOThroughputTotal	:	502.28 MB/S
IOThroughputWrite	:	251 MB/S
MemoryAvailable	:	477.77 GB
MemoryTotal	:	488 GB

7단계: 정리

이 자습서에 따라 EC2 Windows에서 S2D를 사용해 고가용성 스토리지 클러스터를 생성했다면 클러스터에 대해 파일 공유 감시 역할도 하는 배스천 서버에서 두 인스턴스의 저장소 공간 다이렉트 클러스터를 생성한 것입니다. 인스턴스를 실행한 개별 시간 또는 부분 시간에 대한 요금이 청구됩니다. 클러스터가 필요하지 않은 경우 EC2 콘솔 또는 [Windows용 AWS 도구](#)를 사용하여 이 프로젝트를 위해 생성한 리소스를 삭제하십시오. 이렇게 하려면 장애 조치 클러스터 관리 mmc에서 클러스터를 삭제하고, 인스턴스를 종료하고, Active Directory에서 클러스터 및 클러스터의 각 노드에 대한 컴퓨터 객체를 삭제합니다.

추가 리소스

[저장소 공간 다이렉트 계산기\(프리뷰\)](#)

[저장소 공간 다이렉트 계획](#)

[저장소 공간 다이렉트 개요](#)

[저장소 공간 다이렉트의 내결합성 및 저장소 효율성](#)

리소스 및 태그

Amazon EC2는 사용자가 생성하여 사용할 수 있는 서로 다른 리소스를 제공합니다. 이러한 리소스에는 이미지, 인스턴스, 볼륨 및 스냅샷 등이 있습니다. 리소스를 생성하면 리소스에 고유 리소스 ID가 할당됩니다.

일부 리소스에는 사용자가 정의하는 값으로 태그를 붙일 수 있어 쉽게 정리하고 식별할 수 있습니다.

다음 주제에서는 리소스와 태그에 대한 설명과 이를 이용한 작업 방법에 대해 살펴보겠습니다.

내용

- [리소스 위치 \(p. 1132\)](#)
- [리소스 ID \(p. 1133\)](#)
- [리소스 목록화 및 필터링 \(p. 1138\)](#)
- [Amazon EC2 리소스에 태그 지정 \(p. 1143\)](#)
- [Amazon EC2 서비스 할당량 \(p. 1154\)](#)
- [Amazon EC2 사용 보고서 \(p. 1156\)](#)

리소스 위치

일부 리소스는 모든 리전에서 사용할 수 있고(글로벌) 일부 리소스는 상주하는 리전 또는 가용 영역에만 해당합니다.

리소스	유형	설명
AWS 계정	전 세계	모든 리전에 동일한 AWS 계정을 사용할 수 있습니다.
키 페어	글로벌 또는 리전	Amazon EC2를 사용하여 생성하는 키 페어는 이를 생성한 리전과 연동됩니다. 자체 RSA 키 페어를 생성하여 사용할 리전에 업로드할 수 있습니다. 따라서 각 리전에 업로드하여 키를 전역적으로 사용 가능하게 만들 수 있습니다. 자세한 내용은 Amazon EC2 키 페어 및 Windows 인스턴스 (p. 900) 단원을 참조하십시오.
Amazon EC2 리소스 식별자	리전	AMI ID, 인스턴스 ID, EBS 볼륨 ID, EBS 스냅샷 ID 등 각 리소스 식별자는 해당 리전에 둑여 있으며, 리소스를 생성한 리전에서만 사용할 수 있습니다.
사용자가 공급한 리소스 이름	리전	보안 그룹 이름, 키 페어 이름 등 각 리소스 이름은 해당 리전에 둑여 있으며, 리소스를 생성한 리전에서만 사용할 수 있습니다. 여러 리전에서 동일한 이름을 가진 리소스를 생성할 수는 있지만, 이 경우에도 각 리소스들이 서로 관계를 가지게 되는 것은 아닙니다.
AMI	리전	AMI는 Amazon S3 내에서 파일이 위치하고 있는 리전에 둑여 있습니다. 한 리전의 AMI를 다른 리전으로 복사할

리소스	유형	설명
		수 있습니다. 자세한 내용은 AMI 복사 (p. 103) 단원을 참조하십시오.
탄력적 IP 주소	리전	탄력적 IP 주소는 리전에 둑여 있으며 동일한 리전의 인스턴스에만 연결할 수 있습니다.
보안 그룹	리전	보안 그룹은 리전에 둑여 있으며 동일한 리전의 인스턴스에만 배정할 수 있습니다. 보안 그룹 규칙을 사용해서 인스턴스가 그 리전 바깥의 인스턴스와 통신하게 할 수는 없습니다. 다른 리전의 인스턴스에서 나오는 트래픽은 WAN 대역폭으로 간주됩니다.
EBS 스냅샷	리전	EBS 스냅샷은 리전에 둑여 있으며 동일한 리전에서 볼륨을 생성하는 데만 사용할 수 있습니다. 한 리전의 스냅샷을 다른 리전으로 복사할 수 있습니다. 자세한 내용은 Amazon EBS 스냅샷 복사 (p. 985) 단원을 참조하십시오.
EBS 볼륨	가용 영역	Amazon EBS 볼륨은 가용 영역에 둑여 있으며 동일한 가용 영역의 인스턴스에만 연결될 수 있습니다.
인스턴스	가용 영역	인스턴스는 이를 실행한 가용 영역에 둑여 있습니다. 그러나 인스턴스 ID는 그 리전에 둑여 있습니다.

리소스 ID

리소스가 생성되면 각 리소스마다 고유 리소스 ID가 할당됩니다. 리소스 ID를 사용하여 Amazon EC2 콘솔에서 리소스를 확인할 수 있습니다. 명령줄 도구 또는 Amazon EC2 API를 사용하여 Amazon EC2로 작업할 경우 특정 명령의 리소스 ID가 필요합니다. 예를 들어, [stop-instances](#) AWS CLI 명령을 사용하여 인스턴스를 중지할 경우 명령에 인스턴스 ID를 지정해야 합니다.

리소스 ID 길이

리소스 ID는 리소스 식별자(예: 스냅샷은 snap) 다음에 하이픈과 고유한 문자와 숫자 조합이 오는 형식을 갖습니다. 2016년 1월부터 Amazon EC2 및 Amazon EBS 리소스 유형에 더 긴 ID를 도입하고 있습니다. 영문자 문자 조합 길이는 8자 형식이었는데, 새 ID는 17자 형식입니다(예: 인스턴스 ID i-1234567890abcdef0).

지원되는 리소스 유형에는 옵트인 기간이 있고, 그 기간 동안 리소스 ID 형식 및 기한을 선택할 수 있습니다. 기한 이후에는 리소스가 기본적으로 더 긴 ID 형식을 갖습니다. 특정 리소스 유형에 대한 기한이 지난 이후에는 해당 리소스 유형에 대해 더 긴 ID 형식을 비활성화할 수 없습니다.

리소스 유형마다 옵트인 기간과 기한이 다릅니다. 다음 표에는 지원되는 리소스 유형이 옵트인 기간 및 기한과 함께 나와 있습니다.

Resource type	옵트인 기간	기한
instance snapshot reservation volume	더 이상 사용할 수 없음	2016년 12월 15일
bundle conversion-task customer-gateway dhcp-options elastic-ip-allocation	2018년 2월 9일 ~2018년 6월 30일	2018년 6월 30일
elastic-ip-association export-task flow-log image import-task		

Resource type	옵트인 기간	기한
internet-gateway network-acl network-acl-association network-interface network-interface-attachment prefix-list route-table route-table-association security-group subnet subnet-cidr-block-association vpc vpc-cidr-block-association vpc-endpoint vpc-peering-connection vpn-connection vpn-gateway		

옵트인 기간 사이

옵트인 기간 사이에는 리소스에 대한 더 긴 ID를 활성화 또는 비활성화할 수 있습니다. 한 리소스 유형에 더 긴 ID를 활성화한 후에는 생성하는 모든 새 리소스에 더 긴 ID가 생성됩니다.

Note

리소스 ID는 생성 이후 변경되지 않습니다. 따라서 옵트인 기간 사이에 더 긴 ID를 활성화 또는 비활성화하는 것은 기존 리소스 ID에 영향을 미치지 않습니다.

AWS 계정을 생성한 시점에 따라, 지원되는 리소스 유형에 기본적으로 더 긴 ID를 사용할 수 있습니다. 하지만 각 리소스 유형에 대해 설정된 기한까지는 더 긴 ID 사용을 옵트아웃할 수 있습니다. 자세한 내용은 [Amazon EC2 FAQ](#)의 더 긴 EC2 및 EBS 리소스 ID를 참조하십시오.

기한 이후

기한이 지난 이후에는 리소스 유형에 대해 더 긴 ID를 비활성화할 수 없습니다. 생성하는 모든 새로운 리소스는 더 긴 ID를 포함하여 생성됩니다.

더 긴 ID 작업

더 긴 ID의 설정 또는 설정 해제는 IAM 사용자 및 IAM 역할별로 가능합니다. 기본적으로 IAM 사용자 또는 역할은 루트 사용자와 동일한 설정으로 지정됩니다.

목차

- [더 긴 ID 설정 보기 \(p. 1134\)](#)
- [더 긴 ID 설정 수정 \(p. 1135\)](#)

더 긴 ID 설정 보기

콘솔 또는 명령줄 도구를 사용하여 긴 ID를 지원하는 리소스 유형을 볼 수 있습니다.

콘솔을 사용하여 더 긴 ID 설정을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 더 긴 ID 설정을 보고자 하는 리전을 선택합니다.
3. 대시보드의 계정 속성에서 리소스 ID 길이 관리를 선택하십시오.

- 고급 리소스 ID 관리를 확장하여 더 긴 ID를 지원하는 리소스 유형과 기한을 봅니다.

명령줄을 사용하여 더 긴 ID 설정을 보려면

다음 명령 중 하나를 사용합니다.

- [describe-id-format\(AWS CLI\)](#)

```
aws ec2 describe-id-format --region region
```

- [Get-EC2IdFormat\(Windows PowerShell용 AWS 도구\)](#)

```
Get-EC2IdFormat -Region region
```

명령줄을 사용하여 특정 IAM 사용자 또는 IAM 역할에 대한 더 긴 ID 설정을 보려면

다음 명령 중 하나를 사용하고 요청 내에서 IAM 사용자, IAM 역할 또는 루트 계정 사용자의 ARN을 지정합니다.

- [describe-identity-id-format\(AWS CLI\)](#)

```
aws ec2 describe-identity-id-format --principal-arn arn-of-iam-principal --region region
```

- [Get-EC2IdentityIdFormat\(Windows PowerShell용 AWS 도구\)](#)

```
Get-EC2IdentityIdFormat -PrincipalArn arn-of-iam-principal -Region region
```

명령줄을 사용하여 특정 리전에 대해 집계된 더 긴 ID 설정을 보려면

[describe-aggregate-id-format](#) AWS CLI 명령을 사용하여 전체 리전에 대해 집계된 더 긴 ID 설정은 물론 각 리소스 유형에 대한 모든 ARN의 집계된 더 긴 ID 설정을 봅니다. 이 명령은 빠른 감사를 수행하여 특정 리전이 더 긴 ID에 대해 완전히 옵트인되었는지 확인하는 데 유용합니다.

```
aws ec2 describe-aggregate-id-format --region region
```

명시적으로 사용자 지정 더 긴 ID 설정을 정의한 사용자를 식별하려면

[describe-principal-id-format](#) AWS CLI 명령을 사용하여 명시적으로 더 긴 ID 설정을 지정한 루트 사용자와 모든 IAM 역할 및 IAM 사용자에 대한 더 긴 ID 형식 설정을 봅니다. 이 명령은 기본 더 긴 ID 설정을 재정의한 IAM 사용자 및 IAM 역할을 식별하는 데 유용합니다.

```
aws ec2 describe-principal-id-format --region region
```

더 긴 ID 설정 수정

콘솔 또는 명령줄 도구를 사용하여 여전히 옵트인 기간 내에 있는 리소스 유형의 더 긴 ID 설정을 수정할 수 있습니다.

Note

이 단원의 AWS CLI 및 Windows PowerShell용 AWS 도구 명령은 각 리전별로만 사용됩니다. 별도로 지정되지 않는 한 기본 리전이 적용됩니다. 다른 리전의 설정을 수정하려면 명령에 `region` 파라미터를 포함하십시오.

콘솔을 사용하여 더 긴 ID 설정을 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 더 긴 ID 설정을 수정하고자 하는 리전을 선택합니다.
3. 대시보드의 계정 속성에서 리소스 ID 길이 관리를 선택하십시오.
4. 다음 중 하나를 수행하십시오.
 - 모든 리전 전체의 모든 IAM 사용자에 대해 모든 지원 리소스의 더 긴 ID를 활성화하려면 긴 ID로 전환, 예, 긴 ID로 전환을 선택합니다.

Important

IAM 사용자 및 IAM 역할에서 이 작업을 수행하려면 `ec2:ModifyIdentityIdFormat` 권한이 필요합니다.

- IAM 사용자 계정에 대한 특정 리소스 유형의 더 긴 ID 설정을 수정하려면 고급 리소스 ID 관리를 확장하고, 내 IAM 역할/사용자 열에 있는 해당 확인란을 선택하여 더 긴 ID를 활성화하거나, 확인란을 선택 해제하여 더 긴 ID를 비활성화합니다.
- 모든 IAM 사용자에 대한 특정 리소스 유형의 더 긴 ID 설정을 수정하려면 고급 리소스 ID 관리를 확장하고, 모든 IAM 역할/사용자 열에 있는 해당 확인란을 선택하여 더 긴 ID를 활성화하거나, 확인란을 선택 해제하여 더 긴 ID를 비활성화합니다.

명령줄을 사용하여 IAM 사용자 계정에 대한 더 긴 ID 설정을 수정하려면

다음 명령 중 하나를 사용합니다.

Note

루트 사용자로서 이런 명령을 사용 중인 경우에는 IAM 사용자 또는 역할이 스스로 이러한 설정을 명시적으로 재정의하지 않는 한 이러한 변경 사항이 전체 AWS 계정에 적용됩니다.

- [modify-id-format\(AWS CLI\)](#)

```
aws ec2 modify-id-format --resource resource_type --use-long-ids
```

명령을 사용하여 모든 지원되는 리소스에 대한 더 긴 ID 설정을 수정할 수 있습니다. 이를 수행하려면 `resource_type` 파라미터를 `all-current`로 대체합니다.

```
aws ec2 modify-id-format --resource all-current --use-long-ids
```

Note

더 긴 ID를 비활성화하려면 `use-long-ids` 파라미터를 `no-use-long-ids`로 대체합니다.

- [Edit-EC2IdFormat \(Windows PowerShell용 AWS 도구\)](#)

```
Edit-EC2IdFormat -Resource resource_type -UseLongId boolean
```

명령을 사용하여 모든 지원되는 리소스에 대한 더 긴 ID 설정을 수정할 수 있습니다. 이를 수행하려면 `resource_type` 파라미터를 `all-current`로 대체합니다.

```
Edit-EC2IdFormat -Resource all-current -UseLongId boolean
```

명령줄을 사용하여 특정 IAM 사용자 또는 IAM 역할에 대한 더 긴 ID 설정을 수정하려면

다음 명령 중 하나를 사용하고 요청 내에서 IAM 사용자, IAM 역할 또는 루트 사용자의 ARN을 지정합니다.

- modify-identity-id-format(AWS CLI)

```
aws ec2 modify-identity-id-format --principal-arn arn-of-iam-principal --  
resource resource_type --use-long-ids
```

명령을 사용하여 모든 지원되는 리소스에 대한 더 긴 ID 설정을 수정할 수 있습니다. 이를 수행하려면 all-current 파라미터에 대해 --resource를 지정합니다.

```
aws ec2 modify-identity-id-format --principal-arn arn-of-iam-principal --resource all-  
current --use-long-ids
```

Note

더 긴 ID를 비활성화하려면 use-long-ids 파라미터를 no-use-long-ids로 대체합니다.

- Edit-EC2IdentityIdFormat(Windows PowerShell용 AWS 도구)

```
Edit-EC2IdentityIdFormat -PrincipalArn arn-of-iam-principal -Resource resource_type -  
UseLongId boolean
```

명령을 사용하여 모든 지원되는 리소스에 대한 더 긴 ID 설정을 수정할 수 있습니다. 이를 수행하려면 all-current 파라미터에 대해 -Resource를 지정합니다.

```
Edit-EC2IdentityIdFormat -PrincipalArn arn-of-iam-principal -Resource all-current -  
UseLongId boolean
```

긴 ID 설정에 대한 액세스 제어

IAM 사용자와 역할은 관련된 IAM 정책을 통해 명시적으로 권한을 부여 받지 않는 한, 기본적으로 다음 작업을 사용할 수 있는 권한이 없습니다.

- ec2:DescribeIdFormat
- ec2:DescribeIdentityIdFormat
- ec2:DescribeAggregateIdFormat
- ec2:DescribePrincipalIdFormat
- ec2:ModifyIdFormat
- ec2:ModifyIdentityIdFormat

예를 들어 IAM 역할은 정책 명령문에서 "Action": "ec2:*" 요소를 통해 모든 Amazon EC2 작업을 사용할 수 있는 권한을 가질 수 있습니다.

IAM 사용자와 역할이 그들 스스로 또는 내 계정에 있는 다른 사용자와 역할에 대해 더 긴 리소스 ID 설정을 보거나 수정하지 못하게 하려면, IAM 정책에 다음 명령문을 포함시키십시오.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Deny",  
            "Action": [  
                "ec2:ModifyIdFormat",  
                "ec2:DescribeIdFormat",  
                "ec2:ModifyIdentityIdFormat",  
                "ec2:DescribeIdentityIdFormat"  
            ]  
        }  
    ]  
}
```

```
    "ec2:DescribeIdentityIdFormat",
    "ec2:DescribeAggregateIdFormat",
    "ec2:DescribePrincipalIdFormat"
],
"Resource": "*"
}
}
```

다음 작업에 대한 리소스 수준 권한은 지원되지 않습니다.

- `ec2:DescribeIdFormat`
- `ec2:DescribeIdentityIdFormat`
- `ec2:DescribeAggregateIdFormat`
- `ec2:DescribePrincipalIdFormat`
- `ec2:ModifyIdFormat`
- `ec2:ModifyIdentityIdFormat`

리소스 목록화 및 필터링

사용자는 Amazon EC2 콘솔을 이용하여 리소스의 유형 목록을 획득할 수 있습니다. 사용자는 해당 명령 또는 API 작업을 이용하여 리소스의 각 유형 목록을 획득할 수 있습니다. 리소스가 많은 경우 사용자는 결과를 필터링하여 특정 기준에 부합하는 리소스만을 포함시킬 수 있습니다.

목차

- [고급 검색 \(p. 1138\)](#)
- [콘솔을 이용하여 리소스 목록화 \(p. 1139\)](#)
- [콘솔을 이용하여 리소스 필터링 \(p. 1140\)](#)
- [CLI 및 API를 이용하여 목록화 및 필터링 \(p. 1141\)](#)

고급 검색

고급 검색을 사용하면 필터 조합을 이용한 검색으로 정밀한 결과를 얻을 수 있습니다. 필터링은 키워드, 사용자 지정 태그 키 및 사전 정의된 리소스 속성으로 수행될 수 있습니다.

사용 가능한 검색 유형:

- **키워드로 검색**

키워드로 검색하려면 검색하려는 키워드를 검색 상자에 입력 또는 붙여넣기한 다음 Enter를 누릅니다. 예를 들어, 특정 인스턴스를 검색하려면 인스턴스 ID를 입력합니다.

- **필드로 검색**

또한, 리소스와 관련이 있는 필드, 태그 및 속성으로 검색하는 것도 가능합니다. 예를 들어, 중지 상태인 모든 인스턴스를 검색하려면

1. 검색 상자에 **Instance State**를 입력합니다. 입력하기 시작하면 추천 필드 목록이 표시됩니다.
2. 목록에서 인스턴스 상태를 선택합니다.
3. 추천 값 목록에서 중지됨을 선택합니다.
4. 목록을 미세 조정하려면 검색 상자를 클릭하여 추가 검색 옵션을 선택합니다.

- **고급 검색**

사용자는 여러 필터를 추가하여 어드밴스 쿼리를 생성할 수 있습니다. 예를 들어, 태그별 검색을 하여 프로덕션 스택에서 실행 중인 Flying Mountain 프로젝트 인스턴스를 확인한 다음 속성별 검색을 하면 모든 t2.micro 인스턴스 또는 모든 us-west-2a 인스턴스 또는 두 인스턴스 모두가 표시됩니다.

- 역검색

사용자는 지정된 값에 일치하지 않는 리소스를 검색할 수 있습니다. 예를 들어, 종료되지 않은 모든 인스턴스를 목록화하려면 Instance State 필드로 검색한 다음 느낌표(!)와 함께 종료된 값을 접두사로 사용합니다.

- 부분 검색

필드별로 검색하는 경우 부분 문자열을 입력하여 해당 필드에 있는 문자열에 포함된 모든 리소스를 검색할 수 있습니다. 예를 들어, 인스턴스 유형으로 검색한 후 t2를 입력하면 모든 t2.micro, t2.small 또는 t2.medium 인스턴스를 검색할 수 있습니다.

- 정규식

필드의 값이 특정 패턴에 맞아야 하는 경우 정규식을 유용하게 활용할 수 있습니다. 예를 들어, 이름 태그별로 검색한 다음 ^s.*를 입력하면 's'로 시작하는 이름 태그를 갖는 모든 인스턴스가 표시됩니다. 정규식 검색은 대소문자를 구별하지 않습니다.

검색 결과를 확인한 이후에는 차후에 편리하게 참조할 수 있도록 URL을 즐겨찾기에 등록할 수 있습니다. 인스턴스가 수 천개 있는 경우 필터링하고 즐겨찾기를 등록하면 검색을 반복할 필요가 없어 시간을 크게 줄일 수 있습니다.

검색 필터 결합

일반적으로, 키 필드가 동일한 다중 필터(예를 들어, tag:Name, search, Instance State)는 자동으로 OR로 조인됩니다. AND로 조인되면 대부분의 필터가 논리에 맞지 않기 때문에 이 설정은 의도적인 것입니다. 예를 들어, Instance State=running AND Instance State=stopped로 검색하면 검색 결과가 제공되지 않을 것입니다. 많은 경우에 서로 다른 키 필드에서 보완적인 검색 용어를 사용함으로써 검색 결과를 조정할 수 있고 이 경우 AND 규칙이 자동으로 대체 적용됩니다. tag: Name:=All values and tag:Instance State=running으로 검색을 수행하면 두 기준 모두를 만족하는 검색 결과가 제공됩니다. 결과를 미세 조정하기 위해서는 검색 결과가 만족스러울 때까지 필터를 하나씩 제거하면 됩니다.

콘솔을 이용하여 리소스 목록화

사용자는 콘솔을 이용하여 자주 사용하는 Amazon EC2 리소스의 유형 목록을 확인할 수 있습니다. 추가 리소스를 확인하려면 명령줄 인터페이스 또는 API 작업을 사용합니다.

콘솔을 이용하여 EC2 리소스를 목록화하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 AMI 또는 인스턴스 등 리소스에 해당하는 옵션을 선택합니다.

EC2 Dashboard New

Events **New**

Tags

Reports

Limits

▼ INSTANCES

Instances

Instance Types

Launch Templates **New**

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts **New**

Scheduled Instances

Capacity Reservations

▼ IMAGES

AMIs

Bundle Tasks

▼ ELASTIC BLOCK STORE

Volumes

3. 이 페이지는 사용 가능한 모든 리소스를 표시합니다.

콘솔을 이용하여 리소스 필터링

사용자는 Amazon EC2 콘솔을 이용하여 자주 사용하는 리소스 유형을 필터링 및 정렬할 수 있습니다. 예를 들어, 인스턴스 페이지의 검색 창을 사용하여 태그, 속성 또는 키워드별로 인스턴스를 정렬할 수 있습니다.

또한, 사용자는 각 페이지에서 필드를 검색하여 특정 속성 또는 값이 있는 리소스를 검색할 수 있습니다. 정규식을 사용하여 부분 또는 다중 문자열을 검색하는 것도 가능합니다. 예를 들어, MySG 보안 그룹을 사용하는 모든 인스턴스를 검색하여 검색 필드에 MySG를 입력합니다. 그러면 MySG 및 MySG2 등 문자열의 일부로 MySG3가 있는 모든 값이 검색 결과에 표시됩니다. 결과를 MySG만으로 제한하려면 검색 필드에 \bMySG\b를 입력합니다. 유형이 m1.small 또는 m1.large인 모든 인스턴스를 목록화하려면 검색 필드에 m1.small|m1.large를 입력합니다.

상태가 **us-east-1b**인 **available** 가용 영역에서 볼륨을 목록화하려면

1. 탐색 창에서 볼륨을 선택합니다.
2. 검색 상자를 클릭하고 메뉴에서 연결 상태를 선택한 후 분리됨을 선택합니다. (분리된 볼륨은 동일 가용 영역에 있는 인스턴스에 연결하는 데 사용될 수 있습니다.)
3. 검색 상자를 다시 클릭한 후 상태를 선택하고 사용 가능을 선택합니다.
4. 검색 상자를 다시 클릭한 후 가용 영역을 선택하고 **us-east-1b**를 선택합니다.

5. 이 기준에 부합하는 모든 볼륨이 표시됩니다.

퍼블릭 64비트 Amazon EBS 지원 Windows AMI를 목록화하려면,

1. 탐색 창에서 AMI를 선택합니다.
2. 필터 창에서 퍼블릭 이미지, EBS 이미지를 선택한 후 목록에서 your Linux distribution(Linux 배포)Windows를 선택합니다.
3. 검색 필드에 x86_64을 입력합니다.
4. 이 기준에 부합하는 모든 AMI가 표시됩니다.

CLI 및 API를 이용하여 목록화 및 필터링

각 리소스 유형에는 사용자가 해당 유형의 리소스를 목록화하기 위해 사용하는 해당 CLI 명령 및 API 작업이 있습니다. 결과 리소스 목록은 길이가 길 수 있기 때문에 결과를 필터링하여 특정 기준에 부합하는 리소스만 포함시키는 것이 더 빠르고 유용할 수 있습니다.

필터링 고려 사항

- 단일 요청에서 여러 필터와 여러 필터 값을 지정할 수 있습니다.
- 또한 필터 값과 함께 와일드카드를 사용할 수 있습니다. 별표(*)는 0개 이상의 문자에 해당하고 물음표(?)는 0개 또는 1개의 문자에 해당합니다.
- 필터 값은 대소문자를 구분합니다.
- 검색에는 와이드카드 문자의 리터럴 값이 포함될 수 있고 문자 앞에 백슬래시를 사용하면 벗어날 수 있습니다. 예를 들어, *amazon\?\\ 값은 리터럴 문자열 *amazon?\ 을 검색합니다.

지원되는 필터

각 Amazon EC2 리소스에 대해 지원되는 필터를 보려면 다음 설명서를 참조하십시오.

- AWS CLI: [AWS CLI Command Reference-Amazon EC2](#)의 describe 명령입니다.
- Windows PowerShell용 도구: [PowerShell용 AWS 도구 Cmdlet Reference-Amazon EC2](#)의 Get 명령입니다.
- 쿼리 API: [Amazon EC2 API Reference](#)의 Describe API 작업입니다.

Example 예: 단일 필터 지정

`describe-instances`를 사용하여 Amazon EC2 인스턴스를 나열할 수 있습니다. 필터가 없으면 응답에는 모든 리소스에 대한 정보가 포함됩니다. 다음 명령을 사용하여 출력에 실행 중인 인스턴스만 포함할 수 있습니다.

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=running
```

실행 중인 인스턴스의 인스턴스 ID만 나열하려면 다음과 같이 --query 파라미터를 추가합니다.

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=running --query "Reservations[*].Instances[*].InstanceId" --output text
```

다음은 예제 출력입니다.

```
i-0ef1f57f78d4775a4
i-0626d4edd54f1286d
```

```
i-04a636d18e83cfacb
```

Example 예: 여러 필터 또는 필터 값 지정

여러 필터 또는 여러 필터 값을 지정하는 경우, 리소스는 결과에 포함할 모든 필터와 일치해야 합니다.

다음 명령을 사용하여 유형이 m5.large 또는 m5d.large인 모든 인스턴스를 나열할 수 있습니다.

```
aws ec2 describe-instances --filters Name=instance-type,Values=m5.large,m5d.large
```

다음 명령을 사용하여 유형이 t2.micro인 종지된 모든 인스턴스를 나열할 수 있습니다.

```
aws ec2 describe-instances --filters Name=instance-state-name,Values=stopped Name=instance-type,Values=t2.micro
```

Example 예: 필터 값에 와일드카드 사용

`describe-snapshots`를 사용하여 EBS 스냅샷을 설명할 때 `description` 필터에 대한 필터 값으로 데이터베이스를 지정하면 명령은 설명이 "database"인 스냅샷만 반환합니다.

```
aws ec2 describe-snapshots --filters Name=description,Values=database
```

* 와일드카드는 0개 이상의 문자와 일치합니다. *database*를 필터 값으로 지정하는 경우 명령은 설명에 database라는 단어가 포함된 스냅샷만 반환합니다.

```
aws ec2 describe-snapshots --filters Name=description,Values=*database*
```

? 와일드카드는 정확히 1개 문자와 일치합니다. database?를 필터 값으로 지정하는 경우 명령은 설명이 "database"이거나 "database" 뒤에 한 문자가 있는 스냅샷만 반환합니다.

```
aws ec2 describe-snapshots --filters Name=description,Values=database?
```

database????를 지정하면 명령은 설명에 "database" 뒤에 최대 4개 문자가 있는 스냅샷만 반환합니다. "database" 뒤에 5개 이상의 문자가 있는 설명은 제외됩니다.

```
aws ec2 describe-snapshots --filters Name=description,Values=database????
```

Example 예: 날짜를 기준으로 필터링

AWS CLI를 사용하면 JMESPath를 통해 표현식을 사용하여 결과를 필터링할 수 있습니다. 예를 들어, 다음 `describe-snapshots` 명령은 지정된 날짜(2020-03-31로 표시) 이전에 AWS 계정에서 생성된 모든 스냅샷의 ID(123456789012로 표시)를 표시합니다. 소유자를 지정하지 않으면 모든 퍼블릭 스냅샷이 결과에 포함됩니다.

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query "Snapshots[?(StartTime<=`2020-03-31`)].[SnapshotId]" --output text
```

다음 명령은 지정된 날짜 범위에 생성된 모든 스냅샷의 ID를 표시합니다.

```
aws ec2 describe-snapshots --filters Name=owner-id,Values=123456789012 --query "Snapshots[?(StartTime>=`2019-01-01`)&&(StartTime<=`2019-12-31`)].[SnapshotId]" --output text
```

태그를 기준으로 필터링

태그에 따라 리소스 목록을 필터링하는 방법에 대한 예는 [명령줄을 사용한 태그 작업 \(p. 1151\)](#) 단원을 참조 하십시오.

Amazon EC2 리소스에 태그 지정

고유 메타데이터를 태그의 형태로 각 리소스에 배정하면 인스턴스, 이미지 및 기타 Amazon EC2 리소스를 쉽게 관리할 수 있습니다. 태그를 사용하면 용도, 소유자 또는 환경을 기준으로 하는 등 AWS 리소스를 다양한 방식으로 분류할 수 있습니다. 이 기능은 동일 유형의 리소스가 많을 때 유용합니다. 지정한 태그에 따라 특정 리소스를 빠르게 식별할 수 있습니다. 이 주제에서는 태그를 설명하고 태그를 생성하는 방법을 보여 줍니다.

Warning

태그 키 및 해당 값은 다양한 API 호출에 의해 반환됩니다. `DescribeTags`에 대한 액세스를 거부해도 다른 API가 반환한 태그에 대한 액세스는 자동으로 거부되지 않습니다. 민감한 데이터를 태그에 포함하지 않는 것이 가장 좋습니다.

목차

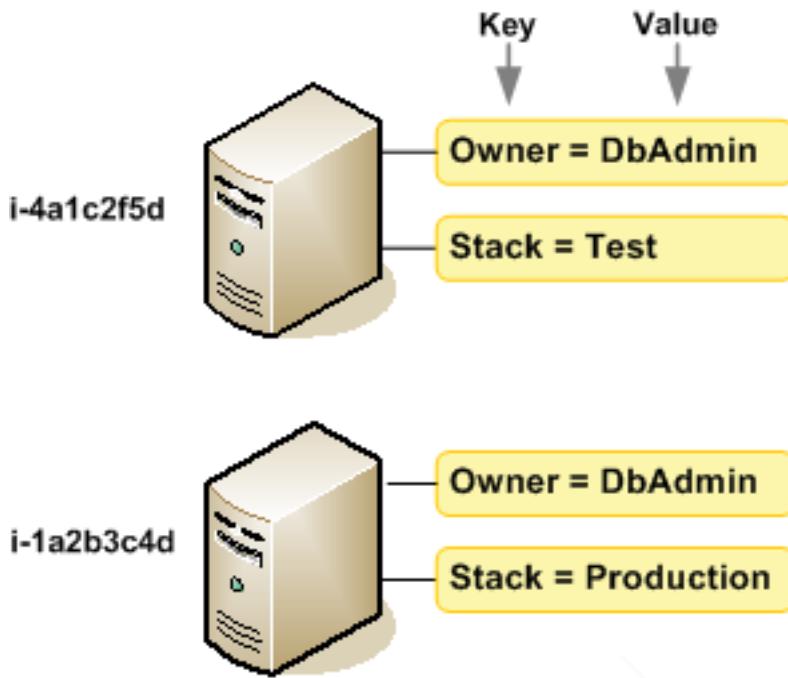
- [태그 기본 사항 \(p. 1143\)](#)
- [리소스에 태그 지정 \(p. 1144\)](#)
- [태그 제한 \(p. 1147\)](#)
- [리소스에 결제용 태그 지정 \(p. 1147\)](#)
- [콘솔을 사용한 태그 작업 \(p. 1148\)](#)
- [명령줄을 사용한 태그 작업 \(p. 1151\)](#)
- [CloudFormation을 사용하여 리소스에 태그 추가 \(p. 1154\)](#)

태그 기본 사항

태그란 AWS 리소스에 할당되는 레이블을 말합니다. 각 태그는 사용자가 정의하는 키와 선택적 값으로 구성됩니다.

태그를 사용하면 용도, 소유자 또는 환경을 기준으로 하는 등 AWS 리소스를 다양한 방식으로 분류할 수 있습니다. 예를 들어, 계정의 Amazon EC2 인스턴스에 대해 각 인스턴스의 소유자나 스택 수준을 추적하는 데 도움이 되는 태그 세트를 정의할 수 있습니다.

다음 다이어그램은 태그 지정 방식을 설명합니다. —이 예에서는 두 가지 태그, 즉 `Owner`라는 키가 있는 태그 하나와 `Stack`이라는 키가 있는 태그 하나를 각 인스턴스에 배정했습니다. 또한 각 태그에는 연결된 값이 있습니다.



각 리소스 유형에 대한 요건을 충족하는 태그 키 세트를 고안하는 것이 좋습니다. 일관된 태그 키 세트를 사용하면 리소스를 보다 쉽게 관리할 수 있습니다. 추가하는 태그에 따라 리소스를 검색하고 필터링할 수 있습니다. 효과적인 리소스 태그 지정 전략을 구현하는 방법에 대한 자세한 내용은 AWS 백서 [태그 지정 모범 사례](#)를 참조하십시오.

태그는 Amazon EC2에는 의미가 없으며 엄격하게 문자열로 해석됩니다. 또한 태그는 리소스에 자동으로 배정되지 않습니다. 태그 키와 값을 편집할 수 있으며 언제든지 리소스에서 태그를 제거할 수 있습니다. 태그의 값을 빈 문자열로 설정할 수 있지만 태그의 값을 Null로 설정할 수는 없습니다. 해당 리소스에 대해 키가 기존 태그와 동일한 태그를 추가하는 경우 새 값이 이전 값을 덮어씁니다. 리소스를 삭제하면 리소스 태그도 삭제됩니다.

AWS Management 콘솔, AWS CLI, Amazon EC2 API를 사용하여 태그 관련 작업을 수행할 수 있습니다.

AWS Identity and Access Management(IAM)을 사용하는 경우 AWS 계정에서 태그를 생성, 편집 또는 삭제 할 수 있는 권한이 있는 사용자를 제어할 수 있습니다. 자세한 내용은 [Amazon EC2의 자격 증명 및 액세스 관리](#) (p. 836) 단원을 참조하십시오.

리소스에 태그 지정

계정에 이미 존재하는 대부분의 Amazon EC2 리소스에 태그를 지정할 수 있습니다. 아래의 표 (p. 1145)에 태그 지정을 지원하는 리소스가 나와 있습니다.

Amazon EC2 콘솔을 사용하는 경우, 관련 리소스 화면에서 태그 탭을 사용하여 리소스에 태그를 적용하거나 태그 화면을 사용할 수 있습니다. 일부 리소스 화면을 사용하면 리소스를 생성할 때 리소스에 대해 태그를 지정할 수 있습니다. 예를 들어 **Name**의 키가 있는 태그와 지정하는 값이 있습니다. 대부분의 경우, 콘솔은 리소스 생성 직후(리소스 생성 중이 아니라) 태그를 적용합니다. 콘솔은 **Name** 태그에 따라 리소스를 조작할 수 있지만 이 태그는 Amazon EC2 서비스에 대한 의미가 없습니다.

Amazon EC2 API, AWS CLI 또는 AWS SDK를 사용하는 경우, `CreateTags` EC2 API 작업을 사용하여 기존 리소스에 태그를 적용할 수 있습니다. 또한 일부 리소스 생성 작업에서는 리소스 생성 시 리소스의 태그를

지정할 수 있습니다. 리소스 생성 도중 태그를 적용할 수 없는 경우, 리소스 생성 프로세스가 끌백됩니다. 이는 태그를 사용하여 리소스가 생성되거나 아예 리소스가 생성되지 않도록 하고 언제든 태그 지정되지 않은 리소스가 남지 않게 합니다. 생성 시 리소스에 태그를 지정하면 리소스 생성 후 사용자 지정 태그 지정 스크립트를 실행할 필요가 없습니다.

다음 표는 태그를 지정할 수 있는 Amazon EC2 리소스와 Amazon EC2 API, AWS CLI 또는 AWS SDK를 사용하여 생성 시 태그를 지정할 수 있는 리소스를 설명합니다.

Amazon EC2 리소스 태그 지정 지원

리소스	태그 지원	생성 시 태그 지정 지원
AFI	예	예
AMI	예	아니요
번들 작업	아니요	아니요
용량 예약	예	예
통신 사업자 게이트웨이	예	예
클라이언트 VPN 엔드포인트	예	예
클라이언트 VPN 경로	아니요	아니요
고객 게이트웨이	예	예
전용 호스트	예	예
전용 호스트 예약	예	예
DHCP 옵션	예	예
EBS 스냅샷	예	예
EBS 볼륨	예	예
EC2 집합	예	예
외부 전용 인터넷 게이트웨이	예	예
탄력적 IP 주소	예	아니요
Elastic Graphics 액셀러레이터	예	아니요
인스턴스	예	예
인스턴스 스토어 볼륨	해당 사항 없음	해당 사항 없음
인터넷 게이트웨이	예	예
IP 주소 풀(BYOD)	예	예
키 페어	예	예
시작 템플릿	예	예
시작 템플릿 버전	아니요	아니요
로컬 게이트웨이	예	아니요
로컬 게이트웨이 라우팅 테이블	예	아니요

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
리소스에 태그 지정

리소스	태그 지원	생성 시 태그 지정 지원
로컬 게이트웨이 가상 인터페이스	예	아니요
로컬 게이트웨이 가상 인터페이스 그룹	예	아니요
로컬 게이트웨이 라우팅 테이블 VPC 연결	예	아니요
로컬 게이트웨이 라우팅 테이블 가상 인터페이스 그룹 연결	예	아니요
NAT 게이트웨이	예	예
네트워크 ACL	예	예
네트워크 인터페이스	예	예
배치 그룹	예	예
접두사 목록	예	예
예약 인스턴스	예	아니요
예약 인스턴스 목록	아니요	아니요
라우팅 테이블	예	예
스팟 플릿 요청	예	예
스팟 인스턴스 요청	예	예
보안 그룹	예	예
서브넷	예	예
트래픽 미러 필터	예	예
트래픽 미러 세션	예	예
트래픽 미러 대상	예	예
전송 게이트웨이	예	예
전송 게이트웨이 라우팅 테이블	예	예
전송 게이트웨이 VPC 연결	예	예
가상 프라이빗 게이트웨이	예	예
VPC	예	예
VPC 엔드포인트	예	예
VPC 엔드포인트 서비스	예	예
VPC 엔드포인트 서비스 구성	예	예
VPC 흐름 로그	예	예
VPC 피어링 연결	예	예

리소스	태그 지원	생성 시 태그 지정 지원
VPN 연결	예	예

Amazon EC2 콘솔에서 Amazon EC2 인스턴스 시작 마법사를 사용하여 생성할 때 인스턴스와 볼륨에 태그를 지정할 수 있습니다. 볼륨 화면을 사용하여 생성 시 EBS 볼륨에 태그를 지정하거나 스냅샷 화면을 사용하여 EBS 스냅샷에 태그를 지정할 수 있습니다. 또는 리소스를 만들 때 리소스 생성 Amazon EC2 API(예: [RunInstances](#))를 사용하여 태그를 적용하십시오.

생성 시 태그를 지원하는 Amazon EC2 API 작업에 IAM 정책의 태그 기반 리소스 수준 권한을 적용하여 생성 시 리소스에 태그를 지정할 수 있는 사용자와 그룹을 세밀하게 제어할 수 있습니다. 리소스를 생성하면 태그가 즉시 적용되기 때문에 생성 단계부터 리소스를 적절하게 보호할 수 있습니다. 따라서 태그를 기반으로 리소스 사용을 제어하는 리소스 권한이 즉시 발효됩니다.— 이에 따라 더욱 정확한 리소스 추적 및 보고가 가능합니다. 새 리소스에서 태그 지정 사용을 적용하고 리소스에서 어떤 태그 키와 값이 설정되는지 제어할 수 있습니다.

IAM 정책에서 [CreateTags](#) 및 [DeleteTags](#) Amazon EC2 API 작업에 리소스 수준 권한을 적용하여 기존 리소스에서 어떤 태그 키와 값이 설정되는지 제어할 수도 있습니다. 자세한 내용은 [예제: 리소스에 태그 지정 \(p. 874\)](#) 단원을 참조하십시오.

결제를 위한 리소스 태그 지정에 대한 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 [비용 할당 태그 사용](#) 단원을 참조하십시오.

태그 제한

태그에 적용되는 기본 제한은 다음과 같습니다.

- 리소스당 최대 태그 수 - 50개
- 각 리소스에 대해 각 태그 키는 고유하며 하나의 값만 가질 수 있습니다.
- 최대 키 길이 – UTF-8의 유니코드 문자 128자
- 최대 값 길이 – 유니코드 문자 256자(UTF-8)
- EC2는 태그에 모든 문자를 사용할 수 있지만, 다른 서비스에는 제한이 적용되기도 합니다. 서비스에서 허용되는 문자는 UTF-8로 표현할 수 있는 문자, 숫자 및 공백과 특수 문자 + - = . _ : / @입니다.
- 태그 키와 값은 대/소문자를 구분합니다.
- aws: 접두사는 AWS용으로 예약되어 있습니다. 태그에 이 접두사가 있는 태그 키가 있는 경우 태그의 키 또는 값을 편집하거나 삭제할 수 없습니다. aws: 접두사가 지정된 태그는 리소스당 태그 수 제한에 포함되지 않습니다.

태그에만 기초하여 리소스를 종료, 중지 또는 삭제할 수 없습니다. 리소스 식별자를 지정해야 합니다. 예를 들어 DeleteMe라는 태그 키로 태그를 지정한 스냅샷을 삭제하려면 해당 스냅샷의 리소스 식별자(예: [DeleteSnapshots](#))를 지정하여 snap-1234567890abcdef0 작업을 사용해야 합니다.

퍼블릭 또는 공유 리소스에 태그를 지정할 수 있지만, 배정하는 태그는 AWS 계정에만 사용할 수 있으며 해당 리소스를 공유하는 다른 계정에는 사용할 수 없습니다.

모든 리소스에 태그를 지정할 수는 없습니다. 자세한 내용은 [Amazon EC2 리소스 태그 지정 지원 \(p. 1145\)](#) 단원을 참조하십시오.

리소스에 결제용 태그 지정

태그를 사용하여 비용 구조를 반영하도록 AWS 청구서를 구성할 수 있습니다. 이렇게 하려면 가입하여 태그 키 값이 포함된 AWS 계정 청구서를 가져옵니다. 태그를 사용한 비용 할당 보고서 설정에 대한 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 [월간 비용 할당 보고서](#) 단원을 참조하십시오. 결합된

리소스의 비용을 확인하려면 태그 키 값을 동일한 리소스에 따라 결제 정보를 구성할 수 있습니다. 예를 들어, 특정 애플리케이션 이름으로 여러 리소스에 태그를 지정한 다음 결제 정보를 구성하여 여러 서비스에 걸친 해당 애플리케이션의 총 비용을 볼 수 있습니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 [비용 할당 태그 사용](#)을 참조하십시오.

Note

방금 보고서를 활성화한 경우, 24시간 후에 이번 달의 데이터를 볼 수 있습니다.

비용 할당 태그는 비용에 기여하는 리소스를 나타낼 수 있지만, 리소스를 삭제하거나 비활성화한다고 해서 비용이 항상 절감되는 것은 아닙니다. 예를 들어, 원본 데이터가 포함된 스냅샷을 삭제하더라도 다른 스냅샷에서 참조하는 스냅샷 데이터는 보존됩니다. 자세한 내용은 AWS Billing and Cost Management 사용 설명서의 [Amazon Elastic Block Store 블룸 및 스냅샷](#)을 참조하십시오.

Note

태그된 탄력적 IP 주소는 비용 할당 보고서에 나타나지 않습니다.

콘솔을 사용한 태그 작업

Amazon EC2 콘솔을 사용하여 동일 리전의 모든 Amazon EC2 리소스에서 사용 중인 태그를 볼 수 있습니다. 리소스와 리소스 유형별로 태그를 볼 수 있으며, 지정된 태그와 연결되어 있는 각 리소스 유형의 항목 수를 볼 수 있습니다. 또한 Amazon EC2 콘솔을 사용하여 한 번에 하나 이상의 리소스에서 태그를 적용하거나 제거할 수도 있습니다.

리소스 나열 시 필터 사용에 대한 자세한 내용은 [리소스 목록화 및 필터링 \(p. 1138\)](#) 단원을 참조하십시오.

태그를 중앙에서 통합 생성 및 관리할 수 있는 AWS Management 콘솔의 Tag Editor는 사용하기 쉽고 최상의 결과를 발휘합니다. 자세한 내용은 AWS Management Console 시작하기의 [Tag Editor 작업](#)을 참조하십시오.

작업

- [태그 표시 \(p. 1148\)](#)
- [개별 리소스에 대한 태그 추가 및 삭제 \(p. 1149\)](#)
- [리소스 그룹에 태그 추가 및 삭제 \(p. 1150\)](#)
- [인스턴스 시작 시 태그 추가 \(p. 1150\)](#)
- [태그를 기준으로 리소스 목록 필터링 \(p. 1151\)](#)

태그 표시

Amazon EC2 콘솔에는 두 가지 방법으로 태그를 표시할 수 있습니다. 개별 리소스에 대한 태그를 표시하거나 모든 리소스에 대한 태그를 표시할 수 있습니다.

개별 리소스에 대한 태그 표시

Amazon EC2 콘솔에서 리소스 관련 페이지를 선택하면 이러한 리소스의 목록이 표시됩니다. 예를 들어, 텁색 창에서 인스턴스를 선택하는 경우 콘솔에 Amazon EC2 인스턴스 목록이 표시됩니다. 이러한 목록 중 하나(예: 인스턴스)에서 리소스를 선택하는 경우 해당 리소스에서 태그를 지원하면 관련 태그를 보고 관리할 수 있습니다. 대부분의 리소스 페이지에서는 세부 정보 창의 태그 탭에서 태그를 볼 수 있습니다.

키가 동일한 태그의 값을 모두 표시하는 열을 하나 리소스 목록에 추가할 수 있습니다. 이 열을 사용하면 태그별로 리소스 목록을 정렬하고 필터링할 수 있습니다. 리소스에 새 열을 추가하여 태그를 표시하는 방법에는 두 가지가 있습니다.

- 태그 탭에서 열 표시를 선택합니다. 새 열이 콘솔에 추가됩니다.

- 열 표시/숨기기 기어 모양 아이콘을 선택하고 열 표시/숨기기 대화 상자의 태그 키에서 태그 키를 선택합니다.

모든 리소스에 대한 태그 표시

Amazon EC2 콘솔의 탐색 창에서 태그를 선택하여 모든 리소스의 태그를 표시할 수 있습니다. 다음 이미지는 리소스 유형별로 사용 중인 모든 태그를 나열하는 태그 창을 보여줍니다.

Manage Tags						
Filter:		Search Keys	X	Search Values	X	1 to 7 of 7 Tags
	Tag Key	Tag Value	Total	Instances	AMIs	Volumes
Manage Tag	Name	DNS Server	1	1	0	0
Manage Tag	Owner	TeamB	2	0	0	2
Manage Tag	Owner	TeamA	2	0	0	2
Manage Tag	Purpose	Project2	1	0	0	1
Manage Tag	Purpose	Logs	1	0	0	1
Manage Tag	Purpose	Network Management	1	1	0	0
Manage Tag	Purpose	Project1	2	0	0	2

개별 리소스에 대한 태그 추가 및 삭제

리소스 페이지에서 개별 리소스에 대한 태그를 직접 관리할 수 있습니다.

개별 리소스에 태그를 추가하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 모음에서 요구에 맞는 리전을 선택합니다. 일부 Amazon EC2 리소스는 리전 간에 공유될 수 있지만 그렇지 않은 리소스도 있으므로 잘 선택해야 합니다. 자세한 정보는 [리소스 위치 \(p. 1132\)](#) 단원을 참조하십시오.
- 탐색 창에서 리소스 유형(예: 인스턴스)을 선택합니다.
- 리소스 목록에서 리소스를 선택한 다음 태그, 태그 추가/편집을 선택합니다.
- 태그 추가/편집 대화 상자에서 각 태그에 대한 키와 값을 지정하고 저장을 선택합니다.

개별 리소스에서 태그를 삭제하려면

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 모음에서 요구에 맞는 리전을 선택합니다. 일부 Amazon EC2 리소스는 리전 간에 공유될 수 있지만 그렇지 않은 리소스도 있으므로 잘 선택해야 합니다. 자세한 정보는 [리소스 위치 \(p. 1132\)](#) 단원을 참조하십시오.
- 탐색 창에서 리소스 유형(예: 인스턴스)을 선택합니다.
- 리소스 목록에서 리소스를 선택하고 태그를 선택합니다.
- 태그 추가/편집을 선택하고 태그의 삭제 아이콘을 선택한 다음 저장을 선택합니다.

리소스 그룹에 태그 추가 및 삭제

리소스 그룹에 태그를 추가하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 요구에 맞는 리전을 선택합니다. 일부 Amazon EC2 리소스는 리전 간에 공유될 수 있지만 그렇지 않은 리소스도 있으므로 잘 선택해야 합니다. 자세한 정보는 [리소스 위치 \(p. 1132\)](#) 단원을 참조하십시오.
3. 탐색 창에서 태그를 선택합니다.
4. 콘텐츠 창 맨 위에서 태그 관리를 선택합니다.
5. 필터에서 태그를 추가할 리소스 유형(예: 인스턴스)을 선택합니다.
6. 리소스 목록에서 태그를 추가할 각 리소스 옆의 확인란을 선택합니다.
7. 태그 추가에서 키 및 값에 태그 키와 값을 입력한 다음 태그 추가를 선택합니다.

Note

태그 키가 기존 태그와 동일한 새 태그를 추가하는 경우 새 태그가 기존 태그를 덮어씁니다.

리소스 그룹에서 태그를 제거하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 요구에 맞는 리전을 선택합니다. 일부 Amazon EC2 리소스는 리전 간에 공유될 수 있지만 그렇지 않은 리소스도 있으므로 잘 선택해야 합니다. 자세한 정보는 [리소스 위치 \(p. 1132\)](#) 단원을 참조하십시오.
3. 탐색 창에서 태그와 태그 관리를 선택합니다.
4. 사용 중인 태그를 보려면 열 표시/숨기기 기어 모양 아이콘을 선택하고, 열 표시/숨기기 대화 상자에서 조회할 태그 키를 선택한 다음 닫기를 클릭합니다.
5. 필터에서 태그를 제거할 리소스 유형(예: 인스턴스)을 선택합니다.
6. 리소스 목록에서 태그를 제거할 각 리소스 옆에 있는 확인란을 선택합니다.
7. 태그 제거의 키 상자에 태그의 이름을 입력한 다음 태그 제거를 선택합니다.

인스턴스 시작 시 태그 추가

마법사 시작을 사용하여 태그를 추가하려면

1. 탐색 모음에서 인스턴스를 시작할 지역을 선택합니다. 일부 Amazon EC2 리소스는 리전 간에 공유될 수 있지만 그렇지 않은 리소스도 있으므로 잘 선택해야 합니다. 요구 사항을 충족하는 리전을 선택합니다. 자세한 정보는 [리소스 위치 \(p. 1132\)](#) 단원을 참조하십시오.
2. 인스턴스 시작을 선택합니다.
3. Amazon 머신 이미지(AMI) 선택 페이지에서는 Amazon 머신 이미지(AMI)라 불리는 일련의 기본 구성들을 목록으로 표시합니다. 사용할 AMI를 선택하고 선택을 선택합니다. AMI 선택에 대한 자세한 내용은 [Windows AMI 찾기 \(p. 68\)](#) 단원을 참조하십시오.
4. 인스턴스 세부 정보 구성 페이지에서 필요에 따라 인스턴스 설정을 구성하고 다음: 스토리지 추가(Next: Add Storage)를 선택합니다.
5. 스토리지 추가 페이지에서 인스턴스에 대한 추가 스토리지 볼륨을 지정할 수 있습니다. 모두 마쳤으면 다음: 태그 추가(Next: Add Tags)를 선택합니다.
6. 태그 추가 페이지에서 인스턴스나 볼륨 또는 이 둘의 태그를 지정합니다. 태그 추가를 선택하여 리소스에 한 개 이상의 태그를 인스턴스에 추가할 수 있습니다. 모두 마쳤으면 다음: 보안 그룹 구성(Next: Configure Security Group)을 선택합니다.

7. 보안 그룹 구성 페이지에서 소유하는 기존 보안 그룹 중 하나를 선택하거나 마법사를 통해 새 보안 그룹을 생성합니다. 작업을 마치면 검토 후 시작(Review and Launch)을 선택합니다.
8. 설정을 검토합니다. 선택한 항목에 만족하면 시작을 선택합니다. 기존 키 페어를 선택하거나 새 키 페어를 생성하고, 승인 확인란을 선택하고 인스턴스 시작을 선택합니다.

태그를 기준으로 리소스 목록 필터링

하나 이상의 태그 키와 태그 값에 따라 리소스 목록을 필터링할 수 있습니다.

태그를 기준으로 리소스 목록을 필터링하려면

1. 다음과 같이 태그 열을 표시합니다.
 - a. 리소스를 선택합니다.
 - b. 세부 정보 창에서 태그를 선택합니다.
 - c. 목록에서 태그를 찾고 열 표시를 선택합니다.
2. 태그 열의 오른쪽 위에 있는 필터 아이콘을 선택하여 필터 목록을 표시합니다.
3. 태그 값을 선택하고 필터 적용을 선택하여 결과 목록을 필터링합니다.

Note

필터에 대한 자세한 내용은 [리소스 목록화 및 필터링 \(p. 1138\)](#) 단원을 참조하십시오.

명령줄을 사용한 태그 작업

create 명령에 대한 태그 사양 파라미터를 사용하여 생성 시 여러 EC2 리소스에 태그를 추가할 수 있습니다. 리소스에 대한 describe 명령을 사용하여 리소스에 대한 태그를 볼 수 있습니다. 다음 명령을 사용하여 기존 리소스에 대한 태그를 추가, 업데이트 또는 삭제할 수도 있습니다.

작업	AWS CLI	Windows PowerShell용 AWS 도구
하나 이상의 태그를 추가하거나 덮어씁니다.	create-tags	New-EC2Tag
하나 이상의 태그를 삭제합니다.	delete-tags	Remove-EC2Tag
하나 이상의 태그에 대해 설명합니다.	describe-tags	Get-EC2Tag

작업

- [리소스 생성 시 태그 추가 \(p. 1151\)](#)
- [기존 리소스에 태그 추가 \(p. 1152\)](#)
- [태그가 지정된 리소스 설명 \(p. 1153\)](#)

리소스 생성 시 태그 추가

다음 예제는 리소스를 생성할 때 태그를 적용하는 방법을 보여 줍니다.

JSON 형식의 파라미터를 명령줄에 입력하는 방법은 운영 체제에 따라 다릅니다. Linux, macOS, Unix 및 Windows PowerShell에서는 작은 따옴표(')를 사용하여 JSON 데이터 구조를 룩습니다. Windows 명령줄에서 명령을 사용할 때 작은따옴표를 사용하지 마십시오. 자세한 내용은 [AWS Command Line Interface에 대한 파라미터 값 지정](#)을 참조하십시오.

Example 예제: 인스턴스를 시작하고 인스턴스와 볼륨에 태그 적용

다음 `run-instances` 명령은 인스턴스를 시작하고 키가 `webserver`이고 값이 `production`인 태그를 인스턴스에 적용합니다. 또한 이 명령은 생성되는 EBS 볼륨(이 경우에는 루트 볼륨)에 키가 `cost-center`이고 값이 `cc123`인 태그를 적용합니다.

```
aws ec2 run-instances \
    --image-id ami-abc12345 \
    --count 1 \
    --instance-type t2.micro \
    --key-name MyKeyPair \
    --subnet-id subnet-6e7f829e \
    --tag-specifications 'ResourceType=instance,Tags=[{"Key=webserver,Value=production"}]' \
    'ResourceType=volume,Tags=[{"Key=cost-center,Value=cc123"}]'
```

시작 중에 인스턴스와 볼륨에 동일한 태그 키와 값을 적용할 수 있습니다. 다음 명령은 인스턴스를 시작하고 키가 `cost-center`이고 값이 `cc123`인 태그를 인스턴스와 생성되는 일체의 EBS 볼륨에 적용합니다.

```
aws ec2 run-instances \
    --image-id ami-abc12345 \
    --count 1 \
    --instance-type t2.micro \
    --key-name MyKeyPair \
    --subnet-id subnet-6e7f829e \
    --tag-specifications 'ResourceType=instance,Tags=[{"Key=cost-center,Value=cc123"}]' \
    'ResourceType=volume,Tags=[{"Key=cost-center,Value=cc123"}]'
```

Example 예제: 볼륨 생성 및 태그 적용

다음 `create-volume` 명령은 볼륨을 생성하고 2개의 태그(`purpose=production` 및 `cost-center=cc123`)를 적용합니다.

```
aws ec2 create-volume \
    --availability-zone us-east-1a \
    --volume-type gp2 \
    --size 80 \
    --tag-specifications 'ResourceType=volume,Tags=[{"Key=purpose,Value=production"}, \
    {"Key=cost-center,Value=cc123"}]'
```

기존 리소스에 태그 추가

다음 예제에서는 `create-tags` 명령을 사용하여 기존 리소스에 태그를 추가하는 방법을 보여줍니다.

Example 예제: 리소스에 태그 추가

이 예제에서는 지정된 이미지에 `Stack=production` 태그를 추가하거나 태그 키가 `Stack`인 AMI의 기존 태그를 덮어씁니다. 이 명령이 성공하면 출력이 반환되지 않습니다.

```
aws ec2 create-tags \
    --resources ami-78a54011 \
    --tags Key=Stack,Value=production
```

Example 예제: 여러 리소스에 태그 추가

이 예제에서는 AMI와 인스턴스에 대해 두 개의 태그를 추가(또는 덮어쓰기)합니다. 태그 중 하나에 값이 없는 키(`webserver`)만 포함되어 있습니다(값을 빈 문자열로 설정). 다른 태그는 키(`stack`)와 값(`Production`)으로 구성됩니다. 이 명령이 성공하면 출력이 반환되지 않습니다.

```
aws ec2 create-tags \
```

```
--resources ami-1a2b3c4d i-1234567890abcdef0 \
--tags Key=webserver,Value= Key=stack,Value=Production
```

Example 예제: 특수 문자에 태그 추가

이 예제에서는 **[Group]=test** 태그를 인스턴스에 추가합니다. 대괄호([및])는 이스케이프해야 하는 특수 문자입니다.

Linux 또는 OS X를 사용하는 경우 특수 문자를 이스케이프하려면 특수 문자가 있는 요소를 큰 따옴표(")로 묶은 다음 전체 키 및 값 구조를 작은 따옴표(')로 묶으십시오.

```
aws ec2 create-tags \
--resources i-1234567890abcdef0 \
--tags 'Key="[Group]",Value=test'
```

Windows를 사용하는 경우 특수 문자를 이스케이프하려면 다음과 같이 특수 문자가 있는 요소를 큰 따옴표 ("")로 묶은 다음 각 큰 따옴표 문자 앞에 백슬래시(\)를 붙입니다.

```
aws ec2 create-tags ^
--resources i-1234567890abcdef0 ^
--tags Key=\"[Group]\",Value=test
```

Windows PowerShell을 사용하는 경우 특수 문자를 이스케이프하려면 다음과 같이 특수 문자가 있는 값을 큰 따옴표("")로 묶고 각 큰 따옴표 문자 앞에 백슬래시(\)를 붙인 다음 전체 키 및 값 구조를 작은 따옴표(')로 묶습니다.

```
aws ec2 create-tags ^
--resources i-1234567890abcdef0 ^
--tags 'Key=\"[Group]\",Value=test'
```

태그가 지정된 리소스 설명

다음 예제에서는 **describe-instances** 필터를 사용하여 특정 태그를 가진 인스턴스를 보는 방법을 보여줍니다. 모든 EC2 **describe** 명령은 이 구문을 사용하여 단일 리소스 유형에서 태그별로 필터링합니다. 또는 **describe-tags** 명령을 사용하여 EC2 리소스 유형에서 태그별로 필터링할 수 있습니다.

Example 예제: 지정된 태그 키를 가진 인스턴스에 대한 설명 제공

다음 명령은 태그 값과 상관없이 **Stack** 태그를 가진 인스턴스에 대한 설명을 제공합니다.

```
aws ec2 describe-instances \
--filters Name=tag-key,Values=Stack
```

Example 예제: 지정된 태그를 가진 인스턴스에 대한 설명 제공

다음 명령은 **Stack=production** 태그를 가진 인스턴스에 대한 설명을 제공합니다.

```
aws ec2 describe-instances \
--filters Name=tag:Stack,Values=production
```

Example 예제: 지정된 태그 값을 가진 인스턴스에 대한 설명 제공

다음 명령은 태그 키와 상관없이 값이 **production**인 태그를 가진 인스턴스에 대한 설명을 제공합니다.

```
aws ec2 describe-instances \
--filters Name=tag-value,Values=production
```

Example 예제: 지정된 태그를 가진 모든 EC2 리소스에 대한 설명 제공

다음 명령은 **Stack=Test** 태그가 있는 모든 EC2 리소스에 대해 설명합니다.

```
aws ec2 describe-tags \
--filters Name=key,Values=Stack Name=value,Values=Test
```

CloudFormation을 사용하여 리소스에 태그 추가

다음 예에서는 **AWS::EC2::Instance** 리소스에 **Stack=Production** 태그를 추가합니다.

Example 예: YAML

```
TagSpecifications:
- ResourceType: "instance"
  Tags:
    - Key: "Stack"
      Value: "Production"
```

Example 예: JSON

```
"TagSpecifications": [
{
  "ResourceType": "instance",
  "Tags": [
    {
      "Key": "Stack",
      "Value": "Production"
    }
  ]
}]
```

Amazon EC2 서비스 할당량

Amazon EC2는 사용자가 사용할 수 있는 서로 다른 리소스를 제공합니다. 이러한 리소스로는 이미지, 인스턴스, 볼륨 및 스냅샷이 있습니다. AWS 계정을 생성하면 이러한 리소스에 대한 기본 할당량(제한이라고도 함)이 리전별로 설정됩니다. 예를 들어 한 리전에서 시작할 수 있는 최대 인스턴스 수가 있습니다. 즉, 미국 서부(오레곤) 리전에서 인스턴스를 시작할 때 해당 요청에 의해 해당 리전의 최대 인스턴스 수가 초과되어서는 안 됩니다.

Amazon EC2 콘솔은 Amazon EC2 및 Amazon VPC 콘솔이 관리하는 리소스의 제한 정보를 제공합니다. 사용자는 이러한 제한을 높이도록 요청할 수 있습니다. 제공되는 제한 정보를 사용하여 AWS 인프라를 관리하십시오. 실제로 필요할 시점보다 미리 제한 증가를 요청하도록 계획하십시오.

자세한 내용은 Amazon Web Services 일반 참조의 [Amazon EC2 엔드포인트 및 할당량](#)을 참조하십시오.

현재 제한 조회

Amazon EC2 콘솔의 제한(Limits) 페이지에서는 리전별로 Amazon EC2 및 Amazon VPC에서 제공하는 리소스의 현재 제한을 볼 수 있습니다.

현재 제한을 조회하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

-
2. 탐색 모음에서 리전을 선택합니다.



Africa (Cape Town) af-south-1

Asia Pacific (Hong Kong) ap-east-1

Asia Pacific (Mumbai) ap-south-1

Asia Pacific (Osaka-Local) ap-northeast-3

Asia Pacific (Seoul) ap-northeast-2

Asia Pacific (Singapore) ap-southeast-1

Asia Pacific (Sydney) ap-southeast-2

Asia Pacific (Tokyo) ap-northeast-1

Canada (Central) ca-central-1

Europe (Frankfurt) eu-central-1

Europe (Ireland) eu-west-1

Europe (London) eu-west-2

Europe (Milan) eu-south-1

Europe (Paris) eu-west-3

Europe (Stockholm) eu-north-1

Middle East (Bahrain) me-south-1

South America (São Paulo) sa-east-1

3. 탐색 창에서 제한을 선택합니다.

4. 목록에서 리소스를 찾습니다. 검색 필드를 사용하여 리소스 이름이나 리소스 그룹으로 목록을 필터링할 수 있습니다. 현재 제한 열에는 해당 리소스에 대한 계정의 현재 최대값이 표시됩니다.

증가 요청

Amazon EC2 콘솔의 제한(Limits) 페이지에서 Amazon EC2 또는 Amazon VPC에서 리전별로 제공하는 리소스의 증가를 요청할 수 있습니다.

또는 Service Quotas를 사용하여 증가를 요청하십시오. 자세한 내용은 Service Quotas 사용 설명서의 [할당량 증가 요청](#)을 참조하세요.

Amazon EC2 콘솔을 사용하여 증가를 요청하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 리전을 선택합니다.
3. 탐색 창에서 제한을 선택합니다.
4. 목록에서 리소스를 선택하고 사용량 제한 증가 요청을 선택합니다.
5. 제한 증가 양식에서 필수 필드를 기재합니다. 응답은 사용자가 지정한 연락 수단을 통해 제공됩니다.

포트 25를 사용하여 전송되는 이메일 관련 제한

Amazon EC2는 기본적으로 모든 인스턴스에 대해 포트 25에서 트래픽을 제한합니다. 이 제한을 제거하도록 요청할 수 있습니다. 자세한 내용은 AWS Knowledge Center의 [EC2 인스턴스에서 포트 25의 제한을 제거할 수 있습니까?](#)를 참조하십시오.

Amazon EC2 사용 보고서

AWS는 EC2 인스턴스의 비용 및 사용, 예약 인스턴스의 사용을 분석할 수 있는 AWS Cost Explorer라는 무료 보고 도구를 제공합니다. 지난 13개월까지의 데이터를 볼 수 있고 향후 3개월 동안의 지출을 예상해볼 수 있습니다. Cost Explorer를 사용하면 시간 경과에 따라 AWS 리소스에 지출하는 금액의 패턴을 보고, 추가 질의가 필요한 영역을 식별하며, 비용을 이해하는 데 사용할 수 있는 추세를 알아볼 수 있습니다. 또한 데이터의 시간 범위를 지정하고 일별 또는 월별 시간 데이터를 볼 수도 있습니다.

Cost Explorer를 사용할 경우 사용자가 답변을 할 수 있는 질문 중 일부 예는 다음과 같습니다.

- 각 인스턴스 유형의 인스턴스에 얼마나 비용을 지출하고 있습니까?
- 특정 부서가 인스턴스를 사용하고 있는 시간은 어느 정도입니까?
- 가용 영역에 걸쳐 사용자의 인스턴스 사용이 어떻게 분포되어 있습니까?
- AWS 계정에 걸쳐 사용자의 인스턴스 사용이 어떻게 분포되어 있습니까?
- 예약 인스턴스를 얼마나 잘 사용하고 있습니까?
- 예약 인스턴스를 통해 비용을 절약하고 있습니까?

보고서 저장을 포함하여 Cost Explorer에서 보고서 작업에 대한 자세한 내용은 [Cost Explorer로 비용 분석](#)을 참조하세요.

Windows Server를 실행하는 Amazon EC2 인스턴스에 대한 자습서

다음 자습서는 Windows Server를 실행하는 EC2 인스턴스를 사용하는 일반 작업을 수행하는 방법을 설명합니다.

자습서

- [자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WordPress 블로그 배포 \(p. 1157\)](#)
- [자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WAMP 서버 설치 \(p. 1161\)](#)
- [자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WIMP 서버 설치 \(p. 1163\)](#)
- [자습서: Amazon EC2에서 애플리케이션의 가용성 향상 \(p. 1167\)](#)
- [자습서: Amazon EC2에 Windows HPC 클러스터 설치 \(p. 1170\)](#)

자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WordPress 블로그 배포

이 자습서를 참조하여 Windows Server를 실행하는 Amazon EC2 인스턴스에 WordPress 블로그를 설치하고 배포할 수 있습니다.

Linux 인스턴스에서 WordPress 블로그를 호스팅하려면 Linux 인스턴스용 Amazon EC2 사용 설명서의 [자습서: Amazon EC2를 통한 WordPress 블로그 호스팅](#)을 참조하십시오. 데이터베이스가 분리된 고가용성 솔루션이 필요하다면 AWS Elastic Beanstalk 개발자 안내서의 [고가용성 WordPress 웹 사이트 배포](#)를 참조하십시오.

사전 조건

시작하기 전에 다음을 수행해야 합니다.

- Windows Server AMI에서 Amazon EC2 인스턴스를 실행합니다. 자세한 내용은 [자습서: Amazon EC2 Windows 인스턴스 시작하기 \(p. 16\)](#) 단원을 참조하십시오.
- AWS 프리 티어(자격을 갖춘 경우)를 사용하면 무료 Windows t2.micro 인스턴스를 시작하여 12개월 동안 사용할 수 있습니다. AWS 프리 티어를 사용하여 새 애플리케이션을 시작하거나, 기존 애플리케이션을 테스트하거나, AWS를 체험해 볼 수 있습니다. 자격 및 주요 기능에 대한 자세한 내용은 [AWS 프리 티어 제품 페이지](#)를 참조하십시오.

Important

일반 인스턴스를 시작한 후 해당 인스턴스를 사용하여 WordPress 웹 사이트를 배포할 경우 종료하기 전까지 인스턴스에 대해 스탠다드 Amazon EC2 사용 요금이 청구됩니다. Amazon EC2 사용 요금에 대한 자세한 내용은 [Amazon EC2 제품 페이지](#)를 참조하십시오.

- 인스턴스를 시작하려는 보안 그룹에 인바운드 트래픽을 위한 80(HTTP), 443(HTTPS) 및 3389(RDP) 포트가 열려 있는지 확인합니다. 80 및 443 포트는 인스턴스 외부의 컴퓨터에서 HTTP 및 HTTPS에 연결하는데 사용됩니다. 따라서 이러한 포트가 열려 있지 않을 경우 인스턴스 외부에서 WordPress 사이트에 액세스할 수 없습니다. 포트 3389를 통해 원격 데스크톱 프로토콜로 인스턴스에 연결할 수 있습니다.

- 인스턴스에 연결합니다.

Microsoft 웹 플랫폼 설치 관리자 설치

Microsoft 웹 플랫폼 설치 관리자를 사용하여 서버에 WordPress를 설치하고 구성할 수 있습니다. 이 도구를 사용하면 웹 애플리케이션과 웹 사이트를 IIS 서버에 간편하게 배포할 수 있습니다. 자세한 내용은 [Microsoft 웹 플랫폼 설치 관리자](#) 단원을 참조하십시오.

Microsoft 웹 플랫폼 설치 관리자를 설치하는 방법

1. [사전 조건](#) (p. 1157)의 조건을 모두 충족하는지 확인합니다.
2. 인스턴스에 연결합니다.
3. 웹에서 필요한 소프트웨어를 다운로드하여 설치할 수 있게 Internet Explorer 보안 강화 구성을 비활성화 합니다.
 - a. 서버 관리자를 엽니다.
 - Windows Server 2008 R2의 서버 요약 아래에 있는 보안 정보 섹션에서 IE ESC 구성을 클릭합니다.
 - Windows Server 2012 R2에서 왼쪽 창에 있는 로컬 서버를 클릭합니다. 속성 창에서 IE 보안 강화 구성을 찾습니다. 사용을 클릭합니다.
 - b. 관리자 아래에서 사용 안 함을 클릭하고 확인을 클릭합니다.
 - c. 서버 관리자를 닫습니다.
 - d. 웹에서 소프트웨어 설치를 마치면 Internet Explorer 보안 강화 구성을 다시 활성화합니다.
4. 최신 버전의 [Microsoft 웹 플랫폼 설치 관리자](#)를 다운로드하여 설치합니다.

WordPress 설치

이제 웹 플랫폼 설치 프로그램을 사용하여 서버에 WordPress를 배포할 수 있습니다.

WordPress를 설치하려면

1. Visual Studio 2012 업데이트 4 이상용 Visual C++ 재배포 가능 패키지를 [다운로드](#)하여 설치합니다.

Important

64비트에서는 PHP가 실행되지 않으므로 사용자의 운영 체제가 64비트 운영 체제일지라도 32비트 버전을 설치해야 합니다.

웹 플랫폼 관리자를 연 후 애플리케이션을 클릭합니다.

2. WordPress를 선택하고 추가를 클릭한 후 설치를 클릭합니다.
3. 필수 구성 요소 페이지에서 사용할 데이터베이스에 대한 MySQL을 선택합니다. 암호 및 암호 다시 입력 상자에 MySQL 데이터베이스에 사용할 관리자 암호를 입력한 다음 계속을 클릭합니다.

보안 암호 생성에 대한 자세한 내용은 <https://identitysafe.norton.com/password-generator/> 단원을 참조하십시오. 기존 암호를 재사용하지 마십시오. 새로 설정한 암호는 안전한 장소에 보관하십시오.

4. 타사 애플리케이션 소프트웨어, Microsoft 제품(IIS 서버 포함) 및 구성 요소 목록을 보려면 동의함을 클릭합니다. 웹 플랫폼 설치 관리자에서 소프트웨어를 설치한 후 새 사이트를 구성하라는 메시지가 나타납니다.
5. 구성 페이지의 'WordPress 애플리케이션 이름' 상자에서 기본 애플리케이션 이름을 지우고 이 상자를 비워둔 다음 다른 상자의 기본 정보는 그대로 둔 상태에서 계속을 클릭합니다.
6. 예를 클릭하여 폴더 내용을 덮어쓰도록 수락합니다.

보안 키 구성

WordPress를 사용하여 사이트에 고유한 인증 키와 솔트를 생성하고 입력할 수 있습니다. 이러한 키 및 솔트 값은 WordPress 사용자가 로컬 컴퓨터에 저장하는 브라우저 쿠키에 암호 계층을 제공합니다. 기본적으로 긴 무작위 값을 추가해서 사이트의 보안성을 강화할 수 있습니다.

보안 키에 대한 자세한 내용은 http://codex.wordpress.org/Editing_wp-config.php#Security_Keys 단원을 참조하십시오.

보안 키를 구성하려면

1. <https://api.wordpress.org/secret-key/1.1/salt/>를 방문해서 키 값의 세트를 무작위로 생성하고 이를 복사하여 설치 마법사에 붙여 넣을 수 있습니다. 다음 단계에서는 Windows 설치와 작동하도록 메모장에서 이러한 값을 수정하는 방법을 보여 줍니다.
2. 페이지의 모든 텍스트를 클립보드에 복사합니다. 예를 들면 아래와 같습니다.

Note

아래 값은 예시 목적만을 위한 것입니다. 설치 시 이 값을 사용하지는 마십시오.

```
define('AUTH_KEY',           '3#U$$+[RXN8:b^-L_0(WU_+ c+WFKI~c]o]-bHw+/'
Aj[wTwSiZ<Qb[mghEXcRh-]');
define('SECURE_AUTH_KEY',    'Zsz._P=l/|y.Lq)Xjlkws1y5NJ76E6EJ.AV0pCKZZB,*-*r ?6OP$eJT@;
+(ndLg');
define('LOGGED_IN_KEY',      'ju)qwre3V*+8f_zOWf?{LlGsQ]Ye@2Jh^,8x>)Y |;([Iw]Pi+LG#A4R?
7N`YB3');
define('NONCE_KEY',          'P(g62HeZxEes/LnI^i=H,[XwK9I&[2s :?ON)VJM%?;v2v]v+;
+^9eXuah@::Cj');
define('AUTH_SALT',          'C$DpB4Hj[JK:{ql`sRVa:{:7yShy(9A@5wg`JJVb1fk%_-Bx*M4(qc[Qg
%JT!h');
define('SECURE_AUTH_SALT',   'd!uRu#)+q#{f$Z?Z9uFPG.${+S{n~1M&%@-gL>U>NV<zpD-@2-Es7Q1O-
bp28EKv');
define('LOGGED_IN_SALT',     ':j{00P*owZf)kVD+FVLn-->. |Y%Ug4#I^*LVd9QeZ^&XmK/e(76miC+&W&
+^OP/');
define('NONCE_SALT',         '-97r*V/cgxLmp?Zy4zUU4r99QQ_rGs2LTd%P; /
_e1tS)8_B/, .6[=UK<J_y9?JWG');
```

3. 시작, 모든 프로그램, 보조프로그램, 메모장을 차례로 클릭하여 메모장 창을 엽니다.
4. 복사한 텍스트를 메모장 창에 붙여 넣습니다.
5. Windows WordPress 설치에서는 키 및 솔트 값에 달러 기호(\$)를 사용할 수 없으므로, 다른 문자(예: S)로 대체해야 합니다. 메모장 창에서 편집을 클릭한 후 바꾸기를 클릭합니다.
6. 찾을 내용 상자에 \$를 입력합니다.
7. 바꿀 내용 상자에 s를 입력합니다.
8. 모두 바꾸기를 클릭하여 모든 달러 기호를 s 문자로 바꿉니다.
9. 바꾸기 창을 닫습니다.
10. 메모장 창에서 수정한 키 및 솔트 값을 설치 마법사의 해당 상자에 붙여 넣습니다. 예를 들어, 메모장 창의 AUTH_KEY 값을 마법사의 인증 키 상자에 붙여 넣어야 합니다.

아래 예에 표시된 것처럼 값을 둘러싸고 있는 작은 따옴표 또는 다른 텍스트는 제외하고 실제 값만 포함합니다.

메모장 창에서 수정된 AUTH_KEY 줄:

```
define('AUTH_KEY',           '3#USS+[RXN8:b^-L_0(WU_+ c+WFKI~c]o]-bHw+/'
Aj[wTwSiZ<Qb[mghEXcRh-');
```

이 텍스트를 마법사의 인증 키 상자에 붙여 넣습니다.

3#USS+[RXN8:b^-L_0(WU_+ c+WFKI~c]o]-bHw+)/Aj[wTwSiZ<Qb[mghEXcRh-

- 계속 및 마침을 클릭하여 웹 플랫폼 설치 관리자 마법사를 완료합니다.

사이트 제목 및 관리자 구성

웹 플랫폼 설치 관리자 마법사를 완료하면 <http://localhost/wp-admin/install.php>의 WordPress 설치에 대한 브라우저 창이 열립니다. 이 페이지에서 사이트의 제목과 관리 사용자를 구성하여 블로그를 조정합니다.

설치를 완료하려면

- WordPress 환영합니다 페이지에 다음 정보를 입력하고 WordPress 설치하기를 클릭합니다.

필드	값
사이트 제목	WordPress 사이트의 이름을 입력합니다.
사용자 이름	WordPress 관리자의 이름을 입력합니다. 보안을 위해 사용자의 기본 사용자 이름 admin보다 도용하기 더 어려운 고유 이름을 선택해야 합니다.
암호	보안성이 강력한 암호를 입력하고 이를 한 번 더 입력해서 확인합니다. 기존 암호를 재사용하지 마십시오. 새로 설정한 암호는 안전한 장소에 보관하십시오.
이메일 주소	알림에 사용할 이메일 주소를 입력합니다.
프라이버시	검색 엔진에서 사이트를 인덱싱하도록 허용하려면 선택합니다.

- 로그인을 클릭합니다.
- 로그인 페이지에서 사용자 이름에 사용자 이름을 입력하고 비밀번호에 이전에 입력한 사이트 암호를 입력합니다.

WordPress 사이트를 퍼블릭으로 만들기

로컬 호스트에서 WordPress 블로그를 볼 수 있으므로 이제 다른 사용자가 이 웹 사이트를 볼 수 있도록 이 웹 사이트를 인스턴스의 기본 사이트로 게시할 수 있습니다. 다음 절차에서는 로컬 호스트 대신 인스턴스의 퍼블릭 DNS 이름을 가리키도록 WordPress 설정을 수정하는 프로세스를 안내합니다.

WordPress 사이트의 기본 설정을 구성하려면

- 인스턴스에서 브라우저를 연 다음 <http://localhost/wp-admin>으로 이동하여 WordPress 대시보드를 엽니다. 자격 증명을 묻는 메시지가 표시되면 사용자 이름에 사용자 이름을 입력하고 비밀번호에 사이트 암호를 입력합니다.
- 대시보드 창에서 설정을 클릭합니다.
- 일반 설정 페이지에 다음 정보를 입력하고 변경 사항 저장을 클릭합니다.
 - WordPress 주소(URL) - 인스턴스의 퍼블릭 DNS 주소입니다. 예를 들어, URL은 <http://ec2-203-0-113-25.compute-1.amazonaws.com>와 비슷합니다.

Amazon EC2 콘솔을 사용하여 인스턴스에 대한 퍼블릭 DNS를 얻을 수 있습니다. 이렇게 하려면 인스턴스를 선택하고 퍼블릭 DNS 열을 확인합니다. 이 열이 숨겨져 있는 경우 표시/숨기기 아이콘을 클릭하고 퍼블릭 DNS를 선택합니다.

- 사이트 주소(URL) - 인스턴스에 대해 WordPress 주소(URL)에서 설정한 것과 동일한 퍼블릭 DNS 주소입니다.
4. 새 사이트를 보려면 WordPress를 호스팅 중인 인스턴스와 다른 컴퓨터에서 브라우저를 열고 웹 주소 필드에 인스턴스의 퍼블릭 DNS 주소를 입력합니다. WordPress 사이트가 나타납니다.

축하합니다! 이제 Windows 인스턴스에서 WordPress 사이트를 배포했습니다.

다음 단계

이 인스턴스가 더 이상 필요하지 않은 경우 요금이 반복적으로 부과되지 않도록 인스턴스를 제거할 수 있습니다. 자세한 내용은 [인스턴스 정리 \(p. 19\)](#)를 참조하십시오.

운영하는 WordPress 블로그가 유명해지고 그에 따라 보다 많은 컴퓨팅 파워 또는 스토리지가 필요하게 될 경우 다음 단계를 고려하십시오.

- 인스턴스에서 스토리지 공간을 확장합니다. 자세한 내용은 [Amazon EBS 탄력적 볼륨 \(p. 1023\)](#) 단원을 참조하십시오.
- MySQL 데이터베이스를 [Amazon RDS](#)로 이동하여 이 서비스의 자동 확장 기능을 이용합니다.
- 더 큰 인스턴스 유형으로 마이그레이션합니다. 자세한 내용은 [인스턴스 유형 변경 \(p. 194\)](#) 단원을 참조하십시오.
- 인스턴스를 더 추가합니다. 자세한 내용은 [자습서: Amazon EC2에서 애플리케이션의 가용성 향상 \(p. 1167\)](#) 단원을 참조하십시오.

WordPress에 대한 자세한 내용은 <http://codex.wordpress.org>에서 WordPress Codex 도움 문서를 참조하십시오. 설치 문제 해결에 대한 자세한 내용은 <https://wordpress.org/support/article/how-to-install-wordpress/#common-installation-problems>를 참조하십시오. WordPress 블로그 보안 강화에 대한 자세한 내용은 <https://wordpress.org/support/article/hardening-wordpress/>를 참조하십시오. 최신 상태로 WordPress 블로그 유지에 대한 자세한 내용은 <https://wordpress.org/support/article/updating-wordpress/>를 참조하십시오.

자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WAMP 서버 설치

이 자습서에서는 Windows Server를 실행하는 EC2 인스턴스에 PHP 및 MySQL과 함께 Apache 웹 서버를 설치하는 방법을 보여 줍니다. 이 소프트웨어 구성을 WAMP 서버 또는 WAMP 스택(Windows, Apache, MySQL, PHP)이라고도 합니다. Linux에서 유사한 서버를 만드는 방법은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [자습서: LAMP 웹 서버 설치](#) 단원을 참조하십시오.

WAMP 스택은 설치를 쉽게 함으로써 개발자들이 즉시 실행할 수 있도록 설계되었습니다. 다음과 같은 이유로 인해 프로덕션 환경에서 사용하는 것은 적합하지 않습니다.

- 기본 구성이 대부분의 프로덕션 환경을 위한 보안 요구를 충족하지 않습니다.
- 단일 프로덕션 서버의 서로 다른 소프트웨어 구성 요소를 업그레이드하고 패치하는 작업이 서버 가용성에 영향을 줍니다.
- WAMP 원클릭 설치 관리자는 파일을 표준 위치에 배치하지 않는데, 이는 중요한 구성 파일들을 찾기 어렵게 만들 수 있습니다.

그러나 EC2 인스턴스에 WAMP 스택을 만들어서 제어된 테스트 환경에서 웹 프로젝트를 시제품화할 수 있습니다. 예를 들어, 고정 웹 사이트를 호스팅하거나 데이터베이스에서 정보를 읽고 쓰는 동적 PHP 애플리케이션을 배포할 수 있습니다.

WAMP 스택을 설치하는 데 사용할 수 있는 여러 타사 솔루션이 있으며, 본 자습서에서는 Bitnami WAMP 스택을 사용합니다. 자세한 내용은 [검토: 웹 개발자를 위한 WAMP 스택 단원](#)을 참조하십시오.

사전 조건

- Windows Server 2008 R2 또는 2012 R2 기본 인스턴스를 프로비저닝하십시오. 인터넷에서 연결할 수 있는 퍼블릭 도메인 이름 시스템(DNS) 이름으로 기본 인스턴스를 구성해야 합니다. 자세한 내용은 [자습서: Amazon EC2 Windows 인스턴스 시작하기 \(p. 16\)](#) 단원을 참조하십시오.
- 인스턴스에 대한 보안 그룹에서 다음 포트를 열어 놓았는지 확인합니다.
 - 포트 80(HTTP 인바운드 및 아웃바운드) - 인스턴스 밖의 컴퓨터에서 HTTP를 사용하여 연결할 수 있습니다.
 - 포트 443(HTTPS 인바운드 및 아웃바운드) - 인스턴스 밖의 컴퓨터에서 HTTPS를 사용하여 연결할 수 있습니다.
 - 포트 3389(RDP 인바운드만) - RDP(원격 데스크톱 프로토콜)를 사용하여 인스턴스에 연결할 수 있습니다. 보안을 위해 RDP 액세스를 조직의 IP 주소 범위로 제한하십시오.

WAMP 서버를 설치하려면

1. Microsoft 원격 데스크톱을 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
2. 웹에서 필요한 소프트웨어를 다운로드하여 설치할 수 있게 Internet Explorer 보안 강화 구성을 비활성화 합니다.
 - a. 인스턴스에서 서버 관리자를 엽니다.
 - b. [Windows Server 2008 R2] 서버 요약 아래에 있는 보안 정보에서 IE ESC 구성을 클릭합니다.
[Windows Server 2012 R2] 왼쪽 창에 있는 로컬 서버를 클릭합니다. 속성 창에서 IE 보안 강화 구성을 찾습니다. 사용을 클릭합니다.
 - c. 관리자 아래에서 사용 안 함을 클릭하고 확인을 클릭합니다.
 - d. 서버 관리자를 닫습니다.
 - e. 웹에서 소프트웨어 설치를 마치면 Internet Explorer 보안 강화 구성을 다시 활성화합니다.
3. 인스턴스가 최신 보안 업데이트와 버그 수정을 사용하도록 소프트웨어 업데이트를 설치합니다.
 - EC2Config - 최신 버전의 EC2Config 서비스를 [다운로드](#)하여 설치합니다. 자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 단원을 참조하십시오.
 - Windows 업데이트 - 인스턴스에 최신 보안 및 소프트웨어 업데이트가 설치되도록 Windows 업데이트를 실행합니다. 제어판에서 시스템 및 보안을 클릭합니다. Windows 업데이트 섹션에서 업데이트 확인을 클릭합니다.
4. WAMP 스택을 다운로드하여 설치합니다. 본 자습서에서는 [이 WAMP 스택](#)을 다운로드하여 설치하는 것이 좋지만 다른 Bitnami WAMP 스택을 다운로드하여 설치할 수도 있습니다. 설치하는 스택에 관계없이 소셜 미디어 계정을 사용하여 로그인하거나 무료 Bitnami 계정을 만들라는 메시지가 나타납니다. 로그인 후 Bitnami 설치 마법사를 실행합니다.
5. 설치가 완료되면 테스트 페이지를 검색하여 Apache 웹 서버가 제대로 구성되어 있고 실행되는지 확인하십시오. 다른 컴퓨터에서 웹 브라우저를 열고 WAMP 서버의 퍼블릭 DNS 주소 또는 퍼블릭 IP 주소를 입력합니다. 인스턴스의 퍼블릭 DNS 주소는 Amazon EC2 콘솔의 퍼블릭 DNS 열에 표시됩니다. 이 열이 숨겨져 있으면 표시/숨기기 아이콘을 클릭하고 퍼블릭 DNS를 선택합니다.

Important

Bitnami 테스트 페이지가 보이지 않는 경우 [고급 보안이 설정된 Windows 방화벽]을 사용하여 포트 80을 통한 HTTP 프로토콜과 포트 443을 통한 HTTPS 프로토콜을 허용하는 사용자 지정

규칙을 만듭니다. 자세한 내용은 Microsoft TechNet의 [네트워크 보안](#)을 참조하십시오. 또한 해당 인스턴스의 보안 그룹에 HTTP(포트 80) 연결을 허용하는 규칙이 있는지 확인하십시오. 자세한 내용은 [보안 그룹에 규칙 추가 \(p. 914\)](#) 단원을 참조하십시오.

6. 웹에서 PHP 파일을 보는 방식으로 WAMP 서버를 테스트합니다. 다음 단계를 수행하려면 인스턴스에 관리자로 로그인해야 합니다.
 - a. 아래의 코드를 포함하는 `phpinfo.php`라는 파일을 생성하고 이 파일을 Apache 루트 디렉터리에 넣습니다. 다음은 기본 경로입니다. `C:\Bitnami\wampstack-<version_number>\apache2\htdocs`

```
<?php phpinfo(); ?>
```
 - b. 웹 브라우저에서는 방금 생성한 파일의 URL을 입력합니다. 이 URL은 인스턴스의 퍼블릭 DNS 주소에 슬래시(/)와 파일 이름이 추가된 형태입니다. 예: `http://my.public.dns.amazonaws.com/phpinfo.php`.
 - c. PHP 정보 페이지가 표시되는지 확인합니다. 페이지가 표시되지 않는 경우 올바른 퍼블릭 DNS 주소를 입력했는지 확인하십시오. 또한 Windows 폴더 옵션이 알려진 파일 확장명을 표시하도록 구성되어 있는지 확인하십시오. 기본적으로 폴더 옵션은 알려진 파일 확장명을 숨깁니다. 메모장에서 파일을 생성하고 루트 디렉터리에 저장한 경우에는 `phpinfo.php` 파일이 `phpinfo.php.txt`로 잘못 저장될 수 있습니다.
 - d. 보안을 위해 WAMP 서버 테스트를 마치면 `phpinfo.php` 파일을 삭제하십시오.
7. 기본 기능을 비활성화하고 루트 암호를 설정하여 MySQL 보안을 강화하십시오. `mysql_secure_installation` Perl 스크립트가 이러한 작업을 자동으로 수행할 수 있습니다. 스크립트를 실행하려면 Perl을 설치해야 합니다.
 - a. [Perl 프로그래밍 언어](#) 웹 사이트에서 Perl을 다운로드하여 설치하십시오.
 - b. `C:\Bitnami\wampstack-<version_number>\mysql\bin` 디렉터리에서 `mysql_secure_installation`을 두 번 클릭합니다.
 - c. 메시지가 나타나면 Bitnami WAMP 스택 설치 관리자를 실행할 때 입력한 MySQL 루트 계정 암호를 입력하고 Enter 키를 누릅니다.
 - d. **n**을 입력하여 암호 변경을 건너뜁니다.
 - e. **y**를 눌러서 익명 사용자 계정을 제거합니다.
 - f. **y**를 입력하여 원격 루트 로그인을 비활성화합니다.
 - g. **y**를 눌러서 테스트 데이터베이스를 제거합니다.
 - h. **y**를 눌러서 권한 테이블을 다시 로드하고 변경사항을 저장합니다.

이 자습서의 단계를 성공적으로 완료한 경우 WAMP 서버가 제대로 작동하고 있는 것입니다. 테스트를 계속 하려면 `C:\Bitnami\wampstack-<version_number>\apache2\htdocs` 폴더에 내용을 추가하고 인스턴스에 대한 퍼블릭 DNS 주소를 사용하여 해당 내용을 볼 수 있습니다.

Important

지금 바로 사용할 계획이 없는 경우 MySQL 서버를 종지하는 것이 좋습니다. 필요할 때 서버를 다시 시작할 수 있습니다.

자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WIMP 서버 설치

이 자습서에서는 Windows Server를 실행하는 EC2 인스턴스에 PHP 및 MySQL과 함께 Microsoft 인터넷 정보 서비스(IIS) 웹 서버를 설치하는 방법을 보여 줍니다. 이 소프트웨어 구성을 WIMP 서버 또는 WIMP 스택(Windows, IIS, MySQL, PHP)이라고도 합니다.

WIMP 스택은 설치를 쉽게 함으로써 개발자들이 즉시 실행할 수 있도록 설계되었습니다. 다음과 같은 이유로 인해 프로덕션 환경에서 사용하는 것은 적합하지 않습니다.

- 기본 구성이 대부분의 프로덕션 환경을 위한 보안 요건을 충족하지 않습니다.
- 단일 프로덕션 서버의 서로 다른 소프트웨어 구성 요소를 업그레이드하고 패치하는 작업이 서버 가용성에 영향을 줍니다.
- WAMP 원클릭 설치 관리자는 파일을 표준 위치에 배치하지 않는데, 이는 중요한 구성 파일들을 찾기 어렵게 만들 수 있습니다.

그러나 EC2 인스턴스에 WIMP 스택을 만들어서 제어된 테스트 환경에서 웹 프로젝트를 시제품화 할 수 있습니다. 예를 들어, 고정 웹 사이트를 호스팅하거나 데이터베이스에서 정보를 읽고 쓰는 동적 PHP 애플리케이션을 배포할 수 있습니다.

사전 조건

- Windows Server 2008 R2 또는 2012 R2 기본 인스턴스를 프로비저닝하십시오. 인터넷에서 연결할 수 있는 퍼블릭 도메인 이름 시스템(DNS) 이름으로 기본 인스턴스를 구성해야 합니다. 자세한 내용은 [자습서: Amazon EC2 Windows 인스턴스 시작하기 \(p. 16\)](#) 단원을 참조하십시오.
- 인스턴스에 대한 보안 그룹에서 다음 포트를 열어 놓았는지 확인합니다.
 - 포트 80(HTTP 인바운드 및 아웃바운드) - 인스턴스 밖의 컴퓨터에서 HTTP를 사용하여 연결할 수 있습니다.
 - 포트 443(HTTPS 인바운드 및 아웃바운드) - 인스턴스 밖의 컴퓨터에서 HTTPS를 사용하여 연결할 수 있습니다.
 - 포트 3389(RDP 인바운드만) - RDP(원격 데스크톱 프로토콜)를 사용하여 인스턴스에 연결할 수 있습니다. 보안을 위해 RDP 액세스를 조직의 IP 주소 범위로 제한하십시오.
- Read the best practices for installing PHP on the [Microsoft 웹 플랫폼](#)에 PHP를 설치하는 모범 사례를 참조하십시오.

인스턴스 준비

인스턴스를 준비하려면

1. Microsoft 원격 데스크톱을 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
2. 웹에서 필요한 소프트웨어를 다운로드하여 설치할 수 있게 Internet Explorer 보안 강화 구성을 비활성화 합니다.
 - a. 인스턴스에서 서버 관리자를 엽니다.
 - b. [Windows Server 2008 R2] 서버 요약 아래에 있는 보안 정보에서 IE ESC 구성을 선택합니다.
[Windows Server 2012 R2] 왼쪽 창에 있는 로컬 서버를 선택합니다. 속성 창에서 IE 보안 강화 구성을 찾습니다. 사용을 선택합니다.
 - c. 관리자에서 사용 안 함을 선택하고 확인을 선택합니다.
 - d. 서버 관리자를 닫습니다.

Note

웹에서 소프트웨어 설치를 마치면 Internet Explorer 보안 강화 구성을 다시 활성화합니다.

3. 인스턴스가 최신 보안 업데이트와 버그 수정을 사용하도록 소프트웨어 업데이트를 설치합니다.
 - a. EC2Config - 최신 버전의 EC2Config 서비스를 [다운로드](#)하여 설치합니다. 이 서비스를 설치하는 방법에 대한 자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 단원을 참조하십시오.

- b. Windows 업데이트 - 인스턴스에 최신 보안 및 소프트웨어 업데이트가 설치되도록 Windows 업데이트를 실행합니다. 제어판에서 시스템 및 보안을 선택합니다. Windows 업데이트 섹션에서 업데이트 확인을 선택합니다.

IIS 웹 서버 설치

IIS는 Windows Server의 기능으로, 서버 관리자를 통해 설치됩니다. 사용하게 될 절차는 인스턴스에서 실행 중인 Windows Server 버전에 따라 다릅니다.

Windows Server 2012에 IIS 설치

1. Windows 인스턴스에서 시작, 서버 관리자를 선택한 다음 역할 및 기능 추가를 선택합니다.
2. 시작하기 전에 페이지에서 다음을 선택합니다.
3. [Select installation type] 페이지에서 [Role-based or feature-based installation]을 선택하고 [Next]를 선택합니다.
4. Select destination server(대상 서버 선택) 페이지에서 서버 폴에 있는 인스턴스를 선택하고 다음을 선택합니다.
5. Select server roles(서버 역할 선택) 페이지에서 웹 서버(IIS)를 선택하고 Add features(기능 추가)를 선택한 후 다음을 선택합니다.
6. Select features(기능 선택) 페이지에서 기본 기능을 유지하고 .NET Framework 4.5 Features(.NET Framework 4.5 기능)를 확장한 후 ASP.NET 4.5를 선택하고 다음을 선택합니다.
7. [Web Server Role (IIS)] 페이지에서 [Next]를 선택합니다.
8. Select role services(역할 서비스 선택) 페이지에서 기본 서비스를 유지하고 애플리케이션 개발을 선택합니다.
9. 애플리케이션 개발을 확장하고 다음 기능을 선택합니다. 이러한 기능을 선택할 때 메시지가 표시되면 Add Features(기능 추가)를 선택합니다.
 - a. .NET Extensibility 3.5
 - b. .NET Extensibility 4.5
 - c. 애플리케이션 초기화
 - d. ASP.NET 3.5
 - e. ASP.NET 4.5
 - f. CGI
10. [Next]를 선택합니다.
11. Confirm installation selections(설치 선택 확인) 페이지에서 Restart the destination server automatically if required(필요 시 대상 서버를 자동으로 다시 시작)를 선택합니다. 확인 메시지가 표시되면 [Yes]를 선택합니다.
12. 설치를 선택하고 설치가 완료되면 닫기를 선택합니다.
13. Windows 업데이트를 다시 실행합니다.

Windows Server 2008에 IIS 설치

1. Windows 인스턴스에서 시작, 서버 관리자를 선택한 다음 역할을 선택합니다.
2. 역할 추가를 선택합니다.
3. 시작하기 전에 페이지에서 다음을 선택합니다.
4. Select Server Roles(서버 역할 선택) 페이지에서 웹 서버(IIS)를 선택합니다.
5. Select Role Services(역할 서비스 선택) 페이지의 애플리케이션 개발 아래에서 ASP.NET을 선택합니다.

- a. 메시지가 나타나면 Add Required Role Services(필수 역할 서비스 추가)를 선택합니다.
- b. CGI를 선택합니다.
- c. [Next]를 선택합니다.
6. Confirm Installation Selections(설치 설정 확인)에서 설치를 선택합니다.
7. Windows 업데이트를 다시 실행합니다.

웹 서버가 실행되는지 확인하려면

설치가 완료되면 IIS 시작 페이지로 이동하여 IIS 웹 서버가 제대로 구성되어 있고 실행되는지 확인하십시오. 다른 컴퓨터에서 웹 브라우저를 열고 WIMP 서버의 퍼블릭 DNS 주소 또는 퍼블릭 IP 주소를 입력합니다. 인스턴스의 퍼블릭 DNS 주소는 Amazon EC2 콘솔의 퍼블릭 DNS 열에 표시됩니다. 이 열이 숨겨져 있으면 Show/Hide(표시/숨기기) 아이콘을 선택하고 퍼블릭 DNS를 선택합니다.

Important

Bitnami 테스트 페이지가 보이지 않는 경우 [고급 보안이 설정된 Windows 방화벽]을 사용하여 포트 80을 통한 HTTP 프로토콜과 포트 443을 통한 HTTPS 프로토콜을 허용하는 사용자 지정 규칙을 만듭니다. 자세한 내용은 Microsoft TechNet의 [네트워크 보안](#)을 참조하십시오. 또한 해당 인스턴스의 보안 그룹에 HTTP(포트 80) 연결을 허용하는 규칙이 있는지 확인하십시오. 자세한 내용은 [보안 그룹에 규칙 추가 \(p. 914\)](#) 단원을 참조하십시오.

MySQL 및 PHP 설치

Microsoft 웹 플랫폼 설치 관리자를 사용하여 MySQL 및 PHP를 다운로드하고 설치할 수 있습니다.

MySQL 및 PHP를 설치하려면

1. Windows Server 인스턴스에서 최신 버전인 [Microsoft Web Platform Installer 5.0](#)을 다운로드하여 설치합니다.
2. Microsoft 웹 플랫폼 설치 관리자에서 제품 탭을 선택합니다.
3. MySQL Windows 5.5를 선택한 다음 추가를 선택합니다.
4. PHP 5.6.0을 선택한 다음 추가를 선택합니다.
5. [Install]을 선택합니다.
6. 사전 조건 페이지에서 MySQL 기본 데이터베이스 관리자 계정에 대한 암호를 입력한 후 계속을 선택합니다.
7. 설치가 완료되면 마침을 선택하고 종료를 선택하여 웹 플랫폼 설치 관리자를 닫습니다.

서버 테스트

웹에서 PHP 파일을 보는 방식으로 서버를 테스트합니다. 다음 단계를 수행하려면 인스턴스에 관리자로 로그인해야 합니다.

WIMP 서버를 테스트하려면

1. [Visual Studio 2012 업데이트 4 x86용 Visual C++ 재배포 가능 패키지](#)를 다운로드하고 설치합니다. 서버가 64비트 서버인 경우에도 x86 패키지를 설치해야 합니다.
2. 다음 코드를 포함하는 `phpinfo.php`라는 파일을 생성하고 이 파일을 IIS 루트 디렉터리에 넣습니다. 다음은 기본 경로입니다. `C:\inetpub\wwwroot`.

```
<?php phpinfo(); ?>
```

3. 웹 브라우저에서는 방금 생성한 파일의 URL을 입력합니다. `http://my.public.dns.amazonaws.com/phpinfo.php` 예제와 같이 이 URL은 인스턴스의 퍼블릭 DNS 주소에 슬래시(/)와 파일 이름이 추가된 형태입니다.
4. PHP 정보 페이지가 표시되는지 확인합니다. 페이지가 표시되지 않는 경우 올바른 퍼블릭 DNS 주소를 입력했는지 확인하십시오. 또한 Windows 폴더 옵션이 알려진 파일 확장명을 표시하도록 구성되어 있는지 확인하십시오. 기본적으로 폴더 옵션은 알려진 파일 확장명을 숨깁니다. 메모장에서 파일을 생성하고 루트 디렉터리에 저장한 경우에는 `phpinfo.php` 파일이 `phpinfo.php.txt`로 잘못 저장될 수 있습니다.
5. 보안을 위해 WAMP 서버 테스트를 마치면 `phpinfo.php` 파일을 삭제하십시오.
6. 기본 기능을 비활성화하고 루트 암호를 설정하여 MySQL 보안을 강화하십시오.
`mysql_secure_installation` Perl 스크립트가 이러한 작업을 자동으로 수행할 수 있습니다. 스크립트를 실행하려면 Perl을 설치해야 합니다.
 - a. [Perl 프로그래밍 언어](#) 웹 사이트에서 Perl을 다운로드하여 설치하십시오.
 - b. `C:\Program Files\MySQL\MySQL Server 5.5\bin` 디렉터리에서 `mysql_secure_installation`을 두 번 클릭합니다.
 - c. 메시지가 나타나면 현재 루트 암호를 입력하고 Enter 키를 누릅니다.
 - d. `n`을 입력하여 암호 변경을 건너뜁니다.
 - e. `y`를 눌러서 익명 사용자 계정을 제거합니다.
 - f. `y`를 입력하여 원격 루트 로그인을 비활성화합니다.
 - g. `y`를 눌러서 테스트 데이터베이스를 제거합니다.
 - h. `y`를 눌러서 권한 테이블을 다시 로드하고 변경사항을 저장합니다.

이제 WIMP 웹 서버가 완전히 동작하는 상태가 됩니다. `C:\inetpub\wwwroot`의 IIS document root에 콘텐츠를 추가하면 인스턴스에 대한 퍼블릭 DNS 주소에서 해당 콘텐츠를 볼 수 있습니다.

Important

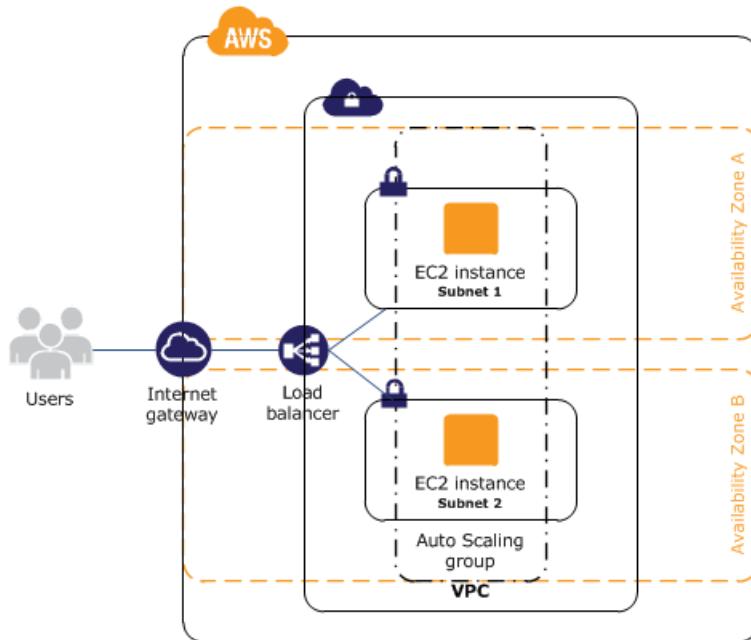
지금 바로 사용할 계획이 없는 경우 MySQL 서버를 종지하는 것이 좋습니다. 필요할 때 서버를 다시 시작할 수 있습니다.

자습서: Amazon EC2에서 애플리케이션의 가용성 향상

단일 EC2 인스턴스에서 애플리케이션 또는 웹 사이트 실행을 시작한 후 시간이 지나면서 수요를 충족하기 위해 두 개 이상의 인스턴스가 필요한 지점까지 트래픽이 증가하는 경우를 가정해 봅니다. AMI에서 여러 EC2 인스턴스를 시작한 다음 Elastic Load Balancing을 사용하여 애플리케이션에 대한 수신 트래픽을 EC2 인스턴스 간에 분산할 수 있습니다. 이렇게 하면 애플리케이션의 가용성이 향상됩니다. 인스턴스를 여러 가용 영역에 배치하면 애플리케이션의 내결함성도 향상됩니다. 가용 영역 하나가 중단되면 트래픽이 다른 가용 영역으로 라우팅됩니다.

Amazon EC2 Auto Scaling을 사용하여 애플리케이션에 대한 실행 인스턴스 수를 항상 최소한으로 유지할 수 있습니다. Amazon EC2 Auto Scaling은 인스턴스나 애플리케이션이 비정상일 때를 감지하고 자동으로 교체하여 애플리케이션의 가용성을 유지합니다. 또한 Amazon EC2 Auto Scaling을 사용하면 지정한 기준을 사용하여 필요에 따라 자동으로 Amazon EC2 용량을 확장하거나 축소할 수 있습니다.

이 자습서에서는 Amazon EC2 Auto Scaling과 Elastic Load Balancing을 사용하여 로드 밸런서 뒤에 지정된 수의 정상 EC2 인스턴스를 유지합니다. 트래픽이 로드 밸런서로 이동한 다음 인스턴스로 라우팅되므로 이러한 인스턴스에는 퍼블릭 IP 주소가 필요 없습니다. 자세한 내용은 [Amazon EC2 Auto Scaling](#) 및 [Elastic Load Balancing](#) 단원을 참조하십시오.



내용

- [사전 조건 \(p. 1168\)](#)
- [애플리케이션 확장 및 로드 밸런싱 \(p. 1168\)](#)
- [로드 밸런서 테스트 \(p. 1170\)](#)

사전 조건

이 자습서에서는 다음을 이미 완료했다고 가정합니다.

1. 두 개 이상의 가용 영역에 있는 퍼블릭 서브넷 하나로 가상 사설 클라우드(VPC)를 생성하였습니다.
2. VPC에서 인스턴스를 시작하였습니다.
3. 인스턴스에 연결하여 인스턴스를 사용자 지정하였습니다. 소프트웨어 및 애플리케이션 설치, 데이터 복사, 추가 EBS 볼륨 연결 등을 예로 들 수 있습니다. 인스턴스에서 웹 서버를 설정하는 방법에 대한 자세한 내용은 [자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WAMP 서버 설치 \(p. 1161\)](#) 섹션 또는 [자습서: Windows Server를 실행하는 Amazon EC2 인스턴스에 WIMP 서버 설치 \(p. 1163\)](#) 단원을 참조하십시오.
4. 인스턴스에서 애플리케이션을 테스트하여 인스턴스가 올바르게 구성되었는지 확인하였습니다.
5. 인스턴스에서 사용자 지정 Amazon 머신 이미지(AMI)를 생성하였습니다. 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오.
6. (선택 사항) 더 이상 필요하지 않은 인스턴스를 종료하였습니다.
7. 필요한 AWS에 대한 액세스 권한을 애플리케이션에 부여하는 IAM 역할을 생성하였습니다. 자세한 내용은 [IAM 콘솔을 사용하여 IAM 역할을 생성하려면 다음을 수행합니다. \(p. 893\)](#) 단원을 참조하십시오.

애플리케이션 확장 및 로드 밸런싱

다음 절차를 사용하여 로드 밸런서를 만들고, 인스턴스에 대한 시작 구성을 만든 다음, 두 개 이상의 인스턴스가 포함된 Auto Scaling 그룹을 만들고, 로드 밸런서를 Auto Scaling 그룹과 연결합니다.

애플리케이션을 확장하고 로드 밸런싱하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 로드 밸런싱 아래에서 로드 밸런서를 선택합니다.
3. 로드 밸런서 생성을 선택하십시오.
4. Application Load Balancer에서 생성을 선택합니다.
5. 로드 밸런서 구성 페이지에서 다음 작업을 수행하십시오.
 - a. Name에 로드 밸런서 이름을 입력합니다. 예: **my-1b**.
 - b. 체계의 internet-facing은 기본값으로 유지합니다.
 - c. 포트 80에서 HTTP 트래픽을 수락하는 리스너를 뜻하는 리스너는 기본값으로 유지합니다.
 - d. 가용 영역에서 인스턴스에 사용한 VPC를 선택합니다. 가용 영역을 선택한 후 해당 가용 영역의 퍼블릭 서브넷을 선택합니다. 두 번째 가용 영역에 대해 이 절차를 반복합니다.
 - e. 다음: 보안 설정 구성을 선택합니다.
6. 이 자습서의 경우 안전한 리스너를 사용하지 않고 있습니다. 다음: 보안 그룹 구성을 선택합니다.
7. 보안 그룹 구성 페이지에서 다음 작업을 수행하십시오.
 - a. Create a new security group을 선택합니다.
 - b. 보안 그룹의 이름과 설명을 입력하거나 기본 이름과 설명을 유지합니다. 이 새 보안 그룹에는 리스너에 구성된 포트로 보내는 트래픽을 허용하는 규칙이 포함되어 있습니다.
 - c. 다음: 라우팅 구성을 선택합니다.
8. 라우팅 구성 페이지에서 다음을 수행합니다.
 - a. 대상 그룹에서는 기본값인 새 대상 그룹을 유지합니다.
 - b. 이름에 대상 그룹의 이름을 입력합니다.
 - c. 프로토콜을 HTTP로, 포트를 80으로, 대상 유형을 인스턴스로 유지합니다.
 - d. 상태 검사의 기본 프로토콜과 경로를 유지합니다.
 - e. Next: Register Targets(다음: 대상 등록)를 선택합니다.
9. Amazon EC2 Auto Scaling을 사용하여 EC2 인스턴스를 대상 그룹에 추가해야 하므로 대상 등록 페이지에서 다음: 검토를 선택하여 다음 페이지로 계속합니다.
10. 검토 페이지에서 생성을 선택합니다. 로드 밸런서가 생성된 후 닫기를 선택합니다.
11. 탐색 창의 AUTO SCALING에서 구성 시작을 선택합니다.
 - Amazon EC2 Auto Scaling를 처음 사용하는 경우 시작 페이지가 표시됩니다. Auto Scaling 그룹 생성을 선택하여 Auto Scaling 그룹 생성 마법사를 시작한 후 시작 구성 생성을 선택합니다.
 - 그렇지 않으면 시작 구성 생성을 선택합니다.
12. AMI 선택 페이지에서 My AMIs(내 AMI) 탭을 선택한 후 [사전 조건 \(p. 1168\)](#)에서 생성한 AMI를 선택합니다.
13. 인스턴스 유형 선택 페이지에서 인스턴스 유형을 선택한 후 다음: 세부 정보 구성을 선택합니다.
14. 세부 정보 구성 페이지에서 다음을 수행합니다.
 - a. 이름에 시작 구성의 이름을 입력합니다(예 **my-launch-config**).
 - b. IAM 역할에서, [사전 조건 \(p. 1168\)](#)에서 만든 IAM 역할을 선택합니다.
 - c. (선택 사항) 스타트업 스크립트를 실행해야 하는 경우 고급 세부 정보를 확장하고 사용자 데이터에 스크립트를 입력합니다.
 - d. Skip to review(검토로 이동)를 선택합니다.
15. 검토 페이지에서 보안 그룹 편집을 선택합니다. 기존의 보안 그룹을 선택하거나 새로 만들 수 있습니다. 이 보안 그룹은 로드 밸런서의 HTTP 트래픽과 상태 확인을 허용해야 합니다. 인스턴스에 퍼블릭 IP 주소가 있는 경우 인스턴스에 연결해야 한다면 선택적으로 RDP 트래픽을 허용할 수 있습니다. 작업을 마쳤으면 검토를 선택합니다.
16. 검토 페이지에서 시작 구성 생성을 선택합니다.

17. 메시지가 나타나면 기존 키 페어를 선택하거나 새 키 페어를 만들거나 키 페어 없이 진행합니다. 승인 확인 인란을 선택한 다음 시작 구성 생성을 선택합니다.
18. 시작 구성이 생성된 후에는 Auto Scaling 그룹을 만들어야 합니다.
 - Amazon EC2 Auto Scaling을 처음 사용하고 Auto Scaling 그룹 생성 마법사를 사용 중인 경우 다음 단계로 자동으로 이동합니다.
 - 그렇지 않으면 이 시작 구성은 사용하여 Auto Scaling 그룹 생성을 선택합니다.
19. Auto Scaling 그룹 세부 정보 구성 페이지에서 다음 작업을 수행하십시오.
 - a. 그룹 이름에 Auto Scaling 그룹의 이름을 입력합니다. 예: **my-asg**.
 - b. 그룹 크기에 인스턴스 수를 입력합니다(예: 2). 각 가용 영역에서 인스턴스 수를 대략적으로 동일하게 유지하는 것이 좋습니다.
 - c. 네트워크에서 VPC를 선택하고 서브넷에서 2개의 퍼블릭 서브넷을 선택합니다.
 - d. 고급 세부 정보 아래에서 하나 이상의 로드 밸런서에서 트래픽 수신을 선택합니다. 대상 그룹에서 대상 그룹을 선택합니다.
 - e. Next: Configure scaling policies(다음: 조정 정책 구성)를 선택합니다.
20. Amazon EC2 Auto Scaling을 사용하여 그룹을 지정된 크기로 유지할 것이므로 Configure scaling policies(조정 정책 구성) 페이지에서 검토를 선택합니다. 나중에 이 Auto Scaling 그룹을 수동으로 확장하거나, 일정에 따라 그룹을 확장하도록 구성하거나, 필요에 따라 그룹을 확장하도록 구성할 수 있습니다.
21. 검토 페이지에서 Auto Scaling 그룹 생성을 선택합니다.
22. 그룹이 생성되면 닫기를 선택합니다.

로드 밸런서 테스트

클라이언트가 로드 밸런서에 요청을 보내면 로드 밸런서는 그 요청을 등록된 인스턴스 중 하나로 라우팅합니다.

로드 밸런서를 테스트하려면

1. 인스턴스가 준비되었는지 확인합니다. Auto Scaling 그룹 페이지에서 Auto Scaling 그룹을 선택한 후 인스턴스를 선택합니다. 처음에는 인스턴스가 Pending 상태로 되어 있습니다. 상태가 InService이면, 사용할 준비가 된 것입니다.
2. 인스턴스가 로드 밸런서에 등록되어 있는지 확인합니다. 대상 그룹 페이지에서 대상 그룹을 선택한 후 대상 탭을 선택합니다. 인스턴스의 상태가 initial이면, 아직 등록 중일 수도 있습니다. 인스턴스의 상태가 healthy이면 사용할 준비가 된 것입니다. 인스턴스가 준비되면 다음과 같이 로드 밸런서를 테스트할 수 있습니다.
3. 로드 밸런서 페이지에서 로드 밸런서를 선택합니다.
4. 설명 탭에서 DNS 이름을 찾습니다. 이름은 다음과 같은 형식으로 되어 있습니다.

my-lb-xxxxxxxxxx.us-west-2.elb.amazonaws.com

5. 웹 브라우저에서 로드 밸런서의 DNS 이름을 주소 표시줄에 붙여넣기하고 Enter 키를 누릅니다. 웹 사이트가 표시됩니다.

자습서: Amazon EC2에 Windows HPC 클러스터 설치

Amazon EC2 인스턴스를 사용하여 확장 가능한 Windows HPC(고성능 컴퓨팅) 클러스터를 시작할 수 있습니다. Windows HPC 클러스터에는 Active Directory 도메인 컨트롤러, DNS 서버, 헤드 노드 및 하나 이상의 컴퓨팅 노드가 필요합니다.

Amazon EC2에 Windows HPC 클러스터를 설치하려면 다음 작업을 수행합니다.

- 1단계: 보안 그룹 만들기 (p. 1171)
- 2단계: Active Directory 도메인 컨트롤러 설치 (p. 1174)
- 3단계: 헤드 노드 구성 (p. 1174)
- 4단계: 컴퓨팅 노드 설치 (p. 1176)
- 5단계: HPC 컴퓨팅 노드 용량 조정(선택 사항) (p. 1177)

고성능 컴퓨팅에 대한 자세한 정보는 [AWS의 HPC\(고성능 컴퓨팅\)](#) 단원을 참조하십시오.

사전 조건

VPC에서 인스턴스를 시작해야 합니다. 기본 VPC를 사용하거나 기본이 아닌 VPC를 만들 수 있습니다. 자세한 내용은 [시작하기](#)(출처: Amazon VPC 사용 설명서)를 참조하십시오.

1단계: 보안 그룹 만들기

Windows PowerShell용 도구를 사용하여 도메인 컨트롤러, 도메인 멤버, HPC 클러스터에 대한 보안 그룹을 만듭니다.

보안 그룹을 만들려면

1. [New-EC2SecurityGroup](#) cmdlet를 사용하여 도메인 컨트롤러에 대한 보안 그룹을 만듭니다. 출력에 표시된 보안 그룹 ID를 메모해둡니다.

```
PS C:\> New-EC2SecurityGroup -VpcId vpc-id -GroupName "SG - Domain Controller" -Description "Active Directory Domain Controller"
```

2. [New-EC2SecurityGroup](#) cmdlet를 사용하여 도메인 멤버에 대한 보안 그룹을 만듭니다. 출력에 표시된 보안 그룹 ID를 메모해둡니다.

```
PS C:\> New-EC2SecurityGroup -VpcId vpc-id -GroupName "SG - Domain Member" -Description "Active Directory Domain Member"
```

3. [New-EC2SecurityGroup](#) cmdlet를 사용하여 HPC 클러스터에 대한 보안 그룹을 만듭니다. 출력에 표시된 보안 그룹 ID를 메모해둡니다.

```
PS C:\> New-EC2SecurityGroup -VpcId vpc-id -GroupName "SG - Windows HPC Cluster" -Description "Windows HPC Cluster Nodes"
```

보안 그룹에 규칙을 추가하려면

1. 다음 규칙을 만들어 도메인 컨트롤러 보안 그룹에 추가합니다. 자리 표시자 보안 그룹 ID를 도메인 멤버 보안 그룹의 ID로 바꾸고 자리 표시자 CIDR 블록을 네트워크의 CIDR 블록으로 바꿉니다.

```
PS C:\> $sg_dm = New-Object Amazon.EC2.Model.UserIdGroupPair
PS C:\> $sg_dm.GroupId = "sg-12345678
PS C:\> $r1 = @{
    IpProtocol="UDP"; FromPort="123"; ToPort="123"; UserIdGroupPairs=$sg_dm
}
PS C:\> $r2 = @{
    IpProtocol="TCP"; FromPort="135"; ToPort="135"; UserIdGroupPairs=$sg_dm
}
PS C:\> $r3 = @{
    IpProtocol="UDP"; FromPort="138"; ToPort="138"; UserIdGroupPairs=$sg_dm
}
PS C:\> $r4 = @{
    IpProtocol="TCP"; FromPort="49152"; ToPort="65535"; UserIdGroupPairs=$sg_dm
}
```

```
PS C:\> $r5 = @{ IpProtocol="TCP"; FromPort="389"; ToPort="389"; UserIdGroupPairs=$sg_dm }
PS C:\> $r6 = @{ IpProtocol="UDP"; FromPort="389"; ToPort="389"; UserIdGroupPairs=$sg_dm }
PS C:\> $r7 = @{ IpProtocol="TCP"; FromPort="636"; ToPort="636"; UserIdGroupPairs=$sg_dm }
PS C:\> $r8 = @{ IpProtocol="TCP"; FromPort="3268"; ToPort="3269"; UserIdGroupPairs=$sg_dm }
PS C:\> $r9 = @{ IpProtocol="TCP"; FromPort="53"; ToPort="53"; UserIdGroupPairs=$sg_dm }
PS C:\> $r10 = @{ IpProtocol="UDP"; FromPort="53"; ToPort="53"; UserIdGroupPairs=$sg_dm }
PS C:\> $r11 = @{ IpProtocol="TCP"; FromPort="88"; ToPort="88"; UserIdGroupPairs=$sg_dm }
PS C:\> $r12 = @{ IpProtocol="UDP"; FromPort="88"; ToPort="88"; UserIdGroupPairs=$sg_dm }
PS C:\> $r13 = @{ IpProtocol="TCP"; FromPort="445"; ToPort="445"; UserIdGroupPairs=$sg_dm }
PS C:\> $r14 = @{ IpProtocol="UDP"; FromPort="445"; ToPort="445"; UserIdGroupPairs=$sg_dm }
PS C:\> $r15 = @{ IpProtocol="ICMP"; FromPort="-1"; ToPort="-1"; UserIdGroupPairs=$sg_dm }
PS C:\> $r16 = @{ IpProtocol="UDP"; FromPort="53"; ToPort="53";
IpRanges="203.0.113.25/32" }
PS C:\> $r17 = @{ IpProtocol="TCP"; FromPort="3389"; ToPort="3389";
IpRanges="203.0.113.25/32" }
```

- Grant-EC2SecurityGroupIngress cmdlet를 사용하여 도메인 컨트롤러 보안 그룹에 규칙을 추가합니다.

```
PS C:\> Grant-EC2SecurityGroupIngress -GroupId sg-1a2b3c4d -IpPermission @(
$r1, $r2, $r3, $r4, $r5, $r6, $r7, $r8, $r9, $r10, $r11, $r12, $r13, $r14, $r15, $r16, $r17 )
```

보안 그룹 규칙의 자세한 내용은 Microsoft 글, [도메인 및 트러스트를 위한 방화벽을 구성하는 방법](#)을 참조하십시오.

- 다음 규칙을 만들어 도메인 멤버 보안 그룹에 추가합니다. 자리 표시자 보안 그룹 ID를 도메인 컨트롤러 보안 그룹의 ID로 바꿉니다.

```
PS C:\> $sg_dc = New-Object Amazon.EC2.Model.UserIdGroupPair
PS C:\> $sg_dc.GroupId = "sg-1a2b3c4d"
PS C:\> $r1 = @{ IpProtocol="TCP"; FromPort="49152"; ToPort="65535"; UserIdGroupPairs=$sg_dc }
PS C:\> $r2 = @{ IpProtocol="UDP"; FromPort="49152"; ToPort="65535"; UserIdGroupPairs=$sg_dc }
PS C:\> $r3 = @{ IpProtocol="TCP"; FromPort="53"; ToPort="53"; UserIdGroupPairs=$sg_dc }
PS C:\> $r4 = @{ IpProtocol="UDP"; FromPort="53"; ToPort="53"; UserIdGroupPairs=$sg_dc }
```

- Grant-EC2SecurityGroupIngress cmdlet를 사용하여 도메인 멤버 보안 그룹에 규칙을 추가합니다.

```
PS C:\> Grant-EC2SecurityGroupIngress -GroupId sg-12345678 -IpPermission @(
$r1, $r2, $r3, $r4 )
```

- 다음 규칙을 만들어 HPC 클러스터 보안 그룹에 추가합니다. 자리 표시자 보안 그룹 ID를 HPC 클러스터 보안 그룹의 ID로 바꾸고 자리 표시자 CIDR 블록을 네트워크의 CIDR 블록으로 바꿉니다.

```
$sg_hpc = New-Object Amazon.EC2.Model.UserIdGroupPair
PS C:\> $sg_hpc.GroupId = "sg-87654321"
PS C:\> $r1 = @{ IpProtocol="TCP"; FromPort="80"; ToPort="80"; UserIdGroupPairs=$sg_hpc }
```

```
PS C:\> $r2 = @{ IpProtocol="TCP"; FromPort="443"; ToPort="443"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r3 = @{ IpProtocol="TCP"; FromPort="1856"; ToPort="1856"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r4 = @{ IpProtocol="TCP"; FromPort="5800"; ToPort="5800"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r5 = @{ IpProtocol="TCP"; FromPort="5801"; ToPort="5801"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r6 = @{ IpProtocol="TCP"; FromPort="5969"; ToPort="5969"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r7 = @{ IpProtocol="TCP"; FromPort="5970"; ToPort="5970"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r8 = @{ IpProtocol="TCP"; FromPort="5974"; ToPort="5974"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r9 = @{ IpProtocol="TCP"; FromPort="5999"; ToPort="5999"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r10 = @{ IpProtocol="TCP"; FromPort="6729"; ToPort="6730"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r11 = @{ IpProtocol="TCP"; FromPort="7997"; ToPort="7997"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r12 = @{ IpProtocol="TCP"; FromPort="8677"; ToPort="8677"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r13 = @{ IpProtocol="TCP"; FromPort="9087"; ToPort="9087"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r14 = @{ IpProtocol="TCP"; FromPort="9090"; ToPort="9092"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r15 = @{ IpProtocol="TCP"; FromPort="9100"; ToPort="9163"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r16 = @{ IpProtocol="TCP"; FromPort="9200"; ToPort="9263"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r17 = @{ IpProtocol="TCP"; FromPort="9794"; ToPort="9794"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r18 = @{ IpProtocol="TCP"; FromPort="9892"; ToPort="9893"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r19 = @{ IpProtocol="UDP"; FromPort="9893"; ToPort="9893"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r20 = @{ IpProtocol="TCP"; FromPort="6498"; ToPort="6498"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r21 = @{ IpProtocol="TCP"; FromPort="7998"; ToPort="7998"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r22 = @{ IpProtocol="TCP"; FromPort="8050"; ToPort="8050"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r23 = @{ IpProtocol="TCP"; FromPort="5051"; ToPort="5051"; UserIdGroupPairs=$sg_hpc }
PS C:\> $r24 = @{ IpProtocol="TCP"; FromPort="3389"; ToPort="3389"; IpRanges="203.0.113.25/32" }
```

6. [Grant-EC2SecurityGroupIngress cmdlet](#)를 사용하여 HPC 클러스터 보안 그룹에 규칙을 추가합니다.

```
PS C:\> Grant-EC2SecurityGroupIngress -GroupId sg-87654321 -IpPermission @($r1, $r2, $r3, $r4, $r5, $r6, $r7, $r8, $r9, $r10, $r11, $r12, $r13, $r14, $r15, $r16, $r17, $r18, $r19, $r20, $r21, $r22, $r23, $r24)
```

보안 그룹 규칙의 자세한 내용은 Microsoft 글, [HPC 클러스터 네트워킹: Windows 방화벽 구성](#)을 참조하십시오.

7. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
8. 탐색 창에서 보안 그룹을 선택합니다. 세 보안 그룹이 모두 목록에 표시되고 필수 규칙이 있는지 확인합니다.

2단계: Active Directory 도메인 컨트롤러 설치

Active Directory 도메인 컨트롤러는 HPC 환경에서 인증 및 중앙 집중식 리소스 관리를 제공하며 설치 시 필수 항목입니다. Active Directory를 설치하려면 HPC 클러스터의 도메인 컨트롤러 역할을 할 인스턴스를 시작하고 이를 구성합니다.

HPC 클러스터에 대한 도메인 컨트롤러를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 콘솔 대시보드에서 인스턴스 시작을 선택합니다.
3. AMI 선택 페이지에서 Windows Server용 AMI를 선택한 다음 선택을 선택합니다.
4. 마법사의 다음 페이지에서 인스턴스 유형을 선택한 다음 다음: 인스턴스 세부 정보 구성(Next: Configure Instance Details)을 선택합니다.
5. 인스턴스 세부 정보 구성 페이지의 네트워크에서 VPC를 선택하고 서브넷에서 서브넷을 선택합니다. 마법사의 다음 페이지에서 인스턴스에 대한 추가 스토리지를 지정할 수 있습니다.
6. 태그 추가 페이지에서 인스턴스의 Name 태그 값으로 Domain Controller를 입력하고 다음: 보안 그룹 구성(Next: Configure Security Group)을 선택합니다.
7. 보안 그룹 구성 페이지에서 기존 보안 그룹 선택>Select an existing security group)을 선택하고 SG - Domain Controller 보안 그룹을 선택한 후 검토 후 시작(Review and Launch)을 선택합니다.
8. 시작을 선택합니다.
9. 탐색 창에서 탄력적 IP를 선택합니다.
10. 새 주소 할당을 선택합니다. 할당을 선택합니다. 닫기를 선택합니다.
11. 생성한 탄력적 IP 주소를 선택하고 작업, 주소 연결을 선택합니다. 인스턴스에 도메인 컨트롤러 인스턴스를 선택합니다. 연결을 선택합니다.

생성한 인스턴스에 연결하고 서버를 HPC 클러스터의 도메인 컨트롤러로 구성합니다.

인스턴스를 도메인 컨트롤러로 구성하려면 다음을 수행합니다.

1. Domain Controller 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
2. 서버 관리자를 열고 Active Directory 도메인 서비스 역할을 추가합니다.
3. 서버 관리자를 사용하거나 DCPromo.exe를 실행하여 서버를 도메인 컨트롤러로 승격합니다.
4. 새 포리스트에 새 도메인을 생성합니다.
5. FQDN(Fully Qualified Domain Name)으로 **hpc.local**을 입력합니다.
6. 포리스트 기능 수준을 Windows Server 2008 R2로 선택합니다.
7. DNS 서버 옵션을 선택했는지 확인하고 다음을 선택합니다.
8. 예, DHCP 서버에서 자동으로 할당된 IP 주소를 사용합니다(권장되지 않음).를 선택합니다.
9. 메시지가 나타나면 예를 선택해 계속합니다.
10. 마법사를 완료한 다음 완료 시 다시 시작을 선택합니다.
11. 인스턴스에 **hpc.local\administrator**로 연결합니다.
12. 도메인 사용자 **hpc.local\hpcuser**를 생성합니다.

3단계: 헤드 노드 구성

HPC 클라이언트는 헤드 노드에 연결합니다. 헤드 노드는 예약된 작업을 처리합니다. 인스턴스를 시작하고, HPC Pack을 설치하고, 클러스터를 구성하여 헤드 노드를 구성합니다.

인스턴스를 시작한 후 필요한 사용자 계정을 사용하여 hpc.local 도메인의 멤버로 구성합니다.

인스턴스를 헤드 노드로 구성하려면 다음을 수행합니다.

1. 인스턴스를 시작하고 이름을 **HPC-Head**로 지정합니다. 인스턴스를 시작할 때 SG - Windows HPC Cluster 및 SG - Domain Member 보안 그룹을 모두 선택합니다.
2. 인스턴스에 연결하고 다음 명령을 사용하여 기존 DNS 서버 주소를 가져옵니다.

IPConfig /all

3. HPC-Head NIC의 TCP/IPv4 속성을 업데이트하여 Domain Controller 인스턴스의 탄력적 IP 주소를 기본 DNS로 포함한 후 이전 단계에서 확인한 DNS IP 주소를 추가합니다.
4. hpc.local(도메인 관리자 계정)의 자격 증명을 사용하여 머신을 hpc.local\administrator 도메인에 가입시킵니다.
5. hpc.local\hpcuser를 로컬 관리자로 추가합니다. 자격 증명을 입력하라는 메시지가 나타나면 hpc.local\administrator를 사용하고 인스턴스를 다시 시작합니다.
6. HPC-Head에 hpc.local\hpcuser로 연결합니다.

HPC Pack을 설치하려면 다음을 수행합니다.

1. hpc.local\hpcuser 계정을 사용하여 HPC-Head 인스턴스에 연결합니다.
2. 서버 관리자를 사용하여 관리자에 대한 Internet Explorer 보안 강화 구성(IE ESC)을 해제합니다.
 - a. 서버 관리자에서 보안 정보 아래에 있는 IE ESC 구성을 선택합니다.
 - b. 관리자에 대한 IE ESC를 해제합니다.
3. HPC-Head에 HPC Pack을 설치합니다.
 - a. [Microsoft Download Center](#)에서 HPC-Head에 HPC Pack을 다운로드합니다. HPC-Head의 Windows Server 버전에 맞는 HPC Pack을 선택합니다.
 - b. 폴더에 파일의 압축을 풀고 폴더를 연 후 setup.exe를 두 번 클릭합니다.
 - c. 설치 페이지에서 헤드 노드를 생성하여 새 HPC 클러스터 생성(Create a new HPC cluster by creating a head node)을 선택하고 다음을 선택합니다.
 - d. 헤드 모드에 모든 데이터베이스를 설치하는 기본 설정을 수락하고 다음을 선택합니다.
 - e. 마법사를 완료합니다.

헤드 노드에 HPC 클러스터를 구성하려면 다음을 수행합니다.

1. HPC Cluster Manager를 시작합니다.
2. 배포 작업 목록(Deployment To-Do List)에서 네트워크 구성(Configure your network)을 선택합니다.
 - a. 마법사에서 기본 옵션(5)을 선택하고 다음(Next)을 선택합니다.
 - b. 모든 화면에서 기본값을 수락하여 마법사를 완료하고 서버 업데이트 방법 및 고객 피드백 참여 여부를 선택합니다.
 - c. 구성을 선택합니다.
3. 네트워크 자격 증명 제공(Provide Network Credentials)을 선택하고 hpc.local\hpcuser 자격 증명을 입력합니다.
4. 새 노드의 이름 구성(Configure the naming of new nodes)을 선택한 다음 확인을 선택합니다.
5. 노드 템플릿 생성(Create a node template)을 선택합니다.
 - a. 컴퓨팅 노드 템플릿(Compute node template), 다음(Next)을 차례로 선택합니다.
 - b. 운영 체제 없이(Without operating system)를 선택하고 기본값으로 진행합니다.
 - c. Create를 선택합니다.

4단계: 컴퓨팅 노드 설치

인스턴스를 시작하고, HPC Pack을 설치한 다음, 클러스터에 노드를 추가하여 컴퓨팅 노드를 설치합니다.

먼저 인스턴스를 시작한 후 필요한 사용자 계정을 사용하여 hpc.local 도메인의 멤버로 구성합니다.

컴퓨팅 노드의 인스턴스를 구성하려면 다음을 수행합니다.

1. 인스턴스를 시작하고 이름을 HPC-Compute로 지정합니다. 인스턴스를 시작할 때 SG - Windows HPC 클러스터(SG - Windows HPC Cluster) 및 SG - 도메인 멤버(SG - Domain Member) 보안 그룹을 선택합니다.
2. 인스턴스에 로그인하고 다음 명령을 사용하여 HPC-Compute에서 기존 DNS 서버 주소를 가져옵니다.

IPConfig /all

3. HPC-Compute NIC의 TCP/IPv4 속성을 업데이트하여 Domain Controller 인스턴스의 탄력적 IP 주소를 기본 DNS로 포함합니다. 그런 다음 이전 단계에서 확인한 DNS IP 주소를 추가합니다.
4. hpc.local(도메인 관리자 계정)의 자격 증명을 사용하여 머신을 hpc.local\administrator 도메인에 가입시킵니다.
5. hpc.local\hpcuser를 로컬 관리자로 추가합니다. 자격 증명을 입력하라는 메시지가 나타나면 hpc.local\administrator를 사용하고 다시 시작합니다.
6. HPC-Compute에 hpc.local\hpcuser로 연결합니다.

컴퓨팅 노드에 HPC Pack을 설치하려면 다음을 수행합니다.

1. HPC-Compute 계정을 사용하여 hpc.local\hpcuser 인스턴스에 연결합니다.
2. 서버 관리자를 사용하여 관리자에 대한 Internet Explorer 보안 강화 구성(IE ESC)을 해제합니다.
 - a. 서버 관리자에서 보안 정보 아래에 있는 IE ESC 구성을 선택합니다.
 - b. 관리자에 대한 IE ESC를 해제합니다.
3. HPC-Compute에 HPC Pack을 설치합니다.
 - a. Microsoft Download Center에서 HPC-Compute에 HPC Pack을 다운로드합니다. HPC-Compute의 Windows Server 버전에 맞는 HPC Pack을 선택합니다.
 - b. 폴더에 파일의 압축을 풀고 폴더를 연 후 setup.exe를 두 번 클릭합니다.
 - c. 설치(Installation) 페이지에서 새 컴퓨팅 노드를 생성하여 기존 HPC 클러스터에 결합(Join an existing HPC cluster by creating a new compute node)을 선택하고 다음(Next)을 선택합니다.
 - d. HPC-Head 인스턴스의 정규화된 이름을 지정하고 기본값을 선택합니다.
 - e. 마법사를 완료합니다.

클러스터 구성을 완료하려면 헤드 노드에서 클러스터에 컴퓨팅 노드를 추가합니다.

클러스터에 컴퓨팅 노드를 추가하려면 다음을 수행합니다.

1. HPC-Head 인스턴스에 hpc.local\hpcuser로 연결합니다.
2. HPC Cluster Manager를 엽니다.
3. 노드 관리(Node Management)를 선택합니다.
4. 컴퓨팅 노드가 승인되지 않음(Unapproved) 버킷에 표시되는 경우 노드를 마우스 오른쪽 버튼으로 클릭하고 노드 추가(Add Node)를 선택합니다.
 - a. 이미 구성된 컴퓨팅 노드 또는 브로커 노드 추가(Add compute nodes or broker nodes that have already been configured)를 선택합니다.

- b. 노드 옆의 확인란을 선택하고 추가(Add)를 선택합니다.
5. 노드를 마우스 오른쪽 버튼으로 클릭하고 온라인으로 가져오기(Bring Online)를 선택합니다.

5단계: HPC 컴퓨팅 노드 용량 조정(선택 사항)

컴퓨팅 노드의 용량을 조정하려면 다음을 수행합니다.

1. HPC-Compute 인스턴스에 hpc.local\hpcuser로 연결합니다.
2. HP Pack 설치 패키지에서 로컬로 다운로드한 파일을 모두 삭제합니다. 이미 설치 프로그램을 실행하여 이미지에 해당 파일을 생성했으므로 이러한 파일은 AMI에 복제할 필요가 없습니다.
3. C:\Program Files\Amazon\Ec2ConfigService에서 sysprep2008.xml 파일을 엽니다.
4. <settings pass="specialize"> 마지막에 다음 섹션을 추가합니다. hpc.local, password 및 hpcuser를 환경에 맞게 바꾸어야 합니다.

```
<component name="Microsoft-Windows-UnattendedJoin" processorArchitecture="amd64"
    publicKeyToken="31bf3856ad364e35"
    language="neutral" versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/
    WMIConfig/2002/State"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Identification>
        <UnsecureJoin>false</UnsecureJoin>
        <Credentials>
            <Domain>hpc.local</Domain>
            <Password>password</Password>
            <Username>hpcuser</Username>
        </Credentials>
        <JoinDomain>hpc.local</JoinDomain>
    </Identification>
</component>
```

5. Save sysprep2008.xml.
6. 시작, 모든 프로그램, EC2ConfigService 설정을 선택합니다.
 - a. 일반 탭을 선택하고 컴퓨터 이름 설정 확인란의 선택을 취소합니다.
 - b. 번들 탭을 선택한 다음 Sysprep 실행 및 지금 종료(Run Sysprep and Shutdown Now)를 선택합니다.
7. Amazon EC2 콘솔을 엽니다.
8. 탐색 창에서 인스턴스를 선택합니다.
9. 인스턴스 상태가 중지가 되기를 기다립니다.
10. 인스턴스를 선택하고 작업, 이미지, 이미지 생성을 선택합니다.
11. 이미지 이름과 이미지 설명을 지정하고 이미지 생성을 선택하여 인스턴스로 AMI를 생성합니다.
12. 종료된 원래 HPC-Compute 인스턴스를 시작합니다.
13. hpc.local\hpcuser 계정을 사용하여 헤드 노드에 연결합니다.
14. HPC Cluster Manager에서 현재 오류 상태로 표시되는 이전 노드를 삭제합니다.
15. Amazon EC2 콘솔의 탐색 창에서 AMI를 선택합니다.
16. 생성한 AMI를 사용하여 클러스터에 노드를 더 추가합니다.

생성한 AMI에서 추가 컴퓨팅 노드를 시작할 수 있습니다. 이러한 노드는 도메인에 자동으로 가입되지만, HPC Cluster Manager에서 헤드 노드를 사용하여 이미 구성된 노드로 클러스터에 추가한 후 온라인화해야 합니다.

EC2 Windows 인스턴스 문제 해결

다음 절차 및 팁은 Amazon EC2 Windows 인스턴스 관련 문제를 해결하는 데 도움이 됩니다.

목차

- [인스턴스 시작 문제 해결 \(p. 1178\)](#)
- [Windows 인스턴스 연결 문제 해결 \(p. 1181\)](#)
- [연결할 수 없는 인스턴스 문제 해결 \(p. 1188\)](#)
- [기억나지 않거나 만료된 Windows 관리자 암호 재설정 \(p. 1196\)](#)
- [인스턴스 중지 문제 해결 \(p. 1204\)](#)
- [인스턴스 종료 문제 해결 \(p. 1205\)](#)
- [사용 EC2Rescue for Windows Server \(p. 1206\)](#)
- [진단 인터럽트 보내기\(고급 사용자 전용\) \(p. 1217\)](#)
- [Windows 인스턴스의 일반적인 문제 \(p. 1218\)](#)
- [일반적인 메시지를 통한 Windows 인스턴스 문제 해결 \(p. 1222\)](#)

인스턴스 문제 해결에 대한 자세한 내용은 [사용 EC2Rescue for Windows Server \(p. 1206\)](#) 단원을 참조하십시오. PV 드라이버 관련 문제를 해결하는 방법에 대한 자세한 내용은 [PV 드라이버 문제 해결 \(p. 534\)](#) 단원을 참조하십시오.

인스턴스 시작 문제 해결

다음 문제 때문에 인스턴스를 시작할 수 없습니다.

시작 문제

- [인스턴스 제한 초과됨 \(p. 1178\)](#)
- [부족한 인스턴스 용량 \(p. 1179\)](#)
- [인스턴스 즉시 종료 \(p. 1179\)](#)
- [Windows 시작 직후 높은 CPU 사용률 \(p. 1180\)](#)

인스턴스 제한 초과됨

설명

새 인스턴스를 시작하려 할 때 혹은 종지된 인스턴스를 다시 시작하려 할 때 `InstanceLimitExceeded` 오류가 발생합니다.

원인

새 인스턴스를 시작하거나 종지된 인스턴스를 다시 시작하려 할 때 `InstanceLimitExceeded` 오류가 발생하면, 한 리전에서 시작할 수 있는 인스턴스 제한에 도달한 것입니다. AWS 계정을 생성할 때 리전별로 실행할 수 있는 인스턴스의 기본 제한이 설정됩니다.

솔루션

리전을 기준으로 인스턴스 한도 증가를 요청할 수 있습니다. 자세한 내용은 [Amazon EC2 서비스 할당량 \(p. 1154\)](#) 단원을 참조하십시오.

부족한 인스턴스 용량

설명

새 인스턴스를 시작하려 할 때 혹은 종지된 인스턴스를 다시 시작하려 할 때 `InsufficientInstanceCapacity` 오류가 발생합니다.

원인

인스턴스를 시작하거나 종지된 인스턴스를 다시 시작하려 할 때 `InsufficientInstanceCapacity` 오류가 발생하면 현재 AWS에 요청에 대한 서비스를 제공할 수 있을 만큼 가용 온디맨드 용량이 충분하지 않은 것입니다.

솔루션

다음에 따라 문제를 해결하십시오.

- 몇 분 정도 기다린 후 다시 요청을 제출합니다. 용량은 자주 변할 수 있습니다.
- 인스턴스 수가 줄어든 새 요청을 제출하십시오. 예를 들어 단일 요청을 통해 인스턴스 15개를 시작하는 경우 인스턴스 5개에 대해 요청 3개 또는 인스턴스 1개 대신 요청 15개를 시도합니다.
- 인스턴스를 시작하고 있는 경우 가용 영역을 지정하지 않고 새 요청을 제출하십시오.
- 인스턴스를 시작하고 있는 경우 이후의 단계에서 크기를 조정할 수 있는 다른 인스턴스 유형을 사용하여 새 요청을 제출하십시오. 자세한 내용은 [인스턴스 유형 변경 \(p. 194\)](#) 단원을 참조하십시오.
- 클러스터 배치 그룹으로 인스턴스를 시작한 경우 용량 부족 오류가 발생할 수 있습니다. 자세한 내용은 [배치 그룹 규칙 및 제한 사항 \(p. 764\)](#) 단원을 참조하십시오.
- 어떤 기간에 대해서는 Amazon EC2 용량을 예약할 수 있는 온디맨드 용량 예약을 생성해 보십시오. 자세한 내용은 [온디맨드 용량 예약 \(p. 354\)](#)를 참조하십시오.
- 장기 용량 예약인 예약 인스턴스를 구입해 보십시오. 자세한 내용은 [Amazon EC2 예약 인스턴스](#) 단원을 참조하십시오.

인스턴스 즉시 종료

설명

인스턴스가 다시 시작한 후 `pending` 상태에서 `terminated` 상태로 즉시 변경됩니다.

원인

인스턴스가 즉시 종료되는 이유에는 다음과 같이 몇 가지가 있습니다.

- EBS 볼륨 제한에 도달했습니다.
- EBS 스냅샷이 손상되었습니다.
- 루트 EBS 볼륨이 암호화되었는데 사용자는 암호 해독을 위하여 KMS 키에 액세스할 권한이 없습니다.
- 인스턴스를 시작하는 데 사용한 인스턴스 스토어 지원 AMI에 필수 부분(`image.part.xx` 파일).

솔루션

Amazon EC2 콘솔 또는 AWS Command Line Interface를 사용하여 종료 이유를 파악할 수 있습니다.

Amazon EC2 콘솔을 사용해서 종료 이유를 파악하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
3. 설명 탭의 상태 전환 이유 레이블 옆에서 이유를 확인합니다.

AWS Command Line Interface를 사용해서 종료 이유를 파악하는 방법

1. `describe-instances` 명령을 사용하여 인스턴스 ID를 지정합니다.

```
aws ec2 describe-instances --instance-id instance_id
```

2. 명령으로 반환된 JSON 응답을 검토하고 StateReason 응답 요소의 값을 확인합니다.

다음 코드 블록은 StateReason 응답 요소의 예를 보여 줍니다.

```
"StateReason": {  
    "Message": "Client.VolumeLimitExceeded: Volume limit exceeded",  
    "Code": "Server.InternalError"  
},
```

이 문제를 해결하려면

확인한 종료 이유에 따라 다음 작업 중 하나를 선택합니다.

- 이유가 **Client.VolumeLimitExceeded: Volume limit exceeded**인 경우, EBS 볼륨 한도에 도달한 것입니다. 자세한 내용은 [인스턴스 볼륨 제한 \(p. 1104\)](#) 단원을 참조하십시오. Amazon EBS 볼륨 제한 증가 요청을 제출하려면 AWS Support 센터에서 [사례 생성](#) 양식을 작성하십시오. 자세한 내용은 [Amazon EC2 서비스 할당량 \(p. 1154\)](#) 단원을 참조하십시오.
- 이유가 **Client.InternalError: Client error on launch**인 경우, 루트 볼륨이 암호화되었는데 사용자는 암호 해독을 위하여 KMS 키에 액세스할 권한이 없습니다. 필요한 KMS 키에 액세스할 권한을 얻으려면 IAM 사용자에게 해당 KMS 권한을 추가하십시오. 자세한 정보는 AWS Key Management Service Developer Guide의 [AWS KMS에서 키 정책 사용](#)을 참조하십시오.

Windows 시작 직후 높은 CPU 사용률

Windows Update가 업데이트를 확인하지만 다운로드 및 설치 여부는 직접 선택(기본 인스턴스 설정)으로 설정되어 있는 경우 이 검사를 실행하면 인스턴스 CPU가 50~99%까지 소비될 수 있습니다. 이로 인해 애플리케이션에 문제가 발생할 경우에는 제어판에서 Windows Update 설정을 수동으로 변경하거나, 혹은 Amazon EC2 사용자 데이터 필드에서 다음 스크립트를 사용하십시오.

```
reg add "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\WindowsUpdate\Auto Update" /v  
AUOptions /t REG_DWORD /d 3 /f net stop wuauserv net start wuauserv
```

이 스크립트를 실행할 때는 /d 값을 지정합니다. 기본값은 3입니다. 가능한 값은 다음을 포함합니다.

1. 업데이트를 확인하지 않음
2. 업데이트를 확인하지만 다운로드 및 설치 여부는 직접 선택

3. 업데이트를 다운로드하지만 설치 여부는 직접 선택
4. 업데이트 자동 설치

Amazon EBS 지원 인스턴스에 대한 사용자 데이터를 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 인스턴스를 선택합니다.
3. 작업을 선택하고 인스턴스 상태를 선택한 후 중지를 선택합니다.
4. 확인 대화 상자가 나타나면 예, 중지를 선택합니다. 인스턴스가 중지하는 데 몇 분 정도 걸릴 수 있습니다.
5. 인스턴스가 선택된 상태에서 작업을 선택하고 인스턴스 설정을 선택한 후 사용자 데이터 보기/변경을 선택합니다. 인스턴스가 실행 중일 때는 사용자 데이터를 변경할 수 없지만 볼 수는 있습니다.
6. 사용자 데이터 보기/변경 대화 상자에서 사용자 데이터를 업데이트하고 저장을 선택합니다.

인스턴스의 사용자 데이터를 변경한 후에는 인스턴스를 실행할 수 있습니다. 자세한 내용은 [사용자 데이터 실행 \(p. 570\)](#) 단원을 참조하십시오.

Windows 인스턴스 연결 문제 해결

다음은 Windows 인스턴스에 연결하는 중에 발생할 수 있는 문제와 오류 메시지입니다.

목차

- [원격 데스크톱으로 원격 컴퓨터에 연결할 수 없음 \(p. 1181\)](#)
- [macOS RDP 클라이언트 사용 중 오류 발생 \(p. 1184\)](#)
- [RDP에 바탕 화면이 아닌 빈 화면 표시 \(p. 1184\)](#)
- [관리자가 아닌 사용자 계정으로 인스턴스에 원격 로그온할 수 없음 \(p. 1184\)](#)
- [AWS 시스템 관리자를 사용하여 원격 데스크톱 연결 문제 해결 \(p. 1184\)](#)
- [원격 레지스트리를 사용하여 EC2 인스턴스에서 원격 데스크톱 활성화 \(p. 1187\)](#)

원격 데스크톱으로 원격 컴퓨터에 연결할 수 없음

인스턴스 연결과 관련된 문제를 해결하려면 다음을 수행합니다.

- 올바른 퍼블릭 DNS 호스트 이름을 사용하고 있는지 확인합니다. Amazon EC2 콘솔에서 인스턴스를 선택하고 세부 정보 창에서 퍼블릭 DNS(IPv4)를 확인합니다. 인스턴스가 VPC에 있는데 퍼블릭 DNS 이름이 표시되지 않는 경우 DNS 호스트 이름을 활성화해야 합니다. 자세한 내용은 [Amazon VPC 사용 설명서의 VPC에서 DNS 사용](#)을 참조하십시오.
- 인스턴스에 퍼블릭 IPv4 주소가 있는지 확인합니다. 퍼블릭 IP 주소가 없을 경우 인스턴스와 탄력적 IP 주소를 연결할 수 있습니다. 자세한 내용은 [탄력적인 IP 주소 \(p. 721\)](#) 단원을 참조하십시오.
- IPv6 주소를 사용해 인스턴스에 연결하려면 로컬 컴퓨터에 IPv6 주소가 있고 IPv6를 사용하도록 구성되어 있어야 합니다. Windows Server 2008 SP2 AMI 이하 버전에서 인스턴스를 시작한 경우, 인스턴스에 할당된 IPv6 주소를 인식하도록 인스턴스가 자동으로 구성되지 않습니다. 자세한 내용은 [Amazon VPC 사용 설명서의 인스턴스에서 IPv6 구성하기](#)를 참조하십시오.
- 보안 그룹에 RDP 액세스를 허용하는 규칙이 있는지 확인합니다. 자세한 내용은 [보안 그룹 생성 \(p. 13\)](#) 단원을 참조하십시오.
- 암호를 복사했는데 `Your credentials did not work` 오류가 표시되는 경우 메시지가 나타날 때 수동으로 입력해 봅니다. 암호를 복사할 때 빠진 문자가 있거나 공백 문자가 잘못 들어갔을 수 있습니다.

- 인스턴스가 상태 확인을 통과했는지 확인합니다. 자세한 정보는 [인스턴스 상태 확인 \(p. 651\)](#) 및 [상태 확인에 실패한 인스턴스 문제 해결](#)(Linux 인스턴스용 Amazon EC2 사용 설명서)을 참조하십시오.
- 서브넷의 라우팅 테이블에 VPC 외부로 향하는 모든 트래픽을 VPC의 인터넷 게이트웨이로 전송하는 경로가 있는지 확인합니다. 자세한 정보는 Amazon VPC 사용 설명서의 [사용자 지정 라우팅 테이블 만들기](#)(인터넷 게이트웨이) 단원을 참조하십시오.
- Windows 방화벽이나 기타 방화벽 소프트웨어가 인스턴스로 전송되는 RDP 트래픽을 차단하지 않는지 확인합니다. Windows 방화벽을 비활성화하고 보안 그룹 규칙을 사용하여 인스턴스의 액세스를 제어하는 것이 좋습니다. [SSM 에이전트를 사용하여 Windows 방화벽 비활성화에 대한 AWSSupport-TroubleshootRDP \(p. 1184\)](#)를 사용할 수 있습니다. AWS 시스템 관리자에 대해 구성되지 않은 Windows 인스턴스에서 Windows 방화벽을 비활성화하려면 [AWSSupport-ExecuteEC2Rescue \(p. 1186\)](#)를 사용하거나 다음 수동 단계를 사용하십시오.

수동 단계

- 해당 인스턴스를 중지하고 루트 볼륨을 분리합니다.
- 해당 인스턴스와 동일한 가용 영역에서 임시 인스턴스를 시작합니다.

Warning

임시 인스턴스가 원본 인스턴스와 동일한 AMI를 기반으로 하는 경우 추가 단계를 수행해야 합니다. 그렇지 않으면 디스크 서명 충돌로 인해 루트 볼륨 복원 후 원본 인스턴스를 부팅할 수 없습니다. 또는 임시 인스턴스에 대한 다른 AMI를 선택합니다. 예를 들어, 원본 인스턴스에서 Windows Server 2008 R2용 AWS Windows AMI를 사용할 경우 Windows Server 2012용 AWS Windows AMI를 사용하여 임시 인스턴스를 시작합니다.

- 해당 인스턴스의 루트 볼륨을 이 임시 인스턴스에 연결합니다. 임시 인스턴스에 연결하고 디스크 관리 유ти리티를 열어서 드라이브를 온라인 상태로 만듭니다.
- Regedit를 열고 HKEY_LOCAL_MACHINE을 선택합니다. 파일 메뉴에서 Hive 로드를 선택합니다. 드라이브를 선택하고, Windows\System32\config\SYSTEM 파일을 열고, 메시지가 나타나면 키 이름을 임의로 지정합니다.
- 방금 로드한 키를 선택하고 ControlSet001\Services\SharedAccess\Parameters\FirewallPolicy 경로로 이동합니다. 이름이 xxxxProfile 형식인 각 키를 선택하고 EnableFirewall을 1에서 0으로 변경합니다. 키를 한 번 더 선택하고 파일 메뉴에서 Hive 로드 취소를 선택합니다.
- (선택 사항) 임시 인스턴스가 원본 인스턴스와 동일한 AMI를 기반으로 하는 경우 다음 단계를 수행해야 합니다. 그렇지 않으면 디스크 서명 충돌로 인해 루트 볼륨 복원 후 원본 인스턴스를 부팅할 수 없습니다.

Warning

다음 절차에서는 레지스트리 편집기를 사용하여 Windows 레지스트리를 편집하는 방법을 설명합니다. Windows 레지스트리에 대해 또는 레지스트리 편집기를 사용하여 안전하게 수정하는 방법을 잘 알지 못하는 경우 [레지스트리 구성](#)을 참조하십시오.

- 명령 프롬프트를 열고 regedit.exe를 입력한 후 Enter 키를 누릅니다.
- 레지스트리 편집기의 컨텍스트 메뉴(마우스 오른쪽 버튼 클릭)에서 HKEY_LOCAL_MACHINE을 선택한 후 찾기를 선택합니다.
- Windows Boot Manager를 입력한 후 다음 찾기를 선택합니다.
- 키 11000001을 선택합니다. 이 키는 이전 단계에서 찾은 키 바로 위의 키입니다.
- 오른쪽 창에서 Element를 선택한 후 컨텍스트 메뉴(마우스 오른쪽 버튼 클릭)에서 수정을 선택합니다.
- 데이터에서 오프셋 0x38의 4바이트 디스크 서명을 찾습니다. 바이트를 거꾸로 하여 디스크 서명을 만들고 기록해둡니다. 예를 들어, 다음 데이터가 나타내는 디스크 서명은 E9EB3AA5입니다.

...

```
0030 00 00 00 00 01 00 00 00  
0038 A5 3A EB E9 00 00 00 00  
0040 00 00 00 00 00 00 00 00  
...
```

- g. 명령 프롬프트 창에서 다음 명령을 실행하여 Microsoft DiskPart를 시작합니다.

```
diskpart
```

- h. 다음 DiskPart 명령을 실행하여 볼륨을 선택합니다. 디스크 관리 유ти리티를 사용하여 디스크 번호가 1인지 확인할 수 있습니다.

```
DISKPART> select disk 1  
  
Disk 1 is now the selected disk.
```

- i. 다음 DiskPart 명령을 실행하여 디스크 서명을 봅니다.

```
DISKPART> uniqueid disk  
  
Disk ID: OC764FA8
```

- j. 이전 단계에서 표시된 디스크 서명이 앞에서 기록한 BCD의 디스크 서명과 일치하지 않을 경우 다음 DiskPart 명령을 사용하여 일치하도록 디스크 서명을 변경합니다.

```
DISKPART> uniqueid disk id=E9EB3AA5
```

7. 디스크 관리 유ти리티를 사용하여 드라이브를 오프라인으로 설정합니다.

Note

임시 인스턴스가 영향을 받는 인스턴스와 동일한 운영 체제를 실행하고 있는 경우 드라이브가 자동으로 오프라인 상태가 되므로 사용자가 수동으로 드라이브를 오프라인으로 설정할 필요가 없습니다.

8. 임시 인스턴스에서 볼륨을 분리합니다. 임시 인스턴스를 더 이상 사용하지 않는 경우 해당 인스턴스는 종료해도 됩니다.
9. 영향을 받은 인스턴스의 루트 볼륨을 /dev/sda1로서 연결하여 이를 복원합니다.
10. 인스턴스를 시작합니다.

- 네트워크 수준 인증이 Active Directory 도메인의 일부가 아닌 인스턴스에 비활성화되어 있는지 확인합니다([AWSSupport-TroubleshootRDP \(p. 1184\)](#) to [NLA 비활성화 사용](#)).
- 원격 데스크톱 서비스(TermService) 시작 유형이 '자동'으로 설정되어 있으며 해당 서비스가 시작된 상태인지 확인합니다([AWSSupport-TroubleshootRDP \(p. 1184\)](#) to [RDP 서비스 활성화 및 시작 사용](#)).
- 기본값이 3389인 을바른 원격 데스크톱 프로토콜 포트에 연결 중인지 확인합니다([현재 RDP 포트 읽기](#)에 대한 [AWSSupport-TroubleshootRDP \(p. 1184\)](#) 및 [3389로 다시 변경 사용](#)).
- 원격 데스크톱 연결이 사용자의 인스턴스에 허용되어 있는지 확인합니다([원격 데스크톱 연결 활성화](#)에 대한 [AWSSupport-TroubleshootRDP \(p. 1184\)](#) 사용).
- 암호가 만료되지 않았는지 확인합니다. 암호가 만료된 경우 재설정할 수 있습니다. 자세한 내용은 [기억나지 않거나 만료된 Windows 관리자 암호 재설정 \(p. 1196\)](#) 단원을 참조하십시오.
- 인스턴스에 생성한 사용자 계정으로 연결하려고 하면 The user cannot connect to the server due to insufficient access privileges 오류가 발생하는 경우 사용자에게 로컬 로그인 권한을 부여했는지 확인합니다. 자세한 내용은 [멤버에게 로컬 로그인 권한 부여](#)를 참조하십시오.
- 동시에 허용되는 최대 RDP 세션 수를 초과한 경우 Your Remote Desktop Services session has ended. Another user connected to the remote computer, so your connection was lost. 메시지와 함께 세션이 종료됩니다. 기본적으로 허용되는 인스턴스의 동시 RDP 세션 수는 2개입니다.

macOS RDP 클라이언트 사용 중 오류 발생

Microsoft 웹 사이트의 원격 데스크톱 연결을 사용하여 Windows Server 2012 R2 인스턴스에 연결하려는 경우 다음 오류가 발생할 수 있습니다.

```
Remote Desktop Connection cannot verify the identity of the computer that you want to connect to.
```

Mac App Store에서 Microsoft Remote Desktop 앱을 다운로드한 후 이 앱을 사용하여 인스턴스에 연결합니다.

RDP에 바탕 화면이 아닌 빈 화면 표시

다음에 따라 문제를 해결하십시오.

- 콘솔 출력에서 상세 정보를 확인합니다. Amazon EC2 콘솔을 사용하여 인스턴스의 콘솔 출력을 확인하려면 인스턴스를 선택하고, 작업을 선택하고, 인스턴스 설정을 선택한 다음 시스템 로그 가져오기를 선택합니다.
- 최신 버전의 RDP 클라이언트를 실행하는지 확인합니다.
- RDP 클라이언트에 기본 설정을 적용해 봅니다. 자세한 내용은 [원격 세션 환경](#)을 참조하십시오.
- 원격 데스크톱 연결을 사용하는 경우 다음과 같이 /admin 옵션과 함께 시작해 봅니다.

```
mstsc /v:instance /admin
```

- 서버에서 전체 화면 애플리케이션을 실행하는 경우 응답이 중지되었을 수 있습니다. Ctrl+Shift+Esc를 눌러 Windows 작업 관리자를 시작하고 애플리케이션을 닫습니다.
- 서버 사용률이 과다한 경우 응답이 중지되었을 수 있습니다. Amazon EC2 콘솔을 사용하여 인스턴스를 모니터링하려면 인스턴스를 선택하고 모니터링 탭을 선택합니다. 인스턴스 유형을 더욱 큰 유형으로 변경해야 하는 경우 [인스턴스 유형 변경](#) (p. 194) 단원을 참조하십시오.

관리자가 아닌 사용자 계정으로 인스턴스에 원격 로그 온할 수 없음

관리자 계정이 아닌 사용자 계정에서 Windows 인스턴스에 원격으로 로그온할 수 없는 경우 로컬에서 로그온할 수 있는 사용자 권한을 받았는지 확인합니다. [사용자 또는 그룹에 도메인의 도메인 컨트롤러에 로컬로 로그온할 수 있는 권한 부여](#)를 참조하십시오.

AWS 시스템 관리자를 사용하여 원격 데스크톱 연결 문제 해결

AWS 시스템 관리자를 통해 RDP를 사용하는 Windows 인스턴스에 연결하는 문제를 해결할 수 있습니다.

AWSSupport-TroubleshootRDP

사용자는 AWSSupport-TroubleshootRDP 자동화 문서를 사용하여 RDP 포트, 네트워크 계층 인증(NLA) 및 Windows 방화벽 프로파일 등 원격 데스크톱 프로토콜(RDP) 연결에 영향을 미치는 대상 인스턴스의 일반 설정을 확인하거나 수정할 수 있습니다. 기본적으로 이 문서는 이러한 설정 값을 읽고 출력합니다.

AWSSupport-TroubleshootRDP 자동화 문서는 EC2 인스턴스, 온프레미스 인스턴스 및 AWS 시스템 관리자(관리형 인스턴스)에서 사용할 수 있도록 설정된 VM(가상 머신)에 사용할 수 있습니다. 또한 시스템 관리자

와 함께 사용하도록 설정되지 않은 Windows Server용 EC2 인스턴스에도 사용할 수 있습니다. AWS 시스템 관리자와 함께 사용하도록 인스턴스를 설정하는 방법에 대한 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [AWS 시스템 관리자 관리형 인스턴스](#)를 참조하십시오.

AWSSupport-TroubleshootRDP 문서 사용 문제를 해결하려면

1. [시스템 관리자 콘솔](#)에 로그인합니다.
2. 손상된 인스턴스와 동일한 리전에 있는지 확인합니다.
3. [AWSSupport-TroubleshootRDP](#) 문서를 엽니다.
4. Execution Mode(실행 모드)에서 Simple execution(단순 실행)을 선택합니다.
5. Input parameters(입력 파라미터)의 InstanceId에서 Show interactive instance picker(대화형 인스턴스 선택기 표시)를 활성화합니다.
6. Amazon EC2 인스턴스를 선택합니다.
7. 예제 (p. 1185)를 검토한 후 실행을 선택합니다.
8. 실행 상황을 모니터링하려면 실행 상태에서 상태가 대기 중에서 성공으로 바뀔 때까지 기다립니다. 출력을 확장하여 결과를 봅니다. 실행된 단계에서 개별 단계의 출력을 보려면 스텝 ID를 선택합니다.

AWSSupport-TroubleshootRDP 예

다음 예제는 AWSSupport-TroubleshootRD를 통해 일반적인 문제 해결 작업을 수행하는 방법을 보여줍니다. 예제 AWS CLI `start-automation-execution` 명령이나 AWS Management 콘솔에 대해 제공된 링크를 사용할 수 있습니다.

Example 예: 현재의 RDP 상태 확인

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --  
parameters "InstanceId=instance_id" --region region_code
```

AWS 시스템 관리자 콘솔:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-  
TroubleshootRDP?region=region#documentVersion=$LATEST
```

Example 예: Windows 방화벽 비활성화

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --  
parameters "InstanceId=instance_id,Firewall=Disable" --region region_code
```

AWS 시스템 관리자 콘솔:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-  
TroubleshootRDP?region=region_code#documentVersion=$LATEST&Firewall=Disable
```

Example 예: 네트워크 수준 인증 비활성화

AWS CLI:

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
AWS 시스템 관리자를 사용하
여 원격 데스크톱 연결 문제 해결

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --  
parameters "InstanceId=instance_id,NLASettingAction
```

AWS 시스템 관리자 콘솔:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-  
TroubleshootRDP?region=region_code#documentVersion
```

Example 예: RDP 서비스 시작 유형을 '자동'으로 설정하고 RDP 서비스를 시작

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --  
parameters "InstanceId=instance_id,RDPServiceStartupType=Auto, RDPServiceAction=Start" --  
region region_code
```

AWS 시스템 관리자 콘솔:

```
https://console.aws.amazon.com/systems-manager/automation/execute/  
AWSSupport-TroubleshootRDP?region=region_code#documentVersion=  
$LATEST&RDPServiceStartupType=Auto&RDPServiceAction=Start
```

Example 예: 기본 RDP 포트(3389) 복원

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --  
parameters "InstanceId=instance_id,RDPPortAction=Modify" --region region_code
```

AWS 시스템 관리자 콘솔:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-  
TroubleshootRDP?region=region_code#documentVersion=$LATEST&RDPPortAction=Modify
```

Example 예: 원격 연결 허용

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --  
parameters "InstanceId=instance_id,RemoteConnections=Enable" --region region_code
```

AWS 시스템 관리자 콘솔:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-  
TroubleshootRDP?region=region_code#documentVersion=$LATEST&RemoteConnections=Enable
```

AWS Support-ExecuteEC2Rescue

AWS Support-ExecuteEC2Rescue 자동화 문서는 [사용 EC2Rescue for Windows Server \(p. 1206\)](#)를 사용하
여 EC2 인스턴스 연결 및 RDP 문제를 자동으로 해결하고 이를 복원합니다. 자세한 정보는 [연결할 수 없는
인스턴스에서 EC2Rescue 도구 실행](#)을 참조하십시오.

AWSSupport-ExecuteEC2Rescue 문서는 해당 인스턴스를 중지하고 재시작해야 합니다. 시스템 관리자 자동화는 인스턴스를 중지하고 Amazon 머신 이미지(AMI)를 생성합니다. 인스턴스 스토어 볼륨에 저장되어 있는 데이터는 사라집니다. 탄력적 IP 주소를 사용하지 않는 경우 퍼블릭 IP 주소가 변경됩니다. 자세한 정보는 AWS 시스템 관리자 사용 설명서의 [연결할 수 없는 인스턴스에서 EC2Rescue 도구 실행](#)을 참조하십시오.

AWSSupport-ExecuteEC2Rescue 문서 사용 문제를 해결하려면

1. [시스템 관리자 콘솔](#)을 엽니다.
2. 손상된 Amazon EC2 인스턴스와 동일한 리전에 있는지 확인합니다.
3. [AWSSupport-ExecuteEC2Rescue](#) 문서를 엽니다.
4. Execution Mode(실행 모드)에서 Simple execution(단순 실행)을 선택합니다.
5. 입력 파라미터 섹션에서 연결할 수 없는 인스턴스의 Amazon EC2 인스턴스 ID를 UnreachableInstanceId에 입력합니다.
6. (선택 사항) Amazon EC2 인스턴스의 문제 해결을 위해 운영 체제 로그를 수집하려면 LogDestination에 Amazon Simple Storage Service(Amazon S3) 버킷 이름을 입력합니다. 지정된 버킷으로 로그가 자동으로 업로드됩니다.
7. 실행을 선택합니다.
8. 실행 상황을 모니터링하려면 실행 상태에서 상태가 대기 중에서 성공으로 바뀔 때까지 기다립니다. 출력을 확장하여 결과를 봅니다. 실행된 단계에서 개별 단계의 출력을 보려면 스텝 ID를 선택합니다.

원격 레지스트리를 사용하여 EC2 인스턴스에서 원격 데스크톱 활성화

연결할 수 없는 인스턴스가 AWS Systems Manager 세션 관리자에 의해 관리되지 않는 경우 원격 레지스트리를 사용하여 원격 데스크톱을 활성화할 수 있습니다.

1. EC2 콘솔에서 연결할 수 없는 인스턴스를 중지합니다.
2. 연결할 수 없는 인스턴스의 루트 볼륨을 동일한 가용 영역에 있는 다른 인스턴스에 연결합니다.
3. 루트 볼륨을 연결한 인스턴스에서 디스크 관리를 엽니다. 디스크 관리를 열려면 다음을 실행하십시오.

`diskmgmt.msc`

4. 영향을 받는 인스턴스의 루트 볼륨을 마우스 오른쪽 버튼으로 클릭하고 온라인을 선택합니다.
5. 다음 명령을 실행하여 Windows 레지스트리 편집기를 엽니다.

`regedit`

6. 레지스트리 편집기 콘솔 트리에서 HKEY_LOCAL_MACHINE을 선택한 다음 파일>Load Hive(Hive 로드)를 선택합니다.
7. 연결된 볼륨의 드라이브를 선택하고 \Windows\System32\config\로 이동하여 SYSTEM을 선택한 다음 열기를 선택합니다.
8. 키 이름에 Hive의 고유한 이름을 입력하고 확인을 선택합니다.
9. 레지스트리를 변경하기 전에 레지스트리 Hive를 백업합니다.
 - a. 레지스트리 편집기 콘솔 트리에서 로드한 Hive HKEY_LOCAL_MACHINE*your key name*을 선택합니다.
 - b. 파일>내보내기를 선택합니다.
 - c. Export Registry File(레지스트리 파일 내보내기) 대화 상자에서 백업 복사본을 저장할 위치를 선택한 다음 파일 이름 필드에 백업 파일의 이름을 입력합니다.

- d. 저장을 선택합니다.
10. 레지스트리 편집기 콘솔 트리에서 HKEY_LOCAL_MACHINE\your key name\ControlSet001\Control\Terminal Server로 이동한 다음 세부 정보 창에서 fDenyTSConnections를 두 번 클릭합니다.
11. Edit DWORD(DWORD 편집) 값 상자의 Value data(값 데이터) 필드에 0을 입력합니다.
12. 확인을 선택합니다.

Note

Value data(값 데이터) 필드의 값이 1인 경우 인스턴스에서 원격 데스크톱 연결을 거부합니다.
0 값은 원격 데스크톱 연결을 허용합니다.

13. 레지스트리 편집기와 디스크 관리 콘솔을 닫습니다.
14. EC2 콘솔에서 루트 볼륨을 연결한 인스턴스에서 분리하고 연결할 수 없는 인스턴스에 다시 연결합니다.
연결할 수 없는 인스턴스에 볼륨을 연결할 때 디바이스 필드에 /dev/sda1을 입력합니다.
15. 연결할 수 없는 인스턴스를 다시 시작합니다.

연결할 수 없는 인스턴스 문제 해결

SSH 또는 RDP를 통해 인스턴스에 연결할 수 없는 경우 인스턴스의 스크린샷을 캡처하여 이미지로 볼 수 있습니다. 즉, 인스턴스의 상태에 관한 가시성이 제공되므로 더 빠르게 문제를 해결할 수 있습니다. Windows Server 2008 이상을 실행하는 인스턴스에서 [EC2 Rescue \(p. 1206\)](#)를 사용하여 오프라인 인스턴스에서 날짜를 수집하고 분석할 수도 있습니다.

- [연결할 수 없는 인스턴스의 스크린샷을 얻는 방법 \(p. 1188\)](#)
- [공통적인 스크린샷 \(p. 1189\)](#)

연결할 수 없는 인스턴스의 스크린샷을 얻는 방법

인스턴스가 실행 중이거나 인스턴스가 종단된 후에 스크린샷을 얻을 수 있습니다. 스크린샷의 데이터 전송 비용은 따로 들지 않습니다. 이미지는 JPG 형식으로 생성되며, 100KB보다 크지 않습니다. 인스턴스가 NVIDIA GRID 드라이버를 사용하고 있거나, 베어메탈 인스턴스(*.metal 유형의 인스턴스)에 있거나, ARM 기반 Graviton 또는 Graviton 2 프로세서로 구동되는 경우에는 이 기능이 지원되지 않습니다. 이 기능은 다음 리전에서 사용 가능합니다.

- 아시아 태평양(홍콩) 리전
- 아시아 태평양(도쿄) 리전
- 아시아 태평양(서울) 리전
- 아시아 태평양(싱가포르) 리전
- 아시아 태평양(시드니) 리전
- 아시아 태평양(뭄바이) 리전
- 미국 동부(버지니아 북부) 지역
- 미국 동부(오하이오) 리전
- 미국 서부(오레곤) 지역
- 미국 서부(캘리포니아 북부) 리전
- 유럽(아일랜드) 리전
- 유럽(프랑크푸르트) 리전
- 유럽(밀라노) 리전

- 유럽(런던) 리전
- 유럽(파리) 리전
- 유럽(스톡홀름) 리전
- 유럽(파리) 리전
- 남아메리카(상파울루) 리전
- 캐나다(중부) 리전
- 중동(바레인) 리전
- 아프리카(케이프타운) 리전
- 중국(베이징) 리전
- 중국(닝샤) 리전

콘솔을 사용하여 실행 중인 인스턴스의 스크린샷을 가져오려면

1. <https://console.aws.amazon.com/ec2>에서 Amazon EC2 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 인스턴스를 선택합니다.
3. 캡처할 인스턴스를 선택합니다.
4. 작업, 인스턴스 설정을 차례로 선택합니다.
5. 인스턴스 스크린샷 가져오기를 선택합니다. 다운로드할 이미지를 마우스 오른쪽 버튼으로 클릭하고 저장합니다.

명령줄을 사용하여 실행 중인 인스턴스의 스크린샷을 가져오려면

다음 명령 중 하나를 사용할 수 있습니다. 반환되는 출력은 base64-encoded입니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon EC2에 액세스 \(p. 3\)](#) 단원을 참조하십시오.

- [get-console-screenshot\(AWS CLI\)](#)
- [GetConsoleScreenshot\(Amazon EC2 Query API\)](#)

API 호출의 경우 반환되는 내용은 base64-encoded입니다. 명령줄 도구의 경우 디코딩은 자동으로 수행됩니다.

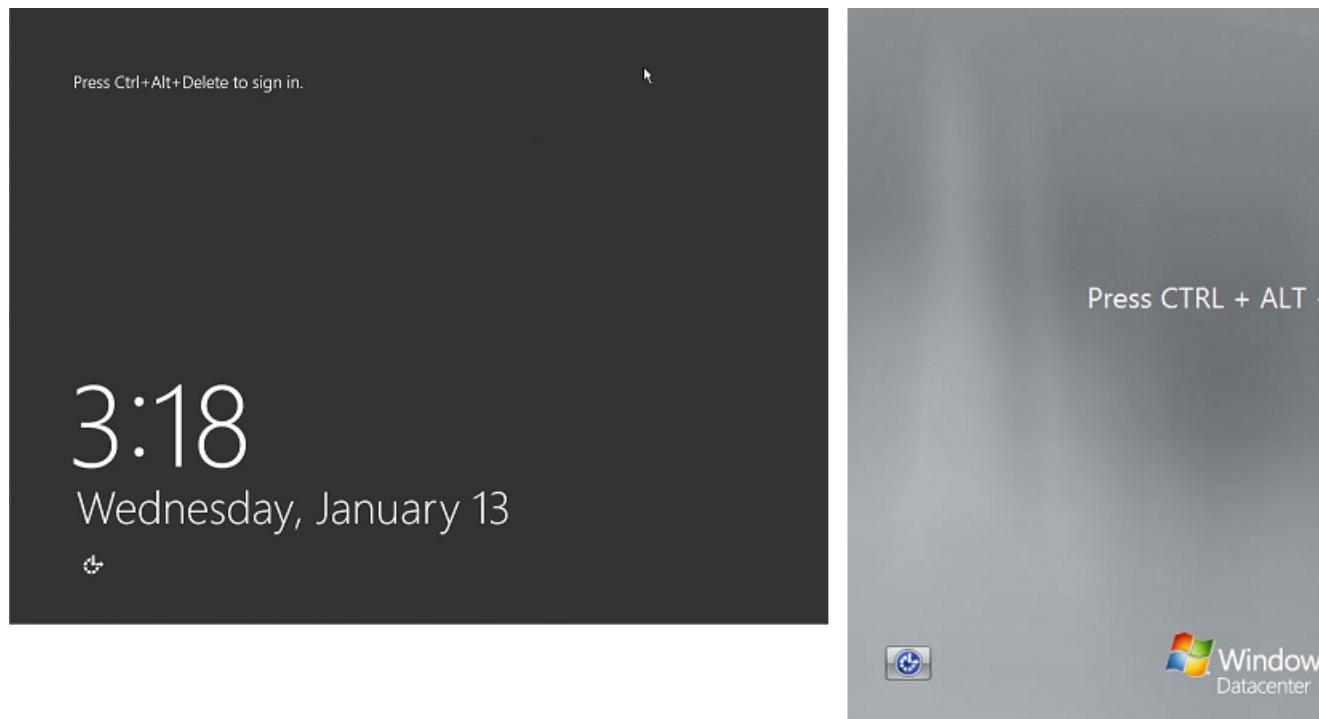
공통적인 스크린샷

다음 정보를 사용하면 서비스에서 반환되는 스크린샷을 바탕으로 연결할 수 없는 인스턴스의 문제 해결에 도움을 받을 수 있습니다.

- [로그온 화면\(Ctrl+Alt+Delete\) \(p. 1189\)](#)
- [복구 콘솔 화면 \(p. 1192\)](#)
- [Windows 부팅 관리자 화면 \(p. 1193\)](#)
- [Sysprep 화면 \(p. 1194\)](#)
- [준비 화면 \(p. 1194\)](#)
- [Windows 업데이트 화면 \(p. 1195\)](#)
- [Chkdsk \(p. 1196\)](#)

로그온 화면(Ctrl+Alt+Delete)

콘솔 스크린샷 서비스에서 다음을 반환했습니다.



인스턴스가 로그온 중에 연결할 수 없게 되면 네트워크 구성 또는 Windows 원격 데스크톱 서비스에 문제가 있을 수 있습니다. 프로세스에서 대량의 CPU를 사용 중인 경우 인스턴스가 응답하지 않을 수도 있습니다.

네트워크 구성

다음 정보를 사용하여 AWS, Microsoft Windows와 로컬(또는 온프레미스) 네트워크 구성이 인스턴스에 대한 액세스를 차단하고 있지 않은지 확인합니다.

AWS 네트워크 구성

구성	확인
보안 그룹 구성	포트 3389가 보안 그룹으로 열려 있는지 확인합니다. 올바른 퍼블릭 IP 주소에 연결 중인지 확인합니다. 인스턴스가 탄력적 IP와 연결되지 않은 경우, 퍼블릭 IP는 인스턴스가 중지/시작된 후 바뀝니다. 자세한 내용은 원격 데스크톱으로 원격 컴퓨터에 연결할 수 없음 (p. 1181) 단원을 참조하십시오.
VPC 구성(네트워크 ACL)	Amazon VPC에 대한 ACL(액세스 제어 목록)이 액세스를 차단하고 있지 않은지 확인합니다. 자세한 내용은 Amazon VPC 사용 설명서의 네트워크 ACL 을 참조하십시오.
VPN 구성	가상 프라이빗 네트워크(VPN)를 사용하여 VPC에 연결 중인 경우 VPN 터널 연결을 확인하십시오. 자세한 내용은 Amazon VPC에 대한 VPN 터널 연결의 문제 해결 방법 단원을 참조하십시오.

Windows 네트워크 구성

구성	확인
Windows 방화벽	Windows 방화벽이 인스턴스에 대한 연결을 차단하고 있지 않은지 확인하십시오. 원격 데스크톱 문제 해결 섹션 원격 데스크톱으로 원격 컴퓨터에 연결할 수 없음 (p. 1181) 의 7번째 글머리 기호 항목에 설명되어 있는 것처럼 Windows 방화벽을 비활성화합니다.
고급 TCP/IP 구성(정적 IP 사용)	정적 IP 주소를 구성했기 때문에 인스턴스가 응답하지 않을 수도 있습니다. VPC의 경우 네트워크 인터페이스를 생성 (p. 742) 하고 인스턴스에 연결 (p. 744) 합니다. EC2 Classic의 경우 DHCP를 활성화합니다.

로컬 또는 온프레미스 네트워크 구성

로컬 네트워크 구성이 액세스를 차단하고 있지 않은지 확인합니다. 연결할 수 없는 인스턴스와 똑같은 VPC에서 다른 인스턴스에 연결해 보십시오. 다른 인스턴스에 액세스할 수 없는 경우 로컬 네트워크 관리자와 협력하여 로컬 정책이 액세스를 제한하고 있는지 확인합니다.

원격 데스크톱 서비스 문제

인스턴스가 로그온 중에 연결할 수 없게 되면 해당 인스턴스에 원격 데스크톱 서비스(RDS) 문제가 있을 수 있습니다.

원격 데스크톱 서비스 구성

구성	확인
RDS가 실행 중임	인스턴스에서 RDS가 실행 중인지 확인합니다. Microsoft Management Console(MMC) 서비스 스냅인(<code>services.msc</code>)을 사용하여 인스턴스에 연결합니다. 서비스 목록에서 원격 데스크톱 서비스가 실행인지 확인합니다. 그렇지 않은 경우 이 서비스를 시작한 다음 시작 유형을 자동으로 설정합니다. 서비스 스냅인을 사용하여 인스턴스에 연결할 수 없으면 인스턴스에서 루트 볼륨을 분리하고, 볼륨의 스냅샷을 생성하거나 볼륨에서 AMI를 만들고, 원본 볼륨을 보조 볼륨과 동일한 가용 영역에 있는 다른 인스턴스에 연결하고, 시작 레지스트리 키를 수정합니다 . 위 작업을 마쳤으면 루트 볼륨을 원본 인스턴스에 다시 연결합니다. 볼륨 분리에 대한 자세한 내용은 Windows 인스턴스에서 Amazon EBS 볼륨 분리 (p. 964) 단원을 참조하십시오.
RDS가 활성화됨	서비스가 시작되더라도 비활성화될 수 있습니다. 원격 레지스트리를 사용하여 EC2 인스턴스에서 원격 데스크톱 활성화 (p. 1187) 에 설명되어 있는 것처럼 인스턴스에서 루트 볼륨을 분리하고, 볼륨의 스냅샷을 생성하거나 볼륨에서 AMI를 만들고, 원본 볼륨을 보조 볼륨과 동일한 가용 영역에 있는 다른 인스턴스에 연결하고, Terminal Server(터미널 서버) 레지스트리 키를 수정하여 서비스를 활성화합니다. 위 작업을 마쳤으면 루트 볼륨을 원본 인스턴스에 다시 연결합니다. 자세한 내용은 Windows 인스턴스에서 Amazon EBS 볼륨 분리 (p. 964) 단원을 참조하십시오.

높은 CPU 사용률

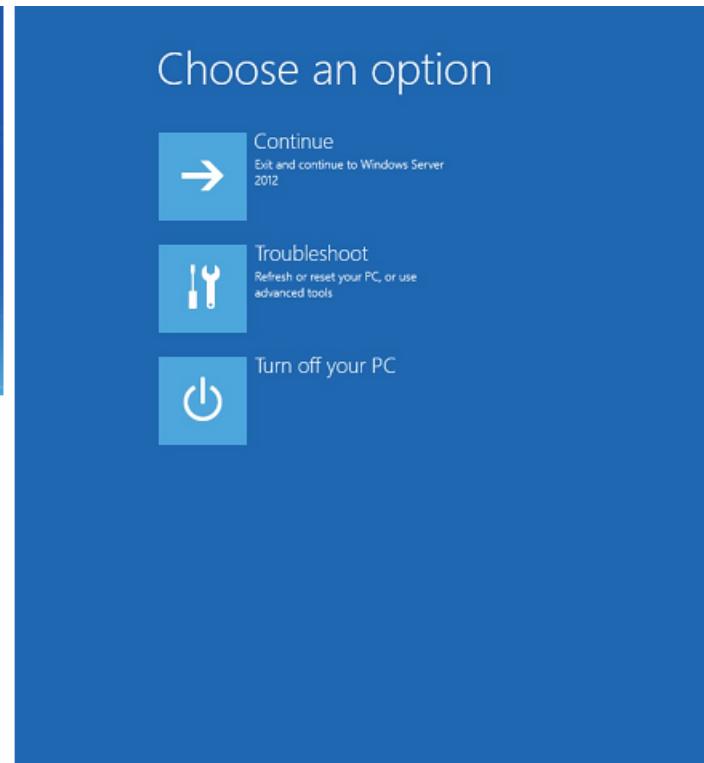
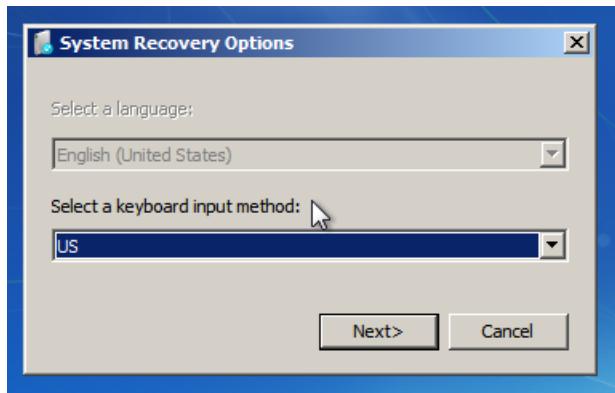
Amazon CloudWatch를 사용하여 인스턴스에서 CPUUtilization (Maximum) 측정치를 확인합니다. CPUUtilization (Maximum) 수치가 높으면 CPU 사용률이 낮아지기를 기다렸다가 다시 연결해 보십시오. 다음과 같은 원인으로 CPU 사용률이 높아질 수 있습니다.

- Windows 업데이트
- 보안 소프트웨어 검사
- 사용자 지정 시작 스크립트
- 작업 스케줄러

자세한 내용은 Amazon CloudWatch 사용 설명서의 [특정 리소스에 대한 통계 얻기](#)를 참조하십시오. 추가적인 문제 해결 팁은 [Windows 시작 직후 높은 CPU 사용률 \(p. 1180\)](#) 단원을 참조하십시오.

복구 콘솔 화면

콘솔 스크린샷 서비스에서 다음을 반환했습니다.



bootstatuspolicy가 ignoreallfailures로 설정되어 있지 않은 경우 운영 체제가 복구 콘솔로 부팅되고 이 상태로 고착될 수 있습니다. 다음 절차에 따라 bootstatuspolicy 구성을 ignoreallfailures로 변경합니다.

기본적으로, ASW에서 제공하는 퍼블릭 Windows AMI에 대한 정책 구성은 ignoreallfailures로 설정됩니다.

1. 연결할 수 없는 인스턴스를 중지합니다.
2. 루트 볼륨의 스냅샷을 생성합니다. 루트 볼륨은 /dev/sda1로서 인스턴스에 연결됩니다.

연결할 수 없는 인스턴스에서 루트 볼륨을 분리하고, 볼륨의 스냅샷을 생성하거나 볼륨에서 AMI를 만들고, 이를 보조 볼륨과 동일한 가용 영역에 있는 다른 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에서 Amazon EBS 볼륨 분리 \(p. 964\)](#) 단원을 참조하십시오.

Warning

임시 인스턴스가 원본 인스턴스와 동일한 AMI를 기반으로 하는 경우 추가 단계를 수행해야 합니다. 그렇지 않으면 디스크 서명 충돌로 인해 루트 볼륨 복원 후 원본 인스턴스를 부팅할 수 없습니다. 또는 임시 인스턴스에 대한 다른 AMI를 선택합니다. 예를 들어 원본 인스턴스에서 Windows Server 2008 R2용 AMI를 사용할 경우 Windows Server 2012용 AMI를 사용하여 임시 인스턴스를 시작합니다. 동일한 AMI를 기반으로 임시 인스턴스를 생성해야 하는 경우 디스크 서명 충돌을 피하기 위해 [원격 데스크톱으로 원격 컴퓨터에 연결할 수 없음 \(p. 1181\)](#)의 6단계를 참조하십시오.

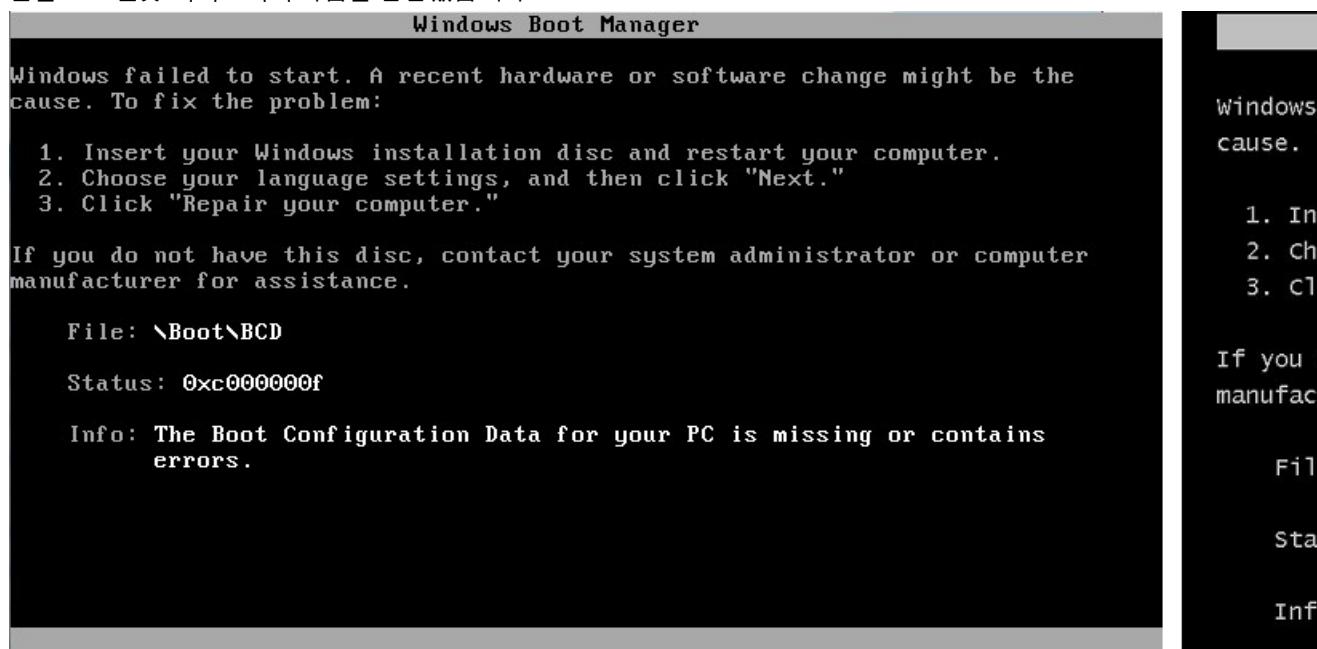
3. 인스턴스에 로그인하고 명령 프롬프트에서 다음 명령을 실행하여 bootstatuspolicy 구성은 ignoreallfailures로 변경합니다.

```
bcdeedit /store Drive Letter:\boot\bcd /set {default} bootstatuspolicy ignoreallfailures
```

4. 볼륨을 연결하지 못했던 인스턴스에 다시 연결하고 인스턴스를 다시 시작합니다.

Windows 부팅 관리자 화면

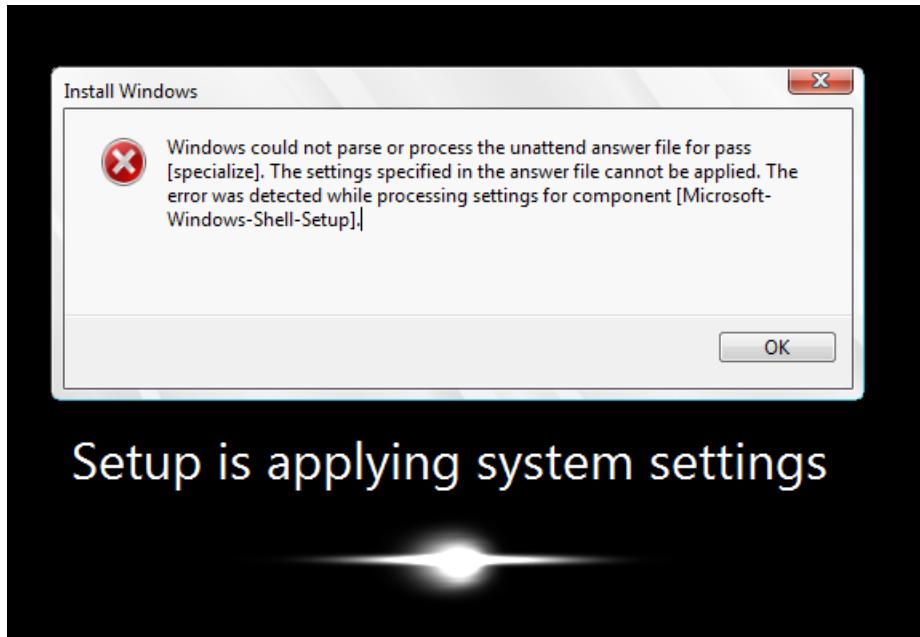
콘솔 스크린샷 서비스에서 다음을 반환했습니다.



운영 체제의 시스템 파일 및/또는 레지스트리에서 치명적 손상이 발생했습니다. 인스턴스가 이 상태로 고착되면 최근 백업 AMI에서 인스턴스를 복구하거나 대체 인스턴스를 시작해야 합니다. 인스턴스 상의 데이터에 액세스할 필요가 있는 경우에는 연결할 수 없는 인스턴스에서 모든 루트 볼륨을 분리하고, 이들 볼륨의 스냅샷을 생성하거나 볼륨에서 AMI를 만들고, 이들을 보조 볼륨과 동일한 가용 영역에 있는 다른 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에서 Amazon EBS 볼륨 분리 \(p. 964\)](#) 단원을 참조하십시오.

Sysprep 화면

콘솔 스크린샷 서비스에서 다음을 반환했습니다.



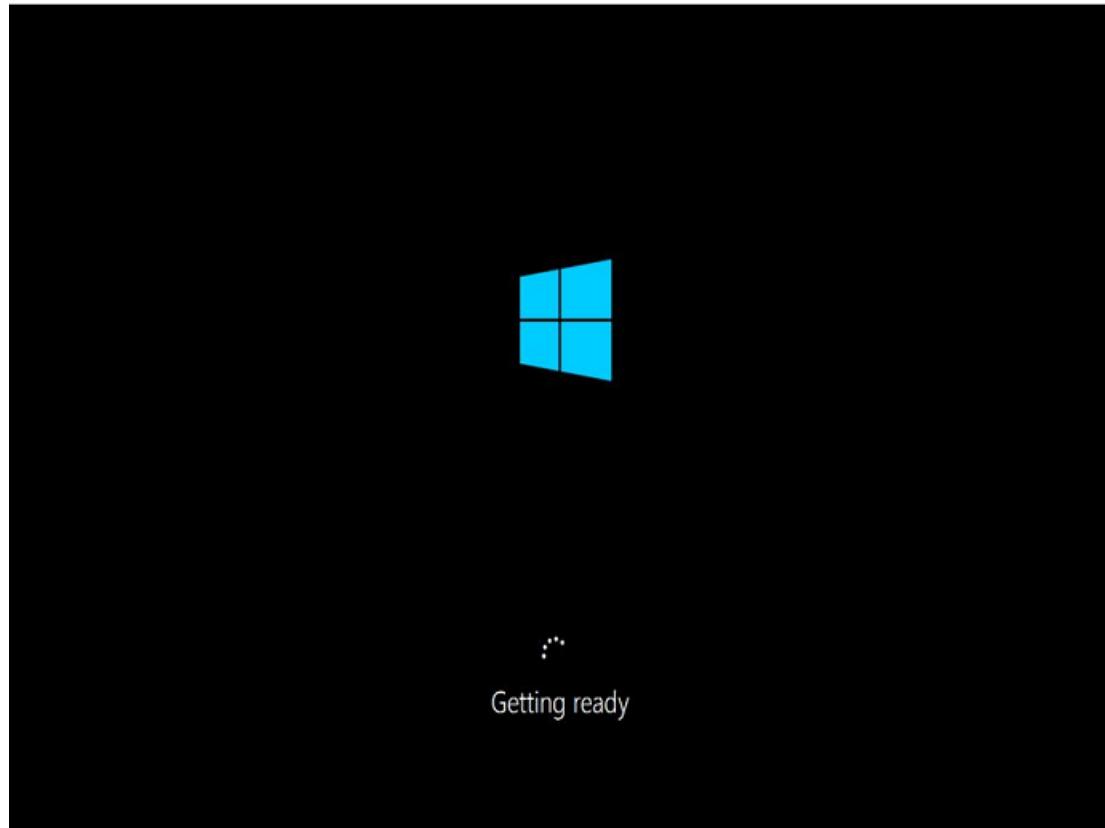
Setup is applying system settings



sysprep.exe를 호출하기 위해 EC2Config Service를 사용하지 않았거나 Sysprep 실행 중에 운영 체제에 장애가 발생한 경우 이 화면이 나타날 수 있습니다. 이 문제를 해결하려면 [Sysprep을 이용한 표준 Amazon 마신 이미지 생성 \(p. 87\)](#) 단원을 참조하십시오.

준비 화면

콘솔 스크린샷 서비스에서 다음을 반환했습니다.



인스턴스 콘솔 스크린샷 서비스를 반복적으로 새로 고쳐 진행률 링이 돌고 있는지 확인합니다. 링이 돌고 있으면 운영 체제가 시작할 때까지 기다립니다. Amazon CloudWatch를 사용하여 인스턴스에서 CPUUtilization (Maximum) 측정치를 확인하여 운영 체제가 활성 상태인지 알아볼 수도 있습니다. 진행률 링이 돌고 있지 않으면 인스턴스가 부팅 프로세스에서 더 진행되지 못하고 중단될 수 있습니다. 인스턴스를 재부팅합니다. 다시 부팅했는데도 이 문제가 해결되지 않으면 최근 백업 AMI에서 인스턴스를 복구하거나 대체 인스턴스를 시작하십시오. 인스턴스 상의 데이터에 액세스할 필요가 있는 경우에는 연결할 수 없는 인스턴스에서 루트 볼륨을 분리하고, 볼륨의 스냅샷을 생성하거나 볼륨에서 AMI를 만듭니다. 그런 다음, 보조 볼륨과 동일한 가용 영역에 있는 다른 인스턴스에 연결합니다.

Windows 업데이트 화면

콘솔 스크린샷 서비스에서 다음을 반환했습니다.



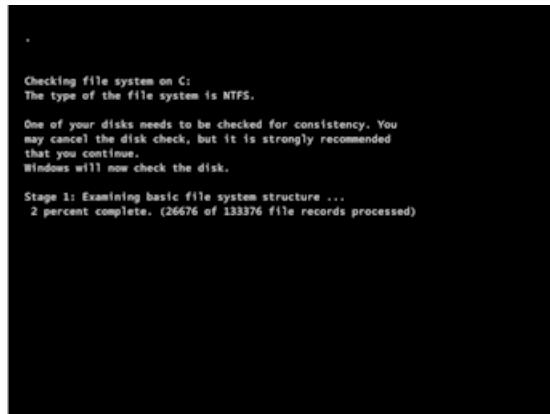
Windows 업데이트 프로세스에서는 레지스트리가 업데이트됩니다. 업데이트를 마칠 때까지 기다리십시오. 업데이트 중에 다시 부팅하거나 인스턴스를 중지하면 데이터가 손상될 수 있으므로 그렇게 하지 마십시오.

Note

Windows 업데이트 프로세스는 업데이트 중에 서버의 리소스를 소비할 수 있습니다. 이 문제가 자주 발생하면 더 빠른 인스턴스 유형과 더 빠른 EBS 볼륨의 사용을 고려해 보십시오.

Chkdsk

콘솔 스크린샷 서비스에서 다음을 반환했습니다.



Windows는 드라이브에서 chkdsk 시스템 도구를 실행하여 파일 시스템 레지스트리를 확인하고 논리적 파일 시스템 오류를 수정합니다. 프로세스가 완료될 때까지 기다립니다.

기억나지 않거나 만료된 Windows 관리자 암호 재설정

Windows 관리자 암호가 기억나지 않거나 만료되어 Windows Amazon EC2 인스턴스에 액세스할 수 없는 경우 암호를 재설정할 수 있습니다.

Note

로컬 관리자 암호를 재설정하는 데 필요한 수동 단계를 자동으로 적용하는 AWS 시스템 관리자 자동화 문서가 있습니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [Amazon EC2 인스턴스에서 암호 및 SSH 키 재설정](#) 단원을 참조하십시오.

관리자 암호를 수동으로 재설정하는 방식에서는 EC2Config 또는 EC2Launch를 사용합니다.

- Windows Server 2016 이전의 Windows AMI에 대해서는 EC2Config 서비스를 사용합니다.
- Windows Server 2016 이후 AMI에 대해서는 EC2Launch 서비스를 사용합니다.

이러한 절차는 인스턴스 생성에 사용한 키 페어를 분실한 경우 인스턴스에 연결할 수 있는 방법에 대해서도 설명합니다. Amazon EC2는 퍼블릭 키를 사용하여 암호 등의 데이터를 암호화하고, 프라이빗 키를 사용하여 데이터를 해독합니다. 퍼블릭 키와 프라이빗 키를 키 페어라고 합니다. Windows 인스턴스에서는 키 페어를 사용하여 관리자 암호를 가져오고 RDP를 사용하여 로그인합니다.

Note

인스턴스에서 로컬 관리자 계정을 비활성화했거나 인스턴스가 시스템 관리자에 대해 구성된 경우 EC2Rescue 및 Run Command를 사용하여 로컬 관리자 암호를 재활성화 및 재설정할 수도 있습니다. 자세한 내용은 [Systems Manager Run Command와 함께 EC2Rescue for Windows Server 사용](#)을 참조하십시오.

목차

- [EC2Config를 사용하여 Windows 관리자 암호 재설정 \(p. 1197\)](#)
- [EC2Launch를 사용하여 Windows 관리자 암호 재설정 \(p. 1201\)](#)

EC2Config를 사용하여 Windows 관리자 암호 재설정

Windows Server 2016 이전의 Windows AMI를 사용하고 있는 동안 Windows 관리자 암호를 잊은 경우 EC2Config 서비스를 사용하여 새 암호를 생성할 수 있습니다.

Windows Server 2016 이후 AMI를 사용한다면 [EC2Launch를 사용하여 Windows 관리자 암호 재설정 \(p. 1201\)](#) 단원을 참조하십시오.

Note

인스턴스에서 로컬 관리자 계정을 비활성화했거나 인스턴스가 시스템 관리자에 대해 구성된 경우 EC2Rescue 및 Run Command를 사용하여 로컬 관리자 암호를 재활성화 및 재설정할 수도 있습니다. 자세한 내용은 [Systems Manager Run Command와 함께 EC2Rescue for Windows Server 사용을 참조하십시오.](#)

Note

로컬 관리자 암호를 재설정하는 데 필요한 수동 단계를 자동으로 적용하는 AWS 시스템 관리자 자동화 문서가 있습니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [Amazon EC2 인스턴스에서 암호 및 SSH 키 재설정](#) 단원을 참조하십시오.

EC2Config를 사용하여 Windows 관리자 암호를 재설정하려면 다음 작업을 수행해야 합니다.

- [1단계: EC2Config 서비스가 실행 중인지 확인 \(p. 1197\)](#)
- [2단계: 인스턴스에서 루트 볼륨 분리 \(p. 1197\)](#)
- [3단계: 임시 인스턴스에 볼륨 연결 \(p. 1198\)](#)
- [4단계: 구성 파일 수정 \(p. 1199\)](#)
- [5단계: 원본 인스턴스 다시 시작 \(p. 1200\)](#)

1단계: EC2Config 서비스가 실행 중인지 확인

관리자 암호를 재설정하기 전에 EC2Config 서비스가 설치되어 실행 중인지 확인합니다. 이 단원의 뒷부분에서 EC2Config 서비스를 사용하여 관리자 암호를 재설정하게 됩니다.

EC2Config 서비스가 실행 중인지 확인하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택한 후 암호를 재설정할 인스턴스를 선택합니다. 이 절차에서는 이 인스턴스를 원본 인스턴스라고 합니다.
3. 작업, 인스턴스 설정, 시스템 로그 가져오기를 차례로 선택합니다.
4. EC2 에이전트 항목을 찾습니다(예: EC2 에이전트: Ec2Config 서비스 v3.18.1118). 이 항목이 보이면 EC2Config 서비스가 실행 중인 것입니다.

시스템 로그 출력이 비어 있거나 EC2Config 서비스가 실행 중이지 않을 경우 인스턴스 콘솔 스크린샷 서비스를 사용하여 인스턴스 문제를 해결합니다. 자세한 내용은 [연결할 수 없는 인스턴스 문제 해결 \(p. 1188\)](#) 단원을 참조하십시오.

2단계: 인스턴스에서 루트 볼륨 분리

암호가 저장된 볼륨이 인스턴스에 루트 볼륨으로 연결되어 있는 경우 EC2Config 서비스를 사용하여 관리자 암호를 재설정할 수 없습니다. 원본 인스턴스에서 볼륨을 분리해야 이 볼륨을 임시 인스턴스에 부 볼륨으로 연결할 수 있습니다.

인스턴스에서 루트 볼륨을 분리하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 암호 재설정이 필요한 인스턴스를 선택하고 작업, 인스턴스 상태, 중지를 선택합니다. 인스턴스 상태가 중지됨으로 변경된 후 이어서 다음 단계를 수행합니다.
4. (선택 사항) 이 인스턴스를 시작할 때 지정한 프라이빗 키가 있는 경우 다음 단계로 계속합니다. 그렇지 않으면 다음 단계를 사용하여 새 키 페어를 사용하여 시작한 새 인스턴스로 인스턴스를 바꿉니다.
 - a. Amazon EC2 콘솔을 사용하여 새 키 페어를 생성합니다. 새 키 페어에 분실한 프라이빗 키와 동일한 이름을 지정하려면 먼저 기존 키 페어를 삭제해야 합니다.
 - b. 바꿀 인스턴스를 선택합니다. 인스턴스의 인스턴스 유형, VPC, 서브넷, 보안 그룹 및 IAM 역할을 기록해둡니다.
 - c. 작업, 이미지, 이미지 생성을 차례로 선택합니다. 이미지의 이름과 설명을 입력하고 이미지 생성, 보류 중인 이미지 보기를 선택합니다. 이미지 상태가 사용 가능으로 변경된 후 이어서 다음 단계를 수행합니다.
 - d. 이미지를 선택하고 시작을 선택합니다.
 - e. 바꿀 인스턴스와 동일한 인스턴스 유형, VPC, 서브넷, 보안 그룹 및 IAM 역할을 선택한 후 시작을 선택하여 마법사를 완료합니다.
 - f. 메시지가 나타나면 새 인스턴스에 대해 생성한 키 페어를 선택하고 승인 확인란을 선택한 후 인스턴스 시작을 선택합니다.
 - g. (선택 사항) 원본 인스턴스에 연결된 탄력적 IP 주소가 있는 경우 이 주소를 새 인스턴스와 연결합니다. 원본 인스턴스에 루트 볼륨 외에도 EBS 볼륨이 있는 경우 해당 볼륨을 새 인스턴스로 전송합니다.
 - h. 더 이상 필요하지 않으므로 중지된 인스턴스를 종료합니다. 이 절차의 나머지 단계에서 원본 인스턴스에 대한 모든 참조는 방금 생성한 이 인스턴스에 적용됩니다.
5. 다음과 같은 방법으로 루트 볼륨을 원본 인스턴스에서 분리합니다.
 - a. 원본 인스턴스의 설명 창에서 루트 디바이스로 나열된 EBS 볼륨의 ID를 기록해둡니다.
 - b. 탐색 창에서 볼륨을 선택합니다.
 - c. 앞 단계에서 적어둔 볼륨을 볼륨 목록에서 선택한 다음 작업, Detach Volume(볼륨 분리)을 선택합니다. 볼륨 상태가 사용 가능으로 변경된 후 이어서 다음 단계를 수행합니다.

3단계: 임시 인스턴스에 볼륨 연결

그다음에는 임시 인스턴스를 시작하여 이 인스턴스에 볼륨을 부 볼륨으로 연결합니다. 이것은 구성 파일을 수정하는 데 사용하는 인스턴스입니다.

임시 인스턴스를 시작하고 볼륨을 연결하려면

1. 다음과 같이 임시 인스턴스를 시작합니다.
 - a. 탐색 창에서 인스턴스, 인스턴스 시작을 차례로 선택한 후 AMI를 선택합니다.

Important

디스크 서명 충돌을 방지하려면 다른 Windows 버전에 대한 AMI를 선택해야 합니다. 예를 들어, 원본 인스턴스에서 Windows Server 2012 R2를 실행하는 경우 Windows Server 2008 R2용 기본 AMI를 사용하여 임시 인스턴스를 시작합니다.

- b. 기본 인스턴스 유형을 그대로 두고 다음: 인스턴스 세부 정보 구성을 선택합니다.
 - c. 인스턴스 세부 정보 구성 페이지에서 서브넷에 대해 원래 인스턴스와 동일한 가용 영역을 선택한 다음 검토 및 시작을 선택합니다.

Important

임시 인스턴스는 원래 인스턴스와 동일한 가용 영역에 있어야 합니다. 임시 인스턴스가 다른 가용 영역에 있으면 원래 인스턴스의 루트 볼륨을 인스턴스에 연결할 수 없습니다.

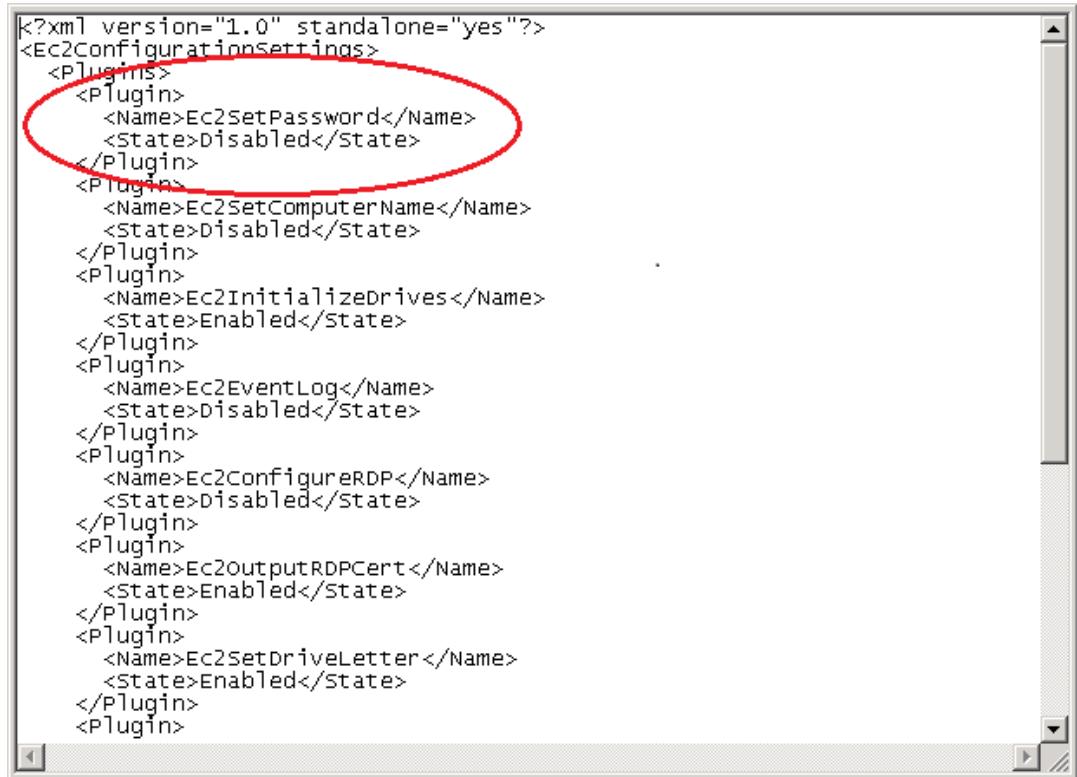
- d. 인스턴스 시작 검토 페이지에서 시작을 선택합니다.
 - e. 메시지가 나타나면 새 키 페어를 생성하고, 컴퓨터에서 안전한 위치에 다운로드한 후 인스턴스 시작을 선택합니다.
2. 다음과 같은 방법으로 볼륨을 임시 인스턴스에 부 볼륨으로 연결합니다.
- a. 탐색 창에서 볼륨을 선택하고 원본 인스턴스에서 분리한 볼륨을 선택한 후 작업, 볼륨 연결을 선택합니다.
 - b. 볼륨 연결 대화 상자에서 인스턴스에 임시 인스턴스의 이름이나 ID를 입력한 후 목록에서 해당 인스턴스를 선택합니다.
 - c. 디바이스에 대해 **xvdf**를 입력하고(아직 없는 경우) 연결을 선택합니다.

4단계: 구성 파일 수정

볼륨을 임시 인스턴스에 부 볼륨으로 연결한 후 구성 파일에서 `Ec2SetPassword` 플러그인을 수정합니다.

구성 파일을 수정하려면

1. 임시 인스턴스에서 부 볼륨에 있는 구성 파일을 다음과 같이 수정합니다.
 - a. 임시 인스턴스를 시작하여 이 인스턴스에 연결합니다.
 - b. Disk Management(디스크 관리) 유ти리티를 열고 [Amazon EBS 볼륨을 Windows에서 사용할 수 있도록 만들기](#) 지침에 따라 드라이브를 온라인 상태로 만듭니다.
 - c. 두 번째 볼륨으로 이동한 다음, 메모장과 같은 텍스트 편집기를 사용하여 `\Program Files\Amazon\Ec2ConfigService\Settings\config.xml`을 엽니다.
 - d. 스크린샷에 표시된 것처럼 파일 맨 위에서 이름이 `Ec2SetPassword`인 플러그인을 찾습니다. 상태를 `Disabled`에서 `Enabled`로 변경하고 파일을 저장합니다.



```
<?xml version="1.0" standalone="yes"?>
<Ec2ConfigurationSettings>
  <Plugins>
    <Plugin>
      <Name>Ec2SetPassword</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2SetComputerName</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2InitializeDrives</Name>
      <State>Enabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2EventLog</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2ConfigureRDP</Name>
      <State>Disabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2OutputRDPCert</Name>
      <State>Enabled</State>
    </Plugin>
    <Plugin>
      <Name>Ec2SetDriveLetter</Name>
      <State>Enabled</State>
    </Plugin>
  </Plugins>
</Ec2ConfigurationSettings>
```

2. 구성 파일을 수정한 후에 다음과 같이 임시 인스턴스에서 부 볼륨을 분리합니다.
 - a. Disk Management(디스크 관리) 유ти리티를 사용하여 볼륨을 오프라인으로 설정합니다.
 - b. 임시 인스턴스에서 연결을 해제하고 Amazon EC2 콘솔로 돌아갑니다.
 - c. 탐색 창에서 볼륨을 선택하고 해당 볼륨을 선택한 후 작업, Detach Volume(볼륨 분리)을 선택합니다. 볼륨의 상태가 사용 가능으로 변경되면 다음 단계로 계속합니다.

5단계: 원본 인스턴스 다시 시작

구성 파일을 수정한 후 볼륨을 원본 인스턴스에 루트 볼륨으로 다시 연결하고 키 페어를 사용해 인스턴스에 연결하여 관리자 암호를 검색합니다.

1. 다음과 같은 방법으로 볼륨을 원본 인스턴스에 다시 연결합니다.
 - a. 탐색 창에서 볼륨을 선택하고 임시 인스턴스에서 분리한 볼륨을 선택한 후 작업, 볼륨 연결을 선택합니다.
 - b. 볼륨 연결 대화 상자에서 인스턴스에 원본 인스턴스의 이름이나 ID를 입력한 다음 해당 인스턴스를 선택합니다.
 - c. 디바이스에 **/dev/sda1**을 입력합니다.
 - d. 연결을 선택합니다. 볼륨 상태가 **in-use**로 변경된 후 이어서 다음 단계를 수행합니다.
2. 탐색 창에서 인스턴스를 선택합니다. 원본 인스턴스를 선택하고 작업, 인스턴스 상태, 시작을 선택합니다. 확인 메시지가 나타나면 예, 시작을 선택합니다. 인스턴스 상태가 **running**로 변경된 후 이어서 다음 단계를 수행합니다.
3. 새 키 페어의 프라이빗 키를 사용하여 새 Windows 관리자 암호를 가져온 다음 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.

Important

인스턴스를 중지했다가 시작하면 인스턴스가 새 퍼블릭 IP 주소를 가져옵니다. 현재 퍼블릭 DNS 이름을 사용하여 인스턴스에 연결해야 합니다. 자세한 내용은 [인스턴스 수명 주기 \(p. 371\)](#) 단원을 참조하십시오.

4. (선택 사항) 임시 인스턴스를 더 이상 사용하지 않는 경우 해당 인스턴스는 종료해도 됩니다. 임시 인스턴스를 선택하고 작업, 인스턴스 상태, 종료를 선택합니다.

EC2Launch를 사용하여 Windows 관리자 암호 재설정

Windows Server 2016 이후 AMI를 사용하는 동안 Windows 관리자 암호를 잊어버렸다면, EC2Launch를 이용하여 새 암호를 생성할 수 있는 EC2Rescue 도구를 사용할 수 있습니다.

Windows Server 2016 이전의 Windows Server AMI를 사용하는 경우 [EC2Config를 사용하여 Windows 관리자 암호 재설정 \(p. 1197\)](#) 단원을 참조하십시오.

Warning

인스턴스를 중지하면 인스턴스 스토어 볼륨의 데이터가 삭제됩니다. 인스턴스 스토어 볼륨의 데이터를 유지하려면 영구 스토리지에 백업하십시오.

Note

인스턴스에서 로컬 관리자 계정을 비활성화했거나 인스턴스가 시스템 관리자에 대해 구성된 경우 EC2Rescue 및 Run Command를 사용하여 로컬 관리자 암호를 재활성화 및 재설정할 수도 있습니다. 자세한 내용은 [Systems Manager Run Command와 함께 EC2Rescue for Windows Server 사용](#)을 참조하십시오.

Note

로컬 관리자 암호를 재설정하는 데 필요한 수동 단계를 자동으로 적용하는 AWS 시스템 관리자 자동화 문서가 있습니다. 자세한 내용은 AWS 시스템 관리자 사용 설명서의 [Amazon EC2 인스턴스에서 암호 및 SSH 키 재설정](#) 단원을 참조하십시오.

EC2Launch를 사용하여 Windows 관리자 암호를 재설정하려면 다음 작업을 수행해야 합니다.

- 1단계: 인스턴스에서 루트 볼륨 분리 (p. 1201)
- 2단계: 임시 인스턴스에 볼륨 연결 (p. 1202)
- 3단계: 관리자 암호 재설정 (p. 1203)
- 4단계: 원본 인스턴스 다시 시작 (p. 1203)

1단계: 인스턴스에서 루트 볼륨 분리

암호가 저장된 볼륨이 인스턴스에 루트 볼륨으로 연결되어 있는 경우 EC2Launch 서비스를 사용하여 관리자 암호를 재설정할 수 없습니다. 원본 인스턴스에서 볼륨을 분리해야 이 볼륨을 임시 인스턴스에 부 볼륨으로 연결할 수 있습니다.

인스턴스에서 루트 볼륨을 분리하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 암호 재설정이 필요한 인스턴스를 선택하고 작업, 인스턴스 상태, 종지를 선택합니다. 인스턴스 상태가 중지됨으로 변경된 후 이어서 다음 단계를 수행합니다.
4. (선택 사항) 이 인스턴스를 시작할 때 지정한 프라이빗 키가 있는 경우 다음 단계로 계속합니다. 그렇지 않으면 다음 단계를 사용하여 새 키 페어를 사용하여 시작한 새 인스턴스로 인스턴스를 바꿉니다.

- a. Amazon EC2 콘솔을 사용하여 새 키 페어를 생성합니다. 새 키 페어에 분실한 프라이빗 키와 동일한 이름을 지정하려면 먼저 기존 키 페어를 삭제해야 합니다.
 - b. 바꿀 인스턴스를 선택합니다. 인스턴스의 인스턴스 유형, VPC, 서브넷, 보안 그룹 및 IAM 역할을 기록해둡니다.
 - c. 작업, 이미지, 이미지 생성을 차례로 선택합니다. 이미지의 이름과 설명을 입력하고 이미지 생성, 보류 중인 이미지 보기 를 선택합니다. 이미지 상태가 사용 가능으로 변경된 후 이어서 다음 단계를 수행합니다.
 - d. 이미지를 선택하고 시작을 선택합니다.
 - e. 바꿀 인스턴스와 동일한 인스턴스 유형, VPC, 서브넷, 보안 그룹 및 IAM 역할을 선택한 후 시작을 선택하여 마법사를 완료합니다.
 - f. 메시지가 나타나면 새 인스턴스에 대해 생성한 키 페어를 선택하고 승인 확인란을 선택한 후 인스턴스 시작을 선택합니다.
 - g. (선택 사항) 원본 인스턴스에 연결된 탄력적 IP 주소가 있는 경우 이 주소를 새 인스턴스와 연결합니다. 원본 인스턴스에 루트 볼륨 외에도 EBS 볼륨이 있는 경우 해당 볼륨을 새 인스턴스로 전송합니다.
 - h. 더 이상 필요하지 않으므로 중지된 인스턴스를 종료합니다. 이 절차의 나머지 단계에서 원본 인스턴스에 대한 모든 참조는 방금 생성한 이 인스턴스에 적용됩니다.
5. 다음과 같은 방법으로 루트 볼륨을 원본 인스턴스에서 분리합니다.
- a. 원본 인스턴스의 설명 창에서 루트 디바이스로 나열된 EBS 볼륨의 ID를 기록해둡니다.
 - b. 탐색 창에서 볼륨을 선택합니다.
 - c. 앞 단계에서 적어둔 볼륨을 볼륨 목록에서 선택한 다음 작업, Detach Volume(볼륨 분리)을 선택합니다. 볼륨 상태가 사용 가능으로 변경된 후 이어서 다음 단계를 수행합니다.

2단계: 임시 인스턴스에 볼륨 연결

그다음에는 임시 인스턴스를 시작하여 이 인스턴스에 볼륨을 부 볼륨으로 연결합니다. 이것은 EC2Launch를 실행하는 데 사용하는 인스턴스입니다.

임시 인스턴스를 시작하고 볼륨을 연결하려면

1. 다음과 같이 임시 인스턴스를 시작합니다.
 - a. 탐색 창에서 인스턴스, 인스턴스 시작을 차례로 선택한 후 AMI를 선택합니다.
- Important**
- 디스크 서명 충돌을 방지하려면 다른 Windows 버전에 대한 AMI를 선택해야 합니다. 예를 들어, 원본 인스턴스에서 Windows Server 2012 R2를 실행하는 경우 Windows Server 2008 R2용 기본 AMI를 사용하여 임시 인스턴스를 시작합니다.
- b. 기본 인스턴스 유형을 그대로 두고 다음: 인스턴스 세부 정보 구성 을 선택합니다.
 - c. 인스턴스 세부 정보 구성 페이지에서 서브넷에 대해 원래 인스턴스와 동일한 가용 영역을 선택한 다음 검토 및 시작 을 선택합니다.

Important

임시 인스턴스는 원래 인스턴스와 동일한 가용 영역에 있어야 합니다. 임시 인스턴스가 다른 가용 영역에 있으면 원래 인스턴스의 루트 볼륨을 인스턴스에 연결할 수 없습니다.

- d. 인스턴스 시작 검토 페이지에서 시작을 선택합니다.
 - e. 메시지가 나타나면 새 키 페어를 생성하고, 컴퓨터에서 안전한 위치에 다운로드한 후 인스턴스 시작을 선택합니다.
2. 다음과 같은 방법으로 볼륨을 임시 인스턴스에 부 볼륨으로 연결합니다.

- a. 탐색 창에서 볼륨을 선택하고 원본 인스턴스에서 분리한 볼륨을 선택한 후 작업, 볼륨 연결을 선택합니다.
- b. 볼륨 연결 대화 상자에서 인스턴스에 임시 인스턴스의 이름이나 ID를 입력한 후 목록에서 해당 인스턴스를 선택합니다.
- c. 디바이스에 대해 **xvdf**를 입력하고(아직 없는 경우) 연결을 선택합니다.

3단계: 관리자 암호 재설정

그다음에는 임시 인스턴스에 연결한 후 EC2Launch를 사용하여 관리자 암호를 재설정합니다.

관리자 암호를 재설정하려면

1. 임시 인스턴스에 연결하고 인스턴스에서 EC2Rescue for Windows Server 도구를 사용하여 관리자 암호를 다음과 같이 재설정합니다.
 - a. [EC2Rescue for Windows Server](#) zip 파일을 다운로드하여 압축을 푼 후 EC2Rescue.exe를 실행합니다.
 - b. 라이선스 계약에서 라이선스 계약을 읽고, 약관에 동의하는 경우 I Agree(동의함)를 선택합니다.
 - c. EC2Rescue for Windows Server 시작 화면에서 다음을 선택합니다.
 - d. Select mode(모드 선택)에서 Offline instance(오프라인 인스턴스)를 선택합니다.
 - e. Select a disk(디스크 선택) 화면에서 xvdf 디바이스를 선택하고 다음을 선택합니다.
 - f. 디스크 선택을 확인한 후 예를 선택합니다.
 - g. 볼륨 로드가 완료되면 확인을 선택합니다.
 - h. Select Offline Instance Option(오프라인 인스턴스 옵션 선택)에서 Diagnose and Rescue(진단 및 복구)를 선택합니다.
 - i. 요약 화면에서 정보를 검토한 다음 다음을 선택합니다.
 - j. Detected possible issues(감지된 잠재적 문제) 화면에서 Reset Administrator Password(관리자 암호 재설정)를 선택하고 다음을 선택합니다.
 - k. 확인 화면에서 Rescue(복구), 확인을 선택합니다.
 - l. 완료 화면에서 완료를 선택합니다.
 - m. EC2Rescue for Windows Server 도구를 종료하고 임시 인스턴스에서 연결을 해제한 후 Amazon EC2 콘솔로 돌아갑니다.
2. 다음과 같은 방법으로 부(xvdf) 볼륨을 임시 인스턴스에서 분리합니다.
 - a. 탐색 창에서 인스턴스를 선택하고 임시 인스턴스를 선택합니다.
 - b. 임시 인스턴스의 설명 창에서 xvdf로 나열된 EBS 볼륨의 ID를 기록해둡니다.
 - c. 탐색 창에서 볼륨을 선택합니다.
 - d. 앞 단계에서 적어둔 볼륨을 볼륨 목록에서 선택한 다음 작업, Detach Volume(볼륨 분리)을 선택합니다. 볼륨 상태가 사용 가능로 변경된 후 이어서 다음 단계를 수행합니다.

4단계: 원본 인스턴스 다시 시작

EC2Launch를 사용해 관리자 암호를 재설정한 후 볼륨을 원본 인스턴스에 루트 볼륨으로 다시 연결하고 키 페어를 사용해 인스턴스에 연결하여 관리자 암호를 검색합니다.

원본 인스턴스를 다시 시작하려면

1. 다음과 같은 방법으로 볼륨을 원본 인스턴스에 다시 연결합니다.
 - a. 탐색 창에서 볼륨을 선택하고 임시 인스턴스에서 분리한 볼륨을 선택한 후 작업, 볼륨 연결을 선택합니다.

- b. 볼륨 연결 대화 상자에서 인스턴스에 원본 인스턴스의 이름이나 ID를 입력한 다음 해당 인스턴스를 선택합니다.
- c. 디바이스에 **/dev/sda1**을 입력합니다.
- d. 연결을 선택합니다. 볼륨 상태가 **in-use**로 변경된 후 이어서 다음 단계를 수행합니다.
2. 탐색 창에서 인스턴스를 선택합니다. 원본 인스턴스를 선택하고 작업, 인스턴스 상태, 시작을 선택합니다. 확인 메시지가 나타나면 예, 시작을 선택합니다. 인스턴스 상태가 **running**로 변경된 후 이어서 다음 단계를 수행합니다.
3. 새 키 페어의 프라이빗 키를 사용하여 새 Windows 관리자 암호를 가져온 다음 인스턴스에 연결합니다. 자세한 내용은 [Windows 인스턴스에 연결 \(p. 436\)](#) 단원을 참조하십시오.
4. (선택 사항) 임시 인스턴스를 더 이상 사용하지 않는 경우 해당 인스턴스는 종료해도 됩니다. 임시 인스턴스를 선택하고 작업, 인스턴스 상태, 종료를 선택합니다.

인스턴스 중지 문제 해결

Amazon EBS 인스턴스를 중지한 후 이 인스턴스가 **stopping** 상태로 멈춰 있는 것 같이 보일 경우 기본 호스트 컴퓨터에 문제가 있을 수 있습니다.

인스턴스가 **running** 상태에 있지 않은 동안에는 인스턴스 사용 요금이 부과되지 않습니다.

인스턴스의 콘솔 또는 AWS CLI 사용을 강제로 중단합니다.

- 인스턴스의 콘솔 사용을 강제로 중단하려면 중지된 인스턴스를 선택하고 작업, 인스턴스 상태, 종지, 예, 강제 중지를 선택합니다.
- 인스턴스의 AWS CLI 사용을 강제로 중단하려면 **stop-instances** 명령과 다음과 같은 **--force** 옵션을 사용합니다.

```
aws ec2 stop-instances --instance-ids i-0123ab456c789d01e --force
```

10분 후에도 인스턴스가 중지되지 않는 경우 [Amazon EC2 forum](#)에 도움을 요청하는 글을 게시합니다. 해결 방법을 신속히 찾아내려면 인스턴스 ID를 포함하고 자신이 이미 수행했던 단계에 대해 설명하십시오. 지원 플랜이 있는 경우에는 [지원 센터](#)에서 기술 지원 사례를 요청할 수 있습니다.

대체 인스턴스 생성

[Amazon EC2 forum](#) 또는 [지원 센터](#)의 지원을 기다리는 동안 문제 해결을 시도하려면 대체 인스턴스를 생성합니다. 중지된 인스턴스의 AMI를 생성하고 새로운 AMI를 사용하여 새 인스턴스를 시작합니다.

콘솔을 사용하여 대체 인스턴스를 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택하고 멈춘 인스턴스를 선택합니다.
3. 작업, 이미지, 이미지 생성을 차례로 선택합니다.
4. 이미지 생성 대화 상자에서 다음 필드를 채운 다음 이미지 생성을 선택합니다.
 - a. AMI에 대한 이름 및 설명을 지정합니다.
 - b. 재부팅 안 함을 선택합니다.

자세한 내용은 [실행 중인 인스턴스에서 Windows AMI 생성 \(p. 85\)](#) 단원을 참조하십시오.

5. AMI에서 새로운 인스턴스를 시작하고 새로운 인스턴스가 작동하는지 확인합니다.
6. 멈춰 있는 인스턴스를 선택하고 작업, 인스턴스 상태, 종료를 차례로 선택합니다. 또한 인스턴스가 종료 중 상태로 멈추는 경우 Amazon EC2에서 몇 시간 내에 해당 인스턴스를 자동으로 종료합니다.

CLI를 사용하여 대체 인스턴스를 생성하려면

1. [create-image](#)(AWS CLI) 명령 및 다음 --no-reboot 옵션을 사용하여 멈춰 있는 인스턴스에서 AMI를 생성합니다.

```
aws ec2 create-image --instance-id i-0123ab456c789d01e --name "AMI" --description "AMI for replacement instance" --no-reboot
```

2. [run-instances](#)(AWS CLI) 명령을 다음과 같이 사용하여 AMI에서 새 인스턴스를 시작합니다.

```
aws ec2 run-instances --image-id ami-1a2b3c4d --count 1 --instance-type c3.large --key-name MyKeyPair --security-groups MySecurityGroup
```

3. 새로운 인스턴스가 작동 중인지 확인합니다.
4. [terminate-instances](#)(AWS CLI) 명령을 다음과 같이 멈춰 있는 인스턴스를 종료합니다.

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0
```

이전 절차에 설명된 대로 AMI를 만들 수 없으면 다음과 같이 대체 인스턴스를 설정할 수 있습니다.

(대안) 콘솔을 사용하여 대체 인스턴스를 생성하려면

1. 인스턴스를 선택하고 설명, 블록 디바이스를 선택합니다. 각 볼륨을 선택하고 볼륨 ID를 적습니다. 어느 볼륨이 루트 볼륨인지 적어두어야 합니다.
2. 탐색 창에서 볼륨을 선택합니다. 인스턴스에 해당하는 각 볼륨을 선택하고 작업, 스냅샷 생성을 차례로 선택합니다.
3. 탐색 창에서 스냅샷을 선택합니다. 방금 만든 스냅샷을 선택한 후 작업, 볼륨 생성을 선택합니다.
4. 멈춰 있는 인스턴스와 동일한 운영 체제에서 인스턴스를 시작합니다. 루트 볼륨의 볼륨 ID와 디바이스 이름을 적어둡니다.
5. 탐색 창에서 인스턴스를 선택하고 방금 시작한 인스턴스를 선택한 후, 작업, 인스턴스 상태를 차례로 선택하고 중지를 선택합니다.
6. 탐색 창에서 볼륨을 선택하고 중지된 인스턴스의 루트 볼륨을 선택한 후, 작업, 볼륨 분리를 선택합니다.
7. 멈춰 있는 인스턴스에서 만든 루트 볼륨을 선택하고 작업, 볼륨 연결을 선택한 후, 이 볼륨을 새 인스턴스에 루트 볼륨으로 연결합니다(기록해 놓은 디바이스 이름 사용). 루트 이외의 다른 추가 볼륨을 인스턴스에 연결합니다.
8. 탐색 창에서 인스턴스를 선택하고 대체 인스턴스를 선택합니다. 작업, 인스턴스 상태, 시작을 차례로 선택합니다. 인스턴스가 작동 중인지 확인합니다.
9. 멈춰 있는 인스턴스를 선택하고 작업, 인스턴스 상태, 종료를 차례로 선택합니다. 또한 인스턴스가 종료 중 상태로 멈추는 경우 Amazon EC2에서 몇 시간 내에 해당 인스턴스를 자동으로 종료합니다.

인스턴스 종료 문제 해결

인스턴스가 `running` 상태에 있지 않은 동안에는 인스턴스 사용 요금이 부과되지 않습니다. 다시 말해서, 인스턴스를 종료할 때 인스턴스의 상태가 `shutting-down`으로 변경되는 즉시 해당 인스턴스에 대한 요금 발생이 중지되는 것입니다.

지연된 인스턴스 종료

인스턴스가 몇 분 이상 `shutting-down` 상태로 유지되는 경우 인스턴스에 의해 실행 중인 종료 스크립트로 인한 지연이 발생했을 수 있습니다.

또 한 가지 예상 원인은 기본 호스트 컴퓨터 관련 문제입니다. 인스턴스가 몇 시간 동안 *shutting-down* 상태로 유지되는 경우 Amazon EC2는 해당 인스턴스를 멈춰 있는 인스턴스로 간주하여 강제로 종료합니다.

인스턴스가 종료 중 상태로 멈춰 있는 것처럼 보이며 이 상태로 몇 시간 이상이 경과된 경우 [Amazon EC2 forum](#)에 도움을 요청하는 글을 게시하십시오. 해결 방법을 신속히 찾아내려면 인스턴스 ID를 포함하고 자신이 이미 수행했던 단계에 대해 설명하십시오. 지원 플랜이 있는 경우에는 [지원 센터](#)에서 기술 지원 사례를 요청할 수 있습니다.

종료된 인스턴스가 계속 표시됨

인스턴스를 종료한 후에도 인스턴스는 잠깐 동안 콘솔에서 표시된 후 삭제됩니다. 상태가 `terminated`로 표시됩니다. 몇 시간이 지난 후에도 해당 항목이 삭제되지 않으면 Support에 문의하십시오.

인스턴스가 자동으로 시작되거나 종료됨

일반적으로 다음과 같은 동작은 사용자가 Amazon EC2 Auto Scaling 또는 EC2 집합을 사용하여 정의한 기준에 따라 컴퓨팅 리소스의 규모를 자동으로 확장/축소해 왔음을 의미합니다.

- 인스턴스를 종료하면 새 인스턴스가 자동으로 시작됩니다.
- 인스턴스를 시작하면 인스턴스 중 하나가 자동으로 종료됩니다.
- 인스턴스를 종지하면 인스턴스가 종료되고 새 인스턴스가 자동으로 시작됩니다.

자동 조정을 중지하려면 [Amazon EC2 Auto Scaling 사용 설명서](#) 또는 [EC2 집합을 사용하여 인스턴스 시작 \(p. 400\)](#) 단원을 참조하세요.

사용 EC2Rescue for Windows Server

EC2Rescue for Windows Server는 가능한 문제를 진단하고 해결하기 위해 Amazon EC2 Windows Server 인스턴스에서 실행하는 편리한 도구입니다. 이 도구는 로그 파일을 수집하고 문제를 해결하며, 문제가 발생할 가능성이 있는 영역을 사전에 찾아내는 데 매우 유용하게 사용할 수 있습니다. 또한 다른 인스턴스의 Amazon EBS 루트 볼륨을 검사하고 해당 볼륨을 사용하여 Windows Server 인스턴스 문제를 해결하기 위한 관련 로그를 수집할 수 있습니다.

EC2Rescue for Windows Server에는 모든 원본의 데이터를 수집하는 데이터 수집기 모듈과 미리 정의된 여러 규칙에 대해 수집된 데이터의 구문을 분석하여 문제를 식별하고 제안 사항을 제공하는 분석기 모듈이라는 두 가지 모듈이 있습니다.

EC2Rescue for Windows Server 도구는 Windows Server 2008 R2 및 그 이후 버전을 실행하는 Amazon EC2 인스턴스에서만 실행됩니다. 이 도구는 시작되면 Amazon EC2 인스턴스에서 실행 중인지 여부를 확인합니다.

Note

Linux 인스턴스를 사용하는 경우 [Linux용 EC2Rescue](#)를 참조하십시오.

내용

- [EC2Rescue for Windows Server GUI 사용 \(p. 1206\)](#)
- [명령줄에서 EC2Rescue for Windows Server 사용 \(p. 1210\)](#)
- [시스템 관리자 명령 실행과 함께 EC2Rescue for Windows Server 사용 \(p. 1215\)](#)

EC2Rescue for Windows Server GUI 사용

EC2Rescue for Windows Server는 오프라인 인스턴스에서 다음 분석을 수행할 수 있습니다.

옵션	설명
Diagnose and Rescue	<p>EC2Rescue for Windows Server는 다음 서비스 설정을 확인하고 해당 문제를 해결할 수 있습니다.</p> <ul style="list-style-type: none">• System Time<ul style="list-style-type: none">• RealTimeisUniversal - RealTimeisUniversal 레지스트리 키가 활성화되었는지 여부를 확인합니다. 비활성화되었을 경우 시간대가 UTC 외의 값으로 설정되면 Windows 시스템 시간에 오차가 발생합니다.• Windows 방화벽<ul style="list-style-type: none">• Domain networks - Windows 방화벽 프로필이 활성화되었는지 아니면 비활성화되었는지를 확인합니다.• Private networks - Windows 방화벽 프로필이 활성화되었는지 아니면 비활성화되었는지를 확인합니다.• Guest or public networks - Windows 방화벽 프로필이 활성화되었는지 아니면 비활성화되었는지를 확인합니다.• 원격 데스크톱<ul style="list-style-type: none">• Service Start - 원격 데스크톱 서비스의 활성화 여부를 확인합니다.• Remote Desktop Connections - 연결이 활성화되었는지 여부를 확인합니다.• TCP Port - 원격 데스크톱 서비스에서 데이터를 수신하는 포트를 확인합니다.• EC2Config(Windows Server 2012 R2 및 이전 버전)<ul style="list-style-type: none">• Installation - 설치되어 있는 EC2Config 버전을 확인합니다.• Service Start - EC2Config 서비스의 활성화 여부를 확인합니다.• Ec2SetPassword - 새로운 관리자 암호를 생성합니다.• Ec2HandleUserData - 인스턴스의 다음 부팅 시 사용자 데이터 스크립트를 실행할 수 있습니다.• EC2Launch(Windows Server 2016 이상)<ul style="list-style-type: none">• Installation - 설치되어 있는 EC2Launch 버전을 확인합니다.• Ec2SetPassword - 새로운 관리자 암호를 생성합니다.• 네트워크 인터페이스

옵션	설명
	<ul style="list-style-type: none"> • DHCP Service Startup - DHCP 서비스의 활성화 여부를 확인합니다. • Ethernet detail - 네트워크 드라이버 버전에 대한 정보를 표시합니다(확인될 경우). • DHCP on Ethernet - DHCP가 활성화되었는지 여부를 확인합니다.
복원	<p>다음 조치 중 하나를 취하십시오.</p> <ul style="list-style-type: none"> • Last Known Good Configuration - 인스턴스를 마지막으로 확인된 부팅 가능 상태로 부팅하려고 시도합니다. • Restore registry from backup - \Windows\System32\config\RegBack의 레지스트리를 복원합니다.
Capture Logs	분석을 위해 인스턴스에 대한 로그를 캡처할 수 있습니다.

EC2Rescue for Windows Server는 활성 인스턴스 및 오프라인 인스턴스에서 다음 데이터를 수집할 수 있습니다.

항목	설명
Event Log	애플리케이션, 시스템, EC2Config 이벤트 로그를 수집합니다.
Memory Dump	인스턴스에 있는 메모리 덤프 파일을 수집합니다.
EC2Config File	EC2Config 서비스에서 생성되는 로그 파일을 수집합니다.
EC2Launch File	EC2Launch 스크립트가 생성하는 로그 파일을 수집합니다.
SSM Agent File	SSM 에이전트가 생성하는 로그 파일을 수집합니다.
Sysprep Log	Windows System Preparation 도구가 생성하는 로그 파일을 수집합니다.
Driver SetupAPI Log	Windows SetupAPI 로그(setupapi.dev.log 및 setupapi.setup.log)를 수집합니다.
레지스트리	SYSTEM 및 SOFTWARE Hive를 수집합니다.
System Information	MSInfo32를 수집합니다.
Boot Configuration	HKEY_LOCAL_MACHINE\BCD00000000 Hive를 수집합니다.
Windows Update Log	인스턴스에 설치된 업데이트에 대한 정보를 수집합니다.

항목	설명
	<p>Note</p> <p>Windows 업데이트 로그는 Windows Server 2016 이후 인스턴스에서는 캡처되지 않습니다.</p>

비디오 안내

Brandon씨가 EC2Rescue for Windows Server의 Diagnose and Rescue 기능을 사용하는 방법을 보여줍니다.

[AWS Knowledge Center 비디오: EC2Rescue의 Diagnose and Rescue 기능을 사용하려면 어떻게 해야 하나요?](#)

오프라인 인스턴스 분석

오프라인 인스턴스 옵션은 Windows 인스턴스와 관련된 부팅 문제를 해결할 때 유용합니다.

오프라인 인스턴스에서 작업을 수행하려면

- 작동하는 Windows Server 인스턴스에서 [EC2Rescue for Windows Server](#) 도구를 다운로드하여 파일의 압축을 풁니다.

다음의 PowerShell 명령을 실행하여 Internet Explorer ESC(보안 강화 구성)의 변경 없이 EC2Rescue를 다운로드할 수 있습니다.

```
PS C:\> Invoke-WebRequest https://s3.amazonaws.com/ec2rescue/windows/EC2Rescue_latest.zip -OutFile $env:USERPROFILE\Desktop\EC2Rescue_latest.zip
```

이 명령을 사용하면 현재 로그인한 사용자의 데스크톱으로 .zip 형식의 EC2Rescue 파일이 다운로드됩니다.

- 문제가 있는 인스턴스가 아직 중지되지 않은 경우 해당 인스턴스를 중지합니다.
- 오류가 발생한 인스턴스에서 EBS 루트 볼륨을 분리한 후 이 볼륨을 EC2Rescue for Windows Server가 설치되고 작동하는 Windows 인스턴스에 연결합니다.
- 작동하는 인스턴스에서 EC2Rescue for Windows Server 도구를 실행하고 오프라인 인스턴스를 선택합니다.
- 새로 탑재한 디스크를 선택한 후 다음을 선택합니다.
- 디스크 선택을 확인한 후 예를 선택합니다.
- 수행할 오프라인 인스턴스 옵션을 선택한 후 다음을 선택합니다.

EC2Rescue for Windows Server 도구가 볼륨을 스캔한 후, 선택된 로그 파일을 바탕으로 문제 해결 정보를 수집합니다.

활성 인스턴스에서 데이터 수집

활성 인스턴스에서 로그와 기타 데이터를 수집할 수 있습니다.

활성 인스턴스에서 데이터를 수집하려면

- Windows 인스턴스에 연결합니다.
- [EC2Rescue for Windows Server](#) 도구를 Windows 인스턴스에 다운로드한 후 파일의 압축을 풁니다.

다음의 PowerShell 명령을 실행하여 Internet Explorer ESC(보안 강화 구성)의 변경 없이 EC2Rescue를 다운로드할 수 있습니다.

```
PS C:\> Invoke-WebRequest https://s3.amazonaws.com/ec2rescue/windows/  
EC2Rescue_latest.zip -OutFile $env:USERPROFILE\Desktop\EC2Rescue_latest.zip
```

이 명령을 사용하면 현재 로그인한 사용자의 데스크톱으로 .zip 형식의 EC2Rescue 파일이 다운로드됩니다.

3. EC2Rescue for Windows Server 애플리케이션을 열고 라이선스 계약에 동의합니다.
4. 다음, 현재 인스턴스, 로그 캡처(Capture logs)를 차례로 선택합니다.
5. 수집할 데이터 항목을 선택하고 수집...(Collect...)을 선택합니다. 경고를 읽고 예를 선택하여 계속 진행합니다.
6. ZIP 파일의 파일 이름과 위치를 선택한 다음 저장을 선택합니다.
7. EC2Rescue for Windows Server에서 작업을 완료하면 포함된 폴더 열기(Open Containing Folder)를 선택하여 ZIP 파일을 봅니다.
8. 마침을 클릭합니다.

명령줄에서 EC2Rescue for Windows Server 사용

EC2Rescue for Windows Server 명령줄 인터페이스(CLI)를 사용해 프로그래밍 방식으로 EC2Rescue for Windows Server 플러그인("작업"이라고 부름)을 실행할 수 있습니다.

EC2Rescue for Windows Server 도구에는 두 가지 실행 모드가 있습니다.

- /online — 이 모드에서는 EC2Rescue for Windows Server가 설치된 인스턴스에 대해 로그 파일 수집과 같은 작업을 수행할 수 있습니다.
- /offline:<device_id> — 이 모드에서는 EC2Rescue for Windows Server가 설치된 인스턴스에서 별도의 Amazon EC2 Windows 인스턴스에 연결된 오프라인 루트 볼륨에 대해 작업을 수행할 수 있습니다.

[EC2Rescue for Windows Server](#) 도구를 Windows 인스턴스에 다운로드한 후 파일의 압축을 풁니다. 다음 명령을 사용하여 도움말 파일을 볼 수 있습니다.

```
EC2RescueCmd.exe /help
```

EC2Rescue for Windows Server는 Amazon EC2 Windows 인스턴스에 대해 다음 작업을 수행할 수 있습니다.

- [수집 작업 \(p. 1210\)](#)
- [복구 작업 \(p. 1212\)](#)
- [복원 작업 \(p. 1214\)](#)

수집 작업

EC2Rescue for Windows Server는 활성 인스턴스 및 오프라인 인스턴스에서 다음 데이터를 수집할 수 있습니다. 모든 로그, 전체 로그 그룹 또는 특정 그룹 내 개별 로그를 수집할 수 있습니다.

로그 그룹	사용 가능한 로그	설명
all		모든 사용 가능한 로그를 수집합니다.

로그 그룹	사용 가능한 로그	설명
system-info	'MSInfo32 Output'	MSInfo32를 수집합니다.
eventlog	<ul style="list-style-type: none"> 'Application' 'System' 'EC2ConfigService' 	애플리케이션, 시스템, EC2Config 이벤트 로그를 수집합니다.
memory-dump	<ul style="list-style-type: none"> 'Memory Dump File' 'Mini Dump Files' 	인스턴스에 있는 메모리 덤프 파일을 수집합니다.
ec2config	<ul style="list-style-type: none"> 'Log Files' 'Configuration Files' 	EC2Config 서비스에서 생성되는 로그 파일을 수집합니다.
ec2launch	<ul style="list-style-type: none"> 'Logs' 'Config' 	EC2Launch 스크립트가 생성하는 로그 파일을 수집합니다.
ssm-agent	'Log Files'	SSM 에이전트가 생성하는 로그 파일을 수집합니다.
sysprep	'Log Files'	Windows System Preparation 도구가 생성하는 로그 파일을 수집합니다.
driver-setup	<ul style="list-style-type: none"> 'SetupAPI Log Files' 'DPInst Log File' 'AWS PV Setup Log File' 	Windows SetupAPI 로그 (setupapi.dev.log 및 setupapi.setup.log)를 수집합니다.
registry	<ul style="list-style-type: none"> 'SYSTEM' 'SOFTWARE' 'BCD' 	SYSTEM 및 SOFTWARE Hive를 수집합니다.
gpresult	'GPResult Output'	그룹 정책 보고서를 수집합니다.
egpu	<ul style="list-style-type: none"> 'Event Log' 'System Files' 	엘라스틱 GPU와 관련된 이벤트 로그를 수집합니다.
boot-config	'BCDEDIT Output'	HKEY_LOCAL_MACHINE \BCD00000000 Hive를 수집합니다.
windows-update	'Log Files'	인스턴스에 설치된 업데이트에 대한 정보를 수집합니다.
		<p style="text-align: center;">Note</p> <p style="text-align: center;">Windows 업데이트 로그는 Windows Server 2016 인스턴스에서 캡처되지 않습니다.</p>

다음과 같은 옵션을 사용할 수 있습니다.

- /output:<outputFilePath> - 수집된 로그 파일을 zip 형식으로 저장하는 데 필요한 대상 파일 경로 위치입니다.

- /no-offline - 오프라인 모드에서 사용되는 선택적 속성입니다. 작업 완료 후 볼륨을 오프라인으로 설정하지 않습니다.
- /no-fix-signature - 오프라인 모드에서 사용되는 선택적 속성입니다. 작업 완료 후 발생할 수 있는 디스크 서명 충돌을 해결하지 않습니다.

예

다음은 EC2Rescue for Windows Server CLI를 사용하는 예제입니다.

온라인 모드 예제

모든 사용 가능한 로그를 수집합니다.

```
EC2RescueCmd /accepteula /online /collect:all /output:<outputFilePath>
```

특정 로그 그룹만 수집합니다.

```
EC2RescueCmd /accepteula /online /collect:ec2config /output:<outputFilePath>
```

로그 그룹 내 개별 로그를 수집합니다.

```
EC2RescueCmd /accepteula /online /collect:'ec2config.Log Files,driver-setup.SetupAPI Log Files' /output:<outputFilePath>
```

오프라인 모드 예제

EBS 볼륨에서 모든 사용 가능한 로그를 수집합니다. 볼륨은 device_id value로 지정합니다.

```
EC2RescueCmd /accepteula /offline:xvdf /collect:all /output:<outputFilePath>
```

특정 로그 그룹만 수집합니다.

```
EC2RescueCmd /accepteula /offline:xvdf /collect:ec2config /output:<outputFilePath>
```

복구 작업

EC2Rescue for Windows Server는 다음 서비스 설정을 확인하고 해당 문제를 해결할 수 있습니다.

서비스 그룹	사용 가능한 작업	설명
all		
system-time	'RealTimeIsUniversal'	System Time <ul style="list-style-type: none">• RealTimeisUniversal - RealTimeisUniversal 레지스트리 키가 활성화되었는지 여부를 확인합니다. 비활성화되었을 경우 시간대가 UTC 외의 값으로 설정되면 Windows 시스템 시간에 오차가 발생합니다.
firewall	• 'Domain networks'	Windows 방화벽

서비스 그룹	사용 가능한 작업	설명
	<ul style="list-style-type: none"> 'Private networks' 'Guest or public networks' 	<ul style="list-style-type: none"> Domain networks - Windows 방화벽 프로필이 활성화되었는지 아니면 비활성화되었는지를 확인합니다. Private networks - Windows 방화벽 프로필이 활성화되었는지 아니면 비활성화되었는지를 확인합니다. Guest or public networks - Windows 방화벽 프로필이 활성화되었는지 아니면 비활성화되었는지를 확인합니다.
rdp	<ul style="list-style-type: none"> 'Service Start' 'Remote Desktop Connections' 'TCP Port' 	원격 데스크톱 <ul style="list-style-type: none"> Service Start - 원격 데스크톱 서비스의 활성화 여부를 확인합니다. Remote Desktop Connections - 연결이 활성화되었는지 여부를 확인합니다. TCP Port - 원격 데스크톱 서비스에서 데이터를 수신하는 포트를 확인합니다.
ec2config	<ul style="list-style-type: none"> 'Service Start' 'Ec2SetPassword' 'Ec2HandleUserData' 	EC2Config <ul style="list-style-type: none"> Service Start - EC2Config 서비스의 활성화 여부를 확인합니다. Ec2SetPassword - 새로운 관리자 암호를 생성합니다. Ec2HandleUserData - 인스턴스의 다음 부팅 시 사용자 데이터 스크립트를 실행할 수 있습니다.
ec2launch	'Reset Administrator Password'	새 Windows 관리자 암호를 생성합니다.
network	'DHCP Service Startup'	네트워크 인터페이스 <ul style="list-style-type: none"> DHCP Service Startup - DHCP 서비스의 활성화 여부를 확인합니다.

다음과 같은 옵션을 사용할 수 있습니다.

- ./level:<level> - 작업이 트리거해야 하는 확인 레벨에 대한 선택적 속성입니다. 허용되는 값은 information, warning, error, all입니다. 기본적으로 error로 설정됩니다.
- ./check-only - 보고서를 생성하지만 오프라인 볼륨은 수정하지 않는 선택적 속성입니다.
- ./no-offline - 작업 완료 후 볼륨을 오프라인으로 설정하는 것을 금지하는 선택적 속성입니다.
- ./no-fix-signature - 작업 완료 후 발생 가능한 디스크 서명 충돌을 해결하지 않는 선택적 옵션입니다.

복구 예제

다음은 EC2Rescue for Windows Server CLI를 사용하는 예제입니다. 볼륨은 device_id value를 사용하여 지정합니다.

볼륨에서 식별된 문제를 모두 해결하려고 시도합니다.

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:all
```

볼륨에서 특정 서비스 그룹의 모든 문제를 해결하려고 시도합니다.

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:firewall
```

볼륨에서 특정 서비스 그룹의 특정 항목을 해결하려고 시도합니다.

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:rdp.'Service Start'
```

볼륨에서 해결을 시도할 여러 문제를 지정합니다.

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:'system-
time.RealTimeIsUniversal,ec2config.Service Start'
```

복원 작업

EC2Rescue for Windows Server는 다음 서비스 설정을 확인하고 해당 문제를 해결할 수 있습니다.

서비스 그룹	사용 가능한 작업	설명
마지막으로 확인된 양호한 구성을 복원	lkgc	Last Known Good Configuration - 인스턴스를 마지막으로 확인된 부팅 가능 상태로 부팅하려고 시도 합니다.
최신 백업으로부터 Windows 레지 스트리를 복원	regback	Restore registry from backup - \Windows\System32\config \RegBack의 레지스트리를 복원 합니다.

다음과 같은 옵션을 사용할 수 있습니다.

- /no-offline — 작업 완료 후 볼륨을 오프라인으로 설정하는 것을 금지하는 선택적 속성입니다.
- /no-fix-signature — 작업 완료 후 발생 가능한 디스크 서명 충돌을 해결하지 않는 선택적 옵션입니다.

복원 예제

다음은 EC2Rescue for Windows Server CLI를 사용하는 예제입니다. 볼륨은 device_id value를 사용하여 지정합니다.

볼륨에서 마지막으로 확인된 양호한 구성을 복원합니다.

```
EC2RescueCmd /accepteula /offline:xvdf /restore:lkgc
```

볼륨에서 마지막 Windows 레지스트리 백업을 복원합니다.

```
EC2RescueCmd /accepteula /offline:xvdf /restore:regback
```

시스템 관리자 명령 실행과 함께 EC2Rescue for Windows Server 사용

AWS Support는 시스템 관리자 지원 인스턴스와 인터페이스하여 EC2Rescue for Windows Server를 실행하는 시스템 관리자 명령 실행 문서를 제공합니다. 이 Run Command 문서를 AWSSupport-RunEC2RescueForWindowsTool이라고 합니다.

이 시스템 관리자 Run Command 문서는 다음의 작업을 수행합니다.

- EC2Rescue for Windows Server를 다운로드하고 확인합니다.
- 간편하게 도구와 상호 작용하도록 PowerShell 모듈을 가져옵니다.
- 제공된 명령 및 파라미터를 사용하여 EC2RescueCmd를 실행합니다.

시스템 관리자 Run Command 문서는 세 가지 파라미터를 수락합니다.

- 명령—EC2Rescue for Windows Server 작업. 현재 허용되는 값은 다음과 같습니다.
 - ResetAccess — 로컬 관리자 암호를 재설정합니다. 현재 인스턴스의 로컬 관리자 암호가 재설정되고 임의로 생성된 암호가 Parameter Store에 /EC2Rescue/Password/<INSTANCE_ID>로 안전하게 저장됩니다. 이 작업을 선택하고 파라미터를 제공하지 않으면 기본 KMS 키를 사용하여 암호가 자동으로 암호화됩니다. 선택 사항으로, 파라미터에서 KMS Key ID를 지정하여 고유의 키로 암호를 암호화할 수 있습니다.
 - CollectLogs — /collect:all 작업으로 EC2Rescue for Windows Server를 실행합니다. 이 작업을 선택할 경우 로그를 업로드할 Amazon S3 버킷 이름이 Parameters에 포함되어 있어야 합니다.
 - FixAll — /rescue:all 작업으로 EC2Rescue for Windows Server를 실행합니다. 이 작업을 선택할 경우 Parameters에 복구할 블록 디바이스 이름이 포함되어야 합니다.
- 파라미터 — 지정된 명령에 대해 전달할 PowerShell 파라미터입니다.

Note

ResetAccess 작업이 작동하려면 암호화된 암호를 Parameter Store에 쓸 수 있도록 Amazon EC2 인스턴스에 다음 정책을 연결해야 합니다. 이 정책을 관련 IAM 역할에 연결하고 나서 몇 분 기다렸다가 인스턴스의 암호를 재설정하십시오.

기본 KMS 키 사용:

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ssm:PutParameter"  
            ],  
            "Resource": [  
                "arn:aws:ssm:region:account_id:parameter/EC2Rescue/Passwords/<instanceid>"  
            ]  
        }  
    ]  
}
```

사용자 지정 KMS 키 사용:

```
{
```

```
"Version": "2012-10-17",
"Statement": [
    {
        "Effect": "Allow",
        "Action": [
            "ssm:PutParameter"
        ],
        "Resource": [
            "arn:aws:ssm:region:account_id:parameter/EC2Rescue/Passwords/<instanceid>"
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "kms:Encrypt"
        ],
        "Resource": [
            "arn:aws:kms:region:account_id:key/<kmskeyid>"
        ]
    }
]
```

다음 절차는 Amazon EC2 콘솔에서 이 문서의 JSON을 보는 방법을 설명합니다.

시스템 관리자 Run Command 문서의 JSON을 보려면

1. <https://console.aws.amazon.com/systems-manager/home>에서 시스템 관리자 콘솔을 엽니다.
2. 탐색 창에서 공유 서비스(Shared Services)를 확장한 다음 문서를 선택합니다.
3. 검색 창에서 소유자를 내 소유 또는 Amazon 소유로 설정하고 문서 이름 접두사(Document name prefix)를 AWSSupport-RunEC2RescueForWindowsTool로 설정합니다.
4. AWSSupport-RunEC2RescueForWindowsTool 문서를 선택하고 콘텐츠(Contents)를 선택하여 JSON을 봅니다.

예제

다음은 시스템 관리자 명령 실행 문서를 사용하여 AWS CLI에서 EC2Rescue for Windows Server를 실행하는 몇 가지 예제입니다. AWS CLI를 사용한 명령 전송에 대한 자세한 내용은 [AWS CLI Command Reference](#)를 참조하십시오.

오프라인 루트 볼륨에서 식별된 모든 문제 해결 시도

다음과 같이 Amazon EC2 Windows 인스턴스에 연결된 오프라인 루트 볼륨에서 식별된 문제를 모두 해결하려고 시도합니다.

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue offline volume xvdf" --parameters "Command=FixAll, Parameters='xvdf'" --output text
```

현재 Amazon EC2 Windows 인스턴스에서 로그 수집

다음과 같이 현재 온라인 Amazon EC2 Windows 인스턴스에서 모든 로그를 수집하여 Amazon S3 버킷으로 업로드합니다.

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue online log collection to S3" --parameters "Command=CollectLogs, Parameters='YOURS3BUCKETNAME'" --output text
```

오프라인 Amazon EC2 Windows 인스턴스 볼륨에서 로그 수집

Amazon EC2 Windows 인스턴스에 연결된 오프라인 볼륨에서 로그를 수집하여 미리 서명된 URL을 사용해 Amazon S3로 업로드합니다.

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSsupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue offline log collection to S3" --parameters "Command=CollectLogs, Parameters=\"-Offline -BlockDeviceName xvdf -S3PreSignedUrl 'YOURSPRESIGNEDURL'\" --output text
```

로컬 관리자 암호 재설정

다음 예제에서는 로컬 관리자 암호를 재설정하는 데 사용할 수 있는 방법을 보여 줍니다. 출력에는 Parameter Store에 대한 링크가 제공됩니다. 여기에서 임의로 생성된 보안 암호를 찾은 다음 이 암호를 사용하여 RDP를 통해 Amazon EC2 Windows 인스턴스에 로컬 관리자로 연결할 수 있습니다.

기본 KMS 키 alias/aws/ssm을 사용하여 온라인 인스턴스의 로컬 관리자 암호를 재설정합니다.

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSsupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue online password reset" --parameters "Command=ResetAccess" --output text
```

KMS 키를 사용하여 온라인 인스턴스의 로컬 관리자 암호를 재설정합니다.

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSsupport-RunEC2RescueForWindowsTool" --comment "EC2Rescue online password reset" --parameters "Command=ResetAccess, Parameters=a133dc3c-a2g4-4fc6-a873-6c0720104bf0" --output text
```

Note

이 예제에서 KMS 값은 a133dc3c-a2g4-4fc6-a873-6c0720104bf0입니다.

진단 인터럽트 보내기(고급 사용자 전용)

Warning

진단 인터럽트를 고급 사용자를 대상으로 하는 기능입니다. 잘못 사용하면 인스턴스에 부정적인 영향을 줄 수 있습니다. 진단 인터럽트를 인스턴스에 전송하면 인스턴스가 고장 나거나 재부팅될 수 있으며 이는 데이터 손실로 이어질 수 있습니다.

도달할 수 없거나 응답이 없는 Windows 인스턴스에 진단 인터럽트를 보내 중단 오류를 수동으로 트리거 할 수 있습니다. 중단 오류는 흔히 블루 스크린 오류라고 합니다.

일반적으로 중단 오류가 발생하면 Windows 운영 체제가 충돌하고 다시 부팅되지만 특정 동작은 해당 구성에 따라 다릅니다. 중단 오류가 발생하면 운영 체제는 커널 메모리 덤프와 같은 디버깅 정보를 파일에 쓸 수도 있습니다. 이 정보를 사용하여 근본 원인 분석을 수행하여 인스턴스를 디버그할 수 있습니다.

메모리 덤프 데이터는 인스턴스 자체에서 운영 체제에 의해 로컬로 생성됩니다.

인스턴스에 진단 인터럽트를 전송하기 전에 운영 시스템에 대한 문서를 읽고 필요한 구성을 변경하시기를 권장합니다.

목차

- 지원되는 인스턴스 유형 (p. 1218)
- 사전 조건 (p. 1218)

- 진단 인터럽트 보내기 (p. 1218)

지원되는 인스턴스 유형

진단 인터럽트는 A1을 제외한 모든 Nitro 기반 인스턴스 유형에서 지원됩니다. 자세한 내용은 [Nitro 시스템에 구축된 인스턴스 \(p. 118\)](#) 단원을 참조하십시오.

사전 조건

진단 인터럽트를 사용하기 전에 중단 오류가 발생할 때 필요한 조치를 수행하도록 인스턴스의 운영 체제를 구성해야 합니다.

중단 오류가 발생할 때 메모리 덤프를 생성하도록 Windows를 구성하려면

1. 인스턴스에 연결합니다.
2. 제어판을 열고 시스템, 고급 시스템 설정을 선택합니다.
3. [시스템 속성] 대화 상자에서 [권한] 탭을 선택합니다.
4. 시작 및 복구 섹션에서 설정...을 선택합니다.
5. 시스템 오류 섹션에서 필요에 따라 설정을 구성한 다음 확인을 선택하십시오.

Windows 중단 오류 구성에 대한 자세한 내용은 [Windows의 메모리 덤프 파일 옵션 개요](#)를 참조하십시오.

진단 인터럽트 보내기

필요한 구성 변경을 완료하면 AWS CLI 또는 Amazon EC2 API를 사용하여 인스턴스에 진단 인터럽트를 보낼 수 있습니다.

인스턴스에 진단 인터럽트를 보내려면(AWS CLI)

`send-diagnostic-interrupt` 명령을 사용하고 인스턴스 ID를 지정하십시오.

```
aws ec2 send-diagnostic-interrupt --instance-id i-1234567890abcdef0
```

인스턴스에 진단 인터럽트를 보내는 방법(Windows PowerShell용 AWS 도구)

`Send-EC2DiagnosticInterrupt` cmdlt를 사용하고 인스턴스 ID를 지정하십시오.

```
PS C:\> Send-EC2DiagnosticInterrupt-InstanceId i-1234567890abcdef0
```

Windows 인스턴스의 일반적인 문제

다음은 Windows Server를 실행하는 EC2 인스턴스와 관련된 일반적인 문제를 해결하는 데 도움이 되는 문제 해결 팁입니다.

문제

- EBS 볼륨이 Windows Server 2016 이상에서 초기화를 수행하지 않음 (p. 1219)
- DSRM(Directory Services Restore Mode)로 EC2 Windows 인스턴스 부팅 (p. 1219)
- 인스턴스의 네트워크 연결이 끊어지거나 예약된 작업이 예정 시간에 실행되지 않음 (p. 1221)
- 콘솔 출력을 가져올 수 없음 (p. 1221)

-
- 네트워크에서 Windows Server 2012 R2를 이용할 수 없는 경우 (p. 1222)

EBS 볼륨이 Windows Server 2016 이상에서 초기화를 수행하지 않음

Windows Server 2016 이상용 Amazon 머신 이미지(AMI)에서 생성한 인스턴스는 EBS 볼륨 초기화를 비롯한 다양한 시작 작업에 EC2Launch 서비스를 사용합니다. 기본적으로 EC2Launch는 두 번째 볼륨을 초기화하지 않습니다. 이러한 작업을 초기화하도록 EC2Launch를 자동적으로 구성할 수도 있습니다.

드라이브 문자를 볼륨에 매핑하려면

- 구성할 인스턴스에 연결하고 C:\ProgramData\Amazon\EC2-Windows\Launch\Config\DriveLetterMappingConfig.json 텍스트 편집기에 파일을 엽니다.
- 다음 형식을 이용해 볼륨 설정을 지정합니다.

```
{  
    "driveLetterMapping": [  
        {  
            "volumeName": "sample volume",  
            "driveLetter": "H"  
        }  
    ]  
}
```

- 변경 내용을 저장하고 파일을 닫습니다.
- 디스크를 초기화할 EC2Launch 스크립트를 수행하기 위해서 Windows PowerShell을 열고 다음 명령을 실행합니다.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1
```

인스턴스를 부팅할 때마다 디스크를 초기화하려면 다음과 같이 -Schedule 플래그를 추가하십시오.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1 -Schedule
```

DSRM(Directory Services Restore Mode)로 EC2 Windows 인스턴스 부팅

Microsoft Active Directory를 실행 중인 인스턴스에서 시스템 오류 또는 그 밖에 심각한 문제가 발생할 경우 DSRM(Directory Services Restore)이라는 특수한 버전의 안전 모드로 부팅해 인스턴스 문제를 해결할 수 있습니다. DSRM에서 Active Directory를 복구할 수 있습니다.

DSRM에 대한 드라이버 지원

DSRM을 활성화하고 인스턴스로 부팅하는 방법은 인스턴스가 실행 중인 드라이버에 따라 다릅니다. EC2 콘솔에서는 시스템 로그에서 인스턴스에 대한 드라이버 버전 세부 정보를 볼 수 있습니다. 다음 표에서는 DSRM이 지원되는 드라이버를 보여 줍니다.

드라이버 버전	DSRM 지원 여부	다음 단계
Citrix PV 5.9	아니요	백업에서 인스턴스를 복원합니다. DSRM을 활성화할 수 없습니다.

드라이버 버전	DSRM 지원 여부	다음 단계
AWS PV 7.2.0	아니요	이 드라이버에 대한 DSRM은 지원되지 않지만, 그래도 인스턴스에서 루트 볼륨을 분리하고, 볼륨의 스냅샷을 생성하거나 볼륨에서 AMI를 만들고, 이를 보조 볼륨과 동일한 가용 영역에 있는 다른 인스턴스에 연결할 수 있습니다. (이 섹션의 설명처럼) DSRM을 활성화할 수 있습니다.
AWS PV 7.2.2 이상	예	루트 볼륨을 분리하고 다른 인스턴스에 연결하고 DSRM(이 섹션에 설명된 대로)을 활성화합니다.
향상된 네트워킹	예	루트 볼륨을 분리하고 다른 인스턴스에 연결하고 DSRM(이 섹션에 설명된 대로)을 활성화합니다.

향상된 네트워킹을 활성화하는 방법에 대한 자세한 내용은 [VPC 상의 Windows 인스턴스에서 향상된 네트워킹 사용](#)을 참조하십시오. AWS PV 드라이버 업그레이드에 대한 자세한 내용은 [Windows 인스턴스에서 PV 드라이버 업그레이드 \(p. 528\)](#) 단원을 참조하십시오.

DSRM으로 부팅하도록 인스턴스 구성

EC2 Windows 인스턴스는 운영 체제가 실행되기 전에 네트워크에 연결되지 않습니다. 이러한 이유로 키보드에서 F8 버튼을 눌러 부팅 옵션을 선택할 수 없습니다. 다음 절차 중 하나를 사용하여 DSRM으로 EC2 Windows Server 인스턴스를 부팅할 수 있습니다.

Active Directory가 손상되었는데도 인스턴스가 여전히 실행 중이라고 의심된다면 System Configuration 대화 상자나 명령 프롬프트를 사용하여 DSRM으로 부팅하도록 인스턴스를 구성할 수 있습니다.

System Configuration 대화 상자를 사용하여 DSRM으로 온라인 인스턴스를 부팅하려면

1. 실행 대화 상자에 `msconfig`를 입력하고 Enter 키를 누릅니다.
2. 부팅 탭을 선택합니다.
3. 부팅 옵션 아래에서 안전 부팅을 선택합니다.
4. Active Directory 복구를 선택한 다음 확인을 선택합니다. 시스템에서 서버를 재부팅하라는 메시지를 표시합니다.

명령줄을 사용하여 DSRM으로 온라인 인스턴스를 부팅하려면

명령 프롬프트 창에서 다음 명령을 실행합니다.

```
bcdeedit /set safeboot dsrepair
```

인스턴스가 오프라인이고 연결할 수 없으면 루트 볼륨을 분리하고 다른 인스턴스에 연결하여 DSRM 모드를 활성화해야 합니다.

DSRM으로 오프라인 인스턴스를 부팅하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택합니다.
3. 문제가 발생한 인스턴스를 찾습니다. 인스턴스에 대한 컨텍스트(오른쪽 클릭) 메뉴를 열고, 인스턴스 상태를 선택한 다음 중지를 선택합니다.
4. 인스턴스 시작을 선택하고 문제가 발생한 인스턴스와 동일한 가용 영역에 임시 인스턴스를 생성합니다. 다른 버전의 Windows를 사용하는 인스턴스 유형을 선택합니다. 예를 들어, 인스턴스가 Windows Server 2008이면 Windows 2008 R2 인스턴스를 선택합니다.

Important

문제가 발생한 인스턴스와 동일한 가용 영역에서 인스턴스를 생성하지 않는 경우에는 문제가 발생한 인스턴스의 루트 볼륨을 새 인스턴스에 연결할 수 없습니다.

5. 탐색 창에서 볼륨을 선택합니다.
6. 문제가 발생한 인스턴스의 루트 볼륨을 찾습니다. 볼륨을 [분리](#)하고 이전에 생성한 임시 인스턴스에 [연결](#)합니다. 기본 디바이스 이름(xvdf)으로 연결합니다.
7. 원격 데스크톱을 사용하여 임시 인스턴스에 연결한 후 디스크 관리 유ти리티를 사용하여 [볼륨을 사용할 수 있도록 지정](#)합니다.
8. 명령 프롬프트를 열고 다음 명령을 실행합니다. 방금 연결한 보조 볼륨의 실제 드라이브 문자로 D를 바꿉니다.

```
bcdedit /store D:\Boot\BCD /set {default} safeboot dsrepair
```

9. 디스크 관리(Disk Management) 유ти리티에서 이전에 연결한 드라이브를 선택하고 컨텍스트(오른쪽 클릭) 메뉴를 열고 오프라인(Offline)을 선택합니다.
10. EC2 콘솔에서 임시 인스턴스로부터 문제가 발생한 볼륨을 분리하고 디바이스 이름 /dev/sda1을 사용하여 원래 인스턴스에 다시 연결합니다. 볼륨을 루트 볼륨으로 지정하려면 이 디바이스 이름을 지정해야 합니다.
11. 인스턴스를 [시작](#)합니다.
12. 인스턴스가 EC2 콘솔에서 상태 확인을 통과하면 원격 데스크톱을 사용하여 인스턴스에 연결하고 DSRM 모드로 부팅되는지 확인합니다.
13. (선택 사항) 앞에서 임시로 생성한 인스턴스는 이 절차에서 삭제하거나 중단합니다.

인스턴스의 네트워크 연결이 끊어지거나 예약된 작업이 예정 시간에 실행되지 않음

인스턴스를 다시 시작하면 네트워크 연결이 끊어지는 경우 인스턴스의 시간이 잘못 설정되었을 수 있습니다.

기본적으로 Windows 인스턴스는 협정 세계 시(UTC)를 사용합니다. 인스턴스의 시간을 다른 표준 시간대로 설정한 후 인스턴스를 다시 시작하면 시간이 어긋나면서 인스턴스가 일시적으로 IP 주소를 손실합니다. 인스턴스의 네트워크 연결은 이후에 복구되지만 여기에 몇 시간이 걸릴 수 있습니다. 인스턴스의 네트워크 연결이 복구되는 데 걸리는 시간은 UTC와 다른 표준 시간대의 차이에 따라 달라집니다.

이와 같은 시간 문제로 인해 예약된 작업이 예정 시간에 실행되지 않을 수도 있습니다. 이러한 경우 인스턴스의 시간이 정확하지 않기 때문에 예약된 작업이 예정 시간에 실행되지 않는 것입니다.

UTC 이외의 표준 시간대를 지속적으로 사용하려면 RealTimelsUniversal 레지스트리 키를 설정해야 합니다. 이 키가 없으면 인스턴스가 다시 시작된 후 UTC가 사용됩니다.

네트워크 연결이 끊어지게 만든 시간 문제를 해결하려면 다음을 수행합니다.

1. 권장 PV 드라이버를 사용하고 있는지 확인합니다. 자세한 내용은 [Windows 인스턴스에서 PV 드라이버 업그레이드 \(p. 528\)](#) 단원을 참조하십시오.
2. HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation\RealTimelsUniversal 레지스트리 키가 있으며 값이 1로 설정되었는지 확인합니다.

콘솔 출력을 가져올 수 없음

Windows 인스턴스의 경우 인스턴스 콘솔은 Windows 부팅 프로세스 중에 수행된 작업의 출력을 표시합니다. Windows가 성공적으로 부팅된 경우 마지막으로 기록되는 메시지는 Windows is Ready to use입니다.

다. 콘솔에 이벤트 로그 메시지를 표시할 수도 있지만 이 기능은 기본적으로 해제된 상태입니다. 자세한 내용은 [EC2 서비스 속성 \(p. 500\)](#) 단원을 참조하십시오.

Amazon EC2 콘솔을 사용하여 인스턴스의 콘솔 출력을 확인하려면 인스턴스를 선택하고, 작업을 선택하고, 인스턴스 설정을 선택한 다음 시스템 로그 가져오기를 선택합니다. 명령줄을 사용하여 콘솔 출력을 확인하려면 [get-console-output\(AWS CLI\)](#) 또는 [Get-EC2ConsoleOutput\(Windows PowerShell용 AWS 도구\)](#) 명령 중 하나를 사용합니다.

Windows Server 2012 R2 및 이전 버전을 실행하는 인스턴스에서 콘솔 출력이 비어 있는 경우 EC2Config 서비스에 구성 파일 오류 또는 Windows 부팅 실패 등의 문제가 발생한 것일 수 있습니다. 문제를 해결하려면 최신 버전의 EC2Config를 다운로드하여 설치합니다. 자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 단원을 참조하십시오.

네트워크에서 Windows Server 2012 R2를 이용할 수 없는 경우

Windows Server 2012 R2를 네트워크에서 이용할 수 없을 때 문제 해결에 대한 자세한 내용은 [인스턴스를 재부팅한 후 Windows Server 2012 R2에서 네트워크와 스토리지 연결이 끊김 \(p. 534\)](#) 단원을 참조하십시오.

일반적인 메시지를 통한 Windows 인스턴스 문제 해결

이 섹션에는 공통 메시지를 바탕으로 문제를 해결하는 데 도움이 되는 팁이 포함되어 있습니다.

주제

- "Password is not available" (p. 1222)
- "Password not available yet" (p. 1223)
- "Cannot retrieve Windows password" (p. 1223)
- "Waiting for the metadata service" (p. 1223)
- "Unable to activate Windows" (p. 1226)
- "Windows is not genuine (0x80070005)" (p. 1227)
- "No Terminal Server License Servers available to provide a license" (p. 1227)
- "Some settings are managed by your organization.(일부 설정이 사용자의 조직에 의해 관리됩니다.)(Windows Server 2019)" (p. 1228)

"Password is not available"

원격 데스크톱을 사용하여 Windows 인스턴스에 연결하려면 계정과 암호를 지정해야 합니다. 제공되는 계정과 암호는 인스턴스를 시작하는 데 사용한 AMI를 기준으로 합니다. 관리자 계정의 자동 생성 암호를 검색할 수도 있고, AMI가 생성된 원래 인스턴스에서 사용하는 계정과 암호를 사용할 수도 있습니다.

Windows 인스턴스가 무작위 암호를 생성하도록 구성되지 않은 경우 콘솔을 사용하여 자동 생성 암호를 검색할 때 다음 메시지가 표시됩니다.

Password is not available.
The instance was launched from a custom AMI, or the default password has changed. A password cannot be retrieved for this instance. If you have forgotten your password, you can

reset it using the Amazon EC2 configuration service. For more information, see [Passwords for a Windows Server instance](#).

인스턴스의 콘솔 출력을 확인하여 인스턴스를 시작하는 데 사용한 AMI가 암호 생성이 해제된 상태로 생성되었는지 확인합니다. 암호 생성이 해제된 경우 콘솔 출력에 다음 내용이 포함됩니다.

Ec2SetPassword: Disabled

암호 생성이 해제되어 있으며 원래 인스턴스의 암호를 모르는 경우 이 인스턴스의 암호를 재설정할 수 있습니다. 자세한 내용은 [기억나지 않거나 만료된 Windows 관리자 암호 재설정 \(p. 1196\)](#) 단원을 참조하십시오.

"Password not available yet"

원격 데스크톱을 사용하여 Windows 인스턴스에 연결하려면 계정과 암호를 지정해야 합니다. 제공되는 계정과 암호는 인스턴스를 시작하는 데 사용한 AMI를 기준으로 합니다. 관리자 계정의 자동 생성 암호를 검색할 수도 있고, AMI가 생성된 원래 인스턴스에서 사용하는 계정과 암호를 사용할 수도 있습니다.

몇 분 이내에 암호가 제공됩니다. 암호가 제공되지 않는 경우 콘솔을 사용하여 자동 생성 암호를 검색할 때 다음 메시지가 표시됩니다.

Password not available yet.
Please wait at least 4 minutes after launching an instance before trying to retrieve the auto-generated password.

4분이 지나도록 암호를 받지 못한 경우 EC2Config가 해제되었을 수 있습니다. 콘솔 출력이 비어 있는지 확인해 봅니다. 자세한 내용은 [콘솔 출력을 가져올 수 없음 \(p. 1221\)](#) 단원을 참조하십시오.

또한 관리 포털에 액세스하는 데 사용되는 AWS Identity and Access Management(IAM) 계정에 ec2:GetPasswordData 작업이 허용되었는지 확인합니다. IAM 권한에 대한 자세한 내용은 [IAM이란?](#)을 참조하십시오.

"Cannot retrieve Windows password"

관리자 계정의 자동 생성 암호를 검색하려면 인스턴스 시작 시에 지정한 키 페어의 프라이빗 키를 사용해야 합니다. 인스턴스 시작 시에 키 페어를 지정하지 않은 경우 다음 메시지가 표시됩니다.

Cannot retrieve Windows password

이 인스턴스를 종료하고 같은 AMI를 사용하여 키 페어를 지정하면서 새 인스턴스를 시작할 수 있습니다.

"Waiting for the metadata service"

Windows 인스턴스는 인스턴스 메타데이터 정보를 가져와야 활성화될 수 있습니다. 기본적으로 WaitForMetaDataAvailable 설정은 EC2Config 서비스가 인스턴스 메타데이터에 액세스되는 것을 기다린 후에 부팅 프로세스를 진행하도록 합니다. 자세한 내용은 [인스턴스 메타데이터 및 사용자 데이터 \(p. 576\)](#) 단원을 참조하십시오.

인스턴스가 인스턴스 연결성 테스트에 실패한 경우 다음 방법으로 이 문제를 해결하십시오.

- VPC에 대한 CIDR 블록을 검사합니다. IP 주소의 범위가 224.0.0.0~255.255.255.255인 VPC로 시작한 경우(Class D 및 Class E IP 주소 범위) Windows 인스턴스가 올바르게 부팅되지 않습니다. 이러한 IP 주소 범위는 예약되어 있으므로 호스트 디바이스에 할당해서는 안 됩니다. [RFC 1918](#) 규격에 따라 사설(비 공개적으로 라우팅 가능) IP 주소 범위에 속하는 CIDR 블록으로 VPC를 생성하는 것이 좋습니다.

- 시스템이 고정 IP 주소로 구성되었을 수 있습니다. [네트워크 인터페이스를 생성 \(p. 742\)](#)하고 [인스턴스에 연결 \(p. 744\)](#)합니다.
- 연결할 수 없는 Windows 인스턴스의 DHCP를 활성화하려면 다음을 수행합니다.
 - 해당 인스턴스를 중지하고 루트 볼륨을 분리합니다.
 - 해당 인스턴스와 동일한 가용 영역에서 임시 인스턴스를 시작합니다.

Warning

임시 인스턴스가 원본 인스턴스와 동일한 AMI를 기반으로 하는 경우 추가 단계를 수행해야 합니다. 그렇지 않으면 디스크 서명 충돌로 인해 루트 볼륨 복원 후 원본 인스턴스를 부팅할 수 없습니다. 또는 임시 인스턴스에 대한 다른 AMI를 선택합니다. 예를 들어, 원본 인스턴스에서 Windows Server 2008 R2용 AWS Windows AMI를 사용할 경우 Windows Server 2012용 AWS Windows AMI를 사용하여 임시 인스턴스를 시작합니다.

- 해당 인스턴스의 루트 볼륨을 이 임시 인스턴스에 연결합니다. 임시 인스턴스에 연결하고 디스크 관리 유ти리티를 열어서 드라이브를 온라인 상태로 만듭니다.
- 임시 인스턴스에서 Regedit를 열고 HKEY_LOCAL_MACHINE을 선택합니다. 파일 메뉴에서 Hive 로드를 선택합니다. 드라이브를 선택하고, Windows\System32\config\SYSTEM 파일을 열고, 메시지가 나타나면 키 이름을 임의로 지정합니다.
- 방금 로드한 키를 선택하고 ControlSet001\Services\Tcpip\Parameters\Interfaces 경로로 이동합니다. 각 네트워크 인터페이스가 GUID에 따라 나열됩니다. 올바른 네트워크 인터페이스를 선택합니다. DHCP를 비활성화하고 고정 IP 주소를 할당한 경우 EnabledDHCP는 0으로 설정됩니다. DHCP를 활성화하려면 EnabledDHCP를 1로 설정하고 NameServer, SubnetMask, IPAddress 및 DefaultGateway 키가 있는 경우 해당 키를 삭제합니다. 키를 한 번 더 선택하고 파일 메뉴에서 Hive 로드 취소를 선택합니다.

Note

네트워크 인터페이스가 여러 개일 경우 DHCP를 사용할 인터페이스를 식별해야 합니다. 해당 네트워크 인터페이스를 확인하려면 NameServer, SubnetMask, IPAddress, DefaultGateway 키 값을 확인합니다. 이들 값은 이전 인스턴스의 정적 구성을 표시합니다.

- (선택 사항) DHCP가 이미 활성화되어 있다면 메타데이터 서비스로 연결되는 경로가 없는 경우일 수 있습니다. EC2Config를 업데이트하면 문제가 해결됩니다.
 - 최신 버전의 EC2Config 서비스를 [다운로드](#)하여 설치합니다. 이 서비스 설치에 대한 자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 단원을 참조하십시오.
 - 연결한 드라이브의 .zip 디렉터리에 Temp 파일의 압축을 풁니다.
 - Regedit를 열고 HKEY_LOCAL_MACHINE을 선택합니다. 파일 메뉴에서 Hive 로드를 선택합니다. 드라이브를 선택하고, Windows\System32\config\SOFTWARE 파일을 열고, 메시지가 나타나면 키 이름을 임의로 지정합니다.
 - 방금 로드한 키를 선택하고 Microsoft\Windows\CurrentVersion 경로로 이동합니다. RunOnce 키를 선택합니다. 이 키가 없는 경우 CurrentVersion을 마우스 오른쪽 버튼으로 클릭하고 새로 생성을 가리킨 후 키를 선택하고 키의 이름을 RunOnce로 지정합니다. 마우스 오른쪽 버튼을 클릭하고 키를 가리킨 후 문자열 값을 선택합니다. Ec2Install을 이름으로, c:\Temp\Ec2Install.exe -q를 데이터로 입력합니다.
 - 키를 한 번 더 선택하고 파일 메뉴에서 Hive 로드 취소를 선택합니다.
- (선택 사항) 임시 인스턴스가 원본 인스턴스와 동일한 AMI를 기반으로 하는 경우 다음 단계를 수행해야 합니다. 그렇지 않으면 디스크 서명 충돌로 인해 루트 볼륨 복원 후 원본 인스턴스를 부팅할 수 없습니다.

Warning

다음 절차에서는 레지스트리 편집기를 사용하여 Windows 레지스트리를 편집하는 방법을 설명합니다. Windows 레지스트리에 대해 또는 레지스트리 편집기를 사용하여 안전하게 수정하는 방법을 잘 알지 못하는 경우 [레지스트리 구성](#)을 참조하십시오.

- a. 명령 프롬프트를 열고 regedit.exe를 입력한 후 Enter 키를 누릅니다.
- b. 레지스트리 편집기의 컨텍스트 메뉴(마우스 오른쪽 버튼 클릭)에서 HKEY_LOCAL_MACHINE을 선택한 후 찾기를 선택합니다.
- c. Windows Boot Manager를 입력한 후 다음 찾기를 선택합니다.
- d. 키 11000001을 선택합니다. 이 키는 이전 단계에서 찾은 키 바로 위의 키입니다.
- e. 오른쪽 창에서 Element를 선택한 후 컨텍스트 메뉴(마우스 오른쪽 버튼 클릭)에서 수정을 선택합니다.
- f. 데이터에서 오프셋 0x38의 4바이트 디스크 서명을 찾습니다. 바이트를 거꾸로 하여 디스크 서명을 만들고 기록해둡니다. 예를 들어, 다음 데이터가 나타내는 디스크 서명은 E9EB3AA5입니다.

```
...
0030  00 00 00 00 01 00 00 00
0038  A5 3A EB E9 00 00 00 00
0040  00 00 00 00 00 00 00 00
...
```

- g. 명령 프롬프트 창에서 다음 명령을 실행하여 Microsoft DiskPart를 시작합니다.

```
diskpart
```

- h. 다음 DiskPart 명령을 실행하여 볼륨을 선택합니다. 디스크 관리 유ти리티를 사용하여 디스크 번호가 1인지 확인할 수 있습니다.

```
DISKPART> select disk 1

Disk 1 is now the selected disk.
```

- i. 다음 DiskPart 명령을 실행하여 디스크 서명을 봅니다.

```
DISKPART> uniqueid disk

Disk ID: 0C764FA8
```

- j. 이전 단계에서 표시된 디스크 서명이 앞에서 기록한 BCD의 디스크 서명과 일치하지 않을 경우 다음 DiskPart 명령을 사용하여 일치하도록 디스크 서명을 변경합니다.

```
DISKPART> uniqueid disk id=E9EB3AA5
```

8. 디스크 관리 유ти리티를 사용하여 드라이브를 오프라인으로 설정합니다.

Note

임시 인스턴스가 영향을 받는 인스턴스와 동일한 운영 체제를 실행하고 있는 경우 드라이브가 자동으로 오프라인 상태가 되므로 사용자가 수동으로 드라이브를 오프라인으로 설정할 필요가 없습니다.

9. 임시 인스턴스에서 볼륨을 분리합니다. 임시 인스턴스를 더 이상 사용하지 않는 경우 해당 인스턴스는 종료해도 됩니다.
10. 볼륨을 /dev/sda1로 연결하여 해당 인스턴스의 루트 볼륨을 복원합니다.
11. 영향을 받는 인스턴스를 시작합니다.

인스턴스에 연결된 경우 인스턴스에서 인터넷 브라우저를 열고 다음과 같이 메타데이터 서버 URL을 입력합니다.

```
http://169.254.169.254/latest/meta-data/
```

메타데이터 서버에 접속할 수 없는 경우 다음 방법으로 문제를 해결하십시오.

- 최신 버전의 EC2Config 서비스를 [다운로드](#)하여 설치합니다. 이 서비스 설치에 대한 자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 단원을 참조하십시오.
- Windows 인스턴스에서 RedHat PV 드라이버를 실행하는지 여부를 확인합니다. 실행하는 경우 Citrix PV 드라이버를 업데이트합니다. 자세한 내용은 [Windows 인스턴스에서 PV 드라이버 업그레이드 \(p. 528\)](#) 단원을 참조하십시오.
- 방화벽, IPSec 및 프록시 설정으로 인해 메타데이터 서비스(169.254.169.254) 또는 KMS 서버(TargetKMServer의 C:\Program Files\Amazon\Ec2ConfigService\Settings\ActivationSettings.xml 요소에 지정된 주소)로 송신되는 트래픽이 차단되지 않는지 확인합니다.
- 다음 명령을 사용하여 메타데이터 서비스(169.254.169.254)로 연결되는 경로가 있는지 확인합니다.

```
route print
```

- 인스턴스의 가용 영역에 영향을 줄 수 있는 네트워크 문제가 있는지 확인합니다. <http://status.aws.amazon.com/> 위치로 이동합니다.

"Unable to activate Windows"

Windows 인스턴스는 Windows KMS 정품 인증을 사용합니다. 인스턴스가 KMS 서버에 접속할 수 없는 경우 A problem occurred when Windows tried to activate. Error Code 0xC004F074 메시지가 표시됩니다. 180일마다 Windows 정품을 인증해야 합니다. EC2Config는 정품 인증 기간이 만료되기 전에 KMS 서버 접속을 시도하여 Windows의 정품 인증 상태를 유지합니다.

Windows 정품 인증에 문제가 발생하는 경우, 다음 절차에 따라 문제를 해결하십시오.

EC2Config의 경우(Windows Server 2012 R2 AMI 이전)

- 최신 버전의 EC2Config 서비스를 [다운로드](#)하여 설치합니다. 이 서비스 설치에 대한 자세한 내용은 [최신 EC2Config 설치 \(p. 498\)](#) 단원을 참조하십시오.
- 인스턴스에 로그인하여 C:\Program Files\Amazon\Ec2ConfigService\Settings\config.xml 파일을 엽니다.
- config.xml 파일에서 Ec2WindowsActivate 플러그인을 찾습니다. 상태를 활성화로 변경한 다음 변경 사항을 저장합니다.
- Windows Services 스냅인에서 EC2Config 서비스를 재시작하거나 인스턴스를 재부팅합니다.

이렇게 해도 정품 인증 문제가 해결되지 않는 경우, 다음과 같은 추가 단계를 수행하십시오.

- KMS 대상 설정: C:\> slmgr.vbs /skms 169.254.169.250:1688
- Windows 정품 인증: C:\> slmgr.vbs /ato

EC2Launch의 경우(Windows Server 2016 AMI 이상)

- EC2Launch 모듈을 가져옵니다.

```
PS C:\> Import-Module "C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Ec2Launch.psd1"
```

- Add-Routes 함수를 호출합니다.

```
PS C:\> Add-Routes
```

- Set-ActivationSettings 함수를 호출합니다.

```
PS C:\> Set-Activationsettings
```

- 그런 다음 아래 스크립트를 실행하여 Windows를 활성화합니다.

```
PS C:\> cscript "${env:SYSTEMROOT}\system32\slmgr.vbs" /ato
```

EC2Config와 EC2Launch의 경우, 그래도 정품 인증 오류가 발생한다면 다음 정보를 확인해 보십시오.

- KMS 서버로 연결되는 경로가 있는지 확인합니다. C:\Program Files\Amazon\Ec2ConfigService\Settings\ActivationSettings.xml을 열고 TargetKMServer 요소를 찾습니다. 다음 명령을 실행하여 이러한 KMS 서버 주소가 나열되어 있는지 확인합니다.

```
route print
```

- KMS 클라이언트 키가 설정되어 있는지 확인합니다. 다음 명령을 실행하고 출력을 확인합니다.

```
C:\Windows\System32\slmgr.vbs /dlv
```

출력에 Error: product key not found 메시지가 있는 경우, KMS 클라이언트 키가 설정되지 않은 것입니다. KMS 클라이언트 키가 설정되지 않은 경우 이 Microsoft 문서([KMS 클라이언트 설정 키](#))의 설명에 따라 클라이언트 키를 조회하고 다음 명령을 실행하여 KMS 클라이언트 키를 설정합니다.

```
C:\Windows\System32\slmgr.vbs /ipk client_key
```

- 시스템의 시간과 표준 시간대가 올바른지 확인합니다. Windows Server 2008 이상에서 UTC 이외의 표준 시간대를 사용하는 경우 1 레지스트리 키를 추가하고 값을 HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation\RealTimeIsUniversal로 설정하여 시간을 정확하게 맞춥니다..
- Windows 방화벽이 활성화된 경우 다음 명령을 사용하여 임시로 비활성화합니다.

```
netsh advfirewall set allprofiles state off
```

"Windows is not genuine (0x80070005)"

Windows 인스턴스는 Windows KMS 정품 인증을 사용합니다. 인스턴스에서 정품 인증 프로세스를 완료할 수 없는 경우 Windows 사본이 정품이 아니라는 메시지가 표시됩니다.

["Unable to activate Windows" \(p. 1226\)](#)의 권장 사항을 따르십시오.

"No Terminal Server License Servers available to provide a license"

Windows Server의 기본 라이선스는 원격 데스크톱 사용자 2명 동시 연결입니다. 3명 이상의 사용자가 원격 데스크톱으로 Windows 인스턴스에 액세스해야 하는 경우 원격 데스크톱 서비스 CAL(Client Access License)을 구입하고 원격 데스크톱 세션 호스트 및 원격 데스크톱 라이선싱 서버 역할을 설치해야 합니다.

다음과 같은 문제가 있는지 확인합니다.

- 최대 동시 RDP 세션 수를 초과했습니다.
- Windows 원격 데스크톱 서비스 역할을 설치했습니다.

-
- 라이선스가 만료되었을 때 인스턴스에 연결할 수 없습니다.
다음과 같은 방법을 시도할 수 있습니다.
 - 다음과 같이 명령줄에서 /admin 파라미터를 사용하여 인스턴스에 연결합니다.

```
mstsc /v:instance /admin
```

자세한 내용은 Microsoft 문서([Access Remote Desktop Via Command Line](#))를 참조하십시오.

- 인스턴스를 중지하고 Amazon EBS 볼륨을 분리한 후 같은 가용 영역의 다른 인스턴스에 연결하여 데이터를 복구합니다.

"Some settings are managed by your organization. (일부 설정이 사용자의 조직에 의해 관리됩니다.)(Windows Server 2019)

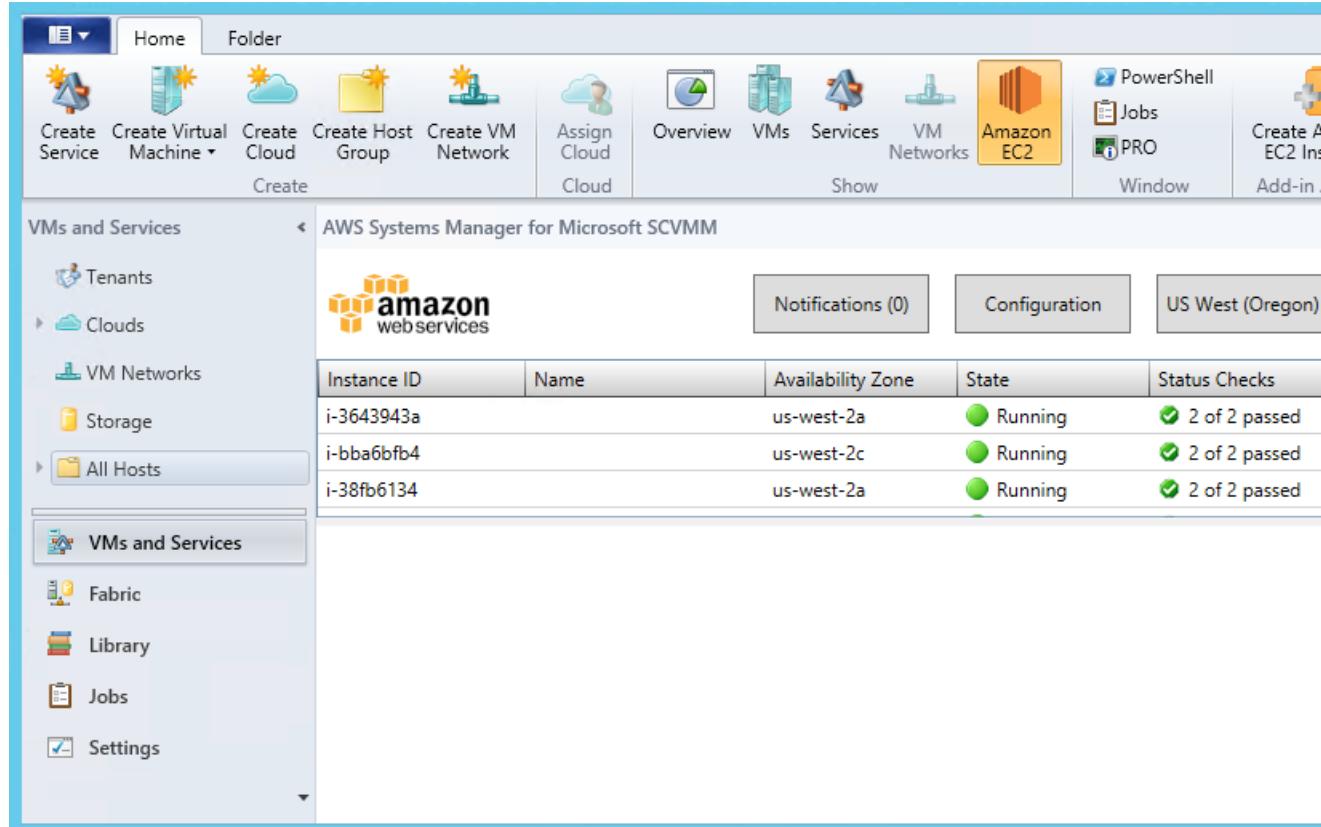
최신 Windows Server 2019 AMI에서 시작한 인스턴스에서 "Some settings are managed by your organization.(일부 설정이 사용자의 조직에 의해 관리됩니다.)"이라는 Windows 업데이트 대화 상자 메시지를 표시할 수 있습니다. 이 메시지는 Windows Server 2019의 변경 사항으로 인해 나타나며 업데이트 설정 관리 기능 또는 Windows 업데이트 동작에는 영향을 주지 않습니다.

이 경고를 제거하려면 다음 단계를 따릅니다.

1. gpedit.msc에서 computer Configuration(컴퓨터 구성) > Administrative Templates(관리 템플릿) > Windows Components(Windows 구성 요소) > Windows updates(Windows 업데이트)로 이동합니다. Configure Automatic Update(자동 업데이트 구성)을 편집하고 이를 활성으로 전환합니다.
2. 명령 프롬프트에서 gpupdate /force로 그룹 정책을 업데이트합니다.
3. Windows 업데이트 설정을 닫은 다음 다시 열니다. 사용자에게는 위와 같이 사용자의 설정이 사용자의 조직에 의해 관리된다는 메시지가 표시되며, 이어서 "We'll automatically download updates, except on metered connections (where charges may apply). In that case, we'll automatically download those updates required to keep Windows running smoothly.(측정된 연결(요금 발생 가능)을 제외하고 업데이트를 자동으로 다운로드합니다. 이 경우에는 원활한 Windows 실행 유지에 필요한 업데이트를 자동으로 다운로드합니다.)"라는 메시지도 표시됩니다.
4. 1단계의 gpedit.msc로 돌아가 그룹 정책을 구성되지 않음으로 재설정합니다. 또 다른 gpupdate /force를 수행합니다.
5. 명령 프롬프트를 닫고 몇 분간 기다립니다.
6. Windows 업데이트 설정 다시 열니다. "Some settings are managed by your organization(일부 설정이 사용자의 조직에 의해 관리됩니다.)"라는 메시지가 표시되지 않아야 합니다.

Microsoft System Center DPM용 AWS 시스템 관리자

Microsoft System Center Virtual Machine Manager(SCVMM)를 위한 AWS 시스템 관리자는 Microsoft SCVMM에서 EC2 인스턴스 등의 AWS 리소스를 관리하기 위한 쉽고 간편한 인터페이스를 제공하며, VMM 콘솔에 대한 추가 기능으로 구현됩니다. 자세한 정보는 [AWS Add-ins for Microsoft System Center](#)를 참조하십시오.



기능

- 관리자는 사용자에게 SCVMM에서 EC2 인스턴스를 관리할 수 있는 권한을 부여할 수 있습니다.
- 사용자는 필요한 권한이 있는 경우 인스턴스를 시작, 보기, 재부팅, 중지, 시작 및 종료할 수 있습니다.
- 사용자는 Windows 인스턴스에 대한 암호를 가져오고 RDP를 사용하여 인스턴스에 연결할 수 있습니다.
- 사용자는 Linux 인스턴스에 대한 퍼블릭 DNS 이름을 가져오고 SSH를 사용하여 인스턴스에 연결할 수 있습니다.
- 사용자는 SCVMM에서 Amazon EC2로 Hyper-V Windows 가상 머신을 가져올 수 있습니다.

제한 사항

- 사용자는 SCVMM에 로그인하는 데 사용할 수 있는 계정이 있어야 합니다.

- Linux 가상 머신을 SCVMM에서 Amazon EC2로 가져올 수 없습니다.
- AWS 리소스를 생성하고 관리하는 포괄적인 도구가 아닙니다. SCVMM 사용자는 추가 기능을 사용하여 EC2 인스턴스 관리를 위한 기본 작업을 빠르게 시작할 수 있습니다. 추가 AWS 리소스 관리는 향후 릴리스에서 지원될 수 있습니다.

요구 사항

- AWS 계정
- Microsoft System Center VMM 2012 R2 또는 System Center VMM 2012 SP1(최신 업데이트 릴리스 포함)

시작하기

시작하려면 다음 설명서를 참조하십시오.

- [설정 \(p. 1230\)](#)
- [EC2 인스턴스 관리 \(p. 1234\)](#)
- [문제 해결 \(p. 1241\)](#)

Microsoft SCVMM을 위해 AWS 시스템 관리자 설정

AWS 시스템 관리자를 설정하면 조직의 사용자가 AWS 리소스에 액세스할 수 있습니다. 이 프로세스는 계정 생성, 추가 기능 배포 및 자격 증명 제공으로 구성됩니다.

작업

- [AWS에 가입 \(p. 1230\)](#)
- [사용자에 대한 액세스 설정 \(p. 1230\)](#)
- [추가 기능 배포 \(p. 1233\)](#)
- [AWS 자격 증명 제공 \(p. 1233\)](#)

AWS에 가입

Amazon Web Services에 가입하면 AWS의 모든 서비스에 AWS 계정이 자동으로 등록됩니다. 사용한 서비스에 대해서만 청구됩니다.

이미 AWS 계정이 있다면 다음 작업으로 건너뛰십시오. AWS 계정이 없는 경우에는 아래 단계를 수행하여 계정을 만드십시오.

AWS 계정에 가입하려면 다음을 수행합니다.

1. <https://portal.aws.amazon.com/billing/signup>을 엽니다.
2. 온라인 지시 사항을 따릅니다.

등록 절차 중 전화를 받고 전화 키패드를 사용하여 확인 코드를 입력하는 과정이 있습니다.

사용자에 대한 액세스 설정

시스템 관리자를 처음으로 사용할 때 AWS 자격 증명을 제공해야 합니다. 여러 사용자가 고유한 자격 증명과 권한을 사용하여 동일한 AWS 계정에 액세스할 수 있게 하려면 각 사용자에 대해 IAM 사용자를 생성합니다.

제한된 작업을 수행하는 권한을 부여하는 정책으로 하나 이상의 그룹을 생성할 수 있습니다. 그런 다음 하나 이상의 IAM 사용자를 생성하고 각 사용자를 적절한 그룹에 추가할 수 있습니다.

Administrators 그룹을 생성하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 그룹을 선택한 다음, 새 그룹 생성을 선택합니다.
3. 그룹 이름 상자에서 **Administrators**를 지정하고 다음 단계를 선택합니다.
4. 정책 연결 페이지에서 AdministratorAccess AWS 관리형 정책을 선택합니다.
5. 다음 단계를 선택한 다음, 그룹 생성을 선택합니다.

Amazon EC2에 대한 제한된 권한을 가진 그룹을 생성하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 그룹을 선택한 다음, 새 그룹 생성을 선택합니다.
3. 그룹 이름 상자에서 그룹에 대한 의미 있는 이름을 지정한 다음, 다음 단계를 선택합니다.
4. 정책 연결 페이지에서 AWS 관리형 정책을 선택하지 않고 즉, 다음 단계를 선택한 다음 그룹 생성을 선택합니다.
5. 방금 만든 그룹 이름을 선택합니다. 권한 탭에서 인라인 정책을 선택한 다음, 여기를 클릭합니다.
6. 사용자 지정 정책 라디오 버튼을 선택한 다음 선택을 선택합니다.
7. 정책 이름과 Amazon EC2에 대한 제한된 액세스를 허용하는 정책 문서를 입력한 다음, 정책 적용을 선택합니다. 예를 들어, 다음 사용자 지정 정책 중 하나를 지정할 수 있습니다.

이 그룹의 사용자에게 EC2 인스턴스에 대한 정보만 보는 권한 부여

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "ec2:Describe*",  
                "iam>ListInstanceProfiles"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

이 그룹의 사용자에게 추가 기능에서 지원하는 EC2 인스턴스의 모든 작업을 수행할 권한 부여

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "iam>ListInstanceProfiles", "iam:PassRole",  
                "ec2:Describe*", "ec2>CreateKeyPair",  
                "ec2>CreateTags", "ec2>DeleteTags",  
                "ec2:RunInstances", "ec2:GetPasswordData",  
                "ec2:RebootInstances", "ec2:StartInstances",  
                "ec2:StopInstances", "ec2:TerminateInstances"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

}

이 그룹의 사용자에게 VM을 Amazon EC2로 가져올 권한 부여

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "s3>ListAllMyBuckets", "s3>CreateBucket",  
                "s3>DeleteBucket", "s3>DeleteObject",  
                "s3:GetBucketLocation", "s3GetObject",  
                "s3>ListBucket", "s3PutObject",  
                "ec2DescribeTags", "ec2CancelConversionTask",  
                "ec2DescribeConversionTasks", "ec2DescribeInstanceAttribute",  
                "ec2CreateImage", "ec2AttachVolume",  
                "ec2ImportInstance", "ec2ImportVolume",  
                "dynamodbDescribeTable", "dynamodbCreateTable",  
                "dynamodbScan", "dynamodbPutItem", "dynamodbUpdateItem"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

IAM 사용자를 생성하려면 사용자의 AWS 자격 증명을 가져오고 사용자에게 권한을 부여합니다.

1. 탐색 창에서 사용자와 사용자 추가를 차례로 선택합니다.
2. 사용자 이름을 입력합니다.
3. 이 사용자 세트에게 부여할 액세스 권한의 유형을 선택합니다. 이 사용자가 AWS Management 콘솔에도 액세스해야 하는 경우 프로그래밍 방식 액세스(Programmatic access)와 AWS Management Console 액세스(AWS Management Console access)를 선택합니다.
4. 콘솔 암호 유형(Console password type)에서 다음 중 하나를 선택합니다.
 - Autogenerated password(자동 생성된 비밀 번호). 각 사용자는 현재 유효한 암호 정책(있는 경우)에 따라 임의로 생성되는 암호를 받습니다. Final(최종) 페이지에 이르면 암호를 보거나 다운로드할 수 있습니다.
 - Custom password(사용자 지정 비밀 번호). 입력란에 입력하는 암호가 각 사용자에게 할당됩니다.
5. 다음: 권한(Next: Permissions)을 선택합니다.
6. 권한 설정 페이지에서 그룹에 사용자 추가를 선택합니다. 적절한 그룹을 선택합니다.
7. 다음: 검토(Next: Review)와 사용자 생성을 차례로 선택합니다.
8. 사용자의 액세스 키(액세스 키 ID와 보안 액세스 키)를 보려면 보고 싶은 각 암호와 보안 액세스 키 옆에 있는 표시를 선택합니다. 액세스 키를 저장하려면 .csv 다운로드(Download .csv)를 선택한 후 안전한 위치에 파일을 저장합니다.

Note

이 단계를 완료한 후에는 보안 액세스 키를 검색할 수 없으며, 키를 분실한 경우 새로 생성해야 합니다.

9. 닫기를 선택합니다.

추가 기능 배포

System Center VMM에 대한 추가 기능은 .zip 파일로 배포됩니다. 추가 기능을 배포하려면 다음 절차를 사용합니다.

추가 기능을 배포하려면

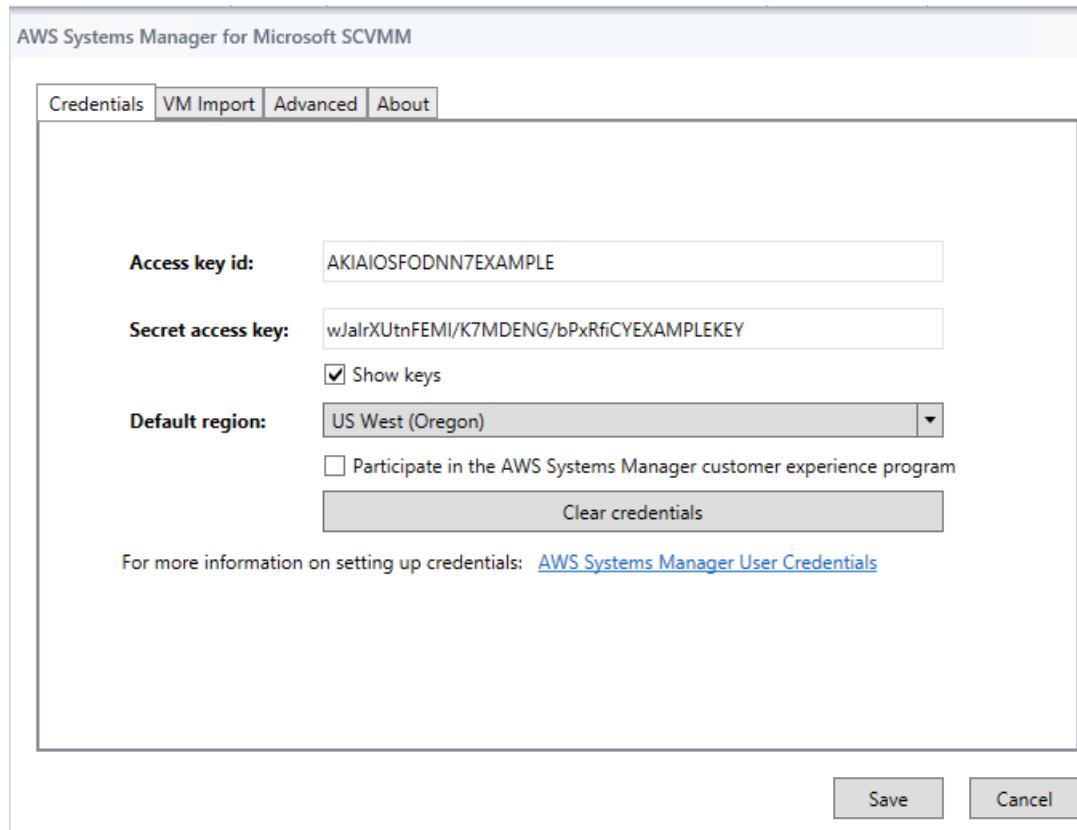
1. 인스턴스에서 [AWS 시스템 관리자 for Microsoft System Center Virtual Machine Manager](#)로 이동한 다음 SCVMM을 클릭합니다. aws-systems-manager-1.5.zip 파일을 인스턴스에 저장합니다.
2. VMM 콘솔을 엽니다.
3. 탐색 창에서 설정을 클릭한 다음 콘솔 추가 기능(Console Add-In)을 클릭합니다.
4. 리본에서 콘솔 추가 기능 가져오기(Import Console Add-in)를 클릭합니다.
5. 추가 기능 선택(Select an Add-in) 페이지에서 찾아보기를 클릭하고 다운로드한 추가 기능에 대한 aws-systems-manager-1.5.zip 파일을 선택합니다.
6. 신뢰할 수 있는 기관에서 서명하지 않은 어셈블리가 추가 기능에 있다는 경고를 무시합니다. 이 추가 기능을 계속 설치(Continue installing this add-in anyway)를 선택한 후 다음을 클릭합니다.
7. 요약 페이지에서 완료를 클릭합니다.
8. 추가 기능을 가져온 경우 작업은 Completed 상태가 됩니다. 작업 창을 닫을 수 있습니다.

AWS 자격 증명 제공

시스템 관리자를 처음 사용할 때 AWS 자격 증명을 제공해야 합니다. 액세스 키는 AWS에 사용자를 식별합니다. 두 가지 유형의 액세스 키 즉, 액세스 키 ID(예: AKIAIOSFODNN7EXAMPLE)와 보안 액세스 키(예: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY)가 있습니다. 액세스 키를 받으면 안전한 곳에 저장해야 합니다.

AWS 자격 증명을 제공하려면

1. VMM 콘솔을 엽니다.
2. 탐색 창에서 VM 및 서비스(VMs and Services)를 클릭합니다.
3. 리본에서 Amazon EC2를 클릭합니다.
4. 자격 증명 탭에서 AWS 자격 증명을 지정하고 기본 리전을 선택한 다음 저장을 클릭합니다.



언제든지 이러한 자격 증명을 변경하려면 구성요소를 클릭합니다.

AWS 시스템 관리자 for Microsoft SCVMM을 사용하여 EC2 인스턴스 관리

AWS 자격 증명을 사용하여 시스템 관리자 콘솔에 로그인한 후 EC2 인스턴스를 관리할 수 있습니다.

작업

- [EC2 인스턴스 생성 \(p. 1234\)](#)
- [인스턴스 보기 \(p. 1237\)](#)
- [인스턴스에 연결 \(p. 1237\)](#)
- [인스턴스 재부팅 \(p. 1238\)](#)
- [인스턴스 중지 \(p. 1238\)](#)
- [인스턴스 시작 \(p. 1238\)](#)
- [인스턴스 종료 \(p. 1238\)](#)

EC2 인스턴스 생성

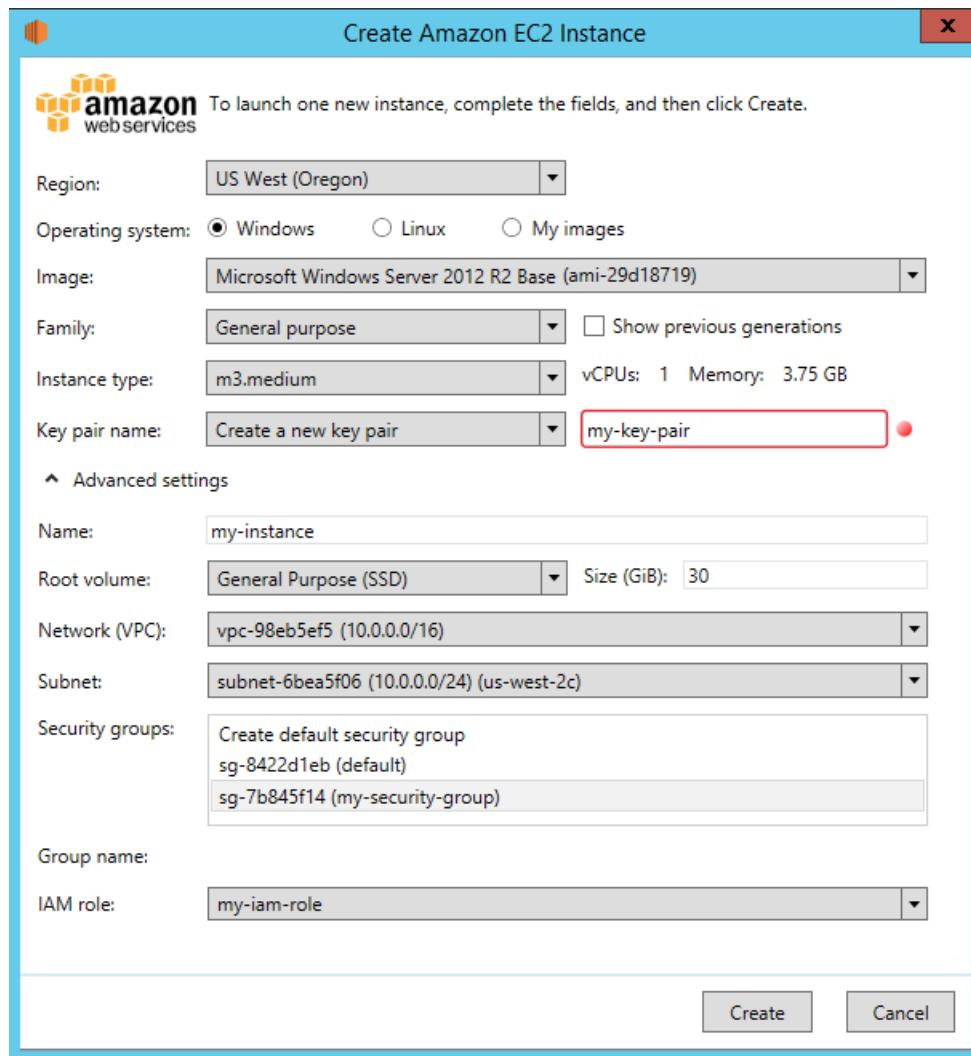
관리자가 부여한 권한에 따라 인스턴스를 생성할 수 있는지 여부가 결정됩니다.

사전 조건

- Virtual Private Cloud(VPC)의 가용 영역에 인스턴스를 시작할 서브넷이 있어야 합니다. VPC 생성에 대한 자세한 내용은 [Amazon VPC 시작 안내서](#) 단원을 참조하십시오.

EC2 인스턴스를 생성하려면

- SCVMM을 엽니다.
- 리본에서 Amazon EC2 인스턴스 생성(Create Amazon EC2 Instance)을 클릭합니다.
- Amazon EC2 인스턴스 생성 대화 상자에서 다음을 수행합니다.
 - 인스턴스에 해당하는 리전을 선택합니다. 기본 리전으로 구성된 리전이 기본적으로 선택되어 있습니다.
 - 인스턴스에 대한 템플릿(AMI라고 함)을 선택합니다. Amazon에서 제공하는 AMI를 사용하려면 Windows 또는 Linux를 선택한 다음 이미지에서 AMI를 선택합니다. 생성한 AMI를 사용하려면 내 이미지를 선택한 다음 이미지에서 AMI를 선택합니다.
 - 인스턴스에 대한 인스턴스 유형을 선택합니다. 패밀리에서 최신 인스턴스 패밀리 중 하나를 선택한 다음 인스턴스 유형에서 인스턴스 유형을 선택합니다. 목록에 이전 세대 인스턴스 패밀리를 포함하려면 이전 세대 표시>Show previous generations를 선택합니다. 자세한 내용은 [Amazon EC2 인스턴스 및 이전 세대 인스턴스](#)를 참조하십시오.
 - 키 페어를 생성하거나 선택합니다. 키 페어를 생성하려면 키 페어 이름에서 `create a new key pair`를 선택하고 강조 표시된 필드에 키 페어의 이름을 입력합니다(예: my-key-pair).
 - (선택 사항) 고급 설정에서 인스턴스에 대한 표시 이름을 지정합니다.
 - (선택 사항) 고급 설정] 아래의 네트워크(VPC)에서 VPC를 선택합니다. 이 목록에는 Amazon VPC 콘솔을 사용하여 생성한 VPC와 기본 VPC(있는 경우)를 비롯하여 리전에 대한 모든 VPC가 포함되어 있습니다. 이 리전에 기본 VPC가 있는 경우 기본적으로 해당 VPC가 선택됩니다. "There is no VPC available for launch or import operations in this region"이라는 텍스트가 표시되는 경우 Amazon VPC 콘솔을 사용하여 이 리전에서 VPC를 생성해야 합니다.
 - (선택 사항) 고급 설정 아래의 서브넷에서 서브넷을 선택합니다. 이 목록에는 기본 서브넷을 비롯하여 선택된 VPC에 대한 모든 서브넷이 포함되어 있습니다. 이 목록이 비어 있는 경우 Amazon VPC 콘솔을 사용하여 VPC에 서브넷을 추가하거나 다른 VPC를 선택해야 합니다. 그렇지 않으면 서브넷이 자동으로 선택됩니다.
 - (선택 사항) 고급 설정에서 보안 그룹을 생성하거나 하나 이상의 보안 그룹을 선택합니다. `Create default security group`을 선택하면 모든 사용자에게 RDP 및 SSH 액세스 권한을 부여하는 보안 그룹이 생성됩니다. Amazon EC2 또는 Amazon VPC 콘솔을 사용하여 보안 그룹을 수정할 수 있습니다. 그룹 이름 상자에 이 보안 그룹에 대한 이름을 입력할 수 있습니다.
 - (선택 사항) 고급 설정에서 IAM 역할을 선택합니다. 이 목록이 비어 있는 경우 IAM 콘솔을 사용하여 역할을 생성할 수 있습니다.



4. 생성을 클릭합니다. 키 페어를 생성 중인 경우 .pem 파일을 저장하라는 메시지가 나타납니다. 인스턴스에 로그인하는 데 필요하므로 이 파일을 안전한 곳에 저장하십시오. 인스턴스가 시작되었다는 확인 메시지가 나타납니다. Close(닫기)를 클릭합니다.

인스턴스를 생성하면 해당 인스턴스를 시작한 리전의 인스턴스 목록에 생성된 인스턴스가 표시됩니다. 처음에는 인스턴스가 pending 상태입니다. running 상태로 변경된 이후에 인스턴스를 사용할 수 있습니다.

이 페이지에 설명된 대로 시스템 관리자를 사용하여 인스턴스의 수명 주기를 관리할 수 있습니다. 다음을 비롯한 다른 작업을 수행하려면 AWS Management 콘솔을 사용해야 합니다.

- [인스턴스에 Amazon EBS 볼륨 연결 \(p. 950\)](#)
- [인스턴스와 탄력적 IP 주소 연결 \(p. 725\)](#)
- [종료 방지 기능 활성화 \(p. 456\)](#)

인스턴스 보기

관리자가 부여한 권한에 따라 인스턴스를 보고 인스턴스에 대한 자세한 정보를 가져올 수 있는지 여부가 결정됩니다.

인스턴스를 보고 자세한 정보를 가져오려면

1. [AWS 시스템 관리자 콘솔](#)을 엽니다.
2. 리전 목록에서 리전을 선택합니다.
3. 인스턴스 목록에서 인스턴스를 하나 이상 선택합니다.
4. 아래쪽 창에서 각 인스턴스 옆의 아래쪽 화살표를 클릭하여 인스턴스에 대한 자세한 정보를 봅니다.

^ i-343e9f3a (my-instance)

Virtual machine information		Networking
Instance ID:	i-343e9f3a	Public DNS name:
Name:	my-instance	Public IP address:
State:	Running	Private DNS name:
Launch time:	1/20/2015 12:26:48 PM -08:00 (1 minute ago)	ip-10-0-0-147.us-west-2.compute.internal
Instance type:	m3.medium	Private IP address:
Tenancy:	default	Vpc ID:
Image ID:	ami-29d18719	Subnet ID:
Operating system:	Windows	Network interfaces:

인스턴스에 연결

인스턴스를 시작할 때 지정된 키 페어에 대한 프라이빗 키(.pem 파일)가 있는 경우 EC2 인스턴스에 로그인할 수 있습니다. 인스턴스에 연결하는 데 사용할 도구는 Windows 인스턴스인지 Linux 인스턴스인지에 따라 다릅니다.

Windows EC2 인스턴스에 연결하려면

1. AWS 시스템 관리자를 엽니다.
2. 인스턴스 목록에서 인스턴스를 선택하고 마우스 오른쪽 버튼으로 클릭한 다음 Windows 암호 찾기(Retrieve Windows Password)를 클릭합니다.
3. 기본 Windows 관리자 암호 찾기(Retrieve Default Windows Administrator Password) 대화 상자에서 찾아보기를 클릭합니다. 키 페어에 대한 프라이빗 키 파일을 선택한 다음 열기를 클릭합니다.
4. 암호 해독을 클릭합니다. 암호를 저장하거나 클립보드에 복사합니다.
5. 인스턴스를 선택하고 마우스 오른쪽 버튼으로 클릭한 다음 RDP를 통해 연결(Connect via RDP)을 클릭합니다. 자격 증명을 묻는 메시지가 표시되면 관리자 계정의 이름과 이전 단계에서 저장한 암호를 사용합니다.
6. 인증서가 자체 서명되어 신뢰할 수 있는 인증 기관의 보안 인증서가 아니라는 경고가 나타날 수 있습니다. 계속하려면 예를 클릭합니다.

연결에 실패할 경우 Windows 인스턴스용 Amazon EC2 사용 설명서의 [Troubleshooting Windows Instances](#)를 참조하십시오.

Linux EC2 인스턴스에 연결하려면

1. AWS 시스템 관리자를 엽니다.
2. 인스턴스 목록에서 인스턴스를 선택합니다.
3. 아래쪽 창에서 인스턴스 ID 옆의 아래쪽 화살표를 클릭하여 인스턴스에 대한 자세한 정보를 봅니다.

4. 퍼블릭 DNS 이름을 찾습니다. 인스턴스에 연결하려면 이 정보가 필요합니다.
5. PuTTY를 사용하여 인스턴스에 연결합니다. 단계별 지침은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Connect to Your Linux Instance from Windows Using PuTTY](#)를 참조하십시오.

인스턴스 재부팅

관리자가 부여한 권한에 따라 인스턴스를 재부팅할 수 있는지 여부가 결정됩니다.

인스턴스를 재부팅하려면

1. AWS 시스템 관리자를 엽니다.
2. 인스턴스 목록에서 인스턴스를 선택합니다.
3. 인스턴스를 마우스 오른쪽 버튼으로 클릭한 후 재설정(재부팅)(Reset (Reboot))을 클릭합니다.
4. 확인 메시지가 표시되면 예를 클릭합니다.

인스턴스 중지

관리자가 부여한 권한에 따라 인스턴스를 중지할 수 있는지 여부가 결정됩니다.

인스턴스를 중지하려면

1. AWS 시스템 관리자를 엽니다.
2. 인스턴스 목록에서 인스턴스를 선택합니다.
3. 인스턴스를 마우스 오른쪽 버튼으로 클릭한 후 종료(정지)(Shut Down (Stop))를 클릭합니다.
4. 확인 메시지가 표시되면 예를 클릭합니다.

인스턴스 시작

관리자가 부여한 권한에 따라 인스턴스를 시작할 수 있는지 여부가 결정됩니다.

인스턴스를 시작하려면

1. AWS 시스템 관리자를 엽니다.
2. 인스턴스 목록에서 인스턴스를 선택합니다.
3. 인스턴스를 마우스 오른쪽 버튼으로 클릭한 후 작동(시작)(Power On (Start))을 클릭합니다.
4. 확인 메시지가 표시되면 예를 클릭합니다.

인스턴스를 시작하려 할 때 할당량 오류가 발생하면 동시 실행 인스턴스 제한에 도달한 것입니다. AWS 계정의 기본 제한은 20입니다. 추가 실행 인스턴스가 필요할 경우 [Amazon EC2 인스턴스 제한 증가 요청](#)에서 양식을 작성하십시오.

인스턴스 종료

관리자가 부여한 권한에 따라 인스턴스를 종료할 수 있는지 여부가 결정됩니다.

인스턴스를 종료하려면

1. AWS 시스템 관리자를 엽니다.
2. 인스턴스 목록에서 인스턴스를 선택합니다.

3. 인스턴스를 마우스 오른쪽 버튼으로 클릭한 후 삭제(종료)(Delete (Terminate))를 클릭합니다.
4. 확인 메시지가 표시되면 예를 클릭합니다.

AWS 시스템 관리자 for Microsoft SCVMM을 사용하여 가상 머신 가져오기

SCVMM에서 Amazon EC2로 가져온 가상 머신에서 EC2 인스턴스를 시작할 수 있습니다.

Important

Linux 가상 머신을 SCVMM에서 Amazon EC2로 가져올 수 없습니다.

내용

- [사전 조건 \(p. 1239\)](#)
- [가상 머신 가져오기 \(p. 1239\)](#)
- [가져오기 작업 상태 확인 \(p. 1240\)](#)
- [가져온 인스턴스 백업 \(p. 1241\)](#)

사전 조건

- VM이 준비되어 있는지 확인합니다. 자세한 정보는 VM Import/Export 사용 설명서에서 [VM 준비](#)를 참조하십시오.
- AWS 시스템 관리자에서 구성을 클릭하고 VM 가져오기 탭을 선택한 후 다음 설정을 검토합니다.
 - S3 버킷 접두사: 업로드 할 디스크 이미지를 가져오기 전에 디스크 이미지에 대한 버킷을 생성합니다. 버킷의 이름은 여기에 나열된 접두사로 시작하고 리전을 포함합니다(예: us-east-2). 디스크 이미지를 가져온 후 삭제하려면 가져오기 후 S3 버킷 정리(Clean up S3 bucket after import)를 선택합니다.
 - VM 이미지 내보내기 경로(VM image export path): VM에서 내보낸 디스크 이미지의 위치입니다. 디스크 이미지를 가져온 후 삭제하려면 가져오기 후 내보내기 경로 정리(Clean up export path after import)를 선택합니다.
 - 대체 Hyper-V PowerShell 모듈 경로(Alternate Hyper-V PowerShell module path): Hyper-V PowerShell 모듈의 위치입니다(스탠다드 위치에 설치되지 않은 경우). 자세한 내용은 Microsoft TechNet 라이브러리의 [To install the Hyper-V Management Tools](#)를 참조하십시오.

가상 머신 가져오기

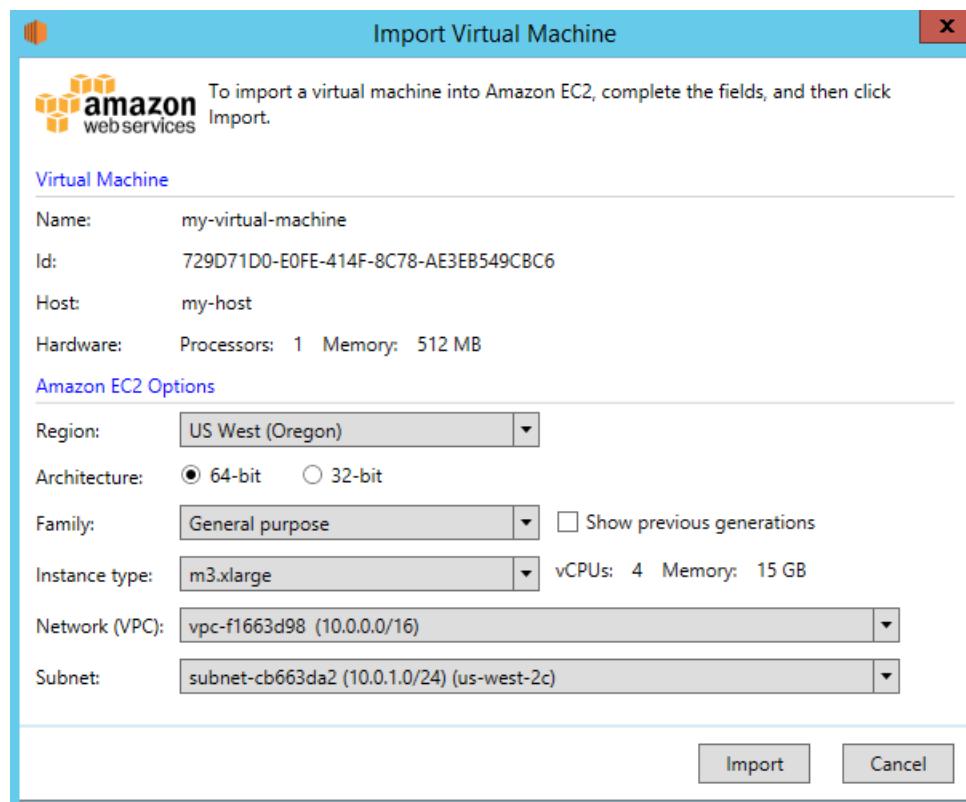
관리자가 부여한 권한에 따라 HyperV Windows 가상 머신을 SCVMM에서 AWS로 가져올 수 있는지 여부가 결정됩니다.

가상 머신을 가져오려면

1. SCVMM을 엽니다.
2. 리본에서 VM을 클릭합니다. 목록에서 가상 머신을 선택합니다.
3. 리본에서 Amazon EC2로 VM 가져오기(Import VM to Amazon EC2)를 클릭합니다.
4. 가상 머신 가져오기(Import Virtual Machine) 대화 상자에서 다음 작업을 수행합니다.
 - a. 인스턴스에 해당하는 리전을 선택합니다. 기본 리전으로 구성된 리전이 기본적으로 선택되어 있습니다.
 - b. 인스턴스에 대한 인스턴스 유형을 선택합니다. 패밀리에서 최신 인스턴스 패밀리 중 하나를 선택한 다음 인스턴스 유형에서 인스턴스 유형을 선택합니다. 목록에 이전 세대 인스턴스 패밀리를 포함하

려면 이전 세대 표시(Show previous generations)를 선택합니다. 자세한 내용은 [Amazon EC2 인스턴스 및 이전 세대 인스턴스](#)를 참조하십시오.

- c. 네트워크(VPC)(Network (VPC))에서 VPC를 선택합니다. 이 목록에는 Amazon VPC 콘솔을 사용하여 생성한 VPC와 기본 VPC(있는 경우)를 비롯하여 리전에 대한 모든 VPC가 포함되어 있습니다. 이 리전에 기본 VPC가 있는 경우 기본적으로 해당 VPC가 선택됩니다. "There is no VPC available for launch or import operations in this region"이라는 텍스트가 표시되는 경우 Amazon VPC 콘솔을 사용하여 이 리전에서 VPC를 생성해야 합니다.
- d. 서브넷[에서 서브넷을 선택합니다. 이 목록에는 기본 서브넷을 비롯하여 선택된 VPC에 대한 모든 서브넷이 포함되어 있습니다. 이 목록이 비어 있는 경우 Amazon VPC 콘솔을 사용하여 VPC에 서브넷을 추가하거나 다른 VPC를 선택해야 합니다. 그렇지 않으면 서브넷이 자동으로 선택됩니다.



5. 가져오기를 클릭합니다. VM 가져오기(VM Import) 탭에서 필요한 정보를 지정하지 않은 경우 필요한 정보를 제공하라는 오류 메시지가 나타납니다. 그렇지 않으면 가져오기 작업이 시작되었다는 확인 메시지가 나타납니다. Close(닫기)를 클릭합니다.

가져오기 작업 상태 확인

가져오기 작업을 완료하는 데 몇 시간 정도 걸릴 수 있습니다. 현재 상태를 보려면 AWS System Manager를 열고 알림을 클릭합니다.

가져오기 작업이 진행되면서 다음과 같은 알림이 표시됩니다.

- Import VM: Created Import VM Task
- Import VM: Export VM Disk Image Done
- Import VM: Upload to S3
- Import VM: Image Conversion Starting

- Import VM: Image Conversion Done
- Import VM: Import Complete

변환되는 각 디스크 이미지에 대해 Import VM: Upload to S3, Import VM: Image Conversion Starting 및 Import VM: Image Conversion Done 알림이 표시됩니다.

가져오기 작업이 실패하면 Import VM: Import Failed 알림이 표시됩니다. 가져오기 작업 관련 문제를 해결하는 방법에 대한 자세한 내용은 [VM 가져오기 오류 \(p. 1242\)](#) 단원을 참조하십시오.

가져온 인스턴스 백업

가져오기 작업이 완료되면 인스턴스는 종료될 때까지 실행됩니다. 인스턴스가 종료된 이후에는 인스턴스에 연결하거나 인스턴스를 복구할 수 없습니다. 필요한 경우 가져온 인스턴스와 동일한 소프트웨어로 새 인스턴스를 시작할 수 있게 하려면 가져온 인스턴스에서 Amazon 머신 이미지(AMI)를 생성합니다. 자세한 내용은 [사용자 지정 Windows AMI 생성 \(p. 84\)](#) 단원을 참조하십시오.

Microsoft SCVMM을 위해 AWS 시스템 관리자 문제 해결

다음은 일반적인 오류 및 문제 해결 단계입니다.

내용

- [오류: 추가 기능을 설치할 수 없음 \(p. 1241\)](#)
- [설치 오류 \(p. 1241\)](#)
- [로그 파일 확인 \(p. 1242\)](#)
- [VM 가져오기 오류 \(p. 1242\)](#)
- [추가 기능 제거 \(p. 1242\)](#)

오류: 추가 기능을 설치할 수 없음

다음 오류 메시지가 표시되는 경우 VMM 콘솔을 실행 중인 컴퓨터에서 [KB2918659](#)를 설치해 보십시오. 자세한 내용은 [System Center 2012 서비스 팩 1 업데이트 릴리스 5 설명](#)을 참조하십시오. 이 문제를 해결하기 위해 이 기사에 나열된 모든 업데이트를 설치할 필요는 없으며, KB2918659만 설치하면 됩니다.

```
Add-in cannot be installed
The assembly "Amazon.Scvmm.Addin" referenced to by add-in component "AWS Systems Manager
for
Microsoft SCVMM" could not be found in the add-in package. This could be due to the
following
reasons:
1. The assembly was not included with the add-in package.
2. The AssemblyName attribute for the add-in does not match the name of the add-in
assembly.
3. The assembly file is corrupt and cannot be loaded.
```

설치 오류

설치 중 다음 오류 중 하나가 발생하는 경우 SCVMM 문제가 원인일 수 있습니다.

```
Could not update managed code add-in pipeline due to the following error:
```

Access to the path 'C:\Program Files\Microsoft System Center 2012\Virtual Machine Manager \Bin\AddInPipeline\PipelineSegments.store' is denied.

Could not update managed code add-in pipeline due to the following error:
The required folder 'C:\Program Files\Microsoft System Center 2012\Virtual Machine Manager \Bin\AddInPipeline\HostSideAdapters' does not exist.

Add-in cannot be installed
The assembly "Microsoft.SystemCenter.VirtualMachineManager.UIAddIns.dll" referenced by the add-in assembly "Amazon.ScVMM.AddIn" could not be found in the add-in package. Make sure that this assembly was included with the add-in package.

이 문제를 해결하려면 다음 단계 중 하나를 수행해 보십시오.

- 인증된 사용자에게 C:\Program Files\Microsoft System Center 2012\Virtual Machine Manager\Bin\AddInPipeline 폴더를 읽고 실행할 수 있는 권한을 부여합니다. Windows 탐색기에서 폴더를 마우스 오른쪽 버튼으로 클릭하고 속성을 선택한 다음 보안 탭을 선택합니다.
- SCVMM 콘솔을 닫았다가 관리자로 시작합니다. 시작 메뉴에서 SCVMM을 찾은 다음 마우스 오른쪽 버튼으로 클릭하고 관리자 권한으로 실행을 선택합니다.

로그 파일 확인

추가 기능을 사용하는 데 문제가 있으면 생성된 로그 파일 %APPDATA%\Amazon\SCVMM\ec2addin.log에서 유용한 정보가 있는지 확인합니다.

VM 가져오기 오류

로그 파일 %APPDATA%\Amazon\SCVMM\ec2addin.log에는 가져오기 작업의 상태에 대한 자세한 정보가 포함되어 있습니다. 다음은 SCVMM에서 Amazon EC2로 VM을 가져올 때 로그 파일에서 기록될 수 있는 일반적인 오류입니다.

오류: Hyper-V VirtualMachine 객체를 추출할 수 없음

해결 방법: Hyper-V PowerShell 모듈에 대한 경로를 구성합니다.

오류: 작업을 수행할 수 있는 권한이 없음

이 오류는 일반적으로 Hyper-V가 VM 이미지를 구성된 경로에 저장할 수 없을 때 발생합니다. 이 문제를 해결하려면 다음과 같이 실행하십시오.

- Hyper-V 서버에 디렉터리를 만듭니다. 예: C:\vmimages.
- Hyper-V에 방금 만든 디렉터리를 공유합니다. SCVMM을 실행하는 사용자에게 디렉터리 액세스 권한이 주어집니다.
- 플러그인에서 내보내기 경로를 \\hyperv\vmimages로 설정합니다.
- 내보내기를 수행합니다.

이미지는 Hyper-V 서버의 로컬 디렉터리로 내보내집니다. SCVMM 플러그인은 Hyper-V에서 이미지를 가져와 Amazon S3에 업로드합니다.

추가 기능 제거

추가 기능을 제거해야 하는 경우 다음 절차를 사용합니다.

추가 기능을 제거하려면

1. VMM 콘솔을 엽니다.
2. 설정 작업 영역을 선택한 다음 콘솔 추가 기능(Console Add-Ins)을 클릭합니다.
3. Microsoft SCVMM용 AWS 시스템 관리자를 선택합니다.
4. 리본에서 제거를 클릭합니다.
5. 확인 메시지가 표시되면 예를 클릭합니다.

추가 기능을 설치한 후 다시 설치할 때 다음과 같은 오류가 발생하는 경우 오류 메시지에 제안된 경로를 삭제합니다.

```
Error (27301)
There was an error while installing the add-in. Please ensure that the following path does
not
exist and then try the installation again.

C:\Program Files\Microsoft System Center 2012\Virtual Machine Manager\Bin\AddInPipeline\
AddIns\EC2WINDOWS...
```

AWS Management Pack for Microsoft System Center

Amazon Web Services(AWS)는 클라우드에서 엔터프라이즈 애플리케이션과 빅 데이터 프로젝트부터 소셜 게임과 모바일 앱에 이르는 거의 모든 것을 실행할 수 있도록 완전한 인프라 및 애플리케이션 서비스 세트를 제공합니다.— AWS Management Pack for Microsoft System Center는 AWS에서 실행 중인 애플리케이션에 대한 가용성 및 성능 모니터링 기능을 제공합니다.

AWS Management Pack를 사용하면 Microsoft System Center Operations Manager에서 인스턴스와 볼륨 등의 AWS 리소스에 액세스하여 성능 데이터를 수집하고 AWS 리소스를 모니터링할 수 있습니다. AWS Management Pack은 System Center Operations Manager를 확장한 것입니다. AWS Management Pack은 System Center 2012 — Operations Manager 및 System Center Operations Manager 2007 R2에 각각 사용 할 수 있는 두 가지 버전이 있습니다.

AWS Management Pack은 Amazon CloudWatch 지표와 경보를 사용하여 AWS 리소스를 모니터링합니다. Amazon CloudWatch 지표는 Microsoft System Center에 성능 카운터로 표시되고, Amazon CloudWatch 경보는 알림으로 표시됩니다.

다음과 같은 리소스를 모니터링할 수 있습니다.

- EC2 인스턴스
- EBS 볼륨
- ELB 로드 밸런서
- Amazon EC2 Auto Scaling 그룹 및 가용 영역
- Elastic Beanstalk 애플리케이션
- CloudFormation 스택
- CloudWatch 경보
- CloudWatch 사용자 지정 측정치

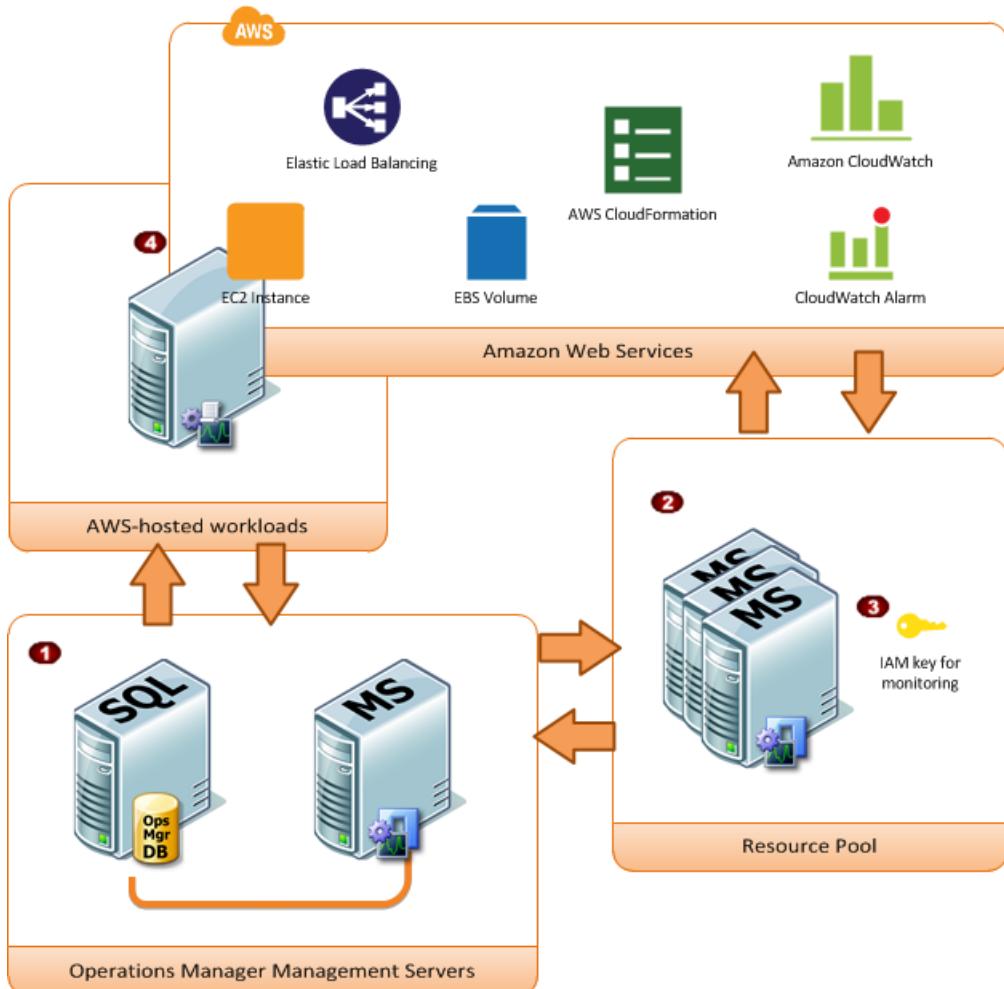
내용

- [AWS Management Pack for System Center 2012 개요 \(p. 1244\)](#)
- [AWS Management Pack for System Center 2007 R2 개요 \(p. 1246\)](#)
- [AWS Management Pack 다운로드 \(p. 1247\)](#)
- [AWS Management Pack 배포 \(p. 1248\)](#)
- [AWS Management Pack 사용 \(p. 1260\)](#)
- [AWS Management Pack 업그레이드 \(p. 1279\)](#)
- [AWS Management Pack 제거 \(p. 1280\)](#)
- [AWS Management Pack 문제 해결 \(p. 1281\)](#)

AWS Management Pack for System Center 2012 개요

AWS Management Pack for System Center 2012 — Operations Manager에서는 하나 이상의 관리 서버를 포함하는 리소스 풀을 사용하여 AWS 리소스를 검색하고 모니터링합니다. 사용하는 AWS 리소스가 늘어나면 풀에 관리 서버를 추가할 수 있습니다.

다음 다이어그램은 AWS Management Pack의 주요 구성 요소를 보여 줍니다.



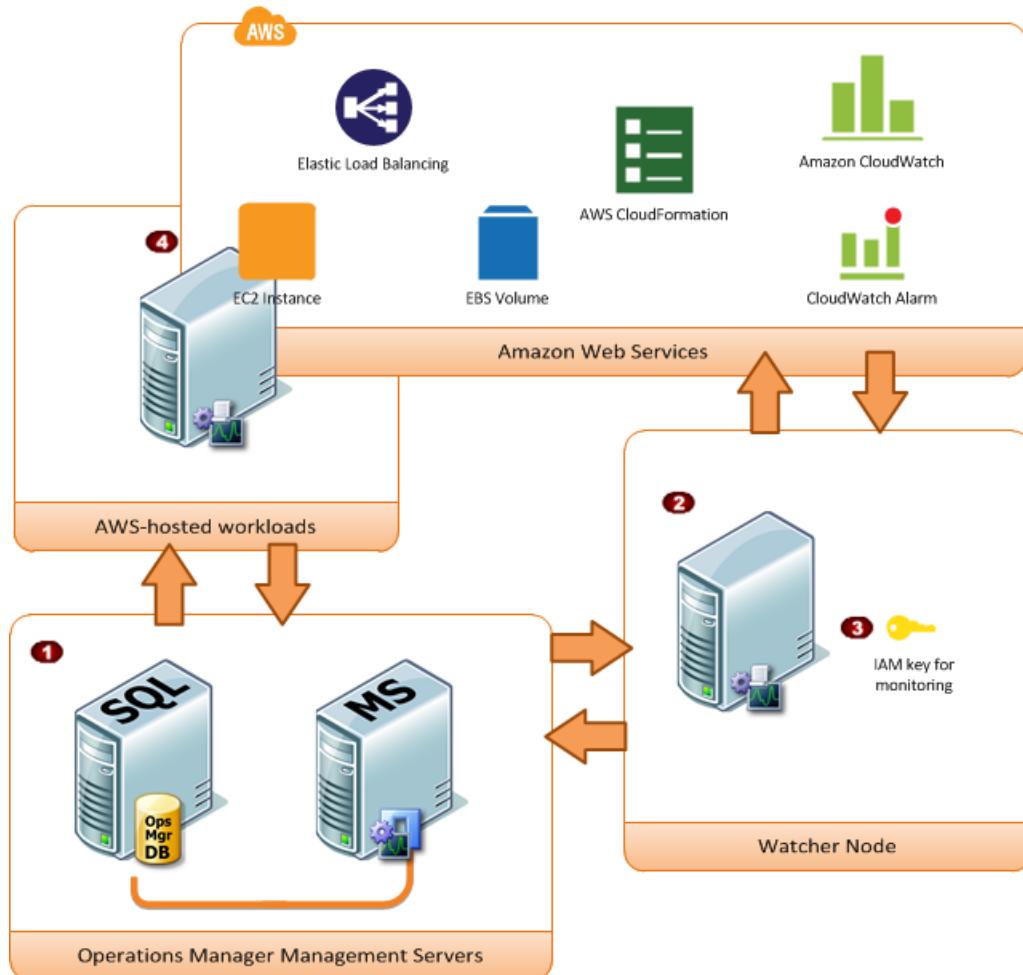
항목	구성 요소	설명
①	Operations Manager 인프라	하나 이상의 관리 서버와 종속 항목(예: Microsoft SQL Server 및 Microsoft Active Directory 도메인)입니다. 이러한 서버를 온-premises 또는 AWS 클라우드에 배포할 수 있으며, 두 시나리오 모두 지원됩니다.
②	리소스 풀	.NET용 AWS SDK를 사용하여 AWS와 통신하는 데 사용되는 하나 이상의 관리 서버입니다. 이러한 서버는 인터넷에 연결되어 있어야 합니다.
③	AWS 자격 증명	관리 서버에서 AWS API를 호출하는 데 사용되는 액세스 키 ID 및 보안 액세스 키입니다. AWS Management Pack을 구성하는 동안 이러한 자격 증명을 지정해야 합니다. 읽기 전용 권한이 있는 IAM 사용자를 생성하고 해당 자격 증명을 사용하는 것이 좋습니다. IAM 사용자 생성에 대한 자세한 내용은 IAM 사용 설명서에서 Adding a New User to Your AWS Account 를 참조하십시오.
④	EC2 인스턴스	AWS 클라우드에서 실행 중인 가상 컴퓨터입니다. 인스턴스에 따라 Operations Manager Agent 설치 여부가 달립니다.

항목	구성 요소	설명
		Operations Manager Agent를 설치할 때 인스턴스 상태 이외에 운영 체제 및 애플리케이션 상태를 확인할 수 있습니다.

AWS Management Pack for System Center 2007 R2 개요

AWS Management Pack for System Center Operations Manager 2007 R2에서는 System Center 환경에 연결하고 인터넷에 액세스할 수 있는 지정된 컴퓨터(감시자 노드)를 사용해 AWS API를 호출하여 AWS 리소스에 대한 정보를 원격으로 검색하고 수집합니다.

다음 다이어그램은 AWS Management Pack의 주요 구성 요소를 보여 줍니다.



항목	구성 요소	설명
①	Operations Manager 인프라	하나 이상의 관리 서버와 종속 항목(예: Microsoft SQL Server 및 Microsoft Active Directory 도메인)입니다. 이러한 서버를 온프레미스 또는 AWS 클라우드에 배포할 수 있으며, 두 시나리오 모두 지원됩니다.

항목	구성 요소	설명
②	감시자 노드	.NET용 AWS SDK를 사용하여 AWS와 통신하는 데 사용되는 지정된 에이전트 관리형 컴퓨터입니다. 감시자 노드는 온프레미스나 AWS 클라우드에 배포할 수 있지만, 에이전트 관리형 컴퓨터여야 하며, 인터넷에 연결되어 있어야 합니다. 감시자 노드를 하나만 사용하여 AWS 계정을 모니터링할 수 있습니다. 그러나 하나의 감시자 노드에서 여러 AWS 계정을 모니터링할 수 있습니다. 감시자 노드 설정에 대한 자세한 내용은 Microsoft System Center 설명서의 Windows 에이전트 배포 를 참조하십시오.
③	AWS 자격 증명	감시자 노드에서 AWS API를 호출하는데 사용되는 액세스 키 ID 및 보안 액세스 키입니다. AWS 관리 팩을 구성하는 동안 이러한 자격 증명을 지정해야 합니다. 읽기 전용 권한이 있는 IAM 사용자를 생성하고 해당 자격 증명을 사용하는 것이 좋습니다. IAM 사용자 생성에 대한 자세한 내용은 IAM 사용 설명서에서 Adding a New User to Your AWS Account 를 참조하십시오.
④	EC2 인스턴스	AWS 클라우드에서 실행 중인 가상 컴퓨터입니다. 인스턴스에 따라 Operations Manager Agent 설치 여부가 다릅니다. Operations Manager Agent를 설치할 때 인스턴스 상태 이외에 운영 체제 및 애플리케이션 상태를 확인할 수 있습니다.

AWS Management Pack 다운로드

시작하려면 AWS Management Pack를 다운로드합니다. AWS Management Pack은 무료입니다. 모니터링 구성 방법이나 모니터링하는 AWS 리소스 수에 따라 Amazon CloudWatch 요금이 발생할 수 있습니다.

System Center 2012

AWS Management Pack을 다운로드하기 전에 시스템이 다음 시스템 요구 사항과 사전 조건을 충족하는지 확인하십시오.

시스템 요구 사항

- System Center Operations Manager 2012 R2 또는 System Center Operations Manager 2012 SP1
- 누적 업데이트 1 이상. AWS 리소스를 모니터링하는 관리 서버와 AWS Management Pack에서 모니터링 할 감시자 노드 및 에이전트를 실행 중인 에이전트에 업데이트를 배포해야 합니다. AWS 리소스를 모니터링하는 모든 컴퓨터에 최신 Operations Manager 업데이트를 배포하는 것이 좋습니다.
- Microsoft.Unix.Library MP 버전 7.3.2026.0 이상

사전 조건

- 데이터 센터에는 인터넷에 연결할 수 있는 관리 서버가 한 대 이상 구성되어 있어야 합니다. 관리 서버에 Microsoft .NET Framework 버전 4.5 이상 및 PowerShell 2.0 이상이 설치되어 있어야 합니다.
- 관리 서버에 대한 작업 계정에 관리 서버에 대한 로컬 관리자 권한이 있어야 합니다.

AWS Management Pack을 다운로드하려면

1. [AWS Add-Ins for Microsoft System Center](#) 웹 사이트에서 SCOM 2012를 클릭합니다.
2. AWS-SCOM-MP-2.5.zip을 컴퓨터에 저장하고 압축을 풁니다.

AWS Management Pack 배포 (p. 1248) 항목을 계속 진행합니다.

System Center 2007 R2

AWS Management Pack를 다운로드하기 전에 시스템이 다음 시스템 요구 사항과 사전 조건을 충족하는지 확인하십시오.

시스템 요구 사항

- System Center Operations Manager 2007 R2
- Microsoft.Unix.Library MP 버전 6.1.7000.256 이상

사전 조건

- 데이터 센터에 인터넷에 연결할 수 있고 감시자 노드로 지정된 에이전트 관리형 컴퓨터가 있어야 합니다. 감시자 노드에서 Allow this agent to act as a proxy and discover managed objects on other computers(이 에이전트가 프록시 역할을 하고 다른 컴퓨터에서 관리된 객체를 검색하도록 허용) 에이전트 프록시 옵션을 활성화해야 합니다. 감시자 노드에 Microsoft .NET Framework 버전 3.5.1 이상 및 PowerShell 2.0 이상이 설치되어 있어야 합니다.
- 감시자 노드의 작업 계정에 감시자 노드에 대한 로컬 관리자 권한이 있어야 합니다.
- 감시자 노드에 에이전트가 설치되어 있고, 감시자 노드에서 인터넷에 액세스할 수 있으며 데이터 센터의 관리 서버와 통신할 수 있는지 확인해야 합니다. 자세한 내용은 Microsoft System Center 설명서의 [Windows 에이전트 배포](#)를 참조하십시오.

AWS Management Pack를 다운로드하려면

1. [AWS Add-Ins for Microsoft System Center](#) 웹 사이트에서 SCOM 2007을 클릭합니다.
2. AWS-MP-Setup-2.5.msi를 컴퓨터로 저장합니다.

AWS Management Pack 배포 (p. 1248) 항목을 계속 진행합니다.

AWS Management Pack 배포

AWS Management Pack를 배포하기 전에 다운로드해야 합니다. 자세한 내용은 [AWS Management Pack 다운로드 \(p. 1247\)](#) 단원을 참조하십시오.

작업

- 1단계: AWS Management Pack 설치 (p. 1248)
- 2단계: 감시자 노드 구성 (p. 1250)
- 3단계: AWS Run As 계정 생성 (p. 1250)
- 4단계: Add Monitoring Wizard 실행 (p. 1254)
- 5단계: 포트 및 앤드포인트 구성 (p. 1259)

1단계: AWS Management Pack 설치

AWS Management Pack를 다운로드한 후 하나 이상의 AWS 계정을 모니터링하도록 구성해야 합니다.

System Center 2012

AWS Management Pack를 설치하려면

1. Operations 콘솔의 이동 메뉴에서 관리를 클릭한 후 Management Packs(관리 팩)을 클릭합니다.
2. 작업 창에서 Import Management Packs(관리 팩 가져오기)를 클릭합니다.
3. Select Management Packs(관리 팩 선택) 페이지에서 추가를 클릭한 후 Add from disk(디스크에서 추가)를 클릭합니다.
4. Select Management Packs to import(가져올 관리 팩 선택) 대화 상자에서 Amazon.AmazonWebServices.mpb 파일을 다운로드한 위치에서 이 파일을 선택한 후 열기를 클릭합니다.
5. Select Management Packs(관리 팩 선택) 페이지의 가져오기 목록에서 Amazon Web Services 관리 팩을 선택한 후 설치를 클릭합니다.

Note

System Center Operations Manager는 가져오기 목록에서 오류 아이콘이 표시된 관리 팩을 가져오지 않습니다.

6. Import Management Packs(관리 팩 가져오기) 페이지에 가져오기 프로세스의 진행률이 표시됩니다. 문제가 발생할 경우 상태 정보를 확인하려면 목록에서 관리 팩을 선택합니다. Close(닫기)를 클릭합니다.

System Center 2007 R2

AWS Management Pack를 설치하려면

관리 팩은 Microsoft 시스템 설치 관리자 파일(AWS-MP-Setup.msi)로 배포됩니다. 관리 팩에는 Amazon.AmazonWebServices.mp 파일과 감시자 노드, 루트 관리 서버 및 운영 콘솔에 필요한 DLL이 포함되어 있습니다.

1. 실행 AWS-MP-Setup.msi.

Note

루트 관리 서버, 운영 콘솔 및 감시자 노드가 다른 컴퓨터에 있는 경우 각 컴퓨터에서 설치 관리를 실행해야 합니다.

2. Welcome to the Amazon Web Services Management Pack Setup Wizard(Amazon Web Services 관리 팩 설정 마법사 시작) 화면에서 다음을 클릭합니다.
3. 최종 사용자 라이선스 계약 화면에서 라이선스 계약을 읽고, 내용에 동의할 경우 I accept the terms in the License Agreement(라이선스 계약 약관에 동의) 확인란을 선택하고 다음을 클릭합니다.
4. Custom Setup(사용자 지정 설정) 화면에서 설치할 기능을 선택하고 다음을 클릭합니다.

Operations Console

Amazon.AmazonWebServices.UI.Pages.dll을 설치한 후 GAC(Global Assembly Cache)에 등록하고 Amazon.AmazonWebServices.mp를 설치합니다.

Root Management Server

Amazon.AmazonWebServices.Modules.dll,
Amazon.AmazonWebServices.SCOM.SDK.dll 및 .NET용 AWS SDK(AWSSDK.dll)를 설치한 다음 GAC에 등록합니다.

AWS Watcher Node

Amazon.AmazonWebServices.Modules.dll 및
Amazon.AmazonWebServices.SCOM.SDK.dll을 설치한 다음 .NET용 AWS SDK(AWSSDK.dll)를 설치하고 GAC에 등록합니다.

5. Ready to install Amazon Web Services Management Pack(Amazon Web Services 관리 팩 설치 준비 완료) 화면에서 설치를 클릭합니다.
6. Completed the Amazon Web Services Management Pack Setup Wizard(Amazon Web Services 관리 팩 설치 마법사 완료) 화면에서 마침을 클릭합니다.

Note

필수 DLL이 복사되어 GAC에 등록되고 운영 콘솔을 실행 중인 컴퓨터의 Program Files (x86)/Amazon Web Services Management Pack 폴더에 관리 팩 파일(*.mp)이 복사됩니다. 그런 다음 관리 팩을 System Center로 가져와야 합니다.

7. Operations 콘솔의 이동 메뉴에서 관리를 클릭한 후 Management Packs(관리 팩)을 클릭합니다.
8. 작업 창에서 Import Management Packs(관리 팩 가져오기)를 클릭합니다.
9. Select Management Packs(관리 팩 선택) 페이지에서 추가를 클릭한 후 Add from disk(디스크에서 추가)를 클릭합니다.
10. Select Management Packs to import(가져올 관리 팩 선택) 대화 상자에서 딜렉터리를 c:\Program Files (x86)\Amazon Web Services Management Pack으로 변경하고 Amazon.AmazonWebServices.mp 파일을 선택한 후 열기를 클릭합니다.
11. Select Management Packs(관리 팩 선택) 페이지의 가져오기 목록에서 Amazon Web Services 관리 팩을 선택한 후 설치를 클릭합니다.

Note

System Center Operations Manager는 가져오기 목록에서 오류 아이콘이 표시된 관리 팩을 가져오지 않습니다.

12. Import Management Packs(관리 팩 가져오기) 페이지에 가져오기 프로세스의 진행률이 표시됩니다. 문제가 발생할 경우 상태 정보를 확인하려면 목록에서 관리 팩을 선택합니다. Close(닫기)를 클릭합니다.

2단계: 감시자 노드 구성

System Center Operations Manager 2007 R2에서 감시자 노드는 감시자 노드 컴퓨터 밖에서도 검색을 실행 하므로 감시자 노드에서 프록시 에이전트 옵션을 활성화해야 합니다. 프록시 에이전트를 사용하면 검색에서 다른 컴퓨터의 객체에 액세스할 수 있습니다.

Note

시스템이 많은 리소스로 구성되어 있는 경우 하나의 관리 서버를 감시자 노드로 구성하는 것이 좋습니다. 별도의 감시자 노드 관리 서버가 있으면 성능을 향상할 수 있습니다.

System Center 2012 — Operations Manager를 사용할 경우 이 단계를 건너뛸 수 있습니다.

System Center Operations Manager 2007 R2에서 프록시 에이전트를 활성화하려면

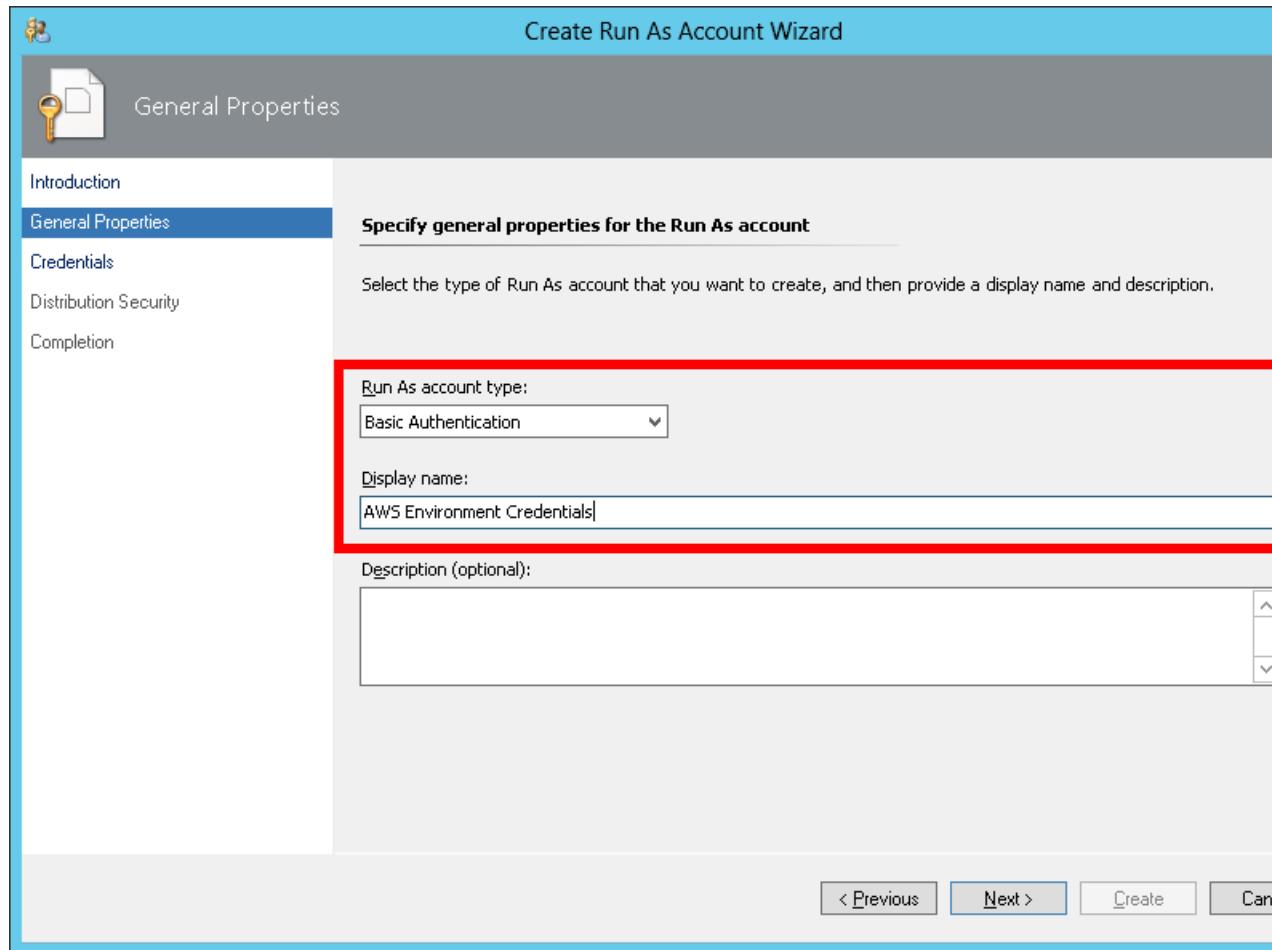
1. 운영 콘솔의 이동 메뉴에서 관리를 클릭합니다.
2. 관리 작업 영역의 디바이스 관리에서 Agent Managed(관리되는 에이전트)를 클릭합니다.
3. Agent Managed(관리되는 에이전트) 목록에서 감시자 노드를 마우스 오른쪽 버튼으로 클릭한 후 속성을 클릭합니다.
4. Agent Properties(에이전트 속성) 대화 상자에서 보안 탭을 클릭하고 Allow this agent to act as proxy and discover managed objects on other computers(이 에이전트가 프록시 역할을 하고, 다른 컴퓨터에서 관리되는 객체를 검색하도록 허용)를 선택한 후 확인을 클릭합니다.

3단계: AWS Run As 계정 생성

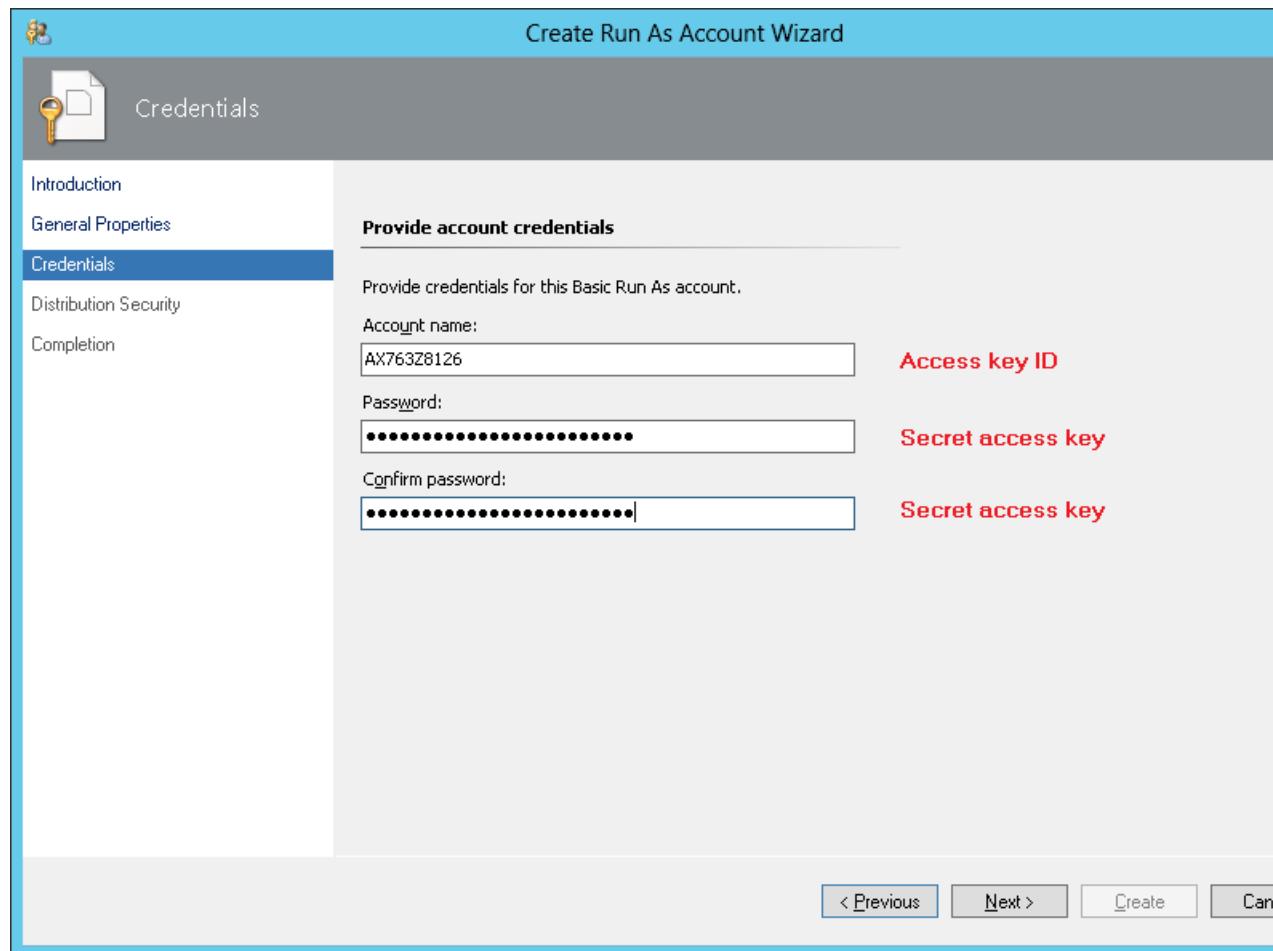
AWS 리소스에 대한 AWS Management Pack 액세스를 허용하는 자격 증명을 설정해야 합니다.

AWS Run As 계정을 생성하려면

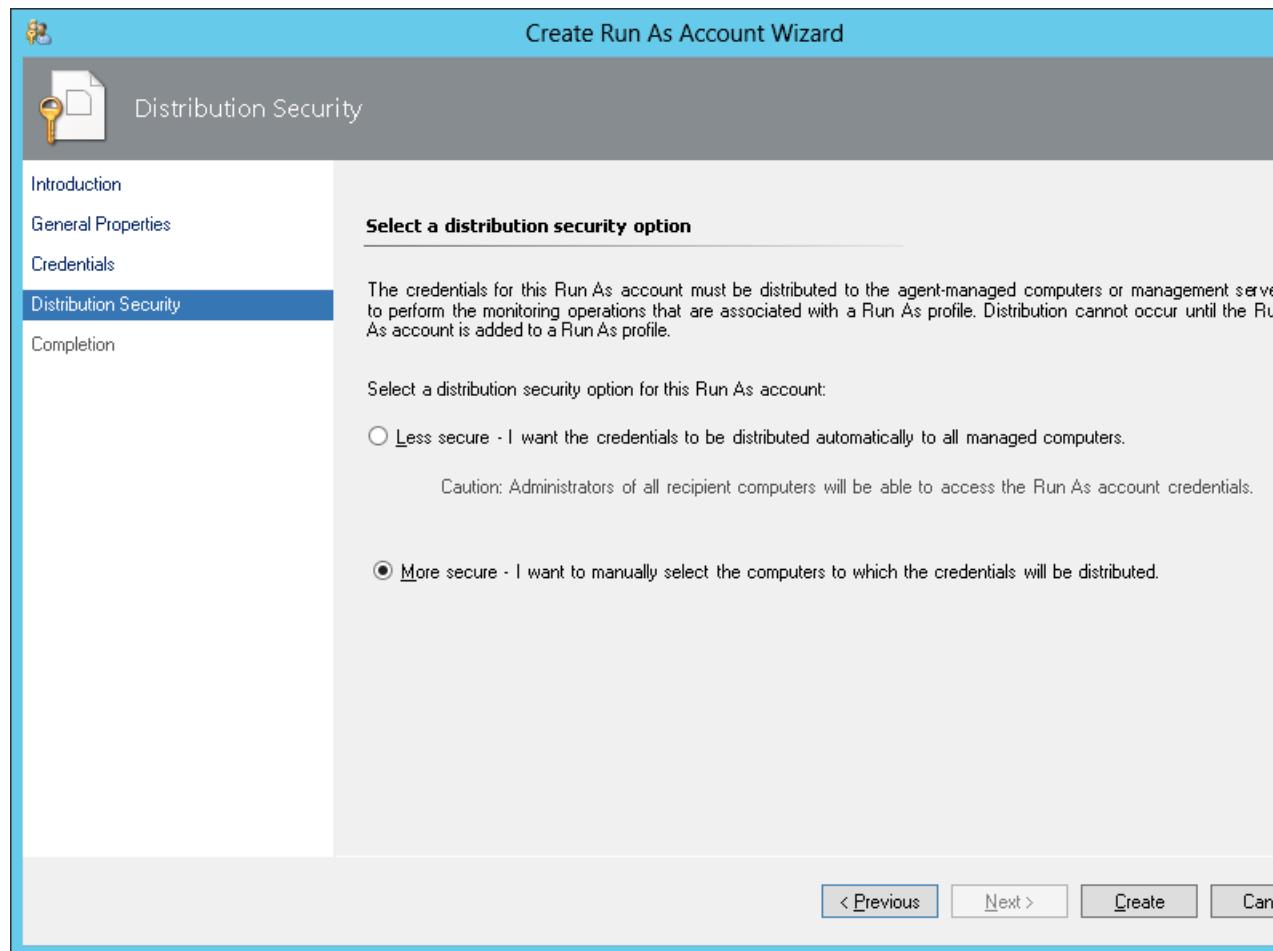
- 필요한 최소 액세스 권한을 가진 IAM 사용자를 생성하는 것이 좋습니다. 예를 들어, ReadOnlyAccess AWS 관리형 정책이 대부분의 경우에 적용됩니다. 절차를 완료하려면 이 사용자에 대한 액세스 키(액세스 키 ID 및 보안 액세스 키)가 필요합니다. 자세한 내용은 IAM 사용 설명서에서 [IAM 사용자의 액세스 키 관리](#)를 참조하십시오.
- 운영 콘솔의 이동 메뉴에서 관리를 클릭합니다.
- 관리 작업 영역에서 Run As Configuration(구성으로 실행) 노드를 확장한 후 계정을 선택합니다.
- 계정 창을 마우스 오른쪽 버튼으로 클릭한 후 Create Run As Account(Run As 계정 만들기)를 클릭합니다.
- Create Run As Account Wizard(Run As 계정 만들기 마법사)의 일반 속성 페이지에 있는 Run As account type(Run As 계정 유형) 목록에서 기본 인증을 선택합니다.
- 표시 이름(예: "My IAM Account")과 설명을 입력한 후 다음을 클릭합니다.



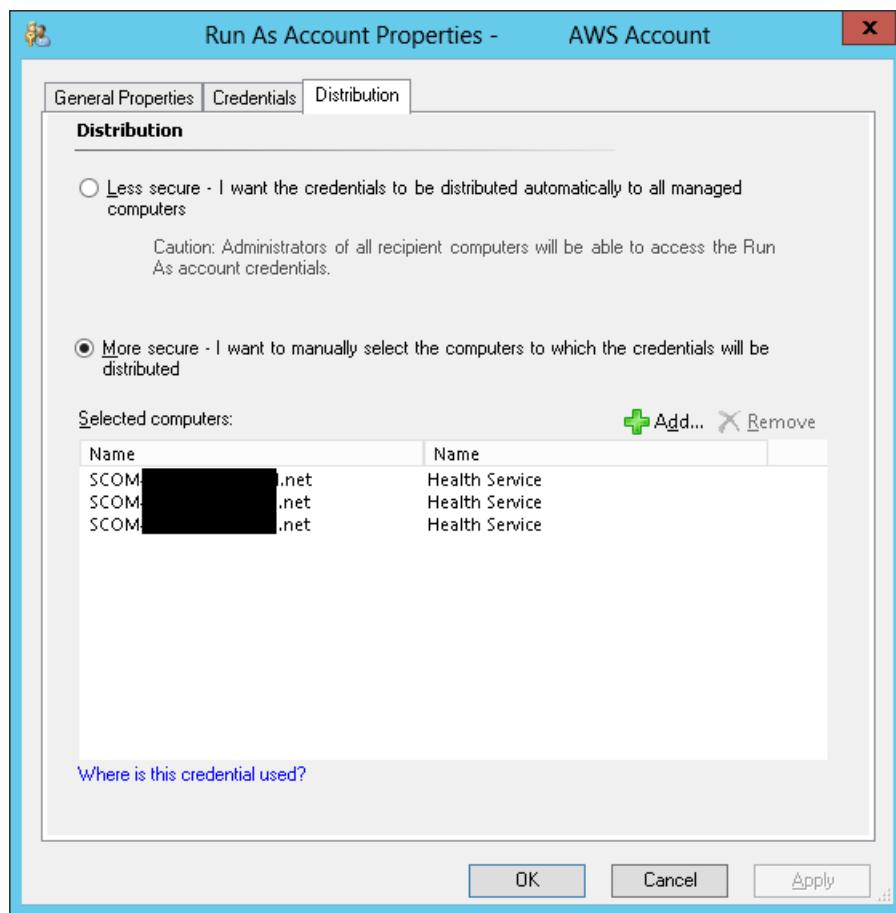
- 자격 증명 페이지의 계정 이름 상자에 액세스 키 ID를 입력하고 암호 상자에 보안 액세스 키를 입력한 후 다음을 클릭합니다.



8. 배포 보안 페이지에서 More secure - I want to manually select the computers to which the credentials will be distributed(보안 강화 - 자격 증명을 배포할 컴퓨터를 직접 선택하겠습니다)를 선택한 후 생성을 클릭합니다.



9. Close(닫기)를 클릭합니다.
10. 계정 목록에서 방금 생성한 계정을 선택합니다.
11. 작업 창에서 속성을 클릭합니다.
12. 속성 대화 상자에서 More Secure(보안 강화) 옵션이 선택되어 있고 AWS 리소스를 모니터링하는 데 사용할 모든 관리 서버가 나열되는지 확인합니다.



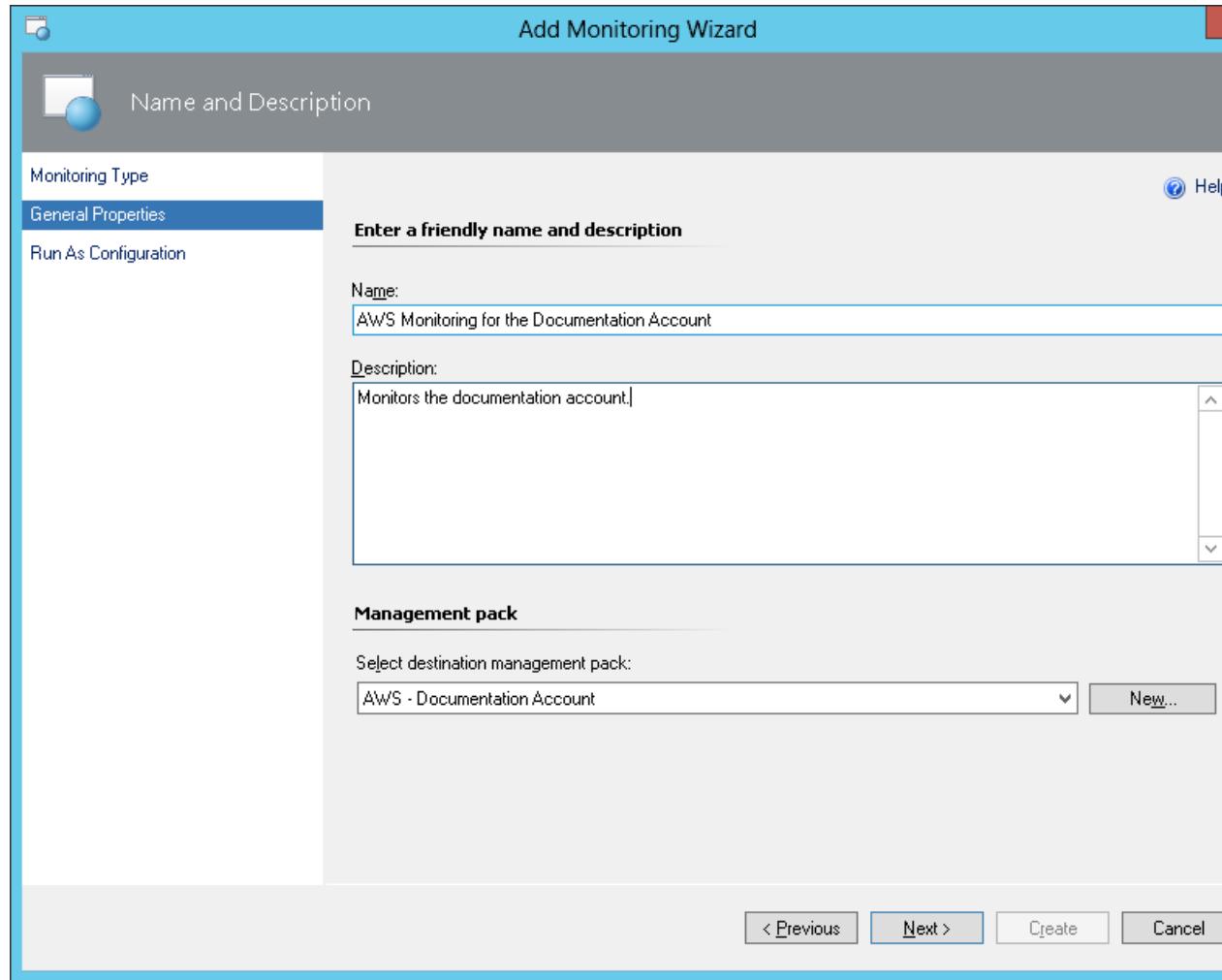
4단계: Add Monitoring Wizard 실행

운영 콘솔의 Authoring(저작) 작업 영역에서 제공하는 모니터링 추가 마법사를 사용하여 특정 AWS 계정을 모니터링하도록 AWS Management Pack를 구성할 수 있습니다. 이 마법사에서는 모니터링할 AWS 계정에 대한 설정을 포함하는 관리 팩을 생성합니다. 이 마법사를 실행하여 각 AWS 계정을 모니터링해야 합니다. 예를 들어, 두 AWS 계정을 모니터링하려면 마법사를 두 번 실행해야 합니다.

System Center 2012

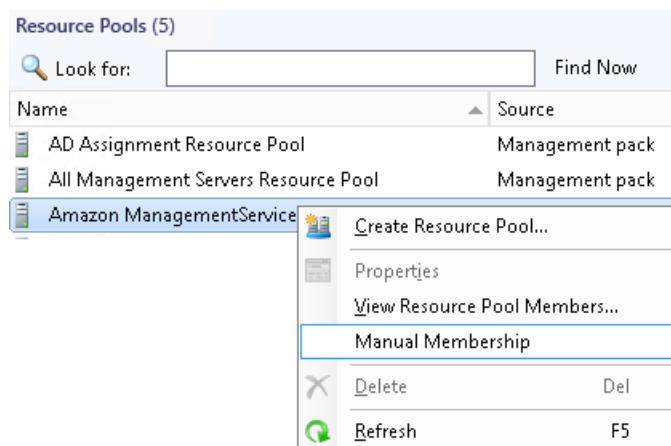
System Center 2012 — Operations Manager에서 Add Monitoring Wizard를 실행하려면

1. 운영 콘솔의 이동 메뉴에서 Authoring(저작)을 클릭합니다.
2. Authoring(저작) 작업 영역에서 Management Pack Templates(관리 팩 템플릿) 노드를 확장하고 Amazon Web Services를 마우스 오른쪽 버튼으로 클릭한 후 Add Monitoring Wizard(모니터링 추가 마법사)를 클릭합니다.
3. Add Monitoring Wizard(모니터링 추가 마법사)의 Select the monitoring type(모니터링 유형 선택) 목록에서 Amazon Web Services를 선택한 후 다음을 클릭합니다.
4. 일반 속성 창의 이름 상자에 이름(예: "My AWS Resources")을 입력합니다. 설명 상자에 설명을 입력합니다.
5. Select destination management pack(대상 관리 팩 선택) 목록에서 설정을 저장할 기존 관리 팩을 선택하거나 New(새로 만들기)를 클릭하여 새 관리 팩을 생성합니다. 다음을 클릭합니다.

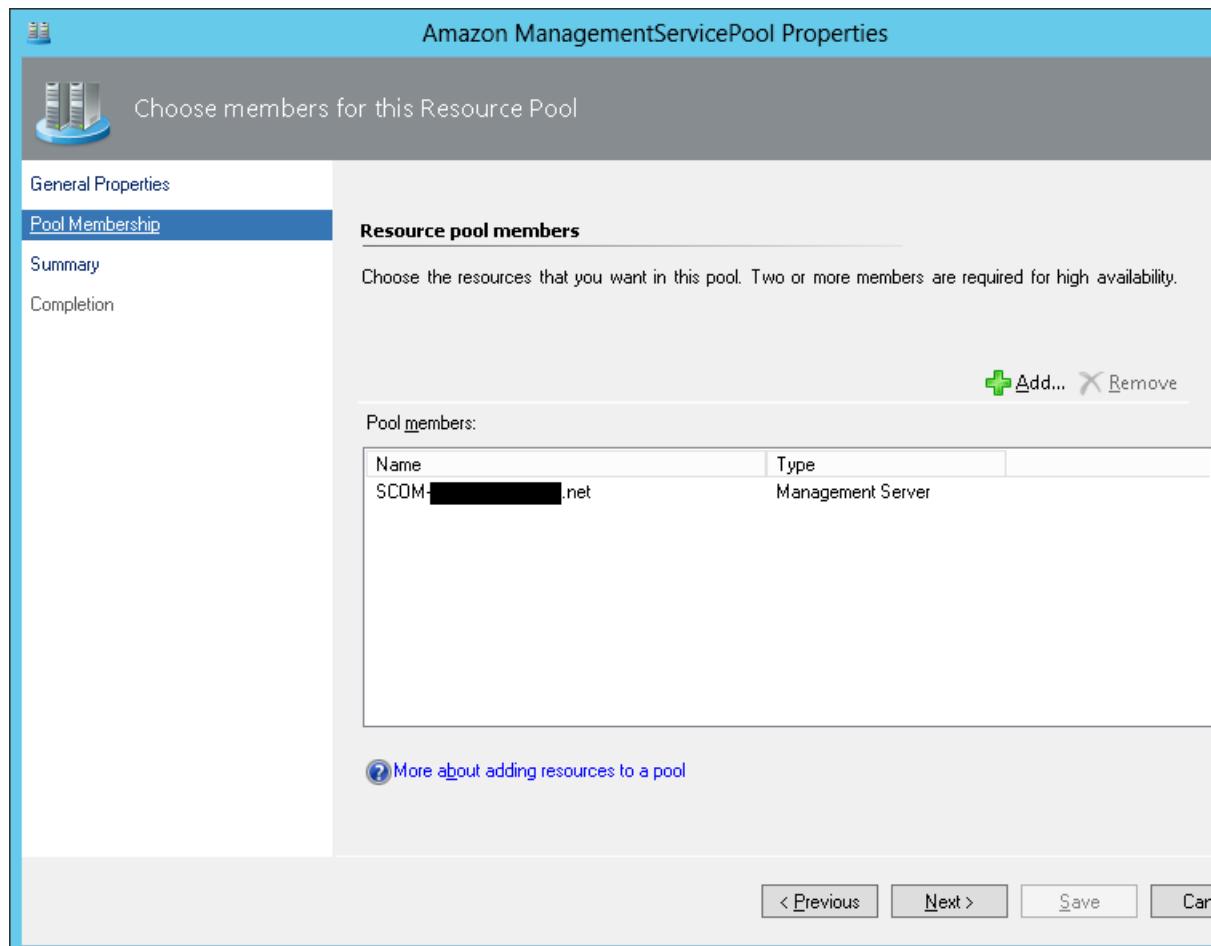


기본적으로 관리 팩 객체를 생성할 때 규칙 또는 모니터를 비활성화하거나 재정의를 생성합니다. 그러면 Operations Manager에서 기본 관리 팩에 설정을 저장합니다. 사용자 지정 설정을 기본 관리 팩에 저장하는 대신 사용자 지정할 봉인된 각 관리 팩에 대해 별도의 관리 팩을 생성해야 합니다.

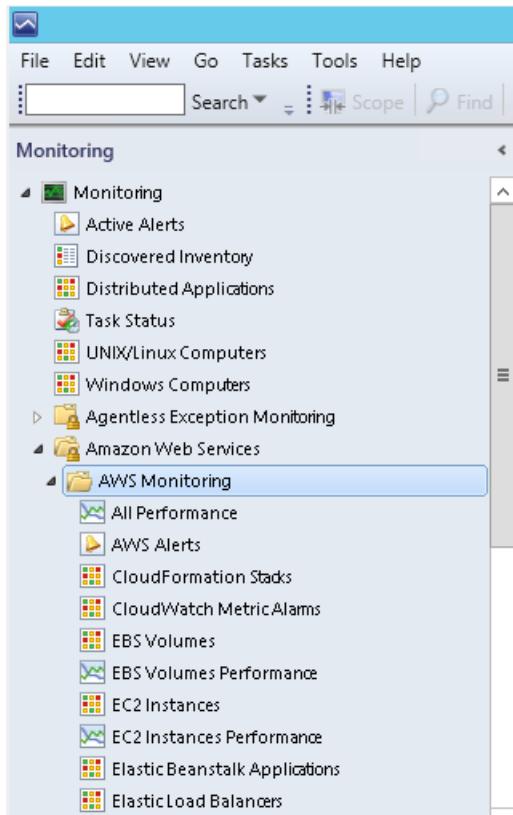
6. AWS Management Pack에서 자동으로 리소스 폴을 생성하고 관리 서버를 추가합니다. 서버 멤버십을 제어하려면 다음과 같이 변경합니다.
 - a. 이동 메뉴에서 관리를 클릭합니다.
 - b. Resource Pools(리소스 폴) 노드를 클릭합니다.
 - c. Resource Pools(리소스 폴) 창에서 AWS Resource Pool(AWS 리소스 폴)을 마우스 오른쪽 버튼으로 클릭하고 Manual Membership(수동 멤버십)을 선택합니다.



- d. Resource Pools(리소스 폴) 창에서 AWS Resource Pool(AWS 리소스 폴)을 마우스 오른쪽 버튼으로 클릭하고 속성을 선택합니다.
- e. Pool Membership(풀 멤버십) 페이지에서 AWS 리소스를 모니터링하지 않을 관리 서버를 제거합니다.



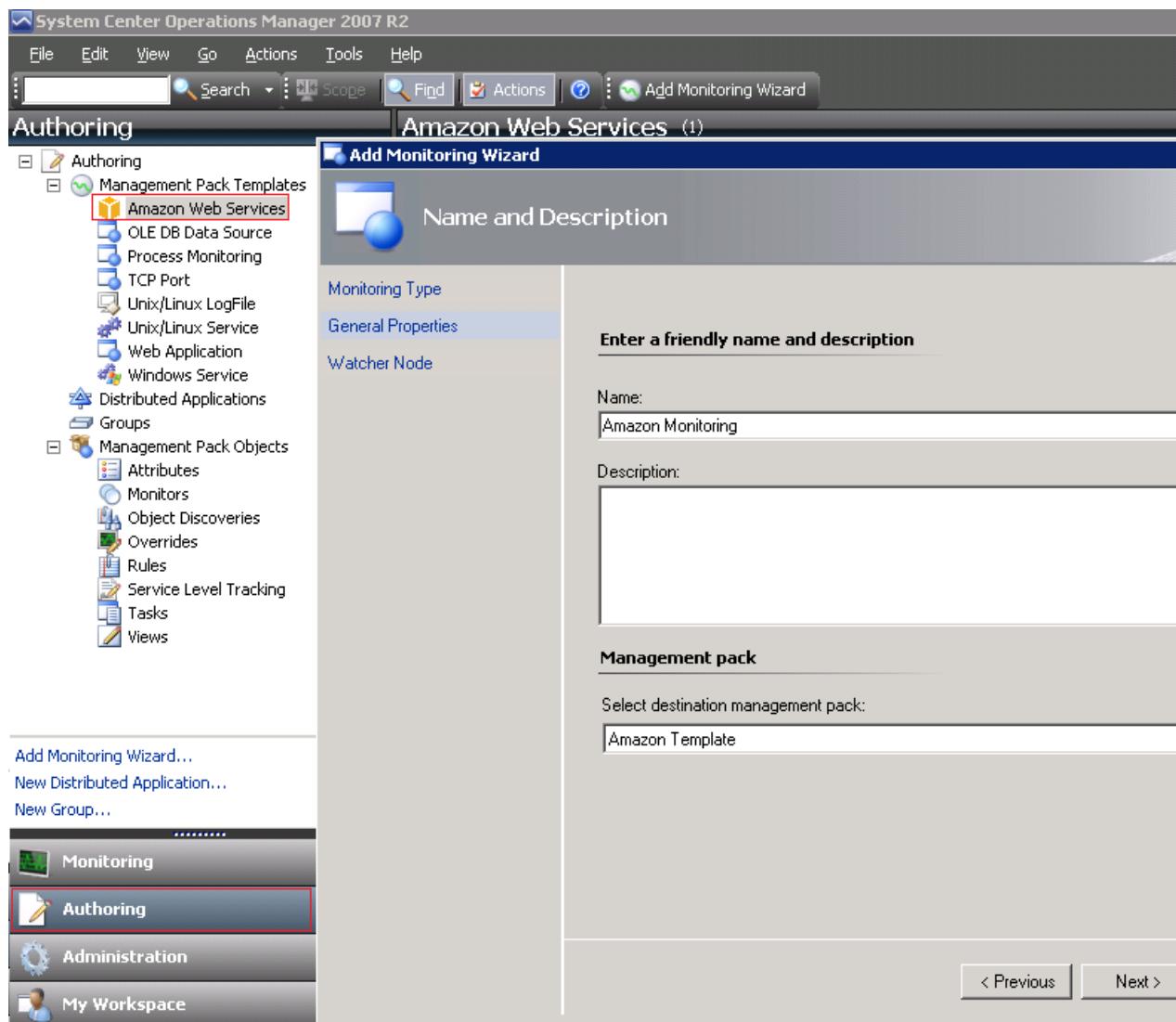
7. 구성된 AWS Management Pack은 운영 콘솔의 모니터링 작업 영역에 Amazon Web Services 폴더의 하위 폴더로 표시됩니다.



System Center 2007 R2

System Center Operations Manager 2007에서 Add Monitoring Wizard를 실행하려면

1. 운영 콘솔의 이동 메뉴에서 Authoring(저작)을 클릭합니다.
2. Authoring(저작) 작업 영역에서 Management Pack Templates(관리 팩 템플릿) 노드를 확장하고 Amazon Web Services를 마우스 오른쪽 버튼으로 클릭한 후 Add Monitoring Wizard(모니터링 추가 마법사)를 클릭합니다.
3. Add Monitoring Wizard(모니터링 추가 마법사)의 Select the monitoring type(모니터링 유형 선택) 목록에서 Amazon Web Services를 선택한 후 다음을 클릭합니다.
4. 일반 속성 창의 이름 상자에 이름(예: "My AWS Resources")을 입력합니다. 설명 상자에 설명을 입력합니다.
5. Select destination management pack(대상 관리 팩 선택) 드롭다운 목록에서 설정을 저장할 기존 관리 팩을 선택하거나 New(새로 만들기)를 클릭하여 새 관리 팩을 생성합니다. 다음을 클릭합니다.



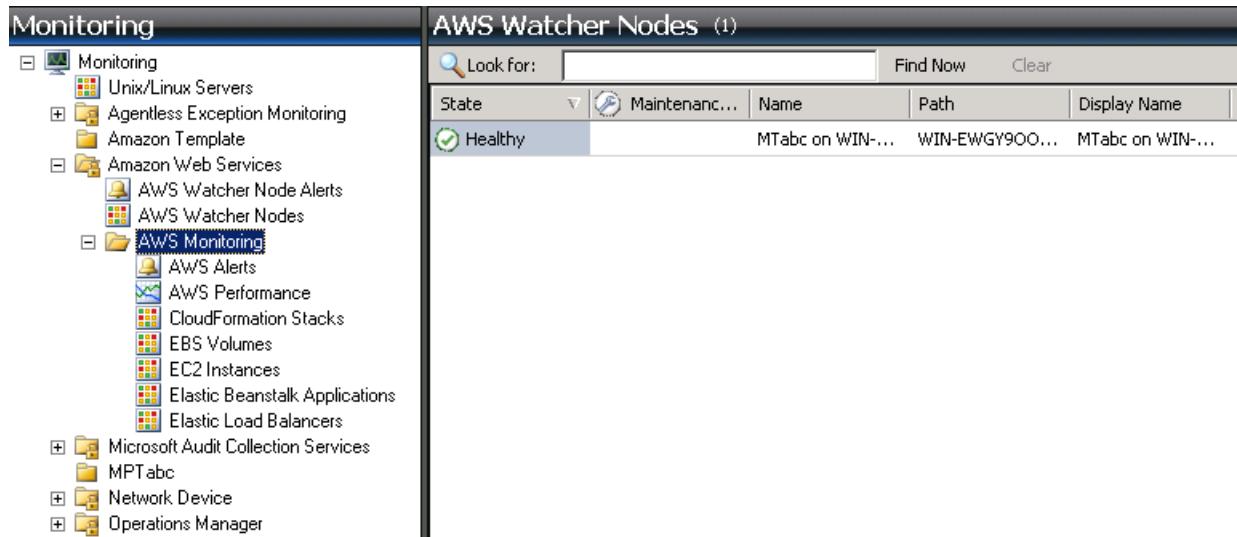
기본적으로 관리 팩 객체를 생성할 때 규칙 또는 모니터를 비활성화하거나 재정의를 생성합니다. 그러면 Operations Manager에서 기본 관리 팩에 설정을 저장합니다. 사용자 지정 설정을 기본 관리 팩에 저장하는 대신 사용자 지정할 봉인된 각 관리 팩에 대해 별도의 관리 팩을 생성해야 합니다.

6. Watcher Node Configuration(감시자 노드 구성) 페이지의 Watcher Node(감시자 노드) 목록에서 감시자 노드 역할을 할 에이전트 관리형 컴퓨터를 선택합니다.
7. Select AWS Run As account(AWS Run As 계정 선택) 드롭다운 목록에서 이전에 생성한 Run As 계정을 선택한 후 생성을 클릭합니다.
8. 구성된 AWS Management Pack에서 감시자 노드를 검색합니다. 감시자 노드가 검색되었는지 확인하려면 운영 콘솔에서 모니터링 작업 영역으로 이동합니다. 새 Amazon Web Services 폴더와 Amazon Watcher Nodes 하위 폴더가 표시됩니다. 이 하위 폴더에는 감시자 노드가 표시됩니다. AWS Management Pack에서 AWS에 대한 감시자 노드 연결을 자동으로 확인하여 모니터링합니다. 감시자 노드가 검색되면 이 목록에 표시됩니다. 감시자 노드가 준비되면 Healthy 상태로 변경됩니다.

Note

AWS Management Pack에서 AWS에 연결하려면 .NET용 AWS SDK, 모듈 및 스크립트를 감시자 노드에 배포해야 합니다. 이 작업은 10분 정도 걸릴 수 있습니다. 감시자 노드가 표시되지 않

거나 상태가 Not Monitored로 표시되는 경우 인터넷 연결과 IAM 권한을 확인하십시오. 자세한 내용은 [AWS Management Pack 문제 해결 \(p. 1281\)](#) 단원을 참조하십시오.



The screenshot shows the 'Monitoring' section of the Microsoft System Center Operations Manager interface. On the left, a tree view lists various monitoring categories: Monitoring, Unix/Linux Servers, Agentless Exception Monitoring, Amazon Template, Amazon Web Services (which is expanded to show AWS Watcher Node Alerts, AWS Watcher Nodes, AWS Monitoring, AWS Alerts, AWS Performance, CloudFormation Stacks, EBS Volumes, EC2 Instances, Elastic Beanstalk Applications, and Elastic Load Balancers), Microsoft Audit Collection Services, MPTabc, Network Device, and Operations Manager. On the right, a table titled 'AWS Watcher Nodes (1)' displays one node: MTabc on WIN-... with a 'State' of 'Healthy'. The table includes columns for State, Maintenance, Name, Path, and DisplayName.

9. 감시자 노드가 검색되면 종속 검색이 트리거되고 AWS 리소스가 운영 콘솔의 모니터링 작업 영역에 추가됩니다.

AWS 리소스 검색은 20분 이내에 완료됩니다. Operations Manager 환경, AWS 환경, 관리 서버의 부하, 감시자 노드의 부하 등에 따라 이 프로세스가 더 오래 걸릴 수 있습니다. 자세한 내용은 [AWS Management Pack 문제 해결 \(p. 1281\)](#) 단원을 참조하십시오.

5단계: 포트 및 엔드포인트 구성

AWS Management Pack for Microsoft System Center가 AWS 서비스와 통신할 수 있어야 이러한 서비스의 성능을 모니터링하고 System Center에서 경고를 제공할 수 있습니다. 모니터링에 성공하려면 다음 서비스에 대한 AWS 엔드포인트에 포트 80 및 443의 아웃바운드 HTTP 호출을 허용하도록 관리 팩 서버의 방화벽을 구성해야 합니다.

이렇게 하면 다음 AWS 서비스에 대한 모니터링을 수행할 수 있습니다.

- Amazon Elastic Compute Cloud(EC2)
- Elastic Load Balancing
- Amazon EC2 Auto Scaling
- AWS Elastic Beanstalk
- Amazon CloudWatch
- AWS CloudFormation

AWS 관리 팩은 .NET용 AWS SDK의 공인 API를 사용하여 포트 80 및 443을 통해 해당 서비스로부터 정보를 검색합니다. 각 서버에 로그인하고 포트 80 및 443에 대한 아웃바운드 방화벽 규칙을 활성화합니다.

방화벽 애플리케이션에서 더 세부적인 설정을 지원하는 경우 각 서비스에 특정 엔드포인트를 구성할 수 있습니다. 끝점은 웹 서비스의 진입점인 URL입니다. 예를 들어, ec2.us-west-2.amazonaws.com은 Amazon EC2 서비스의 진입점입니다. 방화벽에 엔드포인트를 구성하려면 실행하는 AWS 서비스에 대한 [특정 엔드포인트 URL](#)을 찾고, 방화벽 애플리케이션에서 해당 엔드포인트를 지정합니다.

AWS Management Pack 사용

AWS Management Pack를 사용하여 AWS 리소스의 상태를 모니터링할 수 있습니다.

내용

- [보기 \(p. 1260\)](#)
- [검색 \(p. 1274\)](#)
- [모니터 \(p. 1275\)](#)
- [규칙 \(p. 1276\)](#)
- [이벤트 \(p. 1276\)](#)
- [상태 모델 \(p. 1277\)](#)
- [AWS Management Pack 사용자 지정 \(p. 1279\)](#)

보기

AWS Management Pack에서는 운영 콘솔의 모니터링 작업 영역에 다음과 같은 보기가 표시됩니다.

보기

- [EC2 인스턴스 \(p. 1260\)](#)
- [Amazon EBS 볼륨 \(p. 1262\)](#)
- [Elastic Load Balancer \(p. 1264\)](#)
- [AWS Elastic Beanstalk 애플리케이션 \(p. 1266\)](#)
- [AWS CloudFormation 스택 \(p. 1268\)](#)
- [Amazon 성능 보기 \(p. 1270\)](#)
- [Amazon CloudWatch 측정치 경보 \(p. 1271\)](#)
- [AWS 경고 \(p. 1272\)](#)
- [감시자 노드\(System Center Operations Manager 2007 R2\) \(p. 1273\)](#)

EC2 인스턴스

모든 가용 영역과 리전의 특정 AWS 계정에 대한 EC2 인스턴스 상태를 표시합니다. 또한 이 보기에는 Virtual Private Cloud(VPC)에서 실행 중인 EC2 인스턴스가 포함됩니다. AWS Management Pack에서 태그를 검색 하므로 해당 태그를 사용하여 목록을 검색하고 필터링할 수 있습니다.

The screenshot shows the Microsoft System Center Operations Manager (SCOM) interface for monitoring EC2 Instances. The title bar reads "EC2 Instances - scom-2012 - Operations Manager". The left navigation pane under "Monitoring" includes sections for Active Alerts, Discovered Inventory, Distributed Applications, Task Status, UNIX/Linux Computers, Windows Computers, Agentless Exception Monitoring, Amazon Web Services, Personal AWS Account, and EC2 Instances. The main pane displays a list titled "EC2 Instances (103)" with columns for State, Maintenance, and Name. The list includes various instances such as "Win 2012 R2 m3", "ARP - Test", "Classic", "SCOM 2007 All", and "Default-Environment". A "Detail View" section for the selected instance "Default-Environment" provides detailed properties like Display Name, Full Path Name, Region, Configuration ID, Instance ID, Availability Zone, Image ID, Private DNS Name, Public DNS Name, Instance Type (t1.micro), and Security Groups. The status of the instance is shown as "Healthy".

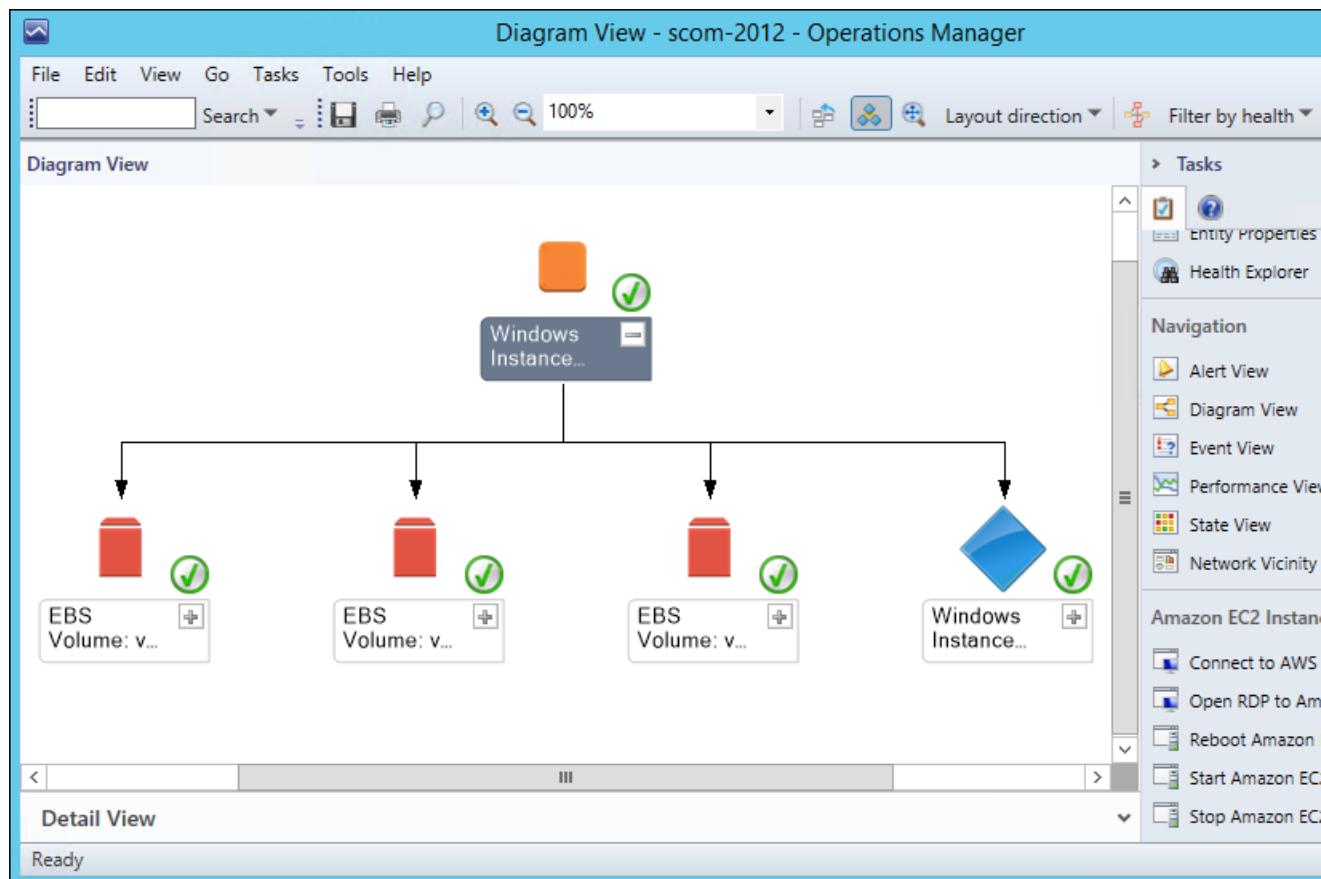
EC2 Instances (103)		
State	Maintain...	Name
Healthy		Win 2012 R2 m3
Healthy		ARP - Test
Healthy		Classic
Healthy		SCOM 2007 All
Critical		Default-Environment
Healthy		SCOM 2012 Environment - DC
Healthy		prios
Healthy		SCVMM - All in one
Healthy		applicationTwo-env
Healthy		SCOM 2012 Environment - MS 1
Healthy		SCOM 2007 DC
Healthy		SCOM 2012 Environment - SQL
Healthy		metricgathertest

EC2 인스턴스를 선택한 경우 인스턴스 상태 작업을 수행할 수 있습니다:

- Amazon Console 열기: 웹 브라우저에서 AWS Management 콘솔을 시작합니다.
- Amazon EC2 인스턴스에 대한 RDP 열기: 선택된 Windows 인스턴스에 대한 RDP 연결을 엽니다.
- Amazon EC2 인스턴스 재부팅: 선택한 EC2 인스턴스를 재부팅합니다.
- Amazon EC2 인스턴스 시작: 선택한 EC2 인스턴스를 시작합니다.
- Amazon EC2 인스턴스 중지: 선택한 EC2 인스턴스를 중지합니다.

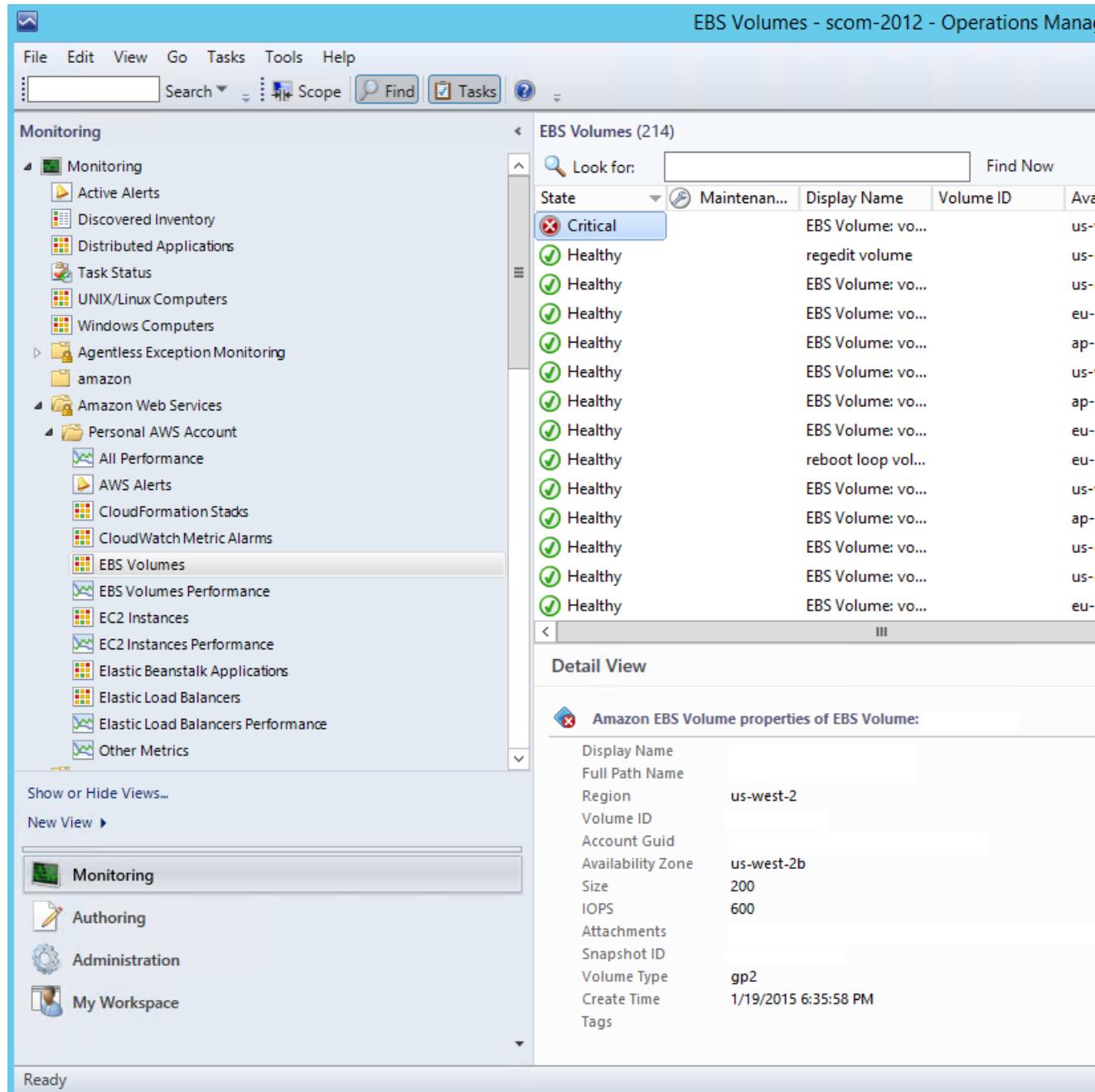
EC2 인스턴스 다이어그램 보기

다른 구성 요소와 인스턴스의 관계를 표시합니다.



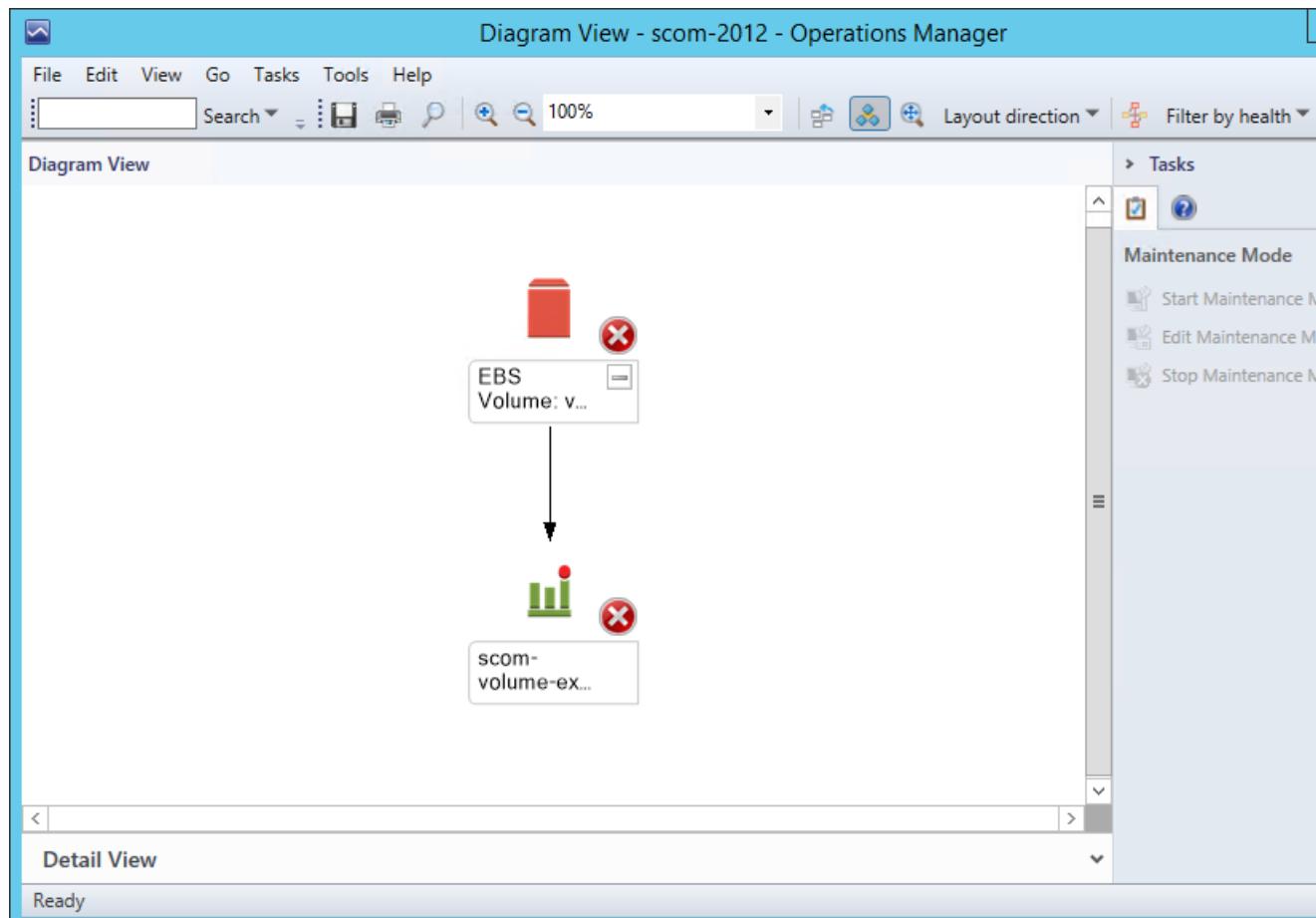
Amazon EBS 볼륨

모든 가용 영역과 리전의 특정 AWS 계정에 대한 모든 Amazon EBS 볼륨 상태를 표시합니다.



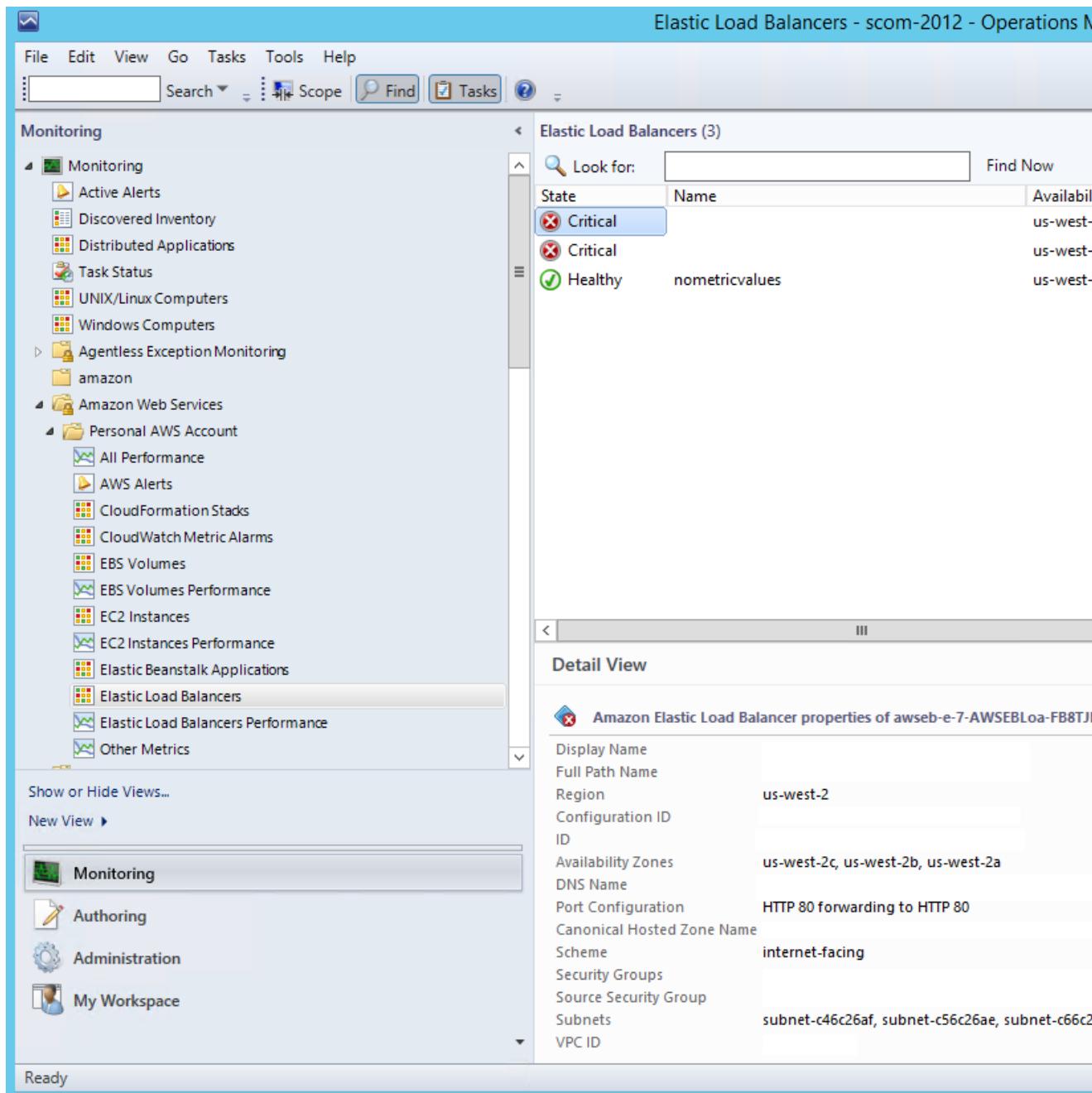
Amazon EBS 볼륨 다이어그램 보기

Amazon EBS 볼륨 및 연결된 정보를 표시합니다. 다음 그림에 예가 나와 있습니다.



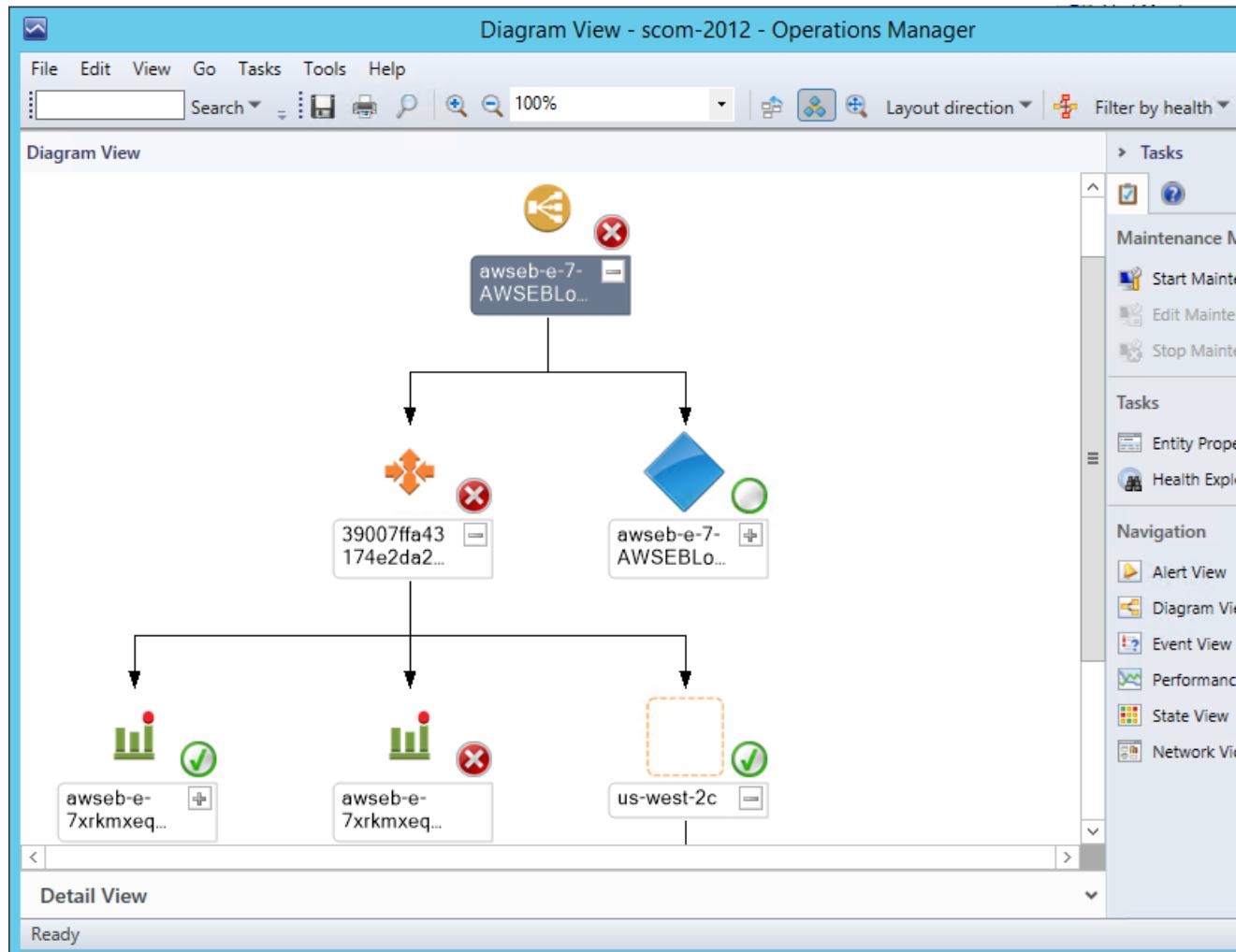
Elastic Load Balancer

모든 리전의 특정 AWS 계정에 대한 모든 로드 밸런서의 상태를 표시합니다.



Elastic Load Balancing 다이어그램 보기

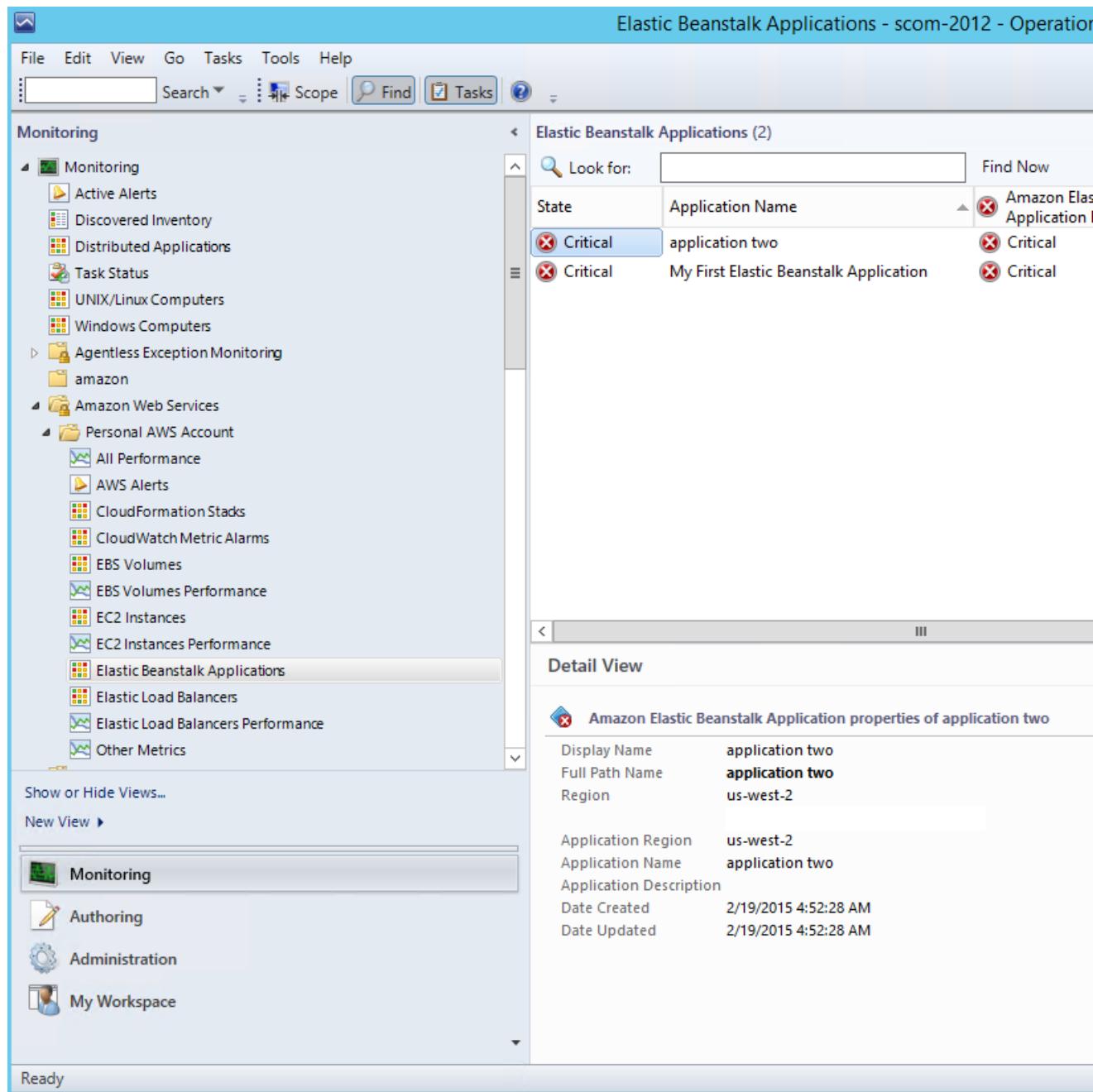
다른 구성 요소와 Elastic Load Balancing의 관계를 표시합니다. 다음 그림에 예가 나와 있습니다.



AWS Elastic Beanstalk 애플리케이션

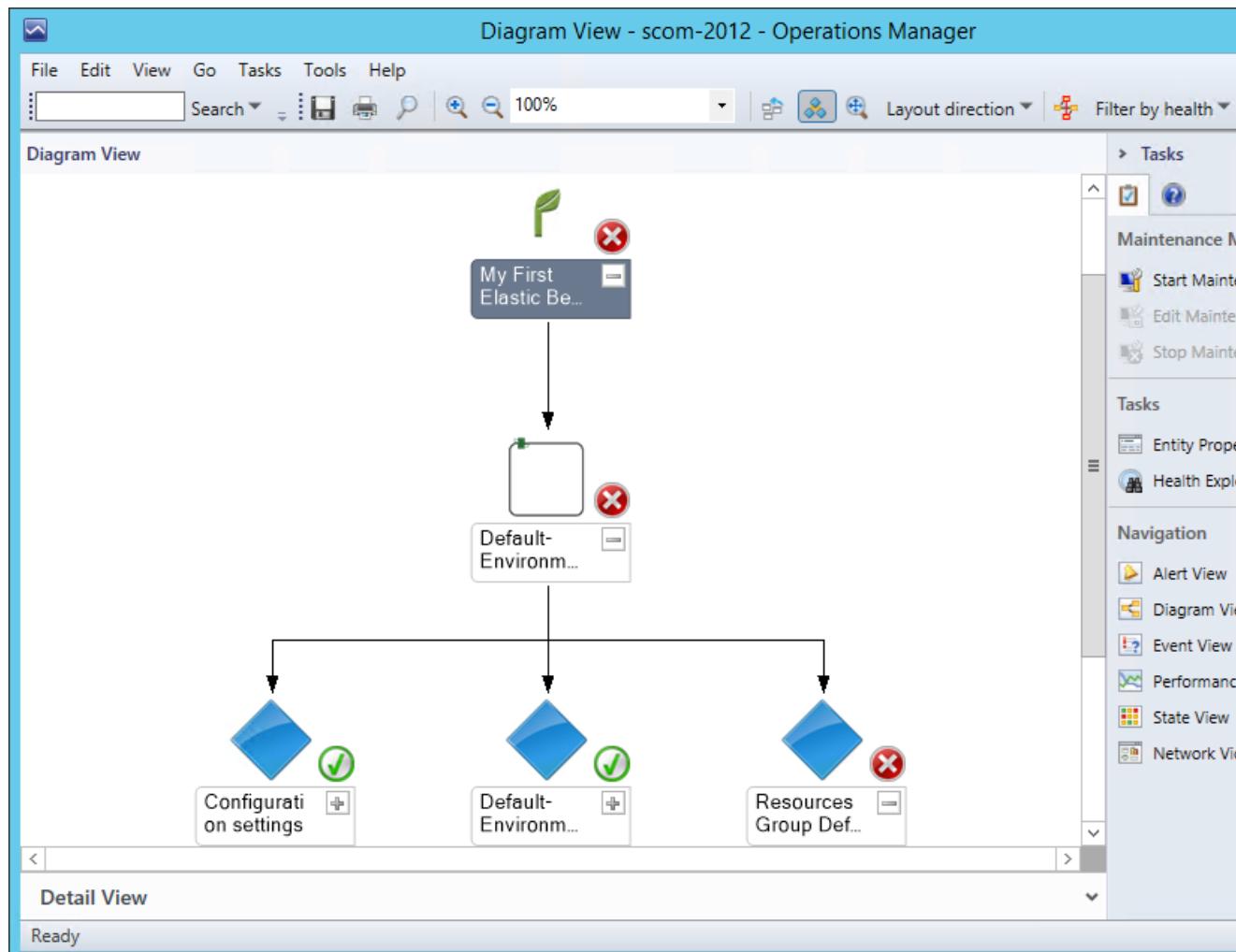
검색된 모든 AWS Elastic Beanstalk 애플리케이션의 상태를 표시합니다.

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
보기



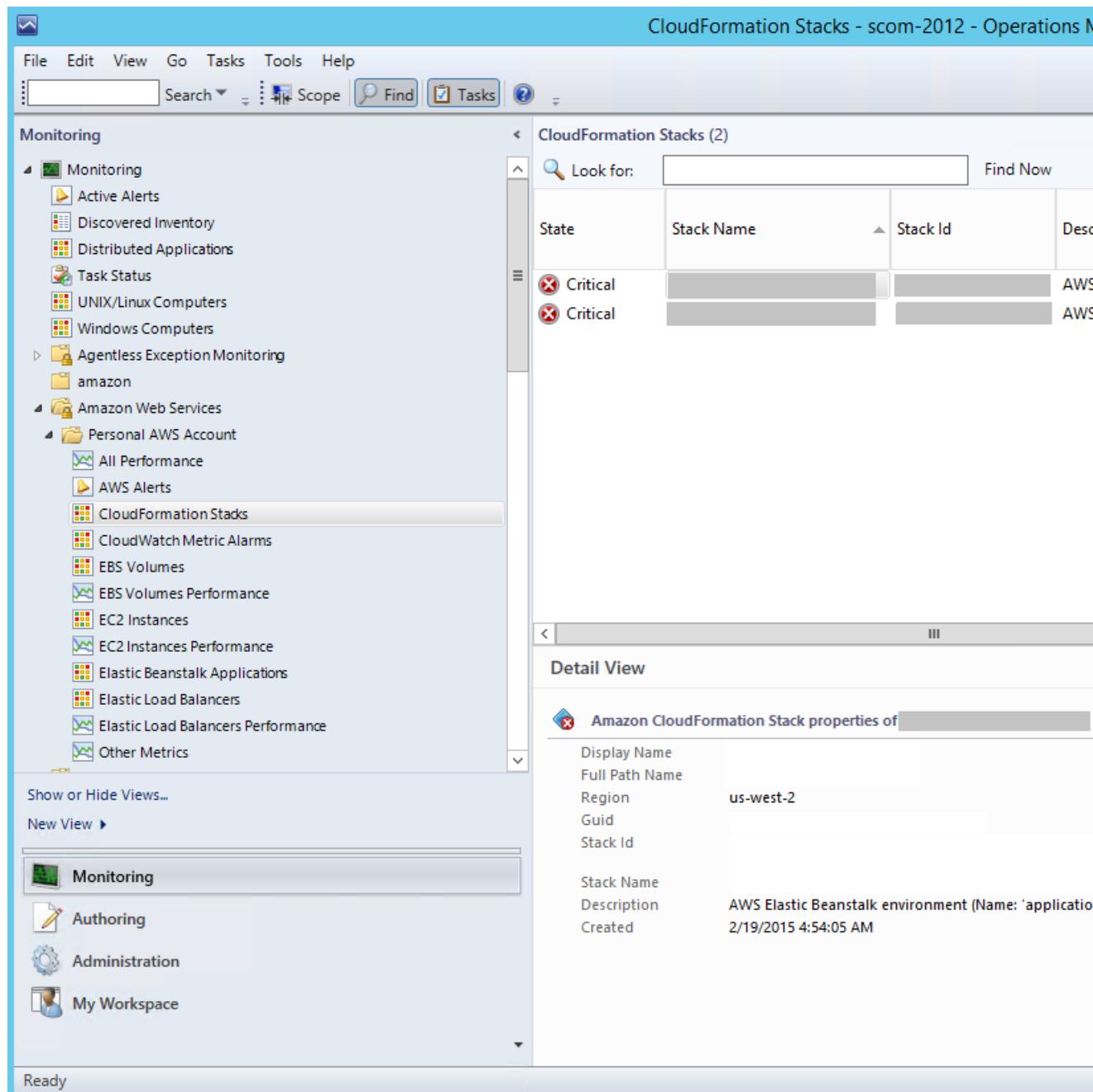
AWS Elastic Beanstalk 애플리케이션 다이어그램 보기

AWS Elastic Beanstalk 애플리케이션, 애플리케이션 환경, 애플리케이션 구성 및 애플리케이션 리소스 객체를 표시합니다.



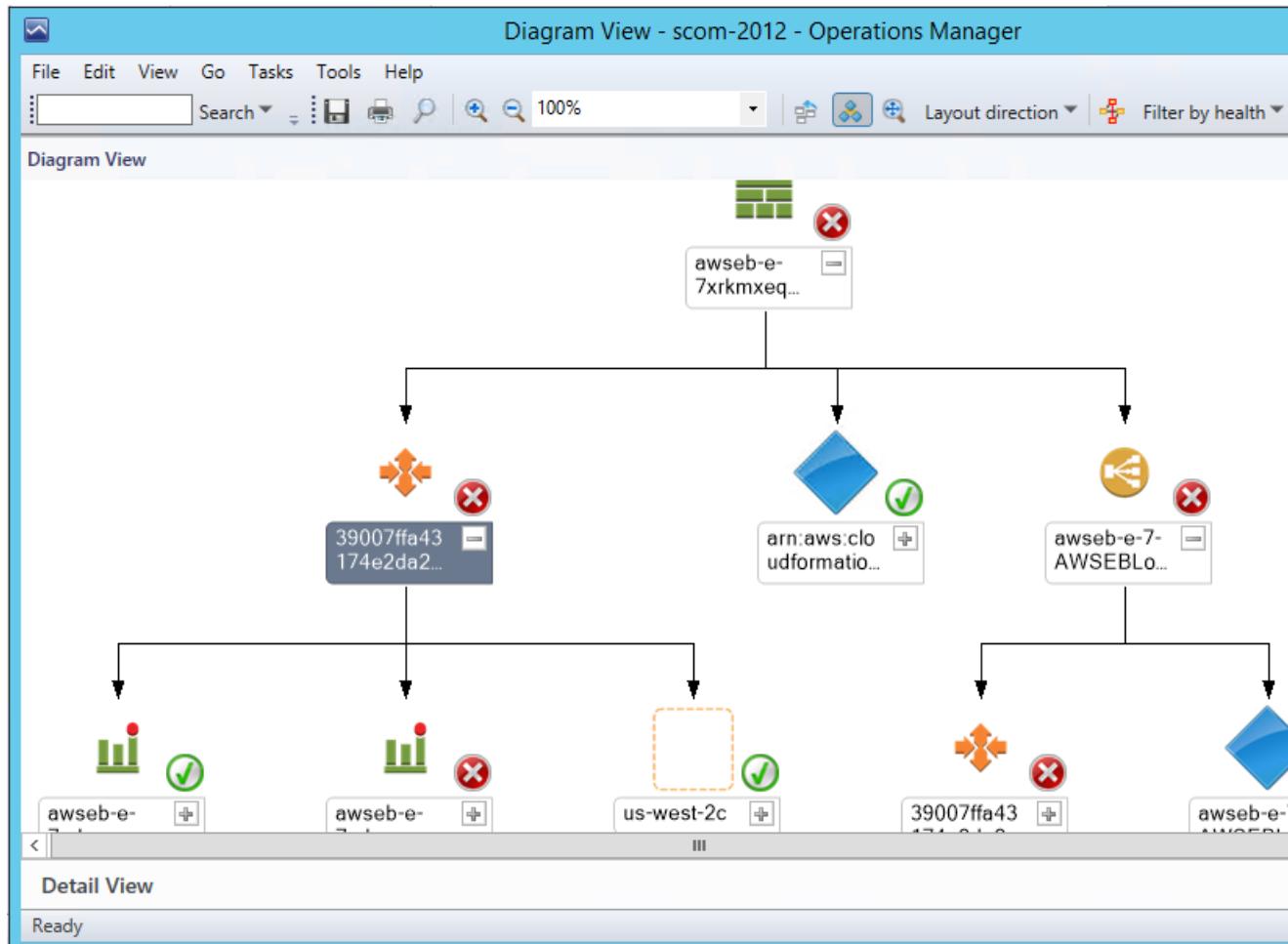
AWS CloudFormation 스택

모든 리전의 특정 AWS 계정에 대한 모든 AWS CloudFormation 스택의 상태를 표시합니다.



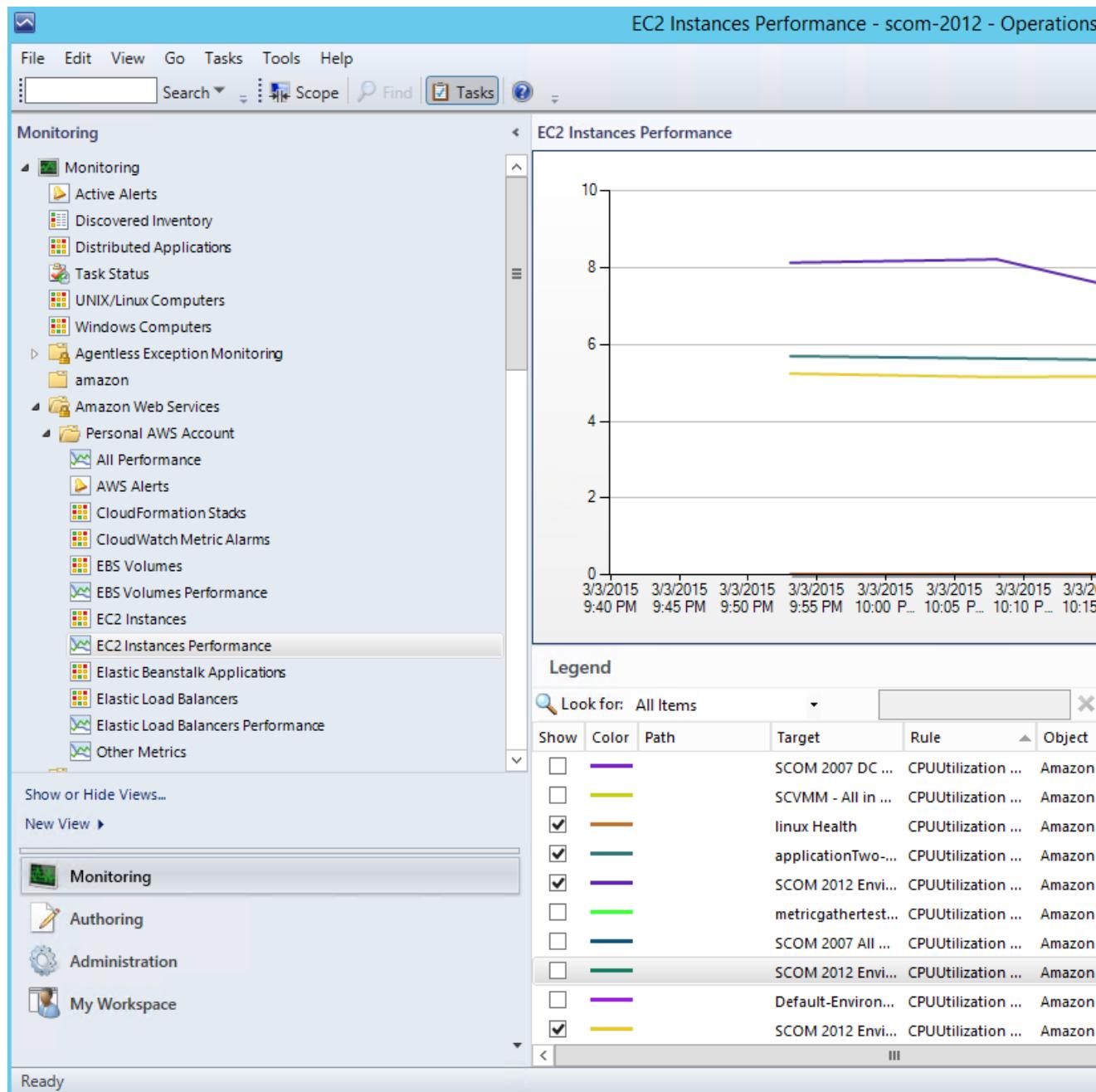
AWS CloudFormation 스택 다이어그램 보기

다른 구성 요소와 AWS CloudFormation 스택의 관계를 표시합니다. AWS CloudFormation 스택은 Amazon EC2 또는 Elastic Load Balancing 리소스를 포함할 수 있습니다. 다음 그림에 예가 나와 있습니다.



Amazon 성능 보기

Amazon EC2, Amazon EBS 및 Elastic Load Balancing에 대한 Amazon CloudWatch 측정치, 사용자 지정 측정치 및 CloudWatch 경보에서 생성된 측정치를 표시합니다. 또한 각 리소스에 대한 별도의 성능 보기와 있습니다. 기타 측정치 성능 보기에 사용자 지정 측정치 및 CloudWatch 경보에서 생성된 측정치가 포함됩니다. 이러한 지표에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서의 [CloudWatch 지표를 게시하는 AWS 서비스](#)를 참조하세요. 다음 그림에 예가 나와 있습니다.



Amazon CloudWatch 측정치 경보

검색된 AWS 리소스와 관련된 Amazon CloudWatch 경보를 표시합니다.

CloudWatch Metric Alarms - scom-2012 - Operations

The screenshot shows the SCOM interface with the title bar "CloudWatch Metric Alarms - scom-2012 - Operations". The menu bar includes File, Edit, View, Go, Tasks, Tools, and Help. Below the menu is a toolbar with Search, Scope, Find, Tasks, and Help buttons.

The left pane displays a navigation tree under "Monitoring":

- Monitoring
 - Active Alerts
 - Discovered Inventory
 - Distributed Applications
 - Task Status
 - UNIX/Linux Computers
 - Windows Computers
- Agentless Exception Monitoring
- amazon
- Amazon Web Services
 - Personal AWS Account
 - All Performance
 - AWS Alerts
 - CloudFormation Stacks
 - CloudWatch Metric Alarms
 - EBS Volumes
 - EBS Volumes Performance
 - EC2 Instances
 - EC2 Instances Performance
 - Elastic Beanstalk Applications
 - Elastic Load Balancers
 - Elastic Load Balancers Performance
 - Other Metrics

- Show or Hide Views...
- New View ▾

The right pane shows a list titled "CloudWatch Metric Alarms (11)" with columns for State, Alarm Name, and Metric Name. The "State" column uses icons: red for Critical (4 entries), green for Healthy (7 entries). The "Alarm Name" and "Metric Name" columns list specific alarms and their corresponding metrics.

State	Alarm Name	Metric Name
Critical	dynamo test alarm	ProvisionedWriteCapacityUnits
Critical	scom-volume-exists-test	VolumeReadBytes
Critical	awseb-e-qazu95f2zm-stack-A...	NetworkOut
Critical	elb alarm	HealthyHostCount
Critical	awseb-e-7xrkmxeqvy-stack-A...	NetworkOut
Healthy	awseb-e-qazu95f2zm-stack-A...	NetworkOut
Healthy	awseb-e-7xrkmxeqvy-stack-A...	NetworkOut
Healthy	testalarm	VolumeReadBytes
Healthy	az_alarm	Latency
Healthy	awsec2-i-cc4811c4-High-CPU...	CPUUtilization
Healthy	scom-bug-alarm	CPUUtilization

Below the list is a "Detail View" section for the "dynamo test alarm" entry:

Display Name	dynamo test alarm, Metric Name: ProvisionedWriteCapacityUnits
Full Path Name	dynamo test alarm, Metric Name: ProvisionedWriteCapacityUnits
Region	us-west-2
Alarm ID	arn:aws:cloudwatch:us-west-2:946130359068:alarm:dynamo-test-alarm-39007ffa43174e2da200cb945151a2bd
Description	should always alarm
Condition	ProvisionedWriteCapacityUnits <= 1000
Alarm Name	dynamo test alarm
Metric Name	ProvisionedWriteCapacityUnits
Namespace	AWS/DynamoDB
Threshold	1000
Unit	

AWS 경고

객체가 중요한 상태인 경우 AWS 관리 팩이 생성하는 경고를 표시합니다.

The screenshot shows the System Center Operations Manager 2007 R2 interface with the title bar "AWS Alerts - scom-2012 - Operations Manager". The left navigation pane is titled "Monitoring" and includes sections for Active Alerts, Discovered Inventory, Distributed Applications, Task Status, UNIX/Linux Computers, Windows Computers, Agentless Exception Monitoring, amazon, and Amazon Web Services. Under "Personal AWS Account", "AWS Alerts" is selected. The main workspace displays "AWS Alerts (5)" with a list of critical alerts:

Icon	Source	Name
Red X	dynamo test al...	Amazon CloudWatch Metric Alert
Red X	scom-volume-...	Amazon CloudWatch Metric Alert
Red X	awseb-e-qazu9...	Amazon CloudWatch Metric Alert
Red X	awseb-e-7xrkm...	Amazon CloudWatch Metric Alert
Red X	elb alarm, Metr...	Amazon CloudWatch Metric Alert

A detailed view of the first alert, "Amazon CloudWatch Metric Alert", is shown in a modal window. The "Key Details" section includes:

- Alert source: dynamo test alarm, Metric Name: ProvisionedThroughputExceeded
- Severity: Critical
- Priority: Medium
- Age: 2 Hours, 37 Minutes

The "Alert Description" section states: "The metric alarm dynamo test alarm, Metric Name: ProvisionedThroughputExceeded has entered the critical state. Threshold Crossed: 1 datapoint (5.0) was greater than or equal to the threshold (5.0) for 1 second(s)."

감시자 노드(System Center Operations Manager 2007 R2)

모니터링 중인 모든 AWS 계정의 감시자 노드 상태를 보여 줍니다. 정상 상태는 감시자 노드가 올바르게 구성되어 있고 AWS와 통신할 수 있음을 의미합니다.



검색

검색은 AWS Management Pack에서 모니터링하는 AWS 리소스입니다. AWS Management Pack에서는 다음과 같은 객체를 검색합니다.

- Amazon EC2 인스턴스
- EBS 볼륨
- ELB 로드 밸런서
- AWS CloudFormation 스택
- Amazon CloudWatch 경보
- AWS Elastic Beanstalk 애플리케이션
- Amazon EC2 Auto Scaling 그룹 및 가용 영역

다음 리소스의 경우 Amazon CloudWatch 측정치가 생성됩니다.

- Amazon EC2 인스턴스
- EBS 볼륨
- Elastic Load Balancing
- 사용자 지정 Amazon CloudWatch 측정치
- 기존 Amazon CloudWatch 경보의 측정치

Amazon CloudWatch 지표 검색의 경우 다음과 같은 지침이 적용됩니다.

- AWS CloudFormation 스택은 기본 Amazon CloudWatch 지표가 없습니다.
- 중지된 Amazon EC2 인스턴스나 사용되지 않은 Amazon EBS 볼륨은 기본 Amazon CloudWatch 지표에 대한 데이터를 생성하지 않습니다.
- Amazon EC2 인스턴스를 시작한 후 Amazon CloudWatch 지표가 Operations Manager에 표시되는 데 최대 30분까지 걸릴 수 있습니다.

- AWS 리소스가 종료되더라도 Amazon CloudWatch에서는 모니터링 데이터를 2주간 보관합니다. 이 데이터는 Operations Manager에 표시됩니다.
- 지원되지 않는 리소스에 대한 기존 Amazon CloudWatch 경보가 측정치를 생성하고 Amazon CloudWatch 경보와 연결됩니다. 이러한 측정치는 기타 측정치 성능 보기에서 확인할 수 있습니다.

또한 AWS Management Pack에서는 다음과 같은 관계를 검색합니다.

- AWS CloudFormation 스택과 해당 Elastic Load Balancing 또는 Amazon EC2 리소스
- Elastic Load Balancing 로드 밸런서와 해당 EC2 인스턴스
- Amazon EC2 인스턴스 및 해당 EBS 볼륨
- Amazon EC2 인스턴스 및 해당 운영 체제
- AWS Elastic Beanstalk 애플리케이션과 해당 환경, 구성 및 리소스

AWS Management Pack에서는 EC2 인스턴스와 해당 인스턴스에서 실행 중인 운영 체제를 자동으로 검색합니다. 이 관계를 검색하려면 인스턴스에 Operations Manager Agent를 설치하여 구성하고 해당 운영 체제 관리 팩을 Operations Manager에 가져와야 합니다.

검색은 리소스 풀(System Center 2012) 또는 감시자 노드(System Center 2007 R2)의 관리 서버에서 실행됩니다.

검색	간격(초)
Amazon 리소스 검색(SCOM 2012) EC2 인스턴스, Amazon EBS 볼륨, 로드 밸런서 및 CloudFront 스택을 검색합니다.	14400
AWS Elastic Beanstalk 검색 AWS Elastic Beanstalk를 검색하고 환경, 리소스 및 구성과의 관계를 검색합니다.	14400
CloudWatch 경보 검색 CloudWatch 지표를 사용하여 생성된 경보를 검색합니다.	900
사용자 지정 CloudWatch 측정치 검색 사용자 지정 CloudWatch 측정치를 검색합니다.	14400
감시자 노드 검색(SCOM 2007 R2) 루트 관리 서버를 대상으로 하며 감시자 노드 객체를 생성합니다.	14400

모니터

모니터는 AWS 리소스의 상태를 측정하는 데 사용됩니다. 모니터는 리소스 풀(System Center 2012) 또는 감시자 노드(System Center 2007 R2)의 관리 서버에서 실행됩니다.

모니터링	간격(초)
AWS CloudFormation 스택 상태	900
Amazon CloudWatch 지표 경보	300

모니터링	간격(초)
Amazon EBS 볼륨 상태	900
Amazon EC2 인스턴스 상태	900
Amazon EC2 인스턴스 시스템 상태	900
AWS Elastic Beanstalk 상태	900
감시자 노드와 Amazon 클라우드 연결(SCOM 2007 R2)	900

규칙

규칙은 Amazon CloudWatch 지표를 기반으로 알림을 생성하고 분석 및 보고를 위한 데이터를 수집합니다.

규칙	간격(초)
AWS 리소스 검색 규칙(SCOM 2007 R2)	14400
감시자 노드를 대상으로 하며 AWS API를 사용하여 EC2 인스턴스, EBS 볼륨, 로드 밸런서, AWS CloudFormation 스택 등의 AWS 리소스에 대한 객체를 검색합니다(CloudWatch 지표나 경보는 검색하지 않음). 검색이 완료되면 Not Monitored(모니터링되지 않음) 상태의 객체를 확인합니다.	
Amazon Elastic Block Store 볼륨 성능 지표 데이터 수집 규칙	900
Amazon EC2 인스턴스 성능 지표 데이터 수집 규칙	900
Elastic Load Balancing 밸런싱 성능 지표 데이터 수집 규칙	900
사용자 지정 CloudWatch 측정치 데이터 수집 규칙	900

이벤트

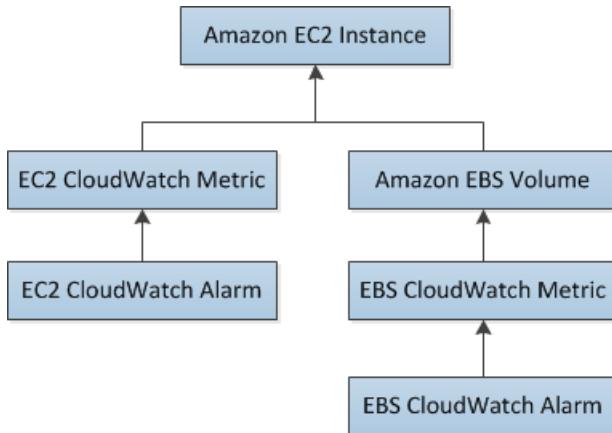
이벤트는 모니터링된 리소스와 관련된 활동에 대해 보고하며 Operations Manager 이벤트 로그에 기록됩니다.

이벤트 ID	설명
4101	Amazon EC2 인스턴스 검색(일반 검색) 완료
4102	Elastic Load Balancing 지표 검색 Amazon EBS 볼륨 지표 검색 Amazon EC2 인스턴스 지표 검색 완료
4103	Amazon CloudWatch 지표 경보 검색 완료
4104	Amazon Windows 컴퓨터 검색 완료
4105	Amazon 지표 경보 수집 완료
4106	EC2 인스턴스 컴퓨터 관계 검색 완료

이벤트 ID	설명
4107	AWS CloudFormation 스택 상태 수집 완료
4108	감시자 노드 가용성 상태 수집 완료
4109	Amazon 지표 수집 규칙 완료
4110	Amazon 인스턴스 상태 변경 작업 완료
4111	EC2 인스턴스 상태 모니터 상태 완료
4112	Amazon EBS 볼륨 상태 모니터 상태 완료
4113	Amazon EC2 인스턴스 예약된 이벤트 모니터 상태 계산됨
4114	Amazon EBS 예약된 이벤트 모니터 상태 계산됨
4115	Elastic Beanstalk 검색 완료
4116	Elastic Beanstalk 환경 상태 상태 계산됨
4117	Elastic Beanstalk 환경 작업 상태 계산됨
4118	Elastic Beanstalk 환경 구성 상태 계산됨

상태 모델

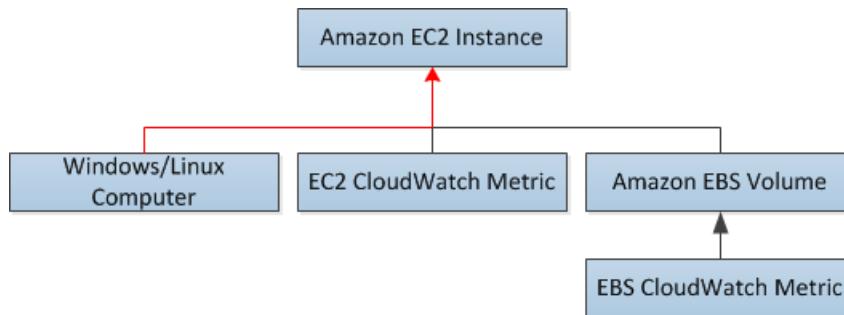
다음은 AWS Management Pack에 의해 정의된 상태 모델을 보여 주는 그림입니다.



CloudWatch 경보에 대한 상태는 해당 CloudWatch 지표로 룰업됩니다. Amazon EC2의 CloudWatch 지표에 대한 상태는 EC2 인스턴스로 룰업됩니다. 마찬가지로 Amazon EBS의 CloudWatch 지표에 대한 상태는 Amazon EBS 볼륨으로 룰업됩니다. EC2 인스턴스에 사용되는 Amazon EBS 볼륨에 대한 상태는 EC2 인스턴스로 룰업됩니다.

EC2 인스턴스와 운영 체제 사이의 관계가 검색된 경우 운영 체제 상태가 EC2 인스턴스로 룰업됩니다.

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
상태 모델

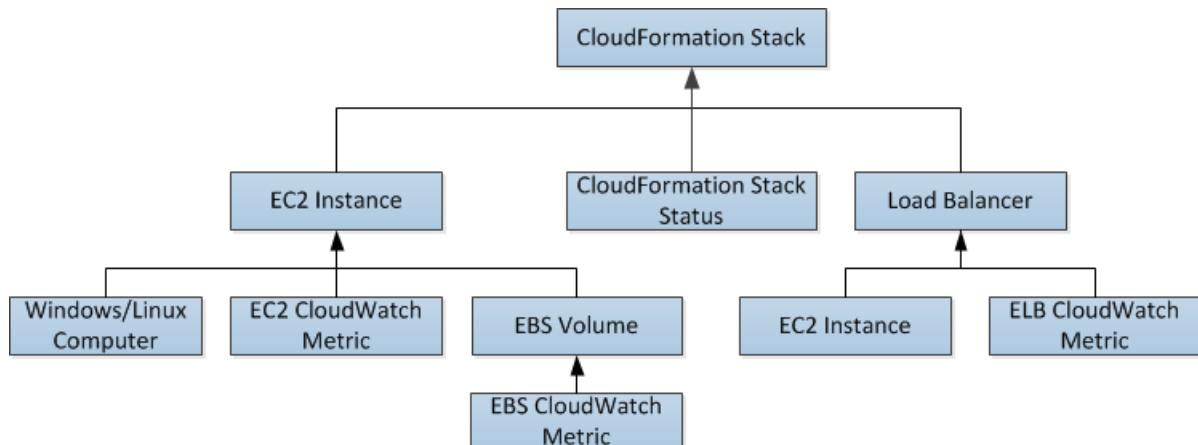


AWS CloudFormation 스택의 상태는 AWS CloudFormation 스택 자체의 상태와 해당 리소스(로드 밸런서 및 EC2 인스턴스)의 상태에 따라 다릅니다.

다음 표에서는 AWS CloudFormation 스택의 상태가 해당 상태와 어떻게 대응하는지를 보여 줍니다.

상태	AWS CloudFormation 스택 상태	참고
Error	CREATE_FAILED DELETE_IN_PROGRESS DELETE_FAILED UPDATE_ROLLBACK_FAILED	사용 가능성이 높음
경고	UPDATE_ROLLBACK_IN_PROGRESS UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS UPDATE_ROLLBACK_COMPLETE	문제 발생 후 복구 중
Healthy	CREATE_COMPLETE UPDATE_IN_PROGRESS UPDATE_COMPLETE_CLEANUP_IN_PROGRESS UPDATE_COMPLETE	사용 가능

AWS CloudFormation 스택에 대한 전체 상태 모델은 다음과 같습니다.



AWS Management Pack 사용자 지정

검색, 규칙 및 모니터의 주기를 변경하기 위해 시간 간격(초)을 재정의할 수 있습니다.

주기를 변경하려면

1. Operations Manager 도구 모음에서 이동을 클릭한 후 Authoring(저작)을 클릭합니다.
2. Authoring(저작) 창에서 Management Pack Objects(관리 팩 객체)를 확장한 후 객체를 클릭하여 변경합니다(예: Object Discoveries(객체 검색), 규칙 또는 모니터).
3. 도구 모음에서 범위를 클릭합니다.
4. Scope Management Pack Objects(범위 관리 팩 객체) 대화 상자에서 View all targets(모든 대상 보기)를 클릭합니다.
5. 범위를 Amazon 객체로 제한하려면 Look for(검색 대상) 필드에 Amazon을 입력합니다.
6. 구성할 객체를 선택하고 확인을 클릭합니다.
7. Operations Manager 가운데 창에서 구성할 객체를 마우스 오른쪽 버튼으로 클릭하고 재정의를 클릭한 후 구성할 재정의 유형을 클릭합니다.
8. Override Properties(속성 재정의) 대화 상자를 사용하여 객체에 대해 다른 값과 설정을 구성합니다.

Tip

검색, 규칙 또는 모니터링 객체를 비활성화하려면 Operations Manager 가운데 창에서 비활성화 할 객체를 마우스 오른쪽 버튼으로 클릭하고 재정의를 클릭한 후 Disable the Rule(규칙 비활성화)을 클릭합니다. 예를 들어, AWS Elastic Beanstalk 애플리케이션을 실행하지 않거나 사용자 지정 Amazon CloudWatch 측정치를 사용하는 경우 규칙을 비활성화할 수 있습니다.

재정의 생성에 대한 자세한 내용은 Microsoft TechNet 웹 사이트의 [대상 지정 및 재정의를 사용하여 모니터링 조정](#)을 참조하십시오.

사용자 지정 규칙 및 모니터 생성에 대한 자세한 내용은 Microsoft TechNet 웹 사이트의 [Authoring for System Center 2012 - Operations Manager](#) 또는 [System Center Operations Manager 2007 R2 Management Pack Authoring Guide](#)를 참조하십시오.

AWS Management Pack 업그레이드

AWS Management Pack를 업데이트하는 데 사용하는 절차는 System Center의 버전에 따라 다릅니다.

System Center 2012

AWS Management Pack을 업그레이드하려면

1. [AWS Add-Ins for Microsoft System Center](#) 웹 사이트에서 SCOM 2012를 클릭합니다. AWS-SCOM-MP-2.0-2.5.zip을 컴퓨터에 저장하고 압축을 풉니다. .zip 파일에는 Amazon.AmazonWebServices.mpb 포함되어 있습니다.
2. Operations 콘솔의 이동 메뉴에서 관리를 클릭한 후 Management Packs(관리 팩)을 클릭합니다.
3. 작업 창에서 Import Management Packs(관리 팩 가져오기)를 클릭합니다.
4. Select Management Packs(관리 팩 선택) 페이지에서 추가를 클릭한 후 Add from disk(디스크에서 추가)를 클릭합니다.
5. Select Management Packs to import(가져올 관리 팩 선택) 대화 상자에서 Amazon.AmazonWebServices.mpb 파일을 다운로드한 위치에서 이 파일을 선택한 후 열기를 클릭합니다.

6. Select Management Packs(관리 팩 선택) 페이지의 가져오기 목록에서 Amazon Web Services 관리 팩을 선택한 후 설치를 클릭합니다.

설치 버튼이 비활성화된 경우에는 현재 버전으로 업그레이드가 지원되지 않는 것이므로 현재 버전을 설치하려면 AWS Management Pack를 제거해야 합니다. 자세한 내용은 [AWS Management Pack 제거 \(p. 1280\)](#) 단원을 참조하십시오.

System Center 2007 R2

AWS Management Pack을 업그레이드하려면

1. Management Server에서 [AWS Add-Ins for Microsoft System Center](#) 웹 사이트로 이동하고 SCOM 2007을 클릭합니다. AWS-MP-Setup-2.5.msi를 저장하고 실행합니다.
2. 다음을 클릭하고 지침에 따라 이전에 설치한 구성 요소를 업그레이드합니다.
3. 루트 관리 서버, 운영 콘솔 및 감시자 노드가 다른 컴퓨터에 있는 경우 각 컴퓨터에서 설치 프로그램을 다운로드하여 실행해야 합니다.
4. 감시자 노드에서 관리자로 명령 프롬프트 창을 열고 다음 명령을 실행합니다.

```
C:\> net stop HealthService
The System Center Management service is stopping.
The System Center Management service was stopped successfully.

C:\> net start HealthService
The System Center Management service is starting.
The System Center Management service was started successfully.
```

5. Operations 콘솔의 이동 메뉴에서 관리를 클릭한 후 Management Packs(관리 팩)을 클릭합니다.
6. 작업 창에서 Import Management Packs(관리 팩 가져오기)를 클릭합니다.
7. Select Management Packs(관리 팩 선택) 페이지에서 추가를 클릭한 후 Add from disk(디스크에서 추가)를 클릭합니다.
8. Select Management Packs to import(가져올 관리 팩 선택) 대화 상자에서 디렉터리를 c:\Program Files (x86)\Amazon Web Services Management Pack으로 변경하고 Amazon.AmazonWebServices.mp 파일을 선택한 후 열기를 클릭합니다.
9. Select Management Packs(관리 팩 선택) 페이지의 가져오기 목록에서 Amazon Web Services 관리 팩을 선택한 후 설치를 클릭합니다.

설치 버튼이 비활성화된 경우에는 현재 버전으로 업그레이드가 지원되지 않는 것으로 먼저 AWS Management Pack을 제거해야 합니다. 자세한 내용은 [AWS Management Pack 제거 \(p. 1280\)](#) 단원을 참조하십시오.

AWS Management Pack 제거

AWS Management Pack을 제거해야 하는 경우 다음 설치를 사용합니다.

System Center 2012

AWS Management Pack을 제거하려면

1. Operations 콘솔의 이동 메뉴에서 관리를 클릭한 후 Management Packs(관리 팩)을 클릭합니다.
2. Amazon Web Services를 마우스 오른쪽 버튼으로 클릭하고 삭제를 선택합니다.
3. Dependent Management Packs(종속 관리 팩) 대화 상자에서 종속 관리 팩을 확인한 후 닫기를 클릭합니다.

4. 종속 관리 팩을 마우스 오른쪽 버튼으로 클릭하고 삭제를 선택합니다.
5. Amazon Web Services를 마우스 오른쪽 버튼으로 클릭하고 삭제를 선택합니다.

System Center 2007 R2

AWS Management Pack를 제거하려면

1. 이전 섹션에서 System Center 2012에 대해 설명한 1-5단계를 수행합니다.
2. 제어판에서 [프로그램 및 기능]을 엽니다. Amazon Web Services Management Pack(Amazon Web Services 관리 팩)을 선택하고 제거를 클릭합니다.
3. 루트 관리 서버, 운영 콘솔 및 감시자 노드가 다른 컴퓨터에 있는 경우 각 컴퓨터에서 이 절차를 반복해야 합니다.

AWS Management Pack 문제 해결

다음은 일반적인 오류, 이벤트 및 문제 해결 단계입니다.

내용

- [오류 4101 및 4105 \(p. 1281\)](#)
- [오류 4513 \(p. 1281\)](#)
- [이벤트 623 \(p. 1282\)](#)
- [이벤트 2023 및 2120 \(p. 1282\)](#)
- [이벤트 6024 \(p. 1282\)](#)
- [System Center 2012 — Operations Manager에 대한 일반 문제 해결 \(p. 1282\)](#)
- [System Center 2007 R2에 대한 일반 문제 해결 \(p. 1283\)](#)

오류 4101 및 4105

다음 오류 중 하나가 발생하는 경우 AWS Management Pack를 업그레이드해야 합니다. 자세한 내용은 [AWS Management Pack 업그레이드 \(p. 1279\)](#) 단원을 참조하십시오.

```
Error 4101
Exception calling "DescribeVolumes" with "1" argument(s): "AWS was not able to validate
the
provided access credentials"
```

```
Error 4105
Exception calling "DescribeApplications" with "0" argument(s): "The security token
included
in the request is invalid"
```

오류 4513

다음 오류 중 하나가 발생하는 경우 AWS Management Pack를 업그레이드해야 합니다. 자세한 내용은 [AWS Management Pack 업그레이드 \(p. 1279\)](#) 단원을 참조하십시오.

```
Error 4513
The callback method DeliverDataToModule failed with exception "Resolution of the
dependency
```

```
failed, type = "Amazon.SCOM.SDK.Interfaces.IMonitorSdk", name = "(none)".  
Exception occurred while: Calling constructor Amazon.SCOM.SDK.CloudWatch.AwsMonitorSdk  
(System.String awsAccessKey, System.String awsSecretKey).  
Exception is: InvalidOperationException - Collection was modified; enumeration operation  
may not execute.
```

이벤트 623

Windows 이벤트 로그에 다음 이벤트가 있는 경우 [KB975057](#)에 설명된 솔루션을 따르십시오.

```
Event ID: 623  
HealthService (process_id) The version store for instance instance ("name") has reached  
its maximum size of size MB. It is likely that a long-running transaction is preventing  
cleanup of the version store and causing it to build up in size. Updates will be rejected  
until the long-running transaction has been completely committed or rolled back.  
Possible long-running transaction:  
SessionId: id  
Session-context: value  
Session-context ThreadId: id  
Cleanup: value
```

이벤트 2023 및 2120

Windows 이벤트 로그에 다음 이벤트가 있는 경우 자세한 내용은 [Event ID 2023 and 2120](#)을 참조하십시오.

```
Event ID: 2023  
The Health Service has removed some items from the send queue for management group  
"Servers"  
since it exceeded the maximum allowed size of size megabytes.
```

```
Event ID: 2120  
The Health Service has deleted one or more items for management group "Servers" which  
could  
not be sent in 1440 minutes.
```

이벤트 6024

Windows 이벤트 로그에 다음 이벤트가 있는 경우 자세한 내용은 [SCOM 2012 - Event ID 6024](#)를 참조하십시오.

```
Event ID: 6024  
LaunchRestartHealthService.js : Launching Restart Health Service. Health Service exceeded  
Process\Handle Count or Private Bytes threshold.
```

System Center 2012 — Operations Manager에 대한 일반 문제 해결

다음 문제 해결 절차를 수행해 보십시오.

- System Center 2012 — Operations Manager에 대한 최신 업데이트 룰업을 설치했는지 확인합니다. AWS Management Pack에는 업데이트 룰업 1 이상이 필요합니다.
- Add Monitoring Wizard를 실행하여 AWS Management Pack를 가져온 후 구성했는지 확인합니다. 자세한 내용은 [1단계: AWS Management Pack 설치 \(p. 1248\)](#) 단원을 참조하십시오.

- AWS 리소스를 검색하는 동안 10–20분 정도 기다립니다.
- 관리 서버가 제대로 구성되어 있는지 확인합니다.
 - 관리 서버는 인터넷에 연결되어 있어야 합니다.
 - 관리 서버에 대한 작업 계정은 관리 서버에 대한 로컬 관리자 권한이 있어야 합니다.
 - 관리 서버에 .NET Framework 4.5 이상이 있어야 합니다.
- AWS Run As 계정이 유효한지 확인합니다.
 - 액세스 키 ID 및 보안 액세스 키의 값이 올바릅니다.
 - 액세스 키가 활성 상태인 경우: AWS Management 콘솔의 탐색 모음에서 사용자 이름을 클릭하고 보안 자격 증명을 클릭합니다.
 - IAM 사용자에게 최소한 읽기 전용 액세스 권한이 있습니다. 읽기 전용 액세스는 리소스의 상태를 변경하지 않는 사용자 작업(예: 모니터링)을 허용하지만, 인스턴스 시작 또는 중지와 같은 사용자 작업은 허용하지 않습니다.
 - Amazon CloudWatch 지표가 Not Monitored(모니터링되지 않음)로 표시되는 경우 Amazon CloudWatch 지표에 대해 Amazon CloudWatch 경보가 하나 이상 정의되어 있는지 확인합니다.
 - 이벤트 로그에서 자세한 문제 해결 정보를 확인합니다.
 - 관리 서버의 Operations Manager 이벤트 로그를 확인합니다. 자세한 내용은 [이벤트 \(p. 1276\)](#) 항목에서 AWS Management Pack가 Operations Manager 이벤트 로그에 작성하는 이벤트 목록을 참조하십시오.

System Center 2007 R2에 대한 일반 문제 해결

다음 문제 해결 절차를 수행해 보십시오.

- Add Monitoring Wizard를 실행하여 AWS Management Pack을 가져온 후 구성했는지 확인합니다. 자세한 내용은 [1단계: AWS Management Pack 설치 \(p. 1248\)](#) 단원을 참조하십시오.
- AWS 리소스를 검색하는 동안 10–20분 정도 기다립니다.
- 감시자 노드가 제대로 구성되어 있는지 확인합니다.
 - 프록시 에이전트가 활성화되어 있습니다. 자세한 내용은 [2단계: 감시자 노드 구성 \(p. 1250\)](#) 단원을 참조하십시오.
 - 감시자 노드가 인터넷에 연결되어 있습니다.
 - 감시자 노드에 대한 작업 계정에 로컬 관리자 권한이 있습니다.
 - 감시자 노드에 .NET Framework 3.5.1 이상이 있어야 합니다.
- 감시자 노드가 정상인지 확인하고 모든 경고를 해결합니다. 자세한 내용은 [보기 \(p. 1260\)](#) 단원을 참조하십시오.
- AWS Run As 계정이 유효한지 확인합니다.
 - 액세스 키 ID 및 보안 액세스 키의 값이 올바릅니다.
 - 액세스 키가 활성 상태인 경우: AWS Management 콘솔의 탐색 모음에서 사용자 이름을 클릭하고 보안 자격 증명을 클릭합니다.
 - IAM 사용자에게 최소한 읽기 전용 액세스 권한이 있습니다. 읽기 전용 액세스는 리소스의 상태를 변경하지 않는 사용자 작업(예: 모니터링)을 허용하지만, 인스턴스 시작 또는 중지와 같은 사용자 작업은 허용하지 않습니다.
 - Amazon CloudWatch 지표가 Not Monitored(모니터링되지 않음)로 표시되는 경우 Amazon CloudWatch 지표에 대해 Amazon CloudWatch 경보가 하나 이상 정의되어 있는지 확인합니다.
 - 이벤트 로그에서 자세한 문제 해결 정보를 확인합니다.
 - 관리 서버와 감시자 노드의 Operations Manager 이벤트 로그를 확인합니다. 자세한 내용은 [이벤트 \(p. 1276\)](#) 항목에서 AWS Management Pack가 Operations Manager 이벤트 로그에 작성하는 이벤트 목록을 참조하십시오.

문서 기록

다음 표에서는 2019년 이후의 Amazon EC2 설명서에 대한 중요 추가 사항을 설명합니다. 사용자로부터 받은 의견을 수렴하기 위해 설명서가 자주 업데이트됩니다.

update-history-change	update-history-description	update-history-date
M5a 및 R5a에 대한 최대 절전 모드 지원	M5a 및 R5a 인스턴스 유형에서 실행되는 새로 시작된 인스턴스를 최대 절전 모드로 전환할 수 있습니다.	August 28, 2020
인스턴스 메타데이터는 인스턴스 위치 및 배치 정보를 제공합니다.	placement 범주 아래의 새 인스턴스 메타데이터 필드: 리전, 배치 그룹 이름, 파티션 번호, 호스트 ID 및 가용 영역 ID.	August 24, 2020
Amazon EBS에 대해 프로비저닝된 IOPS SSD(io2) 볼륨	프로비저닝된 IOPS SSD(io2) 볼륨은 99.999%의 볼륨 내구성을 제공하도록 설계되었으며 AFR은 0.001% 이하입니다.	August 24, 2020
C5ad 인스턴스 (p. 1284)	2세대 AMD EYPC 프로세서를 갖춘 새로운 컴퓨팅 최적화 인스턴스	August 13, 2020
Wavelength Zone	Wavelength Zone은 Wavelength 인프라가 배포된 통신 사업자 위치의 격리된 영역입니다.	August 6, 2020
용량 예약 그룹	AWS 리소스 그룹을 사용하여 용량 예약의 논리적 모음을 생성한 다음 해당 그룹으로 대상 인스턴스를 시작할 수 있습니다.	July 29, 2020
빠른 스냅샷 복원	본인에게 공유된 스냅샷에 대해 빠른 스냅샷 복구를 활성화 할 수 있습니다.	July 21, 2020
EC2Launch v2 (p. 461)	인스턴스 시작 중, 인스턴스가 중지된 후 나중에 시작된 경우, 인스턴스가 재시작된 경우 및 온디맨드로 EC2Launch v2를 사용해 작업을 수행할 수 있습니다. EC2Launch v2는 모든 버전의 Windows Server를 지원하며 EC2Launch와 EC2Config를 대체합니다.	June 30, 2020
G4용 베어 메탈 인스턴스 (p. 1284)	애플리케이션이 호스트 서버의 물리적 리소스에 직접 액세스할 수 있는 새 인스턴스입니다.	June 5, 2020
C5a 인스턴스 (p. 1284)	2세대 AMD EYPC 프로세서를 갖춘 새로운 컴퓨팅 최적화 인스턴스	June 4, 2020

고유 IPv6 주소 가져오기	온프레미스 네트워크에서 AWS 계정으로 IPv6 주소 범위의 부분 또는 전체를 가져올 수 있습니다.	May 21, 2020
시스템 관리자 파라미터를 사용하여 인스턴스 시작	인스턴스를 시작할 때 AMI 대신 AWS 시스템 관리자 파라미터를 지정할 수 있습니다.	May 5, 2020
예약된 이벤트 알림 사용자 지정	이메일 알림에 태그를 포함하도록 예약된 이벤트 알림을 사용자 지정할 수 있습니다.	May 4, 2020
전용 호스트의 Windows Server	Amazon에서 제공하는 Windows Server AMI를 사용하여 전용 호스트에서 최신 Windows Server 버전을 실행할 수 있습니다.	April 7, 2020
스팟 인스턴스 중지 및 시작	이제 중지 중단 동작을 사용하는 대신 Amazon EBS에 의해 지원되는 스팟 인스턴스를 중지했다가 원활 때 시작할 수 있습니다.	January 13, 2020
리소스에 태그 지정 (p. 1284)	외부 전용 인터넷 게이트웨이, 로컬 게이트웨이, 로컬 게이트웨이 라우팅 테이블, 로컬 게이트웨이 가상 인터페이스, 로컬 게이트웨이 가상 인터페이스 그룹, 로컬 게이트웨이 라우팅 테이블 VPC 연결 및 로컬 게이트웨이 라우팅 테이블 가상 인터페이스 그룹 연결에 태그를 지정할 수 있습니다.	January 10, 2020
세션 관리자를 사용하여 인스턴스에 연결	Amazon EC2 콘솔에서 인스턴스로 세션 관리자 세션을 시작할 수 있습니다.	December 18, 2019
전용 호스트 및 호스트 리소스 그룹	이제 전용 호스트를 호스트 리소스 그룹과 함께 사용할 수 있습니다.	December 2, 2019
전용 호스트 공유	이제 AWS 계정 전체에 전용 호스트를 공유할 수 있습니다.	December 2, 2019
계정 수준의 기본 크레딧 사양	AWS 리전별로 계정 수준에서 성능 순간 확장 가능 인스턴스 패밀리마다 기본 크레딧 사양을 설정할 수 있습니다.	November 25, 2019
인스턴스 유형 검색	필요에 맞는 인스턴스 유형을 찾을 수 있습니다.	November 22, 2019
전용 호스트 (p. 1284)	이제 한 인스턴스 패밀리의 여러 인스턴스 유형을 지원하도록 전용 호스트를 구성할 수 있습니다.	November 21, 2019

Amazon EBS 빠른 스냅샷 복원	EBS 스냅샷에서 빠른 스냅샷 복원을 활성화하여 스냅샷에서 생성된 EBS 볼륨이 생성 시 완전히 초기화되고 모든 프로비저닝된 성능을 즉시 제공하도록 할 수 있습니다.	November 20, 2019
인스턴스 메타데이터 서비스 버전 2	인스턴스 메타데이터를 요청하는 세션 지향 방법인 인스턴스 메타데이터 서비스 버전 2를 사용할 수 있습니다	November 19, 2019
온디맨드 Windows 인스턴스에 대한 최대 절전 모드 지원	온디맨드 Windows 인스턴스를 최대 절전 모드로 실행할 수 있습니다.	October 14, 2019
예약 인스턴스 대기열 구매	최대 3년 전에 예약 인스턴스 구매를 대기할 수 있습니다.	October 4, 2019
G4 인스턴스 (p. 1284)	NVIDIA Tesla GPU를 제공하는 새 인스턴스입니다.	September 19, 2019
진단 인터럽트	연결할 수 없거나 응답이 없는 인스턴스에 진단 인터럽트를 보내 블루 스크린/중지 오류를 트리거 할 수 있습니다.	August 14, 2019
용량 최적화 할당 전략	이제 EC2 집합 또는 스팟 집합을 사용하여 시작 종인 인스턴스 수에 대한 용량이 최적화된 스팟 플에서 스팟 인스턴스를 시작할 수 있습니다.	August 12, 2019
온디맨드 용량 예약 공유	이제 AWS 계정에 걸쳐 용량 예약을 공유할 수 있습니다.	July 29, 2019
리소스에 태그 지정 (p. 1284)	작성 시 실행 템플릿에 태그를 지정할 수 있습니다.	July 24, 2019
호스트 복구	전용 호스트에서 예상치 못한 하드웨어 오류가 발생하는 경우 새 호스트에서 인스턴스를 자동으로 다시 시작합니다.	June 5, 2019
Amazon EBS 다중 볼륨 스냅샷	EC2 인스턴스에 연결된 여러 EBS 볼륨에서 정확한 특정 시점, 데이터 조정 및 충돌 일치 스냅샷을 생성할 수 있습니다.	May 29, 2019
리소스에 태그 지정 (p. 1284)	전용 호스트 예약에 태그를 지정할 수 있습니다.	May 27, 2019
Amazon EBS 암호화 기본 제공	리전에서 기본적으로 암호화를 사용하도록 설정하면 해당 리전에서 생성한 모든 새 EBS 볼륨이 EBS 암호화용 기본 CMK를 사용하여 암호화됩니다.	May 23, 2019

리소스에 태그 지정 (p. 1284)	VPC 엔드포인트, 엔드포인트 서비스 및 엔드포인트 서비스 구성 을 태그 지정할 수 있습니다.	May 13, 2019
VSS 애플리케이션이 일치하는 스냅샷	AWS 시스템 관리자 Run Command를 사용하여 Windows 인스턴스에 연결된 모든 Amazon EBS 볼륨에 대해 애플리케이션이 일치하는 스냅샷을 생성합니다.	May 13, 2019
Windows에서 Linux로 Microsoft SQL Server 데이터베이스를 위한 리플랫포밍 어시스턴트	기존 Microsoft SQL Server 워크 로드를 Windows에서 Linux 운영 체제로 이전합니다.	May 8, 2019
I3en 인스턴스 (p. 1284)	새 I3en 인스턴스는 최대 100Gbps의 네트워크 대역폭을 사용할 수 있습니다.	May 8, 2019
Windows 자동 업그레이드	AWS 시스템 관리자를 사용하여 EC2 Windows 인스턴스의 자동 업그레이드를 수행합니다.	May 6, 2019
T3a 인스턴스 (p. 1284)	AMD EYPC 프로세서를 제공하는 새 인스턴스입니다.	April 24, 2019
M5ad 및 R5ad 인스턴스 (p. 1284)	AMD EYPC 프로세서를 제공하는 새 인스턴스입니다.	March 27, 2019
리소스에 태그 지정 (p. 1284)	전용 호스트 예약에 사용자 지정 태그를 할당하여 예약을 다양한 방식으로 분류할 수 있습니다.	March 14, 2019
M5, M5d, R5, R5d, z1d에 대한 베어 메탈 인스턴스 (p. 1284)	애플리케이션이 호스트 서버의 물리적 리소스에 직접 액세스할 수 있는 새 인스턴스입니다.	February 13, 2019

지난 몇 년의 기록

다음 표에서는 2018년 및 이전 몇 년의 Amazon EC2 설명서에 대한 중요 추가 사항을 설명합니다.

기능	API 버전	설명	릴리스 날짜
파티션 배치 그룹	2016-11-15	파티션 배치 그룹은 인스턴스를 논리적 파티션에 분산해, 한 파티션에 있는 인스턴스가 다른 파티션의 인스턴스와 기본 하드웨어를 공유하지 않게 합니다. 자세한 내용은 파티션 배치 그룹 (p. 763) 단원을 참조하십시오.	2018년 12월 20일
p3dn.24xlarge 인스턴스	2016-11-15	새 p3dn.24xlarge 인스턴스는 100Gbps의 네트워크 대역폭을 제공합니다.	2018년 12월 7일
100Gbps의 네트워크 대역폭을 제공하는 인스턴스	2016-11-15	새 C5n 인스턴스는 최대 100Gbps의 네트워크 대역폭을 사용할 수 있습니다.	2018년 11월 26일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
스팟 콘솔에서 인스턴스 플릿 권장	2016-11-15	스팟 콘솔은 애플리케이션 요구에 맞는 최소 하드웨어 사양(vCPU, 메모리 및 스토리지)을 충족하기 위해 스팟 모범 사례(인스턴스 다각화)에 기초한 인스턴스 플릿을 권장합니다. 자세한 내용은 스팟 집합 요청 생성 (p. 277) 단원을 참조하십시오.	2018년 11 월 20일
새 EC2 집합 요청 유형: instant	2016-11-15	EC2 집합에서는 이제 새로운 요청 유형인 instant를 지원합니다. 이 요청 유형은 인스턴스 유형 및 구매 모델 간에 용량을 동기적으로 프로비저닝하는 데 사용할 수 있습니다. instant 요청은 시작된 인스턴스를 API 응답으로 반환하며 추가 작업을 수행하지 않으므로 인스턴스가 시작되는 경우와 시점을 제어할 수 있습니다. 자세한 내용은 EC2 집합 요청 유형 (p. 403) 단원을 참조하십시오.	2018년 11 월 14일
AMD EYPC 프로세서를 제공하는 인스턴스	2016-11-15	새로운 범용(M5a) 및 메모리 최적화된 인스턴스(R5a)는 마이크로서비스, 중소 규모 데이터베이스, 가상 데스크톱, 개발 및 테스트 환경, 비즈니스 애플리케이션 등을 위한 저가형 옵션을 제공합니다.	2018년 11 월 6일
스팟 절감 정보	2016-11-15	단일 스팟 집합 또는 모든 스팟 인스턴스에 대해 스팟 인스턴스 사용에서 얻는 절감을 볼 수 있습니다. 자세한 내용은 스팟 인스턴스 구입으로 절감되는 비용 (p. 253) 단원을 참조하십시오.	2018년 11 월 5일
CPU 옵션 최적화를 위한 콘솔 지원	2016-11-15	인스턴스를 시작하면 Amazon EC2 콘솔을 사용하여 특정 워크로드 또는 비즈니스 필요에 맞도록 CPU 옵션을 최적화할 수 있습니다. 자세한 내용은 CPU 옵션 최적화 (p. 541) 단원을 참조하십시오.	2018년 10 월 31일
인스턴스에서 시작 템플릿 생성을 위한 콘솔 지원	2016-11-15	Amazon EC2 콘솔을 사용하는 새 시작 템플릿을 위한 기초로 인스턴스를 사용하는 시작 템플릿을 생성할 수 있습니다. 자세한 내용은 시작 템플릿 생성 (p. 383) 단원을 참조하십시오.	2018년 10 월 30일
온디맨드 용량 예약	2016-11-15	원하는 기간 동안 특정 가용 영역의 Amazon EC2 인스턴스에 대해 용량을 예약할 수 있습니다. 이렇게 하면 예약 인스턴스(RI)에서 제공되는 결제 할인과는 별도로 용량 예약을 생성 및 관리할 수 있습니다. 자세한 내용은 온디맨드 용량 예약 (p. 354) 단원을 참조하십시오.	2018년 10 월 25일
고유 IP 주소 가져오기 (BYOIP)	2016-11-15	온프레미스 네트워크에서 AWS 계정으로 모든 퍼블릭 IPv4 주소 범위의 일부 또는 전체를 가져올 수 있습니다. 주소 범위를 AWS로 가져오고 나면 이러한 주소가 계정에 주소 폴로 나타납니다. 주소 폴에서 탄력적 IP 주소를 생성하여 AWS 리소스와 함께 사용할 수 있습니다. 자세한 내용은 Amazon EC2의 고유 IP 주소 가져오기(BYOIP) (p. 716) 단원을 참조하십시오.	2018년 10 월 23일
g3s.xlarge 인스턴스	2016-11-15	g3s.xlarge 인스턴스의 도입과 함께 액셀러레이티드 컴퓨팅 G3 인스턴스 패밀리를 확장합니다.	2018년 10 월 11일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
생성 및 콘솔 지원 시 전용 호스트 태그	2016-11-15	생성 시 전용 호스트를 태그 지정할 수 있으므로 Amazon EC2 콘솔을 사용하여 전용 호스트 태그를 관리할 수 있습니다. 자세한 내용은 전용 호스트 할당 (p. 324) 단원을 참조하십시오.	2018년 10 월 08일
고용량 메모리 인스턴스	2016-11-15	이러한 인스턴스는 대량의 인 메모리 데이터베이스를 실행하도록 설계되었습니다. 베어 메탈 성능에 호스트 하드웨어에 대한 직접 액세스를 제공합니다. 자세한 내용은 메모리 최적화 인스턴스 (p. 168) 단원을 참조하십시오.	2018년 9 월 27일
f1.4xlarge 인스턴스	2016-11-15	f1.4xlarge 인스턴스의 도입과 함께 액셀러레이티드 컴퓨팅 F1 인스턴스 패밀리를 확장합니다.	2018년 9 월 25일
스팟 집합에 대한 예약된 조정 작업의 콘솔 지원	2016-11-15	날짜 및 시간을 기준으로 플릿의 현재 용량을 늘리거나 줄입니다. 자세한 내용은 예약된 조정을 사용하여 스팟 집합 조정 (p. 303) 단원을 참조하십시오.	2018년 9 월 20일
T3 인스턴스	2016-11-15	T3; 인스턴스는 차세대 버스트 가능 범용 인스턴스 유형으로서, 기본 수준의 CPU 성능을 제공하면서도 필요에 따라 언제든지 CPU 사용량을 버스트할 수 있습니다. 자세한 내용은 성능 순간 확장 가능 인스턴스 (p. 129) 단원을 참조하십시오.	2018년 8 월 21일
EC2 집합을 위한 할당 전략	2016-11-15	온디맨드 용량을 가격(최저 가격 우선)을 기준으로 채울지 또는 우선 순위(최우선 순위 우선)를 기준으로 채울지 지정할 수 있습니다. 대상 스팟 용량을 할당할 스팟 폴 수를 지정할 수 있습니다. 자세한 내용은 스팟 인스턴스를 위한 할당 전략 (p. 403) 단원을 참조하십시오.	2018년 7 월 26일
스팟 집합을 위한 할당 전략	2016-11-15	온디맨드 용량을 가격(최저 가격 우선)을 기준으로 채울지 또는 우선 순위(최우선 순위 우선)를 기준으로 채울지 지정할 수 있습니다. 대상 스팟 용량을 할당할 스팟 폴 수를 지정할 수 있습니다. 자세한 내용은 스팟 인스턴스의 할당 전략 (p. 246) 단원을 참조하십시오.	2018년 7 월 26일
R5 및 R5d 인스턴스	2016-11-15	R5 및 R5d 인스턴스는 고성능 데이터베이스, 배포된 인 메모리 캐시 및 인 메모리 분석에 가장 적합합니다. R5d 인스턴스에는 NVMe 인스턴스 스토어 볼륨이 연결되어 있습니다. 자세한 내용은 메모리 최적화 인스턴스 (p. 168) 단원을 참조하십시오.	2018년 7 월 25일
z1d 인스턴스	2016-11-15	이러한 인스턴스는 전자 설계 자동화(EDA) 및 관계형 데이터베이스 등 대용량 메모리와 함께 높은 코어당 성능을 필요로 하는 애플리케이션을 위해 설계되었습니다. 이러한 인스턴스에는 NVME 인스턴스 스토어 볼륨이 연결되어 있습니다. 자세한 내용은 메모리 최적화 인스턴스 (p. 168) 단원을 참조하십시오.	2018년 7 월 25일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
스냅샷 수명 주기 자동화	2016-11-15	Amazon 데이터 수명 주기 관리자를 사용하여 EBS 볼륨의 스냅샷 생성 및 삭제를 자동화할 수 있습니다. 자세한 내용은 Amazon EBS 스냅샷 수명 주기 자동화 (p. 1012) 단원을 참조하십시오.	2018년 7 월 12일
시작 템플릿 CPU 옵션	2016-11-15	명령행 도구를 사용하여 시작 템플릿을 생성할 때 특정 워크로드 또는 비즈니스 요구 사항에 맞춰 CPU 옵션을 최적화할 수 있습니다. 자세한 내용은 시작 템플릿 생성 (p. 383) 단원을 참조하십시오.	2018년 7 월 11일
전용 호스트 태그	2016-11-15	전용 호스트에 태그를 지정할 수 있습니다. 자세한 내용은 전용 호스트 태그 지정 (p. 334) 단원을 참조하십시오.	2018년 7 월 3일
i3.metal 인스턴스	2016-11-15	i3.metal 인스턴스에서는 애플리케이션이 프로세서, 메모리 등 호스트 서버의 물리적 리소스에 직접 액세스할 수 있습니다. 자세한 내용은 스토리지 최적화 인스턴스 (p. 177) 단원을 참조하십시오.	2018년 5 월 17일
최신 콘솔 출력 받기	2016-11-15	<code>get-console-output</code> AWS CLI 명령을 사용하면 일부 인스턴스 유형의 최신 콘솔 출력을 검색할 수 있습니다.	2018년 5 월 9일
CPU 옵션 최적화	2016-11-15	인스턴스를 시작하면 특정 워크로드 또는 비즈니스 필요에 맞도록 CPU 옵션을 최적화할 수 있습니다. 자세한 내용은 CPU 옵션 최적화 (p. 541) 단원을 참조하십시오.	2018년 5 월 8일
EC2 집합	2016-11-15	EC2 집합을 사용하여 서로 다른 EC2 인스턴스 유형과 가용 영역, 온디맨드 인스턴스, 예약 인스턴스 및 스팟 인스턴스 구매 모델에서 인스턴스 그룹을 시작할 수 있습니다. 자세한 내용은 EC2 집합을 사용하여 인스턴스 시작 (p. 400) 단원을 참조하십시오.	2018년 5 월 2일
스팟 집합의 온디맨드 인스턴스	2016-11-15	항상 인스턴스 용량을 사용할 수 있도록 스팟 집합 요청에 온디맨드 용량에 대한 요청을 포함할 수 있습니다. 자세한 내용은 스팟 집합 작동 방식 (p. 246) 단원을 참조하십시오.	2018년 5 월 2일
생성 시 EBS 스냅샷 태그 지정	2016-11-15	생성 중 스냅샷에 태그를 적용할 수 있습니다. 자세한 내용은 Amazon EBS 스냅샷 생성 (p. 969) 단원을 참조하십시오.	2018년 4 월 2일
배치 그룹 변경	2016-11-15	배치 그룹 내로 또는 밖으로 인스턴스를 이동하거나, 인스턴스의 배치 그룹을 변경할 수 있습니다. 자세한 내용은 인스턴스의 배치 그룹 변경 (p. 770) 단원을 참조하십시오.	2018년 3 월 1일
더 긴 리소스 ID	2016-11-15	더 많은 리소스 유형에 대해 더 긴 ID 형식을 활성화 할 수 있습니다. 자세한 내용은 리소스 ID (p. 1133) 단원을 참조하십시오.	2018년 2 월 9일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
네트워크 성능 개선	2016-11-15	클러스터 배치 그룹 외부의 인스턴스에서 이제 다른 인스턴스 또는 Amazon S3 사이에서 네트워크 트래픽을 전송 또는 수신할 때 증가된 대역폭의 혜택을 누릴 수 있습니다. 자세한 내용은 네트워킹 및 스토리지 기능 (p. 119) 단원을 참조하십시오.	2018년 1월 24일
탄력적 IP 주소 태그 지정	2016-11-15	탄력적 IP 주소에 태그를 지정할 수 있습니다. 자세한 내용은 탄력적 IP 주소 태그 지정 (p. 724) 단원을 참조하십시오.	2017년 12월 21일
Amazon Time Sync Service	2016-11-15	Amazon Time Sync Service를 사용하여 인스턴스에서 정확한 시간을 유지할 수 있습니다. 자세한 내용은 단원을 참조하십시오. Windows 인스턴스에 대한 시간 설정 (p. 556)	2017년 11월 29일
T2 무제한	2016-11-15	T2 무제한 인스턴스는 필요한 기간 동안 기준 이상으로 워크로드를 버스트할 수 있습니다. 자세한 내용은 성능 순간 확장 가능 인스턴스 (p. 129) 단원을 참조하십시오.	2017년 11월 29일
시작 템플릿	2016-11-15	시작 템플릿에는 인스턴스를 시작하기 위한 파라미터가 전부 또는 일부 포함되어 있기 때문에 인스턴스를 시작할 때마다 이를 지정할 필요가 없습니다. 자세한 내용은 시작 템플릿에서 인스턴스 시작 (p. 381) 단원을 참조하십시오.	2017년 11월 29일
분산형 배치	2016-11-15	서로 떨어져 있어야 하는 중요 인스턴스의 수가 적은 애플리케이션에서는 분산형 배치 그룹이 권장됩니다. 자세한 내용은 분산형 배치 그룹 (p. 764) 단원을 참조하십시오.	2017년 11월 29일
H1 인스턴스	2016-11-15	H1 인스턴스는 고성능 빅 데이터 워크로드용으로 설계되었습니다. 자세한 내용은 스토리지 최적화 인스턴스 (p. 177) 단원을 참조하십시오.	2017년 11월 28일
M5 인스턴스	2016-11-15	M5 인스턴스는 차세대 범용 컴퓨팅 인스턴스로서, 이들은 컴퓨팅, 메모리, 스토리지 및 네트워크 리소스를 균형 있게 제공합니다.	2017년 11월 28일
스팟 인스턴스 최대 절전 모드	2016-11-15	스팟 서비스는 중단 시 스팟 인스턴스를 최대 절전 모드로 전환할 수 있습니다. 자세한 내용은 중단된 스팟 인스턴스를 최대 절전 모드로 전환 (p. 312) 단원을 참조하십시오.	2017년 11월 28일
스팟 집합 대상 추적	2016-11-15	스팟 집합에 대해 대상 추적 조정 정책을 설정할 수 있습니다. 자세한 내용은 대상 추적 정책을 사용하여 스팟 집합 조정 (p. 301) 단원을 참조하십시오.	2017년 11월 17일
Elastic Load Balancing 와 스팟 집합 통합	2016-11-15	스팟 집합에 하나 이상의 로드 밸런서를 연결할 수 있습니다.	2017년 11월 10일
X1e 인스턴스	2016-11-15	X1e 인스턴스는 고성능 데이터베이스, 인메모리 데이터베이스 및 기타 메모리 집약적인 엔터프라이즈 애플리케이션에 가장 적합합니다. 자세한 내용은 메모리 최적화 인스턴스 (p. 168) 단원을 참조하십시오.	2017년 11월 28일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
C5 인스턴스	2016-11-15	C5 인스턴스는 컴퓨팅 집약적 애플리케이션을 위하여 설계되었습니다. 자세한 내용은 컴퓨팅 최적화 인스턴스 (p. 163) 단원을 참조하십시오.	2017년 11 월 6일
전환형 예약 인스턴스 병합 및 분할	2016-11-15	둘 이상의 전환형 예약 인스턴스를 새 전환형 예약 인스턴스 하나로 교환(병합)할 수 있습니다. 전환형 예약 인스턴스 한 개를 작은 예약 여러 개로 분리하는 수정 과정을 이용할 수도 있습니다. 자세한 내용은 전환형 예약 인스턴스 교환 (p. 232) 단원을 참조하십시오.	2017년 11 월 6일
P3 인스턴스	2016-11-15	P3 인스턴스는 차세대 컴퓨팅 최적화 GPU 인스턴스입니다. 자세한 내용은 Windows 액셀러레이티드 컴퓨팅 인스턴스 (p. 182) 단원을 참조하십시오.	2017년 10 월 25일
VPC 테넌시 수정	2016-11-15	VPC의 인스턴스 테넌시 속성을 dedicated에서 default로 변경할 수 있습니다. 자세한 내용은 VPC의 테넌시 변경 (p. 354) 단원을 참조하십시오.	2017년 10 월 16일
중단 시 종지	2016-11-15	스팟 인스턴스가 중단되면 Amazon EC2가 이를 종지하거나 종료하도록 지정할 수 있습니다. 자세한 내용은 중단 동작 (p. 311) 단원을 참조하십시오.	2017년 9 월 18일
NAT 게이트웨이에 태그 지정	2016-11-15	NAT 게이트웨이에 태그를 지정할 수 있습니다. 자세한 내용은 리소스에 태그 지정 (p. 1144) 단원을 참조하십시오.	2017년 9 월 7일
보안 그룹 규칙 설명	2016-11-15	보안 그룹 규칙에 설명을 추가할 수 있습니다. 자세한 내용은 보안 그룹 규칙 (p. 909) 단원을 참조하십시오.	2017년 8 월 31일
Elastic Graphics	2016-11-15	Elastic Graphics 액셀러레이터를 인스턴스에 연결하여 애플리케이션의 그래픽 성능을 높이십시오. 자세한 내용은 Amazon Elastic Graphics (p. 636) 단원을 참조하십시오.	2017년 8 월 29일
탄력적 IP 주소 복구	2016-11-15	VPC에서 사용하기 위해 탄력적 IP 주소를 해제한 경우 복구할 수 있습니다. 자세한 내용은 탄력적 IP 주소 복구 (p. 727) 단원을 참조하십시오.	2017년 8 월 11일
스팟 집합 인스턴스 태그 지정	2016-11-15	시작하는 인스턴스에 태그를 자동으로 지정하도록 스폷 집합을 구성할 수 있습니다.	2017년 7 월 24일
G3 인스턴스	2016-11-15	G3 인스턴스는 DirectX 또는 OpenGL을 사용하는 그래픽 애플리케이션을 위해 비용 효율적인 고성능 플랫폼을 제공합니다. 또한 G3 인스턴스는 해상도가 최대 4096x2160인 모니터 4대를 지원하는 NVIDIA GRID 가상 워크스테이션 기능도 제공합니다. 자세한 내용은 Windows 액셀러레이티드 컴퓨팅 인스턴스 (p. 182) 단원을 참조하십시오.	2017년 7 월 13일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
생성 시 태그 리소스	2016-11-15	생성 단계에서 인스턴스와 볼륨에 태그를 적용할 수 있습니다. 자세한 내용은 리소스에 태그 지정 (p. 1144) 단원을 참조하십시오. 또한 태그 기반 리소스 권한을 사용하여 적용되는 태그를 제어할 수도 있습니다. 자세한 내용은 생성 시 리소스 태그 지정에 대한 권한 부여 (p. 843) 단원을 참조하십시오.	2017년 3 월 28일
I3 인스턴스	2016-11-15	I3 인스턴스는 차세대 스토리지 최적화 인스턴스를 나타냅니다. 자세한 내용은 스토리지 최적화 인스턴스 (p. 177) 단원을 참조하십시오.	2017년 2 월 23일
연결된 EBS 볼륨을 수정합니다	2016-11-15	대부분의 EC2 인스턴스에 연결된 대다수 EBS의 경우, 볼륨을 분리하거나 인스턴스를 중지하지 않고도 볼륨 크기, 유형, IOPS를 수정할 수 있습니다. 자세한 내용은 Amazon EBS 탄력적 볼륨 (p. 1023) 단원을 참조하십시오.	2017년 2 월 13일
IAM 역할 연결	2016-11-15	기존 인스턴스에 대한 IAM 역할을 연결, 분리하거나 교체할 수 있습니다. 자세한 내용은 Amazon EC2의 IAM 역할 (p. 890) 단원을 참조하십시오.	2017년 2 월 9일
전용 스팟 인스턴스	2016-11-15	Virtual Private Cloud(VPC)의 단일 테넌트 하드웨어에서 스팟 인스턴스를 실행할 수 있습니다. 자세한 내용은 스팟 인스턴스의 테넌시 지정 (p. 256) 단원을 참조하십시오.	2017년 1 월 19일
IPv6 지원	2016-11-15	IPv6 CIDR를 VPC와 서브넷에 연결하고 IPv6 주소를 VPC의 인스턴스에 할당할 수 있습니다. 자세한 내용은 Amazon EC2 인스턴스 IP 주소 지정 (p. 702) 을 참조하십시오.	2016년 12 월 1일
R4 인스턴스	2016-09-15	R4 인스턴스는 차세대 메모리 최적화 인스턴스를 나타냅니다. R4 인스턴스는 BI(비즈니스 인텔리전스), 데이터 마이닝 및 분석, 인 메모리 데이터베이스, 분산형 웹 규모 인 메모리 캐싱, 비정형 빅 데이터의 애플리케이션 성능 실시간 처리 등 메모리 집약적이고 지연 시간에 민감한 워크로드에 매우 적합합니다. 자세한 내용은 메모리 최적화 인스턴스 (p. 168) 를 참조하십시오.	2016년 11 월 30일
새로운 t2.xlarge 및 t2.2xlarge 인스턴스 유형	2016-09-15	T2 인스턴스는 중간 정도의 기본 성능을 발휘하면서 작업의 필요에 따라 성능을 크게 높이는 버스트 기능을 제공하도록 설계되었습니다. 이러한 인스턴스는 경제적인 가격으로 제한된 시간 동안 빠른 응답성과 뛰어난 성능이 필요한 애플리케이션용입니다. 자세한 내용은 성능 순간 확장 가능 인스턴스 (p. 129) 단원을 참조하십시오.	2016년 11 월 30일
P2 인스턴스	2016-09-15	P2 인스턴스는 NVIDIA Tesla K80 GPU를 사용하며, CUDA 또는 OpenCL 프로그래밍 모델을 사용하는 일반 GPU 컴퓨팅에 맞게 설계되었습니다. 자세한 내용은 Windows 액셀러레이티드 컴퓨팅 인스턴스 (p. 182) 단원을 참조하십시오.	2016년 9 월 29일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
m4.16xlarge 인스턴스	2016-04-01	64개의 vCPU와 256GiB RAM을 장착한 m4.16xlarge 인스턴스를 도입하여 일반 M4 제품군의 범위를 확장합니다.	2016년 9 월 6일
스팟 집합의 자동 조정		이제 스팟 집합에 대한 조정 정책을 설정할 수 있습니다. 자세한 내용은 스팟 집합의 자동 조정 (p. 299) 단원을 참조하십시오.	2016년 9 월 1일
ENA(Elastic Network Adapter)	2016-04-01	이제 ENA를 사용하여 네트워킹 수준을 높일 수 있습니다. 자세한 내용은 향상된 네트워킹 유형 (p. 751) 단원을 참조하십시오.	2016년 6 월 28일
더 긴 ID 보기 및 수정을 위한 지원 기능 향상	2016-04-01	이제 다른 IAM 사용자, IAM 역할 또는 루트 사용자에 대한 더 긴 ID 설정을 보고 수정할 수 있습니다. 자세한 내용은 리소스 ID (p. 1133) 단원을 참조하십시오.	2016년 6 월 23일
AWS 계정 간 암호화된 Amazon EBS 스냅샷 복사	2016-04-01	이제 AWS 계정 간에 암호화된 EBS 스냅샷을 복사할 수 있습니다. 자세한 내용은 Amazon EBS 스냅샷 복사 (p. 985) 단원을 참조하십시오.	2016년 6 월 21일
인스턴스 콘솔의 스크린샷 캡처	2015-10-01	이제 접속할 수 없는 인스턴스를 디버깅할 때 추가 정보를 얻을 수 있습니다. 자세한 내용은 단원을 참조하십시오. 연결할 수 없는 인스턴스 문제 해결 (p. 1188)	2016년 5 월 24일
X1 인스턴스	2015-10-01	인 메모리 데이터베이스, 빅 데이터 처리 엔진 및 HPC(고성능 컴퓨팅) 애플리케이션을 실행하기 위해 설계된 메모리 최적화 인스턴스입니다. 자세한 내용은 메모리 최적화 인스턴스 (p. 168) 단원을 참조하십시오.	2016년 5 월 18일
두 가지 새로운 EBS 볼륨 유형	2015-10-01	이제 처리량에 최적화된 HDD(st1) 및 콜드 HDD(sc1) 볼륨을 생성할 수 있습니다. 자세한 내용은 Amazon EBS 볼륨 유형 (p. 932) 단원을 참조하십시오.	2016년 4 월 19일
Amazon EC2에 대한 새로운 NetworkPacketsIn 및 NetworkPacketsOut 측정치를 추가함.		Amazon EC2에 대한 새로운 NetworkPacketsIn 및 NetworkPacketsOut 측정치를 추가함. 자세한 내용은 인스턴스 지표 (p. 669) 단원을 참조하십시오.	2016년 3 월 23일
스팟 집합에 대한 CloudWatch 지표		이제 스팟 집합에 대한 CloudWatch 지표를 얻을 수 있습니다. 자세한 내용은 스팟 집합에 대한 CloudWatch 지표 (p. 297) 단원을 참조하십시오.	2016년 3 월 21일
예약된 인스턴스	2015-10-01	정기 예약 인스턴스(정기 인스턴스)를 사용하여 지정된 시작 시간과 기간에 따라 매일, 매주 또는 매월 반복적으로 용량 예약을 구입할 수 있습니다. 자세한 내용은 정기 예약 인스턴스 (p. 236) 단원을 참조하십시오.	2016년 1 월 13일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
더 긴 리소스 ID	2015-10-01	일부 Amazon EC2 및 Amazon EBS 리소스 유형에 좀 더 긴 ID를 도입하고 있습니다. 옵트 인 기간 동안 지원되는 리소스 유형에 더 긴 ID 형식을 사용할 수 있습니다. 자세한 내용은 리소스 ID (p. 1133) 단원을 참조하십시오.	2016년 1 월 13일
ClassicLink DNS 지원	2015-10-01	연결된 EC2-Classic 인스턴스와 VPC의 인스턴스 사이에서 처리되는 DNS 호스트 이름이 퍼블릭 IP 주소가 아니라 프라이빗 IP 주소로 확인되도록 VPC에 대해 ClassicLink DNS 지원을 비활성화할 수 있습니다. 자세한 내용은 ClassicLink DNS 지원 활성화 (p. 815) 단원을 참조하십시오.	2016년 1 월 11일
새 t2.nano 인스턴스 유형	2015-10-01	T2 인스턴스는 중간 정도의 기본 성능을 발휘하면서 작업의 필요에 따라 성능을 크게 높이는 버스트 기능을 제공하도록 설계되었습니다. 이러한 인스턴스는 경제적인 가격으로 제한된 시간 동안 빠른 응답성과 뛰어난 성능이 필요한 애플리케이션용입니다. 자세한 내용은 성능 순간 확장 가능 인스턴스 (p. 129) 단원을 참조하십시오.	2015년 12 월 15일
전용 호스트	2015-10-01	Amazon EC2 전용 호스트는 고객 전용의 인스턴스 용량을 갖춘 물리적 서버입니다. 자세한 내용은 전용 호스트 (p. 320) 단원을 참조하십시오.	2015년 11 월 23일
스팟 인스턴스 기간	2015-10-01	이제 스팟 인스턴스의 지속 시간을 지정할 수 있습니다. 자세한 정보는 스팟 인스턴스의 지속 시간 정의 (p. 256) 단원을 참조하십시오.	2015년 10 월 6일
스팟 집합 요청 수정	2015-10-01	이제 스팟 집합 요청의 목표 용량을 수정할 수 있습니다. 자세한 내용은 스팟 집합 요청 수정 (p. 287) 단원을 참조하십시오.	2015년 9 월 29일
스팟 집합 다각화 할당 전략	2015-04-15	단일 스팟 집합 요청을 사용하여 여러 스팟 풀에서 스팟 인스턴스를 할당할 수 있습니다. 자세한 정보는 스팟 인스턴스의 할당 전략 (p. 246) 단원을 참조하십시오.	2015년 9 월 15일
스팟 집합 인스턴스 가중 치 부여	2015-04-15	이제 각 인스턴스 유형이 애플리케이션의 성능에 기여하는 용량 단위를 정의하고, 그에 따라 각 스팟 풀의 스팟 인스턴스에 대해 지불할 금액을 조정할 수 있습니다. 자세한 정보는 스팟 집합 인스턴스 가중 치 부여 (p. 249) 단원을 참조하십시오.	2015년 8 월 31일
새로운 재부팅 경보 작업과 경보 작업에 사용할 새로운 IAM 역할		재부팅 경보 작업과 경보 작업에 사용할 새로운 IAM 역할이 추가되었습니다. 자세한 내용은 인스턴스를 중지, 종료, 재부팅 또는 복구하는 경보 만들기 (p. 688) 단원을 참조하십시오.	2015년 7 월 23일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
새 t2.large 인스턴스 유형		T2 인스턴스는 중간 정도의 기본 성능을 발휘하면서 작업의 필요에 따라 성능을 크게 높이는 버스트 기능을 제공하도록 설계되었습니다. 이러한 인스턴스는 경제적인 가격으로 제한된 시간 동안 빠른 응답성과 뛰어난 성능이 필요한 애플리케이션용입니다. 자세한 내용은 성능 순간 확장 가능 인스턴스 (p. 129) 단원을 참조하십시오.	2015년 6 월 16일
M4 인스턴스		컴퓨팅, 메모리 및 네트워크 리소스의 균형을 제공하는 차세대 범용 인스턴스입니다. M4 인스턴스는 AVX2가 포함된 사용자 지정 2.4GHz intel® Xeon® E5 2676v3(Haswell) 프로세서에 의해 구동됩니다.	2015년 6 월 11일
스팟 집합	2015-04-15	별도의 스팟 인스턴스 요청을 관리하는 대신 스팟 인스턴스의 모음 또는 집합을 관리할 수 있습니다. 자세한 내용은 스팟 집합 작동 방식 (p. 246) 단원을 참조하십시오.	2015년 5 월 18일
탄력적 IP 주소의 EC2-Classic 마이그레이션	2015-04-15	EC2-Classic에서 사용하도록 할당한 탄력적 IP 주소를 VPC에서 사용하도록 마이그레이션할 수 있습니다. 자세한 내용은 EC2-Classic에서 탄력적 IP 주소 마이그레이션 (p. 807) 단원을 참조하십시오.	2015년 5 월 15일
디스크가 여러 개 있는 VM을 AMI로 가져오기	2015-03-01	VM Import 프로세스는 이제 디스크가 여러 개 있는 VM을 AMI로 가져오기를 지원합니다. 자세한 내용은 VM Import/Export 사용 설명서에서 VM Import/Export를 사용하여 VM을 이미지로 가져오기 를 참조하십시오.	2015년 4 월 23일
새 g2.8xlarge 인스턴스 유형		새 g2.8xlarge 인스턴스는 4개의 고성능 NVIDIA GPU를 기반으로 제공되므로 대규모 렌더링, 트랜스코딩, 기계 학습 및 대량 병렬 처리 성능을 필요로 하는 기타 서버 측 워크로드를 비롯한 GPU 컴퓨팅 위크로드에 매우 적합합니다.	2015년 4 월 7일
D2 인스턴스		직접 연결 인스턴스 스토리지에서 대량 데이터에 순차적으로 액세스해야 하는 애플리케이션에 최적화된 차세대 Amazon EC2 집약적 스토리지 인스턴스입니다. D2 인스턴스는 집약적 스토리지 제품군에 최적의 가격/성능을 제공하도록 설계되었습니다. 2.4GHz intel® E5 2676v3(Haswell) 프로세서로 구동되는 D2 인스턴스는 보강된 컴퓨팅 능력, 증가된 메모리 및 향상된 네트워킹을 제공함으로써 HS1 인스턴스를 능가합니다. 또한 D2 인스턴스는 6TB, 12TB, 24TB 및 48TB 스토리지 옵션을 통해 네 가지 인스턴스 크기로 사용할 수 있습니다. 자세한 내용은 스토리지 최적화 인스턴스 (p. 177) 단원을 참조하십시오.	2015년 3 월 24일
시스템 관리자		시스템 관리자(를) 통해 EC2 인스턴스를 구성하고 관리할 수 있습니다.	2015년 2 월 17일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
Microsoft SCVMM 1.5용 시스템 관리자		<p>이제 Microsoft SCVMM용 시스템 관리자(를) 사용하여 인스턴스를 시작하고 SCVMM에서 Amazon EC2(으)로 VM을 가져올 수 있습니다. 자세한 내용은 EC2 인스턴스 생성 (p. 1234) 및 가상 머신 가져오기 (p. 1239) 단원을 참조하십시오.</p>	2015년 1월 21일
EC2 인스턴스용 자동 복구		<p>사용자는 Amazon EC2 인스턴스를 모니터링하고 기본 하드웨어 장애나 복구에 AWS 개입이 필요한 문제로 인해 인스턴스가 손상된 경우 인스턴스를 자동으로 복구하는 Amazon CloudWatch 경보를 만들 수 있습니다. 복구된 인스턴스는 인스턴스 ID, IP 주소 및 모든 인스턴스 메타데이터를 포함하여 원본 인스턴스와 동일합니다.</p> <p>자세한 내용은 인스턴스 복구 (p. 460) 단원을 참조하십시오.</p>	2015년 1월 12일
C4 인스턴스		<p>경제적인 가격으로 매우 우수한 성능을 제공하는 컴퓨팅에 최적화된 차세대 인스턴스입니다. C4 인스턴스는 사용자 지정 2.9GHz Intel® E5-2666 v3(Haswell) 프로세서를 기반으로 합니다. 터보 부스트 기능이 추가된 C4 인스턴스의 프로세서 클록 속도는 싱글/듀얼 코어의 터보 기능이 있는 3.5Ghz 만큼 빠릅니다. C3 컴퓨팅에 최적화된 인스턴스의 기능이 확장된 C4 인스턴스는 EC2 인스턴스 중에서 최고의 프로세서 성능을 고객에게 제공합니다. 이러한 인스턴스는 트래픽 양이 높은 웹 애플리케이션, 광고 서비스, 배치성 프로세스, 비디오 인코딩, 분산 분석, 고에너지 물리학, 게임 분석 및 계산 유체 역학에 매우 적합합니다.</p> <p>자세한 내용은 컴퓨팅 최적화 인스턴스 (p. 163) 단원을 참조하십시오.</p>	2015년 1월 11일
ClassicLink	2014-10-01	<p>ClassicLink를 사용하면 EC2-Classic 인스턴스를 계정의 VPC에 연결할 수 있습니다. VPC 보안 그룹을 EC2-Classic 인스턴스에 연결할 수 있으므로 EC2-Classic 인스턴스와 VPC의 인스턴스가 프라이빗 IP 주소를 사용하여 서로 통신할 수 있습니다. 자세한 내용은 ClassicLink (p. 809) 단원을 참조하십시오.</p>	2015년 1월 7일
스팟 인스턴스 종료 공지		<p>스팟 인스턴스 중단을 방지하는 가장 좋은 방법은 내결함성을 유지하도록 애플리케이션을 설계하는 것입니다. 또한 Amazon EC2가 스팟 인스턴스를 종료하기 2분 전에 경고하는 스팟 인스턴스 종료 공지를 이용할 수 있습니다.</p> <p>자세한 내용은 스팟 인스턴스 종단 공지 (p. 314) 단원을 참조하십시오.</p>	2015년 1월 5일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
시스템 관리자 for Microsoft SCVMM		시스템 관리자 for Microsoft SCVMM은 EC2 인스턴스 등의 AWS 리소스를 관리할 수 있는 간단하고 사용하기 쉬운 인터페이스를 제공합니다. 이 도구는 Microsoft SCVMM에서 제공합니다. 자세한 내용은 Microsoft System Center DPM용 AWS 시스템 관리자 (p. 1229) 단원을 참조하십시오.	2014년 10 월 29일
DescribeVolumes 페이지 매김 지원	2014-09-01	이제 <code>DescribeVolumes</code> API 호출에서는 <code>MaxResults</code> 및 <code>NextToken</code> 파라미터를 사용한 결과의 페이지 매김을 지원합니다. 자세한 내용은 Amazon EC2 API Reference의 DescribeVolumes 를 참조하십시오.	2014년 10 월 23일
Amazon CloudWatch Logs에 대한 지원 추가		Amazon CloudWatch Logs를 사용하여 인스턴스나 다른 원본에서 시스템, 애플리케이션 및 사용자 지정 로그 파일을 모니터링, 저장 및 액세스할 수 있습니다. Amazon CloudWatch 콘솔, AWS CLI의 CloudWatch Logs 명령 또는 CloudWatch Logs SDK를 사용하여 CloudWatch Logs에서 관련 로그 데이터를 가져올 수 있습니다.	2014년 7 월 10일
T2 인스턴스	2014-06-15	T2 인스턴스는 중간 정도의 기본 성능을 발휘하면서 작업의 필요에 따라 성능을 크게 높이는 버스트 기능을 제공하도록 설계되었습니다. 이러한 인스턴스는 경제적인 가격으로 제한된 시간 동안 빠른 응답성과 뛰어난 성능이 필요한 애플리케이션용입니다. 자세한 내용은 성능 순간 확장 가능 인스턴스 (p. 129) 단원을 참조하십시오.	2014년 6 월 30일
새 EC2 서비스 제한 페이지		Amazon EC2 콘솔의 EC2 서비스 제한 페이지에서는 리전별로 Amazon EC2 및 Amazon VPC에서 제공하는 리소스의 현재 제한을 볼 수 있습니다.	2014년 6 월 19일
Amazon EBS 범용 SSD 볼륨	2014-05-01	범용 SSD 볼륨은 광범위한 작업에서 이상적으로 사용될 수 있는 비용 효과적인 스토리지를 제공합니다. 이러한 볼륨은 시간을 연장할 경우 3 IOPS의 버스트 기능까지 지원되어 자연 시간이 한 자릿수 밀리초에 불과하며 3,000 IOPS/GiB를 기본 성능으로 제공합니다. 범용 SSD 볼륨 크기는 1GiB~1TiB입니다. 자세한 내용은 범용 SSD(gp2) 볼륨 (p. 934) 단원을 참조하십시오.	2014년 6 월 16일
Windows Server 2012 R2		Windows Server 2012 R2용 AMI에 새 AWS PV 드라이버가 사용됩니다. 자세한 내용은 AWS PV 드라이버 (p. 524) 단원을 참조하십시오.	2014년 6 월 3일
AWS 관리 팩		이제 System Center Operations Manager 2012 R2에 대해 AWS 관리 팩이 지원됩니다. 자세한 내용은 AWS Management Pack for Microsoft System Center (p. 1244) 단원을 참조하십시오.	2014년 5 월 22일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
Amazon EBS 암호화	2014-05-01	Amazon EBS 암호화에서는 EBS 데이터 볼륨 및 스냅샷에 대한 완벽한 암호를 제공하므로 보안 키 관리 인프라를 구축하고 유지 관리할 필요가 없습니다. EBS 암호화는 Amazon 관리 키를 사용하여 데이터를 암호화하여 상주 데이터에 대한 보안을 활성화합니다. EC2 인스턴스를 호스트하는 서버에서 암호화가 이루어지기 때문에 EC2 인스턴스와 EBS 스토리지 간 이동하는 데이터도 암호화됩니다. 자세한 내용은 Amazon EBS 암호화 (p. 1033) 단원을 참조하십시오.	2014년 5 월 21일
R3 인스턴스	2014-02-01	경제적인 가격의 GiB RAM이 장착된 고성능 차세대 메모리 최적화 인스턴스. 이러한 인스턴스는 R3 인스턴스의 고급 네트워킹 기능, 우수한 컴퓨팅 성능 및 vCPU당 높은 메모리를 활용할 수 있는 관계형 및 NoSQL 데이터베이스, 메모리 내 분석 솔루션, 공학 계산 및 기타 메모리 집약형 애플리케이션에 매우 적합합니다. Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 Amazon EC2 인스턴스 유형 을 참조하십시오.	2014년 4 월 9일
Amazon EC2 사용 보고서		Amazon EC2 사용 보고서는 EC2 사용에 대한 비용 및 사용량 데이터를 보여 주는 보고서 세트입니다. 자세한 내용은 Amazon EC2 사용 보고서 (p. 1156) 단원을 참조하십시오.	2014년 1 월 28일
추가 M3 인스턴스	2013-10-15	이제 M3 인스턴스 크기 <code>m3.medium</code> 및 <code>m3.large</code> 가 지원됩니다. Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 Amazon EC2 인스턴스 유형 을 참조하십시오.	2014년 1 월 20일
I2 인스턴스	2013-10-15	이러한 인스턴스는 매우 높은 IOPS. 또한 I2 인스턴스는 인스턴스 간 지연 시간과 네트워크 지터를 낮추고 PPS(Packet Per Second) 성능을 크게 높이는 향상된 네트워킹 기능을 지원합니다. 자세한 내용은 스토리지 최적화 인스턴스 (p. 177) 단원을 참조하십시오.	2013년 12 월 19일
M3 인스턴스 업데이트	2013-10-15	M3 인스턴스 크기 <code>m3.xlarge</code> 및 <code>m3.2xlarge</code> 에서는 이제 SSD 볼륨이 장착된 인스턴스 스토어를 지원합니다.	2013년 12 월 19일
RunInstances에 대한 리소스 수준의 권한	2013-10-15	이제 AWS Identity and Access Management에서 정책을 생성하여 Amazon EC2 RunInstances API 작업에 대한 리소스 수준의 권한을 제어할 수 있습니다. 자세한 내용과 정책 예는 Amazon EC2의 자격 증명 및 액세스 관리 (p. 836) 단원을 참조하십시오.	2013년 11 월 20일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
C3 인스턴스	2013-10-15	<p>경제적인 가격으로 매우 우수한 성능을 제공하는 컴퓨팅에 최적화된 인스턴스입니다. 또한 C3 인스턴스는 인스턴스 간 지연 시간과 네트워크 지터를 낮추고 PPS(Packet Per Second) 성능을 크게 높여주는 향상된 네트워킹 기능을 지원합니다. 이러한 인스턴스는 트래픽 양이 높은 웹 애플리케이션, 광고 서비스, 배치성 프로세스, 비디오 인코딩, 분산 분석, 고에너지 물리학, 게임 분석 및 계산 유체 역학에 매우 적합합니다.</p> <p>Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 Amazon EC2 인스턴스 유형을 참조하십시오.</p>	2013년 11월 14일
AWS Marketplace에서 인스턴스 시작		<p>이제 AWS Marketplace 시작 마법사를 사용하여 Amazon EC2에서 인스턴스를 시작할 수 있습니다. 자세한 내용은 AWS Marketplace 인스턴스 시작 (p. 398) 단원을 참조하십시오.</p>	2013년 11월 11일
G2 인스턴스	2013-10-01	<p>이러한 인스턴스는 엄청난 병렬 처리 능력을 필요로 하는 비디오 제작 서비스, 3D 가상화, 스트리밍 그래픽 집약형 애플리케이션 및 기타 서버 쪽 작업에 매우 적합합니다. 자세한 내용은 Windows 액셀러레이티드 컴퓨팅 인스턴스 (p. 182) 단원을 참조하십시오.</p>	2013년 11월 4일
새로운 시작 마법사		<p>재설계된 새로운 EC2 시작 마법사가 제공됩니다. 자세한 내용은 인스턴스 시작 마법사를 사용하여 인스턴스 시작 (p. 376) 단원을 참조하십시오.</p>	2013년 10월 10일
Amazon EC2 예약 인스턴스 수정	2013-08-15	<p>이제 리전에서 예약 인스턴스를 수정할 수 있습니다.</p>	2013년 9월 11일
퍼블릭 IP 주소 배정	2013-07-15	<p>이제는 VPC에서 인스턴스를 시작할 때 퍼블릭 IP 주소를 배정할 수 있습니다. 자세한 내용은 인스턴스 시작 시 퍼블릭 IPv4 주소 할당 (p. 706) 단원을 참조하십시오.</p>	2013년 8월 20일
리소스 수준의 권한 부여	2013-06-15	<p>Amazon EC2에서는 새로운 Amazon 리소스 이름(ARN)과 조건 키를 지원합니다. 자세한 내용은 Amazon EC2에 대한 IAM 정책 (p. 838) 단원을 참조하십시오.</p>	2013년 7월 8일
증분형 스냅샷 사본	2013-02-01	<p>이제 증분형 스냅샷 사본을 사용할 수 있습니다. 자세한 내용은 Amazon EBS 스냅샷 복사 (p. 985) 단원을 참조하십시오.</p>	2013년 6월 11일
AWS 관리 팩		<p>AWS 관리 팩이 Amazon EC2 인스턴스와 인스턴스 내부에서 실행 중인 Windows 또는 Linux 운영 체제를 연결합니다. AWS 관리 팩은 Microsoft System Center Operations Manager의 확장 기능입니다. 자세한 내용은 AWS Management Pack for Microsoft System Center (p. 1244) 단원을 참조하십시오.</p>	2013년 5월 8일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
새 태그 페이지		Amazon EC2 콘솔에 새 태그 페이지가 있습니다. 자세한 내용은 Amazon EC2 리소스에 태그 지정 (p. 1143) 단원을 참조하십시오.	2013년 4 월 4일
추가 EBS에 최적화된 인스턴스 유형	2013-02-01	이제 인스턴스 유형 c1.xlarge, m2.2xlarge, m3.xlarge 및 m3.2xlarge를 EBS에 최적화된 인스턴스로 시작할 수 있습니다. 자세한 내용은 Amazon EBS-최적화 인스턴스 (p. 1049) 단원을 참조하십시오.	2013년 3 월 19일
PV 드라이버		Windows AMI에서 반가상화된(PV) 드라이버를 업그레이드하는 방법은 Windows 인스턴스에서 PV 드라이버 업그레이드 (p. 528) 단원을 참조하십시오.	2013년 3 월
리전 간 AMI 복사	2013-02-01	리전 간에 AMI를 복사하여 둘 이상의 AWS 리전에서 일관된 인스턴스를 빠르고 쉽게 시작할 수 있습니다. 자세한 내용은 AMI 복사 (p. 103) 단원을 참조하십시오.	2013년 3 월 11일
인스턴스를 기본 VPC로 시작	2013-02-01	AWS 계정에서는 리전별로 EC2-Classic 또는 VPC에서, 또는 VPC에서만 인스턴스를 시작할 수 있습니다. VPC로만 인스턴스를 시작할 수 있는 경우 사용자를 위한 기본 VPC가 생성됩니다. 사용자가 기본이 아닌 VPC를 직접 생성하여 인스턴스 시작 시 지정한 경우가 아니면 인스턴스 시작 시 해당 인스턴스가 기본 VPC로 시작됩니다.	2013년 3 월 11일
고용량 메모리 클러스터 (cr1.8xlarge) 인스턴스 유형	2012-12-01	대용량 메모리가 고성능 CPU 및 네트워크 성능과 결합되었습니다. 이러한 인스턴스는 메모리 내 분석, 그래프 분석 및 공학 컴퓨팅 애플리케이션에 매우 적합합니다.	2013년 1 월 21일
고용량 스토리지 (hs1.8xlarge) 인스턴스 유형	2012-12-01	고용량 스토리지 인스턴스는 인스턴스당 매우 높은 스토리지 밀도와 높은 순차 읽기/쓰기 성능을 제공합니다. 이러한 인스턴스는 데이터 웨어하우스, 하둡/MapReduce 및 병렬 파일 시스템에 매우 적합합니다.	2012년 12 월 20일
EBS 스냅샷 복사	2012-12-01	스냅샷 사본으로 데이터 백업, 새 Amazon EBS 볼륨 또는 Amazon 머신 이미지(AMI)를 생성할 수 있습니다. 자세한 내용은 Amazon EBS 스냅샷 복사 (p. 985) 단원을 참조하십시오.	2012년 12 월 17일
프로비저닝된 IOPS SSD 볼륨에 대한 EBS 지표 및 상태 확인 업데이트	2012-10-01	EBS 지표를 업데이트하여 프로비저닝된 IOPS SSD 볼륨용 새 지표 두 개를 포함했습니다. 자세한 내용은 Amazon EBS에 대한 Amazon CloudWatch 지표 (p. 1076) 단원을 참조하십시오. 또한 프로비저닝된 IOPS SSD 볼륨에 대한 새로운 상태 확인도 추가했습니다. 자세한 내용은 EBS 볼륨 상태 확인 (p. 958) 단원을 참조하십시오.	2012년 11 월 20일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
Windows Server 2012 지원		<p>이제 Amazon EC2에서는 사전 구성된 여러 Windows Server 2012 AMI를 제공합니다. 이러한 AMI는 모든 리전과 모든 64비트 인스턴스 유형에서 즉시 사용할 수 있습니다. AMI는 다음 언어를 지원합니다.</p> <ul style="list-style-type: none"> • 영어 • 중국어 간체 • 중국어 번체 • 중국어 번체(총공) • 일본어 • 한국어 • 포르투갈어 • 포르투갈어(브라질) • 체코어 • 네덜란드어 • 프랑스어 • 독일어 • 헝가리어 • 이탈리아어 • 폴란드어 • 러시아어 • 스페인어 • 스웨덴어 • 터키어 	2012년 11 월 19일
M3 인스턴스	2012-10-01	새로운 M3 extra-large 및 M3 double-extra-large 인스턴스 유형을 추가했습니다. Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 Amazon EC2 인스턴스 유형 을 참조하십시오.	2012년 10 월 31일
스팟 인스턴스 요청 상태	2012-10-01	스팟 인스턴스 요청 상태를 사용하면 스팟 요청의 상태를 쉽게 확인할 수 있습니다.	2012년 10 월 14일
Amazon EC2 예약 인스턴스 마켓플레이스	2012-08-15	예약 인스턴스 마켓플레이스은 더 이상 필요하지 않은 Amazon EC2 예약 인스턴스를 보유한 판매자와 추가 용량을 원하는 구매자를 연결합니다. 예약 인스턴스 마켓플레이스을 통해 구매 및 판매되는 예약 인스턴스는 다른 예약 인스턴스와 동일하게 작동합니다. 단, 이러한 인스턴스는 스탠다드 약정보다 사용 기간이 짧을 수 있으며 다른 가격으로 판매할 수 있습니다.	2012년 9 월 11일
Amazon EBS용 프로비저닝된 IOPS SSD	2012-07-20	프로비저닝된 IOPS SSD 볼륨은 일관적이고 빠른 응답 시간을 이용하는 데이터베이스 애플리케이션처럼 I/O 집약적 작업을 위한 예측 가능하고 우수한 성능을 제공합니다. 자세한 내용은 Amazon EBS 볼륨 유형 (p. 932) 단원을 참조하십시오.	2012년 7 월 31일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
Amazon EC2용 고성능 I/O 인스턴스	2012-06-15	고성능 I/O 인스턴스는 SSD 기반 로컬 인스턴스 스토리지를 사용하여 지연 시간을 줄이고 디스크 I/O 성능을 높입니다.	2012년 7 월 18일
Amazon EC2 인스턴스의 IAM 역할	2012-06-01	<p>Amazon EC2의 IAM 역할은 다음을 제공합니다.</p> <ul style="list-style-type: none"> Amazon EC2 인스턴스에서 실행 중인 애플리케이션의 AWS 액세스 키 Amazon EC2 인스턴스에서 AWS 액세스 키 자동 순환 Amazon EC2 인스턴스에서 실행 중이며 AWS 서비스에 요청하는 애플리케이션에 대한 세분화된 사용 권한 	2012년 6 월 11일
더 쉽게 시작하고 중단 가능성을 처리할 수 있게 하는 스팟 인스턴스 기능		<p>이제 다음과 같이 스팟 인스턴스를 관리할 수 있습니다.</p> <ul style="list-style-type: none"> Auto Scaling 시작 구성을 사용하여 스팟 인스턴스에 대해 지불할 금액을 지정하고 스팟 인스턴스에 대해 지불할 금액을 지정하기 위한 일정을 설정합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Auto Scaling 그룹에서 스팟 인스턴스 시작을 참조하십시오. 인스턴스가 시작되거나 종료될 때 알림을 받습니다. AWS CloudFormation 템플릿을 사용하여 AWS 리소스를 포함하는 스택에서 스팟 인스턴스를 시작합니다. 	2012년 6 월 7일
EC2 인스턴스 내보내기와 Amazon EC2 상태 확인을 위한 타임스탬프	2012-05-01	<p>원래 EC2로 가져왔던 Windows Server 인스턴스에 대한 내보내기 지원을 추가했습니다.</p> <p>상태 확인이 실패한 날짜 및 시간을 나타내는 인스턴스 상태 및 시스템 상태의 타임스탬프에 대한 지원을 추가했습니다.</p>	2012년 5 월 25일
EC2 인스턴스 내보내기와, Amazon VPC에 대한 인스턴스 및 시스템 상태 확인 시 타임스탬프	2012-05-01	<p>Citrix Xen, Microsoft Hyper-V 및 VMware vSphere로 EC2 인스턴스 내보내기 지원을 추가했습니다.</p> <p>인스턴스 및 시스템 상태 확인 시 타임스탬프 지원을 추가했습니다.</p>	2012년 5 월 25일
클러스터 컴퓨팅 에이트 엑스트라 라지 인스턴스	2012-04-01	VPC에서 cc2.8xlarge 인스턴스에 대한 지원을 추가했습니다.	2012년 4 월 26일
AWS Marketplace AMI	2012-04-01	AWS Marketplace AMI 지원을 추가했습니다.	2012년 4 월 19일
미디엄 인스턴스, 모든 AMI	2011-12-15	새로운 인스턴스 유형에 대한 지원과 64비트 정보를 추가했습니다.	2012년 3 월 7일
예약된 인스턴스 요금 계층	2011-12-15	예약 인스턴스 요금 계층에 기본 제공되는 할인 요금을 활용하는 방법을 설명하는 새로운 섹션을 추가했습니다.	2012년 3 월 5일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
Amazon Virtual Private Cloud의 EC2 인스턴스 용 ENI	2011-12-01	VPC의 EC2 인스턴스용 ENI(탄력적 네트워크 인터페이스)에 대한 새로운 섹션을 추가했습니다. 자세한 내용은 탄력적 네트워크 인터페이스 (p. 729) 단원을 참조하십시오.	2011년 12 월 21일
Amazon EC2 예약 인스턴스를 위한 새로운 제공 유형	2011-11-01	예상되는 인스턴스 사용을 처리하는 다양한 예약 인스턴스 상품 중에서 선택할 수 있습니다.	2011년 12 월 1일
Amazon EC2 인스턴스 상태	2011-11-01	인스턴스에 영향을 줄 수 있는 AWS에서 계획한 예약 이벤트를 포함하여 인스턴스의 상태에 대한 추가 세부 정보를 볼 수 있습니다. 이러한 운영 활동에는 보안 패치나 소프트웨어 업데이트를 적용하는 데 필요한 인스턴스 재부팅이나 하드웨어 문제가 있는 경우에 필요한 인스턴스 증지가 포함됩니다. 자세한 내용은 인스턴스 상태 모니터링 (p. 651) 단원을 참조하십시오.	2011년 11 월 16일
Amazon EC2 클러스터 컴퓨팅 인스턴스 유형		Amazon EC2에 클러스터 컴퓨팅 에이트 엑스트라지(cc2.8xlarge) 지원을 추가했습니다.	2011년 11 월 14일
Amazon VPC의 스팟 인스턴스	2011-07-15	Amazon VPC의 스팟 인스턴스 지원에 대한 정보를 추가했습니다. 이 업데이트로 사용자가 Virtual Private Cloud(VPC)에서 스팟 인스턴스를 시작할 수 있습니다. 스팟 인스턴스 사용자가 VPC에서 스팟 인스턴스를 시작하면 Amazon VPC의 혜택을 누릴 수 있습니다.	2011년 10 월 11일
CLI 도구 사용자를 위한 간소화된 VM Import 프로세스	2011-07-15	이제 <code>ImportInstance</code> 및 <code>ImportVolume</code> 기능의 향상으로 VM Import 프로세스가 간소화되어 가져오기 작업을 생성한 후 이미지가 Amazon EC2로 업로드됩니다. 또한 <code>ResumeImport</code> 가 도입되면서 사용자가 완료되지 않은 업로드를 작업이 중지된 시점부터 다시 시작할 수 있습니다.	2011년 9 월 15일
VHD 파일 형식으로 가져오기 지원		이제 VM Import에서 가상 머신 이미지 파일을 VHD 형식으로 가져올 수 있습니다. VHD 파일 형식은 Citrix Xen 및 Microsoft Hyper-V 가상화 플랫폼과 호환됩니다. 이번 릴리스에 포함된 VM Import 기능에서는 이제 RAW, VHD 및 VMDK(VMware ESX 호환) 이미지 형식을 지원합니다. 자세한 내용은 VM Import/Export 사용 설명서 단원을 참조하십시오.	2011년 8 월 24일
Windows Server 2003 R2 지원		VM Import에서 이제 Windows Server 2003(R2)을 지원합니다. 이번 릴리스에 포함된 VM Import 기능은 Amazon EC2에서 지원하는 모든 Windows Server 버전을 지원합니다.	2011년 8 월 24일
VMware vCenter용 Amazon EC2 VM Import 커넥터 업데이트		VMware vCenter 가상 어플라이언스용 Amazon EC2 VM Import 커넥터 버전 1.1(커넥터)에 대한 정보를 추가했습니다. 이 업데이트에는 인터넷 액세스에 대한 프록시 지원, 오류 처리 개선, 작업 진행률 표시줄 정확도 향상 및 여러 버그 수정 사항이 포함되어 있습니다.	2011년 6 월 27일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
스팟 인스턴스 가용 영역 요금 변경	2011-05-15	스팟 인스턴스 가용 영역 요금 기능에 대한 정보를 추가했습니다. 이 릴리스에서는 스팟 인스턴스 요청과 스팟 가격 기록을 쿼리할 때 반환되는 정보의 일부로서 새 가용 영역 요금 옵션을 추가했습니다. 이러한 추가를 통해 스팟 인스턴스를 특정 가용 영역으로 시작하는 데 필요한 가격을 보다 쉽게 확인할 수 있습니다.	2011년 5 월 26일
AWS Identity and Access Management		AWS Identity and Access Management(IAM)에 대한 정보를 추가했습니다. 사용자는 IAM을 통해 일반적으로 Amazon EC2 리소스와 함께 사용할 수 있는 Amazon EC2 작업을 지정할 수 있습니다. 자세한 내용은 Amazon EC2의 자격 증명 및 액세스 관리 (p. 836) 단원을 참조하십시오.	2011년 4 월 26일
전용 인스턴스		Amazon Virtual Private Cloud(Amazon VPC) 내에서 시작되는 전용 인스턴스는 호스트 하드웨어 수준에서 물리적으로 구분되어 있는 인스턴스입니다. 전용 인스턴스에서는 탄력적인 온디맨드 프로비저닝을 포함한 다양한 혜택과 함께 Amazon VPC와 AWS 클라우드를 활용하고 사용하는 서비스에 대해서만 요금을 지불할 수 있으며, 하드웨어 수준에서 Amazon EC2 컴퓨팅 인스턴스를 구분할 수 있습니다. 자세한 내용은 전용 인스턴스 (p. 349) 단원을 참조하십시오.	2011년 3 월 27일
AWS Management Console에 예약 인스턴스 업데이트		AWS Management Console 업데이트로 사용자는 더욱 더 쉽게 추가 예약 인스턴스를 보고 전용 예약 인스턴스를 비롯한 추가 예약 인스턴스를 구매할 수 있습니다.	2011년 3 월 27일
Windows Server 2008 R2 지원		이제 Amazon EC2에서는 사전 구성된 여러 Windows Server 2008 R2 AMI를 제공합니다. 이러한 AMI는 t1.micro 및 HPC 패밀리를 제외하고는 대부분의 64비트 인스턴스 유형과 모든 리전에서 즉시 사용할 수 있습니다. AMI에서는 여러 언어를 지원합니다.	2011년 3 월 15일
메타데이터 정보	2011-01-01	2011년 1월 1일 릴리스의 변경 내용을 반영하여 메타데이터에 대한 정보를 추가했습니다. 자세한 내용은 인스턴스 메타데이터 및 사용자 데이터 (p. 576) 및 인스턴스 메타데이터 카테고리 (p. 591) 단원을 참조하십시오.	2011년 3 월 11일
VMware vCenter용 Amazon EC2 VM Import 커넥터		VMware vCenter 가상 어플라이언스용 Amazon EC2 VM Import 커넥터(커넥터)에 대한 정보를 추가했습니다. 이 커넥터는 VMware vSphere Client가 통합된 VMware vCenter용 플러그 인으로 VMware 가상 머신을 Amazon EC2로 가져오는 데 사용할 수 있는 그래픽 사용자 인터페이스를 제공합니다.	2011년 3 월 3일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
강제 볼륨 분리		이제 AWS Management Console을 사용하여 Amazon EBS 볼륨을 인스턴스에서 강제 분리할 수 있습니다. 자세한 내용은 Windows 인스턴스에서 Amazon EBS 볼륨 분리 (p. 964) 단원을 참조하십시오.	2011년 2 월 23일
인스턴스 종료 방지		이제 AWS Management Console을 사용하여 인스턴스가 종료되는 것을 방지할 수 있습니다. 자세한 내용은 종료 방지 기능 활성화 (p. 456) 단원을 참조하십시오.	2011년 2 월 23일
VM Import	2010-11-15	VM Import에 대한 정보를 추가했습니다. VM Import 기능을 사용하면 가상 머신이나 볼륨을 Amazon EC2로 가져올 수 있습니다. 자세한 내용은 VM Import/Export 사용 설명서 단원을 참조하십시오.	2010년 12 월 15일
인스턴스 기본 모니터링	2010-08-31	EC2 인스턴스 기본 모니터링에 대한 정보를 추가했습니다.	2010년 12 월 12일
필터와 태그	2010-08-31	리소스 목록, 필터링 및 태그에 대한 정보를 추가했습니다. 자세한 내용은 리소스 목록화 및 필터링 (p. 1138) 및 Amazon EC2 리소스에 태그 지정 (p. 1143) 단원을 참조하십시오.	2010년 9 월 19일
멱등성 인스턴스 시작	2010-08-31	인스턴스 실행 시 멱등성 유지에 대한 정보를 추가했습니다.	2010년 9 월 19일
マイ크로 인스턴스	2010-06-15	Amazon EC2는 특정 애플리케이션 유형을 위한 t1.micro 인스턴스 유형을 제공합니다. 자세한 내용은 다음을 참조하십시오. 성능 순간 확장 가능 인스턴스 (p. 129) .	2010년 9 월 8일
Amazon EC2용 AWS Identity and Access Management		Amazon EC2가 이제 AWS Identity and Access Management(IAM)와 통합되었습니다. 자세한 내용은 Amazon EC2의 자격 증명 및 액세스 관리 (p. 836) 단원을 참조하십시오.	2010년 9 월 2일
클러스터 인스턴스	2010-06-15	Amazon EC2는 고성능 컴퓨팅(HPC) 애플리케이션을 위한 클러스터 컴퓨팅 인스턴스를 제공합니다. Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 Amazon EC2 인스턴스 유형 을 참조하십시오.	2010년 7 월 12일
Amazon VPC IP 주소 지정	2010-06-15	Amazon VPC 사용자는 이제 IP 주소를 지정하여 VPC에서 시작된 인스턴스를 배정할 수 있습니다.	2010년 7 월 12일
Amazon EBS 볼륨에 대한 Amazon CloudWatch 모니터링		이제 Amazon EBS 볼륨에 대해 Amazon CloudWatch 모니터링을 자동으로 사용할 수 있습니다. 자세한 내용은 Amazon EBS에 대한 Amazon CloudWatch 지표 (p. 1076) 단원을 참조하십시오.	2010년 6 월 14일
고용량 메모리 엑스트라 라지 인스턴스	2009-11-30	이제 Amazon EC2에서는 고용량 메모리 엑스트라 라지 (m2.xlarge) 인스턴스 유형을 지원합니다. Amazon EC2 인스턴스 유형별 하드웨어 사양에 대한 자세한 내용은 Amazon EC2 인스턴스 유형 을 참조하십시오.	2010년 2 월 22일

Amazon Elastic Compute Cloud
Windows Instances용 사용 설명서
지난 몇 년의 기록

기능	API 버전	설명	릴리스 날짜
Windows용 예약 인스턴스		이제 Amazon EC2에서 Windows용 예약 인스턴스를 지원합니다.	2010년 2월 22일