
Amazon EC2 Auto Scaling

사용 설명서



Amazon EC2 Auto Scaling: 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 EC2 Auto Scaling이란 무엇입니까?	1
Auto Scaling 구성 요소	1
시작하기	2
Amazon EC2 Auto Scaling에 액세스	2
Amazon EC2 Auto Scaling 가격	2
PCI DSS 준수	2
관련 서비스	3
Auto Scaling의 이점	3
예: 가변적인 수요에 대응	3
예: 웹 앱 아키텍처	5
예: 가용 영역 전반에 인스턴스 분산	6
Auto Scaling 수명 주기	7
확장	7
서비스 상태의 인스턴스	8
축소	8
인스턴스 연결	8
인스턴스 분리	8
수명 주기 후크	9
대기 모드 시작 및 종료	9
Service Quotas	9
설정	11
AWS에 가입	11
Amazon EC2 사용 준비	11
시작하기	12
1단계: 시작 템플릿 생성	12
2단계: Auto Scaling 그룹 생성	14
3단계: Auto Scaling 그룹 확인	16
(선택사항) Auto Scaling 그룹에서 인스턴스 종료	17
4단계: 다음 단계	17
5단계: (선택 사항) 조정 인프라 삭제	17
자습서: 조정 및 로드 밸런싱된 애플리케이션 설정	19
사전 조건	19
애플리케이션 배포(콘솔)	20
시작 템플릿 생성 또는 선택	20
시작 구성 생성 또는 선택	21
Auto Scaling 그룹 생성	22
(선택 사항) 로드 밸런서가 Auto Scaling 그룹에 연결되었는지 확인합니다.	23
애플리케이션 배포(AWS CLI)	24
시작 템플릿 생성	24
시작 구성 생성	24
로드 밸런서가 있는 Auto Scaling 그룹 생성	25
다음 단계	25
AWS 리소스 정리	25
시작 템플릿	26
Auto Scaling 그룹을 위한 시작 템플릿 만들기	26
시작 구성을 시작 템플릿에 복사	32
시작 구성을 시작 템플릿으로 교체	32
시작 구성	34
시작 구성 만들기	34
EC2 인스턴스를 사용하여 시작 구성 만들기	35
EC2 인스턴스를 사용하여 시작 구성 만들기	36
인스턴스에서 시작 구성을 만들고 블록 디바이스 재정의(AWS CLI)	37
시작 구성 만들기 및 인스턴스 유형 재정의(AWS CLI)	38
시작 구성 변경	39

VPC의 인스턴스 시작	40
기본 VPC	40
VPC의 IP 어드레싱	40
인스턴스 배치 태닝시	41
VPC에 대한 EC2-Classic 인스턴스 링크 설정	42
Auto Scaling 그룹	44
여러 인스턴스 유형 및 구매 옵션 사용	45
할당 전략	45
온디맨드 인스턴스 비율 제어	46
스팟 인스턴스 모범 사례	47
사전 조건	48
여러 구매 옵션으로 Auto Scaling 그룹 생성	48
시작 템플릿을 사용하여 그룹 생성하기	54
시작 구성을 사용하여 그룹 생성하기	57
EC2 인스턴스를 사용하여 그룹 생성하기	59
EC2 인스턴스에서 Auto Scaling 그룹 생성(콘솔)	60
EC2 인스턴스에서 Auto Scaling 그룹 생성(AWS CLI)	60
시작 마법사를 사용하여 그룹 생성하기	61
Auto Scaling 그룹 및 인스턴스 태그 지정	62
태그 제한	62
태그 지정 수명 주기	63
Auto Scaling 그룹에 태그 추가 및 태그 수정	63
태그 삭제	65
Auto Scaling 그룹에 로드 밸런서 사용	66
Elastic Load Balancing 유형	66
로드 밸런서 연결	67
ELB 상태 확인 추가	70
가용 영역 추가	71
Auto Scaling 그룹에서 스팟 인스턴스 시작	73
인스턴스 가중치 사용	74
단위 시간당 가격	74
고려 사항	75
Auto Scaling 그룹의 가중치 추가 또는 수정	76
인스턴스 유형 권장 사항	79
제한 사항	80
결과	80
권장 사항 보기	80
권장 사항 평가를 위한 고려 사항	81
최대 인스턴스 수명을 기준으로 인스턴스 교체	81
Auto Scaling 그룹 병합	83
영역 병합(AWS CLI)	83
Auto Scaling 인프라 삭제	85
Auto Scaling 그룹 삭제	85
(선택 사항) 시작 구성 삭제	85
(선택 사항) 시작 템플릿 삭제	86
(선택 사항) 로드 밸런서 삭제	86
(선택 사항) CloudWatch 경보 삭제	87
그룹 조정	88
조정 옵션	88
용량 제한 설정	89
일정한 수의 인스턴스 유지	90
수동 조정	90
Auto Scaling 그룹의 크기 변경(콘솔)	90
Auto Scaling 그룹의 크기 변경(AWS CLI)	91
Auto Scaling 그룹에 EC2 인스턴스 연결	92
Auto Scaling 그룹에서 EC2 인스턴스 분리	96
예약된 조정	99

고려 사항	99
예약된 작업 생성 및 관리(콘솔)	100
예약된 작업 생성 및 관리(AWS CLI)	101
동적 조정	102
조정 정책 작동 방식	102
조정 정책 유형	103
여러 조정 정책	103
대상 추적 조정 정책	104
단계 및 단순 조정 정책	107
Amazon SQS에 따른 조정	115
조정 활동 확인	119
조정 정책 비활성화	121
조정 정책 삭제	122
조정 휴지	123
예: 휴지	124
기본 휴지	125
조정 특정 휴지	125
휴지 및 다중 인스턴스	125
휴지 및 수명 주기 후크	125
Auto Scaling 인스턴스 종료	126
기본 종료 정책	126
종료 정책 사용자 지정	127
인스턴스 축소 보호	128
수명 주기 후크	131
수명 주기 후크 작동 방식	131
고려 사항	132
알림 준비	133
수명 주기 후크 추가	133
수명 주기 후크 사용자 지정 작업 완료	134
알림 테스트	135
수명 주기 후크 알림 구성	135
일시적으로 인스턴스 제거	138
대기 상태를 작동하는 방법	139
대기 상태의 인스턴스 상태	140
인스턴스를 일시적으로 제거(콘솔)	140
인스턴스를 일시적으로 제거(AWS CLI)	140
조정 일시 중지	142
조정 프로세스	143
일시 중지 선택	143
조정 프로세스 일시 중지 및 재개(콘솔)	145
조정 프로세스 일시 중지 및 재개(AWS CLI)	146
Auto Scaling 인스턴스 및 그룹 모니터링	147
상태 확인	148
인스턴스 상태	148
인스턴스 상태 확인	148
상태 확인 유예 기간	149
비정상 인스턴스 교체	149
사용자 지정 상태 확인	149
Amazon SNS 알림	150
SNS 알림	151
Amazon SNS 구성	151
알림을 보내도록 Auto Scaling 그룹 구성	152
알림 구성 테스트	153
조정 이벤트 알림 수신 확인	153
알림 구성 삭제	154
Amazon CloudWatch Events	155
Auto Scaling 이벤트	155

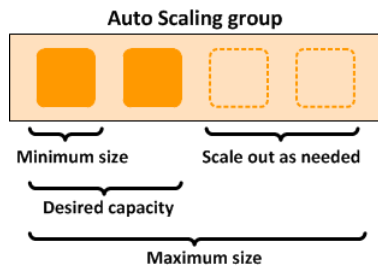
Lambda 함수 만들기	159
Lambda 함수로 이벤트 라우팅	159
Amazon CloudWatch 지표	160
Auto Scaling 그룹 지표	161
Auto Scaling 그룹 지표의 차원	162
Auto Scaling 그룹 지표 활성화	162
Auto Scaling 인스턴스에 대한 모니터링 구성	163
CloudWatch 지표 보기	164
Amazon CloudWatch 경보 만들기	165
AWS CloudTrail 로깅	166
CloudTrail의 Amazon EC2 Auto Scaling 정보	166
Amazon EC2 Auto Scaling 로그 파일 항목 이해	167
Personal Health Dashboard	168
보안	170
데이터 보호	170
AWS KMS를 사용하여 데이터 암호화	171
ID 및 액세스 관리	171
액세스 제어	171
Amazon EC2 Auto Scaling에서 IAM을 사용하는 방법	172
서비스 연결 역할	176
자격 증명 기반 정책 예제	179
시작 템플릿 지원	188
Amazon EC2 인스턴스에서 실행되는 애플리케이션의 IAM 역할	192
암호화된 볼륨에 사용할 경우 필요한 CMK 키 정책	193
규정 준수 확인	196
복원성	197
인프라 보안	197
VPC 엔드포인트를 사용한 프라이빗 연결	198
인터페이스 VPC 엔드포인트 생성	198
VPC 엔드포인트 정책 생성	198
문제 해결	200
오류 메시지 검색	200
인스턴스 시작 실패	202
보안 그룹 <보안 그룹 이름>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다.	203
키 페어 <EC2 인스턴스와 연결된 키 페어>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다.	203
요청된 구성이 현재 지원되지 않습니다.	203
AutoScalingGroup <Auto Scaling 그룹 이름>을(를) 찾을 수 없습니다.	203
요청된 가용 영역이 더 이상 지원되지 않습니다. 가용 영역을 지정하지 않거나...	203
요청된 인스턴스 유형(<인스턴스 유형>)이 요청된 가용 영역(<인스턴스 가용 영역>)에서 지원되지 않습니다...	204
이 서비스를 구독하고 있지 않습니다. https://aws.amazon.com/ 단원을 참조하십시오.	204
잘못된 디바이스 이름이 업로드되었습니다. EC2 인스턴스 시작에 실패했습니다.	204
파라미터 virtualName에 대한 값(<인스턴스 스토리지 디바이스와 연결된 이름>)이 잘못되었습니다...	204
EBS 블록 디바이스 매핑이 인스턴스 스토어 AMI에 대해 지원되지 않습니다.	205
배치 그룹은 'm1.large' 유형의 인스턴스와 함께 사용할 수 없습니다. EC2 인스턴스 시작에 실패했습니다.	205
Client.InternalError: 시작 시 클라이언트 오류.	205
AMI 문제	206
AMI ID <AMI ID>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다.	206
AMI <AMI ID>이(가) 보류 중이며 실행할 수 없습니다. EC2 인스턴스 시작에 실패했습니다.	207
파라미터 virtualName의 값(<ami ID>)이 잘못되었습니다.	207
요청된 인스턴스 유형의 아키텍처(i386)가 ami-6622f00f(x86_64)용 매니페스트의 아키텍처와 일치하지 않습니다. EC2 인스턴스 시작에 실패했습니다.	207
로드 밸런서 문제	207
로드 밸런서 <시작 환경>을 찾을 수 없습니다. 로드 밸런서 구성의 유효성 검사에 실패했습니다.	208

이름이 <로드 밸런서 이름>인 활성 로드 밸런서가 없습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다.	208
EC2 인스턴스 <인스턴스 ID>이(가) VPC에 없습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다.	208
EC2 인스턴스 <인스턴스 ID>이(가) VPC에 있습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다.	208
요청에 포함된 보안 토큰이 잘못되었습니다. 로드 밸런서 구성의 유효성 검사에 실패했습니다.	209
용량 제한	209
현재 요청한 가용 영역(<요청한 가용 영역>)에 충분한 <인스턴스 유형> 용량을 갖고 있지 않습니다....	209
<인스턴스 수>개의 인스턴스가 이미 실행 중입니다. EC2 인스턴스 시작에 실패했습니다.	209
리소스	210
문서 기록	211

Amazon EC2 Auto Scaling이란 무엇입니까?

Amazon EC2 Auto Scaling를 통해 애플리케이션의 로드를 처리할 수 있는 정확한 수의 Amazon EC2 인스턴스를 보유하도록 보장할 수 있습니다. Auto Scaling 그룹이라는 EC2 인스턴스 모음을 생성합니다. 각 Auto Scaling 그룹의 최소 인스턴스 수를 지정할 수 있으며, Amazon EC2 Auto Scaling에서는 그룹의 크기가 이 값 아래로 내려가지 않습니다. 각 Auto Scaling 그룹의 최대 인스턴스 수를 지정할 수 있으며, Amazon EC2 Auto Scaling에서는 그룹의 크기가 이 값을 넘지 않습니다. 원하는 용량을 지정한 경우 그룹을 생성한 다음에는 언제든지 Amazon EC2 Auto Scaling에서 해당 그룹에서 이만큼의 인스턴스를 보유할 수 있습니다. 조정 정책을 지정했다면 Amazon EC2 Auto Scaling에서는 애플리케이션의 늘어나거나 줄어드는 수요에 따라 인스턴스를 시작하거나 종료할 수 있습니다.

예를 들어, 다음 Auto Scaling 그룹의 경우 최소 인스턴스 수 1개, 희망 인스턴스 용량 2개, 최대 인스턴스 수 4개가 됩니다. 사용자가 정의한 조정 정책에 따라 인스턴스 수가 최소 및 최대 인스턴스 수 내에서 지정하는 조건에 따라 조절됩니다.



Amazon EC2 Auto Scaling의 이점에 대한 자세한 내용은 [Auto Scaling의 이점 \(p. 3\)](#)을 참조하십시오.

Auto Scaling 구성 요소

다음 표는 Amazon EC2 Auto Scaling의 핵심 구성 요소가 나와 있습니다.

	<p>Groups</p> <p>EC2 인스턴스는 그룹에 정리되어 조정 및 관리 목적의 논리적 단위로 처리할 수 있습니다. 그룹을 생성할 때 EC2 인스턴스의 최소 및 최대 인스턴스 수와 원하는 인스턴스 수를 지정할 수 있습니다. 자세한 내용은 Auto Scaling 그룹 (p. 44) 단원을 참조하십시오.</p>
	<p>구성 템플릿</p> <p>그룹은 시작 템플릿 또는 시작 구성을 EC2 인스턴스에 대한 구성 템플릿으로 사용합니다. 인스턴스의 AMI ID, 인스턴스 유형, 키 페어, 보안 그룹, 블록 디바이스 매핑 등의 정보를 지정할 수 있습니다. 자세한 내용은 시작 템플릿 (p. 26) 및 시작 구성 (p. 34) 단원을 참조하십시오.</p>
	<p>조정 옵션</p> <p>Amazon EC2 Auto Scaling은 Auto Scaling 그룹을 조정하는 다양한 방법을 제공합니다. 예를 들어, 지정한 조건의 발생(동적 확장) 또는 일정 에 따라 조정하도록 그룹을 구성할 수 있습니다. 자세한 내용은 조정 옵션 (p. 88) 단원을 참조하십시오.</p>

시작하기

Amazon EC2 Auto Scaling을 처음 이용한다면 시작하기 전에 [Auto Scaling 수명 주기](#) (p. 7)를 검토하는 것이 좋습니다.

먼저 [Amazon EC2 Auto Scaling 시작하기](#) (p. 12) 자습서를 완료하여 Auto Scaling 그룹을 하나 생성한 다음 해당 그룹에서 인스턴스가 종료될 때 어떻게 응답하는지 확인합니다. 실행 중인 EC2 인스턴스가 이미 있는 경우, 기존 EC2 인스턴스를 사용하여 Auto Scaling 그룹을 생성하고 이 인스턴스를 언제든지 그룹에서 제거할 수 있습니다.

Amazon EC2 Auto Scaling에 액세스

AWS 계정을 등록한 경우에는 AWS Management 콘솔에 로그인하고, 콘솔 홈 페이지에서 EC2를 선택한 다음, 탐색 창에서 Auto Scaling 그룹을 선택하여 Amazon EC2 Auto Scaling에 액세스할 수 있습니다.

[Amazon EC2 Auto Scaling API](#)를 사용하여 Amazon EC2 Auto Scaling에 액세스할 수도 있습니다. Amazon EC2 Auto Scaling은 쿼리 API를 제공합니다. 이 리퀘스트들은, HTTP나 HTTPS의 메시지 교환 방식인 GET이나 POST이며, 미리 정해진 이름인 "Action"을 쿼리 변수로 사용합니다. Amazon EC2 Auto Scaling에 관련된 API 작업에 대한 자세한 정보는 [작업](#)(Amazon EC2 Auto Scaling API 참조)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 정보는 [AWS SDK 및 도구](#) 단원을 참조하십시오.

명령줄 인터페이스를 선호하는 고객의 경우 다음과 같은 옵션이 있습니다.

AWS명령줄 인터페이스(CLI)

다양한 AWS 제품에서 사용되는 명령어를 제공하며 Windows, macOS, Linux를 지원합니다. 시작하려면 [AWS Command Line Interface 사용 설명서](#) 단원을 참조하십시오. 자세한 내용은 AWS CLI Command Reference의 [autoscaling\(자동 조정\)](#) 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. 자세한 내용은 [PowerShell용 AWS 도구 Cmdlet Reference](#)를 참조하십시오.

AWS 액세스 자격 증명에 대한 자세한 내용은 Amazon Web Services 일반 참조에서 [AWS 보안 자격 증명 정보](#)를 참조하십시오. Amazon EC2 Auto Scaling 호출용 엔드포인트와 리전에 대한 자세한 내용은 AWS General Reference의 [Amazon EC2 Auto Scaling 리전 및 엔드포인트](#)를 참조하십시오.

Amazon EC2 Auto Scaling 가격

Amazon EC2 Auto Scaling에 대한 추가 비용은 없으며, 따라서 우선 사용해 보고 AWS 아키텍처가 주는 이점에 대해 알아볼 수 있습니다.

PCI DSS 준수

Auto Scaling에서는 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.

관련 서비스

애플리케이션에 대해 모든 확장 가능한 AWS 리소스의 자동 조정을 구성하려면 AWS Auto Scaling을 사용합니다. AWS Auto Scaling를 통해 Auto Scaling 그룹의 동적 조정 정책 정의 프로세스를 간소화하고 Predictive Scaling을 사용하여 예측된 트래픽 변경에 앞서 Amazon EC2 용량을 확장할 수 있습니다. 자세한 내용은 [AWS Auto Scaling 사용 설명서](#)를 참조하십시오.

Elastic Load Balancing은 애플리케이션의 수신 트래픽을 Auto Scaling 그룹의 여러 인스턴스로 자동으로 분산해 줍니다. 자세한 내용은 [Elastic Load Balancing 사용 설명서](#)를 참조하십시오.

인스턴스와 Amazon EBS 볼륨에 관련된 기본 통계 정보를 모니터링하려면 Amazon CloudWatch를 사용합니다. 자세한 내용은 [Amazon CloudWatch 사용 설명서](#)를 참조하십시오.

계정에 대해 이루어진 Amazon EC2 Auto Scaling API 호출을 모니터링하려면 AWS CloudTrail을 사용하십시오. 로깅된 데이터에는 AWS Management 콘솔, 명령줄 도구 및 기타 서비스가 수행한 호출이 포함됩니다. 자세한 정보는 [AWS CloudTrail User Guide](#)를 참조하십시오.

Auto Scaling의 이점

Amazon EC2 Auto Scaling를 애플리케이션 아키텍처에 추가하는 것은 AWS 클라우드의 이점을 극대화할 수 있는 한 방법입니다. Amazon EC2 Auto Scaling를 사용하면 애플리케이션에서는 다음 이점을 누릴 수 있습니다.

- **내결함성 향상.** Amazon EC2 Auto Scaling에서는 인스턴스가 비정상 상태일 때 이를 감지하여 종료한 다음 이를 대체할 인스턴스를 시작할 수 있습니다. 여러 개의 가용 영역을 사용하도록 Amazon EC2 Auto Scaling을 구성할 수도 있습니다. 하나의 가용 영역이 사용 불가 상태가 되면 Amazon EC2 Auto Scaling에서는 다른 가용 영역에서 새 인스턴스를 시작하여 이에 대처할 수 있습니다.
- **가용성 향상.** Amazon EC2 Auto Scaling은 애플리케이션이 항상 현재 트래픽 요구를 처리할 수 있는 올바른 용량을 갖추도록 도와줍니다.
- **비용 관리 개선.** Amazon EC2 Auto Scaling는 필요에 따라 용량을 동적으로 확장 및 축소할 수 있습니다. 사용한 EC2 인스턴스에 대해서만 비용을 지불하므로, 인스턴스가 필요할 때 이를 시작하고 필요 없어지면 종료함으로써 비용을 절감합니다.

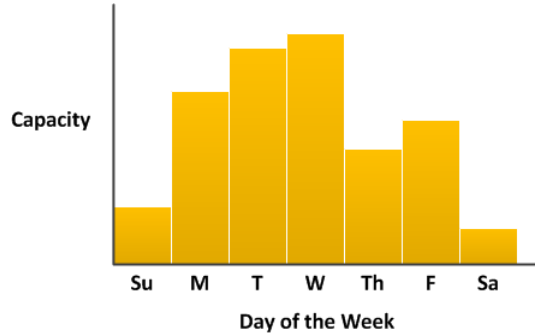
목차

- 예: 가변적인 수요에 대응 (p. 3)
- 예: 웹 앱 아키텍처 (p. 5)
- 예: 가용 영역 전반에 인스턴스 분산 (p. 6)

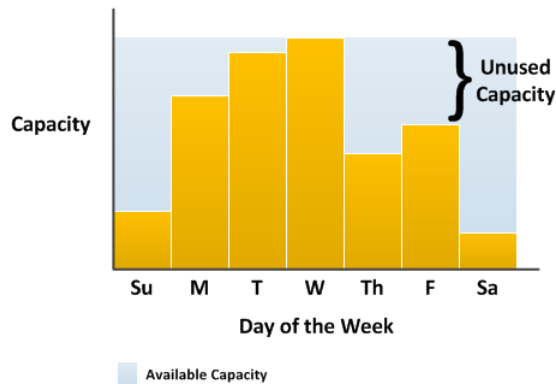
예: 가변적인 수요에 대응

Amazon EC2 Auto Scaling의 이점 중 몇 가지를 시연하기 위해 AWS에서 실행되는 기본 웹 애플리케이션을 살펴보겠습니다. 이 애플리케이션을 사용하여 직원들은 회의에 사용하려는 회의실을 찾을 수 있습니다. 한 주의 시작과 끝에는 이 애플리케이션의 사용량이 최소 수준입니다. 주 중반쯤에는 보다 많은 직원들이 회의 일정을 잡고 있으며, 따라서 애플리케이션에 대한 수요가 눈에 띄게 증가합니다.

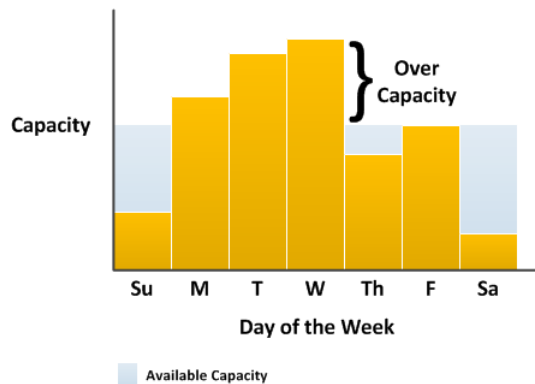
다음 그래프는 일주일 동안 애플리케이션의 용량이 얼마나 사용되었는지를 나타냅니다.



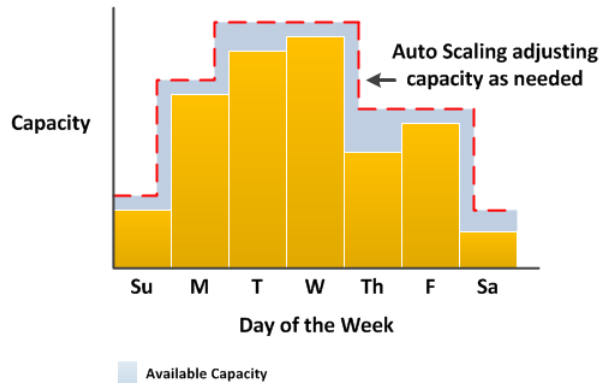
일반적으로 이러한 용량 변동을 계획하는 방법에는 두 가지가 있습니다. 첫 번째 옵션은 충분한 서버를 추가하여 애플리케이션에 항상 수요를 충족할 만큼의 충분한 용량이 확보되도록 하는 것입니다. 하지만 이 옵션의 단점은 애플리케이션에 이만큼의 용량이 필요 없는 날들도 있다는 것입니다. 여분의 용량이 미사용으로 남아 있다는 것은 결국 애플리케이션 실행 유지 비용의 상승을 의미합니다.



두 번째 옵션은 애플리케이션에 대한 평균 수요를 처리할 수 있는 충분한 용량이 확보되도록 하는 것입니다. 이 옵션을 선택하면 가끔씩 사용되는 장비를 구매하지 않아도 되므로 비용이 절감됩니다. 그러나 애플리케이션에 대한 수요가 용량을 초과할 경우 부정적인 고객 경험을 초래할 위험이 있습니다.

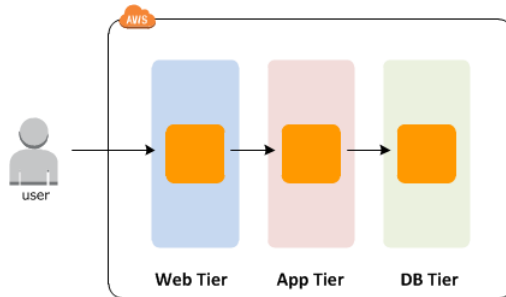


애플리케이션에 Amazon EC2 Auto Scaling을 추가함으로써, 세 번째 옵션을 사용할 수 있습니다. 즉, 필요 시에만 애플리케이션에 새 인스턴스를 추가하고 더 이상 필요 없을 때 이를 종료할 수 있습니다. Amazon EC2 Auto Scaling은 EC2 인스턴스를 사용하므로 사용한 인스턴스에 대해서만 인스턴스 사용 시에 비용을 지불하면 됩니다. 이제 여러분은 비용을 최소화하면서도 최상의 고객 경험을 제공하는 비용 효율적인 아키텍처를 갖게 되었습니다.

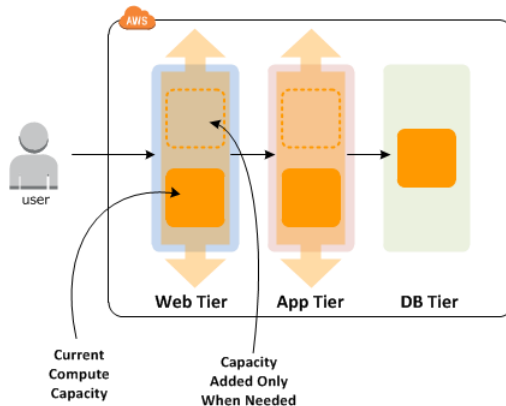


예: 웹 앱 아키텍처

일반적인 웹 앱 시나리오에서는 고객 트래픽 볼륨을 처리하기 위해 여러 개의 앱 사본을 동시에 실행합니다. 이러한 다수의 애플리케이션 사본은 동일한 EC2 인스턴스(클라우드 서버)에서 호스팅되며, 각각에서 고객 요청이 처리됩니다.



Amazon EC2 Auto Scaling에서는 사용자를 대신하여 이러한 EC2 인스턴스의 시작과 종료를 관리합니다. Auto Scaling 그룹에서 EC2 인스턴스를 시작하거나 종료하는 시기를 결정하는 조건(Amazon CloudWatch 경보 등)의 집합을 정의합니다. Auto Scaling 그룹을 네트워크 아키텍처에 추가하면 애플리케이션의 가용성과 내결함성을 향상시킬 수 있습니다.



필요한 만큼 Auto Scaling 그룹을 생성할 수 있습니다. 예를 들어, 각 티어별로 Auto Scaling 그룹을 생성할 수 있습니다.

Auto Scaling 그룹의 인스턴스 간 트래픽을 분산하기 위해 로드 밸런서를 아키텍처에 도입할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹에 로드 밸런서 사용 \(p. 66\)](#) 단원을 참조하십시오.

예: 가용 영역 전반에 인스턴스 분산

EC2 인스턴스와 같은 AWS 리소스가 고가용성 데이터 센터에 구축되어 있습니다. 추가적인 확장성과 안정성을 제공하기 위해 이러한 데이터 센터는 서로 다른 물리적 위치에 존재합니다. 리전은 지리적으로 분산되는 대규모 위치입니다. 각 리전에는 다른 가용 영역에서 발생한 장애를 고립시키기 위해 설계된 가용 영역이라고 하는 여러 개의 개별 장소가 있습니다. 가용 영역은 같은 리전에 있는 다른 가용 영역에 대해 저렴하고 지연 시간이 짧은 네트워크 연결을 제공합니다. 자세한 내용은 Amazon Web Services 일반 참조의 [리전 및 엔드포인트](#) Amazon EC2 Auto Scaling 를 참조하십시오.

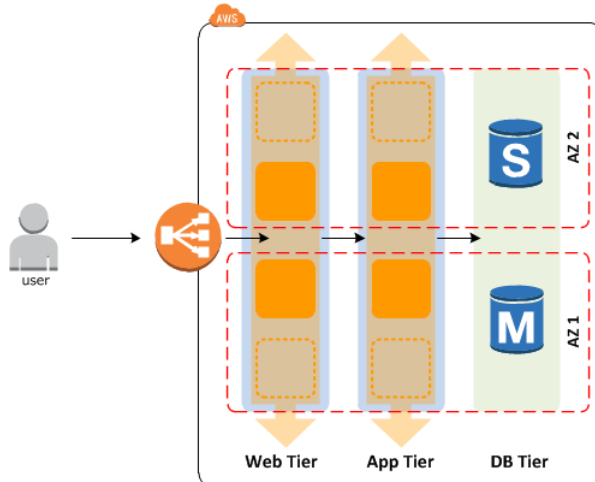
Amazon EC2 Auto Scaling를 사용하여 리전 내 여러 가용 영역에 걸쳐 Auto Scaling 그룹을 확장함으로써 지리적 이중화를 통한 보안 및 안정성을 확보할 수 있습니다. 하나의 가용 영역이 비정상 또는 사용 불가 상태가 되었을 때, Auto Scaling에서는 영향을 받지 않은 가용 영역에서 새 인스턴스를 시작합니다. 비정상 가용 영역이 정상 상태로 복귀하는 경우 지정된 모든 가용 영역에 걸쳐 애플리케이션 인스턴스를 자동으로 고르게 재배포합니다.

Auto Scaling 그룹은 동일한 리전 내에 한 개 이상의 가용 영역에 속하는 EC2 인스턴스를 포함할 수 있습니다. 하지만 Auto Scaling 그룹은 여러 리전을 포괄할 수 없습니다.

VPC 내 Auto Scaling 그룹의 경우, 서브넷에서 EC2 인스턴스가 시작됩니다. Auto Scaling 그룹을 생성하거나 업데이트할 경우 EC2 인스턴스의 서브넷을 선택합니다. 가용 영역당 서브넷을 한 개 이상 선택할 수 있습니다. 자세한 정보는 Amazon VPC 사용 설명서의 [VPC 및 서브넷](#)을 참조하십시오.

인스턴스 분산

Amazon EC2 Auto Scaling은 인스턴스를 Auto Scaling 그룹에 대해 활성화된 가용 영역 간에 고르게 분산하려고 시도합니다. 이는 최소의 인스턴스로 가용 영역에서 새 인스턴스를 시작하려고 하는 방식으로 Amazon EC2 Auto Scaling에 의해 수행됩니다. 하지만 이 시도가 실패하는 경우 성공할 때까지 Amazon EC2 Auto Scaling은 다른 가용 영역에서 인스턴스 시작을 계속 시도합니다. VPC 내 Auto Scaling 그룹의 경우, 한 가용 영역에 여러 서브넷이 있으면 Amazon EC2 Auto Scaling이 가용 영역에서 서브넷을 무작위로 선택합니다.



재분배 활동

특정 작업으로 인해 Auto Scaling 그룹이 가용 영역 간에 불균형하게 분배될 수 있습니다. Amazon EC2 Auto Scaling가 가용 영역을 재분배하여 보상합니다. 다음 작업으로 인해 재분배 활동이 발생할 수 있습니다.

- 그룹의 가용 영역을 변경한 경우
- 명시적으로 인스턴스를 종료 또는 분리하여 그룹이 불균형해지는 경우
- 이전에 용량 부족을 겪었던 가용 영역이 복구되어 추가적인 가용 용량이 확보된 경우
- 이전에 스팟 가격이 최대 가격보다 높았던 가용 영역이 이제 스팟 가격이 하락하여 최대 가격 아래로 떨어진 경우

재분배 시 Amazon EC2 Auto Scaling에서는 이전 인스턴스를 종료하기 전에 새 인스턴스를 시작하여 재분배로 인해 애플리케이션의 성능이나 가용성이 저하되지 않도록 합니다.

Amazon EC2 Auto Scaling에서는 이전 인스턴스 종료 전에 새 인스턴스를 시작하려 하므로 지정된 최대 용량에 도달하거나 이에 근접하면 재분배 활동을 지연시키거나 완전히 중지할 수 있습니다. 이 문제를 피하기 위해 시스템에서는 재분배 활동 중에 그룹의 지정된 최대 용량을 10% 여유(또는 인스턴스 1개의 여유 중 큰 쪽)만큼 일시적으로 초과할 수 있습니다. 그룹이 최대 용량에 도달하거나 이에 근접하여 재분배가 필요한 경우, 또는 사용자 요청에 의한 영역 재조정이나 영역 가용성 문제에 대처하기 위한 경우에만 이 여유분만큼 확장됩니다. 확장은 그룹에 재분배가 필요한 동안에만 유지됩니다(대개 몇 분 정도).

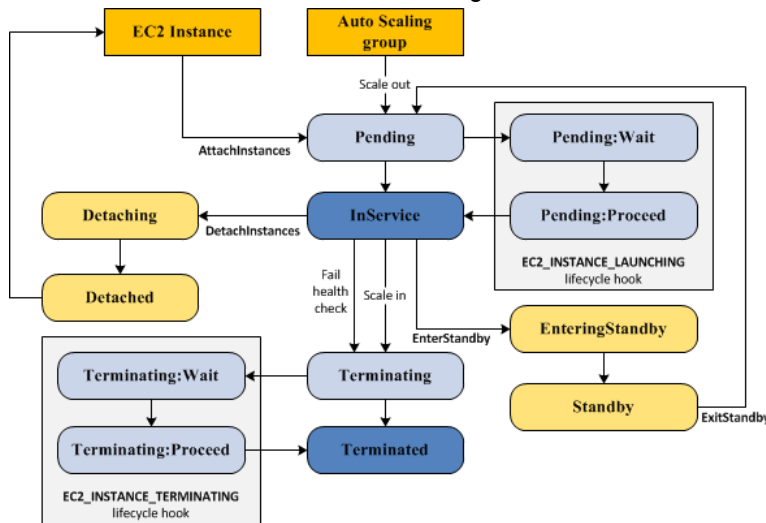
Auto Scaling 수명 주기

Auto Scaling 그룹의 EC2 인스턴스에는 다른 EC2 인스턴스와는 다른 경로, 즉 수명 주기가 있습니다. 수명 주기는 Auto Scaling 그룹이 인스턴스를 시작하고 서비스에 들어갈 때 시작됩니다. 수명 주기는 인스턴스를 종료하거나 Auto Scaling 그룹이 인스턴스를 서비스에서 제외시키고 이를 종료할 때 끝납니다.

Note

인스턴스가 시작되는 즉시 인스턴스에 대한 요금이 청구되며, 아직 서비스되지 않는 시간도 포함됩니다.

다음 그림에서는 Amazon EC2 Auto Scaling 수명 주기에서 인스턴스 상태 간 전환을 보여 줍니다.



확장

다음 확장 이벤트는 Auto Scaling 그룹에 EC2 인스턴스를 시작하고 이를 그룹에 연결하라고 지시합니다.

- 그룹의 크기를 수동으로 늘립니다. 자세한 내용은 [Amazon EC2 Auto Scaling 수동 조정 \(p. 90\)](#) 단원을 참조하십시오.
- 지정된 수요 증가에 따라 그룹의 크기를 자동으로 늘리는 조정 정책을 만듭니다. 자세한 내용은 [Amazon EC2 Auto Scaling의 동적 조정 \(p. 102\)](#) 단원을 참조하십시오.
- 특정 시간에 그룹의 크기를 늘리도록 조정을 일정 기반으로 설정합니다. 자세한 내용은 [Amazon EC2 Auto Scaling의 예약된 조정 \(p. 99\)](#) 단원을 참조하십시오.

확장 이벤트가 발생하면 Auto Scaling 그룹이 할당된 시작 구성을 사용하여 필요한 수의 EC2 인스턴스를 시작합니다. 이러한 인스턴스는 **Pending** 상태에서 시작됩니다. Auto Scaling 그룹에 수명 주기 후크를 추가하

면 여기에서 사용자 지정 작업을 수행할 수 있습니다. 자세한 내용은 [수명 주기 후크 \(p. 9\)](#) 단원을 참조하십시오.

각 인스턴스가 완전히 구성되고 Amazon EC2 상태 확인을 통과하면, Auto Scaling 그룹에 연결되고 InService 상태에 들어갑니다. 이 인스턴스는 원하는 Auto Scaling 그룹 용량에서 감산됩니다.

서비스 상태의 인스턴스

인스턴스는 다음 중 하나가 발생할 때까지 InService 상태로 유지됩니다.

- 축소 이벤트가 발생하면 Amazon EC2 Auto Scaling은 Auto Scaling 그룹의 크기를 줄이기 위해 이 인스턴스를 종료합니다. 자세한 내용은 [축소 시 Auto Scaling 인스턴스 종료 제어 \(p. 126\)](#) 단원을 참조하십시오.
- 인스턴스를 Standby 상태로 설정하는 경우 자세한 내용은 [대기 모드 시작 및 종료 \(p. 9\)](#) 단원을 참조하십시오.
- Auto Scaling 그룹에서 인스턴스를 분리합니다. 자세한 내용은 [인스턴스 분리 \(p. 8\)](#) 단원을 참조하십시오.
- 인스턴스가 필요한 수의 상태 확인에 실패한 경우, Auto Scaling 그룹에서 제거, 종료 및 교체됩니다. 자세한 내용은 [Auto Scaling 인스턴스 상태 확인 \(p. 148\)](#) 단원을 참조하십시오.

축소

다음 축소 이벤트는 Auto Scaling 그룹이 그룹에서 EC2 인스턴스를 분리하고 이를 종료하라고 지시합니다.

- 그룹의 크기를 수동으로 줄입니다. 자세한 내용은 [Amazon EC2 Auto Scaling 수동 조정 \(p. 90\)](#) 단원을 참조하십시오.
- 지정된 수요 감소에 따라 그룹의 크기를 자동으로 줄이는 조정 정책을 만듭니다. 자세한 내용은 [Amazon EC2 Auto Scaling의 동적 조정 \(p. 102\)](#) 단원을 참조하십시오.
- 특정 시간에 그룹의 크기를 줄이도록 조정을 일정 기반으로 설정합니다. 자세한 내용은 [Amazon EC2 Auto Scaling의 예약된 조정 \(p. 99\)](#) 단원을 참조하십시오.

생성한 확장 이벤트 각각에 대해 이에 상응하는 축소 이벤트를 생성하는 것이 중요합니다. 이렇게 하면 애플리케이션에 할당된 리소스와 그러한 리소스의 수요를 가능한 한 가깝게 일치시킬 수 있습니다.

축소 이벤트가 발생하면 Auto Scaling 그룹에서 하나 이상의 인스턴스를 분리합니다. Auto Scaling 그룹이 종료 정책을 사용하여 종료할 인스턴스를 결정합니다. Auto Scaling 그룹에서 분리되어 종료 중인 인스턴스는 Terminating 상태로 들어가며, 다시 서비스 상태로 돌아갈 수 없습니다. Auto Scaling 그룹에 수명 주기 후크를 추가하면 여기에서 사용자 지정 작업을 수행할 수 있습니다. 마지막으로 인스턴스가 완전히 종료되고 Terminated 상태로 들어갑니다.

인스턴스 연결

Auto Scaling 그룹에 특정 기준을 충족하는 실행 중인 EC2 인스턴스를 연결할 수 있습니다. 인스턴스가 연결되면 Auto Scaling 그룹의 일부로 관리됩니다.

자세한 내용은 [Auto Scaling 그룹에 EC2 인스턴스 연결 \(p. 92\)](#) 단원을 참조하십시오.

인스턴스 분리

Auto Scaling 그룹에서 인스턴스를 분리할 수 있습니다. 인스턴스를 분리한 후에는 이를 Auto Scaling 그룹과 별도로 관리하거나 다른 Auto Scaling 그룹에 연결할 수 있습니다.

자세한 내용은 [Auto Scaling 그룹에서 EC2 인스턴스 분리 \(p. 96\)](#) 단원을 참조하십시오.

수명 주기 후크

인스턴스를 시작하거나 종료할 때 사용자 지정 작업을 수행할 수 있도록 Auto Scaling 그룹에 수명 주기 후크를 추가할 수 있습니다.

Amazon EC2 Auto Scaling은 확장 이벤트에 응답하면 하나 이상의 인스턴스를 시작합니다. 이러한 인스턴스는 Pending 상태에서 시작됩니다. Auto Scaling 그룹에 `autoscaling:EC2_INSTANCE_LAUNCHING` 수명 주기 후크를 추가한 경우, 인스턴스가 Pending 상태에서 Pending:Wait 상태로 이동합니다. 수명 주기 작업을 완료하면 인스턴스가 Pending:Proceed 상태로 들어갑니다. 인스턴스가 완전히 구성되면 Auto Scaling 그룹에 연결되고 InService 상태로 들어갑니다.

Amazon EC2 Auto Scaling은 축소 이벤트에 응답할 경우 하나 이상의 인스턴스를 종료합니다. Auto Scaling 그룹에서 이러한 인스턴스가 분리되고 Terminating 상태로 들어갑니다. Auto Scaling 그룹에 `autoscaling:EC2_INSTANCE_TERMINATING` 수명 주기 후크를 추가한 경우, 인스턴스가 Terminating 상태에서 Terminating:Wait 상태로 이동합니다. 수명 주기 작업을 완료하면 인스턴스가 Terminating:Proceed 상태로 들어갑니다. 인스턴스가 완전히 종료되면 Terminated 상태로 들어갑니다.

자세한 내용은 [Amazon EC2 Auto Scaling 수명 주기 후크 \(p. 131\)](#) 단원을 참조하십시오.

대기 모드 시작 및 종료

InService 상태인 인스턴스를 Standby 상태로 변경할 수 있습니다. 이를 통해 서비스에서 인스턴스를 제거하거나 문제를 해결하거나 변경한 다음 다시 서비스 상태로 되돌릴 수 있습니다.

Standby 상태의 인스턴스는 계속해서 Auto Scaling 그룹에서 관리됩니다. 그러나, 이러한 인스턴스를 다시 서비스 상태로 되돌리기 전까지는 애플리케이션의 활성 부분이 아닙니다.

자세한 내용은 [Auto Scaling 그룹에서 일시적으로 인스턴스 제거 \(p. 138\)](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling 서비스 할당량

AWS 계정에는 Amazon EC2 Auto Scaling에 대한 다음과 같은 기본 할당량(이전에는 제한이라고 함)이 있습니다.

기본 할당량

- 리전별 시작 구성: 200
- 리전별 Auto Scaling 그룹: 200

계정에 대한 현재 할당량을 보려면 <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 열고 제한 페이지로 이동합니다. `describe-account-limits` 명령을 사용할 수도 있습니다. 증가를 요청하려면 [Auto Scaling 제한 양식](#)을 사용하십시오.

Auto Scaling 그룹 할당량

- Auto Scaling 그룹당 조정 정책: 50
- Auto Scaling 그룹당 예약된 작업: 125
- Auto Scaling 그룹당 수명 주기 후크: 50
- Auto Scaling 그룹당 SNS 주제: 10
- Auto Scaling 그룹당 Classic Load Balancer: 50
- Auto Scaling 그룹당 대상 그룹: 50

조정 정책 할당량

- 조정 정책당 단계 조정: 20

Auto Scaling API 할당량

- 한 번에 최대 20개의 인스턴스 ID를 포함하여 [AttachInstances](#), [DetachInstances](#), [EnterStandby](#), [ExitStandby](#)를 사용할 수 있습니다.
- 한 번에 최대 10개의 로드 밸런서를 포함하여 [AttachLoadBalancers](#), [DetachLoadBalancers](#)를 사용할 수 있습니다.
- 한 번에 최대 10개의 대상 그룹을 포함하여 [AttachLoadBalancerTargetGroups](#), [DetachLoadBalancerTargetGroups](#)를 사용할 수 있습니다.

다른 서비스의 서비스 할당량에 대한 자세한 내용은 Amazon Web Services 일반 참조의 [서비스 엔드포인트 및 할당량](#)을 참조하십시오.

Amazon EC2 Auto Scaling 설정

Amazon EC2 Auto Scaling를 사용하기 전에, 먼저 다음 작업을 완료합니다.

작업

- [AWS에 가입](#) (p. 11)
- [Amazon EC2 사용 준비](#) (p. 11)

AWS에 가입

AWS 계정을 생성하면 모든 AWS 서비스에 자동으로 계정이 등록됩니다. 사용한 서비스에 대해서만 지불하면 됩니다. EC2 인스턴스에 대해 지불하는 비용에 추가되는 비용 없이 Amazon EC2 Auto Scaling를 사용할 수 있습니다.

AWS 계정이 없는 경우 다음과 같이 AWS에 가입합니다.

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.

AWS에서는 가입 프로세스가 완료된 후 확인 이메일을 보냅니다.

Amazon EC2 사용 준비

Amazon EC2를 처음으로 사용하는 경우 Amazon EC2 문서에 나와 있는 작업을 완료하십시오. 자세한 정보는 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2 설정](#) 또는 Windows 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2 설정](#)을 참조하십시오.

Amazon EC2 Auto Scaling 시작하기

Amazon EC2 Auto Scaling을 사용할 때 특정 기본 구성 요소를 사용하여 시작해야 합니다. 이 자습서에서는 Amazon EC2 Auto Scaling에 대한 기본 인프라를 생성하기 위한 빌딩 블록을 설정하는 프로세스를 안내합니다.

애플리케이션에 사용할 Auto Scaling 그룹을 생성하기 전에 AWS 클라우드에서 실행할 애플리케이션을 철저히 검토하십시오. 다음을 고려하십시오.

- Auto Scaling 그룹을 분산할 가용 영역 수
- 사용할 수 있는 기존 리소스(예: 보안 그룹 또는 Amazon 머신 이미지(AMI))
- 조정 기능을 사용하여 용량을 늘리거나 줄일지, 아니면 단지 일정한 수의 서버가 항상 실행되도록 하려고 하는지 Amazon EC2 Auto Scaling은 두 기능을 동시에 수행할 수 있습니다.
- 애플리케이션의 성능과 가장 연관성이 높은 지표
- 서버 시작 및 구성에 걸린 시간

애플리케이션을 잘 이해할수록 Auto Scaling 아키텍처를 더 효과적으로 구축할 수 있습니다.

지침은 다음과 같습니다.

- EC2 인스턴스를 정의하는 구성 템플릿을 생성합니다. 기본 설정에 따라 시작 템플릿이나 시작 구성 지침을 선택할 수 있습니다.
- 인스턴스 상태가 이상이 있는 경우에도 고정된 수의 인스턴스를 계속 유지하도록 Auto Scaling 그룹을 생성합니다.
- 선택에 따라 이 기본 인프라를 삭제합니다.

이 자습서에서는 사용자가 EC2 인스턴스 시작 작업에 익숙하고 키 페어와 보안 그룹을 이미 생성한 것으로 가정합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2 설정](#) 단원을 참조하십시오.

시작하려면 [프리 티어](#) 자격을 갖춘 단일 Linux 인스턴스를 실행할 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 않은 경우 이 자습서를 따르면 인스턴스를 시작하는 시점부터 Auto Scaling 그룹(이 자습서의 최종 작업)을 삭제하여 인스턴스 상태가 `terminated`로 변경될 때까지 표준 Amazon EC2 사용 요금이 발생합니다.

작업

- [1단계: 시작 템플릿 생성](#) (p. 12)
- [2단계: Auto Scaling 그룹 생성](#) (p. 14)
- [3단계: Auto Scaling 그룹 확인](#) (p. 16)
- [4단계: 다음 단계](#) (p. 17)
- [5단계: \(선택 사항\) 조정 인프라 삭제](#) (p. 17)

1단계: 시작 템플릿 생성

이 단계에서는 Amazon EC2 Auto Scaling이 사용자를 위해 생성하는 EC2 인스턴스 유형을 지정하는 시작 템플릿을 생성합니다. 사용할 Amazon 머신 이미지(AMI) ID, 인스턴스 유형, 키 페어 및 보안 그룹 같은 정보를 포함시킵니다.

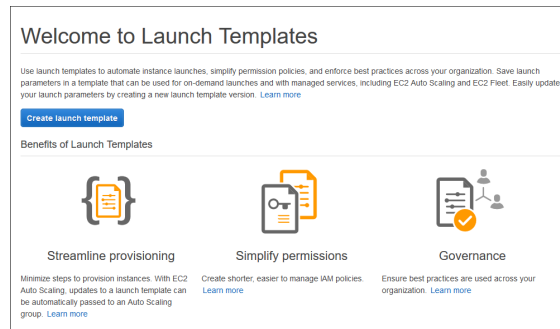
시작 템플릿 생성에 대한 자세한 지침은 [Auto Scaling 그룹을 위한 시작 템플릿 만들기](#) (p. 26) 단원을 참조하십시오.

Note

또는 시작 템플릿을 사용하는 대신 시작 구성을 사용하여 Auto Scaling 그룹을 만들 수 있습니다. 시작 구성 지침은 [시작 구성 생성](#) 단원을 참조하십시오.

시작 템플릿을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 AWS 리전을 선택합니다. 생성한 Amazon EC2 Auto Scaling 리소스는 지정된 리전과 연결됩니다.
3. 탐색 창에서 시작 템플릿을 선택합니다.
4. 시작 템플릿 생성을 선택합니다.



5. 새 템플릿 생성을 선택합니다. 시작 템플릿 이름에 이름을 입력합니다(예: my_template).
6. 템플릿 버전 설명에서 시작 템플릿에 대한 설명을 입력하면 이 템플릿의 용도(예: test launch template for an Auto Scaling group)를 기억하는 데 도움이 됩니다.
7. AMI ID는 빠른 시작 목록에서 Amazon Linux 2(HVM) 버전을 선택합니다. Amazon 머신 이미지(AMI)는 인스턴스의 기본 구성 템플릿 역할을 합니다.
8. 인스턴스 유형에서 지정한 AMI와 호환되는 하드웨어 구성을 선택합니다. 프리 티어 Linux 서버는 t2.micro 인스턴스입니다.

Note

계정이 12개월이 아직 안 된 경우 특정 사용 한도 내에서 무료로 t2.micro 인스턴스를 사용할 수 있습니다. 자세한 정보는 [AWS 프리 티어](#) 단원을 참조하십시오.

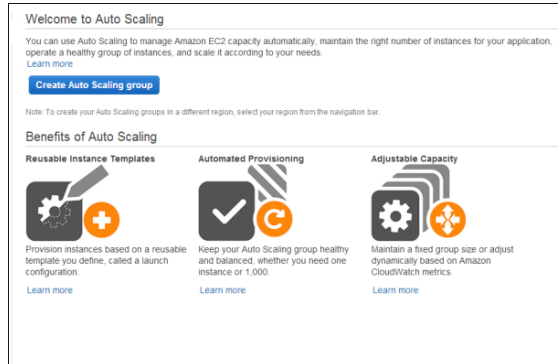
9. (선택 사항) 키 페어 이름에서 기존 키 페어를 선택합니다. SSH를 통해 Amazon EC2 인스턴스에 연결하기 위해 키 페어를 사용합니다. 인스턴스 연결은 이 자습서의 내용에 포함되어 있지 않습니다. 따라서 인스턴스에 연결하려는 경우가 아닌 한 키 페어를 지정할 필요가 없습니다.
10. 네트워크 유형을 VPC로 설정합니다.
11. 보안 그룹에 Auto Scaling 그룹의 VPC로 사용할 VPC의 보안 그룹을 지정합니다. 보안 그룹을 지정하지 않을 경우 인스턴스가 VPC의 기본 보안 그룹에 자동으로 연결됩니다.
12. 네트워크 인터페이스는 비워 둘 수 있습니다. 이 설정을 비워두면 인스턴스에 대해 선택한 IP 주소를 가진 기본 네트워크 인터페이스가 생성됩니다(네트워크 인터페이스가 설정된 서브넷 기준). 대신에 네트워크 인터페이스를 지정하도록 선택한 경우에는 보안 그룹이 네트워크 인터페이스의 일부여야 합니다.
13. 아래로 스크롤하여 Create launch template(시작 템플릿 만들기)를 선택합니다.
14. 확인 페이지에서 Auto Scaling 그룹 생성을 선택합니다.

현재 시작 템플릿을 사용하고 있지 않고 시작 템플릿을 생성하지 않으려는 경우에는 대신 시작 구성을 생성할 수 있습니다.

시작 구성은 Amazon EC2 Auto Scaling이 사용자를 위해 생성하는 EC2 인스턴스 유형을 지정한다는 점에서 시작 템플릿과 비슷합니다. 사용할 Amazon 머신 이미지(AMI)의 ID, 인스턴스 유형, 키 페어, 보안 그룹 등의 정보를 포함시켜서 시작 구성을 생성합니다.

시작 구성을 생성하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 모음에서 AWS 리전을 선택합니다. 생성한 Auto Scaling 리소스는 생성한 리전과 연결됩니다.
3. 탐색 창의 Auto Scaling에서 Auto Scaling 그룹을 선택합니다.
4. Auto Scaling 시작 페이지에서 Auto Scaling 그룹 생성을 선택합니다.



5. Auto Scaling 그룹 생성 페이지에서, Launch Configuration, Create a new launch configuration(시작 구성, 새 시작 구성 만들기)을 선택한 후 다음 단계를 선택합니다.
6. AMI 선택 단계에는 인스턴스에 대한 템플릿 역할을 하는 Amazon 머신 이미지(AMI)라는 기본 구성 목록이 있습니다. Amazon Linux 2 AMI는 선택을 선택합니다.
7. 인스턴스 유형 선택 단계에서 인스턴스의 하드웨어 구성을 선택합니다. 기본 t2.micro 인스턴스를 그대로 유지하는 것이 좋습니다. [Next: Configure details]를 선택합니다.

Note

계정이 12개월이 아직 안 된 경우 특정 사용 한도 내에서 무료로 t2.micro 인스턴스를 사용할 수 있습니다. 자세한 정보는 [AWS 프리 티어](#) 단원을 참조하십시오.

8. 세부 정보 구성 단계에서 다음을 수행합니다.
 - a. 이름에 시작 구성의 이름을 입력합니다(예 my-first-1c).
 - b. 고급 세부 정보에서 IP 주소 형식을 선택합니다. VPC의 인스턴스에 인터넷 연결을 제공하려는 경우 퍼블릭 IP 주소를 할당하는 옵션을 선택합니다. 인스턴스가 기본 VPC로 시작되는 경우 기본값은 퍼블릭 IP 주소를 할당하는 것입니다. 인스턴스에 인터넷 연결을 제공하려는데 기본 VPC가 있는지 확실치 않은 경우, Assign a public IP address to every instance(모든 인스턴스에 퍼블릭 IP 주소 할당)를 선택합니다.
 - c. Skip to review(검토로 이동)를 선택합니다.
9. 검토 단계에서 보안 그룹 편집을 선택합니다. 지침을 따라 기존 보안 그룹을 선택한 다음 [Review]를 선택합니다.
10. 검토 단계에서 시작 구성 생성을 선택합니다.
11. 지침에 따라 기존 키 페어 선택 또는 새 키 페어 생성 단계를 완료합니다. 인스턴스 연결은 이 자습서의 내용에 포함되어 있지 않습니다. 따라서 인스턴스에 연결하려는 경우가 아닌 한 키 페어 없이 [Proceed without a key pair]를 선택할 수 있습니다.
12. [Create launch configuration]을 선택합니다.
13. Create an Auto Scaling group using this launch configuration(이 시작 구성을 사용하여 Auto Scaling 그룹 생성)을 선택합니다. Auto Scaling 그룹을 생성하는 마법사가 표시됩니다.

2단계: Auto Scaling 그룹 생성

Auto Scaling 그룹은 EC2 인스턴스의 모음으로, Amazon EC2 Auto Scaling의 핵심입니다. Auto Scaling 그룹을 생성할 때 인스턴스에 대한 서버넷과 시작할 초기 인스턴스 수와 같은 정보를 포함합니다.

시작 템플릿 또는 시작 구성을 생성한 후 중단한 시점부터 계속하려면 다음 절차를 따르십시오.

Note

Amazon EC2 Auto Scaling에서는 최근에 Auto Scaling 그룹 인터페이스를 변경했습니다. 기본적으로 기존 사용자 인터페이스가 표시되지만 새 사용자 인터페이스로 전환할 수 있습니다. 이 주제에는 둘 모두에 대한 단계가 포함되어 있습니다.
새 사용자 인터페이스로 전환하려면 Auto Scaling 그룹 페이지 상단에 있는 배너에서 Go to the new console(새 콘솔로 이동)을 선택합니다.

Auto Scaling 그룹을 생성하려면(새 콘솔)

1. 새 콘솔의 Auto Scaling 그룹 페이지에서 Create an Auto Scaling group(Auto Scaling 그룹 생성)을 선택합니다.
2. Choose launch template or configuration(시작 템플릿 또는 구성 선택) 페이지의 Auto Scaling group name(Auto Scaling 그룹 이름)에 Auto Scaling 그룹의 이름을 입력합니다.
3. 시작 템플릿에서 다음 옵션 중 하나를 선택합니다.
 - a. [시작 템플릿에만 해당] 생성한 시작 템플릿을 선택한 다음, 스케일 아웃 시 사용할 시작 템플릿의 기본 버전을 선택합니다.
 - b. [시작 구성에만 해당] Switch to launch configuration(시작 구성으로 전환)을 선택한 다음, 생성한 시작 구성을 선택합니다.
4. 다음을 선택합니다.

네트워크 설정을 구성할 수 있게 해주고 여러 인스턴스 유형에서 온디맨드 및 스팟 인스턴스를 시작할 수 있는 옵션을 제공하는 Configure settings(설정 구성) 페이지가 나타납니다(템플릿 시작을 선택한 경우).

5. [시작 템플릿에만 해당] Purchase options and instance types(구매 옵션 및 인스턴스 유형) 설정을 Adhere to the launch template(시작 템플릿 준수)로 유지하여 Auto Scaling 그룹을 신속하게 생성 및 구성합니다.
6. 네트워크를 선택한 AWS 리전의 기본 VPC로 두거나 자체 VPC를 선택합니다. 기본 VPC는 인스턴스에 인터넷 연결을 제공하도록 자동으로 구성됩니다. 이 VPC는 해당 리전의 각 가용 영역에 퍼블릭 서브넷을 포함시킵니다.
7. 서브넷에서, 포함하고자 하는 각 가용 영역의 서브넷을 선택합니다. 여러 가용 영역의 서브넷을 사용하여 가용성을 높일 수 있습니다.
8. 이 자습서의 나머지 기본값을 유지하고 Skip to review(검토로 이동)를 선택합니다.

Note

그룹의 초기 크기는 원하는 용량에 따라 결정됩니다. 기본값은 1 인스턴스입니다.

9. 검토 페이지에서 그룹에 대한 정보를 검토한 다음 Create Auto Scaling Group(Auto Scaling 그룹 생성)을 선택합니다.

새 사용자 인터페이스를 사용하는 것이 좋지만 이전 사용자 인터페이스도 임시로 사용할 수 있습니다.

Auto Scaling 그룹을 생성하려면(이전 콘솔)

1. Auto Scaling 그룹 세부 정보 구성 단계에서 다음을 수행합니다.
 - a. 그룹 이름에 Auto Scaling 그룹의 이름을 입력합니다(예: my-first-asg).

- [시작 템플릿] 시작 템플릿 버전에서 스케일 아웃 시 사용할 시작 템플릿의 기본 버전을 선택합니다.
 - [시작 템플릿] 플릿 구성에서 시작 템플릿 준수를 선택합니다.
 - 이 자습서에서는 [Group size]를 기본값인 1 인스턴스로 둡니다.
 - 네트워크를 선택한 AWS 리전의 기본 VPC로 두거나 자체 VPC를 선택합니다. 기본 VPC는 인스턴스에 인터넷 연결을 제공하도록 자동으로 구성됩니다. 이 VPC는 해당 리전의 각 가용 영역에 퍼블릭 서브넷을 포함시킵니다.
 - 서브넷에서, 포함하고자 하는 각 가용 영역의 서브넷을 선택합니다. 여러 가용 영역의 서브넷을 사용하여 가용성을 높일 수 있습니다.
 - [Next: Configure scaling policies]를 선택합니다.
- [Configure scaling policies] 페이지에서 [Keep this group at its initial size]와 [Review]를 차례로 선택합니다.
 - 검토 페이지에서 Auto Scaling 그룹 생성을 선택합니다.
 - Auto Scaling 그룹 생성 상태 페이지에서 닫기를 선택합니다.

3단계: Auto Scaling 그룹 확인

이제 Auto Scaling 그룹을 생성했으므로 해당 그룹에서 EC2 인스턴스를 시작했는지 확인할 준비가 완료되었습니다.

Auto Scaling group(Auto Scaling 그룹) 페이지에서 다음 절차를 완료합니다.

Auto Scaling 그룹에서 EC2 인스턴스를 시작했는지 확인하려면

- 방금 생성한 Auto Scaling 그룹 옆의 확인란을 선택합니다.
사용 가능한 첫 번째 탭은 Auto Scaling 그룹에 대한 정보를 표시하는 세부 정보 탭입니다.
- 두 번째 탭인 활동을 선택합니다. Activity history(활동 기록)에서 Auto Scaling 그룹과 연결된 활동의 진행률을 볼 수 있습니다. 상태 옆에는 인스턴스의 현재 상태가 표시됩니다. 인스턴스가 시작되는 동안 상태 옆에 `preInService`가 표시됩니다. 인스턴스가 시작되면 상태가 `successful`로 변경됩니다. 새로 고침 버튼을 사용하여 인스턴스의 현재 상태를 볼 수도 있습니다. (이전 콘솔: Activity History(활동 기록) 탭에서는 인스턴스의 상태를 볼 수 있습니다. 인스턴스가 시작되는 동안 상태 옆에 `In progress`이 표시됩니다.)
- 인스턴스 관리 탭의 인스턴스에서 인스턴스의 상태를 볼 수 있습니다. (이전 콘솔: 인스턴스 탭에서는 인스턴스의 상태를 볼 수 있습니다.)
- 인스턴스가 성공적으로 시작되었는지 확인합니다. 인스턴스를 시작하는 데 약간 시간이 걸립니다.

수명 주기 옆에는 인스턴스의 상태가 표시됩니다. 이제 Auto Scaling 그룹에서 EC2 인스턴스가 시작되었고 `InService` 수명 주기 상태인지 확인합니다.

상태 확인 옆에 해당 인스턴스에 대한 EC2 인스턴스 상태 확인 결과가 표시됩니다.

(선택사항) Auto Scaling 그룹에서 인스턴스 종료

이러한 단계를 사용하여 Amazon EC2 Auto Scaling 작동 방식, 특히 필요 시 새 인스턴스를 시작하는 방법에 대해 자세히 알아볼 수 있습니다. 이 자습서에서 생성한 Auto Scaling 그룹의 최소 크기는 하나의 인스턴스입니다. 따라서 실행 중인 인스턴스를 종료하는 경우에는 Amazon EC2 Auto Scaling에서 새 인스턴스를 시작해서 교체해야 합니다.

1. 인스턴스 관리 탭의 인스턴스에서 인스턴스의 ID를 선택합니다. (이전 콘솔: 인스턴스 탭에서 인스턴스의 ID 선택 가능)

이렇게 하면 Amazon EC2 콘솔의 인스턴스 페이지로 이동하여 인스턴스를 종료할 수 있습니다.
2. [Actions], [Instance State], [Terminate]를 차례로 선택합니다. 확인 메시지가 나타나면 [Yes, Terminate]를 선택합니다.
3. 탐색 창에서 Auto Scaling 그룹을 선택합니다. Auto Scaling 그룹을 선택하고 활동 탭을 선택합니다. (이전 콘솔: 활동 기록 탭)

Auto Scaling 그룹의 기본 휴지는 300초(5분)이므로 조정 활동이 표시될 때까지 약 5분이 소요됩니다. 활동 기록에서 조정 활동이 시작되면 첫 번째 인스턴스를 종료하기 위한 항목과 새 인스턴스를 시작하기 위한 항목이 표시됩니다.
4. 인스턴스 관리 탭의 인스턴스 섹션에 새 인스턴스만 표시됩니다. (이전 콘솔: 인스턴스 탭에서 이를 확인할 수 있음)
5. 탐색 창에서 인스턴스를 선택합니다. 이 페이지에는 종료된 인스턴스와 실행 중인 새로운 인스턴스가 모두 표시됩니다.

4단계: 다음 단계

방금 생성한 Auto Scaling을 위한 기본 인프라를 삭제하려면 다음 단계로 이동합니다. 그렇지 않으면 이 인프라를 기본으로 사용하여 다음 중 하나 이상을 시도해 볼 수 있습니다.

- Auto Scaling 그룹을 수동으로 조정합니다. 자세한 내용은 [용량 제한 설정 \(p. 89\)](#) 및 [수동 조정 \(p. 90\)](#) 단원을 참조하십시오.
- 리소스 사용률 변경에 따라 자동으로 조정하는 방법을 알아봅니다. 로드가 증가하면 수요를 처리하기 위해 Auto Scaling 그룹을 확장(인스턴스 추가)할 수 있습니다. 자세한 내용은 [대상 추적 조정 정책 \(p. 104\)](#) 단원을 참조하십시오.
- Auto Scaling 그룹이 조정될 때마다 알림을 보내도록 SNS 알림을 구성합니다. 자세한 내용은 [Amazon SNS 알림 \(p. 150\)](#) 단원을 참조하십시오.

5단계: (선택 사항) 조정 인프라 삭제

조정 인프라를 삭제하거나 Auto Scaling 그룹만 삭제하고 나중에 사용하기 위해 해당 시작 템플릿 또는 시작 구성을 유지할 수 있습니다.

[AWS 프리 티어](#) 내에 없는 인스턴스를 시작한 경우 추가 비용을 방지하기 위해 인스턴스를 종료해야 합니다. 인스턴스를 종료하면 인스턴스와 연결된 데이터도 삭제됩니다.

Auto Scaling 그룹을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹 옆에 있는 확인란을 선택합니다.
4. 삭제를 선택합니다. (이전 콘솔: 작업, 삭제를 선택합니다.) 확인 메시지가 나타나면 삭제를 선택합니다.

이름 옆의 로딩 아이콘은 Auto Scaling 그룹이 삭제 중임을 나타냅니다. 삭제가 수행되면 Desired(희망), 최소 및 최대 옆에 Auto Scaling 그룹의 0 인스턴스가 표시됩니다. 인스턴스를 종료하고 그룹을 삭제하는 데 몇 분 정도 걸립니다. 목록을 새로 고침하여 상태를 확인합니다.

현재 시작 템플릿을 유지하려면 아래 절차를 건너뛰십시오.

시작 템플릿을 삭제하려면

1. 탐색 창에서 시작 템플릿을 선택합니다.
2. 시작 템플릿을 선택합니다.
3. [Actions]와 [Delete template]를 차례로 선택합니다. 확인 메시지가 나타나면 Delete launch template을 선택합니다.

현재 시작 구성을 유지하려면 아래 절차를 건너뛰십시오.

시작 구성을 삭제하려면

1. 탐색 창의 Auto Scaling 아래에서 시작 구성을 선택합니다.
2. 시작 구성을 선택합니다.
3. [Actions], [Delete launch configuration]을 선택합니다. 확인 메시지가 나타나면 예, 삭제합니다.를 선택합니다.

자습서: 조정 및 로드 밸런싱된 애플리케이션 설정

Important

이 자습서를 살펴보기 전에 먼저 [Amazon EC2 Auto Scaling 시작하기 \(p. 12\)](#) 입문 자습서를 검토하는 것이 좋습니다.

Elastic Load Balancing 로드 밸런서에 Auto Scaling 그룹을 등록하면 로드 밸런싱된 애플리케이션을 설정하는 데 도움이 됩니다. Elastic Load Balancing은 Amazon EC2 Auto Scaling과 함께 작동하여 정상적인 Amazon EC2 인스턴스 전체에 들어오는 트래픽의 로드 밸런싱을 수행합니다. 이렇게 하면 애플리케이션의 확장성과 가용성이 향상됩니다. 단일 가용 영역 또는 여러 가용 영역 내에서 Elastic Load Balancing을 활성화하여 애플리케이션의 내결함성을 높일 수 있습니다. 자세한 내용은 [Auto Scaling 그룹에 로드 밸런서 사용 \(p. 66\)](#) 단원을 참조하십시오.

이 자습서에서는 Auto Scaling 그룹을 만들 때 로드 밸런싱된 애플리케이션을 설정하는 기본 단계에 대해 설명합니다. 기존 Auto Scaling 그룹을 로드 밸런서에 등록하려면 대신 [로드 밸런서 연결 \(p. 67\)](#) 단원을 참조하십시오.

Elastic Load Balancing은 Application Load Balancer, Network Load Balancer, Classic Load Balancer의 세 가지 유형의 로드 밸런서를 지원합니다. Application Load Balancer 또는 Network Load Balancer를 사용하는 것이 좋습니다. 그러나 애플리케이션에 필요한 기능을 지원하는 경우 Classic Load Balancer를 계속 사용할 수 있습니다. 자세한 내용은 [Elastic Load Balancing 유형 \(p. 66\)](#) 단원을 참조하십시오.

동적 트래픽에 대해 확장 가능하고 로드 밸런싱된 아키텍처를 배포하는 절차는 다음 단계로 구성됩니다.

- 먼저 기본 설정에 따라 시작 템플릿 또는 시작 구성 지침을 선택합니다. 시작 템플릿 또는 시작 구성을 생성할 때는 사용할 Amazon Machine Image(AMI)의 ID, 인스턴스 유형, 키 페어, 블록 디바이스 매핑 등의 정보를 포함합니다. 필수 입력 사항은 인스턴스 유형과 AMI입니다.
- 그런 다음 Auto Scaling 그룹을 구성하고 로드 밸런서를 연결합니다.
- 마지막으로 Auto Scaling 그룹을 생성하고 로드 밸런서가 연결되어 있는지 확인합니다.

목차

- [사전 조건 \(p. 19\)](#)
- [애플리케이션 배포\(콘솔\) \(p. 20\)](#)
- [애플리케이션 배포\(AWS CLI\) \(p. 24\)](#)
- [다음 단계 \(p. 25\)](#)
- [AWS 리소스 정리 \(p. 25\)](#)

사전 조건

- 사용할 기존 VPC 및 보안 그룹이 있습니다. 기본 VPC를 사용하도록 선택하는 경우 로드 밸런서를 시작한 후 로드 밸런서의 보안 그룹에 적절한 인바운드 규칙이 있는지 확인해야 합니다. 기본 규칙은 어떠한 인바운드 트래픽도 허용하지 않습니다.
- 사용하도록 만든 로드 밸런서가 있습니다. Auto Scaling 그룹에 사용할 로드 밸런서에 대해 동일한 가용 영역을 선택해야 합니다.
- (선택 사항) 필요한 AWS에 대한 액세스를 애플리케이션에 부여하는 IAM 역할이 있습니다.

- (선택 사항) Amazon EC2 인스턴스의 소스 템플릿으로 정의된 Amazon Machine Image(AMI)가 있습니다. 지금 생성하려면 인스턴스를 시작합니다. IAM 역할(만든 경우)을 지정하고 필요한 구성 스크립트를 사용자 데이터로 지정합니다. 인스턴스에 연결하여 인스턴스를 사용자 지정합니다. 예를 들어, 소프트웨어와 애플리케이션을 설치할 수 있습니다. 인스턴스에서 애플리케이션을 테스트하여 인스턴스가 올바르게 구성되었는지 확인합니다. 이 업데이트된 구성을 사용자 지정 AMI로 저장합니다. 나중에 필요하지 않을 경우 인스턴스를 종료할 수 있습니다. 이 새로운 사용자 지정 AMI에서 인스턴스를 시작하면 해당 AMI를 만들 때 지정한 사용자 정의 값을 포함하게 됩니다.

애플리케이션 배포(콘솔)

다음 단원에서는 애플리케이션 배포 프로세스를 단계별로 설명합니다.

작업

- 시작 템플릿 생성 또는 선택 (p. 20)
- 시작 구성 생성 또는 선택 (p. 21)
- Auto Scaling 그룹 생성 (p. 22)
- (선택 사항) 로드 밸런서가 Auto Scaling 그룹에 연결되었는지 확인합니다. (p. 23)

시작 템플릿 생성 또는 선택

사용하려는 시작 템플릿이 이미 있으면 다음 절차를 사용하여 해당 구성을 선택합니다.

기존 시작 템플릿을 선택하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 로드 밸런서를 만들 때 사용한 AWS 리전을 선택합니다.
3. 탐색 창에서 시작 템플릿을 선택합니다.
4. 시작 템플릿을 선택합니다.
5. 작업, Auto Scaling 그룹 생성을 선택합니다.

또는 다음 절차를 사용하여 새 시작 템플릿을 생성합니다.

시작 템플릿을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 로드 밸런서를 만들 때 사용한 AWS 리전을 선택합니다.
3. 탐색 창에서 시작 템플릿을 선택합니다.
4. 시작 템플릿 생성을 선택합니다.
5. 시작 템플릿의 이름과 설명을 제공합니다. 시작 템플릿의 이름은 3~125자여야 하고, (), / _ 문자를 포함할 수 있습니다.
6. [AMI ID]에서 인스턴스에 대한 AMI의 ID를 입력합니다.
7. [Instance type]에서 지정한 AMI와 호환되는 인스턴스에 대한 하드웨어 구성을 선택합니다.
8. (선택 사항) [Key pair name]에서 인스턴스와 연결할 때 사용할 키 페어 이름을 입력합니다.
9. [Network Interfaces]에서 다음을 수행합니다.
 - a. [Add network interface]를 선택합니다.
 - b. (선택 사항) 기본이 아닌 VPC에 있는 인스턴스와 연결하려면 [Auto-assign public IP]에서 [Enable]을 선택합니다.
 - c. [Security group ID]에서 인스턴스에 대한 보안 그룹을 지정합니다.

- d. 종료 시 삭제에 대해, Auto Scaling 그룹이 축소되고 네트워크 인터페이스가 연결된 인스턴스를 종료할 때 네트워크 인터페이스를 삭제할지 선택합니다.
10. (선택 사항) 인스턴스에 자격 증명을 안전하게 배포하려면 고급 세부 정보, IAM 인스턴스 프로파일에서 IAM 역할의 Amazon 리소스 이름(ARN)을 입력합니다.
11. (선택 사항) 인스턴스에 대한 사용자 데이터 또는 구성 스크립트를 지정하려면 [Advanced details], [User data]에 붙여넣습니다.
12. [Create launch template]를 선택합니다.
13. 확인 페이지에서 Auto Scaling 그룹 생성을 선택합니다.

시작 구성 생성 또는 선택

사용하려는 시작 구성이 이미 있으면 다음 절차를 사용하여 해당 구성을 선택합니다.

기존 시작 구성을 선택하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 로드 밸런서를 만들 때 사용한 AWS 리전을 선택합니다.
3. 탐색 창의 Auto Scaling에서 Auto Scaling 그룹을 선택합니다.
4. 다음 페이지에서 Auto Scaling 그룹 생성을 선택합니다.
5. Auto Scaling 그룹 생성 페이지에서 시작 구성을 선택하고 기존 시작 구성을 선택한 후 다음 단계를 선택합니다.

또는 다음 절차를 사용하여 새 시작 구성을 생성합니다.

시작 구성을 생성하는 방법

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 로드 밸런서를 만들 때 사용한 AWS 리전을 선택합니다.
3. 탐색 창의 Auto Scaling에서 Auto Scaling 그룹을 선택합니다.
4. 다음 페이지에서 Auto Scaling 그룹 생성을 선택합니다.
5. Auto Scaling 그룹 생성 페이지에서, Launch Configuration, Create a new launch configuration을 선택한 후 다음 단계를 선택합니다.
6. [Choose AMI] 페이지에서 사용자 지정 AMI를 선택합니다.
7. [Choose Instance Type] 페이지에서 인스턴스의 하드웨어 구성을 선택한 다음 [Next: Configure details]를 선택합니다.
8. [Configure Details] 페이지에서 다음을 수행합니다.
 - a. Name에 시작 구성의 이름을 입력합니다.
 - b. (선택 사항) EC2 인스턴스에 자격 증명을 안전하게 배포하려면 IAM 역할을 선택합니다.
 - c. (선택 사항) 기본이 아닌 VPC에 있는 인스턴스에 연결해야 하는 경우, [Advanced Details], [IP Address Type]에서 [Assign a public IP address to every instance]를 선택합니다.
 - d. (선택 사항) 인스턴스에 대해 사용자 데이터 또는 구성 스크립트를 지정하려면 [Advanced Details], [User data]에 구성 스크립트를 붙여넣습니다.
 - e. [Skip to review]를 선택합니다.
9. [Review] 페이지에서 [Edit security groups]를 선택합니다. 지침을 따라 기존 보안 그룹을 선택한 다음 [Review]를 선택합니다.
10. [Review] 페이지에서 [Create launch configuration]을 선택합니다.
11. [Select an existing key pair or create a new key pair] 페이지에서 나열된 옵션 중 하나를 선택합니다. 승인 확인란을 선택한 다음 [Create launch configuration]을 선택합니다.

Warning

인스턴스에 연결해야 하는 경우 [Proceed without a key pair]를 선택하지 마십시오.

위의 과정을 완료하였다면 Auto Scaling 그룹을 생성하는 마법사를 실행할 준비가 된 것입니다.

Auto Scaling 그룹 생성

시작 템플릿 또는 시작 구성을 선택 또는 생성한 후 중단한 시점부터 계속하려면 다음 절차를 따르십시오.

Note

Amazon EC2 Auto Scaling에서 Auto Scaling 그룹 인터페이스가 변경되었습니다. 기본적으로 기존 사용자 인터페이스가 표시되지만 새 사용자 인터페이스로 전환할 수 있습니다. 이 섹션에는 두 가지 모두에 대한 단계가 포함되어 있습니다.
새 사용자 인터페이스로 전환하려면 Auto Scaling 그룹 페이지 상단에 있는 배너에서 Go to the new console(새 콘솔로 이동)을 선택합니다.

Auto Scaling 그룹을 생성하려면(새 콘솔)

1. 새 콘솔의 Auto Scaling 그룹 페이지에서 Create an Auto Scaling group(Auto Scaling 그룹 생성)을 선택합니다.
2. Choose launch template or configuration(시작 템플릿 또는 구성 선택) 페이지의 Auto Scaling group name(Auto Scaling 그룹 이름)에 Auto Scaling 그룹의 이름을 입력합니다.
3. 시작 템플릿에서 다음 옵션 중 하나를 선택합니다.
 - a. [시작 템플릿만 해당] 생성한 시작 템플릿을 선택한 다음 확장 시 Auto Scaling 그룹에서 시작 템플릿의 기본 버전, 최신 버전 또는 특정 버전 중 어느 버전을 사용할지 선택합니다.
 - b. [시작 구성에만 해당] Switch to launch configuration(시작 구성으로 전환)을 선택한 다음, 생성한 시작 구성을 선택합니다.
4. [Next]를 선택합니다.

네트워크 설정을 구성할 수 있게 해주고 여러 인스턴스 유형에서 온디맨드 인스턴스 및 스팟 인스턴스에 대한 다각화를 시작할 수 있는 옵션을 제공하는 Configure settings(설정 구성) 페이지가 나타납니다(시작 템플릿을 선택한 경우).
5. [시작 템플릿만 해당] Purchase options and instance types(구매 옵션 및 인스턴스 유형)를 Adhere to the launch template(시작 템플릿 준수)으로 설정한 상태를 유지하여 시작 템플릿에서 지정된 EC2 인스턴스 유형 및 구매 옵션을 사용합니다.
6. 네트워크의 경우 로드 밸런서 및 보안 그룹에 사용한 VPC를 선택합니다.
7. 서브넷의 경우 포함할 각 가용 영역의 서브넷을 하나 이상 선택합니다.
8. [Next]를 선택합니다.
9. Specify load balancing and health checks(로드 밸런싱 및 상태 확인 지정) 페이지의 로드 밸런싱에서 Enable load balancing(로드 밸런싱 활성화)을 선택합니다.
10. 다음 중 하나를 수행하십시오.
 - a. Application Load Balancer 또는 Network Load Balancer를 선택한 다음 로드 밸런서 대상 그룹을 선택합니다.
 - b. Classic Load Balancer를 선택한 다음 로드 밸런서를 선택합니다.
11. (선택 사항) Elastic Load Balancing 상태 확인을 사용하려면 상태 확인에 대해 Health check type(상태 확인 유형)에서 ELB를 선택합니다.
12. Auto Scaling 그룹 구성을 완료했으면 Skip to review(검토로 이동)를 선택합니다.
13. 검토 페이지에서 Auto Scaling 그룹의 세부 정보를 검토합니다. [Edit]를 선택하여 변경할 수 있습니다. 모두 마쳤으면 Auto Scaling 그룹 생성을 선택합니다.

Auto Scaling 그룹을 생성하려면(이전 콘솔)

1. [Configure Auto Scaling group details] 페이지에서 다음을 수행합니다.
 - a. 그룹 이름에서 Auto Scaling 그룹의 이름을 입력합니다.
 - b. [시작 템플릿] 시작 템플릿 버전에서, Auto Scaling 그룹이 확장 시 시작 템플릿의 기본 버전을 사용할지, 최신 버전을 사용할지 또는 특정 버전을 사용할지를 선택합니다.
 - c. [시작 템플릿] 템플릿 구성에서 시작 템플릿 준수를 선택하여 시작 템플릿에 지정된 EC2 인스턴스 유형과 구입 옵션을 사용합니다.
 - d. 그룹 크기에 Auto Scaling 그룹의 초기 인스턴스 수를 입력합니다.
 - e. 시작 구성 또는 템플릿에서 VPC가 필요한 인스턴스 유형(예: T2 인스턴스)을 선택한 경우, [Network]에 VPC를 선택해야 합니다. 그렇지 않고 계정에서 EC2-Classical을 지원하고 VPC가 필요하지 않은 인스턴스 유형을 선택한 경우, [Launch into EC2-Classical] 또는 [VPC] 중 하나를 선택할 수 있습니다.
 - f. 이전 단계에서 [VPC]를 선택한 경우 [Subnet]에서 하나 이상의 서브넷을 선택합니다. [EC2-Classical]을 선택한 경우 [Availability Zone(s)]에서 하나 이상의 가용 영역을 선택합니다.
 - g. 고급 세부 정보에서 Receive traffic from Elastic Load Balancer(s)(Elastic Load Balancer로부터 트래픽 수신)을 선택한 후 다음 중 하나를 수행합니다.
 - [Classic Load Balancers] [Load Balancers]에서 로드 밸런서를 선택합니다.
 - [애플리케이션/네트워크 로드 밸런서] [Target Groups]에서 대상 그룹을 선택합니다.
 - h. (선택 사항) Elastic Load Balancing 상태 확인을 사용하려면 고급 세부 정보, 상태 확인 유형에서 ELB를 선택합니다.
 - i. Next: Configure scaling policies(다음: 조정 정책 구성)를 선택합니다.
2. Configure scaling policies(조정 정책 구성) 페이지에서 Keep this group at its initial size(이 그룹을 초기 크기로 유지)를 선택합니다.
3. Auto Scaling 그룹 구성을 완료했으면 검토를 선택합니다.
4. Auto Scaling 그룹의 세부 정보를 검토합니다. [Edit]를 선택하여 변경할 수 있습니다. 모두 마쳤으면 Auto Scaling 그룹 생성을 선택합니다.

로드 밸런서가 연결된 Auto Scaling 그룹을 생성한 후, 로드 밸런서는 온라인 상태가 될 때 새 인스턴스를 자동으로 등록합니다. 이 시점에서는 인스턴스가 하나만 있으므로 등록할 사항이 많이 없습니다. 그러나 이제 원하는 그룹의 용량을 업데이트하여 인스턴스를 더 추가할 수 있습니다. 단계별 지침은 [수동 조정 \(p. 90\)](#)을 참조하십시오.

(선택 사항) 로드 밸런서가 Auto Scaling 그룹에 연결되었는지 확인합니다.

로드 밸런서가 Auto Scaling 그룹에 연결되었는지 확인하려면(새 콘솔)

1. Auto Scaling 그룹 옆에 있는 확인란을 선택합니다.

이전 사용자 인터페이스가 표시되는 경우 콘솔 페이지 상단에 표시되는 배너에서 Go to the new console(새 콘솔로 이동)을 선택하여 새 사용자 인터페이스로 전환합니다.
2. 세부 정보 탭에서 로드 밸런싱은 연결된 로드 밸런서 대상 그룹 또는 Classic Load Balancer를 표시합니다.
3. 활동 탭에서, Activity history(활동 기록)의 상태 열에 Auto Scaling 그룹이 성공적으로 인스턴스를 시작했는지 여부가 표시됩니다.
4. 인스턴스 관리 탭의 인스턴스에서 현재 실행 중인 인스턴스의 상태를 볼 수 있습니다.

수명 주기 열에는 인스턴스의 상태가 표시됩니다. 인스턴스가 트래픽을 수신할 준비가 되면 상태가 InService로 전환됩니다.

상태 열에는 해당 인스턴스에 대한 상태 확인 결과가 표시됩니다.

로드 밸런서가 Auto Scaling 그룹에 연결되었는지 확인하려면(기존 콘솔)

1. Auto Scaling 그룹을 선택합니다.
2. 세부 정보 탭의 로드 밸런서에 연결된 모든 로드 밸런서가 표시되고, 대상 그룹에 연결된 모든 대상 그룹이 표시됩니다.
3. 활동 기록 탭의 상태 열에 Auto Scaling 인스턴스의 상태가 표시됩니다. 인스턴스가 시작 중이면 해당 상태가 `In progress`가 됩니다. 인스턴스가 시작되면 상태가 `Successful`로 변경됩니다.
4. 인스턴스 탭의 수명 주기 열에 Auto Scaling 인스턴스의 상태가 표시됩니다. 인스턴스가 트래픽을 수신할 준비가 되면 상태가 `InService`로 전환됩니다.

[Health Status] 열에는 해당 인스턴스에 대한 상태 확인 결과가 표시됩니다.

애플리케이션 배포(AWS CLI)

다음 단원에서는 애플리케이션 배포 프로세스를 단계별로 설명합니다.

작업

- 시작 템플릿 생성 (p. 24)
- 시작 구성 생성 (p. 24)
- 로드 밸런서가 있는 Auto Scaling 그룹 생성 (p. 25)

시작 템플릿 생성

사용하려는 시작 템플릿이 이미 있으면 이 단계를 건너뛰십시오.

시작 템플릿을 생성하려면

다음 `create-launch-template` 명령을 사용하고 시작 템플릿 생성을 위한 정보를 포함하는 JSON 파일을 전달합니다. 기본이 아닌 VPC의 인스턴스에 공용 IP 주소를 할당하려면 네트워크 인터페이스 속성을 추가하고 표시된 대로 `"NetworkInterfaces": {"AssociatePublicIpAddress": true}`로 설정하십시오.

```
aws ec2 create-launch-template --launch-template-name my-launch-template --version-  
description my-version-description \  
--launch-template-data '{"NetworkInterfaces":  
[{"DeviceIndex":0,"AssociatePublicIpAddress":true,"Groups":  
["sg-903004f8"],"DeleteOnTermination":true}], "ImageId":"ami-01e24be29428c15b2", "InstanceType":"t2.micro"
```

시작 구성 생성

사용하려는 시작 구성이 이미 있으면 이 단계를 건너뛰십시오.

시작 구성을 생성하려면

다음 `create-launch-configuration` 명령을 사용합니다. 기본이 아닌 VPC의 인스턴스에 공용 IP 주소를 할당하려면 표시된 대로 `--associate-public-ip-address` 옵션을 사용합니다.

```
aws autoscaling create-launch-configuration --launch-configuration-name my-launch-config \  
--image-id ami-01e24be29428c15b2 --instance-type t2.micro --associate-public-ip-address \  
--security-groups sg-eb2af88e
```

로드 밸런서가 있는 Auto Scaling 그룹 생성

그룹을 생성할 때 Auto Scaling 그룹에 기존 로드 밸런서를 연결할 수 있습니다. 실행 구성 또는 실행 템플릿을 사용하여 Auto Scaling 그룹이 시작하는 인스턴스를 자동으로 구성할 수 있습니다.

Classic Load Balancer가 연결된 Auto Scaling 그룹을 만들려면

다음 `create-auto-scaling-group` 명령과 `--load-balancer-names` 옵션을 사용하여 Classic Load Balancer가 연결된 Auto Scaling 그룹을 생성합니다.

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg \
  --launch-configuration-name my-launch-config \
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782" \
  --load-balancer-names "my-load-balancer" \
  --max-size 5 --min-size 1 --desired-capacity 2
```

Application Load Balancer 또는 Network Load Balancer에 대해 대상 그룹이 연결된 Auto Scaling 그룹을 만들려면

다음 `create-auto-scaling-group` 명령과 `--target-group-arns` 옵션을 사용하여 대상 그룹이 연결된 Auto Scaling 그룹을 생성합니다.

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg \
  --launch-template "LaunchTemplateName=my-launch-template,Version=1" \
  --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782" \
  --target-group-arns "arn:aws:elasticloadbalancing:region:123456789012:targetgroup/my-
  targets/1234567890123456" \
  --max-size 5 --min-size 1 --desired-capacity 2
```

다음 단계

이 자습서를 완료했으므로 다음 내용을 자세히 알아볼 수 있습니다.

- 조정 정책을 사용하도록 Auto Scaling 그룹을 구성할 수 있습니다. 이렇게 하면 그룹이 자동으로 Amazon EC2 용량을 조정하여 애플리케이션이 수신하는 트래픽 양의 변화를 처리할 수 있습니다. 자세한 내용은 [동적 조정 \(p. 102\)](#) 단원을 참조하십시오.
- 자동 조정 일정을 만드는 방법을 학습하려는 경우 [예약된 조정 \(p. 99\)](#)에서는 일회성 또는 반복 조정 작업을 사용하여 Auto Scaling 그룹을 확장하도록 예약하는 방법을 소개합니다.
- Elastic Load Balancing 상태 확인을 사용하도록 Auto Scaling 그룹을 구성할 수 있습니다. 로드 밸런서 상태 확인을 활성화한 경우 인스턴스가 상태 확인에 실패하면 Auto Scaling 그룹은 해당 인스턴스를 비정상적으로 간주하고 교체합니다. 자세한 내용은 [ELB 상태 확인 추가 \(p. 70\)](#) 단원을 참조하십시오.
- 동일한 AWS 리전의 추가 가용 영역으로 애플리케이션을 확장하여 서비스 중단 시 내결함성을 높일 수 있습니다. 자세한 내용은 [가용 영역 추가 \(p. 71\)](#) 단원을 참조하십시오.

AWS 리소스 정리

이제 자습서를 성공적으로 완료했습니다. 사용하지 않는 리소스에 대해 계정에 불필요한 요금이 청구되지 않도록 하려면 이 자습서에서 작성한 리소스를 정리해야 합니다. 단계별 지침은 [Auto Scaling 인프라 삭제 \(p. 85\)](#)를 참조하십시오.

시작 템플릿

시작 템플릿은 인스턴스 구성 정보를 지정한다는 점에서 [시작 구성 \(p. 34\)](#)과 동일합니다. Amazon 머신 이미지(AMI)의 ID, 인스턴스 유형, 키 페어, 보안 그룹 및 EC2 인스턴스를 시작하는 데 사용할 기타 파라미터 등이 포함됩니다. 하지만 시작 구성 대신에 시작 템플릿을 정의하면 여러 템플릿 버전을 사용할 수 있습니다. 버전 관리를 사용하면 전체 파라미터 세트의 하위 세트를 만든 후 이를 재사용하여 여러 템플릿 또는 템플릿 버전을 만들 수 있습니다. 예를 들어 태그 또는 네트워크 구성 등의 일반적인 구성 파라미터를 정의하는 기본 템플릿을 생성하고, 동일한 템플릿의 다른 버전의 일부로 다른 파라미터를 지정할 수 있습니다.

Amazon EC2의 최신 기능을 사용하려면 시작 구성 대신 시작 템플릿을 사용하는 것이 좋습니다.

시작 템플릿의 경우 온디맨드 인스턴스 및 스팟 인스턴스를 모두 사용하여 여러 인스턴스 유형에 용량을 프로비저닝하여 원하는 규모, 성능 및 비용을 달성할 수도 있습니다. 자세한 내용은 [여러 인스턴스 유형 및 구매 옵션이 포함된 Auto Scaling 그룹 \(p. 45\)](#) 단원을 참조하십시오.

현재 시작 구성을 사용하는 경우, 시작 구성을 사용하여 생성된 Auto Scaling 그룹을 업데이트할 때 시작 템플릿을 지정할 수 있습니다.

Auto Scaling 그룹에 사용할 시작 템플릿을 생성하려면 처음부터 템플릿을 생성하거나, 기존 템플릿의 새로운 버전을 생성하거나, 시작 구성, 실행 중인 인스턴스 또는 다른 템플릿의 파라미터를 복사하면 됩니다.

다음 주제에서는 Auto Scaling 그룹에서 사용할 시작 템플릿 생성 및 사용에 대한 가장 일반적인 절차를 설명합니다. 시작 템플릿에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [시작 템플릿](#) 단원을 참조하십시오.

목차

- [Auto Scaling 그룹을 위한 시작 템플릿 만들기 \(p. 26\)](#)
- [시작 구성을 시작 템플릿에 복사 \(p. 32\)](#)
- [시작 구성을 시작 템플릿으로 교체 \(p. 32\)](#)

Auto Scaling 그룹을 위한 시작 템플릿 만들기

시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하려면 먼저 Amazon 머신 이미지(AMI)의 ID 및 인스턴스 유형 등 EC2 인스턴스를 시작하는 데 필요한 파라미터를 포함하는 시작 템플릿을 생성해야 합니다.

다음 절차는 새로운 시작 템플릿을 만드는 절차입니다. 새로운 템플릿은 처음부터 정의된 파라미터를 사용하거나, 기존 시작 템플릿을 사용합니다. 시작 템플릿을 생성한 후 [시작 템플릿을 사용하여 Auto Scaling 그룹 생성하기 \(p. 54\)](#)의 지침에 따라 Auto Scaling 그룹을 생성할 수 있습니다.

사전 요구 사항

필요한 IAM 권한에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [시작 템플릿 사용 제어](#)를 참조하십시오. 시작 템플릿을 사용하기 위해 Auto Scaling 그룹을 생성하고 업데이트하는 데 필요한 권한에 대한 자세한 내용은 [Amazon EC2 Auto Scaling 자격 증명 기반 정책 예제 \(p. 179\)](#) 단원을 참조하십시오.

고려 사항

Auto Scaling 그룹에 사용할 시작 템플릿을 생성할 때 다음 사항에 유의하십시오.

- 시작 템플릿을 사용하면 인스턴스 유형 중 하나를 시작하거나 인스턴스 유형과 온디맨드 및 스팟 구매 옵션의 조합을 시작할 수 있습니다. 자세한 내용은 [여러 인스턴스 유형 및 구매 옵션이 포함된 Auto Scaling 그룹 \(p. 45\)](#) 단원을 참조하십시오. 이러한 조합으로 인스턴스를 시작하는 것은 지원되지 않습니다.

- 추가 세부 정보에 스팟 인스턴스 요청을 지정하는 경우
- EC2-Classic
- 템플릿에 대해 네트워크 유형(VPC 또는 EC2-Classic), 서브넷 및 가용 영역을 구성할 경우, Auto Scaling 그룹에서 지정한 설정이 우선하므로 이러한 설정은 무시됩니다.
- 네트워크 인터페이스를 지정할 경우, 템플릿의 보안 그룹 섹션이 아니라 네트워크 인터페이스의 일부로 보안 그룹을 구성해야 합니다.
- 여러 개의 네트워크 인터페이스를 지정할 수 없습니다.
- 특정 프라이빗 IP 주소를 지정할 수 없습니다. 인스턴스가 시작되면 인스턴스가 시작되는 서브넷의 CIDR 범위에서 프라이빗 주소가 할당됩니다. VPC 또는 서브넷의 CIDR 범위 지정에 관한 자세한 내용은 [Amazon VPC 사용 설명서](#)를 참조하십시오.
- 사용할 기존 네트워크 인터페이스를 지정하려면 디바이스 인덱스가 0(eth0)이어야 합니다. 이 시나리오에서는 Auto Scaling 그룹 생성을 위해 CLI 또는 API를 사용해야 합니다. CLI `create-auto-scaling-group` 명령 또는 API `CreateAutoScalingGroup` 작업을 사용하여 그룹을 생성하는 경우, 서브넷(VPC 영역 식별자) 파라미터 대신 가용성 영역 파라미터를 지정해야 합니다.
- 호스트 배치 선호도를 사용하거나 호스트 ID를 선택하여 특정 호스트를 대상으로 지정할 수 없습니다.
- 호스트 리소스 그룹을 지정한 경우에만 호스트 테넌시(전용 호스트) 지원을 사용할 수 있습니다. 자세한 내용은 AWS License Manager 사용자 안내서의 [호스트 리소스 그룹](#)을 참조하십시오. 라이선스 구성 연결을 기반으로 하는 각 AMI는 한 번에 하나의 호스트 리소스 그룹에만 매핑할 수 있습니다.

Auto Scaling 그룹의 새로운 시작 템플릿을 생성하려면(새 EC2 콘솔)

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 이름을 입력하고 초기 버전에 대한 설명을 제공하십시오.
4. 시작 템플릿 내용 아래에서 인스턴스 사양으로 사용할 각 필수 필드와 선택적 필드를 작성합니다.
 - a. (필수) Amazon machine image(AMI): 인스턴스를 시작할 AMI의 ID를 선택합니다. 사용 가능한 모든 AMI를 검색하거나 빠른 시작 목록에서 자주 사용하는 AMI중 하나를 선택할 수 있습니다. 필요한 AMI가 표시되지 않으면 [적합한 AMI를 찾아](#) 해당 ID를 기록해 두고 사용자 지정 값으로 지정할 수 있습니다.
 - b. (권장) 인스턴스 유형: [인스턴스 유형](#)을 선택합니다. 인스턴스 유형이 지정한 AMI와 호환되는지 확인합니다.
 - c. (권장) Key pair(login)(키 페어(로그인)): 인스턴스와 연결할 때 사용할 [키 페어](#)를 지정합니다.
5. Network settings(네트워크 설정)에서 하나 이상의 VPC 또는 EC2-Classic 보안 그룹을 구성하려면 다음 단계를 완료합니다.
 - a. Networking platform(네트워킹 플랫폼): VPC로 인스턴스를 시작할 것인지, EC2-Classic으로 인스턴스를 시작할 것인지(해당되는 경우) 선택합니다. 그러나 Amazon EC2 Auto Scaling의 경우 Auto Scaling 그룹 설정이 우선하기 때문에 시작 템플릿의 네트워크 유형과 가용 영역 설정이 무시됩니다.
 - b. 보안 그룹: [보안 그룹](#)을 하나 이상 선택하거나, 하나 이상의 VPC 보안 그룹을 네트워크 인터페이스의 일부로 구성하도록 비워둡니다. 두 곳 모두에서 보안 그룹을 지정할 수 없습니다. EC2-Classic을 사용하는 경우 EC2-Classic용으로 특별히 생성된 보안 그룹을 사용해야 합니다.
6. 실행된 각 인스턴스에는 연관된 루트 디바이스 볼륨(Amazon EBS 볼륨 또는 인스턴스 스토어 볼륨)이 있습니다. 실행될 때 인스턴스에 연결할 추가 EBS 볼륨 또는 인스턴스 스토어 볼륨을 지정할 수 있습니다. AMI에서 지정한 볼륨 외에도 인스턴스에 연결할 볼륨을 지정하기로 선택한 경우, 스토리지(볼륨) 아래에서 새 볼륨 추가를 선택하고 다음 정보를 입력합니다.
 - a. 볼륨 유형: 볼륨 유형은 선택한 인스턴스 유형에 따라 달라집니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2 인스턴스 스토어](#) 및 [Amazon EBS 볼륨](#)을 참조하십시오.
 - b. 디바이스 이름: 볼륨의 디바이스 이름을 지정합니다.

- c. 스냅샷: 볼륨이 생성되는 서버넷의 ID를 입력합니다.
- d. 크기(GiB): Amazon EBS 지원 볼륨에서 스토리지 크기를 지정합니다. 스냅샷에서 볼륨을 생성하고 있고 볼륨 크기를 지정하지 않은 경우에는 스냅샷 크기가 기본값이 됩니다.
- e. 볼륨 유형: Amazon EBS 볼륨의 경우 [볼륨 유형](#)을 선택합니다.
- f. IOPS: 프로비저닝된 IOPS SSD 볼륨의 경우 볼륨이 지원해야 하는 최대 IOPS(초당 입/출력 작업) 수를 입력합니다.
- g. 종료 시 삭제: Amazon EBS 볼륨의 경우 연결된 인스턴스가 종료되면 볼륨을 삭제할지 여부를 선택합니다.
- h. 암호화: Amazon EBS 볼륨의 암호화 상태를 변경하려면 예를 선택합니다. 이 파라미터를 설정하는 데 따르는 기본 효과는 아래 표에 설명된 것과 같이 선택한 볼륨 소스에 따라 달라집니다. 어느 경우든 지정된 CMK에 대한 사용 권한이 있어야 합니다. 암호화된 볼륨 지정에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EBS 암호화](#)를 참조하십시오.

암호화 결과

Encrypted 파라미터 설정	볼륨 소스	기본 암호화 상태	참고
아니요	새(빈) 볼륨	암호화되지 않음*	해당 사항 없음
	암호화되지 않은 소유 스냅샷	암호화되지 않음*	
	암호화된 소유 스냅샷	동일한 키로 암호화됨	
	암호화되지 않은 공유 스냅샷	암호화되지 않음*	
	암호화된 공유 스냅샷	기본 CMK로 암호화됨	
예	새 볼륨	기본 CMK로 암호화됨	기본값이 아닌 CMK를 사용하려면 키 파라미터에 값을 지정합니다.
	암호화되지 않은 소유 스냅샷	기본 CMK로 암호화됨	
	암호화된 소유 스냅샷	동일한 키로 암호화됨	
	암호화되지 않은 공유 스냅샷	기본 CMK로 암호화됨	
	암호화된 공유 스냅샷	기본 CMK로 암호화됨	

* 기본 암호화가 사용 설정된 경우 새로 생성된 모든 볼륨(Encrypted 매개 변수가 Yes로 설정되었는지 여부)은 기본 CMK를 사용하여 암호화됩니다. Encrypted 및 Key 매개 변수를 모두 설정하면 기본값이 아닌 CMK를 지정할 수 있습니다.

- i. (선택 사항) 키: 이전 단계에서 예를 선택한 경우, 볼륨을 암호화할 때 사용할 CMK(고객 마스터 키)를 입력하십시오. 앞에서 AWS Key Management Service를 사용하여 생성한 임의의 CMK를 입력합니다. 액세스 권한이 있는 키의 전체 ARN을 붙여넣을 수 있습니다. 자세한 내용은 [AWS Key Management Service Developer Guide](#) 및 [암호화된 볼륨에 사용할 경우 필요한 CMK 키 정책 \(p. 193\)](#) 주제를 참조합니다. 참고: Amazon EBS는 비대칭 CMK를 지원하지 않습니다.

Note

암호화 파라미터를 설정하지 않고 CMK를 입력하면 오류가 발생합니다.

- 7. 인스턴스 태그에서 키와 값의 조합을 제공하여 태그를 지정합니다. 인스턴스 또는 볼륨 또는 이 둘 모두에 태그를 지정할 수 있습니다.
- 8. 네트워크 인터페이스에서 네트워크 인터페이스 추가를 선택하고 다음 정보를 입력합니다. VPC에서 인스턴스를 시작할 때 기본 네트워크 인터페이스(eth0)라고 하는 기본 네트워크 인터페이스를 지정할 수 있습니다. 네트워크 인터페이스는 하나만 지정할 수 있습니다. 다음 필드에 주의하십시오.
 - a. 디바이스: eth0을 디바이스 이름으로 지정합니다(디바이스 인덱스가 0인 디바이스).

- b. 네트워크 인터페이스: 인스턴스를 시작할 때 AWS가 새 네트워크 인터페이스를 생성하도록 비워두거나, 기존 네트워크 인터페이스의 ID를 입력합니다. ID를 지정할 경우 Auto Scaling 그룹이 인스턴스 하나로 제한됩니다.
 - c. 설명: 설명하는 이름을 입력합니다.
 - d. 서브넷: 서브넷을 지정할 수 있지만 Amazon EC2 Auto Scaling의 경우 Auto Scaling 그룹 설정이 우선하기 때문에 무시됩니다.
 - e. 퍼블릭 IP 자동 할당: 디바이스 인덱스가 eth0인 네트워크 인터페이스에 퍼블릭 IP 주소를 자동으로 할당할 것인지 여부를 지정합니다. 이 설정은 새로운 단일 네트워크 인터페이스에서만 활성화될 수 있습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스 시작 시 퍼블릭 IPv4 주소 할당](#)을 참조하십시오.
 - f. 보안 그룹 ID: 기본 네트워크 인터페이스(eth0)를 할당할 [보안 그룹](#) 하나 이상의 ID를 입력합니다. Auto Scaling 그룹이 인스턴스를 시작하는 VPC에 각 보안 그룹을 구성해야 합니다. 항목을 쉼표로 구분합니다.
 - g. 종료 시 삭제: Auto Scaling 그룹이 축소되고 네트워크 인터페이스가 연결된 인스턴스를 종료할 때 네트워크 인터페이스를 삭제할지 선택합니다.
9. 고급 세부 정보에서 필드를 볼 수 있도록 섹션을 확장하고 인스턴스를 위한 추가 파라미터를 지정합니다.
- 구매 옵션: 스팟 인스턴스를 요청하고 인스턴스 시간당 지불하고자 하는 최고 가격을 지정할 수 있습니다. 이를 Auto Scaling 그룹과 함께 사용하려면 종료 날짜가 없는 일회성 요청을 지정해야 합니다. 자세한 내용은 [Auto Scaling 그룹에서 스팟 인스턴스 시작 \(p. 73\)](#) 단원을 참조하십시오.

Important

Auto Scaling 그룹을 구성할 때 여러 인스턴스 유형과 구매 옵션을 지정할 경우, 이러한 필드를 비워둡니다. 자세한 내용은 [여러 인스턴스 유형 및 구매 옵션이 포함된 Auto Scaling 그룹 \(p. 45\)](#) 단원을 참조하십시오.

- IAM 인스턴스 프로파일: 인스턴스에 연결할 AWS Identity and Access Management(IAM) 인스턴스 프로파일을 지정합니다. 자세한 내용은 [Amazon EC2 인스턴스에서 실행되는 애플리케이션의 IAM 역할 \(p. 192\)](#) 단원을 참조하십시오.
- 종료 방식: 이 필드는 Amazon EC2 Auto Scaling의 경우 무시되므로 비워둘 수 있습니다. Amazon EC2 Auto Scaling의 기본 방식은 인스턴스를 종료하는 것입니다.
- Stop - Hibernate behavior(중지 - 수면 동작): 이 필드는 Amazon EC2 Auto Scaling의 경우 무시되므로 비워둘 수 있습니다. Amazon EC2 Auto Scaling의 기본 방식은 인스턴스를 종료하는 것입니다.
- 종료 방지: Auto Scaling 그룹에서 축소하는 경우 추가 종료 방지를 제공하지만 Amazon EC2 Auto Scaling에서 무시됩니다. 축소 시 Auto Scaling 그룹이 특정 인스턴스를 종료할지 여부를 제어하려면 [인스턴스 축소 보호 \(p. 128\)](#)를 사용합니다.
- 세부 CloudWatch 모니터링: Amazon CloudWatch를 사용하여 인스턴스에 대한 세부 모니터링을 활성화할지 여부를 선택합니다. 추가 요금이 발생합니다. 자세한 내용은 [Amazon CloudWatch를 사용하여 Auto Scaling 그룹 및 인스턴스 모니터링 \(p. 160\)](#) 단원을 참조하십시오.
- T2/T3 Unlimited(T2/T3 무제한): (T2 및 T3 인스턴스에만 유효) 애플리케이션이 필요한 시간 만큼 기존 이상으로 버스트할 수 있도록 할지 여부를 선택합니다. 추가 요금이 적용될 수 있습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [Auto Scaling 그룹을 사용하여 성능 순간 확장 가능 인스턴스를 무제한으로 시작](#)을 참조하십시오.
- 배치 그룹 이름: 인스턴스를 시작할 배치 그룹을 지정합니다. 하나의 배치 그룹에서 모든 인스턴스 유형을 시작할 수 있는 것은 아닙니다. 다른 배치 그룹을 지정하는 CLI 명령을 사용하여 Auto Scaling 그룹을 구성할 경우, Auto Scaling 그룹에 지정된 배치 그룹을 사용하기 위해 무시됩니다.
- EBS 최적 인스턴스: Amazon EBS I/O에 대한 추가, 전용 용량을 제공합니다. 모든 인스턴스 유형이 이 기능을 지원하는 것은 아니며, 추가 요금이 적용됩니다.
- 테넌시: 공유 하드웨어(공유), 전용 하드웨어(전용), 호스트 리소스 그룹, 전용 호스트(전용 호스트)에서 인스턴스를 실행하도록 선택할 수 있습니다. 추가 요금이 적용될 수 있습니다.
- 테넌시 호스트 리소스 그룹: 전용 호스트에서 사용할 BYOL AMI의 호스트 리소스 그룹을 지정합니다. 자세한 내용은 AWS License Manager 사용자 안내서의 [호스트 리소스 그룹](#)을 참조하십시오. 참고: 이

기능을 사용하기 전에 계정에 전용 호스트가 이미 할당되어 있지 않아도 됩니다. 인스턴스가 전용 호스트에서 자동으로 시작됩니다.

- RAM 디스크 ID: AMI와 연결된 RAM 디스크의 ID입니다. 반가상화(PV) AMI에서만 유효합니다.
- 커널 ID: AMI와 연결된 커널 ID입니다. 반가상화(PV) AMI에서만 유효합니다.
- License configurations(라이선스 구성): 사용할 라이선스 구성을 지정할 수 있습니다.
- 사용자 데이터: 시작 과정에서 인스턴스를 구성하거나 구성 스크립트를 실행할 때 사용할 사용자 데이터를 지정할 수 있습니다.

10. 시작 템플릿 생성을 선택합니다.

현재 새 EC2 콘솔을 사용하지 않는 경우 이전 EC2 콘솔을 사용하여 시작 템플릿을 생성할 수 있습니다.

Auto Scaling 그룹의 새로운 시작 템플릿을 생성하려면(이전 EC2 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 시작 템플릿을 선택합니다.
3. 새 템플릿 생성을 선택합니다. 시작 템플릿의 이름을 입력하고 초기 버전에 대한 설명을 제공하십시오.
4. 다른 템플릿을 기반으로 새 템플릿을 생성하기로 선택한 경우:
 - a. 소스 템플릿에서 기존 시작 템플릿을 선택합니다.
 - b. 소스 템플릿 버전에서 새로운 시작 템플릿의 토대가 되는 시작 템플릿 버전을 선택합니다.
5. 시작 템플릿 내용에 다음 정보를 입력합니다.
 - a. AMI ID: 인스턴스를 시작할 Amazon 머신 이미지(AMI)의 ID를 선택합니다. AMI 검색 대화 상자를 사용하여 사용 가능한 모든 AMI를 검색할 수 있습니다. 빠른 시작 목록에서 자주 사용하는 AMI 중 하나를 선택합니다. 필요한 AMI가 없는 경우에는 AWS Marketplace 또는 커뮤니티 AMI 목록을 선택하여 **적합한 AMI를 선택합니다**.
 - b. 인스턴스 유형: **인스턴스 유형**을 선택합니다. 인스턴스 유형이 지정한 AMI와 호환되는지 확인합니다.
 - c. 키 페어 이름: 인스턴스와 연결할 때 사용할 **키 페어**를 지정합니다.
 - d. 네트워크 유형: 인스턴스를 VPC로 인스턴스를 시작할지 또는 EC2-Classic으로 시작할지를 지정할 수 있습니다(해당되는 경우). 그러나 Amazon EC2 Auto Scaling의 경우 Auto Scaling 그룹 설정이 우선하기 때문에 시작 템플릿의 네트워크 유형과 가용 영역 설정이 무시됩니다.
 - e. 보안 그룹: 하나 이상의 VPC 또는 EC2-Classic **보안 그룹**을 선택하거나, 하나 이상의 VPC 보안 그룹을 네트워크 인터페이스의 일부로 구성하도록 비워둡니다. 두 곳 모두에서 보안 그룹을 지정할 수 없습니다. EC2-Classic을 사용하는 경우 EC2-Classic용으로 특별히 생성된 보안 그룹을 사용해야 합니다.
6. 네트워크 인터페이스에서 기본 네트워크 인터페이스(eth0)를 지정합니다. 네트워크 인터페이스는 하나만 지정할 수 있습니다.
7. 스토리지(볼륨)에서 블록 디바이스 매핑을 사용하여 인스턴스에 대한 추가 스토리지 볼륨을 지정합니다.
8. 인스턴스 태그에서 키와 값의 조합을 제공하여 태그를 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.
10. 시작 템플릿 생성을 선택합니다.

기존 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 및 설명을 입력합니다. 필요에 따라 다른 시작 파라미터를 조정하고 시작 템플릿 생성을 선택합니다.

명령줄을 사용하여 시작 템플릿을 생성하려면

다음 명령 중 하나를 사용할 수 있습니다:

- [create-launch-template](#) (AWS CLI)
- [New-EC2LaunchTemplate](#) (Windows PowerShell용 AWS 도구)

다음과 같이 [create-launch-template](#) 명령을 사용하여 시작 템플릿을 생성합니다. Auto Scaling 그룹이 인스턴스를 시작하는 VPC의 보안 그룹에 해당하는 Groups에 값을 지정합니다. Auto Scaling 그룹의 속성으로 VPC 및 서브넷을 지정합니다.

```
aws ec2 create-launch-template --launch-template-name my-template-for-auto-scaling --  
version-description version1 \  
--launch-template-data '{"NetworkInterfaces":  
[{"DeviceIndex":0,"AssociatePublicIpAddress":true,"Groups":  
["sg-7c227019"],"DeleteOnTermination":true},"ImageId":"ami-01e24be29428c15b2","InstanceType":"t2.micro",  
[{"ResourceType":"instance","Tags":[{"Key":"purpose","Value":"webserver"}]}]}'
```

다음 [describe-launch-templates](#) 명령을 사용하여 시작 템플릿 *my-template-for-auto-scaling*을 설명합니다.

```
aws ec2 describe-launch-templates --launch-template-names my-template-for-auto-scaling
```

다음 [describe-launch-template-versions](#) 명령을 사용하여 지정된 *my-template-for-auto-scaling* 시작 템플릿 버전을 설명합니다.

```
aws ec2 describe-launch-template-versions --launch-template-id lt-068f72b72934aff71
```

다음은 응답의 예입니다.

```
{  
  "LaunchTemplateVersions": [  
    {  
      "VersionDescription": "version1",  
      "LaunchTemplateId": "lt-068f72b72934aff71",  
      "LaunchTemplateName": "my-template-for-auto-scaling",  
      "VersionNumber": 1,  
      "CreatedBy": "arn:aws:iam::123456789012:user/Bob",  
      "LaunchTemplateData": {  
        "TagSpecifications": [  
          {  
            "ResourceType": "instance",  
            "Tags": [  
              {  
                "Key": "purpose",  
                "Value": "webserver"  
              }  
            ]  
          }  
        ],  
        "ImageId": "ami-01e24be29428c15b2",  
        "InstanceType": "t2.micro",  
      }  
    }  
  ]  
}
```



```
    "NetworkInterfaces": [
      {
        "DeviceIndex": 0,
        "DeleteOnTermination": true,
        "Groups": [
          "sg-7c227019"
        ],
        "AssociatePublicIpAddress": true
      }
    ],
    "DefaultVersion": true,
    "CreateTime": "2019-02-28T19:52:27.000Z"
  }
}
```

시작 구성을 시작 템플릿에 복사

다음 절차를 이용하여, 기존 시작 구성에서 옵션을 복사하여 새 시작 템플릿을 만드십시오.

기존 시작 구성에서 시작 템플릿을 생성하여 Auto Scaling 그룹이 시작 템플릿을 사용하도록 쉽게 업데이트 할 수 있습니다. 시작 구성과 마찬가지로 시작 템플릿에는 인스턴스 시작에 필요한 전체 또는 일부 파라미터가 포함될 수 있습니다. 시작 템플릿을 사용하면 새 인스턴스를 더 빠르고 쉽게 시작하기 위해 여러 가지 버전의 템플릿을 만들 수도 있습니다.

시작 구성에서 시작 템플릿을 만들려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Launch Configurations]를 선택합니다.
3. 복사할 시작 구성을 선택하고 Copy to launch template(시작 템플릿으로 복사)을 선택합니다. 이렇게 하면 새 시작 템플릿을 사용자가 선택한 시작 구성과 동일한 이름과 옵션으로 설정합니다.
4. New launch template name(새 시작 템플릿 이름)의 경우, 시작 구성의 이름(기본값)을 사용하거나 새 이름을 입력할 수 있습니다. 시작 템플릿 이름은 고유한 이름이어야 합니다.
5. (선택 사항) 새 시작 템플릿을 사용하여 Auto Scaling 그룹을 만들려면 Create an Auto Scaling group using the new template(새 템플릿을 사용하여 Auto Scaling 그룹 만들기)을 선택합니다. 자세한 내용은 [시작 템플릿을 사용하여 Auto Scaling 그룹 생성하기 \(p. 54\)](#) 단원을 참조하십시오.
6. [Submit]을 선택합니다.

시작 템플릿을 생성하고 나면 필요에 따라 사용자가 생성한 시작 템플릿으로 기존 Auto Scaling 그룹을 업데이트할 수 있습니다. 자세한 내용은 [시작 구성을 시작 템플릿으로 교체 \(p. 32\)](#) 단원을 참조하십시오.

시작 구성을 시작 템플릿으로 교체

기존의 시작 구성이 있는 Auto Scaling 그룹을 편집할 경우, 시작 구성을 시작 템플릿으로 교체하는 옵션이 있습니다. 이 옵션을 사용하면 현재 사용하고 있는 Auto Scaling 그룹에 시작 템플릿을 사용할 수 있습니다. 이렇게 함으로써 시작 템플릿의 버전 관리 및 기타 기능을 활용할 수 있습니다.

한 Auto Scaling 그룹의 시작 구성을 교체하면 새 인스턴스는 새로운 시작 템플릿을 사용하여 시작되지만 기존 인스턴스는 영향을 받지 않습니다. 이 경우 Auto Scaling 그룹의 기존 인스턴스를 종료하여 시작 템플릿을 사용하는 새 인스턴스를 강제로 시작할 수 있습니다. 또는 자동 조정을 허용하여 [종료 정책 \(p. 126\)](#)에 따라 이전 인스턴스를 새 인스턴스로 점진적으로 대체할 수 있습니다. AWS CloudFormation을 통해 몇 번의 클릭만으로 업데이트의 배포를 자동화할 수도 있습니다.

사전 요구 사항

Auto Scaling 그룹에서 시작 구성을 교체하려면 먼저 시작 템플릿을 만들어야 합니다. 시작 템플릿을 만드는 가장 쉬운 방법은 시작 구성에서 시작 템플릿을 복사하는 것입니다. 자세한 내용은 [시작 구성을 시작 템플릿에 복사 \(p. 32\)](#) 단원을 참조하십시오.

시작 구성을 시작 템플릿으로 바꾸면 `ec2:RunInstances` 권한이 확인됩니다. 시작 템플릿을 사용하려고 하는데 충분한 권한이 없는 경우 시작 템플릿을 사용할 권한이 없다는 오류가 발생합니다.

Auto Scaling 그룹의 시작 구성을 교체하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
3. Auto Scaling 그룹을 선택하고 Details, Edit(세부 사항, 편집)를 선택합니다.
4. Launch Instances Using(인스턴스 시작 시 사용할 대상)에서, 시작 템플릿 옵션을 선택합니다.
5. 시작 템플릿에서 사용자의 시작 템플릿을 선택합니다.
6. 시작 템플릿 버전에서 필요한 시작 템플릿 버전을 선택합니다. 한 시작 템플릿을 여러 버전으로 만든 다음에는 Auto Scaling 그룹이 확장 시 시작 템플릿의 기본 버전을 사용할지 최신 버전을 사용할지를 선택합니다.
7. 완료되면 [Save]를 선택합니다.

명령줄을 사용하여 시작 구성을 대체하려면

다음 명령 중 하나를 사용할 수 있습니다:

- `update-auto-scaling-group`(AWS CLI)
- `Update-ASAutoScalingGroup`(Windows PowerShell용 AWS 도구)

시작 구성

시작 구성은 Auto Scaling 그룹에서 EC2 인스턴스를 시작하는 데 사용하는 인스턴스 구성 템플릿입니다. 시작 구성을 만들 때 인스턴스에 대한 정보를 지정합니다. Amazon 머신 이미지(AMI)의 ID, 인스턴스 유형, 키 페어, 하나 이상의 보안 그룹, 블록 디바이스 매핑 등을 포함시키십시오. 이전에 EC2 인스턴스를 시작한 적이 있었다면 바로 이러한 정보를 지정하여 인스턴스를 시작했을 것입니다.

여러 개의 Auto Scaling 그룹을 가진 시작 구성을 지정할 수 있습니다. 하지만 Auto Scaling 그룹에 대해서는 한 번에 한 개의 시작 구성만 지정할 수 있으며 시작 구성을 한 번 생성한 후에는 수정할 수 없습니다. Auto Scaling 그룹의 시작 구성을 변경하려면 시작 구성을 생성한 후 이 구성으로 Auto Scaling 그룹을 업데이트해야 합니다.

Auto Scaling 그룹을 생성할 때마다 시작 구성, 시작 템플릿, 또는 EC2 인스턴스를 지정해야 한다는 점을 명심하십시오. EC2 인스턴스를 사용하여 Auto Scaling 그룹을 생성하면 Amazon EC2 Auto Scaling에서 자동으로 사용자를 위해 시작 구성을 생성하고 이 구성을 Auto Scaling 그룹에 연결합니다. 자세한 내용은 [EC2 인스턴스를 사용하여 Auto Scaling 그룹 만들기 \(p. 59\)](#) 단원을 참조하십시오. 또는 시작 템플릿을 사용하는 경우 시작 구성 또는 EC2 인스턴스 대신에 시작 템플릿을 지정할 수 있습니다. 자세한 내용은 [시작 템플릿 \(p. 26\)](#) 단원을 참조하십시오.

목차

- [시작 구성 만들기 \(p. 34\)](#)
- [EC2 인스턴스를 사용하여 시작 구성 만들기 \(p. 35\)](#)
- [Auto Scaling 그룹 시작 구성 변경 \(p. 39\)](#)
- [VPC의 Auto Scaling 인스턴스 시작 \(p. 40\)](#)

시작 구성 만들기

시작 구성을 만들 때 시작할 EC2 인스턴스에 대한 정보를 지정해야 합니다. Amazon 머신 이미지(AMI), 인스턴스 유형, 키 페어, 보안 그룹 및 블록 디바이스 매핑을 포함시키십시오. 또는 실행 중인 EC2 인스턴스의 속성을 사용하여 시작 구성을 만들 수 있습니다. 자세한 내용은 [EC2 인스턴스를 사용하여 시작 구성 만들기 \(p. 35\)](#) 단원을 참조하십시오.

Note

Auto Scaling 그룹을 생성할 때 시작 템플릿, 시작 구성 또는 EC2 인스턴스를 지정할 수 있습니다. Amazon EC2의 최신 기능을 사용하려면 시작 템플릿을 사용하는 것이 좋습니다. 자세한 내용은 [시작 템플릿 \(p. 26\)](#) 단원을 참조하십시오.

시작 구성을 만들었으면 이제 Auto Scaling 그룹을 만들 수 있습니다. 자세한 내용은 [시작 구성을 사용하여 Auto Scaling 그룹 생성하기 \(p. 57\)](#) 단원을 참조하십시오.

Auto Scaling 그룹은 한 번에 한 개의 시작 구성과 연결되며, 시작 구성을 생성한 후에는 수정할 수 없습니다. 따라서 기존 Auto Scaling 그룹의 시작 구성을 변경하려면 새 시작 구성으로 그룹을 업데이트해야 합니다. 자세한 내용은 [Auto Scaling 그룹 시작 구성 변경 \(p. 39\)](#) 단원을 참조하십시오.

콘솔을 사용하여 시작 구성을 만들려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에는 현재 리전이 표시됩니다. 요구에 맞는 Auto Scaling 그룹의 리전을 선택합니다.
3. 탐색 창의 Auto Scaling 아래에서 시작 구성을 선택합니다.

4. 다음 페이지에서 시작 구성 생성을 선택합니다.
5. [Choose AMI] 페이지에서 AMI를 선택합니다.
6. [Choose an Instance Type] 페이지에서 인스턴스의 하드웨어 구성을 선택합니다. [Next: Configure details]를 선택합니다.

Note

T2 인스턴스는 VPC의 서브넷에서 시작되어야 합니다. t2.micro 인스턴스를 선택했으나 VPC가 없는 경우 하나가 자동으로 생성됩니다. 이 VPC에는 해당 리전의 가용 구역별 퍼블릭 서브넷이 포함됩니다.

7. [Configure Details] 페이지에서 다음을 수행합니다.
 - a. [Name]에 시작 구성의 이름을 입력합니다.
 - b. (선택사항) 구매 옵션의 경우 스팟 인스턴스를 요청하여 인스턴스 시간당 지불할 최고 가격을 지정할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹에서 스팟 인스턴스 시작 \(p. 73\)](#) 단원을 참조하십시오.
 - c. (선택 사항) [IAM role]에서 인스턴스와 연결할 역할을 선택합니다. 자세한 내용은 [Amazon EC2 인스턴스에서 실행되는 애플리케이션의 IAM 역할 \(p. 192\)](#) 단원을 참조하십시오.
 - d. (선택 사항) 기본적으로 Auto Scaling 인스턴스에 대해 기본 모니터링 기능이 활성화됩니다. Auto Scaling 인스턴스에 대해 세부 모니터링을 활성화하려면 CloudWatch 세부 모니터링 활성화를 선택합니다.
 - e. [Advanced Details]의 [IP Address Type]에서 옵션을 선택합니다. VPC의 인스턴스에 연결하려면 퍼블릭 IP 주소를 할당하는 옵션을 선택해야 합니다. 인스턴스에 연결하려는데 기본 VPC가 있는지 확실치 않은 경우, [Assign a public IP address to every instance]를 선택합니다.
 - f. [Skip to review]를 선택합니다.
8. [Review] 페이지에서 [Edit security groups]를 선택합니다. 지침을 따라 기존 보안 그룹을 선택한 다음 [Review]를 선택합니다.
9. [Review] 페이지에서 [Create launch configuration]을 선택합니다.
10. [Select an existing key pair or create a new key pair]에서 나열된 옵션 중 하나를 선택합니다. 승인 확인란을 선택한 다음 [Create launch configuration]을 선택합니다.

Warning

인스턴스에 연결해야 하는 경우 키 페어 없이 계속을 선택하지 마십시오.

명령줄을 사용하여 시작 구성을 만들려면

다음 명령 중 하나를 사용할 수 있습니다:

- [create-launch-configuration](#) (AWS CLI)
- [New-ASLaunchConfiguration](#) (Windows PowerShell용 AWS 도구)

EC2 인스턴스를 사용하여 시작 구성 만들기

Amazon EC2 Auto Scaling은 실행 중인 EC2 인스턴스의 속성을 사용하여 시작 구성을 만들 수 있는 옵션을 제공합니다.

Tip

[EC2 인스턴스에서 직접 Auto Scaling 그룹을 만들 수 있습니다 \(p. 59\)](#). 이 기능을 사용해도 Amazon EC2 Auto Scaling이 시작 구성을 자동으로 만듭니다.

지정된 인스턴스에 현재 시작 구성에서 지원되지 않는 속성이 있는 경우, Auto Scaling 그룹에서 시작하는 인스턴스는 기존 EC2 인스턴스와 동일하지 않을 수도 있습니다.

처음부터 시작 구성을 새로 만드는 것과 기존 EC2 인스턴스에서 시작 구성을 만드는 것 사이에는 차이점이 있습니다. 처음부터 시작 구성을 새로 만드는 경우 이미지 ID, 인스턴스 유형, 선택적 리소스(스토리지 디바이스 등) 및 옵션 설정(모니터링 등)을 지정합니다. 실행 중인 인스턴스에서 시작 구성을 생성할 경우, Amazon EC2 Auto Scaling은 지정된 인스턴스에서 시작 구성에 대한 속성을 가져옵니다. 또한 속성은 인스턴스가 시작된 AMI에 대한 블록 디바이스 매핑에서도 파생되며, 실행 후 추가된 추가 블록 디바이스는 무시합니다.

실행 중인 인스턴스를 사용하여 시작 구성을 만들 때 해당 속성을 동일한 요청의 일부로 지정하여 AMI, 블록 디바이스, 키 페어, 인스턴스 프로파일, 인스턴스 유형, 커널, 모니터링, 배치 테넌시, ramdisk, 보안 그룹, 스팟 가격, 사용자 데이터, 해당 인스턴스에 연결된 퍼블릭 IP 주소가 있는지, 해당 인스턴스가 EBS에 최적화되어 있는지 등의 속성을 재정의할 수 있습니다.

다음 예를 통해 EC2 인스턴스에서 시작 구성을 만드는 방법을 확인합니다.

예제

- [EC2 인스턴스를 사용하여 시작 구성 만들기 \(p. 36\)](#)
- [인스턴스에서 시작 구성을 만들고 블록 디바이스 재정의\(AWS CLI\) \(p. 37\)](#)
- [시작 구성 만들기 및 인스턴스 유형 재정의\(AWS CLI\) \(p. 38\)](#)

EC2 인스턴스를 사용하여 시작 구성 만들기

기존 Amazon EC2 인스턴스의 속성을 사용하여 시작 구성을 만들기 위해 해당 인스턴스의 ID를 지정합니다.

Important

지정된 인스턴스를 시작할 때 사용될 AMI가 늘 존재해야 합니다.

EC2 인스턴스에서 시작 구성 만들기(콘솔)

콘솔을 사용하여 실행 중인 EC2 인스턴스에서 시작 구성과 Auto Scaling 그룹을 만들 수 있으며 새 Auto Scaling 그룹에 해당 인스턴스를 추가할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹에 EC2 인스턴스 연결 \(p. 92\)](#) 단원을 참조하십시오.

EC2 인스턴스에서 시작 구성 만들기(AWS CLI)

다음 `create-launch-configuration` 명령을 사용하여 인스턴스에서 인스턴스와 동일한 속성으로 시작 구성을 만듭니다. 시작 후 추가된 블록 디바이스는 무시됩니다.

```
aws autoscaling create-launch-configuration --launch-configuration-name my-lc-from-instance
--instance-id i-a8e09d9c
```

다음 `describe-launch-configurations` 명령을 사용하면 시작 구성을 설명하고 관련 속성이 해당 인스턴스의 속성과 일치하는지 확인할 수 있습니다.

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-lc-from-instance
```

다음은 응답의 예입니다.

```
{
  "LaunchConfigurations": [
    {
      "UserData": null,
      "EbsOptimized": false,
      "LaunchConfigurationARN": "arn",
      "InstanceMonitoring": {
```

```
        "Enabled": false
      },
      "ImageId": "ami-05355a6c",
      "CreatedTime": "2014-12-29T16:14:50.382Z",
      "BlockDeviceMappings": [],
      "KeyName": "my-key-pair",
      "SecurityGroups": [
        "sg-8422d1eb"
      ],
      "LaunchConfigurationName": "my-lc-from-instance",
      "KernelId": "null",
      "RamdiskId": null,
      "InstanceType": "t1.micro",
      "AssociatePublicIpAddress": true
    }
  ]
}
```

인스턴스에서 시작 구성을 만들고 블록 디바이스 재정의(AWS CLI)

기본적으로 Amazon EC2 Auto Scaling은 시작 구성을 만들기 위해 사용자가 지정한 EC2 인스턴스의 속성을 사용합니다. 하지만 블록 디바이스는 인스턴스가 아니라 인스턴스를 시작하는 데 사용된 AMI에서 파생됩니다. 시작 구성에 블록 디바이스를 추가하려면 시작 구성에서 블록 디바이스 매핑을 재정의해야 합니다.

Important

지정된 인스턴스를 시작할 때 사용될 AMI가 늘 존재해야 합니다.

인스턴스에서 시작 구성을 만들고 블록 디바이스 재정의

다음 `create-launch-configuration` 명령을 사용하면 EC2 인스턴스를 사용하되 사용자 지정 블록 디바이스 매핑으로 시작 구성을 만들 수 있습니다.

```
aws autoscaling create-launch-configuration --launch-configuration-name my-lc-from-  
instance-bdm --instance-id i-a8e09d9c \\  
--block-device-mappings "[{\"DeviceName\":\"/dev/sda1\", \"Ebs\":{\"SnapshotId\":  
\"snap-3decf207\"}}, {\"DeviceName\":\"/dev/sdf\", \"Ebs\":{\"SnapshotId\": \"snap-  
eed6ac86\"}}]"
```

다음 `describe-launch-configurations` 명령을 사용하면 시작 구성을 설명하고 해당 구성에 사용자 지정 블록 디바이스 매핑이 사용되었는지를 확인할 수 있습니다.

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-lc-from-  
instance-bdm
```

다음 예제 응답은 시작 구성에 대한 내용입니다.

```
{
  "LaunchConfigurations": [
    {
      "UserData": null,
      "EbsOptimized": false,
      "LaunchConfigurationARN": "arn",
      "InstanceMonitoring": {
        "Enabled": false
      },
      "ImageId": "ami-c49c0dac",
      "CreatedTime": "2015-01-07T14:51:26.065Z",
```

```
    "BlockDeviceMappings": [
      {
        "DeviceName": "/dev/sda1",
        "Ebs": {
          "SnapshotId": "snap-3decf207"
        }
      },
      {
        "DeviceName": "/dev/sdf",
        "Ebs": {
          "SnapshotId": "snap-eed6ac86"
        }
      }
    ],
    "KeyName": "my-key-pair",
    "SecurityGroups": [
      "sg-8637d3e3"
    ],
    "LaunchConfigurationName": "my-lc-from-instance-bdm",
    "KernelId": null,
    "RamdiskId": null,
    "InstanceType": "t1.micro",
    "AssociatePublicIpAddress": true
  }
]
```

시작 구성 만들기 및 인스턴스 유형 재정의(AWS CLI)

기본적으로 Amazon EC2 Auto Scaling은 시작 구성을 만들기 위해 사용자가 지정한 EC2 인스턴스의 속성을 사용합니다. 요구 사항에 따라 해당 인스턴스의 속성을 재정의하여 사용자가 요구하는 값을 사용하고자 할 수 있습니다. 예를 들면 사용자는 인스턴스 유형을 재정의할 수 있습니다.

Important

지정된 인스턴스를 시작할 때 사용될 AMI가 늘 존재해야 합니다.

시작 구성 만들기 및 인스턴스 유형 재정의

다음 `create-launch-configuration` 명령을 사용하면 EC2 인스턴스를 사용하되 해당 인스턴스(예: t2.micro)와 다른 인스턴스 유형(예: t2.medium)으로 시작 구성을 만들 수 있습니다.

```
aws autoscaling create-launch-configuration --launch-configuration-name my-lc-from-  
instance-changetype \  
--instance-id i-a8e09d9c --instance-type t2.medium
```

다음 `describe-launch-configurations` 명령을 사용하면 시작 구성을 설명하고 관련 속성이 해당 인스턴스 유형이 재정의되었는지 확인할 수 있습니다.

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-lc-from-  
instance-changetype
```

다음 예제 응답은 시작 구성에 대한 내용입니다.

```
{
  "LaunchConfigurations": [
    {
      "UserData": null,
      "EbsOptimized": false,
      "LaunchConfigurationARN": "arn",

```

```
    "InstanceMonitoring": {  
      "Enabled": false  
    },  
    "ImageId": "ami-05355a6c",  
    "CreatedTime": "2014-12-29T16:14:50.382Z",  
    "BlockDeviceMappings": [],  
    "KeyName": "my-key-pair",  
    "SecurityGroups": [  
      "sg-8422d1eb"  
    ],  
    "LaunchConfigurationName": "my-lc-from-instance-changetype",  
    "KernelId": "null",  
    "RamdiskId": null,  
    "InstanceType": "t2.medium",  
    "AssociatePublicIpAddress": true  
  }  
]  
}
```

Auto Scaling 그룹 시작 구성 변경

Auto Scaling 그룹은 한 번에 한 개의 시작 구성과 연결되며, 시작 구성을 생성한 후에는 수정할 수 없습니다. 기존 Auto Scaling 그룹의 시작 구성을 변경하기 위해 기존 시작 구성을 새 시작 구성의 기초로 사용할 수 있습니다. 그런 다음 새로운 시작 구성을 사용하도록 Auto Scaling을 업데이트합니다.

한 Auto Scaling 그룹의 시작 구성을 변경하면 새 인스턴스는 새로운 구성 옵션을 사용하여 시작되지만 기존 인스턴스는 영향을 받지 않습니다. 이 경우 Auto Scaling 그룹의 기존 인스턴스를 종료하여 새 구성을 사용하는 새 인스턴스를 강제로 시작할 수 있습니다. 또는 자동 조정을 허용하여 [종료 정책 \(p. 126\)](#)에 따라 이전 인스턴스를 새 인스턴스로 점진적으로 대체할 수 있습니다. AWS CloudFormation을 통해 몇 번의 클릭만으로 업데이트된 시작 구성의 배포를 자동화할 수도 있습니다.

Auto Scaling 그룹에 대한 시작 구성을 변경하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Launch Configurations]를 선택합니다.
3. 시작 구성을 선택한 다음 [Actions], [Copy launch configuration]을 선택합니다. 이는 새 시작 구성을 원래 시작 구성과 동일한 옵션으로 설정하지만 이름에는 "Copy"를 추가합니다.
4. [Copy Launch Configuration] 페이지에서 필요에 따라 구성 옵션을 편집하고 [Create launch configuration]을 선택합니다.
5. 확인 페이지에서 Auto Scaling 그룹 보기를 선택합니다.
6. Auto Scaling 그룹을 선택한 다음 세부 정보, 편집을 선택합니다.
7. [Launch Configuration]에서 새 시작 구성을 선택한 다음 [Save]를 선택합니다.

Auto Scaling 그룹에 대한 시작 구성을 변경하려면(AWS CLI)

1. `describe-launch-configurations` 명령을 사용하여 현재 시작 구성을 지정합니다.
2. `create-launch-configuration` 명령을 사용하여 시작 구성을 새로 만듭니다.
3. `update-auto-scaling-group` 명령과 `--launch-configuration-names` 파라미터를 사용하여 Auto Scaling 그룹에 대한 시작 구성을 업데이트합니다.

Auto Scaling 그룹에 대한 시작 구성을 변경하려면(Windows PowerShell용 도구)

1. `Get-ASLaunchConfiguration` 명령을 사용하여 현재 시작 구성을 지정합니다.
2. `New-ASLaunchConfiguration` 명령을 사용하여 시작 구성을 새로 만듭니다.

3. [Update-ASAutoScalingGroup](#) 명령과 `-LaunchConfigurationName` 파라미터를 사용하여 Auto Scaling 그룹에 대한 시작 구성을 업데이트합니다.

VPC의 Auto Scaling 인스턴스 시작

Amazon Virtual Private Cloud(Amazon VPC)을 사용하면 AWS 클라우드의 격리된 사설 영역에서 가상 네트워크 환경을 재구성할 수 있습니다. 이제 가상 네트워크 환경을 완벽하게 제어할 수 있게 됩니다. 자세한 내용은 [Amazon VPC 사용 설명서](#) 섹션을 참조하십시오.

Virtual Private Cloud(VPC)로 Auto Scaling 그룹과 같은 AWS 리소스를 시작할 수 있습니다. VPC에서 Auto Scaling 그룹은 사실상 Amazon EC2에 대한 방식과 마찬가지로 작동하고 같은 기능 집합을 지원합니다.

Amazon VPC의 서브넷은 해당 VPC의 IP 주소 범위 중 한 부분에 따라 정의되는 가용 구역 이내의 하위분류 단위입니다. 서브넷을 사용하면 보안 및 운영상의 필요에 따라 인스턴스를 그룹화할 수 있습니다. 서브넷은 서브넷이 만들어진 원래 가용 영역 내에서만 존재합니다. 서브넷 안에서 Auto Scaling 인스턴스를 시작합니다.

인터넷과 서브넷의 인스턴스 사이에 통신할 수 있도록 설정하려면 인터넷 게이트웨이를 만들어 VPC에 연결해야 합니다. 인터넷 게이트웨이를 사용하면 서브넷 안의 리소스가 Amazon EC2 네트워크 엣지를 통해 인터넷에 연결되도록 할 수 있습니다. 서브넷 트래픽이 인터넷 게이트웨이로 라우팅되는 경우 해당 서브넷을 퍼블릭 서브넷이라고 합니다. 서브넷 트래픽이 인터넷 게이트웨이로 라우팅되지 않는 경우 해당 서브넷을 프라이빗 서브넷이라고 합니다. 반드시 인터넷에 연결되어야 하는 리소스의 퍼블릭 서브넷과 인터넷에 연결되지 않아도 되는 리소스의 프라이빗 서브넷을 사용합니다.

먼저 VPC 환경을 만들어야 새 VPC의 Auto Scaling 인스턴스를 시작할 수 있습니다. VPC와 서브넷을 만든 후 해당 서브넷 안에서 Auto Scaling 인스턴스를 시작합니다. 퍼블릭 서브넷이 하나인 VPC를 만드는 가장 쉬운 방법은 VPC 마법사를 사용하는 것입니다. 자세한 내용은 [Amazon VPC 시작 안내서](#)를 참조하십시오.

목차

- [기본 VPC \(p. 40\)](#)
- [VPC의 IP 어드레싱 \(p. 40\)](#)
- [인스턴스 배치 테넌시 \(p. 41\)](#)
- [VPC에 대한 EC2-Classic 인스턴스 링크 설정 \(p. 42\)](#)

기본 VPC

2013년 12월 4일 이후에 AWS 계정을 생성하였거나 새 리전에서 Auto Scaling 그룹을 만드는 경우, 사용자를 위한 기본 VPC가 생성됩니다. 기본 VPC는 가용 영역별로 기본 서브넷과 함께 제공됩니다. 기본 VPC가 있는 경우 기본적으로 기본 VPC에서 Auto Scaling 그룹이 만들어집니다.

기본 VPC에 대한 자세한 내용을 알아보고 계정이 기본 VPC와 함께 제공되는지 확인하려면 Amazon VPC 개발자 안내서의 [기본 VPC 및 서브넷](#)을 참조하십시오.

VPC의 IP 어드레싱

VPC의 Auto Scaling 인스턴스를 시작할 때 해당 인스턴스에는 서브넷의 주소 범위에 속하는 프라이빗 IP 주소가 자동으로 할당됩니다. 이를 통해 인스턴스가 VPC의 다른 인스턴스와 통신할 수 있게 됩니다.

인스턴스에 퍼블릭 IP 주소를 할당하도록 시작 구성을 설정할 수 있습니다. 인스턴스에 퍼블릭 IP 주소를 할당하면 AWS에서 인터넷이나 기타 서비스와 통신할 수 있게 됩니다.

인스턴스에 퍼블릭 IP 주소를 사용하고 IPv6 주소를 자동으로 할당하도록 구성된 서브넷으로 인스턴스를 시작하면 IPv4 및 IPv6 주소를 모두 수신합니다. 그렇지 않으면 IPv4 주소만 수신합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [IPv6 주소](#)를 참조하십시오.

인스턴스 배치 테넌시

테넌시는 EC2 인스턴스가 물리적 하드웨어에 분산되는 방식을 정의하고 요금에 영향을 줍니다. 다음 세 가지 테넌시 옵션을 사용할 수 있습니다.

- 공유(default) – 여러 AWS 계정이 동일한 물리적 하드웨어를 공유할 수 있습니다.
- 전용 인스턴스(dedicated) – 인스턴스가 단일 테넌트 하드웨어에서 실행됩니다.
- 전용 호스트(host) – 인스턴스가 구성을 제어할 수 있는 격리된 서버인 전용 호스트에서 실행됩니다.

Important

시작 구성 또는 시작 템플릿을 사용하여 EC2 인스턴스에 대한 테넌시를 구성할 수 있습니다. 그러나 host 테넌시 값을 시작 구성과 함께 사용할 수 없습니다. default 또는 dedicated 테넌시 값만 사용합니다. host의 테넌시 값을 사용하려면 시작 템플릿을 사용해야 합니다. 자세한 내용은 [Auto Scaling 그룹을 위한 시작 템플릿 만들기 \(p. 26\)](#) 단원을 참조하십시오.

전용 인스턴스는 호스트 하드웨어 수준에서 전용이 아닌 인스턴스와 다른 AWS 계정에 속하는 인스턴스로부터 물리적으로 격리됩니다. VPC를 만들 때 기본적으로 해당 테넌시 속성이 default로 설정됩니다. 해당 VPC에서 인스턴스가 단일 테넌시 인스턴스로 실행될 수 있도록 dedicated의 테넌시 값으로 인스턴스를 시작할 수 있습니다. 그렇지 않으면 기본적으로 공유 테넌시 인스턴스로 실행됩니다. VPC의 테넌시 속성을 dedicated로 설정하는 경우, 해당 VPC에서 시작한 모든 인스턴스는 단일 테넌시 인스턴스로 실행됩니다. 자세한 내용은 [Linux 인스턴스용 Amazon EC2 사용 설명서의 전용 인스턴스](#)를 참조하십시오. 요금 정보는 [Amazon EC2 전용 인스턴스](#) 제품 페이지를 참조하십시오.

시작 구성을 만들 때 인스턴스 배치 테넌시의 기본값은 null이며 해당 인스턴스 테넌시는 VPC의 테넌시 속성에 의해 제어됩니다. 다음 표에는 VPC에서 시작되는 Auto Scaling 인스턴스의 인스턴스 배치 테넌시가 요약되어 있습니다.

시작 구성 테넌시	VPC 테넌시 = default	VPC 테넌시 = dedicated
not specified	공유 테넌시 인스턴스	전용 인스턴스
default	공유 테넌시 인스턴스	전용 인스턴스
dedicated	전용 인스턴스	전용 인스턴스

시작 구성(AWS CLI)을 사용하여 테넌시를 구성하려면

[create-launch-configuration](#) 명령과 `--placement-tenancy` 옵션을 사용하여 시작 구성에 대한 인스턴스 배치 테넌시를 default 또는 dedicated로 지정할 수 있습니다. 예를 들어 시작 구성 테넌시를 dedicated로 설정하려면 다음 명령을 실행합니다.

```
aws autoscaling create-launch-configuration --launch-configuration-name my-launch-config --placement-tenancy dedicated --image-id ...
```

다음 [describe-launch-configurations](#) 명령을 사용하면 시작 구성의 인스턴스 배치 테넌시를 확인할 수 있습니다.

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-launch-config
```

다음은 전용 인스턴스를 만드는 시작 구성에 대한 예제 출력입니다. 단, 사용자가 인스턴스 배치 테넌시를 명시적으로 설정해야만 PlacementTenancy 파라미터가 이 명령에 대한 출력의 일부로 포함됩니다.


```
{
  "LaunchConfigurations": [
    {
      "UserData": null,
      "EbsOptimized": false,
      "PlacementTenancy": "dedicated",
      "LaunchConfigurationARN": "arn",
      "InstanceMonitoring": {
        "Enabled": true
      },
      "ImageId": "ami-b5a7ea85",
      "CreatedTime": "2015-03-08T23:39:49.011Z",
      "BlockDeviceMappings": [],
      "KeyName": null,
      "SecurityGroups": [],
      "LaunchConfigurationName": "my-launch-config",
      "KernelId": null,
      "RamdiskId": null,
      "InstanceType": "m3.medium"
    }
  ]
}
```

VPC에 대한 EC2-Classic 인스턴스 링크 설정

EC2-Classic에서 Auto Scaling 그룹의 인스턴스를 시작하는 경우 ClassicLink를 사용하여 VPC에 해당 인스턴스를 링크할 수 있습니다. ClassicLink를 사용하면 VPC에 대한 한 개 이상의 보안 그룹을 Auto Scaling 그룹의 EC2-Classic 인스턴스와 연결할 수 있습니다. 이렇게 하면 연결된 EC2-Classic 인스턴스와 VPC의 인스턴스 간에 프라이빗 IP 주소를 사용하여 통신할 수 있습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [ClassicLink](#)를 참조하십시오.

Auto Scaling 그룹에서 실행 중인 EC2-Classic 인스턴스가 있는 경우 활성화된 ClassicLink로 VPC에 해당 인스턴스를 링크할 수 있습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [VPC에 인스턴스 링크](#)를 참조하십시오. 또는 시작 시 EC2-Classic 인스턴스를 VPC에 자동으로 연결하는 시작 구성을 사용하도록 Auto Scaling 그룹을 업데이트할 수 있습니다. 그런 다음 실행 중인 인스턴스를 종료하여 Auto Scaling 그룹이 VPC에 연결된 새 인스턴스를 시작할 수 있게 합니다.

VPC에 링크(콘솔)

다음 절차에 따라, EC2-Classic 인스턴스를 지정된 VPC에 링크하는 시작 구성을 만들고 해당 시작 구성을 사용하여 기존 Auto Scaling 그룹을 업데이트합니다.

Auto Scaling의 EC2-Classic 인스턴스를 VPC에 링크하려면

1. VPC에 의해 ClassicLink가 활성화됐는지 확인합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [Viewing Your ClassicLink-Enabled VPCs](#)를 참조하십시오.
2. EC2-Classic 인스턴스를 연결하려는 VPC에 대한 보안 그룹을 만듭니다. 연결된 EC2-Classic 인스턴스와 VPC의 인스턴스 간에 통신을 제어하는 규칙을 추가합니다.
3. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
4. 탐색 창에서 [Launch Configurations]를 선택합니다.
5. [Create launch configuration]을 선택합니다.
6. [Choose AMI] 페이지에서 AMI를 선택합니다.
7. [Choose an Instance Type] 페이지에서 인스턴스 유형을 선택한 다음 [Next: Configure details]를 선택합니다.
8. [Configure details(세부 정보 구성)] 페이지에서 다음을 수행합니다.
 - a. 시작 구성의 이름을 입력합니다.

- b. [Advanced Details(어드밴스 정보)]를 확장하고 필요한 [IP Address Type(IP 주소 형식)]을 선택한 다음 [Link to VPC(VPC에 링크)]를 선택합니다.
 - c. [VPC]에서 ClassicLink가 1단계에 활성화된 VPC를 선택합니다.
 - d. [Security Groups]에서 2단계 보안 그룹을 선택합니다.
 - e. [Skip to review]를 선택합니다.
9. [Review] 페이지에서 필요한 내용을 수정한 다음 [Create launch configuration]을 선택합니다. [Select an existing key pair or create a new key pair]에서 옵션을 선택하고 승인 확인란을 선택(있는 경우)한 다음 [Create launch configuration]을 선택합니다.
10. 메시지가 나타나면 지침에 따라 새 시작 구성을 사용하는 Auto Scaling 그룹을 만듭니다. [Network(네트워크)]에서 Launch into EC2-Classic(EC2-Classic에서 시작)을 선택해야 합니다. 그렇지 않으면 취소를 선택한 다음 다음과 같이 기존 Auto Scaling 그룹에 시작 구성을 추가합니다.
 - a. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
 - b. Auto Scaling 그룹을 선택하고 작업, 편집을 선택합니다.
 - c. [Launch Configuration]에서 새 시작 구성을 선택한 다음 [Save]를 선택합니다.

VPC에 연결(AWS CLI)

다음 절차에 따라, EC2-Classic 인스턴스를 지정된 VPC에 링크하는 시작 구성을 만들고 해당 시작 구성을 사용하여 기존 Auto Scaling 그룹을 업데이트합니다.

Auto Scaling 그룹의 EC2-Classic 인스턴스를 VPC에 링크하려면

1. VPC에 의해 ClassicLink가 활성화됐는지 확인합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [Viewing Your ClassicLink-Enabled VPCs](#)를 참조하십시오.
2. EC2-Classic 인스턴스를 연결하려는 VPC에 대한 보안 그룹을 만듭니다. 연결된 EC2-Classic 인스턴스와 VPC의 인스턴스 간에 통신을 제어하는 규칙을 추가합니다.
3. 다음과 같이 [create-launch-configuration](#) 명령을 사용하는 시작 구성을 생성합니다. vpc_id 값을 1단계에서 ClassicLink를 활성화한 VPC의 ID로 지정하고, group_id를 2단계의 보안 그룹으로 지정합니다.

```
aws autoscaling create-launch-configuration --launch-configuration-name classiclink-config \  
  --image-id ami_id --instance-type instance_type \  
  --classic-link-vpc-id vpc_id --classic-link-vpc-security-groups group_id
```

4. 이전 단계에서 만든 시작 구성으로 기존 Auto Scaling 그룹(예: my-asg 등)을 업데이트합니다. 이 Auto Scaling에서 시작하는 모든 새 EC2-Classic 인스턴스는 링크된 EC2-Classic 인스턴스입니다. 다음과 같이 명령을 실행합니다.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg \  
  --launch-configuration-name classiclink-config
```

또는 [create-auto-scaling-group](#)을 사용하여, 새로 생성하는 Auto Scaling 그룹에 이 시작 구성을 사용할 수 있습니다.

Auto Scaling 그룹

Auto Scaling 그룹에는 자동 조정 및 관리 목적의 논리적 그룹으로 취급되는 Amazon EC2 인스턴스 모음이 포함되어 있습니다. Auto Scaling 그룹을 통해 상태 검사 교체 및 조정 정책과 같은 Amazon EC2 Auto Scaling 기능도 사용할 수 있습니다. Auto Scaling 그룹 및 자동 조정에서 인스턴스 수를 유지하는 이 두 가지가 Amazon EC2 Auto Scaling 서비스의 핵심 기능입니다.

Auto Scaling 그룹의 크기는 사용자가 원하는 용량으로 설정한 인스턴스 수에 따라 달라집니다. 수동으로 또는 자동 조정을 사용하여 수요에 맞게 크기를 조정할 수 있습니다.

Auto Scaling 그룹은 원하는 용량을 충족하도록 충분한 인스턴스를 실행하여 시작합니다. 그룹 내 인스턴스에 대한 주기적인 상태 확인을 수행하여 이 인스턴스 수를 유지합니다. Auto Scaling 그룹은 인스턴스 상태가 이상이 있는 경우에도 고정된 수의 인스턴스를 계속 유지합니다. 인스턴스가 비정상 상태가 되면 그룹에서는 비정상 인스턴스를 종료하고 이를 대체할 다른 인스턴스를 시작합니다. 자세한 내용은 [Auto Scaling 인스턴스 상태 확인 \(p. 148\)](#) 단원을 참조하십시오.

조정 정책을 사용하여 바뀌는 조건을 충족하도록 그룹의 인스턴스 수를 동적으로 늘리거나 줄일 수 있습니다. 조정 정책의 효력이 발생되면, Auto Scaling 그룹이 해당 그룹의 희망 용량을 사용자가 지정하는 최소 및 최대 용량 값 사이에서 조절하고 필요에 따라 인스턴스를 시작 또는 종료합니다. 일정에서도 확장이 가능합니다. 자세한 내용은 [Auto Scaling 그룹의 크기 조정 \(p. 88\)](#) 단원을 참조하십시오.

Auto Scaling 그룹은 온디맨드 인스턴스, 스팟 인스턴스 또는 둘 모두를 시작할 수 있습니다. 시작 템플릿을 사용하도록 그룹을 구성 하는 경우에만 Auto Scaling 그룹에 대해 여러 구매 옵션을 지정할 수 있습니다. (Amazon EC2의 최신 기능을 사용하려면 시작 구성 대신 시작 템플릿을 사용하는 것이 좋습니다.)

스팟 인스턴스를 사용하면 온디맨드 가격과 비교하여 대폭 할인된 금액으로 미사용 Amazon EC2 용량에 액세스할 수 있습니다. 자세한 내용은 [Amazon EC2 스팟 인스턴스](#)를 참조하십시오.. 스팟 인스턴스와 온 디맨드 인스턴스 간에는 다음과 같은 주요 차이점이 있습니다.

- 스팟 인스턴스의 가격은 수요에 따라 변동합니다.
- 스팟 인스턴스의 가용성 또는 가격이 변동함에 따라 Amazon EC2가 개별 스팟 인스턴스를 종료할 수 있습니다.

스팟 인스턴스가 종료되면 Auto Scaling 그룹에서는 대체 인스턴스를 시작하여 그룹의 원하는 용량을 유지하려고 합니다.

인스턴스가 시작되고 사용자가 여러 가용 영역을 지정한 경우, 희망 용량이 이러한 가용 영역에 분산됩니다. 크기 조정 작업이 발생하는 경우, 사용자가 지정한 모든 가용 영역에서 Amazon EC2 Auto Scaling이 자동으로 균형을 유지합니다.

이 섹션에서는 Auto Scaling 그룹 생성을 위한 단계를 상세히 설명합니다. Auto Scaling 그룹에 대해 익숙하지 않은 사용자는 [Amazon EC2 Auto Scaling 시작하기 \(p. 12\)](#)를 시작하고 Amazon EC2 Auto Scaling에서 사용되는 다양한 구성 요소에 대해 알아보십시오.

목차

- 여러 인스턴스 유형 및 구매 옵션이 포함된 Auto Scaling 그룹 (p. 45)
- 시작 템플릿을 사용하여 Auto Scaling 그룹 생성하기 (p. 54)
- 시작 구성을 사용하여 Auto Scaling 그룹 생성하기 (p. 57)
- EC2 인스턴스를 사용하여 Auto Scaling 그룹 만들기 (p. 59)
- Amazon EC2 시작 마법사를 사용하여 Auto Scaling 그룹 만들기 (p. 61)
- Auto Scaling 그룹 및 인스턴스 태그 지정 (p. 62)
- Auto Scaling 그룹에 로드 밸런서 사용 (p. 66)

- [Auto Scaling 그룹에서 스팟 인스턴스 시작 \(p. 73\)](#)
- [Amazon EC2 Auto Scaling 인스턴스 가중치 부여 \(p. 74\)](#)
- [인스턴스 유형에 대한 권장 사항 가져오기 \(p. 79\)](#)
- [최대 Auto Scaling 인스턴스 수명을 기준으로 인스턴스 교체 \(p. 81\)](#)
- [Auto Scaling 그룹을 단일 다중 영역 그룹으로 병합 \(p. 83\)](#)
- [Auto Scaling 인프라 삭제 \(p. 85\)](#)

여러 인스턴스 유형 및 구매 옵션이 포함된 Auto Scaling 그룹

단일 Auto Scaling 그룹 내에서 다수의 온디맨드 인스턴스 및 스팟 인스턴스를 자동으로 확장할 수 있습니다. 스팟 인스턴스 사용에 대한 할인 외에도, 예약 인스턴스와 일치하는 인스턴스 유형을 지정하면 일반 온디맨드 인스턴스의 할인된 가격도 적용됩니다. 온디맨드 인스턴스와 예약 인스턴스의 유일한 차이점은 예약 인스턴스를 사전에 구매해야 한다는 것입니다. 이러한 모든 요소를 결합하면 Amazon EC2 인스턴스의 비용 절감을 최적화하며 동시에 애플리케이션에서 원하는 규모 및 성능을 얻을 수 있습니다.

Auto Scaling 그룹을 생성하는 경우, 우선 시작 템플릿에 있는 공통 구성 파라미터를 지정하고 선택합니다. Auto Scaling 그룹을 구성할 때, 다음을 수행할 수 있습니다.

- 시작할 온디맨드 및 스팟 용량과 온디맨드 기본 부분(선택 사항)을 지정합니다.
- 인스턴스 유형을 하나 이상 선택합니다(필요에 따라 시작 템플릿에 지정된 인스턴스 유형을 재정의).
- 각 인스턴스 유형에 개별 가중치를 할당합니다. 예를 들어 인스턴스 유형마다 서로 다른 vCPU, 메모리, 스토리지 또는 네트워크 대역폭 기능을 제공하는 경우 이렇게 하는 것이 유용할 수 있습니다.
- 각 구매 옵션 내에서 Amazon EC2 Auto Scaling이 인스턴스 유형에 용량을 배포해야 하는 방법을 정의합니다.

여러 가용 영역에서 실행되는 여러 인스턴스 유형에 애플리케이션을 배포하면 가용성이 높아집니다. 한 개의 인스턴스 유형을 사용할 수 있지만, 용량이 부족한 인스턴스 풀에서 인스턴스를 시작하지 않기 위해 몇 개의 인스턴스 유형을 사용하는 것이 가장 좋습니다. 스팟 인스턴스에 대한 Auto Scaling 그룹의 요청을 하나의 스팟 인스턴스 풀에서 충족할 수 없는 경우, 온디맨드 인스턴스를 시작하는 대신 다른 스팟 인스턴스 풀에서 계속 시도하면 스팟 인스턴스의 비용 절감 효과를 활용할 수 있습니다.

할당 전략

다음의 할당 전략은 Auto Scaling 그룹이 가능한 인스턴스 유형에서 온디맨드 및 스팟 용량을 충족하는 방법을 결정합니다.

각 케이스에서 Amazon EC2 Auto Scaling은 우선 지정한 가용 영역에서 인스턴스가 고르게 균형이 유지되어 있는지를 확인합니다. 그 다음, 지정한 할당 전략에 따라 인스턴스 유형을 시작합니다.

온디맨드 인스턴스

온디맨드 인스턴스에 대한 할당 전략은 prioritized Amazon EC2 Auto Scaling에서 시작 템플릿 재정의 목록에 있는 인스턴스 유형 순서를 사용하여 온디맨드 용량을 채울 때 우선 사용할 인스턴스 유형을 결정합니다.

예를 들어 세 가지 시작 템플릿 재정의의 `c5.large`, `c4.large`, `c3.large` 순으로 지정한 것으로 가정해 보겠습니다. 온디맨드 인스턴스가 시작되면 Auto Scaling 그룹이 `c5.large`, `c4.large`, `c3.large` 순으로 시작하여 온디맨드 용량을 채웁니다. `c4.large`에서 사용하지 않은 예약 인스턴스가 있는 경우, 인스턴스 유형의 우선순위를 설정하고 `c4.large`를 최우선 인스턴스 유형으로 지정함으로써 예약 인스턴스에 가장 높은 순위를 부여할 수 있습니다. `c4.large` 인스턴스가 시작되면 예약 인스턴스 요금을 받게 됩니다.

스팟 인스턴스

Amazon EC2 Auto Scaling에서는 스팟 인스턴스에서 이용할 수 있는 두 가지 유형의 할당 전략을 제공합니다.

capacity-optimized

Amazon EC2 Auto Scaling이 시작하는 인스턴스의 수에 대한 최적의 용량의 스팟 인스턴스 풀에서 인스턴스를 할당합니다. 이러한 배포 방식은 예비 EC2 용량을 가장 효율적으로 활용하는 데 도움이 됩니다.

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.

lowest-price

Amazon EC2 Auto Scaling에서는 사용자가 지정한 수(N)의 스팟 인스턴스 풀에서 인스턴스를 할당하며, 충족 시점에 단위당 가격이 가장 낮은 풀에서 제공됩니다.

예를 들어 인스턴스 유형 4개와 가용 영역 4개를 지정한 경우, Auto Scaling 그룹은 16개나 되는 스팟 인스턴스 풀에서 얻을 가능성이 있습니다. 할당 전략에서 스팟 풀(N=2) 2개를 지정한 경우, Auto Scaling 그룹은 최소 8개의 유닛당 가격이 가장 낮은 다른 스팟 풀에서 스팟 용량을 충족할 수 있습니다.

시작하려면 capacity-optimized 할당 전략을 선택하고 애플리케이션에 적절한 몇 개의 인스턴스 유형을 설정할 것을 권장합니다. 또한, 인스턴스 시작 시 선택할 Amazon EC2 Auto Scaling에 대한 가용 영역 범위도 정의할 수 있습니다.

원할 경우, 스팟 인스턴스의 최고 가격을 지정할 수 있습니다. 최대 가격을 지정하지 않을 경우 기본 최대 가격은 온디맨드 가격이지만, 여전히 스팟 인스턴스가 제공하는 높은 할인을 적용받습니다. 이러한 할인은 새로운 [스팟 요금 모델](#)을 사용해 실현된 안정적인 스팟 요금 때문에 가능해진 것입니다.

스팟 인스턴스에 대한 할당 전략에 대한 자세한 내용은 AWS 블로그의 [Amazon EC2 스팟 인스턴스에 대한 용량 최적화 할당 전략 소개](#)를 참조하십시오.

온디맨드 인스턴스 비율 제어

온디맨드 실행되는 Auto Scaling 그룹의 인스턴스 비율을 완전히 제어할 수 있습니다. 인스턴스 용량을 항상 확보하려면 온디맨드 인스턴스로 시작할 그룹의 백분율을 지정하고, 필요한 경우 온디맨드 인스턴스의 처음 시작 기본 개수를 선택합니다. 온디맨드 인스턴스의 기본 용량을 지정하기로 한 경우, Auto Scaling 그룹은 그룹 확장 시 이 온디맨드 인스턴스 기본 용량이 먼저 시작되도록 합니다. 기본 용량을 초과하는 용량은 온디맨드 백분율을 사용하여 시작할 온디맨드 인스턴스 및 스팟 인스턴스 수를 결정합니다. 온디맨드 백분율에 0에서 100까지의 숫자를 지정할 수 있습니다.

이 수가 증가하면 Auto Scaling 그룹은 다음과 같이 동작합니다.

예: 확장 동작

인스턴스 배포	구매 옵션 전체에서 실행되는 인스턴스의 총 수			
	10	20	30	40
예 1				
온디맨드 기본: 10	10	10	10	10
기본을 초과하는 온디맨드 비율: 50%	0	5	10	15

인스턴스 배포	구매 옵션 전체에서 실행되는 인스턴스의 총 수			
스팟 비율: 50%	0	5	10	15
예 2				
온디맨드 기본: 0	0	0	0	0
기본을 초과하는 온디맨드 비율: 0%	0	0	0	0
스팟 비율: 100%	10	20	30	40
예 3				
온디맨드 기본: 0	0	0	0	0
기본을 초과하는 온디맨드 비율: 60%	6	12	18	24
스팟 비율: 40%	4	8	12	16
예 4				
온디맨드 기본: 0	0	0	0	0
기본을 초과하는 온디맨드 비율: 100%	10	20	30	40
스팟 비율: 0%	0	0	0	0
예 5				
온디맨드 기본: 12	10	12	12	12
기본을 초과하는 온디맨드 비율: 0%	0	0	0	0
스팟 비율: 100%	0	8	18	28

스팟 인스턴스 모범 사례

Auto Scaling 그룹을 생성하고 스팟 인스턴스를 요청하기 전에 [스팟 모범 사례](#)를 검토합니다. 요청을 계획하여 원하는 인스턴스 유형을 최저 가격으로 프로비저닝하려면 이러한 모범 사례가 필요합니다. 또한, 다음을 수행하는 것이 좋습니다.

- 기본 최고 가격인 온디맨드 가격을 사용하십시오. 시작하는 스팟 인스턴스에 대해 스팟 가격만 지불합니다. 스팟 가격이 최고 가격 이내인 경우, 가용성에 따라 요청이 실행됩니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [요금 및 비용절감](#)을 참조하십시오.
- 여러 인스턴스 유형으로 Auto Scaling 그룹을 생성합니다. 가격은 가용 영역의 각 인스턴스 유형과 관계없이 변동되므로, 인스턴스 유형이 유연한 경우 대부분 같은 가격으로 좀 더 많은 컴퓨팅 파워를 확보할 수 있습니다.
- 마찬가지로 가장 많이 사용되는 인스턴스 유형으로 제한하지 마십시오. 가격은 수요에 따라 변동되므로, 인기 있는 인스턴스 유형(예: 최근에 출시된 인스턴스 패밀리 등)은 요금이 급변하는 경향이 있습니다. 사용 빈도가 낮은 이전 세대 인스턴스 유형을 선택하면 비용을 낮추고 중단 빈도는 줄일 수 있습니다.
- lowest-price 할당 전략을 선택하고 웹 서비스를 실행하는 경우 많은 수의 스팟 풀을 지정합니다(예: N=10). 많은 수의 스팟 풀을 지정하면 가용 영역 중 하나에 있는 풀을 일시적으로 사용할 수 없게 될 경우 스팟 인스턴스 중단의 영향을 줄일 수 있습니다. 일괄 처리나 다른 미션 크리티컬하지 않은 애플리케이션

을 실행하는 경우, 스팟 풀을 더 작은 수(예 : $N = 2$)로 지정할 수 있습니다. 이렇게 하면 가용 영역별로 사용할 수 있는 가장 저렴한 가격의 스팟 풀에서만 스팟 인스턴스를 프로비저닝할 수 있습니다.

- 스팟 인스턴스 중단 공지를 사용하여 스팟 인스턴스의 상태를 모니터링합니다. 예를 들어 Amazon CloudWatch Events에서 Amazon SNS 주제, AWS Lambda 함수 또는 기타 대상에 EC2 스팟 2분 경고를 자동으로 보내는 규칙을 설정할 수 있습니다. 자세한 내용은 [Amazon CloudWatch Events 사용 설명서](#) 및 Linux 인스턴스용 Amazon EC2 사용 설명서의 [스팟 인스턴스 중단 공지](#)를 참조하십시오.

최고 가격을 지정하려면 AWS CLI 또는 API를 사용하여 Auto Scaling 그룹을 만들면 되지만 주의해야 합니다. 최고 가격이 선택한 인스턴스 유형의 스팟 가격보다 낮으면 스팟 인스턴스가 시작되지 않습니다.

사전 조건

시작 템플릿이 Auto Scaling 그룹과 함께 사용되도록 구성합니다. 자세한 정보는 [Auto Scaling 그룹을 위한 시작 템플릿 만들기 \(p. 26\)](#) 단원을 참조하십시오.

IAM 사용자는 `ec2:RunInstances` 작업을 호출 할 수 있는 권한이 있는 경우에만 시작 템플릿을 사용하여 Auto Scaling 그룹을 생성할 수 있습니다. 자세한 내용은 [Amazon EC2 Auto Scaling 자격 증명 기반 정책 예제 \(p. 179\)](#) 단원을 참조하십시오.

여러 구매 옵션으로 Auto Scaling 그룹 생성

다음 단계를 수행하고, 확장 가능한 온디맨드 인스턴스 및 스팟 인스턴스 풀릿을 생성합니다.

Amazon EC2 Auto Scaling에서 Auto Scaling 그룹 인터페이스가 변경되었습니다. 기본적으로 기존 사용자 인터페이스가 표시되지만 새 사용자 인터페이스로 전환할 수 있습니다. 이 주제에는 둘 모두에 대한 단계가 포함되어 있습니다.

여러 구매 옵션으로 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 AWS 리전과 동일한 리전을 선택합니다.
3. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
4. Auto Scaling 그룹 생성을 선택합니다.
5. 시작 템플릿 또는 구성 선택 페이지에서 다음을 수행합니다.
 - a. Auto Scaling 그룹 이름에서 Auto Scaling 그룹의 이름을 입력합니다.
 - b. 시작 템플릿에서 기존 시작 템플릿을 선택합니다.
 - c. 시작 템플릿 버전에서, Auto Scaling 그룹이 확장 시 시작 템플릿의 기본 버전을 사용할지, 최신 버전을 사용할지 또는 특정 버전을 사용할지를 선택합니다.
 - d. 시작 템플릿이 사용하려는 모든 옵션을 지원하는지 확인한 후 다음을 선택합니다.
6. 설정 구성 페이지에서 구매 옵션 및 인스턴스 유형에 대해 구매 옵션 및 인스턴스 유형 결합을 선택합니다.
7. 인스턴스 배포에서 다음을 수행합니다.

기본 설정을 유지하려면 이러한 단계를 건너뛸 수 있습니다.

- a. 선택적 온디맨드 기본에서 온디맨드 인스턴스에서 충족해야 하는 Auto Scaling 그룹의 초기 용량에 대한 최소 인스턴스 수를 지정합니다.
- b. 기본을 초과하는 온디맨드 비율에서 선택적 온디맨드 기본 양을 넘어서는 추가 용량에 대한 온디맨드 인스턴스 및 스팟 인스턴스의 비율을 지정합니다.
- c. 가용 영역별 스팟 할당 전략의 경우 용량 최적화의 기본 설정을 유지하는 것이 좋습니다. 기본값을 유지하지 않으려면 최저 가격을 선택한 다음 분산할 최저 가격의 스팟 인스턴스 풀 수를 입력합니다.

8. 인스턴스 유형의 경우 권장 사항을 시작점으로 사용하여 시작할 수 있는 인스턴스 유형을 선택합니다. 이렇게 하지 않는 경우 인스턴스 유형을 삭제하고 필요에 따라 나중에 추가할 수 있습니다.
9. (선택 사항) 인스턴스 유형의 순서를 변경하려면 화살표를 사용합니다. 인스턴스 유형을 추가하는 순서로 온디맨드 인스턴스의 우선 순위가 결정됩니다. 목록 맨 위의 인스턴스 유형은 Auto Scaling 그룹이 온디맨드 용량을 시작할 때 가장 우선적으로 처리됩니다.
10. (선택 사항) [인스턴스 가중치 \(p. 74\)](#)를 사용하려면 인스턴스를 Auto Scaling 그룹의 용량에 얼마나 가산할지에 해당하는 상대적 가중치를 각 인스턴스 유형에 할당합니다.
11. 네트워크에서 VPC에 대해 시작 템플릿에 지정한 보안 그룹의 VPC를 선택합니다. 인스턴스 유형과 구매 옵션을 조합한 인스턴스 시작은 EC2-Classic에서 지원되지 않습니다.
12. 서브넷은 지정한 VPC에 있는 서브넷 하나 이상을 선택합니다. 여러 가용 영역의 서브넷을 사용하여 가용성을 높일 수 있습니다. Amazon EC2 Auto Scaling을 사용한 고가용성에 대한 자세한 내용은 [가용 영역에 인스턴스 분산 \(p. 6\)](#)을 참조하십시오.
13. 다음을 선택합니다.

또는 나머지 기본값을 그대로 두고, 검토로 이동을 선택할 수 있습니다.
14. 로드 밸런싱 및 상태 확인 지정 페이지에서 다음 옵션을 구성하고 다음을 선택합니다.
 - a. (선택 사항) Elastic Load Balancing(ELB) 로드 밸런서에 Amazon EC2 인스턴스를 등록하려면 로드 밸런싱 활성화를 선택합니다. Application Load Balancer 또는 Network Load Balancer를 연결하려면 기존 대상 그룹을 선택하거나 새 대상 그룹을 생성합니다. Classic Load Balancer를 연결하려면 기존 로드 밸런서를 선택하거나 새 로드 밸런서를 생성합니다.
 - b. (선택 사항) ELB 상태 확인을 활성화하려면 상태 확인에 대해 상태 확인 유형에서 ELB를 선택합니다.
 - c. (선택 사항) 상태 확인 유예 기간에 인스턴스가 서비스에 배치된 후 Amazon EC2 Auto Scaling가 인스턴스의 상태를 확인할 때까지 걸리는 시간을 입력합니다. 이는 Amazon EC2 Auto Scaling가 인스턴스를 비정상으로 표시하여 작업 처리 전에 이를 종료하지 않도록 하기 위한 것입니다. 기본값은 300초입니다.
15. 그룹 크기 및 조정 정책 구성 페이지에서 다음 옵션을 구성하고 다음을 선택합니다.
 - a. (선택 사항) 원하는 용량에 대해 시작할 초기 인스턴스 수를 입력합니다. 이 수를 최소 또는 최대 용량 제한을 벗어나는 값으로 변경하는 경우 최소 용량 또는 최대 용량 값을 업데이트해야 합니다. 자세한 내용은 [Auto Scaling 그룹에 대한 용량 제한 설정 \(p. 89\)](#) 단원을 참조하십시오.
 - b. (선택 사항) Auto Scaling 그룹 크기를 자동으로 조정하려면 대상 추적 조정 정책을 선택하고 지침을 따릅니다. 자세한 내용은 [대상 추적 조정 정책 \(p. 106\)](#)을 참조하십시오.
 - c. (선택 사항) 인스턴스 축소 보호에서 인스턴스 축소 보호를 활성화할지 여부를 선택합니다. 자세한 내용은 [인스턴스 축소 보호 \(p. 128\)](#) 단원을 참조하십시오.
16. (선택 사항) 알림을 받으려면 알림 추가에 대해 알림을 구성하고 다음을 선택합니다. 자세한 내용은 [Auto Scaling 그룹 조정 시 Amazon SNS SNS 알림 받기 \(p. 150\)](#) 단원을 참조하십시오.
17. (선택 사항) 태그를 추가하려면 태그 추가를 선택하고 각 태그에 태그 키와 값을 제공한 후 다음을 선택합니다. 자세한 내용은 [Auto Scaling 그룹 및 인스턴스 태그 지정 \(p. 62\)](#) 단원을 참조하십시오.
18. 검토 페이지에서 Auto Scaling 그룹 생성을 선택합니다.

여러 구매 옵션으로 Auto Scaling 그룹 생성(이전 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 시작 템플릿을 만들 때 사용한 AWS 리전과 동일한 리전을 선택합니다.
3. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
4. Auto Scaling 그룹 생성을 선택합니다.
5. 시작 템플릿을 선택하고, 시작 템플릿을 선택한 후 다음 단계를 선택합니다.
6. Configure Auto Scaling group details (그룹 세부 정보 구성) 페이지에서 그룹 이름에 Auto Scaling 그룹의 이름을 입력합니다.

7. 시작 템플릿 버전에서, Auto Scaling 그룹이 확장 시 시작 템플릿의 기본 버전을 사용할지, 최신 버전을 사용할지 또는 특정 버전을 사용할지를 선택합니다.
8. 플릿 구성에서, Combine purchase options and instances(구매 옵션과 인스턴스 결합)을 선택하고 온디맨드 및 스팟 구매 옵션을 사용하여 여러 인스턴스 유형에 걸쳐 인스턴스를 실행합니다.
9. [인스턴스 유형]에서 다음을 수행합니다.
 - a. (선택 사항) 재정으로 사용할 인스턴스 유형을 선택합니다. 기본적으로 단일 인스턴스 유형은 선택한 시작 템플릿에서 지정되지만 다른 인스턴스 유형을 지정하려는 경우 변경할 수 있습니다. 여러 인스턴스 유형을 선택할 때 인스턴스 유형을 추가하는 순서는 온디맨드 인스턴스의 우선 순위를 설정합니다. 목록 맨 위의 인스턴스 유형은 Auto Scaling 그룹이 온디맨드 용량을 시작할 때 가장 우선적으로 처리됩니다. 이렇게 하지 않는 경우 지금 인스턴스 유형을 제거하고 필요에 따라 나중에 추가할 수 있습니다.
 - b. (선택 사항) [인스턴스 가중치 \(p. 74\)](#)를 사용하려면 인스턴스를 Auto Scaling 그룹의 용량에 얼마나 가산할지에 해당하는 상대적 가중치를 각 인스턴스 유형에 할당합니다.
10. 인스턴스 배포에서 기본 인스턴스 배포 설정을 유지할지 바꾸지 선택합니다. Use the default settings to get started quickly(기본 설정을 사용하여 빠르게 시작하기) 상자의 선택을 취소하면 기본 설정을 편집할 수 있습니다.
11. 기본 설정을 편집할 경우에는 다음 정보를 제공합니다.
 - 최고 스팟 가격에서, 기본값 사용을 선택하고 온디맨드 가격으로 최대 스팟 가격을 제한합니다. 혹은 Set your maximum price(최고 가격 설정)를 선택하고 값을 입력하여 스팟 인스턴스에 대해 시간당 인스턴스 한 개에 지불할 최대 가격을 지정합니다.
 - 스팟 할당 전략의 경우, 기본값은 Launch Spot Instances optimally based on the available Spot capacity per Availability Zone(가용 영역당 사용 가능한 스팟 용량에 따라 최적으로 스팟 인스턴스를 시작)입니다. 가장 저렴한 전략을 사용하려면 Diversify Spot Instances across your N lowest priced instance types per Availability Zone(가용 영역당 N개의 가장 저렴한 인스턴스 유형으로 스팟 인스턴스를 분산)을 선택하고 사용할 스팟 풀 수를 입력하거나 기본값(2)을 수락합니다.
 - Optional On-Demand Base(선택적 온디맨드 기본)에서, 온디맨드 인스턴스에서 충족해야 하는 Auto Scaling 그룹의 초기 용량 최소 인스턴스 수를 지정합니다.
 - 기본을 초과하는 온디맨드 비율에서 선택적 온디맨드 기본 양을 넘어서는 추가 용량에 대한 온디맨드 인스턴스 및 스팟 인스턴스의 비율을 지정합니다.
12. 그룹 크기에 Auto Scaling 그룹의 초기 인스턴스 수를 입력합니다.
13. 네트워크는 Auto Scaling 그룹의 VPC를 선택합니다.

Note

인스턴스 유형과 구매 옵션을 조합한 인스턴스 시작은 EC2-Classic에서 지원되지 않습니다.

14. 서브넷은 지정된 VPC에 있는 서브넷 하나 이상을 선택합니다. 여러 가용 영역의 서브넷을 사용하여 가용성을 높일 수 있습니다. Amazon EC2 Auto Scaling을 사용한 고가용성에 대한 자세한 내용은 [가용 영역에 인스턴스 분산 \(p. 6\)](#)을 참조하십시오.
15. (선택 사항) 로드 밸런서에 Amazon EC2 인스턴스를 등록하려면, 고급 세부 정보를 선택하고 Receive traffic from one or more load balancers(하나 이상의 로드 밸런서에서 트래픽 수신), 그리고 하나 이상의 Classic Load Balancer 또는 대상 그룹을 선택합니다.
16. Next: Configure scaling policies(다음: 조정 정책 구성)를 선택합니다.
17. Configure scaling policies(조정 정책 구성) 페이지에서 다음 옵션 중 하나를 선택한 다음 Next: Configure Notifications(다음: 구성 알림)를 선택합니다.
 - Auto Scaling 그룹의 크기를 필요에 따라 수동으로 조정하려면 Keep this group at its initial size(이 그룹을 초기 크기로 유지)를 선택합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 수동 조정 \(p. 90\)](#) 단원을 참조하십시오.
 - 지정하는 기준에 따라 Auto Scaling 그룹의 크기를 자동으로 조정하려면 Use scaling policies to adjust the capacity of this group(조정 정책을 사용하여 이 그룹의 용량 조정)을 선택한 후 지침을 따릅니다. 자세한 내용은 [조정 정책 구성 \(p. 106\)](#)을 참조하십시오.

18. (선택 사항) 알림을 받으려면 [Add notification]을 선택하고 알림을 구성한 다음 [Next: Configure Tags]를 선택합니다.
19. (선택 사항) 태그를 추가하려면, [Edit tags]를 선택하고 각 태그에 태그 키와 값을 제공한 다음 [Review]를 선택합니다.

또는 나중에 태그를 추가할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹 및 인스턴스 태그 지정 \(p. 62\)](#) 단원을 참조하십시오.

20. 검토 페이지에서 Auto Scaling 그룹 생성을 선택합니다.
21. Auto Scaling 그룹 생성 상태 페이지에서 닫기를 선택합니다.

명령줄을 사용해 여러 구매 옵션으로 Auto Scaling 그룹 생성

다음 명령 중 하나를 사용할 수 있습니다:

- `create-auto-scaling-group` (AWS CLI)
- `New-ASAutoScalingGroup`(Windows PowerShell용 AWS 도구)

예: **capacity-optimized** 할당 전략을 사용한 스팟 인스턴스 시작

다음 `create-auto-scaling-group` 명령은 다음을 지정하는 Auto Scaling 그룹을 생성합니다.

- 온디맨드 인스턴스로 시작할 그룹의 백분율(0) 및 온디맨드 인스턴스의 처음 시작 기본 개수(1)
- 우선순위(c3.large, c4.large, c5.large)에 따라 시작할 인스턴스 유형
- 각각 다른 가용 영역에 해당하는 인스턴스를 시작하는 서브넷(subnet-5ea0c127, subnet-6194ea3b, subnet-c934b782)
- 시작 템플릿(my-launch-template) 및 시작 템플릿 버전(\$Default)

Amazon EC2 Auto Scaling가 온디맨드 용량을 채우려고 시도하는 경우 c3.large 인스턴스 유형을 먼저 시작합니다. 스팟 인스턴스는 스팟 인스턴스 용량에 따라 각 가용 영역의 최적의 스팟 풀에서 나옵니다.

```
aws autoscaling create-auto-scaling-group --cli-input-json file:///~/config.json
```

다음은 예제 config.json 파일입니다.

```
{
  "AutoScalingGroupName": "my-asg",
  "MixedInstancesPolicy": {
    "LaunchTemplate": {
      "LaunchTemplateSpecification": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "$Default"
      },
      "Overrides": [
        {
          "InstanceType": "c3.large"
        },
        {
          "InstanceType": "c4.large"
        },
        {
          "InstanceType": "c5.large"
        }
      ]
    },
    "InstancesDistribution": {
      "OnDemandBaseCapacity": 1,
      "OnDemandPercentageAboveBaseCapacity": 0,

```

```
        "SpotAllocationStrategy": "capacity-optimized"
    },
    "MinSize": 1,
    "MaxSize": 5,
    "DesiredCapacity": 3,
    "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
    "Tags": []
}
```

예: 두 개의 풀로 분산된 **lowest-price** 할당 전략을 사용한 스팟 인스턴스 시작

다음 **create-auto-scaling-group** 명령은 다음을 지정하는 Auto Scaling 그룹을 생성합니다.

- 온디맨드 인스턴스의 처음 시작 기본 개수를 지정하지 않고 온디맨드 인스턴스로 시작할 그룹의 백분율 (50)
- 우선순위(c3.large, c4.large, c5.large)에 따라 시작할 인스턴스 유형
- 각각 다른 가용 영역에 해당하는 인스턴스를 시작하는 서브넷(subnet-5ea0c127, subnet-6194ea3b, subnet-c934b782)
- 시작 템플릿(my-launch-template) 및 시작 템플릿 버전(\$Latest)

Amazon EC2 Auto Scaling가 온디맨드 용량을 채우려고 시도하는 경우 c3.large 인스턴스 유형을 먼저 시작합니다. 스팟 용량의 경우, Amazon EC2 Auto Scaling에서는 각 가용 영역의 두 개의 가장 낮은 가격 풀에서 스팟 인스턴스를 고르게 시작하려고 시도합니다.

```
aws autoscaling create-auto-scaling-group --cli-input-json file://~/config.json
```

다음은 예제 config.json 파일입니다.

```
{
  "AutoScalingGroupName": "my-asg",
  "MixedInstancesPolicy": {
    "LaunchTemplate": {
      "LaunchTemplateSpecification": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "$Latest"
      },
      "Overrides": [
        {
          "InstanceType": "c3.large"
        },
        {
          "InstanceType": "c4.large"
        },
        {
          "InstanceType": "c5.large"
        }
      ]
    },
    "InstancesDistribution": {
      "OnDemandPercentageAboveBaseCapacity": 50,
      "SpotAllocationStrategy": "lowest-price",
      "SpotInstancePools": 2
    }
  },
  "MinSize": 1,
  "MaxSize": 5,
  "DesiredCapacity": 3,
  "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
  "Tags": []
}
```

```
}
```

그룹에서 새 인스턴스를 시작했는지 확인하기 위해

다음 [describe-auto-scaling-groups](#) 명령을 사용합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-names my-asg
```

다음 예의 응답은 원하는 용량이 3이고 해당 그룹에 실행 중인 인스턴스가 3개임을 보여 줍니다.

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "ServiceLinkedRoleARN": "arn",
      "TargetGroupARNs": [],
      "SuspendedProcesses": [],
      "DesiredCapacity": 3,
      "MixedInstancesPolicy": {
        "InstancesDistribution": {
          "SpotAllocationStrategy": "lowest-price",
          "OnDemandPercentageAboveBaseCapacity": 50,
          "OnDemandAllocationStrategy": "prioritized",
          "SpotInstancePools": 2,
          "OnDemandBaseCapacity": 0
        },
        "LaunchTemplate": {
          "LaunchTemplateSpecification": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "$Latest",
            "LaunchTemplateId": "lt-050555ad16a3f9c7f"
          },
          "Overrides": [
            {
              "InstanceType": "c3.large"
            },
            {
              "InstanceType": "c4.large"
            },
            {
              "InstanceType": "c5.large"
            }
          ]
        }
      },
      "EnabledMetrics": [],
      "Tags": [],
      "AutoScalingGroupName": "my-asg",
      "DefaultCooldown": 300,
      "MinSize": 1,
      "Instances": [
        {
          "ProtectedFromScaleIn": false,
          "AvailabilityZone": "us-west-2a",
          "LaunchTemplate": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1",
            "LaunchTemplateId": "lt-050555ad16a3f9c7f"
          },
          "InstanceId": "i-0aae8709d49eeba4f",
          "HealthStatus": "Healthy",
          "LifecycleState": "InService"
        },
        {
```

```
    "ProtectedFromScaleIn": false,
    "AvailabilityZone": "us-west-2b",
    "LaunchTemplate": {
      "LaunchTemplateName": "my-launch-template",
      "Version": "1",
      "LaunchTemplateId": "lt-050555ad16a3f9c7f"
    },
    "InstanceId": "i-0c43f6003841d2d2b",
    "HealthStatus": "Healthy",
    "LifecycleState": "InService"
  },
  {
    "ProtectedFromScaleIn": false,
    "AvailabilityZone": "us-west-2c",
    "LaunchTemplate": {
      "LaunchTemplateName": "my-launch-template",
      "Version": "1",
      "LaunchTemplateId": "lt-050555ad16a3f9c7f"
    },
    "InstanceId": "i-0feb4cd6677d39903",
    "HealthStatus": "Healthy",
    "LifecycleState": "InService"
  }
],
"MaxSize": 5,
"VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
"HealthCheckGracePeriod": 0,
"TerminationPolicies": [
  "Default"
],
"LoadBalancerNames": [],
"CreatedTime": "2019-02-17T02:29:12.853Z",
"AvailabilityZones": [
  "us-west-2a",
  "us-west-2b",
  "us-west-2c"
],
"HealthCheckType": "EC2",
"NewInstancesProtectedFromScaleIn": false
}
]
```

추가 예제는 다음([Amazon EC2 Auto Scaling 인스턴스 가중치 부여 \(p. 74\)](#))을 참조하십시오.

시작 템플릿을 사용하여 Auto Scaling 그룹 생성하기

Auto Scaling 그룹에서 시작할 Amazon EC2 인스턴스를 구성하려면 시작 템플릿, 시작 구성, 또는 EC2 인스턴스를 지정할 수 있습니다. 다음 절차는 시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하는 방법을 보여줍니다.

시작 템플릿에서는 Auto Scaling 그룹이 확장 이벤트가 발생할 때 시작 템플릿의 기본 버전 또는 최신 버전을 동적으로 사용하도록 구성할 수 있습니다. 예를 들어, Auto Scaling 그룹을 구성하여 시작 템플릿의 현재 기본 버전을 선택할 수 있습니다. 그룹에서 시작할 EC2 인스턴스의 구성을 변경하려면 시작 템플릿의 기본 버전을 새로 생성하거나 지정합니다. 또는 그룹에서 EC2 인스턴스를 시작하는 데 사용하는 시작 템플릿의 특정 버전을 선택할 수 있습니다. 이러한 설정은 그룹을 업데이트하여 언제든지 변경할 수 있습니다.

각 시작 템플릿에는 AMI 및 인스턴스 유형과 같이 Amazon EC2가 인스턴스를 시작하는 데 필요한 정보가 포함되어 있습니다. 시작 템플릿을 준수하는 Auto Scaling 그룹을 만들 수 있습니다. 또는 시작 템플릿의 인스턴스 유형을 재정의하고 온디맨드 인스턴스와 스팟 인스턴스를 결합할 수 있습니다. 자세한 내용은 [여러 인스턴스 유형 및 구매 옵션이 포함된 Auto Scaling 그룹 \(p. 45\)](#) 단원을 참조하십시오.

Auto Scaling 그룹은 원하는 용량과 가용 영역 및 VPC 서브넷과 같이 Amazon EC2가 인스턴스를 시작하는 데 필요한 추가 정보를 지정합니다. 고정된 수의 인스턴스로 용량을 설정하거나, 실제 수요를 근거로 용량을 조정하는 자동 조정을 활용할 수 있습니다.

사전 조건

- EC2 인스턴스를 시작하는 데 필요한 파라미터를 포함하는 시작 템플릿이 있어야 합니다. 이러한 파라미터와 Auto Scaling 그룹에 사용할 시작 템플릿을 생성할 때 적용되는 제한에 대한 자세한 내용은 [Auto Scaling 그룹을 위한 시작 템플릿 만들기 \(p. 26\)](#) 단원을 참조하십시오.
- 시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하고 인스턴스의 EC2 리소스도 생성할 수 있는 인스턴스 IAM 권한이 있어야 합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 자격 증명 기반 정책 예제 \(p. 179\)](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling에서 Auto Scaling 그룹 인터페이스가 변경되었습니다. 기본적으로 기존 사용자 인터페이스가 표시되지만 새 사용자 인터페이스로 전환할 수 있습니다. 이 주제에는 둘 모두에 대한 단계가 포함되어 있습니다.

시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하려면(새 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 시작 템플릿을 만들 때 사용한 AWS 리전과 동일한 리전을 선택합니다.
3. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
4. Auto Scaling 그룹 생성을 선택합니다.
5. 시작 템플릿 또는 구성 선택 페이지에서 다음을 수행합니다.
 - a. Auto Scaling 그룹 이름에서 Auto Scaling 그룹의 이름을 입력합니다.
 - b. 시작 템플릿에서 기존 시작 템플릿을 선택합니다.
 - c. 시작 템플릿 버전에서, Auto Scaling 그룹이 확장 시 시작 템플릿의 기본 버전을 사용할지, 최신 버전을 사용할지 또는 특정 버전을 사용할지를 선택합니다.
 - d. 시작 템플릿이 사용하려는 모든 옵션을 지원하는지 확인한 후 다음을 선택합니다.
6. 설정 구성 페이지에서 구매 옵션 및 인스턴스 유형에 대해 시작 템플릿 준수를 선택하여 시작 템플릿에 지정된 EC2 인스턴스 유형 및 구매 옵션을 사용합니다.
7. 네트워크에서 VPC에 대해 시작 템플릿에 지정한 보안 그룹의 VPC를 선택합니다. 인스턴스 유형과 구매 옵션을 조합한 인스턴스 시작은 EC2-Classic에서 지원되지 않습니다.
8. 서브넷은 지정한 VPC에 있는 서브넷 하나 이상을 선택합니다. 여러 가용 영역의 서브넷을 사용하여 가용성을 높일 수 있습니다. Amazon EC2 Auto Scaling을 사용한 고가용성에 대한 자세한 내용은 [가용 영역에 인스턴스 분산 \(p. 6\)](#)을 참조하십시오.
9. 다음을 선택합니다.

또는 나머지 기본값을 그대로 두고, 검토로 이동을 선택할 수 있습니다.
10. (선택 사항) 로드 밸런싱 및 상태 확인 지정 페이지에서 다음 옵션을 구성하고 다음을 선택합니다.
 - a. 로드 밸런서에 Amazon EC2 인스턴스를 등록하려면 로드 밸런싱 활성화를 선택하고 기존 로드 밸런서를 선택하거나 새 로드 밸런서를 생성합니다. 애플리케이션 로드 밸런서 또는 네트워크 로드 밸런서를 사용하는 경우 인스턴스 ID별로 대상을 등록할 대상 그룹을 선택합니다.
 - b. (선택 사항) Elastic Load Balancing(ELB) 상태 확인을 활성화하려면 상태 확인의 상태 확인 유형에서 ELB를 선택합니다. 이러한 상태 확인은 부하 분산을 활성화할 때 선택 사항입니다.
 - c. 상태 확인 유예 기간에 인스턴스가 서비스에 배치된 후 Amazon EC2 Auto Scaling이 인스턴스의 상태를 확인할 때까지 걸리는 시간을 입력합니다. 이 설정의 의도는 Amazon EC2 Auto Scaling이 인스턴스를 비정상으로 표시하여 작업 처리 전에 이를 종료하지 않도록 하기 위한 것입니다. 기본값은 300초입니다.
11. (선택 사항) 그룹 크기 및 조정 정책 구성 페이지에서 다음 옵션을 구성하고 다음을 선택합니다.

- a. (선택 사항) 원하는 용량에 대해 시작할 초기 인스턴스 수를 입력합니다. 이 수를 최소 또는 최대 용량 제한을 벗어나는 값으로 변경하는 경우 최소 용량 또는 최대 용량 값을 업데이트해야 합니다. 자세한 내용은 [Auto Scaling 그룹에 대한 용량 제한 설정 \(p. 89\)](#) 단원을 참조하십시오.
 - b. Auto Scaling 그룹 크기를 자동으로 조정하려면 대상 추적 조정 정책을 선택하고 지침을 따릅니다. 자세한 내용은 [대상 추적 조정 정책 \(p. 106\)](#)을 참조하십시오.
 - c. 인스턴스 축소 보호에서 인스턴스 축소 보호를 활성화할지 여부를 선택합니다. 자세한 내용은 [인스턴스 축소 보호 \(p. 128\)](#) 단원을 참조하십시오.
12. (선택 사항) 알림을 받으려면 알림 추가에 대해 알림을 구성하고 다음을 선택합니다. 자세한 내용은 [Auto Scaling 그룹 조정 시 Amazon SNS SNS 알림 받기 \(p. 150\)](#) 단원을 참조하십시오.
 13. (선택 사항) 태그를 추가하려면 태그 추가를 선택하고 각 태그에 태그 키와 값을 제공한 후 다음을 선택합니다. 자세한 내용은 [Auto Scaling 그룹 및 인스턴스 태그 지정 \(p. 62\)](#) 단원을 참조하십시오.
 14. 검토 페이지에서 Auto Scaling 그룹 생성을 선택합니다.

시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하려면(이전 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 시작 템플릿을 만들 때 사용한 AWS 리전과 동일한 리전을 선택합니다.
3. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
4. Auto Scaling 그룹 생성을 선택합니다.
5. 시작 템플릿을 선택하고, 시작 템플릿을 선택한 후 다음 단계를 선택합니다.
6. Configure Auto Scaling group details (그룹 세부 정보 구성) 페이지에서 그룹 이름에 Auto Scaling 그룹의 이름을 입력합니다.
7. 시작 템플릿 버전에서, Auto Scaling 그룹이 확장 시 시작 템플릿의 기본 버전을 사용할지, 최신 버전을 사용할지 또는 특정 버전을 사용할지를 선택합니다.
8. 플릿 구성에서 시작 템플릿 준수를 선택하여 시작 템플릿에 지정된 EC2 인스턴스 유형과 구입 옵션을 사용합니다.

Note

Combine purchase options and instances(구매 옵션과 인스턴스 결합)을 선택하여 온디맨드 구입 옵션과 스팟 구입 옵션을 모두 사용하여 여러 인스턴스 유형에 대해 인스턴스를 시작합니다. 자세한 내용은 [여러 인스턴스 유형 및 구매 옵션이 포함된 Auto Scaling 그룹 \(p. 45\)](#)을 참조하십시오.

9. 그룹 크기에 Auto Scaling 그룹의 초기 인스턴스 수를 입력합니다.
10. 네트워크는 Auto Scaling 그룹의 VPC를 선택합니다.
11. 서브넷은 지정된 VPC에 있는 서브넷 하나 이상을 선택합니다. 여러 가용 영역의 서브넷을 사용하여 가용성을 높일 수 있습니다. Amazon EC2 Auto Scaling을 사용한 고가용성에 대한 자세한 내용은 [가용 영역에 인스턴스 분산 \(p. 6\)](#)을 참조하십시오.
12. (선택 사항) 로드 밸런서에 Amazon EC2 인스턴스를 등록하려면, 고급 세부 정보를 선택하고 Receive traffic from one or more load balancers(하나 이상의 로드 밸런서에서 트래픽 수신), 그리고 하나 이상의 Classic Load Balancer 또는 대상 그룹을 선택합니다.
13. Next: Configure scaling policies(다음: 조정 정책 구성)를 선택합니다.
14. Configure scaling policies(조정 정책 구성) 페이지에서 다음 옵션 중 하나를 선택한 다음 Next: Configure Notifications(다음: 구성 알림)를 선택합니다.
 - Auto Scaling 그룹의 크기를 필요에 따라 수동으로 조정하려면 Keep this group at its initial size(이 그룹을 초기 크기로 유지)를 선택합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 수동 조정 \(p. 90\)](#) 단원을 참조하십시오.
 - 지정하는 기준에 따라 Auto Scaling 그룹의 크기를 자동으로 조정하려면 Use scaling policies to adjust the capacity of this group(조정 정책을 사용하여 이 그룹의 용량 조정)을 선택한 후 지침을 따릅니다. 자세한 내용은 [조정 정책 구성 \(p. 106\)](#)을 참조하십시오.

15. (선택 사항) 알림을 받으려면 [Add notification]을 선택하고 알림을 구성한 다음 [Next: Configure Tags]를 선택합니다.
16. (선택 사항) 태그를 추가하려면, [Edit tags]를 선택하고 각 태그에 태그 키와 값을 제공한 다음 [Review]를 선택합니다.

또는 나중에 태그를 추가할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹 및 인스턴스 태그 지정 \(p. 62\)](#) 단원을 참조하십시오.
17. 검토 페이지에서 Auto Scaling 그룹 생성을 선택합니다.
18. Auto Scaling 그룹 생성 상태 페이지에서 닫기를 선택합니다.

명령줄을 사용하여 Auto Scaling 그룹을 생성하려면

다음 명령 중 하나를 사용할 수 있습니다:

- `create-auto-scaling-group` (AWS CLI)
- `New-ASAutoScalingGroup` (Windows PowerShell용 AWS 도구)

시작 구성을 사용하여 Auto Scaling 그룹 생성하기

Auto Scaling 그룹을 생성할 때 Amazon EC2 인스턴스, 인스턴스의 서브넷 및 초기 인스턴스 수를 구성하는데 필요한 정보를 지정해야 합니다.

Important

Amazon EC2 인스턴스를 구성하기 위해 시작 템플릿, 시작 구성, 또는 EC2 인스턴스를 지정할 수 있습니다. Amazon EC2의 최신 기능을 사용하려면 시작 템플릿을 사용하는 것이 좋습니다. 자세한 내용은 [시작 템플릿 \(p. 26\)](#) 단원을 참조하십시오.

다음 절차는 시작 구성을 사용하여 Auto Scaling 그룹을 생성하는 방법을 보여줍니다. 시작 구성을 생성한 후에는 수정할 수가 없지만 그룹에 대한 시작 구성을 교체할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹 시작 구성 변경 \(p. 39\)](#) 단원을 참조하십시오.

사전 요구 사항

시작 구성을 생성합니다. 자세한 내용은 [시작 구성 만들기 \(p. 34\)](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling에서 Auto Scaling 그룹 인터페이스가 변경되었습니다. 기본적으로 기존 사용자 인터페이스가 표시되지만 새 사용자 인터페이스로 전환할 수 있습니다. 이 주제에는 둘 모두에 대한 단계가 포함되어 있습니다.

시작 구성을 사용하여 Auto Scaling 그룹을 생성하려면(새 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 시작 템플릿을 만들 때 사용한 AWS 리전과 동일한 리전을 선택합니다.
3. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
4. Auto Scaling 그룹 생성을 선택합니다.
5. Choose launch template or configuration(시작 템플릿 또는 구성 선택) 페이지의 Auto Scaling group name(Auto Scaling 그룹 이름)에 Auto Scaling 그룹의 이름을 입력합니다.
6. 시작 구성을 선택하려면 다음을 수행합니다.
 - a. 시작 템플릿에서 시작 구성으로 전환을 선택합니다.
 - b. 시작 구성에 대해 기존 시작 구성을 선택합니다.
 - c. 시작 구성이 사용하려는 모든 옵션을 지원하는지 확인한 후 다음을 선택합니다.
7. 설정 구성 페이지에서 네트워크의 VPC에 대해 시작 구성에서 지정한 보안 그룹의 VPC를 선택합니다. 인스턴스 유형과 구매 옵션을 조합한 인스턴스 시작은 EC2-Classic에서 지원되지 않습니다.

8. 서브넷은 지정된 VPC에 있는 서브넷 하나 이상을 선택합니다. 여러 가용 영역의 서브넷을 사용하여 가용성을 높일 수 있습니다. Amazon EC2 Auto Scaling을 사용한 고가용성에 대한 자세한 내용은 [가용 영역에 인스턴스 분산 \(p. 6\)](#)을 참조하십시오.
9. 다음을 선택합니다.

또는 나머지 기본값을 그대로 두고, 검토로 이동을 선택할 수 있습니다.
10. (선택 사항) 로드 밸런싱 및 상태 확인 지정 페이지에서 다음 옵션을 구성하고 다음을 선택합니다.
 - a. 로드 밸런서에 Amazon EC2 인스턴스를 등록하려면 로드 밸런싱 활성화를 선택하고 기존 로드 밸런서를 선택하거나 새 로드 밸런서를 생성합니다. 애플리케이션 로드 밸런서 또는 네트워크 로드 밸런서를 사용하는 경우 인스턴스 ID별로 대상을 등록할 대상 그룹을 선택합니다.
 - b. (선택 사항) Elastic Load Balancing(ELB) 상태 확인을 활성화하려면 상태 확인의 상태 확인 유형에서 ELB를 선택합니다. 이러한 상태 확인은 부하 분산을 활성화할 때 선택 사항입니다.
 - c. 상태 확인 유예 기간에 인스턴스가 서비스에 배치된 후 Amazon EC2 Auto Scaling이 인스턴스의 상태를 확인할 때까지 걸리는 시간을 입력합니다. 이 설정의 의도는 Amazon EC2 Auto Scaling이 인스턴스를 비정상적으로 표시하여 작업 처리 전에 이를 종료하지 않도록 하기 위한 것입니다. 기본값은 300초입니다.
11. (선택 사항) 그룹 크기 및 조정 정책 구성 페이지에서 다음 옵션을 구성하고 다음을 선택합니다.
 - a. (선택 사항) 원하는 용량에 대해 시작할 초기 인스턴스 수를 입력합니다. 이 수를 최소 또는 최대 용량 제한을 벗어나는 값으로 변경하는 경우 최소 용량 또는 최대 용량 값을 업데이트해야 합니다. 자세한 내용은 [Auto Scaling 그룹에 대한 용량 제한 설정 \(p. 89\)](#) 단원을 참조하십시오.
 - b. Auto Scaling 그룹 크기를 자동으로 조정하려면 대상 추적 조정 정책을 선택하고 지침을 따릅니다. 자세한 내용은 [대상 추적 조정 정책 \(p. 106\)](#)을 참조하십시오.
 - c. 인스턴스 축소 보호에서 인스턴스 축소 보호를 활성화할지 여부를 선택합니다. 자세한 내용은 [인스턴스 축소 보호 \(p. 128\)](#) 단원을 참조하십시오.
12. (선택 사항) 알림을 받으려면 알림 추가에 대해 알림을 구성하고 다음을 선택합니다. 자세한 내용은 [Auto Scaling 그룹 조정 시 Amazon SNS SNS 알림 받기 \(p. 150\)](#) 단원을 참조하십시오.
13. (선택 사항) 태그를 추가하려면 태그 추가를 선택하고 각 태그에 태그 키와 값을 제공한 후 다음을 선택합니다. 자세한 내용은 [Auto Scaling 그룹 및 인스턴스 태그 지정 \(p. 62\)](#) 단원을 참조하십시오.
14. 검토 페이지에서 Auto Scaling 그룹 생성을 선택합니다.

시작 구성을 사용하여 Auto Scaling 그룹을 생성하려면(이전 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 화면 상단의 탐색 모음에서 시작 구성을 생성할 때 사용한 AWS 리전과 동일한 리전을 선택합니다.
3. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
4. Auto Scaling 그룹 생성을 선택합니다.
5. Auto Scaling 그룹 생성 페이지에서 시작 구성을 선택하고 기존 시작 구성을 선택한 후 다음 단계를 선택합니다.

Note

시작 구성이 없는 경우, Auto Scaling 그룹을 만드는 단계를 진행하기 전에 먼저 시작 구성을 만들도록 요청하는 메시지가 표시됩니다.

6. Configure Auto Scaling group details(그룹 세부 정보 구성) 페이지에서 다음을 수행합니다.
 - a. 그룹 이름에서 Auto Scaling 그룹의 이름을 입력합니다.
 - b. 그룹 크기에 Auto Scaling 그룹의 초기 인스턴스 수를 입력합니다.
 - c. 네트워크는 Auto Scaling 그룹의 VPC를 선택합니다.
 - d. 서브넷에서 VPC의 서브넷을 하나 이상 선택합니다. 여러 가용 영역의 서브넷을 사용하여 가용성을 높일 수 있습니다. Amazon EC2 Auto Scaling을 사용한 고가용성에 대한 자세한 내용은 [가용 영역에 인스턴스 분산 \(p. 6\)](#)을 참조하십시오.

- e. (선택 사항) 로드 밸런서에 Amazon EC2 인스턴스를 등록하려면, 고급 세부 정보를 선택하고 Receive traffic from one or more load balancers(하나 이상의 로드 밸런서에서 트래픽 수신), 그리고 하나 이상의 Classic Load Balancer 또는 대상 그룹을 선택합니다.
- f. Next: Configure scaling policies(다음: 조정 정책 구성)를 선택합니다.
7. Configure scaling policies(조정 정책 구성) 페이지에서 다음 옵션 중 하나를 선택한 다음 Next: Configure Notifications(다음: 구성 알림)를 선택합니다.
 - Auto Scaling 그룹의 크기를 필요에 따라 수동으로 조정하려면 Keep this group at its initial size(이 그룹을 초기 크기로 유지)를 선택합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 수동 조정 \(p. 90\)](#) 단원을 참조하십시오.
 - 지정하는 기준에 따라 Auto Scaling 그룹의 크기를 자동으로 조정하려면 Use scaling policies to adjust the capacity of this group(조정 정책을 사용하여 이 그룹의 용량 조정)을 선택한 후 지침을 따릅니다. 자세한 내용은 [조정 정책 구성 \(p. 106\)](#)을 참조하십시오.
8. (선택 사항) 알림을 받으려면 [Add notification]을 선택하고 알림을 구성한 다음 [Next: Configure Tags]를 선택합니다.
9. (선택 사항) 태그를 추가하려면, [Edit tags]를 선택하고 각 태그에 태그 키와 값을 제공한 다음 [Review]를 선택합니다.

또는 나중에 태그를 추가할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹 및 인스턴스 태그 지정 \(p. 62\)](#) 단원을 참조하십시오.
10. 검토 페이지에서 Auto Scaling 그룹 생성을 선택합니다.
11. Auto Scaling 그룹 생성 상태 페이지에서 닫기를 선택합니다.

명령줄을 사용하여 Auto Scaling 그룹을 생성하려면

다음 명령 중 하나를 사용할 수 있습니다:

- [create-auto-scaling-group](#) (AWS CLI)
- [New-ASAutoScalingGroup](#) (Windows PowerShell용 AWS 도구)

EC2 인스턴스를 사용하여 Auto Scaling 그룹 만들기

Auto Scaling 그룹을 생성할 때 Amazon EC2 인스턴스, 인스턴스의 서브넷 및 초기 인스턴스 수를 구성하는데 필요한 정보를 지정해야 합니다.

Amazon EC2 인스턴스를 구성하기 위해 시작 구성, 시작 템플릿, 또는 EC2 인스턴스를 지정할 수 있습니다. 다음 절차는 EC2 인스턴스를 사용하여 그룹을 생성하는 방법을 보여줍니다. 시작 구성 또는 시작 템플릿을 사용하려면 [시작 구성을 사용하여 Auto Scaling 그룹 생성하기 \(p. 57\)](#) 또는 [시작 템플릿을 사용하여 Auto Scaling 그룹 생성하기 \(p. 54\)](#)을 참조하십시오.

EC2 인스턴스를 사용하여 Auto Scaling 그룹을 생성하면 Amazon EC2 Auto Scaling에서 사용자를 위해 시작 구성을 생성하고 이 구성을 Auto Scaling 그룹에 연결합니다. 이 시작 구성의 이름은 Auto Scaling 그룹과 동일하며 지정된 인스턴스에서 AMI ID, 인스턴스 유형 및 가용 영역 등의 속성을 가져옵니다.

제한 사항

다음은 EC2 인스턴스에서 Auto Scaling 그룹을 생성할 때 적용되는 제한 사항입니다.

- 식별된 인스턴스에 태그가 있는 경우 해당 태그가 새 Auto Scaling 그룹의 Tags 속성에 복사되지 않습니다.
- Auto Scaling 그룹에는 인스턴스를 시작하는 데 사용된 AMI의 블록 디바이스 매핑이 포함됩니다. 인스턴스 시작 후 연결된 블록 디바이스는 포함되지 않습니다.
- 식별된 인스턴스가 하나 이상의 로드 밸런서에 등록되어 있으면 로드 밸런서에 대한 정보가 새 Auto Scaling 그룹의 로드 밸런서 또는 대상 그룹 속성에 복사되지 않습니다.

사전 조건

시작하기 전에 Amazon EC2 콘솔 또는 [describe-instances](#) 명령(AWS CLI)을 사용하여 EC2 인스턴스의 ID를 확인합니다. EC2 인스턴스는 다음 기준을 충족해야 합니다.

- 인스턴스가 Auto Scaling 그룹을 생성할 가용 영역에 있습니다.
- 인스턴스가 다른 Auto Scaling 그룹의 구성원이 아닙니다.
- 인스턴스는 `running` 상태에 있습니다.
- 인스턴스를 시작할 때 사용되는 AMI가 항상 있어야 합니다.

목차

- [EC2 인스턴스에서 Auto Scaling 그룹 생성\(콘솔\)](#) (p. 60)
- [EC2 인스턴스에서 Auto Scaling 그룹 생성\(AWS CLI\)](#) (p. 60)

EC2 인스턴스에서 Auto Scaling 그룹 생성(콘솔)

콘솔을 사용하여 실행 중인 EC2 인스턴스에서 Auto Scaling 그룹을 생성하고 새 Auto Scaling 그룹에 해당 인스턴스를 추가할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹에 EC2 인스턴스 연결](#) (p. 92) 단원을 참조하십시오.

EC2 인스턴스에서 Auto Scaling 그룹 생성(AWS CLI)

다음 `create-auto-scaling-group` 명령을 사용하여 EC2 인스턴스 `i-7f12e649`에서 `my-asg-from-instance`를 사용하여 Auto Scaling 그룹을 생성합니다.

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg-from-instance \
--instance-id i-7f12e649 --min-size 1 --max-size 2 --desired-capacity 2
```

다음 `describe-auto-scaling-groups` 명령을 사용하여 Auto Scaling 그룹을 설명합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg-from-instance
```

다음 예의 응답은 해당 그룹의 원하는 용량이 2이고 그룹에 실행 중인 인스턴스가 2개이며 시작 구성의 이름이 `my-asg-from-instance`임을 보여 줍니다.

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 0,
      "SuspendedProcesses": [],
      "DesiredCapacity": 2,
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-asg-from-instance",
      "DefaultCooldown": 300,
      "MinSize": 1,
      "Instances": [
        {
          "InstanceId": "i-6bd79d87",
          "AvailabilityZone": "us-west-2a",
          "HealthStatus": "Healthy",
          "LifecycleState": "InService",
          "LaunchConfigurationName": "my-asg-from-instance"
        }
      ]
    }
  ]
}
```

```
    },
    {
      "InstanceId": "i-6cd79d80",
      "AvailabilityZone": "us-west-2a",
      "HealthStatus": "Healthy",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-asg-from-instance"
    }
  ],
  "MaxSize": 2,
  "VPCZoneIdentifier": "subnet-6bea5f06",
  "TerminationPolicies": [
    "Default"
  ],
  "LaunchConfigurationName": "my-asg-from-instance",
  "CreatedTime": "2014-12-29T16:14:50.397Z",
  "AvailabilityZones": [
    "us-west-2a"
  ],
  "HealthCheckType": "EC2"
}
]
```

다음 `describe-launch-configurations` 명령을 사용하여 시작 구성 `my-asg-from-instance`를 설명합니다.

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-asg-from-instance
```

Amazon EC2 시작 마법사를 사용하여 Auto Scaling 그룹 만들기

Amazon EC2 시작 마법사를 사용하여 하나의 절차로 시작 구성 및 Auto Scaling 그룹을 만들 수 있습니다. 이는 두 개 이상의 인스턴스를 시작하며, Amazon EC2 시작 마법사에서 이미 선택한 설정으로부터 새 시작 구성 및 Auto Scaling 그룹을 만들 경우에 유용합니다. 이 옵션을 사용하여 기존 시작 구성을 사용하는 Auto Scaling 그룹은 만들 수 없습니다.

시작 마법사를 사용하여 시작 구성 및 Auto Scaling 그룹 만들기

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 대시보드에서 [Launch Instance]를 선택합니다.
3. AMI를 선택하고 다음 페이지에서 인스턴스 유형을 선택한 다음, [Next: Configure Instance Details]를 선택합니다.
4. Number of instances(인스턴스 수)에 시작할 인스턴스의 수를 입력한 다음, Launch into Auto Scaling Group(Auto Scaling 그룹 시작)을 선택합니다. 페이지의 기타 구성 세부 정보는 추가하지 않아도 됩니다.
5. 확인 페이지에서 [Create Launch Configuration]을 선택합니다.
6. 시작 구성 마법사의 3단계로 전환됩니다. AMI와 인스턴스 유형은 Amazon EC2 시작 마법사에서 선택한 사항에 따라 이미 선택되어 있습니다. 시작 구성의 이름을 입력하고, 필요에 따라 설정을 구성한 다음, [Next: Add Storage]를 선택합니다.
7. 볼륨을 추가로 구성한 다음 [Next: Configure Security Group]을 선택합니다.
8. 새 보안 그룹을 만들거나 기존 그룹을 선택한 다음, [Review]를 선택합니다.
9. 시작 구성의 세부 정보를 검토한 다음, [Create launch configuration]을 선택하여 키 페어를 선택하고 시작 구성을 만듭니다.

10. Auto Scaling 그룹 세부 정보 구성 페이지에 만든 시작 구성이 이미 선택되어 있으며, Amazon EC2 시작 마법사에서 지정한 인스턴스의 수가 그룹 크기에 채워져 있습니다. 그룹 이름을 입력하고, VPC 및 서브넷(필요한 경우)을 지정한 다음, [Next: Configure scaling policies]를 선택합니다.
11. Configure scaling policies(조정 정책 구성) 페이지에서 다음 옵션 중 하나를 선택한 다음, 검토를 선택합니다.
 - Auto Scaling 그룹의 크기를 필요에 따라 수동으로 조정하려면 Keep this group at its initial size(이 그룹을 초기 크기로 유지)를 선택합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 수동 조정 \(p. 90\)](#) 단원을 참조하십시오.
 - 지정하는 기준에 따라 Auto Scaling 그룹의 크기를 자동으로 조정하려면 Use scaling policies to adjust the capacity of this group(조정 정책을 사용하여 이 그룹의 용량 조정)을 선택한 후 지침을 따릅니다. 자세한 내용은 [조정 정책 구성 \(p. 106\)](#)을 참조하십시오.
12. [Review] 페이지에서 필요할 경우 태그 또는 알림을 추가하고, 기타 구성 세부 정보를 편집할 수 있습니다. 모두 마쳤으면 Auto Scaling 그룹 생성을 선택합니다.

Auto Scaling 그룹 및 인스턴스 태그 지정

태그를 사용하면 용도, 소유자, 환경 등에 따라 Auto Scaling 그룹을 다양한 방식으로 분류할 수 있습니다.

각 Auto Scaling 그룹에 여러 태그를 추가할 수 있습니다. 또한 Auto Scaling 그룹의 태그를 시작된 Amazon EC2 인스턴스로 전파할 수 있습니다. 인스턴스에 태그를 지정하면 AWS 청구서에서 태그별로 인스턴스 비용 할당을 확인할 수 있습니다. 자세한 정보는 AWS Billing and Cost Management 사용 설명서의 [비용 할당 태그 사용](#)을 참조하십시오.

또한, 해당 계정에서 태그를 생성, 편집 또는 삭제할 수 있는 권한이 있는 IAM 사용자 및 그룹을 제어할 수 있습니다. 자세한 내용은 [사용할 수 있는 태그 키 및 태그 값 제어 \(p. 181\)](#) 단원을 참조하십시오. 그러나 사용자가 Auto Scaling 그룹에서 태그 지정 작업을 수행하지 못하도록 제한하는 정책을 설정해도, 인스턴스를 시작한 후에 수동으로 인스턴스의 태그를 변경하는 것을 막을 수 없습니다. Amazon EC2 리소스에 태그를 지정(또는 해제)하는 IAM 정책에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [예제: 리소스에 태그 지정](#)을 참조하십시오.

태그는 Amazon EBS 볼륨으로 전파되지 않습니다. Amazon EBS 볼륨에 태그를 추가하려면 시작 템플릿에서 태그를 지정합니다. 하지만 시작 템플릿에서 인스턴스 태그를 구성할 때는 주의해야 합니다. 시작 템플릿이 Auto Scaling 그룹에 대해 지정된 키가 있는 인스턴스 태그를 지정하는 경우 Amazon EC2 Auto Scaling은 해당 인스턴스 태그의 값을 Auto Scaling 그룹에서 지정한 값으로 재정의합니다. 시작 템플릿의 태그 지정에 대한 자세한 내용은 [Auto Scaling 그룹을 위한 시작 템플릿 만들기 \(p. 26\)](#) 단원을 참조하십시오.

목차

- [태그 제한 \(p. 62\)](#)
- [태그 지정 수명 주기 \(p. 63\)](#)
- [Auto Scaling 그룹에 태그 추가 및 태그 수정 \(p. 63\)](#)
- [태그 삭제 \(p. 65\)](#)

태그 제한

태그에 적용되는 기본 제한은 다음과 같습니다.

- 리소스당 최대 태그 수는 50개입니다.
- 단일 호출을 사용하여 추가하거나 제거할 수 있는 최대 태그 수는 25입니다.
- 최대 키 길이는 유니코드 문자 128자입니다.
- 최대 값 길이는 유니코드 문자 256자입니다.

- 태그 키와 값은 대/소문자를 구분합니다.
- 태그 이름이나 값에서 `aws:` 접두사는 사용하지 마십시오. 이 단어는 AWS용으로 예약되어 있습니다. 이 접두사를 가진 태그 이름이나 값은 편집 또는 삭제할 수 없으며 Auto Scaling 그룹당 태그 제한으로도 계수되지 않습니다.

태그 지정 수명 주기

Amazon EC2 인스턴스에 태그를 전파하기로 선택하면 태그가 다음과 같이 관리됩니다.

- Auto Scaling 그룹에서 인스턴스를 시작하면 리소스를 생성한 후가 아니라 리소스를 생성하는 동안 인스턴스에 태그를 추가합니다.
- Auto Scaling 그룹은 `aws:autoscaling:groupName` 키와 Auto Scaling 그룹 이름 값을 사용하여 인스턴스에 태그를 자동으로 추가합니다.
- 기존 인스턴스를 연결하면 Auto Scaling 그룹이 인스턴스에 태그를 추가하여 동일한 태그 키를 가진 기존 태그를 덮어씁니다. 또한 키가 `aws:autoscaling:groupName`이고 값이 Auto Scaling 그룹의 이름인 태그를 추가합니다.
- Auto Scaling 그룹에서 인스턴스를 분리하면 `aws:autoscaling:groupName` 태그만 제거됩니다.
- 사용자가 수동으로 축소하거나 Auto Scaling 그룹이 자동으로 축소하면 종료 중인 인스턴스에서 모든 태그가 제거됩니다.

Auto Scaling 그룹에 태그 추가 및 태그 수정

Auto Scaling 그룹에 태그를 추가할 때 Auto Scaling 그룹에서 시작되는 인스턴스에 추가할지 여부를 지정할 수 있습니다. 태그를 수정하면 변경 후 태그의 업데이트 버전이 Auto Scaling 그룹에서 시작되는 인스턴스에 추가됩니다. Auto Scaling 그룹의 태그를 생성하거나 수정하는 경우 이러한 변경 사항은 Auto Scaling 그룹에서 이미 실행 중인 인스턴스에는 적용되지 않습니다.

목차

- [태그 추가 또는 수정\(콘솔\)](#) (p. 63)
- [태그 추가 또는 수정\(AWS CLI\)](#) (p. 64)

태그 추가 또는 수정(콘솔)

Amazon EC2 콘솔을 사용하여 다음을 수행합니다.

- 태그를 생성할 때 새 Auto Scaling 그룹에 태그 추가
- 기존 Auto Scaling 그룹의 태그 추가, 수정 또는 삭제

생성 시 Auto Scaling 그룹에 태그를 지정하려면

Amazon EC2 콘솔을 사용하여 Auto Scaling 그룹을 생성할 때 Auto Scaling 그룹 생성 마법사의 **Configure Tags**(태그 구성) 페이지에서 태그 키와 값을 지정할 수 있습니다. Auto Scaling 그룹에서 시작된 인스턴스에 태그를 전파하려면 선택된 태그에 대한 **Tag New Instances**(새 인스턴스 태그 지정) 옵션을 유지해야 합니다. 그렇지 않은 경우 선택을 취소할 수 있습니다.

기존 Auto Scaling 그룹의 태그를 추가하거나 수정하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 **[Auto Scaling Groups]**를 선택합니다.
3. 기존 그룹 옆의 확인란을 선택합니다.

Auto Scaling 그룹 창 아래에 새 창이 나타나 선택한 그룹에 대한 정보가 표시됩니다.

- 상세 정보 탭의 태그에서 편집을 선택합니다. (이전 콘솔: 태그 탭에서 태그 추가/편집을 선택합니다.)
- 기존의 태그를 수정하려면 [Key] 및 [Value]를 수정합니다.
- 새 태그를 추가하려면 [Add tag]를 선택하고 [Key] 및 [Value]를 편집합니다. Auto Scaling 그룹에서 시작되는 인스턴스에 태그를 자동으로 추가하려면 새 인스턴스에 태그 지정을 선택한 채로 둡니다. 그렇지 않으면 선택 취소하십시오.
- 태그 추가를 마쳤으면 저장을 선택합니다. (이전 콘솔: 저장을 선택합니다.)

태그 추가 또는 수정(AWS CLI)

다음 예에서는 AWS CLI를 사용하여 Auto Scaling 그룹을 생성할 때 태그를 추가하고 기존 Auto Scaling 그룹에 대한 태그를 추가 또는 수정하는 방법을 보여줍니다.

생성 시 Auto Scaling 그룹에 태그를 지정하려면

- `create-auto-scaling-group` 명령을 사용하여 새 Auto Scaling 그룹을 생성하고 Auto Scaling 그룹에 태그 (예: `env=prod`)를 추가합니다. 태그는 Auto Scaling 그룹에서 시작되는 모든 인스턴스에도 추가됩니다.

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg \
--launch-configuration-name my-launch-config --min-size 1 --max-size 3 \
--vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782" \
--tags Key=env,Value=prod,PropagateAtLaunch=true
```

기존 Auto Scaling 그룹의 태그를 생성하거나 수정하려면

- `create-or-update-tags` 명령을 사용하여 태그를 생성하거나 수정합니다. 예를 들어, 다음 명령은 `Name=my-asg` 및 `cost-center=cc123` 태그를 추가합니다. 태그는 이 변경 후에 Auto Scaling 그룹에서 시작되는 모든 인스턴스에도 추가됩니다. 이 키를 가진 태그가 이미 있으면 기존 태그가 교체됩니다. Amazon EC2 콘솔은 각 인스턴스의 표시 이름을 Name 키에 지정된 이름(대소문자 구분)과 연결합니다.

```
aws autoscaling create-or-update-tags \
--tags ResourceId=my-asg,ResourceType=auto-scaling-group,Key=Name,Value=my-asg,PropagateAtLaunch=true \
ResourceId=my-asg,ResourceType=auto-scaling-group,Key=cost-center,Value=cc123,PropagateAtLaunch=true
```

Auto Scaling 그룹에 대한 태그를 모두 나열하려면

- 다음 `describe-tags` 명령을 사용하여 지정된 Auto Scaling 그룹의 태그 목록을 표시합니다.

```
aws autoscaling describe-tags --filters Name=auto-scaling-group,Values=my-asg
```

다음은 응답의 예입니다.

```
{
  "Tags": [
    {
      "ResourceType": "auto-scaling-group",
      "ResourceId": "my-asg",
      "PropagateAtLaunch": true,
      "Value": "prod",
      "Key": "env"
    }
  ]
}
```



```
]
}
```

- 또는 다음 `describe-auto-scaling-groups` 명령을 사용하여 태그가 Auto Scaling 그룹에 추가되었는지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

다음은 응답의 예입니다.

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 0,
      "SuspendedProcesses": [],
      "DesiredCapacity": 1,
      "Tags": [
        {
          "ResourceType": "auto-scaling-group",
          "ResourceId": "my-asg",
          "PropagateAtLaunch": true,
          "Value": "prod",
          "Key": "env"
        }
      ],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-asg",
      ...
    }
  ]
}
```

태그 삭제

언제든지 Auto Scaling 그룹과 연결된 태그를 삭제할 수 있습니다.

목차

- [태그 삭제\(콘솔\) \(p. 65\)](#)
- [태그 삭제\(AWS CLI\) \(p. 66\)](#)

태그 삭제(콘솔)

태그를 삭제하려면(새 콘솔)

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
- 기존 그룹 옆의 확인란을 선택합니다.

Auto Scaling 그룹 창 아래에 새 창이 나타나 선택한 그룹에 대한 정보가 표시됩니다.

- 상세 정보 탭의 태그에서 편집을 선택합니다.
- 태그 옆에 있는 제거를 선택합니다.
- 업데이트를 선택합니다.

태그를 삭제하려면(이전 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling에서 Auto Scaling 그룹을 선택합니다.
3. 목록에서 기존 사용자를 선택합니다.
4. 태그 탭에서 태그 추가/편집을 선택합니다. Auto Scaling 그룹 태그 추가/편집 페이지에 Auto Scaling 그룹에 있는 기존의 모든 태그가 나열됩니다.
5. 태그 옆에 있는 삭제 아이콘을 선택합니다.
6. Save를 선택합니다.

태그 삭제(AWS CLI)

`delete-tags` 명령을 사용하여 태그를 삭제합니다. 예를 들어, 다음 명령은 키가 `env`인 태그를 삭제합니다.

```
aws autoscaling delete-tags --tags "ResourceId=my-asg,ResourceType=auto-scaling-group,Key=env"
```

태그 키는 지정해야 하지만 값은 지정하지 않아도 됩니다. 값을 지정했는데 해당 값이 잘못된 경우 태그가 삭제되지 않습니다.

Auto Scaling 그룹에 로드 밸런서 사용

로드 밸런서는 Auto Scaling 그룹으로 들어오는 모든 웹 트래픽에 대하여 단일 접점의 역할을 합니다. 인스턴스가 그룹에 추가되면 로드 밸런서에 등록해야 하며 그렇지 않은 경우 트래픽이 라우팅되지 않습니다. 그룹에서 인스턴스가 제거되면 로드 밸런서에서 등록을 취소해야 하며 그렇지 않으면 트래픽이 계속해서 라우팅됩니다.

Auto Scaling 그룹에 Elastic Load Balancing 로드 밸런서를 사용하려는 경우, 로드 밸런서 또는 대상 그룹에 EC2 인스턴스를 등록할 필요가 없습니다. Elastic Load Balancing을 사용 설정하면 Auto Scaling 그룹에서 시작한 인스턴스가 자동으로 로드 밸런서 또는 대상 그룹에 등록되고, Auto Scaling 그룹에 의해 종료된 인스턴스는 로드 밸런서 또는 대상 그룹에서 자동으로 등록이 취소됩니다.

또한 등록된 인스턴스의 상태를 모니터링하는 Elastic Load Balancing 상태 확인을 구성하여 로드 밸런서 또는 대상 그룹이 정상적인 인스턴스에만 트래픽을 라우팅하도록 할 수 있습니다.

Elastic Load Balancing 유형

Elastic Load Balancing은 Classic Load Balancer, Application Load Balancer 및 Network Load Balancer라는 세 가지 유형의 로드 밸런서를 제공하며, 세 유형 모두 Auto Scaling 그룹에 사용할 수 있습니다. Classic Load Balancer에서는 로드 밸런서에 인스턴스를 등록합니다. Application Load Balancer 및 Network Load Balancer에서는 대상 그룹에 대상으로 인스턴스를 등록합니다.

Classic Load Balancer

전송 계층(TCP/SSL) 또는 애플리케이션 계층(HTTP/HTTPS)에서 라우팅 및 로드 밸런싱합니다. Classic Load Balancer는 EC2-Classic 또는 VPC를 지원합니다.

Application Load Balancer

애플리케이션 계층(HTTP/HTTPS)에서 라우팅 및 로드 밸런싱하며 경로 기반 라우팅을 지원합니다. Application Load Balancer는 가상 프라이빗 클라우드(VPC)의 EC2 인스턴스와 같은 하나 이상의 등록된 대상의 포트에 요청을 라우팅 할 수 있습니다.

Note

Application Load Balancer 대상 그룹에는 instance의 대상 유형이 있어야 합니다. 자세한 내용은 Application Load Balancer 사용 설명서의 [대상 유형](#)을 참조하십시오.

Network Load Balancer

패킷 콘텐츠가 아닌 TCP 패킷 헤더에서 추출된 주소 정보를 기반으로 한 전송 계층(TCP/UDP Layer-4)에서 라우팅 및 로드 밸런싱합니다. Network Load Balancer는 트래픽 버스트를 처리하고, 클라이언트의 소스 IP를 보유하고, 로드 밸런서의 수명 동안 고정 IP를 사용할 수 있습니다.

Note

Network Load Balancer 대상 그룹에는 instance의 대상 유형이 있어야 합니다. 자세한 내용은 Network Load Balancer 사용 설명서의 [대상 유형](#)을 참조하십시오.

Elastic Load Balancing에 대한 자세한 내용은 다음 주제를 참조하십시오.

- [Elastic Load Balancing란 무엇입니까?](#)
- [Classic Load Balancer란 무엇입니까?](#)
- [Application Load Balancer란 무엇입니까?](#)
- [Network Load Balancer란 무엇입니까?](#)

Elastic Load Balancing와 Amazon EC2 Auto Scaling의 통합에 대한 자세한 내용은 다음 주제를 참조하십시오.

주제

- [Auto Scaling 그룹에 로드 밸런서 연결 \(p. 67\)](#)
- [Auto Scaling 그룹에 Elastic Load Balancing 상태 확인 추가 \(p. 70\)](#)
- [규모가 늘어나고 로드 밸런싱된 애플리케이션을 추가 가용 영역까지 확장 \(p. 71\)](#)

Auto Scaling 그룹에 로드 밸런서 연결

이 주제에서는 Elastic Load Balancing 로드 밸런서를 기존 Auto Scaling 그룹에 연결하는 방법을 설명합니다. Auto Scaling 그룹을 만들 때 그룹에 로드 밸런서를 연결하려면 [자습서: 조정 및 로드 밸런싱된 애플리케이션 설정 \(p. 19\)](#)을 참조하십시오.

Amazon EC2 Auto Scaling은 기존 Auto Scaling 그룹에 하나 이상의 로드 밸런서를 연결할 수 있도록 Elastic Load Balancing과 통합합니다. 로드 밸런서를 연결한 후에는 그룹의 인스턴스를 자동으로 등록하고 인스턴스 간에 수신 트래픽을 분산합니다.

로드 밸런서를 연결하면 그룹의 인스턴스를 등록하는 동안 인스턴스가 Adding 상태로 전환됩니다. 그룹의 모든 인스턴스가 로드 밸런서에 등록된 후에는 인스턴스가 Added 상태로 전환됩니다. 등록된 인스턴스가 하나 이상 상태 확인을 통과한 후에는 인스턴스가 InService 상태로 전환됩니다. 로드 밸런서가 InService 상태로 전환되면 Amazon EC2 Auto Scaling이 비정상적으로 보고된 모든 인스턴스를 종료하고 교체할 수 있습니다. 등록된 인스턴스 중 상태 확인을 통과한 인스턴스가 없는 경우(예: 잘못 구성된 상태 확인으로 인해), 로드 밸런서가 InService 상태로 전환되지 않으므로 Amazon EC2 Auto Scaling은 인스턴스를 종료하고 교체할 수 없습니다.

로드 밸런서를 분리하면 그룹의 인스턴스를 등록 해제하는 동안 인스턴스가 Removing 상태로 전환됩니다. 인스턴스는 등록 해제된 후에도 계속 실행됩니다. 연결 드레이닝이 활성화된 경우, Elastic Load Balancing은 인스턴스를 등록 해제하기 전에 진행 중인 요청이 완료되거나 최대 제한 시간이 만료될 때까지 (먼저 일어나는 쪽을) 기다립니다. 기본적으로, Connection draining은 Application Load Balancer에 대해 항상 활성화되지만 Classic Load Balancer에 꼭 활성화되어야 합니다. 자세한 내용은 Classic Load Balancer 사용 설명서의 [연결 드레이닝](#)을 참조하십시오.

목차

- 사전 요구 사항 (p. 68)
- 로드 밸런서 연결(콘솔) (p. 68)
- 로드 밸런서 연결(AWS CLI) (p. 69)

사전 요구 사항

시작하기 전에 Auto Scaling 그룹과 같은 AWS 리전에 Application Load Balancer 또는 Network Load Balancer를 생성합니다. 새 로드 밸런서를 사용하는 것이 좋지만 원하는 기능을 지원하는 경우 Classic Load Balancer를 계속 사용할 수 있습니다. 다양한 유형의 로드 밸런서에 대한 자세한 내용은 [Elastic Load Balancing 유형 \(p. 66\)](#) 단원을 참조하십시오.

(선택 사항) Elastic Load Balancing 상태 확인을 사용하도록 Auto Scaling 그룹을 구성하려면 [Auto Scaling 그룹에 Elastic Load Balancing 상태 확인 추가 \(p. 70\)](#) 단원을 참조하십시오.

Note

Amazon EC2 Auto Scaling에서 Auto Scaling 그룹 인터페이스가 변경되었습니다. 기본적으로 기존 사용자 인터페이스가 표시되지만 새 사용자 인터페이스로 전환할 수 있습니다. 이 주제에는 둘 모두에 대한 단계가 포함되어 있습니다.

로드 밸런서 연결(콘솔)

다음 절차에 따라 기존 Auto Scaling 그룹에 로드 밸런서를 연결합니다.

그룹에 로드 밸런서를 연결하려면(새 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
3. 기존 그룹 옆의 확인란을 선택합니다.

Auto Scaling 그룹 창 아래에 새 창이 나타나 선택한 그룹에 대한 정보가 표시됩니다.
4. 세부 정보 탭의 로드 밸런싱에서 편집을 선택합니다.
5. 로드 밸런싱에서 다음 중 하나를 수행합니다.
 - a. [Application/Network Load Balancer] Choose a target group for your load balancer(로드 밸런서에 대해 대상 그룹 선택)에서 대상 그룹을 선택합니다.
 - b. [Classic Load Balancer] Choose a load balancer(로드 밸런서 선택)에서 로드 밸런서를 선택합니다.
6. 업데이트를 선택합니다.

그룹에 로드 밸런서를 연결하려면(이전 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. 목록에서 기존 사용자를 선택합니다.
4. 세부 정보 탭에서 편집을 선택합니다.
5. 다음 중 하나를 수행하십시오.
 - a. [Classic Load Balancer] 클래식 로드 밸런서에서 로드 밸런서를 선택합니다.
 - b. [애플리케이션/네트워크 로드 밸런서] [Target Groups]에서 대상 그룹을 선택합니다.
6. Save를 선택합니다.

더 이상 로드 밸런서가 필요하지 않으면 다음 절차에 따라 Auto Scaling 그룹에서 분리합니다.

그룹에서 로드 밸런서를 분리하려면(새 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
3. 기존 그룹 옆의 확인란을 선택합니다.

Auto Scaling 그룹 창 아래에 새 창이 나타나 선택한 그룹에 대한 정보가 표시됩니다.
4. 세부 정보 탭의 로드 밸런싱에서 편집을 선택합니다.
5. 로드 밸런싱에서 다음 중 하나를 수행합니다.
 - a. [Application/Network Load Balancer] Choose a target group for your load balancer(로드 밸런서에 대해 대상 그룹 선택)에서 대상 그룹 옆에 있는 삭제 아이콘(X)을 선택합니다.
 - b. [Classic Load Balancer] Choose a load balancer(로드 밸런서 선택)에서 로드 밸런서 옆에 있는 삭제 아이콘(X)을 선택합니다.
6. 업데이트를 선택합니다.

그룹에서 로드 밸런서를 분리하려면(이전 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. 목록에서 기존 사용자를 선택합니다.
4. 세부 정보 탭에서 편집을 선택합니다.
5. 다음 중 하나를 수행하십시오.
 - a. [Classic Load Balancer] 클래식 로드 밸런서에서 로드 밸런서를 제거합니다.
 - b. [애플리케이션/네트워크 로드 밸런서] [Target Groups]에서 대상 그룹을 제거합니다.
6. Save를 선택합니다.

로드 밸런서 연결(AWS CLI)

Application Load Balancer 또는 Network Load Balancer에 대한 대상 그룹을 연결하려면

다음 [attach-load-balancer-target-groups](#) 명령을 사용하여 지정된 대상 그룹을 Auto Scaling 그룹에 연결합니다.

```
aws autoscaling attach-load-balancer-target-groups --auto-scaling-group-name my-asg \  
--target-group-arns my-targetgroup-arn
```

Application Load Balancer 또는 Network Load Balancer에 대한 대상 그룹을 연결 해제하려면

더 이상 필요하지 않으면 다음 [detach-load-balancer-target-groups](#) 명령을 사용하여 Auto Scaling 그룹에서 대상 그룹을 분리합니다.

```
aws autoscaling detach-load-balancer-target-groups --auto-scaling-group-name my-asg \  
--target-group-arns my-targetgroup-arn
```

Classic Load Balancer를 연결하려면

다음 [attach-load-balancers](#) 명령을 사용하여 지정된 로드 밸런서를 Auto Scaling 그룹에 연결합니다.

```
aws autoscaling attach-load-balancers --auto-scaling-group-name my-asg \  
--load-balancer-names my-lb
```

Classic Load Balancer를 분리하려면

더 이상 필요하지 않으면 다음 `detach-load-balancers` 명령을 사용하여 Auto Scaling 그룹에서 로드 밸런서를 분리합니다.

```
aws autoscaling detach-load-balancers --auto-scaling-group-name my-asg \
--load-balancer-names my-lb
```

Auto Scaling 그룹에 Elastic Load Balancing 상태 확인 추가

Auto Scaling 그룹의 기본 상태 검사는 EC2 상태 확인만 해당합니다. 한 인스턴스에서 이러한 상태 검사에 실패할 경우 Auto Scaling 그룹은 해당 인스턴스를 비정상 상태로 간주하여 교체합니다. 자세한 내용은 [Auto Scaling 인스턴스 상태 확인 \(p. 148\)](#) 단원을 참조하십시오.

하나 이상의 대상 그룹(Application Load Balancer 및 Network Load Balancer), 하나 이상의 로드 밸런서(Classic Load Balancer) 또는 둘 다를 Auto Scaling 그룹에 연결할 수 있습니다. 그러나 기본적으로 그룹은 인스턴스를 비정상 상태로 간주하지 않고 Elastic Load Balancing 상태 확인에 실패할 경우 인스턴스를 교체합니다.

그룹에서 로드 밸런서가 제공하는 추가 테스트에 따라 인스턴스의 상태를 확인할 수 있도록 하려면 선택적으로 Elastic Load Balancing 상태 확인을 사용하도록 Auto Scaling 그룹을 구성할 수 있습니다. 로드 밸런서가 정기적으로 ping을 전송하거나, 연결을 시도하거나, 요청을 전송하여 EC2 인스턴스를 테스트합니다. 이러한 테스트를 상태 확인이라고 합니다.

Elastic Load Balancing 상태 검사에 대한 자세한 내용은 다음 주제를 참조하십시오.

- Classic Load Balancer 사용 설명서에서 [Classic Load Balancer에 대한 상태 확인 구성](#)
- Application Load Balancer 사용 설명서의 [대상 그룹에 대한 상태 확인](#)
- Network Load Balancer 사용 설명서의 [대상 그룹에 대한 상태 확인](#)

Elastic Load Balancing 상태 확인을 사용하도록 Auto Scaling 그룹을 구성하면 EC2 상태 확인 또는 Elastic Load Balancing 상태 확인에 실패할 경우 인스턴스를 비정상 상태로 간주합니다. 여러 로드 밸런서 대상 그룹 또는 Classic Load Balancer를 그룹에 연결할 경우 인스턴스를 정상으로 간주하려면 모두 해당 인스턴스가 정상이라고 보고해야 합니다. 그 중 하나가 인스턴스를 비정상 상태로 보고하면 다른 곳에서 정상으로 보고하더라도 Auto Scaling 그룹은 인스턴스를 교체합니다.

다음 절차는 Auto Scaling 그룹에 Elastic Load Balancing 상태 확인을 추가하는 방법을 설명합니다.

목차

- [상태 검사 추가하기\(콘솔\) \(p. 70\)](#)
- [상태 검사 추가하기\(AWS CLI\) \(p. 71\)](#)

상태 검사 추가하기(콘솔)

다음 절차를 사용하여 로드 밸런서가 연결된 Auto Scaling 그룹에 유예 기간이 300초인 Elastic Load Balancing(ELB) 상태 확인을 추가합니다.

상태 확인을 추가하려면 다음을 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
3. 기존 그룹 옆의 확인란을 선택합니다.

Auto Scaling 그룹 창 아래에 새 창이 나타나 선택한 그룹에 대한 정보가 표시됩니다.

4. 세부 정보 탭의 상태 확인에서 편집을 선택합니다. (이전 콘솔: 세부 정보 탭에서 편집을 선택합니다.)
5. 상태 확인 유형에서 ELB 상태 확인 활성화를 선택합니다. (이전 콘솔: ELB를 선택합니다.)
6. 상태 확인 유예 기간에 300을 입력합니다.
7. 업데이트를 선택합니다. (이전 콘솔: 저장을 선택합니다.)
8. 인스턴스 관리 탭의 인스턴스에서 인스턴스의 상태를 볼 수 있습니다. (이전 콘솔: 인스턴스 탭에서는 인스턴스의 상태를 볼 수 있습니다.) 상태 열에 새로 추가된 상태 확인 결과가 표시됩니다.

상태 검사 추가하기(AWS CLI)

다음 `update-auto-scaling-group` 명령을 사용하여 유예 기간이 300초인 상태 확인을 생성합니다.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-lb-asg \
--health-check-type ELB --health-check-grace-period 300
```

규모가 늘어나고 로드 밸런싱된 애플리케이션을 추가 가용 영역까지 확장

리전 내 여러 가용 영역에 걸쳐 Auto Scaling 그룹으로 확장한 다음 로드 밸런서를 연결하여 들어오는 트래픽을 해당 영역에 분산시킴으로써 안전하고 안정적인 지리적 이중화를 활용할 수 있습니다. 들어오는 트래픽은 로드 밸런서에서 활성화된 모든 가용 영역에 고르게 분산됩니다.

Note

Auto Scaling 그룹은 동일한 리전 내의 여러 가용 영역에 속하는 Amazon EC2 인스턴스를 포함할 수 있습니다. 그러나 Auto Scaling 그룹은 여러 리전의 인스턴스를 포함할 수는 없습니다.

하나의 가용 영역이 비정상 또는 사용 불가 상태가 되었을 때 영향을 받지 않은 영역에서 Amazon EC2 Auto Scaling이 새 인스턴스를 시작합니다. 비정상 가용 영역이 정상 상태로 복귀하는 경우 Auto Scaling 그룹의 모든 가용 영역에서 Amazon EC2 Auto Scaling이 해당 애플리케이션 인스턴스를 고르게 자동으로 재배포합니다. 이는 최소의 인스턴스로 가용 영역에서 새 인스턴스를 시작하려고 하는 방식으로 Amazon EC2 Auto Scaling에 의해 수행됩니다. 하지만 시도가 실패하는 경우 성공할 때까지 Amazon EC2 Auto Scaling은 다른 가용 영역에서의 시작을 계속 시도합니다.

Auto Scaling 그룹에 가용 영역을 추가한 다음 로드 밸런서에서 해당 영역을 활성화하는 방식으로, 규모가 늘어나고 로드 밸런싱된 애플리케이션의 가용성을 늘릴 수 있습니다. 새 가용 구역을 활성화한 후 로드 밸런서는 활성화된 모든 영역에 걸쳐 고르게 트래픽을 라우팅하기 시작합니다.

목차

- 가용 영역 추가(콘솔) (p. 71)
- 가용 영역 추가(AWS CLI) (p. 72)

가용 영역 추가(콘솔)

Auto Scaling 그룹을 추가 서브넷으로 확장하려면 다음 절차를 수행합니다.

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
3. 기존 그룹 옆의 확인란을 선택합니다.

Auto Scaling 그룹 창 아래에 새 창이 나타나 선택한 그룹에 대한 정보가 표시됩니다.

4. 상세 정보 탭의 네트워크에서 편집을 선택합니다. (이전 콘솔: 세부 정보 탭에서 편집을 선택합니다.)

5. 서브넷에서 가용 영역에 해당하는 서브넷을 선택합니다.
6. 업데이트를 선택합니다. (이전 콘솔: 저장을 선택합니다.)
7. 로드 밸런서의 가용 영역을 업데이트하여 Auto Scaling 그룹과 동일한 영역을 공유하게 하려면 다음 단계를 완료합니다.
 - a. 탐색 창에서 로드 밸런서를 클릭합니다.
 - b. 로드 밸런서를 선택합니다.
 - c. 다음 중 하나를 수행하십시오.
 - [EC2-Classic에서 Classic Load Balancer] 인스턴스 탭에서 가용 영역 편집을 선택합니다. 가용 영역 추가 및 제거 페이지에서 추가할 가용 영역을 선택합니다.
 - [VPC에서 Classic Load Balancer] 인스턴스 탭에서 가용 영역 편집을 선택합니다. [Add and Remove Subnets] 페이지의 [Available subnets]에서 추가할 서브넷에 추가 아이콘(+)을 선택합니다. 그러면 서브넷이 [Selected subnets] 아래로 이동합니다.
 - [Application Load Balancer] 설명 탭의 가용 영역에서 편집을 선택합니다. 가용 영역에 추가할 서브넷 중 하나에 추가 아이콘(+)을 선택합니다. 그러면 서브넷이 [Selected subnets] 아래로 이동합니다.
 - d. Save를 선택합니다.

가용 영역 추가(AWS CLI)

사용하는 명령은 로드 밸런서가 VPC의 Classic Load Balancer인지, EC2-Classic의 Classic Load Balancer인지 또는 Application Load Balancer인지에 따라 다릅니다.

VPC의 Classic Load Balancer에 연결된 Auto Scaling 그룹의 경우

1. 다음 `update-auto-scaling-group` 명령을 사용하여 Auto Scaling 그룹을 서브넷에 추가합니다.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg \
  --vpc-zone-identifier subnet-41767929 subnet-cb663da2 --min-size 2
```

2. 다음 `describe-auto-scaling-groups` 명령을 사용하여 새 서브넷의 인스턴스가 로드 밸런서로부터 트래픽을 허용할 준비가 되었는지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

3. 다음 `attach-load-balancer-to-subnets` 명령을 사용하여 Classic Load Balancer에 대해 새 서브넷을 활성화합니다.

```
aws elb attach-load-balancer-to-subnets --load-balancer-name my-lb \
  --subnets subnet-41767929
```

EC2-Classic의 Classic Load Balancer에 연결된 Auto Scaling 그룹의 경우

1. 다음 `update-auto-scaling-group` 명령을 사용하여 Auto Scaling 그룹에 가용 영역을 추가합니다.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg \
  --availability-zones us-west-2a us-west-2b us-west-2c --min-size 3
```

2. 다음 `describe-auto-scaling-groups` 명령을 사용하여 새 가용 영역의 인스턴스가 로드 밸런서로부터의 트래픽을 허용할 준비가 되었는지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```


3. 다음 [enable-availability-zones-for-load-balancer](#) 명령을 사용하여 Classic Load Balancer에 대해 새 가용 영역을 활성화합니다.

```
aws elb enable-availability-zones-for-load-balancer --load-balancer-name my-lb \
--availability-zones us-west-2c
```

Application Load Balancer가 있는 Auto Scaling 그룹의 경우

1. 다음 [update-auto-scaling-group](#) 명령을 사용하여 Auto Scaling 그룹을 서브넷에 추가합니다.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg \
--vpc-zone-identifier subnet-41767929 subnet-cb663da2 --min-size 2
```

2. 다음 [describe-auto-scaling-groups](#) 명령을 사용하여 새 서브넷의 인스턴스가 로드 밸런서로부터 트래픽을 허용할 준비가 되었는지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

3. 다음 [set-subnets](#) 명령을 사용하여 Application Load Balancer에 대해 새 서브넷을 활성화합니다.

```
aws elbv2 set-subnets --load-balancer-arn my-lb-arn \
--subnets subnet-41767929 subnet-cb663da2
```

Auto Scaling 그룹에서 스팟 인스턴스 시작

스팟 인스턴스는 애플리케이션이 실행되는 시간을 유연하게 조정할 수 있고 애플리케이션을 중단할 수 있는 경우 선택할 수 있는 온디맨드 인스턴스에 비해 비용 효과적인 방법입니다. 이 주제에서는 Auto Scaling 그룹 자체가 아닌 시작 구성 또는 시작 템플릿에서 설정을 지정하여 Auto Scaling 그룹에서 스팟 인스턴스만 시작하는 방법을 설명합니다.

Important

Auto Scaling 그룹 설정의 일부로 스팟 인스턴스를 시작하는 데 사용되는 것과 동일한 설정을 지정할 수 있습니다. Auto Scaling 그룹의 일부로 설정을 지정할 때 추가 옵션을 지정할 수 있습니다. 예를 들어, 스팟 인스턴스만 시작할지 아니면 온디맨드 인스턴스와 스팟 인스턴스의 결합을 시작할지 여부를 지정할 수 있습니다. 자세한 내용은 [여러 인스턴스 유형 및 구매 옵션이 포함된 Auto Scaling 그룹](#) (p. 45) 단원을 참조하십시오.

Amazon EC2 Auto Scaling를 사용하여 스팟 인스턴스를 시작하기 전에, 먼저 Amazon EC2를 사용하여 스팟 인스턴스를 시작하고 관리하는 데 익숙해지는 것이 좋습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [스팟 인스턴스](#)를 참조하십시오.

온디맨드 인스턴스 대신 스팟 인스턴스를 시작하는 시작 구성이나 시작 템플릿을 생성하려면 다음 사항을 고려하십시오.

- **최고 가격 설정.** 시작 구성 또는 시작 템플릿의 일부로 최대 결제 요금을 설정할 수 있습니다. 스팟 요금이 최고 가격 이내인 경우 스팟 인스턴스 용량에 따라 요청이 실행됩니다. 시작하는 스팟 인스턴스에 대해 스팟 가격만 지불합니다. 스팟 인스턴스의 가격이 Auto Scaling 그룹에서 실행 중인 인스턴스의 최고 가격보다 오르는 경우, Amazon EC2는 인스턴스를 종료합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Pricing and Savings\(요금 및 절감\)](#)을 참조하십시오.
- **최고 가격 변경.** 새 요금이 포함된 시작 구성 또는 시작 템플릿 버전을 생성해야 합니다. 새 시작 구성을 통해 이를 Auto Scaling 그룹과 연결해야 합니다. 시작 템플릿을 사용하여 Auto Scaling 그룹에서 기본 템플릿을 사용하지 또는 최신 버전의 템플릿을 사용할지를 구성할 수 있습니다. 이렇게 하면 Auto Scaling 그룹과 자동으로 연결됩니다. 그러한 인스턴스에 사용되는 시작 구성 또는 시작 템플릿에 지정된 최고 가격이 현재 스팟 가격보다 높은 한, 기존 인스턴스는 계속해서 실행됩니다.

- 스팟 인스턴스 유지. 스팟 인스턴스가 종료되면 Auto Scaling 그룹에서는 대체 인스턴스를 시작하여 그룹의 원하는 용량을 유지하려고 합니다. 최고 가격이 스팟 가격보다 높으면 스팟 인스턴스가 시작됩니다. 그렇지 않을 경우 또는 요청이 성공하지 않을 경우 시도를 계속합니다.
- 여러 가용 영역을 포괄하며 밸런싱합니다. 여러 가용 영역을 지정하면 Amazon EC2 Auto Scaling이 지정된 영역을 포괄하여 스팟 요청을 분배합니다. 한 가용 영역에서 최고 가격이 너무 낮아 요청을 실행할 수 없으면, Amazon EC2 Auto Scaling이 다른 영역에서 요청이 실행되었는지 확인합니다. 그런 경우 이 실패한 요청을 취소하고 요청이 실행된 가용 영역을 포괄하여 재분배합니다. 실행된 요청이 없는 가용 영역에서 미래 요청에 성공할 만큼 가격이 떨어지면, Amazon EC2 Auto Scaling이 모든 가용 영역을 포괄하여 다시 밸런싱합니다. 자세한 내용은 [재분배 활동 \(p. 6\)](#) 단원을 참조하십시오.
- 스팟 인스턴스 종료. Amazon EC2 Auto Scaling에서는 온디맨드 인스턴스를 종료하거나 대체할 수 있는 것처럼 스팟 인스턴스를 종료하거나 대체할 수 있습니다. 자세한 내용은 [축소 시 Auto Scaling 인스턴스 종료 제어 \(p. 126\)](#) 단원을 참조하십시오.
- 스팟 인스턴스 중단 공지. 스팟 인스턴스 중단 공지를 사용하여 스팟 인스턴스의 상태를 모니터링할 수 있습니다. 예를 들어 Amazon CloudWatch Events에서 Amazon SNS 주제, AWS Lambda 함수 또는 기타 대상에 EC2 스팟 2분 경고를 자동으로 보내는 규칙을 설정할 수 있습니다. 자세한 내용은 [Amazon CloudWatch Events 사용 설명서](#) 및 Linux 인스턴스용 Amazon EC2 사용 설명서의 [스팟 인스턴스 중단 공지](#)를 참조하십시오.

Amazon EC2 Auto Scaling 인스턴스 가중치 부여

여러 인스턴스 유형을 시작하도록 Auto Scaling 그룹을 구성하는 경우 인스턴스 가중치를 사용하여 각 인스턴스가 그룹의 용량에 기여하는 용량 유닛 수를 정의할 수 있습니다. 이를 통해 애플리케이션의 성능에 직접 매핑되는 방식으로 각 인스턴트 유형의 상대적 가중치를 지정할 수 있습니다. 코어(vCPU) 또는 메모리(GiB) 등과 같은 특정 애플리케이션 요구 사항에 따라 인스턴스에 가중치를 부여할 수 있습니다.

예를 들어 최소 8개의 vCPU 및 15GiB의 RAM에서 최적의 성능을 발휘하는 컴퓨팅 집약적 애플리케이션을 실행하는 경우, c5.2xlarge를 기본 유닛으로 사용할 때 다음 EC2 인스턴스 유형은 모두 애플리케이션 요구 사항을 충족합니다.

인스턴스 유형의 예

인스턴스 유형	vCPU	메모리(GiB)
c5.2xlarge	8	16
c5.4xlarge	16	32
c5.12xlarge	48	96
c5.18xlarge	72	144
c5.24xlarge	96	192

기본적으로 모든 인스턴스 유형은 동일한 가중치로 처리됩니다. Amazon EC2 Auto Scaling이 인스턴스를 시작하면 해당 인스턴스 유형이 크든 작든 각 인스턴스가 그룹의 원하는 용량에 가산됩니다.

그런데 인스턴스 가중치를 사용하는 경우 각 인스턴스 유형에 연결할 용량 유닛 수를 지정하는 숫자 값을 할당하게 됩니다. 예를 들어 인스턴스 간의 크기가 서로 다른 경우 c5.2xlarge 인스턴스의 가중치는 2, c5.4xlarge(2배 더 큼)의 가중치는 4와 같은 식으로 지정할 수 있습니다. 그런 다음 Amazon EC2 Auto Scaling에서 인스턴스를 시작하면 가중치가 원하는 용량에 가산됩니다.

단위 시간당 가격

다음 표는 미국 동부(버지니아, 오키오)의 여러 가용 영역에 있는 스팟 인스턴스의 시간당 가격과 동일한 리전의 온디맨드 인스턴스 가격을 비교한 것입니다. 표시된 가격은 현재 가격이 아니라 예를 든 것이며, 인스턴스 시간당 가격입니다.

예: 인스턴스 시간당 스팟 가격

인스턴스 유형	us-east-1a	us-east-1b	us-east-1c	온디맨드 가격
c5.2xlarge	0.180 USD	0.191 USD	0.170 USD	0.34 USD
c5.4xlarge	0.341 USD	0.361 USD	0.318 USD	0.68 USD
c5.12xlarge	0.779 USD	0.777 USD	0.777 USD	2.04 USD
c5.18xlarge	1.207 USD	1.475 USD	1.357 USD	3.06 USD
c5.24xlarge	1.555 USD	1.555 USD	1.555 USD	4.08 USD

인스턴스 가중치를 사용하면 단위 시간당 사용량을 기준으로 비용을 평가할 수 있습니다. 단위 시간당 가격은 인스턴스 유형에 따른 가격을 인스턴스가 나타내는 유닛 수로 나누어 계산합니다. 온디맨드 인스턴스의 경우, 하나의 인스턴스 유형을 배포하는 경우 동일한 인스턴스 유형의 다른 크기를 배포하더라도 단위 시간당 가격이 동일합니다. 그러나 단위 시간당 스팟 가격은 스팟 풀에 따라 다릅니다.

가중치가 부여된 인스턴스에서 단위 시간당 가격이 어떻게 계산되는지 이해하는 가장 쉬운 방법은 예를 살펴보는 것입니다. 예를 들어 간단한 계산을 위해 us-east-1a에서만 스팟 인스턴스를 시작한다고 할 경우 단위 시간당 가격은 아래와 같습니다.

예: 단위 시간당 스팟 가격의 예

인스턴스 유형	us-east-1a	인스턴스 가중치	단위 시간당 가격
c5.2xlarge	0.180 USD	2	0.090 USD
c5.4xlarge	0.341 USD	4	0.085 USD
c5.12xlarge	0.779 USD	12	0.065 USD
c5.18xlarge	1.207 USD	18	0.067 USD
c5.24xlarge	1.555 USD	24	0.065 USD

고려 사항

인스턴스 가중치를 효과적으로 구현하려면 다음 주요 사항을 고려해야 합니다.

- 먼저 애플리케이션의 실제 성능 요구 사항을 반영하는 몇 가지 인스턴스 유형을 선택합니다. 그런 다음 가중치를 지정하여 각 인스턴스 유형을 Auto Scaling 그룹의 원하는 용량에 얼마나 가산할지 결정합니다. 가중치는 그룹의 현재 인스턴스와 향후 인스턴스에 적용됩니다.
- 가중치에 매우 큰 범위를 선택하지 않도록 주의하십시오. 예를 들어 한 인스턴스 유형에 가중치 1을 지정하고 다음으로 큰 인스턴스 유형에 가중치 200을 지정하는 것은 좋지 않습니다. 가장 작은 가중치와 가장 큰 가중치의 차이가 너무 커서도 안 됩니다. 가중치 차이가 너무 큰 인스턴스 유형이 하나라도 있으면 지속적인 비용 성능 최적화에 부정적인 영향을 줄 수 있습니다.
- Auto Scaling 그룹의 크기는 인스턴스가 아닌 용량 유닛으로 측정됩니다. 예를 들어 가중치가 vCPU를 기반으로 하는 경우 원하는 코어 수, 최소 및 최대 코어 수를 지정해야 합니다.
- 원하는 용량이 가장 큰 가중치보다 최소 2~3배 더 크도록 가중치와 원하는 용량을 설정합니다.
- 스팟 최고가를 설정하려는 경우 가장 비용이 많이 드는 인스턴스 유형에 충분히 높은 인스턴스 시간당 가격을 지정해야 합니다. Amazon EC2 Auto Scaling은 가용 영역의 현재 스팟 가격이 최고가보다 낮고 사용 가능한 용량이 있는 경우 스팟 인스턴스를 프로비저닝합니다. 스팟 인스턴스에 대한 요청이 하나의 스팟 인스턴스 풀에서 이행될 수 없는 경우 스팟 인스턴스의 비용 절감 효과를 활용하기 위해 다른 스팟 풀에서 계속 시도합니다.

인스턴스 가중치 사용으로 다음과 같은 새로운 동작이 발생합니다.

- 현재 용량이 원하는 용량 이상이 될 수 있습니다. Amazon EC2 Auto Scaling은 원하는 용량이 완전히 충족될 때까지 인스턴스를 프로비저닝하려고 하므로 초과 요금이 발생할 수 있습니다. 예를 들어 두 인스턴스 유형 `c5.2xlarge`와 `c5.12xlarge`를 지정하고 `c5.2xlarge`에 인스턴스 가중치 2, `c5.12xlarge`에 인스턴스 가중치 12를 할당한 경우, 원하는 용량을 충족하기까지 5개 유닛이 남은 상태에서 Amazon EC2 Auto Scaling이 `c5.12xlarge`을 프로비저닝하면 원하는 용량이 7개 유닛만큼 초과됩니다.
- Amazon EC2 Auto Scaling이 원하는 용량에 도달하도록 인스턴스를 프로비저닝할 때는 가용 영역 전반에 인스턴스를 배포하고 온디맨드 및 스팟 인스턴스에 대한 할당 전략을 적용하는 것이 초과 요금을 방지하는 것보다 우선합니다.
- Amazon EC2 Auto Scaling은 사용자가 기본 설정한 할당 전략을 사용하여 가용 영역 간의 균형을 유지하기 위해 최대 용량 제한을 초과할 수 있습니다. Amazon EC2 Auto Scaling에서 부과하는 하드 제한은 원하는 용량과 가장 큰 가중치를 합한 값과 같습니다.

기존 그룹의 가중치를 추가하거나 수정할 때는 다음 사항에 유의하십시오.

- 기존 Auto Scaling 그룹에 인스턴스 가중치를 추가할 때는 그룹에서 이미 실행 중인 모든 인스턴스 유형을 포함해야 합니다.
- 기존 인스턴스 가중치를 수정하면 Amazon EC2 Auto Scaling이 새 가중치를 기준으로 원하는 용량에 도달하기 위해 인스턴스를 시작하거나 종료합니다.
- 인스턴스 유형을 제거하는 경우, 인스턴스 유형이 제거되었더라도 해당 인스턴스 유형의 실행 중인 인스턴스는 마지막으로 업데이트된 가중치 값을 계속 갖게 됩니다.

Auto Scaling 그룹의 가중치 추가 또는 수정

기존 Auto Scaling 그룹에 가중치를 추가하거나 새 Auto Scaling 그룹을 생성할 때 가중치를 추가할 수 있습니다. 또한 기존 Auto Scaling 그룹을 업데이트하여 새 구성 옵션(스팟 및 온디맨드 사용량, 스팟 할당 전략, 인스턴스 유형)을 정의할 수 있습니다. 원하는 스팟 또는 온디맨드 인스턴스 수를 변경하면 Amazon EC2 Auto Scaling이 새 구매 옵션에 맞추어 기존 인스턴스를 점진적으로 교체합니다.

인스턴스 가중치를 사용하여 Auto Scaling 그룹을 생성하기 전에 인스턴스 유형이 여러 개인 그룹을 시작하는 데 익숙해지는 것이 좋습니다. 자세한 내용과 추가 예는 [여러 인스턴스 유형 및 구매 옵션이 포함된 Auto Scaling 그룹 \(p. 45\)](#) 단원을 참조하십시오.

다음 예에서는 AWS CLI를 사용하여 Auto Scaling 그룹을 생성할 때 가중치를 추가하는 방법과 기존 Auto Scaling 그룹의 가중치를 추가 또는 수정하는 방법을 보여줍니다. JSON 파일에서 다양한 파라미터를 구성한 다음 JSON 파일을 Auto Scaling 그룹의 유일한 파라미터로 참조할 수 있습니다.

Auto Scaling 그룹을 생성할 때 가중치를 추가하려면

- `create-auto-scaling-group` 명령을 사용하여 새 Auto Scaling 그룹을 생성합니다. 예를 들어 다음 명령은 새 Auto Scaling 그룹을 만들고 다음을 지정하여 인스턴스 가중치를 추가합니다.
 - 온디맨드 인스턴스로 시작할 그룹의 백분율(0) 및 온디맨드 인스턴스의 처음 시작 기본 개수(10)
 - 각 가용 영역의 스팟 인스턴스에 대한 할당 전략(`capacity-optimized`)
 - 우선순위(`m4.16xlarge`, `m5.24xlarge`)에 따라 시작할 인스턴스 유형
 - 인스턴스 유형(16, 24) 간의 상대적 크기 차이(vCPU)에 해당하는 인스턴스 가중치
 - 각각 다른 가용 영역에 해당하는 인스턴스를 시작하는 서브넷(`subnet-5ea0c127`, `subnet-6194ea3b`, `subnet-c934b782`)
 - 시작 템플릿(`my-launch-template`) 및 시작 템플릿 버전(`$Latest`)

```
aws autoscaling create-auto-scaling-group --cli-input-json file://~/config.json
```

다음은 예제 config.json 파일입니다.

```
{
  "AutoScalingGroupName": "my-asg",
  "MixedInstancesPolicy": {
    "LaunchTemplate": {
      "LaunchTemplateSpecification": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "$Latest"
      },
      "Overrides": [
        {
          "InstanceType": "m4.16xlarge",
          "WeightedCapacity": "16"
        },
        {
          "InstanceType": "m5.24xlarge",
          "WeightedCapacity": "24"
        }
      ]
    },
    "InstancesDistribution": {
      "OnDemandBaseCapacity": 10,
      "OnDemandPercentageAboveBaseCapacity": 0,
      "SpotAllocationStrategy": "capacity-optimized"
    }
  },
  "MinSize": 160,
  "MaxSize": 720,
  "DesiredCapacity": 480,
  "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
  "Tags": []
}
```

기존 Auto Scaling 그룹의 가중치를 추가하거나 수정하려면

- `update-auto-scaling-group` 명령을 사용하여 가중치를 추가하거나 수정합니다. 예를 들어 다음을 지정하여 아래 명령을 실행하면 기존 Auto Scaling 그룹의 인스턴스 유형에 가중치가 추가됩니다.
 - 우선 순위(c5.18xlarge, c5.24xlarge, c5.2xlarge, c5.4xlarge)에 따라 시작할 인스턴스 유형
 - 인스턴스 유형(18, 24, 2, 4) 간의 상대적 크기 차이(vCPU)에 해당하는 인스턴스 가중치
 - 가장 큰 가중치보다 크게 증가된 새 원하는 용량

```
aws autoscaling update-auto-scaling-group --cli-input-json file://~/config.json
```

다음은 예제 config.json 파일입니다.

```
{
  "AutoScalingGroupName": "my-existing-asg",
  "MixedInstancesPolicy": {
    "LaunchTemplate": {
      "Overrides": [
        {
          "InstanceType": "c5.18xlarge",
          "WeightedCapacity": "18"
        },
        {

```

```
        "InstanceType": "c5.24xlarge",  
        "WeightedCapacity": "24"  
    },  
    {  
        "InstanceType": "c5.2xlarge",  
        "WeightedCapacity": "2"  
    },  
    {  
        "InstanceType": "c5.4xlarge",  
        "WeightedCapacity": "4"  
    }  
]  
},  
"MinSize": 0,  
"MaxSize": 100,  
"DesiredCapacity": 100  
}
```

Auto Scaling 그룹의 가중치를 확인하려면

- 가중치를 확인하려면 다음 `describe-auto-scaling-groups` 명령을 사용합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

다음은 응답의 예입니다.

```
{  
  "AutoScalingGroups": [  
    {  
      "AutoScalingGroupName": "my-asg",  
      "AutoScalingGroupARN": "arn",  
      "MixedInstancesPolicy": {  
        "LaunchTemplate": {  
          "LaunchTemplateSpecification": {  
            "LaunchTemplateId": "lt-0b97f1e282EXAMPLE",  
            "LaunchTemplateName": "my-launch-template",  
            "Version": "$Latest"  
          },  
          "Overrides": [  
            {  
              "InstanceType": "m4.16xlarge",  
              "WeightedCapacity": "16"  
            },  
            {  
              "InstanceType": "m5.24xlarge",  
              "WeightedCapacity": "24"  
            }  
          ]  
        },  
        "InstancesDistribution": {  
          "OnDemandAllocationStrategy": "prioritized",  
          "OnDemandBaseCapacity": 10,  
          "OnDemandPercentageAboveBaseCapacity": 0,  
          "SpotAllocationStrategy": "capacity-optimized"  
        }  
      },  
      "MinSize": 160,  
      "MaxSize": 720,  
      "DesiredCapacity": 480,  
      "DefaultCooldown": 300,  
      "AvailabilityZones": [  
        "us-east-1a",  
        "us-east-1b",  
        "us-east-1c"  
      ]  
    }  
  ]  
}
```

```

        "us-west-2a",
        "us-west-2b",
        "us-west-2c"
    ],
    "LoadBalancerNames": [],
    "TargetGroupARNs": [],
    "HealthCheckType": "EC2",
    "HealthCheckGracePeriod": 0,
    "Instances": [
        {
            "InstanceId": "i-027327f0ace86f499",
            "InstanceType": "m5.24xlarge",
            "AvailabilityZone": "us-west-2a",
            "LifecycleState": "InService",
            "HealthStatus": "Healthy",
            "LaunchTemplate": {
                "LaunchTemplateId": "lt-0b97f1e282EXAMPLE",
                "LaunchTemplateName": "my-launch-template",
                "Version": "7"
            },
            "ProtectedFromScaleIn": false,
            "WeightedCapacity": "24"
        },
        {
            "InstanceId": "i-0ec0d761cc134878d",
            "InstanceType": "m4.16xlarge",
            "AvailabilityZone": "us-west-2a",
            "LifecycleState": "Pending",
            "HealthStatus": "Healthy",
            "LaunchTemplate": {
                "LaunchTemplateId": "lt-0b97f1e282EXAMPLE",
                "LaunchTemplateName": "my-launch-template",
                "Version": "7"
            },
            "ProtectedFromScaleIn": false,
            "WeightedCapacity": "16"
        },
        ...
    ]
}

```

인스턴스 유형에 대한 권장 사항 가져오기

AWS는 AWS Compute Optimizer에서 제공하는 기능을 사용하여 성능 향상, 비용 절감 또는 두 가지 모두를 지원하는 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 최근 지표 기록을 분석합니다. 그런 다음 컴파일된 데이터를 사용하여 기존 성능 워크로드를 처리하는 데 가장 적합하게 최적화된 Amazon EC2 인스턴스 유형을 권장합니다. 권장 사항은 시간당 인스턴스 요금과 함께 반환됩니다.

Note

Compute Optimizer에서 권장 사항을 받으려면 먼저 Compute Optimizer를 옵트인해야 합니다. 자세한 내용은 AWS Compute Optimizer 사용 설명서의 [AWS AWS Compute Optimizer 시작하기](#)를 참조하십시오.

목차

- [제한 사항 \(p. 80\)](#)

- [결과](#) (p. 80)
- [권장 사항 보기](#) (p. 80)
- [권장 사항 평가를 위한 고려 사항](#) (p. 81)

제한 사항

Compute Optimizer는 현재 M, C, R, T 및 X 인스턴스 유형에 대한 권장 사항을 생성합니다. 다른 인스턴스 유형은 Compute Optimizer에서 고려하지 않습니다. 다른 인스턴스 유형을 사용하면 권장 사항에서 제외됩니다.

현재 Compute Optimizer에서는 원하는 용량, 최소 용량 및 최대 용량에 대해 동일한 값을 가지며 단일 인스턴스 유형을 시작하도록 구성된 Auto Scaling 그룹에 대한 권장 사항을 생성합니다.

결과

Compute Optimizer는 Auto Scaling 그룹에 대한 결과를 다음과 같이 분류합니다.

- 최적화되지 않음 – 워크로드에 더 나은 성능을 제공할 수 있는 권장 사항을 Compute Optimizer에서 확인한 경우 Auto Scaling 그룹이 최적화되지 않은 것으로 간주됩니다.
- 최적화됨 – 선택한 인스턴스 유형에 따라 워크로드를 실행하도록 그룹이 올바르게 프로비저닝되었다고 Compute Optimizer에서 판단한 경우 Auto Scaling 그룹이 최적화된 것으로 간주됩니다. 최적화된 리소스의 경우 Compute Optimizer에서 새로운 인스턴스 유형을 권장하는 경우가 있습니다.
- 없음 – 이 Auto Scaling 그룹에 대한 권장 사항이 없습니다. 이 문제는 Compute Optimizer를 옵트인한지 12시간이 지나지 않았거나, Auto Scaling 그룹이 실행된지 30시간이 지나지 않았거나, Auto Scaling 그룹 또는 인스턴스 유형이 Compute Optimizer에서 지원되지 않는 경우에 발생할 수 있습니다. 자세한 내용은 이전 섹션의 [제한 사항](#) (p. 80) 단원을 참조하십시오.

권장 사항 보기

Compute Optimizer를 옵트인한 후 Auto Scaling 그룹에 대해 생성되는 결과 및 권장 사항을 볼 수 있습니다. 최근에 옵트인한 경우 권장 사항을 최대 12시간 동안 사용하지 못할 수 있습니다.

Auto Scaling 그룹에 대해 생성된 권장 사항을 보려면

1. <https://console.aws.amazon.com/compute-optimizer/>에서 Compute Optimizer 콘솔을 엽니다.

대시보드 페이지가 열립니다.

2. 모든 Auto Scaling 그룹에 대한 권장 사항 보기를 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 세부 정보 보기를 선택합니다.

기본 테이블 설정에 따라 미리 구성된 보기에서 최대 세 개의 서로 다른 인스턴스 권장 사항이 표시되도록 보기가 변경됩니다. 또한 Auto Scaling 그룹에 대한 최근 CloudWatch 지표 데이터(평균 CPU 사용률, 평균 네트워크 입력 및 평균 네트워크 출력)를 제공합니다.

권장 사항 중 하나를 사용할지 여부를 결정합니다. 성능 향상, 비용 절감 또는 이 두 가지 모두를 위해 최적화할 것인지 여부를 결정합니다.

Auto Scaling 그룹의 인스턴스 유형을 변경하려면 새 시작 템플릿 또는 시작 구성을 생성해야 합니다. 새 시작 템플릿 또는 시작 구성을 사용하도록 Auto Scaling 그룹을 업데이트하고 그룹의 원하는 용량을 늘리면 새 인스턴스 유형으로 새 인스턴스가 시작됩니다. 또는 Auto Scaling 그룹의 기존 인스턴스를 종료하여 새 시작

템플릿 또는 시작 구성을 사용하는 대체 인스턴스를 강제로 시작할 수 있습니다. CloudFormation을 사용하면 업데이트를 자동화하는 옵션도 있습니다.

권장 사항 평가를 위한 고려 사항

새 인스턴스 유형으로 이동하기 전에 다음 사항을 고려하십시오.

- 권장 사항은 사용량을 예측하지 않습니다. 권장 사항은 최근 14일 기간 동안의 사용량을 기준으로 합니다. 향후 사용량 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 및 T2 인스턴스와 같은 성능 순간 확장 가능 인스턴스에 대한 권장 사항을 제공할 수 있습니다. 기존 이상으로 주기적으로 버스트하는 경우 새 인스턴스 유형의 vCPU에 따라 계속 버스트할 수 있어야 합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [성능 순간 확장 가능 인스턴스에 대한 CPU 크레딧 및 기존 성능](#)을 참조하십시오.
- 예약 인스턴스를 구매한 경우 온디맨드 인스턴스에 대한 요금이 예약 인스턴스로 청구될 수 있습니다. 현재 인스턴스 유형을 변경하기 전에 먼저 예약 인스턴스 사용률 및 적용 범위에 미치는 영향을 평가합니다.
- 가능한 경우 최신 인스턴스로의 변환을 고려합니다.
- 다른 인스턴스 패밀리로 마이그레이션할 때 현재 인스턴스 유형과 새 인스턴스 유형이 가상화, 아키텍처 또는 네트워크 유형 측면에서 호환되어야 합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스 크기 조정을 위한 호환성](#) 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.

추가 리소스

이 페이지의 주제 외에도 다음 리소스를 참조하십시오.

- [Amazon EC2 인스턴스 유형](#)
- [AWS Compute Optimizer 사용 설명서](#)

최대 Auto Scaling 인스턴스 수명을 기준으로 인스턴스 교체

최대 인스턴스 수명 기능은 허용되는 최대 시간 동안 서비스된 인스턴스를 교체하는 작업을 수행합니다. 이 주제에서는 이 기능의 주요 측면과 Auto Scaling 그룹에 이 기능을 구성하는 방법을 설명합니다.

먼저 Auto Scaling 그룹에 대한 최대 인스턴스 수명 제한을 구성합니다. 이 제한은 인스턴스가 서비스될 수 있는 최대 시간(초)을 지정합니다. 이 최대 시간은 그룹의 현재 인스턴스와 향후 인스턴스에 적용됩니다. 인스턴스가 최대 시간에 가까워지면 AWS에 의해 종료되며 다시 사용될 수 없습니다.

인스턴스가 항상 최대 시간이 끝날 때 종료되는 것은 아닙니다. 경우에 따라, 최대 인스턴스 수명 제한이 구성된 후 Amazon EC2 Auto Scaling이 즉시 인스턴스 교체를 시작해야 할 수 있습니다. 이는 동시에 모든 인스턴스를 교체하는 상황을 방지하기 위한 것입니다.

필요에 따라 Auto Scaling 그룹의 특정 인스턴스를 교체하지 않으려는 경우 인스턴스 보호 기능을 사용할 수 있습니다. 자세한 내용은 [인스턴스 축소 보호](#) (p. 128) 단원을 참조하십시오.

Important

최소 604,800초(7일)의 시간을 지정해야 합니다. 이는 최대 인스턴스 수명을 위한 최소 요구 사항입니다. 이전에 설정한 값을 해제하려면 새 값 0을 지정합니다.

최대 인스턴스 수명을 구성하려면(콘솔)

일반적인 방법으로 Auto Scaling 그룹을 생성합니다. Auto Scaling 그룹을 생성한 후 그룹을 편집하여 최대 인스턴스 수명을 지정합니다.

최대 인스턴스 수명을 구성하려면(AWS CLI)

AWS CLI를 사용하여 최대 인스턴스 수명을 지정하는 경우 기존 Auto Scaling 그룹에 이 제한을 적용할 수 있습니다. 또한 새 Auto Scaling 그룹을 생성할 때도 이 제한을 적용할 수 있습니다.

새 Auto Scaling 그룹의 경우 `create-auto-scaling-group` 명령을 사용합니다.

```
aws autoscaling create-auto-scaling-group --cli-input-json file://~/config.json
```

다음은 예제 config.json 파일입니다.

```
{
  "AutoScalingGroupName": "my-asg",
  "LaunchTemplate": {
    "LaunchTemplateName": "my-launch-template",
    "Version": "$Latest"
  },
  "MinSize": 1,
  "MaxSize": 5,
  "MaxInstanceLifetime": 2592000,
  "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
  "Tags": []
}
```

기존 Auto Scaling 그룹의 경우 `update-auto-scaling-group` 명령을 사용합니다.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-existing-asg --max-instance-lifetime 2592000
```

Auto Scaling 그룹의 최대 인스턴스 수명을 확인하려면

`describe-auto-scaling-groups` 명령을 사용합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

다음은 응답의 예입니다.

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupName": "my-asg",
      "AutoScalingGroupARN": "arn",
      "LaunchTemplate": {
        "LaunchTemplateId": "lt-0b97f1e282EXAMPLE",
        "LaunchTemplateName": "my-launch-template",
        "Version": "$Latest"
      },
      "MinSize": 1,
      "MaxSize": 5,
      "DesiredCapacity": 1,
    }
  ]
}
```

```
    "DefaultCooldown": 300,
    "AvailabilityZones": [
      "us-west-2a",
      "us-west-2b",
      "us-west-2c"
    ],
    "LoadBalancerNames": [],
    "TargetGroupARNs": [],
    "HealthCheckType": "EC2",
    "HealthCheckGracePeriod": 0,
    "Instances": [
      {
        "InstanceId": "i-04d180b9d5fc578fc",
        "InstanceType": "t2.small",
        "AvailabilityZone": "us-west-2b",
        "LifecycleState": "Pending",
        "HealthStatus": "Healthy",
        "LaunchTemplate": {
          "LaunchTemplateId": "lt-0b97f1e282EXAMPLE",
          "LaunchTemplateName": "my-launch-template",
          "Version": "7"
        },
        "ProtectedFromScaleIn": false
      }
    ],
    "CreatedTime": "2019-11-14T22:56:15.487Z",
    "SuspendedProcesses": [],
    "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
    "EnabledMetrics": [],
    "Tags": [],
    "TerminationPolicies": [
      "Default"
    ],
    "NewInstancesProtectedFromScaleIn": false,
    "ServiceLinkedRoleARN": "arn",
    "MaxInstanceLifetime": 2592000
  }
}
```

Auto Scaling 그룹을 단일 다중 영역 그룹으로 병합

별도의 단일 영역 Auto Scaling 그룹을 여러 가용 영역을 포괄하는 단일 그룹으로 병합하려면 단일 그룹 중 하나의 영역을 다중 영역 그룹으로 재지정합니다. 그런 다음 다른 그룹을 삭제합니다. 이 프로세스는 새로운 다중 영역 그룹이 원래 단일 영역 그룹과 동일한 가용 영역 중 하나에 있는 경우 로드 밸런서 보유에 관계 없이 모든 그룹에 적용됩니다.

다음 예에서는 2개의 서로 다른 가용 영역, us-west-2a 및 us-west-2c에 동일한 그룹이 2개 있는 것으로 가정합니다. 이러한 두 그룹은 다음 사양을 공유합니다.

- 최소 크기 = 2
- 최대 크기 = 5
- 원하는 용량 = 3

영역 병합(AWS CLI)

다음 절차를 사용하여 my-group-a 및 my-group-c를 단일 그룹으로 병합합니다. 이 단일 그룹은 us-west-2a 및 us-west-2c를 모두 포함합니다.

별도의 단일 영역 그룹을 단일 다중 영역 그룹으로 병합하려면

1. my-group-a에 지원되는 가용 영역에 us-west-2c 가용 영역을 추가하려면 다음 `update-auto-scaling-group` 명령을 사용하십시오. 모든 단일 영역 그룹의 인스턴스를 허용하려면 이 그룹의 최대 크기를 늘리십시오.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-group-a \
  --availability-zones "us-west-2a" "us-west-2c" \
  --max-size 10 --min-size 4
```

2. 다음 `set-desired-capacity` 명령을 사용하여 my-group-a의 크기를 늘립니다.

```
aws autoscaling set-desired-capacity --auto-scaling-group-name my-group-a \
  --desired-capacity 6
```

3. (선택 사항) 다음 `describe-auto-scaling-groups` 명령을 사용하여 my-group-a가 새로운 크기로 변경되었는지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-group-a
```

4. 다음 `update-auto-scaling-group` 명령을 사용하여 my-group-c에서 인스턴스를 제거하십시오.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-group-c \
  --min-size 0 --max-size 0
```

5. (선택 사항) 다음 `describe-auto-scaling-groups` 명령을 사용하여 my-group-c에 남아 있는 인스턴스가 없는지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-group-c
```

다음은 예제 출력입니다.

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 300,
      "SuspendedProcesses": [],
      "DesiredCapacity": 0,
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-group-c",
      "DefaultCooldown": 300,
      "MinSize": 0,
      "Instances": [],
      "MaxSize": 0,
      "VPCZoneIdentifier": "null",
      "TerminationPolicies": [
        "Default"
      ],
      "LaunchConfigurationName": "my-launch-config",
      "CreatedTime": "2015-02-26T18:24:14.449Z",
      "AvailabilityZones": [
        "us-west-2c"
      ],
      "HealthCheckType": "EC2"
    }
  ]
}
```

6. `delete-auto-scaling-group` 명령을 사용하여 `my-group-c`를 삭제합니다.

```
aws autoscaling delete-auto-scaling-group --auto-scaling-group-name my-group-c
```

Auto Scaling 인프라 삭제

조정 인프라를 완전히 삭제하려면 다음 작업을 완료하십시오.

작업

- [Auto Scaling 그룹 삭제 \(p. 85\)](#)
- [\(선택 사항\) 시작 구성 삭제 \(p. 85\)](#)
- [\(선택 사항\) 시작 템플릿 삭제 \(p. 86\)](#)
- [\(선택 사항\) 로드 밸런서 삭제 \(p. 86\)](#)
- [\(선택 사항\) CloudWatch 경보 삭제 \(p. 87\)](#)

Auto Scaling 그룹 삭제

Auto Scaling 그룹을 삭제하면 원하는 값과 최소, 최대값이 0으로 설정됩니다. 그리고 인스턴스가 종료됩니다. 인스턴스를 삭제하면 연결된 로그 또는 데이터, 그리고 인스턴스의 모든 볼륨도 모두 삭제됩니다. 하나 이상의 인스턴스를 종료하지 않으려면 Auto Scaling 그룹을 삭제하기 전에 인스턴스를 분리할 수 있습니다.

Auto Scaling 그룹을 삭제하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹 페이지에서 Auto Scaling 그룹 옆의 확인란을 선택하고 삭제를 선택합니다(이전 콘솔: 작업, 삭제를 선택합니다).
4. 확인 메시지가 나타나면 Delete를 선택합니다.

이름 옆의 로딩 아이콘은 Auto Scaling 그룹이 삭제 중임을 나타냅니다. Desired(희망), 최소 및 최대 열에 Auto Scaling 그룹에 대한 인스턴스가 0으로 표시됩니다. 인스턴스를 종료하고 그룹을 삭제하는 데 몇 분 정도 걸립니다. 목록을 새로 고침하여 상태를 확인합니다.

Auto Scaling 그룹을 삭제하려면(AWS CLI)

다음 `delete-auto-scaling-group` 명령을 사용하여 Auto Scaling 그룹을 삭제합니다.

```
aws autoscaling delete-auto-scaling-group --auto-scaling-group-name my-asg
```

그룹에 진행 중인 인스턴스 또는 조정 활동이 있는 경우 `delete-auto-scaling-group` 명령을 `--force-delete` 옵션과 함께 사용합니다. 그러면 Amazon EC2 인스턴스도 종료됩니다.

```
aws autoscaling delete-auto-scaling-group --auto-scaling-group-name my-asg --force-delete
```

(선택 사항) 시작 구성 삭제

나중에 시작 구성을 사용하려면 이 단계를 건너뛸 수 있습니다.

시작 구성을 삭제하는 방법(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 시작 구성을 선택합니다.
3. 시작 구성 페이지에서 시작 구성을 선택한 다음 작업, Delete launch configuration(시작 구성 삭제)를 차례로 선택합니다.
4. 확인 메시지가 나타나면 예, 삭제합니다를 선택합니다.

시작 구성을 삭제하는 방법(AWS CLI)

다음 `delete-launch-configuration` 명령을 사용합니다.

```
aws autoscaling delete-launch-configuration --launch-configuration-name my-launch-config
```

(선택 사항) 시작 템플릿 삭제

시작 템플릿 또는 시작 템플릿의 한 가지 버전만 삭제할 수 있습니다. 시작 템플릿을 삭제하면 모든 해당 버전이 삭제됩니다.

나중에 시작 템플릿을 사용하려면 이 단계를 건너뛸 수 있습니다.

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 선택한 다음, 다음 중 하나를 수행하십시오.
 - [Actions]와 [Delete template]를 차례로 선택합니다. 확인 메시지가 나타나면 Delete launch template을 선택합니다.
 - [Actions]와 [Delete template version]을 차례로 선택합니다. 삭제할 버전을 선택하고 시작 템플릿 버전 삭제를 선택합니다.

시작 템플릿 삭제(AWS CLI)

`delete-launch-template` 명령을 사용하여 템플릿과 모든 버전을 삭제합니다.

```
aws ec2 delete-launch-template --launch-template-id lt-068f72b72934aff71
```

또는 `delete-launch-template-versions` 명령을 사용하여 시작 템플릿의 특정 버전을 삭제할 수 있습니다.

```
aws ec2 delete-launch-template-versions --launch-template-id lt-068f72b72934aff71 --  
versions 1
```

(선택 사항) 로드 밸런서 삭제

Auto Scaling 그룹이 Elastic Load Balancing 로드 밸런서에 연결되지 않았거나 로드 밸런서를 나중에 사용하려면 이 단계를 건너뛰십시오.

로드 밸런서를 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Load Balancers]를 클릭합니다.

3. 로드 밸런서를 선택한 다음 작업, 삭제를 선택합니다.
4. 확인 메시지가 나타나면 예, 삭제합니다를 선택합니다.

대상 그룹을 삭제하려면(콘솔)

1. 탐색 창에서 [Target Groups]를 선택합니다.
2. 대상 그룹을 선택하고 [Actions], [Delete]를 차례로 선택합니다.
3. 확인 메시지가 표시되면 [Yes]를 선택합니다.

Auto Scaling 그룹과 연결된 로드 밸런서를 삭제하려면(AWS CLI)

Application Load Balancer 및 Network Load Balancer의 경우, 다음 `delete-load-balancer` 및 `delete-target-group` 명령을 사용합니다.

```
aws elbv2 delete-load-balancer --load-balancer-arn my-load-balancer-arn
aws elbv2 delete-target-group --target-group-arn my-target-group-arn
```

Classic Load Balancer의 경우 다음 `delete-load-balancer` 명령을 사용합니다.

```
aws elb delete-load-balancer --load-balancer-name my-load-balancer
```

(선택 사항) CloudWatch 경보 삭제

Auto Scaling 그룹과 관련된 CloudWatch 경보를 삭제하려면 다음 단계를 완료하십시오.

Auto Scaling 그룹이 CloudWatch 경보와 연결되지 않았거나 나중에 경보를 사용하려면 이 단계를 건너뛸 수 있습니다.

Note

Auto Scaling 그룹을 삭제하면 대상 추적 조정 정책에 대해 Amazon EC2 Auto Scaling이 관리하는 CloudWatch 경보가 자동으로 삭제됩니다.

CloudWatch 경보를 삭제하려면(콘솔)

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 [Alarms]를 선택합니다.
3. 경보를 선택하고 작업, 삭제를 선택합니다.
4. 확인 메시지가 나타나면 Delete를 선택합니다.

CloudWatch 경보를 삭제하려면(AWS CLI)

`delete-alarms` 명령을 사용합니다. 한번에 하나 이상 경보를 삭제할 수 있습니다. 예를 들어, 다음 명령을 사용하여 Step-Scaling-AlarmHigh-AddCapacity 및 Step-Scaling-AlarmLow-RemoveCapacity 경보를 삭제합니다.

```
aws cloudwatch delete-alarms --alarm-name Step-Scaling-AlarmHigh-AddCapacity Step-Scaling-AlarmLow-RemoveCapacity
```

Auto Scaling 그룹의 크기 조정

조정은 애플리케이션의 컴퓨팅 용량을 늘리거나 줄이는 기능입니다. 조정은 이벤트와 함께 시작되거나 Auto Scaling 그룹에 Amazon EC2 인스턴스를 시작 또는 종료하도록 지시하는 조정 작업과 함께 시작됩니다.

Amazon EC2 Auto Scaling에서는 여러 가지 방법으로 애플리케이션의 요구 사항에 가장 적합하게 조정 기능을 조절할 수 있습니다. 그러므로 애플리케이션을 충분히 이해하는 것이 중요합니다. 다음 사항에 유의하십시오.

- Amazon EC2 Auto Scaling이 애플리케이션 아키텍처에서 수행해야 하는 역할은 무엇인가? 자동 조정을 주로 용량을 늘리거나 줄이는 수단으로 생각하는 것이 일반적이지만, 이는 안정적인 서버 수를 유지하려는 경우에도 유용합니다.
- 나에게 중요한 비용 제약 조건은 무엇인가? Amazon EC2 Auto Scaling에서는 EC2 인스턴스를 사용하므로 사용한 리소스에 대해서만 비용을 지불합니다. 비용 제약 조건을 알면 애플리케이션의 확장 시기와 규모를 결정하는 데 도움이 될 수 있습니다.
- 애플리케이션에 중요한 지표는 무엇인가? Amazon CloudWatch는 Auto Scaling 그룹과 함께 사용할 수 있는 여러 가지 지표를 지원합니다.

목차

- [조정 옵션 \(p. 88\)](#)
- [Auto Scaling 그룹에 대한 용량 제한 설정 \(p. 89\)](#)
- [Auto Scaling 그룹에서 일정한 수의 인스턴스 유지 \(p. 90\)](#)
- [Amazon EC2 Auto Scaling 수동 조정 \(p. 90\)](#)
- [Amazon EC2 Auto Scaling의 예약된 조정 \(p. 99\)](#)
- [Amazon EC2 Auto Scaling의 동적 조정 \(p. 102\)](#)
- [Amazon EC2 Auto Scaling의 조정 휴지 \(p. 123\)](#)
- [축소 시 Auto Scaling 인스턴스 종료 제어 \(p. 126\)](#)
- [Amazon EC2 Auto Scaling 수명 주기 후크 \(p. 131\)](#)
- [Auto Scaling 그룹에서 일시적으로 인스턴스 제거 \(p. 138\)](#)
- [조정 프로세스의 일시 중지 및 재개 \(p. 142\)](#)

조정 옵션

Amazon EC2 Auto Scaling은 Auto Scaling 그룹을 조정하는 다양한 방법을 제공합니다.

항상 현재 인스턴스 수준 유지 관리

지정된 수의 실행 인스턴스를 항상 유지하도록 Auto Scaling 그룹을 구성할 수 있습니다. 현재 인스턴스 수준을 유지 관리하기 위해 Amazon EC2 Auto Scaling 그룹에서 Auto Scaling 그룹 내에 실행 중인 인스턴스에 대한 주기적인 상태 확인을 수행합니다. Amazon EC2 Auto Scaling에서 인스턴스가 비정상 상태임을 확인하면 해당 인스턴스를 종료한 다음 새 인스턴스를 시작합니다. 자세한 내용은 [Auto Scaling 그룹에서 일정한 수의 인스턴스 유지 \(p. 90\)](#) 단원을 참조하십시오.

수동 조정

수동 조정은 리소스를 확장하는 가장 기본적인 방법으로, Auto Scaling 그룹에서 최대, 최소 또는 원하는 용량의 변경 사항만 지정하는 경우 사용할 수 있습니다. Amazon EC2 Auto Scaling은 인스턴스를 생성 또는 종료하는 프로세스를 관리하여 업데이트된 용량을 유지 관리합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 수동 조정 \(p. 90\)](#) 단원을 참조하십시오.

일정을 기반으로 조정

일정에 따른 확장은 확장 작업이 시간 및 날짜 함수에 따라 자동으로 수행됨을 의미합니다. 그룹의 인스턴스 수를 늘려야 할지 또는 줄여야 할지 정확히 아는 경우에 유용합니다. 그 이유는 예측 가능한 일정에 따라 수요가 증가하기 때문입니다. 자세한 내용은 [Amazon EC2 Auto Scaling의 예약된 조정 \(p. 99\)](#) 단원을 참조하십시오.

온디맨드 기반 조정

조정 정책을 사용하여 리소스를 조정하는 보다 향상된 방식을 사용하면 조정 프로세스를 제어하는 파라미터를 정의할 수 있습니다. 예를 들어, 현재 인스턴스 2개에서 웹 애플리케이션이 실행되고 있고 사용자가 애플리케이션 로드에서 변경이 있는 경우 Auto Scaling 그룹의 CPU 사용량을 50% 정도로 유지시키려 한다고 가정해 보겠습니다. 이 방법은 이러한 조건이 언제 바뀔지 모를 때, 조건 변화에 따라 조정할 경우에 유용합니다. Amazon EC2 Auto Scaling이 사용자에게 응답하도록 설정할 수 있습니다. 자세한 내용은 [Amazon EC2 Auto Scaling의 동적 조정 \(p. 102\)](#) 단원을 참조하십시오.

예측 조정 사용

Amazon EC2 Auto Scaling 및 AWS Auto Scaling를 함께 사용하여 여러 서비스에서 리소스를 조정할 수도 있습니다. AWS Auto Scaling는 예측적 조정과 동적 조정(각기 사전 예방식 접근 방식 및 사후 대처식 접근 방식)을 결합하여 Amazon EC2 용량을 보다 신속하게 조정함으로써 최적의 가용성 및 성능을 유지하도록 도와줍니다. 자세한 내용은 [AWS Auto Scaling 사용 설명서](#)를 참조하십시오.

Auto Scaling 그룹에 대한 용량 제한 설정

Auto Scaling 그룹의 크기를 구성하려면 최소, 최대 및 원하는 용량을 설정합니다. 최소 및 최대 용량은 Auto Scaling 그룹을 생성하는 데 필수적이며, 원하는 용량은 선택 사항입니다. 원하는 용량이 처음에 정의되지 않은 경우 최소 용량으로 기본 설정됩니다.

Note

기본적으로 콘솔에서 Auto Scaling 그룹을 생성할 때 최소, 최대 및 원하는 용량은 하나의 인스턴스로 설정됩니다. 원하는 용량을 변경하면 지정된 용량이 Auto Scaling 그룹을 만든 직후에 시작되는 총 인스턴스 수가 됩니다.

Auto Scaling 그룹은 최소 용량과 최대 용량에 서로 다른 값을 갖는 한 탄력적입니다. Auto Scaling 그룹의 원하는 용량을 변경(수동 조정 또는 자동 조정)하기 위한 모든 요청은 이러한 제한 범위 내에서 이루어져야 합니다.

그룹을 자동으로 조정하도록 선택한 경우, 최대 제한에 따라 Amazon EC2 Auto Scaling에서 수요 증가에 맞춰 필요 시 인스턴스 수를 확장할 수 있습니다. 최소 제한은 항상 특정 수의 인스턴스가 실행되도록 합니다.

이러한 제한은 수동으로 Auto Scaling 그룹을 조정할 때도 적용됩니다. 자동 조정을 해제하고 그룹을 일시적 또는 영구적으로 일정한 크기로 실행하려는 경우를 예로 들 수 있습니다.

콘솔에서 용량 설정에 액세스하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. 보려는 Auto Scaling 그룹을 선택합니다.

4. 세부 정보 탭에서 최소, 최대 및 원하는 용량에 대한 현재 설정을 보거나 변경합니다.

Auto Scaling 그룹에서 일정한 수의 인스턴스 유지

Auto Scaling 그룹을 생성하면 Auto Scaling 그룹이 충분한 EC2 인스턴스를 시작하여 최소 용량(또는 지정된 목표 용량)을 충족합니다.

일정한 수의 인스턴스가 필요한 경우 최소, 최대 및 원하는 용량에 동일한 값을 설정하면 됩니다. Auto Scaling 그룹에 연결된 기타 조정 조건이 없는 경우에는 그룹은 인스턴스가 비정상 상태가 될 경우에도 이 개수의 실행 인스턴스를 유지합니다.

동일한 인스턴스 수를 유지하기 위해 Amazon EC2 Auto Scaling 그룹에서는 Auto Scaling 그룹 내 실행 중인 인스턴스에 대한 주기적인 상태 확인을 수행합니다. 인스턴스가 비정상 상태임이 확인되면 해당 인스턴스를 종료한 다음 새 인스턴스를 시작합니다. 실행 인스턴스를 중지 또는 종료할 경우 인스턴스가 비정상인 것으로 간주되어 교체됩니다. 상태 확인 대체에 대한 자세한 내용은 [Auto Scaling 인스턴스 상태 확인 \(p. 148\)](#) 단원을 참조하십시오.

Auto Scaling 그룹을 수동으로 조정하려면 원하는 용량을 조정하여 Amazon EC2 Auto Scaling에서 유지할 인스턴스 수를 업데이트합니다. 원하는 용량을 최소 및 최대 용량 범위를 벗어난 값으로 조정하려면 먼저 이러한 제한을 업데이트해야 합니다.

Amazon EC2 Auto Scaling 수동 조정

언제든지 기존 Auto Scaling 그룹의 크기를 수동으로 변경할 수 있습니다. 또한 Auto Scaling 그룹의 용량을 원하는 대로 업데이트하거나 Auto Scaling 그룹에 연결된 인스턴스를 업데이트할 수도 있습니다. 자동 조정이 필요하지 않거나 일정한 수의 인스턴스로 용량을 유지해야 할 경우 그룹을 수동으로 조정하는 것이 유용할 수 있습니다.

Auto Scaling 그룹의 크기 변경(콘솔)

Auto Scaling 그룹의 원하는 용량을 변경하면 Amazon EC2 Auto Scaling에서 인스턴스를 시작하거나 종료하는 프로세스를 관리하여 새 그룹 크기를 유지합니다.

다음 예에서는 최소 크기가 1이고 최대 크기가 5인 Auto Scaling 그룹을 생성한 것으로 가정합니다. 따라서 그룹에는 현재 실행 중인 인스턴스가 1개입니다.

Auto Scaling 그룹의 크기를 변경하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. [Details] 탭에서 [Edit]를 선택합니다.
5. 원하는 용량에서 하나씩 원하는 용량을 늘립니다. 예를 들어, 현재 값이 1인 경우 2를 입력합니다.

원하는 용량은 그룹의 최대 크기보다 작거나 같아야 합니다. 원하는 용량에 대한 새 값이 최대 용량보다 크면 최대 용량을 업데이트해야 합니다.

6. 작업을 마쳤으면 변경 사항 저장을 선택합니다.

이제 Auto Scaling 그룹에서 추가 인스턴스를 시작했는지 확인합니다.

Auto Scaling 그룹의 크기가 변경되었는지 확인하려면

1. 활동 탭에서 활동 기록의 상태 열에 인스턴스의 현재 상태가 표시됩니다. 인스턴스 상태가 성공으로 변경될 때까지 새로 고침 버튼을 사용합니다. 이는 Auto Scaling 그룹이 새 인스턴스를 성공적으로 시작했음을 나타냅니다.

Note

인스턴스가 시작되지 않으면 [Amazon EC2 Auto Scaling 문제 해결 \(p. 200\)](#)에서 문제 해결 팁을 찾을 수 있습니다.

2. 인스턴스 관리 탭의 인스턴스에서 수명 주기 열에 인스턴스의 상태가 표시됩니다. 인스턴스를 시작하는데 약간 시간이 걸립니다. 인스턴스가 시작되면 상태가 InService로 변경됩니다. 이제 Auto Scaling 그룹에서 1의 새 인스턴스를 시작했고 InService 상태에 있음을 알 수 있습니다.

Auto Scaling 그룹의 크기 변경(AWS CLI)

Auto Scaling 그룹의 크기를 변경하면 Amazon EC2 Auto Scaling에서 인스턴스를 시작하거나 종료하는 프로세스를 관리하여 새 그룹 크기를 유지합니다. 기본 동작은 기본 휴지 기간이 완료되기를 기다리는 것이 아니지만, 이 기본 동작을 재정의하여 휴지 기간이 완료되기를 기다릴 수 있습니다. 자세한 내용은 [Amazon EC2 Auto Scaling의 조정 휴지 \(p. 123\)](#) 단원을 참조하십시오.

다음 예에서는 최소 크기가 1이고 최대 크기가 5인 Auto Scaling 그룹을 생성한 것으로 가정합니다. 따라서 그룹에는 현재 실행 중인 인스턴스가 1개입니다.

Auto Scaling 그룹의 크기를 변경하려면

`set-desired-capacity` 명령을 사용하여 다음 예와 같이 Auto Scaling 그룹의 크기를 변경합니다.

```
aws autoscaling set-desired-capacity --auto-scaling-group-name my-asg \
  --desired-capacity 2
```

Auto Scaling 그룹의 기본 휴지 기간을 지키기로 한 경우, 다음 예에 표시된 것처럼 `--honor-cooldown` 옵션을 지정해야 합니다.

```
aws autoscaling set-desired-capacity --auto-scaling-group-name my-asg \
  --desired-capacity 2 --honor-cooldown
```

Auto Scaling 그룹의 크기를 확인하려면

다음 예와 같이 `describe-auto-scaling-groups` 명령을 사용하여 Auto Scaling 그룹의 크기가 변경되었음을 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

다음은 시작된 그룹 및 인스턴스의 세부 정보를 포함한 예제 출력입니다.

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "ServiceLinkedRoleARN": "arn",
      "TargetGroupARNs": [],
      "SuspendedProcesses": [],
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      }
    }
  ],
  "NextToken": null
}
```

```
    "Tags": [],
    "EnabledMetrics": [],
    "LoadBalancerNames": [],
    "AutoScalingGroupName": "my-asg",
    "DefaultCooldown": 300,
    "MinSize": 1,
    "Instances": [
      {
        "ProtectedFromScaleIn": false,
        "AvailabilityZone": "us-west-2a",
        "LaunchTemplate": {
          "LaunchTemplateName": "my-launch-template",
          "Version": "1",
          "LaunchTemplateId": "lt-050555ad16a3f9c7f"
        },
        "InstanceId": "i-05b4f7d5be44822a6",
        "HealthStatus": "Healthy",
        "LifecycleState": "Pending"
      },
      {
        "ProtectedFromScaleIn": false,
        "AvailabilityZone": "us-west-2a",
        "LaunchTemplate": {
          "LaunchTemplateName": "my-launch-template",
          "Version": "1",
          "LaunchTemplateId": "lt-050555ad16a3f9c7f"
        },
        "InstanceId": "i-0c20ac468fa3049e8",
        "HealthStatus": "Healthy",
        "LifecycleState": "InService"
      }
    ],
    "MaxSize": 5,
    "VPCZoneIdentifier": "subnet-c87f2be0",
    "HealthCheckGracePeriod": 300,
    "TerminationPolicies": [
      "Default"
    ],
    "CreatedTime": "2019-03-18T23:30:42.611Z",
    "AvailabilityZones": [
      "us-west-2a"
    ],
    "HealthCheckType": "EC2",
    "NewInstancesProtectedFromScaleIn": false,
    "DesiredCapacity": 2
  ]
}
```

DesiredCapacity에 새 값이 표시됩니다. Auto Scaling 그룹에서 추가 인스턴스를 시작했습니다.

Auto Scaling 그룹에 EC2 인스턴스 연결

Amazon EC2 Auto Scaling은 기존 Auto Scaling 그룹에 하나 이상의 EC2 인스턴스를 연결하여 자동 조정을 활성화하는 옵션을 제공합니다. 인스턴스가 연결되면 Auto Scaling 그룹의 일부가 됩니다.

연결할 인스턴스는 다음 기준을 충족해야 합니다.

- 인스턴스는 running 상태에 있습니다.
- 인스턴스를 시작할 때 사용되는 AMI가 항상 있어야 합니다.
- 인스턴스가 다른 Auto Scaling 그룹의 구성원이 아닙니다.
- 인스턴스가 Auto Scaling 그룹에 정의된 가용 영역 중 하나로 시작됩니다.

- Auto Scaling 그룹에 연결된 클래식 로드 밸런서가 있는 경우, 인스턴스와 로드 밸런서가 모두 EC2- 또는 동일한 VPC에 있어야 합니다. 그룹에 연결된 대상 그룹이 있는 경우, 인스턴스와 로드 밸런서는 모두 동일한 VPC에 있어야 합니다.

인스턴스를 연결하면 연결되는 인스턴스 수에 따라 그룹의 원하는 용량을 늘립니다. 연결할 인스턴스 수와 원하는 용량의 합이 그룹의 최대 크기를 초과하는 경우 요청이 실패합니다.

연결된 로드 밸런서가 있는 Auto Scaling 그룹에 인스턴스를 연결하면 해당 인스턴스가 로드 밸런서에 등록됩니다. 연결된 대상 그룹이 있는 Auto Scaling 그룹에 인스턴스를 연결하면 해당 인스턴스가 대상 그룹에 등록됩니다.

이번 예에서는 다음과 같이 구성된 Auto Scaling 그룹을 사용합니다.

- Auto Scaling 그룹 이름 = my-asg
- 최소 크기 = 1
- 최대 크기 = 5
- 원하는 용량 = 2
- 가용 영역 = us-west-2a

인스턴스 연결(콘솔)

기존 Auto Scaling 그룹에 기존 인스턴스를 연결하거나 새 Auto Scaling 그룹을 생성하여 연결할 수 있습니다.

새 Auto Scaling 그룹에 인스턴스를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스를 선택한 후 인스턴스를 선택합니다.
3. [Actions], [Instance Settings], [Attach to Auto Scaling Group]을 차례로 선택합니다.
4. Auto Scaling 그룹에 연결 페이지에서 새 Auto Scaling 그룹을 선택하고 해당 그룹의 이름을 입력한 다음 연결을 선택합니다.

새 Auto Scaling 그룹이 Auto Scaling 그룹에 대해 지정한 이름과 동일한 새 시작 구성을 사용하여 생성됩니다. 이 시작 구성은 연결한 인스턴스의 설정(예: 보안 그룹 및 IAM 역할)을 가져옵니다. Auto Scaling 그룹은 연결한 인스턴스의 설정(예: 가용 영역 및 서브넷)을 가져오고 원하는 용량과 1의 최소 크기를 가집니다.

5. (선택 사항) Auto Scaling 그룹의 설정을 변경하려면 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다. 새 Auto Scaling 그룹을 선택하고 편집을 선택하여 필요한 설정을 변경한 다음 저장을 선택합니다.

기존의 Auto Scaling 그룹에 인스턴스를 연결하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. (선택 사항) 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다. Auto Scaling 그룹을 선택하고 Auto Scaling 그룹의 최대 크기가 다른 인스턴스를 추가할 수 있을 만큼 큰지 확인합니다. 그렇지 않으면 [Edit]를 선택하여 최대 크기를 늘린 다음 [Save]를 선택합니다.
3. 탐색 창에서 인스턴스를 선택한 후 인스턴스를 선택합니다.
4. [Actions], [Instance Settings], [Attach to Auto Scaling Group]을 차례로 선택합니다.
5. Auto Scaling 그룹에 연결 페이지에서 기존 Auto Scaling 그룹, 인스턴스를 차례로 선택한 다음 연결을 선택합니다.
6. 인스턴스가 기준을 충족하지 않으면 세부 정보가 포함된 오류 메시지가 표시됩니다. 예를 들어 Auto Scaling 그룹과 동일한 가용 영역에 있지 않을 수도 있습니다. 닫기를 선택하고 기준에 맞는 인스턴스로 다시 시도합니다.

인스턴스 연결(AWS CLI)

Auto Scaling 그룹에 인스턴스를 연결하려면

1. 다음 `describe-auto-scaling-groups` 명령을 사용하여 특정 Auto Scaling 그룹을 명시합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-names my-asg
```

다음 예의 응답은 원하는 용량이 2이고 해당 그룹에 실행 중인 인스턴스가 2개임을 보여 줍니다.

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "ServiceLinkedRoleARN": "arn",
      "TargetGroupARNs": [],
      "SuspendedProcesses": [],
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      },
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-asg",
      "DefaultCooldown": 300,
      "MinSize": 1,
      "Instances": [
        {
          "ProtectedFromScaleIn": false,
          "AvailabilityZone": "us-west-2a",
          "LaunchTemplate": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1",
            "LaunchTemplateId": "lt-050555ad16a3f9c7f"
          },
          "InstanceId": "i-05b4f7d5be44822a6",
          "HealthStatus": "Healthy",
          "LifecycleState": "Pending"
        },
        {
          "ProtectedFromScaleIn": false,
          "AvailabilityZone": "us-west-2a",
          "LaunchTemplate": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1",
            "LaunchTemplateId": "lt-050555ad16a3f9c7f"
          },
          "InstanceId": "i-0c20ac468fa3049e8",
          "HealthStatus": "Healthy",
          "LifecycleState": "InService"
        }
      ],
      "MaxSize": 5,
      "VPCZoneIdentifier": "subnet-c87f2be0",
      "HealthCheckGracePeriod": 300,
      "TerminationPolicies": [
        "Default"
      ],
      "CreatedTime": "2019-03-18T23:30:42.611Z",
      "AvailabilityZones": [
        "us-west-2a"
      ]
    }
  ]
}
```

```
    ],  
    "HealthCheckType": "EC2",  
    "NewInstancesProtectedFromScaleIn": false,  
    "DesiredCapacity": 2  
  }  
]  
}
```

2. 다음 `attach-instances` 명령을 사용하여 Auto Scaling 그룹에 인스턴스를 연결합니다.

```
aws autoscaling attach-instances --instance-ids i-0787762faf1c28619 --auto-scaling-  
group-name my-asg
```

3. 다음 `describe-auto-scaling-groups` 명령을 사용하여 인스턴스가 연결되었는지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-names my-asg
```

다음 예의 응답은 원하는 용량이 1 증가하여 3개 인스턴스가 되었고 새 인스턴스는 i-0787762faf1c28619임을 보여 줍니다.

```
{  
  "AutoScalingGroups": [  
    {  
      "AutoScalingGroupARN": "arn",  
      "ServiceLinkedRoleARN": "arn",  
      "TargetGroupARNs": [],  
      "SuspendedProcesses": [],  
      "LaunchTemplate": {  
        "LaunchTemplateName": "my-launch-template",  
        "Version": "1",  
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"  
      },  
      "Tags": [],  
      "EnabledMetrics": [],  
      "LoadBalancerNames": [],  
      "AutoScalingGroupName": "my-asg",  
      "DefaultCooldown": 300,  
      "MinSize": 1,  
      "Instances": [  
        {  
          "ProtectedFromScaleIn": false,  
          "AvailabilityZone": "us-west-2a",  
          "LaunchTemplate": {  
            "LaunchTemplateName": "my-launch-template",  
            "Version": "1",  
            "LaunchTemplateId": "lt-050555ad16a3f9c7f"  
          },  
          "InstanceId": "i-05b4f7d5be44822a6",  
          "HealthStatus": "Healthy",  
          "LifecycleState": "Pending"  
        },  
        {  
          "ProtectedFromScaleIn": false,  
          "AvailabilityZone": "us-west-2a",  
          "LaunchTemplate": {  
            "LaunchTemplateName": "my-launch-template",  
            "Version": "1",  
            "LaunchTemplateId": "lt-050555ad16a3f9c7f"  
          },  
          "InstanceId": "i-0c20ac468fa3049e8",  
          "HealthStatus": "Healthy",  
          "LifecycleState": "InService"  
        }  
      ]  
    }  
  ]  
}
```

```
{
  "ProtectedFromScaleIn": false,
  "AvailabilityZone": "us-west-2a",
  "LaunchTemplate": {
    "LaunchTemplateName": "my-launch-template",
    "Version": "1",
    "LaunchTemplateId": "lt-050555ad16a3f9c7f"
  },
  "InstanceId": "i-0787762faf1c28619",
  "HealthStatus": "Healthy",
  "LifecycleState": "InService"
}
],
"MaxSize": 5,
"VPCZoneIdentifier": "subnet-c87f2be0",
"HealthCheckGracePeriod": 300,
"TerminationPolicies": [
  "Default"
],
"CreatedTime": "2019-03-18T23:30:42.611Z",
"AvailabilityZones": [
  "us-west-2a"
],
"HealthCheckType": "EC2",
"NewInstancesProtectedFromScaleIn": false,
"DesiredCapacity": 3
}
]
```

Auto Scaling 그룹에서 EC2 인스턴스 분리

Auto Scaling 그룹에서 인스턴스를 제거(분리)할 수 있습니다. 인스턴스가 분리되면 이러한 인스턴스를 나머지 Auto Scaling 그룹에서 독립적으로 관리할 수 있습니다. 인스턴스를 분리하면 다음을 수행할 수 있습니다.

- 인스턴스를 Auto Scaling 그룹 밖으로 이동한 후 다른 그룹에 연결합니다. 자세한 내용은 [Auto Scaling 그룹에 EC2 인스턴스 연결 \(p. 92\)](#) 단원을 참조하십시오.
- 애플리케이션을 실행하는 기존 인스턴스를 사용하여 Auto Scaling 그룹을 생성하고 테스트합니다. 그런 다음 테스트가 완료되면 이러한 인스턴스를 Auto Scaling 그룹에서 분리할 수 있습니다.

인스턴스를 분리하는 경우 분리할 인스턴스 수로 Auto Scaling 그룹의 원하는 용량을 줄이는 옵션이 제공됩니다. 용량을 줄이지 않도록 선택하면 Amazon EC2 Auto Scaling에서 새 인스턴스를 시작하여 분리한 인스턴스를 교체합니다. 용량을 줄이더라도 동일한 가용 영역에서 여러 인스턴스를 분리하는 경우, AZRebalance 프로세스를 일시 중지하지 않는 한 Amazon EC2 Auto Scaling가 가용 영역을 재분배할 수 있습니다. 자세한 내용은 [조정 프로세스 \(p. 143\)](#) 단원을 참조하십시오.

분리하려는 인스턴스의 수가 Auto Scaling 그룹의 크기를 최소 용량 아래로 감소할 경우, 그 인스턴스를 분리하려면 먼저 그룹에 대한 최소 용량을 줄여야 합니다.

연결된 로드 밸런서가 있는 Auto Scaling 그룹에서 인스턴스를 분리하면, 해당 인스턴스가 로드 밸런서에서 등록 취소됩니다. 연결된 대상 그룹이 있는 Auto Scaling 그룹에서 인스턴스를 분리하면, 해당 인스턴스가 대상 그룹에서 등록 취소됩니다. 로드 밸런서에서 연결 드레이닝이 활성화된 경우 Amazon EC2 Auto Scaling은 진행 중인 요청이 완료될 때까지 기다립니다.

이번 예에서는 다음과 같이 구성된 Auto Scaling 그룹을 사용합니다.

- Auto Scaling 그룹 이름 = my-asg
- 최소 크기 = 1
- 최대 크기 = 5

- 원하는 용량 = 4
- 가용 영역 = us-west-2a

인스턴스 분리(콘솔)

다음 절차에 따라 Auto Scaling 그룹에서 인스턴스를 분리합니다.

기존의 Auto Scaling 그룹에서 인스턴스를 분리하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 인스턴스 관리 탭의 인스턴스에서 인스턴스를 선택하고 작업, 분리를 선택합니다. (이전 콘솔: 인스턴스 탭에서 인스턴스를 분리할 수 있습니다.)
5. 인스턴스 분리 대화 상자에서 확인란을 선택하여 교체 인스턴스를 시작하도록 하거나, 확인란을 선택하지 않고 원하는 용량을 줄입니다. 인스턴스 분리를 선택합니다.

인스턴스 분리(AWS CLI)

다음 절차에 따라 Auto Scaling 그룹에서 인스턴스를 분리합니다.

기존의 Auto Scaling 그룹에서 인스턴스를 분리하려면

1. 다음 [describe-auto-scaling-instances](#) 명령을 사용하여 현재 인스턴스 목록을 표시합니다.

```
aws autoscaling describe-auto-scaling-instances
```

다음 예의 응답은 그룹에 인스턴스 4개가 실행 중임을 보여 줍니다.

```
{
  "AutoScalingInstances": [
    {
      "ProtectedFromScaleIn": false,
      "AvailabilityZone": "us-west-2a",
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      },
      "InstanceId": "i-05b4f7d5be44822a6",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService"
    },
    {
      "ProtectedFromScaleIn": false,
      "AvailabilityZone": "us-west-2a",
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      },
      "InstanceId": "i-0c20ac468fa3049e8",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService"
    }
  ],
}
```



```
{
  "ProtectedFromScaleIn": false,
  "AvailabilityZone": "us-west-2a",
  "LaunchTemplate": {
    "LaunchTemplateName": "my-launch-template",
    "Version": "1",
    "LaunchTemplateId": "lt-050555ad16a3f9c7f"
  },
  "InstanceId": "i-0787762faf1c28619",
  "AutoScalingGroupName": "my-asg",
  "HealthStatus": "HEALTHY",
  "LifecycleState": "InService"
},
{
  "ProtectedFromScaleIn": false,
  "AvailabilityZone": "us-west-2a",
  "LaunchTemplate": {
    "LaunchTemplateName": "my-launch-template",
    "Version": "1",
    "LaunchTemplateId": "lt-050555ad16a3f9c7f"
  },
  "InstanceId": "i-0f280a4c58d319a8a",
  "AutoScalingGroupName": "my-asg",
  "HealthStatus": "HEALTHY",
  "LifecycleState": "InService"
}
]
```

2. 다음 `detach-instances` 명령을 사용하여 인스턴스를 분리하고 원하는 용량으로 줄입니다.

```
aws autoscaling detach-instances --instance-ids i-05b4f7d5be44822a6 \
  --auto-scaling-group-name my-asg --should-decrement-desired-capacity
```

3. 다음 `describe-auto-scaling-instances` 명령을 사용하여 인스턴스가 분리되었는지 확인합니다.

```
aws autoscaling describe-auto-scaling-instances
```

다음 예의 응답은 이제 인스턴스 3개가 실행 중임을 보여 줍니다.

```
{
  "AutoScalingInstances": [
    {
      "ProtectedFromScaleIn": false,
      "AvailabilityZone": "us-west-2a",
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      },
      "InstanceId": "i-0c20ac468fa3049e8",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService"
    },
    {
      "ProtectedFromScaleIn": false,
      "AvailabilityZone": "us-west-2a",
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      },
      "InstanceId": "i-0f280a4c58d319a8a",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService"
    },
    {
      "ProtectedFromScaleIn": false,
      "AvailabilityZone": "us-west-2a",
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      },
      "InstanceId": "i-0787762faf1c28619",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService"
    }
  ]
}
```

```
{
  "InstanceId": "i-0787762faf1c28619",
  "AutoScalingGroupName": "my-asg",
  "HealthStatus": "HEALTHY",
  "LifecycleState": "InService"
},
{
  "ProtectedFromScaleIn": false,
  "AvailabilityZone": "us-west-2a",
  "LaunchTemplate": {
    "LaunchTemplateName": "my-launch-template",
    "Version": "1",
    "LaunchTemplateId": "lt-050555ad16a3f9c7f"
  },
  "InstanceId": "i-0f280a4c58d319a8a",
  "AutoScalingGroupName": "my-asg",
  "HealthStatus": "HEALTHY",
  "LifecycleState": "InService"
}
]
```

Amazon EC2 Auto Scaling의 예약된 조정

예약된 조정을 사용하면 자체 조정 일정을 설정할 수 있습니다. 예를 들어, 매주 수요일에 웹 애플리케이션에 대한 트래픽이 증가하고 목요일까지 높은 상태로 유지되다가 금요일에 줄어들기 시작한다고 가정해 보겠습니다. 웹 애플리케이션의 예측 가능한 트래픽 패턴에 따라 조정 활동을 계획할 수 있습니다. 확장 작업이 시간 및 날짜 함수에 따라 자동으로 수행됩니다.

Note

예측 가능한 로드 변경을 기반으로 하는 조정의 경우, AWS Auto Scaling의 예측 조정기능을 사용할 수도 있습니다. 자세한 내용은 [AWS Auto Scaling 사용 설명서](#)를 참조하십시오.

Auto Scaling 그룹이 일정을 기반으로 조정을 수행하도록 구성하려면 예약된 작업을 생성합니다. 예약 작업은 지정 시간에 Amazon EC2 Auto Scaling에 조정 작업을 수행하라고 지시합니다. 예약된 조정 작업을 생성하려면 조정 작업이 시행되어야 하는 시작 시간을 지정하고, 조정 작업을 위한 최소 및 최대 크기와 원하는 크기를 새로 지정합니다. 지정된 시간에 Amazon EC2 Auto Scaling은 조정 작업에서 정의한 대로 최소, 최대 크기와 원하는 크기에 대한 새 값으로 그룹을 업데이트합니다.

한 번만 조정하거나 반복되는 일정으로 조정하는 예약된 작업을 생성할 수 있습니다.

고려 사항

예약된 작업을 만들 경우 다음 사항에 유의해야 합니다.

- 예약된 작업은 최소, 최대 및 원하는 크기를 예약된 작업에서 지정한 시간에 예약된 작업에 의해 지정된 값으로 설정합니다. 이전 값을 추적하여 종료 시간 이후에 이전 값으로 반환하지 않습니다.
- 예약된 작업은 대체로 수 초 내에 실행됩니다. 하지만 작업이 예약된 시작 시간에서 최대 2분까지 지연될 수 있습니다. 이것은 Auto Scaling 그룹 내의 작업은 지정된 순서대로 실행하기 때문이며 예약된 작업의 예약된 시작 시간이 서로 가까운 경우 실행하는 데 더 많은 시간이 소요될 수 있습니다.
- 그룹 전체가 아니라 동일한 그룹 내에서 예약된 작업의 실행 순서가 보장됩니다.
- 예약된 작업은 고유한 시간 값을 가져야 합니다. 또 다른 조정 작업이 이미 예약된 경우 한 번에 하나의 활동을 예약하려고 시도하면 충돌을 알리는 오류 메시지와 함께 호출이 거부됩니다.
- Auto Scaling 그룹당 최대 125개의 예약된 작업을 만들 수 있습니다.
- 예약된 작업은 종료 시간이 되면 해당 계정에 존속하지 않습니다.

- 예약된 작업을 삭제하지 않고도 예약된 조정을 일시적으로 비활성화할 수 있습니다. 자세한 내용은 [조정 프로세스의 일시 중지 및 재개 \(p. 142\)](#) 단원을 참조하십시오.
- 휴지 기간은 지원되지 않습니다.
- Amazon EC2 이외의 리소스에 대한 조정 작업을 예약할 수도 있습니다. 자세한 내용은 Application Auto Scaling 사용 설명서에서 [예약된 조정](#)을 참조하십시오.

예약된 작업 생성 및 관리(콘솔)

한 번만 조정하거나 반복 일정으로 조정하는 예약된 작업을 콘솔에서 생성할 수 있습니다. 다음 절차를 완료하여 Auto Scaling 그룹을 조정하기 위한 예약된 작업을 만듭니다.

Auto Scaling 그룹을 위한 예약된 작업을 만들려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 자동 조정 탭의 Scheduled actions(예약된 작업)에서 Create scheduled action(예약된 작업 생성)을 선택합니다. (이전 콘솔: Scheduled Actions(예약된 작업) 탭에서 예약된 작업을 생성할 수 있습니다.)
5. Create scheduled action(예약된 작업 생성) 대화 상자에서 다음을 수행합니다.
 - 이름에서 예약된 작업의 이름을 입력합니다.
 - 다음 값 중 최소 하나를 사용하여 그룹의 크기를 지정합니다. 최소, 최대 또는 원하는 용량.
 - 반복 시간 옵션을 선택합니다. Once(한 번)를 선택하면 지정된 시간에 작업을 수행합니다. Cron을 선택하는 경우, 작업을 수행하는 시기를 지정하는 Cron 식을 UTC로 입력합니다. Every(모든)로 시작하는 옵션을 선택하면 Cron 식이 생성됩니다.
 - 반복에서 Once(한 번)를 선택한 경우, 시작 시간에 작업을 실행할 날짜와 시간을 지정합니다.
 - 반복 작업의 경우 시작 시간과 종료 시간 모두에 값을 지정할 수 있습니다. 시작 시간을 지정하면 작업이 수행되는 가장 빠른 시간이 시작 시간에 해당합니다. 종료 시간을 지정하면 이 시간 이후에는 작업이 반복되지 않습니다.
6. 생성을 선택합니다.

예약된 작업 업데이트

요구 사항이 변경되는 경우, 예약된 작업을 업데이트할 수 있습니다.

예약된 작업을 업데이트하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 자동 조정 탭의 Scheduled actions(예약된 작업)에서 예약된 작업을 선택합니다. (이전 콘솔: Scheduled Actions(예약된 작업) 탭에서 예약된 작업을 선택할 수 있습니다.)
5. [Actions], [Edit]를 선택합니다.
6. Edit scheduled action(예약된 작업 편집) 대화 상자에서 다음을 수행합니다.
 - 최소, 최대 또는 원하는 용량을 사용하여 필요에 따라 그룹의 크기를 업데이트합니다.
 - 필요에 따라 지정된 반복을 업데이트합니다.
 - 필요에 따라 시작 및 종료 시간을 업데이트합니다.
7. Save를 선택합니다.

예약된 작업 삭제

더 이상 필요 없는 예약된 작업은 삭제할 수 있습니다.

예약된 작업을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 자동 조정 탭의 Scheduled actions(예약된 작업)에서 예약된 작업을 선택합니다. (이전 콘솔: Scheduled Actions(예약된 작업) 탭에서 예약된 작업을 선택할 수 있습니다.)
5. [Actions], [Delete]를 선택합니다.
6. 확인 메시지가 나타나면 예, 삭제합니다를 선택합니다.

예약된 작업 생성 및 관리(AWS CLI)

한 번만 조정하거나 `put-scheduled-update-group-action` 명령을 사용하여 반복 일정으로 조정하도록 예약된 작업을 생성 및 업데이트할 수 있습니다.

한 번만 조정하려면

일회성 일정을 지정하여 특정 날짜와 시간(UTC)에 Auto Scaling 그룹을 자동으로 조정할 수 있습니다.

- 지정된 시간에 Auto Scaling 그룹에서 실행 중인 인스턴스의 수를 줄이려면 다음 명령을 사용합니다. `--start-time`에 지정된 날짜와 시간 현재 해당 그룹에 인스턴스가 2개 이상 있는 경우, 그룹은 인스턴스 1개로 조정됩니다.

```
aws autoscaling put-scheduled-update-group-action --scheduled-action-name my-one-time-action \
  --auto-scaling-group-name my-asg --start-time "2019-05-13T08:00:00Z" --desired-capacity 1
```

- 지정된 시간에 Auto Scaling 그룹에서 실행 중인 인스턴스의 수를 늘리려면 다음 명령을 사용합니다. `--start-time`에 지정된 날짜와 시간 현재 해당 그룹에 인스턴스가 3개 미만 있는 경우, 그룹은 인스턴스 3개로 확장됩니다.

```
aws autoscaling put-scheduled-update-group-action --scheduled-action-name my-one-time-action \
  --auto-scaling-group-name my-asg --start-time "2019-05-12T08:00:00Z" --desired-capacity 3
```

반복되는 일정으로 조정하려면

UTC에서 Unix Cron 구문 형식을 사용하여 반복 일정을 지정할 수 있습니다. 이 형식은 다음과 같이 공백으로 구분된 다섯 개의 필드로 구성됩니다. [Minute] [Hour] [Day_of_Month] [Month_of_Year] [Day_of_Week]. 이 형식에 대한 자세한 내용은 [Crontab](#)을 참조하십시오.

다음 `put-scheduled-update-group-action` 명령을 사용하여 매년 1월, 6월, 12월의 첫째 날 00:30에 실행되는 예약된 작업을 만듭니다.

```
aws autoscaling put-scheduled-update-group-action --scheduled-action-name my-recurring-action \
  --auto-scaling-group-name my-asg --recurrence "30 0 1 1,6,12 *" --desired-capacity 3
```

예약 작업 삭제

예약된 작업을 삭제하려면

다음 `delete-scheduled-action` 명령을 사용합니다.

```
aws autoscaling delete-scheduled-action --scheduled-action-name my-recurring-action
```

Amazon EC2 Auto Scaling의 동적 조정

동적 확장을 구성할 때 수요 변화에 대응하여 Auto Scaling 그룹의 용량을 조정하는 방법을 정의합니다.

예를 들어, 현재 인스턴스 2개에서 웹 애플리케이션이 실행되고 있고 사용자가 애플리케이션 로드에서 변경이 있는 경우 Auto Scaling 그룹의 CPU 사용량을 50% 정도로 유지시키려 한다고 가정해 보겠습니다. 이로 인해 과도한 유휴 리소스를 유지하지 않고도 트래픽 급증을 처리할 수 있는 추가 용량을 확보할 수 있습니다.

조정 정책을 생성하여 이러한 요구 사항에 맞게 Auto Scaling 그룹을 동적으로 조정하도록 구성할 수 있습니다. Amazon EC2 Auto Scaling은 그룹을 확장하여(인스턴스를 추가함) 피크 타임에 높은 수요를 처리하고 그룹을 축소하여(인스턴스 수를 줄임) 사용률이 낮은 기간 동안 비용을 절감할 수 있습니다.

목차

- [조정 정책 작동 방식](#) (p. 102)
- [조정 정책 유형](#) (p. 103)
- [여러 조정 정책](#) (p. 103)
- [Amazon EC2 Auto Scaling의 대상 추적 조정 정책](#) (p. 104)
- [Amazon EC2 Auto Scaling에 대한 단계 및 단순 조정 정책](#) (p. 107)
- [Amazon SQS에 따른 조정](#) (p. 115)
- [Auto Scaling 그룹에 대한 조정 활동 확인](#) (p. 119)
- [Auto Scaling 그룹에 대한 조정 정책 비활성화](#) (p. 121)
- [조정 정책 삭제](#) (p. 122)

조정 정책 작동 방식

조정 정책은 Amazon EC2 Auto Scaling에게 특정 CloudWatch 지표를 추적하도록 지시하며, 연결된 CloudWatch 경보가 ALARM(경보) 상태에 있을 때 수행할 작업을 정의합니다. 경보를 트리거하는 데 사용되는 지표는 Auto Scaling 그룹의 모든 인스턴스에서 보내는 지표를 집계한 것입니다. (예를 들어, 하나는 60% CPU이고 다른 하나는 40% CPU인 두 개의 인스턴스로 구성된 Auto Scaling 그룹이 있다고 가정해 보겠습니다. 평균적으로 50% CPU입니다.) 정책이 적용되면 경보가 트리거될 때 Amazon EC2 Auto Scaling가 그룹의 원하는 용량을 늘리거나 줄입니다.

조정 정책이 실행될 때 용량 계산에서 그룹의 최소 및 최대 크기 범위를 벗어나는 숫자를 생성하는 경우 Amazon EC2 Auto Scaling은 새 용량이 최소 및 최대 크기 제한을 벗어나지 않도록 합니다. 용량은 두 가지 방법 중 하나로 측정됩니다. 인스턴스 기준으로 원하는 용량을 설정할 때 선택한 것과 동일한 단위를 사용 또는 용량 유닛을 사용([인스턴스 가중치](#) (p. 74)가 적용된 경우).

- 예제 1: Auto Scaling 그룹 최대 용량 3, 현재 용량 2, 인스턴스 3개를 추가하는 조정 정책. 이 조정 정책을 실행할 때 Amazon EC2 Auto Scaling은 그룹에 인스턴스를 하나만 추가하여 그룹이 최대 크기를 초과하지 않도록 합니다.
- 예제 2: Auto Scaling 그룹 최대 용량 2, 현재 용량 3, 인스턴스 2개를 제거하는 조정 정책. 이 조정 정책을 실행할 때 Amazon EC2 Auto Scaling은 그룹에서 인스턴스를 하나만 제거하여 그룹이 최소 크기보다 작아지지 않도록 합니다.

원하는 용량이 최대 크기 제한에 도달하면 확장이 중지됩니다. 수요가 감소하고 용량이 감소하면 Amazon EC2 Auto Scaling이 다시 확장할 수 있습니다.

그룹에 두 개 이상의 인스턴스 유형이 있고 인스턴스 가중치를 사용하는 경우는 예외입니다. 이 경우 Amazon EC2 Auto Scaling은 최대 크기 제한을 초과하여 확장할 수 있지만 최대 인스턴스 가중치까지만 확장할 수 있습니다. 그 의도는 최대한 새로운 원하는 용량에 근접하면서도 그룹에 지정된 할당 전략을 고수하는 것입니다. 할당 전략에 따라 인스턴스가 생성되는 인스턴스 풀이 결정됩니다. 가중치는 각 인스턴스가 그룹의 용량에 기여하는 용량 유닛 수를 결정합니다. 가중치가 클수록 숫자가 높아집니다.

- 예제 3: Auto Scaling 그룹 최대 용량 12, 현재 용량 10, 용량 유닛 5를 추가하는 조정 정책. 인스턴스 유형에는 1, 4 또는 6의 세 가지 가중치 중 하나가 할당됩니다. 조정 정책을 실행할 때 Amazon EC2 Auto Scaling은 할당 전략에 따라 가중치가 6인 인스턴스 유형을 시작하도록 선택합니다. 이 확장 이벤트의 결과는 원하는 용량이 12이고 현재 용량이 16인 그룹입니다.

조정 정책 유형

Amazon EC2 Auto Scaling은 다음과 같은 조정 정책 유형을 지원합니다.

- 대상 추적 조정—특정 지표의 목표 값을 기준으로 그룹의 현재 용량을 늘리거나 줄입니다. 이 작업은 온도 조절기가 집안 온도를 유지하는 방식과 비슷합니다. 사용자가 온도만 선택하면 나머지는 온도 조절기가 알아서 합니다.
- 단계 조정—그룹의 현재 용량을 일련의 조정 조절에 따라 늘리거나 줄이며 경보 위반의 크기에 따라 달라지는 단계 조절이라고 합니다.
- 단순 조정—그룹의 현재 용량을 단일 조정 조절에 따라 늘리거나 줄입니다.

Auto Scaling 그룹의 인스턴스 수에 비례하여 증가하거나 감소하는 사용률 수치를 기준으로 조정하는 경우 대상 추적 조정 정책을 사용하는 것이 좋습니다. 그렇지 않은 경우 단계 조정 정책을 사용하는 것이 좋습니다.

여러 조정 정책

대부분의 경우, 대상 추적 조정 정책이 충분하여 Auto Scaling 그룹이 자동으로 확장 또는 축소되도록 구성할 수 있습니다. 대상 추적 조정 정책을 사용하면 원하는 결과를 선택하고, 그러한 결과를 얻기 위해 필요에 따라 Auto Scaling 그룹이 인스턴스를 추가 및 제거하도록 할 수 있습니다.

고급 조정 구성의 경우, Auto Scaling 그룹에 하나 이상의 조정 정책이 있을 수 있습니다. 예를 들어, 하나 이상의 대상 추적 조정 정책, 하나 이상의 단계 조정 정책, 또는 둘 모두를 정의할 수 있습니다. 이로 인해 유연성이 늘어나고 여러 시나리오를 처리할 수 있습니다.

여러 개의 정책이 함께 작동하는 방식에 대해 알아보기 위해, Auto Scaling 그룹을 사용하는 애플리케이션과 해당 그룹 EC2 인스턴스에 요청을 전송하는 Amazon SQS 대기열을 고려해 보겠습니다. 애플리케이션에서 최적의 수준으로 작업이 수행되도록 보장하기 위해, Auto Scaling 그룹이 확장되어야 하는 시기를 제어하는 두 개의 정책이 마련되어 있습니다. 하나는 대상 추적 정책으로, 사용자 지정 지표를 사용하여 대기열에 있는 SQS 메시지 수에 따라 용량을 추가 및 제거합니다. 또 다른 하나는 단계 정책으로, 지정된 시간 동안 인스턴스가 90% 사용률을 초과할 때 Amazon CloudWatchCPUUtilization지표를 사용하여 용량을 추가합니다.

동시에 여러 정책이 실행되는 경우, 각 정책이 동시에 Auto Scaling 그룹으로 하여금 확장(또는 축소)하도록 지시할 수 있습니다. 예를 들어, EC2 인스턴스는 CPUUtilization 지표에 대해 CloudWatch 경보를 트리거하고 동시에 SQS 대기열에서 사용자 지정 지표에 대해 경보를 트리거할 수 있습니다.

이러한 상황이 발생하는 경우, Amazon EC2 Auto Scaling은 확장 및 축소를 위한 최대 용량을 제공하는 정책을 선택합니다. 예를 들어 CPU 사용률에 대한 정책에서 1개의 인스턴스를 시작하는 한편, SQS 대기열에 대한 정책에서는 2개의 인스턴스를 시작하는 경우를 생각해 볼 수 있습니다. 만일 두 정책의 확장 조건이 동시에 충족되는 경우, Amazon EC2 Auto Scaling은 SQS 정책에 우선권을 줍니다. 이렇게 하면 Auto Scaling 그룹에서 인스턴스 두 개가 시작됩니다.

가장 큰 용량을 제공하는 정책에 우선 순위를 부여하는 접근법은 다른 확장 기준을 사용하는 정책에도 적용됩니다. 예를 들어 어떤 정책은 인스턴스 세 개를 종료하고, 다른 정책은 인스턴스 수를 25% 줄이며, 축소 시점 현재 해당 그룹에 인스턴스가 여덟 개 있는 경우, Amazon EC2 Auto Scaling 그룹은 그 그룹에 가장 많은 수의 인스턴스를 제공하는 정책에 우선권을 부여합니다. 이로 인해 Auto Scaling 그룹은 인스턴스 두 개를 종료하게 됩니다($8 \times 25\% = 2$). 목표는 Amazon EC2 Auto Scaling에서 인스턴스를 너무 많이 제거하지 않도록 하는 것입니다.

Amazon EC2 Auto Scaling의 대상 추적 조정 정책

대상 추적 조정 정책에 따라 조정 지표를 선택하고 대상 값을 설정합니다. Amazon EC2 Auto Scaling은 조정 정책을 트리거하는 CloudWatch 경보를 생성 및 관리하면서 지표와 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 지표를 목표 값에 가깝게 유지하는 것 외에도 로드 패턴 변화로 인한 지표의 변화에 따라 조정되기도 합니다.

예를 들어, 다음과 같은 경우에 대상 추적 조정을 사용할 수 있습니다.

- 대상 추적 조정 정책을 구성하여 Auto Scaling 그룹의 평균 총 CPU 사용량을 40%로 유지하는 경우
- 대상 추적 조정 정책을 구성하여 Elastic Load Balancing 대상 그룹의 대상 1개당 요청 수를 Auto Scaling 그룹에 필요한 1000개로 유지하는 경우

Amazon EC2 인스턴스 측정 항목을 1분 주기로 조정하는 것이 좋습니다. 이 경우 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. Amazon EC2 인스턴스는 기본 모니터링, 즉 5분 주기로 인스턴스의 측정치 데이터를 사용하도록 기본적으로 활성화되어 있습니다. 하지만 세부 모니터링을 활성화하여 인스턴스의 측정치 데이터를 가져오는 주기를 1분으로 바꿀 수 있습니다. 자세한 내용은 [Auto Scaling 인스턴스에 대한 모니터링 구성 \(p. 163\)](#) 단원을 참조하십시오.

고려 사항

다음 사항에 유의하십시오.

- 대상 추적 조정 정책은 지정한 측정치가 목표 값을 초과할 때 한해서 Auto Scaling 그룹을 확장하도록 되어 있습니다. 대상 추적 조정 정책에서는 지정한 측정치가 목표 값보다 작을 때 Auto Scaling 그룹을 확장할 수 없습니다.
- 대상 값과 실제 지표 데이터 포인트 사이에는 차이가 발생할 수 있습니다. 추가하거나 제거할 인스턴스 수를 결정할 때마다 항상 반올림 또는 내림을 통해 어림짐작으로 동작하기 때문입니다. 이는 인스턴스를 부족하게 추가하거나 너무 많이 제거하는 일을 방지하기 위해서입니다. 하지만 인스턴스가 줄어서 Auto Scaling 그룹이 작아지는 경우에는 그룹의 사용량이 목표 값에서 멀어질 수도 있습니다. 예를 들어 CPU 사용률 목표값을 50%로 설정한 후 Auto Scaling 그룹이 목표값을 초과한다고 가정해 보겠습니다. 1.5개의 인스턴스를 추가하면 CPU 활용률이 50% 가까이 감소할 것을 알 수 있습니다. 하지만 1.5개의 인스턴스를 추가할 수 없기 때문에 반올림을 통해 인스턴스 2개를 추가합니다. 그러면 CPU 사용률이 50% 아래로 떨어지는 동시에 애플리케이션은 리소스를 충분히 확보하게 됩니다. 마찬가지로 인스턴스를 1.5개 제거하면 CPU 사용률이 증가하여 50%를 상회한다고 판단할 경우에는 인스턴스를 1개만 제거합니다.
- 인스턴스가 더 많고 규모가 큰 Auto Scaling 그룹의 경우, 활용도가 더 많은 수의 인스턴스에 분산되므로 인스턴스를 추가 또는 제거하면 대상 값과 실제 측정치 데이터 포인트 간 차이가 줄어 듭니다.
- 애플리케이션 가용성을 보장하기 위해, Auto Scaling 그룹은 가능한 한 빨리 측정치에 비례하여 확장되고 더 서서히 축소됩니다.
- Auto Scaling 그룹은 여러 조정 정책을 동시에 적용할 수 있습니다. 자세한 내용은 [여러 조정 정책 \(p. 103\)](#) 단원을 참조하십시오.
- 각각 다른 측정치를 사용한다는 전제 하에 다수의 대상 추적 조정 정책을 Auto Scaling 그룹에 구성할 수 있습니다. Amazon EC2 Auto Scaling은 항상 가용성을 우선시하므로, 대상 추적 정책이 확장 또는 축소를 허용하는지에 따라 그 동작이 달라 집니다. 대상 추적 정책 중 하나라도 Auto Scaling 그룹 확장을 허용할

경우 확장 가능한 대상을 확장하지만, 모든 대상 추적 정책(축소 부분이 활성화됨)이 축소를 허용하는 경우에만 대상을 축소합니다.

- 대상 추적 조정 정책에서 축소 부분을 비활성화할 수 있습니다. 이 기능은 여러 방법을 사용하여 Auto Scaling 그룹을 확장할 수 있는 유연성을 제공합니다. 예를 들어 축소를 위한 대상 추적 조정 정책을 사용하는 동시에 다른 조정 정책 유형을 사용할 수 있습니다.
- 대상 추적 조정 정책을 위해 구성된 CloudWatch 경보는 편집하거나 삭제하지 마십시오. 대상 추적 조정 정책을 삭제하면 해당 조정 정책과 연결된 CloudWatch 경보가 자동으로 삭제됩니다.
- Amazon EC2 이외의 리소스에 대한 대상 추적 조정 기능을 사용할 수도 있습니다. 자세한 내용은 Application Auto Scaling 사용 설명서의 [Target Tracking Scaling Policies\(대상 추적 조정 정책\)](#)을 참조하십시오.

지표 사용

Amazon EC2 콘솔을 사용하여 미리 정의된 지표를 기반으로 대상 추적 조정 정책을 적용할 수 있습니다. 대신 Amazon EC2 Auto Scaling CLI 또는 API를 사용하여 사전 정의된 또는 사용자 지정 지표를 기반으로 대상 추적 조정 정책을 적용할 수 있습니다. 다음과 같은 사전 정의 지표를 사용할 수 있습니다.

- Auto Scaling 그룹의 `ASGAverageCPUUtilization`—평균 CPU 활용도
- `ASGAverageNetworkIn`—Auto Scaling 그룹에 의해 모든 네트워크 인터페이스에서 받은 평균 바이트 수.
- `ASGAverageNetworkOut`—Auto Scaling 그룹에 의해 모든 네트워크 인터페이스에서 보낸 평균 바이트 수.
- `ALBRequestCountPerTarget`—Application Load Balancer 대상 그룹에서 대상별로 이행된 요청 수.

사용자 지정된 지표를 지정하여 사용 가능한 다른 Amazon CloudWatch 측정치를 선택할 수 있습니다.

지표를 선택하는 경우, 다음 사항에 유의하십시오.

- 모든 지표를 대상 추적에 사용할 수 있는 것은 아닙니다. 사용자 지정된 지표를 지정할 때 이 방법이 중요할 수 있습니다. 측정치는 유효한 사용량 수치로서 인스턴스의 사용량을 설명해야 합니다. 지표 값은 Auto Scaling 그룹의 인스턴스 수에 비례하여 증가하거나 감소합니다. 즉, 지표 데이터를 사용하여 인스턴스 수에 따라 비례적으로 확장하거나 축소할 수 있습니다. 예를 들어 Auto Scaling 그룹의 데이터 로드가 인스턴스로 분산되는 경우에는 Auto Scaling 그룹의 CPU 사용량(즉 지표 차원 `AutoScalingGroupName`의 Amazon EC2 지표 `CPUUtilization`)이 유효합니다.
- 다음 지표는 대상 추적에 사용할 수 없습니다.
 - Auto Scaling 그룹에 대한 로드 밸런서가 수신하는 요청 수(`Elastic Load Balancing` 지표 `RequestCount`). 로드 밸런서가 수신하는 요청 수는 Auto Scaling 그룹의 사용량에 따라 변경되지 않습니다.
 - 로드 밸런서 요청 지연 시간(`Elastic Load Balancing` 지표 `Latency`). 요청 지연 시간은 사용량 증가에 따라 늘어날 수는 있지만 반드시 비례하지는 않습니다.
 - CloudWatch SQS 대기열 지표 `ApproximateNumberOfMessagesVisible`. 대기열의 메시지 수는 대기열의 메시지를 처리하는 Auto Scaling 그룹의 크기에 비례하여 변경되지 않을 수 있습니다. 하지만 Auto Scaling 그룹의 EC2 인스턴스당 대기열의 메시지 수를 측정하는 사용자 지정 지표는 작동합니다. 자세한 내용은 [Amazon SQS에 따른 조정 \(p. 115\)](#) 단원을 참조하십시오.
- `ALBRequestCountPerTarget` 지표를 사용하지 않으면 지정된 지표의 데이터가 충분하지 않은 경우, 대상 추적 조정 정책이 Auto Scaling 그룹을 축소하지 않습니다. 그러한 이유는 `ALBRequestCountPerTarget` 지표가 관련 데이터가 없는 기간 동안 0을 내보내고, 대상 추적 정책이 지표 데이터로 하여금 활용도가 적은 추세를 해석하도록 요구하기 때문입니다. 대상 그룹에 요청이 라우팅되지 않은 경우 Auto Scaling 그룹을 인스턴스 0개로 축소하려면, 그룹의 최소 용량을 0으로 설정해야 합니다.

인스턴스 워업

새로 시작된 인스턴스가 워업되는 데 걸리는 시간(초)을 지정할 수 있습니다. 워업 시간이 만료될 때까지 인스턴스는 Auto Scaling 그룹의 집계된 측정치에 포함되지 않습니다.

확장 중에는 워업 중인 인스턴스를 그룹의 현재 용량의 부분으로 간주하지 않습니다. 이를 통해 필요 이상으로 인스턴스를 추가하지 않게 됩니다.

축소 중에는 종료 중인 인스턴스를 그룹의 현재 용량의 부분으로 간주하지 않습니다. 따라서 Auto Scaling 그룹에서 인스턴스를 필요 이상으로 빼지 않습니다.

확장 활동이 진행 중인 동안에는 축소 활동을 시작할 수 없습니다.

Auto Scaling 그룹에 대한 대상 추적 조정 정책 생성(콘솔)

대상 추적 조정 정책을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택하고 최소 및 최대 크기 제한이 적절하게 설정되어 있는지 확인합니다. 예를 들어 그룹이 이미 최대 크기인 경우 크기를 축소하려면 새 최대값을 지정합니다. Amazon EC2 Auto Scaling은 최소 용량 이하 또는 최대 용량 이상으로 그룹을 확장하지 않습니다. 그룹을 업데이트하려면 편집을 선택하고 최소 및 최대 크기를 지정한 다음 저장을 선택합니다.
4. 자동 조정 탭의 조정 정책에서 정책 추가를 선택합니다. (이전 콘솔: 조정 정책 탭에서 정책을 추가할 수 있습니다.)
5. 정책을 정의하려면 다음을 수행합니다.
 - a. 정책 유형의 경우 기본적으로 Target tracking scaling(대상 추적 조정)을 그대로 둡니다.
 - b. 정책의 이름을 지정합니다.
 - c. Metric type(지표 유형)에서 지표를 선택합니다.
 - d. 측정치의 [Target value]를 지정합니다.
 - e. (선택 사항) Instances need(인스턴스 필요)에 인스턴스 워업 값을 지정합니다. 이렇게 하면 새로 시작된 인스턴스가 CloudWatch 지표에 반영될 때까지의 시간을 제어할 수 있습니다.
 - f. (선택 사항) Disable scale in to create only a scale-out policy(축소 정책을 비활성화하여 확장 정책만 생성)을 선택합니다. 이렇게 하면 원할 경우 유형이 다를 때마다 축소 정책을 별도로 생성할 수 있습니다.
6. 생성을 선택합니다.

대상 추적 조정 정책 생성(AWS CLI)

다음과 같이 AWS CLI를 사용하여 Auto Scaling 그룹의 대상 추적 조정 정책을 구성합니다.

작업

- 1단계: Auto Scaling 그룹 생성 (p. 106)
- 2단계: 대상 추적 조정 정책 생성 (p. 107)

1단계: Auto Scaling 그룹 생성

`create-auto-scaling-group` 명령을 통해 시작 구성 `my-launch-config`를 사용하여 `my-asg`라는 Auto Scaling 그룹을 만듭니다. 사용하려는 시작 구성이 없으면 구성을 생성할 수 있습니다. 자세한 내용은 [create-launch-configuration](#) 단원을 참조하십시오.

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg \
```

```
--launch-configuration-name my-launch-config \  
--vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782" \  
--max-size 5 --min-size 1
```

2단계: 대상 추적 조정 정책 생성

애플리케이션의 부하가 달라지면 Auto Scaling 그룹에서 실행 중인 EC2 인스턴스 수를 늘리거나 줄이라고 이 그룹에 동적으로 알려 주는 대상 추적 조정 정책을 생성할 수 있습니다.

예제: 대상 추적 구성 파일

다음은 config.json으로 저장해야 하는 대상 추적 구성 파일의 예제입니다.

```
{  
  "TargetValue": 40.0,  
  "PredefinedMetricSpecification":  
    {  
      "PredefinedMetricType": "ASGAverageCPUUtilization"  
    }  
}
```

또는 사용자 지정 지표를 생성하고 CloudWatch의 각 파라미터에 대한 값을 추가하여 조정에서 사용되는 지표를 사용자 지정할 수 있습니다. 다음은 지정된 지표의 평균 사용률을 40%로 유지하는 대상 추적 구성의 예입니다.

```
{  
  "TargetValue":40.0,  
  "CustomizedMetricSpecification":{  
    "MetricName":"MyUtilizationMetric",  
    "Namespace":"MyNamespace",  
    "Dimensions":[  
      {  
        "Name":"MyOptionalMetricDimensionName",  
        "Value":"MyOptionalMetricDimensionValue"  
      }  
    ],  
    "Statistic":"Average",  
    "Unit":"Percent"  
  }  
}
```

예: cpu40-target-tracking-scaling-policy

앞에서 생성한 config.json 파일과 함께 [put-scaling-policy](#) 명령을 사용하여 cpu40-target-tracking-scaling-policy이라는 이름으로 Auto Scaling 그룹의 평균 CPU 사용량을 40%로 유지하는 조정 정책을 만듭니다.

```
aws autoscaling put-scaling-policy --policy-name cpu40-target-tracking-scaling-policy \  
--auto-scaling-group-name my-asg --policy-type TargetTrackingScaling \  
--target-tracking-configuration file://config.json
```

Amazon EC2 Auto Scaling에 대한 단계 및 단순 조정 정책

단계 조정 및 단순 조정에서는, 조정 프로세스를 트리거하는 CloudWatch 경보에 대한 조정 지표와 임계값을 선택합니다. 지정된 평가 기간 동안 임계값을 위반했을 때 Auto Scaling 그룹을 어떻게 조정할지 정의해야 합니다.

대상 추적 조정 정책을 사용하여 Application Load Balancer의 RequestCountPerTarget 지표 또는 평균 CPU 사용률 같은 지표에 따라 확장하는 것이 좋습니다. 용량이 증가할 때 감소하고 용량이 감소할 때 증가하는 지표를 사용하여 비례적으로 확장하거나 대상 추적을 사용하여 인스턴스 수를 늘릴 수 있습니다. 이를 통해 Amazon EC2 Auto Scaling이 애플리케이션의 수요 곡선을 근접하여 추종할 수 있습니다. 자세한 내용은 [대상 추적 조정 정책 \(p. 104\)](#) 단원을 참조하십시오. 단계 조정을 고급 구성에 대한 추가 정책으로 사용할 수도 있습니다. 예를 들어, 수요가 일정 수준에 도달할 때 보다 적극적인 응답을 구성할 수 있습니다.

단계 조정 정책과 단순 조정 정책의 차이점

단계 조정 정책과 단순 조정 정책은 사용할 수 있는 두 가지 동적 조정 옵션입니다. 두 옵션 모두 조정 정책에 대한 CloudWatch 경보를 생성해야 합니다. 또한 두 옵션 모두 경보에 대한 높은 임계값과 낮은 임계값을 지정해야 합니다. 두 가지 모두 인스턴스를 추가 또는 제거할지 여부와 몇 개를 추가 또는 제거할지를 정의하거나 그룹을 정확한 크기로 설정해야 합니다.

두 정책 유형 간의 주요 차이점은 단계 조정 정책을 통해 얻을 수 있는 단계 조절입니다. 단계 조절이 적용되고 Auto Scaling 그룹의 현재 용량을 늘리거나 줄이는 경우 조절은 경보 위반의 크기에 따라 달라집니다.

대부분의 경우에는 단계 조정 정책이 단순 조정 정책보다 더 낫습니다(단일 단계 조절만 있는 경우에도).

단순 조정에서 기본 문제는 조정 활동이 시작된 후 정책은 조정 활동 또는 상태 확인 대체가 완료되고 휴지 기간이 만료될 때까지 기다린 후 추가 경보에 응답해야 한다는 것입니다. 휴지 기간을 지정하여 이전 활동의 효과가 표시되기 전에 에서 추가 조정 활동을 시작하지 않도록 할 수 있습니다.

반면 단계 조정에서는 조정 활동 또는 상태 확인 대체가 진행 중인 동안에도 정책이 추가 경보에 계속 응답할 수 있습니다. 따라서 위반된 모든 경보는 경보 메시지 수신 시 Amazon EC2 Auto Scaling에 의해 평가됩니다.

Amazon EC2 Auto Scaling은 초기에 단순 조정 정책만 지원했습니다. 그래서 대상 추적 및 단계 정책을 도입하기 전에는 조정 정책을 생성하면 정책이 단순 조정 정책으로 처리되었습니다.

단계 조절

단계 조정 정책을 생성할 때 경보 위반의 크기에 따라 인스턴스 수를 동적으로 자동 조정되도록 하나 이상의 단계 조절을 지정합니다. 각 단계 조절은 다음을 지정합니다.

- 지표 값의 하한값입니다.
- 지표 값의 상한값입니다.
- 조정 유형을 기반으로 축소하거나 확장하는 양입니다.

CloudWatch는 CloudWatch 경보와 연결된 지표의 통계를 기반으로 지표 데이터 포인트를 집계합니다. 경보가 위반되면 적절한 조정 정책이 트리거됩니다. Amazon EC2 Auto Scaling은 지정된 집계 유형을 CloudWatch가 제공하는 가장 최근의 지표 데이터 포인트에 적용합니다(원시 지표 데이터가 아님). 이 집계된 지표 값을 단계 조절로 정의된 상한값 및 하한값과 비교하여 어느 단계의 조절을 수행할 것인지 결정합니다.

위반 임계값과 연계하여 상한값과 하한값을 지정합니다. 예를 들어 현재 용량과 원하는 용량이 모두 10인 Auto Scaling 그룹이 있다고 가정해 보겠습니다. 위반 임계값이 50%이고 스케일 아웃 및 스케일 인 정책이 있는 CloudWatch 경보가 발행됩니다. 각 정책에 대해 조절 유형이 PercentChangeInCapacity(또는 콘솔의 그룹 백분율)인 단계 조절 집합이 있습니다.

예제: 스케일 아웃 정책에 대한 단계 조절

하한	상한	조절
0	10	0
10	20	10
20	null	30

하한	상한	조절
-10	0	0
-20	-10	-10
null	-20	-30

Metric value					
-infinity	30%	40%	60%	70%	infinity

-30%		-10% Unchanged		+10%	+30%

- 집계된 지표 값이 40보다 크고 60보다 작으면 현재 용량과 원하는 용량이 유지됩니다.
- 측정치 값이 60이 되면 확장 정책의 2단계 조절(10개의 인스턴스에 10% 합산)에 따라 그룹의 원하는 용량을 1개의 인스턴스만큼 늘려 11개의 인스턴스로 만듭니다. 새 인스턴스가 실행되고 지정된 워업 시간이 만료된 후 그룹의 현재 용량을 11개의 인스턴스로 늘립니다. 용량이 이렇게 증가한 후에도 지표 값이 70으로 증가하면 그룹의 원하는 용량이 14개의 인스턴스까지 증가합니다(인스턴스가 3개씩 증가). 이는 스케일 아웃 정책의 세 번째 단계 조절을 기반으로 합니다(11개 인스턴스의 30%인 3.3개 인스턴스가 3개로 반올림되어 추가).
- 측정치 값이 40이 될 경우 축소 정책의 2단계 조절에 따라(14개의 인스턴스에 10%인 1.4개의 인스턴스를 빼고 1개의 인스턴스로 반내림) 그룹의 원하는 용량을 1개의 인스턴스만큼 더 줄여 13개의 인스턴스로 만듭니다. 이렇게 용량이 감소한 후에도 지표 값이 30으로 떨어지면 그룹의 원하는 용량이 10개의 인스턴스로 감소합니다(인스턴스가 3개씩 감소). 이는 스케일 인 정책의 세 번째 단계 조절을 기반으로 합니다(13개 인스턴스 30%인 3.9개 인스턴스가 3개로 반내림되어 제거).

- AWS Management 콘솔을 사용하는 경우 상한값과 하한값을 절대값으로 지정합니다. API 또는 CLI를 사용하는 경우 위반 임계값에 상관 관계가 있는 상한값과 하한값을 지정합니다.
- 단계 조절의 범위는 중복되거나 격차가 있어서는 안 됩니다.
- 1단계 조절에만 null 하한값(negative infinity)이 포함될 수 있습니다. 1단계 조절에 음의 하한값이 포함될 경우 null 하한값으로 단계 조절을 해야 합니다.
- 1단계 조절에만 null 상한값(positive infinity)이 포함될 수 있습니다. 1단계 조절에 양의 상한값이 포함될 경우 null 상한값으로 단계 조절을 해야 합니다.
- 상한 및 하한값은 동일한 단계 조절에서 null이 될 수 없습니다.
- 측정치 값이 위반 임계값을 초과할 경우 하한값은 포함되고 상한값은 제외됩니다. 측정치 값이 위반 임계값 미만일 경우 하한값은 제외되고 상한값은 포함됩니다.

선택한 조정 조절 유형에 따라 최적의 조정 작업을 수행하는 조정 정책을 정의할 수 있습니다. 조절 유형을 Auto Scaling 그룹의 현재 용량의 백분율이나 용량 단위로 지정할 수 있습니다. 일반적으로 인스턴스 가중치 부여 기능을 사용하지 않는 한 용량 단위는 하나의 인스턴스를 의미합니다.

109

- **ChangeInCapacity** — 그룹의 현재 용량을 지정된 값만큼 늘리거나 줄입니다. 양의 값은 용량을 늘리고, 음의 조절 값은 용량을 줄입니다. 예제: 그룹의 현재 용량이 3개의 인스턴스이고 조절이 5개인 경우 이 정책이 수행되면 용량 단위가 총 8개가 되도록 용량에 5개의 용량 단위를 추가합니다.
- **ExactCapacity** — 그룹의 현재 용량을 지정된 값으로 변경합니다. 이 조절 유형에는 양의 값을 지정합니다. 예제: 그룹의 현재 용량이 3개의 인스턴스이고 조절이 5개인 경우 이 정책이 수행되면 용량을 5개의 용량 단위로 변경합니다.
- **PercentChangeInCapacity** — 그룹의 현재 용량을 지정된 퍼센트만큼 늘리거나 줄입니다. 양의 값은 용량을 늘리고, 음의 값은 용량을 줄입니다. 예제: 현재 용량이 10개이고 조절이 10%인 경우 이 정책이 수행되면 용량 단위가 총 11개가 되도록 용량에 1개의 용량 단위를 추가합니다.

Note

결과 값이 정수가 아닌 경우 다음과 같이 반올림(반내림)합니다.

- 1보다 큰 값은 반내림합니다. 예를 들어, 12.7은 12로 반내림합니다.
- 0과 1 사이의 값은 1로 반올림합니다. 예를 들어, .67은 1로 반올림합니다.
- 0과 -1 사이의 값은 1로 반내림합니다. 예를 들어, -.58은 -1로 반내림합니다.
- -1보다 작은 값은 반올림합니다. 예를 들어, -6.67은 -6로 반내림합니다.

PercentChangeInCapacity에서는 **MinAdjustmentMagnitude** 파라미터를 사용하여 조정할 최소 인스턴스 수를 지정할 수도 있습니다. 예를 들어 25퍼센트를 추가하는 정책을 생성하고 최소 증분으로 2개의 인스턴스를 지정한다고 가정해 보십시오. Auto Scaling 그룹에 4개 인스턴스가 있고 조정 정책이 실행되는 경우 4개 인스턴스의 25퍼센트는 1개 인스턴스입니다. 그러나 최소 증분을 2로 지정했기 때문에 2개의 인스턴스가 추가됩니다.

[인스턴스 가중치 \(p. 74\)](#)를 사용하면 **MinAdjustmentMagnitude** 파라미터를 0이 아닌 값으로 설정하는 효과가 변경됩니다. 값은 용량 유닛입니다. 조정할 최소 인스턴스 수를 설정하려면 이 파라미터를 최대 인스턴스 가중치 이상의 값으로 설정합니다.

인스턴스 가중치 부여를 사용하는 경우 Auto Scaling 그룹의 현재 용량이 필요에 따라 원하는 용량을 초과할 수 있다는 점에 유의하십시오. 절대 감소 수 또는 백분율에서 감소하는 양이 현재 용량과 원하는 용량 간의 차이보다 작으면 조정 작업이 수행되지 않습니다. 경보가 트리거되면 조정 정책의 결과를 확인할 때 이러한 동작을 고려해야 합니다. 예를 들어, 원하는 용량이 30개이고 현재 용량이 32개라고 가정해 보겠습니다. 경보가 트리거되면 조정 정책이 원하는 용량을 1씩 줄이게 되고 조정 작업이 수행되지 않습니다.

인스턴스 워업

단계 정책을 생성하는 경우 새로 시작된 인스턴스가 워업되는 데 걸리는 시간(초)을 지정할 수 있습니다. 워업 시간이 만료될 때까지 인스턴스는 Auto Scaling 그룹의 집계된 측정치에 포함되지 않습니다.

단계 조정 단원의 예제를 사용해 새 인스턴스가 여전히 워밍업되는 동안 지표가 60개가 되고 그 이후에 62개로 증가한다고 가정해 보십시오. 현재 용량은 아직 인스턴스 10개이므로 인스턴스 1개(10개 인스턴스의 10 퍼센트)가 추가됩니다. 하지만 그룹의 필요한 용량이 이미 11개 인스턴스이므로 조정 정책이 원하는 용량을 더 이상 늘리지 않습니다. 지표가 70이 되고 새 인스턴스가 여전히 워업 중이면 3개의 인스턴스(10개 인스턴스의 30퍼센트)를 추가해야 합니다. 하지만 그룹의 필요한 용량이 이미 11이므로 새로운 필요 용량 13개 인스턴스에 대해 2개의 인스턴스만 추가합니다.

확장 중에는 워업 중인 인스턴스를 그룹의 현재 용량의 부분으로 간주하지 않습니다. 따라서 여러 개의 경보 위반이 동일한 단계 조절 범위에 해당하는 경우 단일 조정 활동이 이루어집니다. 이를 통해 필요 이상으로 인스턴스를 추가하지 않게 됩니다.

축소 중에는 종료 중인 인스턴스를 그룹의 현재 용량의 부분으로 간주하지 않습니다. 따라서 Auto Scaling 그룹에서 인스턴스를 필요 이상으로 빼지 않습니다.

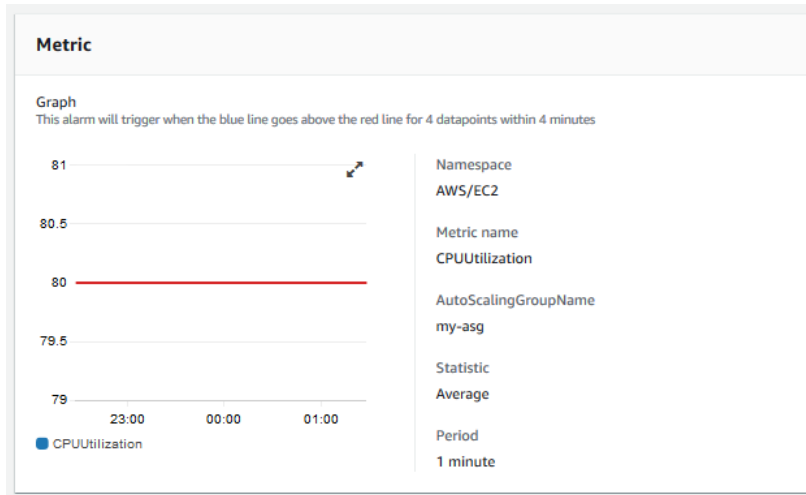
확장 활동이 진행 중인 동안에는 축소 활동을 시작할 수 없습니다.

Auto Scaling 그룹에 대한 단계 조정 정책 생성(콘솔)

다음 절차는 Amazon EC2 Auto Scaling 콘솔을 사용하여 단계 조정 정책 두 개를 생성하는 방법을 보여줍니다. 하나는 그룹의 용량을 30%까지 늘리는 확장 정책이고, 다른 하나는 그룹의 용량을 2개 인스턴스로 줄이는 축소 정책입니다.

단계 조정 정책에서는 CloudWatch 경보를 사용하여 Auto Scaling 그룹에서 스케일 아웃 또는 스케일 인이 필요한 시점을 결정합니다. 각 CloudWatch 경보는 한 가지 지표를 주시하여 해당 지표가 정책에 지정한 임계값을 벗어날 경우 Amazon EC2 Auto Scaling에 메시지를 보냅니다. 경보를 사용하여 AWS에서 사용하는 서비스에서 CloudWatch로 전송하는 지표를 모니터링할 수 있습니다. 또는 자체 지표를 생성하고 모니터링할 수 있습니다. CloudWatch 경보를 생성할 때 경보 변경 시 이메일 알림을 보내기 위한 Amazon SNS 주제를 지정할 수 있습니다.

예: 평균 CPU 사용률 지표를 모니터링하는 CloudWatch 경보



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, Auto Scaling 그룹별을 선택하고 그룹에 대한 CPU Utilization 지표를 선택한 다음 측정치 선택을 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 이름(예: Step-Scaling-AlarmHigh-AddCapacity) 및 설명(예: ASG CPU above 80%)을 입력하고 다음을 선택합니다.
14. 경보 생성을 선택합니다.

확장을 위한 단계 조정 정책을 생성하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택하고 최소 및 최대 크기 제한이 적절하게 설정되어 있는지 확인합니다. 예를 들어 그룹이 이미 최대 크기인 경우 확장하려면 새 최대값을 지정해야 합니다. Amazon EC2 Auto Scaling은 최소 용량 이하 또는 최대 용량 이상으로 그룹을 조정하지 않습니다. 그룹을 업데이트하려면 편집을 선택하고 최소 및 최대 크기를 지정한 다음 저장을 선택합니다.
4. 자동 조정 탭의 조정 정책에서 정책 추가를 선택합니다. (이전 콘솔: 조정 정책 탭에서 정책을 추가할 수 있습니다.)
5. 확장(용량 증가)을 위한 정책을 정의하려면 다음을 수행합니다.
 - a. 정책 유형에서 단계 조정을 선택합니다. (이전 콘솔: Create a scaling policy with steps(단계를 사용하여 조정 정책 생성)를 선택합니다.)
 - b. 정책의 이름을 지정합니다.
 - c. Execute policy when(정책 실행 시기)에서 경보를 선택합니다. 아직 경보를 생성하지 않았다면 지금 생성합니다. 경보 임계값을 80% 이상으로 설정합니다.
 - d. 다음 작업을 수행을 사용하여 이 정책을 실행할 때 적용할 현재 그룹 크기 변경을 지정합니다. 특정 인스턴스 개수 또는 기존 그룹 크기의 백분율을 추가하거나 그룹을 정확한 크기로 설정할 수 있습니다.

예를 들어 Add를 선택하고 다음 필드에 30을 입력한 다음 percent of group을 선택합니다. 기본적으로 이 단계 조정의 하한값은 경보 임계값이고 상한값은 양의 무한대입니다.
 - e. 다른 단계를 추가하려면 Add step(단계 추가)를 선택한 다음 조정할 양과 경보 임계값에 상대적인 단계 하한 및 상한을 정의합니다.
 - f. 조정할 최소 인스턴스 수를 설정하려면 Add capacity units in increments of at least 1 capacity units의 숫자 필드를 업데이트합니다.
 - g. 인스턴스에 필요에서 인스턴스 워밍업 값을 지정합니다. 여기에서 새롭게 시작된 인스턴스가 CloudWatch 측정치에 기여하는 시간을 제어할 수 있습니다.
6. Create를 선택합니다.

축소를 위한 단계 조정 정책을 생성하려면

1. 정책 추가를 선택하여 확장 정책을 생성한 후 중단한 지점부터 계속합니다.
2. 축소(용량 감소)를 위한 정책을 정의하려면 다음을 수행합니다.
 - a. 정책 유형에서 단계 조정을 선택합니다. (이전 콘솔: Create a scaling policy with steps(단계를 사용하여 조정 정책 생성)를 선택합니다.)
 - b. 정책의 이름을 지정합니다.

- c. Execute policy when(정책 실행 시기)에서 경보를 선택합니다. 아직 경보를 생성하지 않았다면 지금 생성합니다. 경보 임계값을 40% 이하로 설정합니다.
- d. 다음 작업을 수행을 사용하여 이 정책을 실행할 때 적용할 현재 그룹 크기 변경을 지정합니다. 특정 인스턴스 개수 또는 기존 그룹 크기의 백분율을 제거하거나 그룹을 정확한 크기로 설정할 수 있습니다.

예를 들어 Remove를 선택하고 다음 필드에 2을 입력한 다음 capacity units을 선택합니다. 기본적으로 이 단계 조정의 상한값은 경보 임계값이고 하한값은 음의 무한대입니다.

- e. 다른 단계를 추가하려면 Add step(단계 추가)를 선택한 다음 조정할 양과 경보 임계값에 상대적인 단계 하한 및 상한을 정의합니다.

3. 생성을 선택합니다.

단계적 조정 정책 구성(AWS CLI)

다음과 같이 AWS CLI를 사용하여 Auto Scaling 그룹의 단계 조정 정책을 구성합니다.

작업

- 1단계: Auto Scaling 그룹 생성 (p. 113)
- 2단계: 조정 정책 생성 (p. 113)
- 3단계: CloudWatch 경보 생성 (p. 115)

1단계: Auto Scaling 그룹 생성

다음 `create-auto-scaling-group` 명령을 사용하여 시작 구성 `my-launch-config`로 `my-asg`라는 Auto Scaling 그룹을 만듭니다. 사용하려는 시작 구성이 없으면 구성을 생성할 수 있습니다. 자세한 내용은 [create-launch-configuration](#) 단원을 참조하십시오.

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg \
--launch-configuration-name my-launch-config \
--vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782" \
--max-size 5 --min-size 1
```

2단계: 조정 정책 생성

애플리케이션의 부하가 달라질 때 수행할 작업을 Auto Scaling에 알려주는 단계 조정 정책 또는 단순 조정 정책을 생성할 수 있습니다.

예: `my-step-scale-out-policy`

다음 `put-scaling-policy` 명령을 사용하여 다음과 같은 단계 조정에 따라 그룹 용량을 늘리는 `my-step-scale-out-policy`라는 단계 조정 정책(조절 유형 `PercentChangeInCapacity`)을 생성합니다 (CloudWatch 경보 임계값을 60%로 가정).

- 지표 값이 70% 이상이고 80% 미만인 경우 인스턴스 수를 10% 늘립니다.
- 지표 값이 80% 이상이고 90% 미만인 경우 인스턴스 수를 20% 늘립니다.
- 지표 값이 90% 이상이면 인스턴스 수를 30% 증가시킵니다.

```
aws autoscaling put-scaling-policy \
--auto-scaling-group-name my-asg \
--policy-name my-step-scale-out-policy \
--policy-type StepScaling \
--adjustment-type PercentChangeInCapacity \
--metric-aggregation-type Average \
```



```
--step-adjustments  
MetricIntervalLowerBound=10.0,MetricIntervalUpperBound=20.0,ScalingAdjustment=10 \  
  
MetricIntervalLowerBound=20.0,MetricIntervalUpperBound=30.0,ScalingAdjustment=20 \  
MetricIntervalLowerBound=30.0,ScalingAdjustment=30 \  
--min-adjustment-magnitude 1
```

출력에 정책에 대한 ARN이 포함됩니다. 이 ARN을 안전한 위치에 저장합니다. CloudWatch 경보를 생성할 때 필요합니다.

```
{  
  "PolicyARN": "arn:aws:autoscaling:region:123456789012:scalingPolicy:4ee9e543-86b5-4121-b53b-aa4c23b5bbcc:autoScalingGroupName/my-asg:policyName/my-step-scale-in-policy"  
}
```

예: my-step-scale-in-policy

다음 `put-scaling-policy` 명령을 사용하여 그룹 용량을 인스턴스 2개씩 줄이는 `ChangeInCapacity` 조절 유형으로 my-step-scale-in-policy라는 단계 조정 정책을 생성합니다.

```
aws autoscaling put-scaling-policy \  
  --auto-scaling-group-name my-asg \  
  --policy-name my-step-scale-in-policy \  
  --policy-type StepScaling \  
  --adjustment-type ChangeInCapacity \  
  --step-adjustments MetricIntervalUpperBound=0.0,ScalingAdjustment=-2
```

출력에는 해당 정책의 고유 이름 역할을 하는 ARN이 포함됩니다. 나중에 ARN 또는 해당 정책 이름과 그룹 이름의 조합을 사용하여 정책을 지정할 수 있습니다. 이 ARN을 안전한 위치에 저장합니다. CloudWatch 경보를 생성할 때 필요합니다.

```
{  
  "PolicyARN": "arn:aws:autoscaling:region:123456789012:scalingPolicy:ac542982-cbeb-4294-891c-a5a941dfa787:autoScalingGroupName/my-asg:policyName/my-step-scale-out-policy"  
}
```

단순 조정 정책의 예

또는 이전 CLI 명령 대신 다음 CLI 명령을 사용하여 단순 조정 정책을 생성할 수 있습니다. 이러한 명령의 출력에는 CloudWatch 경보를 생성해야 하는 각 정책에 대한 ARN이 포함됩니다. 간단 조정 정책을 사용하기 때문에 휴지 기간이 적용됩니다.

예: my-simple-scale-out-policy

다음 `put-scaling-policy` 명령을 사용하여 그룹 용량을 30% 늘리는 `PercentChangeInCapacity` 조절 유형으로 my-simple-scale-out-policy라는 단순 조정 정책을 생성합니다.

```
aws autoscaling put-scaling-policy --policy-name my-simple-scale-out-policy \  
  --auto-scaling-group-name my-asg --scaling-adjustment 30 \  
  --adjustment-type PercentChangeInCapacity
```

예: my-simple-scale-in-policy

다음 `put-scaling-policy` 명령을 사용하여 그룹 용량을 인스턴스 2개만큼 줄이는 `ChangeInCapacity` 조절 유형으로 my-simple-scale-in-policy라는 단순 조정 정책을 생성합니다.

```
aws autoscaling put-scaling-policy --policy-name my-simple-scale-in-policy \  
  --auto-scaling-group-name my-asg --scaling-adjustment -2 \  
  --adjustment-type ChangeInCapacity
```

3단계: CloudWatch 경보 생성

2단계에서는 지정하는 조건이 변경되는 경우 Auto Scaling 그룹에 확장 및 축소하는 방법에 대한 지침을 제공하는 조정 정책을 생성하였습니다. 이 단계에서는 조사하려는 지표를 식별하여 경보를 생성하고 조정 조건을 정의한 다음 해당 경보를 조정 정책과 연결합니다.

예: AddCapacity

다음 CloudWatch [put-metric-alarm](#) 명령을 사용하여 평균 CPU 임계값 60%를 기준으로 최소 2회 연속 2분간의 평가 기간 동안 Auto Scaling 그룹의 크기를 증가시키는 경보를 생성합니다. 사용자 지정 지표를 사용하려면 `--metric-name`에 지표 이름을 지정하고, `--namespace`에 네임스페이스를 지정합니다.

```
aws cloudwatch put-metric-alarm --alarm-name Step-Scaling-AlarmHigh-AddCapacity \
  --metric-name CPUUtilization --namespace AWS/EC2 --statistic Average \
  --period 120 --evaluation-periods 2 --threshold 60 \
  --comparison-operator GreaterThanOrEqualToThreshold \
  --dimensions "Name=AutoScalingGroupName,Value=my-asg" \
  --alarm-actions PolicyARN
```

예: RemoveCapacity

다음 CloudWatch [put-metric-alarm](#) 명령을 사용하여 평균 CPU 임계값 40%를 기준으로 최소 2회 연속 2분간의 평가 기간 동안 Auto Scaling 그룹의 크기를 감소시키는 경보를 생성합니다. 사용자 지정 지표를 사용하려면 `--metric-name`에 지표 이름을 지정하고, `--namespace`에 네임스페이스를 지정합니다.

```
aws cloudwatch put-metric-alarm --alarm-name Step-Scaling-AlarmLow-RemoveCapacity \
  --metric-name CPUUtilization --namespace AWS/EC2 --statistic Average \
  --period 120 --evaluation-periods 2 --threshold 40 \
  --comparison-operator LessThanOrEqualToThreshold \
  --dimensions "Name=AutoScalingGroupName,Value=my-asg" \
  --alarm-actions PolicyARN
```

Amazon SQS에 따른 조정

이 단원에서는 Amazon Simple Queue Service(Amazon SQS) 대기열의 변화하는 요구 사항에 따라 Auto Scaling 그룹을 조정하는 방법을 보여 줍니다. Amazon SQS는 지속적이고 사용 가능한 보안 호스팅 대기열을 제공하며 이를 통해 분산 소프트웨어 시스템과 구성 요소를 통합 및 분리할 수 있습니다. Amazon SQS에 대해 익숙하지 않을 경우 [Amazon Simple Queue Service 개발자 안내서](#)를 참조하십시오.

Amazon SQS 대기열의 활동에 따른 조정에 대해 고려할 경우 다음과 같은 여러 시나리오가 있습니다. 사용자가 이미지를 업로드하고 그러한 이미지를 온라인으로 사용하는 웹 앱을 예로 들어 보겠습니다. 각 이미지를 게시하려면 먼저 크기를 변경하고 인코딩해야 합니다. 이 앱은 일반적인 업로드 양을 처리하도록 구성된 Auto Scaling 그룹에서 EC2 인스턴스를 실행합니다. 인스턴스 수준을 항상 최신으로 유지하기 위해 비정상 인스턴스는 종료되고 교체됩니다. 앱은 이미지의 원시 비트맵 데이터를 처리하기 위해 Amazon SQS 대기열에 배치합니다. 그리고 이미지를 처리한 후, 사용자가 볼 수 있는 처리된 이미지를 게시합니다.

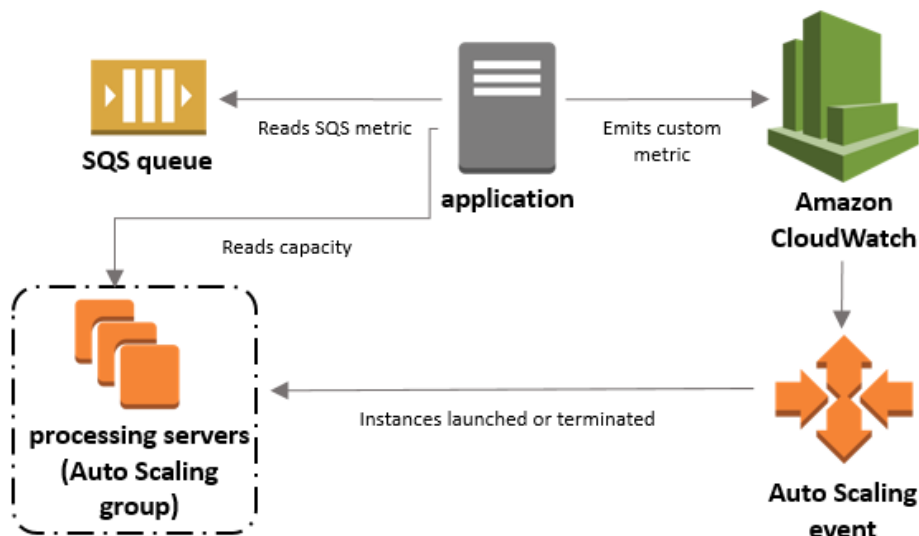
이 아키텍처는 이미지 업로드 수가 항상 일정할 경우 잘 작동합니다. 업로드 수준이 변경되면 어떻게 될까요? 업로드가 예측 가능한 일정으로 늘어나거나 줄어들면 시간과 날짜를 지정하여 조정 활동을 수행하도록 할 수 있습니다. 자세한 내용은 [Amazon EC2 Auto Scaling의 예약된 조정 \(p. 99\)](#) 단원을 참조하십시오. Auto Scaling 그룹을 조정하는 보다 동적인 방법은 정책별 조정으로, 조정 프로세스를 제어하는 파라미터를 정의할 수 있습니다. 예를 들어, 평균 업로드 수가 특정 수준에 도달하면 EC2 인스턴스의 풀릿을 확장하도록 요청하는 정책을 생성할 수 있습니다. 이는 이러한 조건이 언제 바뀔지 모를 때, 조건 변화에 따라 조정할 경우에 유용합니다.

이러한 구성을 위해 세 가지 방법을 사용할 수 있습니다.

- SQS 대기열의 메시지를 처리하기 위해 EC2 인스턴스를 관리하는 Auto Scaling 그룹.
- Auto Scaling 그룹의 EC2 인스턴스당 대기열의 메시지 수를 측정하여 Amazon CloudWatch로 전송되는 사용자 지정 지표.

- 사용자 지정 지표와 목표값에 따라 조정하도록 Auto Scaling을 구성하는 대상 추적 정책. CloudWatch 경보는 조정 정책을 호출합니다.

다음 다이어그램은 이 구성의 아키텍처를 보여 줍니다.



효과적인 지표와 목표값 선택

Amazon SQS 대기열의 메시지 수가 필요한 인스턴스 수를 단독으로 정의하지는 않습니다. 하지만 플릿의 인스턴스 수는 메시지를 처리하는 데 걸리는 시간 그리고 시간이나 허용되는 지연 시간(대기열 지연) 등과 같은 여러 요인의 영향을 받습니다.

해결 방법은 인스턴스당 백로그 지표와 목표값을 사용하여 인스턴스당 허용 백로그를 유지하는 것입니다. 이 값은 다음과 같이 계산할 수 있습니다.

- 인스턴스당 백로그: 인스턴스당 백로그를 확인하려면 Amazon SQS 지표 `ApproximateNumberOfMessages`를 사용하여 SQS 대기열의 길이(대기열에서 가져올 수 있는 메시지 수)를 확인합니다. 이 값을 플릿의 실행 용량(Auto Scaling 그룹에서 상태가 `InService`인 인스턴스 수)으로 나눠서 인스턴스당 백로그를 산출합니다.
- 인스턴스당 허용되는 백로그: 목표값을 확인하려면 먼저 애플리케이션에서 허용할 수 있는 지연 시간을 계산하십시오. 그런 다음 이 허용 지연 시간 값을, EC2 인스턴스가 메시지를 처리하기 위해 소요하는 평균 시간으로 나눕니다.

예제로 설명하기 위해 현재 `ApproximateNumberOfMessages`가 1,500개이고 플릿의 실행 용량이 10이라고 가정해 보겠습니다. 각 메시지의 평균 처리 시간 0.1초이고 가장 긴 허용 지연 시간이 10초인 경우, 인스턴스당 허용 가능한 백로그는 $10 / 0.1$ 인 100입니다. 즉, 목표 추적 정책의 목표값은 100입니다. 인스턴스당 백로그가 현재 150($1500 / 10$)이므로, 플릿은 5개의 인스턴스를 추가하여 목표값 비율을 유지합니다.

다음은 사용자 지정 지표를 만들고, Auto Scaling 그룹이 해당 수치에 따라 조정을 수행하도록 구성하는 목표 추적 조정 정책을 생성하는 예제입니다.

Amazon SQS로 조정 구성

다음 단원에서는 AWS CLI를 사용하여 SQS 대기열의 자동 조정을 설정하는 방법을 설명합니다. 이 절차에서는 대기열(표준 또는 FIFO), Auto Scaling 그룹, 대기열을 사용하는 애플리케이션에서 실행 중인 EC2 인스턴스가 이미 있는 것으로 가정합니다.

작업

- 1단계: CloudWatch 사용자 지정 지표 생성 (p. 117)
- 2단계: 대상 추적 조정 정책 생성 (p. 117)
- 3단계: 조정 정책 테스트 (p. 118)

1단계: CloudWatch 사용자 지정 지표 생성

먼저 AWS 계정에서 지표를 읽고 사용자 지정 계산 지표를 만듭니다. 그런 다음, 앞 단원에서 권장한 인스턴스당 백로그 지표를 계산합니다. 마지막으로 이 수치를 1분 단위로 CloudWatch에 게시합니다.

가능한 경우, 사용률 변화에 신속하게 대응하기 위해 EC2 인스턴스 지표를 1분 간격(세부 모니터링)으로 조정해야 합니다. 주기를 5분으로 하면 응답 시간이 느려질 뿐만 아니라 오랜 시간이 지난 측정치 데이터를 기준으로 조정하게 됩니다. EC2 인스턴스는 기본 모니터링, 즉 5분 주기로 인스턴스의 측정치 데이터를 사용하도록 기본적으로 활성화됩니다. 하지만 세부 모니터링을 활성화하여 인스턴스의 측정치 데이터를 가져오는 주기를 1분으로 바꿀 수 있습니다. 자세한 내용은 [Amazon CloudWatch를 사용하여 Auto Scaling 그룹 및 인스턴스 모니터링 \(p. 160\)](#) 단원을 참조하십시오.

CloudWatch 사용자 지정 지표를 생성하려면

1. SQS `get-queue-attributes` 명령을 사용하여 대기열에서 대기 중인 메시지 수 (ApproximateNumberOfMessages)를 확인합니다.

```
aws sqs get-queue-attributes --queue-url https://sqs.region.amazonaws.com/123456789/MyQueue \
--attribute-names ApproximateNumberOfMessages
```

2. 그룹의 실행 용량(InService 수명 주기 상태의 인스턴스 수)을 확인하려면 `describe-auto-scaling-groups` 명령을 사용합니다. 이 명령은 Auto Scaling 그룹의 인스턴스와 해당 수명 주기 상태를 반환합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-names my-asg
```

3. ApproximateNumberOfMessages 지표를 플릿의 실행 용량 지표로 나눠서 인스턴스당 백로그를 계산합니다.
4. AWS CLI 또는 API를 사용하여 결과를 CloudWatch 사용자 지정 지표로 1분마다 게시합니다. 사용자 지정 지표는 지표 이름과 선택한 네임스페이스를 사용하여 정의됩니다. 사용자 지정 지표를 위한 네임스페이스는 "AWS/"로 시작할 수 없습니다. 사용자 지정 지표 게시에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서의 [사용자 지정 지표 게시](#) 주제를 참조하십시오.

다음은 CLI `put-metric-data` 명령의 예입니다.

```
aws cloudwatch put-metric-data --metric-name MyBacklogPerInstance --
namespace MyNamespace \
--unit None --value 20 --
dimensions MyOptionalMetricDimensionName=MyOptionalMetricDimensionValue
```

애플리케이션에서 해당 지표를 생성하면 데이터가 CloudWatch로 전송됩니다. 이 지표는 CloudWatch 콘솔에 표시됩니다. AWS Management 콘솔에 로그인한 후 CloudWatch 페이지로 이동하여 지표에 액세스할 수 있습니다. 그런 다음 지표 페이지로 이동하거나 검색 상자에서 검색하여 지표를 볼 수 있습니다. 지표 보기에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서의 [사용 가능 지표 보기](#)를 참조하십시오.

2단계: 대상 추적 조정 정책 생성

이제 애플리케이션의 부하가 달라지면 Auto Scaling 그룹에서 실행 중인 EC2 인스턴스 수를 늘리거나 줄이라고 이 그룹에 동적으로 알려 주는 대상 추적 조정 정책을 생성합니다. 조정 지표가 그룹 용량에 비례하여 증가하거나 감소하는 사용률 지표이므로 대상 추적 조정 정책을 사용할 수 있습니다.

대상 추적 조정 정책을 생성하려면

1. 다음 명령을 사용하여 홈 디렉터리의 config.json 파일에 조정 정책에 대한 목표값을 지정합니다.

TargetValue에, 인스턴스당 허용 백로그 지표를 계산하여 입력합니다. 이 값을 계산하려면 보통 지연 시간 값을 확인하여, 이 값을 메시지를 처리하는 데 걸리는 평균 시간으로 나눕니다.

```
$ cat ~/config.json
{
  "TargetValue":100,
  "CustomizedMetricSpecification":{
    "MetricName":"MyBacklogPerInstance",
    "Namespace":"MyNamespace",
    "Dimensions":[
      {
        "Name":"MyOptionalMetricDimensionName",
        "Value":"MyOptionalMetricDimensionValue"
      }
    ],
    "Statistic":"Average",
    "Unit":"None"
  }
}
```

2. `put-scaling-policy` 명령과 전 단계에서 만든 config.json 파일을 사용하여 조정 정책을 생성합니다.

```
aws autoscaling put-scaling-policy --policy-name my-scaling-policy --auto-scaling-
group-name my-asg \
  --policy-type TargetTrackingScaling --target-tracking-configuration file:///~/
config.json
```

이 정책은 확장과 축소에 대해 한 개씩 두 개의 경보를 생성합니다. 또한 CloudWatch에 등록된 정책의 Amazon 리소스 이름(ARN)을 반환합니다. CloudWatch는 지표가 위반될 때마다 이 이름을 사용하여 조정 기능을 호출합니다.

3단계: 조정 정책 테스트

설정을 마친 후, 조정 정책이 작동하는지 확인합니다. SQS 대기열의 메시지 수를 늘린 다음 Auto Scaling 그룹에서 추가 EC2 인스턴스를 시작했는지 확인하여 테스트할 수 있습니다. 또한 SQS 대기열의 메시지 수를 줄인 다음 Auto Scaling 그룹에서 EC2 인스턴스를 종료했는지 확인하여 테스트할 수 있습니다.

확장 기능을 테스트하려면

1. [자습서: Amazon SQS 대기열로 메시지 전송](#)의 단계를 수행하여 대기열에 메시지를 추가합니다. 인스턴스당 백로그 지표가 목표값을 초과하도록 대기열의 메시지 수를 늘려야 합니다.

변경 사항에 따라 CloudWatch 경보가 발생하기까지 몇 분 정도 걸릴 수 있습니다.

2. `describe-auto-scaling-groups` 명령을 사용하여 그룹에서 새 인스턴스를 시작했는지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

축소 기능을 테스트하려면

1. [자습서: Amazon SQS 대기열로 메시지 전송](#)의 단계를 수행하여 대기열에서 메시지를 제거합니다. 인스턴스당 백로그 지표가 목표값 미만이 되도록 대기열의 메시지 수를 줄여야 합니다.

변경 사항에 따라 CloudWatch 경보가 발생하기까지 몇 분 정도 걸릴 수 있습니다.

2. `describe-auto-scaling-groups` 명령을 사용하여 그룹에서 새 인스턴스를 종료했는지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

Auto Scaling 그룹에 대한 조정 활동 확인

조정 정책이 생성되면 Amazon EC2 Auto Scaling은 해당 지표를 기준으로 정책 평가를 시작합니다. 지표가 지정된 수의 평가 기간에 대한 임계값을 위반할 경우 지표 경보가 ALARM(경보) 상태가 됩니다. 즉, 조정 정책이 생성 직후 조정 작업을 트리거할 수 있습니다. Amazon EC2 Auto Scaling이 조정 정책에 응답하여 용량을 변경한 후 계정에서 조정 활동을 확인할 수 있습니다. Amazon EC2 Auto Scaling으로부터 조정 작업이 트리거되었음을 알리는 이메일 알림을 받으려면 [Auto Scaling 그룹 조정 시 Amazon SNS SNS 알림 받기 \(p. 150\)](#)의 지침을 따르십시오.

Auto Scaling 그룹의 조정 활동을 보려면(새 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹의 이름을 검색합니다.

인스턴스 옆에는 현재 실행 중인 인스턴스 수가 표시됩니다. 인스턴스가 시작되거나 종료되는 동안 상태 옆에 “용량 업데이트 중” 상태가 표시됩니다. 몇 분 동안 기다렸다가 보기를 새로 고쳐 최신 상태를 확인합니다. 조정 활동이 완료되면 인스턴스 및 원하는 용량 옆에 새 값이 표시됩니다.

Note

인스턴스 가중치를 사용하는 경우 Weighted capacity(가중치 적용 용량) 옆은 그룹에 포함된 용량 유닛 수를 측정합니다. 이 옆이 숨겨져 있으면 색선의 오른쪽 상단 모서리에서 기어 모양 아이콘을 선택한 다음 Weighted capacity(가중치 적용 용량)을 활성화합니다.

4. Auto Scaling 그룹 옆의 확인란을 선택합니다.
 - a. 활동 탭에서 활동 기록의 상태 옆에 Auto Scaling 그룹이 성공적으로 인스턴스를 시작 또는 종료했는지 여부가 표시됩니다.
 - b. 인스턴스 관리 탭의 인스턴스에서 현재 실행 중인 인스턴스의 상태를 볼 수 있습니다. 수명 주기 옆에는 인스턴스의 상태가 포함됩니다. 인스턴스를 시작하는 데 약간 시간이 걸립니다. 인스턴스가 시작되면 해당 수명 주기 상태가 InService로 변경됩니다.

Auto Scaling 그룹의 조정 활동을 보려면(AWS CLI)

다음 `describe-scaling-activities` 명령을 사용합니다.

```
aws autoscaling describe-scaling-activities --auto-scaling-group-name my-asg
```

다음은 예제 출력입니다.

조정 활동은 시작 시간으로 정렬됩니다. 아직 진행 중인 활동이 먼저 설명됩니다.

```
{
  "Activities": [
    {
      "ActivityId": "5e3a1f47-2309-415c-bfd8-35aa06300799",
      "AutoScalingGroupName": "my-asg",
      "Description": "Terminating EC2 instance: i-06c4794c2499af1df",
      "Cause": "At 2020-02-11T18:34:10Z a monitor alarm TargetTracking-my-asg-AlarmLow-b9376cab-18a7-4385-920c-dfa3f7783f82 in state ALARM triggered policy my-target-tracking-policy changing the desired capacity from 3 to 2. At 2020-02-11T18:34:31Z an instance
```

```
was taken out of service in response to a difference between desired and actual capacity,
shrinking the capacity from 3 to 2. At 2020-02-11T18:34:31Z instance i-06c4794c2499af1df
was selected for termination.",
  "StartTime": "2020-02-11T18:34:31.268Z",
  "EndTime": "2020-02-11T18:34:53Z",
  "StatusCode": "Successful",
  "Progress": 100,
  "Details": "{\"Subnet ID\":\"subnet-5ea0c127\",\"Availability Zone\":\"us-west-2a
\\\"...}\"
},
...
]
```

Auto Scaling 그룹의 크기를 확인하려면(AWS CLI)

`describe-auto-scaling-groups` 명령을 사용합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

다음 예는 그룹 및 현재 실행 중인 인스턴스의 세부 정보가 표시된 출력입니다.

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "ServiceLinkedRoleARN": "arn",
      "TargetGroupARNs": [],
      "SuspendedProcesses": [],
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      },
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-asg",
      "DefaultCooldown": 300,
      "MinSize": 1,
      "Instances": [
        {
          "ProtectedFromScaleIn": false,
          "AvailabilityZone": "us-west-2a",
          "LaunchTemplate": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1",
            "LaunchTemplateId": "lt-050555ad16a3f9c7f"
          },
          "InstanceId": "i-05b4f7d5be44822a6",
          "HealthStatus": "Healthy",
          "LifecycleState": "Pending"
        },
        {
          "ProtectedFromScaleIn": false,
          "AvailabilityZone": "us-west-2a",
          "LaunchTemplate": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1",
            "LaunchTemplateId": "lt-050555ad16a3f9c7f"
          },
          "InstanceId": "i-0c20ac468fa3049e8",
          "HealthStatus": "Healthy",
```



```

        "LifecycleState": "InService"
      },
      "MaxSize": 5,
      "VPCZoneIdentifier": "subnet-c87f2be0",
      "HealthCheckGracePeriod": 300,
      "TerminationPolicies": [
        "Default"
      ],
      "CreatedTime": "2019-03-18T23:30:42.611Z",
      "AvailabilityZones": [
        "us-west-2a"
      ],
      "HealthCheckType": "EC2",
      "NewInstancesProtectedFromScaleIn": false,
      "DesiredCapacity": 2
    }
  ]
}

```

Auto Scaling 그룹에 대한 조정 정책 비활성화

이 항목에서는 Auto Scaling 그룹에 포함된 인스턴스 수를 변경하지 않도록 조정 정책을 일시적으로 비활성화하는 방법을 설명합니다. 조정 정책을 비활성화하면 구성 세부 정보가 보존되므로 정책을 신속하게 다시 활성화할 수 있습니다. 이는 필요하지 않을 때 일시적으로 정책을 삭제했다가 나중에 다시 생성하는 것보다 쉽습니다.

조정 정책을 비활성화하면 조정 정책이 비활성화된 동안 위반된 지표 경보에 대해 Auto Scaling 그룹이 확장 또는 축소되지 않습니다. 그러나 진행 중인 조정 활동은 중지되지 않습니다.

비활성화된 조정 정책도 Auto Scaling 그룹에 추가할 수 있는 조정 정책 수 할당량에 포함됩니다.

조정 정책을 비활성화하려면(새 콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 자동 조정 탭의 조정 정책에서 조정 정책을 선택한 다음 작업, 비활성화를 선택합니다.

조정 정책을 다시 활성화할 준비가 되면 이러한 단계를 반복한 다음 작업, 활성화를 선택합니다. 조정 정책을 다시 활성화한 후 현재 ALARM(경보) 상태에 있는 경보가 있으면 Auto Scaling 그룹이 즉시 조정 작업을 시작할 수 있습니다.

조정 정책을 비활성화하려면(AWS CLI)

다음과 같이 `put-scaling-policy` 명령을 `--no-enabled` 옵션과 함께 사용합니다. 정책을 생성할 때 지정하는 대로 명령에서 모든 옵션을 지정합니다.

```

aws autoscaling put-scaling-policy --auto-scaling-group-name my-asg \
  --policy-name my-scaling-policy --policy-type TargetTrackingScaling \
  --estimated-instance-warmup 360 \
  --target-tracking-configuration '{ "TargetValue": 70, "PredefinedMetricSpecification":
  { "PredefinedMetricType": "ASGAverageCPUUtilization" } }' \
  --no-enabled

```

조정 정책을 다시 활성화하려면(AWS CLI)

다음과 같이 `put-scaling-policy` 명령을 `--enabled` 옵션과 함께 사용합니다. 정책을 생성할 때 지정하는 대로 명령에서 모든 옵션을 지정합니다.


```
aws autoscaling put-scaling-policy --auto-scaling-group-name my-asg \
  --policy-name my-scaling-policy --policy-type TargetTrackingScaling \
  --estimated-instance-warmup 360 \
  --target-tracking-configuration '{ "TargetValue": 70, "PredefinedMetricSpecification":
  { "PredefinedMetricType": "ASGAverageCPUUtilization" } }' \
  --enabled
```

조정 정책을 설명하려면(AWS CLI)

[describe-policies](#) 명령을 사용하여 조정 정책의 활성화 상태를 확인합니다.

```
aws autoscaling describe-policies --auto-scaling-group-name my-asg \
  --policy-names my-scaling-policy
```

다음은 예제 출력입니다.

```
{
  "ScalingPolicies": [
    {
      "AutoScalingGroupName": "my-asg",
      "PolicyName": "my-scaling-policy",
      "PolicyARN": "arn:aws:autoscaling:us-
west-2:123456789012:scalingPolicy:1d52783a-b03b-4710-
bb0e-549fd64378cc:autoScalingGroupName/my-asg:policyName/my-scaling-policy",
      "PolicyType": "TargetTrackingScaling",
      "StepAdjustments": [],
      "Alarms": [
        {
          "AlarmName": "TargetTracking-my-asg-AlarmHigh-9ca53fdd-7cf5-4223-938a-
ae1199204502",
          "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmHigh-9ca53fdd-7cf5-4223-938a-
ae1199204502"
        },
        {
          "AlarmName": "TargetTracking-my-asg-AlarmLow-7010c83d-d55a-4a7a-
abe0-1cf8b9de6d6c",
          "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmLow-7010c83d-d55a-4a7a-
abe0-1cf8b9de6d6c"
        }
      ],
      "TargetTrackingConfiguration": {
        "PredefinedMetricSpecification": {
          "PredefinedMetricType": "ASGAverageCPUUtilization"
        },
        "TargetValue": 70.0,
        "DisableScaleIn": false
      },
      "Enabled": true
    }
  ]
}
```

조정 정책 삭제

더 이상 필요 없는 조정 정책은 삭제할 수 있습니다. 조정 정책의 유형에 따라 CloudWatch 경보를 삭제해야 할 수도 있습니다. 대상 추적 조정 정책을 삭제하면 관련된 모든 CloudWatch 경보도 삭제됩니다. 단계 조정 정책 또는 단순 조정 정책을 삭제하면 기본 경보 작업이 삭제되지만 연결된 작업이 더 이상 없는 경우에도 CloudWatch 경보는 삭제되지 않습니다.

조정 정책을 삭제하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 자동 조정 탭의 조정 정책에서 조정 정책을 선택한 다음 작업, 삭제를 선택합니다. (이전 콘솔: 조정 정책 탭에서 정책을 삭제할 수 있습니다.)
5. 확인 메시지가 나타나면 예, 삭제합니다를 선택합니다.
6. (선택 사항) 단계 조정 정책 또는 단순 조정 정책을 삭제한 경우 다음을 수행하여 정책과 연관된 CloudWatch 경보를 삭제하십시오. 나중에 사용할 수 있도록 경보를 유지하려면 다음 하위 단계를 건너뛸 수 있습니다.
 - a. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
 - b. 탐색 창에서 [Alarms]를 선택합니다.
 - c. 경보(예: Step-Scaling-AlarmHigh-AddCapacity)를 선택하고 작업, 삭제를 선택합니다.
 - d. 확인 메시지가 나타나면 Delete를 선택합니다.

조정 정책을 삭제하려면(AWS CLI)

다음 `delete-policy` 명령을 사용합니다.

```
aws autoscaling delete-policy --auto-scaling-group-name my-asg \
  --policy-name my-step-scale-in-policy
```

CloudWatch 경보를 삭제하려면(AWS CLI)

단계 조정 정책 및 단순 조정 정책의 경우 `delete-alarms` 명령을 사용하여 정책과 관련된 CloudWatch 경보를 삭제하십시오. 나중에 경보를 사용하려면 이 단계를 건너뛸 수 있습니다. 한번에 하나 이상 경보를 삭제할 수 있습니다. 예를 들어, 다음 명령을 사용하여 Step-Scaling-AlarmHigh-AddCapacity 및 Step-Scaling-AlarmLow-RemoveCapacity 경보를 삭제합니다.

```
aws cloudwatch delete-alarms --alarm-name Step-Scaling-AlarmHigh-AddCapacity Step-Scaling-AlarmLow-RemoveCapacity
```

Amazon EC2 Auto Scaling의 조정 휴지

휴지 기간은 Auto Scaling 그룹이 이전 조정 활동이 적용되기 전에 추가 인스턴스를 시작 또는 종료하지 않게 합니다. 인스턴스 워밍업 기간 또는 다른 애플리케이션 요구 사항에 따라 시간 길이를 구성할 수 있습니다.

Amazon EC2 Auto Scaling은 단순 조정 정책을 사용할 때 휴지 기간을 지원합니다. 기타 조정 정책을 사용할 때는 지원하지 않습니다. Auto Scaling 그룹이 단순 조정 정책을 사용하여 동적으로 조정하면 휴지 기간이 완료될 때까지 대기한 후에 조정 활동을 재개합니다. Auto Scaling 그룹에 연결된 기본 휴지 기간을 사용하거나 정책에 휴지 기간을 지정하여 기본값을 무시할 수 있습니다. 단순 조정에 대한 자세한 내용은 [Amazon EC2 Auto Scaling에 대한 단계 및 단순 조정 정책 \(p. 107\)](#) 단원을 참조하십시오.

Auto Scaling 그룹을 수동으로 조정하는 경우 기본적으로 휴지 기간이 적용되지 않지만 기본값을 재정의하고 휴지 기간을 정의할 수 있습니다. 인스턴스가 비정상적 상태인 경우 그룹은 휴지 기간이 완료될 때까지 대기하지 않고 비정상적 인스턴스를 교체합니다. 수동 조정에 관한 자세한 내용은 [Amazon EC2 Auto Scaling 수동 조정 \(p. 90\)](#) 단원을 참조하십시오.

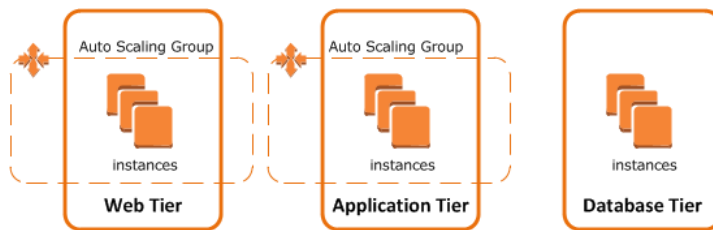
Amazon EC2 Auto Scaling은 기본 휴지 기간과 조정 특정 휴지 기간을 모두 지원합니다.

목차

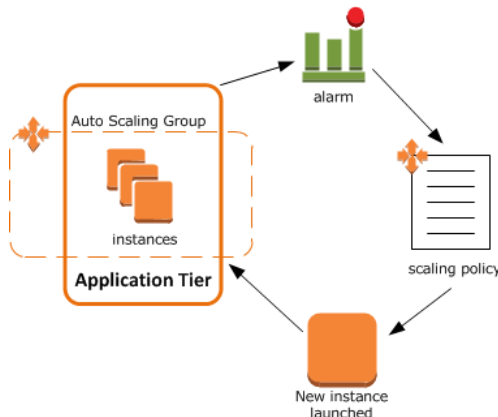
- 예: 휴지 (p. 124)
- 기본 휴지 (p. 125)
- 조정 특정 휴지 (p. 125)
- 휴지 및 다중 인스턴스 (p. 125)
- 휴지 및 수명 주기 후크 (p. 125)

예: 휴지

다음 시나리오를 고려하십시오. AWS에 실행 중인 웹 애플리케이션이 있습니다. 이 웹 애플리케이션은 웹, 애플리케이션 및 데이터베이스의 3가지 기본 티어로 구성됩니다. 애플리케이션에 트래픽 요구 사항을 충족하는 리소스가 항상 유지되도록 하려면 웹 티어용과 애플리케이션 티어용으로 Auto Scaling 그룹을 2개 생성합니다.



애플리케이션 티어용 Auto Scaling 그룹에 적절한 수의 EC2 인스턴스가 있도록 하려면 인스턴스의 CPUUtilization 지표가 90%를 초과할 때마다 확장하는 CloudWatch 경보를 생성 (p. 165)합니다. 경보가 발생하면 그룹에서 다른 인스턴스를 시작하고 구성합니다.



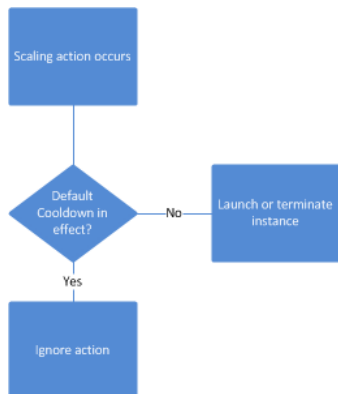
이러한 인스턴스는 구성 스크립트를 사용하여 인스턴스가 서비스 상태가 되기 전에 소프트웨어를 설치하고 구성합니다. 결과적으로 인스턴스가 시작된 다음 완전한 서비스 상태가 될 때까지 2 - 3분 가량 소요됩니다. 실제 시간은 인스턴스 크기 및 완료할 시작 스크립트가 있는지 여부 등의 여러 요인에 따라 다릅니다.

이제 트래픽이 급증하면 CloudWatch 경보가 울립니다. 경보가 울리면 그룹에서 인스턴스를 시작하여 수요 증가를 지원합니다. 하지만 여기에는 한 가지 문제가 있는데, 인스턴스를 시작하는 데 몇 분이 소요된다는 것입니다. 이 시간 동안 CloudWatch 경보가 계속 울릴 수 있으며 그 결과 경보가 울릴 때마다 Auto Scaling 그룹에서 또 다른 인스턴스를 시작하게 됩니다.

그러나 휴지 기간이 적용되면 Auto Scaling 그룹은 인스턴스를 시작한 다음 지정된 시간이 경과할 때까지 단순 조정 정책 또는 수동 조정으로 인한 조정 활동을 일시 중지합니다. (기본값은 300초입니다.) 이 시간 동안 새로 시작된 인스턴스는 애플리케이션 트래픽을 처리할 수 있습니다. 휴지 기간이 만료되면 일시 중지된 조정 작업이 재개됩니다. CloudWatch 경보가 다시 울리면 Auto Scaling 그룹은 또 다른 인스턴스를 시작하고 다시 휴지 기간이 적용됩니다. 하지만 추가 인스턴스가 CPU 사용률을 되돌릴 만큼 충분한 경우 그룹은 현재 크기를 계속 유지합니다.

기본 휴지

기본 휴지 기간은 Auto Scaling 그룹을 생성할 때 적용됩니다. 기본값은 300초입니다. 이 휴지 기간은 단순 조정 정책을 위한 모든 조정 활동에 적용되며, 경우에 따라 수동 조정 활동에 적용되도록 요청할 수 있습니다.



AWS Management 콘솔, [create-auto-scaling-group](#) 명령(AWS CLI) 또는 [CreateAutoScalingGroup](#) API 작업을 사용하여 Auto Scaling 그룹을 만들 때 기본 휴지 기간을 구성할 수 있습니다.

AWS Management 콘솔, [update-auto-scaling-group](#) 명령(AWS CLI) 또는 [UpdateAutoScalingGroup](#) API 작업을 사용하여 언제든지 기본 휴지 기간을 변경할 수 있습니다.

조정 특정 휴지

Auto Scaling 그룹에 대해 기본 휴지 기간을 지정하는 것 외에도 특정 단순 조정 정책 또는 수동 조정에 적용되는 휴지를 생성할 수 있습니다. 조정 특정 휴지 기간은 기본 휴지 기간보다 우선합니다.

조정 특정 휴지는 특정 기준 또는 지표에 따라 인스턴스를 종료하는 축소 정책에서 자주 사용됩니다.— 이 정책이 인스턴스를 종료하기 때문에 Amazon EC2 Auto Scaling은 추가 인스턴스를 종료할지 여부를 결정하는데 더 적은 시간이 소요됩니다. 따라서 300초의 기본 휴지 기간은 너무 깁니다. 이때 축소 정책에 180초의 조정 특정 휴지 기간을 적용하면 그룹이 더 빨리 축소되므로 비용을 절감할 수 있습니다.

AWS Management 콘솔, [put-scaling-policy](#) 명령(AWS CLI) 또는 [PutScalingPolicy](#) API 작업을 사용하여 조정 특정 휴지 기간을 생성할 수 있습니다.

휴지 및 다중 인스턴스

앞의 단원에는 단일 인스턴스를 시작 또는 종료할 때 휴지 기간이 Auto Scaling 그룹에 어떤 영향을 미치는지 보여주는 예가 나와 있습니다. 하지만 일반적으로 Auto Scaling 그룹은 한 번에 둘 이상의 인스턴스를 시작합니다. 예를 들어, 특정 측정치 임계값이 충족되는 경우 그룹에서 인스턴스 3개를 시작하도록 선택할 수 있습니다.

여러 인스턴스를 시작하는 경우 휴지 기간(기본 휴지 또는 조정 특정 휴지)은 마지막 인스턴스가 시작될 때 적용되기 시작합니다.

휴지 및 수명 주기 후크

수명 주기 후크를 Auto Scaling 그룹에 추가할 수 있습니다. 이러한 후크를 사용하여 Auto Scaling 그룹 내에서 인스턴스가 시작되고 종료되는 방식을 제어할 수 있습니다. 인스턴스가 서비스 상태가 되거나 종료되기 전에 인스턴스에 대한 작업을 수행할 수 있습니다.

수명 주기 후크는 Auto Scaling 그룹, 수동 조정 또는 단순 조정 정책에 대해 구성된 모든 휴지 기간에 영향을 미칠 수 있습니다. 휴지 기간은 해당 인스턴스가 대기 상태를 벗어날 때까지 시작되지 않습니다.

축소 시 Auto Scaling 인스턴스 종료 제어

각 Auto Scaling 그룹에서 네트워크 아키텍처로부터 인스턴스를 추가(확장)하거나 인스턴스를 제거(축소)하는 시점을 제어할 수 있습니다. 원하는 용량을 조정하여 수동으로 그룹 크기를 조정하거나 예약된 조정 또는 조정 정책을 사용하여 프로세스를 자동화할 수 있습니다.

이 항목에서는 기본 종료 정책과 사용자 지정 종료 정책을 구성하는 데 사용할 수 있는 옵션에 대해 설명합니다. 종료 정책을 사용하면 축소 이벤트가 발생할 때 먼저 종료할 인스턴스를 제어할 수 있습니다.

또한 이 주제에서는 인스턴스 축소 보호 기능을 사용하여 자동 축소 동안 특정 인스턴스가 종료되지 않도록 하는 방법을 설명합니다. Auto Scaling 그룹 인스턴스의 경우, 축소 이벤트가 발생할 때 Amazon EC2 Auto Scaling 기능을 사용하여 인스턴스를 보호합니다. 인스턴스가 실제로 종료되지 않도록 방지하려면, Amazon EC2 종료 방지를 사용합니다.

Note

Amazon EC2 Auto Scaling은 Auto Scaling 그룹에 [여러 유형의 구매 옵션 \(p. 45\)](#)이 포함된 독특한 상황에서 두 가지 유형(스팟 또는 온디맨드) 중 어느 것을 먼저 종료해야 할지 확인하며, 가용 영역 간에 인스턴스 균형을 유지하도록 설정된 경우 해당 유형의 인스턴스가 가장 많은 가용 영역을 선택하여 균형을 유지합니다. 그런 다음 기본 또는 사용자 지정 종료 정책을 적용합니다.

목차

- [기본 종료 정책 \(p. 126\)](#)
- [종료 정책 사용자 지정 \(p. 127\)](#)
- [인스턴스 축소 보호 \(p. 128\)](#)

기본 종료 정책

기본 종료 정책은 인스턴스를 고르게 분포시켜 가용 영역을 높이도록 설계되었습니다. 기본 정책은 다양한 시나리오를 다루기 위해 일반성과 유연성이 유지됩니다.

기본 종료 정책은 다음과 같이 동작합니다.

1. 인스턴스가 가장 많으며 축소로부터 보호되지 않는 인스턴스가 최소 하나 이상 있는 가용 영역을 확인합니다.
2. 종료할 인스턴스를 결정하고 종료 중인 온디맨드 또는 스팟 인스턴스의 할당 전략에 나머지 인스턴스를 맞춥니다. 이는 [할당 전략 \(p. 45\)](#)을 지정하는 Auto Scaling 그룹에만 적용됩니다.

예를 들어 인스턴스가 시작된 후 기본 설정 인스턴스 유형의 우선 순위를 변경할 경우, 축소 이벤트가 발생했을 때 Amazon EC2 Auto Scaling이 온디맨드 인스턴스를 우선 순위가 낮은 인스턴스 유형에서 점진적으로 이동하려고 시도합니다.

3. 이 중에서 가장 오래된 시작 템플릿 또는 구성을 사용하는 인스턴스가 있는지 확인합니다.

a. [시작 템플릿을 사용하는 Auto Scaling 그룹의 경우]

시작 구성을 사용하는 인스턴스가 없는 한 가장 오래된 시작 템플릿을 사용하는 인스턴스가 있는지 확인합니다. Amazon EC2 Auto Scaling은 시작 템플릿을 사용하는 인스턴스 이전에 시작 구성을 사용하는 인스턴스를 종료합니다.

b. [시작 구성을 사용하는 Auto Scaling 그룹의 경우]

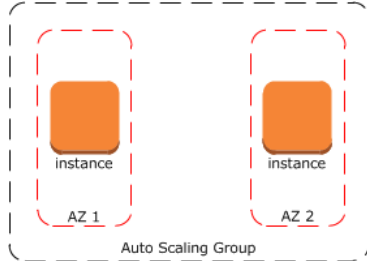
이 중에서 가장 오래된 시작 구성을 사용하는 인스턴스가 있는지 확인합니다.

4. 위의 기준에 따라 종료할 비보호 인스턴스가 여러 개 있는 경우, 다음 결제 시간에 가장 근접한 인스턴스가 무엇인지 확인합니다. 다음 번 결제 시간에 가장 근접한 비보호 인스턴스가 여러 개인 경우 이러한 인스턴스 중 하나를 임의로 종료합니다.

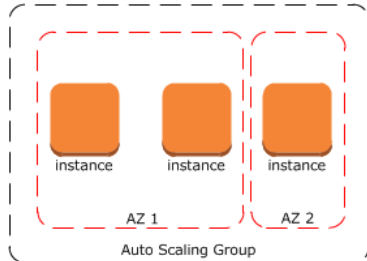
다음 결제 시간에 가장 근접한 인스턴스를 종료하면 시간당 청구되는 인스턴스의 사용을 극대화할 수 있습니다. 또는 해당 Auto Scaling 그룹에서 Amazon Linux 또는 Ubuntu를 사용하는 경우 EC2 사용량은 초단위로 요금이 부과됩니다. 자세한 내용은 [Amazon EC2 요금](#)을 참조하십시오.

예

시작 구성을 사용하는 Auto Scaling 그룹을 고려합니다. 여기에는 인스턴스 유형 1개, 가용 영역 2개, 두 인스턴스의 목표 용량, 특정 임계값이 충족될 때 인스턴스 수가 1씩 증감하는 조정 정책이 있습니다. 이 그룹에 있는 2개의 인스턴스는 다음과 같이 배포됩니다.



확장 정책에 대한 임계값이 충족되면 정책이 적용되고 Auto Scaling 그룹에서 새로운 인스턴스를 시작합니다. 이제 Auto Scaling 그룹에 3개의 인스턴스가 있고 다음과 같이 배포됩니다.



축소 정책에 대한 임계값이 충족되면 정책이 적용되고 Auto Scaling 그룹에서 인스턴스 한 개를 종료합니다. 그룹에 특정 종료 정책을 할당하지 않은 경우 기본 종료 정책을 사용합니다. 인스턴스가 2개인 가용 영역을 선택하고 가장 오래된 시작 구성을 시작하는 인스턴스를 종료합니다. 인스턴스가 동일한 시작 구성에서 시작된 경우 Auto Scaling 그룹은 다음 번 결제 시간에 가장 근접한 인스턴스를 선택한 후 이를 종료합니다.

Amazon EC2 Auto Scaling을 사용한 고가용성에 대한 자세한 내용은 [가용 영역에 인스턴스 분산 \(p. 6\)](#)을 참조하십시오.

종료 정책 사용자 지정

기본 정책을 사용자 지정 정책으로 대체하여 애플리케이션의 현재 버전이 있는 인스턴스를 유지하는 것과 같은 일반적인 사용 사례를 지원할 수 있습니다.

종료 정책을 사용자 지정할 때 가용 영역에 그룹에서 사용하는 다른 가용 영역 외에 추가 인스턴스가 있으면 종료 정책은 불균형이 있는 가용 영역의 인스턴스에 적용됩니다. 그룹에서 사용하는 가용 영역이 균형적인 경우 종료 정책은 그룹에 대한 모든 가용 영역에 걸쳐 적용됩니다.

Amazon EC2 Auto Scaling은 현재 다음 사용자 지정 종료 정책을 지원합니다.

- **OldestInstance** 그룹에서 가장 오래된 인스턴스를 종료합니다. 이 옵션은 Auto Scaling 그룹의 인스턴스를 새로운 EC2 인스턴스 유형으로 업그레이드할 때 유용합니다. 따라서 이전 유형의 인스턴스를 새로운 유형의 인스턴스로 점진적으로 교체할 수 있습니다.
- **NewestInstance** 그룹에서 가장 새로운 인스턴스를 종료합니다. 이 정책은 새로운 시작 구성을 테스트 하지만 프로덕션 상태로 유지하고 싶지 않은 경우에 유용합니다.

- `OldestLaunchConfiguration`를 선택하십시오. 가장 오래된 시작 구성을 가진 인스턴스를 종료합니다. 이 정책은 그룹을 업데이트하고 이전 구성에서 인스턴스를 단계적으로 종료할 때 유용합니다.
- `ClosestToNextInstanceHour`를 선택하십시오. 다음 번 결제 시간에 가장 근접한 인스턴스를 종료합니다. 이 정책은 시간제로 요금이 청구되는 인스턴스의 사용을 극대화할 수 있도록 합니다.
- `Default` 기본 종료 정책에 따라 인스턴스를 종료합니다. 이 정책은 그룹에 대한 조정 정책이 둘 이상일 경우에 유용합니다.
- `OldestLaunchTemplate`. 가장 오래된 시작 템플릿이 있는 인스턴스를 종료합니다. 이 정책을 사용하는 경우, 최신이 아닌 시작 템플릿을 사용하는 인스턴스가 먼저 종료되고 그 다음으로 최신 시작 템플릿의 가장 오래된 버전을 사용하는 인스턴스가 종료됩니다. 이 정책은 그룹을 업데이트하고 이전 구성에서 인스턴스를 단계적으로 종료할 때 유용합니다.
- `AllocationStrategy`. Auto Scaling 그룹의 인스턴스를 종료하여 종료 중인 인스턴스 유형 (스팟 인스턴스 또는 온디맨드 인스턴스)에 대한 할당 전략에 나머지 인스턴스를 맞춥니다. 이 정책은 선호하는 인스턴스 유형이 변경된 경우 유용합니다. 스팟 할당 전략이 `lowest-price`인 경우, 최저가 스팟 풀 N개 전체에서 스팟 인스턴스 배포를 점차적으로 재분배할 수 있습니다. 스팟 할당 전략이 `capacity-optimized`인 경우, 사용 가능한 스팟 용량이 더 여유 있는 스팟 풀에서 스팟 인스턴스 배포를 점차적으로 재분배할 수 있습니다. 또한 점차적으로 우선 순위가 낮은 유형의 온디맨드 인스턴스를 우선 순위가 높은 유형의 온디맨드 인스턴스로 대체할 수 있습니다.

종료 정책을 사용자 지정하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. [Actions]에 대해 [Edit]을 선택합니다.
5. 세부 정보 탭의 고급 구성에서 종료 정책을 찾습니다. 종료 정책을 하나 이상의 선택합니다. 정책을 여러 개 선택할 경우 적용되어야 하는 순서대로 나열합니다. 기본 정책을 사용할 경우 목록에서 기본 정책이 제일 마지막에 오도록 합니다.
6. 저장을 선택합니다.

종료 정책을 사용자 지정하려면(AWS CLI)

다음 명령 중 하나를 사용합니다.

- `create-auto-scaling-group`
- `update-auto-scaling-group`

이러한 정책을 개별적으로 사용하거나 정책 목록으로 결합할 수 있습니다. 예를 들어, 다음 명령으로 Auto Scaling 그룹을 업데이트하여 먼저 `OldestLaunchConfiguration` 정책을 사용한 다음 `ClosestToNextInstanceHour` 정책을 사용합니다.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --termination-policies "OldestLaunchConfiguration" "ClosestToNextInstanceHour"
```

Default 종료 정책을 사용하는 경우 이 정책이 종료 정책 목록의 맨 마지막에 오도록 합니다. 예: `--termination-policies "OldestLaunchConfiguration" "Default"`.

인스턴스 축소 보호

축소 시 Auto Scaling 그룹에서 특정 인스턴스를 종료할지 여부를 제어하려면 인스턴스 축소 보호를 사용합니다. Auto Scaling 그룹 또는 개별 Auto Scaling 인스턴스의 인스턴스 축소 보호 설정을 활성화할 수 있습니다. Auto Scaling 그룹에서 인스턴스를 시작할 때 해당 인스턴스는 Auto Scaling 그룹의 인스턴스 축소 보호

설정을 상속합니다. Auto Scaling 그룹 또는 Auto Scaling 인스턴스의 인스턴스 축소 보호 설정을 언제든지 변경할 수 있습니다.

인스턴스 상태가 `InService`이면 인스턴스 축소 보호가 시작됩니다. 종료로부터 보호되는 인스턴스를 분리하면 인스턴스 축소 보호 설정이 손실됩니다. 인스턴스를 그룹에 다시 연결하면 해당 그룹의 현재 인스턴스 축소 보호 설정을 상속합니다.

축소 및 축소 이벤트 발생 시 Auto Scaling 그룹의 모든 인스턴스가 종료로부터 보호되는 경우, 원하는 용량을 줄입니다. 그러나 Auto Scaling 그룹은 인스턴스 축소 보호 설정이 비활성화될 때까지 필요한 수의 인스턴스를 종료하지 않습니다.

인스턴스 축소 보호는 다음 경우에 Auto Scaling 인스턴스를 보호하지 않습니다.

- Amazon EC2 콘솔, `terminate-instances` 명령 또는 `TerminateInstances` 작업을 통한 수동 종료. 수동 종료로부터 Auto Scaling 인스턴스를 보호하려면 Amazon EC2 종료 방지 기능을 활성화합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [종료 방지 기능 활성화](#)를 참조하십시오.
- 인스턴스가 상태 확인을 통과하지 못한 경우 상태 확인 교체. 자세한 내용은 [Auto Scaling 인스턴스 상태 확인 \(p. 148\)](#) 단원을 참조하십시오. Amazon EC2 Auto Scaling의 비정상 인스턴스 종료를 방지하려면 `ReplaceUnhealthy` 프로세스를 일시 중단하십시오. 자세한 내용은 [조정 프로세스의 일시 중지 및 재개 \(p. 142\)](#) 단원을 참조하십시오.
- 스팟 인스턴스 중단 스팟 인스턴스는 용량을 더 이상 사용할 수 없거나 스팟 가격이 최고가를 초과하는 경우 종료됩니다.

작업

- [그룹의 인스턴스 축소 보호 활성화 \(p. 129\)](#)
- [그룹의 인스턴스 축소 보호 설정 수정 \(p. 129\)](#)
- [인스턴스의 인스턴스 축소 보호 설정 수정 \(p. 130\)](#)

그룹의 인스턴스 축소 보호 활성화

Auto Scaling 그룹을 만들 때 인스턴스 축소 보호를 활성화할 수 있습니다. 기본적으로 인스턴스 축소 보호는 비활성화되어 있습니다.

인스턴스 축소 보호를 활성화하려면(새 콘솔)

Auto Scaling 그룹을 만들 때 그룹 크기 및 조정 정책 구성 페이지의 인스턴스 축소 보호에서 인스턴스 축소 보호 활성화 옵션을 선택합니다.

인스턴스 축소 보호를 활성화하려면(이전 콘솔)

Auto Scaling 그룹을 만드는 경우, Auto Scaling 그룹 세부 정보 구성 페이지에서 고급 세부 정보 아래의 인스턴스 보호에서 `Protect From Scale In` 옵션을 선택합니다.

인스턴스 축소 보호를 활성화하려면(AWS CLI)

다음 `create-auto-scaling-group` 명령을 사용하여 인스턴스 축소 보호를 활성화합니다.

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg --new-instances-protected-from-scale-in ...
```

그룹의 인스턴스 축소 보호 설정 수정

Auto Scaling 그룹의 인스턴스 축소 보호 설정을 활성화하거나 비활성화할 수 있습니다. 인스턴스 축소 보호 설정이 활성화되면 활성화 후 시작된 모든 새 인스턴스에는 인스턴스 축소 보호가 활성화됩니다. 이전에 시작된 인스턴스는 각 인스턴스에 대해 인스턴스 축소 보호 설정을 개별적으로 활성화하지 않는 한 축소로부터 보호되지 않습니다.

그룹의 인스턴스 축소 보호 설정을 변경하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. [Details] 탭에서 [Edit]를 선택합니다.
5. 고급 구성의 인스턴스 축소 보호에서 기본적으로 모든 새 인스턴스에서 인스턴스 축소 보호 활성화를 선택합니다. (이전 콘솔: 인스턴스 보호에서 축소 보호를 선택합니다.)
6. 저장을 선택합니다.

그룹의 인스턴스 축소 보호 설정을 변경하려면(AWS CLI)

다음 `update-auto-scaling-group` 명령을 사용하여 지정된 Auto Scaling 그룹의 인스턴스 축소 보호를 활성화합니다.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --new-instances-protected-from-scale-in
```

다음 명령을 사용하여 지정된 그룹의 인스턴트 축소 보호를 비활성화합니다.

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --no-new-instances-protected-from-scale-in
```

인스턴스의 인스턴스 축소 보호 설정 수정

기본적으로 인스턴스는 속해 있는 Auto Scaling 그룹의 인스턴스 축소 보호 설정을 가져옵니다. 그러나 인스턴스의 인스턴스 축소 보호를 언제든지 활성화하거나 비활성화할 수 있습니다.

인스턴스의 인스턴스 축소 보호 설정을 변경하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 인스턴스 관리 탭의 인스턴스에서 인스턴스를 선택합니다. (이전 콘솔: 인스턴스 탭에서 인스턴스를 선택할 수 있습니다.)
5. 인스턴스 축소 보호를 활성화하려면 작업, 축소 보호 설정을 선택합니다. 메시지가 표시되면 축소 보호 설정을 선택합니다.
6. 인스턴스 축소 보호를 비활성화하려면 작업, 축소 보호 제거를 선택합니다. 메시지가 표시되면 축소 보호 제거를 선택합니다.

인스턴스의 인스턴스 축소 보호 설정을 변경하려면(AWS CLI)

다음 `set-instance-protection` 명령을 사용하여 지정된 인스턴스의 인스턴스 축소 보호를 활성화합니다.

```
aws autoscaling set-instance-protection --instance-ids i-5f2e8a0d --auto-scaling-group-name my-asg --protected-from-scale-in
```

다음 명령을 사용하여 지정된 인스턴스의 인스턴스 축소 보호를 비활성화합니다.

```
aws autoscaling set-instance-protection --instance-ids i-5f2e8a0d --auto-scaling-group-name my-asg --no-protected-from-scale-in
```

Amazon EC2 Auto Scaling 수명 주기 후크

수명 주기 후크를 사용하면 Auto Scaling 그룹에서 인스턴스를 시작하거나 종료할 때 인스턴스를 일시 중지하여 사용자 지정 작업을 수행할 수 있습니다. 인스턴스가 일시 중지되는 경우, `complete-lifecycle-action` 명령 또는 `CompleteLifecycleAction` 작업을 사용하여 수명 주기 작업을 완료할 때까지 혹은 제한 시간이 끝날 때까지(기본 1시간) 대기 상태로 유지됩니다.

예를 들어 새로 시작된 인스턴스가 시작 시퀀스를 완료하고 수명 주기 후크가 인스턴스를 일시 중지한다고 가정해 보겠습니다. 인스턴스가 대기 상태에 있는 동안 인스턴스에 소프트웨어를 설치하거나 구성하여 트래픽 수신을 시작하기 전에 인스턴스가 준비를 마치도록 합니다. 수명 주기 후크 사용의 또 다른 예로, 축소 이벤트가 발생하는 경우 종료 인스턴스가 먼저 로드 밸런서에서 등록 취소됩니다(Auto Scaling 그룹이 탄력적 로드 밸런싱과 함께 사용되는 경우). 그러면 인스턴스가 종료되기 전에 수명 주기 후크가 인스턴스를 일시 중지합니다. 예를 들어, 인스턴스가 대기 상태에 있는 동안 인스턴스가 완전히 종료되기 전에 인스턴스에 연결하여 로그 또는 다른 데이터를 다운로드할 수 있습니다.

각 Auto Scaling 그룹은 다수의 수명 주기 후크를 보유할 수 있습니다. 그러나 Auto Scaling 그룹당 보유할 수 있는 후크 수에 제한이 있습니다. 자세한 내용은 [Amazon EC2 Auto Scaling 서비스 할당량 \(p. 9\)](#) 단원을 참조하십시오.

목차

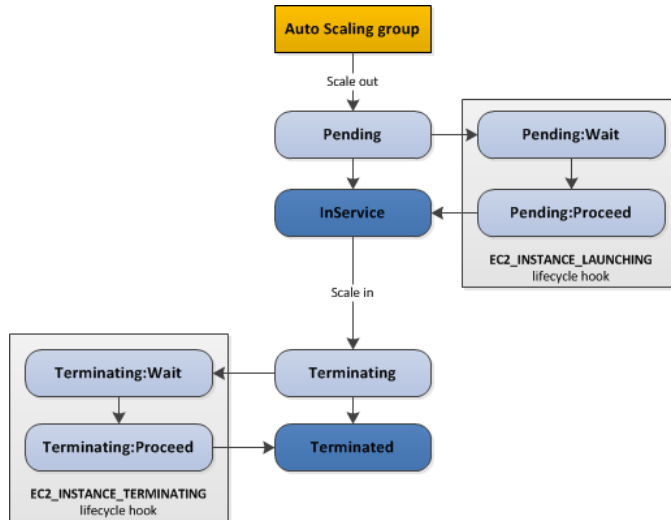
- 수명 주기 후크 작동 방식 (p. 131)
- 고려 사항 (p. 132)
- 알림 준비 (p. 133)
- 수명 주기 후크 추가 (p. 133)
- 수명 주기 후크 사용자 지정 작업 완료 (p. 134)
- 알림 테스트 (p. 135)
- Amazon EC2 Auto Scaling 수명 주기 후크에 대한 알림 구성 (p. 135)

수명 주기 후크 작동 방식

Auto Scaling 그룹에 수명 주기 후크를 추가하면 다음과 같이 작동합니다.

1. Auto Scaling 그룹이 인스턴스를 시작하여 확장 이벤트에 응답하고, 인스턴스를 종료하여 축소 이벤트에 응답합니다.
2. 수명 주기 후크는 인스턴스를 대기 상태(`Pending:Wait` 또는 `Terminating:Wait`)로 설정합니다. 인스턴스는 사용자가 계속 진행하거나 제한 시간이 종료될 때까지 일시 중지 상태로 유지됩니다.
3. 다음 옵션 중 하나 이상을 사용하여 사용자 지정 작업을 수행할 수 있습니다.
 - 수명 주기 작업이 발생할 때 Lambda 함수를 호출하도록 CloudWatch 이벤트를 정의합니다. Amazon EC2 Auto Scaling에서 CloudWatch 이벤트에 수명 주기 작업에 대한 이벤트를 제출하면 Lambda 함수가 호출됩니다. 이 이벤트에는 시작되거나 종료되는 인스턴스에 대한 정보와 수명 주기 작업을 제어하는 데 사용할 수 있는 토큰이 포함되어 있습니다.
 - 수명 주기 후크에 대한 알림 대상을 정의합니다. 그러면 Amazon EC2 Auto Scaling에서 알림 대상에 메시지를 보냅니다. 이 메시지에는 시작되거나 종료되는 인스턴스에 대한 정보와 수명 주기 작업을 제어하는 데 사용할 수 있는 토큰이 포함되어 있습니다.
 - 인스턴스를 시작할 때 인스턴스에서 실행되는 스크립트를 만듭니다. 스크립트는 실행되는 인스턴스의 ID를 사용하여 수명 주기 작업을 제어할 수 있습니다.
4. 기본적으로 인스턴스는 한 시간 동안 대기 상태로 유지됩니다. 그 후 Auto Scaling 그룹에서 시작 또는 종료 프로세스(`Pending:Proceed` 또는 `Terminating:Proceed`)를 진행합니다. 시간이 더 필요한 경우 하트비트를 기록하여 제한 시간을 다시 시작할 수 있습니다. 제한 시간이 종료되기 전에 마칠 경우 수명 주기 작업을 완료하면 시작 또는 종료 프로세스를 진행할 수 있습니다.

다음 그림에서는 이 프로세스에서 인스턴스 상태 간 전환을 보여 줍니다.



Auto Scaling 그룹에서 인스턴스의 전체 수명 주기에 대한 자세한 내용은 [Auto Scaling 수명 주기 \(p. 7\)](#) 단원을 참조하십시오.

고려 사항

Auto Scaling 그룹에 수명 주기 후크를 추가하면 인스턴스를 시작하고 종료하는 방식을 보다 강력하게 제어할 수 있습니다. 수명 주기 후크를 Auto Scaling 그룹에 추가하는 경우, 그룹이 계속해서 예상대로 작동되도록 하려면 몇 가지 고려할 사항이 있습니다.

인스턴스를 대기 상태로 유지

인스턴스는 한정된 시간 동안 대기 상태를 유지할 수 있습니다. 기본값은 1시간(3600초)입니다. 다음 방법을 통해 이 시간을 조절할 수 있습니다.

- 수명 주기 후크를 만들 때 수명 주기 후크의 하트비트 제한 시간을 설정합니다. `put-lifecycle-hook` 명령과 `--heartbeat-timeout` 파라미터를 사용합니다. `PutLifecycleHook` 작업에서 `HeartbeatTimeout` 파라미터를 사용합니다.
- 제한 시간이 종료되기 전에 마칠 경우 `complete-lifecycle-action` 명령 또는 `CompleteLifecycleAction` 작업을 사용하여 다음 상태로 진행합니다.
- `record-lifecycle-action-heartbeat` 명령 또는 `RecordLifecycleActionHeartbeat` 작업을 사용하여 하트비트를 기록함으로써 제한 시간 종료를 연장합니다. 그러면 수명 주기 후크를 만들 때 지정한 제한 시간을 기준으로 제한 시간이 늘어납니다. 예를 들어 제한 시간 값이 1시간이고 이 명령을 30분 후에 호출한 경우, 인스턴스는 추가 1시간 동안 대기 상태로 유지됩니다(총 90분).

인스턴스를 대기 상태로 유지할 수 있는 최대 시간은 48시간 혹은 하트비트 제한 시간 100회 중 적은 쪽입니다.

휴지 및 단순 조정

단순 조정 정책으로 인해 Auto Scaling 그룹에서 인스턴스를 시작하거나 종료하면 [휴지 \(p. 123\)](#)가 적용됩니다. 휴지 기간은 Auto Scaling 그룹이 필요한 것보다 많은 인스턴스를 시작하거나 종료하지 않도록 합니다. 수명 주기 작업이 발생하고 인스턴스가 대기 상태로 전환되면 단순 조정 정책으로 인한 조정 활동이 일시 중지됩니다. 인스턴스가 `InService` 상태로 들어가면 휴지 기간이 시작됩니다. 휴지 기간이 만료되면 일시 중지된 조정 활동이 재개됩니다.

상태 확인 유예 기간

수명 주기 후크를 추가할 경우 상태 확인 유예 기간은 수명 주기 후크 작업이 완료되고 인스턴스가 InService 상태로 전환되기까지 시작되지 않습니다.

스팟 인스턴스

스팟 인스턴스와 함께 수명 주기 후크를 사용할 수 있습니다. 그러나 수명 주기 후크는 용량을 더 이상 사용할 수 없는 경우 인스턴스 종료를 막지 않습니다. 또한 스팟 인스턴스가 종료될 때 수명 주기 작업을 완료해야 합니다(complete-lifecycle-action 명령 또는 CompleteLifecycleAction 작업 사용).

알림 준비

인스턴스가 대기 상태로 전환되는 것에 대한 알림을 구성할 수 있습니다. Amazon CloudWatch Events, Amazon SNS 또는 Amazon SQS를 사용하여 알림을 받을 수 있습니다. 자세한 내용은 [수명 주기 후크 알림 구성](#) (p. 135) 단원을 참조하십시오.

혹은 인스턴스가 시작될 때 인스턴스를 구성하는 스크립트가 있는 경우, 수명 주기 작업이 발생할 때 알림을 받을 필요가 없습니다. 아직 없는 경우 인스턴스 메타데이터에서 인스턴스의 인스턴스 ID를 검색하도록 스크립트를 업데이트합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스 메타데이터 검색](#)을 참조하십시오.

수명 주기 후크 추가

Auto Scaling 그룹에 수명 주기 후크를 추가할 때 Auto Scaling 그룹에서 인스턴스가 시작되거나 종료되는 시점에 수명 주기 후크를 실행할지 여부를 지정할 수 있습니다.

내용

- [수명 주기 후크 추가\(콘솔\)](#) (p. 133)
- [수명 주기 후크 추가\(AWS CLI\)](#) (p. 134)

수명 주기 후크 추가(콘솔)

기존 Auto Scaling 그룹에 수명 주기 후크를 추가하려면 다음 단계를 수행합니다. 인스턴스가 시작되거나 종료될 때 후크를 사용할지 여부와 프로세스를 중단하거나 계속하기 전에 수명 주기 후크가 완료될 때까지 기다리는 시간을 지정할 수 있습니다.

수명 주기 후크를 추가하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 인스턴스 관리 탭의 Lifecycle hooks(수명 주기 후크)에서 Create lifecycle hook(수명 주기 후크 생성)을 선택합니다. (이전 콘솔: Lifecycle Hooks(수명 주기 후크) 탭에서 수명 주기 후크를 생성할 수 있습니다.)
5. 수명 주기 후크를 정의하려면 다음을 수행합니다.
 - a. Lifecycle hook name(수명 주기 후크 이름)에 수명 주기 후크의 이름을 지정합니다.
 - b. Lifecycle transition(수명 주기 전환)에서 Instance launch(인스턴스 시작) 또는 Instance terminate(인스턴스 종료)를 선택합니다.
 - c. 인스턴스가 대기 상태로 유지되는 시간을 제어하도록 Heartbeat timeout(하트비트 제한 시간)에 제한 시간 값을 지정합니다. 값은 30초~7200초여야 합니다. 제한 시간 동안에는 예를 들어 새로 시작된 인스턴스에 로그인하고 애플리케이션을 설치하거나 사용자 지정 작업을 수행할 수 있습니다.

- d. Default result(기본 결과)에 수명 주기 후크 제한 시간이 경과하거나 예기치 못한 오류가 발생할 때 Auto Scaling 그룹에서 수행하는 작업을 지정합니다. [ABANDON(중단)] 또는 [CONTINUE(계속)]을 선택할 수 있습니다.

인스턴스가 시작되는 경우, [CONTINUE(계속)]는 작업이 성공적임을 나타내며 Auto Scaling 그룹에서 인스턴스를 서비스 상태로 전환합니다. 그에 반해 ABANDON(중단)은 사용자 지정 작업이 실패했음을 나타내며, Auto Scaling이 인스턴스를 종료할 수 있습니다. 인스턴스가 종료되는 경우, [ABANDON(중단)] 및 [CONTINUE(계속)] 모두를 통해 인스턴스를 종료할 수 있습니다. 그러나 [ABANDON(중단)]은 수명 주기 후크와 같은 남아 있는 모든 작업을 중지하는 반면, [CONTINUE(계속)]는 다른 모든 수명 주기 후크를 완료합니다.
 - e. (선택 사항) Notification metadata(알림 메타데이터)에 Amazon EC2 Auto Scaling이 알림 대상에 메시지를 전송할 때마다 포함할 추가 정보를 지정합니다.
6. 생성을 선택합니다.

수명 주기 후크 추가(AWS CLI)

`put-lifecycle-hook` 명령을 사용하여 수명 주기 후크를 생성하고 업데이트합니다.

확장 시 작업을 수행하려면 다음 명령을 사용합니다.

```
aws autoscaling put-lifecycle-hook --lifecycle-hook-name my-hook --auto-scaling-group-name my-asg \
  --lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING
```

축소 시 작업을 수행하려면 그 대신 다음 명령을 사용합니다.

```
aws autoscaling put-lifecycle-hook --lifecycle-hook-name my-hook --auto-scaling-group-name my-asg \
  --lifecycle-transition autoscaling:EC2_INSTANCE_TERMINATING
```

Amazon SNS 또는 Amazon SQS를 사용하여 알림을 받으려면 알림 대상 및 IAM 역할을 지정해야 합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 수명 주기 후크에 대한 알림 구성](#) (p. 135) 단원을 참조하십시오.

예를 들어 SNS 주제를 알림 대상으로 지정하려면 다음 옵션을 추가합니다.

```
--notification-target-arn arn:aws:sns:region:123456789012:my-sns-topic --role-arn arn:aws:iam::123456789012:role/my-notification-role
```

이 주제는 다음 키/값 쌍에 해당하는 테스트 알림을 보냅니다.

```
"Event": "autoscaling:TEST_NOTIFICATION"
```

수명 주기 후크 사용자 지정 작업 완료

Auto Scaling 그룹에서 확장 또는 축소 이벤트에 응답할 때 인스턴스를 대기 상태로 설정하고 수명 주기 후크가 구성된 방식에 따라 알림을 전송합니다.

수명 주기 후크 사용자 지정 작업을 완료하려면

1. 알림을 받은 후 인스턴스가 대기 상태에 있는 동안 사용자 지정 작업을 수행할 수 있습니다.
2. (선택 사항) 사용자 지정 작업을 완료할 시간이 더 필요한 경우, `record-lifecycle-action-heartbeat` 명령을 사용하여 제한 시간을 재시작하고 인스턴스를 대기 상태로 유지합니다. 다음 명령과 같이 알림에서 받은 수명 주기 작업 토큰을 지정할 수 있습니다.

```
aws autoscaling record-lifecycle-action-heartbeat --lifecycle-hook-name my-launch-hook \
--auto-scaling-group-name my-asg --lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635
```

또는 다음 명령과 같이 이전 단계에서 검색한 인스턴스의 ID를 지정할 수도 있습니다.

```
aws autoscaling record-lifecycle-action-heartbeat --lifecycle-hook-name my-launch-hook \
--auto-scaling-group-name my-asg --instance-id i-1a2b3c4d
```

- 제한 시간이 종료되기 전에 사용자 지정 작업을 마치는 경우, Auto Scaling 그룹에서 인스턴스를 계속 시작하거나 종료할 수 있도록 `complete-lifecycle-action` 명령을 사용합니다. 다음 명령과 같이 수명 주기 작업 토큰을 지정할 수 있습니다.

```
aws autoscaling complete-lifecycle-action --lifecycle-action-result CONTINUE \
--lifecycle-hook-name my-launch-hook --auto-scaling-group-name my-asg \
--lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635
```

또는 다음 명령과 같이 인스턴스의 ID를 지정할 수도 있습니다.

```
aws autoscaling complete-lifecycle-action --lifecycle-action-result CONTINUE \
--instance-id i-1a2b3c4d --lifecycle-hook-name my-launch-hook \
--auto-scaling-group-name my-asg
```

알림 테스트

시작 이벤트에 대해 알림이 발생하도록 하려면 Auto Scaling 그룹의 용량을 1만큼 늘려 Auto Scaling 그룹을 업데이트합니다. 인스턴스 시작 후 몇 분 내에 알림을 받게 됩니다.

원하는 용량을 변경하려면(콘솔)

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
- 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
- Auto Scaling 그룹을 선택합니다.
- [Details] 탭에서 [Edit]를 선택합니다.
- 원하는 용량에 대해 현재 값을 1씩 늘립니다. 이 값이 최대 용량을 초과하는 경우 최대 용량 값도 1씩 늘려야 합니다.
- Save를 선택합니다.
- 몇 분 후 이벤트 알림이 전송됩니다. 이 테스트에서 시작한 인스턴스가 추가로 필요하지 않으면 원하는 용량을 1만큼 줄일 수 있습니다. 몇 분 후 이벤트 알림이 전송됩니다.

Amazon EC2 Auto Scaling 수명 주기 후크에 대한 알림 구성

인스턴스가 대기 상태로 전환될 때 알림을 트리거하는 수명 주기 후크를 Auto Scaling 그룹에 추가할 수 있습니다. Lambda 함수를 호출하거나 사용자 지정 작업을 수행할 수 있도록 이메일 알림을 수신하는 것과 같은 다양한 이유로 이러한 알림을 구성할 수 있습니다. 이 주제에서는 Amazon CloudWatch Events, Amazon SNS 및 Amazon SQS를 사용하여 알림을 구성하는 방법을 설명합니다. 선호하는 옵션을 선택하십시오. 혹은

은 인스턴스가 시작될 때 인스턴스를 구성하는 스크립트가 있는 경우, 수명 주기 작업이 발생할 때 알림을 받을 필요가 없습니다.

Important

알림을 위한 AWS 리소스는 항상 수명 주기 후크를 생성하는 것과 동일한 AWS 리전에서 생성해야 합니다. 예를 들어 Amazon SNS를 사용하여 알림을 구성하는 경우, Amazon SNS 주제는 수명 주기 후크와 동일한 리전에 있어야 합니다.

알림 옵션

- [CloudWatch 이벤트를 사용하여 Lambda로 알림 라우팅 \(p. 136\)](#)
- [Amazon SNS를 이용해 알림 받기 \(p. 137\)](#)
- [Amazon SQS를 이용해 알림 받기 \(p. 138\)](#)

CloudWatch 이벤트를 사용하여 Lambda로 알림 라우팅

수명 주기 작업이 발생할 때 Lambda 함수를 호출하도록 CloudWatch 이벤트를 이용해 대상을 설정할 수 있습니다.

CloudWatch 이벤트를 이용해 알림을 설정하려면

1. [Lambda 함수 만들기 \(p. 159\)](#)의 단계를 사용하여 Lambda 함수를 만들고 Amazon 리소스 이름(ARN)을 적어둡니다. 예: `arn:aws:lambda:region:123456789012:function:my-function`.
2. 다음 `put-rule` 명령을 사용하여 수명 주기 작업에 맞는 CloudWatch 이벤트 규칙을 만듭니다.

```
aws events put-rule --name my-rule --event-pattern file://pattern.json --state ENABLED
```

다음 예제에서는 인스턴스 시작 수명 주기 작업을 위한 `pattern.json`을 보여 줍니다.

```
{
  "source": [ "aws.autoscaling" ],
  "detail-type": [ "EC2 Instance-launch Lifecycle Action" ]
}
```

다음 예제에서는 인스턴스 종료 수명 주기 작업을 위한 `pattern.json`을 보여 줍니다.

```
{
  "source": [ "aws.autoscaling" ],
  "detail-type": [ "EC2 Instance-terminate Lifecycle Action" ]
}
```

3. 다음 `add-permission` 명령을 사용하여 규칙에 Lambda 함수를 호출할 수 있는 권한을 부여합니다. 이 명령은 CloudWatch 이벤트 서비스 보안 주체(`events.amazonaws.com`)를 신뢰할 수 있게 하고, 지정된 규칙에 따라 권한의 범위를 지정합니다.

```
aws lambda add-permission --function-name LogScheduledEvent --statement-id my-scheduled-event \
  --action 'lambda:InvokeFunction' --principal events.amazonaws.com --source-arn
arn:aws:events:region:123456789012:rule/my-scheduled-rule
```

4. 다음 `put-targets` 명령을 사용하여 수명 주기 작업이 발생할 때 Lambda 함수를 호출하는 대상을 생성합니다.

```
aws events put-targets --rule my-rule --targets
Id=1,Arn=arn:aws:lambda:region:123456789012:function:my-function
```


- 이 지침을 따른 후에는 다음 단계 [수명 주기 후크 추가](#) (p. 133)로 계속 진행합니다.

Auto Scaling 그룹에서 확장 또는 축소 이벤트에 응답할 때 인스턴스를 대기 상태로 설정합니다. 인스턴스가 대기 상태에 있는 동안 Lambda 함수가 호출됩니다. 이벤트 데이터에 대한 자세한 내용은 [Auto Scaling 이벤트](#) (p. 155) 단원을 참조하십시오.

Amazon SNS를 이용해 알림 받기

수명 주기 작업이 발생할 때 알림을 받도록 Amazon SNS를 이용해 알림 대상을 설정합니다.

Amazon SNS를 이용해 알림을 설정하려면

- 다음 `create-topic` 명령을 사용하여 Amazon SNS 주제를 생성합니다. 자세한 내용은 Amazon Simple Notification Service 개발자 안내서의 [주제 생성](#)을 참조하십시오.

```
aws sns create-topic --name my-sns-topic
```

대상의 ARN(예: `arn:aws:sns:region:123456789012:my-sns-topic`)을 적어둡니다.

- Amazon EC2 Auto Scaling에 대해 알림 대상에 액세스하는 권한을 부여할 수 있는 서비스 역할(수임 역할)을 생성합니다.

IAM 역할을 생성하고 Amazon EC2 Auto Scaling에 해당 역할을 맡기려면

- <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
- 탐색 창에서 Roles(역할), Create new role(새 역할 생성)을 선택합니다.
- Select type of trusted entity(신뢰할 수 있는 유형의 엔터티 선택) 아래에서 AWS service(AWS 서비스)를 선택합니다.
- Choose the service that will use this role(이 역할을 사용할 서비스 선택) 아래에 있는 목록에서 EC2 Auto Scaling을 선택합니다.
- Select your use case(사용 사례 선택) 하단에서 EC2 Auto Scaling Notification Access (EC2 Auto Scaling 알림)을 선택한 후, Next:Permissions(다음:권한)를 선택합니다.
- 다음: 태그를 선택합니다. 선택적으로 태그를 키-값 페어로 연결하여 메타데이터를 역할에 추가할 수 있습니다. 그런 다음 다음: 검토를 선택합니다.
- 검토 페이지에서 역할의 이름(예: my-notification-role)을 입력하고 역할 만들기를 선택합니다.
- 역할 페이지에서 방금 만든 역할을 선택하여 요약 페이지를 엽니다. 역할 ARN을 기록해 둡니다. 예: `arn:aws:iam::123456789012:role/my-notification-role`. 다음 절차에서 수명 주기 후크를 생성하는 경우 역할 ARN을 지정합니다.

- 이 지침을 따른 후에는 다음 단계 [수명 주기 후크 추가\(AWS CLI\)](#) (p. 134)로 계속 진행합니다.

Auto Scaling 그룹에서 확장 또는 축소 이벤트에 응답할 때 인스턴스를 대기 상태로 설정합니다. 인스턴스가 대기 상태에 있는 동안 알림 대상에게 메시지가 전송됩니다. 메시지에 포함되는 이벤트 데이터는 다음과 같습니다.

- `LifecycleActionToken` - 수명 주기 작업 토큰.
- `AccountId` - AWS 계정 ID.
- `AutoScalingGroupName` - Auto Scaling 그룹 이름.
- `LifecycleHookName` - 수명 주기 후크 이름.
- `EC2InstanceId` - EC2 인스턴스의 ID.
- `LifecycleTransition` - 수명 주기 후크 유형.

다음은 알림 메시지 예입니다.

```
Service: AWS Auto Scaling
Time: 2019-04-30T20:42:11.305Z
RequestId: 18b2ec17-3e9b-4c15-8024-ff2e8ce8786a
LifecycleActionToken: 71514b9d-6a40-4b26-8523-05e7ee35fa40
AccountId: 123456789012
AutoScalingGroupName: my-asg
LifecycleHookName: my-hook
EC2InstanceId: i-0598c7d356eba48d7
LifecycleTransition: autoscaling:EC2_INSTANCE_LAUNCHING
NotificationMetadata: null
```

Amazon SQS를 이용해 알림 받기

수명 주기 작업이 발생할 때 알림을 받도록 Amazon SQS를 이용해 알림 대상을 설정합니다.

Important

FIFO 대기열은 수명 주기 후크와 호환되지 않습니다.

Amazon SQS를 이용해 알림을 설정하려면

1. Amazon SQS를 사용하여 대상을 만듭니다. 자세한 내용은 Amazon Simple Queue Service 개발자 안내서의 [Amazon SQS 시작하기](#)를 참조하십시오. 대상의 ARN(예: `arn:aws:sqs:region:123456789012:my-sqs-queue`)을 적어둡니다.
2. Amazon EC2 Auto Scaling에 대해 알림 대상에 액세스하는 권한을 부여할 수 있는 서비스 역할(수임 역할)을 생성합니다.

IAM 역할을 생성하고 Amazon EC2 Auto Scaling에 해당 역할을 맡기려면

- a. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
 - b. 탐색 창에서 Roles(역할), Create new role(새 역할 생성)을 선택합니다.
 - c. Select type of trusted entity(신뢰할 수 있는 유형의 엔터티 선택) 아래에서 AWS service(AWS 서비스)를 선택합니다.
 - d. Choose the service that will use this role(이 역할을 사용할 서비스 선택) 아래에 있는 목록에서 EC2 Auto Scaling을 선택합니다.
 - e. Select your use case(사용 사례 선택) 하단에서 EC2 Auto Scaling Notification Access (EC2 Auto Scaling 알림)을 선택한 후, Next:Permissions(다음:권한)를 선택합니다.
 - f. 다음: 태그를 선택합니다. 선택적으로 태그를 키-값 페어로 연결하여 메타데이터를 역할에 추가할 수 있습니다. 그런 다음 다음: 검토를 선택합니다.
 - g. 검토 페이지에서 역할의 이름(예: my-notification-role)을 입력하고 역할 만들기를 선택합니다.
 - h. 역할 페이지에서 방금 만든 역할을 선택하여 요약 페이지를 엽니다. 역할 ARN을 기록해 둡니다. 예: `arn:aws:iam::123456789012:role/my-notification-role`. 다음 절차에서 수명 주기 후크를 생성하는 경우 역할 ARN을 지정합니다.
3. 이 지침을 따른 후에는 다음 단계 [수명 주기 후크 추가\(AWS CLI\)](#) (p. 134)로 계속 진행합니다.

Auto Scaling 그룹에서 확장 또는 축소 이벤트에 응답할 때 인스턴스를 대기 상태로 설정합니다. 인스턴스가 대기 상태에 있는 동안 알림 대상에게 메시지가 전송됩니다.

Auto Scaling 그룹에서 일시적으로 인스턴스 제거

InService 상태의 인스턴스를 Standby 상태로 설정하고, 인스턴스를 업데이트하거나 문제 해결한 다음, 해당 인스턴스를 서비스 상태로 되돌릴 수 있습니다. 대기 상태의 인스턴스는 Auto Scaling 그룹에 여전히 속하나, 애플리케이션 트래픽을 처리하지는 못합니다.

Important

대기 상태의 인스턴스에 대해 요금이 청구됩니다.

예를 들어, Auto Scaling 그룹의 시작 구성을 언제든지 변경할 수 있으며, Auto Scaling 그룹에서 시작하는 이후의 모든 인스턴스는 이 구성을 사용하게 됩니다. 그러나 Auto Scaling 그룹은 현재 서비스 중인 인스턴스를 업데이트하지 않습니다. 이러한 인스턴스를 종료하고 Auto Scaling 그룹이 해당 인스턴스를 대체하도록 할 수 있습니다. 또는 인스턴스 상태를 대기로 설정하고 소프트웨어를 업데이트한 후 인스턴스를 다시 서비스 상태로 설정할 수 있습니다.

목차

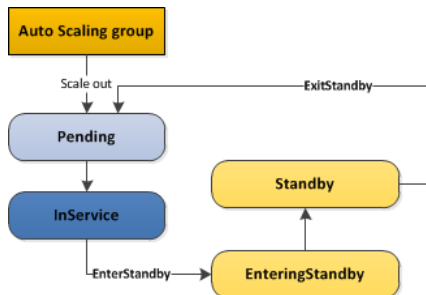
- 대기 상태를 작동하는 방법 (p. 139)
- 대기 상태의 인스턴스 상태 (p. 140)
- 인스턴스를 일시적으로 제거(콘솔) (p. 140)
- 인스턴스를 일시적으로 제거(AWS CLI) (p. 140)

대기 상태를 작동하는 방법

대기 상태는 Auto Scaling 그룹에서 인스턴스를 일시적으로 제거할 수 있도록 다음과 같이 작동합니다.

1. 인스턴스를 대기 상태로 설정합니다. 인스턴스는 대기 상태를 끝낼 때까지 이 상태로 유지됩니다.
2. Auto Scaling 그룹에 연결된 로드 밸런서 또는 대상 그룹이 있는 경우, 로드 밸런서 또는 대상 그룹에서 해당 인스턴스가 등록 취소됩니다.
3. 기본적으로 인스턴스를 대기 상태로 설정하면 희망 용량으로 지정한 값은 감소합니다. 그러면 이 인스턴스가 대기 상태에 있는 동안 추가 인스턴스를 시작하지 않도록 합니다. 또는 용량이 감소되지 않도록 지정할 수 있습니다. 이 옵션을 지정하는 경우, Auto Scaling 그룹이 인스턴스를 시작하고 대기 상태에 있는 인스턴스를 대체합니다. 하나 이상의 인스턴스가 대기 상태에 있는 동안 애플리케이션의 용량을 유지하기 위해서입니다.
4. 인스턴스를 업데이트하거나 문제를 해결할 수 있습니다.
5. 대기 상태를 끝내 인스턴스를 서비스 상태로 되돌립니다.
6. 대기 상태였던 인스턴스를 다시 서비스 상태로 설정한 이후 원하는 용량이 증가합니다. 인스턴스를 대기 상태로 설정할 때 용량을 줄이지 않은 경우, Auto Scaling 그룹은 필요 이상의 인스턴스가 있음을 감지합니다. 그런 다음 종료 정책을 적용하여 그룹의 크기를 줄입니다. 자세한 내용은 [축소 시 Auto Scaling 인스턴스 종료 제어 \(p. 126\)](#) 단원을 참조하십시오.
7. Auto Scaling 그룹에 연결된 로드 밸런서 또는 대상 그룹이 있는 경우, 해당 인스턴스가 로드 밸런서 또는 대상 그룹에 등록됩니다.

다음 그림에서는 이 프로세스에서 인스턴스 상태 간 전환을 보여 줍니다.



Auto Scaling 그룹에서 인스턴스의 전체 수명 주기에 대한 자세한 내용은 [Auto Scaling 수명 주기 \(p. 7\)](#) 단원을 참조하십시오.

대기 상태의 인스턴스 상태

Amazon EC2 Auto Scaling은 대기 상태의 인스턴스에 대한 상태 확인을 수행하지 않습니다. 인스턴스가 대기 상태에 있는 동안 이 인스턴스의 상태는 대기 상태로 설정되기 전의 상태를 반영합니다. 인스턴스를 서비스 상태로 설정할 때까지 Amazon EC2 Auto Scaling은 해당 인스턴스에 대한 상태 확인을 수행하지 않습니다.

예를 들어, 정상인 인스턴스를 대기 상태로 설정한 후 종료하면 Amazon EC2 Auto Scaling은 해당 인스턴스를 정상으로 계속 보고합니다. 대기 상태에 있던 종료된 인스턴스를 다시 서비스 상태로 전환하려면, Amazon EC2 Auto Scaling은 인스턴스에 대한 상태 확인을 수행하고, 인스턴스가 종료 중이거나 비정상적으로 판단되면 대체 인스턴스를 시작합니다.

인스턴스를 일시적으로 제거(콘솔)

다음 절차는 현재 서비스 중인 인스턴스를 업데이트하는 일반적인 프로세스를 보여 줍니다.

인스턴스를 일시적으로 제거하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. 인스턴스 관리 탭의 인스턴스에서 인스턴스를 선택합니다. (이전 콘솔: 인스턴스 탭에서 인스턴스를 선택할 수 있습니다.)
5. [Actions], [Set to Standby]를 선택합니다.
6. Set to Standby(대기로 설정) 페이지에서 확인란을 선택하여 교체 인스턴스를 시작합니다. 혹은 확인란을 선택하지 않고 원하는 만큼 용량을 줄입니다. [Set to Standby]를 선택합니다.
7. 필요에 따라 인스턴스를 업데이트하거나 문제 해결할 수 있습니다. 모두 마쳤으면 다음 단계를 계속하여 인스턴스를 서비스 상태로 되돌립니다.
8. 인스턴스를 선택하고 [Actions], [Set to InService]를 선택합니다. Set to InService(서비스 중으로 설정) 대화 상자에서 Set to InService(서비스 중으로 설정)를 선택합니다.

인스턴스를 일시적으로 제거(AWS CLI)

다음 절차는 현재 서비스 중인 인스턴스를 업데이트하는 일반적인 프로세스를 보여 줍니다.

인스턴스를 일시적으로 제거하려면

1. 다음 `describe-auto-scaling-instances` 명령을 사용하여 업데이트할 인스턴스를 식별합니다.

```
aws autoscaling describe-auto-scaling-instances
```

다음은 응답의 예입니다.

```
{
  "AutoScalingInstances": [
    {
      "ProtectedFromScaleIn": false,
      "AvailabilityZone": "us-west-2a",
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      }
    }
  ]
}
```

```
    },
    "InstanceId": "i-05b4f7d5be44822a6",
    "AutoScalingGroupName": "my-asg",
    "HealthStatus": "HEALTHY",
    "LifecycleState": "InService"
  },
  ...
]
}
```

2. 다음 `enter-standby` 명령을 사용하여 인스턴스를 Standby 상태로 바꿉니다. `--should-decrement-desired-capacity` 옵션을 사용하면 원하는 용량을 줄이므로 Auto Scaling 그룹에서는 대체 인스턴스를 시작하지 않습니다.

```
aws autoscaling enter-standby --instance-ids i-05b4f7d5be44822a6 \
--auto-scaling-group-name my-asg --should-decrement-desired-capacity
```

다음은 응답의 예입니다.

```
{
  "Activities": [
    {
      "Description": "Moving EC2 instance to Standby: i-05b4f7d5be44822a6",
      "AutoScalingGroupName": "my-asg",
      "ActivityId": "3b1839fe-24b0-40d9-80ae-bcd883c2be32",
      "Details": "{\"Availability Zone\":\"us-west-2a\"}",
      "StartTime": "2014-12-15T21:31:26.150Z",
      "Progress": 50,
      "Cause": "At 2014-12-15T21:31:26Z instance i-05b4f7d5be44822a6 was moved to
standby
      in response to a user request, shrinking the capacity from 4 to 3.",
      "StatusCode": "InProgress"
    }
  ]
}
```

3. (선택 사항) 다음 Standby 명령을 사용하여 인스턴스가 `describe-auto-scaling-instances` 상태가 되었는지 확인합니다.

```
aws autoscaling describe-auto-scaling-instances --instance-ids i-05b4f7d5be44822a6
```

다음은 응답의 예입니다. 인스턴스가 현재 Standby 상태임을 확인할 수 있습니다.

```
{
  "AutoScalingInstances": [
    {
      "ProtectedFromScaleIn": false,
      "AvailabilityZone": "us-west-2a",
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      },
      "InstanceId": "i-05b4f7d5be44822a6",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "Standby"
    },
    ...
  ]
}
```

- 필요에 따라 인스턴스를 업데이트하거나 문제 해결할 수 있습니다. 모두 마쳤으면 다음 단계를 계속하여 인스턴스를 서비스 상태로 되돌립니다.
- 다음 `exit-standby` 명령을 사용하여 인스턴스를 다시 서비스 상태로 설정합니다.

```
aws autoscaling exit-standby --instance-ids i-05b4f7d5be44822a6 --auto-scaling-group-name my-asg
```

다음은 응답의 예입니다.

```
{
  "Activities": [
    {
      "Description": "Moving EC2 instance out of Standby: i-05b4f7d5be44822a6",
      "AutoScalingGroupName": "my-asg",
      "ActivityId": "db12b166-cdcc-4c54-8aac-08c5935f8389",
      "Details": "{\"Availability Zone\":\"us-west-2a\"}",
      "StartTime": "2014-12-15T21:46:14.678Z",
      "Progress": 30,
      "Cause": "At 2014-12-15T21:46:14Z instance i-05b4f7d5be44822a6 was moved out of standby in response to a user request, increasing the capacity from 3 to 4.",
      "StatusCode": "PreInService"
    }
  ]
}
```

- (선택 사항) 다음 `describe-auto-scaling-instances` 명령을 사용하여 인스턴스가 다시 서비스 상태가 되었는지 확인합니다.

```
aws autoscaling describe-auto-scaling-instances --instance-ids i-05b4f7d5be44822a6
```

다음은 응답의 예입니다. 인스턴스의 상태가 `InService`입니다.

```
{
  "AutoScalingInstances": [
    {
      "ProtectedFromScaleIn": false,
      "AvailabilityZone": "us-west-2a",
      "LaunchTemplate": {
        "LaunchTemplateName": "my-launch-template",
        "Version": "1",
        "LaunchTemplateId": "lt-050555ad16a3f9c7f"
      },
      "InstanceId": "i-05b4f7d5be44822a6",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService"
    },
    ...
  ]
}
```

조정 프로세스의 일시 중지 및 재개

이 주제에서는 Auto Scaling 그룹의 조정 프로세스를 하나 이상 일시 중지했다가 재개하는 방법을 설명합니다. Amazon EC2 Auto Scaling의 일시 중지-재개 기능을 사용하기로 했을 때 고려해야 할 문제도 설명합니다.

Important

인스턴스를 재부팅하거나 문제를 해결해야 하는 경우에는 일시 중지-재개 대신 대기 기능을 사용하십시오. 자세한 내용은 [Auto Scaling 그룹에서 일시적으로 인스턴스 제거 \(p. 138\)](#) 단원을 참조하십시오. 인스턴스 축소 보호 기능을 사용하여 자동 축소 동안 특정 인스턴스가 종료되지 않도록 할 수도 있습니다. 자세한 내용은 [인스턴스 축소 보호 \(p. 128\)](#) 단원을 참조하십시오.

사용자가 시작하는 일시 중지 외에, Amazon EC2 Auto Scaling에서도 인스턴스 시작에 반복적으로 실패하는 Auto Scaling 그룹의 프로세스를 일시 중지할 수 있습니다. 이를 관리적 일시 중지라고도 합니다. 관리적 일시 중지는 24시간 이상 인스턴스를 시작하려고 했지만 시작하지 못한 Auto Scaling 그룹에 가장 흔히 적용됩니다. 사용자는 Amazon EC2 Auto Scaling에서 관리상의 이유로 일시 중지한 프로세스를 재개할 수 있습니다.

목차

- [조정 프로세스 \(p. 143\)](#)
- [일시 중지 선택 \(p. 143\)](#)
- [조정 프로세스 일시 중지 및 재개\(콘솔\) \(p. 145\)](#)
- [조정 프로세스 일시 중지 및 재개\(AWS CLI\) \(p. 146\)](#)

조정 프로세스

Amazon EC2 Auto Scaling에는 크게 Launch 및 Terminate의 두 가지 프로세스 유형이 있습니다. Launch 프로세스는 새 Amazon EC2 인스턴스를 Auto Scaling 그룹에 추가하여 용량을 늘립니다. Terminate 프로세스는 그룹에서 Amazon EC2 인스턴스를 제거하여 용량을 줄입니다.

그 밖의 Amazon EC2 Auto Scaling 프로세스 유형은 특정한 조정 기능과 관계가 있습니다.

- **AddToLoadBalancer**—인스턴스가 시작될 때 연결된 로드 밸런서 또는 대상 그룹에 인스턴스를 추가합니다.
- **AlarmNotification**—이 그룹의 조정 정책과 연결된 CloudWatch 경보의 알림을 수락합니다.
- **AZRebalance**—예를 들어 이전에 사용할 수 없었던 가용 영역이 정상 상태로 돌아오는 경우 등 그룹의 균형이 깨졌을 때, 그룹 내 EC2 인스턴스의 수를 지정된 모든 가용 영역에 고르게 분산시켜 균형을 잡습니다. 자세한 내용은 [재분배 활동 \(p. 6\)](#) 단원을 참조하십시오.
- **HealthCheck**—인스턴스의 상태를 확인하고 Amazon EC2 또는 Elastic Load Balancing가 Amazon EC2 Auto Scaling에 인스턴스가 비정상임을 알리면 해당 인스턴스를 비정상 상태로 표시합니다. 이 프로세스로 사용자가 수동으로 설정한 인스턴스의 상태를 재정의할 수 있습니다. 자세한 내용은 [Auto Scaling 인스턴스 상태 확인 \(p. 148\)](#) 단원을 참조하십시오.
- **ReplaceUnhealthy**—비정상 상태로 표시된 인스턴스를 종료한 다음 새로운 인스턴스를 생성하여 이를 교체합니다.
- **ScheduledActions**—사용자가 생성한 예정된 조정 작업 또는 AWS Auto Scaling의 예측 조정 기능으로 생성된 예정된 조정 작업을 수행합니다.

일시 중지 선택

각각의 프로세스 유형을 개별적으로 일시 중지하고 재개할 수 있습니다. 이 단원에서는 조정 프로세스를 일시 중지하기로 결정하기 전에 취해야 할 행동과 몇 가지 지침을 제시합니다. 개별 프로세스를 일시 중지하면 다른 프로세스와 충돌할 수 있으니 주의하십시오. 프로세스를 일시 중지하는 이유에 따라 여러 프로세스를 함께 일시 중지해야 할 수도 있습니다.

다음은 개별 프로세스 유형을 일시 중지할 때 어떤 일이 발생하는지에 대한 설명입니다.

Warning

Launch 또는 Terminate 프로세스 유형을 일시 중지하면 다른 프로세스 유형이 제대로 작동하지 못하게 될 수 있습니다.

Terminate

- Auto Scaling 그룹은 이 프로세스가 일시 중지된 동안 발생하는 경고 또는 예정된 작업에 따라 축소되지 않습니다. 이와 함께 다음 프로세스가 중단됩니다.
- AZRebalance는 여전히 활성이지만 정상적으로 작동하지 않습니다. 오래된 인스턴스를 종료하지 않고도 새 인스턴스를 시작할 수 있습니다. 이로 인해 Auto Scaling은 최대 크기보다 최대 10% 더 크게 확장될 수 있습니다. 재분배 활동 중에 일시적으로 허용되기 때문입니다. 그리고 Auto Scaling 그룹은 Terminate 프로세스를 재개할 때까지 최대 크기 이상으로 유지될 수 있습니다. Terminate가 재개되고 가용 영역 간에 Auto Scaling 그룹의 균형이 더 이상 유지되지 않거나 다른 가용 영역이 지정된 경우, AZRebalance가 서서히 그룹을 재분배합니다.
- ReplaceUnhealthy는 비활성 상태이지만 HealthCheck는 아닙니다. Terminate가 재개되면 ReplaceUnhealthy 프로세스는 즉시 실행되기 시작합니다. Terminate가 일시 중지된 동안 비정상적으로 표시되었던 인스턴스는 즉시 교체됩니다.

Launch

- Auto Scaling 그룹은 이 프로세스가 일시 중지된 동안 발생한 경로나 예약된 작업에 따라 확장되지 않습니다. AZRebalance는 이 그룹의 재분배를 중지합니다. ReplaceUnhealthy는 비정상 인스턴스를 계속 종료하지만 교체용 인스턴스를 시작하지는 않습니다. 사용자가 Launch를 재개하면 재분배 활동 및 상태 검사에 따른 교체 항목이 다음과 같이 처리됩니다.
- 가용 영역 간에 Auto Scaling 그룹의 균형이 더 이상 유지되지 않거나 다른 가용 영역이 지정된 경우, AZRebalance가 서서히 그룹을 재분배합니다.
- ReplaceUnhealthy는 Launch가 일시 중지되었던 동안 종료된 인스턴스를 즉시 교체합니다.

AddToLoadBalancer

- Amazon EC2 Auto Scaling은 인스턴스를 시작하지만 로드 밸런서 또는 대상 그룹에 추가하지는 않습니다. AddToLoadBalancer 프로세스를 재개하면 이 프로세스는 인스턴스 시작 시 로드 밸런서 또는 대상 그룹에 해당 인스턴스를 다시 추가하기 시작합니다. 하지만 이 프로세스가 일시 중지되면 인스턴스가 시작되어도 인스턴스가 추가되지 않습니다. 따라서 이러한 인스턴스를 수동으로 등록해야 합니다.

AlarmNotification

- CloudWatch 경고 임계값이 위반되면 Amazon EC2 Auto Scaling은 조정 정책을 실행하지 않습니다. AlarmNotification을 일시 중지하면 조정 정책이나 그와 연결된 CloudWatch 경보를 삭제하지 않고도 그룹 조정 정책에 따라 조정 이벤트가 트리거되지 않도록 일시적으로 중지시킬 수 있습니다. AlarmNotification을 재개하면 Amazon EC2 Auto Scaling은 현재 위반 상태인 경고 임계값으로 정책을 평가합니다.

AZRebalance

- Auto Scaling 그룹은 특정 이벤트 후 인스턴스를 재분배하려 하지 않습니다. 그러나 확장 또는 축소 이벤트가 발생하는 경우 조정 프로세스는 계속해서 가용 영역의 균형을 맞추려고 시도합니다. 예를 들어, 확장 시에는 최소한의 인스턴스로 가용 영역의 인스턴스를 시작합니다. AZRebalance가 일시 중지된 동안 그룹의 균형이 깨어졌고 사용자가 이를 재개하는 경우, Amazon EC2 Auto Scaling은 그룹을 재분배하려고 합니다. 이를 위해 먼저 Launch를 호출한 다음 Terminate를 호출합니다.

HealthCheck

- Amazon EC2 Auto Scaling은 EC2 및 Elastic Load Balancing 상태 확인의 결과에 따라 인스턴스를 비정상적으로 표시하던 동작을 중지합니다. 그러나 사용자 지정 상태 검사는 계속 정상적으로 이루어집니다. HealthCheck를 일시 중지한 뒤 필요하면 그룹 내 인스턴스의 상태 검사를 수동으로 설정하고 ReplaceUnhealthy에서 이를 교체하도록 할 수 있습니다.

ReplaceUnhealthy

- Amazon EC2 Auto Scaling은 비정상 상태로 표시된 인스턴스를 교체하던 동작을 중지합니다. EC2 또는 Elastic Load Balancing 상태 확인에 실패한 인스턴스는 여전히 비정상 상태로 표시됩니다. 사용자가 ReplaceUnhealthy 프로세스를 재개하는 즉시 Amazon EC2 Auto Scaling은 이 프로세스가 중지된 동안 비정상 상태로 표시되었던 인스턴스를 교체합니다. ReplaceUnhealthy 프로세스는 먼저 Terminate, 이어서 Launch의 순서로 기본 프로세스 유형을 둘 다 호출합니다.

ScheduledActions

- Amazon EC2 Auto Scaling은 일시 중지 기간 동안 실행되도록 예약된 조정 작업을 실행하지 않습니다. ScheduledActions를 재개하면 Amazon EC2 Auto Scaling은 실행 시간이 아직 경과하지 않은 예약된 작업만 평가합니다.

시작 및 종료 모두 일시 중지

Launch 및 Terminate 프로세스 유형을 모두 일시 중지하면 다음과 같은 상황이 됩니다.

- Auto Scaling 그룹은 조정 활동을 시작하거나 원하는 용량을 유지할 수 없습니다.
- 가용 영역 간에 그룹의 균형이 깨어지더라도, Amazon EC2 Auto Scaling은 Auto Scaling 그룹에 지정된 가용 영역 간에 인스턴스를 고르게 재분배하려고 하지 않습니다.
- Auto Scaling 그룹은 비정상 상태로 표시된 인스턴스를 교체할 수 없습니다.

사용자가 Launch 및 Terminate 프로세스 유형을 재개하면, Amazon EC2 Auto Scaling은 이 프로세스가 중지된 동안 비정상 상태로 표시되었던 인스턴스를 교체하고 그룹을 재분배하려고 시도할 수 있습니다. 조정 활동도 재개됩니다.

추가 고려 사항

Launch 및 Terminate가 일시 중지된 동안 몇 가지 외부 작업도 영향을 받을 수 있습니다.

- 스팟 인스턴스 중단 — Terminate가 일시 중지되었는데 Auto Scaling 그룹에 스팟 인스턴스가 있는 경우, 더 이상 사용 가능한 스팟 용량이 없어도 종료는 가능합니다. Launch가 일시 중지된 경우, Amazon EC2 Auto Scaling은 이 프로세스가 다시 사용 가능하게 되어도 다른 스팟 인스턴스 풀이나 동일한 스팟 인스턴스 풀에서 교체용 인스턴스를 시작할 수 없습니다.
- 인스턴스 연결 및 분리 — Launch 및 Terminate가 일시 중지된 경우, Auto Scaling 그룹에 연결된 인스턴스를 분리할 수 있지만 새 인스턴스를 그룹에 연결할 수는 없습니다. 인스턴스를 연결하려면 먼저 Launch를 재개해야 합니다.

Note

인스턴스를 분리한 직후에 수동으로 종료하려는 경우, 그 대신 [terminate-instance-in-auto-scaling-group](#) CLI 명령을 호출하면 됩니다. 이 명령은 지정된 인스턴스를 종료하고 원하는 그룹 용량을 선택적으로 조정합니다. 또한 Auto Scaling 그룹에 수명 주기 후크를 사용하는 경우에는 인스턴스 종료를 위해 사용자 지정된 작업을 실행한 뒤에 인스턴스가 완전히 종료됩니다.

- 대기 인스턴스 — Launch가 일시 중지된 동안에는 Standby 상태의 인스턴스를 서비스 상태로 되돌릴 수 없습니다. 인스턴스를 서비스 상태로 되돌리려면 먼저 Launch를 재개해야 합니다.

조정 프로세스 일시 중지 및 재개(콘솔)

개별 프로세스 또는 모든 프로세스를 일시 중지 및 재개할 수 있습니다.

프로세스를 일시 중지 및 재개하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. [Details] 탭에서 [Edit]를 선택합니다.
5. Suspended processes(일시 중지된 프로세스)에서 일시 중지할 프로세스를 선택합니다.

일시 중지된 프로세스를 재개하려면 Suspended processes(일시 중지된 프로세스)에서 해당 프로세스를 제거합니다.

6. 저장을 선택합니다.

조정 프로세스 일시 중지 및 재개(AWS CLI)

개별 프로세스 또는 모든 프로세스를 일시 중지 및 재개할 수 있습니다.

프로세스를 일시 중지하려면

다음과 같이 `suspend-processes` 명령을 `--scaling-processes` 옵션과 함께 사용합니다.

```
aws autoscaling suspend-processes --auto-scaling-group-name my-asg --scaling-processes AlarmNotification
```

모든 프로세스를 일시 중지하려면

다음과 같이 `suspend-processes` 명령을 사용합니다(`--scaling-processes` 옵션 생략).

```
aws autoscaling suspend-processes --auto-scaling-group-name my-asg
```

일시 중지된 프로세스를 재개하려면

다음과 같이 `resume-processes` 명령을 실행합니다.

```
aws autoscaling resume-processes --auto-scaling-group-name my-asg --scaling-processes AlarmNotification
```

일시 중지된 모든 프로세스를 재개하려면

다음과 같이 `resume-processes` 명령을 사용합니다(`--scaling-processes` 옵션 생략).

```
aws autoscaling resume-processes --auto-scaling-group-name my-asg
```

Auto Scaling 인스턴스 및 그룹 모니터링

모니터링은 Amazon EC2 Auto Scaling 및 AWS 솔루션의 안정성, 가용성 및 성능을 유지하는 중요한 역할을 합니다. 다중 지점 오류가 발생할 경우 보다 쉽게 디버깅 수 있도록 AWS 솔루션의 모든 부분에서 모니터링 데이터를 수집해야 합니다. AWS는 Amazon EC2 Auto Scaling을 감시하고 문제가 있을 때 보고하며 적절한 경우 자동 작업을 수행할 수 있는 모니터링 도구를 제공합니다.

상태 확인

Amazon EC2 Auto Scaling은 Auto Scaling 그룹의 인스턴스에 대한 주기적인 상태 확인을 수행하여 상태가 비정상인 인스턴스를 식별합니다. Amazon EC2 상태 확인, Elastic Load Balancing 상태 확인 또는 사용자 지정 상태 확인을 사용하여 인스턴스의 상태를 확인하도록 Auto Scaling 그룹을 구성할 수 있습니다. 자세한 내용은 [Auto Scaling 인스턴스 상태 확인 \(p. 148\)](#) 단원을 참조하십시오.

Amazon Simple Notification Service 알림

Amazon EC2 Auto Scaling이 인스턴스를 시작하거나 종료할 때 Amazon SNS 알림을 보내도록 Auto Scaling 그룹을 구성할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹 조정 시 Amazon SNS SNS 알림 받기 \(p. 150\)](#) 단원을 참조하십시오.

CloudWatch 이벤트

Auto Scaling 그룹이 인스턴스를 시작하거나 종료할 때 또는 수명 주기 작업이 발생할 때 Amazon EC2 Auto Scaling에서 Amazon CloudWatch Events에 이벤트를 제출할 수 있습니다. 이렇게 하면 이러한 이벤트가 발생할 때 다른 AWS 서비스에서 자동화된 작업을 트리거할 수도 있습니다. 자세한 내용은 [Auto Scaling 그룹 조정 시 CloudWatch 이벤트 수신 \(p. 155\)](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling에서 수행한 API 호출에 대해 트리거하는 규칙을 작성할 수도 있습니다. 자세한 내용은 Amazon CloudWatch Events 사용 설명서의 [AWS CloudTrail을 사용하여 AWS API 호출을 트리거하는 CloudWatch 이벤트 규칙 생성](#)을 참조하십시오.

CloudWatch 경보

CloudWatch는 AWS 리소스에 대한 특정 지표를 자동으로 모니터링하여 비정상 애플리케이션 동작을 탐지할 수 있도록 합니다. CloudWatch 경보를 구성하고 지표의 값이 예상과 다르거나 특정 이상이 탐지될 때 이메일을 전송하는 Amazon SNS 알림을 설정할 수 있습니다. 예를 들어 네트워크 활동이 지표의 예상 값보다 갑자기 높아지거나 낮아지면 알림을 받을 수 있습니다. 자세한 내용은 [Amazon CloudWatch 사용 설명서](#) 단원을 참조하십시오.

CloudWatch 대시보드

CloudWatch 대시보드는 CloudWatch 콘솔의 사용자 정의 가능한 홈 페이지입니다. 이 페이지를 사용하면 여러 리전에 분산된 리소스까지 포함한 리소스를 단일 보기에서 모니터링할 수 있습니다. CloudWatch 대시보드를 사용해 AWS 리소스에 대한 지표 및 경보를 보여주는 사용자 지정 뷰를 생성할 수 있습니다. 자세한 내용은 [Amazon CloudWatch 사용 설명서](#) 단원을 참조하십시오.

CloudTrail 로그

AWS CloudTrail을 사용하면 해당 AWS 계정을 대신하여 Amazon EC2 Auto Scaling API에 대해 이루어진 호출을 추적할 수 있습니다. CloudTrail은 이 정보를 지정된 Amazon S3 버킷의 로그 파일에 정보를 저장합니다. 이러한 로그 파일을 사용하여 Auto Scaling 그룹의 활동을 모니터링할 수 있습니다. 로그에는 수행된 요청, 요청의 발신 IP 주소, 요청한 사람, 요청한 시간 등이 기록됩니다. 자세한 내용은 [AWS CloudTrail을 사용하여 Amazon EC2 Auto Scaling API 호출 로깅 \(p. 166\)](#) 단원을 참조하십시오.

CloudWatch Logs

CloudWatch Logs를 사용하면 Amazon EC2 인스턴스, CloudTrail 및 다른 소스에서 로그 파일을 모니터링, 저장, 액세스할 수 있습니다. CloudWatch Logs는 로그 파일의 정보를 모니터링하고 특정 임계값이 충족될 때 이를 알릴 수 있습니다. 또한 매우 내구력 있는 스토리지에 로그 데이터를 저장할 수 있습니다. 자세한 정보는 [Amazon CloudWatch Logs User Guide](#)를 참조하십시오.

AWS Personal Health Dashboard

Personal Health Dashboard(PHD)는 정보를 표시하고 AWS 리소스의 상태 변경에 따라 트리거되는 알림도 제공합니다. 이 정보는 최근 이벤트와 예정된 이벤트를 범주별로 보여주는 대시보드와 지난 90일 간의 모든 이벤트를 보여주는 전체 이벤트 로그의 두 가지 방법으로 표시됩니다. 자세한 내용은 [Amazon EC2 Auto Scaling에 대한 Personal Health Dashboard 알림 \(p. 168\)](#) 단원을 참조하십시오.

Auto Scaling 인스턴스 상태 확인

Auto Scaling 인스턴스의 상태는 정상 또는 비정상입니다. Auto Scaling 그룹의 모든 인스턴스는 정상 상태에서 시작됩니다. Amazon EC2 Auto Scaling에서 해당 인스턴스가 비정상 상태라는 알림을 수신하지 않는 한 인스턴스는 정상 상태로 간주됩니다. 이 알림은 Amazon EC2, Elastic Load Balancing 또는 사용자 지정 상태 확인 중 하나 이상에서 전송될 수 있습니다.

Amazon EC2 Auto Scaling에서 상태가 비정상인 것으로 표시한 인스턴스는 교체가 예정입니다. 인스턴스를 교체하지 않으려면 개별 Auto Scaling 그룹의 상태 확인 프로세스를 일시 중지하면 됩니다.

인스턴스 상태

Amazon EC2 Auto Scaling은 다음 중 하나 이상을 사용하여 인스턴스의 상태를 확인할 수 있습니다.

- 인스턴스를 손상할 수 있는 하드웨어 및 소프트웨어 문제를 확인하기 위해 Amazon EC2가 제공하는 상태 검사입니다. 인스턴스 상태 검사와 시스템 상태 검사가 모두 포함됩니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [상태 검사 유형](#)을 참조하십시오.
- Elastic Load Balancing에서 제공한 상태 확인.
- 사용자 지정 상태 확인.

EC2 상태 검사는 Amazon EC2 Auto Scaling의 기본 상태 확인이며, 특별한 구성이 필요 없습니다. 그러나 추가 검사를 지정하여 Auto Scaling 그룹에서 실시하는 기본 상태 확인을 사용자 지정할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹에 Elastic Load Balancing 상태 확인 추가 \(p. 70\)](#) 및 [사용자 지정 상태 확인 \(p. 149\)](#) 단원을 참조하십시오.

인스턴스 상태 확인

인스턴스가 완전히 구성되고 초기 상태 확인을 통과하면 해당 인스턴스는 Amazon EC2 Auto Scaling에 의해 정상으로 판정되고 InService 상태에 들어갑니다. Amazon EC2 Auto Scaling은 Auto Scaling 그룹의 인스턴스에 대한 주기적인 상태 확인을 수행하여 상태가 비정상인 인스턴스를 식별합니다.

Amazon EC2 Auto Scaling 상태 확인은 Amazon EC2 상태 검사 결과를 사용하여 인스턴스의 상태를 확인합니다. 인스턴스가 running 이외의 상태이거나 시스템 상태가 `impaired`이면 Amazon EC2 Auto Scaling은 해당 인스턴스를 비정상 상태로 간주하여 교체 인스턴스를 시작합니다. 인스턴스가 다음 상태일 때도 위와 같이 실행됩니다.

- `stopping`
- `stopped`
- `terminating`
- `terminated`

Auto Scaling 그룹에 로드 밸런서 또는 대상 그룹을 연결한 경우에는 Elastic Load Balancing가 OutOfService로 보고할 때 인스턴스를 비정상 상태로 표시하도록 그룹을 구성할 수 있습니다. 로드 밸런서에 연결 드레이닝이 활성화된 경우 Amazon EC2 Auto Scaling이 대기하는 시간은 조정 이벤트나 상태 확인 대체로 인해 인스턴스를 종료하기 전에 내부 요청이 완료되거나 최대 제한 시간이 만료되는 경우 중 먼저 일어나는 쪽에 해당합니다. Elastic Load Balancing 상태 확인을 사용하도록 Auto Scaling 그룹을 구성하면 Amazon EC2 Auto Scaling이 EC2 상태 검사와 Elastic Load Balancing 상태 확인을 모두 실시하여 인스턴스의 상태를 확인합니다. 자세한 내용은 [Auto Scaling 그룹에 Elastic Load Balancing 상태 확인 추가 \(p. 70\)](#) 단원을 참조하십시오.

사용자 지정 상태 확인을 사용할 경우 이 상태 확인 정보를 Amazon EC2 Auto Scaling으로 보내서 Amazon EC2 Auto Scaling이 이 정보를 사용하게 할 수 있습니다. 예를 들어, 인스턴스가 예상대로 작동하지 않는다고 판단되는 경우 인스턴스의 상태를 Unhealthy로 설정할 수 있습니다. Amazon EC2 Auto Scaling은 다음에 인스턴스에 대해 상태 확인을 수행할 때, 인스턴스가 비정상이라고 판단하면 교체 인스턴스를 시작합니다.

상태 확인 유예 기간

대부분의 경우, 서비스 상태가 된 직후의 Auto Scaling 인스턴스는 워 업을 거쳐야 상태 확인을 통과할 수 있습니다. Amazon EC2 Auto Scaling은 상태 확인 유예 기간이 끝날 때까지 기다린 후 인스턴스의 상태를 확인합니다. Amazon EC2 상태 확인과 Elastic Load Balancing 상태 확인은 상태 확인 유예 기간이 끝나기 전에 완료될 수 있습니다. 하지만 Amazon EC2 Auto Scaling은 상태 확인 유예 기간이 종료되기 전에는 그러한 상태를 반영하지 않습니다. 인스턴스에 충분한 워 업 시간을 제공하려면 상태 확인 유예 기간이 애플리케이션의 예상 시작 시간을 포함하도록 해야 합니다. 수명 주기 후크를 추가할 경우 유예 기간은 수명 주기 후크 작업이 완료되고 인스턴스가 InService 상태로 전환되기까지 시작되지 않습니다.

상태 확인 유예 기간은 초 단위입니다. 따라서 예를 들어 300초를 지정하면 5분 간격이 생깁니다.

비정상 인스턴스 교체

Amazon EC2 또는 Elastic Load Balancing 상태 확인 결과 인스턴스가 비정상 상태로 표시된 후에는 거의 즉시 교체가 예약됩니다. 인스턴스 상태가 자동으로 복구되는 일은 없습니다. `SetInstanceHealth` 작업(또는 `set-instance-health` 명령)을 호출하여 수동으로 끼어들어서 인스턴스의 상태를 다시 정상으로 설정할 수는 있습니다. 이미 인스턴스가 종료된 경우에는 오류 메시지가 발생합니다. 인스턴스를 비정상 상태로 표시하는 것과 실제 인스턴스 종료 간의 시간 간격이 매우 짧으므로, `SetInstanceHealth` 작업(또는 `set-instance-health` 명령)을 통해 인스턴스 상태를 다시 정상으로 설정하려는 시도는 일시 중지된 그룹에만 유용할 수 있습니다. 자세한 내용은 [조정 프로세스의 일시 중지 및 재개 \(p. 142\)](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling에서는 비정상 인스턴스의 종료를 위한 새 조정 활동을 생성한 다음 이를 종료합니다. 나중에 다른 조정 활동에 의해 새 인스턴스가 시작되어 종료된 인스턴스를 대체합니다.

인스턴스가 종료되면 연결된 모든 Elastic IP 주소와의 연결이 해제되고 새 인스턴스와 자동으로 연결되지 않습니다. 이러한 Elastic IP 주소를 새 인스턴스에 수동으로 연결해야 합니다. 이와 유사하게, 인스턴스가 종료되면 여기 연결된 EBS 볼륨이 분리됩니다. 이러한 EBS 볼륨을 새 인스턴스에 수동으로 연결해야 합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [엘라스틱 IP 주소 연결 해제 및 다른 인스턴스와 다시 연결](#) 및 [Amazon EBS 볼륨을 인스턴스에 연결](#)을 참조하십시오.

사용자 지정 상태 확인

자체 상태 확인 시스템이 있는 경우에는 시스템에서 Amazon EC2 Auto Scaling로 직접 인스턴스의 상태 정보를 전송할 수 있습니다.

다음 `set-instance-health` 명령을 사용하여 지정된 인스턴스의 상태를 Unhealthy로 설정합니다.

```
aws autoscaling set-instance-health --instance-id i-123abc45d --health-status Unhealthy
```

다음 `describe-auto-scaling-groups` 명령을 사용하여 인스턴스 상태가 Unhealthy인지 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-names my-asg
```

다음은 인스턴스의 상태가 Unhealthy이고 종료될 것임을 보여 주는 응답의 예입니다.

```
{
  "AutoScalingGroups": [
    {
      ....
      "Instances": [
        {
          "ProtectedFromScaleIn": false,
          "AvailabilityZone": "us-west-2a",
          "LaunchTemplate": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1",
            "LaunchTemplateId": "lt-050555ad16a3f9c7f"
          },
          "InstanceId": "i-123abc45d",
          "HealthStatus": "Unhealthy",
          "LifecycleState": "Terminating"
        },
        ...
      ]
    }
  ]
}
```

Auto Scaling 그룹 조정 시 Amazon SNS SNS 알림 받기

Amazon EC2 Auto Scaling이 언제 Auto Scaling 그룹에서 EC2 인스턴스를 시작하거나 종료하는지를 알면 유용합니다. Amazon SNS는 구독 클라이언트 또는 엔드포인트로 알림을 배달하거나 전송하는 것을 조정하고 관리합니다. Auto Scaling 그룹이 조정될 때마다 SNS 알림을 보내도록 Amazon EC2 Auto Scaling을 구성할 수 있습니다.

Amazon SNS는 HTTP 또는 HTTPS POST, 이메일(SMTP, 일반 텍스트 또는 JSON 형식) 또는 Amazon SQS 대기열에 게시된 메시지로 알림을 배달할 수 있습니다. 자세한 내용은 Amazon Simple Notification Service 개발자 안내서의 [Amazon SNS란 무엇입니까?](#) 단원을 참조하십시오.

예를 들어, Auto Scaling 그룹이 `autoscaling: EC2_INSTANCE_TERMINATE` 알림 유형을 사용하도록 구성하면 Auto Scaling 그룹에서 인스턴스 종료 시 이메일 알림이 전송됩니다. 이 이메일에는 종료된 인스턴스의 세부 정보(예: 인스턴스 ID 및 인스턴스 종료 사유)가 포함됩니다.

Tip

원하는 경우 Auto Scaling 그룹이 조정되거나 수명 주기 작업이 발생할 때 Amazon CloudWatch Events를 사용하여 Lambda 함수를 호출하도록 대상을 구성할 수 있습니다. 자세한 내용은 [Auto Scaling 그룹 조정 시 CloudWatch 이벤트 수신 \(p. 155\)](#) 단원을 참조하십시오.

목차

- [SNS 알림 \(p. 151\)](#)
- [Amazon SNS 구성 \(p. 151\)](#)
- [알림을 보내도록 Auto Scaling 그룹 구성 \(p. 152\)](#)
- [알림 구성 테스트 \(p. 153\)](#)
- [조정 이벤트 알림 수신 확인 \(p. 153\)](#)

- 알림 구성 삭제 (p. 154)

SNS 알림

Amazon EC2 Auto Scaling은 다음과 같은 이벤트가 발생하는 경우 Amazon SNS 알림을 보낼 수 있도록 지원합니다.

이벤트	설명
autoscaling:EC2_INSTANCE_LAUNCH	인스턴스 시작 성공
autoscaling:EC2_INSTANCE_LAUNCH_ERROR	인스턴스 시작 실패
autoscaling:EC2_INSTANCE_TERMINATE	인스턴스 종료 성공
autoscaling:EC2_INSTANCE_TERMINATE_ERROR	인스턴스 종료 실패

메시지에 포함되는 정보는 다음과 같습니다.

- Event - 이벤트
- AccountId - AWS 계정 ID.
- AutoScalingGroupName - Auto Scaling 그룹 이름.
- AutoScalingGroupARN - Auto Scaling 그룹의 ARN
- EC2InstanceId - EC2 인스턴스의 ID.

다음 예를 참조하십시오.

```
Service: AWS Auto Scaling
Time: 2016-09-30T19:00:36.414Z
RequestId: 4e6156f4-a9e2-4bda-a7fd-33f2ae528958
Event: autoscaling:EC2_INSTANCE_LAUNCH
AccountId: 123456789012
AutoScalingGroupName: my-asg
AutoScalingGroupARN: arn:aws:autoscaling:region:123456789012:autoScalingGroup...
ActivityId: 4e6156f4-a9e2-4bda-a7fd-33f2ae528958
Description: Launching a new EC2 instance: i-0598c7d356eba48d7
Cause: At 2016-09-30T18:59:38Z a user request update of AutoScalingGroup constraints to ...
StartTime: 2016-09-30T19:00:04.445Z
EndTime: 2016-09-30T19:00:36.414Z
StatusCode: InProgress
StatusMessage:
Progress: 50
EC2InstanceId: i-0598c7d356eba48d7
Details: {"Subnet ID":"subnet-id","Availability Zone":"zone"}
```

Amazon SNS 구성

Amazon SNS를 사용하여 이메일 알림을 전송하려면 먼저 주제를 생성한 다음 해당 주제에 이메일 주소를 구독해야 합니다.

Amazon SNS 주제 생성

SNS 주제는 논리적 액세스 지점으로 Auto Scaling 그룹에서 알림을 전송하는 데 사용하는 통신 채널입니다. 주제의 이름을 지정하여 주제를 생성합니다.

자세한 내용은 Amazon Simple Notification Service 개발자 안내서의 [주제 생성](#)을 참조하십시오.

Amazon SNS 주제를 구독합니다.

Auto Scaling 그룹에서 주제로 전송하는 알림을 받으려면 엔드포인트가 해당 주제를 구독해야 합니다. 이 절차에서 엔드포인트에 Amazon EC2 Auto Scaling의 알림을 받을 이메일 주소를 지정합니다.

자세한 내용은 의 주제 구독을 참조하십시오.

Amazon SNS 구독 확인

Amazon SNS는 이전 단계에서 지정한 이메일 주소로 확인 이메일을 보냅니다.

다음 단계로 넘어가기 전에 AWS 알림의 이메일을 열어 구독 확인 링크를 선택하십시오.

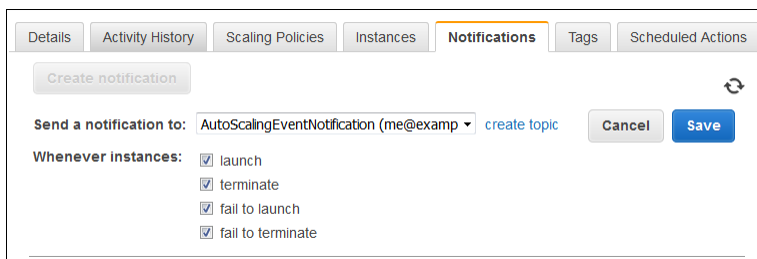
AWS에서 확인 메시지를 받게 됩니다. 이제 Amazon SNS가 지정된 이메일 주소로 이메일 형식의 알림을 주고받을 수 있도록 구성됩니다.

알림을 보내도록 Auto Scaling 그룹 구성

Auto Scaling 그룹에서 인스턴스 시작 또는 종료 등의 조정 이벤트가 발생하는 경우 Amazon SNS에 알림을 보내도록 구성할 수 있습니다. Amazon SNS는 지정한 이메일 주소로 인스턴스에 대한 정보가 들어 있는 알림을 보냅니다.

Amazon SNS 그룹에서 Auto Scaling 알림을 보내도록 구성하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. [Notifications] 탭에서 [Create notification]을 선택합니다.
5. [Create notifications] 창에서 다음을 수행합니다.
 - a. [Send a notification to:]에서 SNS 주제를 선택합니다.
 - b. [Whenever instances]에서 알림을 보낼 이벤트를 선택합니다.
 - c. Save를 선택합니다.



Amazon SNS 그룹에서 Auto Scaling 알림을 보내도록 구성하려면(AWS CLI)

다음 `put-notification-configuration` 명령을 사용합니다.

```
aws autoscaling put-notification-configuration --auto-scaling-group-name my-  
asg --topic-arn arn --notification-types "autoscaling:EC2_INSTANCE_LAUNCH"  
"autoscaling:EC2_INSTANCE_TERMINATE"
```

알림 구성 테스트

시작 이벤트에 대해 알림이 발생하도록 하려면 Auto Scaling 그룹의 용량을 1만큼 늘려 Auto Scaling 그룹을 업데이트합니다. Amazon EC2 Auto Scaling에서 EC2 인스턴스를 시작하면 몇 분 안에 이메일 알림이 전송됩니다.

원하는 용량을 변경하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling 아래에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. [Details] 탭에서 [Edit]를 선택합니다.
5. [Desired]에서 현재 값을 1만큼 늘립니다. 이 값이 [Max]를 넘으면 [Max] 값도 1만큼 늘려야 합니다.
6. Save를 선택합니다.
7. 몇 분 후 시작 이벤트에 대한 알림 이메일이 전송됩니다. 이 테스트에서 시작한 인스턴스가 추가로 필요하지 않으면 [Desired]를 1만큼 줄일 수 있습니다. 몇 분 후 종료 이벤트에 대한 알림 이메일이 전송됩니다.

원하는 용량을 변경하려면(AWS CLI)

다음 `set-desired-capacity` 명령을 사용합니다.

```
aws autoscaling set-desired-capacity --auto-scaling-group-name my-asg --desired-capacity 2
```

조정 이벤트 알림 수신 확인

Amazon SNS의 메시지 이메일을 확인하고 이메일을 엽니다. Auto Scaling 그룹의 조정 이벤트에 대한 알림을 받은 후 Auto Scaling 그룹의 설명을 보고 조정 이벤트를 확인할 수 있습니다. 이때 시작 또는 종료된 인스턴스 ID 등의 알림 이메일 관련 정보가 필요합니다.

Auto Scaling 그룹에서 새 인스턴스를 시작했는지 확인하려면(콘솔)

1. Auto Scaling 그룹을 선택합니다.
2. [Activity History] 탭의 [Status] 옆에 인스턴스의 현재 상태가 표시됩니다. 예를 들어, 알림에서 인스턴스가 시작되었음을 나타내는 경우 새로 고침 버튼을 사용하여 시작 활동이 [Successful] 상태인지 확인합니다.
3. [Instances] 탭에서 알림 이메일에서 받은 ID를 가진 인스턴스의 현재 [Lifecycle] 상태를 확인할 수 있습니다. 새 인스턴스가 시작되면 해당 수명 주기 상태가 InService로 변경됩니다.

Auto Scaling 그룹에서 새 인스턴스를 시작했는지 확인하려면(AWS CLI)

다음 `describe-auto-scaling-groups` 명령을 사용하여 변경된 Auto Scaling 그룹의 크기가 변경되었음을 확인합니다.

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

다음 예의 출력은 그룹에 인스턴스 2개가 실행 중임을 보여 줍니다. 이메일 알림에서 받은 ID를 가진 인스턴스를 확인합니다.

```
{
  "AutoScalingGroups": [
```

```
{
  "AutoScalingGroupARN": "arn",
  "HealthCheckGracePeriod": 0,
  "SuspendedProcesses": [],
  "DesiredCapacity": 2,
  "Tags": [],
  "EnabledMetrics": [],
  "LoadBalancerNames": [],
  "AutoScalingGroupName": "my-asg",
  "DefaultCooldown": 300,
  "MinSize": 1,
  "Instances": [
    {
      "InstanceId": "i-d95eb0d4",
      "AvailabilityZone": "us-west-2b",
      "HealthStatus": "Healthy",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    },
    {
      "InstanceId": "i-13d7dc1f",
      "AvailabilityZone": "us-west-2a",
      "HealthStatus": "Healthy",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    }
  ],
  "MaxSize": 5,
  "VPCZoneIdentifier": null,
  "TerminationPolicies": [
    "Default"
  ],
  "LaunchConfigurationName": "my-lc",
  "CreatedTime": "2015-03-01T16:12:35.608Z",
  "AvailabilityZones": [
    "us-west-2b",
    "us-west-2a"
  ],
  "HealthCheckType": "EC2"
}
```

알림 구성 삭제

언제든지 Amazon EC2 Auto Scaling 알림 구성을 삭제할 수 있습니다.

Amazon EC2 Auto Scaling 알림 구성을 삭제하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창의 Auto Scaling에서 Auto Scaling 그룹을 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. [Notifications] 탭에서 알림 옆의 [Delete]를 선택합니다.

Amazon EC2 Auto Scaling 알림 구성 삭제(AWS CLI)

다음 delete-notification-configuration 명령을 사용합니다.

```
aws autoscaling delete-notification-configuration --auto-scaling-group-name my-asg --topic-arn arn
```

Auto Scaling 그룹과 연결된 모든 설명 및 Amazon SNS 주제를 삭제하는 방법은 Amazon Simple Notification Service 개발자 안내서의 [정리](#)를 참조하십시오.

Auto Scaling 그룹 조정 시 CloudWatch 이벤트 수신

Amazon CloudWatch Events로 AWS 서비스를 자동화하여 애플리케이션 가용성 문제나 리소스 변경 같은 시스템 이벤트에 대응할 수 있습니다. AWS 서비스 이벤트는 거의 실시간으로 CloudWatch 이벤트로 전송됩니다. 원하는 이벤트만 표시되도록 간단한 규칙을 작성한 후 규칙과 일치하는 이벤트 발생 시 실행할 자동화 작업을 지정할 수 있습니다.

CloudWatch 이벤트를 사용하면 Lambda 함수나 Amazon SNS 주제 등 다양한 대상을 설정하고 JSON 형식으로 된 이벤트를 받을 수 있습니다. 자세한 내용은 [Amazon CloudWatch Events 사용 설명서](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling이 언제 Auto Scaling 그룹의 EC2 인스턴스를 시작하거나 종료하는지 알면 도움이 됩니다. Auto Scaling 그룹이 규모 조정될 때마다 CloudWatch 이벤트에 이벤트를 보내도록 Amazon EC2 Auto Scaling을 구성할 수 있습니다.

Note

Amazon EC2가 스팟 인스턴스를 회수하려고 할 때도 2분 짜리 경고를 받을 수 있습니다. 스팟 인스턴스 중단 이벤트의 예제는 Amazon EC2 User Guide for Linux Instances의 [Spot Instance Interruption Notices](#)를 참조하십시오.

목차

- [Auto Scaling 이벤트](#) (p. 155)
- [Lambda 함수 만들기](#) (p. 159)
- [Lambda 함수로 이벤트 라우팅](#) (p. 159)

Auto Scaling 이벤트

Amazon EC2 Auto Scaling은 다음과 같은 이벤트가 발생하는 경우 CloudWatch 이벤트에 이벤트를 보낼 수 있도록 지원합니다.

- [EC2 인스턴스-시작 수명 주기 작업](#) (p. 155)
- [EC2 인스턴스 시작 성공](#) (p. 156)
- [EC2 인스턴스 시작 실패](#) (p. 156)
- [EC2 인스턴스-종료 수명 주기 작업](#) (p. 157)
- [EC2 인스턴스 종료 성공](#) (p. 158)
- [EC2 인스턴스 종료 실패](#) (p. 158)

EC2 인스턴스-시작 수명 주기 작업

Amazon EC2 Auto Scaling이 수명 주기 후크로 인해 인스턴스를 Pending:Wait 상태로 이동시켰습니다.

이벤트 데이터

다음은 이 이벤트의 예제 데이터입니다.

```
{
```

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance-launch Lifecycle Action",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn"
  ],
  "detail": {
    "LifecycleActionToken": "87654321-4321-4321-4321-210987654321",
    "AutoScalingGroupName": "my-asg",
    "LifecycleHookName": "my-lifecycle-hook",
    "EC2InstanceId": "i-1234567890abcdef0",
    "LifecycleTransition": "autoscaling:EC2_INSTANCE_LAUNCHING",
    "NotificationMetadata": "additional-info"
  }
}
```

EC2 인스턴스 시작 성공

Amazon EC2 Auto Scaling이 인스턴스를 시작했습니다.

이벤트 데이터

다음은 이 이벤트의 예제 데이터입니다.

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance Launch Successful",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn",
    "instance-arn"
  ],
  "detail": {
    "StatusCode": "InProgress",
    "Description": "Launching a new EC2 instance: i-12345678",
    "AutoScalingGroupName": "my-auto-scaling-group",
    "ActivityId": "87654321-4321-4321-4321-210987654321",
    "Details": {
      "Availability Zone": "us-west-2b",
      "Subnet ID": "subnet-12345678"
    },
    "RequestId": "12345678-1234-1234-1234-123456789012",
    "StatusMessage": "",
    "EndTime": "yyyy-mm-ddThh:mm:ssZ",
    "EC2InstanceId": "i-1234567890abcdef0",
    "StartTime": "yyyy-mm-ddThh:mm:ssZ",
    "Cause": "description-text"
  }
}
```

EC2 인스턴스 시작 실패

Amazon EC2 Auto Scaling이 인스턴스를 시작하지 못했습니다.

이벤트 데이터

다음은 이 이벤트의 예제 데이터입니다.

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance Launch Unsuccessful",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn",
    "instance-arn"
  ],
  "detail": {
    "StatusCode": "Failed",
    "AutoScalingGroupName": "my-auto-scaling-group",
    "ActivityId": "87654321-4321-4321-4321-210987654321",
    "Details": {
      "Availability Zone": "us-west-2b",
      "Subnet ID": "subnet-12345678"
    },
    "RequestId": "12345678-1234-1234-1234-123456789012",
    "StatusMessage": "message-text",
    "EndTime": "yyyy-mm-ddThh:mm:ssZ",
    "EC2InstanceId": "i-1234567890abcdef0",
    "StartTime": "yyyy-mm-ddThh:mm:ssZ",
    "Cause": "description-text"
  }
}
```

EC2 인스턴스-종료 수명 주기 작업

Amazon EC2 Auto Scaling이 수명 주기 후크로 인해 인스턴스를 Terminating:Wait 상태로 이동시켰습니다.

이벤트 데이터

다음은 이 이벤트의 예제 데이터입니다.

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance-terminate Lifecycle Action",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn"
  ],
  "detail": {
    "LifecycleActionToken": "87654321-4321-4321-4321-210987654321",
    "AutoScalingGroupName": "my-asg",
    "LifecycleHookName": "my-lifecycle-hook",
    "EC2InstanceId": "i-1234567890abcdef0",
    "LifecycleTransition": "autoscaling:EC2_INSTANCE_TERMINATING",
    "NotificationMetadata": "additional-info"
  }
}
```


EC2 인스턴스 종료 성공

Amazon EC2 Auto Scaling이 인스턴스를 종료했습니다.

이벤트 데이터

다음은 이 이벤트의 예제 데이터입니다.

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance Terminate Successful",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn",
    "instance-arn"
  ],
  "detail": {
    "StatusCode": "InProgress",
    "Description": "Terminating EC2 instance: i-12345678",
    "AutoScalingGroupName": "my-auto-scaling-group",
    "ActivityId": "87654321-4321-4321-4321-210987654321",
    "Details": {
      "Availability Zone": "us-west-2b",
      "Subnet ID": "subnet-12345678"
    },
    "RequestId": "12345678-1234-1234-1234-123456789012",
    "StatusMessage": "",
    "EndTime": "yyyy-mm-ddThh:mm:ssZ",
    "EC2InstanceId": "i-1234567890abcdef0",
    "StartTime": "yyyy-mm-ddThh:mm:ssZ",
    "Cause": "description-text"
  }
}
```

EC2 인스턴스 종료 실패

Amazon EC2 Auto Scaling이 인스턴스를 종료하지 못했습니다.

이벤트 데이터

다음은 이 이벤트의 예제 데이터입니다.

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance Terminate Unsuccessful",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn",
    "instance-arn"
  ],
  "detail": {
    "StatusCode": "Failed",
    "AutoScalingGroupName": "my-auto-scaling-group",
  }
```

```
{
  "ActivityId": "87654321-4321-4321-4321-210987654321",
  "Details": {
    "Availability Zone": "us-west-2b",
    "Subnet ID": "subnet-12345678"
  },
  "RequestId": "12345678-1234-1234-1234-123456789012",
  "StatusMessage": "message-text",
  "EndTime": "yyyy-mm-ddThh:mm:ssZ",
  "EC2InstanceId": "i-1234567890abcdef0",
  "StartTime": "yyyy-mm-ddThh:mm:ssZ",
  "Cause": "description-text"
}
```

Lambda 함수 만들기

다음 절차를 사용하여 Auto Scaling 이벤트를 처리하는 Lambda 함수를 만듭니다.

Lambda 함수를 만들려면

1. <https://console.aws.amazon.com/lambda/>에서 AWS Lambda 콘솔을 엽니다.
2. Lambda를 처음 사용하는 경우 시작 페이지가 표시되면 지금 시작하기를 선택합니다. 그렇지 않은 경우 Lambda 함수 생성을 선택합니다.
3. 블루프린트 선택 페이지에서 필터에 hello-world를 입력하고 hello-world 블루프린트를 선택합니다.
4. 트리거 구성 페이지에서 다음을 선택합니다.
5. 함수 구성 페이지에서 다음을 수행합니다.
 - a. Lambda 함수의 이름과 설명을 입력합니다.
 - b. Lambda 함수의 코드를 편집합니다. 예를 들어, 다음 코드는 단순히 이벤트를 기록합니다.

```
console.log('Loading function');

exports.handler = function(event, context) {
  console.log("AutoScalingEvent()");
  console.log("Event data:\n" + JSON.stringify(event, null, 4));
  context.succeed("...");
};
```

- c. 사용할 기존 역할이 있는 경우 [Role]에서 [Choose an existing role]을 선택한 다음 [Existing role]에서 역할을 선택합니다. 또는 새 역할을 만들려면 [Role]에서 여러 옵션 중 하나를 선택한 다음 지시를 따릅니다.
 - d. (선택 사항) [Advanced settings]에서 필요한 사항을 수정합니다.
 - e. [Next]를 선택합니다.
6. [Review] 페이지에서 [Create function]을 선택합니다.

Lambda 함수로 이벤트 라우팅

다음 절차를 사용하여 Lambda 함수로 Auto Scaling 이벤트를 라우팅합니다.

Lambda 함수로 이벤트를 라우팅하려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 [Events]를 선택합니다.
3. [Create rule]을 선택합니다.

4. [Event selector]에서 [Auto Scaling]을 이벤트 소스로 선택합니다. 기본적으로 규칙이 모든 Auto Scaling 그룹의 모든 Auto Scaling 이벤트에 적용됩니다. 또는 특정 이벤트 또는 특정 Auto Scaling 그룹을 선택할 수 있습니다.
5. 대상에서 대상 추가를 선택합니다. Lambda 함수를 대상 유형으로 선택한 다음 Lambda 함수를 선택합니다.
6. [Configure details]를 선택합니다.
7. 규칙 정의에 규칙의 이름과 설명을 입력합니다.
8. [Create rule]을 선택합니다.

규칙을 테스트하려면 Auto Scaling 그룹의 크기를 변경합니다. Lambda 함수의 예제 코드를 사용한 경우, CloudWatch Logs에 이벤트가 기록됩니다.

규칙을 테스트하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Auto Scaling 그룹을 선택한 다음 Auto Scaling 그룹을 선택합니다.
3. [Details] 탭에서 [Edit]를 선택합니다.
4. [Desired] 값을 변경한 다음 [Save]를 선택합니다.
5. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
6. 탐색 창에서 [Logs]를 선택합니다.
7. Lambda 함수의 로그 그룹을 선택합니다(예: /aws/lambda/*my-function*).
8. 이벤트 데이터를 보려면 로그 스트림을 선택합니다. 다음과 유사한 데이터가 표시됩니다.

```
Event Data
▼ 2016-02-22T17:48:20.778Z ealfjqinxq6pwo9d Loading function
▼ START RequestId: 7560439b-d98c-11e5-932d-f52757e7aee0 Version: $LATEST
▼ 2016-02-22T17:48:20.813Z 7560439b-d98c-11e5-932d-f52757e7aee0 AutoScalingEvent()
▼ 2016-02-22T17:48:20.814Z 7560439b-d98c-11e5-932d-f52757e7aee0 Event data:
{
  "version": "0",
  "id": "df9b0c8c-89c8-4748-92cb-ac68a9029ada",
  "detail-type": "EC2 Instance Launch Successful",
  "source": "aws.autoscaling",
```

Amazon CloudWatch를 사용하여 Auto Scaling 그룹 및 인스턴스 모니터링

지표는 CloudWatch의 기본 개념입니다. 지표는 CloudWatch에 게시된 시간 순서별 데이터 요소 세트를 나타냅니다. 지표는 모니터링할 변수로, 데이터 요소는 시간에 따른 변수의 값을 나타내는 것으로 간주합니다. 이러한 지표를 사용하여 시스템이 예상대로 수행되고 있는지 확인할 수 있습니다.

Amazon EC2 Auto Scaling은 CloudWatch에 Auto Scaling 그룹에 대한 데이터 포인트를 게시합니다. 이러한 지표는 1분 단위로 사용할 수 있습니다. 이 간격은 구성할 수 없습니다. 이러한 지표를 활성화해야 합니다. 다음 섹션에서는 이를 활성화하는 방법을 안내합니다.

Amazon EC2는 CloudWatch에 Auto Scaling 인스턴스를 설명하는 데이터 요소를 게시합니다. Amazon EC2 인스턴스 모니터링의 간격은 구성 가능합니다. 다음 섹션에서는 Auto Scaling 인스턴스에 대한 인스턴스 모니터링을 구성하는 지침을 제공합니다.

Note

Amazon EC2 모니터링은 Auto Scaling 그룹의 EC2 인스턴스뿐 아니라 모든 EC2 인스턴스에 사용할 수 있습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [CloudWatch를 사용하여 인스턴스 모니터링](#)을 참조하십시오.

목차

- [Auto Scaling 그룹 지표 \(p. 161\)](#)
- [Auto Scaling 그룹 지표의 차원 \(p. 162\)](#)
- [Auto Scaling 그룹 지표 활성화 \(p. 162\)](#)
- [Auto Scaling 인스턴스에 대한 모니터링 구성 \(p. 163\)](#)
- [CloudWatch 지표 보기 \(p. 164\)](#)
- [Amazon CloudWatch 경보 만들기 \(p. 165\)](#)

Auto Scaling 그룹 지표

AWS/AutoScaling 네임스페이스에는 다음 지표가 포함되어 있습니다.

지표	설명
GroupMinSize	Auto Scaling 그룹의 최소 크기입니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupMaxSize	Auto Scaling 그룹의 최대 크기입니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupDesiredCapacity	Auto Scaling 그룹에서 유지 관리를 시도하는 인스턴스의 수입니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupInServiceInstances	Auto Scaling 그룹의 일부로 실행되는 인스턴스의 수입니다. 이 지표에는 보류 중이거나 종료되는 인스턴스가 포함되지 않습니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupPendingInstances	보류 중인 인스턴스의 수입니다. 보류 중인 인터페이스는 아직 서비스되지 않습니다. 이 지표에는 서비스되거나 종료되는 인스턴스가 포함되지 않습니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupStandbyInstances	Standby 상태에 있는 인스턴스의 수입니다. 이 상태의 인스턴스는 계속해서 실행되지만 적극적으로 서비스되지는 않습니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupTerminatingInstances	종료 과정이 진행 중인 인스턴스의 수입니다. 이 지표에는 서비스되거나 보류 중인 인스턴스가 포함되지 않습니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupTotalInstances	Auto Scaling 그룹의 인스턴스 총 수입니다. 이 지표는 서비스되거나, 보류 중이거나, 종료되는 인스턴스의 수를 식별합니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.

AWS/AutoScaling 네임스페이스에는 [인스턴스 가중치 부여 \(p. 74\)](#) 기능을 사용하는 그룹에 대한 다음과 같은 지표가 포함되어 있습니다. 인스턴스 가중치가 적용되지 않을 경우, 다음 지표는 앞의 표에 정의된 해당 지표와 같습니다.

지표	설명
GroupInServiceCapacity	Auto Scaling 그룹의 일부로 실행 중인 용량 단위의 수입니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupPendingCapacity	보류 중인 용량 단위의 수입니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupStandbyCapacity	standby 상태에 있는 용량 단위의 수입니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupTerminatingCapacity	종료 과정이 진행 중인 용량 단위의 수입니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.
GroupTotalCapacity	Auto Scaling 그룹의 총 용량 단위 수입니다. 보고 기준: 지표 모음이 활성화된 경우 보고됩니다.

Auto Scaling 그룹 지표의 차원

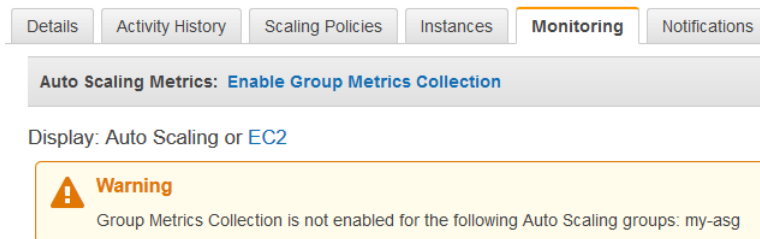
그룹 이름을 기준으로 Auto Scaling 그룹의 지표를 필터링하려면 `AutoScalingGroupName` 차원을 사용하십시오.

Auto Scaling 그룹 지표 활성화

Auto Scaling 그룹 지표를 활성화하면 Auto Scaling이 매분 CloudWatch로 샘플링된 데이터를 전송합니다.

그룹 메트릭을 활성화하려면(콘솔)

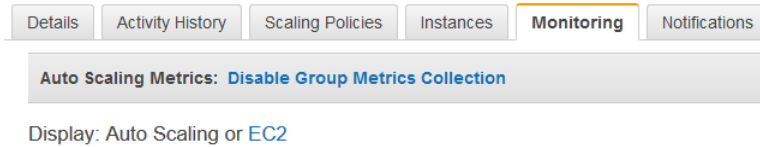
1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. [Monitoring] 탭에서 [Auto Scaling Metrics]에 대하여 [Enable Group Metrics Collection]을 선택합니다. 이 옵션이 보이지 않으면 [Display]에서 [Auto Scaling]을 선택합니다.



그룹 지표를 비활성화하려면(콘솔)

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
3. Auto Scaling 그룹을 선택합니다.

4. [Monitoring] 탭에서 [Auto Scaling Metrics]에 대하여 [Disable Group Metrics Collection]을 선택합니다. 이 옵션이 보이지 않으면 [Display]에서 [Auto Scaling]을 선택합니다.



그룹 메트릭을 활성화하려면(AWS CLI)

`enable-metrics-collection` 명령을 사용하여 한 개 또는 여러 개 그룹 지표를 활성화합니다. 예를 들어, 다음 명령은 GroupDesiredCapacity 지표를 활성화합니다.

```
aws autoscaling enable-metrics-collection --auto-scaling-group-name my-asg \
--metrics GroupDesiredCapacity --granularity "1Minute"
```

--metrics 옵션을 생략하면 모든 지표가 활성화됩니다.

```
aws autoscaling enable-metrics-collection --auto-scaling-group-name my-asg \
--granularity "1Minute"
```

그룹 지표를 비활성화하려면(AWS CLI)

`disable-metrics-collection` 명령을 사용합니다. 예를 들어, 다음 명령은 모든 Auto Scaling 그룹 지표를 비활성화합니다.

```
aws autoscaling disable-metrics-collection --auto-scaling-group-name my-asg
```

Auto Scaling 인스턴스에 대한 모니터링 구성

시작 구성이나 템플릿을 사용하여 EC2 인스턴스에 대한 모니터링을 구성할 수 있습니다. 모니터링은 인스턴스가 시작될 때마다 활성화되며 기본 모니터링(5분 간격) 또는 세부 모니터링(1분 간격)이 적용됩니다. 세부 모니터링의 경우 추가 요금이 부과됩니다. 자세한 내용은 [Amazon CloudWatch 요금](#)을 참조하십시오.

Note

기본적으로 기본 모니터링은 시작 템플릿을 만들거나, AWS Management 콘솔을 사용하여 시작 구성을 생성할 때 활성화됩니다. AWS CLI 또는 SDK를 사용하여 시작 구성을 생성하면 기본적으로 세부 모니터링이 활성화됩니다.

새 EC2 인스턴스에서 활성화되는 모니터링 유형을 변경하려면 시작 템플릿을 업데이트하거나, Auto Scaling 그룹을 업데이트하여 새로운 시작 구성을 사용하게 합니다. 기존 인스턴스는 이전에 활성화된 모니터링 유형을 계속 사용합니다. 모든 인스턴스를 업데이트하려면 인스턴스를 종료해서 Auto Scaling 그룹에서 교체하게 하거나, `monitor-instances` 및 `unmonitor-instances`를 사용하여 인스턴스를 개별적으로 업데이트하십시오.

Auto Scaling 그룹과 연결된 CloudWatch 경보가 있는 경우 `put-metric-alarm` 명령을 사용하여 각 경보를 업데이트하십시오. 각 기간이 모니터링 유형과 맞는지 확인하십시오(기본 모니터링의 경우 300초, 세부 모니터링의 경우 60초). 세부 모니터링에서 기본 모니터링으로 변경하면서 5분 기간과 일치하도록 경보를 업데이트하지 않으면 경보가 계속해서 1분마다 통계를 검사합니다. 따라서 전체 5개 기간 중 4개에서 사용 가능한 데이터를 찾지 못할 수 있습니다.

CloudWatch 모니터링을 구성하려면(콘솔)

AWS Management 콘솔을 사용하여 시작 구성을 생성할 경우 세부 정보 구성 페이지에서 CloudWatch 세부 모니터링 활성화를 선택합니다. 그렇지 않으면 기본 모니터링이 활성화됩니다. 자세한 내용은 [시작 구성 만들기 \(p. 34\)](#) 단원을 참조하십시오.

AWS Management 콘솔을 사용하여 시작 템플릿에 대한 세부 모니터링을 활성화하려면 모니터링의 Advanced Details(고급 세부 정보) 섹션에서 활성화를 선택합니다. 그렇지 않으면 기본 모니터링이 활성화됩니다. 자세한 내용은 [Auto Scaling 그룹을 위한 시작 템플릿 만들기 \(p. 26\)](#) 단원을 참조하십시오.

CloudWatch 모니터링을 구성하려면(AWS CLI)

시작 구성의 경우 `create-launch-configuration` 명령과 `--instance-monitoring` 옵션을 사용합니다. 이 옵션을 `true`로 설정하여 세부 모니터링을 활성화하거나 `false`로 설정하여 기본 모니터링을 활성화합니다.

```
--instance-monitoring Enabled=true
```

시작 템플릿의 경우 `create-launch-template` 명령을 사용하고 시작 템플릿 생성을 위한 정보를 포함하는 JSON 파일을 전달합니다. 모니터링 특성을 `"Monitoring":{"Enabled":true}`로 설정하여 세부 모니터링을 활성화하거나 `"Monitoring":{"Enabled":false}`로 설정하여 기본 모니터링을 활성화합니다.

CloudWatch 지표 보기

Amazon EC2 콘솔을 사용해 Auto Scaling 그룹과 인스턴스에 대한 CloudWatch 지표를 볼 수 있습니다. 이 측정치들은 모니터링 그래프로 표시됩니다.

혹은 CloudWatch 콘솔을 사용해 이런 지표를 볼 수 있습니다.

Amazon EC2 콘솔을 사용한 메트릭 확인

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 [Auto Scaling Groups]를 선택합니다.
3. Auto Scaling 그룹을 선택합니다.
4. [Monitoring] 탭을 선택합니다.
5. (선택 사항) 시간을 기준으로 결과를 필터링하려면 [Showing data for]에서 시간 범위를 선택합니다.
6. 그룹별 지표를 보려면 [Display]에서 [Auto Scaling]을 선택합니다. 단일 지표를 크게 보려면 그래프를 선택합니다. 다음 지표를 사용할 수 있는 그룹은 다음과 같습니다.
 - 최소 그룹 크기 - GroupMinSize—
 - 최대 그룹 크기 - GroupMaxSize—
 - 원하는 용량 - GroupDesiredCapacity—
 - 서비스 상태의 인스턴스 - GroupInServiceInstances—
 - 보류 중인 인스턴스 - GroupPendingInstances—
 - 대기 상태의 인스턴스 - GroupStandbyInstances—
 - 종료 중인 인스턴스 - GroupTerminatingInstances—
 - 총 인스턴스 - GroupTotalInstances—
7. 인스턴스별 지표를 보려면 [Display]에서 [EC2]를 선택합니다. 단일 지표를 크게 보려면 그래프를 선택합니다. 다음 지표를 사용할 수 있는 인스턴스는 다음과 같습니다.
 - CPU 사용률 - CPUUtilization—
 - 디스크 읽기 - DiskReadBytes—
 - 디스크 읽기 연산 - DiskReadOps—
 - 디스크 쓰기 - DiskWriteBytes—
 - 디스크 쓰기 연산 - DiskWriteOps—
 - 네트워크 입력 - NetworkIn—

- 네트워크 출력 - NetworkOut—
- 상태 확인 실패(해당되는 경우) - StatusCheckFailed—
- 상태 확인 실패(인스턴스) - StatusCheckFailed_Instance—
- 상태 확인 실패(시스템) - StatusCheckFailed_System—

CloudWatch 콘솔을 사용한 메트릭 확인

자세한 내용은 [Auto Scaling 그룹별 통계 집계](#)를 참조하십시오.

CloudWatch 지표를 보는 방법(AWS CLI)

모든 Auto Scaling 그룹에 대한 지표를 모두 보려면 다음 [list-metrics](#) 명령을 사용합니다.

```
aws cloudwatch list-metrics --namespace "AWS/AutoScaling"
```

단일 Auto Scaling 그룹의 지표를 보려면 다음과 같이 `AutoScalingGroupName` 차원을 지정합니다.

```
aws cloudwatch list-metrics --namespace "AWS/AutoScaling" --dimensions  
Name=AutoScalingGroupName,Value=my-asg
```

모든 Auto Scaling 그룹의 단일 지표를 보려면 다음과 같이 해당 지표의 이름을 지정합니다.

```
aws cloudwatch list-metrics --namespace "AWS/AutoScaling" --metric-name  
GroupDesiredCapacity
```

Amazon CloudWatch 경보 만들기

지표를 모니터링하는 한 가지 목적은 응용 프로그램이 예상대로 작동하는지 확인하는 것입니다. 지표가 허용되는 임계값을 초과하는 경우 CloudWatch 경보로 조치를 트리거할 수 있습니다. CloudWatch가 지원하는 모든 경보 조치를 지정할 수 있습니다.

모니터링하는 지표를 확인하여 경보를 구성합니다. 예를 들어, Auto Scaling 그룹에서 EC2 인스턴스의 평균 CPU 사용량을 감시하도록 경보를 구성할 수 있습니다. 해당 조치는 그룹의 평균 CPU 사용이 지정한 연속 기간에 대해 지정한 임계값을 위반할 때 사용자에게 전송되는 통지가 될 수 있습니다. 예를 들어, 지표가 1분마다 4 연속 기간 동안 70% 이상으로 유지되는 경우.

자세한 내용은 Amazon CloudWatch 사용 설명서의 [Amazon CloudWatch 경보 사용](#)을 참조하십시오.

평균 CPU 사용률에 기반하여 CloudWatch 경보를 만들려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 경보, 경보 생성을 선택합니다.
3. 지표 선택과 EC2를 선택합니다.
4. 결과를 필터링하기 위한 지표 범주를 선택합니다. Auto Scaling 그룹 지표를 보려면 Auto Scaling 그룹별을 선택합니다. 개별 인스턴스에 대한 지표만 필요하면 인스턴스별 지표를 선택하십시오.
5. 아래와 같이 지표를 선택합니다.
 - a. 경보를 작성하려는 Auto Scaling 그룹 또는 인스턴스가 들어있는 행과 CPUUtilization 지표를 선택하십시오.
 - b. [Graphed metrics] 탭을 선택합니다.
 - c. 통계의 경우 평균을 선택합니다.
 - d. 기간에서 예를 들어 1분과 같은 경보에 대한 평가 기간을 선택합니다. 경보를 평가할 때 각 기간이 하나의 데이터 포인트로 집계됩니다.

Note

시간이 짧을수록 경보가 더 민감해집니다.

- e. 지표 선택을 선택합니다.
6. 조건에서 임계 조건을 정의하여 경보를 정의합니다. 예를 들어, 지표 값이 70% 이상일 때마다 경보를 트리거하는 임계값을 정의할 수 있습니다.
7. 추가 구성에서 경고할 데이터포인트에 대해 알람을 트리거하는 ALARM 상태에 있어야 하는 데이터포인트(평가 기간)의 수를 지정합니다(예: 4개 중 4개). 그러면 다수의 연속 기간이 위반되면 ALARM 상태가 되는 경보가 생성됩니다.
8. 누락된 데이터 처리에 대해서는 옵션 중 하나를 선택하십시오. CPUUtilization과 같이 지속적으로 데이터를 보고하는 지표의 경우, 누락된 데이터포인트는 무엇인가가 잘못되었음을 나타낼 수 있으므로 누락된 데이터를 잘못된 데이터로 처리를 선택하는 것이 좋습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서의 [CloudWatch 경보가 누락된 데이터를 처리하는 방법 구성](#) 단원을 참조하십시오.
9. [Next]를 선택합니다.
10. 조치 구성에서 수행할 조치를 정의합니다.
11. [Next]를 선택합니다.
12. 설명 추가에서 경보의 이름과 설명을 입력하고 다음을 선택하십시오.
13. 경보 생성을 선택합니다.

AWS CloudTrail를 사용하여 Amazon EC2 Auto Scaling API 호출 로깅

Amazon EC2 Auto Scaling은 Amazon EC2 Auto Scaling을 사용하는 사용자, 역할 또는 AWS 서비스가 수행한 작업에 대한 레코드를 제공하는 서비스인 AWS CloudTrail과 통합됩니다. CloudTrail은 Amazon EC2 Auto Scaling에 대한 모든 API를 이벤트로 캡처합니다. 캡처되는 호출에는 Amazon EC2 Auto Scaling 콘솔로부터의 호출과 Amazon EC2 Auto Scaling API에 대한 코드 호출이 포함됩니다.

추적을 생성하면 Amazon EC2 Auto Scaling 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 Auto Scaling에 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](#)를 참조하십시오.

CloudTrail의 Amazon EC2 Auto Scaling 정보

CloudTrail은 계정 생성 시 AWS 계정에서 활성화됩니다. Amazon EC2 Auto Scaling에서 활동이 수행되면 해당 활동은 이벤트 기록에서 다른 AWS 서비스 이벤트와 함께 CloudTrail 이벤트 로그에 기록됩니다. AWS 계정에서 최신 이벤트를 확인, 검색 및 다운로드할 수 있습니다. 자세한 내용은 [CloudTrail 이벤트 기록에서 이벤트 보기](#)를 참조하십시오.

Amazon EC2 Auto Scaling 이벤트를 포함하여 AWS 계정에 이벤트를 지속적으로 기록하려는 경우 추적을 생성합니다. 추적은 CloudTrail이 Amazon 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 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 또는 CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 정보는 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 Auto Scaling 작업이 CloudTrail에서 로깅되고 [Amazon EC2 Auto Scaling API 참조](#)에 문서화됩니다. 예를 들어, CreateLaunchConfiguration, DescribeAutoScalingGroup, UpdateAutoScalingGroup 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 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 userIdentity](#)요소를 참조하십시오.

Amazon EC2 Auto Scaling 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 포함되어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 호출의 주문 스택 추적이 아니므로 특정 순서로 표시되지 않습니다.

다음은 CreateLaunchConfiguration 작업을 보여 주는 CloudTrail 로그 항목이 나타난 예제입니다.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "Root",
    "principalId": "123456789012",
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-08-21T17:05:42Z"
      }
    }
  },
  "eventTime": "2018-08-21T17:07:49Z",
  "eventSource": "autoscaling.amazonaws.com",
  "eventName": "CreateLaunchConfiguration",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "Coral/Jakarta",
  "requestParameters": {
    "ebsOptimized": false,
    "instanceMonitoring": {
      "enabled": false
    },
    "instanceType": "t2.micro",
    "keyName": "EC2-key-pair-oregon",
    "blockDeviceMappings": [
      {
        "deviceName": "/dev/xvda",
        "ebs": {
          "deleteOnTermination": true,
          "volumeSize": 8,
          "snapshotId": "snap-01676e0a2c3c7de9e",

```

```
        "volumeType": "gp2"
      }
    },
    "launchConfigurationName": "launch_configuration_1",
    "imageId": "ami-6cd6f714d79675a5",
    "securityGroups": [
      "sg-00c429965fd921483"
    ]
  },
  "responseElements": null,
  "requestID": "0737e2ea-fb2d-11e3-bfd8-99133058e7bb",
  "eventID": "3fcfb182-98f8-4744-bd45-b38835ab61cb",
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

Amazon EC2 Auto Scaling에 대한 Personal Health Dashboard 알림

Personal Health Dashboard(PHD)는 Amazon EC2 Auto Scaling에서 보내는 알림에 대한 지원을 제공합니다. 이러한 알림은 애플리케이션에 영향을 줄 수 있는 리소스 성능 또는 가용성 문제를 알리고 수정 지침을 제시합니다. 지금은 누락된 보안 그룹 및 시작 템플릿과 관련된 이벤트만 사용할 수 있습니다.

Personal Health Dashboard는 AWS Health 서비스의 일부입니다. 설정이 필요하지 않으며, 계정에서 인증된 사용자면 누구나 볼 수 있습니다. 자세한 내용은 [AWS Personal Health Dashboard 시작하기](#) 단원을 참조하십시오.

다음 메시지와 유사한 메시지를 받으면 조치를 취하기 위한 경보로 처리해야 합니다.

예: 누락된 보안 그룹 때문에 Auto Scaling 그룹이 확장되지 않습니다.

Hello,

At 2020-01-11 04:00 UTC, we detected an issue with your Auto Scaling group [ARN] in AWS account 123456789012.

A security group associated with this Auto Scaling group cannot be found. Each time a scale out operation is performed, it will be prevented until you make a change that fixes the issue.

We recommend that you review and update your Auto Scaling group configuration to change the launch template or launch configuration that depends on the unavailable security group.

Sincerely,
Amazon Web Services

예: 누락된 시작 템플릿 때문에 Auto Scaling 그룹이 확장되지 않습니다.

Hello,

At 2020-01-11 04:00 UTC, we detected an issue with your Auto Scaling group [ARN] in AWS account 123456789012.

The launch template associated with this Auto Scaling group cannot be found. Each time a scale out operation is performed, it will be prevented until you make a change that fixes the issue.

We recommend that you review and update your Auto Scaling group configuration and specify an existing launch template to use.

Sincerely,
Amazon Web Services

Amazon EC2 Auto Scaling의 보안

AWS에서는 클라우드 보안을 가장 중요하게 생각합니다. AWS 고객은 보안에 매우 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 EC2 Auto Scaling에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 [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 EC2 Auto Scaling 사용 시 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 항목에서는 보안 및 규정 준수 목표를 충족하도록 Amazon EC2 Auto Scaling을 구성하는 방법을 보여줍니다. 또한 Amazon EC2 Auto Scaling 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법을 알아봅니다.

주제

- [Amazon EC2 Auto Scaling 및 데이터 보호 \(p. 170\)](#)
- [Amazon EC2 Auto Scaling의 ID 및 액세스 관리 \(p. 171\)](#)
- [Amazon EC2 Auto Scaling의 규정 준수 확인 \(p. 196\)](#)
- [Amazon EC2 Auto Scaling의 복원성 \(p. 197\)](#)
- [Amazon EC2 Auto Scaling의 인프라 보안 \(p. 197\)](#)
- [Amazon EC2 Auto Scaling 및 인터페이스 VPC 엔드포인트 \(p. 198\)](#)

Amazon EC2 Auto Scaling 및 데이터 보호

Amazon EC2 Auto Scaling에서는 AWS **공동 책임 모델**을 준수하며, 이 모델에는 데이터 보호에 대한 규정과 지침이 포함되어 있습니다. AWS는 모든 AWS 서비스를 실행하는 글로벌 인프라의 보호를 책임집니다. AWS는 고객 콘텐츠 및 개인 데이터의 처리를 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 올린 모든 개인 데이터를 책임집니다.

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.

이름 및 태그와 같은 메타데이터나 자유 형식 필드에 고객의 계정 번호와 같은 중요한 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.

AWS KMS를 사용하여 데이터 암호화

AWS Key Management Service 고객 마스터 키(CMK)를 사용하여 클라우드에 저장된 Amazon EBS 볼륨 데이터를 암호화하도록 Auto Scaling 그룹을 구성할 수 있습니다. Amazon EC2 Auto Scaling은 데이터를 암호화하기 위해 AWS 관리형 및 고객 관리형 CMK를 지원합니다. 시작 구성을 사용할 때는 고객 관리형 CMK를 지정하는 `KmsKeyId` 옵션을 사용할 수 없습니다. 대신 시작 템플릿을 사용하십시오. 자세한 내용은 [Auto Scaling 그룹을 위한 시작 템플릿 만들기 \(p. 26\)](#) 및 [암호화된 볼륨에 사용할 경우 필요한 CMK 키 정책 \(p. 193\)](#) 단원을 참조하십시오. 기본적으로 암호화를 사용하여 새 EBS 볼륨 및 스냅샷 복사본의 암호화를 적용할 수도 있습니다.

EBS 볼륨 및 스냅샷 등 Amazon EC2 서비스 내에서 관리되는 데이터 보호에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2의 데이터 보호](#) 및 [기본 암호화](#)를 참조하십시오.

AWS KMS에 대한 자세한 내용은 [AWS Key Management Service란 무엇입니까?](#) 단원을 참조하십시오.

관련 항목

- Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2에서 데이터 보호](#)

Amazon EC2 Auto Scaling의 ID 및 액세스 관리

AWS Identity and Access Management(IAM)는 관리자가 AWS 리소스에 대한 액세스를 안전하게 제어할 수 있도록 지원하는 AWS 서비스입니다. IAM 관리자는 Amazon EC2 Auto Scaling 리소스를 사용하기 위해 인증(로그인) 및 권한 부여(권한 있음)할 수 있는 사람을 제어합니다. IAM은 추가 비용 없이 사용할 수 있는 AWS 서비스입니다.

Amazon EC2 Auto Scaling을 사용하려면 AWS 계정과 AWS 자격 증명이 필요합니다. AWS 계정의 보안을 강화하려면 AWS 계정 자격 증명 대신 IAM 사용자를 사용해 액세스 자격 증명을 제공하는 것이 좋습니다. 자세한 내용은 [AWS 계정 루트 사용자 자격 증명과 IAM 사용자 자격 증명](#)(AWS 일반 참조)과 [IAM 모범 사례](#)(IAM 사용 설명서)를 참조하십시오.

IAM 사용자 개요 및 해당 사용자가 계정의 보안에 중요한 이유에 대해서는 AWS 일반 참조의 [AWS 보안 자격 증명](#)을 참조하십시오.

IAM 작업에 대한 자세한 내용은 [IAM 사용 설명서](#) 단원을 참조하십시오.

액세스 제어

요청을 인증하는 데 유효한 자격 증명이 있더라도 권한이 없다면 Amazon EC2 Auto Scaling 리소스를 생성하거나 액세스할 수 없습니다. 예를 들어 Auto Scaling 그룹을 생성하고 시작 구성을 생성하는 등의 권한이 있어야 합니다.

다음 단원에서는 IAM 관리자가 Amazon EC2 Auto Scaling 작업을 수행할 수 있는 대상을 제어하여 Amazon EC2 Auto Scaling 리소스를 보호할 수 있도록 IAM을 사용하는 방법에 대한 세부 정보를 제공합니다.

먼저 Amazon EC2 항목을 읽는 것이 좋습니다. Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2의 ID 및 액세스 관리](#) 단원을 참조하십시오. 이 단원의 각 항목을 읽으면 Amazon EC2에서 제공하는 액세스 제어 권한은 무엇인지, Amazon EC2 Auto Scaling 리소스 권한과 어떻게 결합될 수 있는지 잘 알 수 있을 것입니다.

주제

- [Amazon EC2 Auto Scaling에서 IAM을 사용하는 방법 \(p. 172\)](#)
- [Amazon EC2 Auto Scaling에 대한 서비스 연결 역할 \(p. 176\)](#)
- [Amazon EC2 Auto Scaling 자격 증명 기반 정책 예제 \(p. 179\)](#)
- [시작 템플릿 지원 \(p. 188\)](#)
- [Amazon EC2 인스턴스에서 실행되는 애플리케이션의 IAM 역할 \(p. 192\)](#)
- [암호화된 볼륨에 사용할 경우 필요한 CMK 키 정책 \(p. 193\)](#)

Amazon EC2 Auto Scaling에서 IAM을 사용하는 방법

IAM을 사용하여 Amazon EC2 Auto Scaling에 대한 액세스를 관리하려면 먼저 어떤 IAM 기능을 Amazon EC2 Auto Scaling에 사용할 수 있는지를 이해해야 합니다. Amazon EC2 Auto Scaling 및 기타 AWS 서비스에서 IAM을 사용하는 방법을 전체적으로 알아보려면 IAM 사용 설명서의 [IAM으로 작업하는 AWS 서비스](#)를 참조하십시오.

Note

사용자가 Auto Scaling 그룹을 구성할 수 있으려면 특정 IAM 권한 및 Amazon EC2 Auto Scaling 서비스 연결 역할이 필요합니다.

주제

- [Amazon EC2 Auto Scaling 자격 증명 기반 정책 \(p. 172\)](#)
- [Amazon EC2 Auto Scaling 리소스 기반 정책 \(p. 175\)](#)
- [ACL\(액세스 제어 목록\) \(p. 175\)](#)
- [Amazon EC2 Auto Scaling 태그 기반 권한 부여 \(p. 175\)](#)
- [Amazon EC2 Auto Scaling IAM 역할 \(p. 175\)](#)

Amazon EC2 Auto Scaling 자격 증명 기반 정책

IAM 자격 증명 기반 정책을 사용하면 허용되거나 거부되는 작업을 지정할 수 있으며, 작업이 허용되거나 거부되는 조건도 지정할 수 있습니다. Amazon EC2 Auto Scaling은 특정 작업, 리소스 및 조건 키를 지원합니다. JSON 정책에서 사용하는 모든 요소에 대해 알아보려면 IAM 사용 설명서의 [IAM JSON 정책 요소 참조](#)를 참조하십시오.

Actions

IAM 자격 증명 기반 정책의 Action 요소는 정책에 따라 허용되거나 거부되는 특정 작업에 대해 설명합니다. 일반적으로 정책 작업의 이름은 연결된 AWS API 작업의 이름과 동일합니다. 이 작업은 연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에서 사용됩니다.

Amazon EC2 Auto Scaling의 정책 작업은 작업 앞에 `autoscaling:` 접두사를 사용합니다. 정책 문에는 Action 또는 NotAction 요소가 포함되어야 합니다. Amazon EC2 Auto Scaling은 이 서비스로 수행할 수 있는 작업을 설명하는 고유한 작업 집합을 정의합니다.

단일 명령문에서 여러 작업을 지정하려면 다음 예제와 같이 쉼표로 구분합니다.

```
"Action": [
    "autoscaling:CreateAutoScalingGroup",
    "autoscaling:UpdateAutoScalingGroup"
```

와일드카드(*)를 사용하여 여러 작업을 지정할 수 있습니다. 예를 들어, `Describe`라는 단어로 시작하는 모든 작업을 지정하려면 다음 작업을 포함합니다.

```
"Action": "autoscaling:Describe"
```

Amazon EC2 Auto Scaling 작업의 목록을 보려면 Amazon EC2 Auto Scaling API 참조의 [작업](#)을 참조하십시오.

리소스

Resource 요소는 작업이 적용되는 객체를 지정합니다. 문에는 Resource 또는 NotResource 요소가 반드시 추가되어야 합니다. ARN을 사용하거나 문이 모든 리소스에 적용됨을 표시하는 와일드카드(*)를 사용하여 리소스를 지정합니다.

ARN을 사용하여 IAM 정책이 적용되는 리소스를 식별하여 특정 Auto Scaling 그룹 및 시작 구성에 대한 액세스를 제한할 수 있습니다. ARN 형식에 대한 자세한 내용은 [Amazon 리소스 이름\(ARN\) 및 AWS 서비스 네임스페이스](#)를 참조하십시오.

Auto Scaling 그룹에는 다음 ARN이 있습니다.

```
"Resource":  
  "arn:aws:autoscaling:region:123456789012:autoScalingGroup:uuid:autoScalingGroupName/asg-name"
```

시작 구성에는 다음 ARN이 있습니다.

```
"Resource":  
  "arn:aws:autoscaling:region:123456789012:launchConfiguration:uuid:launchConfigurationName/lc-name"
```

CreateAutoScalingGroup 작업을 통해 Auto Scaling 그룹을 지정하려면 다음과 같이 UUID를 *로 바꿔야 합니다.

```
"Resource":  
  "arn:aws:autoscaling:region:123456789012:autoScalingGroup:*:autoScalingGroupName/asg-name"
```

CreateLaunchConfiguration 작업을 통해 시작 구성을 지정하려면 다음과 같이 UUID를 *로 바꿔야 합니다.

```
"Resource":  
  "arn:aws:autoscaling:region:123456789012:launchConfiguration:*:launchConfigurationName/lc-name"
```

모든 Amazon EC2 Auto Scaling 작업이 리소스 수준 권한을 지원하는 것은 아닙니다. 리소스 수준 권한을 지원하지 않는 작업의 경우 "*"를 리소스로 사용해야 합니다.

다음 Amazon EC2 Auto Scaling 작업은 리소스 수준 권한을 지원하지 않습니다.

- DescribeAccountLimits
- DescribeAdjustmentTypes
- DescribeAutoScalingGroups
- DescribeAutoScalingInstances
- DescribeAutoScalingNotificationTypes
- DescribeLaunchConfigurations
- DescribeLifecycleHooks
- DescribeLifecycleHookTypes
- DescribeLoadBalancers

- DescribeLoadBalancerTargetGroups
- DescribeMetricCollectionTypes
- DescribeNotificationConfigurations
- DescribePolicies
- DescribeScalingActivities
- DescribeScalingProcessTypes
- DescribeScheduledActions
- DescribeTags
- DescribeTerminationPolicyTypes

각 리소스의 ARN을 지정할 수 있는 작업에 대해 알아보려면 IAM 사용 설명서의 [Amazon EC2 Auto Scaling의 작업, 리소스 및 조건 키](#)를 참조하십시오.

조건 키

Condition 요소(또는 Condition 블록)를 사용하면 정책이 발효되는 조건을 지정할 수 있습니다. Condition 요소는 선택 사항입니다. 같음, 미만 등 [조건 연산자](#)를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 빌드할 수 있습니다.

한 문에서 여러 Condition 요소를 지정하거나 단일 Condition 요소에서 여러 키를 지정하는 경우 AWS는 논리적 AND 작업을 사용하여 평가합니다. 단일 조건 키의 여러 값을 지정하는 경우 AWS는 논리적 OR 작업을 사용하여 조건을 평가합니다. 문의 권한을 부여하기 전에 모든 조건을 충족해야 합니다.

조건을 지정할 때 자리 표시자 변수를 사용할 수도 있습니다. 예를 들어, IAM 사용자에게 IAM 사용자 이름으로 태그가 지정된 경우에만 리소스에 액세스할 수 있는 권한을 부여할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 [IAM 정책 요소: 변수 및 태그](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling에서는 자체 조건 키 세트를 정의하고 일부 전역 조건 키 사용도 지원합니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 [AWS 전역 조건 컨텍스트 키](#)를 참조하십시오.

다음은 Amazon EC2 Auto Scaling 전용 조건 키입니다.

- autoscaling:ImageId
- autoscaling:InstanceType
- autoscaling:InstanceTypes
- autoscaling:LaunchConfigurationName
- autoscaling:LaunchTemplateVersionSpecified
- autoscaling:LoadBalancerNames
- autoscaling:MaxSize
- autoscaling:MinSize
- autoscaling:ResourceTag/*key*
- autoscaling:SpotPrice
- autoscaling:TargetGroupARNs
- autoscaling:VPCZoneIdentifiers

조건 키를 사용할 수 있는 작업과 리소스를 알아보려면 IAM 사용 설명서의 [Amazon EC2 Auto Scaling가 정의한 작업](#)을 참조하십시오.

예제

Amazon EC2 Auto Scaling 자격 증명 기반 정책의 예를 보려면 [Amazon EC2 Auto Scaling 자격 증명 기반 정책 예제 \(p. 179\)](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling 리소스 기반 정책

Amazon EC2 Auto Scaling에서는 리소스 기반 정책을 지원하지 않습니다.

ACL(액세스 제어 목록)

Amazon EC2 Auto Scaling은 ACL(액세스 제어 목록)을 지원하지 않습니다.

Amazon EC2 Auto Scaling 태그 기반 권한 부여

Amazon EC2 Auto Scaling에 대해 생성하는 자격 증명 기반 정책에서 태그 기반의 리소스 수준 권한을 적용할 수 있습니다. 이를 통해 사용자가 생성, 수정, 사용 또는 삭제할 수 있는 리소스를 더욱 정확하게 제어할 수 있습니다.

IAM 정책에 태그를 사용하려면 다음 조건 키를 사용하여 정책의 [조건 요소](#)에 태그 정보를 제공합니다.

- 특정 태그가 있는 Auto Scaling 그룹에 대한 사용자 작업을 허용(또는 거부)하려면 `autoscaling:ResourceTag`/`tag-key`: `tag-value`를 사용합니다.
- 요청에 특정 태그가 존재하도록 (또는 존재하지 않도록) 요구하려면 `aws:RequestTag`/`tag-key`: `tag-value`를 사용합니다.
- 요청에 특정 태그 키가 존재하도록 (또는 존재하지 않도록) 요구하려면 `aws:TagKeys` [`tag-key`, ...]를 사용합니다.

Auto Scaling 그룹의 태그를 기반으로 Auto Scaling 그룹에 대한 액세스를 제어하는 정책 예제를 보려면 [삭제할 수 있는 조정 정책 제어](#) (p. 183) 단원을 참조하십시오. 태그를 기반으로 액세스를 제어하는 다른 예는 [Amazon EC2 Auto Scaling 자격 증명 기반 정책 예제](#) (p. 179) 단원을 참조하십시오.

Amazon EC2 Auto Scaling IAM 역할

IAM 역할은 특정 권한을 가지고 있는 AWS 계정 내 개체입니다.

Amazon EC2 Auto Scaling에서 임시 자격 증명 사용

임시 자격 증명을 사용하여 페더레이션을 통해 로그인하거나, IAM 역할을 수임하거나, 교차 계정 역할을 수임할 수 있습니다. [AssumeRole](#) 또는 [GetFederationToken](#) 같은 AWS STS API 작업을 호출하여 임시 보안 자격 증명을 가져옵니다.

Amazon EC2 Auto Scaling은 임시 자격 증명 사용을 지원합니다.

서비스 연결 역할

서비스 연결 역할을 사용하면 AWS 서비스에서 다른 서비스의 리소스에 액세스하여 사용자 대신 작업을 완료할 수 있습니다. 서비스 연결 역할은 IAM 계정에 나타나고, 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할의 권한을 볼 수 있지만 편집할 수는 없습니다.

Amazon EC2 Auto Scaling은 서비스 연결 역할을 지원합니다. Amazon EC2 Auto Scaling 서비스 연결 역할을 생성 또는 관리하는 방법에 대한 자세한 내용은 [Amazon EC2 Auto Scaling에 대한 서비스 연결 역할](#) (p. 176) 단원을 참조하십시오.

서비스 역할

이 기능을 사용하면 서비스가 사용자를 대신해 **서비스 역할**을 맡을 수 있습니다. 이 역할을 사용하면 서비스는 다른 서비스의 리소스에 액세스해 사용자를 대신해 작업을 완료할 수 있습니다. IAM 관리자가 이 역할에 대한 권한을 변경할 수 있습니다. 그러나 권한을 변경하면 서비스의 기능이 손상될 수 있습니다.

Amazon EC2 Auto Scaling은 수명 주기 후크 알림을 위한 서비스 역할을 지원합니다. 자세한 내용은 [Amazon EC2 Auto Scaling 수명 주기 후크](#) (p. 131) 단원을 참조하십시오.

Amazon EC2 Auto Scaling에서 IAM 역할 선택

Amazon EC2에서 실행되는 애플리케이션이 수임할 수 있는 IAM 역할을 이전에 생성한 경우 시작 템플릿 또는 시작 구성을 생성할 때 이 역할을 선택할 수 있습니다. Amazon EC2 Auto Scaling은 선택할 수 있는 역할 목록을 제공합니다. 이러한 역할을 생성할 때 애플리케이션에 필요한 특정 API 호출에 대한 액세스를 제한하는 최소 권한 IAM 정책을 연결하는 것이 중요합니다. 자세한 내용은 [Amazon EC2 인스턴스에서 실행되는 애플리케이션의 IAM 역할 \(p. 192\)](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling에 대한 서비스 연결 역할

Amazon EC2 Auto Scaling은 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. 또한 서비스 연결 역할을 사용하면 AWS CloudTrail을 통해 모든 API 호출을 볼 수 있습니다. Amazon EC2 Auto Scaling이 사용자를 대신하여 수행하는 모든 작업을 추적할 수 있기 때문에 요구 사항 모니터링 및 감사에 유용합니다. 자세한 내용은 [AWS CloudTrail를 사용하여 Amazon EC2 Auto Scaling API 호출 로깅 \(p. 166\)](#) 단원을 참조하십시오.

다음 단원에서는 Amazon EC2 Auto Scaling 서비스 연결 역할을 생성하고 관리하는 방법에 대해 설명합니다. 먼저 IAM 엔티티(사용자, 그룹, 역할 등)가 서비스 연결 역할을 생성, 편집 또는 삭제할 수 있도록 권한을 구성해야 합니다. 자세한 정보는 IAM 사용 설명서의 [서비스 연결 역할 사용](#)을 참조하십시오.

개요

Amazon EC2 Auto Scaling 서비스 연결 역할에는 두 가지 유형이 있습니다.

- 이름이 `AWSServiceRoleForAutoScaling`인 계정의 기본 서비스 연결 역할. 다른 서비스 연결 역할을 지정하지 않은 경우 이 역할이 Auto Scaling 그룹에 자동으로 할당됩니다.
- 역할을 생성할 때 지정하는 사용자 지정 접미사가 있는 서비스 연결 역할(예: `AWSServiceRoleForAutoScaling_mysuffix`).

사용자 지정 접미사 서비스 연결 역할의 권한은 기본 서비스 연결 역할의 권한과 동일합니다. 두 경우 모두 역할을 편집할 수 없으며 Auto Scaling 그룹에서 사용 중인 역할은 삭제할 수 없습니다. 유일한 차이점은 역할 이름 접미사입니다.

AWS Key Management Service 키 정책을 편집할 때 Amazon EC2 Auto Scaling에서 시작한 인스턴스를 고객 관리형 CMK로 암호화할 수 있도록 역할을 지정할 수 있습니다. 그러나 특정 고객 관리형 CMK에 세분화된 액세스를 제공하려는 경우 사용자 지정 접미사 서비스 연결 역할을 사용해야 합니다. 사용자 지정 접미사 서비스 연결 역할은 다음 기능을 제공합니다.

- CMK에 대한 향상된 제어 기능
- CloudTrail 로그에서 API를 호출한 Auto Scaling 그룹을 추적하는 기능

일부 사용자만 액세스할 수 있는 고객 관리형 CMK를 생성하는 경우 다음 단계를 통해 사용자 지정 접미사 서비스 연결 역할을 사용할 수 있습니다.

1. 사용자 지정 접미사를 사용하여 서비스 연결 역할을 생성합니다. 자세한 내용은 [서비스 연결 역할 생성\(수동\) \(p. 177\)](#) 단원을 참조하십시오.
2. 서비스 연결 역할에 고객 관리형 CMK에 대한 액세스 권한을 부여합니다. 서비스 연결 역할에서 CMK를 사용할 수 있도록 허용하는 키 정책에 대한 자세한 내용은 [암호화된 볼륨에 사용할 경우 필요한 CMK 키 정책 \(p. 193\)](#) 단원을 참조하십시오.
3. IAM 사용자 또는 역할에 사용자가 생성한 서비스 연결 역할에 대한 액세스 권한을 부여합니다. IAM 정책 생성에 대한 자세한 내용은 [전달할 수 있는 서비스 연결 역할 제한\(PassRole 사용\) \(p. 186\)](#) 단원을 참조하십시오.

하십시오. 사용자가 해당 역할을 서비스에 전달할 권한 없이 서비스 연결 역할을 지정하려고 하면 오류가 발생합니다.

서비스 연결 역할에 의해 부여된 권한

Amazon EC2 Auto Scaling은 AWSServiceRoleForAutoScaling 서비스 연결 역할 또는 사용자 지정 접미사 서비스 연결 역할을 사용하여 사용자를 대신하여 지정된 리소스에 대해 다음 작업을 수행합니다.

- `ec2:AttachClassicLinkVpc`
- `ec2:CancelSpotInstanceRequests`
- `ec2:CreateFleet`
- `ec2:CreateTags`
- `ec2>DeleteTags`
- `ec2:Describe*`
- `ec2:DetachClassicLinkVpc`
- `ec2:ModifyInstanceAttribute`
- `ec2:RequestSpotInstances`
- `ec2:RunInstances`
- `ec2:TerminateInstances`
- `elasticloadbalancing:Register*`
- `elasticloadbalancing:Deregister*`
- `elasticloadbalancing:Describe*`
- `cloudwatch>DeleteAlarms`
- `cloudwatch:DescribeAlarms`
- `cloudwatch:PutMetricAlarm`
- `sns:Publish`

이 역할은 해당 역할을 수입할 `autoscaling.amazonaws.com` 서비스를 신뢰합니다.

서비스 연결 역할 생성(자동)

Amazon EC2 Auto Scaling은 사용자 지정 접미사 서비스 연결 역할을 수동으로 생성하고 그룹을 생성할 때 지정하지 않는 한 Auto Scaling 그룹을 처음 생성할 때 AWSServiceRoleForAutoScaling 서비스 연결 역할을 생성합니다.

Important

서비스 연결 역할을 생성할 IAM 권한이 있어야 합니다. 그러한 권한이 없으면 자동 생성은 실패합니다. 자세한 내용은 IAM 사용 설명서의 [서비스 연결 역할](#)과 이 설명서의 [서비스 연결 역할을 생성하는데 필요한 권한 \(p. 186\)](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling은 2018년 3월 서비스 연결 역할을 지원하기 시작했습니다. 그 이전에 Auto Scaling 그룹을 생성한 경우 Amazon EC2 Auto Scaling에서 고객의 AWS 계정에 AWSServiceRoleForAutoScaling 역할을 생성했습니다. 자세한 내용은 IAM 사용 설명서의 [내 AWS 계정에 표시되는 새 역할](#)을 참조하십시오.

서비스 연결 역할 생성(수동)

서비스 연결 역할을 만들려면(콘솔 사용)

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.

2. 탐색 창에서 [Roles], [Create role]을 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 Auto Scaling 및 EC2 Auto Scaling 사용 사례를 선택합니다.
5. 다음: 권한, 다음: 태그, 다음 검토를 차례로 선택합니다. 참고: 생성하는 동안에는 서비스 연결 역할에 태그를 연결할 수 없습니다.
6. 검토 페이지에서 역할 이름을 비워 두고 이름이 AWSServiceRoleForAutoScaling인 서비스 연결 역할을 생성하거나, 접미사를 입력하여 이름이 AWSServiceRoleForAutoScaling_*suffix*인 서비스 연결 역할을 생성합니다.
7. (선택 사항) Role description(역할 설명)에서 서비스 연결 역할에 대한 설명을 편집합니다.
8. Create role을 선택합니다.

서비스 연결 역할을 만들려면(AWS CLI)

다음 [create-service-linked-role](#) CLI 명령을 사용하여 이름이 AWSServiceRoleForAutoScaling_*suffix*인 Amazon EC2 Auto Scaling에 대한 서비스 연결 역할을 생성합니다.

```
aws iam create-service-linked-role --aws-service-name autoscaling.amazonaws.com --custom-suffix suffix
```

이 명령의 출력에는 서비스 연결 역할의 ARN이 포함되며, 이 정보를 사용하여 CMK에 대한 서비스 연결 역할 액세스를 허용할 수 있습니다.

```
{
  "Role": {
    "RoleId": "ABCDEF0123456789ABCDEF",
    "CreateDate": "2018-08-30T21:59:18Z",
    "RoleName": "AWSServiceRoleForAutoScaling_suffix",
    "Arn": "arn:aws:iam::123456789012:role/aws-service-role/autoscaling.amazonaws.com/AWSServiceRoleForAutoScaling_suffix",
    "Path": "/aws-service-role/autoscaling.amazonaws.com/",
    "AssumeRolePolicyDocument": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Action": [
            "sts:AssumeRole"
          ],
          "Principal": {
            "Service": [
              "autoscaling.amazonaws.com"
            ]
          },
          "Effect": "Allow"
        }
      ]
    }
  }
}
```

자세한 정보는 IAM 사용 설명서의 [서비스 연결 역할 생성](#)을 참조하십시오.

서비스 연결 역할 편집

Amazon EC2 Auto Scaling용으로 생성된 서비스 연결 역할은 편집할 수 없습니다. 서비스 연결 역할을 생성한 후에는 역할의 이름 또는 해당 권한을 변경할 수 없습니다. 그러나 역할의 설명은 편집할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 [서비스에 연결 역할 편집](#) 단원을 참조하십시오.

서비스 연결 역할 삭제

Auto Scaling 그룹을 사용하지 않는 경우 해당 서비스 연결 역할을 삭제하는 것이 좋습니다. 역할을 삭제하면 사용되지 않거나 적극적으로 모니터링 및 유지 관리되지 않는 엔터티를 가질 수 없습니다.

먼저 관련 AWS 리소스를 삭제한 후에만 서비스 연결 역할을 삭제할 수 있습니다. 따라서 리소스에 대한 Amazon EC2 Auto Scaling 권한을 실수로 취소하는 것을 방지할 수 있습니다. 한 서비스 연결 역할을 여러 Auto Scaling 그룹에 사용하는 경우 서비스 연결 역할을 삭제하기 전에 해당 역할을 사용하는 모든 Auto Scaling 그룹을 삭제해야 합니다. 자세한 내용은 [Auto Scaling 인프라 삭제 \(p. 85\)](#) 단원을 참조하십시오.

IAM을 사용하여 서비스 연결 역할을 삭제할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 [서비스에 연결 역할 삭제](#) 단원을 참조하십시오.

AWSServiceRoleForAutoScaling 서비스 연결 역할을 삭제한 경우 Auto Scaling 그룹을 생성하고 다른 서비스 연결 역할을 지정하지 않으면 Amazon EC2 Auto Scaling에서 역할이 다시 생성됩니다.

Amazon EC2 Auto Scaling 서비스 연결 역할을 지원하는 리전

Amazon EC2 Auto Scaling에서는 서비스를 사용할 수 있는 모든 AWS 리전에서 서비스 연결 역할 사용을 지원합니다.

Amazon EC2 Auto Scaling 자격 증명 기반 정책 예제

기본적으로 IAM 사용자 및 역할은 Amazon EC2 Auto Scaling 리소스를 생성하거나 수정할 수 있는 권한이 없습니다. 또한 AWS Management 콘솔, AWS CLI 또는 AWS API를 사용해 작업을 수행할 수 없습니다. IAM 관리자는 지정된 리소스에서 특정 API 작업을 수행할 수 있는 권한을 사용자와 역할에게 부여하는 IAM 정책을 생성해야 합니다. 그런 다음 관리자는 해당 권한이 필요한 IAM 사용자 또는 그룹에 이러한 정책을 연결해야 합니다.

이러한 예제 JSON 정책 문서를 사용하여 IAM 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 [JSON 탭에서 정책 생성](#)을 참조하십시오.

주제

- [정책 모범 사례 \(p. 180\)](#)
- [미리 정의된 AWS 관리형 정책 \(p. 180\)](#)
- [고객 관리형 정책 예 \(p. 181\)](#)
- [서비스 연결 역할을 생성하는데 필요한 권한 \(p. 186\)](#)
- [API에 필요한 권한 \(p. 187\)](#)

다음은 권한 정책의 예입니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "autoscaling:CreateAutoScalingGroup",
        "autoscaling:UpdateAutoScalingGroup",
        "autoscaling>DeleteAutoScalingGroup"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": { "autoscaling:ResourceTag/environment": "test" }
      }
    }
  ],
}
```

```
{
  "Effect": "Allow",
  "Action": [
    "autoscaling:*LaunchConfiguration*",
    "autoscaling:Describe*"
  ],
  "Resource": "*"
}
```

이 샘플 정책은 Auto Scaling 그룹이 `environment=test` 태그를 사용하는 경우에만 그룹을 생성, 수정 및 삭제할 수 있는 권한을 사용자에게 제공합니다. 시작 구성은 태그를 지원하지 않고, `Describe` 작업은 리소스 수준 권한을 지원하지 않기 때문에 조건 없이 별도의 명령문에 지정해야 합니다. IAM 정책 명령문의 요소에 대한 자세한 내용은 [Amazon EC2 Auto Scaling 자격 증명 기반 정책 \(p. 172\)](#) 단원을 참조하십시오.

정책 모범 사례

자격 증명 기반 정책은 매우 강력합니다. 이 정책은 계정에서 사용자가 Amazon EC2 Auto Scaling 리소스를 생성, 액세스 또는 삭제할 수 있는지 여부를 결정합니다. 이 작업으로 인해 AWS 계정에 비용이 발생할 수 있습니다. 자격 증명 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따르십시오.

- AWS 관리형 정책을 사용하여 시작하기 – Amazon EC2 Auto Scaling 사용을 빠르게 시작하려면 AWS 관리형 정책을 사용하여 필요한 권한을 직원에게 부여합니다. 이 정책은 이미 계정에서 사용할 수 있으며 AWS에 의해 유지 관리 및 업데이트됩니다. 자세한 내용은 IAM 사용 설명서의 [AWS 관리형 정책으로 권한 사용 시작하기](#)를 참조하십시오.
- 최소 권한 부여 – 사용자 지정 정책을 생성할 때는 작업을 수행하는 데 필요한 권한만 부여합니다. 최소한의 권한 조합으로 시작하여 필요에 따라 추가 권한을 부여합니다. 처음부터 권한을 많이 부여한 후 나중에 줄이는 방법보다 이 방법이 안전합니다. 자세한 내용은 IAM 사용 설명서의 [최소 권한 부여](#)를 참조하십시오.
- 중요한 작업에 대해 MFA 활성화 – 보안을 강화하기 위해 IAM 사용자가 중요한 리소스 또는 API 작업에 액세스하려면 멀티 팩터 인증(MFA)을 사용해야 합니다. 자세한 내용은 IAM 사용 설명서의 [AWS에서 Multi-Factor Authentication\(MFA\) 사용하기](#)를 참조하십시오.
- 보안 강화를 위해 정책 조건 사용 – 실제로 가능한 경우, 자격 증명 기반 정책이 리소스에 대한 액세스를 허용하는 조건을 정의합니다. 예를 들어 요청을 할 수 있는 IP 주소의 범위를 지정하도록 조건을 작성할 수 있습니다. 지정된 날짜 또는 시간 범위 내에서만 요청을 허용하거나, SSL 또는 MFA를 사용해야 하는 조건을 작성할 수도 있습니다. 자세한 내용은 IAM 사용 설명서의 [IAM JSON 정책 요소: 조건](#)을 참조하십시오.

Note

일부 Auto Scaling API 작업의 경우 작업을 통해 생성하거나 수정할 수 있는 특정 Amazon EC2 Auto Scaling 그룹을 정책에 포함할 수 있습니다. 개별 Auto Scaling 그룹 ARN을 지정하여 이러한 작업에 대한 대상 리소스를 제한할 수 있습니다. 그러나 특정 태그가 있는 Auto Scaling 그룹에서 작업을 허용(또는 거부)하는 태그 기반 정책을 사용하는 것이 좋습니다.

미리 정의된 AWS 관리형 정책

AWS에 의해 생성된 관리형 정책은 일반 사용 사례에서 필요한 권한을 부여합니다. 필요한 액세스를 기반으로 IAM 사용자에게 이러한 정책을 연결할 수 있습니다. 각 정책은 Amazon EC2 Auto Scaling에 대한 API 작업 모두 또는 일부에 대한 액세스 권한을 부여합니다.

다음은 Amazon EC2 Auto Scaling에 대한 AWS 관리형 정책입니다.

- `AutoScalingConsoleFullAccess` - AWS Management 콘솔을 사용하여 Amazon EC2 Auto Scaling에 전체 액세스할 수 있는 권한을 부여합니다.
- `AutoScalingConsoleReadOnlyAccess` - AWS Management 콘솔을 사용하여 Amazon EC2 Auto Scaling에 읽기 전용으로 액세스할 수 있는 권한을 부여합니다.

- `AutoScalingFullAccess` - Amazon EC2 Auto Scaling에 대한 모든 액세스 권한을 부여합니다.
- `AutoScalingReadOnlyAccess` - Amazon EC2 Auto Scaling에 대한 읽기 전용 액세스 권한을 부여합니다.

또한 `AmazonEC2FullAccess` 정책을 사용하여 모든 Amazon EC2 리소스 및 관련 서비스(Amazon EC2 Auto Scaling, CloudWatch 및 Elastic Load Balancing 포함)에 전체 액세스 권한을 부여할 수 있습니다.

고객 관리형 정책 예

IAM 사용자 또는 그룹이 Amazon EC2 Auto Scaling 작업을 수행할 수 있는 권한을 허용하거나 거부하는 고유한 사용자 지정 IAM 정책을 만들 수 있습니다. 지정된 권한이 필요한 IAM 사용자 또는 그룹에 이러한 사용자 지정 정책을 연결할 수 있습니다. 다음 예제에서는 몇 가지 일반적인 사용 사례의 권한을 보여 줍니다.

정책을 처음 생성하는 경우 먼저 계정에서 IAM 사용자를 생성하고 정책을 사용자에게 연결하는 것이 좋습니다. 콘솔을 사용하여 정책을 사용자에게 연결할 때 각 정책의 효과를 확인할 수 있습니다.

Auto Scaling 그룹을 생성하고 업데이트할 때 일부 작업에서는 다른 특정 작업을 수행해야 합니다. IAM 정책 명령문의 `Action` 요소에서 다른 작업을 지정할 수 있습니다. 예를 들어 Elastic Load Balancing, CloudWatch 및 Amazon SNS에 대한 추가 API 작업이 있으며 이는 사용자에게 제공하려는 액세스 권한에 따라 필요할 수 있습니다.

주제

- 사용할 수 있는 태그 키 및 태그 값 제어 (p. 181)
- 태그를 기반으로 Auto Scaling 리소스에 대한 액세스 제어 (p. 182)
- Auto Scaling 그룹의 용량 제한 제어 (p. 183)
- 전달할 수 있는 IAM 역할 제어 (PassRole 사용) (p. 184)
- 사용자가 Auto Scaling 그룹 용량을 변경하도록 허용 (p. 184)
- 사용자가 시작 구성을 생성하고 사용하도록 허용 (p. 185)
- 사용자가 시작 템플릿을 생성하고 사용하도록 허용 (p. 186)

사용할 수 있는 태그 키 및 태그 값 제어

IAM 정책의 조건을 사용하여 Auto Scaling 그룹에 적용할 수 있는 태그 키와 태그 값을 제어할 수 있습니다.

사용자에게 특정 태그를 지정할 경우에만 Auto Scaling 그룹에 태그를 지정하거나 생성할 수 있는 권한을 부여하려면 `aws:RequestTag` 조건 키를 사용하십시오. 특정 태그 키만 허용하려면 `aws:TagKeys` 조건 키와 `ForAllValues` 변경자를 사용합니다.

다음 정책은 사용자에게 요청에 `environment` 키를 포함하는 태그를 지정하도록 요구합니다. `"?*"` 값은 태그 키에 대한 일부 값이 있음을 나타냅니다. 와일드카드를 사용하려면 `StringLike` 조건 연산자를 사용해야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "autoscaling:CreateAutoScalingGroup",
      "autoscaling:CreateOrUpdateTags"
    ],
    "Resource": "*",
    "Condition": {
      "StringLike": { "aws:RequestTag/environment": "?*" }
    }
  }]
}
```



```
}

```

다음 정책은 사용자가 `purpose=webserver` 및 `cost-center=cc123` 태그가 있는 Auto Scaling 그룹에만 태그를 지정할 수 있도록 지정하고 `purpose` 및 `cost-center` 태그만 허용합니다(다른 태그는 지정할 수 없음).

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "autoscaling:CreateAutoScalingGroup",
      "autoscaling:CreateOrUpdateTags"
    ],
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:RequestTag/purpose": "webserver",
        "aws:RequestTag/cost-center": "cc123"
      },
      "ForAllValues:StringEquals": { "aws:TagKeys": ["purpose", "cost-center"] }
    }
  }]
}
```

다음 정책은 사용자에게 요청에 태그를 한 개 이상 지정하도록 요구하고, `cost-center` 키와 `owner` 키만 허용합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "autoscaling:CreateAutoScalingGroup",
      "autoscaling:CreateOrUpdateTags"
    ],
    "Resource": "*",
    "Condition": {
      "ForAnyValue:StringEquals": { "aws:TagKeys": ["cost-center", "owner"] }
    }
  }]
}
```

Note

조건의 경우 조건 키는 대소문자를 구분하지 않고 조건 값은 대소문자를 구분합니다. 따라서 태그 키의 대소문자 구별을 설정하려면 태그 키가 조건의 값으로 지정된 `aws:TagKeys` 조건 키를 사용합니다.

태그를 기반으로 Auto Scaling 리소스에 대한 액세스 제어

또한 `autoscaling:ResourceTag` 조건 키를 사용하여 Auto Scaling 그룹에 연결된 태그를 기반으로 액세스를 제어하도록 IAM 정책에 태그 정보를 제공할 수 있습니다.

Auto Scaling 그룹을 생성 및 관리하기 위한 액세스와 조정 정책 제어

다음 정책은 이름에 `Scaling`이라는 문자열이 있는 모든 Amazon EC2 Auto Scaling 작업을 사용할 수 있는 권한을 사용자에게 부여합니다. 단, Auto Scaling 그룹에 `purpose=webserver` 태그가 있어야 합니다. `Describe` 작업은 리소스 수준 권한을 지원하지 않기 때문에 조건 없이 별도의 명령문에 지정해야 합니다.

```
{

```

```
"Version": "2012-10-17",
"Statement": [
{
  "Effect": "Allow",
  "Action": ["autoscaling:*Scaling*"],
  "Resource": "*",
  "Condition": {
    "StringEquals": { "autoscaling:ResourceTag/purpose": "webserver" }
  }
},
{
  "Effect": "Allow",
  "Action": "autoscaling:Describe*Scaling*",
  "Resource": "*"
}]
}
```

삭제할 수 있는 조정 정책 제어

다음 정책은 사용자가 `autoscaling:DeletePolicy` 작업을 사용하여 조정 정책을 삭제할 수 있도록 허용합니다. 그러나 작업 대상 Auto Scaling 그룹에 `environment=production` 태그가 있는 경우 작업을 거부합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "autoscaling:DeletePolicy",
      "Resource": "*"
    },
    {
      "Effect": "Deny",
      "Action": "autoscaling:DeletePolicy",
      "Resource": "*",
      "Condition": {
        "StringEquals": { "autoscaling:ResourceTag/environment": "production" }
      }
    }
  ]
}
```

Auto Scaling 그룹의 용량 제한 제어

Amazon EC2 Auto Scaling를 사용하면 생성할 수 있는 Auto Scaling 그룹의 크기를 제한할 수 있습니다. 다음 정책은 사용자에게 `allowed=true` 태그가 있는 모든 Auto Scaling 그룹을 생성하고 업데이트할 수 있는 권한을 부여합니다. 단, 해당 작업이 최소 크기를 1 미만으로 지정하지 않아야 하거나, 최대 크기를 10보다 크게 지정하지 않아야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "autoscaling:CreateAutoScalingGroup",
        "autoscaling:UpdateAutoScalingGroup"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": { "autoscaling:ResourceTag/allowed": "true" },
        "NumericGreaterThanEqualsIfExists": { "autoscaling:MinSize": 1 },
        "NumericLessThanEqualsIfExists": { "autoscaling:MaxSize": 10 }
      }
    }
  ]
}
```

```
}
  }]
}
```

전달할 수 있는 IAM 역할 제어 (PassRole 사용)

사용자가 인스턴스 프로파일(IAM 역할용 컨테이너)을 지정하는 시작 템플릿 또는 시작 구성을 사용하여 Auto Scaling 그룹을 생성할 수 있도록 하려면 다음 예제와 같이 사용자가 역할을 전달할 수 있도록 허용하는 명령문이 포함된 정책을 사용해야 합니다. ARN을 지정함으로써 정책은 사용자에게 이름이 `gateam-`으로 시작하는 역할만 전달할 권한을 부여합니다. 자세한 내용은 [Amazon EC2 인스턴스에서 실행되는 애플리케이션의 IAM 역할](#) (p. 192) 단원을 참조하십시오.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/gateam-*",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": [
            "ec2.amazonaws.com"
          ]
        }
      }
    }
  ]
}
```

사용자가 Auto Scaling 그룹 용량을 변경하도록 허용

다음 정책은 사용자에게 `SetDesiredCapacity` 및 `TerminateInstanceInAutoScalingGroup` API 작업을 사용할 수 있는 권한을 부여합니다. Resource 요소는 와일드카드(*)를 사용하여 사용자가 Auto Scaling 그룹의 용량을 변경할 수 있음을 나타냅니다.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "autoscaling:SetDesiredCapacity",
      "autoscaling:TerminateInstanceInAutoScalingGroup"
    ],
    "Resource": "*"
  }]
}
```

태그를 사용하여 Auto Scaling 그룹에 대한 액세스를 제어하지 않는 경우, 앞의 명령문을 조정하여 사용자에게 이름이 `devteam-`으로 시작하는 Auto Scaling 그룹의 용량만 변경할 수 있는 권한을 부여할 수 있습니다. ARN 값을 지정하는 방법에 대한 자세한 내용은 [리소스](#) (p. 173) 단원을 참조하십시오.

```
"Resource":
"arn:aws:autoscaling:region:123456789012:autoScalingGroup:*:autoScalingGroupName/devteam-
*"
```

여러 ARN을 목록에 포함시켜 지정할 수도 있습니다. UUID를 포함하면 특정 Auto Scaling 그룹에 대한 액세스 권한을 부여할 수 있습니다. 새 그룹의 UUID는 동일한 이름을 가진 삭제된 그룹의 UUID와 다릅니다.

```
"Resource": [  
  "arn:aws:autoscaling:region:123456789012:autoScalingGroup:7fe02b8e-7442-4c9e-8c8e-85fa99e9b5d9:autoScalingGroup/devteam-1",  
  "arn:aws:autoscaling:region:123456789012:autoScalingGroup:9d8e8ea4-22e1-44c7-a14d-520f8518c2b9:autoScalingGroupName/devteam-2",  
  "arn:aws:autoscaling:region:123456789012:autoScalingGroup:60d6b363-ae8b-467c-947f-f1d308935521:autoScalingGroupName/devteam-3"  
]
```

사용자가 시작 구성을 생성하고 사용하도록 허용

다음 정책은 사용자에게 인스턴스 유형이 t2.micro이고 시작 구성의 이름이 gateam-으로 시작할 경우 시작 구성을 생성할 수 있는 권한을 부여합니다. ARN 값을 지정하는 방법에 대한 자세한 내용은 [리소스 \(p. 173\)](#) 단원을 참조하십시오. 이름이 gateam-으로 시작하는 경우에만 Auto Scaling 그룹의 시작 구성을 지정할 수 있습니다.

이 명령문의 마지막 부분에서는 사용자에게 시작 구성을 설명하고 AWS 계정의 특정 Amazon EC2 리소스에 액세스할 수 있는 권한을 부여합니다. 이렇게 하면 사용자에게 Amazon EC2 Auto Scaling 콘솔에서 시작 구성을 생성하고 관리할 수 있는 최소 권한이 부여됩니다.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": "autoscaling:CreateLaunchConfiguration",  
      "Resource": "arn:aws:autoscaling:region:123456789012:launchConfiguration:*:launchConfigurationName/gateam-*",  
      "Condition": {  
        "StringEquals": { "autoscaling:InstanceType": "t2.micro" }  
      }  
    },  
    {  
      "Effect": "Allow",  
      "Action": [  
        "autoscaling:CreateAutoScalingGroup",  
        "autoscaling:UpdateAutoScalingGroup"  
      ],  
      "Resource": "*",  
      "Condition": {  
        "StringLikeIfExists": { "autoscaling:LaunchConfigurationName": "gateam-*" }  
      }  
    },  
    {  
      "Effect": "Allow",  
      "Action": [  
        "autoscaling:DescribeLaunchConfigurations",  
        "ec2:DescribeImages",  
        "ec2:DescribeVolumes",  
        "ec2:DescribeInstances",  
        "ec2:DescribeInstanceAttribute",  
        "ec2:DescribeKeyPairs",  
        "ec2:DescribeSecurityGroups",  
        "ec2:DescribeSpotInstanceRequests",  
        "ec2:DescribeSpotPriceHistory",  
        "ec2:DescribeVpcClassicLink",  
        "ec2:DescribeVpcs",  
        "ec2:DescribeSubnets"  
      ],  
      "Resource": "*"   
    }  
  ]  
}
```

```
}
```

이 정책에 API 작업을 추가하여 다음과 같이 사용자에게 더 많은 옵션을 제공할 수 있습니다.

- `iam:ListInstanceProfiles`: 인스턴스 프로파일을 표시합니다.
- `ec2:CreateSecurityGroup`: 새 보안 그룹을 생성합니다.
- `ec2:AuthorizeSecurityGroupIngress`: 인바운드 규칙을 추가합니다.
- `ec2:CreateKeyPair`: 새 키 페어를 생성합니다.

사용자가 시작 템플릿을 생성하고 사용하도록 허용

예제 정책은 [시작 템플릿 지원](#) (p. 188) 단원을 참조하십시오.

서비스 연결 역할을 생성하는데 필요한 권한

Amazon EC2 Auto Scaling에는 AWS 계정의 사용자가 Amazon EC2 Auto Scaling API 작업을 처음 호출할 때 서비스 연결 역할을 생성할 수 있는 권한이 필요합니다. 서비스 연결 역할이 아직 존재하지 않으면 Amazon EC2 Auto Scaling에서 해당 역할을 계정에 생성합니다. 서비스 연결 역할은 Amazon EC2 Auto Scaling에게 사용자를 대신하여 다른 서비스를 호출할 수 있도록 권한을 부여합니다.

역할 자동 생성이 성공하려면 사용자가 `iam:CreateServiceLinkedRole` 작업에 대한 권한을 보유해야 합니다.

```
"Action": "iam:CreateServiceLinkedRole"
```

다음은 사용자가 Amazon EC2 Auto Scaling에 대한 Amazon EC2 Auto Scaling 서비스 연결 역할을 생성할 수 있도록 허용하는 권한 정책의 예입니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:CreateServiceLinkedRole",
      "Resource": "arn:aws:iam::*:role/aws-service-role/autoscaling.amazonaws.com/  
AWSServiceRoleForAutoScaling",
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "autoscaling.amazonaws.com"
        }
      }
    }
  ]
}
```

전달할 수 있는 서비스 연결 역할 제한(PassRole 사용)

사용자가 사용자 지정 접미사 서비스 연결 역할을 Auto Scaling 그룹에 전달해야 하는 경우 필요한 액세스 권한에 따라 사용자 또는 역할에 정책을 연결해야 합니다. 이 정책은 사용자가 액세스해야 하는 서비스 연결 역할로만 제한하는 것이 좋습니다. 사용자 지정 접미사 서비스 연결 역할에 대한 자세한 내용은 [Amazon EC2 Auto Scaling에 대한 서비스 연결 역할](#) (p. 176) 단원을 참조하십시오.

다음 예는 다른 서비스 연결 역할에 다른 키에 대한 액세스 권한을 부여하는 경우 고객 관리 CMK의 보안을 용이하게 하는 데 유용합니다. 필요에 따라 개발 팀을 위한 CMK, 품질 보증 팀을 위한 CMK 및 재무 팀을 위한 CMK가 있을 수 있습니다. 먼저 필수 CMK에 액세스할 수 있는 서비스 연결 역할을 생성합니다(예: 서비스 연결 역할 `AWSServiceRoleForAutoScaling_devteamkeyaccess`). 그런 다음 서비스 링크된 역할을 Auto Scaling 그룹에 전달할 권한을 부여하려면 표시된 대로 정책을 IAM 사용자에게 연결하십시오.

다음 예제의 정책은 사용자에게 이름이 devteam-으로 시작하는 Auto Scaling 그룹을 생성하기 위해 **AWSServiceRoleForAutoScaling_devteamkeyaccess** 역할을 전달할 수 있는 권한을 부여합니다. 다른 서비스와 연결된 역할을 지정하려고 하면 오류가 발생합니다. 서비스 연결 역할을 지정하지 않으려는 경우 기본 AWSServiceRoleForAutoScaling 역할이 대신 사용됩니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/aws-service-role/autoscaling.amazonaws.com/AWSServiceRoleForAutoScaling_devteamkeyaccess",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": [
            "autoscaling.amazonaws.com"
          ]
        },
        "StringLike": {
          "iam:AssociatedResourceARN": [
            "arn:aws:autoscaling:region:123456789012:autoScalingGroup:*:autoScalingGroupName/devteam-*"
          ]
        }
      }
    }
  ]
}
```

API에 필요한 권한

Amazon EC2 Auto Scaling API에서 다음 작업을 호출할 때 사용자는 Amazon EC2 및 IAM에서 특정 작업을 수행할 수 있는 권한이 있어야 합니다. IAM 정책 명령문의 Action 요소에서 다음 작업을 지정합니다.

시작 구성을 사용하여 Auto Scaling 그룹 만들기

- autoscaling:CreateAutoScalingGroup
- iam:CreateServiceLinkedRole(기본 서비스 연결 역할을 사용하고 해당 역할이 아직 존재하지 않는 경우 필요함)

시작 템플릿을 사용하여 Auto Scaling 그룹 만들기

- autoscaling:CreateAutoScalingGroup
- ec2:RunInstances
- iam:CreateServiceLinkedRole(기본 서비스 연결 역할을 사용하고 해당 역할이 아직 존재하지 않는 경우 필요함)

시작 템플릿을 사용하는 Auto Scaling 그룹 업데이트

- autoscaling:UpdateAutoScalingGroup
- ec2:RunInstances

시작 구성 생성

- autoscaling:CreateLaunchConfiguration

- `ec2:DescribeImages`
- `ec2:DescribeInstances`
- `ec2:DescribeInstanceAttribute`
- `ec2:DescribeKeyPairs`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSpotInstanceRequests`
- `ec2:DescribeVpcClassicLink`

시작 템플릿 지원

Amazon EC2 Auto Scaling에서는 Auto Scaling 그룹에 Amazon EC2 시작 템플릿 사용을 지원합니다. 사용자가 시작 템플릿에서 Auto Scaling 그룹을 생성하도록 허용하는 것이 좋습니다. 이렇게 하면 EC2의 최신 기능을 사용할 수 있기 때문입니다. 또한 사용자는 [혼합 인스턴스 정책](#)을 사용하려면 시작 템플릿을 지정해야 합니다.

`AmazonEC2FullAccess` 정책을 사용하여 사용자에게 AWS 계정의 Amazon EC2 Auto Scaling 리소스, 시작 템플릿 및 기타 EC2 리소스로 작업할 수 있는 완전한 액세스 권한을 부여할 수 있습니다. 또는 고유한 사용자 지정 IAM 정책을 생성하여 사용자에게 특정 API 작업을 수행할 수 있는 세분화된 권한을 부여할 수 있습니다.

다른 정책에서 이미 권한을 부여한 경우가 아니라면 다음 예제 정책을 사용하여 사용자에게 시작 템플릿으로 작업할 수 있는 부여합니다. 시작 구성(Amazon EC2 Auto Scaling의 이전 기능)에서만 Auto Scaling 그룹을 생성하는 사용자에게는 이러한 권한을 부여할 필요가 없습니다.

Amazon EC2의 IAM 정책에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [IAM 정책](#)을 참조하십시오.

주제

- 예: 리소스 수준 권한을 사용하여 액세스 제어 (p. 186)
- 예: 사용자가 시작 템플릿 및 시작 템플릿 버전을 생성하고 관리하도록 허용 (p. 190)
- 예: 인스턴스 메타데이터 서비스 버전 2(IMDSv2) 사용 필요 (p. 191)
- 필요한 추가 권한 (p. 191)

시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하거나 업데이트하는 IAM 사용자 또는 역할은 `ec2:RunInstances` 작업을 사용할 권한이 있어야 합니다. 이 권한이 없는 사용자에게는 시작 템플릿을 사용할 권한이 없다는 오류가 발생합니다.

`ec2:RunInstances`에 대한 사용자 권한은 시작 템플릿을 사용하여 Auto Scaling 그룹을 생성하거나 업데이트할 때만 확인됩니다. `Latest` 또는 `Default` 시작 템플릿을 사용하도록 구성된 Auto Scaling 그룹의 경우 새 버전의 시작 템플릿을 생성할 때 권한이 확인되지 않습니다. 권한을 확인하려면 사용자는 특정 버전의 시작 템플릿을 사용하도록 Auto Scaling 그룹을 구성해야 합니다.

예: 리소스 수준 권한을 사용하여 액세스 제어

사용자 권한을 부여할 때 Amazon EC2는 `ec2:RunInstances` 작업 호출의 일부로 생성되는 리소스에 대해 리소스 수준 권한을 지정하여 사용자가 작업할 수 있는 리소스를 제어할 수 있습니다. 이는 권장되는 방법입니다.

다음 예제에서는 리소스 수준 권한을 사용하여 특정 시작 템플릿에 대한 액세스를 제한합니다. 또한 사용자가 `ec2:RunInstances` 권한을 허용할 때 시작할 수 있는 인스턴스 구성을 제어할 수 있는 몇 가지 가능한 방법을 보여줍니다.

이 예에는 4개의 명령문이 있습니다.

- 첫 번째 명령문은 지정된 리전에 위치하고 environment=test 태그가 있는 템플릿을 시작할 수 있는 사용자 액세스를 제한합니다.
- 두 번째 명령문은 사용자가 생성 시 인스턴스와 볼륨에 태그를 지정하도록 허용합니다. 이 부분은 시작 템플릿에 지정된 태그가 있는 경우에 필요합니다.
- 세 번째 명령문에서는 사용자가 인스턴스를 특정 서브넷(subnet-1a2b3c4d)으로 시작하여 특정 보안 그룹(sg-1a2b3c4d)을 사용하고 특정 AMI(ami-1a2b3c4d)를 사용해야 합니다. 또한 사용자에게 인스턴스를 시작하는 데 필요한 추가 리소스(네트워크 인터페이스 및 볼륨)에 대한 액세스 권한을 부여합니다.
- 네 번째 명령문은 사용자가 특정 인스턴스 유형(t2.micro)의 인스턴스만 시작하도록 허용합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "arn:aws:ec2:region:123456789012:launch-template/*",
      "Condition": {
        "StringEquals": { "ec2:ResourceTag/environment": "test" }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:CreateTags",
      "Resource": "arn:aws:ec2:region:123456789012:*/*",
      "Condition": {
        "StringEquals": { "ec2:CreateAction": "RunInstances" }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:region:123456789012:subnet/subnet-1a2b3c4d",
        "arn:aws:ec2:region:123456789012:security-group/sg-1a2b3c4d",
        "arn:aws:ec2:region:123456789012:network-interface/*",
        "arn:aws:ec2:region:123456789012:volume/*",
        "arn:aws:ec2:region::image/ami-1a2b3c4d"
      ]
    },
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "arn:aws:ec2:region:123456789012:instance/*",
      "Condition": {
        "StringEquals": { "ec2:InstanceType": "t2.micro" }
      }
    }
  ]
}
```

또한 Auto Scaling 그룹을 생성하고 업데이트할 때 시작 구성을 사용할 수 없도록 사용자의 액세스를 제한하는 IAM 정책을 생성할 수 있습니다. 선택적으로 새 버전을 사용하도록 Auto Scaling 그룹이 업데이트될 때마다 인스턴스를 시작할 때 완료할 작업에 대한 IAM 권한이 확인되도록 하기 위해 사용할 특정 버전을 설정하도록 요구할 수도 있습니다.

다음 명령문은 사용자에게 시작 템플릿을 지정하는 경우 Auto Scaling 그룹을 생성하거나 업데이트할 수 있는 권한을 부여합니다. Latest 또는 Default 시작 템플릿 버전을 지정하기 위해 버전 번호를 생략하거나 시작 구성을 대신 지정하면 작업이 실패합니다.

```
{
```



```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "autoscaling:CreateAutoScalingGroup",
      "autoscaling:UpdateAutoScalingGroup"
    ],
    "Resource": "*",
    "Condition": {
      "Bool": { "autoscaling:LaunchTemplateVersionSpecified": "true" }
    }
  },
  {
    "Effect": "Deny",
    "Action": [
      "autoscaling:CreateAutoScalingGroup",
      "autoscaling:UpdateAutoScalingGroup"
    ],
    "Resource": "*",
    "Condition": {
      "Null": { "autoscaling:LaunchConfigurationName": "false" }
    }
  }
]
```

또는 사용자가 특정 버전의 시작 템플릿이 아닌 Latest 또는 Default 버전의 시작 템플릿을 지정하도록 요구할 수도 있습니다. 이렇게 하려면 다음 예제와 같이 `autoscaling:LaunchTemplateVersionSpecified`의 값을 `false`로 변경합니다.

```
"Condition": {
  "Bool": { "autoscaling:LaunchTemplateVersionSpecified": "false" }
}
```

예: 사용자가 시작 템플릿 및 시작 템플릿 버전을 생성하고 관리하도록 허용

사용자에게 시작 템플릿과 시작 템플릿 버전을 생성, 수정, 설명 및 삭제할 수 있는 권한을 부여하는 정책을 생성할 수 있습니다. 사용자에게 이러한 권한을 추가하기 전에 Linux 인스턴스용 Amazon EC2 사용 설명서의 [시작 템플릿 사용 제어](#) 섹션을 검토하십시오. 추가 정책 예제는 Linux 인스턴스용 Amazon EC2 사용 설명서의 [예: 시작 템플릿 작업](#)을 참조하십시오.

Important

Latest 또는 Default 시작 템플릿 버전을 사용하도록 구성된 그룹의 경우 인스턴스를 시작할 때 완료할 작업에 대한 권한은 새 버전의 시작 템플릿을 생성할 때 Amazon EC2 Auto Scaling에 의해 확인되지 않습니다. 이는 시작 템플릿 버전을 생성하고 관리할 수 있는 권한을 설정할 때 중요한 고려 사항입니다.

다음 정책은 사용자에게 시작 템플릿을 생성할 수 있는 권한을 부여합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:CreateLaunchTemplate",
      "Resource": "*"
    }
  ]
}
```

```
}  
}
```

예: 인스턴스 메타데이터 서비스 버전 2(IMDSv2) 사용 필요

Important

모든 새 인스턴스에서 IMDSv2를 사용해야 하는 경우 Auto Scaling 그룹에서 시작 템플릿을 사용해야 합니다.

보안을 강화하기 위해 IMDSv2가 필요한 시작 템플릿 사용을 요구하도록 사용자의 권한을 설정할 수 있습니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [인스턴스 메타데이터 서비스 구성](#)을 참조하십시오.

다음 정책에서는 IMDSv2("ec2:MetadataHttpTokens":"required"로 표시) 사용을 요구하도록 인스턴스도 옵트인되지 않으면 사용자가 ec2:RunInstances 작업을 호출할 수 없도록 지정합니다. 인스턴스가 IMDSv2를 요구하도록 지정하지 않으면 UnauthorizedOperation 작업을 호출할 때 ec2:RunInstances 오류가 발생합니다.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "RequireImdsV2",  
      "Effect": "Deny",  
      "Action": "ec2:RunInstances",  
      "Resource": "arn:aws:ec2:*:*:instance/*",  
      "Condition": {  
        "StringNotEquals": { "ec2:MetadataHttpTokens": "required" }  
      }  
    }  
  ]  
}
```

기존 Auto Scaling 그룹을 업데이트하려면 인스턴스 메타데이터 옵션이 구성된 새 시작 템플릿을 사용하거나 인스턴스 메타데이터 옵션이 구성된 현재 시작 템플릿의 새 버전을 사용해야 합니다.

Tip

새 시작 템플릿을 사용하는 대체 인스턴스를 강제로 시작하려면 그룹의 기존 인스턴스를 종료하면 됩니다. Amazon EC2 Auto Scaling은 즉시 새 인스턴스를 시작하여 종료한 인스턴스를 대체합니다. 또는 조정 정책을 사용하는 경우 그룹의 원하는 용량을 늘려 새 인스턴스를 시작할 수 있습니다. 축소를 위한 정책 조건이 충족되면 Auto Scaling 그룹은 그룹의 종료 정책에 따라 이전 인스턴스를 점진적으로 종료합니다.

필요한 추가 권한

지원하려는 시나리오에 따라 IAM 정책 명령문의 Action 요소에 이러한 추가 작업을 지정할 수 있습니다.

콘솔에서 Auto Scaling 그룹을 생성하고 업데이트하려면 사용자는 Amazon EC2에서 다음 권한도 가지고 있어야 합니다.

- ec2:DescribeLaunchTemplates
- ec2:DescribeLaunchTemplateVersions
- ec2:DescribeVpcs
- ec2:DescribeSubnets
- ec2:DescribeAvailabilityZones

이러한 추가적인 최소 권한이 없으면 시작 템플릿과 네트워크 옵션을 Auto Scaling 그룹 마법사에 로드할 수 없으며 사용자는 시작 템플릿을 사용하여 인스턴스를 시작하기 위해 마법사를 단계별로 실행할 수 없습니다.

정책에 더 많은 작업을 추가하여 사용자에게 마법사에서 더 많은 옵션을 제공할 수 있습니다. 예를 들어 Elastic Load Balancing API 작업에 대한 권한을 추가하여 사용자가 마법사의 3단계에서 기존 로드 밸런서 목록에서 선택하도록 허용할 수 있습니다.

시작 템플릿이 IAM 역할의 인스턴스 프로파일을 지정하는 경우 `iam:PassRole`을 사용하여 사용자가 역할을 Amazon EC2에 전달하도록 허용(또는 거부)합니다. 이 권한이 없는 사용자에게는 시작 템플릿을 사용할 권한이 없다는 오류가 발생합니다. 정책에 대한 예는 [전달할 수 있는 IAM 역할 제어 \(PassRole 사용\)](#) (p. 184)을 참조하십시오.

Amazon EC2 콘솔에서 프로비저닝을 확인하려면 사용자에게 추가 권한(예: 인스턴스를 볼 수 있는 `ec2:DescribeInstances` 권한, 인스턴스 상태를 표시할 수 있는 `ec2:DescribeInstanceStatus` 권한 또는 태그를 표시할 수 있는 `ec2:DescribeTags` 권한)이 필요할 수 있습니다.

Amazon EC2 인스턴스에서 실행되는 애플리케이션의 IAM 역할

Amazon EC2 인스턴스에서 실행되는 애플리케이션이 다른 AWS 서비스에 액세스하려면 자격 증명이 있어야 합니다. 이러한 자격 증명을 안전하게 제공하려면 IAM 역할을 사용하십시오. 이 역할은 애플리케이션이 다른 AWS 리소스에 액세스할 때 사용할 수 있는 임시 권한을 제공합니다. 역할의 권한에 따라 애플리케이션에서 수행할 수 있는 작업이 결정됩니다.

이 인스턴스에서 실행되는 애플리케이션은 인스턴스 프로파일 메타데이터를 통해 역할의 임시 자격 증명에 액세스할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 [IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여](#)를 참조하십시오.

Auto Scaling 그룹 인스턴스의 경우, 시작 템플릿 또는 시작 구성을 생성하고 인스턴스와 연결할 인스턴스 프로파일을 선택해야 합니다. 인스턴스 프로파일은 인스턴스가 시작될 때 인스턴스에 Amazon EC2가 IAM 역할을 전달하도록 허용하는 IAM 역할을 위한 컨테이너입니다. 먼저 AWS 리소스에 액세스하는 데 필요한 권한을 모두 가진 IAM 역할을 생성합니다. 그런 다음 인스턴스 프로파일을 생성하고 여기에 그 역할을 할당합니다. 자세한 내용은 IAM 사용 설명서의 [인스턴스 프로파일 사용](#)을 참조하십시오.

Note

IAM 콘솔에서 Amazon EC2 역할을 생성하는 경우, 콘솔이 역할 생성 단계를 안내하고 IAM 역할과 동일한 이름의 인스턴스 프로파일을 자동으로 만들어 줍니다.

자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2에 대한 IAM 역할](#)을 참조하십시오.

사전 조건

Amazon EC2에서 실행 중인 애플리케이션이 수임할 수 있는 IAM 역할을 만듭니다. 나중에 역할을 부여 받은 애플리케이션이 필요한 API 호출을 할 수 있도록 적절한 권한을 선택하십시오.

Important

애플리케이션이 다른 AWS 서비스에 대해 필요한 최소 권한만 가지도록 역할을 생성하는 것이 가장 좋습니다.

IAM 역할)를 생성하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 [Roles], [Create role]을 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 EC2와 EC2 사용 사례를 선택합니다. 다음: 권한을 선택합니다.
5. Attach permissions policy(권한 정책 연결)에서 필요한 권한을 포함한 AWS 관리형 정책을 선택합니다. Next: Tags(다음: 태그)를 선택한 후 Next: Review(다음: 검토)를 선택합니다.
6. 검토 페이지에서 역할의 이름을 입력하고 역할 생성을 선택합니다.

인스턴스 프로파일을 사용할 때는 인스턴스 프로파일을 지정하는 시작 템플릿 또는 시작 구성을 사용하여 Auto Scaling 그룹을 생성하는 IAM 사용자에게 `iam:PassRole` 권한이 필요합니다. 정책에 대한 예는 [전달할 수 있는 IAM 역할 제어 \(PassRole 사용\)](#) (p. 184)을 참조하십시오.

시작 구성 생성

AWS Management 콘솔을 사용하여 시작 구성을 생성하는 경우 세부 정보 구성 페이지의 IAM 역할에서 역할을 선택합니다. 자세한 내용은 [시작 구성 만들기](#) (p. 34) 단원을 참조하십시오.

AWS CLI에서 `create-launch-configuration` 명령을 사용하여 시작 구성을 생성할 경우 다음 예제와 같이 인스턴스 프로파일 이름을 지정합니다.

```
aws autoscaling create-launch-configuration --launch-configuration-name my-lc-with-  
instance-profile \  
--image-id ami-01e24be29428c15b2 --instance-type t2.micro \  
--iam-instance-profile my-instance-profile
```

시작 템플릿 생성

AWS Management 콘솔을 사용하여 시작 템플릿을 생성하는 경우, 고급 세부 정보 섹션의 IAM 인스턴스 프로파일에서 역할을 선택합니다. 자세한 내용은 [Auto Scaling 그룹을 위한 시작 템플릿 만들기](#) (p. 26) 단원을 참조하십시오.

AWS CLI에서 `create-launch-template` 명령을 사용하여 시작 템플릿을 생성할 경우 다음 예제와 같이 인스턴스 프로파일 이름을 지정합니다.

```
aws ec2 create-launch-template --launch-template-name my-lt-with-instance-profile --  
version-description version1 \  
--launch-template-data  
'{"ImageId":"ami-01e24be29428c15b2","InstanceType":"t2.micro","IamInstanceProfile":  
{ "Name": "my-instance-profile" } }'
```

암호화된 볼륨에 사용할 경우 필요한 CMK 키 정책

암호화된 Amazon EBS 스냅샷 또는 암호화된 볼륨을 지정하는 시작 템플릿을 생성하거나 기본적으로 암호화를 활성화하는 경우 다음 AWS Key Management Service 고객 마스터 키(CMK) 중 하나를 선택하여 데이터를 암호화할 수 있습니다.

- **AWS 관리형 CMK** — Amazon EBS에서 생성, 소유 및 관리하는 계정의 암호화 키입니다. 이 키는 새 계정을 위한 기본 암호화 키입니다. 고객 관리형 CMK를 지정하지 않는 한 AWS 관리형 CMK가 암호화에 사용됩니다.
- **고객 관리형 CMK** — 생성, 소유 및 유지 관리하는 사용자 지정 암호화 키입니다. 자세한 정보는 AWS Key Management Service Developer Guide의 [키 생성](#) 단원을 참조하십시오.

참고: Amazon EBS는 비대칭 CMK를 지원하지 않습니다.

Amazon EC2 Auto Scaling은 기본 AWS 관리형 CMK를 사용하여 AWS 계정의 암호화된 볼륨을 보호하는 데 추가 승인이 필요하지 않습니다.

Amazon EBS 암호화에 대해 고객 관리형 CMK를 지정하는 경우 Amazon EC2 Auto Scaling이 사용자를 대신해 인스턴스를 시작할 수 있도록 적절한 [서비스 연결 역할](#) (p. 176)에 CMK에 대한 액세스 권한을 부여해야 합니다. 이렇게 하려면 CMK를 생성할 때 나중에 CMK의 키 정책을 수정해야 합니다.

키 정책 구성

이 페이지의 예를 사용하여 고객 관리형 CMK에 대한 Amazon EC2 Auto Scaling 액세스 권한을 부여하는 키 정책을 구성하십시오. Amazon EC2 Auto Scaling에서 사용하려면 CMK의 키 정책에 최소한 두 개의 정책 명령문을 추가해야 합니다.

- 첫 번째 명령문을 사용하면 Principal 요소에 지정된 IAM 자격 증명에서 CMK를 직접 사용하도록 할 수 있습니다. 여기에는 CMK에서 AWS KMS Encrypt, Decrypt, ReEncrypt*, GenerateDataKey* 및 DescribeKey 작업을 수행할 수 있는 권한이 포함되어 있습니다.
- 두 번째 명령문은 Principal 요소에 지정된 IAM 자격 증명에서 권한 부여를 통해 AWS KMS 또는 다른 보안 주체와 통합된 AWS 서비스에 자체 권한의 하위 집합을 위임할 수 있습니다. 이를 통해 사용자는 CMK를 사용하여 사용자 대신 암호화된 리소스를 생성할 수 있습니다.

CMK 정책에 새 정책 명령문을 추가할 경우 정책의 기존 섹션을 변경하지 마십시오.

다음 각 예에서 키 ID 또는 서비스 연결 역할의 이름과 같이 바뀌어야 하는 인수는 ##### ## ### ## #로 표시됩니다. 대부분의 경우 서비스 연결 역할의 이름을 Amazon EC2 Auto Scaling 서비스 연결 역할의 이름으로 대체할 수 있습니다. 그러나 시작 구성을 사용하여 스팟 인스턴스를 시작하는 경우 AWSServiceRoleForEC2Spot 역할을 사용하십시오.

다음 리소스를 참조하십시오.

- AWS CLI를 사용하여 CMK를 생성하려면 [create-key](#)를 참조하십시오.
- AWS CLI를 사용하여 CMK 정책을 업데이트하려면 [put-key-policy](#)를 참조하십시오.
- 키 ID와 Amazon 리소스 이름(ARN)을 찾으려면 AWS Key Management Service Developer Guide의 [키 ID 및 ARN 찾기](#)를 참조하십시오.
- Amazon EC2 Auto Scaling 서비스 연결 역할에 대한 자세한 내용은 [Amazon EC2 Auto Scaling에 대한 서비스 연결 역할](#) (p. 176) 단원을 참조하십시오.

콘솔에서 키 정책 편집

다음 단원의 예에서는 키 정책을 변경하는 유일한 방법인 키 정책에 명령문을 추가하는 방법을 보여줍니다. 키 정책을 변경하는 가장 쉬운 방법은 키 정책에 IAM 콘솔의 기본 보기를 사용하고 IAM 엔터티(사용자 또는 역할)를 적절한 키 정책에 대한 키 사용자 중 하나로 만드는 것입니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [AWS Management Console의 기본 보기 사용](#)을 참조하십시오.

Important

주의할 점이 있습니다. 콘솔의 기본 보기 정책 명령문에는 CMK에서 AWS KMS Revoke 작업을 수행할 수 있는 권한이 포함되어 있습니다. 계정의 CMK에 AWS 계정 액세스 권한을 부여한 경우 실수로 이 권한 부여를 취소하면 외부 사용자가 더 이상 암호화된 데이터나 데이터 암호화에 사용된 키에 액세스할 수 없습니다.

예: CMK에 대한 액세스를 허용하는 CMK 키 정책 섹션

고객 관리형 CMK의 키 정책에 다음 두 정책 명령문을 추가하여 예의 ARN을 CMK에 대한 액세스가 허용된 적절한 서비스 연결 역할의 ARN으로 바꿉니다. 이 예에서 정책 섹션에서는 AWSServiceRoleForAutoScaling이라는 서비스 연결 역할에 고객 관리형 CMK를 사용할 수 있는 권한을 부여합니다.

```
{
  "Sid": "Allow service-linked role use of the CMK",
  "Effect": "Allow",
```

```
"Principal": {
  "AWS": [
    "arn:aws:iam::<123456789012>:role/aws-service-role/
autoscaling.amazonaws.com/AWSServiceRoleForAutoScaling"
  ],
  "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::<123456789012>:role/aws-service-role/
autoscaling.amazonaws.com/AWSServiceRoleForAutoScaling"
    ]
  },
  "Action": [
    "kms:CreateGrant"
  ],
  "Resource": "*",
  "Condition": {
    "Bool": {
      "kms:GrantIsForAWSResource": true
    }
  }
}
```

예: CMK에 대한 교차 계정 액세스를 허용하는 CMK 키 정책 섹션

고객 관리형 CMK가 Auto Scaling 그룹과 다른 계정에 있는 경우 CMK에 대한 액세스를 허용하는 키 정책과 함께 권한 부여를 사용해야 합니다. 자세한 내용은 AWS Key Management Service Developer Guide의 [권한 부여 사용](#)을 참조하십시오.

먼저 다음 두 가지 정책 명령문을 CMK의 키 정책에 추가하고 예의 ARN을 외부 계정의 ARN으로 바꾼 다음 키를 사용할 수 있는 계정을 지정하십시오. 이렇게 하면 IAM 정책을 사용하여 지정된 계정의 IAM 사용자 또는 역할에게 다음 CLI 명령을 사용하여 CMK에 대한 권한 부여를 생성할 수 있는 권한을 부여할 수 있습니다. AWS 계정에 CMK에 대한 전체 액세스 권한을 부여할 경우 그 자체로는 모든 IAM 사용자 또는 역할에 CMK에 대한 액세스 권한이 부여되지는 않습니다.

```
{
  "Sid": "Allow external account 111122223333 use of the CMK",
  "Effect": "Allow",
  "Principal": {
    "AWS": [
      "arn:aws:iam::<111122223333>:root"
    ]
  },
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*",
    "kms:DescribeKey"
  ]
}
```

```
],
  "Resource": "*"
}
```

```
{
  "Sid": "Allow attachment of persistent resources in external account 111122223333",
  "Effect": "Allow",
  "Principal": {
    "AWS": [
      "arn:aws:iam::111122223333:root"
    ]
  },
  "Action": [
    "kms:CreateGrant"
  ],
  "Resource": "*"
}
```

그런 다음 외부 계정에서 적절한 서비스 연결 역할에 관련 권한을 위임하는 권한 부여를 생성합니다. 권한 부여의 Grantee Principal 요소는 적절한 서비스 연결 역할의 ARN입니다. key-id는 CMK의 ARN입니다. 다음은 111122223333 계정의 서비스 연결 역할 AWSServiceRoleForAutoScaling에 444455556666 계정에서 CMK를 사용할 수 있는 권한을 부여하는 [create-a-grant](#) CLI 명령의 예입니다.

```
aws kms create-grant \
  --region us-west-2 \
  --key-id arn:aws:kms:us-west-2:444455556666:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d \
  --grantee-principal arn:aws:iam::111122223333:role/aws-service-role/
autoscaling.amazonaws.com/AWSServiceRoleForAutoScaling \
  --operations "Encrypt" "Decrypt" "ReEncryptFrom" "ReEncryptTo" "GenerateDataKey"
"GenerateDataKeyWithoutPlaintext" "DescribeKey" "CreateGrant"
```

이 명령이 성공하려면 요청을 하는 사용자에게 CreateGrant 작업에 대한 권한이 있어야 합니다. 다음 예의 IAM 정책에서는 111122223333 계정의 IAM 사용자 또는 역할이 444455556666 계정의 CMK에 대한 권한 부여를 생성하도록 허용합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow creation of grant for the CMK in external account 444455556666",
      "Effect": "Allow",
      "Action": "kms:CreateGrant",
      "Resource": "arn:aws:kms:us-west-2:444455556666:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    }
  ]
}
```

암호화된 볼륨이 있는 인스턴스를 시작할 때 필요한 고객 관리형 CMK에 대한 교차 계정 액세스를 구성하는데 문제가 있는 경우, [문제 해결 섹션 \(p. 205\)](#)을 참조하십시오.

자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서 및 [AWS Key Management Service Developer Guide](#)의 [Amazon EBS Encryption](#) 단원을 참조하십시오.

Amazon EC2 Auto Scaling의 규정 준수 확인

여러 AWS 규정 준수 프로그램의 일환으로 타사 감사자가 Amazon Web Services(AWS) 서비스의 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 서비스를 참조하십시오](#). 일반적인 내용은 [AWS 규정 준수 프로그램을 참조하십시오](#).

AWS Artifact를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 [AWS 아티팩트에서 보고서 다운로드](#)를 참조하십시오.

Amazon EC2 Auto Scaling를 사용할 때 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표, 관련 법률 및 규정에 따라 결정됩니다. AWS는 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 개발자 안내서의 [규칙을 사용하여 리소스 평가](#) – 이 AWS Config 서비스는 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 EC2 Auto Scaling의 복원성

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다.

AWS 리전에서는 물리적으로 분리되고 격리된 다수의 가용 영역을 제공하며 이러한 가용 영역은 짧은 지연 시간, 높은 처리량 및 높은 중복성을 갖춘 네트워크에 연결되어 있습니다.

가용 영역을 사용하면 중단 없이 영역 간에 자동으로 장애 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS 리전 및 가용 영역에 대한 자세한 내용은 [AWS 글로벌 인프라](#)를 참조하십시오.

관련 항목

- Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2의 복원성](#)

Amazon EC2 Auto Scaling의 인프라 보안

관리형 서비스인 Amazon EC2 Auto Scaling은 [Amazon Web Services: 보안 프로세스 개요](#) 백서에서 설명하는 AWS 글로벌 네트워크 보안 절차를 통해 보호됩니다.

AWS에서 게시한 API 호출을 사용하여 네트워크를 통해 Amazon EC2 Auto Scaling에 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 관련된 보안 액세스 키를 사용하여 서명해야 합니다. 또는 [AWS Security Token Service\(AWS STS\)](#)를 사용하여 임시 보안 자격 증명을 생성하여 요청에 서명할 수 있습니다.

관련 항목

- Linux 인스턴스용 Amazon EC2 사용 설명서의 [Amazon EC2의 인프라 보안](#)

Amazon EC2 Auto Scaling 및 인터페이스 VPC 엔드포인트

인터페이스 VPC(Virtual Private Cloud) 엔드포인트를 생성하여 VPC와 Amazon EC2 Auto Scaling API 간에 프라이빗 연결을 설정할 수 있습니다. 인터넷을 통해 트래픽을 보내지 않고 이 연결을 사용하여 VPC에서 Amazon EC2 Auto Scaling API를 호출할 수 있습니다. 엔드포인트는 Amazon EC2 Auto Scaling API에 안정적이고 확장 가능한 연결을 제공합니다. 이 작업은 인터넷 게이트웨이, NAT 인스턴스 또는 VPN 연결이 필요하지 않습니다.

인터페이스 VPC 엔드포인트는 프라이빗 IP 주소를 사용하여 AWS 서비스 간에 프라이빗 통신을 활성화하는 기능인 AWS PrivateLink에 의해 구동됩니다. 자세한 내용은 [AWS PrivateLink](#)를 참조하십시오.

Note

인터페이스 VPC 엔드포인트를 통해 액세스하려는 각 API를 명시적으로 활성화해야 합니다. 예를 들어 Amazon EC2 API 작업을 호출할 때 사용하도록 `ec2.region.amazonaws.com`에 대한 인터페이스 VPC 엔드포인트를 구성해야 할 수도 있습니다. 자세한 내용은 Amazon VPC 사용 설명서의 인터페이스 [VPC 엔드포인트\(AWS PrivateLink\)](#)를 참조하십시오.

인터페이스 VPC 엔드포인트 생성

다음 서비스 이름을 사용하여 Amazon EC2 Auto Scaling에 대한 엔드포인트를 생성합니다.

- `com.amazonaws.region.autoscaling` — Amazon EC2 Auto Scaling API 작업에 대한 엔드포인트를 생성합니다.

자세한 내용은 Amazon VPC 사용 설명서의 [인터페이스 엔드포인트 생성](#)을 참조하십시오.

엔드포인트가 기본 DNS 호스트 이름(예: `autoscaling.us-east-1.amazonaws.com`)을 사용하여 지원되는 서비스에 API 요청을 할 수 있도록 프라이빗 DNS를 활성화합니다. 이 설정은 AWS 서비스에 대한 엔드포인트를 생성할 때 기본적으로 활성화됩니다. 자세한 내용은 Amazon VPC 사용 설명서의 [인터페이스 엔드포인트를 통해 서비스 액세스](#)를 참조하십시오.

Amazon EC2 Auto Scaling 설정을 변경할 필요가 없습니다. Amazon EC2 Auto Scaling은 퍼블릭 엔드포인트 또는 프라이빗 인터페이스 VPC 엔드포인트 중 사용 중인 엔드포인트를 사용하여 다른 AWS 서비스를 호출합니다.

VPC 엔드포인트 정책 생성

VPC 엔드포인트에 정책을 연결하여 Amazon EC2 Auto Scaling 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

다음 예에서는 엔드포인트를 통해 조정 정책을 생성할 수 있는 모든 사용자 권한을 거부하는 VPC 엔드포인트 정책을 보여 줍니다. 또한 이 정책 예에서는 모든 사용자에게 다른 모든 작업을 수행할 수 있는 권한을 부여합니다.

```
{
  "Statement": [
    {
```

```
        "Action": "*",
        "Effect": "Allow",
        "Resource": "*",
        "Principal": "*"
    },
    {
        "Action": "autoscaling:PutScalingPolicy",
        "Effect": "Deny",
        "Resource": "*",
        "Principal": "*"
    }
]
}
```

자세한 내용은 Amazon VPC 사용 설명서의 [VPC 엔드포인트 정책 사용](#)을 참조하십시오.

Amazon EC2 Auto Scaling 문제 해결

Amazon EC2 Auto Scaling은 문제를 해결할 때 도움이 되도록 구체적이고 서술적인 오류를 제공합니다. 조정 활동에 관한 설명에서 오류 메시지를 확인할 수 있습니다.

목차

- [오류 메시지 검색](#) (p. 200)
- [Amazon EC2 Auto Scaling 문제 해결: EC2 인스턴스 시작 실패](#) (p. 202)
- [Amazon EC2 Auto Scaling 문제 해결: AMI 문제](#) (p. 206)
- [Amazon EC2 Auto Scaling 문제 해결: 로드 밸런서 문제](#) (p. 207)
- [Auto Scaling 문제 해결: 용량 제한](#) (p. 209)

오류 메시지 검색

조정 활동의 설명에서 오류 메시지를 검색하려면, 다음과 같이 `describe-scaling-activities` 명령을 사용합니다.

```
aws autoscaling describe-scaling-activities --auto-scaling-group-name my-asg
```

다음은 `StatusCode`에 활동의 현재 상태가 포함되고 `StatusMessage`에 오류 메시지가 포함된 응답의 예입니다.

```
{
  "Activities": [
    {
      "Description": "Launching a new EC2 instance: i-4ba0837f",
      "AutoScalingGroupName": "my-asg",
      "ActivityId": "f9f2d65b-f1f2-43e7-b46d-d86756459699",
      "Details": "{\"Availability Zone\":\"us-west-2c\"}",
      "StartTime": "2013-08-19T20:53:29.930Z",
      "Progress": 100,
      "EndTime": "2013-08-19T20:54:02Z",
      "Cause": "At 2013-08-19T20:53:25Z a user request created an
AutoScalingGroup...",
      "StatusCode": "Failed",
      "StatusMessage": "The image id 'ami-4edb0327' does not exist. Launching EC2
instance failed."
    }
  ]
}
```

다음 표에는 오류 메시지의 유형과 문제 해결에 사용할 수 있는 문제 해결 리소스에 대한 링크가 나와 있습니다.

EC2 인스턴스 시작 실패

문제	오류 메시지
Auto Scaling 그룹	AutoScalingGroup <Auto Scaling 그룹 이름>을(를) 찾을 수 없습니다. (p. 203)

문제	오류 메시지
가용 영역	요청된 가용 영역이 더 이상 지원되지 않습니다. 가용 영역을 지정하지 않거나... (p. 203)
AWS 계정	이 서비스를 구독하고 있지 않습니다. https://aws.amazon.com/ 단원을 참조하십시오. (p. 204)
블록 디바이스 매핑	잘못된 디바이스 이름이 업로드되었습니다. EC2 인스턴스 시작에 실패했습니다. (p. 204)
블록 디바이스 매핑	파라미터 virtualName에 대한 값(<인스턴스 스토리지 디바이스와 연결된 이름>)이 잘못되었습니다... (p. 204)
블록 디바이스 매핑	EBS 블록 디바이스 매핑이 인스턴스 스토어 AMI에 대해 지원되지 않습니다. (p. 205)
인스턴스 유형 및 가용 영역	요청된 인스턴스 유형(<인스턴스 유형>)이 요청된 가용 영역(<인스턴스 가용 영역>)에서 지원되지 않습니다... (p. 204)
키 페어	키 페어 <EC2 인스턴스와 연결된 키 페어>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다. (p. 203)
실행 구성	요청된 구성이 현재 지원되지 않습니다. (p. 203)
배치 그룹	배치 그룹은 'm1.large' 유형의 인스턴스와 함께 사용할 수 없습니다. EC2 인스턴스 시작에 실패했습니다. (p. 205)
보안 그룹	보안 그룹 <보안 그룹 이름>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다. (p. 203)
서비스 연결 역할	Client.InternalError: 시작 시 클라이언트 오류. (p. 205)

AMI 문제

문제	오류 메시지
AMI ID	AMI ID <AMI ID>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다. (p. 206)
AMI ID	AMI <AMI ID>이(가) 보류 중이며 실행할 수 없습니다. EC2 인스턴스 시작에 실패했습니다. (p. 207)
AMI ID	파라미터 virtualName의 값(<ami ID>)이 잘못되었습니다. (p. 207)
아키텍처 불일치	요청된 인스턴스 유형의 아키텍처(i386)가 ami-6622f00f(x86_64)용 매니페스트의 아키텍처와 일치하지 않습니다. EC2 인스턴스 시작에 실패했습니다. (p. 207)

로드 밸런서 문제

문제	오류 메시지
로드 밸런서를 찾을 수 없음	로드 밸런서 <시작 환경>을 찾을 수 없습니다. 로드 밸런서 구성의 유효성 검사에 실패했습니다. (p. 208)
VPC의 인스턴스	EC2 인스턴스 <인스턴스 ID>이(가) VPC에 없습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다. (p. 208)

문제	오류 메시지
활성 로드 밸런서 없음	이름이 <로드 밸런서 이름>인 활성 로드 밸런서가 없습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다. (p. 208)
보안 토큰	요청에 포함된 보안 토큰이 잘못되었습니다. 로드 밸런서 구성의 유효성 검사에 실패했습니다. (p. 209)

용량 제한

문제	오류 메시지
용량 제한	<인스턴스 수>개의 인스턴스가 이미 실행 중입니다. EC2 인스턴스 시작에 실패했습니다. (p. 209)
가용 영역의 용량 부족	현재 요청한 가용 영역(<요청한 가용 영역>)에 충분한 <인스턴스 유형> 용량을 갖고 있지 않습니다... (p. 209)

Amazon EC2 Auto Scaling 문제 해결: EC2 인스턴스 시작 실패

이 페이지에는 시작에 실패한 EC2 인스턴스, 잠재적인 원인, 문제 해결을 위해 취할 수 있는 조치에 대한 정보가 나와 있습니다.

오류 메시지를 검색하려면 [오류 메시지 검색](#) (p. 200)를 참조하십시오.

EC2 인스턴스 시작에 실패한 경우, 다음 오류 메시지 중 하나 이상이 표시될 수 있습니다.

오류 메시지

- 보안 그룹 <보안 그룹 이름>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다. (p. 203)
- 키 페어 <EC2 인스턴스와 연결된 키 페어>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다. (p. 203)
- 요청된 구성이 현재 지원되지 않습니다. (p. 203)
- AutoScalingGroup <Auto Scaling 그룹 이름>을(를) 찾을 수 없습니다. (p. 203)
- 요청된 가용 영역이 더 이상 지원되지 않습니다. 가용 영역을 지정하지 않거나... (p. 203)
- 요청된 인스턴스 유형(<인스턴스 유형>)이 요청된 가용 영역(<인스턴스 가용 영역>)에서 지원되지 않습니다... (p. 204)
- 이 서비스를 구독하고 있지 않습니다. <https://aws.amazon.com/> 단원을 참조하십시오. (p. 204)
- 잘못된 디바이스 이름이 업로드되었습니다. EC2 인스턴스 시작에 실패했습니다. (p. 204)
- 파라미터 virtualName에 대한 값(<인스턴스 스토리지 디바이스와 연결된 이름>)이 잘못되었습니다... (p. 204)
- EBS 블록 디바이스 매핑이 인스턴스 스토어 AMI에 대해 지원되지 않습니다. (p. 205)
- 배치 그룹은 'm1.large' 유형의 인스턴스와 함께 사용할 수 없습니다. EC2 인스턴스 시작에 실패했습니다. (p. 205)
- Client.InternalError: 시작 시 클라이언트 오류. (p. 205)

보안 그룹 <보안 그룹 이름>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다.

- 원인: 시작 구성에 지정된 보안 그룹이 삭제되었을 수 있습니다.
- 솔루션:
 1. `describe-security-groups` 명령을 사용하여 계정에 연결된 보안 그룹 목록을 가져옵니다.
 2. 목록에서 사용할 보안 그룹을 선택합니다. 보안 그룹을 만들려면 `create-security-group` 명령을 사용합니다.
 3. 새 시작 구성을 생성합니다.
 4. `update-auto-scaling-group` 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

키 페어 <EC2 인스턴스와 연결된 키 페어>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다.

- 원인: 인스턴스를 시작할 때 사용한 키 페어가 삭제되었을 수 있습니다.
- 솔루션:
 1. `describe-key-pairs` 명령을 사용하여 사용 가능한 키 페어 목록을 가져옵니다.
 2. 목록에서 사용할 키 페어를 선택합니다. 키 페어를 만들려면 `create-key-pair` 명령을 사용합니다.
 3. 새 시작 구성을 생성합니다.
 4. `update-auto-scaling-group` 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

요청된 구성이 현재 지원되지 않습니다.

- 원인: 시작 구성의 일부 옵션이 현재 지원되지 않을 수 있습니다.
- 솔루션:
 1. 새 시작 구성을 생성합니다.
 2. `update-auto-scaling-group` 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

AutoScalingGroup <Auto Scaling 그룹 이름>을(를) 찾을 수 없습니다.

- 원인: Auto Scaling 그룹이 삭제되었을 수 있습니다.
- 솔루션: 새 Auto Scaling 그룹을 생성합니다.

요청된 가용 영역이 더 이상 지원되지 않습니다. 가용 영역을 지정하지 않거나...

- 오류 메시지: 요청된 가용 영역이 더 이상 지원되지 않습니다. 가용 영역을 지정하지 않거나 <사용 가능한 가용 영역 목록>을 선택하여 요청을 다시 시도하십시오. EC2 인스턴스 시작에 실패했습니다.
- 원인: Auto Scaling 그룹에 연결된 가용 영역이 현재 사용 가능하지 않을 수 있습니다.

- 솔루션: Auto Scaling 그룹을 오류 메시지의 권장 사항으로 업데이트합니다.

요청된 인스턴스 유형(<인스턴스 유형>)이 요청된 가용 영역(<인스턴스 가용 영역>)에서 지원되지 않습니다...

- 오류 메시지: 요청된 인스턴스 유형(<인스턴스 유형>)이 요청된 가용 영역(<인스턴스 가용 영역>)에서 지원되지 않습니다. 가용 영역을 지정하지 않거나 <이 인스턴스 유형을 지원하는 가용 영역 목록>을 선택하여 요청을 다시 시도하십시오. EC2 인스턴스 시작에 실패했습니다.
- 원인: 시작 구성에 연결된 인스턴스 유형이 현재 Auto Scaling 그룹에 지정된 가용 영역에서 사용 가능하지 않을 수 있습니다.
- 솔루션: Auto Scaling 그룹을 오류 메시지의 권장 사항으로 업데이트합니다.

이 서비스를 구독하고 있지 않습니다. <https://aws.amazon.com/> 단원을 참조하십시오.

- 원인: AWS 계정이 만료되었을 수 있습니다.
- 솔루션: <https://aws.amazon.com/>으로 이동한 후 지금 가입을 선택하여 새 계정을 개설합니다.

잘못된 디바이스 이름이 업로드되었습니다. EC2 인스턴스 시작에 실패했습니다.

- 원인: 시작 구성의 블록 디바이스 매핑에 사용할 수 없거나 현재 지원되지 않는 블록 디바이스 이름이 포함되었을 수 있습니다.
- 솔루션:
 1. [describe-volumes](#) 명령을 사용하여 볼륨이 인스턴스에 노출되는 방식을 확인합니다.
 2. 볼륨 설명에 나와 있는 디바이스 이름을 사용하여 새 시작 구성을 생성합니다.
 3. [update-auto-scaling-group](#) 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

파라미터 virtualName에 대한 값(<인스턴스 스토리지 디바이스와 연결된 이름>)이 잘못되었습니다...

- 오류 메시지: 파라미터 virtualName에 대한 값(<인스턴스 스토리지 디바이스와 연결된 이름>)이 잘못되었습니다. 예상되는 형식: 'ephemeralNUMBER'. EC2 인스턴스 시작에 실패했습니다.
- 원인: 블록 디바이스와 연결된 가상 이름에 지정된 형식이 잘못되었습니다.
- 솔루션:
 1. virtualName 파라미터의 디바이스 이름을 지정하여 새 시작 구성을 생성합니다. 디바이스 이름 형식에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [인스턴스 스토리지 디바이스 이름](#)을 참조하십시오.
 2. [update-auto-scaling-group](#) 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

EBS 블록 디바이스 매핑이 인스턴스 스토어 AMI에 대해 지원되지 않습니다.

- 원인: 시작 구성에 지정된 블록 디바이스 매핑이 사용자 인스턴스에 대해 지원되지 않습니다.
- 솔루션:
 1. 사용자 인스턴스 유형에서 지원되는 블록 디바이스 매핑으로 새 시작 구성을 생성합니다. 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [블록 디바이스 매핑](#)을 참조하십시오.
 2. `update-auto-scaling-group` 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

배치 그룹은 'm1.large' 유형의 인스턴스와 함께 사용할 수 없습니다. EC2 인스턴스 시작에 실패했습니다.

- 원인: 클러스터 배치 그룹에 잘못된 인스턴스 유형이 포함되어 있습니다.
- 솔루션:
 1. 배치 그룹에서 지원되는 유효한 인스턴스 유형에 대한 자세한 내용은 Linux 인스턴스용 Amazon EC2 사용 설명서에서 [배치 그룹](#)을 참조하십시오.
 2. [배치 그룹](#)에 자세히 나와 있는 지침에 따라 새 배치 그룹을 생성합니다.
 3. 또는, 지원되는 인스턴스 유형으로 새 시작 구성을 생성합니다.
 4. `update-auto-scaling-group` 명령을 사용하여 새로운 배치 그룹 또는 시작 구성을 통해 Auto Scaling 그룹을 업데이트합니다.

Client.InternalError: 시작 시 클라이언트 오류.

- 원인: 이 오류는 Auto Scaling 그룹이 암호화된 EBS 볼륨을 갖는 인스턴스를 시작하려고 하는데 [서비스 연결 역할 \(p. 176\)](#)이 암호화에 사용되는 [고객 관리형 CMK](#)에 대한 액세스 권한이 없을 때 발생할 수 있습니다.
- 솔루션: Auto Scaling 그룹이 인스턴스를 시작하도록 허용하는 추가 설정이 필요합니다. 다음 표는 오류 해결 단계를 요약한 것입니다. 자세한 내용은 <https://forums.aws.amazon.com/thread.jspa?threadID=277523>을 참조하십시오.

시나리오	다음 단계
시나리오 1: CMK와 Auto Scaling 그룹이 동일한 AWS 계정에 있습니다.	다음과 같이 서비스 연결 역할이 CMK를 사용하도록 허용합니다. <ol style="list-style-type: none">1. 이 Auto Scaling 그룹에 어떤 서비스 연결 역할을 사용할지 결정합니다.2. CMK의 키 정책을 업데이트하여 서비스 연결 역할이 CMK를 사용하도록 허용합니다.3. Auto Scaling 그룹이 서비스 연결 역할을 사용하도록 업데이트합니다.
시나리오 2: CMK와 Auto Scaling 그룹이 서로 다른 AWS 계정에 있습니다.	두 가지 해결 방안이 있습니다. 해결 방안 1: 동일한 AWS 계정의 CMK를 Auto Scaling 그룹으로 사용합니다.

시나리오	다음 단계
	<ol style="list-style-type: none"> 1. 스냅샷을 복사하고 Auto Scaling 그룹과 동일한 계정에 속한 다른 CMK로 다시 암호화합니다. 2. 서비스 연결 역할이 새 CMK를 사용하도록 허용합니다. 시나리오 1에 대한 단계를 참조하십시오. <p>해결 방안 2: Auto Scaling 그룹과 다른 AWS 계정의 CMK를 계속 사용합니다.</p> <ol style="list-style-type: none"> 1. 이 Auto Scaling 그룹에 어떤 서비스 연결 역할을 사용할지 결정합니다. 2. Auto Scaling 그룹 계정(루트 사용자)이 CMK에 액세스하도록 허용합니다. 3. 권한 부여를 생성할 수 있는 Auto Scaling 그룹 계정에서 IAM 사용자 또는 역할을 정의합니다. 4. 피부여자 보안 주체로서의 서비스 연결 역할로 CMK에 권한 부여를 생성합니다. 5. Auto Scaling 그룹이 서비스 연결 역할을 사용하도록 업데이트합니다.

Amazon EC2 Auto Scaling 문제 해결: AMI 문제

이 페이지에는 AMI와 관련된 문제, 잠재적인 원인, 문제 해결을 위해 취할 수 있는 조치에 대한 정보가 나와 있습니다.

오류 메시지를 검색하려면 [오류 메시지 검색 \(p. 200\)](#)를 참조하십시오.

AMI 관련 문제로 인해 EC2 인스턴스 시작에 실패한 경우, 다음 오류 메시지 중 하나 이상이 표시될 수 있습니다.

오류 메시지

- AMI ID <AMI ID>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다. (p. 206)
- AMI <AMI ID>이(가) 보류 중이며 실행할 수 없습니다. EC2 인스턴스 시작에 실패했습니다. (p. 207)
- 파라미터 virtualName의 값(<ami ID>)이 잘못되었습니다. (p. 207)
- 요청된 인스턴스 유형의 아키텍처(i386)가 ami-6622f00f(x86_64)용 매니페스트의 아키텍처와 일치하지 않습니다. EC2 인스턴스 시작에 실패했습니다. (p. 207)

AMI ID <AMI ID>이(가) 존재하지 않습니다. EC2 인스턴스 시작에 실패했습니다.

- 원인: 시작 구성을 생성한 후 AMI가 삭제되었을 수 있습니다.
- 솔루션:
 1. 유효한 AMI를 사용하여 새 시작 구성을 생성합니다.
 2. [update-auto-scaling-group](#) 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

AMI <AMI ID>이(가) 보류 중이며 실행할 수 없습니다. EC2 인스턴스 시작에 실패했습니다.

- 원인: AMI를 방금 전에 생성하여(실행 중인 인스턴스의 스냅샷을 만들거나 기타 다른 방법으로) 아직 사용 가능한 상태가 아닐 수 있습니다.
- 솔루션: AMI가 사용 가능해질 때까지 기다린 다음 시작 구성을 생성해야 합니다.

파라미터 virtualName의 값(<ami ID>)이 잘못되었습니다.

- 원인: 잘못된 값입니다. virtualName 파라미터는 디바이스와 연결된 가상 이름을 나타냅니다.
- 솔루션:
 1. virtualName 파라미터에 대한 인스턴스 가상 디바이스의 이름을 지정하여 새 시작 구성을 생성합니다.
 2. `update-auto-scaling-group` 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

요청된 인스턴스 유형의 아키텍처(i386)가 ami-6622f00f(x86_64)용 매니페스트의 아키텍처와 일치하지 않습니다. EC2 인스턴스 시작에 실패했습니다.

- 원인: 시작 구성에 언급된 InstanceType의 아키텍처가 이미지 아키텍처와 일치하지 않습니다.
- 솔루션:
 1. 요청된 인스턴스 유형의 AMI 아키텍처를 사용하여 새 시작 구성을 생성합니다.
 2. `update-auto-scaling-group` 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

Amazon EC2 Auto Scaling 문제 해결: 로드 밸런서 문제

이 페이지에는 Auto Scaling 그룹과 연결된 로드 밸런서로 인해 발생한 문제, 잠재적인 원인, 문제 해결을 위해 취할 수 있는 조치에 대한 정보가 나와 있습니다.

오류 메시지를 검색하려면 [오류 메시지 검색 \(p. 200\)](#)을 참조하십시오.

Auto Scaling 그룹에 연결된 로드 밸런서 관련 문제로 인해 EC2 인스턴스 시작에 실패한 경우, 다음 오류 메시지 중 하나 이상이 표시될 수 있습니다.

오류 메시지

- 로드 밸런서 <시작 환경>을 찾을 수 없습니다. 로드 밸런서 구성의 유효성 검사에 실패했습니다. (p. 208)
- 이름이 <로드 밸런서 이름>인 활성 로드 밸런서가 없습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다. (p. 208)
- EC2 인스턴스 <인스턴스 ID>이(가) VPC에 없습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다. (p. 208)

- EC2 인스턴스 <인스턴스 ID>이(가) VPC에 있습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다. (p. 208)
- 요청에 포함된 보안 토큰이 잘못되었습니다. 로드 밸런서 구성의 유효성 검사에 실패했습니다. (p. 209)

로드 밸런서 <시작 환경>을 찾을 수 없습니다. 로드 밸런서 구성의 유효성 검사에 실패했습니다.

- 원인 1: 로드 밸런서가 삭제되었습니다.
- 솔루션 1:
 1. 로드 밸런서가 아직 존재하는지 확인합니다. [describe-load-balancers](#) 명령을 사용할 수 있습니다.
 2. 응답 목록에 해당 로드 밸런서가 있다면 원인 2를 확인하십시오.
 3. 응답 목록에 해당 로드 밸런서가 없다면, 새 로드 밸런서를 생성한 다음 새 Auto Scaling 그룹을 생성하거나 로드 밸런서 없이 새 Auto Scaling 그룹을 생성할 수 있습니다.
- 원인 2: Auto Scaling 그룹을 만들 때 로드 밸런서 이름이 올바른 순서로 지정되어 있지 않았습니다.
- 솔루션 2: 새 Auto Scaling 그룹을 생성하여 마지막에 로드 밸런서 이름을 지정합니다.

이름이 <로드 밸런서 이름>인 활성 로드 밸런서가 없습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다.

- 원인: 지정된 로드 밸런서가 삭제되었을 수 있습니다.
- 솔루션: 새 로드 밸런서를 생성한 다음 새 Auto Scaling 그룹을 생성하거나 로드 밸런서 없이 새 Auto Scaling 그룹을 생성할 수 있습니다.

EC2 인스턴스 <인스턴스 ID>이(가) VPC에 없습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다.

- 원인: 지정된 인스턴스가 VPC에 존재하지 않습니다.
- 솔루션: 인스턴스에 연결된 로드 밸런서를 삭제하거나 새 Auto Scaling 그룹을 생성할 수 있습니다.

EC2 인스턴스 <인스턴스 ID>이(가) VPC에 있습니다. 로드 밸런서 구성을 업데이트하는 데 실패했습니다.

- 원인: 로드 밸런서는 EC2-Classic에 있지만 Auto Scaling 그룹이 VPC에 없습니다.
- 솔루션: 로드 밸런서 및 Auto Scaling 그룹이 동일한 네트워크(EC2-Classic 또는 VPC)에 있는지 확인합니다.

요청에 포함된 보안 토큰이 잘못되었습니다. 로드 밸런서 구성의 유효성 검사에 실패했습니다.

- 원인: AWS 계정이 만료되었을 수 있습니다.
- 솔루션: AWS 계정이 유효한지 확인합니다. <https://aws.amazon.com/>으로 이동 후 지금 가입을 선택하여 새 계정을 개설합니다.

Auto Scaling 문제 해결: 용량 제한

이 페이지에는 Auto Scaling 그룹의 용량 제한과 관련된 문제, 잠재적인 원인, 문제 해결을 위해 취할 수 있는 조치에 대한 정보가 나와 있습니다. Amazon EC2 Auto Scaling 제한에 대한 자세한 내용은 [Amazon EC2 Auto Scaling 서비스 할당량](#) (p. 9) 단원을 참조하십시오.

오류 메시지를 검색하려면 [오류 메시지 검색](#) (p. 200)를 참조하십시오.

Auto Scaling 그룹의 용량 제한 관련 문제로 인해 EC2 인스턴스 시작에 실패한 경우, 다음 오류 메시지 중 하나 이상이 표시될 수 있습니다.

오류 메시지

- 현재 요청한 가용 영역(<요청한 가용 영역>)에 충분한 <인스턴스 유형> 용량을 갖고 있지 않습니다... (p. 209)
- <인스턴스 수>개의 인스턴스가 이미 실행 중입니다. EC2 인스턴스 시작에 실패했습니다. (p. 209)

현재 요청한 가용 영역(<요청한 가용 영역>)에 충분한 <인스턴스 유형> 용량을 갖고 있지 않습니다...

- 오류 메시지: 현재 요청한 가용 영역(<요청한 가용 영역>)에 충분한 <인스턴스 유형> 용량을 갖고 있지 않습니다. 시스템에서 추가 용량을 프로비전하기 위한 작업이 진행 중입니다. 요청에서 가용 영역을 지정하지 않거나 <현재 이 인스턴스 유형을 지원하는 가용 영역 목록>을 선택하여 지금 <인스턴스 유형> 용량을 확보할 수 있습니다. EC2 인스턴스 시작에 실패했습니다.
- 원인: 현재 Auto Scaling에서는 요청한 가용 영역에서 해당 인스턴스 유형을 지원하지 않습니다.
- 솔루션:
 1. 오류 메시지의 권장 사항에 따라 새 시작 구성을 생성합니다.
 2. [update-auto-scaling-group](#) 명령을 사용하여 Auto Scaling 그룹을 새로운 시작 구성으로 업데이트합니다.

<인스턴스 수>개의 인스턴스가 이미 실행 중입니다. EC2 인스턴스 시작에 실패했습니다.

- 원인: Auto Scaling 그룹이 `DesiredCapacity` 파라미터로 설정된 제한에 도달했습니다.
- 솔루션:
 - [update-auto-scaling-group](#) 명령을 사용하고 `--desired-capacity` 파라미터에 새로운 값을 제공하여 Auto Scaling 그룹을 업데이트합니다.
 - EC2 인스턴스 수의 제한에 도달한 경우 증가분을 요청할 수 있습니다. 자세한 내용은 [AWS 서비스 제한](#)을 참조하십시오.

Auto Scaling 리소스

다음 표에는 이 서비스를 이용할 때 참조할 수 있는 관련 리소스가 나와 있습니다.

- [Amazon EC2 Auto Scaling](#) – Amazon EC2 Auto Scaling에 대한 정보를 제공하는 기본 웹 페이지입니다.
- [Amazon EC2 기술 FAQ](#) – Amazon EC2 및 Amazon EC2 Auto Scaling에 대한 고객 질문의 답입니다.
- [Amazon EC2 토론 포럼](#) – 커뮤니티에서 도움을 구할 수 있습니다.
- [AWS Auto Scaling 사용 설명서](#) – AWS Auto Scaling 콘솔을 사용하면 여러 서비스의 확장 기능을 보다 쉽게 사용할 수 있습니다. AWS Auto Scaling를 통해 Auto Scaling 그룹의 동적 조정 정책 정의 프로세스를 간소화하고 예측 조정을 사용하여 예측된 트래픽 변경에 앞서 Amazon EC2 용량을 확장할 수 있습니다.

AWS에 대해 더 자세히 배울 수 있는 추가 리소스가 아래에 있습니다.

- [교육 및 워크숍](#) – 역할 기반의 과정 및 전문 과정은 물론 자습형 실습에 대한 링크를 통해 AWS 기술을 연마하고 실용적인 경험을 쌓을 수 있습니다.
- [AWS 개발자 도구](#) – AWS 애플리케이션을 개발 및 관리하기 위한 개발자 도구, SDK, IDE 도구 키트 및 명령줄 도구 링크.
- [AWS 백서](#) – AWS 솔루션 아키텍트 또는 기타 기술 전문가가 아키텍처, 보안 및 경제 등의 주제에 대해 작성한 포괄적 AWS 기술 백서 목록의 링크.
- [AWS Support 센터](#) – AWS 지원 사례를 생성 및 관리하는 허브. 또한 포럼, 기술 FAQ, 서비스 상태 및 AWS Trusted Advisor 등의 기타 유용한 자료에 대한 링크가 있습니다.
- [AWS Support](#) – 클라우드에서 1대 1로 애플리케이션을 구축 및 실행하도록 지원하는 빠른 응답 지원 채널인 AWS Support에 대한 정보가 포함된 기본 웹 페이지.
- [문의처](#) – AWS 결제, 계정, 이벤트, 침해 및 기타 문제에 대해 문의할 수 있는 중앙 연락 창구입니다.
- [AWS 사이트 약관](#) – 저작권 및 상표, 사용자 계정, 라이선스 및 사이트 액세스와 기타 주제에 대한 세부 정보.

문서 기록

다음 표에서는 2018년 7월 이후의 Amazon EC2 Auto Scaling 설명서에 대한 중요 추가 사항을 설명합니다. 이 설명서에 대한 업데이트 알림을 받으려면 RSS 피드를 구독하면 됩니다.

update-history-change	update-history-description	update-history-date
안내서 변경 사항 (p. 211)	IAM에 대한 다양한 개선 사항 설명서 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 템플릿 지원 시작 및 Amazon EC2 Auto Scaling ID 기반 정책 예 를 참조하십시오.	March 4, 2020
조정 정책 비활성화 (p. 211)	이제 조정 정책을 비활성화했다가 다시 활성화할 수 있습니다. 이 기능을 사용하면 나중에 정책을 다시 활성화할 수 있도록 구성 세부 정보를 유지하면서 조정 정책을 일시적으로 비활성화할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Auto Scaling 그룹에 대한 조정 정책 비활성화 를 참조하십시오.	February 18, 2020
알림 기능 추가 (p. 211)	Amazon EC2 Auto Scaling은 이제 누락된 보안 그룹 또는 시작 템플릿으로 인해 Auto Scaling 그룹을 확장할 수 없는 경우 AWS Personal Health Dashboard에 이벤트를 보냅니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Amazon EC2 Auto Scaling에 대한 Personal Health Dashboard 알림 을 참조하십시오.	February 12, 2020
안내서 변경 사항 (p. 211)	Amazon EC2 Auto Scaling 사용 설명서의 Amazon EC2 Auto Scaling에서 IAM을 사용하는 방식 , Amazon EC2 Auto Scaling 자격 증명 기반 정책 예제 , 암호화된 볼륨에 사용할 경우 필요한 CMK 키 정책 및 Auto Scaling 인스턴스 및 그룹 모니터링 섹션에 대한 다양한 개선 및 수정 사항입니다.	February 10, 2020
안내서 변경 사항 (p. 211)	인스턴스 가중치를 사용하는 Auto Scaling 그룹에 대한 문서가 개선되었습니다. “용량 유닛”을 사용하여 원하는 용량을 측정할 때 조정 정책을 사용하는 방법을 알아봅니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 조정 정책 작동 방식 및 조정 조절 유형 을 참조하십시오.	February 6, 2020

새로운 “보안” 장 (p. 211)	Amazon EC2 Auto Scaling 사용 설명서의 새로운 보안 장은 Amazon EC2 Auto Scaling을 사용할 때 공동 책임 모델 을 적용하는 방법을 이해하는 데 도움이 됩니다. 이번 업데이트에서는 사용 설명서의 “Amazon EC2 Auto Scaling 액세스 제어” 장이 새롭고 더욱 유용한 내용인 Amazon EC2 Auto Scaling의 ID 및 액세스 관리 단원으로 대체되었습니다.	February 4, 2020
전용 호스트 및 호스트 리소스 그룹 (p. 211)	호스트 리소스 그룹을 지정하는 시작 템플릿을 만드는 방법을 보여주는 안내서가 업데이트되었습니다. 이렇게 하면 전용 호스트에서 사용할 BYOL AMI를 지정하는 시작 템플릿이 있는 Auto Scaling 그룹을 생성할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Auto Scaling 그룹에 대한 시작 템플릿 생성 을 참조하십시오.	December 3, 2019
인스턴스 유형에 대한 권장 사항 (p. 211)	AWS Compute Optimizer는 성능 향상, 비용 절감 또는 두 가지 모두를 위한 Amazon EC2 인스턴스 권장 사항을 제공합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 인스턴스 유형에 대한 권장 사항 가져오기 를 참조하십시오.	December 3, 2019
Amazon VPC 엔드포인트 지원 (p. 211)	이제 VPC와 Amazon EC2 Auto Scaling 간에 프라이빗 연결을 설정할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Amazon EC2 Auto Scaling 및 인터페이스 VPC 엔드포인트 를 참조하십시오.	November 22, 2019
최대 인스턴스 수명 (p. 211)	이제 인스턴스가 서비스될 수 있는 최대 시간을 지정하여 인스턴스를 자동으로 교체할 수 있습니다. Amazon EC2 Auto Scaling은 이 제한에 가까워지는 인스턴스를 점진적으로 교체합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 최대 인스턴스 수명을 기준으로 Auto Scaling 인스턴스 교체 를 참조하십시오.	November 19, 2019

인스턴스 가중치 부여 (p. 211)	여러 인스턴스 유형이 포함된 Auto Scaling 그룹의 경우, 각 인스턴스 유형이 그룹의 용량에 기여하는 용량 유닛 수를 선택적으로 지정할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Amazon EC2 Auto Scaling 인스턴스 가중치 부여 를 참조하십시오.	November 19, 2019
최소 인스턴스 유형 수 (p. 211)	스팟, 온디맨드 및 예약 인스턴스 그룹에 대한 추가 인스턴스 유형을 더 이상 지정할 필요가 없습니다. 모든 Auto Scaling 그룹에 대해 이제 최소 인스턴스 유형은 한 개입니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Auto Scaling 그룹의 여러 인스턴스 유형 및 구매 옵션 을 참조하십시오.	September 16, 2019
새로운 스팟 할당 전략 지원 (p. 211)	Amazon EC2 Auto Scaling은 이제 새로운 "용량 최적화" 스팟 할당 전략을 지원합니다. 이 전략은 사용 가능한 스팟 용량에 따라 최적으로 선택된 스팟 인스턴스 풀을 사용하여 사용자의 요청을 이행합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Auto Scaling 그룹의 여러 인스턴스 유형 및 구매 옵션 을 참조하십시오.	August 12, 2019
안내서 변경 사항 (p. 211)	Amazon EC2 Auto Scaling 설명서의 서비스 연결 역할 사용 및 암호화된 볼륨에 사용할 경우 필요한 CMK 키 정책 주제가 개선되었습니다.	August 1, 2019
태깅에 대한 지원 향상 (p. 211)	Amazon EC2 Auto Scaling은 이제 인스턴스를 시작하는 동일한 API 호출의 일부로 Amazon EC2 인스턴스에 태그를 추가합니다. 자세한 내용은 Auto Scaling 그룹 및 인스턴스에 태그 지정 을 참조하십시오.	July 26, 2019
안내서 변경 사항 (p. 211)	조정 프로세스 일시 중단 및 다시 시작 주제의 Amazon EC2 Auto Scaling 설명서가 향상되었습니다. 고객 관리형 정책 예 를 업데이트하여 사용자가 특정 사용자 지정 접미사 서비스 연결 역할만 Amazon EC2 Auto Scaling에 전달할 수 있도록 허용하는 정책 예를 포함합니다.	June 13, 2019

새로운 Amazon EBS 기능 지원 (p. 211)	시작 템플릿 주제에 새로운 Amazon EBS 기능에 대한 지원 내용을 추가했습니다. 스냅샷에서 복원할 때 EBS 볼륨의 암호화 상태를 변경합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Auto Scaling 그룹에 대한 시작 템플릿 생성 을 참조하십시오.	May 13, 2019
안내서 변경 사항 (p. 211)	다음 섹션에서 개선된 Amazon EC2 Auto Scaling 문서: 축소 시 Auto Scaling 인스턴스 종료 제어 , Auto Scaling 그룹 , Auto Scaling 그룹의 여러 인스턴스 유형 및 구매 옵션 , Amazon EC2 Auto Scaling의 동적 조정 .	March 12, 2019
인스턴스 유형 및 구매 옵션 조합 지원 (p. 211)	단일 Auto Scaling 그룹 내에서 구매 옵션(스팟, 온디맨드, 예약 인스턴스) 및 인스턴스 유형 간에 인스턴스를 프로비저닝하고 자동으로 확장합니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Auto Scaling 그룹의 여러 인스턴스 유형 및 구매 옵션 을 참조하십시오.	November 13, 2018
Amazon SQS에 따라 조정에 대한 내용이 업데이트됨 (p. 211)	사용자 지정 지표를 사용하여 Amazon SQS 대기열의 수요 변화에 따라 Auto Scaling 그룹을 조정할 수 있는 방법을 설명하기 위해 설명서를 업데이트했습니다. 자세한 내용은 Amazon EC2 Auto Scaling 사용 설명서의 Amazon SQS에 따른 조정 을 참조하십시오.	July 26, 2018

다음 표에서는 2018년 7월 이전에 Amazon EC2 Auto Scaling 설명서에서 변경된 중요 사항에 대해 설명합니다.

기능	설명	릴리스 날짜
대상 추적 조정 정책을 지원	몇 가지 간단한 단계를 통해 애플리케이션용 동적 조정을 설정할 수 있습니다. 자세한 내용은 Amazon EC2 Auto Scaling의 대상 추적 조정 정책 을 참조하십시오.	2017년 7월 12일
리소스 수준 권한에 대한 지원	리소스 수준에서 액세스를 제어하려면 IAM 정책을 만듭니다. 자세한 내용은 Amazon EC2 Auto Scaling 리소스에 대한 액세스 제어 를 참조하십시오.	2017년 5월 15일
모니터링 개선	Auto Scaling 그룹 지표는 더 이상 세부 모니터링 활성화를 요구하지 않습니다. 이제 콘솔의 [Monitoring] 탭에서 그룹 지표 수집을 활성화하고 지표 그래프를 볼 수 있습니다. 자세한 내용은 Amazon CloudWatch를 사용하여 Auto Scaling 그룹과 인스턴스 모니터링 을 참조하십시오.	2016년 8월 18일

기능	설명	릴리스 날짜
Application Load Balancers 지원	새 그룹 또는 기존 Auto Scaling 그룹에 하나 이상의 대상 그룹을 연결합니다. 자세한 내용은 Auto Scaling 그룹에 로드 밸런서 연결 을 참조하십시오.	2016년 8월 11일
수명 주기 후크 이벤트	Amazon EC2 Auto Scaling은 수명 주기 후크를 실행할 때 CloudWatch 이벤트에 이벤트를 보냅니다. 자세한 내용은 Auto Scaling 그룹 조정 시 CloudWatch 이벤트 확인 을 참조하십시오.	2016년 2월 24일
인스턴스 보호	축소 시 Amazon EC2 Auto Scaling이 특정 종료 인스턴스를 선택하지 않도록 합니다. 자세한 내용은 인스턴스 보호 를 참조하십시오.	2015년 12월 07일
단계 조정 정책	경보 위반의 크기에 따라 조정할 수 있는 조정 정책을 생성합니다. 자세한 내용은 조정 정책 유형 을 참조하십시오.	2015년 7월 06일
로드 밸런서 업데이트	기존 Auto Scaling 그룹에 로드 밸런서를 연결하거나 로드 밸런서에서 분리합니다. 자세한 내용은 Auto Scaling 그룹에 로드 밸런서 연결 을 참조하십시오.	2015년 6월 11일
ClassicLink 지원	Auto Scaling 그룹의 EC2-Classic 인스턴스를 VPC에 연결하고, 연결된 EC2-Classic 인스턴스와 VPC의 인스턴스 간에 프라이빗 IP 주소로 서로 통신합니다. 자세한 내용은 VPC에 EC2-Classic 인스턴스 연결 을 참조하십시오.	2015년 1월 19일
수명 주기 후크	새로 시작된 인스턴스나 종료되는 인스턴스에 대해 작업을 수행하는 동안 보류 상태로 유지합니다. 자세한 내용은 Amazon EC2 Auto Scaling 수명 주기 후크 를 참조하십시오.	2014년 7월 30일
인스턴스 분리	Auto Scaling 그룹에서 인스턴스를 분리합니다. 자세한 내용은 Auto Scaling 그룹에서 EC2 인스턴스 분리 를 참조하십시오.	2014년 7월 30일
인스턴스를 대기 상태로 설정	InService 상태인 인스턴스를 standby 상태로 변경합니다. 자세한 내용은 Auto Scaling 그룹에서 인스턴스 임시 제거 를 참조하십시오.	2014년 7월 30일
태그 관리	AWS Management 콘솔을 사용하여 Auto Scaling 그룹을 관리합니다. 자세한 내용은 그룹 및 인스턴스에 태그 지정 을 참조하십시오.	2014년 5월 01일
전용 인스턴스 지원	시작 구성을 생성할 때 배치 테넌시 속성을 지정하여 전용 인스턴스를 시작합니다. 자세한 내용은 인스턴스 배치 테넌시 를 참조하십시오.	2014년 4월 23일
EC2 인스턴스에서 그룹 또는 시작 구성 생성	EC2 인스턴스를 사용하여 새 Auto Scaling 그룹 또는 새 시작 구성을 생성합니다. EC2 인스턴스를 사용하여 시작 구성을 생성하는 방법에 대한 자세한 내용은 EC2 인스턴스를 사용하여 시작 구성 만들기를 참조하십시오 . EC2 인스턴스를 사용하여 Auto Scaling 그룹을 만드는 방법에 대한 자세한 내용은 EC2 인스턴스를 사용하여 Auto Scaling 그룹 만들기를 참조하십시오 .	2014년 1월 02일
인스턴스 연결	기존 Auto Scaling 그룹에 인스턴스를 연결하여 EC2 인스턴스에 대한 자동 조정을 활성화합니다. 자세한 내용은 Auto Scaling 그룹에 EC2 인스턴스 연결 을 참조하십시오.	2014년 1월 02일

기능	설명	릴리스 날짜
계정 제한 보기	계정의 Auto Scaling 리소스에 대한 제한을 봅니다. 자세한 내용은 제한 을 참조하십시오.	2014년 1월 02일
Amazon EC2 Auto Scaling 콘솔 지원	AWS Management 콘솔을 사용하여 Amazon EC2 Auto Scaling에 액세스합니다. 자세한 내용은 Amazon EC2 Auto Scaling 시작하기 단원을 참조하십시오.	2013년 12월 10일
퍼블릭 IP 주소 배정	VPC로 시작된 인스턴스에 퍼블릭 IP 주소를 배정합니다. 자세한 내용은 VPC에서 Auto Scaling 인스턴스 시작 을 참조하십시오.	2013년 9월 19일
인스턴스 종료 정책	EC2 인스턴스를 종료할 때 사용할 Amazon EC2 Auto Scaling 인스턴스 종료 정책을 지정합니다. 자세한 내용은 축소 조정 중 Auto Scaling 인스턴스 종료 제어 를 참조하십시오.	2012년 9월 17일
IAM 역할 지원	IAM 인스턴스 프로파일로 EC 인스턴스를 시작합니다. 이 기능을 사용하여 IAM 역할을 인스턴스에 할당하여 애플리케이션에서 기타 AWS 서비스에 안전하게 액세스할 수 있습니다. 자세한 내용은 IAM 역할을 사용하여 Auto Scaling 인스턴스 시작 을 참조하십시오.	2012년 6월 11일
스팟 인스턴스 지원	시작 구성에서 스팟 인스턴스 입찰 가격을 지정하여 Auto Scaling 그룹에서 스팟 인스턴스를 실행합니다. 자세한 내용은 Auto Scaling 그룹에서 스팟 인스턴스 시작 을 참조하십시오.	2012년 6월 7일
그룹 및 인스턴스 태그 지정	Auto Scaling 그룹의 태그를 지정하고 태그가 생성된 후 시작된 EC2 인스턴스에도 이 태그가 적용되도록 지정합니다. 자세한 내용은 그룹 및 인스턴스에 태그 지정 을 참조하십시오.	2012년 1월 26일
Amazon SNS 지원	<p>Amazon SNS를 사용하여 Amazon EC2 Auto Scaling가 시작 또는 EC2 인스턴스 종료 시 알림을 수신할 수 있습니다. 자세한 내용은 Auto Scaling 그룹 조정 시 SNS 알림 받기를 참조하십시오.</p> <p>Amazon EC2 Auto Scaling에는 또한 다음의 새 기능이 추가되었습니다.</p> <ul style="list-style-type: none"> Cron 구문을 사용하여 조정 활동을 설정하는 기능. 자세한 정보는 PutScheduledUpdateGroupAction API 명령을 참조하십시오. 로드 밸런서(LoadBalancer)에 시작한 인스턴스를 추가하지 않고 확장할 수 있도록 하는 새로운 구성 설정. 자세한 정보는 ProcessType API 데이터 형식을 참조하십시오. 인스턴스가 먼저 종료될 때까지 기다리지 않고 Auto Scaling 그룹과 여기에 연결된 인스턴스를 삭제하도록 Amazon EC2 Auto Scaling에 지시하는 <code>DeleteAutoScalingGroup</code> 작업의 <code>ForceDelete</code> 플래그. 자세한 정보는 DeleteAutoScalingGroup API 작업을 참조하십시오. 	2011년 7월 20일
예약된 조정 작업	예약된 조정 작업에 대한 지원을 추가했습니다. 자세한 내용은 Amazon EC2 Auto Scaling의 예약된 확장 을 참조하십시오.	2010년 2월 12일
Amazon VPC 지원	Amazon VPC에 대한 지원이 추가되었습니다. 자세한 내용은 VPC에서 Auto Scaling 인스턴스 시작 을 참조하십시오.	2010년 2월 12일

기능	설명	릴리스 날짜
HPC 클러스터 지원	HPC(고성능 컴퓨팅) 클러스터에 대한 지원이 추가되었습니다.	2010년 2월 12일
상태 확인 지원	Amazon EC2 Auto Scaling 관리형 EC2 인스턴스에 Elastic Load Balancing 상태 확인 사용에 대한 지원을 추가했습니다. 자세한 내용은 Auto Scaling을 통해 ELB 상태 확인 사용 을 참조하십시오.	2010년 2월 12일
CloudWatch 경보 지원	기존 트리거 메커니즘이 제거되고 CloudWatch 경보 기능을 사용하도록 Amazon EC2 Auto Scaling을 재설계했습니다. 자세한 내용은 Amazon EC2 Auto Scaling에 대한 동적 조정 을 참조하십시오.	2010년 2월 12일
조정 일시 중지 및 재개	조정 프로세스의 일시 중지 및 재개에 대한 지원이 추가되었습니다.	2010년 2월 12일
IAM 지원	IAM에 대한 지원이 추가되었습니다. 자세한 내용은 Amazon EC2 Auto Scaling 리소스에 대한 액세스 제어 를 참조하십시오.	2010년 2월 12일