

AWS

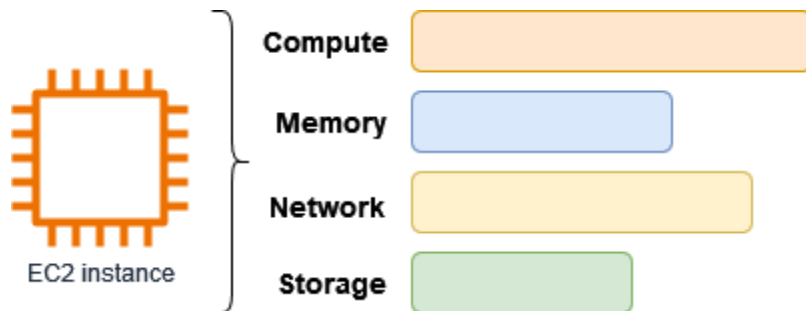
Amazon Web Services(AWS)는 전 세계적으로 분포한 데이터 센터에서 제공하고 있는 클라우드.

EC2 인스턴스

Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스는 AWS 클라우드의 가상 서버.

EC2 인스턴스를 시작할 때 지정하는 인스턴스 유형에 따라 인스턴스에 사용할 수 있는 하드웨어가 결정된다.

인스턴스 유형마다 서로 다른 컴퓨팅, 메모리, 네트워크 및 스토리지 리소스의 조합을 제공.



Amazon EC2의 기능

Amazon EC2는 다음의 대략적인 기능을 제공

- 인스턴스
가상 서버.
- Amazon Machine Images (AMIs)
서버에 필요한 구성 요소(운영 체제와 추가 소프트웨어 포함)를 패키징하는 인스턴스용 사전 구성 템플릿.
- 인스턴스 타입
인스턴스의 다양한 CPU, 메모리, 스토리지, 네트워킹 용량 및 그래픽 하드웨어 구성.
- Amazon EBS 볼륨
Amazon Elastic Block Store(Amazon EBS)를 사용하는 데이터에 대한 영구 스토리지 볼륨.
- 인스턴스 스토어 볼륨
인스턴스를 중단, 최대 절전 모드로 전환 또는 종료할 때 삭제되는 임시 데이터용 스토리지 볼륨.
- 키 페어
인스턴스에 대한 보안 로그인 정보. AWS는 퍼블릭 키를 저장하고 사용자는 프라이빗 키를 안전

한 장소에 저장하면 된다.

- 보안 그룹

인스턴스에 도달할 수 있는 프로토콜, 포트 및 소스 IP 범위와 인스턴스가 연결할 수 있는 대상 IP 범위를 지정할 수 있는 가상 방화벽.

관련 서비스

Amazon EC2와 함께 사용할 서비스

Amazon EC2를 사용하여 배포한 인스턴스와 함께 다른 AWS 서비스를 사용할 수 있다.

- Amazon EC2 Auto Scaling

애플리케이션의 로드를 처리할 수 있는 정확한 수의 Amazon EC2 인스턴스를 유지하는 데 도움.

- AWS Backup

Amazon EC2 인스턴스와 여기에 연결된 Amazon EBS 볼륨의 백업 자동화.

- Amazon CloudWatch

인스턴스와 Amazon EBS 볼륨을 모니터링.

- Elastic Load Balancing

수신되는 애플리케이션 트래픽을 여러 인스턴스로 자동 분산.

- Amazon GuardDuty

EC2 인스턴스에 대한 잠재적 무단 사용 또는 악의적 사용 탐지.

- EC2 Image Builder

안전하고 최신상태 서버 이미지의 생성, 관리 및 배포를 자동화.

- AWS Launch Wizard

개별 AWS 리소스를 수동으로 식별하고 준비할 필요 없이 타사 애플리케이션에 대한 AWS 리소스를 크기 조정, 구성 및 배포.

- AWS Systems Manager

안전한 end to end 관리 솔루션을 사용하여 EC2 인스턴스에서 대규모로 작업 수행 가능.

- 추가 컴퓨팅 서비스

Amazon EC2를 사용하는 대신 다른 AWS 컴퓨팅 서비스를 사용하여 인스턴스를 시작할 수 있다.

- Amazon Lightsail

저렴하고 예측 가능한 월별 요금으로 프로젝트를 신속하게 배포하는 데 필요한 리소스를 제공하는 클라우드 플랫폼인 Amazon Lightsail을 사용하여 웹 사이트 또는 웹 애플리케이션을 구축한다.

- Amazon Elastic Container Service(Amazon ECS)

EC2 인스턴스 클러스터에서 컨테이너화된 애플리케이션을 배포, 관리하고 규모 조정.

- Amazon Elastic Kubernetes Service(Amazon EKS)

AWS에서 Kubernetes 애플리케이션을 실행.

Amazon EC2 액세스

다음 방법으로 Amazon EC2 인스턴스를 생성하고 관리할 수 있다.

- Amazon EC2 콘솔

Amazon EC2 인스턴스 및 리소스를 생성하고 관리하는 간단한 웹 인터페이스.

AWS Management Console에 로그인한 후 콘솔 홈페이지에서 EC2를 선택하여 Amazon EC2에 액세스할 수 있다.

- AWS Command Line Interface

명령줄 쉘의 명령을 사용하여 AWS 서비스와 상호 작용할 수 있다.

Windows, Mac, Linux에서 지원.

- AWS CloudFormation

Amazon EC2는 AWS CloudFormation을 사용한 리소스 생성을 지원.

AWS 리소스를 설명하는 템플릿(JSON 또는 YAML 형식)을 생성하면 AWS CloudFormation에서 이러한 리소스를 준비하고 구성한다.

CloudFormation 템플릿을 재사용하여 동일한 리전 및 계정 또는 여러 리전 및 계정에 동일한 리소스를 여러 번 준비할 수 있다.

- AWS SDK

HTTP나 HTTPS 요청을 직접 보내는 대신, 개발자가 각 언어가 제공하는 고유의 API를 사용하여 애플리케이션을 빌드하도록 라이브러리, 샘플 코드, 자습서 및 기타 리소스를 제공.

이러한 라이브러리는 요청 암호화 서명, 요청 재시도, 오류 응답 처리와 같은 작업을 자동화하는 기본 기능을 제공하므로 더 쉽게 시작할 수 있다.

- AWS Tools for PowerShell

PowerShell용 도구를 사용하면 PowerShell 명령줄에서 AWS 리소스에 대한 작업을 스크립팅할 수 있다.

- Query API

Amazon EC2에서는 쿼리 API를 제공.

이러한 요청은 HTTP나 HTTPS의 메시지 교환 방식인 GET이나 POST이며, 미리 정해진 이름인 "Action"을 쿼리 변수로 사용.

Amazon EC2 요금

- 프리 티어

Amazon EC2를 무료로 시작할 수 있다.

- 온디맨드 인스턴스

장기 약정이나 선결제 없이 초 단위로, 최소 60초 사용한 인스턴스에 대한 요금을 지불하는 방식.

- 절감형 플랜

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

- 예약 인스턴스

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

- 스팟 인스턴스

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

- 전용 호스트

온디맨드 또는 절감형 플랜의 일부로 고객 전용 물리적 EC2 서버를 사용하여 비용을 절감.

기존 서버 기반 소프트웨어 라이선스를 사용하면 규정 준수 요구 사항을 충족하는 데 도움을 받을 수 있다.

- 온디맨드 용량 예약

원하는 기간 동안 특정 가용 영역의 EC2 인스턴스에 대해 용량을 예약.

- 초당 청구

청구서에서 사용하지 않은 분 및 초 단위의 비용을 제거.

- 추정, 결제 및 비용 최적화

AWS 사용 사례에 대한 예상 비용을 계산하려면 AWS Pricing Calculator를 사용.

Amazon EC2 시작하기

Amazon EC2를 사용하여 인스턴스에서 실행되는 운영 체제와 애플리케이션을 설정하고 구성할 수 있다.

이미지 - 인스턴스에서 실행할 소프트웨어(예: 운영 체제)가 포함된 템플릿.

키 페어 - 인스턴스에 연결할 때 ID를 증명하는 데 사용하는 보안 자격 증명 세트.
퍼블릭 키는 인스턴스에 있고 프라이빗 키는 컴퓨터에 있다.

네트워크 - Virtual Private Cloud(VPC)는 AWS 계정 전용 가상 네트워크다.
빠르게 시작할 수 있도록 계정에는 각 AWS 리전에 기본 VPC가 제공되며 각 기본 VPC에는 각 가용 영역에 기본 서브넷이 있다.

보안 그룹 - 인바운드 및 아웃바운드 트래픽을 제어하는 가상 방화벽 역할을 한다.

EBS 볼륨 - 이미지의 루트 볼륨이 필요하고, 필요에 따라 데이터 볼륨을 추가할 수 있다.

EC2는 보안 그룹, 키 페어, EBS 루트 볼륨이 있는 인스턴스.

- 1단계: 인스턴스 시작

<https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 연다.

화면 상단의 탐색 모음에 현재 AWS 리전(예: 오하이오주)이 표시된다. 선택된 리전을 사용하거나 선택적으로 가까운 리전을 선택할 수 있다.

EC2 콘솔 대시보드의 시작 인스턴스 창에서 인스턴스 시작 선택.

이름 및 태그(Name and tags) 아래의 이름(Name)에 인스턴스를 설명하는 이름을 입력.

애플리케이션 및 OS 이미지(Amazon Machine Image)(Application and OS Images (Amazon Machine Image))에서 다음을 수행

이름 및 태그 정보

이름


myweb1

추가 태그 추가


▼ 애플리케이션 및 OS 이미지(Amazon Machine Image) 정보

AMI는 인스턴스를 시작하는 데 필요한 소프트웨어 구성(운영 체제, 애플리케이션 서버 및 애플리케이션)이 포함된 템플릿입니다. 아래에서 찾고 있는 항목이 보이지 않으면 AMI를 검색하거나 찾아보십시오.


Quick Start




Amazon Linux




macOS




Ubuntu



Windows



Red Hat



더 많은 AMI 찾아보기

AWS, Marketplace 및 커뮤니티의 AMI 포함

Amazon Machine Image(AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

ami-074e6efaf93d85776 (64비트 (x86)) / ami-08fdd91d87f63bb09 (64비트 (Arm))

프리 티어 사용 가능

빠른 시작을 선택한 다음 인스턴스의 운영 체제(OS)를 선택. 첫 번째 Linux 인스턴스의 경우 Amazon Linux를 선택하는 것이 좋다.

Amazon Machine Image(AMI)에서 프리 티어 사용 가능으로 표시된 AMI를 선택한다.

인스턴스 유형에서 인스턴스 유형으로 프리 티어에 적합한 t2.micro를 선택.
t2.micro를 사용할 수 없는 리전에서는 t3.micro가 프리 티어에 적합하다.

키 페어(로그인)의 키 페어 이름에서 기존 키 페어를 선택하거나 새 키 페어 생성을 선택하여 첫 번째 키 페어를 생성.

▼ 인스턴스 유형 정보

인스턴스 유형

t2.micro

프리 티어 사용 가능

패밀리: t2 1 vCPU 1 GiB 메모리 현재 세대: true

온디맨드 Linux 요금: 0.0116 USD 시간당 온디맨드 SUSE 요금: 0.0116 USD 시간당

온디맨드 Windows 요금: 0.0162 USD 시간당 온디맨드 RHEL 요금: 0.0716 USD 시간당

☐ 모든 세대

[인스턴스 유형 비교](#)

▼ 키 페어(로그인) 정보

키 페어를 사용하여 인스턴스에 안전하게 연결할 수 있습니다. 인스턴스를 시작하기 전에 선택한 키 페어에 대한 액세스 권한이 있는지 확인하세요.

키 페어 이름 - 필수

myweb1key

 새 키 페어 생성

키 페어 생성



키 페어 이름

키 페어를 사용하면 인스턴스에 안전하게 연결할 수 있습니다.

myweb1key1

이름에는 최대 255개의 ASCII 문자를 포함할 수 있습니다. 앞 또는 뒤에 공백을 포함할 수 없습니다.

키 페어 유형

☒ RSA

RSA 암호화된 프라이빗 및 퍼블릭 키 페어

☐ ED25519

ED25519 encrypted private and public key pair


프라이빗 키 파일 형식

☒ .pem

OpenSSH와 함께 사용

☐ .ppk

PuTTY와 함께 사용

 When prompted, store the private key in a secure and accessible location on your computer. 나중에 인스턴스에 연결할 때 필요합니다. [자세히 알아보기](#)

취소

키 페어 생성

주의

키 페어 없이 진행(권장되지 않음)을 선택하면 이 자습서에 설명된 방법을 사용하여 인스턴스에 연결할 수 없다.

네트워크 설정에서 기본 VPC를 선택하고, 선택한 가용 영역에서 기본 서브넷을 사용하는 옵션을 선택하고, 어디서나 인스턴스에 연결할 수 있는 규칙으로 보안 그룹을 구성한 것을 확인한다. 첫 번째 인스턴스의 경우에는 기본 설정을 사용하는 것이 좋다.

▼ 네트워크 설정 정보

[편집](#)

네트워크 정보

vpc-02c8a870ed7bba6ab

서브넷 정보

기본 설정 없음(가용 영역의 기본 서브넷)

퍼블릭 IP 자동 할당 정보

활성화

방화벽(보안 그룹) 정보

보안 그룹은 인스턴스에 대한 트래픽을 제어하는 방화벽 규칙 세트입니다. 특정 트래픽이 인스턴스에 도달하도록 허용하는 규칙을 추가합니다.

☒ 보안 그룹 생성☐ 기존 보안 그룹 선택

다음 규칙을 사용하여 'launch-wizard-1'(이)라는 새 보안 그룹을 생성합니다.

☒ 에서 SSH 트래픽 허용

인스턴스 연결에 도움

위치 무관



0.0.0.0/0

☐ 인터넷에서 HTTPS 트래픽 허용

예를 들어 웹 서버를 생성할 때 엔드포인트를 설정하려면

☐ 인터넷에서 HTTP 트래픽 허용

예를 들어 웹 서버를 생성할 때 엔드포인트를 설정하려면

 소스가 0.0.0.0/0인 규칙은 모든 IP 주소에서 인스턴스에 액세스하도록 허용합니다. 알려진 IP 주소의 액세스만 허용하도록 보안 그룹을 설정하는 것이 좋습니다. 

스토리지 구성에서 루트 볼륨은 구성했지만 데이터 볼륨은 구성하지 않은 것을 확인할 수 있다. 테스트 목적으로는 이 정도면 충분하다.

▼ 스토리지 구성 정보

어드밴스드

1x GiB 루트 볼륨 (암호화되지 않음)

i 프리 티어를 사용할 수 있는 고객은 최대 30GB의 EBS 범용(SSD)또는 마그네틱 스토리지를 사용할 수 있습니다. **X**

새 볼륨 추가

선택한 AMI에 인스턴스가 허용하는 것보다 많은 인스턴스 스토어 볼륨이 포함되어 있습니다. AMI에서 처음 0개의 인스턴스 스토어 볼륨에만 액세스할 수 있습니다.

0 x 파일 시스템

편집

요약 패널에서 인스턴스 구성 요약을 검토하고 준비가 되면 인스턴스 시작을 선택한다.

인스턴스에 연결 정보

다음 옵션 중 하나를 사용하여 인스턴스 i-0ab80e82d3798ae4c (myweb1)에 연결

EC2 인스턴스 연결

Session Manager

SSH 클라이언트

EC2 직렬 콘솔

인스턴스 ID

i-0ab80e82d3798ae4c (myweb1)

연결 유형

☒ EC2 Instance Connect를 사용하여 연결
퍼블릭 IPv4 주소가 있는 EC2 인스턴스 연결 브라우저 기반 클라이언트를 사용하여 연결합니다.

☐ EC2 인스턴스 연결 엔드포인트를 사용하여 연결
프라이빗 IPv4 주소 및 VPC 엔드포인트가 있는 EC2 인스턴스 연결 브라우저 기반 클라이언트를 사용하여 연결합니다.

퍼블릭 IP 주소

3.142.133.129

사용자 이름

인스턴스를 시작하는 데 사용되는 AMI에 정의된 사용자 이름을 입력합니다. 사용자 지정 사용자 이름을 정의하지 않은 경우 기본 사용자 이름인 ubuntu를(를) 사용합니다.

i 참고: 대부분의 경우 기본 사용자 이름 ubuntu은(는) 정확합니다. 하지만 AMI 사용 지침을 읽고 AMI 소유자가 기본 AMI 사용자 이름을 변경했는지 확인하십시오.

취소

연결

실행이 성공하면 성공 알림에서 인스턴스의 ID를 선택하여 인스턴스 페이지를 열고 시작 상태를 모니터링한다.

인스턴스에 대한 확인란을 선택.

초기 인스턴스 상태는 pending.

인스턴스가 시작되면 상태가 running로 변경된다.

상태 및 경보 탭을 선택. 인스턴스가 상태 검사를 통과하면 연결 요청을 받을 준비가 된 것이다.

- 2단계: 인스턴스에 연결

사용하는 절차는 인스턴스의 운영 체제에 따라 달라진다.

Linux 인스턴스

모든 SSH 클라이언트를 사용하여 Linux 인스턴스에 연결할 수 있다. 컴퓨터에서 Windows를 실행 중인 경우 터미널을 열고 ssh 명령을 실행하여 SSH 클라이언트가 설치되어 있는지 확인. 명령을 찾을 수 없으면 Windows용 OpenSSH를 설치.

SSH를 사용하여 인스턴스에 연결하려면

<https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 연다.

탐색 창에서 Instances(인스턴스)를 선택.

찾은 인스턴스를 선택한 다음 [Connect]를 선택.

인스턴스에 연결 페이지에서 SSH 클라이언트 탭을 선택.

(선택 사항) 인스턴스를 시작할 때 키 페어를 만들고 Linux 또는 macOS를 실행하는 컴퓨터에 프라이빗 키(.pem 파일)를 다운로드한 경우 예제 chmod 명령을 실행하여 프라이빗 키에 대한 권한을 설정한다.

key-pair-name.pem은 프라이빗 키 파일의 이름,

ec2-user는 이미지와 연결된 사용자 이름,

@ 기호 뒤의 문자열은 인스턴스의 퍼블릭 DNS 이름

⇒ `ssh -i key-pair-name.pem ec2-user@ec2-198-51-100-1.us-east-2.compute.amazonaws.com`

컴퓨터의 터미널 창에서 이전 단계에서 저장한 ssh 명령을 실행한다.

현재 디렉터리에 프라이빗 키 파일이 없는 경우 이 명령에 키 파일의 정규화된 경로를 지정해야 한다.

다음은 응답의 예다.

The authenticity of host 'ec2-198-51-100-1.us-east-2.compute.amazonaws.com (198-51-100-1)' can't be established.

ECDSA key fingerprint is l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.

Are you sure you want to continue connecting (yes/no)?

(선택 사항) 보안 알림의 지문이 인스턴스를 처음 시작할 때 콘솔 출력에 포함된 인스턴스 지문과 일치하는지 확인. 콘솔 출력을 가져오려면 작업, 모니터링 및 문제 해결, 시스템 로그 가져오기를 선택. 이들 지문이 일치하지 않으면 누군가가 메시지 가로채기(man-in-the-middle) 공격을 시도하고 있는 것일 수 있다. 이들 지문이 일치하면 다음 단계를 계속 진행한다.

yes를 입력.

다음은 응답의 예다.

Warning: Permanently added 'ec2-198-51-100-1.us-east-2.compute.amazonaws.com' (ECDSA) to the list of known hosts.

- 3단계: 인스턴스 정리

생성한 인스턴스와 볼륨을 완료한 후에는 인스턴스를 종료하여 정리해야 한다.

중요

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

해당 인스턴스 또는 프리 티어 한도에 포함되는 사용량에 대한 요금 발생은 인스턴스 상태가 shutting down 또는 terminated로 변경되는 즉시 발생이 중지된다. 나중에 사용하기 위해 인스턴스를 유지하되 요금 또는 프리 티어 한도에 포함되는 사용량이 발생하지 않도록 하려면 지금 인스턴스를 중지한 다음 나중에 다시 시작하면 된다.

인스턴스를 종료하려면

탐색 창에서 Instances(인스턴스)를 선택 => 인스턴스 목록에서 종료할 인스턴스를 선택.

인스턴스 상태에서 인스턴스 종료를 선택.

확인 메시지가 나타나면 종료를 선택.

Amazon EC2가 인스턴스를 종료한다. 인스턴스는 종료한 후에도 잠시 동안 콘솔에서 표시된 상태로 유지되며, 그 이후 항목이 자동으로 삭제된다. 종료된 인스턴스는 콘솔 디스플레이에서 직접

제거할 수 없다.

Amazon RDS

Amazon Relational Database Service(Amazon RDS)는 AWS 클라우드에서 관계형 데이터베이스를 더 쉽게 설치, 운영 및 확장할 수 있는 웹 서비스.

- Amazon RDS의 장점

Amazon RDS는 관리형 데이터베이스 서비스로, 대부분의 관리 작업을 담당

Amazon RDS를 사용하면 번거로운 수동 프로세스를 처리할 필요가 없어 애플리케이션과 사용자에게 집중할 수 있다.

Amazon RDS는 완전관리형이 아닌 데이터베이스 배포와 비교해서 다음과 같은 주요 이점을 제공합니다.

이미 익숙하게 사용하는 IBM Db2, MariaDB, Microsoft SQL Server, MySQL, Oracle Database, PostgreSQL 등의 데이터베이스 엔진을 이용할 수 있다.

Amazon RDS는 백업, 소프트웨어 패치, 자동 장애 감지 및 복구를 관리한다.

자동화된 백업을 설정하거나 고유한 백업 스냅샷을 수동으로 생성할 수 있고 이러한 백업을 사용하여 데이터베이스를 복원할 수 있다. Amazon RDS 복원 프로세스는 안정적이고 효율적이다.

기본 DB 인스턴스 및 문제 발생 시 장애 조치를 수행할 수 있는 동기식 보조 DB 인스턴스에서 가용성을 높일 수 있다. 읽기 전용 복제본을 사용하여 읽기 스케일을 높일 수도 있다.

데이터베이스 패키지의 보안 외에도 AWS Identity and Access Management(IAM)을 사용하여 사용자 및 권한을 정의하도록 액세스를 제어할 수 있다. 데이터베이스를 Virtual Private Cloud(VPC)에 배치하여 데이터베이스를 보호할 수도 있다.

- Amazon EC2 및 온프레미스(on-premises) 배포와의 책임 비교

대부분의 관계형 데이터베이스 배포에 대해 Amazon RDS를 기본적으로 선택하는 것이 좋다. 다음과 같은 대안을 사용하면 소프트웨어와 하드웨어를 관리하는 데 더 많은 시간이 들 수 있다는 단점이 있다.

온프레미스 배포

온프레미스 서버를 구매하면 CPU, 메모리, 스토리지 및 IOPS가 함께 번들로 제공. 사용자가 서버, 운영 체제 및 데이터베이스 소프트웨어를 모두 관리해야 한다.

Amazon EC2

Amazon Elastic Compute Cloud(Amazon EC2)는 AWS 클라우드에서 확장 가능한 컴퓨팅 용량을 제공. 온프레미스 서버와 달리 CPU, 메모리, 스토리지, IOPS가 분리되어 독립적으로 규모를 조정할 수 있다. AWS에서는 하드웨어 계층을 관리하여 온프레미스 데이터베이스 서버를 관리해야 하는 부담을 일정 부분 덜어준다.

Amazon EC2에서 데이터베이스를 실행하면 사용자 오류가 발생하기 쉽다는 단점이 있다. 예를 들어, 운영 체제 또는 데이터베이스 소프트웨어를 수동으로 업데이트하면 원치 않는 애플리케이션 다운타임을 초래할 수 있다. 모든 변경 사항을 확인하여 문제를 파악하고 수정하려면 몇 시간이 걸리기도 한다.

다음 표에서는 온프레미스 데이터베이스, Amazon EC2 및 Amazon RDS의 관리 모델을 비교한다.

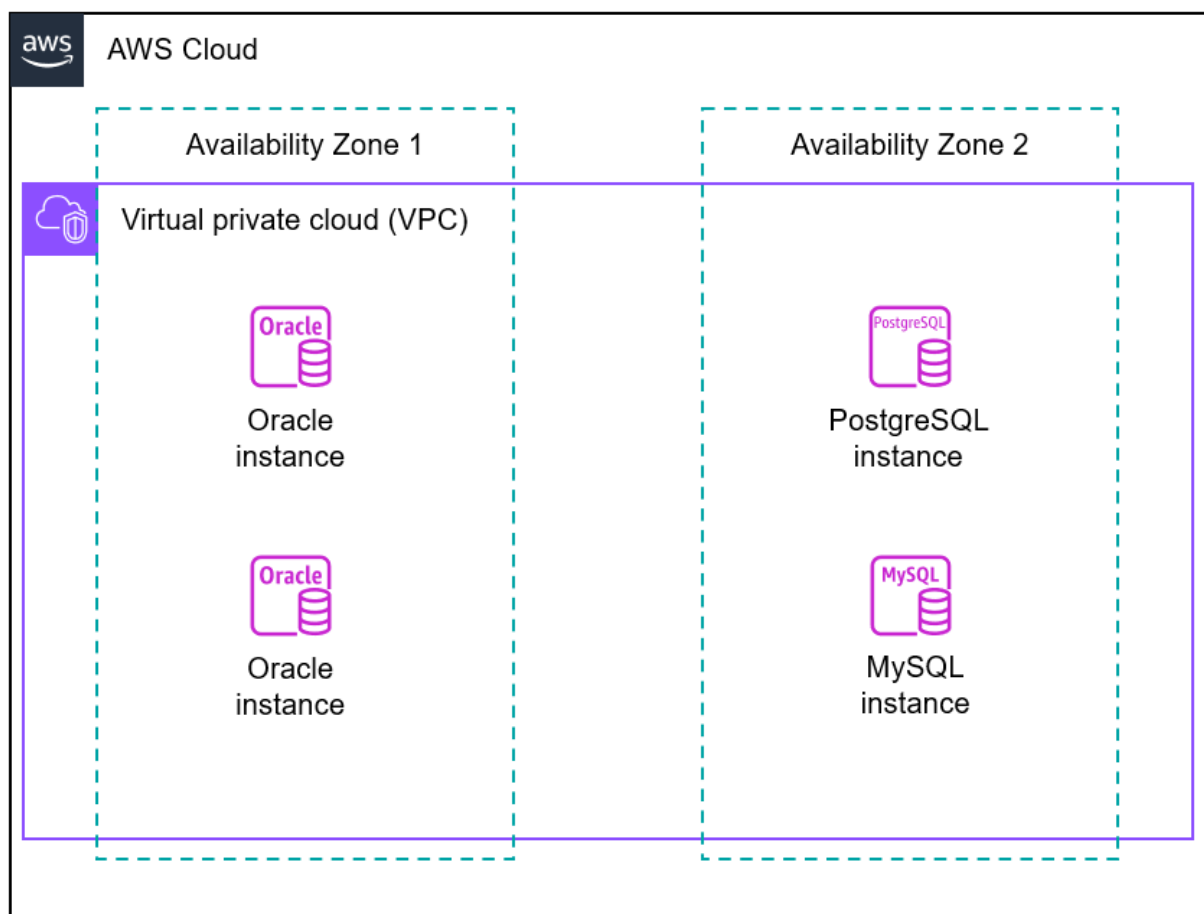
기능	온프레미스 관리	Amazon EC2 관리	Amazon RDS 관리
애플리케이션 최적화	고객	고객	고객
확장성	고객	고객	AWS
높은 가용성	고객	고객	AWS
데이터베이스 백업	고객	고객	AWS
데이터베이스 소프트웨어 패치	고객	고객	AWS
데이터베이스 소프트웨어 설치	고객	고객	AWS
운영 체제(OS) 패치	고객	고객	AWS
OS 설치	고객	고객	AWS
서버 유지 관리	고객	AWS	AWS
하드웨어 수명	고객	AWS	AWS
전력, 네트워크 및 냉각	고객	AWS	AWS

Amazon RDS 공동 책임 모델

Amazon RDS는 DB 인스턴스 및 DB 클러스터의 소프트웨어 구성 요소와 인프라 호스팅을 담당한다. 사용자는 성능을 개선하기 위해 SQL 쿼리를 조정하는 프로세스인 쿼리 튜닝을 할 수 있다. 쿼리 성능은 데이터베이스 디자인, 데이터 크기, 데이터 배포, 애플리케이션 워크로드 및 쿼리 패턴에 따라 크게 달라질 수 있다. 모니터링 및 튜닝은 RDS 데이터베이스에 대해 사용자가 소유하는 매우 개별화된 프로세스다. Amazon RDS 성능 개선 도우미를 비롯한 도구를 사용하여 문제가 있는 쿼리를 식별할 수 있다.

- Amazon RDS DB 인스턴스

DB 인스턴스는 AWS 클라우드에 있는 격리된 데이터베이스 환경이다. Amazon RDS의 기본 요소는 DB 인스턴스다. DB 인스턴스에 사용자가 만든 데이터베이스가 하나 이상 포함될 수 있다. 다음 다이어그램은 2개의 가용 영역을 포함하고 각 가용 영역에 2개의 DB 인스턴스가 포함되어 있는 Virtual Private Cloud(VPC)를 보여준다.

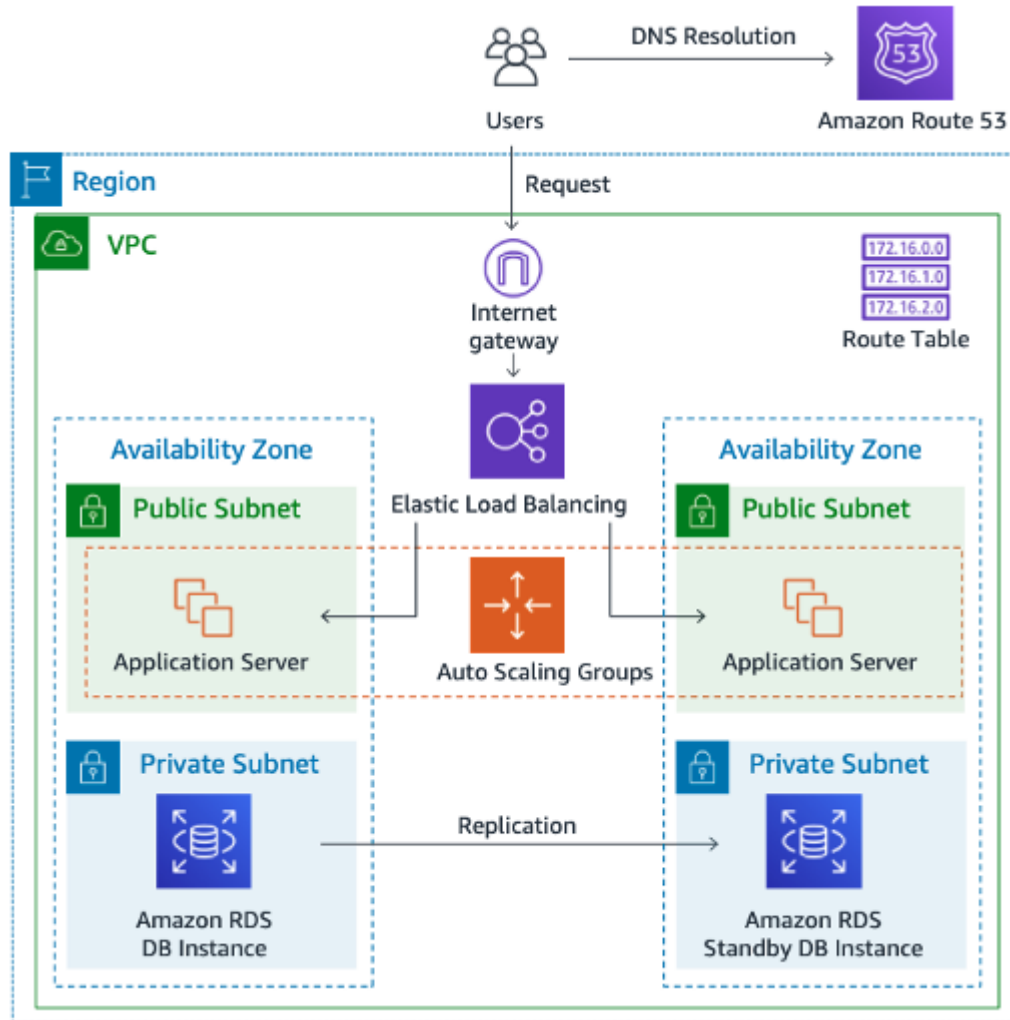


독립 실행형 데이터베이스 인스턴스와 함께 사용하는 것과 동일한 도구 및 애플리케이션을 사용하여 DB 인스턴스에 액세스할 수 있다. AWS Command Line Interface(AWS CLI), Amazon RDS API

또는 AWS Management Console을 사용하여 DB 인스턴스를 생성하고 수정할 수 있다.

Amazon RDS 애플리케이션 아키텍처: 사례

다음 이미지는 데이터베이스 스토리지로 Amazon RDS DB 인스턴스를 사용하는 동적 웹 사이트의 일반적인 사용 사례를 보여준다.



아키텍처의 주요 구성 요소

Elastic Load Balancing

AWS는 Elastic Load Balancing을 통해 사용자 트래픽을 라우팅한다. 로드 밸런서는 워크로드를 가상 서버와 같은 다수의 컴퓨팅 리소스로 분산. 위 사례에서 Elastic Load Balancer는 클라이언트 요청을 애플리케이션 서버로 전달한다.

애플리케이션 서버

애플리케이션 서버는 RDS DB 인스턴스와 상호 작용합니다. AWS의 애플리케이션 서버는 일반적으로 확장 가능한 컴퓨팅 용량을 제공하는 EC2 인스턴스에 호스팅된다. 애플리케이션 서버는 동

일한 Virtual Private Cloud(VPC) 내에서 서로 다른 가용 영역을 사용하는 퍼블릭 서브넷에 있다.

RDS DB 인스턴스

EC2 애플리케이션 서버는 RDS DB 인스턴스와 상호 작용한다. DB 인스턴스는 동일한 Virtual Private Cloud(VPC) 내 서로 다른 가용 영역의 프라이빗 서브넷에 있다. 서브넷은 프라이빗이므로, 인터넷에서의 요청은 허용되지 않는다.

기본 DB 인스턴스는 ~~읽기 복제본~~이라고 하는 다른 DB 인스턴스로 복제된다. 두 DB 인스턴스는 모두 VPC 내의 프라이빗 서브넷에 있으므로 인터넷 사용자가 직접 액세스할 수 없다.

DB 엔진

DB 엔진은 DB 인스턴스에서 실행되는 특정 관계형 데이터베이스 소프트웨어다. Amazon RDS는 다음 데이터베이스 엔진을 지원한다.

- IBM Db2
- MariaDB
- Microsoft SQL Server
- MySQL
- Oracle Database
- PostgreSQL

각 DB 엔진에는 지원되는 고유한 기능이 있으며, DB 엔진의 각 버전에는 특정 기능이 포함될 수 있다. Amazon RDS 기능에 대한 지원은 AWS 리전 및 각 DB 엔진의 특정 버전에 따라 다르다. 또한 DB 엔진마다 관리하는 데이터베이스의 동작을 제어하는 DB 파라미터 그룹에 파라미터 집합이 있다.

DB 인스턴스 클래스

DB 인스턴스 클래스는 DB 인스턴스의 컴퓨팅 및 메모리 용량을 결정하며, DB 인스턴스 클래스는 DB 인스턴스 클래스 유형과 크기로 구성된다. Amazon RDS는 다음 인스턴스 클래스 유형을 지원한다. 여기서 별표(*)는 세대, 선택적 속성 및 크기를 나타낸다.

- 범용 – db.m*
- 메모리 최적화 – db.z*, db.x*, db.r*
- 컴퓨팅 최적화 – db.c*
- 성능 버스트 가능 – db.t*

인스턴스 클래스마다 서로 다른 컴퓨팅, 메모리 및 스토리지 기능을 제공한다. 예를 들어, db.m7g는 AWS Graviton3 프로세서로 구동되는 7세대 범용 DB 인스턴스 클래스 유형이다. DB 인스턴스를 생성할 때 db.m7g.2xlarge와 같은 DB 인스턴스 클래스를 지정하게 된다. 여기서 2xlarge는 크기를 의미한다. 그래서 요구 사항에 가장 적합한 DB 인스턴스 클래스를 선택할 수 있다. 시간이

지나면서 요구 사항이 바뀌면 DB 인스턴스 클래스를 변경할 수 있다. 예를 들어, db.m7g.2xlarge 인스턴스 규모를 db.m7g.4xlarge로 조정할 수 있다.

DB 인스턴스 스토리지

Amazon EBS는 내구성이 있는 블록 수준 스토리지 볼륨을 제공하여 실행 중인 인스턴스에 연결하는 것이 가능하다. DB 인스턴스 스토리지는 다음과 같은 유형으로 제공된다.

- 범용(SSD)

비교적 비용이 효율적인 스토리지 유형으로 중간 크기 DB 인스턴스에서 실행하는 광범위한 워크로드에 이상적이다. 범용 스토리지는 개발 및 테스트 환경에 가장 적합하다.

- 프로비저닝된 IOPS(PIOPS)

이 스토리지 유형은 낮은 I/O 지연 시간과 일관된 I/O 처리량을 요구하는 I/O 집약적인 워크로드, 특히 데이터베이스 워크로드의 요구를 충족하도록 설계되었다. 프로비저닝된 IOPS 스토리지는 프로덕션 환경에 가장 적합하다.

- Magnetic

Amazon RDS는 이전 버전과의 호환성을 위해 마그네틱 스토리지를 지원한다. 새 스토리지가 필요할 경우 범용 SSD 또는 프로비저닝된 IOPS SSD를 사용하는 것이 좋다.

스토리지 유형은 성능 특성과 가격이 다르다. 데이터베이스 요건에 따라 스토리지 성능과 비용을 조정할 수 있다.

DB 인스턴스는 각각 스토리지 유형과 지원하는 데이터베이스 엔진에 따라 최소/최대 스토리지 요구 사항이 있다. 충분한 스토리지를 보유하여 데이터베이스를 확장할 수 있는 여유를 확보하는 것이 중요하다. 또한 스토리지가 충분하면 콘텐츠를 작성하거나 항목을 기록할 수 있는 기능을 DB 엔진에 도입할 수 있다.

Amazon Virtual Private Cloud(VPC)의 DB 인스턴스

Amazon Virtual Private Cloud(Amazon VPC) 서비스를 사용해 가상 사설 클라우드(VPC)에서 DB 인스턴스를 실행할 수 있다. VPC를 사용하면 가상 네트워킹 환경을 완벽하게 제어할 수 있다. 자기만의 IP 주소 범위를 선택하고, 서브넷을 생성하고, 라우팅 및 액세스 제어 목록을 구성할 수 있다.

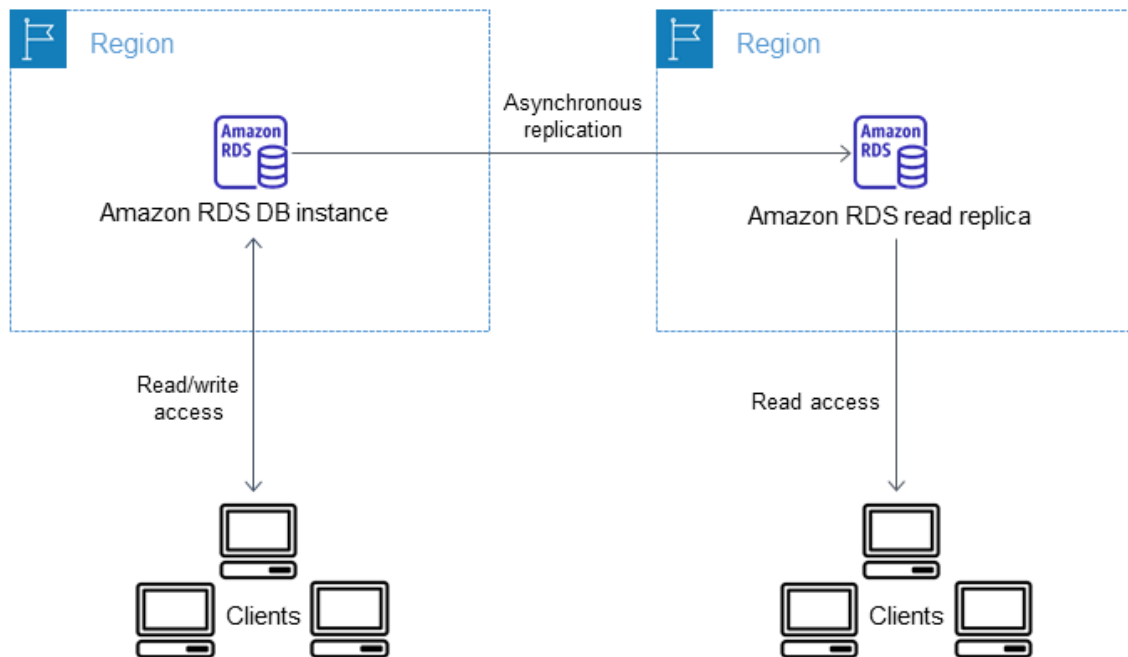
Amazon RDS의 기본 기능은 VPC 실행 중인지 여부에 관계없이 동일하다. Amazon RDS는 백업, 소프트웨어 패치, 자동 장애 감지 및 복구를 관리한다. VPC에서 DB 인스턴스를 실행하는 데는 추가 비용이 들지 않는다.

Amazon RDS는 DB 인스턴스를 기준으로 시간을 동기화하는데 NTP(Network Time Protocol)를 사용한다.

AWS 리전 및 가용 영역

Amazon 클라우드 컴퓨팅 리소스는 전 세계 여러 리전의 가용성이 높은 데이터 센터 시설에 하우징된다(예: 북미, 유럽 또는 아시아). 각 데이터 센터 위치를 AWS 리전이라고 한다. Amazon RDS를 사용하여 여러 리전에서 DB 인스턴스를 생성할 수 있다.

다음 시나리오는 한 리전의 RDS DB 인스턴스가 다른 리전의 대기 DB 인스턴스에 비동기적으로 복제되는 것을 보여준다. 한 리전을 사용할 수 없게 되더라도 다른 리전의 인스턴스는 계속 사용할 수 있다.



가용 영역

AWS 리전마다 가용 영역 또는 AZ(availability zone)라는 고유한 위치가 여러 개 포함된다. 각 가용 영역은 다른 가용 영역에서 발생한 장애에서 격리되도록 설계되었다. 각 가용 영역은 같은 AWS 리전에 있는 다른 가용 영역에 대해 저렴하고 지연 시간이 짧은 네트워크 연결을 제공하도록 설계되었다. 별도의 가용 영역에서 DB 인스턴스를 시작함으로써 단일 위치에서 장애가 발생할 경우 애플리케이션을 보호할 수 있다.

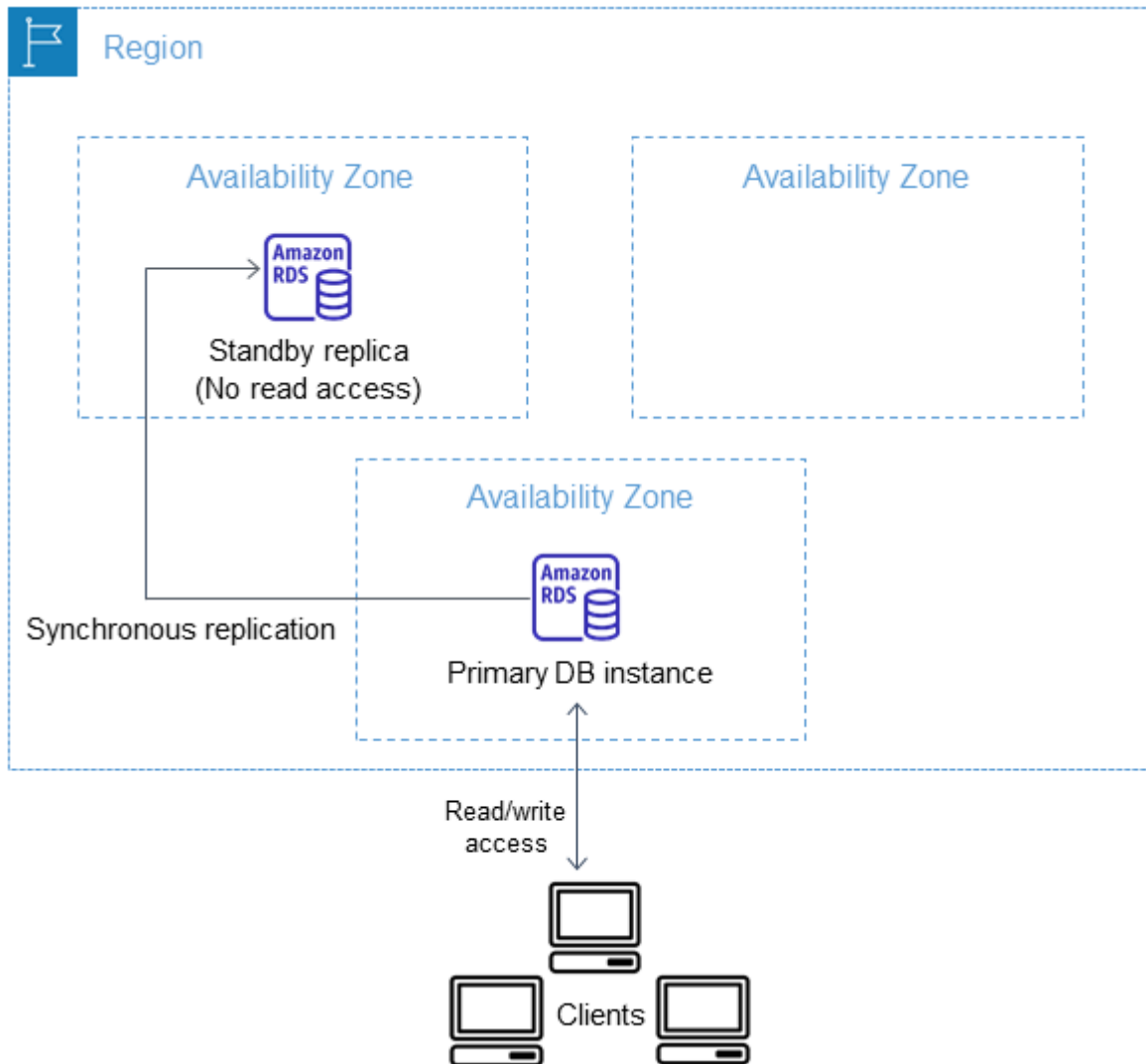
다중 AZ 배포

여러 가용 영역에서 DB 인스턴스를 실행할 수 있는데 이를 다중 AZ 배포라는 옵션이라 한다. 이 옵션을 선택하면 Amazon은 다른 AZ에서 하나 이상의 보조 대기 DB 인스턴스를 자동으로 프로비저닝하고 유지한다. 기본 DB 인스턴스는 가용 영역 전체에서 각 보조 DB 인스턴스로 복제된다. 다중 AZ 배포는 다음과 같은 이점을 제공.

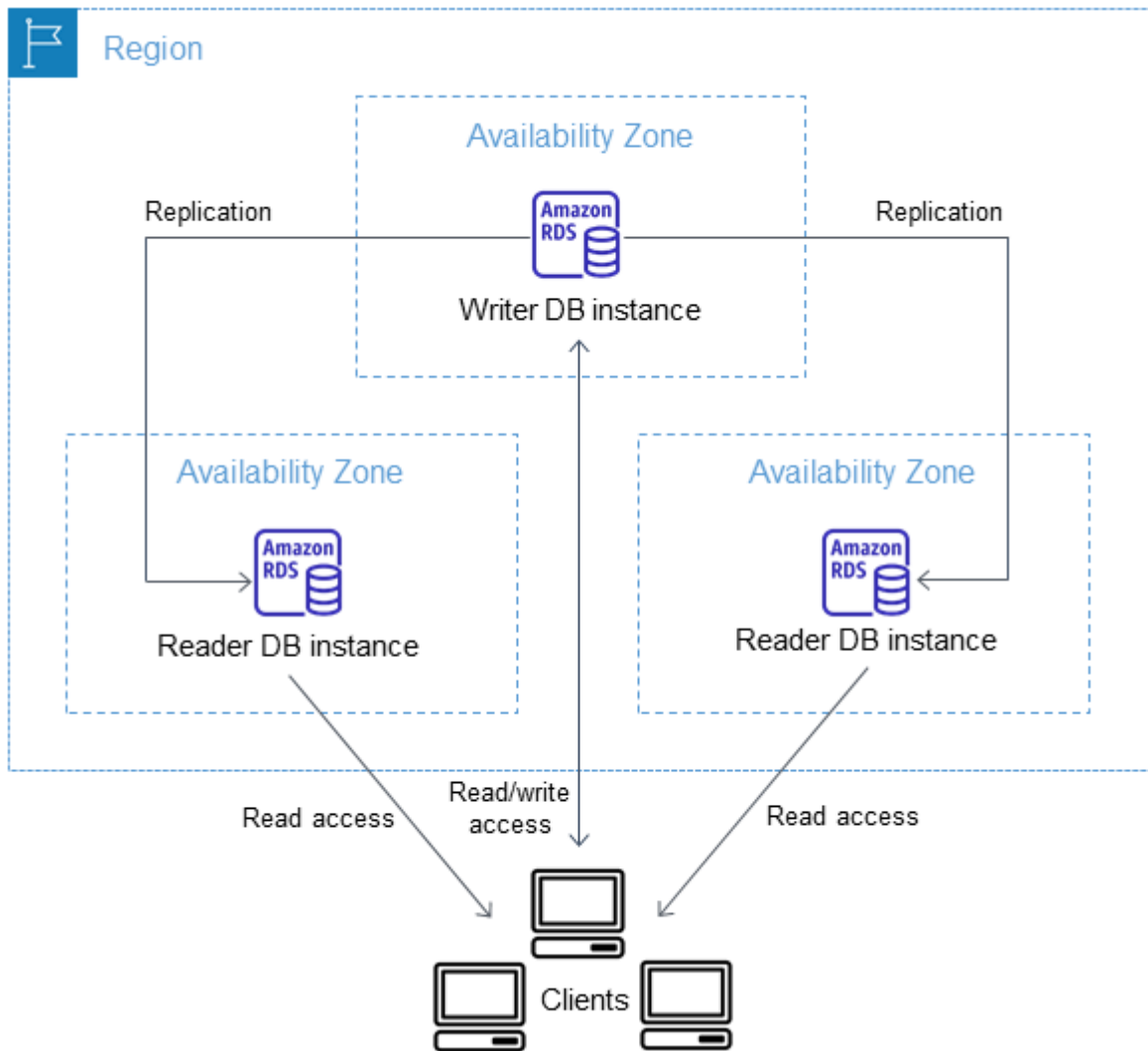
- 데이터 중복성 및 장애 조치 지원 제공
- I/O 작동 중지 제거

- 시스템 백업 중 지연 시간 급증 최소화
- 보조 DB 인스턴스에서 읽기 트래픽 처리(다중 AZ DB 클러스터 배포만 해당)

다음 다이어그램은 Amazon RDS가 다른 가용 영역에서 동기식 대기 복제본을 자동으로 프로비저닝하고 유지 관리하는 다중 AZ DB 인스턴스 배포를 보여준다. 복제본 데이터베이스는 읽기 트래픽을 제공하지 않는다.



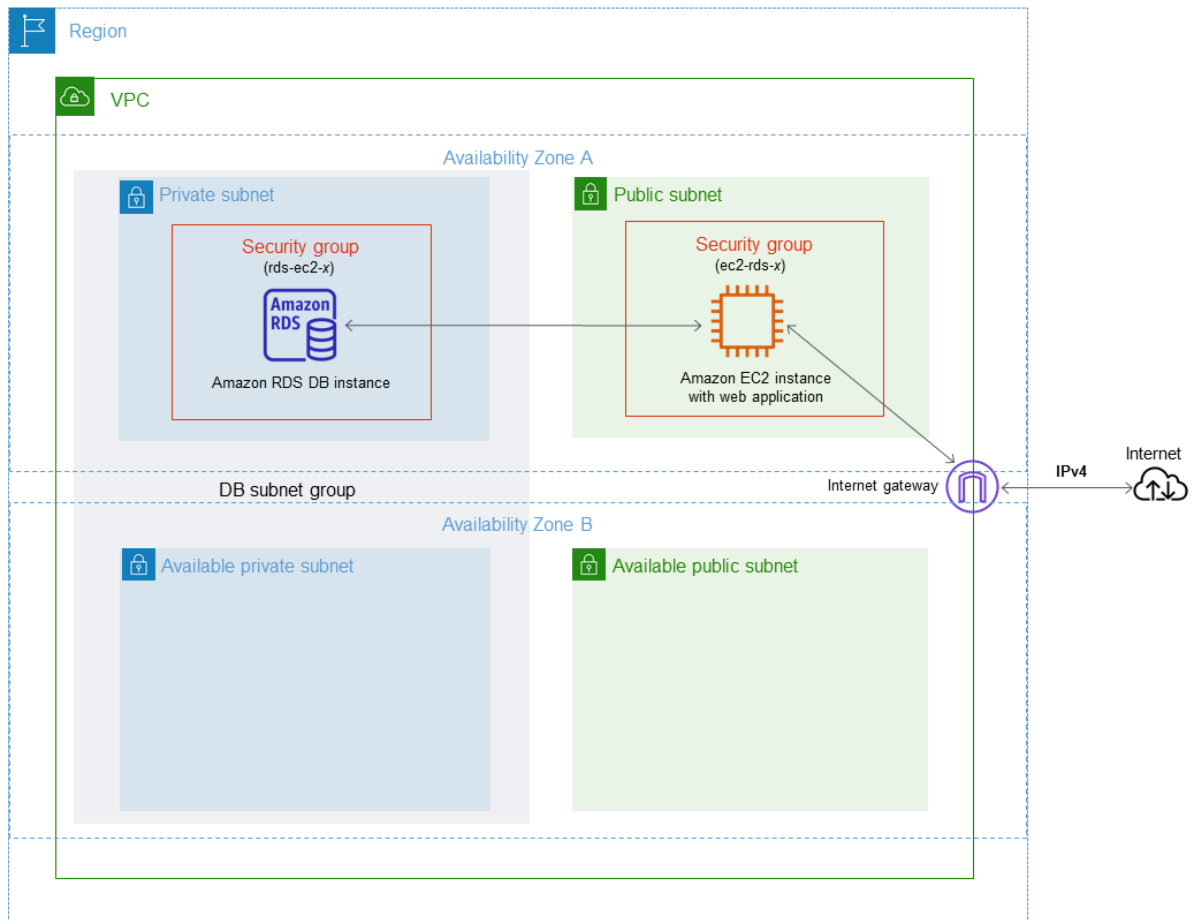
다음 다이어그램은 동일한 AWS 리전에 있는 3개의 개별 가용 영역에 라이더 DB 인스턴스와 2개의 리더 DB 인스턴스가 있는 다중 AZ DB 클러스터 배포를 보여준다. 3개의 DB 인스턴스 모두 읽기 트래픽을 처리할 수 있다.



보안 그룹을 통한 액세스 제어

보안 그룹은 지정한 IP 주소 범위 또는 Amazon EC2 인스턴스에 대한 액세스를 허용하여 DB 인스턴스에 대한 액세스를 제어한다. 보안 그룹을 하나 이상의 DB 인스턴스에 적용할 수 있다.

VPC에서 DB 인스턴스의 일반적인 용도는 동일한 VPC에 있는 애플리케이션 서버와 데이터를 공유하는 것이다. 다음 예제에서는 ec2-rds-*x* VPC 보안 그룹을 사용하여 클라이언트 애플리케이션의 IP 주소를 소스로 사용하는 인바운드 규칙을 정의한다. 애플리케이션 서버는 이 보안 그룹에 속한다. rds-ec2-*x*라는 이름의 보조 보안 그룹은 ec2-rds-*x*를 소스로 지정하고 RDS DB 인스턴스에 연결한다. 보안 그룹 규칙에 따르면 클라이언트 애플리케이션은 DB 인스턴스에 직접 액세스할 수 없지만, EC2 인스턴스는 DB 인스턴스에 액세스할 수 있다.



Amazon RDS 모니터링

모니터링은 Amazon RDS 및 기타 AWS 솔루션의 신뢰성, 가용성 및 성능을 유지하는 데 있어서 중요한 부분이다. AWS에서는 Amazon RDS를 모니터링하고, 이상이 있을 때 이를 보고하고, 적절할 경우 자동 조치를 취할 수 있도록 여러 모니터링 도구를 제공한다.

다양한 자동 및 수동 도구를 사용하여 DB 인스턴스의 성능과 상태를 추적할 수 있다.

Amazon RDS DB 인스턴스 상태 및 권장 사항

Amazon RDS 콘솔, AWS CLI 또는 RDS API를 사용하여 인스턴스의 현재 상태에 대한 세부 정보를 본다. 또한 DB 인스턴스, 읽기 전용 복제본, DB 파라미터 그룹과 같은 데이터베이스 리소스에 대한 자동화된 권장 사항에 응답할 수 있다.

Amazon RDS에 대한 Amazon CloudWatch 지표

Amazon CloudWatch 서비스를 사용하여 DB 인스턴스의 성능 및 상태를 모니터링할 수 있다. Amazon RDS 콘솔에 CloudWatch 성능 차트가 표시된다. Amazon RDS는 각각의 활성 데이터베이스 인스턴스에 대해 1분마다 CloudWatch로 지표를 자동 전송한다. CloudWatch에서 Amazon RDS 지표에 대한 추가 요금은 표시되지 않는다.

Amazon CloudWatch 경보를 사용하면 특정 기간 동안 단일 Amazon RDS 지표를 볼 수 있다. 그런 다음 설정한 임계값과 지표 값을 비교하여 하나 이상의 작업을 수행할 수 있다.

Amazon RDS 성능 개선 도우미 및 운영 체제 모니터링

성능 개선 도우미는 데이터베이스의 부하를 평가하고 조치를 취할 시점과 위치를 결정한다. Amazon RDS 확장 모니터링은 운영 체제에 대한 지표를 실시간으로 확인한다.

통합 AWS 서비스

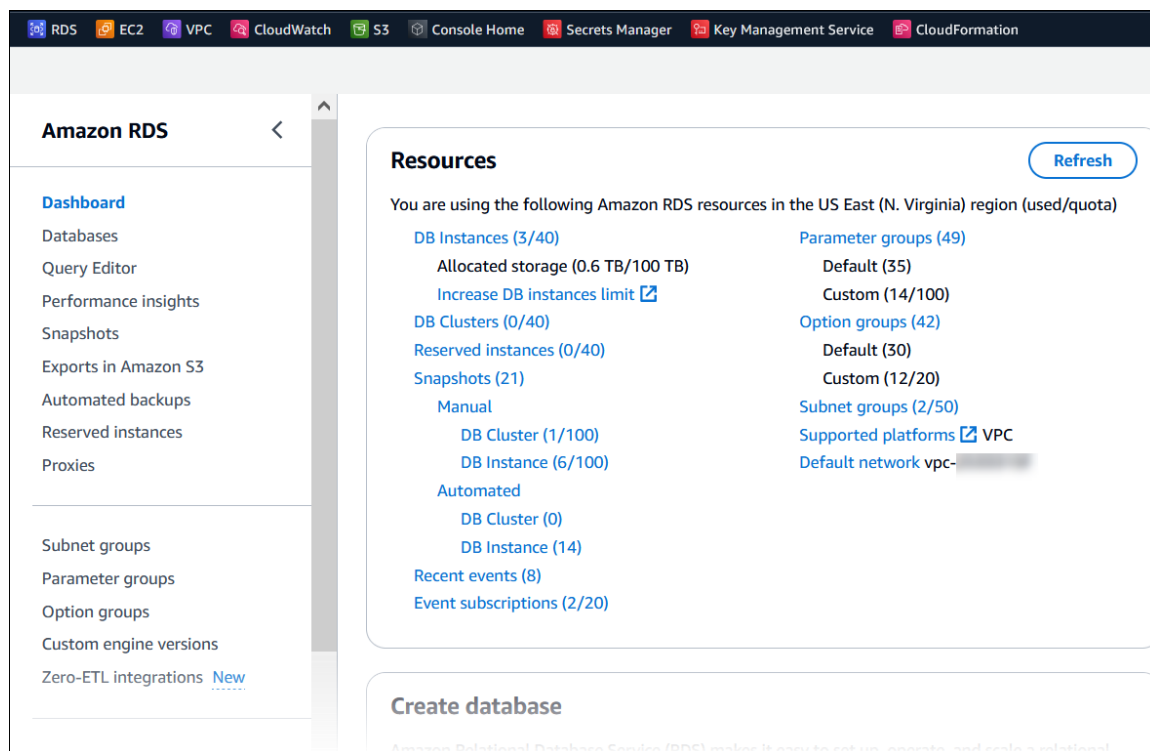
Amazon RDS는 Amazon EventBridge, Amazon CloudWatch Logs, Amazon DevOps Guru와 통합된다.

Amazon RDS에 대한 사용자 인터페이스

Amazon RDS와 상호 작용하는 방법에는 여러 가지가 있다.

AWS Management Console

AWS Management Console은 간단한 웹 기반 사용자 인터페이스. 콘솔에서 프로그래밍 없이 DB 인스턴스를 관리할 수 있다. Amazon RDS 콘솔에 액세스하려면 AWS Management Console에 로그인하고 <https://console.aws.amazon.com/rds/>에서 Amazon RDS 콘솔을 연다.



명령줄 인터페이스

AWS Command Line Interface(AWS CLI)를 사용하여 Amazon RDS API에 대화식으로 액세스할 수 있다.

Amazon RDS API

개발자라면 API를 사용하여 프로그래밍 방식으로 Amazon RDS에 액세스할 수 있다.

애플리케이션을 개발하는 경우에는 AWS 소프트웨어 개발 키트(SDK) 중 하나를 사용하는 것이 좋다. AWS SDK가 인증, 재시도 로직, 오류 처리 등 낮은 레벨 정보를 처리하기 때문에 개발자는 애플리케이션 로직에 더욱 집중할 수 있다. AWS SDK는 다양한 언어로 제공된다.

AWS 그 밖에도 는 라이브러리, 샘플 코드, 자습서 등 보다 쉽게 시작하는 데 도움이 되는 리소스를 제공한다.

MySQL DB 인스턴스 생성

이 예시에서는 **간편 생성**을 사용하여 db.t3.micro DB 인스턴스 클래스에서 MySQL 데이터베이스 엔진을 실행하는 DB 인스턴스를 생성

간편 생성(Easy create)을 사용하여 MySQL DB 인스턴스를 생성하려면

1. <https://console.aws.amazon.com/rds/>에서 AWS Management Console에 로그인한 후 Amazon RDS 콘솔을 연다.
2. Amazon RDS 콘솔의 오른쪽 상단에서 이전에 EC2 인스턴스에 사용한 AWS 리전 선택
3. 탐색 창에서 **데이터베이스**를 선택.
4. **[데이터베이스 생성(Create database)]**을 선택하고 **[표준 생성]**이 선택되어 있는지 확인
5. 구성에서 **MySQL**을 선택

데이터베이스 생성

데이터베이스 생성 방식 선택 정보

☒ 표준 생성

가용성, 보안, 백업 및 유지 관리에 대한 옵션을 포함하여 모든 구성 옵션을 설정합니다.

☐ 손쉬운 생성

권장 모범 사례 구성을 사용합니다. 일부 구성 옵션은 데이터베이스를 생성한 후 변경할 수 있습니다.

엔진 옵션

엔진 유형 정보

☐ Aurora (MySQL Compatible)



☐ Aurora (PostgreSQL Compatible)



☒ MySQL



☐ MariaDB



☐ PostgreSQL



☐ Oracle

ORACLE

6. DB 엔진 버전 선택. 템플릿에서 프리 티어 선택

▼ 필터 숨기기

☐ 다중 AZ DB 클러스터를 지원하는 버전 표시 [정보](#)
기본 DB 인스턴스 1개와 읽기 가능한 대기 DB 인스턴스 2개로 다중 AZ DB 클러스터를 생성합니다. 다중 AZ DB 클러스터는 최대 2배 빠른 트랜잭션 커밋 지연 시간과 일반적으로 35초 미만의 자동 장애 조치를 제공합니다.

☐ Amazon RDS Optimized Writes를 지원하는 버전 표시 [정보](#)
Amazon RDS Optimized Writes는 추가 비용 없이 쓰기 처리량(throughput)을 최대 2배 늘립니다.

엔진 버전
MySQL 8.0.32 ▼

템플릿

해당 사용 사례를 충족하는 샘플 템플릿을 선택하세요.

☐ 프로덕션
고가용성 및 빠르고 일관된 성능을 위해 기본값을 사용하세요.

☐ 개발/테스트
이 인스턴스는 프로덕션 환경 외부에서 개발 용도로 마련되었습니다.

☒ 프리 티어
Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS. [정보](#)

7. DB 인스턴스 식별자에 사용할 이름을 입력

8. 마스터 사용자 이름에 root 입력

설정

DB 인스턴스 식별자 [정보](#)
DB 인스턴스 이름을 입력하세요. 이름은 현재 AWS 리전에서 AWS 계정이 소유하는 모든 DB 인스턴스에 대해 고유해야 합니다.

DB 인스턴스 식별자는 대소문자를 구분하지 않지만 'mydbinstance'와 같이 모두 소문자로 저장됩니다. 제약: 1~60자의 영숫자 또는 하이픈으로 구성되어야 합니다. 첫 번째 문자는 글자여야 합니다. 하이픈 2개가 연속될 수 없습니다. 하이픈으로 끝낼 수 없습니다.

▼ 자격 증명 설정
마스터 사용자 이름 [정보](#)
DB 인스턴스의 마스터 사용자에 로그인 ID를 입력하세요.

1~16자의 영숫자. 첫 번째 문자는 글자여야 합니다.
☐ AWS Secrets Manager에서 마스터 보안 인증 정보 관리
Secrets Manager에서 마스터 사용자 보안 인증을 관리합니다. RDS는 사용자 대신 암호를 생성하고 수명 주기 동안 이를 관리할 수 있습니다.

📘 Secrets Manager에서 마스터 사용자 보안 인증 정보를 관리하는 경우 일부 RDS 기능은 지원되지 않습니다. 자세히 알아보기 [🔗](#)

9. DB 인스턴스에 자동 생성된 마스터 암호를 사용하려면 **암호 자동 생성**을 선택. 마스터 암호를 입력하려면 **암호 자동 생성** 선택을 해제한 다음, **마스터 암호**와 **암호 확인**에 동일한 암호를 입력
10. 인스턴스 구성에서 유.t3.micro 선택

☐ 암호 자동 생성

Amazon RDS에서 사용자를 대신하여 암호를 생성하거나 사용자가 직접 암호를 지정할 수 있습니다.

마스터 암호 정보

제약 조건: 8자 이상의 인쇄 가능한 ASCII 문자. 다음은 포함할 수 없습니다. /(슬래시), '(작은따옴표), "(큰따옴표) 및 @(앳 기호).

마스터 암호 확인 정보

인스턴스 구성

아래의 DB 인스턴스 구성 옵션은 위에서 선택한 엔진에서 지원하는 옵션으로 제한됩니다.



Amazon RDS Optimized Writes – 신규 정보

☐ Amazon RDS Optimized Writes를 지원하는 인스턴스 클래스 표시

DB 인스턴스 클래스 정보

- ☐ 스탠다드 클래스(m 클래스 포함)
- ☐ 메모리 최적화 클래스(r 및 x 클래스 포함)
- ☒ 버스터블 클래스(t 클래스 포함)

db.t3.micro

2 vCPUs 1 GiB RAM 네트워크: 2,085Mbps

☐ 이전 세대 클래스 포함

11. 스토리지 구성

스토리지 유형은 [범용 SSD], 크기는 [20], 스토리지 자동 조정 활성화 체크, 최대 임계값 1000 입력

스토리지

스토리지 유형 [정보](#)

범용 SSD(gp2)
볼륨 크기에 따라 기준 성능 결정

할당된 스토리지 [정보](#)

20

GiB

최솟값은 20GiB이고, 최댓값은 6,144GiB입니다.

스토리지 자동 조정 [정보](#)

애플리케이션의 필요에 따라 데이터베이스 스토리지의 동적 조정 지원을 제공합니다.

- ☒ **스토리지 자동 조정 활성화**
이 기능을 활성화하면 지정한 임계값 초과 후 스토리지를 늘릴 수 있습니다.

최대 스토리지 임계값 [정보](#)

데이터베이스를 지정한 임계값으로 자동 조정하면 요금이 부과됩니다.

1000

GiB

최솟값은 22GiB이고, 최댓값은 6,144GiB입니다.

12. 이전에 생성한 EC2 인스턴스와의 연결을 설정하려면 **EC2 컴퓨팅 리소스에 연결** 선택
이전에 생성한 EC2 인스턴스를 선택

연결 정보



컴퓨팅 리소스

이 데이터베이스의 컴퓨팅 리소스에 대한 연결을 설정할지를 선택합니다. 연결을 설정하면 컴퓨팅 리소스가 이 데이터베이스에 연결할 수 있도록 연결 설정이 자동으로 변경됩니다.

☐ EC2 컴퓨팅 리소스에 연결 안 함

이 데이터베이스의 컴퓨팅 리소스에 대한 연결을 설정하지 않습니다. 나중에 컴퓨팅 리소스에 대한 연결을 수동으로 설정할 수 있습니다.

☒ EC2 컴퓨팅 리소스에 연결

이 데이터베이스의 EC2 컴퓨팅 리소스에 대한 연결을 설정합니다.

EC2 인스턴스 정보

이 데이터베이스의 컴퓨팅 리소스로 추가할 EC2 인스턴스를 선택합니다. VPC 보안 그룹이 이 EC2 인스턴스에 추가됩니다. VPC 보안 그룹은 EC2 인스턴스가 데이터베이스에 액세스하도록 허용하는 인바운드 규칙과 함께 데이터베이스에 추가됩니다.

i-0ab80e82d3798ae4c
myweb1



컴퓨팅 리소스를 추가할 때 일부 VPC 설정을 변경할 수 없음

EC2 컴퓨팅 리소스를 추가하면 이 데이터베이스에 대한 VPC, DB 서브넷 그룹 및 퍼블릭 액세스 설정이 자동으로 선택됩니다. EC2 인스턴스가 데이터베이스에 액세스할 수 있도록 하기 위해 VPC 보안 그룹 rds-ec2-X가 데이터베이스에 추가되고 ec2-rds-X라는 다른 그룹이 EC2 인스턴스에 추가됩니다. 컴퓨팅 리소스를 제거해야만 데이터베이스의 새 보안 그룹을 제거할 수 있습니다.

Virtual Private Cloud(VPC) 정보

VPC를 선택합니다. VPC는 이 DB 인스턴스의 가상 네트워킹 환경을 정의합니다.

Default VPC (vpc-02c8a870ed7bba6ab)
3 서브넷, 3 Availability Zones



해당 DB 서브넷 그룹이 있는 VPC만 나열됩니다.



데이터베이스를 생성한 후에는 VPC를 변경할 수 없습니다.

13. 백업 보존 기간 7일, 암호화 활성화 체크

백업 보존 기간 [정보](#)

자동 백업이 유지되는 일수(1~35)입니다.

7

▼

 일

백업 기간 [정보](#)

RDS에서 매일 자동 백업을 수행하는 시간 범위(UTC)입니다.

- ☐ 기간 선택
- ☒ 기본 설정 없음

☒ 스냅샷으로 태그 복사

백업 복제 [정보](#)

☐ 다른 AWS 리전에 복제 활성화

복제를 활성화하면 현재 리전뿐만 아니라 재해 복구를 위해 선택한 리전에도 DB 인스턴스의 백업이 자동으로 생성됩니다.

암호화

☒ 암호화 활성화

지정된 인스턴스를 암호화하려면 이 옵션을 선택합니다. AWS Key Management Service 콘솔을 사용하여 마스터 키 ID와 별칭이 생성된 후 해당 항목이 목록에 표시됩니다. [정보](#)

AWS KMS 키 [정보](#)

(default) aws/rds

▼

계정

585899940942

KMS 키 ID

alias/aws/rds

로그 내보내기

Amazon CloudWatch Logs로 게시할 로그 유형 선택

- ☐ 감사 로그
- ☐ 에러 로그
- ☐ 일반 로그

14. 초기 데이터베이스 이름 입력. 나머지 기본값

모니터링

모니터링

☐ Enhanced 모니터링 활성화

Enhanced 모니터링 지표를 활성화하면 다른 프로세스 또는 스레드에서 CPU를 사용하는 방법을 확인하려는 경우에 유용합니다.

▼ 추가 구성

데이터베이스 옵션, 암호화 켜짐, 백업 켜짐, 역추적 꺼짐, 유지 관리, CloudWatch Logs, 삭제 방지 켜짐.

데이터베이스 옵션

초기 데이터베이스 이름 [정보](#)

myweb1db

데이터베이스 이름을 지정하지 않으면 Amazon RDS에서 데이터베이스를 생성하지 않습니다.

DB 파라미터 그룹 [정보](#)

default.mysql8.0


옵션 그룹 [정보](#)

default:mysql-8-0

백업

☒ 자동 백업을 활성화합니다.

데이터베이스의 특정 시점 스냅샷을 생성합니다.

 Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to details [here](#).

가용 영역 정보

us-east-2a ▼

인증 기관 - 선택 사항 정보

서버 인증서를 사용하면 Amazon 데이터베이스에 대한 연결이 이루어지고 있는지 검증하여 추가 보안 계층을 제공합니다. 프로비저닝하는 모든 데이터베이스에 자동으로 설치되는 서버 인증서를 확인하여 이를 수행합니다.

rds-ca-2019 (기본값) ▼

인증 기관을 선택하지 않으면 RDS에서 대신 인증 기관을 선택합니다.

▶ 추가 구성

데이터베이스 인증

데이터베이스 인증 옵션 정보

- ☒ 암호 인증
데이터베이스 암호를 사용하여 인증합니다.
- ☐ 암호 및 IAM 데이터베이스 인증
AWS IAM 사용자 및 역할을 통해 데이터베이스 암호와 사용자 자격 증명을 사용하여 인증합니다.
- ☐ 암호 및 Kerberos 인증
권한이 부여된 사용자가 Kerberos 인증을 사용하여 이 DB 인스턴스에서 인증하도록 허용하려는 디렉터리를 선택합니다.

③ 데이터베이스를 생성한 후에는 VPC를 변경할 수 없습니다.

DB 서브넷 그룹 정보

DB 서브넷 그룹을 선택합니다. DB 서브넷 그룹은 선택한 VPC에서 DB 인스턴스가 어떤 서브넷과 IP 범위를 사용할 수 있는지를 정의합니다.

☐ 기존 항목 선택
기존 서브넷 그룹 선택

☒ 자동 설정
RDS가 사용자를 위해 새 서브넷 그룹을 생성하거나 기존 서브넷 그룹을 재사용합니다.

DB 서브넷 그룹 이름

rds-ec2-db-subnet-group-1 ▼

새 DB 서브넷 그룹이 생성되었습니다.

퍼블릭 액세스 정보

☐ 예
RDS는 데이터베이스에 퍼블릭 IP 주소를 할당합니다. VPC 외부의 Amazon EC2 인스턴스 및 다른 리소스가 데이터베이스에 연결할 수 있습니다. VPC 내부의 리소스도 데이터베이스에 연결할 수 있습니다. 데이터베이스에 연결할 수 있는 리소스를 지정하는 VPC 보안 그룹을 하나 이상 선택합니다.

☒ 아니요
RDS는 퍼블릭 IP 주소를 데이터베이스에 할당하지 않습니다. VPC 내부의 Amazon EC2 인스턴스 및 다른 리소스만 데이터베이스에 연결할 수 있습니다. 데이터베이스에 연결할 수 있는 리소스를 지정하는 VPC 보안 그룹을 하나 이상 선택합니다.

VPC 보안 그룹(방화벽) 정보

데이터베이스에 대한 액세스를 허용할 VPC 보안 그룹을 하나 이상 선택합니다. 보안 그룹 규칙이 적절한 수신 트래픽을 허용하는지 확인합니다.

☐ 기존 항목 선택
기존 VPC 보안 그룹 선택

☒ 새로 생성
새 VPC 보안 그룹 생성

새 VPC 보안 그룹 이름

myweb1_security

③ Amazon RDS는 컴퓨팅 리소스와의 연결을 허용하는 새 VPC 보안 그룹 *rds-ec2-1*을(를) 추가합니다.

15. 데이터베이스 생성 선택

DB 인스턴스의 마스터 사용자 이름 및 암호를 보려면 **View credential details(자격 증명 세부 정보 보기)**를 선택한다.

DB 인스턴스를 마스터 사용자로 연결하려면 화면에 나타난 사용자 이름과 암호를 사용

중요

마스터 사용자 암호를 다시 볼 수는 없다. 따라서 기록을 해두지 않으면 이를 변경해야 한다.

DB 인스턴스가 사용 가능한 상태가 되고 난 후에 마스터 사용자 암호를 변경해야 하는 경우에는 다음과 같은 방법으로 DB 인스턴스를 수정할 수 있다.

16. 데이터베이스 목록에서 새 MySQL DB 인스턴스의 이름을 선택하면 세부 정보가 표시된다.

DB 인스턴스를 사용할 준비가 될 때까지 DB 인스턴스의 상태는 **생성 중**이다.

Summary			
DB identifier database-test1	CPU -	Status ⌚ Creating	Class db.r6g.large
Role Instance	Current activity	Engine MySQL Community	Region & AZ us-east-1c

상태가 **Available**(사용 가능)로 변경되면 DB 인스턴스에 연결할 수 있다. DB 인스턴스 클래스와 스토리지의 양에 따라 새 인스턴스를 사용할 수 있을 때까지 최대 20분이 걸릴 수 있다.

MySQL DB 인스턴스에 연결

표준 SQL 클라이언트 애플리케이션을 사용해 DB 인스턴스에 연결할 수 있다. 이 예시에서는 mysql 명령줄 클라이언트를 사용해 MySQL DB 인스턴스에 연결한다.

MySQL DB 인스턴스에 연결하는 방법

1. DB 인스턴스의 엔드포인트(DNS 이름)와 포트 번호를 찾는다.
 - a. <https://console.aws.amazon.com/rds/>에서 AWS Management Console에 로그인한 후 Amazon RDS 콘솔을 연다.
 - b. Amazon RDS 콘솔의 오른쪽 상단에서 DB 인스턴스의 AWS 리전을 선택
 - c. 탐색 창에서 **Databases**(데이터베이스)를 선택
 - d. 세부 정보를 표시하고자 하는 MySQL DB 인스턴스 이름을 선택
 - e. **Connectivity & security**(연결 및 보안) 탭에서 엔드포인트를 복사. 또한 포트 번호 확인. DB 인스턴스에 연결하려면 엔드포인트와 포트 번호가 모두 필요하다.

RDS > Databases > database-test1

database-test1

Summary

DB identifier database-test1	CPU <div><div></div>2.58%</div>
Role Instance	Current activity <div><div></div>0 Connections</div>

Connectivity & security

Monitoring

Logs & events

Configuration

Connectivity & security

<h4>Endpoint & port</h4> <div>Endpoint database-test1.123456789012.us-east-1.rds.amazonaws.com</div> <div>Port 3306</div>	<h4>Networking</h4> <div>Availability Zone us-east-1c</div> <div>VPC vpc-</div> <div>Subnet group default</div>
---------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------

2. 앞에서 만든 EC2 인스턴스에 연결

SSH를 사용하여 EC2 인스턴스에 연결하는 것이 좋다. Windows, Linux 또는 Mac에 SSH 클라이언트 유틸리티가 설치된 경우 다음 명령 형식을 사용하여 인스턴스에 연결할 수 있다.

```
ssh -i location_of_pem_file ec2-user@ec2-instance-public-dns-name
```

예를 들어 ec2-database-connect-key-pair.pem이 Linux의 /dir1에 저장되어 있고, EC2 인스턴스의 퍼블릭 IPv4 DNS가 ec2-12-345-678-90.compute-1.amazonaws.com이라고 가정하면 SSH 명령은 다음과 같이 표시된다.

```
ssh -i /dir1/ec2-database-connect-key-pair.pem ec2-user@ec2-12-345-678-90.compute-1.amazonaws.com
```

3. EC2 인스턴스에서 소프트웨어를 업데이트하여 최신 버그 수정 및 보안 업데이트를 받는다. 이렇게 하려면 다음 실행.

참고

-y 옵션을 사용하면 확인 여부를 묻지 않고 업데이트를 설치. 설치 전에 업데이트 정보를 확인하려면 이 옵션을 생략

```
sudo dnf update -y
```

4. Amazon Linux 2023에서 MariaDB의 mysql 명령줄 클라이언트를 설치하려면 다음 명령을 실행

```
sudo dnf install mariadb105
```

5. 설치한 클라이언트 프로그램에서 MySQL DB 인스턴스에 연결.

```
mysql -h endpoint -P 3306 -u admin -p
```

사용자에 대한 암호를 입력하면 다음과 유사한 출력이 나타난다.

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MySQL connection id is 3082

Server version: 8.0.28 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>

6. SQL 명령을 실행

DB 인스턴스 삭제

1. AWS Management Console에 로그인한 후 <https://console.aws.amazon.com/rds/>에서 Amazon RDS 콘솔을 연다.
2. 탐색 창에서 **데이터베이스**를 선택
3. 삭제할 DB 인스턴스를 선택

4. [**Actions**]에 대해 [Delete]를 선택
5. **최종 스냅샷을 생성하시겠습니까?** 및 **자동 백업 보존**을 선택 해제
6. 확인을 완료하고 **삭제**를 선택

Amazon EC2의 네트워킹

Amazon VPC를 사용하면 Virtual Private Cloud(VPC)라는 AWS 계정 전용 가상 네트워크로 Amazon EC2 인스턴스와 같은 AWS 리소스를 시작할 수 있다. 인스턴스를 시작할 때 VPC에서 서브넷을 선택할 수 있다. 인스턴스는 논리적 가상 네트워크 카드인 기본 네트워크 인터페이스로 구성된다. 인스턴스는 서브넷의 IPv4 주소에서 기본 프라이빗 IP 주소를 수신하며, 이 주소는 기본 네트워크 인터페이스에 할당된다.

인스턴스가 Amazon의 퍼블릭 IP 주소 풀에서 퍼블릭 IP 주소를 수신하는지 여부를 제어할 수 있다. 인스턴스의 퍼블릭 IP 주소는 인스턴스가 중지되거나 종료될 때까지 인스턴스와 연결된다. 영구 퍼블릭 IP 주소가 필요한 경우 AWS 계정에 탄력적 IP 주소를 할당하고 이를 인스턴스 또는 네트워크 인터페이스에 연결할 수 있다. 탄력적 IP 주소는 사용자가 해제할 때까지 AWS 계정과 연결되며, 필요한 경우 하나의 인스턴스에서 다른 인스턴스로 이동할 수 있다. 자신의 IP 주소 범위를 AWS 계정에 가져와서 주소 풀로 표시한 다음 주소 풀에서 탄력적 IP 주소를 할당할 수 있다.

네트워크 성능을 높이고 지연 시간을 줄이기 위해 배치 그룹으로 인스턴스를 시작할 수 있다. 향상된 네트워킹을 사용하여 PPS(초당 패킷) 성능을 크게 높일 수 있다. 지원되는 인스턴스 유형에 연결할 수 있는 네트워크 디바이스인 Elastic Fabric Adapter(EFA)를 사용하여 고성능 컴퓨팅 및 기계 학습 애플리케이션을 가속화할 수 있다.

● 리전 및 영역

Amazon EC2는 전 세계의 여러 곳에서 호스팅되고 있다. 이 위치는 AWS 리전, 가용 영역, Local Zones, AWS Outposts 및 Wavelength Zone으로 구성된다.

- 각 리전은 개별 지리 영역
- 가용 영역은 각 리전 내에 있는 여러 격리된 위치
- Local Zones에서는 최종 사용자에게 가까운 여러 위치에 컴퓨팅, 스토리지 등의 리소스를 배치할 수 있는 기능을 제공
- AWS Outposts는 네이티브 AWS 서비스, 인프라 및 운영 모델을 사실상 모든 데이터 센터, 코로케이션 공간 또는 온프레미스 시설로 옮길 수 있다.
- Wavelength Zone을 사용하면 개발자는 5G 디바이스 및 최종 사용자에게 매우 짧은 지연 시간을 제공하는 애플리케이션을 빌드할 수 있다. Wavelength는 표준 AWS 컴퓨팅 및 스토리지 서비스를 통신 사업자의 5G 네트워크 엣지에 배포한다.

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

리전

각 리전은 다른 리전에서 격리되도록 설계되었다. 이를 통해 가장 강력한 내결함성 및 안정성을 달성할 수 있다.

리소스를 볼 때 지정한 리전에 연결된 리소스만 표시된다. 리전이 서로 격리되어 있고 여러 리전에 리소스가 자동으로 복제되지 않기 때문이다.

인스턴스를 시작할 때 동일한 리전에 있는 AMI를 선택해야 한다. AMI가 다른 리전에 있는 경우 해당 AMI를 사용 중인 리전에 복사할 수 있다.

사용 가능한 리전

계정을 통해 자신이 사용할 수 있는 리전을 결정한다.

- 하나의 AWS 계정은 여러 개의 리전을 제공하므로 사용자는 자신의 요구 사항에 맞는 위치에서 Amazon EC2 인스턴스를 시작할 수 있다. 예를 들어 유럽의 고객들과 좀더 가까운 곳에 위치하거나 또는 법적 요구사항을 준수하기 위해 유럽에 소재한 위치에서 인스턴스를 실행할 필요가 있을 수 있다.
- AWS GovCloud(미국 서부) 계정은 AWS GovCloud(미국 서부) 리전 및 AWS GovCloud(미국 동부) 리전에 액세스할 수 있다.
- Amazon AWS(중국) 계정은 오직 베이징 및 닝샤 리전에 대한 액세스 권한을 제공한다.

다음 표에는 AWS 계정이 제공하는 리전이 나열되어 있다.

코드	이름	요트인 상태
us-east-1	미국 동부(버지니아 북부)	불 필요
us-east-2	미국 동부(오하이오)	불 필요
us-west-1	미국 서부(캘리포니아 북부)	불 필요
us-west-2	미국 서부(오레곤)	불 필요
af-south-1	아프리카(케이프타운)	필수

코드	이름	옵트인 상태
ap-east-1	아시아 태평양(홍콩)	필수
ap-south-2	아시아 태평양(하이데라바드)	필수
ap-southeast-3	아시아 태평양(자카르타)	필수
ap-southeast-5	아시아 태평양(말레이시아)	필수
ap-southeast-4	아시아 태평양(멜버른)	필수
ap-south-1	아시아 태평양(뭄바이)	불필요
ap-northeast-3	아시아 태평양(오사카)	불필요
ap-northeast-2	아시아 태평양(서울)	불필요
ap-southeast-1	아시아 태평양(싱가포르)	불필요
ap-southeast-2	아시아 태평양(시드니)	불필요
ap-northeast-1	아시아 태평양(도쿄)	불필요
ca-central-1	캐나다(중부)	불필요
ca-west-1	캐나다 서부(캘거리)	필수
cn-north-1	중국(베이징)	불필요
cn-northwest-1	중국(닝샤)	불필요
eu-central-1	유럽(프랑크푸르트)	불필요
eu-west-1	유럽(아일랜드)	불필요
eu-west-2	유럽(런던)	불필요
eu-south-1	유럽(밀라노)	필수
eu-west-3	유럽(파리)	불필요
eu-south-2	유럽(스페인)	필수
eu-north-1	유럽(스톡홀름)	불필요
eu-central-2	유럽(취리히)	필수
il-central-1	이스라엘(텔아비브)	필수
me-south-1	중동(바레인)	필수
me-central-1	중동(UAE)	필수

코드	이름	유프인 상태
sa-east-1	남아메리카(상파울루)	불 필요

리전 엔드포인트

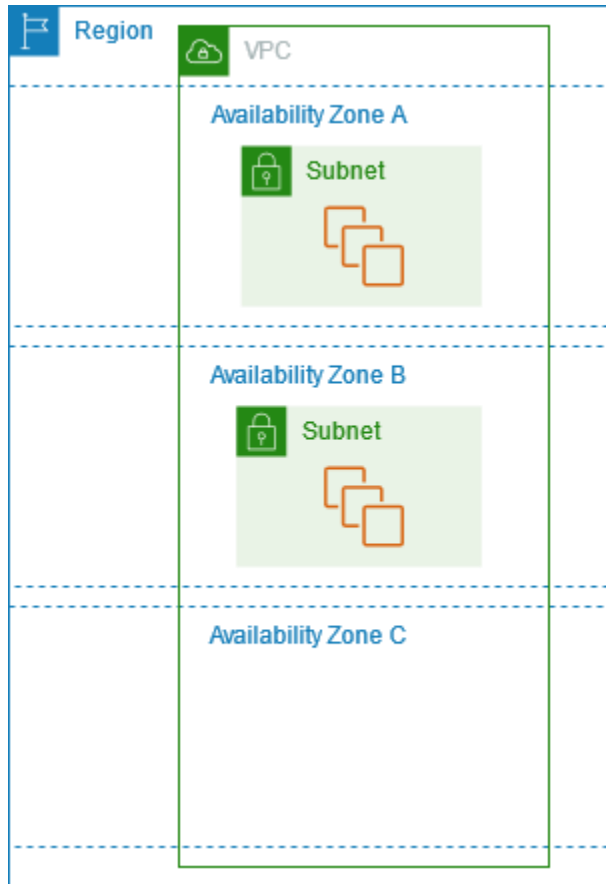
명령줄 인터페이스 또는 API 작업을 사용해서 인스턴스로 작업할 때는 리전 엔드포인트를 지정해야 한다.

가용 영역

각 리전은 가용 영역이라고 알려진 격리된 위치를 여러 개 가지고 있다. 가용 영역의 코드는 리전 코드와 문자 식별자를 조합한 것이다. 예: us-east-1a.

인스턴스를 시작할 때 리전과 VPC(Virtual Private Cloud)를 선택한 다음 가용 영역 중 하나에서 직접 서브넷을 선택하거나 자동으로 서브넷이 선택되게 할 수 있다. 복수의 가용 영역에 걸쳐 인스턴스를 배포했을 때 하나의 인스턴스에 장애가 발생한 경우에 대비하여, 다른 가용 영역의 인스턴스가 장애가 발생한 인스턴스 관련 요청을 처리할 수 있도록 애플리케이션을 설계할 수 있다. 또한 탄력적 IP 주소를 사용하여 한 가용 영역에서 인스턴스의 장애가 발생한 경우 다른 가용 영역의 인스턴스로 주소를 신속하게 매핑함으로써 인스턴스의 장애를 마스킹할 수 있다.

다음 다이어그램은 AWS 리전의 여러 가용 영역을 보여준다. 가용 영역 A와 가용 영역 B에는 각각 하나의 서브넷이 있고 각 서브넷에는 인스턴스가 있습니다. 가용 영역 C에는 서브넷이 없으므로 이 가용 영역으로 인스턴스를 시작할 수 없다.



가용 영역이 시간에 따라 커지면서 가용 영역을 확장할 수 있는 역량 부족으로 인해 가용 영역이 제한될 수 있다. 이런 문제가 생긴 경우 제한을 받는 가용 영역에서 인스턴스를 실행하지 못하도록 한다(해당 가용 영역에서 이미 인스턴스를 보유하고 있는 경우는 제외). 또 최종적으로는 새 계정에 대해서는 가용 영역의 목록에서 제한을 받는 가용 영역을 제거하게 될 수도 있다. 따라서 어떤 리전에 대해 한 계정에서 사용 가능한 가용 영역의 수는 다른 계정과 다를 수 있다.

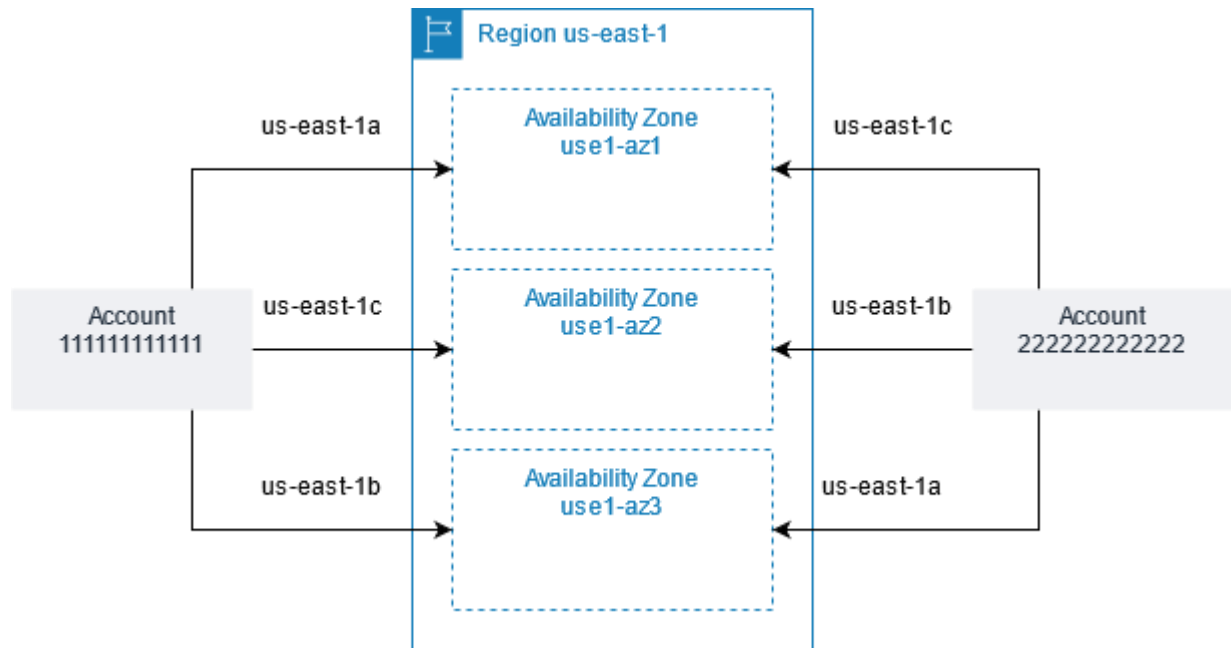
AZ ID

리전의 가용 영역에 걸쳐 리소스가 배포될 수 있도록 가장 오래된 리전에서 각 AWS 계정의 코드에 가용 영역을 독립적으로 매핑한다. 예를 들어 AWS 계정의 us-east-1a는 다른 AWS 계정에 대한 us-east-1a와 물리적 위치가 동일하지 않을 수 있다.

가용 영역을 매핑하는 리전을 포함한 모든 리전의 계정 전체에서 가용 영역을 조정하려면 가용 영역에 대한 고유하고 일관된 식별자인 AZ ID를 사용한다. 예를 들어 use1-az1은 us-east-1 리전의 AZ ID이고, 모든 AWS 계정에서 물리적 위치가 동일하다. 계정의 AZ ID를 보고 다른 계정의 리소스와 관련된 리소스의 물리적 위치를 확인할 수 있다. 예를 들어 AZ ID가 use1-az2인 가용 영역의 서브넷을 다른 계정과 공유하면 이 서브넷은 AZ ID가 use1-az2인 가용 영역의 계정에서 사용할 수 있다.

계정의 AZ ID를 보려면 EC2 대시보드의 **서비스 상태** 패널을 확인하거나 describe-availability-zones AWS CLI 명령을 사용한다.

다음 다이어그램은 가용 영역 코드를 AZ ID에 매핑하는 서로 다른 두 개의 계정을 보여준다.



사용 가능한 가용 영역

다음 목록에 표시된 대로 각 리전에는 여러 가용 영역이 있다.

- 미국 동부(버지니아 북부) - use1-az1 | use1-az2 | use1-az3 | use1-az4 | use1-az5 | use1-az6
- 미국 동부(오하이오) - use2-az1 | use2-az2 | use2-az3
- 미국 서부(캘리포니아 북부) - usw1-az1 | usw1-az2 | usw1-az3 †
- 미국 서부(오레곤) - usw2-az1 | usw2-az2 | usw2-az3 | usw2-az4
- 아프리카(케이프타운) - afs1-az1 | afs1-az2 | afs1-az3
- 아시아 태평양(홍콩) - ape1-az1 | ape1-az2 | ape1-az3
- 아시아 태평양(하이데라바드) - aps2-az1 | aps2-az2 | aps2-az3
- 아시아 태평양(자카르타) - apse3-az1 | apse3-az2 | apse3-az3
- 아시아 태평양(말레이시아) - apse5-az1 | apse5-az2 | apse5-az3
- 아시아 태평양(멜버른) - apse4-az1 | apse4-az2 | apse4-az3
- 아시아 태평양(뭄바이) - aps1-az1 | aps1-az2 | aps1-az3
- 아시아 태평양(오사카) - apne3-az1 | apne3-az2 | apne3-az3
- 아시아 태평양(서울) - apne2-az1 | apne2-az2 | apne2-az3 | apne2-az4
- 아시아 태평양(싱가포르) - apse1-az1 | apse1-az2 | apse1-az3
- 아시아 태평양(시드니) - apse2-az1 | apse2-az2 | apse2-az3
- 아시아 태평양(도쿄) - apne1-az1 | apne1-az2 | apne1-az3 | apne1-az4
- 캐나다(중부) - cac1-az1 | cac1-az2 | cac1-az4
- 캐나다 서부(캘거리) - caw1-az1 | caw1-az2 | caw1-az3

- 유럽(프랑크푸르트) – euc1-az1 | euc1-az2 | euc1-az3
- 유럽(아일랜드) – euw1-az1 | euw1-az2 | euw1-az3
- 유럽(런던) – euw2-az1 | euw2-az2 | euw2-az3
- 유럽(밀라노) – eus1-az1 | eus1-az2 | eus1-az3
- 유럽(파리) – euw3-az1 | euw3-az2 | euw3-az3
- 유럽(스페인) – eus2-az1 | eus2-az2 | eus2-az3
- 유럽(스톡홀름) – eun1-az1 | eun1-az2 | eun1-az3
- 유럽(취리히) – euc2-az1 | euc2-az2 | euc2-az3
- 이스라엘(텔아비브) – ilc1-az1 | ilc1-az2 | ilc1-az3
- 중동(바레인) – mes1-az1 | mes1-az2 | mes1-az3
- 중동(UAE) – mec1-az1 | mec1-az2 | mec1-az3
- 남아메리카(상파울루) – sae1-az1 | sae1-az2 | sae1-az3
- AWS GovCloud(미국 동부) – usge1-az1 | usge1-az2 | usge1-az3
- AWS GovCloud(미국 서부) – usgw1-az1 | usgw1-az2 | usgw1-az3

가용 영역의 인스턴스

인스턴스를 시작할 때 특정 고객과 가까운 곳에 인스턴스를 배치하거나 법률 또는 기타 요구 사항을 준수할 수 있도록 적절한 리전을 선택한다. 각각의 개별적인 가용 영역에서 인스턴스를 시작함으로써 리전의 단일 위치에서 장애가 발생할 경우 애플리케이션을 보호할 수 있다.

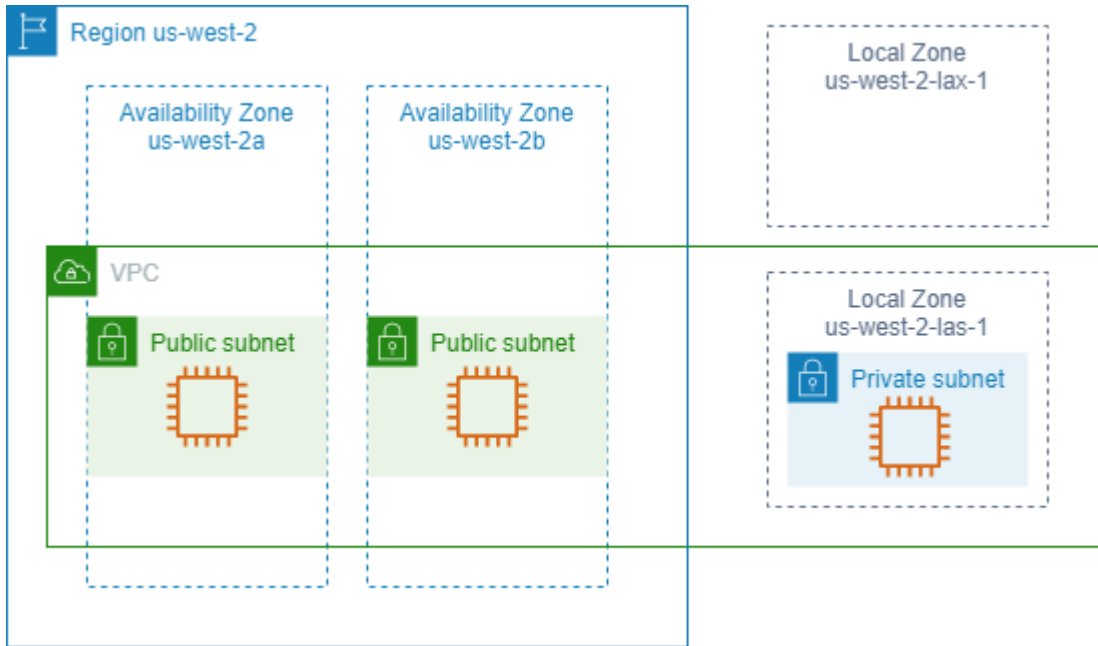
인스턴스를 시작할 때 상황에 따라 사용 중인 리전의 가용 영역을 지정할 수 있다. 가용 영역을 지정하지 않으면 가용 영역이 자동으로 선택된다. 초기 인스턴스를 실행할 때는 기본 가용 영역을 그대로 사용하는 것이 좋다. 이를 통해 시스템 상태 및 가용 용량에 따라 사용자에게 가장 알맞은 가용 영역을 선택할 수 있기 때문이다. 추가 인스턴스를 시작하는 경우 새 인스턴스가 실행 중인 인스턴스와 가까이 있거나 분리되어 있어야 하는 경우에만 가용 영역을 지정한다.

로컬 영역

Local Zone은 사용자와 지리적으로 근접한 AWS 리전의 확장이다. Local Zones는 자체 연결을 통해 인터넷에 연결되며 AWS Direct Connect를 지원하므로 Local Zone에 생성된 리소스를 짧은 지연 시간의 통신을 통해 로컬 사용자에게 제공할 수 있다.

로컬 영역에 대한 코드는 물리적 위치를 나타내는 식별자가 뒤에 붙는 리전 코드이다. 로스앤젤레스의 us-west-2-lax-1을 예로 들 수 있습니다.

다음 다이어그램에서는 AWS 리전 us-west-2, 가용 영역 2개, Local Zones 2개를 보여준다. VPC는 가용 영역과 Local Zones 중 하나에 걸쳐 있다. VPC의 각 영역에는 하나의 서브넷이 있고 각 서브넷에는 인스턴스가 있다.



사용 가능한 로컬 영역

로컬 영역의 인스턴스

Local Zone을 사용하려면 먼저 사용하도록 설정해야 한다. 그런 다음 로컬 영역에서 서브넷을 생성한다. 인스턴스를 시작할 때 로컬 영역 서브넷을 지정하여 로컬 영역의 로컬 영역 서브넷에 배치할 수 있다.

로컬 영역에서 인스턴스를 시작할 때 네트워크 경계 그룹의 IP 주소도 할당한다. 네트워크 경계 그룹은 AWS가 IP 주소를 알릴 때 사용하는 가용 영역, Local Zones 또는 Wavelength Zone의 고유한 집합(예: us-west-2-lax-1a)이다. 네트워크 경계 그룹에서 다음 IP 주소를 할당할 수 있다.

- Amazon에서 제공하는 탄력적 IPv4 주소
- Amazon 제공 IPv6 VPC 주소(로스앤젤레스 영역에서만 사용 가능)

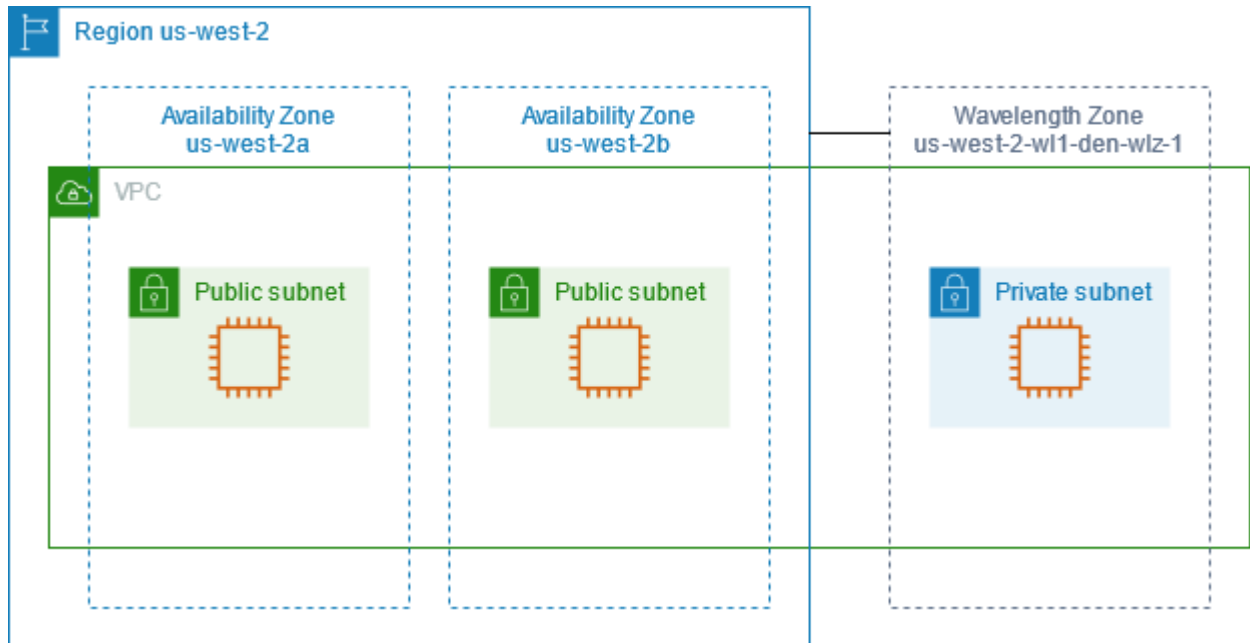
Wavelength Zone

AWS Wavelength을(를) 사용하면 개발자는 모바일 디바이스 및 최종 사용자에게 매우 짧은 지연 시간을 제공하는 애플리케이션을 빌드할 수 있다. Wavelength는 표준 AWS 컴퓨팅 및 스토리지 서비스를 통신 사업자의 5G 네트워크 엣지에 배포한다. 개발자는 Virtual Private Cloud(VPC)를 하나 이상의 Wavelength Zone으로 확장한 다음, Amazon EC2 인스턴스와 같은 AWS 리소스를 사용하여 매우 짧은 지연 시간으로 리전의 AWS 서비스에 연결해야 하는 애플리케이션을 실행할 수 있다.

Wavelength Zone은 Wavelength 인프라가 배포된 통신 사업자 위치의 격리된 영역이다. Wavelength Zone은 리전에 연결되어 있다. Wavelength Zone은 리전의 논리적 확장이며, 리전의 제어 플레인에 의해 관리된다.

Wavelength 영역에 대한 코드는 물리적 위치를 나타내는 식별자가 뒤에 붙는 리전 코드이다. 보스턴의 us-east-1-wl1-bos-wlz-1을 예로 들 수 있다.

다음 다이어그램에서는 AWS 리전 us-west-2, 가용 영역 2개, Wavelength 영역 1개를 보여준다. VPC는 가용 영역과 Wavelength 영역에 걸쳐 있다. VPC의 각 영역에는 하나의 서브넷이 있고 각 서브넷에는 인스턴스가 있다.



Wavelength 영역의 인스턴스

Wavelength Zone을 사용하려면 먼저 Wavelength Zone을 옵트인해야 한다. 그런 다음 Wavelength 영역에서 서브넷을 생성한다. 인스턴스를 시작할 때 Wavelength 서브넷을 지정할 수 있다. 또한 네트워크 경계 그룹에서 통신 사업자 IP 주소를 할당한다. 네트워크 경계 그룹은 AWS가 IP 주소를 알리는 가용 영역, Local Zones 또는 Wavelength Zone의 고유한 집합(예: us-east-1-wl1-bos-wlz-1)이다.