

[AWS] 🐼 RDS 개념 & 아키텍처 정리 [이론편]

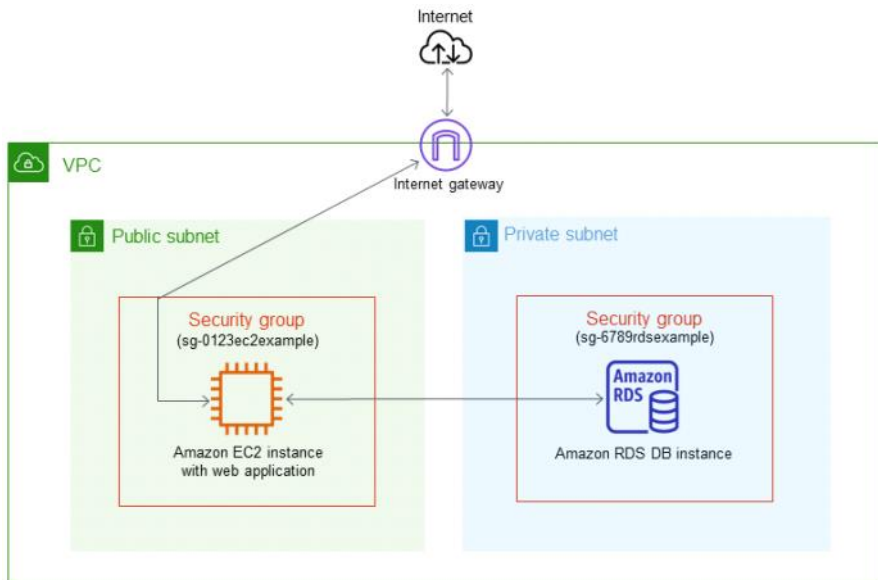
2023년 6월 14일 수요일 오후 2:24

RDS (Relational Database Service) 란?

AWS RDS란 관계형 데이터베이스를 간편하게 클라우드에서 설정, 운영, 확장이 가능하도록 지원하는 웹 서비스이다.

RDS는 MySQL이나 오라클 같은 데이터베이스의 설치, 모니터링, 백업, 알람 등 관리를 대신해주며, 하드웨어 프로비저닝, 데이터베이스 설정, 패치 및 백업과 같이 잦은 운영 작업을 자동화하여 비용 효율적이고 크기 조정 가능한 DB 서비스를 제공한다.

따라서 RDS를 통해 개발자는 DB 인프라를 구성하는데 힘을 들이지 않고, 개발이라는 본질적인 작업에 집중할 수 있게되는 장점이 있다.



물론 EC2 자체가 컴퓨터니까 EC2 인스턴스에 직접 데이터베이스를 설치해서 사용해도 된다.

하지만 AWS RDS는 EC2에 RDB(관계형 데이터베이스)를 직접 구축하여운영할 때보다 더 많은 부분을 자동으로 관리할 수 있어 편리하기 때문에 많이 애용된다.

Amazon EC2	AWS RDS
앱 최적화	앱 최적화
스케일링	스케일링
고 가용성 처리	고 가용성 처리
데이터베이스 백업	데이터베이스 백업
DB S/W 패치	DB S/W 패치
DB S/W 설치	DB S/W 설치
OS 패치	OS 패치
OS 설치	OS 설치
서버 설치 및 정비	서버 설치 및 정비
랙 & 스택	랙 & 스택
전력, 냉/난방 제어, 회선	전력, 냉/난방 제어, 회선

RDS의 데이터베이스의 제공 방식은 EC2와 비슷하다.

EC2 인스턴스를 생성해서 컴퓨팅을 사용하듯이, RDS 인스턴스를 생성해서 DB를 사용하는 원리이다.

하지만 EC2같이 유저가 시스템에 직접 로그인은 불가능하다.

그래서 RDS 인스턴스의 OS패치, 관리 등은 AWS가 전담 한다.

또한 RDS는 데이터베이스 모니터링 기능을 지원해준데, DB에서 발생하는 여러 로그 (Error Log, General Log 등)를 CloudWatch와 연동하여 확인도 가능하다.

RDS는 기본적으로 VPC 안에서 동작하며, 기본적으로 public IP를 부여하지 않아 외부에서 접근 불가능하다.

다만, 설정에 따라 public으로 오픈 가능하며 대신 로드밸런서 같이 DNS로만 접근이 가능하다.

VPC 내에서 동작하는 서비스이니 당연히 서브넷과 보안그룹 지정도 필요하다.

EC2 인스턴스를 생성할때 EC2 타입과 EBS 타입을 고르듯이,

RDS 인스턴스를 생성할때도 인스턴스 타입을 지정해주어야 하며, 스토리지는 EBS를 그대로 활용하기 때문에 EBS 타입의 선택도 필요하다.

그리고 RDS는 유동적으로 데이터 저장소 용량을 증설하는게 아닌, 생성시 EBS의 용량을 지정해서 생성한다. (추후에 용량 증설 가능)

cf) AWS 데이터베이스 서비스인 Amazon Aurora는 용량지정 X, 사용한만큼만 비용 지불

RDS의 가장 큰 특징은 **파라미터 그룹(Parameter Group)** 시스템인데, 이는 DB의 설정값들을 모아 그룹화한 개념이다.

이 DB 설정들을 모은 그룹을 각 DB 인스턴스에 적용시켜 DB의 설정값을 적용하는 시스템이다.

왜냐하면 위에서 말했듯이 직접 RDS 인스턴스 수정이 불가능 하기 때문에 이런 우회적인 방법으로 설정값을 세팅하는 원리이다.

Tip

AWS 프리티어로는 RDS를 12개월동안 단일 AZ, db.t2.micro 인스턴스를 750시간 무료 사용할 수 있다.

RDS 데이터베이스 종류

RDS에서는 Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle, MS SQLServer 총 6개의 데이터베이스 엔진 중에서 원하는 DBMS를 선택할 수 있다.

또한 AWS Databae Migration Services를 사용하여 기존 데이터베이스를 Amazon RDS로 손쉽게 마이그레이션 또는 복제 할 수 있다.



Info

Amazon Aurora는 MySQL 및 PostgreSQL호환 관계형 데이터베이스로, 오픈 소스 데이터베이스의 간편 성과 비용 효율성을 결합한 것이다. Amazon Aurora의 속도는 표준 MySQL 데이터베이스보다 5배, PostgreSQL 데이터베이스보다 3배 빠르다고 한다. 또한 상용 데이터베이스의 보안, 가용성 및 안전성을 1/10의 비용으로 제공하기도 한다.

PostgreSQL은 오픈 소스 관계형 데이터베이스 중 기능도 많고 성능도 좋은 거의 원탑의 데이터베이스이다.

MySQL은 세계적으로 가장 많이 사용되는 오픈 소스 관계형 데이터베이스다. Amazon RDS를 통해 비용 효율적이고 크기 조정이 가능한 MySQL 서버를 몇 분 안에 생성할 수 니다. 백업, 소프트웨어 패치, 모니터링, 확장 & 축소, 복제 같은 시간이 걸리는 작업은 모두 관리되므로 사용자는 개발에만 집중할 수 있다.

MariaDB는 MySQL을 개발한 개발자가 만든 오픈 소스 관계형 데이터베이스 이다. MySQL 업그레이드 판 이라고 봐도 된다. MySQL과 동일하게 Amazon RDS를 통해 효율적인 MariaDB 데이터베이스를 생성할 수 있고, 모든 시간 소모적 작업을 대신 처리해 준다.

Oracle는 오라클사의 유료 관계형 데이터베이스로서, RDS를 사용해 클라우드에서 손쉽게 배포, 설정, 운영 할 수 있는 완전 관리형 상용 데이터베이스이다. 다만 유료 데이터베이스 라 라이선스 비용이 든다.

SQL Server(MSSQL)는 Microsoft에서 개발한 관계형 데이터베이스 관리 시스템으로 Amazon RDS를 통해 손쉽게 배포, 운영, 확장이 가능하다.

RDS의 특징 & 기능

1. 관리 부담 감소

- 사용 편의성 : 몇 분 이내에 데이터베이스 인스턴스를 시작하고 애플리케이션을 연결할 수 있다. DB 파라미터 그룹을 사용하면 데이터베이스를 세부적으로 제어하고 튜닝할 수도 있다.
- 자동 소프트웨어 패치 : RDS는 최신 패치를 통해 배포를 지원하는 관계형 데이터베이스 소프트웨어가 최신 상태로 유지되도록 한다. 데이터베이스 인스턴스의 패치 여부와 시기를 선택적으로 제어할 수 있다.
- 모범 사례 권장 시스템 : RDS에서는 데이터베이스 인스턴스의 구성과 사용 지표를 분석하여 모범 사례 지침을 제공한다. 고객은 권장 조치를 즉시 수행하거나 다음 유지 관리 기간 에 수행하도록 예약하거나 완전히 무시할 수 있다.

2. 확장성

- 즉각적인 컴퓨팅 규모 조정 : 배포에 사용할 컴퓨팅 및 메모리 리소스를 최대 vCPU 32개와 RAM 244G의 범위 내에서 확장하거나 축소할 수 있다. 컴퓨팅 규모 조정 작업은 일반적으로 몇 분이면 완료된다
- 간편한 스토리지 규모 조정 : 스토리지에 대한 요구가 증가함에 따라 추가 스토리지를 프로비저닝할 수도 있다. 또한 DB 가동 중단 없이 즉시 스토리지 확장이 가능하다.
- **읽기 전용 복제본 시스템** : DB 인스턴스의 복제본을 하나 이상 생성하여 대량의 애플리케이션 읽기 트래픽을 처리할 수 있는 기능을 제공 한다.

4. 가용성 및 내구성

- 자동 백업 : RDS는 데이터베이스와 트랜잭션 로그를 백업하고 사용자가 지정한 보존 기간 동안 이를 모두 저장할 수 있다. 이를 통해 데이터베이스를 보존 기간 중 어느 시점(초 단위)으로나 복원할 수 있다(최근 5분 전까지 가능). 자동 백업 보존 기간은 최대 35일로 구성할 수 있다.
- 데이터베이스 스냅샷 : 사용자는 원하는 경우 언제든지 데이터베이스 스냅샷으로 새 RDB 인스턴스를 생성할 수 있다.
- **다중 AZ 구성 배포** : 다중 AZ 구성은 현재 서비스되는 RDB에 문제가 되어도 다른 AZ에 있는 RDB로 빠르게 장애 복구를 할 수 있다. 가용성 및 내구성을 높여주므로 프로덕션 데이터베이스 워크로드에 적합하다.
- 자동 호스트 교체 : 하드웨어에 장애가 발생할 경우, 배포를 지원하는 컴퓨팅 인스턴스를 자동으로 교체한다.

5. 보안

- 저장 데이터 및 전송 데이터 암호화 : RDS를 사용하면 사용자가 AWS Key Management Service(KMS)를 통해 관리하는 키를 사용해 데이터베이스를 암호화할 수 있다. 자동 백업, 읽기 전용 복제본 및 스냅샷과 마찬가지로 기본 스토리지에 저장된 데이터가 암호화 된다.
- 네트워크 격리 : VPC에서 데이터베이스 인스턴스를 실행하여 방화벽 설정을 구성하고 데이터베이스 인스턴스에 대한 네트워크 액세스를 제어할 수 있다.
- IAM 리소스 수준 권한 : RDS는 IAM과 통합되어 사용자 및 그룹이 특정 Amazon RDS 리소스(데이터베이스 인스턴스, 스냅샷, 파라미터 그룹 및 옵션 그룹)에서 수행할 수 있는 작업을 제어하는 기능을 제공한다. 또한 RDS 리소스에 태그를 지정하고, IAM 사용자 및 그룹이 태그가 동일하고 연관된 값을 가진 리소스 그룹에서 수행할 수 있는 작업을 제어할 수 있다.
(예를 들어, 개발자는 "개발" 데이터베이스 인스턴스를 수정할 수 있고, "프로덕션" 데이터베이스 인스턴스는 데이터베이스 관리자만 수정할 수 있도록 IAM 규칙을 구성할 수 있다.)

6. 관리 효율성

- 모니터링 및 지표 : RDS는 추가 비용 없이 데이터베이스 인스턴스에 대한 **Amazon CloudWatch**와 연동해 지표를 제공한다. RDS 관리 콘솔을 사용하면 컴퓨팅/메모리/스토리지 용량 사용률, I/O 작업, 인스턴스 연결 등 주요 작업 지표를 보고 성능 문제를 신속하게 감지할 수 있는 편리한 도구인 성능 개선 도우미에 액세스할 수 있다.
- 이벤트 알림 : Amazon SNS를 통해 데이터베이스 이벤트를 이메일이나 SMS 텍스트 메시지로 알려줄 수 있다.

7. 비용 효율성

- 사용한 만큼만 비용 지불
- 예약 인스턴스 : Amazon RDS 예약 인스턴스는 1년 또는 3년 약정 기간 동안 DB 인스턴스를 예약할 수 있는 옵션을 제공하므로 온디맨드 인스턴스 요금과 비교하여 DB 인스턴스의 시간당 요금을 대폭 할인받을 수 있다.
- 인스턴스 중지 및 시작 : Amazon RDS를 활용하면 한 번에 최대 7일까지 데이터베이스 인스턴스를 쉽게 중지했다 시작할 수 있다. 따라서 데이터베이스를 상시 구동하지 않아도 되는 개발 및 테스트 작업에 데이터베이스를 쉽고 저렴하게 이용할 수 있다.

RDS 백업 시스템

자동 백업

자동 백업(Automated Backups) 줄여서 AB는 매일마다 스냅샷과 트랜잭션 로그를 참고하여 자동으로 백업을 해준다.

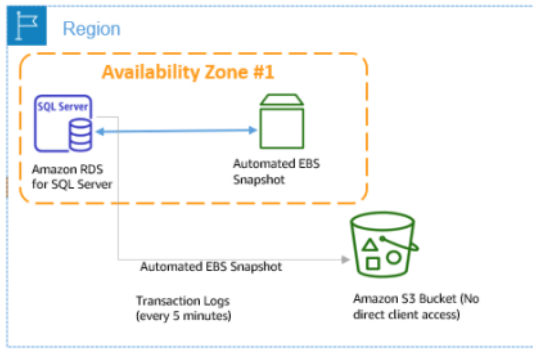
RDS에서는 디폴트로 AB 기능이 설정되어 있다.

그리고 AB를 통해 데이터베이스를 Retention Period(1~35일) 안의 과거 특정 시간으로 되돌아갈 수도 있다.

단, 롤백 동작은 과거 상태로 그대로 돌아가는게 아닌, 다른 DB 인스턴스를 새로 생성해서 스냅샷을 적용 시키는 형식임을 유의하자.

RDB 백업 정보는 S3에 저장되며, AB동안 약간의 I/O suspension(딜레이)이 존재할 수 있다.

그나마 Multi-az로 하면 Standby를 통해 백업을 수행하기 때문에 딜레이가 덜하다.



수동 백업 (DB 스냅샷)

AB(자동 백업)이 자동으로 스냅샷을 떠서 백업하는 것이라면, 수동 백업은 유저 혹은 다른 프로세스로부터 요청에 따라 만들어지는 DB 스냅샷이다.

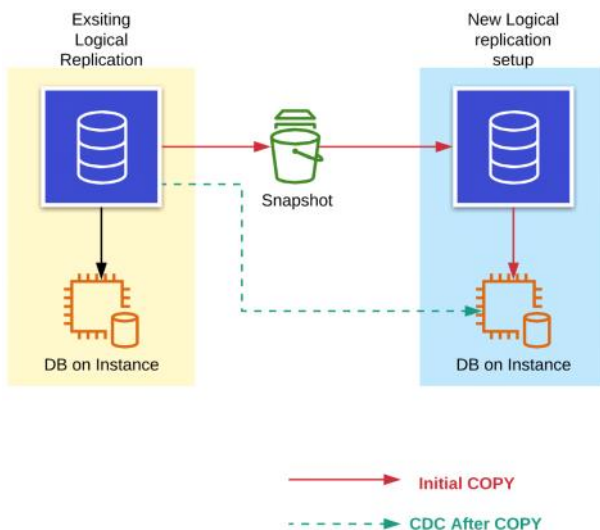
즉, EC2 스냅샷을 뜻듯이 사용자에게 의해 수동적으로 진행되는 백업이다.

만약 원본 RDS를 삭제한다고 하더라도, 스냅샷은 S3 버킷에 그대로 존재한다. 따라서 스냅샷만으로 RDS 인스턴스를 복원시킬 수 있다.

반대로 AB 백업 기능은 인스턴스를 삭제할 때 스냅샷도 모두 없어진다는 특징이 있다.

이 역시 스냅샷의 복구는 항상 새로운 DB Instance를 생성 하여 수행되며,

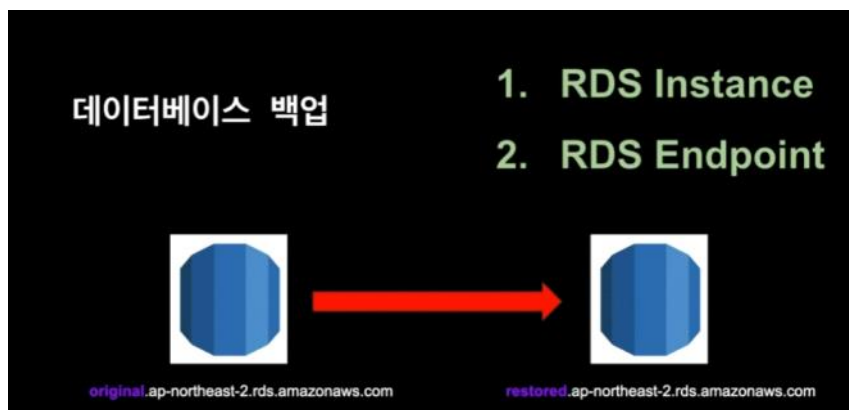
만약 데이터베이스를 복구해야 한다면, 새로운 DB를 만들고 기존 DB의 연결을 끊고 새로 만든 DB에 연결해주는 작업이 필요하다.



RDS 백업 복원시 도메인 변경

원본 RDS 인스턴스를 가지고 새로운 DB를 복원시 새로운 인스턴스와 Endpoint가 생성된다.

원본 DNS는 앞에 original인 반면, 복원된 것은 앞에 restored가 붙게된다.



RDS 인증 여러가지 방법

데이터베이스 인증

데이터베이스 인증 옵션 정보

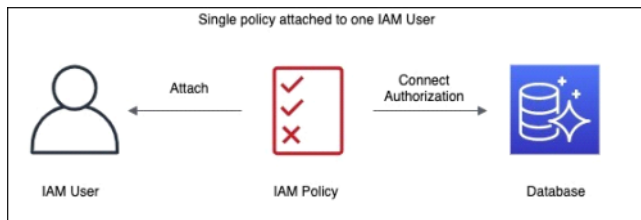
- ☒ 암호 인증
데이터베이스 암호를 사용하여 인증합니다.
- ☐ 암호 및 IAM 데이터베이스 인증
AWS IAM 사용자 및 역할을 통해 데이터베이스 암호와 사용자 자격 증명을 사용하여 인증합니다.
- ☐ 암호 및 Kerberos 인증
권한이 부여된 사용자가 Kerberos 인증을 사용하여 이 DB 인스턴스에서 인증하도록 허용하려는 디렉토리를 선택합니다

1. 전통적인 유저/패스워드 방식

- AWS Secret Manager(기타 유저 패스워드 인증 관리 서비스)와 연동하여 자동 로테이션 가능하게 할 수 있다.
- 일정주기마다 자동으로 패스워드를 변경해주시기도 하고 다른 서비스들이 RDS나 다른 서비스에 접근할 때 패스워드를 하드코딩 할 필요없게 해준다.

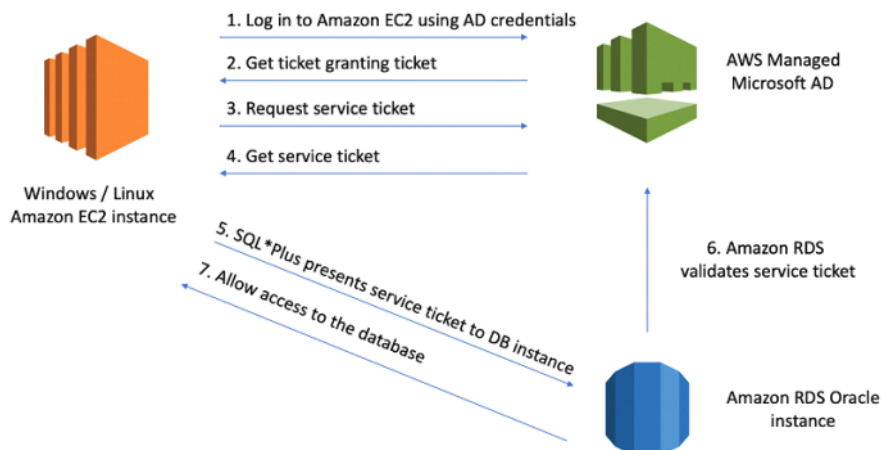
2. IAM DB 인증

- 데이터베이스를 IAM 유저 크레덴셜/Role을 통해 관리 가능



3. Kerberos 인증

- 마이크로소프트의 액티브 디렉토리 안에 포함되어있는 프로토콜
- 네트워크 상에 직접 아이디/패스워드를 입력하지 않아도 되게끔 만들어주는 대칭키 기반의 프로토콜



RDS 가격 모델

- 컴퓨팅 파워 + 스토리지 용량 + 백업 용량 + 네트워크 비용을 모두 합쳐서 지불
- EC2 인스턴스를 구매해 사용했던 것처럼, RDS에는 Reserved Instance(RI)를 구매해서 사용하는 시스템이다.

- EC2와 마찬가지로 일정기간을 계약하여 저렴한 가격에 서비스를 이용
- 3년 약정이면 거의 56%이상 저렴
- 단, 인스턴스 타입 변경이 쉽지않기 때문에 수요에 대한 신중하게 결정

Purchase Reserved Instances

☒ Only show offerings that reserve capacity

Availability

Platform Linux/UN...

Zone Any

Tenancy Default

Offering Class Converti...

Instance Type c4.large

Term Any

Payment Option Any

Search

Seller	Term	Effective Rate	Uprfront Price	Hourly Rate	Availability Zone	Payment Option	Offering Class	Quantity Available	Desired Quantity	
AWS	36 months	\$0.070	\$0.00	\$0.070	us-east-1a	No Uprfront	convertible	Unlimited	1	Add to Cart
AWS	36 months	\$0.070	\$0.00	\$0.070	us-east-1b	No Uprfront	convertible	Unlimited	1	Add to Cart
AWS	36 months	\$0.070	\$0.00	\$0.070	us-east-1c	No Uprfront	convertible	Unlimited	1	Add to Cart
AWS	36 months	\$0.070	\$0.00	\$0.070	us-east-1e	No Uprfront	convertible	Unlimited	1	Add to Cart
AWS	36 months	\$0.060	\$797.00	\$0.030	us-east-1a	Partial Uprfront	convertible	Unlimited	1	Add to Cart
AWS	36 months	\$0.060	\$797.00	\$0.030	us-east-1b	Partial Uprfront	convertible	Unlimited	1	Add to Cart
AWS	36 months	\$0.060	\$797.00	\$0.030	us-east-1c	Partial Uprfront	convertible	Unlimited	1	Add to Cart
AWS	36 months	\$0.060	\$797.00	\$0.030	us-east-1e	Partial Uprfront	convertible	Unlimited	1	Add to Cart

You currently have no items in your cart.















Cancel

View Cart

RDS 요금 정책

요금 발생 유형	요금 정책
온디맨드	인스턴스에서 실행한 컴퓨팅 파워에 대해서 시간당 요금 지불
예약	온디맨드보다 저렴. 1년~3년 약정 기간에 DB 인스턴스 예약
DB Storage	범용(SSD) 스토리지 : 프로비저닝한 스토리지에 대해 요금 청구 (I/O는 요금 청구 없음) 프로비저닝 IOPS(SSD) 스토리지 : DB에 필요한 I/O 용량을 지정하거나 프로비저닝 가능. 프로비저닝한 처리량 및 스토리지에 대해 비용 청구(I/O 요금청구 없음)
백업 스토리지	리전별로 할당. 리전 전체 데이터베이스 스토리지의 최대 100프로에 해당하는 백업 스토리지는 추가비용 없음 DB 인스턴스 종료 후에 백업스토리지에 월별 GiB당 요금청구 추가 백업 스토리지에는 월별 GiB당 요금 청구
스냅샷 내보내기	스냅샷 용량에 따라 요금 지불. 동일한 스냅샷에서 추가로 데이터 내보내는건 요금 없음. RDS내에서 데이터 내보내거나 S3로 내보냄
데이터 전송	동일한 가용 영역에서 Amazon RDS와 Amazon EC2 인스턴스 간에 전송된 데이터는 무료 같은 리전의 서로 다른 가용 영역에서 Amazon EC2 인스턴스와 Amazon RDS DB 인스턴스 간에 전송된 데이터의 경우, 양쪽 모두에 Amazon EC2 리전 데이터 전송 요금이 청구

그 외의 AWS 데이터베이스 서비스

								
	관계형 DB	키-값	문서	인-메모리	그래프	시계열	원장	
	 Amazon RDS for MySQL, PostgreSQL, Oracle, SQL Server	 Amazon Aurora for MySQL & PostgreSQL	 Amazon DynamoDB	 Amazon DocumentDB with MongoDB compatibility	 Amazon ElastiCache for Redis & Memcached	 Amazon Neptune	 Amazon Timestream	 Amazon QLDB
주요 특징	참조 무결성, ACID 트랜잭션, Schema-on-Write	Key lookup 빠른 응답시간, 빠른 처리량	각종 문서 저장 인덱싱	1/1000 초 미만의 응답시간. 메모리 스토어	쉽고 빠르게 데이터 관계 생성 및 탐색	시간에 따른 순차 데이터 수집, 저장 및 처리	모든 변경에 대한 완전, 불변, 검증 가능한 기록	
요구 사항	웹 서비스, 상품 카탈로그, ERP, CRM	실시간 입찰 장바구니 OLTP	콘텐츠 관리, 개인에 맞춰 편집된 환경 관리	리더 보드, 실시간 분석, 캐싱	소셜 네트워킹 추천 서비스	사물 인터넷, 시계열 로그 분석	헬스케어, 금융 정보, 공급망 관리, 판매 이력	

AWS 데이터베이스 서비스는 RDS 이외에 다양한 데이터베이스 서비스를 제공한다

- Amazon DocumentDB : 문서 기반 데이터베이스를 제공한다.
- Amazon Neptune : 그래프 데이터베이스를 제공한다.
- Amazon Timestream : 시계열 데이터베이스를 제공한다.
- Amazon Quantum Ledger Database : 장부 데이터베이스를 제공한다.

- Amazon ElastiCache : 인 메모리 데이터베이스이다. 메모리를 활용하여 처리 속도가 빠르며 Redis 용과 Memcached용이 있다.

RDS 구성 아키텍처

데이터베이스는 관리가 중요하다.

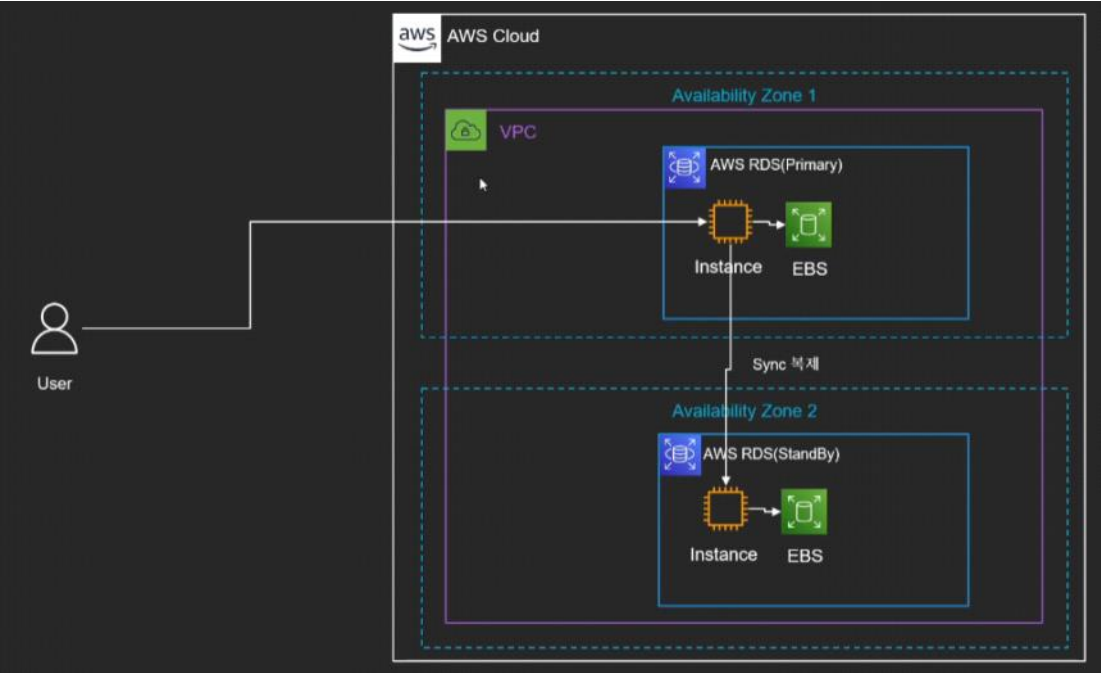
만일 어느 데이터베이스가 장애가 나면 재빨리 다른 데이터베이스로 옮겨 서비스를 지속하는 등 고가용적인 관리가 필요하다.

또한 트래픽이 몰려 데이터베이스 서버가 터지는걸 대비해 분산 기법도 이용해야 한다.

AWS RDS는 데이터베이스를 보다 효율적으로 관리할수 있게 하는 인프라 아키텍처 방법 3가지를 제공한다.

	Multi-AZ	Multi-Region	Read Replica
목적	고가용성	DR/로컬 퍼포먼스	확장성/성능
복제방식	Sync	Async	Async
액티브	Primary DB만 읽기/쓰기 가능	Read만 가능	Read만 가능
백업	자동백업(Standby기준)	자동 백업 가능	기본적으로 백업 x
엔진 업데이트	Primary만 업데이트	각 리전별로 다른 업데이트	DB별로 다른 업데이트
FailOver	자동으로 Standby로 Failover	수동으로 Failover	수동으로 Failover

RDS Multi AZ 구조



RDS 멀티 AZ는 위 사진에서 보듯이, 두개 이상의 AZ에 걸쳐 데이터베이스를 구축하고 원본과 다른 DB(Standby)를 자동으로 동기화(Sync)하는 구조이다.

Tip

Standby 란?

돌발 사태로 예정된 기능이 이뤄지지 못할 경우를 대비한 '임시'를 뜻함

Multi AZ는 AWS에 의해서 자동으로 관리가 이루어 진다.

만약 RDS DB를 만들고, DB에 특정 레코드를 insert 할 시, 다른 AZ(Availability Zone)에 똑같은 복제본이 만들어지게 된다.

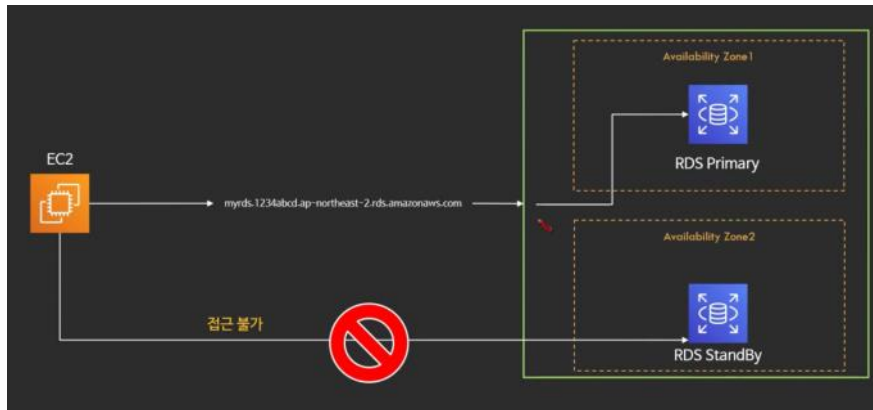
그리고 유사시 우리가 현재 사용하고 있는 메인 DB에 문제가 생길 경우 RDS는 이를 즉시 발견하고 다른 AZ에 만들어진 복제본을 원본 DB를 승격시켜 그대로 사용하게 된다. 이를 Disaster Recovery(재해 복구)라고 부른다.

Multi AZ 재해 복구 원리

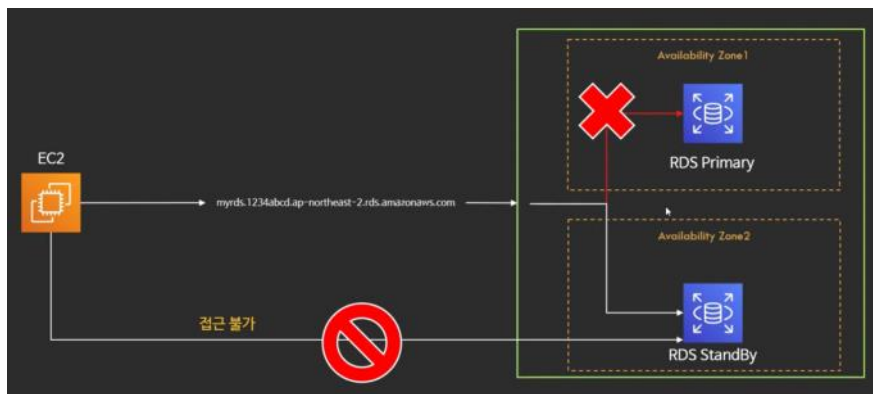
일반적인 경우 DNS 주소로 RDS Primary 인스턴스에 접속하여 사용한다.

그런데 아래 사진에서 보듯이 기본적으로 Standby DB는 접근 불가능하다. (DNS가 부여되지 않음)

장애 복구용 백업 인스턴스 이기 때문에 Stanby DB는 숨겨져 있기 때문이다.



그러다 만약에 **RDS Primary에 장애가 발생하면**, 다음 과 같이 장애 복구가 이루어 진다.



1. RDS Primary에 문제가 생기면, 자동으로 DNS를 Standby 인스턴스로 연동을 해준다.
2. 그리고 기존에 RDS Primary와의 연결을 끊어준다.
3. 이것을 **fail over(fail났을 때 복구하는 과정)**라고 한다.
4. EC2 입장에서는 동일한 address로 바라보고 있기 때문에 장애를 느끼지 못하게 된다.
5. 굉장히 빠른 속도로 장애복구 가능 O

Multi AZ 용도

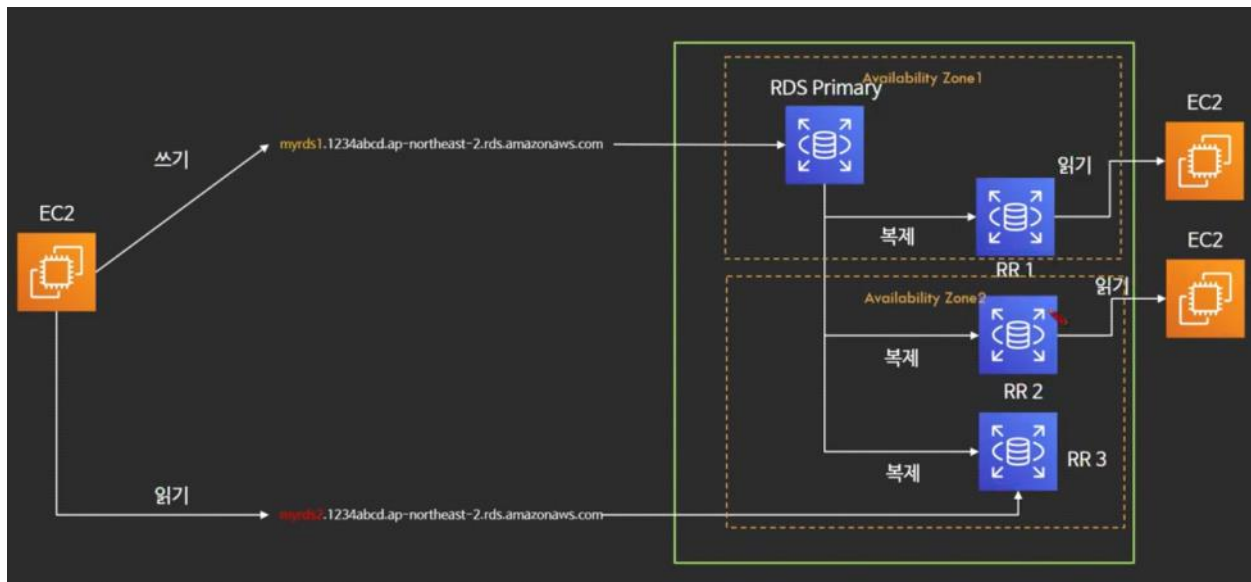
멀티 AZ 구조의 단점은 예비용 인스턴스가 하나 더 들고 있다는 것이기 때문에 비용이 두배가 든다는 점이다.

그리고 예비용 만들어진 인스턴스는 활용을 안하고 그저 만일을 대비해서 그저 대기 상태로 된다.

따라서 멀티 AZ 구조는 **퍼포먼스의 상승 효과가 아닌 안정성을 위한 서비스** 이다.

그래서 Multi AZ는 복제본을 만든다고 해서 성능이 더 좋아지는 건 아니지만, 만약 성능개선이 주목적이라면, 이 다음에 배울 Read Replicas를 이용해야 한다.

Read Replica (읽기 전용 복제본)



Read Replica(읽기 전용 복제본)은 위 그림에서 볼수 있듯이, RDS Primary를 복제해서 DB 쓰기는 RDS Primary에서 처리하고, DB 읽기는 복제한 복제본에서 처리하는 구조이다.

그래서 읽기 전용 복제본이라는 이름이다.

이렇게 쓰기와 읽기를 나눠놓은 이유는, 만일 서비스에서 DB 읽기 위주의 작업이 많은 경우 Read Replica를 여러 개 만들어서 부하를 분산할 수 있기 때문이다.

즉, 쓰기 작업은 마스터 DB 인스턴스에 하고 읽기 작업은 Read Replica에 할당하면 마스터 DB 인스턴스의 부하를 줄일 수 있는 원리이다.

Read Replica의 복제본들은 같은 AZ에 있을수도있고 다른 AZ에 있을수도 있고 심지어 다른 리전에 존재할 수도 있다.

그리고 Read Replica는 멀티 AZ와는 달리 각각의 복제본 인스턴스에 DNS가 각각 부여되어서 접근이 가능하다. (당연히 읽기 동작을 하려면 접근해야 되기 때문에)

이렇게 데이터베이스를 복제하여 쓰기와 읽기를 분별하여 트래픽을 분산하는 기술을 데이터베이스 Replication 이라고 한다.

흔히 서버를 이중화 하는 것처럼 데이터베이스를 이중화 한다고 이해하면 된다.

데이터베이스 Replication

Replication이란 백업과 성능 향상을 위해서 데이터베이스를 여러 대의 서버에 복제하는 행위를 뜻한다.

원본 데이터가 위치하는 서버를 마스터라고 하고, 그 원본을 복제한 서버를 슬레이브라고 칭한다.

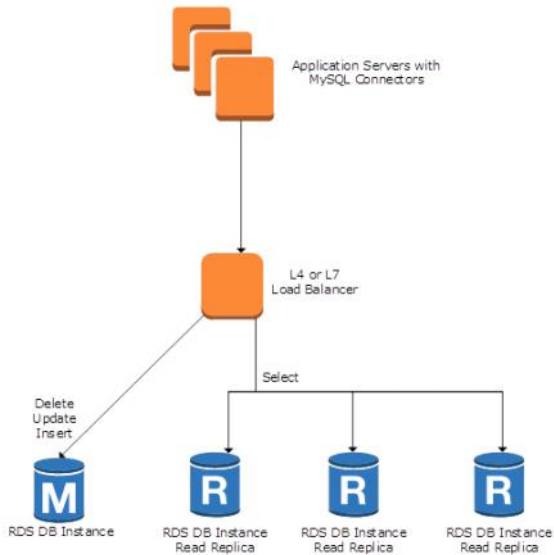
데이터베이스의 작업은 읽기와 쓰기로 구분할 수 있다.

SQL로 말하면 읽기는 SELECT 구문이고, 쓰기는 INSERT, UPDATE, DELETE 트랜잭션이 된다.

쓰기 작업같은 경우 저장된 데이터가 변경되기 때문에, 만일 복제된 서버에 쓰기 작업을 맡기게 된다면 복제된 서버들 간에 동일한 형태를 유지하는데 많은 비용이 든다.

그래서 보통 한대의 서버에만 쓰기 작업을 하고, 그 서버의 데이터를 복제해서 여러 대의 슬레이브 서버를 만든 후에 슬레이브에서는 읽기 작업만을 수행하게 구성한다.

따라서 Read Replica는 RDS DB 인스턴스의 읽기 전용 인스턴스가 된다.



위와 같이 Delete, Update, Insert 작업이 실행될 땐 RDS DB Instance로 요청이 들어가고, Select 작업을 수행해야 한다면 Read Replica에 요청이 들어가게 되는 것이다.

이를 통해 읽기 위주의 작업이 많은 경우 부하를 분산할 수 있게 된다.

따라서 Read Replica는 성능 극대화를 위해 존재하기에, Read heavy work가 많을 시 서버 다운이 일어날 수 있기 때문에 Read Replica를 사용해야 한다.

Read Replica vs Multi-AZ

둘다 RDS Primary 인스턴트를 복제한다는 점은 같지만, 앞서 Multi-AZ 복제와 혼동하면 안된다.

1. 복제의 목적

Multi-AZ 복제는 서비스가 항상 가동해야 하는 가용성을 위한 것이지, 부하 분산을 통한 성능 향상이 목적이 아니다.

Read Replica 기능은 Disaster Recovery(재해 복구) 용도가 아니다. 즉, 안정성을 위한 서비스가 아닌 퍼포먼스를 위한 서비스이다.

Multi-AZ 복제본 같은 경우는 그저 안정성을 위해 아무것도 하지않는 생 인스턴스를 미리 준비해 가동하는 것이지만, Read Replica는 실제로 다른 EC2들과 상호작용을 하기 때문이다.

2. 복구 방법

그리고 Read Replica는 Multi-AZ 와 달리 fail over(자동복구)가 불가능하다.

만약 RDS Primary가 고장났을경우, 관리자가 직접 수동으로 쓰기 DNS를 복제본에 연결해 복구를 해줘야 한다.

3. 복제 데이터 일관성

Multi-AZ 복제는 쓰기 작업을 실시한 즉시 자동으로 예비 인스턴스에 복제된다. 따라서 두 DB 인스턴스의 데이터가 항상 일치하는 것을 보장한다.

그러나, 복제된 예비 인스턴스에서는 읽기 작업을 할 수 없다.

Read Replica의 경우 쓰기 작업을 실시하면 약간의 시간차를 가지고 데이터가 복제되게 된다. 따라서 데이터가 일치하지 않을 수 있다.

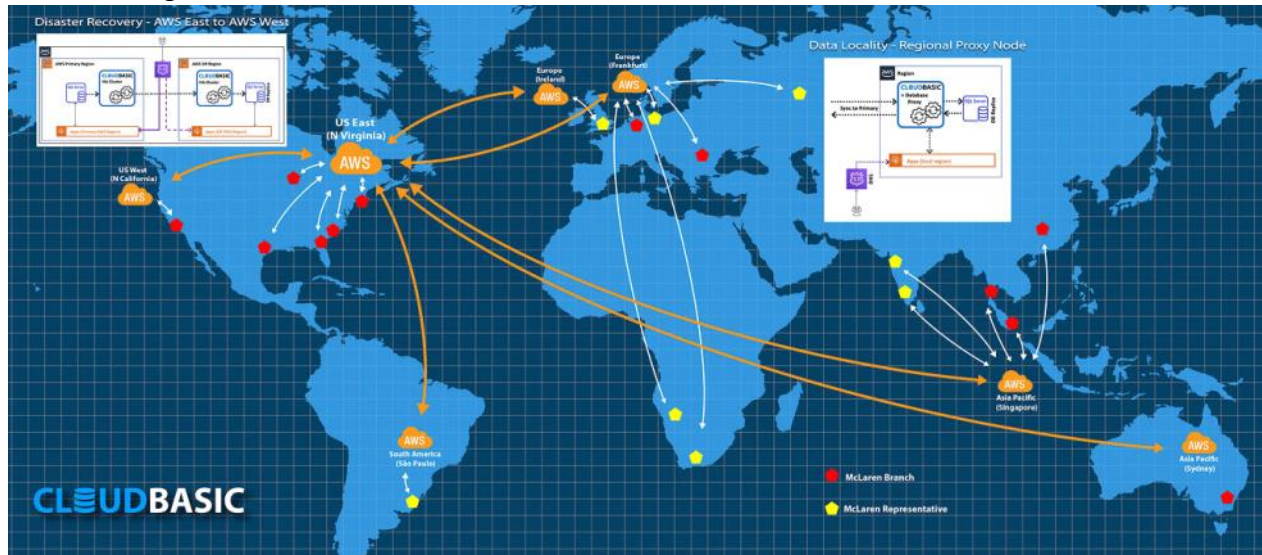
Tip

만일 S3를 배웠다면, PUT동작과 UPDATE/DELETE 동작간의 데이터 일관성 최종 일관성과 같은 원리이다.

Read Replica (읽기 전용 복제본) 정리

- 총 **5개**까지 생성 가능
- 복제본의 복제본도 생성 가능
- 각각의 복제본은 고유 DNS가 할당 됨 → 접근 가능
- 원본 DB의 장애 발생 시 **수동으로 DNS 변경이 필요함**
- 복제본 자체에 Multi-AZ 설정하여 둘이 조합하여 인프라 구성 가능
- 단, 자동 백업이 활성화 되어있어야 읽기 전용 복제본 생성 가능
- 각 DB간의 데이터나 엔진 버전이 다를 수 있음 (멀티 AZ는 완전히 똑같은 DB 인스턴스가 생성되는 것이기 때문에 버전이 같지만, 이견 아니다)

RDS Multi Region



- 다른 리전에 지속적으로 동기화 시키는 DB 클러스터 생성 (Async 복제)
- 예를 들어 유럽 국가에 서비스를 한다고 가정하자.
서울에 마스터 리전을 두고 유럽에 Read Replica를 둔다면 유럽 현지에서는 Read Replica로 읽기 때문에 latency(지연시간)가 줄어든다.
우리나라에서는 읽기 전용 복제본에 데이터 복제만 해주면 된다.
- RDS 멀티 리전은 주로 로컬 퍼포먼스 또는 DR(disaster recovery)시나리오로 활용
- 로컬 퍼포먼스 : 그 지역에 있는 사람들이 빨리 읽을 수 있게 해줌
- DR(disaster recovery)시나리오 : 만약 서울에 폭탄이 터져서 리전이 fail이 된다면 다른리전을 마스터로 만드는 구성
- 각 리전별로 자동 백업 가능
- 리전별로 Multi-AZ 가능