

Module 5

인스턴스 스토어 (Amazon Elastic Block Amazon EBS)

- 블록 수준 스토리지
- 인스턴스와 수명이 동일함 → 인스턴스 종료 시 스토어 데이터 손실

Amazon Elastic Block Store (Amazon EBS)

- 인스턴스에서 사용할 수 있는 **블록 수준 스토리지 볼륨 제공** 서비스
- 인스턴스를 중지, 종료하더라도 연결된 EBS 볼륨의 모든 데이터 사용 가능

Amazon EBS 스냅샷

- **증분 백업** - 처음 백업 시 모든 데이터 복사, 이후로는 가장 최근의 스냅샷 이후 변경된 블록만 저장
- **전체백업** : 가장 최근 백업하지 않았지만 이후 변경되지 않은 데이터도 포함

객체 스토리지

데이터

이미지, 동영상, 텍스트 문서 또는 기타 유형의 파일

메타데이터

데이터의 내용, 사용 방법, 객체 크기

키

고유 식별자

Amazon Simple Storage Service (Amazon S3)

- 데이터 : **객체** 형태로 **버킷** 에 저장 (버킷 : 파일 디렉토리)
- 데이터를 3개 이상의 시설에 저장 → 데이터 손실에 대한 피해가 적음
- 파일 업로드 시 **권한 설정** → 파일에 대한 표시 여부 및 액세스 제어 가능

Amazon S3 스토리지 클래스

- 사용한 만큼만 비용 지불
- 고려 요소 **데이터 검색 빈도 필요 데이터 가용성**

1. S3 Standard

- 자주 액세스하는 데이터용으로 설계
- 최소 3개의 가용 영역에 데이터 저장

객체에 대한 **고가용성** 제공 → 웹사이트, 콘텐츠 배포, 데이터 분석 등 광범위하게 적용

비용이 높음

2. S3 Standard-Infrequent Access (S3 Standard-IA)

- 자주 액세스하지 않는 데이터
- S3 Standard와 유사하지만 저렴하고 검색 가격은 높음

자주 액세스하지 않지만 **필요에 따라 고가용성이 요구되는 데이터**에 이상적

Standard와 마찬가지로 **3개의 가용 영역**에 데이터 저장

3. S3 One zone-Infrequent Access (S3 One Zone-IA)

- **단일 가용 영역**에 데이터 저장
- S3 Standard-IA 보다 낮은 스토리지 가격
- 스토리지 비용을 절감하려는 경우 가용 영역 장애가 발생했을 때 데이터를 쉽게 재현

4. S3 Intelligent-Tiering

- 액세스 패턴을 알 수 없거나 **자주 변화하는 데이터**에 이상적
- 객체당 소량의 월별 모니터링 및 **자동화** 요금 부과

연속으로 액세스하지 않으면 자동으로 **S3 Standard-IA**로 이동

자주 사용하지 않는 액세스 계층에 액세스하면 **S3 Standard**로 이동

5. Glacier

- **데이터 보관용**으로 설계된 저비용 스토리지
- 객체를 몇 분에서 몇 시간 이내에 검색

고객 레코드, 이전 사진, 비디오 파일 저장

6. Glacier Deep Archive

- 보관에 이상적인 가장 저렴한 객체 스토리지 클래스
- 객체를 12시간 이내에 검색

Glacier 보다 더 빠른 검색 가능

Amazon Elastic File System(Amazon EFS)

- **관리형 파일 시스템**

- 여러 인스턴스가 EFS의 데이터에 동시 액세스 가능
- **Amazon EBS 볼륨** : 가용 영역 수준의 리소스
- EC2를 EBS에 연결하려면 같은 AZ에 있어야 함 (HDD 역할)
- EFS는 필요에 따라서 확장 / 축소 가능, EBS는 그렇지 않음

파일 스토리지

- 여러 클라이언스 (사용자, 애플리케이션, 서버)가 공유 파일 폴더에 저장된 데이터에 액세스 가능
- 블록 스토리지를 로컬 파일 시스템과 함께 파일 구성, 파일 경로를 통해 액세스
- 파일 스토리지는 많은 서비스 / 리소스가 동일한 데이터에 액세스할 때 좋음

EFS VS EBS

EBS

- **단일 가용 영역** 에 데이터 저장
- EC2와 EBS 모두 **동일한 가용 영역에 상주** 필요

EFS

- 리전별 서비스, **여러 가용 영역** 에 데이터 저장
- 중복 스토리지 사용 → 파일 시스템이 위치한 **리전의 모든 가용 영역**에서 데이터에 액세스 가능
- 온프레미스 서버 : AWS Direct Connect로 액세스 가능

관계형 데이터베이스

- SQL을 사용하여 데이터 저장, 쿼리 → 일관성, 확장 가능성

Amazon RDS

- AWS 클라우드에서 관계형 DB 실행하는 서비스
- 하드웨어 프로비저닝, 데이터베이스 설정, 패치 적용 백업 자동화

RDS DB 엔진

1. Amazon Aurora
 - 엔터프라이즈급 관계형 DB
 - MySQL, PostgreSQL 과 호환
 - 리소스 안정성, 가용성 유지 / 불필요한 I/O 작업을 줄여 비용 절감
2. PostgreSQL
3. MySQL
4. Maria DB
5. Oracle Database
6. Microsoft SQL Server

Amazon DynamoDB

- 서버리스 데이터베이스
- 데이터 중복 저장
- SQL 사용 X, 비관계형 DB
- 유동적이고 빠른 속도로 액세스 할 때 이용
- 용량 변화에 맞춰 자동으로 크기를 조정하면서도 일관된 성능 유지

비관계형 DB

- NoSQL 데이터베이스
- `key - value` 로 접근

Amazon Redshift

- 빅데이터 분석 에 사용할 수 있는 데이터 웨어하우징 서비스
- 원본에서 데이터 수집 → 데이터 관계 및 추세 파악

AWS Database Migration Service

- 관계 / 비관계형 DB 및 기타 유형의 데이터 저장소 마이그레이션
- 원본 - 대상 DB 간 데이터 이동
- 마이그레이션 동안 DB 작동 → 애플리케이션 가동 중지 시간 감소

Example

- 프로덕션 사용자에게 영향 없이 애플리케이션 테스트 지원
- 여러 DB → 단일 DB 결합
- 데이터 진행 중 복제본 → 원본 전송

추가 데이터베이스 서비스

Amazon DocumentDB

- Dynamo DB 이상의 속성이 필요한 경우 사용
- MongoDB 워크로드를 지원하는 문서 DB 서비스
- 콘텐츠 관리 시스템에 적합 (사용자 프로필 등)

Amazon Neptune

- 소셜 네트워킹 및 추천 엔진에 사용
- 불변성을 요구하는 은행 또는 재무기록

Amazon QLDB (Quantum Ledger Database)

- 변경 불가능한 기록 시스템
- 속도 개선 가능

Amazon ElasticCache

- 캐싱 계층 제공 : **Memcached** 및 **Redis** 유형
- 읽기 시간 향상

Amazon Managed Blockchain

- 오픈소스 프레임워크 사용 → 블록체인 네트워크 생성 및 관리
- **분산형 원장 시스템**

Aamazon DynamoDb Acceleration

- DynamoDB 용 메모리 캐시 → 응답 시간 향상