* AWS
* Storage

**Amazon EBS (Elastic Block Store)**

EBS 볼륨은 EC2 Instance를 위해 안정적이고 분리 가능한 블록 수준 Storage를 제공 하며 EBS 볼륨이 Instance에 직접 연결되어 짧은 지연시간을 제공한다.

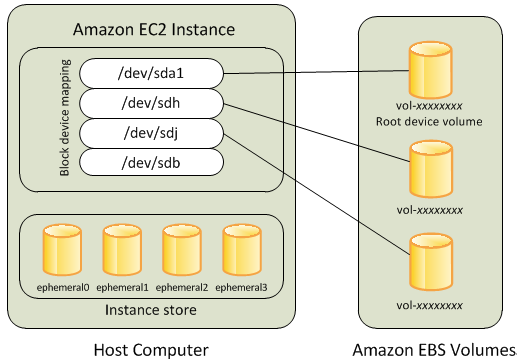
종류: SSD기반, HDD기반

SSD 기반으로는 General Purpose SSD, Provisioned IOPS SSD 가 존재하고 HDD기반으로는 Throughput Optimized HDD, Cold HDD가 존재한다.

성능의 경우 SSD기반은 IOPS를 기반으로 하며, HDD는 Throughput 기반으로 측정 한다.

\*참고 - Instance Store와의 차이점

Instance Store는 실제 AWS 물리적인 서버의 디스크 공간이다. EC2는 Stop, Start시 물리적인 서버의 Host 위치가 변경 되는데, 이렇게 되면 Instance Store에 저장된 데이터는 날라간 것이다. 따라서 휘발성이라 표현하며, 지속 보관이 필요한 데이터는 EBS볼륨에 저장을 하여야 한다. 그럼에도 불구하고 Instance Store는 실제 EC2가 올라가는 물리서버의 물리적인 디스크로 EBS볼륨보다도 속도가 더 빠른 저장공간이다.



EBS도 결국 EC2 인스턴스에 연결이 되어 있는 볼륨으로 다른 트래픽이 증가하면 I/O 성능이 떨어진다. EBS 최적화 인스턴스 (EBS Optimized Instances)는 이러한 트래픽 간 경합을 최소화 하기 위해서 전용 대역폭을 제공하는 것이다. 그렇게 함으로써 EBS 성능을 극대화 최적화 시키고자 하는 옵션이다.

**Amazon EFS (Elastic File System)**

여러 EC2 인스턴스에서 동일한 파일 시스템을 사용해야 하는 경우 사용한다.

\*참고 – S3와의 차이점

S3는 객체 스토어 시스템이므로 파일 내 문자를 하나 수정하더라도 전체 파일을 덮어쓰는 방식이다. EFS는 파일스토리지 시스템으로 파일 내 문자 블록만 수정하므로 다양한 크기의 파일을 높은 처리량으로 변경하려면 EFS방식이 우수하다.

* Compute

**EC2 (Elastic Compute Cloud)**

-Instance Type

패밀리 이름+세대 번호+인스턴스 크기로 구성된다. Ex) m2.large, m5.xlarge

m2, m5 등 숫자는 세대 번호를 의미 하며 최신 세대 일수록 숫자가 높다. 최신 세대라 비용이 더 나올 것이라 생각할 수도 있지만 그렇지 않으므로 인스턴스를 고를 때 최신 세대를 선택 하는 것이 유리하다.

인스턴스 크기는 vCPU, 네트워크 대역폭과 연계되어 있으므로 사항을 고려한다.

인스턴스 유형을 고를 때는 인스턴스의 유형에 따라 고유한 장점이 있으므로 이를 고려하여 선택한다.

<T>

기본 수준의 CPU성능에 기본 수준 이상의 순간 성능을 제공한다. T 시리즈에서는 시간당 CPU Credit 항목이 있다. 이는 사용하지 않는 시간 동안 Credit을 모으고 사용할 때 순간 확장을 지원한다.

<M>

컴퓨팅, 메모리 및 네트워크 리소스를 균형 있게 제공한다. 범용 인스턴스로 생각하면 된다.

<C>

컴퓨팅 집약적 워크로드에 최적화 되었으며 컴퓨팅 속도당 가격이 저렴하다.

<X>

RAM의 GiB당 요금이 저렴하여 고성능 데이터베이스, 인 메모리 데이터베이스 및 HPC에 적합하다.

<R>

메모리 집약적 Application에 최적화 되어 고성능 데이터베이스, 데이터 마이닝 및 분석 등에 사용된다.

<P>

고성능 GPU 기반 인스턴스다. Deep learning, 기계 학습 등에 사용된다.

<H>

높은 디스크 처리량 및 컴퓨팅과 메모리의 균형을 제공한다.

<I>

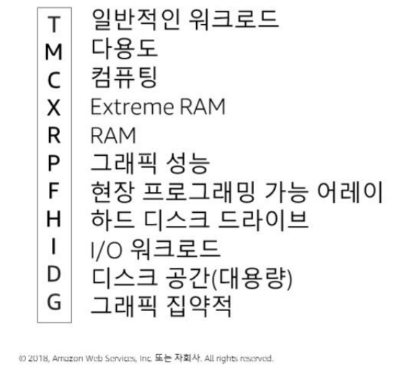
짧은 지연 시간, 매우 높은 랜덤 I/O 성능, 높은 순차 읽기처리량에 최적화된 SSD 기반 인스턴스 스토리지를 제공하며, 낮은 비용으로 높은 IOPS를 제공한다.

<D>

디스크 처리량 성능당 가장 낮은 요금을 제공하여 대량 병렬 처리 데이터 웨어하우징등에 사용한다.

<G>

그래픽 집약적 Application에 최적화 되어 있다. 3D 가상화, 동영상 Encoding, 3D 렌더링 등에 사용된다.



-Instance Option

1 Amazon EC2 전용 인스턴스

전용 인스턴스는 단일 고객을 위한 전용 하드웨어 상의 VPC에서 실행되는 EC2 인스턴스다. 전용 인스턴스는 다른 AWS계정에 속하는 인스턴스로부터 호스트 하드웨어 수준에서 물리적으로 격리 된다.

즉, 인스턴스 중지, 시작 시 호스트 (EC2가 올라간 AWS 실제 물리서버)는 변경 될 수 있지만, 해당 호스트에는 전용 인스턴스만 올라간다.

2) Amazon EC2 전용 호스트

전용 호스트는 고객 전용의 인스턴스 용량을 제공하는 물리적 EC2 서버다. EC2 인스턴스 중지, 시작 시에도 계속 동일한 호스트에서 EC2가 시작된다.

-Placement Group

1) Cluster Placement Group

네트워크 성능 향상을 위하여 EC2 인스턴스들을 인접 위치에 배치하는 것이다. 단일 가용 영역 내에 있는 인스턴스들을 논리적 그룹으로 묶는다. 이 그룹에서는 가장 짧은 지연시간과 가장 높은 초당 패킷 네트워크 성능을 제공한다. 인스턴스를 나중에 그룹에 추가 할 경우 용량 부족 오류가 나타날 가능성이 있어 필요한 모든 인스턴스를 동시에 시작하는 것이 좋다.

2) Distributed Placement Group

기반 하드웨어 장애에 대한 대응을 위하여 EC2 인스턴스들을 중복되지 않도록 개별 장비에 배치하는 것이다.

즉, 의도적으로 다른 기본 하드웨어에 배치되는 인스턴스의 그룹이다.

3) Partition Placement Group

파티션 배치 그룹은 파티션에 분산된 인스턴스 그룹이다.

* Database

-Database 유형

1) Relational Database

엄격한 스키마 규칙 및 데이터 품질 적용이 필요하며, 데이터베이스 트랜잭션이 안정적으로 처리되도록 보장하는

ACID 규정 준수가 필요한 경우 사용한다.

Ex) Amazon RDS, Amazon Redshift, Amazon Aurora

Amazon Aurora는 MySQL 및 PostgreSQL과 호환이 가능한 완전 관리형 데이터베이스다.

2) NoSQL

NoSQL 데이터베이스는 키 값 Pair, 문서 및 그래프를 비롯한 다양한 스토리지 모델 중 하나를 사용하여 데이터를 저장한다. 데이터 베이스를 수평적으로 확장 해야 할 경우, 데이터가 기존 스키마에 적합하지 않은 경우, 읽기/쓰기 속도가 기존 SQL DB에서 경제적으로 지원할 수 있는 범위를 초과한 경우 선택한다.

Ex) Amazon DynamoDB, Amazon ElastiCache, Amazon Neptune

DynamoDB의 경우 단순한 데이터를 대용량으로 보유할 때, 신속하고 간편하게 확장해야 할 때, 복잡한 연결이 필요하지 않을 때 사용한다.

**읽기 일관성 옵션**

1) 최종적 일관된 읽기(Eventually Consistent Read)

DynamoDB 테이블의 데이터를 읽을 때, 응답은 최근 완료된 쓰기 작업의 결과를 반영하지 않을 수 있다.

2) 강력한 일관된 읽기(Strongly Consistent Read)

복제 본에 모든 업데이트를 반영하여 가장 최신 데이터로 응답을 반환한다.

-Database보안

액세스 제어를 통해 보안을 유지한다. DB인스턴스 최초 생성시에 마스터 사용자 계정을 생성하는데 이를 통해 사용자를 제한한다.

DB 보안 그룹을 통해 Amazon RDS 인스턴스에 대한 액세스를 제어한다. DB보안 그룹은 EC2 보안 그룹과 유사하지만 호환되지 않는다. 기본 값은 Access Deny로 명시적 승인을 통해 접속할 수 있다.

저장된 Amazon RDS DB인스턴스와 스냅샷을 암호화 할 수 있다. 암호화를 활성화하면 인스턴스의 자동 백업, 읽기 전용 복제 본 및 스냅샷이 AES-256을 사용해 암호화됩니다.