CloudComputing

Amazon S3 (Simple Storage Service)

1. Amazon S3 개요

1.1 개요

- **Amazon S3 (Simple Storage Service)**는 인터넷용 스토리지 서비스
  - 간단한 웹 서비스 인터페이스를 사용하여 웹에서 언제 어디서나 원하는 양의 데이터를 저장하고 검색
  - 높은 내구성과 가용성 제공
    - Amazon S3는 99.999999999% (11 9s) 내구성을 제공합니다. 데이터를 여러 가용 영역(AZ)에 복제하여 높은 내구성을 보장합니다.
    - 또한 99.99%의 가용성을 제공하여 데이터 접근 가능성을 높입니다.
  - 높은 확장성을 제공하는 경제적인 인프라 제공
    - 원하는 만큼 데이터를 저장하고 필요할 때 액세스할 수 있습니다. 향후 스토리지 요구를 추정할 필요 없이 필요에 따라 확장 또는 축소할 수 있으므로 비즈니스 민첩성이 크게 향상됩니다.
  - 데이터 암호화 (Encryption)
    - SSL을 통한 데이터 전송과 데이터 업로드 후 자동 암호화를 지원
  - 액세스 관리
    - S3는 세밀한 접근 제어를 위해 IAM(Identity and Access Management), 버킷 정책, ACL(Access Control List)을 지원합니다. 이를 통해 데이터에 대한 접근 권한을 사용자 단위, 그룹 단위로 설정할 수 있습니다.
  - 저렴한 비용으로 대용량의 데이터를 저장 - 예를 들어, 자주 사용되는 데이터를 S3 Standard에 저장하다가 일정 기간(예: 30일)이 지나면 S3 Standard-IA로, 이후 더 오래 사용되지 않은 데이터는 S3 Glacier로 이동하도록 정책을 설정할 수 있습니다.

1.2 기본 개념

- Amazon S3는 데이터를 버킷 내에 객체로 저장
  - 버킷: 객체의 컨테이너
  - 객체: 파일+ 메타데이터 (파일에 대한 설명)
  - 키: 버킷 내 객체의 고유한 식별자
  - 데이터 위치: URL: http://[bucket-name].S3.amazonaws.com/[key]

1.3 주요 기능

- 버킷 생성 및 구성
  - 버킷에 객체 업로드 및 관리

    - 버킷 및 객체의 접근 권한 설정
        - 버킷 및 객체 접근 권한 설정은 데이터 보안을 유지하면서 원하는 사용자나 서비스가 데이터에 접근할 수 있도록 조정하는 데 중요한 역할을 합니다.
- 버전 관리
    - S3는 객체에 대해 버전 관리를 지원합니다. 이 기능을 활성화하면 기존 객체를 덮어써도 이전 버전을 보관하여 데이터를 복구할 수 있습니다.
- 수명 주기 정책
    - 특정 시점 이후에 객체를 자동으로 삭제하거나 저렴한 스토리지 클래스로 전환할 수 있도록 수명 주기 정책을 설정할 수 있습니다. 이를 통해 비용을 최적화할 수 있습니다.
- 정적 웹 호스팅
    - S3 버킷을 사용하여 정적 웹사이트(HTML, CSS, JavaScript 기반)를 호스팅할 수 있습니다.
- 멀티파트 업로드 (Multipart Upload)
    - 큰 파일을 S3에 업로드할 때, 멀티파트 업로드를 통해 파일을 여러 부분으로 나누어 병렬로 업로드할 수 있습니다.
- Cross-Region Replication (CRR)
    - 데이터를 여러 리전에 자동으로 복제하여 다른 리전에서도 데이터 접근을 가능하게 합니다.
    - 재해 복구 및 데이터 복제에 유용합니다.

1.4 사용 시나리오

- 백업 및 스토리지
    - 데이터 백업 및 기타 스토리지 서비스를 제공합니다.
- 애플리케이션 호스팅
    - HTML, CSS, JavaScript와 같은 정적 파일을 제공하는 데 적합하며, 서버리스 방식으로 간단한 웹사이트나 웹 애플리케이션을 저비용으로 배포할 수 있습니다.
- 미디어 호스팅
    - 동영상, 사진 또는 음악 업로드 및 다운로드를 호스팅하는 중복 방식의 확장 가능하고 가용성이 높은 인프라를 구축합니다.
- 소프트웨어 제공
    - 고객이 다운로드할 수 있는 소프트웨어 애플리케이션을 호스팅합니다

1.5 Amazon S3 액세스 방식

다음 방법 중 하나를 사용하여 Amazon S3에서 작업할 수 있습니다

1. AWS Management Console

   이 콘솔은 Amazon S3 및 AWS 리소스를 관리하기 위한 웹 기반 사용자 인터페이스입니다. AWS 계정에 가입한 고객은 AWS Management Console에 로그인한 후 AWS Management Console 홈페이지에서 S3를 선택하여 Amazon S3 콘솔에 액세스할 수 있습니다.

2. AWS Command Line Interface

   AWS 명령줄 도구를 통해 시스템 명령줄에서 명령을 실행하거나 스크립트를 구축하여 AWS(S3 등) 작업을 수행할 수 있습니다

3. AWS SDK

AWS에서는 다양한 프로그래밍 언어 및 플랫폼(Java, Python, Ruby, .NET, iOS, Android 등)을 위한 라이브러리와 샘플 코드로 구성된 소프트웨어 개발 키트(SDK)를 제공합니다. AWS SDK를 사용하면 편리하게 S3 및 AWS에 프로그래밍 방식으로 액세스할 수 있습니다.

2. Amazon S3 실습

2.1 Amazon S3 콘솔 사용

- Amazon S3 콘솔은 Amazon S3 작업에 사용할 수 있는 인터페이스 중 하나
- 콘솔을 사용하면 코드를 생성하지 않고도 Amazon S3 작업 수행 가능
- 주요 항목
  - S3 버킷 생성 및 구성
  - 객체 업로드 및 관리
  - 버킷 권한 설정
  - 객체 권한 설정

2.1. 1 S3 버킷 생성 및 구성

- Amazon S3는 사용자가 지정한 AWS 리전에 버킷을 생성
- 버킷 생성에 따른 요금은 발생하지 않으며, 객체를 버킷에 저장하거나 버킷에서 객체를 전송한 경우에만 요금이 부과됨
- Amazon S3 버킷 이름은 버킷을 만든 AWS 리전과 상관없이 전역적으로 고유
- 주요 기능
  - 버킷 생성
  - 정적 웹 사이트 호스팅 구성

2.1.2 객체 업로드 및 관리

- 객체를 Amazon S3 버킷에 업로드하려면 해당 버킷에 대해 쓰기 권한이 있어야 함
- 객체는 이미지, 백업, 데이터, 동영상 등 임의의 파일 형식
- 업로드할 수 있는 파일의 최대 크기는 78GB
- 버킷에 저장할 수 있는 객체 수에는 제한이 없음
- 주요 기능
  - S3 버킷에 파일 및 폴더를 업로드하는 방법

2.1.3 버킷 권한 설정

Amazon S3의 버킷 권한 설정 방법은 S3 버킷과 그 안에 있는 객체에 대한 접근 제어를 관리하기 위한 다양한 방법을 제공합니다. AWS는 S3 버킷에 대한 보안 및 접근 관리를 철저하게 할 수 있도록 여러 가지 도구를 제공하며, 이를 통해 세밀하게 권한을 설정할 수 있습니다. S3 버킷 권한을 설정하는 주요 방법에는 버킷 정책(Bucket Policy), ACL(Access Control List), IAM 정책(IAM Policies), 등이 있습니다.

1. 버킷 정책 (Bucket Policy)

   버킷 정책은 JSON 형식으로 작성된 정책 문서를 통해 버킷에 대한 권한을 정의하는 방식입니다. 버킷 정책을 사용하면 **IAM 사용자, AWS 계정, 또는 모든 사용자(퍼블릭)**가 버킷 및 그 안의 객체에 대해 어떤 작업을 할 수 있는지를 정의할 수 있습니다.

   버킷 정책의 특징:

   - JSON 형식: 버킷 정책은 JSON 문서로 작성되며, 다양한 조건을 설정할 수 있습니다.
   - 버킷 수준에서 적용: 버킷에 있는 모든 객체에 일괄적으로 적용됩니다.
   - 세밀한 조건 설정: 특정 IP 주소, 날짜, MFA 인증 여부와 같은 조건을 설정하여 더욱 세밀한 제어가 가능합니다.

   버킷 정책 예시:

   ```
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": "*",
         "Action": "s3:GetObject",
         "Resource": "arn:aws:s3:::example-bucket/*",
         "Condition": {
           "IpAddress": {
             "aws:SourceIp": "203.0.113.0/24"
           }
         }
       }
     ]
   }
   ```

   위 정책은 example-bucket 버킷에 있는 객체들에 대해 특정 IP 주소 범위(203.0.113.0/24)에서 읽기(s3:GetObject) 작업을 허용합니다.

2. ACL (Access Control List)

   ACL은 버킷이나 객체 수준에서 권한을 부여하는 간단한 접근 제어 방식입니다. ACL은 버킷 소유자 외에 특정 사용자 또는 그룹에게 읽기 및 쓰기 권한을 부여할 수 있으며, 버킷 또는 객체에 대해 개별적으로 설정됩니다.

   ACL의 특징:

   - 간단한 권한 설정: 각 객체나 버킷에 대한 읽기/쓰기 권한을 간단하게 설정할 수 있습니다.
   - 퍼블릭 접근 제어 가능: ACL을 통해 객체나 버킷을 퍼블릭하게 공개하거나 특정 사용자에게만 권한을 부여할 수 있습니다.
   - 객체별로 설정 가능: ACL은 객체 단위로 권한을 설정할 수 있어, 객체마다 개별 권한을 부여할 수 있습니다.

   S3 콜솔을 사용하여 버킷에 대한 ACL 권한을 설정하는 방법

3. IAM 정책 (IAM Policies)

IAM 정책은 AWS 리소스 전체에 대한 권한을 제어하는 방식으로, S3 버킷 및 객체에 대한 권한도 제어할 수 있습니다. IAM 정책은 IAM 사용자, 그룹, 역할에 대해 권한을 부여하며, 이를 통해 특정 AWS 사용자만이 S3 리소스에 접근하도록 제어할 수 있습니다.

IAM 정책의 특징:

- JSON 형식: IAM 정책도 JSON 문서로 작성됩니다.
- IAM 사용자 및 역할과의 통합: IAM 정책을 통해 특정 사용자나 역할에 대해 세밀하게 권한을 설정할 수 있습니다.
- 리소스 기반 권한: IAM 정책은 S3뿐만 아니라 다른 AWS 서비스에 대한 접근 권한도 동시에 관리할 수 있습니다.

IAM 정책 예시:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::example-bucket"
    },
    {
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::example-bucket/*"
    }
  ]
}
```

2.1.4 객체 권한 설정

- ACL(액세스 통제 목록)을 사용하여 S3 객체에 대한 액세스 권한을 관리하는 방법을 설명
  - ACL은 버킷과 객체에 액세스 권한을 부여하는 리소스 기반 액세스 정책
- 버킷 및 객체 권한은 서로 독립적
  - 객체는 해당 버킷으로부터 권한을 상속하지 않음
    - 만약 버킷을 만들고 어떤 사용자에게 쓰기 엑세스 권한을 부여한 경우에, 해당 사용자로부터 명시적으로 권한을 부여 받지 않는 한 해당 사용자 객체를 당신은 접근하지 못합니다.
- S3 콘솔을 사용하여 객체에 대한 ACL 권한을 설정하는 방법

2.2 AWS SDK 사용