Service Accounts and Roles, Fundamentals

서비스 계정

Service Accounts는 사용자 대신 가상 머신에 권한을 부여하는 특별한 유형의 Google 계정이다.
Service Accounts는 서비스 계정을 사용하여 서비스의 Google API를 호출하므로 사용자가 직접 관여하지 않는다.
*주로 API 및 Google Cloud 서비스에 대한 안전하게 관리되는 연결을 보장하는데 사용되고 신뢰할 수 있는 연결에 대한 액세스 권한을 부여하며 악의적인 연결을 거부하는 것은 모든 Google Cloud 프로젝트의 필수 보안 기능이다.

Compute Engine VM은 서비스 계정으로 실행될 수 있으며 해당 계정에는 필요한 리소스에 액세스할 수 있는 권한이 부여될 수 있다.
이러한 방식으로 서비스 계정은 서비스의 ID이며 서비스 계정의 권한은 서비스가 액세스할 수 있는 리소스를 제어하고 서비스 계정은 계정에 고유한 이메일 주소로 식별된다.

서비스 계정 유형

사용자 관리 서비스 계정

Google Cloud Console을 사용하여 새 클라우드 프로젝트를 만들고 프로젝트에 Compute Engine API가 사용 설정된 경우 기본적으로 Compute Engine 서비스 계정이 생성된다.
프로젝트에 App Engine 애플리케이션이 포함된 경우 기본적으로 프로텍트에 기본 App Engine 서비스 계정이 생성된다.

Google 관리 서비스 계정

사용자 관리 서비스 계정 외에도 프로젝트의 IAM 정책 또는 콘솔에서 일부 추가 서비스 계정을 볼 수 있다.
이러한 서비스 계정은 Google에서 만들고 소유하고, 서로 다른 Google 서비스를 나타내며 각 계정에는 Google Cloud 프로젝트에 액세스할 수 있는 IAM 역할이 자동으로 부여된다.

Google API 서비스 계정

사용자를 대신하여 내부 Google 프로세스를 실행하도록 특별히 설계되었으며 콘솔의 서비스 계정 섹션에 나열되지 않는다.
기본적으로 계정에는 프로젝트에 대한 프로젝트 편집자 역할이 자동으로 부여되며 콘솔의 IAM 섹션에 나열된다.
이 서비스 계정은 프로젝트가 삭제된 경우에만 삭제된다.
*Google 서비스는 프로젝트에 대한 액세스 권한이 있는 계정에 의존하므로 프로젝트에서 서비스 계정의 역할을 제거하거나 변경해서는 안된다.

서비스 계정 생성 및 관리

새 Cloud 프로젝트를 만들면 GCP는 해당 프로젝트 아래에 Compute Engine 서비스 계정 1개와 App Engine 서비스 계정 1개를 자동으로 생성한다.
프로젝트에 최대 98개의 추가 서비스 계정을 만들어 리소스에 대한 액세스를 제어할 수 있다.

서비스 계정 만들기

서비스 계정을 만드는 것은 프로젝트에 구성원을 추가하는 것과 유사하지만 서비스 계정은 개별 사용자가 아닌 애플리케이션에 속한다.

1. 서비스 계정 만들기
    - gcloud iam service-accounts create `service account name` --display-name `내 서비스 계정 이름`

서비스 계정에 역할 부여

IAM 역할을 부여할 때 서비스 계정을 리소스 또는 ID로 취급할 수 있다.
애플리케이션은 서비스 계정을 ID로 사용하여 Google Cloud 서비스에 인증한다.
*서비스 계정으로 실행 중인 Compute Engine 가상 머신이 있는 경우 프로젝트(리소스)의 서비스 계정(ID)에 역할을 부여할 수 있다.

동시에 VM을 시작할 수 있는 사용자를 제어할 수도 있다. 사용자(ID)에게 서비스 계정(리소스)에 대한 serviceAccountUser 역할을 부여하여 이를 수행할 수 있다.

특정 리소스에 대한 서비스 계정에 역할 부여

서비스 계정이 Cloud Platform 프로젝트의 리소스에 대한 특정 작업을 완료할 수 있는 권한을 갖도록 서비스 계정에 역할을 부여한다.

1. 이전에 생성된 서비스 계정에 역할 부여
    - gcloud projects add-iam-policy-binding $DEVSHELL_PROJECT_ID --member serviceAccount:\serviceaccountname\@DEVSHELL_PROJECT_ID.iam.gserviceaccount.com --role roles/editor

역할 이해

ID가 Google Cloud API를 호출할 때 Google Cloud IAM에서는 ID에 리소스를 사용할 수 있는 적절한 권한이 있어야 한다.
사용자, 그룹 또는 서비스 계정에 역할을 부여하여 권한을 부여할 수 있다.

역할 유형

Cloud IAM에는 세 가지 유형의 역할이 있다.

1. Cloud IAM 도입 이전에 존재했던 소유자, 편집자, 뷰어 역할을 포함하는 기본 역할
2. 특정 서비스에 대한 세분화된 액세스를 제공하고 Google Cloud에서 관리하는 사전 정의된 역할
3. 사용자 지정 권한 목록에 따라 세분화된 액세스를 제공하는 사용자 정의 역할
   *Reference: https://cloud.google.com/iam/docs/understanding-roles?hl=ko

클라이언트 라이브러리를 사용하여 서비스 계정에서 BigQuery 엑세스

필요한 역할이 있는 서비스 계정의 도움을 받아 인스턴스에서 BigQuery 공개 데이터 세트 쿼리

서비스 계정 만들기

1. Google Cloud Console > Menu > IAM & Admin > Service Accounts > Create service account
2. 필요한 세부 정보 입력
   example)
   Service account name: bigquery-qwiklab
   Grant service account access to project
   Role 1) BigQuery Data Viewer
   Role 2) BigQuery User

VM 인스턴스 만들기

1. VM 인스턴스로 이동하여 세부 정보 입력
   *Service account 란에 전에 생성한 서비스 계정인 bigquery-qwiklab 지정

Compute Engine에서 필요한 명령어들 입력

1. Google Cloud Console > Menu > Compute Engine > VM Instances > bigquery-instance의 SSH
   - sudo apt-get update
   - sudo apt-get install -y git python3-pip
   - pip3 install --pip upgrade
   - pip3 install google-cloud-bigquery
   - pip3 install pyarrow
   - pip3 install pandas
   - pip3 install db-dtypes
2. 파이썬 예제 파일 생성
   example)

```
echo "
from google.auth import compute_engine
```

```
        from google.cloud import bigquery
        credentials = compute_engine.Credentials(

    service_account_email='YOUR_SERVICE_ACCOUNT')
        query = '''
        SELECT
          year,
          COUNT(1) as num_babies
        FROM
          publicdata.samples.natality
        WHERE
          year > 2000
        GROUP BY
          year
        '''
        client = bigquery.Client(
            project='Your Project ID',
            credentials=credentials)
        print(client.query(query).to_dataframe())
        " > query.py
```

3. 프로젝트 ID를 query.py에 추가
    - sed -i -e "s/Your Project ID/$(gcloud config get-value project)/g" query.py
4. sed 명령이 파일의 프로젝트 ID를 변경했는지 실행해서 확인
    - cat query.py
5. 서비스 계정 이메일을 query.py에 추가
    - sed -i -e "s/YOUR_SERVICE_ACCOUNT/bigquery-qwiklab@$(gcloud config get-value project).iam.gserviceaccount.com/g" query.py
6. sed 명령이 파일의 서비스 계정 이메일을 변경했는지 실행해서 확인
    - cat query.py
7. 쿼리 실행
    - python3 query.py