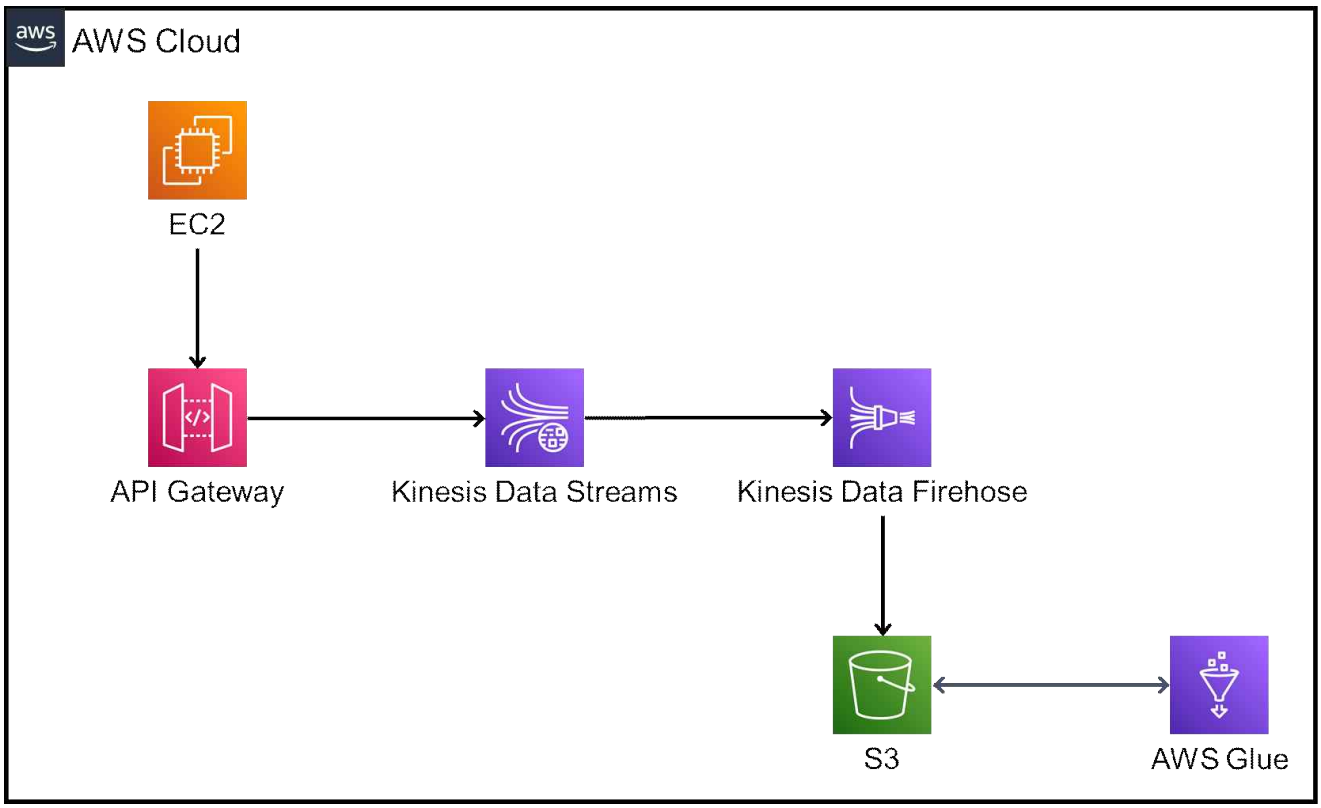
2023 지방기능경기대회 과제

직 종 명	클라우드컴퓨팅	과 제 명	Automation	과제번호	제2과제
경기시간	4시간	비 번 호		심사위원 확 인	(인)

1. 요구사항

개발 효율을 위하여 ETL 파이프라인을 구성하고자 합니다. AWS 환경에서 개발하고 있기 때문에 모두 AWS solution을 활용하여 파이프라인을 구성할 예정입니다. API Gateway를 통해 로그를 받고 Kinesis를 통해 S3에 저장합니다. Glue를 이용하여 로그를 변환할 수 있어야 합니다.

다이어그램



Software Stack

AWS
- VPC
- EC2
- Kinesis Data Streams
- Kinesis Data Firehose
- S3
- Glue

2. 선수 유의사항

1) 기계 및 공구 등의 사용 시 안전에 유의하시고, 필요 시 안전장비 및 복장 등을 착용하여 사고를 예방하여 주시기 바랍니다.
2) 작업 중 화상, 감전, 찰과상 등 안전사고 예방에 유의하시고, 공구나 작업도구 사용 시 안전보호구 착용 등 안전수칙을 준수하시기 바랍니다.
3) 작업 중 공구의 사용에 주의하고, 안전수칙을 준수하여 사고를 예방하여 주시기 바랍니다.
4) 경기 시작 전 가벼운 스트레칭 등으로 긴장을 풀어주시고, 작업도구의 사용 시 안전에 주의하십시오.
5) 선수의 계정에는 비용제한이 존재하며, 이보다 더 높게 과금될 시 계정 사용이 불가능할 수 있습니다.
6) 문제에 제시된 괄호박스 <>는 변수를 뜻함으로 선수가 적절히 변경하여 사용해야 합니다.
7) 문제의 효율을 위해 Security Group의 80/443 outbound는 anyopen하여 사용할 수 있도록 합니다.
8) Bastion EC2는 채점시 사용되기 때문에 종료되어 불이익을 받지 않도록 주의해 주시기 바랍니다.
9) 모든 리소스는 서울(ap-northeast-2) 리전에 구성합니다.

3. Bastion 서버

EC2를 활용해 Bastion 서버를 구성합니다. 해당 서버는 public 존에 위치하고 stop 후 재시작하더라도 public ip가 변경돼서는 안 됩니다. 외부에서 Bastion 서버에 SSH로 접속할 수 있도록 구성합니다. 채점 시에도 사용함으로 인스턴스가 종료되어 불이익을 받지 않도록 합니다. Bastion 서버 내 어떤 사용자에서 awscli 명령어를 호출하여도 PowerUserAccess policy 권한을 갖도록 설정해야 합니다. 아래 요구사항에 따라 Bastion 서버를 구성합니다.

- EC2 type : t3.small
- 이미지 : Amazon Linux2
- VPC : 기본 VPC
- Subnet : Public Subnet
- 권한 : AWS PowerUser policy
- 설치 패키지 : awscli, curl, jq
- Tag : Name=wsi-bastion-ec2

4. S3

아래 요구사항에 따라 S3 버킷을 생성합니다. 지급받은 titles.json 파일을 버킷 생성 후 <S3버킷>/data/ref/titles.json에 업로드합니다. 지급받은 samplelog.json 파일을 <S3버킷>/data/raw/2022/01/01/samplelog.json에 업로드합니다. 아래 모든 서비스에서 해당 버킷만을 사용하도록 합니다.

- 버킷 이름 : wsi-<비번호>-<4자리 임의 영문>-etl

5. API Gateway

"Content-Type: application/json" 헤더, JSON 형식의 데이터(Body)가 포함된 POST 요청을 받은 경우 Body의 단일 데이터 레코드를 Kinesis Data Streams에 씁니다. 데이터가 온전히 전달되도록 합니다. 아래 요구사항에 따라 API Gateway를 구성하고 배포합니다.

- API 이름(유형) : wsi-api(REST API)
- 리소스 : /api
- 배포 스테이지 이름 : prod

예) {"uuid":"1234", "device_ts":"2022-01-01 09:10:25"} 등과 같이 JSON 포맷의 데이터가 포함된 바디를 요청하면 Kinesis Data Streams에 전달되어야 합니다.

6. Kinesis Data Streams

API Gateway로 부터 받은 데이터를 수집하여 Kinesis Data Firehose에 정보를 전송합니다. 아래 요구사항에 따라 Data Stream을 구성합니다.

- Data Stream 이름 : wsi-data-stream
- 용량 모드 : 프로비저닝
- 프로비저닝 된 샤드 : 1

7. Kinesis Data Firehose

Kinesis Data Stream을 데이터 원본으로 사용하여 4번에서 생성한 S3에 저장합니다. 동적 파티셔닝은 비활성화하며 압축 또는 암호화를 사용하지 않습니다. 아래 요구사항에 따라 Delivery Stream을 구성합니다.

- Delivery Stream 이름 : wsi-delivery-stream
- 저장 위치 : <4번에서 생성한 S3 버킷>/data/raw/<YYYY>/<MM>/<dd>/<HH>/

8. Glue 크롤러

<4번에서 생성한 S3 버킷>/data 폴더 하위의 모든 폴더 및 파일을 크롤링하는 크롤러를 하나 생성합니다. 한 번 이상 크롤러를 실행하여 데이터 카탈로그 테이블이 구성되도록 합니다. 아래 요구사항에 따라 크롤러 및 데이터 카탈로그를 구성합니다.

- 크롤러 이름 : wsi-glue-crawler
- 데이터베이스 이름 : wsi-glue-database
- 테이블 이름 : data 폴더 하위 폴더 이름(ref, raw)

8. Glue Studio 작업

크롤러로 생성한 데이터 카탈로그 테이블을 소스로 사용하여 아래와 같이 데이터가 변환되도록 구성합니다. samplelog.json파일과 titles.json 파일을 참고합니다. 변환된 데이터는 <4번에서 생성한 S3 버킷>/result/ 폴더에 JSON 포맷으로 저장되고 아래에 명시된 데이터 카탈로그 테이블에 업데이트되어야 합니다. 작업 이름은 wsi-glue-job으로 설정합니다.

- 변환된 데이터 : {"title_id":66,"title":"Start-Up","uuid":"00003c25-40bc-43c5-a8a7-b64e40f25b32", "device_ts":"2022-06-11 09:10:25.013","device_id":4,"device_type":"IPhone"}
- 데이터베이스 이름 : wsi-glue-database
- 테이블 이름 : result

9. Glue Studio 워크플로

크롤링 후 작업을 실행시키는 워크플로를 구성합니다. 워크플로 실행 시 wsi-glue-crawler를 실행시키고 크롤링이 성공하면 wsi-glue-job이 실행되어야 합니다.

- 워크플로 이름 : wsi-glue-workflow