2023년도 전국기능경기대회

직 종 명	클라우드컴퓨팅	과 제 명	Trouble-shooting	과제번호	제 3과제
경기시간	2시간	비 번 호		심사위원 확 인	(인)

1. 요구사항

1-1. Container image build problems

1-2. S3 object problems

1-3. IAM permission control

1-4. IaC problems

S/W Stack

trouble-shooting으로 미제공

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 비용의 제한이 존재하며, 이보다 더 높게 요금이 부과될 시 계정 사용이 불가능할 수 있습니다.
6) 문제에 제시된 괄호 박스 <>는 변수를 뜻함으로 선수가 적절히 변경하여 사용해야 합니다.
7) EC2 인스턴스의 TCP 80/443 outbound는 anyopen하여 사용할 수 있도록 합니다.
8) 과제의 Bastion 서버에서 대부분의 채점이 이루어짐으로 인스턴스를 생성하지 않았거나 종료된 상태면 채점이 불가능하니 각별히 주의하도록 합니다.
9) 과제 종료 시 진행 중인 테스트를 모두 종료하여 서버에 부하가 발생하지 않도록 합니다.
10) 별도 언급이 없는 경우, ap-northeast-2 리전에 리소스를 생성하도록 합니다.
11) 모든 리소스의 이름, 태그, 변수과 변수는 대소문자를 구분합니다.

3. Bastion 서버

EC2를 활용해 Bastion 서버를 구성합니다. 네트워크 위치는 무관하며 외부에서 접근할 수 있도록 구성합니다. Bastion 서버 자체는 채점하지 않으나, 이후 과제 풀이를 위해 다음과 같이 생성해야 합니다. 별도 언급이 없는 경우 채점은 Bastion에 root 계정으로 SSH 접근하여 awscli 등 명령어를 수행하여 진행됩니다.

- Instance type : t3.small
- AMI : Amazon Linux2
- 설치 패키지 : awscli, jq, curl, docker
- Tag : Name=wsi-bastion
- EC2 IAM Role : 모든 리소스에 대해 full access를 가지는 role을 생성하여 붙입니다.

4. Container image build problems

1) 지급파일 소개

foo 애플리케이션을 실행하기 위한 컨테이너 이미지를 생성하고자 합니다. 제공된 day3-container-image-build.zip 파일을 Bastion 서버의 /home/ec2-user/container/ 경로에 압축 해제합니다. 압축파일 내에는 Dockerfile과 어플리케이션 코드가 들어 있습니다.

해당 어플리케이션은 GET /v1/foo API를 제공합니다. 어플리케이션 코드에는 문제가 없습니다. 따라서 과제 수행시 어플리케이션 소스 코드를 변경해서는 안됩니다. 어플리케이션은 8080 포트를 사용합니다.

2) 문제

지급된 Dockerfile을 통해 빌드를 **정상 수행**하고 **ECR Scan을 수행시 취약점**이 발견되지 않도록합니다. 정상 빌드가 된다면 Bastion에서 localhost:8080으로 API 호출이 가능해야 합니다. 채점시 docker run -d -p 8080:8080 <image> 명령어를 이용해 컨테이너 실행 후 http://localhost:8080/v1/foo 로 호출하여 일부 채점을 진행합니다. 호출시 {"application":"foo"} body와 200 응답코드를 반환해야 합니다.

3) 구성

도커이미지를 저장하기 위하여 foo 이름의 ECR Repository를 생성하고 Scan-on-push 기능을 활성화합니다. 이미지 빌드 및 업로드를 위하여 Bastion 서버 /home/ec2-user/container/build.sh 경로에 스크립트를 생성합니다. build.sh를 실행시키면 foo:<time> 이름/태그를 갖는 이미지를 빌드합니다. 또한 해당 이미지는 동일한 <time> tag 값을 이용해 foo repository에 푸쉬해야 합니다. <time> tag는 한국시간 기준으로 YYYYMMDD-hhmmss 형식을 가집니다. 2023년 10월 18일 21시 10분 5초일 경우 foo:20231018-211005 이 되어야 합니다. build.sh 실행 후 5분 이내에 이미지가 빌드되고 업로드 되어야 합니다. 컨테이너가 실행되면 별도의 작업 없어도 어플리케이션이 동작해야

합니다

- 로컬 이미지 이름 예제 : foo:20231018-211005
- ECR에 업로드된 이미지 이름 예제 : ecr.xxxxx/foo:20231018-211005
- 사용하는 파일: day3-container-image-build.zip
- 채점시 build.sh 실행 명령어 : cd /home/ec2-user/container; ./build.sh

5. S3 object problems

1) 지급파일 소개

제공된 day3-objects.zip 파일은 민감정보를 포함하고 있는 csv 파일들이 존재합니다. 압축을 해제하고 파일이름 수정없이 모든 파일을 private S3 버킷 최상위 경로에 업로드 합니다. 파일들을 public S3 버킷에는 직접 업로드하지 않도록 합니다.

2) 문제

실수로 민감정보가 포함된 데이터를 S3에 업로드 하였습니다. 파일들이 모두 다른 포맷을 가지고 있고 양도 많아 하나하나 처리하기 힘든 상황입니다. 효율적인 방법을 활용하여 **민감정보를 모두 제거** 하는 것이 필요합니다.

3) 구성

AWS Macie를 활용하면 민감정보를 효율적으로 제거할 수 있습니다. private S3 버킷에 존재하는 데이터에서 Macie를 활용해 민감데이터를 찾아 제거한 뒤 public S3에 저장되도록 구성합니다. 단, 민감데이터만 삭제하고 파일 이름은 그대로 유지되어야 합니다. 민감데이터가 포함된 컬럼과 생성해야할 버킷 정보는 아래와 같습니다.

- 사용하는 파일: day3-objects.zip
- Sensitive data columns: **licensenumber, epid**
- Private S3 bucket(기존 데이터 저장용): wsi-day3-private-<임의의 영문 4자리>
- Public S3 bucket(변환된 데이터 저장용): wsi-day3-public-<임의의 영문 4자리>

6. IAM permission control

1) 지급파일 소개

제공된 day3-iam.zip 파일에는 IAM Policy를 정의한 json파일이 존재합니다. 압축을 해제하여 과제 해결에 사용하도록 합니다. 해당 파일은 Bastion에 특정 경로에 위치 시킬필요없이 IAM 계정 생성시 참고하여 사용하면 됩니다.

2) 문제

EC2 Tag 정보를 활용하여 IAM 유저가 EC2를 생성하는 권한을 관리하고자 합니다. IAM red-user는 **Key=project, Value=red 라는 Tag가 존재하는 EC2만 생성 가능**해야 합니다. 하지만 현재 가지고 있는 iam-policy.json를 red-user에게 적용하였을 때 EC2 인스턴스 생성

에 문제가 있습니다.

3) 구성
AWS console 로그인이 가능한 red-user 이름의 IAM 계정을 생성합니다. 주어진 iam-policy.json을 수정하여 red-user가 해당 Tag 정보를 가지는 EC2만 생성 가능하고, 다른 Tag 정보를 가지는 EC2는 생성할 수 없도록 IAM policy를 적용합니다. 만약 다른 태그 정보를 가지고 있다면 생성 및 수정은 불가합니다. 다만 EC2의 running 상태, 인스턴스 타입 등과 같은 정보는 Tag 설정 상관없이 확인 가능해야 합니다.
- 사용하는 파일: day3-iam.zip

7. IaC problems

1) 지급파일 소개
제공된 day-iac.zip 파일에는 cloudformation 스택을 정의한 yaml파일이 존재합니다. 압축을 해제하여 cloudformation 스택을 생성하는 것에 사용하도록 합니다. 해당 파일은 Bastion에 특정 위치에 위치시킬 필요 없습니다.

2) 문제
제공된 yaml파일으로 cloudformation 스택을 생성하여 S3 및 KMS 키를 프로비저닝하고자 합니다. 하지만 yaml파일에 문제가 있어 cloudformation 스택 생성에 실패하고 있습니다.

3) 구성
yaml파일을 사용하여 day3-wsi-s3-bucket-stack 이름의 cloudformation 스택을 생성합니다. 반드시 제공된 yaml파일을 사용하여 cloudformation 스택을 생성해야 하며, 스택 생성 오류의 원인과 관련 없는 문장을 수정할 경우 감점의 원인이 될 수 있습니다. cloudformation 스택 생성이 성공적으로 이뤄지면, day3-wsi-temp-bucket-<임의의 영문 4자리> 이름을 가진 s3 버킷과 day3-wsi-temp-bucket-<임의의 영문 4자리> 이름을 가진 kms 키가 프로비저닝 됩니다.
- 사용하는 파일: day3-iac.zip
- cloudformation stack 이름: day3-wsi-s3-bucket-stack
- Parameter(BucketName): day3-wsi-temp-bucket-<임의의 영문 4자리>