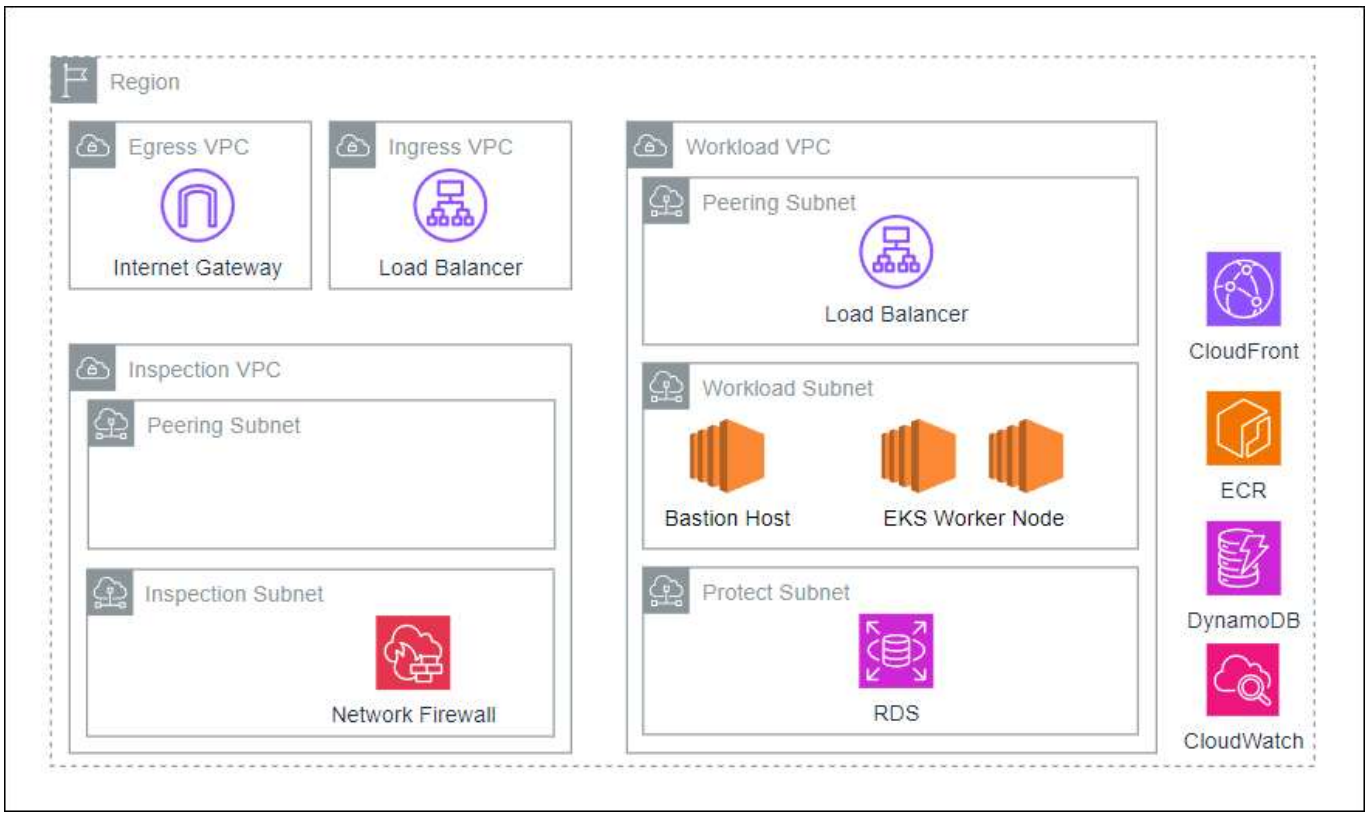
2024년도 전국기능경기대회 과제

직 종 명	클라우드컴퓨팅	과제명	Solution Architecture	과제번호	제1과제
경기시간	4시간	비번호		심사위원 확 인	(인)

1. 요구사항

당신은 WorldSkills의 시스템 중 인증 시스템에 대한 Infrastructure 설계와 운영을 담당하는 업무를 맡았습니다. 인증 시스템은 MSA로 구성되어 있습니다. 주어진 요구사항과 클라우드의 설계원칙인 고가용성, 확장성, 비용, 보안 등을 잘 고려하여 인프라를 구축해야 합니다.

다이어그램



Software Stack

AWS	개발언어/프레임워크
- S3 - VPC - EC2 - ELB - EKS - ECR - RDS - DynamoDB - CloudFront - CloudWatch - AWS Network Firewall	- Golang/gin

2. 선수 유의사항

※ 다음 유의사항을 고려하여 요구사항을 완성하시오.

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) 문제에 제시된 괄호박스 < > 는 변수를 뜻함으로 선수가 적절히 변경하여 사용해야 합니다.
6) 문제 풀이와 채점의 효율을 위해 Security Group의 80/443 Outbound는 Anyopen하여 사용할 수 있도록 합니다.
7) Bastion EC2는 채점 시 사용되기 때문에 종료되거나 연결 문제, 권한 문제 등으로 발생할 수 있는 불이익을 받지 않도록 주의하시기를 바랍니다.
8) 모든 리소스는 서울(ap-northeast-2) 리전에 구성합니다.
9) 제공자료는 수정 없이 사용합니다. 제공자료를 수정해서 사용하면 불이익을 받을 수 있습니다.
10) 과제 지에 별다른 요구사항이 없는 경우 선수가 AWS Best Practices를 참고하여 알아서 구성합니다.

3. Application

주어진 3개의 어플리케이션을 배포하고 사용자에게 제공해야 합니다. 각 어플리케이션의 부록 3의 Application을 참고합니다.

4. Networking

클라우드 인프라 구성을 위하여 기본적인 네트워크 구성을 시행합니다. 부록 1의 정보를 참고하여 AWS VPC를 생성합니다.

4-1) Transit Gateway

생성된 VPC들의 Transit Gateway을 통해서 각 VPC간 통신이 가능하도록 구성합니다. 단, VPC간 통신 시 각 VPC에 있는 peering subnet끼리만 통신이 되어야합니다. 또한 wsc-prod-vpc의 workload subnet의 경우 다른 VPC와 통신이 필요할 경우 peering subnet에 생성된 Private NAT을 통해서 통신이 되어야합니다.(부록 2의 네트워크 흐름도 참고) 이때 생성되는 Transit Gateway Attachement, Transit Gateway Route Table의 경우 선수가 알아서 구성합니다.

- Transit Gateway Name : wsc-vpc-tgw

4-3) Network Firewall

- network firewall name : wsc-inspect-firewall
- network firewall policies : wsc-inspect-rules
- network firewall rule : wsc-deny
- network firewall subnet : wsc-inspect-secure-sn-a, wsc-inspect-secure-sn-c
- logging type : Flow

4-4) Network Firewall Rule

아래 적용되는 규칙의 경우 Stateful 형태의 Suricata rule을 사용하여 차단되어야합니다.
(채점 시 해당 규칙을 모두 PASS 시키므로 이점 유의하여 구성합니다.)

- wsc-prod-vpc에 있는 시스템에서 외부 ifconfig.io로 향하는 HTTP/HTTPS 차단
- wsc-prod-vpc에 있는 시스템에서 외부와 tls 1.0, tls 1.1로 통신 시 차단

5. Bastion Host

wsc-prod-vpc에 Bastion Host를 생성하여 Session Manger를 통해서만 접근이 가능하도록 구성합니다. 구축된 리소스 및 인프라에 접근 및 관리할 수 있도록 구성합니다. 원활한 채점을 위해 아래 표시된 패키지는 미리 설치 및 root 유저에서도 실행이 가능해야합니다.

- name : wsc-prod-bastion
- instance type : t3.medium

  - subnet : wsc-prod-workload-sn-c
- OS : amazon linux 2023
  - Packages : awscli2, kubectl, eksctl, curl, ping, jq, openssl, mysql

6. DynamoDB

부록 3에 작성된 참고하여 DynamoDB Table을 생성합니다. 보안을 위해 VPC 내부에서 운영 중인 리소스/시스템에서 접근 시 endpoint gateway를 통해 인터넷을 거치지 않고 내부로 통신이 가능하도록 구성합니다.
경기 종료 전 해당 Table에 생성된 item을 모두 삭제하여 아무 item도 없는 table 상태로 만들어야합니다. (미진행 시 채점시 불이익을 받을 수 있습니다.)

7. Container Registry

Container Image들을 효율 적으로 관리하기 위해 Registry를 생성합니다. 그 후 각 애플리케이션별로 Registry를 따로 생성 및 관리되도록 구성합니다.
  - customer Container Registry Name: customer
  - product Container Registry Name: product
  - order Container Registry Name: order

8. Container Ochestration

배포된 애플리케이션을 모두 EKS Cluster에서 효율적으로 운영이 되도록 구성합니다. 모든 Pod들은 "wsc-prod"이라는 namespace에서 Pod 형태로 운영이 되어야합니다. 또한 외부에서 클러스터 접근이 불가능해야합니다. 각 Deployment의 이름은 <application name>-deploy로 지정합니다. (e.g customer-deploy)
  - EKS Cluster Name: wsc-prod-cluster
  - EKS Access Type: Private
  - EKS Node Instance Type: m5.large
  - EKS Node Name Tag: wsc-prod-nodegroup

9. Relational Database Service

부록 3에 작성된 DB Table을 참고하여 "wscdb"라는 데이터베이스에 Table을 생성합니다.
  - RDS Instance identifier name: wsc-prod-db-cluster
  - DB Engine: Aurora for MySQL 3.05
  - DB Instance Type: db.t3.medium
  - DB User: skill
  - DB Passworkd: Skill53##

10. Load Balancing

wsc-ingress-vpc에서 ALB 생성 후 사용자가 해당 ELB로 접근 시 wsc-prod-vpc에서 생성한 ALB를 통해 애플리케이션에 접근이 가능하도록 구성합니다.

- wsc-ingress-vpc Load Balancer
    - LB Type : Layer 7에서 동작하는 LB
    - LB Name : wsc-ingress-lb
- wsc-prod-vpc Load Balancer
    - LB Name : Layer 7에서 동작하는 LB
    - LB Name : wsc-prod-lb

11. Web Hosting

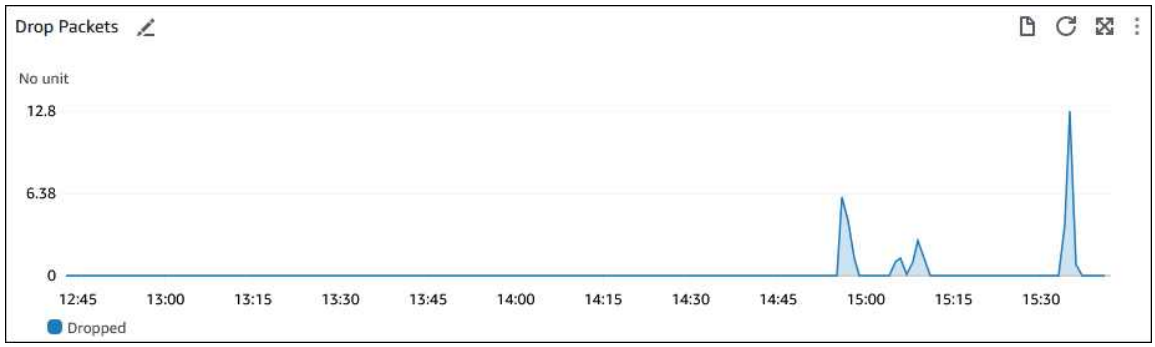
제공된 Front-end 애플리케이션을 S3에 업로드합니다. CloudFront를 사용해 /static/*으로 접근 시 해당 Front-end 애플리케이션에 문제없이 접속할 수 있도록 구성합니다. 캐싱을 활용하여 사용자들에게 쾌적한 환경을 제공해합니다. 한편, Back-end 애플리케이션도 /v1/*에 대한 요청을 CloudFront로 제공해야하며, 캐싱을 실시해서는 안됩니다.

- S3 Bucket Name: wsc-frontend-<랜덤 문자열 4자리>
- CloudFront Protocol: HTTPS를 통하여 접근 시 문제가 없도록 구성
- CloudFront Origin: S3 Bucket, wsc-ingress-vpc에 생성된 ELB
- Edge: 한국뿐만 아니라 전 세계의 유저가 빠른 속도로 접근이 가능하도록 구성
- Tag: Name=wsc-prod-cdn

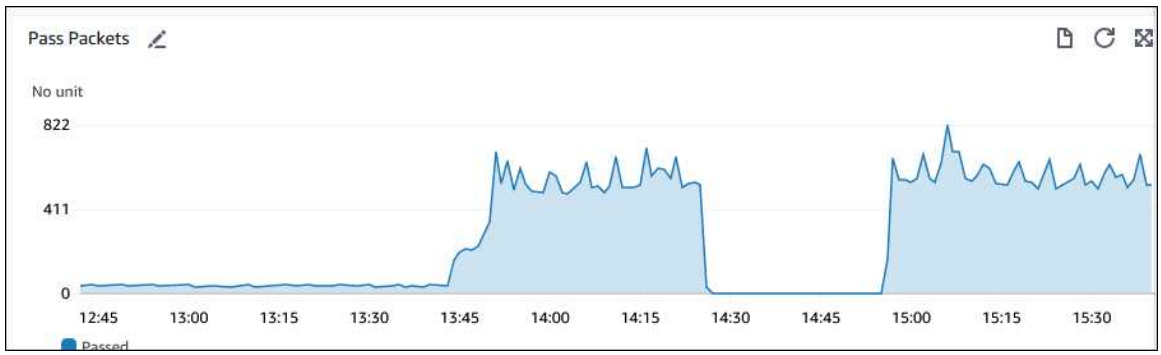
12. Monitoring

Network Firewall에서 발생하는 이벤트를 CloudWatch Dashboard를 통해 관제를 위해 아래 요구사항을 참고하여 구축합니다. (모든 위젯의 Period는 1분으로 설정)

- CloudWatch Dashboard : wsc-security-inspect
- widget
  - widget name : Drop Packets
    - description : Network Firewall에서 Drop된 패킷들을 그래프로 표현합니다.
    - x-array Label : Dropped



  - widget name : Pass Packets
    - description : Network Firewall에서 Pass된 패킷들을 그래프로 표현합니다.
    - x-array Label : Passed



부록 1

- VPC 정보 - wsc-prod-vpc

Name Tag	CIDR
wsc-prod-vpc	10.100.0.0/16

Subnets 정보

Name Tag	CIDR	Availability Zone
wsc-prod-peering-sn-a	10.100.1.0/24	ap-northeast-2a
wsc-prod-peering-sn-c	10.100.2.0/24	ap-northeast-2c
wsc-prod-workload-sn-a	10.100.10.0/24	ap-northeast-2a
wsc-prod-workload-sn-c	10.100.11.0/24	ap-northeast-2c
wsc-prod-protect-sn-a	10.100.20.0/24	ap-northeast-2a
wsc-prod-protect-sn-c	10.100.21.0/24	ap-northeast-2c

- VPC 정보 - wsc-inspect-vpc

Name Tag	CIDR
wsc-inspect-vpc	100.64.0.0/16

Subnets 정보

Name Tag	CIDR	Availability Zone
wsc-inspect-secure-sn-a	100.64.0.32/28	ap-northeast-2a
wsc-inspect-secure-sn-c	100.64.0.48/28	ap-northeast-2c
wsc-inspect-peering-sn-a	100.64.0.64/28	ap-northeast-2a
wsc-inspect-peering-sn-c	100.64.0.80/28	ap-northeast-2c

- VPC 정보 - **wsc-ingress-vpc**

Name Tag	CIDR
wsc-ingress-vpc	172.20.0.0/16

Subnets 정보

Name Tag	CIDR	Availability Zone
wsc-ingress-pub-sn-a	172.20.0.32/28	ap-northeast-2a
wsc-ingress-pub-sn-c	172.20.0.64/28	ap-northeast-2c
wsc-ingress-peering-sn-a	172.20.0.96/28	ap-northeast-2a
wsc-ingress-peering-sn-c	172.20.0.128/28	ap-northeast-2c

- VPC 정보 - **wsc-egress-vpc**

Name Tag	CIDR
wsc-egress-vpc	172.22.0.0/16

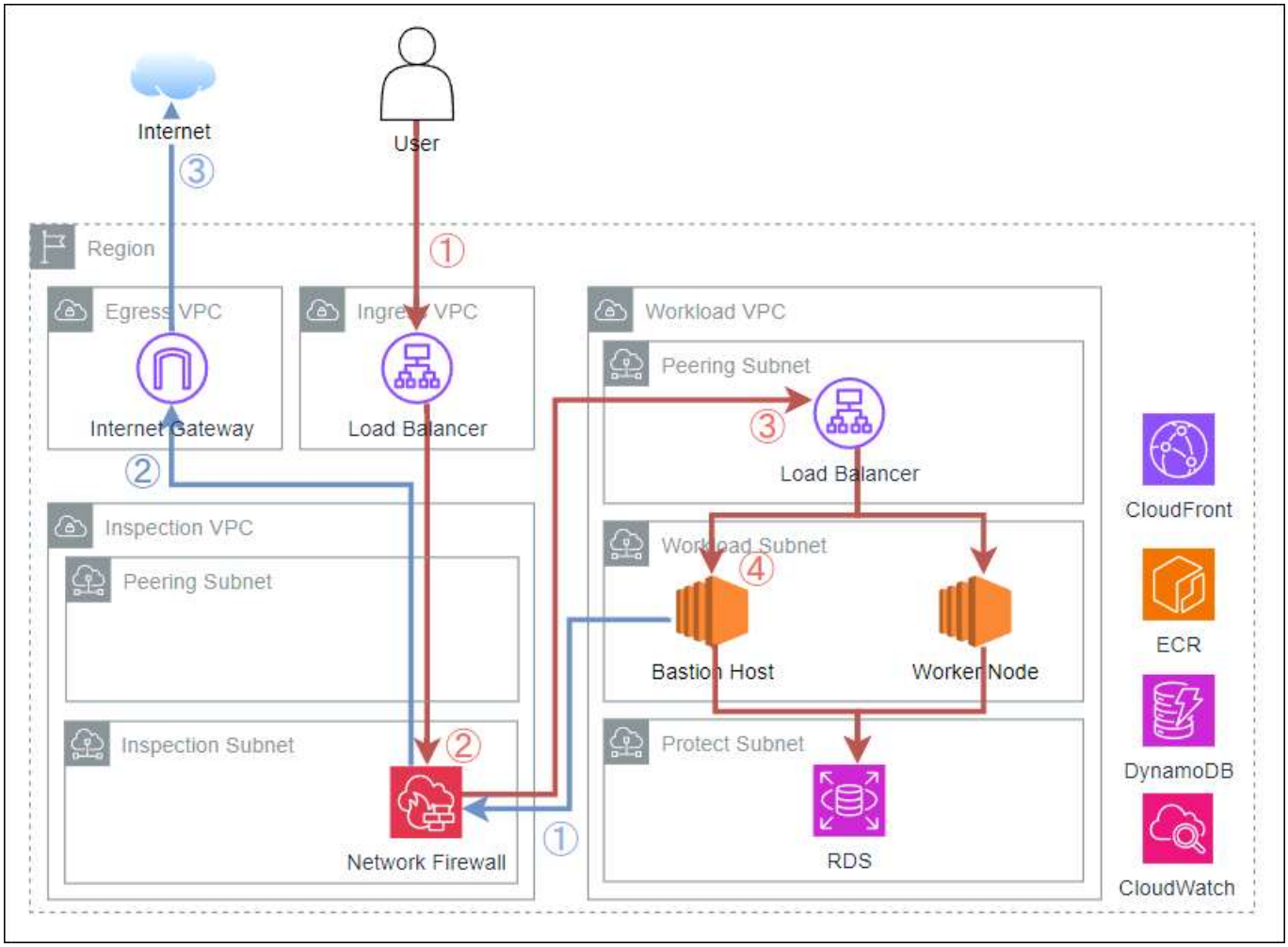
Subnets 정보

Name Tag	CIDR	Availability Zone
wsc-egress-pub-sn-a	172.22.0.32/28	ap-northeast-2a
wsc-egress-pub-sn-c	172.22.0.64/28	ap-northeast-2c
wsc-egress-peering-sn-a	172.22.0.96/28	ap-northeast-2a
wsc-egress-peering-sn-c	172.22.0.128/28	ap-northeast-2c

부록 2

네트워크 흐름도

일반 사용자가 wsc-ingress-vpc를 통해 공개된 주소로 접근하여 Workload VPC에 운영 중인 API 서비스에 접근이 되어야합니다.



외부 사용자가 서비스 접근 시

1. wsc-ingress-vpc에 생성된 ELB로 접근
2. wsc-inspect-vpc에 생성된 Network Firewall에서 트래픽 필터링
3. Workload VPC에서 운영 중인 시스템에 접근

Workload VPC 시스템이 외부에 접근 시

1. workload subnet에서 출발 시 peering subnet 대역으로 NAT 후 통신
2. wsc-inspect-vpc에 생성된 Network Firewall에서 트래픽 필터링
3. wsc-egress-vpc를 통해 외부와 통신

부록 3

- 모든 어플리케이션은 golang/gin을 사용하여 개발되었으며, x86 시스템에서 빌드하였습니다.
- 모든 어플리케이션은 tcp 8080 포트를 사용합니다.
- 모든 어플리케이션은 표준 출력으로 접근 로그를 출력합니다.
- 모든 어플리케이션은 /healthcheck 경로로 상태 확인을 제공합니다.

customer

- API Spec

Path	Method	Request Format
/v1/customer	GET	Query String
		?id=xxxxxxxx
/v1/customer	POST	Request Body
		'{"id":"xxxxxx","name":"xxxxxxx","gender":"xxxxxx"}'

- RDBMS Table [table의 이름은 customer로 설정해야합니다. (하드코딩되어 있음.)]

Column Name	Method	ETC
id	VARCHAR(255)	-
name	VARCHAR(255)	-
gender	VARCHAR(255)	-

product

- API Spec

Path	Method	Request Format
/v1/product	GET	Query String
		?id=xxxxxxxx
/v1/product	POST	Request Body
		'{"id":"xxxxxx","name":"xxxxxxx","category":"xxxxxx"}'

- RDBMS Table [table의 이름은 product로 설정해야합니다. (하드코딩되어 있음.)]

Column Name	Method	ETC
id	VARCHAR(255)	-
name	VARCHAR(255)	-
category	VARCHAR(255)	-

- OS Environment

Environment Key	Description
MYSQL_USER	RDBMS 연결에 사용할 사용자명
MYSQL_PASSWORD	RDBMS 연결에 사용할 사용자 암호
MYSQL_HOST	RDBMS 연결에 사용할 호스트 이름
MYSQL_PORT	RDBMS 연결에 사용할 포트번호
MYSQL_DBNAME	RDBMS 연결에 사용할 데이터베이스 이름

order

- API Spec

Path	Method	Request Format
/v1/order	GET	Query String
		?id=xxxxxxxx
/v1/order	POST	Request Body
		'{"id":"xxxxxx","customerid":"xxxxxxx","productid":"xxxxxx"}'

- DynamoDB Table [table의 이름은 order로 설정해야합니다. (하드코딩되어 있음.)]

Key Name	Data Type	ETC
id	String	-
customerid	String	-
productid	String	-

- OS Environment

Environment Key	Description
AWS_REGION	DynamoDB 연결에 사용할 리전 코드 (e.g ap-northeast-2)