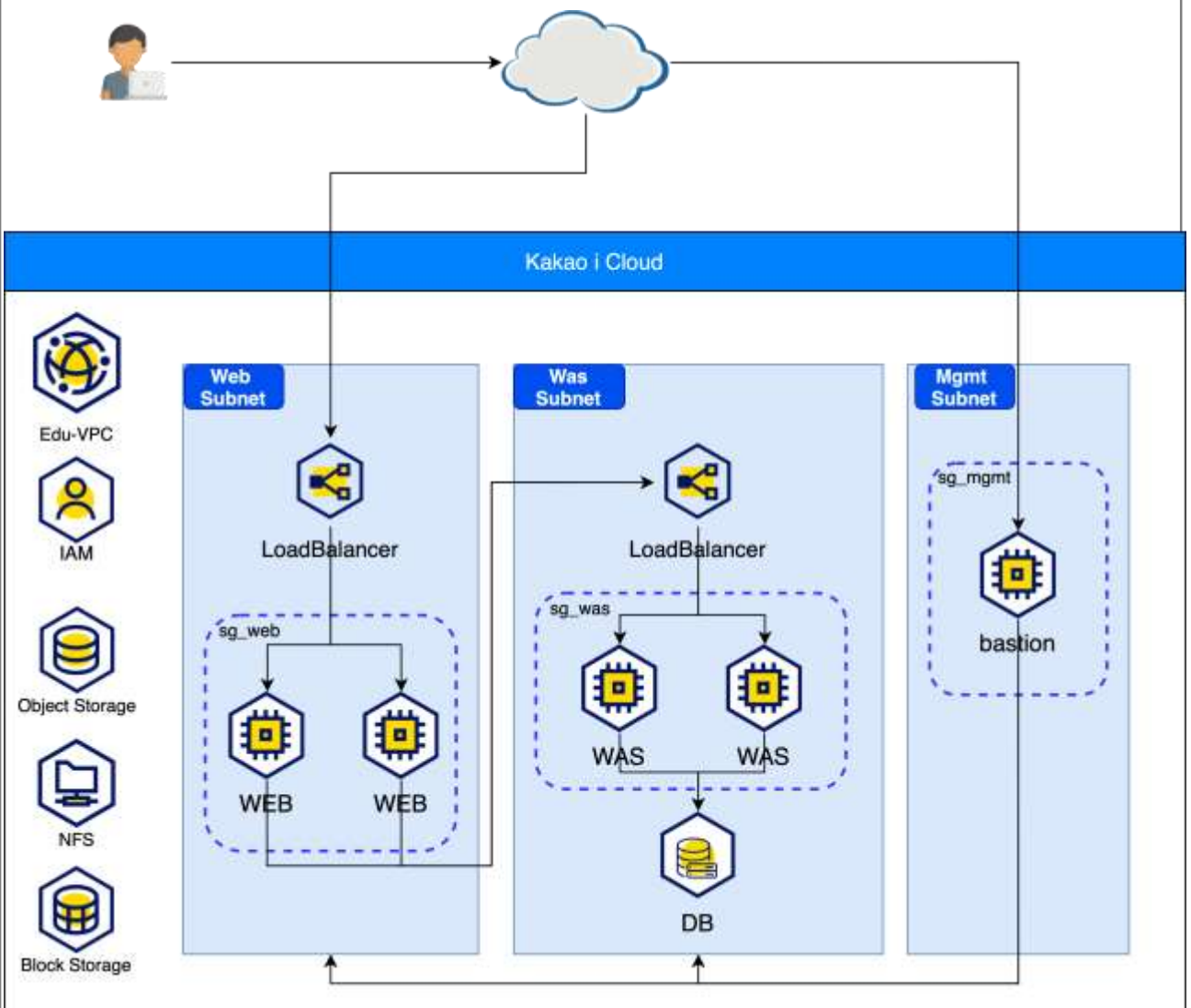


Kakao i Cloud 실습 가이드



목 차

1. 회원가입
2. 네트워크 환경 구성 (시연)
3. 보안 그룹 및 키페어 생성
4. bastion VM 생성 및 고정IP 할당
5. bastion VM 생성 후 체크
6. Mysql 인스턴스 생성
7. 애플리케이션 구동을 위한 WEB, WAS VM 설정
8. 볼륨 추가 및 스냅샷 생성
9. 공유 파일 시스템(NFS) 생성
10. 로드밸런서 생성 및 연결 (SSL 오프로딩)
11. 오브젝트 스토리지 생성

[1. WSL을 통한 리눅스 사용 – Windows 사용자만]

1. appx 파일 다운로드

<https://kep.li/4DRGg7o0pa>

2. Ubuntu_over_WSL_V2.txt 파일 다운로드 받기

<https://kep.li/4DVR5mo0hb>

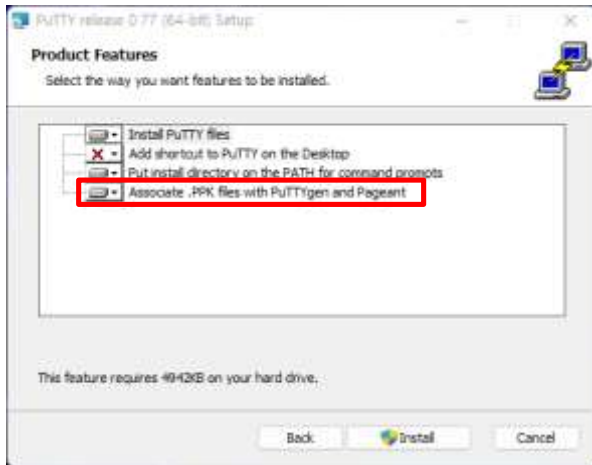
3. Ubuntu_over_WSL_V1.txt의 내용을 따라 Ubuntu 설치

[2. putty 설치 및 pem to ppk 파일 변환 – Windows 사용자만]

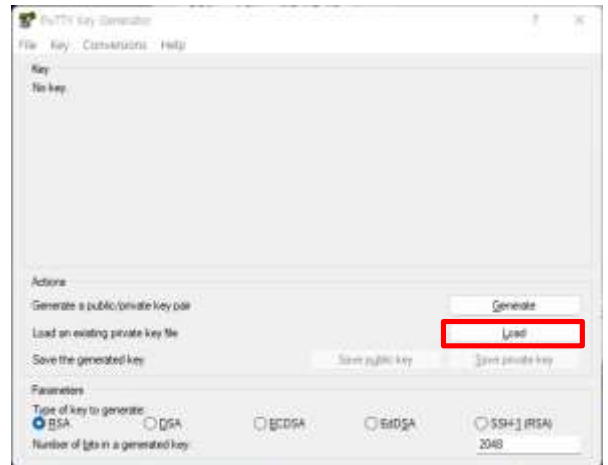
1. Putty 다운로드

<https://the.earth.li/~sgtatham/putty/latest/w64/putty-64bit-0.77-installer.msi>

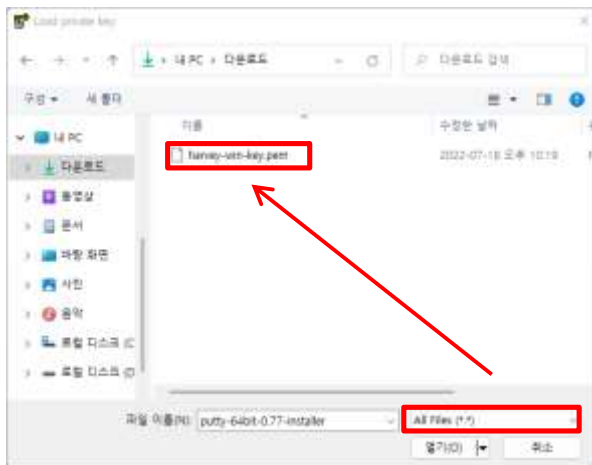
2. Putty 설치 중 puttygen 필수 선택



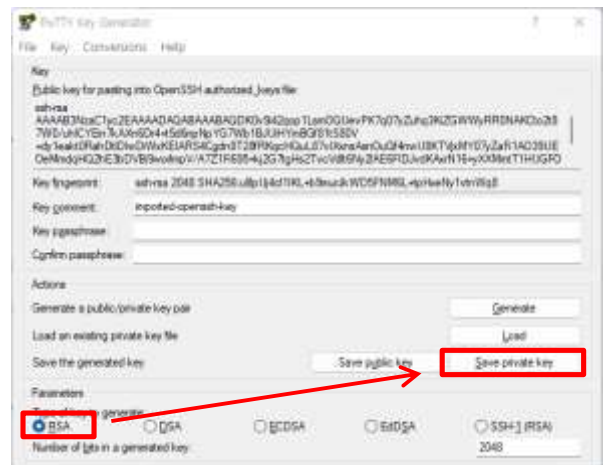
3. puttygen 실행 후 Load 클릭



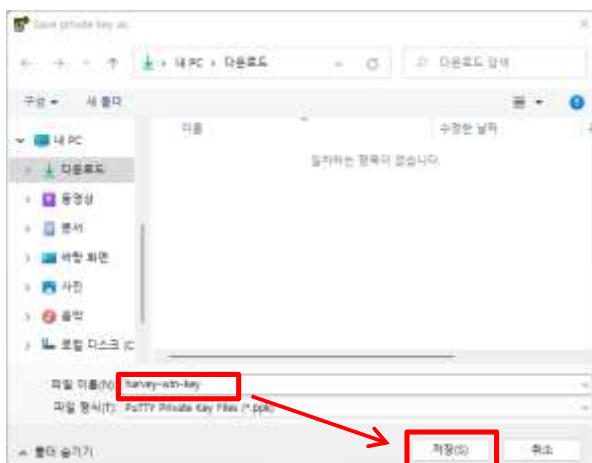
4. All files 로 변경 후 .pem 파일 선택



5. RSA 선택 확인 후 Save private key 클릭

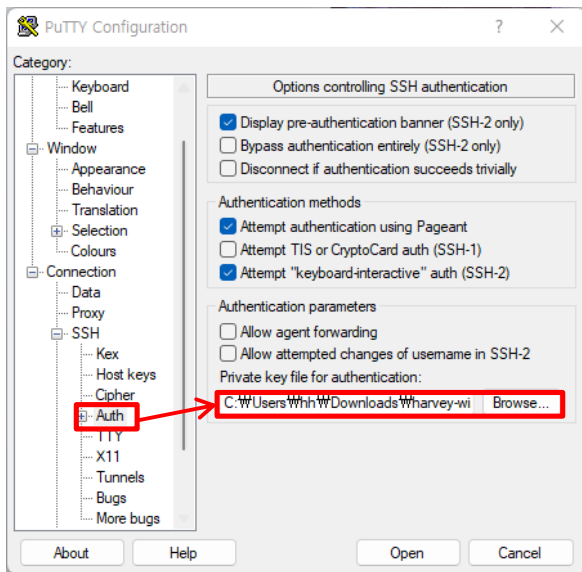


6. 키 이름을 입력 후 저장

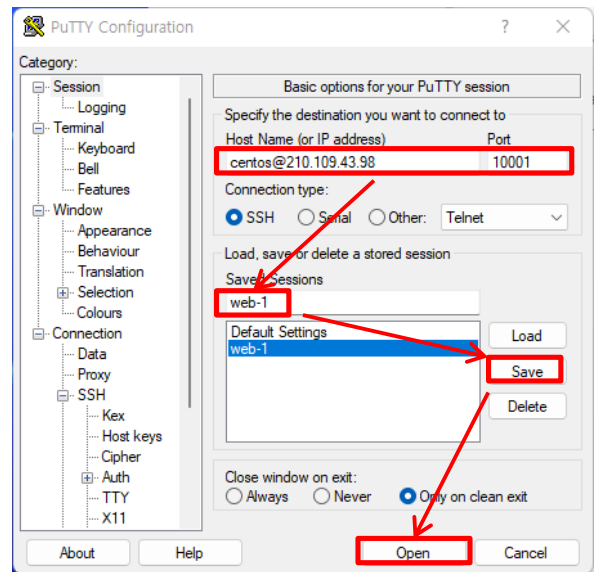


[putty를 사용한 ssh 접속 – Windows 사용자만]

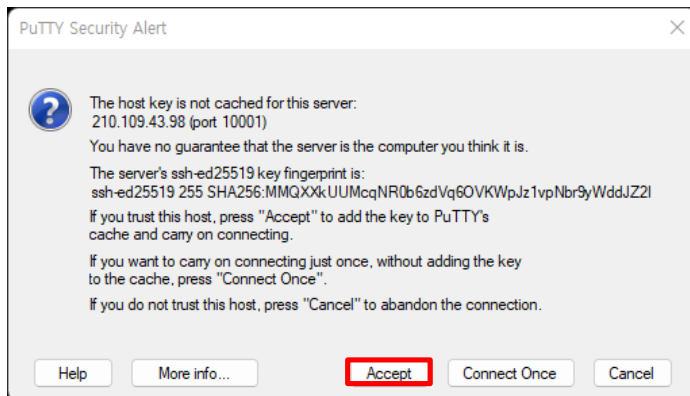
1. Connection – SSH – Auth 에서 ppk 첨부



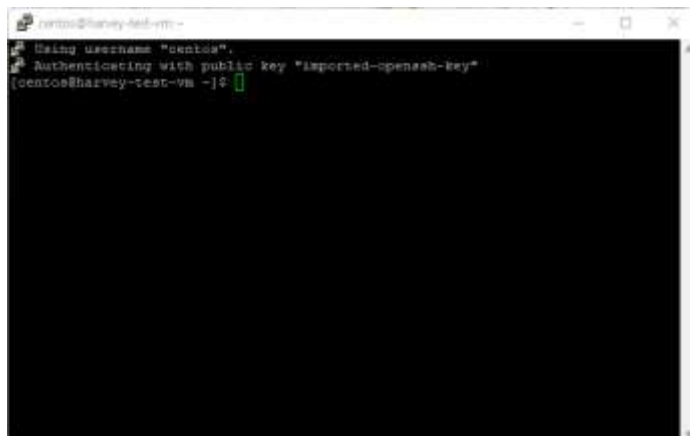
2. Session 에서 IP와 포트 입력, 저장 후 Open




3. Accept 클릭



4. 접속 완료



1. kakao i cloud 포털 : <https://www.kakaoicloud.com>
2. 가이드 : 포털 우측상단의  (사용자 가이드) 참조
3. 회원 가입 절차 (기관 → 팀 → 프로젝트 생성되어야 모든 절차가 완료됨)
 - 3-1. 포털 홈페이지 우측하단의 **사전 컨설팅 신청** 버튼 클릭
 - a. 회사명, 사업자 등록번호, 이름, 이메일, 휴대폰 번호와 문의내용등을 입력
 - b. 동의 내용등을 선택 후 **사전 컨설팅 신청** 버튼을 클릭
 - 3-2. 회원가입
 - a. [3-1]에서 입력한 이메일로 “회원 가입 코드”를 발급
 - b. 받은 이메일의 가입코드를 복사한 후 **회원 가입** 버튼을 클릭
 - c. “가입 코드”입력 후 “약관 동의”를 진행
 - d. 조직 정보 입력
 - 사업자의 정보 및 청구/세금 계산서등 받을 사람의 정보 입력
 - e. 회원 정보 입력
 - 클라우드 조직 이름(1회 입력 변경불가), 담당자 이름, 담당자 이메일, 휴대전화 입력
 - f. **신청완료** 버튼 클릭
 - 3-3. 담당자에게 가입 결과 이메일 통보
 - a. 담당자의 **비밀번호 등록** 클릭
 - b. 비밀번호 입력 후 비밀번호 등록 완료로 회원가입 완료
 - 3-4. 프로젝트 생성 요청
 - a. 담당자(팀 관리자)가 클라우드 콘솔 로그인
 - b. 상단 조직을 선택하여 조직범위 IAM이동
 - c. 오른쪽 상단 **+ 프로젝트 만들기** 버튼을 클릭
 - d. 프로젝트 정보 입력 후 **만들기** 버튼을 클릭 프로젝트 생성 완료

4. 팀/프로젝트 멤버관리

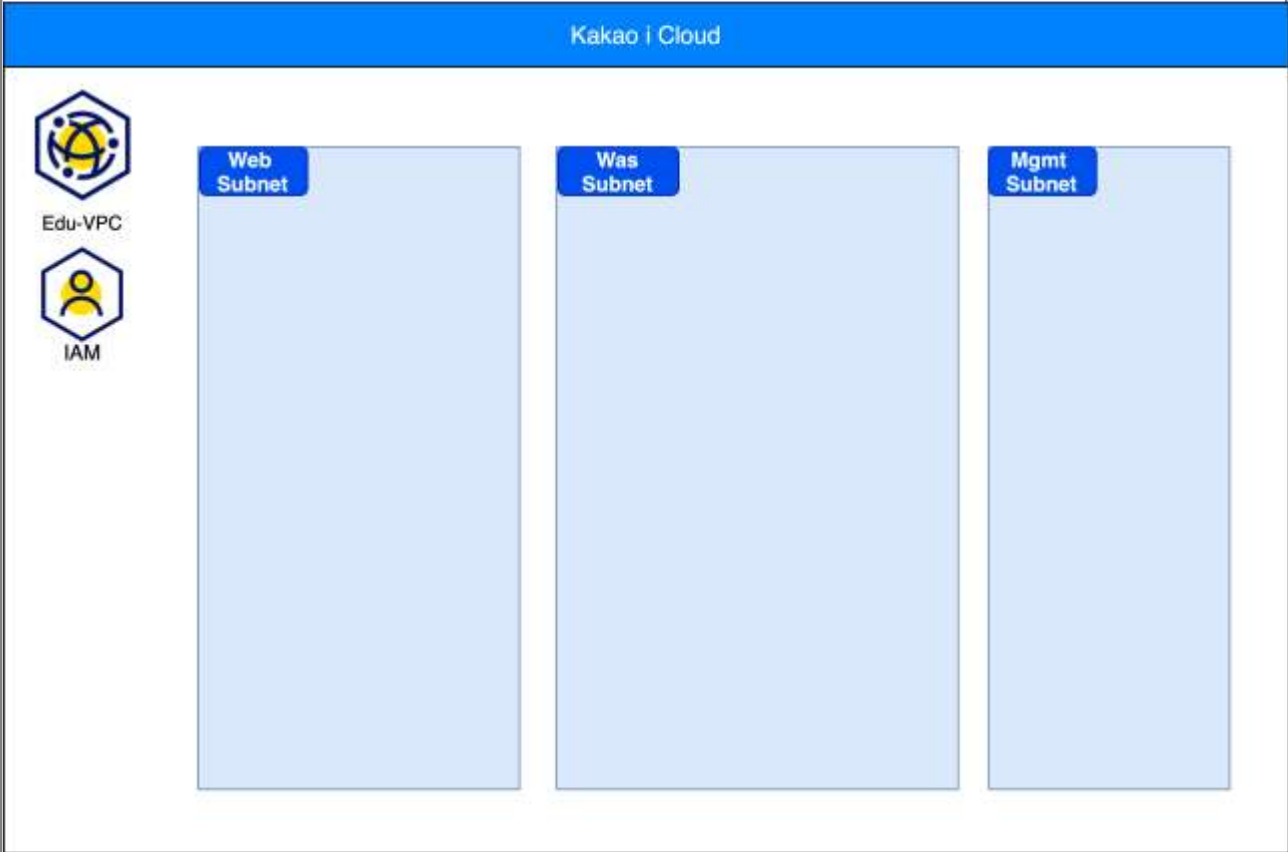
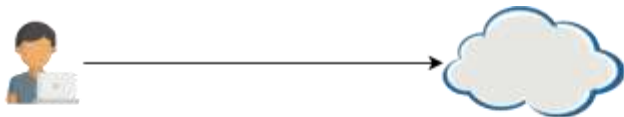
4-1. 팀 멤버 추가

- IAM의 유저(User) 탭에서 **+ 사용자 등록** 버튼클릭
- 일반 및 간편 등록의 방법으로 추가할 인원의 e-mail 입력
- 등록** 버튼 클릭


4-2. 프로젝트 멤버 추가 (4-1 완료 후)

- 프로젝트(Project) 탭에서 만들어진 프로젝트(Project)에 **⋮** 실행(Action) 목록 클릭
- “프로젝트 레벨 역할 관리” 실행(Action) 선택 후 사용자 이메일, 레벨 역할 선택
- 다음** 클릭 후 **적용** 클릭함으로 프로젝트 멤버 추가 완료

실습 시나리오 (네트워크 확인)



1. 포탈 로그인 및 Console 클릭
2. VPC 확인
 - 2-1. Console 화면
 - a. 좌측 메뉴 > **Networking** > **VPC** 클릭
 - 2-2. VPC 화면(VPC 탭)
 - a. 우측 상단의 VPC 탭 클릭
 - b. VPC 목록에서 “**edu-vpc**” 확인

VPC 이름	상태	IP CIDR 블록	공유된 서브넷
 edu-vpc Share Default	● Active	172.16.0.0/16	3개

3. 서브넷 확인
 - 3-1. Console 화면
 - a. 좌측 메뉴 > **Networking** > **VPC** 클릭
 - 3-2. Subnet 화면(Subnet 탭)
 - a. 우측 상단의 Subnet 탭 클릭
 - b. 서브넷 목록에서 해당 Project에 공유된 Subnet 목록을 확인
 - External Subnet : “**edu-subnet-web**”
 - Internal Subnet : “**edu-subnet-was**”
 - Management Subnet : “**edu-subnet-mgmt**”

서브넷	상태	IP CIDR 블록	라우팅 테이블
edu-subnet-web a28ba9e-5ed1-45ed-aae-c8d0ca8577ff	● Active	172.16.131.0/24	main 043a6dc0-573e-4b24-9701-47a7c943b8f1
edu-subnet-was a94f69cc-0a6f-4b4d-b1c3-418f123915cf	● Active	172.16.132.0/24	main 043a6dc0-573e-4b24-9701-47a7c943b8f1
edu-subnet-mgmt 436d8f79-ed90-4876-b4c6-7c5e880a4362	● Active	172.16.133.0/24	main 043a6dc0-573e-4b24-9701-47a7c943b8f1

4. Routing Table 확인

4-1. Console 화면

a. 좌측 메뉴 > **Networking > VPC** 클릭

4-2. Routing Table 화면 (Routing Table 탭)

a. 우측 상단의 Routing Table 탭 클릭

b. 라우팅 테이블 목록에서 프로젝트에 소속된 라우팅 테이블을 확인

라우팅 테이블	상태	VPC	연결된 서브넷	기본
main [a3a4c3c-573a-4b24-9701-47a27c9432bf]	Active	edu-vpc	4개	예

5. Routing Table 상세 확인

5-1. 라우팅 테이블 이름 하단의 uuid를 클릭

5-2. 라우팅 탭에서 라우트 목록 확인

* Internet Gateway(igw)가 없는 경우 Public IP를 통한 접근불가

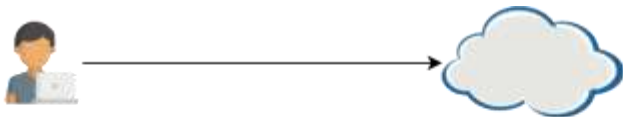
목적지	Target 타입	Target 이름	상태
0.0.0.0/0	igw	igw_45f7ee44-64b8-4cdb-812c-b5456ee98577	Active
172.16.0.0/16	Local	Local	Active

6. 연결된 서브넷 목록 확인

6-1. 연결된 서브넷 탭에서 라우트 테이블에 연결된 서브넷(Subnet) 목록 확인

서브넷 이름	서브넷 ID	IP CIDR 블록
main	f50deb5a-0a79-42f3-af6b-349315f39ee3	172.16.0.0/20
edu-subnet-web	e25baa8a-1ed1-45e0-aaae-c8c0ce6971f3	172.16.131.0/24
edu-subnet-was	e64066cc-0dcf-4046-b1c7-418ff25937cf	172.16.132.0/24
edu-subnet-mgmt	436dd578-ab90-487b-b4c4-7c5e885b4363	172.16.133.0/24

실습 시나리오 (보안 그룹)



Kakao | Cloud



Edu-VPC



IAM

Web
Zone

sg_web

Was
Zone

sg_was

Mgmt
Zone

sg_mgmt

1. Console 로그인
2. Bastion용 보안 그룹 생성
 - 2-1. Console 화면
 - a. 좌측 **메뉴** > **Networking** > **VPC** 로 이동
 - b. **Security** 탭으로 이동
 - c. 우측상단의 **+ 시큐리티 그룹 만들기** 버튼 클릭
 3. 시큐리티 그룹 만들기 화면
 - 3-1. 시큐리티 그룹 이름에 **sg_mgmt** 입력
 - 3-2. 인바운드 정책 탭에서 규칙 생성
 - a. 인바운드 정책 추가를 위하여 **+ 추가하기** 를 클릭
 - b. 각 항목을 아래와 같이 선택하거나 입력
 - 프로토콜: **"TCP"** 선택
 - 포트 번호 : **81** 입력
 - 패킷 출발지(Source): **본인 및 고객의 인터넷 IP** 입력 (e.g. 10.0.0.0/32)
 - * <https://findip.kr>와 같은 곳에서 공인 IP 주소를 확인하여 입력
 - * 제한 없이 모든 곳에서 요청을 보내거나 받을 때에는 아래와 같이 입력 합니다.
 - IPv4 : 0.0.0.0/0
 - IPv6 : :::0
 - c. [추가하기] 를 클릭
 - 프로토콜: **"TCP"** 선택
 - 포트 번호 : **10001-10099** 입력
 - 패킷 출발지(Source): **본인 및 고객의 인터넷 IP** 입력 (e.g. 10.0.0.0/32)
 - d. **만들기** 버튼 클릭
 - 3-3. 필요 시 상기 내용을 반복하거나 응용 필요

**** 과제 ****

1. 보안 그룹 실습[1-3]번을 참조하여 서비스 보안그룹을 생성해 보시오.

1-1. **sg_web** 보안그룹 생성,

edu_subnet_mgmt에서 들어오는 요청에 대해 22번 포트 허용 룰(정책) 추가

- edu_subnet_mgmt에 존재하는 베스천에서 edu_subnet_web에 존재하는 web vm으로 접근하기 위함

동일 서브넷에서 80 포트에 대한 요청 허용 룰(정책) 추가

- 로드밸런서를 고려하여, 80포트에 대한 패킷 출발지는 동일 서브넷으로 제한하도록 보안그룹을 설정

1-2. **sg_was** 보안그룹 생성,

edu_subnet_mgmt에서 들어오는 요청에 대해 22번 포트 허용 룰(정책) 추가

edu_subnet_web 에서 들어오는 80 포트에 대한 요청 허용 룰(정책) 추가

- sg_was이 적용되는 VM은 망분리의 형태로 Internal 에서만 접근될 수 있도록 하기 위함

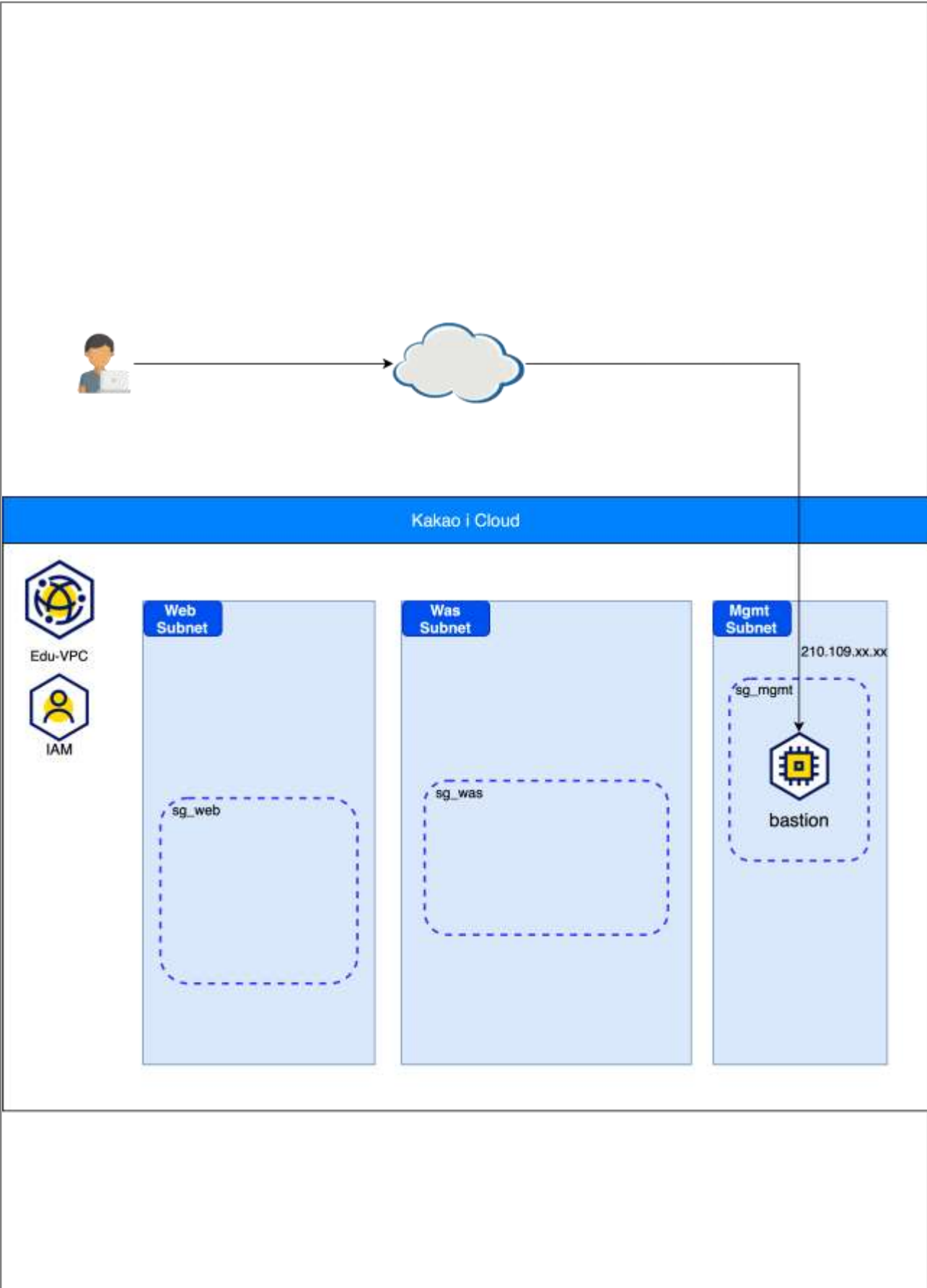
동일 서브넷에서 8080 포트에 대한 요청 허용 룰(정책) 추가

- 로드밸런서를 고려하여, 8080포트에 대한 패킷 출발지는 동일 서브넷으로 제한하도록 보안그룹을 설정

<서브넷 목록 참고>

리우팅 연결된 서브넷		
서브넷 이름	서브넷 ID	IP CIDR 블록
main	f50deb5a-0a79-42f3-af6b-349315f39ee3	172.16.0.0/20
edu-subnet-web	e26baa8a-1ed1-45e0-aaae-c8c0ce6971f3	172.16.131.0/24
edu-subnet-was	e64066cc-0dcd-4046-b1c7-418ff25937cf	172.16.132.0/24
edu-subnet-mgmt	436dd578-ab90-4878-b4c4-7c5e885b4363	172.16.133.0/24

실습 시나리오 (Bastion VM 생성 및 고정IP 할당)



1. 키페어 생성

1-1. Console 화면

- a. 좌측 **메뉴** > **Computing** > **Virtual Machine**로 이동

1-2. 키페어 화면

- a. 좌측상단의 **Key Pair** 탭 클릭
- b. **+ 키페어 만들기** 클릭

1-3. 키페어 생성 화면

- 생성방법 : **새로 생성하기** 선택
- 키페어 이름: 생성할 이름 입력
- **만들기** 버튼을 클릭 시 키페어 개인키가 자동으로 다운로드 됨

1-4. (Only Mac/Linux)

다운로드된 개인키의 파일에 대한 권한을 터미널을 통하여 (600으로)변경 합니다.

```
$ chmod 600 키페어_이름.pem
```

```
$ ls -al *.pem
```

```
-rw----- 1 deploy root 5674 Jul 14 11:46 키페어_이름.pem
```

(* 다운로드 받은 파일 확장자가 .cer인 경우 .pem 대신 .cer 입력)

1-5. (Only Windows)

Putty를 사용하는 경우, 실습 교재 **3페이지**를 참고하여 .pem (또는 .cer) 파일을 .ppk 파일로 변환 합니다.


1. Console 로그인
2. bastion용 VM 생성
 - 2-1. Console 화면
 - a. 좌측 메뉴 > **Computing** > **Virtual Machine**로 이동
 - 2-2. Virtual Machine 화면
 - a. 우측 상단의 **+ 인스턴스 만들기** 버튼 클릭
 - 2-3. 인스턴스 만들기
 - a. 생성 화면에서 각 항목을 선택 또는 입력
 - 이미지 : 내 이미지 탭에서 **"Edu-Bastion"** 유형 **선택**
 - 인스턴스 개수: **1** 입력
 - 인스턴스 이름 : **bastion** 입력
 - 인스턴스 타입 : **"A1-2-STD"** 선택
 - 볼륨 크기 : **20** 입력
 - 키 페어 : 생성된 키페어 선택
 - * (옵션) 키페어: 이전에 생성한 키페어/신규 키페어 생성으로 만든 키페어 선택
 - **다음** 버튼 클릭
 - 네트워크 구성 : **"edu-vpc"** 선택
 - 서브넷 : **"edu-subnet-mgmt"** 선택
 - 시큐리티그룹 : 기존에 생성된 시큐리티 그룹 중 **"sg_mgmt"** 선택
 - **다음** 버튼 클릭
 - b. **만들기** 버튼 클릭
 - 2-4. 인스턴스 목록
 - a. 리스트에서 생성된 **"bastion"**을 클릭하여 상세 화면 이동

3. Public IP 할당

3-1. Console 화면

- a. 좌측 **메뉴** > **Computing** > **Virtual Machine** 로 이동

3-2. 인스턴스 목록

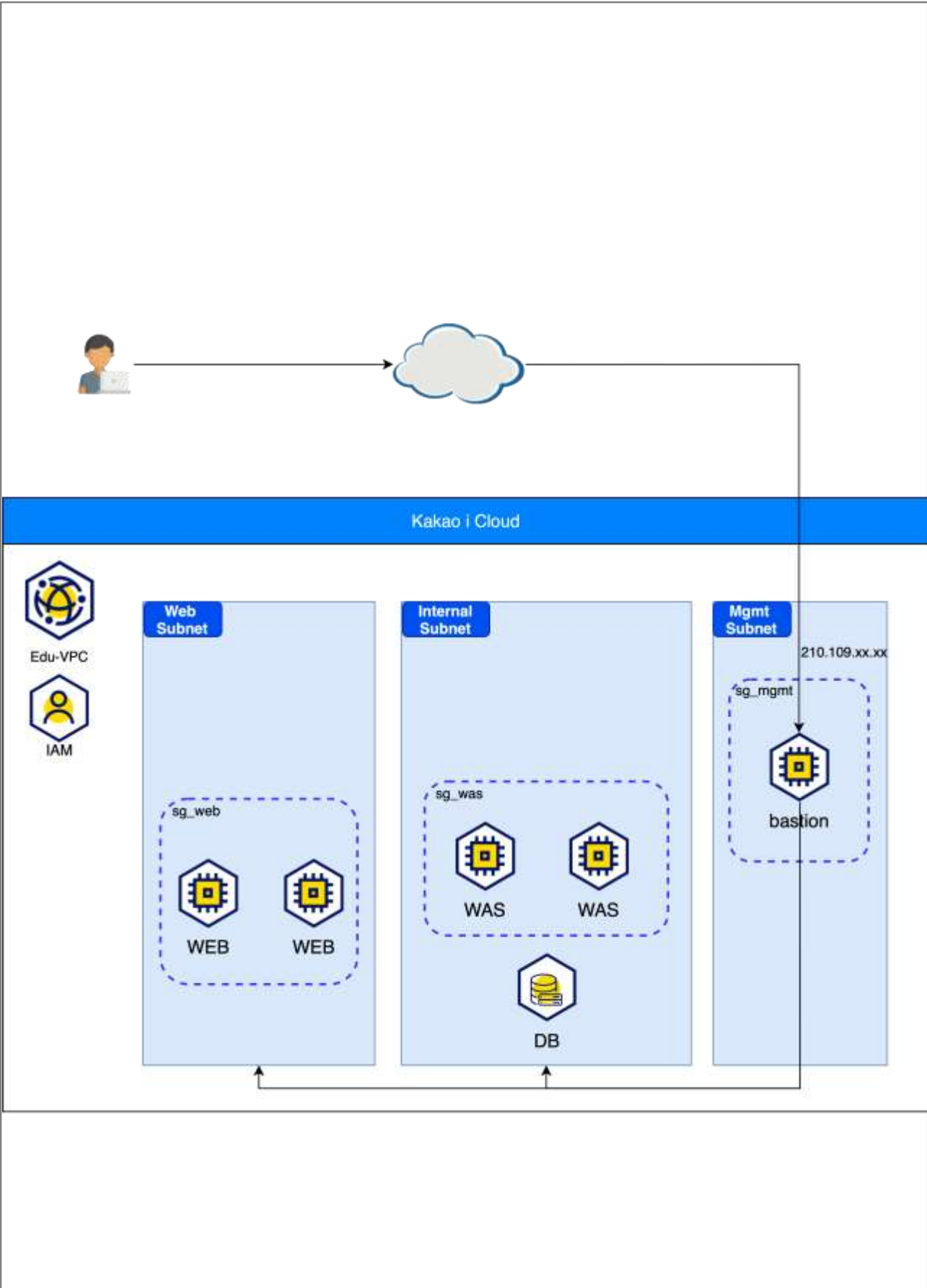
- a. **"bastion"** 인스턴스  액션(Action) 버튼 클릭
- b. Action 팝업에서 **"공인 IP 연결"** 선택

3-3. 공인 IP 연결 설정 화면

- 인스턴스 이름 : **"bastion"** 확인
- 공인 IP 할당 : **"새로운 공인 IP를 생성하고 자동으로 할당"** 선택
확인 버튼 클릭

3-4. 생성된 공인 IP 확인

실습 시나리오 (VM 생성 후 체크)



1. **과제**

이전 가이드(보안그룹 및 키페어, bastion VM 생성 및 고정IP 할당)를 참조하여 서비스 VM 생성해 보시오.

1-1. edu_subnet_web에 WEB VM 2대(VM사양은 최소사양, ubuntu, 보안 그룹은 sg_web)

1-2. edu_subnet_was에 WAS VM 2대(VM사양은 최소사양, ubuntu, 보안 그룹은 sg_was)

2. bastion VM 로그인

2-1. Public IP 확인

2-2. 인터넷 브라우저 탭 추가

- http://\${Public IP}:81 입력

a. 접속이 안될 시 보안 그룹 설정 확인 필요

3. bastion VM → Nginx Proxy Manager 설정

3-1. 로그인 화면

a. 초기 admin@example.com / changeme 로그인 후 계정 변경 가능

3-2. Dashboard 화면

a. 상단 메뉴에서 **Hosts > Streams**를 클릭

3-3. Streams 화면

a. 우측 상단의 **Add Stream** 클릭

- Incoming Port: **10001-10099** 숫자 중 하나를 선택하여 입력(e.g. 10001)

- Forward Host: bastion을 통해 접속할 VM의 **\${사실 IP}** 입력

- Forward Port: VM에 접속할 SSH 포트번호 **"22"**를 입력

b. **Save** 버튼을 클릭하여 저장

4. 외부에서 접속 시도

(Only Mac/Linux)

터미널에서 bastion Public IP와 설정한 포트번호(예 : 10001)로 접속 시도

```
$ ssh -i {개인키} ubuntu@{bastion의 공인 IP} -p10001  
ex) ssh -i harvey.pem ubuntu@210.109.43.99 -p10001
```

(Only Windows)

Putty를 사용하는 경우, 실습 교재 4 페이지를 참고하여 접속 시도

* 실습에 활용되는 ubuntu이미지의 SSH의 접근 계정은 "ubuntu"입니다.

1. Console 로그인
2. MySQL 인스턴스 그룹 생성
 - 2-1. Console 화면
 - a. 좌측 메뉴 > Data Store > MySQL 로 이동
 - 2-2. MySQL 화면
 - a. 우측 상단의 + 인스턴스 그룹 만들기 버튼 클릭
 - 2-3. 인스턴스 그룹 만들기
 - a. 기본 설정
 - 인스턴스 그룹 이름 : **librarydb** 입력
 - b. MySQL 설정
 - MySQL 사용자 이름 : **library** 입력
 - MySQL 비밀번호 : **\${비밀번호}** 입력
 - c. 인스턴스 가용성/개수
 - **“단일(Primary 1개)”** 선택
 - * **“고가용성(Primary 1개, Standby 1개)”**는 현재 중비중
 - d. 인스턴스 타입
 - **“A1-2-CO”** 선택
 - e. 스토리지 타입/크기
 - **100** GB 입력
 - f. VPC 구성 :
 - VPC : edu-vpc 선택 확인
 - 서브넷 : **“edu-subnet-was”** 선택
 - g. **생성하기** 버튼 클릭

1. 아래 실습은 각 WAS VM에 ssh로 접속한 상태에서 각각 진행합니다.

2. 환경 설치

1-1. MySQL Client 설치

```
$ sudo apt-get update
$ sudo apt-get install -y mysql-client
```

3. Mysql 환경 확인 구성

3-1. Mysql client가 설치된 WAS VM에서 다음을 실행합니다.

a. MySQL 접속

```
$ mysql -h ${MYSQL-Primary 엔드포인트} -u ${DB_USERNAME} -p
Enter password: ${DB_PASSWORD}

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 6917
Server version: 5.7.33-log MySQL Community Server (GPL)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

4. MySQL내 설치되어 있는 Databases를 확인한다.

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql>
```

5. WAS VM에 Application이 설치 후 이후 변화가 있는지 확인한다.

1. 아래 실습은 각 WEB VM에 ssh로 접속한 상태에서 각각 진행합니다.

2. 환경 설치

2-1. nginx 설치

```
$ sudo apt update
$ sudo apt install -y nginx
$ sudo systemctl start nginx
$ sudo systemctl enable nginx
```

3. nginx 환경 구성

3-1. 웹서버가 revers-proxy 역할로 정적 데이터 처리 하며, 동적데이터는 WAS instance로 전달하도록 웹서버 설정을 한다.

```
$ sudo rm /etc/nginx/sites-enabled/default
$ sudo vi /etc/nginx/conf.d/web.conf
-----아래 내용 붙여넣고 IP 수정 -----
server {
listen *:80;
server_name _;
access_log /var/log/nginx/access_library.log;
error_log /var/log/nginx/error_library.log;
location / {
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "Upgrade";
    proxy_buffering off;
    proxy_request_buffering off;
    proxy_hide_header X-Frame-Options;
    proxy_pass http://{WAS_LB_IP}:8080;
}
}
```

3-2. 웹서버를 재시작 합니다.

```
$ sudo systemctl restart nginx
```

1. 아래 실습은 각 WAS VM에 ssh로 접속한 상태에서 각각 진행합니다.

2. 환경 설치

1-1. java 설치

```
$ sudo apt-get update
$ sudo apt-get install -y openjdk-11-jdk
```

3. Application 배포 환경 구성

3-1. 각 WAS VM에 따라 build 및 배포 환경을 구성합니다.

a. WAS VM 1번, 2번에 명령어를 실행 합니다.

```
$ sudo mkdir -p /data/logs
$ sudo chown -R ubuntu:ubuntu /data
```

3-2. library app 레포지터리 가져오기(repo clone)

```
$ cd /data
$ git clone https://github.com/kep-cloudnative/library-monolithic.git
```

3-3. build 및 구동을 위한 script를 받는다.

```
$ cd /data
$ git clone https://github.com/kep-cloudnative/library-script.git script
```

3-4 mysql DB 생성시 설정값들에 맞게 값을 변경한다.

```
$ vi /data/script/app-env.sh
```

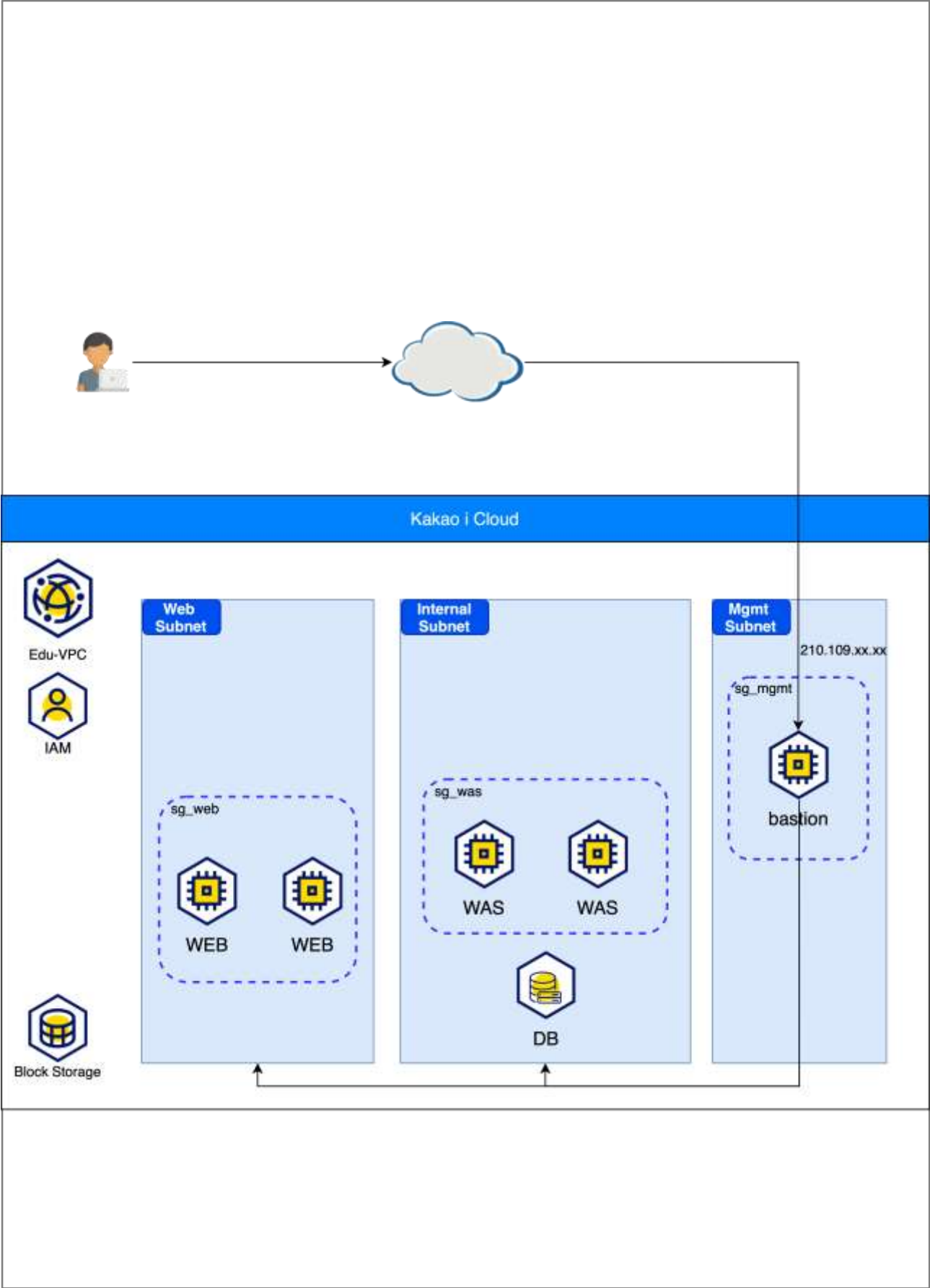
```
export PROFILE="dev"
export MYSQL_HOST="${MYSQL_ENDPOINT}"
export DB_USERNAME="${DB_USERNAME}"
export DB_PASSWORD="${DB_PASSWORD}"
```

3-5 application을 build를 진행 후 서버를 기동한다.

```
$ ./data/script/app-build.sh # 약 2분소요
$ ./data/script/start-app.sh
```

4. Application 실행 후 정상 작동하는지 확인한다.

실습 시나리오 (볼륨 추가 - 볼륨)



1. 볼륨 생성

1-1. Console 화면

- 좌측 메뉴 > **Computing** > **Virtual Machine** 으로 이동
- Volume 탭으로 이동
- 우측 상단의 **+ 볼륨 만들기** 버튼 클릭

1-2. 볼륨 만들기

- 볼륨 이름 : **add_volume** 볼륨 이름 작성
- 볼륨 타입/크기 : **10** GB 입력
- 만들기** 버튼을 클릭한 후 볼륨 리스트에서 확인

2. 볼륨을 VM에 연결

2-1. Volume 탭의 목록 화면

- 새로 생성된 “add_volume” 볼륨의 상태가 “**AVAILABLE**”인지 확인
- “add_volume” 해당 라인의 **:** 액션(Action) 클릭
- 볼륨 연결 설정** 선택

2-2. 볼륨 연결 설정화면

- 연결할 인스턴스: “**#{was #1}**” 서버 클릭
- 저장** 버튼 클릭

2-3. Volume 연결 확인

- Volume 목록에서 상태가 “add_volume” Volume의 상태가 “**IN-USE**”인지 확인
- 연결된 인스턴스 가 “**/dev/vdb**” 인지 확인

3. VM에 연결된 볼륨 확인

3-1. Virtual Machine 화면

- 상단에 Instance 탭으로 이동
- 2-2에서 연결한 Instance “\${was #1}” 선택
- Instance의 볼륨 탭에서 연결한 볼륨의 상태가 **In-Use**인지 확인

3-2. 파티셔닝 및 포맷, 마운트

- 이하 실습은 해당 VM에 ssh로 접속한 상태에서 진행
- ls -al /dev/vdb 하여 “vdb”파일이 있는지 확인

```
$ ls -al /dev/vdb  
brw-rw----. 1 root disk 253, 16 Jul 14 13:00 /dev/vdb
```

- fdisk를 이용하여 volume을 포맷

```
$ sudo fdisk /dev/vdb
```

Command (m for help): **n**

Partition type:

p primary (0 primary, 0 extended, 4 free)

e extended

Select (default p): **p**

Partition number (1-4, default 1): **1**

First sector (2048-209715199, default 2048): **(엔터)**

Using default value 2048

Last sector, +sectors or +size{K,M,G} (2048-209715199, default 209715199): **(엔터)**

Using default value 209715199

Partition 1 of type Linux and of size 100 GiB is set

Command (m for help): **w**

The partition table has been altered!

Calling ioctl() to re-read partition table.

Syncing disks.

d. ls -al /dev/ 하여 “vdb1”파일 확인

```
$ ls -al /dev | grep vdb1
brw-rw----. 1 root disk 253, 16 Jul 14 13:00 /dev/vdb1
```

e. /dev/vdb1의 파일 시스템의 정보 확인

```
$ sudo mkfs.xfs /dev/vdb1
meta-data=/dev/vdb1      isize=512    agcount=4, agsize=6553536 blks
                =          sectsz=512   attr=2, projid32bit=1
                =          crc=1        finobt=0, sparse=0
data        =          bsize=4096   blocks=26214144, imaxpct=25
                =          sunit=0     swidth=0 blks
naming      =version 2        bsize=4096   ascii-ci=0 ftype=1
log         =internal log    bsize=4096   blocks=12799, version=2
                =          sectsz=512   sunit=0 blks, lazy-count=1
realtime    =none           extsz=4096   blocks=0, rtextents=0
```

f. Volume을 연결할 Directory를 만들고 mount, 마운트된 볼륨 확인

```
$ sudo mount /dev/vdb1 /data/logs
```

```
$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
udev	7.8G	0	7.8G	0%	/dev
tmpfs	1.6G	1.3M	1.6G	1%	/run
/dev/vda1	97G	14G	84G	15%	/
tmpfs	7.9G	0	7.9G	0%	/dev/shm
tmpfs	5.0M	0	5.0M	0%	/run/lock
tmpfs	7.9G	0	7.9G	0%	/sys/fs/cgroup
/dev/loop0	68M	68M	0	100%	/snap/ixd/22753
/dev/loop1	48M	48M	0	100%	/snap/snapd/17029
/dev/loop2	64M	64M	0	100%	/snap/core20/1623
/dev/loop3	48M	48M	0	100%	/snap/snapd/17336
/dev/vda15	105M	5.2M	100M	5%	/boot/efi
tmpfs	1.6G	4.0K	1.6G	1%	/run/user/1000
/dev/vdb1	10G	104M	9.9G	2%	/data/logs


```
$ sudo chown -R ubuntu:ubuntu /data/logs
```

1. 스냅샷 생성

1-1. Console 화면

a. 좌측 **메뉴** > **Computing** > **Virtual Machine**으로 이동

1-2. Volume 탭 클릭

a. 볼륨 목록에서 연결된 인스턴스가 bastion의 /dev/vba 행에서  액션(Action) 클릭

1-3. 스냅샷 만들기

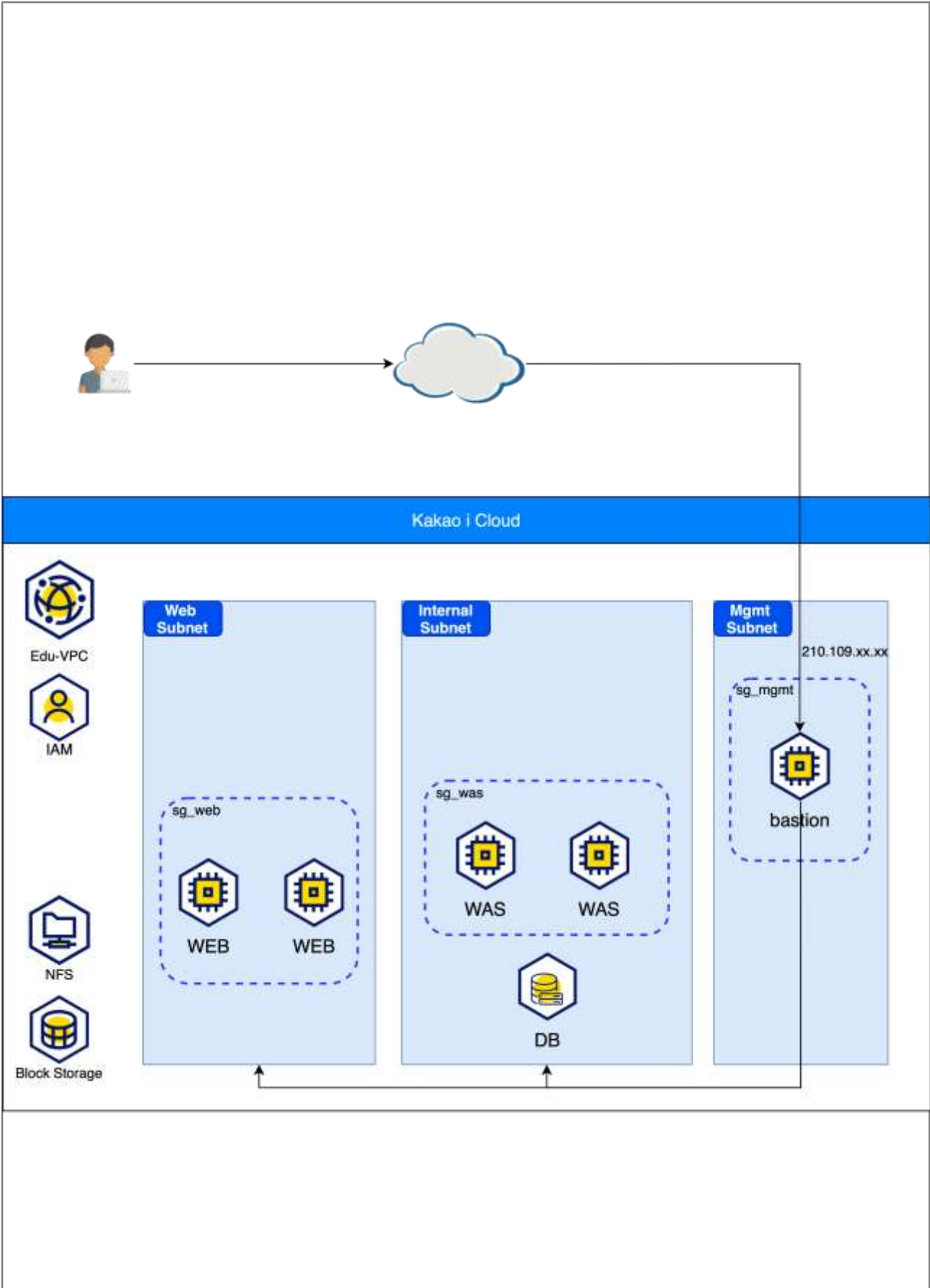
a. 생성 방법 : **"Full"** 선택

- Incremental(증분) 스냅샷 생성시 최초 Full 스냅샷 한 번이 필요합니다.

b. 스냅샷 이름 : **bastion-snapshot** 입력

1-4. **만들기** 버튼 클릭

실습 시나리오 (File Storage 생성)



1. File Storage 생성

1-1. Console 화면

- a. 좌측 메뉴 > **Storage > File Storage**으로 이동
- b. 우측 상단의 **+ 인스턴스 만들기** 버튼 클릭

1-2. 인스턴스 만들기

- a. 인스턴스 정보 작성
 - 인스턴스 이름 : **web_data** 볼륨 이름 작성
 - 프로토콜 : **"NFS"** 선택
 - 인스턴스 타입 : **"Basic"** 선택
 - 볼륨 크기: **1 TB** 입력
 - vpc : **"edu-vpc"** 선택
 - subnet : **"edu-subnet-web"** 선택
 - 접근 제어 설정 : **"설정된 VPC내 모든 사설 IP에서의 접근을 허용합니다."** 선택
- b. 마운트 정보 설정 : **webdata-01** 입력
- c. **생성하기** 버튼을 클릭한 후 File Storage 인스턴스 목록에서 확인

2. File Storage 를 가상 서버(Virtual Machine)에 연결

2-1. File Storage를 연결하고자 하는 가상서버의 SSH로 접근합니다.

- a. 19페이지 4-2의 SSH 접속을 참고하여 "web" 1번 가상서버에 접속
- b. NFS 패키지를 설치

```
$ sudo apt install -y nfs-common
```

- c. NFS Mount 하기

- 마운트 디렉토리 생성 및 확인

```
$ sudo mkdir /mnt/nfs-storage && ls -al /mnt
drwxr-xr-x 2 deploy deploy 4096 Dec 13 06:17 nfs-storage
```

- 마운트 및 확인

```
$ sudo mount -t nfs ${마운트 정보} /mnt/nfs-storage
```

```
$ mount | grep nfs-storage
172.16.236.212:/webdata-01 on /mnt/nfs-storage type nfs4 (rw,relatime,vers=4.2,...
```

- "web" 2번 가상서버에 접속하여 a~c 까지 진행합니다.

3. File Storage 에 파일을 만들어 확인

3-1. "web" 1번 가상서버에서 파일 생성

a. 18페이지 4-2의 SSH 접속을 참고하여 "web" 1번 가상서버에 접속

b. File Storage 마운트 위치에 파일 생성

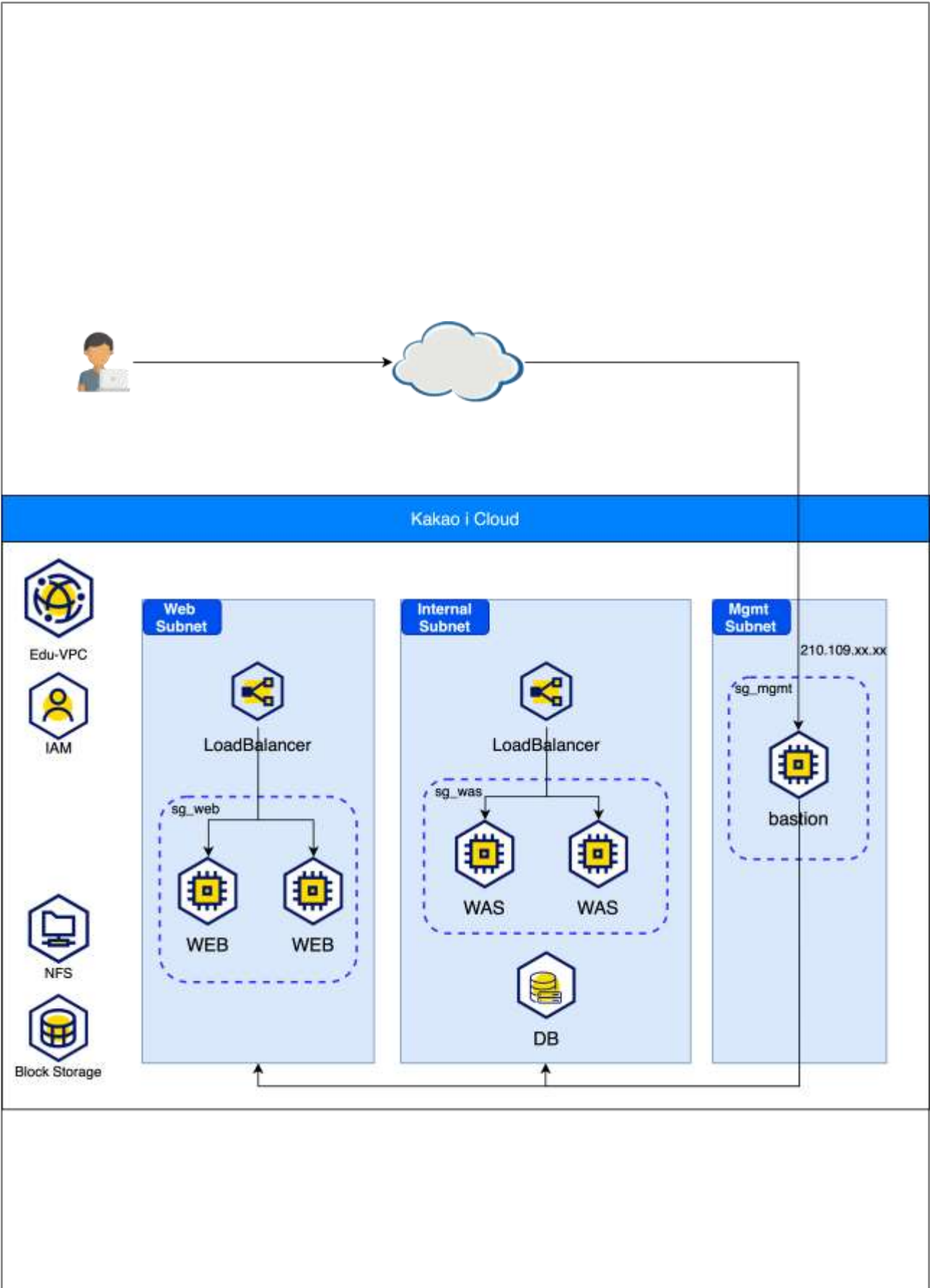
```
$ cd /mnt/nfs-storage  
$ sudo touch sample.txt
```

c. 19페이지 4-2의 SSH 접속을 참고하여 "web" 2번 가상서버에 접속

d. File Storage 마운트 위치에 파일 생성 확인

```
$ ls -al /mnt/nfs-storage  
total 24  
drwxr-xr-x 3 root root 4096 Oct 25 03:45 ./  
drwxr-xr-x 3 root root 4096 Oct 25 03:43 ../  
drwx----- 2 root root 16384 Oct 25 03:40 lost+found/  
-rw-r--r-- 1 root root 0 Oct 25 03:45 sample.txt
```


실습 시나리오 (로드밸런서 생성)



1. 로드 밸런서 생성

1-1. 좌측 메뉴 > **Networking > Load Balancer**로 이동

1-2. 우측 상단 **+ 로드밸런서 만들기** 버튼 클릭

1-2. 기본 정보 설정

a. 로드 밸런서 정보 입력

- 로드밸런서 이름: **was_lb** 입력

b. 네트워크 설정

- VPC : **"edu-vpc"** 선택

- 서브넷 : **"edu-subnet-was"** 선택

1-3. **다음** 버튼 클릭

2. 연결 정보 설정

a. 리스너/타겟 그룹 설정

- 리스너 프로토콜 : **"HTTP"** 선택

- 리스너 포트 : **8080** 포트 입력

- 타겟 그룹 프로토콜 : **"HTTP"** 선택

- 타겟 그룹 알고리즘 : **"Round Robin"** 선택

e. 헬스체크 설정

- 타입 : **"HTTP"** 선택

- HTTP 메서드 : **"GET"** 선택

- HTTP 버전 : **"1.1"** 선택

- HTTP 상태 코드 : **200** 입력

- 체크경로 : **/** 문자 입력

- 체크주기 : **10** 초 입력

- 타임아웃 : **5** 초 입력

- 상태전환 기준 (성공) : **3** 회 연속 입력

- 상태전환 기준 (실패) : **3** 회 연속 입력

f. **다음** 버튼 클릭

3. 연결 지원 설정(선택)

a. 연결자원 등록

- 연결할 자원에서 **“전체 선택”**을 선택
- 포트 설정 : 8080 입력 및 **등록** 버튼 클릭
- 연결 자원 : \${연결한 VM 이름} | \${Private IP} :8080 확인

b. **다음** 버튼 클릭

4. 검토

a. **만들기** 검토

b. [만들기] 버튼 클릭

1. 로드 밸런서 생성

1-1. 좌측 메뉴 > **Networking > Load Balancer**로 이동

1-2. 우측 상단 **+ 로드밸런서 만들기** 버튼 클릭

1-3. 기본 정보 설정

a. 로드 밸런서 정보 입력

- 로드밸런서 이름: **web_lb** 입력

b. 네트워크 설정

- VPC : **"edu-vpc"** 선택

- 서브넷 : **"edu-subnet-web"** 선택

1-4. **다음** 버튼 클릭

2. 연결 정보 설정

a. 리스너/타겟 그룹 설정

- 리스너 프로토콜 : **"HTTPS"** 선택
- 리스너 포트 : **443** 포트 확인 또는 입력
- 타겟 그룹 프로토콜 : **"HTTP"** 선택
- 타겟 그룹 알고리즘 : **"Round Robin"** 선택

b. SSL 설정

- SSL 인증서 : **"인증서 등록하기"** 선택

c. 인증서 다운로드

- <https://kep.li/4DNHRYw0ra>

d. 인증서 등록하기 팝업

- 인증서 프라이빗 키 : PEM 인코딩된 인증서 프라이빗 키를 입력해주세요
- 인증서 본문 : PEM 인코딩된 인증서 본문을 입력해 주세요.
- **추가** 버튼 클릭

e. SSL 설정

- 최소 TLS 버전 **"TLSv1.2"** 선택

f. 헬스체크 설정

- 타입 : **"HTTP"** 선택
- HTTP 메서드 : **"GET"** 선택
- HTTP 버전 : **"1.1"** 선택
- HTTP 상태 코드 : **200** 입력
- 체크경로 : **/** 문자 입력
- 체크주기 : **10** 초 입력
- 타임아웃 : **5** 초 입력
- 상태전환 기준 (성공) : **3** 회 연속 입력
- 상태전환 기준 (실패) : **3** 회 연속 입력
- **다음**

g. [다음] 버튼 클릭

3. 연결 자원 설정(선택)

a. 연결자원 등록

- 연결할 자원에서 **“전체 선택”**을 선택
- 포트 설정 : **80** 입력 및 [등록] 버튼 클릭
- 연결 자원 : \${연결한 VM 이름} | \${Private IP} :80 확인 (e.g. web-1|172.16.132.2:80)

b. **다음** 버튼 클릭

4. 검토

a. 전체 적인 검토

b. **만들기** 버튼 클릭

* 보안 프로그램이나 권한 문제로 인해 hosts 파일을 수정할 수 없는 경우, SSL 오프로딩을 확인하기 위해 HTTPS(443) 접속이 불가능합니다. HTTP(80)로 로드밸런서 접속은 가능.

4. SSL 오프로딩 확인

4-1. Local 환경에 SSL 교육 Domain 등록

a. (Only Mac/Linux)

```
$ sudo vi /etc/hosts
```

#/etc/hosts에 3-2의 Load balancer Public IP 와 교육 domain을 등록한다.
{LoadBalancer의 공인 IP} kic-edu.kep.com
ex) 210.109.42.123 edu-gov.kep.com

b. (Only Windows)

Windows의 경우 '윈도우키+R' → 실행창 → 열기에 "**wt.exe**" 입력 후

Ctrl + Shift + Enter

```
C:/> notepad c:\windows\system32\drivers\etc\hosts
```

#/etc/hosts에 3-2의 Load balancer Public IP 와 교육 domain을 등록한다.
{LoadBalancer의 공인 IP} kic-edu.kep.com
ex) 210.109.42.123 edu-gov.kep.com

또는 hosts 파일 수정하여 바탕화면에 저장 후 덮어쓰기 (관리자 권한 필요)

5. 웹서버 에서 로드밸런싱된 로그 확인

5-1. 2개의 WEB VM에 ssh로 접속한 상태에서 각각 아래 명령어 실행

```
$ sudo tail -f /var/log/messages
```

1. 아래 실습은 MySQL Client 설치 WAS VM에 ssh로 접속한 상태에서 각각 진행합니다.

2. MySQL 접속

1-1. MySQL 접속

```
$ mysql -h ${MYSQL-Primary 엔드포인트} -u ${DB_USERNAME} -p
Enter password: ${DB_PASSWORD}
```

3. 책 데이터 입력 및 확인

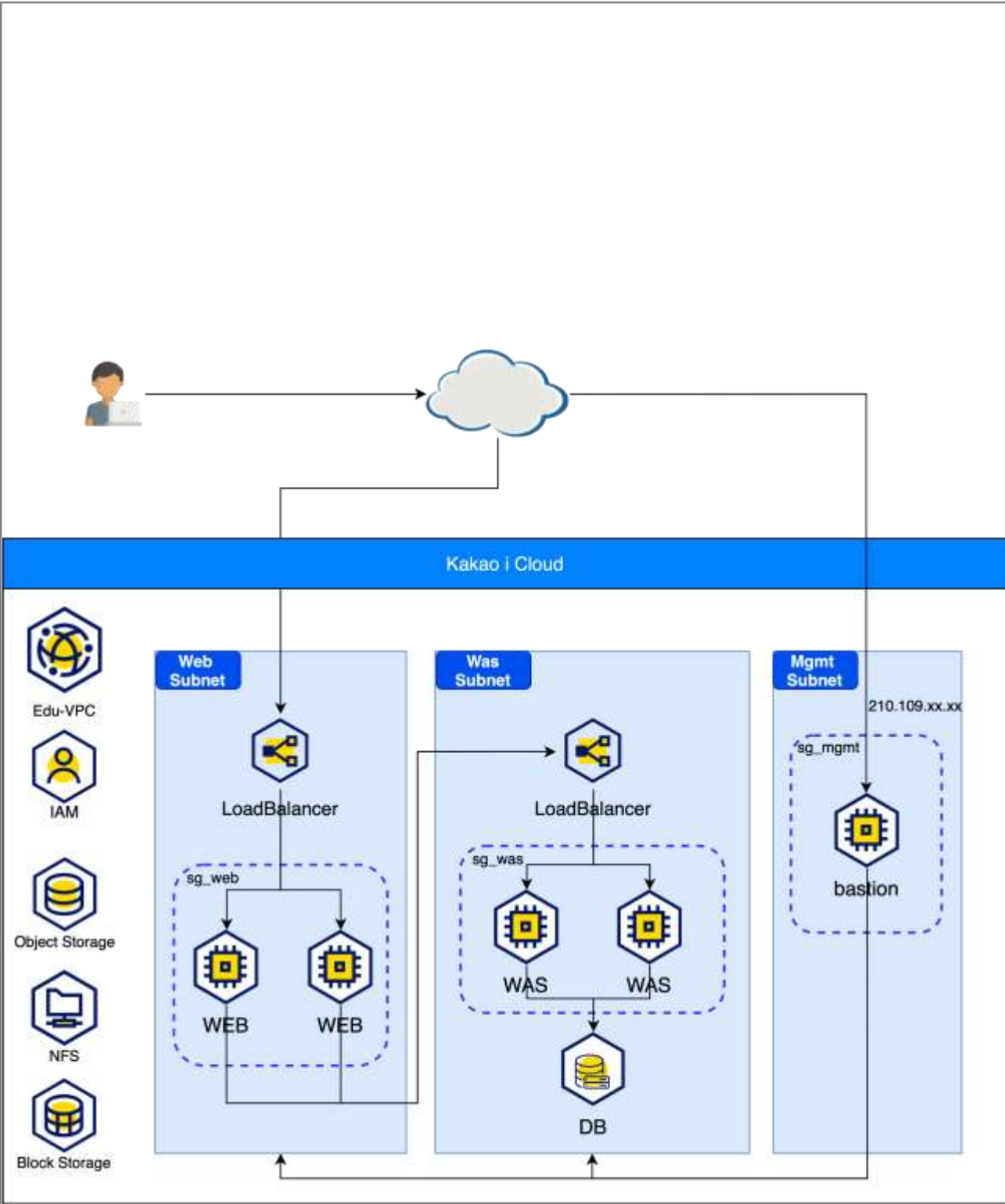
3-1. SAMPLE APP DB로 변경하여 추가 데이터를 입력한다.

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| librarydb |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql> use library;
Database changed
mysql> INSERT INTO book (book_Id, name, publish_Date, company, writer, status,
quantity, category, recommended, image_Url) VALUES ( 'COM.01.111', '클라우드
부트캠프', NOW(), 'kakaoenterprise','jeckie','최초등록', 5, '컴퓨터', true, '${OBJECT
URL}');
```

4. 브라우저의 SAMPLE APP화면에서 추가된 정보가 정상적으로 나오는지 확인한다.

실습 시나리오 (오브젝트 스토리지 생성)



1. 오브젝트 스토리지를 위한 버킷 생성 및 업로드

1-1. Console 화면

- a. 좌측 메뉴 > **Storage** > **Object Storage**로 이동

1-2. 오브젝트 스토리지(Object Storage) 화면

- a. 우측 상단 **+ 새 버킷 만들기** 버튼 클릭 시 버킷 생성 화면 팝업

1-3. 새 버킷 만들기 화면

- a. 타입 : **"HOT"** 선택
* **"Cold(자원 예정)"**는 현재 준비중
- b. 이름 : **obj-storage** 입력
- c. **만들기** 버튼을 클릭하여 생성
만들기

1-4. 생성된 버킷을 클릭해 버킷의 상세화면으로 이동

- a. **파일 업로드** 버튼을 클릭하여 테스트용 파일을 업로드

2. Life Cycle 설정 및 권한 설정

2-1. Object Storage 화면

- a. 상기에서 생성된 버킷을 클릭해 버킷의 상세화면으로 이동

2-2. 버킷 상세 화면

- a. 우측 상단 **Life Cycle 설정** 버튼 클릭 시 Life Cycle 설정 화면 팝업
- b. 대상 객체에서 **전체 객체** 또는 **Prefix 지정** 선택
(Prefix 지정 선택 시 Prefix 입력)
- c. 유지 기간에 일자를 입력하고 **정책 적용** 버튼을 클릭 후 **닫기** 버튼 클릭

2-3. 버킷 상세 화면

- a. [권한] 탭 클릭 후 [접근(Access-Control)] 탭으로 이동
- b. 우측의 [접근 설정] 버튼 클릭 시 접근 설정 화면 팝업
- b. 액세스 권한 항목에서 **퍼블릭 액세스 차단** 또는 **퍼블릭 액세스 허용** 선택
(퍼블릭 액세스 허용 선택 시, 허용할 IP 또는 CIDR을 추가 후, **저장** 버튼 클릭)
(구글검색에서 my ip로 확인된 공인 IP를 입력하여 테스트)

3. 오브젝트 스토리지 액세스

3-1. Object Storage 화면

a. 생성된 버킷 클릭 후 업로드 된 파일의 파일 정보 클릭하여 [URL 복사](#) | 버튼 클릭

3-2. 신규 브라우저 탭

a. 신규 브라우저 탭을 열어서 URL 붙여넣기로 업로드한 파일에 접근 확인

실습 시나리오 (최종형상)

