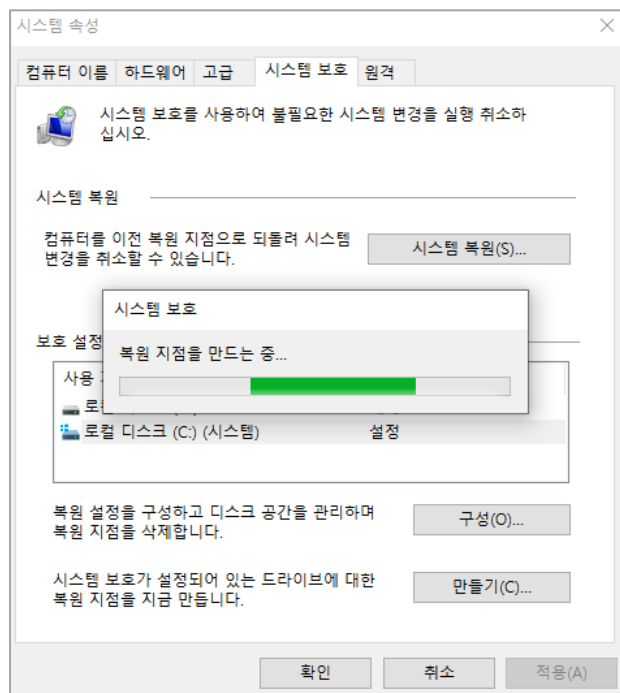


23-12-11

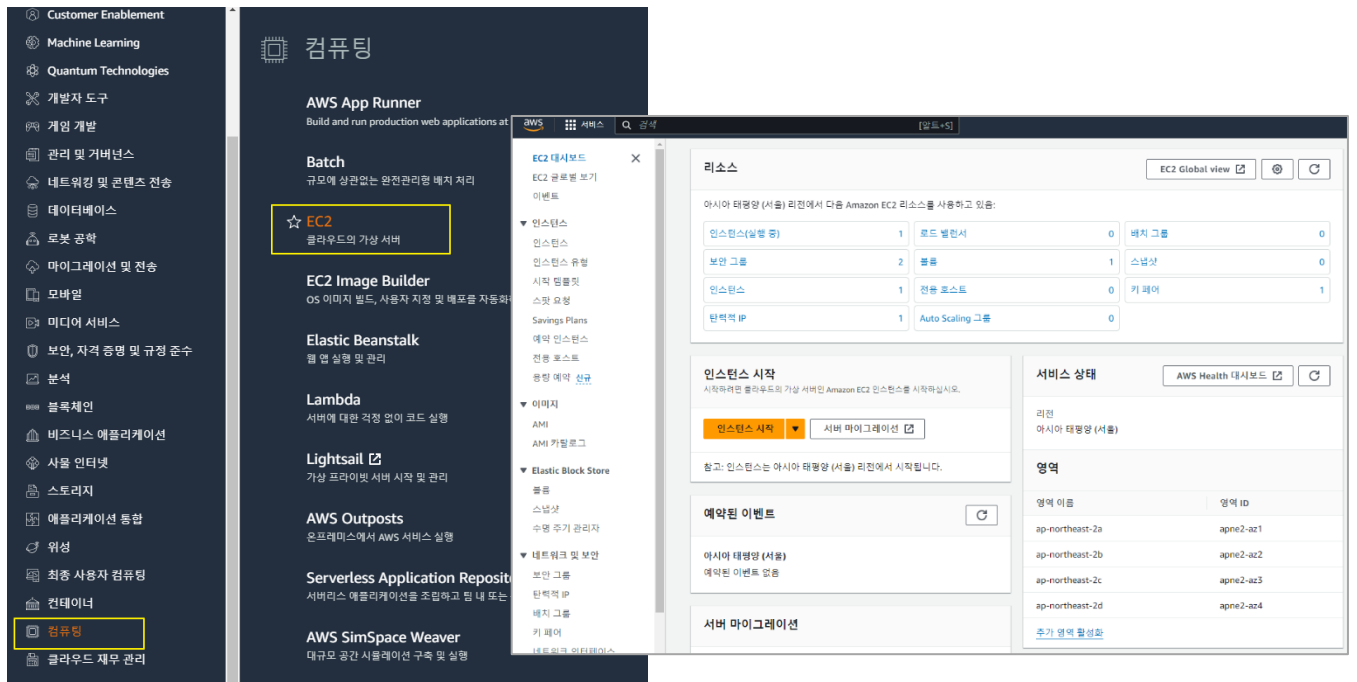
- 컴퓨터 포맷
- AWS 계정 생성
- Putty로 마리아 DB 설치

복원지점 만들기

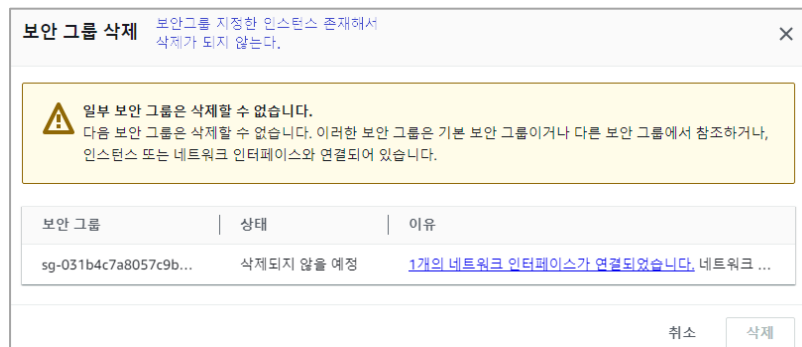


만들었다 12-12

- AWS



모든 프로그램 사용하지 않는다, EC2 사용



인스턴스 실행된 상태에서 보안그룹 삭제하려고 할 때 삭제되지 않는다

● 키페어 생성

키 페어 유형
정보

☒ RSA
☐ ED25519

Key pair type

Amazon EC2 supports ED25519 and RSA key pair types.

ED25519 limitations

- You can use ED25519 keys with Linux and Mac instances only.

자세히 알아보기

Amazon EC2 key pairs

리눅스와 맥 전용이라 ED25519는 건들이지 않는다,
맥을 사용하면 ED25519를 선택하고 생성해야 됨

프라이빗 키 파일 형식

☒ .pem

OpenSSH와 함께 사용

☐ .ppk

PuTTY와 함께 사용

윈도우로만 한다면ppk로 해도 된다 그러나 다른 사람에게 키를 줄 수 없다

➔ 우리는 pem으로 할 것

● 보안그룹 생성

인바운드 규칙

유형 정보

사용자 지정 TCP

프로토콜 정보

TCP

포트 범위 정보

3306

소스 정보

내 IP

규칙 추가

모범에 못한 ip주소이다
공유기에서 192.168. 이라는 주소로 바꿔서 주는 것

```

C:\Users\you>ipconfig

Windows IP 구성

이더넷 어댑터 이더넷:

   연결별 DNS 접미사. . . . . : Davolink
   링크-로컬 IPv6 주소 . . . . : fe80::8a42:3e95:6195:b7a8%11
   IPv4 주소 . . . . . : 192.168.219.107
   서브넷 마스크 . . . . . : 255.255.255.0
   기본 게이트웨이 . . . . . : 192.168.219.1

C:\Users\you>
  
```

포트 번호를3306(DB포트번호)으로 설정하고 소스를 내PC로 설정하면 위와 같은 번호가 찍히는데
그건 CMD에서 확인해보니 나의 ip가 아닌 것을 확인할 수 있음

➔ 모뎀에 쏜 ip주소이고 공유기에서 192.168.219.107로 바꿔주는 것

인바운드 규칙

유형 정보	프로토콜 정보	포트 범위 정보	소스 정보	설명 - 선택 사항 정보
사용자 지정 TCP	TCP	3306	Anywh... 어느곳이든 아이피 주소를 알면 접근가능 외부에서 접속가능하게 설정하려면 이렇게 해야됨	
			0.0.0.0/0 0.0.0.0/0 X	

규칙 추가

anywhere ipv4 모든 ipv4라는 것

VPC 정보
vpc-04d9e5d53be3b734f

인바운드 규칙

아무런 의미도 주반이로

유형 정보	프로토콜 정보	포트 범위 정보	소스 정보	설명 - 선택 사항 정보
사용자 지정 TCP	TCP	3306	Anywh... 0.0.0.0/0 0.0.0.0/0 X	
MySQL/Aurora	TCP	3306	사용자 ...	

규칙 추가

사용자지정해준 3306과 MySQL의 고정된 3306은 포트번호 같은 것이기 때문에
유형은 아무런 의미가 없다. 그래서 둘중 하나만 사용해주시면됨

⇒ 최종적으로 인바운드 규칙!

인바운드 규칙

유형 정보	프로토콜 정보	포트 범위 정보	소스 정보	설명 - 선택 사항 정보
MySQL/Aurora	TCP	3306	Anywh... 0.0.0.0/0 0.0.0.0/0 X	
SSH	TCP	22	Anywh... 0.0.0.0/0 0.0.0.0/0 X	
HTTP	TCP	80 아파치 웹서버 통신 서버	Anywh... 0.0.0.0/0 0.0.0.0/0 X	
HTTPS	TCP	443 보안교육에 대한 통신 포트 443일	Anywh... 0.0.0.0/0 0.0.0.0/0 X	
사용자 지정 TCP	TCP	8080 톰캣 서버	Anywh... 0.0.0.0/0 0.0.0.0/0 X	tomcat

규칙 추가

⇒ 최종적으로 아웃바운드 규칙!

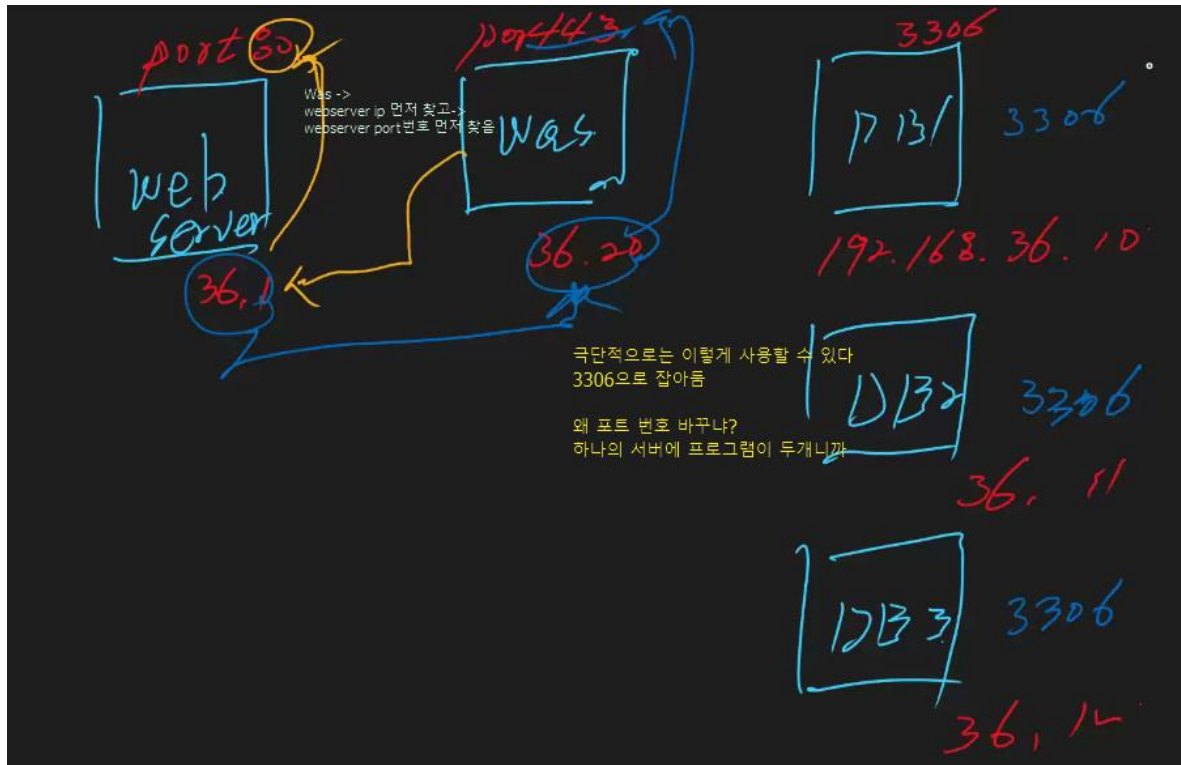
아웃바운드 규칙

유형 정보	프로토콜 정보	포트 범위 정보	대상 정보	설명 - 선택 사항 정보
모든 트래픽	전체	전체	Anywh... 0.0.0.0/0 0.0.0.0/0 X	

규칙 추가

기본적으로 위에는 존재해야 함
내부망 있어야 함다 그러면 사용자지정이 아니라
ip로 바뀌어야함

내부접속 안되게 하려고 하면
ip더먹만 내보내겠다고 설정하면 됨.
0.0.0.0/0 -> 모든 아이피 내보내겠다는 것



디비서버 와스 서버는 외부접속 허용 해야하는가?

이것과 탄력적 아이피랑은 상관 없다

4개가 있다면 탄력적 아이피는 4개가 필요

서버 4개를 설치하니까

그래서 aws 과금이 되기 시작한다 탄력적 아이피 개수가 늘어나니까

ip를 먼저 찾고 포트를 찾음

이전과 다르게 위에서 키생성과 보안그룹생성을 해둔 것을 선택해서 사용할 것(인스턴스 시작)

▼ 키 페어(로그인) 정보

키 페어를 사용하여 인스턴스에 안전하게 연결할 수 있습니다. 인스턴스를 시작하기 전에 선택한 키 페어에 대한 액세스 권한이 있는지 확인하세요.

키 페어 이름 - 필수

12-11-you

새 키 페어 생성

▼ 네트워크 설정 정보

네트워크 정보

vpc-05318589d19bf85bc

서브넷 정보

기본 설정 없음(가용 영역의 기본 서브넷)

퍼블릭 IP 자동 할당 정보

활성화

방화벽(보안 그룹) 정보

보안 그룹은 인스턴스에 대한 트래픽을 제어하는 방화벽 규칙 세트입니다. 특정 트래픽이 인스턴스에 도달하도록 허용하는 규칙을 추가합니다.

☐ 보안 그룹 생성
 ☒ 기존 보안 그룹 선택

일반 보안 그룹 정보

보안 그룹 선택

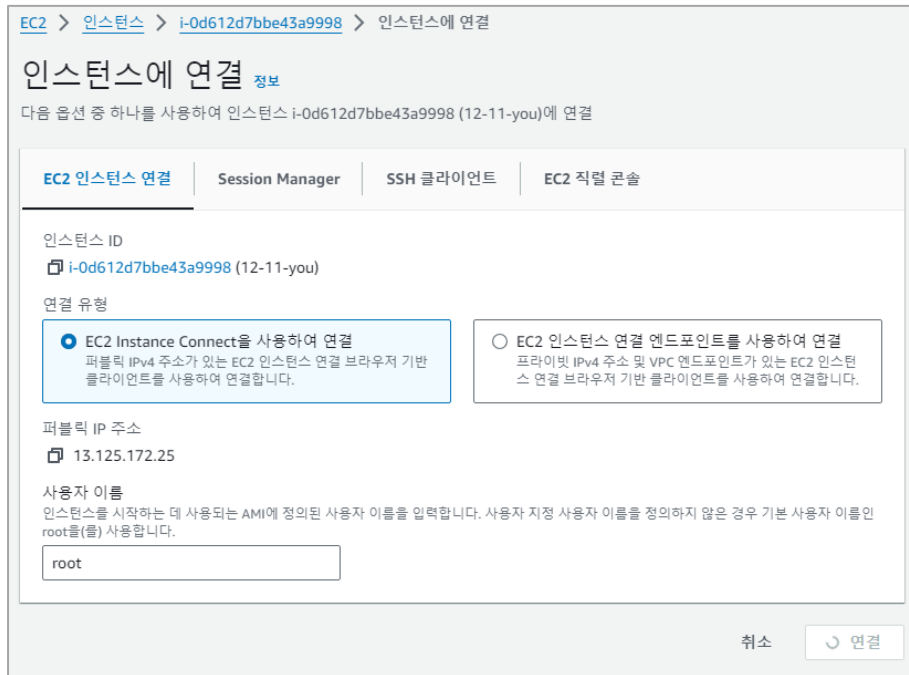
12-11-you sg-0678e5ffe8c2db1ee

VPC: vpc-05318589d19bf85bc

여기서 추가 또는 제거하는 보안 그룹은 모든 네트워크 인터페이스에서 추가 또는 제거됩니다.

편집

보안 그룹 규칙 비교



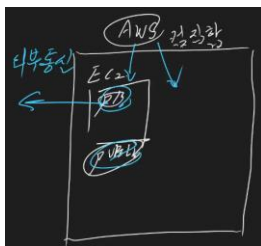
인스턴스 연결 부분

EC2 instance connect ->

EC2 인스턴스 연결 엔드포인트를 사용하여 연결 -> 공인아이피

공인아이피-외부와 접속가능한 아이피는 공인아이피

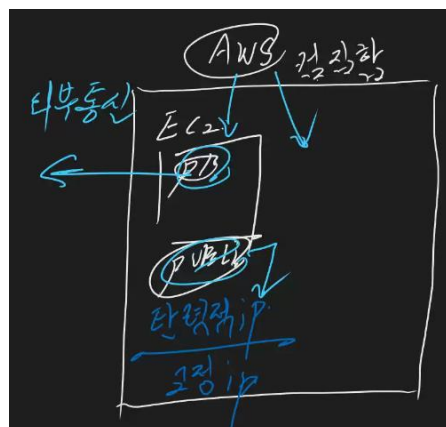
사설아이피-외부 접속과 불가능하다, 왜 사설아이피 만들었냐면 모든 사람에게 공인아이피를 줄 수 없기 때문에 (한정적) / 내부 인터넷 망을 사용하기 위해 내부 주소체계가 없으면 확인할 수 없다 그때 사용하는 것이 사설아이피



퍼블릭 아이피 받은 것 PI

AWS 는 컴퓨터의 집합체이다

그 안에 EC2를 설치한 것



PI 외부와 통신해야됨 -> 그러려면 공인 IP가 필요하다

IP 주소를 부여해야만 외부 인터넷과 통신이 가능하다

그것이 Public IP이다

외부와 연결이 되어야 하기 때문에 위에 EC2에 Public IP는 공인아이피이다.

그러면 탄력적 ip는 공인 ip가 됨

i-Od612d7bbe43a9998 (12-11-you)

PublicIPs: 13.125.172.25

PrivateIPs: 172.31.39.164

(공인아이피)

AWS 안에서만 사용하는 ip
(사설아이피)

Ping 공인아이피를 cmd에서 입력하면 ping이 간 것을 확인
(인바운드 규칙에서 ipv4 아이피 전체 가져오는 것 설정하기)

IPv4	모든 ICMP - IPv4	ICMP	전체	0.0.0.0/0
------	----------------	------	----	-----------

```

C:\Users\you>ping 43.201.213.38

Ping 43.201.213.38 32바이트 데이터 사용:
43.201.213.38의 응답: 바이트=32 시간=4ms TTL=48
43.201.213.38의 응답: 바이트=32 시간=3ms TTL=48
43.201.213.38의 응답: 바이트=32 시간=4ms TTL=48
43.201.213.38의 응답: 바이트=32 시간=13ms TTL=48

43.201.213.38에 대한 Ping 통계:
    패킷: 보냄 = 4, 받음 = 4, 손실 = 0 (0% 손실),
왕복 시간(밀리초):
    최소 = 3ms, 최대 = 13ms, 평균 = 6ms

C:\Users\you>

```

인스턴스 종료는 삭제

동적ip – 공유기 에서 남아있는 ip를 주고 컴퓨터를 끄면 그 아이피를 회수함

Ls -al 모든 리스트 보여라, 모든 목록 보여주라고 하는 것.

Parameters

Type of key to generate:

☒ RSA

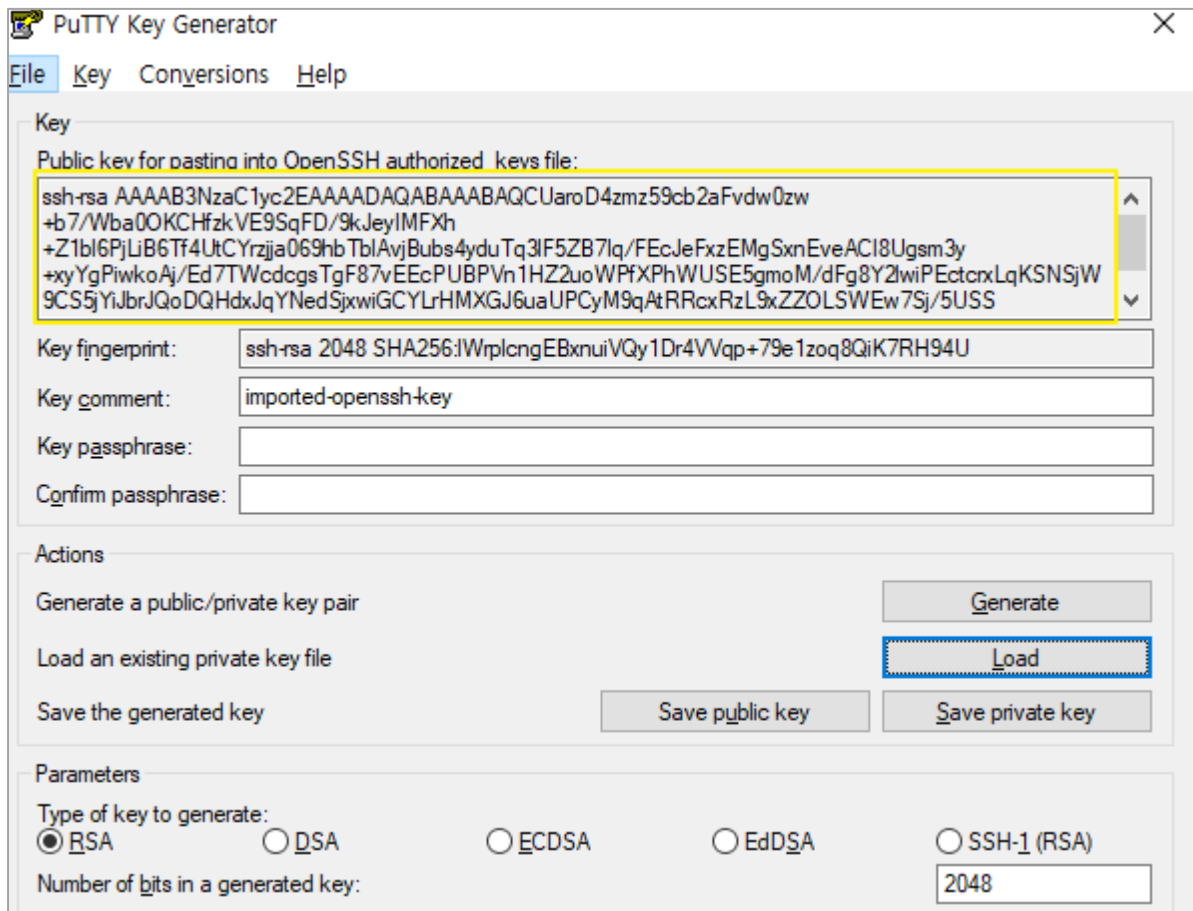
☐ DSA

☐ ECDSA

☐ EdDSA

☐ SSH-1 (RSA)

putty에서 키 생성해도 되지만 키 생성한 것을 가지고 올 것이다.

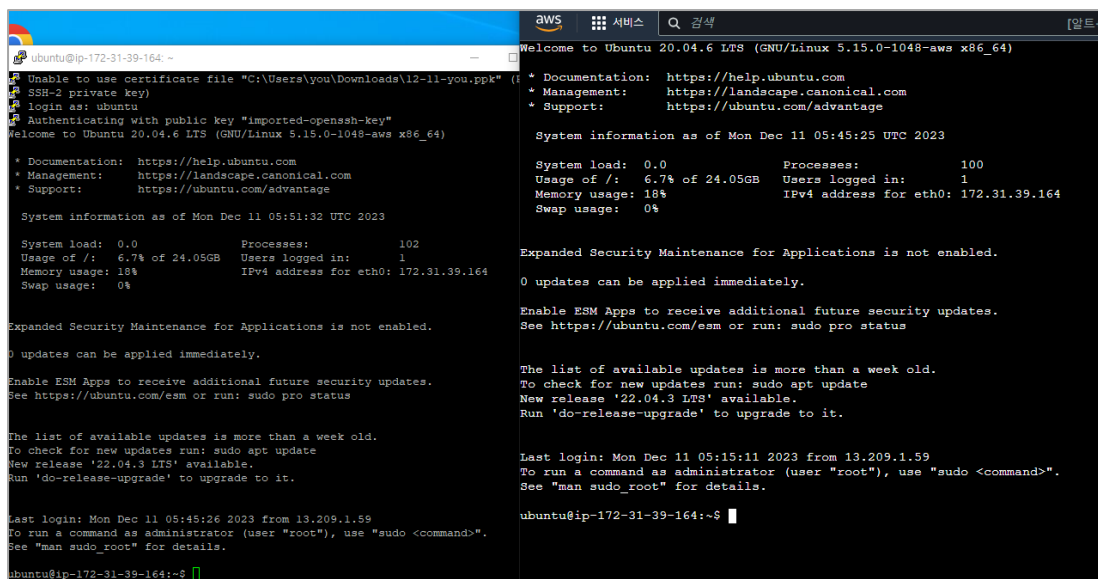


ssh 방식이고 A 부터 끝까지 다 암호이다 이걸 풀려면 슈퍼컴퓨터로 15년걸린다.

대칭키가 아닌 비대칭 키로 값이 늘어날 것

블록체인에서 최고의 값이다.

➔ Save private key



웹과 putty의 내용이 동일하다!

Webserver 5개 사용할 거면

5개를 꺼도 다시 사용해야 되니까 탄력적 ip를 사용한다.

웹에서 했던 것처럼 업데이트와 업그레이드 진행 ->

```
ubuntu@ip-172-31-39-164:~$ sudo apt-get install mariadb-server
```

이제 마리아 디비를 설치할 것 (최신 버전으로 설치)

운영체제 안에서 필요한 프로그램을 설치했다

```
ubuntu@ip-172-31-39-164:~$ cd /
ubuntu@ip-172-31-39-164:/$ ls -al
total 72
drwxr-xr-x 19 root root 4096 Dec 11 05:14 .
drwxr-xr-x 19 root root 4096 Dec 11 05:14 ..
lrwxrwxrwx 1 root root 7 Oct 25 21:51 bin -> usr/bin
drwxr-xr-x 4 root root 4096 Dec 11 06:05 boot
drwxr-xr-x 17 root root 3220 Dec 11 05:14 dev
drwxr-xr-x 97 root root 4096 Dec 11 06:10 etc
drwxr-xr-x 3 root root 4096 Dec 11 04:44 home
lrwxrwxrwx 1 root root 7 Oct 25 21:51 lib -> usr/lib
lrwxrwxrwx 1 root root 9 Oct 25 21:51 lib32 -> usr/lib32
lrwxrwxrwx 1 root root 9 Oct 25 21:51 lib64 -> usr/lib64
lrwxrwxrwx 1 root root 10 Oct 25 21:51 libx32 -> usr/libx32
drwx----- 2 root root 16384 Oct 25 21:54 lost+found
drwxr-xr-x 2 root root 4096 Oct 25 21:51 media
drwxr-xr-x 2 root root 4096 Oct 25 21:51 mnt
drwxr-xr-x 2 root root 4096 Oct 25 21:51 opt
dr-xr-xr-x 166 root root 0 Dec 11 05:14 proc
drwx----- 4 root root 4096 Dec 11 04:44 root
drwxr-xr-x 27 root root 940 Dec 11 06:10 run
lrwxrwxrwx 1 root root 8 Oct 25 21:51 sbin -> usr/sbin
drwxr-xr-x 8 root root 4096 Oct 25 21:56 snap
drwxr-xr-x 2 root root 4096 Oct 25 21:51 srv
dr-xr-xr-x 13 root root 0 Dec 11 05:14 sys
drwxrwxrwt 12 root root 4096 Dec 11 06:10 tmp
drwxr-xr-x 14 root root 4096 Oct 25 21:52 usr
drwxr-xr-x 13 root root 4096 Oct 25 21:53 var
ubuntu@ip-172-31-39-164:/$
```

중요한 것들은 노란색으로 표시해놨다

```
ubuntu@ip-172-31-39-164:/$ cd home
ubuntu@ip-172-31-39-164:/home$
```

루트 밑에 첫번째 디렉토리에 온 것(home)

```
ubuntu@ip-172-31-39-164:/home$ ls -al
total 12
drwxr-xr-x 3 root root 4096 Dec 11 04:44 .
drwxr-xr-x 19 root root 4096 Dec 11 05:14 ..
drwxr-xr-x 4 ubuntu ubuntu 4096 Dec 11 06:03 ubuntu
ubuntu@ip-172-31-39-164:/home$
```

검색해보니까 ubuntu 가 있다

```

ubuntu@ip-172-31-39-164:/home$ cd ubuntu
ubuntu@ip-172-31-39-164:~$ ls -al
total 28
drwxr-xr-x 4 ubuntu ubuntu 4096 Dec 11 06:03 .
drwxr-xr-x 3 root    root    4096 Dec 11 04:44 ..
-rw-r--r-- 1 ubuntu ubuntu 220 Feb 25 2020 .bash_logout
-rw-r--r-- 1 ubuntu ubuntu 3771 Feb 25 2020 .bashrc
drwx----- 2 ubuntu ubuntu 4096 Dec 11 04:57 .cache
-rw-r--r-- 1 ubuntu ubuntu 807 Feb 25 2020 .profile
drwx----- 2 ubuntu ubuntu 4096 Dec 11 04:44 .ssh
-rw-r--r-- 1 ubuntu ubuntu 0 Dec 11 06:03 .sudo_as_admin_successful
ubuntu@ip-172-31-39-164:~$

```

Ubuntu로 이동하면 다시 ~로 되는 것을 확인할 수 있다

ls -al 을 입력해주면 위와 같이 파일들이 나옴

☆ 아래 mysql -u root -p 로 변경하는 방법 나온다 (한번 더 해봤다 여기했을 때는 농침)

- Date 시간이 표준시간으로 되어 있는 것을 Asia/Seoul로 바꿔주는 방법

```

MariaDB [(none)]> select @@global.time_zone, @@session.time_zone;
+-----+-----+
| @@global.time_zone | @@session.time_zone |
+-----+-----+
| SYSTEM             | SYSTEM              |
+-----+-----+
1 row in set (0.000 sec)

```

select @@global.time_zone, @@session.time_zone;

(마리아 디비 안에서 어떻게 설정되어있는지 확인)

set global time_zone='Asia/Seoul'; set time_zone='Asia/Seoul';

```

ubuntu@ip-172-31-39-164:~$ sudo timedatectl set-timezone Asia/Seoul && sudo timedatectl set-ntp true
ubuntu@ip-172-31-39-164:~$ date
Mon Dec 11 16:10:42 KST 2023
ubuntu@ip-172-31-39-164:~$

```

sudo timedatectl set-timezone Asia/Seoul && sudo timedatectl set-ntp true

- Purge로 삭제하는 방법

```

Options marked [*] produce a lot of output - pipe it through 'less' or 'more' !
ubuntu@ip-172-31-39-164:~$ sudo dpkg -l | grep mariadb
ii  mariadb-client-10.3      1:10.3.38-0ubuntu0.20.04.1  amd64      MariaDB dat
base client binaries
ii  mariadb-client-core-10.3 1:10.3.38-0ubuntu0.20.04.1  amd64      MariaDB dat
base core client binaries
ii  mariadb-common           1:10.3.38-0ubuntu0.20.04.1  all        MariaDB com
on metapackage
ii  mariadb-server           1:10.3.38-0ubuntu0.20.04.1  all        MariaDB dat
base server (metapackage depending on the latest version)
ii  mariadb-server-10.3      1:10.3.38-0ubuntu0.20.04.1  amd64      MariaDB dat
base server binaries
ii  mariadb-server-core-10.3 1:10.3.38-0ubuntu0.20.04.1  amd64      MariaDB dat
base core server files
ubuntu@ip-172-31-39-164:~$

```

위에 하나하나 purge해주기

purge mariadb-server 이런식으로

```
ubuntu@ip-172-31-39-164:~$ sudo apt-get purge mariadb-server -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  galera-3 libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl
  libencode-locale-perl libfcgi-perl libhtml-parser-perl libhtml-tagset-perl
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl
  liblwp-mediatypes-perl libmysqlclient21 libsnappy1v5 libterm-readkey-perl libtimedate-perl
  liburi-perl mariadb-client-10.3 mariadb-client-core-10.3 mariadb-common mariadb-server-10.3
  mariadb-server-core-10.3 mysql-common socat
Use 'sudo apt autoremove' to remove them.
The following packages will be REMOVED:
  mariadb-server*
0 upgraded, 0 newly installed, 1 to remove and 4 not upgraded.
After this operation, 69.6 kB disk space will be freed.
(Reading database ... 62812 files and directories currently installed.)
Removing mariadb-server (1:10.3.38-0ubuntu0.20.04.1) ...
ubuntu@ip-172-31-39-164:~$
```

```
ubuntu@ip-172-31-39-164:~$ sudo dpkg -l | grep mariadb
ubuntu@ip-172-31-39-164:~$ sudo apt-get update
Hit:1 http://ap-northeast-2.ec2.archive.ubuntu.com/ubuntu
```

sudo dpkg -l | grep mariadb 하면 다 삭제된 거면 나오지 않음

이렇게 하고 처음부터 다시해보기

AWS 에서 인스턴스 보안그룹 키페어 탄력적IP를 각각 한 개씩 더 만들고 서로 ping 되는 것 확인

두번째 만든 것도 putty 따로 만들어서 mariadb설치함

```
*** System restart required ***
Last login: Mon Dec 11 15:47:50 2023 from 112.221.230.98
ubuntu@ip-172-31-39-164:~$ ping 3.34.179.217
PING 3.34.179.217 (3.34.179.217) 56(84) bytes of data.
64 bytes from 3.34.179.217: icmp_seq=1 ttl=63 time=0.524 ms
64 bytes from 3.34.179.217: icmp_seq=2 ttl=63 time=0.570 ms
64 bytes from 3.34.179.217: icmp_seq=3 ttl=63 time=0.582 ms
64 bytes from 3.34.179.217: icmp_seq=4 ttl=63 time=0.540 ms
64 bytes from 3.34.179.217: icmp_seq=5 ttl=63 time=0.556 ms
64 bytes from 3.34.179.217: icmp_seq=6 ttl=63 time=0.551 ms
64 bytes from 3.34.179.217: icmp_seq=7 ttl=63 time=0.607 ms
64 bytes from 3.34.179.217: icmp_seq=8 ttl=63 time=0.484 ms
64 bytes from 3.34.179.217: icmp_seq=9 ttl=63 time=0.493 ms
64 bytes from 3.34.179.217: icmp_seq=10 ttl=63 time=0.447 ms
64 bytes from 3.34.179.217: icmp_seq=11 ttl=63 time=0.498 ms
64 bytes from 3.34.179.217: icmp_seq=12 ttl=63 time=0.477 ms
64 bytes from 3.34.179.217: icmp_seq=13 ttl=63 time=0.546 ms
64 bytes from 3.34.179.217: icmp_seq=14 ttl=63 time=0.619 ms
^C64 bytes from 3.34.179.217: icmp_seq=15 ttl=63 time=0.474 ms
^C
--- 3.34.179.217 ping statistics ---
15 packets transmitted, 15 received, 0% packet loss, time 14316ms
rtt min/avg/max/mdev = 0.447/0.531/0.619/0.049 ms
ubuntu@ip-172-31-39-164:~$
```

● Putty 마리아 디비 설치

```
43.203.64.187 - PuTTY
Unable to use certificate file "C:\Users\you\Downloads\12-11-you2.ppk" (PuTTY
SSH-2 private key)
login as: ubuntu
```

```
ubuntu@ip-172-31-43-148:~$ sudo apt-get update
```

업데이트 sudo apt-get update

```
ubuntu@ip-172-31-43-148:~$ sudo apt-get upgrade
```

업그레이드 sudo apt-get upgrade

```
ubuntu@ip-172-31-43-148:~$ sudo apt-get install mariadb-server
```

마리아디비 서버 putty로 설치 sudo apt-get install mariadb-server

```
ubuntu@ip-172-31-43-148:~$ sudo mariadb
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 36
Server version: 10.3.38-MariaDB-0ubuntu0.20.04.1 Ubuntu 20.04
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]>
```

mysql -u root -p 를 하려고 하면 에러가 나서 그걸로 로그인 할 수 있도록 바꿔줄 것이다.

```
ubuntu@ip-172-31-43-148:~$ sudo mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL
SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user.  If you've just installed MariaDB,
you haven't set the root password yet, the password will be blank,
so you should just press enter here.
```

root비밀번호 설정하기

```
Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the
root user without the proper authorisation.

You already have a root password set, so you can safely answer 'n'

Change the root password? [Y/n]
```

비밀번호 설정하고 y 계속 누르기

3. MySQL Server root 비밀번호 설정

mysql server를 설치하면서 특별히 물어보는 것도 없었는데, root 비밀번호도 안들어보고 설치가 진행되었습니다.

조기 root 비밀번호를 설정하는대는 크게 두가지 정도 있습니다.

첫번째 방법은 mysql_secure_installation 명령어를 통해서 root 비밀번호를 설정할 수 있습니다.

```
$ sudo mysql_secure_installation
```

mysql_secure_installation을 실행시키면,

- 1) 첫번째 질문에 VALIDATE PASSWORD PLUGIN을 사용할지 물어봅니다.
- 2) root 비밀번호를 묻으면, 원하는 root 비밀번호를 동일하게 두번 입력해 줍니다.
- 3) Anonymous User를 삭제할지 묻으면 Y를 입력해서 삭제해 줍니다.
- 4) 외부에서 root 계정으로 접근을 허용할지 묻습니다.
- 5) Test DB를 삭제할지 묻으면 필요 없으니 Y를 입력해서 삭제해 줍니다.
- 6) 마지막으로 privileges table을 reload 할지 묻으면 Y를 입력해서 reload 해줍니다.

두번째 방법은 이전에 포스팅 해놓은 내용이 있어서 이 포스팅을 참조하시면 쉽게 하실 수 있습니다!

```
ubuntu@ip-172-31-43-148:~$ sudo mariadb
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 45
Server version: 10.3.38-MariaDB-0ubuntu0.20.04.1 Ubuntu 20.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]>
```

마리아디비로 들어와서

```
MariaDB [(none)]> select host, user, plugin from mysql.user;
+-----+-----+-----+
| host      | user | plugin      |
+-----+-----+-----+
| localhost | root | unix_socket |
+-----+-----+-----+
1 row in set (0.000 sec)
```

Plugin이 어떤 상태인지 확인함

```
MariaDB [(none)]> update mysql.user set plugin='mysql_native_password' where user='root';
Query OK, 1 row affected (0.000 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

바꿔주기 mysql -u root -p로 바꿔주는 명령어

```
MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.000 sec)
```

마리아 설치후 구문

```
MariaDB [(none)]> exit
Bye
ubuntu@ip-172-31-43-148:~$ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 46
Server version: 10.3.38-MariaDB-0ubuntu0.20.04.1 Ubuntu 20.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

이후 되는 것을 확인할 수 있다.

sudo mariadb로 mariadb를 들어왔다면 -> 변경해준 뒤로는 ->

mysql -u root -p로 들어올 수 있게 바꿔줌 / 그랬을 때 plugin이 어떻게 보이는지 알 수 있음

```
MariaDB [(none)]> select host, user, plugin from mysql.user;
+-----+-----+-----+
| host      | user | plugin      |
+-----+-----+-----+
| localhost | root | unix_socket |
+-----+-----+-----+
1 row in set (0.000 sec)

MariaDB [(none)]> update mysql.user set plugin='mysql_native_password' where user='root';
Query OK, 1 row affected (0.000 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [(none)]> select host, user, plugin from mysql.user;
+-----+-----+-----+
| host      | user | plugin      |
+-----+-----+-----+
| localhost | root | mysql_native_password |
+-----+-----+-----+
1 row in set (0.000 sec)
```