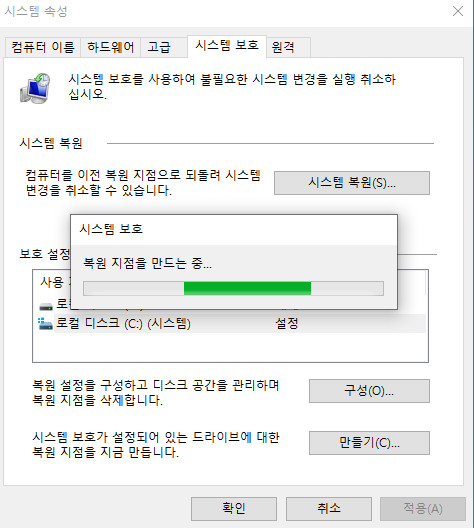
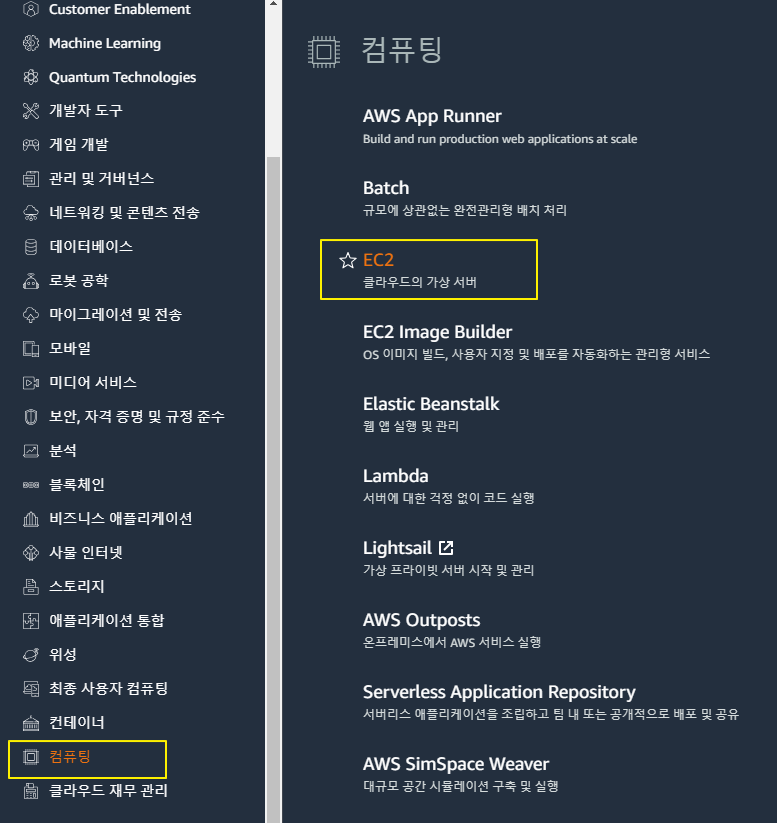
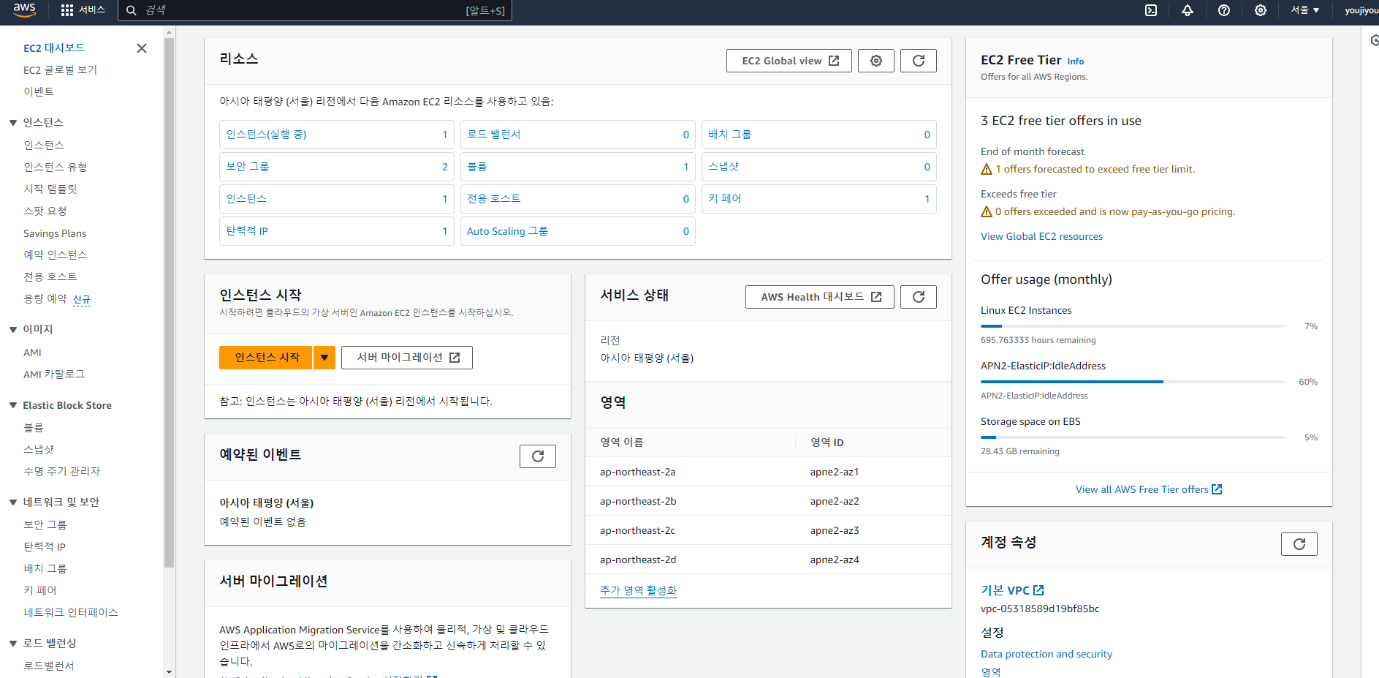
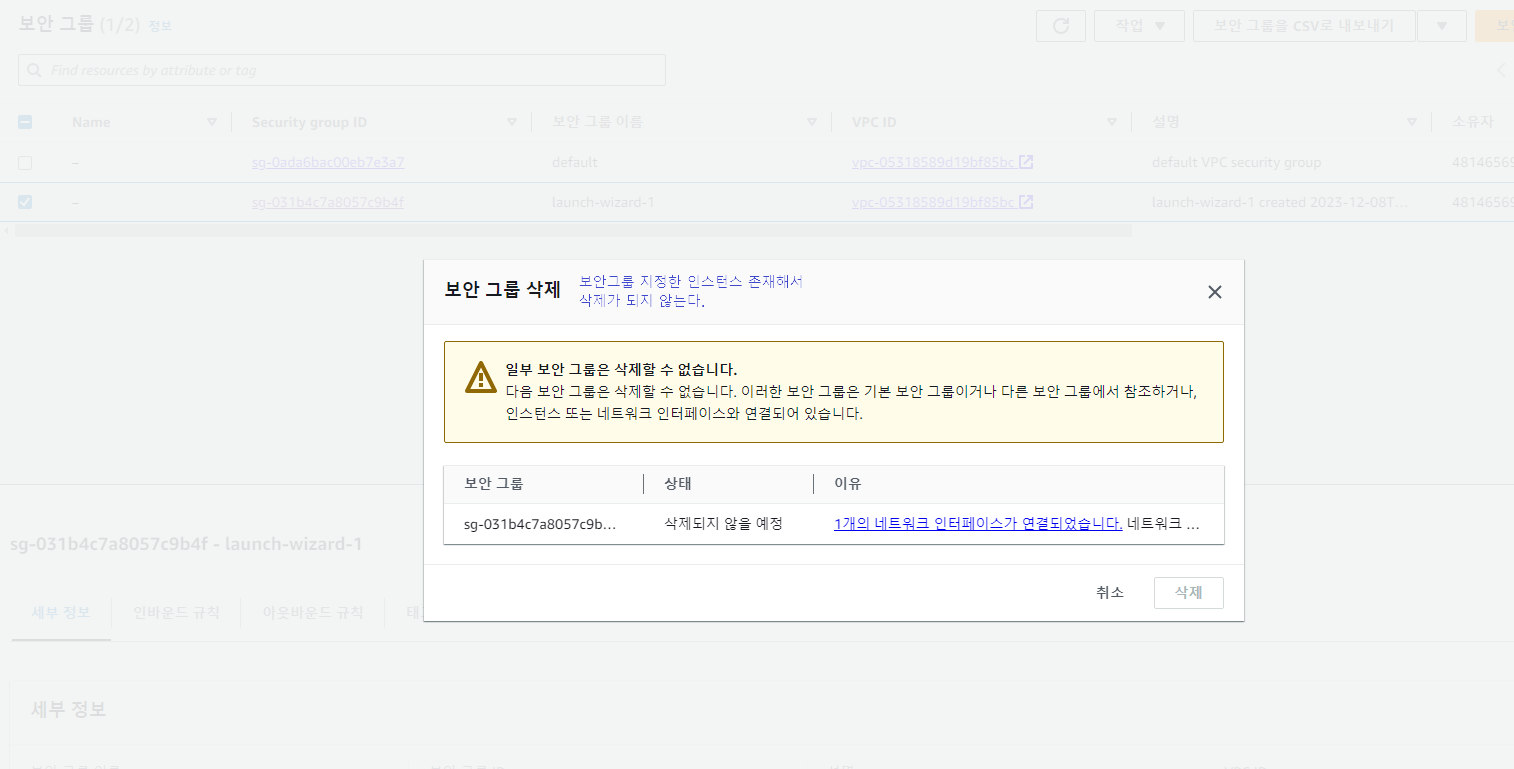
23-12-11

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

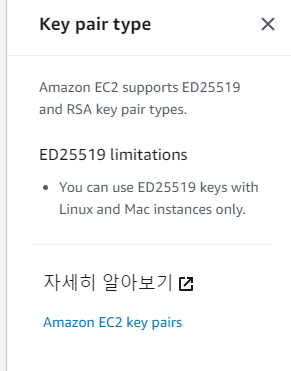
복원지점 만들기  
  
만들었다 12-12

* AWS

  
모든 프로그램 사용하지 않는다, EC2 사용

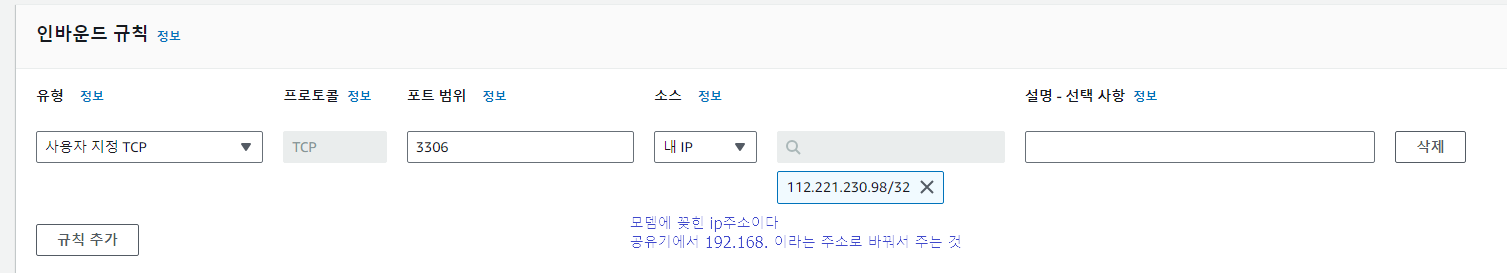
  
인스턴스 실행된 상태에서 보안그룹 삭제하려고 할 때 삭제되지 않는다

* 키페어 생성

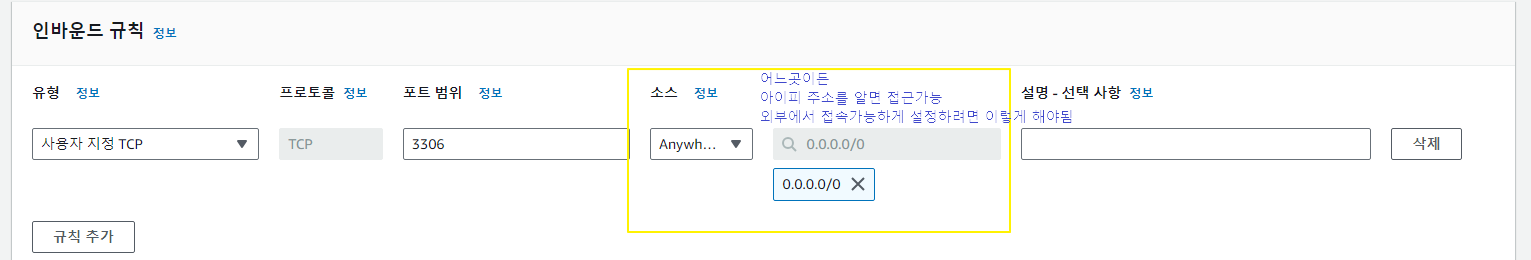
  
리눅스와 맥 전용이라 ED25519는 건들이지 않는다,   
맥을 사용하면 ED25519를 선택하고 생성해야 됨

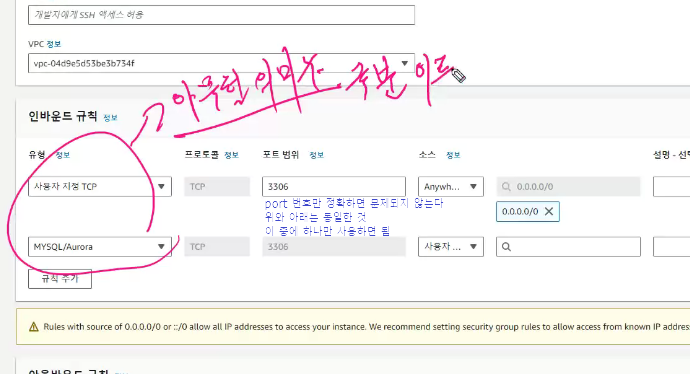
  
윈도우로만 한다면ppk로 해도 된다 그러나 다른 사람에게 키를 줄 수 없다

* 우리는 pem으로 할 것
* 보안그룹 생성

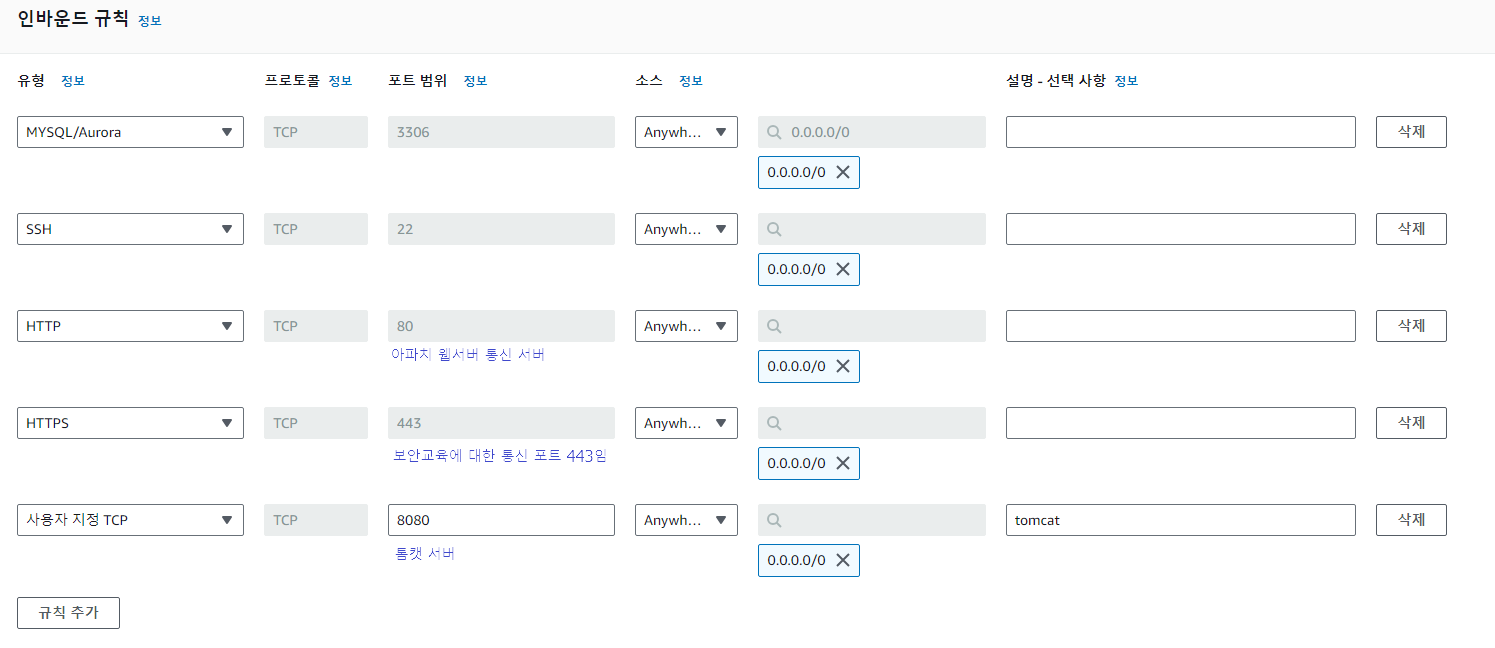
  
포트 번호를3306(DB포트번호)으로 설정하고 소스를 내PC로 설정하면 위와 같은 번호가 찍히는데  
그건 CMD에서 확인해보니 나의 ip가 아닌 것을 확인할 수 있음

* 모뎀에 꽂힌 ip주소이고 공유기에서 192.168.219.107로 바꿔주는 것

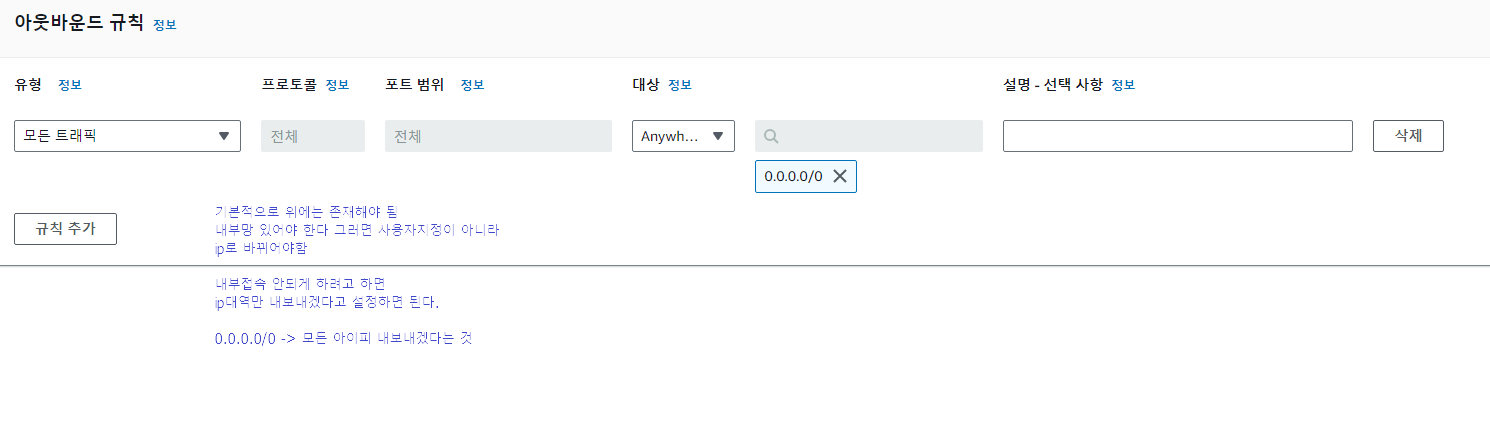
  
anywhere ipv4 모든 ipv4라는 것

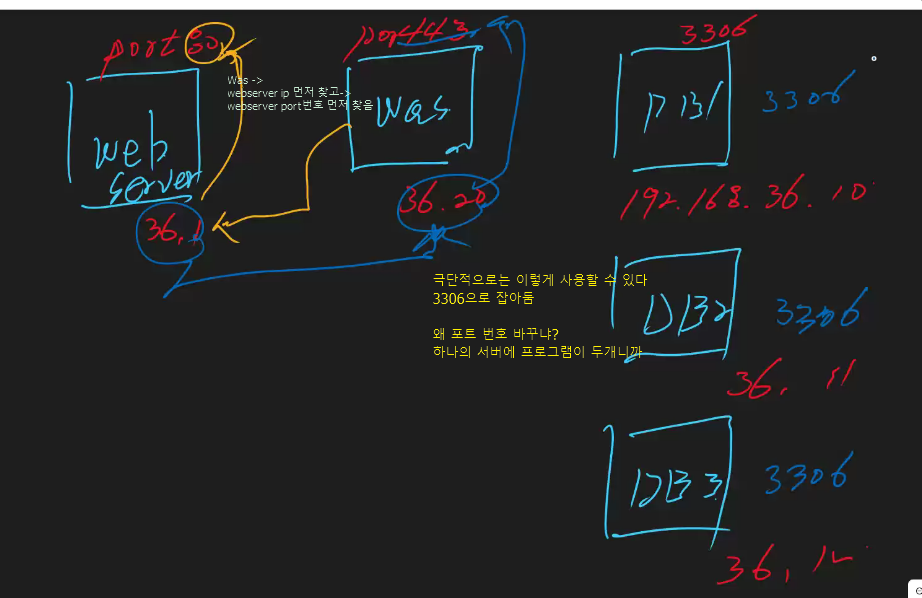
  
사용자지정해준 3306과 MYSQL의 고정된 3306은 포트번호 같은 것이기 때문에  
유형은 아무런 의미가 없다. 그래서 둘중 하나만 사용해주면됨

* 최종적으로 인바운드 규칙!



* 최종적으로 아웃바운드 규칙!



  
디비서버 와스 서버는 외부접속 허용 해야하는가?

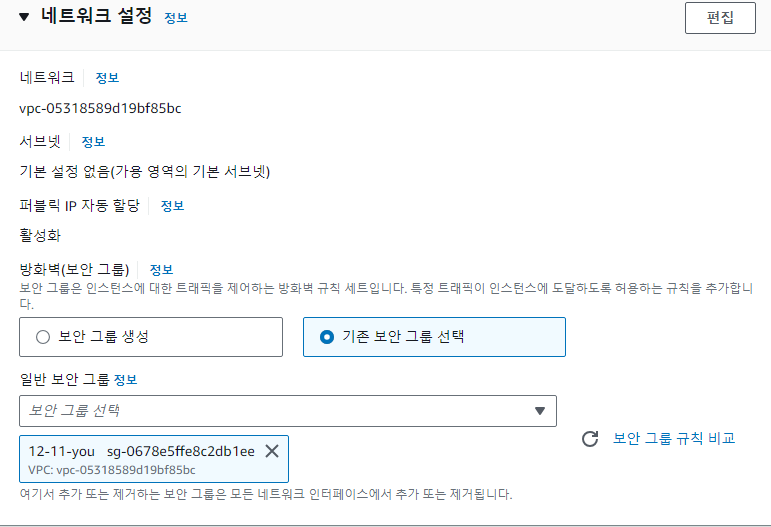
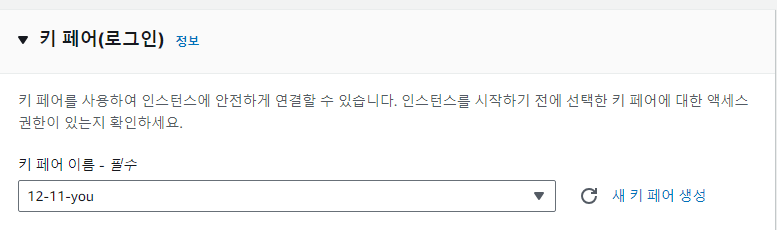
이것과 탄력적 아이피랑은 상관 없다

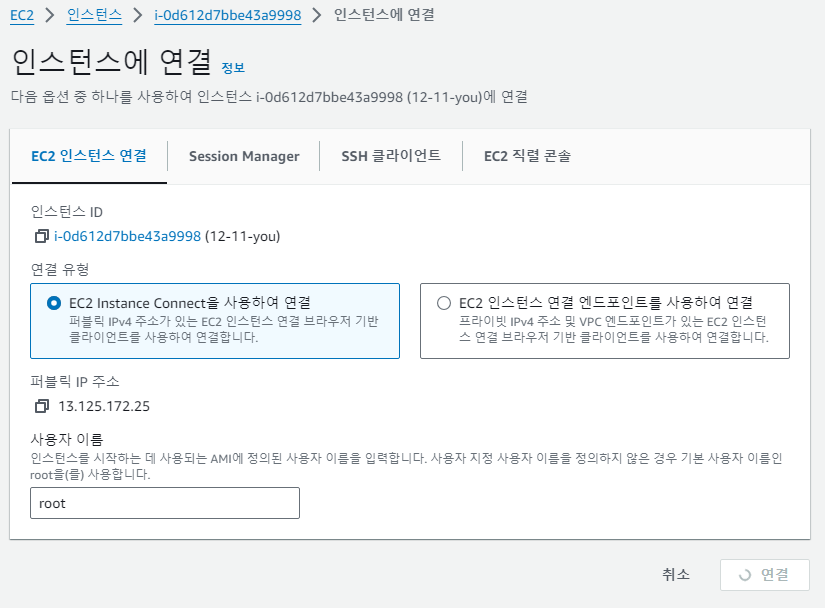
4개가 있다면 탄력적 아이피는 4개가 필요

서버 4개를 설치하니까

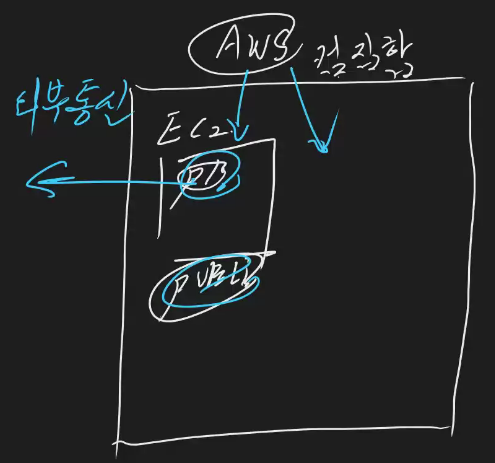
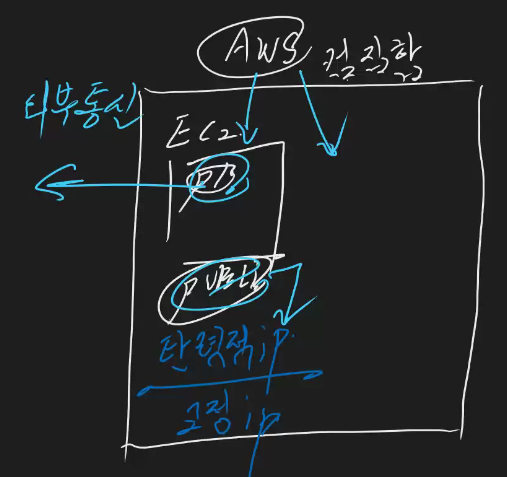
그래서 aws 과금이 되기 시작한다 탄력적 아이피 개수가 늘어나니까

Ip를 먼저 찾고 포트를 찾음

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


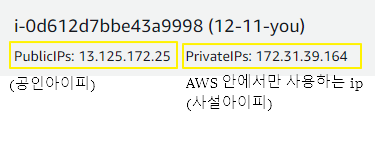
  
인스턴스 연결 부분  
EC2 instance connect ->   
EC2 인스턴스 연결 엔드포인트를 사용하여 연결 -> 공인아이피  
공인아이피-외부와 접속가능한 아이피는 공인아이피

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



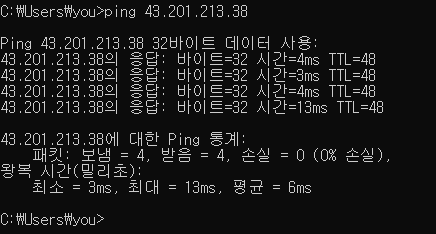
퍼블릭 아이피 받은 것 PI  
AWS 는 컴퓨터의 집합체이다  
그 안에 EC2를 설치한 것  
  
PI 외부와 통신해야됨 -> 그러려면 공인 IP가 필요하다  
IP 주소를 부여해야만 외부 인터넷과 통신이 가능하다  
그것이 Public IP이다   
외부와 연결이 되어야 하기 때문에 위에 EC2에 Public IP는 공인아이피이다.

그러면 탄력적 ip는 공인 ip가 됨



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





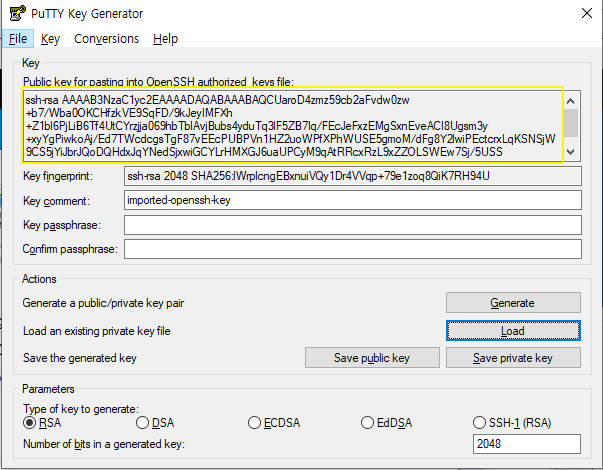
인스턴스 종료는 삭제

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

Ls -al 모든 리스트 보여라, 모든 목록 보여주라고 하는 것.

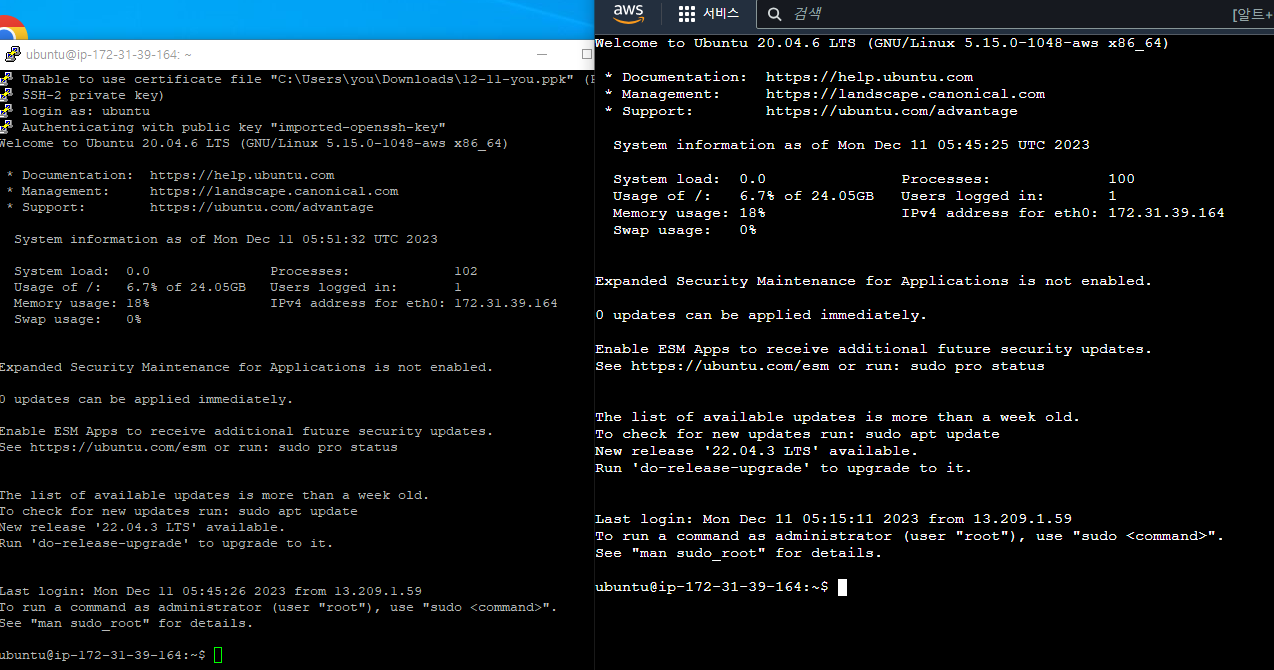


putty에서 키 생성해도 되지만 키 생성한 것을 가지고 올 것이다.

  
ssh 방식이고 A 부터 끝까지 다 암호이다 이걸 풀려면 슈퍼컴퓨터로 15년걸린다.  
대칭키가 아닌 비대칭 키로 값이 늘어날 것

블록체인에서 최고의 값이다.

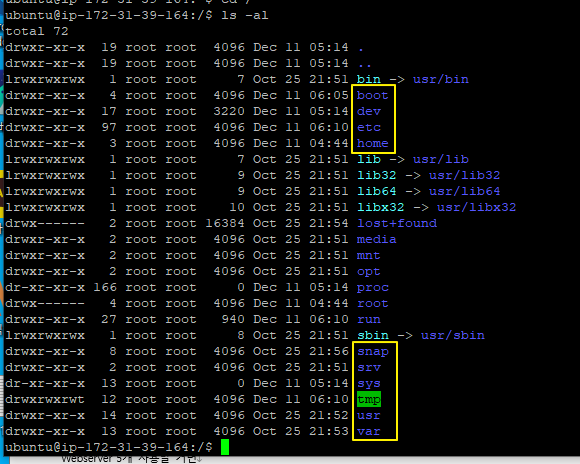
* Save private key

  
웹과 putty의 내용이 동일하다!

Webserver 5개 사용할 거면  
5개를 꺼도 다시 사용해야 되니까 탄력적 ip를 사용한다.

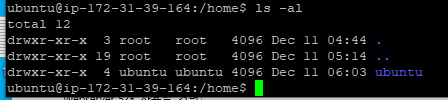
웹에서 했던 것처럼 업데이트와 업그레이드 진행 ->  
sudo apt-get update  
sudo atp-get upgrade

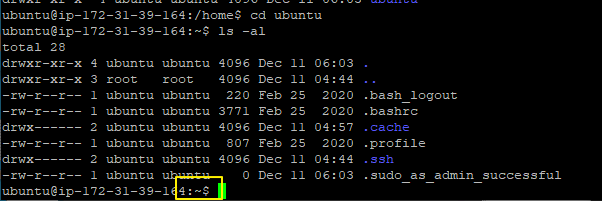
  
이제 마리아 디비를 설치할 것 (최신 버전으로 설치)  
**운영체제 안에서 필요한 프로그램을 설치했다**

  
중요한 것들은 노란색으로 표시해놨다



루트 밑에 첫번째 디렉토리에 온 것(home)

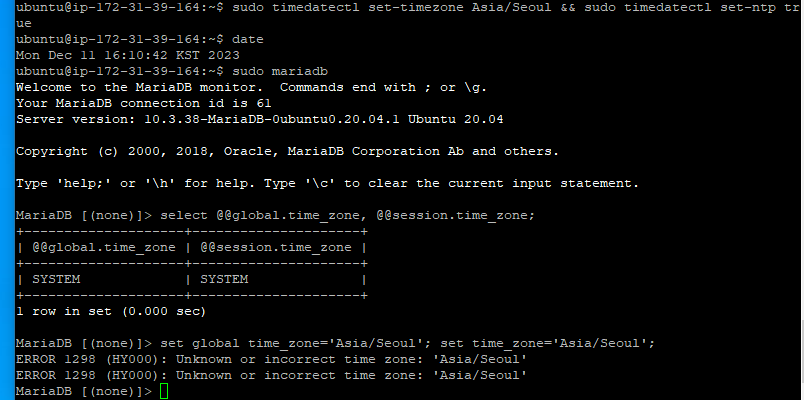
  
검색해보니까 ubuntu 가 있다



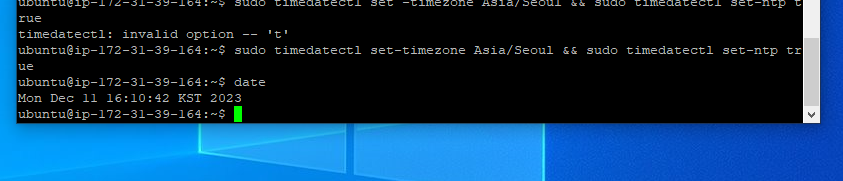
Ubuntu로 이동하면 다시 ~로 되는 것을 확인할 수 있다  
ls -al 을 입력해주면 위와 같이 파일들이 나옴

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

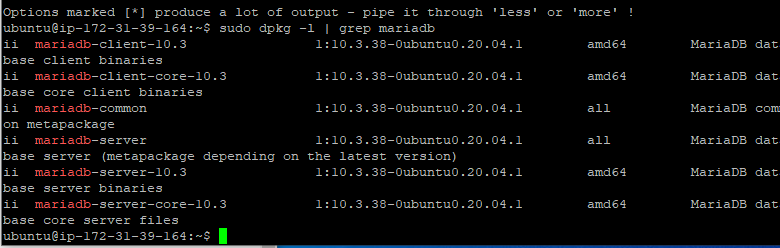
* Date 시간이 표준시간으로 되어 있는 것을 Asia/Seoul로 바꿔주는 방법

  
select @@global.time\_zone, @@session.time\_zone;   
(마리아 디비 안에서 어떻게 설정되어있는지 확인)

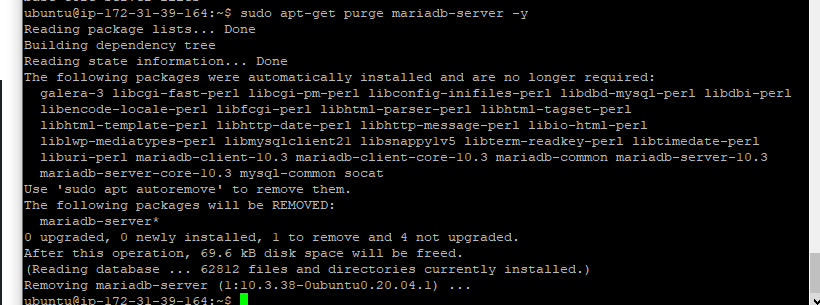
set global time\_zone='Asia/Seoul'; set time\_zone='Asia/Seoul';

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

* Purge로 삭제하는 방법



위에 하나하나 purge해주기  
purge mariadb-server 이런식으로

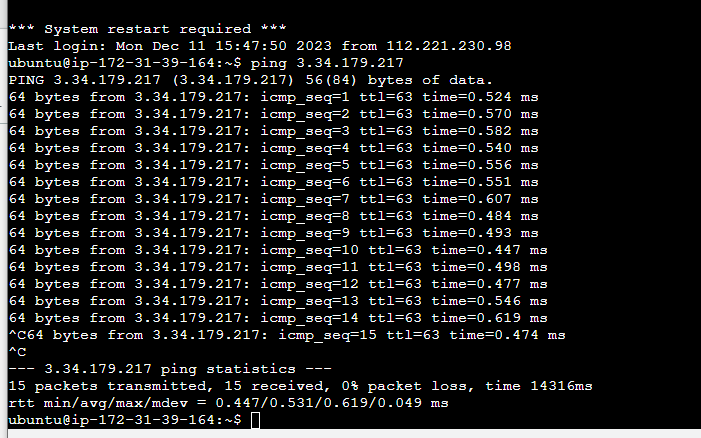


  
sudo dpkg -l | grep mariadb 하면 다 삭제된 거면 나오지 않음

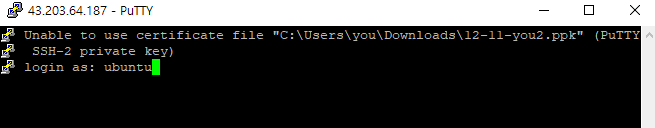
이렇게 하고 처음부터 다시해보기

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

두번째 만든 것도 putty 따로 만들어서 mariadb설치함



* **Putty 마리아 디비 설치**

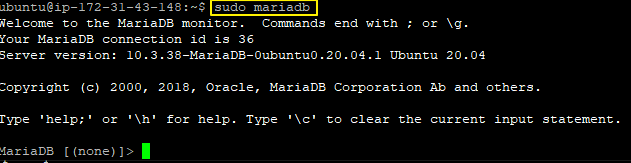


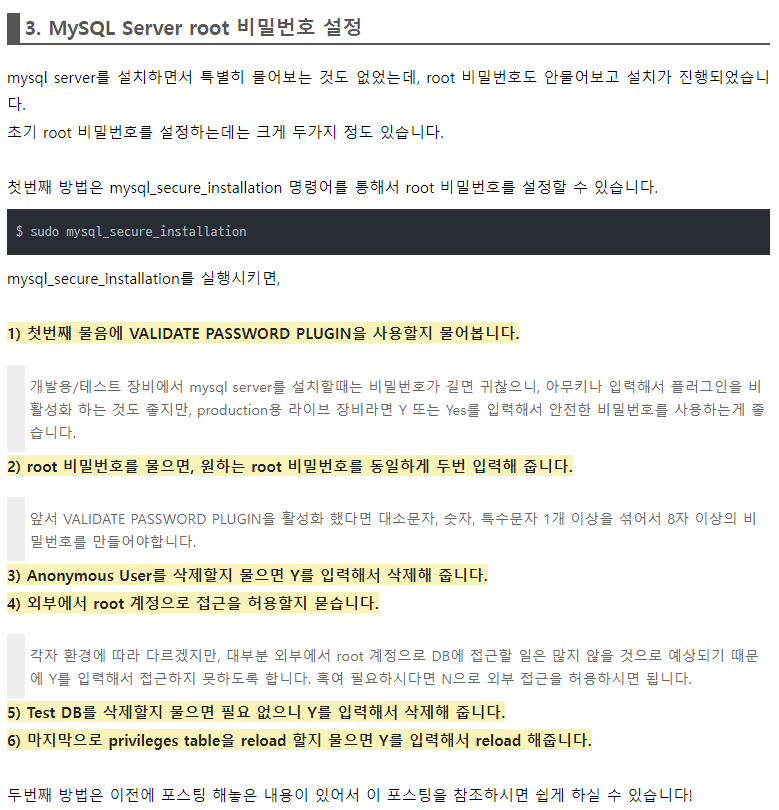


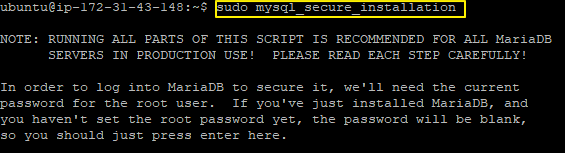
업데이트 sudo apt-get update

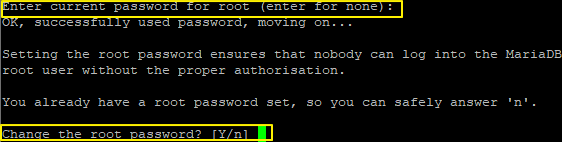
  
업그레이드 sudo apt-get upgrade

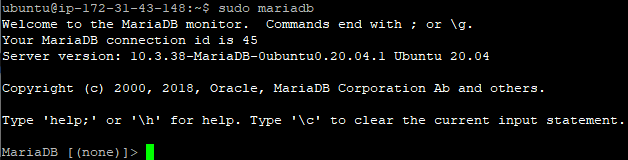
  
마리아디비 서버 putty로 설치 sudo apt-get install mariadb-server

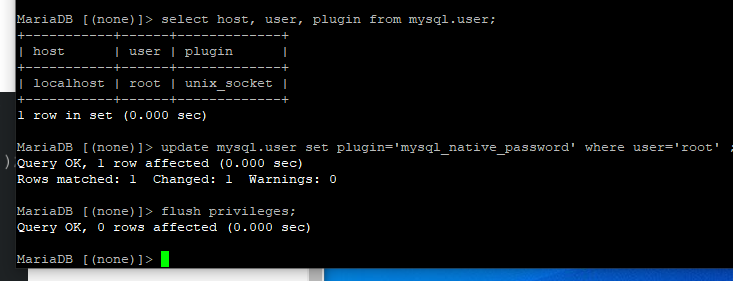
  
mysql -u root -p 를 하려고 하면 에러가 나서 그걸로 로그인 할 수 있도록 바꿔줄 것이다.



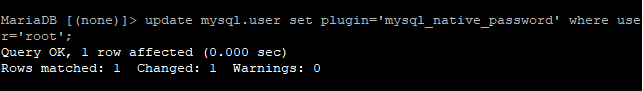
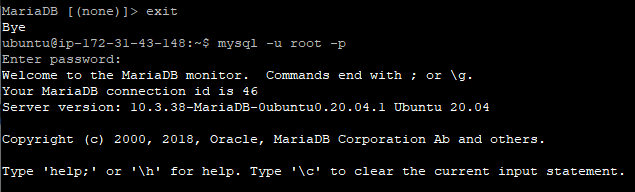
  
root비밀번호 설정하기

  
비밀번호 설정하고 y계속 누르기

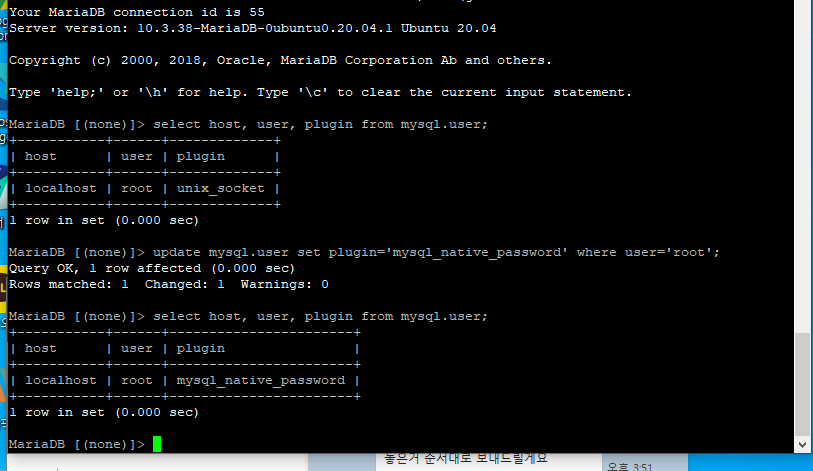
  
마리아디비로 들어와서



Plugin이 어떤 상태인지 확인함

  
바꿔주기 mysql -u root -p로 바꿔주는 명령어  
  
마리아 설치후 구문  
  
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