

## [서버 배포 가이드]

### 1. aws ec2 인스턴스를 실행한다.

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Publ
<input type="checkbox"/>	Metabusing		Running	m4.xlarge	2/2 checks passed	No alarms	ap-northeast-2a	ec2-

### 2. ssh로 연결한다.

```
login as: ubuntu
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-1023-aws x86_64)
 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Mon May 23 05:51:03 UTC 2022

System load: 0.0          Processes: 221
Usage of /: 52.4% of 9.63GB Users logged in: 0
Memory usage: 11%        IPv4 address for ens3: 172.31.6.220
Swap usage: 0%

 * Ubuntu Pro delivers the most comprehensive open source security and
 * compliance features.
 *
 * https://ubuntu.com/aws/pro

53 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Sun May 22 '9:29:58 2022 from
ubuntu@ip-172-31-6-220:~$
```

### 3. '\$ sudo apt-get install nginx'를 이용하여 nginx를 설치한다.

### 4. /etc/nginx 로 들어간 후 nginx.conf 파일을 깃 'nginx' 폴더에 있는 'nginx.conf' 파일과 동일하게 수정한다. (혹은 깃 폴더의 nginx.conf 파일을 사용한다.)

koi-win	add Nginx
mime.types	add Nginx
nginx.conf	add Nginx
nginx.conf.save	add Nginx
nginx.conf.save.1	add Nginx
nginx.conf.save.2	add Nginx

- server\_name을 통해 도메인을 지정해준다. (현재 도메인은 사용하고 있는 ec2의 고정ip로 지정되어있다.)

- 라우터를 지정해 주어 upstream을 통해 리버스 프록시가 이루어지도록 한다.

### 5. 캡스톤 깃 디렉토리에서 /APIServer로 들어간다.

### 6. npm start 명령어를 통해 api, media 서버를 실행한다. (백그라운드로 돌아가도록 설정하였다.)

```
ubuntu@ip-172-31-6-220:~$ cd /home/localWS/capstone-2022-11/APIserver/
ubuntu@ip-172-31-6-220:~/APIserver$ npm start

> apiserver@1.0.0 start /home/localWS/capstone-2022-11/APIserver
> node server8884.js & node server8885.js & node server8886.js & node server8887.js
> node server8888.js & node server8889.js & node server8890.js & node server8891.js

ubuntu@ip-172-31-6-220:~/APIserver$ 8080 PORT에 API 서버 연결합니다.
8891 PORT에 API 서버 연결합니다.
8890 PORT에 API 서버 연결합니다.
8889 PORT에 API 서버 연결합니다.
8886 PORT에 API 서버 연결합니다.
8890 PORT에 API 서버 연결합니다.
8886 PORT에 API 서버 연결합니다.
8889 PORT에 API 서버 연결합니다.
8894 PORT에 API 서버 연결합니다.
8888 PORT에 API 서버 연결합니다.
8886 PORT에 API 서버 연결합니다.
8891 PORT에 API 서버 연결합니다.
8886 PORT에 API 서버 연결합니다.
8885 PORT에 API 서버 연결합니다.
8890 PORT에 API 서버 연결합니다.
8889 PORT에 API 서버 연결합니다.
8889 PORT에 API 서버 연결합니다.
8887 PORT에 API 서버 연결합니다.
8888 PORT에 API 서버 연결합니다.
8891 PORT에 API 서버 연결합니다.
8885 PORT에 API 서버 연결합니다.
8889 PORT에 API 서버 연결합니다.
8888 PORT에 API 서버 연결합니다.
8889 PORT에 API 서버 연결합니다.
8884 PORT에 API 서버 연결합니다.
8884 PORT에 API 서버 연결합니다.
8884 PORT에 API 서버 연결합니다.
KongodB connecting Success!!
KongodB connecting Success!!
KongodB connecting Success!!
KongodB connecting Success!!
KongodB connecting Success!!
KongodB connecting Success!!
```

-> 제대로 서버가 연결 되었다면 위와 같이 지정한 포트번호와, 서버 접속이 완료되었다는 출력이 뜬다.

7. /AgoraTokenServer 로 들어간다.

8. npm start 명령어를 통해 Agora Token 서버를 실행한다. (백그라운드로 돌아가도록 설정하였다.)

```
ubuntu@ip-:/home/localWS/capstone-2022-11$ cd AgoraTokenServer/
ubuntu@ip-:/home/localWS/capstone-2022-11/AgoraTokenServer$ npm start
> agoratokenserver@1.0.0 start /home/localWS/capstone-2022-11/AgoraTokenServer
> node index.js
ubuntu@ip-:/home/localWS/capstone-2022-11/AgoraTokenServer$ Listening on port: 8082
```

-> 제대로 서버가 연결 되면 위와같이 포트에 연결되었다는 출력이 뜬다.

## [서버 종료 가이드]

1. lsof -i:[port] 명령어를 통해 백그라운드로 실행되어있는 포트들(9개)의 pid를 찾는다.

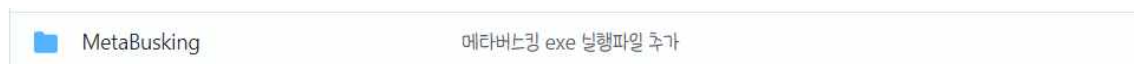
```
ubuntu@ip-:/home/localWS/capstone-2022-11$ lsof -i:8884
COMMAND  PID  USER  FD  TYPE  DEVICE  SIZE/OFF  NODE NAME
node     17711 ubuntu 22u  IPv6  203737      0t0  TCP *:8884 (LISTEN)
```

2. kill -7 [pid] 명령어를 통해 서버로 실행되는 port들의 프로세스를 지운다.

```
ubuntu@ip-:/home/localWS/capstone-2022-11$ kill -7 17711
ubuntu@ip-:/home/localWS/capstone-2022-11$ lsof -i:8884
ubuntu@ip-:/home/localWS/capstone-2022-11$
```

## [실행파일 가이드]

1. 깃 레포지토리에 'Metabusing' 폴더를 다운받는다.



2. 폴더 내부에 'Metabusing.exe'를 실행한다.

