배포가이드

**부울경 2반 6팀 웨안오딩**

배포 가이드

* **배포환경**

AWS EC2 ( Ubuntu 20.04.4 LTS )

Nginx/1.18.0 (Ubuntu)

.jar 파일의 spring boot 어플리케이션

https를 적용한 도메인/퍼블릭 주소에서 spring boot 어플리케이션 배포

openvidu 는 8447로 접근

* **Openvidu 배포**

배포를 위한 root 권한 얻기

|  |
| --- |
| $ sudo su  $ cd /opt |

오픈비두 설치

|  |
| --- |
| $ curl <https://s3-eu-west-1.amazonaws.com/aws.openvidu.io/install\_openvidu\_latest.sh> | bash  $ cd openvidu |

도메인 주소 설정 및 환경설정

|  |
| --- |
| $ nano .env  # OpenVidu configuration  # ----------------------  # Documentation: https://docs.openvidu.io/en/stable/reference-docs/openvidu-config/  # NOTE: This file doesn't need to quote assignment values, like most shells do.  # All values are stored as-is, even if they contain spaces, so don't quote them.  # Domain name. If you do not have one, the public IP of the machine.  # For example: 198.51.100.1, or openvidu.example.com  DOMAIN\_OR\_PUBLIC\_IP=i7e206.p.ssafy.io  # OpenVidu SECRET used for apps to connect to OpenVidu server and users to access to OpenVidu Dashboard  OPENVIDU\_SECRET=MY\_SECRET  # Certificate type:  # - selfsigned: Self signed certificate. Not recommended for production use.  # Users will see an ERROR when connected to web page.  # - owncert: Valid certificate purchased in a Internet services company.  # Please put the certificates files inside folder ./owncert  # with names certificate.key and certificate.cert  # - letsencrypt: Generate a new certificate using letsencrypt. Please set the  # required contact email for Let's Encrypt in LETSENCRYPT\_EMAIL  # variable.  CERTIFICATE\_TYPE=letsencrypt  # If CERTIFICATE\_TYPE=letsencrypt, you need to configure a valid email for notifications  LETSENCRYPT\_EMAIL=user@example.com  # Proxy configuration  # If you want to change the ports on which openvidu listens, uncomment the following lines  # Allows any request to http://DOMAIN\_OR\_PUBLIC\_IP:HTTP\_PORT/ to be automatically  # redirected to https://DOMAIN\_OR\_PUBLIC\_IP:HTTPS\_PORT/.  # WARNING: the default port 80 cannot be changed during the first boot  # if you have chosen to deploy with the option CERTIFICATE\_TYPE=letsencrypt  HTTP\_PORT=8448  # Changes the port of all services exposed by OpenVidu.  # SDKs, REST clients and browsers will have to connect to this port  HTTPS\_PORT=8447 |

설정 후 오픈비두 서버 실행

|  |
| --- |
| $ ./openvidu start  $ ./openvidu stop #openvidu 서버 종료  $ ./openvidu restart # openvidu 서버 재실행 |

* **Nginx 설치**

Ubuntu에 nginx 설치

|  |
| --- |
| $ sudo apt-get update #운영체제에서 사용 가능한 패키지들과 그 버전에 대한 정보(리스트) 업데이트  $ sudo apt install nginx -y #nginx 설치하기  $ nginx -v #설치한 nginx 버전 확인  $ sudo service nginx status #nginx running 상태 확인 |

* **ssl 인증서 발급**

certbot으로 https페이지를 만들기 위한 인증서를 발급받아야한다.

certbot 설치

|  |
| --- |
| $ add-apt-repository ppa:certbot/certbot  $ apt-get update  $ apt-get install python3-certbot-nginx |

설치된 certbot을 이용하여 도메인에 대한 SSL 인증서만 발급받는다.

|  |
| --- |
| $ certbot certonly --nginx -d i7e206.p.ssafy.io |

5개의 파일이 생성되었는지 확인한다.

|  |
| --- |
| $ sudo su  $ cd /etc/letsencrypt/live/i7e206.p.ssafy.io  $ ls –al  total 12  drwxr-xr-x 2 root root 4096 Aug 2 15:49 .  drwx------ 3 root root 4096 Aug 2 15:49 ..  -rw-r--r-- 1 root root 692 Aug 2 15:49 README  lrwxrwxrwx 1 root root 41 Aug 2 15:49 cert.pem -> ../../archive/i7e206.p.ssafy.io/cert1.pem  lrwxrwxrwx 1 root root 42 Aug 2 15:49 chain.pem -> ../../archive/i7e206.p.ssafy.io/chain1.pem  lrwxrwxrwx 1 root root 46 Aug 2 15:49 fullchain.pem -> ../../archive/i7e206.p.ssafy.io/fullchain1.pem  lrwxrwxrwx 1 root root 44 Aug 2 15:49 privkey.pem -> ../../archive/i7e206.p.ssafy.io/privkey1.pem |

추가로 90일마다 만료되는 인증서 자동 갱신하기

|  |
| --- |
| $ sudo certbot renew --dry-run |

* **Nginx 설정파일**

/etc/nginx/sites-available 아래 test.conf 파일 생성

|  |
| --- |
| server {  listen 80;  server\_name i7e206.p.ssafy.io;  return 301 https://i7e206.p.ssafy.io$request\_uri;  }  server {  listen 443 ssl http2;  server\_name i7e206.p.ssafy.io;  ssl\_certificate /etc/letsencrypt/live/i7e206.p.ssafy.io/fullchain.pem;  ssl\_certificate\_key /etc/letsencrypt/live/i7e206.p.ssafy.io/privkey.pem;  location / {  root /home/ubuntu/dist;  try\_files $uri $uri/ /index.html;  }  location /api {  proxy\_pass http://localhost:8080;  proxy\_set\_header Host $http\_host;  proxy\_set\_header X-Real-IP $remote\_addr;  proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;  proxy\_set\_header X-Forwarded-Proto $scheme;  }  }  server {  if ($host = i7e206.p.ssafy.io) {  return 301 https://$host$request\_uri;  }  listen 80;  server\_name i7e206.p.ssafy.io;  return 404;  } |

명령어를 통해서 문법을 확인한다.

|  |
| --- |
| $ sudo nginx -t |

test.conf 를 저장하고 빠져나온 뒤 sites-enabled에 심볼릭 링크를 만든다.

|  |
| --- |
| $ sudo ln -s /etc/nginx/sites-available/test.conf /etc/nginx/sites-enabled |

sites-enabled로 이동하여 test.conf 가 추가된 것을 확인한다.

|  |
| --- |
| $ cd /etc/nginx/sites-enabled  $ ls –l  lrwxrwxrwx 1 root root 36 Aug 2 16:00 test.conf -> /etc/nginx/sites-available/test.conf |

Nginx 재시작

|  |
| --- |
| $ sudo service nginx restart |

* **FrontEnd build**

깃 clone한 후 frontend 폴더에서

|  |
| --- |
| $ npm install  $ npm run build |

이 과정을 거치면 backend/src/main/resources/dist 폴더가 생성됨

생성된 dist 폴더를 /home/ubuntu/에 저장

* **Backend .jar 파일**

Backend 폴더에서 다음 명령어 실행

|  |
| --- |
| $ gradlew build |

Backend/build/libs 폴더에 jar파일이 생성됨

생성된 jar 파일을 /home/ubuntu/에 저장

* **서버 실행**

|  |
| --- |
| $ sudo java –jar ssafy-web-project-1.0-SNAPSHOT.jar |

* **배포시 참고사항**

|  |
| --- |
| **backend/resource/application.properties**  #DB  spring.datasource.url=jdbc:mysql://i7e206.p.ssafy.io:3306/wedding\_db?useUnicode=true&characterEncoding=utf8&serverTimezone=Asia/Seoul&zeroDateTimeBehavior=convertToNull&rewriteBatchedStatements=true  spring.datasource.hikari.username=${ datasource.username }  spring.datasource.hikari.password=${ datasource.password }  #apikey  key.imp\_key = ${imp\_key}  key.imp\_secret = ${imp\_secret} |

local에서 사용시 ${ } 안의 데이터에 바로 기입 후 사용

배포 시 jar파일을 실행할 때 아래의 명령어와 같이 데이터와 함께 넘겨줘야함

java -jar ssafy-web-project-1.0-SNAPSHOT.jar --datasource.username=username --datasource.password=password --imp\_key=example --imp\_secret=example