





# 배포

- nginx 안켜짐
  - 8080 포트 죽이기  
`sudo kill $(sudo lsof -t -i:8080)`
- 도메인과 IP 연결 문제
  - 증상: 발급받은 도메인(recourta.ga)를 입력했으나 EC2 서버로 접속이 되지 않음
  - 원인: recourta.ga와 EC2 서버의 public IP 주소를 연결해주지 않아서 생긴 문제
  - 해결: 발급받은 recourta.ga 도메인으로 접속하는 경우, 배포받은 EC2 서버의 public IP주소로 리다이렉트 되도록 도메인 발급 사이트에서 DNS 설정을 해줘야 함

 Services ▾ Partners ▾ About Freenom ▾ Support ▾  English ▾

DNS MANAGEMENT for recourta.ga

[« Back to domain details](#)

**Modify Records**

Name	Type	TTL	Target	
	A	3600	13.209.85.149	<a href="#">Delete</a>
WWW	A	3600	13.209.85.149	<a href="#">Delete</a>

[Save Changes](#)

- https 설정 시 포트번호 문제
  - 증상: https 연결이 되지 않아 openvidu 실행 시 비디오가 뜨지 않음

- 원인: ssl 포트 번호를 8443, 5443번 등으로 잘못 설정함
- 해결: 주소창에 `/`, `/recourta`, `/fe` 를 입력하면 443 포트로 접근하여 https 설정 proxy\_pass를 통해 각각 5173번 포트, 8081번 포트, 3175번 포트에 연결되게 만듦.

```
// recourta.ga 파일

// 443번으로 recourta.ga/{uri}요청이 오면 http로 넘김
server {

    // recourta.ga/ 는 5173번 포트에 연결
    location /{
        proxy_pass http://localhost:5173;
    }

    // recourta.ga/recourta는 8081번의 recourta uri로 연결
    location /recourta {
        proxy_pass http://localhost:8081/recourta;
    }

    location /fe {
        proxy_pass http://localhost:3175;
    }

    // 443번 포트(https)에 SSL 인증서 적용
    listen 443 ssl; # managed by Certbot
    ssl_certificate /etc/letsencrypt/live/recourta.ga/fullchain.pem; # managed by Certbot
    ssl_certificate_key /etc/letsencrypt/live/recourta.ga/privkey.pem; # managed by Certbot
    # include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot
    # ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot
}

// 80번 포트(http)로 recourta.ga/{uri} 요청이 오면, https(443번 포트)로 리다이렉트
server {
    if ($host = recourta.ga) {
        return 301 https://$host$request_uri;
    } # managed by Certbot

    listen 80;
    server_name recourta.ga;
    return 404; # managed by Certbot
}
```

- openvidu 서버 연결 문제

- 증상: openvidu에 커스텀 앱을 띄웠으나, openvidu 세션에 참가해도 영상이 뜨지 않음
- 원인: openvidu 서버에 요청을 보낼 때, 포트번호가 틀림
- 해결: EC2 머신의 `/opt/openvidu/.env` 파일을 수정하여 해결

```
# 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.

// 발급받았던 도메인 입력
# 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=recourta.ga
```

```

# 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

// SSL 인증서를 발급받을때 입력했던 이메일
# If CERTIFICATE_TYPE=letsencrypt, you need to configure a valid email for notifications
LETSENCRYPT_EMAIL=melon@soda@gmail.com

# Proxy configuration
# If you want to change the ports on which openvidu listens, uncomment the following lines

// http://recourta.ga:8442로 오는 요청은 전부 https://recourta.ga:4443으로 리다이렉트
# 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=5442

# Changes the port of all services exposed by OpenVidu.
# SDKs, REST clients and browsers will have to connect to this port
HTTPS_PORT=4443

```

```

// docker-compose.override.yml 파일

version: '3.1'

services:
  # -----
  #
  #   Change this if your want use your own application.
  #   It's very important expose your application in port 5442
  #   and use the http protocol.
  #
  #   Default Application
  #
  #   Openvidu-Call Version: 2.22.0
  #
  # -----
  app:
    // openvidu에 연결하고 싶은 이미지 이름을 적어준 뒤, 컨테이너로 띄우면 됨
    #   image: openvidu-vue
    #   image: openvidu-tutorials/openvidu-insecure-vue
    restart: on-failure
    network_mode: host
    environment:
      - SERVER_PORT=5442
      - OPENVIDU_URL=http://localhost:5443
      - OPENVIDU_SECRET=${OPENVIDU_SECRET}
      - CALL_OPENVIDU_CERTTYPE=${CERTIFICATE_TYPE}
    logging:
      options:
        max-size: "${DOCKER_LOGS_MAX_SIZE:-100M}"

```

- EC2 머신에서 스프링 서버 컨테이너가 실행되지 않음
  - 증상: DB Connection에 실패함
  - 원인: MySQL5InnoDB를 찾지 못해서 생긴 문제
  - 해결: application.properties 파일에서 연결되는 DB의 종류를 MySQL5Dialect로 바꿔줌
  
- 스프링 서버가 POST 요청을 받자마자 죽음
  - 증상: POST 요청을 보내자마자 net::ERR\_CONNECTION\_REFUSED 발생
  - 원인: 스프링 서버와 Mysql을 같은 네트워크에서 실행시키지 않아서 발생한 문제
  - 해결: 스프링 서버 이미지 docker run 실행시 —network="host" 옵션 추가
    - docker run —rm —network="host" backend
    - 스프링 서버 컨테이너 실행 시 host 네트워크에서 돌아가게 함으로써, MySQL과 연결할 수 있도록 해줌
    - 도커 컨테이너들은 각자 자신만의 네트워크를 가지고 있음
    - 컨테이너끼리 연결하려면 —link 옵션이 필요함 ⇒ 도커에서 더이상 권장하지 않음
    - 조금 더 나은 환경을 위해서는, 사용자 정의 브릿지 네트워크 생성 후 해당 네트워크에서 컨테이너들을 실행하는 것이 바람직함
    - 참고 사이트