



# EC2에 OpenVidu 배포하기

Tags

Server

## 목차

[EC2에 OpenVidu 배포하기](#)

[사전 준비 사항](#)

[배포하기](#)

## EC2에 OpenVidu 배포하기

### 사전 준비 사항

#### On premises - OpenVidu Docs

OpenVidu is deployed in production as a set of Docker containers managed with Docker Compose. You can deploy OpenVidu in any modern Linux distribution. This procedure installs the following services: OpenVidu Server (openvidu-server): this is the brain of OpenVidu platform. In charge of the signaling plane.

<https://docs.openvidu.io/en/2.22.0/deployment/ce/on-premises/#1-prerequisites>

#### 1. Docker 설치

#### 2. Docker Compose 설치

#### 3. Domain name

WebRTC는 HTTPS를 필요로 하는데, SSL 인증서를 발급받기 위해 필요

#### 4. Port 설정

- 포트 열어두기
  - 22 TCP: to connect using SSH to admin OpenVidu.
  - 80 TCP: if you select Let's Encrypt to generate an SSL certificate this port is used by the generation process.
  - 443 TCP: OpenVidu server and application are published by default in standard https port.
  - 3478 TCP+UDP: used by STUN/TURN server to resolve clients IPs.
  - 40000 - 57000 TCP+UDP: used by Kurento Media Server to establish media connections.
  - 57001 - 65535 TCP+UDP: used by TURN server to establish relayed media connections.

OpenVidu가 사용하는 Port들


- 포트 관련 문제로 OpenVidu 설치 전, Nginx 설치 금지

**Free ports inside the server:** OpenVidu platform services will need the following ports to be available in the machine: 80, 443, 3478, 5442, 5443, 6379 and 8888. If some of these ports is used by any process, OpenVidu platform won't work correctly. It is a typical error to have an NGINX process in the system before installing OpenVidu. Please uninstall it.

## 배포하기

### On premises - OpenVidu Docs

OpenVidu is deployed in production as a set of Docker containers managed with Docker Compose. You can deploy OpenVidu in any modern Linux distribution. This procedure installs the following services: OpenVidu Server (openvidu-server): this is the brain of OpenVidu platform. In charge of the signaling plane.

 <https://docs.openvidu.io/en/2.22.0/deployment/ce/on-premises/#deployment-instructions>

#### 1. 루트 권한 필요

```
sudo su
```

#### 2. `/opt` 에 Openvidu 설치하기

```
cd /opt
```

#### 3. OpenVidu 실행 시 필요한 파일들 다운로드 및 설치

```
curl https://s3-eu-west-1.amazonaws.com/aws.openvidu.io/install_openvidu_latest.sh | bash
```

### OpenVidu 플랫폼 설치가 완료되면,

#### 1. openvidu 폴더로 이동

```
cd openvidu
```

#### 2. `.env` 파일 설정하기

```
nano .env
```

위의 명령어를 입력하면 아래와 같은 설정 내용이 나타난다.

```

1 # OpenVidu configuration
2 # -----
3 # Documentation: https://docs.openvidu.io/en/stable/reference-docs/openvidu-config/
4
5 # NOTE: This file doesn't need to quote assignment values, like most shells do.
6 # All values are stored as-is, even if they contain spaces, so don't quote them.
7
8 # Domain name. If you do not have one, the public IP of the machine.
9 # For example: 198.51.100.1, or openvidu.example.com
10 DOMAIN_OR_PUBLIC_IP=i7a604.p.ssafy.io
11
12 # OpenVidu SECRET used for apps to connect to OpenVidu server and users to access to OpenVidu Dashboard
13 OPENVIDU_SECRET={비밀번호}
14
15 # Certificate type:
16 # - selfsigned: Self signed certificate. Not recommended for production use.
17 #               Users will see an ERROR when connected to web page.
18 # - owncert:    Valid certificate purchased in a Internet services company.
19 #               Please put the certificates files inside folder ./owncert
20 #               with names certificate.key and certificate.cert
21 # - letsencrypt: Generate a new certificate using letsencrypt. Please set the
22 #                 required contact email for Let's Encrypt in LETSENCRYPT_EMAIL
23 #                 variable.
24 CERTIFICATE_TYPE=letsencrypt
25
26 # If CERTIFICATE_TYPE=letsencrypt, you need to configure a valid email for notifications
27 LETSENCRYPT_EMAIL={이메일}

```

- `DOMAIN_OR_PUBLIC_IP` : 도메인명 입력
- `OPENVIDU_SECRET` : OpenVidu 비밀번호 설정
- `CERTIFICATE_TYPE` : letsencrypt
  - selfsigned는 배포 시, 사용자가 error들을 볼 수 있어, 적절하지 않음
- `LETSENCRYPT_EMAIL` : letsencrypt를 사용하는 경우, 이메일 입력

### 3. OpenVidu 시작하기

시작 전, Nginx 컨테이너를 포함하여 기존에 만들어두었던 모든 컨테이너들을 삭제 후 실행

∴ OpenVidu에서 80번, 443번 포트를 사용하여, 포트 충돌이 일어나기 때문

```
./openvidu start
```

위 명령어를 입력하면, OpenVidu를 실행하기 위한 여러 개 컨테이너들을 실행시킨다.

그와 동시에 OpenVidu에서 HTTPS를 사용하기 위해, SSL 인증서를 자동으로 발급 받는다.

`https://{도메인명}` 으로 접속 시, Welcome to OpenVidu 문구가 나온다면 성공한 것!

기존에 **80번, 443번** 포트로 배포한 우리 서비스를 **OpenVidu와 함께 돌아가도록** 하기 위해

외부에서 OpenVidu로 접근할 때 사용하는 포트 번호인 80번, 443번을 변경해줄 필요가 있다.

#### 1. OpenVidu 중지하기

```
./openvidu stop
```

이전에 실행시켜놓은 OpenVidu 관련 컨테이너들을 모두 Stop 시킨다.

## 2. OpenVidu 설정 변경하기

```
29 # Proxy configuration
30 # If you want to change the ports on which openvidu listens, uncomment the following lines
31
32 # Allows any request to http://DOMAIN_OR_PUBLIC_IP:HTTP_PORT/ to be automatically
33 # redirected to https://DOMAIN_OR_PUBLIC_IP:HTTPS_PORT/.
34 # WARNING: the default port 80 cannot be changed during the first boot
35 # if you have chosen to deploy with the option CERTIFICATE_TYPE=letsencrypt
36 HTTP_PORT=8081
37
38 # Changes the port of all services exposed by OpenVidu.
39 # SDKs, REST clients and browsers will have to connect to this port
40 HTTPS_PORT=8443
```

기존 80번, 443번 포트를 다른 사용하지 않는 포트 번호로 변경한다.

## 3. OpenVidu 실행하기

```
./openvidu start
```

<https://{도메인명}:8443> 으로 접속 시, 아까와 같은 문구를 확인할 수 있음

## 4. 우리 서비스를 위한 Docker Compose 실행하기

```
cd ~/common
```

우리 서비스를 위한 `docker-compose.yml` 파일이 위치한 폴더로 이동

```
docker compose build && docker compose up
```

build 후, up 시키고, <https://{도메인명}> 으로 접속 시,

우리 서비스의 index 페이지가 잘 나타남을 확인