

12 - PRÁCTICA - Servidor Unifi Controller sobre Docker

[Volver ao curso](#)

Preparación VM

Preparemos una máquina virtual Ubuntu 22.04 LTS. Pide al profesor la IP. Entra por SSH con el usuario y contraseña que te diga el profesor:

```
ssh root@10.207.0.X
```

Copiar código

Pero claro, SSH no permite inicio con root. El profesor debe preparar la máquina para que te deje entrar:

```
echo "PermitRootLogin yes" | tee -a /etc/ssh/sshd_config
```

Copiar código

Instalación de Docker

Lo primero es actualizar todo:

```
apt update && apt upgrade -y
```

Copiar código

Después instalamos Docker. Eliminamos paquetes que provocan conflicto:

```
for pkg in docker.io docker-doc docker-compose docker-  
compose-v2 podman-docker containerd runc; do sudo apt-get  
remove $pkg; done
```

[Copiar código](#)

Preparamos el repositorio de Docker y añadimos la clave GPG oficial:

```
sudo apt-get update
```

[Copiar código](#)

```
sudo apt-get install -y ca-certificates curl
```

[Copiar código](#)

```
sudo install -m 0755 -d /etc/apt/keyrings
```

[Copiar código](#)

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg  
-o /etc/apt/keyrings/docker.asc
```

[Copiar código](#)

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

[Copiar código](#)

[Copiar código](#)

Add the repository to Apt sources:

```
echo \
  "deb [arch=$(dpkg --print-architecture) signed-
  by=/etc/apt/keyrings/docker.asc]
  https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" |
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

[Copiar código](#)

```
sudo apt-get update
```

[Copiar código](#)

Instalamos Docker:

```
sudo apt-get install -y docker-ce docker-ce-cli
containerd.io docker-buildx-plugin docker-compose-plugin
```

[Copiar código](#)

Reiniciamos:

```
sudo reboot
```

[Copiar código](#)

Probamos Docker:

```
sudo docker run hello-world
```

[Copiar código](#)

Vamos a usar contenedores Docker dentro de nuestra máquina virtual. Es decir, máquinas virtuales dentro de máquinas virtuales.

Docker muy resumido

Imagen: máquina virtual sin crear (de una imagen se pueden hacer infinitas máquinas virtuales).

```
docker image ls
```

[Copiar código](#)

Se descargan con `docker pull xxxxxxxx`

Contenedor: la máquina virtual propiamente dicha.

```
docker ps
```

[Copiar código](#)

Ojo porque `docker ps` sólo muestra contenedores en ejecución

```
docker container ls --all
```

[Copiar código](#)

Arrancar/crear contenedor:

- A partir de imagen (se crea un contenedor). `docker run`
- A partir de contenedor ya creado. `docker container start`

Los contenedores fundamentalmente necesitan dos recursos:

- Carpetas o archivos del anfitrión (`--mount`)
- Abrir puertos del contenedor al anfitrión (`-d -p`)

Entrar en contenedor. No suele hacer falta ya que el contenedor expone el servicio que haga falta (Navidrome, iperf3, una web o lo que sea):

```
docker exec -it mycontainer bash
```

Copiar código

Ejemplo iperf3

Mapeo de puertos: `host:guest`

Si no especificamos protocolo, est `/tcp`

```
docker run -d --name=iperf3 -p 5201:5201 -p 5201:5201/udp -p  
5201:5201/sctp --restart unless-stopped  
ajoergensen/iperf3:latest --server
```

Copiar código

Ejemplo homer

[Docker Hub](#) | [Github](#)

Creamos una carpeta en el host:

```
mkdir ~/homer/
```

[Copiar código](#)

Añadimos permisos de lectura y escritura para Docker:

```
chmod a+rw ~/homer
```

[Copiar código](#)

En esa carpeta se almacenarán los archivos de configuración del guest (contenedor). La carpeta se poblará en la primera ejecución del contenedor.

Con `-v` mapeamos carpetas del host dentro del guest (contenedor):

Vamos a hacer que homer aparezca en el puerto 80 del host:

```
docker run -d \
  -p 80:8080 \
  -v ~/homer:/www/assets \
  --restart=always \
  b4bz/homer:latest \
  --user 0:0
```

[Copiar código](#)

Editamos la web:

```
nano ~/homer/config.yml
```

[Copiar código](#)

Ejemplo Supysonic

<https://danielrios.me/apuntes/2122spav/S04/index.html>

Unifi Controller

Imagen de Unifi Controller creada por LinuxServer.io:

<https://docs.linuxserver.io/images/docker-unifi-controller/>

Puedes ver todas las imágenes en la “tienda”:

<https://hub.docker.com/r/linuxserver/unifi>

Instalación a lo cafre

```
docker run -d \
  --name=unifi-controller \
  -e PUID=1000 \
  -e PGID=1000 \
  -e TZ=Etc/UTC \
  -e MEM_LIMIT=1024 `#optional` \
  -e MEM_STARTUP=1024 `#optional` \
  -p 8443:8443 \
  -p 3478:3478/udp \
  -p 10001:10001/udp \
  -p 8080:8080 \
  -p 1900:1900/udp `#optional` \
  -p 8843:8843 `#optional` \
  -p 8880:8880 `#optional` \
  -p 6789:6789 `#optional` \
  -p 5514:5514/udp `#optional` \
  -v /path/to/data:/config \
  --restart unless-stopped \
  lscr.io/linuxserver/unifi-controller:latest
```

Copiar código

Instalación con Docker Compose

Información: <https://docs.linuxserver.io/general/docker-compose/>

```
apt install -y docker-compose
```

Copiar código

```
mkdir unifi-controller && cd unifi-controller
```

Copiar código

```
nano docker-compose.yml
```

Copiar código

Pega esto:

```
---
version: "2.1"
services:
  unifi-controller:
    image: lscr.io/linuxserver/unifi-controller:latest
    container_name: unifi-controller
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=Etc/UTC
      - MEM_LIMIT=1024 #optional
      - MEM_STARTUP=1024 #optional
    volumes:
      - /path/to/data:/config
    ports:
```

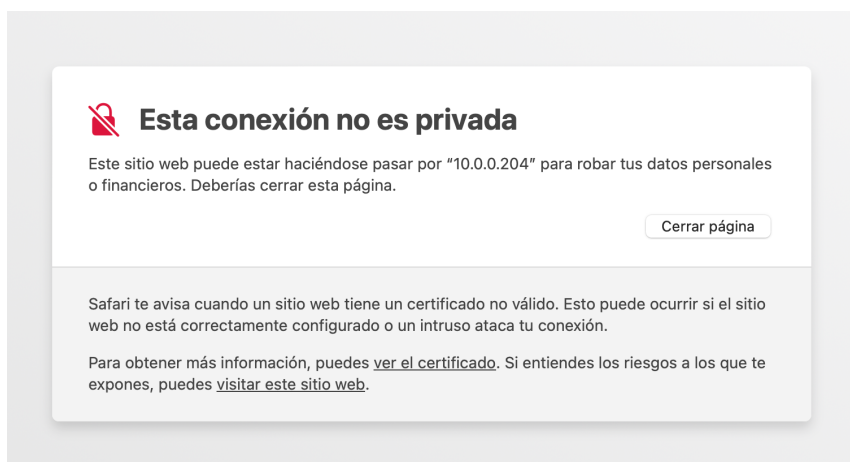
```
- 8443:8443
- 3478:3478/udp
- 10001:10001/udp
- 8080:8080
- 1900:1900/udp #optional
- 8843:8843 #optional
- 8880:8880 #optional
- 6789:6789 #optional
- 5514:5514/udp #optional
restart: unless-stopped
```

[Copiar código](#)

Guarda y cierra con CTRL + X

Explotación de Unifi Controller

Entra en: <https://10.207.0.x:8443/> y acepta el certificado autofirmado:



Dale un nombre provisional al servidor:



Name Your UniFi Network Server

Centrally manage all of your servers at unifi.ui.com.

Server Name

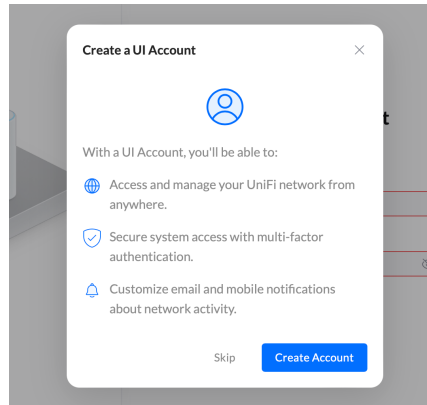
UniFi Network - Daniel Clase [Borrar](#)

Country/Region

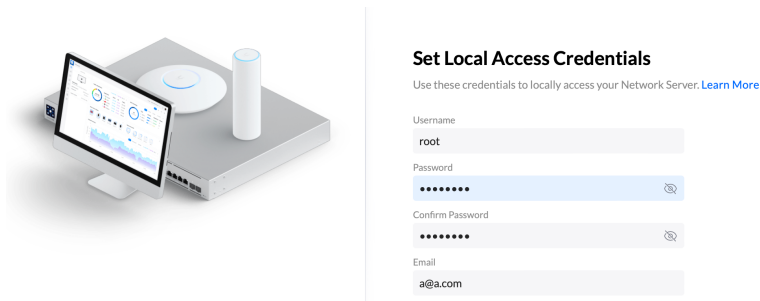
Spain

☒ I agree to [End User License Agreement](#) and [Terms of Service](#).

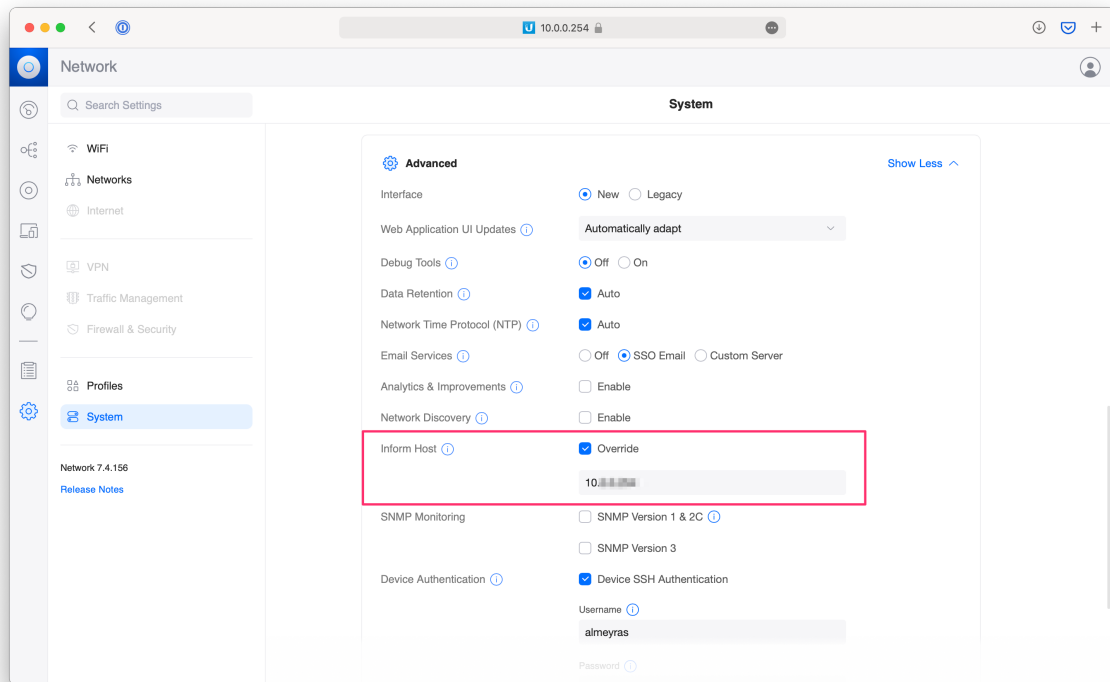
Dale a advanced y skip:



Ponle credenciales en local:



Debes resetear las antenas e irlas **adoptando**. IMPORTANTE: como es una máquina virtual (contenedor), no sabe la IP del anfitrión y no va a permitir adoptar. Debes poner la IP en Settings → System → Advanced → Inform Host:



Imágenes servidor

