

Práctica 6: Despliegue de medidas de control de acceso

Servidor de autenticación:

- VM de administración
- FreeRadius (Se puede hacer con docker)
- Configurar SSH de la forma más segura posible
- Desactivar telnet SOLO cuando se confirme que SSH funciona correctamente

Desde la máquina de salto nos conectamos a los routers y desde estos, por SSH a los Switches.

Máquina virtual para FreeRadius

Se va a crear una máquina virtual con debian para arrancar FreeRadius en un container de Docker. Primero se instala docker con los siguientes comandos:

```
# Add Docker's official GPG key:
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/debian/gpg -o
/etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
echo \
  "deb [arch=$(dpkg --print-architecture) signed-
  by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/debian \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update

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

Como alternativa podemos crear el container con un docker-compose, para ello debemos crear un documento llamado docker-compose.yml con el siguiente contenido

[docker-compose.yml](#)

```
version: '3.2'

services:
  freeradius:
    image: "2stacks/freeradius"
```

```
ports:
  - "1812:1812/udp"
  - "1813:1813/udp"
#volumes:
  #- "./configs/radius/users:/etc/raddb/users"
  #- "./configs/radius/clients.conf:/etc/raddb/clients.conf"
environment:
  #- DB_NAME=radius
  - DB_HOST=mysql
  #- DB_USER=radius
  #- DB_PASS=radpass
  #- DB_PORT=3306
  #- RADIUS_KEY=testing123
  #- RAD_CLIENTS=10.0.0.0/24
  - RAD_DEBUG=yes
depends_on:
  - mysql
links:
  - mysql
restart: always
networks:
  - backend

mysql:
  image: "mysql"
  command: --default-authentication-plugin=mysql_native_password
  ports:
    - "3306:3306"
  volumes:
    - "./configs/mysql/master/data:/var/lib/mysql"
    #- "./configs/mysql/master/conf.d:/etc/mysql/conf.d"
    - "./configs/mysql/radius.sql:/docker-entrypoint-
initdb.d/radius.sql"
  environment:
    - MYSQL_ROOT_PASSWORD=radius
    - MYSQL_USER=radius
    - MYSQL_PASSWORD=radpass
    - MYSQL_DATABASE=radius
  restart: always
  networks:
    - backend

networks:
  backend:
    ipam:
      config:
        - subnet: 10.0.0.0/24
```

Una vez instalado docker se procede a crear el container para FreeRadius con el siguiente comando:

```
docker pull freeradius/freeradius-server
docker run --rm -d --name my-radiusv2 -p 1812-1813:1812-1813/udp
freeradius/freeradius-server

docker run --name my-radius -d freeradius/freeradius-server
```

Tras eso instalamos docker compose en nuestro equipo:

```
sudo apt install docker-compose
```

Y tras eso ejecutamos el comando:

```
docker compose up -d
```

From:

<https://www.knoppia.net/> - **Knoppia**

Permanent link:

<https://www.knoppia.net/doku.php?id=redes:lab6>

Last update: **2024/10/16 16:42**

