Reto 2: MQTT Seguro

Claudia Viñals Perlado

Índice

Explicación de los pasos seguidos

A. Estudio de certificados y estructura de conexión
B. Creación del entorno sin seguridad y pruebas
C. Creación de certificados de CA, servidor y clientes
D. Pruebas de conectividad

Certificados ejecutados por script menos el de usuario

```
cvp@ClaudiaPortatil:~$ cd /home/cvp/repos/Entrega-2-iot
cvp@ClaudiaPortatil:~/repos/Entrega-2-iot$ chmod +x generate_certificates.sh
```

```
# Directorio donde se guardarán los certificados y claves del servidor y CA
CERT_DIR="/home/cvp/Entrega-2-iot/mosquitto/server_certs"

# Directorio donde se guardarán los certificados y claves de los clientes
CLIENT_CERT_DIR="/home/cvp/Entrega-2-iot/certs_clientes"

# Nombre del cliente base
CLIENT_BASE_NAME="ClaudiaPortatil"
SERVER_BASE_NAME="Entrega-2"
# Generar un certificado de autoridad (CA)
openssl req -new -x509 -days 3650 -extensions v3_ca -keyout "$CERT_DIR/ca.key" -out "$CERT_DIR/ca.crt" -subj "/CN=MyCA/OU=MyOrganization/C=US"

# Generar una clave privada y un certificado autofirmado para el servidor
openssl req -new -nodes -newkey rsa:2048 -keyout "$CERT_DIR/server.key" -out "$CERT_DIR/server.csr" -subj "/CN=$SERVER_BASE_NAME"
openssl x509 -req -days 365 -in "$CERT_DIR/server.csr" -CA "$CERT_DIR/ca.crt" -CAkey "$CERT_DIR/ca.key" -CAcreateserial -out "$CERT_DIR/server.crt"

# # Limpiar archivos temporales
rm -f "$CERT_DIR/server.csr" "$CERT_DIR/ca.srl"

echo "Certificados y claves del servidor generados con éxito en $CERT_DIR"

# Generar claves privadas y certificados autofirmados para cada cliente
for i in {1..4}; do
    CLIENT_NAME="${CLIENT_BASE_NAME}${i}"
    openssl req -new -nodes -newkey rsa:2048 -keyout "$CLIENT_CERT_DIR/$CLIENT_NAME.key" -out "$CLIENT_CERT_DIR/$CLIENT_NAME.csr" -subj "/CN=$CLIENT_BASE_NAME"
    openssl x509 -req -days 365 -in "$CLIENT_CERT_DIR/$CLIENT_NAME.csr" -CA "$CERT_DIR/ca.crt" -CAkey "$CERT_DIR/ca.key" -CAcreateserial -out "$CLIENT_CERT_DIR/$CLIENT_NAME.crt"
    # Limpiar archivos temporales
    rm -f "$CLIENT_CERT_DIR/$CLIENT_NAME.csr"
done

# Copiar los certificados de los clientes al directorio del servidor
cp "$CLIENT_CERT_DIR"/*.crt "$CERT_DIR"
cp "$CLIENT_CERT_DIR"/*.key "$CERT_DIR"
# Establecer los permisos adecuados para el servidor y los clientes
chmod 600 "$CERT_DIR"/* # Solo de leer y escribir para el usuario root
chmod 600 "$CLIENT_CERT_DIR"/* #Solo leer y escribir para root y lectura para el resto de usuarios

echo "Certificados y claves generados con éxito en $CERT_DIR y $CLIENT_CERT_DIR"
```

Cambia los permisos del archivo para hacerlo ejecutable

Instrucciones de uso

- Confirmar permisos de lectura y escritura a los certificados
- Comprobar IP de WSL y del contenedor.

```
cvp@ClaudiaPortatil:~$ docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' mosquitto
172.18.0.2
cvp@ClaudiaPortatil:~/Entrega-2-iot/app$ ip route show | grep -i default | awk '{ print $3}'
172.28.128.1
```

```
version: "3"  # Versión de Docker Compose que se está utilizando

services:  # Definición de los servicios que se ejecutarán

  mosquitto:  # Nombre del servicio, en este caso, el servidor Mosquitto
    image: eclipse-mosquitto:2.0.18  # Imagen Docker a utilizar para el
    container_name: mosquitto  # Nombre del contenedor que se creará a p
    environment:  # Variables de entorno que se pasarán al contenedor
      - TZ=Europe/Madrid  # Configuración de la zona horaria del contene
    volumes:  # Volumenes para montar directorios del host en el contene
      - /home/cvp/Entrega-2-iot/mosquitto/config:/mosquitto/config  # Mo
      - /home/cvp/Entrega-2-iot/mosquitto/server_certs:/mosquitto/certs

    ports:  # Mapeo de puertos entre el host y el contenedor
      - 8883:8883  # Puerto 8883 del host mapeado al puerto 8883 del cor
    restart: unless-stopped  # Política de reinicio del contenedor en ca
    hostname: Entrega-2 #declara el nombre del hostname
```

```
cvp@ClaudiaPortatil:~/Entrega-2-iot$ docker exec -it mosquitto sh
/ # hostname
Entrega-2
/ # hostname -i
172.18.0.2
```

Instrucciones de uso

- Ejecutar por terminal con seguridad

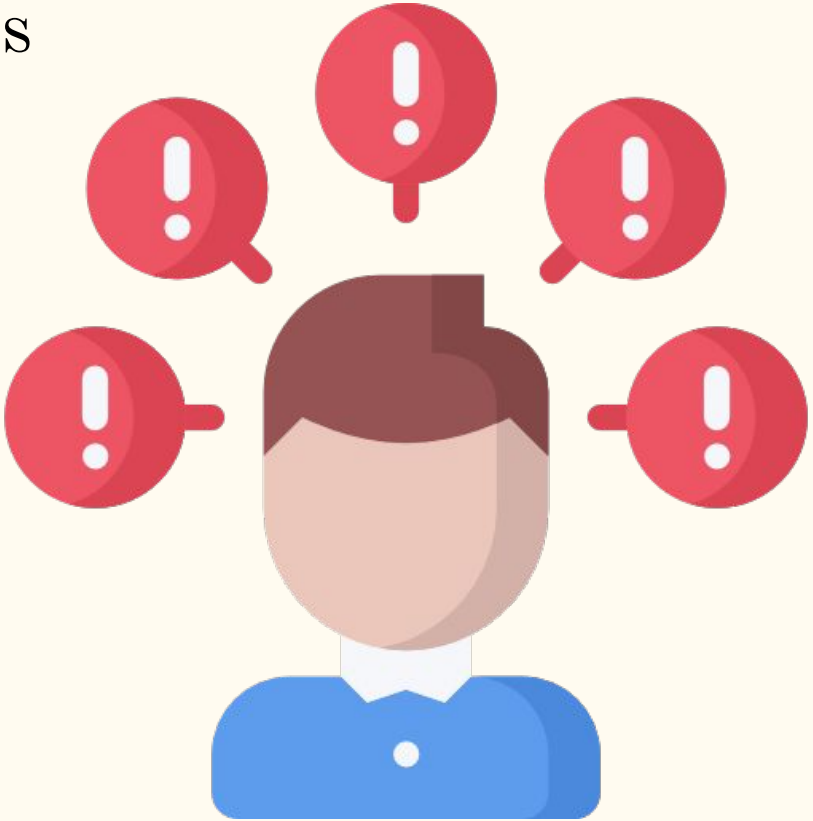
```
cvp@ClaudiaPortatil:~/Entrega-2-iot$ docker exec -it mosquitto sh
```

```
/ # mosquitto_sub -h Entrega-2 -p 8883 --cafile /mosquitto/certs/ca.crt --cert /mosquitto/certs/ClaudiaPortatil3.crt --key /m
osquitto/certs/ClaudiaPortatil3.key --username admin --pw admin -t "topic/1"
```

```
/ # echo "Hola MQTT" | mosquitto_pub -h Entrega-2 -p 8883 --cafile /mosquitto/certs/ca.crt --cert /mosquitto/certs/ClaudiaPor
tatil3.crt --key /mosquitto/certs/ClaudiaPortatil3.key --username admin --pw admin -t "topic/1" -l
```

Problemas / Retos encontrados

- Versiones y compatibilidad
- Información confusa
- Estructuración del proyecto
- Permisos para los diferentes usuarios



Posibles vías de mejora

- Creación de los certificados con SAN sin errores:

Se han hecho están comentados, pero a la hora de utilizar los ajustes estos dar errores de configuración.

- No ejecutar el broker en el contenedor:

Porque dificulta el trabajo a la hora de configurar el servidor y acceder a él.

Alternativas posibles

- Encriptación de datos de extremo a extremo
- Firewalls y políticas de seguridad de red
- OAuth

Muestra del reto