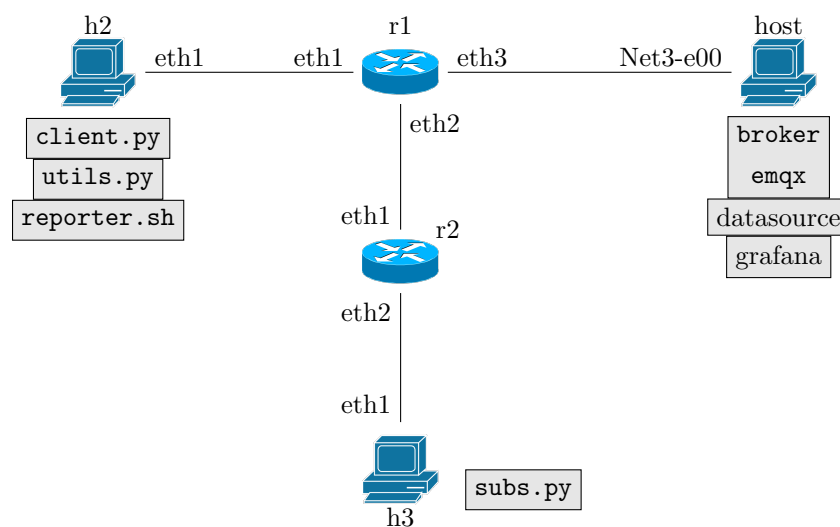




INTEGRACIÓN

Introducción

En esta práctica¹ se integrarán todas las componentes MQTT en el siguiente escenario de red:



Lanzar escenario

En primer lugar debe arrancar la máquina virtual utilizada en el curso. Sírvase del comando de sesiones anteriores. Todas las indicaciones de ahora en adelante asumen que está dentro de la máquina virtual.

Descargue en la misma carpeta los ficheros `install.sh` y `vnx-integracion.xml`. A continuación abra una terminal y ejecute

¹Este material está protegido por la licencia CC BY-NC-SA 4.0.

```
$ cd ~/Downloads
$ sudo ./install.sh
```

en caso de que haya descargado los ficheros en la carpeta `~/Downloads`. Espere unos 5 min para que termine de desplegarse todo el escenario.

Configuración de red

Plan de direccionamiento IP

En este apartado tendrá que configurar las direcciones IP de las interfaces: `h2.eth1`, `r1.eth1`, `r1.eth2`, `r2.eth1`, `r2.eth2`, `h3.eth1`. Para ello tiene que decidir qué prefijos (por ejemplo `10.1.0.0/24`) tendrán las siguientes subredes:

- Net0: compuesta por `h2.eth1` y `r1.eth1`.
- Net1: compuesta por `r1.eth2` y `r2.eth1`.
- Net2: compuesta por `r2.eth2` y `h3.eth1`.

Una vez haya definido los prefijos de cada subred, decida qué direcciones IP tendrá cada una de las interfaces que hay en cada subred. Para ello acceda a la consola del dispositivo (login: root, password: xxxx) y ejecute

```
$ ip address add IP/PREF dev IF
```

donde `IP/PREF` es la dirección a asignar a la interfaz `IF` y `PREF` es la longitud del prefijo.

Nota: la red Net3 interconecta su host (la máquina virtual) con el router virtual r1. Esta red ya viene configurada.

Reenvío IP

En este apartado debe configurar las tablas de direccionamiento de los equipos `h2`, `h3`, `r1` y `r2`. Debe configurar el direccionamiento para que tenga **ping** entre todos los dispositivos del escenario.

Para ello utilice el siguiente comando en cada una de las terminales de los equipos

```
$ ip route add SUBNET via NEXTHOP
```

Con este comando especifica que para llegar a la subred `SUBNET` (por ejemplo, `10.1.0.0/24`) tiene que enviar el tráfico al siguiente salto `NEXTHOP` (por ejemplo, `10.1.0.2`).

Nota: recuerde que el broker EMQX se ejecuta como un contenedor dentro de la máquina virtual. El contenedor suele desplegarse en la subred `172.17.0.0/16`. Por

tanto, deberá añadir rutas en todos los equipos del escenario para poder alcanzar el dicha subred.

Servicio de Detección de Anomalías

Una vez tenga configurado el escenario de red, tiene que arrancar el servicio de detección de anomalías que ha desarrollado.

En concreto, tiene que ejecutar:

- el `client.py` (sirviéndose del `utils.py` y `reporter.sh`) en el h2;
- el `subs.py` en el h3; y
- el broker EMQX, grafana, y el datasource en el host (la máquina virtual).

Nota: todos los archivos que tenga en la carpeta `/home/upm/shared` de su máquina virtual están accesibles en la carpeta `/shared` de cada componente del escenario. Se recomienda tener todos los scripts en la carpeta `/home/upm/shared` para facilitarle la edición y ejecución en cada componente.

Arrancar broker EMQX, datasource y Grafana

Lo primero que va a realizar es arrancar el contenedor del broker EMQX y Grafana (junto al plugin del datasource). Para ello, acceda al host (la máquina virtual) y ejecute los comandos ya utilizados en prácticas anteriores:

```
$ docker run -d --name emqx -p 1883:1883 -p 8083:8083 \
-p 8883:8883 -p 8084:8084 -p 18083:18083 \
emqx/emqx:5.0.2

$ sudo docker run -d -p 3000:3000 --name=grafana \
-e "GF_INSTALL_PLUGINS=grafana-mqtt-datasource" \
grafana/grafana-enterprise
```

puede comprobar que todo se ha arrancado correctamente si ve dos contenedores al ejecutar

```
$ docker ps
```

Para comprobar que h2 y h3 pueden acceder al broker, ejecute un `ping` al broker EMQX. Puede ver la dirección IP del broker EMQX usando el siguiente comando:

```
$ docker inspect $(docker ps | grep 5.0.2 \
| cut -d' ' -f1 | grep "IPAddress")
```

Arrancar el cliente

Una vez haya comprobado que hay conexión entre todos los equipos vamos a comenzar a generar métricas en h2 con el `reporter.sh`. Para ello ejecute el siguiente comando en la terminal

```
$ sudo lxc-attach h2 -- cd /shared/DIR; ./reporter.sh
```

donde `DIR` es la carpeta donde se encuentran todos sus archivos. Esto comenzará a generar el `report.csv` en la carpeta correspondiente.

Acto seguido, vamos a ejecutar nuestro `client.py`. Acceda a la terminal h2 y ejecute el cliente actualizando la dirección IP del broker. Si la configuración de red es correcta, debería ver cómo el broker va recibiendo mensajes.

Note: compruebe que los `PUBLISH` envían un JSON con las métricas de cada línea de reporte.

Visualización de métricas

En la máquina virtual, acceda a grafana y siga los pasos de prácticas anteriores para visualizar cómo va evolucionando la temperatura.

Arrancar el suscriptor/detector de anomalías

El programa `subs.py` lo vamos a ejecutar en la consola de h3. Para ello, acceda a la carpeta `/shared/DIR` y ejecute el script. Recuerde que la dirección IP del broker debe corresponder con la del contenedor corriendo en el escenario.

Si la configuración de red es correcta, debería recibir `PUBLISH` desde el broker y en pantalla deberían mostrarse las alertas de su script.

Evaluación

La evaluación de la integración se llevará a cabo por medio de un examen práctico en el que todas las personas del grupo deben demostrar que:

- saben configurar direcciones IP;
- saben configurar una tabla de direccionamiento;
- comprenden la interconectividad IP del escenario;
- conocen la mensajería MQTT intercambiada en el escenario;
- conocen las distintas QoS disponibles en MQTT;

- saben transformar los datos en Grafana para visualizar la serie temporal;
- comprenden el funcionamiento de su `client.py` y `subs.py`;
- saben analizar trazas de Wireshark correspondientes al intercambio de mensajes MQTT; y
- pueden desplegar el escenario de integración.

Atención I: se recomienda, a modo de ensayo, que cada componente sea capaz de realizar la integración por si misma.

Atención II: se recomienda, a modo de ensayo, realizar capturas en wireshark en la interfaz r1-eth3. Esta captura le mostrará todos los mensajes MQTT de su servicio.