

## RETO 2: Learning switch

### 1 Programación de un learning switch

En primer lugar, es necesario añadir las líneas de código que faltan al código del learning switch, utilizando la interfaz descrita en el apartado anterior. Las partes de código que faltan están indicadas mediante comentarios. Podéis encontrar el código en:

```
~/pox/pox/taller_sdn_ee24/reto2learningswitch/l2_learning.py
```

La máquina virtual consta del IDE para python PyCharm Community y de varios editores de texto (gedit, emacs, vim).

Antes del inicio del reto 3, se añadirá al repositorio de la práctica la solución, ya que es necesario para el siguiente reto.

### 2 Verificación del comportamiento del learning switch.

En este caso, hemos de verificar que los hosts responden a pings enviados entre ellos, cuando el controlador se está comportando como un “learning switch” de nivel 2.

Primero, cierra el controlador POX presionando Ctrl-C en la ventana en la que se está ejecutando el programa del controlador y ejecuta el ejemplo “l2\_learning”:

```
$ cd ~/pox
$ ./pox.py log.level --DEBUG taller_sdn_ee24.reto2learningswitch.l2_learning
```

NOTA: el código para la aplicación l2\_learning se proporciona en ~/pox/pox/taller\_sdn\_ee24/reto2learningswitch/l2\_learning.py.

Ahora ejecuta mininet:

```
$ sudo mn --topo single,3 --mac --switch ovsk --controller remote
```

Al igual que en el caso anterior, vamos a crear xterms para cada host y visualizar el tráfico en cada uno. En la consola de Mininet, arrancar 3 xterms:

```
mininet> xterm h1 h2 h3
```

Organiza las xterms de forma que puedan verse todas en la pantalla al mismo tiempo.

En las xterms para h2 y h3, ejecuta tcpdump, una utilidad para imprimir los paquetes “vistos” por cada host.

```
# tcpdump -XX -n -i h2-eth0
# tcpdump -XX -n -i h3-eth0
```

En la xterm para h1, ejecuta un ping:

```
# ping -c1 10.0.0.2
```

El switch examina cada paquete y aprende el mapeo entre la MAC de origen y el puerto por el que se ha recibido dicho paquete. Si el destino del paquete está ya asociado con algún puerto, el paquete se enviará únicamente a dicho puerto. Si el destino del paquete no está asociado con ningún puerto, el paquete se enviará en modo broadcast por todos los puertos del switch.

Con el comando “ovs-ofctl” podemos echar un vistazo a los flujos que el controlador POX ha instalado en el switch OpenFlow s1:

```
$ sudo ovs-ofctl dump-flows s1
```