

PRÁCTICA 2 - REDES DEFINIDAS POR SOFTWARE

Unai Sainz de la Maza Gamboa

Diseño y Proyectos de Redes

UPV/EHU

CONVENCIÓN DE IPs	2
CONVENCIÓN DE LOS PUERTOS	2
EJECUCIÓN	2

CONVENCIÓN DE IPs

Para la convención de IPs se ha seguido el patrón facilitado en el enunciado de la práctica, que sigue la siguiente especificación:

- Para las ips de los hosts seguimos la misma estructura que para sus nombres, es decir, utilizamos los mismos campos para identificarlos. En el nombre usamos la notación h_s_n , donde s es el switch (dpid) al que está conectado, y n es el host. En el caso de las ips, el valor de s es el que define la subred y n la ip asignada al host.
 - Ejemplo: la ip 10.0.1.2 es la ip asignada al host $n=2$, conectado al switch $s=1$. Pertenecer a la subred 10.0.1.0/24, donde van conectados todos los hosts que caen del switch $s=1$.
- Cada switch tiene su subred propia que cambia en base a su dpid, de la forma 10.0.s.0/24, que a su vez todas pertenecen a la subred 10.0.0.0/16.

CONVENCIÓN DE LOS PUERTOS

En el caso de los puertos, diferenciamos entre puertos asignados por los switches de acceso a los hosts conectados a ellos, y los puertos asignados por el switch de cabecera a los switches de acceso que se conecta a él.

- Switch cabecera -> Switch acceso: se asigna el puerto igual al dpid del switch de acceso. Por ejemplo, el switch 3 va a tener asignado el puerto 3, el switch 10 va a tener asignado el puerto 10, y así sucesivamente.
- Switch acceso -> Host: los switches de acceso tienen el puerto 1 reservado para conectarse con el switch cabecera. Por lo tanto, para los hosts asignamos el puerto igual a su número de host más uno. Es decir, el host $n=8$ va a tener asignado el puerto 9, el host $n=20$ va a tener asignado el puerto 21, y así sucesivamente.

EJECUCIÓN

Se han creado varios scripts de bash para facilitar la ejecución de la práctica:

- topo_exec.sh: este archivo contiene el comando adecuado para lanzar la Mininet. Lo lanza con la opción [--mac] para facilitar la legibilidad de las direcciones MAC, y con la opción [--controller remote] para poder conectarse al controlador remoto. A su vez, la opción [--topo datacenter,3] especifica la topología (definida en el

archivo `topología.py`) y el parámetro `fo`. Ejemplo: `[--topo datacenter,10]` lanza la topología datacenter con `fo=10`.

- `ryu_exec.sh`: ejecuta nuestro controlador (`esqueleto.py`) y no muestra la topología con Flowmanager (no utiliza la opción `[--observe-links]`).
- `ryu_exec_show_topo.sh`: contiene los comandos adecuados para poder visualizar la topología de manera correcta con la herramienta Flowmanager. Utiliza el controlador de ejemplo (`simple_switch_13`) facilitado por ryu.

El orden de ejecución es el siguiente:

- Primero ejecutaremos el script lanzador de la topología: `topo_exec.sh`. Esta se conectará automáticamente al controlador.
- En segundo lugar ejecutamos el script lanzador del controlador: ejecutaremos el archivo `ryu_exec.sh` para utilizar nuestro controlador, o ejecutamos `ryu_exec_show_topo.sh` si solamente queremos mostrar la topología en Flowmanager.

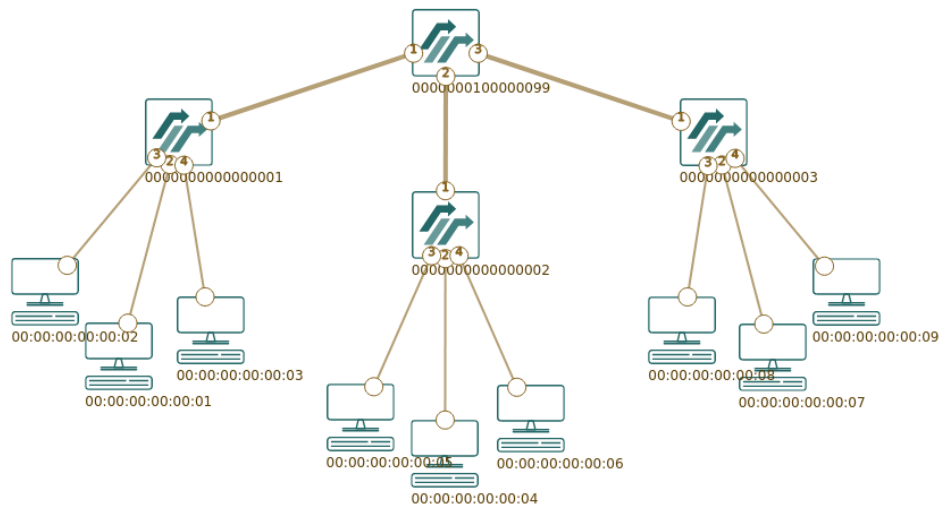
Ejemplo de uso:

- Lanzamos la topología:

`> sh topo_exec.sh`

```
mininet@mininet-vm:~/dpr/Practica_2$ sh topo_exec.sh
*** Creating network
*** Adding controller
Unable to contact the remote controller at 127.0.0.1:6653
Unable to contact the remote controller at 127.0.0.1:6653
Setting remote controller to 127.0.0.1:6653
*** Adding hosts:
h_1_1 h_1_2 h_1_3 h_2_1 h_2_2 h_2_3 h_3_1 h_3_2 h_3_3
*** Adding switches:
Ts s1 s2 s3
*** Adding links:
(Ts, s1) (Ts, s2) (Ts, s3) (s1, h_1_1) (s1, h_1_2) (s1, h_1_3) (s2, h_2_1) (s2, h_2_2) (s2, h_2_3) (s3, h_3_1) (s3, h_3_2) (s3, h_3_3)
*** Configuring hosts
h_1_1 h_1_2 h_1_3 h_2_1 h_2_2 h_2_3 h_3_1 h_3_2 h_3_3
*** Starting controller
c0
*** Starting 4 switches
Ts s1 s2 s3 ...
*** Starting CLI:
mininet>
```

Topología creada con `fo=3`:



- Lanzamos el controlador (en este caso utilizamos nuestro controlador):

> sh ryu_exec.sh

```
mininet@mininet-vm:~/dpr/Practica_2$ sh ryu_exec.sh
loading app /home/mininet/flowmanager/flowmanager.py
You are using Python v3.8.10.final.0
loading app esqueleto.py
loading app ryu.topology.switches
loading app ryu.controller.ofp_handler
creating context wsgi
instantiating app None of DPSet
creating context dpset
instantiating app /home/mininet/flowmanager/flowmanager.py of FlowManager
instantiating app esqueleto.py of TreeController
instantiating app ryu.topology.switches of Switches
instantiating app ryu.controller.ofp_handler of OFPHandler
(5314) wsgi starting up on http://0.0.0.0:8080
(5314) accepted ('127.0.0.1', 60356)
    Total number of ports of switch 3 including control: 5
Installed a flow in 3 from None to 10.0.3.1 via 2
Installed a flow in 3 from None to 10.0.3.2 via 3
Installed a flow in 3 from None to 10.0.3.3 via 4
Installed a flow in 3 from None to 10.0.1.0/24 via 1
Installed a flow in 3 from None to 10.0.2.0/24 via 1
    Total number of ports of switch 1 including control: 5
Installed a flow in 1 from None to 10.0.1.1 via 2
Installed a flow in 1 from None to 10.0.1.2 via 3
Installed a flow in 1 from None to 10.0.1.3 via 4
Installed a flow in 1 from None to 10.0.2.0/24 via 1
Installed a flow in 1 from None to 10.0.3.0/24 via 1
    Total number of ports of switch 4294967449 including control: 4
Installed a flow in 4294967449 from None to 10.0.1.0/24 via 1
Installed a flow in 4294967449 from None to 10.0.2.0/24 via 2
Installed a flow in 4294967449 from None to 10.0.3.0/24 via 3
    Total number of ports of switch 2 including control: 5
Installed a flow in 2 from None to 10.0.2.1 via 2
Installed a flow in 2 from None to 10.0.2.2 via 3
Installed a flow in 2 from None to 10.0.2.3 via 4
Installed a flow in 2 from None to 10.0.1.0/24 via 1
Installed a flow in 2 from None to 10.0.3.0/24 via 1
```