

IPv6 en INET

Práctica 2 – Diseño de Redes

Instrucciones

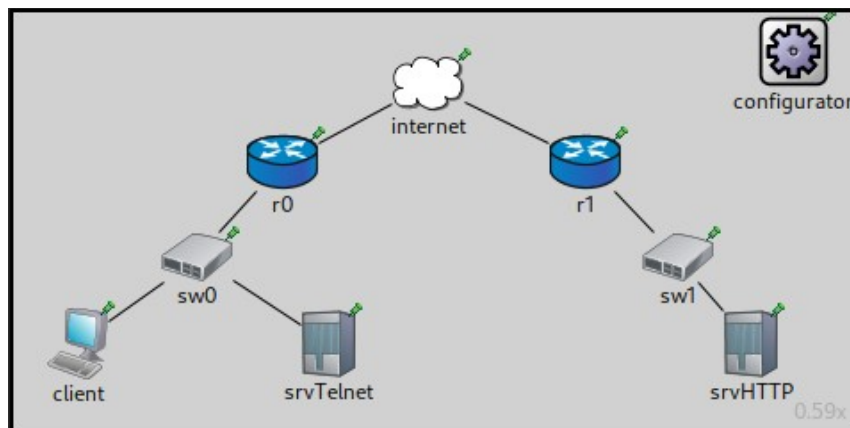
Elabore una **memoria** explicando los resultados de simulación obtenidos al seguir los pasos indicados. Incluya todas las figuras que se piden, así como cualquier figura adicional que considere importante para explicar los resultados de simulación. Las explicaciones deben hacerse a la luz de lo explicado en la teoría de la asignatura.

- El único formato aceptado es **PDF**.
- Utilice la plantilla **LaTeX Overleaf** oficial del Trabajo Fin de Grado disponible en la Wiki FIC.
- **Referencie todas las figuras** en el texto.
- El contenido de las figuras debe ser legible **sin necesidad de hacer zoom**.
- Cuide la **redacción** y la **ortografía**. Ambas serán tenidas en cuenta.
- Utilice el **formato vectorial SVG** para almacenar las figuras. Para incluirlas en la memoria, puede convertirlas a PDF con la utilidad *rsvg-convert*, incluida en el paquete *librsvg2-bin* de Ubuntu. El siguiente comando convierte a PDF todos los archivos SVG de una carpeta:

```
$ for i in *.svg; do rsvg-convert -f pdf -o ${i%.*}.pdf $i; done
```

Escenario

Diseñe una simulación¹ con un cliente que se conectará periódicamente a un servidor telnet en la red local y a un servidor HTTP en una red remota siguiendo la estructura de la imagen:



Nota: el nodo internet es un nodo de tipo `inet.node.ipv6.Router6` al que se le ha cambiado el icono; no utilice un nodo tipo `inet.node.internetcloud.InternetCloud`.

¹ Para conocer algunos de los componentes implementados por INET en IPv6 puede analizar el funcionamiento del ejemplo `examples/ipv6/nclients`. Consulte la documentación disponible en el sitio web de INET.

Configure el cliente con dos app: una de tipo `TelnetApp` y otra de tipo `TcpBasicClientApp` para simular telnet y HTTP, respectivamente:

- Configuración telnet:
 - Número de comandos (`numCommands`): distribución exponencial, media 5.
 - Longitud de comando (`commandLength`): distribución normal truncada (`truncnormal`), media 15B, desviación típica 5B.
 - Longitud de salida (`commandOutputLength`): distribución normal truncada, media 100B, desviación típica 20B.
- Configuración HTTP:
 - Tamaño de solicitud (`requestLength`): distribución normal truncada, media 300B, desviación típica 20B.
 - Tamaño de respuesta (`replyLength`): distribución normal truncada, media 20KiB, desviación típica 2KiB.

El equipo cliente debe conectarse a los servidores varias veces durante la simulación, que durará 30 minutos. Para ello defina valores de `thinkTime` e `idleInterval` apropiados. El cliente debe realizar la primera conexión al servidor HTTP en el instante $t = 10s$ y la primera conexión al servidor telnet en $t = 6s$.

Evite incluir en el fichero *omnet.ini* instrucciones innecesarias para ejecutar la simulación. Para que el simulador muestre las direcciones IP asignadas a las distintas interfaces, asegúrese de incluir en el fichero *omnet.ini* la instrucción:

```
**interfaceTable.displayAddresses = true
```

Incluya el contenido completo de los ficheros .ned y .ini en la memoria y responda a las siguientes cuestiones:

1. Durante los 2 primeros segundos se envían paquetes NS. ¿Cuál es su objetivo? Muestre una captura del **log** que muestren el motivo del envío, **filtrando** por nivel *ipv6.neighbourDiscovery* en uno de los nodos.
2. Elija uno de esos paquetes NS. Muestre una captura del **tráfico de paquetes** en la que se vean las IPs origen y destino del paquete. ¿Cómo se construye la IP destino? Explique el **motivo** de cada una de ellas.
3. ¿Qué dirección MAC destino tiene el paquete elegido? ¿Qué dirección debería tener (calcule **los 6 bytes**) según lo visto en teoría? (**Nota:** para ver direcciones MAC en la ventana de tráfico de paquetes marque “*Show all PDU destination fields*” y “*Show all PDU source fields*” en las opciones de la ventana).
4. ¿Por qué no recibe respuesta ninguno de esos paquete NS? Observe la **tabla de interfaces** del equipo cliente. Muestre el contenido de la tabla antes y después del **timeout de espera de respuesta** y explique qué cambia. (**Nota:** para ver correctamente toda la información de cada interfaz de red copie el contenido con botón derecho → *Copy Value* y péguelo en la memoria, en lugar de usar capturas de pantalla).
5. ¿Cuándo se envía el primer paquete RA? ¿Se envía como respuesta a algún paquete? Muestre capturas del **contenido del paquete RA** y explique los campos que considere más relevantes. (**Nota:** para ver las diferentes partes del paquete expanda *encapsulatedPacket* ⇒ *content* ⇒ *chunks*. El contenido relevante

en este caso estará en *Ipv6RouterAdvertisement*. Expandir todas las *options* para ver el contenido completo).

6. ¿Qué ocurre en un nodo que recibe un RA? Muestre la **tabla de interfaces** del nodo cliente justo antes y justo después de recibir el primer RA y explique los cambios (de nuevo, copie y pegue el texto de la tabla de interfaces en lugar de usar capturas).
7. En torno al instante $t = 6s$ vuelve a enviarse un paquete NS. ¿Cuál es su objetivo? Muestre capturas del **log** (de nuevo filtrando por nivel *ipv6.neighbourDiscovery*) que expliquen el motivo del envío.
8. **Explique** las IPs y MACs origen y destino de dicho NS.
9. ¿Recibe respuesta este NS? Muestre capturas del **tráfico de paquetes** y el **contenido del paquete** de respuesta.
10. Muestre capturas de la **neighbor cache** del nodo que generó el NS justo antes y justo después de recibir el paquete de respuesta y **explíquelas**.
11. ¿Se generan más NS a lo largo de la simulación? ¿Cuál es su objetivo? Muestre capturas del **log** en las que se muestre el motivo del envío.
12. Elija uno y explique IPs y MACs origen y destino. ¿Son del mismo tipo que en los casos anteriores? ¿Por qué?
13. ¿Se generan más RA a lo largo de la simulación? ¿Cuál es su objetivo? ¿Se generan como respuesta a algún paquete?