

Ampliación de Sistemas Operativos y Redes

Examen Práctico. Convocatoria de febrero de 2014

Apellidos, Nombre:

Grupo:

Normas del Examen

1. Durante el examen el alumno sólo podrá consultar el material (libro o apuntes) que haya traído y la información disponible en el sistema.
2. La solución a cada problema debe mostrarse al profesor según se completan.
3. El profesor corregirá cada problema una sola vez.
4. Es necesario anotar los comandos y el resultado de su ejecución solicitados.

Preparación del entorno del examen

Configure la topología de red que se muestra en la Figura 1. Todos los elementos - el router y las máquinas virtuales VM - deben instanciarse con la utilidad vtopol. **Nota:** es posible que sea necesario definir la máquinas base con el comando asorregenerate.

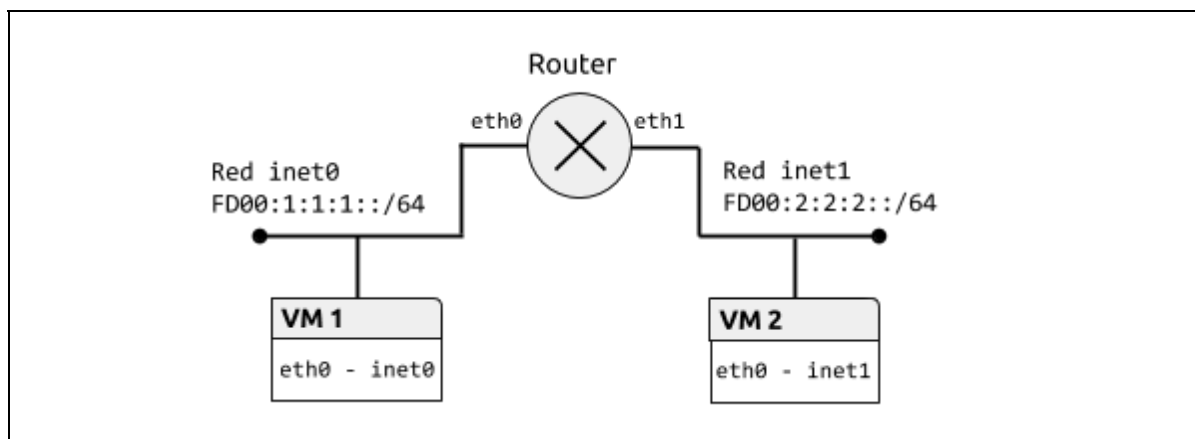


Figura 1: Topología y direccionamiento de las redes usadas en el examen

Problema 1 (0.5 pto). Configure la red inet0 de forma manual, eligiendo adecuadamente una dirección IPv6 para VM1 y Router.

Corrección:

Escriba el comando usado para verificar la configuración de la red:

Problema 2 (1 ptos). Configure la máquina virtual Router para que anuncie el prefijo de la red inet1 (fd00:2:2:2::/64).

Corrección:	Escriba la dirección de la red asignada a VM2:
--------------------	---

Problema 3 (1 ptos). Configure la máquinas de la red VM1, VM2 y Router de forma que VM1 y VM2 puedan comunicarse.

Corrección:	Escriba la tabla de rutas de VM1 (asociada a eth0) y el comando usado:
--------------------	---

Problema 4 (1 ptos). Escriba un programa en lenguaje C que funcione como un servidor stun. El servidor stun podrá recibir tres comandos, especificados en el primer carácter del mensaje enviado por el cliente:

1. 'a', Comando dirección. Responde con la dirección IP del cliente en formato cadena.
2. 'p', Comando puerto. Responde con el puerto del cliente en formato cadena.
3. 'q', Comando salir. El servidor termina su ejecución sin enviar información.

El servidor usará el protocolo UDP y atenderá las peticiones a cualquiera de las direcciones IPv6 que tenga configuradas y el puerto especificado como primer argumento del programa.. **Nota:** Probar el funcionamiento del programa con netcat.

En cada petición el servidor imprimirá un mensaje del estilo:

Host: <dirección IP cliente> Puerto: <puerto cliente>

Ejemplo de funcionamiento:

Servidor: > ./server_udp 9000 Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:54971 <hr/> Cliente: > nc -u -6 fe80::ea9d:87ff:fe8d:1122%eth0 9000 a fe80::ea9d:87ff:fe8d:1122%eth0
--

Corrección:	
--------------------	--

--	--

Problema 5 (1 ptos). Modificar el servidor UDP anterior para que cree dos hijos. Cada hijo se encargará de recibir y tratar la petición del cliente enviando la información (dirección o puerto) solicitada. Además, cuando reciba la petición mostrará una línea que indique el número de hijo (1 ó 2) y su identificador de proceso, según se muestra en la figura. El programa padre quedará suspendido durante 5 minutos.

<p>Servidor:</p> <pre>> ./server_udp 9000 [H:10512,1] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:54971 [H:10513,2] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:54971 [H:10512,1] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:54971 [H:10513,2] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:54971</pre> <hr/> <p>Cliente:</p> <pre>> nc -u -6 fe80::ea9d:87ff:fe8d:1122%eth0 9000 a fe80::ea9d:87ff:fe8d:1122%eth0a fe80::ea9d:87ff:fe8d:1122%eth0p 54971p 54971</pre>
--

Corrección:	
--------------------	--

Problema 6 (0.5 ptos). Añada la gestión adecuada de la finalización de los hijos, de forma que cuando termine alguno de ellos el proceso asociado no quede en estado zombie. El programa principal terminará cuando hayan terminado los dos hijos, e informará de la finalización de cada hijo y su código de salida..

<p>Servidor:</p> <pre>> ./server_udp 9000 [H:11062,1] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689 [H:11063,2] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689 [H:11062,1] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689 [PADRE] Terminó hijo con código de salida 0 [H:11063,2] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689 [PADRE] Terminó hijo con código de salida 0 ></pre>
--

Corrección:	
--------------------	--

--	--

Problema 7 (0.5 ptos). Modifique el comportamiento de los procesos hijo para que envíen la señal SIGUSR1 y SIGUSR2 al padre, respectivamente. El padre instalará un manejador para esas señales e imprimirá un mensaje que indique qué hijo trató la señal y el total de las peticiones tratadas por ese hijo.

Corrección:	
--------------------	--

Servidor:

```
> ./server_udp 9000
[H:11091,1] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689
[PADRE] Petición tratada por hijo 1 (total: 1)
[H:11092,2] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689
[PADRE] Petición tratada por hijo 2 (total: 1)
[H:11091,1] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689
[PADRE] Petición tratada por hijo 1 (total: 2)
[H:11092,2] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689
[PADRE] Petición tratada por hijo 2 (total: 2)
[H:11091,1] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689
[PADRE] Petición tratada por hijo 1 (total: 3)
[H:11092,2] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689
[PADRE] Petición tratada por hijo 2 (total: 3)
[H:11062,1] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689
[PADRE] Terminó hijo con código de salida 0
[H:11063,2] Host:fe80::ea9d:87ff:fe8d:1122%eth0 Puerto:46689
[PADRE] Terminó hijo con código de salida 0
>
```