

Sistemas Telemáticos para Medios Audiovisuales

Práctica 5: Calidad de Servicio en Linux

Departamento de Teoría de la Señal y Comunicaciones
y Sistemas Telemáticos y Computación
(GSyC)

Diciembre 2020

1. Control de tráfico

Antes de comenzar a realizar la práctica, por favor, descarga tus escenarios del siguiente enlace donde deberás introducir tu número de DNI (8 dígitos) con la letra correspondiente:

<http://mobiquo.gsync.es/practicass/stma/p5.html>

Descomprime el fichero que contiene el escenario de NetGUI lab-tc.tgz para realizar la práctica de control de tráfico en Linux.

1.1. Sin control de tráfico ni a la entrada ni a la salida

El *router* **r1** no tiene activado el control de tráfico en ninguna de sus interfaces.

1.1.1. Un flujo de datos

Inicia una captura en la interfaz **r1(eth1)** guardando el contenido en **tc-01.cap**.

Arranca **iperf** en modo servidor UDP en **pc3** y arranca **iperf** en modo cliente UDP en **pc1** para que envíe tráfico a 3 Mbit durante 10 segundos a **pc3**.

Observa en el lado servidor, el informe del tráfico recibido en el sentido **pc1 → pc3**.

Carga la captura en **wireshark** y muestra el flujo de forma gráfica, incluye una imagen en la memoria.

1.1.2. Dos flujos de datos

- Arranca **iperf** en modo servidor UDP en **pc4**.
- Arranca otro **iperf** en modo servidor UDP en **pc3**.
- Inicia una captura de tráfico en la interfaz **eth1** de **r1** guardando el contenido en **tc-02.cap**.
- Escribe (todavía sin ejecutar) el comando que arranca **iperf** en modo cliente UDP en **pc1** para que envíe 3 Mbit al servidor **pc3** en el sentido **pc1 → pc3** durante 10 segundos.
- Escribe (todavía sin ejecutar) el comando que arranca **iperf** en modo cliente UDP en **pc2** para que envíe 3 Mbit al servidor **pc4** en el sentido **pc2 → pc4** durante 10 segundos.

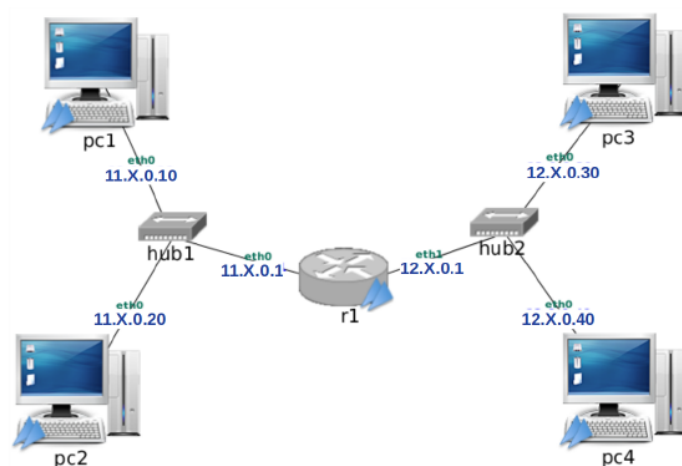


Figura 1: Escenario para control de tráfico.

- Ejecuta los dos comandos anteriores uno a continuación de otro (lo más rápidamente que puedas) para que su ejecución se realice de forma simultánea.
- Interrumpe la captura aproximadamente 10 segundos después de que arrancarás `iperf`.

A continuación analiza los resultados obtenidos:

1. Explica las estadísticas que muestran los servidores.
2. Carga la captura en `wireshark` y muestra cada uno de los flujos de forma gráfica. Explica el ancho de banda medido para cada uno de los flujos. Incluye una imagen de la gráfica en la memoria.

1.2. Control de admisión para el tráfico de entrada

Vamos a configurar `r1` para restringir el tráfico de entrada para 2 flujos de datos que recibe `r1`:

- Flujo 1: origen `pc1` se va a restringir a una velocidad de 1Mbit y una cubeta de 10k.
- Flujo 2: origen `pc2` se va a restringir a una velocidad de 2Mbit y una cubeta de 10k.
- Utiliza `tc` para definir esta configuración en la interfaz `eth0` de `r1` que es la interfaz de entrada de `r1` para los flujos 1 y 2. Ten en cuenta que se aplique primero el filtro del flujo número 1 y después el del número 2. Guarda esta configuración en un fichero de *script*, por ejemplo con el nombre `tc-ingress.sh` que deberá contener las instrucciones que ejecutarías en la línea de comandos:

```
#!/bin/sh

# Esto es un comentario

echo "Borrando la disciplina de cola ingress en la interfaz eth0"
tc qdisc del dev eth0 ingress
```

```
echo "Creando la disciplina de cola ingress en la interfaz eth0"
tc qdisc add ...
...
```

Una vez creado el *script* debes darle permisos de ejecución con la orden:

```
chmod 755 tc-ingress.sh
```

Y por último, para ejecutarlo, debes escribir¹:

```
./tc-ingress.sh
```

- Inicia una captura de tráfico en la interfaz **eth1** de **r1** y guárdala en el fichero **tc-03.cap**.
- Arranca dos clientes y 2 servidores tal y como lo hiciste en el apartado 1.1.2.
- Interrumpe la captura aproximadamente 10 segundos después de que arrancaras **iperf**, cuando los servidores hayan terminado de recibir todo el tráfico.

A continuación analiza los resultados obtenidos:

1. Explica las estadísticas que muestran los servidores.
2. Carga la captura en **wireshark** y muestra cada uno de los flujos de forma gráfica. Explica el ancho de banda medido para cada uno de los flujos. Incluye una imagen de la gráfica en la memoria y el script que has escrito.

1.3. Disciplinas de colas para el tráfico de salida

1.3.1. Token Bucket Filter (TBF)

Mantén la configuración del tráfico de entrada en **r1** que has realizado en el apartado anterior en el *script* **tc-ingress.sh**.

- Define en **r1** para su interfaz **eth1** una disciplina TBF de salida con ancho de banda 1.5 Mbit, latencia 10 ms y tamaño de cubeta 10k y guarda la configuración en un nuevo *script* **tc-egress-tbf.sh**.
- Inicia una captura de tráfico en la interfaz **eth1** de **r1** y guárdala en el fichero **tc-04.cap**.
- Arranca dos clientes y 2 servidores tal y como lo hiciste en el apartado 1.1.2.
- Interrumpe la captura aproximadamente 10 segundos después de que arrancaras **iperf**.

A continuación analiza los resultados obtenidos:

¹Si es la primera vez que ejecutas este *script*, el *router* no tendrá configurada ninguna disciplina de cola en esa interfaz y al usar la instrucción de borrado se mostrará un error, pero la ejecución del *script* continuará y se aplicarán el resto de los cambios que hayas configurado.

1. Explica las estadísticas que muestran los servidores.
2. Carga la captura en **wireshark** y muestra cada uno de los flujos de forma gráfica. Explica el ancho de banda medido para cada uno de los flujos. Incluye una imagen de la gráfica en la memoria y el script que has escrito.

Modifica la configuración de TBF de salida para que ahora tenga una latencia de 20 segundos y realiza la misma prueba que antes guardando la captura en el fichero `tc-05.cap` ². Interrumpe la captura al menos cuando hayan pasado 20 segundos desde que comenzaste a enviar tráfico desde los clientes. A continuación analiza los resultados obtenidos:

1. Explica las estadísticas que muestran los servidores.
2. Carga la captura en **wireshark** y muestra cada uno de los flujos de forma gráfica. Explica el ancho de banda medido para cada uno de los flujos. Incluye una imagen de la gráfica en la memoria y el script que has escrito.

1.3.2. Prioridad (PRIO)

Mantén la configuración del tráfico de entrada en `r1` que has realizado en el apartado anterior en el *script* `tc-ingress.sh`. Borra la disciplina de cola de salida configurada en la interfaz `eth1` de `r1`.

La configuración TBF en el apartado 1.3.1 permite gestionar el ancho de banda de salida para que no supere el valor configurado, en nuestro caso 1.5Mbit. Toma como punto de partida esta configuración para que ahora se atienda el tráfico de salida según diferentes prioridades, configurando una disciplina de cola con prioridad que sea hija de la disciplina TBF.

- Escribe un *script* en `r1`, `tc-egress-prio.sh`, para configurar TBF con los siguientes parámetros: ancho de banda 1.5Mbit, cubeta 10k y latencia 20s. Crea una disciplina de cola hija con prioridad de tal forma que se asignen las siguientes prioridades:
 - Prioridad 1 (más prioritario): tráfico de la dirección IP origen `pc1`.
 - Prioridad 2 (prioridad intermedia): tráfico de la dirección IP origen `pc2`.
 - Prioridad 3 (menos prioritario): no lo vamos a definir.
- Inicia una captura de tráfico en la interfaz `eth1` de `r1` y guarda su contenido en el fichero `tc-06.cap`.
- Arranca 2 servidores para recibir los dos flujos de datos tal y como se hizo en el apartado 1.1.2.
- Arranca dos clientes y 2 servidores tal y como lo hiciste en el apartado 1.1.2.
- Interrumpe la captura aproximadamente 35 segundos después de que arrancaras `iperf`.

A continuación analiza los resultados obtenidos:

²Ten en cuenta que ahora el tráfico quedará en la cola de la disciplina TBF esperando a ser cursado según el ancho de banda que hemos configurado. El cliente terminará de enviar a los 10 segundos y esperará a recibir el informe del servidor. Sin embargo, el servidor no acabará de recibir (y por tanto no enviará el informe) hasta que TBF no termine de atender el tráfico de la cola de salida, que será más de 10 segundos. Al no recibir el cliente el informe del servidor, terminará imprimiendo un **Warning**. De la misma forma cuando el servidor haya terminado de recibir y envíe el informe al cliente, éste ya habrá terminado su ejecución e imprimirá un mensaje indicando que no ha podido enviar el informe al cliente: **Connection refused**.

1. Explica las estadísticas que muestran los servidores.
2. Carga la captura en wireshark y muestra cada uno de los flujos de forma gráfica. Explica el ancho de banda medido para cada uno de los flujos. Incluye una imagen de la gráfica en la memoria y el script que has escrito.

1.3.3. Hierarchical token Bucket (HTB)

Mantén la configuración del tráfico de entrada en **r1** que has realizado en el apartado anterior en el *script* **tc-ingress.sh**. Borra la disciplina de cola de salida configurada en la interfaz **eth1** de **r1**.

- Escribe un *script* en **r1**, **tc-egress-htb.sh**, para configurar en su interfaz **eth1** una disciplina HTB de salida con ancho de banda 1.2 Mbit. Reparte el ancho de banda de esta interfaz de salida de la siguiente forma:
 - 700 kbit para el tráfico con origen en **pc1**, **ceil 700kbit**.
 - 500 kbit para el tráfico con origen en **pc2**, **ceil 500kbit**.
- Inicia una captura de tráfico en la interfaz **eth1** de **r1** y guárdala en el fichero **tc-07.cap**.
- Arranca 2 servidores para recibir los dos flujos de datos tal y como se hizo en el apartado 1.1.2.
- Arranca dos clientes y 2 servidores tal y como lo hiciste en el apartado 1.1.2.
- Interrumpe la captura aproximadamente 35 segundos después de que arrancaras **iperf**.

A continuación analiza los resultados obtenidos:

1. Explica las estadísticas que muestran los servidores.
2. Carga la captura en wireshark y muestra cada uno de los flujos de forma gráfica. Explica el ancho de banda medido para cada uno de los flujos. Incluye una imagen de la gráfica en la memoria y el script que has escrito.

Modifica la configuración de **ceil** en cada uno de los flujos para que puedan utilizar 1.2Mbit. Realiza la misma prueba que antes guardando el tráfico capturado en **tc-08.cap** y analiza los resultados obtenidos:

1. Explica las estadísticas que muestran los servidores.
2. Carga la captura en wireshark y muestra cada uno de los flujos de forma gráfica. Explica el ancho de banda medido para cada uno de los flujos. Incluye una imagen de la gráfica en la memoria y el script que has escrito.

Entrega de la práctica

La práctica se entregará a través del aula virtual, en un fichero comprimido **p5.zip** que debe incluir:

- La memoria
- Las capturas: desde **tc-01.cap** a **tc-08.cap**