# Examen Parcial II de Sistemas Telemáticos para Medios Audiovisuales

GSyC, Universidad Rey Juan Carlos

16 de diciembre de 2014

## HTTP

1. Las cookies almacenadas por un cliente HTTP en un momento dado son las siguientes:

```
Cookie:  Nif=123456789A
         Domain=www.server_one.com
         Path=/dir_2
         Expires=Tue Dec 31 23:12:40 2030

Cookie:  Edad=23
         Domain=www.server_two.com
         Path=/
         Expires=Tue Dec 31 23:12:40 2030

Cookie:  Nombre=Luis
         Domain=www.server_two.com
         Path=/dir_1
         Expires=Tue Dec 31 23:12:40 2030
```

   A continuación, siendo la fecha el 16-Dic-2014, dicho cliente hará la siguiente petición HTTP:

```
GET /dir_1/page_1.html HTTP/1.1
Host: www.server_one.com
```

   Indica qué cookies enviara dicho cliente en esa petición HTTP:

   **(A)** Ninguna

   **(B)** Enviaría:
   `Cookie: Nif=123456789A;`

   **(C)** Enviaría:
   `Cookie: Nombre=Luis;`

   **(D)** Enviaría:
   `Cookie: Edad=23; Nombre=Luis;`

2. Un cliente HTTP envía la siguiente petición a un servidor HTTP:

```
GET /dir_1/page_1.html HTTP/1.1
Host: www.server_one.com
Connection: Close
```

Se sabe que la página pedida contiene 4 imágenes, que también residen en el mismo servidor.

Sabiendo que cuando dicho servidor utiliza conexiones no persistentes utiliza conexiones paralelas, y que dicho servidor utiliza *pipelining* cuando dicho servidor utiliza conexiones persistentes, indica cuál de las siguientes respuestas representa mejor el tiempo aproximado de carga de dicha página completa (el fichero html y las 4 imágenes):

**(A)** 10 RTT más el tipo de transmisión de todos los recursos.

**(B)** 6 RTT más el tipo de transmisión de todos los recursos.

**(C)** 4 RTT más el tipo de transmisión de todos los recursos.

**(D)** 3 RTTs más el tipo de transmisión de todos los recursos.

3. La captura `/opt/stma/http-1.cap` muestra una petición de un formulario desde un cliente HTTP a un servidor HTTP. Indica cuál de las siguientes afirmaciones es correcta.

**(A)** Los datos de dicho formulario se enviarán al servidor mediante un mensaje que necesariamente NO incluirá la cabecera `Content-Length`.

**(B)** Los datos de dicho formulario se enviarán al servidor mediante un mensaje que necesariamente SÍ incluirá la cabecera `Content-Length`.

**(C)** De los paquetes que aparecen en la captura no se puede saber si los datos de dicho formulario se enviarán al servidor mediante un mensaje que incluya o no la cabecera `Content-Length`.

**(D)** El resto de afirmaciones son falsas.

4. Un cliente HTTP envía en el día de hoy la siguiente petición a un servidor proxy-caché HTTP:

```
GET http://www.server_one.com/dir_1/page_1.html HTTP/1.0
```

Dicha petición no incluye ninguna cabecera opcional.

Se sabe que, como consecuencia de dicha petición, el servidor proxy-caché realiza la siguiente petición al servidor final:

```
GET /dir_1/page_1.html HTTP/1.1
Host: www.server_one.com
If-modified-Since: Tue Dec 31 23:12:40 2012
Connection: close
```

Y que el servidor proxy-caché recibe la siguiente respuesta:

```
HTTP/1.1 304 Not Modified
```

Indica cuál de las siguientes afirmaciones es correcta con respecto a lo que ocurre a continuación de que el servidor proxy-caché reciba esa respuesta:

**(A)** El servidor proxy-caché envía al cliente la respuesta:

```
HTTP/1.1 304 Not Modified
```

**(B)** El servidor proxy-caché envía al cliente la respuesta:

```
HTTP/1.1 200 Ok
```

y a continuación le incluye la página pedida.

**(C)** El servidor proxy-caché envía al cliente la respuesta:

```
HTTP/1.1 500 Server error
```

**(D)** El servidor proxy-caché no envía al cliente ninguna respuesta.

## CALIDAD DE SERVICIO

**ATENCIÓN:**

- **Si ya has usado NetGUI con otro diagrama de red, cierra NetGUI y ejecuta** `clean-netgui.sh` **antes de volver a lanzar NetGUI.**
- **Arranca NetGUI, en el menú "Archivo" elige la opción "Abrir" y escribe como nombre de archivo** `/opt/stma/cs`
- **Se cargará el escenario mostrado en la figura 1.**
- **Arranca cada una de las máquinas del escenario, de una en una.**
- **Si en algún momento quieres volver a tener el escenario en su estado inicial, cierra NetGUI, ejecuta** `clean-netgui.sh` **y ejecuta después** `/opt/stma/cs/reset-lab`

---

**En el escenario no se ha configurado ninguna disciplina de colas ni de entrada ni de salida. Si realizas alguna configuración para alguna de las siguientes preguntas, recuerda borrar dicha configuración antes de pasar a otra pregunta**.

---

5. Partiendo de la situación inicial, en `r2` hay 1 *script* para la configuración de disciplina de cola de entrada: `r2-ingress.sh`. Estudia el contenido de dicho *script*. Se ejecuta dicho *script* en `r2` y se configura en la interfaz `eth2` de `r3`, una disciplina de cola TBF con los siguientes valores: rate=1Mbit, burst=10k y latencia=20s.

   Si se realiza el envío simultáneo desde `pc2` y `pc5` utilizando iperf para el envío de 1Mbps de tráfico UDP hacia `pc3` durante 10s, indica cuál de las siguientes afirmaciones sería correcta:

   **(A)** Únicamente `r2` descartará aproximadamente el siguiente tráfico durante el tiempo que dure la transmisión: 300kbps de `pc2` y 500 kbps de `pc5`.

   **(B)** Se descartará aproximadamente el siguiente tráfico durante el tiempo que dure la transmisión:
   - En `r2` se descartará 300kbps de `pc2` y 500 kbps de `pc5`.
   - En `r3` se descartará 200kbps.

   **(C)** No se descarta nada de tráfico. Todo el tráfico alcanzará `pc3` con un determinado retraso a la tasa de 1Mbps.

   **(D)** Únicamente `r3` descartará aproximadamente 1Mbps durante el tiempo que dure la transmisión.

6. Partiendo de la situación inicial del escenario se configura en `r3(eth2)` HTB con limitación de 1Mbit repartido de la siguiente forma:

   - rate=500 kbit para el tráfico de `pc1` con ceil=500kbit.
   - rate=300 kbit para el tráfico de `pc2` con ceil=1Mbit.
   - rate=200 kbit para le tráfico de `pc5` con ceil=1Mbit.

   Se inicia el envío simultáneo de tráfico UDP con `iperf` durante 10s con las siguientes características:

   - desde `pc1` dirigido a `pc3` a 800kbit
   - desde `pc2` dirigido a `pc4` a 400kbit
   - desde `pc5` dirigido a `pc4` a 100kbit.

   Indica cuál de las siguientes afirmaciones sería correcta:

   **(A)** `pc3` recibirá 800kbps durante los 10s que dura la transmisión. `pc4` recibirá 500kbit durante los 10s que dura la transmisión. Después de los 10s aproximadamente no se recibirá más tráfico en `pc3` ni en `pc4`.

   **(B)** `pc3` recibirá 800kbps durante los 10s que dura la transmisión. Después de los 10s aproximadamente no se recibirá más tráfico en `pc3`. `pc4` recibirá 400kbit durante los 10s que dura la transmisión. Después de esos 10s `pc4` seguirá recibiendo el tráfico que se había quedado encolado de `pc2`.

   **(C)** `pc3` recibirá 500kbit durante los 10s que dura la transmisión. Después de esos 10s `pc3` seguirá recibiendo el tráfico que se había quedado encolado de `pc1`. `pc4` recibirá 500kbit durante los 10s que dura la transmisión. Después de los 10s aproximadamente no se recibirá más tráfico en `pc4`.

   **(D)** `pc3` recibirá 500kbps durante los 10s que dura la transmisión. `pc4` recibirá 400kbit durante los 10s que dura la transmisión. Después de esos 10s `pc3` y `pc4` seguirán recibiendo el tráfico que se había quedado encolado de `pc1` y `pc2`.

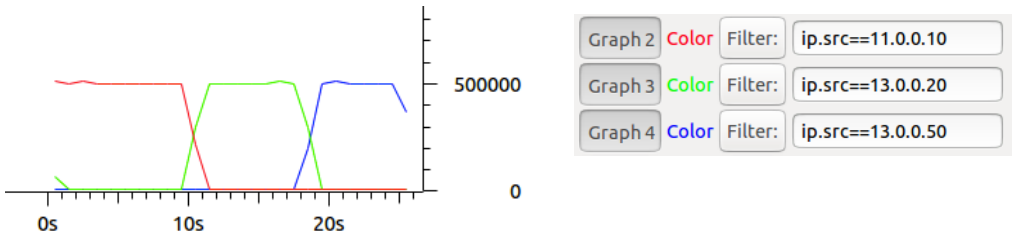
7. Partiendo de la situación inicial del escenario se configuran en `r3(eth2)` unas disciplinas de cola de salida que limiten el tráfico de salida a 1Mbit, con latencia=50s, y que den prioridad al tráfico según su dirección IP origen (de más prioridad a menos prioridad): tráfico de `pc1` (más prioritario), tráfico de `pc2` (prioridad intermedia) y tráfico de `pc5` (tráfico menos prioritario).

   Desde `pc1`, `pc2` y `pc5` se realiza el envío simultáneo de tráfico UDP utilizando `iperf` durante 10s con las siguientes características:

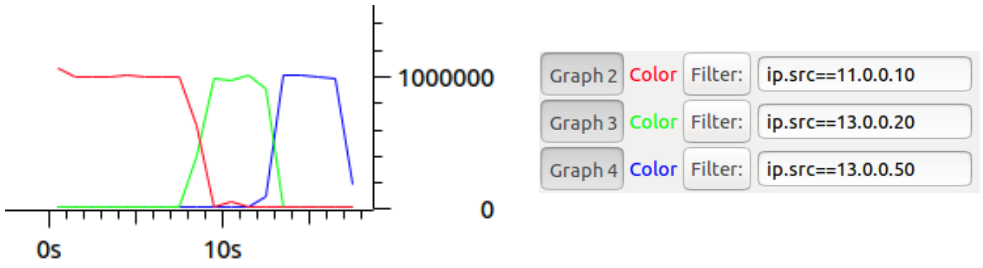
   - `pc1` envía a `pc3` a 1Mbps.
   - `pc2` envía a `pc4` a 500kbps.
   - `pc5` envía a `pc4` a 500kbps.

   Indica cuál de las siguientes gráficas de tráfico sería posible que se capturara en la interfaz `r3(eth2)` (el tráfico se muestra en bits por segundo):

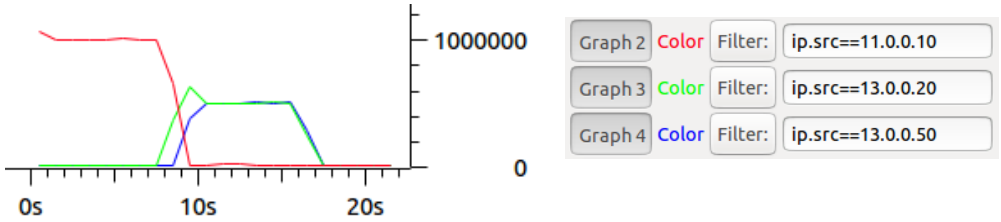
   **(A)** Gráfica 1.




   **(B)** Gráfica 2.




   **(C)** Gráfica 3.




   **(D)** Las 3 gráficas mostradas en el resto de opciones de esta pregunta pueden ser válidas para la configuración que se ha descrito en el enunciado de esta pregunta.

8. Partiendo de la situación inicial del escenario se realiza la siguiente configuración de disciplina de cola HTB en la interfaz `eth2` de `r3`:

```
tc qdisc add dev eth2 root handle 1:0 htb

tc class add dev eth2 parent 1:0 classid 1:1 htb rate 4Mbit

tc class add dev eth2 parent 1:1 classid 1:2 htb rate 3Mbit ceil 4Mbit
tc class add dev eth2 parent 1:1 classid 1:10 htb rate 1Mbit ceil 4Mbit
tc class add dev eth2 parent 1:10 classid 1:11 htb rate 600kbit ceil 4Mbit
tc class add dev eth2 parent 1:10 classid 1:12 htb rate 400kbit ceil 4Mbit

tc filter add dev eth2 parent 1:0 protocol ip prio 1 u32 match ip src 11.0.0.10 flowid 1:2
tc filter add dev eth2 parent 1:0 protocol ip prio 1 u32 match ip src 13.0.0.20 flowid 1:11
tc filter add dev eth2 parent 1:0 protocol ip prio 1 u32 match ip src 13.0.0.50 flowid 1:12
```

Se utiliza `iperf` durante 10 segundos para generar tráfico UDP simultáneamente de la siguiente forma:

- de `pc1` se envían 2 Mbps a `pc3`.
- de `pc2` se envía 1 Mbps a `pc4`.
- de `pc5` se envía 1 Mbps a `pc4`.

Indica cuál de las siguientes afirmaciones es correcta:

**(A)** `pc3` recibe 3 Mbps, `pc4` recibe 1 Mbps durante 10 segundos. Transcurridos aproximadamente los 10 segundos no se recibe más tráfico ni en `pc3` ni en `pc4`.

**(B)** `pc3` recibe 2 Mbps, `pc4` recibe 2 Mbps durante 10 segundos. Transcurridos aproximadamente los 10 segundos no se recibe más tráfico ni en `pc3` ni en `pc4`.

**(C)** `pc3` recibe 2 Mbps, `pc4` recibe 1 Mbps durante 10 segundos. Transcurridos aproximadamente los 10 segundos no se recibe más tráfico en `pc3`, sin embargo, `pc4` sigue recibiendo tráfico durante aproximadamente otros 10 segundos adicionales.

**(D)** `pc3` recibe 2 Mbps, `pc4` recibe 1 Mbps durante 10 segundos. Transcurridos los 10 segundos no se recibe más tráfico en `pc3`, sin embargo, `pc4` sigue recibiendo tráfico durante aproximadamente otros 20 segundos.

## SEGURIDAD

Se ha diseñado un sistema de comunicación que pretende que los usuarios puedan intercambiar información de manera anónima. El objetivo es dificultar que alguien que intercepte uno de los mensajes pueda conocer ni qué nodo envió originalmente el mensaje, ni cuál es el destinatario final del mismo, ni cuál es el contenido del mensaje.

Para conseguir este objetivo el mensaje se va enviando a través de una serie de nodos, elegidos por el nodo origen de la comunicación.

El nodo origen de una comunicación tiene que indicar en el mensaje que envía dos tipos de información:

- La secuencia de nodos que tiene que seguir el mensaje que envía
- El Contenido del Mensaje, que incluye la dirección del nodo que envía originalmente el mensaje, y el texto del mensaje.

Cuando un nodo recibe un mensaje, tiene que enviárselo al primero de los nodos especificados en la secuencia de nodos que viene en el mensaje, eliminando la primera entrada de la secuencia de nodos antes de enviar el mensaje.

**Ejemplo** con 5 ordenadores, $X, B, C, D, Z$, con direcciones IP $IP_X, IP_B, IP_C, IP_D, IP_Z$ respectivamente:

Supongamos que $X$ quiere enviar el texto $mensajeParaZ$ a $Z$ a través de la ruta $X \Rightarrow B \Rightarrow C \Rightarrow D \Rightarrow Z$, y que $X$ conoce $K_B^+, K_C^+, K_D^+, K_Z^+$.

1º) $X$ le envía a $B$ un datagrama IP en cuyo campo de datos va la siguiente información:
- Secuencia de nodos: $< K_B^+(IP_C) \Rightarrow K_C^+(IP_D) \Rightarrow K_D^+(IP_Z) \Rightarrow K_Z^+(IP_Z) >$
- Contenido del Mensaje: $< K_Z^+(IP_X, mensajeParaZ) >$

2º) $B$ descifra el primer componente de la secuencia de nodos recibida, y aprende que el siguiente salto es $C$. $B$ le envía entonces a $C$ un datagrama IP con la siguiente información en su campo de datos:
- Secuencia de nodos: $< K_C^+(IP_D) \Rightarrow K_D^+(IP_Z) \Rightarrow K_Z^+(IP_Z) >$
- Contenido del Mensaje: $< K_Z^+(IP_X, mensajeParaZ) >$

3º) $C$ descifra el primer componente de la secuencia de nodos recibida, y aprende que el siguiente salto es $D$. $C$ le envía a $D$:
- Secuencia de nodos: $< K_D^+(IP_Z) \Rightarrow K_Z^+(IP_Z) >$
- Contenido del Mensaje: $< K_Z^+(IP_X, mensajeParaZ) >$

4º) $D$ descifra el primer componente de la secuencia de nodos recibida, y aprende que el siguiente salto es $Z$. $D$ le envía a $Z$:
- Secuencia de nodos: $< K_Z^+(IP_Z) >$
- Contenido del Mensaje: $< K_Z^+(IP_X, mensajeParaZ) >$

5º) $Z$ descifra el primer y único componente de la secuencia de nodos recibida, y aprende que él es el nodo destinatario. Entonces $Z$ descifra el Contenido del Mensaje, sabiendo así que el mensaje lo ha enviado originalmente $IP_X$, y que el mensaje que le quería transmitir a $Z$ era $mensajeParaZ$.

9. Indica cuál de las siguientes afirmaciones es correcta:

**(A)** Cualquier nodo intermedio puede descifrar la información que aparece en el "Contenido del Mensaje", por lo que el sistema NO garantiza la confidencialidad de dicha información.

**(B)** El sistema NO garantiza la autenticación del origen de la información que aparece en el "Contenido del Mensaje", por lo que cuando $Z$ recibe un mensaje destinado para él, no puede estar seguro de qué nodo se lo ha enviado.

**(C)** Cualquier nodo intermedio, por ejemplo $C$, puede conocer cuál es la dirección IP del nodo origen del mensaje, por ejemplo, $IP_X$, y podría cambiar el "Contenido del Mensaje" sin que el destinatario final, $Z$ en el ejemplo, pueda detectarlo.

**(D)** Para que el sistema garantice tanto la confidencialidad como la autenticación del origen de la información que aparece en el "Contenido del Mensaje" habría que enviar esta información $< K_X^-(IP_X, mensajeParaZ) >$

10. Se desea que el nodo destinatario final de un mensaje pueda conocer cuál ha sido la ruta que dicho mensaje ha seguido. Indica cuál de las siguientes alternativas garantiza que el nodo final y sólo él puede conocer la ruta seguida, y que la información que llega no ha sido alterada.

    **(A)** El origen $X$ incluye esta información en el Contenido del Mensaje:
    $< K_Z^+(IP_X => IP_B => IP_C => IP_D => IP_Z, mensajeParaZ) >$

    **(B)** El origen $X$ incluye esta información en el Contenido del Mensaje:
    $< K_Z^+(K_X^-(IP_X => IP_B => IP_C => IP_D => IP_Z, mensajeParaZ)) >$

    **(C)** No es necesario realizar ningún cambio.

    **(D)** No se puede implementar dicha funcionalidad.

11. Se desea enviar de X a Z una clave de sesión que sólo el origen y destino conozcan: $K_{X-Z}$. Dicha clave de sesión la elige X y se la envía a Z. Indica cómo debería ser el contenido de dicho mensaje que le envíe X a Z para que ambos extremos puedan conocer dicha clave de forma segura:

    **(A)** El origen $X$ incluye esta información en el Contenido del Mensaje:
    $< K_Z^+(IP_X, K_{X-Z}) >$

    **(B)** El origen $X$ incluye esta información en el Contenido del Mensaje:
    $< K_Z^+(IP_X, K_X^+(K_{X-Z})) >$

    **(C)** El origen $X$ incluye esta información en el Contenido del Mensaje:
    $< K_Z^+(IP_X, K_X^-(K_{X-Z})) >$

    **(D)** No se puede implementar dicha funcionalidad.

12. Se sabe que los nodos origen y destino final comparten de forma segura una clave simétrica: $K_{X-Z}$. El origen $X$ incluye esta información en el Contenido del Mensaje:

    $< K_{X-Z}(mensajeParaZ) >$

    Indica cuál de las siguientes afirmaciones es correcta

    **(A)** Se proporciona la propiedad de confidencialidad, pero no proporciona autenticación ni integridad del Contenido del Mensaje.

    **(B)** Se proporciona la propiedad de autenticación, pero no proporciona confidencialidad ni integridad del Contenido del Mensaje.

    **(C)** Se proporciona la propiedad de integridad, pero no proporciona confidencialidad ni autenticación del Contenido del Mensaje.

    **(D)** Se proporcionan las 3 propiedades en el Contenido del Mensaje: confidencialidad, integridad y autenticación

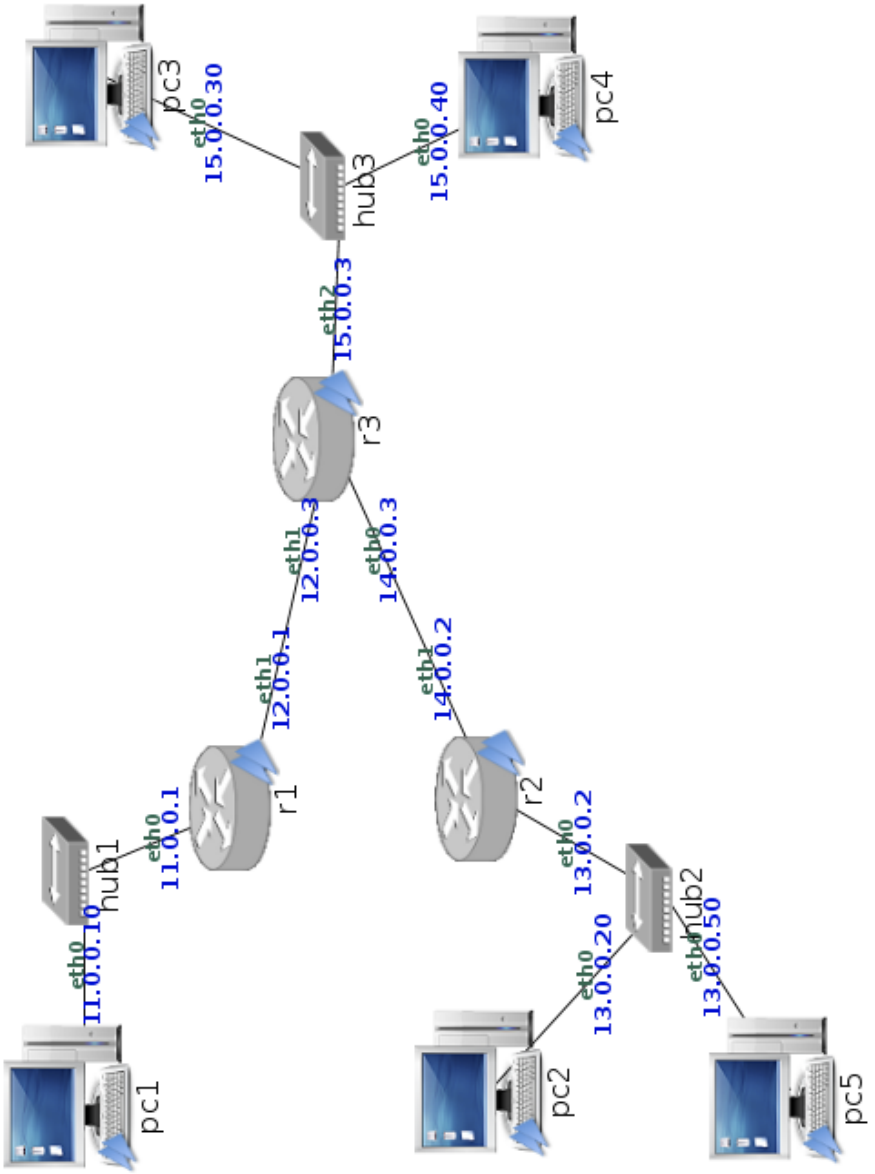



Figura 1: Calidad de servicio