Examen Parcial II de Sistemas Telemáticos para Medios Audiovisuales
HTTP, Calidad de Servicio, Seguridad

GSyC, Universidad Rey Juan Carlos

22 de junio de 2015

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 22-Jun-2015, dicho cliente hace una petición HTTP, en la que incluye únicamente la siguiente cabecera `Cookie`:

```
Cookie: Nombre=Luis
```

   Indica cuál de las siguientes opciones refleja la petición HTTP que dicho cliente ha realizado:

(A)
```
GET /dir_1/page_1.html HTTP/1.1
Host: www.server_two.com
```

(B)
```
GET /page_1.html HTTP/1.1
Host: www.server_two.com
```

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

(D) Dado ese conjunto de *cookies* almacenadas, no es posible realizar ninguna petición que lleve únicamente esa cabecera `Cookie`

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: Keep-Alive
```

Se sabe que la página pedida contiene 5 imágenes, 3 que también residen en el mismo servidor, y 2 que residen en otro servidor. El RTT de las peticiones desde el cliente a cualquiera de los dos servidores es aproximadamente el mismo.

Sabiendo que cuando dicho servidor utiliza conexiones paralelas cuando tiene que abrir nuevas conexiones, y que dicho servidor utiliza *pipelining* cuando sabe que tiene que pedir varios recursos al mismo servidor, indica cuál de las siguientes respuestas representa mejor el tiempo aproximado de carga de dicha página completa (el fichero *html* y las 5 imágenes):

(A) 5 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. Indica cuál de las siguientes afirmaciones es correcta.

(A) En HTTP 1.1, una petición nunca incluye la cabecera `Content-Type`.

(B) En HTTP 1.1, una respuesta nunca incluye la cabecera `Content-Type`.

(C) En HTTP 1.1, una petición siempre incluye la cabecera `Content-Type`.

(D) El resto de afirmaciones son falsas.

4. Se sabe que un servidor proxy-caché HTTP envía en el día de hoy la siguiente petición a un servidor final:

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

Dicha petición no incluye ninguna cabecera adicional.

Indica cuál de las siguientes afirmaciones es correcta:

(A) El servidor proxy-caché tiene en su caché ese recurso, pero ya ha caducado.

(B) El servidor proxy-caché tiene en su caché ese recurso, y aún no ha caducado.

(C) El servidor proxy-caché NO tiene en su caché ese recurso.

(D) El resto de afirmaciones son falsas.

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 `r1` hay un *script* para la configuración de disciplina de cola de entrada: `r1-ingress.sh` y en `r2` hay otro *script* para la configuración de disciplina de cola de entrada: `r2-ingress.sh`. Estudia el contenido de dichos *scripts*.

   Si se ejecutan ambos *scripts* en `r1` y `r2` y se realiza el envío simultáneo desde `pc1`, `pc2` y `pc5` utilizando `iperf` para el envío de 700kbit cada uno de tráfico UDP hacia `pc3` durante 10s, indica cuál de las siguientes afirmaciones sería correcta:

   (A) Únicamente `r2` descartará aproximadamente 200kbps de `pc5` durante el tiempo que dure la transmisión.

   (B) Ningún router descartará nada de tráfico ya que los 200kbps que `r2` podría descartar de `pc5` se compensan con los 200kbps que no está ocupando `pc2`.

   (C) `r1` descartará aproximadamente 100kbps de `pc1` y `r2` descartará aproximadamente 200kbps de `pc5` durante el tiempo que dure la transmisión.

   (D) `r1` descartará aproximadamente100kbps de `pc1` y `r2` no descartará nada de tráfico ya que los 200kbps que `r2` podría descartar de `pc5` se compensan con los 200kbps que no está ocupando `pc2`.

6. Partiendo de la situación inicial del escenario se configura en `r3(eth2)` HTB con limitación de 1.3Mbit repartido de la siguiente forma:

   - rate=200 kbit para el tráfico de `pc1` con ceil=300kbit.
   - rate=500 kbit para el tráfico de `pc2` con ceil=500kbit.
   - rate=600 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 400kbit
   - desde `pc2` dirigido a `pc4` a 300kbit
   - desde `pc5` dirigido a `pc4` a 600kbit.

   Indica cuál de las siguientes afirmaciones sería correcta, después de los 10s que dura el envío de tráfico desde cada fuente:

   (A) `r3` seguirá reenviando tráfico únicamente de `pc5` que se ha quedado encolado.

   (B) `r3` seguirá reenviando tráfico únicamente de `pc1` que se ha quedado encolado.

   (C) `r3` seguirá reenviando tráfico únicamente de `pc1` y `pc5` que se ha quedado encolado.

   (D) `r3` no seguirá reenviando tráfico de ninguna de las fuentes porque no se quedado encolado ya que HTB tiene una limitación de 1.3Mbit.

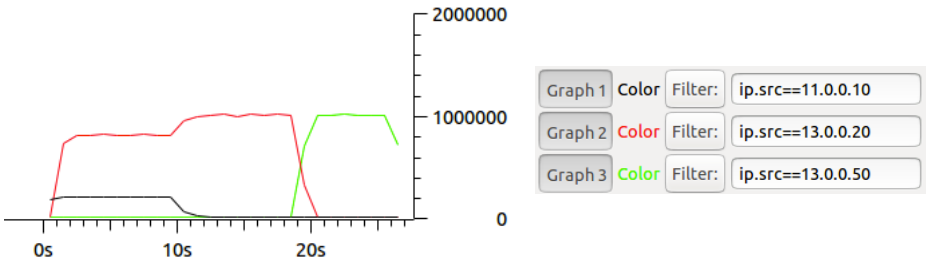
7. Partiendo de la situación inicial del escenario se configuran en `r3(eth2)` unas disciplinas de cola de salida que limiten el tráfico de salida a 1Mbps, con latencia=50s, y que den prioridad al tráfico según su dirección IP origen (de más prioridad a menos prioridad): tráfico de `pc1` (más prioritario),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

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

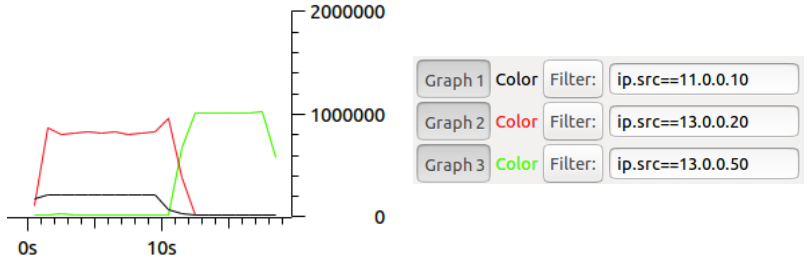
   - `pc1` envía a `pc3` a 200kbps.
   - `pc2` envía a `pc4` a 1600kbps.
   - `pc5` envía a `pc4` a 700kbps.

   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):

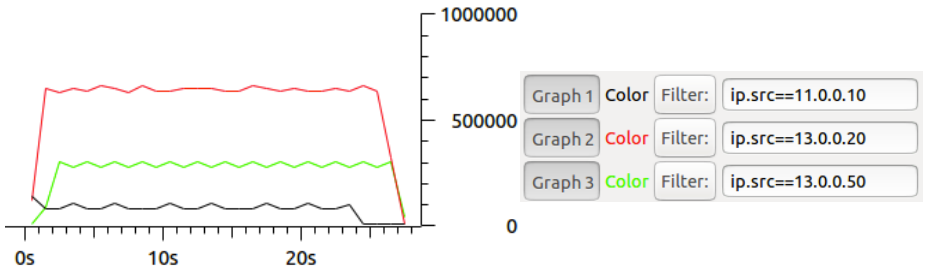
(A) Gráfica 1.



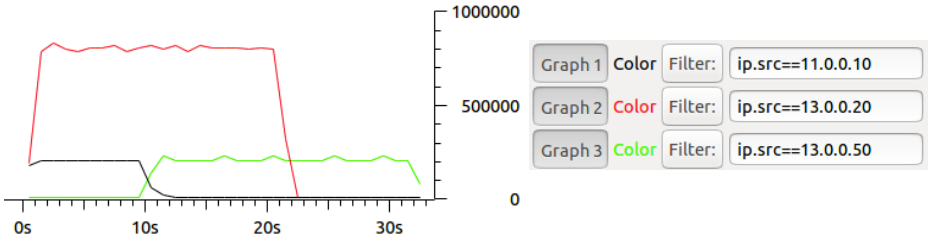
(B) Gráfica 2.



(C) Gráfica 3.



(D) Gráfica 4.



8. Partiendo de la situación inicial del escenario se realiza la siguiente configuración de disciplina de cola TBF en la interfaz `eth2` de `r3` con los siguientes parámetros rate=1Mbit burst=10k latency=60s.

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

   - `pc1` envía a `pc3` a 1Mbps.
   - `pc5` envía a `pc4` a 1Mbps.

   Indica cuál de las siguientes afirmaciones es correcta:

   (A) La tasa de salida de tráfico de `r3` es 1Mbps. Se descarta parte del tráfico de `pc1` y `pc5`.

   (B) La tasa de salida de tráfico de `r3` es 2Mbps. Se descarta parte del tráfico de `pc1` y `pc5`.

   (C) La tasa de salida de tráfico de `r3` es 2Mbps. No se descarta nada de tráfico.

   (D) La tasa de salida de tráfico de `r3` es 1Mbps. No se descarta nada de tráfico.

SEGURIDAD

Un conjunto de amigos $\{A, B, C, D, E\}$ utilizan criptografía de clave pública/privada para intercambiar mensajes. Cada uno de los amigos ha conseguido de forma segura las claves públicas del resto de los amigos. Entre ellos se han puesto de acuerdo para utilizar la función *hash* H.

9. Cada vez que quiere entrar un amigo Z nuevo en el grupo, tienen que comunicarle de forma segura las claves públicas de todos los miembros del grupo K_i^+ donde $i = \{A, B, C, D, E\}$ y los miembros del grupo deben obtener de forma segura la clave pública de Z, K_Z^+. Para ello, Z quedará personalmente con uno de los miembros del grupo, i, y se intercambiarán las claves que necesitan. De esta forma Z ya poseerá las claves de todos los miembros del grupo y sólo quedará que i le comunique de forma segura al resto de miembros del grupo la clave K_Z^+. Se supone que todos los miembros confían en i y que éste no va a enviarles una K_Z^+ falsa.

   Indica cuál es el mejor mecanismo para que el miembro i que posee la clave K_Z^+ se la pueda proporcionar a otro miembro del grupo, j, si le envía un mensaje con el siguiente contenido:

   (A) $mensaje = K_j^+(nombre = Z, clave = K_Z^+)$

   (B) $mensaje = K_i^-(nombre = Z, clave = K_Z^+)$

   (C) $mensaje = \{nombre = Z, clave = K_Z^+\}$
   $firma = K_i^-(H(mensaje))$

   (D) $mensaje = K_i^+(nombre = Z, clave = K_Z^+)$
   $firma = K_i^-(H(mensaje))$

10. Z, que es un nuevo miembro del grupo, le ha pasado a i de forma segura su clave pública K_Z^+ e i le ha dado a Z todas las claves públicas de todos los miembros. Sin embargo otro miembro j, con el que se quiere comunicar Z, aún no tiene K_Z^+. Por ello, Z le pide a i que le ayude a enviarle un mensaje $m1$ a j. Z le enviará un mensaje a i y éste se lo reenviará a j para que j pueda pueda leer $m1$. Z quiere asegurarse de que:

    - i no cotillee el contenido de su mensaje $m1$.
    - j pueda garantizar que en el camino de $Z \rightarrow i$ o en el camino de $i \rightarrow j$ nadie haya alterado el mensaje $m1$. Z y j confían en que i no va alterar $m1$.

    Indica qué mensaje debería enviar Z a i para que éste se lo reenvíe a j:

    (A)

Z → i:	$mensaje = K_j^+(m1)$ $firma = K_Z^-(H(mensaje))$
i recalcula una nueva firma ya que dispone de K_Z^+	
i → j:	$mensaje = K_j^+(m1)$ $firma' = K_i^-(H(mensaje))$

    (B)

Z → i:	$mensaje = K_Z^-(m1)$ $firma = K_Z^-(H(mensaje))$
i recalcula el contenido del mensaje y una nueva firma ya que dispone de K_Z^+	
i → j:	$mensaje' = K_i^-(m1)$ $firma' = K_i^-(H(mensaje'))$

    (C)

Z → i:	$mensaje = K_Z^-(K_j^+(m1))$
i recalcula el contenido del mensaje ya que dispone de K_Z^+	
i → j:	$mensaje' = K_j^+(m1)$

    (D)

Z → i:	$mensaje = K_j^+(m1)$
i añade una firma al mensaje	
i → j:	$mensaje = K_j^+(m1)$ $firma = K_i^-(H(mensaje))$

11. A y B han decidido utilizar clave simétrica para intercambiar mensajes. A elegirá una clave simétrica K_{A-B} y se la enviará a B.

    Indica cuál de los siguientes procedimientos le permite conocer a B la clave simétrica y estar seguro de que A es el que se la ha enviado, para posteriormente utilizarla en el intercambio de mensajes de forma segura:

    (A) A $\rightarrow$ B: $mensaje = K_B^+(K_{A-B})$

    (B) A $\rightarrow$ B: $mensaje = K_A^-(K_{A-B})$

    (C) A $\rightarrow$ B: $mensaje = \{K_{A-B}, K_A^-(H(K_{A-B}))\}$

    (D) A $\rightarrow$ B: $mensaje = \{K_B^+(K_{A-B})), K_A^-(H(K_{A-B}))\}$

12. Uno de los amigos, A, recibe el siguiente mensaje:

    $$mensaje = \{H(m1), K_B^-(H(m1))\}$$

    Indica cuál de las siguientes afirmaciones es correcta:

    (A) A puede conocer el contenido del mensaje m1 pero no puede saber quién lo ha generado.

    (B) A puede conocer el contenido del mensaje m1 y está seguro de que lo ha generado B.

    (C) A no puede conocer el contenido del mensaje m1, pero está seguro de que lo ha generado B.

    (D) A no puede conocer el contenido del mensaje m1, ni tampoco saber quién lo ha generado.

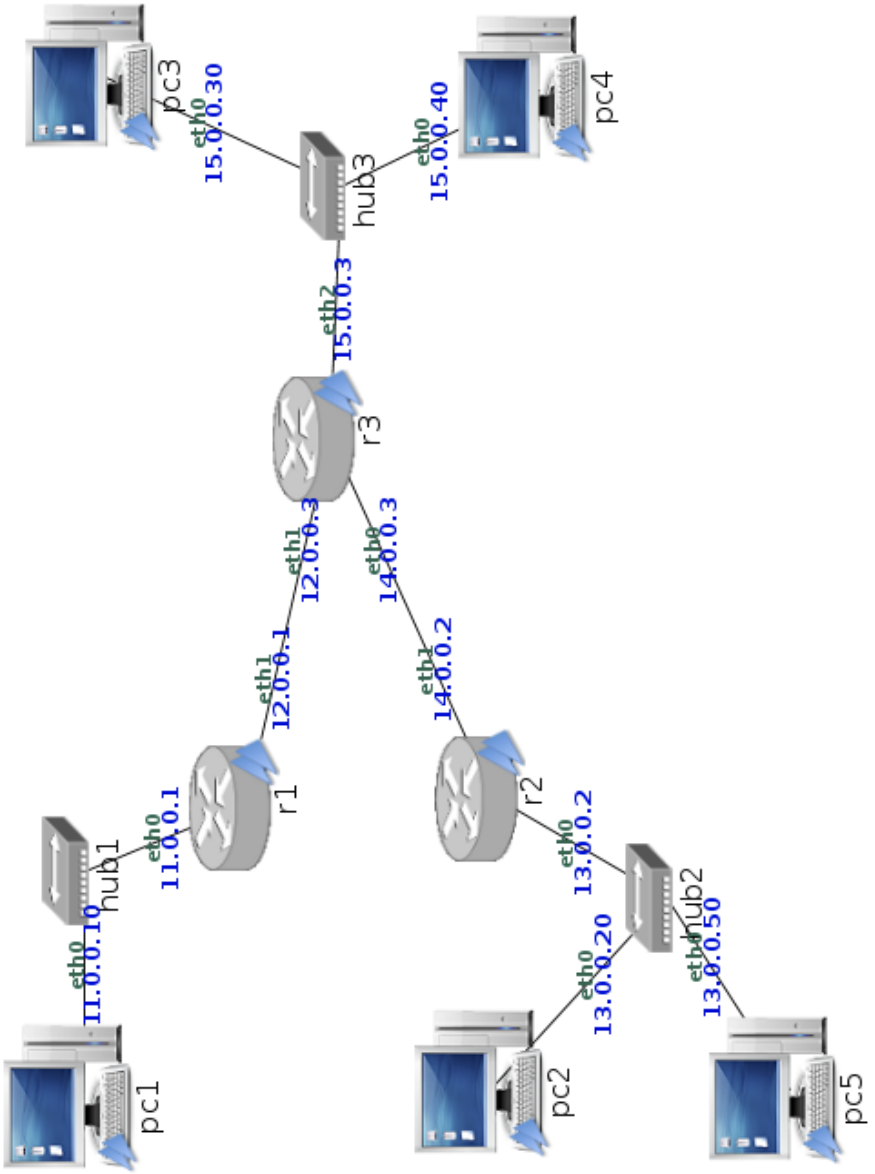


Figura 1: Calidad de servicio