Examen Final de Sistemas Telemáticos para Medios Audiovisuales
Parcial II: Calidad de Servicio, HTTP, Claves, IPtables
Grado de Ingeniería en Sistemas Audiovisuales y Multimedia

GSyC, Universidad Rey Juan Carlos

15 de junio de 2017

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.

---

1. Partiendo de la situación inicial, en `r1` y `r2` hay 2 scripts para la configuración de disciplina de cola de entrada: `r1-ingress.sh` y `r2-ingress.sh`, respectivamente. Estudia el contenido de dichos *scripts*. Si se ejecutan dichos *scripts* y se realiza el envío simultáneo utilizando `iperf` en `pc1`, `pc2` y `pc5` de 1Mbit de tráfico UDP desde cada una de estas máquinas a `pc3`, indica cuánto tráfico recibiría `pc3`:

   (A) 3 Mbits aproximadamente durante 10 segundos, 1 Mbit de `pc1`, 1Mbit de `pc2` y 1Mbit de `pc5`.
   (B) 2.5 Mbits aproximadamente durante 10 segundos, 1 Mbit de `pc1`, 500kbit de `pc2` y 1Mbit de `pc5`.
   (C) 2 Mbits aproximadamente durante 10 segundos, 1 Mbit de `pc1` y 1Mbit de `pc2`.
   (D) 1.5 Mbits aproximadamente durante 10 segundos, 1 Mbit de `pc1` y 500kbit de `pc2`.

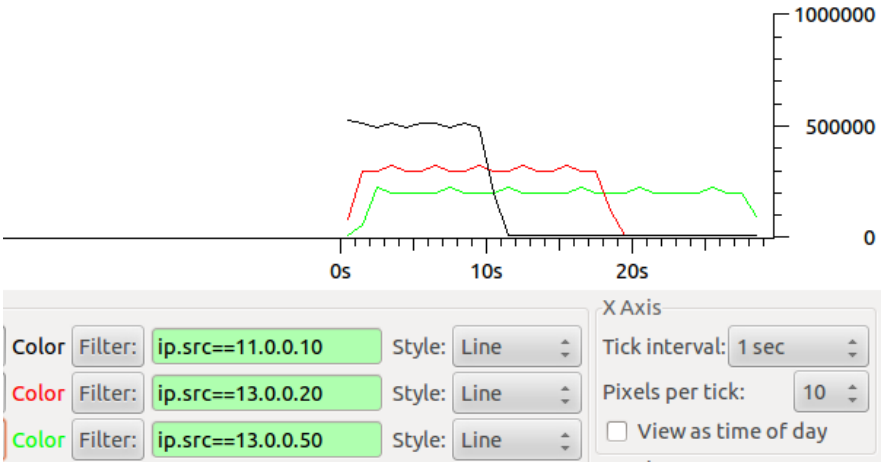
2. Partiendo de la situación inicial del escenario se configura en `r3(eth2)` HTB con limitación de 1Mbit repartido de la siguiente forma:

   - rate=700 kbit para el tráfico de `pc1` con ceil=1Mbit.
   - rate=100 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`: desde `pc1` a 500kbit hacia `pc3` y desde `pc5` a 500kbit hacia `pc4`. Indica cuál de las siguientes afirmaciones es correcta:

   (A) `pc3` recibirá 700kbit y `pc4` recibirá 200kbit durante los 10s que dura la transmisión. Después de esos 10s aproximadamente no se recibirá más tráfico.
   (B) `pc3` recibirá 500kbit y `pc4` recibirá 200kbit durante los 10s que dura la transmisión. Después de esos 10s aproximadamente no se recibirá más tráfico.
   (C) `pc3` recibirá 500kbit y `pc4` recibirá 500kbit durante los 10s que dura la transmisión. Después de esos 10s aproximadamente no se recibirá más tráfico.
   (D) `pc3` recibirá 500kbit y `pc4` recibirá 200kbit durante los 10s que dura la transmisión. Después de esos 10s `pc4` seguirá recibiendo durante más de 10s el tráfico que se había quedado encolado.

3. Partiendo de la situación inicial del escenario se realiza una configuración de disciplina de cola HTB en la interfaz `eth2` de `r3`. Utilizando `iperf` se envía tráfico durante 10 segundos con diferentes anchos de banda destinado a la subred 15.0.0.0/16 desde los siguientes pcs: `pc1`, `pc2` y `pc5` :



Indica cuál de las siguientes afirmaciones es correcta:

(A)
- En `pc1` se ha arrancado `iperf` para que envíe 500kbit aproximadamente
- En `pc2` se ha arrancado `iperf` para que envíe 300kbit aproximadamente
- En `pc5` se ha arrancado `iperf` para que envíe 200kbit aproximadamente

(B)
- En `pc1` se ha arrancado `iperf` para que envíe 500kbit aproximadamente
- En `pc2` se ha arrancado `iperf` para que envíe más de 300kbit
- En `pc5` se ha arrancado `iperf` para que envíe más de 200kbit

(C)
- En `pc1` se ha arrancado `iperf` para que envíe menos de 500kbit
- En `pc2` se ha arrancado `iperf` para que envíe más de 300kbit
- En `pc5` se ha arrancado `iperf` para que envíe más de 200kbit

(D)
- En `pc1` se ha arrancado `iperf` para que envíe 500kbit aproximadamente
- En `pc2` se ha arrancado `iperf` para que envíe 300kbit aproximadamente
- En `pc5` se ha arrancado `iperf` para que envíe más de 200kbit

4. Supón que `r3` tiene la siguiente configuración:

```
tc qdisc add dev eth1 handle ffff: ingress

tc filter add dev eth1 parent ffff: \
   protocol ip prio 6 u32 \
   match ip src 11.0.0.10/32 \
   police rate 256kbit burst 10k drop flowid :1
```

En la figura se muestra un paquete que ha enviado `r3` a través de su interfaz `eth2`:

```
▶ Frame 17: 1512 bytes on wire (12096 bits), 1512 bytes captured (12096 bits)
▶ Ethernet II, Src: 52:9d:bb:35:42:01 (52:9d:bb:35:42:01), Dst: f2:b1:bd:ff:60:97 (f2:b1:bd:ff:60:97)
▼ Internet Protocol Version 4, Src: 11.0.0.10, Dst: 15.0.0.30
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes
  ▶ Differentiated Services Field: 0x68 (DSCP: AF31, ECN: Not-ECT)
    Total Length: 1498
    Identification: 0x3bb4 (15284)
  ▶ Flags: 0x02 (Don't Fragment)
    Fragment offset: 0
    Time to live: 63
    Protocol: UDP (17)
  ▶ Header checksum: 0xdfcf [validation disabled]
    Source: 11.0.0.10
    Destination: 15.0.0.30
```

Si el paquete se había recibido en `r3` sin etiqueta DiffServ, indica cuál de las siguientes configuraciones en `r3` permitiría que `r3` hubiera enviado ese paquete:

(A)

```
tc qdisc add dev eth2 handle 1:0 root dsmark indices 8

tc class change dev eth2 classid 1:1 dsmark mask 0x3 value 0x1a
tc class change dev eth2 classid 1:2 dsmark mask 0x3 value 0x68

tc filter add dev eth2 parent 1:0 protocol ip prio 1 handle 1 tcindex classid 1:1
```

(B)

```
tc qdisc add dev eth2 handle 1:0 root dsmark indices 8

tc class change dev eth2 classid 1:1 dsmark mask 0x3 value 0x1a
tc class change dev eth2 classid 1:2 dsmark mask 0x3 value 0x68

tc filter add dev eth2 parent 1:0 protocol ip prio 1 handle 1 tcindex classid 1:2
```

(C)

```
tc qdisc add dev eth2 handle 1:0 root dsmark indices 8 set_tc_index
tc filter add dev eth2 parent 1:0 protocol ip prio 1 tcindex mask 0xfc shift 2

tc class change dev eth2 classid 1:1 dsmark mask 0x3 value 0x1a
tc class change dev eth2 classid 1:2 dsmark mask 0x3 value 0x68

tc filter add dev eth2 parent 2:0 protocol ip prio 1 handle 0x1a tcindex classid 1:2
```

(D)

```
tc qdisc add dev eth2 handle 1:0 root dsmark indices 8 set_tc_index
tc filter add dev eth2 parent 1:0 protocol ip prio 1 tcindex mask 0xfc shift 2

tc class change dev eth2 classid 1:1 dsmark mask 0x3 value 0x1a
tc class change dev eth2 classid 1:2 dsmark mask 0x3 value 0x68

tc filter add dev eth2 parent 2:0 protocol ip prio 1 handle 0x1a tcindex classid 1:1
```

HTTP

5. En la captura `/opt/stma/http-1.cap` aparecen mensajes correspondientes a la interacción entre un cliente HTTP y un servidor HTTP.

   Sabiendo que el cliente no tenía ninguna *cookie* almacenada ANTES de que realizara la interacción reflejada en la captura, si tras todos los mensajes que aparecen en ella el mismo cliente pidiese HOY la URL:

   `http://www2/dir1/departamento/index.html`

   indica qué cookies enviara dicho cliente en esa petición HTTP:

   (A) Ninguna

   (B) De la captura no puede deducirse qué *cookies* enviaría

   (C) Enviaría:
   `Cookie: Authenticated=YES; UserID=1111;`

   (D) Enviaría:
   `Cookie: Authenticated=YES; UserID=1111; Age=23;`

6. En la captura `/opt/stma/http-1.cap` aparecen mensajes correspondientes a la interacción entre un cliente HTTP y un servidor HTTP. En uno de esos mensajes el cliente envía los datos de un formulario al servidor usando el método POST. Indica cuál de las siguientes afirmaciones es correcta:

   (A) El cliente envía los datos del formulario usando POST debido a que pidió dicho formulario usando GET.

   (B) El cliente envía los datos del formulario usando POST debido a que el tamaño de los datos que contiene el formulario es superior a 255 caracteres.

   (C) El cliente envía los datos del formulario usando POST debido a que en dicho formulario estaba establecido que cuando se subieran sus datos debería hacerse utilizando POST.

   (D) El resto de afirmaciones son incorrectas.

7. Examina la captura `/opt/stma/http-2.cap` e indica cuál de las siguientes afirmaciones es correcta:

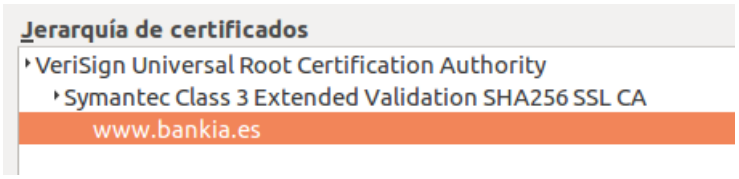
   (A) La captura contiene los mensajes que intercambian un cliente HTTP y el servidor HTTP de nombre www1.

   (B) La captura contiene los mensajes que intercambian un cliente HTTP y un servidor proxy HTTP.

   (C) La captura contiene los mensajes que intercambian un servidor proxy HTTP y el servidor final de nombre www1.

   (D) El resto de afirmaciones son incorrectas.

8. Al acceder a la página web `www.bankia.es` desde el navegador, se observa que la información que muestra la página sobre la jerarquía de certificados es la siguiente:



Indica cuál de las siguientes afirmaciones es correcta:

(A) Symantec Class 3 es una autoridad de certificación. El certificado de Symantec Class 3 estará firmado por VeriSign que es una autoridad de certificación raíz.

(B) El certificado de www.bankia.es está instalado en el navegador por ser certificado de una autoridad de certificación raíz y estará firmado por VeriSign que es otra autoridad de certificación de confianza

(C) El certificado de Verisign está instalado en el navegador y está firmado por la autoridad de certificación raíz `www.bankia.es`.

(D) El certificado de Symantec Class 3 está instalado en el navegador por ser certificado de una autoridad de certificación raíz y este certificado estará autofirmado.

9. En una sistema existe la autoridad de certificación raíz CA1 que ha incluido su propio certificado autofirmado en la aplicación de comunicaciones que se usa dentro de este sistema. Alicia tiene un certificado de su clave pública firmado por CA1. Roberto no tiene ningún certificado de su clave pública.

Indica cuál de las siguientes afirmaciones es correcta:

(A) Cuando Alicia le envía el certificado de su clave pública a Roberto, Roberto puede extraer la K_A^+ y comprobar el certificado con su clave privada K_R^-.

(B) Cuando Alicia le envía el certificado de su clave pública a Roberto, Roberto puede extraer la K_A^+ y comprobar el certificado con la clave privada K_{CA1}^- instalada en el navegador.

(C) Cuando Alicia le envía el certificado de su clave pública a Roberto, Roberto puede extraer la K_A^+ y comprobar el certificado con la clave pública K_{CA1}^+ instalada en el navegador.

(D) Cuando Alicia le envía el certificado de su clave pública a Roberto, Roberto puede extraer la K_A^+ pero no tiene forma de comprobarlo porque él no tiene ningún certificado de su clave pública K_R^+.

10. Alicia y Roberto usan criptografía de clave pública para intercambiar mensajes. Se sabe que Alicia y Roberto se han intercambiado sus claves públicas de forma segura.

Alicia decide que a partir de ahora van a usar criptografía de clave simétrica y para ello Alicia eligirá una K_s y se la comunicará a Roberto.

Indica qué propiedades debería tener el mensaje que Alicia envíe a Roberto con la clave K_s.

(A) Sólo confidencialidad

(B) Sólo autenticidad e integridad

(C) Confidencialidad, autenticidad e integridad

(D) En ningún caso se podría enviar un mensaje que contenga una clave simétrica.

11. Alicia, Roberto y Bárbara usan criptografía de clave pública para intercambiar mensajes. Se sabe que:

- Alicia y Roberto se han intercambiado sus claves públicas de forma segura.
- Alicia y Bárbara se han intercambiado sus claves públicas de forma segura.
- Roberto no tiene la clave pública de Bárbara.
- Bárbara no tiene la clave pública de Roberto.

Roberto y Bárbara confían plenamente en Alicia y saben que ella no va alterar el contenido de los mensajes, por ello deciden usar a Alicia para que realice la labor de intermediario.

Roberto desea enviar mensajes a Bárbara de forma que Bárbara esté segura de que el mensaje que recibe es el mismo que Roberto envió.

Roberto enviará el mensaje Alicia (quién no va alterar el mensaje) y Alicia enviará el mensaje a Bárbara.

Indica cómo podría ser la comunicación para garantizar que Bárbara ha recibido el mismo contenido de mensaje que le envió Roberto :

(A)
- Alicia recibe: $\boxed{\text{m}, \quad K_R^-(H(m))}$
- Alicia envía: $\boxed{\text{m}, \quad K_A^-(H(m))}$

(B)
- Alicia recibe: $\boxed{\text{m}, \quad K_R^-(H(m))}$
- Alicia envía: $\boxed{\text{m}, \quad K_A^-(K_R^-(H(m)))}$

(C)
- Alicia recibe: $\boxed{\text{m}, \quad K_A^+(H(m))}$
- Alicia envía: $\boxed{\text{m}, \quad K_B^+(H(m))}$

(D)
- Alicia recibe: $\boxed{\text{m}, \quad K_A^+(H(m))}$
- Alicia envía: $\boxed{\text{m}, \quad K_A^-(H(m))}$

IPTABLES

En la figura 2 se muestra la conexión de dos pequeñas empresas a Internet a través de un proveedor de servicios de Internet (ISP). Estas entidades quedan representadas en la figura de la siguiente forma:

- Empresa1: tiene las siguientes máquinas `e1-pc1` y `e1-pc2` que pertenecen a una subred privada, `e1-pc3` y `e1-pc4` que pertenecen a una zona DMZ y el *router firewall* `e1-fw`.
- Empresa2: tiene las siguientes máquinas `e2-pc1`, `e2-pc2` que pertenecen a una subred privada y el *router firewall* `e2-fw`.
- ISP: tiene un único *router* `isp-r1`.
- Internet: tiene las siguientes máquinas `i-pc1`, `i-pc2` y los siguientes *routers* `i-r1` y `i-r2`.

Las máquinas `e1-fw` y `e2-fw` están funcionando como *firewalls* a los que se les ha configurado únicamente las siguientes reglas:

- Políticas por defecto para las cadenas de entrada y reenvío (`INPUT` y `FORWARD`) configuradas para **descartar** paquetes.
- Política por defecto para la cadena de salida (`OUPUT`) configurada para **aceptar** paquetes.

Al arrancar los dos *routers* `e1-fw` y `e2-fw` han ejecutado unos *scripts* que aplican las reglas descritas previamente.

---

12. En la máquina `i-pc1` está instalado un servidor UDP esperando recibir mensajes en el puerto 13 (*daytime*). Cuando un cliente le envía cualquier tipo de mensaje, el servidor le devuelve la hora de ese instante.

    Partiendo de la configuración inicial, indica cuál es el **mínimo conjunto de reglas necesarias** en `e1-fw` para permitir que un cliente UDP en `e1-pc1` se comunique con dicho servidor, instalado en la máquina `i-pc1` y puerto 13, y obtenga la hora:

    (A)
    ```
    iptables -t nat -A POSTROUTING -p udp -s 10.0.0.10 -o eth0 -j SNAT --to-source 20.0.1.1
    ```

    (B)
    ```
    iptables -t filter -A FORWARD -p udp --dport 13 -d 20.0.4.10 -s 10.0.0.10 -j ACCEPT
    iptables -t nat -A POSTROUTING -p udp -s 10.0.0.10 -o eth0 -j SNAT --to-source 20.0.1.1
    ```

    (C)
    ```
    iptables -t filter -A FORWARD -p udp --dport 13 -d 20.0.4.10 -s 10.0.0.10 -j ACCEPT
    iptables -t filter -A FORWARD -p udp --sport 13 -s 20.0.4.10 -d 10.0.0.10 -j ACCEPT
    iptables -t nat -A POSTROUTING -p udp -s 10.0.0.10 -o eth0 -j SNAT --to-source 20.0.1.1
    ```

    (D)
    ```
    iptables -t nat -A PREROUTING -p udp --dport 13 -s 20.0.4.10  -j DNAT --to-destination 10.0.0.10
    iptables -t filter -A FORWARD -p udp --dport 13 -d 20.0.4.10 -s 10.0.0.10 -j ACCEPT
    iptables -t filter -A FORWARD -p udp --sport 13 -s 20.0.4.10 -d 10.0.0.10 -j ACCEPT
    iptables -t nat -A POSTROUTING -p udp -s 10.0.0.10 -o eth0 -j SNAT --to-source 20.0.1.1
    ```

13. Partiendo de la situación inicial, se ha realizado una configuración tanto en la tabla `nat` como en la tabla `filter` en `e2-fw` para permitir la siguiente comunicación:

    ```
    e2-fw~:# cat /proc/net/ip_conntrack
    tcp      6 431933 ESTABLISHED src=20.0.6.20 dst=20.0.2.1  sport=36303 dport=7     packets=4 bytes=221
                                  src=10.0.0.20 dst=20.0.6.20 sport=7     dport=36303 packets=3 bytes=169 [ASSURED]
    ```

    Nos fijamos sólo en la tabla `nat` de `e2-fw` en la que se habían definido las siguientes reglas:

    - Regla1:

      ```
      iptables -t nat -A PREROUTING -p tcp --dport 7 -s 20.0.6.20 -d 20.0.2.1 -j DNAT --to-destination 10.0.0.20
      ```

    - Regla2:

      ```
      iptables -t nat -A POSTROUTING -p tcp --sport 7 -s 10.0.0.20 -d 20.0.6.20 -j SNAT --to-source 20.0.2.1
      ```

    Justo después de mostrar la información anterior de `/proc/net/ip_conntrack`, indica qué reglas de la tabla `nat` se han cumplido en `e2-fw`:

    (A) Se ha aplicado la Regla1 a 4 paquetes y se ha aplicado la Regla2 a 3 paquetes.

    (B) Se ha aplicado la Regla1 a 3 paquetes y se ha aplicado la Regla2 a 4 paquetes.

    (C) Sólo se ha aplicado la Regla1 a 1 paquete y no se ha aplicado la Regla2.

    (D) Sólo se ha aplicado la Regla1 a 4 paquetes y no se ha aplicado la Regla2.

14. En la máquina `e1-pc4` está instalado un servidor TCP esperando recibir mensajes en el puerto 7 (*echo*). Cuando un cliente le envía cualquier tipo de mensaje, el servidor le devuelve el mismo mensaje que el cliente le envió.

    Partiendo de la configuración inicial, indica cuál es el **mínimo conjunto de reglas necesarias** en `e1-fw` para permitir que un cliente TCP en `i-pc2` se comunique con dicho servidor, instalado en la máquina `e1-pc4` y puerto 7, y obtenga la respuesta:

    (A)
    ```
    iptables -t filter -A FORWARD -i eth1 -o eth0 -s 20.0.0.40 -d 20.0.6.20 -p tcp --sport 7 -j ACCEPT
    iptables -t filter -A FORWARD -i eth0 -o eth1 -m state --state RELATED,ESTABLISHED -p tcp --dport 7 -j ACCEPT
    ```

    (B)
    ```
    iptables -t filter -A FORWARD -i eth0 -o eth1 -s 20.0.6.20 -d 20.0.0.40 -p tcp --dport 7 -j ACCEPT
    iptables -t filter -A FORWARD -i eth1 -o eth0 -m state --state RELATED,ESTABLISHED -p tcp --sport 7 -j ACCEPT
    ```

    (C)
    ```
    iptables -t filter -A FORWARD -i eth0 -o eth1 -m state --state RELATED,ESTABLISHED -p tcp --dport 7 -j ACCEPT
    ```

    (D)
    ```
    iptables -t filter -A FORWARD -i eth0 -o eth1 -s 20.0.6.20 -d 20.0.0.40 -p tcp --dport 7 -j ACCEPT
    ```

15. Partiendo de la configuración inicial se consulta la tabla `filter` de `e1-fw`:

    ```
    Chain INPUT (policy DROP 2 packets, 168 bytes)
     pkts bytes target     prot opt in     out     source               destination

    Chain FORWARD (policy DROP 0 packets, 0 bytes)
     pkts bytes target     prot opt in     out     source               destination

    Chain OUTPUT (policy ACCEPT 2 packets, 168 bytes)
     pkts bytes target     prot opt in     out     source               destination
    ```

    Indica cuál de las siguientes afirmaciones es correcta:

    (A) `e1-fw` ha reenviado 2 paquetes de `e1-pc4` hacia un pc de Internet y ha reenviado 2 paquetes desde ese pc de Internet dirigidos a `e1-pc4`.

    (B) `e1-fw` ha reenviado 2 paquetes de `e1-pc4` hacia un pc de Internet y ha descartado 2 paquetes desde ese pc de Internet dirigidos a `e1-pc4`.

    (C) `e1-fw` ha enviado 2 paquetes hacia un pc de Internet y ha descartado 2 paquetes desde un pc de Internet dirigidos a `e1-fw`.

    (D) `e1-fw` no ha recibido, enviado, ni reenviado ningún paquete.

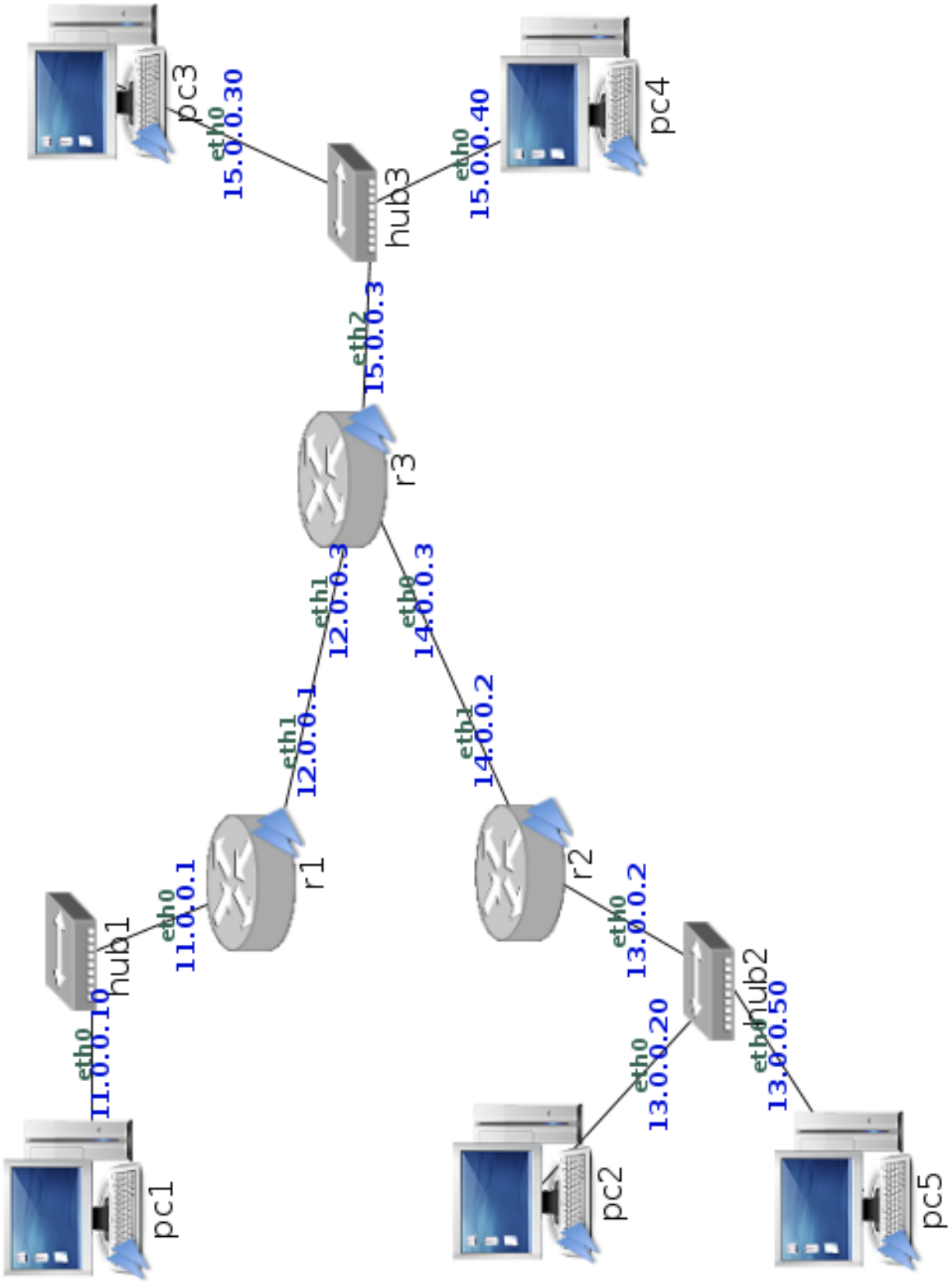


Figura 1: Calidad de servicio

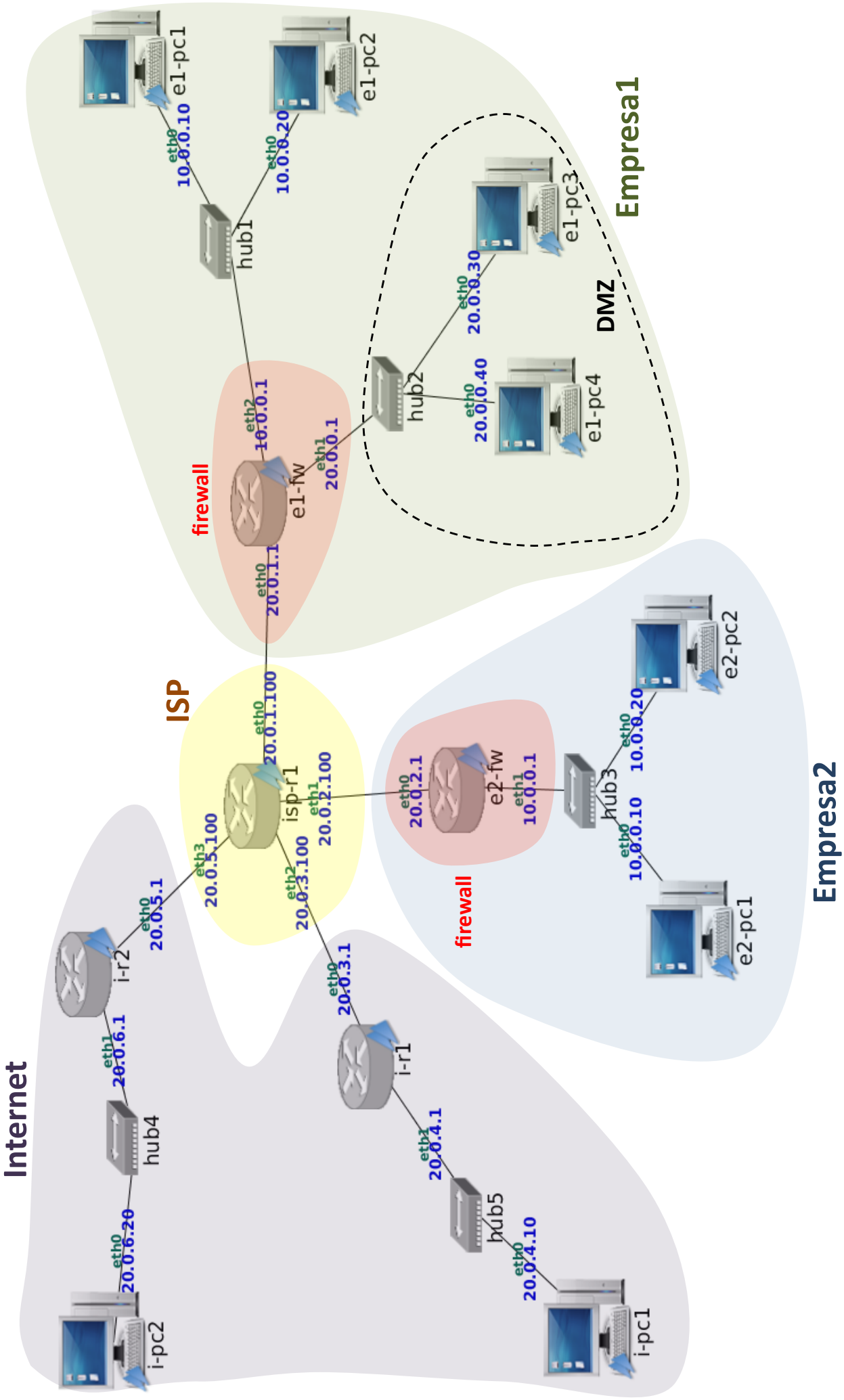


Figura 2: Seguridad