Protocolos para la Transmisión de Audio y Vídeo por Internet

Práctica Final
Opciones Avanzadas
(Versión v2.0.1, 9.1.2021)

January 9, 2021

Contents

1 Funcionalidades avanzadas

Además de la parte básica, se pueden realizar los siguientes requisitos avanzados que se evaluarán de manera complementaria a los requisitos básicos. Los requisitos avanzados se puntuarán con hasta 2 puntos. Cada requisito avanzado viene indicado con la puntuación máxima que se puede obtener.

Asimismo, se indica para cada requisito avanzado su forma de entrega, de manera que se posibilite su corrección (semi)automática. En general, para tal fin se pide que exista un fichero adicional llamado "avanzadas.txt" en el repositorio git.

Los requisitos avanzados son una lista cerrada. Si quieres realizar alguno que no esté incluido, ponte en contacto con los profesores para su aprobación.

A continuación se ofrece una lista de ejemplos de requisitos avanzados.

1.1 Cabecera proxy

Incluir funcionalidad en el proxy para que no retransmita SIP tal y como le llega, si no que añada la cabecera correspondiente marcando que ha pasado por el mismo.

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "Cabecera proxy".

Puntuación máxima: 0.1

1.2 Integración de (c)vlc

Tiene como objetivo lanzar *cvlc* (la instrucción para línea de comandos del programa vlc) desde el *User Agent* cuando se espere recibir audio vía RTP.

De esta manera, si se hubiera enviado audio a la IP 127.0.0.1 y al puerto 23032, habría que llamar a cvlc en la shell de la siguiente manera:

cvlc rtp://@127.0.0.1:23032

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "(c)vlc".

Puntuación máxima: 0.2

1.3 Consistencia del servidor proxy/registrar frente a valores erróneos

Esta funcionalidad adicional tiene como objetivo hacer el servidor Proxy/Registrar consistente frente a errores en los parámetros que se le pasan a los programas vía la línea de comandos o que reciben en los mensajes que intercambian.

Así, se comprobará que los parámetros pasados mediante la línea de comandos son correctos (p.ej., que el puerto sea un entero, que la IP sea de un rango que tenga sentido) o que los mensajes enviados siguen las especificidades del estándar que se detallan en esta práctica (p.ej., que el que hace el INVITE esté registrado, que el que realice el BYE sea uno de los participantes en la conversación, que el SDP esté bien formado y todos los valores sean correctos).

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "Consistencia".

Puntuación máxima: 0.3

1.4 Ejercicios alternativo G

Consiste en realizar el ejercicio alternativo G si este no era de obligatorio cumplimiento en la parte obligatoria de la práctica final.

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "Ejercicio alternativo G".

Puntuación máxima: 0.3

1.5 Ejercicios alternativo H

Consiste en realizar el ejercicio alternativo H si este no era de obligatorio cumplimiento en la parte obligatoria de la práctica final.

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "Ejercicio alternativo H".

Puntuación máxima: 0.3

1.6 Mensajería vía SIP

Siguiendo la extensión de SIP descrita en el RFC 3428, consiste en enviar mensajes de texto vía SIP. Para ello, se ha de implementar un nuevo método (MESSAGE) con sus correspondientes cabeceras (Content-Type y Content-Length). El Proxy-Registrar debería redirigir mensajes a usuarios registrados.

Un ejemplo de instrucción de *shell* para que un UAC envíe un mensaje sería:

```
$ python3 uaclient.py ua1.xml MESSAGE user2@domain.com Watson, come her
```

Esto resultaría en un paquete SIP como el siguiente que se enviaría al Proxy-Registrar (para que éste lo redirija a user2@domain.org):

```
MESSAGE sip:user2@domain.com SIP/2.0
Content-Type: text/plain
Content-Length: 18

Watson, come here.
```

Una vez recibido, el mensaje se imprimirá en el UAS (receptor, en el ejemplo user2@domain.org) por pantalla.

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "SIP MESSAGE".

Puntuación máxima: 0.4

1.7 Integración de (c)vlc con hilos

Tiene como objetivo lanzar *cvlc* (la instrucción para línea de comandos del programa vlc) desde el *User Agent* cuando se espere recibir audio vía RTP (véase ampliación anterior), pero lanzando un hilo. El programa principal deberá terminar el hilo cuando ya no tenga sentido la reproducción de audio.

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "(c)vlc con hilos".

Puntuación máxima: 0.4 (adicionales a los de "(c)vlc")

1.8 Notificaciones vía SIP

Siguiendo la extensión de SIP descrita en el RFC 3428, consiste en tener un sistema de suscripción donde al usuario le notificarían cada vez que otro usuario se ha registrado en el servidor Proxy-Register.

Un ejemplo de instrucción de *shell* para que un UAC (en este caso el de Batman) se pueda suscribir se puede ver a continuación:

```
$ python3 uaclient.py ua1.xml SUBSCRIBE
```

El paquete SIP resultante (a enviar al Proxy-Registrar) sería el siguiente:

```
SUBSCRIBE sip:batman@gotham.org SIP/2.0
```

El Proxy-Registrar responderá con un 200 OK:

```
200 OK SIP/2.0
```

Cada vez que otro usuario se registre (en este caso, Robin) en el Proxy-Registrar, este enviará una notificación al UAS que se ha suscrito (de Batman) como la siguiente (Nótese la cabecera *Reason*):

```
NOTIFY sip:batman@gotham.org SIP/2.0
Reason: SIP ;cause=200 ;text="robin@gotham.org just registered"
```

que el UAS (de Batman) imprimirá por pantalla, para después responder con un 200 OK al Proxy-Registrar:

```
200 OK SIP/2.0
```

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "SIP SUBSCRIBE".

Puntuación máxima: 0.5

1.9 Consistencia del servidor proxy/registrar frente a valores erróneos usando expresiones regulares

Esta funcionalidad adicional tiene el mismo objetivo que la funcionalidad "Consistencia", pero hará uso de la biblioteca de Python de expresiones regulares (regular expressions, re[1]). Se pide crear un patrón para los mensajes SIP y SDP que ha de cumplirse para que sean considerados como válidos.

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "Consistencia con expresiones regulares".

Puntuación máxima: 0.6 (no acumulables a los de "Consistencia")

1.10 Hilos para el envío de audio vía RTP

Tiene como objetivo que al enviar RTP, se genere un nuevo hilo previamente para que el envío de audio vía RTP no bloquee el programa.

El hilo principal debe ser consciente de que hay otro hilo ejecutándose. Así, en caso de recibir un "BYE", en envío de RTP se ha de parar. Por otra parte, si recibe una petición de INVITE mientras está enviando audio vía RTP, responderá con un "480 Temporarily Unavailable".

Forma de entrega: En el fichero "avanzadas.txt" se ha de indicar en una línea: "Hilos para el envío de audio vía RTP".

Puntuación máxima: 0.6

2 Entrega

Las opcionales se entregarán en el mismo repositorio que la parte obligatoria.

[1] https://docs.python.org/3/library/re.html