Memoria Práctica 3
Redes de comunicaciones II

Enrique Cabrerizo Fernández Guillermo Ruiz Álvarez

Curso 2013 - 2014
Universidad Autónoma de Madrid

Índice

1. Introducción y descripción

En este documento se detalla el proceso de modificación del servidor y el cliente IRC para soportar canales seguros. Para realizar este objetivo se crearán certificados firmados por una entidad certificadora creada por los alumnos, se creará un servidor y un cliente de eco para probar la funcionalidad creada y se modificarán el servidor y el cliente IRC creados en las prácticas anteriores utilizando las funciones implementadas para soportar canales seguros.

Para la creación de certificados y de la entidad certificadora se utilizará el software de OpenSSL en un script de bash llamado **cert.bash** que generará los certificados y a su vez los firma a través de la entidad certificadora creada por el mismo script.

2. Organización de directorios y ficheros.

En esta práctica se ha añadido un directorio más donde se almacenan los certificados generados por el alumno y la CA:

El directorio raíz, cuyo nombre es `G-2301-03-P2` contiene los siguientes directorios:

- **bin** contiene los ejecutables creados despues de la compilación y el enlazado.
- **cert** contiene los certificados generados y la CA.
- **doc** contiene la documentación del proyecto en formato PDF.
- **includes** contiene los ficheros de cabecera generados para la elaboración de librerias y el programa principal.
- **lib** contiene las librerias generadas para realizar los programas principales.
- **man** contiene los manuales[1] de las funciones de las librerías. Para acceder a ellos basta ejecutarlos con el programa `man`.
- **obj** contiene los ficheros objeto generados en la compilación.
- **src** contiene los ficheros fuente que generarán un ejecutable. Estos son programas de prueba (tests) o los programas principales.
- **srclib** contiene los ficheros fuente que generarán los objetos de las librerías.

En el directorio raíz se encuentran el fichero Makefile necesario para la compilación y enlazado de los archivos además del fichero de configuración de Doxygen.

[1] Estos manuales han sido generados con doxydoc de doxygen

3. Makefile

Para compilar y enlazar se proporciona un fichero Makefile a ejecutar con el programa `make`.

En esta práctica, además de añadir las nuevas funciones implementadas a las librerías correspondientes, se ha incluido una llamada a ejecución del script en bash creado para la generación y firma de los certificados.

3.1. Directivas

Las directivas proporcionadas son las siguientes:

- **all** genera todos los binarios en el directorio bin[2]. Estos ficheros son:

  G-2301-03-P1-main

  G-2301-03-P2-chat

  y todos los binarios creados para realizar tests.

- **IRC-SERVER** crea el ejecutable del servidor de la primera práctica en el directorio bin.

- **IRC-CLIENT** crea el ejecutable del cliente de la segunda práctica en el directorio bin.

- **bin/*** donde * es el nombre de un ejecutable. En este caso se compilará todo lo necesario para compilar y enlazar para la obtención del ejecutable.

- **obj/*.o** donde * es el nombre de un objeto. En este caso se compilará lo necesario para compilar el objeto.

- **compress**, **pack**, **tar**, **tgz**. Cualquiera de estas directivas ejecutan el comando clean y comprimen la práctica a un fichero tgz que será colocado en el directorio padre del raíz de la práctica.

- **clean**, **clear**. Cualquiera de estas directivas eliminan el fichero tgz, los objeto y los ejecutables.

- **G-2301-03-P2-*.a** genera la librería `G-2301-03-P2-*.a`. las librerias creadas son

  G-2301-03-P2-libconnection.a

  G-2301-03-P2-libirc.a

  G-2301-03-P2-libaudio.a

[2]incluyendo los ficheros objeto necesarios para ello en el directorio obj y las librerias en el directorio lib

3.2. Construcción del Makefile

El archivo Makefile (los comandos utilizados son de GNU Make) se ha realizado siguiendo las siguientes reglas, de forma que puede ser utilizado por cualquier usuario que siga las mismas:

- Seguir la organización de directorios especificada.
- No existen ficheros de cabecera para los ficheros fuente de src.
- Todos los ficheros de srclib tienen un fichero de cabecera asociado.

Para cambiar el nombre de los directorios o añadir ficheros de cabecera extra, basta con modificar las macros al incio del fichero Makefile.

4. Funciones de librería

En esta práctica se ha ampliado el número de librerías ofreciendo tres librerías diferenciadas según la utilidad de las mismas:

- **G-2301-03-P2-libconnection.a** ofrece todas las funciones necesarias de conexión, incluyendo el manejo de hilos, la daemonización y el uso de semáforos y las funciones de conexión segura.
- **G-2301-03-P2-libirc.a** ofrece todas las funciones necesarias relacionadas con IRC. Incluyendo funciones de cliente y servidor.
- **G-2301-03-P2-libaudio.a** ofrece todas las funciones necesarias relacionadas con las llamadas y el manejo de audio.

La descripción precisa de todas estas funciones se encuentra en la páginas de manual (`man pages`) realizadas por los autores en el directorio correspondiente.

5. Canales seguros

5.1. Funciones SSL

Para el soporte de canales seguros, se han implementado las funciones de SLL especificadas en el enunciado usando las funciones específicas de OpenSSL en el fichero **G-2301-03-P3-SSL_funcs.c**.

Dichas funciones se utilizan más tarde tanto para la implementación del servidor y el cliente de eco como para la modificación del cliente y el servidor creados y la transferencia de archivos por canal seguro.

5.2. Certificados y entidad certificadora

Para la creación de certificados y la firma de los mismos a través de la CA originada, se ha escrito un script en bash que es ejecutado junto con la compilación del proyecto.

Este script, a parte de crear la entidad certificadora para la firma de los certificados del cliente y del servidor, realiza la autofirma de dos certificados agregados (uno de cliente y otro de servidor) para comprobar que estos son rechazados por no haber sido firmados por la CA.

5.3. Servidor y cliente de Eco

Para las pruebas de la funcionalidad creada se han implementado un servidor y un cliente de Eco descritos a continuación.

El servidor de Eco utiliza las funciones implementadas para aceptar conexiones a través de un canal seguro y simplemente responde lo que ha recibido. El comando squit termina con el servidor finalizando la conexión segura.

El cliente de Eco utiliza las mismas funciones para enviar datos a través de un canal seguro. El comando cquit termina con el servidor finalizando la conexión segura.

5.4. Soporte de SSL del cliente y el servidor IRC

Para proporcionar funcionalidad de soporte de conexiones seguras al cliente y al servidor implementados se han sustituido las funciones de conexión, envío y recepción por las correspondientes seguras implementadas y descritas en secciones anteriores además de realizar las operaciones de inicialización de contexto y obtención de un canal seguro necesarias para completar dicho objetivo.

6. Transferencia de archivos

Para la tranferencia de archivos por canal seguro se ha realizado lo siguiente: cuando un extremo desea enviar un fichero dicho extremo actúa como servidor, de forma que el extremo contrario ha de realizar una conexión TCP para la recepción del fichero.

Al igual que como se hizo con el sonido, el acuerdo de esta conexión se realiza con comandos de IRC que se implementan en el cliente.

7. Pruebas

Para las pruebas del servidor se han utilizado tanto el certificado firmado por la CA como el autofirmado, obteniendo como resultado un rechazo del certificado que no ha sido firmado por la CA. En el caso del cliente, el resultado ha sido equivalente con los correspondientes certificados. Con wireshark se ha comprobado que los datos enviados en ambos casos aparecen cifrados.

En el caso del servidor y el cliente modificados para el soporte de SSL, se han realizado pruebas con XCHAT para comprobar que la conexión con el servidor se establece correctamente y con wireshark se ha observado que el flujo de datos enviado aparece encriptado.

Para ciertos módulos para los que se ha considerado oportuno, como en el caso de la tranferencia de archivos, se han creado test de prueba, cuyas fuentes se incluirán en el directorio src.