

Ejercicios sobre el paquete de comunicaciones *Lower_Layer_UDP*

Departamento de Teoría de la Señal y Comunicaciones y Sistemas Telemáticos y Computación (GSyC)

Octubre de 2016

Ejercicio 1

Instala en tu ordenador la biblioteca *Lower_Layer*¹:

1. Descarga `lower-layer-3.2.tgz` de la página de la asignatura en el Aula Virtual. En adelante supondremos que el fichero queda guardado en la carpeta Descargas dentro la Carpeta Personal (HOME) del usuario.
2. En una ventana de terminal, escribe las siguientes órdenes:

```
cd /usr/local
sudo rm -rf ll
sudo tar -xvzf ~/Descargas/lower-layer-3.2.tgz
cd ll/lib
sudo gnatmake lower_layer_udp.adb
```

Tras ejecutar estas órdenes la biblioteca queda instalada en el mismo lugar que en los laboratorios Linux de la URJC: en la carpeta `/usr/local/ll/lib`

Ejercicio 2

- Descarga los programas cliente (`client.adb`) y servidor (`server.adb`) de la página de la asignatura en el Aula Virtual. En adelante supondremos que copias dichos programas en la carpeta `intro-llu` debajo de tu `Escritorio`.
- Compila ambos programas. Para ello escribe en una ventana de terminal las siguientes órdenes:

```
cd
cd Escritorio/intro-llu
gnatmake -I/usr/local/ll/lib server.adb
gnatmake -I/usr/local/ll/lib client.adb
```

- Prueba ambos programas. Para ello abre 2 ventanas de terminal diferentes, y colócalas de forma que el contenido completo de ambas pueda verse a la vez. Cámbiate en las dos ventanas a la carpeta donde tengas los programas, y después haz, en este orden, lo siguiente:

1. En una de las ventanas de terminal, ejecuta el programa servidor

¹En las aulas Linux *Lower_Layer* ya está instalado. Estas instrucciones son para instalarlo en otros ordenadores. *Lower_Layer* sólo funciona con el sistema operativo Linux.

2. En la otra ventana de terminal, ejecuta el programa cliente
3. En la ventana del cliente, introduce una cadena de caracteres
4. Comprueba en las dos ventanas el resultado de la ejecución de cliente y servidor.

Ejercicio 3

Modifica el cliente y el servidor de forma que:

- el servidor reciba como argumento en la línea de comandos el puerto en el que se debe atar.
- el cliente reciba como argumentos en la línea de comandos el nombre de la máquina y el puerto en el que está atado el servidor.

Utiliza para ello las funciones `Get_Host_Name` y `To_IP`, así como el paquete `Ada.Command_Line`.

De esta forma para ejecutar el servidor en la máquina `alpha12` y que se ate en puerto `9001`, será necesario lanzar en la máquina `alpha12` el servidor en la forma:

```
./server 9001
```

Y para ejecutar el cliente en cualquier otra máquina (o en la misma `alpha12`) de forma que envíe su mensaje al servidor de `alpha12`, puerto `9001`, será necesario lanzar el cliente en la forma:

```
./client alpha12 9001
```

Nótese que el servidor no recibe el nombre de su máquina como argumento, por lo que debe averiguarlo llamando a la función `Get_Host_Name` de `Lower_Layer_UDP`. Por otro lado, tanto el cliente como el servidor deberán obtener la IP a partir del nombre de máquina usando `To_IP`.

Nótese también que en una instalación de Ubuntu el nombre de la máquina en la que se está es el nombre que se le dio al instalar el sistema operativo. Dicho nombre puede averiguarse con la orden `hostname` en una ventana de terminal.

Ejercicio 4

Modifica el cliente y el servidor de forma que:

- El cliente presentará al usuario un pequeño menú con 3 opciones: enviar al servidor un número entero, enviar al servidor una cadena de caracteres, o salir del programa.
Si el usuario elige enviar un número, el programa le pedirá su valor que almacenará en una variable `Integer`, si el usuario elige enviar una cadena, el programa le pedirá su valor que almacenará en una variable `Unbounded_String`.
- A continuación, el cliente enviará un mensaje al servidor con lo introducido por el usuario.
- El servidor estará preparado para extraer del mensaje que reciba del cliente lo que venga en él, sea entero o cadena. Tras extraerlo, si es un entero, el servidor enviará como respuesta un mensaje al cliente con el doble del entero recibido, y si es una cadena, el servidor enviará como respuesta un mensaje al cliente con la longitud de la cadena.
- El cliente recibirá la respuesta del servidor y mostrará el número recibido en pantalla, y volverá a presentar el menú al usuario, y repetirá todo hasta que el usuario elija en el menú la opción de salir.

MUY IMPORTANTE: Debes establecer un pequeño protocolo entre el cliente y servidor para que puedan “entenderse”, es decir, debes definir los tipos de mensajes que se van a usar, su formato, y las reglas de cuándo se envían o reciben.