

## Práctica 1: Conexión de PCs a través de modem

### 1- Objetivos

En las prácticas de esta asignatura trabajamos con routers CISCO y PCs con sistema operativo Linux. Una de las formas de configurar un router CISCO es a través de una conexión desde el puerto serie de un ordenador al puerto de consola del router. Por ello, lo primero que haremos será aprender a utilizar el puerto serie del PC. Para ello emplearemos un modem conectado al puerto serie de un PC y conectaremos dos PCs empleando modems y una llamada telefónica.

### 2- Material

- PCs
- Módems
- Centralitas telefónicas

### 3- Avisos generales

En los ordenadores dispuestos para la realización de estas prácticas (PC A, B y C) se ha creado una cuenta de nombre `lpr` y password `telemat`. Esta cuenta tiene permisos para ejecutar mediante el comando `sudo` ciertos comandos restringidos normalmente al superusuario. Igualmente se le han otorgado permisos para modificar el contenido de ciertos ficheros del sistema necesarios para la realización de la práctica. Para más detalle diríjense a la documentación sobre los armarios.

Si quieren conservar cualquier fichero entre sesiones guárdenlo en un disquete o un pendrive, dado que no se asegura que los ficheros creados o modificados durante una sesión de prácticas se mantengan para la siguiente.

### 4- Devices (dispositivos)

El acceso a los periféricos en Linux se realiza a través de los *devices*. Los devices están en el sistema de ficheros de Linux por lo que parecen ficheros, pero al abrirlos y leer y escribir sobre ellos en realidad estamos pidiendo al kernel de Linux que envíe o reciba información a un dispositivo determinado. El device que hace referencia al primer puerto serie suele tener de nombre `/dev/ttyS0`. Compruebe su existencia en su ordenador con `ls -l`. Podrá ver que hay varias diferencias con la salida para un fichero normal, por ejemplo:

```
$ ls -l /dev/ttyS0  
crw-rw---- 1 root uucp 4, 64 2009-09-22 20:40 /dev/ttyS0
```

Por un lado vemos que en la sección de permisos, la primera letra, que no es un permiso sino un indicador del tipo de fichero, vale `c`. Esto es lo que indica que éste no es un fichero ordinario, si fuera un fichero ordinario aparecería `-`, si fuera un directorio aparecería `d`. La `c` indica que el fichero es un device para un dispositivo a caracteres. Eso quiere decir que hace referencia a un dispositivo al cual se escribe byte a byte sin emplear un buffer.

También aparecen dos números que son toda la información que contiene este fichero. No es que haya un contenido del fichero, en realidad, el fichero solo sirve para indicar esos dos números. El

primero es lo que se llama el *major number* y hace referencia al driver dentro del kernel que debe emplear el sistema cuando abramos este fichero, leamos o escribamos en él. El segundo número es lo que se llama el *minor number* y se le pasará al driver como argumento (qué haga con él depende del driver en concreto, en uno de disco por ejemplo sirve para indicar la partición sobre la que trabajar). Podemos saber qué driver es el que se emplea con el major number indicado mirando el contenido del fichero `/proc/devices`.

Consulte el manual del comando `mknod` para saber cómo crear devices.

El device `ttys0` se utiliza normalmente para comunicarse con dispositivos conectados al primer puerto serie RS-232. En los PCs A y C se ha conectado un modem analógico(modem1 y modem2 respectivamente) en este puerto mediante su correspondiente cable serie RS-232 (lea la documentación sobre los armarios). Además se dispone de puertos serie USB que se utilizarán para la interconexión entre PCs mediante un cable “USB-USB NULL modem”.

## 5- minicom

El programa minicom nos permite enviar información al puerto serie (RS-232 ó USB) y dar órdenes al posible módem conectado a él, así como configurar la velocidad a la que funcionará el puerto serie. Lo que vamos a hacer a continuación es comunicar dos PCs entre sí empleando sus puertos serie USB y un cable “USB-USB NULL modem” entre ellos.

- Busque en Internet más información sobre este cable “USB-USB NULL modem”. Disponen de la referencia del fabricante en:

<http://www.ftdichip.com/Products/EvaluationKits/USB-USBNullModem.htm>

Revise la documentación sobre los armarios y verá que entre PC A y PC B hay un cable serie de este tipo uniendo los puertos serie USB 1 de ambos.

- Lance en PC A el programa minicom indicándole que cargue la configuración que abre el puerto serie USB 1 de este PC (está indicado en la documentación de los armarios pero por si acaso, es: `minicom ttyUSB1`). Lo que escriba se envía por el puerto serie a la velocidad configurada en ese momento. Pulsando Control-A seguido de Z aparece el menú de configuración de minicom. Aprenda a cambiar la velocidad del puerto serie USB.
- Lance en PC B también el minicom abriendo el puerto serie USB 1.
- Configure los puertos serie de los dos PCs con minicom a la misma velocidad y escriba texto que debería enviarse de un PC a otro.
- Observe qué sucede si configura distintas velocidades y después intenta enviar información.
- Pruebe las opciones ECHO y LINEFEED del minicom y configúrelo para realizar un chat entre los dos PCs.

Checkpoint 1.1: Muestran al responsable de prácticas que la comunicación les funciona correctamente

Observe que la configuración del puerto serie no es algo interno a minicom sino que se queda configurada en el dispositivo. Para probarlo:

- Configure una velocidad en minicom y cierre el programa sin resetear el dispositivo (*Quit with no reset* en el menú)

- Ahora puede probar que `/dev/ttyUSB1` se comporta como un fichero UNIX, sólo que en lugar de escribir a fichero se envía por el puerto serie USB y en lugar de leer de fichero se recibe por el puerto. Con el comando `cat` copie un fichero a `/dev/ttyUSB1` (`cat fichero > /dev/ttyUSB1`) y en el ordenador conectado con cable serie USB copie `/dev/ttyUSB1` a otro fichero (`cat < /dev/ttyUSB1 > fichero`).

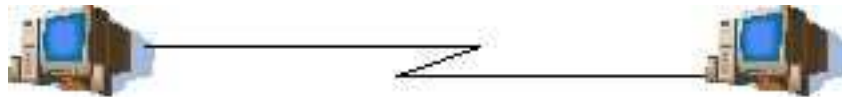


Figura 1.- Dos PCs con enlace punto-a-punto

Una última nota sobre los dispositivos: `/dev/ttyS0`, `/dev/ttyUSB1`, etc. Al comportarse como un fichero UNIX varios programas podrían tenerlo abierto al mismo tiempo pero al llegar datos por el puerto serie el sistema se lo entregará al primero en reaccionar de los programas que estén esperando. Para arbitrar ésto se utiliza un fichero de lock (`/var/lock/LCK..ttyS0`). Un programa, antes de abrir `/dev/ttyS0`, mira si ya existe el fichero lock y si no existe lo crea, con lo que se considera el dueño del dispositivo. Al finalizar el uso del device borra el fichero. `minicom` es uno de los programas que usa esto por lo que si se cierra de forma incorrecta y no borra el fichero se negará a volverse a abrir hasta que desaparezca el lock. Para arreglarlo sólo tiene que borrar el fichero `LCK..ttyS0`

## 6- Modems y comandos Hayes

A continuación vamos a establecer un enlace entre dos PCs empleando sus puertos serie, pero esta vez los PCs representan máquinas en ubicaciones alejadas, por lo tanto no podemos conectarlas mediante un simple cable serie. Lo que vamos a hacer es colocar unos modems que conviertan las señales de nuestros puertos serie en señales que se puedan transmitir por una línea telefónica tradicional. En nuestro caso realizaremos la llamada a través de dos centralitas telefónicas.



Figura 2.- Dos PCs con enlace punto-a-punto por módem

Primero probemos el modem:

- En el PC C abra con `minicom` el puerto serie RS-232 al que está conectado el modem.
- Observe que `minicom` envía una cadena de caracteres para configurar el módem (`AT S7=45 S0=0 L1 V1 X4 &c1 E1 Q0`) y que el módem responde OK. Si no ve esta cadena a lanzar `minicom` escríbala, desde el propio `minicom`, y pulse ENTER para enviársela al modem.
- Pruebe a escribir `AT` y pulsar ENTER. Este tipo de comandos que permiten dar órdenes al módem son un conjunto de comandos estándar que soportan la gran mayoría de modems y que se denominan comandos HAYES. Puede conseguirlos fácilmente en Internet o en el manual de la mayoría de los modems. Si no los encuentra, tiene un listado de ellos aquí:  
<http://docs.kde.org/en/3.1/kdenetwork/kppp/appendix-hayes-commands.html>
- Familiarícese con los comandos HAYES. Averigüe qué hace la cadena de inicialización que emplea `minicom`

A continuación probemos la línea:

- Averigüe los números de teléfono que corresponden a los teléfonos de su armario.
- Prueben las líneas llamándose de un teléfono a otro de su armario.
- Conecte sus módems a las líneas telefónicas a través del panel de parcheo. Consulte para ello la documentación sobre los armarios. ¿Qué tipo de cable empleará en el parcheo?
- Observe qué recibe minicom desde los devices al producirse la llamada

Finalmente establezcamos la comunicación:

- Busque los comandos HAYES necesarios para ordenar al módem realizar una llamada a un número de teléfono.
- Pruebe a ordenar a un módem que llame al otro módem y observe el resultado (o pruebe a llamar a un teléfono y escuchar el intento de conexión).
- Busque el comando HAYES para contestar una llamada.
- Pruebe a contestar una llamada y observe que la conexión tarda en establecerse. El comando hace que el módem descuelgue y acepte la llamada pero durante un tiempo los modems se ponen de acuerdo en la máxima velocidad a la que pueden comunicarse según el tipo de cada módem y la calidad de la línea telefónica. Esto es lo que se conoce como handshaking. A partir de este momento el módem informa al usuario (minicom en este caso) de que ha establecido la conexión. A partir de aquí los datos que enviamos al dispositivo se envían al otro extremo por el módem.
- Compruebe la utilización del comando +++ que si estamos en el modo de transmisión al otro extremo, nos permite volver a dar comandos al módem. Averigüe cómo podemos volver al modo de transmisión y cómo se puede finalizar la conexión. *Tenga en cuenta que la utilización del comando +++ , en el sentido indicado, no requiere que pulse ENTER. ¿Sabría explicar por qué?*
- Utilice lo aprendido hasta ahora para establecer un chat entre los dos PCs a través de una conexión telefónica.

Checkpoint 1.2: Muestre al responsable de prácticas que el chat telefónico les funciona correctamente

Al igual que en el caso de la conexión directa por el puerto serie USB, puede salir del minicom sin resetear el módem, dejando la conexión establecida.

- Compruebe que puede utilizar el comando cat para copiar ficheros.

## 7. Conclusiones

En esta práctica hemos aprendido a emplear el puerto serie de un PC con Linux. Lo hemos usado para realizar una conexión directa entre dos PCs empleando un cable NULL modem o una llamada telefónica con modems. El conocimiento adquirido sobre el manejo del puerto serie nos será muy útil a la hora de configurar los routers Cisco. Volveremos a los modems cuando vayamos a configurar un ISP que ofrezca acceso a Internet por modem.