



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA  
Escuela Técnica Superior de Ingenieros  
de Telecomunicación

PRÁCTICA 1  
**Introducción a la interconexión de  
ordenadores**

*Redes de Ordenadores*

*17 de octubre de 2005*

**Himar Alonso Díaz**

# Índice

1. Introducción	3
2. Conexión de tres ordenadores mediante un <i>hub</i>	3
3. Conexión de dos ordenadores mediante un cable <i>cruzado</i>	5
4. Ampliando la red mediante <i>Uplink</i>	6
5. Ethernet conmutada	6

# 1. Introducción

En esta práctica hemos aprendido algunos fundamentos sobre las redes de área local -LAN-. Para ello hemos realizado varios experimentos, desde una red formada por solo dos ordenadores, hasta una red más extensa para la cual hemos necesitado dos tipos de dispositivos principalmente: los *concentradores* o *hubs*, y un *conmutador* o *switch*.

Hemos trabajado bajo el sistema operativo *GNU/Linux* de modo que para poder ejecutar los programas de configuración de la red hemos hecho uso de *sudo*, un programa que permite al administrador del sistema *-root-* conceder ciertos permisos para poder modificar las configuraciones. Este es nuestro caso, así que al ejecutar cada programa, lo haremos anteponiendo la palabra *sudo*. Sin embargo, dado que este aspecto no resulta relevante en el contenido de la práctica, cada vez que aparezca un comando en este documento, *no* utilizaremos la palabra *sudo*.

Para construir nuestra red, a nivel físico, necesitaremos cable UTP (Unshielded Twisted Pair) y conectores RJ-45 (materiales) y una herramienta *crimpadora* (del inglés: *to crimp -trabar*). Existen varias categorías de cables UTP; a continuación mostramos las características de cada una.<sup>1</sup>

**Categoría 1** Especialmente diseñado para redes telefónicas, el clásico cable empleado en teléfonos y dentro de las compañías telefónicas.

**Categoría 2** Es también empleado para transmisión de voz y datos hasta 4Mbps.

**Categoría 3** Define los parámetros de transmisión hasta 16 MHz. Los cables de categoría 3 están hechos con conductores calibre 24 AWG y tienen una impedancia característica de 100 W. Entre las principales aplicaciones de los cables de categoría 3 encontramos: voz, Ethernet 10Base-T y Token Ring. Parámetro de transmisión Valor para el canal a 16 MHz. Atenuación 14.9 dB. NEXT 19.3 dB . ACR 4.0 dB.

**Categoría 4** Tiene la capacidad de soportar comunicaciones en redes de computadoras a velocidades de 20Mbps.

**Categoría 5** Un verdadero estándar actual dentro de las redes LAN particularmente, con la capacidad de sostener comunicaciones a 100Mbps.

**Categoría 6** Tiene los mismos uso y aplicaciones que los categoría 5, pero su construcción se hace cumpliendo unos mayores requerimientos de calidad, además de poseer entre sus pares un separador de plástico que reduce el efecto de diafonía entre pares, con esto se logra aumentar el ancho de banda hasta los 250MHz.

En esta práctica hemos construido un *cable cruzado* (como se explicará más adelante) para lo cual hemos utilizado un par de conectores RJ-45 y un trozo de cable de Categoría 6.

## 2. Conexión de tres ordenadores mediante un *hub*

El primer experimento realizado ha consistido en conectar 3 ordenadores en red, haciendo uso de un *hub*. Antes de ello hemos comprobado el funcionamiento de los programas *ifdown* e *ifup*. Con `$ifdown eth0` eliminamos la configuración de red del dispositivo *eth0*, que corresponde a la tarjeta de red, de modo que *no* tenemos conexión con ninguna máquina externa.

---

<sup>1</sup>Fuente: <http://www.arqhys.com/arquitectura/cables-tipos.html>

Para volver a conectar las máquinas basta con teclear `$ifup eth0`. El programa *ifup* utiliza un fichero de texto<sup>2</sup> donde se encuentran los parámetros relativos a la conexión (dirección *IP*, máscara de red, puerta de enlace...). Si queremos configurar la red manualmente, como haremos a continuación, debemos utilizar el programa *ifconfig*.

Si ejecutamos `$ifconfig` sin más, se muestra por pantalla una relación detallada de los dispositivos de red de nuestra máquina. En nuestro caso hay dos dispositivos:

- `eth0` - Que como dije antes, corresponde a la tarjeta de red.
- `lo` - Este dispositivo, llamado *Loopback*, no se corresponde con ningún aparato físico conectado al ordenador. Se trata de una conexión *virtual* con nuestra propia máquina. El *looback* puede resultar muy interesante cuando se desea probar un programa de red en un solo PC.

Al conectar los tres ordenadores en red mediante un *hub*, que es lo que pretendemos en este apartado, no podremos utilizar *ifup* para configurar la red, dado que no queremos utilizar una configuración por defecto, sino una manual. Para ello utilizaremos *ifconfig* seguido de unos parámetros que se detallan a continuación:

```
$ifconfig [dispositivo de red] [nombre de host]
```

Utilizaremos el dispositivo de red *eth0*, y como nombres para cada ordenador *-host-* PC1, PC2 y PC3. El procedimiento es igual para cada ordenador, pero por simplificar, supondremos que nos encontramos en el PC1:

```
$ifconfig eth0 pc1
```

Utilizaremos el programa *ping* para comprobar la conexión de red entre los ordenadores. Este programa envía (indefinidamente) paquetes de información al *host* de destino de forma que si la conexión está configurada correctamente, los paquetes son devueltos al *host* de origen. Pararemos el programa con `Ctrl+C`. Si escribimos:

```
$ping pc2
```

Obtenemos por pantalla:

```
64 bytes from 192.168.1.2: icmp_seq=0 ttl=64 time=1.6 ms
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=1.6 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=1.6 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=1.6 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=64 time=1.6 ms
(...)
```

Esto quiere decir que se están enviando paquetes de información de 64 bytes de tamaño al PC2, cuya dirección *IP* es 192.168.1.2, con un tiempo total de 1,6ms.

Una de las particularidades de esta red en la que hemos utilizado un concentrador *-hub-* para conectar los ordenadores, es que la información no *viaja* exclusivamente desde el *host* origen hasta el *host* destino, sino que se envía a todos los puntos de la red. Este hecho peculiar lo podemos comprobar con el programa *tcpdump*. Se trata de un *monitor* de red, que nos muestra todo el tráfico de la misma. Así, si PC2 hace *ping* a PC3, y ejecutamos `$tcpdump3` en PC1, veremos en la pantalla de PC1 los paquetes que se están enviando los otros ordenadores.

---

<sup>2</sup>Este fichero está en `/etc/network/interfaces` en distribuciones *Linux* basadas en *Debian*, como es nuestro caso en el laboratorio (*Ubuntu*).

<sup>3</sup>El comando que utilizamos en el laboratorio tenía los siguientes parámetros: `$tcpdump -ttln ip` que sirven para *filtrar* el contenido de la monitorización. En adelante utilizaré solamente `$tcpdump` por simplicidad.

Extremo 1	Extremo 2
Pin 1	Pin 3
Pin 2	Pin 6
Pin 3	Pin 1
Pin 4	Pin 4
Pin 5	Pin 5
Pin 6	Pin 2
Pin 7	Pin 7
Pin 8	Pin 8

Figura 1: Tabla de patillajes

**Nota:** Si hemos utilizado el comando `$ifup eth0` para *levantar* la conexión de red y más adelante queremos *tirlarla*, debemos emplear `$ifdown eth0`. Sin embargo, si hemos utilizado `$ifconfig eth0 [host]` para configurarla manualmente, cuando queramos eliminar esta configuración, debemos emplear `$ifconfig eth0 down`<sup>4</sup>.

Concluimos que la utilización de un *hub* nos permite disponer de una red de ordenadores conectados entre sí, pero al mismo tiempo tiene algunos inconvenientes, cuya solución veremos más adelante, como son:

**La información** que enviamos es accesible desde cualquier *host* que esté conectado al *hub* aunque éste no sea el destinatario, con lo cual nuestra red puede resultar insegura.

**El ancho de banda** también se ve afectado, dado que cuando se realiza una comunicación entre dos *hosts*, la red queda ocupada, es decir, no se puede realizar más de una comunicación simultáneamente.

### 3. Conexión de dos ordenadores mediante un cable *cruzado*

Disponer de tan solo dos ordenadores conectados entre sí es el caso más sencillo de red que podemos montar. De hecho no necesitaremos un *hub* ya que con un cable podemos conectar ambas máquinas. Eso sí, no podrá ser un cable como los que hemos utilizado hasta ahora. Tendrá que ser un cable *cruzado*, es decir, se cruzan las conexiones desde un conector a otro, para que la señal de *Enviar* de uno se corresponda con la señal de *Recibir* del otro. El patillaje<sup>5</sup> se detalla en la Figura 1.

A la hora de hacer el cable cometimos el error de poner los conectores al revés. No es lo mismo contar las patillas de izquierda a derecha que a la inversa, puesto que las conexiones no son simétricas. Nos dimos cuenta del error cuando vimos por pantalla el mensaje que nos daba el programa `ping: Unreachable host - host` inaccesible.

Cuando se utiliza la *crimpadora* para fijar el conector RJ-45 al cable UTP ya no es posible volver a retirar el conector, de modo que tuvimos que cortar el cable y repetir el procedimiento de *crimpado*. En esta segunda ocasión el resultado fue satisfactorio.

<sup>4</sup>Fuente: `$man ifconfig`.

<sup>5</sup>Fuente: [http://www.ugr.es/~amroldan/utilidades/cableado/cables\\_rj45/rj45.htm](http://www.ugr.es/~amroldan/utilidades/cableado/cables_rj45/rj45.htm)

## 4. Ampliando la red mediante *Uplink*

Cuando se presenta la necesidad de disponer de una red muy grande (tanto que el uso de un sólo *hub* no sea suficiente) podemos conectar nuestra red a un nuevo *hub*, como si de un ordenador más se tratase, mediante el puerto llamado *uplink*. De esta forma podemos crear una red tan amplia como queramos.

Lo que realmente hace el puerto *uplink* es *cruzar* las patillas del cable que conectamos, para así poderlo conectar al otro *hub*, como dije antes, como si nuestra red fuese un solo ordenador. De manera que si nuestro *hub* no tuviese puerto *uplink* bastaría con conectar un cable cruzado en un puerto cualquiera (en nuestro experimento, comprobamos que de ambas formas se obtiene el mismo resultado). Utilizar varios *hubs* de la forma que acabo de explicar equivale completamente a utilizar un solo *hub* más grande, por lo que las pruebas realizadas no aportan novedades:

La comunicación de dos PCs cualesquiera de la red es posible (esto lo comprobamos con *ping*), y además la información que se envía de un PC a otro es *visible* en cualquier punto de la red (esto lo comprobamos con *tcpdump*).

## 5. Ethernet conmutada

Como hemos visto en los apartados anteriores, podemos configurar una red tan amplia como queramos, mediante el uso de *hubs*. Sin embargo, habíamos visto que estos concentradores presentaban unas desventajas considerables, referentes a la privacidad y a la ocupación de la red, y la repercusión que ello tenía en el ancho de banda.

Para dar solución a estos problemas, existe un dispositivo denominado *conmutador*, más conocido por su nombre en inglés *switch*. Este aparato es funcionalmente similar a un *hub*, pero presenta las siguientes diferencias, principalmente:

- El *hub* envía la información del *host* origen a todos los puntos de la red, mientras que el *switch* conecta el *host* origen con el *host* destino, sin que en otro punto de la red pueda observarse esta conexión.
- Como consecuencia de ello, en un *switch* se pueden realizar varias conexiones al mismo tiempo, ya que al ser independientes, no es necesario esperar a que acabe una comunicación para iniciar otra. Esto permite *independizar* el ancho de banda de cada conexión.
- Para la construcción de un *hub* sólo se precisan elementos físicos *-hardware-*, mientras que un *switch* además necesita una pequeña memoria con un *software* que permita gestionar las conexiones.

En el siguiente experimento comprobaremos las ventajas que nos aporta disponer de un *switch* en nuestra red. Conectaremos cada 3 ordenadores, de los 12 del laboratorio, a un *hub* (necesitamos 4 *hubs*) y cada uno de los *hubs* irá conectado a un mismo *switch* a través del puerto *uplink*. Esta es la configuración por defecto de los ordenadores de la sala, así que para activar la red ejecutaremos:

```
$ifup eth0
```

Una vez configurada la red, hemos comprobado (utilizando *ping*) que es posible comunicar dos PCs cualesquiera del laboratorio. Sin embargo, y como novedad, observamos que el *tcpdump* sólo puede *ver* las comunicaciones de red que se dan en la misma *subred* en la que se halla el

monitor. Si por ejemplo tenemos un monitor (*tcpdump*) en un PC del *hub1*, no podremos ver las comunicaciones entre dos PCs que estén conectados a otro *hub* que no sea el 1.

Esta configuración que combina el uso de concentradores y conmutadores es ideal para separar una red en *grupos de trabajo*. Es decir, todas las máquinas están conectadas en red, pero las que estén en un mismo grupo de trabajo (*hub*) no podrán acceder a la información que se está transmitiendo por otro distinto (con lo cual solucionamos el problema de la *privacidad*), y además si un grupo de trabajo está haciendo una transferencia de archivos entre dos de sus PCs, los demás grupos de trabajo no experimentarán una ralentización en su conexión (por lo que también solucionamos el problema del *ancho de banda*).