

Uso de servidores linux

F. Alonso, J.F. Calvo y J.A. Palazón
Universidad de Murcia

jueves, 22 de febrero 2006

Resumen

Este documento ha sido preparado para los alumnos de las asignaturas de *Ecología Terrestre*, optativa para la Licenciatura de Biología, y *Modelización de Sistemas Ambientales*, optativa para la Licenciatura de Ciencias Ambientales, en la Universidad de Murcia, que se imparten en formato “virtual”. El desarrollo público de esta asignatura se hace, respectivamente, desde las páginas <http://fobos.bio.um.es/asignaturas/ET> y <http://fobos.bio.um.es/asignaturas/MSA>.

El documento tiene como objetivo proporcionar los conocimientos y conceptos fundamentales para utilizar una cuenta en un servidor GNU/LINUX, y en particular los programas específicos para el desarrollo de las actividades prácticas de la asignatura (R, GRASS, ...).

Índice

1. Servicios y servidores	1
1.1. ¿Dónde está mi servidor?	2
1.2. ¿Cómo puedo usar un servidor?	2
1.3. ¿Qué tareas puedo realizar en mi servidor?	3
1.3.1. Transferencia de archivos	3
1.3.2. Ejecución de programas instalados en el servidor	3
1.3.3. Explotación de las bases de datos del servidor	3
2. Sesiones de trabajo en servidores	3
2.1. Locales	3
2.1.1. Consola	3
2.1.2. Gráfica	4
2.2. Remotos	6
2.2.1. Consola	6
2.2.2. Gráfica	6
2.3. Otras	7
2.3.1. Conexión en modo texto	7
2.3.2. Conexión gráfica	7
3. Transferencia de archivos	8
4. Archivos y directorios	9
4.1. Nombres	9
4.2. Archivos	9
4.3. Creación y edición de ficheros	10
4.4. Directorios	11
4.5. mc: la navaja suiza	12

1. Servicios y servidores

Un servidor¹ es un ordenador en el que se proporcionan servicios desde programas tales como: servidores web, servidores ftp, servidores de correo, ..., o proporcionan servicios de disco remoto, de impresoras remotas, o cuentas para realizar sesiones de trabajo remotas.

¹<http://es.wikipedia.org/wiki/Servidor>

El siguiente documento describe los procedimientos básicos para explotar adecuadamente las prestaciones de un servidor GNU/LINUX, con especial hincapié en las sesiones de trabajo.

Utilizaremos sólo programas libres, ya sean para plataformas libres (GNU/LINUX) o para las privativas (WIN.XX, en las que las licencias² de uso imponen restricciones). En todos los casos la descarga será además gratuita.

1.1. ¿Dónde está mi servidor?

Existen dos posibilidades para trabajar con un servidor: local y remota. Cuando nos sentamos ante el teclado, monitor y ratón del servidor hablamos de **conexión local**. Cuando usamos un ordenador distinto hablamos de **conexión remota**.

Obviamente, en el primer caso conozco la ubicación del servidor y está accesible para su uso; en el segundo sólo podré acceder conociendo su dirección en Internet.

En nuestro caso el servidor no está disponible para los usuarios, por lo que es necesario una conexión remota. No obstante resulta de interés conocer la manera de actuar en una conexión local pues es posible utilizar un servidor propio fácilmente con una distribución del tipo *cd – live*³.

La dirección de Internet se asocia a un número llamado **IP**, propio de cada ordenador, conectado a Internet. El número IP consta de cuatro valores separados por un punto: **155.54.0.46** y a él se asocia un nombre: **deimos.bio.um.es**. El nombre está compuesto por dos partes: el nombre, propiamente dicho, del servidor, por ejemplo: **deimos** y el dominio, o ubicación lógica del servidor: **bio.um.es**. La conexión entre IP y nombre se hacen merced a los servidores de dominio o de nombre, abreviadamente DNS.

1.2. ¿Cómo puedo usar un servidor?

Es necesario que el administrador del servidor proporcione una cuenta personal a cada usuario, en caso contrario, sólo pueden usarse servicios generales como el correspondiente al web.

Con la cuenta el usuario adquiere un nombre público o **login** y validará su conexión con una contraseña, que puede modificar libremente.

La selección de contraseña ha de ser cuidadosa, se consideran malas contraseñas:

- Palabras que aparezcan en un diccionario.
- Palabras en idiomas extranjeros.
- Transformaciones simples de palabras.
- Nombres propios, nombres de ciudades, e iniciales.
- Palabras solo en mayúsculas o solo en minúsculas.
- Secuencias alfabéticas o de teclado: qwerty, 12345
- Palabras de menos de 4 caracteres.
- Passwords que incluyen números telefónicos, fechas, número de identificación de cuentas, nombre de familiares o mascotas, nombre de sus cantantes o grupos musicales, productos, etc.

En un servidor GNU/LINUX cada usuario acepta, al conectarse, las reglas de uso del mismo, que básicamente son de respeto al sistema y a los demás usuarios.

Habitualmente pueden encontrarse conectados a un servidor numerosos usuarios dado que se trata de servidores multiusuario-multitarea. Esta consideración es importante dado que la realización de realizar tareas que cargue en exceso la memoria o la CPU del servidor será un inconveniente para todos los usuarios.

²<http://www.gnu.org/licenses/licenses.es.html>

³http://es.wikipedia.org/wiki/Live_CD

1.3. ¿Qué tareas puedo realizar en mi servidor?

El usuario de un servidor puede utilizarlo con diversos fines, que fundamentalmente pueden considerarse las siguientes:

1.3.1. Transferencia de archivos

Es posible transferir (copiar) archivos (en inglés: *files*, ficheros) de un ordenador a otro con la ayuda de protocolos estándar, como el **FTP** (*file transfer protocol*). Por ejemplo:

- si dispongo de un archivo que sólo puede ser abierto por un programa que no tengo y está disponible en el servidor.
- si he obtenido un archivo tras un trabajo en el servidor y quiero tenerlo en mi ordenador.
- si necesito descargar archivos que están disponibles en un servidor anónimo de ftp.

1.3.2. Ejecución de programas instalados en el servidor

Tras una conexión, más o menos segura (`telnet` o `ssh`), el servidor dará la posibilidad de ejecutar los programas para los que tenemos permiso de ejecución. Por ejemplo, el programa que permite el cambio de contraseña o la modificación, o el borrado, de un archivo.

1.3.3. Explotación de las bases de datos del servidor

Se trata de un caso particular del anterior, pues utilizando algunos programas podrán consultarse bases de datos que están disponibles para determinados usuarios. Por ejemplo, el grupo de usuarios ET de fobos tiene acceso a una base de datos bibliográfica de la asignatura *Ecología Terrestre*.

2. Sesiones de trabajo en servidores

- Esto parece prometedor, pero ¿tengo que saber algo más antes de embarcarme en el uso de la cuenta?
- Pues sí, veamos, en primer lugar, como acceder al servidor GNU/LINUX.

2.1. Locales

Si nos sentamos al teclado de un servidor y miramos el monitor veremos que nos reclama nuestro `login` y `password`. La forma de hacerlo dependerá de la configuración del sistema. Puede ser en modo texto (llamando también consola) o en modo gráfico. Posteriormente veremos como decidir la manera de trabajar (gráfica o texto) en función de nuestras necesidades o conveniencia, por ejemplo en una conexión lenta donde es preferible usar el modo texto.

2.1.1. Consola

Podemos utilizar una consola en modo texto para empezar a trabajar. Inicialmente veremos un mensaje, muy espartano, como este:

```
login: _
```

o bien uno más informativo:

```

Bienvenido al servidor ET.bio.um.es
Para los alumnos de Ecología Terrestre
Departamento de Ecología e Hidrología
Universidad de Murcia

En caso de problemas póngase en contacto con el
administrador:
----->>>> ET@ET.bio.um.es
ET login: _

```

En ambos casos el cursor, parpadeante, nos indica que espera una acción por parte del usuario:

— Dime ¿quién eres?

Nos identificaremos por nuestro nombre público (`login`) y a continuación introduciremos nuestra contraseña. Se escribirá con “tinta invisible” y no sabremos siquiera cuantos caracteres hemos introducido (para borrar en caso de error utilizaremos `CTRL+U`, eliminando todos los caracteres introducidos). Si no hay errores nos mostrará un mensaje como este:

```

Linux et 2.6.8-1-k7 #1 Thu Nov 25 04:13:37 UTC 2004 i686 GNU/Linux
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

You have mail.

Last login: Mon Mar 14 09:16:51 2005 from deimos.bio.um.es

palazon@deimos:~$

```

Desde este momento pueden ejecutarse ordenes en el servidor; todas serán ejecutadas en forma local. Algunas ordenes pueden poner en marcha una sesión gráfica o una conexión con un servidor remoto.

En un ordenador con sistema operativo GNU/LINUX puede utilizarse varias consolas de texto —habitualmente seis— asociada cada una de ellas a una combinación de teclas `ALT+F1` ... `F6`.

2.1.2. Gráfica

Una vez hemos accedido a una sesión en modo texto en un servidor GNU/LINUX se pondrá en marcha una sesión gráfica con la ayuda de la orden:

```
palazon@deimos:~$ startx
```

Se inicia una sesión gráfica y esta quedará conectada a `ALT+F7`. Puede disfrutarse de más sesiones gráficas sin más que repetir la orden `startx` acompañada de un código (que por defecto es 0):

```
palazon@deimos:~$ startx -- :1
```

Después de esto, una sesión gráfica estará a nuestra disposición. Puede tener diversos aspectos dependiendo del gestor de ventanas o el escritorio elegido (Figura 1), desde los más sencillos que consumen pocos recursos hasta los más sofisticados con efectos tridimensionales...

En muchas ocasiones el servidor GNU/LINUX ofrece un acceso gráfico directamente, solicitando entonces el `login` y la contraseña como inevitable forma de acceso (Figura 2).

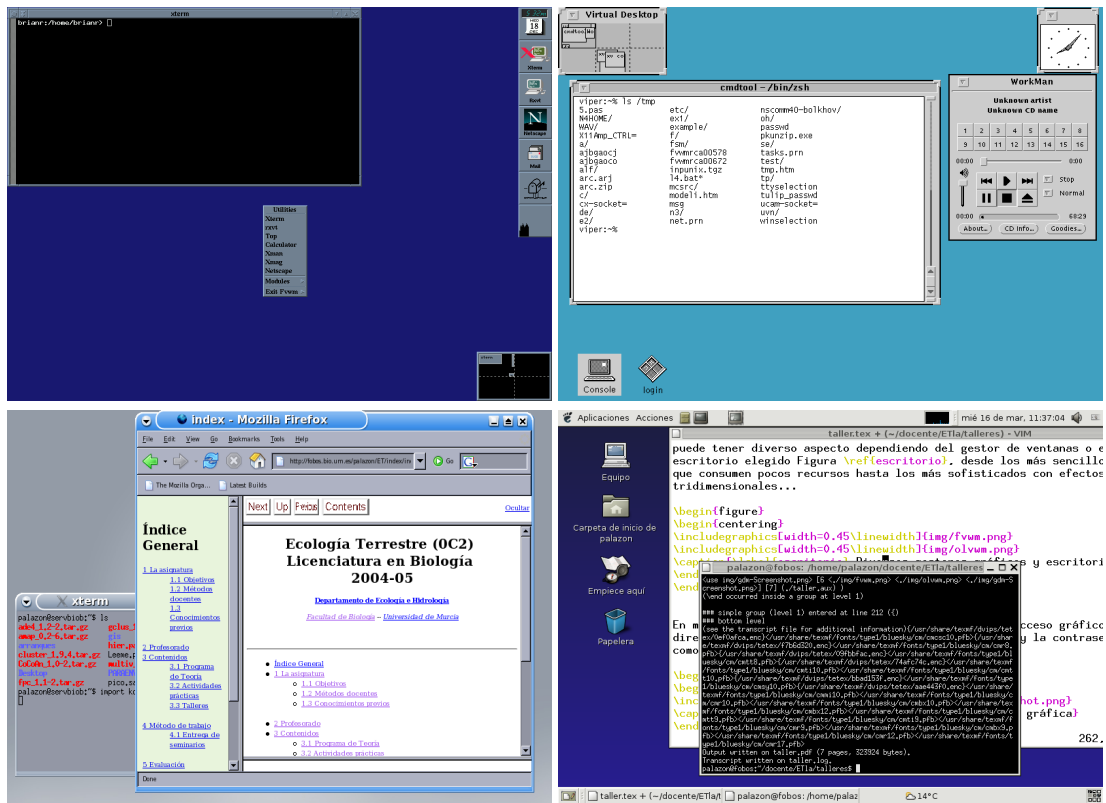


Figura 1: Diversos gestores gráficos y escritorios

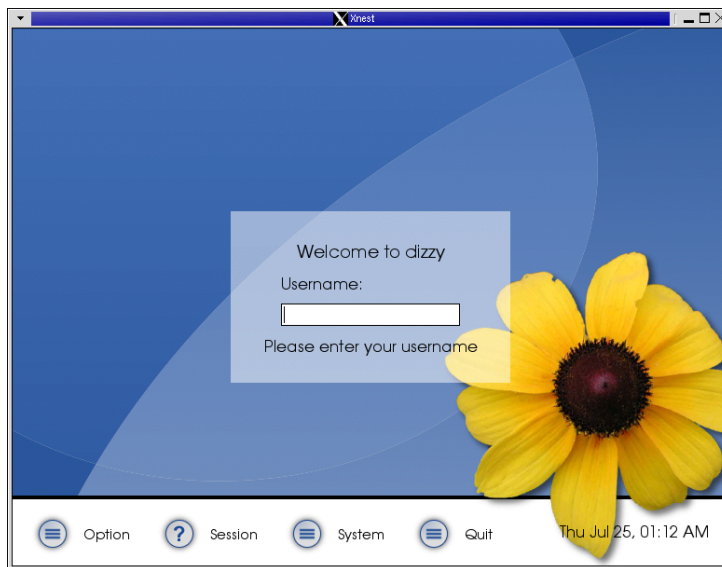


Figura 2: Pantalla de inicio de una sesión gráfica

2.2. Remotos

Una de las principales ventajas de disfrutar de una red universal es la posibilidad de utilizar máquinas remotas para diversos fines, sin duda, el más interesante es acceder a equipos remotos de mejores prestaciones o configuraciones especiales. Para comprobar que estamos en la misma red que el servidor podemos utilizar la orden:

```
palazon@deimos:~$ ping servbiob.bio.um.es
```

lo que devuelve información sobre la conexión con el servidor hasta que pulsemos **CTL+C**:

```
PING localhost.localdomain (127.0.0.1) 56(84) bytes of data. 64 bytes from servbiob.bio.um.es (127.0.0.1): icmp_seq=1
ttl=64 time=0.050 ms

64 bytes from servbiob.bio.um.es (127.0.0.1): icmp_seq=2 ttl=64 time=0.062 ms

64 bytes from servbiob.bio.um.es (127.0.0.1): icmp_seq=3 ttl=64 time=0.100 ms

64 bytes from servbiob.bio.um.es (127.0.0.1): icmp_seq=4 ttl=64 time=0.071 ms
```

2.2.1. Consola

Una conexión de consola o en modo texto se puede conseguir de forma sencilla. Se habla de conexión `telnet` (de características poco seguras), y de conexión segura `ssh`.

Para realizar cualquiera de estas conexiones basta con disponer de un programa que, dependiendo del sistema operativo, nos permita realizar la conexión. En GNU/LINUX podemos utilizar sencillamente (también puede utilizarse esta orden desde el *símbolo del sistema* de WIN.XX):

```
palazon@deimos:~$ telnet servbiob.bio.um.es
```

Si la comunicación es correcta se iniciará una sesión en modo texto como si estuviésemos ante la consola del mismo servidor, tras la consabida identificación del usuario. Desde este momento todas las ordenes dadas en la consola serán realizadas por el servidor, con sus programas y sus archivo de datos.

Puede ocurrir que el servidor no disponga de servicio `telnet`, produciéndose el siguiente mensaje:

```
palazon@deimos:~$ telnet servbiob.bio.um.es
Trying 155.54.4.17...
telnet: Unable to connect to remote host: Connection
refused
```

En este caso debemos utilizar una conexión `ssh`.

```
palazon@deimos:~$ ssh servbiob.bio.um.es
```

Si la cuenta en el servidor no coincide con el `login`, por ejemplo, si en `servbiob`, queremos entrar como usuario `japf`, entonces especificaremos esta opción mediante:

```
palazon@deimos:~$ ssh japf@servbiob.bio.um.es
```

2.2.2. Gráfica

Desde una consola de texto en un servidor local podemos realizar una conexión gráfica de una forma sencilla:

```
palazon@deimos:~$ X -query servbiob.bio.um.es
```

o en su caso:

```
palazon@deimos:~$ Xwrapper -query servbiob.bio.um.es
```

En el caso de tener ya una sesión gráfica local o en otro servidor puede elegirse una nueva mediante:

```
palazon@deimos:~$ X -query servbiob.bio.um.es :5
```

Para conmutar entre tareas sesiones gráficas se utiliza la combinación de teclas `CTL+ALT+Fn` (con los valores 7 a 12 para n), la relación entre la tecla y la sesión gráfica se basa en el orden de inicio de estas.

2.3. Otras

Existen diversos problemas que pueden plantearse en para el uso de servidores GNU/LINUX. El principal, es no estar en un servidor GNU/LINUX. Esto se resolverá inicialmente con la instalación de algunos programas disponibles para distintas plataformas, hablaremos aquí, por ejemplo, de las WIN.XX.

Para hacer una conexión desde una máquina con WIN.XX es posible recurrir al programa `telnet` de que dispone el sistema, pero es bastante flojo y no permite una conexión segura, por ello es necesario descargar algunos programas de licencia abierta (GPL o GNU), por ejemplo, desde la página⁴ de software libre de la Universidad de Murcia.

2.3.1. Conexión en modo texto

Para conexión `telnet` o `ssh` nos bastará con el programa `putty`⁵ que nos permiten realizar ambos tipos de conexiones.

2.3.2. Conexión gráfica

Programa `cygwin` :

El programa produce una “linuxización”, valga el término, del sistema local WIN.XX. Con él puede abrirse una consola desde la que utilizar las ordenes propias de GNU/LINUX: `X -query servidor ...`

Programa `TightVNC` :

Este es un caso muy interesante pues permite utilizar máquinas de forma remota, en lo que se llama `vnc` o *virtual network computing*. En nuestro caso, y siempre que el servidor disponga del programa `vncserver`, procederemos en dos pasos.

1. Es necesario conectarse al servidor, habitualmente por `ssh`, e iniciar una sesión mediante la orden:

```
palazon@deimos:~$ vncserver
New 'X' desktop is deimos:2
Starting applications specified in
/home/palazon/.vnc/xstartup
Log file is /home/palazon/.vnc/deimos:2.log
```

Si es la primera vez que solicitamos este servicio el sistema nos reclamará una contraseña específica del mismo, que puede coincidir con la de conexión habitual. Recordaremos el código de la sesión que hemos iniciado; en este casos es :2.

2. Desde el sistema WIN.XX o incluso desde otro sistema GNU/LINUX podemos utilizar un programa de conexión:

En el primer caso pondremos en marcha el programa `TightVNC` desde su icono, o desde el menú, indicaremos que queremos hacer una conexión a `deimos.bio.um.es:2`, escribiremos la contraseña y tendremos una ventana con la conexión gráfica a nuestro servidor (Figura 3).

Las características de la conexión pueden ser modificadas dependiendo de los resultados deseados y el ancho de banda, por ejemplo, mediante la opción *Conexion options...* puede solicitarse mayor o menor calidad en la visualización. Para obtener el menú de modificación debe pulsarse sobre la barra de estado de la ventana de `vnc`.

Las condiciones de trabajo —color del fondo, tipo de ventanas, gestor de ventanas, ...— del servidor deben cambiarse como si se tratase de una sesión habitual. Las dimensiones de la ventana deben especificarse en el arranque:

```
palazon@deimos:~$ vncserver geometry 640x400
```

con una menor resolución conseguimos mayor velocidad en la comunicación.

⁴<http://www.um.es/atika/softla/CD/softla-redes.html>

⁵<http://fobos.bio.um.es/programas>

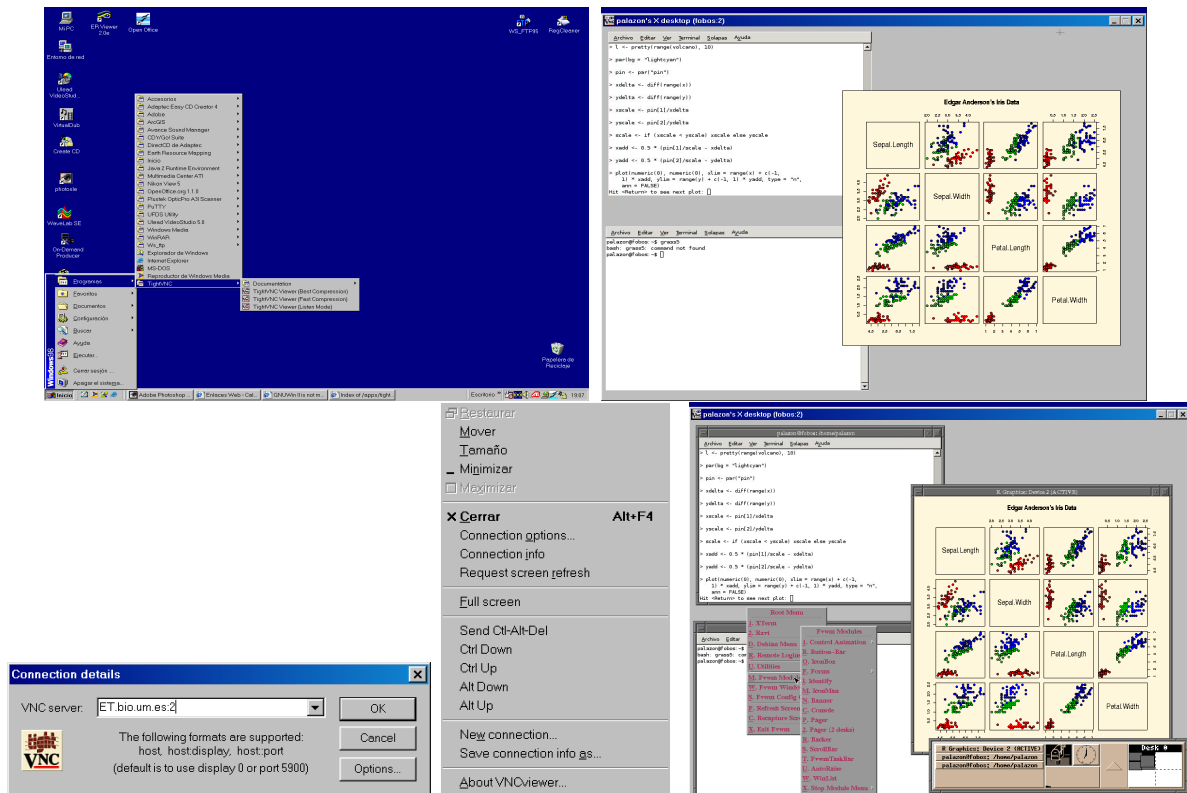


Figura 3: Conectando por vnc desde WIN.XX

- Para terminar la actividad en el servidor bastará con cerrar la ventana. Cada vez que deseemos usar esta sesión podemos recurrir a una nueva llamada, por ejemplo, la invocaremos desde una terminal gráfica de GNU/LINUX

```
palazon@deimos:~$ xvnviewer deimos.bio.um.es:2
```

La sesión se abrirá tal como la dejamos tras la última conexión.

- Para cerrar la sesión es necesario volver a conectarse al servidor y utilizar la orden:

```
palazon@deimos:~$ vncserver -kill :2
```

La sesión se cerrará, también se cerrará en caso de caída del servidor. Conviene cerrar la sesión al finalizar el trabajo.

3. Transferencia de archivos

La transferencia de archivo requiere que uno de los dos ordenadores, al menos, tenga un servicio de ftp. Habitualmente este estará en el servidor GNU/LINUX. Para copiar archivos utilizaremos la orden `ftp servidor` desde una conexión texto o la ventana de ejecutar. Sin embargo puede resultar más agradable utilizar una herramienta gráfica, así desde GNU/LINUX podemos usar el programa `filezilla` (Figura 4).

Los términos habituales en la jerga del `ftp` tienen que ver con copiar en el servidor los archivos (`put`), cogerlos o descargarlos (`get`), si la copia se hace literal (`bin`) o por el contrario se permite la interpretación (`ascii`), ...

Siempre que se desea transferir archivos debe tenerse en cuenta el tamaño y disponibilidad de espacio en los discos.

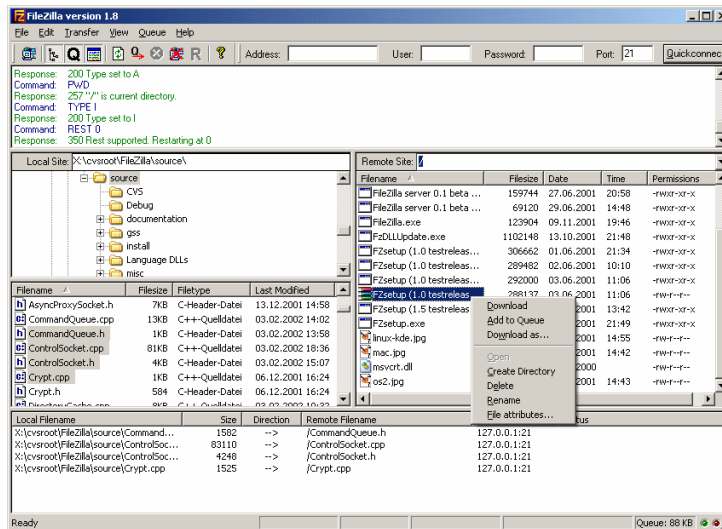


Figura 4: Ventana del programa filezilla

4. Archivos y directorios

4.1. Nombres

Resulta muy importante organizar la información de una forma coherente. Esta tarea se realiza con la ayuda de archivos y directorios (en ocasiones denominados carpetas). Tanto los unos como los otros deben ser denominados con una etiqueta a la que llamamos nombre del archivo o del directorio. Este nombre debe estar compuesto por letras y número, tantos como se considere necesario. En los nombres pueden utilizarse caracteres especiales como el punto, que dependiendo de su posición tiene distinto significado (por ejemplo, si el nombre del objeto empieza por punto es equivalente a decir que se trata de un objeto oculto). Nombres validos de archivos o ficheros son:

```
datos,           Datos,           .datos,
datos.txt,       datos.dir,       datos.csv,
datos.dat,       datos.Sierra_España.dat,  datos-muestreo.dat,
muestreo.dat,...
```

En GNU/LINUX se distingue entre mayúsculas y minúsculas. No es necesario (ni el sistema no tiene por que hacerlo en caso de que lo olvidemos) añadir una extensión a los nombres, esta sólo tiene una misión informativa; en algunos casos resulta conveniente para el propio usuario. Por ejemplo la extensión puede referirse a la versión del archivo `datos.v1`, `datos.v2`, `datos.v3`, ... Esto nos permite disponer de distintas versiones del mismo archivo.

En general deben evitarse nombres con espacios en blanco ("`Muestras de Burete.dat`") o con signos de puntuación ("`definitivo!.dat`"). Los espacios en blanco pueden simularse con distintos caracteres, aunque es preferible el guión bajo: "`Muestras_de.Burete.dat`".

4.2. Archivos

Los archivos contiene información en forma de caracteres. En ellos aparece reflejada la información de diversas maneras: al estilo de una matriz de datos (`.dat`), como un texto (`.txt`), como una relación de items, separados por comas (`.csv`), instrucciones de trabajo para el sistema (`.sh`), ... Estos son los ficheros denominados `ascii` que pueden ser manipulados por diversos programas. Para ver su contenido pueden utilizarse diversos programas (`cat`, `more` o `less`) sin más que escribir el nombre del programa indicando a continuación el nombre del fichero (por ejemplo: `less datos.txt`, pulsando `Q` para terminar).

Existen otro conjunto de ficheros, a los que se denomina "binarios", que contienen códigos especiales que nos impiden su visualización directa (imágenes (`.jpg`, `.tif`, `.dvi`) que exigen un programa que interprete adecuadamente el contenido del archivo. También puede darse el caso de archivos comprimidos (`.gz`, `.zip`, `.tgz`), que sustituyen a los originales para ahorrar espacio en el disco. En otros casos los ficheros son generados

por un programa específico y que presenta un formato (modo en que se codifica la información) específico (hojas de cálculo, programas de dibujo, bases de datos, ...).

Para conocer las características de un fichero se utiliza la orden `file` que produce, por ejemplo:

```
palazon@fobos:~$ file *
l_1369_1.pdf:          PDF document, version 1.3
azor.xyz.dat:          ASCII text
beameruserguide.html: HTML document text
caldum.presentacion.tex: ISO-8859 text
caldum.tex:            LaTeX 2e document text
caratula3.pnm:         Netpbm PPM "rawbits" image data
dinpob.r:              ISO-8859 English text
ET.sh:                 ASCII text
expendenso.svg:        XML document text
impreso.txt:           ISO-8859 English text
matrix_xp.bootskin:    Zip archive data, at least v2.0 to extract
mdeespana.tif:         TIFF image data, little-endian
nohup.out:             writable, regular file, no read permission
NOTAS ECOLOGIA.doc:    Microsoft Office Document
pp.jpg:                JPEG image data, JFIF standard 1.01
pp.ps:                 PostScript document text conforming at level 3.0
pp.tgz:                gzip compressed data, from Unix
primilla.r.png:        PNG image data, 1249 x 912, 16-bit/color RGB, non-interlaced
www.iso:               ISO 9660 CD-ROM filesystem data 'CDROM'
xv_3.10a-26_i386.deb:  Debian binary package (format 2.0), uses gzip compression
```

4.3. Creación y edición de ficheros

Los ficheros en formato ASCII pueden ser modificados con programas de edición más o menos sencillos: desde nano, a vim o emacs, denominados generalmente editores. Las diferencias se basan en sus prestaciones y su complejidad: a mayores prestaciones mayor complejidad. Además de sus prestaciones la elección de un editor debe basarse en su accesibilidad: no podemos utilizar un editor gráfico (gvim o xedit) en una conexión en modo texto. Una buena elección es utilizar, al menos inicialmente, un editor sencillo como nano:

```
palazon@fobos:~$ nano mifichero
```

que proporciona como resultado en la terminal, tal como se refleja en la figura 5. Se muestra información sobre el fichero (en este caso vacío) y sobre las opciones del usuario: por ejemplo, `^X` indica que pulsando `CTL+X` saldremos del programa.

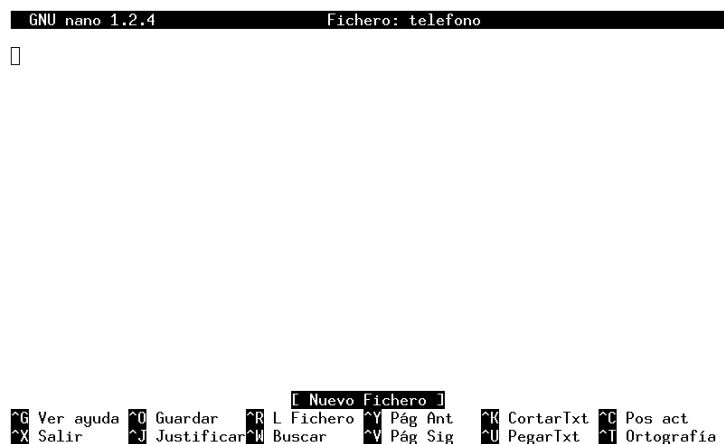


Figura 5: Ventana de edición del programa nano

La aparente sencillez de nano no debe confundirse con su potencia ya que puede editar archivos de más de 50 MB.

Los archivos pueden borrarse con la orden `rm`. **Los ficheros borrados ¡no pueden recuperarse!**

Por ejemplo: `rm mifichero documento7.txt nidos.dat`; permite borrar tres archivos de una vez.

¡Ojo! `rm *` borrar todos los archivos de una vez.

Para comprobar el contenido de un archivo puede utilizarse la orden `cat`. Por ejemplo: `cat datos.txt`. En el caso de archivos muy largos se consigue una presentación adecuada con `more` o `less`.

Para cambiar el nombre a un archivo se utiliza la orden `mv` (`mv viejo.nombre nuevo.nombre`).

4.4. Directorios

Los archivos se organizan en unos archivos especiales denominados directorios cuyo contenido son archivos y otros directorios. La organización de archivos en un sistema GNU/LINUX está muy clara. Todos los usuarios disponen de un directorio personal denominado *home directory*, que le “pertenece” y en el que pueden escribir y borrar archivos y directorios sin más limitación que la capacidad del disco. Este directorio aparece habitualmente como `/home/login`, puede determinarse fácilmente sin más que escribir `pwd` al inicio de la sesión. Como puede apreciarse entre el nombre del archivo y el de los directorios aparece un separador: “/”. Al iniciar una sesión en el servidor la terminal elige como directorio de trabajo nuestro *home directory*. Todos los directorios se organizan a partir de un directorio raíz “/”, de forma jerárquica se organizan subdirectorios, subdirectorios de estos ...

Para realizar distintas tareas rutinarias relacionadas con directorios se utilizan las siguientes ordenes:

- Crear un directorio: **mkdir**. Ej: `mkdir gis`
- Activar un directorio como directorio de trabajo: **cd**. Ej: `cd gis`
- Borrar un directorio (debe estar vacío): **rmdir**. Ej: `rmdir marmenor`
- Volver al *home directory*: **cd**. Ej: `cd`
- Volver al directorio anteriormente activado: **cd -**. Ej: `cd -`

Puede esquematizarse una estructura de directorios mediante la orden `tree`:

```
japf@fobos:~$ tree
.
|-- gis
    |-- burete
    |   |-- localizacion.casas
    |-- marmenor
        |-- localizacion.casas
```

3 directories, 2 files

Considerando este ejemplo, para movernos por la estructura de directorios utilizamos nombres con el nombre absoluto nombre del directorio completo y nombre del archivo: `/home/japf/gis/burete/localizacion.casas`





Puede utilizarse una forma relativa refiriendonos a subdirectorios que esten por debajo del directorio activo, por ejemplo, el usuario antonio puede desde el directorio activo `/home/antonio` hablar de `gis/burete/localizacion.casas` para referirse al fichero indicado antes.


Si por contra su directorio activo es `/home/antonio/gis/marmenor` puede utilizar un nombre relativo al directorio padre (..) mediante `../burete/localizacion.casas`

Para cambiar un archivo de directorio se utiliza la orden `mv`; indicando uno o varios archivos que serán trasladados al directorio que se indique al final. Así, por ejemplo, `mv mifichero documento7.txt nidos.dat palelera`; llevaria los archivos de datos no deseados a un directorio que llamamos `palelera`.












4.5. mc: la navaja suiza

Puede utilizarse una herramienta muy sencilla para modificar archivos y directorios copiando, borrando o desplazando archivos y directorios de un sitio a otro. Se trata del programa mc cuyo aspecto puede verse en la figura 6.

El programa divide la pantalla en varias regiones, que de arriba abajo se corresponden con un menú, dos columnas que muestran cada una el contenido de un directorio, una línea de comandos y un guía de las teclas de función. Los directorios son navegables —utilizando las teclas:    — e independientes, permiten por ejemplo copiar un archivo con facilidad de uno a otro sin más que ponerse sobre el nombre del archivo deseado y pulsar la tecla de copia.

Se puede utilizar en la mayoría de las terminales de texto (por ej: putty). En algunos casos las teclas de función “no funcionan” entonces puede utilizarse la secuencia  seguida de 1, 2, ..., 0.

La tecla de tabulación se emplea para cambiar el directorio activo (izquierdo o derecho).

-  Ayuda Sobre el funcionamiento del programa.
-  Menú de acceso a un menú de usuario.
-  Ver el contenido de un archivo (siguiendo sus características o en bruto).
-  Editar un fichero con un editor propio o elegido por el usuario.
-  Copiar los ficheros marcados a otro directorio, por defecto el indicado por el otro directorio. Para marcar se utiliza la tecla .
-  RenMov permite cambiar a otro directorio o el nombre de los archivos seleccionados.
-  Mkdir para crear un nuevo directorio.
-  Borrar un archivo o directorio: **¡Borrar = perder para siempre!**
-  Menú
 - Izquierdo características para la presentación de la información del directorio izquierdo (forma de ordenar los archivos, ver el árbol de directorios, mostrar directorios remotos via ftp, ...)
 - Archivo para realizar acciones sobre archivos.
 - Utilidades comparación de directorios, tamaños de los directorios, búsqueda de ficheros, ...
 - Opciones configurar qué y como mostrar, redefinir teclas, ...
 - Derecho equivalente a izquierdo para el directorio derecho.
-  Terminar salir del programa.

```

Izquierdo Archivo Utilidades Opciones Derecho
< /disco/et/japf/gis/marmenor v> <- /var v>
  Nombre Tamaño FechaMod Nombre Tamaño FechaMod
  .. DIR-ANI .. DIR-ANI
localiza~n.casas 0 18 abr 19:01
/backups 4096 18 abr 06:29
/cache 4096 18 abr 10:26
/games 4096 3 feb 12:54
/lib 4096 18 abr 10:26
/local 4096 26 jul 2004
/lock 4096 17 abr 21:13
/log 4096 18 abr 10:26
/mail 4096 14 abr 13:32
/opt 4096 6 oct 2004
/run 4096 18 abr 10:27
/scrollkeeper 4096 10 ene 10:41
/spool 4096 16 feb 09:20
/tmp 4096 18 abr 17:54
/www 4096 15 mar 12:59
  .. ..
Ayudita: Para seleccionar directorios añadir "/" tras el patrón a buscar.
palazon@fobos:/disco/et/japf/gis/marmenor$ 
1 Ayuda 2 Menú 3 Ver 4 Editar 5 Copiar 6 RenMov 7 Mkdir 8 Borrar 9 Menú 10 Salir

```

Figura 6: Aspecto del programa mc