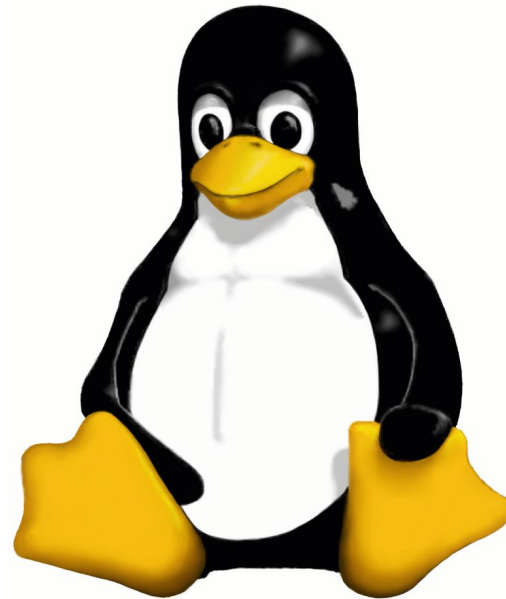


CURSO LINUX: Administración de Sistema y Servicios

(parte 2)



Sistemas de ficheros en Unix

- En un sistema Unix todo son archivos: memoria física, ratón, modems, teclado...
- Filosofía de diseño de gran éxito y potencia, aunque también peligrosa: un simple error de permisos puede permitir modificar todo un disco duro.
- Tipos de archivos:
 - ficheros planos
 - directorios
 - ficheros especiales (dispositivos)
 - orientados a carácter
 - orientados a bloque

Comandos para búsqueda de archivos

- `find` (busca ficheros en directorios)
- `locate` (busca patrones en bases de datos)
- `whereis` (busca el binario, el fuente y la página de ayuda de un fichero)
- `which` (busca comandos en el PATH)

Características de usuarios Unix

- Los sistemas Unix son sistemas multiusuario. Cada usuario tiene una serie de características propias y asociadas:
 - `uid`: identificativo de usuario (debe ser único)
 - `gid`: identificativo de grupo
 - `home`: carpeta de trabajo o personal
 - `shell`: interprete de comandos
- Comandos:
 - `who/whoami/id` (muestra información de usuarios)
 - `su/sudo` (cambia de usuario o privilegios)

Gestión de Usuarios

- `adduser` (crear usuarios)
- `usermod` (modificar usuarios)
- `deluser` (eliminar usuarios)

Gestión de Grupos

- `addgroup` (añade grupo o usuario a grupo)
- `groupmod` (modifica grupo)
- `delgroup` (elimina grupo o usuario de grupo)

Gestión de Usuarios

```
linux:~$ adduser alumno
```

```
linux:~$ addgroup consultec
```

```
linux:~$ adduser --home /home/consultec/alumno --shell  
/bin/sh --uid 5001 --ingroup consultec alumno
```

```
linux:~$ usermod --shell /bin/bash alumno
```

```
linux:~$ adduser alumno softwarelibre
```

- Esto añade a "alumno" al grupo "softwarelibre"

Cambiando el propietario y grupo

- Para cambiar el propietario o grupo puede hacerse a través del nombre o del identificativo (uid o gid).
 - `chown` (cambiar el propietario)
 - `chmod` (cambiar el grupo)



Cambiando el propietario y grupo

```
linux:~$ ls -l ~/documentos/linux.pdf
-rw-rw-r-- 1 alumno alumno 2684 ene  2 21:09 linux.pdf

linux:~$ chown alumno:consultec ~/documentos/linux.pdf

linux:~$ ls -l ~/documentos/linux.pdf
-rw-rw-r-- 1 alumno consultec 2684 ene  2 21:09 linux.pdf

linux:~$ id aktor
uid=500(aktor) gid=500(aktor) grupos=500(aktor)

linux:~$ chown 500 ~/documentos/linux.pdf

linux:~$ ls -l ~/documentos/linux.pdf
-rw-rw-r-- 1 aktor consultec 2684 ene  2 21:09 linux.pdf
```

Administración de permisos de ficheros

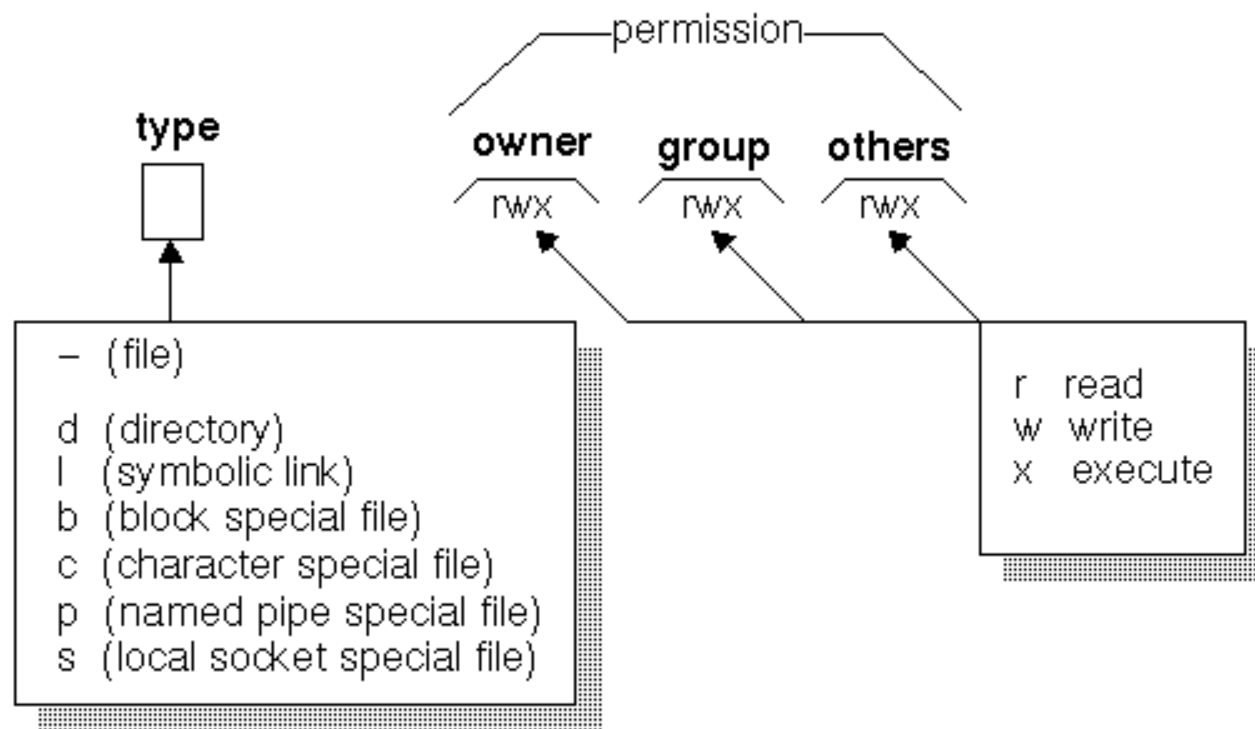
Permisos de ficheros en sistemas Unix

- Todos los archivos en Unix tienen permisos que indican que y quien puede hacer o no hacer una acción con el archivo.
- Es la base de la seguridad de Unix.
- 2 formas de notación:
 - Modo alfabético
 - Modo octal

Administración de permisos de ficheros

Permisos de ficheros en sistemas Unix

- Modo alfabético



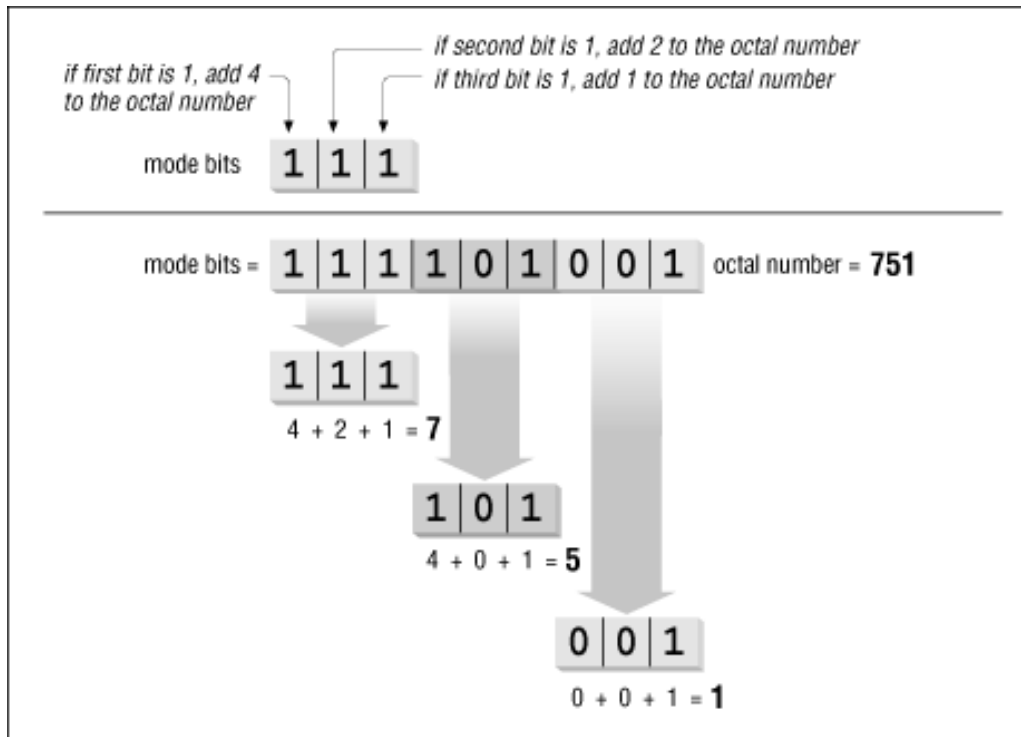
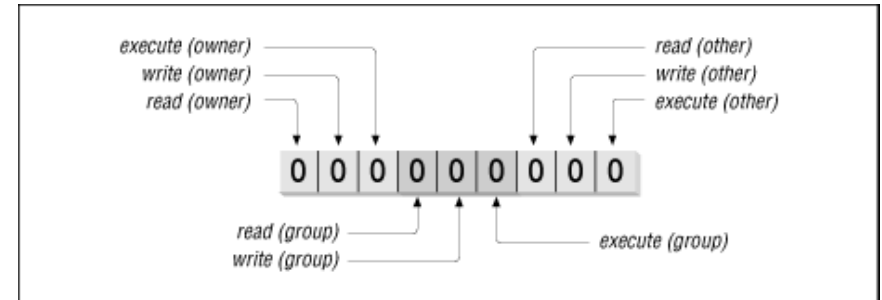
ZK-0536U-R



Administración de permisos de ficheros

Permisos de ficheros en sistemas Unix

- Modo octal



Administración de permisos de ficheros

Cambiando los permisos

- Solo es posible alterar los permisos si el usuario que lo está haciendo es el **propietario** del archivo.
 - `chmod [ugoa][+-][rwx] archivo`
 - u (propietario), g (grupo), o (resto), a (todos)
 - + / -: añadir / eliminar permisos
 - r (lectura), w (escritura), x (ejecución)
 - `chmod [modo_octal] archivo`

Procesos en Unix

- Los sistemas Unix son sistemas multitarea. Se pueden estar ejecutando distintas tareas a la vez y cada tarea es un proceso.
- El kernel es el encargado de asignarle una parte de la memoria para su ejecución.
- Los procesos tienen un dueño y vienen definidos por sus permisos.

Procesos en Unix

- Se caracterizan por:
 - `pid`: número identificador de proceso (único)
 - `ppid`: nº de identificador del proceso padre
 - `user`: dueño del proceso

Trabajando con procesos

- Control de procesos
 - `ps` / `pstree` (resumen de los procesos activos)
 - `top` (muestra las tareas de linux)
- Señales entre procesos
 - `kill` / `killall` (termina o “mata” procesos)
- Control de trabajos
 - `jobs` (visualiza los trabajos en segundo plano)
 - `bg` / `fg` (envía / trae procesos a segundo plano)
 - `&` (ejecución en segundo plano)

Sistemas de archivos locales

- Forma en que el SO organiza, gestiona y mantiene la jerarquía de archivos y directorios en un dispositivo de almacenamiento.
- Linux soporta gran cantidad de sistema de ficheros:
 - Orientados a disco
 - Nativos
 - `ext, ext2, ext3, reiserfs, xfs, jfs`
 - Externos
 - `msdos, vfat, ntfs, iso9660, ufs`
 - Distribuidos
 - `nfs, smb, codafs`

Ficheros de configuración (/etc/)

- Inicio del sistema
 - `hostname` (nombre de la máquina)
 - `inittab` (procesos de arranque y niveles de ejecución)
 - `fstab` / `mtab` (información sobre sistemas de ficheros)
 - `rc.d` (servicios en el arrancar y parar del sistema)

Ficheros de configuración (/etc/) (II)

- Usuarios
 - `passwd` / `group` (usuarios y grupos del sistema)
 - `shadow` (contraseñas cifradas)
 - `sudoers` (lista de que usuarios que pueden ejecutar qué)

Ficheros de comunicación con el kernel (/proc/)

- Es un pseudo sistema de ficheros que se usa como interfaz para hablar con el kernel
 - `cpuinfo` (procesador del sistema)
 - `meminfo` (memoria libre y usada del sistema)
 - `loadavg` (carga del sistema)
 - `filesystems` (sistemas de ficheros soportados por el kernel)
 - `pci` (listado de los dispositivos pci)
 - `sys` (permite modificar variables del sistema)
 - `uptime` (tiempo de funcionamiento)
 - `version` (versión del sistema)



Comandos para búsqueda de archivos (II)

```
linux:~$ find / -name passwd
```

```
linux:~$ locate halt
```

```
linux:~$ whereis cat adduser
```

```
linux:~$ which grep
```

```
linux:~$ whereis cd
```

```
linux:~$ find ~/ -type f -atime +5 -exec rm {} \;
```

Comandos para filtrado de archivos

- `grep` (muestra líneas coincidentes)
- `sed` (editor de streams)
- `cut` (corta texto)
- `tr` (traduce o elimina caracteres)

Comandos para filtrado de archivos (II)

```
linux:~$ grep -ri nameserver /etc/
```

```
linux:~$ grep -c bash /etc/passwd
```

```
linux:~$ sed "s/root/rat/g" /etc/passwd
```

```
linux:~$ cut -f1,6 -d: /etc/passwd
```

```
linux:~$ echo traduccion | tr [a-z] [A-Z]
```

Otros comandos interesantes

- `wc / sort` (cuenta / ordena palabras)
- `df / du` (muestra la ocupación del sistema de ficheros)
- `tar / gunzip / bzip2` (empaqueta / comprime ficheros)
- `history` (muestra el historial del intérprete de comandos)

Otros comandos interesantes

```
linux:~$ df -h
```

```
linux:~$ du -sh /var/log
```

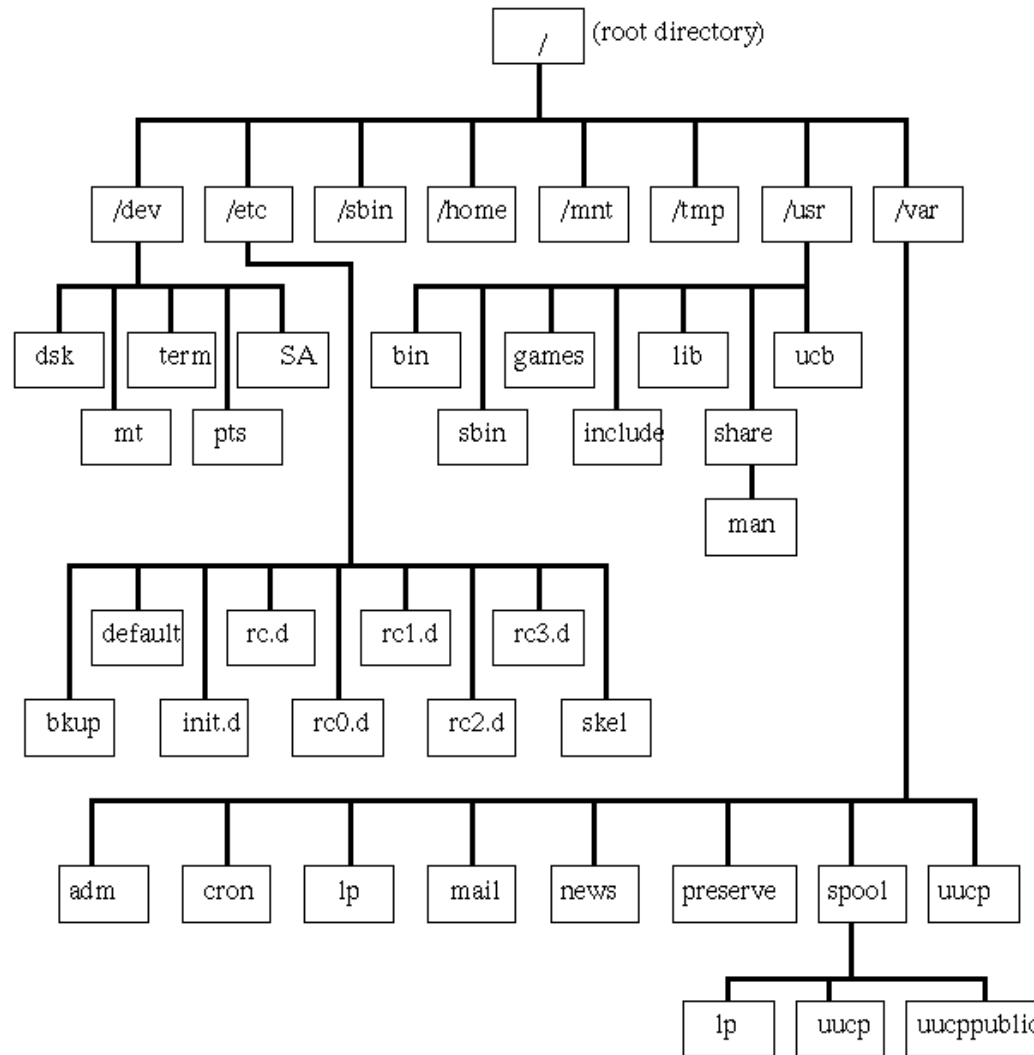
```
linux:~$ tar cvzf /var/log/backup.tgz /etc/ /home/
```

```
linux:~$ history
```

```
linux:~$ sort -r /etc/group
```

```
linux:~$ wc -l /etc/passwd
```

Desplazándose por el sistema de ficheros



Rutas en el sistema de ficheros

- **Absolutas:** desde el directorio raíz hasta el archivo deseado.
- **Relativas:** desde el directorio actual hasta el archivo deseado.

Caracteres especiales

- *: sustituye una cantidad cualquiera de caracteres
- ?: sustituye exactamente un carácter
- []: sustituye un carácter de los indicados

Expansión de comandos y archivos

Comandos para el manejo del sistema de ficheros

- `ls` (lista contenido de directorios)
- `cd` / `pwd` (cambia /muestra la ruta de directorio)
- `touch` (crea fichero vacío o actualiza existente)
- `cp` / `mv` / `rm` (copia / mueve/ elimina fichero)
- `mkdir` / `rmdir` (crea / elimina directorio)
- `man` / `info` (ayuda sobre comandos)
- `echo` (muestra una línea de texto)
- `date` / `cal` (muestra la hora del sistema)
- `file` (muestra el tipo de fichero)
- `halt` / `reboot` (apaga el sistema)

Comandos para el manejo del sistema de ficheros (II)

```
linux:~$ ls -lha /
```

```
linux:~$ pwd
```

```
linux:~$ cd ../../etc
```

```
linux:~$ touch /home/linux/fichero
```

```
linux:~$ date +%d/%m/%y
```

```
linux:~$ cp -a /var/log /backup/
```

```
linux:~$ mkdir -p /tmp/prueba/creacion/directorio
```

```
linux:~$ rm -rf /home/linux/carpeta/
```

Estructura de los Sistemas de Ficheros

- Estándar de la Jerarquía de Ficheros (FHS)
 - Se encarga de definir los directorios principales y sus contenidos en Linux y otros sistemas Unix.
 - Mayor facilidad en el uso de distintos sistemas Unix.

Estructura de los Sistemas de Ficheros (II)

- `/bin/ /sbin/`: comandos esenciales del sistema
- `/boot/`: kernel y archivos de arranque
- `/dev/`: dispositivos esenciales
- `/etc/`: archivos de configuración globales
- `/home/`: directorios personales de usuarios
- `/lib/`: librerías para comandos esenciales
- `/mnt/ /media/`: puntos de montaje
- `/usr/`: ficheros “compartidos” no esenciales
- `/var/`: ficheros variables frecuentemente
- `/tmp/`: ficheros temporales

Comandos para paginar y visualizar archivos

- `cat` (muestra el contenido de ficheros por pantalla)
- `more` / `less` (pagina el contenido de ficheros)
- `head` / `tail` (muestra las primeras / últimas líneas de un fichero)

Comandos para paginar y visualizar archivos (II)

```
linux:~$ cat -n /etc/passwd
```

```
linux:~$ more /etc/fstab
```

```
linux:~$ less /etc/passwd
```

```
linux:~$ cat /etc/passwd /etc/group
```

```
linux:~$ tail -f /var/log/syslog
```

```
linux:~$ head -n 1 /etc/passwd
```

Diferencia Linux <-> Windows

- En Windows los directorios se montan sobre unidades físicas:
 - `C:\Carpeta`
 - `D:\Drivers\vga`
- En Linux los dispositivos se montan en directorios de la estructura de ficheros:
 - `/home/usuario`
 - `/media/cdrom`

Dispositivos y “devices” asociados

- Disco duro IDE y CDROM: /dev/hdXY
 - X: Número de disco/dispositivo_IDE (a, b, c...)
 - Y: Número de partición (1, 2, 3...)

/dev/hda
/dev/hda1
/dev/hdb3
- Disco duro SCSI, SATA y externos USB: /dev/sdXY
 - /dev/sda
/dev/sda1
/dev/sdc4



Particiones primarias y lógicas

- Máximo 4 primarias:

`/dev/hda1` `/dev/hda2` `/dev/hda3` `/dev/hda4`

- Una de ellas se puede establecer como partición extendida:

`/dev/hda4`

- Se pueden crear varias particiones lógicas en una extendida:

`/dev/hda5` `/dev/hda6` ...

- Linux puede arrancar desde una partición lógica, no así Windows que requiere una primaria.

/etc/fstab

```
# /etc/fstab: static file system information.
#
# <file system> <mount point> <type> <options> <dump> <pass>
proc /proc proc defaults 0 0

/dev/hda1 / ext3 defaults,errors=remount-ro 0 1
/dev/hda3 /home ext3 defaults 0 2
/dev/hda2 none swap sw 0 0

/dev/hdb1 /media/hdb1 ext3 defaults 0 1

/dev/hdc /media/cdrom0 udf,iso9660 user,noauto 0 0
```

- La opción “user” permite que un usuario normal (no root) monte la partición.

mount

- Para montar dispositivos manualmente.
- Si el dispositivo existe en fstab:

```
mount punto_de_montaje
ó
mount /dev/dispositivo
```

→ Ej:

```
mount /dev/hdc = mount /media/cdrom0
```

- Si el dispositivo no existe en fstab:

```
mount -t TIPO_SIS_FICHEROS /dev/dispositivo punto_montaje
```

→ Ej:

```
mount -t iso9660 /dev/hdc /media/cdrom0
mount -t ext3 /dev/hdb1 /media/disco_2
mount -t vfat /dev/hda2 /mnt/windowsXP
```

mount (II)

- Notas:
 - ➔ Una partición de tipo VFAT no dispone de permisos ni dueños/grupos así que no se puede usar el comando “chown” ni “chmod” para permitir a usuarios la escritura.
 - La partición por defecto es del usuario que la monta (la puede montar cualquiera si tiene la opción “user”).
 - Se puede fijar el dueño de la partición con la opción “uid”, el grupo con “gid” y los permisos con “umask” (ojo, “umask” es la máscara invertida).
- !! man mount !!

umount

- Desmonta un dispositivo:

```
umount /dev/dispositivo
```

ó

```
umount punto_de_montaje
```

- ¡ Sólo si el dispositivo no está usado por ningún proceso!
 - Si un usuario tiene una terminal en /media/cdrom no se podrá desmontar el dispositivo.
 - Para comprobar qué procesos están usando un dispositivo:

```
lsuf punto_de_montaje
```

Ej:

```
lsuf /media/cdrom
```

Montaje de dispositivos

pmount / pumount

- Igual que mount/umount pero no requiere que exista el punto de montaje.
 - Facilidad para manejar dispositivos extraíbles (pendrives, cds, tarjetas...).
 - Permite el montaje a usuarios normales.

SSH

- telnet seguro al puerto 22 -> datos cifrados.

```
ssh usuario@servidor  
# password: ...
```

- Eventos al iniciar conexión SSH:
 - El cliente comprueba la clave que le ofrece el servidor (la busca por IP en su fichero `.ssh/knownhosts`).
 - ii Si cambia la clave del servidor el cliente SSH nos avisará y tendremos que borrar la línea correspondiente en `.ssh/knownhosts` !!
 - Se establece una clave simétrica de sesión para los datos.
 - Se produce la autenticación del cliente.
 - Usuario/password o por clave pública.
 - Se establece la conexión y el cliente accede al servidor.

Servidor SSH

- Fichero `/etc/ssh/sshd_config`
 - Port 22 -> Puerto en el que escucha
 - PermitRootLogin yes -> ¿Permitimos login de root vía SSH?
 - AllowUsers user1 user2 -> Usuarios permitidos
 - X11Forwarding yes -> ¿Permitimos reenvío de las X?
- `ii man sshd_config !!`
- http://unixhelp.ed.ac.uk/CGI/man-cgi?sshd_config+5

Autenticación por clave pública

- Proceso en el cliente:

→ Genera su clave (sin frase de paso):

```
ssh-keygen -t dsa
```

Ha generado:

- `.ssh/id_dsa.pub` (clave pública)
- `.ssh/id_dsa` (clave privada)

→ Copia su clave pública al servidor:

```
ssh-copy-id -i .ssh/id_dsa.pub usuario@servidor
```

Su clave pública se copia en `.ssh/authorized_keys` de `/home/usuario` en el servidor.

→ Ahora se puede loguear sin introducir el password (ideal para tareas periódicas de backups).

Autenticación por clave pública (II)

- Proceso en el servidor:
 - En `/etc/ssh/sshd_config`:
 - Habilitar autenticación por clave pública:
`PubkeyAuthentication yes`
 - Opcionalmente puede impedir la autenticación por password:
`PasswordAuthentication no`

SCP

- Comando para copiar ficheros sobre SSH:
 - Copiar “fichero” en la home de “usuario” en servidor:
`scp fichero usuario@servidor:`
 - Copiar un directorio recursivamente en /tmp del servidor:
`scp -r dir usuario@servidor:/tmp`
 - Copiar fichero2 desde el servidor a mi home:
`scp usuario@servidor:fichero2 ~`
 - Copiar de máquina B a máquina C:
`scp usuarioB@servidorB:fichero usuarioC@servidorC:`

- Este documento está protegido bajo la licencia Reconocimiento-Compartir-Igual 2.5 España de Creative Common.

<http://creativecommons.org/licenses/by-sa/2.5/es/>

Copyright © 2006 Irontec <contacto@irontec.com>
 Se permite la copia, distribución, uso comercial y realización de la obra, siempre y cuando se reconozca la autoría de la misma, a no sea ser que se obtenga permiso expreso del autor.

