

Servicios Telemáticos

Ingeniería de Telecomunicación

Departamento de Ingeniería Telemática

Curso 2020/2021

Tema/Práctica 00 – 1 Clase

Conceptos y Administración de SSOO. Administración básica de Linux

Javier Muñoz Calle

P00: Objetivos de la práctica

■ Administración básica de Linux:



- Opcional** {
 - Documentación de Apoyo Linux (Opcional, Muy Recomendable) ➡
 - Anexo (Informativo):
 - Compilación Kernel, Arranque, Instalación de software
- Práctica** {
 - Administración Local S.O. Linux □ FP I (1º) ➡
 - Uso del shell
 - Comandos de administración del equipo
 - Administración de la Red □ Fundamentos Internet (2º) ➡
 - Configuración de las tarjetas de red
 - Comandos de red
 - Analizadores de red



P00: Documentación de apoyo Linux

□ Abarca:

- Recordatorio asignaturas anteriores
- Práctica P00
- Herramientas adicionales



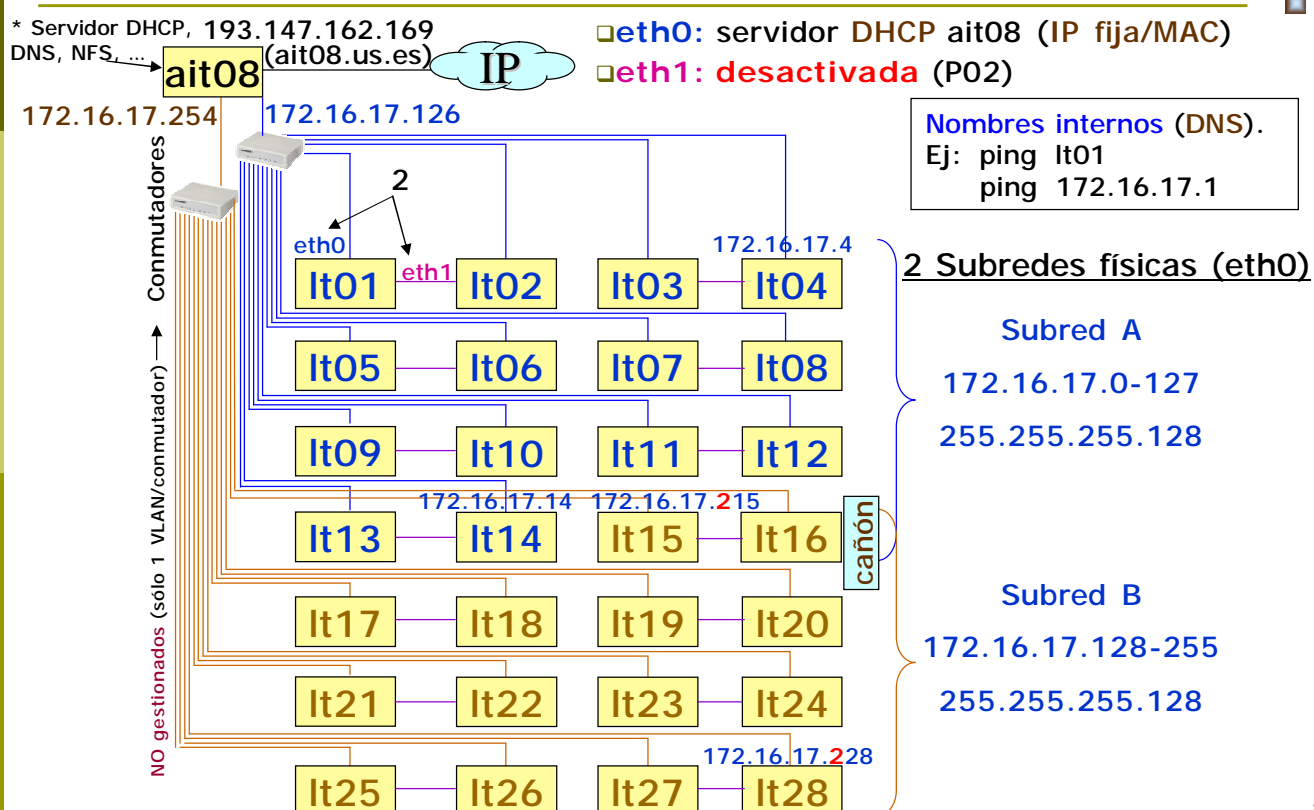
□ *Opcional*, pero **Muy Recomendable**:

- Ayuda a Práctica P00.
- Y a mejorar el rendimiento de trabajo en entornos Linux (útil en el resto de prácticas, otras asignaturas, TFG, ...).

Documentos de apoyo (P00)	Administracion_Linux_Local.pdf	Recomendable: Lectura y Realización previa a la Práctica P00
	Administracion_Linux_Red.pdf	
	Programacion_POSIX_Shell_Script_Linux.pdf	
	Anexo-Linux_Configuracion_Red.pdf	

3

Entorno de trabajo: Red



4

P00: Administración local. Shell: Uso básico

Administración servicios: Modo consola (muchos sin X)

Seguridad, Recursos, Innecesario, ...



- "history" de comandos: **Cursores**  


```
...  
[root@lt30 ~] service ...  
[root@lt30 ~]
```

- Scroll: **May-RePag**, **May-AvPag**



- Aucompletado (comandos/ficheros): **Tab** 


Comandos: primera palabra

```
[dit@lt30 ~] su    
su sudo sum suspend
```

```
[dit@lt30 ~] sus   
↓  
[dit@lt30 ~] suspend
```

Ficheros/Directorios: argumento ≥ 2

```
[dit@lt30 ~] cd /b    
bin/ boot/
```

```
[dit@lt30 ~] cd /bo   
↓  
[dit@lt30 ~] cd /boot
```


- Existen paquetes (como "bash_completion") que añaden al shell funcionalidades avanzadas de autocompletado, incluyendo opciones de comandos (e.g. "service xxx ...").

5

P00: Shell: Variables de entorno

- **Variables** de entorno (**Case Sensitive** "var" ≠ "vAr"): modifican el comportamiento del shell.

```
...  
[dit@lt30 ~] ls
```

HOME	Carpeta personal del usuario actual
 PATH	Ubicación de aplicaciones ⇨
LANG	Lenguaje sistema (lenguajes instalados: "locale -a")
SHELL	Shell actualmente usado
PWD	Directorio actual
...	...

```
[dit@lt30 /] ls /usr/bin/  
... emacs ... man ... mc ...  
[dit@lt30 /] cd /  
[dit@lt30 ~] /usr/bin/emacs  
[dit@lt30 ~] emacs
```

```
[dit@lt30 ~] echo $PATH  
/bin:/usr/bin:/sbin/usr:/sbin:./
```

6

P00: S.O. Linux: Búsqueda de ejecutables

F

□ Búsqueda de aplicaciones usando:

■ Variable PATH:

```
[dit@lt30 ~] echo $PATH  
/bin:/usr/bin:/sbin/usr:/sbin/
```

□ Autocompletado.

```
[dit@lt30 ~] sus  
[dit@lt30 ~] suspend
```

□ Comando "which": busca una aplicación en directorios de PATH

```
[dit@lt30 ~] which su  
/usr/bin/su
```

■ Rutas propias:

□ Comando "whereis" (NO usa PATH): busca aplicación (binarios, fuentes y man) en ubicaciones estándar.

```
[dit@lt30 ~] whereis su  
su: /usr/bin/su  
/usr/share/man/man1/su.1.gz
```

□ Comando "find"

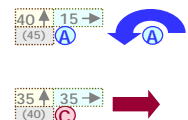
```
[dit@lt30 ~] find / -name su  
/usr/bin/su  
/usr/share/locale/su  
...
```

/mnt/servicios/P00/P00inicio.sh (como "root")

Apartado: 2

Recordar: Linux

Login	Password
dit	dit
root	root



P00: Administración local Linux: Comandos

F

□ Fundamentos de Programación I (1º)

□ Comandos de usuario:

■ Redirecciones: "<", ">", ">>", ">&" (2>&1), "|", ...

■ Comandos básicos: ls, cd, rm, mkdir, mv, ...

■ Ayuda: man [x] comando/fichero (Espacio, q, /cadena+INTRO, n)

Búsqueda de cadenas

□ Comandos de administración:

■ Usuarios: passwd, su [-], ...

■ Sistema de ficheros: ln, chmod, chown, ...

■ Procesos: ps, kill, pkill, pidof, poweroff, ...

■ Ficheros de texto: dos2unix, cat, ...

■ Compresión: tar, gzip, ...

■ Dispositivos: mount, umount

■ Acceso remoto: ssh, sftp, ...

P00: S.O. Linux: Usuarios. Definición

Usuarios definidos en el sistema:

Clave en `/etc/shadow` Grupo principal \$HOME `/etc/passwd` (legible por todos)

login:x:uid:gid:descripción:carpeta_personal:shell

Ejemplo:

`root:x:0:0:root:/root:/bin/bash`
`dit:x:501:501:lt:/home/dit:/bin/bash`
`luis:x:502:500:LUIS:/home/luis:/bin/sh`
 ...

login:
`[dit@lt28 ~]`

`/etc/shadow`

Clave shadow	\$1\$...	\$5\$...	\$6\$...
Cifrado (crypt)	MD5	SHA-256	SHA-512
Nº caracteres	22	43	86

- En `/etc/passwd`: se define el grupo principal al que siempre pertenece cada usuario.
- En `/etc/group`: adicionalmente, se puede asociar dicho usuario a otros grupos.

Grupos definidos (Grupo = conjunto de usuarios):

nombregrupo:x:gid:user1,user2,userN `/etc/group` (legible por todos)

`gdit:x:500:dit,luis`

`/etc/gshadow`

Ejemplo:

`file dit gdit -rw-r-----`
 Usuario Grupo u g o

P00: S.O. Linux: Usuarios, Permisos

Visualización de permisos:

"ls -l -n": UID/GID numéricos (sin traducirlos a nombres con `/etc/passwd` y `/etc/group`).

Buscar ficheros con: `find / -uid xxx [-gid yyy]`

Usuario/Grupo NO existentes en (borrados) `/etc/passwd` y `/etc/group`, respectivamente

Permisos Propietario Grupo Bytes `ls -l` (o "ll")

total	20								
-rwxrwxr-x	1	dit	dit	5224	Dec 30	03:22	hello		
-rw-r-----	1	dit	amigos	221	Dec 30	03:59	hello.c		
-rw-rw-r--	1	dit	root	1514	Dec 30	03:59	hello.s		
drwxrwxr-x	7	501	1000	1024	Dec 31	14:52	posixuft		

`-rwxrwxrwx`

Permisos Otros

Permisos Grupo

Permisos Propietario

Flag Directorio (d=directorio; l=link; -=fichero)

Nº de bloques (tamaño habitual de 512 bytes, pero cambia según sistema) reservados en el disco (se reservan bloques="cajas", usualmente de 4 en 4, aunque los datos no los llenen) por los ficheros listados: `ls -ls` (bloques/fichero), `ls -lh` (tamaños en múltiplos)

Fecha de creación / Última modificación del contenido

Lectura (r):	4
Escritura (w):	2
Ejecución (x):	1

Cambios de permisos y propietarios:

`chmod 755 file`

Owner=rwx, Group=r-x, Other=r-x

`chmod -R 755 dir`

Imprimir permisos en formato numérico (octal):

`stat -c "%a %A %U:%G %n" /dir/file`

`chown usuario[:grupo] file` # Owner=usuario, Group=grupo

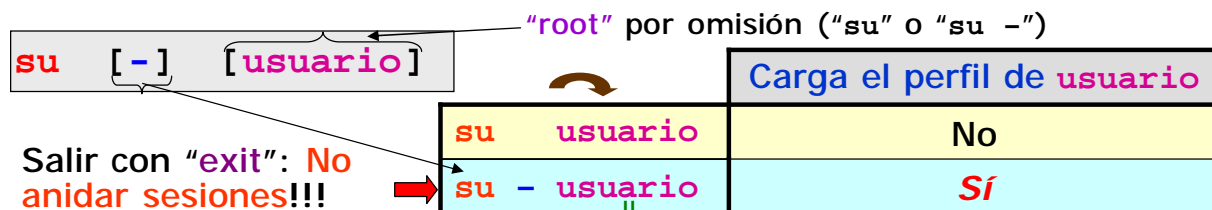
`chown -R usuario[:grupo] dir`



P00: S.O. Linux: Cambio de usuario

F

- **"su"**: sobre sesión actual, **nueva sesión** con usuario indicado.

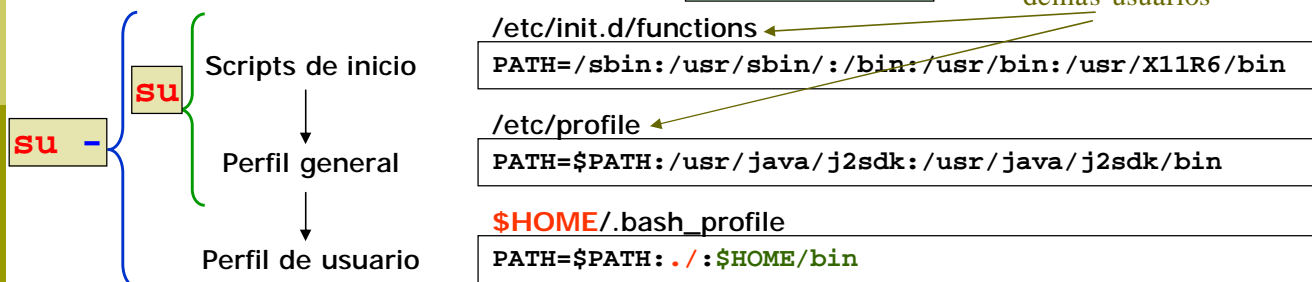


- Salir con **"exit"**: **No anidar sesiones!!!**

- MAL: su, su, su, su, ...
- OK: su, exit, su, exit, ...

login:
[dit@lt28 ~]

Varían del "root" a los demás usuarios



* **"sudo comando"**: permite que un **usuario** pueda **ejecutar** ciertos **comandos** (Ej: mount) **como otro usuario** (según IP), **según indique** **"/etc/sudoers"**

* Usar mejor: **su [-] [usuario] -c "comando"**

11

P00: S.O. Linux: Cambio de usuario (2)

F

- **Variables de entorno: distintas con "su" y "su -"**

```
[dit@lt30 ~] ls /home/dit/bin/
prog
[dit@lt30 ~] cd /; echo $PATH
/bin:/usr/bin:./:/home/dit/bin
[dit@lt30 ~] /home/dit/bin/prog
...Se ejecuta el comando...
[dit@lt30 ~] prog
...Se ejecuta el comando...
```

```
[root@lt30 ~] pwd
/root
[root@lt30 ~] echo $PATH
/usr/sbin:/sbin:/bin:./:/root/
[root@lt30 ~] su dit
[dit@lt30 root] pwd
/root
[dit@lt30 root] prog
-bash: prog_root: command not found
[dit@lt30 root] echo $PATH
/usr/sbin:/sbin:/bin:./:/root/
```

"su" mantiene perfil usuario anterior

```
[root@lt30 ~] pwd
/root
[root@lt30 ~] echo $PATH
/usr/sbin:/sbin:/bin:./:/root/
[root@lt30 ~] su - dit
[dit@lt30 ~] pwd
/home/dit
[dit@lt30 ~] prog
...Se ejecuta el comando...
[dit@lt30 ~] echo $PATH
/bin:/usr/bin:./:/home/dit/bin
```

"su -" carga perfil del usuario

12

P00: Recordatorio permisos

Para cada uno de los usuarios de un fichero se establecen los permisos:

Permiso	Fichero	Directorio
r	4	Lectura
w	2	Escritura
x	1	Ejecución
		(Permiso de búsqueda) Acceso al contenido de los ficheros, es el usado por el sistema para comprobar si un fichero existe, ...

Permisos mínimos para		
Leer el fichero	Modificar el contenido del fichero	Borrar el fichero
--x (1)	--x (1)	--x (1)
--x (1)	--x (1)	-wx (3)
r-- (4)	-w- (2)	-- (0)

Sin "w": No hace falta escribir en el disco (la información se queda)

13

Identificador de discos requerido para operar con ellos (formatear, analizar, ...)

Montaje discos fijos: `/etc/fstab` `/dev/sda1 /dir ext3...`
 Montaje de discos externos: Automontaje no deseable

P00: S.O. Linux: Acceso Dispositivos almacenamiento

Dispositivos de almacenamiento (Discos duros y CDs/DVDs)

IDE /PATA (2 buses)	Canales Primario	Maestro	sda
		Esclavo	sdb
	Secundario	Maestro	sdc
		Esclavo	sdd

SATA, SCSI, USB (lsusb)	sda
Interfaces USB	sdb
	...

Discos (no CD/DVDs) conectados al equipo **fdisk -l** (como root)

Soft Drive (en "kernels <2.6.20" eran "hdxx").

Numeración: en el orden de creación (NO de izquierda a derecha)

DISCO: `/dev/sda`

Extendida: `/dev/sda3`



HD PCs LTxx(MBR)

ext2 Linux (ext2fs)

Estilo de partición	Numeración
MBR	4 Primarias (Lógicas en Extendida)
GPT (G/UUID UEFI)	1, 2, 3, ...



P00: S.O. Linux: Enlaces

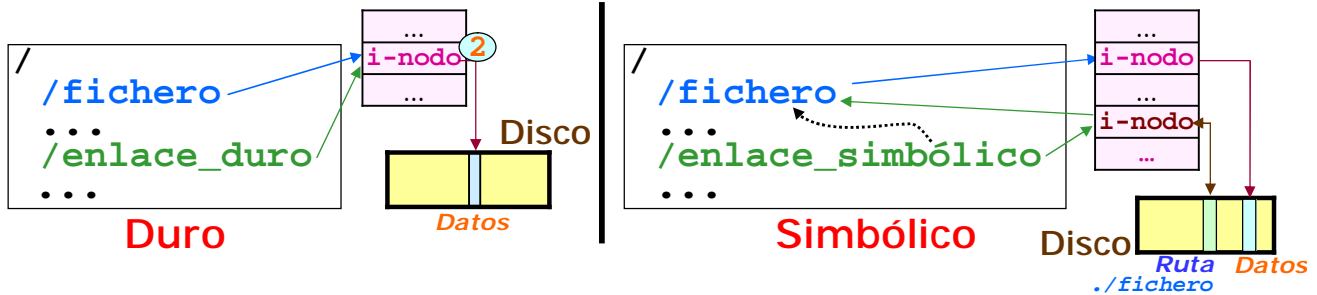
□ SS00 (3°)

F

□ UNIX permite **enlace** a ficheros/directorios.

□ **Duro**: `ln fichero enlace_duro`

□ **Simbólico**: `ln -s fichero enlace_simbólico`



```
[root@lt28 ~] ls -l -i
```

```
39321 -rw-r--r-- 2 dit dit 22 Mar 9 12:56 fich
39321 -rw-r--r-- 2 dit dit 22 Mar 9 12:56 enl_duro
```

```
[root@lt28 ~] ls -l -i
```

```
9261 -rw-r--r-- 1 dit dit 22 Mar 9 12:56 fich
9262 lrwxrwxrwx 1 root root 6 Mar 9 12:59 enl_sim -> ./fich
```

Sólo el usuario propietario puede modificar/borrar el enlace simbólico

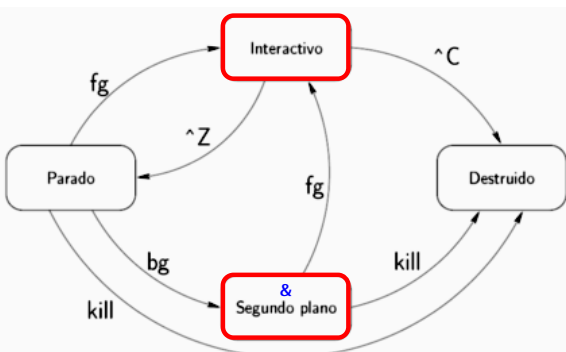
	Afecta a
chmod enl_sim	Fichero apuntado
chown enl_sim	Fichero apuntado
chown -h enl_sim	Enlace simbólico

P00: S.O. Linux: Procesos

□ FP I (1°)

F

ps, pstree	Procesos en ejecución (en memoria)
pidof	PID (número de proceso) del proceso indicado
kill	Eliminar un proceso a partir de su PID
pkill	Eliminar procesos con expresión regular
bg, jobs	Trabajos background (&), parados (Ctrl-Z), recién terminados
fg / bg	Trabajo a Primer plano / Background



```
[dit@lt30 ~] kill 3720
[dit@lt30 ~] kill -9 3720
[dit@lt30 ~] pkill -9 ba
[dit@lt30 ~] firefox &
[dit@lt30 ~] jobs
[1]+ Ejecutando firefox &
[dit@lt30 ~] fg 1
firefox
```

□ `15 = TERM` (default)
□ `"9" = KILL` (no bloqueable)

Nº trabajo (≠ PID)

[Ctrl-C]



P00: S.O. Linux: Procesos (2), Señales

Señales a procesos (man 7 signal): **kill -N/señal proceso**

Señales para terminar procesos (-a + Bruscas)		Significado habitual (según aplicación capture)	Capturable / Ignorable
Nº	Nombre señal		
2	SIGINT, INT (Ctrl-C)	Parar y esperar input (suele "=SIGTERM" en programas no interactivos)	Si
15	SIGTERM, TERM (default) ←	Terminar Limpiamente (suavemente, ordenadamente, liberando recursos, ...)	
1	SIGHUP, HUP	<ul style="list-style-type: none"> Automáticamente enviada a Aplicaciones terminal cuando el usuario se desconecta: =SIGTERM En Demonios (servicios sin interfaz): Recargar configuración 	
3	SIGQUIT, QUIT	Terminar Inmediatamente (sin salvar estado...)	No
9	SIGKILL, KILL ←		

RECORDAR (Funcionamiento de señales): cuando se envía una señal a un proceso (i.e. TERM), la gestiona:

- Por defecto, el SO (NO el proceso): e.g. con TERM cierra el proceso cuando termina las actuales escrituras....
- Si el desarrollador del proceso ha programado la captura (trap) de esa señal, entonces la señal la gestiona el proceso (NO el SO): el comportamiento del proceso al recibir la señal será el programado.

• Ejemplo: el comando "trap" permite definir capturas de señales para el shell actual.

17

Usuarios/Grupos de un proceso:

- Reales: propietarios
- Efectivos: con los que opera (crea ficheros/procesos con ellos, ...).

En general, ambos "= efectivos del padre", pero "root" puede usar otros o cambiarlos.

□ SS00 (3º)

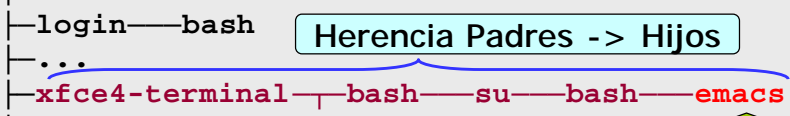
P00: S.O. Linux: Procesos (3)

Ⓕ

□ Características de cada proceso:

- Nombre proceso (pskill, ...) = Nombre del fichero Ejecutable asociado.
 - Pertenece a un Usuario/grupo Efectivos (EUSER/EGROUP o EUID/EGID).
 - ➡ ■ En general, Usuario/grupo proceso Hijo = Usuario/grupo proceso Padre.
 - Ej: [root@lt30 ~] emacs
 - Puede: Proceso padre con "root" => Proceso hijo con otro usuario.
- Ej: Apache se arranca como "root" y abre hijos con usuario "apache".

□ Visualización detallada de procesos:

[root@lt30 ~]# pstree		and emd/systemd ...
<pre>systemd├─... │├─login───bash │├─... │├─xfce4-terminal├─bash──su───bash───emacs │└─... └─...</pre> <div>Herencia Padres -> Hijos</div> 		
4470	pts/4 00:00:00 dit grupo firefox /usr/lib64/firefox/firefox	
...		

Herencia Padres -> Hijos

18

P00: S.O. Linux: Procesos (4)

- Lista de procesos en memoria:**
 - "**ps**": Procesos pertenecientes al **usuario actual** (mismo EUID) y en la **terminal actual** (tty/x, pts/x) desde la que se ejecuta el comando.
 - "**ps ax**": Lista **TOTAL** de **procesos** del sistema (de todos los usuarios)
- Información de cada Proceso en memoria:** **/proc/PID/**

/proc/PID/	Contenido (ver con "cat")
/cmdline	Argumentos con que se invocó al proceso
/exe	Enlace al ejecutable del proceso
/stat	Estado del proceso (formato resumido)
/status	Estado detallado del proceso (Nombre proceso , PID , estado de ejecución Sleeping/Running , ID usuario , memoria , ...): ps aux
...	[dit@lt30 ~] cat /proc/3720/cmdline bash

```
[dit@lt30 ~] ps
PID  TTY      TIME CMD
3720 pts/1    00:00:00 bash
4408 pts/1    00:00:00 ps
```

```
[dit@lt30 ~] ps ax
PID  TTY      STAT    TIME COMMAND
1    ?        Ss      0:02 /usr/lib/syst...
2    ?        S        0:00 [kthreadd]
3    ?        S        0:00 [ksoftirqd/0]
```

P00: S.O. Linux: IDs de Procesos (5)

ID	Descripción
PID	ID único del Proceso (Process ID)
RUID / RGID	ID Usuario/Grupo Reales : propietarios (usuarios que pueden matar el proceso , pararlo , ...). En general, " = efectivos del padre ", pero " root " puede usar otros o cambiarlos. [RUSER / RGROUP : Nombre Usuario/Grupo Reales]
EUID / EGID	UID Usuario/Grupo Efectivos : con los que opera (crea ficheros/procesos con ellos, ...). En general, " = efectivos del padre ", pero " root " puede usar otros o cambiarlos. [EUSER / EGROUP : Nombre Usuario/Grupo Efectivos]
PPID	PID del Proceso Padre o invocador (Parent Process ID)
PGID	PID del Primer proceso del Grupo de procesos (comando compuesto en shell) con el que se ha ejecutado este proceso (Process Group ID o Process Group Leader)
SID	PID del Primer proceso de la Sesión de shell (Session ID) desde la que se ha arrancado el proceso (= PID del shell invocador)
SPID (LWP)	ID único de los Hilos de un proceso. SPID (operating System Process Identifier), LWP (Light Weight Process)

P00: S.O. Linux: IDs de Procesos (6)

```
[root@lt201 ~]# ps -Ao pid,ruid,ruser,rgid,rgroup,euid,euser,egid,egroup,command |
grep -E "bash|COMMAND"
  PID  RUID  RUSER      RGID  RGROUP    EUID  EUSER    EGID  EGROUP  COMMAND
  6627    0   root         0    root         0   root         0   root    -bash
  7116    0   root         0    root         0   root         0   root    grep -E httpd...
```

Hereda (arrow from parent to child)

Padre (parent process)

Hijo (child process)

Usuario/Grupo Reales (Real User/Group)

Usuario/Grupo Efectivos (Effective User/Group)

```
[root@lt201 ~]# ps -Ao pid,ruid,ruser,rgid,rgroup,euid,euser,egid,egroup,command |
grep -E "httpd|COMMAND"
  PID  RUID  RUSER      RGID  RGROUP    EUID  EUSER    EGID  EGROUP  COMMAND
  7098    0   root         0    root         0   root         0   root    /usr/sbin/httpd
  7099   48  apache      48  apache      48  apache     48  apache    /usr/sbin/httpd
  7100   48  apache      48  apache      48  apache     48  apache    /usr/sbin/httpd
  7101   48  apache      48  apache      48  apache     48  apache    /usr/sbin/httpd
  7102   48  apache      48  apache      48  apache     48  apache    /usr/sbin/httpd
  7103   48  apache      48  apache      48  apache     48  apache    /usr/sbin/httpd
  7116    0   root         0    root         0   root         0   root    grep -E httpd...
```

Cambia (arrow from parent to child)

Padre (parent process)

Hijos (children processes)

Usuario/Grupo Reales (Real User/Group)

Usuario/Grupo Efectivos (Effective User/Group)

```
[root@lt201 ~]# ps
  PID TTY          TIME CMD
  6627 pts/3        00:00:00 bash
  7043 pts/3        00:00:00 ps
```

Shell y Proceso Padre invocador (arrow from parent to child)

```
[root@lt201 ~]# echo $$
6627
```

Primer proceso del Grupo (comando compuesto)

```
[root@lt201 ~]# ps -Ao pid,ppid,pgid,sid,command | grep -E "COMMAND|6627"
  PID  PPID  PGID  SID  COMMAND
  6627   6625 6627 6627 -bash
  7038   6627 7038 6627 ps -Ao pid,ppid,pgid,sid,command
  7039   6627 7038 6627 grep --color=auto -E COMMAND|6627
```

Proceso Padre (invocador) (arrow from parent to child)

Primer proceso de la Sesión = Shell

□ SS00 (3°)

P00: S.O. Linux: Procesos Hijos vs Hilos (7)

□ Proceso: programa en ejecución.

- El SO le asigna sus **Recursos**: CPU, Memoria, Archivos, E/S
- Al arrancarse, siempre posee **1 hilo** de ejecución (monohilo).

□ **Thread o Hilo** (proceso ligero o subprocesso): entidad de ejecución.

□ Para Procesamiento en paralelo (varias tareas a la vez), un proceso puede crear:

- **Procesos Hijos**: cada uno controla sus recursos. Llamadas al SO usadas:
 - Padre, `fork()`: crea proceso hijo copia del padre (ambos siguen tras esta línea).
 - Padre o hijo, `exec()`: cesa programa actual y carga otro en memoria.
 - Comunicación entre procesos (IPC): precisa mecanismos del SO (tuberías "c1|c2", ...)
- **Más Hilos** (multihilo): cuando termina el último hilo, se cierra el proceso
 - **Comparten**: Espacio direcciones (Memoria), Variables globales, Ficheros
 - **Independiente**: Contador de programa (posición ejecución), Pila llamadas (Stack), Estado (ejecución, preparado o bloqueado)

```
if ( fork() == 0 )
/* proceso hijo */
else
/* proceso padre */
```

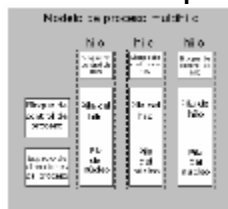
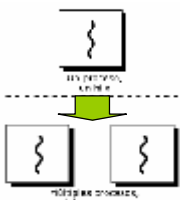
Hilos

Ventajas:
Eficiencia. Más rápida:

- Comunicación entre hilos (misma zona de memoria)
- Conmutación entre hilos
- Creación y Cierre hilos

Inconvenientes

- Sin protección entre hilos



P00: S.O. Linux: Procesos Hijos (8)

F

Procesos Hijos:

PID_Padre = 3013

```
[root@lt01 ~]# ps --forest -o pid=,tty=,stat=,time=,cmd= -g $(ps -o sid= -p 3013)
3013 pts/0    Ss      00:00:00 bash
4936 pts/0    S       00:00:00  \_ su -
4939 pts/0    S+      00:00:00      \_ -bash
5215 pts/0    S       00:00:00          \_ xclock
```

Herencia + Datos

```
[root@lt201 ~]# pstree 3013
bash--su--bash--xclock
```

Herencia

```
[root@lt01 ~]# ps -H -g 3013 -o comm
COMMAND
bash
  su
    bash
      xclock
```

Herencia

```
[root@lt01 ~]# pgrep -P 3013
4936
```

Hijos (PID)

```
[root@lt01 ~]# ps --ppid 3013
  PID TTY          TIME CMD
 4936 pts/0    00:00:00 su
```

Hijos (PID + Datos)

P00: S.O. Linux: Hilos (9)

F

Hijos (Threads) de un Proceso: identificados con un

■ **SPID** (operating System Process IDentifier) = LWP (Light Weight Process)

```
[root@lt201 ~]# pstree
systemd--...
  xfce4-terminal--...
    bash--su--bash--xclock
    gnome-pty-helpe
    2*[xfce4-terminal]
```

Hilos (aparte del principal) entre corchetes Y LLAVES

```
[root@lt201 ~]# pidof xfce4-terminal
3009 → PID_Proceso = 3009
```

```
[root@lt201 ~]# ps -T 3009
  PID SPID TTY      STAT   TIME COMMAND
 3009 3009 ?        Sl      0:13 /usr/bin/xfce4-terminal
 3009 3010 ?        Sl      0:00 /usr/bin/xfce4-terminal
 3009 3014 ?        Sl      0:00 /usr/bin/xfce4-terminal
```

Hilo principal

Otros Hilos

PID Proceso **SPID Hilos**

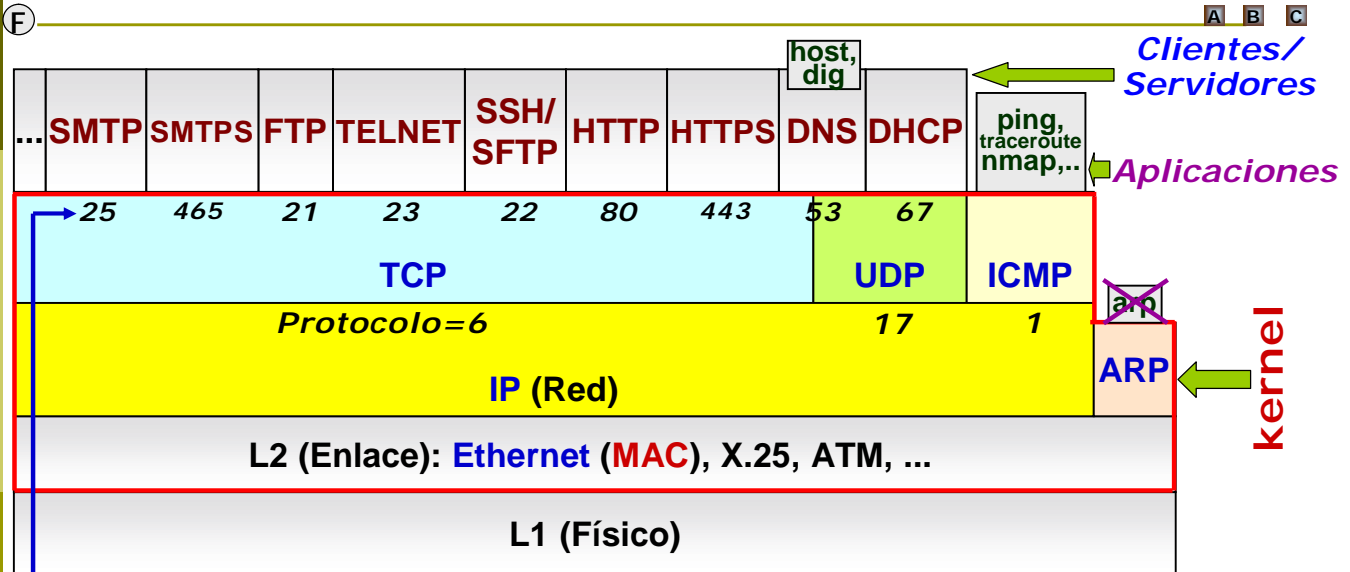
```
[root@lt201 ~]# top -H -p 3009
top - 11:56:56 up 1 day, 22:58, 10 users, load average: 0.00, 0.01, 0.05
Threads: 3 total, 0 running, 3 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 8097040 total, 7048460 free, 237564 used, 811016 buff/cache
KiB Swap: 8191992 total, 8191992 free, 0 used. 7578800 avail Mem
```

Pulsando "H" se cambia entre modo "Hilo" y Modo "Proceso". También "htop"

```
SPID Hilos
  PID USER  PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 3009 dit   20   0 541516 18576 12632 S  0.0  0.2  0:13.12 xfce4-terminal
 3010 dit   20   0 541516 18576 12632 S  0.0  0.2  0:00.00 gdbus
 3014 dit   20   0 541516 18576 12632 S  0.0  0.2  0:00.00 gmain
```

Hilos

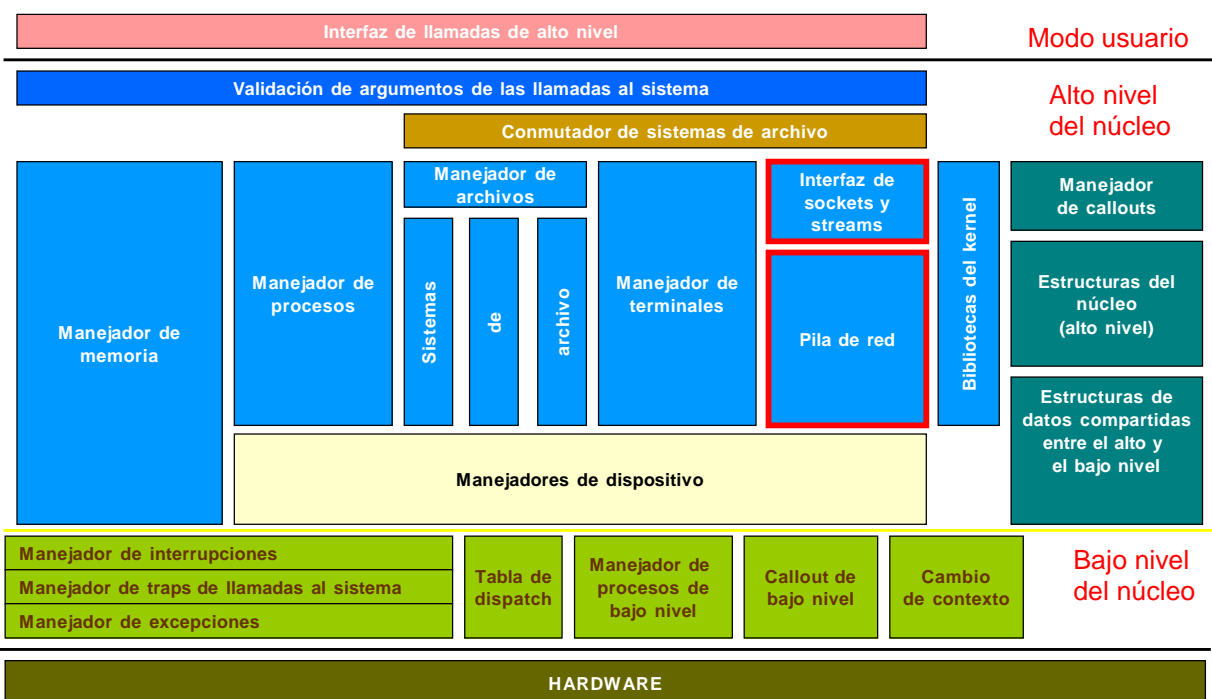
P00: Red: Torres de Protocolos



Asignación de puertos	Rango	Uso habitual*
De sistema (seguros, sólo "root")	0-1023	Servidores
De usuario	1024-65535	Clientes (firefox, ...)

* No obligatorio. e.g.: "68" Cliente DHCP; "8080" muchos Servidores web₂₅

P00: Kernel Linux

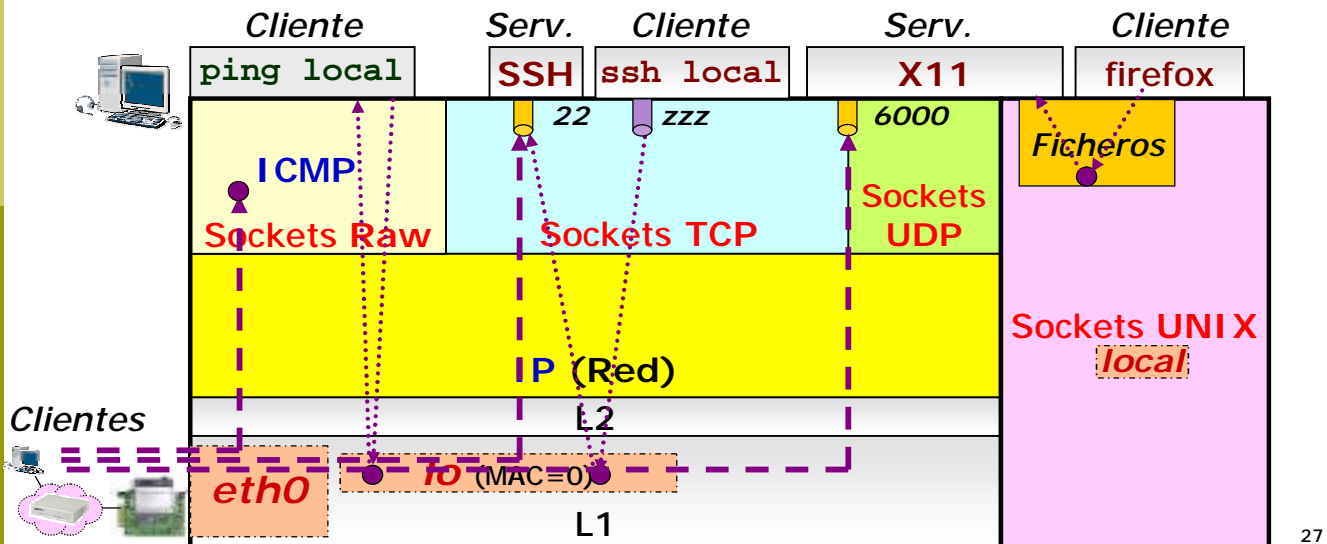


<http://www.kernel.org/>

P00: Red: Sockets. Tipos

□ Sockets en Linux (principales): man 7 socket

- Sockets de Red (Network Sockets): *IP...*
 - Sockets TCP, UDP, SCTP, ... (L4 sobre IP)
 - Socket RAW: acceso directo a IP (ICMP)
- Sockets UNIX: *no red (ficheros "virtuales") => No Wireshark*



27

Documentación de apoyo

Anexo-Linux_Configuracion_Red.pdf

Fundamentos Internet (2º)

P00: Red: Configuración de interfaces

□ En arranque: service network restart (o "ifup eth0")

■ Rama Fedora/CentOS:

```
/etc/sysconfig/network-scripts/ifcfg-eth0
```

DEVICE=eth0	Interfaz (no por el nombre del fichero)
ONBOOT=yes	Activar en el arranque ("yes" por omisión)
BOOTPROTO=dhcp	Configuración DHCP

```
/etc/sysconfig/network-scripts/ifcfg-eth1
```

DEVICE=eth1	Interfaz (no por el nombre del fichero)
ONBOOT=no	No activar en el arranque ("yes" por omisión)
BOOTPROTO=static	Configuración estática
IPADDR=172.16.17.228	IP interfaz eth1
NETMASK=255.255.255.128	Máscara
# NETWORK=172.16.17.128	IP de subred (omitible, se autocalcula)
# BROADCAST=172.16.17.255	IP de difusión (omitible, se autocalcula)
GATEWAY=172.16.17.126	Encaminador/Pasarela

■ Rama Debian/Ubuntu: /etc/network/interfaces

```
[root@lt28 ~]# ip a show eth0
```

```
2: eth0: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 60:02:92:18:5c:a0 brd ff:ff:ff:ff:ff:ff
    inet 172.16.17.228/25 brd 172.16.17.255 scope global dynamic eth0
    valid_lft 52391sec preferred_lft 52391sec)
```

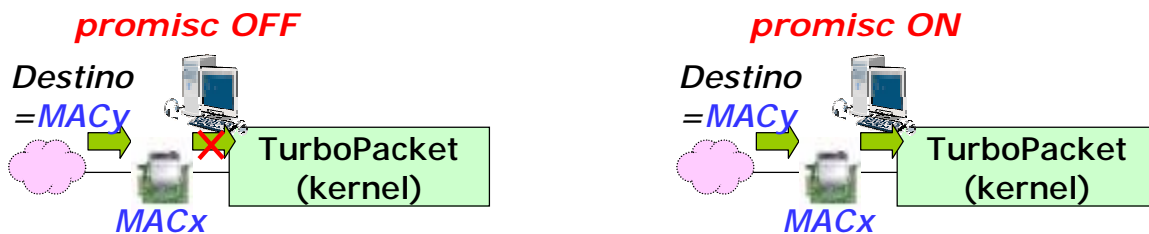
28

P00: Red: Analizadores

□ Analizadores de red en Linux:

Analizador (suelen requerir "root") ←	Modo	<i>Modo promiscuo</i>
<code>tcpdump</code>	Comando	Manual <code>ip link set ethX promisc on/off</code> ^A
<code>wireshark</code>	Gráfico	Automático

□ Modo promiscuo: capturar cualquier "MAC destino".



29

P00: Analizadores de Red/IDS: Acceso a los paquetes

□ Implementación de un Analizador de Red: Software+Hardware

■ Punto de acceso a la red (hardware): Tarjeta Red.

■ Sistema de captura de datos:

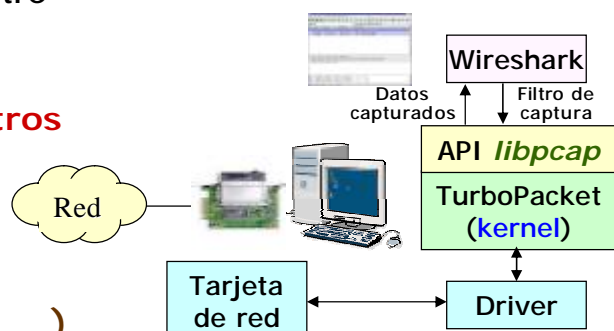
□ Módulo encargado de:

- Comunicarse a bajo nivel con el punto de acceso: **driver** de la tarjeta de red.
- **Captura de los paquetes**: el soporte ofrecido por el kernel del S.O (**TurboPacket**, buffer intermedio a alta velocidad).

□ Interfaz de comunicación (API) entre sistema de captura y software de usuario: librería **libpcap** (Linux; determina la **sintaxis de los filtros de capturas**).

■ Software de usuario o supervisión: analizadores (**Wireshark**).

□ Similar para IDSs (Snort, Suricata, ...).



30

P00: Red: Comandos

Fundamentos Internet (2º)

- Enlace (L2)**
- Protocolo **ARP**: Implementación
 - "ip neighbour" (o arp)
 - arping
- Red IP (L3)**
- Interfaces de red: Configuración
 - "ip address" e "ip link" (o ifconfig)
 - Tabla de reenvío: Construcción y Lectura
 - "ip route" (o route)
- Transporte (L4) y Aplicación (L5)**
- Sockets: Tipos
 - ss (o netstat) (locales), nmap (remotos)
 - ping (ICMP), traceroute (ICMP, UDP)
 - DNS: Resolución
 - host/nslookup, dig
- Analizadores de red: Wireshark, tcpdump

Obsoletos	arp, ifconfig, route	netstat
Sustitutos	ip (multinivel)	ss

31

En la tabla ARP puede haber **VARIAS** líneas:

- Para la misma IP: en distintas "Iface's".
- Con la misma MAC: con distintas IPs (e.g. Proxy ARP).

P00: Red: ARP, Comandos ip n/arp y arping

- Protocolo ARP: usado de forma transparente al usuario:**
- Kernel: gestiona Protocolo y caché ARP (añade/elimina entradas).
 - Cmd ip n(arp): sólo opera sobre caché ARP(NO sobre protocolo)
 - Comando arping: Sólo (pide kernel) envía ARP Request(NO caché)

```
[root@lt28 ~]# ip n arp -n
172.16.17.227 dev eth0 lladdr 00:13:F7:0A:17:DA REACHABLE
172.16.17.254 dev eth0 lladdr 00:13:F7:0A:18:8B STALE

[root@lt28 ~]# ip n show 172.16.17.226 cache arp 172.16.17.227
172.16.17.227 dev eth0 lladdr 00:13:F7:0A:17:DA REACHABLE

[root@lt28 ~]# ip n show 172.16.17.226 arp 172.16.17.226
172.16.17.226 dev eth0 lladdr 00:13:F7:0A:17:DA REACHABLE

[root@lt28 ~]# arping -I eth0 172.16.17.226
ARPING 172.16.17.226 from 172.16.17.228 eth0
Unicast reply from 172.16.17.226 [52:54:00:BF:D9:1E] 1.126ms
^CSent 1 probes (1 broadcast(s))
Received 1 response(s)

[root@lt28 ~]# ip n arp -n
172.16.17.227 dev eth0 lladdr 00:13:F7:0A:17:DA REACHABLE
172.16.17.254 dev eth0 lladdr 00:13:F7:0A:18:8B STALE
```

El kernel ha añadido la entrada a la caché ARP

X arping NO afecta a la caché ARP

P00: Protocolo ARP: Cabecera

IP (Red)	ARP
L2 Ethernet – MAC [Ethertype=2048 (IP), 2054 (ARP)]	
L1 (Físico)	

0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
Tipo dirección física (1=MAC Ethernet)																Tipo dirección de red (0x0800=IP)															
Nº bytes dir. física (6 para MAC)								Nº bytes dir. de red (4 para IPv4)								Tipo de mensaje (1=ARP Req.; 2=ARP Reply; 3=RARP Req; 4=RARP Reply)															
Dirección física del emisor: (6 bytes si MAC)																															
Dirección de red del emisor (4 bytes si IPv4)																															
Dirección física del destinatario (6 bytes si MAC)																															
Dirección de red del destinatario (4 bytes si IPv4)																															

También están en la cabecera Ethernet (salvo si difusión Ethernet)

Sólo en ARP Reply

ARP Request/Reply (RFC 826)

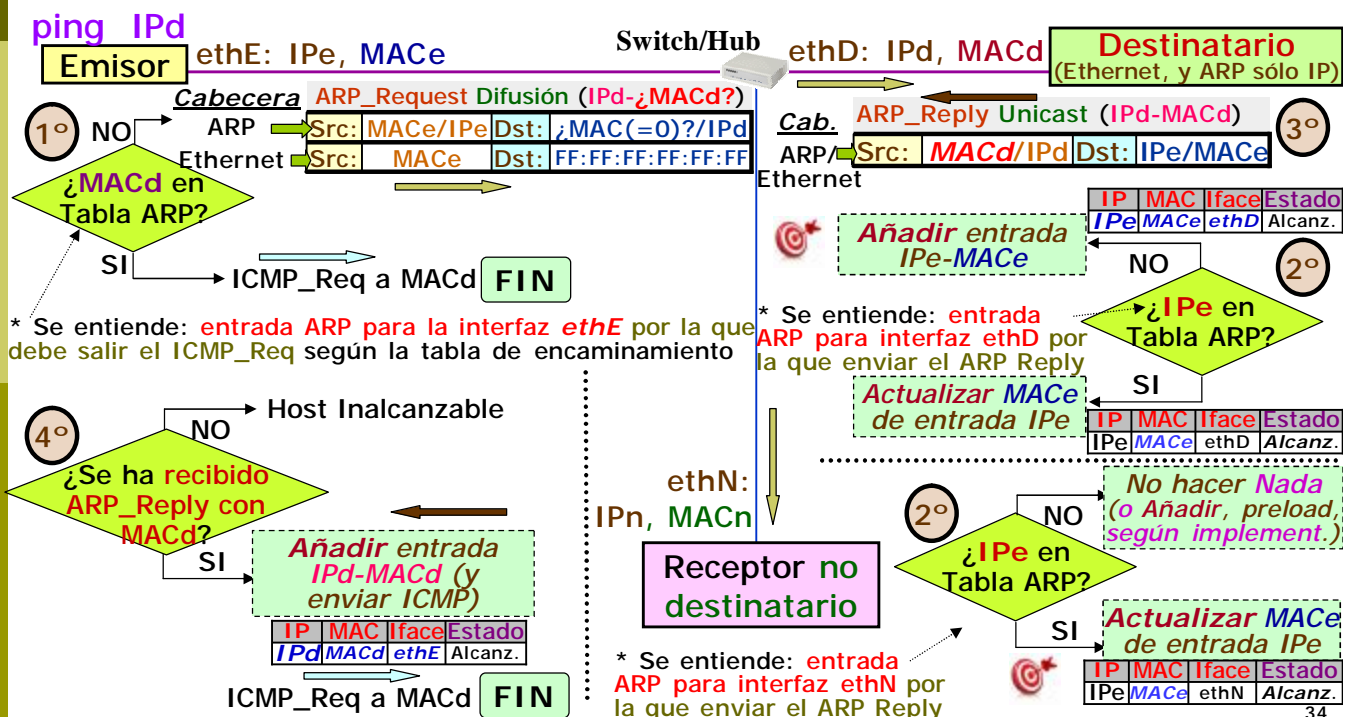
El ARP Reply se construye:

- 1º Usando las direcciones (MAC, IPv4) emisor del ARP Request ahora como destinatario.
- 2º Usando la dirección de red (IPv4) del destinatario del ARP Request ahora como emisor.
- 3º Incluyendo como dirección física (MAC) del emisor la dirección física (MAC) del equipo.

33

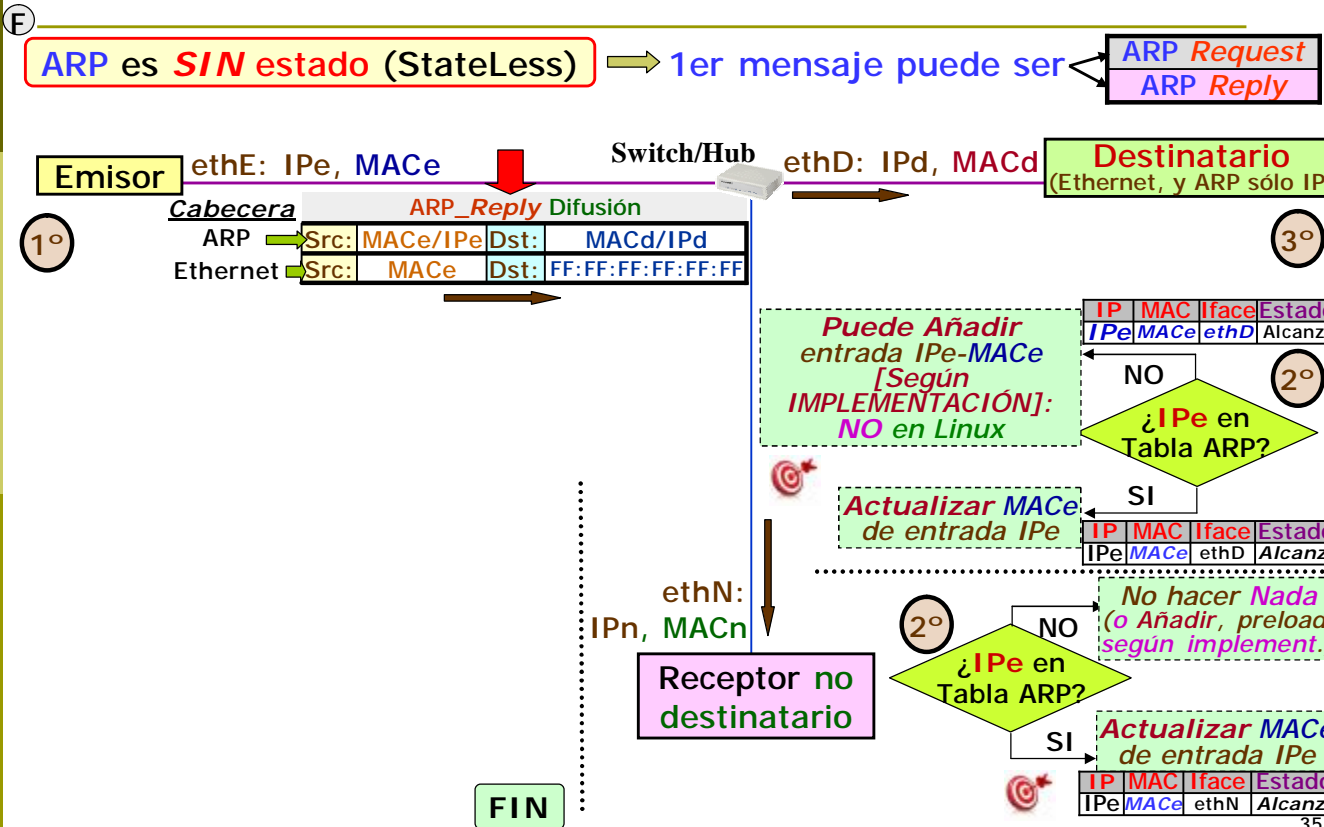
P00: Protocolo ARP: Aprendizaje MACs

ARP [RFC 826] (IPv4): IP -> ¿MAC? (Aprendizaje y Actualización de MACs)



34

P00: Protocolo ARP: Sin Estado



https://wiki.wireshark.org/Gratuitous_ARP

P00: Protocolo ARP: Usos

- F
- Principal (Básico) [RFC 826]: IP->¿MAC? (Comunicación L2).
 - Anuncios ARP (ARP Gratuitos): Informan (no preguntan) de MAC.

ARP	ARP Request	IP_Emisor = IP_Destinataro (= > No pregunta)
Gratuitos	ARP Reply	Enviado directamente, sin ARP Request previo

 - Movilidad IP [RFC 2020]: actualizar Nueva MAC (High-Availability*, e.g.Clusters)
 - Direcciones IPv4 Link-Local "169.254/16" [RFC 3927] (sin DHCP): Informan IP uso.
 - Detección de conflicto de Dirección IP (ACD)[RFC5227]: Informar Nueva MAC de IP
 - Sonda ARP (ARP Probe) [RFC 2131/4.4.1, RFC 5227]: Detección IP en uso (Duplicate Address Detection, DAD), no es ARP Gratuito (IP_Src ≠ IP_Dst).
- Sonda ARP** **ARP Request** **IP_Emisor = 0.0.0.0** (para que los equipos destino no añadan esta IP en su caché ARP; podría no ser finalmente configurada si está en uso)

Generación Manual:

- Sonda ARP Request: `arping -D -I ethX IP_a_comprobar`
 - No vale (va por "lo"): `ping -I ethX IP_ethX`
- ARP Reply Gratuito IP usada: `arping -A -I ethX IP_usada`

* Dos Routers R1,R2 con misma IP en backup (sólo uno activo). Cuando cae R1, entonces R2 manda Anuncio ARP a clientes para que actualicen IP-MAC1 a IP-MAC2

```

DEVICE=eth1 ifcfg-eth1
ONBOOT=no
BOOTPROTO=static
IPADDR=10.0.0.1
NETMASK=255.255.255.0

```

Asumiendo
10.0.0.1 usada
por otro equipo
de la red

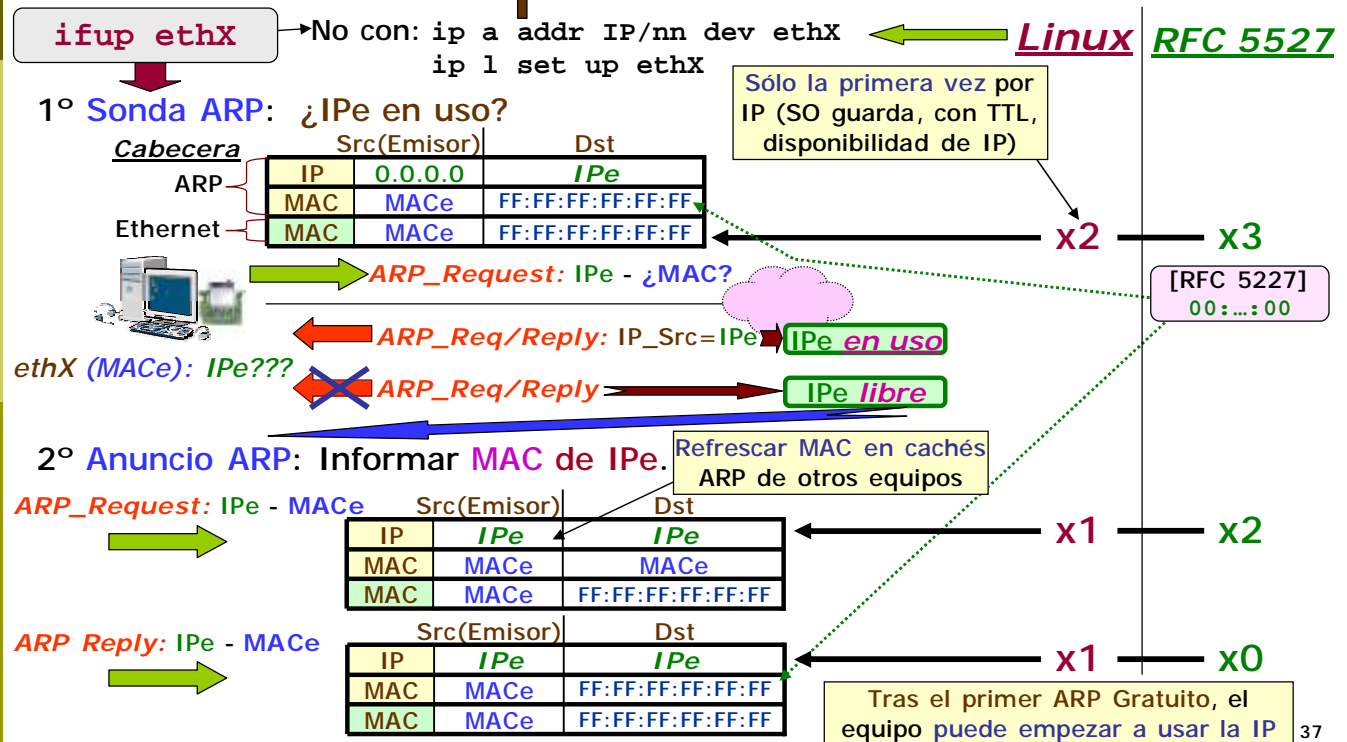
```

[root@lt28 ~]# ifup eth1
ERROR: some other host already uses address 10.0.0.1
[root@lt28 ~]# ip a add 10.0.0.1/24 dev eth1
[root@lt28 ~]# ip a list dev eth1
2: eth1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500...
    link/ether 68:05:ca:2a:03:59 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.1/24 scope global eth1 ...

```

P00: Protocolo ARP: Uso Activación de Interfaz

Uso ARP Activación interfaz: Diferencias entre RFC 5227 y SSOO Linux...



P00: Red: ip a (ifconfig) e Interfaz local

- 127.x.x.x (localhost): Referencia a la propia máquina
- No toda la comunicación local va por "lo": Sockets Unix
- Implementada con la interfaz virtual "lo".

```

[root@lt28 ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> m... ip a show up
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever

2: eth0: <BROADCAST,MULTICAST,PROMISC,UP,L...
    state UP qlen 1000
    link/ether 60:02:92:18:5c:a0 brd ff:ff:ff:ff:ff:ff
    inet 172.16.17.228/25 brd 172.16.17.255
        valid_lft 50218sec preferred_lft 50

3: eth1: <NO-CARRIER,BROADCAST,MULTICAST>
    DOWN qlen 1000
    link/ether 68:05:ca:29:62:6b brd ff:ff:ff:ff:ff:ff

```

ifconfig -a Todas las interfaces de red

ifconfig Interfaces **activas** (UP)

ping 127.0.0.1
ping 127.3.3.3
ping 127.16.17.1

Diagrama:

El diagrama muestra la interfaz local 'lo' (loopback) conectada a un archivo virtual (socket Unix). La interfaz 'lo' tiene la dirección IP 127.0.0.1. El archivo virtual (socket Unix) se utiliza para la comunicación local entre procesos (C y S) y se conecta a la interfaz 'eth0' (interfaz de red) a través de la IP 172.16.17.228.

P00: Red: Tabla de encaminamiento

- ❑ Construcción automática a partir de Configuración Tarjetas de red (automática, service network restart):
- ❑ "1" entrada Directa por interfaz
 - ❑ "1" Pasarela (de última interfaz)

Lectura en orden alfanumérico

```

/etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp

/etc/sysconfig/network-scripts/ifcfg-lo
DEVICE=lo
ONBOOT=yes
BOOTPROTO=static
IPADDR=127.0.0.1
NETMASK=255.0.0.0
NETWORK=127.0.0.0

/etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1
ONBOOT=no
BOOTPROTO=static
IPADDR=172.16.17.54
NETMASK=255.255.255.252
#NETWORK=172.16.17.52
#BROADCAST=172.16.17.55
GATEWAY=172.16.17.53
    
```

No se recoge en la tabla

default = 0.0.0.0 = Cualquier IP

No está

Subred: 172.16.17.52-55

Subred: 172.16.17.128-172.16.17.255

```

[root@lt28 ~]# ip n ← route -n
default via 172.16.17.53 dev eth1
172.16.17.52/30 dev eth1 proto kernel scope link src 172.16.17.54
172.16.17.128/25 dev eth0 proto kernel scope link src 172.16.17.1
    
```

39

- "ifdown/ifup" des/activan lo que indique en ese momento el fichero de configuración.
- Proceso recomendado:**
- 1º ifdown ethX
 - 2º Reconfigurar
 - 3º ifup ethX
- Ejemplo:
 0º BOOTPROTO=dhcp
 1º ifup ethX (crea proceso dhclient).
 2º BOOTPROTO=static
 3º ifdown ethX (no elimina proceso dhclient).
 4º ifup ethX => Tarjeta toma valores manuales pero proceso dhclient los irá cambiando.



P00: Red: Des/Activación tarjetas de red

- ❑ Des/activación de las interfaces de red:

```

[root@lt28 ~]#
ping 172.16.17.228 (dirección_IP_propia) # OK (Wireshark lo)
ip l set down eth0 ← Sólo desactiva (no envía/recibe), pero NO desconfigura, mantiene IP (ifconfig -a)
ping 172.16.17.254 (otra_dirección_IP_de_la_misma_subred) # FALLA
ping 172.16.17.228 (dirección_IP_propia) # OK!! (lo)
/sbin/ifdown eth0 ← Desactiva y Elimina la IP de la interfaz (ya no está en "ip a") usando:
Si ifcfg-xxx con "TYPE=Ethernet" (omisión), flush ip addr flush dev eth0 scope global
Requiere exista: /etc/sysconfig/network-scripts/ifcfg-ppp con DEVICE=eth0
ping 172.16.17.228 (dirección_IP_propia) # FALLA (sin IP)
ip l set up eth0 ← Sólo Activa (NO configura IP), igual que "ifconfig eth0 up"
ip a add 172.16.17.228/25 dev eth0 ← Sólo configura IP y entrada directa (NO pasarela,...). No activa
ping 172.16.17.254 (otra_dirección_IP_de_la_misma_subred) # OK (iface activa)
ping 172.16.17.1 (equipo de otra subred) # FALLA!(no pasarela)
ip r add default via 172.16.17.254 dev eth0 nueva IP, entrada directa, pasarela, ...
pero si DHCP (BOOTPROTO=dhcp) => requiere PID proceso "dhclient" terminado
ping 172.16.17.1 (equipo de otra subred) # FALLA??
ifdown eth0; ifup eth0 # Reconfigura
    
```

IPs de red propias: son enviadas por la lo.

No borra IP de "lo" (scope "host")

Si ifcfg-xxx con "TYPE=Ethernet" (omisión), flush ip addr flush dev eth0 scope global Requiere exista: /etc/sysconfig/network-scripts/ifcfg-ppp con DEVICE=eth0

Si ifcfg-eth0 con "BOOTPROTO=dhcp", existe "/var/run/dhclient-eth0.pid" y contiene PID de un proceso existente (dhclient o no) => ifup NO hace nada. Solución manual: pkill -9 dhclient

P00: Red: ifconfig vs ip (3)

F

ifup ethX

usa

ip a add IP/m dev ethX

Añade nueva dirección IP

ifconfig eth1 IP/m

Asigna/Modifica sólo 1 IP

La IP usada como origen está determinada por la entrada de la tabla de enrutamiento usada (cada IP añade una entrada directa)

La interfaz tiene todas esas direcciones en uso.

Aunque "ifconfig" no las muestre.

```
[root@lt28 ~]# ip a list dev eth1
3: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc ...
    link/ether 68:05:ca:2a:03:59 brd ff:ff:ff:ff:ff:ff
```

```
[root@lt28 ~]# ifconfig eth1 10.0.0.1/24
[root@lt28 ~]# ip a add dev eth1 10.0.0.2/24
[root@lt28 ~]# ifconfig eth1 10.0.0.3/24
```

```
[root@lt28 ~]# ip a list dev eth1
3: eth1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500...
    link/ether 68:05:ca:2a:03:59 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.2/24 scope global eth1 ...
    inet 10.0.0.3/8 brd 10.255.255.255 scope global eth1...
```

```
[root@lt28 ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1
ONBOOT=no
BOOTPROTO=static
IPADDR=10.0.0.5
NETMASK=255.255.255.0
```

- Misma "IP/mask" sólo 1 vez.
- Pero Sí puede aparecer Misma IP con Distinta máscara: IP/mask1, IP/mask2

```
[root@lt28 ~]# ifup eth1
```

```
[root@lt28 ~]# ip a list dev eth1
3: eth1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500...
    link/ether 68:05:ca:2a:03:59 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.2/24 scope global eth1 ...
    inet 10.0.0.3/8 brd 10.255.255.255 scope global eth1...
    inet 10.0.0.5/24 brd 10.0.0.255 scope global se eth1...
```

```
[root@lt28 ~]# ifconfig eth1
eth1: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 10.0.0.2 netmask 255.255.255.0 broadcast...
```

Documentación de apoyo

Anexo-Linux_Configuracion_Red.pdf

P00: Red: Sockets locales

F

Sockets Locales (todos con "netstat -a")

Sockets "de Escucha" (Servidores): ss -l.



```
[root@lt28 ~]# ss -l
```

Netid	State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port
u_str	LISTEN	0	128	/tmp/.X11-unix/X0 55279	* 0
u_dgr	UNCONN	0	0	/run/systemd/shutdown 54540	* 0
u_str	LISTEN	0	128	/var/run/rpcbind.sock 54542	* 0

Mal en "ss" de CentOS (muestra "udp")

Protocolo sobre IP (icmp=1) en fichero "/etc/protocols"

Sockets IP: TCP, UDP, ..., RAW

```
raw UNCONN 0 0 *:* icmp
udp UNCONN 0 0 *:* domain
tcp LISTEN 0 128 *:* ftp
tcp LISTEN 0 128 *:* ssh
...
```

Protocolo/Servicio real no detectado!! (sólo L3/4), traducido usando el fichero "/etc/services"

ftp 21/tcp
ssh 22/tcp

Sockets "de Conexión" (Clientes): "ss" (todos), "-t" (TCP), "-u" (UDP), ...

netstat

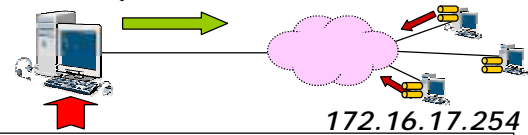
netstat -t

netstat -u

P00: Red: Sockets remotos

■ Sockets Remotos: **nmap** (**hping3**, **nping**, ...)

■ Equipos activos (ICMP Raw):



```
[root@lt28 ~]# nmap -sn 172.16.17.0/24
Host lt11.ait.us.es (172.16.17.11) appears to be up.
```

■ Puertos abiertos (envía SYN TCP, UDP 0 bytes, ...):

```
[root@lt28 ~]# nmap -ss 172.16.17.254
```

PORT	STATE	SERVICE
21/tcp	open	ftp
22/tcp	open	ssh
...		

Escanea puertos más comunes (sobre 1000, en `/usr/share/nmap/nmap-services`)

Protocolo/Servicio real no detectado!!: Nombre estándar del puerto (IANA), traducido usando el fichero `/usr/share/nmap/nmap-services`; (NO usa `/etc/services`)

ftp	21/tcp
ssh	22/tcp
...	

```
[root@lt28 ~]# nmap -p 5-10,20 IP
```

43

<https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml>

P00: Red: Asignación Puertos (Aclaración)

■ ¿Por qué en `/etc/services` se asignan protocolos de transporte a servicios que no los soportan (e.g. 22/udp y 22/tcp a SSH, que sólo soporta TCP)?:

- `/etc/services` (predeterminado): escrito para respetar asignación IANA.
- IANA propone una asociación 1 a 1 (Protocolo de Aplicación-Puerto) válida para todos los protocolo de transporte con puerto: por simplicidad de memorización y configuración. Si no fuese 1 a 1:
 - Para indicar el puerto asignado a un protocolo de aplicación, además del número de puerto, habría que indicar también el protocolo de transporte.
 - Hay protocolos de aplicación que pueden transportarse sobre varios protocolos de transporte (e.g., DNS admite UDP y TCP). Si la especificación de un protocolo de aplicación cambiase, admitiendo un nuevo protocolo de transporte, obligaría a:
 - Actualizar la asignación de puertos IANA (y sus copias, como `/etc/services`).
 - Si ese nuevo protocolo de transporte soportado está asignado a otro servicio: habría que desplazar ese otro servicio a un puerto distinto (o asignar un puerto diferente para ese protocolo de transporte ahora soportado), conllevando graves problemas de gestión (configuración y memorización).

■ Conclusión: se prefiere una asociación 1 a 1, aunque el protocolo de aplicación sólo pueda operar sobre un protocolo de transporte.

44



P00: Red: Comando ip

- ❑ **Comandos Clásicos** Administración Red (ifconfig,...): **Obsoletos**.
- ❑ **"ip"**: **abarca esas funcionalidades y muchas más**.

Sintaxis: `ip [opciones] l/a/n/r/ru/t/ma/mr/mo comando [args]` `man 8 ip`

Activación de Interfaces	<code>ifconfig</code>	<code>ip link/li/l show</code> <code>ip link set dev ethX up/down</code> <code>ip link set dev ethX promisc/arp/dynamic on/off</code> <code>ip link set dev ethX mtu valor_MTU_en_bytes</code>
Configuración IP de interfaces	<code>nameif</code>	<code>ip address/addr/a show Broadcast: FF:FF:FF:FF:FF:FF</code> <code>ip address add IP/mask brd+ dev ethX</code>
Caché L2 (ARP....)	<code>arp</code>	<code>ip neighbour/neighbor/neighbor show</code>
Tabla de enaminamiento	<code>route</code>	<code>ip route/ro/r show</code> <code>ip route add IP/mask via IP_gw</code>
Políticas encam. (RPDB): NATs,...	---	<code>ip rule/ru show</code> <code>ip rule add nat IP from IP_subred/mask</code>
Tunelización (GRE)	<code>iptunnel</code>	<code>ip tunnel/tunn/tunl/t show</code> <code>ip tunnel add nombre local IP1 remote IP2 ttl 16</code>
Direccionamiento multicast	<code>ipmaddr</code>	<code>ip [-0] maddress/maddr/ma/m ls [ethX]</code> <code>ip maddr add MAC dev ethX</code>
Encam. multicast	---	<code>ip mroute/mro/mr ls [ethX]</code>
Monitorización	---	<code>ip monitor/mon/mo file fichero link/address/route</code>


```

[root@lsc30 ~] ip link set dev eth0 name tarjeta ← Clásico: nameif
[root@lsc30 ~] ip link set dev tarjeta address 01:02:03:04:05:06 ← Antes con: macaddress
[root@lsc30 ~] ip link show dev tarjeta
2: tarjeta: <BROADCAST,UP> mtu 1500 qdisc pfifo_fast qlen 100
   link/ether 01:02:03:04:05:06 brd ff:ff:ff:ff:ff:ff
  
```

45

P00: Red: Comandos ip/ss (Equivalencias Básicas)

<code>arp -n</code>	<code>ip neighbour</code>
...	<code>ip n</code>
...	...
<code>ifconfig</code> (activas)	<code>ip address show up</code>
<code>ifconfig -a</code>	<code>ip address [list show]</code>
<code>ifconfig ethX</code>	<code>ip a [l s] ethX</code>
<code>ifconfig ethX up/down</code>	<code>ip l set up/down ethX</code>
...	...
<code>route -n</code> (formato numérico)	<code>ip route [ls sh]</code>
<code>route -n</code> (traducción a nombres)	<code>ip route -r [ls sh]</code>
...	...
<code>ifconfig eth1 10.1.0.1 netmask 255.255.255.0</code>	
<code>ip addr add/del 10.1.0.1/24 dev eth1</code>	
<code>route add default gw 10.1.0.254 netmask 0.0.0.0 dev eth1</code>	
<code>ip route add default via 10.1.0.254 dev eth1</code>	
<code>route add 10.1.0.0 gw 10.1.0.254 netmask 255.255.255.0 dev eth1</code>	
<code>ip route add 10.1.0.0/24 via 10.1.0.254 dev eth1</code>	

- ❑ **"ss"**: sintaxis similar a **"netstat"**
(más filtros de búsqueda)

<code>netstat -a</code>	<code>ss -a</code>
<code>netstat -l</code>	<code>ss -l</code>
<code>netstat -t</code>	<code>ss -t</code>

ip Internet Protocol
ss Socket Statistics

46

P00: Red: Sockets e interfaz "lo"

F

- Si se **desactiva** la interfaz "lo":
 - No habrá **respuestas** de la dirección **127.x.x.x**
 - Los **accesos** a "IP_ethX" no podrán cursarse por "lo".
 - Los servicios que realicen **comunicación interna**:
 - IP (ping, ssh local, http://localhost/, ...): **no funcionarán**
 - No IP (X11, *sistema*, ...): **funcionarán** (sockets **UNIX**)

```
[root@lt28 ~]# ifdown lo
```

```
[root@lt28 ~]# ping 127.0.0.1
```

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data:
```

```
[root@lt28 ~]# ip a show eth0
```

```
2: eth0: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc ...  
    link/ether 60:02:92:18:5c:a0 brd ff:ff:ff:ff:ff:ff  
    inet 172.16.17.228/25 brd 172.16.7.255 scope global dynamic eth0  
    ...
```

```
[root@lt30 ~]# ping 172.16.17.228
```

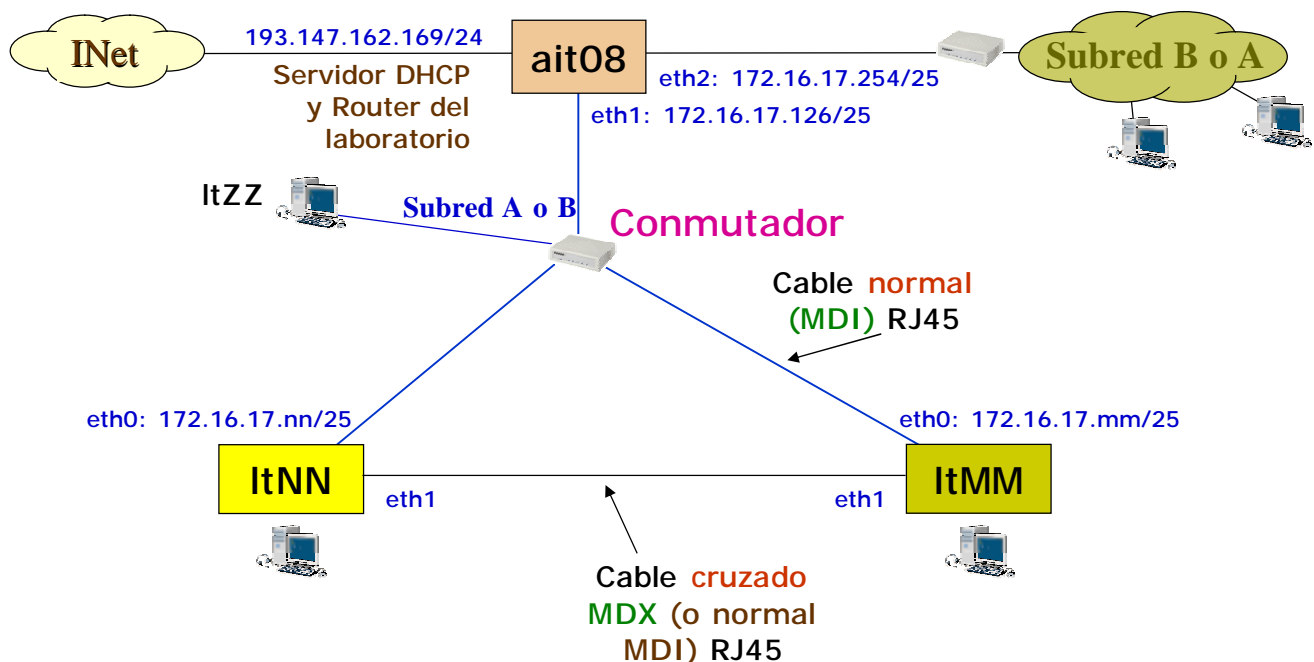
```
connect: Invalid argument
```

□ "ping" conoce destino (excepcionalmente, para "lo" ifdown NO elimina la IP, sólo pone DOWN)=> Intenta enviar por "lo", pero...

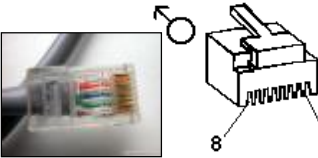

□ "lo" no cursa tráfico => No llega respuesta

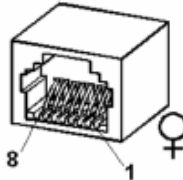


kernel (función "connect()") intenta redirigir ICMP a "lo", encontrando no disponible.

P00: Red: Cableado, Laboratorio



P00: Red: Cableado, Conectores RJ45

Conector	Pineado
  <p>Crimpadora RJ45 macho</p>	<p>EIA/TIA-232</p> <p>1 RxD+</p> <p>2 RxD-</p> <p>3 TxD+</p> <p>4 Sin uso</p> <p>5 Sin uso</p> <p>6 TxD-</p> <p>7 Sin uso</p> <p>8 Sin uso</p>

Conector	Pineado
   <p>Crimpadora de impacto para RJ45 hembra</p>	<p>EIA/TIA-232</p> <p>1 RxD+</p> <p>2 RxD-</p> <p>3 TxD+</p> <p>4 Sin uso</p> <p>5 Sin uso</p> <p>6 TxD-</p> <p>7 Sin uso</p> <p>8 Sin uso</p>

Asignación de Colores

Rx	Verde
Tx	Naranja

Rx	Naranja
Tx	Verde

49

P00: Red: Cableado, Conectores RJ45 (2)

Cable normal:
MDI (Medium
Dependent
Interface)

PCPCPC

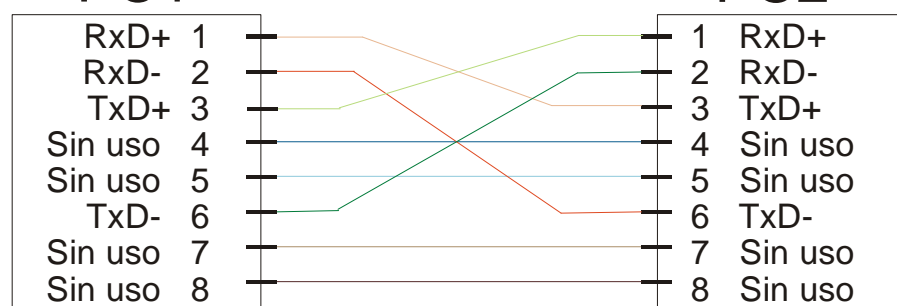
HUB/SWITCH



PC1

PC2

Cable cruzado:
MDX (Cross-
over MDI)






50

P00: Red: Cableado, Interconexión equipos

□ Interconexión de:

- Equipos
- Elementos de red (hubs, switch, routers, ...)

Interconexión	Tipo de cable
	Normal (MDI)
	Cruzado (MDX) o Normal*
	

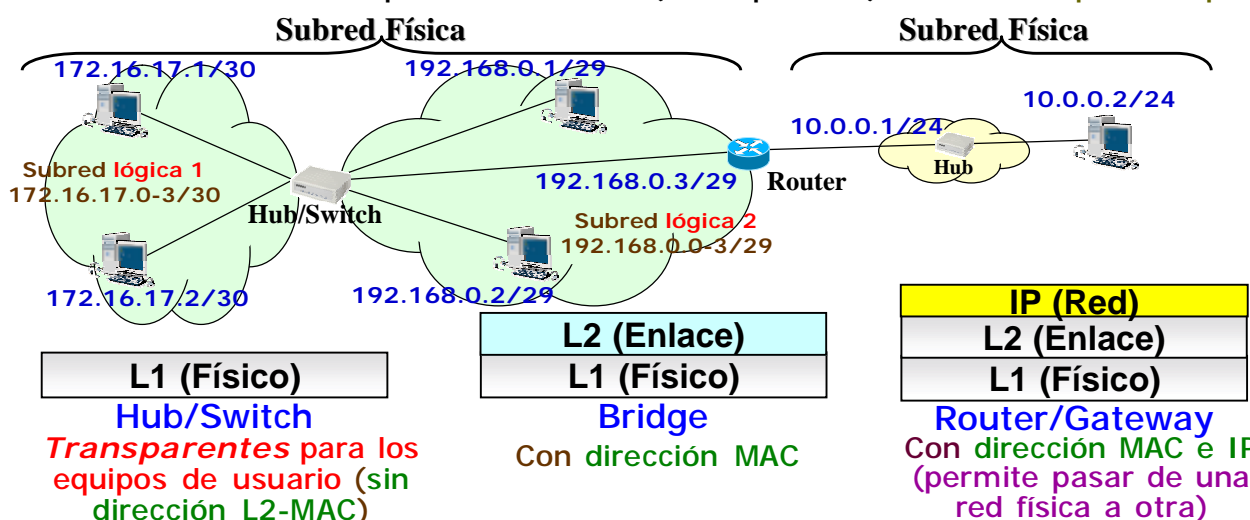
*Las tarjetas y dispositivos de red modernos presentan conectores de tipo **Auto-MDI /MDI-X** (puerto Uplink), con capacidad para realizar automáticamente la inversión Tx-Rx, eliminando la necesidad de usar cables cruzados.

51

P00: Red: Elementos de red

□ Los repetidores/conmutadores ofrecen una comunicación a nivel físico L1, constituyendo una subred física (mismo dominio de colisión), dentro de la cual pueden crearse subredes lógicas (según la configuración de los equipos).

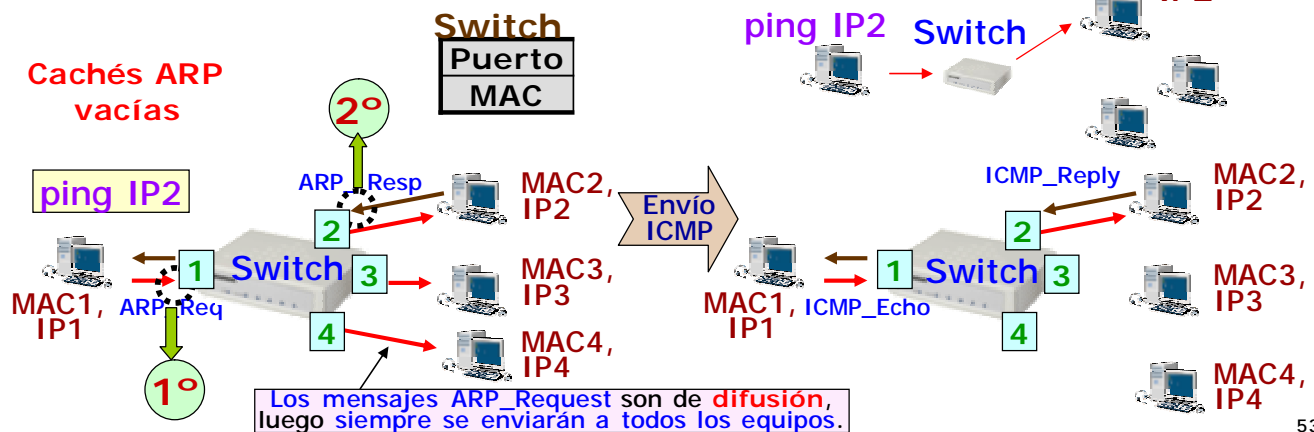
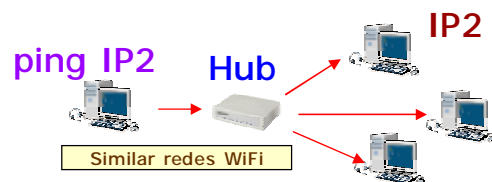
- Los repetidores emulan un medio físico compartido (bus).
- Los conmutadores permiten emular (tras aprender) conexiones punto a punto.



52

P00: Red: Elementos de red (2)

- Encaminador, L3: interconecta subredes (distintos dominios de colisión).
- Repetidor (hub), L1: reenvía por todas las salidas.
- Conmutador (switch), L2: tras aprender (basándose en la MAC; rara vez por IPs), sólo reenvía por la salida del destino.



53

P00: Cuestiones de examen

[TEST]

Cuando en una subred Ethernet basada en IP se envían paquetes a otra máquina, por ejemplo con "http://IP/" o "ping IP" (siendo IP la dirección de un equipo de la misma subred del que sólo conocemos dicha dirección), en el mecanismo de acceso que tiene lugar:

- (a) Siempre se usa la dirección IP para alcanzar el equipo destino y un puerto TCP/UDP para determinar como gestionar el mensaje dentro de ese equipo. B
- ✓ (b) En general, se hará uso, al menos una primera vez, del protocolo ARP. A
- (c) Debe haber siempre un proceso asociado con un servicio de nivel de aplicación a la escucha en la máquina destino, el cual gestionará convenientemente el mensaje que se ha enviado a dicha máquina. C

P00: Cuestiones de examen (2)

Ⓔ[MULTIPLE]

Monitorizando tráfico con el analizador Wireshark mediante la siguiente expresión de filtrado:

(tcp and src port 80) or (ether proto arp and port 42) or proto icmp

- ✓ (a) Podrá visualizar las respuestas HTTP que le envían los servidores web (asumiendo que éstos usan el puerto estándar) a los que se conecte.
- (b) Capturará los paquetes ARP que lleguen por el puerto 42. ☐
- ✓ (c) Normalmente, permitirá comprobar si algún usuario está realizando pings masivos contra la máquina en la que se encuentra el analizador.
- (d) Capturará los mensajes ICMP que usen el protocolo de transporte TCP o UDP.

55

P00: Cuestiones de examen (3)

[RESPUESTA CORTA]

Asumiendo sistemas de particionamiento clásicos MBR (no GPT), se desea:

- ▣ Montar en el directorio /mnt/linux la segunda unidad lógica del disco duro situado como maestro en el segundo canal IDE (secundario maestro), en la que se está instalado el S.O. Linux CentOS con el sistema de ficheros EXT versión 3.
- ▣ Montar en el directorio /mnt/usb un disco duro USB, dotado de una única partición FAT32 (sin unidades lógicas), al que el sistema le ha asignado la letra "e".

Indique cuáles deben ser los valores de los campos "AA", "BB", "CC", "DD" para que los comandos siguientes lograsen el objetivo buscado: ☐

```
mount -t AA BB /mnt/linux; mount -t CC DD /mnt/usb
```

Solución:

AA = ext3; BB = /dev/sdc6;
CC = vfat; DD = /dev/sde1

56

P00: Cuestiones de examen (4)

[RESPUESTA CORTA] - Repaso

Se dispone de dos máquinas en una determinada red, conociéndose las direcciones IP que tienen asignadas dichas máquinas: "172.16.17.9" y "172.16.17.11".

Indique cual es la definición de la subred más pequeña (según la recomendación RFC791, Internet Protocol v4) que podría albergar ambas direcciones (use la notación CIDR para escribir la subred: "IP/máscara").

57

P00: Cuestiones de examen (4)

□ Subred: IP_subred/Máscara

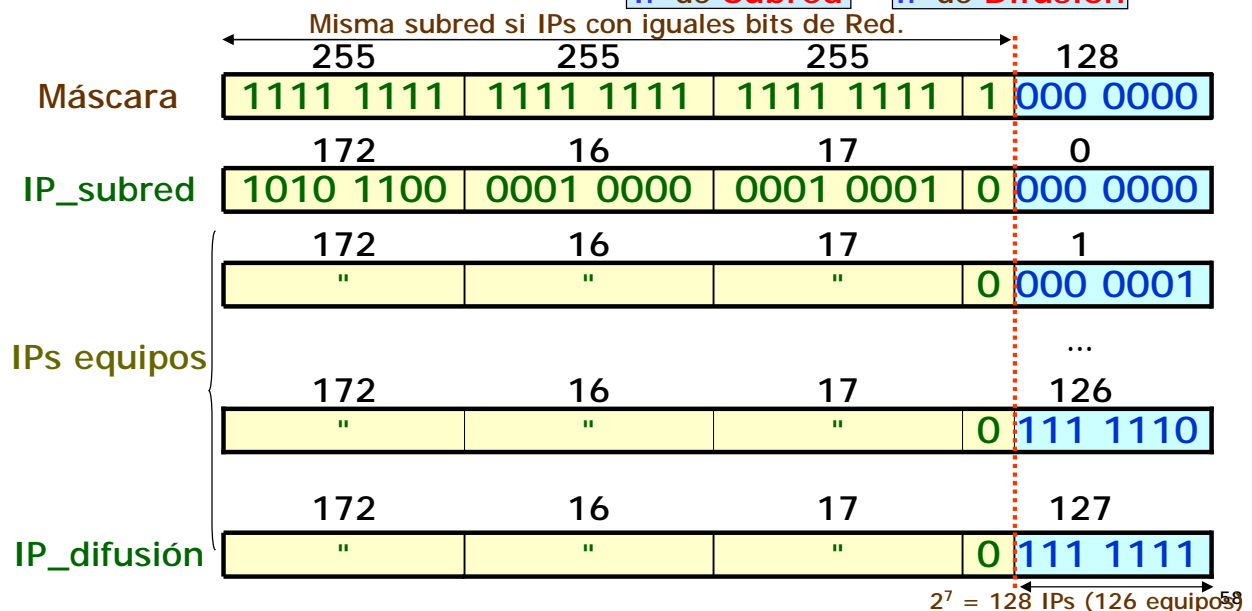
- **Máscara**: Determina cuantas IPs (tamaño) componen la subred

- Notación **INET**: 255.255.255.128

- Notación **CIDR** (Classless InterDomain Routing): IP/25

□ Ej.: 172.16.17.0/25 o 255.255.255.128 => 172.16.17.0 ... 172.16.17.127

IP de Subred IP de Difusión



P00: Cuestiones de examen (4)

Máscara Binaria con: N 0's → Rango con: 2^N IPs

$N = 32 - \text{CIDR}$

N saltos ($2^N - 1 = 255 - \text{INET}$)

Mask CIDR (1's)	Mask INET (Nº "0's")	Mask Binaria	Rango: 10.0.0.0/mask	Total IPs (2^{N-0s})	IPs usables
32	255.255.255.255	1111 1111	10.0.0.0	$2^0 = 1$	1
31	255.255.255.254	1111 1110	10.0.0.0-1	$2^1 = 2$	0
30	255.255.255.252	1111 1100	10.0.0.0-3	$2^2 = 4$	2
29	255.255.255.248	1111 1000	10.0.0.0-7	$2^3 = 8$	6
28	255.255.255.240	1111 0000	10.0.0.0-15	$2^4 = 16$	14
27	255.255.255.224	1110 0000	10.0.0.0-31	$2^5 = 32$	30
26	255.255.255.196	1100 0000	10.0.0.0-63	$2^6 = 64$	62
25	255.255.255.128	1000 0000	10.0.0.0-127	$2^7 = 128$	126
24	255.255.255.0	0000 0000	10.0.0.0-255	$2^8 = 256$	254
23	255.255.254.0	...	10.0.0-1.0-255	$2^9 = 512$	510
22	255.255.252.0	...	10.0.0-3.0-255	$2^{10} = 1024$	1022

59

P00: Cuestiones de examen (4)

	172	16	17	9	← Parte coincidente
IP1	"	"	"	0000 1001	
IP2	172	16	17	11	
	"	"	"	0000 1011	

60

P00: Cuestiones de examen (4)

	172	16	17	9	← Parte coincidente
IP1	"	"	"	0000 1001	
	172	16	17	11	
IP2	"	"	"	0000 1011	

Solución: 172.16.17.8/29

61

P00: Resumen/Conclusiones

- Entorno de Trabajo del Laboratorio
- S.O. Linux: Conceptos y Administración
 - Intérprete de comandos
 - Administración Local: Comandos
 - Red: Kernel, Configuración, Comandos, Analizadores

Entorno Virtualizable ➡

Cuestionarios: Plazos de entrega en Planificación
(Pulsar en "Enviar" antes de fin de Plazo)

62

P00: Reproducción Entorno Virtual (VMWare)

F

63

Instalación de software en Linux

F

Formato de Distribución	Características	Herramientas de uso
Paquetes .rpm (en Debian ".deb")	Software compilado : requiere librerías (y distribución) específicas . Instalación: <input type="checkbox"/> Manual <input type="checkbox"/> Asistida por repositorios (BD aplicaciones) <input type="checkbox"/> Con entorno gráfico	--- rpm -i yum, apt synaptic
Paquetes binarios .bin, .run, .egg, .package	Software compilado : idem rpm con autoinstalador (también "autopackage").	Ejecución directa
Código fuentes (rpms, bin, tgz, tar.bz2, ...)	<input type="checkbox"/> Menos dependencia de la distribución. <input type="checkbox"/> Necesita ser compilado .	Compilar el código

* Descarga de código fuente con subversión: `svn co -username user http://IP/alias/trunk/ /dir/`
 * Existen herramientas que ayudan a la desinstalación, i.e. Checkinstall (monitoriza instalación).

Software Interpretado (Ej: Java)	<input type="checkbox"/> Independiente de plataforma : sólo intérprete (Ej: SDK). <input type="checkbox"/> Distribución: paquetes (rpm) o comprimido (tar.gz)
Comprobación de Integridad del software (.MD5)	<input type="checkbox"/> Crear checksum : <code>md5sum software > file.md5</code> <input type="checkbox"/> Comprobar ("file.md5" contiene la cadena MD5 y el nombre fichero "software". El fichero a comprobar es buscado en la misma carpeta donde se encuentra el ".md5"): <code>md5sum -c file.md5</code>

64

Instalación de software en Linux (2)

- Proceso para compilar el código fuente de los programas: suele ser estándar, sí dependiendo del lenguaje del código.

■ Pasos de Instalación: fichero "INSTALL", "README", ...

Lenguaje	Proceso habitual para la compilación
C/C++	<pre>make clean # Elimina ficheros previamente compilados ./configure # Prepara el entorno (suele instalar en /usr) make # Compila usando el fichero Makefile make install # Copia los ficheros compilados en el sistema</pre> <p>□ No instalar en "/usr": ■ Instalación: <code>./configure --prefix=/usr/local</code></p>
Perl	<pre>perl Makefile.PL # Genera Makefile con patrón "Makefile.pl" make manifest # Genera "MANIFEST" (lista ficheros paquete) make # Compila el código make install # Instala el paquete en el sistema</pre>
Python	<pre>python setup.py build # Genera el bytecode python setup.py install # Instala el paquete en el sistema</pre>

