

Curso Linux

UNIDAD II

[Unidad II@presentación]#

2.1 Administración de usuarios

2.2 La línea de comandos (shell) y utilerías

2.3 Arranque y parada.

2.4 Sistema de archivos

2.5 Servicios del núcleo del sistema

2.6 Kernel de Linux

2.7 Seguridad

[Usuarios@Unidad II]\$

- La representación de alguien (persona o proceso) que pueden trabajar en el sistema. P.ej. : root, postgresql, jcolivar, fmorales.
- Cada usuario posee:
 - tiene un login
 - directorio base \$HOME: */home/dsc/jcolivar*.
 - grupo, por ejemplo: *users, root*.
 - perfil, *bash_profile*

[Usuarios@Unidad II]\$

- `useradd` agrega usuario
- *adduser* `useradd` interactivo
- `userdel` borra usuarios
- `usermod` modifica usuarios

- *groupadd* agrega grupos
- *groupdel* borra grupos
- `groupmod` modifica grupos

[Usuarios@Unidad II]\$

- passwd cambiar contraseña
- login entrar al sistema

- who lista los usuarios del sistema
- w versión especial del listado de usuarios
- whoami indica el usuario actual

[Usuarios@Unidad II]\$

- Linux no soporta Listas de Control de Acceso ACL
- Contraseñas seguras son aquellas que no pertenecen a ningún idioma, contienen números y signo de puntuación.
- Las contraseñas están cifradas.

[/etc/passwd@Unidad II]\$

- gandalf::501:100:Eric Dumas:/home/gandalf:/bin/bash
sgandalf::0:0:Super Eric Dumas:/:/bin/sh
- Nombre del usuario: 8 caracteres máximo*
- Contraseña codificada
- Número de identificación único del usuario (uid). Nota: 0 = root
- Numero del grupo (gid)
- Nombre "oficial" del usuario (llamado GECOS)
- Directorio raíz del usuario
- Intérprete de órdenes utilizado

[useradd@Unidad II]\$

- -c comment
- -d homedir
- -e expire-date
- -f inactive-time tiempo de expiración de cuenta
- -g initial_group
- -G group
- -m directorio home
- -M no crear directorio home
- -s shell
- -u uid

```
[/etc/group@Unidad II]$
```

```
root:x:0:root
```

```
bin:x:1:root,bin,daemon
```

```
tty:x:5:
```

```
disk:x:6:root
```

```
lp:x:7:daemon,lp
```

```
mail:x:12:mail
```

```
news:x:13:news
```

```
uucp:x:14:uucp
```

```
nobody:x:99:
```

```
users:x:100
```

[Quotas@Unidad II]\$

- Kernel: options QUOTA
- archivo /etc/sysconfig (quotas=yes|no)
- check_quotas=YES

- /etc/fstab
- *userquota*
- /dev/sd1s2g/homeufs rw, userquota 1 2

[Quotas@Unidad II]\$

- /dev/sd1s2g/homeufs rw, userquota,
groupquota 1 2
- *quota.user y quota.group*
- *quotacheck, quotaon, o quotaoff*
- quota -v
- edquota -u test

[Unidad II@presentación]#

2.1 Administración de usuarios

2.2 La línea de comandos (shell) y utilerías

2.3 Arranque y parada.

2.4 Sistema de archivos

2.5 Servicios del núcleo del sistema

2.6 Kernel de Linux

2.7 Seguridad

[Shell@Unidad II]\$

- Interfaz en modo de comandos para operar el SO.
- Existen muchos Shells (BASH, CSh, TSh, etc.) los cuales pueden ser personalizados por los usuarios.
- # Indica superusuario (root)
- \$ Indica usuario normal

[Shell@Unidad II]\$

- /etc/shells
- .archivo archivo oculto
- Cambio de contraseña con * a4dd89k* motivo de la cancelación
- /etc/inetd.conf
- El comando más importante es... man

[Shell@Unidad II]\$

- export exporta variable de ambiente
- env ambiente
- set configura variable de ambiente
- history historial

- shutdown apagar la máquina
- poweroff apagar la máquina
- halt apagar la máquina
- reboot reiniciar la máquina

[Shell@Unidad II]\$

- unset variable
 - ls -l; cat /etc/passwd
 - -l
 - -a
 - -1
 - -r
-
- /usr/bin/loadkeys
 - /usr/lib/kdb/keytables/es.map

[Shell@Unidad II]\$

- kill mata procesos
- ps estado de los procesos
- nice aumentar prioridad de procesos
- nohup no dejar colgado un proceso

- sleep dormir proceso
- wake despertar proceso
- awake

[Shell@Unidad II]\$

- ps
- -a todos los procesos con el terminal controlado, excepto el proceso de usuario actual.
- -r ejecución
- -x procesos sin terminal controlado
- -l formato largo

[Shell@Unidad II]\$

- top visor de procesos
- uname -a -r

- /etc/rc.d/init.d/
- /etc/init.d

- Anaconda
- Kudzu

[Shell@Unidad II]\$

- `chmod` cambiar permisos de archivos
- `chown` cambiar dueños de archivos
- `chgrp` cambiar grupo de los archivos

- `mkfs` formatear sistema de archivo
- `printenv` ver variables de entorno
- `sync` vaciar búfer de sistemas de archivo

[Shell@Unidad II]\$

- pwd directorio actual
- cp copiar archivos
- rm borrar archivos
- mv mover archivos
- * todos los caracteres
- ? Un solo caracter
- > redireccionar la salida
- cat
- < redireccionar la entrada

[Shell@Unidad II]\$

- `ls > /tmp/lista_directorio`
- `ls >> /tmp/archivo`
- `grep 'root' < /etc/passwd`

- `sort < archivo1 > archivo2`
- `cat arch1 arch2 arch3 | sort > /dev/lp`

- `a.out > /dev/null`

[Shell@Unidad II]\$

- `clear; ls -l;`
- `kill `cat /var/run/named.pid``
- `su – usuario`
- `mount /mnt/cdrom`
- `umask u=rw,g=,o=`

[Shell@Unidad II]\$

- Otro formato de ayuda es info cuya característica principal es que permite hipertexto en la Shell.
- man 3 printf
- man 1 printf
- ls(1)
- man -k printf

[Shell@Unidad II]\$

- mv, cp
- -f forzar -i interactivo

- mknod /tmp/mi_pipe p
- which ls

- setup
- linuxconf
- netconf

[Shell@Unidad II]\$

- | tuberías
- & proceso en segundo plano

- touch actualiza archivo
- du espacio ocupado por directorio
- df muestra el espacio ocupado por cada sistema de archivo
- uptime fecha y tiempo activo

[Shell@Unidad II]\$

- more ver archivos largos
- less ver archivos largos interactivo
- head muestra las primeras 10 líneas de un archivo
- tail muestra las últimas 10 líneas de un archivo
- file identifica tipo de archivo
- grep búsqueda de patrones

[Shell@Unidad II]\$

- `wc` cuenta el número de palabras, líneas, caracteres de un archivo
- `spell`, `ispell` ortografía
- `cmp` compara dos archivos
- `diff` muestra las diferencias de dos archivos

- `gzip` comprimir archivo
- `gunzip` descomprimir archivo

[Shell@Unidad II]\$

- tr traductor cambia palabras de un archivo por otro
- mail envío de correo
- finger información sobre usuarios
- find busca archivos
- test realiza comparaciones
- dd duplicador de datos

[Shell@Unidad II]\$

- fg trae a primer plano un proceso
 - jobs muestra los trabajos en el sistema
 - bg manda un proceso a segundo plano
 - Ctrl+c interrupción
 - Ctrl+z suspender
-
- clear borra pantalla
 - date fecha
 - man manuales

[Shell@Unidad II]\$

- mc shell visual
- cal calendario

- vi editor de texto
- pico editor de texto y correo
- ed editor de texto en una línea
- joe editor de texto

[Shell@Unidad II]\$

- su cambia de usuario
- sudo ejecutar un comando en nombre a tu usuario
- ln (ln -s links lynx) enlaces

- rmdir borra un directorio
- mkdir crea un directorio

[Shell@Unidad II]\$

- cut quita tablas de un archivo
- read var captura el valor de una variable
- sort ordena las líneas de un archivo
- uniq borra líneas duplicadas sucesivas

- sed y awk programas de procesamiento de patrones

[Shell@Unidad II]\$

- *sum* verifica la suma de comprobación
- *md5sum* suma de comprobación

- *fsck* hace una revisa al sistema de archivo
- *cpio* copiado de archivos a bajo nivel

- *write* escribe un mensaje a un usuario
- *talk* plática

[Shell@Unidad II]\$

- mount monta sistema de archivos
- umount desmonta dispositivo

- mount -a monta todos los sistemas de archivos de /etc/fstab
- -t ext2, FAT, VFAT, FAT32

[Shell@Unidad II]\$

- Opciones de mount
- ro read only
- rw read write
- exec
- noauto no monta de manera automática
- e2fsck
- lost + found

[Shell@Unidad II]\$

\$1 - \$9 parámetros

\$0 nombre del comando actual

\$# número de parámetros

\$? status del último comando ejecutado dado como un string decimal.

\$\$ el número de proceso de este shell

\$! el pid del último comando ejecutado en background.

\$- las opciones actuales suministradas para esta invocación del shell.

\$* @\$@@ un string que contiene todos los argumentos del shell comenzando por el \$1.

[Shell@Unidad II]\$

- **sh -x script argument**
- Cambiar permisos al script de ejecución y firmarlo
- “ Literal simple
- ` Sustituir valor de la variable
- “” Sustituir por lo que haya en las comillas
- Shift avanzar parametros
- test []
- echo
- && ||

[Shell@Unidad II]\$

- test
- -d directorio
- -e archivo sin importar el tipo
- -f archivo normal
- -r leer archivo
- -s tamaño mayor que 0
- -w escribir archivo
- -x archivo ejecutable
- -eq iguales
- -ne no son iguales

[Shell@Unidad II]\$

- -gt mayor
- -ge mayor o igual
- -lt menor
- -le menor o igual
- = cadenas idénticas
- != cadenas no son idénticas
- < menor
- > mayor
- -n longitud de la cadena de texto es distinta de cero
- ! Negación
- -a and
- -o or

[Shell@Unidad II]\$

```
/etc/crontab
```

```
# hacer una copia de seguridad todos los lunes a las 2 de la  
mañana
```

```
0 2 * * 1 /home/gandalf/backup
```

```
# Reiniciar la máquina todos los 1 y 15 de cada mes a la 4:15  
15 4 1,15 * * /sbin/shutdown -r +3
```

- Las 5 primeras columnas son :
 - Minutos (0 a 59) ;
 - Hora (0 a 23) ;
 - Día del mes (1 a 31) ;
 - Mes (1 a 12) ;
 - Día de la semana (0 a 6 : 0 = Domingo, 1 = Lunes).

[Shell@Unidad II]\$

- banner muestra un mensaje
- *_nombre_funcion_ () { _lista_de_ordenes_ }*
- whereis
- awk '{print \$1}' archivo
- sed s/tomate/pera/g a.txt > b.txt

[Shell@Unidad II]\$

- if;
 - then
 - elif;
 - then
 - else
 - fi
-
- [-f "/etc/shadow"] && echo "Este ordenador usa shadow passwords"

[Shell@Unidad II]\$

```
case ... in
... )
    hacer algo aquí;;
esac
```

```
while ...;
do ....
done
```

```
for var in ....;
do ....
done
```

[Shell@Unidad II]\$

```
#!/bin/sh
```

```
test -x /home/www/bin/httpd || exit 0
```

```
case "$1" in #cuál fue el parámetro de invocación
```

```
start)
```

```
    echo -n "Iniciando servidor http"
```

```
    start-stop-daemon --start --quiet --pidfile /var/run/httpd.pid \
```

```
    --exec /home/www/bin/httpd
```

```
    echo "."
```

```
;;
```

```
stop)
```

```
    echo -n "Deteniendo servidor httpd"
```

```
    start-stop-daemon --stop --quiet --pidfile /var/run/httpd.pid
```

```
    echo "."
```

```
;;
```

```
[Shell@Unidad II]$
```

```
restart)
```

```
    /etc/init.d/httpd stop
```

```
    sleep 2
```

```
    /etc/init.d/httpd start
```

```
    echo "."
```

```
;;
```

```
*)
```

```
    echo "Uso: /etc/init.d/httpd {start|stop|restart}"
```

```
    exit 1
```

```
esac
```

```
exit 0
```

```
[Shell@Unidad II]$
```

```
#!/bin/sh
```

```
if ["$SHELL" = "/bin/bash"];
```

```
then
```

```
    echo "tu shell es el bash (bourne again shell)"
```

```
else
```

```
    echo "tu shell no es bash sino $SHELL"
```

```
fi
```

[Shell@Unidad II]\$

```
# /etc/profile
```

```
# System wide environment and startup programs
```

```
# Functions and aliases go in /etc/bashrc
```

```
if ! echo $PATH | /bin/grep -q "/usr/X11R6/bin" ; then
```

```
    PATH="$PATH:/usr/X11R6/bin"
```

```
fi
```

```
USER=`id -un`
```

```
LOGNAME=$USER
```

```
MAIL="/var/spool/mail/$USER"
```

```
HOSTNAME=`/bin/hostname`
```

```
HISTSIZE=1000
```

```
[Shell@Unidad II]$
```

```
#!/bin/sh
```

```
#Programa que borra todos los archivos (correos) que  
contienen la palabra 'To: jazarate'
```

```
if [ -d cd /var/mail ]
```

```
then
```

```
    grep -l "To:" * | xargs rm
```

```
    echo "Se borraron los archivos solicitados"
```

```
else
```

```
    echo "Imposible borrar, no existe el directorio de correo"
```

```
fi
```

```
[Shell@Unidad II]$
```

```
#!/bin/sh
```

```
#Programa que calcula cuantos usuarios hay  
en el sistema
```

```
echo "Numero de usuarios en el sistema: `who  
| awk '{print $1}' | sort -u | wc -l`"
```

[Shell@Unidad II]\$

```
#!/bin/sh
```

```
#Script en shell de linux que calcula el usuario que tiene mas correos en el sistema
```

```
if [ -d /var/mail ]
```

```
then
```

```
    cd /var/mail
```

```
    a=`grep -r -c "Message-ID" * | awk 'BEGIN{FS=":"}{ print $2}' | sort -g -r |  
    awk '{getline X ;print $1}'`
```

```
    clear
```

```
    echo "El usuario con mas correo es: `grep -r -c "Message-ID" * | grep $a |  
    awk 'BEGIN{FS=":"}{ print $1}'`"
```

```
else    # No hay spool de correo en el sistema, salir
```

```
    echo "Lo sentimos pero no existe servidor de correo o no se encontro"
```

```
fi
```

[Shell@Unidad II]\$

```
#!/bin/sh
```

```
#Shell que crea un usuario pasando el nombre como  
parámetro así como su directorio public html
```

```
if [ $# -eq 1 ]
```

```
then #creamos el nuevo usuario
```

```
    useradd $1
```

```
    mkdir /home/$1
```

```
    cd /home/$1
```

```
    mkdir public_html
```

```
else
```

```
    echo    "!Sintaxis incorrecta!.    Uso    correcto:    $0  
    nombre_usuario"
```

```
fi
```

[Unidad II@presentación]#

2.1 Administración de usuarios

2.2 La línea de comandos (shell) y utilerías

2.3 Arranque y parada

2.4 Sistema de archivos

2.5 Servicios del núcleo del sistema

2.6 Kernel de Linux

2.7 Seguridad

[Arranque@Unidad II]\$

- El hardware lee el sector de arranque
- Se carga LILO/GRUB
- Se carga el Kernel
- Se monta el sistema de archivos raíz
- Se inicia init, el abuelo de todos los procesos
- Se lee /etc/inittab

[Arranque@Unidad II]\$

- Ejecuta todos los scripts de nivel 1
- El final del nivel 1 indica que se vaya a 3
- Ejecuta todos los scripts de nivel 3
- El sistema está listo para usarse (se deberá autenticar para acceder al sistema)

[LILO@Unidad II]\$

- Linux LOader, es el cargador del Kernel más común para Linux
- Uso casi exclusivo para arranques desde disco duro
- Configuración en `/etc/lilo.conf`
- Cargador en sector de arranque y en `/boot/map`

[Configuración LILO@Unidad II]

Default = linux

Boot = /dev/hda

Prompt

Timeout = 50

Image = /boot/vmlinuz-2.2.5-15

Label = linux

Root = /dev/hda2

Read-only

Other = /dev/hda1

label = DOS

table = /dev/hda

[grub@Unidad II]\$

- GRand Unified Bootloader, es el cargador del Proyecto GNU
- Muy versátil y configurable, permite arrancar casi cualquier sistema operativo
- Entiende internamente FAT, ext2, Reiser, JFS...
- Configuración en `/boot/grub/menu.lst`.

```
[Configuración GRUB@Unidad II]$
```

```
title Debian GNU/Linux
```

```
root (hd0,1)
```

```
kernel /boot/vmlinuz root=/dev/hda1 ro
```

```
title Debian GNU/Linux, kernel 2.6.10
```

```
root (hd0,1)
```

```
kernel /boot/vmlinuz-2.6.10 root=/dev/hda1 ro
```

[Otros cargadores@Unidad II]\$

- SYSLINUX/ISOLINUX orientado a arranques desde floppy y cd.
- Arrancar en red (DHCP y TFTP). PXELINUX, ETHERBOOT, COMBOOT.

[init@ Unidad II]\$

- Primer proceso del sistema.
- PID=1 (siempre).
- “Padre de todos los procesos”.
- Adopta y recoge a todos los procesos huérfanos.
- No se puede morir.
- Archivo de configuración: /etc/inittab
- Niveles de ejecución: “runlevels”.

[/etc/inittab@Unidad II]\$

- 0 Sistema
- 1 Monousuario
- 2 Multiusuario sin NFS
- 3 Operación Normal
- 4 Sin uso
- 5 Mismo que 3 pero con X Windows
- 6 Rearranque del sistema

[Scripts rc@Unidad II]\$

- /etc/rc.d/rc3.d
- /etc/rc.d/initd
- S arranca servicio
- K parar

- lrwxrwxrwx 1 root root 9 Oct 20 20:05 K24random -> ../init.d/random
lrwxrwxrwx 1 root root 8 Oct 20 18:23 S01fbset -> ../init.d/fbset

- service servicio start | stop | restart | satus

```
[/etc/rc.d/rc.local@Unidad II]$
```

```
#!/bin/sh
```

```
#Limpia de archivos temporales
```

```
echo "Eliminando archivos temporales..."
```

```
find /tmp -type f -mtime +1 | xargs -r rm
```

```
find /tmp -type d -mtime +1 | xargs -r rmdir
```

[Autenticación@Unidad I]\$

- La aplicación pide una contraseña al usuario
- Se comprueba `/etc/nsswitch.conf` para ver donde está la contraseña
- Se lee `/etc/passwd` `/etc/shadow`
- Se comprueba esa entrada
- La aplicación compara las cadenas y si son iguales se tiene acceso

[Login@Unidad II]\$

- Login: usuario
- Password: *****

- [roge@antares /home/roge]\$
- [root@antares /root]#

- [root@antares /root]# **logout** ↵
- [root@antares /root]# **shutdown -r now** ↵

[Parada@Unidad I]\$

- Se invoca al comando shutdown o reboot
- Se llama init para cambiar el nivel de ejecución a 6
- Se ejecutan los scripts del nivel 6 con stop
- Los búferes de salida se escriben en disco (sync)
- Los sistemas de archivos se desmontan
- El sistema se para

[Unidad II@presentación]#

2.1 Administración de usuarios

2.2 La línea de comandos (shell) y utilerías

2.3 Arranque y parada

2.4 Sistema de archivos

2.5 Servicios del núcleo del sistema

2.6 Kernel de Linux

2.7 Seguridad

[Sistema de archivos@Unidad II]\$

- La forma en como se administran la información a través del SO recibe el nombre de sistema de archivo (File System).
- En Linux todo es un archivo. Los dispositivos, los sockets, las tuberías, etc.
- Existen archivos de E/S estándar `STD_OUT` (pantalla), `STD_IN` (teclado), `STD_ERR` (pantalla) que pueden ser redireccionados.

[Sistema de archivos@Unidad II]\$

- Nativo linux ext2
- Intercambio de memoria swap
- CD-ROM iso9660
- Monitorización del sistema proc
- Sistema de archivos de red (NFS) nfs
- FAT básico msdos
- VFAT (Windows 95) vfat
- Minix minix
- System V sysv
- BSD Fast File System bsd

[Sistema de archivos@Unidad II]\$

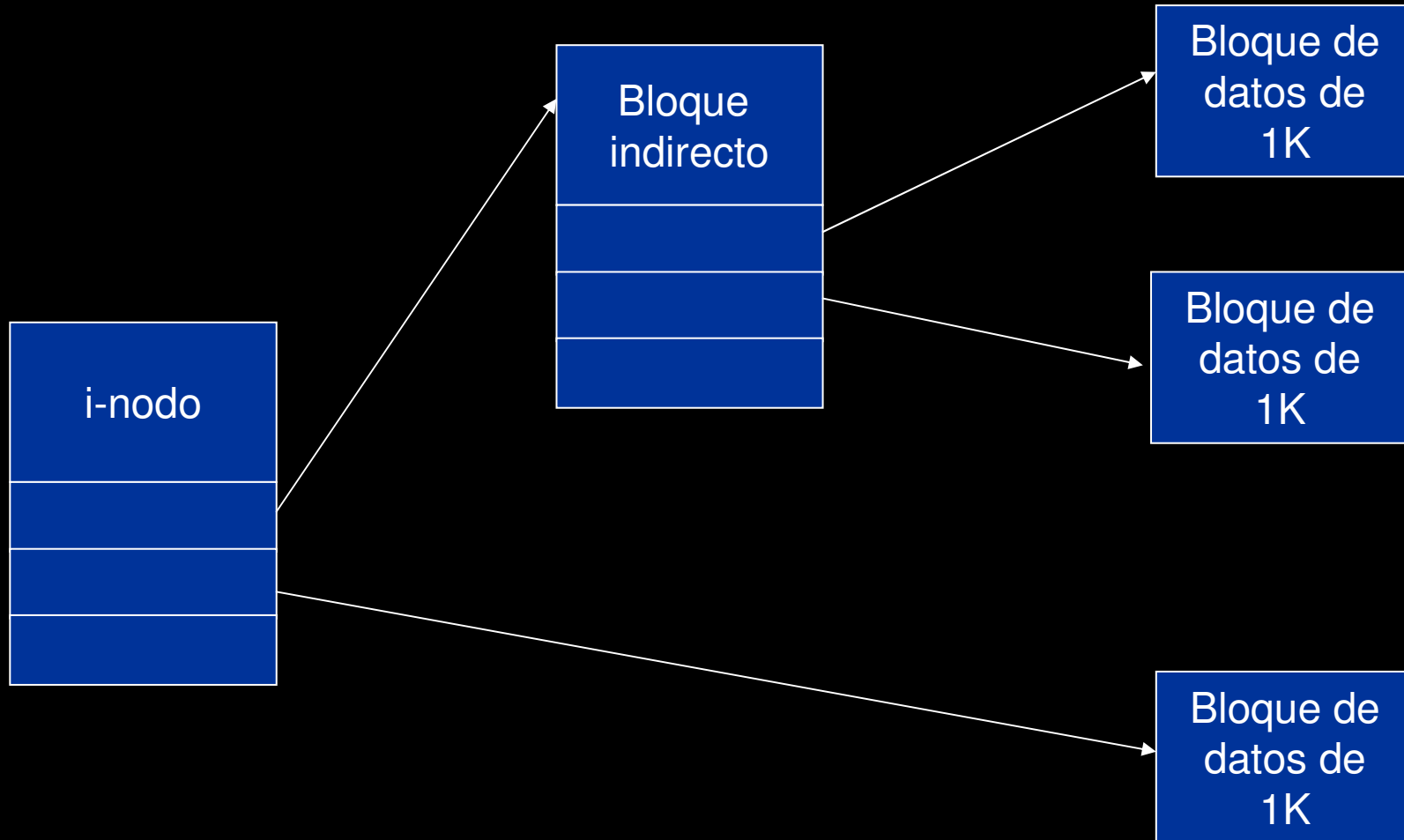
Nombre(8)	Ext(3)	A(1)	Reservado(10)	H(2)	F(2)	PNB(2)	Tamaño (4)
-----------	--------	------	---------------	------	------	--------	------------

MS-DOS

Ni(2)	Nombre(14)
-------	------------

Linux

[Sistema de archivos@Unidad II]\$



[Árbol de directorios@Unidad II]\$

- /bin binarios
- /sbin binarios de root
- /dev dispositivos
- /etc archivos de configuración
- /home archivos de usuarios
- /proc archivos del kernel
- /tmp archivos temporales

[Árbol de directorios@Unidad

- /boot Imágenes del kernel
- /mnt montaje de directorios (/floppy, /cdrom)
- /lib Bibliotecas y módulos
- /root información sobre el administrador
- /usr programas
- /var bitácoras del sistema

[/proc@Unidad II]\$

- /proc/cpuinfo tipo de CPU
- /proc/interrupts tabla IRQ
- /proc/meminfo información sobre la memoria
- /proc/modules igual que lsmod
- /proc/pci periféricos con soporte PCI
- /proc/net/arp tabla ARP
- /proc/net/dev dispositivos de red

[Dispositivos de bloque@Unidad II]\$

- Brw-rw----- 1 root disk 3, 0 May 5 1998
/dev/hda
- 3 major number
- 0 minor number
- 3 controlador de dispositivo pero diferente puerto

[Dispositivo de caracter@Unidad II]\$

- Crw----- 1 root tty 4, 64 May 5 1998
/dev/ttyS0
- P pipe
- R Lectura 4
- W Escritura 2
- X Ejecución 1

[Unidad II@presentación]#

2.1 Administración de usuarios

2.2 La línea de comandos (shell) y utilerías

2.3 Arranque y parada

2.4 Sistema de archivos

2.5 Servicios del núcleo del sistema

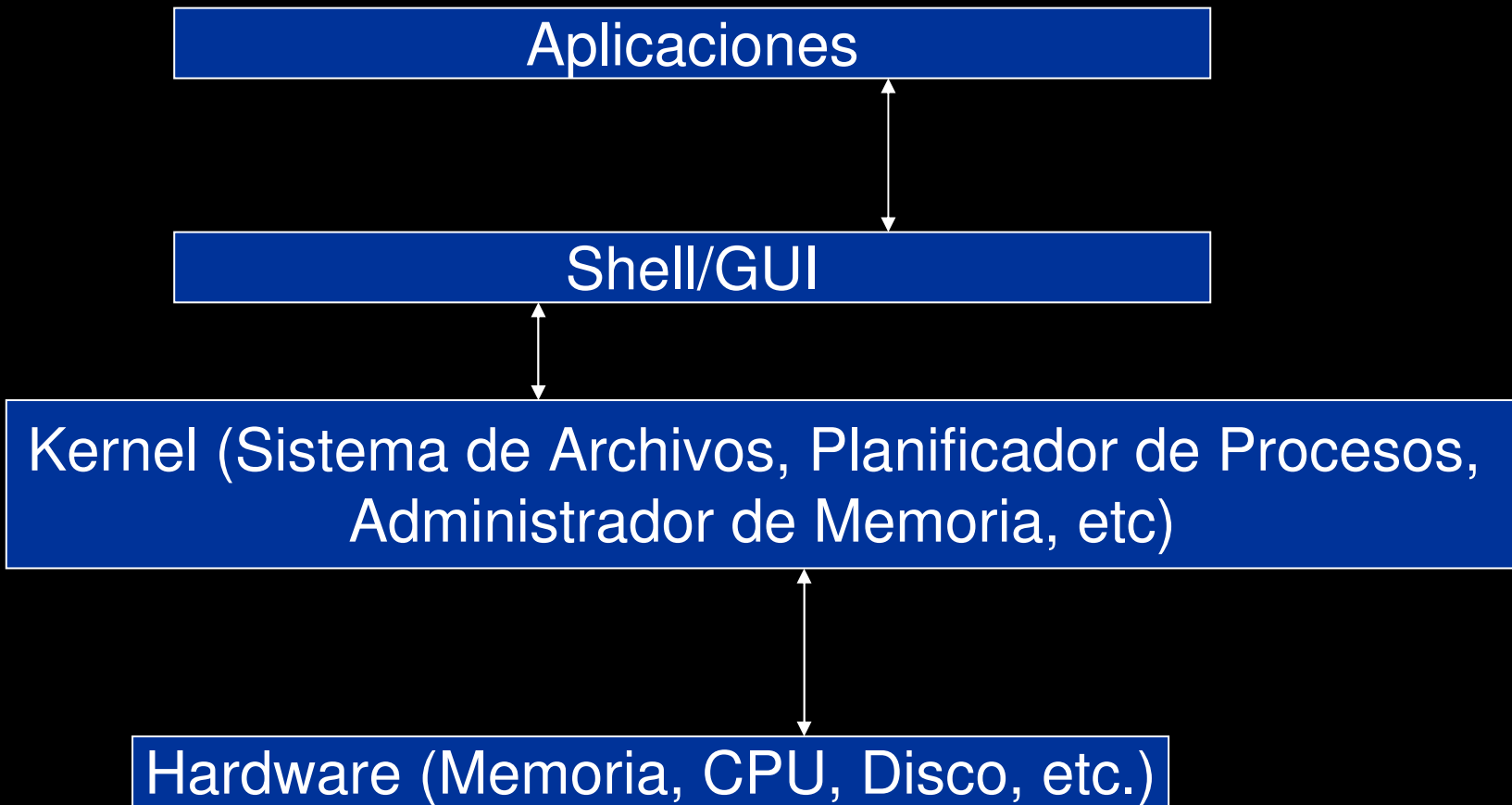
2.6 Kernel de Linux

2.7 Seguridad

[Kernel@Unidad II]\$

- Búsqueda e inicialización de hardware.
- Inicialización de la pila TCP/IP.
- Monta sistema de ficheros raíz (“root filesystem”).
- Prepara la consola, terminales virtuales, etc.
- Ejecuta el primer proceso del sistema, “/sbin/init”.

[Kernel@Unidad II]\$



[Mensajes del kernel]@Unidad II

- Todos los mensajes que genera el kernel se pueden ver con la orden “dmesg”.
- `dmesg | grep quotas`

```
VFS: Diskquotas version dquot_5.6.0  
initialized
```

[Unidad II@presentación]#

2.1 Administración de usuarios

2.2 La línea de comandos (shell) y utilerías

2.3 Arranque y parada

2.4 Sistema de archivos

2.5 Servicios del núcleo del sistema

2.6 Kernel de Linux

2.7 Seguridad

[Kernel@Unidad II]\$

- Impar inestable 2.1, 2.3
- Par estable 2.0, 2.2
- Última versión estable 2.6.19.2 (13/01/2006)

- `/etc/src/linux`

- Desempacar el kernel.

- `make xconfig` gráfico

[Configuración Kernel@Unidad II]\$

- Al Kernel se le pueden pasar una serie de parámetros para:
 - Configurar dispositivos (discos duros, tarjetas de red, puertos serie, memoria (mem=XXX). . .)
 - Indicar sistema de archivos raíz (root=/dev/XXX)
 - Condiciones especiales: monousuario, distinto init (init=XXX). . .
 - Cada cargador tiene su manera de pasarle parámetros al Kernel
 - Dentro del Kernel, los parámetros con los que se ha arrancado se pueden ver en /proc/cmdline

[Configuración Kernel@Unidad II]\$

- LILO linux single modo usuario
- LILO mem=128

- GRUB kernel (hd0,0)/vmlinuz root=/dev/hda1
single

- modconf asistente para módulos
- lsmod, rmmod, insmod, depmod, modprobe

[Instalación del Kernel@Unidad II]

- `make config`
- `make xconfig`

- Escoger los módulos y opciones a instalar

- `make dep; make clean; make zImage; make modules; make modules_install`

[Instalación@Unidad II]\$

- make bzImage
- Kernel compilado:
/usr/src/linux/arch/i386/boot/zImage,
- Mapas: /usr/src/linux/System.map
- /boot/vmlinuz-x.x.x
- System.map mensajes

[Cargador de arranque@Unidad II]\$

Default=/boot/vmlinuz-x.x.x

Image=/boot/vmlinuz-x.x.x

Label= Linux compilado

Root=/dev/hda1

- Parcheado del kernel
- `patch -p0 < parche-2.2.11`

[Unidad II@presentación]#

2.1 Administración de usuarios

2.2 La línea de comandos (shell) y utilerías

2.3 Arranque y parada

2.4 Sistema de archivos

2.5 Servicios del núcleo del sistema

2.6 Kernel de Linux

2.7 Seguridad

[Seguridad@Unidad II]\$

- La seguridad viene inmersa dentro de cada aplicación o en su defecto.
- Verificar siempre los logs del sistema
- Tener actualizado siempre los sistemas con las últimas alertas de seguridad.
- No existen sistemas completamente seguros

[Seguridad@Unidad II]\$

- Configurar adecuadamente los permisos de archivos para todos los usuarios
- Configurar contraseñas cambiantes
- No utilizar la cuenta de root como usuario
- Utilizar enlaces simbólicos y espacio de cuotas para los usuarios
- No activar servicios que no se utilicen
- Cerrar todos los puertos

[Seguridad@Unidad II]\$

- No instalar GUIs a menos que sea extremadamente necesario.
- Instale un Firewall vía SW o HW
- Utilizar protocolos cifrados: HTTPS, SFTP, SSH, etc.
- Hacer respaldo de la información constantemente