There are no translations available.
Descubre en este artículo la estructura del sistema de archivos en Linux..

La estructura del sistema de archivos en Linux

1.- Introducción e historia

Un sistema Linux reside bajo un árbol jerárquico de directorios muy similar a la estructura del sistema de archivos de plataformas Unix.

Originariamente, en los inicios de Linux, este árbol de directorios no seguía un estándar cien por cien, es decir, podíamos encontrar diferencias en él de una distribución a otra.

Todo esto hizo pensar a cierta gente* que, posteriormente, desarrollarían el proyecto FHS (Filesystem Hierarchy Standard, o lo que es lo mismo: Estándar de Jerarquía de Sistema de Ficheros) en otoño de 1993.

* Rusty Russell, Daniel Quinlan y Christopher Yeoh, creadores del estándar FHS entre otras personas.

2.- FHS

FHS se define como un estándar que detalla los nombres, ubicaciones, contenidos y permisos de los archivos y directorios, es decir, un conjunto de reglas que especifican una distribución común de los directorios y archivos en sistemas Linux.

Como se ha mencionado, se creo inicialmente para estandarizar la estructura del sistema de archivos para sistemas GNU/Linux y más tarde, en torno al año 1995, también para su aplicación en sistemas Unix.

FHS no es más que un documento guía, es decir, cualquier fabricante de software independiente o cualquier persona que decida crear una nueva distribución GNU/Linux, podrá aplicarlo o no a la estructura del sistema de archivos, con la ventaja de que si lo integra en el sistema, el entorno de éste será mucho más compatible con la mayoría de las distribuciones.

Es importante saber que el estándar FHS es en cierto modo flexible, es decir, existe cierta libertad en el momento de aplicar las normas. De ahí que existan en la actualidad leves diferencias entre distribuciones GNU/Linux.

Objetivos principales de FHS

- Presentar un sistema de archivos coherente y estandarizado.
- Facilidad para que el software prediga la localización de archivos y directorios instalados.
- Facilidad para que los usuarios prediga la localización de archivos y directorios instalados.
- Especificar los archivos y directorios mínimos requeridos.

El estándar FHS está enfocado a

- Fabricantes de software independiente y creadores de sistemas operativos, para que establezcan una estructura de ficheros lo más compatible posible.
- Usuarios comunes, para que entiendan el significado y el contendido de cada uno de los elementos del sistema de archivos.

Además, **FHS manifiesta algunas diferencias entre varios tipos de archivos que puede haber en el sistema** :

- Archivos compartibles y no compartibles.

Ficheros que son propios de un host determinado y, archivos que pueden compartirse entre diferentes host.

Ejemplo:

-

- Archivos compartibles: los contenidos en /var/www/html (que es el *DocumentRoot* por defecto del servidor Web Apache. Donde se almacena inicialmente el index.html de bienvenida).

- Archivos no compartibles: los contenidos en /boot/grub/ (Subdirectorio donde se ubican los ficheros del gestor de arranque GRUB).
- Archivos estáticos y variables.

Ficheros que no cambian sin la interacción de un administrador del sistema y, archivos que cambian sin la interacción de un administrador del sistema.

Para comprender mejor estos dos tipos, imaginemos los ficheros log (archivos de bitácora) del sistema. Estos cambian sin la intervención del administrador; en consecuencia estos son del tipo variables.

Los demás archivos son estáticos. No cambian su contenido ni tamaño a menos que lo autorice el administrador del sistema (o sea el propio quien lo modifique, por supuesto).

o Archivos estáticos: /etc/password, /etc/shadow.

o Archivos variables: /var/log/messages (log de mensajes generados por el kernel del sistema).

3.- Todo en Linux es un archivo

Cierto, todo en un sistema Linux es un archivo, tanto el Software como el Hardware. Desde el ratón, pasando por la impresora, el reproductor de DVD, el monitor, un directorio, un subdirectorio y un fichero de texto.

De ahí vienen los conceptos de montar y desmontar por ejemplo un CDROM.

El CDROM se monta como un subdirectorio en el sistema de archivos. En ese subdirectorio se ubicará el contenido del disco compacto cuando esté montado y, nada cuando esté desmontado.

Para ver que tenemos montado en nuestra distribución GNU/Linux, podemos ejecutar el comando *mount*.

Este concepto es muy importante para conocer como funciona Linux.

En apartados posteriores, veremos donde ubica Linux los elementos Hardware del PC en el sistema de ficheros.

NOTA	: podemos acceder a los di

Hay que saber que si editamos, por ejemplo, un fichero vinculado a un elemento Hardware, seguramen

En definitiva, no es aconsejable leer o abrir y mucho menos modificar archivos vinculados a elementos

4.- Organización de sistema de archivos según FHS

4.1.- El directorio raíz

Todo surge a partir del directorio raíz (/).

El contenido de este directorio debe ser el adecuado para reiniciar, restaurar, recuperar y/o reparar el sistema, es decir, debe proporcionar métodos, herramientas y utilidades necesarias

para cumplir estas especificaciones.

Además, es deseable que se mantenga los más razonablemente pequeño como sea posible por cuestión de funcionamiento y de seguridad.

Por último, este debe que ser el único directorio en el nivel superior del árbol jerárquico de archivos y, tiene que ser imposible moverse más allá del mismo.

Es el último origen.

Vemos, por ejemplo, un listado de su estructura:

Contenido del directorio raíz

o **/bin**

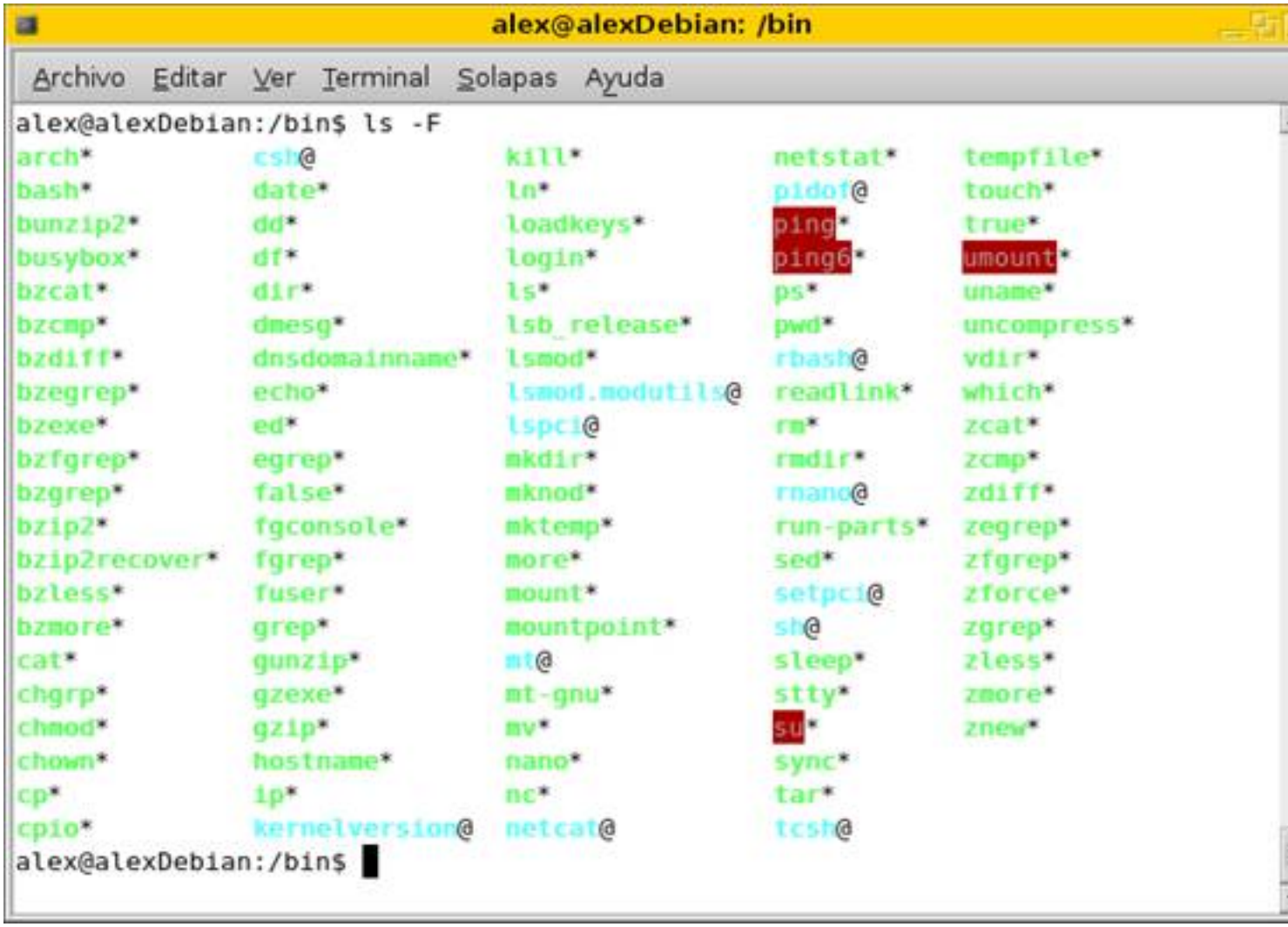
En este directorio se ubica el código binario o compilado de los programas y comandos que

pueden utilizar todos los usuarios del sistema.

La denominación es clara, bin de BINARY (binario en castellano).

No debe haber subdirectorios en /bin.

Estos son, por ejemplo, algunos comandos contenidos en /bin.



* La @ al lado del nombre de un fichero representa un enlace simbólico

o /**boot**

Este directorio contiene todo lo necesario para que funcione el proceso de arranque del sistema.

/boot almacena los datos que se utilizan antes de que el kernel comience a ejecutar programas en modo usuario* .

El núcleo del sistema operativo (normalmente se guarda en el disco duro como un fichero imagen llamado *vmlinuz-versión _ núcleo*) se debe situar en este directorio o, en el directorio raíz.

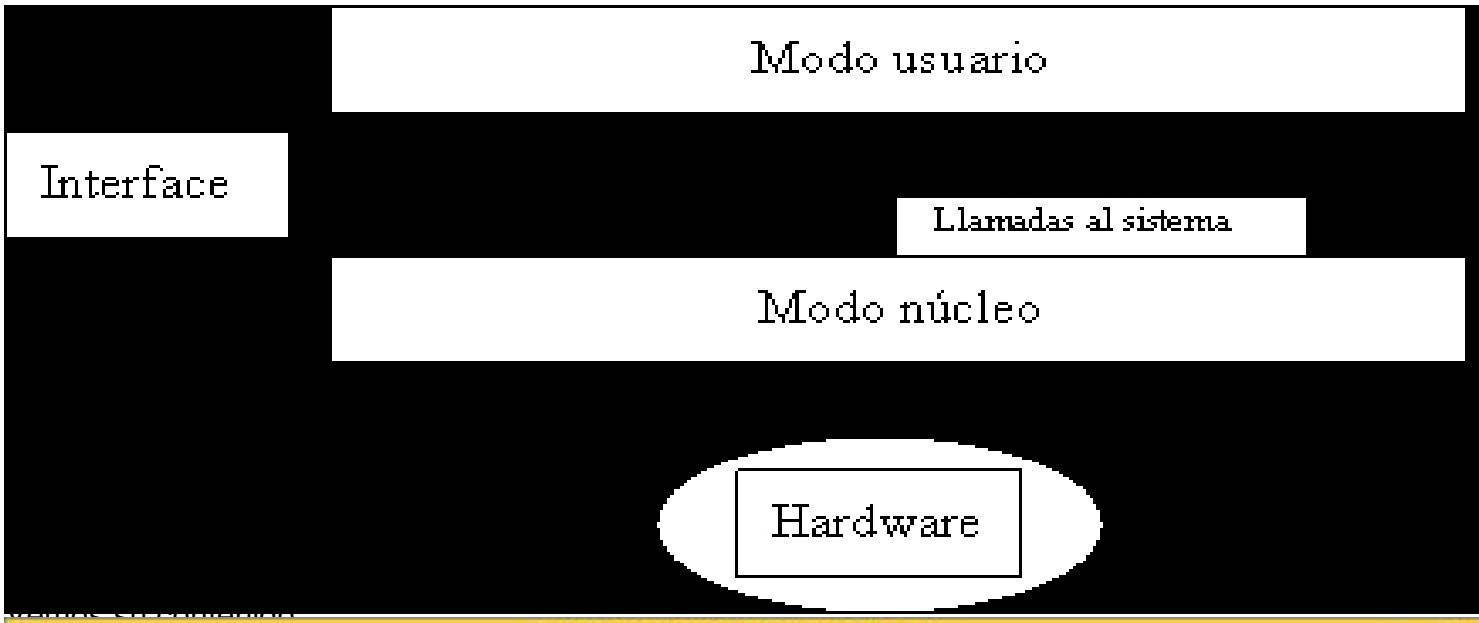
*El núcleo tiene la capacidad de crear dos entornos o modos de ejecución totalmente separados.

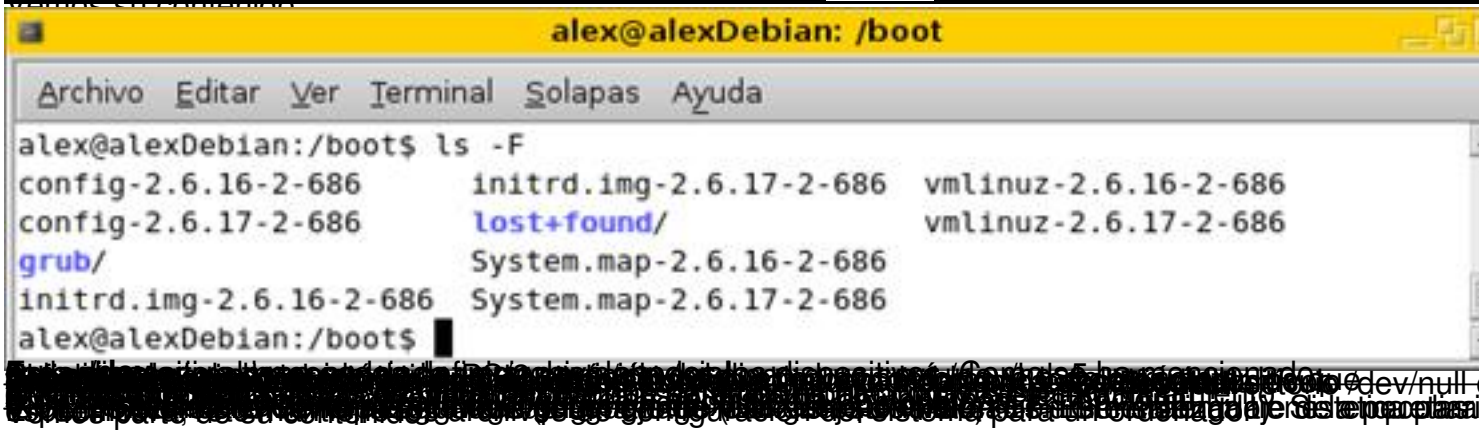
Uno de ellos está reservado para el propio kernel, denominado el modo núcleo ; y el otro está reserva

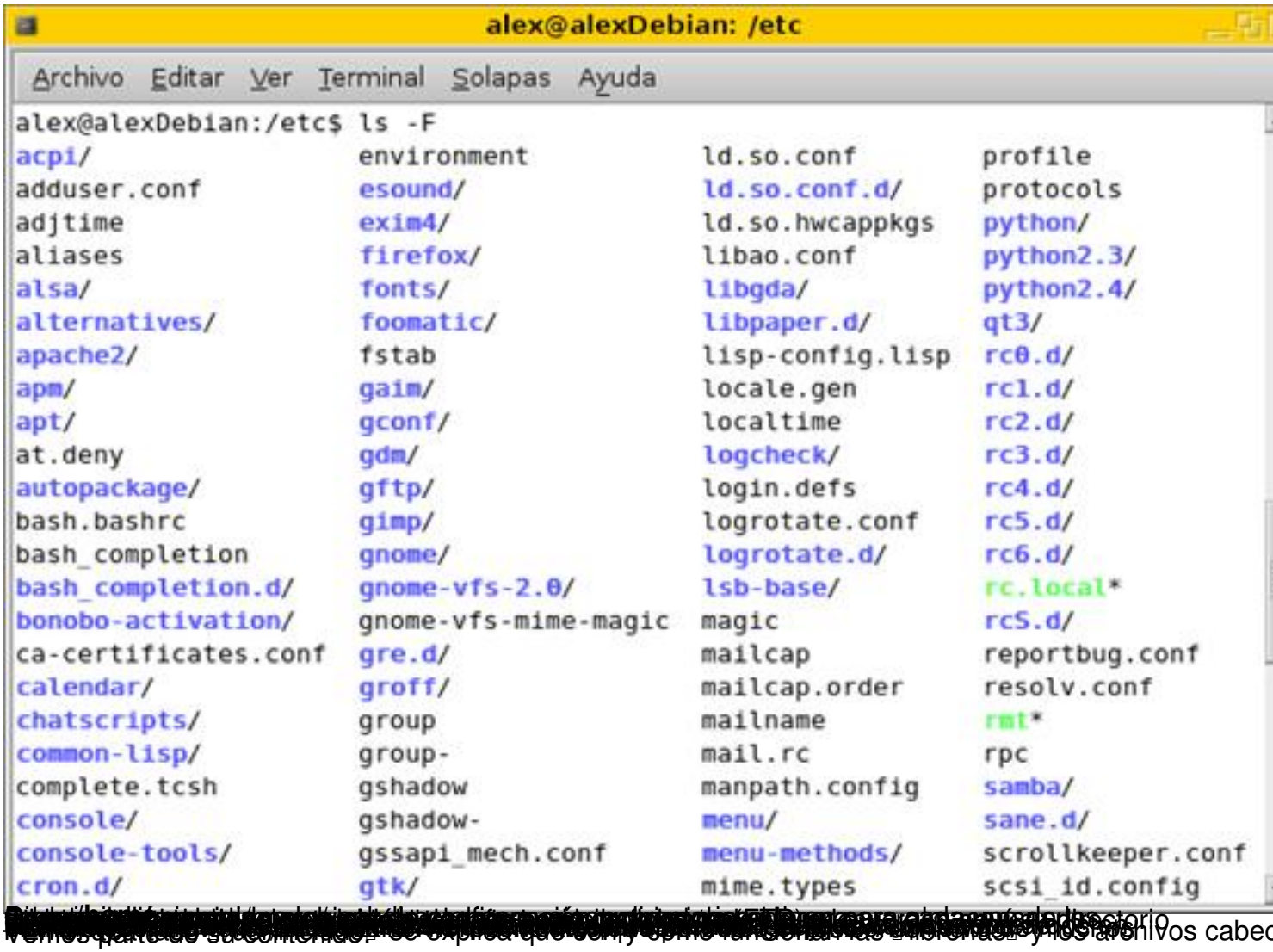
Realmente se crean dos entornos totalmente separados, es decir, cada uno tiene su propia zona de me

Démonos cuenta que esta técnica ofrece mucha seguridad y estabilidad al sistema.

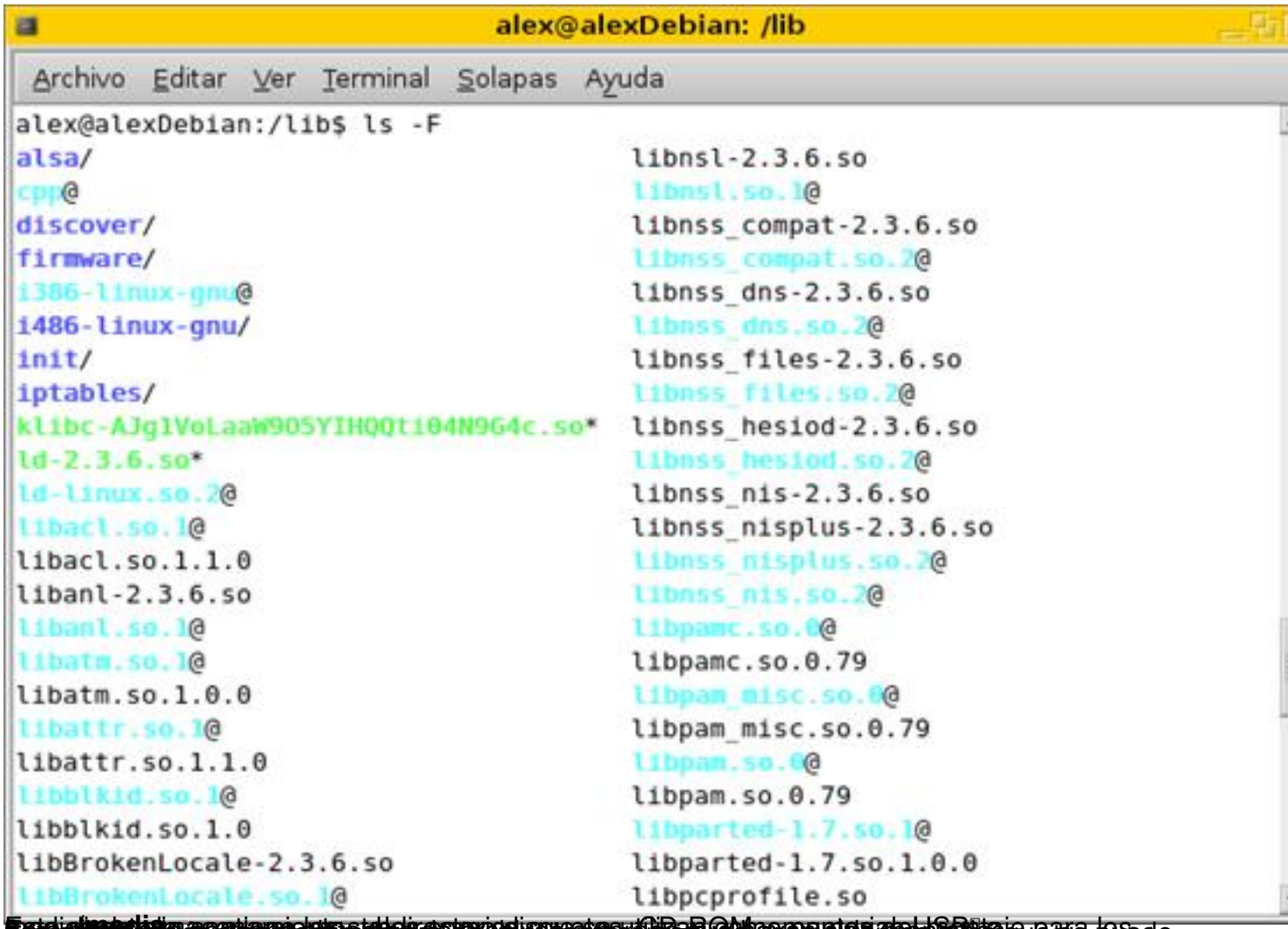
Cuando un proceso del modo usuario necesita recursos del modo kernel (por ejemplo, acceder a l


Modo usuario
Interface
Llamadas al sistema
Modo núcleo
Hardware

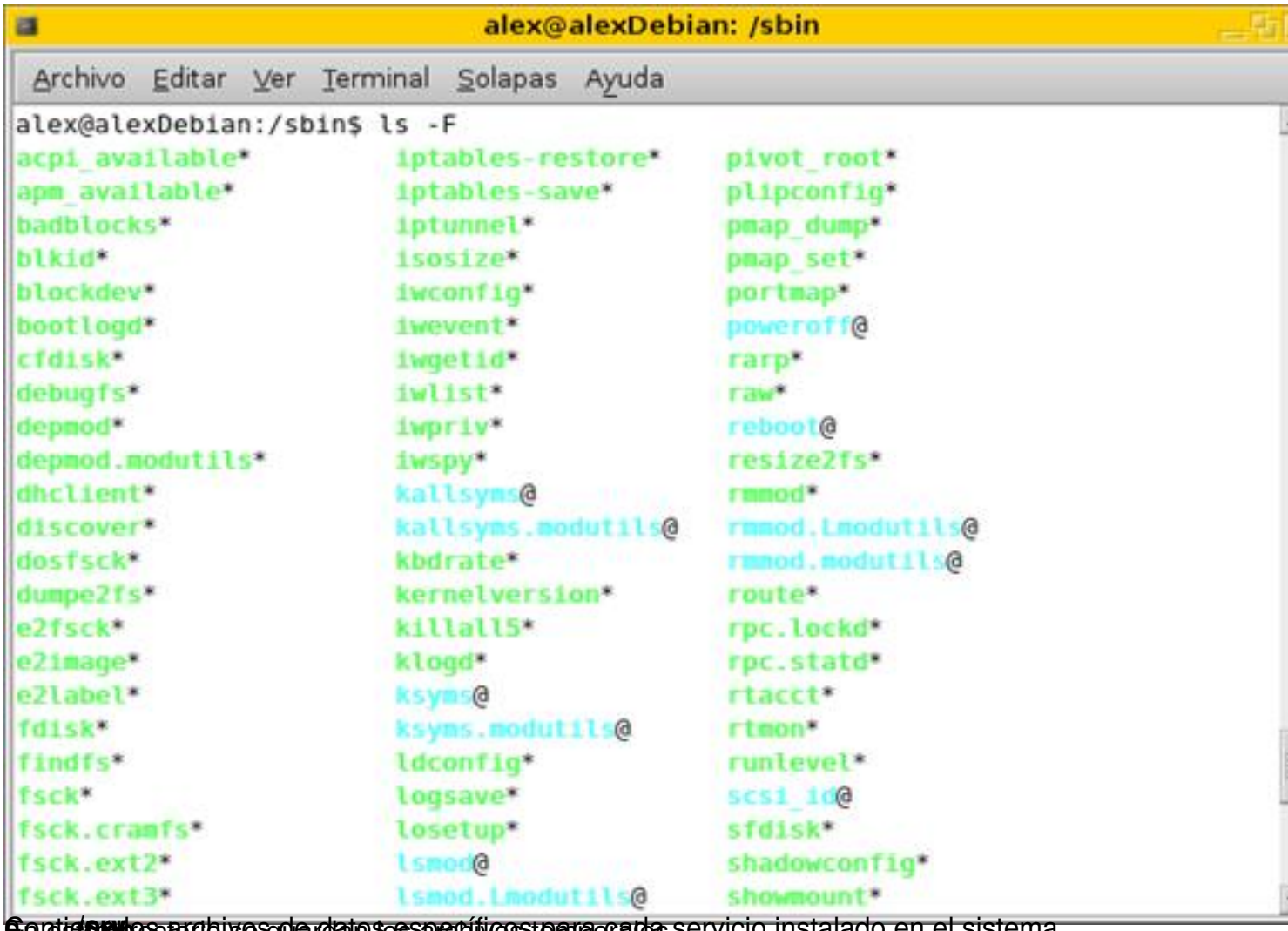

alex@alexDebian: /boot
Archivo Editar Ver Terminal Solapas Ayuda
alex@alexDebian:/boot$ ls -F
config-2.6.16-2-686 initrd.img-2.6.17-2-686 vmlinuz-2.6.16-2-686
config-2.6.17-2-686 lost+found/ vmlinuz-2.6.17-2-686
grub/ System.map-2.6.16-2-686
initrd.img-2.6.16-2-686 System.map-2.6.17-2-686
alex@alexDebian:/boot$



[redacted] archivos cabec


alex@alexDebian: /lib
Archivo Editar Ver Terminal Solapas Ayuda
alex@alexDebian:/lib$ ls -F
alsa/
cpp@
discover/
firmware/
i386-linux-gnu@
i486-linux-gnu/
init/
iptables/
klibc-AJg1VoLaaW9O5YIHQQti04N9G4c.so*
ld-2.3.6.so*
ld-linux.so.2@
libacl.so.1@
libacl.so.1.1.0
libanl-2.3.6.so
libanl.so.1@
libatm.so.1@
libatm.so.1.0.0
libattr.so.1@
libattr.so.1.1.0
libblkid.so.1@
libblkid.so.1.0
libBrokenLocale-2.3.6.so
libBrokenLocale.so.1@
libnsl-2.3.6.so
libnsl.so.1@
libnss_compat-2.3.6.so
libnss_compat.so.2@
libnss_dns-2.3.6.so
libnss_dns.so.2@
libnss_files-2.3.6.so
libnss_files.so.2@
libnss_hesiod-2.3.6.so
libnss_hesiod.so.2@
libnss_nis-2.3.6.so
libnss_nisplus-2.3.6.so
libnss_nisplus.so.2@
libnss_nis.so.2@
libpamc.so.0@
libpamc.so.0.79
libpam_misc.so.0@
libpam_misc.so.0.79
libpam.so.0@
libpam.so.0.79
libparted-1.7.so.1@
libparted-1.7.so.1.0.0
libpcprofile.so

[redacted]



En este directorio se guardan los archivos temporales.

Contiene los archivos de datos específicos para cada servicio instalado en el sistema.

4.2.- El directorio /usr

Es la segunda sección más grande o estructura jerár quica (después del directorio raíz) del sistema de ficheros.

Este directorio está pensado para almacenar datos que se puedan compartir con otros hosts.

Estos datos además deben ser inalterables, es decir, sólo de lectura.

Normalmente, este directorio tiene su partición propia.

Comúnmente, se almacena aquí el software instalado en el sistema.

Vemos su contenido:

Estructura de /usr

o **/usr/bin**

Éste es el directorio primario de comandos ejecutables del sistema.

/usr/bin alberga los archivos ejecutables vinculados al software instalado en el sistema.

o **/usr/include**

Linux está escrito en lenguaje C.

En C es posible utilizar funciones que ya estén predefinidas (como otros muchos lenguajes de programación) para incluirlas en el programa que estemos haciendo. Esta técnica se denomina programación modular.

Estas funciones se llaman comúnmente archivos cabecera (.h de header) y contienen las declaraciones externas de una librería.

La manera de incluir estos archivos cabecera en nuestro programa, es haciendo uso de la directiva *include*; de ahí la denominación del subdirectorio.

Ejemplo programa C:

#include

main()

{	
unsigned	int
char	resp
clrscr	();
do{	

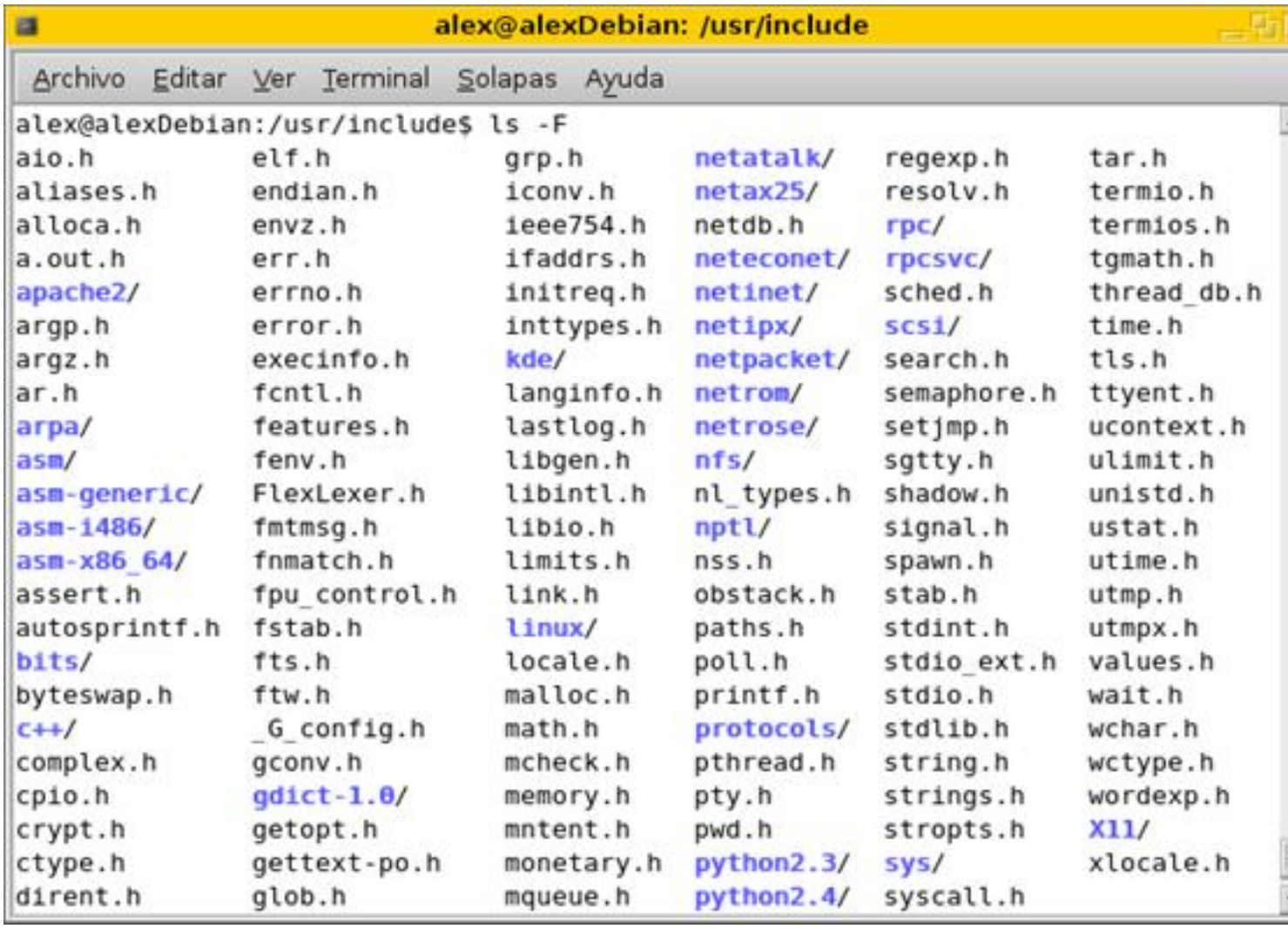
□ □

Todos estos ficheros cabecera (que necesite el software instalado en el sistema) se almacenan en este subdirectorio.

*Una librería no es más que un programa compilado, donde originariamente	se implemento el código fu

La declaración pública del conjunto de funciones de la librería reside en los archivos cabecera.

Vemos parte de su contenido:



o **/usr/lib**

Este directorio incluye librerías compartidas y ficheros binarios pensados para no ser ejecutados directamente por los usuarios del sistema.

o **/usr/local/**

/usr/local/ es para uso del administrador del sistema cuando instala software localmente. Puede usarse para programas y datos que son compartibles entre un grupo de máquinas

Este subdirectorio tiene una estructura similar a la del directorio /usr.

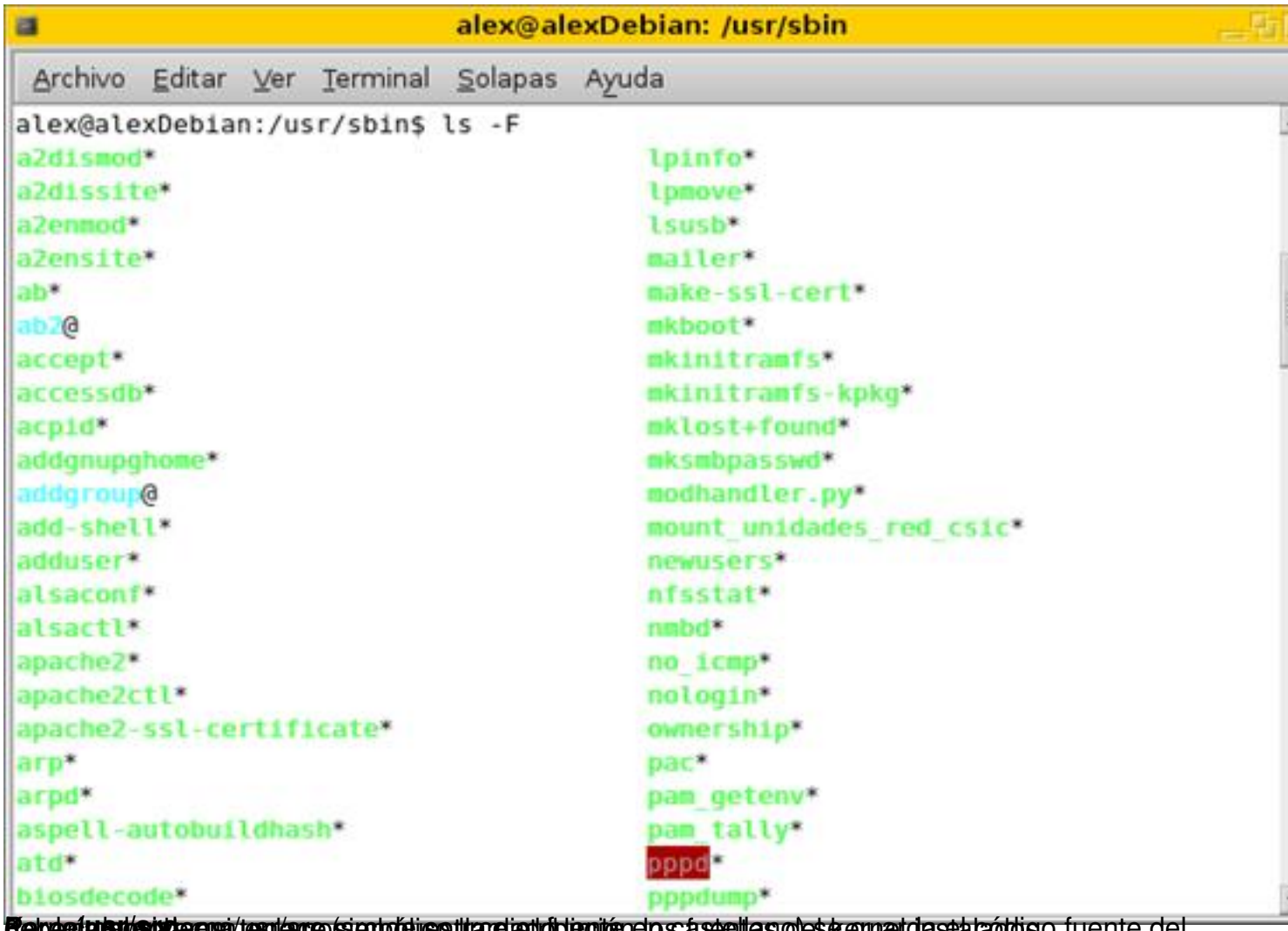
Vemos su contenido:

o **/usr/sbin**

Este directorio contiene comandos y programas no esenciales usados exclusivamente por el administrador de sistema.

Como se ha comentado, los comandos necesarios para la reparación, recuperación y otras funciones esenciales del sistema, se almacenan en /sbin.

Vemos parte de su contenido:



4.3.- El directorio /var

Este directorio va ha contener ficheros de datos variables y temporales, así como archivos spool (ficheros almacenados en "fila" en espera a ejecutarse, como por ejemplo colas de impresión).

Todos los log del sistema y los generados por los servicios instalados, se ubican dentro de la estructura jerárquica de /var. Esto quiere decir que el tamaño global de este directorio va ha crecer constantemente.

La utilidad de /var radica en poder detectar problemas para prevenirlos y solucionarlos.

Es aconsejable montar en una nueva partición este directorio. Si no se pudiera, es preferible ubicar /var fuera de la partición raíz y de la partición /usr.

Vemos su contenido:

Distribución de algunos subdirectorios de /var

o **/var/cache**

Subdirectorio pensado para albergar datos de aplicaciones en cache (usados en un espacio breve de tiempo).

El sistema de paquetes de Debian (apt-get), mantiene y almacena todos los paquetes que nos hemos instalado con el gestor de paquetes Apt-get. Por ejemplo, si ejecutamos:

```
alex@alexDebian:# apt-get install nmap
```

Debian se bajará de algún repositorio especificado en /etc/apt/sources.list el archivo *nmap_version.deb* ,
lo almacenará en /var/cache/apt y lo instalará desde esta ruta.

Posteriormente lo podemos borrar. Por defecto Debian almacena aquí todo los paquetes que nos hemos instalado con su gestor de paquetes Apt-get.

- **/var/lib**

Encontramos aquí información sobre el estado variable de las aplicaciones.

- **/var/lock**

Aquí se almacenan los ficheros que están bloqueados por el sistema.

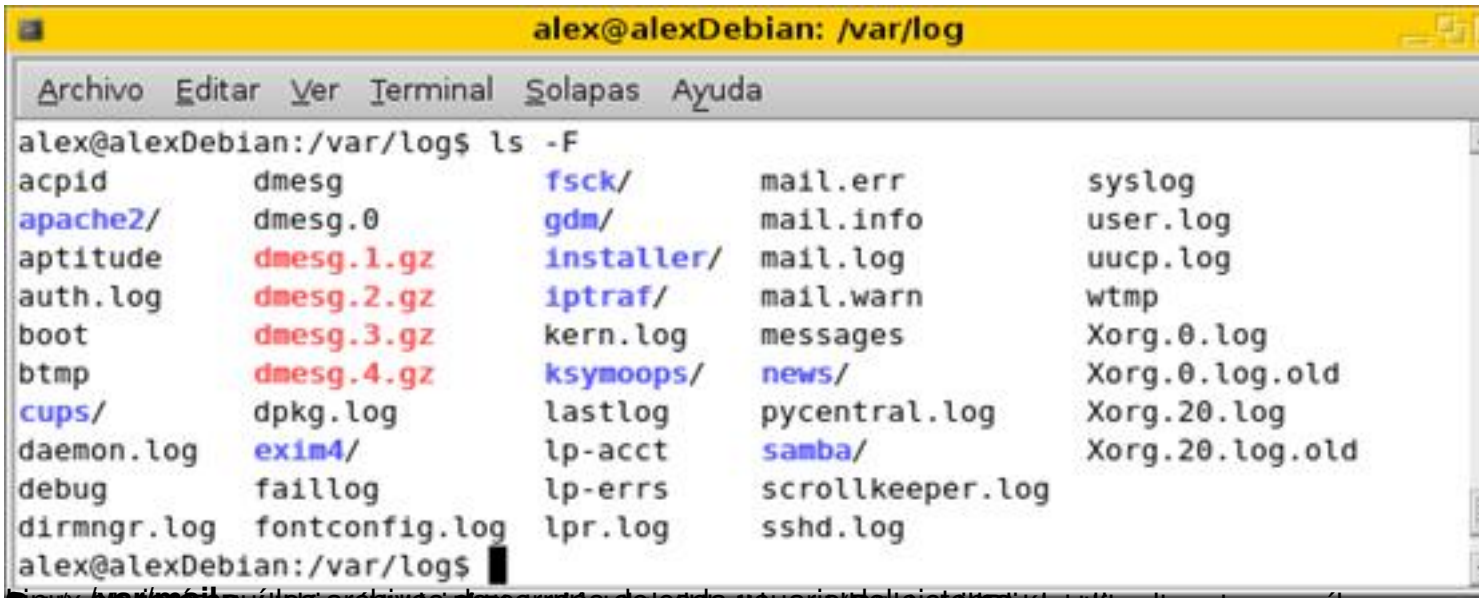
- **/var/log**

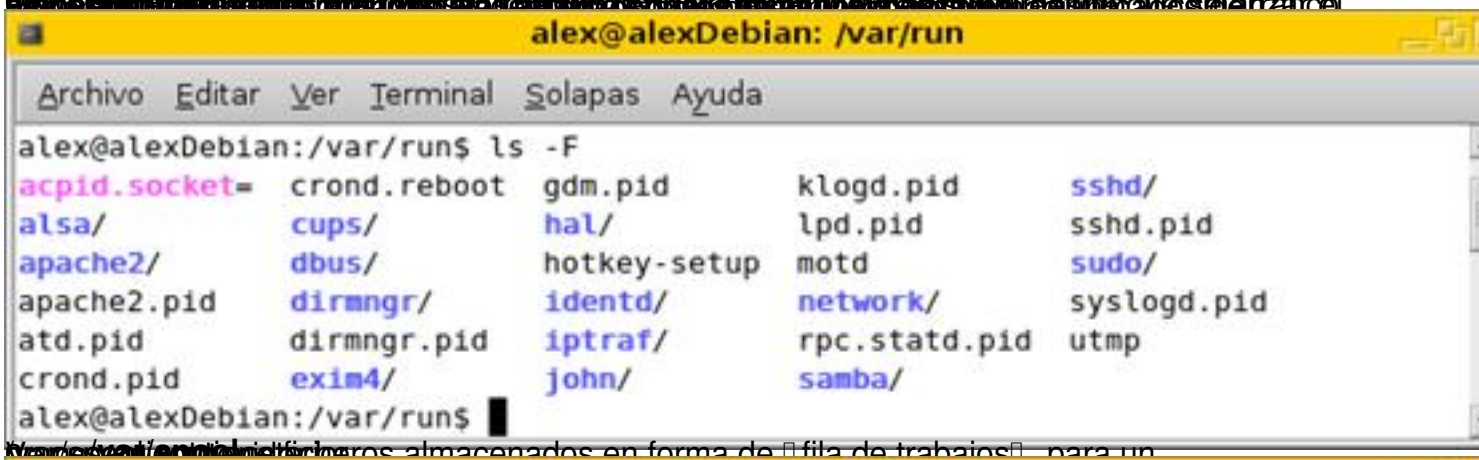
En /var/log se guardan los mensajes de registro generados por el sistema operativo y por diversos servicios.

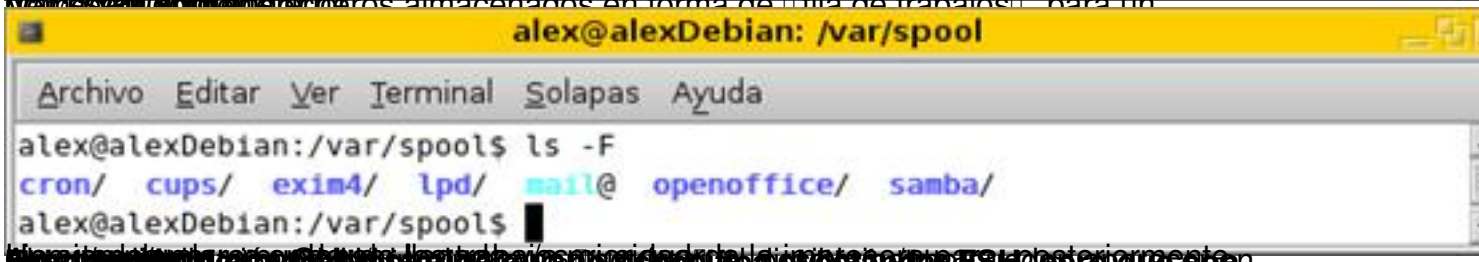
Por ejemplo:

En /var/log/messages son los logs generados por el kernel, en /var/log/httpd/access_log encontramos quien (desde que ip) está accediendo a nuestro servidor Web y, en /var/log/wtmp encontraremos todos los accesos y salidas en el sistema.

Vemos su contenido:







4.4.- Directorio /lost+found. "Perdidos y encontrados"

Las herramientas y utilidades para restaurar y/o reparar el sistema de archivos almacenan los datos en este directorio.

Es un espacio temporal donde se guardan los datos que se recuperan después de una caída del sistema.

Fijémonos que, normalmente en cada partición que creemos existirá un /lost+found en el nivel superior.

Por último, decir que este directorio existe sólo en distribuciones que tengan como sistemas de archivos **ext2** o **ext3**.

4.5.- Directorio /proc

/proc es un sistema de archivos virtual. Se genera y actualiza dinámicamente, es decir,

no se mantiene en el disco duro, se mantiene el la memoria RAM. Es el sistema quien lo crea y lo destruye.

Este directorio contiene información sobre los procesos, el núcleo e información relativa al sistema.

Vemos su contenido:

* Los subdirectorios identificados por un numero, corresponden a los PID de los procesos.

5.- Referencias

o http://www.wikipedia.es /

o http://www.redhat.com/

o http://www.pathname.com/fhs/

o http://www.microteknologias.cl/linux_fhs.html

o http://www.FreeUOC.org / Proyecto LPI

o http://es.tldp.org/

o http://www.hispafuentes.com/

o http://www-128.ibm.com/developerworks/linux/lpi/index.html

o Administración básica de GNU/Linux. Autor: José Ángel de Bustos Pérez

o G.L.U.P. Guía de Linux Para El Usuario. Larry Greenfield

o GNU/Linux: Instalación y primeros pasos. Autores: Matt Welsh, Phil Hughes. Julio de 2002

o Sistema Operativo GNU/Linux básico. UOC. Autores: Roger Borg Viñas. Francesc Aulí Llinas

o Apuntes Sistema Operativo Linux Universidad Politécnica de Madrid. Noviembre 2004

o Guía de referencia Debian. Autor: Osamu Auki. Febrero 2005

o Linux. 6ª Edición. Autores: David Bandel y Robert Napier. Editorial: Prentice Hall