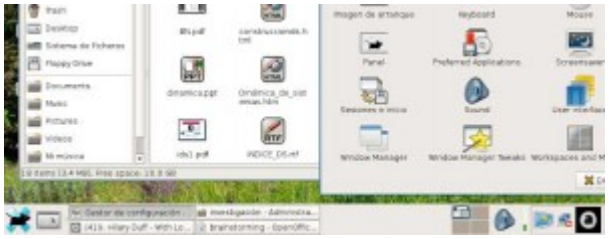
La estructura del sistema de archivos en Linux



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).

- Archivos estáticos: /etc/password, /etc/shadow.
- 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 dispositivos Hardware como si fueran archivos. Realmente son ficheros para Linux; pero no son archivos normales, son archivos binarios (o .exe para los que vengan de Windows).Hay que saber que si editamos, por ejemplo, un fichero vinculado a un elemento Hardware, seguramente sea totalmente ilegible y posiblemente quedará inutilizable y bloqueada nuestra línea de comandos (shell). Es más, corremos el riesgo de corromper los datos y dejar el sistema inestable.En definitiva, no es aconsejable leer o abrir y mucho menos modificar archivos vinculados a elementos Hardware y/o dispositivos, a menos que sepamos con toda seguridad lo que estemos haciendo.

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

- **/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

- **/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.

- **/dev**

Este directorio almacena las definiciones de todos los dispositivos. Como se ha mencionado, cada dispositivo tiene asociado un archivo especial.

Por ejemplo, el contenido de la sexta partición del disco duro será /dev/hda5.

El fichero asociado al ratón tipo PS/2 será /dev/psaux.

Además, es importante saber que los dispositivos pueden ser de bloque o de carácter.

Normalmente los dispositivos de bloque son los que almacenan datos y, los de carácter los que transfieren datos.

En definitiva, la estructura de este directorio es algo complejo. Podríamos dedicar otro artículo entero para poder explicar en profundidad el contenido y funcionamiento especifico del directorio /dev. De momento nos vasta con saber que Linux lo utiliza para asociar dispositivos (devices) con ficheros.

NOTA: El subdirectorio /dev/null es como un agujero negro. Esto es así, puesto que cualquier dato que se almacena aquí, desaparece. Es muy útil para redireccionar los errores por ejemplo:

```
alex@alexDebian# find / name pepe print
2>/dev/null
```

- **/etc**

El directorio /etc contiene archivos necesarios para configuración del sistema.

Archivos que son propios del ordenador y que se utilizan para controlar el funcionamiento diversos programas.

Deben ser ficheros estáticos y nunca pueden ser archivos binarios y/o ejecutables.

Algunos subdirectorios contenidos en /etc:

- X11. Subdirectorio para la configuración del sistema de ventanas. Es opcional.
- SGML. Subdirectorio para la configuración de SGML. Es opcional.
- Xml. Subdirectorio para la configuración de XML. Es opcional.

X11 Sistema de ventanas graficas originario de UNIX en su versión 11. Este sistema tiene la peculiaridad de ser totalmente independiente del sistema operativo. Es una estructura cliente-servidor.XML – eXtensible Markup Language (lenguaje de marcas extensible). Es un metalenguaje de etiquetas.SGML – Standard Generalized Markup Language (Lenguaje de Marcación Generalizado). Sistema para la organización y etiquetado de documentos.

En definitiva, /etc mantiene los archivos de configuración del sistema para un ordenador específico.

- **/home**

Directorio que contiene los subdirectorios que son directorios origen para cada uno de los usuarios del sistema.

Cada subdirectorio /home/user de cada usuario proporciona el lugar para almacenar sus ficheros, así como los archivos de

configuración propios de cada uno.

Es importante saber que también algunos servicios, y no solo usuarios, crean aquí su directorio origen, por ejemplo: el servicio de transferencia de ficheros (FTP).

El administrador tiene su propio directorio home, que es /root.

- **/lib**

El directorio /lib contiene librerías compartidas (similar a las dll□s para los usuarios de Windows) necesarias para arrancar el sistema y para los ficheros ejecutables contenidos en, por ejemplo, /bin.

Normalmente las librerías son ficheros binarios escritos en lenguaje C *.

También contiene módulos del kernel esenciales que permiten el funcionamiento de muchos elementos Hardware. Se ubicarán normalmente en /lib/modules/*versión-del-kernel/*.

- **/media**

Este directorio contiene los subdirectorios que se utilizan como puntos del montaje para los medios de almacenamiento, tales como disquetes, CD-ROM y memorias USB□s.

- **/mnt**

Este directorio contiene sistemas de archivos externos que hayan sido montados.

Las entidades que aparecen dentro de /mnt representan recursos externos a los que se puede acceder a través de este directorio.

- **/opt**

En este directorio (/opt de options, u opciones en castellano)

se suelen instalar complementos o add-ons de los programas.

Las aplicaciones crean un subdirectorio dentro de /opt denominado con el mismo nombre del programa.

- **/root**

Este directorio es el directorio /home del administrador del sistema (root).

- **/sbin**

Los programas y comandos que se utilizan para la administración del sistema se almacenan en /sbin, /usr/sbin y /usr/local/sbin.

/sbin únicamente contiene los ejecutables esenciales para el arranque, recuperación y reparación del sistema.

Todos estos directorios (/sbin, /usr/sbin y /usr/local/sbin) se utilizan con fines administrativos, por tanto, sólo puede ejecutar su contenido el administrador.

- **/srv**

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

- **/tmp**

En este directorio se guardan los archivos temporales.

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.

Estructura de /usr

- **/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.

- **/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:

```
#includemain(){
unsigned int num,masc;
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 fuente de las funciones que la componen.

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

- **/usr/lib**

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

- **/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.

- **/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.

- **/usr/src**

Por lo general, en /usr/src (src de source o, fuente en castellano) se guarda el código fuente del Kernel del sistema.

Para comprobar si tenemos en nuestra distribución los fuentes del kernel instalados, deberíamos ver un enlace simbólico llamado linux.

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.

Distribución de algunos subdirectorios de /var

- **/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.

- **/var/mail**

Linux enviará aquí los archivos de correos de cada usuario del sistema.

- **/var/run**

/var/run contiene archivos con información del sistema que lo describen desde que se arrancó. Generalmente, se borrará todos los archivos que cuelgan de este subdirectorio al comenzar el proceso de arranque.

Estos archivos con información del sistema son los llamados archivos identificados de procesos ó PID, que guardan el identificador del proceso (Process ID).

- **/var/spool**

/var/spool contiene ficheros almacenados en forma de fila de trabajos, para un procesamiento posterior.

Un ejemplo claro puede ser los trabajos que guarda la

impresora para, posteriormente, ejecutarlos por un orden de llegada y/o prioridad.

- **/var/tmp**

Algunos datos temporales se almacenan aquí y que, posiblemente, pueden aparecer en nuestra distribución GNU/Linux para no saturar el directorio /tmp.

Existen otra serie de directorios que no especifica el estándar FSH, pero que son importantes. Los vemos

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.

fuente:di.sld.cu