

Taller de **Linux**



Daniel Delgado

Ingeniería en Computación
delgado.jdds.cr@ieee.org



Principios de Linux

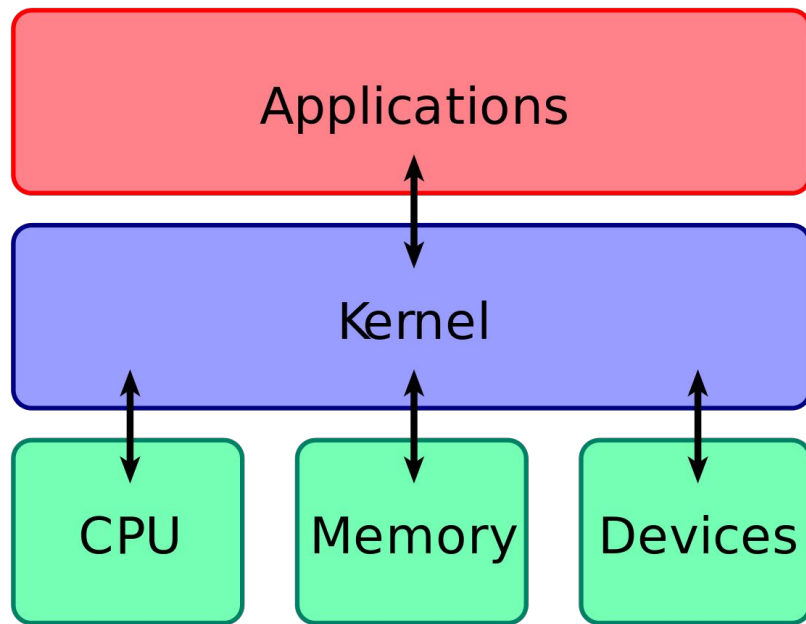


THE LINUX FOUNDATION



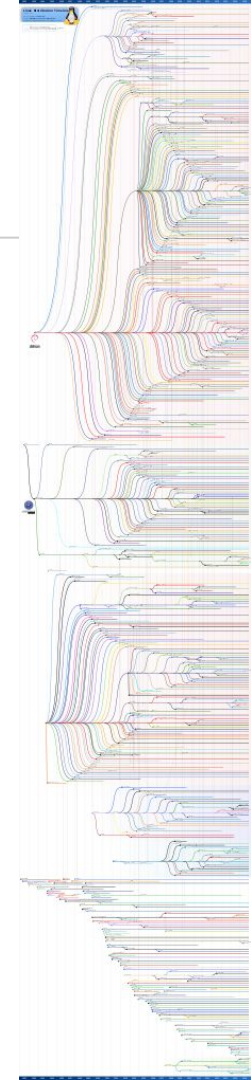
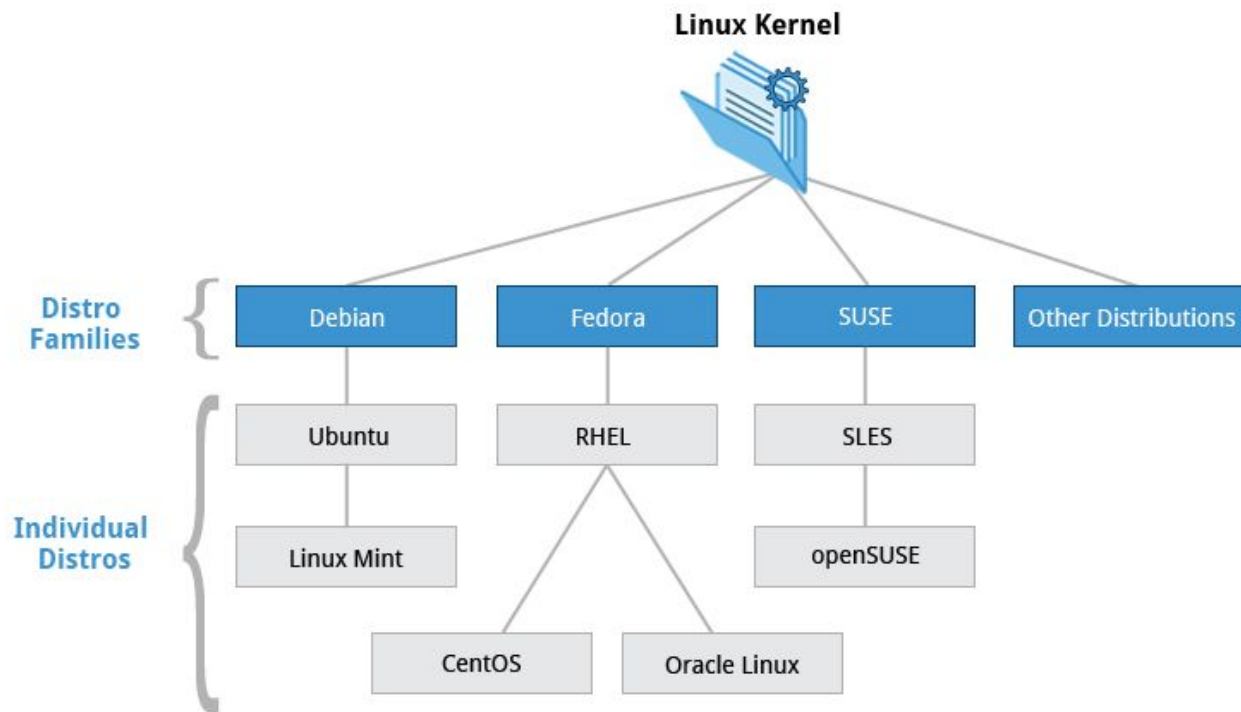


Kernel





Principales Distribuciones





850.000

teléfonos Android
ejecutando Linux



Arquitectura Android





Filosofía

- Versión gratuita y open-source de UNIX
- Nodo superior: root o “/”
- Componentes como objetos, simulando archivos



Partición y sistema de archivos

- Parte lógica del disco duro
- Sistemas de archivos para discos
convencionales: ext2, ext3, ext4, XFS,
Btrfs, JFS, NTFS, FAT, exFAT, etc.

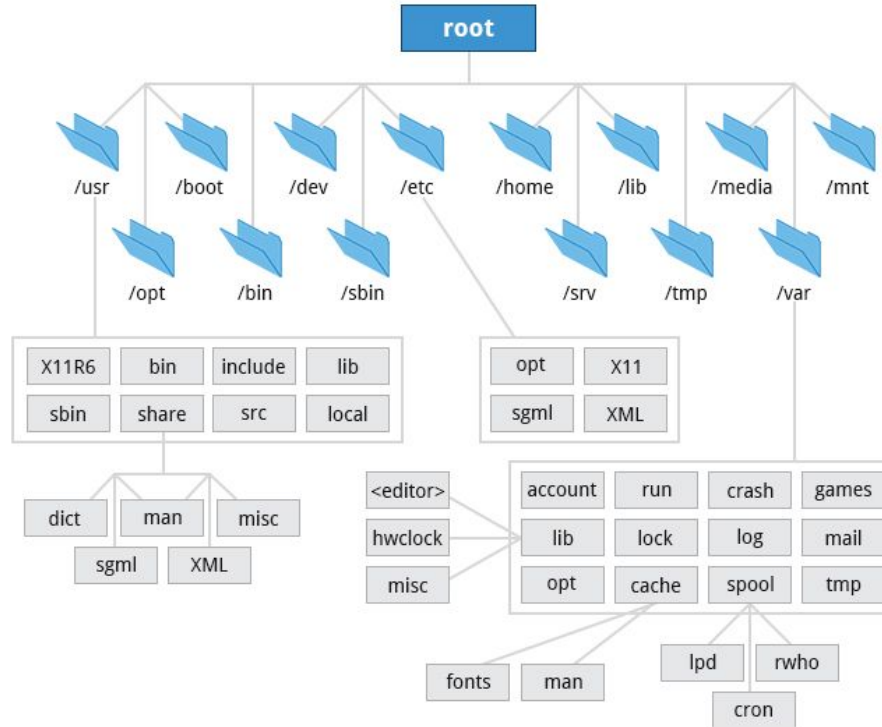


Windows vs. Linux

	Windows	Linux
Partición	Disk1	/dev/sda1
Tipo de sistema de archivos	NTFS/FAT32	EXT3/EXT4/XFS...
Parámetros de montaje	DriveLetter	MountPoint
Directorio base (donde se almacena el SO)	C drive	/

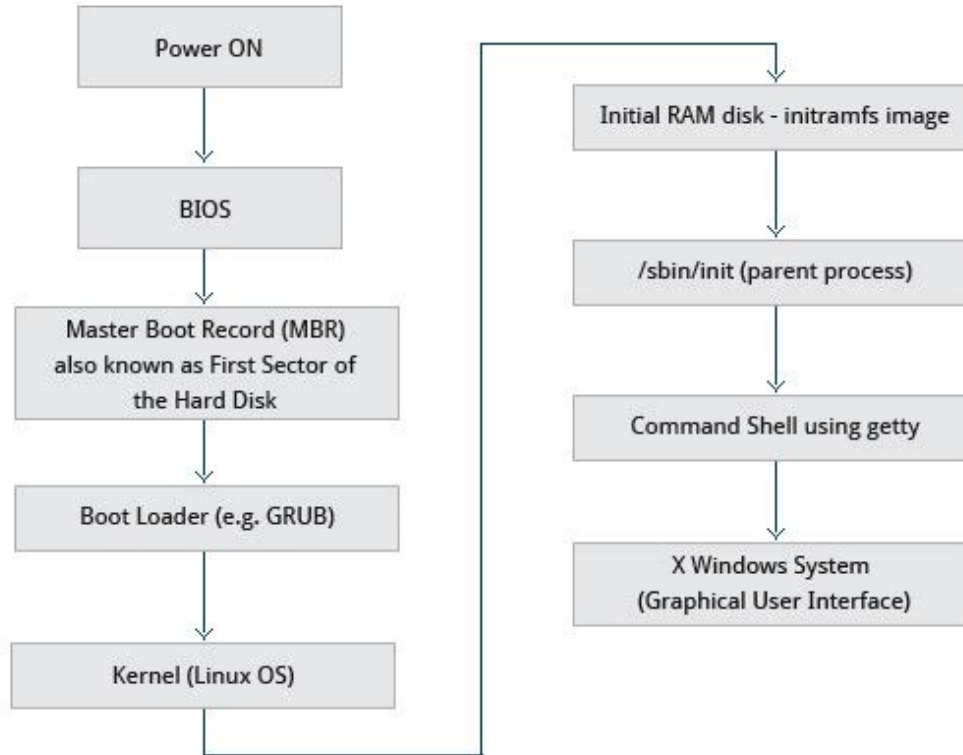


Estándar jerárquico del sistema de archivos Linux





Proceso de arranque





Beneficios

- Gratis
- Código abierto
- Fácil de usar
- Estable y rápido
- Pocos virus
- Actualizaciones cada 6 meses



Usuarios, Grupos y Permisos



Crear y eliminar usuarios

>> adduser [nombre usuario]

>> deluser [nombre usuario]



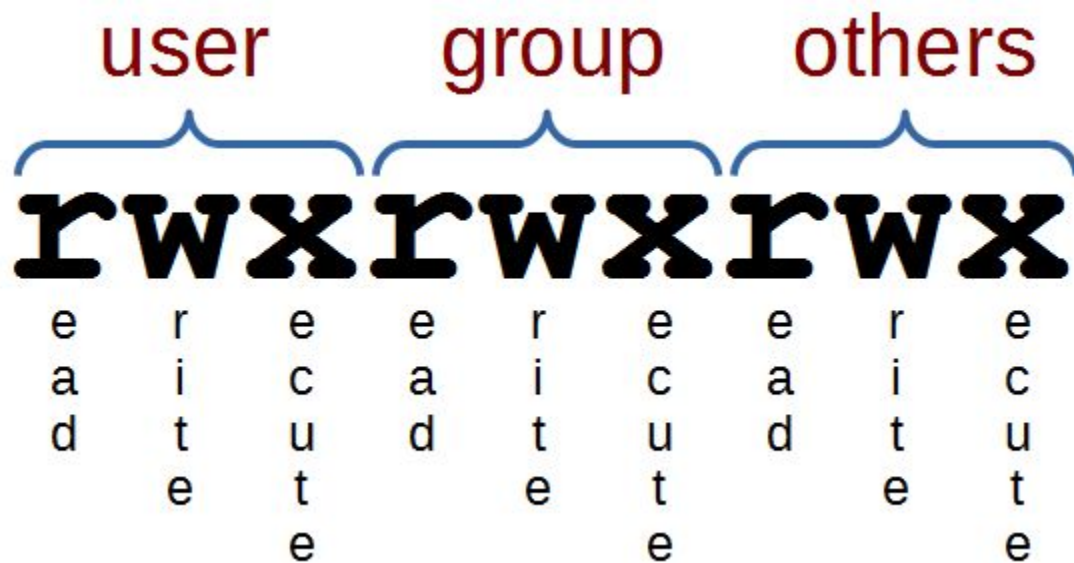
Crear y eliminar grupos

```
>> addgroup [nombre grupo]
```

```
>> delgroup [nombre grupo]
```




Permisos





Cambio de permisos

- Se utiliza el comando `chmod`
 - Método Simbólico
 - Método Octal



Método Simbólico

Símbolo Variable	Significado
u	user: usuario al que pertenece el archivo
g	group: grupo al que pertenece el archivo
o	other: otros usuarios diferentes
a	all: hace referencia a todos los anteriores

Símbolo Acción	Significado
+	añadir permiso
=	especificar permiso exacto
-	eliminar el permiso



Método Octal

	u g o		
	754		
access	r w x	r w x	r w x
binary	4 2 1	4 2 1	4 2 1
enabled	<u>1 1 1</u>	<u>1 0 1</u>	<u>1 0 0</u>
result	<u>4 2 1</u>	<u>4 0 1</u>	<u>4 0 0</u>
total	7	5	4



Ejemplos

```
>> chmod ug+x [archivo]
```

```
>> chmod 750 [archivo]
```



Línea de Comandos



Línea de comandos

```
root@tecmint:/home/avi#
```



Algunas ventajas

- No existe *overhead*
- Puede autenticarse remotamente en diferentes máquinas en red por medio de Internet
- Aplicaciones con GUI pueden iniciarse directamente desde la terminal

*Graphical user interfaces make easy
tasks easier, while command line
interfaces make difficult tasks
possible*





Emuladores de terminal

En entornos gráficos existen aplicaciones que emulan una terminal, ejemplos:

- `gnome-terminal`
- `konsole`
- `terminator`
- `xterm`
- `rxvt`



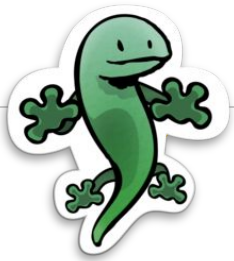
Emuladores de terminal

Se les llamar virtuales porque aunque todas pueden coexistir a la vez, solo una permanece visible.

Para acceder a las terminales virtuales se debe utilizar el comando:

- CTRL+ALT+Fn (Fn puede ser F1, F2, ... , F6)

F7 vuelve a abrir la interfaz gráfica (Debian)



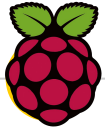
Comandos



Estructura de un comando

La mayoría de los comandos *shell* tienen tres elementos básicos:

- **Comando:** es el nombre del programa que se quiere ejecutar
- **Opciones:** opciones o switches que modifican lo que el comando hace. Usualmente inician con uno o dos guiones.
 - -p o -print
- **Argumentos:** variables de entrada del programa



sudo

sudo le permite al usuario
correr programas utilizando
los privilegios de seguridad
de otro usuario,
generalmente del root
(superman)



Configuración de sudo

1. Ejecutar el comando **su**, lo cual permitirá autenticarse como *root*
2. Crear el archivo de configuración para que le permita a su usuario utilizar sudo
 - Guardarlo en el directorio `/etc/sudoers.d/`
 - `echo "[usuario] ALL=(ALL) ALL" > /etc/sudoers.d/[usuario]`
3. Es recomendable hacer el cambio de permisos para el archivo de la siguiente manera
 - `chmod 440 /etc/sudoers.d/[usuario]`



Rutas absolutas y relativas

- **Absoluto:** /home/[usuario]/Desktop/ejemplo.txt
- **Relativo:** Desktop/ejemplo.txt
 - No inicia con /
- **Atajos:**
 - . (directorio actual)
 - .. (directorio anterior)
 - ~ (directorio home del usuario)



Standard File Streams

- **standard input:** `stdin` (usualmente el teclado)
- **standard output:** `stdout` (usualmente la consola)
- **standard error:** `stderr` (usualmente un archivo log de error)



Comandos esenciales



Comandos básicos

- cd
- cat
- echo
- ls (-l -a)
- mkdir
- rmdir
- man
- login
- exit
- ssh (username@remote-server.com)
- shutdown
- which
- whereis
- tree
- ln (-s)
- touch
- rm
- mv
- diff
- mount / unmount (sudo)
- dd
- tar



Redirección I/O

>> do_something < input_file

>> do_something > output_file

>> do_something 2> error_file



find

>> find [directorio] -name [nombre_archivo]

>> find [directorio] -iname [nombre_archivo] (ignora mayúsculas y minúsculas)

>> find [directorio] -type d -iname [nombre_archivo] (solo directorios)

>> find [directorio] -type f -iname [nombre_archivo] (solo archivos)



Lectura

Comando	Uso
cat	Para leer archivos cortos. No permite <i>scroll-back</i>
tac	Utilizado para leer un archivo de atrás hacia adelante
less	Para leer archivos más largos. Para búsquedas hacia adelante use / y hacia atrás ?
tail	Imprime las últimas 10 líneas del archivo.
head	El opuesto a tail



Mover o eliminar

Comando	Uso
mv	Cambiar el nombre a un archivo
rm	Eliminar un archivo
rm -f	Forzar la eliminación de un archivo
rm -i	Preguntar antes de eliminar un archivo
rm -rf	Eliminar carpeta



¡Gracias!

¿Preguntas?



Práctica 1



Práctica 2