



# UNIX/LINUX

## Conceptos Básicos I



# ÍNDICE

## Contenido

Introducción	1
Antes de UNIX	2
Estructura de Capas de UNIX	3
Filosofía de UNIX	4
Estructura de Directorios	5
El Shell	6
Ejercicios	7
Bibliografía	8

# Introducción

Este curso sobre el sistema operativo linux está constituido por el conjunto de Páginas que se utilizaron para impartir conceptos básicos de este sistema en varias empresas.

Ya que Linux es un sistema operativo de la familia UNIX, en este curso se hace referencia a la historia de UNIX, se considera relevante esta narrativa, ya que hace comprensible los conceptos de abstracción que se desarrollaron para elaborar un sistema operativo independiente del hardware y que son indispensables para entender como un programa usuario interactúa con el sistema operativo.

El sistema operativo Linux esta constituido por componentes, es decir se ignora la estructura interna de los programas que sirven como utilerias o representan dispositivos , pero se sabe que esperan como entrada, que responden como salida y la manera de ensamblarlos con otros componentes.

Este curso es una introducción al uso de los recursos computacionales de un equipo de cómputo Linux utilizando el interprete de comandos. En este curso se revisa las herramientas de edición, consulta y almacenamiento de archivos, manejo de procesos y las herramientas de configuración de los servicios de red.

El curso es interactivo, esto significa que se explican algunos conceptos y se pasa a ejercicios.

Es importante contar con un equipo de pruebas, en este curso se utilizaron 2 equipos virtuales CentOS 8 montados en virtualbox de Oracle.

Cada capítulo cuenta con sección de bibliografía dedicada aquellos que deseen profundizar en los temas.

# Antes de UNIX

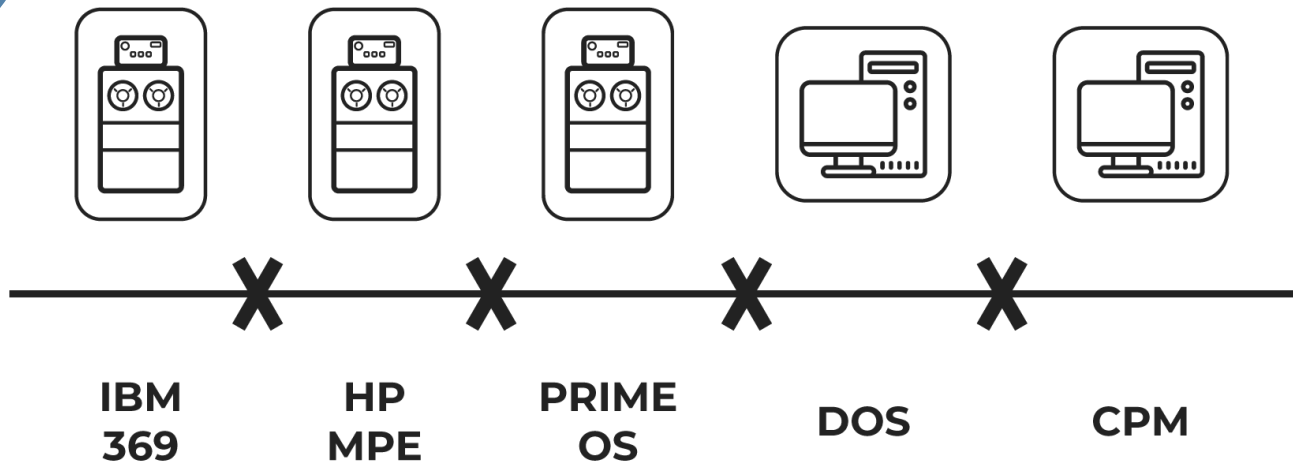


Figura 1: Diagrama de sistemas operativos.

## Relación del Software y Hardware.

- El sistema operativo era construido por el fabricante de hardware en ensamblador.
- El acceso al equipo multiusuario era a través de terminales que se comunicaban con la computadora mediante protocolos propiedad del fabricante.
- Los lenguajes de programación, el intérprete de comandos, manejadores de bases de datos y los sistemas de archivo eran específicos del sistema operativo.

## Profesional de Informática.

- Cambiar de trabajo significaba aprender un nuevo sistema operativo
- Portar una aplicación de un equipo a otro significaba reprogramar
- El equipo de cómputo se dividía en Mainframes y PC's
- Los Mainframes se accedían vía terminales.
- Las PC's eran monousuario y el traspaso de información se hacía con disquetes
- El trabajo de los administradores de sistemas se reducía a escoger un proveedor con el que se vivía toda la vida.

# Estructura de Capas de UNIX

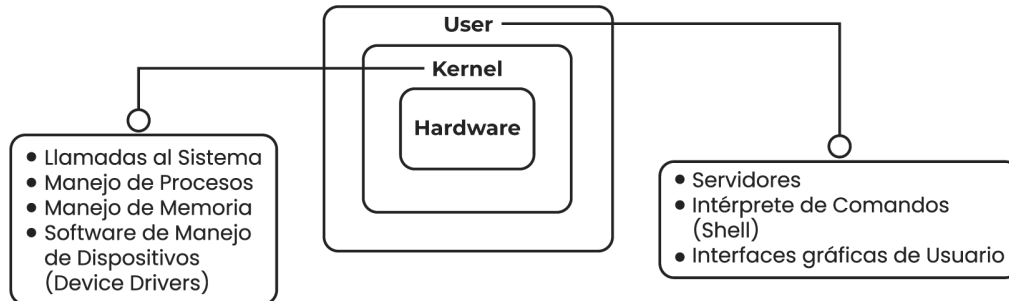


Figura 2: Diagrama de estructura de capas de UNIX.

## Relación del Software y Hardware.

- El sistema operativo es desarrollado en C, lenguaje de programación de alto nivel portable a otras arquitecturas de hardware.
- La interacción con el hardware se hace mediante el kernel, una capa de software que se instala en la memoria del equipo cuando arranca, mediante drivers se comunica con cada dispositivo que reconoce.
- Hay 2 modos de operación: El del sistema (kernel mode) y de usuario (User mode), el modo usuario lo arranca el kernel una vez concluido el proceso de reconocimiento del hardware y el montaje del sistema de archivos /, en modo de usuario, los servicios se arrancan mediante el proceso 1 (init).
- El kernel representa varios dispositivos de hardware como archivos, en esa categoría se encuentran: discos, impresoras, terminales, teclados, etcétera, esto simplifica la interacción de los programadores con los dispositivos, varios de ellos son manipulados como archivos de texto.

## Profesional de Informática.

- Los programas se desarrollan y prueban en una versión de UNIX para PC y ponerlos en producción significa compilarlos en la máquina destino.
- Cambiar de mainframe se vuelve una tarea trivial, el proveedor del servicio se escoge por calidad del equipo y el servicio.
- Las terminales de los mainframes se sustituyen por pc's con programas emuladores.
- La transferencia de información se hace a través de protocolos de red públicos.
- La capacitación del personal se realiza a través de libros de texto, las habilidades aprendidas en un empleo son valoradas como experiencia en el siguiente.

# Filosofía de UNIX

Un sistema de cómputo poderoso y complejo debe permanecer simple, general y extensible. Los conceptos anteriores se reflejan:

## INTERFACES DE USUARIO

### Comandos

Los comandos u ordenes están constituidos por la contracción de un verbo u otra palabra que indica una orden de 2-4 letras la acción se realiza sobre un objeto que puede ser una archivo, una aplicación o un dispositivo, las órdenes complejas pueden tener subórdenes antes del objeto, las opciones de una orden se indican con un `-o--`

La salida de un comando puede dirigirse a un archivo como entrada o a otro comando.

La ejecución fallida de un comando modifica el valor de una variable predefinida del intérprete de comandos, esto permite ejecutar acciones o notificar del resultado al usuario.

Hay varios intérpretes de comandos en UNIX/LINUX con diferencias de sintaxis y funcionalidad, sin embargo todos incluyen estructuras de programación como condiciones y ciclos.

### Gráficas

Similar a los intérpretes de comandos hay de varios "sabores" y cada usuario puede tener un "escritorio" con características particulares. En este curso no se exploran las características de estas interfaces de usuario.

## SISTEMA DE ARCHIVOS

Independientemente del dispositivo de almacenamiento o la organización interna de los datos, la información almacenada en dispositivos magnéticos se presenta como una estructura jerárquica de archivos y directorios.

## INTERFACES DE RED

Pueden ser ethernet, dsl, x.25, vpn's, ppp etcétera, sin embargo los comandos para asignar direcciones de red, ruteo estático o reglas del firewall son las mismas, independientemente del protocolo físico de comunicación.

## CONFIGURACIÓN

Los archivos de configuración del sistema operativo y los servicios en los equipos UNIX/LINUX, siempre se hace en archivos de texto alojados en el directorio /etc.

# Estructura de Directorios

Los sistemas operativos de la familia UNIX presentan una estructura de directorios apegada a un estándar en la imagen a continuación se presenta la estructura de directorios del primer nivel del estándar.

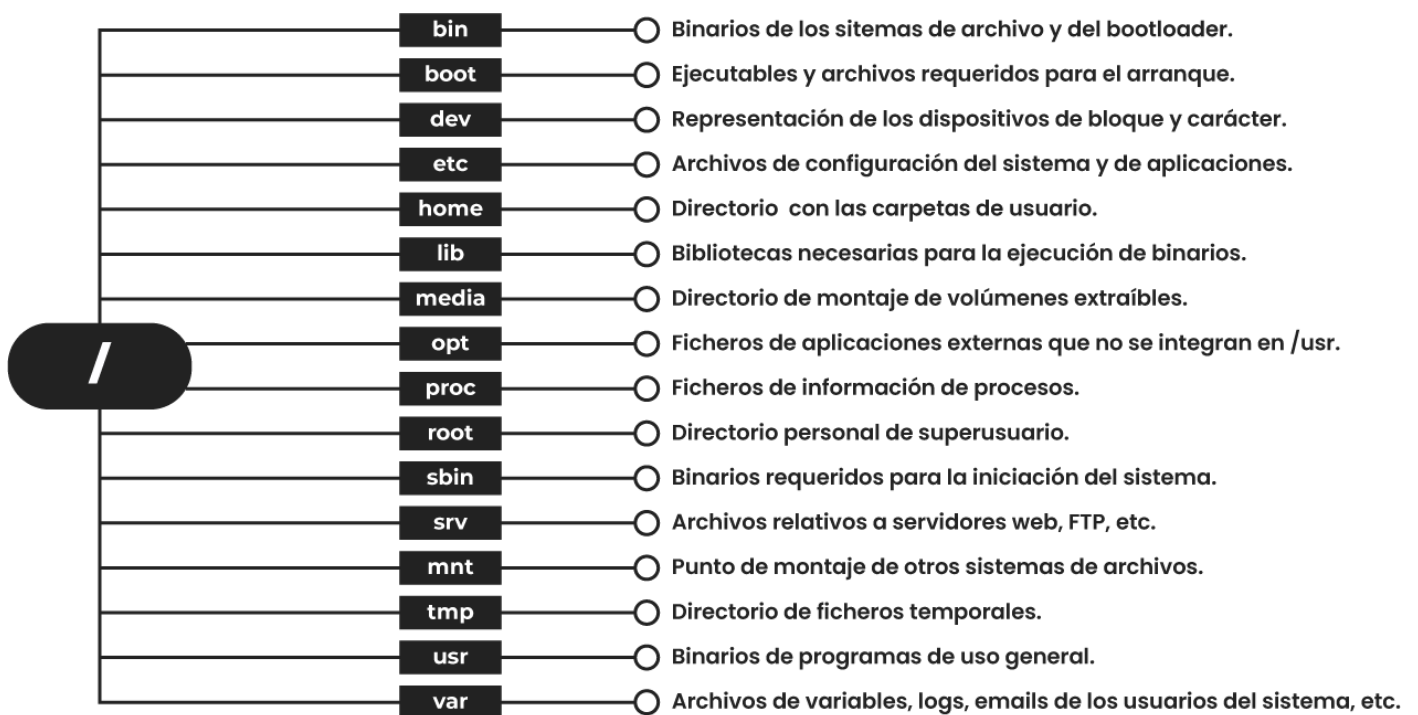


Figura 3: Diagrama de Estructura de Directorios en UNIX.

# El Shell

Mediante el intérprete de comandos (shell) se activan programas, algunos forman parte del sistema operativo y otros son elaborados por el usuario, desde la perspectiva del shell no hay diferencia, hay varios shells disponibles en los ambientes UNIX/LINUX.

- **cs***h*: Bourne shell, Popular en los 70's, se usa en FreeBSD
- **ks***h*: Korn shell, Disponible en equipos UNIX, HP-UX, SunOS, AIX
- **zs***h*: Zeta shell, Usado en Mac OS
- **ba***sh*: Bourne again shell, el estándar en Red Hat, CentOS
- **da***sh*: Debian Almquist shell, el estándar en Debian y Ubuntu.

La diferencia entre ellos es de sintaxis e interactividad.

Cuando los programas forman parte del sistema operativo reciben el nombre de comandos, si los programas son de usuario, simplemente programas o scripts.



# Ejercicios

La sintaxis de un comando en UNIX/LINUX es:

```
Comando -opcion1 -opcion2 objeto símbolo objeto
Donde símbolo puede ser: >< |
```

- Mostrar el contenido de un archivo:  
`cat /etc/passwd`
- Concatenar comandos (La salida de un comando se convierte en la entrada del siguiente)  
`cat /etc/passwd | less`
- La salida de un programa se envía a un archivo:  
`cat /etc/passwd >/tmp/usuarios.txt`
- Selección de un registro mediante una expresión regular:  
`cat /etc/passwd |grep httpd >/tmp/usuarioApache`
- Organizar en orden alfabético la base de datos de usuarios y mostrar por páginas:  
`cat /etc/passwd | sort | less`
- Leer del teclado:  
`while read linea; do echo $linea; done < /dev/stdin`
- Validar el resultado de un comando:  
`cd /tmp; touch archivo`  
`if ls -l ./archivo; then echo "OK"; else echo "NOK"; fi`  
`rm ./archivo`  
`if ls -l ./archivo 2>/dev/null; then echo "OK"; else echo "NOK"; fi`
- Comandos, modificadores y manuales:  
`cd`  
`touch archivo`  
`ls`  
`ls -l`  
`ls -lh`  
`ls -l -a *`  
`man ls`

# Ejercicios

- Con el comando `ps aux` se obtiene la lista de procesos por usuario. ¿Cómo puedo obtener los que corresponden a bash?
- El comando `rpm -q --all`, lista todos los paquetes de software instalados en el sistema. ¿Cómo pasarlo a un archivo?
- El comando que muestra los paquetes instalados del punto anterior lista los paquetes instalados, el comando `wc -l` cuenta las líneas de un archivo de texto. ¿Cómo puedo saber el número de paquetes instalados?

## Bibliografía

Brian Ward. How Linux works. No Starch Press. ISBN: 9781593275679.

Rosen H. Keneth, Rosinski R, Farber James: UNIX. System V Release 4. ISBN: 0-07-881552-5

[Linux HowTo en español](#)

Estándar del sistema de archivos de los sistemas operativos de la familia UNIX.

<http://www.pathname.com/fhs/>