
Resumen Teórico-Práctico SOA

Facultad de Informática, UNLP.

Alumno: Gonzalez, Joaquín Manuel.

Fecha de Realización: 28/09/2023.

Contenido

GNU:	3
Multitarea:	3
GNU/LINUX:	3
POSIX (Portable Operating System Interface):	3
Distribuciones de Linux:	3
Kernel:	4
Intérprete de Comandos (Shell):	4
File System:	4
Particiones:	5
Particionadores:	6
MBR (Master Boot Record):	6
MBC (Master Boot Code):	7
BIOS:	7
GPT (GUID Partition Table):	7
UEFI:	7
Bootloader/Gestor de Arranque:	7
Archivos:	7
Proceso de Arranque System V:	8
Proceso INIT:	8
RunLevels:	8
Archivo /etc/inittab:	9
ScriptsRC (Scripts de Inicio):	9
Proceso de arranque Upstart:	10
SystemD:	10
Archivo /etc/passwd:	10
Archivo /etc/shadow:	10
Archivo /etc/group:	11
Procesos:	11
Background (Segundo Plano):	11
Foreground (Primer Plano):	11
Pipe ():	12
Comandos:	12

GNU:

- Proyecto/Filosofía/Movimiento de software libre iniciado por Richard Stallman con el objetivo de crear un sistema operativo completo, libre, Unix like y sin las restricciones de software propietario.

Multitarea:

- Capacidad de un sistema operativo o computadora para ejecutar múltiples tareas o procesos de una manera aparentemente simultánea.

GNU/LINUX:

- Sistema operativo Unix like gratuito y libre.
- **Provee 4 libertades:**
 - ❖ Libertad de usar el software para cualquier propósito.
 - ❖ Libertad de estudiar su funcionamiento.
 - ❖ Libertad de distribuir copias.
 - ❖ Libertad de mejorar el software.
- Es multiusuario, multitarea, multiprocesador, opensource, case sensitive, tiene código altamente portable y en él todo es un archivo.
- **Componentes fundamentales:**
 - ❖ Kernel.
 - ❖ Intérprete de comandos.
 - ❖ File System.

POSIX (Portable Operating System Interface):

- Estándares desarrollados por la IEEE para definir una interfaz estándar entre sistemas operativos y aplicaciones en sistemas Unix like, permitiendo portabilidad e interoperabilidad entre diferentes plataformas.

Distribuciones de Linux:

- Customización de GNU/Linux formada por una versión de Kernel y determinados programas con sus configuraciones.
- Tienen sus propias características, enfoques y objetivos.

- Se diferencian entre ellas por su Origen y Base, Enfoques y Estabilidad, Gestor de Paquetes, Filosofías y Comunidades, y sus Entornos de Escritorios predeterminados.

Kernel:

- Elemento encargado de gestionar los recursos de hardware y proporcionar servicios esenciales para que los programas y procesos de un sistema computacional pueda funcionar.
- **Su función principal es la administración de Memoria y CPU.**
- Otras funciones que cumple son: Gestión de Dispositivos, File Systems, Comunicación, Sincronización, Interrupciones y Excepciones.
- Se encuentra en **/boot**.
- Es posible tener más de un Kernel en una misma máquina.
- **Es Monolítico:** Un único programa donde se ejecutan todas las funciones y servicios esenciales.
- **Es Híbrido:** Tiene la capacidad de cargar y descargar funcionalidad a través de drivers.

Intérprete de Comandos (Shell):

- Interfaz principal entre un usuario y el sistema operativo, actúa como modo de comunicación y ejecuta programas a partir del ingreso de comandos.
- Los comandos propios y externos al Shell se almacenan en **/bin**.
- No es parte del Kernel.
- Algunos Shell: Bash, Zsh, Bourne, etc.
- Cada usuario puede tener un Shell distinta si quisiera.

File System:

- Estructura y conjunto de reglas que utiliza un sistema operativo para administrar y rastrear la información en el almacenamiento.
- Define cómo se organizan los archivos, cómo se almacenan, cómo se acceden, cómo se nombran y cómo se controla el acceso a los mismos.
- Algunos File System que soporta Linux: FAT, NTFS, EXT3, EXT4, BTRS, etc.

- **Directorios importantes según la FHS (File Hierarchy Standard):**
 - ❖ **/:** Tope de la estructura e escritorios.
 - ❖ **/home:** Contiene los directorios de trabajo de todos los usuarios, excepto el del superusuario (administrador, root).
 - ❖ **/var:** Archivos variables, tales como logs, archivos spool, bases de datos, archivos de correo electrónico temporales, y algunos archivos temporales en general.
 - ❖ **/etc:** Archivos de configuración del sistema.
 - ❖ **/bin:** Archivos binarios y ejecutables.
 - ❖ **/dev:** Archivos de dispositivos y enlaces a los mismos.
 - ❖ **/usr:** Contiene utilidades y aplicaciones multiusuario, es decir, accesibles para todos los usuarios.
 - ❖ **/tmp:** Archivos temporales.

Permisos:

- **Lectura (r):** Permite al usuario leer el contenido de un archivo o directorio y listar su contenido con el comando ls. Vale 4
- **Escritura (w):** Permite la escritura, modificación y eliminación, en caso de directorios, la escritura permite crear nuevos archivos o borrar archivos existentes. Vale 2
- **Ejecución (x):** Permite a un usuario ejecutar el archivo si este es un programa, en caso de directorio, permite entrar al mismo usando el comando cd. Vale 1.

Particiones:

- Divisiones lógicas o segmentos en un dispositivo de almacenamiento, funciona como una unidad separada con su propio File System y espacio de almacenamiento.
- Se usan para organizar y administrar datos eficientemente.
- **Tipos de Particiones:**
 - ❖ **Primarias:** División cruda del disco, se almacena información de la misma en el MBR. Pueden haber 4 por disco.
 - ❖ **Extendidas:** Contenedoras de particiones lógicas. Solo puede existir una de este tipo por disco. No se define un File System sobre ella directamente.

- ❖ **Secundaria:** Ocupa la totalidad o parte de la partición extendida y se le define un tipo de File System, puede tener bootloader.
- Se identifican combinando el nombre del dispositivo de almacenamiento con un número que representa la partición (del 1 al 4). Dependiendo el tipo de disco pueden ser: **/dev/hda1, /dev/hdb2, /dev/hdc3, etc. Ó /dev/sda1, /dev/sdb2, /dev/sd3, etc.**
- Para montar Linux se necesita como mínimo 1 partición para almacenar la raíz:
 - ❖ **Tipo de Partición:** Primaria o Lógica.
 - ❖ **Identificación :** /dev/sdX1.
 - ❖ **Tipo de File System:** Cualquiera compatible.
 - ❖ **Punto de Montaje:** “/”.
- Cada partición puede tener un sistema operativo diferente, otro File System y un gestor de arranque que nos permita escoger el sistema operativo que queramos usar.
- **Solo una partición es marcada como bootable.**

Particionadores:

- **Existen 2 tipos:**
 - ❖ **Destructivos:** Permiten crear y eliminar particiones (fdisk).
 - ❖ **No Destructivos:** Permiten crear, modificar y eliminar particiones (Fips, Gparted).

MBR (Master Boot Record):

- Estructura crítica ubicada en el **cilindro 0, cabeza 0, sector 1** de cada dispositivo de memoria.
- Si existe más de un disco en la máquina, solo uno es asignado como Primary Master Disk.
- **Tiene un tamaño de 512 Bytes:**
 - ❖ **0...446:** MBC (Master Boot Code).
 - ❖ **446...510 (64 Bytes):** Tabla de Particiones.
 - ❖ **510...512:** 2 Bytes libres o para firmar el MBR.
- Contiene información para iniciar el sistema operativo y ubicar la partición activa desde la cual se cargará el bootloader/gestor de arranque.

MBC (Master Boot Code):

- Código que permite arrancar el Sistema operativo. Esta contenido en el MBR.

BIOS:

- Firmware ubicado en la motherboard, su función esencial es inicializar y coordinar el hardware durante el proceso de arranque. El MBR se usa en sistemas con BIOS.

GPT (GUID Partition Table):

- Sistema de particionado que utiliza UEFI para solucionar limitaciones del MBR, como la cantidad de particiones.
- Se pueden crear hasta 128 particiones.
- Se puede ver como sustitución del MBR.

UEFI:

- Firmware reemplazo del BIOS, es una tecnología más avanzada que ofrece una interfaz gráfica, arranque más rápido, mayor compatibilidad, mejor seguridad, configuración más avanzada, entre otras cosas. Hace uso de GPT.
- Es compatible con el MBR tradicional.

Bootloader/Gestor de Arranque:

- Programa que se encarga de iniciar el proceso de carga del sistema operativo.
- Su funcionalidad principal es cargar el sistema operativo desde el disco y transferir el control al sistema operativo cargado.

Archivos:

- Se identifican por su nombre, ubicación, permiso y propietario, tipo y sus metadatos.
- La extensión de los mismos es una cuestión de manejo entre usuarios, la extensión no define su tipo.

Proceso de Arranque System V:

- Se basa en runlevels y el uso de scriptsRC.
- Es sincrónico.
- **Pasos del proceso:**
 1. Se empieza a ejecutar el código del BIOS.
 2. El BIOS ejecuta el POST.
 3. El BIOS lee el sector de arranque (MBR).
 4. Se carga el gestor de arranque (MBC).
 5. El bootloader carga el kernel y el initrd.
 6. Se monta el initrd como file system raíz y se inicializan componentes esenciales.
 7. El Kernel ejecuta el proceso init y se desmonta el initrd.
 8. Se lee el /etc/inittab.
 9. Se ejecutan los scripts del runlevel 1
 - 10.El final del runlevel 1 le indica que vaya al runlevel por defecto.
 - 11.Se ejecutan los scripts del runlevel por defecto.
 - 12.El sistema está listo para usarse.

Proceso INIT:

- Tiene como objetivo llevar al sistema desde un estado de apagado o reinicio hasta un estado funcional completamente cargado y listo
- **Funciones:**
 - ❖ Cargar todos los subprocesos necesarios para el correcto funcionamiento del Sistema Operativo.
 - ❖ Es el encargado de montar los file systems y de hacer disponible los demás dispositivos.
- Padre de todos los procesos.
- Tiene PID = 1 y no posee PPID.

RunLevels:

- Son el modo en que arranca Linux en System V, cada uno es responsable de levantar (iniciar) o bajar (parar) una serie de 7 servicios.

- Su objetivo principal es definir el estado en el que un sistema operativo se encuentra durante su arranque y el conjunto de servicios y entornos que están disponibles en ese estado.

RunLevels:

- ❖ **Runlevel 0 (halt):** Este nivel se utiliza para apagar el sistema de manera segura.
- ❖ **Runlevel 1 (modo de usuario único):** Este nivel se utiliza para tareas de mantenimiento y recuperación del sistema
- ❖ **Runlevel 2 (modo multiusuario sin habilitar la red):** Se usa para iniciar el sistema en modo multiusuario sin habilitar la red.
- ❖ **Runlevel 3 (Full Multiuser Mode Console):** Similar al Runlevel 2 pero con servicios de red habilitados.
- ❖ **Runlevel 4 (No estándar):** Este nivel no se utiliza en muchas distribuciones y por lo tanto no tiene una configuración estándar.
- ❖ **Runlevel 5 (X11 Modo Gráfico o Multiusuario con Interfaz Gráfica):** Este nivel incluye todos los servicios de Runlevel 3, también inicia un sistema de escritorio o entorno gráfico.
- ❖ **Runlevel 6 (Reboot):** En este nivel, el sistema se reinicia.

Archivo /etc/inittab:

- Controla cómo se inicia el sistema operativo, qué servicios y procesos se inician automáticamente y en qué runlevels.
- Contiene líneas de configuración que definen las acciones a realizar en diferentes situaciones en diferentes runlevels.

ScriptsRC (Scripts de Inicio):

- Tienen como finalidad configurar y preparar el sistema operativo durante el proceso de arranque.
- Se encargan de iniciar y detener servicios, configurar el entorno y realizar otras tareas necesarias para poner a funcionar al sistema.
- Están almacenados en el directorio **/etc/init.d**.
- En los directorios **/etc/rcX.d (X = 0..6)** se encuentran los enlaces simbólicos relacionados a cada script. Los enlaces comienzan con **S**

(**start**) para indicar que se debe iniciar el servicio o con **K (stop)** indicando que se debe detener.

- Todos los scripts deben saber comprender el parámetro **K y S**.

Proceso de arranque Upstart:

- Primer reemplazo de System V, este maneja el proceso de arranque a través de **eventos** (sucesos que desencadenan acciones, como por ejemplo el inicio de un servicio) y **trabajos/jobs** (scripts de texto plano que definen las acciones/tareas “unidades de trabajo” a ejecutar ante determinados eventos) definidos en “/etc/init” (.conf).
- Es asincrónico.
- Es compatible con System V.
- Los **Jobs** son los reemplazos de los ScriptsRC.

SystemD:

- Sistema de inicialización y administración de servicios que centraliza la administración de demonios y librerías del sistema.
- Reemplazo de System V y Upstart.
- Los **runlevels** son reemplazados por **targets** y al igual que en Upstart el archivo /etc/inittab no existe más y el demonio systemd reemplaza el proceso init -> este pasa a tener PID 1.
- Puede ejecutar procesos de manera sincrónica y asincrónica.

Archivo /etc/passwd:

- Contiene información básica sobre cada usuario (nombre de usuario, identificación de usuario, identificación de grupo, nombre completo, directorio de inicio y Shell por defecto).
- Usa como delimitador “:”.
- En la **Fila 1** vemos los nombres de usuario, en la **Fila 6** vemos el directorio de inicio del usuario.

Archivo /etc/shadow:

- Contiene las contraseñas encriptadas de los usuarios.

- solo es accesible por el root.

Archivo /etc/group:

- Contiene información sobre los grupos de usuarios (nombre del grupo, identificación del grupo y una lista de miembros).
- Usa como delimitador ":".
- En la **Fila 1** vemos el nombre del grupo y en la **Fila 4** vemos la lista de usuarios pertenecientes separados por ",".

Procesos:

- Instancia de ejecución de un programa de software, cada uno de estos procesos tiene su propio espacio de memoria y recursos asignados, y pueden estar en ejecución en paralelo con otros procesos.
- **PID (Process ID)**: Identificador único de un proceso en el sistema operativo, estos se asignan secuencialmente.
- **PPID (Parent Process ID)**: Se refiere al PID del proceso padre que creó/inició el proceso actual.

Background (Segundo Plano):

- El proceso que se ejecuta no está directamente interactuando con el usuario a través de la terminal. Puede ejecutarse en paralelo y no bloquea la terminal.
- Para ejecutar un proceso en segundo plano solo necesitamos utilizar el símbolo "&" luego de especificar un comando en la terminal.
- Para pasar un proceso de **background a foreground** solo debemos de ejecutar el comando "fg %n" siendo n el PID.

Foreground (Primer Plano):

- Tiene el control sobre la interfaz de la terminal, la bloquea y requiere que el usuario interactúe directamente.
- Para pasar un proceso de **foreground a background** solo debemos de ejecutar el comando "bg %n" siendo n el PID.

Pipe (|):

- Permite la comunicación entre procesos al redirigir la salida estándar de un proceso hacia la entrada estándar de otro.

Comandos:

- **“cat [ruta]”**: Muestra el contenido de uno o varios archivos en la salida estándar.
- **“mkdir [ubicación]”**: Crea uno o más directorios en la ubicación especificada.
- **“cd”**: Se usa para cambiar el directorio de trabajo actual en la línea de comandos.

Parámetros:

- ❖ **“cd”**: Vuelve al directorio de inicio del usuario actual.
- ❖ **“cd [ruta]”**: Cambia a la ruta especificada.
- ❖ **“cd ..”**: Permite moverte un nivel hacia arriba en la estructura de directorios.
- **“touch [arch]”**: Crea archivos vacíos o actualiza las marcas de tiempo de archivos existentes.
- **“ls”**: Lista los archivos y directorios en el directorio actual o en una ubicación especificada.

Parámetros:

- ❖ **“ls -l”**: Lista detallada que incluye información sobre cada elemento (permisos, propietario, grupo, tamaño, etc).
- ❖ **“ls -a”**: Muestra todos los archivos inclusive los ocultos.
- ❖ **“ls -h”**: Muestra los tamaños de los archivos en tamaños legibles.
- ❖ **“ls -R”**: Muestra los contenidos recursivamente, es decir, se mete en profundidad de los sub directorios.
- **“find [ruta]”**: Busca archivos y directorios en base a criterios necesarios utilizando patrones o propiedades dentro del sistema de archivos. **Busca por nombre de archivo.**

Parámetros:

- ❖ **“find [ruta] -name [nombre]”**: Busca por nombre, se pueden usar **asteriscos** para realizar búsquedas parciales o múltiples.

Por ejemplo: **“cadena*”** busca todos los que contengan en su nombre la palabra **“cadena”**.

❖ **“find [ruta] –type”**: Especifica la búsqueda por tipo de archivos, **“f”** para archivos regulares y **“d”** para directorios.

- **“who”**: Muestra información sobre los usuarios conectados al sistema.

Parámetros:

❖ **“who –q”**: Muestra número total de usuarios conectados.

- **“shutdown”**: Programa el apagado o reinicio del sistema.

Parámetros:

❖ **“-h”**: Apaga el sistema.

❖ **“-r”**: Reinicia el sistema.

❖ **“now”**: Apaga o reinicia inmediatamente.

❖ **“HH:MM”**: Establece hora específica de apagado o reinicio.

- **“reboot”**: Reinicia la computadora inmediatamente.
- **“halt”**: Apaga la computadora de manera inmediata y abrupta cortando la energía.
- **“sudo”**: Comando que permite a los usuarios ejecutar comandos con los privilegios de otro usuario, generalmente el superusuario (root).
- **“telinit [PID]”**: Comando que se utiliza para cambiar el runlevel del sistema.
- **“useradd ó adduser”**: Comando que se utiliza agregar un nuevo usuario al sistema, crea una nueva entrada de usuario en los archivos: /etc/passwd, /etc/shadow, /etc/group y /etc/gshadow.
- **“usermod”**: Permite hacer cambios en la configuración de un usuario sin necesidad de eliminar y recrear al mismo.

Parámetros:

❖ **“-g”**: Cambia el grupo primario del usuario.

❖ **“-l”**: Cambia el username.

- **“userdel”**: Se encarga de eliminar un usuario del sistema.
- **“su”**: Permite a un usuario cambiar a otra cuenta de usuario sin necesidad de cerrar y volver a iniciar sesión.
- **“passwd”**: Proporciona una forma segura de cambiar las contraseñas de los usuarios en el sistema.
- **“chmod”**: Se utiliza para cambiar los permisos de acceso a archivos y directorios (Lectura, Escritura, Ejecución).

Parámetros:

❖ **Modos Numéricos:** Especifican los permisos utilizando los números

❖ **Modos Simbólicos:** Permite especificar el cambio de permisos de una manera más legible y comprensible, por ejemplo, `chmod u+r archivo`

Tipos de Permisos:

- **“u”:** Para el propietario.
- **“g”:** Para el grupo al que pertenece el archivo.
- **“o”:** Para el resto de usuarios.
- **“a”:** Para todos los usuarios.

Operadores:

- **“+”:** Agrega permisos.
- **“-”:** Quita permisos.
- **“=”:** Establece permisos específicos.

- **“rmdir”:** Comando utilizado para eliminar directorios vacíos.
- **“cp [opciones] origen destino”:** Duplica archivos y/o directorios, puede copiar unos o varios archivos a un directorio específico o también puede copiar un directorio entero.
- **“mv [opciones] origen destino”:** Cambia la ubicación de un archivo o directorio y también se utiliza para cambiar el nombre de los mismos. Para únicamente cambiar el nombre del archivo solo se debe especificar el nombre viejo y el nombre nuevo, no hay que usar una ruta de destino.
- **“ps”:** Muestra información sobre los procesos en ejecución.

Parámetros:

- ❖ **“-e”:** Muestra todos los procesos del sistema.
- **“pstree”:** Representación gráfica de la jerarquía de procesos.