

Introducción a Sistemas Operativos Practica 1

1. Características de GNU/Linux:

(a) Mencione y explique las características más relevantes de GNU/Linux.

Es un sistema operativo tipo Unix (Unix like), pero libre.

S.O diseñado por miles de programadores, gratuito y de libre distribución (se puede descargar desde la web, un cd, etc).

Existen diversas distribuciones (customizaciones).

Es código abierto, lo que nos permite estudiarlo, personalizarlo, auditar, aprovecharnos de la documentación, etc.

(b) Mencione otros sistemas operativos y compárelos con GNU/Linux en cuanto a los puntos mencionados en el inciso a.

En comparación a Windows de Microsoft o IOS de Mac, Linux ofrece una cantidad de libertades que estos dos últimos no ofrece, al poder ver su código fuente y trabajar sobre él con un objetivo particular, se puede crear una distribución con un fin específico. En cambio los otros dos sistemas operativos deben ser adaptados o forzados para tener un fin específico, ya que no se puede ver su código fuente o su documentación.

(c) ¿Qué es GNU?

GNU No es Unix.

(d) Indique una breve historia sobre la evolución del proyecto GNU

El proyecto fue iniciado por Richard Stallman en 1983 con el fin de crear un Unix Libre (el sistema GNU). Para esto se creó el marco regulatorio GPL (General Public License de GNU). Stallman creó el FSF (Free Software Foundation) con el fin de financiar el proyecto GNU.

Para 1990, GNU ya contaba con un editor de textos (Emacs), un compilador (GCC) y gran cantidad de bibliotecas que componen un Unix típico, pero faltaba el componente principal, El Núcleo (Kernel). En 1988 se abandonó, debido a su complejidad, el núcleo conocido como TRIX. Para esto se decide adoptar como base el núcleo MACH para crear GNU Hurd, el cual tampoco prosperó.

Linus Torvalds estaba trabajando para 1991 en un Kernel denominado Linux, el cual se distribuye bajo la licencia GPL. En 1992 Torvalds y Stallman deciden fusionar ambos proyectos, y es de aquí donde nace GNU/LINUX.

(e) Explique qué es la multitarea, e indique si GNU/Linux hace uso de ella.

La multitarea es la capacidad de un sistema operativo de dar servicio a más de un proceso, haciendo que estos se ejecuten simultáneamente. GNU/Linux hace uso del multitasking.

(f) ¿Qué es POSIX?

Es un sistema operativo portátil para UNIX, creado por Stallman en 1980. Su principal característica es su portabilidad, permitiendo que sea sencillo y flexible de adaptar.

2. Distribuciones de GNU/Linux:

(a) ¿Qué es una distribución de GNU/Linux? Nombre al menos 4 distribuciones de GNU/Linux y cite diferencias básicas entre ellas.

Las distribuciones de GNU/Linux son distintos sistemas operativos con diferentes componentes en sí, pero que funcionan bajo el mismo kernel de Linux. Entre estas se pueden encontrar Arch Linux, Ubuntu, Linux Mint, Fedora, Debian, etc.

(b) ¿En qué se diferencia una distribución de otra?

Las distribuciones se diferencian en su funcionalidad, componentes, el objetivo para el que están creadas, sus capacidades y sus límites.

(c) ¿Qué es Debian? Acceda al sitio 1 e indique cuáles son los objetivos del proyecto y una breve cronología del mismo.

Debian es otra distribución de GNU/Linux bajo el nombre de la misma organización que lo creó.

Los objetivos del proyecto son crear un sistema operativo libre, disponible abiertamente para todos. Al decir libre, no hablan en términos de dinero, si no en términos de software libre.

3. Estructura de GNU/Linux:

(a) Nombre cuales son los 3 componentes fundamentales de GNU/Linux.

- 1 - El kernel. Es el elemento base del SO y es fundamental para su funcionamiento.
- 2 - Espacio de usuario del sistema. Es la capa administrativa para las tareas de todo el sistema, como la configuración e instalación del software.
- 3 - Aplicaciones libres y genéricas. Son el software creado en base a Linux.

(b) Mencione y explique la estructura básica del Sistema Operativo GNU/Linux.

Núcleo: el kernel, el encargado de ejecutar programas y gestionar los dispositivos de hardware. Se encarga de que software y hardware trabajen juntos.

Interprete de comandos: modo de comunicación entre el usuario y el SO. Permite ejecutar programas a partir del ingreso de comandos.

Sistema de archivos: Organiza la forma en que se almacenan los archivos en dispositivos de almacenamiento (fat, ntfs, ext2, ext3, reiser, etc.)

4. Kernel:

(a) ¿Qué es? Indique una breve reseña histórica acerca de la evolución del Kernel de GNU/Linux.

El Kernel es el núcleo del sistema operativo. En 1991 Linus Torvalds inicia la programación de un Kernel Linux basado en Minix (clon de Unix desarrollado por Tenenbaum en 1987, con el fin de crear un SO de uso didáctico).

En 1991 se anunció la primera versión "oficial" de Linux (0.02). En 1992 se combina su desarrollo con GNU, formando GNU/Linux. La versión 1.0 apareció el 14 de marzo.

(b) ¿Cuáles son sus funciones principales?

Es el encargado de que el software y el hardware puedan trabajar juntos. Sus funciones más importantes son la administración de memoria, CPU y E/S.

(c) ¿Cuál es la versión actual? ¿Cómo se definía el esquema de versionado del Kernel en versiones anteriores a la 2.4? ¿Qué cambió en el versionado se impuso a partir de la versión 2.6?

La versión actual y la última, es la versión 6.4.11. El esquema de versionado antes de la versión 2.4, seguía el formato x.y.z, donde x representa la versión principal (major version), y indicaba el número de versión menor (minor version), dando a entender que si era par se consideraba estable, si era impar era en desarrollo o experimental, z era el número de parche, que se incrementa con cada actualización, corrección de errores, etc.

La versión 2.6 fue la más estable e incorpora varios cambios al esquema de versionado:

Se eliminaron las distinciones entre versiones pares e impares, en Y no se jugaba más con el par para estable e impar para desarrollo, todas se consideraban estables. Se introdujo el desarrollo continuo con ventanas de fusión, donde en lugar de mantener versiones separadas para desarrollo y producción, el desarrollo del kernel se volvió más continuo.

También se introdujo una versión menor más dinámica, donde el número de versión menor Y comenzó a incrementarse más regularmente, a medida que lanzaban nuevas características y mejoras. Esto permitió una evolución más gradual y predecible del kernel. Por último, el mantenimiento a largo plazo. Con la serie 2.6 se empezó a ofrecer un soporte prolongado para ciertas versiones, conocidas como LTS (Long Term Support), que reciben actualizaciones de seguridad y correcciones de errores durante un periodo extendido, mientras que otras versiones recibían soporte a corto plazo.

(d) ¿Es posible tener más de un Kernel de GNU/Linux instalado en la misma máquina?

Es posible tener más de un Kernel si se está trabajando con diferentes versiones de Linux.

(e) ¿Dónde se encuentra ubicado dentro del File System?

Normalmente se encuentra en `/usr/src/linux`.

(f) ¿El Kernel de GNU/Linux es monolítico? Justifique.

El kernel es monolítico e híbrido, los drivers y el código del kernel se ejecutan en modo privilegiado, lo que lo hace híbrido es la capacidad de cargar y descargar funcionalidad a través de módulos.

5. Intérprete de comandos (Shell):

(a) ¿Qué es?

El Intérprete de Comandos o CLI (Command Line Interface) es el modo de comunicación entre el usuario y el SO.

(b) ¿Cuáles son sus funciones?

Sus funciones son ejecutar programas a partir del ingreso de comandos.

(c) Mencione al menos 3 intérpretes de comandos que posee GNU/Linux y compárelos entre ellos.

Bash: es el shell por defecto de linux.

ZSH: Incorpora funcionalidades de bash, ksh y tcsh, además de funciones propias.

fish: un shell ambientado en los 90's que es user-friendly.

(d) ¿Dónde se ubican (path) los comandos propios y externos al Shell?

Los comandos externos se suelen encontrar en /usr/bin, por ejemplo, si ejecutamos ls, shell pide al núcleo cargar /usr/bin/ls

Los comandos externos son aquellos archivos que tengan formato binario ejecutable o archivos con formato de texto que representan un script de comandos, esto depende de la ubicación del archivo.

(e) ¿Por qué considera que el Shell no es parte del Kernel de GNU/Linux?

La confiabilidad de la Shell y la capacidad de ser reemplazada son los principales motivos para no tener la Shell en el Kernel.

(f) ¿Es posible definir un intérprete de comandos distinto para cada usuario? ¿Desde dónde se define? ¿Cualquier usuario puede realizar dicha tarea?

Si, es posible definir un intérprete de comandos distintos para cada usuario. Esta configuración se realiza en el archivo /etc/passwd y puede ser definida por el administrador del sistema o por el propio usuario, dependiendo de los permisos.

6. Sistema de Archivos (File System):

(a) ¿Qué es?

Organiza la forma en que se almacenan los archivos en dispositivos de almacenamiento.

(b) Mencione sistemas de archivos soportados por GNU/Linux.

El adoptado por GNU/Linux es Extended. También de forma nativa acepta particiones FAT16, FAT32 y exFAT (v2,v3,v4).

(c) ¿Es posible visualizar particiones del tipo FAT y NTFS en GNU/Linux?

Si, es posible visualizar y también acceder a particiones del tipo FAT y NTFS.

(d) ¿Cuál es la estructura básica de los File System en GNU/Linux? Mencione los directorios más importantes e indique qué tipo de información se encuentra en ellos. ¿A qué hace referencia la sigla FHS?

/: tope de la estructura de directorios, como C:\.

/home: se almacenan archivos de usuarios (Mis documentos).

/var: información que varía de tamaño (logs, bd, spools).

/etc: archivos de configuración.

/bin: archivos binarios y ejecutables.

/dev: enlace a dispositivos.

/usr: aplicaciones de usuarios.

7. Particiones:

(a) Definición. Tipos de particiones. Ventajas y Desventajas.

Una partición es una división lógica de almacenamiento destinada a contener el SO que el usuario desea instalar, en conjunto con su sistema de archivos. Existen:

Partición primaria: división cruda del disco (puede haber 4 por disco). Se almacena información de la misma en el MBR.

Partición Extendida: sirve para contener unidades lógicas en su interior. Solo puede existir una partición de este tipo por disco. No se define un tipo de FS directamente sobre ella.

Partición Lógica: ocupa la totalidad o parte de la partición extendida y se le define un tipo de FS. Las particiones de este tipo se conectan como una lista enlazada.

(b) ¿Cómo se identifican las particiones en GNU/Linux? (Considere discos IDE, SCSI y SATA).

En discos IDE se identificaban las particiones con el prefijo hd, siendo hda para el primer disco IDE, hda1 para la primera partición en el primer disco, hdb2 para la segunda partición en el segundo disco..

En discos SCSI y SATA se identifican con el prefijo sd, siendo sda para el primer disco y sdb para el segundo, y así sucesivamente.

(c) ¿Cuántas particiones son necesarias como mínimo para instalar GNU/Linux? Nómbralas indicando tipo de partición, identificación, tipo de File System y punto de montaje.

En una instalación mínima de GNU/Linux, se necesita una partición Primaria o Lógica, generalmente identificada como sda1, un FS de tipo ext4, y se utiliza como punto de montaje /.

(d) Ejemplifica diversos casos de particionamiento dependiendo del tipo de tarea que se deba realizar en su sistema operativo.

En caso de usarlo como estación de trabajo personal, se utiliza tener una partición raíz, una swap y una home.

En el caso de usarlo para un servidor web, se recomienda usar raíz, swap, var, home, y opt.

En el caso de sistema de desarrollo se recomienda utilizar raíz, swap, home, usr y tmp.

En el caso de usar sistemas compartidos, se recomienda usar particiones windows, raíz, swap, home y una partición compartida.

(e) ¿Qué tipo de software para particionar existe? Menciónelos y compare.

Existen dos tipos de software particionadores. Los destructivos, que permiten crear y eliminar particiones (fdisk). Y los no destructivos, que permiten crear, eliminar y modificar particiones (fips, gparted) <- generalmente las distribuciones permiten hacerlo desde la interfaz de instalación.

8. Arranque (bootstrap) de un Sistema Operativo:

(a) ¿Qué es el BIOS? ¿Qué tarea realiza?

En las arquitecturas x86, el BIOS (Basic I/O System) es el responsable de iniciar la carga del SO a través del MBC. Está cargado en un chip (ROM, NVRAM). Existe en otras arquitecturas pero se lo conoce con otro nombre: Power on Reset + IPL en mainframe. OBP(OpenBoot PROM) en SPARC.

(b) ¿Qué es UEFI? ¿Cuál es su función?

UEFI aporta criptografía, autenticación de red y una interfaz gráfica. Define la ubicación del gestor de arranque. Define la interfaz del gestor de arranque y el firmware. Expone información para los gestores de arranque con:

Información de hardware y configuración de firmware. Punteros a rutinas que implementan los servicios que el firmware ofrece a los bootloaders u otras aplicaciones UEFI. Provee BootManager para cargar aplicaciones UEFI.

(c) ¿Qué es el MBR? ¿Qué es el MBC?

El MBR (Master Boot Record) es una estructura de datos ubicada en el primer sector del disco duro, se utiliza para gestionar el arranque del sistema y la partición del disco.

El MBC es un pequeño código que permite arrancar el SO. La última acción del BIOS es leer el MBC, lo lleva a memoria y lo ejecuta. Si se tiene un sistema instalado -> bootloader del MBC típico. Sino -> uno diferente (multietapa).

(d) ¿A qué hacen referencia las siglas GPT? ¿Qué sustituye? Indique cuál es su formato.

Las siglas GPT hacen referencia a GUID Partition Table, este especifica la ubicación y formato de la tabla de particiones en un disco duro. Sustituye al MBR.

(e) ¿Cuál es la funcionalidad de un “Gestor de Arranque”? ¿Qué tipos existen? ¿Dónde se instalan? Cite gestores de arranque conocidos.

La finalidad del bootloader es la de cargar una imagen de Kernel (sistema operativo) de alguna partición para su ejecución. Existen algunos como GRUB, LILO, NTLDR, GAG, YaST, etc. Estos gestores se suelen instalar en el MBR.

(f) ¿Cuáles son los pasos que se suceden desde que se prende una computadora hasta que el Sistema Operativo es cargado (proceso de bootstrap)?

El proceso de bootstrap es el siguiente:

En arquitecturas x86 el BIOS es el responsable de iniciar la carga del SO a través del MBC. Se carga el programa de booteo (desde el MBR). El gestor de arranque lanzado desde el MBC carga el Kernel, prueba y hace disponibles los dispositivos, para luego pasar el control al proceso **init**. El proceso de arranque se ve como una serie de pequeños programas de ejecución encadenada

(g) Analice el proceso de arranque en GNU/Linux y describa sus principales pasos.

(h) ¿Cuáles son los pasos que se suceden en el proceso de parada (shutdown) de GNU/Linux?

Inicia generalmente mediante un comando como shutdown, halt o poweroff. Se notifica a los usuarios y se bloquean los nuevos inicios de sesión. Se terminan las sesiones activas de usuarios y se terminan todos los procesos que estos activaron. Se detienen los servicios y luego se procede a escribir todos los datos en memoria sobre el disco. Realizado este punto, se desmonta o remonta el sistema de archivos raíz, solo en modo lectura. Finalizado esto, se detiene/apaga el hardware.

(i) ¿Es posible tener en una PC GNU/Linux y otro Sistema Operativo instalado? Justifique.

Es posible, si se particiona correctamente el disco se puede tener una pc GNU/Linux + cualquier otro sistema operativo, siempre y cuando se respeten las particiones que corresponden a cada sistema y que el equipo admita un dual boot, para administrar ambos SO.

9. Archivos:

(a) ¿Cómo se identifican los archivos en GNU/Linux?

Cada archivo tiene un nombre y puede ser identificado por su ruta absoluta, empezando desde el directorio raíz / (ej: /home/usr/doc.txt).

(b) Investigue el funcionamiento de los editores vi y mcedit, y los comandos cat y more.

Vi maneja en memoria el texto entero de un archivo.

mcedit es un editor de textos integrado en el administrador de ficheros para la terminal Midnight Commander.

El comando cat permite concatenar y mostrar el contenido de archivos, deriva de la palabra concatenar y se utiliza para visualizar, unir y crear archivos.

El comando more se utiliza para ver (pero no modificar) el contenido de un archivo o comando, y visualizarlo por paginas.

(c) Cree un archivo llamado “prueba.exe” en su directorio personal usando el vi. El mismo debe contener su número de alumno y su nombre.

(d) Investigue el funcionamiento del comando file. Pruébalo con diferentes archivos. ¿Qué diferencia nota?

10. Indique qué comando es necesario utilizar para realizar cada una de las siguientes acciones.

Investigue su funcionamiento y parámetros más importantes:

(a) Cree la carpeta ISO2017 listo

(b) Acceda a la carpeta (cd) listo

(c) Cree dos archivos con los nombres iso2017-1 e iso2017-2 (touch) listo

(d) Liste el contenido del directorio actual (ls) listo

- (e) Visualizar la ruta donde estoy situado (pwd) /home/limbo/iso_2017
 (f) Busque todos los archivos en los que su nombre contiene la cadena "iso*" (find)
 find . -name "iso*"
 (g) Informar la cantidad de espacio libre en disco (df)

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
udev	974964	0	974964	0%	/dev
tmpfs	201452	1180	200272	1%	/run
/dev/sda1	9671268	5591856	3566544	62%	/
tmpfs	1007248	0	1007248	0%	/dev/shm
tmpfs	5120	8	5112	1%	/run/lock
tmpfs	201448	80	201368	1%	/run/user/1000

- (h) Verifique los usuarios conectado al sistema (who). Solo yo
 (i) Acceder a el archivo iso2017-1 e ingresar Nombre y Apellido
 echo "Nombre Apellido" > iso2017-1
 (j) Mostrar en pantalla las últimas líneas de un archivo (tail). tail iso2017-1

11. Investigue su funcionamiento y parámetros más importantes:

- (a) **shutdown**: se utiliza para apagar, reiniciar o poner en estado de espera la computadora.
 (b) **reboot**: el comando reinicia la computadora en linux.
 (c) **halt**: detiene todos los procesos y apaga la computadora de forma un tanto forzada, anteriormente sólo detenía el s.o, pero ahora detiene todo, incluyendo hardware.
 (d) **locate**: busca archivos en el sistema utilizando una base de datos pre indexada.
 (e) **uname**: muestra información sobre el sistema operativo y el hardware.
 (f) **dmesg**: muestra los mensajes del ring buffer del núcleo del sistema.
 (g) **lspci**: muestra información sobre los PCI/periféricos conectados al sistema.
 (h) **at**: programa la ejecución de comandos/scripts en un momento específico del futuro.
 (i) **netstat**: muestra información sobre conexiones de redes, tablas de enrutamiento, interfaces de red y estadísticas de red.
 (j) **mount**: monta sistemas de archivos y dispositivos en el sistema de archivos del s.o.
 (k) **umount**: desmonta el sistema de archivos que está montado en /mnt.
 (l) **head**: muestra las primeras líneas de 1 o más archivos.
 (m) **losetup**: se utiliza para gestionar dispositivos de bucle (loop devices) que permiten montar archivos de imagen de disco como si fueran dispositivos de bloque.
 (n) **write**: permite enviar mensajes a otros usuarios que están conectados al sistema.
 (ñ) **mkfs**: se usa para crear un sistema de archivos en una partición o en un dispositivo de almacenamiento.
 (o) **fdisk (con cuidado)**: se utiliza para crear, modificar o eliminar particiones en discos duros.

12. Investigue su funcionamiento y parámetros más importantes:

(1) Indique en qué directorios se almacenan los comandos mencionados en el ejercicio anterior.

- (a) shutdown:** se almacena en /usr/sbin/shutdown o sbin/shutdown.
- (b) reboot:** se almacena en /sbin/reboot o /usr/sbin/reboot.
- (c) halt:** se almacena en /sbin/halt o /usr/sbin/halt.
- (d) locate:** se almacena en /usr/bin/locate.
- (e) uname:** se almacena en /bin/uname.
- (f) dmesg:** se almacena en /bin/dmesg o /usr/bin/dmesg.
- (g) lspci:** se almacena en /sbin/lspci o /usr/sbin/lspci.
- (h) at:** se almacena en /usr/bin/at.
- (i) netstat:** se almacena en /bin/netstat o /usr/bin/netstat.
- (j) mount:** se almacena en /bin/mount o /usr/bin/mount.
- (k) umount:** se almacena en /bin/umount o /usr/bin/umount.
- (l) head:** se almacena en /usr/bin/head.
- (m) losetup:** se almacena en /sbin/losetup.
- (n) write:** se almacena en /usr/bin/write.
- (ñ) mkfs:** se almacena en /sbin/mkfs.ext4, /sbin/mkfs.vfat, etc.
- (o) fdisk (con cuidado):** se almacena en /sbin/fdisk.