



Resumen Practica 1

Sistema Operativo

- Intermediario entre el usuario y el hardware, este se encarga de crear un entorno eficiente para ejecutar programas y asegura el correcto funcionamiento del sistema.
- Sus funciones principales son la de administrar memoria, CPU y dispositivos.

GNU/LINUX

- Es un sistema operativo de tipo Unix, libre y gratuito. Este ha sido diseñado por miles de programadores y está disponible en diversas distribuciones. También es de código abierto, esto significa que se permite estudiarlo, personalizarlo y auditarlo

Libertades de GNU

GNU se refiere a 4 libertades principales de los usuarios del software:

Libertad de usar el programa con cualquier propósito, estudiar su funcionamiento, distribuir sus copias y mejorar los programas.

Software Libre

Este mismo permite usar, copiar, estudiar, modificar y redistribuir libremente. Es generalmente sin costo, pero no hay que confundirlo con gratis, ya que pueden haber restricciones. Este mismo suele incluir el código fuente, lo cual permite que haya una corrección más rápida de fallas. Y por ultimo fomenta la libertad de los usuarios para mejorar el software.

Software Propietario

Este tiene un costo asociado, no se puede distribuir libremente como el software libre. No posee su código fuente y la corrección de fallas está a cargo del propietario.

GPL

- Es una licencia pública creada en los años 80 por la FSF y esta misma protege la libre distribución, modificación y uso del software GNU. También asegura que el software bajo esta licencia sea libre, logrando así respetar las 4 libertades principales.
- Su versión actual es la 3

Características de GNU/LINUX

Las principales características serían la de que es multiusuario y multitarea, también es altamente portable y multiprocesador. Tiene varios intérpretes de comandos, los cuales algunos son programables, maneja permisos y usuarios, TODO ES UN ARCHIVO, los directorios pueden estar en diferentes particiones, es case sensitive y por ultimo pero menos importante: Es de código abierto.

Diseño de GNU/LINUX

- Este mismo está enfocado en la portabilidad de las fuentes, o sea la capacidad de un software para ser compilado y ejecutado en diferentes plataformas sin necesidad de modificaciones grandes en el código fuente.
- Su desarrollo es en capas con separación de funciones. Cada capa actúa como una caja negra.
- Permite desarrollo distribuido
- Soporta diversos sistemas de archivos
- Usa memoria virtual, la de RAM + SWAP
- Mayormente es desarrollado en C y ensamblador, con uso de otros lenguajes como por ejemplo Java, Perl y Python.

Kernel

Sistema operativo en un sentido estricto, este permite la interacción entre el software y el hardware al poder ejecutar programas y gestionar dispositivos. Es monolítico híbrido, con administración de memoria, CPU y Entrada/Salida, el cual los drivers y el código fuente se abren en modo privilegiado y se puede cargar y descargar módulos. También está licenciado bajo GPL v2

Intérprete de comandos

O mejor conocido como CLI, es el modo de comunicación entre el usuario y el SO el cual ejecuta programas mediante entrada de comandos; cada usuario puede tener su propia interfaz o SHELL (y son personalizables y programables), ejemplos Korn Shell o Bourne Shell.

Directorios importantes según FHS

/: Raíz de estructura de directorios
/var: Contiene información variable
/etc: Archivos de configuración
/usr: Aplicaciones de usuarios

/home: Almacena archivos de usuario
/bin: Archivos binarios y ejecutables
/dev: Enlaces a dispositivos

Distribuciones GNU/LINUX

Una distribución es una personalización de GNU/LINUX que incluye una versión del kernel y ciertos programas con sus configuraciones específicas.

MBR (Master Boot Record)

MBR es un sector que está reservado en el disco físico (para ser más específicos cilindro 0, cabeza 0, sector 1) el cual se ubica en TODOS los discos y uno solo se designa como Primary Master Disk si hay muchos. Su tamaño es de 512 bytes, donde los primeros contienen el MBC, y a partir del byte 446 se encuentra la tabla de particiones (64 bytes). Por ultimo hay 2 bytes para firmar el MBR.

MBC

Es un pequeño código que permite arrancar el SO; la ultima acción del BIOS es leer y ejecutar dicho MBC. Si hay un SO instalado, se utiliza bootloader típico; sino se usa otro diferente (multietapa)

Particiones

Divisiones lógicas de disco físico. Sistemas como DOS o Windows 95 impiden manejar system files > 2GB. Cada SO instala su propia partición, la cual está formateada con un tipo de sistema de archivos. Se recomienda separar datos de usuario de aplicaciones, y también tener una partición de recuperación.

Limitaciones MBR

- Máximo 4 particiones primarias (3 primarias 1 extendida)
- Partición extendida puede tener particiones lógicas, donde no define un tipo de sistema de archivos directamente.

Tipos de particiones

- Primaria: Divisiones crudas (máx. 4 por disk)
- Extendida: Solo una por disco, contiene unidades lógicas
- Lógica: Parte de la extendida, con un sistema de archivos definido

NT: Se requiere mínimo una partición (/) y al menos dos (/ y SWAP). Para crear particiones se utilizan programas particionadores, que pueden ser tanto destructivos o no destructivos.

Particionado de discos

Los escenarios de particionado son para usar espacio libre no particionado, una partición no utilizada o usar espacio libre de una partición activa (esto es más complicado)

Cambio destructivo se refiere a que se pueden eliminar datos existentes. En cambio no destructivo modifica la partición sin perder datos.

Emuladores y Virtualizadores

- Emulación:
 - Emula hardware y debe implementar todas las instrucciones de la CPU, es costosa y poco eficiente y permite ejecutar arquitecturas diferentes a las

soportadas por el hardware

- Virtualización completa:
 - Logra ejecutar SO's huéspedes en un sistema anfitrión.
 - Utiliza hipervisor
 - El SO huésped debe ser compatible con la arquitectura anfitriona.
 - Más eficiente que la emulación
- Paravirtualización:
 - Permite ejecutar SO's modificados para entornos virtualizados. Y ofrece mayor eficiencia que la virtualización completa.

Diferencias

- Virtualizadores aprovechan mejor CPU.
- Emuladores puede correr cualquier arquitectura; en Virtualizadores solo la arquitectura virtualizada

Gestor de arranque

- Su propósito es cargar la imagen del Kernel del SO desde una partición para su ejecución, posterior a que el BIOS haya completado su tarea.
- Se pueden instalar en dos lugares: En el MBR, que utiliza el MBR gap, o en el Sector de arranque de la partición raíz, también conocido como Volume Boot Record

GRUB

- Grub Legacy:
 - Gestor de arranque múltiple
 - En el MBR, se encuentra en fase 1, carga 1.5

- Fase 1.5: Se encuentra en MBR gap, carga la fase 2
- Fase 2: Ofrece interfaz de usuario y carga kernel seleccionado.
- Está configurado a través de `/boot/grub/menu.lst`, donde se definen:
 - default: SO por defecto
 - timeout: Tiempo de espera para cargar el SO por defecto

GRUB 2:

- Es usado en la mayoría de las distros modernas. Esta mejora soporte para nuevas arquitecturas, y personalización de menús.
- No posee fase 1.5
- Su archivo de configuración está en `/boot/grub/grub.cfg`, que no se debe editar manualmente, se utiliza `grub update` para ello.

Proceso de Arranque

El proceso Bootstrap, inicia máquina y carga SO. En arquitecturas x86 el BIOS se encarga de dicha tarea mediante el MBR que está almacenado en un chip (ROM o NVRAM). En otras arquitecturas, hay equivalentes como Power on Reset.

Pasos:

- BIOS carga el programa de booteo desde MBR
- Gestor de arranque carga Kernel.
- Kernel habilita dispositivos
- Transfiere control al proceso `init`

Problemas del Arranque Basado en BIOS

El BIOS tiene limitaciones en la lectura de Sistemas de Archivos y el MBR está restringido a 446 bytes en el MBR.

EFI y GPT

- EFI: Conecta SO con el firmware
- GPT: Reemplaza al MBR, permitiendo así más particiones y mejor administración del disco.
- Características:
 - Mantiene MBR para compatibilidad con BIOS
 - Usa direccionamiento lógico (LBA) y no cylinder-header-sector
 - Almacena cabecera GPT y tabla de particiones al inicio y final de disco para redundancia.

EFI

UEFI FORUM: Alianza de empresas para modernizar el arranque de sistemas.

Características:

- Provee criptografía y autenticación de red
- Interfaz gráfica mejorada
- Define ubicación de gestor de arranque e interacción con firmware
- Ofrece información sobre hardware y servicios de firmware a los bootloaders.
- Incluye BootManager para cargar apps y driveres desde un file system UEFI.
- Bootloader, como Grub, se transforma en una app UEFI, eliminando necesidad de arranque en varias etapas.

Secure Boot

Garantiza arranque libre de código malicioso. Su funcionamiento es el de validar apps y drivers UEFI para asegurar que no han sido alterados. Este utiliza pares de claves asimétricas para la validación, y almacena claves públicas en el firmware.

Sin la clave privada está vencida o revocada, la verificación puede fallar