# Practica 1

1. Características de linux
   1. Es libre y open source (como mínimo los distro de debian), multitarea, multiusuario y todo es un archivo
   2. Windows. Corea del norte es mas libre que Windows, también es multitarea y multiusuario, no todo es archivo ni mucho menos
   3. Un proyecto de muchas aplicaciones de software libre
   4. En 1983 lo creo Richard Stallman con la idea de hacer el primero S.O libre, no consiguió que el sistema ande bien pero en paralelo fue creando muchas aplicaciones libres con la ayuda de la FSF para luego asociarse con Linus Torvals asi asociar esas aplicaciones al S.O Linux creando GNU/Linux, usando el kernel que hizo Linus y las aplicaciones de GNU.
   5. Si Linux es multitarea, es la capacidad de ejecutar muchos programas al mismo tiempo sin que uno interrumpa al otro (a bajo nivel no es que literalmente las cosas pasan al mismo tiempo pero a alto nivel da esa impresion)
   6. Es una API para programar aplicaciones para Linux asegurando la compatibilidad de estas en sistemas unix
2. Distribuciones de GNU/Linux
   1. Las distro son distitas versiones del Linux básico desarrolladas para uso propio o para colgarse en internet
      1. Knoppix. Distro basada en debian pensada para recuperar hardware dañado. Altamente compatible que hardware de intel
      2. Ubuntu. Distro basada en debian pensada para el usuario promedio como una alternativa directa de Windows
      3. Red hat. Una de las mas antiguas junto a debian y slackware pensaba para servidores
   2. En las ideas con las que se crearon, ya que cada organización enfoca su distro para el lado que quiere
   3. Debian es una de las distros mas viejas y usadas, además de un proyecto de software 100% libre, a diferencia de red hat o linspire por ejemplo que dejaron de ser del todo libres
   4. Kernel, Shell y bootloader
      1. / Tope de la estructura de directorios. Es como el C:\
      2. /home Se almacenan archivos de usuarios (Mis documentos)
      3. /var Informaci´on que var´ıa de tama˜no (logs, BD, spools)
      4. /etc Archivos de configuraci´on
      5. /bin Archivos binarios y ejecutables
      6. /dev Enlace a dispositivos
      7. /usr Aplicaciones de usuarios
3. 1. Es el núcleo del sistema,
      1. Lo inicio Linus Torvalds, cerca del 90, basándose en minix, (kernel similar a unix hecho por su profesor y con fines académicos), con la idea de hacer un S.O libre que funcionara en los Intel 80xxx, cerca de terminarlo lo posteo en internet, lo ayudaron mandándole consejos una vez terminado el proyecto se continuo mejorando en comunidad.
4. 1. Interprete de comandos de Linux original
   2. Interpreta comandos
   3. Shell, bash. Shell fue el interprete original basado en posix, bash una nueva versión con una mejor sintaxis
   4. /bin/sh para Shell y /bin/bash para bash
   5. Para que los usuarios puedan cambiarlo por el que les guste y por una cuestión de seguridad de no permitir que un usuario ejecute comandos o scripts a nivel de kernel.
   6. Si se puede, en /etc/passwd, se debe modificar con el usuario root, o usando sudo
5. 1. La forma en la que el sistema operativo organiza, guarda y recupera la información, para Windows es ntfs, FAT32 y para Linux ext4, ext3 o incluso FAT32 tambien
   2. Ext4, FAT32.
   3. Si se puede
   4. FHS = ***F****ilesystem****H****ierarchy****S****tandard,* es el estándar que define las jerarquias del sistema de archivos de linux  
      Estructura basica.
      1. / Tope de la estructura de directorios. Es como el C:\
      2. /home Se almacenan archivos de usuarios (Mis documentos)
      3. /var Informaci´on que var´ıa de tama˜no (logs, BD, spools)
      4. /etc Archivos de configuraci´on
      5. /bin Archivos binarios y ejecutables
      6. /dev Enlace a dispositivos
      7. /usr Aplicaciones de usuarios
6. 1. particiones
      1. particiones físicas, ventajas: son las principales, desventajas: solo se pueden tener 4
      2. partición extendida. Se utiliza para crear particiones logicas dentro de esta, sola como tal no se puede formatear ni mucho menos guardar cosas
      3. partición lógica: se crean dentro de la partición extendida, estas se pueden crear tantas como sean necesarias (en realidad tienen limite, pero ridículamente alto) lo malo es que no se deberían usar para guardar cosas del sistema operativo.
      4. Volúmenes lógicos: al igual que las particiones logicas no pueden tener un sistema operativo pero por otro lado se pueden agrandar de forma mucho mas sencilla y eficaz, pudiendo usar distintos discos para agrandar un mismo volumen, ej te quedas sin espacio en la unidad C:, te compras un disco lo metes en la pc y usas ese espacio para agrandar la unidad C: en lugar de crear otra, además de utilidades para hacer backups como por ejemplo almacenar la información en espejo con otro disco.
   2. HDx son dispositivos IDE (lectores de CD/DVD, discos duros IDE) y los dispositivos SDx son discos duros SCSI, y también SATA y dispositivos extraíbles.

La letra (hdA hdB...) distingue el dispositivo, en el caso de los dispositivos IDE hda será el maestro del IDE primario, hdb el esclavo del IDE primario, hdc sería el maestro del IDE secundario y hdd el esclavo del IDE secundario; en el caso de los SCSI o SATA cada dispositivo se distingue por su letra después del 'sd'.

Luego los números que siguen a los nombres de los dispositivos distinguen las particiones; hda1 significa la partición primaria del disco duro nombrado como hda y hdb5 sería una partición lógica de un segundo disco duro que está como esclavo en el IDE primario. Los cuatro primeros números están reservados para particiones primarias, de modo que las particiones lógicas se nombran desde el 5; así pues un disco duro con una partición primaria y dos lógicas serían nombradas hda1 la partición primaria y hada5 y hda6 las particiones lógicas.

* 1. El punto de montaje tiene que ser la partición que tiene /boot   
     una sola para pero es recomendable dos, una para la swap y otra para el sistema o /
     1. Swap : partición primaria o lógica, tipo de fs Linux-swap, se va a identificar como hda o sda.
     2. / o sistema operativo, partición primaria, y fs extx, hda o sda
  2. Para crearlas, se utiliza software denominado particionador. Existen 2 tipos:
     1. No destructivo: permiten crear, eliminar y modiﬁcar particiones (ﬁps, gparted) ← generalmente las distribuciones permiten hacerlo desde la interfaz de instalaci´on
     2. Destructivos: permiten crear y eliminar particiones (fdisk)

1. 1. Es el primer software que ejecuta, reconoce el hardware, lo prueba, si da errores avisa, por ejemplo si la ram no anda o no la detecta evita que arranque, luego cargar un gestor de arranque o un sistema operativo directamente
   2. 1. • Define la ubicaci´on de gestor de arranque
      2. • Define la interfaz entre el gestor de arranque y el firmware
      3. • Expone informaci´on para los gestores de arranque con:
      4. • Informaci´on de hardware y configuraci´on del firmware
      5. • Punteros a rutinas que implementan los servicios que el
      6. firmware ofrece a los bootloaders u otras aplicaciones UEFI
      7. • Provee un BootManager para cargar aplicaciones UEFI (e.j.: Grub) y drivers desde un UEFI filesystem
      8. • El booloader ahora es un tipo de aplicaci´on UEFI:
      9. • El Grub ser´a una aplicaci´on UEFI, que reside en el UEFI
      10. filesystem donde est´an los drivers necesarios para arrancar el sistema operativo (FAT32)
      11. • Para el Grub deja de ser necesario el arranque en varias etapas.
      12. Propone mecanismos para un arranque libre de c´odigo malicioso
      13. • Las aplicaciones y drivers UEFI (im´agenes UEFI) son validadas para verificar que no fueron alteradas
      14. • Se utilizan pares de claves asim´etricas
      15. • Se almacenan en el firmware una serie de claves publicas que sirven para validar que las im´agenes est´en firmadas por un proveedor autorizado
      16. • Si la clave privada est´a vencida o fue revocada la verificaci´on puede fallar
   3. Es un sector reservado del disco en el cual se carga el gestor de arranque y almacena la tabla de particiones, el MBC (master boot code) es la parte del MBR que permite arrancar el sistema operativo.
   4. GPT (GUID partition table) especifica la ubicaci´on y formato de la tabla de particiones en un disco duro, Puede verse como una sustituci´on del MBR. GPT usa un modo de direccionamiento l´ogico (logical block addressing LBA), El MBR “heredado” se almacena en el LBA 0, En el LBA 1 est´a la cabecera GPT. La tabla de particiones en s´ı est´a en los bloques sucesivos
   5. La finalidad del bootloader es la de cargar una imagen de Kernel (sistema operativo) de alguna particion para su ejecución, se ejecuta luego del c´odigo del BIOS, Existen 2 modos de instalaci´on, En el MBR (puede llegar a utilizar MBR gap), En el sector de arranque de la particion ra´ız o activa (Volume Boot Record). GRUB, LILO, NTLDR, GAG, YaST, etc
      1. La bios Carga el programa de booteo (desde el MBR)
      2. • El gestor de arranque lanzado desde el MBC carga el Kernel:
      3. • Prueba y hace disponibles los dispositivos
      4. • Luego pasa el control al proceso init
      5. • El proceso de arranque se ve como una serie de peque˜nos programas de ejecuci´on encadenada
      6. El BIOS realiza las tareas de inicio específicas de la [plataforma de hardware](https://es.wikipedia.org/wiki/Plataforma_de_hardware).
      7. Una vez que el [hardware](https://es.wikipedia.org/wiki/Hardware) es reconocido y se inicia correctamente, el BIOS carga y ejecuta el código de la partición de arranque del dispositivo de arranque designado, que contiene la fase 1 de un gestor de arranque Linux. La fase 1 carga la fase 2 (la mayor parte del código del gestor de arranque). Algunos cargadores pueden utilizar una fase intermedia (conocida como la fase 1.5) para lograr esto, ya que los modernos discos de gran tamaño no pueden ser totalmente leídos sin código adicional.
      8. El gestor de arranque a menudo presenta al usuario un menú de opciones posibles de arranque. A continuación, carga el sistema operativo, que descomprime en la memoria, y establece las funciones del sistema como del hardware esencial y la [paginación de memoria](https://es.wikipedia.org/wiki/Paginaci%C3%B3n_de_memoria), antes de llamar a la función start\_kernel().
      9. La función start\_kernel() a continuación realiza la mayor parte de la configuración del sistema ([interrupciones](https://es.wikipedia.org/wiki/Interrupciones), el resto de la [gestión de memoria](https://es.wikipedia.org/wiki/Gesti%C3%B3n_de_memoria), la inicialización del dispositivo, controladores, etc), antes de continuar por separado el proceso inactivo y planificador, y el proceso de *[Init](https://es.wikipedia.org/wiki/Init" \o "Init)* (que se ejecuta en el espacio de usuario).
      10. El planificador toma control efectivo de la gestión del sistema, y el núcleo queda dormido (inactivo).
      11. El proceso Init ejecuta secuencias de comandos ([*Scripts*](https://es.wikipedia.org/wiki/Scripts)) necesarios para configurar todos los servicios y estructuras que no sean del sistema operativo, a fin de permitir que el entorno de usuario sea creado y pueda presentarse al usuario con una pantalla de [inicio de sesión](https://es.wikipedia.org/wiki/Login).
   6. El proceso de shutdown arranca invocando al proceso init, este requiere que se le pase un parámeo de no hacerse se tomara por defecto el 1, siendo init 0 un apagado del sistema, init 6 para reiniciar, ini 1 dios sabe que
   7. Si, se puede, ya que al particionar se puede tener un formato distinto en cada partición, pudiendo tener varios sistemas que requieran distintos formatos y utilizando el bootloader seleccionar cual sistema operativo arrancar.
2. 1. Preguntar?