

LISTA DE COTEJO REPORTE DE INVESTIGACION VALOR: 30 %

NOMBRE DEL DOCENTE: María de los Ángeles Pelayo Vaquero		
<b>DATOS GENERALES DEL PROCESO DE EVALUACIÓN</b>		
NOMBRE DEL ALUMNO: FERMAN ATAXCA SARAHI ESMERALDA		
PRODUCTO: INVESTIGACIÓN 30%	UNIDAD: 1	PERIODO ESCOLAR: FEBRERO – JUNIO 2024

INDICADOR	VALOR	PORCENTAJE OBTENIDO
Presentación - Formato	2 %	2 %
Introducción Idea clara del contenido del trabajo, motivando al lector a continuar con su lectura y revisión	3 %	3 %
Desarrolla el objetivo	5 %	3 %
Desarrollo de la investigación La investigación cumple con el tema solicitado	10 %	8 %
Desarrolla la conclusión de investigación	4 %	4 %
Gramática y ortografía	3 %	3 %
Bibliografía	3 %	3 %
<b>Total</b>	<b>30 %</b>	<b>26 %</b>

LISTA DE COTEJO ENSAYO VALOR: 30 %

NOMBRE DEL DOCENTE: María de los Ángeles Pelayo Vaquero		
<b>DATOS GENERALES DEL PROCESO DE EVALUACIÓN</b>		
NOMBRE DEL ALUMNO: FERMAN ATAXCA SARAHI ESMERALDA		
PRODUCTO: ENSAYO 30%	UNIDAD: 1	PERIODO ESCOLAR: FEBRERO – JUNIO 2024

INDICADOR	VALOR	PORCENTAJE OBTENIDO
Presentación - Formato	3	3
Introducción Idea clara del contenido del trabajo, motivando al lector a continuar con su lectura y revisión	5	3
Cuerpo del ensayo (empleo de artículos científicos)	10	8
Conclusión	5	4
Uso de formato de estilo	2	1
Gramática y ortografía	3	3
Bibliografía (APA o IEEE)	2	2
<b>Total</b>	<b>30 %</b>	<b>24</b>

**ANEXOS**



**INSTITUTO TECNOLÓGICO SUPERIOR DE SAN ANDRÉS  
TUXTLA**



**NOMBRE DEL ALUMNO:**

SARAHÍ ESMERALDA FERMAN ATAXCA

**DOCENTE:**

MARIA DE LOS ANGELES PELAYO VAQUERO

**MATERIA:**

SISTEMAS OPERATIVOS

**SEMESTRE:**

4

**GRUPO:**

410\_A

**CARRERA:**

INGENIERIA EN INFORMATICA

**TEMA:**

SISTEMA OPERATIVO

**FECHA:**

09 DE FEBRERO DEL 2023, SAN ANDRES TUXTLA, VER

# INTRODUCCIÓN

Los sistemas operativos forman la esencia de la interacción entre el hardware y el software de un dispositivo. Funcionan como un intermediario crucial, gestionando recursos, facilitando la ejecución de programas y proporcionando una interfaz para que los usuarios interactúen de manera efectiva con sus dispositivos. Desde los sistemas operativos clásicos hasta las plataformas modernas, su evolución ha sido fundamental para la informática contemporánea. En esta conversación, podemos explorar aspectos específicos o resolver dudas sobre sistemas operativos

## **Objetivo general**

sistema operativo es gestionar eficientemente los recursos de hardware y proporcionar servicios a los programas de aplicación, facilitando así la interacción del usuario con la computadora, Analizar y comprender la estructura, características, función y clasificación de los sistemas Operativos

## **SISTEMA OPERATIVOS**

Cuando enciendes un ordenador portátil, lo primero que ves es cómo se inicia el sistema operativo, es decir, Windows, Linux o macOS, dependiendo del que tengas instalado. Solo cuando este sistema tan importante para el funcionamiento del dispositivo se ha cargado, puedes empezar a ejecutar programas, conectarte a Internet o abrir archivos. Sin embargo, el sistema operativo no se encarga de arrancar el dispositivo, sino que va precedido por la BIOS, el firmware responsable del proceso de inicio real.

Las capacidades de la BIOS son muy rudimentarias y están relacionadas principalmente con el proceso de arranque. Este elemento se encarga de iniciar el propio sistema operativo, que, a continuación, se hace cargo de gestionar todo el ordenador. Gran parte de este proceso tiene lugar en segundo plano. Windows y el resto de sistemas operativos también actúan como interfaces entre el usuario y el ordenador: estos completos programas permiten configurar los dispositivos, instalar y desinstalar programas o ejecutar software.

### **ESTRUCTURA**

El sistema operativo media entre el hardware y el resto del software. Para que los procedimientos de gestión funcionen, el sistema operativo presenta una estructura ligeramente distinta a la mayoría de programas. Podríamos decir que está formado por diferentes capas. El núcleo, su elemento más importante, se encuentra en la capa inferior, la más alejada de la interfaz de usuario. Por lo tanto, este programa también se carga primero. El núcleo es la interfaz que está en contacto directo con el hardware, iniciándolo y transmitiéndole los comandos de los programas que se ejecutan.

El resto de capas se superponen una encima de otra sobre el núcleo del sistema operativo, distanciándose cada vez más de la interacción con el hardware. Cada nivel solo se comunica con el que está por encima o por debajo. Finalmente, la última capa es la interfaz de usuario que, en estos términos, sería el nivel en que el propio usuario se comunica con el software.

## CLASIFICACIÓN

Existen diferentes formas de clasificar y subdividir a los sistemas operativos. A continuación, se enumerarán los criterios, y luego los diferentes grupos que se forman en base a ellos:

De acuerdo al modo de administración de tareas:

- Monotarea: Solamente puede ejecutar un en un momento dado. No puede interrumpir los procesos en acción.
- Multitarea: Es capaz de ejecutar varios procesos al mismo tiempo. Es capaz de asignar los recursos de forma alternada a los procesos que los solicitan, de manera que el usuario percibe que todos funcionan a la vez.

De acuerdo al modo de administración de los usuarios:

- Monousuario: Sólo permite ejecutar los programas de un usuario al mismo tiempo.
- Multiusuario: Si permite que varios usuarios ejecuten simultáneamente sus programas, accediendo a la vez a los recursos de la computadora.

De acuerdo a la forma de manejo de los recursos:

- Centralizado: Si permite usar los recursos de una sola computadora.
- Distribuido: Si permite utilizar los recursos de más de una computadora al mismo tiempo.

## **FUNCIONES**

El sistema operativo es el núcleo de un ordenador: sin este software complejo, el resto de programas no pueden funcionar. Las tareas de las que se encarga son de lo más diversas. Algunas se ejecutan completamente en segundo plano, y muchas ocurren en paralelo.

**Gestión del hardware:** Esta función, una de las más importantes del sistema operativo, suele quedar en segundo plano, es decir, que se oculta al usuario. El programa gestiona todo el hardware, tanto de entrada como de salida. Para ello, utiliza controladores proporcionados por los fabricantes del hardware, que le sirven para recibir y reenviar los comandos de los dispositivos, así como para transferir sus propios comandos al hardware. Así es como funcionan el teclado, el ratón, la pantalla, el disco duro, la tarjeta gráfica y todos los demás componentes de un ordenador.

**Gestión del software:** Por lo general, cuando te bajas algún programa de Internet, puedes elegir entre varias versiones para los diferentes sistemas operativos en la página de descarga, lo que demuestra hasta qué punto la programación de las aplicaciones está relacionada con las especificaciones del sistema operativo. Los sistemas de base de los ordenadores tienen interfaces que regulan la comunicación con todas las aplicaciones. De esta manera, es posible asignarles memoria, permitirles utilizar los recursos del procesador o ejecutar las acciones efectuadas con el teclado y el ratón.

**Gestión de archivos:** Si has redactado un documento, puedes imprimirlo (para lo cual el programa pasa la orden al sistema operativo, que a su vez la pasa a la impresora) o puedes guardarlo como archivo en una carpeta. El hecho de poder trabajar con una estructura de carpetas solo es posible gracias al sistema operativo, ya que en el propio disco duro no existe ese orden.

**Administración de derechos:** En ciertas situaciones, por ejemplo, en el ámbito empresarial, varias personas trabajan con un mismo dispositivo. Sin embargo, no todas deberían poder configurar el sistema. Por lo tanto, los sistemas operativos modernos permiten crear diferentes usuarios y otorgarles derechos individualmente. Además, cada cuenta puede protegerse con una contraseña.

**Orientación al usuario:** Todo el mundo debería poder utilizar un ordenador sin ningún problema, hasta las personas sin muchos conocimientos de informática. Por lo tanto, es importante que el sistema operativo facilite las funciones y opciones lo máximo posible. También los aspectos más importantes deberían ser fáciles de utilizar para el usuario básico, aunque muchos sistemas operativos (especialmente los de PC) ofrezcan opciones adicionales para los profesionales.

**Funciones de red:** Como el sistema operativo gestiona el hardware, también es responsable de la tarjeta de red y, por lo tanto, de la conexión a Internet y otras redes. Por lo general, es posible configurar el ordenador como un nodo de red a través del sistema operativo y, por ejemplo, asignarle una dirección IP concreta. En la configuración, también puedes introducir las especificaciones de la LAN y otras subredes para que el equipo pueda conectarse a otros ordenadores. La configuración de red también permite establecer individualmente el servidor DNS.

**Medidas de seguridad:** Tradicionalmente, la seguridad no es una tarea propia del sistema operativo, aunque también puede sumarse a sus funciones mediante un software adicional. Como los ordenadores constantemente conectados a Internet están expuestos a peligros, los sistemas operativos también han implementado sus propias medidas de seguridad. Por ejemplo, Windows ya lleva incorporado un firewall y un antivirus.

## **CARACTERÍSTICAS**

1. Tienen una gestión de trabajo: Una responsabilidad importante de cualquier programa operativo es planificar las tareas para que sean manejadas por el sistema de computadora.
2. Tienen una gestión de recursos: El manejo de recursos en un sistema de computadoras es otra de las principales preocupaciones del sistema operativo. Obviamente, un programa no puede utilizar un dispositivo si ese hardware no está disponible.
3. Tienen un control de operaciones input/output: La distribución de los recursos de un sistema está estrechamente relacionada con la operación del manejo de control input/output de un software.

Como a menudo es necesario el acceso a un dispositivo en particular antes de que las operaciones input/output puedan comenzar, el sistema operativo debe coordinar las operaciones input/output y los dispositivos en los que actúan.

4. Son sistemas multi tareas: La mayoría de los sistemas operativos modernos permiten que se cumplan múltiples tareas.
5. Cumplen con el proceso de memoria de espacio: En muchos sistemas operativos, cada proceso tiene su propio espacio de memoria. Esto quiere decir que esta memoria no está disponible a otros procesos. Se habla en este caso de la dirección de espacio del proceso.
6. Deben tener mecanismos asociados
7. Tienen un núcleo: La parte esencial de un sistema operativo es su núcleo. Hoy en día los núcleos monolíticos son los más populares.
8. Otorgan conexión: Probablemente, la característica que puede definir a los sistemas operativos es la conexión que provee entre el hardware, el software y, en algunos casos, el interfaz de usuario. Toda computadora requiere de un sistema operativo para poder ser utilizada.

# SISTEMA OPERATIVO LINUX



Fue iniciado por Richard Stallman el proyecto GNU con el propósito de crear un sistema operativo libre

Richard Stallman escribe la primera versión de la licencia GNU GPL



Es anunciado del núcleo Linux públicamente por Finlandés Linux Torvalds

El núcleo Linux es licenciado de nuevo bajo la GNU GPL. Las primeras distribuciones Linux son creadas



Más de 100 desarrolladores trabajan sobre el código de Linux. Este es adaptado al ambiente de GNU

Se presenta la versión 1.0, y esta por primera vez disponible en la red internet.

En marzo, la siguiente rama estable de Linux aparece, la serie 1.2. Más tarde, Linux es transportado a las plataformas informáticas DEC y SUNSPARC.



La versión 2.0 del núcleo Linux es liberada. Éste ahora puede servir varios procesadores al mismo tiempo, y así se hace una alternativa seria para muchas empresas

Adabas, Netscape, Office, Applixware son programas propietarios liberados para Linux en el Mercado

1983

1989

1991

1992

1993

1994

1995

1996

1997

1998

1999

2001

2002

2004

2007

2010

2015

Empresas importantes de informática como IBM, Compaq y Oracle anuncian soporte para Linux.

Aparece la serie 2.2 del núcleo Linux, en enero, con el código de red y el soporte a SMP mejorados



En enero, se libera la serie 2.4 del núcleo Linux. El núcleo Linux ahora soporta hasta 64 Gb de RAM, sistemas de 64 bits, dispositivos USB y un sistema de archivos journaling

La comunidad OpenOffice.org libera la versión 1.0 de su Suite de oficina homónima



Al final del año, la serie 2.6 del núcleo Linux es liberada, después de lo cual Linus Torvalds va a trabajar para el OSDL

Dell llega a ser el primer fabricante principal de computadoras en vender una computadora personal de escritorio con Ubuntu preinstalado

20 de octubre de 2010: se lanzó Linux 2.6.36 con 13.499.457 líneas de código



2 de febrero de 2015: fue lanzada la versión 3.19.

2 de abril de 2015: fue lanzada la versión 4.0

## **Conclusión**

El sistema operativo es muy importante en eso porque es beneficio por el bien de software y también de hardware es para que las cosas van procesando en esa lógica que es interesante.

Me di cuenta que en esos actualmente solo está utilizando en Windows, Mac, Android, etc. Porque es bueno para eso que es simplemente es completamente muy bueno de esos Sistemas Operativos requiere con beneficios y protege en esas tecnologías son cosas nuevas para el bien de computadora y de celular.

Lo bueno ya que la mayoría que está utilizando en Windows, Mac, Android, IOS, por eso lo bueno que es lo que está protegiendo como su propia cuenta y al menos si cuando las cosas pierden sus trabajos en escuela, empresa, etc. Y lo más ayuda es recuperar sus trabajos.

## Bibliografía

- [1] «sistema operativo,» [En línea]. Available: <https://www.ionos.mx/digitalguide/servidores/know-how/el-sistema-operativo/>. [Último acceso: 08 FEBRERO 2024].
- [2] «clasificaciones de los sistema operativos,» [En línea]. Available: <https://www.ejemplos.co/clasificacion-de-los-sistemas-operativos/>. [Último acceso: 08 febrero 2024].
- [3] «caracteristica de los sitema comparativo,» [En línea]. Available: <https://www.lifeder.com/caracteristicas-sistemas-operativos/>. [Último acceso: 08 febrero 2024].
- [4] «funciones de sistema operativos,» [En línea]. Available: <https://www.movistar.es/blog/mi-movistar/funciones-sistema-operativo-importancia/>. [Último acceso: 08 febrero 2024].



**INSTITUTO TECNOLÓGICO SUPERIOR DE SAN ANDRÉS  
TUXTLA**



**NOMBRE DEL ALUMNO:**

SARAHÍ ESMERALDA FERMAN ATAXCA

**DOCENTE:**

MARIA DE LOS ANGELES PELAYO VAQUERO

**MATERIA:**

SISTEMAS OPERATIVOS

**SEMESTRE:**

4

**GRUPO:**

410\_A

**CARRERA:**

INGENIERIA EN INFORMATICA

**FECHA:**

06 DE MARZO DEL 2023, SAN ANDRES TUXTLA, VER

## INTRODUCCIÓN

destaca el proceso fundamental de arranque de una computadora, desde el control inicial de la BIOS hasta la carga del Sistema Operativo (SO) a través del bootloader. Se menciona la importancia del gestor de arranque GRUB en sistemas operativos GNU/Linux, específicamente su versión moderna, GRUB 2. Además, se proporciona información sobre la estructura de versiones del kernel de Linux, diferenciando entre versiones de producción y desarrollo, así como la numeración y significado de sus componentes. Este contexto sienta las bases para comprender la complejidad y la importancia de los elementos clave en el inicio y funcionamiento de sistemas operativos basados en Linux.

La información también aborda aspectos esenciales como la estructura de las versiones del kernel de Linux, detallando la numeración y su significado. Se distinguen las versiones de producción, estables y recomendadas para uso regular, de las versiones de desarrollo, más experimentales y propensas a inestabilidades. Además, se explora el papel de los gestores de arranque múltiple en sistemas Linux, destacando ejemplos como GRUB y LILO.

La conversación también abarca la diversidad de distribuciones de Linux, subrayando que comparten el núcleo del sistema operativo, pero difieren en utilidades, aplicaciones y procedimientos de instalación. Se mencionan varias distribuciones conocidas, desde Debian hasta Ubuntu, destacando su relevancia tanto en entornos domésticos como empresariales.

Asimismo, se proporciona información sobre el formato de versiones del kernel, el sistema de gestión de paquetes en distribuciones de Linux, y se explora la variedad de interfaces de usuario disponibles, como las Interfases Gráficas de Usuario (GUI) y las Consolas de Línea de Comandos (CLI). Además, se presenta un vistazo a los diferentes tipos de shell disponibles en sistemas Linux, cada uno con sus características y utilidades. Esta amalgama de información ofrece una visión integral de los elementos fundamentales en el arranque, desarrollo y funcionamiento de sistemas operativos basados en Linux.

## **VERSION DEL KERNEL**

El Kernel de Linux tiene dos versiones: de producción y de desarrollo.

La versión de producción es estable y recomendada para uso normal, mientras que la de desarrollo es experimental y utilizada por desarrolladores para nuevas características.

La numeración del kernel sigue el formato VV,XX,YY,ZZ, donde VV indica la serie principal, XX antes de la versión 2.6 indicaba desarrollo o producción, YY señala nuevas revisiones, y ZZ indica parches de seguridad y corrección de errores. Letras como "rc" indican lanzamientos no oficiales, y otras letras pueden representar desarrolladores específicos o bifurcaciones en el desarrollo del núcleo.

## **LAS DISTRIBUCIONES DE LINUX**

Como Debian o RedHat, son conjuntos de utilidades basadas en diferentes versiones del kernel. Estas distribuciones incluyen paquetes de software gestionados por sistemas como RPM o APT, facilitando la instalación y actualización de aplicaciones. Cada paquete contiene información sobre el software y sus dependencias.

El paquete es generalmente distribuido en su versión compilada y la instalación y desinstalación de los paquetes es controlada por un sistema de gestión de paquetes. Cada paquete contiene meta-información tal como fecha de creación, descripción del paquete y sus dependencias. El gestor de paquetes permite la búsqueda de paquetes, actualizar las librerías y aplicaciones instaladas, y obtener todas las dependencias. Algunos de los sistemas de paquetes más usados son:

- creado por Red Hat y usado por un gran número de distribuciones de Linux, es el formato de paquetes del Linux Standard Base.
- , paquetes Debian, originalmente introducidos por Debian, pero también utilizados por otros como Knoppix y Ubuntu.

## lsb\_release muestra información de la distribución GNU/Linux del equipo:

**-a, --all** Muestra toda la información

**-i, --id** Muestra el nombre de la distribución GNU/Linux

**-d, --description** Muestra la descripción de la distribución

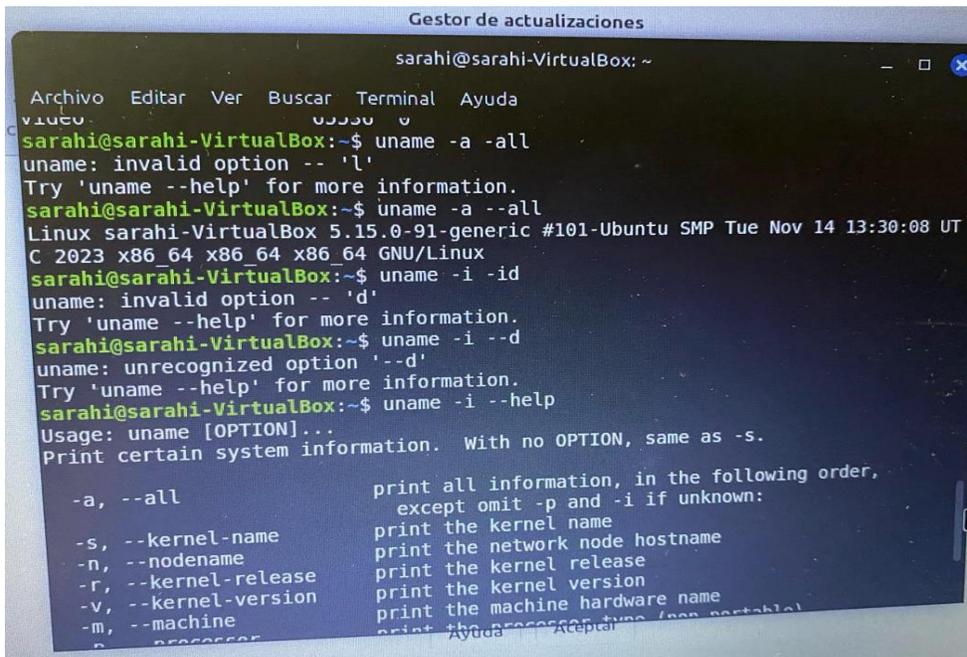
**-r, --release** Muestra la versión de la distribución

**-c, --codename** Muestra el código de la distribución

**-s, --short** Muestra la información solicitada en formato corto, sin el título inicial.

**-h, --help** Muestra la ayuda y finaliza

**-v, --version** Muestra la versión de la especificación LSB de la que es compatible y finaliza



```
Gestor de actualizaciones
sarahi@sarahi-VirtualBox: ~
Archivo Editar Ver Buscar Terminal Ayuda
vluco 0000 0
sarahi@sarahi-VirtualBox:~$ uname -a -all
uname: invalid option -- 'l'
Try 'uname --help' for more information.
sarahi@sarahi-VirtualBox:~$ uname -a --all
Linux sarahi-VirtualBox 5.15.0-91-generic #101-Ubuntu SMP Tue Nov 14 13:30:08 UT
C 2023 x86_64 x86_64 x86_64 GNU/Linux
sarahi@sarahi-VirtualBox:~$ uname -i -id
uname: invalid option -- 'd'
Try 'uname --help' for more information.
sarahi@sarahi-VirtualBox:~$ uname -i --d
uname: unrecognized option '--d'
Try 'uname --help' for more information.
sarahi@sarahi-VirtualBox:~$ uname -i --help
Usage: uname [OPTION]...
Print certain system information.  With no OPTION, same as -s.

-a, --all          print all information, in the following order,
                  except omit -p and -i if unknown:
-s, --kernel-name  print the kernel name
-n, --nodename     print the network node hostname
-r, --kernel-release print the kernel release
-v, --kernel-version print the kernel version
-m, --machine      print the machine hardware name
                  print the processor type (non-portable)
```

```
Gestor de actualizaciones
sarahi@sarahi-VirtualBox: ~
Archivo  Editor  Ver  Buscar  Terminal  Ayuda
uname: unrecognized option '--d'
Try 'uname --help' for more information.
sarahi@sarahi-VirtualBox:~$ uname -i --help
Usage: uname [OPTION]...
Print certain system information.  With no OPTION, same as -s.

-a, --all          print all information, in the following order,
                   except omit -p and -i if unknown:
-s, --kernel-name  print the kernel name
-n, --nodename     print the network node hostname
-r, --kernel-release print the kernel release
-v, --kernel-version print the kernel version
-m, --machine      print the machine hardware name
-p, --processor    print the processor type (non-portable)
-i, --hardware-platform print the hardware platform (non-portable)
-o, --operating-system print the operating system
--help            display this help and exit
--version         output version information and exit

GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Report any translation bugs to <https://translationproject.org/team/>
Full documentation <https://www.gnu.org/software/coreutils/uname/>
or available locally via: info '(coreutils) uname invocation'
sarahi@sarahi-VirtualBox:~$
```

## Tipos de Shell

LINUX dispone de varios programas que se encargan de interpretar los comandos que introduce el usuario y realiza las acciones oportunas en respuesta, estos programas se denominan shell. El shell es capaz de interpretar una gran gama de comandos y sentencias. Permite construir programas y comandos, llamados shellscrips, que nos dan la posibilidad de automatizar diversas tareas.

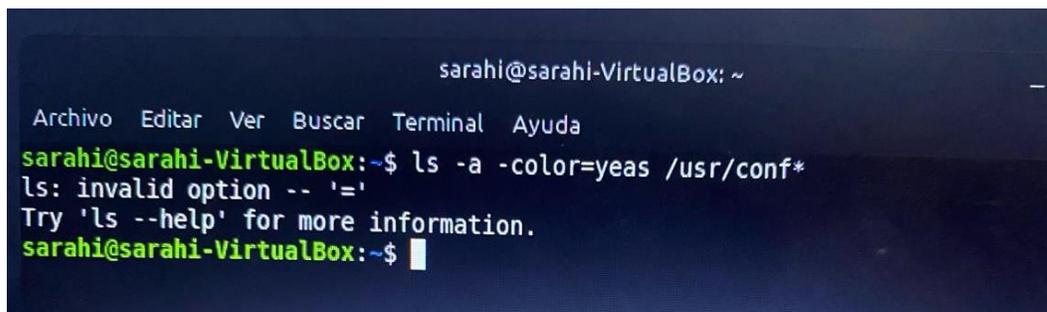
Existen varios tipos de shells orientados a caracteres (modo texto). A estos shells los encontramos como archivos ejecutables en el directorio /bin. Los shells más utilizados son

- ash: A shell
- csh: C shell
- tcsh: una extensión al C shell
- ksh: Korn Shell
- bsh: Bourne Shell
- bash: Bourne Again Shell

## FORMATO DE LA LÍNEA DE COMANDOS

un comando es cualquier archivo ejecutable. Por lo que cada usuario puede construir archivos con una estructura especial que le permita ser ejecutados, y añadirlo a su lista de comandos. Podrá ejecutar comandos desde una terminal de texto. Linux utiliza un formato sencillo para la línea de comandos, en donde la mayoría de ellos está formada por sólo dos letras.

- El nombre del comando es siempre necesario.
- Los comandos siempre van escritos en letra minúscula y la mayoría de las opciones también.
- Las opciones son caracteres o adverbios que siempre van precedidos de un guion.
- Los argumentos pueden o no ser necesarios, sin embargo los nombres de archivos que especifique pueden ser rutas de acceso relativas o absolutas. Por ejemplo: `ls ../../dev`
- En los argumentos podemos utilizar metacaracteres, como son `*` y `?`
- Los espacios son importantes ya que son separadores de campo por defecto.



```
sarahi@sarahi-VirtualBox: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
sarahi@sarahi-VirtualBox:~$ ls -a -color=yeas /usr/conf*  
ls: invalid option -- '='  
Try 'ls --help' for more information.  
sarahi@sarahi-VirtualBox:~$
```

En Linux, tras la instalación, se tiene la opción de interactuar con el sistema a través de una Interfaz de Línea de Comandos (CLI) o una Interfaz Gráfica de Usuario (GUI). La GUI se basa en el sistema de ventanas X Window y ofrece entornos de escritorio como GNOME o KDE. Estos entornos facilitan el acceso y configuración con barras de herramientas, íconos y widgets. Sin embargo, para un control total, se recurre a la interfaz de línea de comandos (CLI).

El sistema gráfico X Window permite elegir entre varios gestores de ventanas como AfterStep, Kwin, Metacity, FVWM, entre otros. Mientras tanto, la CLI permite la multitarea multiusuario mediante consolas de texto accesibles con combinaciones de teclas como Ctrl+ALT+F1.

En las consolas de texto, cada sesión se asocia con un dispositivo tty. Al ingresar a la interfaz gráfica desde una consola de texto (por ejemplo, con Ctrl+ALT+F7), se puede acceder a un entorno de escritorio y abrir terminales desde allí. Los emuladores de terminal incluyen Gnome Terminal, Konsole y LXterm, asociados a archivos pts en /dev.

## **CONCEPTO DE ARRANQUE Y GESTOR DE ARRANQUE GRUB**

El proceso de arranque, o bootstrapping, inicia con la BIOS al encender la computadora, seguido por el POST y la carga del MBR. El bootloader, como GRUB en Linux, identifica unidades booteables y carga el Sistema Operativo en multi-etapas. El kernel realiza diversas tareas, como detectar la CPU y montar la partición raíz. El proceso init, el primer proceso en ejecución, lee /etc/inittab y ejecuta procesos rc para la inicialización del sistema. En terminales virtuales, init ejecuta /etc/getty para permitir conexiones de usuarios y luego el proceso login abre sesiones de trabajo. GRUB, un gestor de arranque múltiple, carga el sistema operativo en dos fases, presentando un menú de inicio en la segunda fase para que el usuario elija el SO.

Además, el gestor de arranque GRUB (Grand Unified Bootloader) se usa comúnmente en sistemas operativos GNU/Linux y ha evolucionado a GRUB 2. Este realiza la carga del sistema

operativo en dos fases: la primera fase, gestionada por la BIOS, carga y ejecuta el código en el MBR; la segunda fase presenta un menú de inicio para que el usuario elija el sistema operativo deseado. En resumen, el proceso abarca desde la BIOS hasta la selección del sistema operativo a través de GRUB.

# EVALUACION UNIDAD 1. SISTEMAS OPERATIVOS

INGENIERIA INFORMATICA ITSSAT ESCOLARIZADO

Se ha registrado el correo del encuestado (221u0499@alumno.itssat.edu.mx) al enviar este formulario.

NOMBRE Y APELLIDOS \*

SARAHÍ ESMERALDA FERMAN ATAXCA

GRUPO: \*

410-A

En el sistema de capas, el sistema THE la capa 2 se encargaba de la comunicación entre cada proceso y la consola del operador (es decir, el usuario). Encima de esta capa, cada proceso tenía en efecto su propia consola de operador. \* 5 puntos

V

F

La idea básica detrás del diseño del kernel es lograr una alta confiabilidad al dividir el sistema operativo en módulos pequeños y bien definidos, solo uno de los cuales (el kernel) se ejecuta en modo kernel y el resto se ejecuta como procesos de usuario ordinarios, sin poder relativamente. En especial, al ejecutar cada driver de dispositivo y sistema de archivos como un proceso de usuario separado, un error en alguno de estos procesos puede hacer que falle ese componente, pero no puede hacer que falle todo el sistema \* 5 puntos

V

F

La comunicación entre clientes y servidores se lleva a cabo comúnmente mediante el paso de mensajes. Para obtener un servicio, un proceso servidor construye un mensaje indicando lo que desea y lo envía al servicio apropiado. después el servicio hace el trabajo y envía de vuelta la respuesta. Si el cliente y el servidor se ejecutan en el mismo equipo se pueden hacer ciertas optimizaciones, pero en concepto estamos hablando sobre el paso de respuestas. \* 5 puntos

V

F

Distintas máquinas virtuales pueden (y con frecuencia lo hacen) ejecutar distintos sistemas operativos. En el sistema VM/370 original, algunas ejecutaban SO/36 o uno de los otros sistemas operativos extensos de procesamiento por lotes o de procesamiento de transacciones, mientras que otros ejecutaban un sistema interactivo de un solo usuario llamado CSM (Conversational Monitor System; Sistema monitor conversacional) para los usuarios interactivos de tiempo compartido. Este último fue popular entre los programadores.

\* 5 puntos

 V F

Un proceso es en esencia un programa en ejecución. Cada proceso tiene asociado un espacio de direcciones, una lista de ubicaciones de memoria que va desde algún mínimo (generalmente 0) hasta cierto valor máximo, donde el proceso puede leer y escribir información. El espacio de direcciones contiene el programa fuente, los datos del programa y su pila.

\* 5 puntos

 V F

Las llamadas al sistema de administración de procesos clave son las que se encargan de la creación y la terminación de los procesos. Considere un ejemplo común. Un proceso llamado interprete de comandos o shell lee comandos de una terminal. El usuario acaba de escribir un comando, solicitando la compilación de un programa. El shell debe entonces crear un proceso para ejecutar el compilador. Cuando ese proceso ha terminado la compilación, ejecuta una llamada al sistema para terminarse a sí mismo.

\* 5 puntos

 V F

Las jerarquías de procesos y de archivos están organizadas en forma de árboles, pero la similitud se detiene ahí. Por lo general, las jerarquías de procesos no son muy profundas (más de tres niveles es algo inusual), mientras que las jerarquías de archivos son comúnmente de cuatro, cinco o incluso más niveles de profundidad. Es común que las jerarquías de procesos tengan un tiempo de vida corto, por lo general de minutos a lo más, mientras que la jerarquía de directorios puede existir por años.

\* 5 puntos

 V F

Cuando cualquier usuario inicia sesión, se inicia un shell. El shell tiene la terminal como entrada estándar y salida estándar.

\* 5 puntos

Empieza por escribir el indicador de comandos (prompt), un carácter tal como un signo de dolar, que indica al usuario que el shell está esperando aceptar un comando. Por ejemplo, si el usuario escribe **date** el shell crea un proceso hijo y ejecuta el programa date como el hijo. Mientras se ejecuta el proceso hijo, el shell espera a que termine. Cuando el hijo termina, el shell escribe de nuevo el indicador y trata de leer la siguiente línea de entrada.

 V F

Este formulario se creó en INSTITUTO TECNOLÓGICO SUPERIOR DE SAN ANDRÉS TUXTLA.

Google Formularios