

CURSO PARA LA OBTENCIÓN DEL

DIPLOMA DE INFORMÁTICA MILITAR

(59130)

* Teoría de Sistemas Operativos

**PORFOLIO**

Alumno: **Xavier Guerrero Fernández**

Fecha de cierre del porfolio: 31 de octubre de 2023

Un porfolio es una colección de documentos del trabajo del estudiante que exhibe su esfuerzo, progreso y logros. Es una forma de recopilar la información que demuestra las habilidades y logros de los estudiantes. Puede servir como forma de evaluación y de autoevaluación.

[1 Capítulo 1. Introducción al Sistema Operativo 13](#_Toc148268507)

[1.1 Preguntas 13](#_Toc148268508)

[1.1.1 Explicar el propósito principal de un sistema operativo 13](#_Toc148268509)

[1.1.2 Características de la vista de usuario del SO. ¿Qué herramientas utiliza el usuario en esta vista para enviar órdenes al sistema operativo? Algunas computadoras tienen poca o ninguna vista de usuario. Pon ejemplos 13](#_Toc148268510)

[1.1.3 Al parecer no hay una estricta separación entre los programas kernel y los programas de aplicación. Exponga cuatro tipos de programas que podíamos incluir en el SO 13](#_Toc148268511)

[1.1.4 Comentar las peculiaridades de los SO en general en cuanto a su dimensionamiento en líneas de código, su prolongado ciclo de vida y su complejidad 13](#_Toc148268512)

[1.1.5 Modelo conceptual del sistema operativo en una perspectiva de arriba hacia abajo. Ídem de abajo hacia arriba. 14](#_Toc148268513)

[1.1.6 Aunque hay muchos profesionales de las ciencias de la computación, solo un pequeño porcentaje de ellos estará involucrado en la creación o modificación de un sistema operativo. Indique cuatro razones para estudiar la teoría de los sistemas operativos en este curso. 14](#_Toc148268514)

[1.1.7 En 1968 Ken Thompson con una pequeña minicomputadora PDP-7 abandonada se propuso escribir una versión simplificada de MULTICS para un solo usuario. ¿Por qué tenía la capacidad de desarrollar un SO? ¿Qué importancia tuvo este hecho en la historia de los SO? 14](#_Toc148268515)

[1.1.8 Enunciado falso: “1960 Doug Engelbart en el Instituto de Investigación de Stanford inventó la interfaz gráfica de usuario, completa con ventanas, íconos, menús y ratón, encargado por Apple. Pero Steve Jobs, quien coinventó la computadora de Xerox PARC compró la interface gráfica de Engelbart para sus máquinas. En 1999, Apple adoptó un núcleo derivado de BSD UNIX que reemplazó al micronúcleo Mach de la Universidad Carnegie Mellon. Por eso el macOS de Apple no es un sistema operativo basado en UNIX”. 15](#_Toc148268516)

[Rectifique adecuadamente ese falso enunciado 15](#_Toc148268517)

[1.1.9 Explicar las diferencias claras entre driver de dispositivo y controlador de dispositivo. 15](#_Toc148268518)

[1.1.10 Mediante el esquema que se acompaña, explicar el concepto IRQ en el SO 15](#_Toc148268519)

[1.1.11 Diferencia entre trampa (TRAP) e interrupción utilizando la siguiente tabla: 16](#_Toc148268520)

[1.1.12 Respecto a la estructura de almacenamiento en niveles, sustituya las manchas rojas por los conceptos: rápido, lento, pequeño, grande, memoria volátil, memoria no volátil, almacenamiento primario, secundario y terciario 16](#_Toc148268521)

[1.1.13 Qué estructura de datos es la de la figura y ponga ejemplos donde el SO la emplea 17](#_Toc148268522)

[1.1.14 Qué estructura de datos es la de la figura y ponga ejemplos donde el SO la emplea 17](#_Toc148268523)

[1.1.15 Un árbol rojo-negro es un árbol binario de búsqueda en el que cada nodo tiene un atributo de color cuyo valor es rojo o negro. Además de los requisitos impuestos a los árboles binarios de búsqueda convencionales, enuncie cuatro reglas para tener un árbol rojo-negro válido. Linux utiliza un árbol rojo-negro (rbtree) en muchos lugares del kernel, indique tres: 17](#_Toc148268524)

[2 Capítulo 2: Llamadas y tipos de sistemas operativos. 18](#_Toc148268525)

[2.1 Preguntas 18](#_Toc148268526)

[2.1.1 ¿Cuál es el propósito del intérprete de comandos? ¿Por qué suele estar separado del núcleo? 18](#_Toc148268527)

[2.1.2 Ventajas de CLI 18](#_Toc148268528)

[2.1.3 Ventajas de GUI 18](#_Toc148268529)

[2.1.4 ¿Cuál es el propósito de las llamadas al sistema? 19](#_Toc148268530)

[2.1.5 Diferencias entre la llamada a una función en el dominio de aplicaciones de usuario y una llamada al sistema. 19](#_Toc148268531)

[2.1.6 Indique los seis tipos de llamadas al sistema que generalmente tienen los SO 19](#_Toc148268532)

[2.1.7 ¿Que es POSIX? 19](#_Toc148268533)

[2.1.8 ¿Cuáles son los tipos de SO según estructura de kernel? De una explicación de una línea de cada uno de ellos 19](#_Toc148268534)

[2.1.9 Comparación de un kernel monolítico de un microkernel 20](#_Toc148268535)

[2.1.10 ¿Qué tipo de núcleo llevan los siguientes SO? 20](#_Toc148268536)

[2.1.11 Utilizando la tabla a continuación, establecer las diferencias entre el kernel de un S.O. monolítico con módulos recargable y un S.O. híbrido 20](#_Toc148268537)

[3 Capítulo 3. Procesos 21](#_Toc148268538)

[3.1 Preguntas 21](#_Toc148268539)

[3.1.1 ¿Qué es un proceso del sistema operativo? ¿Qué recursos del sistema necesita un proceso para realizar satisfactoriamente su tarea? 21](#_Toc148268540)

[3.1.2 ¿Cuales son las obligaciones del S.O. como gestor de procesos? 21](#_Toc148268541)

[3.1.3 ¿Qué suele almacenarse en cada una de las secciones de la memoria ocupada por un proceso? 21](#_Toc148268542)

[3.1.4 Se ha ejecutado el comando “size /bin/more”. Explicar la salida obtenida 21](#_Toc148268543)

[3.1.5 Respecto a los estados de un proceso, rellene adecuadamente la tabla 22](#_Toc148268544)

[3.1.6 ¿Qué es un PCB? ¿Qué información generalmente se suele guardar en un PCB? 22](#_Toc148268545)

[3.2 Ejercicios 22](#_Toc148268546)

[4 Capítulo 4. Multithreads y concurrencia 23](#_Toc148268547)

[4.1 Preguntas 23](#_Toc148268548)

[4.1.1 Se dice que existe una pseudosimultaneidad cuando: 23](#_Toc148268549)

[4.1.2 Respecto a multithreading: 23](#_Toc148268550)

[4.1.3 Se dispone de un ordenador monoprocesador que permite multiprogramación, para lo cual es preciso: 23](#_Toc148268551)

[4.1.4 En relación con los Sistemas Operativos, ¿Qué se entiende por overhead? 23](#_Toc148268552)

[4.1.5 Uno de los beneficios de la programación multiproceso (concurrente) pueden ser (marcar la respuesta FALSA): 23](#_Toc148268553)

[4.1.6 Un kernel de un buen sistema operativo multinúcleo dispone de programas que deben resolver gestiones de multithreding como (SEÑALE LA FALSA): 24](#_Toc148268554)

[4.1.7 La tecnología HyperThreading consiste en 24](#_Toc148268555)

[4.1.8 Una tecnología que “engaña” al sistema operativo al hacerle creer que existen dos núcleos cuando en realidad solo existe uno 24](#_Toc148268556)

[4.1.9 Aplicando la Ley de Amdahl, supongamos que tenemos una aplicación que es 50 por ciento paralela y 50 por ciento serie. Si ejecutamos esta aplicación en un sistema con dos núcleos de procesamiento, podemos obtener una aceleración de: 25](#_Toc148268557)

[4.1.10 El soporte para subprocesos se puede proporcionar a nivel de usuario, o por el kernel 25](#_Toc148268558)

[4.1.11 El modelo de gestión de subprocesos del SO de tipo muchos a uno: 25](#_Toc148268559)

[4.1.12 El modelo de gestión de subprocesos del SO de tipo uno a uno: 25](#_Toc148268560)

[4.2 Ejercicios 25](#_Toc148268561)

[4.2.1 Comandos sobre información de CPU 25](#_Toc148268562)

[4.2.2 ¿Cómo sabemos si tenemos activado el Hyper Threading? 26](#_Toc148268563)

[4.2.3 ¿Cómo contar el número de Threads que está utilizando un proceso de Linux? 27](#_Toc148268564)

[5 Capítulo 5. Scheduling (planificación) 28](#_Toc148268565)

[5.1 Preguntas: 28](#_Toc148268566)

[5.1.1 Además de elegir el proceso idóneo a ejecutar, el scheduler también tiene que hacer un uso eficiente de la CPU, ya que el cambio de contexto es costoso. Indique cúal de las siguientes acciones pone en evidencia el coste computacional de cambio de contexto 28](#_Toc148268567)

[5.1.2 Respecto a que casi todos los procesos alternan ráfagas de computación (instrucciones aritmético lógicas o transferencias con la memoria) con solicitudes de E/S (de disco, red, etc), indique la respuesta verdadera: 28](#_Toc148268568)

[5.1.3 Respecto a la alternancia en los procesos de ráfagas de computación (instrucciones aritmético lógicas o transferencias con la memoria) con solicitudes de E/S (de disco, red, etc), indique la respuesta verdadera: 28](#_Toc148268569)

[5.1.4 Explique brevemente la planificación apropiativa (preemptive) 28](#_Toc148268570)

[5.1.5 Explique brevemente la planificación no apropiativa (no preemptive) 29](#_Toc148268571)

[5.2 Ejercicios: 29](#_Toc148268572)

[5.2.1 Observe la utilización de la CPU mediante el comando monitor en los sistemas Linux, macOS (Figura 4) y UNIX 29](#_Toc148268573)

[5.2.2 Comando vmstat 29](#_Toc148268574)

[5.2.3 Contenido del fichero status de un proceso 29](#_Toc148268575)

[5.2.4 Ejercicio sobre SJF 30](#_Toc148268576)

[5.2.5 Ejercicio de algoritmo de planificación RR 30](#_Toc148268577)

[6 Capítulo 6. Sincronismo 31](#_Toc148268578)

[6.1 Preguntas 31](#_Toc148268579)

[6.1.1 ¿Cuándo se establece la exclusión mutua? 31](#_Toc148268580)

[6.1.2 ¿Qué técnicas se usan en la exclusión mutua? 31](#_Toc148268581)

[6.1.3 ¿Qué es una red de Petri? 31](#_Toc148268582)

[6.1.4 ¿Por qué es útil Red de Petri en el diseño y estudio de sistemas operativos? 31](#_Toc148268583)

[6.1.5 Diferencias entre cerrojos y semáforos 31](#_Toc148268584)

[6.2 Ejercicio 32](#_Toc148268585)

[6.2.1 Diseñar el modelo Productor Consumidor utilizando la herramienta para Redes de Petri PIPE (https://github.com/sarahtattersall/PIPE/releases) 32](#_Toc148268586)

[6.2.2 Crear un modelo de MUTEX (exclusión mutua) con Red de Petri 33](#_Toc148268587)

[6.2.3 Diseñar una Red de Petri que emule el problema de los filósofos cenando como modelo de 5 procesos concurrente que se disputan recursos comunes (tenedores). Explicar los resultados 33](#_Toc148268588)

[7 Capítulo 7. Memoria 34](#_Toc148268589)

[7.1.1 Mencione dos diferencias entre direcciones lógicas y físicas. 34](#_Toc148268590)

[7.1.2 ¿Por qué los tamaños de página son siempre potencias de 2? 34](#_Toc148268591)

[7.1.3 ¿Cuáles son las funciones de una MMU? 34](#_Toc148268592)

[7.1.4 Considere un espacio de direcciones lógicas de 64 páginas de 1024 palabras cada una, mapeadas en una memoria física de 32 marcos. ¿Cuántos bits hay en la dirección lógica? ¿Cuántos bits hay en la dirección física? 34](#_Toc148268593)

[7.1.5 ¿Cuál es el efecto de permitir que dos entradas en una tabla de páginas apunten al mismo marco de página en la memoria? Explique cómo podría usarse este efecto para disminuir la cantidad de tiempo necesario para copiar una gran cantidad de memoria de un lugar a otro. ¿Qué efecto tendría la actualización de algún byte en una página en la otra página? 35](#_Toc148268594)

[7.1.6 6 Suponiendo un tamaño de página de 1 KB, ¿cuáles son los números de página y los desplazamientos para las siguientes referencias de direcciones (proporcionadas como números decimales): 35](#_Toc148268595)

[7.1.7 Considere un espacio de direcciones lógicas de 256 páginas con un tamaño de página de 4 KB, mapeado en una memoria física de 64 marcos. 35](#_Toc148268596)

[¿Cuántos bits se requieren en la dirección lógica? 35](#_Toc148268597)

[b. ¿Cuántos bits se requieren en la dirección física? 36](#_Toc148268598)

[7.1.8 Un ordenador emplea 32 bits para las direcciones virtuales de 32 bits. Dispone de una memoria principal de 512 MB, que emplea páginas de 4 KB. Indique el formato de la dirección virtual (número de bits para páginas y bits para desplazamiento) y el número de marcos de la página de este ordenador. 36](#_Toc148268599)

[7.1.9 9 ¿Qué es la memoria SWAP? ¿Dónde se encuentra la memoria swap? ¿cuándo es necesaria? ¿Qué inconvenientes tiene? 36](#_Toc148268600)

[7.2 EJERCICIOS 36](#_Toc148268601)

[7.2.1 ¿Por qué Windows consume tanta RAM? (superfetch) 36](#_Toc148268602)

[7.2.2 ¿Qué es la presión de la memoria y cómo puedo aliviarla? 37](#_Toc148268603)

[8 Capítulo 8. Entrada/Salida 38](#_Toc148268604)

[8.1 Preguntas 38](#_Toc148268605)

[8.1.1 ¿Qué función realiza un controlador de dispositivo en un sistema operativo? 38](#_Toc148268606)

[8.1.2 ¿Qué es la E/S por bloques en sistemas operativos? 38](#_Toc148268607)

[8.1.3 ¿Qué es una interrupción de hardware en un sistema operativo? 38](#_Toc148268608)

[8.1.4 ¿Qué significa DMA en el contexto de E/S en sistemas operativos? 38](#_Toc148268609)

[8.1.5 ¿Cuál de las siguientes no es una función de un controlador de dispositivo? 38](#_Toc148268610)

[8.1.6 ¿Qué es el pooling de E/S en sistemas operativos? 38](#_Toc148268611)

[8.1.7 ¿Cuál es el propósito principal de las interrupciones de software en un sistema operativo? 38](#_Toc148268612)

[8.1.8 ¿Qué tipo de interrupción es generada por dispositivos de hardware para notificar eventos como la finalización de una transferencia de datos? 39](#_Toc148268613)

[8.1.9 ¿Cuál es el propósito de una tabla de vectores de interrupción en un sistema operativo? 39](#_Toc148268614)

[8.1.10 ¿Qué beneficio proporcionan las interrupciones en un sistema operativo? 39](#_Toc148268615)

[8.2 Ejercicios 39](#_Toc148268616)

[8.2.1 Comando para ver interrupciones 39](#_Toc148268617)

[8.2.2 Comando para ver una lista de interrupciones 39](#_Toc148268618)

[8.2.3 Comando para ver la afinidad 39](#_Toc148268619)

[8.2.4 Lista de los núcleos de CPU a los que se puede asignar una interrupción 40](#_Toc148268620)

[8.2.5 Dispositivos PCI en el sistema 40](#_Toc148268621)

[8.2.6 Módulos kernel cargados en el sistema 40](#_Toc148268622)

[8.2.7 Lista de los dispositivos en el directorio /dev 40](#_Toc148268623)

[8.2.8 Lista de dispositivos USB conectados al sistema 40](#_Toc148268624)

[8.2.9 Detección y carga de controladores de dispositivos 41](#_Toc148268625)

[8.2.10 Configuración de hardware del sistema, incluyendo la lista de dispositivos y controladores asociados 41](#_Toc148268626)

[8.2.11 Información sobre un módulo del kernel, incluyendo detalles sobre el controlador de dispositivo 41](#_Toc148268627)

[8.2.12 Información sobre los dispositivos de bloque 41](#_Toc148268628)

[9 Capítulo 9. Sistemas de archivos 42](#_Toc148268629)

[9.1 Preguntas 42](#_Toc148268630)

[9.1.1 ¿Qué es un inodo en un sistema de archivos Unix o Linux? 42](#_Toc148268631)

[9.1.2 ¿Cuál es el propósito principal de la técnica de "journaling" en un sistema de archivos? 42](#_Toc148268632)

[9.1.3 ¿Cuál es la principal diferencia entre un sistema de archivos FAT32 y NTFS en Windows? 42](#_Toc148268633)

[9.1.4 ¿Cuál de los siguientes sistemas de archivos es más adecuado para sistemas de archivos grandes y aplicaciones de alto rendimiento en Linux? 42](#_Toc148268634)

[9.1.5 ¿Qué hace el comando `tee` en Linux? 42](#_Toc148268635)

[9.1.6 ¿Cuál es el propósito principal de un sistema de archivos en red (NFS) en Linux? 42](#_Toc148268636)

[9.1.7 ¿Qué es el Sistema de Archivos Virtual (VFS) en Linux? 43](#_Toc148268637)

[9.1.8 ¿Cuál es el sistema de archivos predeterminado en sistemas macOS más recientes? 43](#_Toc148268638)

[9.1.9 ¿Qué tipo de sistema de archivos se utiliza comúnmente en discos ópticos como CD-ROMs y DVDs? 43](#_Toc148268639)

[9.1.10 ¿Cuál es el propósito de un sistema de archivos comprimido? 43](#_Toc148268640)

[9.1.11 ¿Qué hace el comando df en sistemas Unix o Linux? 43](#_Toc148268641)

[9.1.12 En un sistema de archivos Unix o Linux, ¿cuál es el propósito principal del directorio /home? 43](#_Toc148268642)

[9.1.13 Algunos sistemas operativos proporcionan un cambio de nombre de llamada al sistema para darle un nuevo nombre a un archivo. ¿Existe alguna diferencia entre usar esta llamada para cambiar el nombre de un archivo y simplemente copiar el archivo a un archivo nuevo con un nuevo nombre y luego eliminar el anterior? 44](#_Toc148268643)

[9.2 Ejercicios 44](#_Toc148268644)

[9.2.1 Pruebe mostrar las particiones de disco en Linux, con el comando Fdisk con "-1" (como se indica a continuación) como usuario sudo o root. 44](#_Toc148268645)

[9.2.2 Mostrar el nombre del fichero de la terminal conectada a la salida estándar y hacer una prueba para mandarle caracteres. 45](#_Toc148268646)

[9.2.3 Ejecutar el comando w (para obtener información de los terminales activos). 45](#_Toc148268647)

[9.2.4 Probar el comando df y comentar la salida: 46](#_Toc148268648)

[9.2.5 Listar información sistema de archivos Linux 47](#_Toc148268649)

[9.2.6 Comprobar el espacio en disco en Linux usando el comando 47](#_Toc148268650)

[10 Capítulo 10. Seguridad 48](#_Toc148268651)

[10.1 Preguntas 48](#_Toc148268652)

[10.1.1 ¿Cuál de las siguientes características hace que una contraseña sea más segura? 48](#_Toc148268653)

[10.1.2 ¿Por qué es importante cambiar regularmente las contraseñas? 48](#_Toc148268654)

[10.1.3 ¿Cuál de las siguientes afirmaciones es cierta con respecto a las contraseñas seguras? 48](#_Toc148268655)

[10.1.4 ¿Cuál de las siguientes es una recomendación importante para crear contraseñas seguras? 48](#_Toc148268656)

[10.1.5 .¿Cuál es un ejemplo de una contraseña segura? 48](#_Toc148268657)

[10.1.6 ¿Por qué es importante no compartir sus contraseñas con nadie? 49](#_Toc148268658)

[10.1.7 ¿Qué es un ataque de fuerza bruta en relación con las contraseñas? 49](#_Toc148268659)

[10.1.8 ¿Cuál de las siguientes afirmaciones es cierta acerca de la autenticación de dos factores (2FA)? 49](#_Toc148268660)

[10.1.9 ¿Qué es el "salting" en el contexto de la seguridad de contraseñas? 49](#_Toc148268661)

[10.1.10 ¿Cuál es el propósito principal del "salting" en la seguridad de contraseñas? 49](#_Toc148268662)

[10.1.11 ¿Cómo afecta el "salting" a las contraseñas almacenadas? 49](#_Toc148268663)

[10.1.12 ¿Cuál de las siguientes afirmaciones es verdadera acerca de las contraseñas salteadas? 49](#_Toc148268664)

[10.1.13 ¿Cuál es una ventaja clave del uso de "salting" en la seguridad de contraseñas? 50](#_Toc148268665)

[10.1.14 ¿Qué información se utiliza típicamente como "sal" en el "salting" de contraseñas? 50](#_Toc148268666)

[10.1.15 ¿Qué es una característica importante de un "sal" efectivo? 50](#_Toc148268667)

[10.1.16 ¿Cuál de las siguientes afirmaciones es cierta con respecto al "salting" de contraseñas? 50](#_Toc148268668)

[10.1.17 .¿Qué es el método de Lamport en seguridad informática? 50](#_Toc148268669)

[10.1.18 ¿Cuál es la característica principal de las contraseñas generadas por el método de Lamport? 50](#_Toc148268670)

[10.1.19 ¿Cómo se almacenan las contraseñas en el método de Lamport? 50](#_Toc148268671)

[10.1.20 ¿Cómo se genera una contraseña de un solo uso en el método de Lamport? 51](#_Toc148268672)

[10.1.21 ¿Cuál es una desventaja potencial del método de Lamport? INvalida 51](#_Toc148268673)

[10.1.22 ¿Cómo se verifica una contraseña de un solo uso en el método de Lamport? 51](#_Toc148268674)

[10.1.23 ¿Qué hace que las contraseñas generadas por el método de Lamport sean seguras? 51](#_Toc148268675)

[10.1.24 ¿Cuál de las siguientes es un ejemplo de factor de autenticación basado en lo que se tiene? 51](#_Toc148268676)

[10.1.25 ¿Qué factor de autenticación se basa en la posesión de un objeto físico? 51](#_Toc148268677)

[10.1.26 ¿Qué tipo de factor de autenticación es más difícil de robar o duplicar? 52](#_Toc148268678)

[10.1.27 ¿Cuál es una ventaja de la autenticación basada en lo que se tiene? 52](#_Toc148268679)

[10.1.28 ¿Qué factor de autenticación se basa en conocimientos previos del usuario? 52](#_Toc148268680)

[10.1.29 ¿Cuál de las siguientes no es una característica biométrica comúnmente utilizada en sistemas de autenticación? 52](#_Toc148268681)

[10.1.30 ¿Qué hace que la biometría sea una forma efectiva de autenticación? 52](#_Toc148268682)

[10.1.31 ¿Cuál de las siguientes afirmaciones es cierta con respecto a la autenticación por huella dactilar? 52](#_Toc148268683)

[10.1.32 ¿Cuál es una consideración importante en la implementación de la autenticación por biometría? 53](#_Toc148268684)

[10.1.33 ¿Qué es un ataque de desbordamiento de búfer? 53](#_Toc148268685)

[10.2 Preguntas 2 53](#_Toc148268686)

[10.2.1 ¿Cuál de los siguientes NO es un factor de autenticación comúnmente utilizado? 53](#_Toc148268687)

[10.2.2 ¿Qué factor de autenticación se basa en algo que el usuario sabe? 53](#_Toc148268688)

[10.2.3 ¿Qué factor de autenticación se basa en algo que el usuario posee? 53](#_Toc148268689)

[10.2.4 ¿Qué factor de autenticación utiliza características físicas únicas del usuario, como su rostro o su voz? 53](#_Toc148268690)

[10.2.5 ¿Para qué se utiliza comúnmente una contraseña de BIOS o UEFI en una computadora? 53](#_Toc148268691)

[10.2.6 ¿Dónde se almacena típicamente la contraseña de BIOS o UEFI en una computadora? 54](#_Toc148268692)

[10.2.7 . ¿Cuál de las siguientes contraseñas se consideraría más débil desde el punto de vista de la seguridad? 54](#_Toc148268693)

[10.2.8 ¿Cuál de estas contraseñas es más difícil de adivinar por fuerza bruta? (k=longitud de la contraseña y n=números de símbolos utilizados) 54](#_Toc148268694)

[10.2.9 ¿Qué hace que una contraseña sea más fuerte desde el punto de vista de la seguridad? 54](#_Toc148268695)

[10.2.10 ¿Cuál de las siguientes contraseñas se consideraría más fuerte? 54](#_Toc148268696)

[10.2.11 ¿Se debe utilizar una contraseña única para diferentes cuentas en línea? 54](#_Toc148268697)

[10.2.12 ¿Qué deberías evitar al crear una contraseña? 55](#_Toc148268698)

[10.2.13 ¿Qué tipo de información se almacena comúnmente en el archivo `/etc/passwd`? 55](#_Toc148268699)

[10.2.14 ¿Dónde se encuentra generalmente el archivo `passwd` en sistemas Unix y Linux? 55](#_Toc148268700)

[10.2.15 ¿Cuál es el propósito del archivo `/etc/passwd`? 55](#_Toc148268701)

[10.2.16 ¿Cuál de las siguientes afirmaciones es cierta acerca del archivo `/etc/shadow`? 55](#_Toc148268702)

[10.2.17 ¿Qué es el hashing de contraseñas? 55](#_Toc148268703)

[10.2.18 ¿Cuál es el propósito principal de utilizar el hashing de contraseñas? 55](#_Toc148268704)

[10.2.19 ¿Cuál es la ventaja de usar un "salt" (sal) en el hashing de contraseñas? 56](#_Toc148268705)

[10.2.20 ¿Qué es el salting en el contexto de la seguridad de contraseñas? 56](#_Toc148268706)

[10.2.21 . ¿Cuál es el propósito principal de agregar sal a las contraseñas antes de hashearlas? 56](#_Toc148268707)

[10.2.22 ¿Cuál de las siguientes afirmaciones es cierta acerca del salting? 56](#_Toc148268708)

[10.2.23 ¿Cuál es una ventaja del salting en la seguridad de contraseñas? 56](#_Toc148268709)

[10.2.24 ¿Qué tipo de sistema de autenticación es el método de Lamport? 56](#_Toc148268710)

[10.2.25 ¿Cómo se genera una contraseña Lamport? 56](#_Toc148268711)

[10.2.26 Una técnica común de ingeniería social online en la que los atacantes envían correos electrónicos o mensajes que parecen provenir de fuentes legítimas, pero las víctimas son dirigidas a estos sitios falsos, donde se les solicita que ingresen sus contraseñas y otra información confidencial. ¿De qué tipo de ataque estamos hablando? 57](#_Toc148268712)

[10.2.27 A la acción de la figura se le suele llamar: 57](#_Toc148268713)

[10.2.28 ¿Qué es un ataque de desbordamiento de búfer? 57](#_Toc148268714)

[10.2.29 ¿Cuál es el objetivo principal de un ataque de desbordamiento de búfer? 57](#_Toc148268715)

[10.2.30 ¿Qué tipo de vulnerabilidad de software es explotada típicamente en un ataque de desbordamiento de búfer? 57](#_Toc148268716)

[10.2.31 ¿Cuál es un objetivo común de un ataque exitoso de desbordamiento de búfer? 57](#_Toc148268717)

[10.2.32 ¿Qué es un "Canario de Pila" en el contexto de la seguridad informática? 58](#_Toc148268718)

[10.2.33 ¿Cuál es el propósito principal de un Canario de Pila? 58](#_Toc148268719)

[10.2.34 ¿Cómo funciona un Canario de Pila? 58](#_Toc148268720)

[10.2.35 . ¿Cuál es uno de los problemas a resolver en la implementación de Canarios de Pila? 58](#_Toc148268721)

[10.2.36 ¿En qué tipo de programas y sistemas operativos es común encontrar la implementación de Canarios de Pila? 58](#_Toc148268722)

[10.2.37 ¿Qué sucede si un atacante intenta desbordar un búfer en un programa que utiliza un Canario de Pila? 58](#_Toc148268723)

[10.2.38 ¿Cuál es uno de los beneficios clave de los Canarios de Pila en términos de seguridad? 59](#_Toc148268724)

[10.2.39 . ¿Qué significa la sigla DEP en el contexto de seguridad informática? 59](#_Toc148268725)

[10.2.40 ¿Cuál es el objetivo principal de DEP? 59](#_Toc148268726)

[10.2.41 ¿Cómo funciona DEP para prevenir la ejecución de código malicioso? 59](#_Toc148268727)

[10.2.42 ¿En qué tipo de sistemas operativos es común encontrar la función DEP? 59](#_Toc148268728)

[10.2.43 ¿Cuál de las siguientes afirmaciones es cierta sobre DEP? 59](#_Toc148268729)

[10.2.44 . ¿Cuál es uno de los beneficios clave de DEP en términos de seguridad? 60](#_Toc148268730)

[10.2.45 ¿Qué significa la sigla ROP en el contexto de seguridad informática? 60](#_Toc148268731)

[10.2.46 ¿Cuál es el objetivo principal de un ataque de ROP? 60](#_Toc148268732)

[10.2.47 ¿Qué hace que ROP sea una técnica de ataque efectiva? 60](#_Toc148268733)

[10.2.48 . ¿Cómo funciona un ataque de ROP? 60](#_Toc148268734)

[10.2.49 ¿Cuál es uno de los desafíos clave para un atacante en un ataque de ROP? 60](#_Toc148268735)

[10.2.50 . ¿En qué tipo de programas y sistemas operativos es más común encontrar ataques de ROP? 61](#_Toc148268736)

[10.2.51 ¿Qué medida de seguridad puede ayudar a mitigar los ataques de ROP? 61](#_Toc148268737)

[10.2.52 . ¿Cuál es el propósito principal de ASLR? 61](#_Toc148268738)

[10.2.53 ¿Cómo funciona ASLR en un sistema operativo? 61](#_Toc148268739)

[10.2.54 ¿Cuál es uno de los beneficios clave de ASLR en términos de seguridad? 61](#_Toc148268740)

[10.2.55 ¿En qué tipo de sistemas operativos se encuentra comúnmente la implementación de ASLR? 61](#_Toc148268741)

[10.2.56 ¿Qué tipo de ataque de seguridad puede dificultarse significativamente mediante la implementación de ASLR? 62](#_Toc148268742)

[10.2.57 ¿Cuál es uno de los posibles desafíos al implementar ASLR? 62](#_Toc148268743)

[10.2.58 . ¿Qué es un ataque de detección de canal lateral? 62](#_Toc148268744)

[10.2.59 ¿Cuál es el objetivo principal de un ataque de detección de canal lateral? 62](#_Toc148268745)

[10.2.60 ¿Qué tipo de información se puede obtener a través de un ataque de detección de canal lateral? 62](#_Toc148268746)

[10.2.61 ¿Cuál de las siguientes afirmaciones es cierta sobre la mitigación de ataques de detección de canal lateral? 62](#_Toc148268747)

[10.2.62 . ¿Cuál es uno de los desafíos de la mitigación de ataques de detección de canal lateral? 63](#_Toc148268748)

[10.2.63 ¿Qué es un ataque de detección de canal lateral en el contexto del algoritmo RSA? 63](#_Toc148268749)

[10.2.64 ¿Cuál es el objetivo principal de un ataque de detección de canal lateral en el contexto de RSA? 63](#_Toc148268750)

[10.2.65 ¿Qué tipo de información se puede obtener a través de un ataque de detección de canal lateral en RSA? 63](#_Toc148268751)

[10.2.66 . ¿Cuál es uno de los canales secundarios utilizados en un ataque de detección de canal lateral en RSA? 63](#_Toc148268752)

[10.2.67 ¿Cuál de las siguientes afirmaciones es cierta sobre la mitigación de ataques de detección de canal lateral en RSA? 63](#_Toc148268753)

[10.2.68 ¿Cuál es uno de los desafíos de la mitigación de ataques de detección de canal lateral en RSA? 64](#_Toc148268754)

[10.2.69 ¿Qué son los ataques de canal lateral a la caché? 64](#_Toc148268755)

[10.2.70 2. ¿Qué tipo de información se puede obtener a través de un ataque de canal lateral a la caché? 64](#_Toc148268756)

[10.2.71 3. ¿Cuál es uno de los canales secundarios utilizados en ataques de canal lateral a la caché? 64](#_Toc148268757)

[10.2.72 . ¿Qué es la "sincronización del reloj" en el contexto de un ataque de canal lateral a la caché? 64](#_Toc148268758)

[10.2.73 . ¿Cuál es uno de los ejemplos más conocidos de ataque de canal lateral a la caché? 64](#_Toc148268759)

[10.2.74 . ¿Cómo pueden mitigarse los ataques de canal lateral a la caché? 64](#_Toc148268760)

[10.2.75 .¿Cuál es uno de los desafíos de la mitigación de ataques de canal lateral a la caché? 65](#_Toc148268761)

[10.2.76 ¿En qué tipo de entornos o escenarios es más probable que ocurran ataques de canal lateral a la caché? 65](#_Toc148268762)

[10.2.77 ¿Qué es la "vulnerabilidad Meltdown" en relación con los ataques de canal lateral a la caché? 65](#_Toc148268763)

[10.2.78 ¿Qué es una "cola de almacenamiento" en una CPU y cómo puede estar relacionada con los ataques de canal lateral a la caché? 65](#_Toc148268764)

[10.2.79 ¿Cuál es uno de los métodos utilizados en la mitigación de ataques de canal lateral a la caché? 65](#_Toc148268765)

[10.2.80 . ¿Cuál es el objetivo principal de un ataque de canal lateral a la caché? 65](#_Toc148268766)

[10.2.81 ¿Cuál es una de las ventajas de la caché de la CPU que también puede ser aprovechada por los atacantes en ataques de canal lateral? 66](#_Toc148268767)

[10.2.82 ¿Cuál es uno de los efectos secundarios no deseados de las medidas de mitigación de ataques de canal lateral a la caché? 66](#_Toc148268768)

[10.2.83 ¿Cuál es una técnica comúnmente utilizada en ataques de canal lateral a la caché para deducir información confidencial? 66](#_Toc148268769)

[10.2.84 ¿Qué es el ataque de canal lateral al caché conocido como "Prime+Probe"? 66](#_Toc148268770)

[10.2.85 ¿Cuál es la fase "Prime" en el ataque Prime+Probe? 66](#_Toc148268771)

[10.2.86 ¿Qué papel desempeña la fase "Probe" en el ataque Prime+Probe? 66](#_Toc148268772)

[10.2.87 . ¿Cuál es el objetivo principal del ataque Prime+Probe? 67](#_Toc148268773)

[10.2.88 ¿Cuál de las siguientes afirmaciones es cierta sobre los ataques de canal lateral a la caché en general? 67](#_Toc148268774)

[10.2.89 ¿Cuál es uno de los desafíos de mitigar el ataque Prime+Probe? 67](#_Toc148268775)

[10.2.90 ¿Qué es la CPU especulativa? 67](#_Toc148268776)

[10.2.91 . ¿Cuál es el propósito principal de la ejecución especulativa en una CPU? 67](#_Toc148268777)

[10.2.92 ¿Qué es un "branch prediction" en una CPU especulativa? 67](#_Toc148268778)

[10.2.93 ¿Cuál es uno de los posibles desafíos de la CPU especulativa relacionado con la seguridad? 67](#_Toc148268779)

[10.2.94 .¿En qué tipo de aplicaciones o escenarios es más beneficioso el uso de CPU especulativa? 68](#_Toc148268780)

[10.2.95 . ¿Cómo afecta la ejecución especulativa al rendimiento general de la CPU? 68](#_Toc148268781)

[10.2.96 ¿Cuál es un beneficio clave de la CPU especulativa en términos de rendimiento? 68](#_Toc148268782)

[10.3 Preguntas adicionales sobre CPU especulativa: 68](#_Toc148268783)

[10.3.1 ¿Cuál es uno de los principales riesgos de seguridad asociados con la CPU especulativa? 68](#_Toc148268784)

[10.3.2 . ¿Cuál es el propósito de la "ejecución fuera de orden" en una CPU especulativa? 68](#_Toc148268785)

[10.3.3 . ¿Cuál es una de las ventajas de la CPU especulativa en términos de rendimiento de aplicaciones de cómputo intensivo? 68](#_Toc148268786)

[10.3.4 . ¿Cuál es una de las desventajas de la ejecución especulativa en términos de consumo de energía? 69](#_Toc148268787)

[10.3.5 . ¿Qué tipo de instrucciones en un programa de computadora pueden beneficiarse más de la ejecución especulativa? 69](#_Toc148268788)

[10.3.6 . ¿Cómo puede mitigarse el riesgo de vulnerabilidades de seguridad asociadas con la CPU especulativa? 69](#_Toc148268789)

# Capítulo 1. Introducción al Sistema Operativo

## Preguntas

### Explicar el propósito principal de un sistema operativo

Los sistemas operativos existen para dos propósitos principales. Una es que está diseñado para asegurarse de que un sistema informático funcione bien mediante la gestión de sus actividades informáticas. Otra es que proporciona un entorno para el desarrollo y ejecución de programas.

El propósito principal de un sistema operativo es permitir al usuario entenderse con el Hardware, llevando a cabo la administración de manera eficiente del mismo y proporcionando al usuario herramientas para los programas de aplicación.

### Características de la vista de usuario del SO. ¿Qué herramientas utiliza el usuario en esta vista para enviar órdenes al sistema operativo? Algunas computadoras tienen poca o ninguna vista de usuario. Pon ejemplos

Características de la vista de usuario del SO:

* Desde una vista usuario se dispone del modo usuario para manejar un subconjunto de instrucciones del SO.
* La vista del usuario de la computadora varía según la interfaz que se utilice.
* Actúan de forma ergonómica, directa y sencilla para sacar el mejor partido de los recursos.
* Es el programa de interfaz de usuario, shell o GUI, entendido como el nivel más bajo de software en modo de usuario que permite al usuario iniciar otros programas.

Herramientas para enviar ordenas al SO: navegador web, un lector de correo electrónico o un reproductor de música. Estos programas también hacen un uso intensivo del sistema operativo.

Computadores con poco o ninguna vista de usuario: Microcontroladores empotrados en domótica y automóvil.

### Al parecer no hay una estricta separación entre los programas kernel y los programas de aplicación. Exponga cuatro tipos de programas que podíamos incluir en el SO

Programas de ayuda, programas de administración, programas de entorno gráfico de gestión, y programas intérpretes de comandos.

### Comentar las peculiaridades de los SO en general en cuanto a su dimensionamiento en líneas de código, su prolongado ciclo de vida y su complejidad

Un SO implica un desarrollo de gran complejidad. Esto supone numerosas líneas de código, así como numerosos recursos materiales y humanos, que también se traducen en términos económicos. Por ello, su elaboración desde cero supone un despliegue de tiempo, presupuesto, medios y personas que no muchas empresas están dispuestas a asumir. Es por ello que un SO se mantiene modificando e implementando funcionalidades a medida que aparecen nuevas tecnologías para poder utilizar periféricos y protocolos de comunicación entre otros. Por ello, se concluye que, aunque estos sistemas operativos tengan en un momento dado un cambio en la arquitectura como el caso de Windows NT, el kernel a penas cambia con el paso del tiempo en muchos de los SSOO que conocemos a día de hoy.

### Modelo conceptual del sistema operativo en una perspectiva de arriba hacia abajo. Ídem de abajo hacia arriba.

El concepto de que un sistema operativo proporciona principalmente abstracciones a los programas de aplicación es una **visión de arriba hacia abajo**.

Una visión alternativa, **de abajo hacia arriba**, sostiene que el sistema operativo está ahí para administrar todas las piezas de un sistema complejo.

### Aunque hay muchos profesionales de las ciencias de la computación, solo un pequeño porcentaje de ellos estará involucrado en la creación o modificación de un sistema operativo. Indique cuatro razones para estudiar la teoría de los sistemas operativos en este curso.

Crucial para programar de manera robusta eficiente y segura.

Fortalecer la seguridad del propio SO, dificultando accesos no autorizados, ya que el SO instalado de serie ofrece vulnerabilidades.

Es una herramienta fundamental en el fortalecimiento de la seguridad de los servidores.

Sacar máximo rendimiento de los recursos de bajo nivel.

### En 1968 Ken Thompson con una pequeña minicomputadora PDP-7 abandonada se propuso escribir una versión simplificada de MULTICS para un solo usuario. ¿Por qué tenía la capacidad de desarrollar un SO? ¿Qué importancia tuvo este hecho en la historia de los SO?

Ken Thomson trabajaba en los laboratorios Bell, desarrollando el SO MULTICS. Se considera el padre del lenguaje B en el que estaba escrito dicho SO y continuó desarrollando este sistema hasta que se abandonó debido a su creciente complejidad. En 1969, por su cuenta y junto a Dennis Ritchie trató de convertir el SO MULTICS abandonado en un sistema monousuario más simple y que tuviera mucho más rendimiento.

Este sistema se acabó transformado formalmente en UNIX en 1970 cuando consiguieron el apoyo económico de los laboratorios Bell. En 1972 decidieron reescribir completamente el código usando lenguaje C haciendo que se pudiera modificar fácilmente para trabajar en múltiples computadoras y convirtiéndose en el padre de una gran familia de sistemas operativos multisistema y multiusuario como System V, XENIX, BSD y LINUX, entre otros.

### Enunciado falso: “1960 Doug Engelbart en el Instituto de Investigación de Stanford inventó la interfaz gráfica de usuario, completa con ventanas, íconos, menús y ratón, encargado por Apple. Pero Steve Jobs, quien coinventó la computadora de Xerox PARC compró la interface gráfica de Engelbart para sus máquinas. En 1999, Apple adoptó un núcleo derivado de BSD UNIX que reemplazó al micronúcleo Mach de la Universidad Carnegie Mellon. Por eso el macOS de Apple no es un sistema operativo basado en UNIX”.

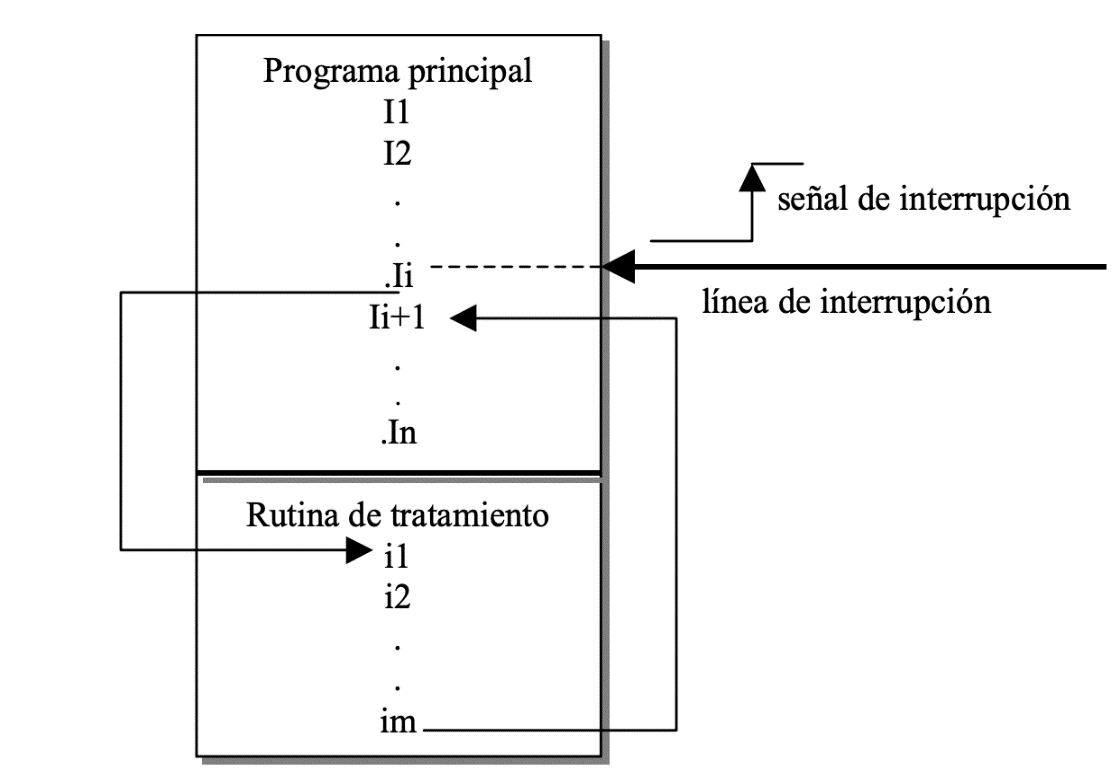
## Rectifique adecuadamente ese falso enunciado

“1960 Doug Engelbart en el Instituto de Investigación de Stanford inventó la interfaz gráfica de usuario, completa con ventanas, íconos, menús y ratón, encargado por Apple (NO FUE ENCARGADA POR APPLE, FUE POR XEROX PARCK). Pero Steve Jobs, quien coinventó la computadora de Xerox PARC (no coinventó la computadora de XEROC PARC) compró (NO LO COMPRÓ, CONSTRUYÓ UNO PROPIO) la interface gráfica de Engelbart para sus máquinas. En 1999, Apple adoptó un núcleo derivado de BSD UNIX que reemplazó al micronúcleo Mach de la Universidad Carnegie Mellon. Por eso el macOS de Apple no es (SI QUE ES) un sistema operativo basado en UNIX”.

### Explicar las diferencias claras entre driver de dispositivo y controlador de dispositivo.

Un controlador de dispositivo se encuentra en el propio dispositivo, es una parte de él, y tiene una parte de hardware y de programa (software). El driver de dispositivo, por el contrario, es un programa que sirve para comunicar ese dispositivo con el Sistema Operativo y que se pueda ejecutar.

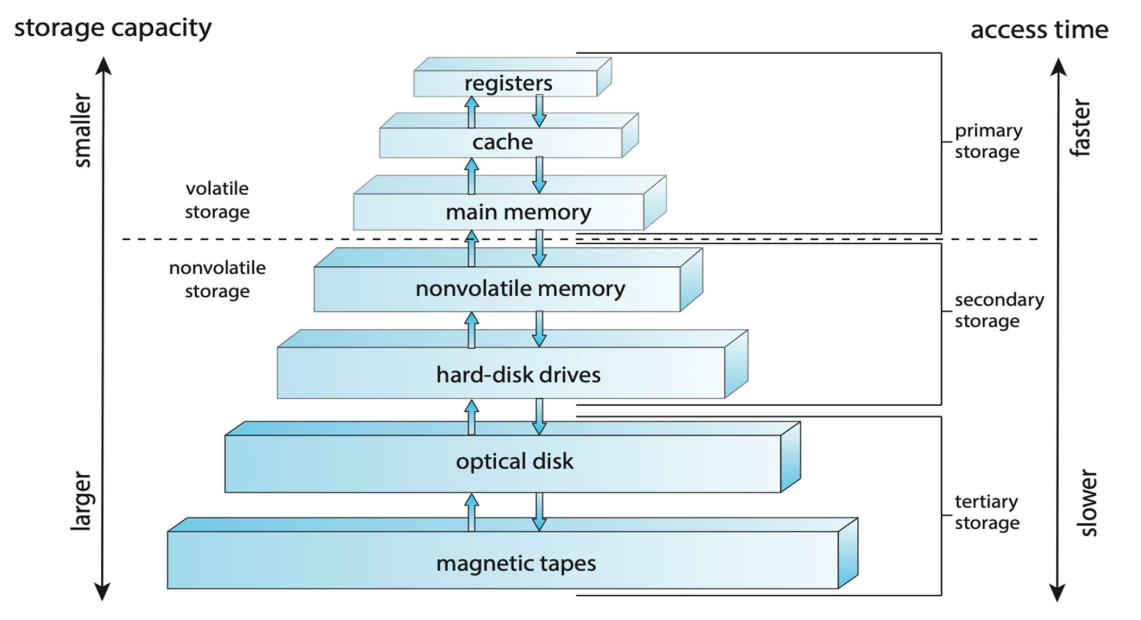
### Mediante el esquema que se acompaña, explicar el concepto IRQ en el SO

Una interrupción es una suspensión temporal de la ejecución de un proceso, para pasar a ejecutar una subrutina de servicio de interrupción, que, por lo general, no forma parte del programa, sino que pertenece al sistema operativo. Una vez finalizada dicha subrutina, se reanuda la ejecución del programa. Las interrupciones son generadas por los dispositivos periféricos habilitando una señal del CPU (llamada IRQ "interrupt request") para solicitar atención del mismo. Las interrupciones son una parte importante de la arquitectura de una computadora.

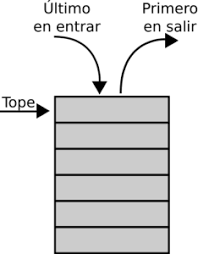
### Diferencia entre trampa (TRAP) e interrupción utilizando la siguiente tabla:

|  |  |  |
| --- | --- | --- |
|  | **Trampa** | **Interrupción** |
| **Método de Generación** | surge como evento durante la ejecución de una instrucción en el programa | generada como un evento por dispositivos de hardware |
| **Funcionalidad principal** | invoca la funcionalidad del SO quien transfiere su control al manipulador de trampas | activa el procesador, quien transfiere el evento al controlador de interrupciones |
| **Ocurrencia** | síncrona, tras la ejecución de una instrucción | asíncrona, no depende de la ejecución de instrucciones, al producirse por hardware no está directamente relacionada con una secuencia de instrucciones |
| **Sinónimos** | interrupción de software | interrupción de hardware |

### Respecto a la estructura de almacenamiento en niveles, sustituya las manchas rojas por los conceptos: rápido, lento, pequeño, grande, memoria volátil, memoria no volátil, almacenamiento primario, secundario y terciario

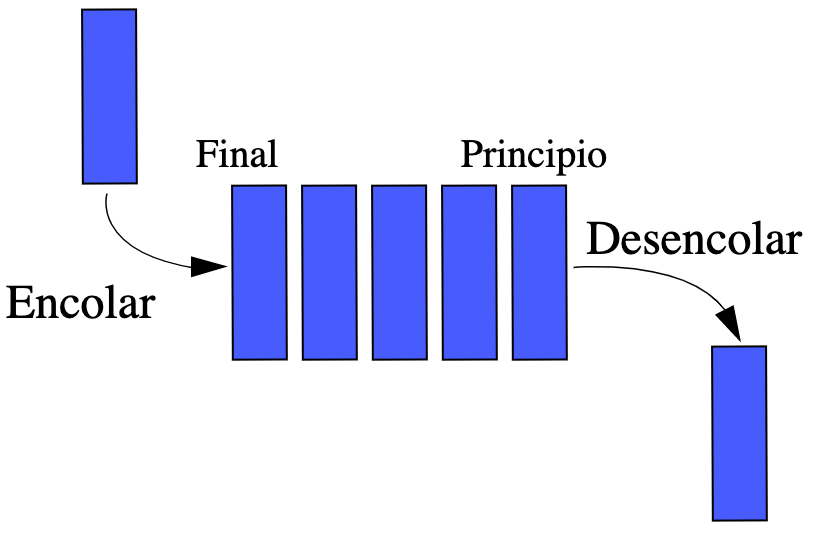


### Qué estructura de datos es la de la figura y ponga ejemplos donde el SO la emplea



Esta estructura de datos se trata de una pila ordenada secuencialmente y que utiliza el principio de último en entrar, primero en salir (LIFO). Un sistema operativo a menudo usa una pila cuando invoca llamadas a funciones. Los parámetros, las variables locales y la dirección de retorno se colocan en la pila cuando se llama a una función; regresar de la llamada a la función saca esos elementos de la pila.

### Qué estructura de datos es la de la figura y ponga ejemplos donde el SO la emplea



Es una estructura de datos ordenada en cola que utiliza el principio de primero en entrar, primero en salir (FIFO): los elementos se eliminan de una cola en el orden en que se insertaron.

### Un árbol rojo-negro es un árbol binario de búsqueda en el que cada nodo tiene un atributo de color cuyo valor es rojo o negro. Además de los requisitos impuestos a los árboles binarios de búsqueda convencionales, enuncie cuatro reglas para tener un árbol rojo-negro válido. Linux utiliza un árbol rojo-negro (rbtree) en muchos lugares del kernel, indique tres:

Además de los requisitos impuestos a los árboles binarios de búsqueda convencionales, se deben satisfacer las siguientes reglas para tener un árbol rojo-negro válido:

* Todo nodo es o bien rojo o bien negro.
* La raíz es negra.
* Todas las hojas (NULL) son negras.
* Todo nodo rojo debe tener dos nodos hijos negros.

Linux utiliza un árbol rojo-negro (rbtree) en muchos lugares del kernel, por ejemplo:

* El programador de E/S utilizan rbtree para realizar un seguimiento de las solicitudes.
* En el control de las unidades de datos en bloque de CD / DVD.
* El código del temporizador de alta resolución usa rbtree para organizar las solicitudes del temporizador.

# Capítulo 2: Llamadas y tipos de sistemas operativos.

## Preguntas

### ¿Cuál es el propósito del intérprete de comandos? ¿Por qué suele estar separado del núcleo?

La función principal del intérprete de comandos es obtener y ejecutar el comando especificado por el usuario. Suele estar separado del núcleo porque la mayoría de los sistemas operativos, incluidos Linux, UNIX y Windows, tratan al intérprete de comandos como un programa especial que se ejecuta cuando se inicia un proceso o cuando un usuario inicia sesión por primera vez (en sistemas interactivos), existiendo sistemas operativos con múltiples intérpretes de comandos para elegir, que se conocen como shells.

### Ventajas de CLI

* Tiene un rendimiento más rápido: Si se sabe cómo usar CLI y se está familiarizado con diferentes comandos, el ciclo pregunta-respuesta se completará más rápido en comparación con el uso de GUI.
* Consume menos memoria: CLI usa menos memoria en comparación con la GUI.
* Se necesita un monitor de baja resolución para su uso a diferencia del GUI.
* Un procesador lento puede funcionar: ya que no necesitan potencia de procesamiento adicional.
* No necesita instalar complejos programas para gestionar entornos gráficos. Todos los SO importantes admiten CLI.
* Autocompletar: Cuando escribe comandos en la CLI, la mayoría de los comandos se completan automáticamente al hacer clic en el botón TAB en el teclado.
* Funciones avanzadas: En CLI se tiene posibilidades de acceso a más funciones avanzadas que en GUI.
* Trabajo remoto: CLI es una opción ágil para comunicación entre dispositivos y operar diferentes comandos CLI en computadoras remotas.
* Historial de comandos: Al ejecutar comandos en CLI, las órdenes se guardan en un recurso de la memoria. Al usar las flechas arriba/abajo del teclado se puede volver a usar esos comandos.

### Ventajas de GUI

Una interfaz gráfica de usuario o GUI es fácil de usar. En lugar de ingresar textos que expresan comandos directamente a través de una CLI, los usuarios emplean un sistema de ventanas y menús basado en ratón caracterizado por una metáfora de escritorio. El usuario mueve el dispositivo ratón para colocar su puntero sobre imágenes o iconos en la pantalla (el escritorio) que representan programas, archivos, directorios y funciones del sistema gráfico. Dependiendo de la ubicación del puntero, hacer clic en un botón del ratón puede invocar un programa, seleccionar un archivo o directorio, conocido como carpeta, o desplegar un menú que contiene comandos.

### ¿Cuál es el propósito de las llamadas al sistema?

Una llamada al sistema es una rutina que permite a una aplicación de usuario solicitar acciones que requieren privilegios especiales. La adición de llamadas al sistema es una de varias maneras de ampliar las funciones proporcionadas por el kernel.

### Diferencias entre la llamada a una función en el dominio de aplicaciones de usuario y una llamada al sistema.

Una llamada al sistema difiere de una función de usuario de varias formas:

* Una llamada al sistema tiene más privilegios que una subrutina normal.
* Una llamada al sistema se ejecuta con el privilegio modo de kernel en el dominio de protección de kernel.
* El código de llamada del sistema y los datos se encuentran en la memoria global del kernel.
* Las rutinas de llamada al sistema pueden crear y utilizar procesos de kernel para realizar el proceso asíncrono.
* Las llamadas del sistema no pueden utilizar bibliotecas compartidas ni ningún símbolo que no se encuentre en el dominio de protección del kernel.

### Indique los seis tipos de llamadas al sistema que generalmente tienen los SO

En general las llamadas al sistema se pueden agrupar aproximadamente en seis categorías principales: control de procesos, gestión de archivos, gestión de dispositivos, mantenimiento de la información, comunicaciones y protección.

### ¿Que es POSIX?

POSIX ([acrónimo](https://es.wikipedia.org/wiki/Acrónimo) de Portable Operating System Interface, y X viene de [UNIX](https://es.wikipedia.org/wiki/UNIX) como seña de identidad de la [API](https://es.wikipedia.org/wiki/Application_Programming_Interface)) es una norma escrita por la [IEEE](https://es.wikipedia.org/wiki/IEEE), que define una interfaz estándar del [sistema operativo](https://es.wikipedia.org/wiki/Sistema_operativo) y el entorno, incluyendo un [intérprete de comandos](https://es.wikipedia.org/wiki/Intérprete_de_comandos) (o "shell").

El término fue sugerido por [Richard Stallman](https://es.wikipedia.org/wiki/Richard_Stallman) en la década de 1980, en respuesta a la demanda del [IEEE](https://es.wikipedia.org/wiki/IEEE), que buscaba un nombre fácil de recordar. La traducción del acrónimo es "Interfaz de Sistema Operativo Portable".

### ¿Cuáles son los tipos de SO según estructura de kernel? De una explicación de una línea de cada uno de ellos

* **Sistemas operativos de kernel monolítico/monolítico con extensiones cargables**: Los sistemas operativos monolíticos se caracterizan por implementar en el núcleo los cuatro componentes fundamentales del sistema operativo.
* **Sistemas operativos de microkernel:** Se caracterizan por disponer de un núcleo que implementa únicamente Planificación de proceso, Mecanismo de comunicación entre procesos y la Gestión de interrupciones. Además, existen procesos servidores que, cuando un proceso cualquiera solicita un servicio a través de una llamada al sistema, el micronúcleo canaliza la petición al proceso servidor correspondiente.
* **Sistemas operativos de kernel híbrido**: El kernel de este tipo de Sistema Operativo el núcleo en cuestión usa conceptos de arquitectura tanto del diseño monolítico como del micronúcleo.

### Comparación de un kernel monolítico de un microkernel

Los sistemas operativos monolíticos se caracterizan por implementar en el núcleo los cuatro componentes fundamentales del sistema operativo, que son la planificación de procesos, la administración de la memoria principal, la administración de ficheros y la gestión de los dispositivos de entrada/salida. Algunos de ellos admiten la inclusión de módulos compilados en tiempo de ejecución del kernel del sistema operativo sin necesidad de recompilar completamente el mismo.

Por otra parte, los microkernel se caracterizan por disponer de un núcleo que implementa únicamente las funciones básicas de Planificación de proceso, Mecanismo de comunicación entre procesos y la Gestión de interrupciones.

### ¿Qué tipo de núcleo llevan los siguientes SO?

|  |  |
| --- | --- |
| Ubuntu | Monolítico con extensiones. |
| Windows 10 | Híbrido. |
| MacOS | Monolítico con extensiones. |
| MacOSX | Híbrido. |
| Android | Monolítico con extensiones. |
| IOS | Híbrido. |
|  |  |

### Utilizando la tabla a continuación, establecer las diferencias entre el kernel de un S.O. monolítico con módulos recargable y un S.O. híbrido

|  |  |  |
| --- | --- | --- |
|  | **Monolítico con Módulos Recargables** | **Híbrido** |
| Arquitectura | monolitica | microkernel |
| Espacio del S.O. en modo usuario | poco | mucho |
| Espacio del S.O. en modo kernel | mucho | poco |
| Modularidad | poca | mucha |
| Modo de operación de los módulos | Kernel | usuario |
| Riesgo | Bajo | alto |
| Coste de diseño | Teoría : alto | Teoria: bajo |
| Coste o facilidad de mantenimiento | alto | bajo |
| Consumo de recursos computacionales | bajo | alto |
| Ejemplos | Linux | Windows 10 |

# Capítulo 3. Procesos

## Preguntas

### ¿Qué es un proceso del sistema operativo? ¿Qué recursos del sistema necesita un proceso para realizar satisfactoriamente su tarea?

El proceso es un programa en ejecución. El estado de la actividad actual de un proceso está representado por el valor del contador del programa y el contenido de los registros del procesador. Por último, los procesos necesitan recursos para ejecutarse como son:

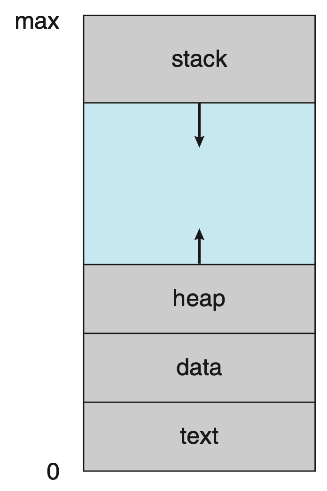
* Tiempo de CPU.
* Memoria.
* Archivos.
* Dispositivos E/S.

### ¿Cuales son las obligaciones del S.O. como gestor de procesos?

Las obligaciones del SO como gestor de procesos son:

* Creación y eliminación de procesos
* Planificación de procesos
* Establecimiento de mecanismos para la sincronización y comunicación de procesos.
* Manejo de bloqueos mutuos

### ¿Qué suele almacenarse en cada una de las secciones de la memoria ocupada por un proceso?

El diseño de la memoria que un proceso necesita generalmente se divide en varias secciones:

* Sección de texto: el código ejecutable
* Sección de datos: variables globales
* Sección Heap: memoria que se asigna dinámicamente durante el tiempo de ejecución del programa
* Sección stack (de pila): almacenamiento temporal de datos al invocar funciones (como parámetros de función, direcciones de retorno y variables locales.

### Se ha ejecutado el comando “size /bin/more”. Explicar la salida obtenida

[root@localhost home]# size /bin/more

   text    data     bss     dec     hex filename

  29527    1824      24   31375    7a8f /bin/more

### Respecto a los estados de un proceso, rellene adecuadamente la tabla

| **Estado** | **Descripción** |
| --- | --- |
| Nuevo | El proceso se está creando. |
| Preparado | El proceso está esperando a ser asignado a un procesador. |
| Activo | Se están ejecutando instrucciones. |
| Bloqueado | El proceso está esperando que ocurra algún evento (como una finalización de E/S o la recepción de una señal). |
| Terminado | El proceso ha terminado la ejecución. |

### ¿Qué es un PCB? ¿Qué información generalmente se suele guardar en un PCB?

Es una estructura de datos utilizada por el SO para administrar cada proceso. Contiene muchos campos de información asociados a un proceso específico como:

* Estado del proceso. El estado puede ser nuevo, listo, en ejecución, en espera, detenido, etc.
* Contador de programa indica la dirección de la siguiente instrucción a ejecutar para este proceso
* Registros de la CPU. Los registros varían en número y tipo, dependiendo de la arquitectura de la computadora. Incluyen acumuladores, registros de índice, punteros de pila y registros de propósito general, además de cualquier información que condicione la ejecución del proceso. Junto con el contador del programa, esta información de estado debe guardarse cuando ocurre una interrupción, para permitir que el proceso continúe correctamente cuando se reanude la ejecución.
* Información de planificación (scheduling) de CPU: prioridad de proceso, punteros a colas de scheduling, etc.
* Información de gestión de memoria: valor de los registros base y límite, y las tablas de páginas o de segmentos, según el sistema de memoria utilizado por el sistema operativo.
* Información de contabilidad de CPU: la cantidad de CPU y el tiempo real utilizado, los límites de tiempo, la identificación de tareas o de proceso, etc.
* Información de estado de E/S: la lista de dispositivos de E/S asignados al proceso, una lista de archivos abiertos, etc.

## Ejercicios

### Comando PS:

Listará (de múltiples formas según las opciones que le pasemos) todos los procesos que se encuentran corriendo en nuestro equipo:

xavier@ubuntuserver:~$ ps

PID TTY TIME CMD

964 pts/0 00:00:00 bash

969 pts/0 00:00:00 ps

### Comando PS AUX:

Listara todos los procesos del sistema, incluyendo los iniciados por el propio sistema.

xavier@ubuntuserver:~$ ps aux

USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND

root 1 0.0 0.5 100308 11380 ? Ss Oct14 0:01 /sbin/init

root 2 0.0 0.0 0 0 ? S Oct14 0:00 [kthreadd]

root 3 0.0 0.0 0 0 ? I< Oct14 0:00 [rcu\_gp]

root 4 0.0 0.0 0 0 ? I< Oct14 0:00 [rcu\_par\_gp]

[…

root 1256 0.0 0.0 0 0 ? I 11:47 0:00 [kworker/u2:0

xavier 1259 0.0 0.1 7480 3264 pts/0 R+ 11:48 0:00 ps aux]

### Comando top

xavier@ubuntuserver:~$ top

top - 11:52:24 up 12:13, 1 user, load average: 0.00, 0.00, 0.00

Tasks: 97 total, 1 running, 96 sleeping, 0 stopped, 0 zombie

%Cpu(s): 0.3 us, 0.7 sy, 0.0 ni, 98.7 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st

MiB Mem : 1964.1 total, 1374.5 free, 176.1 used, 413.5 buff/cache

MiB Swap: 2048.0 total, 2048.0 free, 0.0 used. 1637.8 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND

963 xavier 20 0 17172 8088 5692 S 0.7 0.4 0:00.10 sshd

25 root 20 0 0 0 0 S 0.3 0.0 0:01.81 kcompactd0

1260 xavier 20 0 7832 3712 3056 R 0.3 0.2 0:00.15 top

1 root 20 0 100308 11380 8384 S 0.0 0.6 0:01.69 systemd

2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd

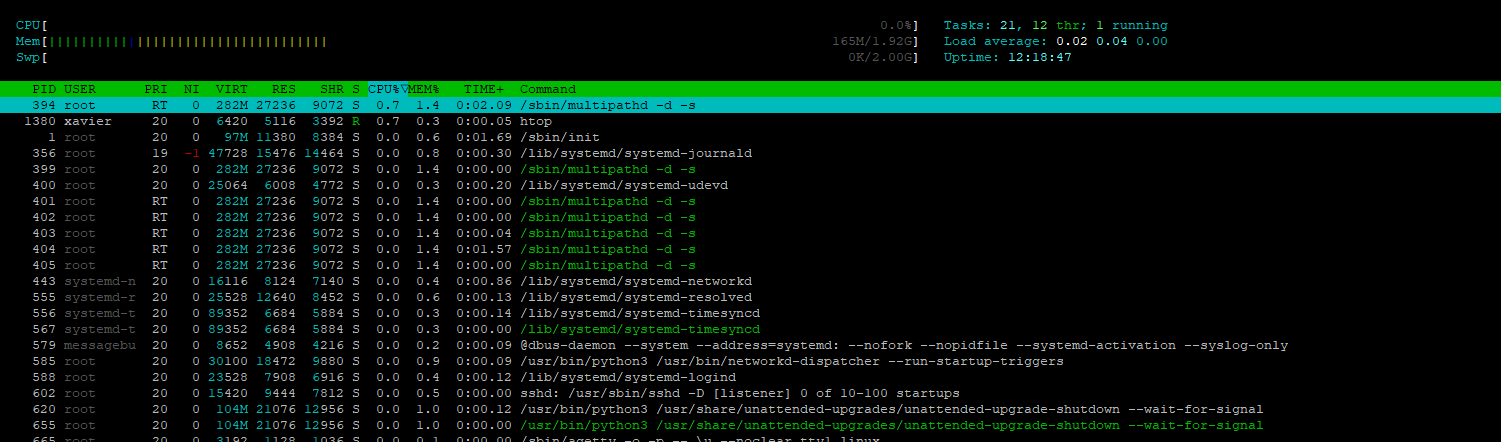
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu\_gp

[...]

El comando Top ofrece un conjunto de estadísticas sobre el estado general del sistema, así como el "top" de los procesos más activos en el sistema. La lista se presenta y actualiza por defecto cada 2 segundos.

* Los datos que presenta TOP son:
* Utilización de CPU (%);
* información sobre el “top” de los procesos;
* Uso de memoria RAM;
* Número de usuarios registrados en el sistema;
* Información de la memoria SWAP;
* Número de tareas;
* PID de tareas (número que se refiere a tareas).

### Comando HTOP

Otro gestor de procesos parecido a top es ‘ htop ’, El comando HTOP es una versión mejorada del comando TOP. La información es similar a la mostrada por el comando TOP, pero presentada en una interfaz más intuitiva.

### Comando pstree

xavier@ubuntuserver:~$ pstree

systemd─┬─agetty

├─dbus-daemon

├─multipathd───6\*[{multipathd}]

├─networkd-dispat

├─packagekitd───2\*[{packagekitd}]

├─polkitd───2\*[{polkitd}]

├─sshd───sshd───sshd───bash───pstree

├─systemd───(sd-pam)

├─systemd-journal

├─systemd-logind

├─systemd-network

├─systemd-resolve

├─systemd-timesyn───{systemd-timesyn}

├─systemd-udevd

└─unattended-upgr───{unattended-upgr}

El comando ***pstree***, nos permitirá ver el **árbol de procesos activos**, en nuestros sistemas GNU Linux.

Difiere del popular ‘***ps***’ en varios aspectos. Entre los que destaca que este último muestra los procesos en una lista en lugar de un diagrama, pero proporciona información más detallada sobre estos.

El programa pstree facilita información sobre la finalización de una serie de procesos relacionados entre sí, esto es, todos los descendientes de un proceso particular. El programa deja claro desde un principio que proceso es el primario y cuales son los secundarios. Esto evita buscar las relaciones entre los diversos procesos de manera manual.

### Comando kill

Los sistemas Linux vienen con la herramienta **KILL**instalada, que usaremos para detener los procesos que necesitemos. Por defecto el comando kill envía una señal denominada TERM a un proceso que le pasaremos mediante su **PID**como argumento. Esta señal TERM pedirá a dicho proceso que termine, permitiéndole gestionar su función de cierre, completando las tareas necesarias y limpiando la información que ha cargado en memoria.

kill [PID del proceso

# Capítulo 4. Multithreads y concurrencia

## Preguntas

### Se dice que existe una pseudosimultaneidad cuando:

a) Una CPU con un solo núcleo tiende que compartir su tiempo de proceso entre varias tareas, tal que en un instante de tiempo está ejecutando varias instrucciones de programa a la vez

b) El SO organiza y calcula los hilos de forma óptima tal que varios usuarios (al frente de diferentes procesos) tienen la impresión de estar atendidos a la vez por una CPU mononúcleo

c) Una CPU multinúcleo donde se procesan varias tareas de forma simultánea.

d) Una CPU con varios microprocesadores donde se procesan varias tareas de forma simultánea

### Respecto a multithreading:

a) La concurrencia o multithreading de una aplicación depende exclusivamente de la gestión del SO

b) El efecto de concurrencia o multithreading de una aplicación depende exclusivamente del programador que la crea.

c) La simultaneidad real de una aplicación creada con hilos o tareas concurrentes necesita que la CPU sea multinúcleo.

d) Una CPU multinúcleo siempre producirá mayor velocidad de proceso que una mononúcleo al ejecutar cualquier aplicación

### Se dispone de un ordenador monoprocesador que permite multiprogramación, para lo cual es preciso:

a) Utilizar técnicas de multiplexación en el tiempo virtual

b) Utilizar técnicas de ejecución de programas en tiempo real

c) Utilizar técnicas de ejecución en tiempo compartido

d) Fragmentar de forma lógica la memoria caché en tantas partes como programas en ejecución tengamos

### **En relación con los Sistemas Operativos, ¿Qué se entiende por** **overhead?**

a) Porcentaje de tiempo que transcurre entre la cabecera (header) del primer paquete y el siguiente

b) Porcentaje de tiempo en el que la CPU está atendiendo a tareas del propio sistema

c) Tiempo que tarda la cabeza del disco, una vez posicionada en la pista, en acceder al sector que le interesa

d) Porcentaje de tiempo que la CPU gasta en tratar los datos de entrada

### Uno de los beneficios de la programación multiproceso (concurrente) pueden ser (marcar la respuesta FALSA):

a) El uso de subprocesos múltiples en una aplicación interactiva puede permitir que un programa continúe ejecutándose incluso si parte de él está bloqueado o está realizando una operación prolongada, lo que aumenta la capacidad de respuesta para el usuario.

b) Supongamos que en un GUI de una aplicación un usuario, al hacer clic en un botón de una pantalla, se desencadena una operación que requiere mucho tiempo de proceso. Una aplicación de thread único no respondería al usuario que quedaría bloqueado hasta que se haya completado la operación. En una aplicación creada con multithreding también habría que esperar a que ese duradero subproceso termine para avisar al usuario con un Thread que puede seguir introduciendo texto.

c) El beneficio de compartir código y datos permite que una aplicación tenga varios subprocesos de actividad diferentes dentro del mismo espacio de direcciones.

d) La creación de subprocesos consume menos tiempo y memoria que la creación de procesos.

### Un kernel de un buen sistema operativo multinúcleo dispone de programas que deben resolver gestiones de multithreding como (SEÑALE LA FALSA):

a) Identificar en las aplicaciones a ejecutar los hilos programados que podrían ser ejecutadas en paralelo en los núcleos individuales de la CPU

b) Resolver la dependencia de una tarea respecto a los datos de otra asegurándose de que las tareas están sincronizadas para adaptarse a la dependencia de los datos

c) Balancear la carga computacional de diferentes tareas concurrentes para optimizar el tiempo de ejecución de la aplicación.

d) Cuando un programa se ejecuta en paralelo en múltiples núcleos, son posibles muchas rutas de ejecución diferentes. Probar y depurar tales programas concurrentes es inherentemente más fácil que probar y depurar aplicaciones de un solo subproceso.

### La tecnología HyperThreading consiste en

a) ‘Simular’ dos procesadores (o núcleos físicos) en uno solo lógico, dividiendo la carga de trabajo entre ambos y por tanto mejorando la velocidad de procesamiento.

b) Es una lógica que Apple ya había implementado desde hace tiempo que consiste en duplicar algunos elementos dentro del procesador como los registros o las memorias caché de tercer nivel

c) Podemos ver procesadores de 2, 4, 6 u 8 núcleos físicos de Intel, que gracias a la tecnología HyperThreading, pueden simular el doble de núcleos de procesamiento.

d) Los procesadores Intel Core i3 poseen solo dos núcleos lógicos pero se comporta como si tuviese 4 núcleos físicos gracias a esta tecnología.

### Una tecnología que “engaña” al sistema operativo al hacerle creer que existen dos núcleos cuando en realidad solo existe uno

a) Es el HyperThreading de las máquinas Macintosh

b) Intel fue pionera con su tecnología HyperThreading,

c) Es la llamada CMT (Cluster Based Multithreading) de los procesadores AMD

d) Es la llamada SMT (Simultaneous Multi-Threading) de Intel

### **Aplicando la Ley de Amdahl, supongamos que tenemos una aplicación que es 50 por ciento paralela y 50 por ciento serie. Si ejecutamos esta aplicación en un sistema con dos núcleos de procesamiento, podemos obtener una aceleración** de:

a) 1,6 veces

b) 1,3 veces

c) 1,2 veces

d) 2,6 veces

### El soporte para subprocesos se puede proporcionar a nivel de usuario, o por el kernel

a) Subprocesos (hilos) a nivel kernel son administrados directamente por la CPU

b) Subprocesos (hilos) a nivel usuario son administrados directamente por el kernel,

c) Linux y macOS, admiten subprocesos del kernel.

d) Prácticamente todos los sistemas operativos contemporáneos admiten subprocesos del kernel, excepto Windows

### El modelo de gestión de subprocesos del SO de tipo muchos a uno:

a) Asigna muchos subprocesos a nivel de usuario (compilador) a un subproceso del kernel y la gestión de subprocesos la realiza la biblioteca de subprocesos en el espacio del SO.

b) Todo el proceso se bloqueará si un subproceso realiza una llamada al sistema de bloqueo.

c) Como cualquier subproceso puede acceder al kernel, múltiples subprocesos pueden ejecutarse en paralelo en un sistema multinúcleo.

d) Es el mejor modelo para aprovechar los múltiples núcleos de una CPU multicore.

### El modelo de gestión de subprocesos del SO de tipo uno a uno:

a) Asigna cada subproceso de usuario a un subproceso de la CPU

b) Permite que varios subprocesos se ejecuten en paralelo en multiprocesadores siempre que la creación de un subproceso de usuario se corresponda con la creación del subproceso del kernel correspondiente

c) Linux, junto con la familia de sistemas operativos Windows, no implementan el modelo uno a uno.

d) Multiplexa muchos threads a nivel de usuario en un número mayor de threads del kernel.

## Ejercicios

### Comandos sobre información de CPU

* Pruebe a ejecutar los comandos de Linux que se describen al final del capítulo relacionados con la información de la CPU.
* Haga una breve explicación e interpretación de los principales resultados obtenidos
* Añada otros comandos con variaciones de los anteriores (utilizando otros parámetros)
* Añada otros comandos con relación a la información de procesos y cores, como:

sudo apt-get install cpufrequtils

cpufreq-info

xavier@ubuntuserver:~$ cpufreq-info

cpufrequtils 008: cpufreq-info (C) Dominik Brodowski 2004-2009

Report errors and bugs to cpufreq@vger.kernel.org, please.

analyzing CPU 0:

no or unknown cpufreq driver is active on this CPU

maximum transition latency: 4294.55 ms.

sudo apt-get install cpuid

cpuid

Si se tiene un Intel core i3 i5 i7

sudo apt-get install i7z

sudo i7z

sudo apt-get install inxi

inxi -C

sudo apt-get install hardinfo

hardinfo

### ¿Cómo sabemos si tenemos activado el Hyper Threading?

El Hyper Threading es una característica que hay que activar desde la BIOS pero para entrar, hay que reiniciar el sistema y eso es algo que no siempre podemos hacer cuando queramos.

Con comandos de Linux también podemos saber si está activo el Hyper Threading sin necesidad de rebotar el sistema. Hay varios métodos para ello:

* Con el comando *lscpu*:

xavier@ubuntuserver:~$ lscpu |grep -i thread

Thread(s) per core: 1

* Con el comando *dmidecode*:

dmidecode -t processor | grep Count

xavier@ubuntuserver:~$ sudo dmidecode -t processor | grep Count

xavier@ubuntuserver:~$

Sin resultado

* Revisando el archivo /sys/devices/system/cpu/smt/active:

xavier@ubuntuserver:~$ cat /sys/devices/system/cpu/smt/active

0

### ¿Cómo contar el número de Threads que está utilizando un proceso de Linux?

Una vez que tenemos el Hyperthreading activado, podemos comprobar si los procesos están utilizando multithreading, si están programados para ello.

En Linux, encontraremos esta información en /proc/PID/status:

xavier@ubuntuserver:~$ pstree -p

systemd(1)─┬─agetty(665)

├─dbus-daemon(579)

├─multipathd(394)─┬─{multipathd}(399)

│ ├─{multipathd}(401)

│ ├─{multipathd}(402)

│ ├─{multipathd}(403)

│ ├─{multipathd}(404)

│ └─{multipathd}(405)

├─networkd-dispat(585)

├─packagekitd(1222)─┬─{packagekitd}(1223)

│ └─{packagekitd}(1224)

├─polkitd(1226)─┬─{polkitd}(1227)

│ └─{polkitd}(1229)

├─sshd(602)─┬─sshd(933)───sshd(963)───bash(964)───pstree(3151)

│ └─sshd(2377)───sshd(2496)───bash(2497)

├─systemd(936)───(sd-pam)(939)

├─systemd-journal(356)

├─systemd-logind(588)

├─systemd-network(443)

├─systemd-resolve(555)

├─systemd-timesyn(556)───{systemd-timesyn}(567)

├─systemd-udevd(400)

└─unattended-upgr(620)───{unattended-upgr}(655)

xavier@ubuntuserver:~$ cat /proc/399/status |grep -i thread

Threads: 7

# Capítulo 5. Scheduling (planificación)

## Preguntas:

### Además de elegir el proceso idóneo a ejecutar, el scheduler también tiene que hacer un uso eficiente de la CPU, ya que el cambio de contexto es costoso. Indique cúal de las siguientes acciones pone en evidencia el coste computacional de cambio de contexto

a) En el cambio de contexto el proceso permanece siempre en modo kernel.

b) Se debe guardar el PCB del proceso actual para que puedan recargarse más tarde.

c) En algunos sistemas, también se debe guardar el mapa de memoria (por ejemplo, los bits de referencia de la memoria de disco duro en la tabla de páginas).

d) En el cambio de proceso la memoria caché y las tablas relacionadas permanecen inalterables, lo que obliga a borrarlas dos veces (al ingresar al kernel y al salir de él).

### Respecto a que casi todos los procesos alternan ráfagas de computación (instrucciones aritmético lógicas o transferencias con la memoria) con solicitudes de E/S (de disco, red, etc), indique la respuesta verdadera:

a) Procesos vinculados a la computación (o vinculados a la CPU) pasan la mayor parte computando instrucciones. Tienen ráfagas de CPU cortas y, por lo tanto, esperas de E/S poco frecuentes

b) Procesos vinculados a E/S, pasan la mayor parte del tiempo esperando a que concluya su trabajo en un dispositivo E/S. Tienen ráfagas de CPU largas y, por lo tanto, esperas de E/S poco frecuentes

c) Procesos vinculados a la computación (o vinculados a la CPU) pasan la mayor parte computando instrucciones. Suelen tener ráfagas de CPU largas y, por lo tanto, esperas de E/S poco frecuentes

d) Procesos vinculados a E/S, pasan la mayor parte del tiempo esperando a que concluya su trabajo en la memoria. Tienen ráfagas de CPU largas y, por lo tanto, esperas de E/S poco frecuentes

### Respecto a la alternancia en los procesos de ráfagas de computación (instrucciones aritmético lógicas o transferencias con la memoria) con solicitudes de E/S (de disco, red, etc), indique la respuesta verdadera:

a) Algunas actividades de E/S son de computación. Por ejemplo, cuando la CPU copia bits a una RAM de video para actualizar la pantalla, está computando, no realizando E/S, aunque la CPU está implicada en ello.

b) Si las CPU se vuelven más rápidas, los procesos tienden a vincularse más a las E/S.

c) El factor clave para considerar que un proceso es de CPU o de E/S es la duración de la ráfaga (o ráfagas) de la CPU, no la duración de la ráfaga de E/S.

d) Los procesos vinculados a E/S necesitan mucho tiempo de CPU, y tienen solicitudes de E/S especialmente cortas.

### Explique brevemente la planificación apropiativa (preemptive)

La planificación apropiativa consiste en el control por parte de la CPU del régimen de trabajo. Es decir, se asignan franjas de tiempo a cada proceso de forma que se ejecuta solamente lo que se pueda en ese espacio de tiempo, guardando los datos para continuar después en un PCB. Cuando vuelve a tocar el proceso se carga el PCB y se inicia por donde se quedó.

### Explique brevemente la planificación no apropiativa (no preemptive)

La planificación no apropiativa consiste en no asumir el control de la ejecución. Es decir, las tareas se ejecutan desde su inicio hasta su final sin ser interrumpidas en ningún momento. Hay otra forma de planificación apropiativa en el que concede franjas de tiempo muy muy largas.

## Ejercicios:

### Observe la utilización de la CPU mediante el comando *monitor* en los sistemas Linux, macOS (Figura 4) y UNIX

Parámetros más relevantes:

* + Tiempo de tránsito (Throughput).
  + Tiempo de proceso completo.
  + Tiempo de espera.
  + Tiempo de respuesta.

### Comando vmstat

Utilizando el comando **vmstat** que está disponible en los sistemas Linux, investigue la frecuencia en la que ocurren los cambios de contexto

Observe el comando y explique su salida

**vmstat 1 3**

Este comando proporciona 3 líneas de salida con un retraso de 1 segundo:

POR EJEMPLO:

24

225

339

Explicación:

* Desde que esta máquina arrancó, ha promediado 24 cambios de contexto por segundo. Y en el último segundo
* En el último segundo, se realizaron 225 cambios de contexto
* En l penúltimo segundo se realizaron 339 cambios de contexto

### Contenido del fichero status de un proceso

Utilice el sistema de archivos /proc para determinar la cantidad de cambios de contexto para un proceso determinado. Por ejemplo, el contenido del archivo /proc/2166/status enumerará varias estadísticas para el proceso con pid = 2166.

**cat /proc/2166/status**

Esta salida muestra el número de cambios de contexto durante la vida útil del proceso.

* Un cambio de contexto voluntario ocurre cuando un proceso ha cedido el control de la CPU porque requiere un recurso que actualmente no está disponible (como el bloqueo de E/S).
* Un cambio de contexto no voluntario ocurre cuando se le quita la CPU a un proceso, como cuando su intervalo de tiempo ha expirado o ha sido reemplazado por un proceso de mayor prioridad.

### Ejercicio sobre SJF.

### Considere tres procesos, todos llegando a la vez a la cola de “listo”, con un tiempo total de ejecución de 10, 20 y 30 unidades, respectivamente.

### Cada proceso dedica el primer 20% del tiempo de ejecución a realizar E/S, el siguiente 70% del tiempo a realizar cálculos y el último 10% del tiempo a realizar E/S nuevamente. El sistema operativo utiliza un algoritmo de programación del tiempo de cálculo restante más corto (SJF) y despacha un nuevo proceso cuando el proceso en ejecución se bloquea en E/S o cuando el proceso en ejecución finaliza su ráfaga de cálculo. Suponga que todas las operaciones de E/S se pueden superponer tanto como sea posible. ¿Durante qué porcentaje de tiempo permanece inactiva la CPU?

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 7 | 1 | 4 | 14 | 2 | 6 | 21 | 3 |
| E/S | CPU | E/S | E/S | CPU | E/S | E/S | CPU | E/S |
| Proceso 1º | | | Proceso 2º | | | Proceso 3º | | |

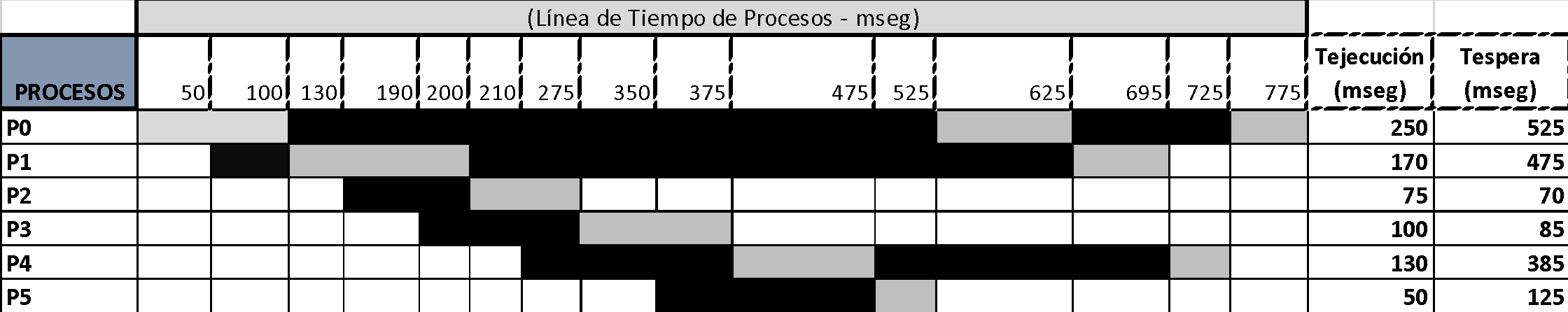
Tiempo total de ejecución de los 3 procesos = 10 + 20 + 30 = 60 unidades de tiempo.

% Tiempo que la CPU permanece inactiva = 18 / 60 \* 100 = 30 %.

### Ejercicio de algoritmo de planificación RR.

### Considere la siguiente tabla con el tiempo de llegada y el tiempo de ejecución de varios procesos en estado “listo”; un cuanto de 100 ms y un algoritmo de scheduling por turnos (RR)

|  |  |  |
| --- | --- | --- |
| **Process name** | **Arrival time** | **Execute time** |
| P0 | 0 | 250 |
| P1 | 50 | 170 |
| P2 | 130 | 75 |
| P3 | 190 | 100 |
| P4 | 210 | 130 |
| P5 | 350 | 50 |



Tiempo de vuelta completa (*turnaround*) = 525 mseg.

Tiempo medio de espera = 277,5 mseg.

# Capítulo 6. Sincronismo

## Preguntas

### ¿Cuándo se establece la exclusión mutua?

Relacionado con la ejecución de procesos concurrentes en el Sistema Operativo, uno de los problemas que se presenta tiene que ver con garantizar que dos o más procesos o subprocesos no se intercepten entre sí o se coordinen en el empleo de recursos críticos. La condición de exclusión mutua establece que solamente se permite a un proceso estar dentro de la misma región crítica. Esto es, que en cualquier momento solamente un proceso puede usar un recurso a la vez.

### ¿Qué técnicas se usan en la exclusión mutua?

Para lograr la exclusión mutua generalmente se usan algunas técnicas para lograr entrar a la región crítica: semáforos, monitores, el algoritmo de Dekker y Peterson, los 'candados'.

### ¿Qué es una red de Petri?

Una Red de Petri es una herramienta matemática con representaciones gráficas que permiten modelar sistemas con flujos de información concurrente, y capaces de mostrar el comportamiento dinámico del sistema.

### **¿Por qué es útil Red de Petri en el diseño y estudio de sistemas** operativos?

Porque existe necesidad de modelar la concurrencia y la sincronización de procesos que los Sistemas Operativos deben implementar, ayudando a resolver los tres problemas a este respecto que se encontraban: comunicación entre procesos, exclusión mutua y sincronización de procesos.

### Diferencias entre cerrojos y semáforos

Los cerrojos (locks) y los semáforos son dos mecanismos de sincronización utilizados en programación concurrente para controlar el acceso a recursos compartidos y evitar problemas de concurrencia, pero difieren en sus características y en cómo se utilizan.

En cuanto **granularidad**:

Los **cerrojos** son mecanismos de sincronización que proporcionan exclusión mutua **a nivel de recurso específico**. Cuando un hilo adquiere un cerrojo, tiene acceso exclusivo al recurso asociado al cerrojo y bloqueará a otros hilos hasta que lo libere.

**Los semáforos**, tanto binarios como de contador, pueden utilizarse para **controlar el acceso a uno o varios recursos compartidos**. Se puede usar un semáforo para **controlar el acceso a múltiples instancias de un recurso** o para **implementar cualquier tipo de lógica de sincronización** que se desee.

**Simplicidad**

Los **cerrojos son relativamente simples de entender y usar**, especialmente en contextos donde se necesita exclusión mutua para un recurso específico.

Los **semáforos pueden ser más flexibles pero también más propensos a errores** si no se utilizan correctamente, ya que puedes implementar una amplia variedad de lógicas de sincronización con ellos.

**Usos típicos**

Los **cerrojos son ideales cuando necesitas proteger el acceso a recursos críticos específicos,** como estructuras de datos compartidas, archivos o bases de datos.

Los **semáforos son útiles cuando necesitas implementar patrones de sincronización más complejos**, como la sincronización entre un número variable de hilos o la implementación de barreras de sincronización.

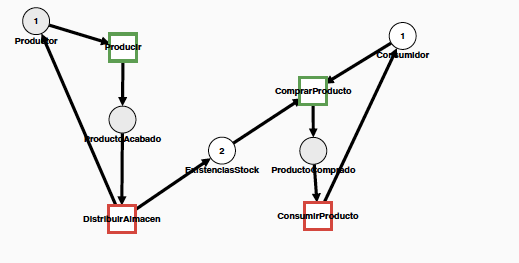
**Funcionalidad**

Los **cerrojos suelen proporcionar una funcionalidad más limitada** en comparación con los semáforos. Su objetivo principal es proporcionar exclusión mutua, lo que significa que un hilo puede bloquear o esperar para adquirir un cerrojo si otro hilo ya lo ha adquirido.

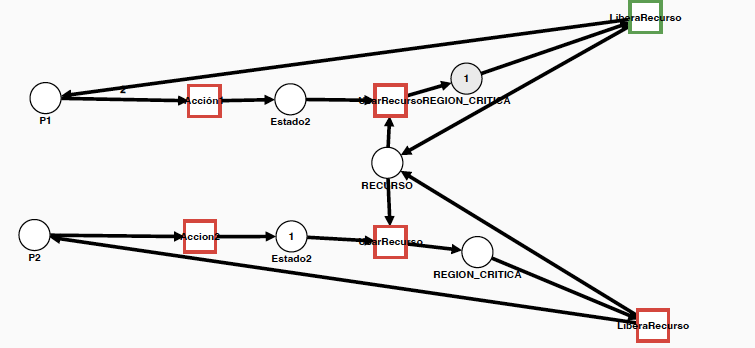
Los **semáforos pueden utilizarse para una variedad de tareas**, desde controlar el acceso a recursos hasta sincronizar múltiples hilos en puntos específicos de ejecución.

## Ejercicio

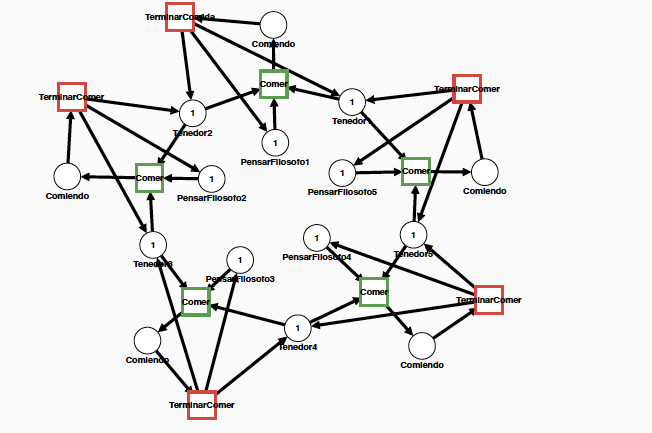
### Diseñar el modelo Productor Consumidor utilizando la herramienta para Redes de Petri PIPE (<https://github.com/sarahtattersall/PIPE/releases>)



### Crear un modelo de MUTEX (exclusión mutua) con Red de Petri



### Diseñar una Red de Petri que emule el problema de los filósofos cenando como modelo de 5 procesos concurrente que se disputan recursos comunes (tenedores). Explicar los resultados



C1, C2, C3, C4 y C5 son los tenedores. Si están marcados están disponibles.

M1, M2, M3, M4 y M5 estado PENSAR de los filósofos 1, 2, 3, 4 y 5. Si están marcados están pensando

E1, E2, E3, E4 y E5 estado COMER de los filósofos 1, 2, 3, 4 y 5. Si están marcados están comiendo.

# Capítulo 7. Memoria

### Mencione dos diferencias entre direcciones lógicas y físicas.

Una **dirección generada por la CPU** se conoce comúnmente como una dirección lógica (también llamada dirección virtual). **Una dirección vista por la Unidad de Gestión de la Memoria** (*Memory Management Unit,* MMU), es decir, la **cargada en el registro de la memoria física**, es conocida comúnmente como una dirección física.

El **espacio de direccionamiento lógico de memoria puede ser mayor que el direccionamiento en memoria física**. No existe correspondencia inmediata entre una dirección virtual y una física, requiriéndose la necesidad de un hardware de traducción de direcciones que vincule la una con la otra. Por ello, el espacio de direccionamiento lógico puede hacer referencia a espacios de memoria que no se encuentren en la memoria física, como cuando una página se aloja en el almacén de respaldo o swap. No obstante, el programa ve la memoria como un solo espacio, que contiene solo este programa.

### ¿Por qué los tamaños de página son siempre potencias de 2?

El tamaño de la página (al igual que el tamaño del marco) está definido por el hardware. El tamaño de una página es una potencia de 2, que suele oscilar entre 4 KB y 1 GB por página, dependiendo de la arquitectura del ordenador, y el motivo es porque siendo potencia de 2 hace que la traducción de una dirección lógica en un número de página y un desplazamiento de página sea particularmente fácil.

Esto se debe a que el tamaño del espacio de direcciones lógicas es una potencia de 2, 2^m. Si el tamaño de una página es una potencia de 2; 2^n, hallar el número de páginas total será una operación sencilla en binario: 2^(m-n), por lo que con m-n bits podremos representar todos los posibles números de página. De esta manera con m bits podremos representaren un sólo número binario el número de página y el desplazamiento. El número de página estará compuesto por los m-n primeros bits y el desplazamiento por los n últimos bits, ya que el tamaño de página es 2^n. De esta manera obtener con una dirección de memoria de m bits obtener el número de página y el desplazamiento se obtiene aplicando la correspondiente máscara binaria, una operación sencilla.

### ¿Cuáles son las funciones de una MMU?

Como funciones de MMU se tienen las siguientes:

* La traducción de las direcciones lógicas (o virtuales) a direcciones físicas (o reales) mediante la Tabla de Páginas.
* El control del Buffer de Traducción Adelantada (TLB, Translation Lookaside Buffer).
* La protección de la memoria.

### Considere un espacio de direcciones lógicas de 64 páginas de 1024 palabras cada una, mapeadas en una memoria física de 32 marcos. ¿Cuántos bits hay en la dirección lógica? ¿Cuántos bits hay en la dirección física?

64 páginas lógicas = 2^6 => 6 bits.

Tamaño de página = 1 KB = 1^10 => 10 bits.

Número de bits de la dirección lógica = 16 bits.

32 marcos = 2^5 => 5 bits.

Tamaño de página = 1 KB = 1^10 = 2^10 => 10 bits.

Número de bits de la dirección lógica = 15 bits.

### ¿Cuál es el efecto de permitir que dos entradas en una tabla de páginas apunten al mismo marco de página en la memoria? Explique cómo podría usarse este efecto para disminuir la cantidad de tiempo necesario para copiar una gran cantidad de memoria de un lugar a otro. ¿Qué efecto tendría la actualización de algún byte en una página en la otra página?

El efecto de permitir que dos entradas en una tabla de páginas apunten al mismo marco de página en la memoria física es poder aprovecharse de la ventaja que supone compartir código común, una consideración que es particularmente importante en un entorno con múltiples procesos, traduciéndose en ahorro de espacio en memoria con copias duplicadas por cada proceso y ahorro de tiempo por latencias debidas a la carga de unas mismas secciones de código a memoria desde el área de swapping u otras áreas de memoria no compartidas.

Ahora bien, estas ventajas están supeditadas a que el código es un código de reentrada, es decir, que no se modifica automáticamente: nunca cambia durante la ejecución, de lo contrario se producirían inconsistencias que ocasionarían errores fatales en ejecución de los procesos y generarían su correspondiente interrupción o trampa.

El hecho de actualizar un byte en una página supondría la modificación del contenido de la otra página ya que realmente las dos páginas apuntan al mismo espacio en memoria física. De ahí que esta técnica sólo es apropiada para datos que no se modifiquen en tiempo de ejecución. Por ejemplo una librería.

### 6 Suponiendo un tamaño de página de 1 KB, ¿cuáles son los números de página y los desplazamientos para las siguientes referencias de direcciones (proporcionadas como números decimales):

**a. 3085:** 3085: 3 x 1024 + 13

**b. 42095:** 41 x 1024 + 111

**C. 215201:** 210 x 1024 + 161

**d. 650000:** 634 x 1024 + 784

**e. 2000001:** 1953 x 10 + 129

### Considere un espacio de direcciones lógicas de 256 páginas con un tamaño de página de 4 KB, mapeado en una memoria física de 64 marcos.

### ¿Cuántos bits se requieren en la dirección lógica?

256 páginas lógicas = 2^8 => 8 bits.

Tamaño de página = 4 KB = 4 x 2^10 = 2^12 => 12 bits.

Número de bits de la dirección lógica = 20 bits.

### b. ¿Cuántos bits se requieren en la dirección física?

64 marcos = 2^6 => 6 bits.

Tamaño de página = 4 KB = 4 x 2^10 = 2^12 => 12 bits.

Número de bits de la dirección lógica = 18 bits.

### Un ordenador emplea 32 bits para las direcciones virtuales de 32 bits. Dispone de una memoria principal de 512 MB, que emplea páginas de 4 KB. Indique el formato de la dirección virtual (número de bits para páginas y bits para desplazamiento) y el número de marcos de la página de este ordenador.

Memoria virtual = 2^32 / 2^12 = 2^20 páginas virtuales = **1.048.576 páginas virtuales.**

Memoria principal = 512 MB = 2^9 x 1024 x 1024 / 4 x 1024 = 2 ^7 x 1024 = 2 ^7 x 2^10 = 2^17 = **131.072 marcos**.

### 9 ¿Qué es la memoria SWAP? ¿Dónde se encuentra la memoria swap? ¿cuándo es necesaria? ¿Qué inconvenientes tiene?

La memoria SWAP es un concepto muy vinculado a la paginación de memoria que se realiza en los Sistemas Operativos, por el cual se hace posible que el espacio total de direccionamiento de memoria para todos los procesos exceda la memoria física real del sistema, aumentando así el grado de multiprogramación en un sistema. Las instrucciones de proceso y los datos sobre los que operan deben estar en la memoria para ser ejecutados.

Sin embargo, un proceso, o una parte de un proceso, puede intercambiarse temporalmente fuera de la memoria a **un almacén de respaldo conocido como memoria SWAP**, como así sucede **en los cambios de contexto o por ejemplo, por tratarse de procesos que llevan bloqueados largo tiempo o han dejado de utilizarse,** pero que posteriormente podrían volver a cargarse en memoria para continuar con la ejecución cuando corresponda. Este intercambio de la memoria principal a un almacenamiento de respaldo externo hace posible que el espacio total de direcciones físicas de todos los procesos exceda la memoria física real del sistema, aumentando así el grado de multiprogramación en un sistema.

Como inconvenientes, cabría subrayar que, al tratarse de un almacén de respaldo externo, la transferencia del proceso o subproceso alojado en esta memoria SWAP va a presentar una latencia o retardo cuando llegue el momento de cargarlo en la memoria principal para proceder a continuar o iniciar su ejecución.

## EJERCICIOS

### ¿Por qué Windows consume tanta RAM? (superfetch)

El sistema operativo Windows, al analizar el uso de RAM, puede dar la sensación de que consume en un alto porcentaje la memoria RAM. Esto es cierto parcialmente porque la mayor parte de ese porcentaje se dedica a la carga de las aplicaciones habituales del usuario de forma que el sistema arranque en términos generales mucho más rápido, por lo que en realidad consume la RAM que prevé que el usuario va a utilizar. Esta técnica para cargar los datos de los programas más usados en la RAM al arrancar es lo que se conoce como superfetch.

### **¿Qué es la presión de la memoria y cómo puedo aliviarla?**

La presión de memoria se refiere a una situación en la que la memoria principal de un dispositivo se encuentra sobrecargada y no se pueden manejar eficientemente todas las aplicaciones y procesos que se están ejecutando. Esto puede llevar a una disminución del rendimiento y a una mayor lentitud del sistema. Para aliviar la presión de memoria, se puede intentar:

* Cerrar aplicaciones innecesarias.
* Reiniciar el dispositivo.
* Desactivar aplicaciones en segundo plano.
* Eliminar archivos y programas no deseados.
* Actualizar el sistema operativo.

# Capítulo 8. Entrada/Salida

## Preguntas

### ¿Qué función realiza un controlador de dispositivo en un sistema operativo?

a) Administrar recursos de CPU

b) Gestionar la memoria principal

c) Interactuar con dispositivos de hardware

d) Controlar el planificador de tareas

### ¿Qué es la E/S por bloques en sistemas operativos?

a) Un método para E/S donde los datos se transfieren uno a uno.

b) Un método para E/S donde los datos se transfieren en bloques más grandes.

c) Un método para E/S que no se utiliza en sistemas modernos.

d) Un método para E/S que solo funciona con dispositivos de almacenamiento.

### ¿Qué es una interrupción de hardware en un sistema operativo?

a) Una solicitud de un programa en ejecución.

b) Una señal generada por un dispositivo de hardware para solicitar atención del sistema operativo.

c) Un error en el código del sistema operativo.

d) Un proceso de planificación de tareas.

### ¿Qué significa DMA en el contexto de E/S en sistemas operativos?

a) Acceso Directo a la Memoria

b) Data Management Algorithm

c) Device Monitoring Application

d) Data Movement Accelerator

### ¿Cuál de las siguientes no es una función de un controlador de dispositivo?

a) Traducir comandos del sistema operativo en operaciones de hardware.

b) Administrar la memoria principal del sistema.

c) Gestionar recursos compartidos.

d) Optimizar el rendimiento del dispositivo.

### ¿Qué es el pooling de E/S en sistemas operativos?

a) Una técnica para gestionar recursos de CPU.

b) Una técnica para compartir recursos de memoria principal.

c) Una técnica para administrar recursos de E/S compartidos de manera eficiente.

d) Una técnica para dividir tareas de E/S en bloques.

### ¿Cuál es el propósito principal de las interrupciones de software en un sistema operativo?

a) Solicitar la atención de un dispositivo de hardware.

b) Iniciar una rutina específica en el sistema operativo para realizar una tarea.

c) Notificar un error en el sistema operativo.

d) Detener la ejecución del programa actual.

### ¿Qué tipo de interrupción es generada por dispositivos de hardware para notificar eventos como la finalización de una transferencia de datos?

a) Interrupción de reloj

b) Interrupción de software

c) Interrupción de tiempo real

d) Interrupción de E/S

### ¿Cuál es el propósito de una tabla de vectores de interrupción en un sistema operativo?

a) Almacenar datos de entrada y salida.

b) Asignar números únicos a los dispositivos de hardware.

c) Gestionar la memoria principal.

d) Priorizar las tareas del sistema operativo.

### ¿Qué beneficio proporcionan las interrupciones en un sistema operativo?

a) Mejoran la eficiencia en la administración de memoria.

b) Aceleran la velocidad de la CPU.

c) Evitan la necesidad de controladores de dispositivo.

d) Permiten una gestión eficiente de recursos y eventos.

## Ejercicios

### Comando para ver interrupciones

En Linux, para ver información relacionada con las interrupciones, utilizar el comando **cat /proc/interrupts**

Presentar un volcado de la salida y comentar esa información.

xavier@ubuntuserver:~$ cat /proc/interrupts

CPU0

0: 38 IO-APIC 2-edge timer

1: 9 IO-APIC 1-edge i8042

8: 0 IO-APIC 8-edge rtc0

9: 0 IO-APIC 9-fasteoi acpi

12: 158 IO-APIC 12-edge i8042

14: 0 IO-APIC 14-edge ata\_piix

15: 45898 IO-APIC 15-edge ata\_piix

18: 139420 IO-APIC 18-fasteoi vmwgfx

19: 245398 IO-APIC 19-fasteoi enp0s3

20: 0 IO-APIC 20-fasteoi vboxguest

21: 25995 IO-APIC 21-fasteoi ahci[0000:00:0d.0], snd\_intel8x0

NMI: 0 Non-maskable interrupts

LOC: 14062210 Local timer interrupts

SPU: 0 Spurious interrupts

PMI: 0 Performance monitoring interrupts

IWI: 0 IRQ work interrupts

RTR: 0 APIC ICR read retries

[…]

La información mostrada en pantalla muestra las CPU en columnas y los dispositivos en filas, de forma que indica las interrupciones que se producen en cada CPU debido a cada dispositivo. Además aporta información de las distintas interrupciones producidas por otros motivos como por ejemplo por el tiempo asignado (LOCAL TIMER), llamadas a funciones u otros.

### Comando para ver una lista de interrupciones

Deseamos obtener más información detallada sobre las interrupciones y su asignación en Linux, para ello utilizamos el comando:

**ls /proc/irq**

Comente la salida y observando cómo se puede navegar a un directorio específico para obtener detalles sobre una interrupción concreta.

xavier@ubuntuserver:~$ ls /proc/irq

0 10 12 14 18 2 21 4 6 8 default\_smp\_affinity

1 11 13 15 19 20 3 5 7 9

xavier@ubuntuserver:~$ ls -al /proc/irq

total 0

dr-xr-xr-x 23 root root 0 Oct 15 12:28 .

dr-xr-xr-x 162 root root 0 Oct 14 23:38 ..

dr-xr-xr-x 9 root root 0 Oct 15 12:28 0

dr-xr-xr-x 10 root root 0 Oct 15 12:28 1

dr-xr-xr-x 9 root root 0 Oct 15 12:28 10

dr-xr-xr-x 9 root root 0 Oct 15 12:28 11

dr-xr-xr-x 10 root root 0 Oct 15 12:28 12

dr-xr-xr-x 9 root root 0 Oct 15 12:28 13

dr-xr-xr-x 10 root root 0 Oct 15 12:28 14

dr-xr-xr-x 10 root root 0 Oct 15 12:28 15

dr-xr-xr-x 10 root root 0 Oct 15 12:28 18

dr-xr-xr-x 10 root root 0 Oct 15 12:28 19

dr-xr-xr-x 9 root root 0 Oct 15 12:28 2

dr-xr-xr-x 10 root root 0 Oct 15 12:28 20

dr-xr-xr-x 11 root root 0 Oct 15 12:28 21

dr-xr-xr-x 9 root root 0 Oct 15 12:28 3

dr-xr-xr-x 9 root root 0 Oct 15 12:28 4

dr-xr-xr-x 9 root root 0 Oct 15 12:28 5

dr-xr-xr-x 9 root root 0 Oct 15 12:28 6

dr-xr-xr-x 9 root root 0 Oct 15 12:28 7

dr-xr-xr-x 10 root root 0 Oct 15 12:28 8

dr-xr-xr-x 10 root root 0 Oct 15 12:28 9

-rw-r--r-- 1 root root 0 Oct 15 12:28 default\_smp\_affinity

Cada dispositivo tiene una carpeta asignada a su id de dispositivo donde se guarda la información referente a las interrupciones. /proc/irq es el directorio padre de cada uno de ellos por lo que /proc/irq/57 me permitirá navegar al dispositivo de E/S número 57 y ver dentro los datos correspondientes.

### Comando para ver la afinidad

**cat /proc/irq/[Número\_IRQ]/smp\_affinity**

Use este comando para ver o cambiar la afinidad de una interrupción a un núcleo de CPU específico.

Por ejemplo, para ver la afinidad de la IRQ 16:

**cat /proc/irq/16/smp\_affinity**

xavier@ubuntuserver:~$ cat /proc/irq/15/smp\_affinity

1

Este comando proporciona un número hexadecimal de 4 dígitos (máscara) que define la afinidad de un proceso. En el caso de la respuesta FFFF el proceso podría ejecutarse en cualquier núcleo. En el caso del 0200, el proceso solo podría ejecutarse en el tercer núcleo.

### Lista de los núcleos de CPU a los que se puede asignar una interrupción

**cat /proc/irq/[Número\_IRQ]/irq\_affinity\_list**

Este comando muestra una lista de los núcleos de CPU a los que se puede asignar una interrupción en sistemas con múltiples núcleos.

Por ejemplo:

**cat /proc/irq/16/irq\_affinity\_list**

**No funciona**

### Dispositivos PCI en el sistema

**lspci -v**

Este comando muestra información detallada sobre los dispositivos PCI en el sistema, incluyendo las interrupciones que utilizan. Se utiliza para ver qué IRQ está asignada a cada dispositivo.

Por ejemplo:

**lspci -v**

xavier@ubuntuserver:~$ lspci -v

00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)

Flags: fast devsel

00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]

Flags: bus master, medium devsel, latency 0

00:01.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01) (prog-if 8a [ISA Compatibility mode controller, supports both channels switched to PCI native mode, supports bus mastering])

Flags: bus master, fast devsel, latency 64

Memory at 000001f0 (32-bit, non-prefetchable) [virtual] [size=8]

Memory at 000003f0 (type 3, non-prefetchable) [virtual]

Memory at 00000170 (32-bit, non-prefetchable) [virtual] [size=8]

Memory at 00000370 (type 3, non-prefetchable) [virtual]

I/O ports at d000 [virtual] [size=16]

Kernel driver in use: ata\_piix

Kernel modules: pata\_acpi

00:02.0 VGA compatible controller: VMware SVGA II Adapter (prog-if 00 [VGA controller])

Subsystem: VMware SVGA II Adapter

Flags: bus master, fast devsel, latency 64, IRQ 18

I/O ports at d010 [size=16]

Memory at e0000000 (32-bit, prefetchable) [size=16M]

Memory at f0000000 (32-bit, non-prefetchable) [size=2M]

Expansion ROM at 000c0000 [virtual] [disabled] [size=128K]

Kernel driver in use: vmwgfx

Kernel modules: vmwgfx

La irq de cada dispositivo es la que se encuentra a la izquierda de la salida de los datos.

### Módulos kernel cargados en el sistema

**lsmod**

Muestra la lista de módulos del kernel cargados en el sistema. Los módulos del kernel a menudo son controladores de dispositivos.

Ejemplo:

**lsmod**

xavier@ubuntuserver:~$ lsmod

Module Size Used by

binfmt\_misc 24576 1

intel\_rapl\_msr 20480 0

joydev 32768 0

snd\_intel8x0 45056 0

intel\_rapl\_common 40960 1 intel\_rapl\_msr

intel\_powerclamp 24576 0

snd\_ac97\_codec 180224 1 snd\_intel8x0

rapl 20480 0

ac97\_bus 16384 1 snd\_ac97\_codec

snd\_pcm 143360 2 snd\_intel8x0,snd\_ac97\_codec

snd\_timer 40960 1 snd\_pcm

input\_leds 16384 0

snd 106496 4 snd\_intel8x0,snd\_timer,snd\_ac97\_codec,snd\_pcm

serio\_raw 20480 0

soundcore 16384 1 snd

vboxguest 45056 0

mac\_hid 16384 0

sch\_fq\_codel 20480 2

dm\_multipath 40960 0

scsi\_dh\_rdac 20480 0

[…]

### Lista de los dispositivos en el directorio /dev

ls /dev: Lista los dispositivos en el directorio /dev. Los nombres de los archivos en este directorio representan dispositivos de hardware y controladores de dispositivos.

Ejemplo:

**ls /dev**

xavier@ubuntuserver:~$ ls -al /dev

total 4

drwxr-xr-x 18 root root 4020 Oct 14 23:39 .

drwxr-xr-x 19 root root 4096 Oct 14 22:56 ..

crw-r--r-- 1 root root 10, 235 Oct 14 23:39 autofs

drwxr-xr-x 2 root root 320 Oct 14 23:39 block

drwxr-xr-x 2 root root 80 Oct 14 23:38 bsg

crw-rw---- 1 root disk 10, 234 Oct 14 23:39 btrfs-control

lrwxrwxrwx 1 root root 3 Oct 14 23:39 cdrom -> sr0

drwxr-xr-x 2 root root 3540 Oct 14 23:39 char

crw--w---- 1 root tty 5, 1 Oct 14 23:39 console

lrwxrwxrwx 1 root root 11 Oct 14 23:38 core -> /proc/kcore

crw------- 1 root root 10, 124 Oct 14 23:39 cpu\_dma\_latency

crw------- 1 root root 10, 203 Oct 14 23:38 cuse

drwxr-xr-x 7 root root 140 Oct 14 23:38 disk

brw-rw---- 1 root disk 253, 0 Oct 14 23:39 dm-0

drwxr-xr-x 2 root root 60 Oct 14 23:28 dma\_heap

drwxr-xr-x 3 root root 100 Oct 14 23:38 dri

crw------- 1 root root 10, 126 Oct 14 23:39 ecryptfs

crw-rw---- 1 root video 29, 0 Oct 14 23:38 fb0

lrwxrwxrwx 1 root root 13 Oct 14 23:38 fd -> /proc/self/fd

crw-rw-rw- 1 root root 1, 7 Oct 14 23:39 full

crw-rw-rw- 1 root root 10, 229 Oct 14 23:39 fuse

crw------- 1 root root 10, 228 Oct 14 23:39 hpet

drwxr-xr-x 2 root root 0 Oct 14 23:38 hugepages

crw------- 1 root root 10, 183 Oct 14 23:39 hwrng

crw------- 1 root root 89, 0 Oct 14 23:38 i2c-0

[…]

### Lista de dispositivos USB conectados al sistema

**lsusb**

Muestra una lista de dispositivos USB conectados al sistema. Esto es especialmente útil para identificar dispositivos USB y buscar información sobre ellos.

Ejemplo:

lsusb

xavier@ubuntuserver:~$ lsusb

xavier@ubuntuserver:~$

Es una maquina virtual sin USB

### Detección y carga de controladores de dispositivos

dmesg

Muestra el registro del kernel, que incluye información sobre la detección y carga de controladores de dispositivos durante el arranque del sistema.

Ejemplo:

xavier@ubuntuserver:~$ sudo dmesg

[sudo] password for xavier:

[ 0.000000] Linux version 5.15.0-86-generic (buildd@lcy02-amd64-086) (gcc (Ubuntu 11.4.0-1ubuntu1~22.04) 11.4.0, GNU ld (GNU Binutils for Ubuntu) 2.38) #96-Ubuntu SMP Wed Sep 20 08:23:49 UTC 2023 (Ubuntu 5.15.0-86.96-generic 5.15.122)

[ 0.000000] Command line: BOOT\_IMAGE=/vmlinuz-5.15.0-86-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv ro

[ 0.000000] KERNEL supported cpus:

[ 0.000000] Intel GenuineIntel

[ 0.000000] AMD AuthenticAMD

[ 0.000000] Hygon HygonGenuine

[ 0.000000] Centaur CentaurHauls

[ 0.000000] zhaoxin Shanghai

[ 0.000000] BIOS-provided physical RAM map:

[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable

[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved

[ 0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] reserved

[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x000000007ffeffff] usable

[ 0.000000] BIOS-e820: [mem 0x000000007fff0000-0x000000007fffffff] ACPI data

[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec00fff] reserved

[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee00fff] reserved

[ 0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffffff] reserved

[ 0.000000] NX (Execute Disable) protection: active

[ 0.000000] SMBIOS 2.5 present.

[…]

### Configuración de hardware del sistema, incluyendo la lista de dispositivos y controladores asociados

**lshw**

Muestra información detallada sobre la configuración de hardware del sistema, incluyendo la lista de dispositivos y controladores asociados.

xavier@ubuntuserver:~$ lshw

WARNING: you should run this program as super-user.

ubuntuserver

description: Computer

width: 64 bits

capabilities: vsyscall32

\*-core

description: Motherboard

physical id: 0

\*-memory

description: System memory

physical id: 0

size: 2GiB

\*-cpu

product: Intel(R) Xeon(R) CPU E3-1265L V2 @ 2.50GHz

vendor: Intel Corp.

physical id: 1

bus info: cpu@0

version: 6.58.9

width: 64 bits

capabilities: fpu fpu\_exception wp vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx rdtscp x86-64 constant\_tsc rep\_good nopl xtopology nonstop\_tsc cpuid tsc\_known\_freq pni pclmulqdq monitor ssse3 cx16 pcid sse4\_1 sse4\_2 x2apic popcnt aes xsave avx rdrand hypervisor lahf\_lm pti fsgsbase

configuration: microcode=25

\*-pci

description: Host bridge

product: 440FX - 82441FX PMC [Natoma]

vendor: Intel Corporation

physical id: 100

bus info: pci@0000:00:00.0

version: 02

width: 32 bits

clock: 33MHz

### Información sobre un módulo del kernel, incluyendo detalles sobre el controlador de dispositivo

modinfo

Proporciona información sobre un módulo del kernel, incluyendo detalles sobre el controlador de dispositivo y los parámetros del módulo.

Ejemplo:

xavier@ubuntuserver:~$ modinfo hid

filename: /lib/modules/5.15.0-86-generic/kernel/drivers/hid/hid.ko

license: GPL

author: Jiri Kosina

author: Vojtech Pavlik

author: Andreas Gal

srcversion: B75407A4CB55DC1C39A290D

depends:

retpoline: Y

intree: Y

name: hid

vermagic: 5.15.0-86-generic SMP mod\_unload modversions

sig\_id: PKCS#7

signer: Build time autogenerated kernel key

sig\_key: 43:38:30:48:E7:40:C8:81:E8:32:59:37:8F:13:6E:29:6C:D4:D8:C3

sig\_hashalgo: sha512

signature: 16:81:4A:F5:4B:93:F5:39:C6:9F:41:30:27:C6:FC:78:93:63:E5:0A:

6F:A8:CB:F9:2D:D0:F2:36:35:C6:48:8B:82:03:44:D5:09:7F:48:A1:

F1:7C:97:70:39:CE:2E:6B:80:6A:C5:50:8F:1B:8F:1F:EE:CA:15:9B:

7C:F2:D1:4B:5B:47:AE:E9:9A:19:AC:74:C0:BE:98:06:F9:5F:7E:70:

5A:70:33:10:5D:E1:B8:F1:CD:8F:ED:CC:F2:98:04:CF:99:37:D7:C0:

E0:55:DA:A1:68:84:DF:3A:F6:51:E2:65:54:D4:70:8C:69:AE:81:6D:

36:BC:88:A1:58:7B:0E:63:6E:B8:AC:F6:CB:7D:DF:25:DD:DD:44:A9:

30:D4:88:ED:7B:16:7F:5C:F4:9F:CC:9A:4D:4E:56:C6:B2:6F:29:DE:

D1:C5:25:E4:43:ED:AD:45:E1:F7:98:7A:23:13:4C:C3:C2:2A:E1:6B:

DC:C2:F1:55:A9:67:71:33:01:86:CE:97:FC:83:47:6A:4C:9C:6E:1D:

05:8B:21:2D:0E:07:F2:3C:85:12:B6:97:30:FB:1F:3D:F9:43:C9:A3:

23:2C:B8:19:99:9C:CD:40:E8:A2:77:51:48:54:E6:CC:FE:66:E3:40:

69:02:61:D1:F3:22:E6:2F:B8:51:D3:17:C0:E8:DA:7A:FC:79:86:3E:

3E:B3:FD:C6:64:94:DD:68:03:73:AD:77:AD:83:57:CD:86:73:7B:D2:

04:BD:B5:C3:41:7B:DD:8B:81:CA:25:51:FE:E8:61:D3:66:BD:B2:32:

9C:B1:6B:F0:2B:42:7D:3B:D8:C1:5A:54:D3:25:48:A8:4E:0C:6B:A2:

7F:8E:A9:6A:BA:1C:C3:1F:DA:8E:D6:06:B0:DC:4A:2C:F8:7C:47:77:

2E:E9:10:19:4E:31:A7:26:CC:B3:70:F3:BB:16:12:41:FE:D4:F3:CD:

DD:65:88:AE:98:B0:51:63:88:47:87:AE:21:3B:D6:04:17:91:DB:E2:

12:24:F5:81:90:67:0D:4C:B8:89:88:25:5F:18:C1:43:77:E2:03:E6:

A3:06:BE:BB:B9:75:6F:9B:90:0E:08:51:D6:21:AE:46:04:E5:2C:FE:

BA:70:76:B7:68:57:6F:E1:6F:9D:19:2E:B9:A9:E2:EB:53:4D:C3:D8:

44:9F:56:4F:90:E4:A4:62:83:85:01:A1:3D:A6:87:D5:E3:34:6D:35:

0E:46:E3:74:85:3A:E3:2B:92:EA:3C:E2:A8:D0:66:13:18:B4:FF:B3:

42:93:68:46:5F:63:FE:1E:22:09:00:6A:67:DE:D0:85:38:97:1C:F1:

04:76:9A:FA:85:77:A9:94:CB:DD:F9:E0

parm: debug:toggle HID debugging messages (int)

parm: ignore\_special\_drivers:Ignore any special drivers and handle all devices by generic driver (int)

### Información sobre los dispositivos de bloque

lsblk

Muestra información sobre los dispositivos de bloque (por ejemplo, discos duros y particiones) en el sistema, incluyendo información sobre los controladores y las rutas de los dispositivos.

Ejemplo:

xavier@ubuntuserver:~$ lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS

sda 8:0 0 50G 0 disk

├─sda1 8:1 0 1M 0 part

├─sda2 8:2 0 2G 0 part /boot

└─sda3 8:3 0 48G 0 part

└─ubuntu--vg-ubuntu--lv 253:0 0 24G 0 lvm /

sr0 11:0 1 1024M 0 rom

# Capítulo 9. Sistemas de archivos

## Preguntas

### ¿Qué es un inodo en un sistema de archivos Unix o Linux?

a) Un archivo de registro de transacciones.

b) Un sistema de archivos comprimido.

c) Una estructura de datos que almacena metadatos de archivos y directorios.

d) Un sistema de archivos de red distribuye metadatos de archivos y directorios.

### ¿Cuál es el propósito principal de la técnica de "journaling" en un sistema de archivos?

a) Reducir el tamaño de los archivos.

b) Mejorar la velocidad de lectura de archivos.

c) Mejorar la integridad de los datos y facilitar la recuperación después de fallos inesperados.

d) Comprimir archivos para ahorrar espacio en disco.

### ¿Cuál es la principal diferencia entre un sistema de archivos FAT32 y NTFS en Windows?

a) FAT32 admite journaling, mientras que NTFS no lo hace.

b) FAT32 es más rápido en la lectura de archivos grandes.

c) NTFS ofrece características avanzadas como journaling y permisos de acceso.

d) FAT32 es más eficiente en el uso del espacio en disco.

### ¿Cuál de los siguientes sistemas de archivos es más adecuado para sistemas de archivos grandes y aplicaciones de alto rendimiento en Linux?

a) Ext4

b) FAT32

c) XFS

d) HFS+

### ¿Qué hace el comando `tee` en Linux?

a) Calcula la cantidad total de espacio en disco utilizado por archivos.

b) Redirige la salida de un comando hacia un archivo y muestra la salida en la terminal.

c) Monta un sistema de archivos en el sistema.

d) Comprime archivos y directorios en un archivo ZIP.

### ¿Cuál es el propósito principal de un sistema de archivos en red (NFS) en Linux?

a) Mejorar la velocidad de acceso a archivos locales.

b) Proporcionar una interfaz gráfica para gestionar archivos y directorios.

c) Comprimir archivos para ahorrar ancho de banda en la red.

d) Compartir archivos y directorios entre sistemas en una red.

### ¿Qué es el Sistema de Archivos Virtual (VFS) en Linux?

a) Una capa de abstracción que proporciona una interfaz unificada para el acceso a sistemas de archivos de diferentes tipos.

b) Un sistema de archivos de red.

c) Un sistema de archivos comprimido.

d) Un sistema de archivos exclusivo de macOS.

### ¿Cuál es el sistema de archivos predeterminado en sistemas macOS más recientes?

a) FAT32

b) NTFS

c) HFS+

d) APFS

### ¿Qué tipo de sistema de archivos se utiliza comúnmente en discos ópticos como CD-ROMs y DVDs?

a) Ext4

b) ISO 9660

c) NTFS

d) XFS

### ¿Cuál es el propósito de un sistema de archivos comprimido?

a) Mejorar la velocidad de lectura de archivos.

b) Ahorrar espacio en disco mediante la reducción del tamaño de los archivos.

c) Facilitar la recuperación de datos después de un fallo del sistema.

d) Aumentar la seguridad de los archivos.

### ¿Qué hace el comando df en sistemas Unix o Linux?

a) Lista los archivos en un directorio específico.

b) Calcula el tamaño de un archivo.

c) Muestra información sobre el uso de espacio en disco y sistemas de archivos montados.

d) Desmonta un sistema de archivos.

### En un sistema de archivos Unix o Linux, ¿cuál es el propósito principal del directorio /home?

a) Almacenar copias de seguridad del sistema.

b) Almacenar archivos temporales del sistema.

c) Almacenar los directorios personales de los usuarios.

d) Almacenar archivos de sistema críticos.

### Algunos sistemas operativos proporcionan un cambio de nombre de llamada al sistema para darle un nuevo nombre a un archivo. ¿Existe alguna diferencia entre usar esta llamada para cambiar el nombre de un archivo y simplemente copiar el archivo a un archivo nuevo con un nuevo nombre y luego eliminar el anterior?

Copiar el archivo implica el duplicado de los datos en disco, con la consiguiente posibilidad de errores y tiempo invertido.

## Ejercicios

### Pruebe mostrar las particiones de disco en Linux, con el comando Fdisk con "-1" (como se indica a continuación) como usuario sudo o root.

**$ sudo fdisk -1**

a08@a08-HP-ProDesk-600-G5-SFF:~$ sudo fdisk -l

Disco /dev/sda: 447,13 GiB, 480103981056 bytes, 937703088 sectores

Disk model: CT480BX500SSD1

Unidades: sectores de 1 \* 512 = 512 bytes

Tamaño de sector (lógico/físico): 512 bytes / 512 bytes

Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes

Tipo de etiqueta de disco: gpt

Identificador del disco: FDCBEDED-6450-4811-834E-D648F23F15B9

Dispositivo Comienzo Final Sectores Tamaño Tipo

/dev/sda1 2048 1050623 1048576 512M Sistema EFI

/dev/sda2 1050624 937701375 936650752 446,6G Sistema de ficheros de Linux

Disco /dev/sdb: 931,51 GiB, 1000204886016 bytes, 1953525168 sectores

Disk model: WDC WD10EZEX-60W

Unidades: sectores de 1 \* 512 = 512 bytes

Tamaño de sector (lógico/físico): 512 bytes / 4096 bytes

Tamaño de E/S (mínimo/óptimo): 4096 bytes / 4096 bytes

Tipo de etiqueta de disco: gpt

Identificador del disco: 9BD5C14D-BEB7-4813-9816-A618FE19853E

Dispositivo Comienzo Final Sectores Tamaño Tipo

/dev/sdb1 2048 1953523711 1953521664 931,5G Sistema de ficheros de Linux

Disco /dev/sdc: 28,85 GiB, 30979129344 bytes, 60506112 sectores

Disk model: TransMemory

Unidades: sectores de 1 \* 512 = 512 bytes

Tamaño de sector (lógico/físico): 512 bytes / 512 bytes

Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes

Tipo de etiqueta de disco: dos

Identificador del disco: 0x94cb125e

Dispositivo Inicio Comienzo Final Sectores Tamaño Id Tipo

/dev/sdc1 8064 60506111 60498048 28,8G 7 HPFS/NTFS/exFAT

### Mostrar el nombre del fichero de la terminal conectada a la salida estándar y hacer una prueba para mandarle caracteres.

a08@a08-HP-ProDesk-600-G5-SFF:~$ echo hola > /dev/pts/0

hola

### Ejecutar el comando [w](https://francisconi.org/linux/comandos/nmap) (para obtener información de los terminales activos).

a08@a08-HP-ProDesk-600-G5-SFF:~$ w

13:11:45 up 1:25, 1 user, load average: 0,36, 0,46, 0,52

USUARIO TTY DESDE LOGIN@ IDLE JCPU PCPU WHAT

a08 tty2 tty2 11:46 1:25m 0.01s 0.00s /usr/libexec/gdm-wayland-session env GNOME\_SHELL\_SESSION\_MODE=ubuntu /usr/bin/gnome-session --session=ubuntu

a08@a08-HP-ProDesk-600-G5-SFF:~$ who

a08 tty2 2023-10-09 11:46 (tty2)

### Probar el comando df y comentar la salida:

DF (disk filesystem – Sistema de archivos de disco) nos proporciona información detallada del espacio usado en el disco. Si deseamos que dentro de la información desplegada este el tipo de sistema de archivos usaremos el parámetro –T y para que sea legible añadiremos h, quedando la sintáis de la siguiente manera:

# df

#df -Th

a08@a08-HP-ProDesk-600-G5-SFF:~$ df

S.ficheros bloques de 1K Usados Disponibles Uso% Montado en

tmpfs 1613936 2368 1611568 1% /run

/dev/sda2 459850824 110566192 325851980 26% /

tmpfs 8069664 672 8068992 1% /dev/shm

tmpfs 5120 4 5116 1% /run/lock

tmpfs 8069664 0 8069664 0% /run/qemu

/dev/sda1 523244 6216 517028 2% /boot/efi

tmpfs 1613932 2472 1611460 1% /run/user/1000

/dev/sdc1 30249020 127848 30121172 1% /media/a08/KIOXIA

a08@a08-HP-ProDesk-600-G5-SFF:~$ df -Th

S.ficheros Tipo Tamaño Usados Disp Uso% Montado en

tmpfs tmpfs 1,6G 2,4M 1,6G 1% /run

/dev/sda2 ext4 439G 106G 311G 26% /

tmpfs tmpfs 7,7G 672K 7,7G 1% /dev/shm

tmpfs tmpfs 5,0M 4,0K 5,0M 1% /run/lock

tmpfs tmpfs 7,7G 0 7,7G 0% /run/qemu

/dev/sda1 vfat 511M 6,1M 505M 2% /boot/efi

tmpfs tmpfs 1,6G 2,5M 1,6G 1% /run/user/1000

/dev/sdc1 ntfs3 29G 125M 29G 1% /media/a08/KIOXIA

### Listar información sistema de archivos Linux

LSBLK (list block devices) nos permite desplegar en pantalla toda la información del bloque de dispositivos que se encuentren disponibles. Agregamos el parámetro –f para listar el tipo de sistema de archivos ejecutando el siguiente comando. De esta forma conoceremos en detalle los diversos sistemas de archivos.

#lsblk -f

a08@a08-HP-ProDesk-600-G5-SFF:~$ lsblk -f

NAME FSTYPE FSVER LABEL UUID FSAVAIL FSUSE% MOUNTPOINTS

sda

├─sda1 vfat FAT32 0F02-33A6 504,9M 1% /boot/efi

└─sda2 ext4 1.0 cfcf4d74-d093-4d25-a419-3e4ebb05e857 310,8G 24% /var/snap/firefox/common/host-hunspell

/

sdb

└─sdb1 ext4 1.0 ec3f58be-a7b2-4c82-91f4-0194ef5ecb5e

sdc

└─sdc1 ntfs KIOXIA C07C9AA17C9A9230 28,7G 0% /media/a08/KIOXIA

sr0

### Comprobar el espacio en disco en Linux usando el comando

# du -h

a08@a08-HP-ProDesk-600-G5-SFF:~$ du -h

4,0K ./.gnupg/private-keys-v1.d

28K ./.gnupg

4,0K ./.mozilla/extensions

8,0K ./.mozilla

104K ./.dotnet/corefx/cryptography/crls

[…]

[…]

[…]

44K ./Escritorio/Nueva carpeta

646M ./Escritorio

80G .

# Capítulo 10. Seguridad

## Preguntas

### ¿Cuál de las siguientes características hace que una contraseña sea más segura?

a) Una contraseña corta

b) Una contraseña larga

c) Una contraseña basada en el nombre del usuario

d) Una contraseña que contiene solo letras minúsculas

### ¿Por qué es importante cambiar regularmente las contraseñas?

a) Para aumentar la complejidad de las contraseñas

b) Para tener menos posibilidades de bloqueo de la cuenta

c) Para hacer que las contraseñas sean más fáciles de recordar

d) No es necesario cambiar regularmente las contraseñas

### ¿Cuál de las siguientes afirmaciones es cierta con respecto a las contraseñas seguras?

a) Deben contener al menos un número y una letra en mayúscula

b) Deben ser iguales a la dirección de correo electrónico del usuario

c) Deben ser compartidas con amigos y colegas

d) No importa si se comparten con otros, siempre que sean fuertes

### ¿Cuál de las siguientes es una recomendación importante para crear contraseñas seguras?

a) Utilizar palabras comunes y fáciles de recordar

b) Utilizar la misma contraseña para todas las cuentas

c) Evitar el uso de caracteres especiales como símbolos y números

d) Utilizar una combinación de letras mayúsculas, minúsculas, números y símbolos

### .¿Cuál es un ejemplo de una contraseña segura?

a) "password123"

b) "miapellido"

c) "P@$$w0rd!"

d) "123456789"

### ¿Por qué es importante no compartir sus contraseñas con nadie?

a) Para mantener la privacidad y la seguridad de sus cuentas

b) Porque es una práctica común y segura compartirlas

c) Para facilitar el acceso de otros a su cuenta

d) No importa si se comparten, siempre que confíe en la persona con la que las comparte

### ¿Qué es un ataque de fuerza bruta en relación con las contraseñas?

a) Un ataque en el que un atacante adivina la contraseña correcta de inmediato

b) Un ataque en el que un atacante intenta todas las combinaciones posibles de contraseñas hasta encontrar la correcta

c) Un ataque en el que un atacante engaña al usuario para que revele su contraseña

d) Un ataque en el que un atacante intercepta el tráfico de red para obtener la contraseña del usuario

### ¿Cuál de las siguientes afirmaciones es cierta acerca de la autenticación de dos factores (2FA)?

a) Es menos segura que una contraseña única

b) Requiere el uso de dos contraseñas

c) Agrega una capa adicional de seguridad al requerir dos formas diferentes de autenticación

d) No es necesario para la mayoría de las cuentas en línea

### ¿Qué es el "salting" en el contexto de la seguridad de contraseñas?

a) Un proceso para eliminar las contraseñas débiles.

b) La adición de información adicional aleatoria a una contraseña antes de almacenarla.

c) Un método para cifrar contraseñas en tiempo real.

d) La técnica de hashing de contraseñas más segura.

### ¿Cuál es el propósito principal del "salting" en la seguridad de contraseñas?

a) Hacer que las contraseñas sean más fáciles de recordar.

b) Aumentar la longitud de las contraseñas.

c) Fortalecer la seguridad almacenando contraseñas de manera más segura.

d) Encriptar las contraseñas antes de almacenarlas.

### ¿Cómo afecta el "salting" a las contraseñas almacenadas?

a) Las contraseñas se vuelven más fáciles de adivinar.

b) Las contraseñas se vuelven más largas.

c) Cada contraseña almacenada es única debido a la adición de información aleatoria.

d) Las contraseñas se vuelven más simples.

### ¿Cuál de las siguientes afirmaciones es verdadera acerca de las contraseñas salteadas?

a) Son vulnerables a ataques de fuerza bruta.

b) Son menos seguras que las contraseñas sin sal.

c) Son más seguras porque evitan el uso de caracteres especiales.

d) Son más resistentes a ataques de diccionario y tablas de hash precalculadas (rainbow tables).

### ¿Cuál es una ventaja clave del uso de "salting" en la seguridad de contraseñas?

a) Simplifica el proceso de autenticación.

b) Permite el almacenamiento de contraseñas en texto claro.

c) Protege contra ataques de diccionario y ataques con tablas de hash precalculadas.

d) Reduce la complejidad de las contraseñas.

### ¿Qué información se utiliza típicamente como "sal" en el "salting" de contraseñas?

a) La contraseña del usuario.

b) La dirección de correo electrónico del usuario.

c) Un valor aleatorio único para cada usuario.

d) El nombre de usuario del usuario.

### ¿Qué es una característica importante de un "sal" efectivo?

a) Debe ser la misma para todos los usuarios.

b) Debe ser secreto y no almacenado junto con las contraseñas.

c) Debe ser igual al nombre de usuario del usuario.

d) Debe ser una cadena de caracteres alfanuméricos fija.

### ¿Cuál de las siguientes afirmaciones es cierta con respecto al "salting" de contraseñas?

a) Es una práctica obsoleta y no se utiliza en la actualidad.

b) Aumenta la velocidad de autenticación.

c) Es una técnica efectiva para proteger las contraseñas almacenadas, especialmente en caso de filtraciones de bases de datos.

d) Solo se aplica a contraseñas cortas.

### .¿Qué es el método de Lamport en seguridad informática?

a) Un método para almacenar contraseñas en texto claro.

b) Un algoritmo de cifrado simétrico.

c) Un método de autenticación basado en contraseñas de un solo uso.

d) Un algoritmo de compresión de datos.

### ¿Cuál es la característica principal de las contraseñas generadas por el método de Lamport?

a) Son contraseñas de texto claro.

b) Son contraseñas estáticas y permanentes.

c) Son contraseñas de un solo uso que cambian con cada autenticación.

d) Son contraseñas alfanuméricas.

### ¿Cómo se almacenan las contraseñas en el método de Lamport?

a) En un servidor centralizado.

b) En una base de datos en línea.

c) Como una lista de hashes de contraseñas.

d) En texto claro.

### ¿Cómo se genera una contraseña de un solo uso en el método de Lamport?

a) Mediante un algoritmo de cifrado asimétrico.

b) Seleccionando una contraseña de una lista predefinida.

c) Aplicando una función hash criptográfica a una contraseña principal.

d) Mediante un algoritmo de compresión de datos.

### ¿Cuál es una desventaja potencial del método de Lamport? INvalida

a) Requiere una gran cantidad de almacenamiento para las contraseñas.

b) Es vulnerable a ataques de diccionario.

c) Las contraseñas de un solo uso pueden ser difíciles de generar.

d) No es adecuado para autenticación en línea.

### ¿Cómo se verifica una contraseña de un solo uso en el método de Lamport?

a) Se compara con una lista de contraseñas almacenadas en texto claro.

b) Se aplica una función hash criptográfica y se compara con la contraseña almacenada.

c) Se envía al servidor central para su verificación.

d) Se cifra con una clave pública y se compara con la contraseña almacenada.

### ¿Qué hace que las contraseñas generadas por el método de Lamport sean seguras?

a) Su longitud, ya que son muy largas.

b) La complejidad de los caracteres utilizados en ellas.

c) El hecho de que son de un solo uso y cambian constantemente.

d) Su almacenamiento en un servidor centralizado.

¿Qué es la autenticación basada en lo que se tiene?

a) Un método de autenticación basado en conocimientos.

b) Un método de autenticación basado en una smartcard

c) Un método de autenticación basado en características físicas del usuario.

d) Un método de autenticación basado en la ubicación geográfica.

### ¿Cuál de las siguientes es un ejemplo de factor de autenticación basado en lo que se tiene?

a) Contraseña.

b) Tarjeta de acceso físico.

c) Huella dactilar.

d) Pregunta de seguridad.

### ¿Qué factor de autenticación se basa en la posesión de un objeto físico?

a) Factor de autenticación basado en conocimientos.

b) Factor de autenticación basado en características físicas.

c) Factor de autenticación basado en posesión.

d) Factor de autenticación basado en ubicación.

### ¿Qué tipo de factor de autenticación es más difícil de robar o duplicar?

a) Factor de autenticación basado en conocimientos.

b) Factor de autenticación basado en características físicas.

c) Factor de autenticación basado en posesión.

d) Factor de autenticación basado en ubicación.

### ¿Cuál es una ventaja de la autenticación basada en lo que se tiene?

a) Es fácil de recordar para los usuarios.

b) Puede proporcionar un nivel adicional de seguridad.

c) No requiere dispositivos adicionales.

d) Es vulnerable a ataques de fuerza bruta.

### ¿Qué factor de autenticación se basa en conocimientos previos del usuario?

a) Factor de autenticación basado en posesión.

b) Factor de autenticación basado en ubicación.

c) Factor de autenticación basado en características físicas.

d) Factor de autenticación basado en contraseñas

### ¿Cuál de las siguientes no es una característica biométrica comúnmente utilizada en sistemas de autenticación?

a) Huella dactilar.

b) Contraseña.

c) Reconocimiento facial.

d) Escaneo de retina.

### ¿Qué hace que la biometría sea una forma efectiva de autenticación?

a) La facilidad de robar o duplicar las características biométricas.

b) La falta de precisión en la identificación de las características biométricas.

c) La singularidad y la dificultad de falsificar las características biométricas.

d) La necesidad de dispositivos costosos.

### ¿Cuál de las siguientes afirmaciones es cierta con respecto a la autenticación por huella dactilar?

a) Es fácil de falsificar mediante impresiones en 3D.

b) Es una forma altamente precisa de autenticación biométrica.

c) Requiere que los usuarios proporcionen una contraseña junto con su huella dactilar.

d) No se utiliza en aplicaciones de seguridad debido a su falta de precisión.

### ¿Cuál es una consideración importante en la implementación de la autenticación por biometría?

a) La necesidad de recordar múltiples contraseñas.

b) La protección de los datos biométricos almacenados.

c) La facilidad de compartir características biométricas con otros usuarios.

d) La falta de precisión en la identificación biométrica.

### ¿Qué es un ataque de desbordamiento de búfer?

a) Un ataque que sobrecarga el búfer de impresión de una impresora.

b) Un ataque en el que un programa o proceso sobrepasa el límite de memoria asignado a un búfer y sobrescribe la memoria adyacente.

c) Un ataque en el que un hacker sobrecarga el almacenamiento en búfer de una base de datos.

d) Un ataque que causa la pérdida de datos en un servidor.

## Preguntas 2

### ¿Cuál de los siguientes NO es un factor de autenticación comúnmente utilizado?

a) Contraseña

b) Huella dactilar

c) Color favorito

d) Tarjeta de acceso físico

### ¿Qué factor de autenticación se basa en algo que el usuario sabe?

a) Biometría

b) Token de seguridad

c) Contraseña

d) Tarjeta de acceso

### ¿Qué factor de autenticación se basa en algo que el usuario posee?

a) Contraseña

b) PIN

c) Certificado digital

d) Tarjeta de acceso físico

### ¿Qué factor de autenticación utiliza características físicas únicas del usuario, como su rostro o su voz?

a) Contraseña

b) Token de seguridad

c) Biometría

d) Tarjeta de acceso físico

### ¿Para qué se utiliza comúnmente una contraseña de BIOS o UEFI en una computadora?

a) Para cifrar el disco duro

b) Para proteger el sistema de ficheros de ataque por desbordamiento

c) Para evitar que se realicen cambios en la configuración del sistema

d) Para autenticar al usuario cuando accede al sistema operativo

### ¿Dónde se almacena típicamente la contraseña de BIOS o UEFI en una computadora?

a) En el disco duro

b) En un chip de memoria flash en la placa madre

c) En la memoria cache próxima a la RAM

d) En un servidor de contraseñas del sistema operativo

### . ¿Cuál de las siguientes contraseñas se consideraría más débil desde el punto de vista de la seguridad?

a) "P@ssw0rd"

b) "Choc0late!"

c) "Tr0ub4dor&3"

d) "S3cur3P@ssw0rd!"

### ¿Cuál de estas contraseñas es más difícil de adivinar por fuerza bruta? (k=longitud de la contraseña y n=números de símbolos utilizados)

a) k=4; n=26

b) k=6; n=26

c) k=4; n=36

d) k=8; n=26

### ¿Qué hace que una contraseña sea más fuerte desde el punto de vista de la seguridad?

a) Su longitud y total de alfanuméricos con símbolos utilizados

b) Su facilidad de memorización del algoritmo

c) Su uso de caracteres alfabéticos solamente

d) Su relación con datos personales, como nombres o fechas de nacimiento

### ¿Cuál de las siguientes contraseñas se consideraría más fuerte?

a) "123456"

b) "P@$$w0rd!"

c) "C0ffeeL0v3rs#2023"

d) "MyName123"

### ¿Se debe utilizar una contraseña única para diferentes cuentas en línea?

a) Si, facilita recordar las contraseñas

b) Si, se tiene menos riesgo de suplantación de identidad

c) No, se corre un riesgo de que un ataque en una cuenta comprometa a las otras

d) Si, para mejor compartir la contraseña con amigos y familiares

### ¿Qué deberías evitar al crear una contraseña?

a) Usar tu nombre o apellido

b) Hacerla lo más larga posible

c) Usar una combinación de letras y números

d) Cambiarla cada vez que inicies sesión

### ¿Qué tipo de información se almacena comúnmente en el archivo `/etc/passwd`?

a) Contraseñas de usuario cifradas

b) Nombres de usuario y contraseñas

c) Nombres de usuario y otra información de cuenta

d) Historial de inicio de sesión

### ¿Dónde se encuentra generalmente el archivo `passwd` en sistemas Unix y Linux?

a) En la carpeta de inicio de cada usuario

b) En la carpeta `/etc`

c) En la carpeta `/var`

d) En la carpeta `/home`

### ¿Cuál es el propósito del archivo `/etc/passwd`?

a) Almacenar las contraseñas de usuario en texto claro y cifrado

b) Almacenar información de grupos de usuarios del sistema

c) Proporcionar información sobre las cuentas de usuario del sistema

d) Registrar el historial de inicio de sesión de los usuarios

### ¿Cuál de las siguientes afirmaciones es cierta acerca del archivo `/etc/shadow`?

a) Sustituye a /etc/passwd en computadores modernos

b) Contiene contraseñas cifradas de usuarios

c) Hay un archivo de esa clase por cada grupo de usuarios

d) Contiene información sobre grupos de usuarios

### ¿Qué es el hashing de contraseñas?

a) Almacenar contraseñas en texto claro en el segmento hash de la memoria

b) Cifrar contraseñas con una clave secreta que el sistema operativo guarda en cache

c) Convertir contraseñas en una cadena de caracteres aleatoria.

d) Aplicar una función criptográfica para convertir una contraseña en un valor hash.

### ¿Cuál es el propósito principal de utilizar el hashing de contraseñas?

a) Aumentar la longitud de las contraseñas

b) Facilitar la recuperación de contraseñas olvidadas.

c) Proteger las contraseñas de los ataques de fuerza bruta

d) Proteger las contraseñas almacenando versiones irreversibles y seguras.

### ¿Cuál es la ventaja de usar un "salt" (sal) en el hashing de contraseñas?

a) Reduce la longitud de las contraseñas

b) Protege contra ataques de fuerza bruta.

c) Facilita la recuperación de contraseñas olvidadas.

d) Añade un bit de control a las contraseñas.

### ¿Qué es el salting en el contexto de la seguridad de contraseñas?

a) Agregar sal al agua para cocinar contraseñas.

b) Agregar un valor aleatorio único a una contraseña antes de aplicar el hashing.

c) Generar una contraseña a partir de la contraseña del grupo de usuarios

d) Eliminar las contraseñas del fichero passwd y generar otra con hashing en shadow

### . ¿Cuál es el propósito principal de agregar sal a las contraseñas antes de hashearlas?

a) Hacer que las contraseñas sean más fáciles de recordar.

b) Aumentar la longitud de las contraseñas hasta el límite del desbordamiento

c) Prevenir ataques de fuerza bruta y tablas arcoíris.

d) Cambiar la contraseña actual por una generada por hashing.

### ¿Cuál de las siguientes afirmaciones es cierta acerca del salting?

a) El mismo valor de sal se utiliza para todas las contraseñas en una base de datos.

b) El valor de sal es secreto y se almacena en el kernel del sistema operativo.

c) El salting solo es necesario para contraseñas largas.

d) El salting agrega un valor único a cada contraseña antes de hashearla.

### ¿Cuál es una ventaja del salting en la seguridad de contraseñas?

a) Facilita la recuperación de contraseñas olvidadas.

b) Hace que las contraseñas sean más cortas.

c) Protege contra ataques de diccionario y tablas arcoíris.

d) Aumenta la velocidad de autenticación.

### ¿Qué tipo de sistema de autenticación es el método de Lamport?

a) Basado en contraseñas

b) Basado en tokens de seguridad

c) Basado en firmas digitales

d) Basado en biometría

### ¿Cómo se genera una contraseña Lamport?

a) A través de una tarjeta de acceso físico.

b) Se calcula aplicando aleatoriamente una función hash a una contraseña cifrada.

c) Se calcula aplicando recursivamente función heap a una contraseña en texto claro.

d) Se calcula aplicando reiteradamente una función hash a una contraseña en texto claro.

### Una técnica común de ingeniería social online en la que los atacantes envían correos electrónicos o mensajes que parecen provenir de fuentes legítimas, pero las víctimas son dirigidas a estos sitios falsos, donde se les solicita que ingresen sus contraseñas y otra información confidencial. ¿De qué tipo de ataque estamos hablando?

a) De fuerza bruta

b) De Fishing

c) De Vishing

d) De botnet

### A la acción de la figura se le suele llamar:

a) Shoulder fishing

b) Fishing

c) Vishing

d) Visoning

### ¿Qué es un ataque de desbordamiento de búfer?

a) Un ataque que sobrecarga el búfer de impresión de una impresora con excesivos trabajos

b) Un ataque en el que un proceso sobrepasa el límite de memoria asignado a un búfer y sobrescribe la memoria adyacente.

c) Un ataque en el que un hacker sobrecarga el almacenamiento en búfer de una base de datos.

d) Un ataque que causa la pérdida de datos en un servidor por desbordamiento de la cache L3

### ¿Cuál es el objetivo principal de un ataque de desbordamiento de búfer?

a) Robar contraseñas del usuario

b) El desbordamiento intencional del espacio de una variable para inyectar código maligno

c) Aumentar la velocidad de procesamiento de una aplicación para llevarla fuera de control del kernel del sistema operativo

d) Aumentar el almacenamiento el búfer de la ALU en un sistema multicore

### ¿Qué tipo de vulnerabilidad de software es explotada típicamente en un ataque de desbordamiento de búfer?

a) Vulnerabilidad de seguridad física mediante tarjetas inteligentes

b) Vulnerabilidad de ingeniería social a través de botnet

c) Vulnerabilidad de software de programación deficiente.

d) Vulnerabilidad de hardware en cuanto se produce Rowhammer

### ¿Cuál es un objetivo común de un ataque exitoso de desbordamiento de búfer?

a) la disminución del rendimiento del sistema en beneficio del proceso principal

b) La eliminación de malware.

c) La ejecución de código malicioso.

d) La mejora de la seguridad de la red.

### ¿Qué es un "Canario de Pila" en el contexto de la seguridad informática?

a) Es un desbordamiento del buffer no controlado por el sistema

b) Una técnica de ataque en la que se coloca un valor aleatorio en la pila y produce desbordamiento de buffer

c) Un valor aleatorio colocado en la pila de una función para detectar desbordamientos de búfer.

d) Un tipo de ataque de denegación de servicio.

### ¿Cuál es el propósito principal de un Canario de Pila?

a) Proteger la información encriptada en la pila.

b) Evitar que los atacantes accedan al heap del proceso en curso usando la técnica de Rowhammer

c) Detectar y prevenir desbordamientos de búfer y ataques de ejecución de código malicioso.

d) Detectar y prevenir los ataques mediante hashing de contraseñas controladas por diccionario.

### ¿Cómo funciona un Canario de Pila?

a) Encripta todos los datos en la pila para evitar cualquier ataque de tipo desbordamiento de contraseña

b) Registra las actividades del sistema operativo para calcular una estadística de los ataques recibidos.

c) Coloca un valor aleatorio en la pila antes de la dirección de retorno de una función y verifica su integridad antes de que la función retorne.

d) Desactiva la caché de la CPU.

### . ¿Cuál es uno de los problemas a resolver en la implementación de Canarios de Pila?

a) Requiere acceso físico al hardware de la computadora.

b) Puede causar conflictos de memoria en el sistema operativo.

c) Puede tener un impacto en el rendimiento de las aplicaciones.

d) Solo es efectivo contra amenazas de red.

### ¿En qué tipo de programas y sistemas operativos es común encontrar la implementación de Canarios de Pila?

a) Solo en sistemas macOS.

b) Solo en sistemas Unix/Linux.

c) En sistemas Windows y otros sistemas operativos.

d) Solo en servidores web.

### ¿Qué sucede si un atacante intenta desbordar un búfer en un programa que utiliza un Canario de Pila?

a) El atacante obtiene acceso a la memoria RAM y limpia las memorias caché

b) El Canario de Pila detecta el desbordamiento y detiene la ejecución del programa.

c) El Canario de Pila borra todas las variables dinámicas de la pila

d) El programa se bloquea y muestra un mensaje de error.

### ¿Cuál es uno de los beneficios clave de los Canarios de Pila en términos de seguridad?

a) Protege contra todos los tipos de amenazas informáticas.

b) Aumenta la velocidad de ejecución de programas.

c) Dificulta la explotación de vulnerabilidades de seguridad que facilitan desbordamientos de búfer.

d) Dificulta la explotación de vulnerabilidades de seguridad que dificultan el hashing de contraseñas

### . ¿Qué significa la sigla DEP en el contexto de seguridad informática?

a) Data Entry Point.

b) Digital Encryption Protocol.

c) Data Execution Prevention.

d) Dynamic Encryption Process.

### ¿Cuál es el objetivo principal de DEP?

a) Acelerar la velocidad de ejecución de programas.

b) Proteger datos almacenados en el disco duro.

c) Evitar que se ejecute código malicioso en regiones de memoria marcadas como no ejecutables.

d) Evitar que se ejecute código malicioso en regiones de memoria marcadas como ejecutables mediante la alteración del contador de programa

### ¿Cómo funciona DEP para prevenir la ejecución de código malicioso?

a) Desactiva por completo la memoria RAM.

b) Marca ciertas regiones de memoria como no ejecutables y evita que el procesador ejecute código en esas áreas.

c) Requiere que todos los programas se ejecuten en modo kernel ya que es más seguro que el modo usuario

d) Cifra y descifra todo el contenido de la memoria en tiempo real

### ¿En qué tipo de sistemas operativos es común encontrar la función DEP?

a) Solo en sistemas macOS.

b) Solo en sistemas Unix/Linux.

c) En sistemas Windows y otros sistemas operativos.

d) Solo en sistemas móviles.

### ¿Cuál de las siguientes afirmaciones es cierta sobre DEP?

a) DEP es una técnica de ataque con código malicioso en regiones de memoria marcadas como no ejecutables.

b) DEP solo protege contra virus y malware conocidos.

c) DEP es efectivo para prevenir la ejecución de código malicioso en regiones de memoria marcadas como no ejecutables.

d) DEP solo es necesario en servidores, no en computadoras personales.

### . ¿Cuál es uno de los beneficios clave de DEP en términos de seguridad?

a) Aumenta la velocidad de ejecución de programas porque la CPU trabaja en forma especulativa

b) Dificulta la explotación de vulnerabilidades de seguridad que permiten descifrar las contraseñas.

c) Dificulta la explotación de vulnerabilidades de seguridad que permiten la ejecución de código malicioso.

d) Protege contra ataques de phishing cuando el ordenador está conectado a WIFI

### ¿Qué significa la sigla ROP en el contexto de seguridad informática?

a) Ransomware Oriented Programming.

b) Return-Oriented Programming.

c) Randomized Operating Procedures.

d) Read-Only Protection.

### ¿Cuál es el objetivo principal de un ataque de ROP?

a) Robar contraseñas en texto claro.

b) Explotar vulnerabilidades en el cifrado de datos.

c) Ejecutar código malicioso utilizando fragmentos existentes de código ejecutable en el programa.

d) Atacar sistemas operativos.

### ¿Qué hace que ROP sea una técnica de ataque efectiva?

a) Utiliza solo código escrito por el atacante.

b) Utiliza solo lenguaje ensamblador.

c) Requiere acceso físico a la computadora objetivo.

d) Utiliza fragmentos de código legítimo en el programa.

### . ¿Cómo funciona un ataque de ROP?

a) Inyecta código malicioso en el programa objetivo.

b) Modifica el sistema de archivos del sistema operativo.

c) Utiliza instrucciones de retorno (ret) para saltar a fragmentos de código existentes en el programa y ejecutar operaciones maliciosas.

d) Cifra los datos en el disco duro.

### ¿Cuál es uno de los desafíos clave para un atacante en un ataque de ROP?

a) Encontrar una vulnerabilidad de día cero en el programa objetivo.

b) Tener acceso físico al hardware de la computadora objetivo.

c) Conocer el lenguaje ensamblador a nivel experto.

d) Escanear la red en busca de vulnerabilidades.

### . ¿En qué tipo de programas y sistemas operativos es más común encontrar ataques de ROP?

a) Solo en programas antivirus.

b) En una amplia variedad de programas y sistemas operativos.

c) Solo en sistemas macOS.

d) Solo en sistemas Unix/Linux.

### ¿Qué medida de seguridad puede ayudar a mitigar los ataques de ROP?

a) Utilizar un antivirus actualizado.

b) No permitir que los programas ejecuten código en absoluto.

c) Implementar técnicas de protección de memoria, como DEP (Data Execution Prevention) o ASLR (Address Space Layout Randomization).

d) Utilizar un firewall de red.

. ¿Qué significa la sigla ASLR en el contexto de seguridad informática?

a) Address Space Layout Randomization.

b) Advanced Security and Log Retrieval.

c) Application Security and Logging Routine.

d) All System Logs Randomized.

### . ¿Cuál es el propósito principal de ASLR?

a) Aumentar la velocidad de ejecución de programas.

b) Proteger la dirección IP de un sistema.

c) Dificultar la explotación de vulnerabilidades de seguridad al aleatorizar la ubicación de la memoria.

d) Encriptar el tráfico de red.

### ¿Cómo funciona ASLR en un sistema operativo?

a) Cambiando aleatoriamente la dirección IP del sistema.

b) Cambiando aleatoriamente la dirección de la puerta de enlace predeterminada.

c) Cambiando aleatoriamente la ubicación de la memoria de las aplicaciones y bibliotecas en el espacio de direcciones virtual.

d) Cambiando aleatoriamente la contraseña del administrador.

### ¿Cuál es uno de los beneficios clave de ASLR en términos de seguridad?

a) Acelerar la velocidad de carga de programas.

b) Facilitar la administración de contraseñas.

c) Dificultar la explotación de vulnerabilidades de desbordamiento de búfer y otros ataques.

d) Proteger los datos almacenados en la nube.

### ¿En qué tipo de sistemas operativos se encuentra comúnmente la implementación de ASLR?

a) Solo en sistemas Windows.

b) Solo en sistemas macOS.

c) En sistemas Unix/Linux, Windows y otros sistemas operativos.

d) En sistemas operativos móviles, pero no en sistemas de escritorio.

### ¿Qué tipo de ataque de seguridad puede dificultarse significativamente mediante la implementación de ASLR?

a) Ataques de fuerza bruta.

b) Ataques de denegación de servicio.

c) Ataques de phishing.

d) Ataques que aprovechan vulnerabilidades de desbordamiento de búfer.

### ¿Cuál es uno de los posibles desafíos al implementar ASLR?

a) Aumento de la velocidad de ejecución de programas.

b) Posible incompatibilidad con aplicaciones antiguas que dependen de direcciones de memoria fijas.

c) Mayor consumo de memoria RAM.

d) Dificultad para administrar contraseñas.

### . ¿Qué es un ataque de detección de canal lateral?

a) Un ataque que utiliza una contraseña débil para acceder a un sistema.

b) Un ataque que aprovecha las vulnerabilidades en el cifrado de datos.

c) Un ataque que explora las fugas de información involuntarias, como el consumo de energía o el tiempo de acceso a la memoria, para deducir información confidencial.

d) Un ataque que utiliza fuerza bruta para romper contraseñas.

### ¿Cuál es el objetivo principal de un ataque de detección de canal lateral?

a) Exponer claves públicas.

b) Exponer claves privadas.

c) Proteger datos encriptados.

d) Aumentar el rendimiento de un sistema.

### ¿Qué tipo de información se puede obtener a través de un ataque de detección de canal lateral?

a) Contraseñas en texto claro.

b) Datos encriptados.

c) Información sobre patrones de consumo de energía, tiempo de acceso a la memoria, u otros canales secundarios.

d) Información sobre la ubicación geográfica de un usuario.

### ¿Cuál de las siguientes afirmaciones es cierta sobre la mitigación de ataques de detección de canal lateral?

a) No existen medidas efectivas para mitigar estos ataques.

b) Desactivar por completo la caché de la CPU es la única solución.

c) Se pueden aplicar contramedidas como el enmascaramiento de datos, la aleatorización de tiempo o el ruido de potencia.

d) Estos ataques solo afectan a servidores y centros de datos.

### . ¿Cuál es uno de los desafíos de la mitigación de ataques de detección de canal lateral?

a) Requiere una gran cantidad de recursos de hardware.

b) Es una técnica que no se puede bloquear.

c) No afecta el rendimiento del sistema.

d) No es una amenaza real en la actualidad.

### ¿Qué es un ataque de detección de canal lateral en el contexto del algoritmo RSA?

a) Un ataque que utiliza contraseñas débiles para descifrar mensajes encriptados con RSA.

b) Un ataque que se enfoca en interceptar el tráfico de red.

c) Un ataque que explora las fugas de información, como el consumo de energía o el tiempo de ejecución, para deducir claves privadas.

d) Un ataque que apunta a descifrar mensajes encriptados utilizando la fuerza bruta.

### ¿Cuál es el objetivo principal de un ataque de detección de canal lateral en el contexto de RSA?

a) Exponer claves públicas.

b) Exponer claves privadas.

c) Proteger datos encriptados.

d) Acelerar el procesamiento de datos.

### ¿Qué tipo de información se puede obtener a través de un ataque de detección de canal lateral en RSA?

a) Contraseñas en texto claro.

b) Datos encriptados.

c) Información sobre patrones de consumo de energía o tiempo de ejecución.

d) Direcciones IP de los servidores.

### . ¿Cuál es uno de los canales secundarios utilizados en un ataque de detección de canal lateral en RSA?

a) Canal de radio.

b) Canal de red.

c) Canal de tiempo.

d) Canal de almacenamiento.

### ¿Cuál de las siguientes afirmaciones es cierta sobre la mitigación de ataques de detección de canal lateral en RSA?

a) No hay medidas efectivas para mitigar estos ataques.

b) La desactivación de la caché de la CPU es la única solución.

c) Se pueden aplicar contramedidas como el enmascaramiento de datos o el ruido de potencia.

d) Estos ataques solo afectan a servidores y centros de datos.

### ¿Cuál es uno de los desafíos de la mitigación de ataques de detección de canal lateral en RSA?

a) Requiere una gran cantidad de recursos de hardware.

b) Es una técnica que no se puede bloquear.

c) No afecta el rendimiento del sistema.

d) Es una amenaza poco común.

### ¿Qué son los ataques de canal lateral a la caché?

a) Ataques que se dirigen a la memoria RAM de un sistema.

b) Ataques que aprovechan las debilidades en el cifrado de datos.

c) Ataques que explotan las filtraciones de información a través de canales secundarios, como el tiempo de acceso a la caché.

d) Ataques que utilizan la fuerza bruta para descifrar contraseñas.

### 2. ¿Qué tipo de información se puede obtener a través de un ataque de canal lateral a la caché?

a) Contraseñas en texto claro.

b) Información sobre la estructura interna de la CPU.

c) Patrones de acceso a la caché, revelando posibles claves o datos confidenciales.

d) Información sobre la ubicación geográfica del usuario.

### 3. ¿Cuál es uno de los canales secundarios utilizados en ataques de canal lateral a la caché?

a) Canal de radio.

b) Canal de red.

c) Canal de tiempo.

d) Canal de energía eléctrica.

### . ¿Qué es la "sincronización del reloj" en el contexto de un ataque de canal lateral a la caché?

a) El proceso de ajustar la hora en un dispositivo.

b) El proceso de sincronizar el reloj de la CPU con el reloj del sistema.

c) El uso de información de tiempo de acceso a la caché para inferir datos secretos.

d) El proceso de medir el rendimiento del procesador.

### . ¿Cuál es uno de los ejemplos más conocidos de ataque de canal lateral a la caché?

a) Ataque de fuerza bruta.

b) Ataque de diccionario.

c) Ataque Spectre.

d) Ataque de denegación de servicio.

### . ¿Cómo pueden mitigarse los ataques de canal lateral a la caché?

a) Utilizando un firewall.

b) Aplicando medidas de seguridad en el software y el hardware para minimizar las filtraciones de información.

c) Desactivando la caché en la CPU.

d) Aumentando la velocidad del reloj de la CPU.

### .¿Cuál es uno de los desafíos de la mitigación de ataques de canal lateral a la caché?

a) Requiere una gran cantidad de recursos de hardware.

b) Es ineficaz y no tiene impacto en la seguridad.

c) No hay medidas efectivas para mitigar estos ataques.

d) Solo se pueden mitigar mediante medidas de seguridad física.

### ¿En qué tipo de entornos o escenarios es más probable que ocurran ataques de canal lateral a la caché?

a) Entornos de redes sociales.

b) Entornos de servidor y centros de datos.

c) Entornos de juegos en línea.

d) Entornos de edición de video.

### ¿Qué es la "vulnerabilidad Meltdown" en relación con los ataques de canal lateral a la caché?

a) Una técnica de mitigación de ataques de canal lateral.

b) Un ataque que explota debilidades en el cifrado de datos.

c) Un tipo de ataque de canal lateral a la caché que afecta principalmente a procesadores Intel.

d) Un método de sincronización de relojes en la CPU.

### ¿Qué es una "cola de almacenamiento" en una CPU y cómo puede estar relacionada con los ataques de canal lateral a la caché?

a) Una lista de instrucciones en espera de ejecución; puede ser un objetivo para ataques de canal lateral.

b) Un tipo de memoria RAM utilizada para almacenar caché de datos.

c) Un componente de hardware utilizado para proteger la CPU de ataques.

d) Una unidad de procesamiento dedicada a ejecutar instrucciones especulativas.

### ¿Cuál es uno de los métodos utilizados en la mitigación de ataques de canal lateral a la caché?

a) Aumentar la velocidad del reloj de la CPU.

b) Utilizar software de seguridad para detectar y bloquear ataques.

c) Desactivar completamente la caché de la CPU.

d) Implementar técnicas de sincronización de reloj más precisas.

### . ¿Cuál es el objetivo principal de un ataque de canal lateral a la caché?

a) Robar contraseñas en texto claro.

b) Obtener información sobre los patrones de acceso a la caché para deducir datos confidenciales.

c) Interceptar tráfico de red.

d) Explotar debilidades en el cifrado de datos en la RAM.

### ¿Cuál es una de las ventajas de la caché de la CPU que también puede ser aprovechada por los atacantes en ataques de canal lateral?

a) Acelera el procesamiento de datos sin ningún riesgo.

b) Almacena datos confidenciales de forma segura.

c) Permite el acceso más rápido a datos comunes, lo que puede revelar patrones de acceso.

d) Protege la CPU de cualquier tipo de ataque.

### ¿Cuál es uno de los efectos secundarios no deseados de las medidas de mitigación de ataques de canal lateral a la caché?

a) Aumento del rendimiento de la CPU.

b) Reducción del consumo de energía.

c) Pérdida de rendimiento en aplicaciones específicas debido a la desactivación de ciertas optimizaciones.

d) Aumento de la velocidad del reloj de la CPU.

### ¿Cuál es una técnica comúnmente utilizada en ataques de canal lateral a la caché para deducir información confidencial?

a) Ataque de fuerza bruta.

b) Ataque de diccionario.

c) Monitoreo de tiempos de acceso a la caché.

d) Ataque de negación de servicio.

### ¿Qué es el ataque de canal lateral al caché conocido como "Prime+Probe"?

a) Un ataque que utiliza patrones de acceso a la caché para deducir información confidencial.

b) Un ataque que desactiva completamente la caché de la CPU.

c) Un ataque que explota vulnerabilidades en el sistema operativo.

d) Un ataque que intercepta el tráfico de red.

### ¿Cuál es la fase "Prime" en el ataque Prime+Probe?

a) La fase de interceptación de datos.

b) La fase de preparación, donde se carga la caché con datos específicos.

c) La fase de extracción de datos confidenciales.

d) La fase de cifrado de datos.

### ¿Qué papel desempeña la fase "Probe" en el ataque Prime+Probe?

a) La fase de preparación de datos.

b) La fase de almacenamiento de datos en la caché.

c) La fase de monitoreo de los tiempos de acceso a la caché para deducir información confidencial.

d) La fase de encriptación de datos.

### . ¿Cuál es el objetivo principal del ataque Prime+Probe?

a) Robar contraseñas en texto claro.

b) Desactivar la caché de la CPU.

c) Obtener información sobre los patrones de acceso a la caché para deducir datos confidenciales.

d) Explotar debilidades en el cifrado de datos en la RAM.

### ¿Cuál de las siguientes afirmaciones es cierta sobre los ataques de canal lateral a la caché en general?

a) Son fácilmente detectables y bloqueables.

b) No tienen impacto en la seguridad de la información.

c) Pueden ser utilizados para revelar patrones de acceso a la caché y deducir datos confidenciales.

d) Solo afectan a las aplicaciones de oficina.

### ¿Cuál es uno de los desafíos de mitigar el ataque Prime+Probe?

a) Requiere una gran cantidad de recursos de hardware.

b) Es una técnica que no se puede bloquear.

c) No tiene efecto en el rendimiento del sistema.

d) No afecta a los servidores y centros de datos.

### ¿Qué es la CPU especulativa?

a) Una CPU que funciona a velocidades más lentas que las CPU convencionales.

b) Una CPU que realiza cálculos de manera más precisa que otras CPUs.

c) Una CPU que anticipa y ejecuta instrucciones antes de saber si son necesarias.

d) Una CPU que está diseñada exclusivamente para juegos.

### . ¿Cuál es el propósito principal de la ejecución especulativa en una CPU?

a) Aumentar la velocidad del reloj de la CPU.

b) Acelerar la velocidad de procesamiento de gráficos.

c) Reducir el consumo de energía.

d) Mejorar el rendimiento general al predecir y ejecutar instrucciones.

### ¿Qué es un "branch prediction" en una CPU especulativa?

a) Una técnica para prevenir ataques de seguridad en la CPU.

b) Una técnica para anticipar el resultado de las instrucciones de salto condicional.

c) Una técnica para aumentar el tamaño de la caché de instrucciones.

d) Una técnica para dividir la CPU en múltiples núcleos.

### ¿Cuál es uno de los posibles desafíos de la CPU especulativa relacionado con la seguridad?

a) La ejecución especulativa puede aumentar el rendimiento, pero no afecta la seguridad.

b) Vulnerabilidades de seguridad como Spectre y Meltdown que pueden exponer datos confidenciales.

c) Falta de compatibilidad con software existente.

d) Mayor consumo de energía en comparación con las CPU convencionales.

### .¿En qué tipo de aplicaciones o escenarios es más beneficioso el uso de CPU especulativa?

a) Aplicaciones de oficina como procesamiento de texto y hojas de cálculo.

b) Aplicaciones de edición de video y diseño gráfico.

c) Aplicaciones de inteligencia artificial y aprendizaje profundo.

d) Aplicaciones de navegación web y correo electrónico.

### . ¿Cómo afecta la ejecución especulativa al rendimiento general de la CPU?

a) No tiene ningún efecto en el rendimiento.

b) Puede aumentar significativamente el rendimiento al ejecutar instrucciones de manera anticipada.

c) Disminuye el rendimiento al ralentizar la CPU.

d) Solo afecta el rendimiento de la GPU.

### ¿Cuál es un beneficio clave de la CPU especulativa en términos de rendimiento?

a) Aumenta la velocidad del reloj de la CPU.

b) Reduce la necesidad de almacenar caché.

c) Permite la ejecución de instrucciones fuera de orden.

d) Acelera la ejecución de instrucciones al predecir y anticiparlas.

## Preguntas adicionales sobre CPU especulativa:

### ¿Cuál es uno de los principales riesgos de seguridad asociados con la CPU especulativa?

a) Consumo excesivo de energía.

b) Rendimiento insuficiente.

c) Vulnerabilidades que podrían exponer datos confidenciales.

d) Incompatibilidad con software antiguo.

### . ¿Cuál es el propósito de la "ejecución fuera de orden" en una CPU especulativa?

a) Realizar cálculos de manera más precisa.

b) Ejecutar instrucciones en el orden en que se presentan en el programa.

c) Acelerar el procesamiento al ejecutar instrucciones de manera anticipada si es posible.

d) Reducir el tamaño de la caché de datos.

### . ¿Cuál es una de las ventajas de la CPU especulativa en términos de rendimiento de aplicaciones de cómputo intensivo?

a) Mayor consumo de energía.

b) Mayor compatibilidad con software antiguo.

c) Capacidad para ejecutar aplicaciones de manera más eficiente.

d) Reducción de la cantidad de RAM necesaria.

### . ¿Cuál es una de las desventajas de la ejecución especulativa en términos de consumo de energía?

a) Reduce significativamente el consumo de energía.

b) Aumenta el consumo de energía debido a la aceleración de instrucciones.

c) No tiene ningún efecto en el consumo de energía.

d) Solo afecta el consumo de energía de la GPU.

### . ¿Qué tipo de instrucciones en un programa de computadora pueden beneficiarse más de la ejecución especulativa?

a) Instrucciones de entrada/salida.

b) Instrucciones de cálculo intensivo.

c) Instrucciones de espera.

d) Instrucciones de RAM.

### . ¿Cómo puede mitigarse el riesgo de vulnerabilidades de seguridad asociadas con la CPU especulativa?

a) Apagar la CPU cuando no se está utilizando.

b) Aplicar parches de seguridad y actualizaciones del sistema operativo.

c) No utilizar CPUs especulativas en absoluto.

d) Utilizar una mayor cantidad de RAM.