Universidad Industrial de Santander

Investigación02 (Taller02 asignado el 31/03/2023)

Presentado por:

Manuel De Angel

Cod. De estudiante 2192510

Presentado a:

Pablo J. Rojas

Sistemas Operacionales

Cod. De asignatura 22972

1/04/2023

1. **En entornos compartidos, los usuarios comparten el sistema. Esto puede dar lugar a varios problemas de seguridad.**

◦ **Mencione dos problemas y explíquelos.**

Algunos de los problemas más comunes incluyen:

Acceso no autorizado: Si los usuarios no protegen adecuadamente sus cuentas con contraseñas fuertes y únicas, otro usuario malintencionado podría acceder a sus datos personales o confidenciales.

Vulnerabilidades en software compartido: Si varios usuarios utilizan el mismo software, cualquier vulnerabilidad de seguridad que afecte a ese software puede ser explotada por un atacante para comprometer todo el sistema compartido.

◦ **Es posible asegurar el mismo grado de seguridad en un sistema compartido que en un sistema dedicado. Explique su respuesta.**

En términos generales, no es posible asegurar el mismo grado de seguridad en un sistema compartido que en un sistema dedicado. Un sistema dedicado es aquel que está diseñado para una tarea específica y se utiliza exclusivamente para esa tarea. Esto significa que los recursos de hardware y software del sistema están optimizados para cumplir con los requisitos específicos de esa tarea y no hay otros procesos que puedan interferir o comprometer la seguridad del sistema. En un sistema dedicado, el nivel de seguridad se puede configurar y personalizar en función de las necesidades específicas del usuario, lo que puede hacer que el sistema sea más seguro; por otro lado, en un sistema compartido, varias personas o procesos pueden acceder a los mismos recursos de hardware y software al mismo tiempo. Esto puede aumentar el riesgo de que se produzcan errores de seguridad, como la introducción de malware o la exposición de datos confidenciales. Además, como no se puede controlar quién más está utilizando el sistema y cómo lo están utilizando, es difícil garantizar que todos los usuarios cumplan con las medidas de seguridad adecuadas. Por lo tanto, aunque se pueden implementar medidas de seguridad para reducir los riesgos en un sistema compartido, siempre existe un mayor riesgo de compromiso de seguridad en comparación con un sistema dedicado.

2. **Un problema común en los OS es la utilización de recursos. Enumere los recursos que deben gestionarse en las siguientes maquina (explique por qué):**

◦ Sistemas embebidos: carecen de seguridad ya que son para amplio acceso por lo que se ve comprometida la información confiada en ellos en ciertas ocasiones que se dé un uso malintencionado.

◦ Mainframe: Dependiendo del tipo de máquina, puede haber recursos de hardware específicos que deben gestionarse adecuadamente, como los dispositivos de entrada/salida, las tarjetas gráficas, etc.

◦ Workstation: el precio a invertir en la Workstation del grupo de trabajo o de la compañía ya que este es un equipo caro con características importantes para el desarrollo del proyecto asi como suministrar máxima potencia a los procesos realizados.

◦ Server: Hay que tener en cuenta el costo energético que este ha de llevar en ka cuenta de la compañía.

◦ Mobile: muchas veces los sistemas operativos que se tienen en la empresa son in-situ, es decir que se trabaja desde la empresa y el teletrabajo es por medio de conexión remota hacia los dispositivos de empleados, pero esto genera un riesgo de seguridad a leak-data ya que no se pueden adecuar al 100% medidas que permitan proteger los datos en red o en accesos no autorizados, por lo que si son mobiles en su totalidad la encriptación será total.

3. **Caracterice dos casos de uso para implementar un OS para servidor y PC:**

Un sistema operativo para servidor y un sistema operativo para PC son diferentes en términos de su diseño y funcionalidad. A continuación, se caracterizan dos casos de uso típicos para implementar cada uno de ellos:

Sistema operativo para servidor: Un sistema operativo para servidor se utiliza para gestionar y controlar los recursos en una red de servidores. Estos sistemas operativos se utilizan en entornos empresariales, centros de datos y otros entornos donde se requiere una alta disponibilidad y escalabilidad.

Servicios en línea: Un caso de uso común para un sistema operativo de servidor es alojar servicios en línea, como sitios web, aplicaciones web y servicios de correo electrónico. Un sistema operativo para servidor debe ser capaz de gestionar múltiples solicitudes de usuarios y garantizar una alta disponibilidad de los servicios alojados.

Bases de datos: Otro caso de uso común para un sistema operativo de servidor es alojar bases de datos. Un sistema operativo de servidor debe ser capaz de gestionar grandes cantidades de datos y garantizar su integridad y seguridad.

Sistema operativo para PC: Un sistema operativo para PC se utiliza para gestionar y controlar los recursos en una computadora personal. Estos sistemas operativos se utilizan en entornos domésticos y empresariales donde se requiere una interfaz de usuario intuitiva y una amplia gama de aplicaciones de software.

Productividad personal: Un caso de uso común para un sistema operativo de PC es para la productividad personal, como el uso de procesadores de texto, hojas de cálculo y

programas de presentación. Un sistema operativo de PC debe ser capaz de proporcionar una interfaz de usuario intuitiva y una amplia gama de herramientas de productividad.

Juegos y entretenimiento: Otro caso de uso común para un sistema operativo de PC es para juegos y entretenimiento. Un sistema operativo de PC debe ser capaz de proporcionar soporte para gráficos de alta calidad, audio y video, así como una amplia gama de juegos y aplicaciones de entretenimiento.

4. **Compare las diferencias entre multiprocesamiento simétrico y asimétrico**

5. **Enumere los requerimientos para que dos máquinas se junten en un cluster y provean un servicio de alta disponibilidad (HA):**

1.1 Redundancia de hardware: Cada máquina debe tener hardware redundante para garantizar que, si un componente falla, el sistema continúe funcionando sin interrupciones.

2.1 Conexión de red de alta velocidad: Las máquinas deben estar conectadas mediante una red de alta velocidad para garantizar una comunicación rápida y confiable entre ellas.

3.1 Software de clustering: Las máquinas deben tener software de clustering instalado y configurado correctamente. El software de clustering permite que las máquinas compartan recursos, como almacenamiento y memoria, y coordinen sus actividades para garantizar la alta disponibilidad del servicio.

4.1 Monitoreo y gestión centralizados: El clúster debe tener un sistema centralizado para monitorear y gestionar el estado de cada máquina y el servicio en general. Esto permite detectar y solucionar rápidamente cualquier problema que surja.

5.1 Políticas de fail over: El clúster debe tener políticas de fail over configuradas para garantizar que, si una máquina falla, otra máquina asuma automáticamente sus funciones. Esto garantiza que el servicio siga funcionando sin interrupciones.

6.1 Sistema de respaldo: El clúster debe tener un sistema de respaldo configurado para garantizar la recuperación de datos en caso de falla del hardware o software.

7.1 Fuentes de alimentación ininterrumpidas (UPS): Cada máquina del clúster debe tener una UPS para garantizar que, si hay una falla en la alimentación eléctrica, el sistema continúe funcionando sin interrupciones.

6. **Compare las diferencias entre una excepción y una interrupción:**

Excepción	interrupción
Es un evento interno que se produce dentro del programa durante su ejecución, como una operación de división por cero, acceso a una ubicación de memoria no válida o una violación de seguridad.	Es un evento externo que proviene de dispositivos de hardware, como temporizadores, dispositivos de entrada/salida o controladores de dispositivos.
Las excepciones son detectadas por el procesador durante la ejecución del programa y generan una interrupción de software.	Puede ser generada por el hardware o por una señal externa, y puede ocurrir en cualquier momento durante la ejecución del programa.
El procesador salta a un código específico de manejo de excepciones en el programa y ejecuta un procedimiento para manejar la excepción.	El procesador detiene la ejecución del programa actual y salta a un código específico que maneja la interrupción.
Después de manejar la excepción, el procesador regresa al programa principal y continúa su ejecución.	Después de manejar la interrupción, el procesador regresa al programa principal y continúa su ejecución.

la principal diferencia entre una interrupción y una excepción es que las interrupciones son eventos externos generados por dispositivos de hardware, mientras que las excepciones son eventos internos que se producen dentro del programa durante su ejecución. Además, el manejo de una interrupción implica saltar a un código específico que maneja la interrupción, mientras que el manejo de una excepción implica saltar a un código específico de manejo de excepciones en el programa.

7. **El DMA (acceso directo a memoria) se usa en dispositivos I/O para evitar uso innecesario de la CPU.**

◦ **¿Como interactuá la CPU con el dispositivo para coordinar la transferencia?**

◦ **¿Como sabe la CPU que las operaciones de memoria se han completado?**

R/= Cuando un dispositivo de entrada/salida (I/O) necesita transferir datos a o desde la memoria del sistema, puede utilizar el DMA (acceso directo a memoria) para realizar la transferencia sin la intervención de la CPU. Para coordinar la transferencia, la CPU y el dispositivo I/O utilizan una técnica llamada programación de DMA.

En la programación de DMA, la CPU configura el controlador de DMA del sistema con información sobre la transferencia, incluyendo la dirección de inicio y el tamaño de la transferencia, así como la dirección de memoria del búfer de datos en el que se almacenarán los datos transferidos. Una vez que se ha configurado el controlador de DMA, el dispositivo I/O activa la transferencia y el controlador de DMA toma el control de la transferencia de datos.

Mientras el controlador de DMA está realizando la transferencia de datos, la CPU puede seguir ejecutando otras tareas sin esperar a que se complete la transferencia. Para saber cuándo se completa la transferencia, el controlador de DMA puede utilizar una señal de interrupción para informar a la CPU de que la transferencia se ha completado.

Cuando la CPU recibe la interrupción del controlador de DMA, puede leer los datos transferidos en el búfer de datos y continuar con la ejecución del programa. Alternativamente, si la transferencia de datos se realiza en el modo de escritura, la CPU puede escribir los datos en el búfer de datos y dejar que el controlador de DMA realice la transferencia a través del bus de sistema.

Por lo que, la programación de DMA permite que los dispositivos I/O transfieran datos a la memoria del sistema sin la intervención de la CPU, lo que mejora el rendimiento del sistema. La CPU y el dispositivo I/O coordinan la transferencia a través del controlador de DMA, y la CPU es informada de la finalización de la transferencia a través de una señal de interrupción generada por el controlador de DMA.

8. **Identifique dos razones por las que la cache es útil. ¿Qué problemas resuelve y causa?**

Reducción del tiempo de acceso a los datos: Cuando la CPU necesita acceder a los datos, primero busca en la caché. Si los datos están presentes en la caché, la CPU puede acceder a ellos más rápidamente que si tuviera que acceder a ellos en la memoria principal del sistema. Esto reduce el tiempo de espera de la CPU y mejora el rendimiento del sistema.

Reducción de la cantidad de tráfico de memoria: Cuando los datos están presentes en la caché, no es necesario que la CPU acceda a ellos en la memoria principal del sistema, lo

que reduce la cantidad de tráfico de memoria en el sistema. Esto puede reducir el consumo de energía y mejorar el rendimiento general del sistema.

Sin embargo, la memoria caché también puede causar algunos problemas, por ejemplo:

Coherencia de la caché: Si varios núcleos de la CPU comparten la misma memoria caché, puede haber problemas de coherencia de la caché. Si un núcleo modifica los datos en la caché, los otros núcleos pueden tener una copia desactualizada de los datos. Esto puede causar errores en el sistema y reducir el rendimiento.

Caché miss: Si la caché no contiene los datos que la CPU necesita, se produce un "caché miss". En este caso, la CPU tiene que acceder a los datos en la memoria principal del sistema, lo que puede ser mucho más lento que acceder a los datos en la caché. Un alto porcentaje de caché miss puede reducir significativamente el rendimiento del sistema.

9. **Explique con un ejemplo, como se manifiesta el problema de mantener la coherencia de los datos de cache en los siguientes entornos:**

◦ Sistema distribuido: En un sistema distribuido en el que varias máquinas comparten datos, la coherencia de la caché puede ser un problema. Por ejemplo, si dos máquinas A y B comparten un archivo, y una copia del archivo se almacena en la caché de cada máquina, si una máquina modifica el archivo, la otra máquina puede tener una copia desactualizada en su caché. Esto puede resultar en una situación en la que ambas máquinas tienen una copia diferente del archivo, lo que puede causar errores y problemas de sincronización.

◦ Sistema multiprocesador: En un sistema multiprocesador en el que varias CPUs comparten la misma memoria caché, la coherencia de la caché también puede ser un problema. Por ejemplo, si dos CPUs acceden a la misma ubicación de memoria, y una CPU modifica el valor en la caché, la otra CPU puede tener una copia desactualizada del valor en su caché. Si la otra CPU intenta leer el valor, puede obtener una copia desactualizada del valor, lo que puede causar errores en el sistema.

◦ Sistema de un solo procesador: En un sistema de un solo procesador, la coherencia de la caché puede ser un problema si el sistema utiliza múltiples niveles de caché. Por ejemplo, si una CPU tiene dos niveles de caché, y un valor se modifica en el nivel inferior de caché, el nivel superior de caché puede tener una copia desactualizada del valor. Si la CPU intenta leer el valor, puede obtener una copia desactualizada del valor, lo que puede causar errores en el sistema.