Seguridad en Sistemas Operativos Modernos

Ciberseguridad – FPUNA

2024-05-28

Contenido

1. Introducción
2. Estados privilegiados y no privilegiados.
3. Procesos e hilos de aplicaciones.
4. Memoria
5. Sistemas de archivos
6. Virtualización. Hipervisores.
7. Creación y operación de tecnología de virtualización.
8. Principios fundamentales de diseño de seguridad aplicados a un sistema operativo.
9. Controles de Acceso.
10. Separación de dominios, aislamiento de procesos, encapsulación de recursos, disminuir privilegios.

Creación y operación de tecnología de virtualización

Tecnología de Virtualización

- La virtualización, si bien ofrece grandes beneficios, también introduce nuevos desafíos de seguridad que deben ser considerados y mitigados para proteger los entornos virtualizados.

Aspectos de seguridad en la creación de entornos virtualizados

- **Elección del hipervisor**: El hipervisor es el componente central de un entorno virtualizado, por lo que es crucial seleccionar uno que tenga un historial probado de seguridad y que reciba actualizaciones de seguridad regulares.
- **Configuración segura del hipervisor**: El hipervisor debe configurarse de forma segura para minimizar la superficie de ataque. Esto incluye medidas como habilitar la autenticación fuerte, deshabilitar servicios innecesarios y aplicar las últimas correcciones de seguridad.
- **Aislamiento de las máquinas virtuales**: Las máquinas virtuales (VM) deben aislarse entre sí y del servidor host para evitar que un ataque en una VM pueda afectar a las demás. Esto se puede lograr mediante el uso de redes virtuales, grupos de seguridad y otras técnicas de aislamiento.
- **Protección de datos**: Los datos almacenados en las VM deben protegerse mediante el cifrado, el control de acceso y otras medidas de seguridad.
- **Gestión de identidades y accesos**: Es importante implementar un sistema de gestión de identidades y accesos (IAM) para controlar el acceso a las VM y a los recursos del servidor host.

Aspectos de seguridad en la operación de entornos virtualizados

- **Monitoreo y registro**: Es importante monitorear los entornos virtualizados para detectar actividades sospechosas y registrar todos los eventos de seguridad.
- **Actualizaciones de software**: Es importante aplicar las últimas actualizaciones de seguridad al hipervisor, a las VM y al software que se ejecuta en las VM.
- **Pruebas de penetración**: Es recomendable realizar pruebas de penetración regulares para identificar y corregir las vulnerabilidades de seguridad.
- **Capacitación del personal**: Es importante capacitar al personal sobre los riesgos de seguridad de la virtualización y las mejores prácticas para mitigarlos.

Herramientas de seguridad para entornos virtualizados

Existen diversas herramientas de seguridad disponibles para ayudar a proteger los entornos virtualizados. Estas herramientas pueden proporcionar funciones como:

- **Análisis de vulnerabilidades**: Identifica las vulnerabilidades en el hipervisor, las VM y el software que se ejecuta en las VM.
- **Detección de intrusiones**: Detecta actividades sospechosas en los entornos virtualizados.
- **Prevención de intrusiones**: Evita que los ataques se ejecuten en los entornos virtualizados.
- **Protección de datos**: Cifra y protege los datos almacenados en las VM.

Principios fundamentales de diseño de seguridad aplicados a un sistema operativo

Principios fundamentales de diseño de seguridad

- Conjunto de directrices que ayudan a crear sistemas operativos más seguros y confiables. Con estos principios, los desarrolladores pueden crear sistemas que sean más resistentes a ataques y que protejan mejor los datos de los usuarios.

Principios fundamentales de diseño de seguridad

- 1. Principio de mínima confianza
- 2. Defensa en profundidad
- 3. Falla segura
- 4. Separación de privilegios
- 5. Menor superficie de ataque
- 6. Protección contra errores de software
- 7. Cifrado
- 8. Gestión de identidades y accesos
- 9. Monitoreo y registro
- 10. Actualizaciones y parches

1. Principio de mínima confianza

Establece que ningún componente del sistema debe tener más privilegios de los que necesita para realizar su función. Esto ayuda a limitar el daño que puede causar un componente si se ve comprometido.

2. Defensa en profundidad

Se deben implementar múltiples capas de seguridad para proteger el sistema. Esto hace que sea más difícil para los atacantes penetrar en el sistema y, si lo logran, les dificulta escalar sus privilegios y acceder a datos confidenciales.

3. Falla segura

El sistema debe diseñarse de manera que, en caso de fallo, se encuentre en un estado seguro. Esto ayuda a prevenir que los errores o fallos del sistema sean explotados por los atacantes.

4. Separación de privilegios

Las tareas críticas del sistema deben realizarse por diferentes componentes con diferentes privilegios. Esto ayuda a prevenir que un atacante que comprometa un componente pueda realizar tareas críticas.

5. Menor superficie de ataque

Se debe minimizar la cantidad de código y interfaces expuestas a los usuarios y atacantes. Esto ayuda a reducir el número de posibles puntos de entrada para ataques.

6. Protección contra errores de software

El sistema debe diseñarse de manera que sea resistente a errores de software. Esto incluye la validación de entrada, la gestión de errores y las pruebas exhaustivas

7. Cifrado

Los datos confidenciales deben cifrarse tanto en reposo como en tránsito. Esto ayuda a proteger los datos contra el acceso no autorizado, incluso si el sistema se ve comprometido.

8. Gestión de identidades y accesos

El sistema debe implementar mecanismos sólidos para la gestión de identidades y accesos. Esto ayuda a garantizar que solo los usuarios autorizados puedan acceder a los recursos del sistema.

9. Monitoreo y registro

El sistema debe registrar los eventos de seguridad y que estos registros deben ser monitoreados para detectar actividades sospechosas. Esto ayuda a identificar y responder a los ataques de manera oportuna.

10. Actualizaciones y parches

El sistema debe mantenerse actualizado con los últimos parches de seguridad. Esto ayuda a proteger el sistema contra las vulnerabilidades conocidas.

Consideraciones de seguridad adicionales

Además de los principios fundamentales de diseño de seguridad, existen otras consideraciones importantes al diseñar un sistema operativo seguro. Estas consideraciones incluyen:

- **Amenazas**: Es importante identificar las amenazas potenciales a las que está expuesto el sistema operativo. Esto ayudará a determinar qué tipos de medidas de seguridad son necesarias.
- **Requisitos de seguridad**: El sistema operativo debe cumplir con todos los requisitos de seguridad relevantes, como los establecidos por las leyes y regulaciones.
- **Facilidad de uso**: Las medidas de seguridad no deben dificultar el uso del sistema operativo para los usuarios legítimos.

Al considerar todos estos factores, es posible crear un sistema operativo que sea seguro, confiable y fácil de usar.

Controles de Acceso

Controles de Acceso

- Los controles de acceso son un conjunto de medidas y mecanismos destinados a **limitar y regular el acceso a los recursos informáticos**, ya sean físicos o virtuales, a usuarios y entidades **autorizadas**.
- Su objetivo principal es proteger la información confidencial, prevenir el uso no autorizado de los recursos del sistema y mantener la integridad de este.

Mecanismos de controles de acceso

Autenticación: Proceso de verificar la identidad de un usuario o entidad.

Autorización: Proceso de determinar los permisos que tiene un usuario o entidad para acceder a un recurso del sistema y qué acciones puede realizar sobre él.

Listas de Control de Acceso (ACL): Estructuras de datos que asocian permisos a usuarios o grupos para recursos específicos.

Tablas de Control de Acceso (MAC): Tablas mantenidas por el sistema operativo que especifican los permisos de acceso para cada recurso y usuario/grupo.

Módulos de Control de Acceso (MAC): Componentes del sistema operativo que implementan la lógica de control de acceso, verificando las credenciales del usuario y los permisos asociados al recurso.

Modelos de controles de acceso

- **Control de acceso basado en listas de control de acceso (ACL)**: Asocia permisos específicos a usuarios o grupos para recursos individuales o grupos de recursos.
- **Control de acceso basado en roles (RBAC)**: Asigna permisos a los usuarios en función de sus roles dentro de la organización.
- **Control de acceso basado en atributos (ABAC)**: Permite definir permisos basados en una combinación de atributos del usuario, el recurso y el entorno.

Autenticación en Sistemas Operativos

- **Inicio de sesión**: Al iniciar el sistema operativo, los usuarios deben identificarse mediante un nombre de usuario y una contraseña u otro método de autenticación, como biometría o tokens de seguridad. El sistema operativo verifica la identidad del usuario y le permite acceder al sistema si las credenciales son válidas.
- **Control de cuentas de usuario**: Cada usuario del sistema operativo tiene una cuenta propia con un identificador único (UID) y un grupo principal (GID). Los permisos de acceso se asignan a usuarios y grupos, lo que permite controlar el acceso a los recursos del sistema en función de la identidad del usuario.

Autorización en Sistemas Operativos

- **Listas de control de acceso (ACL)**: Las ACL son listas asociadas a archivos, directorios y otros recursos del sistema que especifican qué usuarios o grupos tienen qué permisos de acceso (lectura, escritura, ejecución, etc.) sobre esos recursos.
- **Control de acceso basado en roles (RBAC)**: El RBAC asigna permisos a los usuarios en función de sus roles dentro de la organización. Los permisos se definen para cada rol y se aplican a todos los usuarios que tienen ese rol. Esto simplifica la administración de permisos y reduce la necesidad de asignar permisos individuales a cada usuario.
- **Control de acceso basado en atributos (ABAC)**: El ABAC es un modelo de control de acceso más flexible que permite definir permisos basados en una combinación de atributos del usuario, el recurso, el entorno y otras variables. Esto proporciona un control de acceso más granular y adaptable.

Implementación en sistemas operativos comunes

- **Windows**: Utiliza ACL y RBAC principalmente. Las ACL se pueden configurar manualmente o mediante la herramienta **`icacls`**. El RBAC se implementa mediante la directiva de grupo.
- **Linux:** Utiliza principalmente RBAC y SELinux (Security Enhanced Linux). SELinux es un sistema de control de acceso obligatorio que proporciona una capa adicional de seguridad al restringir las acciones que los procesos pueden realizar.
- **macOS**: Similar a Windows, utiliza ACL y RBAC. Las ACL se pueden configurar mediante la herramienta **`chmod`** y el RBAC se implementa mediante la cuenta de usuario y los grupos.

Métodos adicionales de control de acceso

- **Controles de acceso a la red**: Los firewalls y las redes privadas virtuales (VPN) se pueden utilizar para controlar el acceso a los recursos de la red, restringiendo el tráfico entrante y saliente.
- **Control de acceso a aplicaciones**: Las aplicaciones web y de escritorio pueden implementar sus propios controles de acceso, como autenticación de usuarios, autorización basada en roles y control de acceso a datos.

Importancia de los controles de acceso

- **Protegen la información confidencial**: Evitan que usuarios no autorizados accedan a datos sensibles.
- **Previenen el uso no autorizado de recursos**: Impiden que se utilicen los recursos del sistema para actividades maliciosas.
- **Mantienen la integridad del sistema**: Garantizan que solo usuarios autorizados puedan realizar cambios en el sistema.
- **Cumplen con las regulaciones de seguridad**: Ayudan a cumplir con las normas y regulaciones de protección de datos.

Separación de dominios, aislamiento de procesos, encapsulación de recursos, mínimo privilegio

Conceptos fundamentales que contribuyen a la protección de los sistemas y sus datos

Conceptos fundamentales que contribuyen a la protección de los sistemas y sus datos

- Separación de dominios
- Aislamiento de procesos
- Encapsulación de recursos
- Mínimo privilegio

Separación de dominios

- La separación de dominios implica dividir un sistema en secciones aisladas, cada una con sus propios recursos y permisos de acceso.
- Esta segmentación permite limitar el alcance de un ataque o fallo, impidiendo que se propague a otras áreas del sistema.

Aislamiento de procesos

- El aislamiento de procesos asegura que cada aplicación o programa se ejecute en su propio espacio de memoria, evitando que interfiera con otros procesos o acceda a recursos no autorizados.
- Esto reduce el riesgo de que un malware o una vulnerabilidad en un proceso afecte al sistema completo.

Encapsulación de recursos

- La encapsulación de recursos implica proteger los recursos del sistema, como archivos, memoria y dispositivos, mediante mecanismos de control de acceso.
- Esto garantiza que solo los usuarios y procesos autorizados puedan acceder y modificar estos recursos.

Mínimo privilegio

- El principio de "**Mínimo privilegio**" establece que los usuarios y procesos solo deberían tener los permisos necesarios para realizar sus tareas específicas.
- Limitar los privilegios reduce la superficie de ataque y dificulta que un atacante obtenga acceso a recursos críticos del sistema.

Implementación en Sistemas Operativos modernos

- **Control de acceso basado en roles (RBAC)**: Asigna permisos a los usuarios en función de su rol dentro del sistema.
- **Listas de control de acceso (ACL)**: Especifican qué usuarios y procesos pueden acceder a cada recurso y qué acciones pueden realizar.
- **Sandboxes**: Ejecutan aplicaciones en entornos aislados, protegiendo el sistema del malware y las vulnerabilidades.
- **Cuentas de usuario limitadas**: Restrigen el acceso a funciones administrativas y privilegios elevados.

Beneficios en Sistemas Operativos modernos

- **Protección contra malware y ataques:** Reduce la probabilidad de que un atacante comprometa el sistema y sus datos.
- **Minimización de daños:** Limita el alcance de un ataque o fallo, evitando que afecte a todo el sistema.
- **Mejora de la confiabilidad:** Garantiza que el sistema funcione de manera estable y segura.
- **Cumplimiento de normativas:** Ayuda a cumplir con las regulaciones de seguridad de datos y privacidad.