Hardening
Parte 1

Guía de Hardening para servidores Windows/Linux

¿Qué es el Hardening?

El hardening es el proceso de fortalecer la seguridad de un sistema, dispositivo o red para minimizar vulnerabilidades y reducir el riesgo de ciberataques. Este enfoque es fundamental para proteger datos sensibles, garantizar la integridad de la infraestructura, cumplir con normativas de seguridad y mitigar posibles amenazas.

Una vez comprendida la definición de hardening, al profundizar en el concepto, nos encontramos con el siguiente aspecto:

Entendimiento de servidores:

- Arquitectura
- Componentes Críticos
- Actualizaciones y parches

ARQUITECTURA: la arquitectura de un servidor hace referencia a la forma en que los componentes del sistema están configurados para minimizar los riesgos de seguridad. Esto incluye:

1. **Configuración Segura:** asegurar los accesos físicos al servidor, como el uso de dispositivos de autenticación y bloqueo de puertos no necesarios.

2. **Sistema Operativo:** El sistema operativo debe estar configurado de manera segura, desactivando servicios innecesarios, limitando el acceso de usuarios.

3. **Configurar firewalls:** segmentar la red para limitar el acceso no autorizado. Usar VLANs o VPNs para separar la red interna y externa, y asegurar o desactivar las conexiones remotas si no se utilizan mediante protocolos seguros (como SSH en lugar de Telnet).

Componentes Críticos: Una vez entendida la arquitectura, es necesario identificar los componentes críticos para recopilar la mayor cantidad de información y registrar eventos en el momento en que se reciba algún ataque. Por ejemplo, cuando los usuarios utilizan una aplicación web, estos eventos deben ser registrados para poder detectar cualquier tipo de vulnerabilidad.

Actualización y Parches: El proceso de actualización y parcheo es fundamental para asegurar que un servidor sea resistente a las vulnerabilidades Esto incluye:

- **Actualización de Sistema Operativo:** el sistema operativo se mantenga al día con las últimas actualizaciones de seguridad.

- **Parches de Seguridad:** Es esencial aplicar parches de manera oportuna, especialmente aquellos relacionados con vulnerabilidades críticas. La gestión de parches debe incluir una estrategia para la revisión y prueba de parches antes de ser desplegados, minimizando riesgos operacionales.

Estrategias de hardening

Principio del menor privilegio: nos va ayudar a identificar desde que punto un usuario sin privilegios o con los privilegios mínimos logra obtener acceso con administrador.

Segmentación de roles: es útil para segmentar diferentes sectores, por ejemplo: Contabilidad, RR. HH, estos usuarios van tener los minios privilegios a ciertos recursos, directorios, etc.

Políticas de parche y actualizaciones: es muy importante entender que, si tenemos una aplicación corriendo en el servidor, por algún motivo
Ejecutamos algún parche de actualización sin ante testear en un servidor de pruebas se puede dar el caso de que corrompamos la aplicación.

GPO/SeLinux: En caso de Windows, Linux tenemos diferentes herramientas que nos va ayudar crear, políticas para hacer cumplir estos controles por ejemplo personal de RR. HH solo acceso a herramientas de office y navegador web.

Control de accesos y gestión de entidades(objetos): es un proceso que a regular quién puede acceder a los recursos de un servidor y qué acciones pueden realizar por ejemplos: como archivos, directorios o bases de datos. Cada objeto debe tener configurados permisos específicos que definan qué usuarios o grupos pueden leer, escribir, modificar o ejecuta.

Auditoría y monitoreo: El hardening no solo fortalece la seguridad del sistema, sino que también facilita la auditoría de eventos relacionados con los usuarios. Por ejemplo, si un usuario es infectado con malware, será necesario realizar una auditoría o un análisis forense para determinar cómo ocurrió el incidente y qué vectores de ataque fueron utilizados. Además, el monitoreo constante de todos los puntos finales (endpoints) y servidores nos proporciona una visibilidad continua de la red, lo que permite detectar actividades sospechosas más rápidamente y responder de manera más efectiva a posibles amenazas.

Hardening de servidores Linux

Como primer punto importante en Hardening de servidores Linux es la actualización de paquetes

- Sudo apt-get update
- Sudo apt-get upgrade –y

Una vez realizada la respetiva actualización vamos a verificar los servicios que están utilizan con el siguiente comando:

systemctl list-units --type=service --state=running

- **systemctl:** Herramienta para gestionar servicios.
- **list-units:** Lista unidades activas.
- **--type=service:** Filtra las unidades para mostrar solo los servicios.
- **--state=running:** Incluye únicamente los servicios en estado de ejecución.

Para ver el estado de un servicio por ejemplo **apache2.service** utilizamos el siguiente comando:

systemctl status apache2.service

vemos que se encuentra activo

Para desactivar o activar el servicio utilizamos los siguientes comandos.

systemctl stop apache2.service

systemctl start apache2.service

SUDO

en sistemas Linux vamos a tener el comando sudo para elevar privilegios como anterior mente se mencionado no todos los usuarios deben tener privilegios de administrador esto nos va permitir ningún usuario sin premisos pueda realizar cambios, por ejemplo:

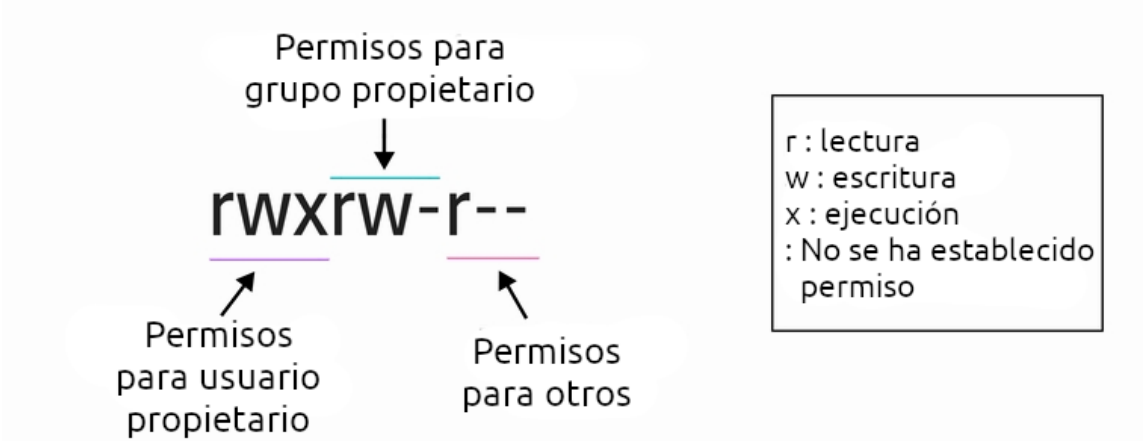
- Permitir sudo a usuarios sin contraseñas

- Permitir a un usuario usar sudo con ciertos comandos

SUID (Set User ID)

Premisos especiales para los usuarios

Un archivo puede tener premisos especiales para que los usuarios pueden ejecutar o tener acceso al mismo



Seguridad del protocolo ssh

Los siguiente que vamos a hablar es la seguridad del protocolo **SSH** algunas buenas prácticas son:

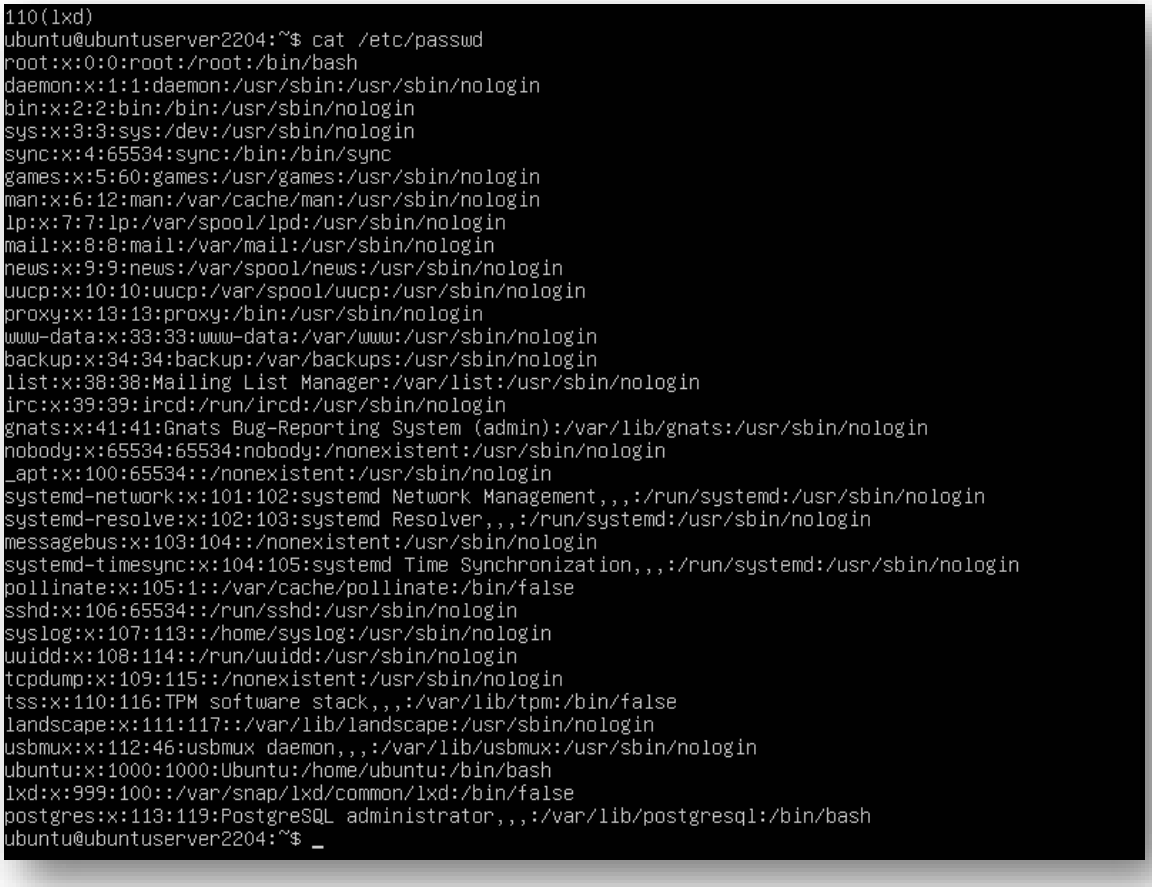
- Cambiar el puerto de escucha
- No permitir el login de los usuarios root
- No permitir contraseñas en blanco
- Numero máximos de intentos
- Número máximo de sesiones

El siguiente tema que vamos a hablar es de **premisos a archivos críticos** algunos de ellos son**:**

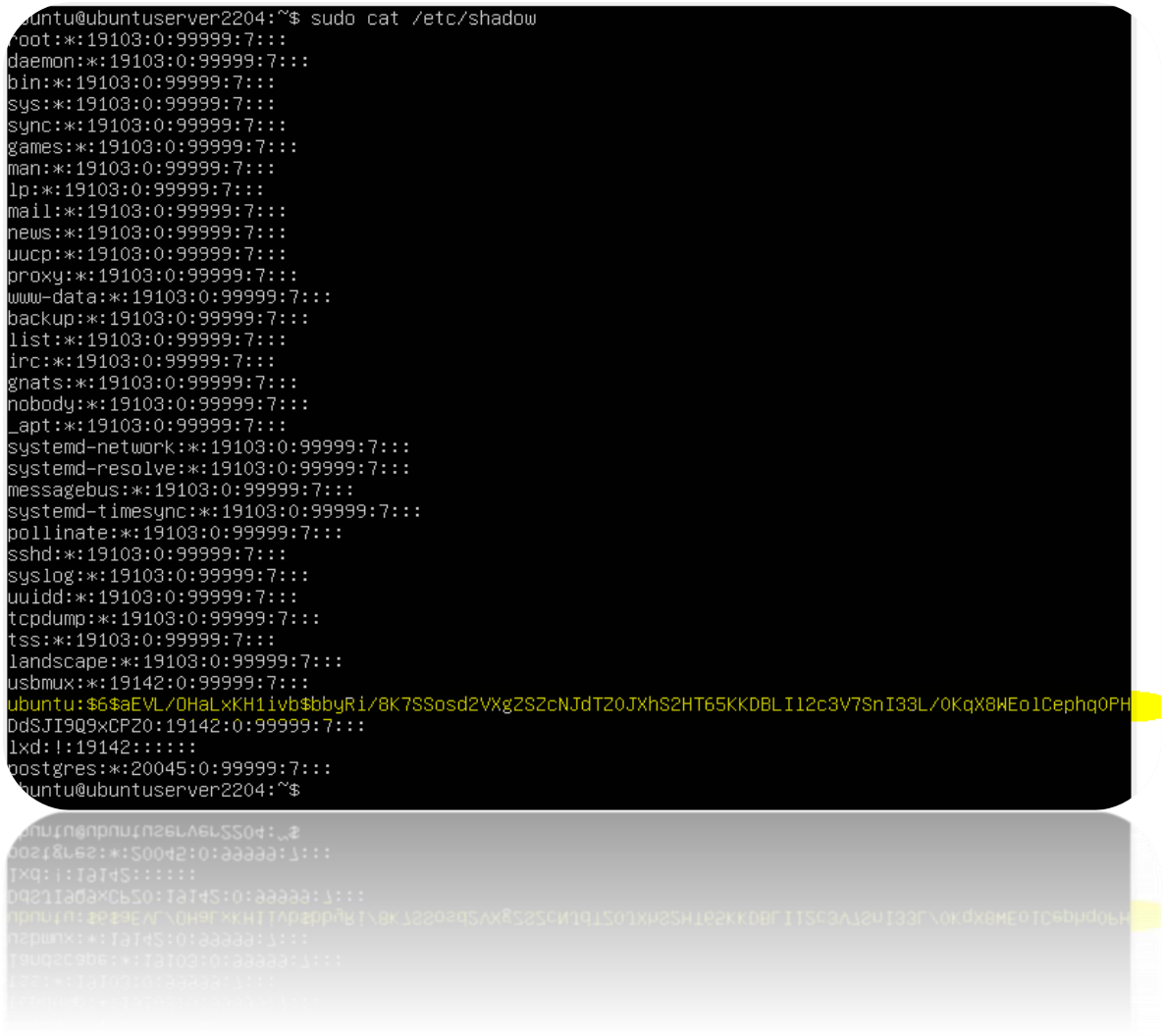
- /etc/shadow
- /etc/gshadow
- /etc/passwd

- /etc/group
- /etc/ssh/sshd_config

Por ejemplo, si estamos utilizando el usuario Ubuntu, este no debería tener acceso a ciertos tipos de archivos sensibles. Un caso concreto es el archivo **/etc/passwd**, donde se puede ver una lista de todos los usuarios registrados en el sistema.



Esto puede permitir que usuarios malintencionados lleven a cabo ataques de fuerza bruta. Otro archivo crítico al que los usuarios sin privilegios no deberían tener acceso es **/etc/shadow**, ya que contiene los hashes de las contraseñas de los usuarios del sistema.



Por últimos vamos a hablar de **bloquear el trafico entrante de la IP**

Si vamos a utilizar SSH, es recomendable permitir el acceso únicamente a usuarios específicos y desde direcciones IP autorizadas. Además, podemos gestionar el tráfico entrante y saliente directamente desde el sistema utilizando iptables, una herramienta que exploraremos más adelante.

Implementar estas medidas de seguridad en sistemas Linux es fundamental para proteger los recursos del servidor y prevenir accesos no autorizados. Sin embargo, la seguridad no se detiene aquí. A continuación, exploraremos cómo aplicar principios de hardening en **servidores Windows** para fortalecer aún más nuestra infraestructura."

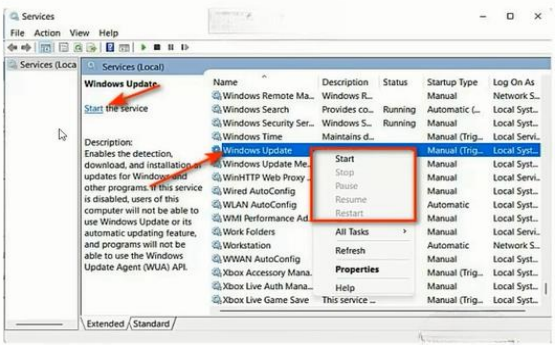
Hardening de servidores Windows

Los más importante como vimos en sección anterior es tener nuestro sistema actualizado, parches al día para evitar vulnerabilidades es fundamento realizar a actualización en un servidor de pruebas una vez que vemos que funciona todo correctamente se despliega en el servidor de producción

Los sistemas Windows como los sistemas Linux tiene una gran variedad de servicio corriendo los cual algunos no sean necesarios ya que por estos servicios podemos ser vulnerables

Listar los servicios

```
Get-Service | Where-Object {$_.Status -eq 'Running'}
```

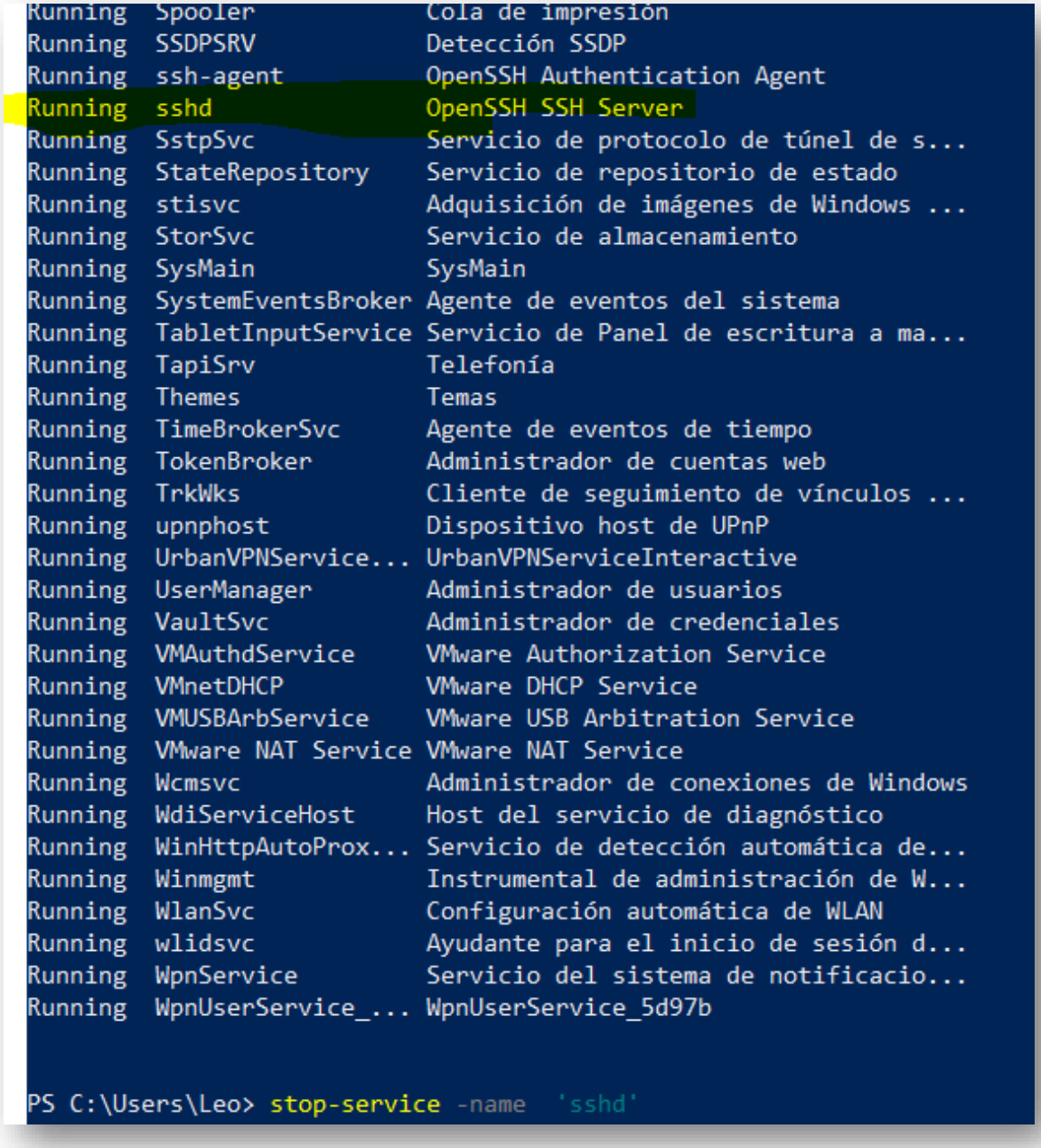


Detener un servicio

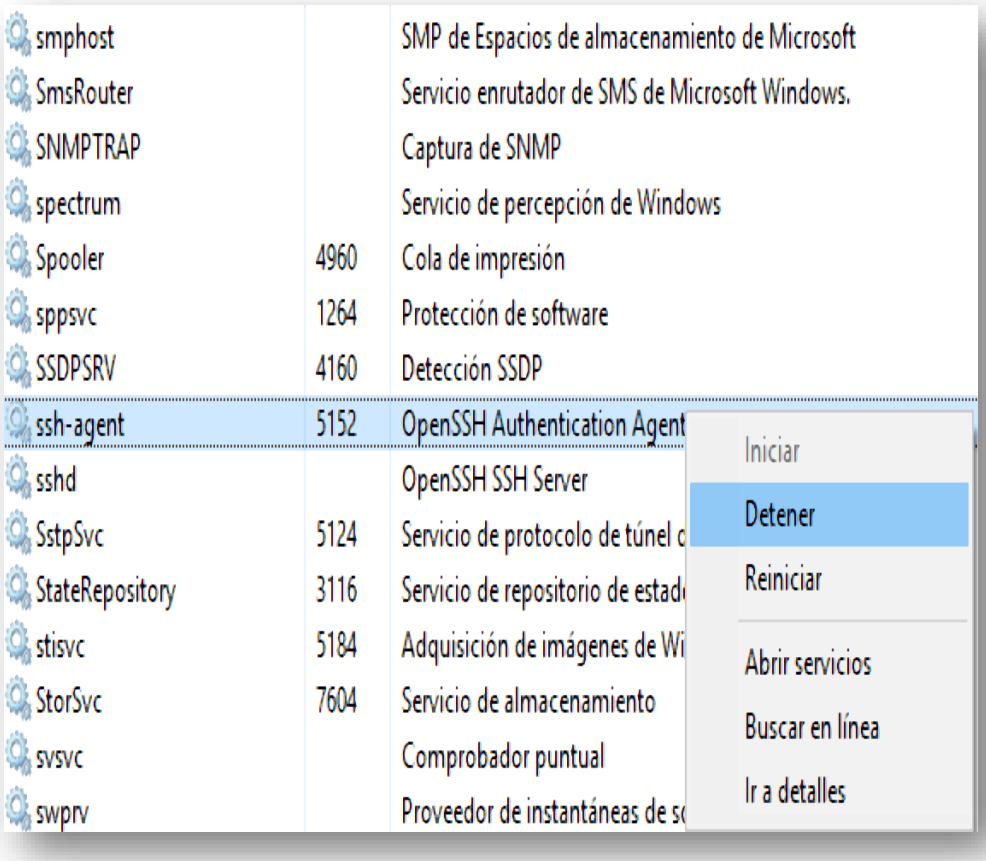
```
stop-service -name ‘nombre’
```

```
set-service -name ‘nombre’ -startuptype disable
```

Por ejemplo, voy detener el servicio SSH desde powershell con el comando **stop-service –name ‘nombre’**



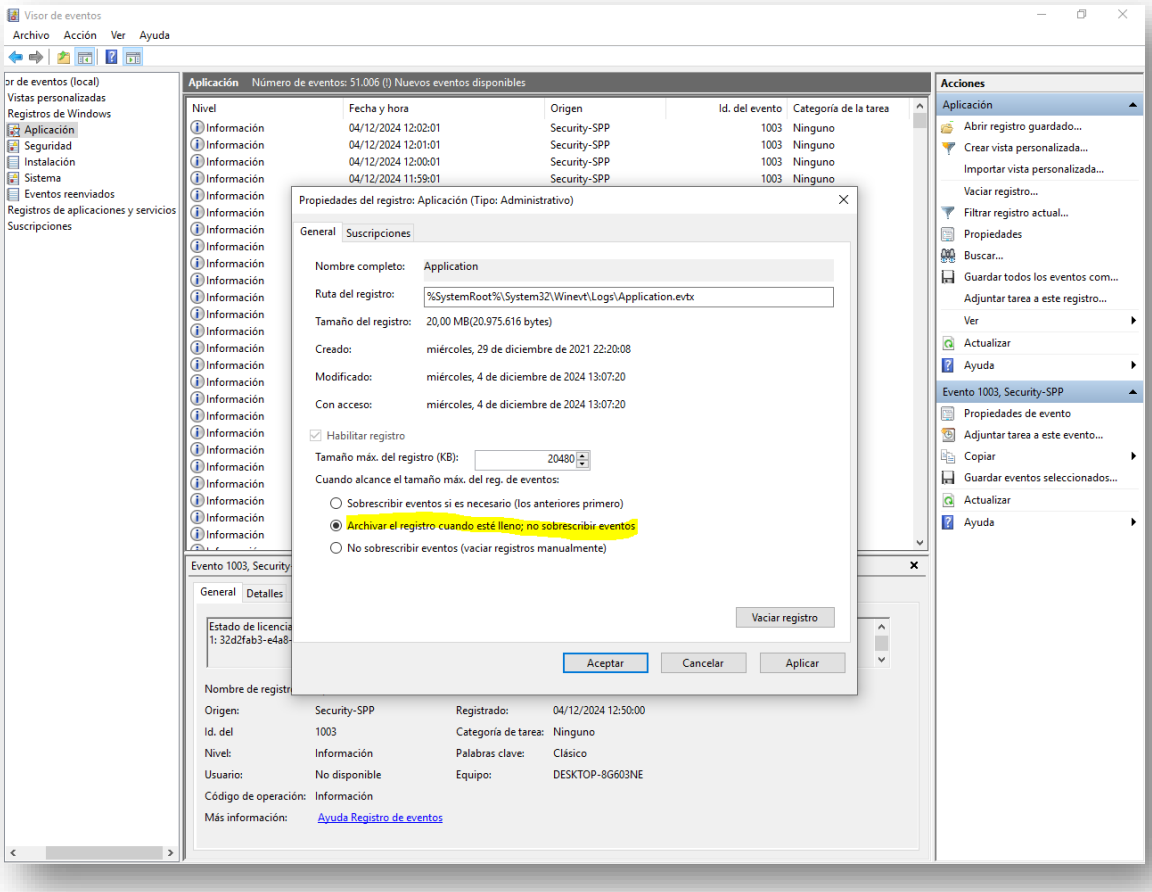
También se puede deshabilitar estos servicios desde el administrador de tarea:



Lo siguiente que vamos a ver son los **Event log** en los vamos a poder encontrar ciertas opciones muy interesantes para realizar un análisis forense tenemos que tener este tipo de registros de los eventos, pero si no realizamos un Hardening estos registros se pueden sobre escribir que, sucedido anteriormente, estamos buscando.

- Tamaño de archivos
- Retención de eventos
- Sobre escribir archivos

Para realizar algún cambio en el registro del sistema, debemos acceder a él y hacer clic derecho sobre el registro que queremos configurar (ya sea de aplicación, seguridad o sistema). Luego, seleccionamos la opción "Propiedades". Es importante elegir la configuración que mejor se adapte a nuestras necesidades. Por lo general, la opción más **recomendada es archivar el registro cuando esté lleno**, ya que evita sobrescribir eventos importantes.



Continuando también podemos trabajar con las: **directivas de seguridad locales** donde podemos modificar, crear ciertas políticas como:

Usuarios

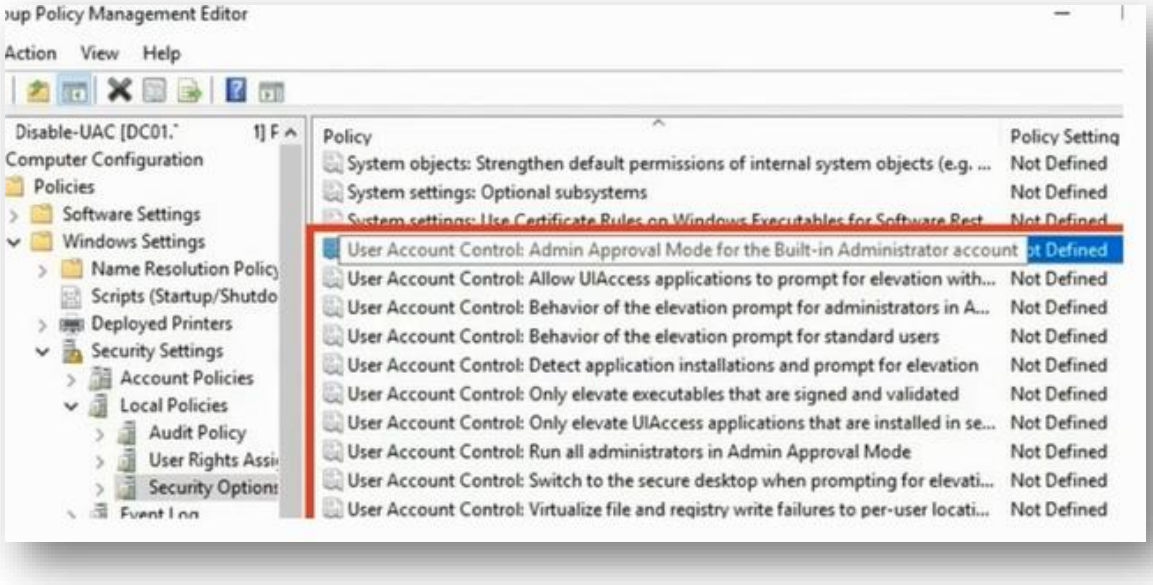
- Contraseña
- Permisos a programas
- Permitir o denegar acceso a recursos u objetos.

- **Firewall**

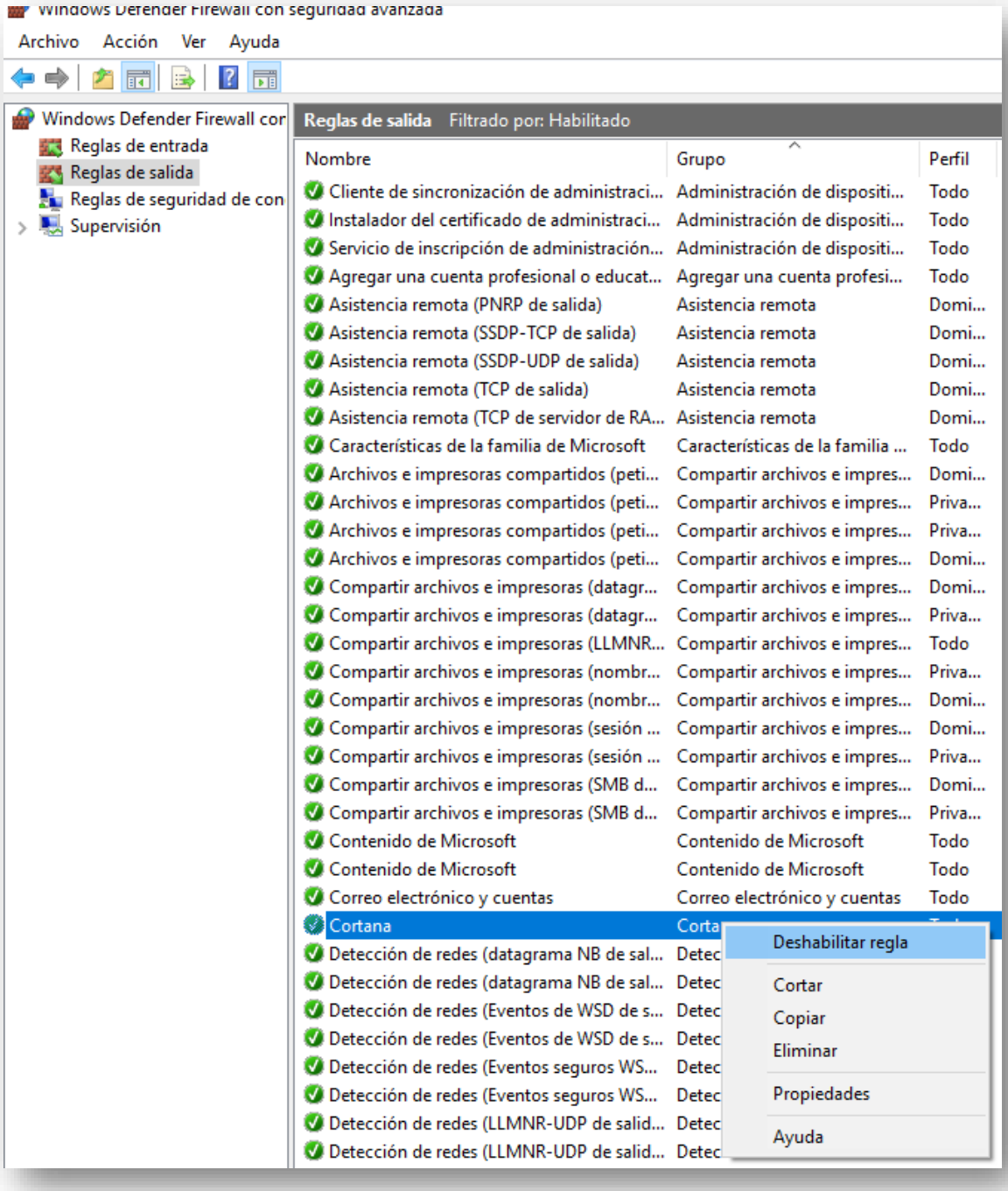
- Bloqueo de icmp

- Bloqueo de puertos

- Bloqueo de IP



En el apartado de Firewall Windows cuenta con su propio firewall donde podemos configurar reglas de entrada y salida:



Si queremos adentrarnos en opciones más avanzadas, podemos utilizar tecnologías como**:**

DEP (Data Execution Prevention)

- se basa en marcar ciertas áreas de la memoria como "no ejecutables". Esto significa que solo el código legítimo cargado en áreas designadas puede ejecutarse

ASLR (Address Space Layout Randomization)

- protege contra desbordamiento de memoria. Se combina con DEP para fortalecer la mucha más seguridad