110.2 Configurar la seguridad del host - Ejercicios

Nota: Estos ejercicios implican visualizar configuraciones de seguridad y, opcionalmente, realizar cambios (con precaución y solo en VMs de prueba). Modificar la configuración de SSH o el firewall puede bloquear tu acceso al sistema si no se hace correctamente.

Ejercicio 10.2.1: Explorando la Configuración del Servidor SSH (`sshd_config`)

- **Objetivo:** Localizar y visualizar las directivas de seguridad importantes en el archivo de configuración de SSH.
- **Requisitos:** Servidor SSH instalado (paquete `openssh-server`). Privilegios de superusuario (`sudo`).
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Localiza el archivo de configuración:** El archivo es `/etc/ssh/sshd_config` en **ambas ramas (Debian y Red Hat)**.
    3. **Visualiza el archivo:** Ejecuta `sudo less /etc/ssh/sshd_config`.
    4. **Busca las siguientes directivas (pueden estar comentadas con #):**
        - `Port`: El puerto de escucha.
        - `PermitRootLogin`: Si se permite el login de root.
        - `PasswordAuthentication`: Si se permite la autenticación por contraseña.
        - `PubkeyAuthentication`: Si se permite la autenticación por clave pública.
        - `AllowUsers`: Si se restringe el acceso a usuarios específicos.
        - `AllowGroups`: Si se restringe el acceso a grupos específicos.
    5. **(Concepto - ¡No lo hagas si no estás seguro!):** Para aplicar cambios, editarías este archivo (`sudo vi /etc/ssh/sshd_config`), guardarías, y luego reiniciarías el servicio SSH: `sudo systemctl restart ssh.service` (o `sshd.service`). **Ten una consola local o un plan de recuperación antes de hacer cambios que puedan impedir el acceso remoto.**

Ejercicio 10.2.2: Configurando Autenticación Basada en Clave SSH (¡En VM de Prueba!)

- **Objetivo:** Configurar la autenticación SSH usando claves en lugar de contraseña.
- **Requisitos:** Tener acceso SSH a la VM. **Dos máquinas (cliente y servidor SSH) o usar localhost como ejemplo.** `openssh-server` en el servidor, `openssh-client` en el cliente. **Realiza este ejercicio solo en una VM de prueba.**
- **Desarrollo Paso a Paso:**
    1. **En la máquina cliente (o en tu propia VM si la usas como cliente y servidor):**
        - Genera un par de claves SSH: Ejecuta `ssh-keygen`. Acepta la ubicación por defecto (`~/.ssh/id_rsa`) y **establece una frase de paso (passphrase)** para proteger la clave privada.

    - Verás los archivos `~/.ssh/id_rsa` (clave privada) y `~/.ssh/id_rsa.pub` (clave pública).
2. **En la máquina cliente, copia la clave pública al servidor (requiere conocer la contraseña del usuario en el servidor al principio):** Ejecuta `ssh-copy-id tu_usuario@<ip_servidor>`. Introduce la contraseña cuando te la pida. Esto añade tu clave pública (`~/.ssh/id_rsa.pub`) al archivo `~/.ssh/authorized_keys` en el servidor, en el directorio personal del usuario `tu_usuario`.
3. **En la máquina cliente, prueba a iniciar sesión vía SSH:** Ejecuta `ssh tu_usuario@<ip_servidor>`. Si la clave se configuró correctamente, te pedirá la **frase de paso** de tu clave privada (no la contraseña del usuario del servidor). Una vez ingresada, deberías iniciar sesión.
4. **(Opcional - En el servidor, para deshabilitar autenticación por contraseña):** Edita `sudo vi /etc/ssh/sshd_config`. Cambia `PasswordAuthentication yes` a `PasswordAuthentication no`. Reinicia el servicio SSH (`sudo systemctl restart ssh.service`). **Asegúrate de que la autenticación por clave funciona ANTES de hacer esto, de lo contrario te bloquearás el acceso.**
5. **Prueba a iniciar sesión de nuevo como en el paso 3.** Ahora solo debería funcionar la autenticación por clave.

Ejercicio 10.2.3: Identificando y Verificando el Servicio de Firewall

- **Objetivo:** Determinar qué servicio de firewall está activo en tu sistema y verificar su estado.
- **Requisitos:** Acceso a la línea de comandos.
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Verifica el estado de `firewalld` (común en Red Hat):** Ejecuta `systemctl status firewalld.service`.
    3. **Verifica el estado de `ufw` (común en Ubuntu):** Ejecuta `systemctl status ufw.service`.
    4. **Conclusión:** Solo uno de ellos (o ninguno si se usa `iptables` o `nftables` directamente) debería estar activo.

Ejercicio 10.2.4: Viendo las Reglas de Firewall (Diferencias Debian vs. Red Hat)

- **Objetivo:** Usar los comandos del firewall de alto nivel para ver las reglas activas.
- **Requisitos:** Identificar el firewall activo (ufw o firewalld).
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Si `firewalld` está activo:** Ejecuta `sudo firewall-cmd --list-all`. Esto muestra la configuración de la zona activa (servicios permitidos, puertos, etc.).

3. **Si `ufw` está activo:** Ejecuta `sudo ufw status`. Muestra si el firewall está activo y las reglas (permisos/denegaciones) configuradas.
4. **Si se usa `iptables` o `nftables` directamente:** Ejecuta `sudo iptables -L`. Esto muestra las cadenas y reglas del filtro de paquetes. (La salida puede ser compleja). Ejecuta `sudo iptables -vnL` para ver contadores de paquetes y bytes.

Ejercicio 10.2.5: Permitiendo/Denegando un Puerto (¡En VM de Prueba!)

- **Objetivo:** Usar el comando del firewall activo para abrir o cerrar un puerto (temporal o permanentemente).
- **Requisitos:** Firewall activo (`ufw` o `firewalld`). Privilegios de superusuario (`sudo`). **Realiza este ejercicio solo en una VM de prueba.** Conoce un puerto para probar (ej: 80 si no tienes web server, o un puerto aleatorio).
- **Desarrollo Paso a Paso:**
  1. Abre una terminal.
  2. **Si `firewalld` está activo:**
     - Permite el tráfico en el puerto 80 (temporal, hasta reiniciar): `sudo firewall-cmd --zone=public --add-port=80/tcp`.
     - Verifica la regla temporal: `sudo firewall-cmd --zone=public --list-ports`.
     - Hazlo permanente: `sudo firewall-cmd --zone=public --add-port=80/tcp --permanent`.
     - Para que el cambio permanente surta efecto sin reiniciar: `sudo firewall-cmd --reload`.
     - Deniega el tráfico en el puerto 80 (permanente): `sudo firewall-cmd --zone=public --remove-port=80/tcp --permanent`.
     - Recarga para aplicar: `sudo firewall-cmd --reload`.
  3. **Si `ufw` está activo:**
     - Permite el tráfico en el puerto 80: `sudo ufw allow 80/tcp`.
     - Verifica la regla: `sudo ufw status`.
     - Deniega el tráfico en el puerto 80: `sudo ufw deny 80/tcp`.
     - Verifica la regla: `sudo ufw status`.
     - Deshabilita el firewall (¡cuidado!): `sudo ufw disable`. Habilita: `sudo ufw enable`.
  4. **(Concepto):** Para probar si el cambio de firewall surtió efecto, intenta conectarte a ese puerto desde otra máquina (o desde localhost si el servicio está corriendo) usando `nc -zv <ip> <puerto>`.

Ejercicio 10.2.6: Listando Servicios en Ejecución

- **Objetivo:** Identificar qué demonios están activos y potenciales puntos de ataque.
- **Requisitos:** Acceso a la línea de comandos.

- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Lista las unidades de servicio activas:** Ejecuta `systemctl list-units --type=service --state=running`.
    3. **Revisa la lista:** Identifica servicios que no reconoces o que no necesitas (ej: servidores de juegos, servicios de escritorio si estás en un servidor headless).
    4. **(Concepto):** Si identificas un servicio innecesario, puedes deshabilitarlo (`sudo systemctl disable <nombre_servicio>`) para que no inicie al arrancar y detenerlo (`sudo systemctl stop <nombre_servicio>`) para terminarlo inmediatamente.