CloudOps Guild

Together, towards mastery in Cloud and DevOps

Este es un checklist de **buenas prácticas y seguridad al conectar por SSH** que te servirá para proteger tus conexiones por SSH.

✅ Checklist de Buenas Prácticas y Seguridad para Conexiones SSH

🔹 1. Configuración de Usuarios y Permisos

✔️ Deshabilitar el acceso SSH para el usuario `root` (`PermitRootLogin no` en `/etc/ssh/sshd_config`).
✔️ Crear usuarios específicos con privilegios mínimos en lugar de usar `root` directamente.
✔️ Aplicar el principio de **menor privilegio** con `sudo` solo cuando sea necesario.
✔️ Configurar permisos correctos en `/home/usuario/.ssh/` y archivos dentro de `.ssh/` (`chmod 700 .ssh/ && chmod 600 .ssh/authorized_keys`).
✔️ Usar grupos específicos para gestionar acceso SSH (`AllowGroups sshusers` en `/etc/ssh/sshd_config`).

🔹 2. Uso de Claves Privadas y Rotación de Claves

✔️ Deshabilitar autenticación por contraseña y exigir claves SSH (`PasswordAuthentication no`).
✔️ Generar claves SSH seguras con `ssh-keygen -t ed25519 -C "tu@email.com"`.
✔️ Evitar el uso de claves RSA de menos de 4096 bits.
✔️ Implementar **rotación periódica de claves** (cada 6-12 meses según criticidad).
✔️ Revocar y eliminar claves de usuarios inactivos o que hayan cambiado de rol.

🔹 3. Habilitación de Bastion Hosts y Uso de VPN

✔️ Utilizar un **Bastion Host** para administrar el acceso SSH en entornos productivos.
✔️ Configurar acceso al Bastion Host solo desde direcciones IP permitidas.
✔️ Usar autenticación en 2 factores (MFA) en el Bastion Host.
✔️ Limitar accesos SSH en servidores internos solo desde el Bastion Host (`AllowUsers usuario@bastion-host`).
✔️ Si es posible, usar una VPN en lugar de exponer SSH a internet.

CloudOps Guild

Together, towards mastery in Cloud and DevOps

🔹 4. Seguridad en el Servidor SSH

✔️ Cambiar el puerto predeterminado de SSH (`Port 2222` en `/etc/ssh/sshd_config`).
✔️ Limitar intentos fallidos con `MaxAuthTries 3`.
✔️ Usar `AllowUsers` o `AllowGroups` en `/etc/ssh/sshd_config` para restringir accesos.
✔️ Deshabilitar protocolos inseguros como SSHv1 (`Protocol 2`).
✔️ Configurar `ClientAliveInterval` y `ClientAliveCountMax` para cerrar sesiones inactivas.
✔️ Utilizar `Banner` para mostrar advertencias antes del inicio de sesión.

🔹 5. Registro y Monitoreo de Accesos

✔️ Habilitar y revisar logs en `/var/log/auth.log` (`journalctl -u ssh`).
✔️ Configurar alertas en herramientas como CloudWatch, Splunk o SIEM para detectar accesos sospechosos.
✔️ Implementar `fail2ban` para bloquear direcciones IP con múltiples intentos fallidos.
✔️ Activar logging detallado (`LogLevel VERBOSE` en `/etc/ssh/sshd_config`).

🔹 6. Protección Adicional con Firewalls y Listas de Control de Acceso

✔️ Configurar reglas en `iptables`, `nftables` o `ufw` para permitir SSH solo desde IPs específicas.
✔️ Si usas AWS, Azure o GCP, restringir acceso SSH con reglas en **Security Groups** o **Firewall Rules**.
✔️ Usar **Port Knocking** o **TCP Wrappers** (`/etc/hosts.allow` y `/etc/hosts.deny`).

🔹 7. Uso de SSH-Agent y Forwarding Seguro

✔️ Evitar el uso de `ssh-agent` con sesiones persistentes abiertas.
✔️ Deshabilitar **SSH Agent Forwarding** (`AllowAgentForwarding no` en `sshd_config`) si no es necesario.
✔️ Si se usa Forwarding, restringirlo a usuarios confiables y servidores de confianza.

🔹 8. Uso de Certificados y Autenticación Avanzada

CloudOps Guild

Together, towards mastery in Cloud and DevOps

✔️ Considerar el uso de **certificados SSH** en lugar de claves estáticas para mayor seguridad.
✔️ Implementar **autenticación multifactor (MFA)** para accesos críticos.
✔️ Explorar herramientas como **AWS SSM Session Manager** para evitar exponer SSH en instancias de AWS.

Este checklist te ayudará a **endurecer la seguridad** de tus conexiones SSH, minimizar riesgos y proteger mejor los accesos a tus servidores. 🚀🔐