proHERRAMIENTAS DE ACCESO REMOTO

Protocolos de administración remota que permite a los usuarios controlar, modificar y administrar sus servidores remotos a través de Internet.

- Telnet
- SSH

TELNET (TELecommunication NETwork)

- Funcionamiento general solo permite acceder en modo terminal (sin gráficos)
- Algunos clientes de telnet puede realizar conexiones soporte gráfico: NetRunner, Putty, Zoc.
- Puerto de conexión 23
- Sistema de comunicación bidireccional semiduplex.
- Se aplica en una conexión TCP
- Su mayor problema es de seguridad

Telnet Puede trabajar bajo en amparo del servidor Inetd, o de forma autónoma (Standlone). **El servidor Inetd** es un servicio de la mayoría de sistemas Unix que gestiona las conexiones de varios servicios como telnet, ftp, daytime, pop, imap, etc. La ejecución de una única instancia reduce la carga del sistema, en comparación a ejecutar cada uno de los servicios de forma individual

Instalación:

Apt-get install telnetd	Servidor Telnet
Apt-get install telnet	Cliente Telnet.

Bajo el Servidor Inetd, para habilitar y deshabilitar el servicio Telnet, se quita el comentario de la línea de que referencia Telnet en el archivo de Configuración es ***/etc/inetd.conf***, y se reinicia el servidor Inetd.

Administrar el servicio inetd: ***service inetd (start|stop|restart …)***
systemctl (start|stop|restart …) inetd

Para Acceder desde un cliente Telnet a un servidor Telnet: ***telnet <Ip del servidor Telnet>***

Protocolo SSH (Secure Shell)

- A diferencia de otros protocolos de comunicación remota tales como FTP o Telnet, SSH encripta la sesión de conexión, haciendo imposible que alguien pueda obtener contraseñas no encriptadas.
- El protocolo SSH es una herramienta que nos permite conectarnos a equipos remotos y compartir información en forma segura.
- SSH usa una arquitectura cliente/servidor.
- SSH permite gestionar sistemas criptográficos con clave pública, para crear un canal seguro tunelizado en la comunicación.
    - RSA (Ron Rivest, Adi Shamir, Len Adleman):

        - Algoritmo Asimétrico cifrador en bloques
    - DSA (Digital Signature Algorithm, Algoritmo de firma digital)
        - Estándar del gobierno federal de EEUU para firmas digitales
    - Un Sistema criptográfico con clave pública, es un algoritmo cifrador que genera una clave pública la cual se distribuye en forma autenticada a los usuarios que se considere, y una clave privada que es guardada por el propietario en forma secreta.
    - Cuando se quiere enviar un mensaje, el emisor busca la clave pública de cifrado del receptor, cifra su mensaje con esa clave, y una vez que el mensaje cifrado llega al receptor, éste se ocupa de descifrarlo usando su clave privada

- SSH Funciona sobre la capa de aplicación del modelo TCP/IP
- SSH atiende peticiones por el puerto 22

Secuencia de Eventos de una Conexión SSH

La seguridad de la comunicación SSH se implementa con los siguientes eventos:

- Se lleva a cabo un 'handshake' (apretón de manos) encriptado para que el cliente pueda verificar que se está comunicando con el servidor correcto.
- Encriptación de la capa de transporte entre cliente y servidor mediante un código simétrico.
- El cliente se autentica ante el servidor.
- El cliente interactúa con la máquina remota sobre la conexión encriptada.

Capa de transporte

- El papel principal de la capa de transporte es facilitar una comunicación segura entre dos hosts durante la autenticación y la subsecuente comunicación.
- Maneja la encriptación y decodificación de datos y proporcionando protección de integridad de los paquetes de datos mientras son enviados y recibidos.
- Proporciona compresión de datos, para acelerar la transmisión.

Autenticación

Cuando la capa de transporte haya construido un túnel seguro para transmitir información entre los dos sistemas, el servidor le dirá al cliente de los diferentes métodos de autenticación soportados, tales como el uso de firmas privadas codificadas con claves o la inserción de una contraseña. El cliente entonces intentará autenticarse ante el servidor mediante el uso de cualquiera de los métodos soportados.

Canales

Luego de una autenticación exitosa sobre la capa de transporte SSH, se abren múltiples *canales* a través de la técnica llamada multiplexar. Cada uno de estos canales maneja la conexión para diferentes sesiones de terminal y para sesiones X11.

OpenSSH (Open Secure Shell)

- Conjunto de aplicaciones que permiten realizar comunicaciones cifradas a través de una red usando como base el protocolo SSH
- Fue creado como una alternativa libre y abierta al programa ***Secure Shell*** que es software propietario de la compañía ***SSH Communicatios Security***
- Actualmente liderado por Theo de Raadt (ingeniero de software actualmente residente en Calgary, Canadá. Fundador y líder de los proyectos OpenBSD y OpenSSH.
- Se considera que OpenSSH es más seguro que el original SSH, por ser código limpio y perfectamente auditado
- Permite conexiones y copiar datos de forma segura mediante la implementación de herramientas como
    - SSH: Secure Shell
    - SCP: Shell Secure Copy

- SFTP: Security File Transfer Protocol

Instalación:

Apt-get install openssh-server Servidor SSH
Apt-get install openssh-client Cliente SSH

Archivos de configuración

OpenSSH dispone de dos conjuntos diferentes de ficheros de configuración: uno dedicado completamente al cliente (ssh, scp, sftp) y otro orientado completamente al servidor sshd

Servidor SSH

Archivos de Configuración del lado del servidor

moduli	Contiene grupos Diffie-Hellman usados para el intercambio de la clave Diffie-Hellman que es imprescindible para la construcción de una capa de transporte seguro. Cuando se intercambian las claves al inicio de una sesión SSH, se crea un valor secreto y compartido que no puede ser determinado por ninguna de las partes individualmente. Este valor se usa para proporcionar la autenticación del host.
ssh_config	Archivo de configuración del sistema cliente SSH por defecto. Este archivo se sobrescribe si hay alguno ya presente en el directorio principal del usuario
sshd_config	El archivo de configuración para el demonio sshd
ssh_host_dsa_key	Clave privada DSA usada por el demonio sshd
ssh_host_dsa_key.pub	Clave pública DSA usada por el demonio sshd
ssh_host_key	Clave privada RSA usada por sshd para la versión 1 del protocolo SSH
ssh_host_key.pub	Clave pública RSA usada por sshd para la versión 1 del protocolo SSH.
ssh_host_rsa_key	Clave privada RSA usada por sshd para la versión 2 del protocolo SSH.
ssh_host_rsa_key.pub	Clave pública RSA usada por sshd para la versión 2 del protocolo SSH

Algunos Parámetros del Archivo de configuración sshd_config

Port 34567	Puerto de escucha de peticiones, por defecto es el 22. Se recomienda utilizar un puerto diferente para una mejor seguridad. (Mayor a 1024)
ListenAddress 0.0.0.0	Especificación porque direcciones locales se escucha las peticiones
Protocol 2	Hay 2 versiones del protocolo SSH, (la versión 2 es más segura, dado que la versión 1 cuenta con algoritmos de cifrado sin soporte en la actualidad)
hostKey /etc/ssh/ssh_host_key	Archivo que contiene las llaves privadas del host para SSH versión 1
HostKey /etc/ssh/ssh_host_rsa_key HostKey /etc/ssh/ssh_host_dsa_key	Archivo que contiene las llaves privadas del host para SSH versión 2
LoginGraceTime 2m	Cantidad de tiempo que la pantalla del login estará disponible para ingresar el nombre de usuario y contraseña, antes de cerrarse (ej: 23, 2m)
StrictModes yes	Verifica los modos y permisos de los archivos de los usuarios y el directorio HOME del usuario antes de aceptar la sesión
PermitRootLogin no	Permite deshabilitar el logeo del sistema a través de la cuenta root
MaxAuthTries 6	Número máximo de intentos de conexión fallidos, antes de cerrarse la ventana
MaxSessions 10	Número máximo de sesiones permitidas
RSAAuthentication yes	Permitir la autenticación por RSA (solo para versión 1)
PubkeyAuthentication yes	Permitir la autenticación por RSA clave pública (solo para versión 2)

AuthorizedKeysFile .ssh/authorized_keys	Ruta donde se guardan las llaves públicas autorizadas
KerberosAuthentication no KerberosOrLocalPasswd yes KerberosTicketCleanup yes #KerberosGetAFSToken no	Kerberos es un protocolo de autenticación, basado en criptográfica de clave simétrica
GSSAPIAuthentication no GSSAPICleanupCredentials yes	Interfaz de programación de aplicaciones para usar sistemas seguros
X11Forwarding no	No permitir acceder con un entorno gráfico

Ejemplo de un Blindado de servicio SSH.
Fichero sshd_conf

Port	34567
Protocol	2
PermitRootLogin	no
MaxAuthTries	2
StrictModes	yes
X11Forwarding	no
LoginGraceTime	30

Cliente SSH

Archivos de Configuración del lado del Cliente
Se encuentran almacenados en el directorio de trabajo (home) de cada usuario

authorized_keys	Este archivo contiene una lista de claves públicas autorizadas. Cuando un cliente se conecta al servidor, el servidor autentica al cliente chequeando su clave pública firmada almacenada dentro de este archivo
id_dsa	Clave privada DSA del usuario
id_dsa.pub	Clave pública DSA del usuario
id_rsa	Clave privada RSA usada por ssh para la versión 2 del protocolo SSH.
id_rsa.pub	Clave pública RSA usada por ssh para la versión 2 del protocolo SSH
identity	Clave privada RSA usada por ssh para la versión 1 del protocolo SSH.
identity.pub	Clave pública RSA usada por ssh para la versión 1 del protocolo SSH
known_hosts	Este archivo contiene las claves de host DSA de los servidores SSH a los cuales el usuario ha accedido. Este archivo es muy importante para asegurar que el cliente SSH está conectado al servidor SSH correcto

Conectarse a un servidor remoto desde un Cliente

Sentencia: ssh [opciones] UsuarioRemoto@IpServidorRemoto [comando]

- ssh 192.168.1.1 -l leal
- ssh leal@192.168.1.1
  Petición de conexión al servidor SSH 192.168.1.1, con el usuario leal
- ssh -X leal@192.168.1.1
  Petición de conexión al servidor SSH, utilizado el servidor gráfico
- ssh –p34567 leal@192.168.1.1
  Petición de conexión al servidor SSH, utilizado un puerto de conexión diferente
- ssh –p34567 leal@192.168.1.1 'ls -l'
  Petición de conexión al servidor SSH, para ejecutar un comando

Para realizar transferencia de archivos desde un cliente y servidor remoto

SCP: Shell Secure Copy

- Permite la transferencia segura de archivos entre un equipo local y un equipo remoto
- Sentencia: scp <RutaRecursoOrigen> <RutaRecursoDestino>
- La ruta del recurso remoto comprende UsuarioRemoto@IpServidor:RutaRecursoRemoto

- scp leal@192.168.1.1:/home/leal/archivo.txt **.**
  Copia desde un servidor remoto el archivo "archivo.txt" a la ruta actual
- scp /home/usuario/archivo.txt leal@192.168.1.1:
  scp archivo.txt leal@192.168.1.1:
  Copia el archivo de la ruta actual, al directorio personal de "leal" en el servidor remoto
- scp –P34567 archivo.txt leal@192.168.1.1:
  Copia especificando el puerto de escucha del servidor remoto
- scp -r leal@192.168.1.1:/carpeta **.**
  Copia recursiva de la carpeta "carpeta" y todo el contenido de la misma

SFTP: Security File Transfer Protocol

- Permite la transferencia segura de archivos en forma interactiva, permite navegar en el servidor.
- Proceso similar al servicio FTP
- Sentencia: sftp UsuarioRemoto@IpServidor

- sftp leal@192.168.1.1
  Petición de conexión al servidor 192.168.1.1, como usuario leal
- sftp –o port=34567 leal@192.168.1.1
  Petición de conexión especificando el puerto de escucha del servidor remoto
- sftp leal@192.168.1.1:/home/leal/prueba.txt **.**
  Copia del servidor remoto el archivo "prueba.txt" al directorio actual

Se puede acceder desde algunos navegadores, o desde clientes gráficos
  - sftp://leal@192.168.1.3
  - Filezilla

Who: Lista los usuarios que están conectados al sistema
Netstat: Lista las conexiones activas del ordenador, tanto entrantes como salientes

Autenticación entre equipos remotos

- Evitar que al conectarse a un equipo remoto solicite la contraseña del usuario.
- Se generan un par de claves RSA y/o DSA para autenticarse
- Algunas distribuciones Linux solo permiten la clave de cifrado RSA

Desde el equipo local y usuario local
- ssh-keygen –t rsa
  Se crea un par de claves, una pública y una privada: *id_dsa* e *id_rsa.pub*
- ssh-copy-id –i id_rsa.pub leal@192.168.1.1
  Se comparte la clave pública con el usuario remoto
  Copiar el contenido del fichero *id_dsa.pub* al fichero *authorized_keys* del equipo remoto