PROTOCOLOS DE ACCESO REMOTO

Descargar PDF

INTRODUCCIÓN

Los servicios de acceso y control remoto permiten, mediante la utilización de determinadas aplicaciones de software, establecer conexiones con equipos a distancia y administrarlos de manera centralizada sin necesidad de acceder directamente a ellos. Estos servicios siguen el **modelo cliente-servidor**. De esta manera la máquina a acceder y controlar remotamente deberá tener instalada una aplicación de tipo servidor y la máquina con la que nos conectamos una aplicación de tipo cliente.

APLICACIONES

Las aplicaciones más frecuentes del acceso remoto son:

- **Administración remota de servidores**
  En inglés RAS (Remote Access Server).
  - Servidores apilados en un rack (armario de comunicaciones) que no disponen de monitor, teclado o ratón.
  - Servidores que no están físicamente presentes por encontrarse en una planta diferente del edificio a nuestra red local.
  - Servidores que estén alojados en otras redes como una nube de Internet, etc.
- **Acceso a datos de un servidor**
  Por ejemplo, usuarios que acceden a datos de un servidor desde otras instalaciones.
- **Acceso a aplicaciones de un servidor**
  Por ejemplo, usuarios que ejecutan aplicaciones en un servidor desde otras instalaciones.
- **Asistencia remota a usuarios**
  Por ejemplo, técnicos que controlan el ordenador de un usuario para modificar su configuración o resolver un problema.
- **Control de dispositivos**
  Por ejemplo, obtener las imágenes de una cámara de vigilancia, apagar o encender un determinado dispositivo en una casa con domótica, etc.

CONDICIONES

Para poder utilizar un servicio de acceso remoto se debe tener en cuenta algunos aspectos:

- **Garantizar la conectividad entre servidor y cliente**
  El servidor debe ser accesible a nivel de red por parte del cliente, aunque el servidor se encuentre en una red diferente.
- **Garantizar la seguridad del acceso al servidor**
  La funcionalidad de los servicios de acceso remoto parece útil e inofensiva, pero puede tener consecuencias impredecibles si no se lleva a cabo con unas condiciones de seguridad bien definidas. Cualquier agujero de seguridad que presenten estos servicios puede permitir el acceso de terceros no deseados a informaciones confidenciales. Por ejemplo, se pueden establecer políticas de control de acceso al servidor mediante la configuración de usuarios y permisos de ejecución en el servidor o establecer condiciones de Firewall para que solamente determinados equipos puedan hacer uso del servicio de acceso remoto.
- **Garantizar la capacidad de la conexión**
  Se debe garantizar el ancho de banda suficiente para trabajar remotamente sin tener paradas o cortes en la conectividad. Por tanto, se debe garantizar una latencia adecuada para el tipo de trabajo remoto que se quiere realizar.

TIPOS ACCESO REMOTO

Básicamente existen dos tipos de acceso remoto, dependiendo de si se utiliza una interfaz de texto o una interfaz gráfica.

INTERFAZ TEXTO

Los servicios de acceso remoto con interfaz de texto o consola normalmente utilizan protocolos de la pila TCP/IP como **telnet**, **r-commands** o **ssh**. No requieren un gran ancho de banda y, por tanto, funcionan adecuadamente con latencias altas.

TELNET

Es un protocolo muy antiguo que abre sesiones de terminal remotas enviando todos los datos sin encriptar, incluso los datos de autenticación, y por tanto son susceptibles de ser capturadas con un sniffer de red como *WireShark*. Esta herramienta se considera obsoleta y no se aconseja utilizarla. Sin embargo, la herramienta telnet es muy útil para entender el funcionamiento de los protocolos más importantes de la capa de aplicación de red como *http* y *https* (servicio web), *ftp* (servicio de transferencia de archivos) y *smtp*, *pop* e *imap* (servicio de correo electrónico).

Telnet utiliza los siguientes **puertos**:

Aplicación	**Puerto**
Servidor	TCP 23
Cliente	TCP efímero

R-COMMANDS

Es una suite de herramientas diseñadas en la universidad de Berkeley para posibilitar a usuarios Unix acceder a sistemas Unix de forma remota y sencilla. Algunas de las herramientas de la suite son *rcp*, *rexec*, *rlogin*, *rsh*, ... Se trata también de conexiones sin encriptar por lo que también están en desuso.

R-commands utiliza los siguientes **puertos**:

Aplicación	**Daemon**	**Puerto**
rcp	rshd	TCP 514
rexec	rexecd	TCP 512
rlogin	rlogind	TCP 513
rsh	rshd	TCP 514
rstat	rstatd	UDP
ruptime	rwhod	UDP 513
rwho	rwhod	UDP 513

SSH

Es un protocolo de acceso remoto que se desarrolló como alternativa segura a *telnet* y *r-commands*. **SSH** (Secure Shell) es un protocolo de red que permite establecer una conexión segura entre dos equipos, normalmente entre un cliente y un servidor, para realizar tareas de administración remota o transferencia de archivos. Actualmente, es el protocolo más utilizado para abrir sesiones de terminal remotas en modo texto. Con *ssh* se establece una sesión remota con un equipo de manera que toda la información se envía encriptada, incluida la información de autenticación. Este protocolo utiliza certificados digitales para controlar el acceso y asegurar que la información viaja encriptada. Además, *ssh* permite el uso de túneles para encriptar y transmitir la información de otras aplicaciones.

SSH utiliza los siguientes **puertos**:

Aplicación	**Puerto**
Servidor	TCP 2
Cliente	TCP efímero

WINRM

Es la implementación de Microsoft **Windows** del **protocolo WS-Management** (Web Service for Management), un protocolo estándar basado en el protocolo SOAP (Simple Object Access Protocol) que permite que interoperen el hardware y los sistemas operativos de diferentes proveedores.

La implementación actual del protocolo WS-Management se basa en las siguientes especificaciones estándar: HTTPS, SOAP mediante HTTP (perfil WS-I), SOAP 1.2, WS-Addressing, WS-Transfer, WS-Enumeration y WS-Eventing. Para obtener más información sobre los estándares WS-Management y los esquemas XML puedes consultar la siguiente url https://dmtf.org/standards/wsman.

Se pueden ejecutar órdenes de administración remota en equipos Windows con la consola WinRS o estableciendo una conexión remota mediante PowerShell. Para establecer la conexión es evidente que tanto en el servidor como en el cliente el servicio WinRM debe estar en funcionamiento.

WinRM utiliza los siguientes **puertos**:

Aplicación	**Puerto**
Servidor	HTTP TCP 5985
Servidor	HTTPS TCP 5986
Cliente	TCP efímero

INTERFAZ GRÁFICA

Los servicios de acceso remoto con interfaz gráfica son aplicaciones para establecer sesiones gráficas. Normalmente son muy dependientes del sistema operativo, es decir, cliente y servidor deben tener el mismo tipo de sistema operativo. Requieren un gran ancho de banda y, por tanto, solo funcionan adecuadamente con latencias bajas.

RDP

El protocolo **RDP** (Remote Desktop Protocol) es **propietario de Microsoft** y permite la transmisión de la información gráficamente entre un cliente y un servidor Windows. La información gráfica que genera el servidor es convertida a un formato propio del protocolo RDP y enviada al cliente. En el cliente, la información del protocolo RDP es interpretada y se reconstruye la imagen gráfica para mostrarla en su terminal. En el cliente, las pulsaciones de teclado, movimiento y pulsaciones del ratón son redirigidas al servidor también en formato RDP, para que el servidor las ejecute. La información transmitida entre cliente y servidor está cifrada y comprimida para mejorar la seguridad y la latencia de la conexión.

Las aplicaciones más características de RDP son los servidores **RDS** (Remote Desktop Services) antes **Terminal Server** en el servidor Windows y los clientes **RDC** (Remote Desktop Connections) en los sistemas Windows de tipo desktop, como Windows 10 o Windows 11. En Linux existe también una versión open source de servidor RDP llamada **XRPD**, se puede utilizar cualquier cliente RDP como *FreeRDP*, *rdesktop* o incluso *RDC de Windows* para realizar la conexión.

RDP utiliza los siguientes **puertos**:

Aplicación	**Puerto**
Servidor	TCP,UDP 3389
Cliente	TCP,UPD efímero

X WINDOW

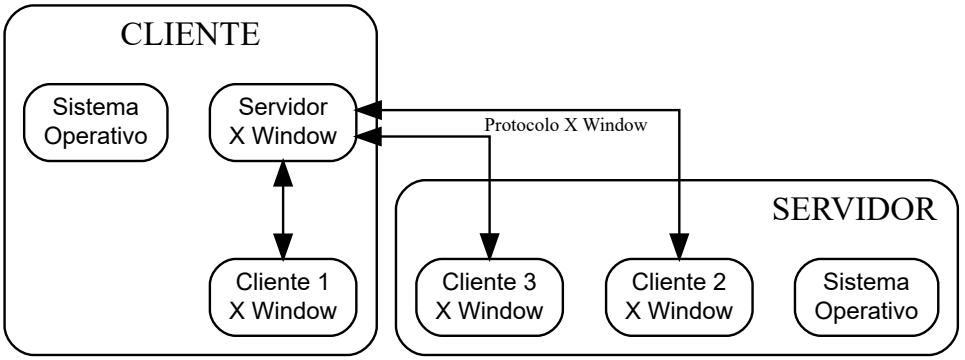
Este sistema implementa las funciones necesarias para controlar los gráficos como las ventanas, el ratón y el teclado en un ordenador. **X Window** se utiliza en la mayor parte de las versiones de Unix y sus derivados para implementar la interfaz gráfica. La **versión actual** es la 11, por lo que se dice **X11**. Entornos de escritorio tan conocidos en **GNU/Linux** como *gnome*, *kde* o *xfce* utilizan internamente X Window a pesar de que son escritorios con apariencia diferente y que cada uno tiene sus particularidades.

Una de las características de X Window es que es independiente del sistema operativo empleado, ya que se trata de una capa de aplicación que además se pensó para trabajar en red. Con el concepto de transparencia de red, X Window puede separar la estación donde se representa la interfaz gráfica de la estación donde se ejecuta la aplicación, evidentemente también puede tratarse de la misma estación. En la terminología X Window, el software que permite dibujar las ventanas e interactuar con ellas se llama servidor X y los programas que envían las peticiones para dibujar las ventanas y reciben las interacciones con ellas se llaman cliente X.

La comunicación entre las aplicaciones clientes X Window y el servidor X Window se realiza sin ningún tipo de encriptación, por lo que si se quiere dotar de seguridad a la transmisión será necesario emplear, por ejemplo, SSH para tunelizar el tráfico X Window o bien utilizar sistemas VPN.

X Window

En el ejemplo, el host cliente tiene instalado X Window en modo servidor. Una aplicación cliente 1 del propio host está utilizando el servidor X Window para mostrar información gráfica en la pantalla. Al mismo tiempo, el host cliente está ejecutando las aplicaciones cliente 2 y cliente 3 en el servidor, pero estas aplicaciones interactúan con el servidor X Window instalado en el host cliente, por tanto dibujan y muestran la información gráfica en el host cliente.



X Windows utiliza los siguientes **puertos**:

Aplicación	Puerto
Servidor	TCP 6000+N (número de pantalla)
Cliente	TCP efímero

XDMCP

Normalmente al iniciar un ordenador **GNU/Linux** aparece una pantalla gráfica para realizar el login del sistema. Este programa, que en caso de autenticar correctamente al usuario carga todo el escritorio, es un administrador de pantalla. El **protocolo** que controla el administrador de pantalla se llama **XDMCP** (X display manager control protocol) y funciona sobre un servidor X Window. Hay diferentes implementaciones de este programa, las más usuales son: *gdm* (gnome), *kdm* (kde) y *xdm* (genérico).

En la mayor parte de los casos el administrador de pantallas solamente permite gestionar un servidor X Windows local, pero gracias al protocolo XDMCP también se puede gestionar servidor X11 remotos. En este caso, un equipo puede mostrar la pantalla de login y después mostrar el escritorio completo, no solo la ventana de una aplicación concreta. La configuración de XDMCP puede ser muy compleja.

XDMCP utiliza los siguientes **puertos**:

Aplicación	Puerto
Servidor	TCP 177
Cliente	TCP efímero

VNC

El sistema **VNC** (Virtual Network Computing) permite mostrar y controlar gráficamente un escritorio remoto. Utiliza el **protocolo RFB** (Remote Frame Buffer Protocol) que se limita a transmitir cada uno de los píxeles del buffer gráfico de la memoria de vídeo del servidor al cliente y los eventos de teclado y ratón del cliente al servidor. Las últimas versiones de RFB han sido ampliadas para soportar nuevas funcionalidades como, por ejemplo, la transferencia de archivos entre el cliente y el servidor.

Por defecto, RFB es un protocolo no seguro, lo que significa que las transmisiones se realizan sin cifrar. La única transmisión que se cifra es la contraseña, aunque tanto la clave de cifrado como la contraseña cifrada pueden ser conseguidas mediante un sniffer y, por tanto, pueden ser descubiertas. Dado que RFB es un protocolo no seguro, si se utiliza VNC en una red que no es de confianza, será necesario combinar su uso con un túnel SSH o con una VPN.

VNC es independiente de la plataforma, hay cliente y servidor para muchos tipos de sistemas operativos y también para Java. Un servidor VNC puede atender a varios clientes al mismo tiempo. El uso del protocolo RFB hace que VNC necesite un ancho de banda grande, por lo que se han desarrollado varios métodos para codificar los píxeles de una manera más comprimida (encoders). VNC permite negociar al cliente y al servidor qué tipo de encoder utilizará en la conexión. El encoder por defecto transmite cada uno de los píxeles de la memoria de vídeo que forman la pantalla completa, después solamente envía los píxeles que cambian.

Una característica fundamental del sistema VNC es que el cliente no conserva ningún estado de la conexión. Así es posible cerrar un cliente y volverlo a abrir, incluso en otro equipo, como si se tratara de un monitor virtual.

Hay que recalcar que tanto VNC como RFB se distribuyen con licencia libre de tipo GPL, es decir, que su código se puede reutilizar y modificar, lo que ha hecho que haya numerosas versiones.

Versiones VNC:

Versión	**Descripción**
VNC	Es el software original. Solo se utiliza como referencia y pruebas de compatibilidad.
TightVNC	Gratis con licencia GPL. Tiene encoders muy avanzados que permiten utilizar menos ancho de banda y además permite la transmisión de archivos entre cliente y servidor.
ReaLVNC	Es la versión desarrollada por algunos de los creadores originales. Hay versión gratuita y empresarial.
UltraVNC	Ofrece una función de transmisión de archivos, ha mejorado los encoders y tiene una función de chat. Además, permite controlar una sola ventana en lugar de todo el escritorio.
xVNC	Configura un servidor VNC para conectar internamente con el sistema X Windows. Por tanto, en lugar de que RFB utilice el buffer de vídeo, utiliza el sistema X Windows.

VNC utiliza los siguientes **puertos**:

Aplicación	**Puerto**
Servidor	TCP 5900+N (número de pantalla)
Servidor WEB con applet Java	TCP 5800+N (número de pantalla)

Aplicación	Puerto
Cliente en modo listening	TCP 5500 El servidor es el que inicia la conexión.
Cliente	TCP efímero

NX

Este sistema permite mostrar y controlar gráficamente un escritorio remoto. NX está basado en el **protocolo DXPC** (Differential X Protocol Compressor) **para un sistema X Windows**. Con NX se realiza una compresión del protocolo X Windows, lo que mejora notablemente la latencia. Además, se utiliza el protocolo SSH para encriptar las transmisiones, mejorando la seguridad. NX es una tecnología propietaria de la empresa NoMachine, aunque la empresa ha distribuido con licencia GPL el núcleo de compresión del protocolo, lo que ha permitido que se desarrollen algunas distribuciones freeware como *FreeNX* o *X2Go*.

NX utiliza los siguientes **puertos**:

Aplicación	Puerto
Servidor	TCP 4000
Cliente	TCP efímero

INTERFAZ ESPECÍFICA

Además de los protocolos y aplicaciones de uso genérico para el acceso remoto basadas en interfaces de texto y/o gráficas existen aplicaciones diseñadas específicamente para la administración remota de dispositivos o servicios concretos. Por ejemplo, impresoras, cámaras de vigilancia, dispositivos de domótica, puntos de acceso, switches, routers, servicios de resolución de nombres, samba, etc..

En el caso de los dispositivos se suelen implementar como programas instalados localmente en el ordenador del administrador o interfaces web proporcionadas por el propio dispositivo a las que se accede mediante un navegador. Suelen estar diseñadas a medida para cada tipo de dispositivo permitiendo su gestión remota de forma sencilla.

En el caso de los servicios es posible administrar remotamente los servicios instalados en un servidor. Por ejemplo, *Apache* (servidor web), *BIND* (servidor DNS) o *Samba* (servidor CIFS). Algunos de los más utilizados son:

Herramienta	Descripción Básica	Protocolo	Puerto
Cockpit	Administración de sistema, hardware y contenedores. Ligero y moderno.	HTTPS	**9090**
Webmin	Configuración profunda de archivos del sistema y servicios de red.	HTTPS	**10000**
ISPConfig	Gestión multi-servidor de hosting, correo y DNS de código abierto.	HTTPS	**8080**

Herramienta	Descripción Básica	Protocolo	Puerto
aaPanel	Panel modular y visual para despliegue rápido de sitios y bases de datos.	HTTPS	**8888**
Virtualmin	Especializado en hosting virtual y dominios (basado en Webmin).	HTTPS	**10000**

TABLA RESUMEN

Interfaz	Protocolo	Seguridad	Puerto	Características
Texto	**Telnet**	No cifrado	TCP 23	Protocolo antiguo, no seguro. Útil solo para pruebas básicas
Texto	**RCommands**	No cifrado	TCP 512–514	Suite Unix clásica (rsh, rlogin…). Obsoleta e insegura
Texto	**SSH**	Cifrado y autenticado	TCP 22	Acceso remoto seguro. Permite túneles y uso intensivo en servidores
Texto	**WinRM**	HTTPS	HTTP 5985 HTTPS 5986	Administración remota Windows mediante PowerShell/WS-Management.
Gráfica	**RDP**	Cifrado y compresión	TCP,UDP 3389	Escritorio remoto Windows. Eficiente y ampliamente utilizado
Gráfica	**XWindow (X11)**	No cifrado	TCP 6000+N	Sistema gráfico de Linux. Requiere túnel SSH para seguridad.
Gráfica	**XDMCP**	No cifrado	TCP 177	Permite login remoto y sesiones gráficas completas. Configuración compleja.
Gráfica	**VNC (RFB)**	No cifrado	TCP 5900+N TCP 5800+N TCP 5500	Escritorio remoto multiplataforma. Requiere cifrado adicional.
Gráfica	**NX**	Cifrado (SSH) + compresión	TCP 4000	Escritorio remoto muy optimizado. Ideal para conexiones lentas.
Específica	**Cockpit**	HTTPS	TCP 9090	Administración ligera de sistema Linux.
Específica	**Webmin**	HTTPS	TCP 10000	Admistración profunda de servicios y sistema Linux
Específica	**ISPConfig**	HTTPS	TCP 8080	Gestión de hosting, correo y DNS en entornos multiservidor.

Interfaz	Protocolo	Seguridad	Puerto	Características
Específica	**aaPanel**	HTTPS	TCP 8888	Panel visual para despliegue de sitios y bases de datos.
Específica	**Virtualmin**	HTTPS	TCP 10000	Administración de hosting avanzada, basado en Webmin.