

TÉCNICAS DE ACCESO REMOTO

Raúl Arribas Redondo

SAD

Contenido

1	SECURE SHELL.....	2
1.1	Instala el servicio SSH en la máquina dmzd	2
1.2	Configura el servicio para que se puedan realizar conexiones como root y que esté permitida la autenticación por contraseña	3
1.3	Verifica que está correctamente configurado realizando una conexión SSH desde la máquina extra	3
1.4	Configurar SSH para conexiones desatendidas	4
1.5	Verifica que está correctamente configurado realizando una conexión SSH desde la máquina extb.....	5
1.6	Muestra los contenidos de los ficheros authorized-keys y known-host	7
2	Redes privadas Virtuales	9
2.1	Realiza una conexión entre los equipos exta y dmzc mediante OpenVPN usando validación TLS	9
2.2	Verifica la correcta conexión a través del túnel mediante un ping desde el cliente VPN al servidor VPN	13
2.3	Muestra la tablas de enrutamiento y las interfaces de red, tanto del cliente VPN como del servidor VPN.....	15
2.4	Ubica la página PruebaConexionVPN.html en el servidor VPN . Usa el túnel VPN previamente establecido para visualizar la página desde el navegador web del cliente VPN . 17	
2.5	Instala y configura un cliente VPN en la máquina virtual Windows 10 del entorno. .	18
2.6	Muestra la tablas de enrutamiento y las interfaces de red, tanto del cliente VPN Windows 10 como del servidor VPN.	18
2.7	Realiza una conexión al servidor usando el túnel VPN y visualiza la página PruebaConexionVPN.html desde un navegador web de la máquina Windows 10	18
3	Guacamole	22
3.1	Instala en el servidor VPN un servicio Guacamole	22
3.2	Configura dicho servicio para que se pueda acceder por SSH al servidor web de la máquina dmza	23
3.3	Utiliza algún mecanismo de cifrado para que la clave de acceso no se pueda leer en el fichero de configuración de Guacamole.	24
3.4	Verifica que la conexión funciona correctamente tras dicho cambio.	25

1 SECURE SHELL

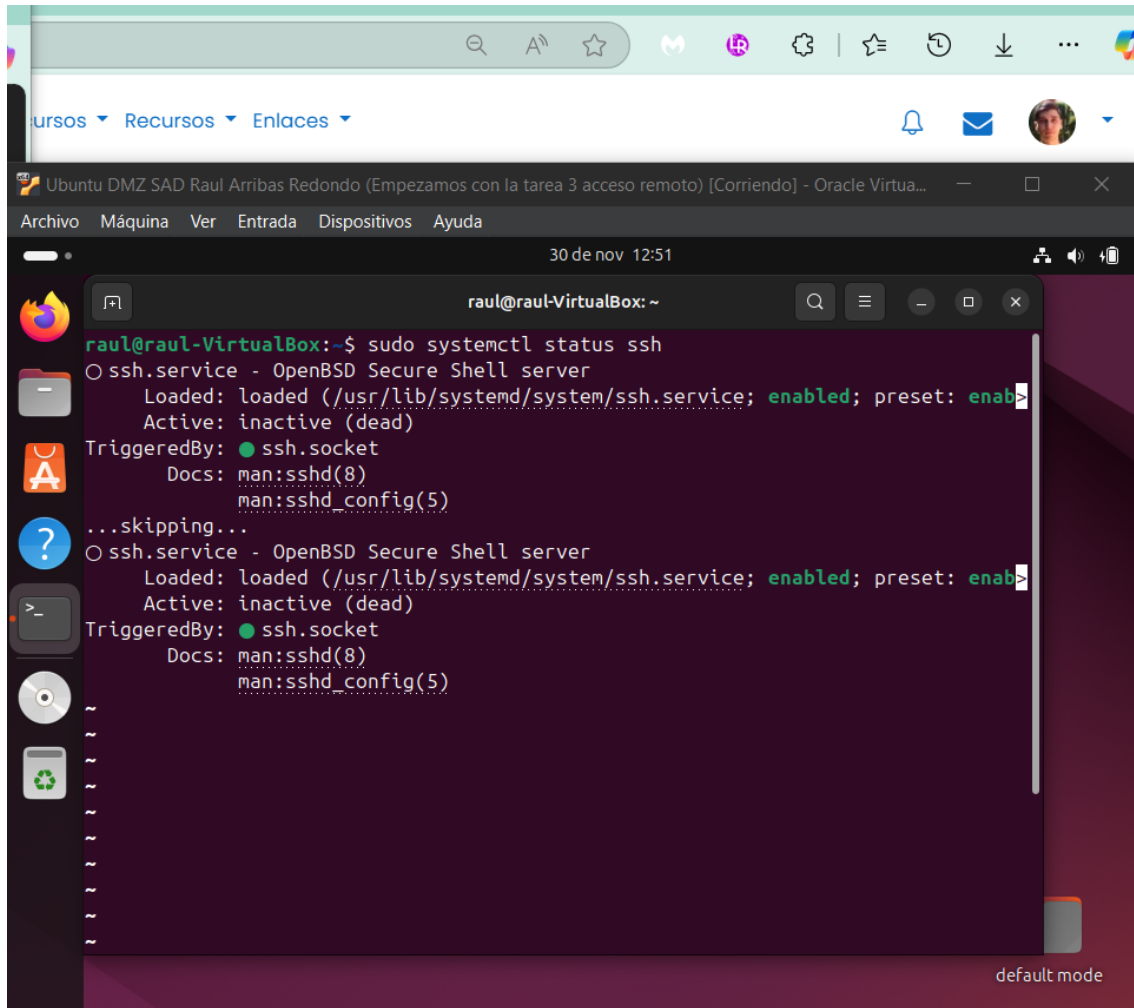
1.1 Instala el servicio SSH en la máquina **dmzd**

Para instalar el servicio SSH, tendremos que ejecutar los siguientes comandos:

- **Sudo apt install openssh-server -y**

Para comprobar que tenemos el servicio SSH activo:

- **Sudo systemctl status ssh**



```
raul@raul-VirtualBox:~$ sudo systemctl status ssh
O ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enab>
   Active: inactive (dead)
   TriggeredBy: ● ssh.socket
   Docs: man:sshd(8)
        man:sshd_config(5)
...skipping...
O ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enab>
   Active: inactive (dead)
   TriggeredBy: ● ssh.socket
   Docs: man:sshd(8)
        man:sshd_config(5)
```

1.2 Configura el servicio para que se puedan realizar conexiones como root y que esté permitida la autenticación por contraseña

Editamos el archivo de configuración de SSH:

- **sudo nano /etc/ssh/sshd_config**

Ponemos estas dos filas como se muestra a continuación:

- **PermitRootLogin yes**
- **PasswordAuthentication yes**
- **pubkeyAuthentication yes**

Guardamos los cambios y reiniciamos el servicio SSH para que se apliquen los cambios:

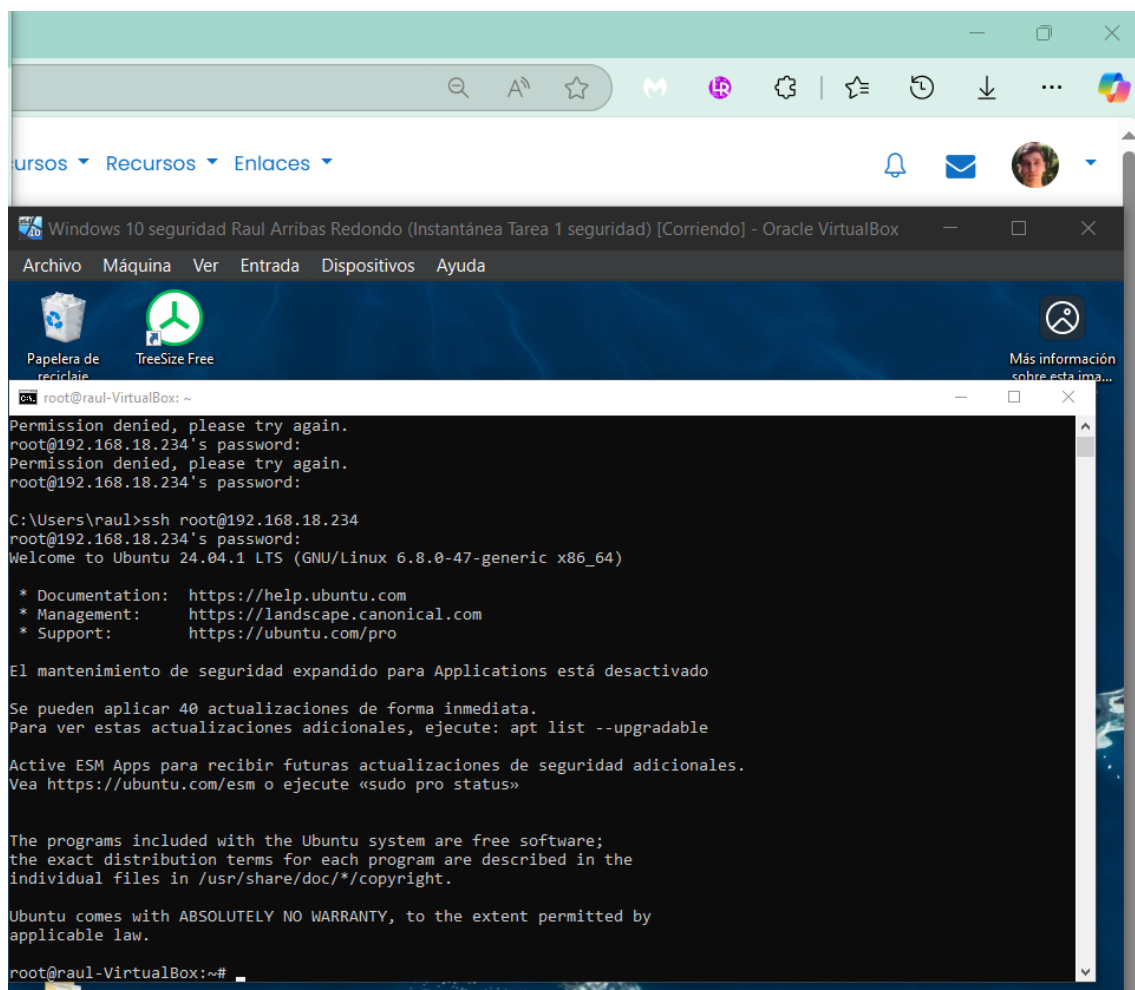
- **Sudo systemctl restart ssh**

1.3 Verifica que está correctamente configurado realizando una conexión SSH desde la máquina extra

Previamente tenemos que asignar una contraseña al usuario ROOT de la maquina de Ubuntu.

Nos conectamos a la maquina virtual Extra (es una maquina virtual con red en modo puente)

Ssh root@192.168.18.234



1.4 Configurar SSH para conexiones desatendidas

En la maquina de Windows, vamos a generar las claves de ssh:

- **ssh-keygen -t rsa -b 4096**

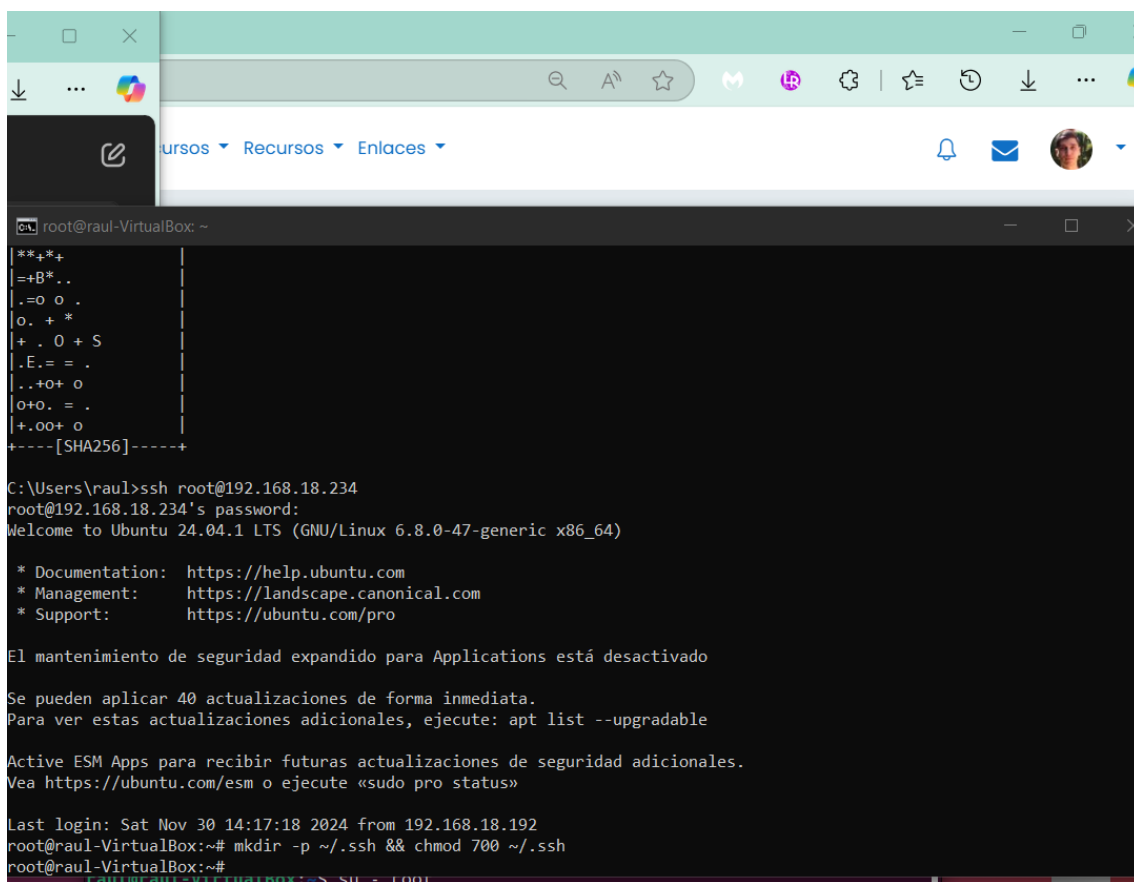
Copiamos el contenido de la clave publica que se ha generado, se encuentra en la ruta `c:\users\raul\.ssh\id_rsa.pub`

Ahora nos conectamos con el usuario root por SSH a la maquina servidor.

- **Ssh [root@192.168.18.234](https://192.168.18.234)**

Creamos la carpeta `.ssh` en la maquina Ubuntu:

- **mkdir -p ~/.ssh && chmod 700 ~/.ssh**



```
root@raul-VirtualBox: ~
|***+
|=+B*..
|. = 0 .
|. . + *
|+ . 0 + S
|.E. = .
|..+0+ 0
|0+0. = .
|+.00+ 0
+----[SHA256]-----+

C:\Users\raul>ssh root@192.168.18.234
root@192.168.18.234's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-47-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

El mantenimiento de seguridad expandido para Applications está desactivado

Se pueden aplicar 40 actualizaciones de forma inmediata.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

Active ESM Apps para recibir futuras actualizaciones de seguridad adicionales.
Vea https://ubuntu.com/esm o ejecute «sudo pro status»

Last login: Sat Nov 30 14:17:18 2024 from 192.168.18.192
root@raul-VirtualBox:~# mkdir -p ~/.ssh && chmod 700 ~/.ssh
root@raul-VirtualBox:~#
```

En la carpta que hemos creado, la editamos con

- **nano ~/.ssh/authorized_keys**

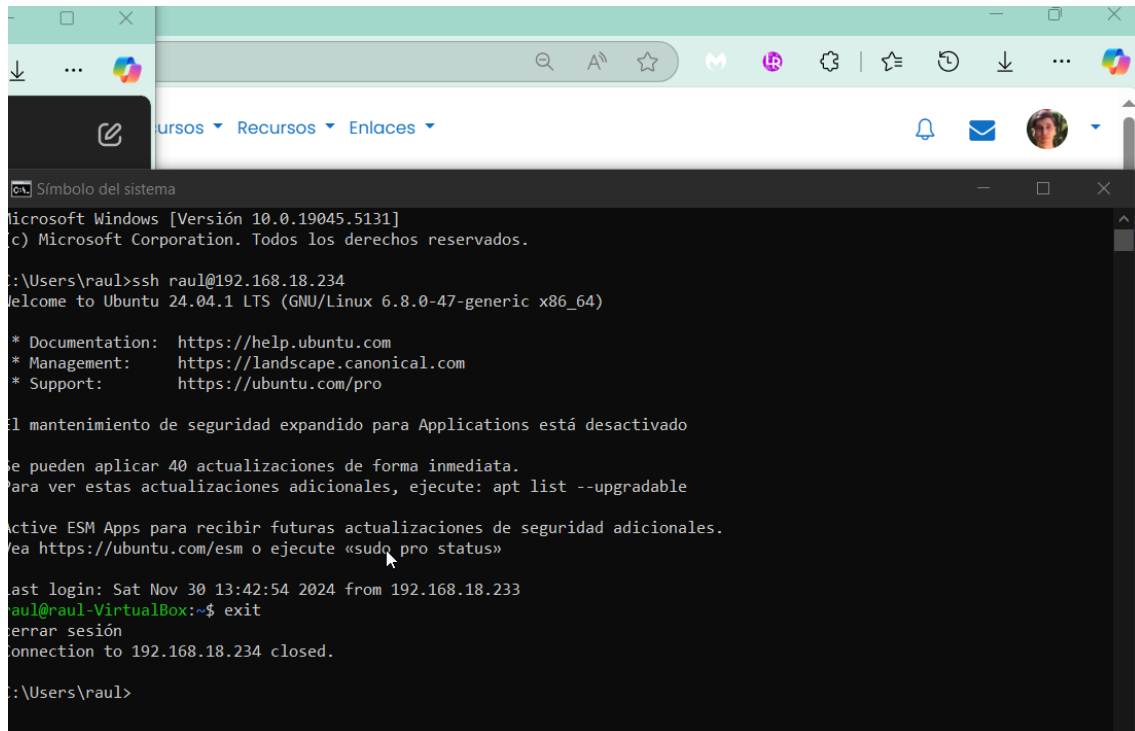
Y copiamos el contenido de la clave publica que nos ha generado antes (el contenido de esta clave se encuentra en: `c:\users\raul\.ssh\id_rsa.pub`)

Una vez hemos copiado el contenido de `id_rsa.pub` a `authorized_key`, tenemos que asignar los permisos :

- **chmod 600 ~/.ssh/authorized_keys**

1.5 Verifica que está correctamente configurado realizando una conexión SSH desde la máquina extb

Ahora nos vamos a poder conectar al usuario raul por ssh sin que nos pida la contraseña. En caso de que queramos acceder con el usuario root sin contraseña tenemos que agregar la clave publica en `authorized_keys` desde el usuario de root en Ubuntu.



```
Microsoft Windows [Versión 10.0.19045.5131]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\raul>ssh raul@192.168.18.234
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-47-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

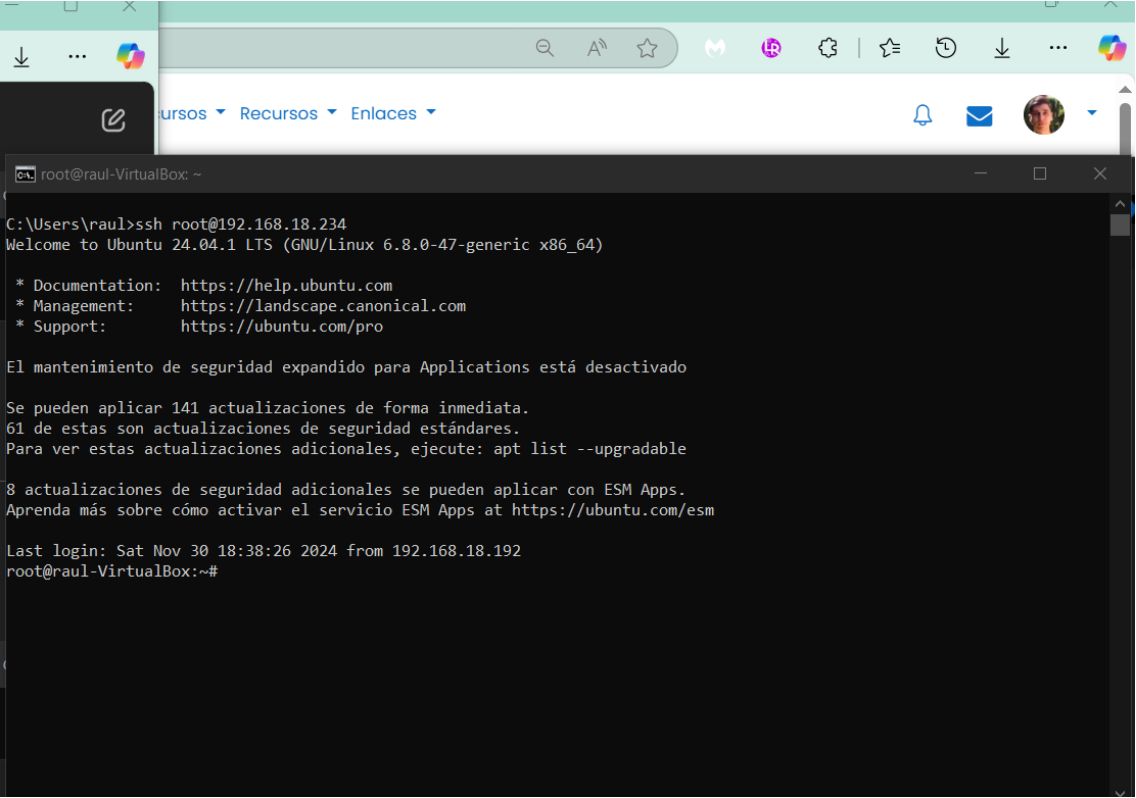
El mantenimiento de seguridad expandido para Applications está desactivado.
Se pueden aplicar 40 actualizaciones de forma inmediata.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

Active ESM Apps para recibir futuras actualizaciones de seguridad adicionales.
Vea https://ubuntu.com/esm o ejecute «sudo pro status»

Last login: Sat Nov 30 13:42:54 2024 from 192.168.18.233
raul@raul-VirtualBox:~$ exit
Cerrar sesión
Connection to 192.168.18.234 closed.

C:\Users\raul>
```

Aquí vemos el ejemplo al conectarnos con el usuario root que no nos pide contraseña:



The screenshot shows a terminal window titled 'root@raul-VirtualBox: ~'. The terminal output displays the SSH connection details and system information. The user is root, and the system is Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-47-generic x86_64). The terminal shows the following text:

```
C:\Users\raul>ssh root@192.168.18.234
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-47-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

El mantenimiento de seguridad expandido para Applications está desactivado

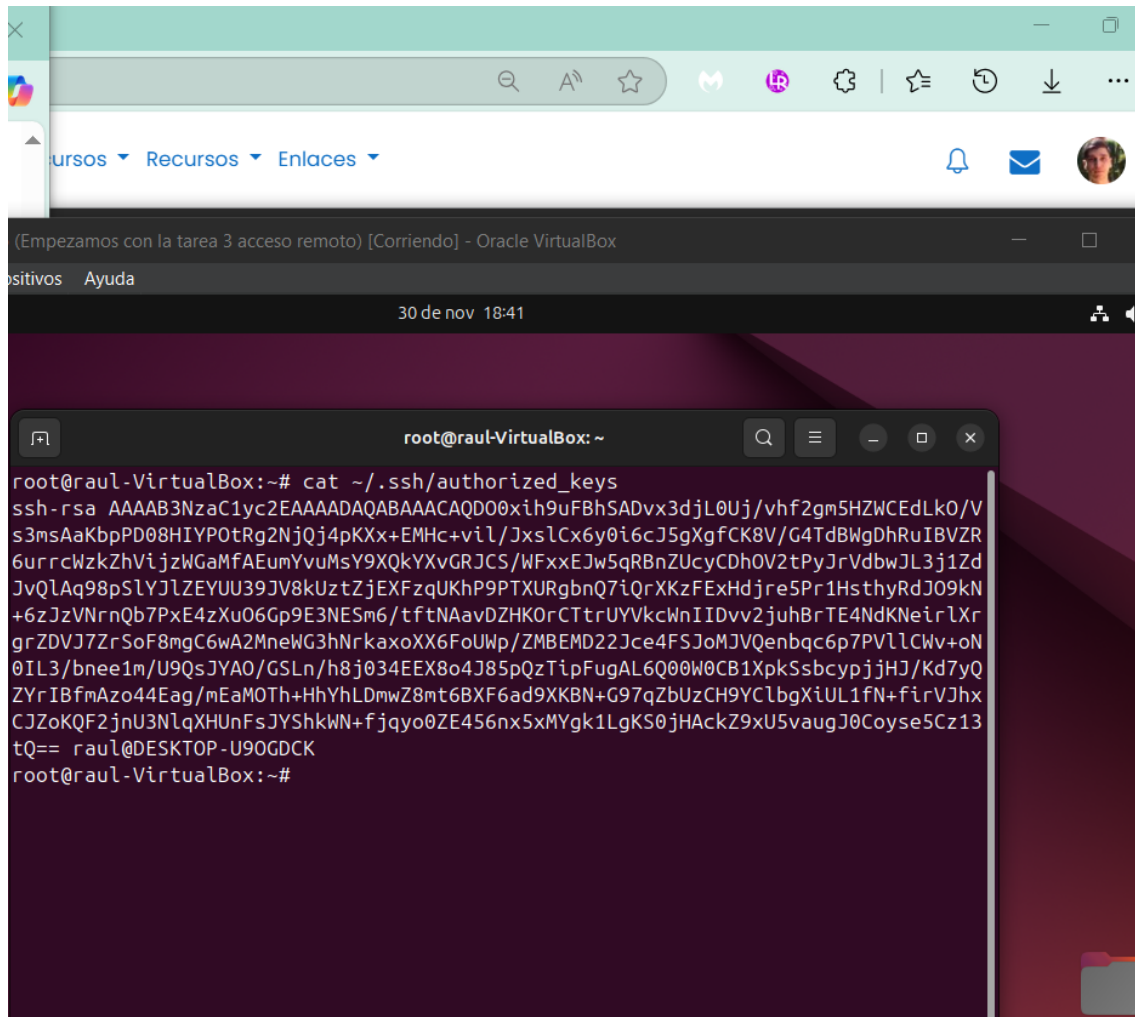
Se pueden aplicar 141 actualizaciones de forma inmediata.
61 de estas son actualizaciones de seguridad estándares.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

8 actualizaciones de seguridad adicionales se pueden aplicar con ESM Apps.
Aprenda más sobre cómo activar el servicio ESM Apps at https://ubuntu.com/esm

Last login: Sat Nov 30 18:38:26 2024 from 192.168.18.192
root@raul-VirtualBox:~#
```

1.6 Muestra los contenidos de los ficheros authorized-keys y known-host

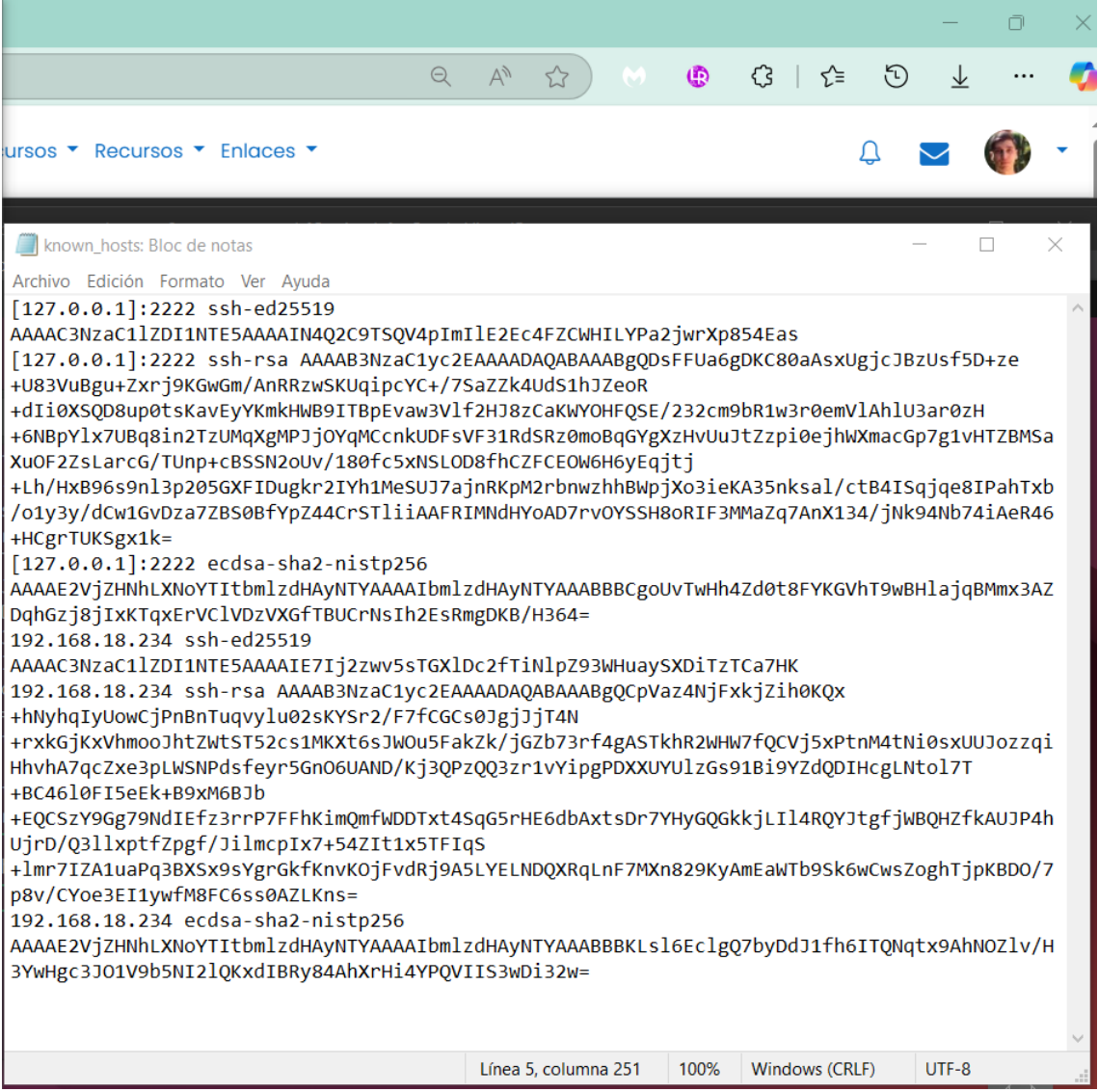
Mostramos el archivo de authorized key desde el usuario root:



The screenshot shows a web browser window at the top with tabs for 'Cursos', 'Recursos', and 'Enlaces'. Below the browser is a terminal window titled 'root@raul-VirtualBox: ~'. The terminal output shows the command 'cat ~/.ssh/authorized_keys' and its result, which is a long string of SSH keys. The terminal window has a title bar with search, menu, and window control icons. The background of the terminal is dark purple with a subtle pattern.

```
root@raul-VirtualBox:~# cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQAD00xih9uFBhSADvx3djl0Uj/vhf2gm5HZWCedLk0/V
s3msAaKbpPD08HIYP0tRg2NjQj4pKXx+EMHc+vll/JxslCx6y0i6cJ5gXgfCK8V/G4TdBWgDhRuIBVZR
6urrcWzkZhViJzWGaMfAEumYvuMsY9XQkYXvGRJCS/WFxxEJw5qRBnZUcyCDh0V2tPyJrVdbwJL3j1Zd
JvQLAq98pSLYJLZEYUU39JV8kUztZjEXFzqUKhP9PTXURgbnQ7iQrXKzFExHdjre5Pr1HsthyRdJ09kN
+6zJzVNrnQb7PxE4zXu06Gp9E3NESm6/tftNAavDZHK0rCTtrUYVkcWnIIDvv2juhBrTE4NdKNeirlXr
grZDVJ7ZrSoF8mgC6wA2MneWG3hNrKaxoXX6FoUWp/ZMBEMD22Jce4FSJoMJVQenbqc6p7PVllCwv+oN
0IL3/bnee1m/U9QsJYA0/GSLn/h8j034EEEX8o4J85pQzTipFugAL6Q00W0CB1XpkSsbccypjjHJ/Kd7yQ
ZYrIBfmAzo44Eag/mEaMOTh+HhYhLDmwZ8mt6BXF6ad9XKBN+G97qZbUzCH9YCLbgXiUL1fN+firVJhx
CJZoKQF2jnU3NlqXHUnFsJYShkWN+fjqyo0ZE456nx5xMYgk1LgKS0jHAckZ9xU5vaugJ0Coyse5Cz13
tQ== raul@DESKTOP-U90GDCK
root@raul-VirtualBox:~#
```

Y el archivo de known_host:



```
known_hosts: Bloc de notas
Archivo Edición Formato Ver Ayuda
[127.0.0.1]:2222 ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIN4Q2C9TSQV4pImIle2Ec4FZCWHILYPa2jwrXp854Eas
[127.0.0.1]:2222 ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDsFFUa6gDKC80aAsxUgjcJBzUsf5D+ze
+U83VuBgu+Zxrj9KGWgm/AnRRzWSKUqipcYC+/7SaZzk4UdS1hJZeoR
+dIi0XSQD8up0tsKavEYKmkHwB9ITBpEvaw3Vlf2HJ8zCaKWYOHFQSE/232cm9bR1w3r0emVlAhlu3ar0ZH
+6NBpYlx7UBq8in2TzUMqXgMPJjOYqMCcnkUDFsVF31RdSRz0moBqGYgXzHvUuJtZzpi0ejhWxmacGp7g1vHTZBMSa
XuOF2ZsLarcG/Tunp+cBSSN2oUv/180fc5xNSLOD8fhCZCFCEOW6H6yEqjtj
+Lh/HxB96s9nl3p205GXFIDugkr2IYh1MeSUJ7ajnRKPM2rbnwzhBwPjXo3ieKA35nksal/ctB4ISjqe8IPahTxb
/o1y3y/dCw1GvDza7ZBS0BfYpZ44CrSTliiAAFRIMNDHYoAD7rVOYSSH8oRIF3MMaZq7AnX134/jNk94Nb74iAeR46
+HCgrTUKSgx1k=
[127.0.0.1]:2222 ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBBCgoUvTwHh4Zd0t8FYKGVhT9wBH1ajqBMmx3AZ
DqhGzj8jIXKTqxErVClVDzVXGfTBUCrNsIh2EsRmgDKB/H364=
192.168.18.234 ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIE7Ij2zwv5sTGXlDc2fTiNlpZ93WHuaySXDITzTca7HK
192.168.18.234 ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCpVaz4NjFxfkjZih0KQx
+hNyhqIyUowCjPnBnTuqvylu02sKYSr2/F7fCGCs0JgjJt4N
+rxkGjKxVhmooJhtZwtST52cs1MKXt6sJW0u5FakZk/jGZb73rf4gASTkhr2WHW7fQCVj5xPtnM4tNi0sxUUJozzqi
HhvhA7qcZxe3pLWSNPdsfeyr5Gn06UAND/Kj3QPzQQ3zr1vYipgPDXXUYUlzGs91Bi9YZdQDIHcgLnto17T
+BC46l0FI5eEk+B9xM6Bjb
+EQCSzY9Gg79NdIEfz3rrP7FFhKimQmfWDDTtxt4SqG5rHE6dbAxtsDr7YHyGQgkkjLil4RQYJtgfjWBQHZfkAUJP4h
Ujrd/Q3llxptfZpgf/JilmcpIx7+54ZIt1x5TFIqs
+lmr7IZA1uaPq3BXSx9sYgrGkfKnvK0jFvdRj9A5LYELNDQXRqLnF7MXn829KyAmEawTb9Sk6wCwsZoghTjpKBDO/7
p8v/CYoe3EI1ywfM8FC6ss0AZLKns=
192.168.18.234 ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBKLs16EclgQ7byDdJ1fh6ITQNqtX9AhNOZlv/H
3YwHgc3JOIv9b5NI2lQKxdIBry84AhXrHi4YPQVIIS3wDi32w=
```

Línea 5, columna 251 100% Windows (CRLF) UTF-8

2 Redes privadas Virtuales

2.1 Realiza una conexión entre los equipos **exta** y **dmzc** mediante **OpenVPN** usando **validación TLS**

Vamos a realizar la instalación desde un script de Github, en este caso desde esta URL:

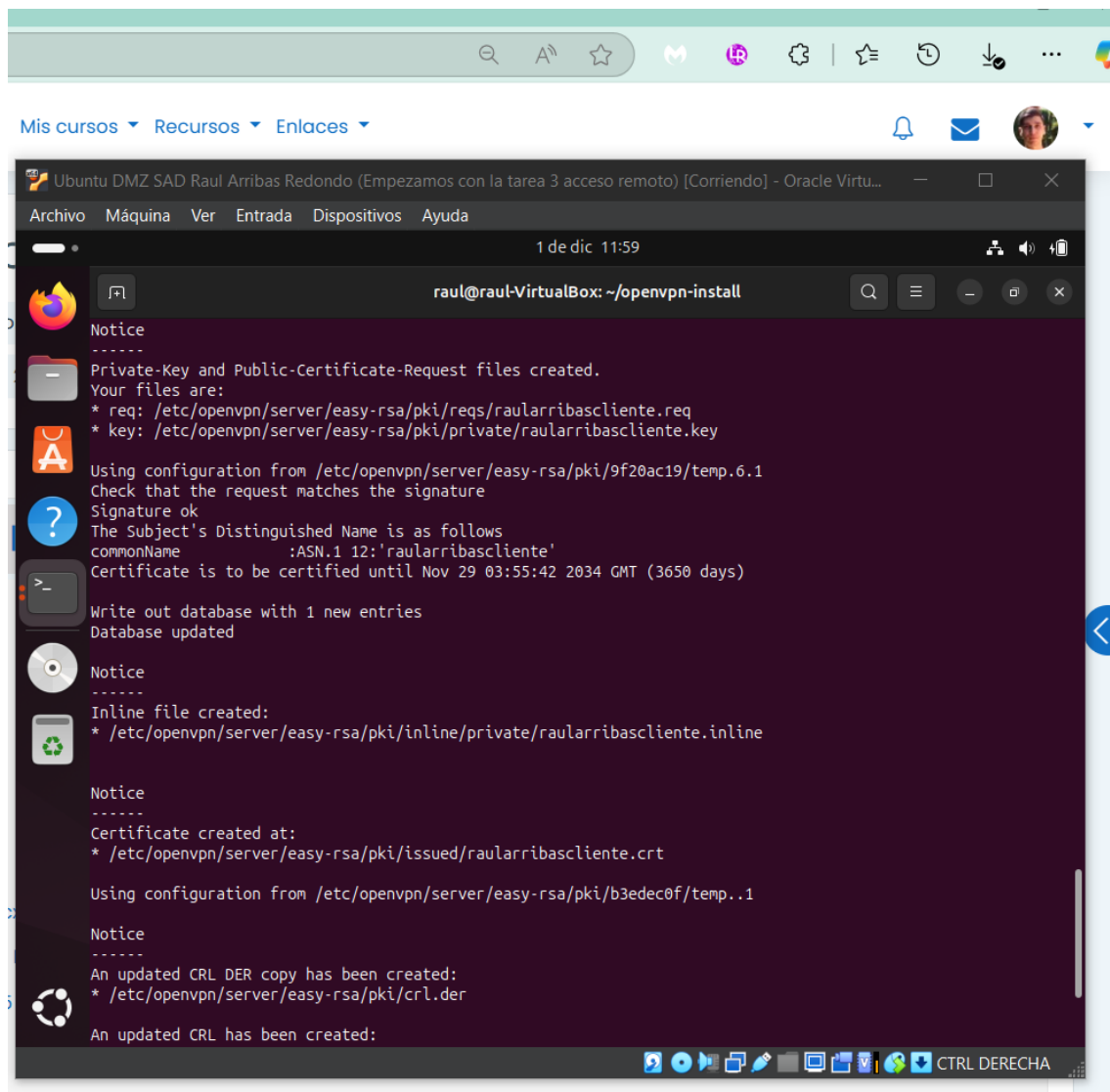
- **git clone** <https://github.com/Nyr/openvpn-install.git>

Para instalarlo le tenemos que dar permisos de ejecución y ejecutamos el instalador **.sh**

- **Sudo chmod +x openvpn-install.sh**
- **Sudo ./openvpn-install.sh**

Seguimos los pasos del instalador, seleccionamos todo por defecto. Agregamos un cliente con el nombre cliente1.

Cuando termine nos aparecerá que se ha creado el archivo **ovpn** del cliente1 que hemos generado, lo mostramos en la siguiente imagen:



```
An updated CRL DER copy has been created:
* /etc/openvpn/server/easy-rsa/pki/crl.der

An updated CRL has been created:
* /etc/openvpn/server/easy-rsa/pki/crl.pem

Created symlink /etc/systemd/system/multi-user.target.wants/openvpn-iptables.service → /etc/systemd/system/openvpn-iptables.service.
Created symlink /etc/systemd/system/multi-user.target.wants/openvpn-server@server.service → /usr/lib/systemd/system/openvpn-server@.service.

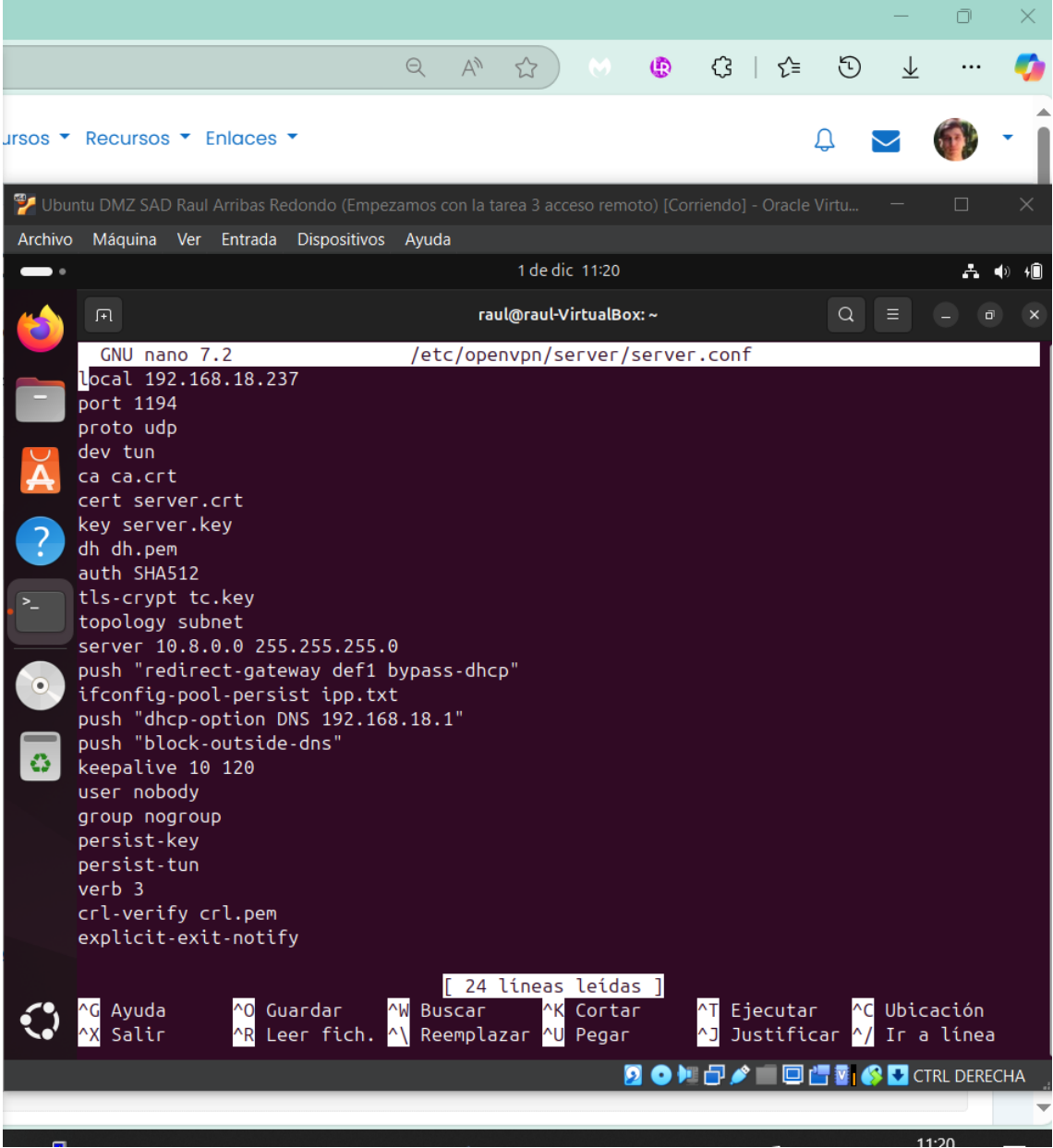
Finished!

The client configuration is available in: /root/raularribascliente.ovpn
New clients can be added by running this script again.
raul@raul-VirtualBox:~/openvpn-install$
```

Una vez hemos creado el usuario cliente, el archivo .ovpn se nos ha guardado en /root/raularribascliente.ovpn

Verificamos que estamos utilizando TLS:

Editamos el archivo **/etc/openvpn/server.conf** y tienen que tener estas líneas:



The screenshot shows a terminal window titled 'raul@raul-VirtualBox: ~' with the file `/etc/openvpn/server/server.conf` open in the nano editor. The configuration file contains the following lines:

```
local 192.168.18.237
port 1194
proto udp
dev tun
ca ca.crt
cert server.crt
key server.key
dh dh.pem
auth SHA512
tls-crypt tc.key
topology subnet
server 10.8.0.0 255.255.255.0
push "redirect-gateway def1 bypass-dhcp"
ifconfig-pool-persist ipp.txt
push "dhcp-option DNS 192.168.18.1"
push "block-outside-dns"
keepalive 10 120
user nobody
group nogroup
persist-key
persist-tun
verb 3
crl-verify crl.pem
explicit-exit-notify
```

At the bottom of the terminal, there is a status bar with keyboard shortcuts: `^G Ayuda`, `^X Salir`, `^O Guardar`, `^R Leer fich.`, `^W Buscar`, `^_ Reemplazar`, `^K Cortar`, `^U Pegar`, `^T Ejecutar`, `^C Ubicación`, `^J Justificar`, `^_ Ir a línea`. The text `[ 24 líneas leídas ]` is also visible.

La línea **tls-crypt tc.key** indica que estás usando un canal TLS para cifrar y autenticar los paquetes de control.

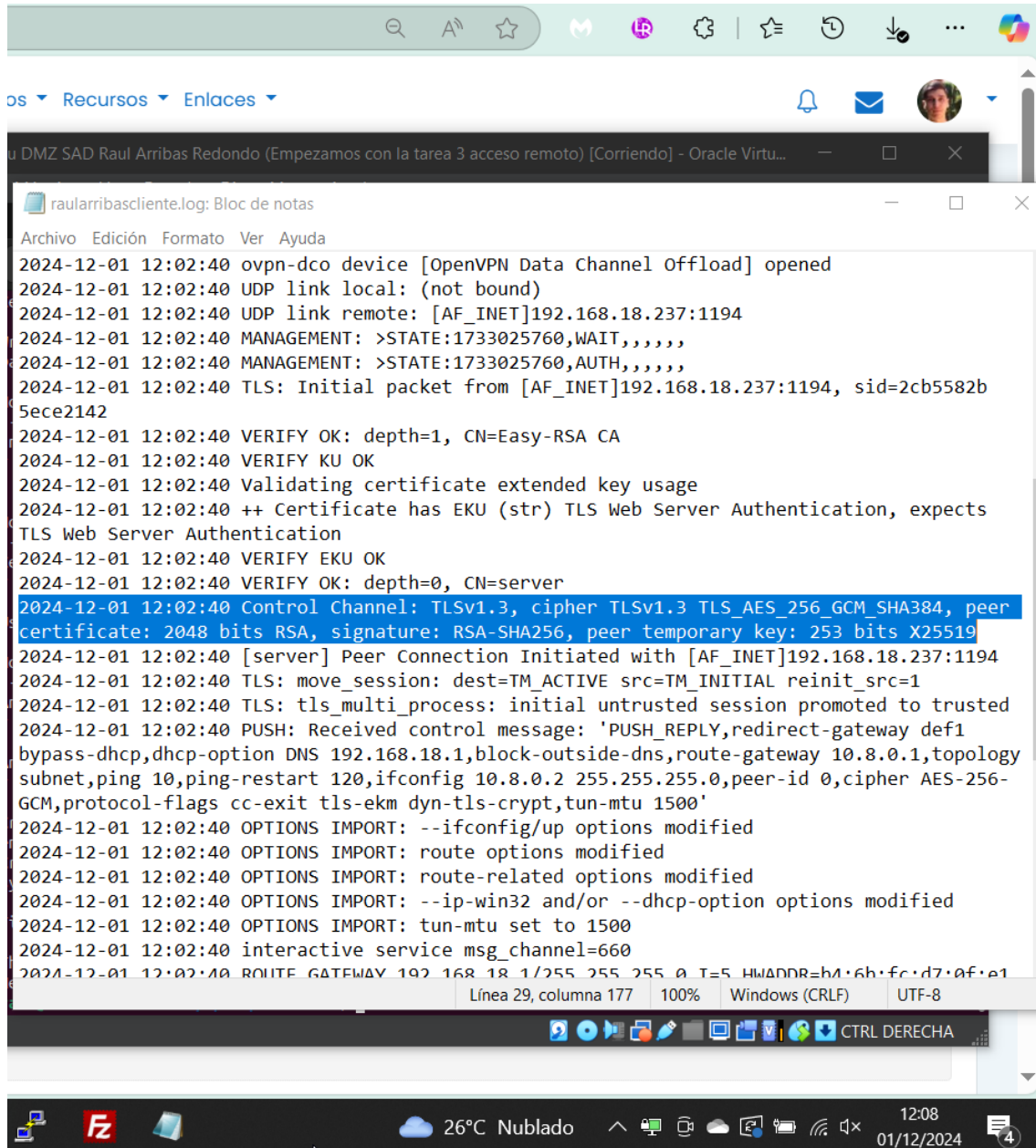
Y por ultimo al archivo de configuración .ovpn le agregamos esta línea:

tls-crypt tc.key

Comprobación de la conexión VPN con TLS

Para conectarnos tenemos que copiar el archivo ovpn a nuestro equipo Windows. Después de copiarlo lo importamos a OpenVPN (tenemos que tener la aplicación previamente instalada de OpenVPN GUI).

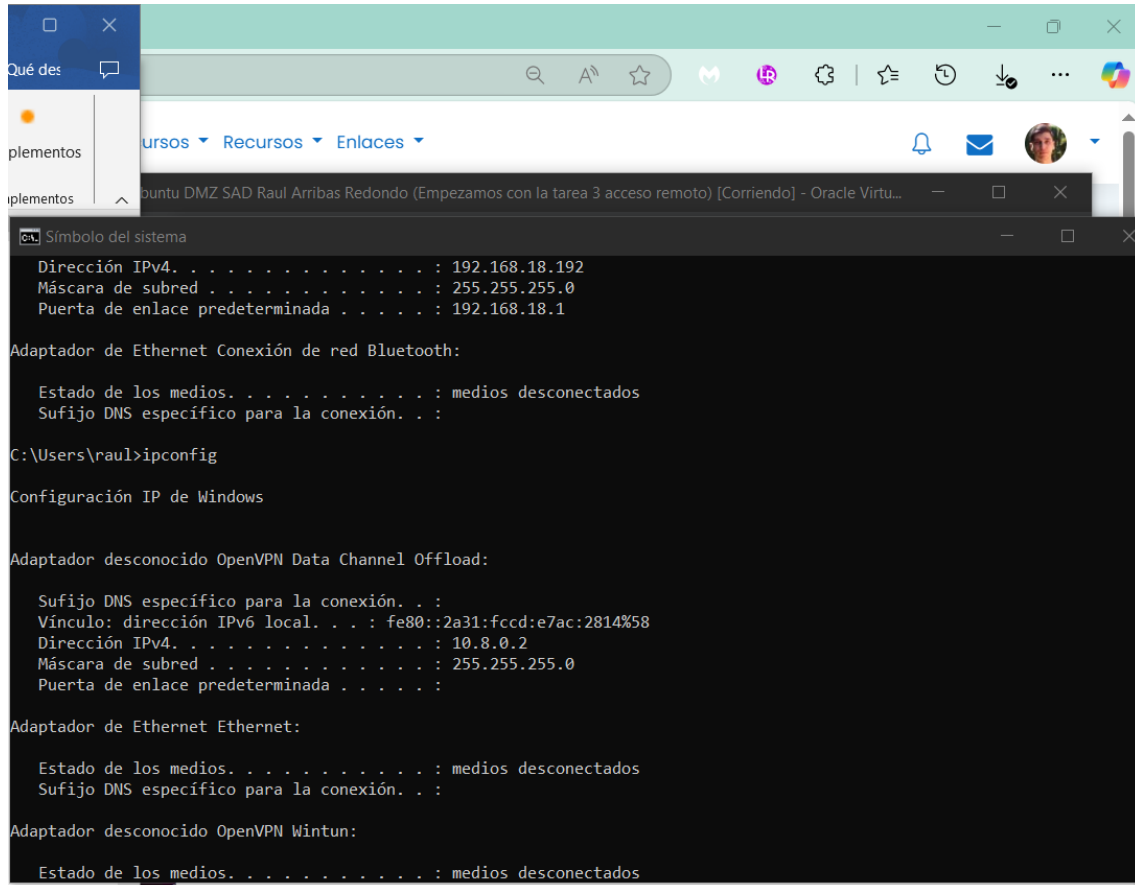
Al conectarnos podemos ver los logs y verificamos que se ha conectado correctamente:



```
raularribascliente.log: Bloc de notas
Archivo Edición Formato Ver Ayuda
2024-12-01 12:02:40 ovpn-dco device [OpenVPN Data Channel Offload] opened
2024-12-01 12:02:40 UDP link local: (not bound)
2024-12-01 12:02:40 UDP link remote: [AF_INET]192.168.18.237:1194
2024-12-01 12:02:40 MANAGEMENT: >STATE:1733025760, WAIT,,,,,
2024-12-01 12:02:40 MANAGEMENT: >STATE:1733025760, AUTH,,,,,
2024-12-01 12:02:40 TLS: Initial packet from [AF_INET]192.168.18.237:1194, sid=2cb5582b5e2142
2024-12-01 12:02:40 VERIFY OK: depth=1, CN=Easy-RSA CA
2024-12-01 12:02:40 VERIFY KU OK
2024-12-01 12:02:40 Validating certificate extended key usage
2024-12-01 12:02:40 ++ Certificate has EKU (str) TLS Web Server Authentication, expects TLS Web Server Authentication
2024-12-01 12:02:40 VERIFY EKU OK
2024-12-01 12:02:40 VERIFY OK: depth=0, CN=server
2024-12-01 12:02:40 Control Channel: TLSv1.3, cipher TLSv1.3 TLS_AES_256_GCM_SHA384, peer certificate: 2048 bits RSA, signature: RSA-SHA256, peer temporary key: 253 bits X25519
2024-12-01 12:02:40 [server] Peer Connection Initiated with [AF_INET]192.168.18.237:1194
2024-12-01 12:02:40 TLS: move_session: dest=TM_ACTIVE src=TM_INITIAL reinit_src=1
2024-12-01 12:02:40 TLS: tls_multi_process: initial untrusted session promoted to trusted
2024-12-01 12:02:40 PUSH: Received control message: 'PUSH_REPLY,redirect-gateway def1 bypass-dhcp,dhcp-option DNS 192.168.18.1,block-outside-dns,route-gateway 10.8.0.1,topology subnet,ping 10,ping-restart 120,ifconfig 10.8.0.2 255.255.255.0,peer-id 0,cipher AES-256-GCM,protocol-flags cc-exit tls-ekm dyn-tls-crypt,tun-mtu 1500'
2024-12-01 12:02:40 OPTIONS IMPORT: --ifconfig/up options modified
2024-12-01 12:02:40 OPTIONS IMPORT: route options modified
2024-12-01 12:02:40 OPTIONS IMPORT: route-related options modified
2024-12-01 12:02:40 OPTIONS IMPORT: --ip-win32 and/or --dhcp-option options modified
2024-12-01 12:02:40 OPTIONS IMPORT: tun-mtu set to 1500
2024-12-01 12:02:40 interactive service msg_channel=660
2024-12-01 12:02:40 ROUTE GATEWAY 192.168.18.1/255.255.255.0 T=5 HWADDR=b4:6b:fc:d7:0f:e1
```

2.2 Verifica la correcta conexión a través del túnel mediante un *ping* desde el *cliente VPN* al *servidor VPN*.

Y también vemos que nos ha dado un dirección Ip del rango de nuestra configuración del servidor VPN 10.8.0.2:



```
Símbolo del sistema
Dirección IPv4. . . . . : 192.168.18.192
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . : 192.168.18.1

Adaptador de Ethernet Conexión de red Bluetooth:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . . :

C:\Users\raul>ipconfig

Configuración IP de Windows

Adaptador desconocido OpenVPN Data Channel Offload:

    Sufijo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . : fe80::2a31:fccd:e7ac:2814%58
    Dirección IPv4. . . . . : 10.8.0.2
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . :

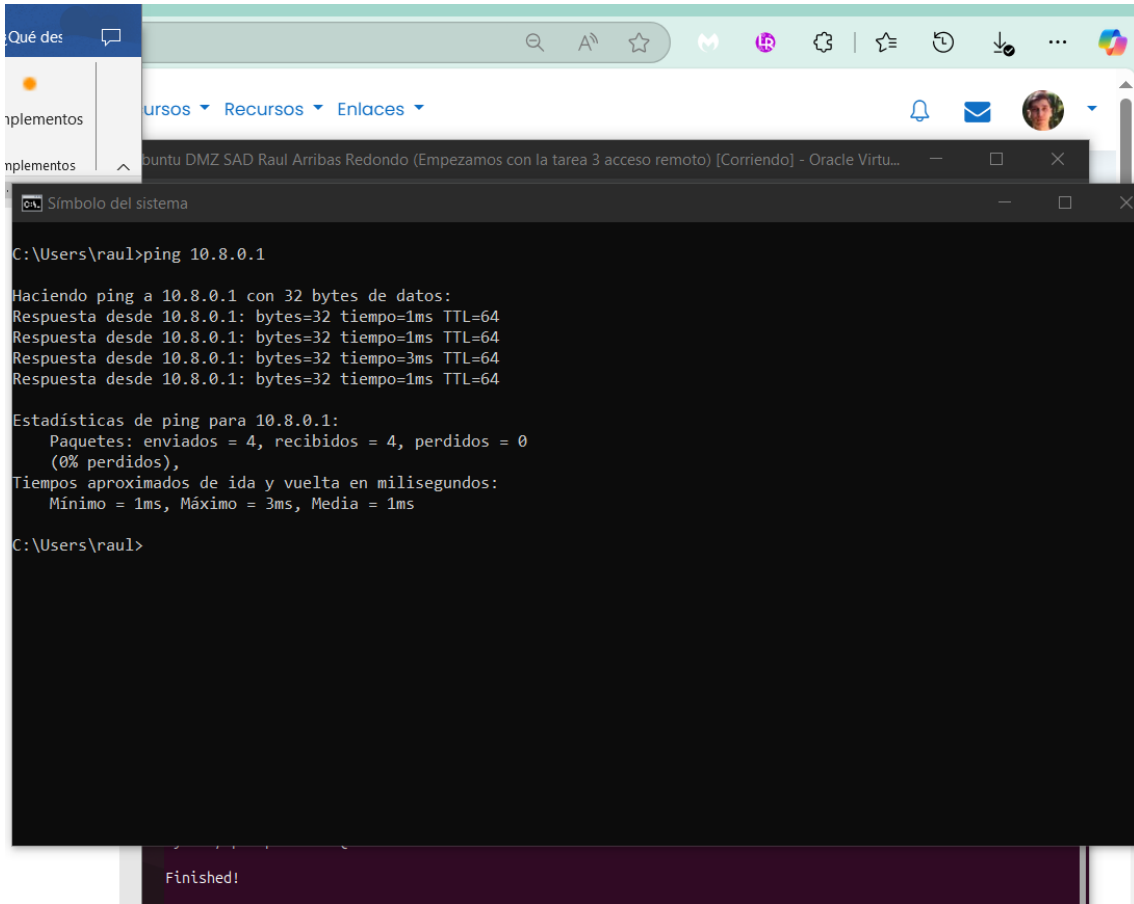
Adaptador de Ethernet Ethernet:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . . :

Adaptador desconocido OpenVPN Wintun:

    Estado de los medios. . . . . : medios desconectados
```

Hacemos PING al servidor VPN, que tiene la IP: 10.8.0.1:



The screenshot shows a Windows desktop environment. In the background, a web browser is open with a page titled 'puntu DMZ SAD Raul Arribas Redondo (Empezamos con la tarea 3 acceso remoto) [Corriendo] - Oracle Virtu...'. In the foreground, a 'Símbolo del sistema' (Command Prompt) window is open. The user has entered the command 'ping 10.8.0.1'. The output shows four successful replies from 10.8.0.1 with varying response times (1ms, 1ms, 3ms, 1ms) and a TTL of 64. The statistics indicate 4 packets sent, 4 received, and 0 lost (0% loss). The round-trip times are approximately 1ms to 3ms, with a minimum of 1ms, maximum of 3ms, and a median of 1ms. The prompt returns to 'C:\Users\raul>'. Below the terminal window, a dark purple bar contains the text 'Finished!'.

```
C:\Users\raul>ping 10.8.0.1

Haciendo ping a 10.8.0.1 con 32 bytes de datos:
Respuesta desde 10.8.0.1: bytes=32 tiempo=1ms TTL=64
Respuesta desde 10.8.0.1: bytes=32 tiempo=1ms TTL=64
Respuesta desde 10.8.0.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 10.8.0.1: bytes=32 tiempo=1ms TTL=64

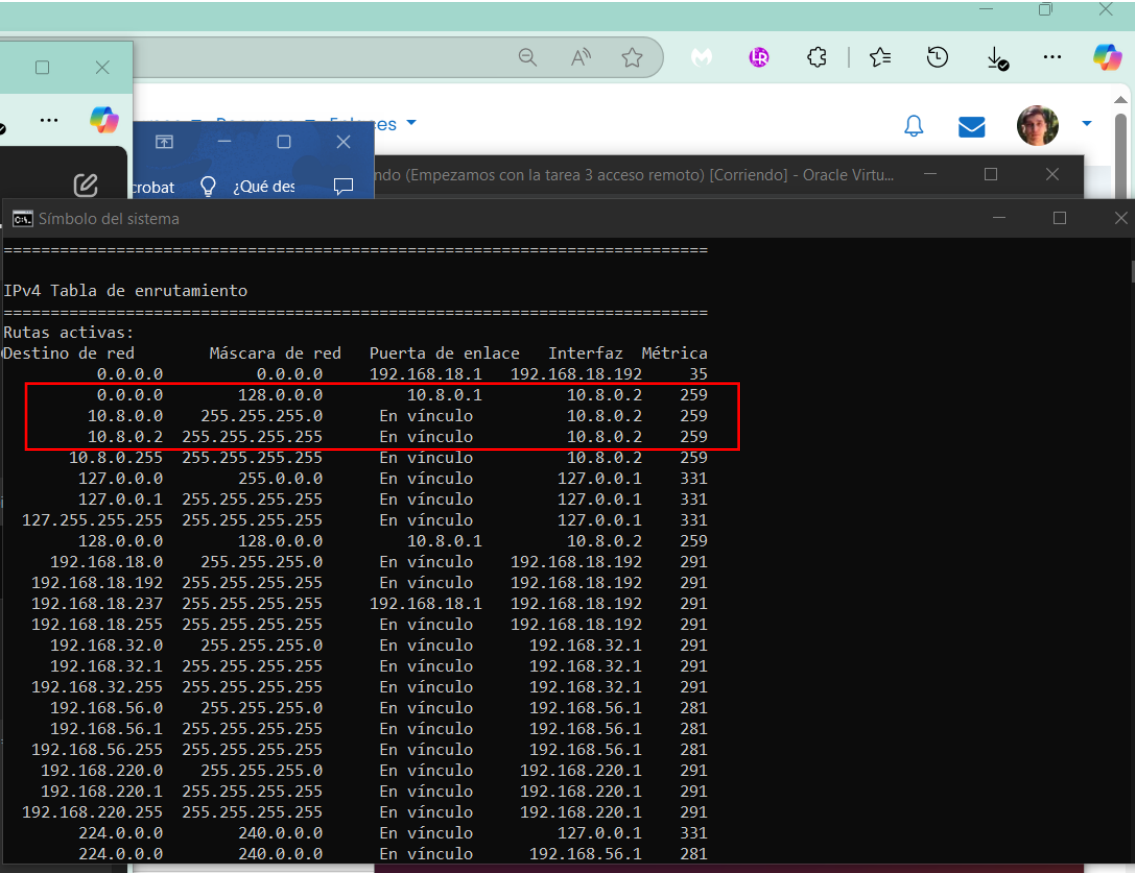
Estadísticas de ping para 10.8.0.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
            (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 1ms, Máximo = 3ms, Media = 1ms

C:\Users\raul>
```

Finished!

2.3 Muestra la tablas de enrutamiento y las interfaces de red, tanto del cliente VPN como del servidor VPN.

Mostrar la tabla de enrutamiento del cliente:

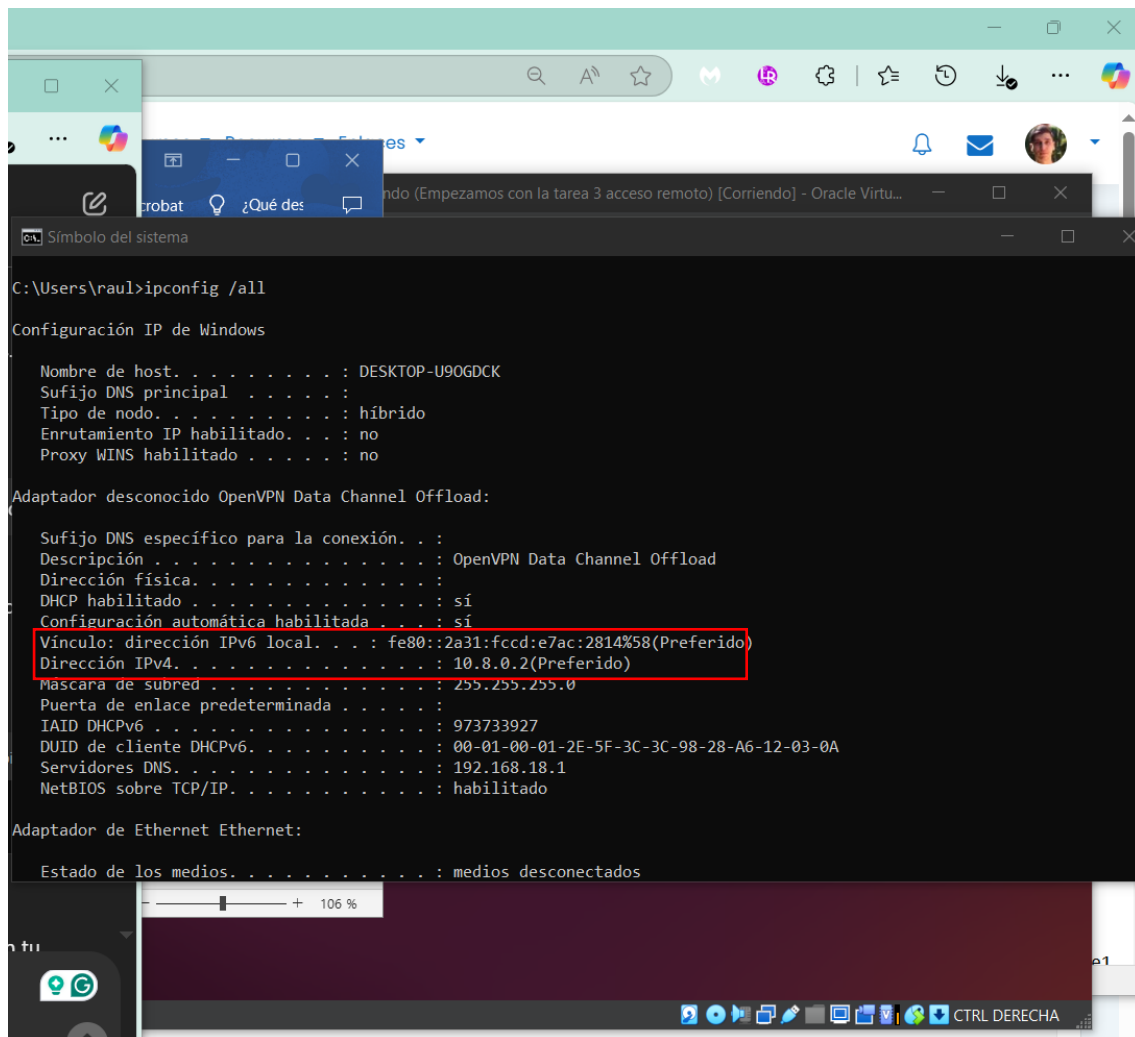


IPv4 Tabla de enrutamiento

Rutas activas:

Destino de red	Máscara de red	Puerta de enlace	Interfaz	Métrica
0.0.0.0	0.0.0.0	192.168.18.1	192.168.18.192	35
0.0.0.0	128.0.0.0	10.8.0.1	10.8.0.2	259
10.8.0.0	255.255.255.0	En vínculo	10.8.0.2	259
10.8.0.2	255.255.255.255	En vínculo	10.8.0.2	259
10.8.0.255	255.255.255.255	En vínculo	10.8.0.2	259
127.0.0.0	255.0.0.0	En vínculo	127.0.0.1	331
127.0.0.1	255.255.255.255	En vínculo	127.0.0.1	331
127.255.255.255	255.255.255.255	En vínculo	127.0.0.1	331
128.0.0.0	128.0.0.0	10.8.0.1	10.8.0.2	259
192.168.18.0	255.255.255.0	En vínculo	192.168.18.192	291
192.168.18.192	255.255.255.255	En vínculo	192.168.18.192	291
192.168.18.237	255.255.255.255	192.168.18.1	192.168.18.192	291
192.168.18.255	255.255.255.255	En vínculo	192.168.18.192	291
192.168.32.0	255.255.255.0	En vínculo	192.168.32.1	291
192.168.32.1	255.255.255.255	En vínculo	192.168.32.1	291
192.168.32.255	255.255.255.255	En vínculo	192.168.32.1	291
192.168.56.0	255.255.255.0	En vínculo	192.168.56.1	281
192.168.56.1	255.255.255.255	En vínculo	192.168.56.1	281
192.168.56.255	255.255.255.255	En vínculo	192.168.56.1	281
192.168.220.0	255.255.255.0	En vínculo	192.168.220.1	291
192.168.220.1	255.255.255.255	En vínculo	192.168.220.1	291
192.168.220.255	255.255.255.255	En vínculo	192.168.220.1	291
224.0.0.0	240.0.0.0	En vínculo	127.0.0.1	331
224.0.0.0	240.0.0.0	En vínculo	192.168.56.1	281

Mostrar interfaz de red del cliente:



```
C:\Users\raul>ipconfig /all

Configuración IP de Windows

Nombre de host. . . . . : DESKTOP-U90GDCK
Sufijo DNS principal . . . . :
Tipo de nodo. . . . . : híbrido
Enrutamiento IP habilitado. . . : no
Proxy WINS habilitado . . . . : no

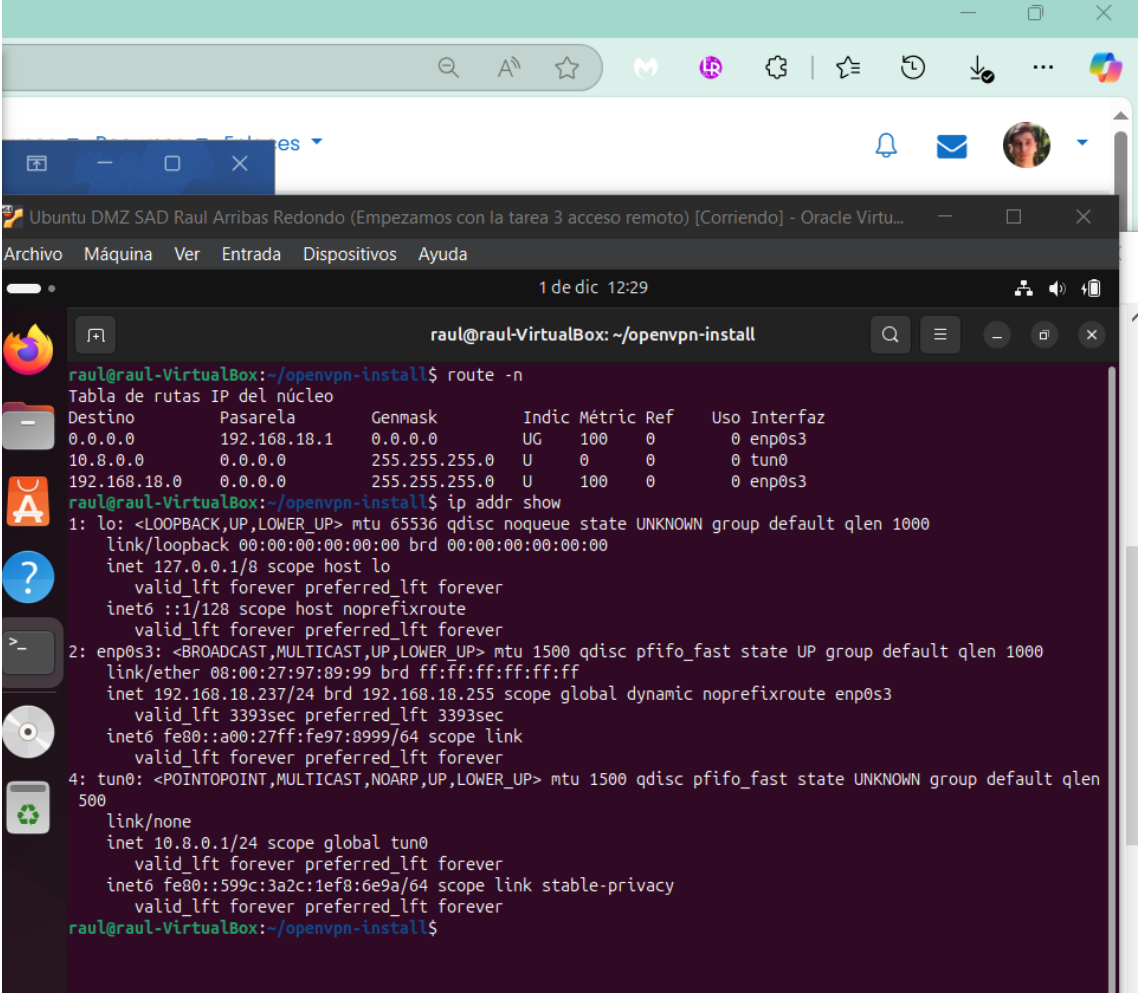
Adaptador desconocido OpenVPN Data Channel Offload:

Sufijo DNS específico para la conexión. . :
Descripción . . . . . : OpenVPN Data Channel Offload
Dirección física. . . . . :
DHCP habilitado . . . . . : sí
Configuración automática habilitada . . . : sí
Vínculo: dirección IPv6 local. . . : fe80::2a31:fccd:e7ac:2814%58(Preferido)
Dirección IPv4. . . . . : 10.8.0.2(Preferido)
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . :
IAID DHCPv6 . . . . . : 973733927
DUID de cliente DHCPv6. . . . : 00-01-00-01-2E-5F-3C-3C-98-28-A6-12-03-0A
Servidores DNS. . . . . : 192.168.18.1
NetBIOS sobre TCP/IP. . . . . : habilitado

Adaptador de Ethernet Ethernet:

Estado de los medios. . . . . : medios desconectados
```

Mostrar la tabla de enrutamiento y la interfaz de red del servidor:



```

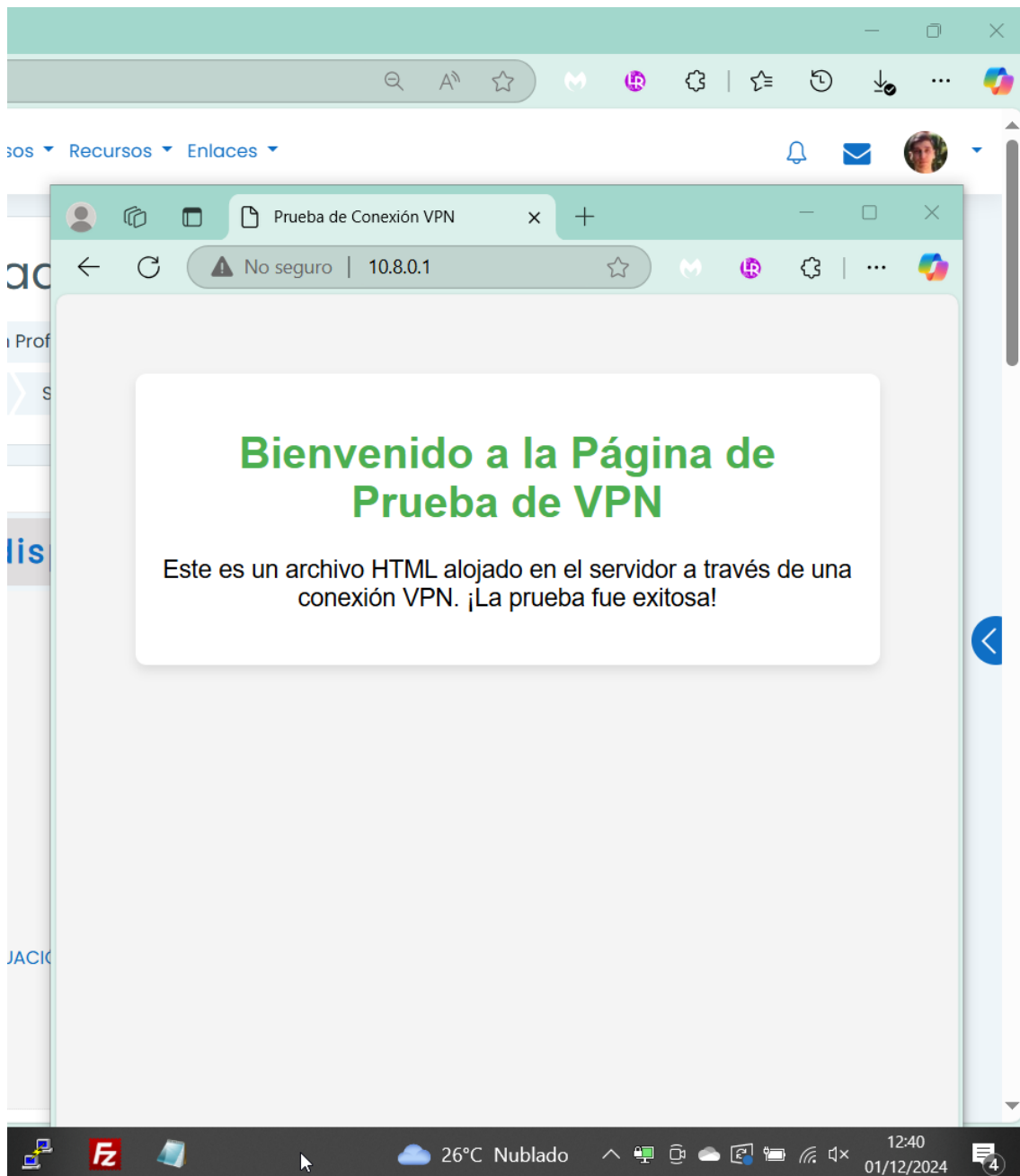
raul@raul-VirtualBox: ~/openvpn-install
raul@raul-VirtualBox:~/openvpn-install$ route -n
Tabla de rutas IP del núcleo
Destino      Pasarela      Genmask      Indic Métric Ref       Uso Interfaz
0.0.0.0      192.168.18.1  0.0.0.0      UG     100    0        0 enp0s3
10.8.0.0     0.0.0.0      255.255.255.0 U       0      0        0 tun0
192.168.18.0 0.0.0.0      255.255.255.0 U     100    0        0 enp0s3
raul@raul-VirtualBox:~/openvpn-install$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
   inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
   link/ether 08:00:27:97:89:99 brd ff:ff:ff:ff:ff:ff
   inet 192.168.18.237/24 brd 192.168.18.255 scope global dynamic noprefixroute enp0s3
       valid_lft 3393sec preferred_lft 3393sec
   inet6 fe80::a00:27ff:fe97:8999/64 scope link
       valid_lft forever preferred_lft forever
4: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
   link/none
   inet 10.8.0.1/24 scope global tun0
       valid_lft forever preferred_lft forever
   inet6 fe80::599c:3a2c:1ef8:6e9a/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
raul@raul-VirtualBox:~/openvpn-install$

```

2.4 Ubica la página [PruebaConexionVPN.html](#) en el *servidor VPN*. Usa el túnel VPN previamente establecido para visualizar la página desde el navegador web del *cliente VPN*.

Hemos instalado el cliente de apache y hemos creado una pagina web que hemos subido a la carpeta `/var/www/html`.

Probamos que la podemos ver desde el cliente con la VPN conectada:



2.5 Instala y configura *un cliente VPN* en la máquina virtual *Windows 10* del entorno.

Ya lo hemos realizado en los pasos anteriores.

2.6 Muestra las tablas de enrutamiento y las interfaces de red, tanto del cliente VPN Windows 10 como del servidor VPN.

Ya lo hemos realizado anteriormente.

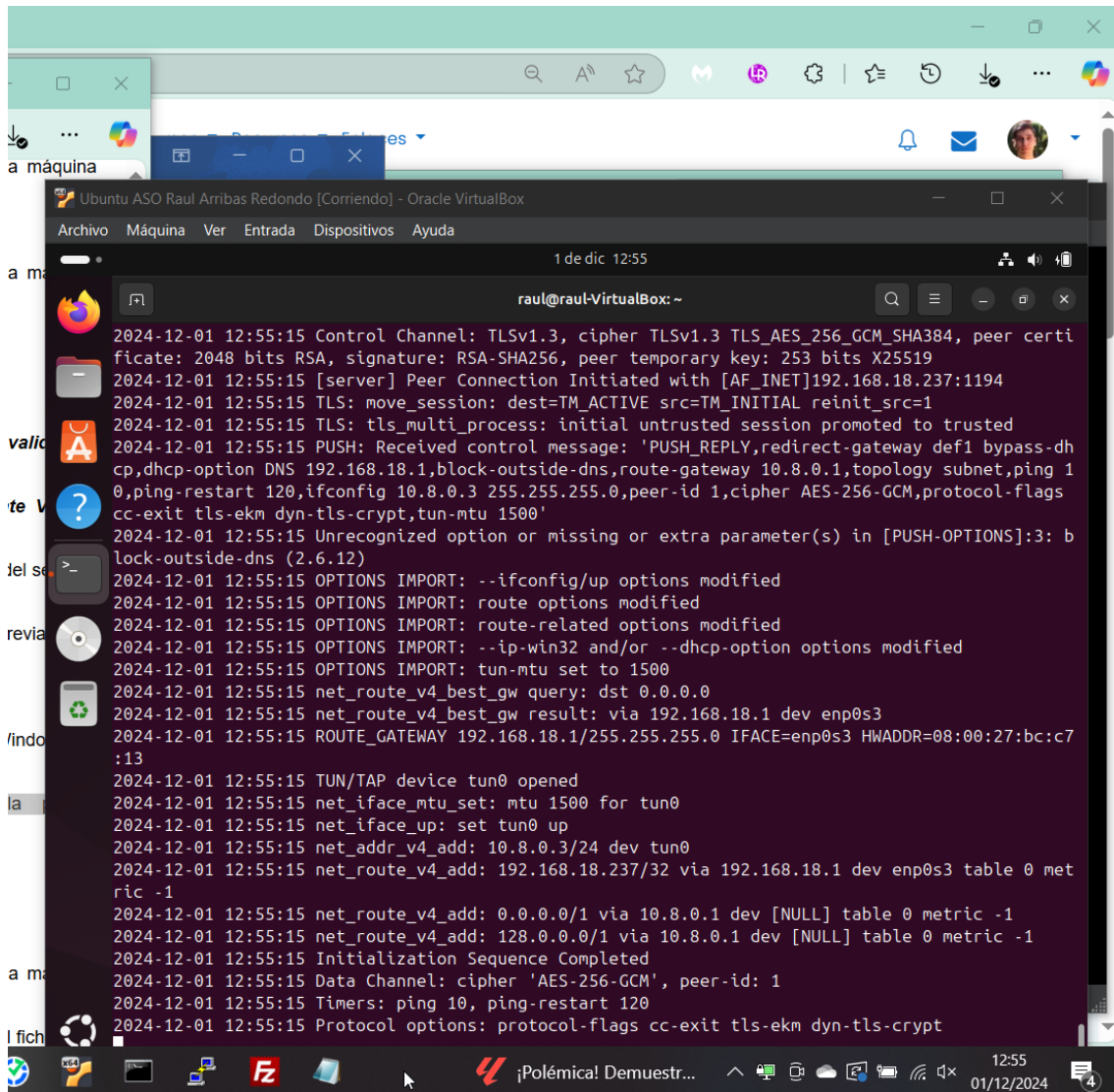
2.7 Realiza una conexión al servidor usando el *túnel VPN* y visualiza la página *PruebaConexionVPN.html* desde un navegador web de la máquina *Windows 10*.

Ya lo hemos realizado anteriormente

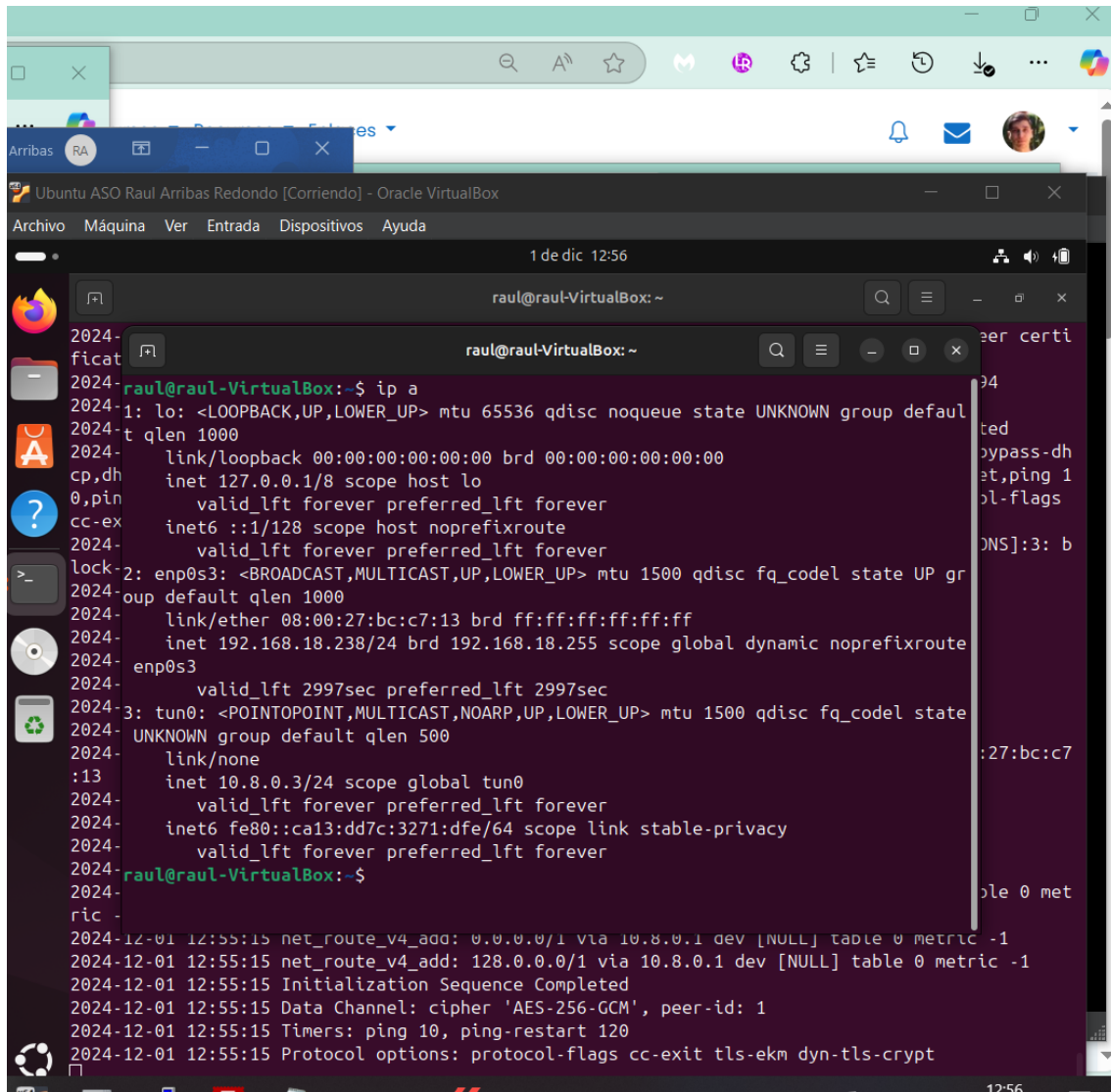
Manual de instalación

He visto que seguramente tenía que conectarme también al servidor openvpn desde un cliente de Ubuntu, así que aquí hago un par de capturas de pantalla para que se vea que asigna una dirección IP del rango de la VPN. Hemos creado un nuevo cliente al que le hemos llamado, `raulubuntucliente.ovpn`.

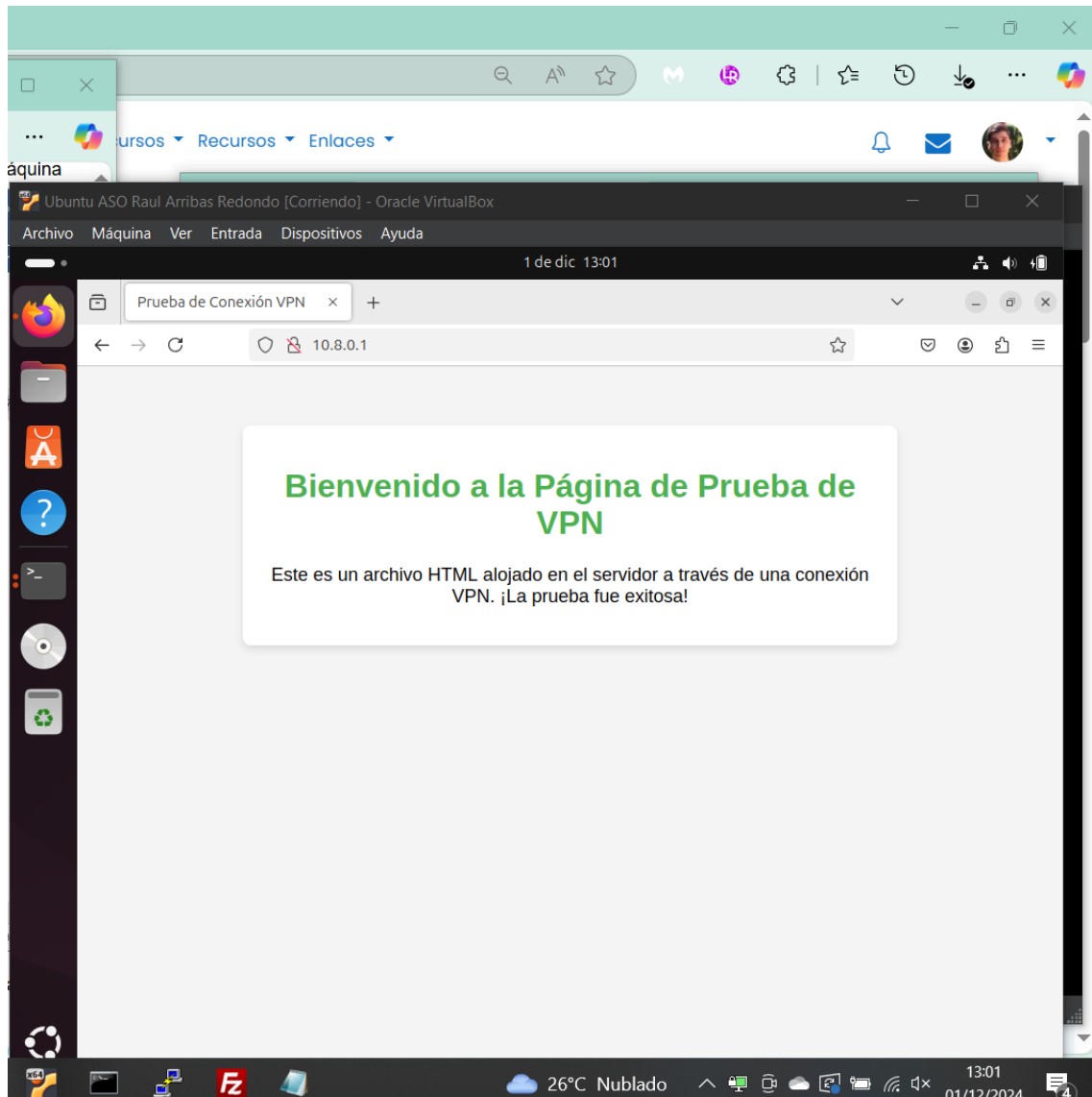
En la siguiente imagen vemos que nos hemos conectado correctamente:



En la siguiente imagen hacemos un `ip a`, para mostrar que estamos conectados a la VPN correctamente, nos ha dado la `ip 10.8.0.3`



Tambien mostramos que podemos acceder a la pagina WEB desde el cliente de Ubuntu:

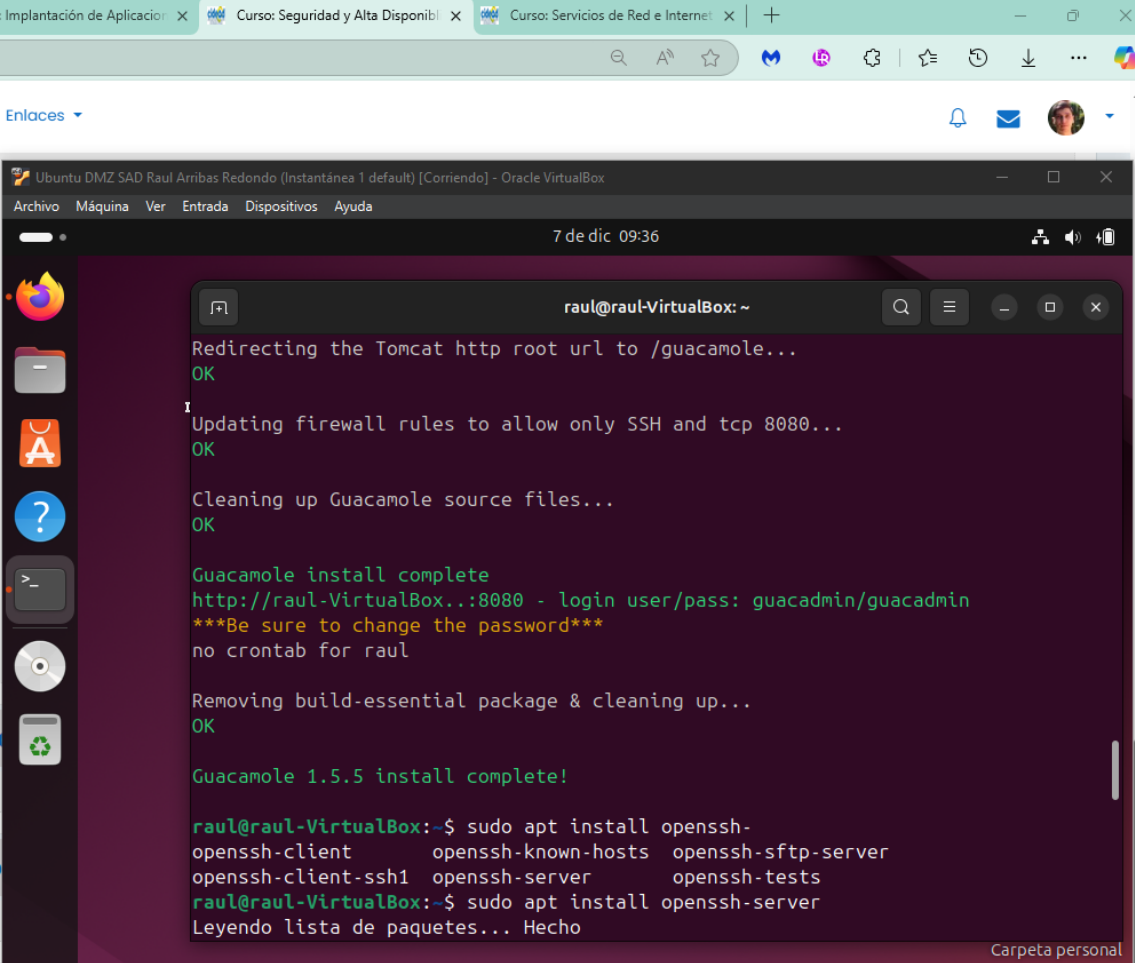


3 Guacamole

3.1 Instala en el servidor *VPN* un servicio *Guacamole*.

Para instalar el servidor Guacamole hemos utilizado este manual de Git-hub [GitHub - itilgent/Easy-Guacamole-Installer: Automated install of Guacamole with options for HTTPS reverse proxy, Active Directory integration, MFA, LetsEncrypt, dark theme, MySQL backup, email alerts & more.](#)

Ejecutamos el Script y tras terminar nos saldrá un pantalla que nos indique que se ha instalado correctamente:



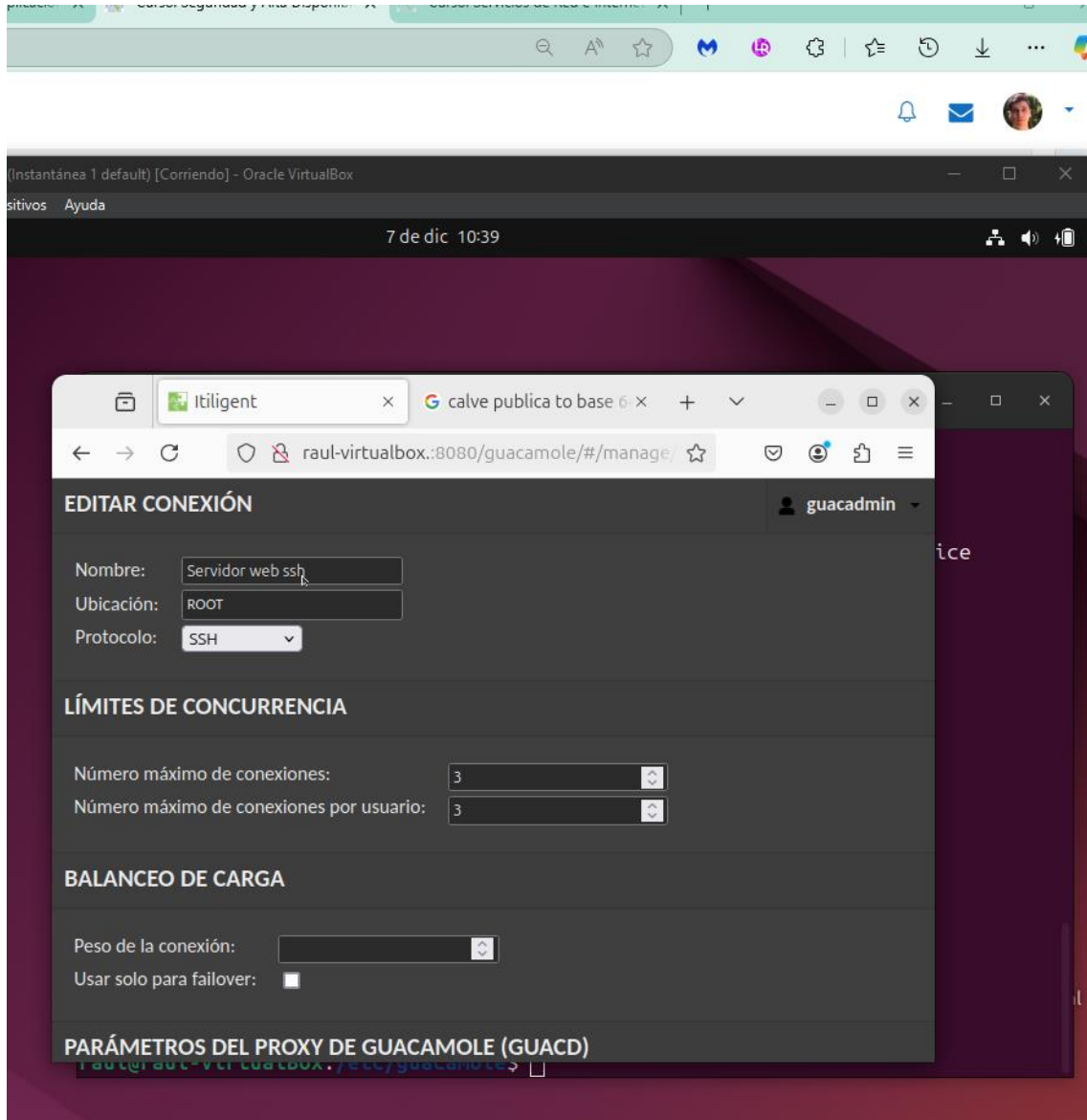
```
raul@raul-VirtualBox: ~  
Redirecting the Tomcat http root url to /guacamole...  
OK  
Updating firewall rules to allow only SSH and tcp 8080...  
OK  
Cleaning up Guacamole source files...  
OK  
Guacamole install complete  
http://raul-VirtualBox.:8080 - login user/pass: guacadmin/guacadmin  
***Be sure to change the password***  
no crontab for raul  
Removing build-essential package & cleaning up...  
OK  
Guacamole 1.5.5 install complete!  
raul@raul-VirtualBox:~$ sudo apt install openssh-  
openssh-client      openssh-known-hosts  openssh-sftp-server  
openssh-client-ssh1  openssh-server        openssh-tests  
raul@raul-VirtualBox:~$ sudo apt install openssh-server  
Leyendo lista de paquetes... Hecho
```

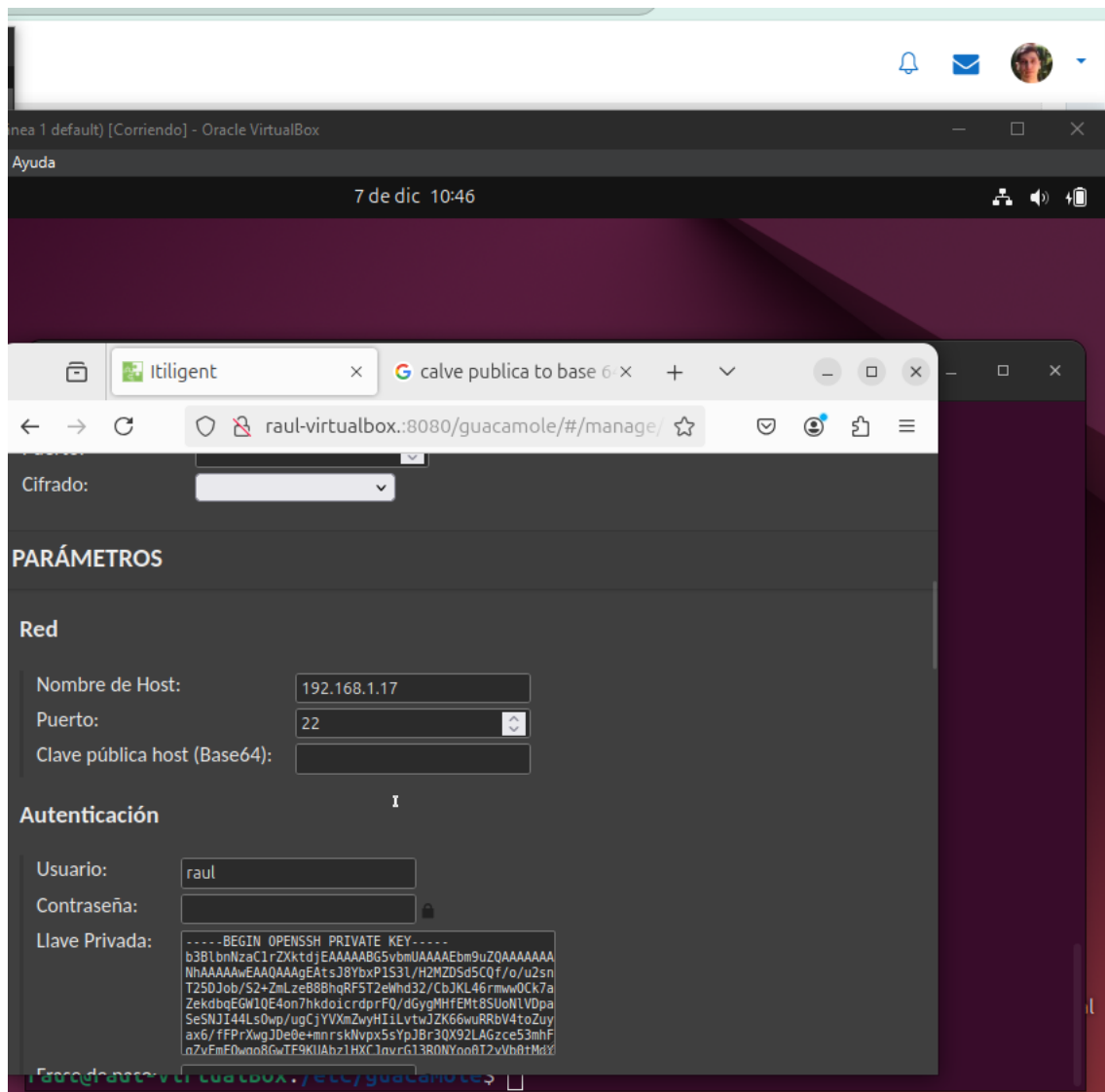
3.2 Configura dicho servicio para que se pueda acceder por SSH al servidor web de la máquina **dmza**.

Vamos a entrar en la URL: <http://raul-virtualbox.:8080/guacamole/#/>

Vamos a crear una nueva conexión SSH

Configuración → Conexiones → Nueva Conexión → SSH





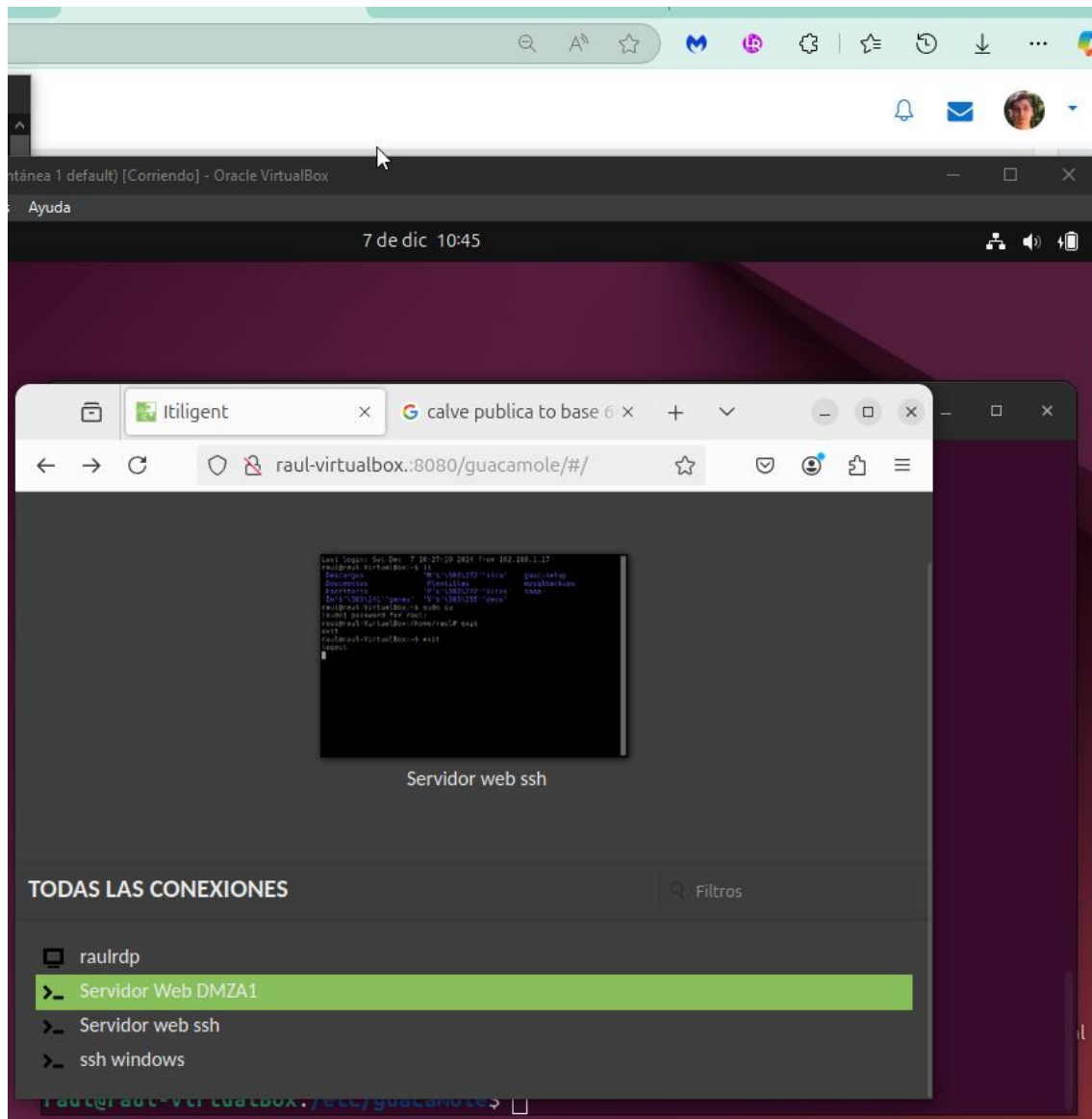
Ponemos la clave privada de nuestro servidor, y Guacamole la guardará en la base de datos de forma segura.

3.3 Utiliza algún mecanismo de cifrado para que la clave de acceso no se pueda leer en el fichero de configuración de Guacamole.

Hecho en el paso anterior.

3.4 Verifica que la conexión funciona correctamente tras dicho cambio.

Entramos a la pagina web: <http://raul-virtualbox.:8080/guacamole> e iniciamos sesión



Pinchamos en el servidor web ssh y comprobamos que accedemos correctamente con nuestra clave privada.

