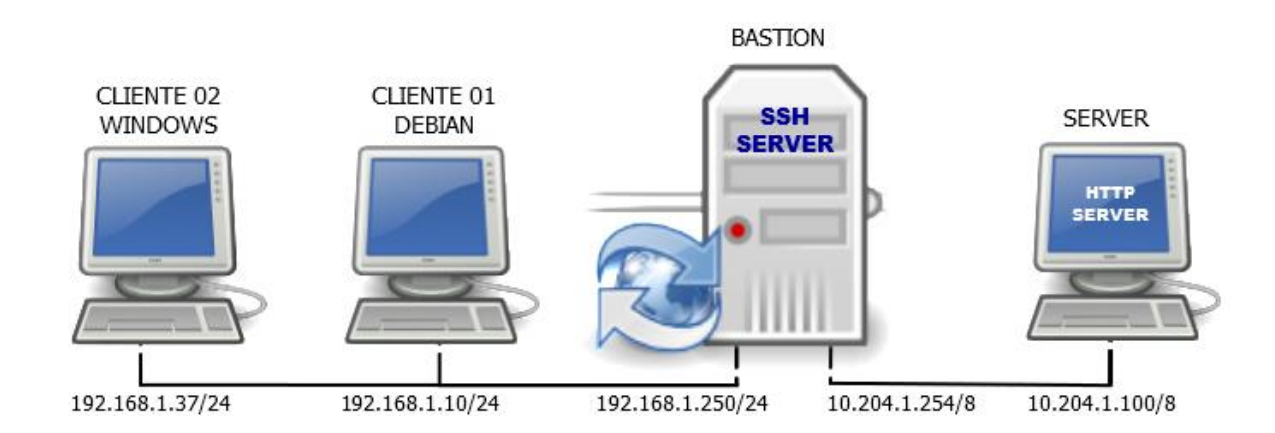
Practica 3 Túneles SSH + anexo



Carlos González Martín

Servicios en red

2º Grado medio de sistemas microinformáticos y redes

Curso 2022/2023

Contenido

[1. Primeros pasos 3](#_Toc121064423)

[2. Configuramos las maquinas 3](#_Toc121064424)

[3. Instalamos paquetes 4](#_Toc121064425)

[4. Configuramos interfaces 5](#_Toc121064426)

[5. Configuramos SSH 5](#_Toc121064427)

[6. Comprobaciones iniciales 6](#_Toc121064428)

[7. Comprobamos túneles 6](#_Toc121064429)

[8. Comprobaciones finales 7](#_Toc121064430)

[9. Socks 8](#_Toc121064431)

[10. Webgrafía 15](#_Toc121064432)

# Ilustraciones

[Ilustración 1: Cambiamos nombre a la maquina 3](#_Toc121064440)

[Ilustración 2: Creamos usuario Batman 4](#_Toc121064441)

[Ilustración 3: instalamos apache 4](#_Toc121064442)

[Ilustración 4: instalamos SSH 5](#_Toc121064443)

[Ilustración 5: configuramos interfaces 5](#_Toc121064444)

[Ilustración 6: configuramos ssh 6](#_Toc121064445)

[Ilustración 7: Comprobamos ssh 6](#_Toc121064446)

[Ilustración 8: abrimos túnel 7](#_Toc121064447)

[Ilustración 9: Conexión mediante lynx 7](#_Toc121064448)

[Ilustración 10: cliente como puerta de enlace 8](#_Toc121064449)

[Ilustración 11: Windows visualizando la página web de server 8](#_Toc121064450)

[Ilustración 12: instalamos vsftpd 9](#_Toc121064451)

[Ilustración 13: Abrimos túnel dinámico 10](#_Toc121064452)

[Ilustración 14: Conectamos por putty 11](#_Toc121064453)

[Ilustración 15: Abrimos nueva sesión 11](#_Toc121064454)

[Ilustración 16: sesión abierta 12](#_Toc121064455)

[Ilustración 17: Configuramos FileZilla 12](#_Toc121064456)

[Ilustración 18: SOCKS 13](#_Toc121064457)

[Ilustración 19: Advertencia de seguridad 13](#_Toc121064458)

[Ilustración 20: Conexión exitosa 14](#_Toc121064459)

[Ilustración 21: Cambiamos directorio conexión 14](#_Toc121064460)

[Ilustración 22: Comprobamos los cambios 14](#_Toc121064461)

[Ilustración 23: Conexión exitosa al nuevo directorio 15](#_Toc121064462)

[Ilustración 24: transferencia exitosa 16](#_Toc121064463)

# Primeros pasos

Para comentar lo que haremos será clonar 3 máquinas virtuales debian11 sin entorno gráfico y 1 maquina Windows 7 que usaremos de cliente.

# Configuramos las maquinas

Para este paso lo que haremos será cambiar los nombres de las máquinas y crearemos los usuarios pertinentes que en este caso es Batman en la maquina bastión.

~# hostnamectl set-hostname bastión



Ilustración : Cambiamos nombre a la maquina

Ahora lo que haremos será hacer un Exit o un reboot.

Esto lo haremos tanto la maquina cliente como en la maquina server.

Luego también creamos el usuario Batman.

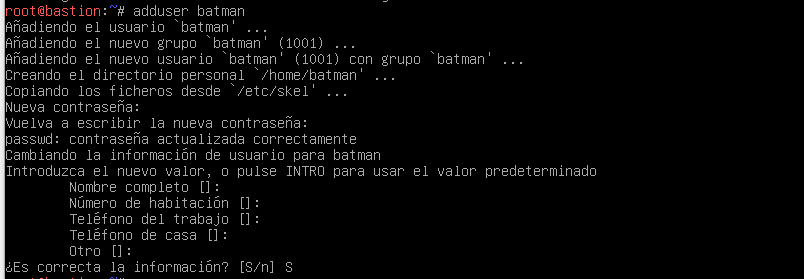


Ilustración : Creamos usuario Batman

# Instalamos paquetes

Para comenzar lo que haremos será instalar en la máquina que hará de servidor WEB apache y lo haremos con el siguiente comando.

~# apt update ; apt install apache2 –y

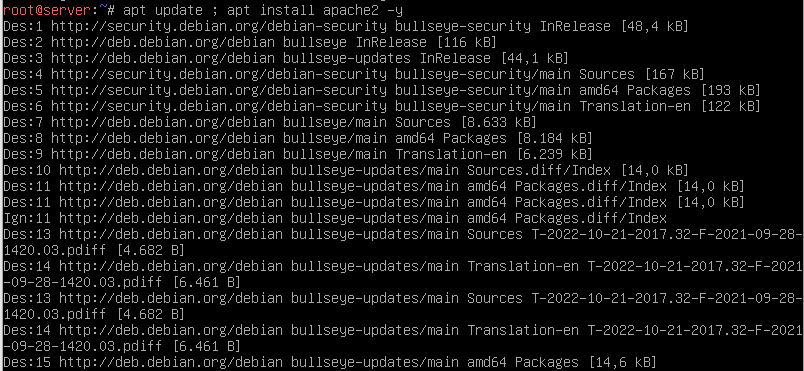


Ilustración : instalamos apache

Si queremos cambiar la página web que viene por defecto y descargarnos por ejemplo una página de internet tendremos que instalar SSH para poder copiar los archivos de la maquina real a la maquina debían.

Luego también tenemos que instalar solo SSH en la maquina bastión.

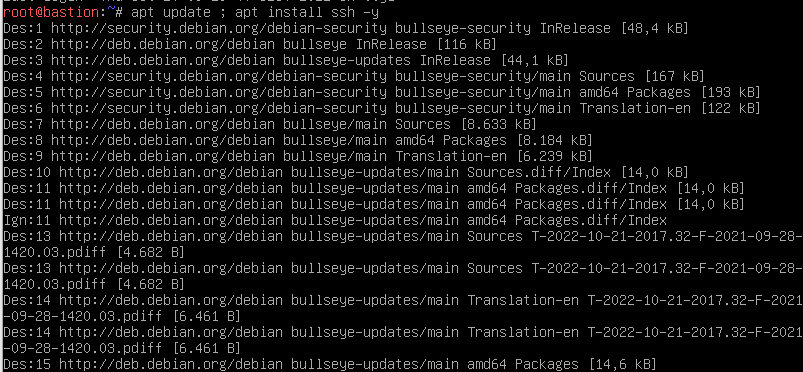


Ilustración : instalamos SSH

En la maquina cliente lo que debemos de instalar es Lynx que es un cliente web.

# Configuramos interfaces

Configuraremos las interfaces de todas las maquinas como en el ejercicio propuesto.

~# nano /etc/network/interfaces

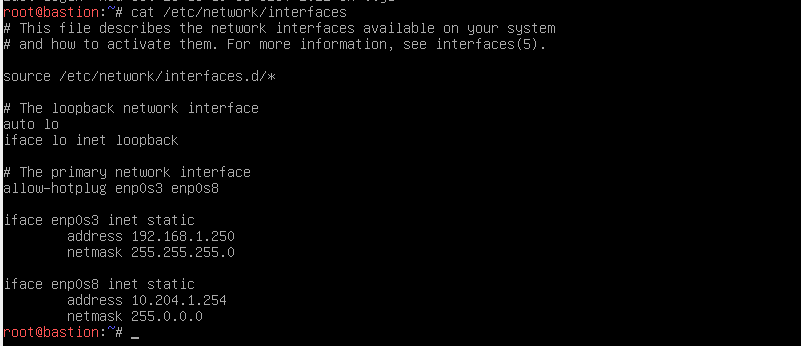


Ilustración 5: configuramos interfaces

En este caso es la maquina bastión como pone en el nombre de la maquina ya que es la máquina que tiene 2 interfaces.

Y lo haremos con el resto de máquinas.

# 5. Configuramos SSH

En este caso lo que haremos será configurar ssh en su respectivo archivo de configuración

~# nano /etc/ssh/sshd\_config



Ilustración 6: configuramos ssh

En este caso es solo permitir el acceso a Batman.

# 6. Comprobaciones iniciales

Lo que haremos será conectarnos directamente a bastión con el usuario Batman y también probaremos si con usuario nos conecta ya que es otro usuario del sistema.

~# ssh batman@192.168.1.250

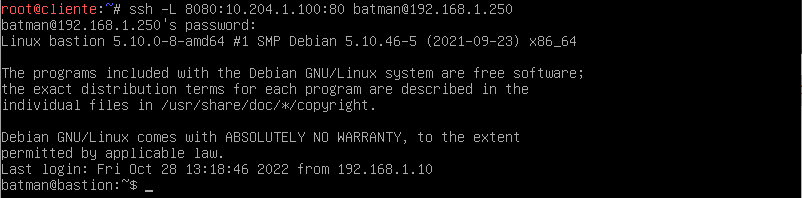
Ilustración 7: Comprobamos ssh

Como hemos permitido que el usuario Batman es el que pueda acceder a la maquina bastión cuando entramos con otro usuario nos saldrá permiso denegado.

# 7. Comprobamos túneles

Lo que tenemos que saber es que si queremos abrir un túnel tiene que ser un número superior al 1024 y mínimo al 65535, lo recomendable es si tenemos en servicio apache que está ejecutándose en el puerto 80 nosotros consultaremos en el 8080.

~# ssh -L 8080:10.204.1.100:80 batman@192.168.1.250

Ilustración 8: abrimos túnel

En este caso estamos abriendo un túnel local por eso el parámetro -L y también la sintaxis es el puerto al que vamos a consultar la página web, el segundo es la ip del servidor apache y el tercero es el puerto donde está ejecutándose apache y para finalizar como si nos conectáramos a bastión con ssh.

Ahora lo que haremos será abrir una nueva sesión que sería con el “ctrl” de la derecha y las teclas de funciones.

~# lynx localhost:8080

Ilustración 9: Conexión mediante lynx

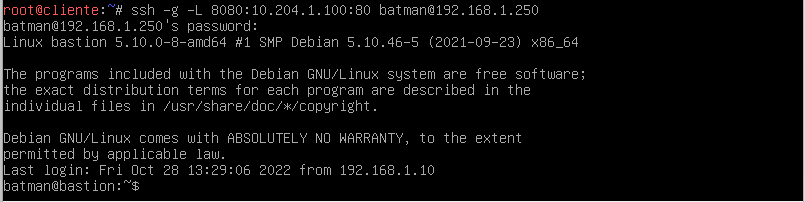
Vemos que ha cargado otra página distinta, pero es en mi caso que modifique el archivo del sitio de apache para que aparezca otro sitio distinto que sería modificarlo en el /var/www/HTML, y en el index.html y si nos hemos descargado una página web y queremos guardarla en el servidor lo recomendable es añadir otra interfaz de red, instalar ssh y copiar el o los archivos.

# 8. Comprobaciones finales

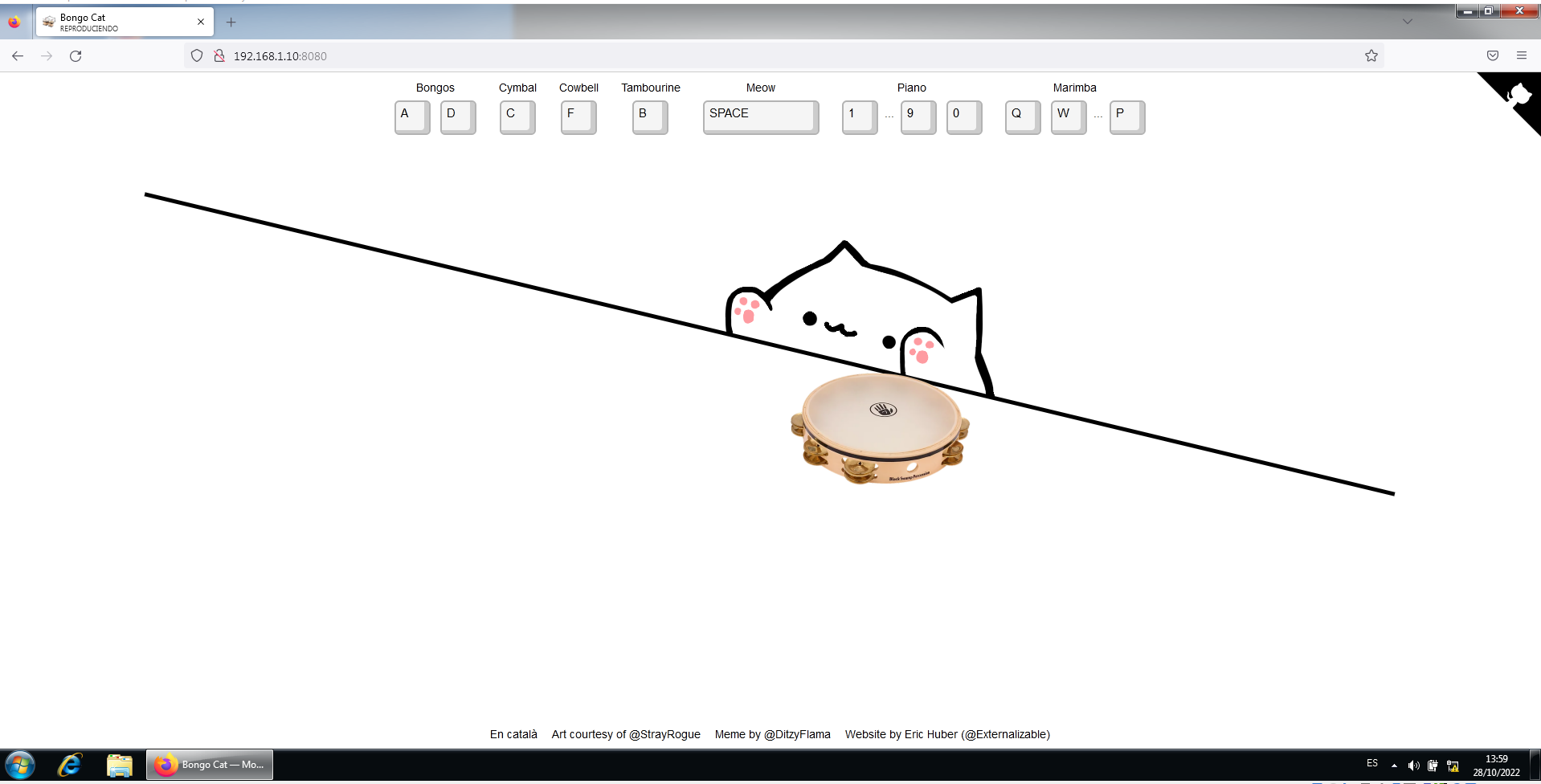
ahora lo que nos pide es que desde el cliente Windows escribamos la ip del cliente debían y el puerto 8080 y cargue la página de server.

que sería haciendo lo mismo que antes, pero añadiendo el parámetro -g entre ssh y el parámetro -L.

~# ssh -g -L 8080:10.204.1.100:80 batman@192.168.1.250

Ilustración 10: cliente como puerta de enlace

ahora a través de la maquina Windows tendremos que ver la página de server.

Ilustración 11: Windows visualizando la página web de server

# 9. Socks

Un servidor proxy podemos definirlo como un servidor, programa o dispositivo que hace la función de intermediación en las peticiones de recursos que realiza un cliente A hacia otro servidor C. Un típico ejemplo, es cuando una máquina A realiza una petición a C. La forma de hacerlo es mediante una petición a B que trasladará la petición a C, por lo que este último no va a conocer que la petición original provenía de A. Luego, también la respuesta pasara por B, que es el proxy y se le ofrecerá al usuario A, que fue quien realizó la petición.

Entre las ventajas del servidor proxy están el control, ya que pueden restringir los derechos de los usuarios. También la velocidad, ya que puede hacer caché y guardar una respuesta para cuando otro usuario lo solicite. No nos olvidemos del anonimato, puesto que nos permite conectarnos de forma anónima a un recurso externo sin tener que revelar nuestra IP. A fin de cuentas, esto lo logramos al acceder a la red a través de un intermediario.

Lo que haremos será instalar “vsftpd” en la maquina server no sin antes incluir o un nuevo interfaz en modo “nat” o en modo “puente”, o usar el mismo interfaz.

~# apt update ; apt install vsftpd -y

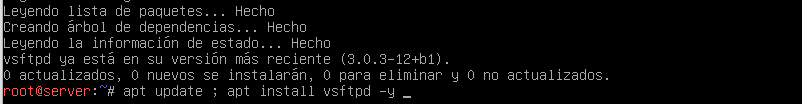


Ilustración 12: instalamos vsftpd

Ahora lo que haremos será usar putty y FileZilla para usar Socks.

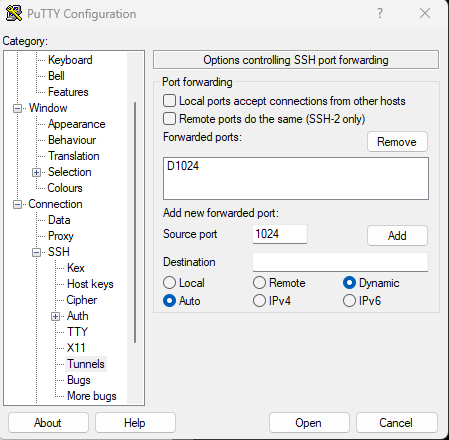


Ilustración 13: Abrimos túnel dinámico

Lo que haremos será configurar los parámetros en este caso lo que haremos será crear un túnel dinámico, en nuestro caso usaremos el puerto 1024 pero podemos usar otro distinto.

Y luego le daremos a “sesión” y pondremos la ip de la maquina bastión.

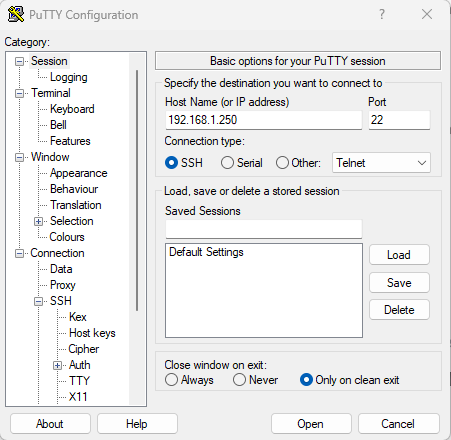


Ilustración 14: Conectamos por putty

Y le daremos a “open” para abrir una sesión.

Si la maquina la hemos conectado anteriormente no nos saldrá la pestaña de seguridad.

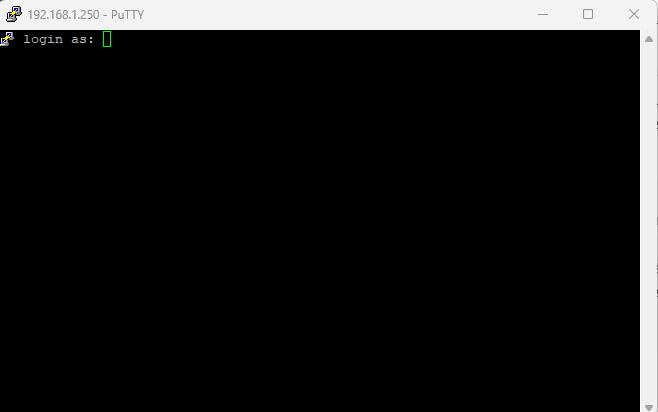


Ilustración 15: Abrimos nueva sesión

Escribiremos el usuario y la contraseña que en nuestro caso es Batman.

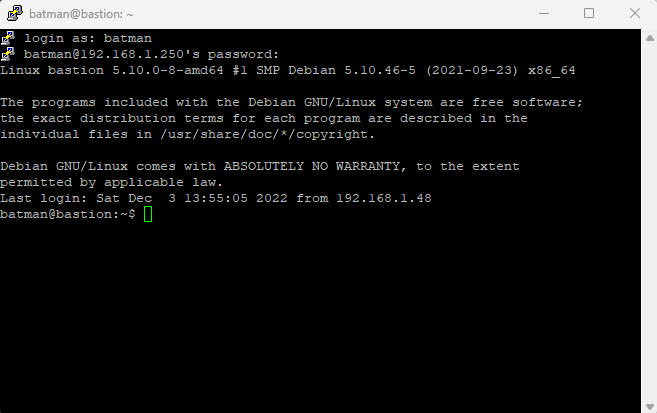


Ilustración 16: sesión abierta

Ahora lo que haremos será entrar en las opciones de FileZilla y configurar los Socks

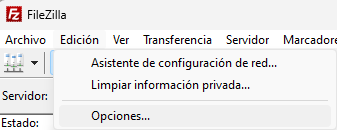


Ilustración 17: Configuramos FileZilla

Una vez en las opciones le daremos a Proxy genérico.

Habilitamos el “SOCKS 5” y pondremos “127.0.0.1” que es localhost y el puerto proxy que en nuestro caso es 1024.

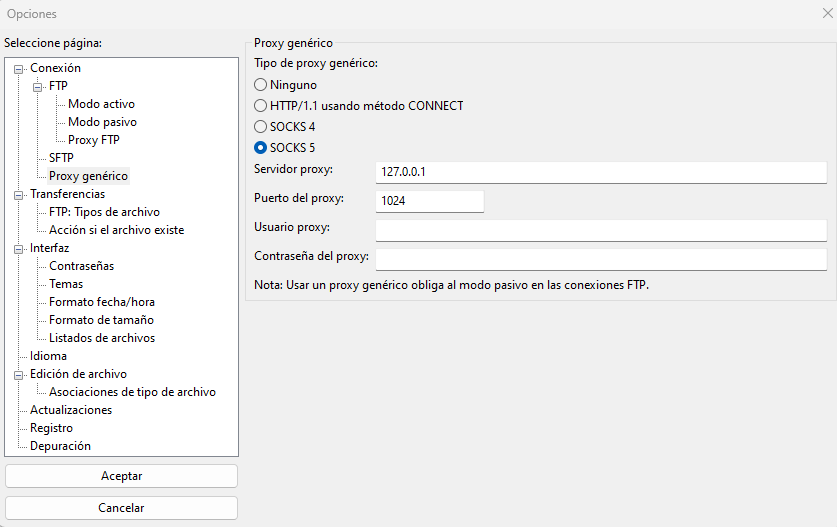


Ilustración 18: SOCKS

Le daremos a aceptar y pondremos la ip de la maquina servidor, el nombre de usuario que en nuestro caso es Robin y la contraseña supersegura que es Robin.

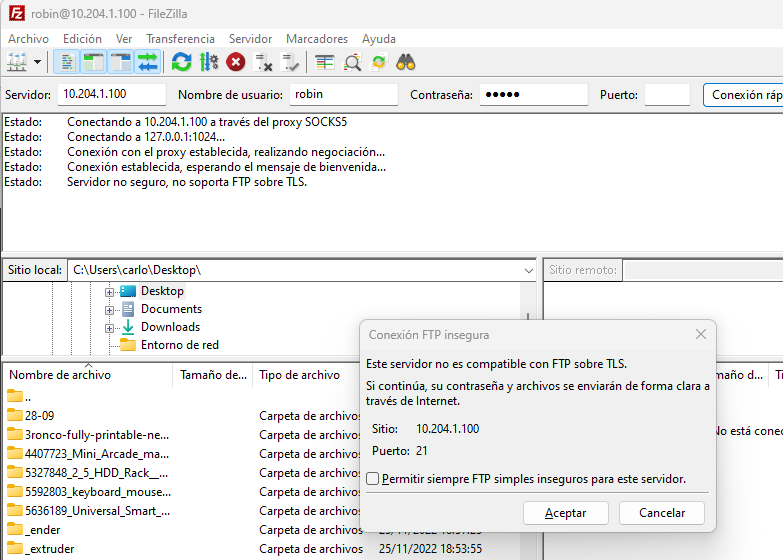


Ilustración 19: Advertencia de seguridad

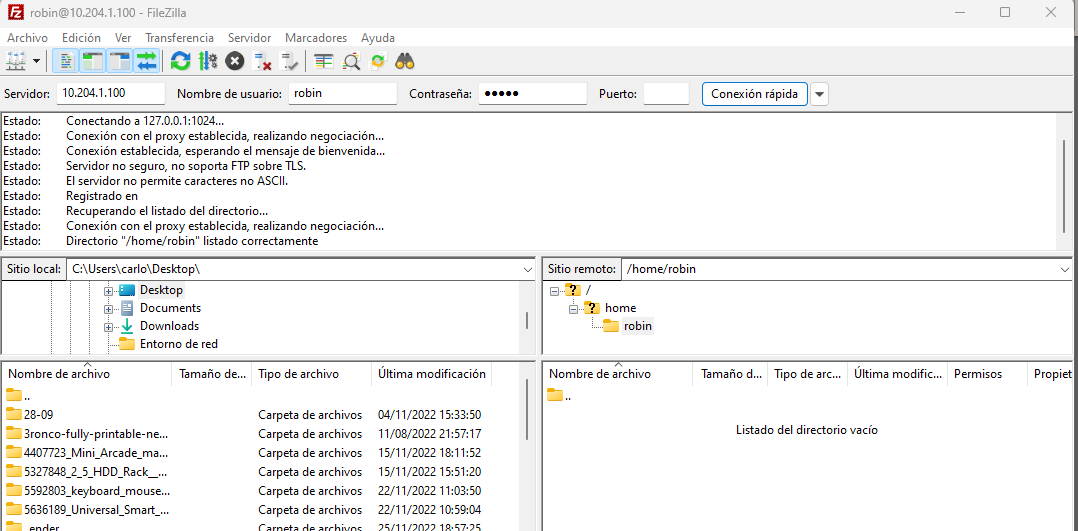


Ilustración 20: Conexión exitosa

Vemos que se ha conectado, pero no en el directorio correcto.

¡OJO! Tenemos que cerrar sesión en FileZilla para poder hacer los siguientes pasos correctamente.

Ahora cambiaremos el directorio de conexión con el siguiente comando.

~# usermod -d /var/www/html robin



Ilustración 21: Cambiamos directorio conexión

Y revisaremos en el “/etc/passwd” si se han aplicado los cambios.

~# cat /etc/passwd

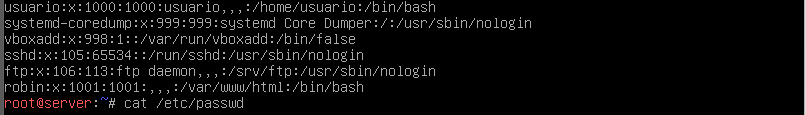


Ilustración 22: Comprobamos los cambios

Y nos conectaremos ahora si con FileZilla y comprobaremos si funciona.

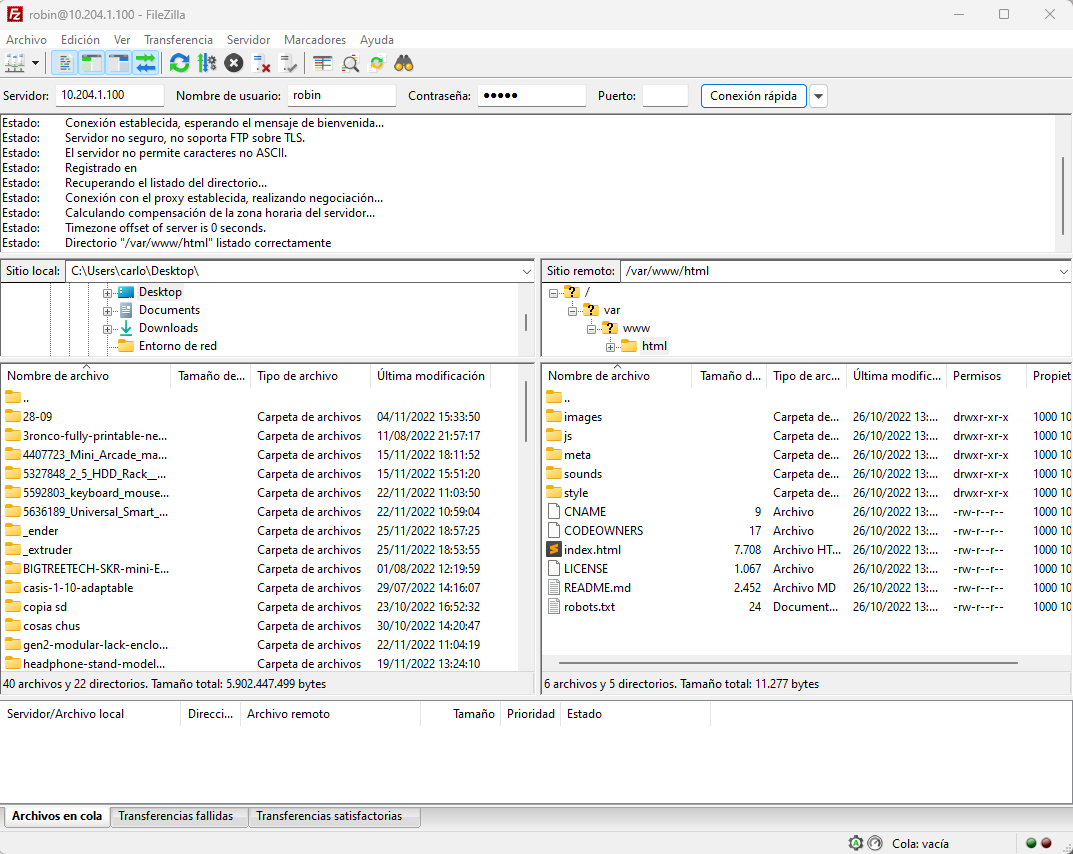


Ilustración 23: Conexión exitosa al nuevo directorio

Ahora nos descargaremos el “index.html” y vemos si nos lo transfiere correctamente o no.

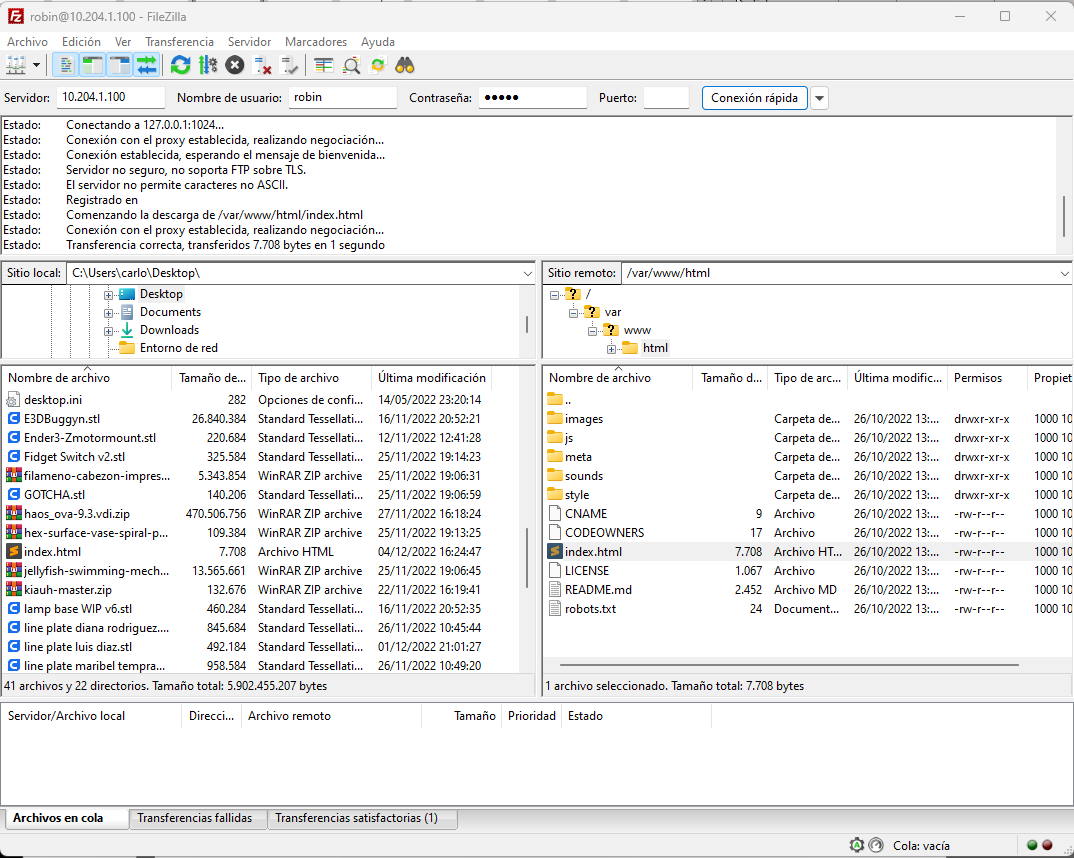


Ilustración 24: transferencia exitosa

Como podemos comprobar nos hemos podido transferir correctamente el archivo desde la maquina server hasta nuestro Windows usando SOCKS.

# 10. Webgrafía

<https://www.redeszone.net/tutoriales/internet/proxy-socks5-que-es-ventajas/>