



OpenSSH es una implementación libre y de código abierto del protocolo SSH (Secure Shell).

SSH

SSH es un protocolo de red seguro que permite a los usuarios conectarse y administrar de forma remota sistemas y servicios a través de una conexión encriptada.

OpenSSH proporciona una suite de herramientas y utilidades que permiten la autenticación segura, el cifrado de datos y la comunicación remota en entornos distribuidos.

Características y Componentes

- **Servidor SSH:** OpenSSH incluye un servidor SSH que se ejecuta en el host remoto y permite a los usuarios autenticarse y establecer sesiones de shell seguras y conexiones de transferencia de archivos.
- **Cliente SSH:** OpenSSH también proporciona un cliente SSH que se ejecuta en la máquina local y permite a los usuarios establecer conexiones seguras con hosts remotos, autenticarse y realizar diversas operaciones remotas.

Características y Componentes

- **Autenticación:** OpenSSH admite varios métodos de autenticación seguros, como clave pública, clave privada, contraseña, autenticación basada en Kerberos y autenticación de tarjetas inteligentes.
- **Túneles SSH:** OpenSSH permite crear túneles seguros a través de conexiones SSH para enrutar el tráfico de red, proporcionando una forma segura de acceder a servicios remotos a través de redes no seguras o restringidas.

Características y Componentes

- **Transferencia de archivos:** OpenSSH incluye la utilidad scp (Secure Copy) para copiar archivos de forma segura entre hosts locales y remotos. También proporciona sftp (SSH File Transfer Protocol) para transferir archivos de manera segura utilizando una interfaz similar a FTP.

Ejemplos prácticos

- **Conexión SSH remota:** Puedes utilizar el cliente SSH de OpenSSH para conectarte a un servidor remoto. Por ejemplo, puedes ejecutar el siguiente comando para iniciar una sesión SSH en un servidor remoto llamado "example.com" con el nombre de usuario "usuario":

```
$ > ssh usuario@example.com
```

Ejemplos prácticos

- ***Transferencia de archivos segura:*** Puedes utilizar el comando scp de OpenSSH para copiar archivos de forma segura entre hosts locales y remotos. Por ejemplo, para copiar un archivo llamado "archivo.txt" desde el servidor remoto a tu máquina local, puedes ejecutar el siguiente comando:

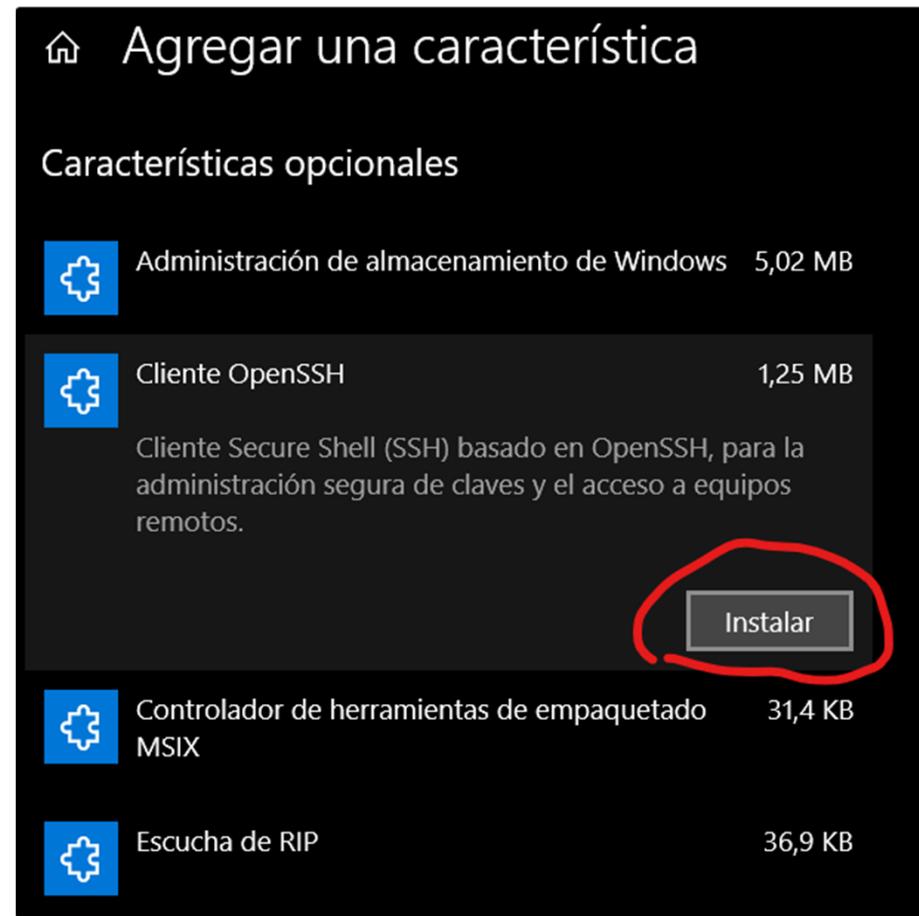
```
$> scp usuario@example.com:/ruta/del/archivo.txt /ruta/local/
```

Ejemplos prácticos

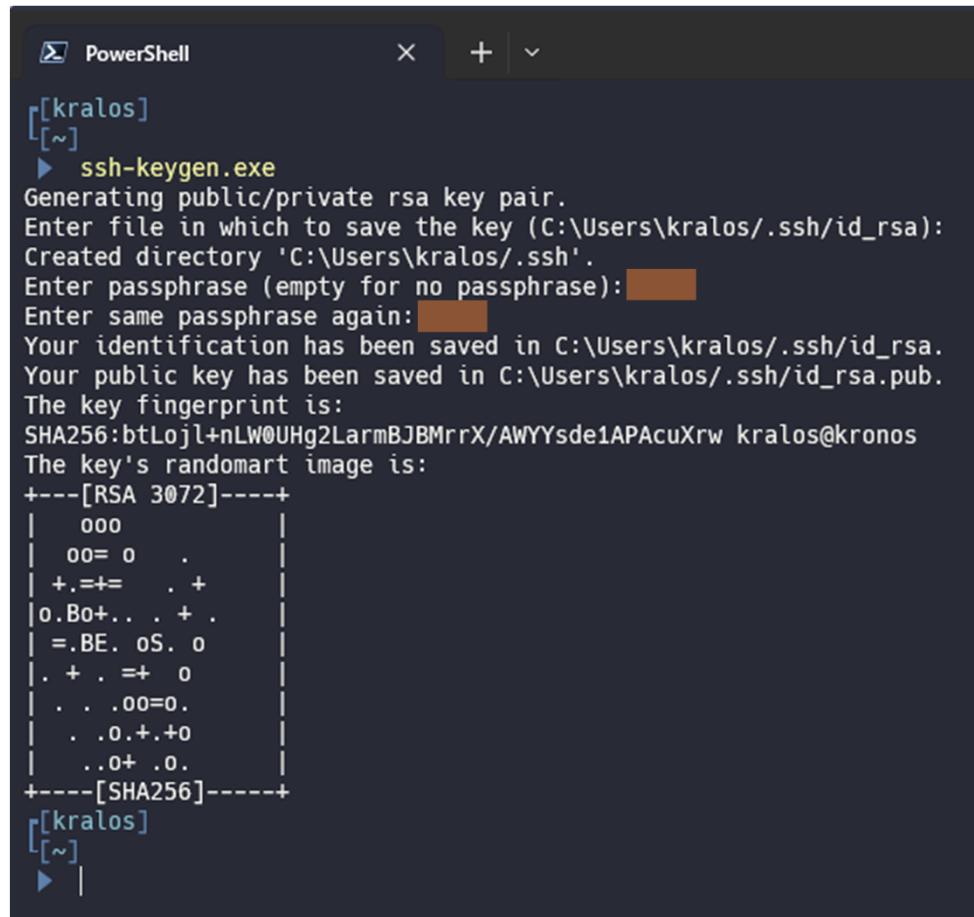
- **Túneles SSH:** OpenSSH permite crear túneles SSH para redirigir el tráfico de red. Por ejemplo, puedes utilizar un túnel SSH para acceder de forma segura a un servicio remoto, como una base de datos, a través de una conexión segura. El siguiente comando crea un túnel SSH que redirige el tráfico del puerto 3306 del servidor remoto al puerto 3306 de tu máquina local:

```
$> ssh -L 3306:localhost:3306 usuario@example.com
```

Ejemplos prácticos en windows



Ejemplos prácticos en windows



```
[kralos]
[~]
▶ ssh-keygen.exe
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\kralos/.ssh/id_rsa):
Created directory 'C:\Users\kralos/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\kralos/.ssh/id_rsa.
Your public key has been saved in C:\Users\kralos/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:btLojl+nLW0UHg2LarmBJBMrrX/AWYYsde1APAcuXrw kralos@kronos
The key's randomart image is:
+---[RSA 3072]---+
|   ooo
|   oo= o   .
|   +.=+= . +
| o.Bo+... . +
|   =.BE. oS. o
|   . + . += o
|   . . .oo=o.
|   . . o.+..o
|   ..o+ .o.
+---[SHA256]---+
[kralos]
[~]
▶ |
```

Ejemplos prácticos en windows

```
[kralos]
[~\.ssh]
▶ ls

Directory: C:\Users\kralos\.ssh

      Mode          LastWriteTime        Length Name
-----          -              -           ----- 
-a---   11/14/2023  4:49 PM       2655 id_rsa
-a---   11/14/2023  4:49 PM        568 id_rsa.pub

[kralos]
[~\.ssh]
▶ |
```

```
[kralos]
[~\.ssh]
▶ type .\id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgQDQyt4VPj/KNMxpXLgy0b/VgZW1LmL18BKjn8XDMUlMyNXgJsEPtNhFDzqlKUa
mmNNYYXVHOrNFnePttxcLACZSKjL2BJL0Vnf3csY40cPCuhrYfDLv2bxpjzFY+F2yLKNh9dknZc+cFbeUaW/OAf1m4XaD73lob6S4
7CMSOZx0fZnWGlmwk0om0+6dklZrE+EL2CrwSinsS7RrMU0iLNY6UtspeTBV0ig9vTj5r88B/jxw+QzxhUFva5dAHVkARFnN5xn
bLrb4to6ALMUjgEYfiwWjGa3+DBS3BhTdWb7pyU5rXsUMDyjnGXoXXvY7LazUrxK2vEYBg55W8qiP1owq52dQhAXwC2YwPefjASz
/epweIeXN2pw0aNqzdYOPAgQmQk3D6NlqgRyVT5G00Tm64cggl7Lb253rj29bwsatDyloDWYNwCzEnmNWh0cSVD7L5G0/4FIHako
iLUU2rhwB6P8EQ5aii5lITgCKuXpyR0jPYJhzr5BiHN0ZKcD/Jc= kralos@kronos
[kralos]
[~\.ssh]
▶ |
```

Ejemplos prácticos en windows

```
[kralos]
[~\.ssh]
▶ type $env:USERPROFILE\.ssh\id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQDQyt4VPj/KNMXPXLgy0b/VgZW1LmL18BKjn8XDMU1MyNXgJsEPtNhbFDzqlKUa
mmNNYYXVHOrNFnePttxLACZSKjL2BJL0Vnf3csY40cPCuhrYfDLv2bxpjzFY+F2yLKNh9dknZc+cFbeUaW/OAf1m4XaD73lob6S4
7CMOSZx0fZnWGlmtwk0om0+6dklZrE+EL2CrwSinsS7RrMU0iLNY6UtspeTBV0ig9vTj5r88B/jxw+QzxhUFva5dAHVkARFnN5xn
bLrb4to6ALMuJgEYfiWjGa3+DBS3BhTdWb7pyU5rXsUMDyjnGXoXXvY7LazUrxEK2vEYBg55W8qiP1owq52dQhAXwC2YwPefjASz
/epweIeXN2pw0aNqzdYOPAgQmQk3D6NlqgRyVT5G00Tm64cggl7Lb253rj29bWsatDylodWYNwCzEnmNWh0cSVD7L5G0/4FIHako
iLUU2rhwB6P8EQ5aii5lITgCKuXpyR0jPYJhzr5BiHN0ZKcD/Jc= kralos@kronos
[kralos]
[~\.ssh]
▶ |
```

```
type $env:USERPROFILE\.ssh\id_rsa.pub | ssh {USER@IP-ADDRESS} "cat >> .ssh/authorized_keys"
```

```
[kralos]
[~]
▶ type $env:USERPROFILE\.ssh\id_rsa.pub | ssh kralos@192.168.7.108 "cat >> .ssh/authorized_keys"
The authenticity of host '192.168.7.108 (192.168.7.108)' can't be established.
ECDSA key fingerprint is SHA256:4cViKEtyr0kdZfy544ba0ZFLzfMprUW4YI0qdixqV+4.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.7.108' (ECDSA) to the list of known hosts.
kralos@192.168.7.108's password: [REDACTED]
[kralos]
[~]
▶
```

Ejemplos prácticos en windows

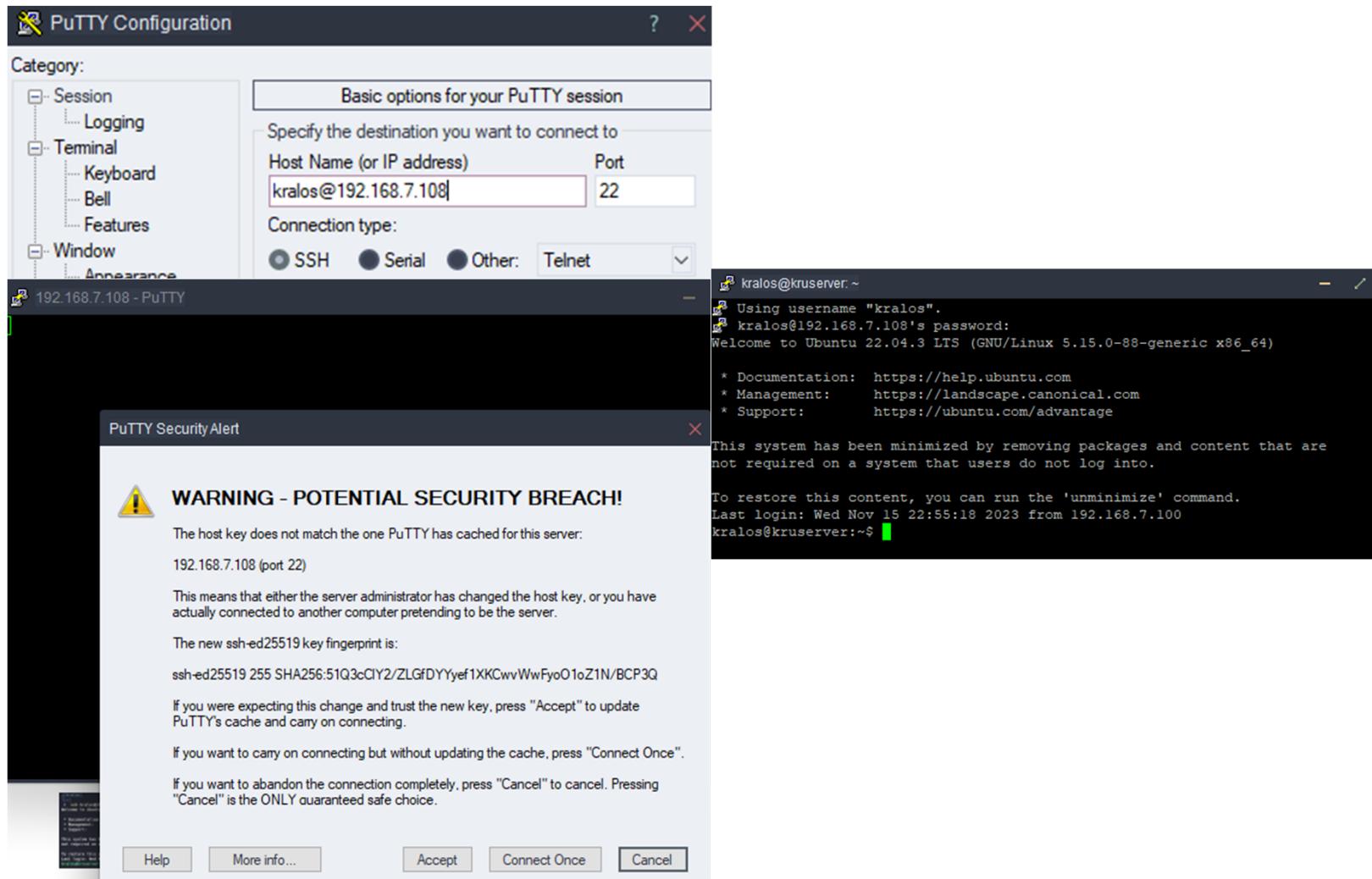
```
[kralos]
[~]
▶ ssh kralos@192.168.7.108
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-88-generic x86_64)

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

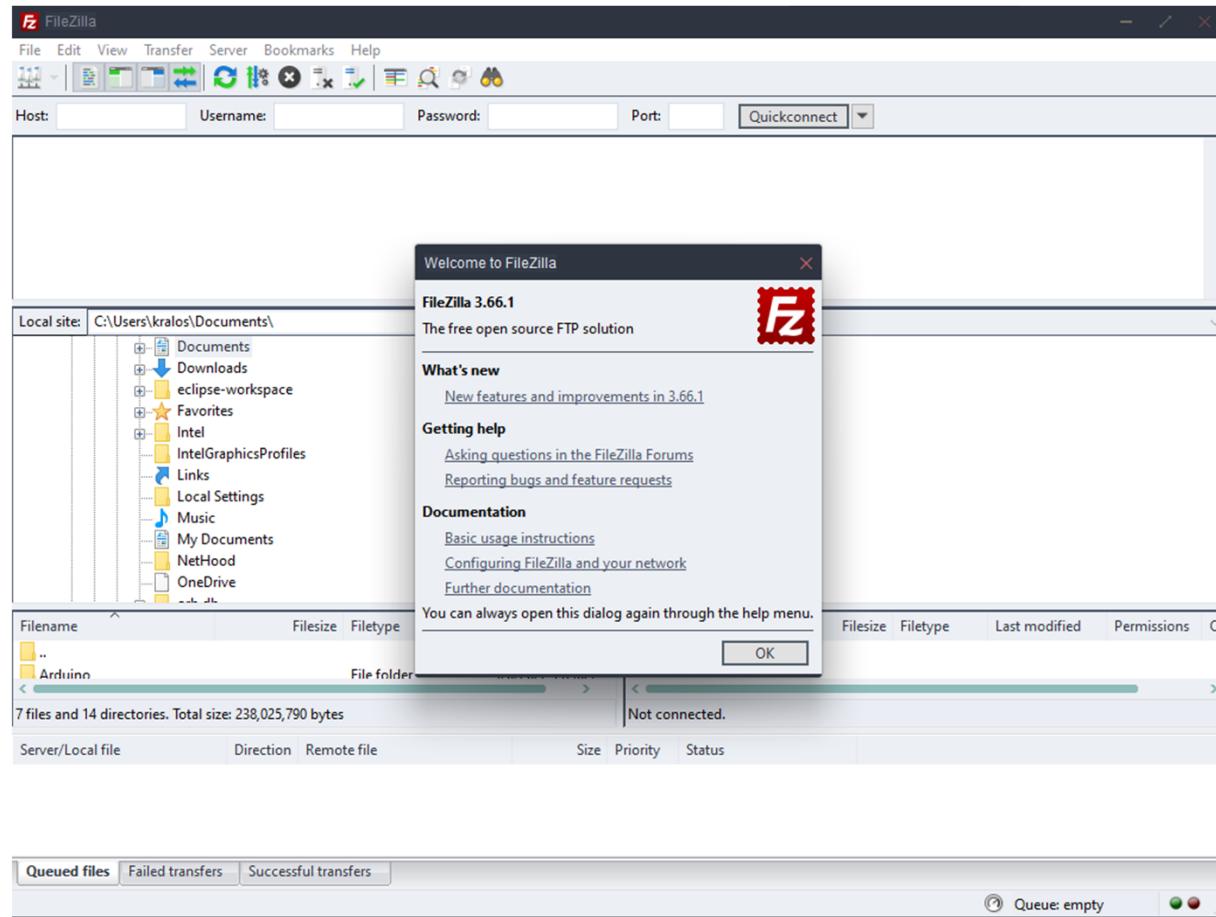
This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Wed Nov 15 18:49:18 2023 from 192.168.7.103
kralos@kruserver:~$ |
```

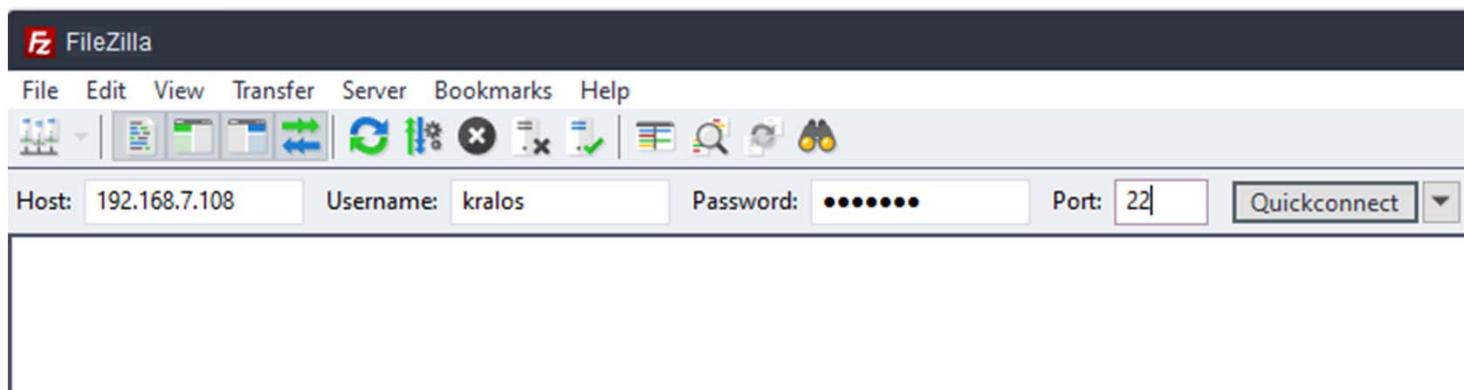
Ejemplos prácticos en windows



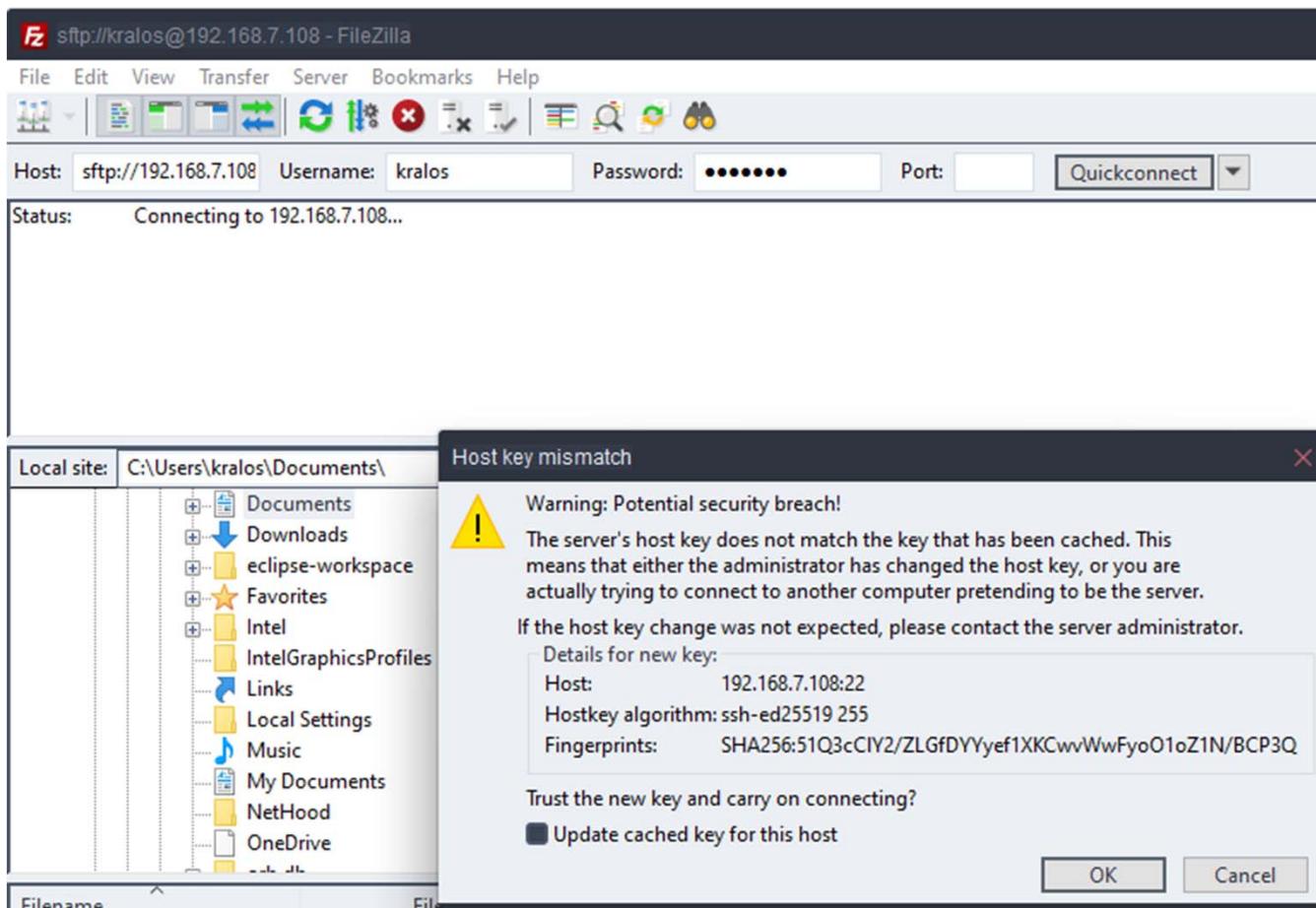
Ejemplos prácticos en windows



Ejemplos prácticos en windows



Ejemplos prácticos en windows



Ejemplos prácticos en windows

