

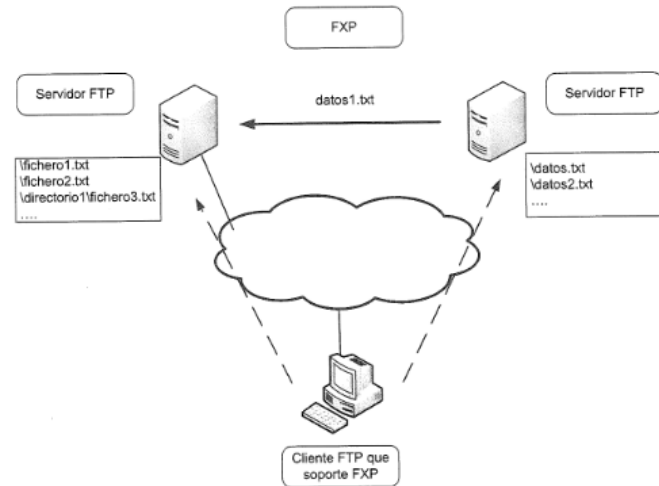
Índice:

Ejercicio 1	2
a) Informe protocolo FXP	2
b)	4

Ejercicio 1

a) Informe protocolo FXP

Solución:



FXP (File eXchange Protocol) es un protocolo que permite la transferencia directa de archivos entre dos servidores FTP (File Transfer Protocol) sin necesidad de pasar por el cliente del usuario.

Ventajas de FXP:

- ✚ **Velocidad de transferencia:** FXP permite la transferencia directa de archivos entre dos servidores, lo que puede resultar más rápida que descargar archivos a través del cliente y luego cargarlos en el otro servidor.
- ✚ **Eficiencia de ancho de banda:** Al evitar el paso por el cliente del usuario, FXP reduce el uso del ancho de banda del usuario, ya que la transferencia se realiza directamente entre servidores.
- ✚ **Menor carga en el cliente:** El cliente del usuario no participa activamente en la transferencia de archivos, lo que significa que la carga del servidor cliente es menor en comparación con las transferencias tradicionales de FTP.

Desventajas de FXP:

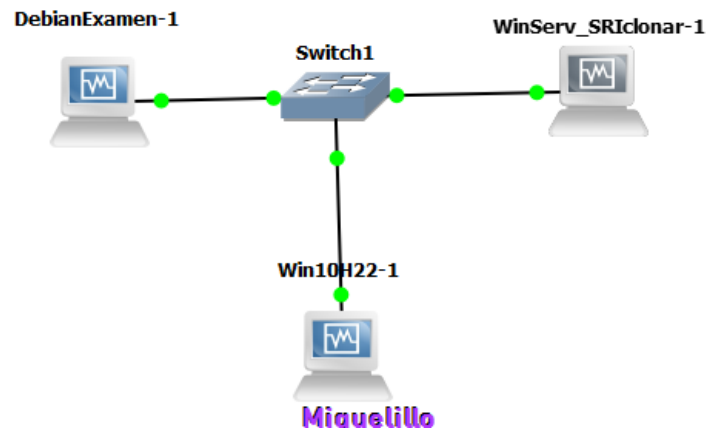
- ✚ **Requisitos del servidor:** Ambos servidores FTP deben admitir y estar configurados para permitir las transferencias FXP. Algunos servidores pueden deshabilitar esta función por razones de seguridad.
- ✚ **Firewalls y NAT:** Puede haber problemas con firewalls y Network Address Translation (NAT) que dificulten la conexión directa entre los servidores. Se deben configurar adecuadamente para permitir las transferencias FXP.

- ✚ **Seguridad:** FXP puede presentar riesgos de seguridad, ya que la transferencia de archivos entre servidores sin intervención del cliente puede ser explotada con exploits si no se toman las precauciones adecuadas.
- ✚ **Limitaciones de comandos:** Algunos comandos específicos de FTP pueden no ser compatibles con FXP, lo que podría limitar ciertas funciones o características.

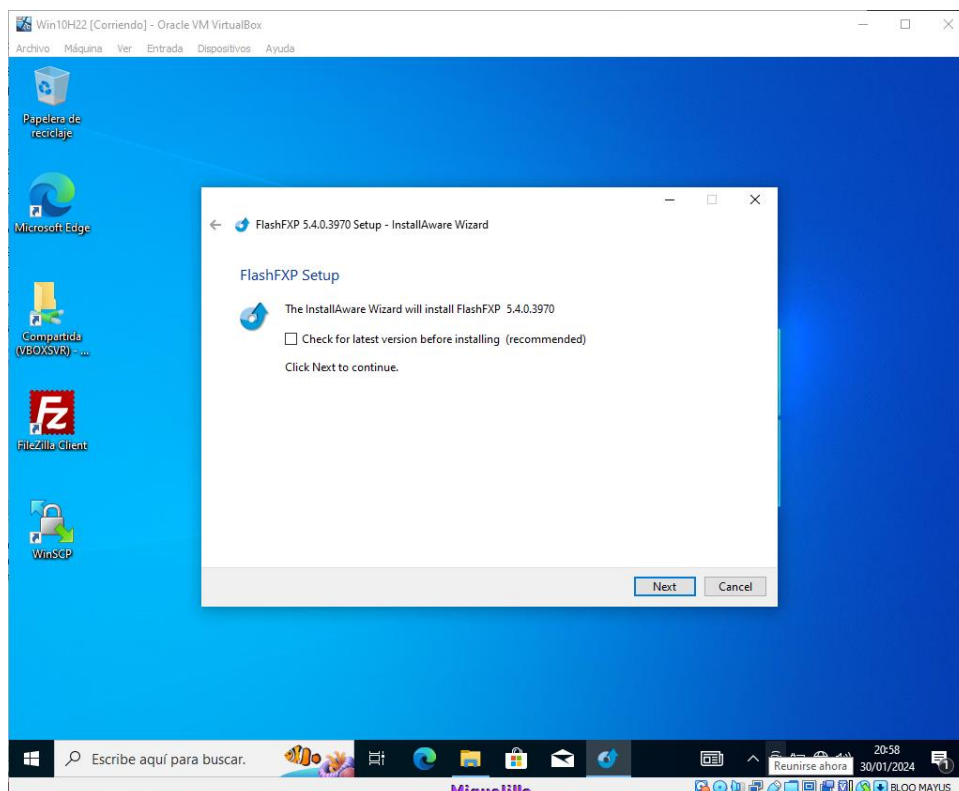
b) Implementar un escenario FXP de transferencia de ficheros entre servidores FTP

Solución:

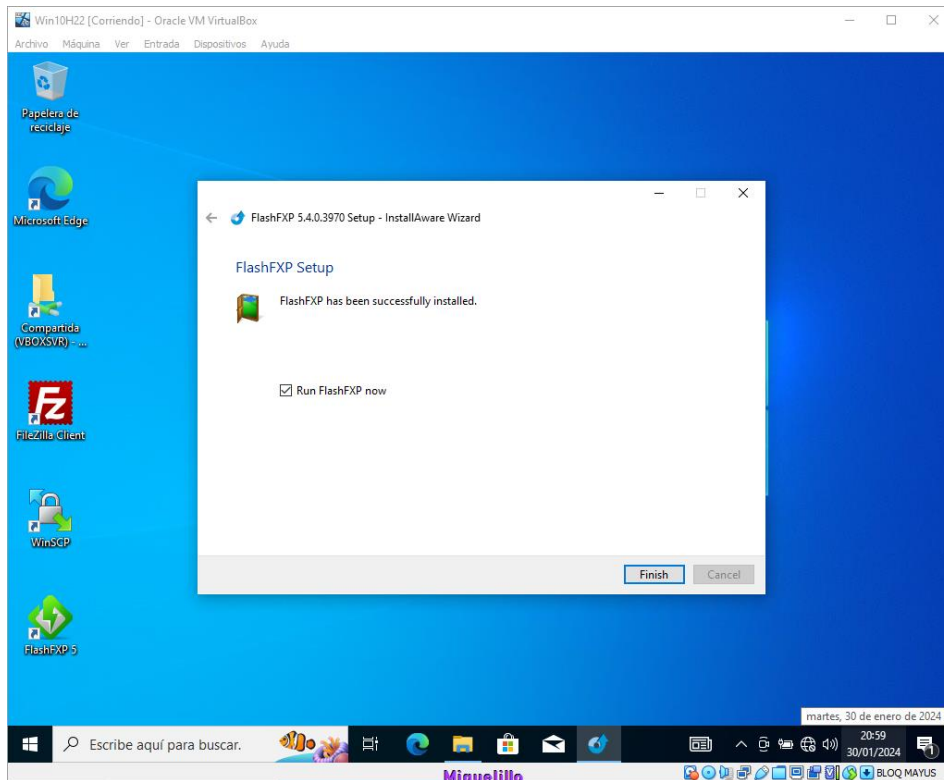
Escenario:



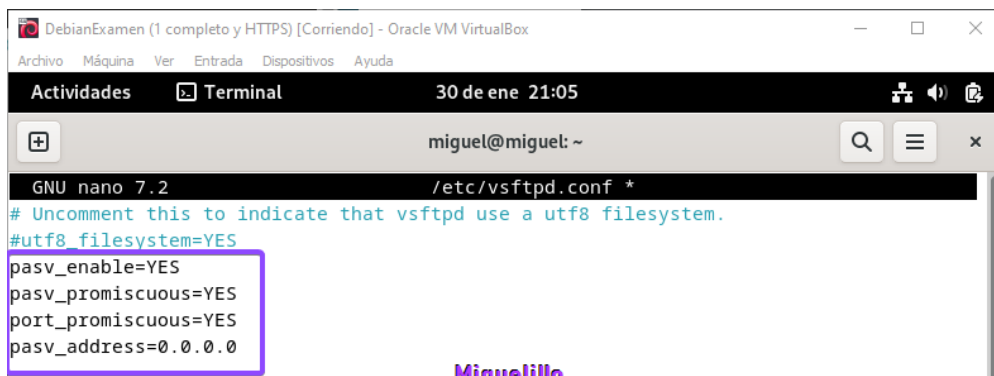
Instalamos el cliente **FlashFXP** con una instalación por defecto



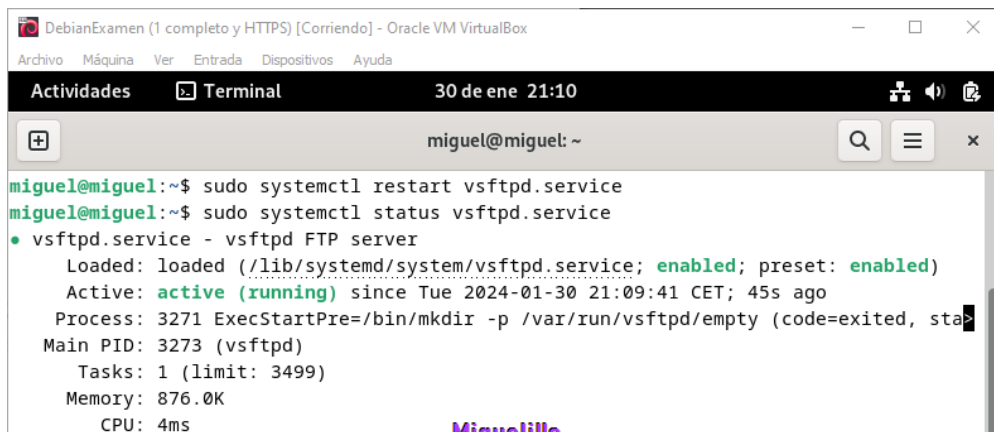
Se termina de instalar



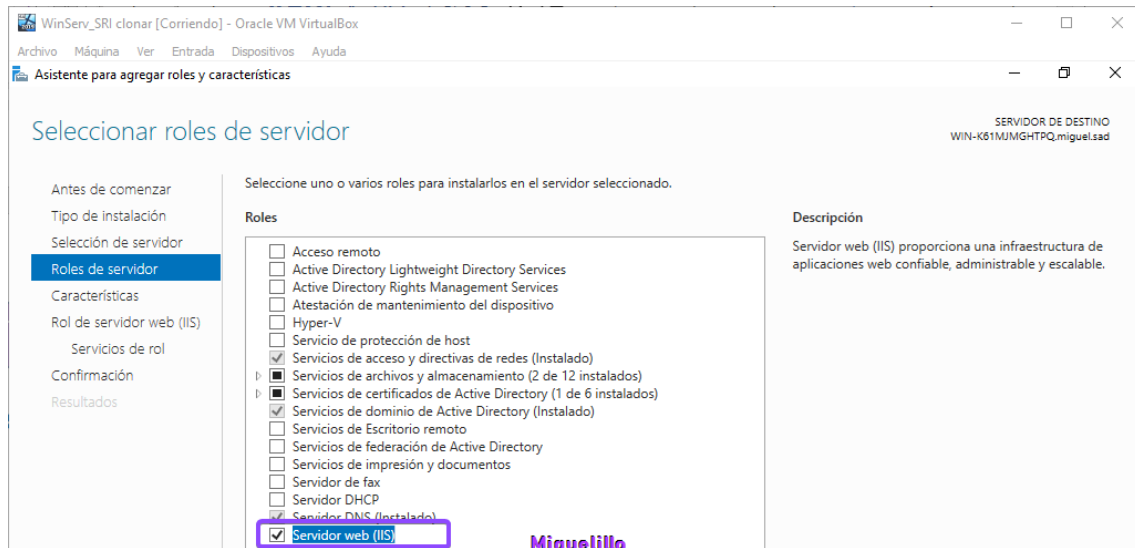
Vamos a configurar el servidor de debian



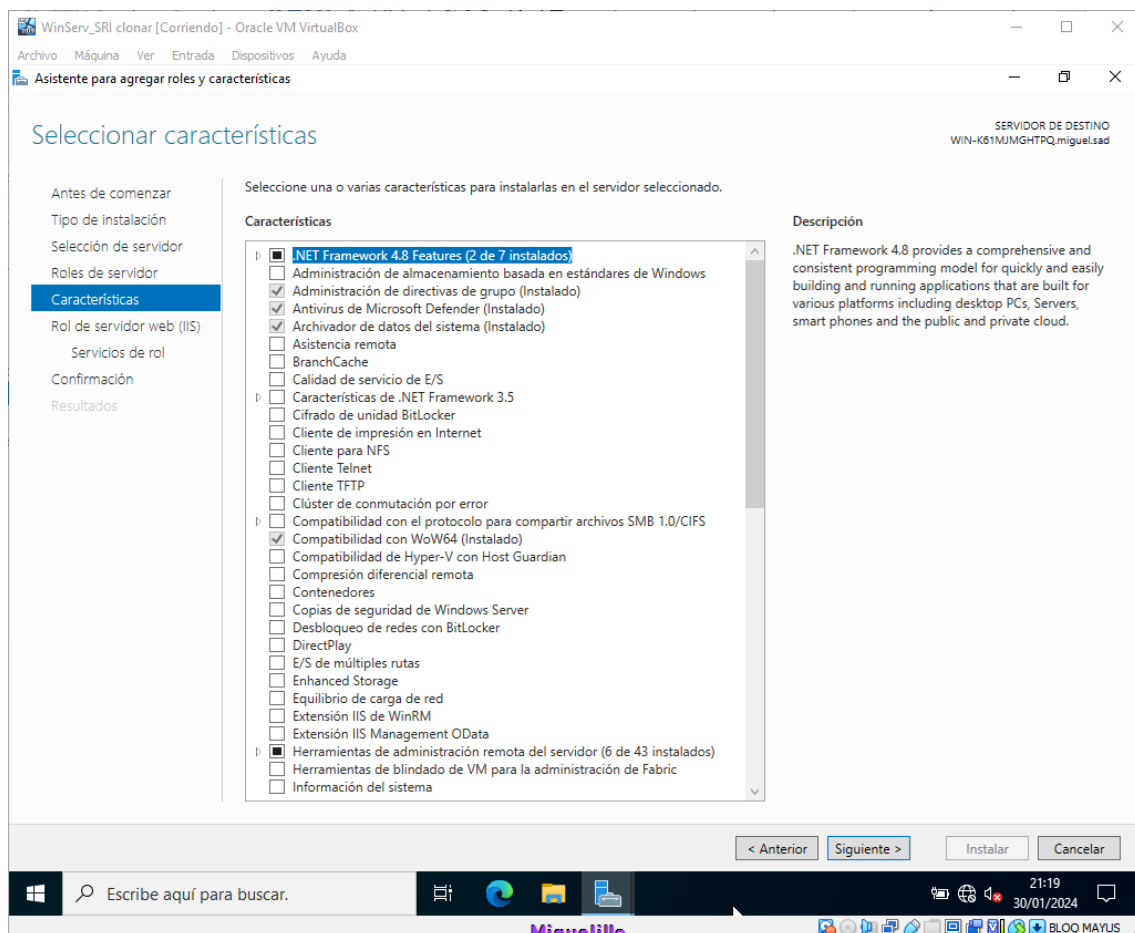
Reiniciamos el servicio y vemos le estado



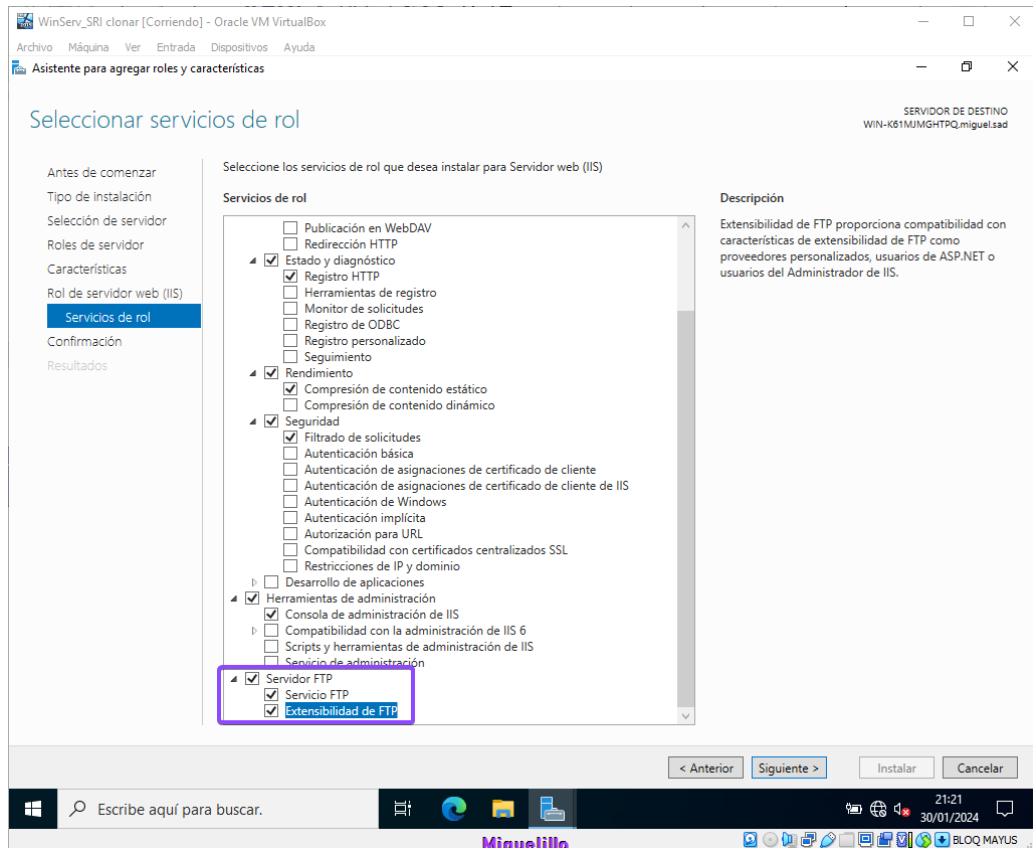
Vamos a instalar FPT y FXP en Windows Server Instalamos el rol de Servidor web(IIS)



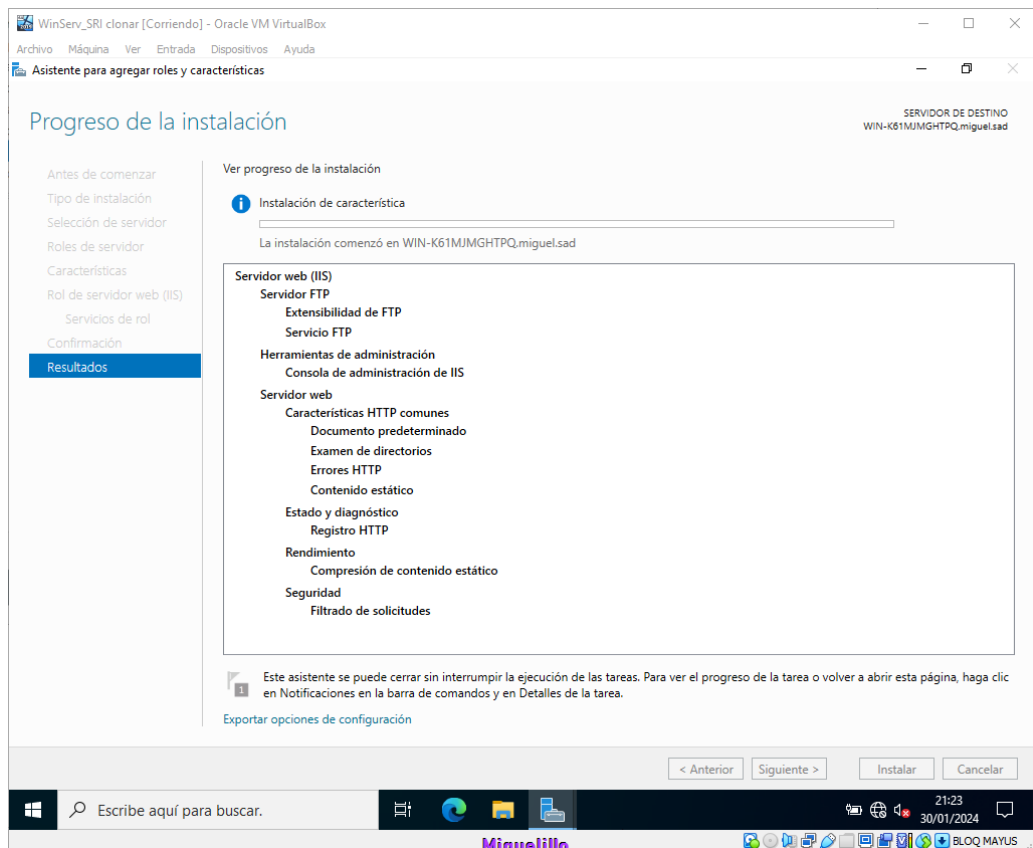
Lo dejamos por defecto



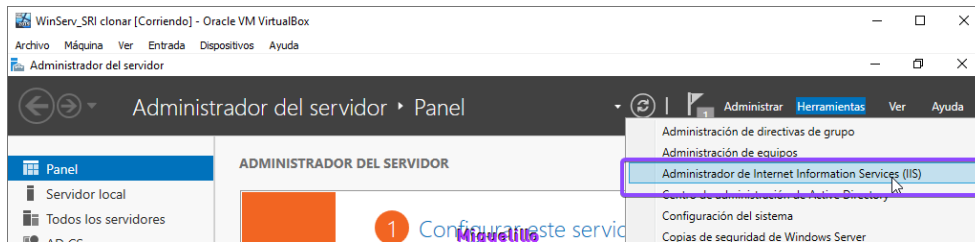
Seleccionamos los servicios de FTP



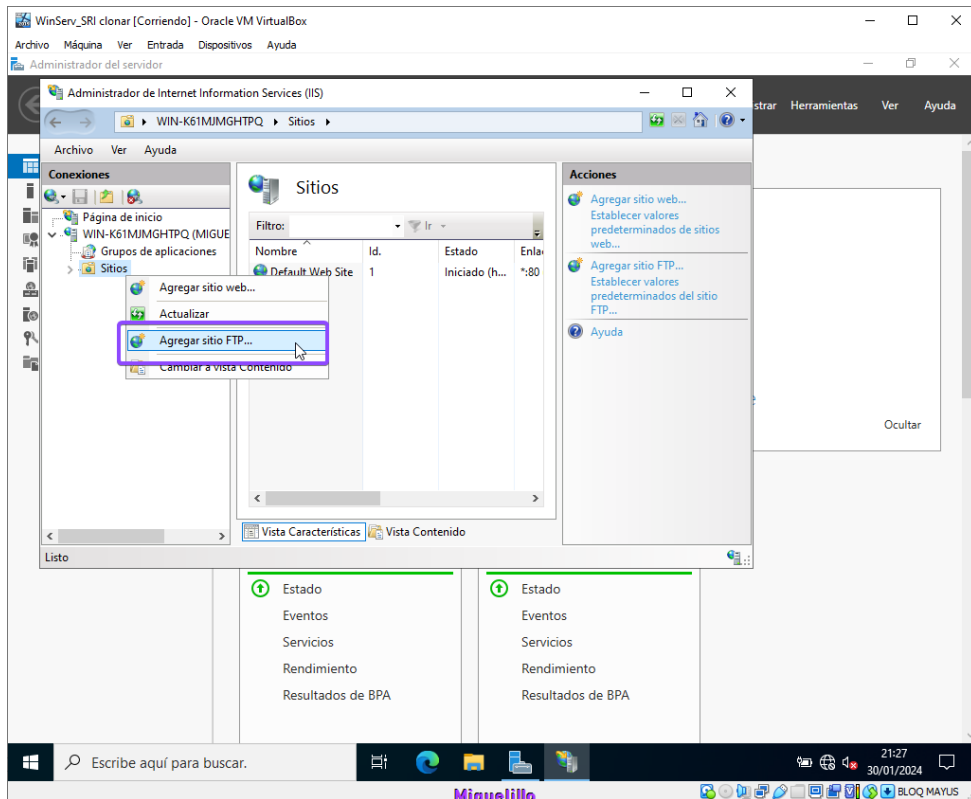
Esperamos que se instale



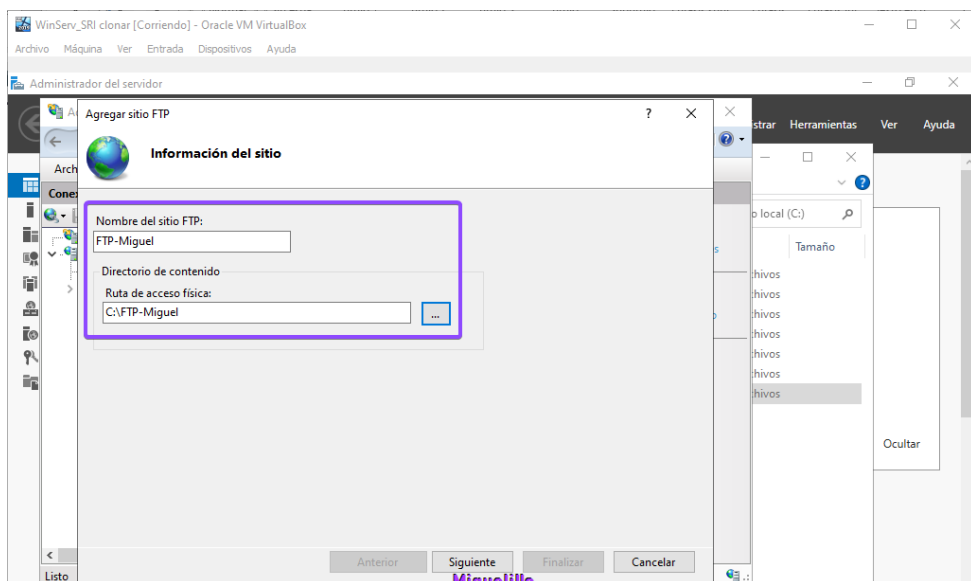
Vamos a Herramientas → Administrador de Internet Information Service (IIS)



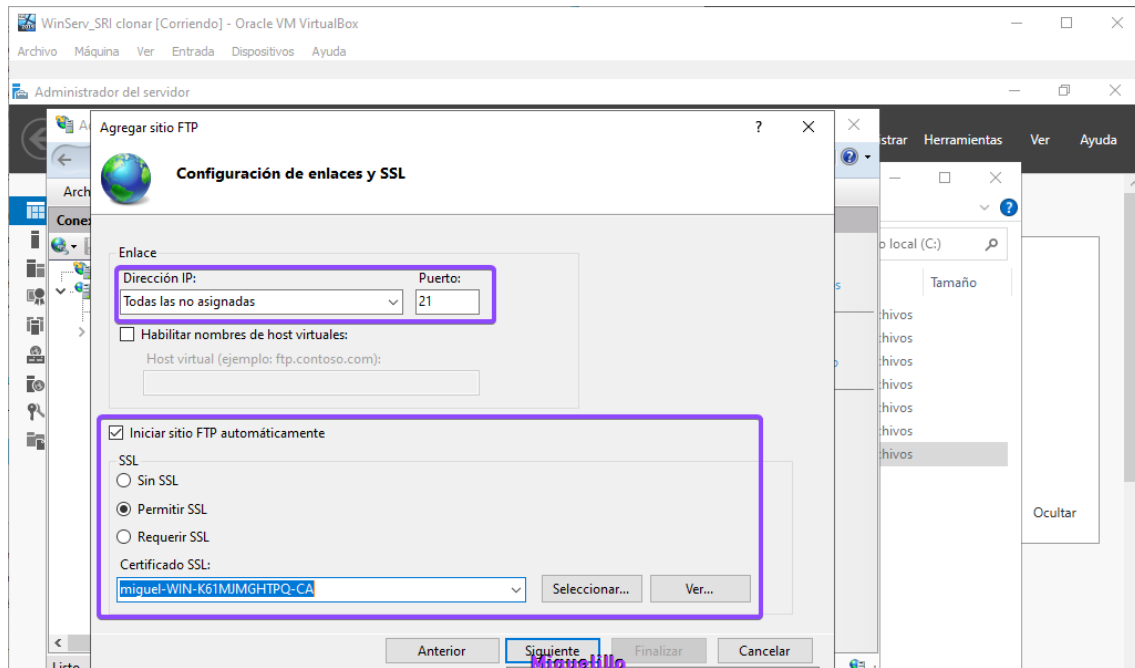
Clicamos en agregar un sitio FTP



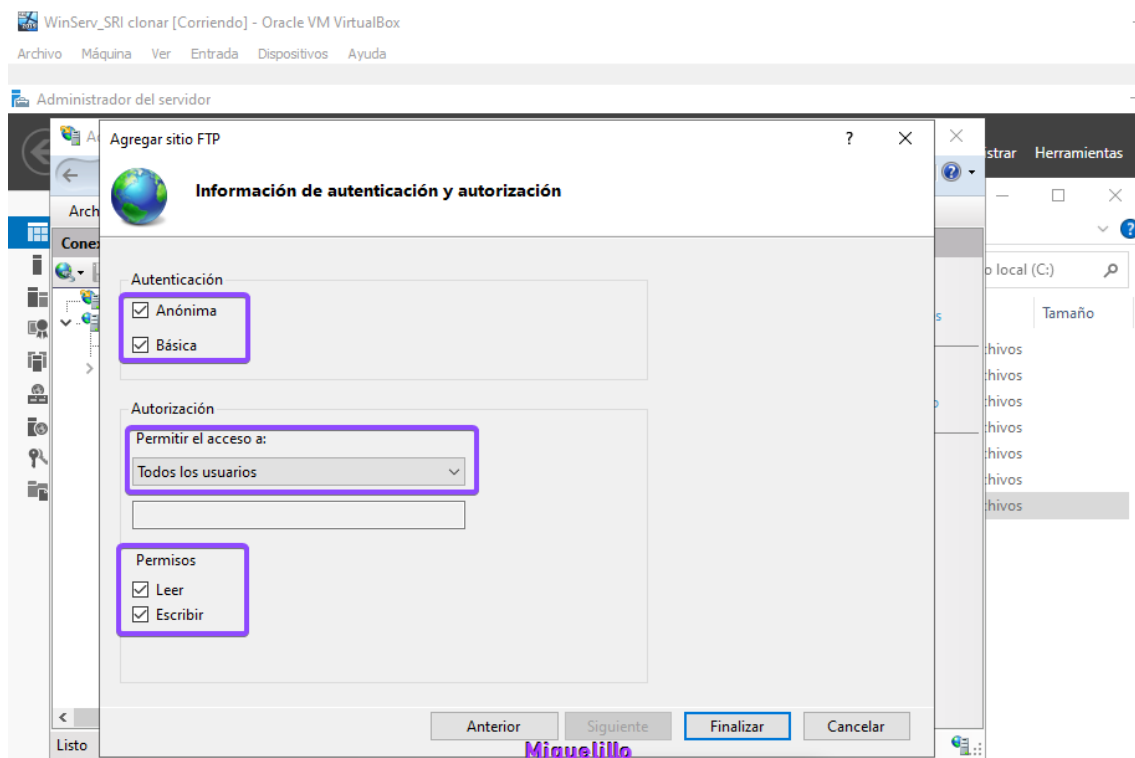
Le ponemos el nombre y la ruta de acceso



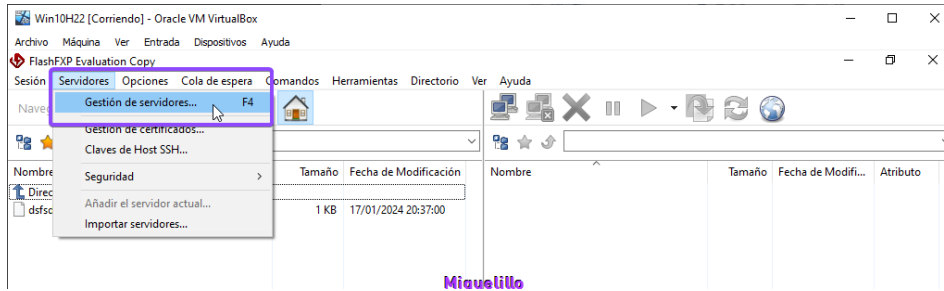
Configuramos el acceso al FTP y los certificados



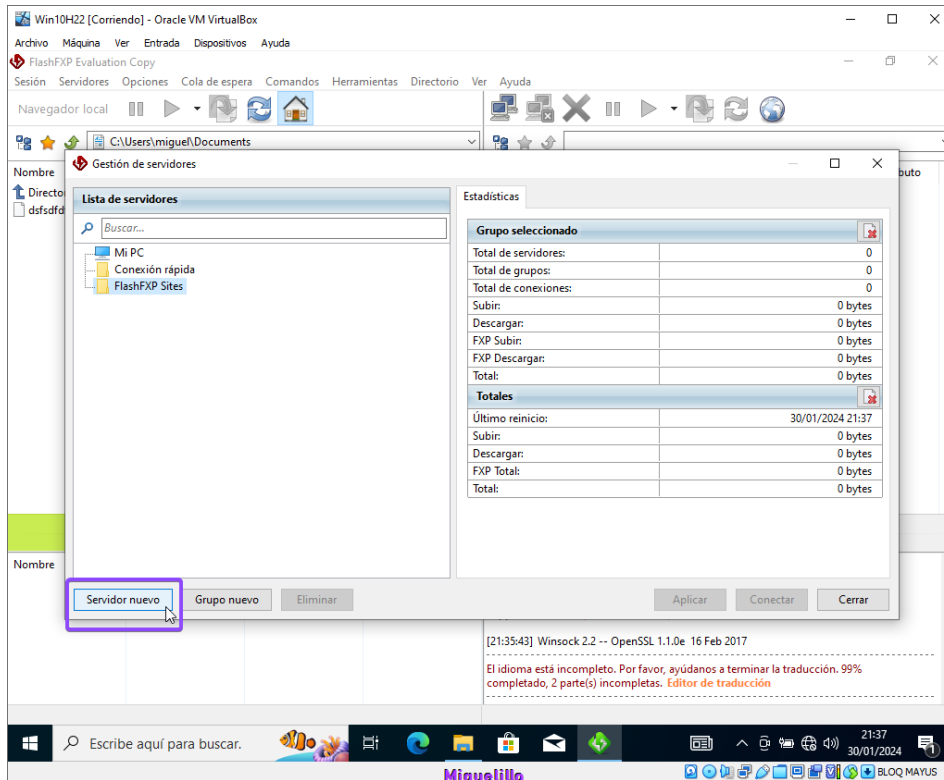
Configuramos la autenticación y los permisos



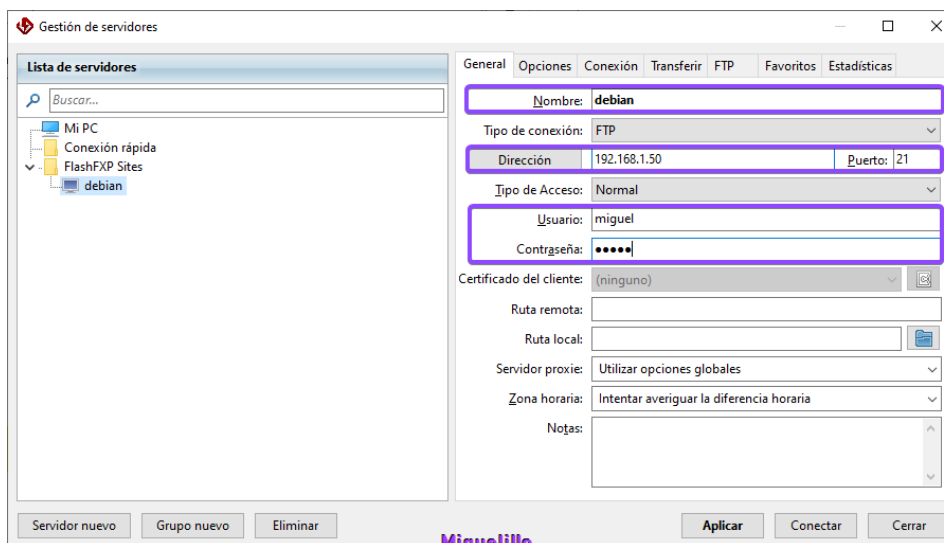
Vamos al cliente **Servidores** → **Gestión de servidores...**



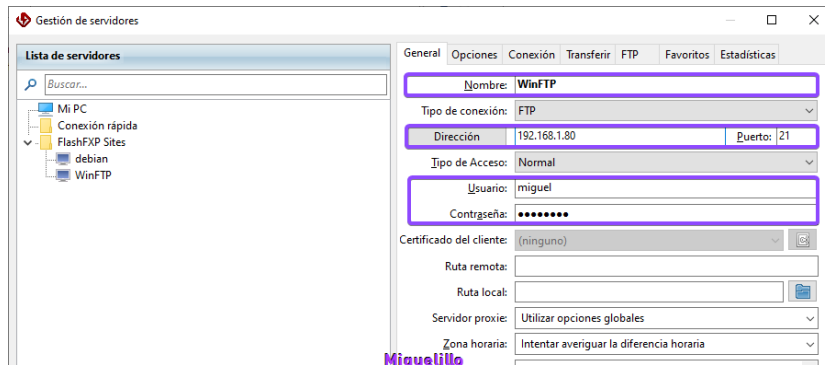
Vamos a **Servidor Nuevo**



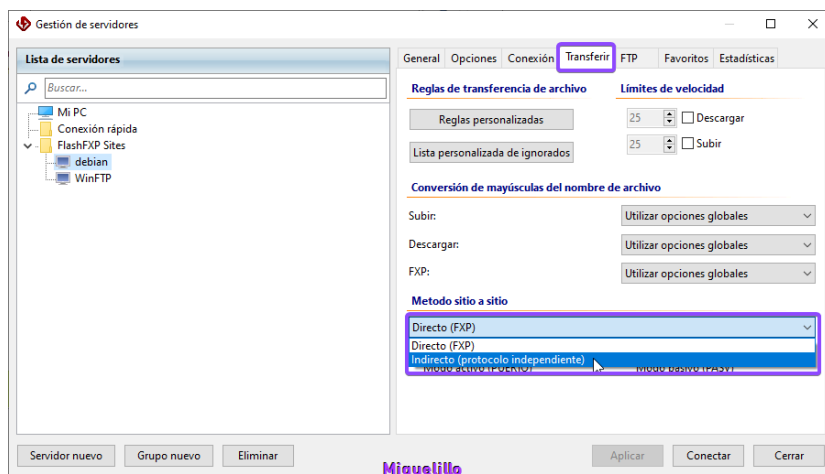
Ponemos el **Nombre, IP, Usuario y Contraseña**



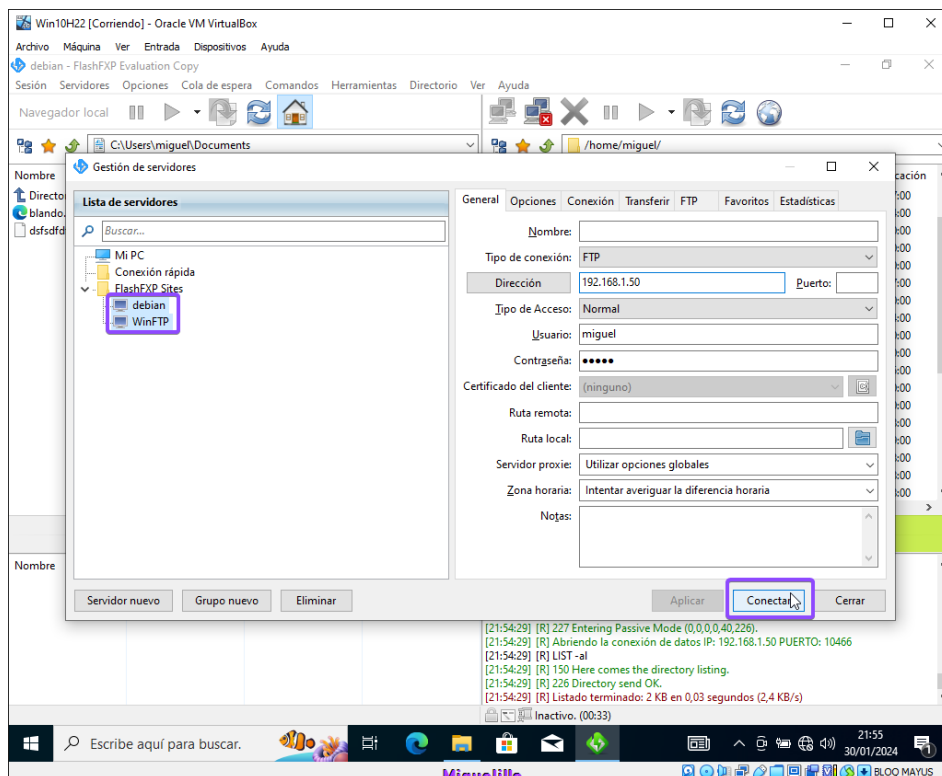
Añadimos le servidor de Windows



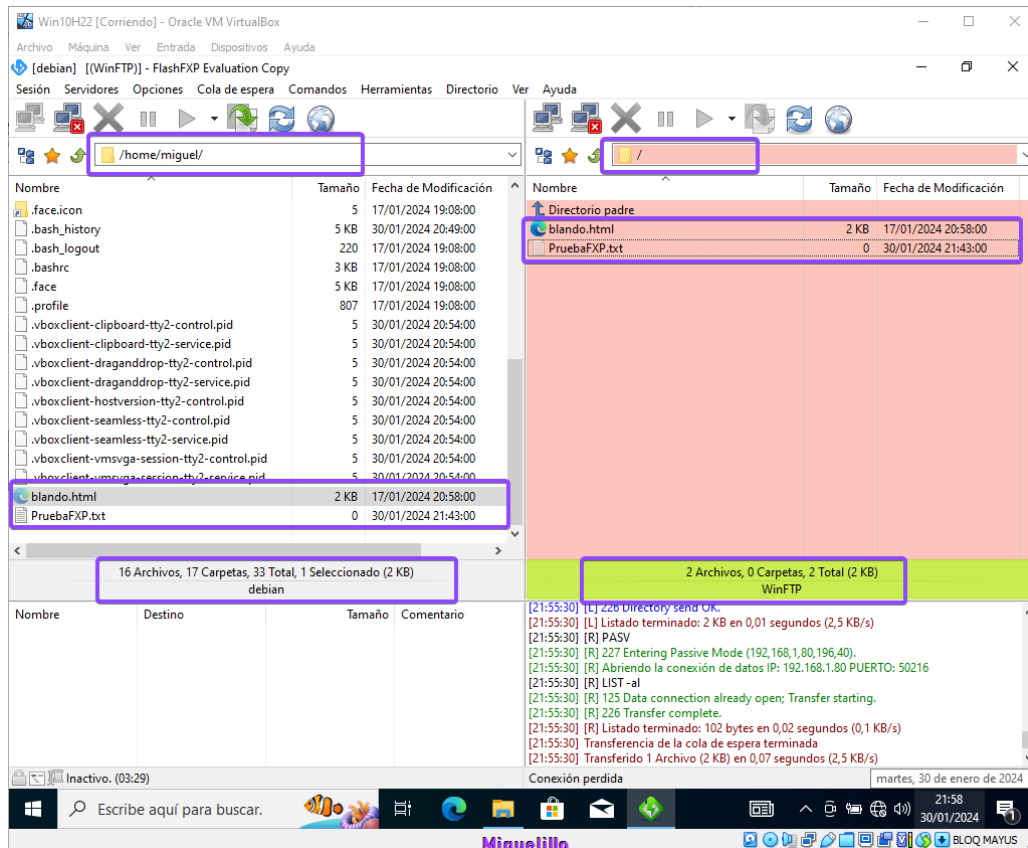
Ponemos los dos en indirecto



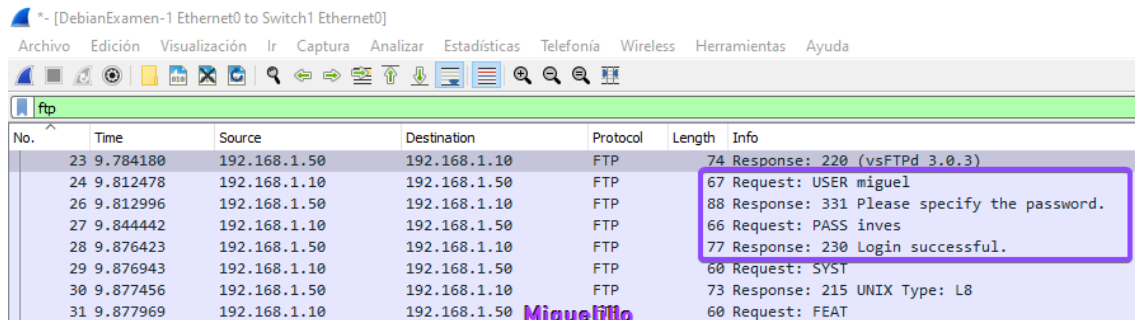
Seleccionamos los dos servidores y clicamos en conectar



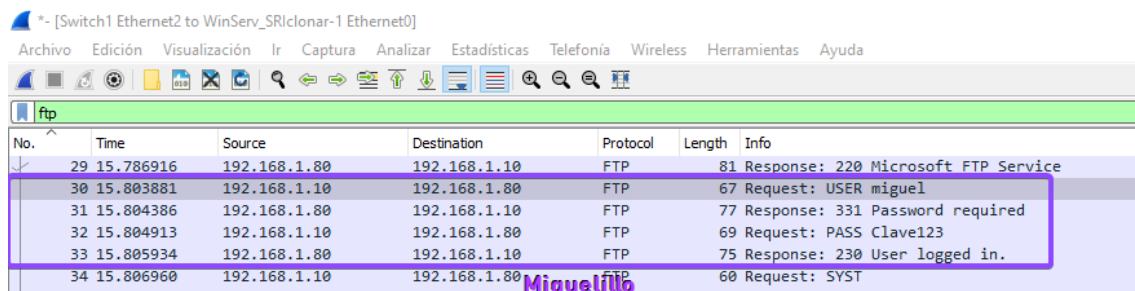
Probamos a mover 2 archivos



Login Debian



Login Windows



Cuando le pasamos el archivo de Windows a Linux

Wireshark interface showing a packet capture on the Ethernet0 interface. The packet list shows an FTP session. The packet details pane shows the File Transfer Protocol (FTP) structure, including the Request command: MDTM and the Request arg: blando.html.

No.	Time	Source	Destination	Protocol	Length	Info
88	31.227374	192.168.1.10	192.168.1.80	FTP	72	Request: MDTM blando.html
89	31.227888	192.168.1.80	192.168.1.10	FTP	74	Response: 213 20240117195816
90	31.228910	192.168.1.10	192.168.1.80	FTP	60	Request: PASV
91	31.229423	192.168.1.80	192.168.1.10	FTP	104	Response: 227 Entering Passive Mode (192,168,1,80,196,66).
95	31.231478	192.168.1.10	192.168.1.80	FTP	72	Request: RETR blando.html
96	31.231478	192.168.1.80	192.168.1.10	FTP	108	Response: 125 Data connection already open; Transfer starting.
101	31.232504	192.168.1.80	192.168.1.10	FTP	78	Response: 226 Transfer complete.

Packet 88 details:

- Frame 88: 72 bytes on wire (576 bits), 72 bytes captured (576 bits) on interface -, id 0
- Ethernet II, Src: PcsCompu_85:40:2d (08:00:27:85:40:2d), Dst: PcsCompu_25:45:34 (08:00:27:25:45:34)
- Internet Protocol Version 4, Src: 192.168.1.10, Dst: 192.168.1.80
- Transmission Control Protocol, Src Port: 62348, Dst Port: 21, Seq: 138, Ack: 514, Len: 18
- File Transfer Protocol (FTP)
 - MDTM blando.html\r\n
 - Request command: MDTM
 - Request arg: blando.html

[Current working directory: /]

Miguelillo

Cuando le pasamos el archivo Linux a Windows

Wireshark interface showing a packet capture on the Ethernet0 interface. The packet list shows an FTP session. The packet details pane shows the File Transfer Protocol (FTP) structure, including the Response code: Closing data connection (226) and the Response arg: Transfer complete.

No.	Time	Source	Destination	Protocol	Length	Info
112	31.251015	192.168.1.10	192.168.1.80	FTP	74	Request: MDTM PruebaFXP.txt
113	31.251524	192.168.1.80	192.168.1.10	FTP	74	Response: 213 20240130204341
114	31.252542	192.168.1.10	192.168.1.80	FTP	60	Request: PASV
115	31.253567	192.168.1.80	192.168.1.10	FTP	104	Response: 227 Entering Passive Mode (192,168,1,80,196,67).
119	31.256642	192.168.1.10	192.168.1.80	FTP	74	Request: RETR PruebaFXP.txt
120	31.257667	192.168.1.80	192.168.1.10	FTP	108	Response: 125 Data connection already open; Transfer starting.
122	31.257667	192.168.1.80	192.168.1.10	FTP	78	Response: 226 Transfer complete.

Packet 122 details:

- Frame 122: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface -, id 0
- Ethernet II, Src: PcsCompu_25:45:34 (08:00:27:25:45:34), Dst: PcsCompu_85:40:2d (08:00:27:85:40:2d)
- Internet Protocol Version 4, Src: 192.168.1.80, Dst: 192.168.1.10
- Transmission Control Protocol, Src Port: 21, Dst Port: 62348, Seq: 820, Ack: 286, Len: 24
- File Transfer Protocol (FTP)
 - 226 Transfer complete.\r\n
 - Response code: Closing data connection (226)
 - Response arg: Transfer complete.

[Current working directory: /]

Miguelillo