2

HERRAMIENTAS DE TRANSFERENCIA DE ARCHIVOS

El término de "transferencia de archivos" suele referirse a un intercambio de información que se produce dentro de una red informática. Si bien es cierto que la transferencia de archivos suele darse a través del modelo FTP (File Transfer Protocol), este intercambio de ficheros también puede ocurrir mediante otras diversas formas.

2.1 PROTOCOLO FTP

El protocolo de transferencia de archivos, aludido genéricamente por el acrónimo FTP, es un protocolo de red que permite la transferencia de archivos entre dos sistemas conectados en TCP (Transmission Control Protocol) bajo una arquitectura de tipo cliente-servidor.

El servicio FTP se ofrece gracias a la capa de aplicación del modelo de capas de red TCP/IP al usuario, que usualmente usan los puertos de red 20 o 21. Sin embargo, no hay que olvidar que, aunque nos puede ofrecer la máxima velocidad durante las conexiones, no suele proporcionar la máxima seguridad ya que todo el intercambio de información se produce sin cifrar, lo que hace más fácil que un usuario malintencionado pueda capturar el tráfico, acceder al servidor o, incluso, conseguir los archivos transferidos.

2.1.1 Diferencia entre Cliente y Servidor FTP

Aunque se verán un poco más en detalle a continuación, podríamos afirmar que, mientras que un cliente FTP no es más que un programa que sirve para conectarse con un servidor de su mismo tipo e intercambiar archivos, un servidor FTP es una aplicación o servicio que hace uso del protocolo FTP.

2.1.1.1 SERVIDOR FTP

Un servidor FTP es un programa o aplicación específica que se ejecuta en un equipo servidor y que suele estar conectado a Internet, además de ofrecer conexiones LAN o MAN, entre otras. Su objetivo no es otro que el de permitir el intercambio de datos entre diferentes sistemas, servidores, equipos u ordenadores.

Por lo general, los servidores FTP no suelen encontrarse en ordenadores de sobremesa ni portátiles, sino que se encuentran en equipos especiales dedicados a ello. Por ello, el usuario de a pie normalmente utilizará el FTP para conectarse remotamente a un servidor con el objetivo de subir o intercambiar información.

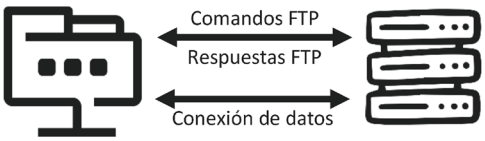
Su uso o aplicación más común suele ser para el alojamiento de páginas web, aunque también puede utilizarse como fuente de respaldo o backup.

2.1.1.2 CLIENTE FTP

Un cliente FTP es un programa o aplicación específica que se ejecuta en un equipo remoto del usuario, véase por ejemplo un ordenador de sobremesa o portátil, y que tiene por objetivo conectarse con un servidor FTP para subir o descargar archivos.

Por lo general, para poder usar un cliente FTP, se necesita conocer el nombre del servidor, el sistema de cifrado si lo tiene (que habitualmente será así), el modo de acceso (que podrá ser incluso anónimo) y el usuario y contraseña. No obstante, habrá otros parámetros que podremos necesitar como el puerto, si usa proxi o no, los directorios local y remoto o el modo de transferencia, que podrá ser activo o pasivo.

Su uso o aplicación más común suele ser para consultar, descargar o actualizar las diferentes páginas de un sitio web, aunque también puede utilizarse como herramienta para respaldar información en un servidor FTP.



Iconos extraídos de https://www.iconfinder.com/

Cabe destacar que algunos clientes FTP pueden venir integrados con el sistema operativo, pero suelen ser de tipo o modo comando, es decir, sin interfaz gráfica. Sin embargo, como veremos, los clientes FTP más frecuentemente utilizados nos proveen de múltiples opciones adicionales que se manipulan a través de una interfaz gráfica que, normalmente, es bastante intuitiva, aunque al principio pueda parecer que no.

Entre los clientes de modo texto o consola tenemos el comando **ftp**, que suele funcionar tanto para Windows como para Linux.

Entre los clientes con interfaz gráfica podemos encontrar múltiples opciones, no obstante, los más frecuentemente utilizados son Filezilla y WinSCP, que son los que se verán un poco en profundidad en este libro.

2.1.2 Filezilla

FileZilla es una plataforma libre de código abierto que nació con el pensamiento de facilitar el almacenamiento de archivos de forma remota a través de una red LAN y se lanzó a principios de 2001. Funciona a nivel de cliente / servidor y soporta los protocolos FTP, SFTP, SSL y TLS.

Originalmente se diseñó para Windows, pero a partir de la versión 3 se volvió multiplataforma y ahora está disponible, además, para GNU/Linux, FreeBSD y macOS.

Presenta tres versiones:

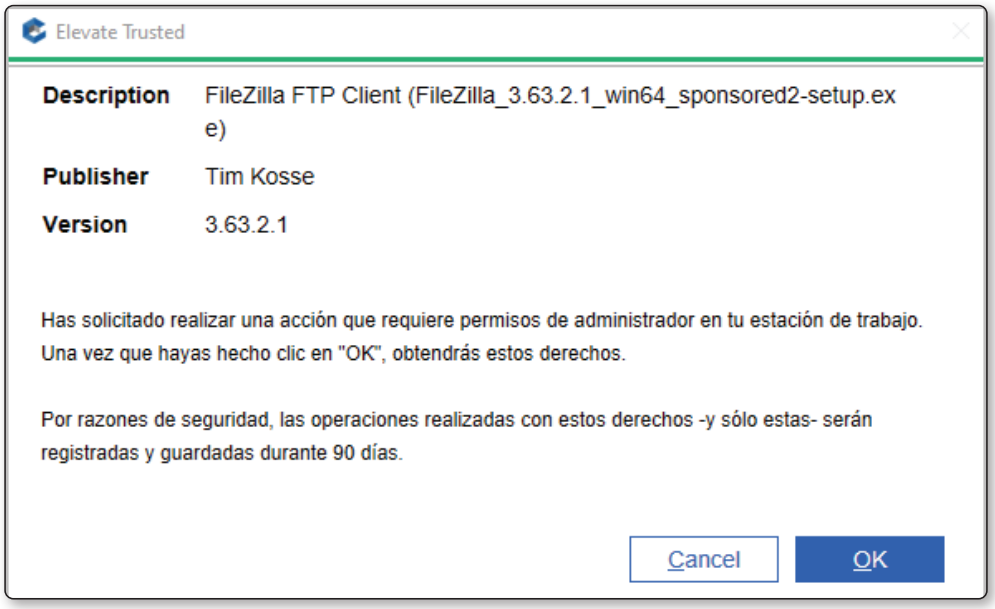
1. La versión que incorpora un instalador y que permite instalar o desinstalarlo fácilmente el software. Esta versión suele estar indicada para usuarios novatos o juniors.

2. La versión con el software comprimido sin instalador y que funciona una aplicación stand-alone. Esta versión suele estar indicada para aquellos usuarios a los que no les gustan los instaladores o que tienen necesidades especiales.

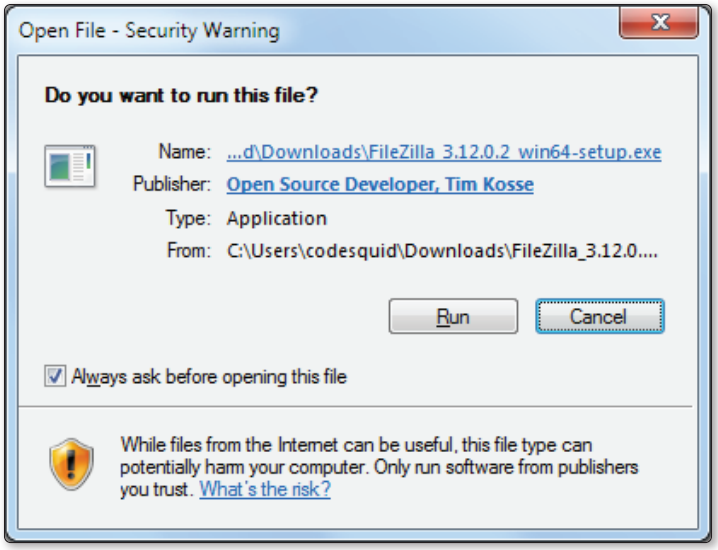
3. La versión con el código fuente que permite ver, modificar y/o compilar el código fuente de FileZilla. Esta versión suele estar indicada para usuarios avanzados que desean conocer o ayudar a mejorar la plataforma.

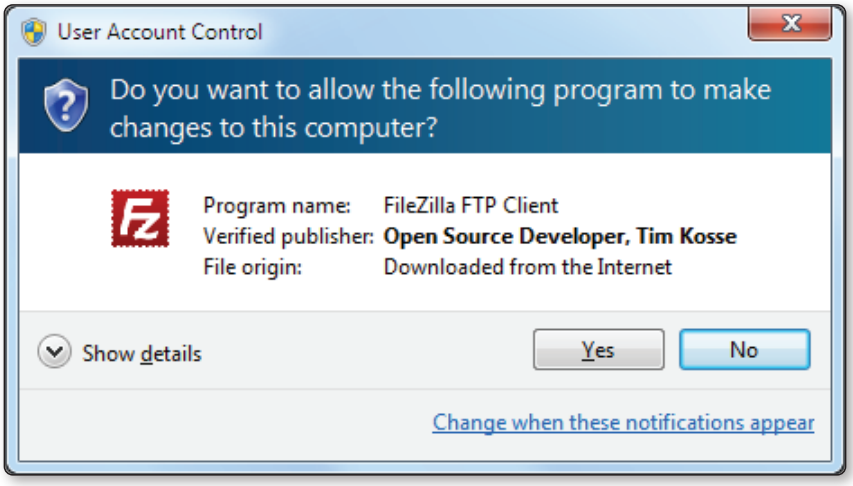
2.1.2.1 INSTALACIÓN EN WINDOWS

La versión de Windows que lleva el instalador normalmente nos solicitará primero derechos de acceso mediante algún mensaje de confirmación:



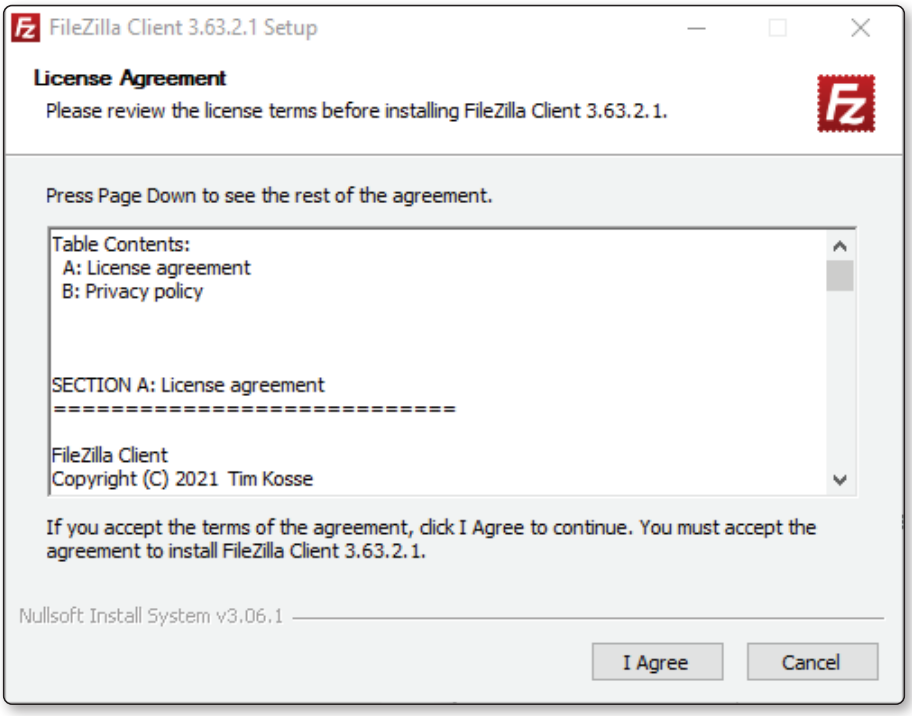
Mensaje de confirmación en Windows 10





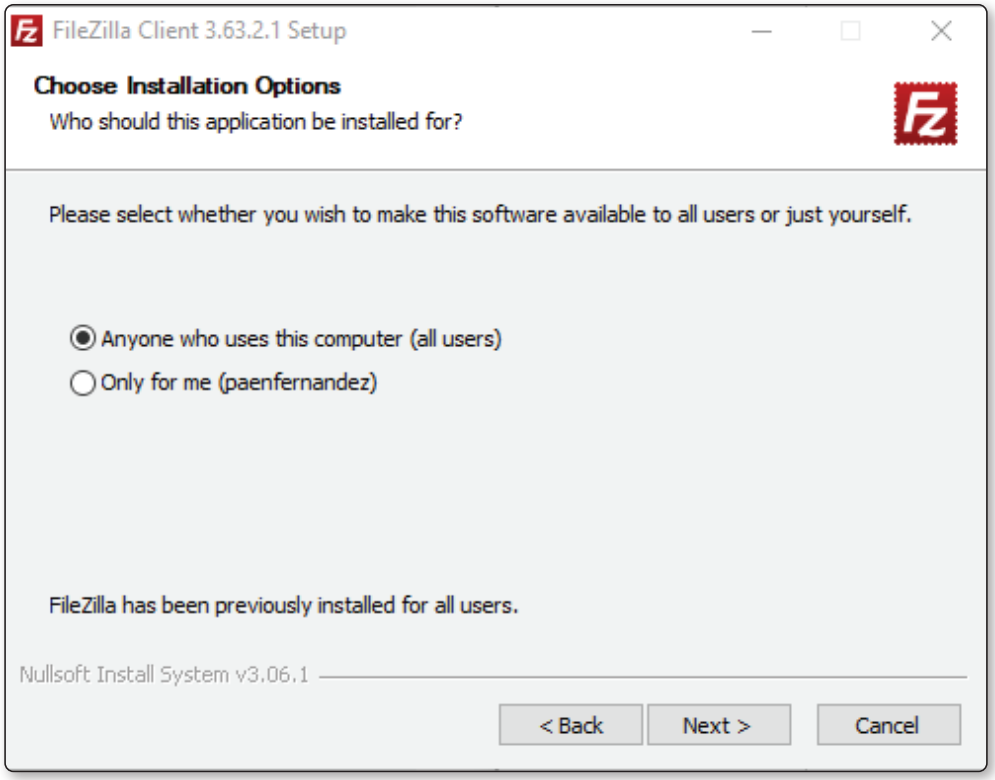
Mensajes de confirmación en Windows 8 extraídas de https://wiki.filezilla-project.org/Client_Installation

A continuación, se nos debería mostrar una pantalla con los términos del acuerdo de licencia.

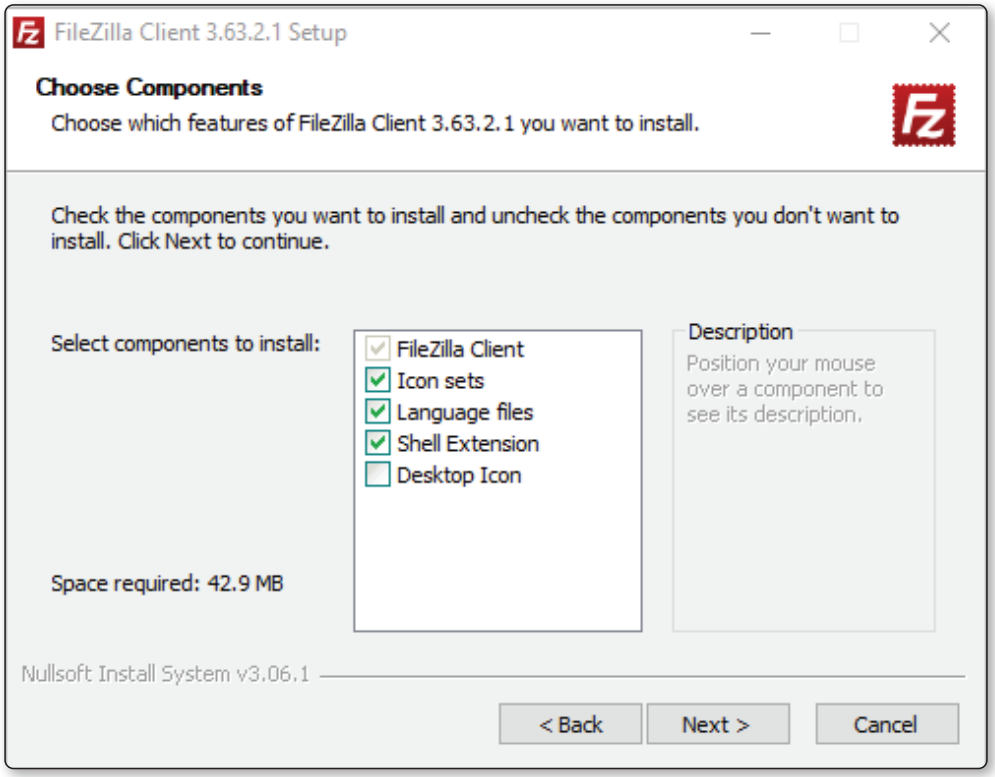


Una vez hayamos leído los términos de acuerdo de licencia y hayamos pulsado en el botón de "**estoy de acuerdo**" o "**continuar**", se nos debería mostrar una pantalla para seleccionar si la instalación va a ser sólo para nosotros o para todos los usuarios. No obstante, es posible que nos aparezca alguna pantalla anterior para ofrecernos algún software patrocinado por Filezilla.

A modo de apunte, cabe destacar que, la opción para todos los usuarios podrá afectar a la ubicación de los iconos del menú Inicio y requerir privilegios administrativos, sin embargo, permitirá que los usuarios tengan su propia configuración.



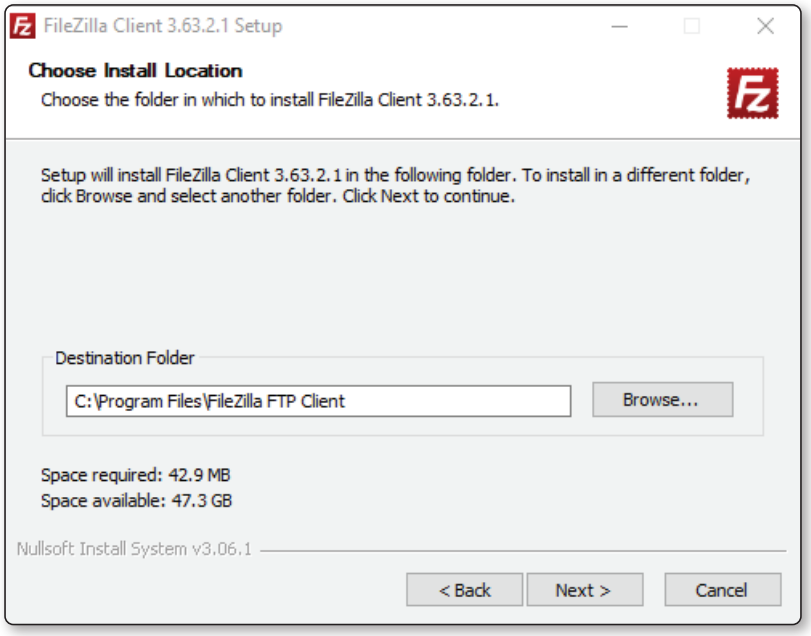
A continuación, deberemos elegir qué partes del cliente de FileZilla se desean instalar. Esto es, si además de instalar el cliente de Filezilla queremos instalar el set de iconos, los archivos de idioma, la extensión de Shell y el acceso directo en el escritorio. Si deseamos ver la descripción de uno o varios de los elementos que componen esta lista podremos hacerlo colocando el puntero del ratón sobre la opción deseada.



NOTA

En lo referente a la extensión de shell es importante saber que la instalación o eliminación requerirá privilegios de administrador y que, aunque FileZilla para Windows solo está disponible como compilación de 32 bits, contiene una extensión de shell tanto para el shell de 32 bits como para el de 64 bits.

Justo después de seleccionar los componentes se nos mostrará una pantalla para poder elegir dónde instalar el software. Aquí, lo recomendable es que, si no se tienen necesidades especiales, se haga clic en "**Siguiente**".



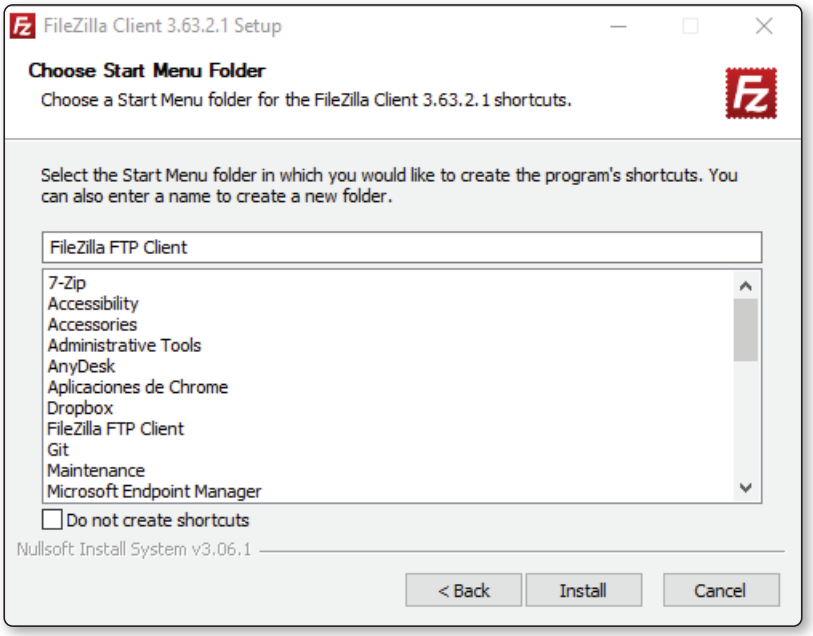
En general, la ruta predeterminada para instalar será el directorio "**Archivos de programa**" o "**Archivos de programa (x86)**".

ⓘ IMPORTANTE

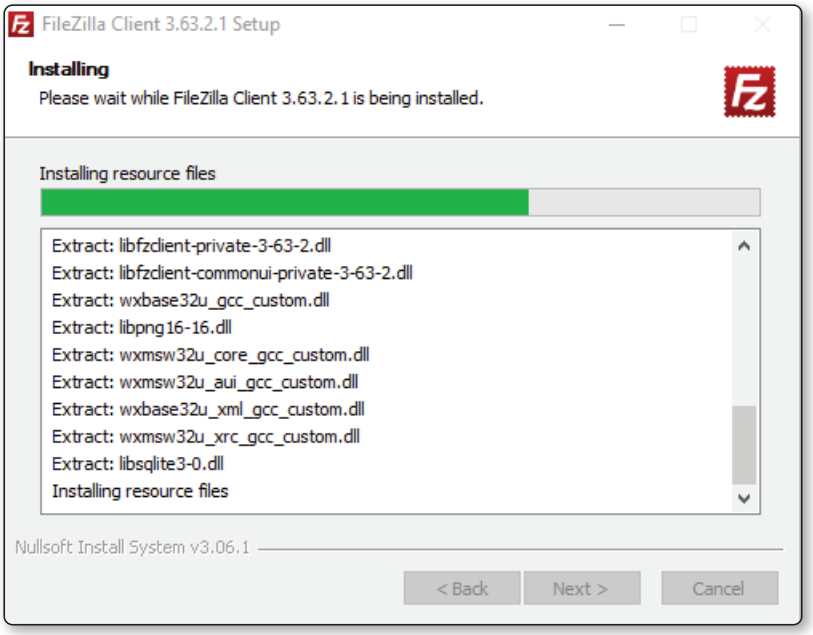
Si se va a actualizar desde una **versión 2**, es importante saber que **nunca debe instalarse en el mismo directorio** puesto que, al no ser una actualización, la mezcla de versiones podría generar un conflicto entre versiones.

Acto seguido, se nos preguntará cómo deseamos nombrar la carpeta con los accesos directos a FileZilla dentro del menú de inicio.

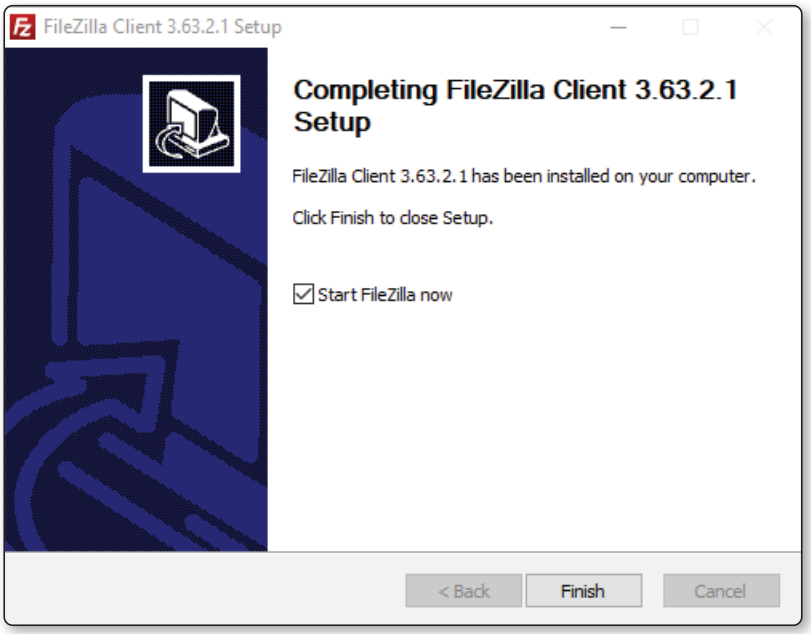
El descriptor predeterminado está establecido a "**FileZilla FTP Client**" y, en principio, este debería ser una buena elección para la mayoría de las personas, sin embargo, es posible cambiarlo al nombre o descriptor que se desee. Aun así, si no se desean accesos directos se puede optar por no instalar estos accesos directos pulsando en la opción de "**No crear accesos directos**".



Finalmente, tras pulsar el botón de "**Instalar**", se nos mostrará una pantalla con el progreso de la instalación la cual, si no se producen errores, debería completarse en unos pocos segundos.



Tras lo cual, deberíamos pulsar en "**Siguiente**" y aparecernos una pantalla de finalización como la mostrada a continuación:



Cuando haya terminado, ya podremos hacer clic en "**Finalizar**" y, si está seleccionada la casilla de verificación de "**Comenzar Filezilla ahora**", se nos abrirá la pantalla de inicio de Filezilla.

Bien, ya sabemos cómo instalar el cliente de FTP de Filezilla, pero, y si no queremos instalarlo o deseamos una versión en stand-alone ¿cómo lo hacemos? Pues con la versión zip.

La puesta en marcha de Filezilla con la versión zip es bastante sencilla siempre que se sepa cómo trabajar con archivos zip y se tengan derechos de administración para los archivos y directorios en Windows. Básicamente debemos extraer el archivo descargado en una carpeta de nuestro disco, por ejemplo, "**C:\Program Files\FileZilla FTP Client**" y, a continuación, ejecutar el archivo "**filezilla.exe**".

Al igual que se puede hacer en la versión con instalador, puede ser una buena idea crear un acceso directo a este archivo en nuestro escritorio, menú de inicio o inicio rápido para poder acceder a él fácilmente.

Por último, pero no menos importante, si deseamos crear una instalación portátil de FileZilla en una memoria USB o cualquier otro tipo de almacenamiento portátil, deberemos extraer el contenido del archivo ZIP en la ubicación deseada y

crear un archivo llamado "**fzdefaults.xml**" en el mismo directorio donde se encuentre el ejecutable de FileZilla.

Si se tienen dudas de cómo crear este último archivo, podemos consultar el de ejemplo que se encuentra en el subdirectorio "**./docs**" y usarlo como plantilla. Dentro de él, encontraremos instrucciones necesarias para establecer la ubicación de configuración en una ruta relativa en su dispositivo portátil.

2.1.2.2 INSTALACIÓN EN MAC OS X

Antes de instalar FileZilla en un sistema Mac OS X, se debe comprobar que la versión que tenemos instalada es la 10.9 o superior. Una vez comprobado, sólo tendremos que descargar el archivo de instalación.

Aunque lo normal será que el paquete de FileZilla se extraiga automáticamente si lo hacemos a través de Safari, es posible que se tenga que extraer manualmente si estamos usando un navegador diferente. Z

Una vez hecho esto, para iniciar FileZilla, únicamente deberemos hacer doble clic en el paquete de aplicaciones extraído.

ⓘ IMPORTANTE

Cuando iniciemos FileZilla en el Finder, observaremos que la ventana de FileZilla está cubierta por el menú superior y nos resulte difícil mover la ventana, porque no podemos hacer clic en la parte superior cubierta de la ventana de FileZilla.

Si esto sucede, normalmente podremos solucionarlo cambiando la resolución de la pantalla a otra y, posteriormente, devolviéndola a nuestra resolución original.

2.1.2.3 INSTALACIÓN EN GNU/LINUX Y OTROS SISTEMAS UNIX

Para la instalación de Filezilla en GNU/Linux y Unix es recomendable utilizar el administrador de paquetes de la distribución que tengamos instalada. No obstante, si estamos utilizando GNU/Linux, también podemos intentar usar los binarios precompilados.

Después de extraer los archivos a la ubicación deseada, podremos iniciar el software llamando al archivo ejecutable "**filezilla**" en el subdirectorio "**bin**".

Como nota final diremos que, debido a las diferencias en las distribuciones, es posible que los archivos binarios proporcionados no funcionen en nuestro sistema. Si esto sucede, siempre podemos compilar FileZilla con el código fuente.

2.1.3 WinSCP

WinSCP es otra aplicación libre de código abierto que emplea una interfaz gráfica de manera conjunta con el protocolo SSH, aunque permite otros protocolos SCP (de Secure Copy Protocol) para la transferencia de archivos de forma segura. Su principal objetivo, al igual que pasa con Filezilla, es el de facilitar la transferencia segura de archivos entre dos equipos montados a modo de cliente/servidor FTP.

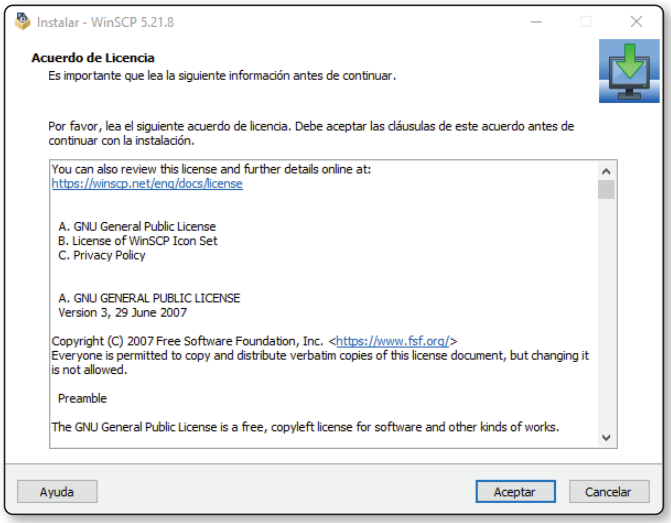
2.1.3.1 INSTALACIÓN

Una vez tengamos descargado el instalable de Windows lo que deberemos hacer es abrir el programa de instalación haciendo doble clic en su icono o nombre de archivo. En este momento nos debería ofrecer la elección del modo de instalación, esto es, "**Instalar para todos los usuarios (recomendado)**" o "**Instalar para mí solamente**".

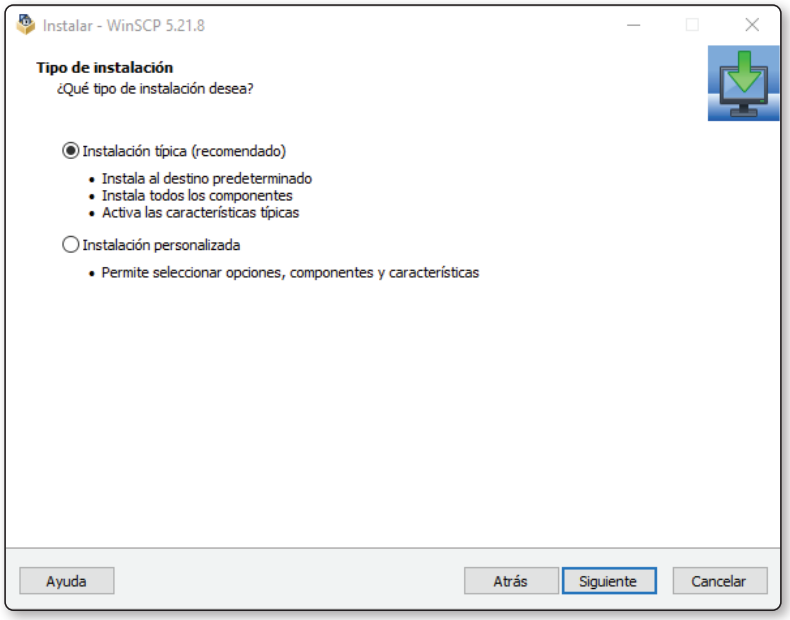
NOTA

Si el idioma de Windows que tenemos instalado no es compatible con el instalador de WinSCP, siempre podremos seleccionar un idioma alternativo tanto para el proceso de instalación como para el posterior uso de WinSCP.

A continuación, se nos debería solicitar, al igual que pasaba con la instalación de Filezilla, los derechos de acceso mediante algún mensaje de confirmación. Una vez permitido el paso, deberíamos poder ver una pantalla con el acuerdo de licencia, el cual describe todas las cláusulas que se deben tener en cuenta al acceder a instalar este software.



Tras haber pulsado en el botón de "**Aceptar**", se nos debería mostrar una pantalla donde se nos da a elegir el tipo de instalación. Esto es, si la queremos típica con el destino predeterminado, todos sus componentes y las características que ellos definen como típicas o, deseamos la instalación personalizada donde se nos irá preguntando qué deseamos para cada opción.

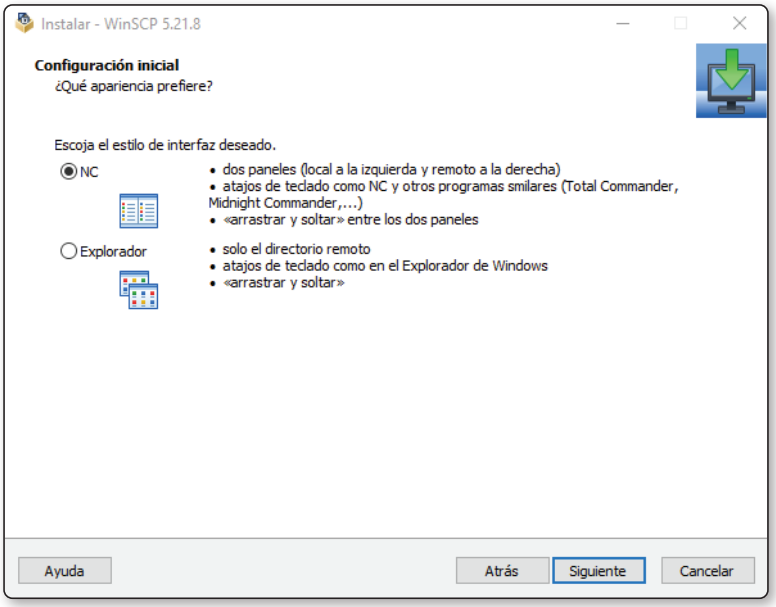


Después de haber seleccionado la opción deseada y, suponiendo que hayamos elegido la opción de "**Instalación típica**", si pulsamos en "**Siguiente**", se nos debería mostrar una pantalla con el estilo de la interfaz deseada.

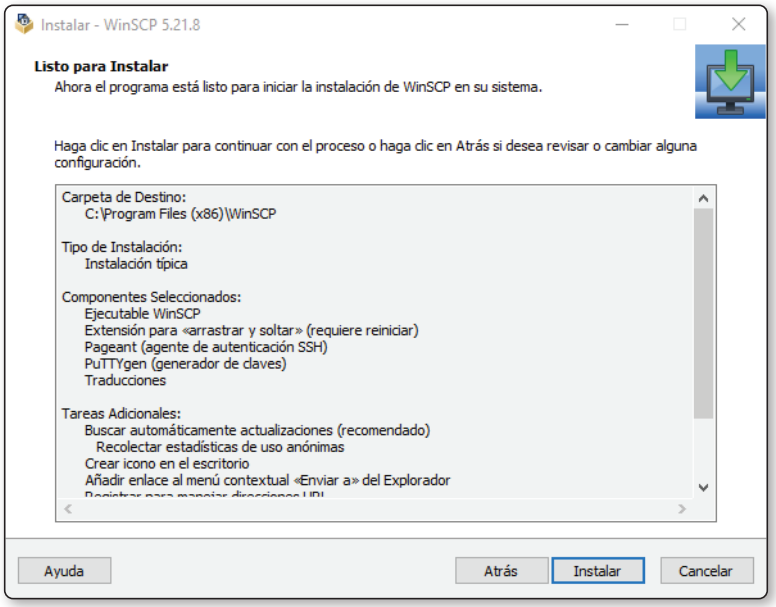
En esta ocasión, se nos debería ofrecer dos opciones. Una denominada "**NC**" que presenta el local a la izquierda y el panel remoto a la derecha, permite usar atajos de teclado como NC y otros programas similares como (Total Commander o Midnight Commander) y la posibilidad de arrastrar y soltar entre paneles.

La otra opción es la que denominan "**Explorador**" que sólo nos mostrará el directorio remoto, nos permitirá utilizar los atajos de teclado como en el Explorador de Windows e, igualmente, la posibilidad de arrastrar y soltar, aunque, eso sí, de otra manera.

En general, la opción más utilizada es la de "**NC**" debido, fundamentalmente, a que es algo más vistosa y fácil de entender.



Una vez seleccionada la interfaz deseada, se nos mostrará la pantalla de "**Listo para instalar**" en donde podremos revisar todas las opciones de instalación que hemos seleccionado durante el proceso y, si estamos de acuerdo, bastará con hacer clic en el botón de "**Instalar**".



NOTA

Durante el proceso de instalación es posible que nos soliciten reiniciar la computadora. Si esto sucede y elegimos no hacerlo, es posible que algunas funciones de WinSCP no estén disponibles hasta que el reinicio se vuelva efectivo.

2.1.4 Modos de conexión para clientes FTP

Cuando trabajamos con protocolos FTP se pueden establecer varios modos de conexión diferentes.

2.1.4.1 MODO ACTIVO

El modo de conexión activa, también conocida como modo de conexión estándar o sólo modo activo, es aquel que se produce cuando el cliente abre un puerto, se pone a escuchar y es el servidor quien se conecta activamente con el cliente.

Cuando un cliente FTP solicita una conexión con un servidor dado, lo que se está haciendo es establecer una comunicación a través de un puerto aleatorio, que normalmente es mayor al 1024, hacia el puerto 21 del servidor FTP y, por esa misma conexión, se le indica al servidor FTP cuál será el puerto de nuestro cliente FTP por el que se desea recibir los datos.

En general podemos afirmar que cuando usamos este modo, el servidor utilizará dos conexiones. Una por el puerto 20, que se usará para transferir los datos y otra por el puerto 21, que se usará para la transmisión de los distintos comandos.

La secuencia de conexión del modo activo será o debería ser como la siguiente descripción de pasos:

1. El cliente FTP emprende una conexión desde un puerto de control aleatorio con valor superior a 1024 y con destino al puerto 21 del servidor.
2. El servidor responde desde el puerto de control.
3. El servidor inicia la conexión del canal de datos. En general este canal será el puerto 20 para el servidor y puerto siguiente al indicado por el cliente en el paso previo.
4. El cliente responde desde el puerto de datos estableciendo así la conexión solicitada.

Por último, cabe destacar que, aunque este modo es bastante utilizado, presenta un inconveniente y es que, si no se ha definido directiva alguna en los firewalls del lado del cliente, la conexión puede llegar a bloquearse. Esto se debe precisamente a que este modo requiere dos conexiones, una para establecer el canal de control y otra para el envío de datos.

2.1.4.2 MODO PASIVO

El modo de conexión pasiva, también conocido como modo pasivo, surge por la cantidad de problemas que, históricamente, creó el modo activo y es aquel que se produce cuando el servidor abre un puerto, se pone a escuchar de manera pasiva y es el cliente quien se conecta con el servidor.

En este modo de conexión, el cliente FTP solicita la conexión con el servidor FTP a través del envío de un comando "**pasv**" y, cuando esto sucede, se establece una comunicación mediante un canal de control que normalmente se corresponde con un valor de puerto aleatorio mayor al 1024 y el puerto 21 del servidor.

Por tanto, para establecer la comunicación, primero se solicitará al servidor que abra un puerto aleatorio mayor que 1024 desde el cliente FTP. Seguidamente, y tras haber recibido la respuesta del servidor, el cliente FTP establecerá una conexión de datos a través del puerto indicado en la petición anterior.

La secuencia de conexión del modo pasivo será o debería ser como la siguiente descripción de pasos:

1. El cliente FTP emprende una conexión desde un puerto de control aleatorio con valor superior a 1024 y con destino al puerto 21 del servidor.
2. El servidor responde desde el puerto de control.
3. El cliente inicia la conexión del canal de datos. En general este canal será el puerto un puerto aleatorio alto con valor superior a 1024 para el servidor y puerto siguiente al indicado por el cliente en el paso previo.
4. El servidor responde desde el puerto de datos estableciendo así la conexión solicitada.

Si comparamos las secuencias del modo activo y el modo pasivo, lo que podremos observar es que la diferencia estriba en quién toma el control, siendo el servidor quien toma el control en el modo activo y el cliente quien toma el control en el modo pasivo.

2.1.5 Modos de acceso

De forma global podemos decir que los modos de acceso en un cliente FTP son anónimos, con usuario y contraseña y de invitado.

2.1.5.1 ACCESO ANÓNIMO

En el modo de acceso anónimo, el usuario puede acceder al contenido del servidor sin una cuenta de usuario asociada. Sin embargo, no se suele usar ya que, para que funcione el servidor debe aceptar conexiones anónimas sin nombre de usuario ni contraseña, lo que puede traer graves problemas de seguridad.

Si el servidor tiene el servicio FTP anónimo instalado y activado, podremos acceder a él con tan solo insertar el nombre de usuario "anonymous". Esto significa que, muy probablemente, no se nos pedirá contraseña alguna, aunque sí que puede que se nos solicite una dirección de correo electrónico o algún otro dato único en el campo de la contraseña a modo de control.

Tras introducir esta palabra clave, lo habitual será tener acceso directo al servidor, eso sí, con menos privilegios de los que tendría un usuario registrado con nombre de usuario y contraseña. Con este tipo de acceso, lo normal será que únicamente podamos acceder a archivos y directorios públicos sin demasiada relevancia ni información privilegiada.

2.1.5.2 ACCESO DE USUARIO

En el modo de acceso de usuario, el usuario puede acceder al contenido del servidor sólo si dispone de un nombre identificativo único y una contraseña.

En general, este tipo de acceso lo tendrán aquellos usuarios que necesiten subir o descargar tanto archivos como directorios y, por tanto, aquellos usuarios que tengan permisos para crear, modificar y/o borrar tanto los archivos, como los directorios.

2.1.5.3 ACCESO DE INVITADO

El modo de acceso de invitado se podría definir como una variación del modo de acceso de usuario ya que, el usuario requiere de un nombre identificativo único y una contraseña. No obstante, únicamente podrá trabajar en el directorio asignado para ese usuario, por lo que no podrá acceder a otros directorios que estén fuera de éste.

2.1.5.4 MODOS DE ACCESO EN FILEZILLA

En Filezilla podemos encontrar las siguientes:

- **Acceso anónimo**: el acceso al servidor se realiza sin una cuenta de usuario asociada. No se suele usar ya que, para que funcione el servidor debe aceptar conexiones anónimas sin nombre de usuario ni contraseña, lo que puede traer graves problemas de seguridad.
- **Acceso normal**: el acceso al servidor se realiza a través de un nombre de usuario y contraseña que, habitualmente, suelen estar almacenados dentro del programa.
- **Preguntar la contraseña**: es igual que la opción anterior, con la diferencia de que no guarda la contraseña y la solicita cada vez que iniciemos una nueva sesión.
- **Interactivo**: el acceso al servidor se realiza tras solicitar la contraseña, pero no mantiene la sesión, por lo que se solicita en cada nueva solicitud de conexión al servidor.
- **Cuenta**: permite especificar una cuenta. Este modo de acceso no es nada frecuente.

2.2 PROTOCOLO FTPS

El protocolo SSH File Transfer Protocol, aludido genéricamente por el acrónimo SFTP o FTP/SSL, es un protocolo de red que permite la transferencia de archivos entre dos sistemas conectados de forma segura.

Entre sus características más notorias podríamos decir que el protocolo SFTP intenta ser más independiente de la plataforma que SCP (Secure Copy Protocol). Esto es porque mientras que, por ejemplo, SCP soporta expansión de comodines especificados por el cliente hasta el servidor, el diseño SFTP no evitando así este problema. Sin embargo, otra cosa que lo diferencia es que suele usar el puerto 22, en vez del puerto 21 como pasa con el protocolo FTP.

Como nota interesante diremos que en múltiples ocasiones los usuarios piensan que el protocolo SFTP se "obtiene" al ejecutar el protocolo FTP sobre SSH o que está vinculado de alguna manera con el Protocolo Simple de Transferencia de Archivos. Esto es un error pues SFTP es un protocolo diferente a FTP que está diseñado por el grupo de trabajo IETF SECSH y tampoco tiene nada que ver con el Protocolo Simple de Transferencia de Archivos, que es un software de línea de comandos que el cliente ejecuta parte de este protocolo.

2.2.1 Métodos de conexión

Existen dos métodos para realizar una conexión SSL/TLS a través de SFTP.

2.2.1.1 SFTP IMPLÍCITO

El SFTP Implícito es un método que establece una conexión cifrada entre el cliente y el servidor previamente a la transferencia de datos. Esta conexión cifrada suele realizarse con TSL o SSL y a través de los puertos 990 (SFTP) y 998 (SFTP Data), sin posibilidad alguna de negociación sobre el puerto ni el tipo de conexión.

2.2.1.2 SFTP EXPLÍCITO

El SFTP Explícito, que es el recomendado por el RFC 4217, es un método que establece una conexión cifrada entre el cliente y el servidor por el puerto 21 y, a continuación, negocia de forma explícita si se debe cambiar a un modo seguro utilizando TSL o SSL o no para transferir la información deseada.

2.2.2 Por qué y cómo usar SFTP

El uso de SFTP es algo casi necesario porque aporta más seguridad y confiabilidad ya que añade un cifrado a las transmisiones y los mensajes se transmiten a través de un canal seguro.

Dado que el Filezilla es uno de los clientes FTP más populares y que está disponible para los principales sistemas operativos, en este curso, lo usaremos como herramienta de apoyo y aprendizaje durante todo el proceso de publicación de páginas web.

Dicho esto, para establecer una comunicación SFTP con Filezilla deberíamos seguir los siguientes pasos:

1. Abrir el panel de conexiones pulsando **Archivo** / **Gestor de sitios**.
2. En la pantalla que nos aparece a continuación, pulsamos en **Nuevo sitio** para crear una nueva conexión.
3. En la misma pantalla se nos creará una nueva entrada en "**Mis sitios**" que pondrá "**Nuevo sitio**" y varias opciones en blanco a la derecha. Justo aquí, a la derecha, hay una opción llamada protocolo en la que deberemos seleccionar **SFTP – SSH File Transfer Protocol**.

4. A continuación, deberemos introducir el nombre o datos de host en donde indica "**Servidor**", el modo de acceso que, en general, será "**Normal**" y el nombre de usuario y contraseña que nos hayan proporcionado.

5. Finalmente, pulsaremos en el botón de **Conectar**.

2.3 OPENSSL

OpenSSL es un proyecto de software libre basado en SSLeay, desarrollado por Eric Young y Tim Hudson que nos provee de un robusto paquete de herramientas relacionadas con la seguridad que suministran funciones criptográficas a otros paquetes como OpenSSH o para el acceso seguro a sitios web con HTTPS.

Además, permite crear certificados digitales que pueden aplicarse a servidores como, por ejemplo, Apache y ayudan a la implementación del SSL, así como otros protocolos relacionados con la seguridad, como el TLS.

El objetivo de OpenSSL no es otro que el de mejorar la ciberseguridad aportando la posibilidad de realizar cifrados simétricos y asimétricos, calculando funciones hash, estableciendo firmas RSA, DSA, AES, creando certificados digitales X509, CSR o CRL y permitiendo hacer pruebas de clientes y servidores SSL/TLS, entre otras cosas.

A continuación, se explica un poco todo lo que se debe conocer de OpenSSL y sus principales funciones.

2.3.1 Certificados SSL y TLS

Tanto un certificado SSL (Secure Sockets Layer), como un certificado TLS (Transport Layer Security), son unos documentos digitales que permiten autenticar o ratificar la identidad de un sitio web y habilitan una conexión segura. La única diferencia entre ambos es que TLS resulta ser una versión más actualizada y segura que SSL.

Ambos certificados se suelen conseguir por medio de lo que se denomina Autoridades Certificadoras, que pueden ser de pago o no, en función del servicio que proporcionen y/o el nivel de validación que puede proporcionar a sus usuarios sobre el dominio en el que se encuentran y la empresa u organización que está detrás de él.

Las Autoridades Certificadoras como Digicert o ZeroSSL pueden ser buenas opciones si lo que se desea es un certificado SSL de pago, sin embargo, si se desea un certificado SSL gratuito la mejor opción suele ser **Let's Encrypt**.

2.3.2 Cifrados asimétricos

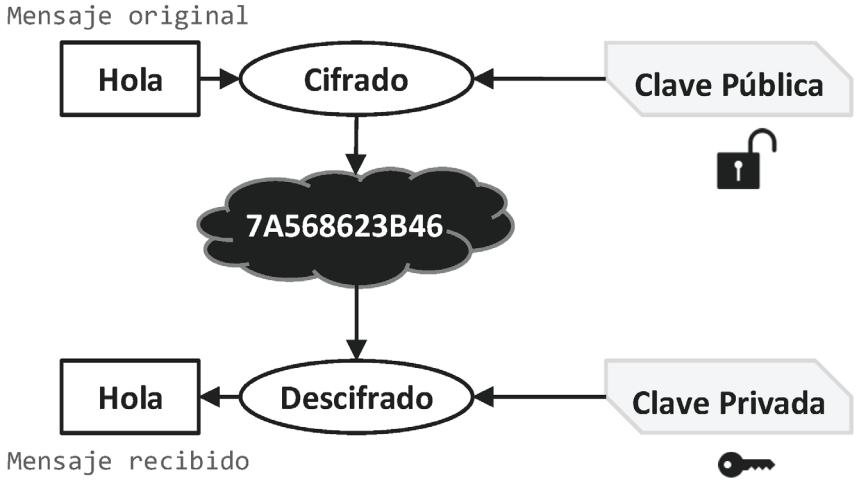
El cifrado asimétrico se caracteriza porque usa lo que se denomina una infraestructura de clave pública o PKI, un tipo de cifrado que se usa para realizar comunicaciones entre dos o más sistemas y/o dispositivos.

Este tipo de cifrado se usa en aplicaciones tan conocidas como WhatsApp, Telegram o Skype y se basa en la premisa de que un contenido o mensaje debe utilizar dos claves diferentes para cifrar o descifrar, sin embargo, aunque sean diferentes una se basa en la otra.

A estas claves se las denomina clave pública y clave privada. La clave pública es la "llave" encargada de realizar la encriptación del mensaje o contenido, la cual posee un valor único para cada iteración o construcción, y es conocida por todos los sistemas o dispositivos con los que se desea interactuar o entablar una comunicación.

Por otro lado, la clave privada es la "llave" encargada para realizar la desencriptación y únicamente conocida por el propietario, lo que implica que no se transmita ni dé a conocer a nadie en ninguna circunstancia.

A continuación, se muestra un esquema del flujo de un cifrado asimétrico:



Para entender mejor este tipo de cifrados, los profesionales suelen recurrir al símil del candado, un supuesto en el que hay un cofre o baúl y un candado, ambos abiertos. En este supuesto, un usuario que actúa como emisor recoge un papel y escribe en él un mensaje que, a continuación, guarda dentro del cofre o baúl y cierra de forma segura poniendo y cerrando el candado. Para que el usuario receptor pueda

leer el mensaje, el usuario emisor le proporciona la llave del candado garantizando, así, que únicamente él pueda acceder a dicho mensaje.

De este supuesto podemos deducir por tanto que el candado es la clave pública, que la llave es la clave privada y que, si algo viene firmado con la clave privada, es que es de quien firmó con la clave pública, lo que significa que podemos afirmar el origen de dicho mensaje o contenido.

Los cifrados asimétricos más utilizados y, posiblemente, conocidos son RSA y DSA, pero también se aplica en certificados digitales que usan el estándar X.509.

2.3.2.1 RSA

El término RSA es un acrónimo formado a partir de los apellidos de sus autores (Rivest, Shamir y Adleman) y que hace referencia a un algoritmo de cifrado de infraestructura de clave pública que resulta particularmente útil para realizar firmas digitales, para transacciones relacionadas con el comercio electrónico o para encriptar y enviar claves simétricas que se usarán posteriormente en comunicaciones cifradas en sistemas mixtos.

Fue descrito por primera vez en 1977 y su seguridad se basa en la problemática que existe al factorizar números enteros grandes primos, es decir, a la dificultad que se produce para expresar un número como el producto de otros menores que lo pueden dividir de forma entera y que, además, son únicamente divisibles por 1 y ellos mismos.

El funcionamiento del algoritmo RSA se basa en crear las claves pública y privada y, posteriormente, ejecutar unas fórmulas para cifrar y descifrar el mensaje o contenido. Sin embargo, para que este sistema se vuelva totalmente seguro, deberemos utilizar longitudes de no menos 1024 bits, siendo recomendable usar una longitud de 2048 bits.

2.3.2.2 DSA

DSA (Digital Signature Algorithm o Algoritmo de Firma digital) es un estándar del Gobierno Federal de los Estados Unidos de América o FIPS que se usa en firmas digitales.

Es un algoritmo puramente asimétrico que tiene como objetivo firmar o establecer la autenticidad de archivos y/o documentos y no suele ser apto para cifrar información. La principal razón de por qué no se utiliza con contenidos o mensajes es porque resulta muy costoso ya que requiere mucho más tiempo de cómputo que el algoritmo RSA.

2.3.2.3 X.509

X.509 es otro estándar de infraestructura de clave pública o PKI que se utiliza como estándar para crear certificados digitales y para la validación de rutas de certificación.

Se cuenta que, en su versión 3, el certificado X.509 se corresponde con un documento que está firmado de manera digital en base al estándar recogido en la Internet Engineering Task Force (IETF) y de acuerdo con la RFC 5280, el cual especifica una longitud de 64 bits para estos certificados y una serie de componentes como son la firma, el número de serie, el emisor, la validez y las extensiones, entre otros.

Para adquirir un certificado X.509 acudir a alguna de las entidades de certificación autorizada como son, por ejemplo, la Dirección General de la Policía, la Autoridad de Certificación ANF, la Agencia Notarial de Certificación ANCERT o el Servicio de Certificación de los Registradores SCR.

En lo referente a su utilización, se suelen usar de la navegación web forma conjunta con protocolos SSL y certificados https, para enviar correos electrónicos encriptados a través del protocolo S/MIME, para firmar códigos y/o documentos electrónicos de diversa índole o para identificar y autenticar de clientes, gobiernos o administraciones públicas.

2.3.3 Cifrados simétricos

El cifrado simétrico se caracteriza porque usa la misma clave para cifrar y descifrar un determinado contenido o mensaje. Aunque pueda parecer que es menos seguro que el cifrado asimétrico, no es así, únicamente tiene objetivos diferentes.

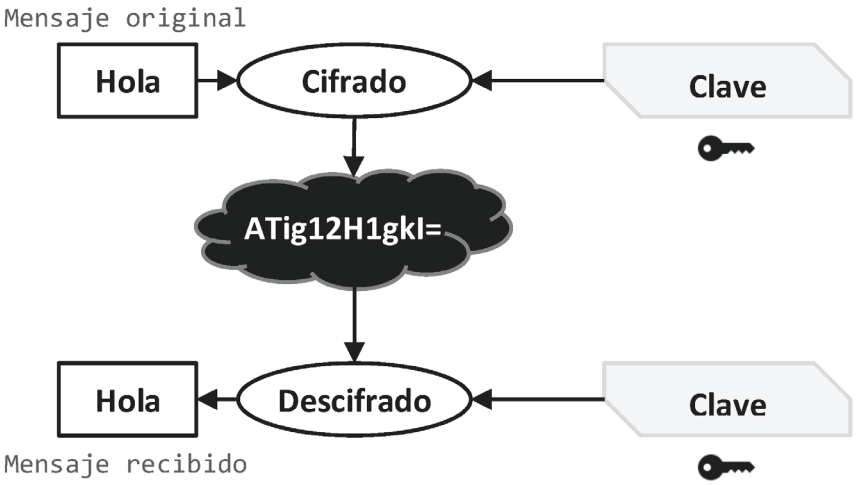
Este tipo de cifrado es el que usaba, por ejemplo, la máquina Enigma durante la Segunda Guerra Mundial y se suele utilizar cuando la cantidad de datos a transmitir es muy grande.

A diferencia del cifrado asimétrico, el cifrado simétrico sólo necesita una única clave o “llave”, lo que significa que, para que se pueda dar o utilizar, todas las partes o interlocutores deben ponerse de acuerdo de antemano sobre cuál será la clave para usar. Una vez de acuerdo, ya se puede empezar a cifrar y descifrar los mensajes o contenidos.

No obstante, también debemos destacar que, el cifrado simétrico presenta dos formas de construcción. En general se le suele denominar tipos de cifrado simétrico y son dos:

- **Cifrado de flujo**: se trata de combinar todos los caracteres o dígitos de un texto plano con una secuencia pseudoaleatoria de igual longitud usando una función matemática simple. El cifrado de flujo más conocido posiblemente sea RC4.
- **Cifrado de bloque**: se trata de dividir el mensaje en bloques de longitud fija y de forma que cada bloque se corresponda con otro diferente. Por ejemplo, si dispusiéramos de un tamaño de bloque de longitud 3 podríamos definir un bloque como "010" y su correspondiente como "110". El cifrado de bloque más conocido posiblemente sea AES.

A continuación, se muestra un esquema del flujo de un cifrado asimétrico:



2.3.3.1 RC4

El algoritmo RC4 o ARC4 es un sistema de cifrado de flujo que se usa en algunos de los protocolos más populares como es TLS/SSL, el cual permite proteger el tráfico de Internet, o WEP (Wired Equivalent Privacy), el cual permite añadir una mayor seguridad a las redes inalámbricas.

Hoy, se considera un algoritmo de criptografía bastante inseguro tanto a nivel de TLS/SSL, como a nivel WEP, sin embargo, existen algunos sistemas basados en RC4 que todavía son potencialmente seguros para un uso común. Eso sí, no se recomienda su uso o aplicación en proyectos de nueva generación.

2.3.3.2 AES

AES (Advanced Encryption Standard o Estándar de Encriptación Avanzada) es un esquema de cifrado por bloques que se convirtió en un estándar de cifrado el 26 de mayo de 2002 para y por el gobierno de los Estados Unidos, aunque se creó en Bélgica.

Podríamos afirmar que el algoritmo de encriptación AES es uno de los más populares cuando se habla de criptografía simétrica y suele ser conocido también como AES GCM, AES-256-GCM o Rijndael.

En cuanto a su funcionamiento, el algoritmo AES toma un bloque de texto plano y se le aplican rondas alternas de bloques de sustitución y permutación. Estos bloques están en 128, 192 o 256 bits, lo que determinará la fuerza del cifrado.

2.4 CONFIGURACIÓN Y CONEXIÓN CON SISTEMAS REMOTOS

2.4.1 Parámetros genéricos y específicos para diferentes servidores

Como ya se ha comentado con anterioridad, para realizar una conexión FTP será necesario especificar la dirección del servidor (que normalmente será URL, aunque puede ser una IP), el puerto (que habitualmente será el 21) y el nombre de usuario y contraseña.

2.4.1.1 CONFIGURACIÓN Y CONEXIÓN DESDE LÍNEA DE COMANDOS EN GNU/LINUX

El comando **ftp** puede utilizarse sin opciones ni parámetros, aunque lo más normal es usarlo, al menos, con la dirección IP o URL. Por tanto, la forma genérica del comando **ftp** podría definirse como:

```
# ftp [opciones] [usuario] [dirección[:puerto]]
```

En lo referente a las opciones existen varias, pero las más recurrentes son las siguientes:

Opción	Significado
-d	Permite habilitar el modo de depuración.
-i	Desactivar el modo interactivo durante la transferencia de varios archivos.
-n	Deshabilitar la posibilidad de realizar inicios de sesión automáticos en la conexión inicial.
-p	Utilizar el modo pasivo para las transferencias de datos, lo cual permite usar FTP a pesar de que exista un firewall que pueda impedirlo.
-v	Mostrar todas las respuestas del servidor remoto, sean del tipo que sean.

Para hacer la conexión podemos recurrir a tres posibles formas:

1. Sin parámetros. Esta forma nos mostrará el símbolo de sistema del servidor FTP, pero no estaremos conectados. Para conectar deberemos introducir el comando FTP **open**:

```
# ftp
ftp> open ftp.ejemplo.es
Conectado a ftp.ejemplo.es
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
220-Sólo acceso autenticado
220 ++++++++++
Usuario (192.168.10.10:(none)):
331 Contraseña requerida
Constraseña:
230-Sesión iniciada
ftp>
```

2. Con únicamente la URL o IP, con o sin puerto. Esta forma nos solicitará que introduzcamos el nombre de usuario y contraseña una vez se haya comprobado que la dirección y el puerto existen y son correctos:

```
# ftp 192.168.10.10:21
Conectado to 192.168.10.10.
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
```

```
220-Sólo acceso autenticado
220 ++++++++++
Usuario (192.168.10.10:(none)):
331 Contraseña requerida
Constraseña:
230-Sesión iniciada
ftp>

# ftp ftp.ejemplo.es
Conectado a ftp.ejemplo.es
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
220-Sólo acceso autenticado
220 ++++++++++
Usuario (ftp.ejemplo.es:(none)):
331 Contraseña requerida
Constraseña:
230-Sesión iniciada
ftp>
```

3. Con el nombre de usuario y/o contraseña, además de la URL o IP. Esta forma no nos solicitará ninguna entrada manual por teclado y nos mostrará el símbolo de sistema del servidor FTP:

```
# ftp admin:admin@192.168.10.10
connected to 192.168.10.10.
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
220-Sólo acceso autenticado
220-Usuario y contraseña correctos
220 ++++++++++
230-Sesión iniciada
ftp>
```

Como en el primer ejemplo, una vez que ya estemos dentro en la consola de comandos de FTP, ya podremos dar órdenes.

Como es habitual, existen múltiples comandos, pero aquí mostramos los más recurrentes:

Comando	Significado
?	Muestra el texto de ayuda.
!	Permite regresar a la línea de comandos del sistema operativo sin cortar la conexión. Una vez que se desee volver a la sesión FTP se debe introducir el comando exit.
bye o quit	Finaliza la conexión con el servidor y cierra el cliente FTP devolviendo el control a la línea de comandos del sistema operativo.
cd	Permite cambiar el directorio de trabajo actual en el sistema remoto.
close o disconnect	Finaliza la conexión sin terminar la sesión FTP, es decir, cierra la conexión, pero mantiene abierto el cliente FTP.
delete	Permite eliminar un archivo del sistema remoto.
get	Permite transferir un archivo desde el sistema remoto a la máquina local.
ls	Permite mostrar el contenido de un directorio en el sistema remoto.
mkdir	Permite crear un directorio en el sistema remoto.
open	Inicia la conexión con el nombre de servidor o IP que se le indique a continuación.
put	Permite transferir un archivo desde la máquina local al sistema remoto.
pwd	Permite mostrar el directorio de trabajo actual en el sistema remoto.
user	Permite cambiar de cuenta a partir del nombre de usuario y contraseña sin necesidad de finalizar la conexión con el servidor.

2.4.1.2 CONFIGURACIÓN Y CONEXIÓN DESDE LÍNEA DE COMANDOS EN WINDOWS

Al igual que sucede en GNU/Linux, el comando **ftp** puede utilizarse sin opciones ni parámetros, aunque lo más normal es usarlo, al menos, con la dirección IP o URL. Por tanto, la forma genérica del comando **ftp** podría definirse cómo:

```
# ftp [opciones] [usuario] [dirección[:puerto]]
```

En lo referente a las opciones existen varias, pero las más recurrentes son las siguientes:

Opción	Significado
-A	Permitir el inicio de sesión anónima.
-d	Permite habilitar el modo de depuración.
-i	Desactivar el modo interactivo durante la transferencia de varios archivos.
-n	Deshabilitar la posibilidad de realizar inicios de sesión automáticos en la conexión inicial.
-v	Suprimir la presentación de las respuestas del servidor remoto.

Para hacer la conexión podemos recurrir a tres posibles formas:

1. Sin parámetros. Esta forma nos mostrará el símbolo de sistema del servidor FTP, pero no estaremos conectados. Para conectar deberemos introducir el comando FTP **open**:

```
# ftp
ftp> open ftp.ejemplo.es
Conectado a ftp.ejemplo.es
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
220-Sólo acceso autenticado
220 ++++++++++
Usuario (192.168.10.10:(none)):
331 Contraseña requerida
Constraseña:
230-Sesión iniciada
ftp>
```

2. Con únicamente la URL o IP, con o sin puerto. Esta forma nos solicitará que introduzcamos el nombre de usuario y contraseña una vez se haya comprobado que la dirección y el puerto existen y son correctos:

```
# ftp 192.168.10.10:21
Conectado to 192.168.10.10.
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
220-Sólo acceso autenticado
220 ++++++++++
Usuario (192.168.10.10:(none)):
331 Contraseña requerida
Constraseña:
230-Sesión iniciada
ftp>

# ftp ftp.ejemplo.es
Conectado a ftp.ejemplo.es
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
220-Sólo acceso autenticado
220 ++++++++++
Usuario (ftp.ejemplo.es:(none)):
331 Contraseña requerida
Constraseña:
230-Sesión iniciada
ftp>
```

3. Con el nombre de usuario y/o contraseña, además de la URL o IP. Esta forma no nos solicitará ninguna entrada manual por teclado y nos mostrará el símbolo de sistema del servidor FTP:

```
# ftp admin:admin@192.168.10.10
connected to 192.168.10.10.
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
220-Sólo acceso autenticado
220-Usuario y contraseña correctos
220 ++++++++++
230-Sesión iniciada
ftp>
```

Como en el primer ejemplo, una vez que ya estemos dentro en la consola de comandos de FTP, ya podremos dar órdenes. Como es habitual, existen múltiples comandos, pero aquí mostramos los más recurrentes:

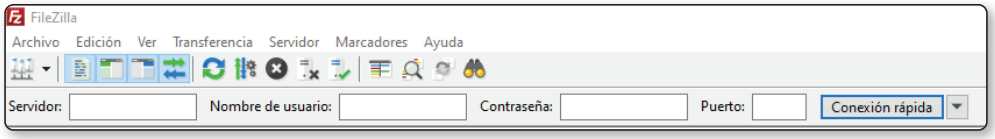
Comando	Significado
?	Muestra el texto de ayuda.
!	Permite regresar a línea de comandos del sistema operativo sin cortar la conexión. Una vez que se desee volver a la sesión FTP se debe introducir el comando exit.
bye o quit	Finaliza la conexión con el servidor y cierra el cliente FTP devolviendo el control a la línea de comandos del sistema operativo.
cd	Permite cambiar el directorio de trabajo actual en el sistema remoto.
close o disconnect	Finaliza la conexión sin terminar la sesión FTP, es decir, cierra la conexión, pero mantiene abierto el cliente FTP.
delete	Permite eliminar un archivo del sistema remoto.
get	Permite transferir un archivo desde el sistema remoto a la máquina local.
ls	Permite mostrar el contenido de un directorio en el sistema remoto.
mkdir	Permite crear un directorio en el sistema remoto.
open	Inicia la conexión con el nombre de servidor o IP que se le indique a continuación.
put	Permite transferir un archivo desde la máquina local al sistema remoto.
pwd	Permite mostrar el directorio de trabajo actual en el sistema remoto.
user	Permite cambiar de cuenta a partir del nombre de usuario y contraseña sin necesidad de finalizar la conexión con el servidor.

2.4.2 Configuración y conexión desde Filezilla

Filezilla permite realizar conexiones con un servidor FTP a través de lo que ellos denominan "**Conexión rápida**" o mediante la opción que han denominado "**Gestor de Sitios**".

2.4.2.1 CONEXIÓN RÁPIDA

Cuando abrimos el cliente FTP Filezilla, en la parte superior tenemos una pantalla como la siguiente:



Si nos fijamos en la parte inferior de la imagen podremos observar que hay varias cajas de texto etiquetadas en donde se nos permite introducir todos los datos necesarios.

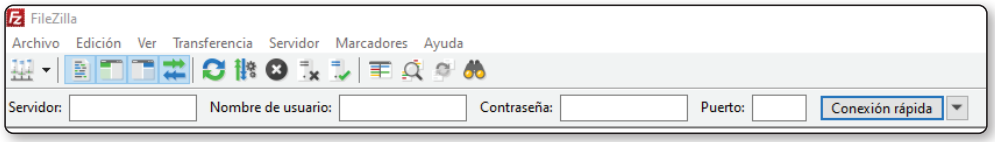
Pues bien, para realizar la conexión rápida deberemos introducir el nombre de host en el campo Servidor, el nombre de usuario en el campo Nombre de usuario y la contraseña en el campo Contraseña. Una vez introducidos los datos ya estaremos en disposición de pulsar en el botón de **conexión rápida** y acceder a nuestro servidor FTP.

Si el campo Puerto se queda vacío la conexión se realizará por el puerto por defecto que, como ya sabemos, habitualmente es el 21.

No obstante, podemos ver que a la derecha de botón de conexión rápida hay otro botón con un icono con forma triangular hacia abajo. Este otro botón funciona como un anexo al botón de conexión rápida permitiéndonos borrar todos los datos de la conexión rápida, borrar todo el historial y/o permitiéndonos realizar una nueva conexión a un servidor al que ya nos conectamos anteriormente.

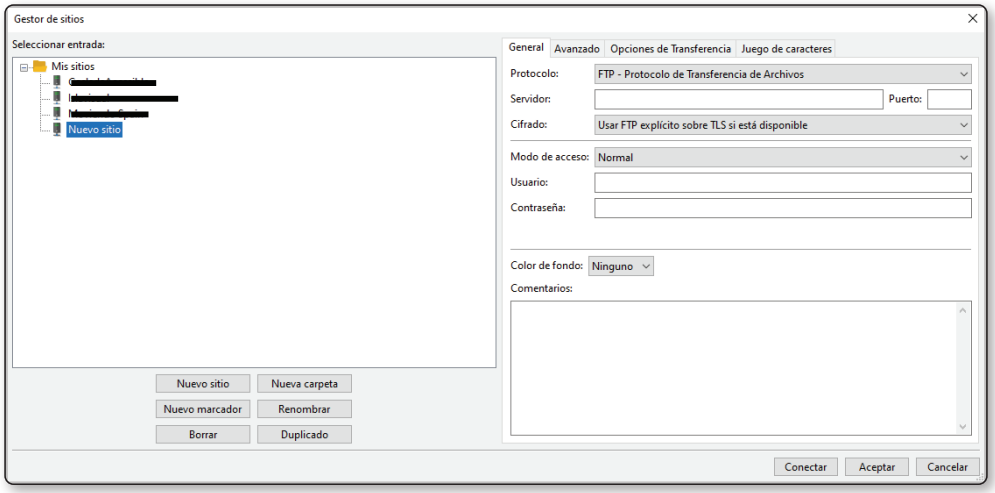
2.4.2.2 GESTOR DE SITIOS

Cuando abrimos el cliente FTP Filezilla, en la parte superior tenemos una pantalla como la siguiente:



Si nos fijamos en el primer botón de la parte superior derecha de la imagen podremos acceder al **Gestor de Sitios**. Ahora bien, también es posible acceder al Gestor de Sitios a través del atajo de teclado **Ctrl-S**.

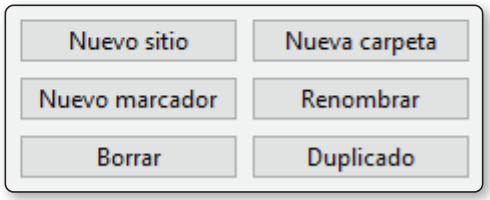
Tras pulsar en dicho botón se nos debería mostrar una pantalla como la mostrada a continuación:



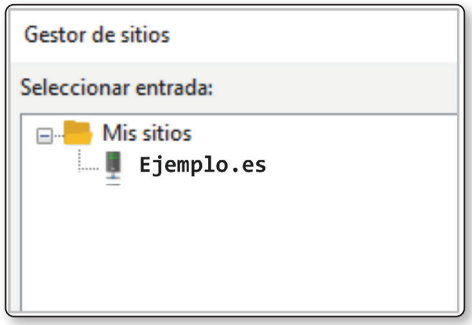
En la parte superior derecha podemos ver **Mis sitios**. Esta zona de la pantalla o sección contendrán todas nuestras conexiones y nos permitirá acceder a ellas siempre que queramos.

Justo debajo de la zona o sección de **Mis sitios** encontraremos una zona o sección de botones o acciones y, a la derecha de ambas, una sección con un conjunto de pestañas denominadas **General**, **Avanzado**, **Opciones de Transferencia** y **Juego de caracteres** que nos permitirán configurar de forma precisa nuestra conexión con el servidor.

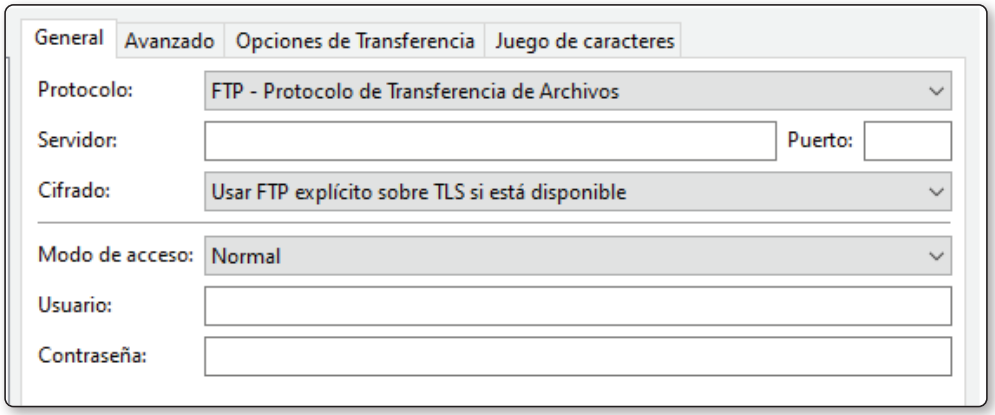
Si focalizamos la atención en la zona de botones o acciones, observaremos varias posibilidades. La primera de todas es la posibilidad de crear un **Nuevo sitio**.



Al pulsar dicho botón, se creará una nueva entrada en "Mis Sitios" con el nombre Nuevo sitio y se quedará seleccionada para que podamos ponerle el nombre que queramos.



A continuación, deberemos introducir el nombre o datos de host, el protocolo, el cifrado, el puerto (si procede), el modo de acceso y el nombre de usuario y contraseña que nos hayan proporcionado.



En **Servidor** deberemos introducir la URL o IP del host y en **Puerto** deberemos insertar el puerto válido, que generalmente será 21 si el protocolo a utilizar es FTP o 22 si el protocolo a utilizar es SFTP.

En **Protocolo** deberemos elegir el que esté configurado para la conexión en el ser. Esto es "FTP – Protocolo de Transferencia de Archivos", "SFTP – SSH File Transfer Protocol" o "Storj – Almacenaje descentralizado en la nube". Aunque la opción por defecto es **FTP – Protocolo de Transferencia de Archivos**, se recomienda usar **SFTP – SSH File Transfer Protocol**, si está disponible.

En **Cifrado** deberemos indicar si se va a utilizar una conexión explícita una conexión explícita o implícita y si va a ser segura o no. Aquí las opciones son "Usar FTP explícito sobre TLS si está disponible", "Requiere FTP explícito sobre TLS", "Requiere FTP implícito sobre TLS" y "Usar sólo FTP plano (inseguro)". Aunque la opción de **Usar FTP explícito sobre TLS si está disponible** es la que se establece por defecto, en la mayoría de los casos suele ser recomendable la opción de **Requiere FTP explícito sobre TLS**.

En lo referente al **Modo de acceso** admite el modo anónimo y varios modos con nombre de usuario y contraseña. En esta opción de configuración lo frecuente es que se establezca a **Normal** o **Preguntar la contraseña**, no obstante, tiene otras opciones. En función de qué modo de acceso seleccionemos, los campos de nombre de usuario y contraseña aparecerán o no habilitados.

Finalmente, cuando ya hayamos introducido todos los datos necesarios, estaremos en disposición de pulsar el botón de **Conectar**, ubicado en la esquina inferior derecha de la pantalla de "Gestor de sitios".

Tras pulsar en el botón de **Conectar** se procederá a realizar la conexión con el servidor y se almacenará la configuración del sitio automáticamente en nuestro Gestor de Sitios teniéndolo disponible para futuras ocasiones desde aquí.

2.5 ÓRDENES PARA TRANSFERIR ARCHIVOS

La transferencia de archivos sólo será posible una vez que estemos conectados al servidor. Como es lógico, la transferencia será bidireccional, es decir, podremos descargar archivos, lo que será equivalente a transferir un archivo desde el servidor remoto hasta nuestro directorio local, o subir archivos, lo que será equivalente a transferir un archivo desde nuestro directorio local hasta el servidor remoto.

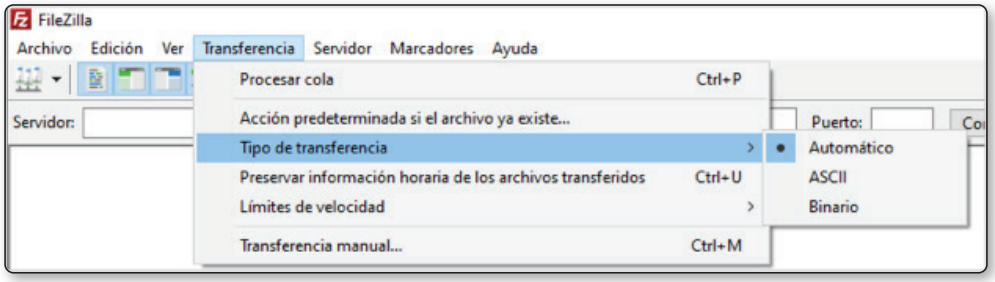
Sin embargo, aunque la transferencia se considere bidireccional, podremos encontrarnos con casos en donde sólo se pueda subir o sólo se pueda descargar.

2.5.1 Formas de transferir archivos

Aunque parezca que no, conocer la forma de transferir es importante. La razón de que sea importante es porque, dependiendo de cómo lo hagamos, podremos alterar o destruir la información que se desea transferir.

En general, la transferencia puede ser a modo de texto plano, lo que suelen denominar **ASCII**, o **binaria**, sin embargo, en los clientes FTP como Filezilla

disponemos de la opción de **Automático**, la cual permite que sea el programa quien decida de qué tipo es.



Para hacernos una idea, el tipo de transferencia **ASCII** será el recomendable o utilizado para transferir archivos HTML o con extensiones como DOCX, TXT, INI o CSV. Sin embargo, el tipo de transferencia **Binario** será el recomendable o utilizado para transferir imágenes (PNG, JPG, WEBP, ...), ejecutables, archivos comprimidos o empaquetados (ZIP, RAR, 7Z, TAR, ...), o de tipo multimedia como audios o vídeos (MP3, MP4, AVI, ...), entre otros.

Para realizar un cambio de tipo de transferencia desde el cliente FTP de línea de comandos debemos recurrir a los comandos **binary** y **ascii**. No obstante, también es frecuente encontrar **bin** como abreviatura del comando **binary**.

```
# ftp
ftp> binary
200 Type set to I

ftp> ascii
200 Type set to A
```

2.5.2 Transferencia de archivos desde GNU/Linux

La transferencia de directorios y archivos entre el servidor remoto y el cliente bajo la línea de comandos puede que se vuelva un proceso bastante tedioso, sin embargo, es posible que sea más rápido y eficiente.

Aunque la mayoría ya los hemos comentado antes, los volveremos a mostrar a modo de recordatorio y refuerzo:

Comando	Significado
cd	Permite cambiar el directorio de trabajo actual en el sistema remoto. `ftp> cd rss`
dir	Muestra el contenido del directorio del sistema remoto. `ftp> dir`
get	Permite transferir un archivo desde el sistema remoto a la máquina local. `ftp> get rss.php`
lcd	Permite cambiar el directorio de trabajo actual en el sistema local. `ftp> lcd rss`
ls	Permite mostrar el contenido de un directorio en el sistema remoto. `ftp> ls -a`
mget	Permite transferir múltiples archivos desde el sistema remoto hasta la máquina local. `ftp> mget index.html cities.json`
mkdir	Permite crear un directorio en el sistema remoto. `ftp> mkdir images`
mput	Permite transferir múltiples archivos desde la máquina local hasta el sistema remoto. `ftp> mput index.html cities.json`
put o send	Permite transferir un archivo desde la máquina local al sistema remoto. `ftp> put rss.php`
pwd	Permite mostrar el directorio de trabajo actual en el sistema remoto. `ftp> pwd`

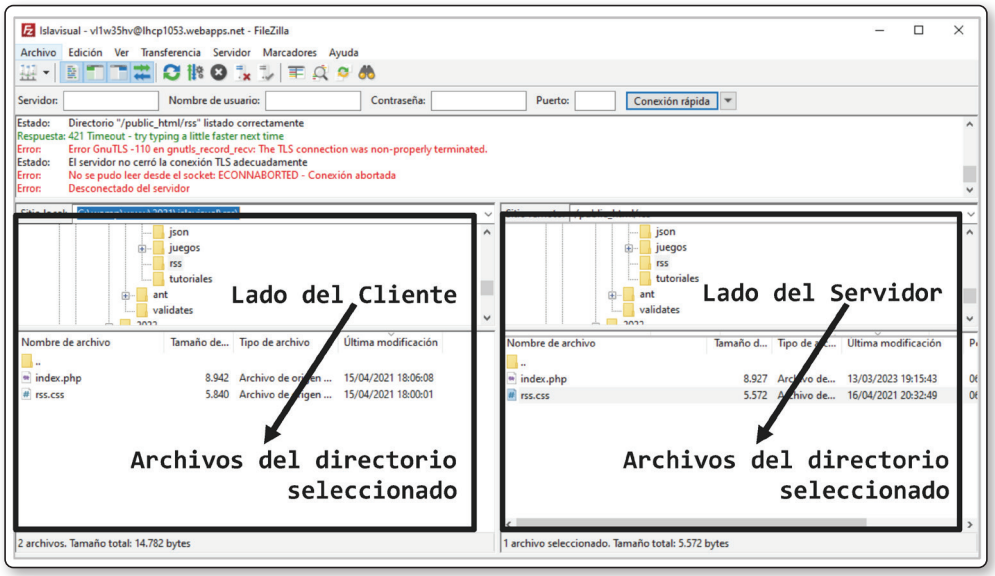
A continuación, se muestran una posible secuencia de comandos dentro de una sesión FTP:

```
# ftp admin:admin@192.168.10.10
connected to 192.168.10.10.
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
220-Sólo acceso autenticado
220-Usuario y contraseña correctos
220 ++++++++++
230-Sesión iniciada
ftp> ls -a
200 PORT command successful
150 Conectando con el puerto 62456
.
..
imgs
js
css
pub
index.html
226-Opciones: -a
```

```
226 7 coincidencias en total
ftp: 2725 bytes recibidos en 0,008 segundos 332,65 a KB/s
ftp> cd imgs
250 OK. Directorio actual es /imgs
150 Conectando con el puerto 47667
ftp> mkdir svg
250 "/imgs/svg creado satisfactoriamente
ftp> lcd imgs
El directorio local es ahora /home/user/imgs
ftp> cd ..
250 OK. Directorio actual es /imgs
150 Conectando con el puerto 32456
ftp> mput *.png
mput logo.png? y
200 Port command succesful
150 Openikng data channel for "/imgs/file logo.png"
226 Successfuly transferred "/imgs/file logo.png"
16155 bytes sent in 0.00 secs (17.329 MB/s)
ftp> quit
```

2.5.3 Transferencia de archivos con Filezilla

La transferencia de directorios y archivos entre el servidor remoto y el cliente bajo Filezilla es muy posible que se vuelva un proceso muy simple de fácil aprendizaje. Tan simple como conocer unos pocos datos y arrastrar y soltar de un lado a otro. Esto es posible, sobre todo, a que la apariencia de la aplicación es análoga o similar a la del explorador de archivos del Windows y Linux.



Si nos fijamos en la imagen anterior podremos observar dos zonas o secciones bien diferenciadas. A la izquierda tenemos el lado del cliente (también denominado local) en donde se encuentra la estructura de directorios y archivos que tenemos en el disco duro de nuestro ordenador. A la derecha tenemos el lado del servidor en donde se encuentra la estructura de directorios y archivos que están publicados, disponibles o subidos en la infraestructura remota.

Además, como se puede apreciar, en cada una de las zonas podemos diferenciar la parte de directorios arriba y la parte de archivos abajo. Por tanto, cuando pulsemos en una de las carpetas de uno de los lados, seguidamente, se mostrarán todos sus archivos abajo.

El problema que aquí podríamos tener, si es que se le puede llamar problema, es que, si cambiamos de directorio en un lado, lo suyo sería que se pusiera en el lado del servidor el mismo directorio, siempre y cuando exista y tenga la misma ruta relativa. Para poder hacer esto Filezilla nos ofrece un botón que nos permite hacer una navegación sincronizada, de forma que el directorio del cliente siempre esté alineado con el directorio del servidor. Este icono es el que está a la izquierda del botón con forma de prismáticos:

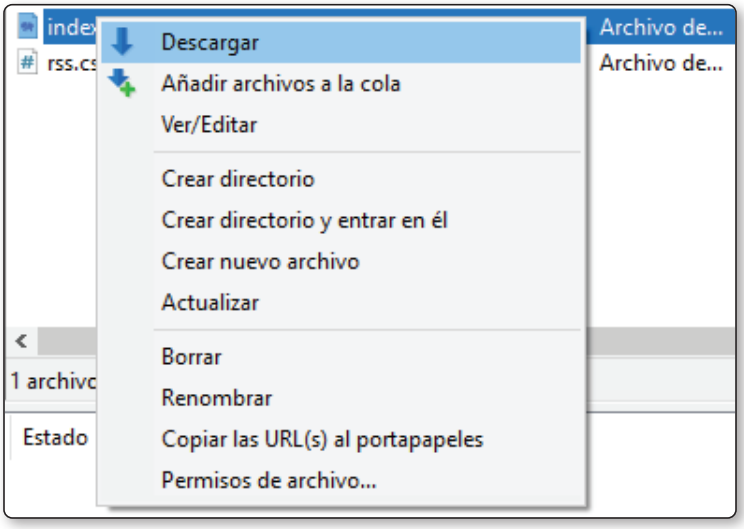
Una vez situados en el directorio local y remoto con los que deseamos trabajar será ya el momento de transferir los archivos deseados. Para ello, tan sólo tendremos que arrastrar de un lado al otro el o los archivos deseados.

Tras lo cual, se procederá a la descarga o subida de dichos ficheros (si es que tenemos permisos de escritura en el servidor) provocando que los archivos de ambos lados sean idénticos, al menos, a nivel de contenido, nombre y extensión.

ⓘ NOTA

Recordemos que en Filezilla es posible seleccionar los archivos de igual forma que se haría en el explorador de archivos, es decir, que es posible seleccionar múltiples archivos usando el puntero del ratón en combinación con las teclas Shift o Mayúscula y/o la tecla Control.

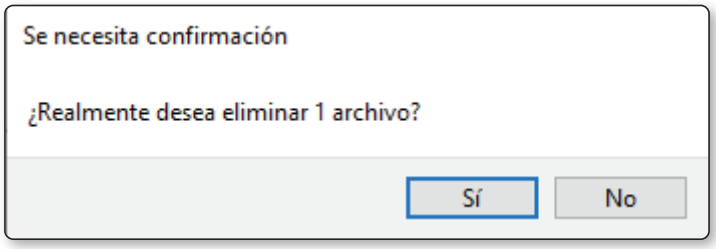
En lo referente a las acciones de creación y eliminación de directorios y archivos, también es posible hacerlo pulsando en el botón derecho o secundario del ratón:



En la imagen anterior podemos apreciar que se nos ofrecen diversas opciones como **Descargar** el archivo seleccionado, **Crear directorio**, **Crear directorio y entrar en él**, **Renombrar** el archivo seleccionado o **Borrar** el archivo seleccionado.

En lo referente al eliminado cabe destacar que es una operación u orden que hay que ejecutar con cautela debido, fundamentalmente, a que no tiene vuelta atrás ya que no existe ni tiene asociada una carpeta con funcionalidad de papelera.

Es por ello por lo que, cuando eliminemos un archivo, sea con la opción de Borrar del menú contextual del ratón, o bien a través de su atajo de teclado, la tecla Supr, se nos mostrará un mensaje de confirmación:



2.6 ÓRDENES PARA ACTUALIZAR Y ELIMINAR ARCHIVOS EN GNU/LINUX

Si lo que deseamos es actualizar directorios o ficheros desde la línea de comandos podemos recurrir a los siguientes comandos:

Comando	Significado
? comando	Muestra la ayuda del comando.
? o help	Muestra todos los comandos disponibles.
append	Permite añadir un archivo local a un archivo en el sistema remoto. `ftp> mput index.html cities.json`
delete	Permite eliminar un archivo del sistema remoto. `ftp> cd rss`
mdelete	Permite eliminar varios archivos en el sistema remoto. `ftp> dir`
prompt	Fuerza a que se muestren los avisos interactivos al transferir varios archivos. `ftp> lcd rss`
rename	Permite cambiar el nombre de un archivo en el sistema remoto. `ftp> ls -a`
reset	Permite eliminar las respuestas de comandos en cola. `ftp> reset`
rmdir	Permite eliminar un directorio en el sistema remoto. `ftp> mget index.html cities.json`
rstatus	Permite ver el estado del sistema remoto. `250 "/imgs/svg creado satisfactoriamente`
status	Permite ver el estado de la interfaz FTP. `ftp> mkdir images`

A continuación, se muestran una posible secuencia de comandos dentro de una sesión FTP:

```
# ftp admin:admin@192.168.10.10
connected to 192.168.10.10.
220-Bienvenido a ejemplo.es ftp server
220-Sin acceso anónimo
220-Sólo acceso autenticado
220-Usuario y contraseña correctos
220 ++++++++++
230-Sesión iniciada
ftp>
ftp> ls -a
200 PORT command successful
150 Conectando con el puerto 62456
```

```
.
..
imgs
js
css
pub
index.html
226-Opciones: -a
226 7 coincidencias en total
ftp: 2725 bytes recibidos en 0,008 segundos 332,65 a KB/s
ftp> rename imgs images
350 Ready for RNTO.
250 Rename successful.
ftp> delete index.html
250 Delete operation successful.
ftp> mdelete css/*.css
mdelete styles.css? y
250 Delete operation successful.
ftp>
ftp> quit
```