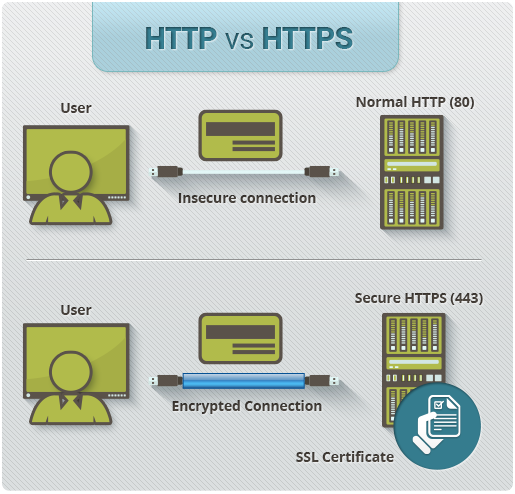
* **What is HTTPS?**

Hyper Text Transfer Protocol Secure (HTTPS) is the secure version of HTTP, the protocol over which data is sent between your browser and the website that you are connected to. The 'S' at the end of HTTPS stands for 'Secure'. It means all communications between your browser and the website are encrypted. HTTPS is often used to protect highly confidential online transactions like online banking and online shopping order forms.

Web browsers such as Internet Explorer, Firefox and Chrome also display a padlock icon in the address bar to visually indicate that a HTTPS connection is in effect.



### How Does HTTPS Work?

HTTPS pages typically use one of two secure protocols to encrypt communications - [SSL (Secure Sockets Layer)](https://www.instantssl.com/ssl.html) or TLS (Transport Layer Security). Both the TLS and SSL protocols use what is known as an 'asymmetric' Public Key Infrastructure (PKI) system. An asymmetric system uses two 'keys' to encrypt communications, a 'public' key and a 'private' key. Anything encrypted with the public key can only be decrypted by the private key and vice-versa.

As the names suggest, the 'private' key should be kept strictly protected and should only be accessible the owner of the private key. In the case of a website, the private key remains securely ensconced on the web server. Conversely, the public key is intended to be distributed to anybody and everybody that needs to be able to decrypt information that was encrypted with the private key.

#### ****What is FTPS?****

FTPS (also known as FTP Secure) is an evolution of the widely used File Transfer Protocol (FTP). Because FTP is not typically considered a secure file transfer channel, FTPS was proposed as an alternate in RFC 2228. FTP provides the foundation for FTPS, but the latter includes an additional encryption layer. In FTPS, FTP data travels through the network using either Secure Sockets Layer (SSL) or Transport Layer Security (TLS) protocols.

Just like FTP does, FTPS also works in a client-server model, utilizing a control channel and a data channel for exchanging FTP commands and data during an FTPS client session.

#### ****How Security Works In FTPS****

An FTPS connection is authenticated with a user ID, password and public key certificate (similar to how HTTPS works). Tools such as OpenSSL allow key certificates to be requested and created. An FTPS client, when connecting to an FTPS server, will first verify the trustworthiness of the server’s certificate.

 When a trusted certificate authority (CA) signs these certificates, it ensures that the client is being connected to a trusted and secure server, which helps protect against man-in-the-middle attack.

* Certificates not signed by a trusted CA, which are known as self-signed certificates, may prompt the FTPS client to generate a warning that the certificate is not valid. The client can choose to accept the certificate or reject the connection.

FTPS (over SSL/TLS) uses X.509 certificates for authentication. These digital certificates include a public encryption key and information about the certificate owner. The public key has two major functions: validation and data encryption. The public key has an associated private key. This private key is stored separately from the certificate, which is used for decrypting the message encrypted by the public key.

#### ****Implicit FTPS and Explicit FTPS****

Implicit FTPS refers to sessions where both the command and data channels are encrypted at all times. An SSL encryption is implied at the beginning of the session, which means secure FTPS connection is mandatory. In this scenario, a non-FTPS client will not be allowed to communicate with the FTPS server. The FTPS server defines a specific port (990) for the client to be used for secure connections.

 Implicit FTPS consumes a lot of network bandwidth and computational resources because encryption happens in both the command and data channels. In a scenario where a user wants to upload non-confidential files to the FTPS server, an explicit FTPS connection would be used instead of an implicit FTPS connection.

 In explicit FTPS, the client directly requests security from the FTPS server. This is an optional request. If a client does not request security, the FTPS server can either allow the client to continue in unsecure mode or refuse or limit the connection.

 Explicit FTPS can be used in scenarios where the requirement is to secure only the command channel (which carries the commands and user authentication,) and not the data channel (which carries non-confidential FTP data). Port 21 is the default port used by the FTP server to communicate with the client. This allows both unsecure FTP and secure FTPS clients to connect to the FTPS server.

 For organizations adhering to federal regulatory compliance standards, implicit FTPS is recommended.

***Benefits of FTPS over FTP***

* Communication can be read and understood by humans
* FTPS can be used for server-to-server file transfer requirements
* SSL/TLS has good authentication mechanisms, including X.509 certificate features
* Many Internet communication frameworks have built-in FTP and SSL/TLS support

**IRC(Internet Relay Chat)**

* Stands for "Internet Relay Chat." IRC is a service that allows people to chat with each other [online](https://techterms.com/definition/online). It operates on a client/server model where individuals use a [client](https://techterms.com/definition/client) program to connect to an IRC [server](https://techterms.com/definition/server). Popular IRC clients include mIRC for Windows and Textual for OS X. Several web-based clients are also available, including KiwiIRC and Mibbit.
* In order to join an IRC conversation, you must choose a [username](https://techterms.com/definition/username) and a channel. Your username, also called a [handle](https://techterms.com/definition/handle), can be whatever you want. It may include letters and numbers, but not spaces. A channel is a specific chat group within an IRC network where users can talk to each other. Some networks publish lists of available channels, while others require you to manually enter channel names in order to join them. Channels always begin with a [hashtag](https://techterms.com/definition/hashtag) followed by a name that represents their intended chat topic, such as "#teenchat," "#politics," or "#sports". Some IRC channels require a [password](https://techterms.com/definition/password) while others are open to the public.
* When you join a channel, the chat [window](https://techterms.com/definition/window) will begin displaying messages people are typing. You can join the conversation by typing your own messages. While channel members can type whatever they want, popular channels are often moderated. That means human operators or automated [bots](https://techterms.com/definition/bot) may kick people out of the channel and even ban users who post offensive remarks or [spam](https://techterms.com/definition/spam) the channel with repeated messages.
* While IRC was designed as a public chat service, it supports other features such as private messaging and file transfers. For example, you can use an IRC command (which typically begins with a forward slash "/") to request a private chat session with another user. Then you can use another IRC command to send the user a [file](https://techterms.com/definition/file) from your local system.
* NOTE: IRC was a popular way for users to connect online before [social media](https://techterms.com/definition/social_media) became prevalent in the early 2000s. Today, many people still use IRC, but social media sites and apps are much more popular.