

**FTP vs SFTP comparison**  
**Author: Leo Ni**

**FTPS (FTP Over SSL):** FTPS provides an extension to FTP via SSL/TLS channels. SSL/TLS provides channel encryption in compensating the security issue of FTP. When FTP and SSL/TLS are used together, it can achieve the objective of wide usage and secure data transmission (a.k.a. FTPS).

**SFTP (SSH FTP):** SFTP is commonly regarded as SSH File Transfer Protocol, which is leveraging SSH to transfer files. (Note: SFTP is not FTP over SSH)

Below is the compare table of Pros and Cons between FTPS (FTP Over SSL) and SFTP (SSH FTP)

		<b>FTPS (FTP Over SSL)</b>	<b>SFTP (SSH FTP)</b>
Mechanism in Brief		Leveraging SSL/TSL	Leveraging SSH File Transfer Protocol
C-I-A Triad	Confidentiality (To protect the data from unauthorized access)	Asymmetric algorithm (RSA, DSA, etc.) Symmetric algorithm (DES/3DES, AES, etc.) Key-exchange Algorithm (Diffie-Hellman, etc.)	
	Integrity (To protect the data from unauthorized modifications)	Depending on software features: Hash functions (Standard Hash such as MD5, SHA1 / Non-Standard Hash functions)	
	Authentication (To Identify Who You Are)	X.509 Certificates (Support PKI)	SSH Keys (Public keys) - Although some SSH software claims to support PKI but never validates the whole certificate chain
Key Management Efforts		Easier to manage the keys (PKI) - as long as the CA is reliable	Special attention should be given to managing of the SSH Keys
General Adoption Trend		Widely used based on existing legacy FTP applications via SSL/ TLS support (No need to adopt new technologies)	Increasing percentage of adopting SFTP due to easier firewall settings
Implementation Difficulty		SSL/TLS is widely built-in supported in most OS	Need to ensure SSH service is installed / enabled
		Secondary Data Channel is required	Only One Channel is required (Easier

	(More complex firewall settings)	firewall settings)
Major Features Supported	Supports very basic File / Data operation methods (i.e. Copy, Upload, Download, Delete, etc.)	Supports powerful File / Data operation methods (i.e. File attribute / permission settings, file lock, etc.)
	Loose Directory Standard	Rigid Directory Standard (Granular level of definition)
Server Side Settings (Brief)	Allow inbound connections on port (i.e. Port 21) Define passive port for file transfers and directory listings Allow inbound connections on passive port range	Allow inbound connections on port 22
Client Side Settings (Brief)	Allow outbound connections to port 21 Passive port range defined by server	Allow outbound connections to port 22
Compatibility	Good compatibility due to simple architecture.	In SFTP standard, some features are marked optional or recommended. Different software / vendors may not strictly follow those standards and it may cause some compatibility issues.
Firewall Settings (Brief)	Need to configure firewall to enable both two connections simultaneously	Just need to enable one connection on firewall (Ease of Firewall Management)