FTPS vs. SFTP: Use case comparison

	FTPS	SFTP
Network Security	FTPSs requirements for at least two ports (and possibly many more depending on the volume of file transfer activity) can make troubleshooting difficult and expose novel attack vectors that become possible thanks to the constantly changing data connection between the client and server. Special attention to the network configuration and server security options can help mitigate these risks.	**Ideal Protocol** SFTP uses a single connection port for all communication between a client and server. This tends to greatly Simplify interoperability concerns and reduces the attack surface when compared with FTPS.
Security	**Tie** Due to FTPSs length of time in the market more devices and systems are compatible with FTPS. However, the lack of standardization for many functions can sometimes lead to client and server interoperability issues.	**Tie** SFTP will generally be accepted by more modern devices and systems (Linux and Unix) but is not ideal for communicating in legacy situations. (for example VCL and .NET frameworks do not offer built-in support).
Configuration	Can cause firewall/transmission issues due to more complex configurations required.	**Ideal Protocol** Primarily due to its streamlined connections that reduce firewall issues.
Performance	**Ideal Protocol** Offers the highest possible secure transfer	SFTP transfers carry a lot more overhead due to the robustness and flexibility of the protocol.
File/Directory Manipulation	FTPS's available commands are limited and not standardized, which can require additional administrative configuration.	**Ideal Protocol** Offers a number of standardized controls and commands for activities such as file directory manipulation, permissions locking, etc.
Server to Server Communications	**Ideal Protocol** Due to limitations in SFTP.	Server-to-server communications are not well-supported.
Internet File Transfer	**Ideal Protocol** Due to SSUTLS support built into many internet communications frameworks.	Can be configured but will require extra Steps.
Authentication	**Ideal Protocol** Certificate visibility offers high degree of trust.	SSH keys can be harder to validate because they usually require the server administrator to securely distribute the server's public key to clients ahead of initial connection.