SSH File Transfer Protocol (SFTP) Explained

Author: Conrad Chung, 2BrightSparks

Introduction

In another article we cover the topic Introducing FTP. In this paper we will be looking at a secure alternative to FTP, the SSH File Transfer Protocol.

What is Secure Shell Protocol (SSH) and SSH File Transfer Protocol (SFTP)?

Secure Shell Protocol (SSH) is a protocol used to establish a secure connection between a remote server and a computer. Secure Shell utilizes public key encryption to provide strong user authentication and secure encrypted communications over the Internet. The latest version is SSH2. It was developed and adopted in 2006 by the Internet Engineering Task Force (IETF) to replace older insecure shell protocols.

SSH File Transfer Protocol (SFTP) is designed as an extension of the SSH2 protocol to provide secure file transfer capability. Both command and data connections are encrypted between the client and the FTP server to allow passwords and other sensitive information to be transferred securely over the network.

How to Connect to SFTP?

Connections to SFTP servers are established either by Password Authentication or using Public/Private Keys Authentication. With password authentication which is securely transmitted, a user requires just a username and password to log in to an SFTP server.

Another method is using key pairs. A pair of public and private keys are generated and the public key is stored in the SFTP server. The client with the private key will authenticate with the server during login and if the keys match, the SFTP client will gain access into the system. A passphrase can also be added on the private key for additional security.

Advantages of using SFTP

Data Exchange – SFTP formats commands and data into special packets and sends them through a single connection. It does not use separate channels like FTP does. This eliminates the inconvenience to server administrators who otherwise need to open up a range of ports for conventional FTP connections.

Security – With FTP, both channels are generally sent unencrypted and this data can be intercepted and read by other parties. Some FTP servers enforce security by connecting via Implicit SSL and Explicit SSL connections. However, these are completely different from SFTP which is based on the SSH2 protocol.

Firewall – SFTP uses Port 22 for server inbound connections and client outbound connections. This makes SFTP firewall friendly as it only needs a single port to be opened through the firewall.

Meta-data – With SFTP, users are able to access meta-data like date, time, size, permissions and other information accurately while some FTP servers, especially older ones, may have trouble getting this information correctly, while others do not provide them at all. This can result in data backup discrepancy as the meta-data of local files does not match those on the server.

SyncBackFree/SE/Pro circumvents these issues by implementing advanced techniques to set the last modification date and time of files on the server. Some older FTP servers do not support these commands at all thus an alternative is to enable the Fast Backup feature in SyncBackSE/Pro which will bypass this issue.

With SFTP, all these issues are by-passed as the meta-data is supported and correctly reported.

Conclusion

Data security and integrity is a crucial priority for corporations and government bodies. System administrators who need to enforce robust security policies will always implement SFTP over FTP as it is inherently more secure. At 2BrightSparks, we provide users the ability to back up to SFTP using the advanced backup solution, SyncBackPro.