What Is Secure File Transfer Protocol (SFTP)?

THE COMPLETE SFTP GUIDE

Version 2.1

Thru®

Contents

What Is SFTP?

SFTP stands for SSH File Transfer Protocol or Secure File Transfer Protocol. It is used to secure file transfers between a remote host server and a client user over a public network like the internet. SFTP ensures that the host and client are validated and authenticated.

Important Facts about SFTP

1

SFTP works in a client-server architecture. Clients always initiate a request to connect and servers passively listen for client requests.

2

The server's and client's identities are verified and the connection is encrypted before files are transferred.

3

File transfer is automatically resumed in the event of a break in connection.

4

SFTP clients can remotely manipulate files on the host server, such as copying or deleting.

SFTP Uses SSH

What Is SSH?

Secure Shell (SSH) encrypts identities, passwords and other transmitted data to protect them from theft or tampering by unauthorized entities. Its default port is port 22. SSH refers to two things:

1. The network protocol
2. SSH handshake process

What Is OpenSSH?

OpenSSH is an open-source implementation of the SSH protocol. SSH1 protocol support is disabled by default. OpenSSH is integrated into base operating systems such as Microsoft and Linux Red Hat.

SSH1 versus SSH2

* SSH1 and SSH2 are not compatible with each other.

SSH1

- Has one monolithic protocol.
- Has weak checking using Cyclic Redundancy Check (CRC)-32.
- Supports only one session channel per connection.
- Allows only one form of authentication per session.

SSH2

- Has separate transport, authentication and connection protocols.
- Has strong cryptographic integrity check using a message authentication code (MAC).
- Supports any number of session channels per connection.
- Allows more than one form of authentication per session.

An SSH Key Analogy

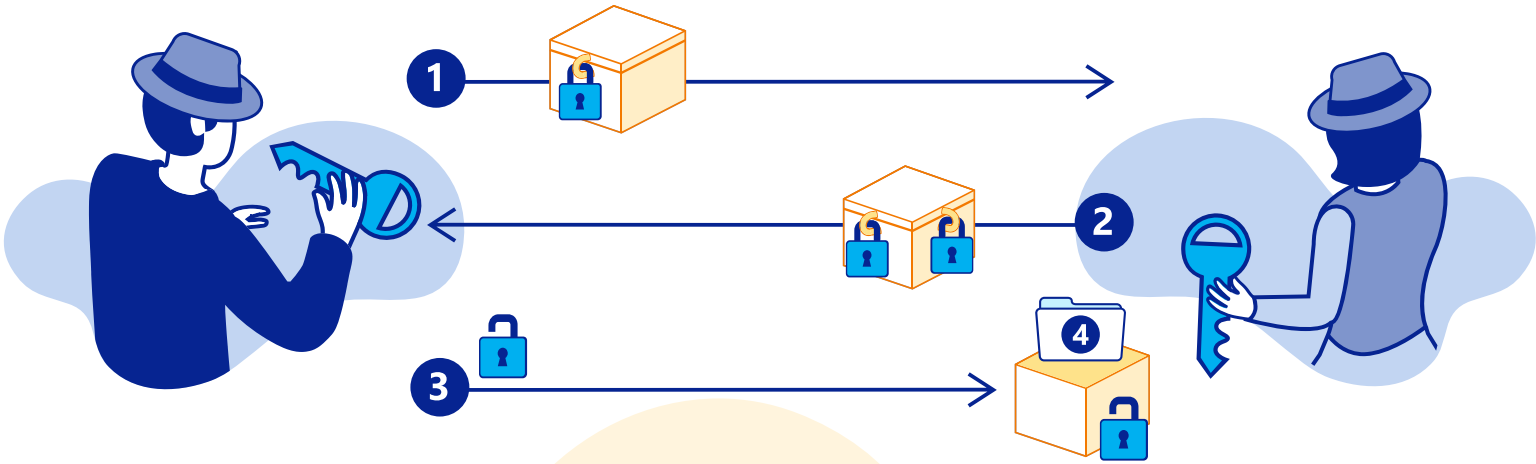
SSH uses keys to authenticate both participants. To understand how SSH keys work, refer to the following analogy:
Jack wants to send a confidential message to his colleague Jill. He needs to verify it is really Jill before he sends the actual message.

1 Jack locks message in a box and sends it to Jill's saved address.

2 Jill receives the box. To verify it is really from Jack, she puts her own lock on the box and sends it to his saved address.

3 Jack recognizes his own lock and Jill's signature lock. He removes his lock and sends it back to her.

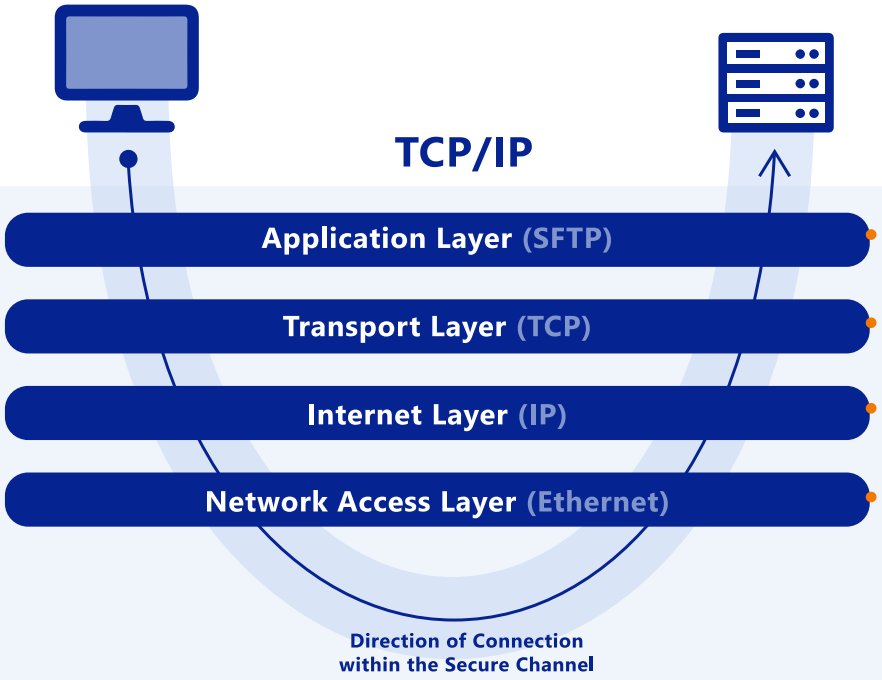
4 Jill receives the box and sees that Jack has removed his lock, so she knows he received it. She removes her own lock to read the secret message.



With an understanding of how SSH keys work, now let's look at how SFTP interacts with other layers of a network →

SFTP in the TCP/IP Model

The file transfer process spans multiple layers of a network. When discussing how SFTP works, it is important to understand how it fits into the Transport Control Protocol/Internet Protocol (TCP/IP) model.



The TCP/IP model helps determine how a computer should be connected to the internet and how data should be transmitted. It is organized into four layers:

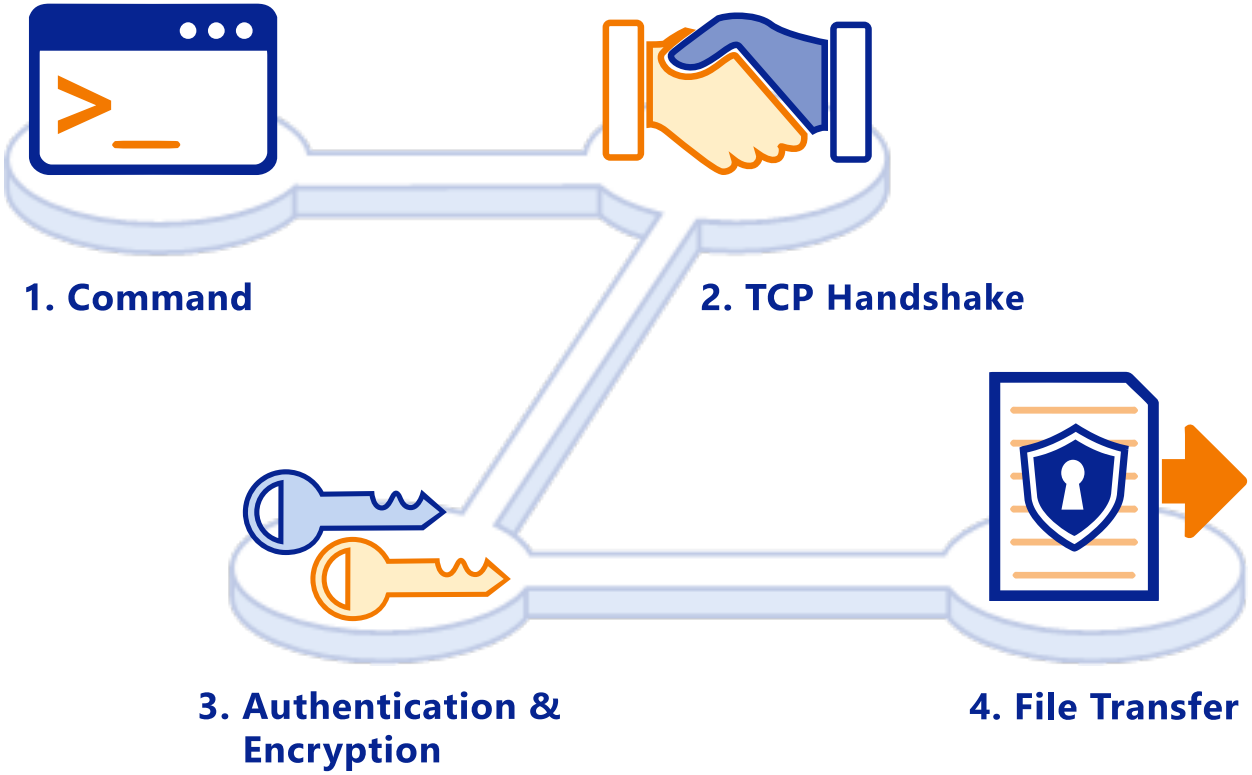
- Protocols that identify communication partners, determine resource availability and synchronize communication.
- Divides the message received from the session layer into segments and sequences them. This ensures data packets are delivered error-free and in order.
- Offers the procedural method for transferring data sequences from source to target with the help of various networks.
- Looks out for hardware addressing and allows for the physical transmission of data.

As a protocol, SFTP lives in the application layer. It uses TCP in the transport layer to execute the TCP handshake and establish a connection across the internet and network layers. Then, it secures the channel so the messages and data traveling across networks are encrypted.

Knowing what layers run underneath SFTP will help with understanding how SFTP works →

How SFTP Works

Authentication, Integrity & Confidentiality



1. User Command

The user runs a command to open the SSH connection.

SFTP can manipulate data remotely to copy files, delete files, etc. This is performed using SFTP commands.

To get a list of available SFTP commands, simply type **`help`** or **`?`**

```
sftp> help
Available commands:
bye        Quit sftp
cd path    Change remote directory to 'path'
…
version    Show SFTP version
!command   Execute 'command' in local shell
!          Escape to local shell
?          Synonym for help
```

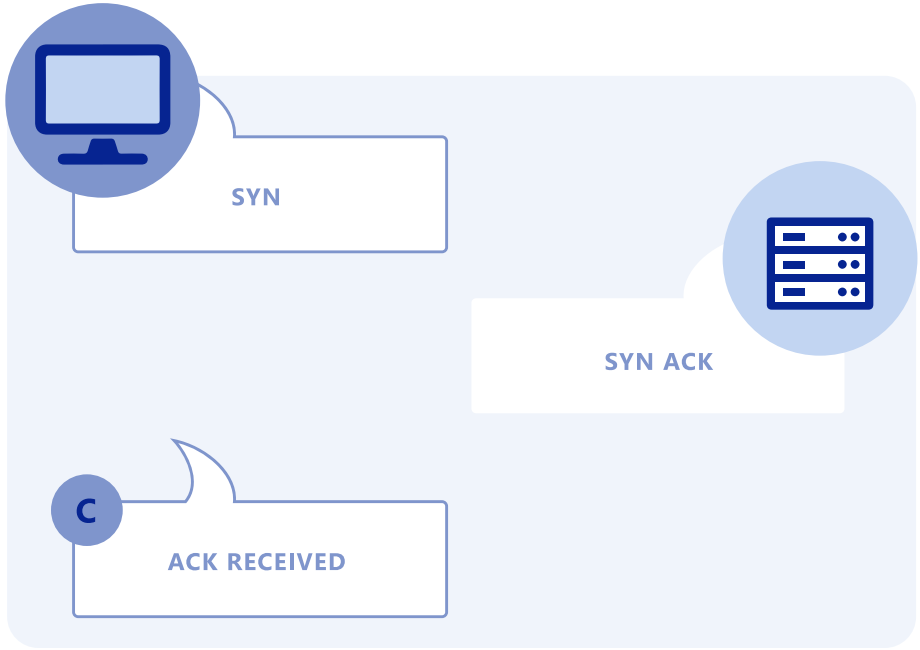
2. TCP Handshake

Before an SFTP file transfer happens, the client and server verify the connection via a three-way TCP handshake.

TCP is a connection-oriented protocol, which means that both computers verify a connection before files are sent.

The handshake occurs in a series of messages between the parties to confirm that they both have access to the correct port in the transport layer (port 22).

If data does not arrive after the handshake is complete, TCP will make sure that it is re-sent.

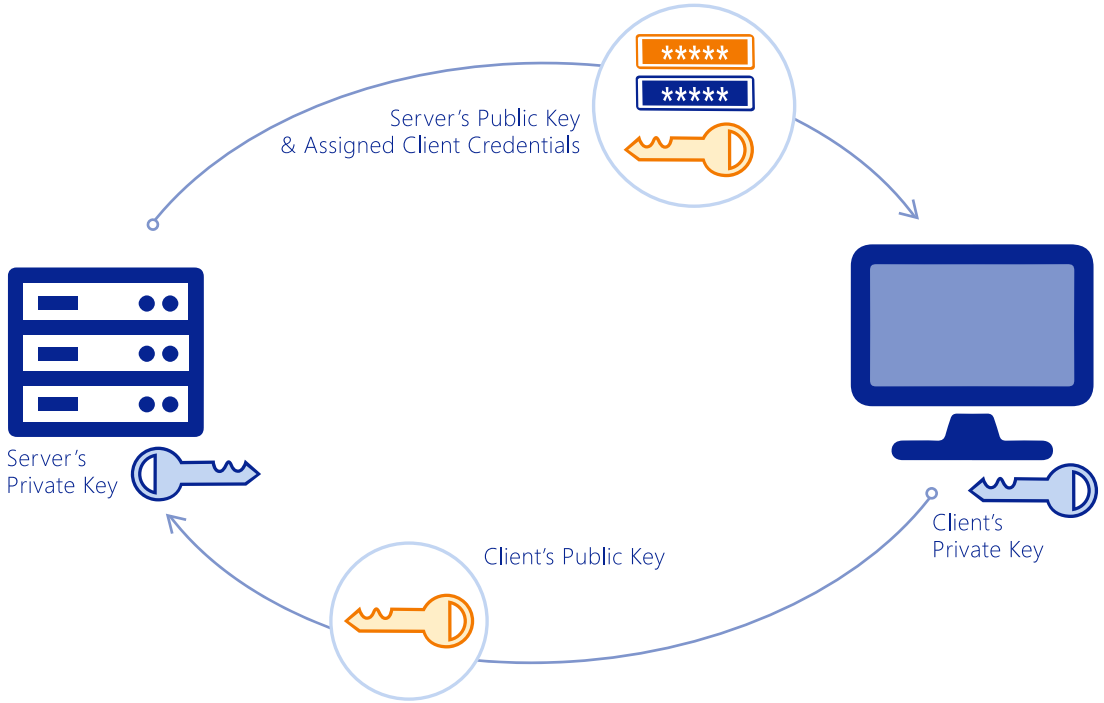


3. Authentication & Encryption

Credentials are created and shared between parties. The credentials validate the server, negotiate a session key and authenticate the client.

The most secure option is for the host server to generate a user and password for the client and for both to create SSH private/public key pairs.

- The first set of key pairs only encrypts the messages between client and server that validate and authenticate the parties.
- The second set of key pairs is used to negotiate the session key and encrypt files.



Authentication Steps (1 of 3)

1. The client verifies the server's identity.
   a. If the client is accessing the server for the first time, the user has to manually verify the server's public key.
   b. If the client is not accessing the server for the first time, the client can verify the server's identity without user involvement.

Public Key Cryptography: How Does it Work?

Public keys can
- Send encrypted data.
- Verify digital signatures.
- Authenticate communicating parties.

Private keys can
- Encrypt and decrypt data.
- Generate digital signatures.
- Authenticate.

*The private key should never be shared because it compromises file transfer security.

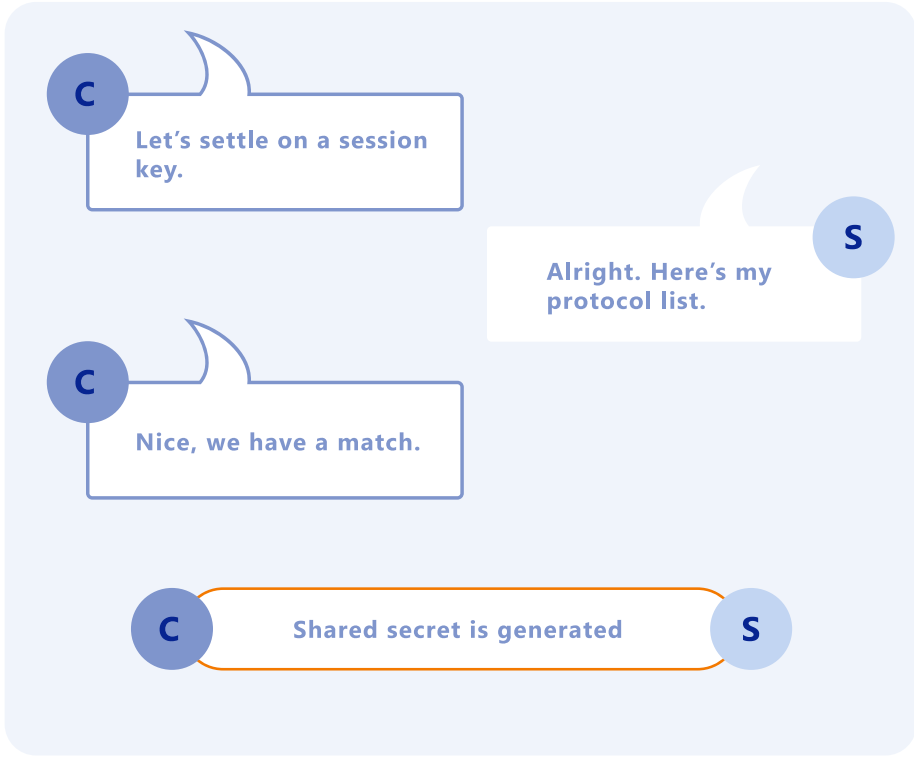
Authentication Steps (2 of 3)

2. Both parties negotiate a session key using the Diffie-Hellman algorithm.

   The session key encrypts the entire session.

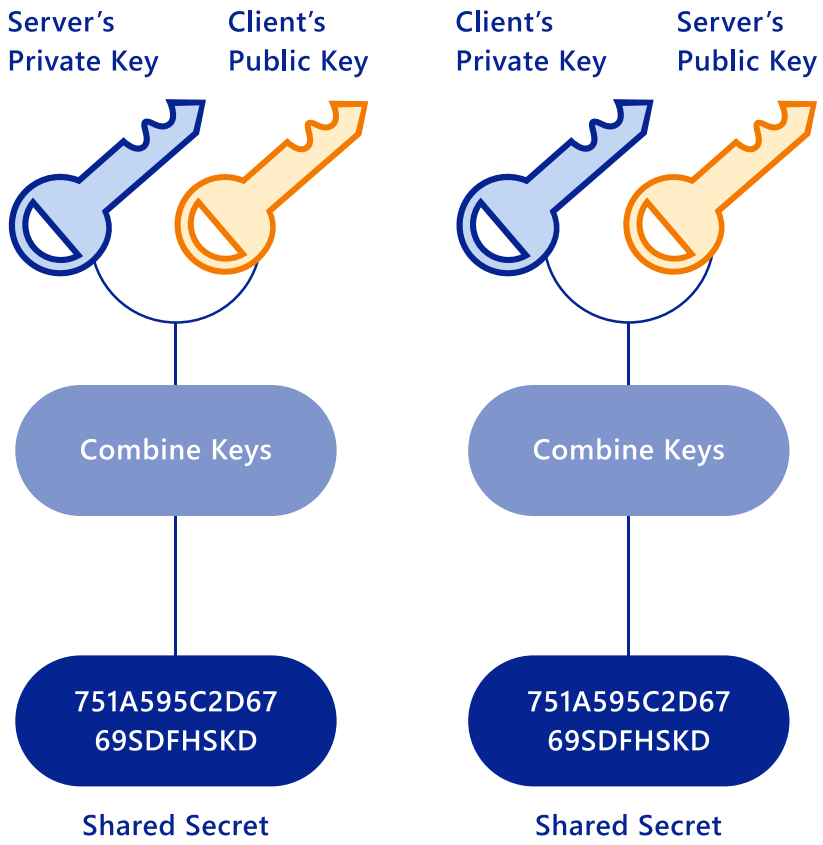
   The Diffie-Hellman algorithm makes it possible for each party to combine their own private key and the public key from the other system to make an identical secret session key.

   The key pairs used to create the session key are separate from the SSH keys used to validate and authenticate the client and server.



Learn more about the Diffie-Hellman Protocol on the next page →

How the Diffie-Hellman Protocol Works

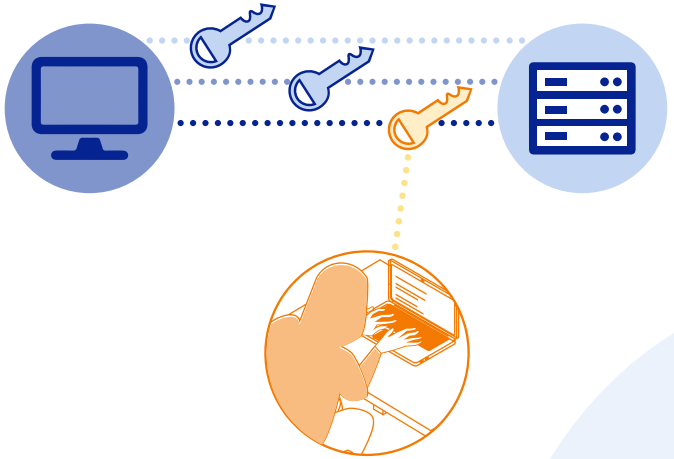


In the Diffie-Hellman key exchange scheme, the key pairs created by the parties are used to compute a shared secret offline. The shared secret is used as the key for a symmetric cipher.

Diffie-Hellman is the basis for many authenticated protocols. It provides *forward secrecy* in Transport Layer Security's ephemeral modes.

What Perfect Forward Secrecy (PFS) Means

- PFS is an encryption system that uses a different session key per transmission.
- Even if a man-in-the-middle attack occurs, the hacker only gets the information from that transmission.
- The stolen keys cannot be used to decrypt past or future transmissions.

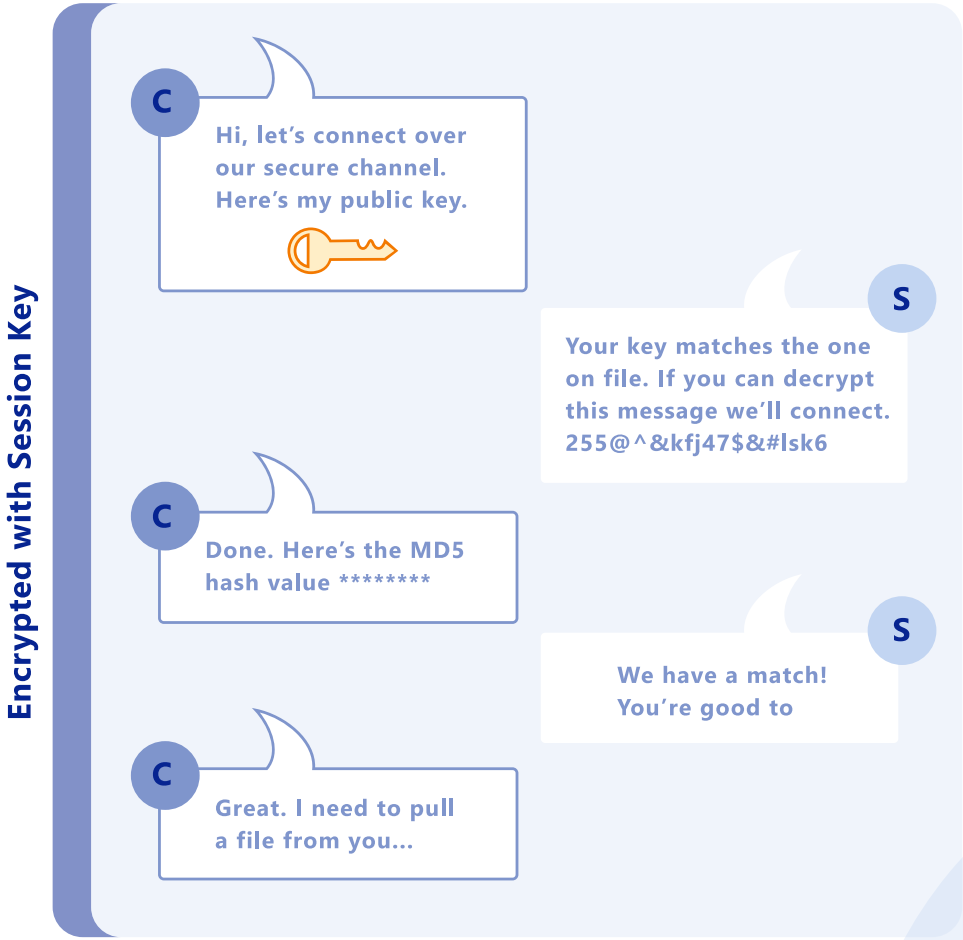


Authentication Steps (3 of 3)

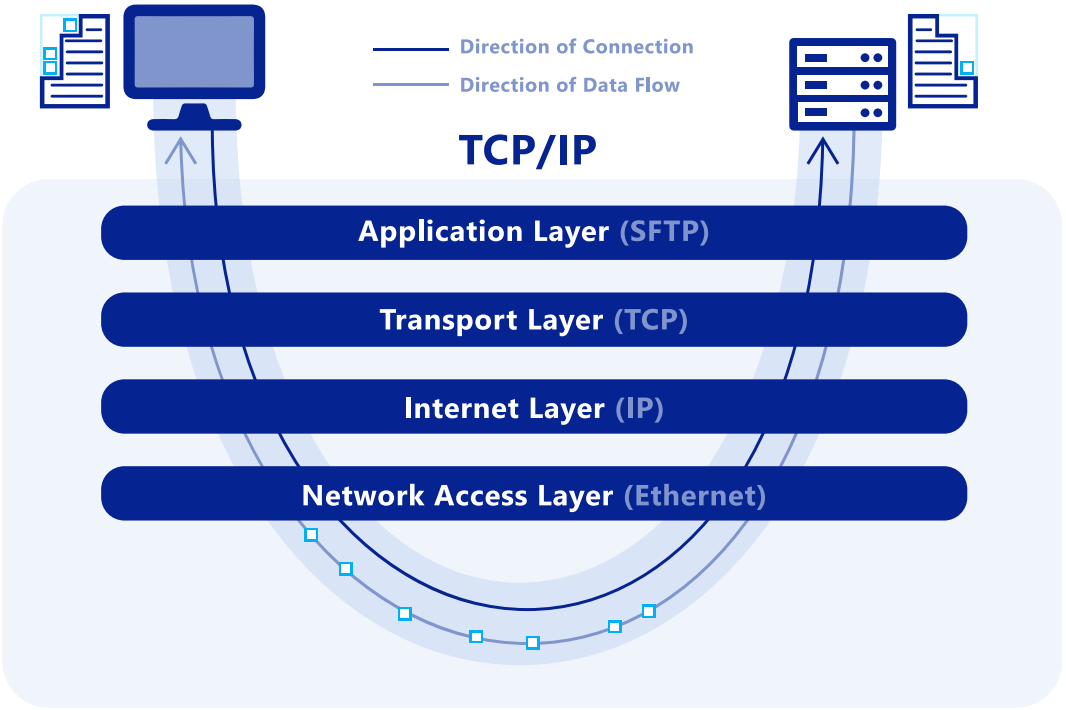
3. The server authenticates the client using an SSH key pair. This pair consists of a public key and a private key.

   Here is how that works:

   1. When the server receives a request, it compares the SSH public key to the public key it has on file.
   2. It sends an encrypted number to the client, which the client decrypts with its private key.
   3. The client combines the decrypted number with the shared session key from before to calculate the MD5 hash value. It sends that back to the server.
   4. The server also calculates the MD5 hash with the number it sent and the session key. If their answers match, the client is authenticated.



4. File Is Transferred



Finally, the file is transferred over the encrypted channel in packets.

Each packet has some of the data being transferred.

At the receiving end, the packets are put back together into the original file.

Want to know more about SFTP?
Go to our online
FAQs »

Find out how MFT adds extra security and visibility to SFTP...

Relying just on SFTP alone may be adequate for some file transfer use cases, however, sensitive, business-critical file transactions require additional enterprise-grade protection, control and visibility. SFTP servers do not have all the security features necessary for compliance with GDPR and other regulations or governance policies.

A managed file transfer (MFT) solution enhances secure file sharing by providing this required functionality. MFT is a file transfer system that utilizes multiple protocols, including SFTP, and is able to act as a client or server to enable push or pull connectivity between the MFT solution and its endpoints. Readily available capabilities of MFT include comprehensive end-to-end security; granular tracking, logging and retention settings; and high availability and disaster recovery (HADR).

Learn more about the added security and visibility of MFT »

Sources

- en.wikipedia.org/wiki/Internet_protocol_suite
- ibm.com/docs/en/ztpf/1.1.0.14?topic=concepts-public-key-cryptography
- wired.com/2016/11/what-is-perfect-forward-secrecy/
- digitalocean.com/community/tutorials/understanding-the-ssh-encryption-and-connection-process
- ipwithease.com/cisco-ssh-version-1-and-2-detailed-comparison/
- openssh.com
- ssh.com
- comparitech.com/net-admin/ssh-vs-sftp/
- maketecheasier.com/scp-vs-sftp/
- investopedia.com/terms/m/message-authentication-code.asp
- kaseya.com/blog/2021/08/10/high-availability/

Thru®