SCP (Secure Copy)

scp (secure copy) command in Linux system is used to copy file(s) between servers in a secure way. The SCP command or secure copy allows the secure transferring of files between the local host and the remote host or between two remote hosts. It uses the same authentication and security as it is used in the Secure Shell (SSH) protocol. SCP is known for its simplicity, security, and pre-installed availability.

Syntax of scp Command in Linux

scp [options] [[user@]host1:]source_file_or_directory ... [[user@]host2:]destination

In this syntax:

- options: These are various options that modify the behavior of the SCP command, such as -i for specifying an identity file, -l for limiting bandwidth, -o for specifying SSH options, -P for specifying a custom SSH port, and -S for specifying a program to use for the encrypted connection.
- [[user@]host1:]source_file_or_directory: This represents the source file or directory. It can be local or on a remote machine specified by user@host1:.
- ...: This indicates that you can specify multiple source files or directories.
- [[user@]host2:]destination: This is the destination where the files or directories will be copied. It can be local or on a remote machine specified by user@host2:

Options in scp command in Linux

Most commonly used Options in scp Command in Linux.

options	Description
-P	**port:** Specifies the port to connect on the remote host.
-p	Preserves modification times, access times, and modes from the original file.
-q	Disables the progress meter.
-r	Recursively copy entire directories.
-s	Name of program to use for the encrypted connection. The program must understand ssh(1) options.

How to Securely copy a file from a local machine to a remote machine in Linux

Syntax:

scp [file_name] remoteuser@remotehost:/remote/directory

Here,

- file_name = The name of the file that needs to be copied.
- remoteuser =The username of the remote host.
- remotehost = The IP address or hostname of the remote host.
- /remote/directory = The directory where the file should be copied on the remote machine.

Example: If we want to copy a file name “test.txt” from local system to a

“remoteuser” = “Jayesh”
“remotehost” = “10.143.90.2”
“/remote/directory” = “/home/jayesh”

Syntax:

scp test.txt jayesh@10.143.90.2:/home/jayesh

```
C:\Users\GFG0229\Desktop>scp test.txt jayesh@10.143.90.2:/home/jayesh
jayesh@10.143.90.2's password:
test.txt
```

How to Securely copy a file from remote machine to our local machine

syntax:

scp user@remotehost:/home/user/file_name

here

- “user” = username of remote system.
- “remotehost” = IP address (or hostname) of remote system.
- “/home/user/file_name” = path of file that has to be copied.
- “.” = this means that we are copying that file in current location in local system.

Example: If we have

“user” = jayesh
“remotehost” = 10.143.90.2
“home/user/file_name” = home/jayesh/test1.txt

syntax:

scp jayesh@10.143.90.2:/home/jayesh/test1.txt

```
C:\Users\GFG0229\Desktop>scp jayesh@10.143.90.2:/home/jayesh/test1.txt .
jayesh@10.143.90.2's password:
test1.txt
```

-P Option in scp Command

It is used to Securely Copy File to a Remote Machine on a Non-Standard SSH Port and specify the port to connect on the remote host. It is useful when our SSH server is listening on a non-standard port.

Syntax:

scp -P port source_file user@hostname:destination_file

Example: If we want to copy a file "test2.txt" from local machine to a remote machine with IP address "10.143.90.2" on port 2222 , user = "jayesh" and location = "/home/jayesh/".

By default, the scp uses ssh over port 22 for transferring the files. Changing the port might be necessary if the designated port 22 is not open on the remote host.

syntax:

scp -P 2222 test2.txt jayesh@10.143.90.2:/home/jayesh/

```
C:\Users\GFG0229\Desktop>scp -P 2222 test2.txt jayesh@10.143.90.2:/home/jaye
sh/
jayesh@10.143.90.2's password:
test2.txt
                                              100%    0     0.0KB/s   00:00
```

-p Option in scp Command

This option is used when we want the original metadata of the file that has been transferred. Basically, it preserves modification time, access time, and modes from the original file.

Syntax:

scp -p source_file user@hostname:destination_file

Example: If we want to copy a file “test3.txt” from local machine to a remote machine with IP address “10.143.90.2”, user = “jayesh” and location = “/home/jayesh/”

Syntax:

scp -p test3.txt jayesh@10.143.90.2:/home/jayesh/

```
C:\Users\GFG0229\Desktop>scp -p test3.txt jayesh@10.143.90.2:/home/jayesh/
jayesh@10.143.90.2's password:
test3.txt
          100%    0     0.0KB/s   00:00
```

-q Option in scp Command

It Securely Copy File with Quiet Mode – Disabling Progress Meter .This option hides the progress of the file transfer on the terminal.

Syntax:

scp -q source_file user@hostname:destination_file

Example: If we want to copy a file “test4.txt” from local machine to a remote machine with IP address “10.143.90.2”, user = “jayesh” and location = “/home/jayesh/”

Syntax:

scp -q test4.txt jayesh@10.143.90.2:/home/jayesh/

```
C:\Users\GFG0229\Desktop>scp -q test4.txt jayesh@10.143.90.2:/home/jayesh/
jayesh@10.143.90.2's password:

C:\Users\GFG0229\Desktop>
```

-r Option in scp Command

This option is used when we want to copy an entire directory and its contents. Which basically means copying entire directory recursively.

Syntax:

scp -r Directory_name user@hostname:destination_file

For Example: If we want to copy a Directory content name “new” from local machine to a remote machine with IP address “10.143.90.2”, user = “jayesh” and location = “/home/jayesh/new1/”

Syntax:

scp -r new jayesh@10.143.90.2:/home/jayesh/new1/

```
C:\Users\GFG0229\Desktop>scp -r new jayesh@10.143.90.2:/home/jayesh/new1/
jayesh@10.143.90.2's password:

C:\Users\GFG0229\Desktop>
```