**14 SCP Command Examples to Securely Copy Files in Linux**

Hello Techies, in this post, we will learn scp command in Linux with 14 different examples.

SCP is a command line tool in linux distributions, which is used to copy files and directories across the systems securely over the network. SCP stands for secure copy as it copies files using ssh protocol.

While copying,  scp command makes ssh connection to remote system. In other words, we can say scp uses the same SSH security mechanism in the backend, it needs either password or keys for authentication.

**SCP Command Syntax**

Copy from Local to Remote Host

# scp <options> <files\_or\_directories> user@target\_host:/<folder>

Copy from Remote Host to local system

# scp <options> user@target\_host:/files   <folder\_local\_system>

Some of the most widely used options in scp command are listed below,

* -C         Enable Compression
* -i           identity File or private key
* -l           limit the bandwidth while copying
* -P          ssh port number of target host
* -p          Preserves permissions, modes and access time of files while copying
* -q          Suppress warning message of SSH
* -r          Copy files and directories recursively
* -v          verbose output

Lets deep dive into scp command examples.

**1) Copy File From Local System to Remote System**

Let’s assume we want to copy jdk rpm package from our local Linux system to remote system (172.20.10.8) using scp command, run

$ scp jdk-linux-x64\_bin.rpm root@172.20.10.8:/opt

root@172.20.10.8's password:

jdk-linux-x64\_bin.rpm                          100%   10MB  27.1MB/s   00:00

$

Above command will copy jdk rpm package file to remote system under /opt folder.

**2) Copy File from Remote to Local System**

Suppose we want to copy a file from remote system to our local system under the /tmp folder, execute the following,

$ scp root@172.20.10.8:/root/Technical-Doc-RHS.odt /tmp

root@172.20.10.8's password:

Technical-Doc-RHS.odt                         100% 1109KB  31.8MB/s   00:00

$ ls -l /tmp/Technical-Doc-RHS.odt

-rwx------. 1 pkumar pkumar 1135521 Oct 19 11:12 /tmp/Technical-Doc-RHS.odt

$

**3) Verbose Output While Copying Files**

We can enable verbose output with -v option while copying files via scp command. Using verbose output we can easily find what exactly is happening in the background. This becomes very useful in debugging connection, authentication and configuration problems.

$ scp -v jdk-linux-x64\_bin.rpm root@172.20.10.8:/opt

Executing: program /usr/bin/ssh host 172.20.10.8, user root, command scp -v -t /opt

OpenSSH\_7.8p1, OpenSSL 1.1.1 FIPS  11 Sep 2018

debug1: Reading configuration data /etc/ssh/ssh\_config

debug1: Reading configuration data /etc/ssh/ssh\_config.d/05-redhat.conf

debug1: Reading configuration data /etc/crypto-policies/back-ends/openssh.config

debug1: /etc/ssh/ssh\_config.d/05-redhat.conf line 8: Applying options for \*

debug1: Connecting to 172.20.10.8 [172.20.10.8] port 22.

debug1: Connection established.

…………

debug1: Next authentication method: password

root@172.20.10.8's password:

**4) Copy Multiple Files to Remote System**

Multiple files can be copied to remote system using scp command in one go. Specify the multiple files separated by space, example is shown below

$ scp install.txt index.html jdk-linux-x64\_bin.rpm root@172.20.10.8:/mnt

root@172.20.10.8's password:

install.txt                                      100%    0     0.0KB/s   00:00

index.html                                     100%   85KB   7.2MB/s   00:00

jdk-linux-x64\_bin.rpm                            100%   10MB  25.3MB/s   00:00

[pkumar@linuxtechi ~]$

**5) Copy Files Across Two Remote Systems**

Using scp command we can copy files and directories between two remote hosts, let’s suppose we have a local Linux system which can connect to two remote Linux systems, so from my local system I can use scp command to copy files across these two systems,

Syntax:

$scp user@remote\_hosts1:/<files\_to\_transfer>  user@remote\_host2:/<folder>

Example is shown below,

$ scp pkumar@172.20.10.9:~/backup-Oct.zip root@172.20.10.8:/tmp

$ ssh root@172.20.10.8 "ls -l /tmp/backup-Oct.zip"

-rwx------. 1 root root 747438080 Oct 19 12:02 /tmp/backup-Oct.zip

**6) Copy Files and Directories Recursively**

Use -r option in scp command to recursively copy the entire directory from one system to another, example is shown below,

$ scp -r Downloads root@172.20.10.8:/opt

Use below command to verify whether Download folder is copied to remote system or not,

$ ssh root@172.20.10.8 "ls -ld /opt/Downloads"

drwxr-xr-x. 2 root root 75 Oct 19 12:10 /opt/Downloads

$

**7) Increase Copy Speed by Enabling Compression**

We can increase the transfer or copy speed by enabling the compression using -C option, it will automatically enable compression at source and decompress at destination.

$ scp -r -C Downloads root@172.20.10.8:/mnt

In the above example we are transferring the Download directory with compression enabled.

**8) Limit Bandwidth While Copying**

Use ‘-l’ option in scp command to put limit on bandwidth usage while copying. Bandwidth is specified in Kbit/s, example is shown below,

$ scp -l 500 jdk-linux-x64\_bin.rpm  root@172.20.10.8:/var

**9) Use Different SSH Port in SCP**

There can be some scenario where ssh port is changed on destination host, so while using scp command we can specify the ssh port number using ‘-P’ option.

$ scp -P 2022 jdk-linux-x64\_bin.rpm  root@172.20.10.8:/var

In above example, ssh port for remote host is “2022”

**10) Preserves Permissions, Modes and Access Time While Copying**

Use ‘-p’ option in scp command to preserve permissions, access time and modes while copying from source to destination

$ scp -p jdk-linux-x64\_bin.rpm  root@172.20.10.8:/var/tmp

jdk-linux-x64\_bin.rpm            100%   10MB  13.5MB/s   00:00

$

**11) Copying Files in Quiet Mode**

Use ‘-q’ option in scp command to suppress transfer progress, warning and diagnostic messages of ssh. Example is shown below,

$ scp -q -r Downloads root@172.20.10.8:/var/tmp

**12) Use Identify File in scp while Copying**

In most of the Linux environments, keys-based authentication is preferred. In scp command we can specify the identify file or private key file using ‘-i’ option, example is shown below,

$ scp -i my\_key.pem -r Downloads root@172.20.10.8:/root

In above example, “my\_key.pem” is the identity file or private key file.

**13) Use Different ssh\_config File in scp**

There are some scenarios where you use different networks to connect to Linux systems, may be some network is behind proxy servers, so in that case we must have different **ssh\_config** file.

Different ssh\_config file in scp command is specified via ‘-F’ option, example is shown below

$ scp -F /home/pkumar/new\_ssh\_config -r Downloads \

root@172.20.10.8:/root

root@172.20.10.8's password:

jdk-linux-x64\_bin.rpm                     100%   10MB  16.6MB/s   00:00

backup-Oct.zip                            100%  713MB  41.9MB/s   00:17

index.html                                100%   85KB   6.6MB/s   00:00

$ww

**14) Use Different Cipher in scp Command**

By default, scp uses ‘AES-128’ cipher to encrypt the files. If you want to use another cipher in scp command then use ‘-c’ option followed by cipher name,

Let’s suppose we want to use ‘3des-cbc’ cipher in scp command while transferring the files, run the following scp command

$ scp -c 3des-cbc -r Downloads root@172.20.10.8:/root

Use the below command to list ssh and scp ciphers,

$ ssh -Q cipher localhost | paste -d , -s -

3des-cbc,aes128-cbc,aes192-cbc,aes256-cbc,rijndael-cbc@lysator.liu.se,\

aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com,\

chacha20-poly1305@openssh.com

$

That’s all from this post, to get more details about scp command, kindly refer its man page. Please do share your feedback and comments in comments section below.

Also Read : [How to Setup Passwordless SSH Login in Linux with Keys](https://www.linuxtechi.com/passwordless-ssh-login-keys-linux/)