NAME

scp — secure copy (remote file copy program)

SYNOPSIS

scp [**-346BCpqrTv**] [**-c** *cipher*] [**-F** *ssh_config*] [**-i** *identity_file*] [**-l** *limit*]
[**-o** *ssh_option*] [**-P** *port*] [**-S** *program*] [[*user@*]*host1*:]*file1* ...
[[*user@*]*host2*:]*file2*

DESCRIPTION

scp copies files between hosts on a network. It uses ssh(1) for data transfer, and uses the same authentication and provides the same security as ssh(1). **scp** will ask for passwords or passphrases if they are needed for authentication.

File names may contain a user and host specification to indicate that the file is to be copied to/from that host. Local file names can be made explicit using absolute or relative pathnames to avoid **scp** treating file names containing ':' as host specifiers. Copies between two remote hosts are also permitted.

The options are as follows:

-3 Copies between two remote hosts are transferred through the local host. Without this option the data is copied directly between the two remote hosts. Note that this option disables the progress meter.

-4 Forces **scp** to use IPv4 addresses only.

-6 Forces **scp** to use IPv6 addresses only.

-B Selects batch mode (prevents asking for passwords or passphrases).

-C Compression enable. Passes the **-C** flag to ssh(1) to enable compression.

-c *cipher*
Selects the cipher to use for encrypting the data transfer. This option is directly passed to ssh(1).

-F *ssh_config*
Specifies an alternative per-user configuration file for **ssh**. This option is directly passed to ssh(1).

-i *identity_file*
Selects the file from which the identity (private key) for public key authentication is read. This option is directly passed to ssh(1).

-l *limit*
Limits the used bandwidth, specified in Kbit/s.

-o *ssh_option*
Can be used to pass options to **ssh** in the format used in ssh_config(5). This is useful for specifying options for which there is no separate **scp** command-line flag. For full details of the options listed below, and their possible values, see ssh_config(5).

AddressFamily
BatchMode
BindAddress
CanonicalDomains
CanonicalizeFallbackLocal
CanonicalizeHostname
CanonicalizeMaxDots

CanonicalizePermittedCNAMEs
CertificateFile
ChallengeResponseAuthentication
CheckHostIP
Ciphers
Compression
ConnectionAttempts
ConnectTimeout
ControlMaster
ControlPath
ControlPersist
GlobalKnownHostsFile
GSSAPIAuthentication
GSSAPIDelegateCredentials
HashKnownHosts
Host
HostbasedAuthentication
HostbasedKeyTypes
HostKeyAlgorithms
HostKeyAlias
HostName
IdentitiesOnly
IdentityAgent
IdentityFile
IPQoS
KbdInteractiveAuthentication
KbdInteractiveDevices
KexAlgorithms
LogLevel
MACs
NoHostAuthenticationForLocalhost
NumberOfPasswordPrompts
PasswordAuthentication
PKCS11Provider
Port
PreferredAuthentications
ProxyCommand
ProxyJump
PubkeyAcceptedKeyTypes
PubkeyAuthentication
RekeyLimit
SendEnv
ServerAliveInterval
ServerAliveCountMax
StrictHostKeyChecking
TCPKeepAlive
UpdateHostKeys
UsePrivilegedPort
User

UserKnownHostsFile
VerifyHostKeyDNS

-P *port*
Specifies the port to connect to on the remote host. Note that this option is written with a capital 'P', because **-p** is already reserved for preserving the times and modes of the file.

-p Preserves modification times, access times, and modes from the original file.

-q Quiet mode: disables the progress meter as well as warning and diagnostic messages from ssh(1).

-r Recursively copy entire directories. Note that **scp** follows symbolic links encountered in the tree traversal.

-S *program*
Name of *program* to use for the encrypted connection. The program must understand ssh(1) options.

-T Disable strict filename checking. By default when copying files from a remote host to a local directory **scp** checks that the received filenames match those requested on the command-line to prevent the remote end from sending unexpected or unwanted files. Because of differences in how various operating systems and shells interpret filename wildcards, these checks may cause wanted files to be rejected. This option disables these checks at the expense of fully trusting that the server will not send unexpected filenames.

-v Verbose mode. Causes **scp** and ssh(1) to print debugging messages about their progress. This is helpful in debugging connection, authentication, and configuration problems.

EXIT STATUS

The **scp** utility exits 0 on success, and >0 if an error occurs.

SEE ALSO

sftp(1), ssh(1), ssh-add(1), ssh-agent(1), ssh-keygen(1), ssh_config(5), sshd(8)

HISTORY

scp is based on the rcp program in BSD source code from the Regents of the University of California.

AUTHORS

Timo Rinne <tri@iki.fi>
Tatu Ylonen <ylo@cs.hut.fi>