smbclient SMB share client

CONCEPTS

> Remote commands act on the share; lcd and ! act on your local machine. Quote paths with spaces.

smbclient -L find shares
↓ pick share
//host/share authenticate
↓ smb: \>
Prompt ls, get, put

> mget and mput ask per file and skip subdirectories until you toggle prompt and recurse.

CONNECT

`smbclient -L <host>`
List shares on a server

`smbclient -L <host> -N`
List shares anonymously

`smbclient //<host>/<share>`
Open a share interactively

`smbclient //<host>/<share> -U <user>`
Connect as a user

`smbclient //<host>/<share> -U <domain>\\<user>`
Connect as a domain user

`smbclient //<host>/<share> -N`
Connect without a password

`smbclient //<host>/<share> -A <file>`
Read credentials from a file

`smbclient //<host>/<share> -I <ip>`
Connect by IP address

`smbclient //<host>/<share> -p <port>`
Use another TCP port

`smbclient //<host>/<share> -m SMB3`
Cap the protocol version

`smbclient //<host>/<share> --use-kerberos=required`
Authenticate with Kerberos

`smbclient //<host>/<share> --client-protection=encrypt`
Require encryption

`smbclient -L <host> -g`
Machine-readable share list

NAVIGATE

`ls [<mask>]`
List remote files

`cd <dir>`
Change remote directory

`lcd <dir>`
Change local directory

`du <dir>`
List with disk usage and free space

`! <command>`
Run a local shell command

`more <file>`
Page a remote file

`help [<command>]`
Show command help

`exit`
End the session

TRANSFER

`get <remote> [<local>]`
Download a file

`put <local> [<remote>]`
Upload a file

`mget <mask>`
Download matching files

`mput <mask>`
Upload matching files

`prompt`
Toggle per-file confirmation

`recurse`
Toggle recursion into subdirectories

`mask <mask>`
Filter for recursive transfers

`scopy <src> <dst>`
Copy on the server

MANAGE FILES

`mkdir <dir>`
Create a remote directory

`rmdir <dir>`
Remove a remote directory

`del <mask>`
Delete files

`deltree <mask>`
Delete recursively

`rename <old> <new>`
Rename or move a file

MANAGE FILES (cont.)

`allinfo <file>`
Show all file information

SCRIPTING

`smbclient //<host>/<share> -c "<cmds>"`
Run commands, then exit

`smbclient //<host>/<share> -U <user>%<pass>`
Give the password inline

`smbclient //<host>/<share> -d <level>`
Set debug level 0-10

`smbclient //<host>/<share> --option="<name>=<value>"`
Override an smb.conf option

TAR BACKUP

`smbclient //<host>/<share> -Tc <file>.tar`
Back up the share to a tar

`smbclient //<host>/<share> -Tx <file>.tar`
Restore a tar to the share

`smbclient //<host>/<share> -Tc <file>.tar <path>`
Back up only a path

RECIPES

Download a whole directory

```
smbclient //<host>/<share> -U
<user>
smb: \> cd <dir>
smb: \> recurse ON
smb: \> prompt OFF
smb: \> mget *
```

One-shot upload

```
smbclient //<host>/<share> -A
~/.smbcred \
  -c "cd <dir>; put report.pdf"
```

Back up a share to a local archive

```
smbclient //<host>/<share> -A
~/.smbcred \
  -Tc backup.tar
```

Find shares, then mount one

```
smbclient -L <host> -U <user>
sudo mount -t cifs //<host>/
<share> /mnt \
  -o username=<user>
```

HELP

`smbclient --help`
Option summary