sshfs Mount over SSH

Your programs ls, cp, editor → FUSE → **sshfs** local mountpoint → ssh + sftp → **sftp-server** on the remote host → disk → **Remote dir** /srv/data

CONCEPTS

sshfs speaks SFTP over a normal ssh login: if ssh <host> works, sshfs <host>: <dir> works. Run it as your own user, not root.

MOUNT AND UNMOUNT

sshfs *<user>*@*<host>*:*<dir>* *<mountpoint>*
Mount a remote directory

sshfs *<host>*: *<mountpoint>*
Mount the remote home directory

fusermount3 -u *<mountpoint>*
Unmount (Linux)

umount *<mountpoint>*
Unmount on macOS and BSD

sshfs -p *<port>* *<host>*: *<mountpoint>*
Use a non-default SSH port

sshfs -o IdentityFile=*<key>* *<host>*: *<mountpoint>*
Pick the key to log in with

MOUNT AND UNMOUNT (cont.)

sshfs -F *<ssh_config>* *<host>*: *<mountpoint>*
Use another ssh config file

sshfs -o ro *<host>*: *<mountpoint>*
Mount read-only

sshfs -C *<host>*: *<mountpoint>*
Compress traffic (slow links)

findmnt -t fuse.sshfs
List active sshfs mounts

sshfs -f *<host>*: *<mountpoint>*
Stay in the foreground

sshfs -d *<host>*: *<mountpoint>*
Debug output (implies -f)

sshfs -o delay_connect *<host>*: *<mountpoint>*
Connect on first access

fusermount3 -uz *<mountpoint>*
Lazy unmount a busy mount

RECONNECT AND HANGS

sshfs -o reconnect,ServerAliveInterval=15 *<host>*: *<mountpoint>*
Recover after the link drops

RECONNECT AND HANGS (cont.)

sshfs -o ServerAliveInterval=15,ServerAliveCountMax=3 *<host>*: *<mountpoint>*
Drop a dead connection after ~45 s

pkill -f 'sshfs.*<mountpoint>'
Kill a hung sshfs, freeing stuck programs

A frozen mount (ls hangs, cd stalls) usually means a dead connection with no keepalive: set ServerAliveInterval together with reconnect.

PERMISSIONS AND SYMLINKS

sshfs -o allow_other *<host>*: *<mountpoint>*
Let other local users use the mount

sshfs -o allow_other,default_permissions *<host>*: *<mountpoint>*
Enforce file modes locally too

sshfs -o idmap=user *<host>*: *<mountpoint>*
Map remote user's IDs to yours

PERMISSIONS AND SYMLINKS (cont.)

sshfs -o uid=*<n>***,gid=***<n>* *<host>*: *<mountpoint>*
Show files as owned by this uid/gid

sshfs -o follow_symlinks *<host>*: *<mountpoint>*
Show remote symlinks as real files

SPEED AND CACHING

sshfs -o dcache_timeout=*<s>* *<host>*: *<mountpoint>*
Cache directory data longer (default 20 s)

sshfs -o dir_cache=no *<host>*: *<mountpoint>*
Always ask the server for fresh listings

sshfs -o auto_cache *<host>*: *<mountpoint>*
Reuse cached files until mtime changes

sshfs -o max_conns=*<n>* *<host>*: *<mountpoint>*
Use several ssh connections

sshfs -o workaround=rename *<host>*: *<mountpoint>*
Allow mv over an existing file

SPEED AND CACHING (cont.)

Chatty workloads (git status, builds, find) are slow over sshfs: raise dcache_timeout, or work on the remote host directly.

MOUNTING AT BOOT

mkdir -p *<mountpoint>*
Create the mountpoint first

fstab line

```
<user>@<host>:<dir>
<mountpoint> fuse.sshfs \
  _netdev,reconnect,
ServerAliveInterval=15,
IdentityFile=<key> 0 0
# one line in the real file
```

fstab mounts run as root, so give an absolute IdentityFile path that root can read, and accept the host key once as root first.

RECIPES

Mount a server for the day

```
mkdir -p ~/mnt/box
sshfs -o reconnect,
ServerAliveInterval=15
<user>@<host>:/srv/data ~/
mnt/box
ls ~/mnt/box
fusermount3 -u ~/mnt/box
```