

# SSH Cheatsheet by Richard Snider

codetriangle.me

## Invoke SSH

```
ssh [user@]hostname [-p port] (open interactive shell)
ssh [user@]hostname [-p port] command (run one command)
```

## Transfer files

```
scp [user@]hostname:remote-path local-path (client to server)
scp local-path [user@]hostname:remote-path (server to client)
sftp [user@]hostname (interactive interface)
sshfs [user@]hostname:path ./remote-files (mount locally)
```

## SSH keys

valid public keys are stored on the server in ~/.ssh/authorized\_keys

```
ssh-keygen -t ed25519 (generate new key)
ssh-keygen -p -f ~/.ssh/id_XXXXX (change passphrase)
ssh-copy-id -i ~/.ssh/id_XXXXX.pub [user@]hostname (upload key)
```

## SSH agent

default keys are id\_rsa, id\_ecdsa, and id\_ed25519

```
eval $(ssh-agent -s) (activate in shell)
eval $(ssh-agent -s -t life) (activate with specific lifetime)
eval $(ssh-agent -s -k) (kill instance in shell)
ssh-agent command (launch alongside command)
```

```
ssh-add [-t life] (adds all default keys)
ssh-add [-t life] ~/.ssh/id_XXXXX (adds specific key)
```

```
ssh-add -D (removes all default keys)
ssh-add -d ~/.ssh/id_XXXXX (removes specific keys)
```

## TMUX

tmux keybindings start with Ctrl-B; type Ctrl-B ? for help

```
tmux (start new session)
tmux [a|attach] (connect to existing session)
```

## Port Forwarding

```
ssh -L8080:remote-hostname:5000 (forward traffic from local to remote)
ssh -R5000:local-hostname:8080 (forward traffic from remote to local)
ssh -D1337 (enable SOCKS proxy on local port 1337)
```

## Use SSH as git server

```
git clone [user@]hostname:path/to/repo (clone from copy on server)
git remote add custom [user@]hostname:path (add to existing repo)
```

## Escape sequences

type ~ after a newline to begin an escape sequence

```
~. disconnect from server ~? list escape sequences
~# list forwarded connections ~Ctrl-Z background SSH
~C add forwarded connection
```

## Configuration

config file is in ~/.ssh/config

```
Host * # matches all hosts
  AddKeysToAgent yes # add keys to agent without ssh-add
  ControlMaster auto # enable SSH multiplexing
  ControlPath ~/.ssh/cm-%r@%h:%p # filename format to manage connections
  ControlPersist 30m # close connections after 30 minutes

Host nickname # match a specific hostname
  Hostname a.long-domain-name.net # "nickname" is now a short alias of this
  User fsjc # log in as this user for this host
```

# SSH Cheatsheet by Richard Snider

codetriangle.me

## Invoke SSH

```
ssh [user@]hostname [-p port] (open interactive shell)
ssh [user@]hostname [-p port] command (run one command)
```

## Transfer files

```
scp [user@]hostname:remote-path local-path (client to server)
scp local-path [user@]hostname:remote-path (server to client)
sftp [user@]hostname (interactive interface)
sshfs [user@]hostname:path ./remote-files (mount locally)
```

## SSH keys

valid public keys are stored on the server in ~/.ssh/authorized\_keys

```
ssh-keygen -t ed25519 (generate new key)
ssh-keygen -p -f ~/.ssh/id_XXXXX (change passphrase)
ssh-copy-id -i ~/.ssh/id_XXXXX.pub [user@]hostname (upload key)
```

## SSH agent

default keys are id\_rsa, id\_ecdsa, and id\_ed25519

```
eval $(ssh-agent -s) (activate in shell)
eval $(ssh-agent -s -t life) (activate with specific lifetime)
eval $(ssh-agent -s -k) (kill instance in shell)
ssh-agent command (launch alongside command)
```

```
ssh-add [-t life] (adds all default keys)
ssh-add [-t life] ~/.ssh/id_XXXXX (adds specific key)
```

```
ssh-add -D (removes all default keys)
ssh-add -d ~/.ssh/id_XXXXX (removes specific keys)
```

## TMUX

tmux keybindings start with Ctrl-B; type Ctrl-B ? for help

```
tmux (start new session)
tmux [a|attach] (connect to existing session)
```

## Port Forwarding

```
ssh -L8080:remote-hostname:5000 (forward traffic from local to remote)
ssh -R5000:local-hostname:8080 (forward traffic from remote to local)
ssh -D1337 (enable SOCKS proxy on local port 1337)
```

## Use SSH as git server

```
git clone [user@]hostname:path/to/repo (clone from copy on server)
git remote add custom [user@]hostname:path (add to existing repo)
```

## Escape sequences

type ~ after a newline to begin an escape sequence

```
~. disconnect from server ~? list escape sequences
~# list forwarded connections ~Ctrl-Z background SSH
~C add forwarded connection
```

## Configuration

config file is in ~/.ssh/config

```
Host * # matches all hosts
  AddKeysToAgent yes # add keys to agent without ssh-add
  ControlMaster auto # enable SSH multiplexing
  ControlPath ~/.ssh/cm-%r@%h:%p # filename format to manage connections
  ControlPersist 30m # close connections after 30 minutes

Host nickname # match a specific hostname
  Hostname a.long-domain-name.net # "nickname" is now a short alias of this
  User fsjc # log in as this user for this host
```