

### SSH keygen

ssh-keygen -t algo -b bit -f <filename>	Generate ssh-key
rsa 4096, ecdsa 521, ed25519	Algorithm -t
ssh-copy-id -i (keyfile) user@host	Copy file to host
ssh-keygen -e -f ~/id_dsa > ~/id_dsa_com.pub	Convert sshkey to OpenSSH format
ssh-add -l	List ssh-key in ssh-Agent
ssh-add /file	Add key to ssh-Agent
ssh-add -d file	Removes key from Agent
ssh-add -D	Removes all keys from Agent
ssh-add -x	lock the ssh-Agent
ssh-add -X	unlocks the ssh-Agent
ssh-add -t 60 /key	key will auto deletet in 60sec
ssh -N	Do not execute a remote command

### SSH Login

ssh user@host -p (Port)	Standard Login SSH
ssh user@host -p (Port) -i /key	Login with key
ssh -L port:localhost:port user@server	Local Forwarding
ssh -R port:localhost:port user@server	Remote Forwarding
ssh -X user@host	X11 Forwarding

. Let's say you're running PostgreSQL on your server, which by default listens on the port 5432.

```
$ ssh -L 9000:localhost:5432 user@example.com
```

The part that changed here is the localhost:5432, which says to forward connections from your local port 9000 to localhost:5432 on your server. Now we can simply connect to our database.

### scp

scp /path user@host:/path	copy from locale to remothost
scp user@host:/path /path	copy from remothost to locale
scp user1@server1:/path/ user2@server2:/path	copy file on the remote server to another remote server

### scp (cont)

scp jane@host:/home/jane/table.csv	copy file remote host to the same
pete@host:/home/pete/	remote host in another location
scp /path/*.ext user@server:/path/	Copy all files of a specific type
scp [-12346BCpqrv] [-c cipher] [-F ssh_config] [-i identity_file] [-l limit] [-o ssh_option] [-P port] [-S program] [[user@]host1:]file1 ... [[user@]host2:]file2	

### wget

-nv	no-verbose
-v	verbose
-q	quiet
-t X	-tries=X
-P	/path
-O	Outout
-nd	no-directories
-m	mirror
-A	ACCLIST
-R	REJLIST
-e robots=off	ignore robots.txt
wget http://url/file	Downloading File
wget -P /home/ http://url	Downloading to specific folder
wget -r http://url.com/dir/	Downloading Files recursively
wget -N http://url/file	Downloading only if file is newer
wget -r -A jpg,png http://url.com/dir/	Downloading Files recursively only jpg,png files
wget -r -R jpg,png http://url.com/dir/	Downloading Files recursively everthing except jpg,png
wget -c http://url/file	Continue stopped download
wget -O filename http://url/FILE	Saves download under other name
wget -m http://url	Mirror website
wget -m -A '*.pdf' -nd -e robots=off http://url	Download all PDF file from webseite
wget --ftp-user=USERNAME --ftp-password=PASSWORD ftp://ftp.url.com/filename.tar.gz	Download via Ftp



### curl

-O	saves file as the originalname
-C -	continue where it stopped
-- output	indivi. filename
-v	verbose

curl <http://www.example.com/readme.txt> Shows file in Terminal (not downloading)

curl -O <ftp://ftp.test.de/test/test.iso> Download file and saves as filename

curl -C - -O <ftp://ftp.test.de/test/test.iso> Starts downloading where it stopped

curl <ftp://url> --user myname:mypassword Download with Userpassword



By **BlackW61**  
[cheatography.com/blackw61/](https://cheatography.com/blackw61/)

Not published yet.  
Last updated 9th July, 2018.  
Page 2 of 2.

Sponsored by **Readability-Score.com**  
Measure your website readability!  
<https://readability-score.com>