



ritlug.com

Linux: A Command Line Perspective!

September 2021

Keep up with RITLUG outside of meetings:

ritlug.com/get-involved, discord.ritlug.com

Command Rapid Fire



pwd

“Present Working Directory” Tells you the absolute path of the folder you are “in”

```
[jeff@LAPTOP-CVJ54CBV] - [~/dot_files/docker] - [2020-08-26 07:46:21]
[0] <git:(master 3d9584a) > pwd
/home/jeff/dot_files/docker
```

ls

“List” Tells you what is in your current directory.

Popular arguments:

- a all files(including hidden)
 - l list view rather than grid
- also shows you permissioning and ownership info

```
jeff@LAPTOP-CVJ54CBV:~/dot_files/docker$ ls
gitea  jupyterlab
jeff@LAPTOP-CVJ54CBV:~/dot_files/docker$ ls -la
total 16
drwxr-xr-x 4 jeff jeff 4096 Aug 26 19:50 .
drwxr-xr-x 4 jeff jeff 4096 Aug 21 12:04 ..
-rw-r--r-- 1 jeff jeff    0 Aug 26 19:50 .hidden_spooky_file
drwxr-xr-x 2 jeff jeff 4096 Aug 21 12:04 gitea
drwxr-xr-x 3 jeff jeff 4096 Aug 21 12:04 jupyterlab
jeff@LAPTOP-CVJ54CBV:~/dot_files/docker$
```

mv

“moves” Moves files. AKA: only way to rename a file

-i flag to avoid accidental overwrites

```
[jeff LAPTOP-CVJ54CBV] - [~/dot_files/docker] - [2020-08-26 07:52:19]
[0] <git:(master 3d9584a) > ls
./ ../ .hidden_spooky_file gitea/ jupyterlab/
[jeff LAPTOP-CVJ54CBV] - [~/dot_files/docker] - [2020-08-26 07:52:20]
[0] <git:(master 3d9584a) > mv .hidden_spooky_file ../.hidden_spooky_file
[jeff LAPTOP-CVJ54CBV] - [~/dot_files/docker] - [2020-08-26 07:52:44]
[0] <git:(master 3d9584a) > cd ..
./      .git/      .ideavimrc  README.md  fixSSHfilePermissions.sh
../     .gitconfig .vimrc      backup.sh  install.sh
.bashrc .hidden_spooky_file .zshrc      docker/    renew_certs.sh
[jeff LAPTOP-CVJ54CBV] - [~/dot_files] - [2020-08-26 07:52:49]
[0] <git:(master 3d9584a) >
```

cp

“Copy” Copy file/folder

-i flag to avoid accidental
overwrites

```
[jeff LAPTOP-CVJ54CBV] - [~] - [2020-08-26 07:54:30]
[0] <> cd temp
./ ../
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:54:31]
[0] <> touch empty1.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:54:39]
[0] <> ls
./ ../ empty1.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:54:47]
[0] <> cp empty1.txt tempty2.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:55:00]
[0] <> ls -la
total 8
drwxr-xr-x  2 jeff jeff 4096 Aug 26 19:55 ./
drwxr-xr-x 15 jeff jeff 4096 Aug 26 19:55 ../
-rw-r--r--  1 jeff jeff   0 Aug 26 19:54 empty1.txt
-rw-r--r--  1 jeff jeff   0 Aug 26 19:55 tempty2.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:55:02]
[0] <>
```

chmod

```
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:55:02]
[0] <> man chmod
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 08:04:07]
[0] <> ls -la
total 8
drwxr-xr-x  2 jeff jeff 4096 Aug 26 19:55 ./
drwxr-xr-x 15 jeff jeff 4096 Aug 26 20:04 ../
-rw-r--r--  1 jeff jeff   0 Aug 26 19:54 empty1.txt
-rw-r--r--  1 jeff jeff   0 Aug 26 19:55 empty2.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 08:04:11]
[0] <> chmod 700 empty1.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 08:04:29]
[0] <> ls -la
total 8
drwxr-xr-x  2 jeff jeff 4096 Aug 26 19:55 ./
drwxr-xr-x 15 jeff jeff 4096 Aug 26 20:04 ../
-rwx----- 1 jeff jeff   0 Aug 26 19:54 empty1.txt
-rw-r--r--  1 jeff jeff   0 Aug 26 19:55 empty2.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 08:04:32]
[0] <> chmod 777 empty1.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 08:04:45]
[0] <> ls -la
total 8
drwxr-xr-x  2 jeff jeff 4096 Aug 26 19:55 ./
drwxr-xr-x 15 jeff jeff 4096 Aug 26 20:04 ../
-rwxrwxrwx  1 jeff jeff   0 Aug 26 19:54 empty1.txt
-rw-r--r--  1 jeff jeff   0 Aug 26 19:55 empty2.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 08:04:47]
[0] <>
```

“Changes File Mode”

Allows you to change to read/write/execute permissions on your file.

Format: `chmod mode file_name`

Popular modes:

- R recursive
- +x executable (for single file)
- 644
- 666
- 755

Mode Format:

- user:group:other
- 4 = read
- 2 = write
- 1 = execute
- 0 = no permissions

chown

“Change file Owner and Group”

Command format:

`chown user:group filename`

Popular Flags:

`-R` recursive

```
root@LAPTOP-CVJ54CBV:/home/jeff/temp# ls -la
total 8
drwxr-xr-x  2 jeff jeff 4096 Aug 26 19:55 .
drwxr-xr-x 15 jeff jeff 4096 Aug 26 20:08 ..
-rwxrwxrwx  1 jeff jeff   0 Aug 26 19:54 empty1.txt
-rw-r--r--  1 jeff jeff   0 Aug 26 19:55 tempty2.txt
root@LAPTOP-CVJ54CBV:/home/jeff/temp# chown root:jeff empty1.txt
root@LAPTOP-CVJ54CBV:/home/jeff/temp# ls -la
total 8
drwxr-xr-x  2 jeff jeff 4096 Aug 26 19:55 .
drwxr-xr-x 15 jeff jeff 4096 Aug 26 20:08 ..
-rwxrwxrwx  1 root jeff   0 Aug 26 19:54 empty1.txt
-rw-r--r--  1 jeff jeff   0 Aug 26 19:55 tempty2.txt
root@LAPTOP-CVJ54CBV:/home/jeff/temp# chown root:root empty1.txt
root@LAPTOP-CVJ54CBV:/home/jeff/temp# ls -la
total 8
drwxr-xr-x  2 jeff jeff 4096 Aug 26 19:55 .
drwxr-xr-x 15 jeff jeff 4096 Aug 26 20:08 ..
-rwxrwxrwx  1 root root   0 Aug 26 19:54 empty1.txt
-rw-r--r--  1 jeff jeff   0 Aug 26 19:55 tempty2.txt
root@LAPTOP-CVJ54CBV:/home/jeff/temp#
```


cat

“conCATenates” file and prints to STDOUT

Command format:

cat filename

```
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 08:10:59]
[0] <> vim empty1.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 08:11:24]
[0] <> cat empty1.txt
No longer an empty file.

Cat!

[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 08:11:27]
[0] <>
```

more, less

Files that enable you to page through output of a large files/logs/commands

Command format:

more/less **filename**

More:

“Basic”

Less:

More, but, you can also do things like go backwards.

```
# Alias for markdown paragraph formatter
alias pp="python3 /home/jeff/scripts/RandomScripts/other/markd

# Alias for moving to common directories
alias blog="cd /home/jeff/Documents/git/NodeJSblog/blogContent

# auto ls when you change directories
function cd {
    builtin cd "$@" && ls
}

# Function that removes all files with a specific extension.
function begone
{
    if [ $# -eq 1 ]
--More--(83%)
```

grep

Allows you to search for text in files via text matching or by using regular expressions. If it matches your pattern, it will print out that line in the folder

Command format:

```
grep flags search_pattern file/folder
```

Popular Flags:

- n print line number
- r recursive -- useful when searching entire folder

```
[jeff LAPTOP-CVJ54CBV] - [~] - [2020-08-26 08:31:24]
[0] > grep -nr "roosay" scripts/
Binary file scripts/bash_manager/src/__pycache__/configuration
scripts/bash_manager/src/configuration.py:141: alias
scripts/bash_manager/src/.zshrc:90:alias roosay="python3
scripts/bash_manager/README.md:18:roosay is a simple pyth
scripts/bash_manager/README.md:22:python3 roosay.py Super
scripts/bash_manager/README.md:26:roosay super cool messa
[jeff LAPTOP-CVJ54CBV] - [~] - [2020-08-26 08:31:46]
[0] <>
```

top, htop

```
last pid: 21417; load averages: 0.35, 0.31, 0.26
65 processes: 1 running, 64 sleeping
CPU: 1.3% user, 0.0% nice, 0.8% system, 0.0% idle
Mem: 374M Active, 1913M Inact, 28G Wired, 887M Buff, 1.28:1 MemFree
ARC: 23G Total, 2812M MFU, 20G MRU, 235K Anon, 21G Compressed, 27G Uncompressed, 1.28:1 MemFree
Swap: 8192M Total, 8192M Free
```

PID	USER	PRI	NI	VIRT	RES	S	CPU%	MEM%	TIME+	Command
3141	root	22	0	161M	147M	S	0.2	0.5	0:11.44	/usr/local/sbin/smbd --daemon --configfile=/usr/local/etc/smbd.conf
21425	jeff	20	0	9232	4392	R	0.0	0.0	0:00.01	htop
21391	jeff	21	0	8936	5948	S	0.0	0.0	0:00.04	-zsh
21390	jeff	20	0	12916	8072	S	0.0	0.0	0:00.00	sshd: jeff@pts/0
21388	root	21	0	12916	8036	S	0.0	0.0	0:00.01	sshd: jeff [priv]
1629	public	20	0	10164	6124	S	0.0	0.0	0:00.56	/usr/libexec/sftp-server -l ERROR -f AUTH
1629	public	20	0	12916	8224	S	0.0	0.0	0:00.75	sshd: public@notty
64691	989	63	20	788M	701M	uw	0.0	0.0	0:00.01	sshd: public [priv]
21417	jeff	1	21	7940K	4072K	CP	0.0	0.0	0:00.62	/usr/libexec/sftp-server -l ERROR -f AUTH
2267	nobody	1	20	6928K	3096K	se	0.0	0.0	0:01.03	sshd: jeff@notty
54248	nobody	1	20	6928K	3016K	se	0.0	0.0	0:00.01	sshd: jeff [priv]
53137	root	1	20	29404K	18456K	se	0.0	0.0	0:00.30	/usr/libexec/sftp-server -l ERROR -f AUTH
231	root	21	20	315M	258M	kq	0.0	0.0	0:00.53	sshd: jeff@notty
5079	root	6	20	122M	42276K	se	0.0	0.0	0:00.01	sshd: jeff [priv]
1629	public	1	20	12916K	8224K	se	0.0	0.0	0:00.20	nginx: worker process
1621	jeff	1	20	12916K	8268K	se	0.0	0.0	0:00.71	/usr/sbin/cron -J 15 -s
3158	root	8	20	33600K	11612K	select	14	4:41	0.00%	rrdcached
21390	jeff	1	20	12916K	8052K	select	13	0:00	0.00%	sshd
2936	root	1	20	12488K	12596K	select	8	0:18	0.00%	ntpd
3325	root	1	20	115M	100M	kqread	7	0:17	0.00%	uwsgi-3.6
53147	root	1	20	37376K	23424K	select	9	0:04	0.00%	winbindd
3532	root	11	20	148M	104M	nanslp	4	28:33	0.00%	collectd
340	root	2	20	122M	106M	piperd	12	1:32	0.00%	python3.6
344	root	2	20	122M	107M	usem	9	1:30	0.00%	python3.6
343	root	2	20	122M	107M	usem	6	1:29	0.00%	python3.6
341	root	2	20	122M	107M	usem	14	1:29	0.00%	python3.6
342	root	2	20	122M	107M	usem	11	1:28	0.00%	python3.6
3475	root	15	20	178M	140M	umtxn	9	0:21	0.00%	uwsgi-3.6
64680	www	1	20	23728K	8380K	kqread	5	0:14	0.00%	nginx

“TOP” CPU processes

du

“Disk Usage”

Command format:

du *flags*

Flags:

- h human readable Gb, MB not bits
- s summary view, total for folder not each file

```
[jeff LAPTOP-CVJ54CBV] - [~] - [2020-08-26 08:20:54]
[0] <> cd .ssh
./ authorized_keys id_rsa known_hosts
./ config id_rsa.pub known_hosts.old
[jeff LAPTOP-CVJ54CBV] - [~/ssh] - [2020-08-26 08:20:54]
[0] > du -sh
36K .
[jeff LAPTOP-CVJ54CBV] - [~/ssh] - [2020-08-26 08:21:00]
[0] <>
```

history

Simply lists the recent commands that you have executed

```
340 vim empty1.txt
341 cat empty1.txt
342 more
343 man more
344 cd /var/logs
345 cd /var
346 cd log
347 ls
348 cd ~
349 more .zsh_history
350 less .zsh_history
351 more .zshrc
-[jeff LAPTOP-CVJ54CBV] - [~] - [
-[0] <>
```


man

“Manual”

Only command you really need to know. This will provide you documentation on how to use most unix commands and other things like c++ functions, etc.

```
MAN(1)                                Manual pager utils                                MAN(1)

NAME
    man - an interface to the system reference manuals

SYNOPSIS
    man [man options] [[section] page ...] ...
    man -k [apropos options] regexp ...
    man -K [man options] [section] term ...
    man -f [whatis options] page ...
    man -l [man options] file ...
    man -w|-W [man options] page ...

DESCRIPTION
    man is the system's manual pager. Each page argument given to
    man is normally the name of a program, utility or function. The
    manual page associated with each of these arguments is then found
    and displayed. A section, if provided, will direct man to look
    only in that section of the manual. The default action is to
    search in all of the available sections following a pre-defined
    order (see DEFAULTS), and to show only the first page found, even
    if page exists in several sections.

    The table below shows the section numbers of the manual followed
    by the types of pages they contain.

    1 Executable programs or shell commands
    2 System calls (functions provided by the kernel)
    3 Library calls (functions within program libraries)
    4 Special files (usually found in /dev)

Manual page man(1) line 1 (press h for help or q to quit)
```

tar

Create and extract tar archives

Command format:

```
tar flags <file> [other file]
```

Popular Flags:

- x extract
- v print what's happening
- f file to operate on
- c create
- a compress based on extension

```
~/Documents  
> tar cvaf ritlug.tar.gz RITLUG  
RITLUG/  
RITLUG/Fall2020-w1.odp  
  
~/Documents  
> tar xvf ritlug.tar.gz  
RITLUG/  
RITLUG/Fall2020-w1.odp
```

file

Tells you what kind of file you have

```
~/Documents  
> file ritlug.tar.gz  
ritlug.tar.gz: gzip compressed data, from Unix, original size modulo 2^32 61440
```

Customization

.bashrc

- Basically bash's config file
 - A set of commands that are executed when bash starts
-

Aliases

The alias command let you create shortcuts to other commands

```
alias g=git
```

Put these in your bashrc

Functions

More than just aliases, function can run a whole set of commands, like their own script

```
hello() {  
    echo hello  
}
```



Others Worth Knowing:

- clear
- who
- echo
- whoami
- ping
- ip address
- rsync
- vim
- nano
- |
- >, <
- curl, wget
- diff
- exit
- finger
- head
- tail
- kill
- mkdir
- tree
- passwd
- gzip
- uname
- w
- ps

Questions



What is SSH?

Other than my favorite Linux command.



SSH is...



- AKA: Secure Socket Shell
 - Open source tool used for connecting to a remote computer in a secure manner
 - Version 1 developed in 1995 and second version developed in 2006
 - Authentication with Diffie-Hellman key exchange
-

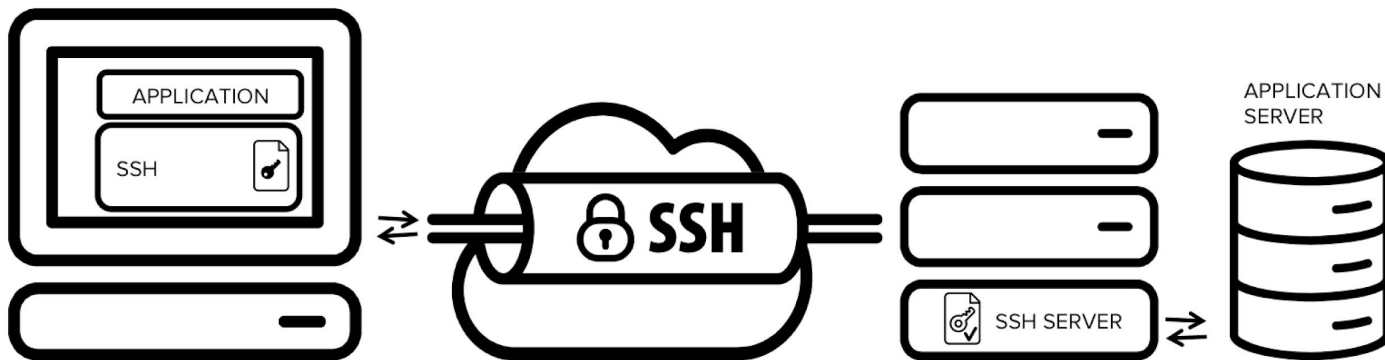
WHY?



SSH is powerful...



- SSH makes connecting to remote computers **very easy**
- Widely implemented
- Supports a plethora of features:
 - Passwordless login with keys
 - File transfer
 - Network port forwarding
 - SSH config files



HOW?



Just pop open a terminal!



- Command syntax
 - `ssh user@hostname -p port -i keyFile`
- The default port used for ssh is 22
- If no user is provided, it will use your current user name

```
[jeff@jrtechs-laptop] - [~] - [2019-09-03 07:51:32]
[1] <> ssh jxr@glados.cs.rit.edu
The authenticity of host 'glados.cs.rit.edu (129.21.22.196)' can't be established.
ECDSA key fingerprint is SHA256:pbfV7PLN905l5ubqUckN2ahZ2iJ9TCG/DGPVxDgMgek.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'glados.cs.rit.edu,129.21.22.196' (ECDSA) to the list of known hosts.
Password: 
```

Keys!

**YOU GET AN SSH KEY
AND YOU GET AN SSH KEY**



EVERYBODY GETS AN SSH KEY





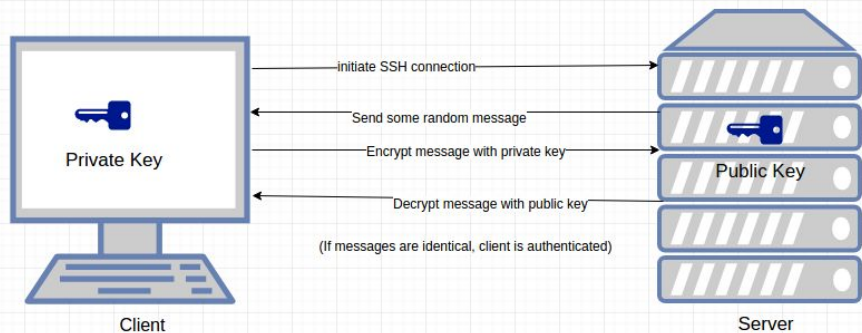
Keys!

- Keys allow you to log into an SSH server without a password.
- SSH uses a public-private key pairing
 - Private key is kept on local machine
 - Public key is distributed to the remote computer

Commands

- ssh-keygen
 - Generates new key pair
- ssh-copy-id
 - Copies your public key to remote server
- ssh-add
 - Adds a private key to the list of keys that ssh will try to connect to remote servers with by default

SSH Authentication



Create a Key

`ssh-keygen -f ~/.ssh/cs-server`

Enter a password if you want
or leave it blank.

```
~  
> ssh-keygen -f ~/.ssh/cs-server  
Generating public/private rsa key pair.  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /home/zethra/.ssh/cs-server  
Your public key has been saved in /home/zethra/.ssh/cs-server.pub  
The key fingerprint is:  
SHA256:Z+05zEoz+IYiuAE+EXSMRfD1lRfaMhXDAPXTCUzKCTI zethra@zethra-xps  
The key's randomart image is:  
+---[RSA 3072]-----+  
| o*E o .==*+=+  
| ..o.+ + +*+=.  
| . . = ++.  
| . o  
| .. S +  
| O . . + O  
| .+ ..+ O  
| ..O . .O.= .  
| .. . . .O.+  
+----[SHA256]-----+
```

Copy Your Public Key to Server

On your machine

- Copy the contents of `~/.ssh/cs-server.pub`
- This is your public key

On the server

- Open `~/.ssh/authorized_keys`
 - You can use nano for this
 - `nano ~/.ssh/authorized_keys`
 - Add your public key on it's own line to the bottom on the file.
-

Now Log In with Your Key

```
ssh -i ~/.ssh/cs-server bag4482@glados.cs.rit.edu
```

No password needed!

Even Better w ssh config!

- Edit the file `~/.ssh/config` on your local machine
- Add a section:

Host cs

HostName glados.cs.rit.edu

User bag4482

IdentityFile ~/.ssh/cs-server

- Now you can just `ssh cs`
-

Questions



Resources



- [SSH Essentials](#)
 - [SSH Config Files](#)
 - [Git Repo and Write Up for Challenge](#)
-

ZSH

```
[jeff LAPTOP-CVJ54CBV] - [~] - [2020-08-26 07:54:30]
[0] <> cd temp
./ ../
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:54:31]
[0] <> touch empty1.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:54:39]
[0] <> ls
./ ../ empty1.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:54:47]
[0] <> cp empty1.txt tempty2.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:55:00]
[0] <> ls -la
total 8
drwxr-xr-x  2 jeff jeff 4096 Aug 26 19:55 ./
drwxr-xr-x 15 jeff jeff 4096 Aug 26 19:55 ../
-rw-r--r--  1 jeff jeff   0 Aug 26 19:54 empty1.txt
-rw-r--r--  1 jeff jeff   0 Aug 26 19:55 tempty2.txt
[jeff LAPTOP-CVJ54CBV] - [~/temp] - [2020-08-26 07:55:02]
[0] <>
```



What is ZSH



Z shell

- Written by Paul Falstad in 1990 while attending Princeton University.
 - The name came from a Professor Zhong Shao since his username was zsh.
 - ZSH is an interactive unix shell that is an extension of Bash with many improvements.
 - ZSH Config file is in the home directory called `.zshrc`
-

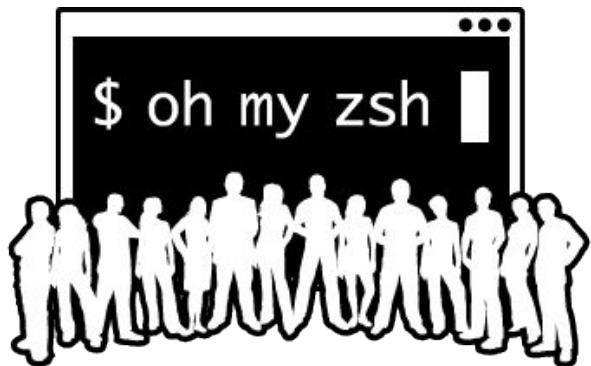


Z shell

Why is ZSH Cool

- Lots of syntactic sugar to make your terminal use easier.
 - Automatic cd if you type name of directory
 - Recursive path expansion
 - Spell correction and approximate completion if you hit tab
 - Plugin and theme support
-

Oh-My-Zsh



- Open source framework for managing Zsh configurations
 - Has 250+ extensions
 - Has 125+ themes
 - Auto updates!
-

Installing Oh-My-Zsh

[Oh My ZSH](#) has a fancy install script you can run if you trust scripts from the internet.

```
sh -c "$(wget  
https://raw.githubusercontent.com/robbyrussell/oh-my-zsh/master/tools/install.sh -O -)"
```



Some of my favorite Oh-my-zsh themes:

birra

```
jeff@zion /mnt/Jeff-Work/Programming/javascript/JrtechsV2
└─$ cd NodeJSBlog
admin          includes          package.json      run
blog           jrtechs.service  photography        server.js
blogContent    LICENSE          previewer.js      sitemap.txt
CODE_OF_CONDUCT.md node_modules      README.md         sites
jeff@zion /mnt/Jeff-Work/Programming/javascript/JrtechsV2/NodeJSBlog <master* >
└─$
```

rkj-repos

```
[jeff jrtechs-laptop] - [~] - [2019-09-25 06:07:00]
└─[0] <> cd dot_files
./      .bashrc          .git/             .ideavimrc  README.md  .zshrc
../     fixSSHfilePermissions.sh .gitconfig        install.sh   .vimrc
[jeff jrtechs-laptop] - [~/dot_files] - [2019-09-25 06:07:13]
└─[0] <git:(master 9a610e6) > cat .zshrc
./      .bashrc          .gitconfig        .vimrc
../     .git/            .ideavimrc        .zshrc
```

Favorite Shells Continued

[Agnosterzak](#):

- Custom Oh-my-zsh theme based on powerline vim plugin and agnoster theme

```
♥ 85% Wed 25 Sep - 18:00 ~
@jeff cd dot_files
./ .bashrc .git/ .ideavimrc README.md .zshrc
../ fixSSHfilePermissions.sh .gitconfig install.sh .vimrc

♥ 85% Wed 25 Sep - 18:01 ~/dot_files | origin Ω master ✓
@jeff cat .zshrc
./ .bashrc .gitconfig .vimrc
../ .git/ .ideavimrc .zshrc
```

Favorite Shells Continued

[starship](#):

- Cross shell theme that's its own program
 - Works with bash, zsh, fish, etc.

```
stargazer on  master is  v0.6.0 via  v1.54.0  
> 1  
contrib doc scripts src target test_data build.rs
```

Plugins

```
36 # Which plugins would you like to load?
37 # Standard plugins can be found in ~/.oh-my-zsh/plugins/*
38 # Custom plugins may be added to ~/.oh-my-zsh/custom/plugins/
39 # Example format: plugins=(rails git textmate ruby lighthouse)
40 # Add wisely, as too many plugins slow down shell startup.
41 plugins=(
42     git
43     history-substring-search
44     z
45     extract
46 )
```

Bash Aliases Still and custom commands Works in ZSH!

```
55 # general bash aliases
56 alias ls="ls -abp --color=auto"
57 alias nano="vim"
58 alias ..="cd .."
59 alias size="echo du -sh && du -sh"
60
61
62 # Bash Manager Things
63 alias roosay="python3 /home/jeff/scripts/bash_manager/src/roosay.py"
64 alias ss="python3 /home/jeff/scripts/bash_manager/src/ssh_manager.py"
65 alias ssh_manager="python3 /home/jeff/scripts/bash_manager/src/ssh_manager.py"
66 alias mm="python3 /home/jeff/scripts/bash_manager/src/mount_ssh_drive.py"
67 alias ssh-mount="python3 /home/jeff/scripts/bash_manager/src/mount_ssh_drive.py"
68 alias quote="python3 /home/jeff/scripts/bash_manager/src/quote.py"
```

More customizations for .bashrc

```
# auto ls when you change directories
function cd {
    builtin cd "$@" && ls
}
```

```
# displays quote when terminal opens
quote
```

Other CLI Programs



Nano Easy to use CLI text editor

```
GNU nano 5.8 src/main.rs
use futures_lite::*;
use get_file::get_file;
use log::{debug, error, info, warn};
use once_cell::sync::{Lazy, OnceCell};
use router::route;
use std::clone::Clone;
use std::convert::TryFrom;
use std::net::SocketAddr;
use std::panic::catch_unwind;
use std::sync::Arc;
use std::{process::exit, str, thread, time::Duration};
use structopt::StructOpt;

pub static SHUTDOWN: Lazy<Sender<>, Receiver<>> =
    Lazy::new(|| async_channel::bounded(1));
// Global executor
pub static EXEC: Lazy<Executor> = Lazy::new(|| {
    // Start executor thread per cpu core
    for n in 1..=CONF.worker_threads {
        thread::Builder::new()
            .name(format!("stargazer-worker-{}", n))
            .spawn(|| loop {
                let _ = catch_unwind(|| {
                    async_io::block_on(EXEC.run(SHUTDOWN.1.recv()))
                });
            })
            .expect("cannot spawn executor thread");
    }
});

Executor::new();
});

static CONF_BACKING: OnceCell<config::Config> = OnceCell::new();
pub static CONF: Lazy<config::Config> = Lazy::new(|| {
    CONF_BACKING
});
```

[Read 506 lines]

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo	M-A Set Mark	M-] To Bracket	M-Q Previous
^X Exit	^R Read File	^E Replace	^P Paste	^J Justify	^/_ Go To Line	M-E Redo	M-6 Copy	^Q Where Was	M-W Next

Vim

More advanced text editor

```
34 use futures_lite::*;
35 use get_file::get_file;
36 use log::{debug, error, info, warn};
37 use once_cell::sync::{Lazy, OnceCell};
38 use router::route;
39 use std::clone::Clone;
40 use std::convert::TryFrom;
41 use std::net::SocketAddr;
42 use std::panic::catch_unwind;
43 use std::sync::Arc;
44 use std::{process::exit, str, thread, time::Duration};
45 use structopt::StructOpt;
46
47 pub static SHUTDOWN: Lazy<(Sender<>, Receiver<>)> =
48     Lazy::new(|| async_channel::bounded(1));
49 /// Global executor
50 pub static EXEC: Lazy<Executor> = Lazy::new(|| {
51     // Start executor thread per cpu core
52     for n in 1..=CONF.worker_threads {
53         thread::Builder::new()
54             .name(format!("stargazer-worker-{}", n))
55             .spawn(|| loop {
56                 let _ = catch_unwind(|| {
57                     async_io::block_on(EXEC.run(SHUTDOWN.1.recv()))
58                 });
59             })
60             .expect("cannot spawn executor thread");
61     }
62     Executor::new()
63 });
64
65 static CONF_BACKING: OnceCell<config::Config> = OnceCell::new();
66 pub static CONF: Lazy<config::Config> = Lazy::new(|| {
67     CONF_BACKING
68         .get()
69         .cloned()
70         .expect("CONF accessed before it was set")
71 });
:set number
```

Neovim Vim with more customization and IDE like features

```
main.rs | build.rs | cli.rs |
42 use std::panic::catch_unwind;
43 use std::sync::Arc;
44 use std::{process::exit, str, thread, time::Duration};
45 use structopt::StructOpt;
46
47 pub static SHUTDOWN: Lazy<Sender<>, Receiver<>> =
48     Lazy::new(|| async_channel::bounded(1));
49 // Global executor
50 pub static EXEC: Lazy<Executor> = Lazy::new(|| {
51     // Start executor thread per cpu core
52     for n in 1..=CONF.worker_threads {
53         thread::Builder::new()
54             .name(format!("stargazer-worker-{}", n))
55             .spawn(|| loop {
56                 let _ = catch_unwind(|| {
57                     async_io::block_on(EXEC.run(SHUTDOWN.1.recv()))
58                 });
59             })
60             .e and(...)~          Ⓢ (Method) [LSP]
61         } and_then(...)~        Ⓢ (Method) [LSP]
62     } as_deref()~               Ⓢ (Method) [LSP]
63     Executor:: as_deref_mut()~  Ⓢ (Method) [LSP]
64 }); as_mut()~                  Ⓢ (Method) [LSP]
65     as_ref()~                   Ⓢ (Method) [LSP]
66 static CONF_BA box~            Ⓢ (Snippet) [LSP]
67 pub static CON call~           Ⓢ (Snippet) [LSP]
68     CONF_BACKI contains(...)~  Ⓢ (Method) [LSP]
69     .get() contains_err(...)~  Ⓢ (Method) [LSP]
70     .cloned()
71     .expect("CONF accessed before it was set")
72 });
73
74 fn main() {
75     use error::Context;
76
```

Tmux

Terminal multiplexer

```
Date: Mon Feb 22 12:47:07 2021 -0500
2 files changed, 7 insertions(+), 4 deletions(-)

tmux on  bat-select [s] took 4s
> gp -f
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 646 bytes | 646.00 KiB/s, done.
Total 5 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To github.com:zethra/tmux.git
+ ad5fdb8...0563841 bat-select -> bat-select (forced update)

tmux on  bat-select
> gau

tmux on  bat-select [+]
> gca
[bat-select 0928fc1] Add battery selection for Linux
Date: Mon Feb 22 12:47:07 2021 -0500
2 files changed, 6 insertions(+), 4 deletions(-)

tmux on  bat-select [s]
> gp -f
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 615 bytes | 153.00 KiB/s, done.
Total 5 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To github.com:zethra/tmux.git
+ 0563841...0928fc1 bat-select -> bat-select (forced update)

tmux on  bat-select
> c
~
> nv
- took 13s
>

0 1 htop 2 zsh
```

```
0 [|||||] 17.2%] Tasks: 118, 561 thr; 1 running
1 [|||||] 15.0%] Load average: 1.84 1.27 1.02
2 [|||||] 23.7%] Uptime: 01:06:20
3 [|||||] 14.3%]
Mem[|||||] 4.44G/15.5G]
Swp[|||||] 0K/4.00G]
```

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
1	root	20	0	161M	11316	8484	S	0.0	0.1	0:01.01	/sbin/init
877	zethra	20	0	2796M	222M	112M	S	20.9	1.4	2:58.26	/usr/bin/kwin_x11
879	zethra	20	0	2796M	222M	112M	S	3.3	1.4	0:31.02	/usr/bin/kwin_x11
1109	zethra	20	0	2796M	222M	112M	S	0.7	1.4	0:12.32	/usr/bin/kwin_x11
882	zethra	20	0	2796M	222M	112M	S	0.0	1.4	0:00.12	/usr/bin/kwin_x11
907	zethra	39	19	2796M	222M	112M	S	0.0	1.4	0:00.00	/usr/bin/kwin_x11
976	zethra	20	0	2796M	222M	112M	S	0.0	1.4	0:00.02	/usr/bin/kwin_x11
1003	zethra	20	0	2796M	222M	112M	S	0.0	1.4	0:05.44	/usr/bin/kwin_x11
1033	zethra	39	19	2796M	222M	112M	S	0.0	1.4	0:00.00	/usr/bin/kwin_x11
1108	zethra	20	0	2796M	222M	112M	S	0.0	1.4	0:00.00	/usr/bin/kwin_x11
15032	zethra	20	0	2796M	222M	112M	S	0.0	1.4	0:00.00	/usr/bin/kwin_x11
16810	zethra	20	0	2796M	222M	112M	S	0.0	1.4	0:00.00	/usr/bin/kwin_x11

F1Help F2Setup F3Search F4Filter F5List F6SortBy F7Nice F8Nice +F9Kill F10Quit

```
~
>
```

bat

Better cat

- Prints file w line numbers and syntax highlighting
- Uses less automatically for larger files

```
22  pub struct Args {
23      /// Print debug information while server is running
24      #[structopt(short = "D", long)]
25      pub debug: bool,
26      /// Path to config file
27      #[structopt(
28          short = "C",
29          long = "config",
30          parse(from_os_str),
31          default_value = "/etc/stargazer.ini"
32      )]
```

delta

Better diff viewer

Regular git diffs are hard
to read

```
diff --git a/src/get_file.rs b/src/get_file.rs
index 8d64049..a5a8fa4 100644
--- a/src/get_file.rs
+++ b/src/get_file.rs
@@ -48,18 +48,12 @@ pub async fn get_file<'a>(
     let path = static_route.root.join(Path::new(&os_str));
     log::debug!("Requested file path: {}", path.display());

-    let mut path_string = path_str.to_owned();
     let (path, path_type, mut mime_type) = unblock(move || {
         if !path.exists() {
             return Err(GemError::NotFound);
         }

         if path.is_dir() {
             // Redirect directories to have a trailing slash
-            if !path_string.ends_with('/') {
-                path_string.push('/');
-                return Err(GemError::Redirect(path_string));
-            }
             let index_path = path.join(&static_route.index);
             if index_path.exists() {
                 let mime_type = get_mime_type(&index_path);
@@ -76,6 +70,12 @@ pub async fn get_file<'a>(
         if path_type == PathType::Dir && !static_route.auto_index {
             return Err(GemError::NotFound);
         }
     }

+    // TODO show I be doing this?
+    if path_type == PathType::Dir && !path_str.ends_with('/') {
+        let mut new_path = path_str.to_owned();
+        new_path.push('/');
+        return Err(GemError::Redirect(new_path));
     }
```

delta

Better diff viewer

src/get_file.rs

```
48: pub async fn get_file<'a>(<div data-bbox="42 322 468 732" data-label="Text">

```
48 | let path = static_route.root.join(Path::new(&os_str));
49 | log::debug!("Requested file path: {}", path.display());
50 |
51 | let mut path_string = path_str.to_owned();
52 | let (path, path_type, mut mime_type) = unblock(move || {
53 | if !path.exists() {
54 | return Err(GemError::NotFound);
55 | }
56 |
57 | if path.is_dir() {
58 | // Redirect directories to have a trailing slash
59 | if !path_string.ends_with('/') {
60 | path_string.push('/');
61 | return Err(GemError::Redirect(path_string));
62 | }
63 | let index_path = path.join(&static_route.index);
64 | if index_path.exists() {
65 | let mime_type = get_mime_type(&index_path);
```


```

```
70: pub async fn get_file<'a>(<div data-bbox="42 822 488 890" data-label="Text">

```
76 | if path_type == PathType::Dir && !static_route.auto_index {
77 | return Err(GemError::NotFound);
78 | }
```


```

```
48 | let path = static_route.root.join(Path::new(&os_str));
49 | log::debug!("Requested file path: {}", path.display());
50 |
51 | let (path, path_type, mut mime_type) = unblock(move || {
52 |     if !path.exists() {
53 |         return Err(GemError::NotFound);
54 |     }
55 |
56 |     if path.is_dir() {
57 |
58 |         let index_path = path.join(&static_route.index);
59 |         if index_path.exists() {
60 |             let mime_type = get_mime_type(&index_path);
```

```
70 | if path_type == PathType::Dir && !static_route.auto_index {
71 |     return Err(GemError::NotFound);
72 | }
73 | // TODO show I be doing this?
74 | if path_type == PathType::Dir && !path_str.ends_with('/') {
75 |     let mut new_path = path_str.to_owned();
```

ripgrep

Like grep but fast and better defaults for searching code

- Recursive and line numbers by default
- Ignores hidden files
- Ignores files ignored by git
- Powers the code search in VS Code

```
> rg get_file
src/get_file.rs
39:pub async fn get_file<'a>(

src/cgi.rs
23:use crate::{get_file::parse_path, router::SCGIAddress};

src/main.rs
22:mod get_file;
35:use get_file::get_file;
448:    get_file(
```


fzf

Command line fuzzy finder

- Search for files or text interactively
- Can be used in scripts

```
test_data/cgi-bin/loop.c
Cargo.lock
test_data/client-cert.pem
test_data/client-key.pem
scripts/clean
> src/cli.rs
19/74
> cl
```