Introduction to the Command Line

VALA Tech Camp 2019

Hugh Rundle

Sae Ra Germaine

Before we start…

Tech Check

https://hugh.run/cli-workshop

Why are we here?

- Who are they?
- What are they using to access Bash?
- Why did they choose this workshop?

What we’re going to learn today

- Terminology
- Core commands and syntax
- Key tools
- Mistakes to avoid
- Where to find more information

```
sed 's/Communism/Tech Camp/g' manifesto.txt | grep —i 'Tech Camp' > communism.txt
```

What is a command line?

…and why Bash?

Navigating

- Left and right arrows to move along a line
- Ctrl + A to go to start of line
- Ctrl + E to go to end of line
- Ctrl + C to cancel an operation
- Up arrow to go to previous command
- Down arrow to go to next command
- Tab to autocomplete

Commands

```
command argument –flag

command argument --flag

command –flag argument

command –flag1 –flag2 argument

command argument –flag argument
```

Commands

```
grep —n "workshop" manifesto.txt
```

Files and directories

- Files are discrete pieces of data with a name and dedicated space in the filing system
- Standard file – e.g. Word document
- Executables are files that can run as a program – e.g. the MS Word program
- Directories (aka folders) are conceptual spaces for organising files

```
/directory/subdirectory/file
```

Five commands to start with

- `whoami`
- `date`
- `pwd`
- `ls`
- `cd`

ls

```
ls /bin
```

cd

```
cd ~

cd ..

↑ + [Enter]
```

Paths

```
/home/hugh/tech_camp/timer.sh

timer.sh

~/tech_camp/timer.sh
```

Names

Names

```
/bin

~/bin

/usr/bin

/usr/local/bin
```

Names

```
~/"my file name with spaces.txt"

~/my_file_name_with_underscores.txt
```

Creating new files & directories

```
mkdir ~/tech-camp

touch myfile.txt
```

Creating new files & directories

```
~/
   tech-camp/
      myfile.txt
      my_new_directory/
```

Copying files and directories

```
cp existingfile newlocation

cp myfile.txt another_dir

cp –r mydir /another/directory

cp -r mydir/* another/directory
```

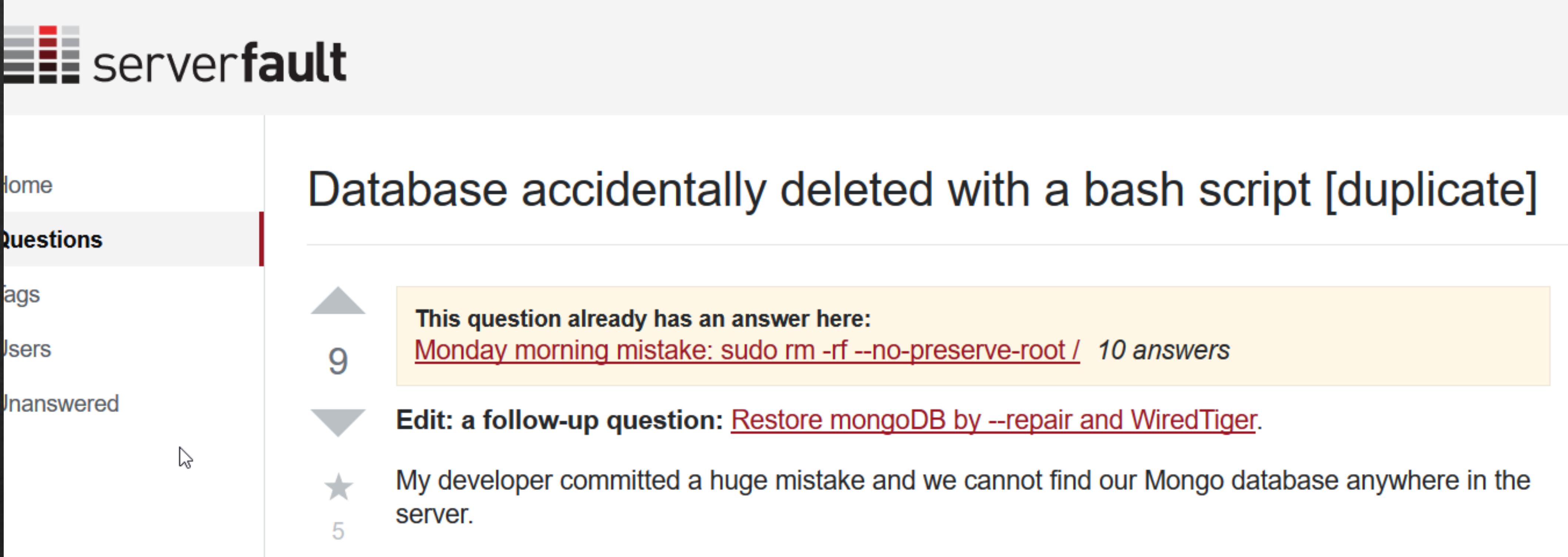
Moving and renaming files and directories

```
mv file /newdirectory
```

```
mv oldname newname
```

```
mv another_dir/myfile.txt another_dir/moved.txt
```

Deleting files and directories



Database accidentally deleted with a bash script [duplicate]

This question already has an answer here:
Monday morning mistake: sudo rm -rf --no-preserve-root / *10 answers*

Edit: a follow-up question: Restore mongoDB by --repair and WiredTiger.

My developer committed a huge mistake and we cannot find our Mongo database anywhere in the server.

Deleting files and directories

```
rm filename

rm –i filename

rm –i another_dir/moved.txt

rm –r another_dir
```

Permissions and Super User

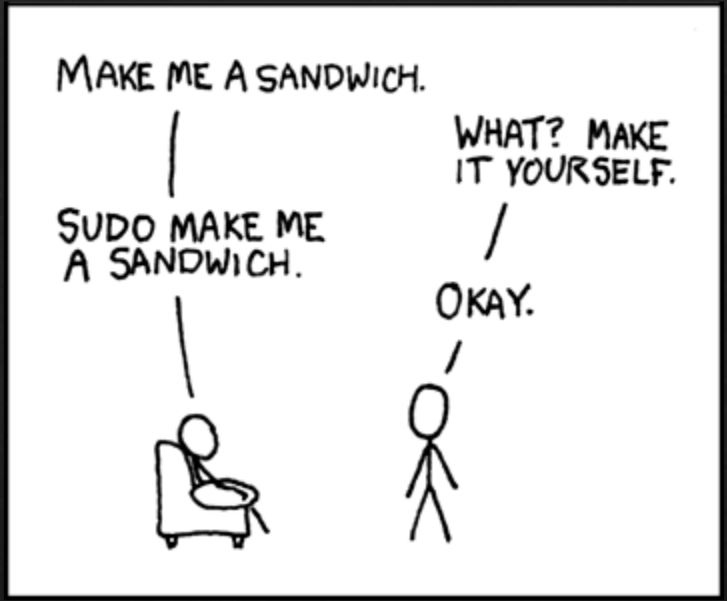
```
ls –l
```

```
-rw-rw-r--
```

```
chmod [flag] mode file
```

```
chmod o-r myfile.txt
```

sudo


MAKE ME A SANDWICH.
WHAT? MAKE IT YOURSELF.
SUDO MAKE ME A SANDWICH.
OKAY.

Ownership

```
sudo chown user:group file

sudo chown hugh:hugh sudofile.md

sudo mkdir newdir

sudo touch newdir/newfile.txt
```

More on ownership

```
sudo chown hugh:hugh newdir

ls –l newdir
```

More on ownership

```
sudo chown —R hugh:hugh newdir

ls —l newdir
```

Do you have a /usr/local/bin ?

```
ls –l /usr/local/bin

sudo mkdir /usr/local/bin

sudo chown hugh:hugh /usr/local/bin
```

RTFM

```
man
```

```
command [optional thing] argument
```

Editing text with Vi and Vim

j go down one line

k go up one line

i insert mode

Esc command mode

: last line mode

:q quit

:q! no, really, quit without saving

:wq save the file and then quit

Editing text with nano

```
^ = Ctrl

Ctrl + O Save file
Ctrl + X Exit, optionally save file
Ctrl + W find text string
```

Printing to the screen

```
cat myfile.txt
echo myfile.txt
```

PATH

```
echo $PATH
```

Streams and redirection

STDIN

STDOUT

STDERR

```
echo "Hello from the command line" > hello.txt

echo "Here is another line." >> hello.txt
```

Shell scripting

```
nano sayhello.sh

echo "Hello World!"

bash sayhello.sh
```

Shell scripting

```
nano timer.sh
```

Shell scripting

```
echo "The time is $(date)"
printf "\a"
```

Shell scripting

```
while sleep 5; do
  echo "The time is $(date)"
  printf "\a"
done
```

Making files executable

```
chmod o-r file
```

```
chmod +x timer.sh
```

Making files executable – shebangs

```
#!/usr/bin/env bash
while sleep 5; do
  echo "The time is $(date +%H:%M:%S)"
  printf "\a"
done
```

Running executables

```
./timer.sh

ln –s source_file target_location

ln –s ~/tech-camp/timer.sh /usr/local/bin/mytimer

ls –l /usr/local/bin/mytimer

mytimer
```

User input

```
#!/usr/bin/env bash
echo "Getting name from input..."
read my_name
echo "Hello $my_name"
while sleep 5; do
  echo "The time is $(date)"
  printf "\a"
done
```

Piping

```
whoami | mytimer
```

Logs

```
/var/log
/usr/local/var/log
```

Logs

```
mytimer >> timer.log
```

Logs

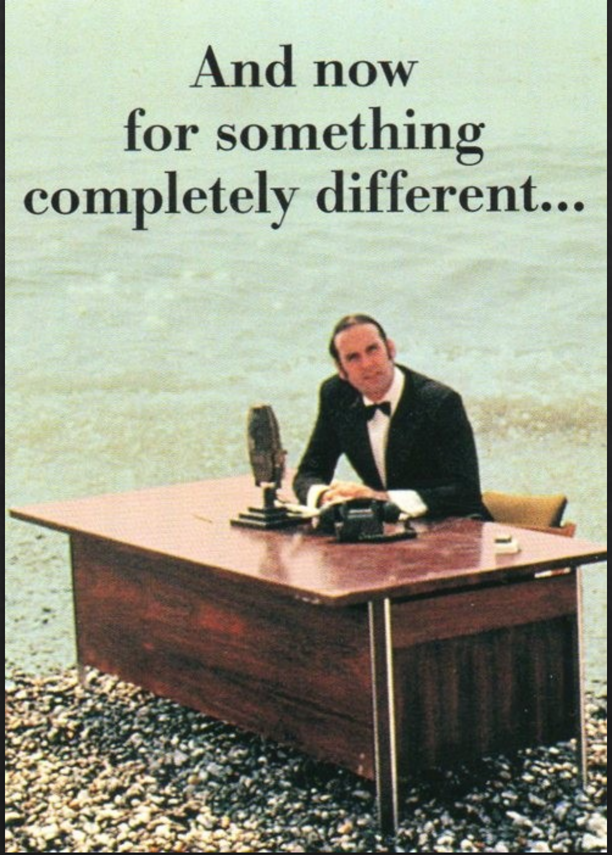
```
whoami | mytimer >> timer.log
```

tail

```
tail timer.log

tail –f timer.log
```

curl



curl

```
curl hugh.li
```

curl

```
curl —L hugh.li
```

curl

```
curl parrot.live
```

curl

```
curl –L hugh.li/manifesto > manifesto.txt
```

grep

```
grep 'workshop' manifesto.txt
```

grep

```
grep —n 'workshop' manifesto.txt
```

grep

```
grep —c 'workshop' manifesto.txt
```

grep

```
grep —ci 'workshop' manifesto.txt
```

grep

```
grep —i '^social*' manifesto.txt
```

grep

```
grep —i 'spect[re][er]' manifesto.txt
```

sed

```
grep —i 'spect[re][er]' manifesto.txt > spectre.txt

sed 's/spectre/librarian/g' spectre.txt
```

sed

```
sed 's/[Ss]pectre/librarian/g' spectre.txt
```

Putting it all together

```
sed 's/Communism/Tech Camp/g' manifesto.txt | grep –i 'Tech Camp' >
communism.txt
```

tar and gzip

```
tar –cf archive.tar file1 file2 file3

tar –cf archive.tar directory

gzip archive.tar
```

tar and gzip

```
gunzip archive.tar.gz

tar –xf archive.tar
```

tar and gzip combined

```
tar –czf archive.tar.gz directory

tar –xzf archive.tar.gz
```

rsync

```
rsync —rz source destination
```

```
rsync —rz --del source destination
```

SSH

```
ssh hugh@123.456.789
```

```
ssh-keygen
```

```
~/.ssh/id_rsa
```

```
ssh-copy-id username@remotehost
```

Projects to work on

https://hugh.run/cli-workshop/projects

Stay in touch

Twitter: @hughrundle

Mastodon: @hugh@ausglam.space

Email: hugh@hughrundle.net