

### ERROR checking

```
sudo apt update && sudo apt upgrade -y
```

upgrades your machine with all tools completely

0 passes

any number other than 0 is a fail error

```
test -d /tmp/temp_dir
```

check to see if directory was created

```
test_rc=$?
```

check return codes

```
echo "mkdir resulted in ${mkdir_rc}, test resulted in ${test_rc}."
```

Check out how an exit status of 0 affects the logical operators:

```
true && echo "We get here because the first part is true!" true || echo "We never see this because the first part is true :(""
```

Check out how an exit status of 1 affects the logical operators:

```
false && echo "Since we only continue after && with an exit status of 0, this is never printed." false || echo "Because we only continue after || with a return code that is not 0, we see this!"
```

We need exactly three arguments, check how many have been passed to # the script.

### ERROR checking (cont)

```
directory_name=$1 file_name=$2 file_content=$3
```

Save the arguments into variables.

```
absolute_file_path=${directory_name}/${file_name}
```

create absolute path

```
if [[ ! -d ${directory_name} ]]; then mkdir ${directory_name} || { echo "Cannot create directory, exiting script!"; exit 1; } fi
```

Check if the directory exists; otherwise, try to create it.

```
if [[ ! -f ${absolute_file_path} ]]; then touch ${absolute_file_path} || { echo "Cannot create file, exiting script!"; exit 1; } fi
```

Try to create the file, if it does not exist.

```
echo ${file_content} > ${absolute_file_path}
```

File has been created, echo the content to it.

```
cp /var/log/dpkg.log dpkg || { echo "Cannot copy dpkg.log to the new directory."; exit 1; }
```

Copy the log file to our new directory.

```
cd $(dirname $0)
```

Change directory to the script location.

```
cp /var/log/dpkg.log dpkg || { echo "Cannot copy dpkg.log to the new directory."; exit 1; }
```

Copy the log file to our new directory.

### ERROR checking (cont)

```
if [[ $# -ne 1 ]]; then echo "Incorrect usage!" echo "Usage: $0 <file or directory path>" exit 1 fi
```

input validation

### git keeper

```
git clone <url>
```

copies url to directory

```
git add <file>
```

keep track of git file

```
git commit -am "-done"
```

makes version of thing done

```
git push
```

pushes version to git server

### for loop

```
for word in $(cat words.txt); do echo "The word is: $word" done
```

for loop example

```
" "
```

bash keep spaces together also prevents asterisk from being expanded

```
' '
```

no variables will be replaced in single quotes plus what double quotes do

### Setting up Samba for file sharing

```
$ sudo yum install samba
```

install samba

```
$ sudo cp /etc/samba/smb.conf /etc/samba/smb.conf.bak
```

backup at each stage

### Setting up Samba for file sharing (cont)

\$ sudo mkdir /SharedFolder/ \$ sudo chmod -R 755 /SharedFolder/ grant user permissions

\$ sudo firewall-cmd --add-service=samba \$ sudo firewall-cmd --reload get past firewall

\$ sudo nano /etc/samba/smb.conf configure samba file

\$ sudo smbpasswd -a <user> create user

\$ sudo systemctl enable smb.service \$ sudo systemctl enable nmb.service \$ sudo systemctl start smb.service \$ sudo systemctl start nmb.service start service

\$ testparm -v listing of all the Samba configuration options and the values to which they are currently set

\$ sudo smbpasswd -a <user> set up password for user

### ARRAYS

ARRAY=(hi "how are" you) set values

echo \${ARRAY[\*]} list variables in array

for x in "\${ARRAY[@]}"; do echo \$x; done list variables in new lines

unset ARRAY[variable] remove variable

### ARRAYS (cont)

ARRAY+=("bland" "hello") add variables

echo \${#ARRAY} get length of array

\${VAR:-WORD} substitute variable

"\$@" shift variables

declare -A aa

\${VAR/PATTERN/STRING} replace parts of a variable

\${VAR:OFFSET:LENGTH} remove substring

### general

lists the contents of files to the terminal window cat

give your own name to a command or sequence of commands alias

information and files from Uniform Resource Locators (URLs) or internet addresses curl

shows the size, used space, and available space on the mounted filesystems of your computer df

compares two text files and shows the differences between them diff

close a terminal window exit

gives you a short dump of information about a user, including the time of the user's last login, the user's home directory, and the user account's full name finger

summary of the memory usage with your computer. free

tells you which groups a user is a member of groups

compresses files gzip

### general (cont)

gives you a listing of the first 10 lines of a file head

terminate a process from the command line kill

view files without opening an editor less

verify that you have network connectivity with another network device ping

lists running processes ps

shut down or reboot your Linux system shutdown

listing of the last 10 lines of a file tail

create an archive file tar

real-time display of the data relating to your Linux machine. top

obtain some system information regarding the Linux computer you're working on uname

lists the currently logged in users. w

find out who you are logged in as or who is logged into an unattended Linux terminal. whoami

### Add users

add user sudo useradd <username>

add user password sudo smbpasswd -a <username>

Add Admin sudo useradd -g wheel <lastnamefirstletter>

### Installing and setting up samba

check samba is not running ps ax | grep "samba|smbd|nmbd|winbindd"

### Installing and setting up samba (cont)

delete any samba files that are already on the system

```
smbd -b | grep "CONFIG-FILE" and smbd -b | egrep "LOCKDIR|STATEDIR|CACHEDIR|PRIVATE_DIR";
```

delete any files that these commands find

provision Samba AD. We'll want to do it interactively so our password can't be seen

```
samba-tool domain provision --use-rfc2307 --interactive
```

for the requests:

Realm: SAP.CSLAB.MORAVIAN.EDU Domain: SAP Server Role: dc DNS backend: SAMBA\_INTERNAL

Set your hostname to our domain:

```
hostname sap.cslab.moravian.edu hostname -b sap.cslab.moravian.edu
```

Now enable samba and make it sure it will run at boot:

```
sudo systemctl start samba sudo systemctl enable samba
```

### virtual environments

python3 -m venv /path/to/new/virtual/environment

create new virtual environment

-h, --help

man pages for environments

--system-site-packages

Give the virtual environment access to the system site-packages dir.

### virtual environments (cont)

--

Try to use symlinks rather than copies, when symlinks are not the default for the platform.

--copies

Try to use copies rather than symlinks, even when symlinks are the default for the platform.

--clear

Delete the contents of the environment directory if it already exists, before environment creation.

--

upgrade

Upgrade the environment directory to use this version of Python, assuming Python has been upgraded in-place.

--without-pip

Skips installing or upgrading pip in the virtual environment (pip is bootstrapped by default)

--prompt PROMPT

Provides an alternative prompt prefix for this environment.

<source>

activate virtual env

<dir>/-bin/activate/

pip install <some project>

install a project

pip install -r requirements.txt

install required files

### Files and Directories

ls

to hide files

ls -a

shows all files in current folder

ls ..

look in directory

ls ../.. -a

look in directory above

ls -a ..

look in hidden directory

./

run program from current folder

pwd

present working directory

tab

completes command

>

moves files somewhere else

rm ~/<filename>

delete file

rm -r <dir>

delete directories

mk dir

create directories

wget <folder>

download folder

tar -xzf <folder>.tar.gz

unpack folder

cd <directory>

change directory

cd \

go to home directory

mv

move

History | grep <command>

find commands with <command> in it

-f

to follow code to the end

-f -u <unit>

filter by the type of unit

!<number>

repeat command on line

script

everything that happened in a session

### ports and firewalls

firewall-cmd --permanent --add-port=<port#>/tcp

add port

firewall-cmd --reload

reboot port

firewall-cmd --list-ports

list ports

### ports an firewalls (cont)

firewall-cmd -- use predefined  
permanent --add-- service instead of  
service=http port #

sudo tcpdump -a -c looks for the past  
1000 1000 connections  
within server

ifconfig show local ip

### Network Diagnostic Tools

ping <we- tracks the time the round trip  
baddress> took to server  
or <IP>

ping -s try a packet of <number>  
<number>

/path/to/- shows whether it can reach a  
tracert host, but also the route it  
<webad- takes  
dress>

dig queries DNS servers and  
returns the information held  
about a particular domain.

### Set up SMB

go to sudo nano /etc/samba/smb.conf  
file

input [homes] comment = Home Direct-  
values ories browsable = no writeable =  
yes valid users = sap.csllab.morav-  
ian.edu\%S [printers] comment =  
All Printers path = /var/spoo-  
l/samba browsable = no guest ok =  
no writeable = no printable = yes

run sudo setsebool -P samba\_enable-  
\_home\_dirs on

### Set up DHCP

Setting ip addr add 192.168.114.1 dev  
up br0;

Static IP  
on  
Server  
(For  
DHCP)

Setting sudo dnf install bridge-utils sudo  
up brctl addbr br0 sudo brctl addif  
Bridging enp1s0 sudo brctl addif eno1  
sudo ifconfig br0 up

DHCP subnet 192.168.114.0 netmask  
Config 255.255.255.0 { range 192.16-  
8.114.1 192.168.114.255; option  
subnet-mask 255.255.255.0;  
option routers 192.168.114.1;  
option broadcast-address  
192.168.114.255; default-lease-  
time 600; max-lease-time 7200;  
host interface0 { hardware  
ethernet 54:04:a6:3f:85:35;  
fixed-address 192.168.114.1; } }

DHCP /etc/sysconfig/network-scripts/if-  
Config cfg-eno1:  
pt.2

DHCP DEVICE=eno1 BOOTPR-  
Config OTO:dhcp TYPE: ethernet  
pt.3 ipaddr=192.168.114.1  
ONBOOT=yes

### Permissions and Owners

ls -l view permissions of files  
and directories

./<filename> execute file

chmod +x <filename>

ls -l added permission to  
execute file

chown bin changes ownership of  
file(can only be used by  
root)

chgrp bin changes group

chmod -x removed permission to  
<filename> execute file

chmod u+x assign execute permission  
<filename> to yourself only

chmod assign read, write, execute  
ug+rwX <fi- permission to user and  
lename> group

etc /shadow can only be run by root

chmod -R recursive

history show all command history

adduser adds user

### Cups printer

set up cups lpadmin -p laserjet -L "-  
Printer" -v usb://HP/Las-  
erJet%202200?serial=0-  
0USBGJ02797

list the available lpinfo -v  
backends and  
printers

The Bonjour dnssd  
(DNS-SD)  
protocol.

The Internet ipp  
Printing Protocol  
(IPP) with  
optional encryp-  
tion.

The Internet ipp  
Printing Protocol  
with mandatory  
encryption.

### Cups printer (cont)

|                                                                                            |                                                                        |
|--------------------------------------------------------------------------------------------|------------------------------------------------------------------------|
| The Line Printer Daemon protocol                                                           | lpd:                                                                   |
| The AppSocket (JetDirect) protocol.                                                        | socket                                                                 |
| Set printer                                                                                | lpadmin -p printername -v device-uri                                   |
| aborts jobs. on printer                                                                    | abort-job                                                              |
| Enables/disables per-printer sharing                                                       | -o printer-is-shared=true/false                                        |
| Enables sharing of printers with other computers and mobile devices on your local network. | --share-printers                                                       |
| Expands printer sharing to any network that can reach your server.                         | --remote-any                                                           |
| create, modify, or delete a class                                                          | lpadmin -p printername -c classname                                    |
| printer is removed from the class                                                          | lpadmin -p printername -r classname                                    |
| deletes the named class                                                                    | lpadmin -x classname                                                   |
| enable debug logging                                                                       | cupscctl --debug-logging                                               |
| disable debug logging                                                                      | cupscctl --no-debug-logging                                            |
| creates a destination for a printer at IP address 11.22.33.44                              | lpadmin -p printername -E -v ipp://11.22.33.44/ipp/print -m everywhere |

### Cups printer (cont)

|                                                                           |                                                                                     |
|---------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| creates a destination for a HP LaserJet printer at IP address 11.22.33.44 | lpadmin -p printername -E -v socket://11.22.33.44 -m drv:///sample.drv/laserjet.ppd |
|---------------------------------------------------------------------------|-------------------------------------------------------------------------------------|

### SSH

|                                                                  |                                           |
|------------------------------------------------------------------|-------------------------------------------|
| \$ ssh -l <account name> <remote system>                         | log in                                    |
| \$ ssh <account name>@<remote system>                            | log in                                    |
| \$ ssh <remote system>                                           | log in if on same local and remote system |
| \$ ssh -l <account name> <remote system> rm <absolute path name> | run a single command                      |
| \$ scp <file name> <account name>@<remote server>:               | copy files from local to remote           |
| \$ scp <account name>@<remote system>:<file name> .              | copy files from remote to local           |
| \$ scp -r <account name>@<remote system>:<directory> .           | copy directory                            |

