

Introduction to Linux

(Tsai) Wei Wu

10/21/2024

10/23/2024

Command Line Interface (CLI)



Help

- User Guide

arch.jhu.edu/guide

arch.jhu.edu/support/faq/

arch.jhu.edu/support/

- Ticket

help@rockfish.jhu.edu

- User ID at Rockfish.
- Detailed explanation of the problem/issue.
- Add snapshots if possible.
- Path to the working directory, scripts and (slurm) output files.

- Training workshop - interactive

- Google

<https://info-ee.surrey.ac.uk/Teaching/Unix/>



JOHNS HOPKINS
UNIVERSITY

Advanced Research
Computing at Hopkins

github.com/tsaiweiwu/arch



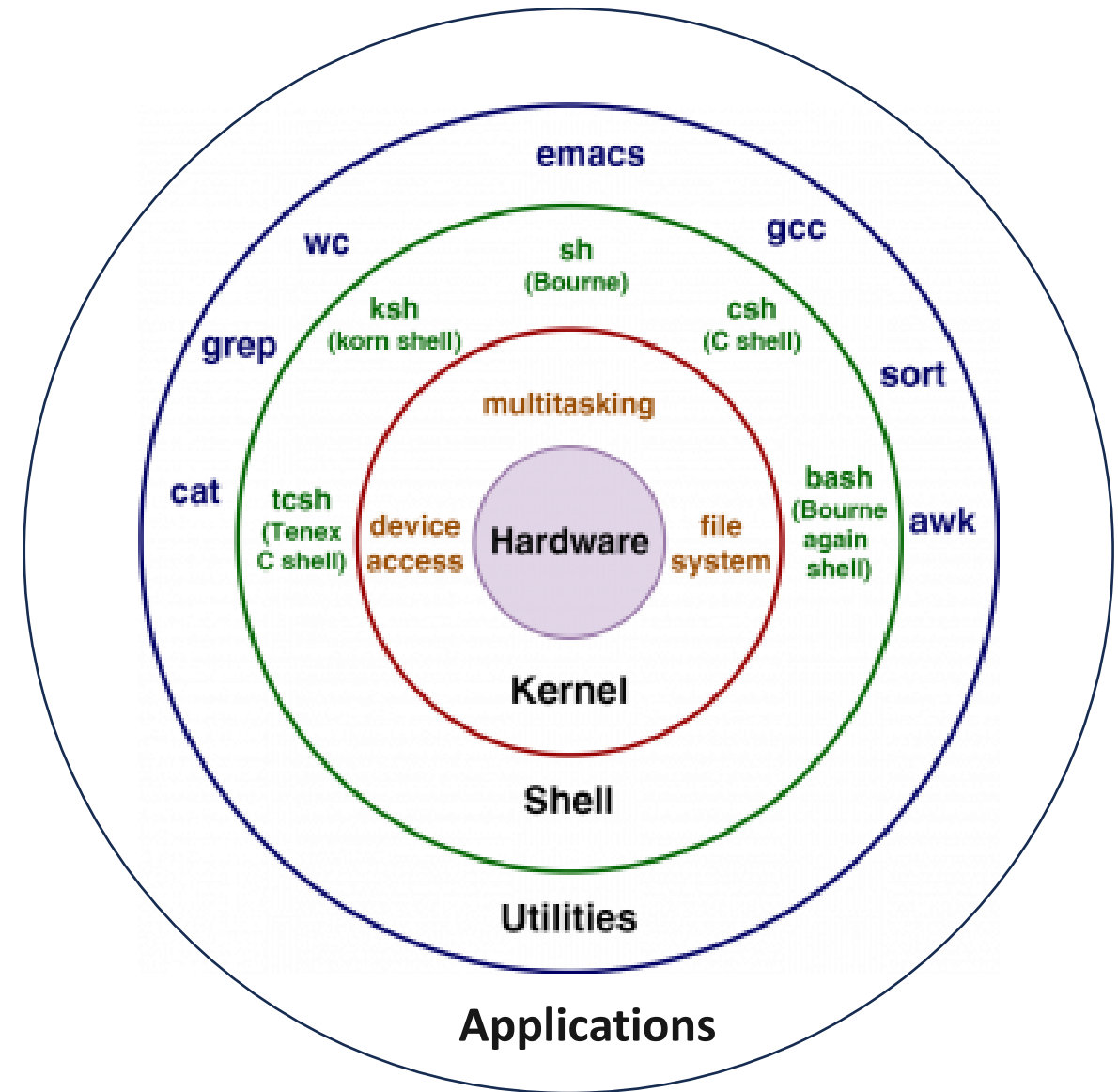
Outline

- Linux Architecture
- Connection and File Transfer
- Login Page
- Environment variables + Hidden Files
- Basic Commands
- Command Redirections
- File Permissions
- Text Editor
- Bash Script



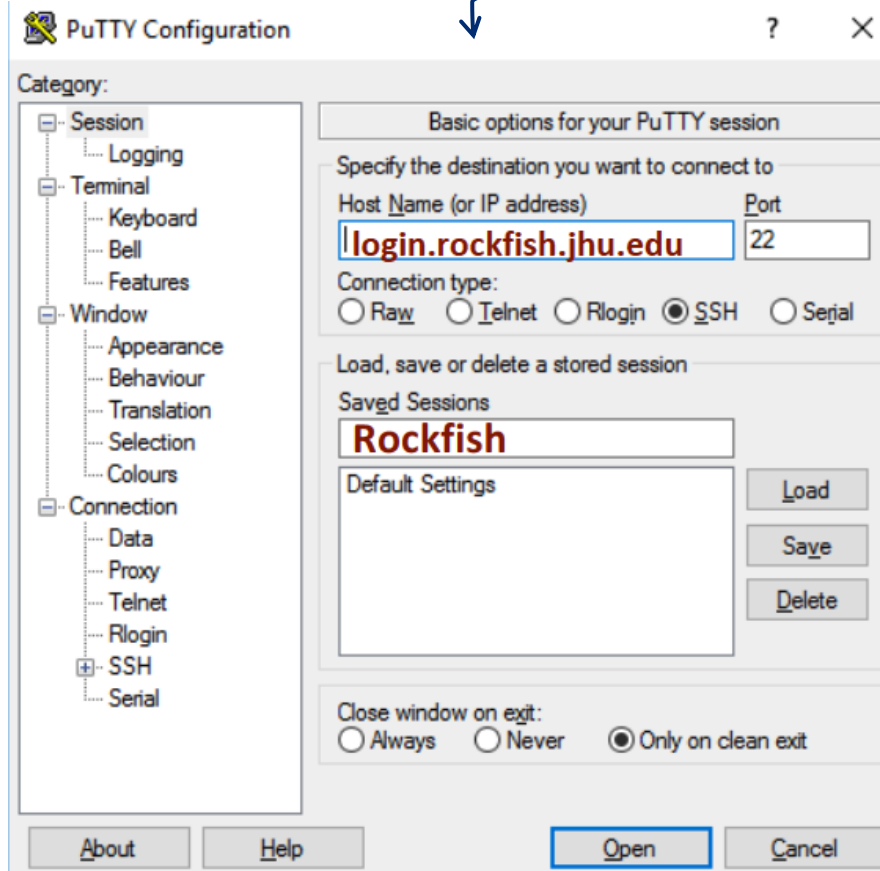
Linux Architecture

- Hardware: CPU, RAM, IO, GPU
- Kernel: core of OS
- Shell: bash
- Utilities:
 - basic commands
 - text editor - vi, emacs, **nano**
 - compilers - gcc, intel, aocc, nvcc



<https://puneetpanwar.com/introduction-to-linux-command-line/>

Connection



- Putty
- MobaXterm
- SecureCRT

gui

- Open OnDemand (OOD)
(requires JHU VPN)
portal.rockfish.jhu.edu

web

- PowerShell
- Windows Sub-Linux System

cli

Connection with CLI

Secure Shell

```
$ ssh -l userid login.rockfish.jhu.edu
```

```
$ ssh userid@login.rockfish.jhu.edu
```

```
$ ssh -XY userid@login.rockfish.jhu.edu
```

For Graphical Applications, but X11 forwarding is painfully slow!!
Use OOD instead.

File Transfer with CLI

environment variables

\$HOME = ~ = /home/\$USER

Secure Copy **For small data**

\$ scp local_file userid@login.rockfish.jhu.edu: ☺

\$ scp **(-r)** local_dir userid@login.rockfish.jhu.edu: \$HOME

\$ scp userid@login.rockfish.jhu.edu:/path/to/file ☹

Globus **For Large data set**



File Transfer with globus.org

use JHED to create an account
on coldfront.rockfish.jhu.edu

Log in to use Globus Web App

Use your existing organizational login

e.g., university, national lab, facility, project

Johns Hopkins

By selecting Continue, you agree to Globus [terms of service](#) and [privacy policy](#).

Continue



Globus uses CILogon to enable you to Log In from this organization. By clicking Continue, you agree to the [CILogon privacy policy](#) and you agree to share your username, email address, and affiliation with CILogon and Globus. You also agree for CILogon to issue a certificate that allows Globus to act on your behalf.

OR














Sign in with GitHub



Sign in with Google





Sign in with ORCID iD





Collection Search


Collection

**Rockfish AWS**
Subscribed Mapped Collection (GCS) on **RFDTN02**
Owner: marcc@globusid.org
Domain: m-4c0e79.8b416.0ec8.data.globus.org

**Rockfish dtn01 user data**
Subscribed Mapped Collection (GCS) on RFDTN01
Owner: a933ac1a-c486-473e-81f5-06baeedb08f2@clients.auth.globus.org
Domain: m-6971d6.5e9e0.bd7c.data.globus.org
Description: RFDTN01 endpoint for rockfish cluster data

**Rockfish dtn02 user data**
Subscribed Mapped Collection (GCS) on **RFDTN02**
Owner: a7627ba9-e25e-44bd-9d48-d64aab883008@clients.auth.globus.org
Domain: m-0a4c68.8b416.0ec8.data.globus.org
Description: RFDTN02 endpoint for rockfish cluster data

**rbradle8 rockfish data**
Guest Collection (GCS) on **Rockfish dtn02 user data**
Owner: rbradle8@johnshopkins.edu
Domain: g-0e4811.8b416.0ec8.data.globus.org
Description: Research data example for transfer to other sites.

**rf-genemaprdata**
Guest Collection (GCS) on **Rockfish dtn01 user data**
Owner: kesohku1@johnshopkins.edu
Domain: g-f8e93f.5e9e0.bd7c.data.globus.org
Description: GeneMAP research data on Rockfish cluster

Exercises

1. Connect to Rockfish with the method you like
2. `$ hostname`
3. Downloading files from internet into your `$HOME`
 - `$ wget https://tinyurl.com/training-url-txt`
 - `$ curl -O https://tinyurl.com/training-url-txt`

Exercises

4. Secure copy folders from RF \$HOME to your laptop
 - `$ cat training-url-txt`
 - `$ git clone git@github.com:tsaiweiwu/arch.git`
5. Secure copy folders from your RF \$HOME to your laptop
 - `$ scp -r user_id@login.rockfish.jhu.edu:$HOME/arch .`


```
[twu73@login02 ~]$
```

- ```
$ cd .. (one folder up)
$ cd ../.. (two folders up)
```

```
tsaiwei@DESKTOP-UCTGA49:~$ ssh -Y twu73@login.rockfish.jhu.edu
twu73@login.rockfish.jhu.edu:~$ password.
Warning: No xauth data; using fake authentication data for X11 forwarding.
Last login: Fri Apr 19 11:03:59 2024 from 73.103.78.70
```

# Rediffusion

```
[STATUS] loading software modules
[STATUS] modules are Lmod (https://lmod.readthedocs.io/en/latest/)
[STATUS] software is Spack (https://spack.readthedocs.io/en/latest/)
[STATUS] the default modules ("module restore") use GCC 9 and OpenMPI 3.1
[STATUS] you can search available modules with: module spider <name>
[STATUS] you can list available modules with: module avail
[STATUS] loading a compiler, MPI, Python, or R will reveal new packages
[STATUS] and you can check your loaded modules with: module list --terse
[STATUS] to hide this message in the future: touch ~/.no_rf_banner
[STATUS] restoring your default module collection now
[STATUS] quota information listed below, you can view by invoking "quotas.py"
```

| Home Usage for user twu73 as of Mon Apr 22 18:00:08 2024 |       |         |         |
|----------------------------------------------------------|-------|---------|---------|
| Used                                                     | Quota | Percent | Files   |
| 13.88 GB                                                 | 50 GB | N/A     | 273,244 |

| GPFS Usage for Group rfdadmin: as of Mon Apr 22 18:30:13 2024 |          |           |        |            |             |         |
|---------------------------------------------------------------|----------|-----------|--------|------------|-------------|---------|
| FS                                                            | Used     | Quota     | Used % | Files      | Files Quota | Files % |
| data                                                          | 50.51 TB | 150.00 TB | 33.00% | 29,756,870 | 61,440,000  | 48.00%  |
| scratch4                                                      | 7.40 TB  | 10.00 TB  | 74.00% | 7,035,797  | 20,480,000  | 34.00%  |
| scratch16                                                     | 2.43 TB  | 10.00 TB  | 24.00% | 3,035,106  | 10,240,000  | 29.00%  |

| GPFS Usage for Group jaimecomb: as of Mon Apr 22 18:30:13 2024 |           |          |        |        |             |         |
|----------------------------------------------------------------|-----------|----------|--------|--------|-------------|---------|
| FS                                                             | Used      | Quota    | Used % | Files  | Files Quota | Files % |
| data                                                           | 34.47 GB  | 1.00 TB  | 3.00%  | 28,613 | 409,600     | 6.00%   |
| scratch4                                                       | 695.76 GB | 10.00 TB | 6.00%  | 41,988 | 1,048,576   | 4.00%   |
| scratch16                                                      | 67.92 GB  | 5.00 TB  | 1.00%  | 71,203 | 5,120,000   | 1.00%   |

# MOTD

# Environment Variables

- Print environment variables

```
$ printenv
```

```
$ printenv | grep HOME
```

```
$ echo $PATH
```

# Hidden Files

- Hidden files names start with with a period

```
~/.bashrc
```

```
~/.bash_profile
```

```
~/.bash_history
```

```
~/.bash_logout
```

<https://www.howtogeek.com/435903/what-are-stdin-stdout-and-stderr-on-linux/>



# Basic commands

| ls      | shows files              |
|---------|--------------------------|
| mv      | move or rename files/dir |
| touch   | create a file            |
| mkdir   | create a folder          |
| cp [-r] | copy files or folder     |
| rm [-r] | remove files or folder   |
|         |                          |

| hostname        | shows the name of the node         |
|-----------------|------------------------------------|
| who             | list users on the node             |
| { top -u \$USER | displays a user's processes        |
| htop            |                                    |
| history         | shows the history of your commands |
|                 |                                    |
| { cat           | view files                         |
| more / less     |                                    |
| head / tail     |                                    |
|                 |                                    |

# Basic commands – Con'td

## list files

\$ ls -l    long format

\$ ls -r    reverse order

\$ ls -a    hidden files

\$ ls -t    modification time

\$ ls -h    human-readable

\$ ls -d    folders

\$ ls -alrt   combination

Type `$ man ls` to see all the flags it can take



# Command Redirect Exercises

- `stdin, stdout, stderr`
- `>, >>, |, &`
- `tee` (output to screen and also to a file)
- `command 2>&1 | tee command.log`
- `ls > File11 ; cat File11`
- list all files in Training and 'pipe' them into file File11
- `ls -l >> File11 ; more File11`
- List (long format) all files in Training and append the output (stdout) into File11
- `ls -lR 2>&1 | tee Capture ; more Capture`
- list all directories (recursively) and display stdout to the screen and into a file

# File Permissions

```
$ ls -l
drwxr-xr-x. 4 root root 68 Jun 13 20:25 tuned
-rw-r--r--. 1 root root 4017 Feb 24 2022 vimrc
```



file permissions

d is a directory

— is a file

ℓ is a soft link



owner group



Time of the last  
modification

# File Permissions

| Permission  | Meaning for files                     | Meaning for directories                               |
|-------------|---------------------------------------|-------------------------------------------------------|
| read (r)    | Contents of the file can be displayed | Contents of directory can be listed                   |
| write (w)   | File can be modified or deleted       | Files can be created in or deleted from directory     |
| execute (x) | File can be run like a program        | Directory can be entered (i.e., the cd command works) |

- u=user, g=group, o=others
- r=read=4, w=write=2, x=execute=1
- `$ chmod g+r <file>`
- `$ chmod -w <file>`
- `$ chmod 755 <file>`



# Exercises

1. Check the hostname Type “hostname”
  2. Type “pwd”
  3. Go back to \$HOME dir
  4. List (long format) all files and directories in your HOME dir.
  5. Redirect the output to a file
  6. Check if the remove command is set to ask for removal
  7. Type “man rm”
1. Create a directory called “Junk2021”
  2. List permissions for Junk2021
  3. Change permissions to User RWX only
  4. Change directory to Junk2021

# Text Editor - Nano

\$ nano

\$ nano filename.txt

## nano's shortcuts

Ctrl+S Save current file

Ctrl+O Offer to write file ("Save as")

Ctrl+X Close buffer, exit from nano

```
#!/bin/bash
#SBATCH --job-name=Simple
#SBATCH --nodes=1
#SBATCH -t 30
#SBATCH --ntasks-per-node=1
#SBATCH --partition=defq
#SBATCH --reservation=Training ### Just for training sessions
#SBATCH --mail-type=end
#SBATCH --mail-user=jcombar1@jhu.edu

source /data/apps/go.sh #### for safety reasons

nl #### find out what modules are loaded

hostname #### type the name of the compute node

sleep 300 ### sleep for 120 to check job
echo "This is it, I am leaving node, Job completed"
```

|             |              |             |               |              |               |
|-------------|--------------|-------------|---------------|--------------|---------------|
| ^G Get Help | ^O Write Out | ^W Where Is | ^K Cut Text   | ^J Justify   | ^C Cur Pos    |
| ^X Exit     | ^R Read File | ^_ Replace  | ^U Uncut Text | ^T To Linter | ^_ Go To Line |

# Bash Scripts

- Anything you can run normally on the command line can be put into a script and it will do exactly the same thing.
- Create a simple bash script

```
myscript.sh
1. #!/bin/bash
2. # A sample Bash script, by Ryan
3.
4. echo Hello World!
```

- How do you run a bash script, e.g. myscript.sh?

\$ ./myscript.sh

\$ bash myscript.sh



# Exercises

1. Create a file called “exercise1.txt”
2. Find permissions for “exercise1.txt”
3. Change permissions for GW for read and execute
4. Edit exercise1.txt (use nano or vi)
5. Add; echo “This is my first script”
6. Execute the script “./exercise1.txt
7. 11. cd ; Remove dir Junk2021



Questions?

THANK YOU.

[help@rockfish.jhu.edu](mailto:help@rockfish.jhu.edu)