

# Linux Shell

*DS 5110/CS 5501: Big Data Systems*

*Spring 2024*

Lecture 2a

Yue Cheng



Some material taken/derived from:

- Wisconsin CS 544 by Tyler Caraza-Harter.

@ 2024 released for use under a [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

# Learning objectives

- Setting up an EC2 VM instance via AWS Academy
- Navigate a Linux file system
- Operating within a Linux shell
- Automate repeated tasks

# What is a Shell?

## A shell program

**Loop:**

```
print(SOME PROMPT)
cmd = get_input()
run(cmd)
```




# If you can type it, you can automate it

## A shell program

Loop:

```
print(SOME PROMPT)
cmd = get_input()
run(cmd)
```



## Script.sh

Shell commands

...  
...

# You can run a shell inside of a shell

The image shows a terminal window titled "bash" with a window icon in the top-left corner and a tab icon in the top-right corner. The terminal content is as follows:

```
Yue ~ $ echo hello world
hello world
Yue ~ $ pwd
/Users/yue
Yue ~ $ bash

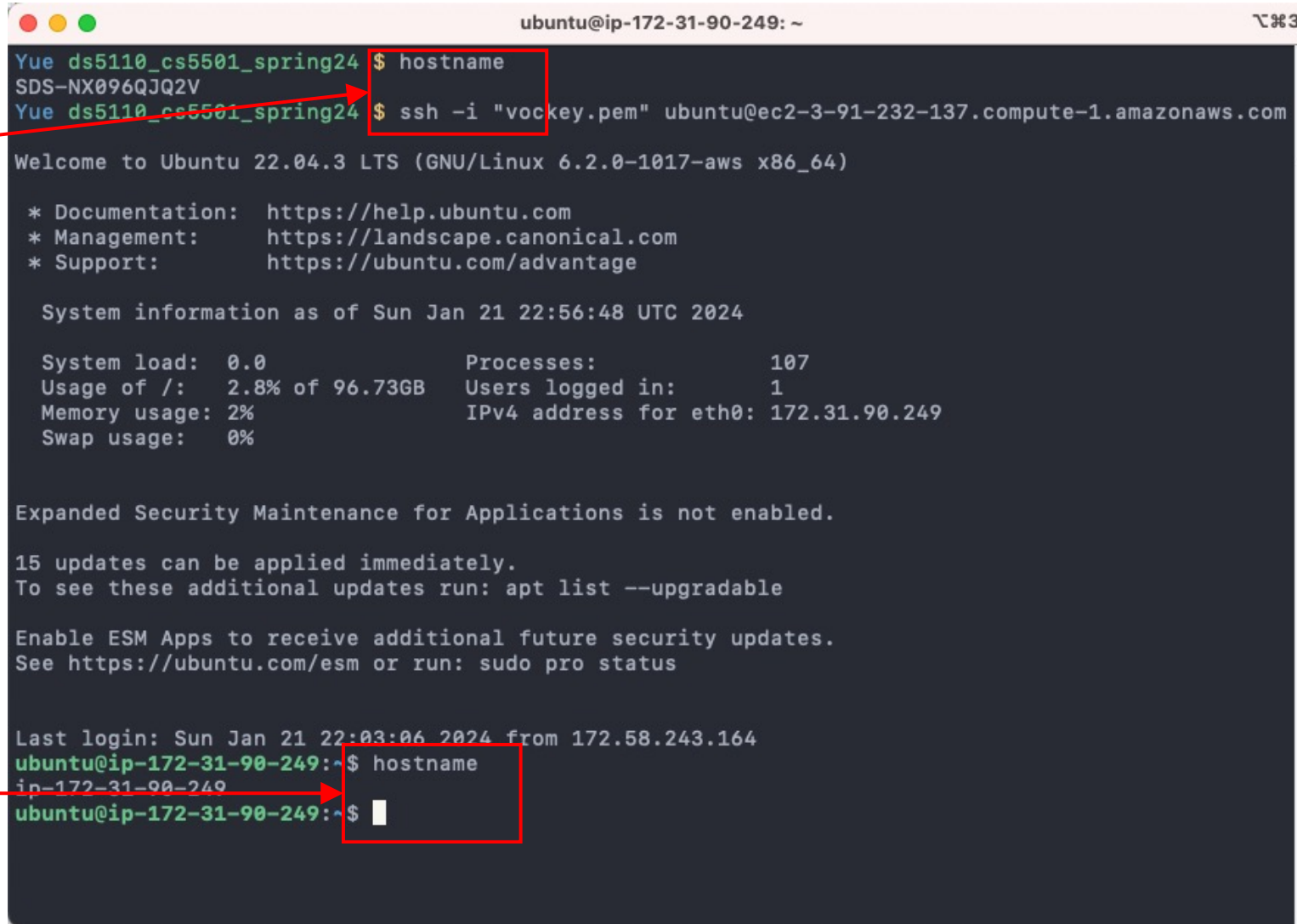
The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh
-s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
bash-3.2$
```

Annotations on the left side of the terminal window:

- zsh prompt**: Three red arrows point to the "Yue ~ \$" prompt on the first, second, and third lines of the terminal output.
- Start running bash**: A blue arrow points to the "bash" command on the third line of the terminal output.
- bash prompt**: A red arrow points to the "bash-3.2\$" prompt on the final line of the terminal output.

# SSH: Secure shell

Running on  
my laptop



The terminal window shows a user named 'Yue' on a laptop with IP 'ds5110\_cs5501\_spring24' and SDS-ID 'NX096QJQ2V'. They execute the command `ssh -i "vockey.pem" ubuntu@ec2-3-91-232-137.compute-1.amazonaws.com`. The terminal output shows the Ubuntu 22.04.3 LTS login banner, system information, and a successful login as 'ubuntu' on IP '172.31.90.249'. A red box highlights the command and the resulting prompt.

```
ubuntu@ip-172-31-90-249: ~  
Yue ds5110_cs5501_spring24 $ ssh -i "vockey.pem" ubuntu@ec2-3-91-232-137.compute-1.amazonaws.com  
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-1017-aws x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:       https://ubuntu.com/advantage  
  
System information as of Sun Jan 21 22:56:48 UTC 2024  
  
System load:  0.0                Processes:            107  
Usage of /:   2.8% of 96.73GB    Users logged in:     1  
Memory usage: 2%                IPv4 address for eth0: 172.31.90.249  
Swap usage:   0%  
  
Expanded Security Maintenance for Applications is not enabled.  
  
15 updates can be applied immediately.  
To see these additional updates run: apt list --upgradable  
  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
Last login: Sun Jan 21 22:03:06 2024 from 172.58.243.164  
ubuntu@ip-172-31-90-249:~$  
ubuntu@ip-172-31-90-249:~$
```

Running on  
my EC2  
VM

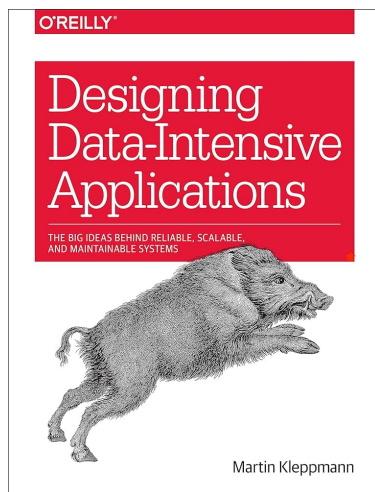
# Announcement

- A0 is out
  - Due by 11am ET, Wed, Jan 31

# Linux pipe

## Unix philosophy

- “Make each program do one thing well. To do a new job, build afresh rather than complicate old programs by adding new ‘features’.”
- “Expect the output of every program to become the input of another, as yet unknown, program. Don’t clutter output with extraneous information. Avoid stringently columnar or binary input formats. Don’t insist on interactive input.”



\* Designing Data-Intensive Applications (“Batch Processing with Unix Tools” of Chapter 10)




# Linux pipe

## Simple Log Analysis

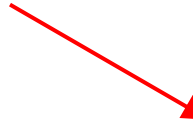
Various tools can take these log files and produce pretty reports about your website traffic, but for the sake of exercise, let's build our own, using basic Unix tools. For example, say you want to find the five most popular pages on your website. You can do this in a Unix shell as follows:

```
cat /var/log/nginx/access.log | ❶  
awk '{print $7}' | ❷  
sort | ❸  
uniq -c | ❹  
sort -r -n | ❺  
head -n 5 | ❻
```



The pipe | connects output of one process to input of the next.

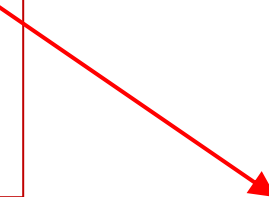
# Standard input and output (I/O)



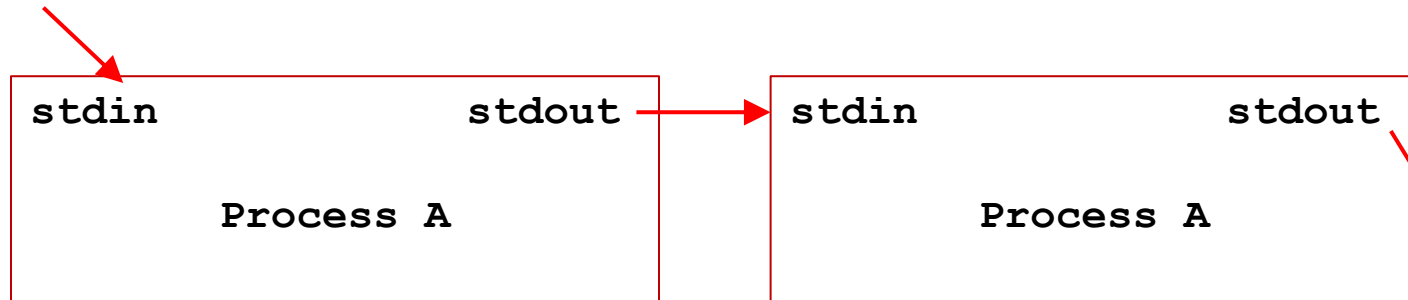
**stdin**

**stdout**

**Process A**



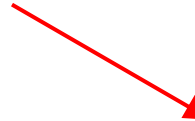
# stdout => stdin



Command  
**A | B**



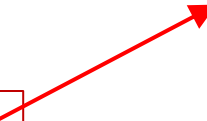
# Redirection



**stdin**

**stdout**

**Process A**



**output.txt**

Command  
**A > output.txt**



# Demos ...