NSSA 220
Task Automation with Interpreted Languages

Linux System Administration

Instructor: Dr. Fahed Jubair

RIT DUBAI

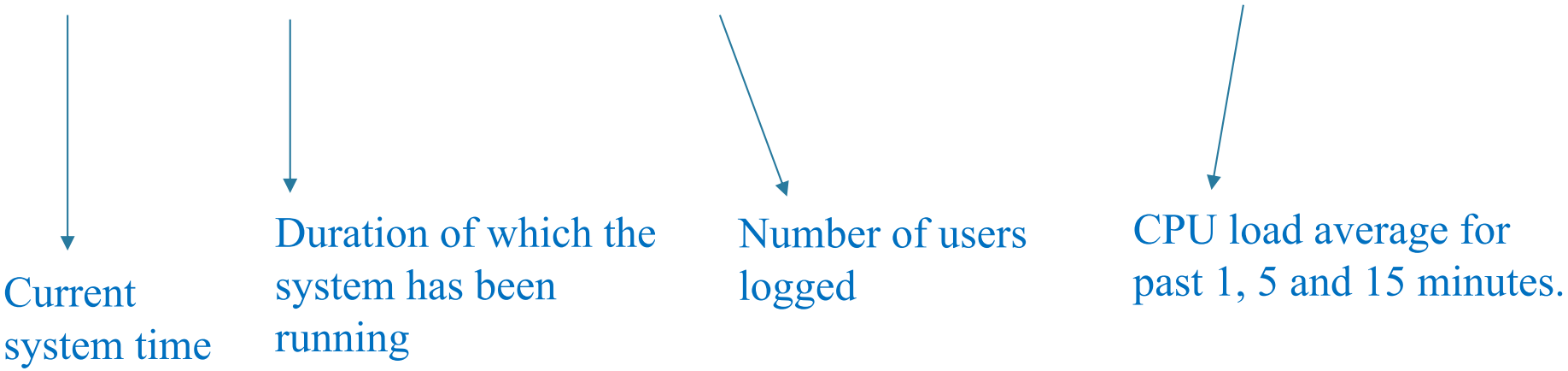
System Administration

- A system administrator is responsible for the upkeep, configuration, and reliable operation of computer systems, especially multi-user computers, such as servers [wikipedia].
- A linux system administrator relies heavily on the terminal, and therefore he/she needs to master Linux commands that are useful for this purpose
- This lecture will cover a subset of Linux commands used for system administration and monitoring

uptime command

- The uptime command shows information about how long the system has been running

```
$ uptime
22:43  up 1 day,  5:01, 2 users, load averages: 2.84 3.34 3.60
```

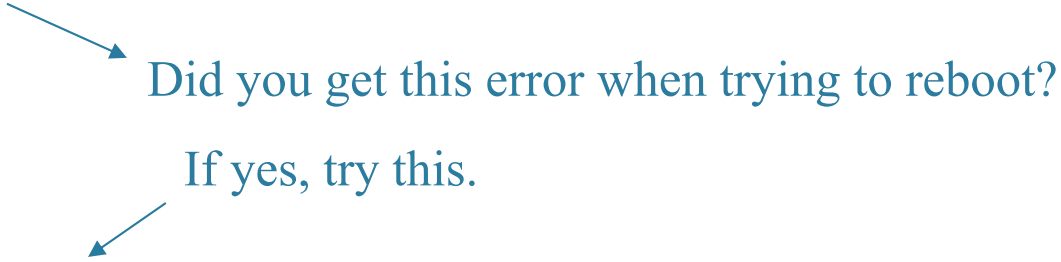


shutdown command

```
$ shutdown 3              # shutdown after 3 minutes
$ shutdown now            # shutdown immediately
$ reboot                  # reboot immediately
$ shutdown –r             # reboot immediately
$ shutdown –r 3           # reboot after 3 minutes
```

sudo command

```
$ reboot
reboot: Operation not permitted
```




```
$ sudo reboot
```

- sudo command is used to gain temporary super privileges without the need to login as a root user

apt command

- apt command stands for advanced package tool
- Used for updating, installing, and removing packages
- Available in Ubuntu and Debian-based Linux distributions

```
$ sudo apt update                          # update all packages
$ sudo apt install <package_name>          # install the specified package
$ sudo apt remove <package_name>           # remove the specified package
$ sudo apt list                            # show available packages
```

who command

- Used to show names of logged in users
- Try it!

```
$ who
```

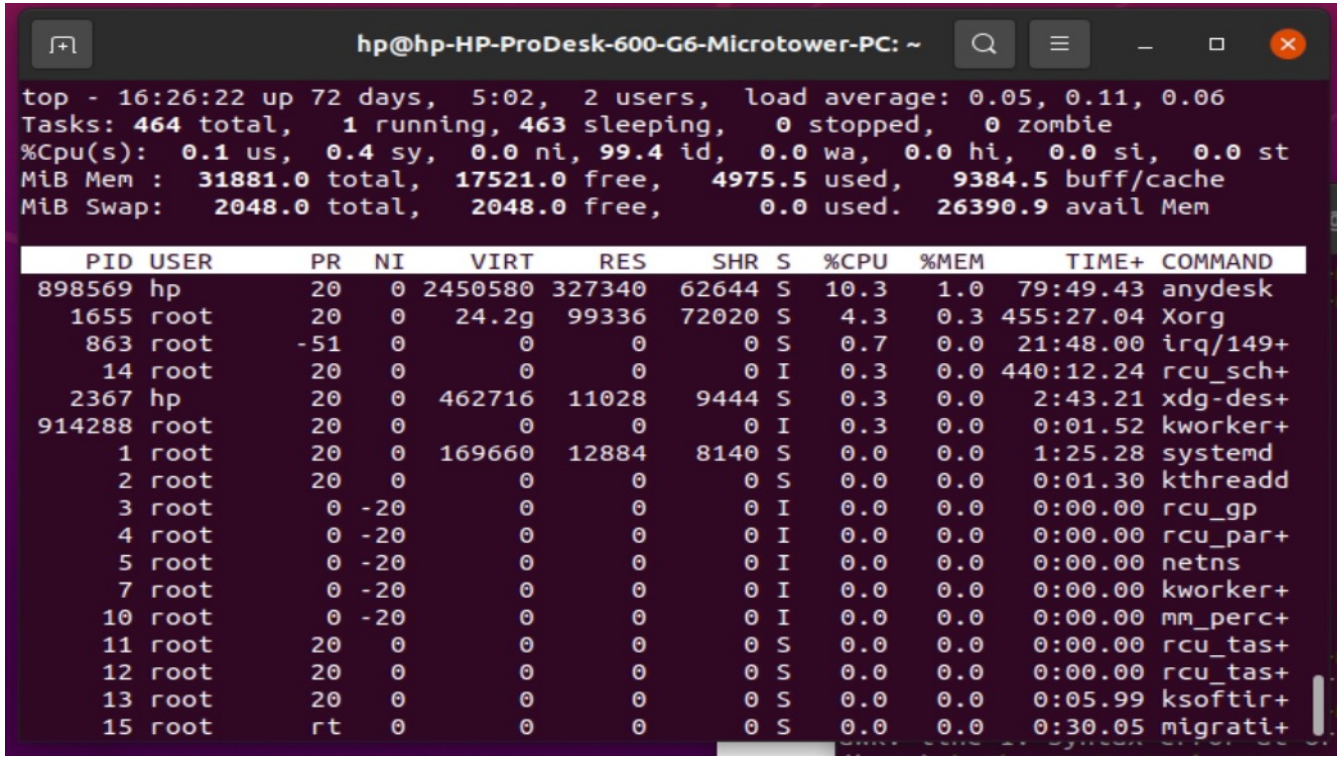
Processes in Linux

- A process is an instance of a running program
- When you execute a command, you are creating a process
- When you run a bash script, you are creating a process
- When you run any application on your machine, you are creating a process
- In Linux, each process is given a unique integer id
- Each process has a priority (represented as an integer)
- Each process has a state (running, sleeping, interrupted, stopped)

top command

- Used to show running process information, including process id, cpu utilization, memory utilization, priority, etc.

```
$ top
```



ps command

- Another command for showing process information

ps aux	Print process information in full format
ps ax	Print process information
ps –u <username>	Print process information for specific user
ps –p <PID>	Print process information for specific process id
ps aux --sort pcpu	Print process information in full format sorted by cpu usage in ascending order
ps aux --sort -pcpu	Print process information in full format sorted by cpu usage in descending order
ps aux --sort pmem	Print process information in full format sorted by memory usage in ascending order
ps aux --sort -pmem	Print process information in full format sorted by memory usage in descending order

kill command

- kill command sends a signal to a specified process.
- Examples:

```
$ kill -9  11132        # sends a kill signal to PID=11132
$ kill -15 11132        # sends a terminate signal to PID=11132
$ kill -0  11132        # sends a null signal to PID=11132
$ kill -2 11132         # sends an interrupt signal to PID=11132
```

- To see available signals, execute the following command:

```
$ kill -l
```

free command

- Used to display information about used and unused memory usage

```
$ free              # display memory usage (in KB, by default)
$ free -g           # display memory usage in GB
$ free -m           # display memory usage in MB
$ free -t           # display total memory
```

df command

- Used to display information about used and free disk space

```
$ df                    # display information for all mounted drives
$ df –h                 # display information in human readable form
$ df –h /home/fjubair   # display information for the given path
$ df –T /               # display type
```

mount and unmount commands

- Used to attach the filesystem of an external device to the filesystem of a system
- Conversely, the *unmount* command will detach an external file system from the local file system

```
$ mount /dev/sda4 /media/    # attach device /dev/sda4 to the path /media
$ unmount /dev/sda4          # detach device /dev/sda4
```

Running in the Background

- Use the ampersand command to run a process in the background
- For example, we can run Firefox in the background like this:

```
$ firefox &
```

- To inquire about currently running processes in the background, use the below command:

```
$ jobs
```

Exercise 1

- Write a bash script that collects the average cpu usage and memory usage for every five seconds in the last 5 minutes, and store this information in a csv file called log.csv

How about we make this exercise a question in Assignment 4?

Exercise 2

- Write a bash script that prints the process ID for the top five processes with highest CPU usage

```
#!/bash/bin

ps aux --sort -pcpu | head -6 | awk '{print $2}' | tail -5
```

Exercise 3

- Write a bash script that checks all processes every five seconds and and kills any process that consumes more than 50% of CPU. Your bash script should run in the background indefinitely.

```
#!/bash/bin

while true
do
    sleep 5
    p=$(ps aux --sort -pcpu | awk '$3 > 50.0 {print $2}')
    for e in $p
    do
       kill -9 $e
   done
done
```

Exercise 4

- Write bash script that runs as background process and shutdown the system after 10:00PM

```
#!/bash/bin

sudo shutdown 22:00
```

wget command

- Used to download files from the internet
- supports HTTP, HTTPs and FTP protocols

```
$ wget <URL>                          # download file
$ wget –o <filename>  <URL>           # store with a different file name
```

ssh command

- Used to connect to a remove server or machine
- Stands for “secure shell”, i.e., ssh uses secure communication protocol
- By default, ssh runs at TCP/IP port 22

```
$ ssh admin@grace.rit.edu
$ ssh admin@129.21.1.40
```

scp command

- Used to transfer files securely between two machines
- scp uses ssh protocol
- Example: the below command will transfer a.txt from the local machine to a remote machine

```
$ scp /home/fjubair/a.txt admin@grace.rit.edu:/home/admin
```

- Example: the below command will transfer a.txt from a remote machine to the local machine nd store it with a different file name

```
$ scp admin@grace.rit.edu:/home/admin/a.txt /home/fjubair/b.txt
```

ifconfig command

- Used to show information about network interfaces such as interface name, IP address, and MAC address
- Also used to configure, enable or disable a network interface

```
$ ifconfig –a              # show information for all network interfaces
$ ifconfig eno1            # show information for ‘eno1’ network interface
$ ifconfig eno1 up         # activate ‘eno1’ network interface
$ ifconfig eno1 down       # deactivate ‘eno1’ network interface
```

- Note: if you get an error “ifconfig command is not found”, then you need to install net-tools package using this command:

```
$ sudo apt install net-tools
```

traceroute command

- Used to measure delays in the network
- Execute the below command to be able to use the command

```
$ sudo apt install inetutils-traceroute
```

- Now, try this command:

```
$ traceroute google.com
```

More Networking commands

```
$ ping google.com          # check for network connectivity
$ netstat                  # show network statistics
$ ss                       # alternative for netstat command
$ route                    # show routing table information
$ host google.com          # show IP address information for a host
$ hostname                 # show local hostname
```

System Configuration Files

- Files that control user permissions, system applications, daemons, and services in a machine
- Usually stored under the /etc folder
- Below are examples of system configuration files:

/etc/hosts	Contains a list of known hosts
/etc/fstab	Contains a list of currently mountable filesystems
/etc/networks	Contains a list of accessible networks to the machine
/etc/groups	Contains groups and users included in each group
/etc/passwd	Contains user information
/etc/shells	Contains available shells to the system

User Configuration Files

- Configuration files that are specific to a user
- Usually stored in the home directory as hidden files (i.e., start with dot)
- Below are examples of user configuration files:

~/.bash_history	List of all previously executed commands
~/.bash_login	A script that runs when the user logins
~/.bash_logout	A script that runs when the user logs out
~/.mail.rc	Init script for mail program

Summary

- System administration is the process of setting up, configuring, managing, and troubleshooting a machine (typically, a server)
- Linux system administrators need to comfortable doing everything from the command line and writing bash scripts
- This lecture is an introduction to system administration in Linux
- If you find system administration to be interesting, note that there is a demand in the market for skilled system administrators