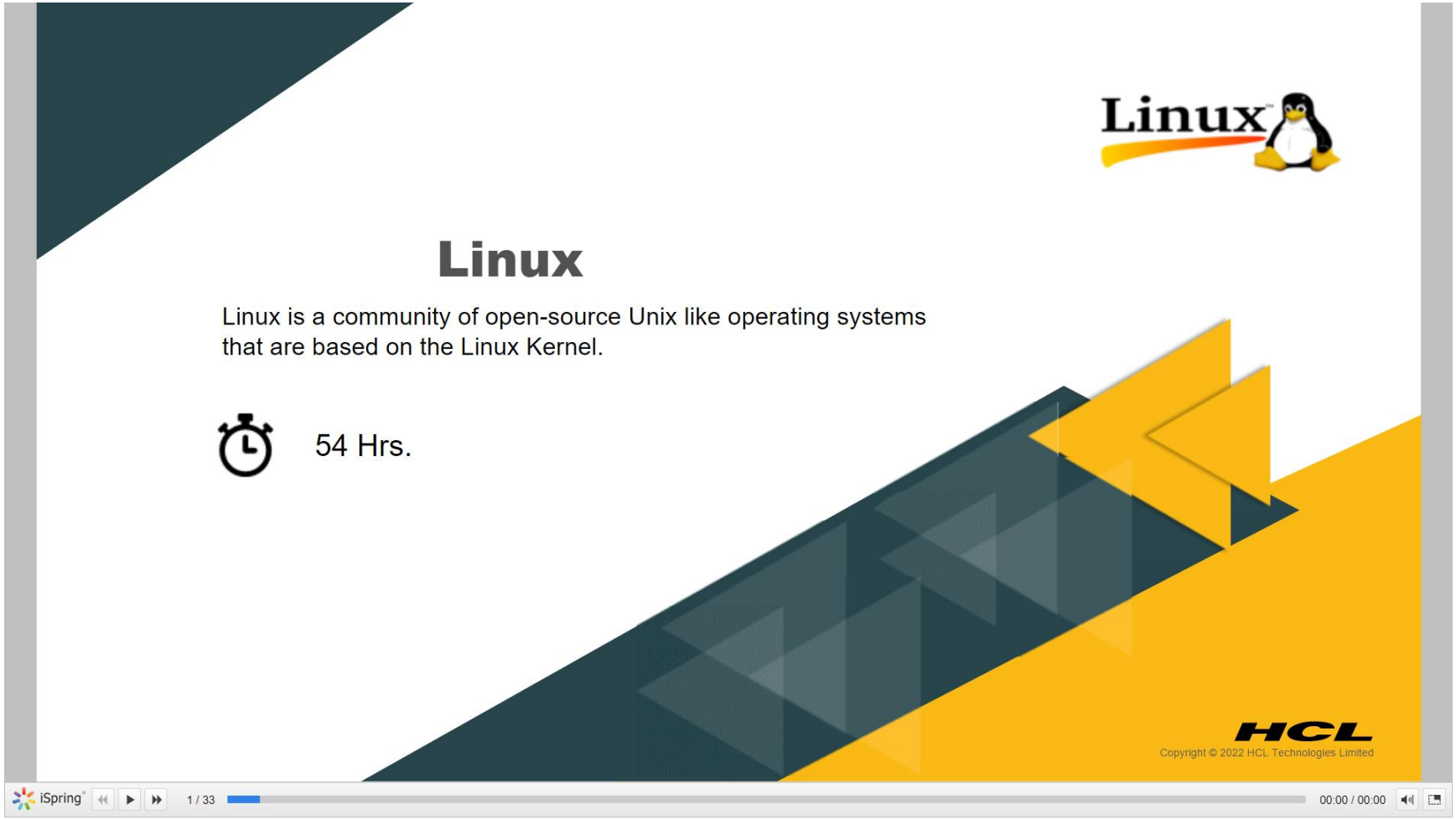
Linux
Linux
Linux is a community of open-source Unix like operating systems that are based on the Linux Kernel.
54 Hrs.
HCL
Copyright © 2022 HCL Technologies Limited
iSpring
1 / 33
00:00 / 00:00

Module Duration

54 hrs.

S. No.	Topic	Duration (in hrs.)
1	Linux History And Introduction	6
2	Linux-Basics	6
3	Vim Editor	6
4	Linux Commands	6
5	File Permissions	6
6	Process Control	6
7	UNIX	6
8	Unix shell scripting	12

** Note: 15 hours of additional hands-on lab practice*

HCL

Learning Objectives

This Topic provides an introduction to LINUX History And Introduction

Learner completing the **course** would know:

- Linux History
- Linux Basics
- VIM Editor
- Linux commands
- File permissions
- Process control
- UNIX Operating System
- Shell Scripting

HCL

Topic Coverage

Topics
Processes
Controlling Jobs
Killing Processes
Monitoring Process Activity

12 Hours

HCL

What is a Process?

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- A process in an executing program with several components and properties.
- A process is a shell command or a program in execution.
- When you log in, a process is created. This process is executing the shell
- When you execute a shell command, a new process is created. When the command terminates, the process dies.
- A single user can have many processes executing at the same time

HCL

ps command

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

•The command ***ps*** is used to determine the status of active processes. The command returns the process id (PID) number and other information such as the amount of CPU time the process has used (TIME) and the command which invoked the process (CMD).

•Options may be combined.

•a includes processes on all terminal

•x includes processes not attached to terminal

•u vikas prints process owner information

•o property1, property2 …. Where property is pid, comm, %cpu, %mem, state, tty, euser, ruser

•E.g.: ps axo pid,%cpu,comm

HCL

Process State

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- Every process has a state property.
  - This describes whether the process is actively using the cpu, in memory but not doing anything (sleep), waiting for a resource to become available (uninterruptable sleep) or terminated, but not flushed from the process list (Zombie).
- Uninterruptable sleep:
  - Process is sleeping and cannot be woken up until an event occurs. It can not be woken up by a signal. Typically, the result of an I/O operation, such as a failed network connection (for NFS hard mounts).

- Just before a process dies, it sends a signal to its parent and waits for an acknowledgement before terminating. Even if the parent process does not immediately acknowledge this signal, all resources except for the process identity number (PID) are released. These are called ZOMBIE PROCESSES and are cleared from the system during the next system reboot and do not adversely affect system performance.

HCL

Finding Processes

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- Lookup for processes
- pgrep –U root
- pgrep –G student
- -U

  Only match processes whose real user id is listed.
- -G

  Only match processes who's real groupid is listed.
- Find a process ID of a running program.
  - pidof gedit
  - pgrep gedit

HCL

signals

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

•Signals are simple messages that can be communicated to process with commands like kill.

•Signal 15, TERM (default)	-	Terminate cleanly
•Signal 9, KILL	-	Terminate immediately
•Signal 1, HUP	-	Re-read configuration files

•Most services are programmed to reload their configuration when they received a HUP.

HCL

Sending Signals to Processes

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

•Kill can send many signals, but processes only respond to the signals they have been programmed to recognize.

- •kill 3428
- •kill -15 3428
- •kill –TERM 3428

•User pidof gedit or pgrep gedit can be used to know the process id of the process, gedit in this case.

•killall – kill process by name

- •killall gedit

HCL

Scheduling priority

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- Scheduling priority determines access to the CPU.
- Values ranges from -20(highest) to 19(lowest), default is 0.
- nice - run a program with modified scheduling priority.
  - nice -n 15 myprog
- Where –n specifies the priority number.
- non-privileged users may not set niceness value to less than zero.

Scheduling priority

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- **renice** - alter priority of running processes
- only the superuser is permitted to raise the priority of currently running process.
- Non-super-users can not increase scheduling priorities of their own processes, even if they were the ones that decreased the priorities in the first place.
- can be used to modify the priority of all of the processes of a particular user.
  - renice -15 -u joe
- Can be used to modify the priority of a process
  - renice -15 –p pid

HCL

Process Management Tools

Processes

Controlling Jobs

Killing Processes

Monitoring Process Activity

- CLI: top command
- GUI: gnome-system-monitor
- Capability:
  - Display real-time process information
  - Allow sorting, killing and renicing.

HCL

Job Control

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- When a process is started from a terminal's command line, it is normally running in the foreground.
- When a command is entered, the shell will not process further input until the process is completed and the shell prompt is redisplayed.
- The typeahead buffer allows you to type other commands, but they will not be processed until the pending process completes, or “returns”.
- Running a command in the background allows another process to run concurrently on the same terminal.
- A background process is still the child of the processes that spawned it.
- The parent process, however, does not wait for the child process to terminate before continuing.
- When a process is started in the background, a new bash “sub-shell” is created.

HCL

Job Control

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- Both find and ps now have the same PPID, the pid of the shell. the shell has spawned two processes, one running in the foreground, and the other in the background.
  - [root@server1 ~]# find / -name abc.txt &
  - [1] 3600
  - [root@server1 ~]# ps -f
  - UID PID PPID C STIME TTY TIME CMD
  - root 3314 3302 0 16:46 pts/2 00:00:00 bash
  - root 3600 3314 3 17:36 pts/2 00:00:00 find / -name abc.txt
  - root 3601 3314 0 17:36 pts/2 00:00:00 ps -f

HCL

Job Control

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- You can run as many jobs in the background as the system load permits.
- Make sure that both standard output and standard error are redirected suitable, using /dev/null, if necessary.
- shell dies on logout, but not its child.
- Kernel reassign the PPID of the find process to the process that has a PID 1.
- This is the system process init, which is the parent of all shells.
- when the user logs out, init takes over the parentage of the find process.

HCL

Suspending a Process

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- Below command check the process “find”.
  - ps -x | grep find
- Foreground jobs can be suspended: temporarily stopped, without being killed, using the <ctrl-z> keystroke.
- once a process is suspended, it can be resumed in the background, using the bg command, or resumed in the foreground, using the fg command.
- job numbers are referenced with %.
- Syntax
  - fg [%job_number]
  - bg [%job_number]

HCL

Listing Background and Suspended Jobs

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- Jobs displays all process running in the background or that are suspended.
- the number in brackets is a job number, used to kill jobs or bring them back to the foreground.
  - [root@server1 ~]# jobs
  - [1]+ Stopped find / -name abc.txt

HCL

Scheduling a process to execute later

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- at time-executes commands at a specified time.
  - [root@server1 ~]# at 11:56 am
    at> find / -name abc.txt >>findresult.txt
    at> <EOT> #press <ctrl-d>
    job 3 at 2005-10-05 11:56
    [root@server1 ~]#
- at –l list
- at –c jobnum show details
- at –d jobnum Delete the job having job number jobnum

HCL

crontab

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- cron file settings are:
- min hour day_of_month month day_of_week command

- min : 0-59
- hour : 0-23 ;0 means 12AM
- day_of_month : 1-31
- month : 1-12 or Jan, Feb, ;not case sensitive
- day_of_week : 0-7 ;0 & 7-Sunday, 1-Monday, not case sensitive

- * : every
- Range are separated by e.g. 8-11 hours
- Non-continuous range are separated by , e.g. 8,9,11 hours
- Mixed range are as e.g.: 0-4,8-12 hours
- 0-23/2 at hour position means every other hour i.e. 0,2,4,6,8,10,12,14,16,18
- */2 at hour position means every two hour

HCL

crontab

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- If there is a mismatch as below: 30 4 1,15 * 5
- Will run command at 4:30AM on the 1st and 15th of each month, plus every Friday.
- Pound-sign (#) are comments
- Job files created by each user is stored in /var/spool/cron which is not accessible by non privileged users.
- E.g.:
  - * * * * * touch > /root/hello.txt

HCL

crontab

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- Cron job of user sunil can be Created/Edit cron by root using -e switch:
- crontab -u sunil -e ;Creating/Edit cron for user sunil by root
- crontab -e ;Creating/Edit cron by user himself
- **View cron jobs**: Cron jobs can be viewed by using l switch as:
- crontab -lu sunil ;View cron jobs of user sunil by root
- crontab -l ;View cron jobs for himself
- **To remove crontab**: cron jobs can be removed using r switch as:
- crontab -ru sunil ;remove crontab job of user sunil by root
- crontab -r ;To remove crontab job by user himself
- Job files created by each user is stored in /var/spool/cron which is not accessible by non privileged users

HCL

System crontab files

- /etc/crontab is master crontab file which runs executables in
  - /etc/cron.hourly
  - /etc/cron.daily
  - /etc/cron.weekly
  - /etc/cron.monthly

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

- /etc/crontab and /etc/cron.d/ are different from user crontabs.
- Here sixth field is a username which will be used to execute the command in the seventh field.

HCL

cron access control

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- If neither /etc/cron.allow nor /etc/cron.deny exists only root is allowed to install new crontab files.
- If cron.allow does not exist, all users listed in cron.deny are not allowed to use cron.
- If the file cron.allow exists, only users listed in it are allowed to use cron, and the cron.deny file is ignored.
- The format of both access control files is one username on each line. Whitespace is not permitted in either file.
- The cron daemon (crond) does not have to be restarted if the access control files are modified.
- The access control files are read each time a user tries to add or delete a cron task.
- Note that denying a user through the user of the above files does not disable their installed crontab.

HCL

The anacron system

- The anacron run cron jobs that did not run when the computer is down
- Configuration file: /etc/anacrontab

```
SHELL=/bin/sh
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root

1       65      cron.daily              run-parts /etc/cron.daily
7       70      cron.weekly             run-parts /etc/cron.weekly
30      75      cron.monthly            run-parts /etc/cron.monthly
```

- Field 1: if the job has not been run in this many days
- Field 2: wait this number of minutes after reboot and then run it.
- Field 3: job identifier
- Field 4: The job to run.

HCL

Grouping Command

- Processes
- Controlling Jobs
- Killing Processes
- Monitoring Process Activity

- Suppose you want to maintain a count of the number of users logged on, along with a time /date stamp, in the log file.
  - date >> logfile
  - who | wc –l >> logfile
- This can also be done as:
  - (date; who | wc –l) >> logfile
- Command inside parentheses are run in their own instance of bash, called a SUBSHELL. The output of all commands run inside a subshell are sent to the subshell's STDOUT and STDERR, making it possible to send multiple programs through the same pipe.

HCL

1. What is a Process?

a) process is executing the shell

b) A process in an executing program with several components and properties

c) When you execute a shell command, a new process is created

d) All of the above

2. Signal 9 represents

a) KILL

b) TERM

c) HUP

d) none of the mentioned

HCL

3. which command is used to run a program with modified scheduling priority

a) modify

b) crontab

c) nice

d) None of the above

4.Which commands used to represent foreground and background PROCESSES

a) TERM

b) fg,bg

c) kill

d) none of the mentioned

HCL

Re-Cap

Discussion Point

- What is Process?
- Controlling jobs
- Killing process
- Monitoring process activity

Instructions & Duration

- The participants will discuss based on their understanding of Process.

  clarify their doubts.
- One of the scholar could be the facilitator along with the trainer.

5 min.

HCL

Cue Card for Assimilation Check

Question Number	Correct Answer	Slide Number
1	d	Slide 6
2	a	Slide 10
3	c	Slide 12
4	b	Slide 18

References

- https://www.guru99.com/introduction-linux.html
- https://www.tutorialspoint.com/operating_system/os_linux.htm

HCL


HCL
www.hcltech.com
$11.18 BILLION | 197,000+ IDEAPRENEURS | 52 COUNTRIES
iSpring
33 / 33
00:00 / 00:00