







CEIS106 Operating Systems











11:11PM

INTRODUCTION

Table Of Content

NAVIGATE THE LINUX FILESYSTEM TREE

CREATE A SHELL SCRIPT

ADD USERS AND GROUPS IN CLI



Introduction

Welcome to my Linux presentation.

Today, we'll cover key topics such as navigating the Linux filesystem, creating and managing directories and files, and locating files easily.

We'll also learn how to create and manage shell scripts, including changing their permissions and setting the PATH variable permanently. User and group management, both through the command line and GUI, will be discussed, along with how to test these settings.

Additionally, we'll explore network configuration, manage interfaces, and use network utilities.

Lastly, we'll focus on monitoring processes, user activities, and network bandwidth usage to keep the system running smoothly and securely.









Back to Agenda Page





NAVIGATING THE LINUX FILESYSTEM IS KEY TO MANAGING YOUR SYSTEM.

student@ubuntu1:~\$ pwd /home/student

The pwd command: shows your current directory, like a GPS showing your location.

mary@ubuntu1:~/Desktop\$ cd ~

The cd command: lets you move to a new directory, similar to setting a destination on a GPS.

Relative Path is based on your current location in the file system, like giving directions from where you are.

Example:

- directory)
- address from the root directory)

An absolute path starts from the root directory (/) and shows the full path to a file or folder, like giving an exact address.

• Relative path: documents/report.txt (from your current

• Absolute path: /home/user/documents/report.txt (full

CREATE DIRECTORIES AND FILES

Creating directories and files is a fundamental skill in Linux, and it's essential for organizing data.

For example

- mkdir myfolder creates a directory called "myfolder".
- Is -I lists the contents of the current directory, showing "myfolder" as a newly created directory.
- cd myfolder changes the current directory to "myfolder".
- touch myfile.txt creates an empty file named "myfile.txt".



In this example create a folder called "/JanFebSession/Course1"





student@ubuntu1:~/JanFebSession/Course1\$ ls -l ~/JanFebSession/Course1

student@ubuntu1:~/JanFebSession/Course1\$ echo "YANETH GONZALEZ

COPY AND REMOVE DIRECTORIES AND FILES

Managing files and folders is straightforward using commands like *rmdir* for removing empty directories and rm for deleting files.

- For instance, on Ubuntu, you can delete an empty directory named "Course3" in "MarFebSession" with rmdir MarFebSession/Course3.
- To remove a file named "files3" in the "Course1" directory under "MarAprSession", use rm MarAprSession/Course1/files3.
- The tree command helps you visualize the directory structure, allowing you to see changes before and after managing directories and files.





rmdir: failed to remove 'MarFebSession/Course3': No such file or directory



LOCATE DIRECTORIES AND FILES

In Linux, navigating and locating directories and files efficiently is essential for managing your system.

One powerful tool for this task is the locate command.

- you can use locate -i course to find files related to "course."
- To locate directories or files ending with "files1," use locate -r /files1\$.
- These commands help users quickly find data in Linux.





CREATE A SHELL SCRIPT

Mastering shell scripting is essential for automating tasks and managing system configurations effectively. A shell script is a series of commands written in a text file that the shell interprets and executes sequentially.

For example

- you can create a shell script named *backup.sh* to automate the backup of important files to a specified directory.
- This script might use commands like tar to archive files and *cp* to copy them to a backup location.

CHANGE SCRIPT FILE PERMISSIONS

In Linux, changing script file permissions is simple.

For example:

- If you want to give everyone the ability to read, write, and execute a file named todolist, you would use the chmod 755 todolist command.
- This command allows full permissions to the owner of the file (7), and read and execute permissions to group members and others (5).
- It ensures that users can interact with the file according to these permissions.
- Understanding and using chmod effectively is essential for managing file access in Linux.

F

command 'pmw' from deb pmw (1:4.30-1) command 'psw' from deb wise (2.4.1-21) command 'pda' from deb speech-tools (1:2.5.0-8build1) command 'pwd' from deb coreutils (8.30-3ubuntu2)

Try: sudo apt install <deb name>

student@ubuntu1:~\$ pwd /home/student student@ubuntu1:~\$ nano todolist student@ubuntu1:~\$ chmod 755 todolist student@ubuntu1:~S ls -l todolist -rwxr-xr-x 1 student student 202 May 15 13:05 todolist student@ubuntu1:~\$./todolist Enter todays's to-do-list (Press ENTER to complete): 1. work 2. family 3. school You entered: 1. work 2. family 3. school student@ubuntu1:~\$ echo "Yaneth Gonzalez > ^C student@ubuntu1:~\$ echo "Yaneth Gonzalez" Yaneth Gonzalez student@ubuntu1:~S date Wed 15 May 2024 01:10:40 PM EDT student@ubuntu1:~\$



SET THE PATH VARIABLE

Linux is widely used and known for its security and flexibility. You can tell Linux where to find programs by setting the PATH variable.

- For instance, to add a directory named "/home/student" to the PATH variable, use:
- **\$ PATH=\$PATH:/home/student**
- This command helps Linux locate and run programs stored in "/home/student" when you type their names in the terminal.
- Understanding and using the PATH variable is important for working efficiently in Linux.

student@ubuntu1:~\$ pwd /home/student student@ubuntu1:~\$ todolist todolist: command not found student@ubuntu1:~\$ echo \$PATH al/games:/snap/bin student@ubuntu1:~\$ PATH=\$PATH:/home/student <-</pre> student@ubuntu1:~\$ echo \$PATH al/games:/snap/bin:/home/student student@ubuntu1:~\$ todolist Enter todays's to-do-list (Press ENTER to complete): school 2. work 3. family You entered: 1. school 2. work 3. family student@ubuntu1:~\$ echo "Yaneth Gonzalez" Yaneth Gonzalez student@ubuntu1:~S date Wed 15 May 2024 01:18:30 PM EDT student@ubuntu1:~\$





MAKE THE PATH VARIABLE PERMANENT

To permanently set the PATH variable in Linux, edit a file that sets environment variables each time you log in.

- This ensures your custom commands and scripts are always available.
- For example, you can add a line like this to your .bashrc file:
- export
 PATH=\$PATH:/path/to/your/directory
- Save the file, and from then on, whenever you open a terminal or log in, Linux will include /path/to/your/directory in the PATH variable.







		-			
ent@ubi	untu1:~	- ۵ =		• •	
sbin:/u student	sr/bin:/sbin:/b	in:/usr/ga	mes:/u	ısr/loc	
so cir. 70	л			stude	nt@ubuntu1: ~
ER to c mily alez"	student@ubuntu1 student@ubuntu1 Enter todays's 1. family 2. so You entered: 1. student@ubuntu1	<pre>l:~/Desktop L:~\$ todoli to-do-list chool 3. wo family 2. L:~\$ echo '</pre>	\$ cd ist t (Pres ork schoo 'YANET	~ SS ENTE ol 3. w H GONZA	R to complete): ork LEZ"
1t@ubun	student@ubuntu3 DATE: command r student@ubuntu3 Wed 15 May 2024 student@ubuntu3	L:~\$ DATE not found L:~\$ date 4 01:30:46 L:~\$	PM ED	т	
old	k				

ADD USERS AND GROUPS IN CLI

Adding users and groups using the command line is an important skill.

You can use the useradd command to create new users and the groupadd command to create new groups.

For example:

- useradd john adds a user named John.
- groupadd developers creates a group called developers.
- usermod -aG developers john adds John to the developers group.

These commands help manage who can use the system and how they are organized.





Exploring user and group settings through graphical interfaces (GUI) is essential for managing system security and access permissions effectively.

 GUI tools like GNOME System Settings provide intuitive interfaces for adding, modifying, and removing users and groups.

For example

 in GNOME, navigating to "Settings" > "Users" allows you to create new user accounts with specific privileges, change passwords, and assign users to supplementary groups. Examples of two users we created

.F1					
ohn(gub	unti	11:~	/D	esk
ohn(gubi	unti	11:~	\$	pwd
home	e/jo	ohn			
ohn(ğubi	untu	11:~	\$	nar
ohn(ğubi	unti	11:~	\$	sou
ash	: e)	крог	t:	1	hor
ohn(ğubı	untu	11:~	\$	tod
odo	lis	t: (OMM	an	d r
ohn(gubi	unti	11:~	\$	nar
ohn(gubi	untu	11:~	\$	sou
ohn(gubi	untu	11:~	\$	tod
Inter	r to	oday	/'s	to	- do
. fa	ami	ly 2	2. f	am	ily
ou e	ente	erec	: 1		far
ohn(gubi	unti	11:~	\$	cat
led :	22 1	lay	202	4	01:
oday	y's	to-	-do-	li	st

ш	mary@ubuntu1: ~
<pre>mary@ubuntu1:~/Desktop\$ cd ~ mary@ubuntu1:~\$ nano .bashrc mary@ubuntu1:~\$ source .bashrc mary@ubuntu1:~\$ todolist Enter today's to-do-list (Press # 1. school 2. school 3. school You entered: 1. school 2. school You entered: 1. school 2. school mary@ubuntu1:~\$ cat MyToDoLists Wed 22 May 2024 01:34:40 PM EDT Today's to-do-list 1. school 2 mary@ubuntu1:~\$ echo "yaneth gon2 yapath copzalez</pre>	ENTER to complete 3. school 2. school 3. scho zalez"

yaneth gonzalez mary@ubuntu1:~\$ date Wed 22 May 2024 01:36:03 PM EDT mary@ubuntu1:~\$



	john@ubu	ntu1: ~	Q =	-	•	×
\$ cd						
.bashrc bashrc student': ist	not a valid	identifier				
found						

):

.bashrc ce .bashrc

family

MyToDoLists 51:49 PM EDT

list (Press ENTER to complete):

1. family 2. family 3. family

ily 2. family 3. family

olist

bol

REMOVE USERS AND GROUPS

In Linux system administration, it's important to know how to remove users and groups for security and resource management.



To remove a user like "john" and delete their home directory

• use sudo userdel -r john.

For removing groups

 use sudo groupdel developers to delete the group named "developers".

These actions help maintain system organization and security

DISCOVER HOST IP CONFIGURATIONS

Knowing how to discover host IP configurations is essential.

Understanding commands like *ifconfig* and *ip addr* show to view network interfaces and their assigned IP addresses.

For example

- running *ifconfig* lists all network interfaces along with their *IP addresses, while ip addr* show provides detailed information about each interface, including IPv4 and IPv6 addresses.
- This knowledge is crucial for setting up and troubleshooting network connectivity in Linux systems, ensuring smooth communication across networks.



DHCP AND THE RESOLV.CONF FILE.

- DHCP (Dynamic Host Configuration Protocol) gives your computer network settings like IP addresses and DNS servers automatically.
- The /etc/resolv.conf file tells your computer where to find DNS servers, which change website names (like google.com) into IP addresses.
- This file is often managed automatically by the system, maybe with a service called systemd-resolved.

For example

- nameserver 192.168.1.1 in the file means the DNS server is at IP address 192.168.1.1. search devry.edu means that if you type a hostname without a domain, it adds ".devry.edu" automatically.
- Also, using the command ping -c 4 192.168.1.1 sends 4 packets to check if the IP address 192.168.1.1 can be reached. If all packets come back, the connection is good.

F

Third party programs must not access this file directly, but only through the # symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a different way, # replace this symïfink by a static file or a different symlink.

See man:systemd-resolved.service(8) for details about the supported modes of # operation for /etc/resolv.conf.

nameserver 192.168.1.1 search devry.edu

student@ubuntu1:/var/lib/dhcp\$ ping -c 4 192.168.1.1 PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data. 64 bytes from 192.168.1.1: icmp seg=1 ttl=64 time=0.338 ms 64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.264 ms 64 bytes from 192.168.1.1: icmp seg=3 ttl=64 time=0.437 ms 64 bytes from 192.168.1.1: icmp seg=4 ttl=64 time=0.283 ms --- 192.168.1.1 ping statistics ---4 packets transmitted, 4 received, 0% packet loss, time 3078ms rtt min/avg/max/mdev = 0.264/0.330/0.437/0.067 ms student@ubuntu1:/var/lib/dhcp\$ echo "yaneth gonzalez"

vaneth gonzalez

student@ubuntu1:/var/lib/dhcp\$ date Thu 30 May 2024 12:49:50 PM EDT student@ubuntu1:/var/lib/dhcp\$



student@ubuntu1: /var/lib/dhcp

MANAGE NETWORK INTERFACES

Managing network interfaces in Linux involves ensuring your computer can connect to networks smoothly.



these interfaces.

For example

eth0, you would use

ifconfig eth0 up

use

\$ ip addr show

your Linux system.





Back to Agenda Page

USE NETWORK UTILITIES

Understanding network utilities is crucial for managing and troubleshooting networks.

For instance

- The *ping* command tests if devices can communicate by sending packets and receiving responses.
- It works by sending *ICMP* (Internet Control Message Protocol) echo request packets to the target host and waiting for ICMP echo reply packets.

ether 00:15:5d:00:ba:04 txqueuelen 1000 (Ethernet) RX packets 4142 bytes 325948 (325.9 KB) RX errors 0 dropped 0 overruns 0 frame 0 TX packets 6445 bytes 524893 (524.8 KB) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

student@ubuntu1:~\$ sudo ifconfig eth0 up student@ubuntu1:~\$ ifconfig eth0 eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500 ether 00:15:5d:00:ba:04 txqueuelen 1000 (Ethernet) RX packets 4167 bytes 328531 (328.5 KB) RX errors 0 dropped 0 overruns 0 frame 0 TX packets 6513 bytes 532097 (532.0 KB) TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

student@ubuntu1:~\$ ehoc "yaneth gonzalez" 🎚 ehoc: command not found student@ubuntu1:~\$ echo "yaneth gonzalez" yaneth gonzalez student@ubuntu1:~\$ date Thu 30 May 2024 01:12:50 PM EDT



```
Х
```

```
inet 192.168.1.107 netmask 255.255.255.0 broadcast 192.168.1.255
inet6 fe80::7b9e:ebf5:11a6:34e4 prefixlen 64 scopeid 0x20<link>
```

MONITOR LINUX PROCESSES

Monitoring processes in Linux is important for keeping track of what your computer is doing.

Commands like

- *ps* shows a basic list of processes currently running in the terminal.
- top- display showing CPU usage, memory usage, and a list of processes sorted by various criteria.
- *htop*-displays a color-coded and interactive list of processes, allowing you to manage and monitor them more easily
- *ps aux*-view of all processes, their owners, and resource utilization, making it useful for detailed system monitoring and troubleshooting.



Monitor user activities

Monitoring user activities in Linux is essential for system security and performance. This involves tracking what users are doing on the system in real-time. Tools like *w* and *who* show who is logged in and what they are doing, and who logged out *last*

Example:

\$ W					
12:30:0	1 up 5 da	ays, <mark>4:10</mark> ,	3 users,	load av	era
USER	TTY	FROM	LOG	IN@ ID	LE
alice	pts/0	192.168.1.5	08:	50 3:	10m
bob	pts/ <mark>1</mark>	192.168.1.6	09 :	15 2:	<mark>05</mark> m
carol	pts/ <mark>2</mark>	192.168.1.7	10	00 1:	00m
\$ who					
alice	pts	5/0	2023-0	06-12 (98:
bob	pts	5/1	2023-0	06-12 0	99:
carol	pts	5/2	2023-0)6-12 1	10:
\$ las	t				
alice	pt	s/0	192	.168.1	. 5
bob	pt	s/1	192	.168.1	6
carol	pt	s/2	192	.168.1	. 7





Back to Agenda Page

MONITOR NETWORK BANDWIDTH USAGE

This involves tracking the amount of data being sent and received over the network to ensure optimal performance and troubleshoot any issues.

Tools like *iftop* and *nload* are commonly used for this purpose.

For example

- *iftop* displays a real-time overview of network traffic, showing which connections are using the most bandwidth.
- nload provides a visual representation of incoming and outgoing traffic.

By mastering these tools, you can effectively manage and optimize network resources.





MONITOR NETWORK BANDWIDTH USAGE

Here is an example:

- if you see "192.168.1.107 => 192.168.1.1" in your output, it shows the IP addresses in a network connection.
- Here, 192.168.1.107 is likely your computer's IP, and 192.168.1.1 is the IP of the device you're communicating with.
- The numbers like "4.93 Kb" or "2.18 Kb" indicate the network bandwidth usage at that moment, measured in kilobits per second (Kbps).
- These values are split into TX (data sent) and RX (data received). If you see "79.9KB peak," it shows the highest bandwidth usage recorded.
- The "TOTAL" value typically represents the total data transferred during the session.nd optimize network resources.

F			student@ubuntu1:	~	Q	≡	D 😣
Display paused	2.5Kb		25.0Kb	37.5Kb	50.0	кь	62.5Kb
192.168.1.107		=> (192.168.1.1		4.93Кb 3.55Кb	2.18Kb 1.69Kb	2.37Кb 1.71Кb
					×		
TX:	70 0/8	neak.	5 83Kb	cates:	4 9365	2 19Kb	2 37Kh

Х

Conclusion

Today, we've explored essential aspects of Linux administration, covering navigating the filesystem, managing files and directories, and efficiently locating data.

We've also learned valuable skills such as scripting, user and group management, network configuration, and system monitoring.

By understanding these core topics, you are now equipped to confidently navigate and manage Linux environments, ensuring both efficiency and security in your system operations.





Back to Agenda Page

