Linux Tasks

07.01.2025

Overview

Before we dive into containers, CI/CD pipelines, or infrastructure as code, we start with the one thing that powers *everything*: Linux. This blog is your beginner-friendly guide to that world-because without understanding Linux, you'll always be guessing in DevOps.

Why Linux Matters in DevOps

1. Linux is not just another operating system-it's the engine room of DevOps.

What is Linux?

Linux is an open-source operating system that acts as the backbone for most modern infrastructure. Think of it as the layer that sits between your applications and the

hardware, managing resources, users, and permissions. It handles memory allocation, process scheduling, file access, device control, and networking. Its modular, secure, and flexible design is why it's the go-to OS for DevOps, developers, and system administrators around the world.

Why It Matters in DevOps:

- Ubiquity: Most production servers run on Linux, not Windows or macOS.
- Containers: Docker, Podman, and container runtimes leverage Linux kernel features like namespaces and cgroups.
- Automation: Bash scripts, cron jobs, and configuration tools like Ansible work seamlessly on Linux.
- Cloud-native: The major cloud providers (AWS, Azure, GCP) offer default Linux-based instances.
- Security & Access: Permissions, firewalls, and auditing in Linux give you fine-grained control over system security.

Simply put, understanding Linux gives you the keys to everything else in DevOps.

Task 1: Navigation

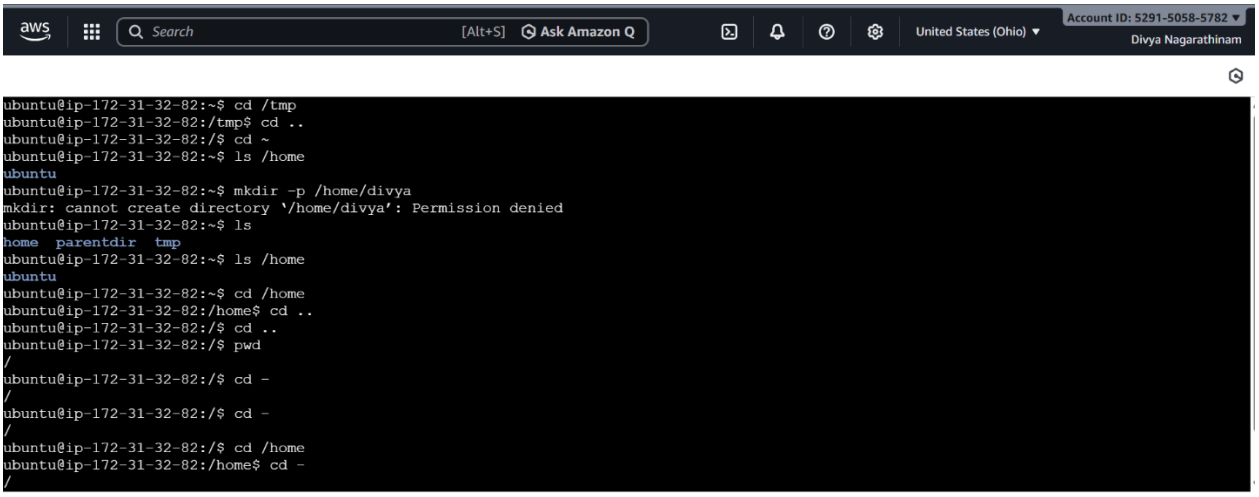
1. **Change directory to `/tmp`**
2. **Move to parent directory**
3. **Change directory to home directory**
4. **Move to parent directory**
5. **Move again to parent directory**
    i. **Where are you at? Verify with a command**
6. **Change to last visited directory**

Task 1 - Output:

1. cd /tmp
2. cd ..
3. cd ~
4. cd ..
5. cd ..

5 i. pwd

6. cd –

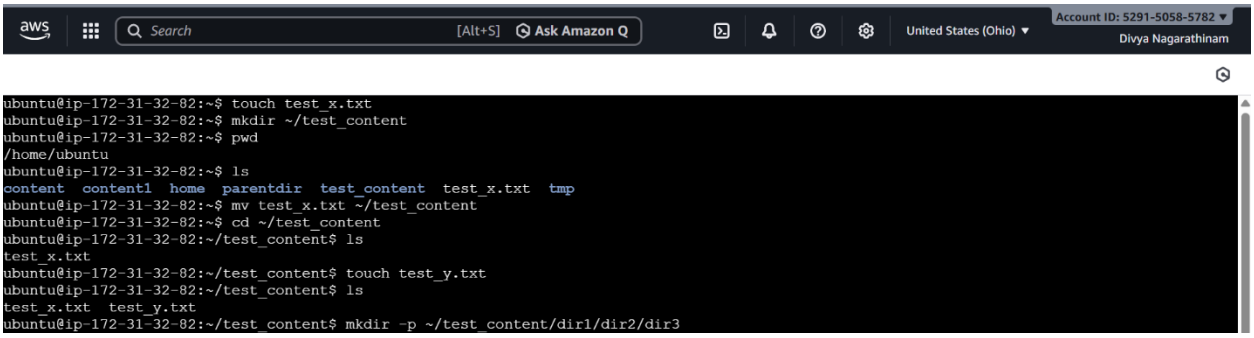


Task 2: Create & Destroy

1. Create a file called `x`
2. Create a directory called `content`
3. Move `x` file to the `content` directory
4. Create a file inside the `content` directory called `y`
5. Create the following directory structure in `content` directory: `dir1/dir2/dir3`
6. Remove the content directory

Task 2 - Output:

1. touch x.txt, echo test >> x.txt
2. mkdir ~/content
3. mv x.txt ~/content
4. cd ~/content, ls, touch y.txt, echo test >> x.txt, ls
5. mkdir -p ~/content/dir1/dir2/dir3, ls -R
6. rm -r ~/content, cd ~, ls



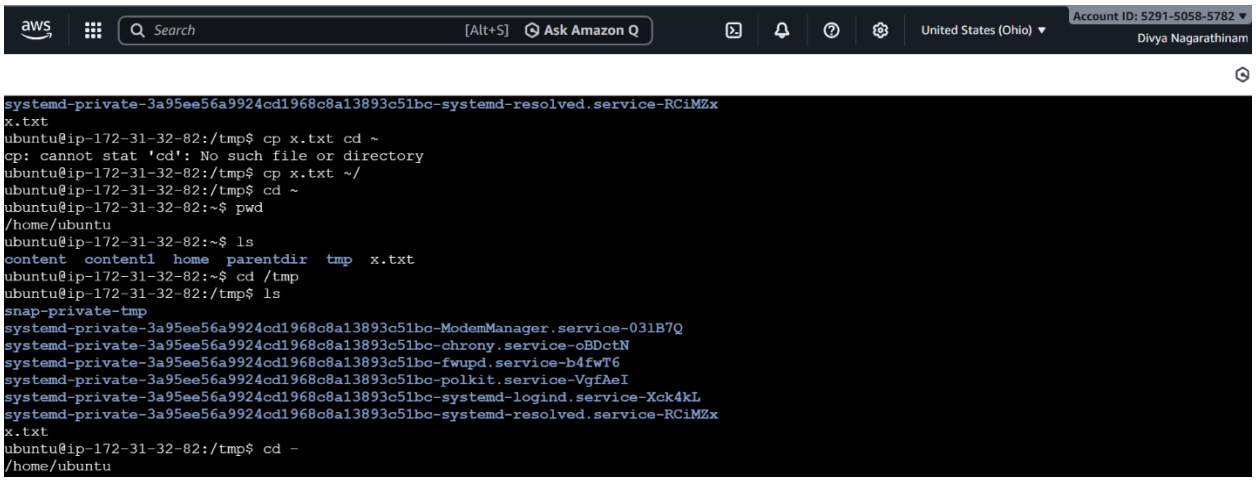

```
ubuntu@ip-172-31-32-82:~/test_content$ ls
dir1  test_x.txt  test_y.txt
ubuntu@ip-172-31-32-82:~/test_content$ ls -r
test_y.txt  test_x.txt  dir1
ubuntu@ip-172-31-32-82:~/test_content$ ls -R
.:
dir1  test_x.txt  test_y.txt

./dir1:
dir2

./dir1/dir2:
dir3

./dir1/dir2/dir3:
ubuntu@ip-172-31-32-82:~/test_content$ rm -r ~/test_content
ubuntu@ip-172-31-32-82:~/test_content$ cd ~
ubuntu@ip-172-31-32-82:~$ ls
content  content1  home  parentdir  tmp
ubuntu@ip-172-31-32-82:~$
```

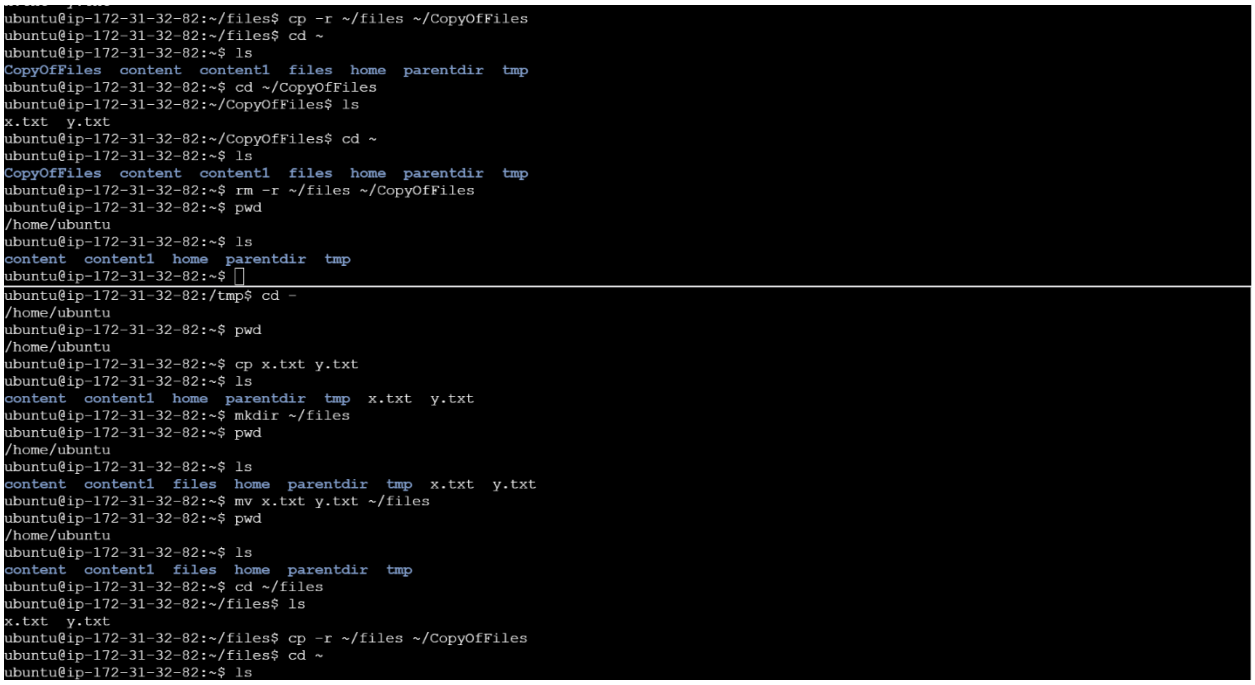
Task 3: Copy Time

1. Create an empty file called `x` in `/tmp`
2. Copy the `x` file you created to your home directory
3. Create a copy of `x` file called `y`
4. Create a directory called `files` and move `x` and `y` there
5. Copy the directory "files" and name the copy `copy_of_files`
6. Rename `copy_of_files` directory to `files2`
7. Remove `files` and `files2` directories

Task 3 - Output:

1. cd /tmp, touch x.txt
2. cp x.txt ~/, cd ~, pwd, ls
3. cp x.txt y.txt,
4. mkdir ~/files, mv x.txt y.txt ~/files, pwd, ls
5. cp -r ~/files ~/copyoffiles
6. mv ~/copyoffiles ~/files2, pwd, ls
6. rm -r ~/files ~/files2, pwd, ls




ubuntu@ip-172-31-32-82:~/files$ cp -r ~/files ~/CopyOfFiles
ubuntu@ip-172-31-32-82:~/files$ cd ~
ubuntu@ip-172-31-32-82:~$ ls
CopyOfFiles content content1 files home parentdir tmp
ubuntu@ip-172-31-32-82:~$ cd ~/CopyOfFiles
ubuntu@ip-172-31-32-82:~/CopyOfFiles$ ls
x.txt y.txt
ubuntu@ip-172-31-32-82:~/CopyOfFiles$ cd ~
ubuntu@ip-172-31-32-82:~$ ls
CopyOfFiles content content1 files home parentdir tmp
ubuntu@ip-172-31-32-82:~$ rm -r ~/files ~/CopyOfFiles
ubuntu@ip-172-31-32-82:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-32-82:~$ ls
content content1 home parentdir tmp
ubuntu@ip-172-31-32-82:~$
ubuntu@ip-172-31-32-82:/tmp$ cd -
/home/ubuntu
ubuntu@ip-172-31-32-82:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-32-82:~$ cp x.txt y.txt
ubuntu@ip-172-31-32-82:~$ ls
content content1 home parentdir tmp x.txt y.txt
ubuntu@ip-172-31-32-82:~$ mkdir ~/files
ubuntu@ip-172-31-32-82:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-32-82:~$ ls
content content1 files home parentdir tmp x.txt y.txt
ubuntu@ip-172-31-32-82:~$ mv x.txt y.txt ~/files
ubuntu@ip-172-31-32-82:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-32-82:~$ ls
content content1 files home parentdir tmp
ubuntu@ip-172-31-32-82:~$ cd ~/files
ubuntu@ip-172-31-32-82:~/files$ ls
x.txt y.txt
ubuntu@ip-172-31-32-82:~/files$ cp -r ~/files ~/CopyOfFiles
ubuntu@ip-172-31-32-82:~/files$ cd ~
ubuntu@ip-172-31-32-82:~$ ls