CompTIA Linux+: System Management

Managing Linux Files

Andrew Mallett

Linux Author and Trainer

@theurbanpenguin   www.theurbanpenguin.com

Overview

In this course you will learn all about managing Linux Systems to help your Administration and DevOps Skills

Linux Distributions

- Alma Linux 8.5
- openSUSE 15.2
- Ubuntu 20.04 LTS

Module Overview

- Managing Directories
- Understanding File Metadata
- Editing Files
- Copy Locally and Remotely
- Creating SSH Git Repos

Distribution Agnostic Lab Systems

CompTIA recommend using a mix of distributions, we include

- Alma Linux 8.5 (RHEL Rebuild)
- Ubuntu 20.04 LTS
- openSUSE Leap 15.2

Building in Vagrant/VirtualBox allows use of our Vagrantfile shared in the exercise files download.

CompTIA Linux+ is a distribution agnostic Linux certification helping you to get started in Linux administration and DevOps

```
$ pwd

$ cd /etc

$ echo $OLDPWD

$ cd $OLDPWD or cd -
```

Directory Operations

Some of the very basic Linux operations involve the cd command. I am sure this is basic to you but let's look at some things you may not know

Demo

Working with Files and Directories

- FHS
- Changing Directories
- Creating Directories
- Listing Directory and File Metadata
- Deleting Directories

```
$ touch file1
$ stat file1
$ stat -c %x file1 #Access
$ stat -c %y file1 #Modify
$ stat -c %z file1 #Change
```

Better Metadata

Field	Meaning
Access	Last read
Modify	Data last modified
Change	Metadata last changed

```
$ touch file1 #Changes both access and modified times, change will always update

$ touch -a file1 #Changes access time

$ touch -m file1 #Changes modified time
```

More on Touch

The touch command is often used to create new empty files but there is more to it

```
$ cp file1 file2
$ cp -a file1 file3
```

Archive Mode

To maintain metadata when copying files we can use the option -a or --archive, ownership, permissions and time stamps are all copied with the file.

-a is equal to -dR --preserve=all

Demo

Understanding File Metadata

- Sorting files my last modified time
- Using stat
- Using touch
- Preserving metadata

```
$ sudo dnf install -y vim-enhanced nano rsync git-core

$ sudo nano /etc/ssh/sshd_config
^W PasswordAuthentication

$ sudo vim /etc/ssh/sshd_config
/Password
```

Using Editors

When working at the Linux CLI, people will soon settle on their favored editor. The editor nano has a short learning curve while vim is more powerful, the learning curve is longer

```
$ sudo sshd -T | grep passwordauthentication
$ sudo sed -Ei 's/(PasswordAuthentication) no/\1 yes/' /etc/ssh/sshd_config
$ sudo systemctl restart sshd
```

Using the Stream Editor

To script or automate edits, always the goal in DevOps, we can look at sed, the Linux stream editor.

The Ubuntu Vagrant system does not enable password-based SSH authentication by default

Demo

Using CLI Editors

- nano
- vim

Demo

Automating Edits

- Using sed to allow Password Authentication

```
ubuntu$ sudo useradd -m user1
ubuntu$ sudo passwd user1
alma$ scp file1 user1@192.168.33.13:
alma$ mkdir docs
alma$ find /usr/share/doc -name '*.html'  -exec cp {} docs/ \;
alma$ rsync -ave ssh docs user1@192.168.33.13:
```

Remote Copy

Whilst scp is ok to copy single files to remote systems, rsync is better equipped to copy complete directory structures and keep them up to data

```
ubuntu$ su - user1 && mkdir project1 && cd project1
ubuntu$ git init --bare
alma$ git clone user1@192.168.33.13:/home/user1/project1
alma$ cd project1 && vim my.sh && chmod 755 my.sh
alma$ git add .
alma$ git config --global user.email "you@example.com"
alma$ git config --global user.name "Your Name"
alma$ git commit -m "initial commit"
alma$ git push origin master
opensuse$ sudo zypper in -y git-core
opensuse$ git clone user1@192.168.33.13:/home/user1/project1
```

Creating SSH Git Repositories

Working in DevOps, you will certainly come across the VCS (Version Control System) git, allowing collaboration in development. Rather than using a hosting system such as GitHub or GitLabs we can create our own SSH Host. There is more on git in later courses

Demo

Remote Copy

- Using scp
- Using rsync

Demo

Git Repositories

- Create repo on Ubuntu
- Add code from Alma
- View code on openSUSE

Summary

We now know:

- Working with directories and shortcuts
- The Linux FHS
- Listing directories including /dev
- Reading metadata
- Modify metadata and timestamps
- Editing files
  - vim
  - nano
- The stream editor
  - sed
- Remote copy operations
- Creating and using Git repos

Managing Linux Filesystems