

# 1- SYLLABUS OVERVIEW AND THEMES

miércoles, 24 de noviembre de 2021 02:02 a. m.

## MODULE 1. UNDERSTANDING LINUX CONCEPTS

WHAT IS LINUX?  
EVERYDAY USE OF LINUX  
UNIX VS. LINUX  
QUIZZES, HOMEWORK AND HANDOUTS

## MODULE 2. DOWNLOAD, INSTALL AND CONFIGURE

WHAT IS ORACLE VIRTUAL BOX?  
DOWNLOADING AND INSTALLING ORACLE VIRTUAL BOX  
CREATING VIRTUAL MACHINE  
LINUX DISTRIBUTIONS  
DIFFERENT WAY TO INSTALL LINUX  
DOWNLOADING AND INSTALLING LINUX (CentOS)  
RED HAT LINUX INSTALLATION  
LINUX DESKTOP (GUI)  
VM MANAGEMENT  
LINUX VS. WINDOWS  
QUIZZES, HOMEWORK AND HANDOUTS

## MODULE 3. SYSTEM ACCESS AND FILE SYSTEM

COMMAND PROMPTS AND GETTING PROMPTS BACK  
ACCESSING LINUX SYSTEM (SSH, TERMINAL CONSOLE)  
DOWNLOAD AND INSTALL PUTTY AND CONNECT VIA PUTTY  
IMPORTANT THINGS TO REMEMBER IN LINUX  
INTRODUCTION TO FILE SYSTEM AND DESCRIPTION  
FILE SYSTEM NAVIGATION COMMANDS (CD, PWD, LS, ETC.)  
FILE SYSTEM PATHS (ABSOLUTE AND RELATIVE)  
DIRECTORY LISTING OVERVIEW  
CREATING FILES AND DIRECTORIES (TOUCH, CP, VI, MKDIR, ETC.)  
FINDING FILES AND DIRECTORIES (FIND, LOCATE AND UPDATEDB)  
WILDCARDS (\*,\$,[])  
SOFT AND HARD LINKS (ln)  
QUIZZES, HOMEWORK AND HANDOUTS

## MODULE 4. LINUX FUNDAMENTALS

COMMAND SYNTAX  
FILES AND DIRECTORY PERMISSIONS (chmod)  
FILE OWNERSHIP  
ACCESS CONTROL LIST (ACL)  
GETTING HELP (man, whatis and --help)  
TAB completion and up arrow keys  
ADDING TEXT TO FILE (echo, redirects and vi)  
STANDARD OUTPUT TO A FILE (tee command)  
PIPES (|)  
FILE MAINTENANCE COMMANDS (cp, rm, mv, mkdir, rmdir, etc.)  
FILE DISPLAY COMMANDS (cat, more, less, head, tail and vi)  
FILTERS/TEXT PROCESSING COMMANDS (cut, sort, grep, awk, uniq, wc)  
COMPARE FILES (diff, cmp)  
COMPRESS, UN-COMPRESS AND TRUNCATE FILES (tar, gzip, gunzip, truncate)  
COMBINING AND SPLITTING FILES (cat and cut)  
LINUX VS. WINDOWS  
QUIZZES, HOMEWORK AND HANDOUTS

## MODULE 5. SYSTEM ADMINISTRATION

LINUX FILE EDITORS (vi text editor)  
DIFFERENCE BETWEEN VI AND VIM EDITORS  
SUBSTITUTION COMMAND (sed)  
USER ACCOUNT MANAGEMENT (useradd, groupadd, usermod, userdel)  
SWITCH USERS AND SUDO ACCESS  
MONITOR USERS (who, last, w and id)  
LINUX DIRECTORY SERVICE - ACCOUNT AUTHENTICATION  
DIFFERENCE BETWEEN ACTIVE DIRECTORY, LDAP, IDM, WinBIND, OpenLDAP etc.  
TALKING TO USERS (users, wall, write)  
SYSTEM UTILITY COMMANDS (date, uptime, hostname, which, cal, bc)  
SYSTEM LOGS MONITOR (/var/log)  
PROCESSES AND SCHEDULES (systemctl, ps, top, kill, crontab, at)  
SYSTEM MONITORING COMMANDS (top, df, dmesg, iostat 1, netstat, free)  
OS MAINTENANCE COMMANDS (shutdown, reboot, halt, init)  
CHANGING SYSTEM HGOSTNAMES (hostnamectl)  
FINDING SYSTEM INFORMATION (uname, redhat-release, \*rel\*, arch)

## MODULE 8. DISK MANAGEMENT AND RUN LEVELS

SYSTEM RUN LEVELS  
LINUX BOOT PROCESS (VERSION 6 AND 7)  
MESSAGE OF THE DAY  
STORAGE  
DISK PARTITION (df, fdisk)  
ADD DISK AND CREATE STANDARD PARTITION  
LOGICAL VOLUME MANAGER (LVM)  
LVM CONFIGURATION DURING INSTALLATION  
ADD VIRTUAL DISK AND CREATE NEW LVM PARTITION (pvcreate, vgcreate, lvcreate)  
EXTEND DISK USING LVM  
ADDING SWAP SPACE  
RAID  
FILE SYSTEM CHECK (fsck and xfs\_repair)  
SYSTEM BACKUP  
NETWORK FILE SYSTEM (NFS)  
DIFFERENCE BETWEEN CentOS/Redhat 5, 6 and 7  
QUIZZES, HOMEWORK AND HANDOUTS

## MODULE 9. ALL ABOUT RESUME

RESUME WORKSHOP  
COVER LETTER  
LINUX JOB DESCRIPTION OR DUTIES  
EXPOSURE TO OTHER TECHNOLOGIES  
HOMEWORK AND HANDOUTS

## MODULE 10. ALL ABOUT INTERVIEW TO GET HIRED

WHAT IS IT?  
IT COMPONENTS  
FACTS ABOUT IT  
LINUX AROUND US  
LINUX OS JOBS  
IT MANAGEMENT JOBS  
APPLY FOR LINUX JOBS AND WHAT TO EXPECT  
INTERVIEW WORKSHOP  
RED HAT CERTIFICATION (RHCSA, RHCE AND LFCS)  
JOIN LINUX COMMUNITY  
HOMEWORK AND HANDOUTS

## MODULE 11. COURSE RECAP

COMMANDS WE HAVE LEARNED  
DON'T GIVE UP  
RECAP - HANDOUT  
FINAL EXAM 100

## MODULE 12. ADDITIONAL RESOURCES

20+ ADITIONAL VIDEOS AND LECTURES

TERMINAL CONTROL KEYS, TERMINAL COMMANDS (clear, exit, script)  
RECOVER ROOT PASSWORD  
SOS REPORT  
ENVIRONMENT VARIABLES  
QUIZZES, HOMEWORK AND HANDOUTS

## MODULE 6. SHELL SCRIPTING

LINUX KERNEL  
WHAT IS SHELL?  
TYPES OF SHELLS  
SHELL SCRIPTING  
BASIC SHELL SCRIPTS  
INPUT AND OUTPUT OF SCRIPT  
IF-THEN SCRIPTS  
FOR LOOP SCRIPTS  
DO-WHILE SCRIPTS  
CASE STATEMENT SCRIPTS  
CHECK REMOTRE SERVERS CONNECTIVITY  
ALIASES  
USER AND GLOBAL ALIASES  
SHELL HISTORY  
QUIZZES, HOMEWORK AND HANDOUTS

## MODULE 7. NETWORKING, SERVERS AND SYSTEM UPDATES

ENABLING INTERNET IN LINUX VM  
NETWORK COMPONENTS  
NETWORK FILES AND COMMANDS (ping,ifconfig, netstat, tcpdump, networking config files)  
NIC INFORMATION  
NIC OR PORT BONDING  
DOWNLOAD FILES WITH URLs (wget)  
CONNECTIVITY COMMANDS (curl and ping)  
FILE TRANSFER PROTOCOL (FTP, SCP and rsync)  
SYSTEM UPDATES AND REPOSITORIES (rpm and yum)  
PATCH MANAGER  
ADVANCE PACKAGE MANAGEMENT  
SETTING UP LOCAL REPOSITORY  
SSH AND TELNET  
DOMAIN NAME SYSTEMS (DNS)  
HOSTNAME/IP LOOKUP (nslookup and dig)  
NTP AND Chrony  
SENDMAIL  
WEB SERVER (httpd)  
CENTRAL LOGGER (rsyslogd)  
SECURING LINUX MACHINE (OS Hardening)  
OpenLDAP INSTALLATION AND CONFIGURATION  
TRACING NETWORK TRAFFIC (traceroute)  
SSH KEYS (Accessing without password)  
QUIZZES, HOMEWORK AND HANDOUTS

# MODULE 1. UNDERSTANDING LINUX CONCEPTS

miércoles, 24 de noviembre de 2021 02:54 a. m.

## What is Linux?

LINUX is an operative system which sits in the middle of your hardware and users

It follows a simple flow Hardware -> OS->Applications->users

Commands used in Linux are NOT SAME as Windows DOS commands

LINUX came up with Linus Torvalds

## UNIX VS LINUX

UNIX was first developed for multi-users and multi-tasking, is a SO developed in 1970-1980. LINUX is mostly free and open source, LINUX is derived of UNIX. UNIX is used by sun as solaris, HP-UX, AIX, etc.

Linux is used by many developers community or companies

UNIX comparatively supports very fewer file systems

LINUX can be installed on a wide variety of computer hardware

UNIX is only installed on certain hardware

## HARD DISK DRIVE

In a HDD the physical location of data is mapped to a logical block address (LBA) that works with the larger address range on HDD

## DISK CACHE

Is a mechanism for improving the time it takes to read from or write to a HDD. Disk cache can be specified portion of RAM or it can be included as part of the hard disk. Holds data that has recently been read and adjacent data areas that are likely to be accessed next. Write caching is provided with some disk caches.

## INSIDE LINUX

### KERNEL

-The core of the UNIX system loaded at system start up memory-resident control program

-Manages entire resources of system, presenting to you and every other user as a coherent system. Provides service to user apps as device management, process scheduling, accomplishing the transfer of data from one part of the machine to another, interpreting and executing instructions from the shell, enforcing file access permissions.

### SHELL

You enter commands at this prompt, is a command interpreter that passes it to the OS kernel to be acted upon. It then displays the results of the operation on the screen

The most commonly shells are:

- Bourne shell(SH)
- C shell (CSH)
- Korn shell
- TC Shell (TCSH)
- Bourne Again Shell (BASH)

Each shell also includes its own programming language. Command files, called "shell scripts" are used to accomplish a series of tasks.

### UTILITIES

Accomplish universal functions (editing, file maintenance, printing, sorting, programming support, online info, etc.)

Modular, single functions can be grouped to perform more complex tasks

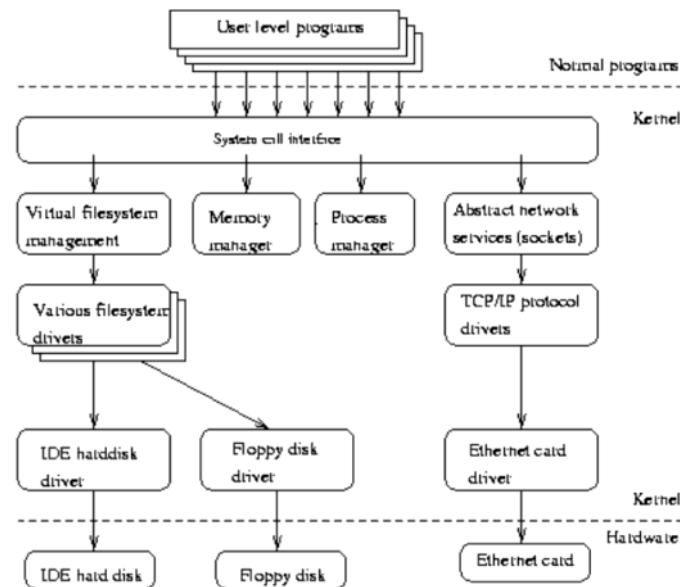
### OPERATING SYSTEM

Is a software that enables the computer hardware to communicate and operate with the computer software, among all the other programs in the computer, the other programs are called applications or applications programs, make request of services through a defined application program interface (API). In addition, users can interact directly with the operating system through a user interface such as command language or a GUI.

- .Manages the sharing of internal memory among multiple applications
- Handles input and output from attached hardware devices
- It sends messages to each application about the status of operation and any errors that may be occurred
- It can offload the management of what are called batch jobs (fex. printing) so that the initiating application is freed from his work
- Provides parallel processing, can manage how to divide the program so that it runs on more than one processor at time

## PARTS OF OPERATING SYSTEM AND IMPORTANT PARTS OF THE KERNEL

### MAIN PARTS OF KERNEL



At the lowest level kernel contains a hardware device driver for each kind of hardware it supports.

### VIRTUAL MEMORY

Virtual memory is using a disk as an extension of RAM so that effective size of usable memory grows correspondingly. The kernel will write the contents of a currently unused block of usable memory to the HDD, when the original contents are needed again, they are read back into memory. The part of the HDD that is used as virtual memory is called the swap space. When you know how much swap space you need, you should go for a swap partition, but if is uncertain you can use swap file first. LINUX allows to use several swap partitions and/or swap files.

Swapping is writing the whole process out to swap space and paging is writing only fixed size parts, usually a few kilobytes at a time

The kernel itself its not the OS, the OS provides more services than a plain kernel.

The important parts of the kernel consists of:

- Process management
- Memory management
- Hardware device drivers
- Filesystem drivers
- Network management
- Various other bits and pieces

## MODULE 2. DOWNLOAD, INSTALL AND CONFIGURE

jueves, 25 de noviembre de 2021 02:13 a. m.

### LINUX DISTRIBUTION

- REDHAD is an enterprise level operating system, they do charge for their support, provide their own customized, reconfigured build OS, is not open source, you cannot change it
- CentOS is exactly the same as REDHAT and is managed by REDHAT, they dont provide support and is free
- FEDORA is free
- SUZE very customizable, they change the code to give their proprietary appliances and their propierty programs
- DEBIAN very popular
- UBUNTU

### WHAT IS ORACLE VIRTUAL BOX?

VirtualBox is a free and open source hypervisor currently being developed by Oracle Corporation  
Is needed that you google Oracle virtual box and download it, after that complete installation.

Is needed that you have installed **VMWare Workstation player**

On isntallation we need to use Linux, Linux 2.6/3.x/4.x of 64 bits according to your computer and using 1 GB of RAM is sufficient and 10 GB of storage. VDI VirtualBox Disk Image

### DIFFERENT WAYS TO INSTALL LINUX

The first one is through the CD or DVD  
The second is through the iso image  
The network boot

### LINUX DOWNLOAD AND INSTALLATION

CentOS 7 is recommended: [http://isoredirect.centos.org/centos/7/isos/x86\\_64/](http://isoredirect.centos.org/centos/7/isos/x86_64/)  
To install you have to create an empty shell and then load the iso file, for this course we select server with GUI, but the recomendable option is to select a console interface.

KDUMP is whe there is something goes wrong with your system and it crashes it create the log files.  
Then in the part of network we have to select configure and assign a name to the connection, in configuration options we have to click on General tab and then select automatically connect to this network when its available.

Same steps to install RedHat or CentOS 8.

### CentOS vs CentOS Stream

CentOS means Community Enterprise Operating System which means that a lot of people work together to make that operating system.

RHEL means Red Hat Enterprise Linux.

CentOS Stream is the actual way to update, this OS keep the next workflow:  
Fedora->CentOS Stream->RHEL (After Feb 2021)  
Before the WF was:  
Fedora->RHEL->CentOS

### CREATE A SNAPSHOT OF VM

A snapshot allows you to save the current state of VM, in case you need to come back and do a restore it.

# LINUX DESKTOP (GUI)

viernes, 26 de noviembre de 2021 02:06 a. m.

## GNOME OR KDE

GNOME only create two icons by default Home and Trash

## VIRTUAL MACHINE MANAGEMENT

Is needed because 80 percent of companies run in a virtual environment, is part of your job to add or remove resources.

Go to setting and the first thing to change is the name if you want, drag and drop is for have something in you main OS and your SO in your VM.

Network section is about a bridged adapter that means that VM takes the IP of your computer and with this communicates.

## LINUX VS. WINDOWS

Linux is free, Windows have a cost

Linux is not user friendly, Windows is user friendly

LINUX is reliable (de confianza) Windos sometimes need a reboot

LINUX is mostly enterprise level softwares, WINDOWS have more applications

LINUX is better for multitasking, WINDOWS have multi-tasking available but with very high CPU or memory resources

Linux is very secure, Windows is some what (un poco) secure

LINUX is Open source, Windows is no an open source OS

## WHO USES LINUX?

US Government and agencies

NASA

Health care (Hospitals)

Bullet train

Traffic control

Financial institutes

Entertainment industries

World e-commerce leaders

# KEYBOARD KEYS USED IN LINUX

sábado, 27 de noviembre de 2021 10:46 p. m.

ESC key is used to go out of editing mode or a task, it will get you out of that. And is other to get out of the previously pressed.

TILDE key is that little curvy thing below of ESC, TILDE is actually taking you to your home directory

BACK THICK is under TILDE key

Exclamation key is under one in the same key of one number, is used

ZION key is to add the email domain, is on the same key as two number

Three number have a hastagh symbol is used in scripting or in the file or any configuration file to comment out

DOLLAR SIGN, is defined as invariable which will cover in the shell scripting module

PERCENT SIGN, in LINUX means something in the beginning of something

AMPERSON, is to put something in the background, in a script AMPERSON SIGN will put the process in the background

STAR is a asterisco

NINE key is to open parentheses

Undescore and under that is HYPEN key, which is also used as an option or dash, so if you're running a command in LINUX then you have to specify parameters or options with it

TAB skipes some spaces

CURLY BRACES key and under that is OPENING BRACKET, in front of that key is CLOSING CURLY BRACES AND CLOSING BRACKET

ENTER KEY is also referred as carriage return

COLIN KEY OR SEMICOLON KEY are used in scripting language commonly used to save the file

Double COAT or single COAT

PIPE is a simple straight line, it is used to combine a few commands together or take the output from first command and give it to the second command, below of that symbol we have BACKSLASH key that is used in windows for the routes

LESS THAN or comma sign

GREATER THAN or period or dot, in LINUX is also referred in LINUX as an input

and output, if you use greater than sign it means appending to the file, if you use less than it means overwrite to your file, less sign will take the input from the output

BACKSLASH is used in LINUX to access a path

CTRL key many times in virtualization if you want to get out of that and release your mouse, you have to use right ctrl key.

# MODULE 3. SYSTEM ACCESS AND FILE SYSTEM

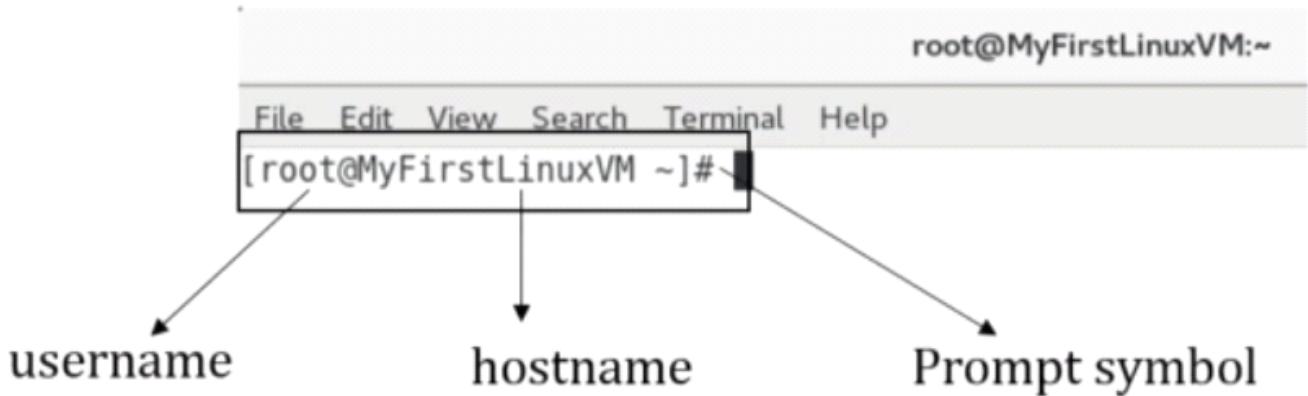
domingo, 28 de noviembre de 2021 04:24 p. m.

## COMMAND PROMPTS AND GETTING PROMTS BACK

It is important cause we will know how could get a command prompt back if it gets stuck

### COMMAND PROMPT

Also referred to simply as a prompt, is a short text at the start of the command line followed by a prompt symbol on a command line interface.



If you log in as a normal user prompt symbol will change to dollar sign \$. Superuser prompt symbol is a hashtag

To get your prompt back you have to use: Ctrl+C

### COMMANDS

whoami, shows the actual user that is logged inside the terminal  
hostname, shows the hostname

# ACCESSING LINUX SYSTEM

domingo, 28 de noviembre de 2021 04:35 p. m.

You can access to a terminal, example each operating system has a different protocol or client that is used to access the system.

Windows	Remote desktop (RDP)
Vmware ESX	vSphere client
Linux	Putty, SecureCRT, SSH from LINUX to LINUX

## NETWORK COMMANDS

CentOS/RHEL 5 or 6 =**ifconfig**

CentOS/RHEL 7=**ip**

CentOS/RHEL 7.5 and up=**ifconfig** has been deprecated

To use ifconfig in 7.5="yum install net-tools"

ifconfig command will list all interface on the LINUX machine

Also we can use

ip a command that lists the same information

ip addr

man ip

# CONNECTING TO LINUX VM THROUGH PUTTY

domingo, 28 de noviembre de 2021 06:44 p. m.

1. Start VM
2. Open terminal and execute command ip a or ifconfig if is not deprecated according to your version of OS
3. Remember that we have to go to settings of the VM and change it to a different settings as you can see in the next picture.

**Host-only Adapter = Allows communication between your PC and the Virtual Machine**

**Bridged Adapter = Allows communication between your PC and VM plus allows communication to the internet**

To enable internet access you have to select bridged adapter, but if it is not allowed you can use the next commands:

```
su -  
ifup <NAME OF INTERFACE>
```

For MAC users you have to open the terminal on your MAC and run the next command

```
ssh -l iaf <username> <ipaddress>
```

For Windows users you have to access to the server through PUTTY placing the ip as ifconfig command or ip a command indicate, after that prompt will be ask for the username and the password, after entering that information you will be able to communicate with the VM through PUTTY.

# IMPORTANT THINGS TO REMEMBER IN LINUX

domingo, 28 de noviembre de 2021 06:59 p. m.

- Linux has a super-user account called root, **root** is the **most powerful** account that can create, modify, delete accounts and make changes to system configuration files.
- Linux is case-sensitive system, this means that abc is NOT same as ABC
- Avoid using spaces when creating files and directories, instead uses dashes or other
- LINUX kernel is not an operating system. It is small software within LINUX operating system that takes commands from users and pass them to system hardware or peripherals
- LINUX is mostly CLI (Command Line Interface) not GUI (Graphical user interface)
- LINUX is very flexible as compared to other OS

# LINUX FILE SYSTEM

domingo, 28 de noviembre de 2021 07:15 p. m.

You can see a file system as a closet, there's a systematic way in your closet where you put all your stuff.

CLOSET SYSTEM= OS FILE SYSTEM

- Commands=/usr/bin/
- Attached devices=/dev
- Applications and programs= /etc or /var

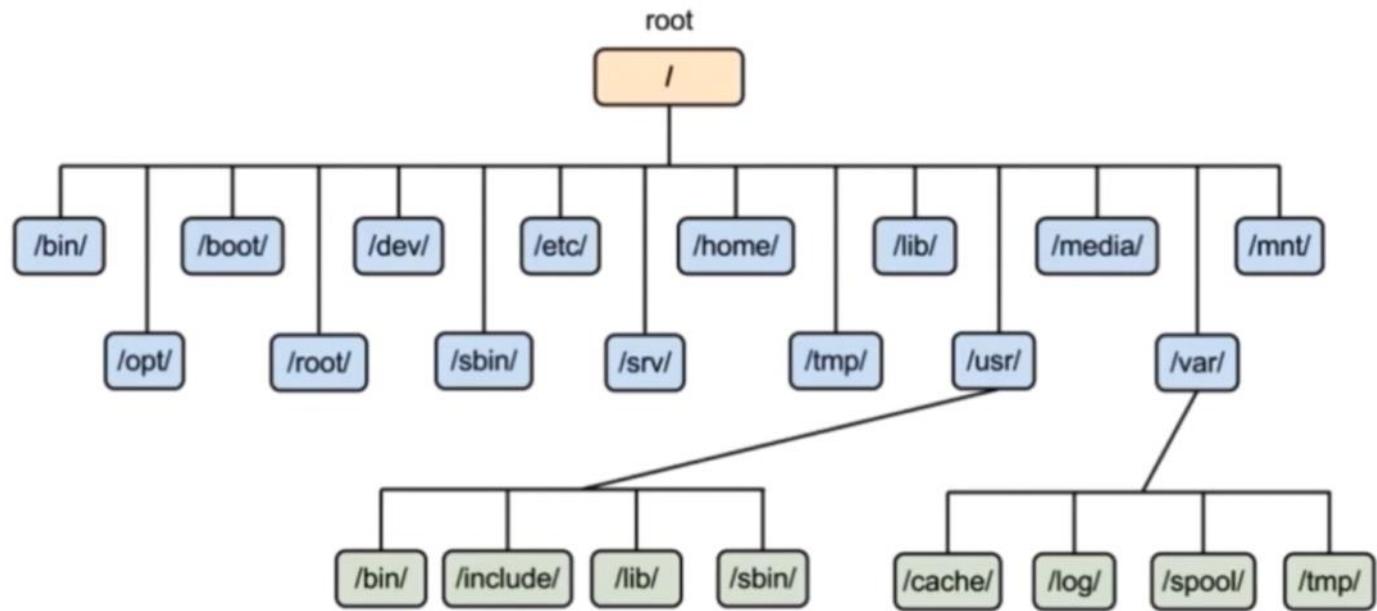
At this way is much easier to OS access to those files

OS store data on disk drives using a structure called a filesystem, consisting of files, directories, and the information needed to access and locate them. Improvements have been made to filesystems with new releases of operating systems, and each new filesystem has been given a different name.

In LINUX there are:

- ext3
- ext4
- XFS
- NTFS
- FAT

LINUX filesystems store information in a hierarchy of directories and files have the next structure:



The first file system or folder always starts with a slash, keep in mind that slash root is not just a slash, slash is the first directory structure where you have to start navigating the file system and then underneath it has different folders

You can go to LINUX file system placing the next commands:

```
cd /  
ls -l
```

# FILESYSTEM STRUCTURE AND ITS DESCRIPTION

domingo, 28 de noviembre de 2021 08:52 p. m.

/boot	Contains files that is used by the boot loader (grub.cfg)
/root	Root user home directory. It is not the same as /
/dev	System devices (disk, cdrom,speakers,etc.)
/etc	Configuration files, before to modify a configuration files please make sure you make a backup first of that file
/bin or /usr/bin	Everyday user commands, as ls
/sbin or /usr/sbin	System/filesystem commands, commands that you need for system type of purposes
/opt	Optional add-on applications (Non part of OS apps), for example oracle vm or another most of this is installed in /opt
/proc	Create files for all the running processes, running processe (Only exist in memory), when you shutdown the LINUX machine, this folder will be empty
/lib or /usr/lib	C programming library files needed by commands and apps, meaning that if you are running a command or an app that needs libraries attached to it they could go and look for, if you want to know which command is used and which library file is used you can use the next command: <b>strace -e open &lt;command&gt;</b>
/tmp	Directory for temporary files
/home	Directory for regular users
/var	System logs, is used for every application logs, if you are troubleshooting issues you can always go into /var and navigate till find the log file that are related to that thing that you are trying to find
/run	System daemons that start very early (systemd and udev) to store temporary runtime files like PID files
/mnt	To mount external filesystem (as it can be NFS)
/media	For cdrom mounts

Grub.cfg is telling the system which OS to start or which version of operating system to start  
**pwd** command shows me the path that I am

# FILE SYSTEM NAVIGATION COMMANDS

lunes, 29 de noviembre de 2021 02:15 a. m.

cd	Means change directory
pwd	Tell you the current directory where you in
ls	List the content of your directory or folder
ls -l	Is a command for see what is inside of a directory including files, every file that name begin with - is a file, d is a directory

# WHAT IS ROOT?

lunes, 29 de noviembre de 2021 02:29 a. m.

There are 3 types of root on LINUX system:

1. ROOT ACCOUNT: it is the most powerful account which has access to all commands and files
2. ROOT AS /: the very first directory in LINUX is also referred as root directory
3. ROOT HOME DIRECTORY: the root user account also has a directory located in /root which is called root home directory

# FILE SYSTEM PATHS - ABSOLUTE AND RELATIVE PATHS

lunes, 29 de noviembre de 2021 02:40 a. m.

There are two paths to navigate to a filesystem:

- Absolute path
- Relative path

An absolute path always begins with a /. This indicates that the path starts at the root directory. For example cd /var/log/samba

A relative path does not begin with /. It identifies a location relative to your current position. For example: cd /var, then cd log, then cd samba

# DIRECTORY LISTING ATTRIBUTES

Lunes, 29 de noviembre de 2021 02:47 a. m.

Using ls -l or ls -ltr you will get the next output as the image

If any file begins with d is a directory

If any file begins with l is a link

If any file begin with a - is a regular file

Type	# of Links	Owner	Group	Size	Month	Day	Time	Name
drwxr-xr-x.	21	root	root	4096	Feb	27	13:33	var
lrwxrwxrwx.	1	root	root	7	Feb	27	13:15	bin
-rw-r--r--	1	Root	Root	0	Mar	2	11:15	testfile

ls -ltr means that I want to list all the files or directories that are created that was last modified and in reverse the oldest one at the top and the newest one at the bottom

# CREATING FILES AND DIRECTORIES (TOUCH, CP,VI,MKDIR)

lunes, 29 de noviembre de 2021 02:52 a. m.

## CREATING FILES

- touch <name of file> create an empty file
- Cp <name of file to copy><new name and location> if you are copying an existing file and creating a new field at the destination location
- vi <name of file> is our editor, to get out of this editor is needed to press shift and colon, then place :wq!

With touch command you can create multiple files just giving the names, touch <file1> <file2> <file3>

## CREATING DIRECTORIES

- mkdir

With mkdir command you can create multiple directories

# COPYING DIRECTORIES

lunes, 29 de noviembre de 2021 03:16 a. m.

To copy a directory is needed to use the next command:

- cp

To copy a directory on LINUX, you have to execute cp command with the -R option for recursive and specify the source and destination directories to be copied:

- cp -R <source\_folder> <destination\_folder>

# LINUX FILE TYPES

lunes, 29 de noviembre de 2021 03:22 a. m.

There are 7 files types:

File Symbol	Meaning
-	Regular file
d	Directory
l	Link
c	Special file or device file, a keyboard something attached to CPU or memory that also is created as a file
s	Socket
p	Named pipe
b	Block device

# FINDING FILES AND DIRECTORIES

lunes, 29 de noviembre de 2021 03:29 a. m.

To begin with this

- Login as yourself
- Mv jerry kramer george /home/your\_name/seinfeld

You can find files using the next commands:

- find
- locate

If we use find command after that put a dot it means that the search will begin from the current location:

```
find . -name "kramer"  
find / -name "ifcfg-enp0s3" to search in a network
```

If we are using locate, the syntax is same

```
locate kramer
```

If locate command does not output any result then as a root run: updatedb  
Also make sure you have mlocate package installed. To install the package as root run: yum install mlocate

To get help you can use man <command>

After this lesson run the following commands:

```
cd /home/yourname/seinfeld  
Mv jerry kramer george /home/yourname
```

# DIFFERENCE BETWEEN FIND AND LOCATE COMMAND

lunes, 29 de noviembre de 2021 03:46 a. m.

Locate command uses a prebuilt database, which should be regularly updated, while find iterates over a filesystem to locate files. Thus, locate is much faster than find, but can be inaccurate if the database (can be seen as cache) is not updated.

To update locate database run updatedb

# CHANGING PASSWORD

lunes, 29 de noviembre de 2021 03:53 a. m.

You should change your initial password as soon as you login, to change a password you have to use the next command:

`passwd userid`(in my case is terron)

This command will show the next info:

Old password

New password

Retype the new password

# WILDCARDS (\*, ?, ^, [])

martes, 30 de noviembre de 2021 12:30 a. m.

Wildcard, in LINUX we use wildcards to get the output that we are looking for.

A wildcard is a character that can be used as substitute for any of a class of characters in a search.

- \* Represents zero or more characters
- ? Represents a single character
- [] represents a range of characters

Using the next command you will delete all files that have the next pattern:

```
[iafzal@MyFirstLinuxOS ~]$ rm abc*
```

To create a lot of files of one time we can use the next command:

```
[root@MyFirstLinuxVM ~]# touch abcd{1..9}-xyz
[root@MyFirstLinuxVM ~]# ls -l
total 8
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd1-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd2-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd3-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd4-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd5-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd6-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd7-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd8-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd9-xyz
-rw-----. 1 root root 1690 Nov 26 01:01 anaconda-ks.cfg
-rw-r--r--. 1 root root 1738 Nov 26 01:24 initial-setup-ks.cfg
```

For example, if we want to filter files using a part of the name at the file we can use the next command:

```
[root@MyFirstLinuxVM ~]# ls -l abc*
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd1-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd2-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd3-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd4-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd5-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd6-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd7-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd8-xyz
-rw-r--r--. 1 root root 0 Nov 29 09:43 abcd9-xyz
```

I'm telling to the prompt that I want to get all files that begin with abc and end with any other word.

If I want to del a files which contains bc I can use the next command:

```
[root@MyFirstLinuxVM ~]# ls -l *[bc]*  
-rw-r--r--. 1 root root 0 Nov 29 09:49 abc1-xyz  
-rw-r--r--. 1 root root 0 Nov 29 09:49 abc2-xyz  
-rw-r--r--. 1 root root 0 Nov 29 09:49 abc3-xyz  
-rw-r--r--. 1 root root 0 Nov 29 09:49 abc4-xyz  
-rw-r--r--. 1 root root 0 Nov 29 09:49 abc5-xyz  
-rw-r--r--. 1 root root 0 Nov 29 09:49 abc6-xyz  
-rw-r--r--. 1 root root 0 Nov 29 09:49 abc7-xyz  
-rw-r--r--. 1 root root 0 Nov 29 09:49 abc8-xyz  
-rw-r--r--. 1 root root 0 Nov 29 09:49 abc9-xyz  
-rw----- 1 root root 1690 Nov 26 01:01 anaconda-ks.cfg  
-rw-r--r--. 1 root root 1738 Nov 26 01:24 initial-setup-ks.cfg
```

Other wildcards

\ slash- as an escape character  
^ caret- the beginning of the line  
\$ dollar sign the end of the line

# SOFT AND HARD LINKS (ln)

martes, 30 de noviembre de 2021 12:53 a. m.

Before to let in the theme we have to understand a definitions:

## INODE

inode=Pointer or number of a file on the hard disk, every time that you create a file, computer assigns a number to that file on a hard disk and associate that number to it. That number is called inode and every time I try to retrieve that file or read that file, it goes and retrieve that number

## SOFT LINK

Soft link is a link that will be removed if file is removed or renamed, if I create a file and after I create a soft link, and after that I remove the source file the soft link will be removed

## HARD LINK

A hard link, deleting, renaming or moving the original file will not affect the hard link, to create a hard link we need to use the next command:

ln (ele-ene) ----> HARD LINK  
ln -s ----->SOFT LINK

Link is just like a shortcut in a regular term, as if you are creating a shortcut on your windows desktop.

We can see the next command:

```
[terron@MyFirstLinuxVM ~]$ pwd  
/home/terron  
[terron@MyFirstLinuxVM ~]$ touch hulk  
[terron@MyFirstLinuxVM ~]$ cd /tmp  
[terron@MyFirstLinuxVM tmp]$ ln -s /home/terron/hulk
```

And then using command:

ls -ltr

We have the next output.

```
drwx----- . 2 terron terron      6 Nov 29 10:45 tracker-extract-files.1000  
lrwxrwxrwx . 1 terron terron 17 Nov 29 10:46 hulk -> /home/terron/hulk
```

## INSERT LINE INTO AN EMPTY FILE

We have to use the next commands:

```
[terron@MyFirstLinuxVM ~]$ echo "Trying to insert a line to hulk file TERRON" > hulk  
[terron@MyFirstLinuxVM ~]$ cat hulk  
Trying to insert a line to hulk file TERRON  
[terron@MyFirstLinuxVM ~]$
```

Cat is to see the content and echo is to insert it using the > operator

NOW if we go to the place that we place the soft link we have the next:

```
[terron@MyFirstLinuxVM ~]$ cd /tmp/  
[terron@MyFirstLinuxVM tmp]$ cat hulk  
Trying to insert a line to hulk file TERRON  
[terron@MyFirstLinuxVM tmp]$ █
```

If we run ls -li, i is for inodes we will have the number of the file that is the inode assigned to that file:

```
[terron@MyFirstLinuxVM ~]$ ls -ltri  
total 4  
 5548024 drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Templates  
10206210 drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Public  
    76729 drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Music  
    76728 drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Downloads  
13839822 drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Documents  
13839821 drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Desktop  
10206211 drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Videos  
 5548025 drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Pictures  
   77871 drwxrwxr-x. 2 terron terron 6 Nov 29 03:31 ejercicios  
 5546890 drwxrwxr-x. 2 terron terron 47 Nov 29 03:32 seindfield  
8540732 -rw-rw-r--. 1 terron terron 45 Nov 29 10:50 hulk  
[terron@MyFirstLinuxVM ~]$ █
```

HARDLINKS only work within the same partition

To create a hardlink we have to use the next command:

---

```
[terron@MyFirstLinuxVM ~]$ pwd  
/home/terron  
[terron@MyFirstLinuxVM ~]$ touch hulk  
[terron@MyFirstLinuxVM ~]$ echo "ADDING A NEW LINE HARDLINK" > hulk  
[terron@MyFirstLinuxVM ~]$ cat hulk  
ADDING A NEW LINE HARDLINK  
[terron@MyFirstLinuxVM ~]$ cd /tmp/  
[terron@MyFirstLinuxVM tmp]$ ln /home/terron/hulk
```

After that we can check if hardlink is created usin ls -litr command as the next picture, and get the last line and see that the hardlink has been created:

```
-rw-rw-r--. 2 terron terron 27 Nov 29 11:09 hulk
```

The difference between hardlink and softlink is that hardlink is no pointing to other file.

Now if we modify the source file the link file will be modified as well.

```
[terron@MyFirstLinuxVM ~]$ echo "HULK IS A SUPER HERO" > hulk  
[terron@MyFirstLinuxVM ~]$ cat hulk  
HULK IS A SUPER HERO  
[terron@MyFirstLinuxVM ~]$ echo "NEW LINE TERRON" >> hulk  
[terron@MyFirstLinuxVM ~]$ cat hulk  
HULK IS A SUPER HERO  
NEW LINE TERRON  
[terron@MyFirstLinuxVM ~]$
```

Now if we delete the source file and then go to the path that we have created the hard link the file keeps exist and if we make a cat of hulk we have the next.

```
[terron@MyFirstLinuxVM tmp]$ cd  
[terron@MyFirstLinuxVM ~]$ rm hulk  
[terron@MyFirstLinuxVM ~]$ cd /tmp/  
[terron@MyFirstLinuxVM tmp]$ ls -lt hulk  
-rw-rw-r--. 1 terron terron 37 Nov 29 11:16 hulk  
[terron@MyFirstLinuxVM tmp]$ cat hulk  
HULK IS A SUPER HERO  
NEW LINE TERRON  
[terron@MyFirstLinuxVM tmp]$
```

This is telling us that in the hard link, when you remove the source file it does not remove the destination.

## HOMEWORK

1. Create 10 files under your home directory

```
[terron@MyFirstLinuxVM ~]$ pwd  
/home/terron  
[terron@MyFirstLinuxVM ~]$ touch jerry kramer george lex clark lois homer bart lisa marge  
[terron@MyFirstLinuxVM ~]$ ls /ltr  
ls: cannot access /ltr: No such file or directory  
[terron@MyFirstLinuxVM ~]$ ls -ltr  
total 0  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Templates  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Public  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Music  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Downloads  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Documents  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Desktop  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Videos  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Pictures  
drwxrwxr-x. 2 terron terron 6 Nov 29 03:31 ejercicios  
drwxrwxr-x. 2 terron terron 47 Nov 29 03:32 seinfeld  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lois  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lisa  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lex  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 jerry  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 homer  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 george
```

2. Create 3 directories under your home directory

```
[terron@MyFirstLinuxVM ~]$ mkdir seinfeld superman simpsons
[terron@MyFirstLinuxVM ~]$ ls -ltr
total 0
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Templates
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Public
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Music
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Downloads
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Documents
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Desktop
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Videos
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Pictures
drwxrwxr-x. 2 terron terron 6 Nov 29 03:31 ejercicios
drwxrwxr-x. 2 terron terron 47 Nov 29 03:32 seinfeld
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lois
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lisa
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lex
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 jerry
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 homer
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 george
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart
drwxrwxr-x. 2 terron terron 6 Nov 29 11:34 superman
drwxrwxr-x. 2 terron terron 6 Nov 29 11:34 simpsons
```

3. Create a new file `jupiter` and write to it as "Jupiter is a planet". Then create a soft link in `/tmp` directory

```
[terron@MyFirstLinuxVM ~]$ touch jupiter
[terron@MyFirstLinuxVM ~]$ echo "JUpiter is a planet" > jupiter
[terron@MyFirstLinuxVM ~]$ cd /tmp/
[terron@MyFirstLinuxVM tmp]$ ln -s /home/terron/jupiter
```

4. Also create a hardlink of `jupiter` in `/tmp` directory

```
[terron@MyFirstLinuxVM tmp]$ ln /home/terron/jupiter
[terron@MyFirstLinuxVM tmp]$ ls -ltri jupiter
10984648 -rw-rw-r--. 2 terron terron 20 Nov 29 11:35 jupiter
[terron@MyFirstLinuxVM tmp]$
```

5. Check de inodes for both links

#### SOFT LINK

```
[terron@MyFirstLinuxVM tmp]$ ls -ltri jupiter
5468171 lrwxrwxrwx. 1 terron terron 20 Nov 29 11:36 jupiter -> /home/terron/jupiter
```

#### HARDLINK

```
[terron@MyFirstLinuxVM tmp]$ ln /home/terron/jupiter
[terron@MyFirstLinuxVM tmp]$ ls -ltri jupiter
10984648 -rw-rw-r--. 2 terron terron 20 Nov 29 11:35 jupiter
[terron@MyFirstLinuxVM tmp]$
```

# LINUX COMMAND SYNTAX

martes, 30 de noviembre de 2021 02:45 a. m.

Commands have options and arguments

## SYNTAX

The typical syntax of commands is:

command option(s) argument(s)

## OPTIONS

The option modifies the way that a command works

Usually consist of a hyphen or dash followed by a single letter

Some commands accept multiple options which can be grouped together after a single hyphen

## ARGUMENTS

Most commands are used together with one or more arguments

Some commands assume a default argument if none is supplied

Arguments are optional for some commands and required by others

Arguments are used to be more specific.

NOTE: remember that hyphen symbol is a dash -

```
[terron@MyFirstLinuxVM tmp]$ cd  
[terron@MyFirstLinuxVM ~]$ ls -ltr *a  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lisa
```

Command: ls

Options: -ltr

Arguments: \*a

## REMOVE A DIRECTORY

To remove a directory without asking us whether they should delete or not we have to use the option -f as the next picture.

```
[terron@MyFirstLinuxVM ~]$ rm -rf superman/  
[terron@MyFirstLinuxVM ~]$ █
```

## GET HELP AND MANUAL FOR ANY COMMAND

To get help about a command we can use the next command

man <command>

As we can see in the next picture.

**NAME**

**rm** - remove files or directories

**SYNOPSIS**

**rm** [OPTION]... FILE...

**DESCRIPTION**

This manual page documents the GNU version of **rm**. **rm** removes each specified file. By default, it does not remove directories.

If the -I or --interactive=once option is given, and there are more than three files or the -r, -R, or --recursive are given, then **rm** prompts the user for whether to proceed with the entire operation. If the response is not affirmative, the entire command is aborted.

Otherwise, if a file is unwritable, standard input is a terminal, and the -f or --force option is not given, or the -i or --interactive=always option is given, **rm** prompts the user for whether to remove the file. If the response is not affirmative, the file is skipped.

**OPTIONS**

Remove (unlink) the FILE(s).

Manual page **rm**(1) line 1 (press h for help or q to quit)

# FILES AND DIRECTORY PERMISSIONS

martes, 30 de noviembre de 2021 03:05 a. m.

This is used to protect your environment, your files and your directories from being viewed by other users or deleted by other users.

UNIX is a multisystem-user system. Every file and directory in your account can be protected from or made accessible to other users by changing its access permissions. Every user has responsibility for controlling access to their files.

Permissions for a file or directory may be restricted to three types

There are 3 types of permissions

- r read
- w write
- x execute= running a program

Each permission can be controlled at three levels

- u user=yourself
- g group= can be people in the same project
- o other= everyone in the system

File or directory permission can be displayed by running ls -l command

- -rwxrwxrwx

This means that - is a regular file, first rwx is permissions to the user, second rwx means permission for group and the last rwx means the permission for other users

## COMMAND TO CHANGE PERMISSIONS

To change permissions on a file we have to use the next command

chmod

```
[terron@MyFirstLinuxVM ~]$ pwd  
/home/terron  
[terron@MyFirstLinuxVM ~]$ ls -l george  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 george  
[terron@MyFirstLinuxVM ~]$ █
```

The info that we can get is - a regular file, the user has permission to read and write, we do not have permission to execute this file because it is not a script. For group a regular file, read and write permissions and not executable, the last other only has permission to read.

To change permissions we can use the next command in the next way.

REMOVE PERMISSIONS

```
[terron@MyFirstLinuxVM ~]$ pwd  
/home/terron  
[terron@MyFirstLinuxVM ~]$ ls -l george  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 george  
[terron@MyFirstLinuxVM ~]$ chmod g-w george  
[terron@MyFirstLinuxVM ~]$ ls -l george  
-rw-r--r--. 1 terron terron 0 Nov 29 11:32 george  
[terron@MyFirstLinuxVM ~]$ █
```

We have to use chmod -g as a group and then name of file as we can see in the last picture.

If we want to change permissions for every user we have to use the next options:

```
[terron@MyFirstLinuxVM ~]$ chmod a+w george  
[terron@MyFirstLinuxVM ~]$ ls -l george  
-rw-rw-rw-. 1 terron terron 0 Nov 29 11:32 george  
[terron@MyFirstLinuxVM ~]$ █
```

To change only permissions for the user we have to use u option as the next picture indicates.

```
[terron@MyFirstLinuxVM ~]$ chmod u-w george  
[terron@MyFirstLinuxVM ~]$ ls -l george  
r--rw-rw-. 1 terron terron 0 Nov 29 11:32 george  
[terron@MyFirstLinuxVM ~]$ █
```

In the directories you have x as a permission to execute cause you can cd into that directory.

To change permission to enter into a directory we can use the next command.

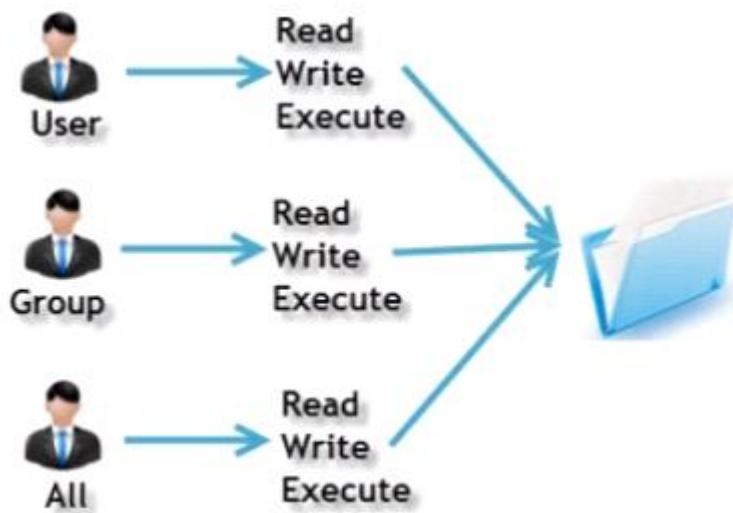
```
[terron@MyFirstLinuxVM ~]$ pwd  
/home/terron  
[terron@MyFirstLinuxVM ~]$ chmod a-x ejercicios/  
[terron@MyFirstLinuxVM ~]$ ls -lt ejercicios/  
total 0  
[terron@MyFirstLinuxVM ~]$ ls -lt ejercicios/  
total 0  
[terron@MyFirstLinuxVM ~]$ cd ejercicios/  
bash: cd: ejercicios/: Permission denied  
[terron@MyFirstLinuxVM ~]$
```

```
[terron@MyFirstLinuxVM ~]$ ls -l  
total 4  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Desktop  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Documents  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Downloads  
drw-rw-r--. 2 terron terron 6 Nov 29 03:31 ejercicios  
[terron@MyFirstLinuxVM ~]$ █
```

Removing executable permission.

# PERMISSIONS USING NUMERIC MODE

martes, 30 de noviembre de 2021 03:41 a. m.



You can also give permission to a file and directory assigned numerically, using the next command.

Number	Permission Type	Symbol
0	No Permission	---
1	Execute	--x
2	Write	-w-
3	Execute + Write	-wx
4	Read	r--
5	Read + Execute	r-x
6	Read +Write	r-w
7	Read + Write +Execute	rwx

NORMAL PERMISSION COMMAND	NUMERIC COMMAND	MEANING
chmod ugo+r FILE	chmod 444 FILE	GIVES PERMISSION TO EVERYONE TO READ THE FILE
chmod	chmod 764 FILE	GIVES PERMISSION TO THE OWNER TO RWX, GROUP TO RW, AND OTHER USERS ONLY TO R

```
[terron@MyFirstLinuxVM ~]$ pwd  
/home/terron  
[terron@MyFirstLinuxVM ~]$ touch sam  
[terron@MyFirstLinuxVM ~]$ chmod 764 sam  
[terron@MyFirstLinuxVM ~]$ ls -ltr sam  
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam  
[terron@MyFirstLinuxVM ~]$ █
```

# FILE OWNERSHIP COMMANDS (chown,chgrp)

miércoles, 1 de diciembre de 2021 12:26 a. m.

This commands are usable to know how the files and directories are owned and how we can change that ownership.

There are 2 owners of a file or directory.

- User
- Group

To change file ownership there are two commands

- chown changes the ownership of a file
- chgrp changes the group ownership of a file

Now if we want to change the ownership of a directory and also the files inside of that directory you have to use the recursive ownership option (cascade)

- -R

To change the owner you have to be superuser.

```
[root@MyFirstLinuxVM ~]# pwd  
/root  
[root@MyFirstLinuxVM ~]# cd /home/terron/  
[root@MyFirstLinuxVM terron]# ls -ltr lisa  
-rw-rw-r--. 1 terron terron 0 Nov 29 19:04 lisa  
[root@MyFirstLinuxVM terron]# chown root lisa  
[root@MyFirstLinuxVM terron]# ls -ltr lisa  
-rw-rw-r--. 1 root terron 0 Nov 29 19:04 lisa  
[root@MyFirstLinuxVM terron]# chgrp root lisa  
[root@MyFirstLinuxVM terron]# ls -ltr lisa  
-rw-rw-r--. 1 root root 0 Nov 29 19:04 lisa  
[root@MyFirstLinuxVM terron]# █
```

Remove files it is not allowed for the permissions

# ACCESS CONTROL LIST (ACL)

miércoles, 1 de diciembre de 2021 12:57 a. m.

Is an additional layer, which allows you to assign permission per user.

ACL provides an additional more flexible permissions mechanism for file systems. It is designed to assist with UNIX file permissions. ACL allows you to give permissions for any user or group to any disk resource.

We can use ACL in the next scenario:

Think of a scenario in which a particular user is not a member of group created by you but still you want to give some read or write access, how can you do it without making user a member of group, here comes ACL, helping us to do this trick.

ACL is used to make a flexible permission mechanism in LINUX

From LINUX man pages, ACLs are used to define more fine-grained discretionary access right for files and directories

Commands to assign and remove ACL permissions are:

- setfacl: set permissions
  - Getfacl: gives you information about the existing permissions of a file
1. To add permission for user: **setfacl -m u:user:rwx /path/to/file**
  2. To add permissions for a group: **setfacl -m g:group:rw /path/to/file**
  3. To allow all files or directories to inherit ACL entries from the directory it is within: **setfacl -rm "entry" /path/to/dir**
  4. To remove a specific entry: for an specific user **setfacl -x u:user /path/to/file**
  5. To remove all entries for all users: **setfacl -b path/to/file**

**NOTE:** As you assign the ACL permission to a file/directory it adds + sign at the end of the permission  
Setting w permission with ACL does not allow to remove a file.

-m is for modify your permissions

```
[root@MyFirstLinuxVM tmp]# getfacl testACLFile
# file: testACLFile
# owner: root
# group: root
user::rw-
group::r--
other::r--

[root@MyFirstLinuxVM tmp]# setfacl -m u:terron:rw /tmp/testACLFile
[root@MyFirstLinuxVM tmp]# getfacl testACLFile
# file: testACLFile
# owner: root
# group: root
user::rw-
user:terron:rw-
group::r--
mask::rw-
other::r--


[root@MyFirstLinuxVM tmp]# █
```

ACL is too important because it allows to assign a specific permission for a specific user.

To remove any user from ACL list we have to use the next command. If it is a one element of ACL you have to use the option -x

```
[root@MyFirstLinuxVM ~]# pwd
/root
[root@MyFirstLinuxVM ~]# cd ..
[root@MyFirstLinuxVM /]# cd tmp/
[root@MyFirstLinuxVM tmp]# getfacl testACLFile
# file: testACLFile
# owner: root
# group: root
user::rw-
user:terron:rw-
group::r--
mask::rw-
other::r--


[root@MyFirstLinuxVM tmp]# setfacl -x u:terron /tmp/testACLFile
[root@MyFirstLinuxVM tmp]# getfacl testACLFile
# file: testACLFile
# owner: root
# group: root
user::rw-
group::r--
mask::r--
other::r--


[root@MyFirstLinuxVM tmp]# █
```

If you want to remove anything from ACL you have to use the next command with the option -b.

```
[root@MyFirstLinuxVM tmp]# setfacl -b /tmp/testACLFile
[root@MyFirstLinuxVM tmp]# getfacl testACLFile
# file: testACLFile
# owner: root
# group: root
user::rw-
group::r--
other::r--


[root@MyFirstLinuxVM tmp]# █
```

# HELP COMMANDS

miércoles, 1 de diciembre de 2021 02:41 a. m.

That are used in LINUX environment that will give you some of the information

There are 3 types of help commands

- **whatis** command
- command **--help**
- **man** command

# TAB COMPLETION AND UP ARROW

miércoles, 1 de diciembre de 2021 02:57 a. m.

Hitting TAB key completes the available commands, files or directories

Hitting the up arrow key it will bring you the last ran command on the terminal

# ADDING TEXT TO FILES (REDIRECTS)

miércoles, 1 de diciembre de 2021 03:01 a. m.

How can we add text or populate an empty file.

There are 3 simply ways to add text to a file

- **vi**, editor that is used to create a file
- **redirect** command output > or >>
- **echo** > or >>

Echo makes echo of the exact same thing that you type.

If we make echo using one arrow > it means that the file will be overwritten and the content of the file will dissapear.

If we want to add a new line inside a file we need to use >> arrows.

```
[terron@MyFirstLinuxVM ~]$ cat george
[terron@MyFirstLinuxVM ~]$ echo "TERRON es un DIOS"
TERRON es un DIOS
[terron@MyFirstLinuxVM ~]$ echo "TERRON es un DIOS">>geo
[terron@MyFirstLinuxVM ~]$ echo "TERRON es un DIOS">>george
[terron@MyFirstLinuxVM ~]$ cat george
TERRON es un DIOS
[terron@MyFirstLinuxVM ~]$ echo "MOUNSTRO COME GALLETAS">>>george
[terron@MyFirstLinuxVM ~]$ echo "HOLA MUNDO">>>george
[terron@MyFirstLinuxVM ~]$ cat george
TERRON es un DIOS
MOUNSTRO COME GALLETAS
HOLA MUNDO
[terron@MyFirstLinuxVM ~]$ █
```

Now if we want to populate a file using an output of a command we can do it placing the next command.

```
[terron@MyFirstLinuxVM ~]$ ls -ltr > george
[terron@MyFirstLinuxVM ~]$ cat george
total 8
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Templates
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Public
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Music
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Downloads
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Documents
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Desktop
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Videos
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Pictures
drw-rw-r--. 2 terron terron 6 Nov 29 03:31 ejercicios
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lois
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lex
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 jerry
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 homer
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart
drwxrwxr-x. 2 terron terron 6 Nov 29 11:34 simpsons
-rw-rw-r--. 2 terron terron 20 Nov 29 11:35 jupiter
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam
```

# INPUT AND OUTPUT REDIRECTS (>,>>,<,STDIN,STDOUT AND STDERR)

miércoles, 1 de diciembre de 2021 03:27 a. m.

Input and output redirects. In LINUX there are 3 redirects in LINUX.

- Standard input (**stdin**) and it has file descriptor number as 0
- Standard output (**stdout**) and it has file descriptor number as 1
- Standard error (**stderr**) and it has file descriptor number as 2

CONSIDER EVERYTHING IN LINUX IS CONSIDERED AS A FILE}

## OUTPUT

- Output (**stdout**) - 1
  - By default when running a command its output goes to the terminal
  - The output of a command can be routed to a file using > symbol
  - If using the same file for additional output or to append to the same file then use >>
  - If you use one > the content of the file will be wiped

If we use the next command: ls -la it will show all the files including the hidden files, the hidden files begin with a dot

## INPUT

- Input (**stdin**) - 0
  - Input is used when feeding file contents to a file
    - cat < listings (same of using cat nameOfFile to see the content of a file)
    - mail -s "office memo" allusers@abc.com <nameOfFile

## ERROR

- Error (**stderr**) - 2
  - When a command is executed we use a keyboard and that is also considered standard input (**stdin** -1)
  - That command output goes on the monitor and that output is (**stdout**-1)
  - If the command produces any error on the screen then it is considered (**stderr**-2)
    - We can use redirects to route errors from the screen
      - ls -l /root 2>errorFile
      - Telnet localhost 2>errorFile

An standard error looks like the next picture

```
[terron@MyFirstLinuxVM ~]$ whoami
terron
[terron@MyFirstLinuxVM ~]$ ls -l /root/
ls: cannot open directory /root/: Permission denied
[terron@MyFirstLinuxVM ~]$
```

If we want to get the standard error we have to do the next:

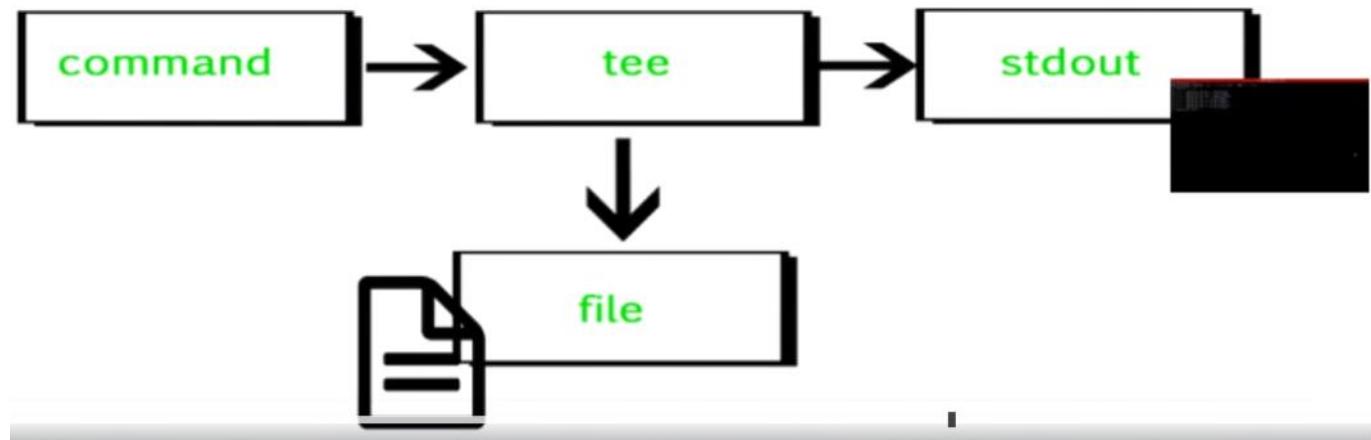
```
[terron@MyFirstLinuxVM ~]$ whoami
terron
[terron@MyFirstLinuxVM ~]$ ls -l /root/
ls: cannot open directory /root/: Permission denied
[terron@MyFirstLinuxVM ~]$ ls -l /root 2>fileError
[terron@MyFirstLinuxVM ~]$ cat fileError
ls: cannot open directory /root: Permission denied
[terron@MyFirstLinuxVM ~]$ █
```

# STANDARD OUTPUT TO A FILE (tee)

jueves, 2 de diciembre de 2021 01:14 a. m.

**tee** command is used to store and view (both at the same time) the output of any command.

**tee** command is named after the T-splitter used in plumbing. It basically breaks the output of a program so that it can be both displayed and saved in a file. It does both the task simultaneously, copies the result into the specified files or variables and also display the result.



This command is recomendly to use when you are trying to view and get at the same time the output.

To make sure that you are logged in the right system is needed to use 3 main commands  
whoami  
pwd  
hostname

```
[terron@MyFirstLinuxVM ~]$ whoami
terron
[terron@MyFirstLinuxVM ~]$ hostname
MyFirstLinuxVM
[terron@MyFirstLinuxVM ~]$ pwd
/home/terron
[terron@MyFirstLinuxVM ~]$ echo "LINUX COURSE" > newFILE
[terron@MyFirstLinuxVM ~]$ cat newFILE
LINUX COURSE
[terron@MyFirstLinuxVM ~]$ echo "NEW THINGS TO ADD"
NEW THINGS TO ADD
[terron@MyFirstLinuxVM ~]$ echo "NEW THINGS TO ADD" | tee newFILE
NEW THINGS TO ADD
[terron@MyFirstLinuxVM ~]$
```

If you want to append a text you have to use the -a option as the next command.

```
[terron@MyFirstLinuxVM ~]$ echo "NEW THINGS TO ADD OVER A NEW LINE" | tee -a newFILE
NEW THINGS TO ADD OVER A NEW LINE
[terron@MyFirstLinuxVM ~]$ cat newFILE
NEW THINGS TO ADD
NEW THINGS TO ADD OVER A NEW LINE
[terron@MyFirstLinuxVM ~]$
```

If we want to know how many characters we have on a file we can use wc command with -c option as the next picture.

```
[terron@MyFirstLinuxVM ~]$ wc -c newFile  
52 newFile
```

```
[terron@MyFirstLinuxVM ~]$ pwd  
/home/terron  
[terron@MyFirstLinuxVM ~]$ whoami  
terron  
[terron@MyFirstLinuxVM ~]$ hostname  
MyFirstLinuxVM  
[terron@MyFirstLinuxVM ~]$ ls -ltr *[a]* | tee newFile  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart  
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam  
  
Templates:  
total 0  
  
Downloads:  
total 0  
[terron@MyFirstLinuxVM ~]$ █
```

```
[terron@MyFirstLinuxVM ~]$ cat newFile  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart  
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam  
  
Templates:  
total 0  
  
Downloads:  
total 0  
[terron@MyFirstLinuxVM ~]$ █
```

We can use this to create multiple files as the next picture shows

```
[terron@MyFirstLinuxVM ~]$ ls -ltr *[a]* | tee newFile newFile1 newFile2  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart  
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam  
  
Templates:  
total 0  
  
Downloads:  
total 0  
[terron@MyFirstLinuxVM ~]$ █
```

# PIPES (|)

jueves, 2 de diciembre de 2021 03:00 a. m.

A pipe is used by the shell to connect the output of one command directly to the input of another command.

The syntax of the command is:

```
command1 [arguments] | command2 [arguments]
```

## MORE COMMAND

More command is a command that gives you the output of a command one page at a time

In combination with a pipe we can use it as the next way

```
[terron@MyFirstLinuxVM etc]$ pwd  
/etc  
[terron@MyFirstLinuxVM etc]$ ls -ltr | more
```

And the output is

```
terron@MyFirstLinuxVM:/etc  
File Edit View Search Terminal Help  
total 1352  
-rw-r--r--. 1 root root 91 Dec 3 2012 numad.conf  
-rw-r--r--. 1 root root 1634 Dec 24 2012 rpc  
-rw-r--r--. 1 root root 28 Feb 27 2013 ld.so.conf  
-rw-r--r--. 1 root root 670293 Jun 7 2013 services  
-rw-r--r--. 1 root root 233 Jun 7 2013 printcap  
-rw-r--r--. 1 root root 0 Jun 7 2013 motd  
-rw-r--r--. 1 root root 942 Jun 7 2013 inputrc  
-rw-r--r--. 1 root root 460 Jun 7 2013 hosts.deny  
-rw-r--r--. 1 root root 370 Jun 7 2013 hosts.allow  
-rw-r--r--. 1 root root 158 Jun 7 2013 hosts  
-rw-r--r--. 1 root root 9 Jun 7 2013 host.conf  
-rw-r--r--. 1 root root 0 Jun 7 2013 exports  
-rw-r--r--. 1 root root 662 Jul 31 2013 logrotate.conf  
-rw-r--r--. 1 root root 2391 Oct 12 2013 libuser.conf  
drwxr-xr-x. 2 root root 6 Jun 9 2014 cron.weekly  
-rw-r--r--. 1 root root 451 Jun 9 2014 crontab  
drwxr-xr-x. 2 root root 6 Jun 9 2014 cron.monthly  
-rw-r--r--. 1 root root 1362 Jun 9 2014 pbm2ppa.conf  
-rw-r--r--. 1 root root 6300 Jun 9 2014 pnm2ppa.conf  
-rw-r--r--. 1 root root 0 Jun 9 2014 wvdial.conf  
-rw-r--r--. 1 root root 1787 Jun 9 2014 request-key.conf  
drwxr-xr-x. 2 root root 6 Jun 9 2014 popt.d  
-rw-r--r--. 1 root root 2872 Jun 9 2014 pinforc  
--More--
```

To go to the other page you have to use space bar.

If you want to go to out you can press q key.

If you use ls command with option ls -l you can use instead ll command.

To get the last line of the command you can use the next command.

```
[terron@MyFirstLinuxVM etc]$ ls -ltr | tail -1  
drwxr-xr-x. 5 root lp      4096 Nov 30 06:46 cups
```

# FILE MAINTENANCE COMMANDS (cp,rm,mv,mkdir,rmdir)

jueves, 2 de diciembre de 2021 03:21 a. m.

- cp command is used to copy a file to another
- rm remove a file
- mv move the location of a file to another or to rename a file
- mkdir make a directory
- rmdir remove a directory, other way to remove a directory is to use rm -r command
- chgrp change the ownership of a file at a group level
- chown change the ownership of a file at user level

The syntax for cp command is: cp [file\_origin] /route or cp [file\_origin] [new\_name\_file]

The syntax command for mv to rename a file is: mv [name\_file] [new\_name\_file], to move the file to another directory you have to use is: mv [file\_name] /route

Also rm 'Rf command will forcefully remove sub'direcotries and its contents as well

# FILE DISPLAY COMMANDS (cat,less,more,head,tail)

jueves, 2 de diciembre de 2021 03:43 a. m.

This commands are used to display all the content of a file

- cat, show the entire content regardless if is one page or two or more pages
- more show the content one page at a time
- less, views the content of a file line by line in reverse order
- head, few lines at the top of a file placing - and a number representing the number of files that you want to see
- tail, few lines at the bottom of a file

If we use a dot with command cp or another it means at this location, as the next picture indicates.

```
[terron@MyFirstLinuxVM ~]$ su -
Password:
Last login: Mon Nov 29 20:53:05 CST 2021 on pts/0
[root@MyFirstLinuxVM ~]# pwd
/root
[root@MyFirstLinuxVM ~]# cp /var/log/messages .
[root@MyFirstLinuxVM ~]# ls -ltr messages
-rw-----. 1 root root 470699 Nov 30 07:41 messages
[root@MyFirstLinuxVM ~]# █
```

less command is used for show line by line pressing the enter key or arrow key, another one is to use space key but if you press space key is to change of page.

The syntax of head or tail command is: head -(#ofLines) name\_File  
head

# FILTERS/TEXT PROCESSORS COMMANDS

jueves, 2 de diciembre de 2021 03:56 a. m.

Text processor commands are very powerfull and gives you an advantage respect others OS.

- cut, allows you to cut the output, it takes the input of a command or if you re reading a content of a file and cut the input and place into the desired output
- awk, allows you to list by the columns
- grep and egrep, are just if you are looking for keywords
- sort, sorts out the output into alphabetical order
- uniq, will not show any duplicates
- wc, word count command, how many words, how many letters and hoy many lines

# CUT TEXT PROCESSOR COMMANDS

jueves, 2 de diciembre de 2021 11:55 p. m.

cut is a command line utility that allows you to cut parts of lines from specified files or piped data and print the result to standard output. It can be used to cut parts of a line by delimiter, byte position, and character

The syntax is:

cut fileName	Does not work
cut --version	Show the version of the command
cut -c1 filename	Only shows the first letter of every line inside the file
cut -c1,2,4	Pick and chose a character
cut -c1-3	List a range of characters
cut -b1-3 filename	List by byte size is same as characters
cut -d: -f 6 /etc/passwd	List first 6th column separated by : in a range of columns inside a text file
cut -d: -f 6-7 /etc/passwd	List 6th and 7th columns separated by :
ls -l   cut -c2-4	Only print user permissions of files/dir

```
[terron@MyFirstLinuxVM seinfeld]$ cat seinfeld-characters
Jerry Seinfeld
Cosmo Kramer
Elaine Benes
George Costanza
Newman mailman
Frank Costanza
Estelle Costanza
Morty Seinfeld
Aton Benes
J Peterman
George Steinbrenner
Uncel Leo
David PUddy
Justin Pitt
Kenny Bania
[terron@MyFirstLinuxVM seinfeld]$ cut -c1,2,3 seinfeld-characters
Jer
Cos
Eli
Geo
New
Fra
Est
Mor
Ato
J P
Geo
Unc
Dav
Jus
Ken
[terron@MyFirstLinuxVM seinfeld]$ █
```

```
[terron@MyFirstLinuxVM seinfeld]$ cut -c1-4 seinfeld-characters
Jerr
Cosm
Elia
Geor
Newm
Fran
Este
Mort
Aton
J Pe
Geor
Uncle
Davi
Just
Kenn
[terron@MyFirstLinuxVM seinfeld]$
```

```
[terron@MyFirstLinuxVM seinfeld]$ cut -c1-4,5-8 seinfeld-characters
Jerry Se
Cosmo Kr
Eliane B
George C
Newman m
Frank Co
Estelle
Morty Se
Aton Ben
J Peterm
George S
Uncel Le
David PU
Justin P
Kenny Ba
[terron@MyFirstLinuxVM seinfeld]$ █
```

/etc/passwd is the file that LINUX uses when you create a new user

```
[terron@MyFirstLinuxVM seinfeld]$ cut -d: -f 6 /etc/passwd
/root
/bin
/sbin
/var/adm
/var/spool/lpd
/sbin
/sbin
/sbin
/var/spool/mail
/root
/usr/games
/var/ftp
/
/
/
/
/var/run/lsm
/var/lib/colord
/var/lib/rpcbind
/usr/share/sane
/run/saslauthd
/etc/abrt
/var/lib/setroubleshoot
/proc
```

```
[terron@MyFirstLinuxVM seinfeld]$ cut -d: -f 6-7 /etc/passwd
/root:/bin/bash
/bin:/sbin/nologin
/sbin:/sbin/nologin
/var/adm:/sbin/nologin
/var/spool/lpd:/sbin/nologin
/sbin:/bin/sync
/sbin:/sbin/shutdown
/sbin:/sbin/halt
/var/spool/mail:/sbin/nologin
/root:/sbin/nologin
/usr/games:/sbin/nologin
/var/ftp:/sbin/nologin
/:/sbin/nologin
/:/sbin/nologin
/:/sbin/nologin
/:/sbin/nologin
```

# awk - TEXT PROCESSORS COMMANDS

viernes, 3 de diciembre de 2021 02:27 a. m.

Is a utility/language designed for data extraction. Mosto of the time it is used to extract fields from a file or from an output

It uses a delimiter, if the delimiter is : you can use that symbol but if is different you can use another symbol

- **awk --version** gives you the version of the command
- **awk '{print \$1}' nameFile** gives you the first column of a file, you can change the number if you want to get the first or second or n column from a file

```
[terron@MyFirstLinuxVM seinfeld]$ awk '{print $1}' seinfeld-characters
Jerry
Cosmo
Eliane
George
Newman
Frank
Estelle
Morty
Aton
J
George
Uncel
David
Justin
Kenny
[terron@MyFirstLinuxVM seinfeld]$ awk '{print $2}' seinfeld-characters
Seindfled
Kramer
Benes
Constanza
mailman
Costanza
Costanza
Seinfeld
Benes
Peterman
Steinbrenner
Leo
PUddy
PItt
Bania
[terron@MyFirstLinuxVM seinfeld]$ █
```

- **ls -l | awk '{print \$1,\$3}'** gives you the first and third column of the output of the first command

```
[terron@MyFirstLinuxVM seinfeld]$ ls -l | awk '{print $1,$3}'
total
-rw-rw-r--. terron
-rw-rw-r--. terron
-rw-rw-r--. terron
-rw-rw-r--. terron
-rw-rw-r--. terron
-
```

- To get the last column of the account you can use \$NF

```
[terron@MyFirstLinuxVM seinfeld]$ ls -l | awk '{print $NF}'  
4  
george  
jerry  
kramer  
puddy  
seinfeld-characters  
[terron@MyFirstLinuxVM seinfeld]$
```

- To search for an specific word of a file

```
[terron@MyFirstLinuxVM seinfeld]$ awk '/Jerry/ {print}' seinfeld-characters  
Jerry Seinfeld  
[terron@MyFirstLinuxVM seinfeld]$ awk '/Seinfeld/ {print}' seinfeld-characters  
Morty Seinfeld  
[terron@MyFirstLinuxVM seinfeld]$ █
```

- What if you wanted to get the list of the fields that are sepparated by a delimiter

```
[terron@MyFirstLinuxVM seinfeld]$ awk -F: '{print $1}' /etc/passwd  
root  
bin  
daemon  
adm  
lp  
sync  
shutdown  
halt  
mail  
operator  
games  
ftp  
nobody  
systemd-network  
dbus  
polkitd  
libstoragemgmt  
colord  
rpc  
saned  
saslauth  
abrt  
setroubleshoot  
rtkit
```

- If you want to replace a column that matches a certain keyword

```
[terron@MyFirstLinuxVM seinfeld]$ echo "Hello terron" | awk '{$2="TERRY"; print $0}'  
Hello TERRY  
██████████
```

- If I want to get lines that have more than 15 byte size we can use the next command

```
[terron@MyFirstLinuxVM seinfeld]$ awk 'length($0)>15' seinfeld-characters  
George Constanza  
Estelle Costanza  
George Steinbrenner  
[terron@MyFirstLinuxVM seinfeld]$
```

- Get the field that matches with a word

```
[terron@MyFirstLinuxVM ~]$ ls -l | awk '{if($NF == "seinfeld") print ($0);}'  
drwxrwxr-x. 2 terron terron 88 Nov 30 07:58 seinfeld  
[terron@MyFirstLinuxVM ~]$
```

- If you want to number all the fields you can use the next command

# grep/egrep TEXT PROCESSOR COMMANDS

domingo, 5 de diciembre de 2021 02:00 p. m.

The grep command which stands for "global regular expression print", processes text line by line and prints any lines which match a specified pattern.

Is a search feature in linux that allows you to search for a specific pattern.

- `grep --version OR grep --help` = Check version or help
- `grep keyword file` = Search for a keyword from a file
- `grep -c keyword file` = Search for a keyword and count
- `grep -i KEYword file` = Search for a keyword ignore case-sensitive
- `grep -n keyword file` = Display the matched lines and their line numbers
- `grep -v keyword file` = Display everything but keyword
- `grep keyword file | awk '{print $1}'` = Search for a keyword and then only give the 1<sup>st</sup> field
- `ls -l | grep Desktop` = Search for a keyword and then only give the 1<sup>st</sup> field

To find a specific word we use:

```
[terron@MyFirstLinuxVM ~]$ cat newFile
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam
```

Templates:

```
total 0
```

Downloads:

```
total 0
```

```
[terron@MyFirstLinuxVM ~]$ grep marge newFile
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge
[terron@MyFirstLinuxVM ~]$
```

To count the coincidences we use:

```
[terron@MyFirstLinuxVM ~]$ cat newFile
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam
```

Templates:

```
total 0
```

Downloads:

```
total 0
```

```
[terron@MyFirstLinuxVM ~]$ grep marge newFile
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge
[terron@MyFirstLinuxVM ~]$ grep -c rw newFile
```



To find a specific word and it does not care about if it is uppercase or lowercase we use option -i

```
-----  
[terron@MyFirstLinuxVM ~]$ grep marge newFile  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge  
[terron@MyFirstLinuxVM ~]$ grep -i rw newFile  
5  
[terron@MyFirstLinuxVM ~]$ echo "RW" >> newFile  
[terron@MyFirstLinuxVM ~]$ grep -i rw newFile  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart  
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam  
RW  
[terron@MyFirstLinuxVM ~]$
```

If we want to see the coincidences and the lines we use -n option:

```
[terron@MyFirstLinuxVM ~]$ grep -c rw newFile  
5  
[terron@MyFirstLinuxVM ~]$ grep -ni rw newFile  
1:-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge  
2:-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer  
3:-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark  
4:-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 bart  
5:-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam  
12:RW  
[terron@MyFirstLinuxVM ~]$ █
```

If we want to see the no coincidences we use -v option, is like if we are saying to the system, give me everything except this search keyword

```
[terron@MyFirstLinuxVM ~]$ echo "no coincidence" >> newFile  
[terron@MyFirstLinuxVM ~]$ grep -v rw newFile  
  
Templates:  
total 0  
  
Downloads:  
total 0  
RW  
no coincidence  
[terron@MyFirstLinuxVM ~]$ █
```

We can combine with pipe the commands, for example if I only want the first column of the file we use:



```
[terron@MyFirstLinuxVM ~]$ grep -v rw newFILE | awk '{print $1}'
```

Templates:

total

Downloads:

total

RW

no

If we want to get only the first 3 characters we use:

```
[terron@MyFirstLinuxVM ~]$ grep -v rw newFILE | awk '{print $1}' | cut -c1-3
```

Tem

tot

Dow

tot

RW

no

```
[terron@MyFirstLinuxVM ~]$ █
```

We can use to find a directory using the next command:

```
[terron@MyFirstLinuxVM ~]$ ls -l | grep desktop
[terron@MyFirstLinuxVM ~]$ ls -l | grep -i desktop
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Desktop
[terron@MyFirstLinuxVM ~]$ █
```

If we want to find is two words we have to use egrep, as the next picture:

```
[terron@MyFirstLinuxVM ~]$ egrep -i "marge|kramer" newFILE
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer
[terron@MyFirstLinuxVM ~]$ █
```

```
[terron@MyFirstLinuxVM ~]$ egrep -i "marge|kramer|sam" newFILE
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 marge
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 kramer
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam
[terron@MyFirstLinuxVM ~]$ █
```



# sort/uniq TEXT PROCESSORS COMMANDS

domingo, 5 de diciembre de 2021 02:36 p. m.

- sort command sorts in alphabetical order
- uniq commands filters out the repeated or duplicate lines

```
[terron@MyFirstLinuxVM ~]$ whoami
terron
[terron@MyFirstLinuxVM ~]$ hostname
MyFirstLinuxVM
[terron@MyFirstLinuxVM ~]$ pwd
/home/terron
[terron@MyFirstLinuxVM ~]$ sort --version
sort (GNU coreutils) 8.22
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
```

Written by Mike Haertel and Paul Eggert.

```
[terron@MyFirstLinuxVM ~]$ █
```

If we want to sort this file we have to use, remember that we can use reverse option:

```
[terron@MyFirstLinuxVM seinfeld]$ sort seinfeld-characters
Aton Benes
Cosmo Kramer
David PUddy
Eliane Benes
Estelle Costanza
Frank Costanza
George Constanza
George Steinbrenner
Jerry Seindfled
J Peterman
Justin Pitt
Kenny Bania
Morty Seinfeld
Newman mailman
Uncel Leo
[terron@MyFirstLinuxVM seinfeld]$ █
```

Now if we want to sort by number of column of field we can use -k# of the field or column as the next picture:

```
[terron@MyFirstLinuxVM seinfeld]$ sort -k2 seinfeld-characters
Kenny Bania
Aton Benes
Eliane Benes
George Constanza
Estelle Costanza
Frank Costanza
Cosmo Kramer
Uncel Leo
Newman mailman
J Peterman
Justin Pitt
David PUddy
Morty Seinfeld
Jerry Seindfled
George Steinbrenner
[terron@MyFirstLinuxVM seinfeld]$
```

Now, if we want to sort the characters using the last column as reference we can use the next command: \$NF argument

```
[terron@MyFirstLinuxVM ~]$ ls -l | sort $NF
drw-rw-r--. 2 terron terron      6 Nov 29 03:31 ejercicios
drwxrwxr-x. 2 terron terron      6 Nov 29 11:34 simpsons
drwxrwxr-x. 2 terron terron     88 Nov 30 07:58 seinfeld
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Desktop
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Documents
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Downloads
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Music
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Pictures
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Public
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Templates
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Videos
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 bart
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 clark
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 homer
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 jerry
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 kramer
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 lex
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 lois
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 marge
-rw-rw-r--. 1 terron terron    1160 Nov 29 21:44 george
-rw-rw-r--. 1 terron terron      18 Nov 29 21:25 geo
-rw-rw-r--. 1 terron terron     283 Nov 30 06:52 newFile1
-rw-rw-r--. 1 terron terron     283 Nov 30 06:52 newFile2
-rw-rw-r--. 1 terron terron     301 Dec  1 01:54 newFile
-rw-rw-r--. 1 terron terron      51 Nov 29 22:16 fileError
-rw-rw-r--. 2 terron terron     20 Nov 29 11:35 jupiter
-rwxrw-r--. 1 terron terron      0 Nov 29 18:36 sam
total 28
[terron@MyFirstLinuxVM ~]$ █
```

Every time that you will use uniq command you have to use sort command first, then add using pipe symbol as the next picture indicates. IF YOU DONT USE SORT COMMAND FIRST THE UNIQ COMMAND WILL NEVER FUNCTION

```
[terron@MyFirstLinuxVM seinfeld]$ echo "Jerry Seindfled" >>seinfeld-characters
[terron@MyFirstLinuxVM seinfeld]$ uniq seinfeld-characters
Jerry Seindfled
Cosmo Kramer
Eliane Benes
George Constanza
Newman mailman
Frank Costanza
Estelle Costanza
Morty Seindfeld
Aton Benes
J Peterman
George Steinbrenner
Uncel Leo
David PUddy
Justin PItt
Kenny Bania
Jerry Seindfled
[terron@MyFirstLinuxVM seinfeld]$ sort seinfeld-characters | uniq
Aton Benes
Cosmo Kramer
David PUddy
Eliane Benes
Estelle Costanza
Frank Costanza
George Constanza
George Steinbrenner
Jerry Seindfled
J Peterman
Justin PItt
Kenny Bania
Morty Seindfeld
Newman mailman
Uncel Leo
[terron@MyFirstLinuxVM seinfeld]$ █
```

If we want to see the coincidences for a line we have to use the uniq option -c

```
[terron@MyFirstLinuxVM seinfeld]$ sort seinfeld-characters | uniq -c
 1 Aton Benes
 1 Cosmo Kramer
 1 David PUddy
 1 Eliane Benes
 1 Estelle Costanza
 1 Frank Costanza
 1 George Constanza
 1 George Steinbrenner
 2 Jerry Seindfled
 1 J Peterman
 1 Justin PItt
 1 Kenny Bania
 1 Morty Seindfeld
 1 Newman mailman
 1 Uncel Leo
```

To show only the repeated lines we can use:

-

```
[terron@MyFirstLinuxVM seinfeld]$ sort seinfeld-characters | uniq -cd
 2 Jerry Seinfeld
[terron@MyFirstLinuxVM seinfeld]$ sort seinfeld-characters | uniq -d
Jerry Seinfeld
[terron@MyFirstLinuxVM seinfeld]$ █
```

---

# wc-TEXT PROCESSOR COMMANDS

domingo, 5 de diciembre de 2021 02:55 p. m.

Stand for word count, reads either standard input or a list of files and generates: **newline count**, **word count** and **byte count**.

- |  |  |
|--|--|
| • <code>wc --version OR wc --help</code> | = Check version or help                            |
| • <code>wc file</code>                   | = Check file line count, word count and byte count |
| • <code>wc -l file</code>                | = Get the number of lines in a file                |
| • <code>wc -w file</code>                | = Get the number of words in a file                |
| • <code>wc -b file</code>                | = Get the number of bytes in a file                |
| • <code>wc DIRECTORY</code>              | = NOT allowed                                      |
| • <code>ls -l   wc -l</code>             | = Number of files                                  |
| • <code>grep keyword   wc -l</code>      | = Number of keyword lines.                         |

```
[terron@MyFirstLinuxVM seinfeld]$ wc -c seinfeld-characters
226 seinfeld-characters
[terron@MyFirstLinuxVM seinfeld]$ wc seinfeld-characters
16 32 226 seinfeld-characters
[terron@MyFirstLinuxVM seinfeld]$
```

In the first column we get number of lines, in the second column we find number of words and in the third column we find number of characters.

```
[terron@MyFirstLinuxVM seinfeld]$ wc -c seinfeld-characters
226 seinfeld-characters
[terron@MyFirstLinuxVM seinfeld]$ wc seinfeld-characters
16 32 226 seinfeld-characters
[terron@MyFirstLinuxVM seinfeld]$ wc -l seinfeld-characters
16 seinfeld-characters
[terron@MyFirstLinuxVM seinfeld]$ wc -2 seinfeld-characters
wc: invalid option -- '2'
Try 'wc --help' for more information.
[terron@MyFirstLinuxVM seinfeld]$ wc -w seinfeld-characters
32 seinfeld-characters
[terron@MyFirstLinuxVM seinfeld]$
```

If you want to get the number of files in a directory you can use:

```
[terron@MyFirstLinuxVM ~]$ ls -l |wc -l
28
[terron@MyFirstLinuxVM ~]$
```

But is better to use:

```
[terron@MyFirstLinuxVM ~]$ ls -l | wc -l
28
[terron@MyFirstLinuxVM ~]$ ls -l | grep drw | wc -l
11
[terron@MyFirstLinuxVM ~]$ ls -l | grep drw
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Desktop
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Documents
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Downloads
drw-rw-r--. 2 terron terron 6 Nov 29 03:31 ejercicios
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Music
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Pictures
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Public
drwxrwxr-x. 2 terron terron 88 Nov 30 07:58 seinfeld
drwxrwxr-x. 2 terron terron 6 Nov 29 11:34 simpsons
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Templates
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Videos
[terron@MyFirstLinuxVM ~]$ █
```

Now if you want to go for the number of coincidences you can merge the commands as the next picture:

```
[terron@MyFirstLinuxVM seinfeld]$ grep -i sein seinfeld-characters | wc -l
3
[terron@MyFirstLinuxVM seinfeld]$ grep -i sein seinfeld-characters
Jerry Seinfeld
Morty Seinfeld
Jerry Seinfeld
[terron@MyFirstLinuxVM seinfeld]$ █
```

# COMPARE FILES (diff and cmp)

domingo, 5 de diciembre de 2021 03:39 p. m.

- diff, this command will compare line by line of two files
- cmp, this command will compare byte by byte

```
[terron@MyFirstLinuxVM ~]$ mkdir superman
[terron@MyFirstLinuxVM ~]$ cd superman/
[terron@MyFirstLinuxVM superman]$ echo "clark kent" > superman-characters
[terron@MyFirstLinuxVM superman]$ cat superman-characters
clark kent
[terron@MyFirstLinuxVM superman]$ echo "louis lane" >> superman-characters
[terron@MyFirstLinuxVM superman]$ echo "General Zod" >> superman-characters
[terron@MyFirstLinuxVM superman]$ echo "clark kent" > superman-characters2
[terron@MyFirstLinuxVM superman]$ echo "louis lane" >> superman-characters2
[terron@MyFirstLinuxVM superman]$ echo "General Zod" >> superman-characters2
[terron@MyFirstLinuxVM superman]$ nano superman-characters2
[terron@MyFirstLinuxVM superman]$ diff superman-characters superman-characters2
3c3
< General Zod
---
> General Peter
[terron@MyFirstLinuxVM superman]$ cmp superman-characters superman-characters2
superman-characters superman-characters2 differ: byte 31, line 3
[terron@MyFirstLinuxVM superman]$ █
```

# COMPRESS AND UN-COMPRESS FILES

domingo, 5 de diciembre de 2021 03:45 p. m.

The purpose of this command is to send a file that had a huge size

- tar, takes a bunch of files together and puts into a file like zip in windows
- gzip, compresses a file
- gzip -d or gunzip, un-compress a file

```
[terron@MyFirstLinuxVM seinfeld] $ tar cvf everything-compressed.tar .  
./  
. /seinfeld-characters  
. /kramer  
. /george  
. /jerry  
. /puddy  
tar: ./everything-compressed.tar: file is the archive; not dumped  
[terron@MyFirstLinuxVM seinfeld] $ ls  
everything-compressed.tar george jerry kramer puddy seinfeld-characters  
[terron@MyFirstLinuxVM seinfeld] $ ls -l  
total 16  
-rw-rw-r--. 1 terron terron 10240 Dec 1 03:19 everything-compressed.tar  
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 george  
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 jerry  
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 kramer  
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 puddy  
-rw-rw-r--. 1 terron terron 226 Dec 1 02:18 seinfeld-characters  
[terron@MyFirstLinuxVM seinfeld] $ mkdir decompressed  
[terron@MyFirstLinuxVM seinfeld] $ mv everything-compressed.tar decompressed/  
[terron@MyFirstLinuxVM seinfeld] $ cd decompressed/  
[terron@MyFirstLinuxVM decompressed] $ ls  
everything-compressed.tar  
[terron@MyFirstLinuxVM decompressed] $ tar xvf everything-compressed.tar  
./  
. /seinfeld-characters  
. /kramer  
. /george  
. /jerry  
. /puddy  
[terron@MyFirstLinuxVM decompressed] $ ls  
everything-compressed.tar george jerry kramer puddy seinfeld-characters  
[terron@MyFirstLinuxVM decompressed] $ █
```

To compress the file we have to use the next command:

```
[terron@MyFirstLinuxVM decompressed] $ gzip everything-compressed.tar
```

And passes from this size:

```
[terron@MyFirstLinuxVM seinfeld] $ ls -l  
total 16  
-rw-rw-r--. 1 terron terron 10240 Dec 1 03:19 everything-compressed.tar  
To:  
-rw-rw-r--. 1 terron terron 414 Dec 1 03:19 everything-compressed.tar.gz
```

To un-compress the file we have to use:

```
[terron@MyFirstLinuxVM decompressed]$ gzip -d everything-compressed.tar.gz
[terron@MyFirstLinuxVM decompressed]$ ls -ltr
total 16
-rw-rw-r--. 1 terron terron      0 Nov 30 07:58 putty
-rw-rw-r--. 1 terron terron      0 Nov 30 07:58 kramer
-rw-rw-r--. 1 terron terron      0 Nov 30 07:58 jerry
-rw-rw-r--. 1 terron terron      0 Nov 30 07:58 george
-rw-rw-r--. 1 terron terron    226 Dec  1 02:18 seinfeld-characters
-rw-rw-r--. 1 terron terron 10240 Dec  1 03:19 everything-compressed.tar
[terron@MyFirstLinuxVM decompressed]$
```

## TRUNCATE FILESIZE (truncate)

domingo, 5 de diciembre de 2021 03:57 p. m.

**truncate** command is often used to shrink or extend the size of a file to the specified size, it will create a losse of info.

```
[terron@MyFirstLinuxVM seinfeld]$ ls -l
total 4
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 george
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 jerry
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 kramer
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 pudgy
-rw-rw-r--. 1 terron terron 226 Dec 1 02:18 seinfeld-characters
[terron@MyFirstLinuxVM seinfeld]$ touch seinfeldwords
[terron@MyFirstLinuxVM seinfeld]$ ls -ltr seinfeldwords
-rw-rw-r--. 1 terron terron 0 Dec 1 03:30 seinfeldwords
[terron@MyFirstLinuxVM seinfeld]$ echo "puffyshirt giddyup yadayada kavorka serenitynow festivus" >seinfeldwords
[terron@MyFirstLinuxVM seinfeld]$ cat seinfeldwords
puffyshirt giddyup yadayada kavorka serenitynow festivus
[terron@MyFirstLinuxVM seinfeld]$ ls -l seinfeldwords
-rw-rw-r--. 1 terron terron 57 Dec 1 03:31 seinfeldwords
[terron@MyFirstLinuxVM seinfeld]$ truncate -s 40 seinfeldwords
[terron@MyFirstLinuxVM seinfeld]$ ls -ltr seinfeldwords
-rw-rw-r--. 1 terron terron 40 Dec 1 03:32 seinfeldwords
[terron@MyFirstLinuxVM seinfeld]$ cat seinfeldwords
puffyshirt giddyup yadayada kavorka sere[terron@MyFirstLinuxVM seinfeld]$ █
```

# COMBINING AND SPLITTING FILES

domingo, 5 de diciembre de 2021 04:05 p. m.

- Multiple files can be combined into one
- One file can be split into multiple files
  - cat file1 file2 file3 > file 4
  - split file14
  - e.g. split -l 300 file.txt childfile

Split file.txt into 300 lines per file and output to childfileaa, childfileab and childfileac

```
[terron@MyFirstLinuxVM seinfeld]$ echo "USA" >countries
[terron@MyFirstLinuxVM seinfeld]$ echo UK >> countries
[terron@MyFirstLinuxVM seinfeld]$ echo UAE >> countries
[terron@MyFirstLinuxVM seinfeld]$ echo CANADA >> countries
[terron@MyFirstLinuxVM seinfeld]$ echo FRANCE >> countries
[terron@MyFirstLinuxVM seinfeld]$ echo SWITZERLAND >> countries
[terron@MyFirstLinuxVM seinfeld]$ echo JAPAN >> countries
[terron@MyFirstLinuxVM seinfeld]$ cat countries | wc -l
7
[terron@MyFirstLinuxVM seinfeld]$ split -l 2 countries sepparatedLnes
[terron@MyFirstLinuxVM seinfeld]$ ls -ltr
total 28
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 puddy
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 kramer
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 jerry
-rw-rw-r--. 1 terron terron 0 Nov 30 07:58 george
-rw-rw-r--. 1 terron terron 226 Dec 1 02:18 seinfeld-characters
-rw-rw-r--. 1 terron terron 40 Dec 1 03:32 seinfeldwords
-rw-rw-r--. 1 terron terron 43 Dec 1 03:47 countries
-rw-rw-r--. 1 terron terron 6 Dec 1 03:48 sepparatedLnesad
-rw-rw-r--. 1 terron terron 19 Dec 1 03:48 sepparatedLnesac
-rw-rw-r--. 1 terron terron 11 Dec 1 03:48 sepparatedLnesab
-rw-rw-r--. 1 terron terron 7 Dec 1 03:48 sepparatedLnesaa
[terron@MyFirstLinuxVM seinfeld]$
[terron@MyFirstLinuxVM seinfeld]$
```

If we do cat command in the new files that has been created we'll get:

```
[terron@MyFirstLinuxVM seinfeld]$ cat sepparatedLnesaa
USA
UK
[terron@MyFirstLinuxVM seinfeld]$ cat sepparatedLnesab
UAE
CANADA
[terron@MyFirstLinuxVM seinfeld]$ cat sepparatedLnesac
FRANCE
SWITZERLAND
[terron@MyFirstLinuxVM seinfeld]$ cat sepparatedLnesad
JAPAN
[terron@MyFirstLinuxVM seinfeld]$
```

# LINUX VS WINDOWS COMMANDS

domingo, 5 de diciembre de 2021 04:23 p. m.

Differences:

Command Description	Windows	Linux
Listing of a directory	dir	ls -l
Rename a file	ren	mv
Copy a file	copy	cp
Move file	move	mv
Clear screen	<b>cls</b>	clear
Delete file	del	rm
Compare contents of files	fc	diff
Search for a word/string in a file	find	grep
Display command help	<b>command /?</b>	man command
Displays your location in the file system	<b>chdir</b>	pwd
Displays the time	<b>time</b>	date

## SUMMARY

domingo, 5 de diciembre de 2021 04:35 p. m.

- To search for a key word or pattern in a file we use grep command.
- The purpose of sort command is sort the output of a command or sort the output of a file
- What is the output of command wc -l file.txt if the file contents are:
  - Firstline
  - Secondline
  - Thirdline
  - fourthline
  - THE ANSWER IS: 4 file.txt
- There are 10 lines in a file, how do you print lines between 6 and 9 (all inclusive)

```
[terron@MyFirstLinuxVM seinfeld]$ cat TestFILE |tail -5 | head -6
line6
line7
line8
line9
line10
[terron@MyFirstLinuxVM seinfeld]$ cat TestFILE |tail -5 | head -1
line6
[terron@MyFirstLinuxVM seinfeld]$ cat TestFILE |tail -5 | head -4
line6
line7
line8
line9
[terron@MyFirstLinuxVM seinfeld]$
```

You're saying to the system that you want to have the output from all from 5 to 10 and only

If we want to see only ten lines of an output we can use,

```
[terron@MyFirstLinuxVM ~]$ head -10 logsFileOutput
Nov 26 01:23:41 MyFirstLinuxVM journal: Runtime journal is
mit 49.5M).
Nov 26 01:23:41 MyFirstLinuxVM kernel: Initializing cgroup
Nov 26 01:23:41 MyFirstLinuxVM kernel: Initializing cgroup
Nov 26 01:23:41 MyFirstLinuxVM kernel: Initializing cgroup
Nov 26 01:23:41 MyFirstLinuxVM kernel: Linux version 3.10.(4.8.5-44) (GCC) #1 SMP Mon Oct 19 16:18:59 UTC 2020
Nov 26 01:23:41 MyFirstLinuxVM kernel: Command line: BOOT_
vm.lv=centos/root rd.lvm.lv=centos/swap rhgb quiet LANG=en_
Nov 26 01:23:41 MyFirstLinuxVM kernel: e820: BIOS-provided
Nov 26 01:23:41 MyFirstLinuxVM kernel: BIOS-e820: [mem 0x00
Nov 26 01:23:41 MyFirstLinuxVM kernel: BIOS-e820: [mem 0x00
Nov 26 01:23:41 MyFirstLinuxVM kernel: BIOS-e820: [mem 0x00
[terron@MyFirstLinuxVM ~]$ █
```

If we want to get the output from a command to save it into a file we can use tee command

```
[root@MyFirstLinuxVM ~]# cat /var/log/messages | tee /home/terron/logsFileOutput█
```

Use cut command to cut the first 4 letters of each line from seinfeld-characters file and output to a different file name (name = filters-files)

To print only one number of columns we have to use the next command:

```
[terron@MyFirstLinuxVM seinfeld]$ awk '{print $2}' seinfeld-characters
Seinfeld
Kramer
Benes
Constanza
mailman
Costanza
Costanza
Seinfeld
Benes
Peterman
Steinbrenner
Leo
PUddy
PITT
Bania
Seinfeld
[terron@MyFirstLinuxVM seinfeld]$ █
```

---

### Example of use of grep

```
[terron@MyFirstLinuxVM seinfeld]$ grep -i seinfeld seinfeld-|tee seinfeld-family
seinfeld-characters  seinfeld-family
[terron@MyFirstLinuxVM seinfeld]$ grep -i seinfeld seinfeld-characters|tee seinfeld-family
Morty Seinfeld
[terron@MyFirstLinuxVM seinfeld]$ cat seinfeld-family
Morty Seinfeld
[terron@MyFirstLinuxVM seinfeld]$ █
```

# SYSTEM INFORMATION

domingo, 5 de diciembre de 2021 05:28 p. m.

We can get the information about the system using the next commands:

- `uname -a` it gives the kernel name, version, hostname, etc you can find all the details in respective files under the proc filesystem, it is easier to use `uname` utility to get these information quickly
- `dmidecode`, is a tool for dumping a computer's DMI (SMBIOS) table in contents in a human-readable format. This table contains a description of the system's hardware components, as well as other useful pieces of information such as serial numbers and BIOS revision. Thanks to this table, you can retrieve this information without having to probe for the actual hardware.

# LINUX FILE EDITOR

domingo, 5 de diciembre de 2021 05:32 p. m.

A text editor is a program which enables you to create and manipulate data (text) in a Linux file. There are several standard text editors available on most LINUX systems

- vi-visual editor
- ed- standard line editor
- ex - extended line editor
- emacs - a full screen editor
- pico - beginner's editor
- vim - advanced version of vi

vi is available in almost every LINUX distribution

VI

## • vi supplies commands for:

- Inserting and deleting text
- Replacing text
- Moving around the file
- Finding and substituting strings
- Cutting and pasting text

## • Most common keys:

- **i** – **insert**
- **Esc** – **Escape out of any mode**
- **r** – **replace**
- **d** – **delete**
- **:q!** – **quit without saving**
- **:wq!** – **quit and save**

When you create a file using vi at the beginnign you will not into insert mode, to change to insert mode you have to use i key . Once you have completed to insert lines you can use shift + zz. This will save all the file.

```
[root@MyFirstLinuxVM ~]# exit
logout
[terron@MyFirstLinuxVM seinfeld]$ vi myFILEVI
[terron@MyFirstLinuxVM seinfeld]$ cat myFILEVI
Hello TERRON
This is my first lesson to learn vi
I'm so excited about it
[terron@MyFirstLinuxVM seinfeld]$
```

The second way to close vi is using : symbol.

Hello TERRON  
This is my first lesson to learn vi  
I'm so excited about it

FM

卷之三

And 'wq!

The equivalence to `ctrl+Z` in vi editor is to press `u` key, to remove a completely line you have to use `d` key and press two times.

To delete a letter you can use in normal mode x key

If you want to add a new line down to the other line you can use o key in normal mode.

If you want to search for an specific word you can use grep command or hit slash and type the word.

Hello TERRON  
This is my first **lesson** to learn vi  
I'm so excited about it  
NEW LINE INSERTED USING O KEY

FM

卷之三

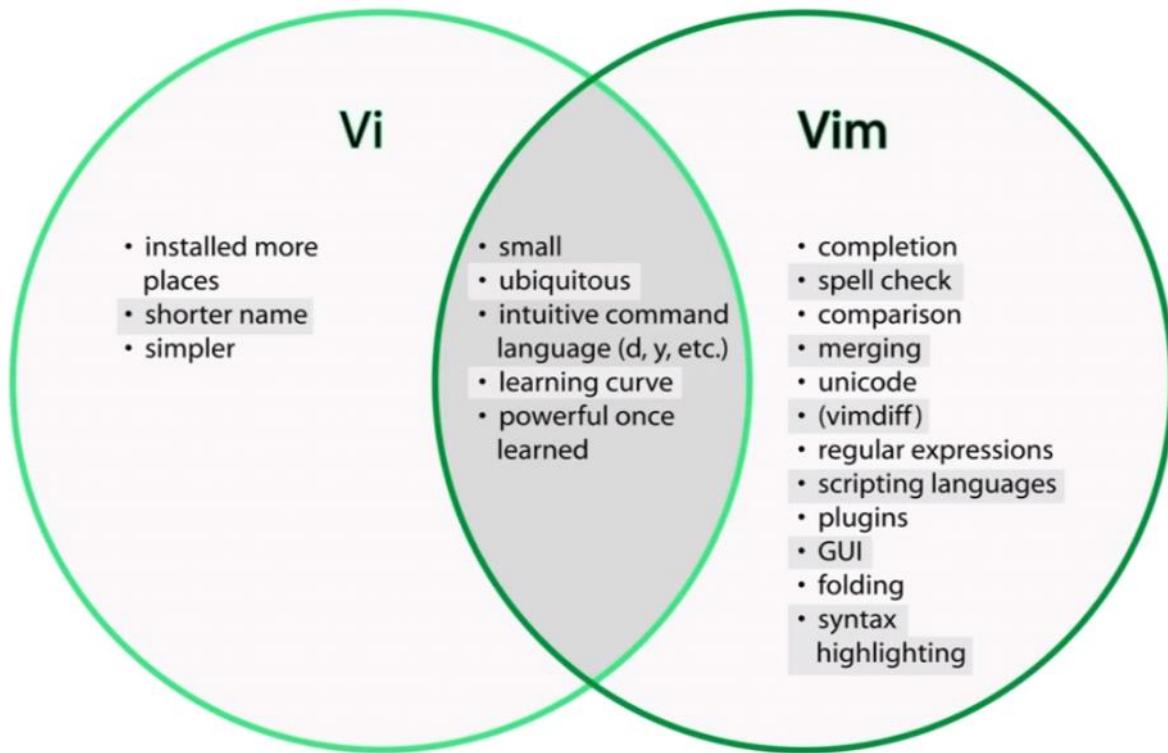
search hit BOTTOM, continuing at TOP

# DIFFERENCE BETWEEN VI AND VIM EDITOR

lunes, 6 de diciembre de 2021 03:03 a. m.

Both editors work in the same manner. Some people recommend learning the vim editor instead of the vi editor. Due to added features, learning and using vim editor is much easier than the vi editor.

- VIM is based on the vi editor
- VIM has all the features as VI with some additions



h,j,k,l keys are used to move.

To navigate the text in terms of words, you can use keys w,b,e, w moves to the start of the next word, e moves to the end of the word, b moves to the beginning of the word.

# "sed" COMMAND

lunes, 6 de diciembre de 2021 03:16 a. m.

Is used to manipulate the content of a file, some of the uses are:

- Replace a string in a file with a new string
- Find and delete a line
- Remove empty lines from a file
- Remove the first or n lines in a file
- To replace tabs with spaces
- Show defined lines from a file
- Substitute within vi editor

If we want to substitute a word with another we can use the next syntax.

- sed 's/FirstWord/ReplaceWord/g' fileName
  - s is for substitute, g is of global

```
[terron@MyFirstLinuxVM seinfeld]$ cat seinfeld-characters
Jerry Seinfeld
Cosmo Kramer
Eliane Benes
George Constanza
Newman mailman
Frank Costanza
Estelle Costanza
Morty Seinfeld
Aton Benes
J Peterman
George Steinbrenner
Uncle Leo
David Puddy
Justin Pitt
Kenny Bania
Jerry Seinfeld
[terron@MyFirstLinuxVM seinfeld]$ sed 's/Kenny/Lenny/g' seinfeld-characters
Jerry Seinfeld
Cosmo Kramer
Eliane Benes
George Constanza
Newman mailman
Frank Costanza
Estelle Costanza
Morty Seinfeld
Aton Benes
J Peterman
George Steinbrenner
Uncle Leo
David Puddy
Justin Pitt
Lenny Bania
Jerry Seinfeld
```

To make a change in the file we have to use -i option

```
[terron@MyFirstLinuxVM seinfeld]$ sed -i 's/Kenny/Lenny/g' seinfeld-characters
[terron@MyFirstLinuxVM seinfeld]$ █
```

To delete a line that have a coincidence we can use /d configuration as the next command indicate:

```
[terron@MyFirstLinuxVM seinfeld]$ sed '/Seinfeld/d' seinfeld-characters
Jerry Seinfeld
Cosmo Kramer
Eliane Benes
George Constanza
Newman mailman
Frank Constanza
Estelle Constanza
Aton Benes
J Peterman
George Steinbrenner
Uncel Leo
David PUddy
Justin Pitt
Lenny Bania
Jerry Seinfeld
[terron@MyFirstLinuxVM seinfeld]$ █
```

To remove empty files of a file we can use the next command:

REMEMBER THAT ^ IS TO INDICATE OF ANYTHING THAT STARTS, \$ MEAN NOTHING

```
[terron@MyFirstLinuxVM seinfeld]$ sed '/^$/d' seinfeld-characters
```

If I want to remove the first line of a file we can use:

```
[terron@MyFirstLinuxVM seinfeld]$ sed '1d' seinfeld-characters
Cosmo Kramer
```

We can del a range of lines using the next command:

```
[terron@MyFirstLinuxVM seinfeld]$ sed '1,10d' seinfeld-characters
George Steinbrenner
Uncel Leo
David PUddy
Justin Pitt
Lenny Bania
Jerry Seinfeld
```

If we want to del tabulator characters we can use:

```
[terron@MyFirstLinuxVM seinfeld]$ sed 's/\t/ /g' seinfeld-characters
Cosmo Kramer
```

To get a range of lines we can get the next command, remeber that p option is used to pick:

```
[terron@MyFirstLinuxVM seinfeld]$ sed -n 12,18p seinfeld-characters
Uncel Leo
David PUddy
Justin Pitt
Lenny Bania
Jerry Seinfeld
```

If we want to exclude a certain range of lines we can use the next command:

```
[terron@MyFirstLinuxVM seinfeld]$ sed 12,18d seinfeld-characters
Jerry Seinfeld
Cosmo Kramer
Eliane Benes
George Constanza
Newman mailman
Frank Constanza
Estelle Constanza
Morty Seinfeld
Aton Benes
J Peterman
George Steinbrenner
```

If we want to add an empty line after a line you can use the next command:

```
[terron@MyFirstLinuxVM seinfeld]$ sed G seinfeld-characters
Jerry Seinfeld

Cosmo Kramer

Eliane Benes
```

For example if we want to replace a word using other word except certain lines we can use the next command:

```
[terron@MyFirstLinuxVM seinfeld]$ sed '8!s/Seinfeld/STERRON/g' seinfeld-characters
Jerry STERRON
Cosmo Kramer
Eliane Benes
George Constanza
Newman mailman
Frank Constanza
Estelle Constanza
Morty Seinfeld
Aton Benes
J Peterman
George Steinbrenner
Uncel Leo
David Puddy
Justin Pitt
Lenny Bania
Jerry STERRON
```

Now in **vi** if you want to replace every character that you have you can use the next command:

```
:%s/Seinfeld/TERRON/█
```

# USER ACCOUNT MANAGEMENT (useradd, groupadd, usermode, userdel, groupdel)

jueves, 9 de diciembre de 2021 03:24 a. m.

This commands are useful to manage accounts in LINUX:

The commands is:

- useradd to create a new user
- groupadd to create a new group
- userdel to delete a user account
- groupdel to delete a group
- usermod to modify a user

This type of records are contained in 3 different files:

- /etc/passwd
- /etc/group
- /etc/shadow

## Example:

```
useradd -g superheros -s /bin/bash -c "user description" -m -d /home/spiderman spiderman
```

For example if we want to create a user called spiderman we have to create first the group superheroes

```
[root@MyFirstLinuxVM ~]# useradd spiderman
[root@MyFirstLinuxVM ~]# id spiderman
uid=1001(spiderman) gid=1001(spiderman) groups=1001(spiderman)
[root@MyFirstLinuxVM ~]# groupadd superheros
```

To check the group that has been created we can check /etc/group file using cat command:

```
[root@MyFirstLinuxVM ~]# cat /etc/group
```

To delete a user is recommended to del the directory of the user, for this we have to go to home directory and then use the next command:

```
[root@MyFirstLinuxVM ~]# cd /home/
[root@MyFirstLinuxVM home]# userdel -r spiderman
[root@MyFirstLinuxVM home]#
```

To delete a group we can use the next command:

```
[root@MyFirstLinuxVM ~]# cd /home/
[root@MyFirstLinuxVM home]# userdel -r spiderman
[root@MyFirstLinuxVM home]# groupdel superheros
[root@MyFirstLinuxVM home]#
```

If we want to modify a user account we can use the next command:

```
[root@MyFirstLinuxVM home]# userdel -r spiderman
[root@MyFirstLinuxVM home]# useradd spiderman
[root@MyFirstLinuxVM home]# groupadd superheros
[root@MyFirstLinuxVM home]# usermod -G superheros spiderman
[root@MyFirstLinuxVM home]# grep spiderman /etc/group
spiderman:x:1001:
superheros:x:1002:spiderman
[root@MyFirstLinuxVM home]# █
```

In this case we are modifying the user to change the group, but not the group ownership of the directories, to change the permissions of the directories we have to use the next command, we have to use -R option to cascade the permission to the group for all subdirectories and files:

```
[root@MyFirstLinuxVM home]# ls -ltr
total 4
drwx----- 19 terron terron 4096 Dec 1 17:50 terron
drwx----- 3 spiderman spiderman 78 Dec 1 18:07 spiderman
[root@MyFirstLinuxVM home]# chgrp -R superheros spiderman
[root@MyFirstLinuxVM home]# ls -ltr
total 4
drwx----- 19 terron terron 4096 Dec 1 17:50 terron
drwx----- 3 spiderman superheros 78 Dec 1 18:07 spiderman
[root@MyFirstLinuxVM home]# █
```

The file that stores all this information we have to use the next command:

```
[root@MyFirstLinuxVM home]# cat /etc/passwd
root:x:0:0::/root:/bin/bash
terron:x:1000:1000:TERRON VICTOR:/home/terron:/bin/bash
spiderman:x:1001:1001::/home/spiderman:/bin/bash
[root@MyFirstLinuxVM home]# █
```

And at the bottom of the file appends all this information:  
terron:x:1000:1000:TERRON VICTOR:/home/terron:/bin/bash  
spiderman:x:1001:1001::/home/spiderman:/bin/bash  
[root@MyFirstLinuxVM home]# █

The first column gives you the name, the second column gives you the encrypted password, third userid, four column gives you the group id, five column description, six column gives you the route of the directory that stores the user data, seven column is the location of the shell for that user.

/etc/group gives you the next information:

First column, gives you group name, second column gives you group password that is the same password that is used by anyone in that group, the third column is the group id, and if it is showed another column means the other user that are also part of this group.

The last file is /etc/shadow

This file stores information about strictly for passwords.

If you notice there is no encrypted password for spider or spiderman in "/etc/shadow" file because we have not set the password for them yet

If we want to get info about certain account we can use the grep command:

```
[root@MyFirstLinuxVM home]# grep spiderman /etc/passwd
spiderman:x:1001:1001::/home/spiderman:/bin/bash
[root@MyFirstLinuxVM home]#
```

In a corporate environment is common that we need to use the next command:

```
[root@MyFirstLinuxVM home]# useradd -g superheros -s /bin/bash -c "TERRONMAN CHARACTER" -m -d /home/TERRONMAN TERRONMAN
[root@MyFirstLinuxVM home]# cat /etc/passwd
```

First line is to create the user -s option is to specify shell, '-d' is to specify the home directory of the user.

Second command is to verify that the user has been created

To change the password or set the password for the new user that has been created we can use:

```
[root@MyFirstLinuxVM home]# passwd TERRONMAN
Changing password for user TERRONMAN.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
[root@MyFirstLinuxVM home]#
```

# ENABLE PASSWORD AGING

jueves, 9 de diciembre de 2021 04:09 a. m.

We can specify the minimum and the maximum days of a password using the next command:

- chage[-m mindays][-M maxdays][-d lastday][-I inactive][-E expiredate][-W warndays] user

Now if we want to change the configuration for every user we can change the file located in /etc/login.defs

This file includes the next info:

- PASS\_MAX\_DAYS 99999
- PASS\_MIN\_DAYS 0
- PASS\_MIN\_LEN 5
- PASS\_WARN\_AGE 7

The command chage is used to set password minimum and maximum age, is used to set parameters around the password it is not used to make any changes to user account, to make any change to the user account is used usermod:

- chage [-d lastday][-m mindays][-M maxdays][-W warndays] [-I inactive][-E expiredate] user
  - -d=3 last password change (lastChanged) Days since Jan 1,1970 that password was last changed
  - -m=4 minimum The minimum number of days required between password changes i.e. the number of days left before the user is allowed to change his/her password
  - -M=5 Maximum, The maximum number of days the password is valid (after that user is forced to change his/her password)
  - -W=6 Warn: The number of days before password is to expire that user is warned that his/her password must be changed
  - -I=7. Inactive: The number of days after password expires that account is disabled
  - -E=8 Expire: days since Jan 1, 1970 that account is disabled i.e. an absolute date specifying when the login may no longer be used

To set the security we can do it as the next picture:

```
[root@MyFirstLinuxVM ~]# chage -m 3 -M 90 -W 10 -I 3 -E 2 babubutt
```

The last command means, that the user cannot make any change in password before 3 days that account has been created, then the maximum life of the password is 90 days, the system will send a warning to the user 10 days before maximum age, the account will be inactive 3 days after password has been changed, the account will expire after 2 days since the account has been inactive.

```
[root@MyFirstLinuxVM ~]# chage -m 3 -M 90 -W 10 -I 3 -E 2 babubutt
[root@MyFirstLinuxVM ~]# grep babu /etc/shadow
babubutt:$6$faUmpQEg$7oX6iNNCgqfRZ0eijzah2z4KRw64IG7WeqcX5VIihoiXNXrjhwU02y9zq99lMSuC0mb4ohlyexwPfjExKUb0.:18963:3:90:10:3:2:
[root@MyFirstLinuxVM ~]#
```

The full command is:

```
[root@MyFirstLinuxVM ~]# useradd babubutt
[root@MyFirstLinuxVM ~]# id babubutt
uid=1003(babubutt) gid=1003(babubutt) groups=1003(babubutt)
[root@MyFirstLinuxVM ~]# grep babu /etc/shadow
babubutt:$6$faUmpQEg$7oX6iNNCgqfRZ0eijzah2z4KRw64IG7WeqcX5VIihoiXNXrjhwU02y9zq99lMSuC0mb4ohlyexwPfjExKUb0.:18963:3:90:10:3:2:
[root@MyFirstLinuxVM ~]# passwd babubutt
Changing password for user babubutt.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
[root@MyFirstLinuxVM ~]# chage -m 3 -M 90 -W 10 -I 3 -E 2 babubutt
[root@MyFirstLinuxVM ~]# grep babu /etc/shadow
babubutt:$6$faUmpQEg$7oX6iNNCgqfRZ0eijzah2z4KRw64IG7WeqcX5VIihoiXNXrjhwU02y9zq99lMSuC0mb4ohlyexwPfjExKUb0.:18963:3:90:10:3:2:
[root@MyFirstLinuxVM ~]#
```

# SWITCH USERS AND SUDO ACCESS (su, sudo)

viernes, 10 de diciembre de 2021 03:07 a. m.

sudo access is a command that allow a user to run root level commands:

- su - username, to change of user
- sudo command, to run certains commands if you are not root
- visudo edits the /etc/sudo, that allows user to add or remove the rights of a certain commands
  - File: /etc/sudoers
- etc is the folder where all the configuration files are stored

```
spiderman@MyFirstLinuxVM:~  
[terron@MyFirstLinuxVM ~]$ su - spiderman  
Password:  
Last failed login: Wed Dec 1 19:25:13 CST 2021 on pts/0  
There were 3 failed login attempts since the last successful login.  
[spiderman@MyFirstLinuxVM ~]$ su -  
Password:  
Last login: Wed Dec 1 19:25:33 CST 2021 on pts/0  
[root@MyFirstLinuxVM ~]# su - spiderman  
Last login: Wed Dec 1 19:26:04 CST 2021 on pts/1  
[spiderman@MyFirstLinuxVM ~]$ █
```

As you can see when you are root you can login into any user because root is very powerfull that theres no needed a password to login into another accounts.

- sudo command is used to run only commands that only have root permissions to run that commands

To give the permission to a user to execute certain root level commands we have to modify the file /etc/sudoers

Then we have to look for %wheel or ALLOWS PEOPLE IN GROUP WHEEL TO RUN ALL COMMANDS

This command is usefull when we want to run root levels commands without be root

After that we have to add a user in the next section if we want to control it at user level:

```
Defaults      secure_path = /sbin:/bin:/usr/sbin:/usr/bin  
  
## Next comes the main part: which users can run what software on  
## which machines (the sudoers file can be shared between multiple  
## systems).  
## Syntax:  
##  
##      user      MACHINE=COMMANDS  
##  
## The COMMANDS section may have other options added to it.  
##  
## Allow root to run any commands anywhere  
root      ALL=(ALL)          ALL  
terron   ALL=(ALL)          ALL █
```

If we want to control it at group level we have to use:

```
[root@MyFirstLinuxVM ~]# usermod -aG wheel terron
[root@MyFirstLinuxVM ~]# grep wheel /etc/group
wheel:x:10:terron
[root@MyFirstLinuxVM ~]#
```

Remember that you are saying to system that every user inside a group can have access to root level commands.

# MONITOR USERS (who, last, w, id)

viernes, 10 de diciembre de 2021 04:01 a. m.

This commands are usefull when we are trying to see what are doing the users in our system.

- who, gives you how many people are logged in, shows every user taht is logged in the system and using the shell

```
[terron@MyFirstLinuxVM ~]$ who
```

```
terron :0          2021-11-30 07:34 (:0)
terron pts/0        2021-12-01 04:43 (:0)
```

- last, show you all the details of every users that have been logged in from since the day one, it shows you what are they doing rebooting the system, whatever happened to the system

```
[terron@MyFirstLinuxVM ~]$ last
```

```
terron pts/1      terron.domain.na Wed Dec  1 19:58 - 20:10  (00:12)
terron pts/1      terron.domain.na Wed Dec  1 19:23 - 19:57  (00:33)
terron pts/0      :0          Wed Dec  1 04:43 still logged in
terron pts/1      terron.domain.na Wed Dec  1 03:10 - 03:17  (00:06)
terron pts/0      :0          Tue Nov 30 07:36 - 04:43  (21:06)
terron :0          :0          Tue Nov 30 07:34 still logged in
reboot system boot 3.10.0-1160.49.1 Tue Nov 30 07:29 - 22:20 (1+14:51)
reboot system boot 3.10.0-1160.el7. Tue Nov 30 07:18 - 22:20 (1+15:02)
terron pts/1      192.168.101.49 Mon Nov 29 09:26 - 11:51  (02:24)
terron pts/1      192.168.101.49 Sun Nov 28 18:56 - 03:57  (09:00)
terron pts/0      :0          Sun Nov 28 18:49 - 07:17  (1+12:27)
terron :0          :0          Sun Nov 28 18:47 - down   (1+12:29)
reboot system boot 3.10.0-1160.el7. Sun Nov 28 18:45 - 07:17  (1+12:31)
terron pts/0      :0          Fri Nov 26 02:00 - 02:01  (00:00)
terron pts/0      :0          Fri Nov 26 01:29 - 02:00  (00:31)
terron :0          :0          Fri Nov 26 01:24 - down   (00:36)
reboot system boot 3.10.0-1160.el7. Fri Nov 26 01:23 - 02:01  (00:37)
```

wtmp begins Fri Nov 26 01:23:44 2021

```
[terron@MyFirstLinuxVM ~]$ █
```

```
[terron@MyFirstLinuxVM ~]$ last | awk '{print $1}' | sort|uniq
```

```
reboot
terron
wtmp
```

- w, gives more information with detail about processes

```
[terron@MyFirstLinuxVM ~]$ w
```

```
22:26:12 up 1 day, 14:56,  2 users,  load average: 0.00, 0.01, 0.05
USER   TTY      FROM             LOGIN@    IDLE   JCPU   PCPU WHAT
terron :0      :0              Tue07    ?xdm?  13:17   1.08s /usr/libexec/gnome-session-binary --session gnome-classic
terron pts/0   :0              04:43    4.00s  0.71s 15.66s /usr/libexec/gnome-terminal-server
```

- finger, this cprogramm have to be installed runing the command: yum install finger -y
- id, this command ghives you informatiuon about you own user

```
[root@MyFirstLinuxVM ~]# id
```

```
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

```
[root@MyFirstLinuxVM ~]# exit
```

```
logout
```

```
[terron@MyFirstLinuxVM ~]$ id
```

```
uid=1000(terron) gid=1000(terron) groups=1000(terron) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

```
[terron@MyFirstLinuxVM ~]$
```

# TALKING TO USERS (users, wall, write)

Junes, 13 de diciembre de 2021 03:06 a. m.

The users are the ones that are logged into your system and some of them have their applications running.

These commands are useful to communicate with that system.

- users, command to find out

```
spiderman@MyFirstLinuxVM:~$ login as: spiderman
spiderman@192.168.101.44's password:
Last login: Wed Dec 1 19:26:52 2021
[spiderman@MyFirstLinuxVM ~]$ users
spiderman terron terron
[spiderman@MyFirstLinuxVM ~]$
```

- wall, broadcast your messages to every user that is logged into your system, you have to put wall, and then write the message and finally press Ctrl+D.

The screenshot shows two terminal windows. The left window is titled 'terron@MyFirstLinuxVM' and displays the user 'terron' logging in. The right window is titled 'spiderman@MyFirstLinuxVM' and shows the 'wall' command being used to broadcast a message to all users. Both windows show the message 'PLEASE CLOSE EVERY PROGRAM AND SAVE YOUR WORK, THIS SYSTEM IS COMING DOWN FOR MAINTENANCE' followed by '-- TERRON' and a green square prompt.

```
terron@MyFirstLinuxVM:~$ File Edit View Search Terminal Help
[terron@MyFirstLinuxVM ~]$ whoami
terron
[terron@MyFirstLinuxVM ~]$ user
bash: user: command not found...
[terron@MyFirstLinuxVM ~]$
Broadcast message from spiderman@MyFirstLinuxVM (pts/1) (Wed Dec 1 22:37:20 2021)

PLEASE CLOSE EVERY PROGRAM AND SAVE YOUR WORK, THIS SYSTEM IS COMING DOWN FOR M
AINTENANCE
-- TERRON

[spiderman@MyFirstLinuxVM ~]$ spiderman@192.168.101.44's password:
Last login: Wed Dec 1 19:26:52 2021
[spiderman@MyFirstLinuxVM ~]$ users
spiderman terron terron
[spiderman@MyFirstLinuxVM ~]$ wall
PLEASE CLOSE EVERY PROGRAM AND SAVE YOUR WORK, THIS SYSTEM IS COMING DOWN FOR MA
INTENANCE
-- TERRON
[spiderman@MyFirstLinuxVM ~]$ Broadcast message from spiderman@MyFirstLinuxVM (pts/1) (Wed Dec 1 22:37:20 2021)
:

PLEASE CLOSE EVERY PROGRAM AND SAVE YOUR WORK, THIS SYSTEM IS COMING DOWN FOR M
AINTENANCE
-- TERRON
[spiderman@MyFirstLinuxVM ~]$
```

Message from spiderman@MyFirstLinuxVM on pts/1 at 22:39 ...  
hey you are using a lot of resources please close the application  
EOF

```
[spiderman@MyFirstLinuxVM ~]$ write terron
hey you are using a lot of resources please close the application
[spiderman@MyFirstLinuxVM ~]$
```

# LINUX DIRECTORY SERVICE -ACCOUNT AUTHENTICATION

lunes, 13 de diciembre de 2021 03:19 a. m.

What if you have thousand of users? How can you manage all the users, there are two types of accounts:

- Local accounts, it is basically used for system administration purposes or for to run major applications
- Domain/Directory accounts, not for local accounts

Now what if you have to create a user account of 10000 different servers, then you have to have a domain or a directory account on a servers, is needed to have the server with ta DB of accounts, then the authentication will be through the server, the server will check if account exists, then the server will compare the received information and will reply OK to go aheadn and log in.

Windows havve have a service called Active Directory.

If you want to log into a server, you use SSL command, now if you want to log in into a server using active directory or some kind of a directory service you will use a protocol called LDAP.

Remember that LINUX is just a protocol its not a directory that you get authenticated for LINUX users. LDAP is not used for LINUX LDAP is a protocol that is used for windows, LINUX or any server. They use this protocol to authenticate against a directory.

# DIFFERENCE BETWEEN ACTIVE DIRECTORY, LDAP, IDM, WindBIND, OpenLDAP, etc.

Lunes, 13 de diciembre de 2021 03:33 a. m.

- Active Directory = Microsoft
- IDM = Identity Manager, included in Red Hat
- WindBIND = Used in Linux to communicate with Windows (SAMBA), allows windows users to get authenticated against Windows Active Directory
- OpenLDAP (open source) = LDAP is a protocol never forget that it stands for loghtweight directory access protocol, OpenLDAP is like an alternative to IDM, remember that IDM is software that you have to buy, if you want to not buy it you will need use OpenLDAP
- IBM Directory Server
- JumpCloud
- LDAP=Lightweight Directory Access Protocol

# SYSTEM UTILITY COMMANDS (date, uptime, hostname, uname,which,cal,bc)

martes, 14 de diciembre de 2021 01:12 a. m.

- date, to get the date
- uptime, says you how many user has been logged into system and how long time has been up and load average
- hostname, indicates that you are logged into the right machine and do not run critical commands on a wrong machine
- uname, return the Linux machine name
- which, means where the commands are located, most commands are located in /usr/bin/

```
[terron@MyFirstLinuxVM ~]$ date
Wed Dec 1 23:52:09 CST 2021
[terron@MyFirstLinuxVM ~]$ uptime
23:52:13 up 1 day, 16:22,  3 users,  load average: 0.24, 0.10, 0.07
[terron@MyFirstLinuxVM ~]$ hostname
MyFirstLinuxVM
[terron@MyFirstLinuxVM ~]$ uname
Linux
[terron@MyFirstLinuxVM ~]$ which pwd
/usr/bin/pwd
```

If we want to know more about how many commands are located in Linux machine you can run the next command:

```
[terron@MyFirstLinuxVM bin]$ ls -l | wc -l
1674
[terron@MyFirstLinuxVM bin]$
```

- cal, calendar

```
[terron@MyFirstLinuxVM bin]$ cal
      December 2021
Su Mo Tu We Th Fr Sa
      1  2  3  4
  5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

[terron@MyFirstLinuxVM bin]$ cal 9 1999
      September 1999
Su Mo Tu We Th Fr Sa
      1  2  3  4
  5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30

[terron@MyFirstLinuxVM bin]$
```

```
[terron@MyFirstLinuxVM bin]$ cal 2021
                           2021

          January           February          March
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
      1  2       1  2  3  4  5  6       1  2  3  4  5  6
  3  4  5  6  7  8  9     7  8  9 10 11 12 13     7  8  9 10 11 12 13
10 11 12 13 14 15 16    14 15 16 17 18 19 20    14 15 16 17 18 19 20
17 18 19 20 21 22 23    21 22 23 24 25 26 27    21 22 23 24 25 26 27
24 25 26 27 28 29 30    28
                           31

          April            May            June
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
      1  2  3       1       1  2  3  4  5
  4  5  6  7  8  9 10     2  3  4  5  6  7  8     6  7  8  9 10 11 12
11 12 13 14 15 16 17    9 10 11 12 13 14 15     13 14 15 16 17 18 19
18 19 20 21 22 23 24    16 17 18 19 20 21 22    20 21 22 23 24 25 26
25 26 27 28 29 30      23 24 25 26 27 28 29    27 28 29 30
                           30 31

          July            August          September
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
      1  2  3       1  2  3  4  5  6  7       1  2  3  4
  4  5  6  7  8  9 10     8  9 10 11 12 13 14     5  6  7  8  9 10 11
11 12 13 14 15 16 17    15 16 17 18 19 20 21    12 13 14 15 16 17 18
18 19 20 21 22 23 24    22 23 24 25 26 27 28    19 20 21 22 23 24 25
25 26 27 28 29 30 31    29 30 31
                           29 30 31

          October          November         December
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
      1  2       1  2  3  4  5  6       1  2  3  4
  3  4  5  6  7  8  9     7  8  9 10 11 12 13     5  6  7  8  9 10 11
10 11 12 13 14 15 16    14 15 16 17 18 19 20    12 13 14 15 16 17 18
17 18 19 20 21 22 23    21 22 23 24 25 26 27    19 20 21 22 23 24 25
24 25 26 27 28 29 30    28 29 30
                           31
```

- bc, binary calculators

```
[terron@MyFirstLinuxVM bin]$ bc
bc 1.06.95
Copyright 1991-1994, 1997, 1998, 2000, 2004, 2006 Free Software Foundation, Inc.
This is free software with ABSOLUTELY NO WARRANTY.
For details type `warranty'.
2+2
4
256*321
82176
[ ]
```

To go out of calculator you have to type quit

# PROCESSES, JOBS AND SCHEDULES

martes, 14 de diciembre de 2021 02:22 a. m.

- Applications also called services, is a program that is running into your machine (NTP,NFS,rsyslog,Apache)
- Script, Something that is written in a file and then packaged it to in a way that it will execute. Shell scripts or commands are list of instructions
- Process, when you run application it opens one process or multiple processes associated with that application
- Daemon, Is something that continuously runs in the background or it doesn't stop, runs until interrupted, keeps listening to the incoming or inbound traffic
- Threads, every process can have multiple threads and application that is running in your background
- Job, also called workorder = something that is created by scheduler like a work order to run those applications and services.

In process commands we have:

- systemctl or service, is used in redhat 7 or 8
- ps, allows you to see the processes that are running on your linux system
- top, you'll see all your processes running in your system depending of what you want to see first based on its load, give you CPU info, memory info
- kill, kills the process name or id
- crontab, is used to schedule this application for your system, when a process is scheduled in crontab it becomes a job.
- at. This command is like crontab, the difference is that it schedules as one time basis or as an ad hoc process

# systemctl COMMAND

martes, 14 de diciembre de 2021 02:45 a. m.

Is needed to have the root permissions

Is also called system controll command, this comand is used to start your application, this command is a new tool to control system services.

This command replaces the older one called services

Usage example:

- `systemctl start|stop|status serviceName.service`

If you want that an application starts when you LINUX machine starts at booth then you have to use enable option of systemctl as the next:

- `systemctl enable serviceName.service`

Another example usage is:

- `systemctl restart|reload serviceName.service`
- `systemctl list-units --all`, this command is used to see what are the list of all the service units that are available to you, then you could run list units, and that give you all the units that are enabled

## **`systemctl list-units --all`**

The output has the following columns:

- **UNIT**: The `systemd` unit name
- **LOAD**: Whether the unit's configuration has been parsed by `systemd`. The configuration of loaded units is kept in memory.
- **ACTIVE**: A summary state about whether the unit is active. This is usually a fairly basic way to tell if the unit has started successfully or not.
- **SUB**: This is a lower-level state that indicates more detailed information about the unit. This often varies by unit type, state, and the actual method in which the unit runs.
- **DESCRIPTION**: A short textual description of what the unit is/does.

The output is:

```
[root@MyFirstLinuxVM ~]# systemctl status firewalld.service
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)
     Active: active (running) since Tue 2021-11-30 07:29:58 CST; 1 day 17h ago
       Docs: man:firewalld(1)
   Main PID: 734 (firewalld)
      Tasks: 2
     CGroup: /system.slice/firewalld.service
             └─734 /usr/bin/python2 -Es /usr/sbin/firewalld --nofork --nopid

Nov 30 07:29:56 MyFirstLinuxVM systemd[1]: Starting firewalld - dynamic firewall daemon...
Nov 30 07:29:58 MyFirstLinuxVM systemd[1]: Started firewalld - dynamic firewall daemon.
Nov 30 07:29:59 MyFirstLinuxVM firewalld[734]: WARNING: AllowZoneDrifting is enabled. This is consider
Hint: Some lines were ellipsized, use -l to show in full.
[root@MyFirstLinuxVM ~]# █
```

To add a service under systemctl management we have to create a unit file in  
/etc/systemd/system/serviceName.service

This means that if you have a third party application and you want that application were managed by  
systemctl command you have to create a service file in the route.

You can power off you entire system using systemctl:

- systemctl poweroff
- systemctl halt, same as poweroff
- systemctl reboot, reebot your entire machine

# ps COMMAND

martes, 14 de diciembre de 2021 03:11 a. m.

ps command stands for process status and it displays all the currently running processes in the LINUX system.

Usage examples:

- ps shows the processes of the current shell

PID = the unique process ID

TTY = terminal type that the user logged-in to

TIME = amount of CPU in minutes and seconds that the process has been running

CMD = name of the command

The output of this command shows as the next:

```
[root@MyFirstLinuxVM ~]# ps
  PID TTY      TIME CMD
13167 pts/1    00:00:00 su
13174 pts/1    00:00:00 bash
13317 pts/1    00:00:00 ps
[root@MyFirstLinuxVM ~]# [ ]
```

00:00:00 means that theres no any request to CPU

- ps -e this command shows you all running processes that you have in the linux system
- ps aux, this command shows all running processes in BSD format
- ps -ef, this command shows all running processes in full format listing (most commonly used)
- ps -u username, thios command shows all processes running by user

```
[iafzal@myfirstlinuxvm ~]$ systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2021-10-13 11:42:34 EDT; 2h 20min ago
       Docs: man:firewalld(1)
     Main PID: 750 (firewalld)
        Tasks: 2
      CGroup: /system.slice/firewalld.service
              └─750 /usr/bin/python -Es /usr/sbin/firewalld --nofork --nopid
[iafzal@myfirstlinuxvm ~]$ ps -ef | grep 750
root    750  1  0 11:42 ?        00:00:01 /usr/bin/python -Es /usr/sbin/fire
walld --nofork --nopid
iafzal  4507 3632  0 14:03 pts/0    00:00:00 grep --color=auto 750
[iafzal@myfirstlinuxvm ~]$
[iafzal@myfirstlinuxvm ~]$
[iafzal@myfirstlinuxvm ~]$
[iafzal@myfirstlinuxvm ~]$ ps -ef | grep firewalld
root    750  1  0 11:42 ?        00:00:01 /usr/bin/python -Es /usr/sbin/fire
walld --nofork --nopid
iafzal  4533 3632  0 14:03 pts/0    00:00:00 grep --color=auto firewalld
[iafzal@myfirstlinuxvm ~]$
```

# top COMMAND

martes, 14 de diciembre de 2021 03:21 a. m.

Is used to show the LINUX processes and it provides a real-time view of the running system

This command shows the summary information of the system and the list of processes or threads which are currently managed by LINUX KERNEL

When the top command is executed then it goes into interactive mode and you can exit out by hitting q

Usage: top, you dont have to be a root to execute it except if you want to kill a process

**PID:** Shows task's unique process id

**USER:** Username of owner of task

**PR:** The "PR" field shows the scheduling priority of the process from the perspective of the kernel

**NI:** Represents a Nice Value of task. A Negative nice value implies higher priority, and positive Nice value means lower priority.

**VIRT:** Total virtual memory used by the task

**RES:** Memory consumed by the process in RAM

**SHR:** Represents the amount of shared memory used by a task

**S:** This field shows the process state in the single-letter form

**%CPU:** Represents the CPU usage

**%MEM:** Shows the Memory usage of task

**TIME+:** CPU Time, the same as 'TIME', but reflecting more granularity through hundredths of a second.

The output is:

```
top - 01:17:43 up 1 day, 17:48, 3 users, load average: 0.00, 0.03, 0.05
Tasks: 201 total, 1 running, 198 sleeping, 2 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.3 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 1014744 total, 76516 free, 677304 used, 260924 buff/cache
KiB Swap: 1048572 total, 970492 free, 78080 used. 166436 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
13142	root	20	0	0	0	0	S	0.3	0.0	0:02.62	kworker/0:0
13562	root	20	0	162224	2384	1572	R	0.3	0.2	0:00.13	top
1	root	20	0	128564	4832	2700	S	0.0	0.5	0:16.64	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.06	kthreadd
4	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0H
6	root	20	0	0	0	0	S	0.0	0.0	0:09.44	ksoftirqd/0
7	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
8	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_bh
9	root	20	0	0	0	0	S	0.0	0.0	0:09.65	rcu_sched
10	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	lru-add-drain
11	root	rt	0	0	0	0	S	0.0	0.0	0:06.72	watchdog/0
13	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kdevtmpfs
14	root	0	-20	0	0	0	S	0.0	0.0	0:00.01	netns
15	root	20	0	0	0	0	S	0.0	0.0	0:00.13	khungtaskd
16	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	writeback
17	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kintegrityd
18	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	bioset
19	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	bioset

To see only the processes that are running for a specific user you can use:

```
[root@MyFirstLinuxVM ~]# top -u terryon
```

To see the absolute path of the command you have to execute top command and then press c:

Without hit c key:

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
12485	terryon	20	0	163300	2664	1268	S	0.3	0.3	0:00.89	sshd
1	root	20	0	128564	4832	2700	S	0.0	0.5	0:16.67	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.06	kthreadd
4	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0H
6	root	20	0	0	0	0	S	0.0	0.0	0:09.45	kssoftirqd/0
7	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0

Hitting c key:

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
12485	terron	20	0	163300	2664	1268	S	0.4	0.3	0:00.94	sshd: terron@pts/1
13643	root	20	0	162252	2380	1572	R	0.4	0.2	0:00.10	top
1	root	20	0	128564	4832	2700	S	0.0	0.5	0:16.67	/usr/lib/systemd/systemd --switched-root --system --deserialize 22
2	root	20	0	0	0	0	S	0.0	0.0	0:00.06	[kthreadd]
4	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	[kworker/0:0H]
6	root	20	0	0	0	0	S	0.0	0.0	0:09.45	[ksoftirqd/0]
7	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	[migration/0]

To kill a process you can use top command and then press k:

top - 01:28:05 up 1 day, 17:58, 3 users, load average: 0.00, 0.05, 0.07											
Tasks: 201 total, 1 running, 198 sleeping, 2 stopped, 0 zombie											
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st											
KiB Mem : 1014744 total, 76516 free, 677300 used, 260928 buff/cache											
KiB Swap: 1048572 total, 970492 free, 78080 used. 166440 avail Mem											
PID to signal/kill [default pid = 13661] 19999999999966											
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
13661	root	20	0	162260	2280	1520	R	6.2	0.2	0:00.01	top

To sort all LINUX processes by memory usage you have to use top then hit M and then P:

top - 01:29:51 up 1 day, 18:00, 3 users, load average: 0.00, 0.04, 0.06											
Tasks: 202 total, 1 running, 199 sleeping, 2 stopped, 0 zombie											
%Cpu(s): 0.0 us, 0.3 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st											
KiB Mem : 1014744 total, 76748 free, 677068 used, 260928 buff/cache											
KiB Swap: 1048572 total, 970492 free, 78080 used. 166672 avail Mem											
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6	root	20	0	0	0	0	S	0.3	0.0	0:09.46	ksoftirqd/0
12485	terron	20	0	163300	2664	1268	S	0.3	0.3	0:01.14	sshd
13671	root	20	0	0	0	0	S	0.3	0.0	0:00.02	kworker/0:3
1	root	20	0	128564	4832	2700	S	0.0	0.5	0:16.67	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.06	[kthreadd]
4	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	[kworker/0:0H]

top command refreshes the information every 3 seconds

# kill COMMAND

martes, 14 de diciembre de 2021 03:48 a. m.

Is used to terminate processes manually

It sends a signal which ultimately terminates or kills a particular process or group of processes

Usage:

**kill [OPTION] [PID]**

OPTION = Signal name or signal number/ID

PID = Process ID

- kill -l to get a list of all signal names or signal numbers, the signal names are telling the process what to do, most commonly signals are:
  - kill PID
  - kill -1 restart
  - kill -2 interrupt for the keyboard just like Ctrl C
  - kill -9 forcefully kill the process
  - kill -15 kill a process gracefully
- Other similar commands to kill are:
  - killall
  - pkill, allows you to kill by the process name

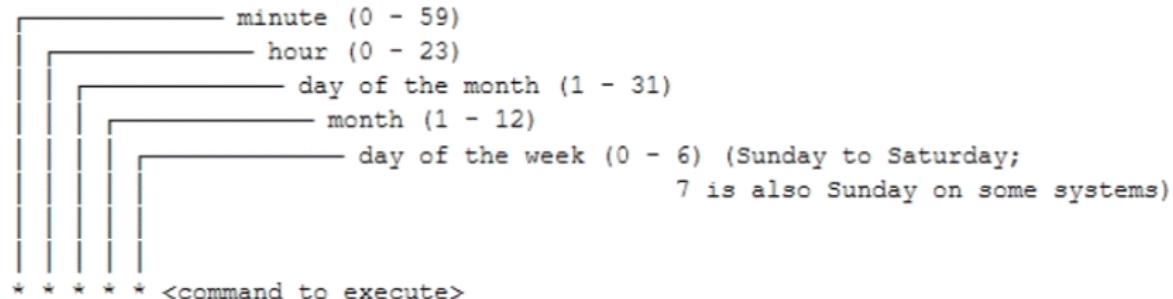
# crontab COMMAND

martes, 14 de diciembre de 2021 03:59 a. m.

crontab command is used to schedule task, the syntax is:

- crontab -e =edit the contrab
- crontab -l =list the crontab entries
- crontab -r =remove the crontab
- crond = crontab daemon /service that manages scheduling
- systemctl status crond= To manage the crond service

When you run crontab -e is needed to specify a certain format as the next picture indicates:



```
[terron@MyFirstLinuxVM ~]$ crontab -e
no crontab for terron - using an empty one
crontab: installing new crontab
[terron@MyFirstLinuxVM ~]$ date
Thu Dec  2 02:48:38 CST 2021
[terron@MyFirstLinuxVM ~]$ ls -ltr
total 556
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Templates
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Public
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Music
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Downloads
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Documents
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Desktop
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Videos
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Pictures
drw-rw-r--. 2 terron terron      6 Nov 29 03:31 ejercicios
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 lois
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 lex
-rw-rw-r--. 1 terron terron      0 Nov 29 11:32 clark
-rw-rw-r--. 2 terron terron     20 Nov 29 11:35 jupiter
-rwxrw-r--. 1 terron terron      0 Nov 29 18:36 sam
-rw-rw-r--. 1 terron terron     18 Nov 29 21:25 geo
-rw-rw-r--. 1 terron terron     51 Nov 29 22:16 fileError
-rw-rw-r--. 1 terron terron    283 Nov 30 06:52 newFIlle2
-rw-rw-r--. 1 terron terron    283 Nov 30 06:52 newFIlle1
-rw-rw-r--. 1 terron terron    301 Dec  1 01:54 newFIlle
drwxrwxr-x. 2 terron terron     44 Dec  1 04:24 simpsons
drwxrwxr-x. 2 terron terron     72 Dec  1 04:33 superman
-rw-r--r--. 1 root   root    531742 Dec  1 04:42 logsFIlleOutput
-rw-rw-r--. 1 terron terron    1853 Dec  1 04:42 mesg-t20
drwxrwxr-x. 2 terron terron    4096 Dec  2 02:45 seinfeld
-rw-r--r--. 1 terron terron    30 Dec  2 02:49 crontab-entry
```

```
[terron@MyFirstLinuxVM ~]$ crontab -l
49 02 * 12 * echo "Tis is my first crontab entry" > crontrab-entry
```

This means that minute will be 49, hour will be 02, every day, 12 is the month, every day of the week.

To remove any entry of the crontab

```
[terron@MyFirstLinuxVM ~]$ crontab -r
[terron@MyFirstLinuxVM ~]$ crontab -l
no crontab for terron
[terron@MyFirstLinuxVM ~]$
```

```
[terron@MyFirstLinuxVM ~]$ systemctl status crond
● crond.service - Command Scheduler
  Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled; vendor preset: enabled)
  Active: active (running) since Tue 2021-11-30 07:30:04 CST; 1 day 19h ago
    Main PID: 1112 (crond)
      Tasks: 1
     CGroup: /system.slice/crond.service
             └─1112 /usr/sbin/crond -n
```

If I stop the service of crontab I'm still be able to create entries to crontab, but as service is stopped system will not execute the task.

# at COMMAND

miércoles, 15 de diciembre de 2021 02:08 a. m.

Is like crontab to schedule jobs but only once

When command is run it will enter interactive mode and you can get out by pressing Ctrl D

## ○ **at HH:MM PM** = Schedule a job

It can be PM or AM.

- atq check all the entries scheduled of at command
- atrm # remove at entry
- atd at daemon/service that manages scheduling
- systemctl status at to manage the atd service

```
[terron@MyFirstLinuxVM ~]$ at 03:05
at> echo "This is mandatory lecture" >at-file
at> <EOT>
job 2 at Thu Dec 2 03:05:00 2021
[terron@MyFirstLinuxVM ~]$ atq
2      Thu Dec 2 03:05:00 2021 a terron
[terron@MyFirstLinuxVM ~]$ ls -ltr at-file
-rw-rw-r--. 1 terron terron 26 Dec 2 03:05 at-file
[terron@MyFirstLinuxVM ~]$ cat at-file
This is mandatory lecture
[terron@MyFirstLinuxVM ~]$ █
```

```
[terron@MyFirstLinuxVM ~]$ atq
2      Thu Dec 2 03:05:00 2021 a terron
[terron@MyFirstLinuxVM ~]$ ls -ltr at-file
-rw-rw-r--. 1 terron terron 26 Dec 2 03:05 at-file
[terron@MyFirstLinuxVM ~]$ cat at-file
This is mandatory lecture
[terron@MyFirstLinuxVM ~]$ atrm
atrm  atrun
[terron@MyFirstLinuxVM ~]$ atrm 2
Cannot find jobid 2
[terron@MyFirstLinuxVM ~]$ atq
[terron@MyFirstLinuxVM ~]$ █
```

Once that task is completed it will be deleted of at entries, to remove any entry you can hit the next command:

- atrm 2

```
[terron@MyFirstLinuxVM ~]$ systemctl status atd
● atd.service - Job spooling tools
   Loaded: loaded (/usr/lib/systemd/system/atd.service; enabled; vendor preset: enabled)
     Active: active (running) since Tue 2021-11-30 07:30:04 CST; 1 day 19h ago
   Main PID: 1115 (atd)
     Tasks: 1
    CGroup: /system.slice/atd.service
        └─1115 /usr/sbin/atd -f
[terron@MyFirstLinuxVM ~]$ █
```

If process is stopped you can add a new entry but no action will be executed because service is stopped.

○ <b>at 2:45 AM 101621</b>	=	Schedule a job to run on Oct 16 <sup>th</sup> , 2021 at 2:45am
○ <b>at 4PM + 4 days</b>	=	Schedule a job at 4pm four days from now
○ <b>at now +5 hours</b>	=	Schedule a job to run five hours from now
○ <b>at 8:00 AM Sun</b>	=	Schedule a job to 8am on coming Sunday
○ <b>at 10:00 AM next month</b>	=	Schedule a job to 10am next month

# ADITIONAL CRON JOBS (hourly, daily, weekly, monthly)

miércoles, 15 de diciembre de 2021 02:18 a. m.

Are 4 different types of cronjobs:

- Hourly
- Daily
- Weekly
- Monthly

All the above crons are setup in:

- /etc/cron.\_\_\_\_\_ (directory)

The timing of each are set in

- /etc/anacrontab -- (except hourly)

For hourly:

- /etc/cron.d/0hourly

```
[terron@MyFirstLinuxVM etc]$ ls -l |grep cron
-rw-----. 1 root root      541 Aug  8  2019 anacrontab
drwxr-xr-x. 2 root root      54 Nov 30 07:20 cron.d
drwxr-xr-x. 2 root root      57 Nov 26 00:50 cron.daily
-rw-----. 1 root root       0 Aug  8  2019 cron.deny
drwxr-xr-x. 2 root root     41 Nov 26 00:50 cron.hourly
drwxr-xr-x. 2 root root      6 Jun  9  2014 cron.monthly
-rw-r--r--. 1 root root    451 Jun  9  2014 crontab
drwxr-xr-x. 2 root root      6 Jun  9  2014 cron.weekly
[terron@MyFirstLinuxVM etc]$ █
```

If you want to run a script daily you can put inside a directory the script file:

```
[terron@MyFirstLinuxVM etc]$ cd cron.daily/
[terron@MyFirstLinuxVM cron.daily]$ ls -l
total 12
-rwx-----. 1 root root 219 Mar 31 2020 logrotate
-rw xr-xr-x. 1 root root 618 Oct 30 2018 man-db.cron
-rwx-----. 1 root root 208 Apr 10 2018 mlocate
[terron@MyFirstLinuxVM cron.daily]$
```

```
[root@MyFirstLinuxVM ~]# cat /etc/cron.d/0hourly
# Run the hourly jobs
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
01 * * * * root run-parts /etc/cron.hourly
[root@MyFirstLinuxVM ~]# cat /etc/cron.monthly
cat: /etc/cron.monthly: Is a directory
[root@MyFirstLinuxVM ~]# cat /etc/anacrontab
# /etc/anacrontab: configuration file for anacron

# See anacron(8) and anacrontab(5) for details.

SHELL=/bin/sh
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
# the maximal random delay added to the base delay of the jobs
RANDOM_DELAY=45
# the jobs will be started during the following hours only
START_HOURS_RANGE=3-22
```

```
RANDOM_DELAY=45
# the jobs will be started during the following hours only
START_HOURS_RANGE=3-22

#period in days    delay in minutes    job-identifier    command
1        5        cron.daily           nice run-parts /etc/cron.daily
7        25       cron.weekly          nice run-parts /etc/cron.weekly
@monthly 45      cron.monthly         nice run-parts /etc/cron.monthly
[root@MyFirstLinuxVM ~]#
```

# PROCESS MANAGEMENT (bg,fg,nice)

miércoles, 15 de diciembre de 2021 03:28 a. m.

For process in the background we have to use the next command:

- BACKGROUND = Ctrl-z, jobs and bg, during execution is press Ctrl-z
- FOREGROUND= fg , command used to become a live process
- Run process even after exit = nohup process &, meaning if you close your terminal and no terminate the process at the moment to close the terminal
  - OR= nohup process > /dev/null 2>&1 &
- KILL a process by name= pkill <nameProecss>
- Process priority = nice (e.g. nice -n 5 process), the niceness scale goes from -20 to 19. The lower the number more priority that task gets
- Process monitoring = top
- List process= ps

```
root@MyFirstLinuxVM:~# sleep 100
^Z
[1]+  Stopped                  sleep 100
root@MyFirstLinuxVM:~# jobs
[1]+  Stopped                  sleep 100
root@MyFirstLinuxVM:~# bg
[1]+ sleep 100 &
root@MyFirstLinuxVM:~# jobs
[1]+  Running                  sleep 100 &
root@MyFirstLinuxVM:~# ps -ef |grep sleep
root    17070 17006  0 04:30 pts/1    00:00:00 sleep 100
root    17078    731  0 04:30 ?        00:00:00 sleep 60
root    17080 17006  0 04:31 pts/1    00:00:00 grep --color=auto sleep
root@MyFirstLinuxVM:~# fg
sleep 100
^Z
[1]+  Stopped                  sleep 100
root@MyFirstLinuxVM:~# 
```

We can execute a command and leave to the background, for that we have to hit Ctrl-Z, then to check status of process we have to use jobs, then to place process at the background we can use bg command, to place the process again to the foreground we have to use fg command.

Every time that you run a process it attaches itself to that terminal, to run a process without interrupt the process when you close the terminal you have to use the next command:

```
[root@MyFirstLinuxVM:~]# nohup sleep 75 &
[2] 17134
[root@MyFirstLinuxVM:~]# nohup: ignoring input and appending output to 'nohup.out'
jobs
[1]+  Stopped                  sleep 100
[2]-  Running                  nohup sleep 75 &
[root@MyFirstLinuxVM:~]# 
```

To avoid to get the message you can type, see the difference:

```
[root@MyFirstLinuxVM ~]# nohup sleep 75 &
[2] 17134
[root@MyFirstLinuxVM ~]# nohup: ignoring input and appending output to 'nohup.out'
jobs
[1]+  Stopped                  sleep 100
[2]-  Running                  nohup sleep 75 &
[root@MyFirstLinuxVM ~]# nohup sleep 73 >dev/null 2>&1 &
[3] 17143
[2]    Done                    nohup sleep 75
[root@MyFirstLinuxVM ~]# -bash: dev/null: No such file or directory
```

Now if we want to say to processor the priority of certain process we can use the next command:

```
[root@MyFirstLinuxVM ~]# nice -n 5 sleep 188
^Z[1]  Done                  sleep 100

[2]+  Stopped                  nice -n 5 sleep 188
[root@MyFirstLinuxVM ~]# bg
[2]+ nice -n 5 sleep 188 &
```

# SYSTEM MONITORING COMMANDS (df,dmesg,iostat 1, netstat,free,top)

miércoles, 15 de diciembre de 2021 03:51 a. m.

- top

```
root@MyFirstLinuxVM:~# top - 04:44:43 up 1 day, 21:15, 3 users, load average: 0.00, 0.01, 0.05
Tasks: 199 total, 1 running, 198 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.3 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 1014744 total, 102224 free, 699236 used, 213284 buff/cache
KiB Swap: 1048572 total, 962300 free, 86272 used. 134300 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
693 libstor+ 20 0 8580 156 124 S 0.3 0.0 0:02.20 lsmd
17211 root 20 0 162224 2380 1572 R 0.3 0.2 0:00.10 top
1 root 20 0 194100 4848 2704 S 0.0 0.5 0:18.07 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.08 kthreadd
4 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 kworker/0:0H
6 root 20 0 0 0 0 S 0.0 0.0 0:11.40 ksoftirqd/0
7 root rt 0 0 0 0 S 0.0 0.0 0:00.00 migration/0
8 root 20 0 0 0 0 S 0.0 0.0 0:00.00 rcu_bh
9 root 20 0 0 0 0 S 0.0 0.0 0:10.90 rcu_sched
10 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 lru-add-drain
11 root rt 0 0 0 0 S 0.0 0.0 0:07.58 watchdog/0
13 root 20 0 0 0 0 S 0.0 0.0 0:00.02 kdevtmpfs
14 root 0 -20 0 0 0 S 0.0 0.0 0:00.02 netns
15 root 20 0 0 0 0 S 0.0 0.0 0:00.14 khungtaskd
16 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 writeback
17 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 histosmited
```

- df, gives you disk partition information, the most common option used with this command is -h, means human readable

```
[root@MyFirstLinuxVM ~]# df
Filesystem      1K-blocks   Used Available Use% Mounted on
devtmpfs          490524     0  490524  0% /dev
tmpfs            507372     0  507372  0% /dev/shm
tmpfs            507372  14188  493184  3% /run
tmpfs            507372     0  507372  0% /sys/fs/cgroup
/dev/mapper/centos-root  8374272 4450028 3924244 54% /
/dev/sdal        1038336 216596  821740 21% /boot
tmpfs           101476     44  101432  1% /run/user/1000
tmpfs           101476     0  101476  0% /run/user/0
[root@MyFirstLinuxVM ~]# 
```

```
[root@MyFirstLinuxVM ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        480M    0  480M  0% /dev
tmpfs          496M    0  496M  0% /dev/shm
tmpfs          496M   14M  482M  3% /run
tmpfs          496M    0  496M  0% /sys/fs/cgroup
/dev/mapper/centos-root  8.0G  4.3G  3.8G 54% /
/dev/sdal       1014M  212M  803M 21% /boot
tmpfs          100M   44K  100M  1% /run/user/1000
tmpfs          100M    0  100M  0% /run/user/0
```

We can use du command that estimates the disk space of the files, this can give you a clue on what files can you delete or get free disk space.

- dmesg, gives you the BIOS information, the system that initialize, what CPU, if there's any issues with memory it will report you, motherboard issue, everything related to the system hardware

```
[root@MyFirstLinuxVm ~]
[ 0.000000] Initializing cgroup subsys cpuset
[ 0.000000] Initializing cgroup subsys cpu
[ 0.000000] Initializing cgroup subsys cpuacct
[ 0.000000] Linux version 3.10.0-1160.49.1.el7.x86_64 (mockbuild@kbuilder.bsys.centos.org) (gcc version 4.8.5 20150623 (Red Hat 4.8.5-44) (GCC) ) #1 SMP Tue Nov 30 15:51:32 UTC 2021
[ 0.000000] Command line: BOOT_IMAGE=/vmlinuz-3.10.0-1160.49.1.el7.x86_64 root=/dev/mapper/centos-root ro crashkernel=auto rd.lvm.lv=centos/root rd.lvm.lv=centos/swap rhgb quiet LANG=en_US.UTF-8
[ 0.000000] e820: BIOS-provided physical RAM map:
[ 0.000000]   BIOS-e820: [mem 0x0000000000000000-0x000000000009ffff] usable
[ 0.000000]   BIOS-e820: [mem 0x000000000000005f-0x00000000005ffff] reserved
[ 0.000000]   BIOS-e820: [mem 0x00000000000000f0-0x0000000000ffff] reserved
[ 0.000000]   BIOS-e820: [mem 0x0000000000010000-0x0000000003ffff] usable
[ 0.000000]   BIOS-e820: [mem 0x0000000003fff0000-0x0000000003ffff] ACPI data
[ 0.000000]   BIOS-e820: [mem 0x000000000fec0000-0x000000000fe00ffff] reserved
[ 0.000000]   BIOS-e820: [mem 0x000000000fee00000-0x000000000fe00ffff] reserved
[ 0.000000]   BIOS-e820: [mem 0x000000000ffcc0000-0x000000000ffffffff] reserved
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] SMBIOS 2.5 present.
[ 0.000000] DMT: innoteck GmbH VirtualBox/VirtualBox, BIOS VirtualBox 12/01/2006
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] e820: update [mem 0x00000000-0x00000ffff] usable ==> reserved
[ 0.000000] e820: remove [mem 0x000a0000-0x000ffff] usable
[ 0.000000] e820: last_pfn = 0x3ffff max_arch_pfn = 0x100000000
[ 0.000000] MTTR default type: uncachable
[ 0.000000] MTTR fixed ranges disabled:
[ 0.000000]   00000-FFFFF uncachable
[ 0.000000] MTTR variable ranges disabled:
[ 0.000000]   0 disabled
[ 0.000000]   1 disabled
[ 0.000000]   2 disabled
[ 0.000000]   3 disabled
[ 0.000000]   4 disabled
[ 0.000000]   5 disabled
[ 0.000000]   6 disabled
[ 0.000000]   7 disabled
[ 0.000000] PAT configuration [0-7]: WB WC UC-UC WB WF UC-UC
[ 0.000000] CPU MTTRs all blank - virtualized system.
[ 0.000000] found SMP MP-table at [mem 0x0009ffff-0x0008ffff] mapped at [ffffffffffff200f0]
[ 0.000000] Base memory trampoline at [ffff97b680090000] 99000 size 24576
[ 0.000000] BRK [0x05a73000, 0x05a73fff] PGTABLE
[ 0.000000] BRK [0x05a74000, 0x05a74fff] PGTABLE
[ 0.000000] BRK [0x05a75000, 0x05a75fff] PGTABLE
[ 0.000000] BRK [0x05a76000, 0x05a76fff] PGTABLE
[ 0.000000] BRK [0x05a77000, 0x05a77fff] PGTABLE
[ 0.000000] RAMDISK: [mem 0x343cb000-0x361ddfff]
[ 0.000000] Early table checksum verification disabled
[ 0.000000] ACPI: RSDP 00000000000e0000 00024 (v02 VBOX )
[ 0.000000] ACPI: XSDP 000000003ff0030 0003C (v01 VBOX )
[ 0.000000] ACPI: FACP 000000003ff000f0 000F4 (v04 VBOX )
[ 0.000000] ACPI: DSDP 000000003ff00470 02325 (v02 VBOX )
[ 0.000000] ACPI: FACS 000000003ff00200 00040
[ 0.000000] ACPI: APIC 000000003ff00240 00054 (v02 VBOX )
[ 0.000000] ACPI: SSDP 000000003ff002a0 001CC (v01 VBOX )
[ 0.000000] ACPI: Local APIC address 0fee0000
[ 0.000000] No NUMA configuration found
[ 0.000000] Faking a node at [mem 0x0000000000000000-0x0000000003ffff]
[ 0.000000] NODE_DATA(0) allocated [mem 0x3ffc9000-0x3ffffeff]
[ 0.000000] crashkernel=auto resulted in zero bytes of reserved memory.
[ 0.000000] kvm-clock: cpu 0, msr 0:3ff78001, primary cpu clock
[ 0.000000] kvm-clock: Using msrs 4b564d01 and 4b564d00
[ 0.000000] kvm-clock: using sched offset of 132242341959989 cycles
[ 0.000000] Zone ranges:
```

- iostat 1, is to get information of input and output statistics, how we are communicating with peripheral devices or system internal devices

```
[root@MyFirstLinuxVM ~]# iostat
Linux 3.10.0-1160.49.1.el7.x86_64 (MyFirstLinuxVM)        12/02/2021      _x86_64_
                                                 (1 CPU)

avg-cpu:  %user   %nice  %system  %iowait   %steal   %idle
          0.06    0.00    0.38     0.00     0.00   99.56

Device:         tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
sda            0.50      18.33         3.03   2965015    489883
dm-0            0.52      18.19         2.55   2941959    412389
dm-1            0.16       0.08         0.56    12772     90620

[root@MyFirstLinuxVM ~]#
```

At the moment that we put 1 it means that it will be refreshed every one second

- netstat, is used to get information about the gateway, subnet mask interface

```
[root@MyFirstLinuxVM home]# netstat -r
```

## Kernel IP routing table

Destination	Gateway	Genmask	Flags	MSS	Window	irtt	Iface
default	gateway	0.0.0.0	UG	0	0	0	enp0s3
192.168.101.0	0.0.0.0	255.255.255.0	U	0	0	0	enp0s3
192.168.122.0	0.0.0.0	255.255.255.0	U	0	0	0	virbr0

- free, gives you the info about the physical memory and the swap space

```
[root@MyFirstLinuxVM home]# free
```

```
total        used        free      shared  buff/cache available
Mem:    1014744     697640     87736      27748     229368   135908
Swap:   1048572      86272    962300
```

- cat /proc/cpuinfo

Remember that proc keeps all the CPU information

```
[root@MyFirstLinuxVM home]# cat /proc/cpuinfo
processor : 0
vendor_id : GenuineIntel
cpu family : 6
model : 60
model name : Intel(R) Core(TM) i5-4300M CPU @ 2.60GHz
stepping : 3
microcode : 0xffffffff
cpu MHz : 2593.998
cache size : 3072 KB
physical id : 0
siblings : 1
core id : 0
cpu cores : 1
apicid : 0
initial apicid : 0
fpu : yes
fpu_exception : yes
cpuid level : 13
wp : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx rdtscp lm constant_tsc rep_good nopl xtTopology nonstop_tsc eagerfpu pni ssse3 cx16 pcid sse4_1 sse4_2 hypervisor lahf_lm invpcid_single fsgsbase invpcid md_clear flush_lid_arch_capabilities
bogomips : 5187.99
clflush size : 64
cache_alignment : 64
address sizes : 39 bits physical, 48 bits virtual
power management:
```

- `cat /proc/meminfo`

`meminfo` gives the memory information

```
[root@MyFirstLinuxVM home]# cat /proc/meminfo
MemTotal:      1014744 kB
MemFree:       87820 kB
MemAvailable:  136012 kB
Buffers:        0 kB
Cached:        178040 kB
SwapCached:    10692 kB
Active:        317748 kB
Inactive:      380404 kB
Active(anon):  240684 kB
Inactive(anon): 307172 kB
Active(file):  77064 kB
Inactive(file): 73232 kB
Unevictable:   0 kB
Mlocked:       0 kB
SwapTotal:     1048572 kB
SwapFree:      962300 kB
Dirty:          0 kB
Writeback:     0 kB
AnonPages:     511624 kB
Mapped:         73552 kB
Shmem:          27744 kB
Slab:           96008 kB
SReclaimable:  51344 kB
SUnreclaim:    44664 kB
KernelStack:   7456 kB
PageTables:    34928 kB
NFS_Unstable:  0 kB
Bounce:         0 kB
WritebackTmp:   0 kB
CommitLimit:   1555944 kB
Committed AS:  4024200 kB
```

# SYSTEM LOGS MONITOR (/var/log)

sábado, 25 de diciembre de 2021 11:19 p. m.

## Log monitoring lesson

Another and most important way of system administration is log monitor

- Log Directory=/var/log
- boot, is the log file when system reboots or boots, record everything
- chronyd=NTP
- cron

```
[root@MyFirstLinuxVM log]# more cron
Nov 26 01:24:04 MyFirstLinuxVM crond[1183]: (CRON) INFO (RANDOM_DELAY will be scaled with factor 93% if used.)
Nov 26 01:24:07 MyFirstLinuxVM crond[1183]: (CRON) INFO (running with inotify support)
Nov 26 01:30:02 MyFirstLinuxVM CROND[3113]: (root) CMD (/usr/lib64/sa/sal 1 1)
Nov 26 01:40:01 MyFirstLinuxVM CROND[3220]: (root) CMD (/usr/lib64/sa/sal 1 1)
Nov 26 01:50:01 MyFirstLinuxVM CROND[3316]: (root) CMD (/usr/lib64/sa/sal 1 1)
```

- maillog, have all the entries when you send or receive an email, every activity is being recorded
- secure, records all the activity of login and logout information in real time

```
[root@MyFirstLinuxVM log]# cat secure | more
Nov 26 01:23:53 MyFirstLinuxVM polkitd[675]: Loading rules from directory /etc/polkit-1/rules.d
Nov 26 01:23:53 MyFirstLinuxVM polkitd[675]: Loading rules from directory /usr/share/polkit-1/rules.d
Nov 26 01:23:53 MyFirstLinuxVM polkitd[675]: Finished loading, compiling and executing 10 rules
Nov 26 01:23:53 MyFirstLinuxVM polkitd[675]: Acquired the name org.freedesktop.PolicyKit1 on the system bus
Nov 26 01:24:03 MyFirstLinuxVM sshd[1168]: Server listening on 0.0.0.0 port 22.
Nov 26 01:24:03 MyFirstLinuxVM sshd[1168]: Server listening on :: port 22.
Nov 26 01:24:31 MyFirstLinuxVM polkitd[675]: Registered Authentication Agent for unix-process:1566:5172 (system bus name :notify-fd 5 --fallback, object path /org/freedesktop/PolicyKit1/AuthenticationAgent, locale en_US.UTF-8)
Nov 26 01:24:32 MyFirstLinuxVM polkitd[675]: Unregistered Authentication Agent for unix-process:1566:5172 (system bus name :notify-fd 5 --fallback, object path /org/freedesktop/PolicyKit1/AuthenticationAgent, locale en_US.UTF-8) (disconnected from bus)
Nov 26 01:24:33 MyFirstLinuxVM gdm-launch-environment: pam_unix(gdm-launch-environment:session): session opened for user root
Nov 26 01:24:38 MyFirstLinuxVM polkitd[675]: Registered Authentication Agent for unix-session:c1 (system bus name :1.14
ct path /org/freedesktop/PolicyKit1/AuthenticationAgent, locale en_US.UTF-8)
Nov 26 01:24:58 MyFirstLinuxVM gdm-password: pam_unix(gdm-password:session): session opened for user terryon by (uid=0)
Nov 26 01:24:59 MyFirstLinuxVM gdm-launch-environment: pam_unix(gdm-launch-environment:session): session closed for user root
Nov 26 01:24:59 MyFirstLinuxVM polkitd[675]: Unregistered Authentication Agent for unix-session:c1 (system bus name :1.14
ct path /org/freedesktop/PolicyKit1/AuthenticationAgent, locale en_US.UTF-8) (disconnected from bus)
Nov 26 01:25:03 MyFirstLinuxVM polkitd[675]: Registered Authentication Agent for unix-session:1 (system bus name :1.176
t path /org/freedesktop/PolicyKit1/AuthenticationAgent, locale en_US.UTF-8)
Nov 26 02:00:11 MyFirstLinuxVM gdm-password: gkr-pam: unlocked login keyring
Nov 26 02:00:40 MyFirstLinuxVM sudo: terryon : user NOT in sudoers ; TTY=pts/0 ; PWD=/home/terryon ; USER=root ; COMMAND=/bin/su
```

```
[root@MyFirstLinuxVM log]# cat secure | more | tail
Dec 2 04:28:18 MyFirstLinuxVM su: pam_unix(su-l:session): session closed for user root
Dec 2 04:28:46 MyFirstLinuxVM su: pam_unix(su-l:session): session opened for user root by terryon(uid=1000)
Dec 2 04:29:14 MyFirstLinuxVM sshd[16996]: Accepted password for root from 192.168.101.43 port 55595 ssh2
Dec 2 04:29:14 MyFirstLinuxVM sshd[16996]: pam_unix(sshd:session): session opened for user root by (uid=0)
Dec 2 05:21:31 MyFirstLinuxVM gdm-password: gkr-pam: unlocked login keyring
Dec 2 05:21:35 MyFirstLinuxVM su: pam_unix(su-l:session): session closed for user root
Dec 2 05:22:28 MyFirstLinuxVM sudo: terryon : user NOT in sudoers ; TTY=pts/0 ; PWD=/home/terryon ; USER=root ; COMMAND=/bin/su
Dec 2 05:22:35 MyFirstLinuxVM su: pam_unix(su:session): session opened for user root by terryon(uid=1000)
Dec 2 05:35:58 MyFirstLinuxVM su: pam_unix(su:session): session closed for user root
Dec 2 05:36:04 MyFirstLinuxVM su: pam_unix(su-l:session): session opened for user root by terryon(uid=1000)
[root@MyFirstLinuxVM log]#
```

In this case we can use the -f parameter to keep all the

- messages, one of the most important commands, every time that is an issue with your machine and allows you to trace the logs through messages log file, this file contains all the hardware information, all the software information, all the application information, all the process information, everything is being logged into this log, so if I wanted to see how many lines this log file have I can use the next command

File Edit View Search Terminal Help

```
[root@MyFirstLinuxVM log]# cat messages | wc -l
8037
[root@MyFirstLinuxVM log]# grep -i error |messages
```

- httpd

# SYSTEM MAINTENANCE COMMANDS (shutdown, reboot,init, halt)

domingo, 26 de diciembre de 2021 12:59 a. m.

Are those that are used by system administrator to reboot system, etc.

- shutdown
- init run levels are from 0-6, 3 is to bring it at multiuser mode
- reboot, only for super user
- halt, shut down the computer and if there are processes that are running

To reboot the computer we can use the command init 6

# CHANGING SYSTEM HOSTNAME

domingo, 26 de diciembre de 2021 01:06 a. m.

Hostname is the name of the system linux machine, this command is used to repurposed or used for something else, to change the hostname we use the next command:

- hostnamectl -set-hostname newhostname

For version of CentOS we have the next files:

- Version 7= Edit /etc/hostname
- Version 6= Edit /etc/sysconfig/network

```
[root@MyFirstLinuxVM ~]# hostname  
MyFirstLinuxVM  
[root@MyFirstLinuxVM ~]# hostnamectl set-hostname terronHost  
[root@MyFirstLinuxVM ~]# hostname  
terronhost  
[root@MyFirstLinuxVM ~]#
```



root@MyFirstLinuxVM:~

# FINDING SYSTEM INFORMATION (uname,dmidecode)

domingo, 26 de diciembre de 2021 01:25 a. m.

Every time that you logged into Linux machine is important to know all the information, all the hardware information, what operating system is running.

- cat /etc/redhat-release
- uname -a

```
[terron@myfirstlinuxvm ~]$ cat /etc/redhat-release
CentOS Linux release 7.9.2009 (Core)
[terron@myfirstlinuxvm ~]$ uname -a
Linux myfirstlinuxvm 3.10.0-1160.49.1.el7.x86_64 #1 SMP Tue Nov 30 15:51:32 UTC
2021 x86_64 x86_64 x86_64 GNU/Linux
[terron@myfirstlinuxvm ~]$ su -
Password:
```

```
Last login: Thu Dec 2 07:35:26 CST 2021 on pts/0
```

- Dmidecode, gives you the BIOS informations

# FINDING SYSTEM ARCHITECTURE (arch)

domingo, 26 de diciembre de 2021 01:40 a. m.

There are two types of architecture, 32 bit and 64 bit

- The main difference is the number of calculations per second they can perform

```
[terron@myfirstlinuxos ~]$ arch  
x86_64  
[terron@myfirstlinuxos ~]$ █
```

---

```
[terron@myfirstlinuxos ~]$ arch  
x86_64  
[terron@myfirstlinuxos ~]$ uname -a  
Linux myfirstlinuxos 3.10.0-1160.49.1.el7.x86_64 #1 SMP Tue Nov 30 15:51:32 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux  
[terron@myfirstlinuxos ~]$
```

# TERMINAL CONTROL KEYS

domingo, 26 de diciembre de 2021 01:43 a. m.

Several key combinations on your keyboard have special effect on terminal

- These control keys are accomplished by holding the CTRL key while typing the second key, for example CTRL-C means to hold the CTRL key while you type the letter C
- The most common control keys are used:
  - CTRL-U ----- erase everything you've typed on the command line
  - CTRL-C ----- stop/kill a command
  - CTRL-Z ----- suspend a command
  - CTRL-D ----- exit from an interactive program (signals end of data)

# TERMINAL COMMANDS (clear, exit, script)

domingo, 26 de diciembre de 2021 01:50 a. m.

Are used to help you manage the terminal most efficiently

- clear clear your screen
- exit exit out of shell, terminal or a user session
- script stores terminal activities in log file that can be named by a user, when a name is not provided by a user, the default filename, typescript is used it record everything that you type on log file

# RECOVER ROOT PASSWORD

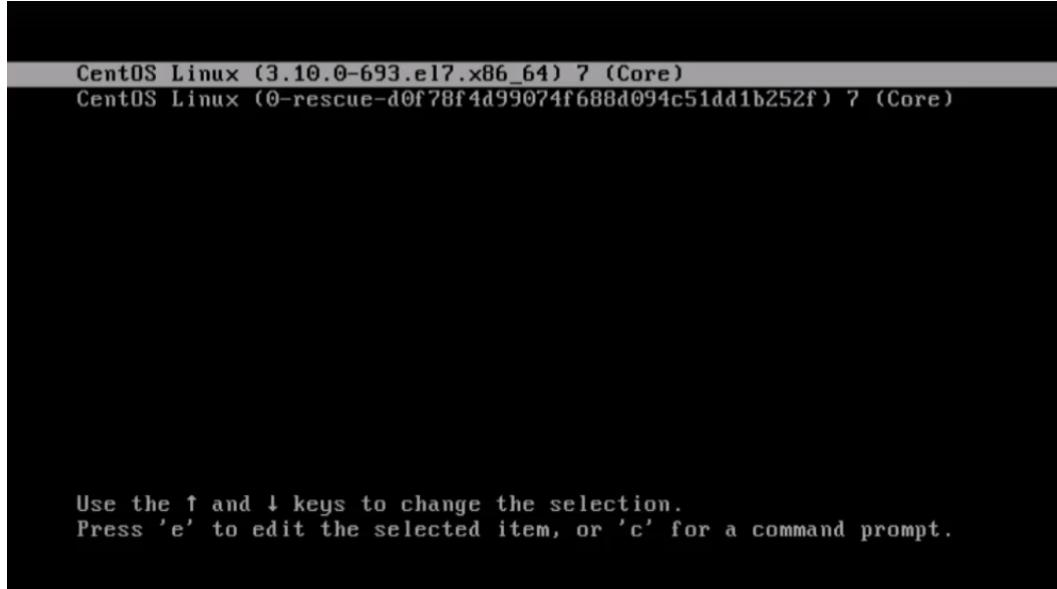
domingo, 26 de diciembre de 2021 01:58 a. m.

Every single administrator in his or life faces this issue where they have to recover root password and why do they face that is because they either have to comply with security standards to change the password.

The steps are:

- Restart computer
- Edit grub file
- Change password
- Reboot

When you turn on the computer it shows two system images, the first is the actual OS:



Select the first option and hit e key, then you have to go down and find the line that says ro:

The image shows a black terminal window with white text. It displays a GRUB configuration script. The script includes several lines of code related to kernel loading, such as "insmod xfs", "set root='hd0,msdos1'", and "search --no-floppy --fs-uuid --set=root". A specific line, "root=/dev/mapper/centos-root", is highlighted with a red box and has a red arrow pointing to the "ro" parameter. Below the script, there is a message: "Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to discard edits and return to the menu. Pressing Tab lists possible completions." A small upward-pointing arrow is located at the top right of the message area.

```
insmod xfs
set root='hd0,msdos1'
if [ x$feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-
efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 --hint='hd0,msdos1' ae84f210-8-
983-4bf2-9bfc-5ba920cef768
else
    search --no-floppy --fs-uuid --set=root ae84f210-8083-4bf2-9bfc-5ba9-
20cef768
fi
linux16 /vmlinuz-3.10.0-693.el7.x86_64 root=/dev/mapper/centos-root ro-
crashkernel=auto rd.lvm.lv=centos/root rd.lvm.lv=centos/swap rhgb quiet LANG=
en_US.UTF-8
initrd16 /initramfs-3.10.0-693.el7.x86_64.img

Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
discard edits and return to the menu. Pressing Tab lists
possible completions.
```

Remove that file and set the next parameter:

```
rw init=/sysroot/bin/sh
ctrl x
chroot /sysroot
passwd root
exit
reboot
```

After to place the firs command line, you have to hit the CTRL-X keys

After the passwd root command you have to update selinux information by running the command:

- touch /.autorelabel

# SOS REPORT

domingo, 26 de diciembre de 2021 02:12 a. m.

Is an international code that is used by many people, SOS command is something that is called when someone is in really deep trouble and needs help.

It is based on RedHat support.

- Collect and package diagnostic and support data
- Package name sos-version
- Command that you will need to run to collect all data and logs is sosreport

# ENVIRONMENT VARIABLES

domingo, 26 de diciembre de 2021 02:18 a. m.

- An environment variable is a dynamic-named value that can affect the way running processes will behave on a computer. They are part of the environment in which a process runs
- In simple words: set of defined rules and values to build an environment

To view all environment variables you have to use the next command:

- printenv OR env

```
;13:*.oga=38;5;13:*.aac=38;5;45:*.au=38;5;45:*.flac=38;5;45:*
=38;5;45:*.wav=38;5;45:*.axa=38;5;45:*.oga=38;5;45:*.spx=38;5
GNOME_TERMINAL_SERVICE=:1.103
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
USERNAME=terron
SESSION_MANAGER=local/unix:@/tmp/.ICE-unix/1792,unix/unix:/tm
GNOME_SHELL_SESSION_MODE=classic
PATH=/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/bin:/
MAIL=/var/spool/mail/terron
DESKTOP_SESSION=gnome-classic
QT_IM_MODULE=ibus
XDG_SESSION_TYPE=x11
PWD=/home/terron
XMODIFIERS=@im=ibus
LANG=en_US.UTF-8
GDM_LANG=en_US.UTF-8
GDMSESSION=gnome-classic
HISTCONTROL=ignoredups
XDG_SEAT=seat0
HOME=/home/terron

```

To view one environment variable you can use the next command:

- echo \$SHELL, specifically shows the shell
- echo \$PATH

```
[terron@myfirstlinuxos ~]$ echo $MAIL
/var/spool/mail/terron
[terron@myfirstlinuxos ~]$ echo $SHELL
/bin/bash
```

To set the environment variables

- export TEST=1
- echo \$TEST

```
[terron@myfirstlinuxos ~]$ export TEST=10
[terron@myfirstlinuxos ~]$ echo $TEST
10
```

To set the environment variable permanently

- vi .bashrc
- TEST='123'
- export TEST

```
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD_PAGER=

# User specific aliases and functions

TESTE='123'
~  
~  
~  
~
```

```
[terron@myfirstlinuxos ~]$ echo $TESTE
123
[terron@myfirstlinuxos ~]$ █
```

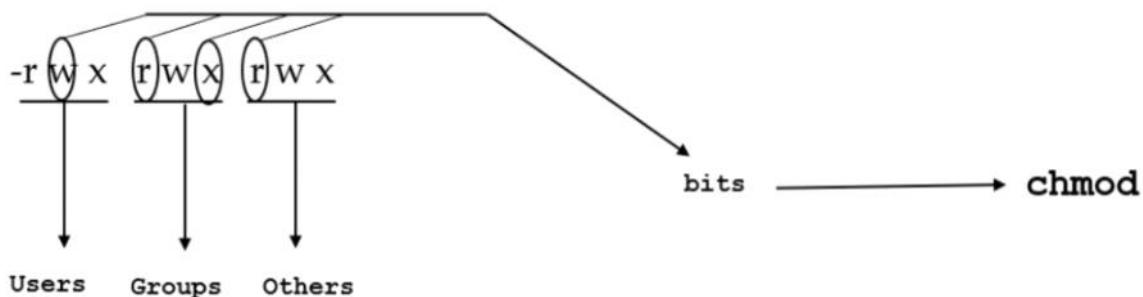
To see a hidden file you can use the next command:

- ls -la

# SPECIAL PERMISSIONS WITH setuid, setgid, and sticky bit

domingo, 26 de diciembre de 2021 02:45 a. m.

All permissions on a file or directory are referred as bits.



There are 3 additional permissions in LINUX:

- setuid: bit tells LINUX to run a program with the effective user id of the owner instead of the executor
- setgid: bit tells LINUX to run a program with the effective group id of the owner instead of the executor
- sticky bit a bit set on files / directories that allows only the owner or root to delete those files

To assign special permissions at the user level we use:

- chmod u+s xyz.sh

To assign special permissions at the group level:

- chmod g+s xyz.sh

To remove special permissions at the user or group level

- chmod u-s xyz.sh
- chmod g-s xyz.sh

To find all executables in LINUX with setuid and setgid permissions

- find / -perm /6000 -type f

```
[iafzal@myfirstlinuxvm ~]$ whoami
root
[iafzal@myfirstlinuxvm ~]$ hostname
myfirstlinuxvm
[iafzal@myfirstlinuxvm ~]$ pwd
/home/iafzal
[iafzal@myfirstlinuxvm ~]$ which passwd
/usr/bin/passwd
[iafzal@myfirstlinuxvm ~]$ ls -l /usr/bin/passwd
-rwsr-xr-x. 1 root root 27832 Jun 10 2014 /usr/bin/passwd
[iafzal@myfirstlinuxvm ~]$ passwd
Changing password for user iafzal.
Changing password for iafzal.
(current) UNIX password: [REDACTED]
```

This is useful to run a command that belongs to another user

## **Please note:**

***These bits work on c  
programming executables not  
on bash shell scripts***

### STICKY BIT

- It is assigned to the last bit of permissions:

**-r w x r w x r w t**

For example in temp directory, this means that every user in the system can write to this directory, but that user even though that user has a right, permissions cannot delete that directory

The sticky is a way to protect a file or directory from delete it, the steps to assign sticky bit is next:

- Become root and create a directory allinone in / = **mkdir /allinone**
  - Assign all rwx permissions to that directory = **chmod 777 /allinone**
  - Become iafzal and create directory inside of /allinone = **mkdir imrandir**
  - Give all rwx permissions to that directory = **chmod 777 imrandir**
  - Create 3 files in that directory = **touch a b c**
  - Open another terminal and login as spiderman
  - Go to /allinone directory and delete imrandir directory = **rm -rf imrandir**
    - You will see the directory is deleted
  - Now become root again and assign sticky bit permission to /allinone = **chmod +t /allinone**
  - Become iafzal and create directory again inside of /allinone = **mkdir imrandir**
  - Give all rwx permissions to that directory = **chmod 777 imrandir**
- 
- Create 3 files in that directory = **touch a b c**
  - Become spiderman user again
  - Go to /allinone directory and try to delete imrandir directory = **rm -rf imrandir**

# SUMMARY

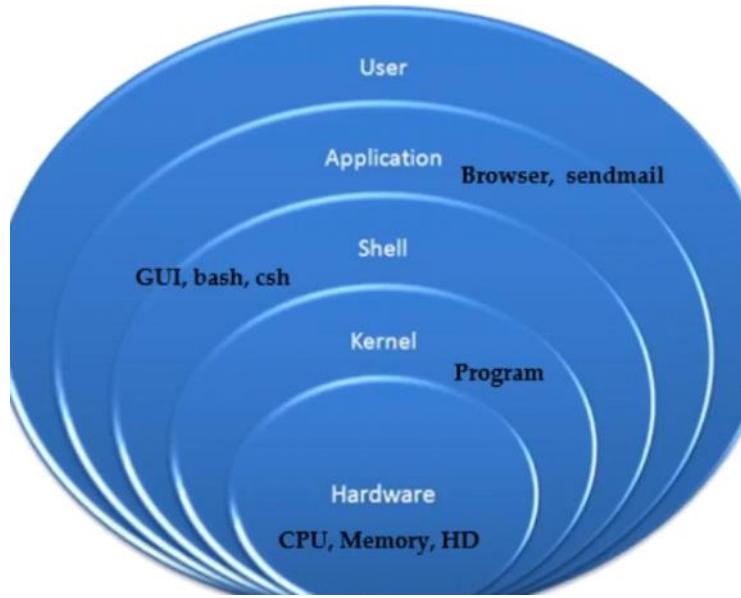
domingo, 26 de diciembre de 2021 11:37 p. m.

- To delete a line in VI editor we use **dd**
- The key that we use in VI to insert and start from new line is **o**
- To create a new user we use **useradd username**
- To modify user attributes we use **usermod command**
- To run commands used by root user only we use **sudo command-name**
- To add a user to sudo configuration file we have to edit **/etc/sudoers** file
- The command **who** and **w** is used to list logged in users
- To find information about an user we can use the command **id user**
- To view and set date we can use **date** command
- An example to view only the month is **cal 1 2008**
- The differences between a daemon and a process is that a process runs and ends after it is done with the task whereas daemon runs all times and wait or listen to protocols
- To start a system service such as ntpd in CentOS/RHEL 7 we have to use the command **systemctl start ntpd**
- The command **ps -ef** list all running process
- We use kill command when the process hung or does not respond to regular commands or sometimes becomes in a zombie process
- The commands that are used for system resource monitoring are:
  - top
  - df
  - free
  - iostat
- The third column in **df -h** command means the disk space usage in actual size
- The difference between shutdown and halt command is that shutdown command stops all the running processes gracefully and then shutdown the system whereas halt is like holding the power button on the physical system
- Is important for a system administrator to monitor logs because log will have all the system messages and system admin can trace the log to find the root cause

# LINUX KERNEL

martes, 28 de diciembre de 2021 02:06 a. m.

Kernel is an interface between HW and SW



# SHELL

martes, 28 de diciembre de 2021 02:12 a. m.

- Is like a container
- Interface between users and kernel/OS
- CLI is a shell

To find your shell you can execute:

- echo \$0
- Available Shells are located in cat /etc/shells
- Ypur shell? Are located in /etc/passwd

# TYPES OF LINUX SHELLS

martes, 28 de diciembre de 2021 02:17 a. m.

- GNOME-LINUX COMMAND GUI SHELL
- KDE-LINUX COMMAND GUI SHELL
- sh - Command line not gui
- bash - born again shell
- csh and tcsh - does not run bash scripts
- ksh - is compatible with shell, improves shell adding floating point an automatic job control, is used in solaris most of time

To list all shells that you have you can use:

- cat /etc/shells

# SHELL SCRIPTING

miércoles, 29 de diciembre de 2021 02:02 a. m.

Shell script is an executable file containing multiple shell commands that are executed sequentially.

The file can contain:

- SHELL (#!/bin/bash)
- COMMENTS (#COMMENTS)
- COMMANDS (echo,p,grep,etc)
- STATEMENTS(IF, WHILE, FOR)

**Shell script should have executable permissions**

**Shell script has to be called from absolute path (/home/userdir(script.bash)**

**If called from current location then ./script.bash**

# BASIC SHELL SCRIPTS

miércoles, 29 de diciembre de 2021 02:06 a. m.

An example to write an script is, remember that you have to add the permission to run the file:

```
[terron@myfirstlinuxos ~]$ mkdir MyScripts
[terron@myfirstlinuxos ~]$ cd MyScripts/
[terron@myfirstlinuxos MyScripts]$ vi outputScreen
[terron@myfirstlinuxos MyScripts]$ ls -ltr outputScreen
-rw-rw-r--. 1 terron terron 45 Dec  3 06:59 outputScreen
[terron@myfirstlinuxos MyScripts]$ chmod a+x outputScreen
[terron@myfirstlinuxos MyScripts]$ ./outputScreen
My First Script on Linux
[terron@myfirstlinuxos MyScripts]$ /home/terron/MyScripts/outputScreen
My First Script on Linux
[terron@myfirstlinuxos MyScripts]$ 
```

```
#!/bin/bash

# DEFINE SMALL COMPROBATIONS BEFORE MAKE ANY CHANGES INTO A LINUX SYSTEM
# DATE WRITTEN 29/12/2021
# WRITTEN BY VICTOR MIGUEL TERRON MACIAS

echo "THE ACTUAL USER IS: "
echo
whoami

echo "THE ACTUAL DIRECTORY ROUTE IS: "
echo
pwd

echo "THE ACTUAL HOSTNAME IS: "
echo
hostname

echo "THE FILES IN THE CURRENT DIRECTORY ARE: "
echo
ls -ltr


```

```
[terron@myfirstlinuxos MyScripts]$ ./outputInitialInfo
THE ACTUAL USER IS:
terron
THE ACTUAL DIRECTORY ROUTE IS:
/home/terron/MyScripts
THE ACTUAL HOSTNAME IS:
myfirstlinuxos
THE FILES IN THE CURRENT DIRECTORY ARE:
total 8
-rwxrwxr-x. 1 terron terron 45 Dec  3 06:59 outputScreen
-rwxrwxr-x. 1 terron terron 354 Dec  3 07:10 outputInitialInfo
[terron@myfirstlinuxos MyScripts]$ 
```

Defining variables:

```
terron@myfirstlinuxos:~/MyScripts
#!/bin/bash

# EXAMPLE OF DEFINING VARIABLES

a='TERRON'
b='MACIAS'
c='VICTOR MIGUEL'

echo "MY NAMES ARE: $c"
echo "MY LAST NAME ARE: $a"
echo "MY SURNAME IS: $b"
~
```

```
[terron@myfirstlinuxos MyScripts]$ ./variableCommands
MY NAMES ARE: VICTOR MIGUEL
MY LAST NAME ARE: TERRON
MY SURNAME IS: MACIAS
[terron@myfirstlinuxos MyScripts]$ 
```

# CREATE SCRIPT TO TAKE INPUT FROM THE USER

miércoles, 29 de diciembre de 2021 02:32 a. m.

- read
- echo

This commands will tell to the bash that have to wait for an input:

[terron@myfirstlinuxos:~/MyScripts]

```
#!/bin/bash
#AUTHOR VICTOR MIGUEL TERRON MACIAS
#DATE 29/12/2021
#INPUT SCRIPTS ON SHELL

echo "My name is VICTOR MIGUEL TERRON MACIAS"
echo
echo 'What is your name?'
read uName
echo
echo 'That is a nice name' $uName
echo
~
```

```
[terron@myfirstlinuxos MyScripts]$ ./inputScript
My name is VICTOR MIGUEL TERRON MACIAS

What is your name?
TERRY

That is a nice name TERRY
```

When you run a command under a variable is needed to use a thick

```
terron@myfirstlinuxos:~/MyScripts
#!/bin/bash
#AUTHOR VICTOR MIGUEL TERRON MACIAS
#DATE 29/12/2021
#INPUT SCRIPTS ON SHELL

#DEFINING VARIABLES
#WHEN YOU RUN A COMMAND IN A VARIABLE IS NEEDED TO USE A THICK
a=`hostname`[REDACTED]
echo "Hello my server name is" $a
echo

echo "My name is VICTOR MIGUEL TERRON MACIAS"
echo
echo 'What is your name?'
read uName
echo
echo 'That is a nice name' $uName
echo
~[REDACTED]
~[REDACTED]
~[REDACTED]
~[REDACTED]
~[REDACTED]
~[REDACTED]
~[REDACTED]
```

```
[terron@myfirstlinuxos MyScripts]$ ./inputScript
Hello my server name is myfirstlinuxos

My name is VICTOR MIGUEL TERRON MACIAS

What is your name?
TERRON

That is a nice name TERRON

[terron@myfirstlinuxos MyScripts]$ [REDACTED]
```

# if-then SCRIPTS

miércoles, 29 de diciembre de 2021 02:51 a. m.

If this happens	= do this
Otherwise	= do that

terron@myfirstlinuxos:~/MyScripts

```
#!/bin/bash

count=100
if [ $count -eq 100 ]
then
    echo 'Count is 100'
else
    echo 'Count is not 100'
fi
~
```

Or:

terron@myfirstlinuxos:~/MyScripts

```
#!/bin/bash

clear
if [ -e /home/terron/error.txt ]
    then
        echo 'FILE EXIST'
    else
        echo 'FILE DOES NOT EXIST'
fi
~
~
~
~
~
~
```

# FOR LOOP SCRIPTS

viernes, 31 de diciembre de 2021 03:31 a. m.

Keep running until specified number of variable

```
[terron@myfirstlinuxos MyScripts]$ vi foorLoop
#!/bin/bash
for i in 1 2 3 4 5
do
    echo "Iteration number $i"
done

~
~
~
```

```
[terron@myfirstlinuxos MyScripts]$ ./foorLoop
Iteration number 1
Iteration number 2
Iteration number 3
Iteration number 4
Iteration number 5
```

Another example is:

```
[terron@myfirstlinuxos:~/MyScripts]
#!/bin/bash
#This script will output terron with different actions

for i in eat run jump play
do
    echo "See TERRON $i"
done

~
```

And the result is:

```
[terron@myfirstlinuxos MyScripts]$ ./simpleForLoopExecute
See TERRON eat
See TERRON run
See TERRON jump
See TERRON play
[terron@myfirstlinuxos MyScripts]$ 
```

# DO WHILE SCRIPTS

viernes, 31 de diciembre de 2021 03:45 a. m.

Are very similar to for loops, the while statement continually executes a block of statements while a particular condition is true or met.

```
#!/bin/bash

# DO WHILE EXAMPLE

count=0
num=10
while[ $count -lt 10 ]
do
    echo
    echo $num 'Seconds left to stop this process $1'
    echo
    sleep 1

num=`expr $num -1`
count= `expr $count + 1`
done
echo
echo $1 process is stopped
echo[]

~
```

The \$1 means the name of the process or process ID entered after running the script

# CASE STATEMENTS SCRIPTS

domingo, 2 de enero de 2022 01:48 a. m.

Where you are giving options on a script, it's like an interactive script.

- If option a is selected = do this
- If option b is selected = do this
- If option c is selected = do this

Most of the installation programs are written in case statements where the script waits for user input to select from choices

```
turon@myfirstlinuxos:~/MyScripts
#!/bin/bash

# AUTHOR VICTOR TURRON

echo
echo "Please choose one of the options below"
echo "a=Display Date and time"
echo "b=List file and directory"
echo "c=List users logged in"
echo "d=Check system uptime"
echo "l=Where am I?"
echo

        read choices
case $choices in
a) date;;
b) ls -ltr;;
c) who;;
d) uptime;;
l) pwd;;
*) echo "Invalid choice BYE"

        esac
~
~
```

# CHECK REMOTE SERVERS CONNECTIVITY

domingo, 2 de enero de 2022 02:24 a. m.

- We're gonna do a script to check the status of remote hosts

```
#!/bin/bash

ping -c1 192.168.1.1
    if [ $? -eq 0 ]
    then
        echo OK
    else
        echo NOT OK
    fi
```

Change the IP to 192.168.1.235

```
Don't show the output
ping -c1 192.168.1.1 &> /dev/null
    if [ $? -eq 0 ]
    then
        echo OK
    else
        echo NOT OK
    fi
```

Define variable

```
#!/bin/bash

hosts="192.168.1.1"
ping -c1 $hosts &> /dev/null
    if [ $? -eq 0 ]
    then
        echo $hosts OK
    else
        echo $hosts NOT OK
    fi
```

Change the IP to 192.168.1.235

The file /dev/null is just a file to throw anything and it will disappear.

```
terron@myfirstlinuxos:~/MyScripts  
#!/bin/bash  
# author: Victor Miguel Terron Macias on 03/01/2021  
# description: This script will ping a remote host and notify  
  
ping -cl 192.168.101.1 &> /dev/null  
    if [ $? -eq 0 ]  
    then  
        echo "PING OK"  
    else  
        echo "PING WENT WRONG"  
    fi  
~
```

```
terron@myfirstlinuxos:~/MyScripts  
#!/bin/bash  
# author: Victor Miguel Terron Macias on 03/01/2021  
# description: This script will ping a remote host and notify  
  
hosts="192.168.101.1"  
ping -cl $hosts &> /dev/null  
    if [ $? -eq 0 ]  
    then  
        echo "PING OK"  
    else  
        echo "PING WENT WRONG"  
    fi  
~  
~  
~  
~  
~  
~  
~
```

To ping multiple servers in one script:

```
[terron@myfirstlinuxos MyScripts]$ cat multiple-pings  
#!/bin/bash  
# AUTHOR: VICTOR MIGUEL TERRON MACIAS  
# DATE: 03/01/2021  
# DESCRIPTION: PING MULTIPLE HOSTS IN A FILE  
hosts="/home/terron/MyScripts/myHost"  
for ip in $(cat $hosts)  
do  
    ping -cl $ip &> /dev/null  
    if [ $? -eq 0 ]  
    then  
        echo "PING TO $ip OK"  
    else  
        echo "PING TO $ip WENT WRONG"  
    fi  
done
```

## ALIASES

Junes, 3 de enero de 2022 02:27 a. m.

Aliases is like a short name or a nickname is like as in python, it allows you to call the commands to avoid larger names.

```
alias ls="ls -al"  
alias pl="pwd; ls"  
alias tell="whoami; hostname; pwd"  
alias dir="ls -l | grep ^d"  
alias lmar="ls -l | grep Mar"  
alias wpa= "chmod a+w"  
alias d="df -h | awk '{print \$6}' | cut -c1-4"
```

```
File Edit View Search Terminal Help  
[terron@myfirstlinuxos ~]$ alias lsltr="ls -ltr"  
[terron@myfirstlinuxos ~]$ lsltr  
total 616  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Templates  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Public  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Music  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Downloads  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Documents  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Desktop  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Videos  
drwxr-xr-x. 2 terron terron 6 Nov 26 01:25 Pictures  
drw-rw-r--. 2 terron terron 6 Nov 29 03:31 ejercicios  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lois  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 lex  
-rw-rw-r--. 1 terron terron 0 Nov 29 11:32 clark  
-rw-rw-r--. 2 terron terron 20 Nov 29 11:35 jupiter  
-rwxrw-r--. 1 terron terron 0 Nov 29 18:36 sam  
-rw-rw-r--. 1 terron terron 18 Nov 29 21:25 geo  
-rw-rw-r--. 1 terron terron 51 Nov 29 22:16 fileError  
-rw-rw-r--. 1 terron terron 283 Nov 30 06:52 newFile2  
-rw-rw-r--. 1 terron terron 283 Nov 30 06:52 newFile1  
-rw-rw-r--. 1 terron terron 301 Dec 1 01:54 newFile  
drwxrwxr-x. 2 terron terron 44 Dec 1 04:24 simpsons  
-rw-r--r--. 1 root root 531742 Dec 1 04:42 logsFileOutput  
-rw-rw-r--. 1 terron terron 1853 Dec 1 04:42 mesg-t20  
drwxrwxr-x. 2 terron terron 4096 Dec 2 02:45 seinfeld  
-rw-r--r--. 1 terron terron 30 Dec 2 02:49 crontrab-entry  
-rw-rw-r--. 1 terron terron 26 Dec 2 03:05 at-file  
----- . . .
```

We can merge two or more commands in an alias as the next picture:

```
[terron@myfirstlinuxos ~]$ alias mcc="pwd; ls"  
[terron@myfirstlinuxos ~]$ mcc  
/home/terron  
at-file Desktop ejercicios lex Music newFile2 seinfeld Templates  
bashrc.original dm-file fileError logsFileOutput MyScripts Pictures simpsons THIS IS THE END OF THE FILE  
clark Documents geo lois newFile Public superman touch  
crontrab-entry Downloads jupiter mesg-t20 newFile1 sam systemdDiskInfo Videos  
[terron@myfirstlinuxos ~]$
```

```
[terron@myfirstlinuxos ~]$ ls -l | grep ^d
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Desktop
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Documents
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Downloads
drw-rw-r--. 2 terron terron      6 Nov 29 03:31 ejercicios
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Music
drwxrwxr-x. 2 terron terron 4096 Dec  4 23:09 MyScripts
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Pictures
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Public
drwxrwxr-x. 2 terron terron 4096 Dec  2 02:45 seinfeld
drwxrwxr-x. 2 terron terron      44 Dec  1 04:24 simpsons
drwxrwxr-x. 2 terron terron      92 Dec  2 10:16 superman
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Templates
drwxr-xr-x. 2 terron terron      6 Nov 26 01:25 Videos
[terron@myfirstlinuxos ~]$
```

```
[terron@myfirstlinuxos ~]$ df -h | awk '{print $6}' | cut -c1-4
Moun
/dev
/dev
/run
/sys
/
/boo
/run
[terron@myfirstlinuxos ~]$
```

```
[terron@myfirstlinuxos ~]$ alias dfawkcut="df -h | awk '{print \$6}' | cut -c1-4"
[terron@myfirstlinuxos ~]$ dfawkcut
Moun
/dev
/dev
/run
/sys
/
/boo
/run
```

To list all aliases that exists you have to use the next command:

```
[terron@myfirstlinuxos ~]$ alias
alias dfawkcut='df -h | awk '\''{print $6}'\'' | cut -c1-4'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l.='ls -d .* --color=auto'
alias ll='ls -l --color=auto'
alias ls='ls --color=auto'
alias lsltr='ls -ltr'
alias mcc='pwd; ls'
alias vi='vim'
alias which='alias | /usr/bin/which --tty-only --read-alias --show-dot --show-tilde'
[terron@myfirstlinuxos ~]$
```

To remove an alias you have to use the next command:

```
[terron@myfirstlinuxos ~]$ unalias dfawkcut
[terron@myfirstlinuxos ~]$ unalias lsltr
```

# USER AND GLOBAL ALIASES

Lunes, 3 de enero de 2022 03:28 a. m.

There are two type of aliases that we can create:

- USER= Applies only to a specific user profile on the terminal that has been created
- GLOBAL= Applies to everyone who has account on the system

USER= /home/user/.bashrc  
GLOBAL=/etc/bashrc

To create an alias for an user you have to edit the file with the next command:

- vi .bashrc

```
terron@myfirstlinuxos:~  
File Edit View Search Terminal Help  
# .bashrc  
  
# Source global definitions  
if [ -f /etc/bashrc ]; then  
    . /etc/bashrc  
fi  
  
# Uncomment the following line if you don't like systemctl's auto-paging feature:  
# export SYSTEMD_PAGER=  
  
# User specific aliases and functions  
  
TESTE='123'  
  
# PERSONAL ALIASES  
alias hh="hostname"  
~  
~  
~  
~
```

To attach an alias you have to become root and edit the file with the next command:

- vi /etc/bashrc

Go to the end of the file and write alias hName="hostname"

```
# GLOBAL ALIASES BY TERRON  
alias hName="hostname"
```

The image shows two terminal windows side-by-side. The left window, titled 'root@myfirstlinuxos:~', displays a command-line session where the user root has run several commands including 'hh', 'su -', 'vi /etc/bashrc', and 'hName'. The right window, titled 'terron@myfirstlinuxos:~', shows a similar session where the user terron has logged in, run 'hh', and 'hName'. Both sessions show the last login details.

```
root@myfirstlinuxos:~$ hh
myfirstlinuxos
[root@myfirstlinuxos ~]# su -
Password:
Last login: Sun Dec  5 00:21:18 CST 2021 on pts/0
[root@myfirstlinuxos ~]# hh
bash: hh: command not found...
[root@myfirstlinuxos ~]# vi /etc/bashrc
[root@myfirstlinuxos ~]# hName
myfirstlinuxos
[root@myfirstlinuxos ~]# 

terron@myfirstlinuxos:~$ hh
terron@192.168.101.44's password:
Last login: Sat Dec  4 23:08:37 2021 from terron.domain.name
[terron@myfirstlinuxos ~]$ hh
myfirstlinuxos
[terron@myfirstlinuxos ~]$ hName
myfirstlinuxos
[terron@myfirstlinuxos ~]$ 
```

# SHELL HISTORY

lunes, 3 de enero de 2022 03:46 a. m.

All commands are recorded

Very effective command for system administrator for troubleshooting executing history command:

```
[root@myfirstlinuxos ~]# history |more
```

```
1 hostname
2 whoiam
3 whoami
4 ls
5 cd /
6 ls -l
7 whoami
8 cd boot/
9 ls -l
10 cd grub2
11 ls -l
12 cd root/
13 ls -ltr
14 ls -all
15 cd root/
16 pwd
17 clean
18 clc
19 clear
20 pwd
21 cd ..
22 cd boot/grub2
23 cd /
24 cd boot/grub2
25 cd boot/
26 cd /boot/
27 cd ..
28 pwd
29 touch terron
30 nano terron
31 vi terron
-- .
```

If you want to run a command that is on history command you can use:

```
[root@myfirstlinuxos ~]# !391
hh
bash: hh: command not found...
[root@myfirstlinuxos ~]# █
```

```
[root@myfirstlinuxos ~]# history | grep awk
399 history | grep awk
[root@myfirstlinuxos ~]#
```

To view other users history of shell commands:

- Become root (su -)
- cat /home/users-dir-name/.bash\_history

# SUMMARY

lunes, 3 de enero de 2022 03:56 a. m.

- KERNEL is an interpreter between hardware and shell
- SHELL is a GUI for users, provides environment for users to execute commands and talk to kernel, Shell is a command line for users
- We do shell scripting to automate tasks

# INTERNET ACCESS TO VM

lunes, 3 de enero de 2022 04:02 a. m.

In order to allow access, we have to make certain changes to our virtual box manager we have to make certain changes:

- Open Virtualbox Manager
- Select the machine you cannot get internet on in the left pane
- Click settings button in the top menu
- Click Network in the left pane in the settings window
- Switched to Bridged Adapter in the Attached to drop-down menu
- Hit OK to save your changes
- Start your VM

# NETWORK COMPONENTS

martes, 4 de enero de 2022 01:02 a. m.

When you are building or installing a new LINUX machine you need to configure IP or other things, mainly you should configure:

- ip address
- Subnet mask is a number 32 bit that masks an IP address, and divides the IP address into network address and host addresses
- Gateway, tells the computer which route you have to pick to send your traffic out and to receive your traffic in
- STATIC VS DHCP
- Interface
- Interface MAC address

# NIC INFORMATION (ethtool)

martes, 4 de enero de 2022 01:54 a. m.

NIC stands for network interface card, that is something that is attaches to your computer, your desktop or your laptop (like ethernet port), NIC can have multiple ports most of the time port is incorrectly referred as NIC in IT

- ethtool enp0s3

To find information about the NIC you have to use ifconfig command, that will give you all the information about the all the interfaces of your computer

If you run ifconfig you will see Other NICs, stands for loop back device is a special interface that the computer uses to communicate within itself.

It works as a diagnostic and troubleshooting and to connect servers running on the local machine, it's like a private communication within itself

Another NICs is virb0, this stands for Virtual Bridge Zero interface that is used for NAT, the virb0, or "Virtual Bridge 0" interface is used for NAT (Network Address Translation). Virtual environments sometimes use it to connect to the outside network, is different for the third interface which is usually labeled as if it's Oracle Virtual, ENPI or if it's regular server you will see eth0.

```
[root@myfirstlinuxos ~]# ethtool enp0s3
Settings for enp0s3:
  Supported ports: [ TP ]
  Supported link modes:  10baseT/Half 10baseT/Full
                         100baseT/Half 100baseT/Full
                         1000baseT/Full
  Supported pause frame use: No
  Supports auto-negotiation: Yes
  Supported FEC modes: Not reported
  Advertised link modes:  10baseT/Half 10baseT/Full
                         100baseT/Half 100baseT/Full
                         1000baseT/Full
  Advertised pause frame use: No
  Advertised auto-negotiation: Yes
  Advertised FEC modes: Not reported
  Speed: 1000Mb/s
  Duplex: Full
  Port: Twisted Pair
  PHYAD: 0
  Transceiver: internal
  Auto-negotiation: on
  MDI-X: off (auto)
  Supports Wake-on: umbg
  Wake-on: d
  Current message level: 0x00000007 (7)
                           drv probe link
  Link detected: yes
```

# NETWORK FILES AND COMMANDS

martes, 4 de enero de 2022 01:31 a. m.

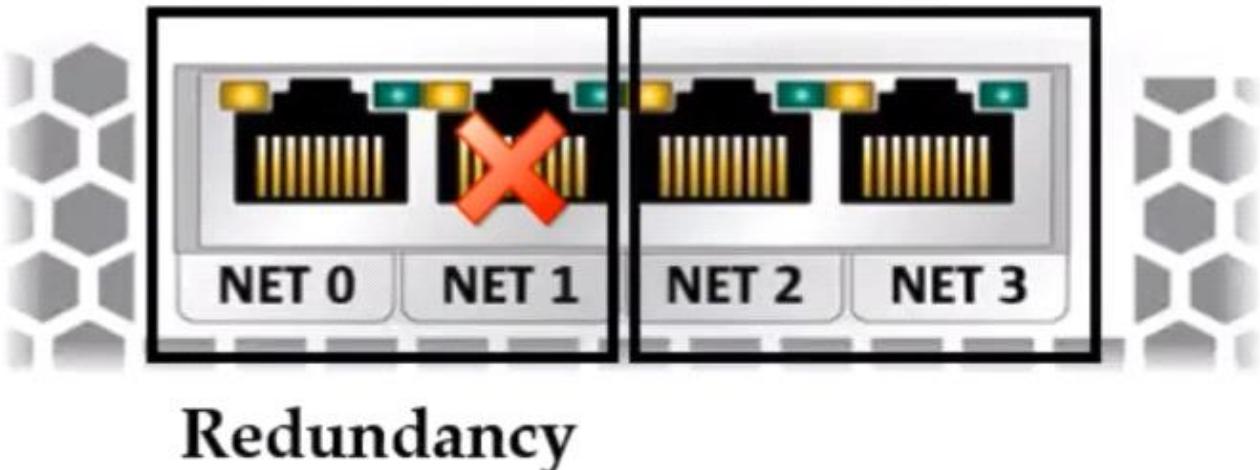
- Interface detection (ethernet port)
- Assigning an IP address (is very important when you want to have a LINUX machine into network)
- Interface configuration files
  - /etc/nsswitch.conf
    - Tells the system where it should resolve its hostname to IP address
  - /etc/hosts
    - Creates like aliases for IP addresses, it's like a DNS service
  - /etc/sysconfig/network
    - Where you specify your hostname
  - /etc/sysconfig/network-scripts/ifcfg-nic
    - Where you specify your IP address on all network, all the subnet mask and gateway, you can change that file using vi and BOOTPROTO= can be setted as static or dhcp, at this way you can set statics IP
  - /etc/resolv.conf
    - Specifies your DNS server
- In Centos version 6 hostname file= /etc/sysconfig/network
- In Centos version 7 hostname file = /etc/hostname
  
- Network commands
  - ping
  - ifconfig, shows details of network interfaces
  - ifup or ifdown, to enable or disable a network interface
  - netstat -rnv, this command gives you information about gateway and how your traffic is flowing and from which interfaces
  - tcpdump -i InterfaceName, traces every single transaction that are leaving your machine and coming into your machine

# NIC OR PORT BONDING

martes, 4 de enero de 2022 02:07 a. m.

NIC BONDING is one of the very important and critical aspect in LINUX network configuration.

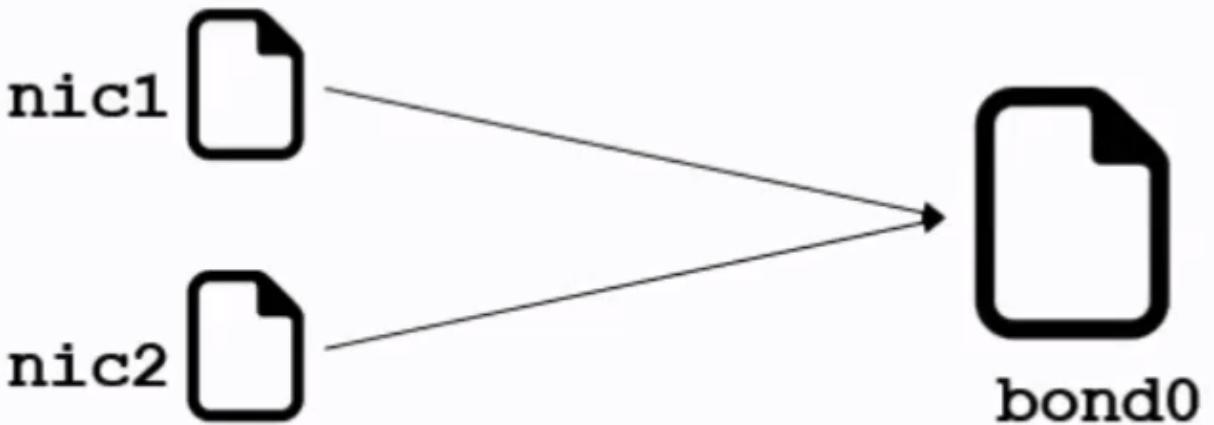
- NIC= Network Interface Card (PC or laptop)
- NIC BONDING is also known as Network bonding. It can be defined as the aggregation or combination of multiple NIC into a single bond interface, its main purpose is to provide high availability and redundancy



We can merge certain ports to be one, for example if NET0 gets unfunctional we can use NET1 port and it will be the same as NET0 port that is redundancy, the high availability is to use NET2 and NET3 to get higher speeds at the same this is achieved as link aggregation.

## NIC BONDING PROCEDURE

- modprobe bonding, to get the information of your driver
- modinfo bonding, to get information of your bonding
- Create /etc/sysconfig/network-scripts/ifcfg-bond0
- Edit /etc/sysconfig/network-scripts/ethernet1
- Edit /etc/sysconfig/network-scripts/ethernet2



And then you use

- Restart network = systemctl restart network

#### STEPS TO SET NETWORK BONDING

- On a VM you have to go to settings/Network and check adapter2 and attached it as Bridged adapter, then start the VM
- Open terminal, and execute ifconfig command to see the network interfaces that you'll have to see what are the network interfaces available to merge, the output will be:

```

enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTI
        inet 192.168.1.5 netmask 255.255.255.0
        inet6 fe80::a00:27ff:fe3:e28 prefixlen 64
          ether 08:00:27:c3:0e:28 txqueuelen 1000
            RX packets 244 bytes 24423 (23.8 KiB)
            RX errors 0 dropped 0 overruns 0
            TX packets 98 bytes 11689 (11.4 KiB)
            TX errors 0 dropped 0 overruns 0 collisions 0

```

```

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTI
        inet 192.168.1.17 netmask 255.255.255.0
        inet6 fe80::38d3:b56c:bf25:7f32 prefixlen 64
          ether 08:00:27:0d:17:9b txqueuelen 1000

```

- Become root
- Follow STEPS OF

REMEMBER THAT AFTER TO CREATE THE FILE YOU HAVE TO SET THE BOOTPROTO=none or static if you want to assign a static IP address

ONBOOT=yes is to enable bond when system reboots

# NEW NETWORK UTILITIES (nmtui,nmcli,nm-connection-editor, GNOME Settings)

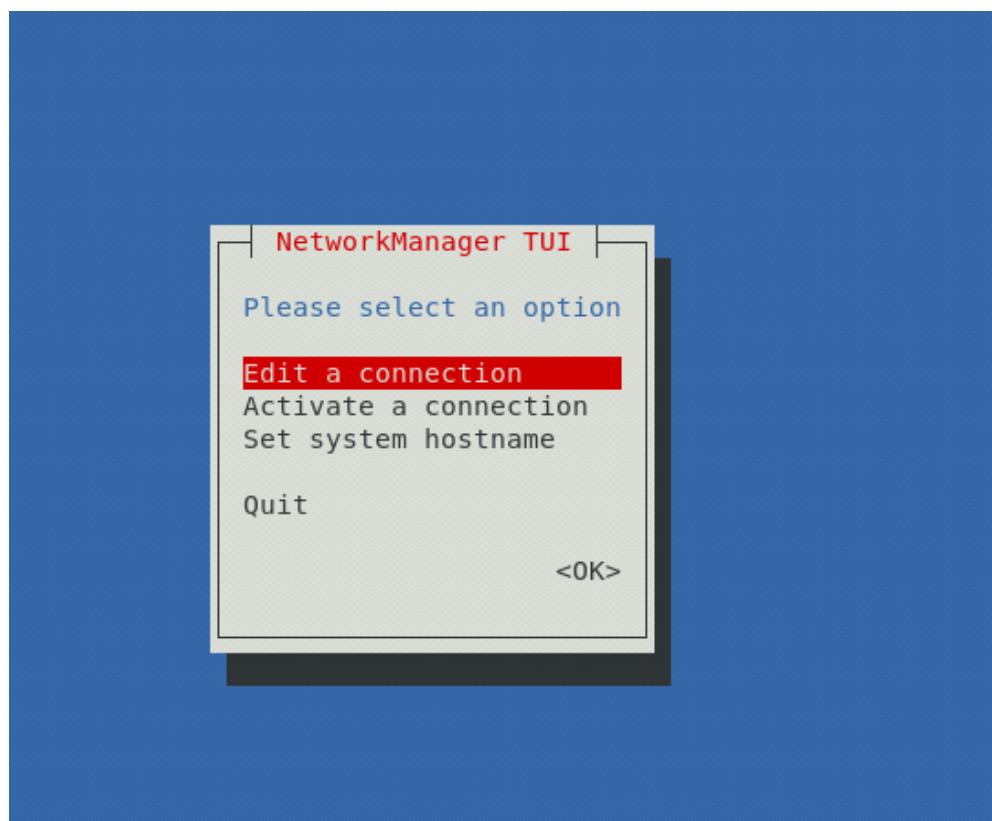
martes, 4 de enero de 2022 03:26 a. m.

What we will learn in this lecture is:

- Getting started with Network Manager
- Network configuration methods
  - nmtui
  - nmcli
  - nm-connection.editor
  - GNOME settings
- NetworkManager is a service that provides set of tools designed specifically to make it easier to manage the networking configuration on LINUX systems and is the default network management service on RHEL 8
- It makes network management easier
- It provides easy setup of connections to the user
- NetworkManager offers management through different tools such as GUI, nmtui, and nmcli

## NETWORK CONFIGURATION METHODS

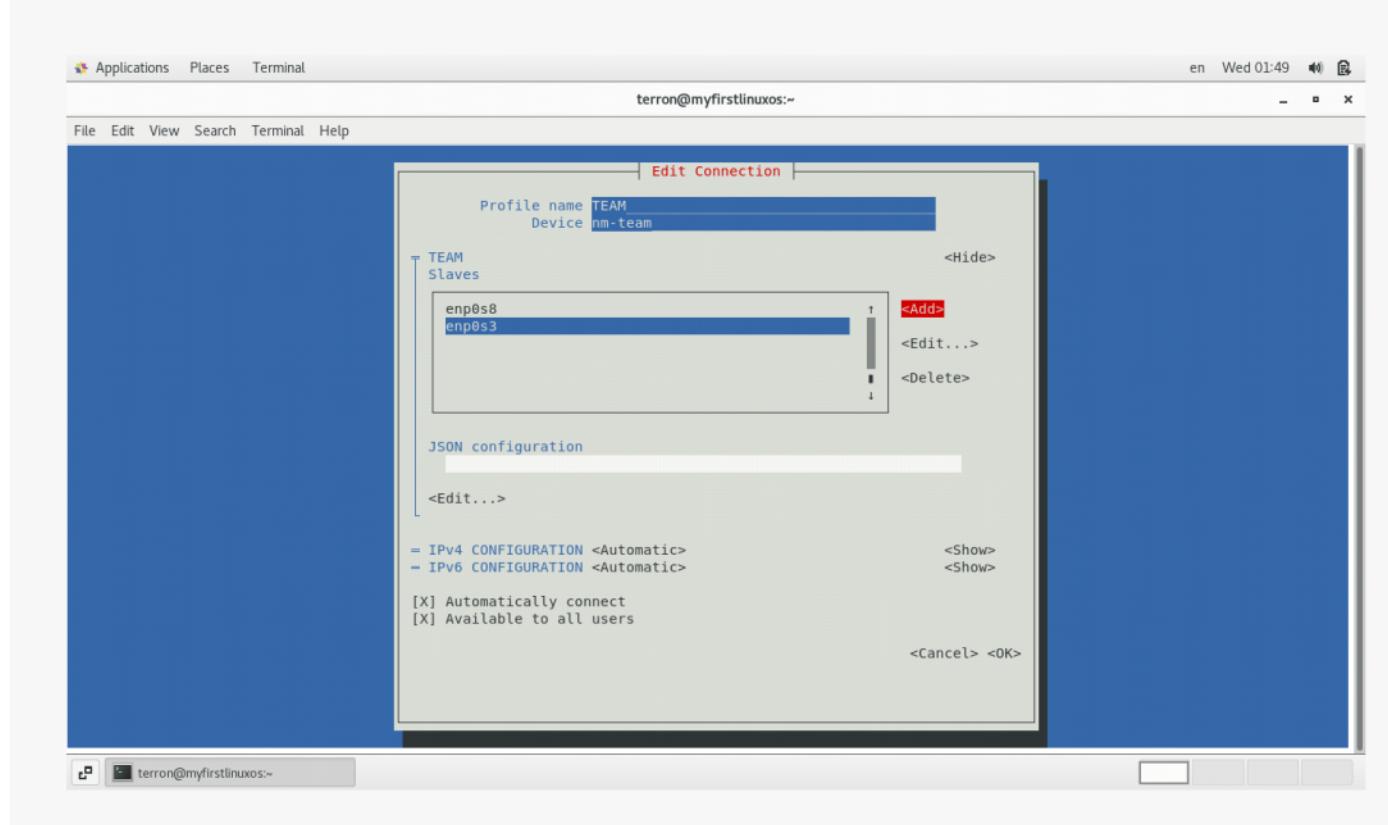
- nmcli - Short for network manager command line interface. This tool is useful when access to a graphical environment is not available and can also be used within scripts to make network configuration changes
- nmtui -Short for network manager text user interface. This tool can be run within any terminal window and allows changes to be made by making menu selections and entering data
  - We have to use the command to enter into this utility, nmtui



- nm-connection-editor - A full graphical management tool providing access to most of the NetworkManager configuration options. It can only be accessed through the desktop or console

- GNOME Settings - The network screen of the GNOME desktop settings application allows basic network management tasks to be performed

There is a new tool that replaces bond, its called team, both perform the same function. Its important to have all the IP of the NIC, remember that all the adapters that will become in a NIC bonding have to be deactivated, you can use nmcli command to deactivate it. After you delete all the adapters it will show wired connection, it has to be deleted too, then click to add connection and click on team, name it and then add the adapters.

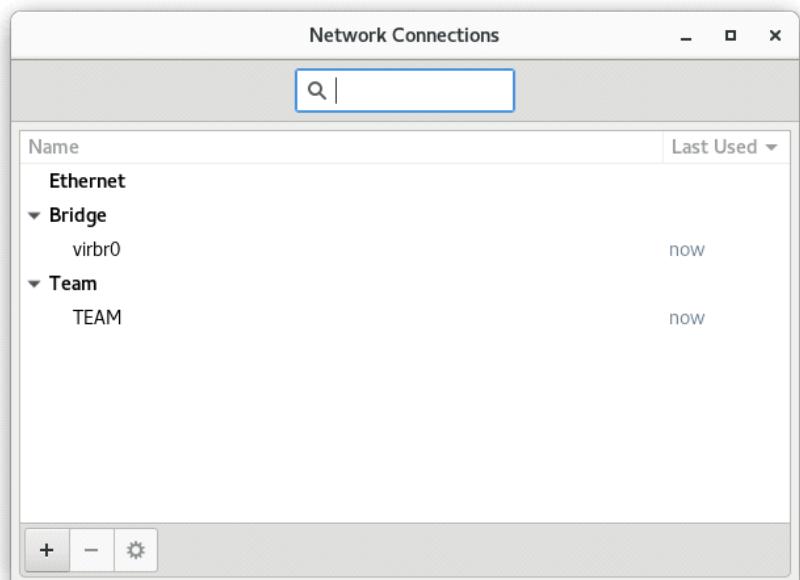


- Nmcli command with parameter con and show as the next picture indicates shows the next information

```
~ [terron@myfirstlinuxos ~]$ nmcli con show
NAME      UUID                                  TYPE      DEVICE
TEAM      44ef62c8-d7e9-43c2-94e9-12d7a53efb8f  team      nm-team
virbr0   59c1086e-cace-4cc7-89e0-f98eaec94be2  bridge    virbr0
enp0s3   13489931-36b3-4f2a-8502-b180fdbba2f1  ethernet  enp0s3
enp0s8   3d8c3aef-68b1-471b-9f99-01d408c5887e  ethernet  enp0s8
[terron@myfirstlinuxos ~]$
```

The command nm-connection-editor shows the next output:

```
[terron@myfirstlinuxos ~]$ nm-connection-editor
```



## DOWNLOADING FILES OR APPS (wget)

miércoles, 5 de enero de 2022 02:03 a. m.

LINUX= wget command stands when you are trying to download a file or an app that you need, w stands for worldwide and get of get it.

- wget <http://website.com/filename>
- Most of the servers in corporate environment do NOT have internet access
- Remember that you have to configure the network adapter in bridged mode

First you have to install yum, to do that you have to become root:

```
[root@myfirstlinuxos ~]# yum install putty
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirror.grid.uchicago.edu
 * extras: repos.hou.layerhost.com
 * updates: mirror.vtti.vt.edu
base
extras
updates
updates/7/x86_64/primary_db
No package putty available.
Error: Nothing to do
[root@myfirstlinuxos ~]# wget https://the.earth.li/~sgtatham/putty/latest/putty-0.76.tar.gz
```

## curl AND ping COMMANDS

miércoles, 5 de enero de 2022 02:23 a. m.

Are useful when we are trying to troubleshoot system, network related issues.

- curl command specifically is for transfer data to or from a server, lets you talk to a server by specifying the location and the data you want to send. This command tells you if page is up
- ping command says you if server is up but does not confirm if the page or the website is up

**curl http://website.com/filename**  
**curl -O http://website.com/filename**

curl command gives you the html source code:

```
[terron@myfirstlinuxos ~]$ curl google.com
<HTML><HEAD><meta http-equiv="content-type" content="text/html; charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.com/">here</A>.
</BODY></HTML>
[terron@myfirstlinuxos ~]$
```

curl -O <http://website.com/filename> allows you to download a file, remember that O is uppercase

```
[terron@myfirstlinuxos ~]$ curl -O http://google.com/index
% Total    % Received % Xferd  Average Speed   Time     Time      Current
          Dload  Upload   Total Spent  Left Speed
100  1566  100  1566    0     0  23971      0 --:--:-- --:--:-- --:--:-- 24468
[terron@myfirstlinuxos ~]$
```

```
@myfirstlinuxos ~]# curl google.com
<HTML><HEAD><meta http-equiv="content-type" content="text/html; charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.com/">here</A>.
</BODY></HTML>
[root@myfirstlinuxos ~]# ping google.com
PING google.com (142.250.65.110) 56(84) bytes of data.
64 bytes from qro0ls24-in-f14.1e100.net (142.250.65.110): icmp_seq=1 ttl=116 time=9.24 ms
64 bytes from qro0ls24-in-f14.1e100.net (142.250.65.110): icmp_seq=2 ttl=116 time=9.29 ms
64 bytes from qro0ls24-in-f14.1e100.net (142.250.65.110): icmp_seq=3 ttl=116 time=9.81 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 9.245/9.451/9.815/0.269 ms
```

To download a file using curl command:

```
[root@myfirstlinuxos ~]# curl -O https://the.earth.li/~sgtatham/putty/latest/putty-0.76.tar.gz
```

# FTP - File transfer protocol

miércoles, 5 de enero de 2022 02:53 a. m.

Is a standard network protocol used for the transfer of computer files between a client and server on a computer network. FTP is built on a client-server model architecture using separate control and data connections between the client and the server.

Protocol is a set of rules used by computers to communicate

Default FTP protocol= 21

Default port for SSA=22

Default port for DNS=53

For this lecture you will need 2 LINUX machines:

- Client=MyFirstLinuxOS
- Server=LinuxCentOS7



We have to make sure that Server is executing `ftpd` service and by default that service is running in port 21.

To do the process we have to make the next processes.

## PROCESS TO SET FTP SERVER

- **Install and Configure FTP on the remote server**

- `# Become root`
  - `# rpm -qa | grep ftp`
  - `# ping www.google.com`
  - `# yum install vsftpd`
  - `# vi /etc/vsftpd/vsftpd.conf` *(make a copy first)*
- 
- Find the following lines and make the changes as shown below.
  - `## Disable anonymous login ##`
    - `anonymous_enable=NO`
  - `## Uncomment ##`
    - `ascii_upload_enable=YES`
    - `ascii_download_enable=YES`
  - `## Uncomment – Enter your Welcome message – This is optional ##`
    - `ftpd_banner=Welcome to UNIXMEN FTP service.`
  - `## Add at the end of this file ##`
    - `use_localtime=YES`
  
  - `# systemctl start vsftpd`
  - `# systemctl enable vsftpd`
  - `# systemctl stop firewalld`
  - `# systemctl disable firewalld`
  - `# useradd iafzal` *(if the user does not exist).*

rpm command is just to make sure that the package is not installed

## PROCESS TO MAKE FTP CLIENT

- Install FTP client on the client server

- # Become root
- # yum install ftp
- # su - iafzal
- \$ touch kruger

- Commands to transfer file to the FTP server:

- ftp 192.168.1.x
- Enter username and password
- bi
- hash
- put kruger
- bye.

```
[terron@myfirstlinuxos ~]$ ftp 192.168.101.49
Connected to 192.168.101.49 (192.168.101.49).
220 Welcome to TERRON  FTP service.
Name (192.168.101.49:terron): terron
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> bi
200 Switching to Binary mode.
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> put fileTransferFTP
local: fileTransferFTP remote: fileTransferFTP
227 Entering Passive Mode (192,168,101,49,28,227).
150 Ok to send data.
226 Transfer complete.
ftp> [REDACTED]
```

## SCP - SECURE TRANSFER PROTOCOL

miércoles, 5 de enero de 2022 03:50 a. m.

SCP (secure transfer protocol) helps to transfer computer files securely from a local to a remote host. It is somewhat similar to the FTP (File transfer protocol) but it adds security and authentication.

Default SCP Port= 22 (same as SSH)

For this lecture we will need two LINUX machines.



The process to transfer a file to the remote server is:

- **SCP commands to transfer file to the remote server:**

- **Login as yourself (iafzal)**
- **touch jack**
- **scp jack iafzal@192.168.1.x:/home/iafzal**
- **Enter username and password**

scp then goes the file that you want to transfer, then the username of the account that exists in the remote server, then the IP address and lastly the route.

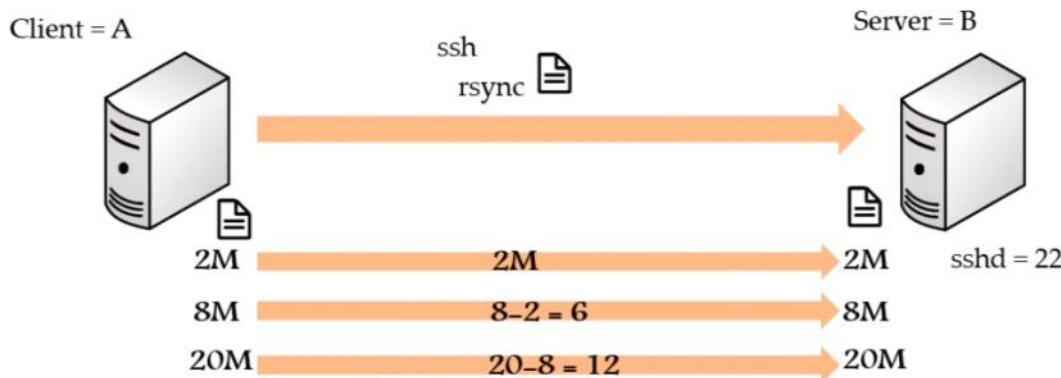
```
root@linuxcentos7:~# echo "SERVER"
[root@linuxcentos7 ~]# ifconfig
[root@linuxcentos7 ~]# ifconfig enp0s3
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.101.49 brd 192.168.101.255
inet6 fe80::8a0:5b93:ib98:4173d/128 brd fe80::ff:fe00:4173d
      inet 192.168.101.49 brd 192.168.101.255
      inet6 fe80::8a0:5b93:ib98:4173d/128 brd fe80::ff:fe00:4173d
      ether 08:00:27:07:58:15 txqueuelen 1000 (Ethernet)
      RX packets 3999 bytes 2193520 (2.0 MiB)
      RX errors 0 dropped 3 overruns 0 frame 0
      TX packets 1673 bytes 211828 (206.8 KiB)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
[root@linuxcentos7 ~]# cd /home/terron/
[root@linuxcentos7 terron]# ls -ltr fileTransfer
-rw-r--r--. 1 root root 8 Jan  5 04:01 fileTransfer
[root@linuxcentos7 terron]# 
```

```
root@myfirstlinuxos:~# echo "CLIENT"
[client@myfirstlinuxos ~]# touch fileTransfer
[client@myfirstlinuxos ~]# vi fileTransfer
[client@myfirstlinuxos ~]# scp fileTransfer root@192.168.101.49:/home/terron/
The authenticity of host '192.168.101.49 (192.168.101.49)' can't be established.
ECDSA key fingerprint is SHA256:M/VKe@mnlmHfiNgzNK+wC82Xm5UE2HG1zgvEY5nxrxlw.
ECDSA key fingerprint is MD5:95:34:38:bb:14:f2:c2:80:2d:41:8a:a4:6f:a3:9f:7b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.101.49' (ECDSA) to the list of known hosts.
root@192.168.101.49's password:
fileTransfer
[client@myfirstlinuxos ~]# 
```

# rsync -Remote Synchronization

miércoles, 5 de enero de 2022 04:02 a. m.

- Is another utility to transfer a file to another machine, rsync is a utility for efficiently transferring and synchronizing files within the same computer or to a remote computer by comparing the modification times and sizes of files. For example if initially the file is of 2 GB and in other times the file are 4 GB it will only transfer the 2 GB of difference.
- rsync is a lot faster than FTP or SCP
- Default rsync port is 22 same as SSH
- This utility is mostly used to backup the files and directories from one server to another
- For this lecture we need 2 linux machines
  - CLIENT=MyFirstLinuxVM
  - SERVER=LinuxCentOS7



- First step: check if rsync is installed using the next command:
  - rpm -qa | grep rsync
- Basic syntax of rsync command
  - **# rsync options source destination**
- Install rsync in your Linux machine (*check if it already exists*)
  - **# yum install rsync** (*On CentOS/Redhat based systems*)
  - **# apt-get install rsync** (*On Ubuntu/Debian based systems*)
- rsync a file on a local machine
  - **\$ tar cvf backup.tar .** (*tar the entire home directory (/home/iafzal)*)
  - **\$ mkdir /tmp/backups**
  - **\$ rsync -zvh backup.tar /tmp/backups/**
- rsync a directory on a local machine
  - **\$ rsync -azvh /home/iafzal /tmp/backups/**
- rsync a file to a remote machine
  - **\$ mkdir /tmp/backups** (*create /tmp/backups dir on remote server*)
  - **\$ rsync -avz backup.tar iafzal@192.168.1.x:/tmp/backups**
- rsync a file from a remote machine
  - **\$ touch serverfile**
  - **\$ rsync -avzh iafzal@192.168.1.x:/home/iafzal/serverfile /tmp/backups**

# SYSTEM UPDATES AND REPOS (rpm, yum)

jueves, 6 de enero de 2022 01:51 a. m.

- yum (CentOS), apt-get (other Linux distributions servers)
- yum command have the config file on:
  - /etc/yum.repos.d

In some companies internet access is not allowed. In that case repos are setup locally on another server within the same network.

- rpm (Red hat package manager)

The main difference between yum and rpm is that rpm is used when you already have a package download it in your system and then you could install it running the command rpm  
yum download the package and install it for you, yum make all the process

- To install a package with rpm you can use the next command:

```
[root@MyFirstLinuxOS ~]# rpm -ihv /tmp/package.rpm
```

- To remove a package with rpm you have to use the next command:
  - rpm -e nameOfPackage
- To remove a package using yum is very easy and smart cause yum automatically detects all the packages and subpackages that has been installed using the next command:
  - yum remove nameOfPackage

# SYSTEM UPGRADE AND PATCH MANAGEMENT

jueves, 6 de enero de 2022 02:05 a. m.

There are two types of upgrades:

- Major version = 5,6,7
- Minor version= 7.3 to 7.4

Major version can not be updated using: yum command

Minor version can be updated using yum update:

- yum update -y (this argument is telling to operating system that every time that need to install)

The difference between yum update and yum upgrade is that upgrade deletes packages, update preserves all the packages

Please note that yum update command will not update if your system is already up to date

## CREATE LOCAL REPOSITORY FROM CD/DVD

jueves, 6 de enero de 2022 02:31 a. m.

A repository is something where all your packages are stored and then you could download it and installed the packages that you wish to.

The command to create this is:

- **createrepo**

We can mount a device using mount and the name of device, is needed that we have the space available, to know how much space do we need is needed to use the next command:

- du -sh .

This command indicates that in the route or directory that I am it will output me the space that we require, to see the available space we can use:

- df -h

```
[root@MyFirstLinuxVM Packages]# cp -rv /run/media/iafzal/CentOS\ 7\ x86_64/Packages/* /local  
repo/
```

If the policy of the company is that we have to delete all the repositories that are inside the directory:

```
[root@MyFirstLinuxVM Packages]# cd /etc/yum.repos.d/  
[root@MyFirstLinuxVM yum.repos.d]# ls -ltr  
total 32  
-rw-r--r--. 1 root root 4768 Apr 28 12:35 CentOS-Vault.repo  
-rw-r--r--. 1 root root 1331 Apr 28 12:35 CentOS-Sources.repo  
-rw-r--r--. 1 root root 630 Apr 28 12:35 CentOS-Media.repo  
-rw-r--r--. 1 root root 314 Apr 28 12:35 CentOS-fasttrack.repo  
-rw-r--r--. 1 root root 649 Apr 28 12:35 CentOS-Debuginfo.repo  
-rw-r--r--. 1 root root 1309 Apr 28 12:35 CentOS-CR.repo  
-rw-r--r--. 1 root root 1664 Apr 28 12:35 CentOS-Base.repo  
[root@MyFirstLinuxVM yum.repos.d]# rm -rf /etc/yum.repos.d/*
```

And then create a vi file and place a name it can be named as local.repo and add the next values:

```
[centos7]  
name=centos7  
baseurl=file:///localrepo/  
enabled=1  
gpgcheck=0
```

Then createrepo using the main route:

```
[root@MyFirstLinuxVM yum.repos.d]# vi local.repo  
[root@MyFirstLinuxVM yum.repos.d]# createrepo /localrepo/
```

To list all the available repositories we have to use the next command:

```
[root@MyFirstLinuxVM yum.repos.d]# yum repolist all  
Loaded plugins: fastestmirror, langpacks  
Determining fastest mirrors  
centos7  
centos7/primary_db  
repo id repo name  
centos7 centos7  
repolist: 3,971  
[root@MyFirstLinuxVM yum.repos.d]#
```

# ADVANCE PACKAGE MANAGEMENT

jueves, 6 de enero de 2022 03:51 a. m.

In advance package management we will learn about:

- Installing packages
- Upgrading those packages
- Deleting
- View package details information
- Identify source or location information
- Packages configuration files

```
[root@myfirstlinuxos ~]# yum install ksh*
```

To insall any package related to another one you can use \* operator.

To find if I have any package installed we have to use:

```
[root@myfirstlinuxos ~]# rpm -qa | grep ksh  
ksh-20120801-143.el7_9.x86_64  
[root@myfirstlinuxos ~]# rpm -qa | grep ksh
```

To remove any package is needed to execute the next command:

```
[root@MyFirstLinuxVM ~]# yum remove ksh*
```

If you do not have internet you need to know the exact location of the file, you can google any package, then check where is located the command and then execute the next command:

```
[root@MyFirstLinuxVM ~]# rpm -hiv ksh-20120801-137.el7.x86_64.rpm
```

rpm -hiv and the location of the package.

To get for specific informaiton we have to use the next command:

```
[root@MyFirstLinuxVM ~]# rpm -qi ksh-20120801-137.el7.x86_64  
Name        : ksh  
Version     : 20120801  
Release     : 137.el7  
Architecture: x86_64  
Install Date: Sat 11 Aug 2018 05:52:54 PM EDT  
Group       : System Environment/Shells  
Size        : 3281571  
License     : EPL  
Signature   : RSA/SHA256, Wed 25 Apr 2018 07:10:35 AM EDT, Key ID 24c6a8a7f4a80eb5  
Source RPM  : ksh-20120801-137.el7.src.rpm  
Build Date  : Wed 11 Apr 2018 07:10:35 AM EDT  
Build Host  : x86-01.bsys.cen...g  
Relocations : (not relocatable)  
Packager    : CentOS BuildSystem <http://bugs.centos.org>  
Vendor      : Centos  
URL         : http://www.kornshell.com/  
Summary     : The Original ATT Korn Shell  
Description :  
KSH-93 is the most recent version of the KornShell by David Korn of  
AT&T Bell Laboratories.  
KornShell is a shell programming language, which is upward compatible  
with "sh" (the Bourne Shell).  
[root@MyFirstLinuxVM ~]#
```

We can delete a package using rpm, to do that we have to execute the next command:

```
[root@MyFirstLinuxVM ~]# rpm -qa | grep ksh
ksh-20120801-137.el7.x86_64
[root@MyFirstLinuxVM ~]# rpm -e ksh-20120801-137.el7.x86_64
```

Now if you want to see details of configuration files and other information you have to execute the next command:

```
[root@MyFirstLinuxVM ~]# rpm -qa | grep ksh
ksh-20120801-137.el7.x86_64
[root@MyFirstLinuxVM ~]# rpm -qc ksh-20120801-137.el7.x86_64
/etc/binfmt.d/kshcomp.conf
/etc/kshrc
/etc/skel/.kshrc
[root@MyFirstLinuxVM ~]#
```

To locate the full path of a command or program we have to execute the next command:

```
[root@MyFirstLinuxVM ~]# which ksh
/usr/bin/ksh
```

Now if we want to know what are the origin of the package or the origin of commands that are included in a package we have to execute:

```
[root@MyFirstLinuxVM ~]# which ksh
/usr/bin/ksh
[root@MyFirstLinuxVM ~]# rpm -qf /usr/bin/ksh
ksh-20120801-137.el7.x86_64
```

# ROLLBACK PATCHES AND UPDATES

sábado, 8 de enero de 2022 08:41 p. m.

Sometimes on cerain updates it can occur certain incompatibility issues, so is needed to rollback updates and patches:

- Virtual Machine
    - If you are in a VM the better option before make any change in the VM is needed to do a snapshot, and if something goes wrong you can revert the VM state to the snapshot.
  - Physical machine
    - Rollback a package or patch
      - yum install <package-name>
      - yum history undo <id>
    - Rollback an update
      - Downgrading a system to minor version (ex: RHEL 7.1 to RHEL 7.0) is not recommended as this might leave the system in undesired or unstable state
      - yum update= will preserve all updates
      - yum upgrade = Upgrade will delete obsolete packages
      - yum history undo <id>

To downgrade a package we have to see the history of modifications:

```
[root@myfirstlinuxos ~]# yum history
Loaded plugins: fastestmirror, langpacks
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
ID      | Login user              | Date and time   | Action(s)    | Altered
-----
 7 | TERRON VICTOR <terron> | 2021-12-06 16:26 | Install      | 1
 6 | TERRON VICTOR <terron> | 2021-12-06 06:28 | Update       | 15
 5 | TERRON VICTOR <terron> | 2021-12-06 06:08 | Install      | 2
 4 | TERRON VICTOR <terron> | 2021-12-05 21:06 | Install      | 1
 3 | TERRON VICTOR <terron> | 2021-12-01 22:28 | Install      | 1
 2 | System <unset>          | 2021-11-30 07:18 | I, U        | 266 EE
 1 | System <unset>          | 2021-11-26 00:42 | Install      | 1348
history list
```

And then executhe the next command:

```
[root@myfirstlinuxos ~]# yum history undo 7
```

That means that the change with ID 7 will be reverted, that means that you will get the older version.

If you execute:

- yum update you cannot revert any change

# SSH and TELNET

sábado, 8 de enero de 2022 09:35 p. m.

2 Main services that allow connections:

- TELNET = Un-secured connection between computers
- SSH = Secured

There are 2 types of packages for most of the services

- Client package
- Server package

```
[root@myfirstlinuxos ~]# ssh localhost
The authenticity of host 'localhost (::1)' can't be established.
ECDSA key fingerprint is SHA256:M/VXe8mn1mHiNqzNK+wc82Xm5UE2HGlzgvEY5nxxrlw.
ECDSA key fingerprint is MD5:95:34:38:bb:14:f2:c2:80:2d:41:8a:a4:6f:a3:9f:7b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
root@localhost's password:
Last login: Sat Jan  8 21:27:22 2022
[root@myfirstlinuxos ~]#
```

# DNS = DOMAIN NAME SYSTEM

sábado, 8 de enero de 2022 09:46 p. m.

## Purpose

Translate the hostname to IP	(A record)
IP to hostname	(PTR record)
Hostname to hostname	(CNAME record)

- Files for DNS configuration, the name of the process is named
  - /etc/named.conf
  - /var/named (is a directory where are located all the zone files where you define all the hostname to IP, IP to hostname or others)
- Service
  - systemctl restart named

To download, install and configure DNS we have to keep in mind the architecture of a setup DNS:

- Master DNS
  - Secondary or slave DNS (replication to the master server)
  - Client
- 
- Domain name= lab.local
  - IP address= My local IP address on enp03s

The steps:

Install DNS package

- yum install bind bind-utils -y

Configure DNS

- Modify /etc/named.conf
- Create two zone files (forward.lab(to IP to HOSTNAME) and reverse.lab(to IP to hostname))
- Modify DNS file permissions and start the service

## HOSTNAME OR IP LOOKUP (nslookup and dig)

domingo, 9 de enero de 2022 01:26 p. m.

This commands resolve IP to hostname or hostname to IP.

```
[root@MyFirstLinuxOS ~]# nslookup  
> www.google.com  
Server: 192.168.1.1  
Address: 192.168.1.1#53
```

**Non-authoritative answer:**

```
Name: www.google.com  
Address: 172.217.11.4
```

When we receive non authoritative answer means that in our modem is no information of that DNS on the local DNS server, so is needed to go out and look for that DNS but in another DNS server.

Is better to use digg because it gives you a lot of information compared than nslookup

```
[root@MyFirstLinuxOS ~]# dig www.hotmail.com  
  
; <>> DiG 9.9.4-RedHat-9.9.4-51.el7_4.2 <>> www.hotmail.com  
;; global options: +cmd  
;; Got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32549  
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1  
  
;; OPT PSEUDOSECTION:  
; EDNS: version: 0, flags:; udp: 4000  
;; QUESTION SECTION:  
;www.hotmail.com. IN A  
  
;; ANSWER SECTION:  
www.hotmail.com. 2763 IN CNAME outlook-fd-0010.live.com.  
outlook-fd-0010.live.com. 1542 IN CNAME a-0010.a-msedge.net.  
a-0010.a-msedge.net. 66 IN A 204.79.197.212  
  
;; Query time: 69 msec  
;; SERVER: 192.168.1.1#53(192.168.1.1)  
;; WHEN: Mon Apr 02 19:20:38 EDT 2018  
;; MSG SIZE rcvd: 128
```

# NETWORK TIME PROTOCOL (ntp)

domingo, 9 de enero de 2022 01:38 p. m.

- **Purpose?**

**Time synchronization**

- **File**

**/etc/ntp.conf**

- **Service**

**systemctl restart ntpd**

- **Command**

**ntpq**

To install the programm is needed to install:

- yum install ntp -y

Remember that NTP runs at port 123

## chrony LATEST VERSION TO REPLACE ntpd

domingo, 9 de enero de 2022 01:44 p. m.

The purpose is:

- **Purpose? = Time synchronization**
- **Package name = chronyd**
- **Configuration file = /etc/chronyd.conf**
- **Log file = /var/log/chrony**
- **Service = systemctl start/restart chronyd**
- **Program command = chronyc.**

```
[root@myfirstlinuxos ~]# chronyc
chrony version 3.4
Copyright (C) 1997-2003, 2007, 2009-2018 Richard P. Curnow and others
chrony comes with ABSOLUTELY NO WARRANTY. This is free software, and
you are welcome to redistribute it under certain conditions. See the
GNU General Public License version 2 for details.

chronyc> sources
210 Number of sources = 4
MS Name/IP address          Stratum Poll Reach LastRx Last sample
=====
^+ 192.36.143.130           1    6    17    10   -914us[-2016us] +/-   82ms
^* customer-187-157-26-30-s> 3    6    17    10   +7586us[+6485us] +/-  101ms
^+ host-45-231-168-6.broadb> 2    6    17    11  -4261us[-5362us] +/-   83ms
^+ bray.walcz.net            2    6    17    10  -2594us[-2594us] +/-  113ms
chronyc> [green]
```

# NEW SYSTEM UTILITY COMMAND (timedatectl)

domingo, 9 de enero de 2022 02:03 p. m.

- The timedatectl command is a new utility for RHEL/CentOS 7/8 based distributions, which comes as a part of the system and service manager
- It is a replacement for old traditional date commandThe timedatectl command shows/change date, time and timezone
- It synchronizes the time with NTP server as well
  - You can either chronyd or ntpd and make the ntp setting in timedatectl as yes
  - Or you can use systemd-timesyncd daemon to synchronize time which is a replacement for ntpd and chronyd
- PLEASE NOTE: REDHAT / CENTOS does not provide this daemon in its standard repo. You will have to download it separately

## Lab exercise:

- To check time status
  - **timedatectl**
- To view all available time zones
  - **timedatectl list-timezones**
- To set the time zone
  - **timedatectl set-timezone "America/New\_York"**
- To set date
  - **timedatectl set-time YYYY-MM-DD**
- To set date and time
  - **timedatectl set-time '2015-11-20 16:14:50'**
- To start automatic time synchronization with a remote NTP server
  - **timedatectl set-ntp true.**

# SENDMAIL

Lunes, 10 de enero de 2022 02:03 a. m.

Is one of the major services that runs for LINUX, is a program that runs in LINUX to send and receive emails it is used to send email because when you have a program running in your system and you can use it to notify when a task has been completed or anything that you want.

- **Purpose?**

**Send and receive emails**

- **Files**

**/etc/mail/sendmail.mc**

**/etc/mail/sendmail.cf**

**/etc/mail**

- **Service**

**systemctl restart sendmail**

Is needed to go to /etc/mail/sendmail.ms then to sendmail.cf

If you want to send a email you have to use the next command:

- **Command**

**mail -s "subject line" email@mydomain.com**

A mail relay server is a server that is setup in a company, an environment which accept your emails that are coming from any of the server and then deliver, when you make vi to a configuration file everithing that starts with dnl is a command so you can ignore it.

```

File Edit View Search Terminal Help
divert(-1)dnl
dnl #
dnl # This is the sendmail macro config file for m4. If you make changes to
dnl # /etc/mail/sendmail.mc, you will need to regenerate the
dnl # /etc/mail/sendmail.cf file by confirming that the sendmail-cf package is
dnl # installed and then performing a
dnl #
dnl #      /etc/mail/make
dnl #
include(`/usr/share/sendmail-cf/m4/cf.m4')dnl
VERSIONID(`setup for linux')dnl
OSTYPE(`linux')dnl
dnl #
dnl # Do not advertize sendmail version.
dnl #
dnl define(`confSMTP_LOGIN_MSG', `$j Sendmail; $b')dnl
dnl #
dnl # default logging level is 9, you might want to set it higher to
dnl # debug the configuration
dnl #
dnl define(`confLOG_LEVEL', `9')dnl
dnl #
dnl # Uncomment and edit the following line if your outgoing mail needs to
dnl # be sent out through an external mail server:
dnl #
dnl define(`SMART_HOST', `smtp.your.provider')dnl
dnl #
define(`confDEF_USER_ID', ``8:12')dnl
dnl define(`confAUTO_REBUILD')dnl
define(`confTO_CONNECT', `1m')dnl
define(`confTRY_NULL_MX_LIST', `True')dnl
define(`confDONT_PROBE_INTERFACES', `True')dnl
define(`PROCMAIL_MAILER_PATH', `/usr/bin/procmail')dnl
define(`ALIAS_FILE', `/etc/aliases')dnl
"sendmail.mc" 178L, 7306C

```

And then look up for:

```

dnl #
dnl define(`SMART_HOST', `smtp.your.provider')dnl
dnl #
define(`confDEF_USER_ID', ``8:12')dnl
dnl define(`confAUTO_REBUILD')dnl
define(`confTO_CONNECT', `1m')dnl

```

Then change it to you server domain, then that you configure it you have to start the service.

```

[root@myfirstlinuxos mail]# vi sendmail.mc
[root@myfirstlinuxos mail]# systemctl start sendmail.service
[root@myfirstlinuxos mail]#

```

To send a email you have to use:

```

[root@myfirstlinuxos mail]# mail -s "SEND EMAIL TEST" victor@terronm.com
TEST EMAIL JUST FOR FUN

EOT
[root@myfirstlinuxos mail]#

```

And to go out of writting the email you have to use CTRL+D

# WEB SERVER (Apache-HTTP)

lunes, 10 de enero de 2022 02:56 a. m.

- Purpose = Serve webpages



- Service or Package name = **httpd**
- Files = **/etc/http/conf/httpd.conf**  
**= /var/www/html/index.html**
- Service
  - systemctl restart httpd**
  - systemctl enable httpd**

- **Log Files = /var/log/httpd/**

To install the package is needed to execute the next command:

- **yum install httpd**

After that package is installed we have to do vi on configuration file:

- **vi /etc/httpd/conf/httpd.conf**

The line with DocumentRoot "/var/www/html" is where every time that you enter to your page is where the index file will be located

# CENTRAL LOGGER (rsyslog)

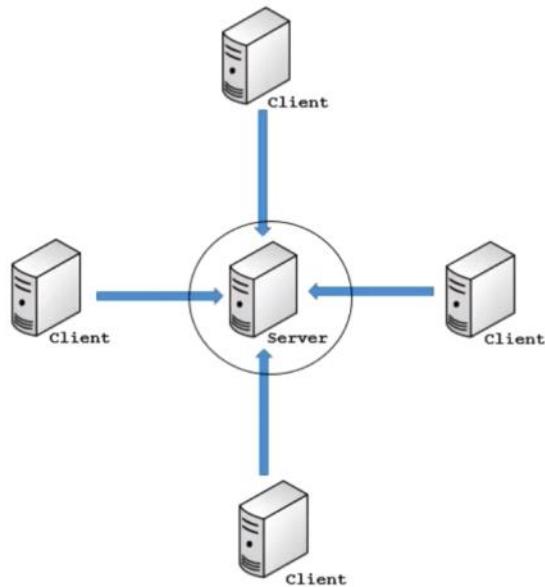
Lunes, 10 de enero de 2022 03:47 a. m.

Is a server that generates logs or collect logs from other servers:

- Purpose = Generate logs or collect logs from other servers
- Service or package name = **rsyslog**
- Configuration file= **/etc/syslog.conf**  
**Correction: /etc/rsyslog.conf**

- Service

```
systemctl restart rsyslog  
systemctl enable rsyslog
```



Is needed when you have a lot of servers to avoid to log in every server and see where is the problem or what server is experiencing troubles, this service is used for that purpose.

You have to edit the config file:

```
#  
# An on-disk queue is created for this action. If the remote host is  
# down, messages are spooled to disk and sent when it is up again.  
#$ActionQueueFileName fwdRule1 # unique name prefix for spool files  
#$ActionQueueMaxDiskSpace 1g    # 1gb space limit (use as much as possible)  
#$ActionQueueSaveOnShutdown on # save messages to disk on shutdown  
#$ActionQueueType LinkedList  # run asynchronously  
#$ActionResumeRetryCount -1   # infinite retries if host is down  
# remote host is: name/ip:port, e.g. 192.168.0.1:514, port optional  
#*[* @@remote-host:514  
# ### end of the forwarding rule ###
```

We can make the supposition that if our client is the console of PUTTY we can send log entries to another central logger that is another server we just have to specify the IP address and the rsyslog runs by default on port 514 in the line that is in the photo.

# SECURING LINUX MACHINE(LINUX OS HARDENING)

Lunes, 10 de enero de 2022 04:09 a. m.

- User accounts
  - Is recommended to use the user id up than 10000
  - chage command, set a date that expires the password
  - cat /etc/shadow file you will see the password and the parameters of chage command or chage -l userName
  - cat /etc/login.defs this is the file that you could also define some of the parameters and those parameters that you have to modify to change default parameters for every user
  - cat /etc/pam.d/system-auth
- Remove un-wanted packages
  - To list all packages you have to use rpm -qa, to count it you have to use rpm -qa | wc -l
  - To remove a package you have to use rpm -e namePackage
- Stop un-used services
  - To get a list of all services that are running or are inactive you have to use systemctl -a
- Check on listening ports
  - netstat -tunlp, with this command you will see all the ports that are active and listening right now
- Secure SSH configuration
  - cat /etc/ssh/sshd\_config, you will see the file and you can change the port and disable the parameter PermitRootLogin CHANGE IT TO NO
- Enable firewall (iptables/firewalld)
  - It is iptables to lower versions of LINUX 6 to lower, and for newer versions it is called as firewalld
  - It serves to filter the inbound and outbound connections, it has to be configured to allow connections through ports
  - The commands that you have to know if you have a GUI is firewalld-config
  - If you do not have a GUI you have to use the next command: firewall-cmd or iptables
  - The configuration files for iptables is cat /etc/sysconfig/iptables-config
  - The configuration file for firewalld is cat /etc/firewalld/firewalld.conf
- Enable SELinux
  - Stands for Security and has Linux, is a secure architecture integrated into the 2.6 kernel using the LINUX security module also known as LSM, this defines the access and transmission rights of every user, every application process and the file systems, this module of security is useful when you want to control the permission of a process of an application, this controls the permission access of a process an application and every access of a file
  - To enable is run the next command: sestatus
  - To disable you have to go to: cat /etc/sysconfig/selinux, and then change the line that says SELINUX=enforcing, remember that SELinux have 3 modes
    - Enforcing, means that SELinux security policies enforced
    - Permissive means that SELinux system prints warnings but does not enforce policies, it will print a warning but it will not stop
    - Disable

```
[root@MyFirstLinuxOS iafzal]# stat laland
  File: 'laland'
  Size: 7          Blocks: 8          IO Block: 4096   regular file
Device: fd00h/64768d  Inode: 5251664      Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1000/  iafzal)  Gid: ( 1000/  iafzal)
Context: unconfined_u:object_r:user_home_t:s0
Access: 2018-06-03 17:34:17.037352147 -0400
Modify: 2018-05-02 09:35:46.960633475 -0400
Change: 2018-05-02 09:36:04.231849918 -0400
 Birth: -
```

The command stat andNameOfFile tells you detailed information about a file

- Another command that allows you to change file SELinux security context is chcon
- Another command that allows you to set SELinux policy compiler is checkpolicy
- Change Listening servers Port Numbers
- Keep your OS up to date (security patching)
  - It is not to make yum update or something like that, is to update everything that is related to security, is recommended to have a notification that keeps you informed about every issue that you can find in every OS

# OpenLDAP installation

martes, 11 de enero de 2022 01:52 a. m.

This lecture is related to the creation of users and management of the users:

- OpenLDAP is an open source implementation of active directory

The service that we have to keep running is called:

- slapd

To start or stop the service have to use the next commands:

- systemctl start slapd
- systemctl enable slapd

Once you have the OpenLDAP file the configuration files are located in:

- /etc/openldap/slapd.d

If you want to know the PID process or the status of the process another way to make that is to use:

```
[root@myfirstlinuxos ~]# ps -ef | grep slapd
root      3844 26300  0 10:28 pts/2    00:00:00 grep --color=auto slapd
[root@myfirstlinuxos ~]#
```

Another useful file is:

```
[root@myfirstlinuxos etc]# pwd
/etc
[root@myfirstlinuxos etc]# more nsswitch.conf
```

# TRACING NETWORK TRAFFIC (traceroute)

martes, 11 de enero de 2022 02:17 a. m.

In this lesson we will actually learn and try to understand how we can troubleshoot any issue in an environment, a command that is mostly used to see where your traffic is going, which gateway or which DNS is going from one to the others and finally how it is going to reach the destination and to view that entire tracing of those traffic you have to use the command:

- traceroute

- The traceroute command is used in Linux to map the journey that a packet of information undertakes from its source to its destination. One use for traceroute is to locate when data loss occurs throughout a network, which could signify a node that's down.
- Because each hop in the record reflects a new server or router between the originating PC and the intended target, reviewing the results of a traceroute scan also lets you identify slow points that may adversely affect your network traffic.

- Example

```
# traceroute www.google.com
```

The output is:

```
[root@myfirstlinuxos etc]# traceroute 192.168.101.44
traceroute to 192.168.101.44 (192.168.101.44), 30 hops max, 60 byte packets
 1 myfirstlinuxos.domain.name (192.168.101.44)  0.045 ms  0.009 ms  0.007 ms
[root@myfirstlinuxos etc]# traceroute www.google.com
traceroute to www.google.com (142.251.34.228), 30 hops max, 60 byte packets
 1 RTK_GW.domain.name (192.168.101.1)  2.365 ms  5.529 ms  1.960 ms
 2 172.16.0.1 (172.16.0.1)  2.728 ms  1.377 ms  1.528 ms
 3 fixed-189-203-176-241.totalplay.net (189.203.176.241)  2.742 ms  2.357 ms  2.587 ms
 4 10.116.101.1 (10.116.101.1)  3.175 ms  3.329 ms  2.863 ms
 5 10.180.49.7 (10.180.49.7)  7.482 ms  5.986 ms  8.653 ms
 6 10.180.49.6 (10.180.49.6)  6.250 ms  8.228 ms  8.214 ms
 7 74.125.119.138 (74.125.119.138)  19.585 ms  25.931 ms  21.281 ms
 8 * *
 9 142.250.211.196 (142.250.211.196)  21.517 ms 142.250.211.192 (142.250.211.192)  20.995 ms 142.250.211.84 (142.250.211.84)  10.863 ms
10 108.170.254.5 (108.170.254.5)  19.041 ms 142.251.78.199 (142.251.78.199)  18.691 ms 108.170.254.8 (108.170.254.8)  20.047 ms
11 142.251.79.213 (142.251.79.213)  19.718 ms 108.170.226.127 (108.170.226.127)  21.155 ms  19.772 ms
12 74.125.243.33 (74.125.243.33)  11.767 ms 108.170.226.127 (108.170.226.127)  20.808 ms qro02s28-in-f4.1e100.net (142.251.34.228)  18.913 ms
[root@myfirstlinuxos etc]#
```

# HOW TO OPEN IMAGE THROUGH COMMAND LINE

martes, 11 de enero de 2022 02:25 a. m.

To open a image we have to install a programm that allow us to open image file  
Command syntax to open image file in GUI.

- Become root
- yum install ImageMagick -y
- display imageNameFile

For CentOS/RHEL 8 version the package name is GraphicsMagick

And the command is dnf install GraphicsMagic

```
[root@myfirstlinuxos ~]# display /home/terron/Desktop/imagenLINUX.jpg
```

# CONFIGURE AND SECURE SSH

martes, 11 de enero de 2022 03:14 a. m.

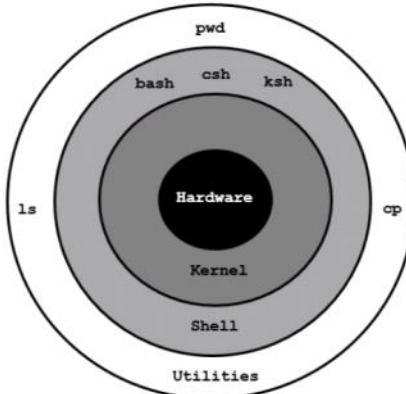
## • SSH

- SSH stands for secure shell

→ provides you with an interface to the Linux system. It takes in your commands and translate them to kernel to manage hardware



- Open SSH is a package/software
- Its service daemon is sshd
- SSH port # 22



NOTE: DAEMON is something that always runs in the background all the time and listens for commands or executions.

- SSH itself is secure, meaning communication through SSH is always encrypted, but there should be some additional configuration can be done to make it more secure
- Following are the most common configuration an administrator should take to secure SSH

### ✓ Configure Idle Timeout Interval

Avoid having an unattended SSH session, you can set an Idle timeout interval

- Become root
- Edit your `/etc/ssh/sshd_config` file and add the following line:
  - `ClientAliveInterval 600`
  - `ClientAliveCountMax 0`
  - `# systemctl restart sshd`

The idle timeout interval you are setting is in seconds (600 secs = 10 minutes). Once the interval has passed, the idle user will be automatically logged out

Another measure that you have to take when configuring system for the first time is to:

## ✓ **Disable root login**

Disabling root login should be one of the measures you should take when setting up the system for the first time. It disable any user to login to the system with root account

- Become root
- Edit your **/etc/ssh/sshd\_config** file and replace PermitRootLogin yes to no
- **PermitRootLogin no**
- **# systemctl restart sshd**

## ✓ **Limit Users' SSH Access**

To provide another layer of security, you should limit your SSH logins to only certain users who need remote access

- Become root
- Edit your **/etc/ssh/sshd\_config** file and add
- **AllowUsers user1 user2**
- **# systemctl restart sshd**

## ✓ **Use a different port**

By default SSH port runs on 22. Most hackers looking for any open SSH servers will look for port 22 and changing can make the system much more secure

- Become root
- Edit your **/etc/ssh/sshd\_config** file and remove # from the following line and change the port number
- **Port 22**
- **# systemctl restart sshd**

# SSH-KEYS - ACCESS REMOTE SERVER WITHOUT PASSWORD

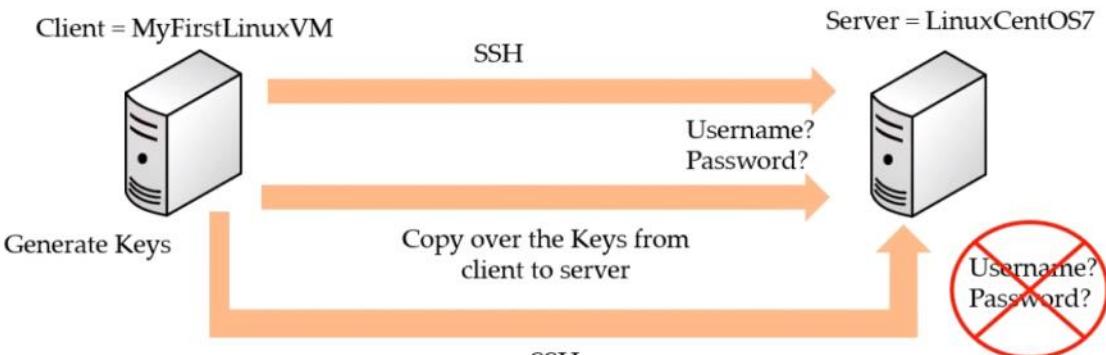
martes, 11 de enero de 2022 03:37 a. m.

Two reasons to access a remote machine:

- Repetitive logins
- Automation through scripts

Keys are generated at user level:

- terron
- root



## Client = MyFirstLinuxVM

Step 1 — Generate the Key

```
# ssh-keygen
```

Step 2 — Copy the Key to the server

```
# ssh-copy-id root@192.168.1.x
```

Step 3 — Login from client to server

```
# ssh root@192.168.1.x
```

```
# ssh -l root 192.168.1.x
```



```
terron@linuxcentos7:~$ Enter file in which to save the key (/home/terron/.ssh/id_rsa): /home/terron/.ssh/id_rsa already exists. Overwrite (y/n)? [terron@myfirstlinuxos ~]$ ssh-keygen Generating public/private rsa key pair. Enter file in which to save the key (/home/terron/.ssh/id_rsa): /home/terron/.ssh/id_rsa already exists. Overwrite (y/n)? y Enter passphrase (empty for no passphrase): Enter same passphrase again: Your identification has been saved in /home/terron/.ssh/id_rsa. Your public key has been saved in /home/terron/.ssh/id_rsa.pub. The key's fingerprint is: SHA256:LdZhfvKwCN4jdKcN1Eil/AfuBHB6ZNPSzXg2E6qCg terron@myfirstlinuxos The key's randomart image is: +---[RSA 2048]----+ | oo | | .o.+. | | ..Boor...| | .o +**o==o| | +..S.=+o o| | E ..oo. ...| | .. o . .| | . . .| | o. | +---[SHA256]----+ [terron@myfirstlinuxos ~]$ ssh-copy-id terron@192.168.101.49 /usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/terron/.ssh/id_rsa.pub" /usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed /usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys terron@192.168.101.49's password: Number of key(s) added: 1 Now try logging into the machine, with: "ssh 'terron@192.168.101.49'" and check to make sure that only the key(s) you wanted were added. [terron@myfirstlinuxos ~]$ ssh terron@192.168.101.49 Last login: Tue Jan 11 04:09:49 2022 from myfirstlinuxos.domain.name [terron@linuxcentos7 ~]$
```

```
[terron@linuxcentos7:~]$ 5°C Parc. nublado 04:11 a. m. 11/01/2022
```

# LINUX WEB-BASED ADMINISTRATION (cockpit)

martes, 11 de enero de 2022 04:13 a. m.

Is a server administration tool sponsored by Red Hat, focused on providing a modern-looking and user-friendly interface to manage and administer servers.

Cockpit is the easy to use, integrated, glanceable, and open web-based interface for your servers. The application is available in most of the linux distributions such as CentOS, RedHat, Ubuntu and Fedora

It is installed in Redhat 8 by default and it is optional in version 7

It can monitor you resources, add or remove accounts, monitor system usage, shut down the system and perform quite a few other tasks all through a very accessible web connection

The steps to install, configure and manage cockpit is:

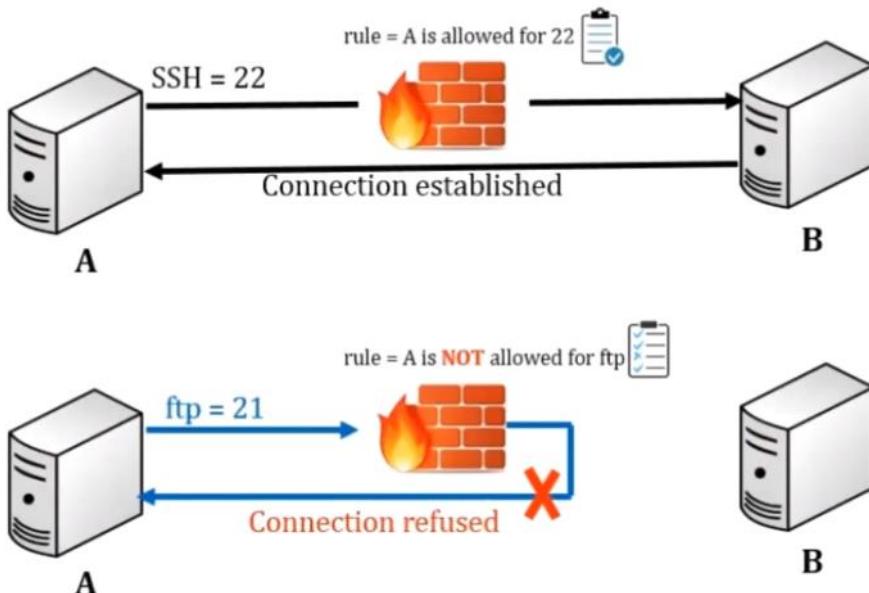
- Check for network connectivity
  - ping -c1 [www.google.com](http://www.google.com)
- Install cockpit package as root
  - yum/dnf install cockpit -y (REDHAT CentOS)
  - apt-get install cockpit (UBUNTU)
- Start and enable the service
  - Systemctl start|enable cockpit
- To enter into the web interface you have to login into 192.168.101.xx:9090

# FIREWALL

miércoles, 12 de enero de 2022 01:08 a. m.

Firewall is a wall that prevents the spread of fire.

- When data moves in and out a server its packet information is tested against the firewall rules to see if it should be allowed or not
- In simple words, a firewall is like a watchman, a bouncer, or a shield that has a set of rules given and based on that rule they decide who can enter and leave
- There are two types of firewalls in IT
  - SOFTWARE=Runs operating system
  - HARDWARE = A dedicated appliance with firewall software



## FIREWALL (**iptables**)

- There are 2 tools to manage firewall in most of the linux distributions
  - Iptables = For older LINUX versions but still widely used
  - Firewalld = For newer versions like 7 and up
- You can run one or other
  - In this lecture we will work with iptables to manage firewall
- You can run one or the other
  - In this lecture we will work with **iptables** to manage firewall
  - Before working with iptables make sure firewalld is not running and disable it
    - service OR systemctl stop firewalld = To stop the service
    - systemctl disable firewalld = To prevent from starting at boot time
    - systemctl mask firewalld = To prevent it from running by other programs
  - Now check if you have iptables-services package installed
    - rpm -qa | grep iptables-services
    - yum install iptables-services - If not installed then
  - Start the service
    - systemctl start iptables
    - systemctl enable iptables
  - To check the iptables rules
    - iptables -L
  - To flush iptables.
    - iptables -F



Remember: The command systemctl mask is for you no start by accident any service that you do not want to start

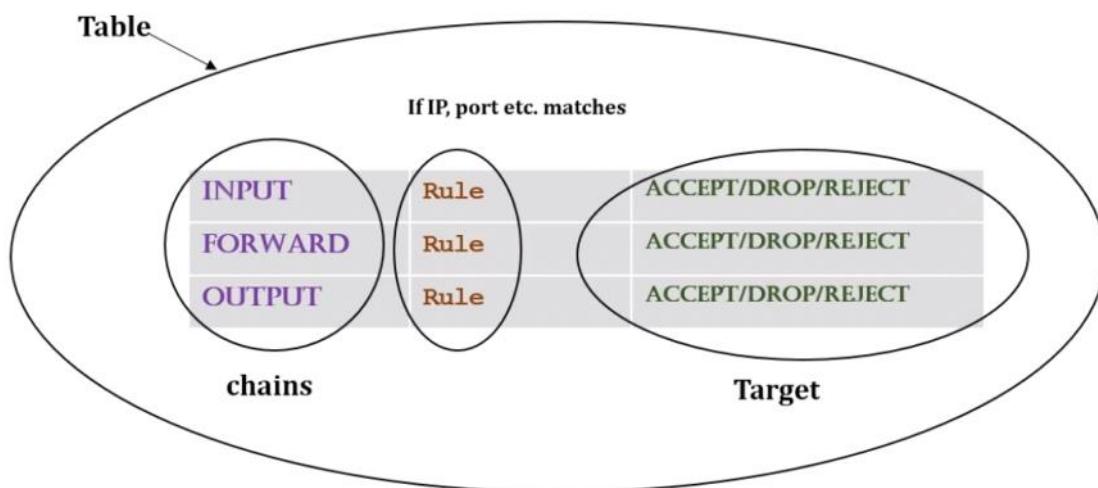
To install iptables-service is needed to use:

- yum install iptables-services -y

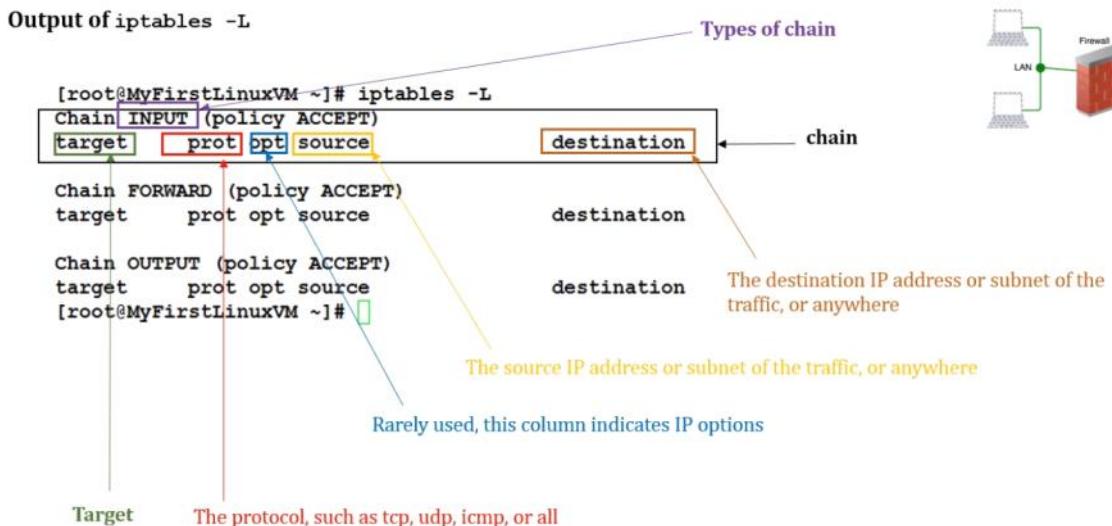
- The function of iptables tool is packet filtering
- The packet filtering mechanism is organized into three different kinds of structures: **tables**, **chains** and **targets**

1. **tables** = table is something that allows you to process packets in specific ways. There are 4 different types of tables, filter, mangle, nat and raw
2. **chains** = The chains are attached to tables, These chains allow you to inspect traffic at various points. There are 3 main chains used in iptables
  - **INPUT** = incoming traffic
  - **FORWARD** = going to a router, from one device to another
  - **OUTPUT** = outgoing traffic
    - chains allow you to filter traffic by adding rules to them
    - Rule = if traffic is coming from 192.168.1.35 then go to defined target
3. **targets** = target decides the fate of a packet, such as allowing or rejecting it. There are 3 different type of targets
  - **ACCEPT** = connection accepted
  - **REJECT** = Send reject response
  - **DROP** = drop connection without sending any response

**Let's draw it out:**



To see the table you have to use the **iptables -L** command



## FIREWALL (firewalld)

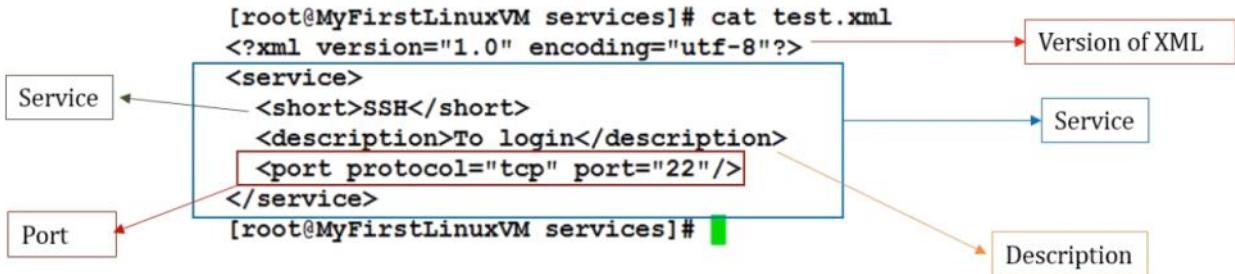
- Firewalld works the same way as iptables but of course it has its own commands
  - **firewall-cmd**
- It has a few pre-defined service rules that are very easy to turn on and off
  - Services such as: NFS, NTP, HTTPD etc.
- Firewalld also has the following:
  - **Table**
  - **Chains**
  - **Rules**
  - **Targets**
- You can run one or the other
  - iptables or firewalld
- Make sure iptables is stopped, disabled and masked
  - **systemctl stop iptables**
  - **systemctl disable iptables**
  - **systemctl mask iptables**
- Now check if firewalld package is installed
  - **rpm -qa | grep firewalld**
- Start firewalld
  - **systemctl start/enable firewalld**
- Check the rule of firewalld
  - **firewall-cmd --list-all**
- Get the listing of all services firewalld is aware of:
  - **firewall-cmd --get-services**
- To make firewalld re-read the configuration added

To make firewalld re-read the configuration added you have to use: **firewall-cmd --reload** command

- The firewalld has multiple zones, to get a list of all zones
  - **firewall-cmd --get-zones**
- To get a list of active zones
  - **firewall-cmd --get-active-zones**

```
[root@MyFirstLinuxVM ~]# firewall-cmd --get-zones
block dmz drop external home internal public trusted work
[root@MyFirstLinuxVM ~]# firewall-cmd --get-active-zones
public
  interfaces: enp0s3
[root@MyFirstLinuxVM ~]#
```

- To get firewall rules for public zone
  - `firewall-cmd --zone=public --list-all`  
OR
  - `firewall-cmd --list-all`
- All services are pre-defined by firewalld. What if you want to add a 3<sup>rd</sup> party service
  - `/usr/lib/firewalld/services/allservices.xml`
  - Simply cp any .xml file and change the service and port number



If we want to add a service for example http:

- `firewall-cmd --add-service=http`

If you want to remove a service you have to use:

- `firewall-cmd --remove-service=http`

If you want to reload the firewalld configuration, this wipes every change that you made into the configuration:

- `firewall-cmd --reload`

To add or remove a service permanently:

- `firewall-cmd --add-service=http --permanent`
- `firewall-cmd --remove-service=http --permanent`

• To add a service (http)

- `firewall-cmd --add-service=http`

• To remove a service

- `firewall-cmd --remove-service=http`

• To reload the firewalld configuration

- `firewall-cmd --reload`

• To add or remove a service permanently

- `firewall-cmd --add-service=http --permanent`
- `firewall-cmd --remove-service=http --permanent`

• To add a service that is not pre-defined by firewalld

- `/usr/lib/firewalld/services/allservices.xml`
- Simply cp any .xml file sap.xml and change the service and port number (32)
- `systemctl restart firewalld`
- `firewall-cmd --get-services` (to verify new service)
- `Firewall-cmd --add-service=sap`

• To add a port

- `firewall-cmd --add-port=1110/tcp`

• To remove a port

- `firewall-cmd --remove-port=1110/tcp`

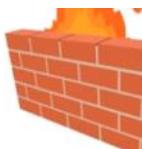
• To reject incoming traffic from an IP address

- `firewall-cmd --add-rich-rule='rule family="ipv4" source address="192.168.0.25" reject'`

• To block and unblock ICMP incoming traffic

- `firewall-cmd --add-icmp-block-inversion`
- `firewall-cmd --remove-icmp-block-inversion`

To avoid that somebody else makes you ping you have to block ICMP incoming traffic.



- To block outgoing traffic to a specific website/IP address

- host -t a [www.facebook.com](http://www.facebook.com) = find IP address
  - firewall-cmd --direct --add-rule ipv4 filter OUTPUT 0 -d 31.13.71.36 -j DROP

```
[root@MyFirstLinuxVM services]# firewall-cmd --direct --add-rule ipv4 filter OUTPUT 0 -d 31.13.71.36 -j DROP
success
[root@MyFirstLinuxVM services]# ping 31.13.71.36
PING 31.13.71.36 (31.13.71.36) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
```

## TUNE SYSTEM PERFORMANCE (tuned, nice and renice)

miércoles, 12 de enero de 2022 03:16 a. m.

This is useful when we are trying to:

- Optimize system performance by selecting a tuning profile managed by the tuned daemon
- Prioritize or de-prioritize specific processes with ht nice and renice commands

Tuned is: Tune is for system tuning and d is for daemon

- It is a systemd service that is used to tune LINUX system performance
- It is installed in CentOS/Redhat version 7 and 8 by default
- tuned package name is named tuned
- The tuned service comes with pre-defined profiles and settings (List of profile will be discussed in the next page)
- Based on selected profile the tuned service automatically adjusts system to get the best performance. E.G. tuned will adjust networking if you are downloading a large file or it will adjust IO settings if it detects high storage read/write
- The tuned daemon applies system settings when the service starts or upon selection of a new tuning profile

Tuned profile	Purpose
balanced	Designed for systems that require a compromise between power saving and performance
desktop	Derived from the balanced profile. Provides faster response of interactive applications
Throughput-performance	Tunes the system for maximum throughput
Latency-performance	Ideal for server systems that require low latency at the expense of power consumption
network-latency	Derived from the latency-performance profile. It enables additional network tuning parameters to provide low network latency
Network-throughput	Derived from the throughput-performance profile. Additional network tuning parameters are applied for maximum network throughput
powersave	Tunes the system for maximum power saving
oracle	Optimized for Oracle database loads based on the throughput-performance profile
virtual-guest	Tunes the system for maximum performance if it runs on a virtual machine
virtual-host	Tunes the system for maximum performance if it acts as a host for virtual machines

- Check if tuned package has been installed  
`rpm -qa | grep tuned`
- Install tuned package if NOT installed already  
`yum install tuned`
- Check **tuned** service status  
`systemctl status|enable|start tuned`  
`systemctl enable tuned (To enable at boot time)`
- Command to change setting for tuned daemon  
**tuned-adm**
- To check which profile is active  
**tuned-adm active**
- To list available profiles  
**tuned-adm list.**
- To change to desired profile  
**tuned-adm profile profile-name**
- Check for tuned recommendation  
**tuned-adm recommend**
- Turn off tuned setting daemon  
**tuned-adm off**
- Change profile through web console  
`Login to https://192.168.1.x:9090`  
`Overview → Configuration → Performance profile`

**nice/renince COMMAND**

- Another way of keeping your system fine-tuned is by prioritizing processes through **nice** and **renice** command
- If a server has 1 CPU then it can execute **1** computation/process at a time as they come in (*first come first served*) while other processes must wait
- With **nice** and **renice** commands we can make the system to give preference to certain processes than others
- This priority can be set at 40 different levels
- The nice level values range from -20 (highest priority) to 19 (lowest priority) and by default, processes inherit their nice level from their parent, which is usually 0.
- To check process priority  
**top**



Nice value is a user-space and priority PR is the process's actual priority that use by Linux kernel. In Linux system priorities are 0 to 139 in which 0 to 99 for real time and 100 to 139 for users

NOTE: Process priority can be viewed through ps command as well with the right options

- ps axo pic,comm,nice,cls --sort=nice
- To set the process priority  
**nice -n # process-name**  
e.g. **nice -n -15 top**
- To change the process priority  
**renice -n # process-name**  
e.g. **renice -n 12 PID.**

renice command is for running processes that you dont want to stop

## RUN CONTAINERS

miércoles, 12 de enero de 2022 03:27 a. m.

The term came from the shipping container.

- The container technology which allowed to developers or programmer to test and build applications on any computer just by putting it in a container (bundled in whith the software code, libraeries and configuration files) and then run on another computer regardless of its architecture
- You can move the application anywhere without moving OS just like the actual physical container that would fit on any.
- An OS can run single or multiple containers at the same time.

### Please Note:

**Container technology is mostly used by developers or programmers who write codes to build applications**

**As a system administrator your job is to install, configure and manage them.**

### What are the Container Software?

Developed by:  
Solomon Hykes



Released on:  
March 20th 2013

Developed by:  
 Red Hat



Released on:  
August 2018

**podman**

- Docker is the software used to create and manage containers
- Just like any other package, docker can be installed on your Linux system and its service or daemon can be controlled through native Linux service management tool

- Podman is an alternative to docker
- Docker is not supported in RHEL 8
- It is daemon less, open source, Linux-native tool designed to develop, manage, and run containers.

Red Hat provides a set of command-line tools that can operate without a container engine, these include:

- **podman** - for directly managing pods and container images (run, stop, start, ps, attach, etc.)
- **buildah** - for building, pushing, and signing container images
- **skopeo** - for copying, inspecting, deleting, and signing images
- **runc** - for providing container run and build features to podman and buildah
- **crun** - an optional runtime that can be configured and gives greater flexibility, control, and security for rootless containers.

The terms that you have to know when talking about containers is:

- **images** - containers can be created through images and containers can be converted to images
- **pods** - Group of containers deployed together on the host. In podman logo there are 3 seals grouped together as pod.

## BUILDIN, RUNNING AND MANAGIND CONTAINERS

The steps are:

- yum/dnf install podman -y
- yum install docker -yo

To install podman

- **yum/dnf install podman -y**
- **yum install docker -y** (*For dockers*)

Creating alias to docker

- **alias docker=podman**

Check podman version

- **podman -v**

Getting help

- **podman --help or man podman**

Check podman environment and registry/repository information

- **podman info** (*If you are trying to load a container image, then it will look at the local machine and then go through each registry by the order listed*)

To search a specific image in repository.

- **podman search httpd**

To list any previously downloaded podman images

- **podman images**

To download available images

- **podman pull docker.io/library/httpd**
- **podman images** (*Check downloaded image status*)

To list podman running containers

- **podman ps**

To run a downloaded httpd containers

- **podman run -dt -p 8080:80/tcp docker.io/library/httpd**  
*(d=detach, t=get the tty shell, p=port)*
- **podman ps**      *or Check httpd through web browser*

To view podman logs.

- **podman logs -l**

To stop a running container

- **podman stop con-name**    (*con-name from podman ps command*)
- **podman ps**        (*To list running containers*)

To run a multiple containers of httpd by changing the port #

- **podman run -dt -p 8081:80/tcp docker.io/library/httpd**
- **podman run -dt -p 8082:80/tcp docker.io/library/httpd**
- **podman ps**

To stop and start a previously running container

- **podman stop|start con-name**

To create a new container from the downloaded image

- **podman create --name httpd docker.io/library/httpd**

To start the newly created container.

- **podman start httpd**

### Manage containers through systemd

- First you have to generate a unit file
  - **podman generate systemd --new --files --name httpd**
- Copy it systemd directory
  - **cp /root/container-httpd.service /etc/systemd/system**
- Enable the service
  - **systemctl enable container-httpd.service**
- Start the service.
  - **systemctl start container-httpd.service**

# RESUME

jueves, 13 de enero de 2022 01:54 a. m.

What is the difference between static and DHCP IP?

- Static is permanent and DHCP changes

In which file you define static IP?

- /etc/sysconfig/network-scripts/ifcfg-nic

What is /etc/hosts file used for?

- It resolves hostname and IP

How to check if you have a package installed in your Linux system?

- rpm

What is the purpose of yum command?

- The command yum is used to upgrade your system, It is used to update your system, It downloads and install a package

What is the purpose of DNS?

- DNS is to resolve hostname to IP, DNS is to resolve IP to hostname, DNS is to resolve hostname to hostname

The default port used for DNS

- 53

NTP configuration file is?

- /etc/ntp.conf

Sendmail configuration files are:

- /etc/mail/sendmail.mc
- /etc/mail/sendmail.cf

The package name of FNS is:

- bind

Which of the following a computer needs to be connected to other computers?

- Network Interface, IP, Subnet mask, Gateway

Which of the following commands are used to check Linux system IP address?

- ip, ifconfig

Why tcpdump command is used for?

- tcpdump is a most powerful and widely used command-line packets sniffer or package analyzer tool which is used to capture or filter TCP/IP packets that received or transferred over a network on a specific interface

Which of the following command is used to check ports status?

- netstat

If you want to find out NIC speed, which command you will run?

- ethtool

Which one of the following will install NIC bonding driver?

- modprobe bonding

For NIC bonding where the ifcfg-bond0 is created?

- /etc/sysconfig/network-scripts

If you are pulling a file from FTP server then which of the following command is used?

- get

How to copy the file "seinfeld.txt" from a remotehost.edu to the local host /tmp directory

- scp your\_username@remotehost.edu:seinfeld.txt /tmp

Sync a File seinfeld.tar on a Local Computer directory /tmp/backups

- rsync -zvh seinfeld.tar /tmp/backups/

What is the difference between "yum update" and "yum upgrade"

- yum update will update the packages and skip removing obsolete packages.  
Yum upgrade will update the packages and remove the obsolete packages.

Which of the following is sshd configuration file?

- /etc/ssh/sshd\_config

How to start chronyd service in Linux?

- systemctl start chronyd

Where is index.html file location in Linux Redhat or CentOS?

- /var/www/html

Which of the following are security measures?

- Enabling firewall, Enabling SELinux, Removing packages that are not needed, Disable or stop un-wanted services

# SYSTEM RUN LEVEL

jueves, 13 de enero de 2022 02:19 a. m.

Its like safe mode in WINDOWS, in LINUX there are 7 levels.

The command is: **init**

**To run this command you have to be root**

Main run level:

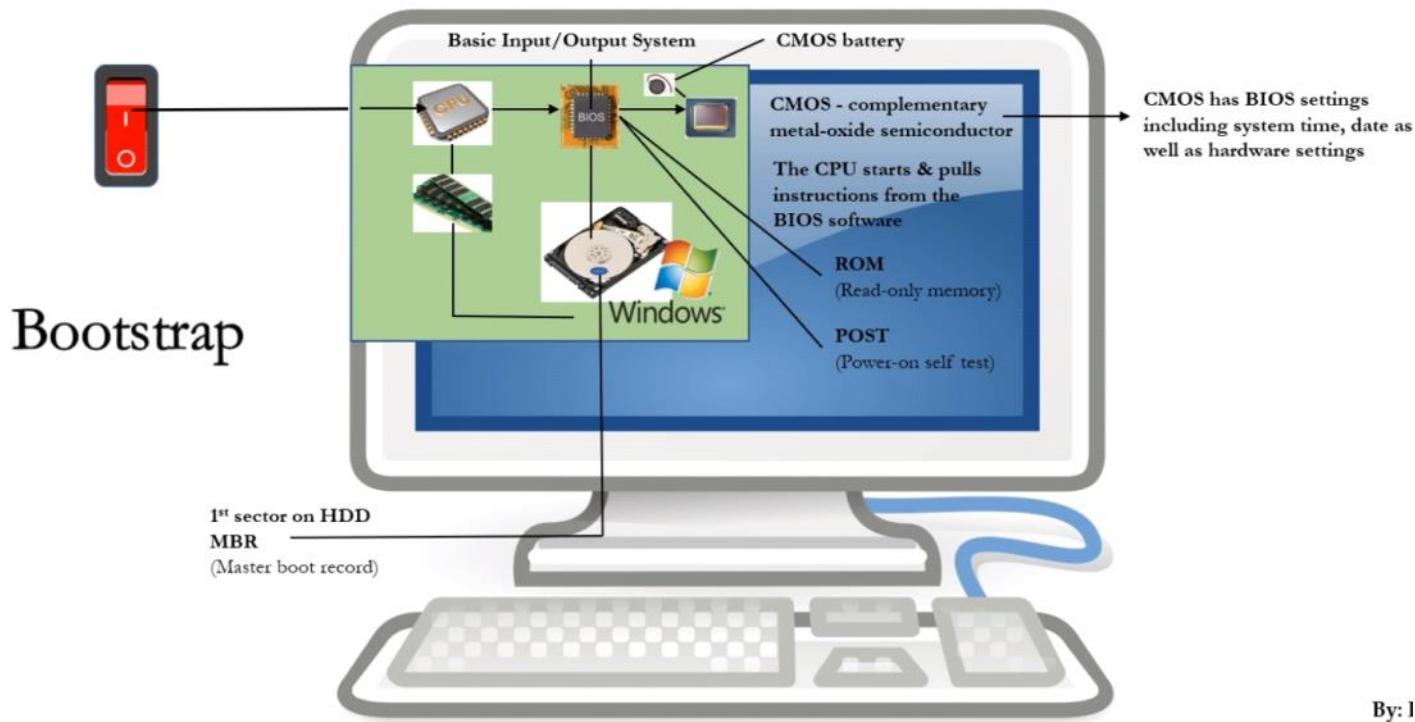
0	Shutdown (or halt the system)
1	Single-user mode; usually aliased as s or S
6	Reboot the system
• 2	Multiuser mode without networking
3	Multiuser mode with networking without GUI
5	Multiuser mode with networking and GUI
4	Undefined or not used/User-definable

- To find the run level that you are have to use:
  - `who -r`

# COMPUTER BOOT PROCESS

jueves, 13 de enero de 2022 02:28 a. m.

No matter how computer you have it is the same



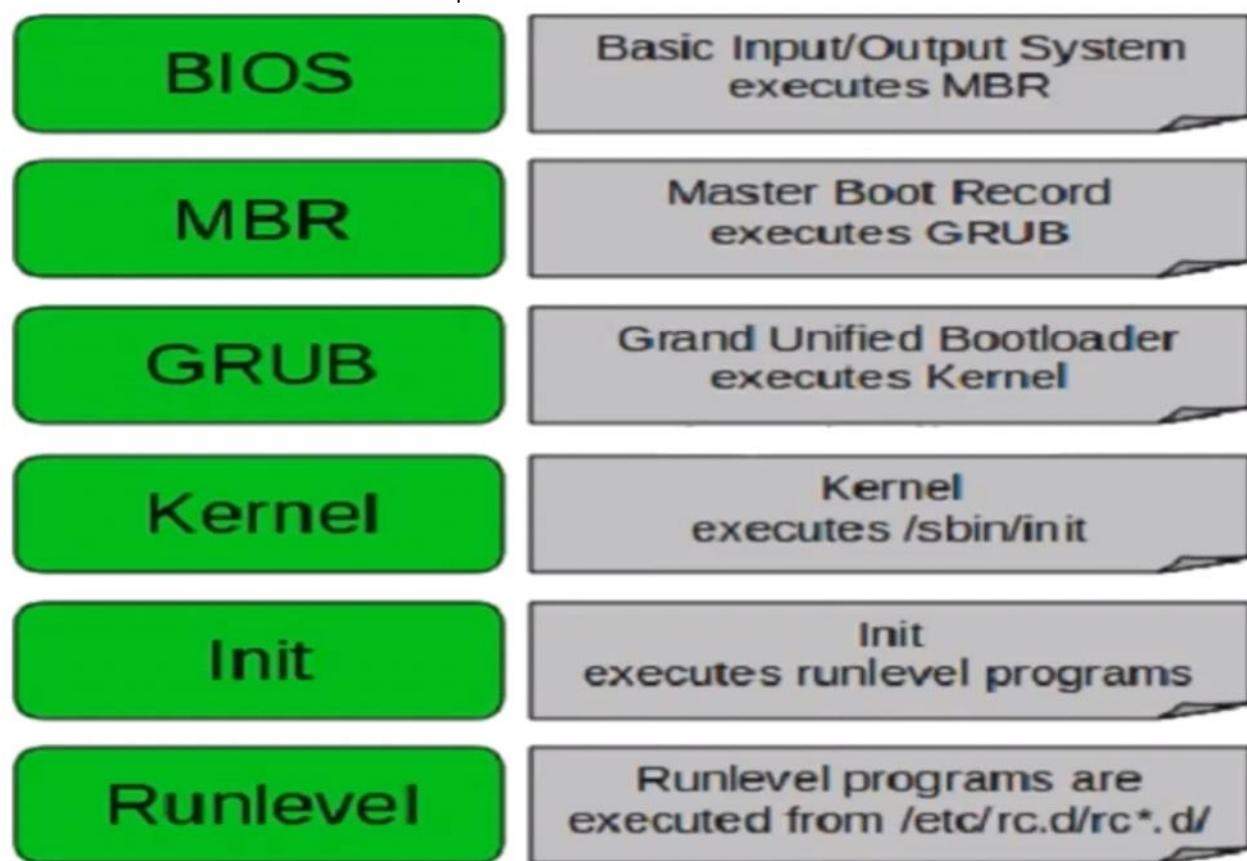
The bootstrap process means how your computer powers on and starts up

By: I

# LINUX BOOT PROCESS

jueves, 13 de enero de 2022 02:39 a.m.

This lesson we will learn how LINUX boots up.



**Newer version of Linux =  
/etc/systemd/system/default.target**

To set new default run-level run  
# **systemctl set-default [new target]**

**Older version of Linux = /etc/inittab**

NOTE: init is an executable command which runs at the time of startup. To set run-level modify:  
/etc/inittab  
init configuration files are located in: /etc/init.d

# LINUX BOOT PROCESS NEWER VERSIONS

jueves, 13 de enero de 2022 03:20 a. m.

The boot sequence changes in CentOS/Redhat 7 and above

- **systemd** is the new service manager in CentOS/RHEL 7 that manages the boot sequence
- It is a backward compatible with SysV init scripts used by previous versions of Redhat LINUX including RHEL 6
- Every system administrator needs to understand the boot process of an OS in order to troubleshoot effectively

BIOS = Basic Input and Output Setting (firmware interface)

POST = Power-On Self-Test started

**MBR = Master Boot Record**

Information saved in the first sector of a hard disk that indicates where the GRUB2 is located so it can be loaded in computer RAM

**GRUB2 = Grand Unified Boot Loader v2**

Loads Linux kernel

/boot/grub2/grub.cfg

**Kernel = Core of Operating System**

Loads required drivers from initrd.img

Starts the first OS process (systemd)

**Systemd = System Daemon (PID # 1)**

It then starts all the required processes

Reads = /etc/systemd/system/default.target to bring the system to the run-level

Total of 7 run-levels (0 thru 6)

## MESSAGE OF THE DAY

jueves, 13 de enero de 2022 03:21 a. m.

A simple message that will appear on the screen every time that you login into a LINUX machine.

- Message of the day file location
  - **/etc/motd**

To edit it you can make a vi under this file and place the message that you want to show every time that somebody log into your server.

## CUSTOMIZE MESSAGE OF THE DAY

jueves, 13 de enero de 2022 03:39 a. m.

- Once again, message of the day is the first message users will see when they login to the Linux machine
- Steps:
  - Create a new file in **/etc/profile.d/motd.sh**
  - Add desired commands in motd.sh file
  - Modify the **/etc/ssh/sshd\_config** file to edit  
**#PrintMotd yes** to **PrintMotd no**
  - Restart sshd service
    - systemctl restart sshd.service**



root@myfirstlinuxos:/etc/profile.d

```
#!/bin/bash
#
echo -e "
#####
#          Welcome to `hostname`#
#          This system is running `cat /etc/redhat-release`#
#          Kernel is `uname -r`#
#          You are logged in as `whoami`#
#####
"
"
```

# STORAGE

jueves, 13 de enero de 2022 03:39 a. m.

Is used to store computer data or information, there are various different types of storage:

- Local storage
  - RAM, HDD, SSD, etc
- DAS (Direct Attached Storage)
  - CD/DVD, USB flash drive, external disk directly attached with USB or other cables
- SAN (Storage Area Network)
  - Storage attached through iSCSI or fiber cable
- NAS (Network Attached Storage)
  - Storage attached over network (TCP/IP)
  - E.g. Samba, NFS, etc.

# DISK PARTITION

jueves, 13 de enero de 2022 04:03 a. m.

In this lesson we will learn a couple commands that are used to show which file system has been partitioned, where is the partition and what is your local filesystem disk size:

- df gives you how much disk is used, where is partitioned to
- fdisk tells you how much the disk total size is and how many partitions have been created out of the disk

```
[root@myfirstlinuxos ~]# df
Filesystem      1K-blocks   Used   Available  Use% Mounted on
devtmpfs          490524     0    490524   0% /dev
tmpfs            507372    84    507288   1% /dev/shm
tmpfs            507372  14212    493160   3% /run
tmpfs            507372     0    507372   0% /sys/fs/cgroup
/dev/mapper/centos-root  8374272 5056244   3318028  61% /
/dev/sdal        1038336  216596   821740  21% /boot
tmpfs            101476     40    101436   1% /run/user/1000
tmpfs            101476     0    101476   0% /run/user/0
[root@myfirstlinuxos ~]# fdisk -l

Disk /dev/sda: 10.7 GB, 10737418240 bytes, 20971520 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000257d3

      Device Boot      Start        End    Blocks   Id  System
/dev/sdal   *       2048    2099199   1048576   83  Linux
/dev/sda2        2099200   20971519    9436160   8e  Linux LVM

Disk /dev/mapper/centos-root: 8585 MB, 8585740288 bytes, 16769024 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/centos-swap: 1073 MB, 1073741824 bytes, 2097152 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

[root@myfirstlinuxos ~]#
```

# ADDING DISK AND CREATE A STANDARD PARTITION

jueves, 13 de enero de 2022 04:06 a. m.

Is important to make an snapshot if something goes wrong adding a new disk you will be unable to boot the partition.

- Turn off the VM
- Take a snapshot
- Go to settings->Storage-> and then hit add a new disk, click on create a new disk->VDI-> Dynamically allocated->Set the space that you want and click to create
- Start VM
- Become root in console run **df -h** command and then fdisk -l | more
- Locate the name of the new disk and execute the command: fdisk /dev/nameOfDisk
- Execute help command
- Hit n that means add a new partition
- Partition type: primary and every let it as default
- Hit **w**
- After completion of process we have to create a filesystem executing the command: **mkfs.xfs /dev/sdb1**

```
Disk /dev/sdb: 2147 MB, 2147483648 bytes, 4194304 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0xdddd0eed

      Device Boot      Start        End      Blocks   Id  System
/dev/sdb1            2048     4194303     2096128   83  Linux
```

- Then we have to mount it, to mount we can create a folder inside the primary hard disk (for example I create data directory) and then we have to execute the command: **mount /dev/sdb1 /data**
- Just remember that every time that your system boots up we have to consider that the system will not mount automatically, to force to system to mount it automatically we have to add the entry into: /etc/fstab doing vi inside this file: **vi /etc/fstab**
- And in the end of the file we have to add: **/dev/sdb1 /data xfs defaults 0 0**
- Those two 0 means that when your system are booting up will run the check process

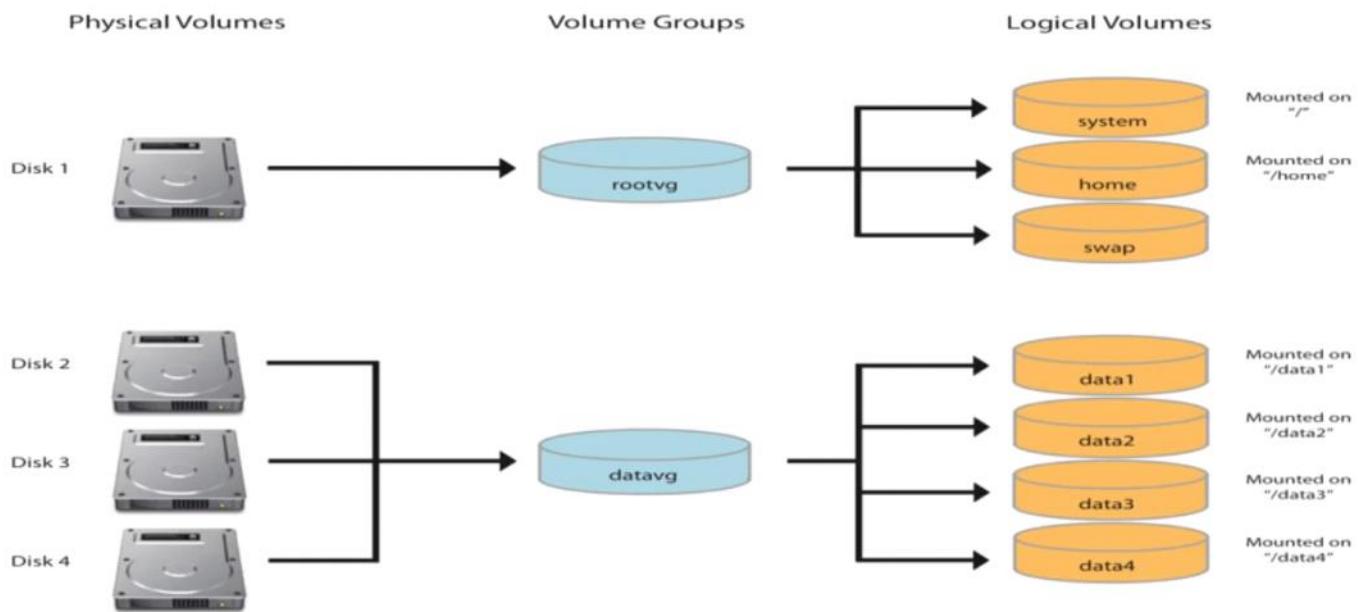
```
File Edit View Search Terminal Help
#
#
# /etc/fstab
# Created by anaconda on Fri Nov 26 00:41:24 2021
#
• # Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/centos-root /          xfs    defaults    0 0
UUID=e4951499-f9a7-4463-b8a9-b5f81c410ccb /boot          xfs    defaults    0 0
/dev/mapper/centos-swap swap      swap    defaults    0 0
/dev/sdb1      /data    xfs    defaults    0 0
~
```

- If everything goes well and you reboot your system and there's no any problem the device will be mounted. You can check that device is mounted executing: **df -h**
- To unmount a device you have to become root and execute: **umount /data**
- To mount it again you can use **mount -a**

# LOGICAL VOLUME MANAGEMENT (LVM)

jueves, 13 de enero de 2022 04:07 a. m.

LVM allows disk to be combined together:

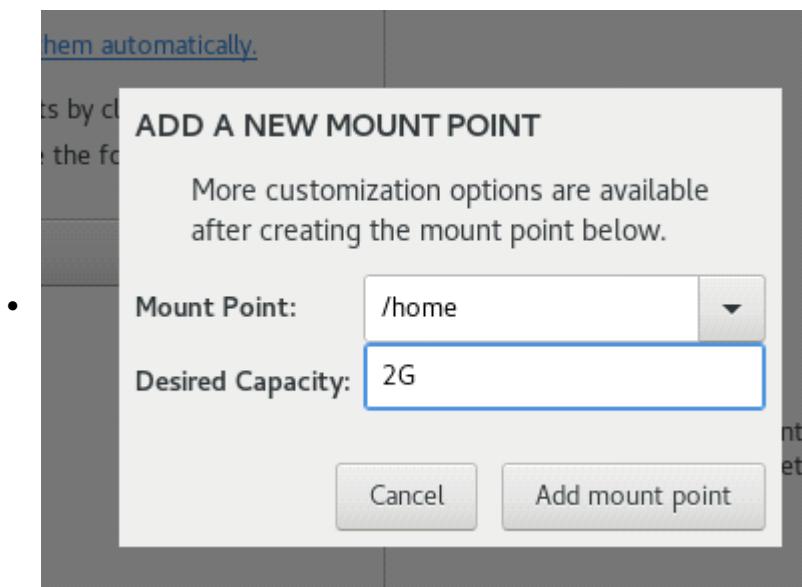
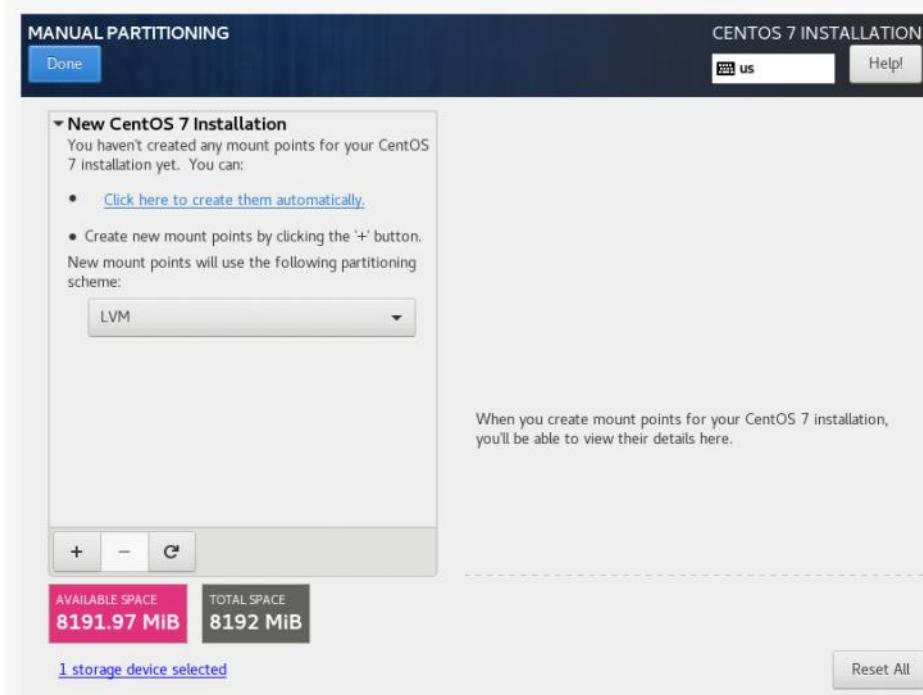


The main advantage of use LVM is that you can add a disk without break a partition or the filesystem, you can add or extend the group.

# LVM CONFIGURATION DURING INSTALLATION

viernes, 14 de enero de 2022 12:31 a. m.

- Install LINUX CentOS with LVM configuration



## MANUAL PARTITIONING

Done

### ▼ New CentOS 7 Installation

#### DATA

/home  
centos\_hostlvm-home

2048 MiB

#### SYSTEM

/boot  
sdal  
/  
centos\_hostlvm-root  
/var  
centos\_hostlvm-var  
swap  
centos\_hostlvm-swap

1024 MiB >

2044 MiB

2048 MiB

1024 MiB

+ - G

sda1

Mount Point:

/boot

Desired Capacity:

1024 MiB

Device Type:

Standard ... ▾

Encrypt

File System:

xfs ▾

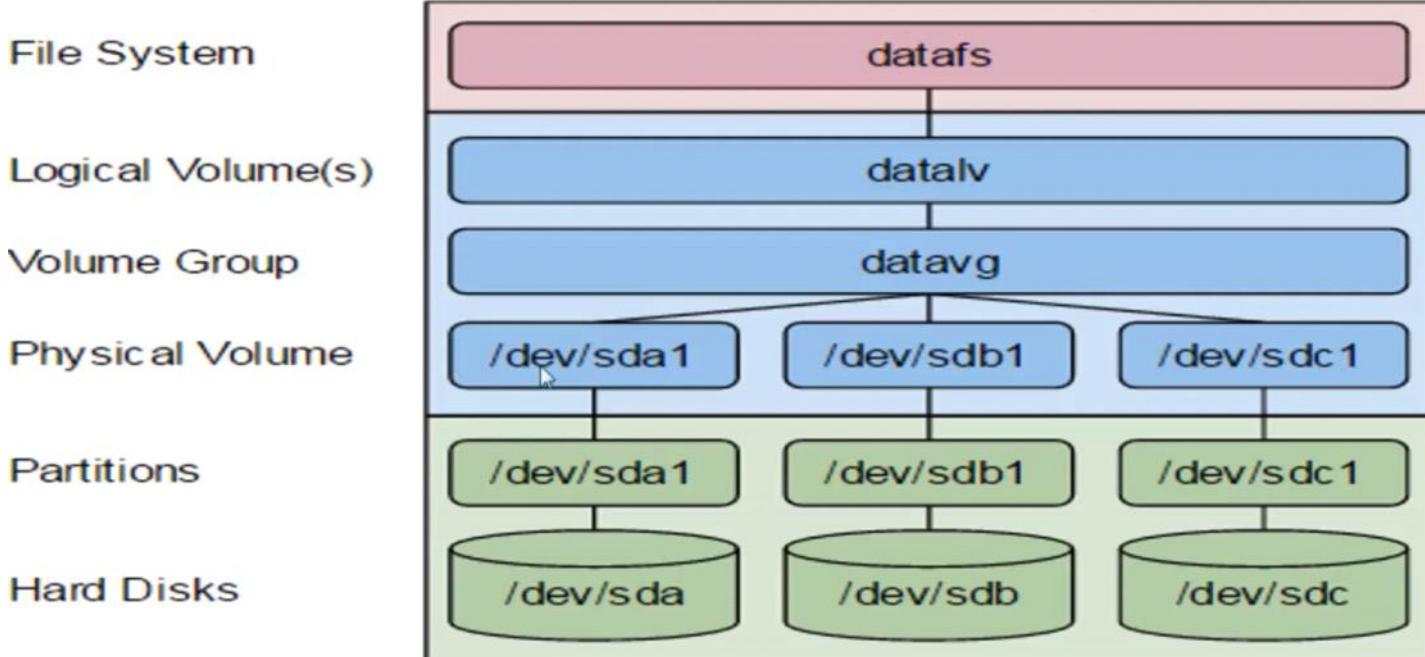
Reformat

Label:

## ADD DISK AND CREATE NEW LVM PARTITION

### (pvcreate,vgcreate,lvcreate)

viernes, 14 de enero de 2022 12:56 a. m.



- We have to execute `fdisk -l` command
- Execute the same commands when you create a partition and change a partition system id hitting `t` key and then placing LVM as `8e` in HEX code
- To create a physical volume you have to execute:  

```
[root@myfirstlinuxos ~]# pvcreate /dev/sdb1
Physical volume "/dev/sdb1" successfully created.
```
- [root@myfirstlinuxos ~]# pvdisplay  
--- Physical volume ---  
PV Name /dev/sda2
- To create a volume group you have to execute:  

```
[root@myfirstlinuxos ~]# vgcreate oracle_lv
No command with matching syntax recognised. Run 'vgcreat
Correct command syntax is:
vgcreate VG_new PV ...
```
- [root@myfirstlinuxos ~]# vgcreate oracle\_lv /dev/sdb1
Volume group "oracle\_lv" successfully created
[root@myfirstlinuxos ~]# vgdisplay oracle\_lv
- To create logical volumes you have to  

```
[root@myfirstlinuxos ~]# lvcreate -n oracle_lv --size 1000M oracle_lv
```
- Logical volume "oracle\_lv" created.
[root@myfirstlinuxos ~]#
- To mount this you have to execute:  

```
[root@myfirstlinuxos ~]# mkfs.xfs /dev/oracle_lv/oracle_lv
meta-data=/dev/oracle_lv/oracle_lv isize=512    agcount=4, agsize=64000 blks
          =                      sectsz=512  attr=2, projid32bit=1
          =                      crc=1      finobt=0, sparse=0
data     =                      bsize=4096   blocks=256000, imaxpct=25
          =                      sunit=0    swidth=0 blks
          naming    =version 2
          log       =internal log
          realtime =none
          extsz=4096   blocks=0, rtextents=0
[root@myfirstlinuxos ~]# mkdir /oracle
[root@myfirstlinuxos ~]# mount /dev/oracle_lv/oracle_lv /oracle/
```
- REMEMBER THAT IF YOU WANT TO MOUNT THIS PARTITION AT BOOT TIME THEN EDIT THE

/etc/fstab FILE AND ADD THE FOLLOWING LINE AT THE BOTTOM OF THE FILE

o /dev/oracle\_lv                /oracle                xfs                defaults    0                0

# EXTEND DISK USING LVM

viernes, 14 de enero de 2022 01:45 a. m.

/oracle = 1.0G

/oracle = Full

## Few Options:

- Delete older files to free up disk space
- Add new physical disk mount to /oracle2
- Create a new virtual disk and mount to /oracle2
- Or extend /oracle through LVM.

To extend a disk we have to:

- Make sure that VM is turned off
- Go to storage in settings and create a new VDI as we know
- Become root
- Execute df -h
- Execute fdisk -l and see the new device
- Execute fdisk /dev/nameOfNewDevice, hit n every in default configuration and then hit t to change partition system id
- Reboot the system
- Become root again
- Execute fdisk -l /dev/nameOfDevice
- Execute pvdisplay
- Execute vgdisplay oracle\_vg
- Execute pvcreate /dev/nameOfDevice
- Execute vgextend oracle\_lv /dev/nameOfDevice
- Execute lvextend -L+1024M /dev/mapper/oracle\_vg\_lv
- Execute xfs\_growfs /dev/oracle\_vg/oracler\_lv

# ADDING SWAP SPACE

viernes, 14 de enero de 2022 02:29 a. m.

SWAP space in LINUX is used when the amount of physical memory (RAM) is full. If the system needs more memory resources and the RAM is full, inactive pages in memory are moved to the swap space. While swap space can help machines with small amount of RAM, it should not be considered a replacement for more RAM. SWAP space is located on hard drives, which have a slower access time than physical memory

- Recommended swap size = Twice the size of RAM
  - M= Amount of RAM in GB, S= Amount of swap in GB, then
    - IF  $M < 2$ 
      - Then  $S = M * 2$
      - Else  $S = M + 2$
  - Commands:
    - dd
    - mkswap
    - swapon or swapoff
- To see the available swap we can make:

```
[root@myfirstlinuxos ~]# free -m
              total        used        free      shared  buff/cache   available
Mem:       990         627         64          17        298        205
Swap:     1023           5        1018
[root@myfirstlinuxos ~]#
```

- To add more swap space we have various options:

- Add new DIM if we have a physical space
- Add available space from disk:
  - Become root
  - Use dd, dd is a command to create a new file

```
swap:          1023           5        1018
[root@myfirstlinuxos ~]# dd if=/dev/zero of=/newSwap bs=1M count=1024
```

- Change permission to the new file executing: chmod go-r newSwap
- To add new swap execute mkswap /newSwap
- To add it directly to swap then execute swapon /newSwap

```
[root@myfirstlinuxos ~]# swapon /newSwap
[root@myfirstlinuxos ~]# free -m
```

```
              total        used        free      shared  buff/cache   available
Mem:       990         606         67          17        317        226
Swap:     2047           5        2042
[root@myfirstlinuxos ~]#
```

- If you want to have this swap space be enabled during boot time you have to specify in the file: /etc/fstab
  - ◆ Add at the bottom of the file:
    - ◊ /newSwap swap swap defaults 0
- If you want to delete swap space you can make:
  - Execute swapoff /newSwap

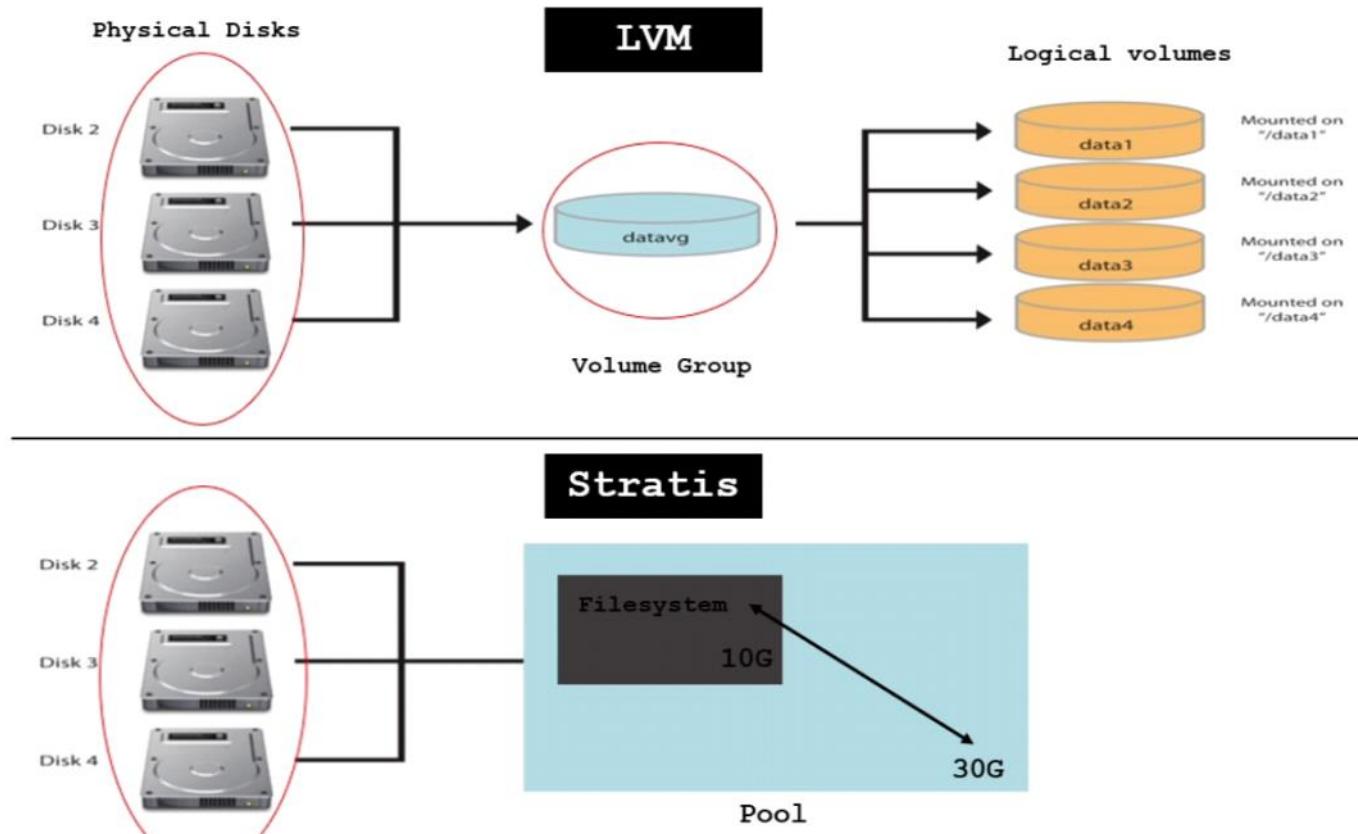
# ADVANCE STORAGE MANAGEMENT WITH STRATIS

viernes, 14 de enero de 2022 02:53 a. m.

- RedHat 8 introduces the next generation volume management solutions called Stratis
- It uses thin provisioning by default
- It's available in CentOS 7.5
- It combines the process of creating logical volume management (LVM) and creation of filesystems into one management
- In LVM if a filesystem gets full you will have to extend it manually whereas stratis extends the filesystem automatically if it has available space in its pool

In this lesson we will learn:

- How to manage multiple storage layers using Stratis local storage management



Implement stratis:

- Install stratis package:
  - Execute yum/dnf install stratis-cli stratisd
- Enable and start stratis service
  - Execute systemctl enable|start stratisd
- Add 2 x 5G new disk from virtualization software and verify at the OS level
  - Execute oracle virtualbox storage setting
  - Execute lsblk
- Create a new stratis pool and verify
  - Execute stratis pool create pool1 /dev/sdb
  - Execute stratis pool list
- Extend the pool
  - Execute stratis pool add-data pool1 /dev/sdc
  - Execute stratis pool list

- Create a new filesystem using stratis

```
stratis filesystem create pool1 fs1
stratis filesystem list          (Filesystem will start with 546 MB)
```
- Create a directory for mount point and mount filesystem

```
mkdir /bigdata
mount /stratis/pool1/fs1 /bigdata
lsblk
```
- Create a snapshot of your filesystem

```
startis filesystem snapshot pool1 fs1 fs1-snap
stratis filesystem list
```
- Add the entry to /etc/fstab to mount at boot

```
UUID="af887afgdja-" /fs1 xfs defaults,x-
systemd.requires=stratisd.service 0 0
```

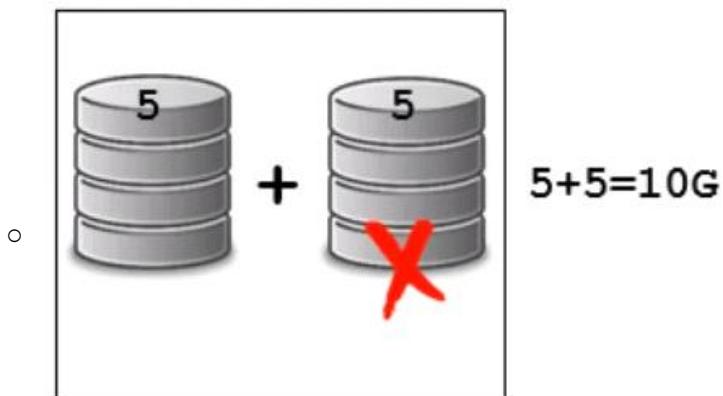
# RAID

viernes, 14 de enero de 2022 03:51 a. m.

RAID stands for Redundant Array of Independent Disks

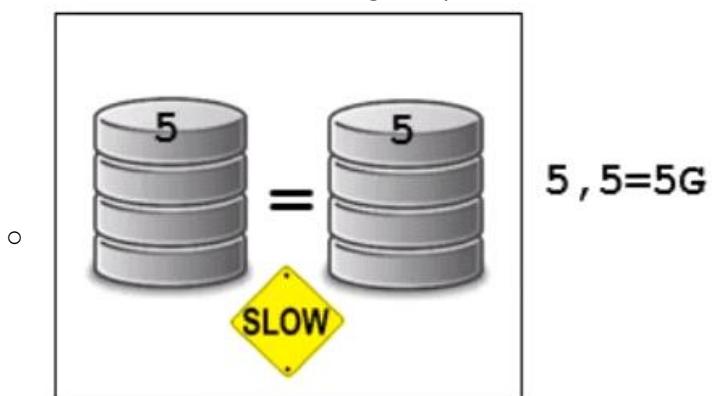
Type of RAID:

- RAID0, is mostly used for data warehouse



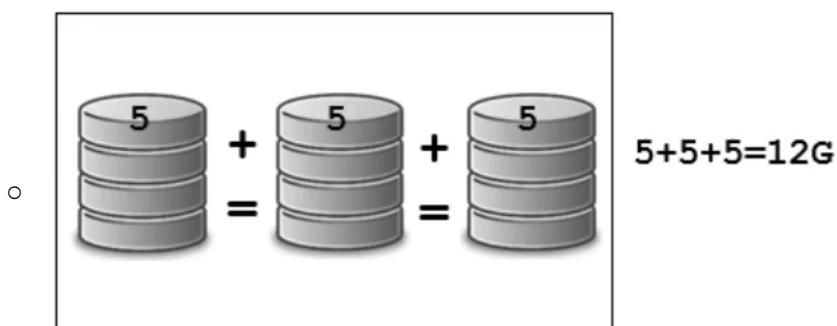
## RAID0

- To make two disk into one and increase capacity
- RAID1, you have one disk and then another this of the same capacity in this case is like if you are marring the disks, this means that if you loose one disk you have the other disk, it is slow because the information have to get duplicated



## RAID1

- RAID5, this means that it will write the first disk and write part of the information on the others disks, this means that if you lose the first disk and the information you can retrieve it because the information is on the others disks but you will not get the add of capacities of the disk that you have, you decrease the capacity because the RAID needs to reserve capacity to save part of the information.



## RAID5

- RAID is mostly configured in physical disks

# FILE SYSTEM CHECK (fsck and xfs\_repair)

viernes, 14 de enero de 2022 04:00 a. m.

This commands are used to repair filesystem:

- fsck command
- xfs\_repair is used to repair filesystems that I build on xfs filetype

- Linux fsck utility is used to check and repair Linux filesystems (ext2, ext3, ext4, etc.)
- Linux xfs\_repair utility is used to check and repair Linux filesystems for xfs filesystem type
- Depending on when was the last time a file system was checked, the system runs the fsck during boot time to check whether the filesystem is in consistent state
- System administrator could also run it manually when there is a problem with the filesystems
- Make sure to execute the fsck on an **unmounted** file systems to avoid any data corruption issues.

If you need to unmount on root directory you cant, you have to set the system in single user mod

The difference between fsck and xfs\_repair is that xfs filesystem was introduced for larger partitions TB, PB, we dont run this utility at boot time is because if the storage is too large you dont want to wait until this tool end to check this filesystem, if you have issues you have to run it manually.

- Force a filesystem check even if it's clean using option -f
- Attempt to fix detected problems automatically using option -y
- The xfs\_repair utility is highly scalable and is designed to repair even very large file systems with many inodes efficiently. Unlike other Linux file systems, xfs\_repair does not run at boot time
- The following are the possible exit codes for fsck command
  - 0 – No errors
  - 1 – Filesystem errors corrected
  - 2 – System should be rebooted
  - 4 – Filesystem errors left uncorrected
  - 8 – Operational error
  - 16 – Usage or syntax error
  - 32 – Fsck canceled by user request
  - 128 – Shared-library error

To know the exit code you have to execute **echo \$?**

How to decide when to use one or other utility, well that's depend on how filesystem type, to know the filetype that you have:

- Execute **df -T**

## SYSTEM BACKUP (dd COMMAND)

viernes, 14 de enero de 2022 04:28 a. m.

There are many different ways that you can do system backup:

There are 5 types of backups:

- System backup (entire image using tools such as acronis, Veeam, Commvault, etc.)
  - Application backup (third party application backup solution)
  - Database backup (Oracle dataguard, SQL backup, etc.)
  - Filesystem backup (tar, gzip directories, etc.)
  - Disk backup or disk cloning (dd command), dd command does not know what is inside a disk, this make the entire back up of entire disk
- 
- dd is a command-line utility for UNIX and UNIX-LIKE operating systems whose primary purpose is to convert and copy files
  - As a result, dd can be used for tasks such as backing up the boot sector of a hard drive, and obtaining a fixed amount of random data
  - Please note the source and destination disk should be the same size
  - To backup or clone an entire hard disk to another hard disk connected to the same system, execute the dd command as shown

```
# dd if=<source file name> of=<target file name> [Options]
# dd if=/dev/sda of=/dev/sdb
```

The second command is copying sda on sdb

To backup/copy the disk partition:

```
# dd if=/dev/sda1 of=/root/sda1.img
```

Restoring this image file to other machine after copying the .img

```
# dd if=/root/sda1.img of=/dev/sdb3
```

Please note that copy the partition to partition we will need: to execute

```
/dev/sda1              1014M  157M  858M  16% /boot
/dev/mapper/oracle_vg-oracle_lv  2.0G   33M  2.0G   2% /oracle
tmpfs                  100M   44K  100M   1% /run/user/1000
/dev/sdb1              2.0G   33M  2.0G   2% /data
[root@MyFirstLinuxVM ~]# dd if=/dev/sda1 /dev/sdb1
```

If we want to copy to partition to filesystem

```
/dev/sda1              1014M  157M  858M  16% /boot
/dev/mapper/oracle_vg-oracle_lv  2.0G   33M  2.0G   2% /oracle
tmpfs                  100M   44K  100M   1% /run/user/1000
/dev/sdb1              2.0G   33M  2.0G   2% /data
[root@MyFirstLinuxVM ~]# dd if=/dev/sda1 /data/boot.img
dd: unrecognized operand '/data/boot.img'
Try 'dd --help' for more information.
[root@MyFirstLinuxVM ~]#
[root@MyFirstLinuxVM ~]# dd if=/dev/sda1 of=/data/boot.img
```

To restore the image you have to:

```
[root@MyFirstLinuxVM ~]# dd if=/data/boot.img /dev/sda1
```

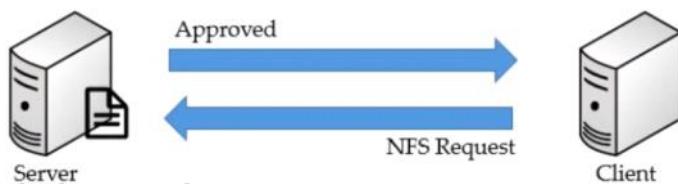
# NETWORK FILESYSTEM (NFS)

viernes, 14 de enero de 2022 11:57 a. m.

This is one of the biggest and majors topics, it is actually about sharing your disk or directories or files.

This is another filesystem, it is not local is called NAT (NETWORK ATTACHED STORAGE). NFS is part of NAS

- NFS stands for Network File System, a file system developed by Sun Microsystems, Inc.
- It is a client server system that allows you to access files across a network and treat them as if they resided in a local file directory
- For example, if you were using a computer linked to a second computer via NFS, you could access files on the second computer as if they resided in a directory on the first computer. This is accomplished through the processes of exporting (the process by which an NFS server provides remote clients with access to its files) and mounting (the process by which client map NFS shared filesystem)



## Steps for NFS Server Configuration

- Install NFS packages  

```
# yum install nfs-utils libnfsidmap      (most likely they are installed)
```
- Once the packages are installed, enable and start NFS services  

```
# systemctl enable rpcbind
# systemctl enable nfs-server
# systemctl start rpcbind, nfs-server, rpc-statd, nfs-idmapd
```
- Create NFS share directory and assign permissions  

```
# mkdir /mypretzels
# chmod a+rwx /mypretzels
```

Read/write  
all changes to the according filesystem are immediately flushed to disk; the respective write operations are being waited for
- Modify `/etc/exports` file to add new shared filesystem  

```
# /mypretzels 192.168.12.7 (rw,sync,no_root_squash) = for only 1 host
# /mylinuxshare *(rw,sync,no_root_squash) = for everyone
```

IP address of client machine  
root on the client machine will have the same level of access to the files on the system as root on the server.

- Then export NFS filesystem:
  - `exportfs -rv`

## Steps for NFS Client Configuration

- Install NFS packages

```
# yum install nfs-utils rpcbind
```
- Once the packages are installed enable and start rpcbind service

```
# service rpcbind start
```
- Make sure firewalld or iptables stopped (if running)

```
# ps -ef | egrep "firewall|iptable"
```
- Show mount from the NFS server

```
# showmount -e 192.168.1.5 (NFS Server IP)
```
- Create a mount point

```
# mkdir /mnt/app
```
- Mount the NFS filesystem

```
# mount 192.168.1.5:/mypretzels /mnt/kramer
```
- Verify mounted filesystem

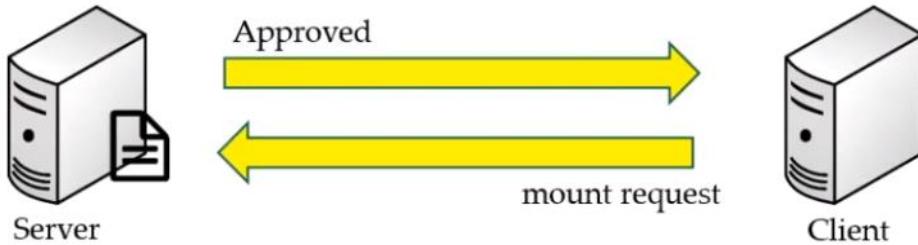
```
# df -h
```
- To umount

```
# umount /mnt/kramer
```

# SAMBA INSTALLATION OR CONFIGURATION

viernes, 14 de enero de 2022 11:25 p. m.

- Samba is a LINUX tool or utility that allows sharing for LINUX resources such as files and printers to with other operating systems
- It works exactly like NFS but the difference is NFS shares within LINUX or UNIX like system whereas SAMBA shares with other OS
- For example, computer A shares filesystem with computer B using samba then computer B will see that shared filesystem as if it is mounted as the local filesystem



- SAMBA shares its filesystem through a protocol called SMB (Server Message Block) which was invented by IBM
- Another protocol used to share SAMBA is through CIFS(Common Internet Filesystem) invented by Microsoft and NMB (NetBios Named Server)
- CIFS became the extension of SMB and now Microsoft has introduced newer versions of SMB v2 and v3 that are mostly used in the industry
- In simple term, most people, when they use either SMB or CIFS, are talking about the same exact thing

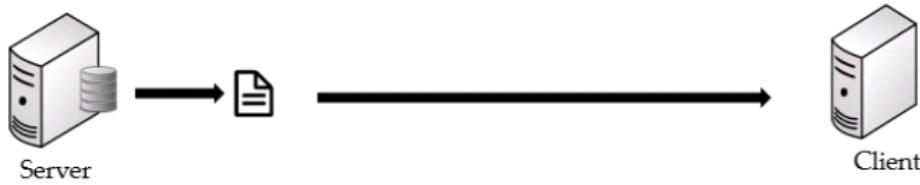
## STEPS TO ENABLE INSTALL AND CONFIGURE SAMBA

- Take snapshot of your VM
- Install samba packages
- Enable samba to be allowed through firewall (Only if you have firewall running)
- Disable firewall
- Create Samba share directory and assign permissions
- Also change the SELinux security context for the samba shared directory
- Or disable SELinux
- Modify `/etc/samba/smb.conf` file to add new shared filesystem
- Verify the setting
- Once the packages are installed, enable and start **Samba** services (smb and nmb)
- Mount Samba share on Windows client
- Mount Samba share on Linux client
- Additional instructions on creating secure Samba share.

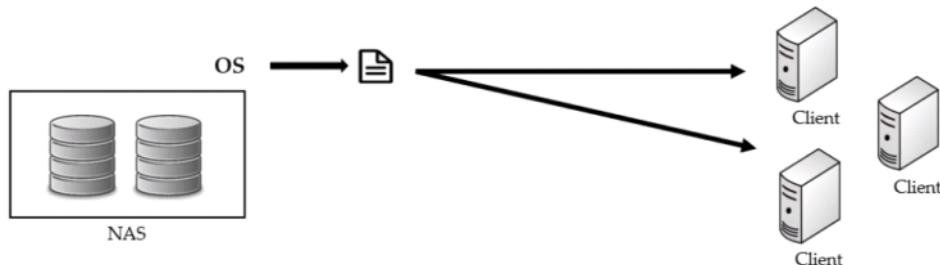
# NAS DRIVE FOR NFS OR SAMBA

sábado, 15 de enero de 2022 12:55 a. m.

- A storage can be carved on a LINUX server, and it can be shared with another LINUX machine through NFS or to a Windows machine through Samba service.



- NFS/Samba or any NAS service can be setup through a dedicated NAS service.

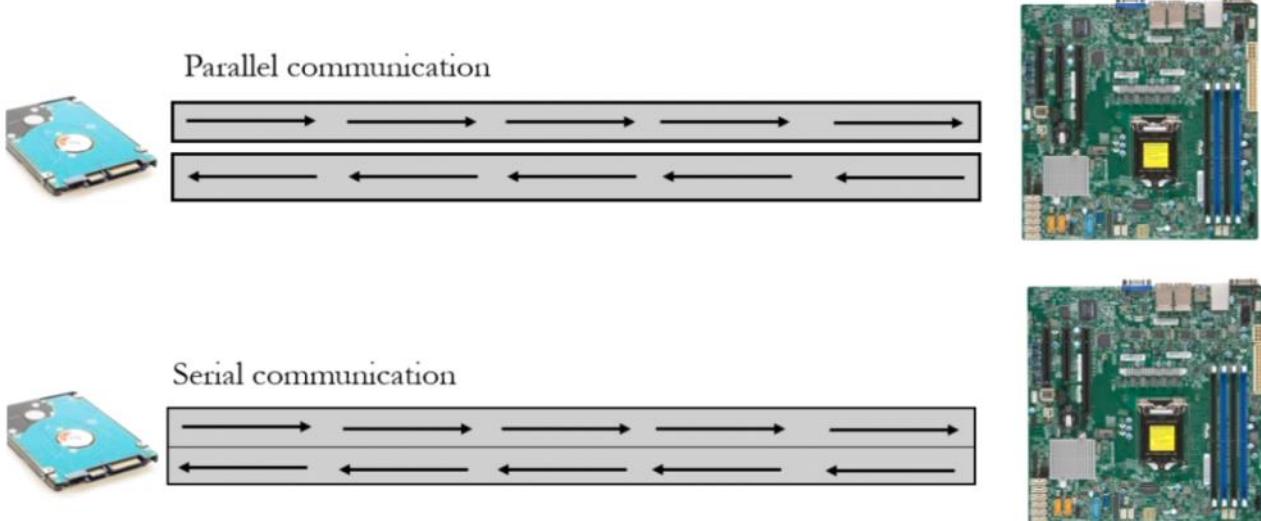


- In this lesson we will learn:
  - Physical layout of a NAS device
  - Setup, configure and manage NAS device
  - Create shared filesystem(NFS and SAMBA)
  - Mount shared folder from the NAS device to LINUX and WINDOWS

# SATA AND SAS

sábado, 15 de enero de 2022 12:55 a. m.

- **SATA** Stands for Serial Advanced Technology Attachment and **SAS** stands for Serial Attached SCSI (SCSI Stands for Small Computer System Interface, typically pronounced as “scuzzy”)
- Both SAS and SATA utilize serial communication. Serial communication means that the highway has both lanes



- The main difference between them is that SAS drives are faster and more reliable than SATA drives
- SAS is generally more expensive, and it's better suited for use in servers or in processing-heavy computer workstations. SATA is less expensive, and it's better suited for desktop file storage
- In a SATA cable, all 4 wires are placed within the same cable. IN SAS cable, the 4 wires are separated into two different cables

Why divide the two wires between 2 cables?

- So you can connect more devices to one another. With SATA cable, you can only link the motherboard and the storage drive. You could hook up an expansion device, but that takes up valuable room inside your computer.
- With SAS cable, you can hook up the motherboard to both a storage drive and another piece of hardware that has SAS connectors.

Here is what the highway look like, metaphorically:

- SATA CABLE: Los Angeles to San Francisco
- SAS CABLE: Los Angeles to San Francisco or Los Angeles

# DIFFERENCE BETWEEN CentOS/RedHat 5, 6 or 7

sábado, 15 de enero de 2022 12:55 a. m.

CentOS and RHEL difference

- CentOS and RHEL difference, both are exactly the same, RHEL provides technical support, CentOS does not provide technical support
- Total versions
- Version 5,6 and 7

SEE CHEATSHEET.

# DIFFERENCE BETWEEN CentOS/ RHEL 7 and 8

sábado, 15 de enero de 2022 01:56 a. m.

- RedHat enterprise LINUX 8 (RHEL 8) is now available for production use with lots of developer-friendly capabilities
- RHEL 8 official release by Red Hat Inc, was announced on May, 2019

	RHEL 8	RHEL 7
General Availability Date	14-Nov-18	10-Jun-14
Code Name	Ootpa	Maipo
Kernel Version	4.18	3.10.0-123
End of Support	May-2029	30-Jun-2024
Last Minor Release	8.x	7.7
Network Time Synchronization	Only Chrony	Chrony and ntpd
GUI Interface (Desktop)	Gnome 3.28	Gnome 3
Default Database	MySQL 8.0, MariaDB 10.3, PostgreSQL 10 and 9.6, and Redis 5.0	MariaDB

	RHEL 8	RHEL 7
Default Firewall	Firewalld, it uses nftables framework in the backend	Firewalld, it uses Iptables framework in the backend
Max Supported (Individual) File & Filesystem Size	XFS= 1024TB	XFS= 500TB
Package Management	By default both are installed, YUM symbolic link to DNF	By default only YUM and DNF can be installed from the Extra repo
Max. RAM Supported	24 TB on x86_64 architecture	12 TB on x86_64 architecture

# SUMMARY

sábado, 15 de enero de 2022 02:01 a. m.

- There are 7 run levels in LINUX OS

What is the run level for system bootup with networking but without GUI?

- 3

What are the 3 main commands to transfer files from one computer to another?

- ftp, scp and rsync

Which command is used to partition a disk in Linux?

- fdisk

Which of the following storage is attached through fiber cable?

- SAN

# RESUME WORKSHOP

sábado, 15 de enero de 2022 02:29 a. m.

- How to build you impressive resume
- Do's and Don'ts
- Adding keywords
- Structure

CHECK HANDOUTS

# COVER LETTER

sábado, 15 de enero de 2022 02:52 a. m.

Is very important when you are sending a resume.

There are two types:

- Cover letter - Entry level
- Cover letter - Experienced letter

SEE HANDOUTS

# POST RESUME AND WHAT TO EXPECT

sábado, 15 de enero de 2022 02:53 a. m.

When you post your resume:

- Recruiters or Companies Human Resources, recruiters are head hunters, they search for:
  - Search for the right candidates
  - Match the education requirement and experience
- The initial activities that they perform are:
  - Interview setup
  - Expectation
  - Rate negotiation

HR:

- They reach you out by telephone or email

After that recruiters call you follows the next step: Job description:

- Recruiters or Companies' Human Resources
- Job description
- Full time or Consultant

## **Full time:**

- Work directly to the hiring company
- Get all the benefits
- Paid on salary basis
- Eligible for vacation and sick time

## **Consultant:**

- Works on per hour basis
- Has no benefits
- Independent contractor

# **USA Tax Forms:**

**- W2**

**- 1099**

When you comes into rate negotiation part:

- Recruiters or Companies' Human Resources
- Job description
- Full time or Consultant
- Rate negotiation

Tell that you are flexible and willing to negotiate

After that:

- Recruiters or Companies' Human Resources
- Job description
- Full time or Consultant
- Rate negotiation
- Updated resume with rate confirmation

**- Full legal name  
- D.O.B = MM/DD  
- Last four SS# or  
your national ID #**

- Recruiters or Companies' Human Resources
- Job description
- Full time or Consultant
- Rate negotiation
- Updated resume with rate confirmation
- Quick interview with the recruiter's manager over the phone

- Recruiters or Companies' Human Resources
  - Job description
  - Full time or Consultant
  - Rate negotiation
  - Updated resume with rate confirmation
  - Quick interview with the recruiter's manager over the phone
  - Meet the recruiter in person
- 
- Recruiters or Companies' Human Resources
  - Job description
  - Full time or Consultant
  - Rate negotiation
  - Updated resume with rate confirmation
  - Quick interview with the recruiter's manager over the phone
  - Meet the recruiter in person
  - Resume submission
- 
- Interview schedule
- 
- First phone interview
  - Second in-person interview
  - Offer letter
- 
- Acceptance and start date.

# LINUX ADMINISTRATOR JOB DUTIES AND JOB TITLES

sábado, 15 de enero de 2022 02:53 a. m.

You can get:

- IT technician
- Help desk technician
- System administrator is an administrator who manages the system and system is comprised of:
  - HW
  - SW
  - OS
  - Networking devices
  - Cables
- Systems Engineer
  - Designing your entire infrastructure

SEE HANDOUTS

# EXPOSURE TO OTHER TECHNOLOGIES

sábado, 15 de enero de 2022 02:53 a. m.

- Operating system
  - Windows, Virtualization technology
- HARDWARE
  - Dell, HP, Cisco about servers
- MONITORING TOOLS
  - Nagios, Splunk, Zenoss
- Cloud
  - Amazon cloud EC2 (Elastic Cloud 2)
- OS Tools
  - Redhat Satellite, Kickstart, Active Directory, DNS, Puppet, etc
- DATABASE
  - SQL, Informix

CHECK HANDOUTS