



INTRODUCTION TO LINUX

Bin Jiang

11/26/2017

Linux Distribution

Linux distribution (often abbreviated as distro) is an operating system made from a software collection.

The software packages are available online in so-called repositories, which are storage locations usually distributed around the world

Linux Distribution

- A Linux kernel
- GNU tools
- Libraries
- Additional software
- Documentation
- A window system (X Window System)
- A window manager
- A desktop environment
- Distribution installers (Debian-Installer and Anaconda)
- Package management systems

Widely used distributions

Debian - Non-commercial distribution and one of the earliest, maintained by a volunteer developer community

- Ubuntu, a desktop and server distribution derived from Debian
- Linux Mint, a distribution based on and compatible with Ubuntu

Widely used distributions

Fedora - Community distribution sponsored by Red Hat.

- Red Hat Enterprise Linux (RHEL) - Derivative of Fedora
- CentOS - A distribution derived from the same sources used by Red Hat

openSUSE - Community distribution mainly sponsored by German company SUSE

How To Install Software In Linux

- APT For Debian based distributions, like, Ubuntu, Linux Mint etc.
 - The APT is the tool, commonly used to install packages, remotely from the software repository
 - The most frequently used commands are update and install

How To Install Software In Linux

```
BigDataTraining@LAPTOP-GKHUKEMA:/$ apt-get
apt 1.0.1ubuntu2 for amd64 compiled on Dec  8 2016 16:23:38
Usage: apt-get [options] command
       apt-get [options] install|remove pkg1 [pkg2 ...]
       apt-get [options] source pkg1 [pkg2 ...]
```

apt-get is a simple command line interface for downloading and installing packages. The most frequently used commands are update and install.

Commands:

- update - Retrieve new lists of packages
- upgrade - Perform an upgrade
- install - Install new packages (pkg is libc6 not libc6.deb)
- remove - Remove packages
- autoremove - Remove automatically all unused packages
- purge - Remove packages and config files
- source - Download source archives
- build-dep - Configure build-dependencies for source packages
- dist-upgrade - Distribution upgrade, see apt-get(8)
- dselect-upgrade - Follow dselect selections
- clean - Erase downloaded archive files
- autoclean - Erase old downloaded archive files
- check - Verify that there are no broken dependencies
- changelog - Download and display the changelog for the given package
- download - Download the binary package into the current directory

How To Install Software In Linux

All of the package management tools will need user to be in root or superuser, for example to install software in debian based distributions you will use apt-get followed by sudo then It will ask you to enter password.

- `sudo apt-get install conky`
- `sudo apt-get remove conky`
- `sudo apt-get update`

How To Install Software In Linux

- yum: For RPM based Linux distributions, like, Fedora, Red Hat
 - As 'apt-get' installs software packages for Debian packages, like that 'yum' installs software packages for RPM packages
 - yum does not keep a local database by default in user's hard disk. So there is no need to update it

How To Install Software In Linux

For instance:

- `yum install ${packagename}`
- `yum remove ${packagename}`
- `yum update ${packagename}`

How To Install Software In Linux

You need to give some command

Usage: yum [options] COMMAND

List of Commands:

check	Check for problems in the rpmdb
check-update	Check for available package updates
clean	Remove cached data
deplist	List a package's dependencies
distribution-synchronization	Synchronize installed packages to the latest available versions
downgrade	downgrade a package
erase	Remove a package or packages from your system
groupinfo	Display details about a package group
groupinstall	Install the packages in a group on your system
grouplist	List available package groups
groupremove	Remove the packages in a group from your system
help	Display a helpful usage message
history	Display, or use, the transaction history
info	Display details about a package or group of packages
install	Install a package or packages on your system
list	List a package or groups of packages
load-transaction	load a saved transaction from filename
makecache	Generate the metadata cache
provides	Find what package provides the given value
reinstall	reinstall a package

How To Install Software In Linux

- Tar Balls

- Linux has Tar Balls (files) ending with extensions, like, .tar, .tar.gz, .tgz, or something else. To unpack a tar ball (file), use the following command:
- `tar -xzvf ${filename}.tar.gz`
- The parameters are x to extract files, z to filter through gzip for decompression, v for verbose mode so you can tell what's going on, f indicating there will be a filename to follow

Linux Permissions

- Linux is based on Unix concept. Linux is a multi-user OS.
- It is based on unix concept of file ownership and permissions to provide security, at the file system level

Owner	Group	Other
rwx	r-x	r-x
4+2+1	4+0+1	4+0+1
7	5	5

Linux Permissions

- Users

- `cat /etc/passwd`

- Superuser

- `sudo`

- `Su`

- `cat /etc/sudoers`

- `echo "zeppelin ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers`

- Group

- `cat /etc/group`

Linux Permissions

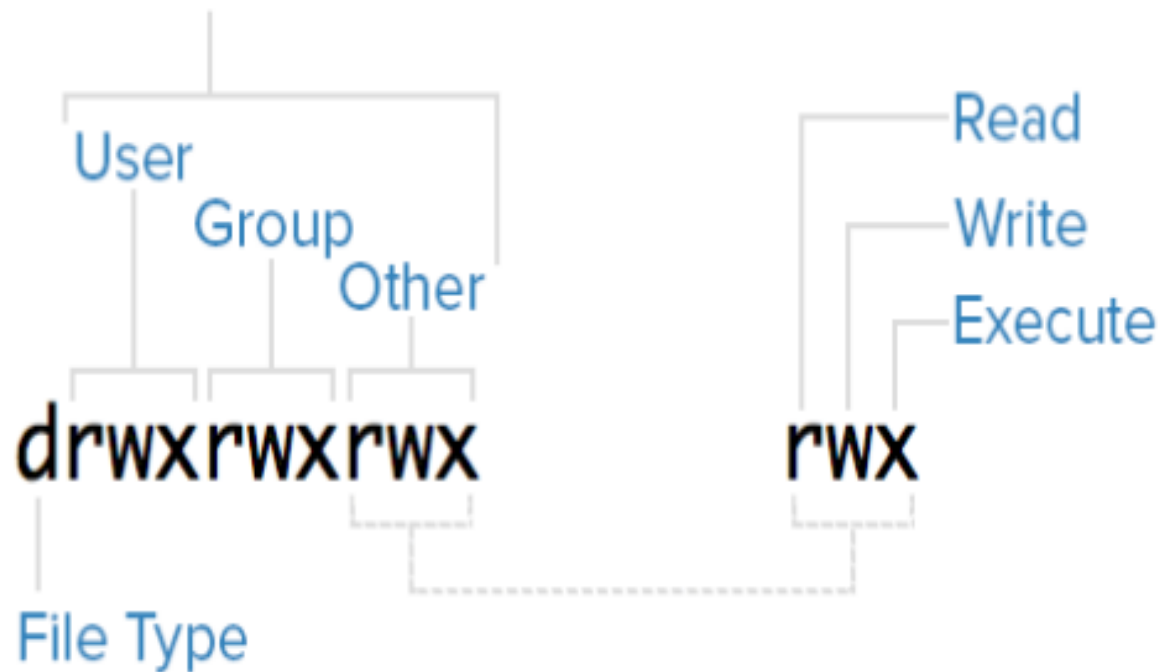
- Ownership and Permission

➤ ls -l

Mode		Owner	Group	File Size	Last Modified	Filename
drwxrwxrwx	2	sammy	sammy	4096	Nov 10 12:15	everyone_directory
drwxrwx---	2	root	developers	4096	Nov 10 12:15	group_directory
-rw-rw----	1	sammy	sammy	15	Nov 10 17:07	group_modifiable
drwx-----	2	sammy	sammy	4096	Nov 10 12:15	private_directory
-rw-----	1	sammy	sammy	269	Nov 10 16:57	private_file
-rwxr-xr-x	1	sammy	sammy	46357	Nov 10 17:07	public_executable
-rw-rw-rw-	1	sammy	sammy	2697	Nov 10 17:06	public_file
drwxr-xr-x	2	sammy	sammy	4096	Nov 10 16:49	publicly_accessible_directory
-rw-r--r--	1	sammy	sammy	7718	Nov 10 16:58	publicly_readable_file
drwx-----	2	root	root	4096	Nov 10 17:05	root_private_directory

Linux Permissions

Permissions Classes



Linux Permissions

- Examples of Modes (and Permissions)
 - ✓ `-rw-----`: A file that is only accessible by its owner
 - ✓ `-rwx-r-x-r-x`: A file that is executable by every user on the system.
 - ✓ `-rw-rw-rw-`: A file that is open to modification by every user on the system.
 - ✓ `drwxr-xr-x`: A directory that every user on the system can read and access
 - ✓ `drwxrwx---`: A directory that is modifiable (including its contents) by its owner and group
 - ✓ `drwxr-x---`: A directory that is accessible by its group

Linux Permissions

- Types of Permissions

- Alphabetic Notation

- Octal Notation

- 4 = read permissions
- 2 = write permissions
- 1 = execute permission

```
rwX rwX rwX = 111 111 111
```

```
rw- rw- rw- = 110 110 110
```

```
rwX --- --- = 111 000 000
```

```
rwX = 111 in binary = 7
```

```
rw- = 110 in binary = 6
```

```
r-x = 101 in binary = 5
```

```
r-- = 100 in binary = 4
```

Linux Permissions

- Modifying Ownership and Permissions

777	(rwxrwxrwx) No restrictions on permissions. Anybody may do anything. Generally not a desirable setting.
755	(rwxr-xr-x) The file's owner may read, write, and execute the file. All others may read and execute the file. This setting is common for programs that are used by all users.
700	(rwx-----) The file's owner may read, write, and execute the file. Nobody else has any rights. This setting is useful for programs that only the owner may use and must be kept private from others.
666	(rw-rw-rw-) All users may read and write the file.
644	(rw-r--r--) The owner may read and write a file, while all others may only read the file. A common setting for data files that everybody may read, but only the owner may change.
600	(rw-----) The owner may read and write a file. All others have no rights. A common setting for data files that the owner wants to keep private.

Linux Permissions

- Modifying Ownership and Permissions
 - `chmod 777 filename`
 - `chown username filename`
 - `chgrp groupname filename`

Linux Shell Script

- Sh—Bourne Shell
- Csh—C Shell
- Ksh—Korn Shell
- Tcsh—enhanced C Shell
- Bash—GNU Bourne Again Shell
- Zsh—extension to Bash, Ksh, and Tcsh
- Pdksh—extension to KSH

Linux Shell Script

- common
- ssh
- sudo
- curl
- wget
- vi
- zip/gzip
- sed
- awk
- edit tools

Linux Shell Script

- Working in shell

- echo \$SHELL
- bash -version
- ./hello.sh

- Various directories

`/bin/`: This contains commands used by a regular user.

`/boot/`: The files required for the operating system startup are stored here.

`/cdrom/`: When CD-ROM is mounted, the CD-ROM files are accessible here.

`/dev/`: The device driver files are stored in this folder. These device driver files will point to hardware-related programs running in kernel.

`/etc/`: This folder contains configuration files and startup scripts.

`/home/`: This folder contains a home folder of all users except the administrator

`/lib/`: The library files are stored in this folder.

`/media/`: External media such as a USB pen drive is mounted in this folder.

`/opt/`: The optional packages are installed in this folder.

`/proc/`: This contains files which give information about kernel and every process running in OS.

`/root/`: This is the administrators home folder.

`/sbin/`: This contains commands used by the administrator or root user.

`/usr/`: This contains secondary programs, libraries, and documentation about user-related programs.

`/var/`: This contains variable data such as http, tftp, and similar other.

`/sys/`: This dynamically creates the `sys` files

Linux Shell Script

- Basic Commons

- `man passwd`
- `whatis passwd`
- `which passwd`
- `whereis passwd`
- `whoami`
- `cat /proc/version`
- `cat /etc/centos-release`
- `lsb_release -a`
- `cat /etc/*release`
- `cat > file_name`
- `echo "Hello World" > hello.c`
- `echo "zeppelin ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers`
- `setuid`
- `setgid`

Linux Shell Script

- Process Basics

- `ps -ef`
- `pstree`
- `man ps`
- `ps -ef | grep "process_name"`
- `kill -9 pid_of_process_to_be_killed`
- `command &`
- `jobs`
- `fg job_number`
- `bg job_number`

Linux Shell Script

- Process Monitoring Tools

- top
- Vmstat

- Understanding "crontab"

- top
- Vmstat

- Text filtering tools

- command | more
- comman | less

Linux Shell Script

- Head and tail
 - `head -3 numbers.txt`
 - `tail -5 numbers.txt`
- The diff command
 - `diff file1 file2`
- IO redirection
 - `echo "Hello world" > log.txt`
 - `ls > log.txt`
 - `find . -name "*.sh" > success_file 2> /dev/null`
 - `find . -name "*.sh" &> log.txt`
 - `find . -name "*.sh" > log.tx 2>&1`
 - `ll /proc | grep "cpuinfo"`

Linux Shell Script

- Logical operators
 - Command1 & command2
 - Command1 && command2
 - Command1 || command2
- Pipes
 - command_1 | command_2
- Understanding variables
 - person="Ganesh Naik"
 - echo \$person
 - env
 - echo \$\$
 - export NAME
 - echo \$1

Linux Shell Script

Compressing data with gzip

- `gzip filename`
- `gunzip filename.gz`
- `gzip -l test.txt.gz`
- `cat file | gzip -c > file.gz`
- `tar -czvzf archive.tar.gz`
- `bzip2 filename`
- `bunzip2 filename.bz2`
- `tar -xjvzf archive.tar.bz2`

Linux Shell Script

Archiving and compressing with zip

- `zip file.zip file`
- `zip -r archive.zip folder1 folder2`
- `unzip file.zip`
- `zip -d arc.zip file.txt`
- `zip file.zip -u newfile`
- `unzip -l archive.zip`

Linux Shell Script

Archiving with tar

- `tar -cf output.tar`
- `tar -cf output.tar file1 file2 file3 folder1`
- `tar -tf archive.tar`
- `tar -tvf archive.tar`
- `tar -xf archive.tar`

Linux Shell Script

sed – noninteractive stream editor

- sed 'command' filename(s)

Linux Shell Script

awk is a program, which has its own programming language for performing data processing and to generate reports

➤ awk 'pattern' filename

Linux Shell Script

- List the current network interface configuration:
 - `ifconfig`
- set the IP address for a network interface
 - `ifconfig wlan0 192.168.0.80`
- Printing the list of network interfaces
 - `ifconfig | cut -c-10 | tr -d ' ' | tr -s '\n'`
- Displaying IP addresses
 - `ifconfig iface_name`

Linux Shell Script

- DNS lookup
 - `echo 192.168.0.9 backupserver >> /etc/hosts`
- Showing routing table information
 - `route`
 - `route -n`
- Let us ping
 - `ping ADDRESS`
 - `ping 192.168.0.1 -c 2`

Linux Shell Script

Running commands on a remote host with SSH

- connect to a remote host with the SSH server running
 - `ssh username@remote_host`
- connect to an SSH server running at port 422
 - `ssh user@localhost -p 422`
- run a command at the remote host and display its output on the local shell
 - `ssh mec@192.168.0.1 'whoami'`

Linux Shell Script

Creating arbitrary sockets

- Set up the listening socket
 - `nc -l 1234`
- Connect to the socket using the following
 - `nc HOST 1234`

Linux Shell Script

Translating with tr

- `echo "HELLO WHO IS THIS" | tr 'A-Z' 'a-z'`
- `echo 12345 | tr '0-9' '9876543210'`
- `echo 87654 | tr '9876543210' '0-9'`
- `echo "Hello 123 world 456" | tr -d '0-9'`

Linux Shell Script

Checksum and verification

- `md5sum filename`
- `md5sum file1 file2 file3`
- `md5deep -rl directory_path > directory.md5`

Linux Shell Script

Downloading from a web page

- `wget URL`
- `wget ftp://example_domain.com/somefile.img -O dloaded_file.img -o log`
- `wget --limit-rate 20k http://example.com/file.iso`
- `wget -Q 100m http://example.com/file1
http://example.com/file2`
- `wget -c URL`
- `wget --mirror --convert-links exampledomain.com`
- `wget --user username --password pass URL`

Linux Shell Script

A primer on cURL

- `curl URL`
- `curl http://example.com --cookie "user=slynux;pass=hack"`
- `curl URL --user-agent "Mozilla/5.0"`