LINUX SOFTWARE PACKAGES MANAGEMENT

A Software Packages in Linux operating system is is an archive of files that contains,

- Binary files to be installed
- Any Configuration files needed for the application
- Metadata about the package

Packages are generally available as RPM packages (.rpm), Debian packages (.deb). Let's understand these package types one by one.

a) RPM PACKAGE (.rpm)

- RPM stands for Red Hat Package Manager

- Red Hat is the major distro for corporate and server-center environments

- Many major distros have adopted RPM as well such as OpenSUSE, Mandriva, etc.

- RPM can be run by itself from the command line or can use an intermediate tool like YUM (Yellowdog Updater, Modified)
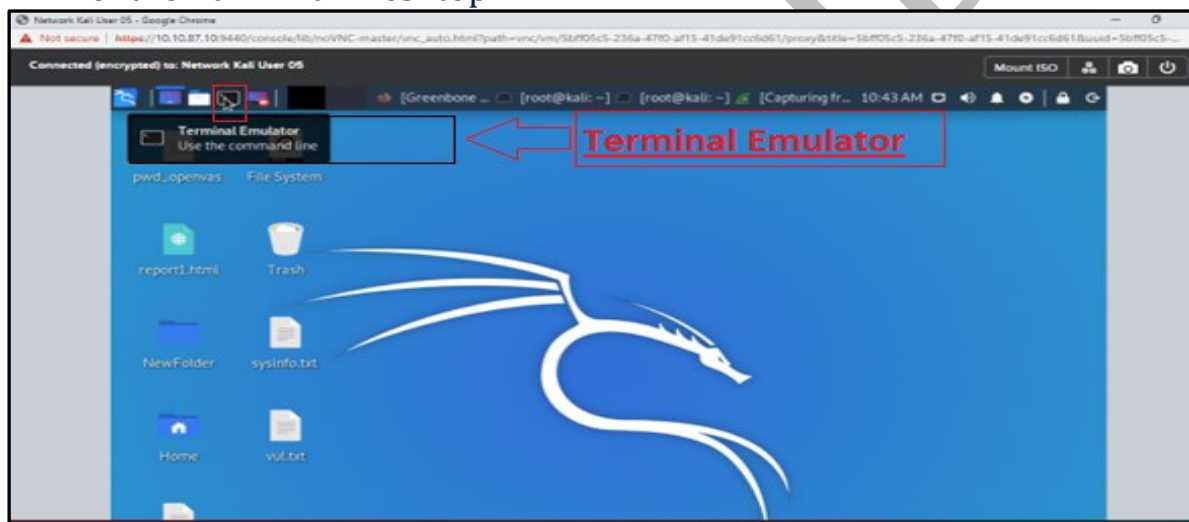
b) DEBIAN PACKAGE (.deb)

This packaging format was developed for the Debian Linux distribution. It is the de-facto standard for Debian Linux and derivatives such as Ubuntu. Each Debian package contains two archive files: one with control information, and the other with installable data. These archive files are in the .tar format

In this lab manual, you will learn about some commands to manage packages in Linux OS. To execute these commands, follow the below-given steps.

1. Connect to the kali Linux machine, created by you, using the RDP protocol.
2. When prompted for the username and password, enter root as username and toor as password. The root is the administrator user of the machine.
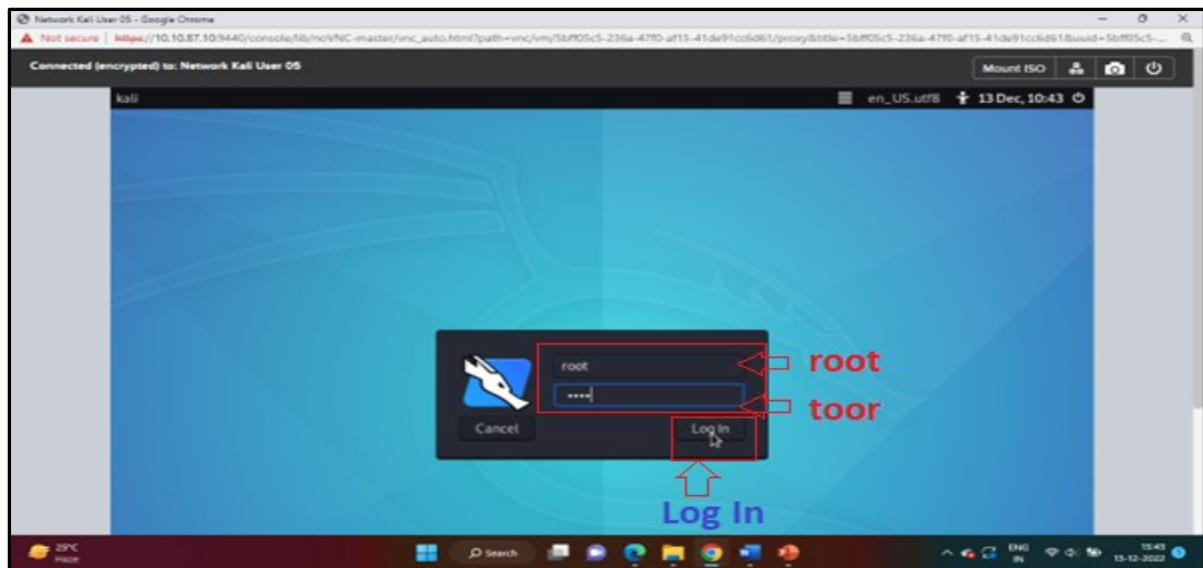
3. click on the black box icon (Terminal Emulator) in the top left corner of the Kali Linux Desktop.

Running the terminal while using the root account, allows you to run various commands with administrator rights.
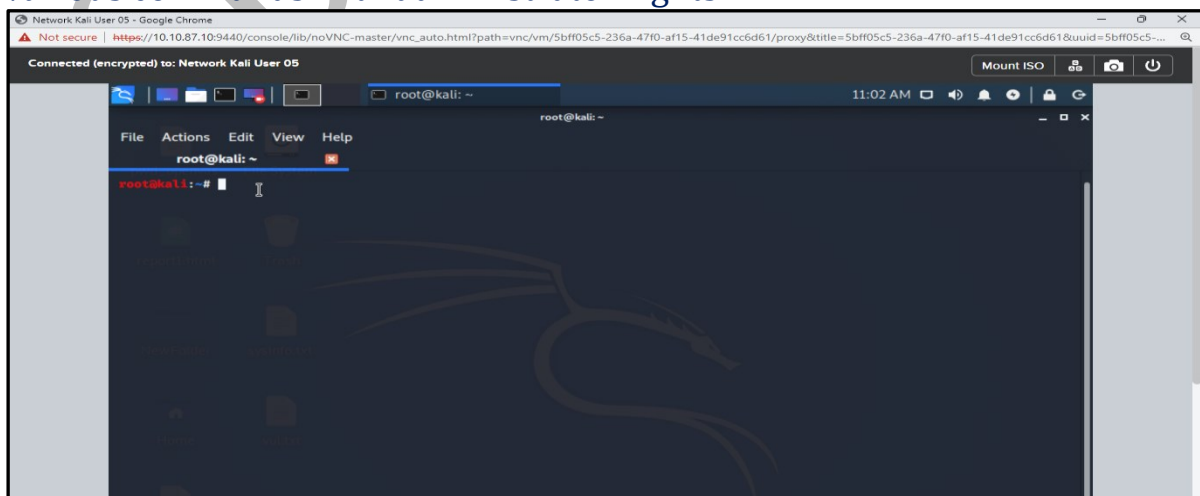
Let's Understand some important commands one by one.

Command to install Debian Packages

- Dpkg –i <package file>

- To install a .deb file, we install it by dpkg command with I option.

- Ex: sudo dpkg –i package_1.8.0_i386.deb

- This dpkg command can also be used to uninstall the package with r option.

- Ex: sudo dpkg –r package

APT and YUM tools for managing packages

- These tools allow you to search for, install, manage, update, and remove software.

A) APT Tool: Let's see few examples how to use apt command:

COMMAND	PURPOSE
$ sudo apt-get update	Get the latest package versions
$ sudo apt-cache search vsftpd	Find package by key word (such as vsftpd)
$ sudo apt-cache show vsftpd	Display information about a package
$ sudo apt-get install vsftpd	Install the vsftpd package
$ sudo apt-get upgrade	Update installed packages if upgrade ready
$ sudo apt-cache pkgnames	List all packages that are installed

c) YUM PACKAGE

"Yum is an automatic updater and package installer/remover for rpm systems. It automatically computes dependencies and figures out what things should occur to install packages. It makes it easier to maintain groups of machines without having to manually update each one using rpm."

For more information, you can go to this link: http://yum.baseurl.org/

Command to install the package using Yum:

- sudo yum install package-name

Updating Software Packages

sudo apt-get update fetches the latest version of the package list from your distro's software repository, and any third-party repositories you may have configured. In other words, it'll figure out what the latest version of each package and dependency is, but will not actually download or install any of those updates.

```
┌──(cdac㊙cdac)-[~]
└─$ sudo apt-get update
[sudo] password for cdac:
Get:1 http://ftp.harukasan.org/kali kali-rolling InRelease [30.6 kB]
Get:2 http://ftp.harukasan.org/kali kali-rolling/main amd64 Packages [18.4 MB]
Get:3 http://ftp.harukasan.org/kali kali-rolling/main amd64 Contents (deb) [42.9 MB]
Get:4 http://ftp.harukasan.org/kali kali-rolling/contrib amd64 Packages [117 kB]
Get:5 http://ftp.harukasan.org/kali kali-rolling/contrib amd64 Contents (deb) [158 kB]
Get:6 http://ftp.harukasan.org/kali kali-rolling/non-free amd64 Packages [212 kB]
Get:7 http://ftp.harukasan.org/kali kali-rolling/non-free amd64 Contents (deb) [942 kB]
Fetched 62.8 MB in 58s (1,088 kB/s)
Reading package lists... Done
```

a) The sudo apt-get upgrade command downloads and installs the updates for each outdated package and dependency on your system.

b) Running sudo apt-get upgrade will not automatically upgrade the outdated packages – you'll still have a chance to review the changes and confirm that you want to perform the upgrades.

```
┌──(cdac㊙cdac)-[~]
└─$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  liblttng-ust-ctl4 liblttng-ust0
Use 'sudo apt autoremove' to remove them.
The following packages have been kept back:
  apache2 apache2-bin apache2-data apache2-utils axel cron curl freerdp2-x11 galera-4
  gnome-control-center graphviz gstreamer1.0-pipewire gstreamer1.0-plugins-bad
  gstreamer1.0-plugins-ugly hydra hydra-gtk ike-scan impacket-scripts kmod libafflib0v5
  libapache2-mod-php8.1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libavcodec58
  libavfilter7 libavformat58 libavutil56 libcrypt-ssleay-perl libcryptsetup12 libcurl4
  libdns-export1110 libegl-mesa0 libevent-core-2.1-7 libevent-openssl-2.1-7
```

```
   vim-tiny vpnc vpnc-scripts wget winexe wireless-regdb wireshark wire
   wordlists xdg-dbus-proxy xdg-desktop-portal xwayland xxd zenity-comm
654 upgraded, 0 newly installed, 0 to remove and 149 not upgraded.
Need to get 927 MB of archives.
After this operation, 37.8 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

c) To see which packages are upgradable
   apt list –upgradable

- kris@pihole:~ $ apt list --upgradable
- Listing... Done
- libcamera0/stable 0~git20220426+18e68a9b-1 armhf [upgradable from: 0~git20220303+e68e0f1e-1]
- raspi-config/stable 20220425 all [upgradable from: 20220419]
- rpi-eeprom/stable 13.13-1 armhf [upgradable from: 13.12-1]