Package Management

Introduction and/or Background

Repositories

A repository is a pool of files that make up a version of a Linux distribution. See the example file below. These repositories are located on mirror sites around the world. See below. Storage is represented as either source code you download and compile or .ISO files in CD or DVD format. Some Linux variants even have a Net Install option. You download a very small .ISO, copy to CD/DVD then boot and the balance is done over the network.

The Linux world is split into camps defined by their package type. They are -

- Debian, Ubuntu, Linux Mint distributions – .DEB file types.
- Fedora, Red Hat, Centos – .yum/.dnf file types.

Simply put, ya can't mix'em at the repository level as the files themselves are ready to use binaries. We focus on the Debian, Ubuntu, Linux Mint distributions in this course. There are two other methods of delivering software in the Debian camp. -

- ppa
- .DEB

A .ppa is a private repository site supported by a development team that has not been accepted into the official repositories yet or by some other demand. A .DEB file is a packaged software program provided by a developer. This option is popular with paid for software. Yes there is software you can buy for Linux. It's not all free. There is an advantage to the .ppa approach. Whenever the software is updated you get access to the updates via an update command.

The Kali Debian repository:
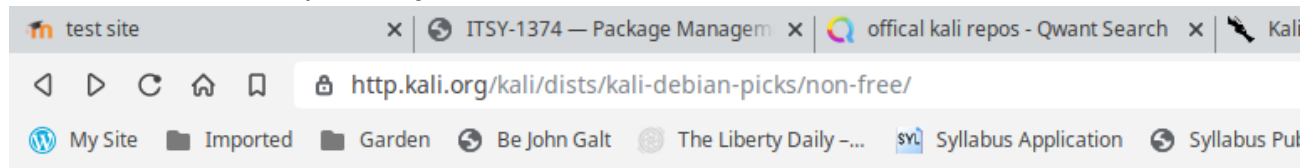
Index of /kali/dists/kali-debian-picks/non-free

Name	Last modified	Size	Description
Parent Directory		-	
binary-amd64/	2019-09-04 06:04	-	
binary-arm64/	2019-08-16 07:47	-	
binary-armel/	2019-08-16 07:47	-	
binary-armhf/	2019-09-04 06:04	-	
binary-i386/	2019-09-04 06:04	-	
debian-installer/	2017-03-16 13:48	-	
source/	2019-09-04 06:04	-	

Apache/2.4.10 (Debian) Server at http.kali.org Port 443

Example mirror sites for Ubuntu:

Ubuntu

Overview Code Bugs Blueprints Translations Answers

Official CD Mirrors for Ubuntu

Ubuntu » **Mirrors**

These mirrors offer ISO images which you can download and burn to CD to make installation disks.

There are **327** mirrors registered for Ubuntu.

Argentina		2 Gbps	4 mirrors
Cooperativa Telefónica de Villa Gobernador Gálvez Ltda.	http rsync	1 Gbps	
Xfree Argentina	http	1 Gbps	
Mendoza CONICET	http	10 Mbps	
Universidad Nacional de Córdoba	http	2 Mbps	
Australia		7 Gbps	8 mirrors
iiNet	http ftp rsync	1 Gbps	
AARNet Pty Ltd	http ftp rsync	1 Gbps	
Netspace Online Systems	http ftp	1 Gbps	
Over The Wire	http ftp	1 Gbps	
Optus	http ftp	1 Gbps	
Servers Australia	http	1 Gbps	
UberGlobal	http rsync	1 Gbps	
Internode	http ftp rsync	100 Mbps	
Austria		100 Mbps	1 mirror
lagis Internet Serviceprovider GmbH	http ftp	100 Mbps	

Package Commands

Keep in mind, generally to run any package command to completion you must be root user or shelled to root either by sudo or su. If your terminal is not showing # you are not ready.

dpkg

dpkg is the base level package manager for Debian based systems.
Example - dpkg -i <.DEB package name>

Here are the common options to use with dpkg -

- i - install
- r - remove
- C - audit for incomplete installs
- –get-selections - list of all installed packages
- –set-selections - set a list of packages to install from a file
- D - debug an package

There are at least 50 command and option combinations that can be used with dpkg. It's one of the reasons that apt and aptitude exist, providing more straightforward access to what dpkg does.

apt

Example -
sudo apt install <package_name>
subcommand options -

- get - get a package from the repository. The older version of apt.
- remove - remove a previously installed package.
- install - install a new package.
- purge - remove header and support files of a package.
- search - look for a package by name.
- show - what is in a package.
- list - displays a compendium of packages available to upgrade.
- update - refresh the current repository package list in the system.
- upgrade - update all the packages in list to the latest version.

The confusing part for most people is the ability to find the right package to install. For example DNS server program is not 'DNS' but 'Bind'.

apt-cache

apt-cache operates at the individual package level. the list subcommand being the lone exception. The apt-cache provides management options on

individual or groups of files. Example: apt-cache show <package_name>

subcommand options -

- showpkg - what is inside a package
- search - find a package named
- depends - Shows a listing of each dependency a package has and all the possible other packages that can fulfill that dependency.
- rdepends - Shows a listing of what depends on a package.

aptitude
aptitude is a convenient command line front end program to the apt command itself. It's more consistent in its presentation and if you really get lost you can just type aptitude by itself and it will provide a menu structure to walk you through whatever you are wanting to accomplish. aptitude has the same subcommand structure as apt above.

Deb Files
As discussed a .DEB is the ready to install package of a particular program. There are a multitude of ways to install a .DEB file -

- dpkg
- apt
- aptitude
- Synaptic

Pick the one you are most comfortable with. They will all work.

Dependencies
In the dim history of computing a developer would/could write a program as a single block of code. Those days are long gone however. These days programs are written in modules each with a specific purpose. A complete program consists of multiple submodules that can be called and used as needed by the main program.

Our interests in dependencies are of a diagnostic bent. Let us say for some reason an install went wrong. We need to determine what failed. Fortunately our first tool is the install itself. Errors in an aptitude install will be displayed on the screen. It might be that the repository where the dependency resides is not online.

If it is not a repository problem the next step, look at the dependency list.
Assume aptitude is having a problem -
apt-cache depends aptitude

Then run,

`ldd /usr/bin/aptitude`

ldd provides dependency, file location and memory allocation. Comparing the two outputs is a dependency missing? If so you may have to install that dependency manually.

Objectives
In this project/lab the student will:
- Gain familiarity with package management tools

Equipment/Supplies Needed
- As specified in Lab 0.0.1.

Procedure
Perform the steps in this lab in the order they are presented to you. Answer all questions and record the requested information. Use the Linux Virtual Machine to perform lab activities as directed. Unless otherwise stated, all tasks done as a non-root user. If root access is needed use the sudo command.

Assignment

apt

Launch Debian. Enter root access using either **su** or **sudo**.

Simply try:

1.   `apt show aptitude`

2.   `apt install aptitude`

Record a screenshot. Place that image in a Word or Writer document. Aptitude is now installed if no errors were reported.

apt-cache

Execute:

3.   `apt-cache show aptitude`

Looking at the output, what is the size of this executable?

Try:

4.   `apt-cache depends aptitude`

and

5.   apt-cache rdepends aptitude

aptitude

You can run aptitude two ways. First:

6.   aptitude

Second:

7.   aptitude install airstrike && airstrike

Record a screenshot. Place that image in a Word or Writer document. The first form brings up a menu structure. The second is similar to the apt command.

.DEB files

Any of the tools so far discussed can also be used on .DEB files. Example:

8.   dpkg -i /path/airstrike.deb

or

9.   aptitude install /path/airstrike.deb

The link to download the file is here:
http://ftp.br.debian.org/debian/pool/main/a/airstrike/airstrike_0.99+1.0pre6a-6_amd64.deb. It's Best to provide the full path to the file. An alternative method is thru file manager. Double click on the file in GUI mode. Due to the extension name, Debian will bring up the Synaptic package manager, ask for system password, then install the package.

ldd

Whether you know it or not, most programs used are built like a jigsaw puzzle. Each developer assembling the pieces to build a whole program. Want to know what the pieces are? ldd is the tool to do just that.

Performing the following actions:

10.   cd /usr/bin

11.   ldd gedit

Each of the various *.*.so files is a library component that is available in the linux subsystem.

Execute:

12.   ldd -v gedit

Record a screenshot. Place that image in a Word or Writer document. At your option explore the dpkg program.

Lab Submissions Proof: Provide screenshots as indicated in the lab; upload your proof to Canvas for grading.

Rubric

Checklist/Single Point Mastery

Concerns Working Towards Proficiency	Criteria Standards for This Competency	Accomplished Evidence of Mastering Competency
	Criteria #1: Record screenshot of executed apt show aptitude and apt install aptitude (25 points)	
	Criteria #2: Record size of apt-cache show aptitude executable (25 points)	
	Criteria #3: Record screenshot of aptitude install airstrike && airstrike (25 points)	
	Criteria #4: Record screenshot of ldd -v gedit (25 points)	