LINUX FOUNDATION

TRAINING

Software Management

A Linux Foundation Training Publication
www.training.linuxfoundation.org

Overview

A Linux distribution is more than a collection of software, it is a commitment by a community to continually compile, test, package, and integrate software from a variety of sources into a working operating system. In fact, not having to do this work yourself is one of the main reasons to choose a distribution over creating your own. As part of this effort, most of the major Linux distributions maintain package repositories: centralised sources of known working compiled software that make up a particular distribution.

Though each Linux distribution manages this process differently, software is generally placed into a repository that corresponds with the community's confidence with its stability. So the newest and least tested versions will go in one repository, while the more stable, thoroughly used versions will go in another. Users can then decide on their level of risk tolerance versus their desire for the newest features.

A software package is the combination of compiled source code and metadata that describes how to install it on a particular operating system, including any additional package it depends on. The two most widely package formats used are .deb packages for Debian and its derivatives like Ubuntu, and .rpm packages for Red Hat Enterprise Linux, CentOS, and more.

Key Ideas

Software package: a container for compiled software and metadata about how to install it. Most common are .deb and .rpm

Software repository: trusted central source of compiled, tested, integrated software packages provided by a distribution.

Low level package management: actions that pertain to an individual package: install, upgrade, query, remove. Does not handle dependency resolution. dpkg command in Debian-derived distributions, and rpm in distributions made up of rpms.

High level package management: System level actions like system updates, accessing repository metadata, multiple package upgrades, and dependency resolution. apt in Debian-derived distributions, yum in rpm-based distributions, zypper in OpenSuse.

Software dependency: Any software package may require other software packages to run. High level package management tools check for and install any unmet dependencies when installing a package.

Example Scenario

In these examples, you'll be performing some common package management commands using high level and low level tools.

Now Do It

1. List all the packages installed on your system.
2. Find out if the bash package is installed on your system.
3. Find out more information about the bash package.
4. Find out what dependencies bash has.
5. Find out what files that bash installs, and where.
6. Find out which package provides the /etc/hosts file.
7. Download the latest information about the contents of your distribution's repositories.
8. Search for KVM in your distribution's software repositories.

If you remember nothing else...

Your distribution provides you with a safe, trusted source of packages. Packages created by and for one distribution probably won't work on another, even if they use the same package format.

Answer Key

1. List all the packages installed on your system.
rpm -qa
dpkg -l

2. Find out if the bash package is installed on your system.
rpm -qa | grep bash
dpkg -l | grep bash

3. Find out more information about the bash package.
rpm -qi bash
dpkg -I bash

4. Find out what dependencies bash has.
rpm -qpR bash
This only works when applied to a .rpm file. To find out the dependencies of an installed package, leave out the -p option. For example, to find out the dependencies of bash, do:
rpm -qR bash

dpkg -I bash

5. Find out what files that bash installs, and where.
rpm -ql bash
dpkg -L bash

6. Find out which package provides the /etc/hosts file.
rpm -qf /etc/hosts
dpkg --search /etc/hosts

7. Download the latest information about the contents of your distribution's repositories.
yum makecache
apt-get update
zypper refresh

8. Search for KVM in your distribution's software repositories.
yum search kvm
apt-cache search kvm
zypper search kvm

LINUX FOUNDATION

Get Certified!

Get more information at http://training.linuxfoundation.org/certification/lfcs