Linux Software Management

Linux System Administration
COMP2018 Summer 2017

OSS Distribution And Installation

- Open Source Software at its simplest is just source code distributed by any of several means (e.g. download from ftp or web server, on CD or DVD, etc.)

- The source code is usually distributed in an archive

- Unpack the archive, compile the code as needed, and copy or move the resulting files into the desired locations in the filesystem

- Maintaining such software is a matter of redoing parts or all of the above

File Archives

- The traditional UNIX archive format is a tarfile, named for the tar utility used to create and extract the archive files
- tar was designed to turn a file hierarchy into a stream of bytes suitable for writing to a backup tape, or recreating that hierarchy from such a stream
- Tape backup devices are uncommon now
- tar is still commonly used but the archives are written to files or read back from files instead of backup tapes
- Those files are often compressed using the gzip utility in a command pipeline, or directly by tar

tar

- tar requires some options to tells it what to do
- These are the most common:
  - c for create archive
  - x for extract files
  - t for table of contents
  - v shows file names as tar processes them
  - f specifies where to read or write the archive to or from, a pathname or - for stdin/stdout
  - z automatically gzips or gunzips the archive
- tar archives are known as tarballs and named with an extension of .tar
- gzip compressed tarfiles are named with an extension of .tgz
- https://www.howtogeek.com/248780/how-to-compress-and-extract-files-using-the-tar-command-on-linux/ has some examples

Packages

- A complete software package will have the necessary files to implement the software, plus scripts to install, configure, reconfigure, and remove the software
- A software package is simply an archive of these files in a format expected by a package manager program
- Packages for Linux are commonly distributed by download from software repositories
- The operating system package repositories (repos) are configured in text files on a Linux system and can be updated or changed as needed

Debian Software Configuration

- dpkg is the tool that does all the package management work and is useful for some manual tasks
  e.g. dpkg -l

- Running the reconfiguration scripts for a package is done using dpkg-reconfigure packagename

- After installation, some software packages are further configured by editing configuration files or by running additional tools, typically installed as part of the package

Debian Software Management

- Reviewing installed software and identifying which files belong to which packages can be done with dpkg
  e.g. dpkg -s packagename
  dpkg -l packagename
  dpkg -L packagename
  dpkg -S filename

- unattended-upgrades is a package you can use to automatically install updates, use
  dpkg-reconfigure unattended-upgrades
  to turn it on or off

Advanced Package Tool

- dpkg is a limited tool in that it works directly with deb files, not repositories, and does not resolve and install dependencies automatically

- apt is a friendlier face for the apt-get tools
  e.g. apt install memstat, apt remove bfgminer

- APT uses a database of software packages kept in /var/cache/apt, updated with apt update

- Always make sure your package database is up to date before doing software installations or upgrades

- You can manually upgrade your software by using apt upgrade, be sure to have an updated software database when doing this

apt

- apt is the replacement for apt-get, apt-cache, and various other apt-* commands, designed to simplify interaction with software packaging, see https://itsfoss.com/apt-vs-apt-get-difference/

- apt has several commonly used subcommands
  - apt search keyword
  - apt install packagename
  - apt update
  - apt upgrade
  - apt remove packagename

Redhat Packages

- Similar to the deb packages, Redhat packages are archives of files that include software, configuration files, and scripts for installation, configuration, reconfiguration, and removal of the package
- The Linux Standards Base defines RPM as the standard package management system
- The rpm command is used for low level package work

The rpm Command

- The rpm command is useful for reviewing installed software
  - rpm -qa will list all installed packages
  - rpm -ql packagename will list the files that a package includes
  - rpm -qi packagename will show the state of a package on the system
  - rpm -qf pathname will tell you what package a file came from if it came from a package
- While the rpm command can be used to install and remove software, it is better to use a more user-friendly command like yum

YUM

- yum is a front end to the rpm utility that allows for high level operations and resolves dependencies
- Several subcommands are commonly used
  - yum search keyword
  - yum install packagename
  - yum update [packagename]
  - yum remove packagename
- yum gets package information directly from the repositiories so there is no local cache to update before upgrading software

Other Package Managers

- There are other package formats and tools
- https://www.linode.com/docs/tools-reference/linux-package-management gives an overview of some of them

Software Repositories

- Linux software packages are usually kept online in software repositories
- The various package management tools have their own methods and configuration files defining how they automatically use these repositories (e.g. the sources.list files in /etc/apt)
- You can always download package files yourself and manually install them if a situation calls for it

Service Management

- There are several ways to manage services once they are installed on servers
- The service command can control a service and inform you about the service (e.g. service sshd status)
- The systemctl command is specific to the init replacement program systemd (see https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/sect-Managing_Services_with_systemd-Services.html for more detail on changes from service to systemctl)
- The journalctl command allows you to view the logs created by systemd

System Logs

- On servers, most installed software is run without user interaction
- Error and informational messages are still generated and get saved for later review as needed
- The default log directory is /var/log and various programs are configured to save log messages in files there
- Many programs do not directly write log files, but instead send messages to a logging service (rsyslog is most common) which evaluates and saves those messages according to the service configuration
- Log files are usually simple text files with messages written in chronological order and timestamped, logwatch can analyze log files
- The logrotate program is periodically run automatically to age these files so that the log storage device doesn't become full
- The kernel has a special ring buffer to hold its messages, which can be viewed and manipulated with the dmesg command