

11: Managing Packages and Software

Managing Software with RPM and YUM

Scenario

One of the other **Develetech** administrators has asked you to demonstrate the software management lifecycle on a CentOS server. You will use the **KornShell (ksh)** as an example of how to use the **rpm** command, and then the **Very Secure FTP daemon (vsftpd)** to demonstrate the **yum** command.

- Objectives

- Completing this activity will help you to use content examples from the following syllabus objectives:
 - 2.1 Given a scenario, conduct software installations, configurations, updates, and removals

1. Use the **rpm** command to manage the software lifecycle

- Log in as **student01** with **Pa22w0rd** as the password.
- Open a terminal.
- Enter **sudo rpm -ivh /Packages/ksh-20120801-139.el7.x86_64.rpm** to install the **ksh** package in verbose mode and with a hash progress bar.
 - Don't forget to use tab completion for long filenames!
- Enter **rpm -qi ksh** to view information on the **ksh** package.
 - **ksh** is another shell environment, similar to **bash**. It is common on Unix servers.
- Enter **sudo rpm -Vv ksh** to verify the **ksh** installation.
- Enter **sudo rpm -ql ksh** to list the files in the **ksh** package.
- Enter **sudo rpm -e ksh** to "erase" or uninstall the **ksh** package.

2. Use the **yum** command to manage the software lifecycle

- Enter **yum info vsftpd** to discover information about the **vsftpd** package.
 - **vsftpd** is the Very Secure FTP service.
- Enter **sudo yum localinstall /Packages/vsftpd-3.0.2-25.el7.x86_64.rpm** to install the **vsftpd** package.
 - You may receive a repeating messages that says an "Existing lock /var/yum/yum.pid: another copy is running." Open a second tab in the terminal (or a second instance of a terminal), and then enter the following command:

```
sudo systemctl kill packagekit
```

- Enter **y** when prompted to complete the installation.
 - If you include a **-y** option with **yum**, it will automatically answer yes to this prompt regarding installing dependencies and not pause the installation.
- Enter **yum info vsftpd** to view information on the **vsftpd** package.

- Enter `yum provides /etc/vsftpd/vsftpd.conf` to discover what package the configuration file belongs to.
- Enter `sudo yum -y remove vsftpd` to uninstall the `vsftpd` package.

Managing Software with dpkg and APT

Scenario

Some Linux systems in Develotech run Ubuntu and other versions of Debian. Just like your Red Hat-based systems, these need to undergo network troubleshooting from time-to-time. So, you'll download and install the `nmap` package on these machines to ensure you have the right toolset for the job. You'll use `dpkg` and `APT`.

Objectives

- Completing this activity will help you to use content examples from the following syllabus objectives:
 - 2.1 Given a scenario, conduct software installations, configurations, updates, and removals

1. Load the Ubuntu VM

- From the desktop menu, select `Applications→System Tools→Virtual Machine Manager`.
- Enter the root password.
- Right-click `ubuntu-vm` and select `Run`.
- Right-click `ubuntu-vm` and select `Open`.
- Wait for the VM to load.
- Select the student account, then enter `Pa22w0rd` as the password.

Recall that there are two primary branches to the Linux family: those distributions derived from Red Hat and those derived from Debian. One of the key differences between the two branches is software management. The Red Hat distributions use `rpm` and `yum`, while the Debian distributions use `dpkg` and `apt`.

2. Update the `APT` database with current package version information

- Right-click the desktop and select `Open Terminal`.
- Enter `sudo apt update`
- Enter the password when prompted.

If the Ubuntu VM has no Internet connectivity, you may need to restart your CentOS host and then reload the VM.

- Verify that the database update operation completed.
- You're presented with the number of packages that can be upgraded. If you wanted to upgrade an existing package, you could use the `apt upgrade {package name}` command. For now, you'll install a new package.

3. Download and install the nmap package

- Enter `sudo apt install nmap`

If you receive a "Could not get lock..." error, it means the APT package manager is automatically checking for updates. You could wait a few moments or enter the following commands to clear the lock:

- `sudo ps aux | grep apt`
- Observe the process ID number for the process related to `/var/lib/apt/apt.systemd.daily`
- `sudo kill -9 PID` where `PID` refers to the process ID number identified above
- `sudo rm /var/lib/dpkg/lock`
- If necessary, repeat the `sudo apt install nmap` command.
- Enter `y` to confirm the operation.

You might need to maximize the VM window or scroll to see the prompt asking you to confirm the operation.

- Wait for the installation to complete.
- Enter `apt show nmap` to discover information about the `nmap` package.
- Enter `nmap localhost` to test the utility, confirming that it executes and checks the VM's basic network functionality.

4. Shut down the VM.

- From the `Virtual Machine Manager (VMM)` interface, select `Virtual Machine→Shut Down→Shut Down`.
- Close the VM window.
- Close the `Virtual Machine Manager` window.

Configuring Repositories

Scenario

While the Linux vendors tend to provide online repositories, one of the concerns with using these is version control of applications. Develotech has decided to manage an internal repository of software packages, making version control much easier. You will configure a local ``YUM`` repository on the CentOS server. You will then make a ``YUM`` repository available using ``Apache HTTP Server``.

Objectives

- Completing this activity will help you to use content examples from the following syllabus objectives:

- 2.1 Given a scenario, conduct software installations, configurations, updates, and removals

1. Configure a local YUM repository

- If necessary, open a terminal.
- Browse to `/Packages` and view the available `.rpm` files.
- Enter `sudo createrepo /Packages` to designate that directory as a YUM repository.
- This may take a few minutes.
- Using `sudo`, create a file named `/etc/yum.repos.d/local-repo.repo` with the text editor of your choice.
- Edit this file by providing the following values:
 - Note that there are three forward slash characters in the `baseurl` path.

```
[local-repo]
name=Local Repository
baseurl=file:///Packages
enabled=1
gpgcheck=0
```

- Save and close the file.

2. Verify the location is recognized as a YUM repository

- Enter `yum clean all`
- Enter `yum repolist` and verify the `local-repo` is displayed.
- Enter `sudo yum -y --enablerepo=local-repo install ksh` to install the `ksh` (KornShell) package from the repository.

3. Install Apache HTTP Server to use as a repository

You can configure an Apache web server as a repository that provides packages to servers on your internal network.

- Enter `sudo yum -y install httpd`
- Enter `sudo systemctl start httpd` to start the Apache service.
- Enter `systemctl status httpd` and verify that Apache is active (running).

4. Designate Apache as a repository

- Enter `sudo ln -s /Packages /var/www/html/packages` to link the `/Packages` directory to Apache.
- Enter `sudo createrepo /var/www/html/packages` to designate the location as a YUM repository.

5. Create the repository reference file

- Using `sudo`, create a file named `/etc/yum.repos.d/internal-repo.repo` with the text editor of your choice.
- Edit this file by providing the following values:
 - Note that there are three forward slash characters in the `baseurl` path.

```
[internal-repo]
name=Internal Repository
baseurl=http://localhost/packages
enabled=1
gpgcheck=0
```

- Save and close the file.

6. Verify the location is recognized as a **YUM** repository

- Enter **yum clean all**
- Enter **yum repolist** and verify the **internal-repo** is displayed.
- Enter **sudo setenforce 0**
- This disables **SELinux**, an access control mechanism that would otherwise prevent access to the web-hosted packages.
- Enter **firefox http://localhost/packages** to see a list of packages from the **Apache web server**.
- Close **Firefox** when you're done.
- Enter **sudo setenforce 1** to re-enable **SELinux**.
- Enter **cd ~** to return to your home directory.

Acquiring Software

Scenario

You are investigating ways of downloading software from the web. Specifically, you are considering writing a script to automate the download process. You will use **wget** and **curl** to try the downloads manually.

- Objectives

-
- Completing this activity will help you to use content examples from the following syllabus objectives:

2.1 Given a scenario, conduct software installations, configurations, updates, and removals

1. Use the **wget utility** to download a file from the web
 - At a terminal, ensure you're in your home directory.
 - Enter **wget https://download.samba.org/pub/samba/samba-latest.tar.gz** to download the most recent source code file for the Samba service.
 - Check your home directory for a file named **samba-latest.tar.gz**
2. Use **curl** to download a file from the web
 - Enter **curl -o nmap-7.70.tar.bz2 https://nmap.org/dist/nmap-7.70.tar.bz2** to download version 7.70 of the Nmap utility.
 - Check your home directory for a file named **nmap-7.70.tar.bz2**
3. Expand a source code **tarball** so that it is ready to be compiled in a later activity
 - Enter **tar -xvjf nmap-7.70.tar.bz2** to extract the files.

- Verify that the source code files were extracted in the `~/nmap-7.70/` directory.
 - You will compile the Nmap utility source files in a later activity. Your current objective is just to acquire and unpack it.

Compiling and Installing an Application

Scenario

Develetech will be relying on Nmap to troubleshoot its networked systems and perform vulnerability assessments. You know that compiling Nmap from source code enables greater flexibility and control. You will do a basic software compile of Nmap.

- Objectives

-
- Completing this activity will help you to use content examples from the following syllabus objectives:
 - 2.1 Given a scenario, conduct software installations, configurations, updates, and removals

1. Install the GCC compiler

- Enter `rpm -qi nmap` to confirm nmap is not already installed.
- Enter `sudo yum -y install gcc-c++ --disablerepo=internal-repo` to install the necessary GCC compiler for Nmap.
- Wait for the package to finish installing.

2. Compile the Nmap source code

- Change to the `~/nmap-7.70` directory.
- Enter `./configure` to generate a makefile based on your system's configuration. This may take a few minutes.
 - Note the `./` characters in the above command. These characters tell bash to "look here" for the executable. By default, bash only checks certain directories for executable commands. Home directories are not usually checked.
- Enter `make` to compile the software based on the makefile instructions.
- Enter `sudo make install` to install the binaries on the system.
- Enter `/usr/local/bin/nmap` and verify that it is installed.
- The help file for nmap will print to the screen, indicating that the program is installed.
- Enter `/usr/local/bin/nmap localhost` to scan the local computer.