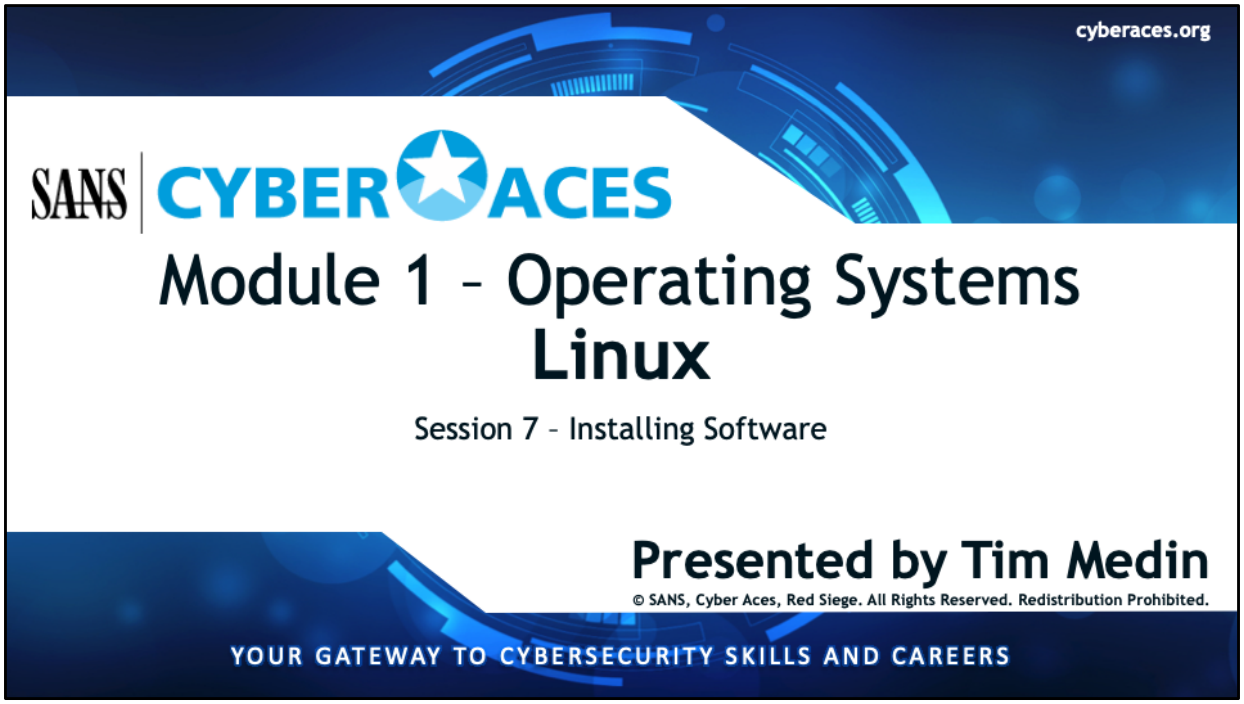
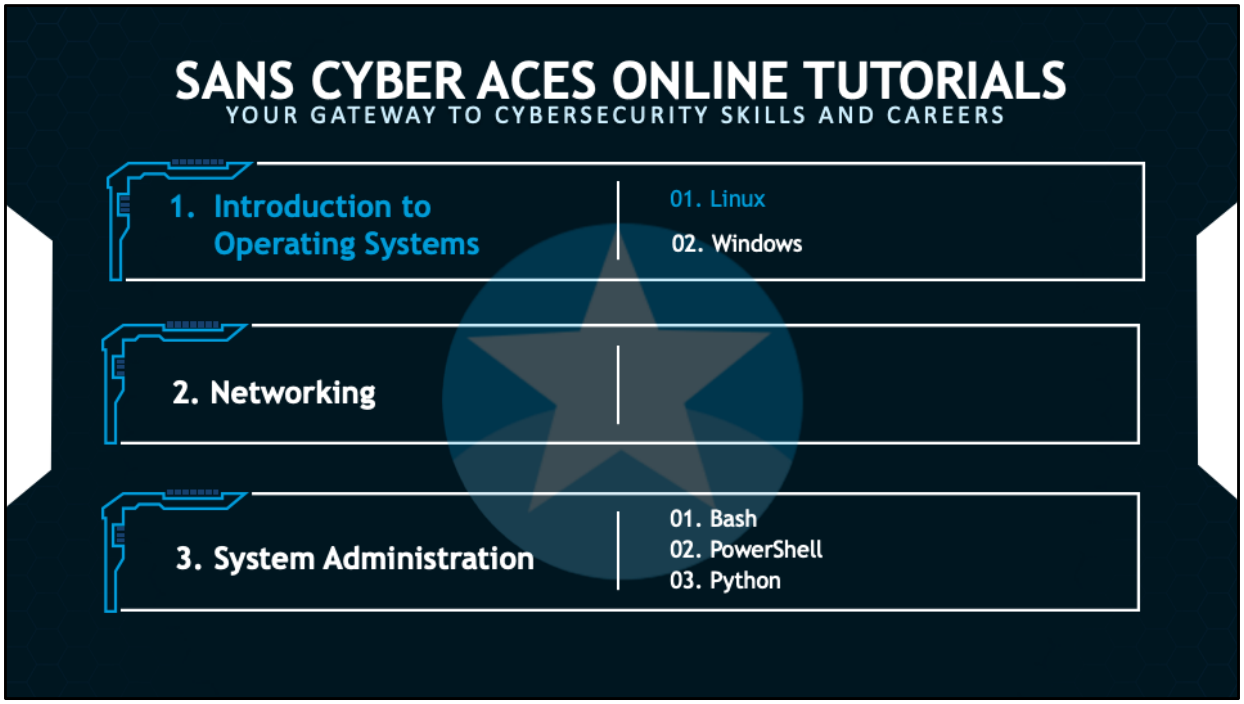


Welcome to Cyber Aces Online Module 1. In this session we will cover the ways to install software on Linux and in CentOS.

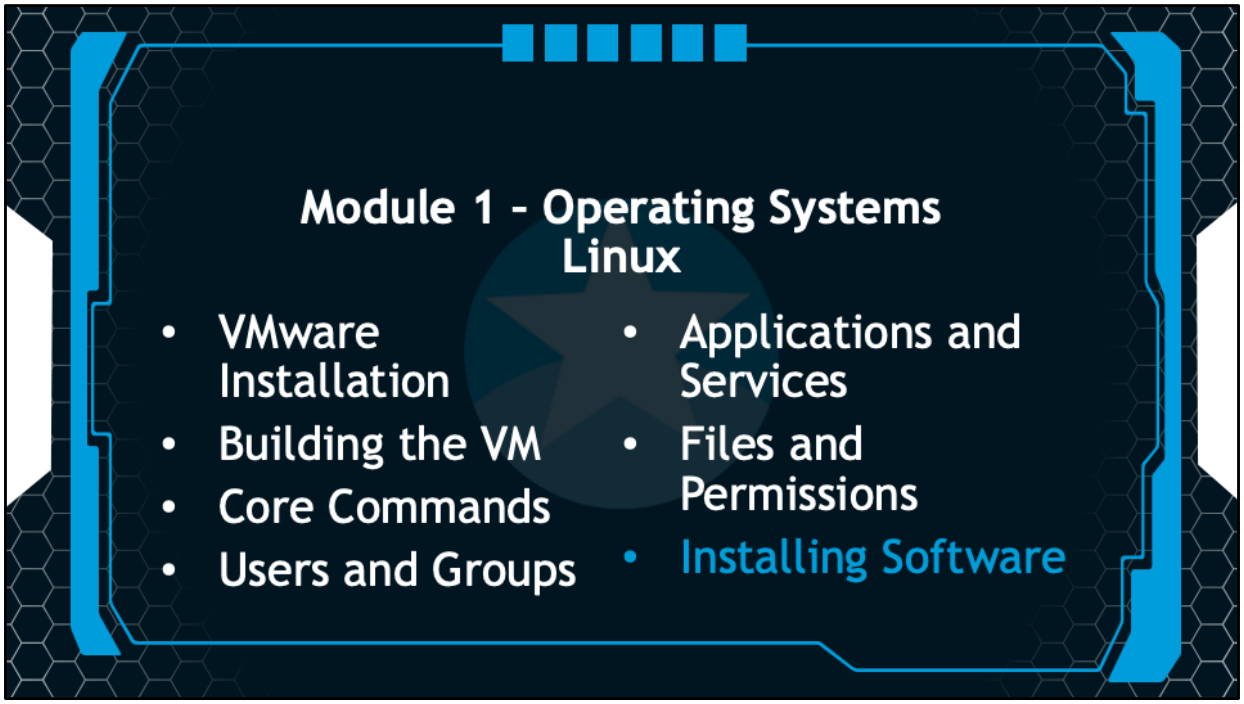
Content in this session has been developed by Tom Hessman, Tim Medin, Mark Baggett, Doug Burks, Michael Coppola, Russell Eubanks, Ed Skoudis, and Red Siege.



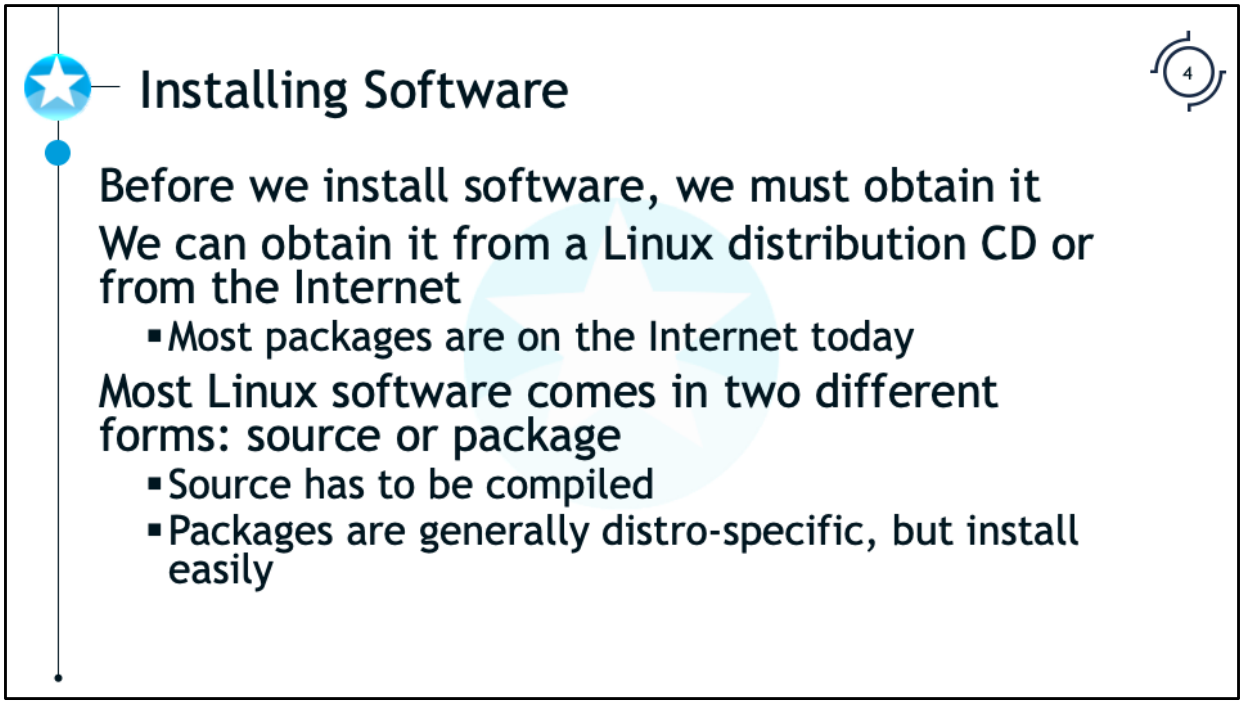
This training material was originally developed to help students, teachers, and mentors prepare for the Cyber Aces Online Competition. This module focuses on the basics of what an operating systems is as well as the two predominant OS's, Windows and Linux. This session is part of Module 1, Introduction to Operating Systems. This module is split into two sections, Linux and Windows. In this session, we will continue our examination of Linux.

The three modules of Cyber Aces Online are Operating Systems, Networking, and System Administration.

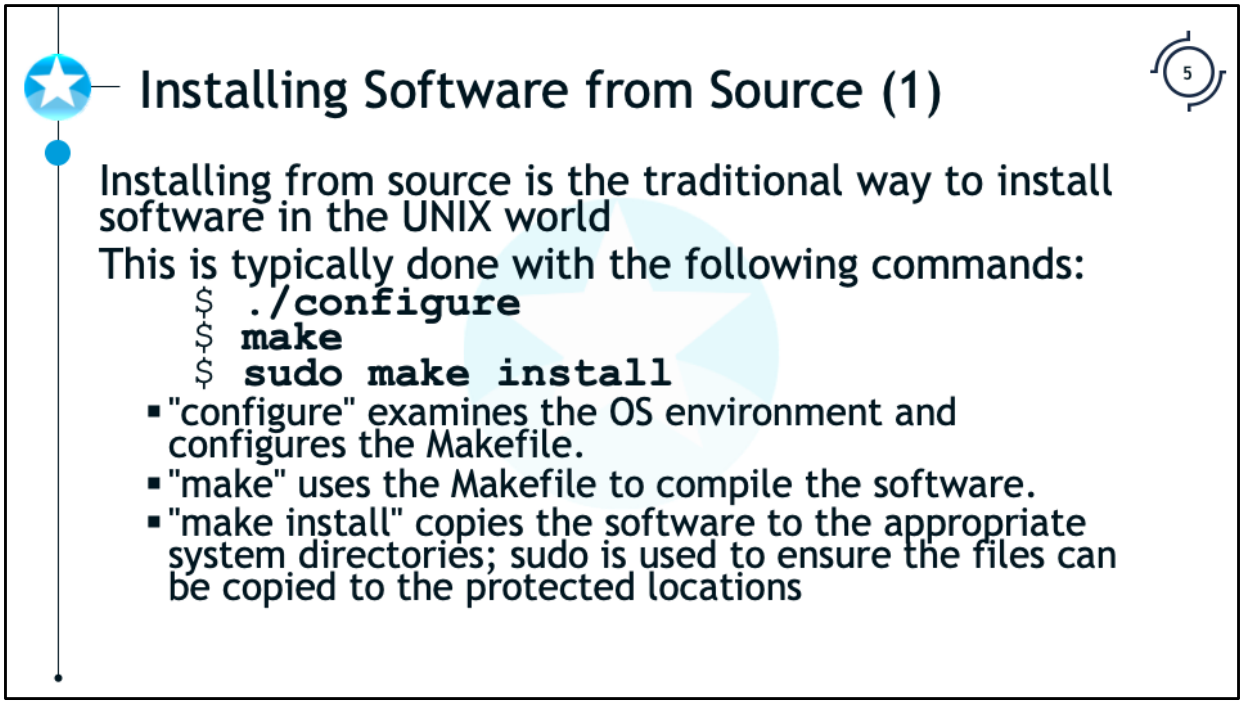
For more information about the Cyber Aces program, please visit the Cyber Aces website at https://CyberAces.org/.



In this session we will discuss different ways to install software on Linux, both from source and from a package.



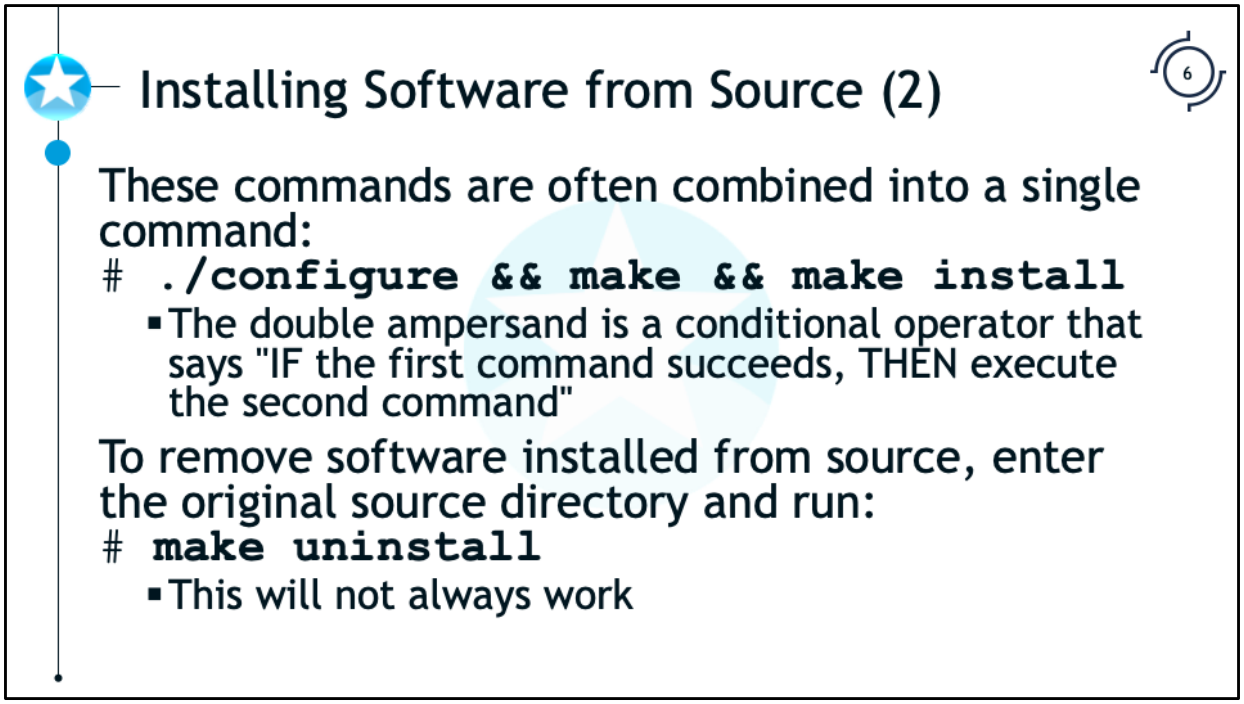
Before we can install software, we must obtain the software itself. We can obtain software from a Linux distribution CD or directly from the Internet. Nowadays, most packages are retrieved from the Internet. Most Linux software downloaded from the Internet comes in two different forms: source or package. Source packages contain source code, and therefore have to be compiled. Packages are generally distro-specific, but they are much easier to install.



The traditional method of software installation in the UNIX world is manually compiling source code into executable form. This is usually done with the following commands:

```
$ ./configure
$ make
$ sudo make install
```

The source tarball downloaded from the Internet contains a file called "configure", which we execute from the current directory by calling "./configure". This command examines the operating system and the software already installed on it and configures the Makefile which will be used in the next step. The "make" command uses the compiler that is already installed in the operating system to create binary executable programs. It references the Makefile created by the "./configure" step and compiles the source code into binary form. Finally, "make install" copies the newly-compiled binaries from the current directory to the appropriate system directories. This last step will usually copy the binaries (executables) to directories only writeable by root, as such you will need root permissions to perform this step.

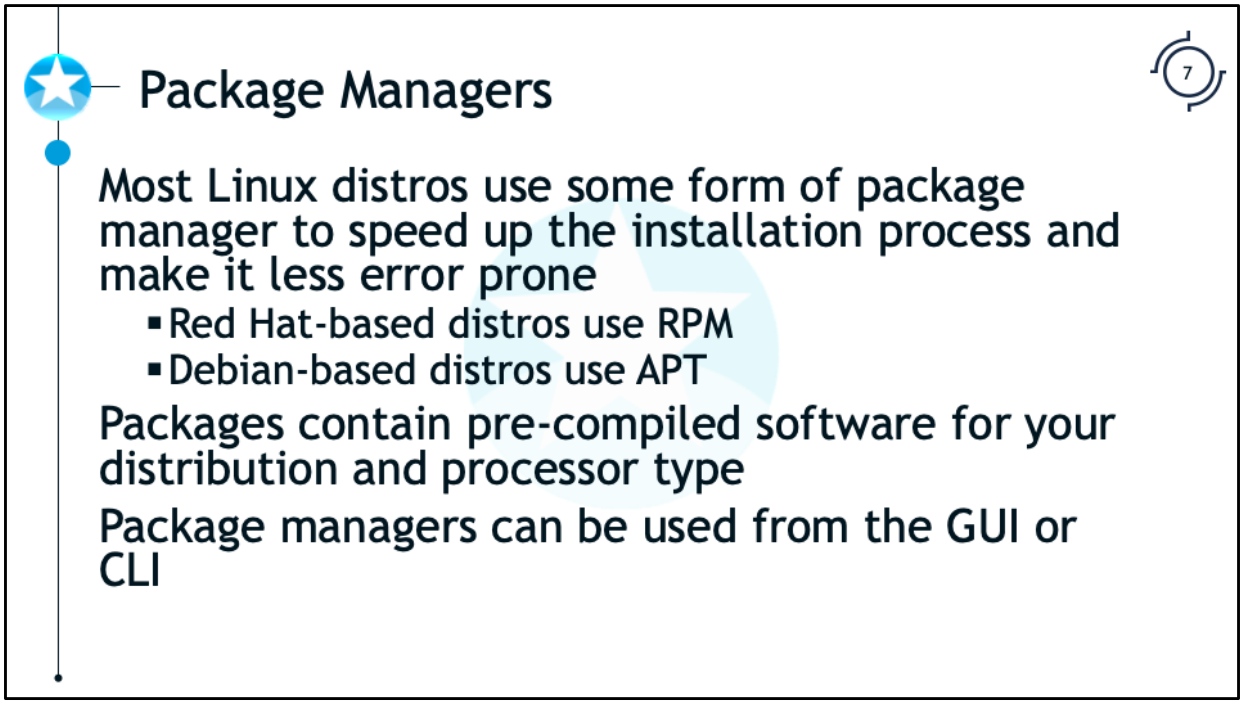


These three commands are commonly joined together on a single line using double ampersands (&&) like this:

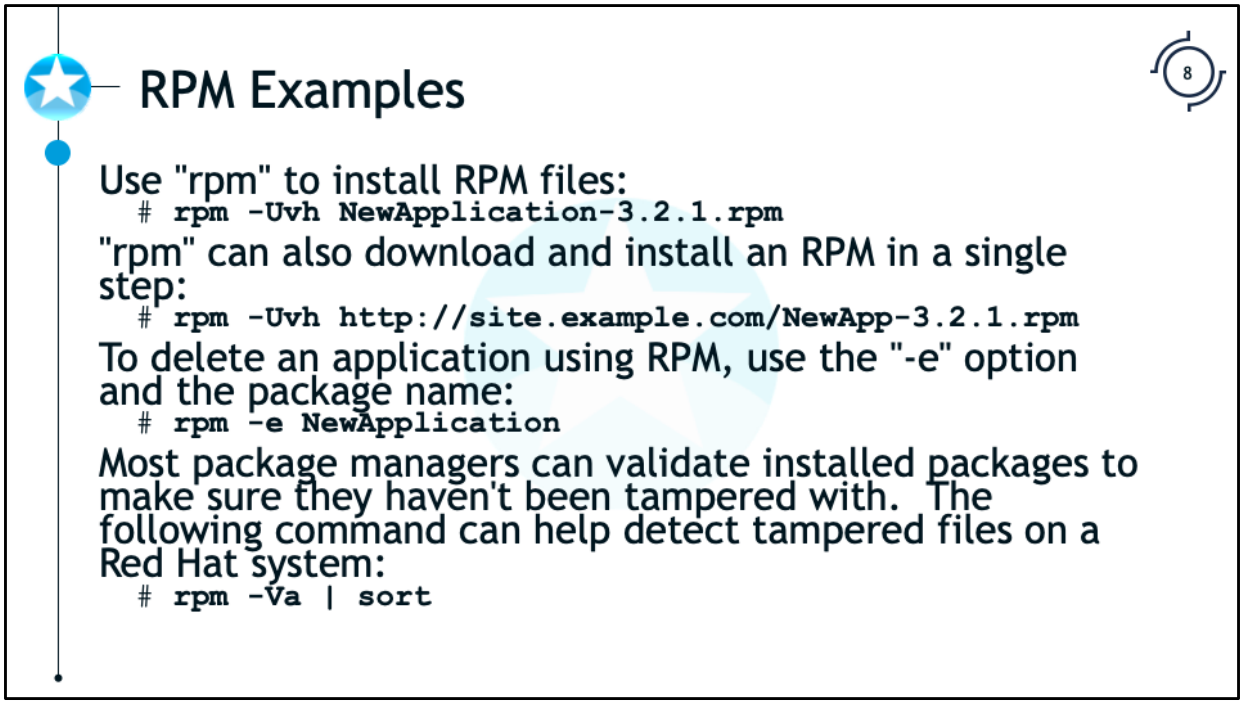
```
# ./configure && make && make install
```

The double ampersand is a conditional operator that says "IF the first command succeeds, THEN execute the next command".

To uninstall a program that was compiled from source, enter the original source directory and type "make uninstall". Unfortunately, not all software contains this feature in its Makefile.



To speed up the installation process and make it less prone to errors, most modern Linux distributions use some form of package manager. Distributions based on Red Hat use RPM, the RPM Package Manager (note the recursive acronym!). Distributions based on Debian (including Ubuntu) use the DEB package format with APT. In either case, the package is a single file that contains the entire application, pre-compiled for your distribution and processor. The package can be installed using a package manager with a graphical user interface, or from the command-line.



To install an RPM package on a Red Hat-based system, use the "rpm" command as follows:

```
# rpm -Uvh NewApplication-3.2.1.rpm
```

The "U" means "install or upgrade", the "v" means to print more verbose information, and the "h" means to print a progress bar during the install.

Red Hat systems can download and install an RPM package in one step using a command like this:

```
# rpm -Uvh http://site.example.com/NewApp-3.2.1.rpm
```

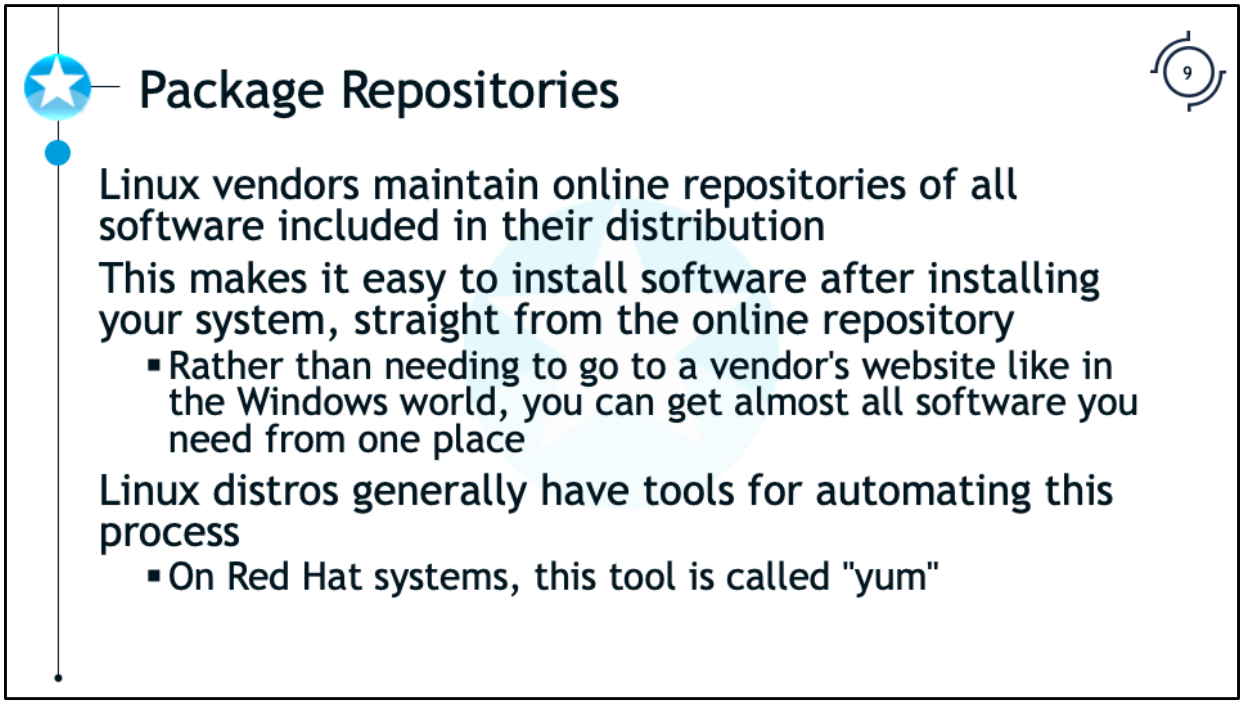
That same application could then be removed from the system with the following command:

```
# rpm -e NewApplication
```

If an attacker compromises a machine and modifies a file that belongs to an RPM package, then the following command can help detect that:

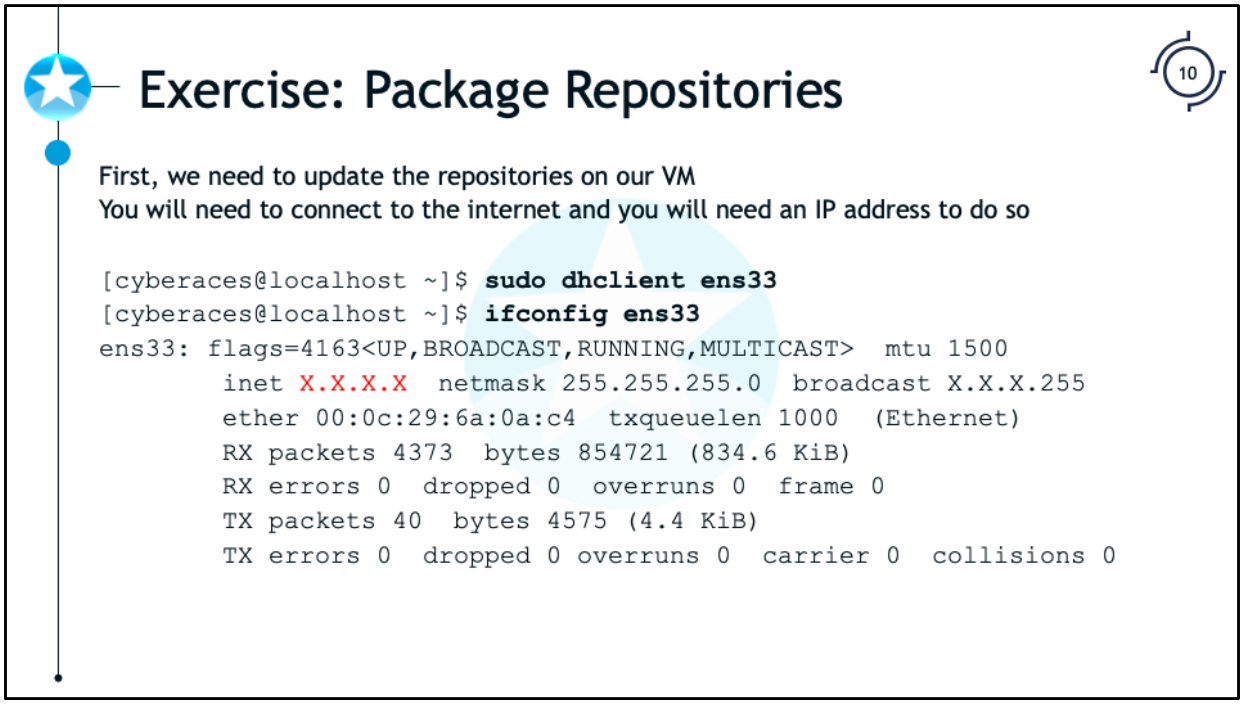
```
# rpm -Va | sort
```

The "V" means to verify packages, and the "a" means "all packages".



Linux vendors maintain online repositories of all of the software they've decided to include in their distribution. This means that if you did a default installation (which doesn't include every single package in the repository) and later decide that you need one of those packages, you can simply install it straight from the repo instead of having to locate it at a third-party site.

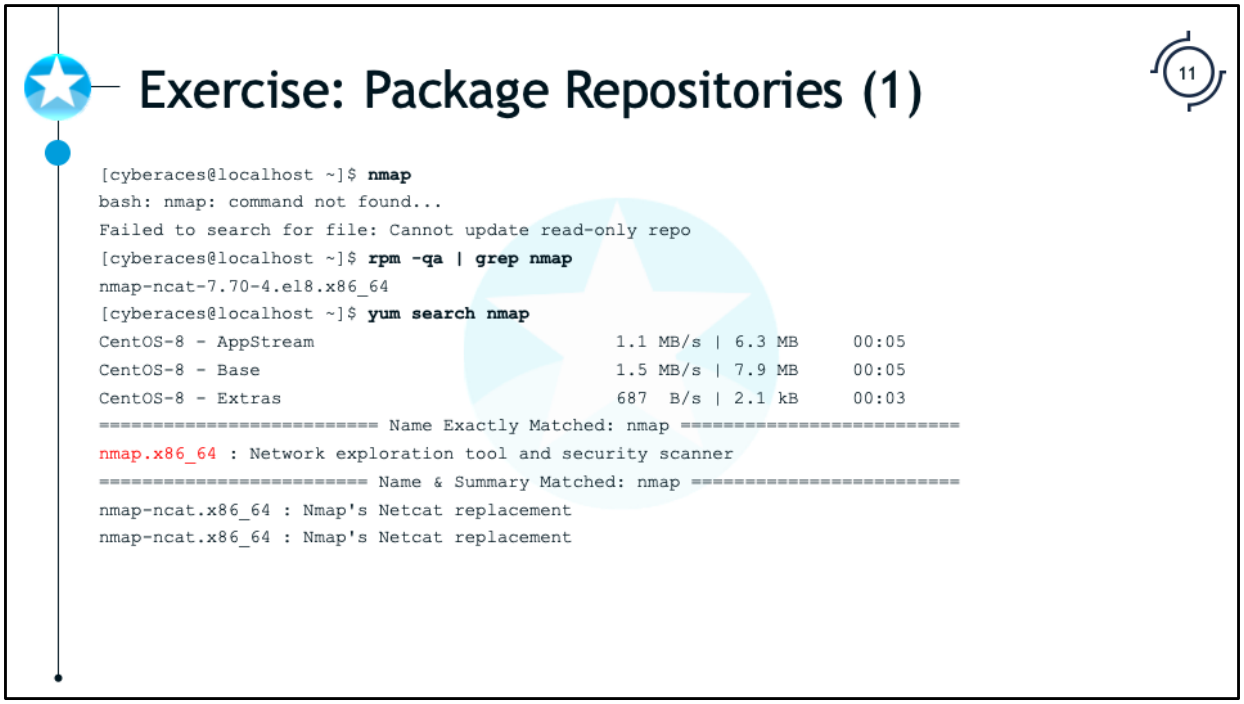
Linux distributions generally have tools for automating this process. On Red Hat systems, this tool is called "yum".



In this lab we are going install software on our system from the internet. To do so, you will need to obtain an IP address using the command below:

$ sudo dhclient ens33

The `sudo` is required to run the `dhclient` command as root. The `dhclient` executable will dynamically obtain an IP address on the `ens33` interface. In the text above, our place holder for our IP address is X.X.X.X, your IP address will be different. You should be able to get to the internet at this point.



1. Try to start a network scanning tool called nmap:

   $ **nmap**

   You should receive a "command not found" error.

2. Let's verify that it's not installed by querying the RPM database:

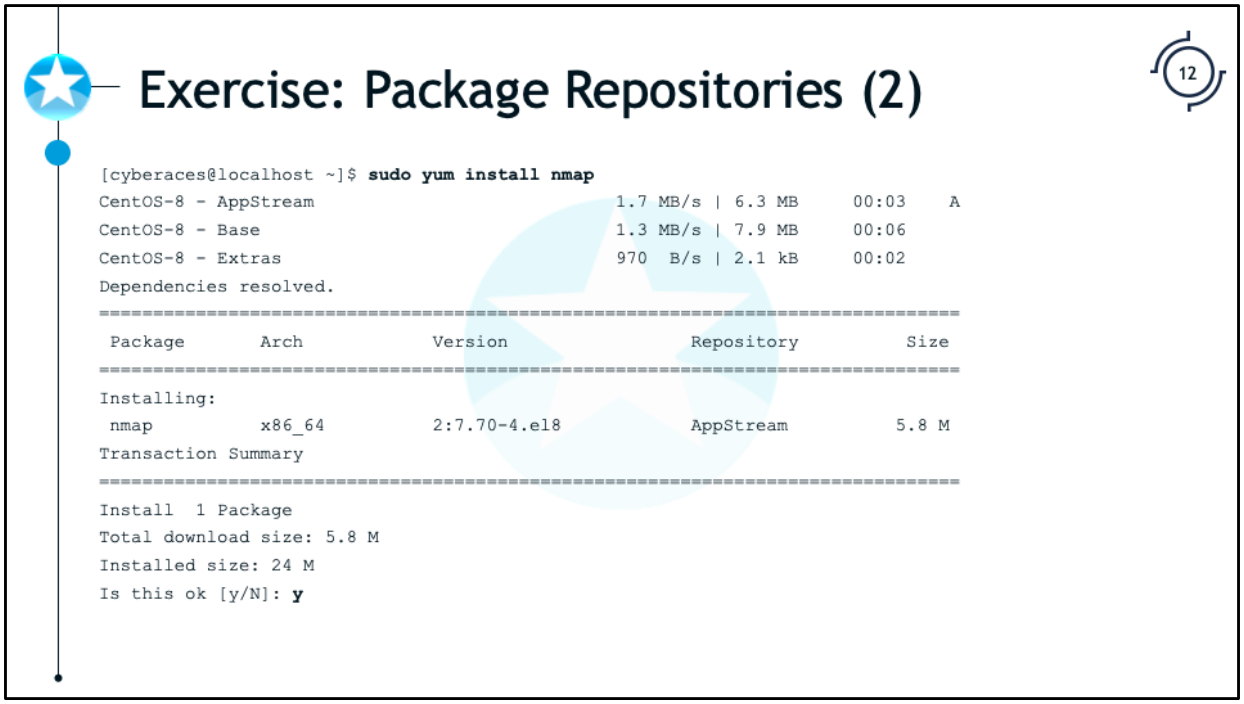
   $ **rpm -qa | grep nmap**

   There is output showing "nmap-ncat" with is Nmap's variant of Netcat called "Ncat". Nmap is not installed.

3. Use the "yum search" command to verify that the repositories have nmap:

   $ **yum search nmap**

   You should find one tool that matched exactly.

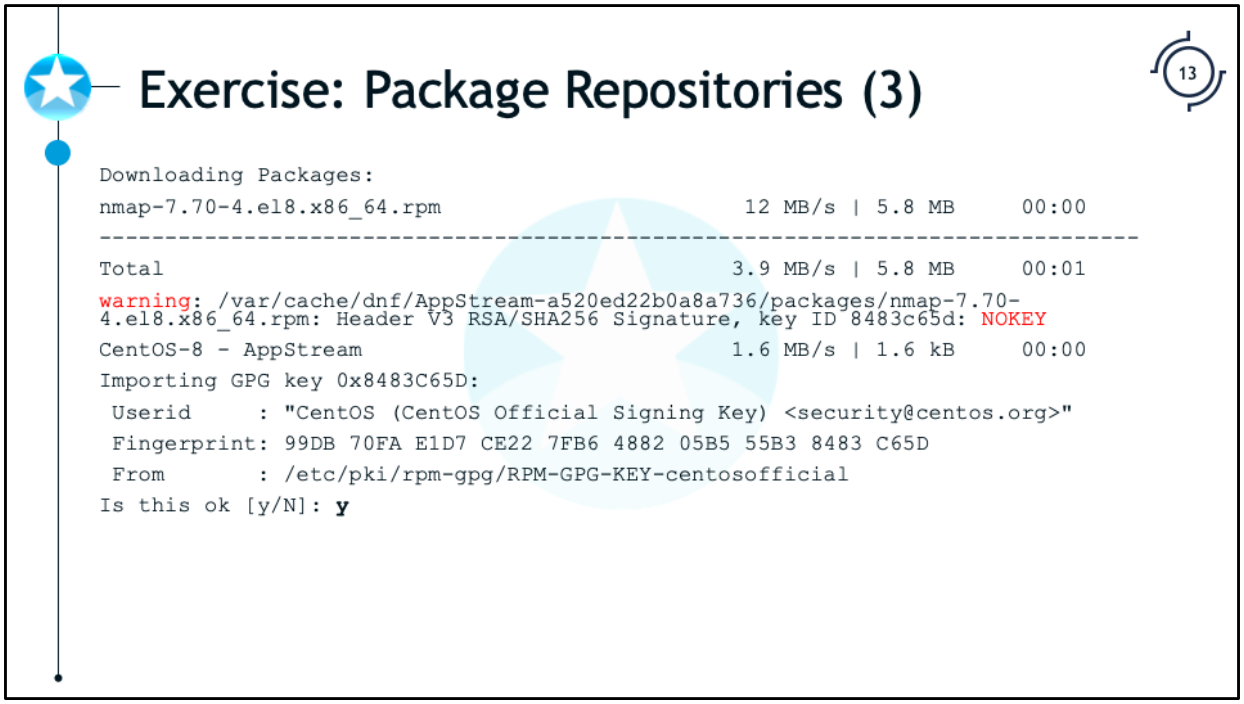


4. Use the "yum install" command to install the nmap RPM from the CentOS repositories:

   `$ sudo yum install nmap`

   Note: This also requires a working internet connection

5. You will be asked if this is okay before continuing; type "y" to continue. Note how it shows you a list of exactly what actions will be performed, including any dependencies that will be affected.



This is more of the output from the "yum install nmap" command on the previous slide.

6. You will be asked if you want to import a key; type "y" to continue. Note how it shows you a list of exactly what actions will be performed.

Exercise: Package Repositories (4)

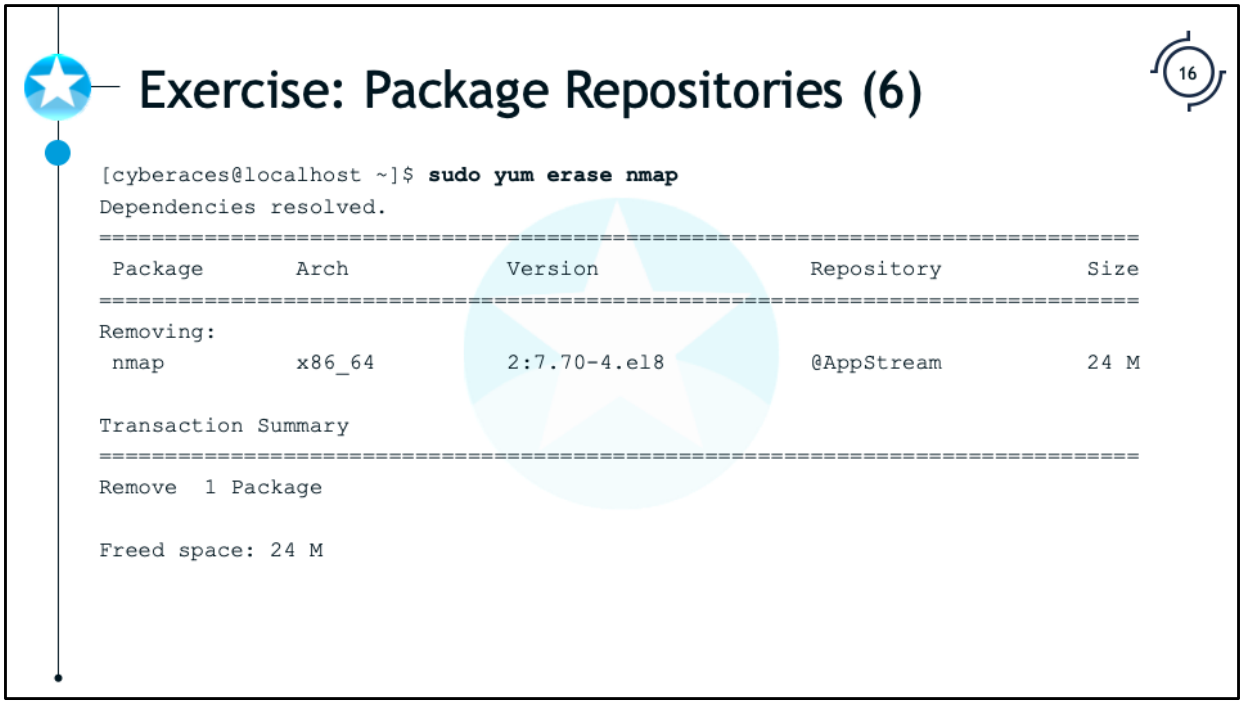

```
Key imported successfully
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                1/1
  Installing       : nmap-2:7.70-4.el8.x86_64                       1/1
  Running scriptlet: nmap-2:7.70-4.el8.x86_64                       1/1
  Verifying        : nmap-2:7.70-4.el8.x86_64                       1/1
Installed:
  nmap-2:7.70-4.el8.x86_64

Complete!
[cyberaces@localhost ~]$
```

This is the rest of the output from the "yum install nmap" command. It has now been successfully installed!

6. The nmap RPM is installed. Verify you can execute nmap:

   $ **nmap -h**

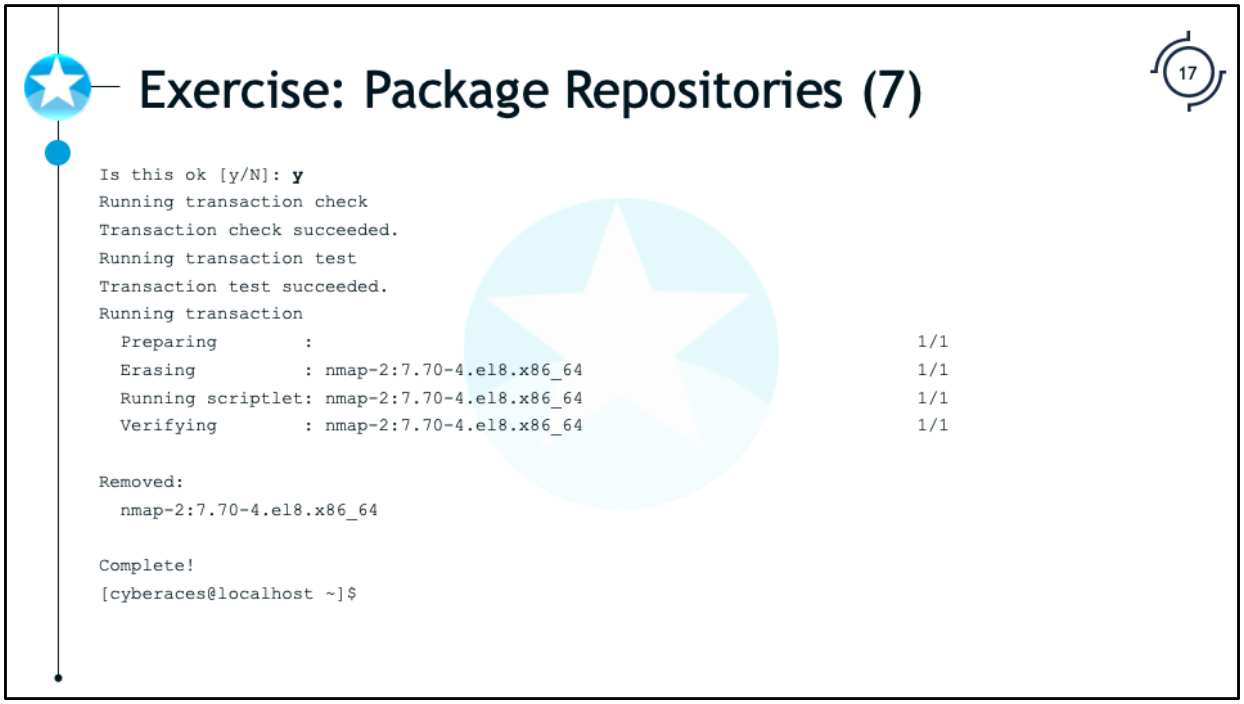
   You should see the usage options for nmap. This verifies that it was installed correctly.



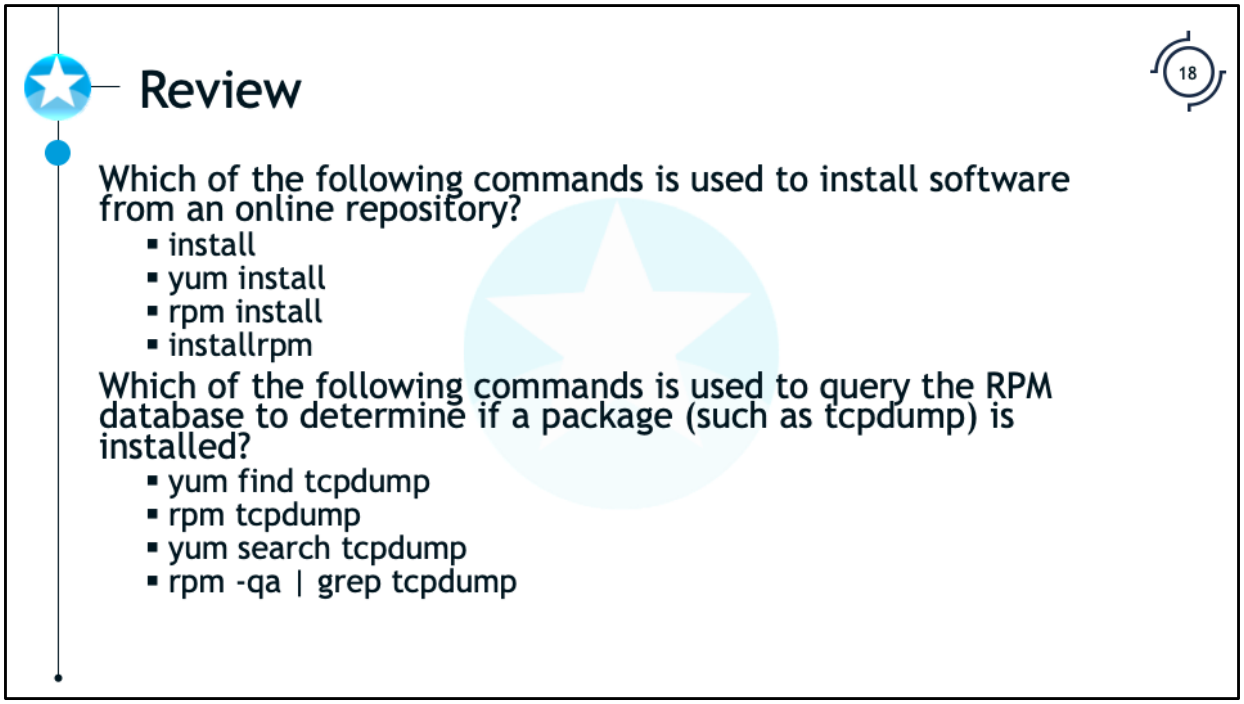
7. Remove the nmap RPM with the "yum erase" command:

   $ **sudo yum erase nmap**

   Note how it shows you a list of exactly what actions will be performed, including any dependencies that will be affected.



8. This is the rest of the output from the "sudo yum erase nmap" command. Note that when you are asked if this transaction is okay, you should type "y".

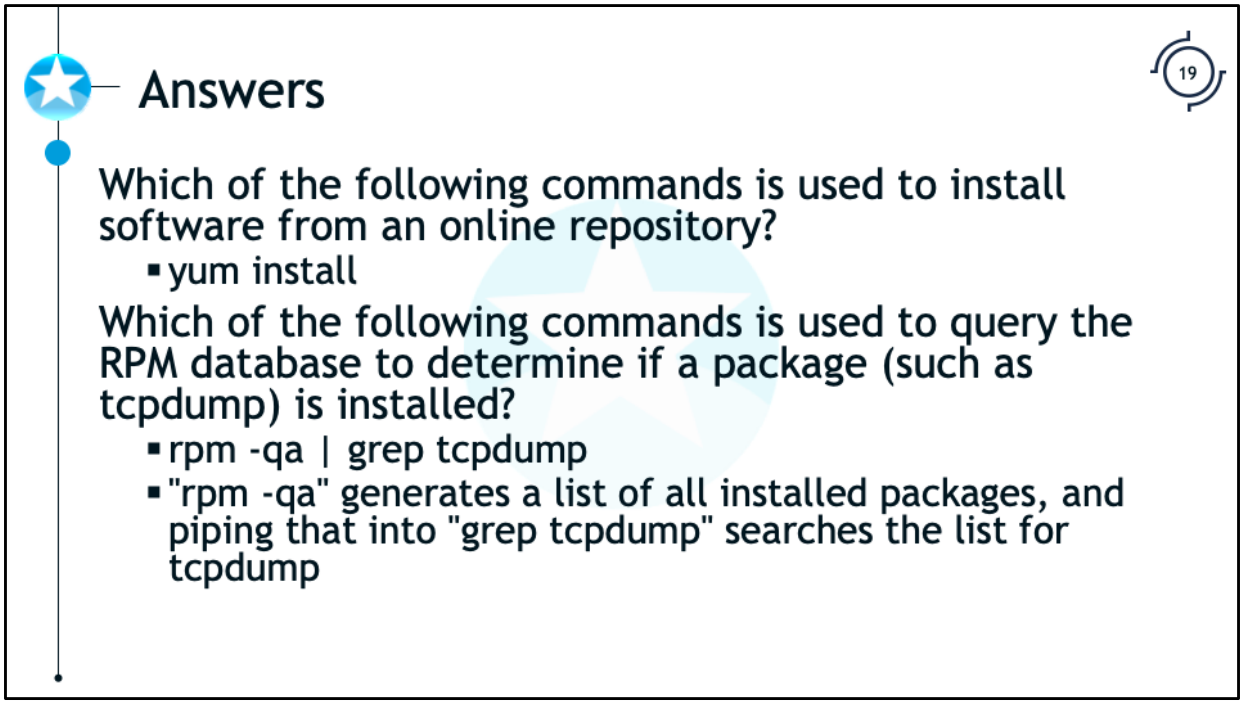


Which of the following commands is used to install software from an online repository?

- install
- yum install
- rpm install
- installrpm

Which of the following commands is used to query the RPM database to determine if a package (such as tcpdump) is installed?

- yum find tcpdump
- rpm tcpdump
- yum search tcpdump
- rpm -qa | grep tcpdump



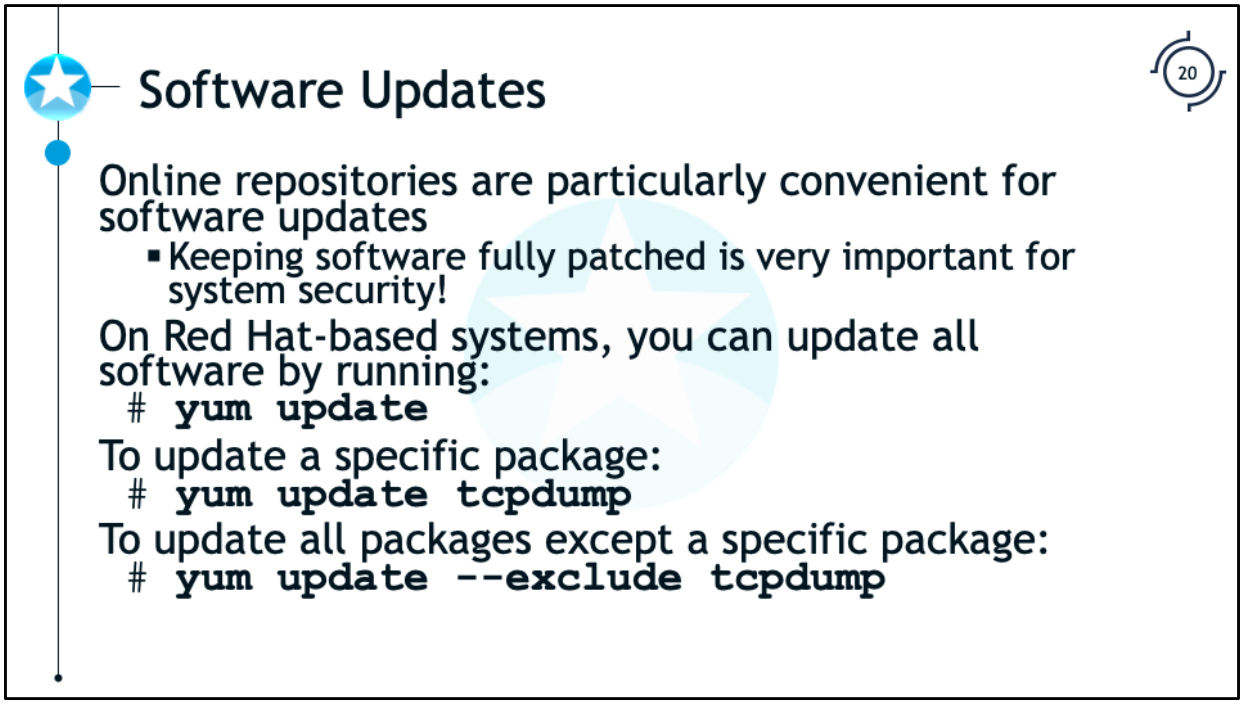
Which of the following commands is used to install software from an online repository?

yum install

Which of the following commands is used to query the RPM database to determine if a package (such as tcpdump) is installed?

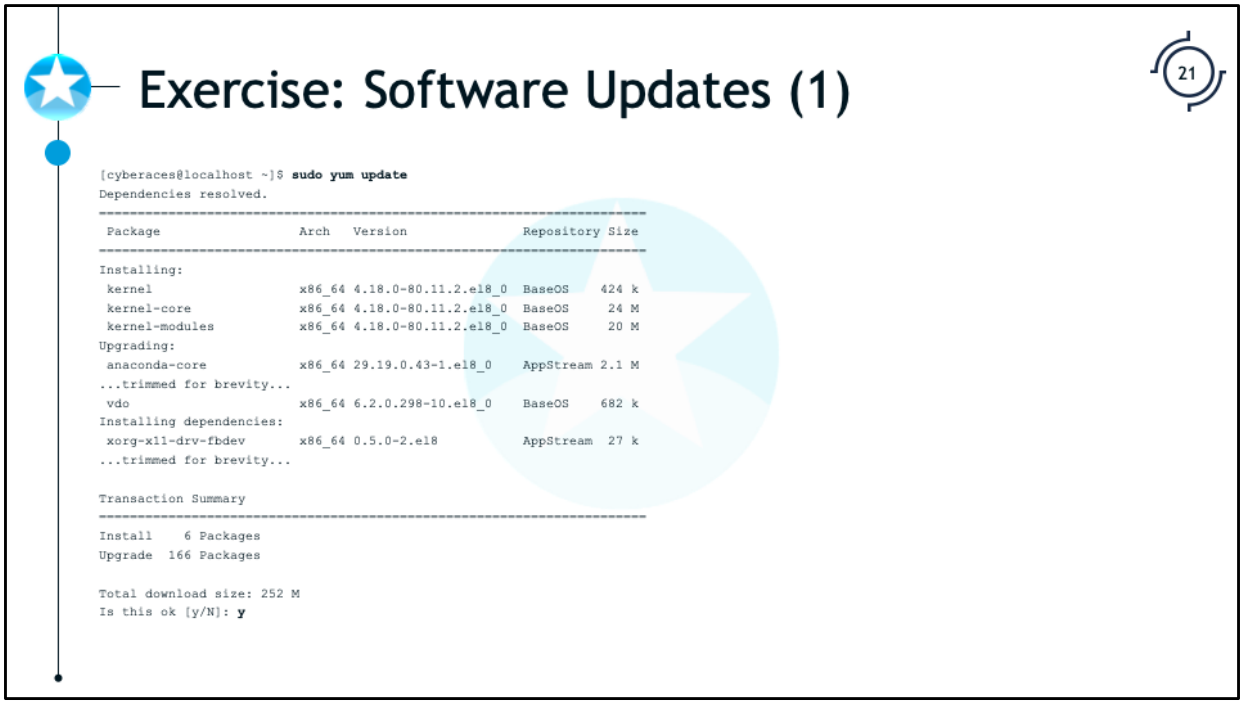
rpm -qa | grep tcpdump

"rpm -qa" generates a list of all installed packages, and piping that into "grep tcpdump" searches the list for tcpdump



Distribution repositories are not only handy for installing new software, but also for keeping your existing software updated. We all know that we have to keep our software updated so that we minimize our vulnerabilities and therefore reduce our risk of compromise. Keeping all of your repo software updated is as simple as running "yum update". You can also update just a specific package by specifying it's name, or leave out certain packages by using the "--exclude" option ("-x" for short). You can also uses the * as a wildcard when specifying package names. For example, the following command would update all software except anything starting with "kernel":

```
# yum update --exclude kernel*
```

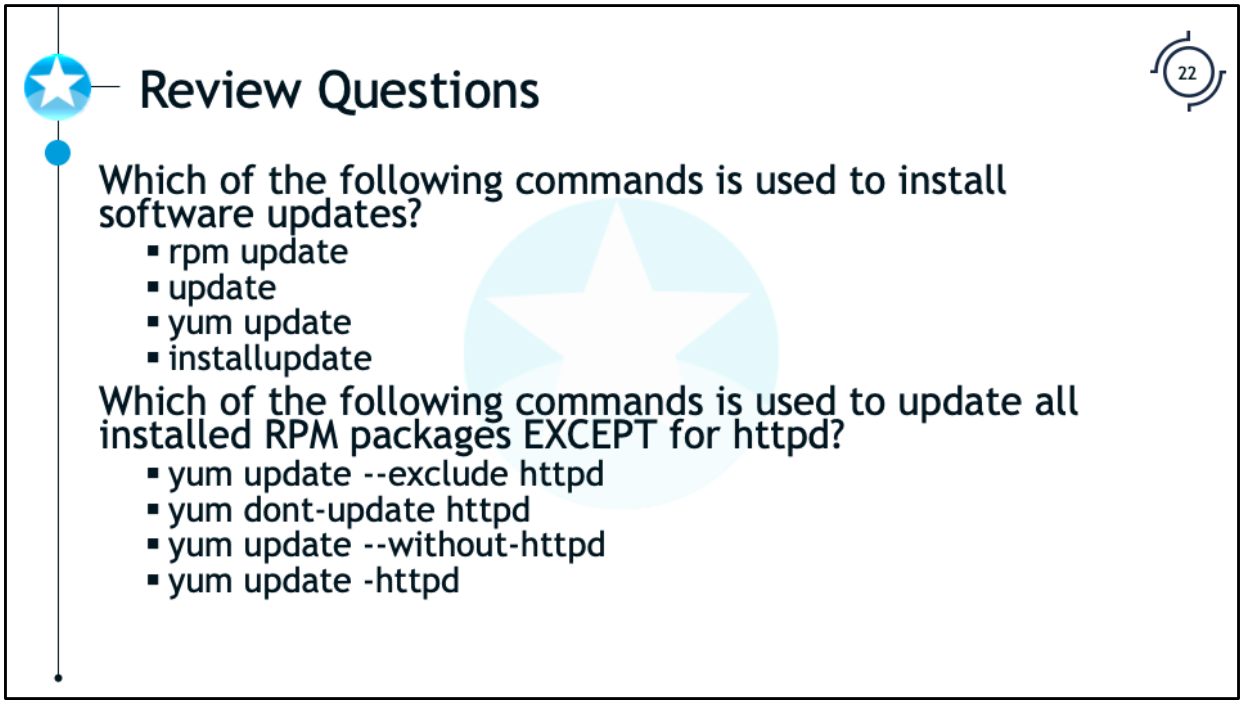


1. In your CentOS VM, open a terminal and search for updates using the command "yum update". You have to have root level access, so we prefix this command with "sudo". Note that this requires a working Internet connection.

   ```
   $ sudo yum update
   ```

2. After a few seconds, Yum will report that a certain number of packages need to be updated and ask if it's OK to continue. Answer "y" to continue installing the updates. This will take a while to complete and will download a large amount of data.

   After you have completed this step, you can close the terminal window.

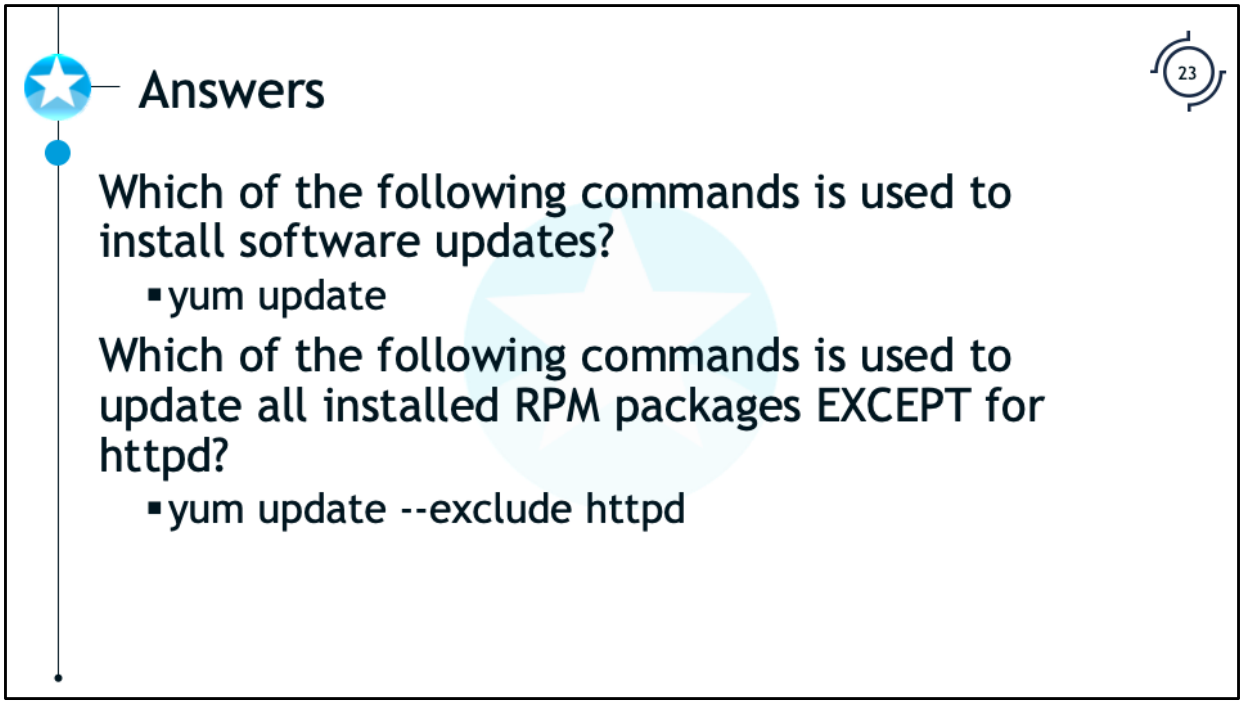


Which of the following commands is used to install software updates?

- rpm update
- update
- yum update
- installupdate

Which of the following commands is used to update all installed RPM packages EXCEPT for httpd?

- yum update --exclude httpd
- yum dont-update httpd
- yum update --without-httpd
- yum update -httpd

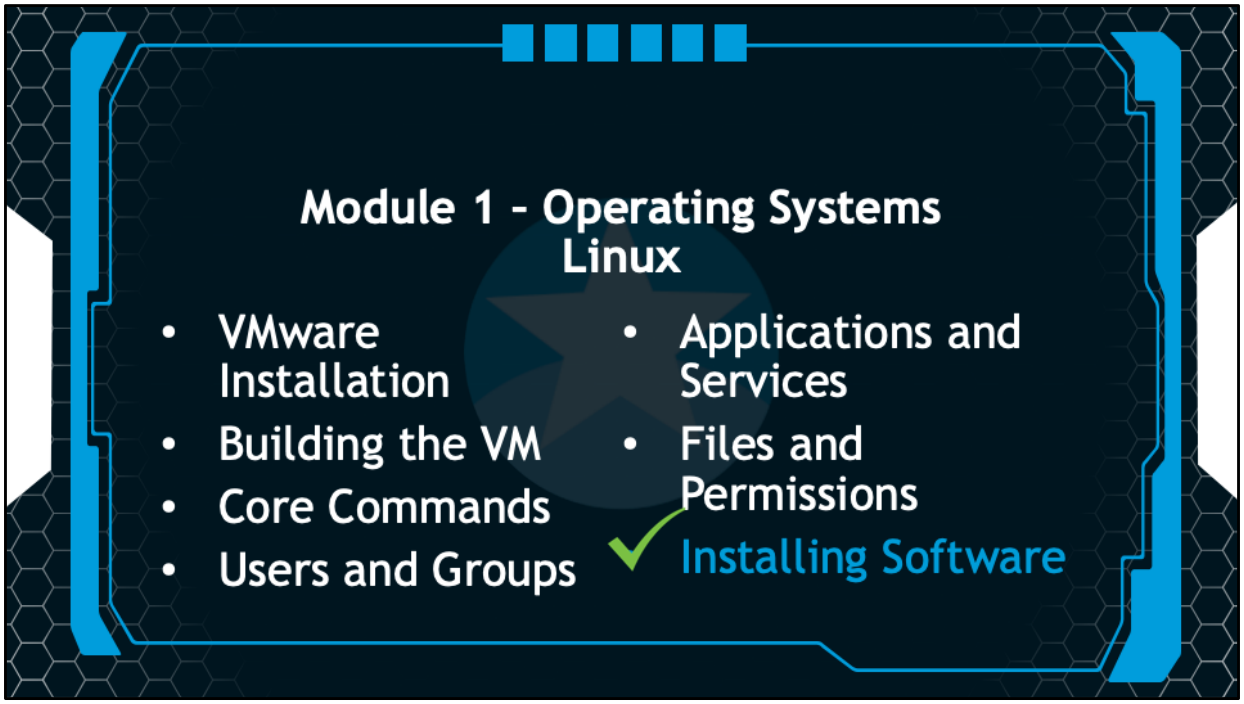


Which of the following commands is used to install software updates?

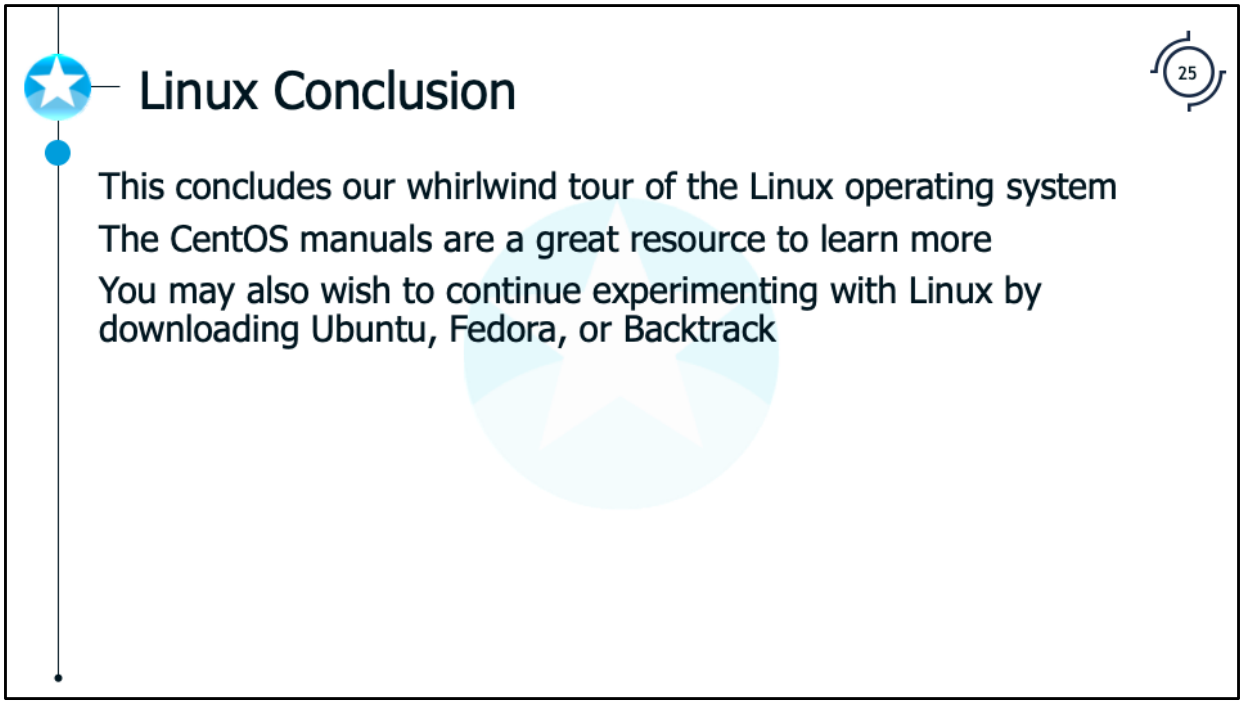
yum update

Which of the following commands is used to update all installed RPM packages EXCEPT for httpd?

yum update --exclude httpd



You have successfully completed the session on installing software in Linux. This was Session 6 of Module 1, Operating Systems – Linux.



This concludes our whirlwind tour of the Linux operating system. Once again, the CentOS manuals are very comprehensive and can be found here:

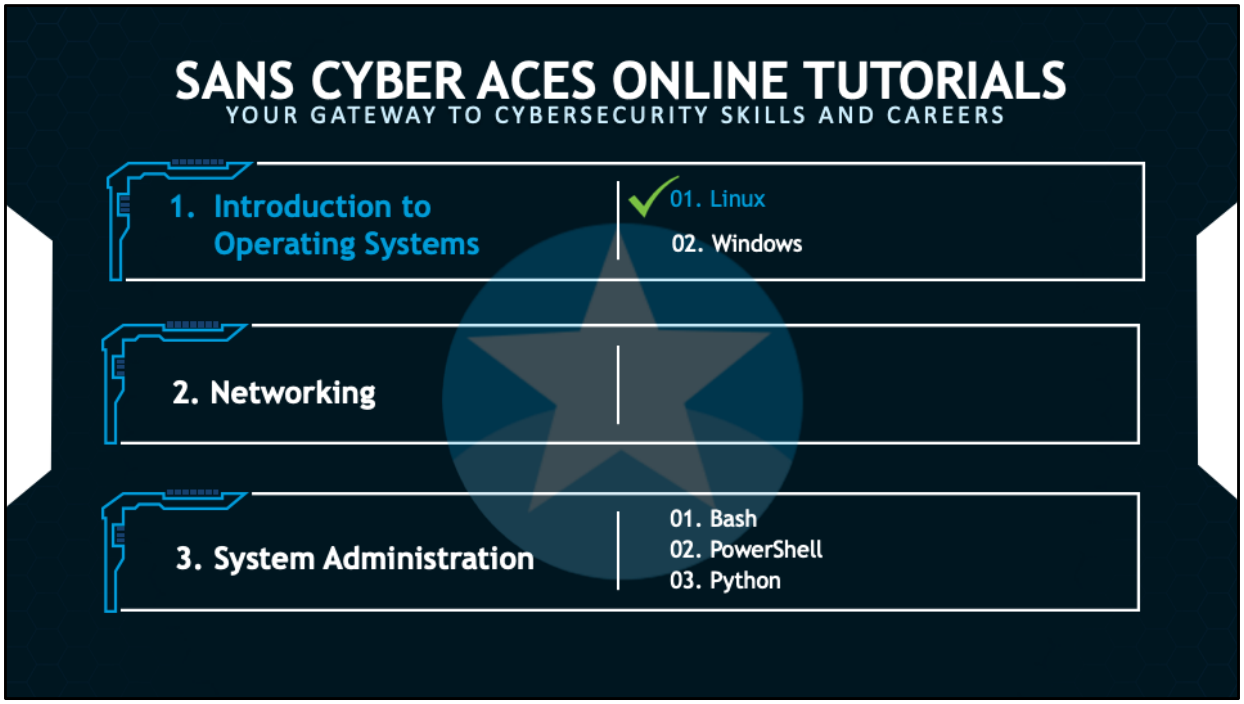
https://redsiege.com/ca/centos-guide

You may wish to continue experimenting with Linux by downloading other free distributions such as Ubuntu, Fedora, and Kali:

http://www.ubuntu.com/

http://fedoraproject.org/

http://www.kali.org/



You have successfully completed the Linux portion of Module 1, Introduction to Operating Systems. In the next series of sessions we will discuss Windows.