Software Management

> **Note**
>
> All labs rely on previous courseware and lab information.

Objectives

In this lab, you will:

- Update the Linux machine using the package manager
- Roll back or downgrade a previously updated package through the package manager
- Install the AWS Command Line Interface (AWS CLI)

Duration

This lab requires approximately **35 minutes** to complete.

AWS service restrictions

In this lab environment, access to AWS services and service actions might be restricted to the ones that you need to complete the lab instructions. You might encounter errors if you attempt to access other services or perform actions beyond the ones that this lab describes.

EN-US

Accessing the AWS Management Console

1. At the top of these instructions, choose Start Lab to launch your lab.
   A **Start Lab** panel opens, and it displays the lab status.

   > **Tip**: If you need more time to complete the lab, choose the Start Lab button again to restart the timer for the environment.

2. Wait until you see the message *Lab status: ready*, then close the **Start Lab** panel by choosing the **X**.

3. At the top of these instructions, choose AWS .
   This opens the AWS Management Console in a new browser tab. The system will automatically log you in.

   > **Tip**: If a new browser tab does not open, a banner or icon is usually at the top of your browser with a message that your browser is preventing the site from opening pop-up windows. Choose the banner or icon and then choose **Allow pop ups**.

4. Arrange the AWS Management Console tab so that it displays along side these instructions. Ideally, you will be able to see both browser tabs at the same time so that you can follow the lab steps more easily.

Task 1: Use SSH to connect to an Amazon Linux EC2 instance

In this task, you will connect to a Amazon Linux EC2 instance. You will use an SSH utility to perform all of these operations. The following instructions vary slightly depending on whether you are using Windows or Mac/Linux.

⊞ Windows Users: Using SSH to Connect

💬 These instructions are specifically for Windows users. If you are using macOS or Linux, skip to the next section.

5. Select the Details drop-down menu above these instructions you are currently reading, and then select Show . A Credentials window will be presented.

Task 2: Update your Linux machine

In this task, you use the yum package manager to update and upgrade the machine, including relevant security packages.

> **Note**
> You may have to use **sudo** to complete this task if you are not root.

21. To validate that you are in the **companyA** home folder, enter `pwd` and press Enter.

    If you are not in this folder, enter `cd companyA` and press Enter.

22. To query repositories for available updates, enter `sudo yum -y check-update` and press Enter.

23. To apply security-related updates, enter `sudo yum update --security` and press Enter.

24. To update packages, enter `sudo yum -y upgrade` and press Enter.

```
![Image shows results after running the sudo yum -y upgrade command. All available updates are listed.]
(images/upgrade.jpg)

*Figure: Once the sudo yum -y upgrade command is ran, the packages are updated and the system will let you
know that you are running the current updated version.*
```

```
>**Note**
>
>Your instance may already be up to date. If this is the case, you can still run through the commands for
practice.
```

25. To view the install of httpd and view the history of updates, enter `sudo yum install httpd -y` and press Enter.

```
[ec2-user@ip-10-0-10-163 companyA]$ sudo yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.48-2.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.48-2.amzn2 for package: httpd-2.4.48-2.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.48-2.amzn2 for package: httpd-2.4.48-2.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.48-2.amzn2.x86_64
```

Submit | Details ▾ | AWS | Start Lab | End Lab | --:-- | Grades

EN-US

25. To view the install of httpd and view the history of updates, enter `sudo yum install httpd -y` and press Enter.

```
[ec2-user@ip-10-0-10-163 companyA]$ sudo yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.48-2.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.48-2.amzn2 for package: httpd-2.4.48-2.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.48-2.amzn2 for package: httpd-2.4.48-2.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.48-2.amzn2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.48-2.amzn2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.48-2.amzn2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.48-2.amzn2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.48-2.amzn2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.48-2.amzn2.x86_64
--> Running transaction check
---> Package apr.x86_64 0:1.6.3-5.amzn2.0.2 will be installed
---> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amz
n2.0.2.x86_64
---> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
---> Package httpd-filesystem.noarch 0:2.4.48-2.amzn2 will be installed
---> Package httpd-tools.x86_64 0:2.4.48-2.amzn2 will be installed
---> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
---> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.1 will be installed
--> Running transaction check
---> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved
```

Figure: This command installs httpd and will also show a list of all previous updates and current packages on the instance.

Task 3: Roll back a package

In this task, you downgrade a package that has been updated through the yum package manager by doing the following:

- Using the yum history to list what has been installed and updated
- Rolling back to the most recent updates in the history list

> **Note**
>
> You may have to use **sudo** to complete this task if you are not root.

Task 3: Roll back a package

In this task, you downgrade a package that has been updated through the yum package manager by doing the following:

- Using the yum history to list what has been installed and updated
- Rolling back to the most recent updates in the history list

> **Note**
>
> You may have to use **sudo** to complete this task if you are not root.

26. To validate that you are in the **companyA** home folder, enter `pwd` and press Enter.
27. To view the history of updates, enter `sudo yum history list` and press Enter. In the output, under the **ID** column, make a note of the number for **EC2 ...** to use in the following steps in this task.

```
![Image shows results of sudo yum history-list](images/history-list.jpg)

*Figure: Once the sudo yum history-list command is finished running, two users will appear (ec2-user and
System) with the date, time, and actions that they did. It also shows how many files that were altered.*
```

```bash
[ec2-user@ companyA]$ sudo yum history list
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
ID  | Login user             | Date and time     | Action(s) | Altered
-------------------------------------------------------------------
2   | EC2 ... <ec2-user>     | <date and time>   | Install   | 9
1   | System <unset>         | <date and time>   | I, O, U   | 18
history list
```

28. To view the most recent set of updates, enter `sudo yum history info <#>` and replace <#> with the history list number

28. To view the most recent set of updates, enter `sudo yum history info <#>` and replace `<#>` with the history list number from the previous step. Once you have adjusted this command with this number, press Enter.

> *Note*:
> The number is found at the top of the history list from step 2.

```
![Results from sudo yum history info <#> command shows detailed information on the specified transaction number.](images/history-info.jpg)

*Figure: Information from the sudo yum history info <#> command shows the following information: begin time, begin rpmdb, end time, end rpmdb, user, return-code, and command line.*
```

```bash
[ec2-user@ companyA]$ sudo yum history info <#>
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Transaction ID  : <#>
Begin time      : <date and time>
Begin rpmdb     :
End time        : <time>
End rpmdb       :
User            : EC2 Default User <ec2-user>
Return-Code     : Success
Command Line    : install httpd -y
```

29. Enter `sudo yum -y history undo <#>` and replace `<#>` with the history list number from the previous steps. Once you have adjusted this command with this number, press Enter.

```
[ec2-user@ip-10-0-10-163 companyA]$ sudo yum -y history undo 2
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Undoing transaction 2, from Thu Aug 26 15:14:07 2021
```

29. Enter `sudo yum -y history undo <#>` and replace <#> with the history list number from the previous steps. Once you have adjusted this command with this number, press Enter.

Figure: Once the sudo yum -y history undo 2 command is ran, it now shows many packages as dep-install.

```bash
[ec2-user@ companyA]$ sudo yum -y history undo <#>

Loaded plugins: extras_suggestions, langpacks, priorities, update-motd

Undoing transaction <#>, from <date>

<list of actions now shown as Dep-Install>
```

Task 4: Install the AWS CLI on Red Hat Linux

In this task, you install the AWS CLI on Amazon Elastic Compute Cloud (Amazon EC2) Linux:

- Ensure that packages are installed and updated
- Install the AWS CLI

Note

Task 4: Install the AWS CLI on Red Hat Linux

In this task, you install the AWS CLI on Amazon Elastic Compute Cloud (Amazon EC2) Linux:

- Ensure that packages are installed and updated
- Install the AWS CLI

> **Note**
>
> You may have to use **sudo** to complete this task if you are not root.

30. To verify that Python is installed, enter the following command and press Enter:

```
python3 --version
```

The output indicates the version of Python that is installed.

> **Note**
>
> To install the AWS CLI, you must have Python 2 version 2.6.5 or later, or Python 3 version 3.3. If one of these versions is not already installed, you must follow the link steps to install Python.

31. To see if the pip package manager is already installed, enter the following command and press Enter.

```
pip3 --version
```

"bash: pip: command not found" indicate this Red Hat instance does *not* have pip installed.

> **Note**
>
> The primary distribution method for the AWS CLI on Linux, Windows, and macOS is pip. pip is a package manager for Python that provides you with an easy way to install, upgrade, and remove Python packages and their dependencies.

32. In order to install the AWS CLI, download the installation file using the **curl** command.

The -o option specifies the file name that the downloaded package is written to. The options on the following example

Submit   Details ▾   AWS   Start Lab   End Lab   --:--   Grades

EN-US

32. In order to install the AWS CLI, download the installation file using the **curl** command.

The `-o` option specifies the file name that the downloaded package is written to. The options on the following example command write the downloaded file to the current directory with the local name `awscliv2.zip`.

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
```

33. Unzip the installer.

The following example command unzips the package and creates a directory named `aws` under the current directory.

```
unzip awscliv2.zip
```

34. Run the install program.

The installation command uses a file named `install` in the newly unzipped `aws` directory. By default, the files are all installed to `/usr/local/aws-cli`, and a symbolic link is created in `/usr/local/bin`. The command includes `sudo` to grant write permissions to those directories.

```
sudo ./aws/install
```

35. To verify that the AWS CLI is now working, enter the following command and press Enter.

```
aws help
```

The **help** command should display the help information for the AWS CLI.

36. At the `:` prompt, enter `q` to exit.

37. At the top of the page above these instructions, choose the **Details** dropdown menu, and then chose **Show**. A **Credentials** window opens.

38. In the **Credentials** window next to **AWS CLI**, choose **Show**. This option displays AWS CLI credentials, including the **aws_access_key_id** and **aws_secret_access_key**. Copy and paste these two keys into a text editor to use in the next task of this lab.

38. In the **Credentials** window next to **AWS CLI**, choose **Show**. This option displays AWS CLI credentials, including the **aws_access_key_id** and **aws_secret_access_key**. Copy and paste these two keys into a text editor to use in the next task of this lab.

> **Note**
>
> There is no way for you to retrieve the secret access key that must be used when configuring the AWS CLI unless it was captured at the time that the keys were created. Fortunately, the secret access key was captured when it was created for this lab.

Task 5: Configure the AWS CLI to connect to your AWS account

39. Return to your terminal window. Enter the following configuration command for the AWS CLI and press Enter:

```
aws configure
```

40. At the prompts, enter the following information:

   - For the **AWS Access Key ID**, leave blank and press Enter.
   - For the **AWS Secret Access Key**, leave blank and press Enter.
   - For the **Default region name**, enter `us-west-2` and press Enter.
   - For the **Default output format**, enter `json` and press Enter.

     After the information is entered, the appropriate credential files are created automatically.

41. To open the credential file, enter the command `sudo nano ~/.aws/credentials`

42. Now paste the entire section copied from the Details window from task 4 into the file.

   For example:

```
[default]
aws access key id  group access key ID.
```

42. Now paste the entire section copied from the Details window from task 4 into the file.

For example:

```
[default]
aws_access_key_id=<your access key ID>
aws_secret_access_key=<your access key>
aws_session_token=<your session token>
```

43. Press `ctrl + o` to save and press enter to save the file as the original file name.

44. Press `ctrl + x` to exit the file.

45. Next, you need to find your instance ID. At the top of your screen above these instructions, choose **AWS** to open the AWS Management Console in a new tab.

46. At the top of the console page in the **Search for service** search box, enter **EC2** and choose **EC2**.

47. In the **Resources** section, choose **Instances (running)**.

48. There is one instance called **Command Host**. Copy and paste the **Instance ID** for the **Command Host** into a text editor to use in the following step.

49. Return to your terminal, and enter the following command. Before you press Enter, replace *<i-1234567890abcdefg>* with the instance ID that you copied from the previous step. After you have adjusted the following command with your instance ID, press Enter. This command describes the instance attributes.

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdefg --attribute instanceType
```

The output should look like the following. (The output should include your instance ID instead of **<i-1234567890abcdefg>**.)

Output:

```
{
    "InstanceId": "i-1234567890abcdefg",
    "InstanceType": {
        "Value": "t3.micro"
```

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdefg --attribute instanceType
```

The output should look like the following. (The output should include your instance ID instead of **<i-1234567890abcdefg>**.)

Output:

```
{
    "InstanceId": "i-1234567890abcdefg"
    "InstanceType": {
        "Value": "t3.micro"
    }
}
```

Lab Complete 🎓

🏁 Congratulations! You have completed the lab.

50. Select End Lab at the top of this page and then select Yes to confirm that you want to end the lab.
    A panel will appear, indicating that "DELETE has been initiated... You may close this message box now."
51. Select the **X** in the top right corner to close the panel.

About the AWS component

Amazon EC2 provides a wide selection of *instance types* optimized to fit different use cases. Instance types comprise varying combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your applications. Each instance type includes one or more *instance sizes* so that you can scale your resources to the requirements of your target workload.

This lab uses a **t3.micro** instance, which should be selected by default. This instance type has 1 virtual CPU and 1 GiB of memory.