

Activity - Install and Configure the AWS CLI

Activity overview

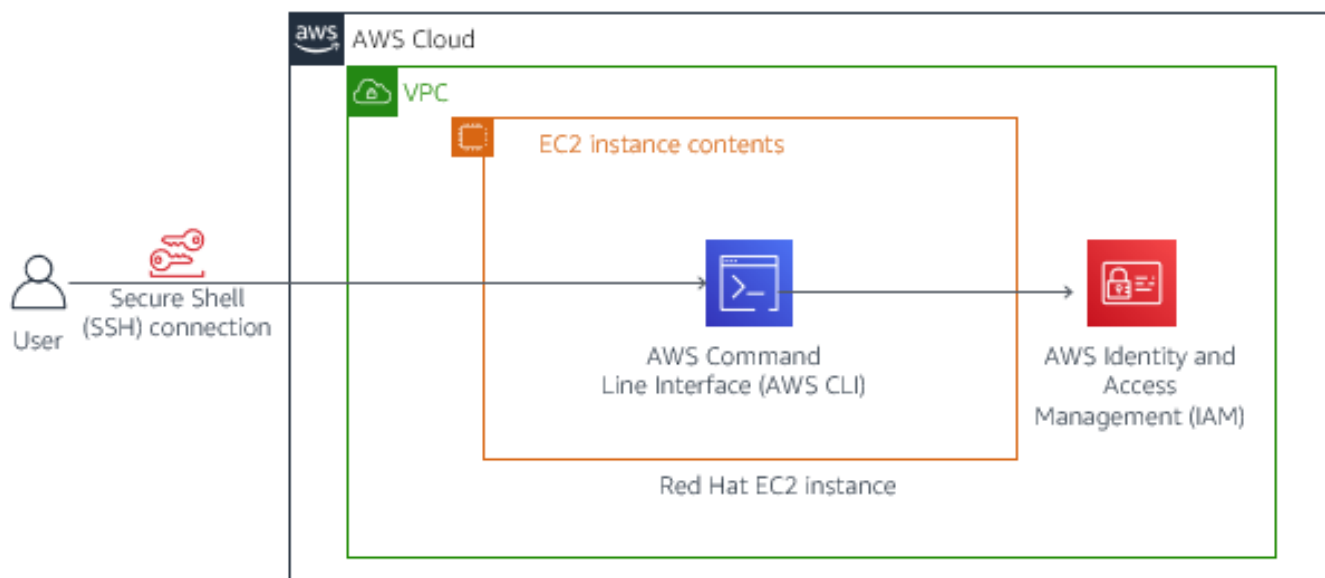
The *AWS Command Line Interface* (AWS CLI) is a command line tool that provides an interface for interacting with products and services from Amazon Web Services (AWS).

Often, people install the AWS CLI directly on their laptop machines. However, in this course—to ensure that all students have the same setup—you will practice using the AWS CLI from an Amazon Elastic Compute Cloud (Amazon EC2) instance.

While some instance types have the AWS CLI pre-installed on them (such as Amazon Linux instances), it is important to know how to install and configure this tool. Therefore, in this activity, you will practice installing the AWS CLI. A Red Hat Linux instance—that does not have the AWS CLI installed on it—is provided for you on Amazon EC2.

You will establish a Secure Shell (SSH) connection to the instance. Then, you will configure the installation with an access key that can connect to an AWS account. Finally, you will practice using the AWS CLI to interact with AWS Identity Access and Management (IAM).

This diagram summarizes the activities you will complete in this activity.



Activity objectives

After completing this activity, you will be able to:

- **Install** and configure the AWS CLI.
- **Connect** the AWS CLI to an AWS account.
- **Access** IAM by using the AWS CLI.

Business case relevance

Establishing AWS CLI Access for the Café





After Sofia discussed with Mateo, how to establish AWS CLI access to the Café AWS account, she is confident that she can install, configure, and start making use of the AWS CLI.

During this activity, you will take on the role of Sofia. You will install the AWS CLI, configure it, and practice using the AWS CLI to query details about the IAM service.

Activity steps

Duration: This activity requires approximately **45 minutes** to complete.

Accessing the AWS Management Console

1. At the top of these instructions, click to launch your lab.

A Start Lab panel opens displaying the lab status.

2. Wait until you see the message "**Lab status: ready**", then click the **X** to close the Start Lab panel.

3. At the top of these instructions, click

This will open 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, there will typically be a banner or icon at the top of your browser indicating that your browser is preventing the site from opening pop-up windows. Click on the banner or icon and 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, to make it easier to follow the lab steps.

Leave this browser tab open. You will return to it in Task 3.

Task 1: Connect to the Red Hat EC2 instance by using SSH

In this task, you will log in to an existing Amazon EC2 instance, which you will use to practice installing and using the AWS CLI.

Windows Users: Using SSH to Connect

 These instructions are for Windows users only.

If you are using macOS or Linux, [skip to the next section](#).

5. Read through the three bullet points in this step before you start to complete the actions, because you will not be able see these instructions when the Details panel is open.

- Click on the drop down menu above these instructions you are currently reading, and then click . A Credentials window will open.
- Click on the **Download PPK** button and save the **labsuser.ppk** file. Typically your browser will save it to the Downloads directory.
- Then exit the Details panel by clicking on the **X**.

6. Download needed software.

- You will use **PuTTY** to SSH to Amazon EC2 instances. If you do not have PuTTY installed on your computer, [download it here](#).

7. Open **putty.exe**

8. Configure PuTTY to not timeout:

- Click **Connection**
- Set **Seconds between keepalives** to

This allows you to keep the PuTTY session open for a longer period of time.

9. Configure your PuTTY session:

- Click **Session**
- **Host Name (or IP address)**: Copy and paste the **IPv4 Public IP address** for the instance. To find it, return to the EC2 Console and click on **Instances**. Check the box next to the instance you want to connect to and in the *Description* tab copy the **IPv4 Public IP** value.
- Back in PuTTY, in the **Connection** list, expand **SSH**
- Click **Auth** (don't expand it)
- Click **Browse**
- Browse to and select the lab#.ppk file that you downloaded
- Click **Open** to select it
- Click **Open**

10. Click **Yes**, to trust the host and connect to it.

11. When prompted **login as**, enter: `ec2-user`

This will connect you to the EC2 instance.

12. [Windows Users: Click here to skip ahead to the next task.](#)

macOS and Linux Users

These instructions are for Mac/Linux users only. If you are a Windows user, [skip ahead to the next task.](#)

13. Read through the three bullet points in this step before you start to complete the actions, because you will not be able see these instructions when the Details panel is open.

- Click on the `Details` drop down menu above these instructions you are currently reading, and then click `Show`. A Credentials window will open.
- Click on the **Download PEM** button and save the **labsuser.pem** file.
- Then exit the Details panel by clicking on the **X**.

14. Open a terminal window, and change directory `cd` to the directory where the labsuser.pem file was downloaded.

For example, run this command, if it was saved to your Downloads directory:

```
cd ~/Downloads
```

15. Change the permissions on the key to be read only, by running this command:

```
chmod 400 labsuser.pem
```

16. Return to the AWS Management Console, and in the EC2 service, click on **Instances**. Check the box next to the instance you want to connect to.

17. In the *Description* tab, copy the **IPv4 Public IP** value.

18. Return to the terminal window and run this command (replace **<public-ip>** with the actual public IP address you copied):

```
ssh -i labsuser.pem ec2-user@<public-ip>
```

19. Type `yes` when prompted to allow a first connection to this remote SSH server.

Because you are using a key pair for authentication, you will not be prompted for a password.

Task 2: Install the AWS CLI on Red Hat Linux

Now that you are connected to the instance, the next step is to install the AWS CLI.

For all steps that are in this section: Complete these steps in the terminal window where you have an active SSH connection to the Red Hat Linux instance running on Amazon EC2.

20. Verify that Python is installed by running the following command:

```
python --version
```

The command output shows that Python version 2.7.5 is installed.

Tip: 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 was not already installed*, you must follow the steps to install Python as documented [here](#).

21. Run the following command to see if the pip package manager is already installed.

```
pip --version
```

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 an easy way to install, upgrade, and remove Python packages and their dependencies.

22. Download and install the Extra Packages for Enterprise Linux (EPEL) repository, and then install pip by using the following commands:

```
sudo yum install -y wget
wget http://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
sudo rpm -ivh epel-release-latest-7.noarch.rpm
sudo yum install -y python-pip
```

23. Verify that pip is now installed.

```
pip --version
```

This time, the command output should return a line that indicates the version that was installed (e.g., 8.1.2).

24. Install the AWS CLI by using pip.

```
pip install awscli --upgrade --user
```

NOTE: The `--upgrade` option tells pip to upgrade any requirements that are already installed. The `--user` option tells pip to install the program to a subdirectory of your user directory, which helps you avoid modifying the libraries that are used by your operating system.

The installation should succeed.

NOTE: For the purposes of this activity, *do not upgrade pip*, even if the command output suggests that you do so.

25. Verify that AWS CLI is now working by running the following command:

```
aws help
```

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


26. At the `:` prompt, type `q` to exit.

Task 3: Observe IAM configuration details in the AWS Management Console

27. Back in the **AWS Management Console** browser tab, click **Services**, and then in the **Find a service...** search bar, type `IAM` to find the IAM service. Select it in the result set.

Note: The IAM page that displays will contain messages indicating that you do not have permission to observe some IAM service details. You can safely ignore these messages.

28. In the IAM service page, in the left navigation pane, click **Users**, and then click the **awsstudent** user, which will be a hyperlink.

29. Click the arrow icon next to **lab_policy**, and then click  **JSON** to see the policy document, which is formatted in JavaScript Object Notation (JSON).

Notice that this IAM policy grants the **awsstudent** user access to specific AWS services in this account.

30. Navigate back to **awsstudent** under **Users** and click the **Security credentials** tab.

Notice that the access key was already created for the **awsstudent** user.

The **Access key ID** is displayed. There is no way to retrieve the *secret access key* that must be used with it, unless it was captured at the time that the key was created. Fortunately, the secret access key was captured when it was created, and it is available in the **Details** dropdown menu above. You will use these credentials in the next section of this activity.

Task 4: Configure the AWS CLI to connect to your AWS Account

31. Return to the active SSH session terminal window and run the configuration command for AWS CLI as follows:

```
aws configure
```

32. At the prompts, enter the following information:

- **AWS Access Key ID:** Click on the `Details` drop down menu above these instructions, and then click `Show`. Copy the **AccessKey** value and paste it into the terminal window.
- **AWS Secret Access Key:** Copy and paste the **SecretKey** value from the same Credentials screen.
- **Default region name:** `eu-west-2`
- **Default output format:** `json`

Task 5: Observe IAM configuration details by using the AWS CLI

33. In the terminal window, test the IAM configuration by running this command:

```
aws iam list-users
```

If the test is successful, you should see a JSON structured response that shows a list of all the IAM users in the account. The result should match what you saw in the AWS Management Console earlier in this activity.

34. Open the **AWS CLI Command Reference** [documentation page](#) for the `iam` command.

35. Scroll down to the list of **Available Commands**.

Activity 1 challenge

Use the AWS CLI Command Reference documentation to figure out how you can use the AWS CLI to download the **lab_policy** JSON-formatted IAM policy document. You saw this policy document in the AWS Management Console earlier in this activity.

Avoid the temptation to use the AWS Management Console. See if you can do this challenge by using only the AWS CLI.

It might take some experimentation to figure out this challenge because the solution is not a simple single command. Have patience, and work with other classmates if it helps!

Tips to help you complete the challenge

- **Tip #1:** In the **AWS CLI Command Reference** documentation page, click the hyperlink of any command you might want to use. You can see what information the command will return, and also see details on how to use the command.|
- **Tip #2:** Try using the `list-policies` command. Set a *scope* to help filter the results.|
- **Tip #3:** You will need to run more than one command to successfully complete this challenge. Some of the output from one command might be required in order to successfully run the next command. For example, you need to know the lab_policy's *Arn* before you can successfully run the `get-policy` command.|
- **Tip #4:** Before you can get the actual JSON representation of the IAM policy to display, you will need to know the policy version. Return to the list of available commands in the documentation to see if any of them look like they might return a policy document.|
- **Tip #5:** Do not forget that you can pipe any terminal output to a new file, by using the `>` command. This could be useful for creating the lab_policy.json file you will turn in at the end of this challenge.|

Activity summary

You successfully installed the AWS CLI on a machine and connected it to an AWS account. You then practiced using the AWS CLI and referencing the AWS CLI Command Reference documentation to look up useful command details.

Key takeaways:

- Anything that you can do in the AWS Management Console can also be done through the AWS CLI.
- You would typically provide a *user name* and *password* to connect to the AWS Management Console, but you needed an *access key ID* and *secret access key* to connect to the same account using the AWS CLI.

In later activities in this course, you will continue to use the AWS CLI.

Tip: If you want to use an AWS CLI installation to connect to a different AWS account, run the `aws configure` command again, and provide the new credentials.

Lab Complete

36. Click `End Lab` at the top of this page and then click **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."

37. Click the **X** in the top right corner to close the panel.