

The Bash Shell

Note

All labs rely on previous courseware and lab information.

Objectives

In this lab, you will:

- Create and work with an alias to backup a complete folder
- Work the PATH variable and add a new folder to it

Duration

This lab requires approximately **30 minutes** to complete.

AWS service restrictions

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

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.
6. Select the **Download PPK** button and save the **labsuser.ppk** file.
Typically your browser will save it to the Downloads directory.
7. Make a note of the **PublicIP** address.
8. Then exit the Details panel by selecting the **X**.
9. Download **PuTTY** to SSH into the Amazon EC2 instance. If you do not have PuTTY installed on your computer, [download it here](#).
10. Open **putty.exe**
11. Configure PuTTY timeout to keep the PuTTY session open for a longer period of time.:
 - Select **Connection**
 - Set **Seconds between keepalives** to **30**
12. Configure your PuTTY session:
 - Select **Session**
 - **Host Name (or IP address)**: Paste the **Public DNS or IPv4 address** of the instance you made a note of earlier.
Alternatively, return to the EC2 Console and select **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**
 - Select **Auth** (*don't expand it*)
 - Select **Browse**
 - Browse to and select the lab#.ppk file that you downloaded
 - Select **Open** to select it
 - Select **Open** again.
13. Select **Yes**, to trust and connect to the host.
14. When prompted **login as**, enter: **ec2-user**
This will connect you to the EC2 instance.

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

macOS and Linux Users

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

16. Select the `Details` drop-down menu above these instructions you are currently reading, and then select `Show`. A Credentials window will be presented.
17. Select the **Download PEM** button and save the **labsuser.pem** file.
18. Make a note of the **PublicIP** address.
19. Then exit the Details panel by selecting the **X**.
20. Open a terminal window, and change directory `cd` to the directory where the *labsuser.pem* file was downloaded. For example, if the *labsuser.pem* file was saved to your Downloads directory, run this command:

```
cd ~/Downloads
```

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

```
chmod 400 labsuser.pem
```

22. Run the below command (*replace **<public-ip>** with the **PublicIP** address you copied earlier*). Alternatively, return to the EC2 Console and select **Instances**. Check the box next to the instance you want to connect to and in the *Description* tab copy the **IPv4 Public IP** value.:

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

23. Type `yes` when prompted to allow the 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: Create an alias for a backup operation

Specifically, you need to create an alias that gives you the ability to back up whatever path you provide it.

Helpful Hint

You may have to use `sudo` to complete this task if you are not root.

Create an alias that uses the `tar` to back up the second parameter provided into the first parameter. The following is a command line example:

Usage example: `backup "fileToSaveTo.tar.gz" "pathToBackUp"`

24. To validate that you are in the home folder in the terminal, enter the following command, and press Enter.

```
pwd
```

Expected output:

```
[ec2-user@ ~]$ pwd  
/home/ec2-user/
```

25. To create an alias called **backup**, enter the following command, and press Enter.

```
alias backup='tar -cvzf '
```

Remember that **tar** is a command that you use to create or extract an archive that contains files and folders.

- **-f** archives the files (**tar** can also archive devices).
- **-v** is the everbose option to display what is put into the archive.
- **-z** compresses the archive into the .gzip format.
- **tar -cf** would work perfectly but would not display what is inside the archive and would not compress it.

26. To use the **backup** alias to back up the **CompanyA** folder, enter the following command, and press Enter.

```
backup backup_companyA.tar.gz CompanyA
```

Expected output:

```
[ec2-user@ ~]$ backup backup_companyA.tar.gz CompanyA  
CompanyA/  
CompanyA/Management/  
CompanyA/Management/Sections.csv  
CompanyA/Management/Promotions.csv  
CompanyA/Employees/  
CompanyA/Employees/Schedules.csv  
CompanyA/Finance/  
CompanyA/Finance/Salary.csv  
CompanyA/HR/  
CompanyA/HR/Managers.csv  
CompanyA/HR/Assessments.csv  
CompanyA/IA/  
CompanyA/SharedFolders/  
CompanyA/bin/  
CompanyA/bin/hello.sh  
[ec2-user@ ~]$
```

Figure: The command `backup backup_companyA.tar.gz CompanyA` shows the entire directory of Company A. The contents include folders such as Management, Employees, Finance, HR, IA, SharedFolders, and bin. Files include the following: Sections.csv, Promotions.csv, Schedules.csv, Salary.csv, Managers.csv, Assessments.csv, and hello.sh.

```
[ec2-user@ ~]$ backup backup_companyA.tar.gz CompanyA
CompanyA/
CompanyA/Management/
CompanyA/Management/Sections.csv
CompanyA/Management/Promotions.csv
CompanyA/Employees/
CompanyA/Employees/Schedules.csv
CompanyA/Finance/
CompanyA/Finance/Salary.csv
CompanyA/HR/
CompanyA/HR/Managers.csv
CompanyA/HR/Assessments.csv
CompanyA/IA/
CompanyA/SharedFolders/
CompanyA/bin/hello.sh
```

27. To verify that the archive is created, enter the `ls` command, and press Enter.

Expected output:

```
[ec2-user@ ~]$ ls
backup_companyA.tar.gz  CompanyA
```

Task 3: Explore and update the PATH environment variable

In this task, you display the PATH environment variable. You then update the variable and add a new directory, in which you can place executables.

28. To navigate to the **bin** folder in the home **CompanyA** directory, enter the following command, and press Enter.

```
cd /home/ec2-user/CompanyA/bin
```

Note:

You can also use the `pwd` command to verify that you are in the home folder, `/home/ec2-user`, and use `cd CompanyA/bin` to enter the `/home/ec2-user/bin` folder.

29. To run the **hello.sh** script, enter the following command, and press Enter.

```
./hello.sh
```

Expected Output:

```
[ec2-user@ bin]$ hello.sh  
hello ec2-user
```

30. To navigate to the parent folder, enter the following command, and press Enter.

```
cd ..
```

Expected Output:

```
[ec2-user@ bin]$ cd ..  
[ec2-user@ CompanyA]$
```

31. To run the **hello.sh** script again, enter the following command, and press Enter.

```
./bin/hello.sh
```

Expected Output:

```
[ec2-user@ CompanyA]$ ./bin/hello.sh  
hello ec2-user
```

32. To run the **hello.sh** script, enter the following command, and press Enter.

```
hello.sh
```

Expected Output:

```
[ec2-user@ CompanyA]$ hello.sh  
  
bash: hello.sh: command not found
```

Note:

Analyze the three different ways you tried to run the hello.sh script. In the next step, you are going to figure out why the third run failed and how to solve this issue.

33. To display the value of the **PATH** variable, enter the following command, and press Enter.

```
echo $PATH
```

Expected Output:

```
[ec2-user@ CompanyA]$ echo $PATH  
/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/ec2-  
user/.local/bin:/home/ec2-user/bin
```

Note:

Remember that the PATH variable is a list of folders where the system looks for executables and libraries. If you enter a command such as **hello.sh**, Linux will look for the script in the current folder and then in all the folders contained in the PATH variable. You can see that **/home/ec2-user/bin** is not listed. There are currently only two ways to run the script:

- Navigate to the **/home/ec2-user/CompanyA/bin** folder, and enter `hello.sh` to run the script.
- Enter `/home/ec2-user/CompanyA/bin/hello.sh` from any folder.

34. To add the **/home/ec2-user/CompanyA/bin** folder to the PATH variable, enter the following command, and press Enter.

```
PATH=$PATH:/home/ec2-user/CompanyA/bin
```

Expected Output:

```
[ec2-user@ CompanyA]$ PATH=$PATH:/home/ec2-user/CompanyA/bin
```

35. To try to run the **hello.sh** script again, enter `hello.sh` and press Enter.

Expected Output:

```
[ec2-user@ CompanyA]$ hello.sh
hello ec2-user
```

Lab Complete

Congratulations! You have completed the lab.

36. 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."

37. 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.

Additional resources

- [Amazon EC2 Instance Types](#)
- [Amazon Machine Images \(AMI\)](#)
- [Status Checks for Your Instances](#)
- [Amazon EC2 Service Quotas](#)
- [Terminate Your Instance](#)

For more information about AWS Training and Certification, see <https://aws.amazon.com/training/>.

Your feedback is welcome and appreciated.

If you would like to share any suggestions or corrections, please provide the details in our [AWS Training and Certification Contact Form](#).

© 2022 Amazon Web Services, Inc. and its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited.