Maintaining Efficient Process Utilization on Your Computer in Linux

**Introduction**

In this lab, you'll use the new commands you learned to do some process maintenance on a Linux virtual machine.

**What you’ll do**

* Terminate a specific process
* Terminate multiple processes

**You will have 60 minutes to complete this lab.**

# **Maintaining Efficient Process Utilization on Linux**

External IP address



content\_copy

username



content\_copy

[file\_download](https://googlecoursera.qwiklabs.com/lab_instances/download_pem/15718899" \t "_blank)

[Download PEM](https://googlecoursera.qwiklabs.com/lab_instances/download_pem/15718899" \t "_blank)

[file\_download](https://googlecoursera.qwiklabs.com/lab_instances/download_ppk/15718899" \t "_blank)

[Download PPK](https://googlecoursera.qwiklabs.com/lab_instances/download_ppk/15718899" \t "_blank)

**Introduction**

In this lab, you'll use the new commands you learned to do some process maintenance on a Linux virtual machine.

**Head's up**: You'll experience a delay as the labs initially load (particularly for Windows labs). So, please **wait a couple of minutes for the labs to load**. Please also make sure to access the labs **directly through Coursera** and not in the Qwiklabs catalog. If you access the labs through the Qwiklabs catalog, you will ***not*** receive a grade. (As you know, a passing grade is required to matriculate through the course.) The grade is calculated when the lab is complete, so be sure to hit "**End Lab**" when you're done!

You'll have 60 minutes to complete this lab.

Start the lab

You'll need to start the lab before you can access the materials in the virtual machine OS. To do this, click the green “Start Lab” button at the top of the screen.

**Note:** For this lab you are going to access the **Linux VM** through your **local SSH Client**, and not use the **Google Console** (**Open GCP Console** button is not available for this lab).

Start Lab

After you click the “Start Lab” button, you will see all the SSH connection details on the left-hand side of your screen. You should have a screen that looks like this:



**Accessing the virtual machine**

Please find one of the three relevant options below based on your device's operating system.

**Note:** Working with Qwiklabs may be similar to the work you'd perform as an **IT Support Specialist**; you'll be interfacing with a cutting-edge technology that requires multiple steps to access, and perhaps healthy doses of patience and persistence(!). You'll also be using **SSH** to enter the labs -- a critical skill in IT Support that you’ll be able to practice through the labs.

Option 1: Windows Users: Connecting to your VM

In this section, you will use the PuTTY Secure Shell (SSH) client and your VM’s External IP address to connect.

**Download your PPK key file**

You can download the VM’s private key file in the PuTTY-compatible **PPK** format from the Qwiklabs Start Lab page. Click on **Download PPK**.



**Connect to your VM using SSH and PuTTY**

1. You can download Putty from [here](https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe)
2. In the **Host Name (or IP address)** box, enter username@external\_ip\_address.

**Note:** Replace **username** and **external\_ip\_address** with values provided in the lab.



1. In the **Category** list, expand **SSH**.
2. Click **Auth** (don’t expand it).
3. In the **Private key file for authentication** box, browse to the PPK file that you downloaded and double-click it.
4. Click on the **Open** button.

**Note:** PPK file is to be imported into PuTTY tool using the Browse option available in it. It should not be opened directly but only to be used in PuTTY.



1. Click **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.

**Common issues**

If PuTTY fails to connect to your Linux VM, verify that:

* You entered **<username>**@**<external ip address>** in PuTTY.
* You downloaded the fresh new PPK file for this lab from Qwiklabs.
* You are using the downloaded PPK file in PuTTY.

Option 2: OSX and Linux users: Connecting to your VM via SSH

**Download your VM’s private key file.**

You can download the private key file in PEM format from the Qwiklabs Start Lab page. Click on **Download PEM**.



**Connect to the VM using the local Terminal application**

A **terminal** is a program which provides a **text-based interface for typing commands**. Here you will use your terminal as an SSH client to connect with lab provided Linux VM.

1. Open the Terminal application.
   * To open the terminal in Linux use the shortcut key **Ctrl+Alt+t**.
   * To open terminal in **Mac** (OSX) enter **cmd + space** and search for **terminal**.
2. Enter the following commands.

**Note:** Substitute the **path/filename for the PEM** file you downloaded, **username** and **External IP Address**.

You will most likely find the PEM file in **Downloads**. If you have not changed the download settings of your system, then the path of the PEM key will be **~/Downloads/qwikLABS-XXXXX.pem**

chmod 600 ~/Downloads/qwikLABS-XXXXX.pem

ssh -i ~/Downloads/qwikLABS-XXXXX.pem username@External Ip Address



Option 3: Chrome OS users: Connecting to your VM via SSH

**Note:** Make sure you are not in **Incognito/Private mode** while launching the application.

**Download your VM’s private key file.**

You can download the private key file in PEM format from the Qwiklabs Start Lab page. Click on **Download PEM**.



**Connect to your VM**

1. Add Secure Shell from [here](https://chrome.google.com/webstore/detail/secure-shell-app/pnhechapfaindjhompbnflcldabbghjo) to your Chrome browser.
2. Open the Secure Shell app and click on **[New Connection]**.



1. In the **username** section, enter the username given in the Connection Details Panel of the lab. And for the **hostname** section, enter the external IP of your VM instance that is mentioned in the Connection Details Panel of the lab.



1. In the **Identity** section, import the downloaded PEM key by clicking on the **Import…** button beside the field. Choose your PEM key and click on the **OPEN** button.

**Note:** If the key is still not available after importing it, refresh the application, and select it from the **Identity** drop-down menu.

1. Once your key is uploaded, click on the **[ENTER] Connect** button below.



1. For any prompts, type **yes** to continue.
2. You have now successfully connected to your Linux VM.

You're now ready to continue with the lab!

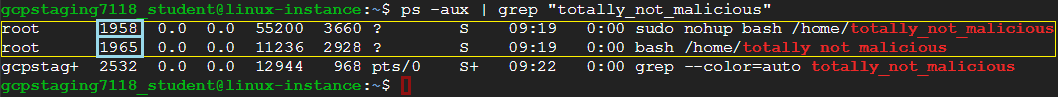
**Terminating a specific process**

The ps -aux command allows you to list all currently running processes on a Linux machine. However, the list of processes is often super long, which makes finding a specific process pretty tough. To filter the processes you're interested in, you can pipe the output of ps through grep.

There are two "malicious" processes currently running on your machine, called "totally\_not\_malicious". You can run ps and grep to find them, using this command:

ps -aux | grep "totally\_not\_malicious"

You should see output similar to this. The top two lines are the two processes, while the last line is the grep process you just used to search for them. Check out the four-digit numbers on the left of each of the rows; these are the process IDs.



To stop a process, you can use the kill command. You need to use sudo to have permission to stop them. You also need to specify the ID of the process, which will likely be different on your machine than what's shown above (the ID is outlined in light blue):

sudo kill [PROCESS ID]

After killing the processes, you can verify that they're no longer running by running the original command again:

ps -aux | grep "totally\_not\_malicious"



Click Check my progress to verify the objective.

Stop the malicious processes

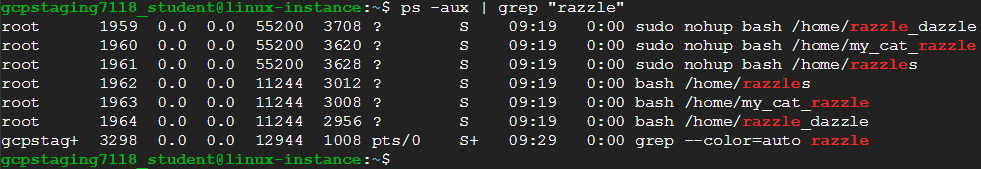
Check my progress

**Terminating multiple processes**

There are also multiple processes running on your computer containing the word "razzle". You can find them in the same way that you found the previous process using ps. Because grep doesn't look for full matches, it can be used to find any process that contains a specific substring:

ps -aux | grep "razzle"

The below shows all six processes that contain the word "razzle". (Again, you can ignore the last process because it's the process running grep.)



To kill each of the processes, you can use the same kill command as before, substituting in each process ID:

sudo kill [PROCESS ID]

To verify that the processes were successfully stopped, you can use the same command you used to find them in the first place:

ps -aux | grep "razzle"

You should only see the process for the grep command, indicating that the other processes are no longer running:



Click Check my progress to verify the objective.

Stop the razzle processes

Check my progress

**Conclusion**

Wohoo! You've successfully used ps to find processes on Linux, and used kill to end them. These are common Linux commands, so we recommend you practice until you feel comfortable using them.

**End your lab**