INTRO TO AMAZON WEB SERVICES

Yaakov Diament

Contents

OVERVIEW OF THE DOCUMENT

This document, and the videos complementing it, are intended to assist newcomers to Amazon Web Services Elastic Cloud Computing, referred to from here on as AWS EC2, in getting their feet wet running code on a server. The instructions below are not meant to encompass the full scope of AWS, which a brief document and short videos could not begin to do, but just to provide a peep through the keyhole. The guide below will describe the necessary procedures to accomplish a narrow set of tasks. The instructions and videos will explain how to create an account, as well as what is necessary to run code on a virtual Amazon Linux machine. The code will be sent to a server either directly from a local machine, or by setting up a git repository on a virtual Amazon Linux machine on an EC2 server. This document will also provide links to several videos throughout the instructions, intended to clarify the procedures discussed below. It is recommended that these videos be watched, as these pages are composed with the assumption that readers will watch the linked videos provided.

One of the most central components of Amazon's EC2 service is the concept of key pairs. As per Amazon's own description found at https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-key-pairs.html,

"Amazon EC2 uses public–key cryptography to encrypt and decrypt login information. Public–key cryptography uses a public key to encrypt a piece of data, such as a password, then the recipient uses the private key to decrypt the data. The public and private keys are known as a key pair.

To log in to your instance, you must create a key pair, specify the name of the key pair when you launch the instance, and provide the private key when you connect to the instance."

In other words, in order to connect to an EC2 server, a user must have a private key file downloaded from the AWS website saved on their machine. In addition, an EC2 instance must be created which allows those with this key pair file to connect and interact with the instance. The technicalities of this process along with further explanation can be found in the pages below.

As was mentioned, the way in which EC2 will be used is by creating a server, or "instance," which is set up with the correct key pair. Amazon sends information between a local machine and an EC2 server in an encrypted form, using SSH cryptography. A machine attempting to connect to the server uses a private key file to decrypt the information and send information to the server securely. This will be further explained in the coming pages.

The following sections will describe how to create an EC2 server with Amazon Linux and connect using a key pair. Connecting in this fashion will allow interaction with the server through a command line, much the way one would interact with a physical machine. This interface will allow a user to either integrate an EC2 instance with GIT, or copy files to the instance directly, allowing applications to be run as though one were running them on another, more powerful, physical machine. The document will conclude with a video and written description of how to configure and run the VisualVM Java Profiler from a local machine to profile code running over an EC2 server. All instructions below are collated in order of what should be understood first, starting with the process of creating and setting up a server.

All videos and instructions are cumulative and assume an understanding of all previously provided videos and information. The video links referenced throughout are provided below for convenience and will be mentioned in the written instructions as they become relevant.

VIDEO CONTENTS

- ☐ Initial key - pair setup: https://www.youtube.com/watch?v=XR12ppYQWBw&feature=youtu.be
- ☐ Creating and launching a server: https://www.youtube.com/watch?v=bPFXqcPxi_c
- ☐ Copying files and directories to a server, directly from a local machine: https://www.youtube.com/watch?v=n5DAU_wuSHY&t=235s
- ☐ Cloning a GitHub repository to a server, to allow all code in the repository to be run: https://www.youtube.com/watch?v=EBxdLjaeHZI&t=5s
- ☐ Connecting the VisualVM profiler to an EC2 instance: https://www.youtube.com/watch?v=e8gMt0wixls&t=7s

1. INITIAL SETUP

To begin, you must sign up for an AWS account. To sign up, go to https://aws.amazon.com/ and choose **Create an AWS Account**.

To access a virtual server, a key pair must be created. This pair consists of a public key which is created by AWS and associated with the virtual server, and a private key, which is like a password, or "key," used to access the server from a local machine. MAKE SURE NOT TO LOSE IT AS IT CANNOT BE RETRIEVED.

A video about key pair setup can be found at https://www.youtube.com/watch?v=XR12ppYQWBw&feature=youtu.be

Use the following steps to access the dashboard of Amazon EC2 and create a key pair:

- ☐ Open the AWS Console at https://console.aws.amazon.com . In the top right menu, next to the user name, is a dropdown menu of regions. This specifies which physical location your server will be in. Make sure the region is set to N. Virginia. These instructions will assume the region is set to North Virginia, and AWS will not necessarily behave the same in a different region.
- ☐ Select "Services" from the navigation bar at the top of the page, find "EC2," and select it. You will now be directed to the EC2 Management Console.
- ☐ In the far-left column, under Network & Security, select Key Pairs. Then select the Create Key Pair button.
- ☐ Name the key pair. We will assume the keypair is named mykey.

- ☐ During creation, a file name mykey.pem will be downloaded. You will now need to set the key up for future use.

- ☐ Download the PuTTY installer. (For Windows it can be found at http://mng.bz/A1bY) . Run the application PuTTYGen. Under Type Of Key To Generate, select the SSH RSA option. Select Load. Because PuTTYgen will, by default, only display .ppk filed, switch the file extension of File Name to All Files. Select the mykey.pem file from where it is saved, and select Open. Select OK in the dialog box. Select Save Private Key, and ignore the warning about saving without a passphrase. The .pem file has now been converted to .ppk, which will be needed later.

2. LAUNCHING A VIRTUAL SERVER

- ☐ For a video about launching a server with Amazon Linux, as well as how to connect to a server using PuTTY, see https://www.youtube.com/watch?v=bPFXqcPxi_c .

- ☐ Go to console.aws.amazon.com, and select Services -> EC2 -> Launch Instance -> Free Tier Eligible Amazon Linux AMI -> Review and Launch -> Launch.

- ☐ Select a key pair which you have set up, or create one, and launch the instance, then select View Instances.

- ☐ Open PuTTY. On the left, select SSH -> Auth, and then go to “Private key file for authentication, select Browse, and select the correct ppk file.

- ☐ Go back to Session. Make sure the Port is set to 22, with connection type SSH. Using Amazon Linux, the Hostname should be

  ec2-user@PublicDNSOfYourInstance. This is visible from the console, or by selecting Connect and looking in the information provided.

- ☐ Select Open. If there is a warning dialog, select Yes. Update the instance if prompted.

- ☐ To terminate a server, go to “actions” and select “terminate.” To stop the server, so that hours will not be billed but saved material will not be lost, go to “actions” and select “stop.”

3. COPYING CODE TO AN EC2 INSTANCE FROM A LOCAL MACHINE

A video guide is available at
https://www.youtube.com/watch?v=n5DAU_wuSHY&t=235s

- ☐ Right click on the desktop of your computer. Select 'Git Bash Here" to open a Git Bash.

To Copy A Single File:

- ☐ Use the cd command in the bash to navigate to the directory the file is in. You may need quotes around the path in the cd command.

- ☐ Type the following command:

  scp -i "path\to\your\privateKeyFile.pem" nameOfFile.theFileExtension ec2-user@**********.amazonaws.com:path/copying/into/on/instance

  The format of this command is scp -i , then the path to the correct pem file, the file to be copied, the user and DNS, (In this case using Amazon Linux, it is ec2-user@TheDNS), colon (no space), and finally the path to copy to in the remote machine.

- ☐ Finally, answer yes when prompted.

To Copy A Directory Along With All Files And Directories Below It:

- ☐ To copy a directory, a slightly different set of command arguments is used.

- ☐ scp -i "path\to\your\privateKeyFile.pem" nameOfFile.theFileExtension ec2-user@**********.amazonaws.com:path/copying/into/on/instance

  For a better understanding, especially at first, see the video linked above.

4. CLONING A GITHUB REPOSITORY TO AN EC2 INSTANCE

For a brief video demonstration, see
https://www.youtube.com/watch?v=EBxdLjaeHZI&t=5s

- ☐ On the remote machine, type the following command: sudo yum install git

- ☐ On the GitHub website open the repository and select "Clone Or Download", select "Clone With HTTPS," and copy the link.

- ☐ On the remote instance, type the command: git clone https://github.com//restOfURLYouCopied

☐ Enter your username and password when prompted, and the repository will be cloned.

5. INSTALLING JAVA 8 ON AMAZON LINUX EC2

By default, an Amazon Linux EC2 server will have Java 7, and will not have any JDK set up. Installing Java 8 consists of two commands, one to install java 8 and its JDK, and one to remove Java 7 from the virtual machine.

Enter the following two commands, one after the other, on the instance:

sudo yum install java-1.8.0-openjdk-devel -y

sudo yum remove java-1.7.0-openjdk

6. CONNECTING VISUALVM PROFILER TO A REMOTE EC2 INSTANCE

For a visual demonstration, see
https://www.youtube.com/watch?v=e8gMt0wixls&t=7s

1) Set up and SSH Tunnel with SOCKS proxy on Port 10,000

☐Open Putty. In **Session,** set **Host Name** as ec2-user@(public DNS).

Example: ec2-user@ec2-54-221-138-0.compute-1.amazonaws.com

☐Go to **Connection -> SSH -> Auth** and under "Private key file for authentication" enter the path to the correct ppk file for the instance.

☐Under **Connection -> SSH -> Tunnels**, type 10000 into **Source Port**, select **Dynamic,** then **Add**. "D10000" should appear in **Forwarded Ports**. Select **Open** to open the tunnel.

☐To run code with JMX monitoring, Start the application with the following arguments added:

```
java -Dcom.sun.management.jmxremote.port=3333 \
    -Dcom.sun.management.jmxremote.ssl=false \
    -Dcom.sun.management.jmxremote.authenticate=false \
    YourJavaApp
```

(see https://docs.oracle.com/javase/7/docs/technotes/guides/visualvm/jmx_connections.html)

☐ If necessary, include classpath and package, or any other necessary arguments, as well. Example:

```
java -cp IntroToCompSci/homework/interfaces/QuizGenerator/src/main/java \
-Dcom.sun.management.jmxremote.port=3333 \
-Dcom.sun.management.jmxremote.ssl=false \
```

-Dcom.sun.management.jmxremote.authenticate=false \

edu.yu.intro.interfaces.quizgenerator.QuizGenerator

(Slashes at the end of each line are added to allow a single command to be typed in the Bash across multiple lines, using the Enter key.)

2) Set Up VisualVM To Use SOCKS Proxy

☐ In the VisualVM menu, select **Tools -> Options -> Network**.
Select **Manual Proxy Settings** and set
SOCKS PROXY as *localhost*, and the port as 10,000.

☐ Make sure to uncheck **Use the same proxy settings for all protocols.**

3) Add Remote Host To VisualVM

☐ Get the **Private IP Address** of the EC2 instance, viewable in the console.

☐ In VisualVM, select **File -> Add Remote Host,** enter the Private IP Address as the **Host Name,** and select OK.

☐ Right click on the host and select **Add JMX Connection.**

☐ Enter **[Private IP Address]:[JMX Port].** In our case the port is 3333. Example: 172.31.16.181:3333

☐ Select okay and wait for VisualVM to connect and begin profiling the currently running application