UNIVERSITY OF NOTRE DAME | MENDOZA COLLEGE OF BUSINESS   GROW THE GOOD IN BUSINESS ™

MSBA Cloud Computing
Server Lab

In this lab, you will work with the AWS Elastic Compute Cloud (EC2). This service provides you with the ability to create and manage virtualized server instances in the cloud. In this lab, you will create an EC2 instance running Microsoft Windows Server, access your server over the Internet, configure it to operate as a web server, and host files on the server. You will also create a web page that incorporates static resources stored in the S3 bucket that you created in the storage lab.

I. Create an EC2 Server Instance

The EC2 instance is the core unit of server-based computing in AWS. An EC2 instance is a virtualized server that runs on shared hardware within an AWS datacenter and runs the operating system of your choice. We will begin by creating an EC2 instance.

1. Navigate to the AWS console and choose EC2 from the Services menu. It is in the Compute section.

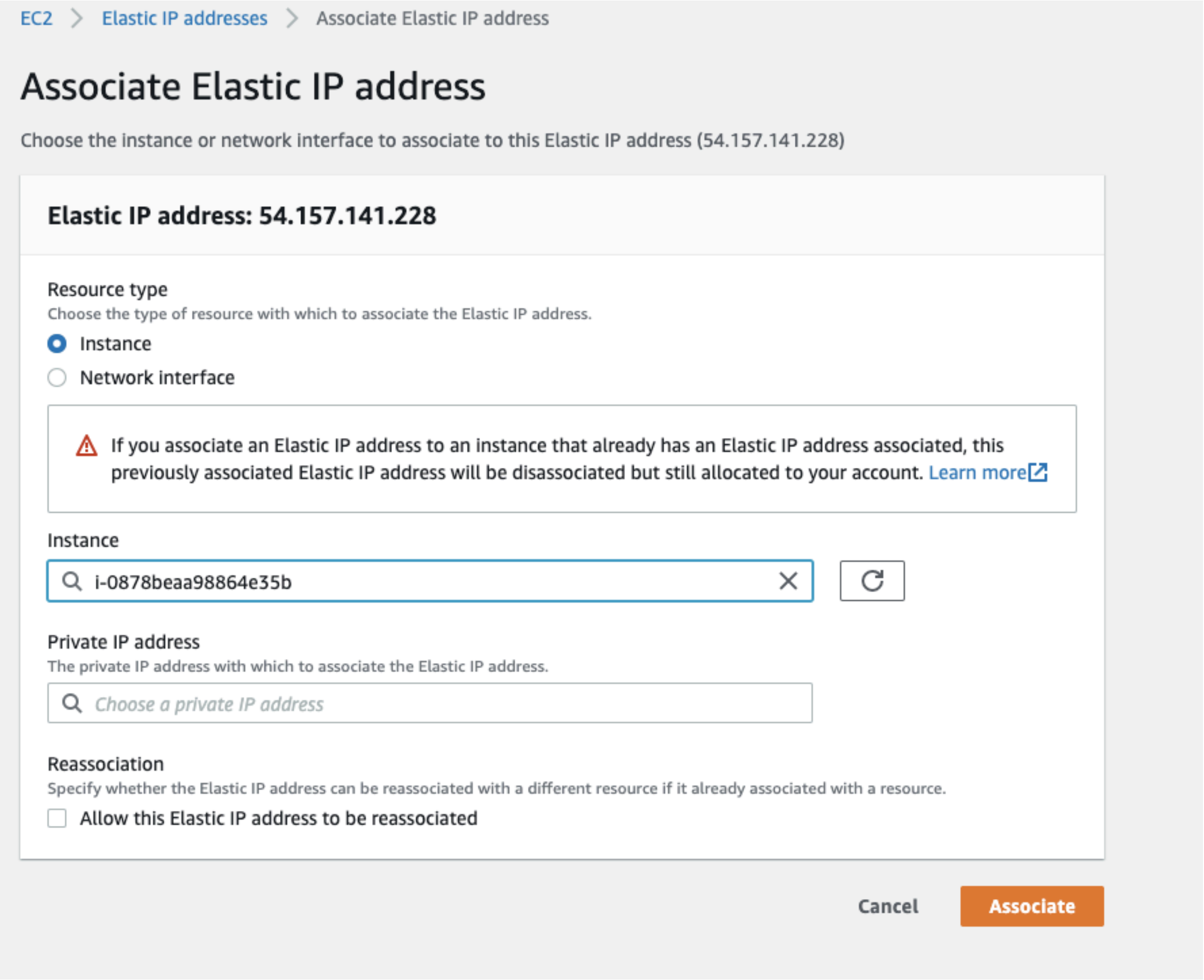
2. Launch an EC2 instance. You will have to make several choices during the instance creation process:
    a. Use the Microsoft Windows Server 2022 Base Amazon Machine Image (AMI). An AMI provides a snapshot of a server that will be the starting point for your work. In this case, you will be creating a standard Windows server.
    b. Choose the t2.micro instance type. **This is extremely important.** EC2 instances are charged against your credit balance at an hourly rate. If you choose a larger instance type, it may deplete your credit more quickly, preventing you from completing other labs in this course.
    c. **Enable** Detailed CloudWatch Monitoring (in the Advanced Details section)
    d. Name your server *netid*-web (e.g. mchapple-web)
    e. Change the EBS storage volume that you will be provisioning to a 30 GiB volume of gp3 storage. (The default is to use gp2.)
    f. In the Key Pair section, choose to create a new key pair using the .pem file format. Make sure to create a new key, rather than using any existing keys that might exist in your account. You can name the key with your NetID or anything else you'd like. **This key file is required to retrieve the password for your instance, so do not lose it! Save it to your Google Drive or some other safe location because you will need it to complete future labs. You will not have another opportunity to download it after completing this step.**
    g. Accept all of the other default options presented to you.

3. Click the View all instances button and then wait until your new instance reaches the “Running” state. You may need to refresh the screen. This will take a few minutes.

II. Assign an Elastic IP Address

In AWS, when you start and stop a server it gets a new IP address each time you restart it. This will be problematic for our lab, so we need to reserve an IP address for the server using the AWS Elastic IP feature.

1. On the AWS console’s EC2 service screen, choose Elastic IPs from the left sidebar.
2. Click the Allocate Elastic IP Address button.
3. Allocate the IP address from the Amazon pool and make note of the address.
4. Select the address that you just created and use the Actions menu to associate it with the EC2 instance that you created in Part I. You can do this using just the instance ID. You do not need to fill in the Private IP address field. Your association screen should look similar to this:



EC2 > Elastic IP addresses > Associate Elastic IP address

Associate Elastic IP address

Choose the instance or network interface to associate to this Elastic IP address (54.157.141.228)

Elastic IP address: 54.157.141.228

Resource type
Choose the type of resource with which to associate the Elastic IP address.

● Instance
○ Network interface

⚠ If you associate an Elastic IP address to an instance that already has an Elastic IP address associated, this previously associated Elastic IP address will be disassociated but still allocated to your account. Learn more

Instance
i-0878beaa98864e35b

Private IP address
The private IP address with which to associate the Elastic IP address.
Choose a private IP address

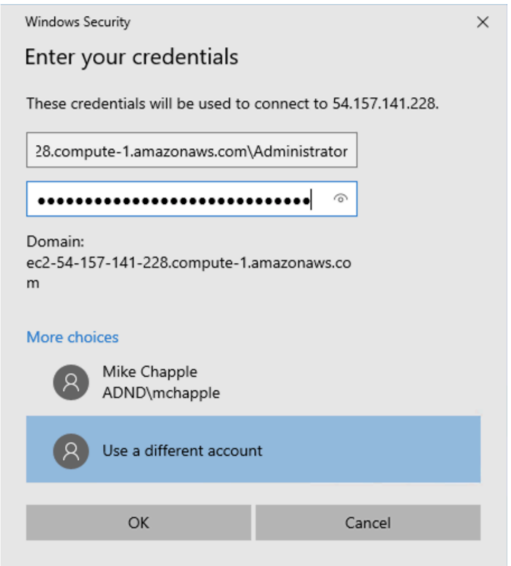
Reassociation
Specify whether the Elastic IP address can be reassociated with a different resource if it already associated with a resource.
□ Allow this Elastic IP address to be reassociated

III. Access Your Server

1. Before you can access your server, you will need to gather two pieces of information from the AWS console:

    a. The DNS name of your server. This will be listed as “Public DNS” or “Public DNS (IPv4)” in the console.
    b. The Administrator password for your server. You can obtain this by right-clicking on the server instance and choosing “Security” and then “Get Windows Password” from the pop-up menu. You will need to provide the private key file to obtain the password. Note that it takes **up to four minutes** from the time you create your server until the Windows password is available.
    c. **Be sure to save this password someplace safe. You will need it for future labs.**

2. Connect to your server. If you are connecting from a Windows system, use the Remote Desktop application. On a Mac, you can download Microsoft Remote Desktop from the App Store. You will need to provide the connection details that you gathered in step 1 above. Use your Public IPv4 address
    a. If you are logging in from your own computer, you should be able to just set the username to Administrator and log in with the password you retrieved above.
    b. If you are logging in from a ND classroom or lab computer, you need to use a more complex username consisting of the public DNS name of your server, followed by a backslash, followed by the word Administrator. For example:
        - `ec2-54-157-141-228.compute-1.amazonaws.com\Administrator`



Windows Security

Enter your credentials

These credentials will be used to connect to 54.157.141.228.

?8.compute-1.amazonaws.com\Administrator

••••••••••••••••••••••••••••••

Domain:
ec2-54-157-141-228.compute-1.amazonaws.co
m

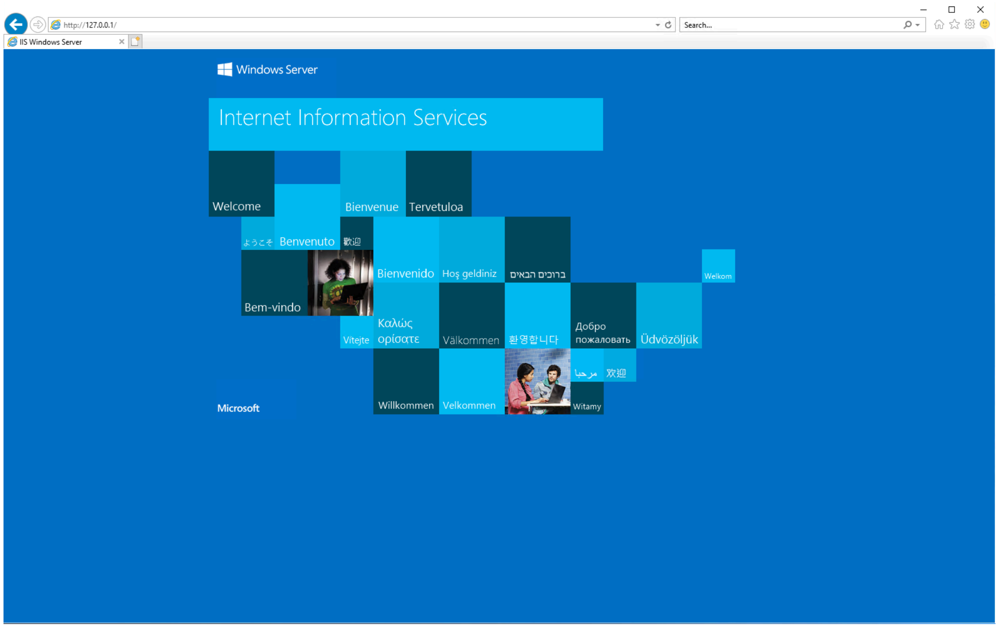
   c. You may see a warning that the connection you are making to the server may not be secure. This is normal and you may safely ignore the warning.

3. The initial connection may take a couple of minutes while the server configures itself for initial use. Eventually, you will see a Windows desktop that you can interact with normally.

IV. Configure Your Server as a Web Server

1. After connecting to your server, open the Server Manager application. Use that application to add the Web Server (IIS) role to your server. You should accept all of the default options for this role.

2. Open Microsoft Edge and navigate to http://127.0.0.1 This is requesting the local web server installed on your server. You should see this page if you successfully completed the steps above:



This is the default web page for the Windows web server. In the next section of the lab, you will replace this default page with one of your own design.

V. Create a Web Page

1. Create a simple web page for use on your server. If you are familiar with HTML, you may create your own page from scratch. If you are not familiar with HTML, you may wish to start from this basic template:
   https://mendozacloud.s3.amazonaws.com/index.htm

You may use any tool you'd like to create the file. Notepad is a reasonable option that is already installed on your server.

For your convenience, here is a copy of the HTML used to create this sample page. You can cut-and-paste it into Notepad on the server and then save it following the instructions in Step 2:

```
<html>
      <head>
            <title>My Web Page</title>
            <style></style>
      </head>
      <body>
            <H2>My Web Page</H2>

            <P>Welcome to my web page.</P>

            <P>This is a picture of the Dome:</P>

            <A HREF='http://www.nd.edu'>
            <IMG
SRC='https://mendoza.nd.edu/wp-content/uploads/2018/10/5.7.09-
Main-Building-Spring-Scenic-min-600x400.jpg'>
            </A>

            <P>If you click it, you will go to the <A
HREF='http://www.nd.edu'>Notre Dame homepage</A>.</P>
      </body>
</html>
```

2. When you are finished, save the file using the name index.html (not as a text file) and save it to the folder C:\inetpub\wwwroot\

3. Reload the http://127.0.0.1 webpage in Edge. You should now see the webpage that you created. You must access this page from Edge on your server. It is not yet accessible over the Internet.

4. Modify the webpage to include an image that you uploaded to your S3 bucket during the storage lab. Directly include the file in the bucket rather than uploading it to your server.

5. Take a screenshot of the entire web browser window. You will need this for your lab report.

VI. Configure Public Access to Your Website

In the previous section, you accessed the website that you created on your server by using a web browser located on the server itself. Return to the EC2 console and check the public IP address of your server. Attempt to access the website using this public IP address from a web browser that is running on your own device (classroom computer, smartphone, etc.). Use a URL beginning with http:// (not https://). For example, if your IP address is 54.81.91.134, you would visit http://54.81.91.134/.

You will find that you are unable to access the site. This is because the security groups that you have already applied do not allow this access. In order to allow web traffic from the Internet to reach your web server, you will need to ensure that both HTTP and HTTPS traffic is allowed to reach your server.

1. Return to the EC2 screen. In the left-hand navigation pane, scroll down to the “Network and Security” section and click on the Security Groups link.

2. Create a new security group by clicking the Create Security Group button. For the name, specify Public Web Access. You can put anything you’d like for the description of the group.

3. Add both HTTP and HTTPS inbound rules to the security group. You should configure your rules to allow access from any IPv4 address.

4. In the left-hand navigation, click on the Instances link and select your instance. Under actions, select Security → Change Security Groups. Add your newly-created Public Access Security Group. (Note that there is already a group added to your instance called launch-wizard-1 or something similar. Leave that group there and add the Public Web Access group as an additional group.)

5. Now that you have allowed HTTP and HTTPS traffic from the Internet to reach your web server, access the public IP address from the web browser on your computer (not the web browser on the EC2 instance). (Note: you should try this using the HTTP address, as your web server is currently configured to support only HTTP traffic).

Take a screenshot showing your web page and the public IP address in the address bar.

VII. Configure Alerting

The purpose of this section is to create an email alert which sends you an email when the average CPU utilization on your server exceeds 75% on a sustained basis.

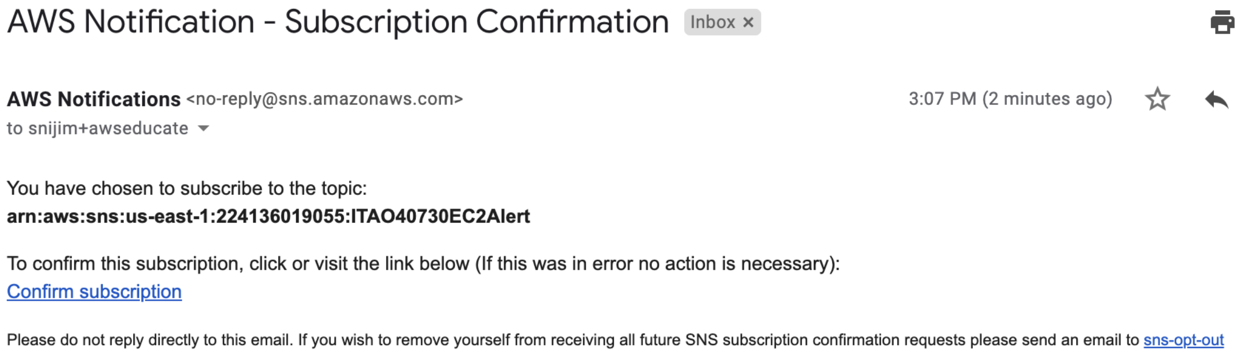
1. Return to the AWS console. In the Search for services bar at the top of the screen, type SNS and navigate to the Simple Notification Service screen.

2. Create a topic called MyEC2Alert specifying standard delivery. Use MyEC2Alert as the display name, and accept all other default options as presented to you.

3. After creating the topic, proceed to create a subscription. You can find this on the lower half of the screen under the Subscriptions tab.

4. Specify MyEC2Alert as the Topic ARN, Email as the Protocol, and your Notre Dame email address as the endpoint.

5. Make sure to confirm your email subscription in order to receive email alerts. Check your nd.edu email for a message similar to the one below, and click the link to confirm the subscription. If you do not see the message, be sure to check your spam folder.



AWS Notification - Subscription Confirmation

AWS Notifications <no-reply@sns.amazonaws.com>
to snijim+awseducate

3:07 PM (2 minutes ago)

You have chosen to subscribe to the topic:
arn:aws:sns:us-east-1:224136019055:ITAO40730EC2Alert

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
Confirm subscription

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to sns-opt-out

6. Navigate back to the EC2 dashboard, and select your running instance.

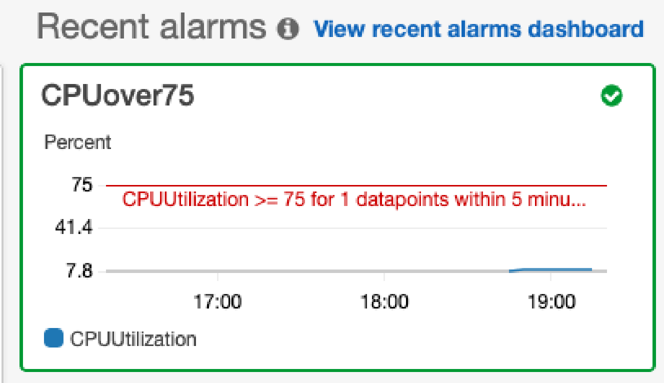
7. From the Actions menu, choose Monitor and troubleshoot → Manage CloudWatch alarms. This will bring you to a screen where you will specify the details of this new alarm.

8. In the Alarm notification section, please select the MyEC2Alert topic you created in step 2 of this section.

9. Do not configure an alarm action. However, please explore the types of options that can be taken when the alarm is triggered and make note of them.

10. In the Alarm thresholds section, choose average CPU Utilization as the metric for your alarm. Configure your alert to occur whenever the average CPU utilization is more than 75 percent for a single five-minute period. (Be careful to set this to 75%, not 0.75%!)

11. To verify the existence of the alarm you just created, type CloudWatch into the Search for services bar at the top of the screen. This will take you to the CloudWatch landing page. You should see your alarm towards the right side of the screen, similar to this:



CPUUtilization >= 75 for 1 datapoints within 5 minu...

VIII. Trigger Alerting

Normally, the alert that you just created would send an alarm when user activity is very high on your web server, causing it to reach 75% of CPU utilization. The server that you just built does not have any real users, so we need to simulate real activity. To do this, we will create and run a batch script that consumes a lot of CPU time. This will have the same effect as if many web users were impacting the server.

1. On your EC2 instance, open notepad and paste in the following code:

```
@echo off
:loop
goto loop
```

2. Choose Save As. In the dialog box that pops up, specify the desktop as the location, **CPUeater.bat** as the filename, and make sure you select “all file types” in the space below the filename.

3. Close out of notepad and get to the desktop of the EC2 instance. Right-click on CPUeater.bat and choose “Run as administrator.” A black box will appear with no words in it - this means the script is running and consuming CPU. If you want to verify that is the case on the server, you can do so with the Task Manager application.

4. It will take at least **5 minutes** for the alert you created to trigger. Wait at least 5 minutes before moving on to the next section of the lab.

IX. Performance Monitoring

In addition to the built-in performance monitoring features of Microsoft Windows, Amazon provides you with a set of monitoring tools called CloudWatch. CloudWatch is quite useful for getting a quick look at your server's performance from a CPU, disk, and networking perspective.

1. Back in the AWS EC2 console, locate the Monitoring tab for your EC2 instance. Review the metrics available to you.

2. Click on the three dots next to the CPU Utilization graph and choose "View in Metrics" to see a larger view. The times that you see will be in Greenwich Mean Time.
   a. If the graph shows no CPU utilization, hit the refresh button near the top-right corner of the screen.
   b. Change the period drop-down to "1 minute" to view more granular data.
   c. Take a screenshot of this graph.

3. Annotate your graph to show the time period where you were running the CPU Eater scripts. You will include this annotated graph in your lab report.

X. Confirm Alert Configuration

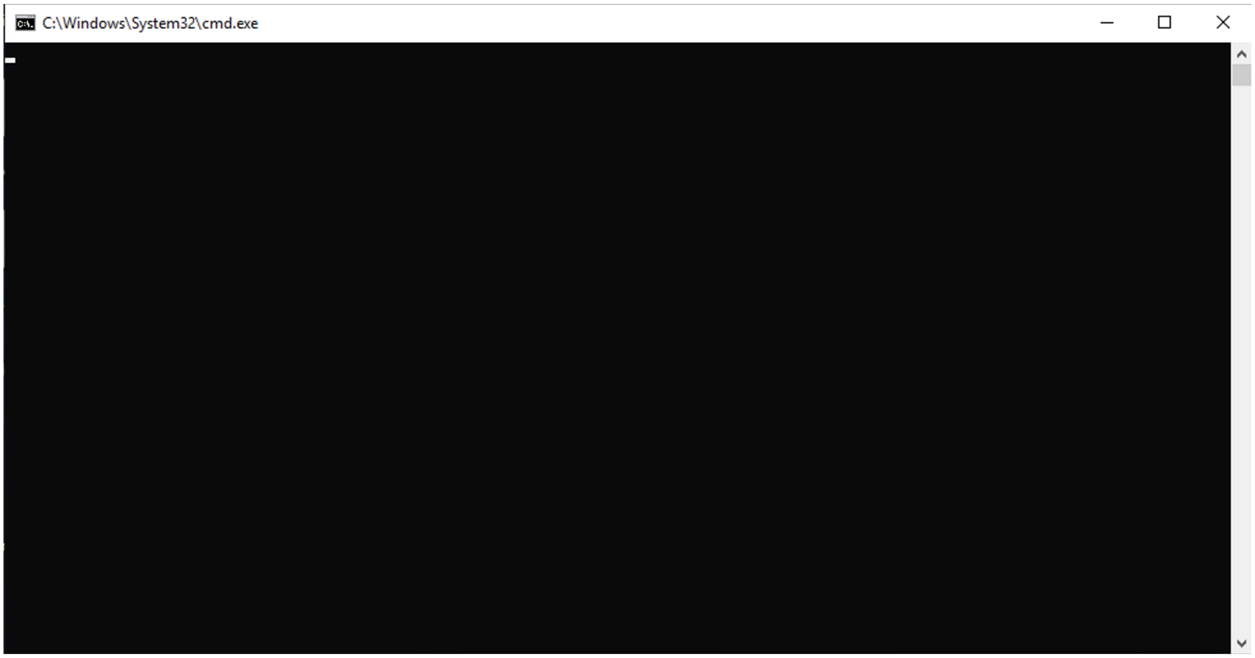
In this final section, you will verify that the MyEC2Alert has triggered successfully.

1. In your nd.edu email account, look for an email with a subject similar to "ALARM: "awsec2-i-0917bc4021103e51e-GreaterThanThreshold-CPU-Utilization" in US East (N. Virginia)." This email serves as confirmation that the alarm you configured earlier in the lab is working, and that the email notification mechanism is in place.

XI. Lab Cleanup

Important: Make sure that you have all of the information you need to complete your lab report before working through this section. It is important that you complete this section to stop your account from continuing to accrue charges. If you leave your server running, you may deplete your AWS credits.

1. Kill the CPUeater.bat file by clicking the X in the top right corner of the black window:



C:\Windows\System32\cmd.exe

2. When you are ready, return to the AWS console, select the EC2 instance that you created and choose Instance State → Stop from the pop-up menu. This is the equivalent of turning your server off and stops charges from accruing. We will be turning it back on in the next lab. **Do not choose Terminate, as that has the effect of irrevocably deleting your instance.**

X. Deliverable

Your deliverable from this in-class lab is a brief lab report describing your results, submitted to Canvas as a PDF document. Your report should contain two sections:

Methodology

Provide a concise narrative explaining what you did in the lab. You should include the following elements from your work:

1. Your name
2. The screenshot that you took in Section V
3. The screenshot that you took in Section VI
4. The possible alarm actions you noted in Section VII
5. The annotated graph that you created in Section IX
6. A copy of the High CPU Alert email that you received in Section X

Please note that this section should not simply be the items above pasted into a report. You should provide a concise narrative explaining what you did and demonstrating that you understood the steps you were following, rather than simply clicking through the lab report.

Analysis

Answer each of the following questions. This section of the lab report should be divided into sections with a separate section answering each question. For each question, assume that the account is not eligible for the AWS Free Tier.

a. Determine the monthly cost for the EC2 instance that you created in this lab by using the information provided on the EC2 Pricing page. Assume that the month has 30 days and the server will run continuously during that month as an on-demand instance in the Northern Virginia region. Include the following costs in your calculation:

    a. Server instance costs
    b. EBS volume cost
    c. Elastic IP address cost

    N.B. that you do not need to include any data transfer costs for any of the scenarios in this lab.

b. You are building a Windows application server for your organization. After conferring with the application administrators, you learn that the server will need at least 50GB of memory to operate effectively. It also needs at least 8 virtual CPU cores, 2 TB of disk space, and an elastic IP address. It will run continuously during the 30-day month in the Northern Virginia region.

    Select an appropriate instance type for this server that meets the requirements specified above in the most cost-effective manner. Explain your choice and determine the monthly cost of operating this server as an on-demand instance.

c. You are building a Linux web server for your organization. After conferring with the appropriate technical staff, you learn that the server's use varies significantly. Most of the time, the server is used only sporadically and requires just 1GB of memory and a single virtual CPU, which it does not use fully. However, the company occasionally runs live promotions that generate a short burst of activity requiring two CPUs and 4GB of memory for 15-30 minutes. The server requires a 20GB disk and an elastic IP address. It will run continuously during the 30-day month in the Northern Virginia region.

    Select an appropriate instance type for this server that meets the requirements specified above in the most cost-effective manner. Use a single instance to solve this problem. You do not need to consider solutions that change instance types during server operation. Explain your choice and determine the monthly cost of operating this server as an on-demand instance.