|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | |
|  | | Lab 1 - Getting started with AWS | | | | |  | |
|  |  | | | | | | |  |
|  | | | |  | |  | | |
|  | | | | Md Abu Ahammed Babu GENIAL Cohort 1 | |  | | |
|  | | | | 10/09/2021—Network Programming & Distributed Applications—Teacher: Evgeny Asipov | |  | | |
|  | | | **Part I** | |  | | | |

**Part II**

|  |
| --- |
| *Task 1*Q1. What is the purpose of security groups in AWS cloud? **Ans:** Security groups in AWS works like a virtual firewall for the EC2 instances. It controls incoming and outgoing traffics using the inbound and outbound rules. |
| ***Task 2*** |
| ***Task 3***  **Q2.** What type of instance you created?  **Ans:** I created t2.micro type instance.  **Q3.** Which AMI you selected, motivate your choice.  **Ans:** I have selected Ubuntu 20.04 type AMI. There are several reasons for choosing this:   * I have worked before with Ubuntu and have familiarity with this OS * It is a lightweight OS and best monolithic kernel for programming * Ubuntu is a Linux based OS which is open source and always keep updating. |
| ***Task 4***  **Q4.** Which file system is configured on your volume?  **Ans:** gp2 file system  **Q5.** Can you change it?  **Ans:** Yes |
| ***Task 5***  **Q6.** What is ip address of your instance?  **Ans:** The IP address of my instance is - 172.31.95.50  **Q7.** What is its public and its private dns name?  **Ans:** Public IPv4 DNS - ec2-54-210-230-205.compute-1.amazonaws.com  Private IPv4 DNS - ip-172-31-95-50.ec2.internal |
| ***Task 6***  **Q8.** What is the public address on your server?  **Ans:** 100.26.191.71  **Q9.** What text have been shown when you open public dns name in web browser?  **Ans:** The Apache2 Ubuntu Default page. |
| ***Task 7*** |
| ***Task 8***  **Q10.** What was server response?  **Ans:** It shows that this site can’t be reached.C:\Users\Rasel\Desktop\q10.PNG  **Q11.** Explain why.  **Ans:** The Apache2 server isn’t installed by default to this instance which is why it is not reachable. |
| ***Task 9***    New AMI |
| ***Task 10***  **Q12.** What was server response?  **Ans:** The server shows the edited page of the instance from which the AMI image was created.C:\Users\Rasel\Desktop\q12.PNG  **Q13.** Explain why.  **Ans:** Creating an AMI is like creating a snaphot of an instance. As the AMI was created from the instance with edited index.html page, every configuration came with the AMI and will be passed to all instances created from this AMI. |
| ***Task 11***  A piece of text “AMI” was added to distinguish it. |
| ***Task 12***  The new instance server page shows like this: |
| From the snap, we can see that the default server page got updated with the addition of “AMI” text. It’s because we updated the index.html inside the new instance. |
| ***Task 13***  If I launch a new instance from the AMI, the default browser page will like as it was in the instance from where we created the AMI as it is like a snapshot of that state of the instance. But if we update the index.html after creating the instance, it won’t change or affect the default page is in AMI. |
| ***Task 14***  **Q14.** Describe your step-by-step problem searching and troubleshooting approach.  **Ans:** The steps of my troubleshooting approach could be like:   * Checking the webserver is properly installed or not. * Making sure that webserver is running. * Checking the webserver configuration file syntax. * Configured ports are open and not blocked by firewall. * DNS setting is correct. * Index file served by the webserver is the right one or not. etc   **Q15.** Demonstrate your approach using appropriate operating system’s commands and tools when troubleshooting communications between your local computer and running AWS instance configured with the web server.  **Ans:** **Step 1** – Make sure the instance is running from the aws EC2 console.  **Step 2** – Access the instance using SSH and install apache2 in Ubuntu or debian AMIs using the commands:  *sudo apt-get update*  *sudo apt-get install apache2*  **Step 3** – Running the systemctl status check using the command:  *sudo systemctl status*  If the status shows inactive then restart the server by:  *sudo systemctl restart*  and then check the status again.  **Step 4** - Run the following command to confirm that the apache2 web server is listening on port 80 (for http) or 443 (for https) for incoming connection requests from user.  *sudo iptables –nvL*  **Step 5** - Run the following commands to allow port 80 and 443 to accept incoming HTTP and HTTPS connection requests:  *sudo iptables -A INPUT -p tcp --dport 80 --syn -m conntrack --ctstate NEW -j ACCEPT*  *sudo iptables -A INPUT -p tcp --dport 443 --syn -m conntrack --ctstate NEW -j ACCEPT*  Step 6 - Run the following command to check for a UFW firewall:  *sudo ufw status verbose*  If UFW is running, use the following command to allowing incoming connection requests on ports 80 and port 443:  *sudo ufw allow in 80/tcp*  *sudo ufw allow 443/tcp* |
| ***Task 15***  **Q16.** Be able to interpret and explain the information about different protocols, their fields etc.  **Ans:** I found information on TCP, HTTP and SSH protocol. The TCP packets come with the destination and source mac addresses at the beginning and followed by the protocol information, version and source IP, destination IP and some random default texts as message body.    Also the HTTP packets contain pretty much same information as the others do but in a form of HTTP format. It includes key information like source and destination mac addresses, IP version, Time to live, source and destination IP addresses and also some dummy message to fill the packet size.    And the SSH packets also contains the mac address of destination and source along with other information about the protocol version, source and destination IP. The source of the SSH packets is our instance. |