Cloud Computing Exercise #15

Creating a VPC with a Public and a Private Subnet

***In the modules section, There is a reference document with console screenshots illustrating how each step should appear. Please refer to it to help avoid any mistakes.***

A. Preparation

1. Sign in to your AWS account as the non-root admin user.
2. Here is what you will accomplish with this lab: You will create a VPC with a public and a private subnet, and launch one EC2 instance in each subnet. You will configure the security groups in such a way that the private EC2 instance will be reachable via the ping command and the SSH protocol from the public subnet, but not from the Intenet. You will need to upload your OpenSSH key file to the public EC2 instance so that you would be able to authenticate yourself with the private EC2 instance. Thus, you will be able to log into the private EC2 instance by first logging into the public EC2 instance from your local machine, and then “hopping over” from the public EC2 instance to the private EC2 instance.

B. Create a VPC and its subnets

1. Go to the list of VPC and start creating a new VPC named “MyVPC”. Use the 172.16.0.0/16 IP address block for the VPC, and select 0 public and private subnets for now. You should see your VPC appearing on the list of VPCs together with the default (unnamed) VPC. Note that a default route table and a default ACL will also be created automatically for the new VPC.
2. Create a public subnet in your VPC. Select “Subnets” on the left-hand side menu, and click on “Create subnet”. Associate the new subnet with the VPC “MyVPC”, name the subnet as “Public Subnet” and place it in the AZ “us-east-1a”/”us-east-2a”. Use the 172.16.100.0/24 IP address subrange for the subnet. Similarly, create a private subnet named “Private Subnet”, associate it with your VPC, place it in the AZ “us-east-1b”/”us-east-2b” and specify the CIDR IP address block 172.16.200.0/24 for the subnet. You should see the two subnets you just created appearing on the list of subnets.

C. Create the Internet Gateway and the Route Tables

1. Create an Internet Gateway for the public subnet. Select “Internet Gateways” on the left-hand side menu and click on “Create Internet Gateway”, and name it “MyGateway”. After it has been created, attach it to you VPC: click on “Actions”, select “Attach to VPC” and select your VPC (not the default one, but the one you just created, MyVPC). The state of the Internet Gateway should change to “Attached”. Note the resource ID (Internet Gateway ID) of the Internet Gateway as you will need it as a target in the public subnet’s route table.
2. Create a route table for the public subnet. Select “Route Tables” on the left-hand side menu and click on “Create route table”. Name it “Public route table” and associate it with your VPC (not the default one, but the one you just created, MyVPC). After the route table has been created, you will see its detail page. Scroll down to see the content of the “Routes” tab, and you should see the VPC-local route entry already added to the route table. Add a default route (destination 0.0.0.0/0) with the Internet Gateway’s resource ID (“MyGateway”) as target. This will make the Internet accessible from the public subnet. Click on “Edit routes” and add the default route to the route table. You should see the default route added to the route table.
3. Associate the public subnet’s route table (“Public route table”) with the public subnet. Click on the “Subnet associations” tab, select “Edit subnet associations”, check the check box of the subnet “Public subnet” in the section of “Subnets without explicit associations”and save the subnet association. Your subnet details page should reflect the above change.
4. Similarly, create a route table for the private subnet named “Private routing table” and associate it with your VPC. The VPC-local route will be installed automatically, and you do not need to add any custom entries here: the subnet is private because there is no route from it to the Internet Gateway. Associate the private route table with the private subnet .

D. Create the firewall rules

1. Create a security group for the public subnet. Select “SECURITY/Security Groups” on the left-hand side menu and click on “Create security group”. Name the security group as “Public SG” and add the description “Public security group”, and associate the security group with your VPC (“MyVPC”). Then, add the following **inbound** rules:
   * Allow SSH from anywhere: Type: Custom TCP, Protocol: TCP, Source: 0.0.0.0/0, Port range: 22
   * Allow ICMP pings from only from the private subnet inside the VPC: Type: Custom ICMP – IPv4, Protocol: Echo request, Source: 172.16.200.0/16

After adding these rules, create the security group.

1. Similarly, create another security group for the private subnet and name it “Private SG” and associate it with your VPC (“MyVPC”). Add the following **inbound rules**:
   * Allow SSH from only the “Public SG” security group: Type: Custom TCP, Protocol: TCP, Source: the “Public SG” security group, Port range: 22
   * Allow ICMP pings only from the public subnet inside the VPC: Type: Custom ICMP – IPv4, Protocol: Echo request, Source: 172.16.100.0/16.

E. Convert your key file to .pem format

1. You can create a new key pair in a .pem format (This is recommended).

If you use Windows/PuTTY for SSH access (.ppk file format), you need to convert your existing private key file to OpenSSH format (.pem). But creating a new key would be easier and faster. If you use Mac or Linux, your key is already in the correct format. For those who want to try converting the key type: Lauch PuTYYgen. This is a key generator/converter utility that was installed on your machine when you installed PuTTY, so you should be able to find it on the list of installed programs in the PuTTY folder. Select “File/Load private key” from the menu, and navigate to your .ppk file (“mykey.ppk”). Choose your .ppk file and click “Open”. Then, select the “Conversions/Export OpenSSH key” menu item, and name the file as “mykey.pem”.

E. Launch an EC2 instance in the public subnet and connect to it

1. Go to the EC2 dashboard and launch an EC2 instance named “Public EC2 Instance” using the “Amazon Linux 2023 AMI” from the AWS Market Place with t2.micro instance type. On the “Network Settings” page, select your VPC (“MyVPC”) for the Network option, and select the VPC’s public subnet for the Subnet option. Set the “Auto-assign Public IP” option to “Enable”. Scroll down to the bottom of the page (Network Interfaces section) and manually assign a private IP address to your instance in the “Primary IP” box (e.g. 172.16.100.4). On the “Security Group” page, select the “Select an existing security group” option and then find and select the security group you have created for the public subnet (“Public SG”). Use your key pair the usual way to log into the EC2 instance. Create a subdirectory “aws” in your home directory.

F. Upload your OpenSSH key to the public EC2 instance

1. Windows users: Open Powershell as an administrator and navigate to your directory where you have stored the key pair file. If you haven’t, open PuttyGen and convert the .pem into an OpenSSH format by clicking on the “conversions” tab. (See the reference document uploaded in the modules section). We can use the “scp” command to securely transfer files between computers over a network. It encrypts the data during transfer, ensuring that sensitive information remains protected. Now use the scp command to transfer the keypair file from our local machine to the directory, “aws” we created in our ssh session of the public ec2.

You can also alternatively use PSFTP which is a similar utility tool. If you have used scp kindly ignore these steps. Instructions are as follows:

Open a command line window (click on the Windows button and then type “cmd+ + Enter) and navigate to the directory where you store your key files, and start a PSFTP session with your EC2 instance. PSFTP is a command-line utility that was installed when you installed PuTTY and can be used for secure file transfer between your local machine and a remote machine (it is a secure FTP client). For more information on PSFTP, please check out https://documentation.help/PuTTY/psftp.html.

Type: psftp -i mykey.ppk ec2-user@<EC2 Public IP address>

Then, change the remote working directory to the “aws” subdirectory. Type: cd aws

Upload the OpenSSH private key file to your EC2 instance. Type: put mykey.pem

Go to your SSH session with your EC2 instance and verify that the mykey.pem file is present in the “aws” subdirectory. Then, change the file permissions of the mykey.pem file so that only you (the logged-in user) can see it. Type: chmod go-rw mykey.pem.

Finally, go back to the Windows command line window and close the PSFTP session. Type: exit.

1. Mac users: Open a terminal window on your local machine and navigate to the directory where you store your key files, and upload it using scp:

Type: scp -i mykey.pem centos@<EC2 Public IP address>:/home/centos/aws/

Go to your SSH session with your EC2 instance and verify that the mykey.pem file is present in the “aws” subdirectory. Then, change the file permissions of the mykey.pem file so that only you (the logged-in user) can see it. Type: chmod go-rw mykey.pem.

G. Launch an EC2 instance in the private subnet and connect to it

1. Go to the EC2 dashboard and launch an EC2 instance named “Private EC2 Instance” using the “Amazon Linux 2023 AMI“ from the AWS Market Place with t2.micro instance type. On the “Network Settings” page, select your VPC (“MyVPC”) for the Network option, and select the VPC’s private subnet for the Subnet option. Attach the key pair (.pem) format which you used for the public subnet. Scroll down to the bottom of the page (Network Interfaces section) and manually assign a private IP address to your instance in the “Primary IP” box (e.g. 172.16.200.4). On the “Security Group” page, select the “Select an existing security group” option and then find and select the security group you have created for the private subnet (“Private SG”).
2. Since this EC2 instance was launched in the private subnet, it is not directly accessible from the Internet. However, the routing and the security groups were set up in such a way that it is accessible from the public subnet, and specifically from the EC2 instance that is running in the public subnet. So to access the “Private EC2 instance”, first you need to access the “Public EC2 instance” and then “hop over” to the “Private EC2 instance”. You already have an SSH session with the public EC2 instance, so now you will log into the private EC2 instance from there.
3. Go to your SSH session with the public EC2 instance and use the ping command to verify connectivity between the two instances. Type: ping 172.16.200.4. You should start receiving successful ping results on the screen.
4. Log into your private EC2 instance from your public EC2 instance using the ssh command.

Type: ssh -i mykey.pem ec2-user@172.16.200.4

You should be able to issue Linux commands to the private EC2 instance now.

E. Clean up after yourself

1. Exit the SSH session (you need to exit twice), terminate the EC2 instances, and delete the VPC “MyVPC”. This will also delete the Internet Gateway, the public and private route tables and the public and private subnets and security groups.
2. Log out of AWS.