**What is a Jumpbox?**

These are the EC2 instances in the public subnet with required access to the private subnets' servers. Generally, we would not want our private servers to be discoverable by anyone outside the VPC. However, we want to access those private servers from the Internet. It is made possible by using a Jumpbox (also called as Bastion Host). It is important to note that the security group attached to the private servers must allow the IP address of the public Jumpbox to connect to it.

# Demo - Accessing Servers in a Private Subnet via Jump Box

Diagram

Description automatically generated

### Create a Jumpbox

In the demo video above, the instructor shows creating a new key-pair, **jumpbox-key.pem**. However, you can use an existing key-pair if available. Remember that key-pairs are specific to an AWS region. Also, the VPC you have to choose while launching the Jumpbox must be the same one in which you have been creating Cloudformation stacks.

Here is the summary of the Jumpbox configuration:

| **Stage** | **Configuration** | **Value** |
| --- | --- | --- |
| 1. | Amazon Machine Image (AMI) | Amazon Linux 2 AMI (HVM), SSD Volume Type Note: You have chosen a **Free Tier Eligible** AMI |
| 2. | Instance Type | t2.micro |
| 3. | Configure Instance Details |  |
|  | a. Number of Instances | 1 |
|  | b. Network | Select the VPC that you created in the previous step |
|  | c. Subnet | Public |
|  | d. Auto-assign Public IP | Enable |
| 4. | Storage | Default |
| 5. | Tags | Name:Jumpbox |
| 6. | Security Group | Create a **new** security group. Add rule to allow SSH type connection on default port 22 from your IP as source |

Once the Jumpbox is up and running, you can connect to it using SSH protocol.

### Connect to the Jumpbox

Since the Jumpbox is a Linux machine, therefore you will have to use SSH protocol to connect to it. Remember that the flow of connection will be

***Your local computer (Mac/Windows/Linux) → Jumpbox(CentOS Linux) → Private servers (Ubuntu Linux).***

If you are running a Windows computer locally, then you would have to convert the **jumpbox-key.pem** to **jumpbox-key.ppk** using the PuTTy utility and use the same PuTTy to connect to the Jumpbox.

### Test the Jumpbox

Recall that the instructor has the following two different login keys: 1). **jumpbox-key.pem** for the Jumpbox and 2). **private-server-devops-key.pem** for the private servers. Let's test if you are able to connect to the private servers via a Jumpbox:

1. Copy the public IP address of the Jumpbox, say **3.17.80.159**
2. Copy and paste the **private servers' login key file** from your local computer to the Jumpbox. Run the following command from your local terminal (replace the file names and the IP address as applicable to you):
3. scp -i jumpbox-key.pem private-server-devops-key.pem ec2-user@3.17.80.159:/home/ec2-user/private-server-devops-key.pem
4. SSH login to the Jumpbox:
5. ssh ec2-user@3.17.80.159 -i jumpbox-key.pem
6. Copy the private IP address of any private server, say **10.0.2.74**.
7. Once you are logged into the Jumpbox, confirm if you have the **private-server-devops-key.pem** key available in the home directory, and then change the access-mode of the key file. Later, try to **SSH login to the private server**:
8. ls
9. *# you must see the private-server-devops-key.pem file*
10. chmod 400 private-server-devops-key.pem
11. ssh -i "private-server-devops-key.pem" ubuntu@10.0.2.74

*Recall that the default user name for a Linux system is ec2-user and for an Ubuntu system is ubuntu.*

1. Lastly, check the status of the running web server in the private instance, as shown in the snapshot below:

Text

Description automatically generated

Checking if the Apache server is running in the Ubuntu machine, using service apache2 status command.

Similarly, you can SSH login to the second private server from the Jumpbox. You have to replace the private IP address of the second server in the commands above.