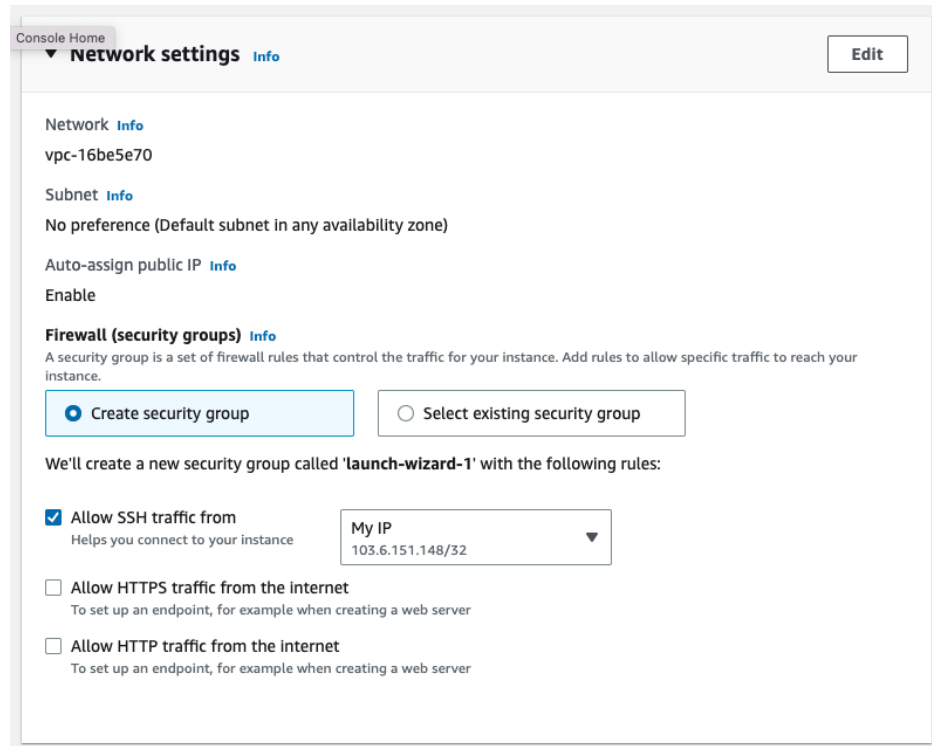


## 1. Connect 2 EC2 Instances Together

1. From your AWS console, create 1 EC2 instance with the name format <YourName>ConnectServer1 e.g. DannyConnectServer1
2. Choose a Linux-based OS of your liking as the AMI
3. Choose a t2.micro instance type for this EC2 instance
4. Under Network Settings, create a new Security Group, but in this case, select:
  - a. SSH - Allow SSH traffic from My IP
  - b. Do not allow any HTTP/ HTTPS connections



Console Home

▼ **Network settings** Info Edit

Network Info  
vpc-16be5e70

Subnet Info  
No preference (Default subnet in any availability zone)

Auto-assign public IP Info  
Enable

**Firewall (security groups)** Info  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group ☐ Select existing security group

We'll create a new security group called 'launch-wizard-1' with the following rules:

☒ Allow SSH traffic from  
Helps you connect to your instance

My IP  
103.6.151.148/32 ▼

☐ Allow HTTPS traffic from the internet  
To set up an endpoint, for example when creating a web server

☐ Allow HTTP traffic from the internet  
To set up an endpoint, for example when creating a web server

- c.
5. Choose your key pair or create a new key pair.
  6. Launch your instance.
  7. Repeat step 1 to 6 again, this time with another EC2 instance name with the format <YourName>ConnectServer2 e.g. DannyConnectServer2 with another new Security Group.
  8. You should now have 2 EC2 instances. Connect to both from your local machine on 2 separate windows.

```
luqmannurhakimbintajuddin@Luomans-MacBook-Pro Downloads % ssh -i "luqman_ec2_jp_keypair.pem" ec2-user@ec2-54-253-156-18.ap-southeast-2.compute.amazonaws.com
The authenticity of host 'ec2-54-253-156-18.ap-southeast-2.compute.amazonaws.com (54.253.156.18)' can't be established.
ED25519 key fingerprint is SHA256:e1rxmSLGIO/IY3mwdxmF1PqxZV5dUytVsKmq6Pz5J6k.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-253-156-18.ap-southeast-2.compute.amazonaws.com' (ED25519) to the list of known hosts.

__|__|__|
_| (___/  Amazon Linux 2 AMI
___|\\___|___|

https://aws.amazon.com/amazon-linux-2/
7 package(s) needed for security, out of 8 available
Run "sudo yum update" to apply all updates.
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory
[ec2-user@ip-172-31-3-0 ~]$
```

9.

10. Find the Public IP of your ConnectServer2, head back to your CLI for ConnectServer1, and enter the command `ping <Public IP ConnectServer2>`. You should notice that the ping command does not return a successful connection output.

Instances (2) info

Find instance by attribute or tag (case-sensitive)

Connect X Clear filters

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
<input type="checkbox"/>	DannyConnectServer1	i-0ac8f88ee6a559126	Running	t2.micro	2/2 checks passed	No alarms	ap-southeast-2a	ec2-54-253-156-18.ap-...	54.253.156.18
<input type="checkbox"/>	DannyConnectServer2	i-0c12a1cb91309190e	Running	t2.micro	2/2 checks passed	No alarms	ap-southeast-2a	ec2-13-239-54-63.ap-s...	13.239.54.63

```
luqmannurhakimbintajuddin@Luomans-MacBook-Pro Downloads % ssh -i "luqman_ec2_jp_keypair.pem" ec2-user@ec2-54-253-156-18.ap-southeast-2.compute.amazonaws.com
The authenticity of host 'ec2-54-253-156-18.ap-southeast-2.compute.amazonaws.com (54.253.156.18)' can't be established.
ED25519 key fingerprint is SHA256:e1rxmSLGIO/IY3mwdxmF1PqxZV5dUytVsKmq6Pz5J6k.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-253-156-18.ap-southeast-2.compute.amazonaws.com' (ED25519) to the list of known hosts.

__|__|__|
_| (___/  Amazon Linux 2 AMI
___|\\___|___|

https://aws.amazon.com/amazon-linux-2/
7 package(s) needed for security, out of 8 available
Run "sudo yum update" to apply all updates.
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory
[ec2-user@ip-172-31-3-0 ~]$ ping 13.239.54.63
PING 13.239.54.63 (13.239.54.63) 56(84) bytes of data.
^C
--- 13.239.54.63 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5121ms

[ec2-user@ip-172-31-3-0 ~]$
```

11. This is because we did not specify connectivity rules from the Security Group for EC2 connectivity.
12. To enable ping, from your AWS console, edit the Inbound Rules for the Security Group of your ConnectServer2.
- Allow ICMP - IPv4 on your Inbound rules

Inbound rules (2)

Filter security group rules

Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
-	sgr-04914363c6ae20dc7	IPv4	All ICMP - IPv4	ICMP	All	0.0.0.0/0	-
-	sgr-097f4b4901e1bf3	IPv4	SSH	TCP	22	103.6.151.148/32	-

b.

13. Once done, try to ping again from the first EC2 instance.

```
[ec2-user@ip-172-31-3-0 ~]$ ping 13.239.54.63
PING 13.239.54.63 (13.239.54.63) 56(84) bytes of data.
64 bytes from 13.239.54.63: icmp_seq=1 ttl=254 time=0.490 ms
64 bytes from 13.239.54.63: icmp_seq=2 ttl=254 time=0.652 ms
64 bytes from 13.239.54.63: icmp_seq=3 ttl=254 time=0.564 ms
64 bytes from 13.239.54.63: icmp_seq=4 ttl=254 time=0.518 ms
^C
--- 13.239.54.63 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3070ms
rtt min/avg/max/mdev = 0.490/0.556/0.652/0.061 ms
[ec2-user@ip-172-31-3-0 ~]$
```

14.

15. This is an example of a ping request utilizing ICMP. If you require SSH connectivity between the 2 servers, you will need to follow the below steps.

16. From your CLI for both EC2 Instances, you will need to create SSH keys. An SSH key is an access credential in the SSH protocol.

- To do this, run "ssh-keygen"
- You do not need to provide any inputs for the prompt; simply press enter until you receive the key's random art image like below:

```
[ec2-user@ip-172-31-3-0 ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ec2-user/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ec2-user/.ssh/id_rsa.
Your public key has been saved in /home/ec2-user/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:0J8noWFXlj+BrQmfHkZvVTAgCRocomOVNrQ+SiEQfb4 ec2-user@ip-172-31-3-0.ap-sou
theast-2.compute.internal
The key's randomart image is:
+---[RSA 2048]---+
|.. oo.. . o+ |
| .++o.o. o.++ o |
|. +oo... + ++.* o |
|o.. . o = oBo= .|
|... . S +o.oo..|
|... E o. . |
|.. |
|.. |
|o.. |
+---[SHA256]---+
[ec2-user@ip-172-31-3-0 ~]$
```

c.

d. Ensure you have done this for both EC2 instances.

17. Your SSH keys will be accessible in the directories:

- .ssh/id\_rsa -> private key

- b. `ssh/id_rsa.pub` -> public key
- 18. On ConnectServer1, copy the content of the public key by running the below command:
  - a. `cat ~/.ssh/id_rsa.pub`
  - b. [Copy the content of this key to your clipboard]
- 19. On ConnectServer2, you will need to paste the content of ConnectServer1's `id_rsa.pub` under the file `.ssh/authorized_keys`
  - a. `cat >> ~/.ssh/authorized_keys`
  - b. [paste your clipboard contents]
  - c. [ctrl+d to exit]
- 20. To SSH to ConnectServer2 from ConnectServer1, run the command below in ConnectServer1:
  - a. `ssh -i ~/.ssh/id_rsa <PrivateIP of ConnectServer2>`
- 21. You'll notice that you are unable to connect to the ConnectServer2 EC2 instance. This is because you have not explicitly allowed SSH connection from ConnectServer1. We will need to update the Security Group of ConnectServer2 again.
  - a. Instead of Allowing SSH from only My IP, update to Allow from Anywhere.
- 22. Try connecting again via the command:
  - a. `ssh -i ~/.ssh/id_rsa <PrivateIP of ConnectServer2>`
- 23. This should allow you to connect from ConnectServer1 to ConnectServer2 via SSH.
- 24. Questions to ponder:
  - a. What firewall rules are needed for instances to send files across?
- 25. Once done, terminate your instances created.

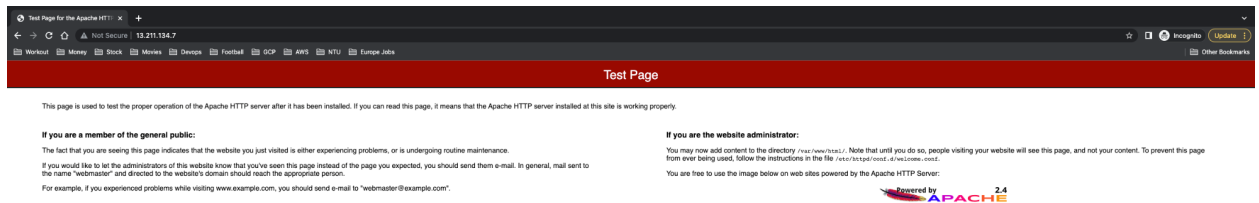
## 2. Create a Simple EC2 Web Server with Apache

1. Create an EC2 Instance with the naming convention `<YourName>WebServer1` e.g. `DannyWebServer1`
2. Choose a Linux-based OS of your liking as the AMI
3. Choose a `t2.micro` instance type for this EC2 instance
4. Under Network Settings, create a new Security Group, but in this case, select:
  - a. SSH - Allow SSH traffic from My IP
  - b. HTTP/ HTTPS - Allow from Anywhere
5. Choose your key pair or create a new key pair.
6. Launch your EC2 instance.
7. Once done, Connect to your EC2 instance from your local machine CLI.
8. Run a yum update on your EC2 instance. This updates any packages that are not up-to-date
  - a. `sudo yum update -y`
9. Install the Apache Web Server
  - a. `sudo yum install -y httpd`

10. Start the Apache Web Server

a. `sudo systemctl start httpd`

11. You should now see this Apache test page if you enter the Public IPv4 address in your Internet Browser.



12. Let's customize this page. To do this, on your EC2 instance, go to the directory /var/www/html. To do this, run

a. `sudo touch index.html`

13. Edit using a editor e.g.

a. `sudo vi index.html`

b. Type "i" to enter into edit mode, and then paste the below snippet, adding your name as well.

c. `<!DOCTYPE html>`

`<html>`

`<body>`

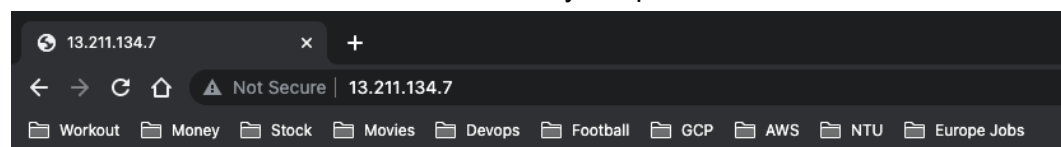
`<h1>Hello World !!</h1>`

`<p>My name is _____</p>`

`</body>`

`</html>`

d. To exit the edit mode, click "esc" followed by ":wq!" and enter.



**Hello World !!**

My name is Danny

e.

f. This is what you should expect.