Security in the cloud

Assignment-1

Security Groups

Assignment Brief

|  |  |
| --- | --- |
| Student Name/ID Number | Yazan Atary / 22034818 |
| **Course Title** | **Security in the cloud** |
| Academic Year | 2023/24 |
| **Assignment Title** | **Security Groups** |
| Issue Date | **15/4/2024** |
| Submission Date | **16/4/2024** |
| IV Name & Date |  |

|  |
| --- |
| **Submission Format** |
| The submission is in this form as clear screen shot under every question.  You can use this website to add your screen shots to the PDF file : <https://www.ilovepdf.com/blog/how-to-insert-an-image-into-pdf> |

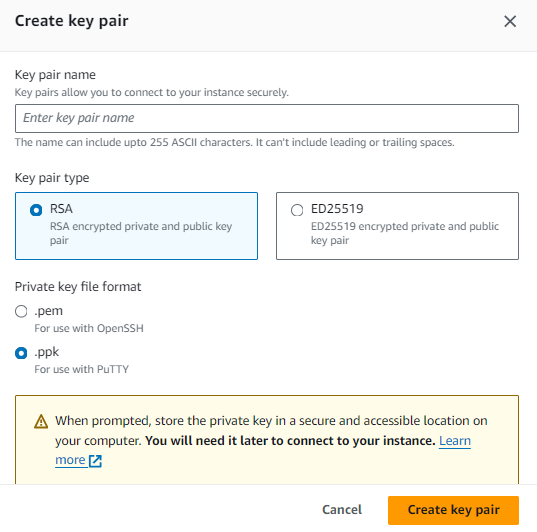
1

**Security Groups**

A security group acts as a virtual firewall for your EC2 instances and the role of the security group is to block all incoming traffic except the ports, protocols and sources that you specify. The rules of a security group control the inbound traffic that's allowed to reach the resources that are associated with the security group. The rules also control the outbound traffic that's allowed to leave them.

1. Creating an EC2 instance

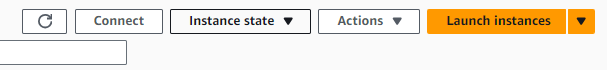
* Go to EC2 service, launch instances.
* Name the instance.
* Choose AMI “amazon Linux”.
* Create new key pair (.ppk).

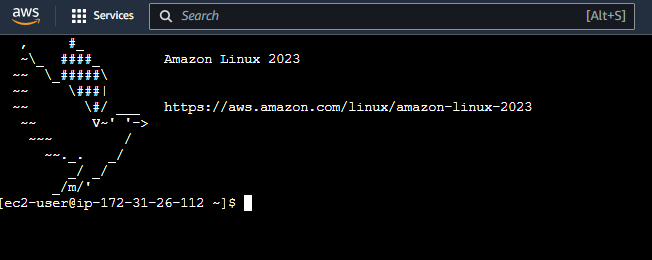


* Leave everything else at default settings and launch the instance.

1. Installing httpd

* Connect to your instance by selecting the connect option.





* Use the following commands to install and start the HTTP server daemon

**#sudo su**

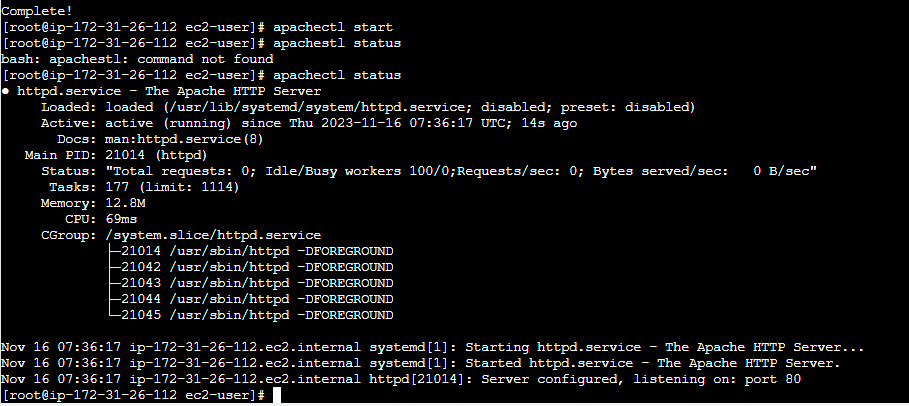
**#yum update -y**

**#yum install -y httpd**

**#apachectl start**

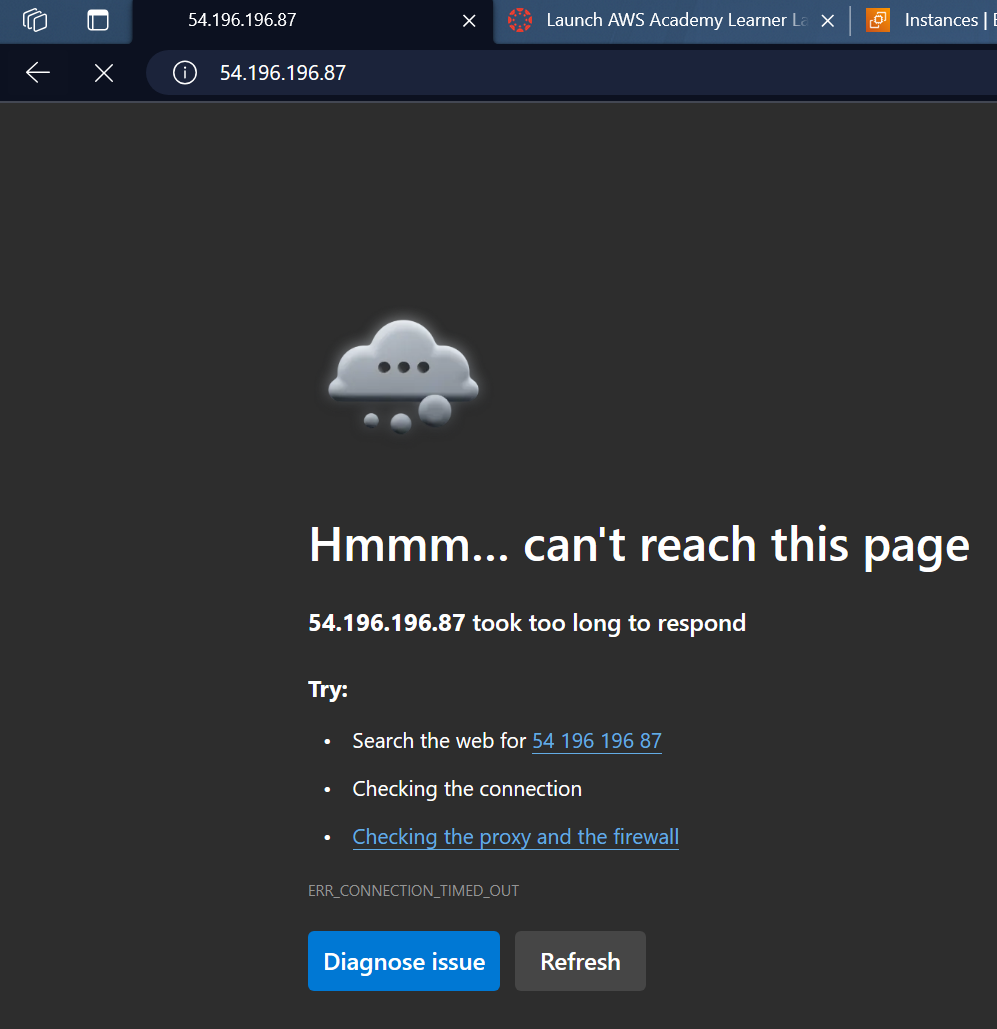
* Check the status of your apache server

**#apachectl status**

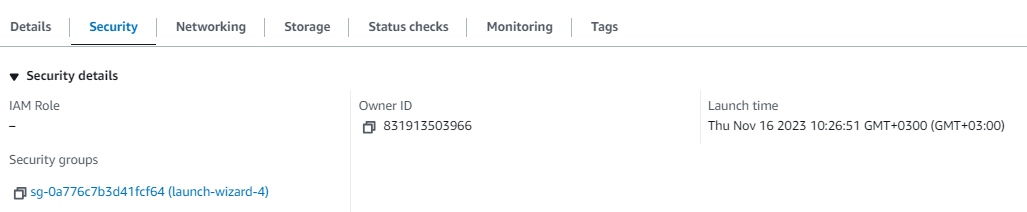


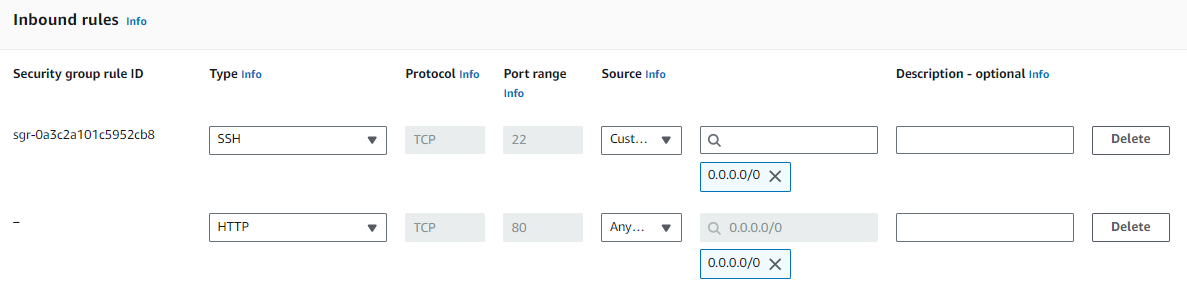
* Now, let’s try to open our web page on the browser.
* Initiate a request from your web browser to the server (<http://YOUR_MACHINE_IP_ADDRESS>)
* Can you enter the website? NO

Provide a screenshot



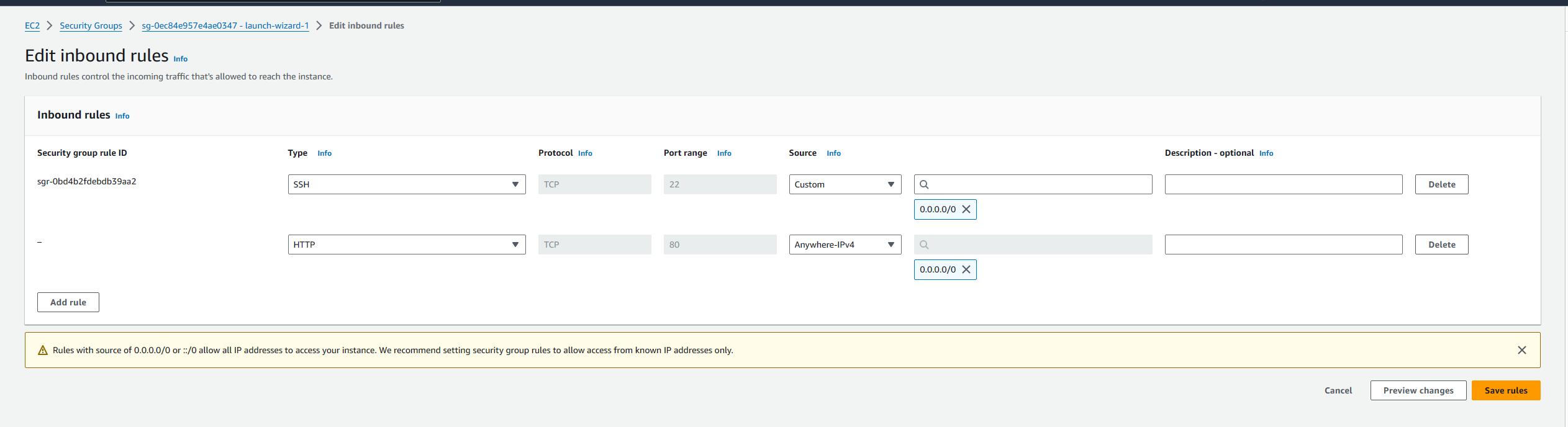
1. Editing security group rules

* Let's edit the inbound rules of the EC2 instance. Go to the security settings of the EC2 instance and enter the security group field.
* Click on “Edit inbound rules”, and add the following rule.



* Now, save settings and let’s try to open our web page again.

Provide a screenshot

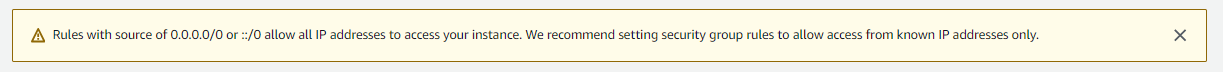


A screenshot of a computer

Description automatically generated

1. Securing our SSH connections

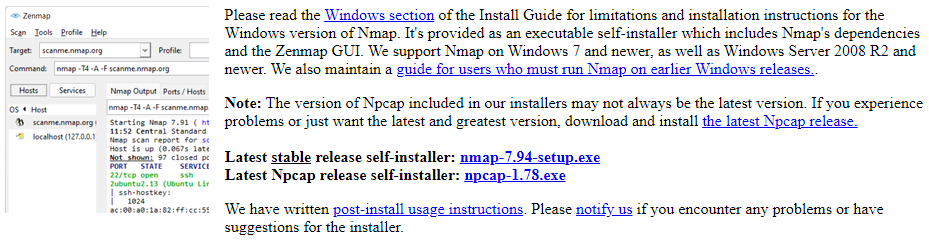
Notice the inbound rule that was implicitly assigned to the SSH rule when we created the EC2 instance. You will observe that the rule allows all IP addresses to access the instance (**0.0.0.0/0**), posing a potential security risk. See the following warning from AWS:



Let's explore this further!

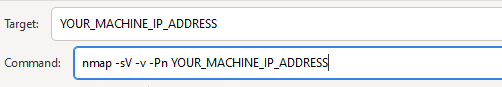
* Install the Nmap tool for windows from the following URL:

[Download the Free Nmap Security Scanner for Linux/Mac/Windows](https://nmap.org/download)



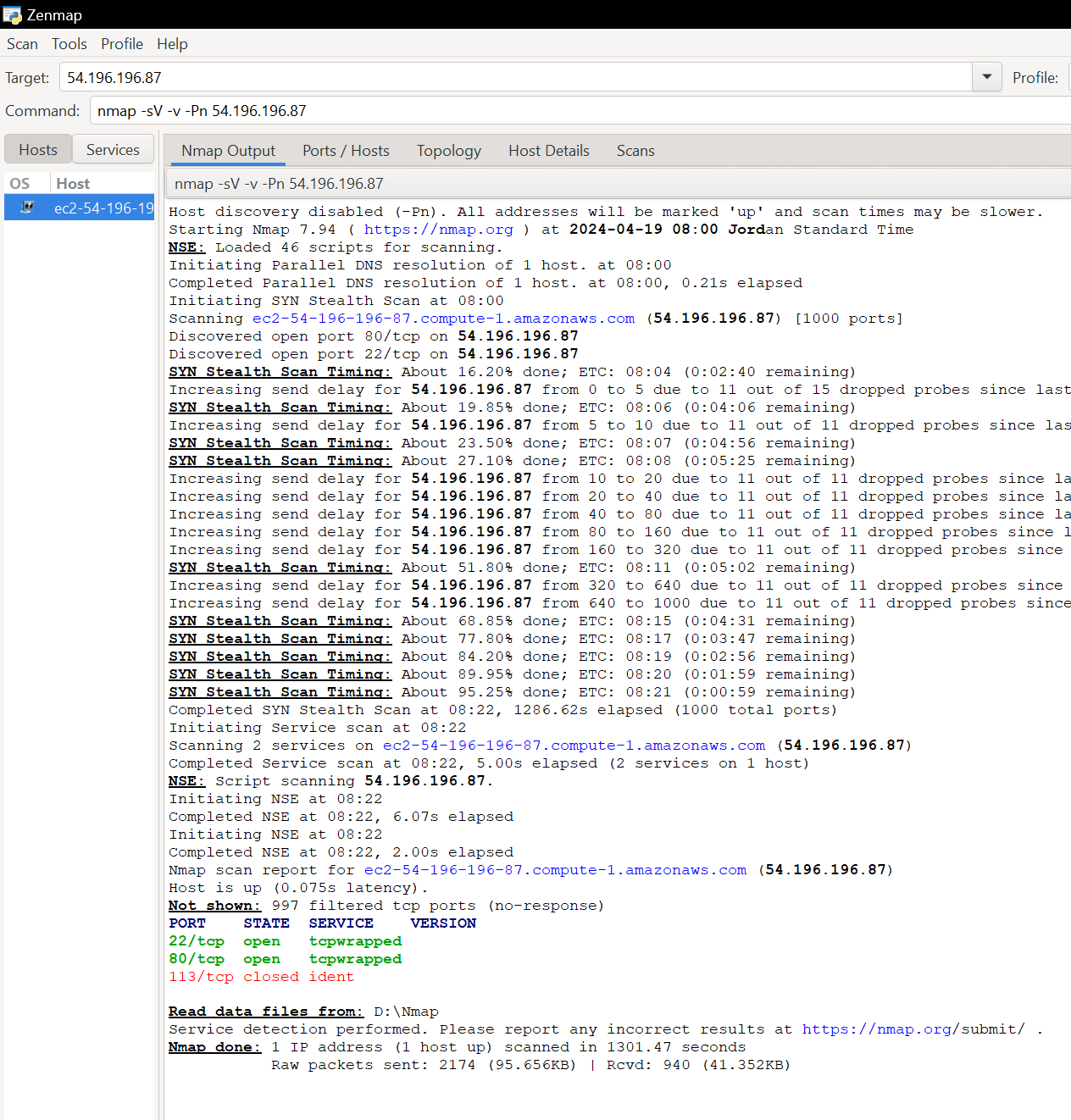
* Now, open Nmap and try to perform the following scan:

**nmap -sV -v -Pn YOUR\_MACHINE\_IP\_ADDRESS**



* Now, review the results of this scan and provide an explanation, highlighting the risks of the previous rule.

Provide a screenshot along with your explanation of the results and the associated risks



**Open Ports:**

* + Port 22: OpenSSH is running on this port.
  + Port 80: Apache HTTP server is running on this port.

As for the question about the risk of adding an inbound rule with the IP address **0.0.0.0/0** for port 80 (HTTP) in the security group:

1. Allowing traffic from **0.0.0.0/0** means that any device or user on the internet can access the specified port (in this case, port 80) on my server.
2. For a web server (port 80), this is common to allow public access to the website. However, it should be done cautiously and only when necessary.
3. It's crucial to ensure that the services (like Apache HTTP server) are properly configured and secured to handle public access. This includes considerations for web application security, server hardening, and regular updates.

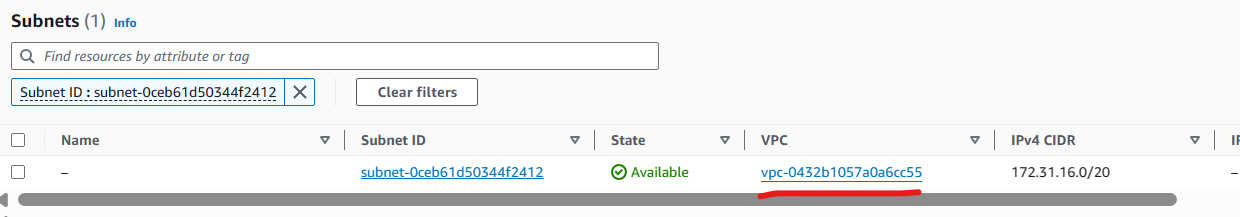
Expanding on the risks associated with an open port configured with the IP address range 0.0.0.0/0, let's specifically consider the SSH port:

1. Unrestricted Access to SSH (Port 22): Allowing traffic from 0.0.0.0/0 implies that any device or user on the internet can attempt to establish an SSH connection to the server, potentially leading to unauthorized access and security breaches.
2. Security Considerations for SSH: While it's common to open the SSH port for remote administration, doing so with a broad IP address range demands heightened caution. This configuration increases the exposure of the server to a wider range of potential threats, including malicious login attempts and brute-force attacks.

-----

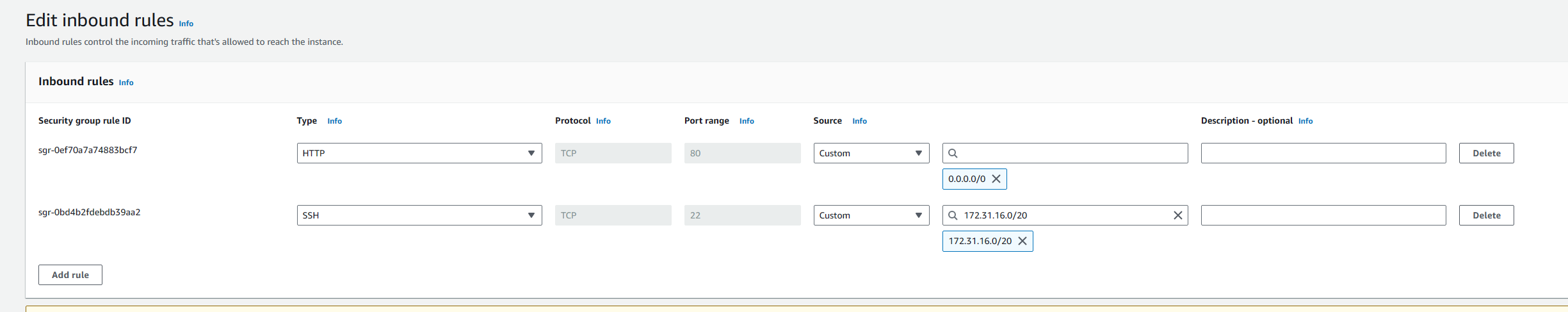
* To further secure our SSH connection, let's assign a specific range of IP addresses to the rule instead of leaving it open.
* Now, you need to edit the SSH rule and include only the IP addresses that belong to the same subnet as your EC2 instance for accessing the machine. Go to the Networking tab of your EC2 instance, discover the IP address range of your instance's subnet.

Provide a screenshot of the IPv4 CIDR of your subnet.



* Replace 0.0.0.0/0 in the rule with this IPv4 CIDR.

Provide a screenshot of the edited rule.



* Now, rerun the scan again using Nmap and examine the results. Subsequently, explain the significance of specifying a specific range of IP addresses and discuss how this practice mitigates risks.

Provide a screenshot of the edited rule.

