STEP-BY-STEP CONFIGURATION:

Step 1: Install UFW

UFW is typically installed by default on Ubuntu, but if it's not installed, you can install it using the following command:

sudo apt update

sudo apt install ufw

Step 2: Check UFW Status

Before starting, check if UFW is active or inactive:

sudo ufw status

If it's inactive, you'll see output like:

Status: inactive

Step 3: Enable UFW

Before enabling UFW, it's a good practice to allow SSH connections (port 22), especially if you're configuring the firewall on a remote server. If UFW is enabled without allowing SSH, you could lock yourself out.

sudo ufw allow ssh

Now, enable UFW:

sudo ufw enable

You should see a confirmation message:

Firewall is active and enabled on system startup

Step 4: Configure Basic Firewall Rules

Here are some basic firewall rules to get you started. Adjust the ports and services according to your needs.

Allow Common Services

- SSH (Port 22):

sudo ufw allow ssh

- HTTP (Port 80):

sudo ufw allow http

- HTTPS (Port 443):

sudo ufw allow https

Allow Specific Port

If you need to allow traffic on a specific port, for example, port 8080:

sudo ufw allow 8080

Allow a Range of Ports

If you want to allow a range of ports, for example, ports 1000 to 2000:

sudo ufw allow 1000:2000/tcp

Allow Specific IP Address

To allow traffic from a specific IP address:

sudo ufw allow from 192.168.1.100

Deny Specific IP Address

To deny traffic from a specific IP address:

sudo ufw deny from 192.168.1.100

Step 5: Check UFW Status and Rules

To check the status of UFW and see the list of rules:

sudo ufw status verbose

Step 6: Disable UFW (if needed)

If you ever need to disable UFW:

sudo ufw disable

Step 7: Testing the Firewall

To ensure the firewall is working as expected, you can perform the following tests:

1. Test SSH Access

From a different machine, try to SSH into your Ubuntu system. If SSH is allowed, you should be able to log in.

2. Test HTTP/HTTPS Access

Try accessing your web server (if you have one running) using a web browser. If the ports are allowed, the website should load.

3. Test a Denied Port

Attempt to access a port that you have specifically denied. For example, if you denied port 8080, try accessing it:

telnet <your-server-ip> 8080

You should see a connection refused or timeout, indicating that the port is blocked.

4. Test IP Restrictions

If you have restricted access to specific IPs, try accessing the server from an allowed and a denied IP to ensure the rules are working.

Step 8: Log and Monitor UFW

UFW logs can be useful for monitoring and debugging. UFW logs are typically stored in the /var/log/ directory.

Enable logging:

sudo ufw logging on

To view the logs:

sudo tail -f /var/log/ufw.log

This command will continuously display new log entries as they are generated.

Step 9: Additional UFW Commands

Here are a few additional UFW commands that might be useful:

Reset UFW to Defaults:

sudo ufw reset

Delete a Specific Rule:

sudo ufw delete allow 8080

List Available Application Profiles:

sudo ufw app list

Allow a Specific Application Profile:

sudo ufw allow 'Nginx Full'

Conclusion

By following these steps, you have successfully installed, configured, and tested UFW on your Ubuntu system. UFW helps you easily manage the firewall rules and protects your system from unauthorized access and potential threats. Regularly review and update your firewall rules to ensure your system remains secure.