Urvish Jaiswal

Firewall Management

For Beginners

Master Firewall Setup in Linux.

Step by step

What is Firewall?

A firewall is like a security guard for your computer or network. Just as a security guard at a building checks IDs and only lets authorized people enter, a firewall monitors and controls incoming and outgoing network traffic based on predetermined security rules.

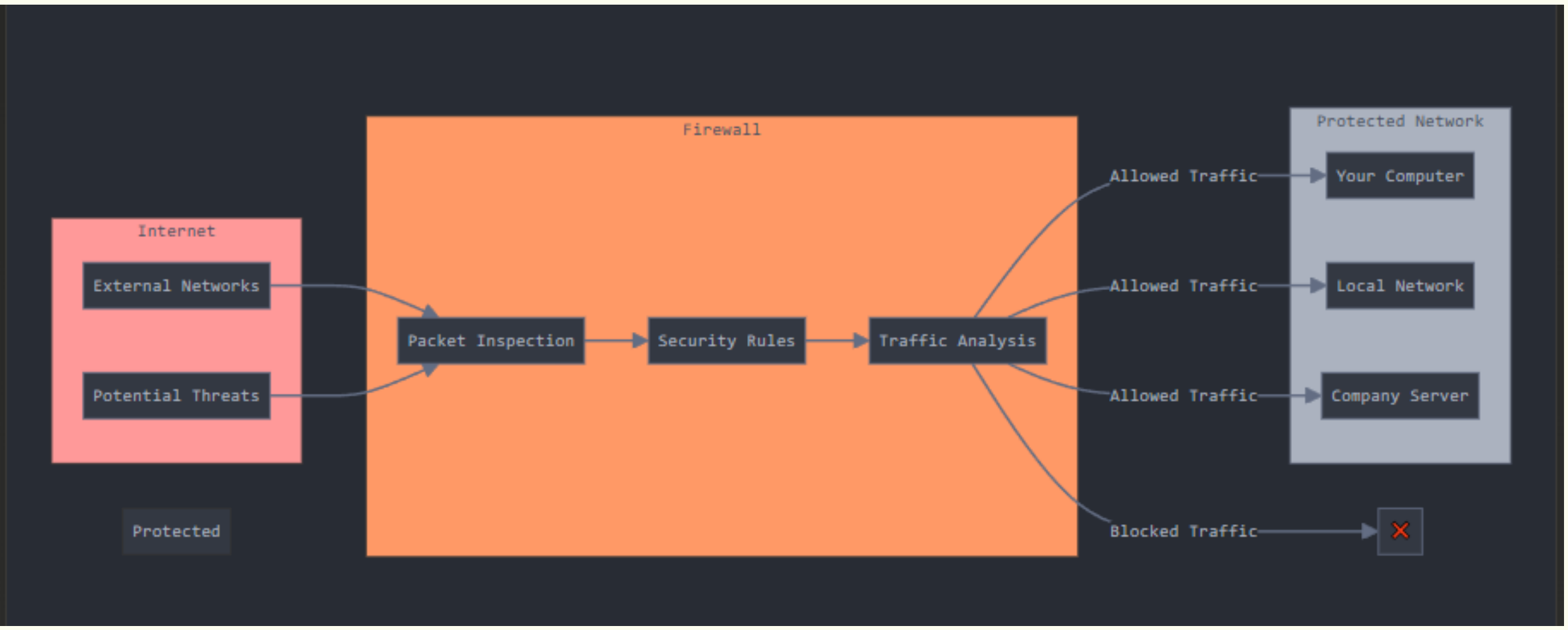
Think of it this way: When you're using the internet, data is constantly flowing in and out of your device. Some of this data is good (like when you're browsing websites or checking email), but some could be harmful (like malware or unauthorized access attempts). The firewall stands between your device/network and the internet, examining all this traffic and deciding what to allow through and what to block.

Diagram for better understanding:

Swipe for more

Urvish Jaiswal

Firewall Management



Now Imagine your house. You have a front door with a lock, and maybe a security guard at the entrance of your neighborhood. A firewall works just like that, but for your computer or network.

There are two main types of firewalls:

Software Firewall

1. It's like having a security guard inside your house
2. It's a program installed on your computer (like

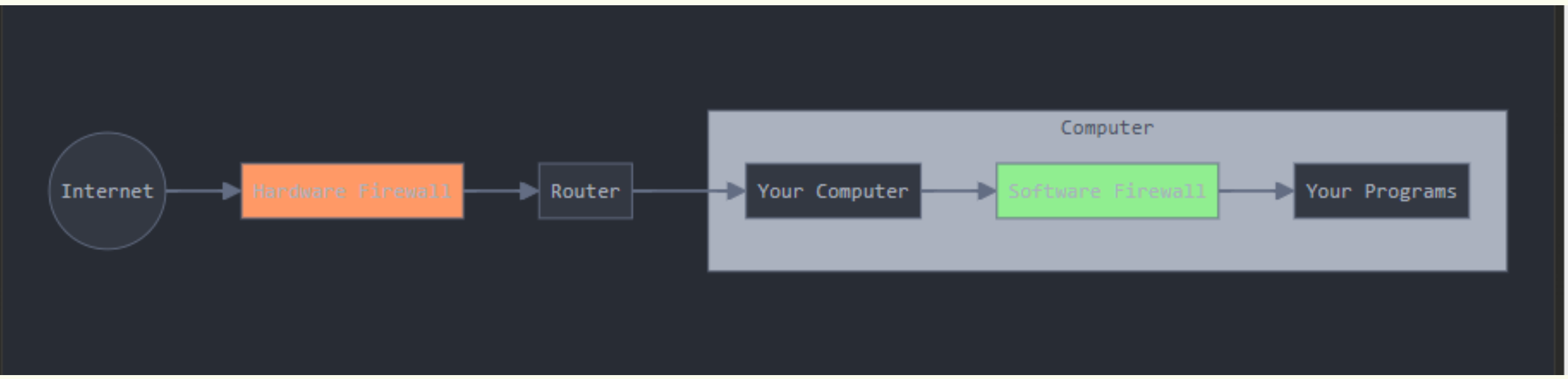
Swipe for more

Windows Defender)
3. Protects just your computer
4. Usually free and easy to use

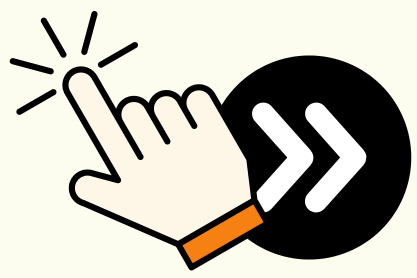
Hardware Firewall

1. It's like the security guard at your neighborhood entrance
2. It's an actual physical device
3. Protects your entire home/office network
4. Usually found in routers or as separate device

Diagram for better understanding:



Swipe for more



Think of it this way:

1. Bad stuff tries to come in from the internet
2. Hardware firewall checks everything coming into your network
3. Software firewall does a final check before letting anything reach your programs

That's really all there is to it! Just like security guards, firewalls keep the bad stuff out while letting the good stuff in.

Swipe for more

Managing Firewall Service

Step 1: Check if Firewall is Active

1. Open the Terminal

- First, open your terminal or connect to your CentOS 9 server via SSH. This is where you'll enter the commands to manage the firewall.

2.Check the Firewall Status-

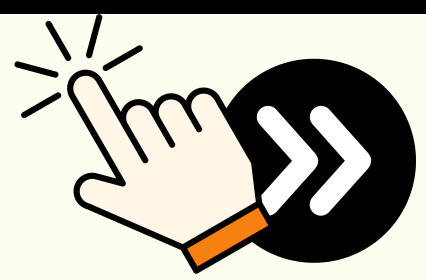
- You can check if the firewall is already running by using:

sudo systemctl status firewalld

```
[root@localhost ~]# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
     Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; preset: enabled)
     Active: active (running) since Wed 2025-01-08 15:44:53 IST; 1min 54s ago
       Docs: man:firewalld(1)
   Main PID: 1007 (firewalld)
      Tasks: 2 (limit: 14212)
     Memory: 42.5M
        CPU: 1.267s
     CGroup: /system.slice/firewalld.service
             └─1007 /usr/bin/python3 -s /usr/sbin/firewalld --nofork --nopid

Jan 08 15:44:52 localhost systemd[1]: Starting firewalld - dynamic firewall daemon...
Jan 08 15:44:53 localhost systemd[1]: Started firewalld - dynamic firewall daemon.
```

Swipe for more



If it says "active (running)," your firewall is already up and running. If not, we'll need to start it.

Step 2: Start and Enable Firewall

1. Start the Firewall Service

If your firewall isn't running, you can start it with this command:

sudo systemctl start firewalld

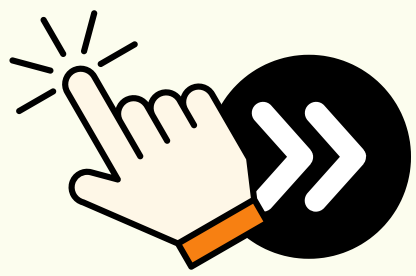
```
[root@localhost ~]# systemctl start firewalld
[root@localhost ~]#
```

2. Enable Firewall on Boot

To make sure your firewall starts automatically whenever your server reboots, use:

sudo systemctl enable firewalld

Swipe for more



```
t@localhost ~]#  systemctl enable firewalld
t@localhost ~]#
```

This saves you from having to start it manually every time.

Step 3: Basic Firewall Commands

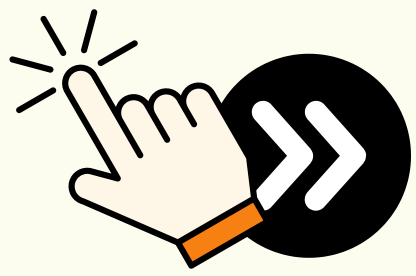
1. **Check Firewall Rules**

   To view the active firewall rules, use:

sudo firewall-cmd --list-all

```
[root@localhost ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client dns http ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Swipe for more



2.Reload Firewall Rules

After making changes to the firewall, reload the service to apply the changes:

sudo firewall-cmd --reload

```
[root@localhost ~]# firewall-cmd --reload
success
[root@localhost ~]#
```

Step 5: Allow Specific Services

1. **List Available Services**

   To see which services can be allowed through the firewall:

sudo firewall-cmd --get-services

Swipe for more

2.Allow a Service

To allow a service (e.g., HTTP), run:

```
[root@localhost ~]# firewall-cmd --permanent --add-service=http
Warning: ALREADY_ENABLED: http
success
```

3.Remove a Service

To disallow a service:

```
[root@localhost ~]# firewall-cmd --permanent --remove-service=http
success
[root@localhost ~]#
```

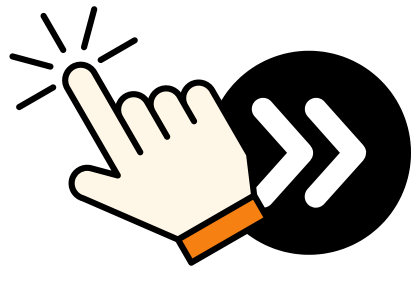
Step 6: Advanced Firewall Configurations

1. Add a Rich Rule

For more complex rules, such as allowing traffic from a specific IP:

sudo firewall-cmd --permanent --add-rich-rule="rule family='ipv4' source address='192.168.1.100' port port=80 protocol=tcp accept"

Swipe for more



```
[root@localhost ~]# sudo firewall-cmd --permanent --add-rich
success
```

PS: Sorry the command is long enough to come under the doc

2. List All Active Zones

To see active zones and their rules:

```
[root@localhost ~]# firewall-cmd --get-active-zones
public
  interfaces: enp0s3
[root@localhost ~]#
```

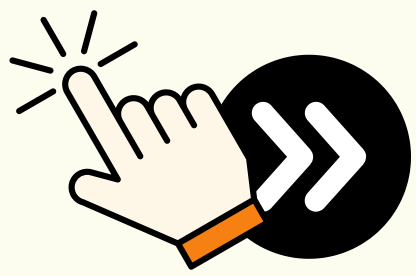
3. List All Zones

To see all listed zones:

firewall-cmd --get-zones

```
[root@localhost ~]# firewall-cmd --get-zones
block dmz drop external home internal libvirt libvirt-routed nm-shared public trusted work
[root@localhost ~]#
```

Swipe for more



Step 7: To Add or Remove a Service Permanently

1. Add a Service

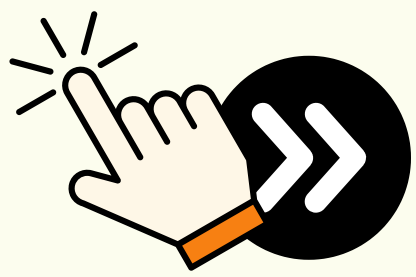
To see active zones and their rules:If you want to allow a specific service (like SSH or HTTP) through your firewall, you can add it permanently with this command:

sudo firewall-cmd --permanent --add-service=ssh

```
[root@localhost ~]# firewall-cmd --permanent --add-service=ssh
success
```

This opens the door for that service. For example, allowing SSH lets you remotely log into your server.

Swipe for more



2.Remove a Service

If you no longer need a service to have access, you can remove it:

sudo firewall-cmd --permanent --remove-service=ssh

```
[root@localhost ~]# sudo firewall-cmd --permanent --remove-service=ssh
success
```

This closes the door for that service, adding an extra layer of security.

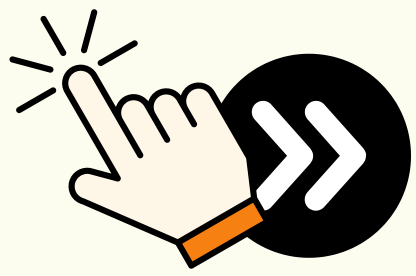
3.Reload to Apply Changes

Don’t forget to reload the firewall to make your changes active:

sudo firewall-cmd --reload

```
[root@localhost ~]# firewall-cmd --reload
success
[root@localhost ~]#
```

Swipe for more



Step 8: To Add or Remove a Port

1. Add a Port

Need to allow traffic through a specific port? Use this command::

sudo firewall-cmd --permanent --add-port=8080/tcp

```
[root@localhost ~]# firewall-cmd --permanent --add-port=8080/tcp
success
```

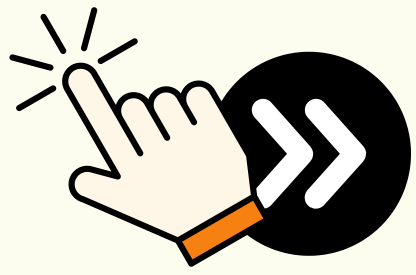
This opens the specified port (in this case, 8080) for TCP traffic.

2.Remove a Port

To block traffic on a port you previously opened:

sudo firewall-cmd --permanent --remove-port=8080/tcp

Swipe for more



```
[root@localhost ~]# sudo firewall-cmd --permanent --remove-port=8080/tcp
success
[root@localhost ~]#
```

This closes the port, stopping any incoming traffic on it.

3.Reload to Apply Changes

Don’t forget to reload the firewall to make your changes active:

sudo firewall-cmd --reload

```
[root@localhost ~]# firewall-cmd --reload
success
[root@localhost ~]#
```

Swipe for more

Step9 : To Block Outgoing Traffic to an IP or URL

1. Block Traffic to an IP

To block your server from making outgoing connections to a specific IP:

sudo firewall-cmd --permanent --add-rich-rule="rule family='ipv4' destination address='192.168.1.100' drop"

```
[root@localhost ~]# firewall-cmd --permanent --add-rich-rule="rule family
success
[root@localhost ~]#
```

This stops any outgoing traffic to that IP address.

2.Block Traffic to a URL

Firewalls usually block based on IPs, but you can block a domain by resolving it to an IP and then

Swipe for more

blocking that IP.

3. Remove the Block

To remove the block on the IP:

sudo firewall-cmd --permanent --remove-rich-rule="rule family='ipv4' destination address='192.168.1.100' drop"

```
oot@localhost ~]# firewall-cmd --permanent --remove-rich-rule="rule family='i
ccess
oot@localhost ~]#
```

This start any outgoing traffic to that IP address.

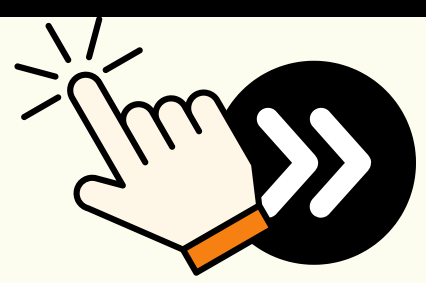
4.Reload to Apply Changes

Don’t forget to reload the firewall to make your changes active:

sudo firewall-cmd --reload

```
[root@localhost ~]# firewall-cmd --reload
success
```

Swipe for more



Step 10: To Block ICMP Incoming Traffic

1. Block ICMP (Ping) Requests

To stop your server from responding to ICMP (ping) requests:

sudo firewall-cmd --permanent --add-icmp-block=echo-request

```
[root@localhost ~]# sudo firewall-cmd --permanent --add-icmp-block=echo-request
success
[root@localhost ~]#
```

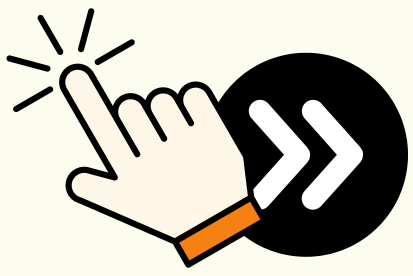
This stops any outgoing traffic to that IP address.

2.Remove the ICMP Block

If you want to allow ping requests again:

sudo firewall-cmd --permanent --remove-icmp-block=echo-request

Swipe for more



```
[root@localhost ~]# sudo firewall-cmd --permanent --remove-icmp-block=echo-request
success
[root@localhost ~]#
```

3.Reload to Apply Changes

Don’t forget to reload the firewall to make your changes active:

sudo firewall-cmd --reload

```
[root@localhost ~]# firewall-cmd --reload
success
[root@localhost ~]#
```

.....

Congratulations on mastering firewall management on CentOS 9🥳! You've fortified your server's security by skillfully managing services, ports, and traffic. With this knowledge, you're well-prepared to tackle network challenges confidently. Enjoy the peace of mind that comes with a secure and efficient system!

Swipe for more

You did a phenomenal job, and many congratulations🥳 on successfully setting up your firewall🎉.

For some, this may have been their first-ever firewall setup, and let me tell you, it's a big accomplishment!

I feel proud, and you should too, even if it was a minor step like opening a config file. Feel proud of every single action you take toward your goal, no matter how small it may seem.

Every step is a vote toward your new identity! Always remember, “The journey of a thousand miles begins with a single step.”

It tooks efforts to make complex things simple for these amazing carousels please help this reach needy ones!!

Love you all!!

Urvish Jaiswal

Help Others