

[nixCraft](#) → [Howto](#) → [Iptables](#) → How To Configure Firewall with UFW on Ubuntu 20.04 LTS

How To Configure Firewall with UFW on Ubuntu 20.04 LTS

Author: Vivek Gite

Last updated: March 2, 2021

[7 comments](#)

How do I set up and configure firewall with UFW on Ubuntu 20.04 LTS server?



UFW is an acronym for an uncomplicated firewall. Securing a network with an uncomplicated firewall is super easy and highly recommended.

This page explains how to set up and secure your Ubuntu 20.04 LTS server with ufw.

Tutorial requirements	
Requirements	Ubuntu Linux 20.04 LTS
Root privileges	Yes
Difficulty	Easy
Est. reading time	15m

Tutorial requirements

Table of contents

- [1 Set up ufw policy](#)
- [2 Open SSH port](#)
- [3 Turn on ufw firewall](#)
- [4 Open ports with ufw](#)
- [5 Block ports with ufw](#)
- [6 Get ufw firewall status](#)
- [7 Delete ufw firewall rules](#)
- [8 Firewall management commands](#)
- [9 IP Masquerading](#)
- [10 Egress filtering](#)
- [11 Conclusion](#)

ADVERTISEMENT

Step 1 – Set Up default UFW policies

To view status of ufw, type:

```
sudo ufw status
```

Sample outputs:

```
Status: inactive
```

The default policy firewall works out great for both the servers and desktop. It is always a good policy to closes all ports on the server and open only required ports

one by one. Let us block all incoming connection and only allow outgoing connections from the Ubuntu 20.04 LTS box:

```
sudo ufw default allow outgoing
sudo ufw default deny incoming
```

Enabling IPv6 support

Make sure the directive `IPV6=yes` do exists in `/etc/default/uw` file. For instance:

```
cat /etc/default/uw
```

Step 2 – Open SSH TCP port 22 connections

The next logical step is to allow incoming SSH ports. We can easily open SSH TCP port 22 using UFW as follows:

```
sudo ufw allow ssh
```

If you are running ssh on TCP port 2222 or TCP port 2323, enter:

```
sudo ufw allow 2222/tcp
sudo ufw allow 2323/tcp
```

Some sysadmins have a static IP address (such as 202.54.2.5) at home or office location. In that case, only allow ssh access from the static IP address such as 202.54.2.5 to Ubuntu server IP address 172.24.13.45:

```
sudo ufw allow proto tcp from 202.54.2.5 to 172.24.13.45 port 22
```

Let us limit ssh port, run:

```
sudo ufw limit ssh
```

See “[How to limit SSH \(TCP port 22\) connections with ufw on Ubuntu Linux](#)” for more information.

Step 3 – Turn on firewall

Now we got basic configuration enabled. In other words, the firewall will drop all incoming traffic except for ssh TCP port 22. Let us true it on the UFW, enter:

```
sudo ufw enable
```



```
root@ln-sg-vpn-001:~# ufw status
Status: inactive
root@ln-sg-vpn-001:~# sudo ufw default allow outgoing
Default outgoing policy changed to 'allow'
(be sure to update your rules accordingly)
root@ln-sg-vpn-001:~# sudo ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
root@ln-sg-vpn-001:~# sudo ufw allow ssh
Rules updated
Rules updated (v6)
root@ln-sg-vpn-001:~# sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
root@ln-sg-vpn-001:~#
root@ln-sg-vpn-001:~# ufw status
Status: active

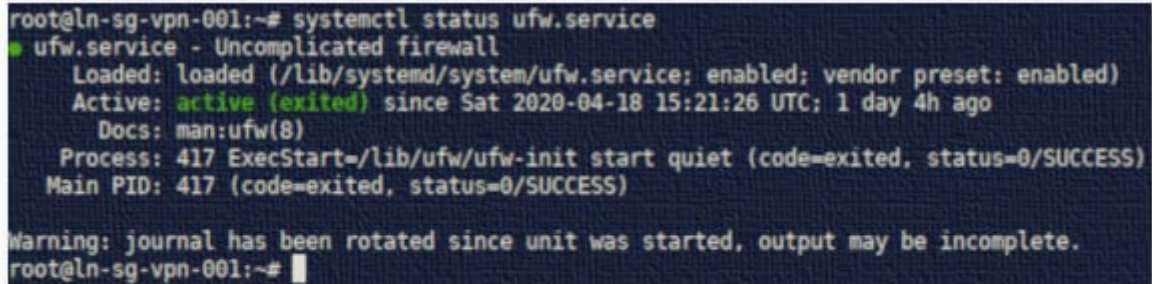
To Action From
--
22/tcp ALLOW Anywhere
22/tcp (v6) ALLOW Anywhere (v6)

root@ln-sg-vpn-001:~#
```

© www.cyberciti.biz

Remember, once UFW enabled, it runs across system reboots too. We can verify that easily as follows using the systemctl command:

```
sudo systemctl status ufw.service
```



```
root@ln-sg-vpn-001:~# systemctl status ufw.service
● ufw.service - Uncomplicated firewall
   Loaded: loaded (/lib/systemd/system/ufw.service; enabled; vendor preset: enabled)
   Active: active (exited) since Sat 2020-04-18 15:21:26 UTC; 1 day 4h ago
     Docs: man:ufw(8)
   Process: 417 ExecStart=/lib/ufw/ufw-init start quiet (code=exited, status=0/SUCCESS)
    Main PID: 417 (code=exited, status=0/SUCCESS)

Warning: journal has been rotated since unit was started, output may be incomplete.
root@ln-sg-vpn-001:~#
```

© www.cyberciti.biz

Want to disable the UFW based firewall? Try

If you need to stop the firewall and disable on system startup, enter:

```
sudo ufw disable
```

Sample outputs:

```
Firewall stopped and disabled on system startup
```

Step 4 – Open specific incoming connections/ports

Let us add more rules. Say you want to open ports and allow IP address with ufw.

The syntax is as follows to open TCP port 80 and 443:

```
sudo ufw allow 80/tcp comment 'accept Apache'
sudo ufw allow 443/tcp comment 'accept HTTPS connections'
```

Open UDP/1194 (OpenVPN) server:

```
sudo ufw allow 1194/udp comment 'OpenVPN server'
```

Allow port ranges via ufw

We can allow port ranges too say, tcp and udp 3000 to 4000:

```
sudo ufw allow 3000:4000/tcp
sudo ufw allow 3000:4000/udp
```

In this example, you want to allow ALL connections from an IP address called 104.22.10.214, enter:

```
sudo ufw allow from 104.22.10.214
```

Let us allow connections from an IP address called 104.22.11.213 to our port 25, enter:

```
sudo ufw allow from 104.22.11.213 to any port 25 proto tcp
```

We can set dest IP 222.222.222.222 for port 25 too:

```
sudo ufw allow from 104.22.11.213 to 222.222.222.222 port 25 proto tcp
```

Allow connection on specific interface

Open port 22 for wg0 interface only:

```
sudo ufw allow in on wg0 to any port 22
```

Say you want to allow connection for TCP port 3306 on lxdbr0 interface from 10.105.28.22, then add:

```
ufw allow in on lxdbr0 from 10.105.28.22 to any port 3306 proto tcp
```

Let us add sub/net instead of single IP address:

```
ufw allow in on lxdbr0 from 10.105.28.0/24 to any port 3306 proto tcp
```

Step 5 – Block and deny incoming connections/ports

Do you want to close ports and block certain IP addresses? The syntax is as follows to deny access. In other words, simply ignoring access to port 25:

```
sudo ufw deny 25/tcp
```

Make sure we deny all connections from an IP address called 203.5.1.43, enter:

```
sudo ufw deny from 203.5.1.43
```

Deny all connections from an IP/subnet called 103.13.42.13/29, enter:

```
sudo ufw deny from 103.13.42.13/29
```

Want to deny access to 1.1.1.2 (say bad guys IP) on port 22? Try:

```
sudo ufw deny from 1.1.1.2 to any port 22 proto tcp
```

Step 6 – Verify status of UFW

Use the status command as follows:

```
sudo ufw status
```

```
Status: active
```

To	Action	From	
--	-----	----	
22/tcp	ALLOW	Anywhere	
80/tcp	ALLOW	Anywhere	# accept Apache
443/tcp	ALLOW	Anywhere	# accept HTTPS connections
1194/udp	ALLOW	Anywhere	# OpenVPN server
3000:4000/tcp	ALLOW	Anywhere	
3000:4000/udp	ALLOW	Anywhere	
Anywhere	ALLOW	104.22.10.214	
25/tcp	ALLOW	104.22.11.213	
222.222.222.222 25/tcp	ALLOW	104.22.11.213	
Anywhere	DENY	203.5.1.43	
Anywhere	DENY	103.13.42.8/29	

22/tcp	DENY	1.1.1.2	
22/tcp (v6)	ALLOW	Anywhere (v6)	
80/tcp (v6)	ALLOW	Anywhere (v6)	# accept Apache
443/tcp (v6)	ALLOW	Anywhere (v6)	# accept HTTPS connections
1194/udp (v6)	ALLOW	Anywhere (v6)	# OpenVPN server
3000:4000/tcp (v6)	ALLOW	Anywhere (v6)	
3000:4000/udp (v6)	ALLOW	Anywhere (v6)	

Want verbose outputs? Try:

```
sudo ufw status verbose
```

Ubuntu 20.04 LTS UFW delete rules

So far we learned how to add, deny, and list the firewall rules. It is time to delete unwanted rules. The syntax is as follows to list all of the current rules in a numbered list format:

```
sudo ufw status numbered
```

Status: active

	To	Action	From	
	--	-----	----	
[1]	22/tcp	ALLOW IN	Anywhere	
[2]	80/tcp	ALLOW IN	Anywhere	# accept Apache
[3]	443/tcp	ALLOW IN	Anywhere	# accept HTTPS connections
[4]	1194/udp	ALLOW IN	Anywhere	# OpenVPN server
[5]	3000:4000/tcp	ALLOW IN	Anywhere	
[6]	3000:4000/udp	ALLOW IN	Anywhere	

To delete 6th rule type the command:

```
sudo ufw delete 6
sudo ufw status numbered
```

See [how to delete a UFW firewall rule on Ubuntu / Debian Linux](https://www.cyberciti.biz/faq/how-to-delete-a-ufw-firewall-rule-on-ubuntu-debian-linux/) tutorial for further information.

Other command used to configure firewall with UFW

Let us learn a few more important commands.

Reset the ufw

```
sudo ufw reset
```

Reload the ufw

```
sudo ufw reload
```

View the firewall logs

By default all UFW entries are logged into `/var/log/ufw.log` file. Use the `NA` command/[more command](#)/`tail` command and other commands to view the ufw logs:

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

Let us print a list of all IP address trying to log in via SSH port but dropped by the UFW:

```
grep 'DPT=22' /var/log/ufw.log | \
egrep -o 'SRC=([0-9]{1,3}[\.]){3}[0-9]{1,3}' | \
awk -F=' ' '{ print $2 }' | sort -u
```

Show the list of rules

```
sudo ufw show listening
sudo ufw show added
```

Outputs:

Added user rules (see 'ufw status' for running firewall):

```
ufw allow from 10.8.0.0/24 to 10.8.0.1 port 22 proto tcp
```

```
ufw allow from 10.8.0.0/24 to 10.8.0.1 port 3128 proto tcp
```

```
ufw allow from 1t9.xxx.yyy.zzz to 1y2.aaa.bbb.ccc port 22 proto tcp
```

Setting up IP Masquerading with ufw

First edit the `/etc/ufw/sysctl.conf` and make sure you have the following line:

```
net/ipv4/ip_forward=1
# IPv6
#net/ipv6/conf/default/forwarding=1
#net/ipv6/conf/all/forwarding=1
```

Next add top of the `/etc/ufw/before.rules` file, before the `*filter` section for `10.0.0.0/8` and `wg0` interface:

```
*nat
:POSTROUTING ACCEPT [0:0]
-A POSTROUTING -s 10.0.0.0/8 -o wg0 -j MASQUERADE
COMMIT
```

Save and close the file. Finally, add the ufw rule to allow the traffic:

```
sudo ufw rule allow in on eth0 out on wg0 from 10.0.0.0/8
```

See “[How to configure ufw to forward port 80/443 to internal server hosted on LAN](#)” for more info.

Setting up egress filtering

Let us say you want to block RFC1918 addresses going out of `eth0` interfaces on your VM connected to the Internet. Add the `ufw rule` rules to reject the traffic:

```
$ sudo ufw route reject out on eth0 to 10.0.0.0/8 comment 'RFC1918 reject'
$ sudo ufw route reject out on eth0 to 172.16.0.0/12 comment 'RFC1918 reject'
$ sudo ufw route reject out on eth0 to 192.168.0.0/16 comment 'RFC1918 reject'
```

Conclusion

In this quick tutorial, you learned how to secure your Ubuntu Linux 20.04 LTS server or desktop with the help of UFW. For more info, please see the ufw [help page here](#).

This entry is **10** of **10** in the **Uncomplicated Firewall (UFW)** series. Keep reading the rest of the series:

1. [Install UFW firewall on Ubuntu 16.04 LTS server](#)
2. [Open ssh port 22 using ufw on Ubuntu/Debian Linux](#)
3. [Configure ufw to forward port 80/443 to internal server hosted on LAN](#)
4. [Block an IP address with ufw on Ubuntu Linux server](#)
5. [Limit SSH \(TCP port 22\) connections with ufw on Ubuntu Linux](#)
6. [Ubuntu Linux Firewall Open Port Command Using UFW](#)
7. [Open DNS port 53 using ufw on Ubuntu/Debian Linux](#)
8. [Set Up a Firewall with UFW on Ubuntu 18.04](#)
9. [Delete a UFW firewall rule](#)
10. Configure Firewall with UFW on **Ubuntu 20.04 LTS**



Get the latest tutorials on Linux, Open Source & DevOps via [RSS feed](#) or [Weekly email newsletter](#).