linode.com

How to Configure a Firewall with UFW

https://www.linode.com/docs/security/firewalls/configure-firewall-with-ufw

Updated by Linode

UFW, or *uncomplicated firewall*, is a frontend for managing firewall rules in Arch Linux, Debian or Ubuntu. UFW is used through the command line (although it has GUIs available), and aims to make firewall configuration easy (or, uncomplicated).

Before You Begin

1. Familiarize yourself with our Getting Started guide and complete the steps for setting your Linode's hostname and timezone.
2. This guide will use `sudo` wherever possible. Complete the sections of our Securing Your Server guide to create a standard user account, harden SSH access and remove unnecessary network services. Do **not** follow the Creating a Firewall section–this guide is an introduction to using UFW, which is a separate method of controlling a firewall than iptables commands.
3. Update your system.

   Arch Linux

   Debian / Ubuntu

   ```
   1 sudo apt-get update && sudo apt-get upgrade
   ```

Install UFW

UFW is included in Ubuntu by default but must be installed in Arch and Debian. Debian will start UFW's systemd unit automatically and enable it to start on reboots, but Arch will not. *This is not the same as telling UFW to enable the firewall rules*, as enabling UFW with systemd or upstart only tells the init system to switch on the UFW daemon.

By default, UFW's rulesets are blank so it is not enforcing any firewall rules–even when the daemon is running. Enforcing your firewall ruleset is covered further down the page.

Arch Linux

1. Install UFW:
2. Start and enable UFW's systemd unit:

```
1 sudo systemctl start ufw
2 sudo systemctl enable ufw
```

Debian / Ubuntu

1. Install UFW

```
1 sudo apt-get install ufw
```

Use UFW to Manage Firewall Rules

Set Default Rules

Most systems will need a only a small number of ports open for incoming connections, and all remaining ports closed. To start with an easy basis of rules, the `ufw default` command can be used to set the default response to incoming and outgoing connections. To deny all incoming and allow all outgoing connections, run:

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

The `ufw default` command also allows for the use of the `reject` parameter.

> Configuring a default reject or deny rule can lock you out of your Linode unless explicit allow rules are in place. Ensure that you have configured allow rules for SSH and other critical services as per the section below before applying default deny or reject rules.

Add Rules

Rules can be added in two ways: By denoting the **port number** or by using the **service name**.

For example, to allow both incoming and outgoing connections on port 22 for SSH, you can run:

You can also run:

Similarly, to **deny** traffic on a certain port (in this example, 111) you would only have to run:

To farther fine-tune your rules, you can also allow packets based on TCP or UDP. The following will allow TCP packets on port 80:

```
1 sudo ufw allow 80/tcp
2 sudo ufw allow http/tcp
```

Whereas this will allow UDP packets on 1725:

Advanced Rules

Along with allowing or denying based solely on port, UFW also allows you to allow/block by IP addresses, subnets, and a IP address/subnet/port combinations.

To allow connections from an IP address:

```
1 sudo ufw allow from 123.45.67.89
```

To allow connections from a specific subnet:

```
1 sudo ufw allow from 123.45.67.89/24
```

To allow a specific IP address/port combination:

```
1 sudo ufw allow from 123.45.67.89 to any port 22 proto tcp
```

`proto tcp` can be removed or switched to `proto udp` depending upon your needs, and all instances of `allow` can be changed to `deny` as needed.

Remove Rules

To remove a rule, add `delete` before the rule implementation. If you no longer wished to allow HTTP traffic, you could run:

```
1 sudo ufw delete allow 80
```

Deleting also allows the use of service names.

Edit UFW's Configuration Files

Although simple rules can be added through the command line, there may be a time when more advanced or specific rules need to be added or removed. Prior to running the rules input through the terminal, UFW will run a file, `before.rules`, that allows loopback, ping, and DHCP. To add to alter these rules edit the `/etc/ufw/before.rules` file. A `before6.rules` file is also located in the same directory for IPv6.

An `after.rule` and an `after6.rule` file also exists to add any rules that would need to be added after UFW runs your command-line-added rules.

An additional configuration file is located at `/etc/default/ufw`. From here IPv6 can be disabled or enabled, default rules can be set, and UFW can be set to manage built-in firewall chains.

UFW Status

You can check the status of UFW at any time with the command: `sudo ufw status`. This will show a list of all rules, and whether or not UFW is active:

```
1  Status: active
2
3  To                         Action      From
4  --                         ------      ----
5  22                         ALLOW       Anywhere
6  80/tcp                     ALLOW       Anywhere
7  443                        ALLOW       Anywhere
8  22 (v6)                    ALLOW       Anywhere (v6)
9  80/tcp (v6)                ALLOW       Anywhere (v6)
10 443 (v6)                   ALLOW       Anywhere (v6)
```

Enable the Firewall

With your chosen rules in place, your initial run of `ufw status` will probably output `Status: inactive`. To enable UFW and enforce your firewall rules:

Similarly, to disable UFW's rules:

> This still leaves the UFW service running and enabled on reboots.

Logging

You can enable logging with the command:

Log levels can be set by running `sudo ufw logging low|medium|high`, selecting either `low`, `medium`, or `high` from the list. The default setting is `low`.

A normal log entry will resemble the following, and will be located at `/var/logs/ufw`:

```
1 Sep 16 15:08:14 <hostname> kernel: [UFW BLOCK] IN=eth0 OUT= MAC=00:00:00:00:00:00:00:00:00
```

The initial values list the date, time, and hostname of your Linode. Additional important values include:

- **[UFW BLOCK]:** This location is where the description of the logged event will be located. In this instance, it blocked a connection.
- **IN:** If this contains a value, then the event was incoming
- **OUT:** If this contain a value, then the event was outgoing

- **MAC:** A combination of the destination and source MAC addresses
- **SRC:** The IP of the packet source
- **DST:** The IP of the packet destination
- **LEN:** Packet length
- **TTL:** The packet TTL, or *time to live*. How long it will bounce between routers until it expires, if no destination is found.
- **PROTO:** The packet's protocal
- **SPT:** The source port of the package
- **DPT:** The destination port of the package
- **WINDOW:** The size of the packet the sender can receive
- **SYN URGP:** Indicated if a three-way handshake is required. `0` means it is not.