Getting CIDR lists

Getting CIDR lists

You will want to get hold of lists of IP addresses if you want to use the *blacknets* feature of *nftfw* (v0.7.0 or later) to block access from certain countries or IP address ranges. If you want to use this system, check that the version of *nftfw* you have installed is configured to support it, see How to check *nftfw* supports_blacknets at bottom of this page.

To use *blacknets*, you need a file or files containing IP networks, one per line, using ‘CIDR’ notation. See Wikipedia if you need more information on CIDR.

There are several sources for lists of networks by country, I’ve used two.

ip2location.com

This company, based in the Isle of Man, has been collating IP addresses for many years. They offer a free download of IP addresses per country in several formats aimed at different applications. Visit their page Block Visitors by Country Using Firewall and scroll to the bottom of the page for the form.

There are three drop-down menus: choose the country, select IPv6, and CIDR and click DOWNLOAD. The file will be downloaded to your machine.

Place the file in *blacknets.d*, remembering to add *.nets* as the file suffix.

The IPv6 file contains all the IPv4 addresses, and *nftfw’s* reader will convert the addresses into the correct format.

The downside of this is that you have to download the file by hand, but it’s easy to use as a starter.

Maxmind Geolocation

If you’ve installed the GeoLite2 database from Maxmind to assist with identifing countries with *nftfwls*, then with some little work you can access their country database and also have your system refresh it once a week.

Step 1 - Getting the Maxmind database

The script you are about to install looks for your MaxMind license information in */etc/GeoIP.conf*, so you need to install the Geolocation system first, see Installing Geolocation.

You’ll need `wget`, so

```
$ sudo apt install wget
```

Navigate to the *nftfw* release and find the *cidrlist* directory. Install the shell script`getgeocountry` that pulls the database from MaxMind:

```
$ sudo cp getgeocountry /var/lib/GeoIP
$ cd /var/lib/GeoIP
$ sudo chown root.root getgeocountry
$ sudo chmod +x getgeocountry
```

run it

```
$ sudo getgeocountry
```

The script will have created several files, culminating in *GeoLite2-Country.db* containing an *sqlite3* database made from the downloaded information.

Make this script run once a week by placing a line in the *cron* file provided as part of the *geoipupdate* package. Edit*/etc/cron.d/geoipupdate* adding:

```
30 7    * * 3   root    /var/lib/GeoIP/getgeocountry
```

I run mine an hour after the weekly run of *geoipupdate*, so please do choose the hour and minute to be different from the world.

Step 2 - Creating the country CIDR file

Return to the *nftfw* distribution and find the *cidrlist* directory again. The `getcountrynet` script uses the database installed in Step 1 to create a country CIDR file. Install the script:

```
$ sudo cp getcountrynet /etc/nftfw
$ cd /etc/nftfw
$ sudo chown USER.USER getcountrynet
$ sudo chmod +x getcountrynet
```

The USER.USER here should be whoever owns the */etc/nftfw* directory (may be */usr/local/etc/nftfw* on your machine). I've installed my command file in this directory on the grounds of ‘keeping everything together’, but you can put it anywhere that's convenient.

Now run the script as the owner of the directory or use sudo and change ownership afterwards. To run, you can give it any number of two letter ISO country codes and it will create a matching file in *blacknets.d*.

```
$ getcountrynet GB fr
```

will create files called *GB.nets* and *fr.nets* in *blacknets.d*. Remember that file systems are case-dependent, so choose your capitalisation and stick with it. I'm assuming you replace the arguments with a country or countries that you want to block.

If you are confident that you've got an appropriate version of *nftfw* (see below), you can now install the new tables, prudently running a test first:

```
$ sudo nftfw -x load
```

All being well, you can then install the new tables:

```
$ sudo nftfw -f load
```

Finally tell *cron* to run the `getcountrynet` script. Again, I've added a new line to the */etc/cron.d/geoipupdate* adding:

```
55 7    * * 3   USER /etc/nftfw/getcountrynet COUNTRIES
```

the USER should be whoever owns the files, and the arguments should match the ones you typed in earlier.

How to check *nftfw* supports blacknets

The *blacknets* feature from v0.7.0 of *nftfw* requires a change in the firewall template file */etc/nftfw/nftfw_init.nft*. You may need to check you've updated the *nftfw_init.nft* file before running *nftfw* with the CIDR files. This file requires updating by hand, and you may have not installed it.

It the file doesn't contain the string *blacknets*, then you need to update it.

```
$ cd /etc/nftfw
$ grep blacknets nftfw_init.nft
```

If the command gives no output, check the copy of the file in the *etc_nftfw* directory using *grep*.

```
$ grep blacknets /etc/nftfw/etc_nftfw/nftfw_init.nft
```

If this gives output, then copy the *originals* file over your running version, carefully re-applying any changes you've made.

If there is no output from *grep* on the copy in *etc_nftfw*, you need to update your *nftfw* installation to a version after v0.7.0. See Updating *nftfw*.

General notes

You can create as many files as you like in *blacknets.d* as long as they are in the correct format. To remove a set from the firewall, simply remove the file.

The files can contain a lot of IP addresses, and processing them can take some time. The reader will cope with automatic detection of IPv4 and IPv6, the conversion of IPv4 addresses when expressed in IPv6 format (which *ip2location.com* uses), and the removal of some addresses that look like networks but are not. It will also remove duplicates and compresses the IP list down to a set of unique networks. It's possible to run CIDR files from both of the sources shown in this document, and reduce them to a minimal set.

Generally, the files in *blacknets.d* change rarely, so *nftfw* will cache processed information on the first reading of the files and will read from the cache when it needs the data to build the firewall. The cache is always reloaded when files in *blacknets.d* alter or come and go. In addition, the `-f` flag to *nftfw* will clear the cache and start again.