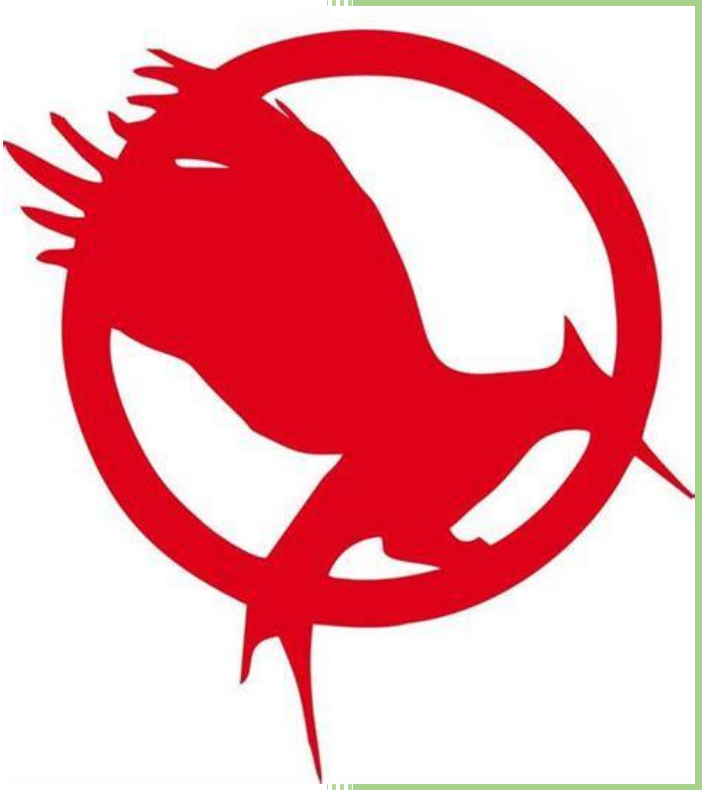
2021

Catching Firewall redirected DNS requests



https://discourse.pi-hole.net/u/jpgpi250

4-11-2021

1. About this manual.

Using this guide will allow you to identify the devices, trying to bypass your pi-hole.

If you are reading this document, using Adobe Reader, you may click on a hyperlink to content in this document. Use the combination <Alt> <left arrow> to return to the previous location.

"Back" and "Forward" buttons can also be added to the toolbar. If you right-click on the tool bar, under "Page Navigation", they are referred to as "Previous View" and "Next View".

This document is hosted on GitHub, you can open the document (pdf), using this link.

Copying and pasting from this manual into Putty doesn't seem to work all the time. If you get an error, try typing the command…

2. Goal.

Most of us have a firewall rule that redirect all DNS queries that target another DNS server than pi-hole (usually 8.8.8.8). It's not always easy to identify the rogue devices and queries. By adding a dnsmasq instance to pi-hole, we can log (and visualize) all rogue queries. There are some down sides to this method.

- The pi-hole web interface will show all these queries coming from the secondary IP address, we will add to achieve this, you'll need to use a different method to identify the rogue devices and queries. Pihole-FTL cannot identify redirected DNS queries, the binary doesn't have that information (asked and answered).

3. Adding the secondary IP address.

Why would we add a secondary IP address?

There are some articles, claiming that adding a second DNS server specification for the client (served by DHCP or static), eliminates the rogue behavior (using 8.8.8.8). If you do this (pihole-

FTL uses all IP addresses on the system – see /etc/dnsmasq.d/01-pihole.conf - interface=eth0), ensure your firewall rule excludes the secondary IP address from being redirected.

On raspbian, it's easy to add a secondary IP address (activated every time the system is rebooted). Create a file /etc/dhcpcd.exit-hook, content:

```
#add secondary IP address
if [ "$reason" = "PREINIT" ]; then
    sudo ip -4 addr add 192.168.2.48/27 dev eth0 label eth0:1
    exit 0
fi
```

Ensure the IP address is unused, in the same subnet as the primary address, uses the correct device and label.

Reboot the system to activate the secondary IP address.

4. **Installing dnsmasq.**

The hart of pi-hole is pihole-FTL, a binary that is based on dnsmasq, with a lot of additional features, such as, but not limited to, regular expressions, databases (query and gravity), deep CNAME inspection, …

Dnsmasq may have been installed on your system, prior to installing pi-hole. The pi-hole installer disables dnsmasq, because pihole-FTL uses the same port (DNS – 53) as dnsmasq.

Dnsmasq and pihole-FTL cannot coexist, while using the same port. In order to find out if dnsmasq is already available, run:

```
which dnsmasq
```

If there is no output (dnsmasq not found on your system) you need to install it:

```
sudo apt-get install –y dnsmasq
```

dnsmasq will not start, due to a port conflict, no worries, we will change that, using the following configuration, which will prepare the system for catching rogue DNS requests, and send them to pi-hole

WARNING: If you installed dnsmasq after pi-hole was already running, you need to ensure dnsmasq doesn't auto start, this to avoid a conflict with pihole-FTL. To achieve this, run:

```
sudo systemcll disable dnsmasq
```

Create a file, /etc/dnsmasq.d/redirect/redirect.conf (you need to create the **subfolder redirect**, to avoid pihole-FTL also using this configuration file), content:

```
cache-size=0
log-queries=extra
log-facility=/var/log/redirect.log
no-resolv
listen-address=192.168.2.48
bind-interfaces
except-interface=lo
port=5558
server=192.168.2.57#53
```

- Change the listen-address into the secondary IP address you configured before.
- Change the port into the port you'll be using on the firewall as redirect port (you are probably using port 53 as redirect port on your firewall, you will need to change the port into an unused port and change the destination IP address into the secondary IP address.
- Change the server directive into the primary IP address and port of the system (pi-hole DNS address).

Once completed, start dnsmasq (don't use the method you're used to start services, we need to force dnsmasq to use a specific config):

```
sudo dnsmasq --conf-file=/etc/dnsmasq.d/redirect/redirect.conf
```

In order to start dnsmasq after a reboot, using the above configuration, create a file /etc/cron.d/dnsmasq, content:

```
@reboot root PATH="$PATH:/home/pi/" /home/pi/dnsmasq.sh >/dev/null 2>&1
```

Also create a script /home/pi/dnsmasq.sh (change the IP to your secondary IP), content:

```
#!/bin/bash
until (/bin/ping -n -w 2 192.168.2.48 | /bin/grep "0% packet loss"); do
  /bin/sleep 5
done
sudo /usr/sbin/dnsmasq --conf-file=/etc/dnsmasq.d/redirect/redirect.conf
```

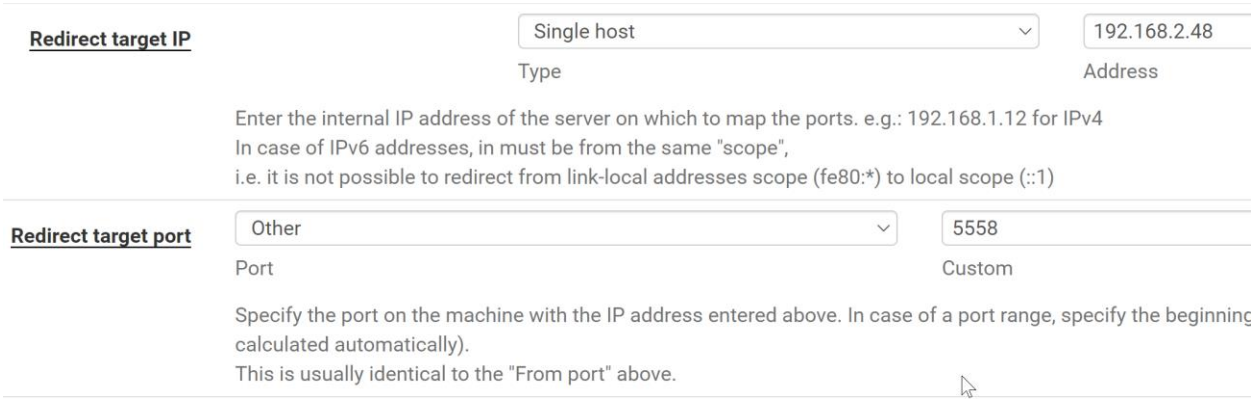
5. **Changing the firewall configuration.**

   This document does not describe how to set up redirection rules on your firewall, to many brands and models…

   Assuming you already have a working rule, you'll need to change this:

   - Replace the NAT IP (redirect target IP).
   - Replace the NAT ports (redirect target port).

   Pfsense screenshot:



   After you have completed this configuration change, you can test the setup. From any workstation, run:

```
dig @8.8.8.8 example.com
```

   You will get a reply (IP 93.184.216.34 in my region).

   The log will show (IP address from the workstation is in the log):

Nov 3 12:48:23 dnsmasq[523]: 756 192.168.2.228/63058 query[A] example.com from 192.168.2.228

Nov 3 12:48:23 dnsmasq[523]: 756 192.168.2.228/63058 forwarded example.com to 192.168.2.57

Nov 3 12:48:24 dnsmasq[523]: 756 192.168.2.228/63058 reply example.com is 93.184.216.34

6. **Visualization.**

   There are many ways to visualize this information. Personally, I've been using grafana to visualize the information, both from pi-hole (not using the web interface) and other sources, such as the redirect log.

   It's not the scope of this document to provide full setup instructions for grafana, I'll only provide some pointers to customize the redirect dashboard.

   - Installing grafana : https://grafana.com/grafana/download?platform=arm
   - Installing sqlite3 plugin: https://grafana.com/grafana/plugins/frser-sqlite-datasource/
   - Installing loki and promtail: https://grafana.com/docs/loki/latest/installation/local/

For loki and promtail, I needed to change some settings in the suggested configuration files, see here.

I reconfigured promtail (/home/pi/grafana/promtail-local-config.yaml) to read only the redirect log:

```
scrape_configs:
- job_name: system
  static_configs:
  - targets:
      - localhost
    labels:
      job: redirectlogs
      __path__: /var/log/pihole/redirect.log
```

You need to start loki and promtail to get results.

If you only want to use the grafana dashboard from time to time (identify rogue DNS requests for a new IOT device), you can run (requires screen – sudo apt-get -y install screen):

```
screen -S loki -dm bash -c "cd /home/pi/grafana; sudo ./loki-linux-arm -config.file=loki-local-config.yaml"
screen -S promtail -dm bash -c "cd /home/pi/grafana; sudo ./promtail-linux-arm --config.file=promtail-local-config.yaml"
```

If you want promtail and loki to run at startup, create a system file for both services, see here.

1. Managing the redirect log file.

   In order to keep the redirect log manageable, I've added some lines to the default log rotation configuration of pi-hole /etc/pihole/logrotate, added content:

```
/var/log/redirect.log {
    su root root
    daily
    copytruncate
    rotate 1
    compress
    notifempty
    nomail
}
```

This will ensure the log remains manageable, containing only today's redirect info. The log is rotated daily (see /etc/cron.d/pihole - Pi-hole: Flush the log daily at 00:00), only one compressed log is kept (previous day).

To compile some usable data for grafana, the following script is run (sometime after midnight, once a day):

```
#!/bin/bash

sourcedir=/home/pi/redirect
redirectdb=${sourcedir}/sqlite3/redirect.db

# create database if it doesn't exist
if [ ! -f ${redirectdb} ]; then
	sudo sqlite3 ${redirectdb} < ${sourcedir}/sqlite3/redirect.sql
fi

timestamp=$(date +%s)
sudo sqlite3 "${redirectdb}" ".timeout = 50000" \
	"update 'info' \
	set value = \"${timestamp}\" \
	where property = 'latest_timestamp';"

# Oct 29 23:21:20 dnsmasq[431]: 1 192.168.2.228/64542 query[A] example.com from 192.168.2.228
while IFS=" " read -r Month Day Time dummy1 Serial dummy2 QueryType Domain dummy2 Client; do
	timestamp=$(date -d "${Month} ${Day} ${Time}" +%s)
	type=$(echo ${QueryType} | sed 's/.*\[\([^]]*\)\].*/\1/g')
	sudo sqlite3 "${redirectdb}" ".timeout = 50000" \
			"insert or ignore into 'queries' \
			( timestamp, serial, type, domain, client )
			values (\"${timestamp}\", \"${Serial}\", \"${type}\", \"${Domain}\", \"${Client}\" );"
done < <(sudo find /var/log/ -name "redirect.*" -exec zgrep 'from' {} \;)

# delete entries older than 8 days
let timestamp=${timestamp}-691200
sudo sqlite3 "${redirectdb}" ".timeout = 50000" \
	"delete from queries
	where timestamp < \"${timestamp}\";"
sudo sqlite3 "${redirectdb}" "VACUUM;"
```

You need to create the directories /home/pi/redirect and /home/pi/redirect/sqlite3 before running the script, and copy the sqlite3 script to create the database (/home/pi/redirect/sqlite3/redirect.sql):

```
PRAGMA foreign_keys=OFF;
BEGIN TRANSACTION;

CREATE TABLE info
(
    property TEXT PRIMARY KEY,
    value TEXT NOT NULL
);

INSERT INTO "info" VALUES('version','1');
INSERT INTO 'info' VALUES('latest_timestamp','0');

CREATE TABLE queries
(
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    timestamp INTEGER NOT NULL,
    serial INTEGER NOT NULL,
    type TEXT NOT NULL,
    domain TEXT NOT NULL,
    client TEXT NOT NULL,
    UNIQUE (timestamp, serial)
);
COMMIT;
```

2. Grafana configuration pointers

   Create the following data sources

   - Sqlite:
     - Name: redirect
     - path: /home/pi/redirect/sqlite3/redirect.db

   You may run into a problem, defining the data source, see here for a discussion, possible workaround.

   - Loki:
     - Name: Loki
     - URL: http://192.168.2.57:3100 (Change the IP to the primary IP of pi-hole)

   Create the following panels (type table) in a dashboard.
   - Sqlite:
     - Visualization: table
     - Data source: redirect

- Query: SELECT strftime('%d/%m/%Y %H:%M:%S', datetime(timestamp, 'unixepoch', 'localtime')), count(domain), type, client, domain FROM queries GROUP BY type, domain, client ORDER BY MAX(timestamp) DESC LIMIT 100;

Result:

redirect (sqlite database - not including today)

time (most recent)	count	type	client	domain
03/11/2021	840	A	192.168.2.240	api-global.netflix.com
03/11/2021 21:50:16	262	A	192.168.2.240	occ-0-1335-1336.1.nflxso.net
03/11/2021 21:50:13	401	A	192.168.2.240	ichnaea.netflix.com
03/11/2021 21:50:13	1140	A	192.168.2.240	nrdp.prod.ftl.netflix.com
03/11/2021 21:50:13	89	A	192.168.2.240	preapp.prod.partner.netflix.net
03/11/2021 21:50:13	66	A	192.168.2.240	push.prod.netflix.com

- Loki:
  - Log browser: **{filename="/var/log/pihole/redirect.log"} |= "from"**
  - Transform, organize fields, hide everything except line

Result:

redirect (loki log view - today only)

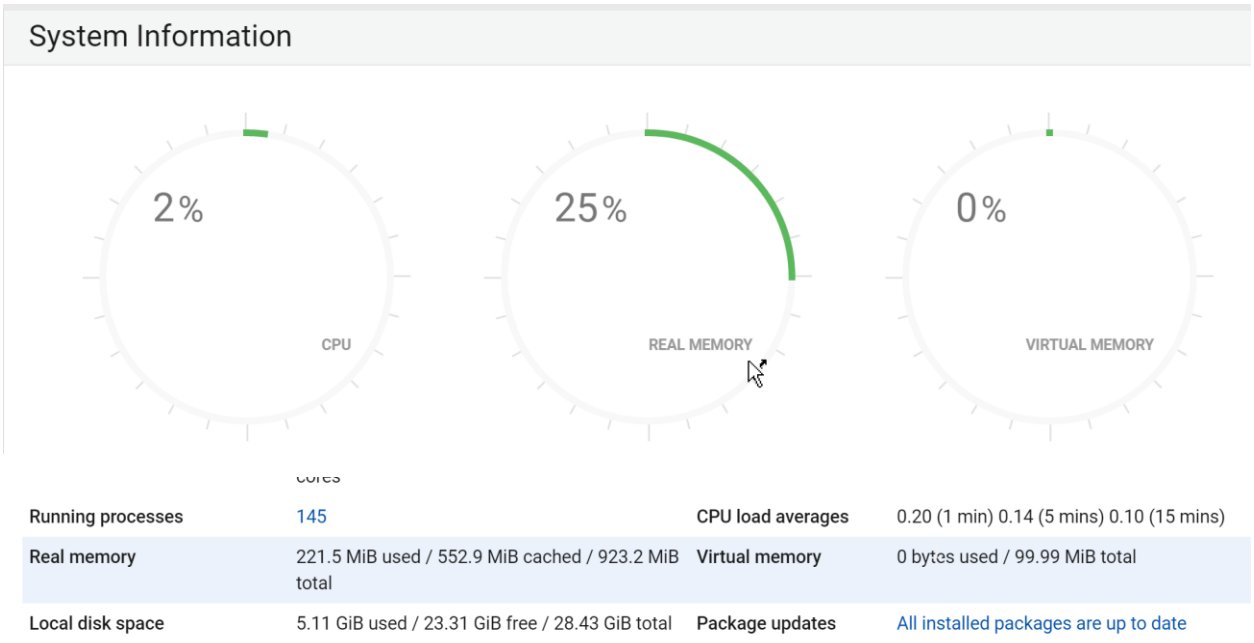
/var/log/pihole/redirect.log
Nov 4 11:13:11 dnsmasq[515]: 22 192.168.2.228/56488 query[A] example.com from 192.168.2.228
Nov 4 11:12:45 dnsmasq[515]: 21 192.168.2.184/40958 query[A] ssl.google-analytics.com from 192.168.2.184
Nov 4 10:05:09 dnsmasq[515]: 20 192.168.2.164/35819 query[A] www.google.com from 192.168.2.164
Nov 4 10:05:09 dnsmasq[515]: 19 192.168.2.164/40974 query[A] www.google.com from 192.168.2.164
Nov 4 09:55:15 dnsmasq[515]: 18 192.168.2.164/33974 query[A] www.google.com from 192.168.2.164
Nov 4 09:55:15 dnsmasq[515]: 17 192.168.2.164/53185 query[A] www.google.com from 192.168.2.164

7. **Hardware requirements.**
   Running pi-hole, webmin, grafana, loki, promtail on a raspberry pi 3B, OS raspbian lite latest version (Linux raspberrypi 5.10.63-v7+ #1459 SMP Wed Oct 6 16:41:10 BST 2021 armv7l GNU/Linux), 32Gb SD card.

You might need to increase the swapfile size (usage increases over time):

```
sudo sed -i '/CONF_SWAPSIZE/s/100/256/g' /etc/dphys-swapfile
```

Webmin System Information:



8. Change Log.

04-11-2021

- Version 1 (draft). Report issues here.