Fritz Speed Documentation

Release 0.1.1

Thomas Westfeld

March 01, 2016

What it does

A small suite of Python scripts to graph the incoming and outgoing transfer speeds from a Fritz!Box. It can be run on a small embedded device such as a RaspberryPi. It uses the RRDtool bindings for python to output the graphs suitable for embedding into a website.

The scripts can be run without superuser privileges as a normal user, he only has to have the right file permissions to read and write to the round robin archive and to write the images.

How It Works

The Fritz!Box does not provide a standard SNMP interface common to most routers. Instead it relies on the TR-064 standard.

The script `monitor-traffic.py` uses this interface to retrieve the counters for uploaded and downloaded bytes from the router and stores them in a round robin archive based on RRDtool. A second script `graph-traffic.py` reads this archive and generates graphs of the traffic for configured intervals of time.

Contents

3.1 Installation

3.1.1 Prerequisites

Fritz Speed is working on a standard Python 2.7 environment and relies on the following packages for querying the Fritz!Box.

- fritzconnection
- lxml
- requests

For storing the data in a round robin database it uses the python bindings to RRDtool.

- python-rrdtool

To get the scripts you need Git to clone the repository using the URL `https://github.com/westfeld/fritz-speed.git`

3.1.2 Installation on a RaspberryPi Running Raspian

3.1.3 1. Downloading the Scripts

To install the Fritz Speed scripts clone the GitHub repository:

```
git clone https://github.com/westfeld/fritz-speed.git
```

3.1.4 2. Installing the Dependencies

There are many ways to install the dependencies but for the common packages we rely on aptitude:

```
aptitude install python-rrdtool python-requests python-lxml python-pip
```

The fritzconnection package is not in the raspbian repository so it will be installed using pip by running:

```
pip install fritzconnection
```

3.1.5 3. Configuration

Configuration basically means setting the location of the round robin archive (rra) file in which the data is stored. In addition the properties and location of the individual graphs have to be defined.

The first step is to copy the example configuration file `fritz-speed.ini.example` to `fritz-speed.ini` to enable further updates easily.

Starting from the default `fritz-speed.ini` file typically only two variables have to be edited: `rra_filename` which defines the round robin archive filename in which the traffic data is stored and `graph_basedir` which is the output directory under which all graph files will be stored.

Note: The user under which both the `monitor-traffic.py` and `graph-traffic.py` scripts are executed has to have read/write permissions to the `rra_filename` and the `graph_basedir` directory.

An example of a typical `fritz-speed.ini` file is shown here:

```
[DEFAULT]
rra_filename: /var/rrdtool/wan_traffic.rra
graph_height: 300
graph_width: 500
graph_basedir: /var/www/htdocs/pics
# graph type for up/downstream: LINEx (where x = line width), AREA
# more information about plotting
# https://oss.oetiker.ch/rrdtool/doc/rrdgraph_graph.en.html
graph_type_up: LINE1
graph_type_down: LINE1
# graph color for up/downstream in RGB from 00 to FF in 000000 format
graph_color_up: ff0000
graph_color_down: 0000ff

# definitions of graphs to be plotted
# each section corresponds to one graph
[day]
# interval in seconds to be plotted
interval: 86400
# title of plot
title: Traffic of last day
# filename of plot
filename: %(graph_basedir)s/traffic_1d.png

[week]
# interval in seconds to be plotted
interval: 604800
# title of plot
title: Traffic of last week
# filename of plot
filename: %(graph_basedir)s/traffic_1w.png

[month]
# interval in seconds to be plotted
interval: 2592000
# title of plot
title: Traffic of last month
# filename of plot
filename: %(graph_basedir)s/traffic_1m.png
```

3.1.6 4. Create Round Robin Archive

To create the empty archive simply run the following script:

```
./create-rra.py
```

During this setup step the maximum up- and downlink speed is queried from the router and set as the maximum valid value in the archive (an error margin of 10% is added to the determined value). This setting is important because if the router's transferred bytes counter is reset this would otherwise lead to a spike in the traffic graph.

3.1.7 5. Running as a Cronjob

To monitor the traffic on the Fritz!Box the `monitor-traffic.py` script is executed periodically, which then reads the traffic counter on the router and stores the value in the round robin archive.

To generate the graphs from the stored data the script `graph-traffic.py` is executed. It would also update already existing images.

Typically one would add the following line to the user's crontab to record the data and update the graphs every minute. To do so the command `crontab -e` opens the user's crontab and the following line is added

```
*/1 * * * * [absolute path to script]/monitor-traffic.py && [absolute path to script]/graph-traffic.p
```

3.2 Changelog

3.2.1 Version 0.1.2

- add the possibility to change the graphing style (line or area and its color) directly from the configuration file.
- renamed the default configuration file so that changes to this file do not interfere with updates coming from the repo.

  Pull request contributed by Olaf Peters (http://magicolf.de)

3.2.2 Version 0.1.1

- When creating a new rra for traffic data, the maximum line speed is queried from the router and used as a valid upper limit. This avoids spikes in the traffic graph originating from traffic counter resets on reconnects.

3.2.3 Version 0.1

Inital version

3.3 License

The Fritz Speed scripts are released under MIT license.

3.3.1 The MIT License (MIT)

Copyright (c) 2016 Thomas Westfeld

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Disclaimer

The product Fritz! and Fritz!Box is a trademark of the AVM GmbH, Berlin, Germany.