

Automatisierte Speedtests zur Durchklage beim ISP

Denkt dran. Es gibt vertragliche Minimums und Maximums. Das Minimum muss konstant, drei Monate lang, zuverlässig und zu verschiedenen Zielen unterschritten worden sein. Das ganze muss direkt an eurem Router hängen (oder es muss zusätzlich nachgewiesen werden dass jede Zwischenkomponente zu jeder Zeit die Minimumbandbreite gekonnt hat).

Mit Grafana

<https://pimylifeup.com/raspberry-pi-internet-speed-monitor>

Führt bitte nichts aus ohne zu verstehen was es ist. Ich fasse hier ganze HowTo's als ein einzelnes Script zusammen. Wer das in Menschensprache verstehen will muss schon auf den Link klicken.

Diese Scripte sind nicht unbedingt als solche ausführbar! Sie zeigen halt nur ohne Plemplem um Blabla das was an Maschineninput gemacht werden muss.

```
sudo apt-get update && sudo apt-get upgrade -y
sudo apt install apt-transport-https gnupg1 dirmngr lsb-release
curl -L https://packagecloud.io/ookla/speedtest-cli/gpgkey | gpg --dearmor | sudo tee
/usr/share/keyrings/speedtestcli-archive-keyring.gpg >/dev/null
echo "deb [signed-by=/usr/share/keyrings/speedtestcli-archive-keyring.gpg]
https://packagecloud.io/ookla/speedtest-cli/debian/ $(lsb_release -cs) main" | sudo tee
/etc/apt/sources.list.d/speedtest.list
sudo apt update
sudo apt install speedtest
curl https://repos.influxdata.com/influxdata-archive.key | gpg --dearmor | sudo tee
/usr/share/keyrings/influxdb-archive-keyring.gpg >/dev/null
echo "deb [signed-by=/usr/share/keyrings/influxdb-archive-keyring.gpg]
https://repos.influxdata.com/debian $(lsb_release -cs) stable" | sudo tee
/etc/apt/sources.list.d/influxdb.list
sudo apt update
sudo apt install influxdb
```

```

sudo systemctl unmask influxdb
sudo systemctl enable influxdb
sudo systemctl start influxdb
# careful, I don't know yet if setting an admin before activating http auth is even possible.
Create the admin after influxdb.conf edit if it isn't possible
influx
CREATE USER "admin" WITH PASSWORD '<password>' WITH ALL PRIVILEGES
CREATE DATABASE <wie auch immer eure Speedtest DB heißen soll>
USE <wie auch immer eure Speedtest DB heißen soll>
CREATE USER "Speedtest-user" WITH PASSWORD '<password>'
GRANT ALL ON "<wie auch immer eure Speedtest DB heißen soll>" to "Speedtest-user"
quit
sudo sed -i '/^\[HTTP\]/a auth-enabled = true\npprof-enabled = true\npprof-auth-enabled =
true\nping-auth-enabled = true' /etc/influxdb/influxdb.conf
sudo systemctl restart influxdb
sudo apt install python3-influxdb
curl https://apt.grafana.com/gpg.key | gpg --dearmor | sudo tee /usr/share/keyrings/grafana-
archive-keyrings.gpg >/dev/null
echo "deb [signed-by=/usr/share/keyrings/grafana-archive-keyrings.gpg] https://apt.grafana.com
stable main" | sudo tee /etc/apt/sources.list.d/grafana.list
sudo apt update
sudo apt install grafana
sudo systemctl enable grafana-server
sudo systemctl start grafana-server
echo "Grafana is http://$(hostname -I):3000"
echo "influx is http://localhost:8086"
(crontab -l ; echo "*/30 * * * * python3 $(getent passwd \$(whoami) | cut -d: -
f6)/speedtest.py") | crontab -
# replace the variables for the db user before pasting!
cat <<EOF > ~/speedtest.py
import re
import subprocess
from influxdb import InfluxDBClient
host_name = socket.gethostname()
response = subprocess.Popen('/usr/bin/speedtest --accept-license --accept-gdpr', shell=True,
stdout=subprocess.PIPE).stdout.read().decode('utf-8')
ping = re.search('Latency:\s+(.*?)\s', response, re.MULTILINE)
download = re.search('Download:\s+(.*?)\s', response, re.MULTILINE)
upload = re.search('Upload:\s+(.*?)\s', response, re.MULTILINE)
jitter = re.search('Latency:.*?jitter:\s+(.*?)ms', response, re.MULTILINE)

```

```

ping = ping.group(1)
download = download.group(1)
upload = upload.group(1)
jitter = jitter.group(1)
speed_data = [
    {
        "measurement" : "internet_speed",
        "tags" : {
            "host": host_name
        },
        "fields" : {
            "download": float(download),
            "upload": float(upload),
            "ping": float(ping),
            "jitter": float(jitter)
        }
    }
]
client = InfluxDBClient('localhost', 8086, 'Speedtest-user', '<password>', '<wie auch immer eure Speedtest DB heißen soll>')
client.write_points(speed_data)
EOF

```

Ich habe hier nicht `/home/pi/speedtest.py` genommen, weil man mit dem RPI imager entsprechend schon User vordefinieren kann. Ich wollte hier eine Lösung haben, die auch für diejenigen gültig ist die das getan haben. Dann gibts nämlich keinen pi user.

Derjenige der diesen Crontab anlegt wird der Script-Ausführer sein.

Spätestens jetzt läuft Grafana und Speedtests werden alle 30 Minuten ins Grafana geschrieben. Webinterface müsste Port 3000 sein. Was ihr hier in Zeile 15 gemacht habt müsstet dort die Logindaten sein.

Unschöner mit Gnuplot

Dreckig interpretiert von <https://supportblog.ch/uberwache-deine-internetspeedtest-cli/> (weil ich nicht erwarte, dass die Seite länger lebt als mein Eintrag)

Das hier ist erst mal nur interpretiert, daher ungetestet:

```

sudo apt-get install curl
curl -s https://install.speedtest.net/app/cli/install.deb.sh | sudo bash
sudo apt-get install speedtest
speedtest-cli -f csv --output-header >>/var/tmp/speedtests.csv
sed -i '"date time",' /var/tmp/speedtests.csv
sudo crontab -l > cron_bkp
sudo echo "22 */4 * * * /bin/bash -c 'echo \"\$(date +\%d.\%m.\%Y\
\%H:\%M:\%S)\",$(/usr/bin/speedtest -f csv)" >>/var/tmp/speedtests.csv'" >> cron_bkp
sudo crontab cron_bkp
sudo rm cron_bkp

```

Vorsicht Zeile 7 - ist meine unsicherste. Sonst crontab -e verwenden.

Das ganze Gnuplot-Gedöns habe ich hier mal weggelassen.

Backup vom Script mach ich aber mal.

```

cat gnuplot_speedtest_data
# Gnuplot for speedtests.csv
reset
set terminal dumb ansirgb feed enhanced size 128, 31 aspect 1;
set tics nomirror scale 0.5;
set autoscale;
set datafile separator ","

set title "Internetgeschwindigkeiten in Mbps\n[" . strftime("%d.%m.%Y %H:%M:%S",time(0)) . "]"

set xlabel "Datetime" offset 0,-1 # label for the X axis
set xdata time
set timefmt "%d.%m.%Y %H:%M:%S"
set format x "%d.%m\n%H:%M" # Display date time format on graph
# Show last 48h
set xrange [time(0)-2*24*60*60:]

set ylabel "Mbps" offset character 2,0 # label for the Y axis
#set yrange [0:] # show whole range from zero

# change legend position
set key right below nobox

set grid

```

```
plot "/var/tmp/speedtests.csv" using 1:($7*8/1000/1000) with line title "Download", '' using  
1:($8*8/1000/1000) with line title "Upload"
```

Aufruf:

```
gnuplot -p gnuplot_speedtest_data
```

Version #22

Erstellt: 2022-05-23 19:29:30 UTC von Konstantin

Zuletzt aktualisiert: 2023-09-04 20:04:51 UTC von Konstantin