

# test\_speed program1\_3 LCWA

## **version: 1.0**

Andi Klein

January 18, 2022

## **1 Overview**

test\_speed is a python program written by Andi Klein. It's main purpose is to collect data from different raspberry pi's distributed across the LCWA to check the bandwidth over 24 hours. The test gets executed every 10 minutes and the data are stored locally. Every hour the program connects to dropbox, and uploads the datafile and the plot file. At midnight every day, the program flushes all the data to dropbox and exits. The computer then does a git pull to a central repository, performs a pull and the restarts. This ensures that any program updates are caught by the clients. Until 2022, the program was using Ookla's speedtest and reporting those numbers. As is documented below, the user can choose the Ookla server, according to their preferences. Over time it has become clear that we would like to have a speedtest which is strictly within the LCWA network, and not a combination of this network and the internet. Several times we noticed strange test results, which were driven by an overloaded Ookla server and not a degradation of the LCWA network. This led to the development of a new addition to the code using iperf3. This iperf3 connection goes to a LCWA iperf3 server, therefore providing a more honest picture of what is going on in our network. This program runs on **Mac** and **Linux** but **NOT** on Windows. There are currently no plans to ever port it to Windows.

## 2 Installation

Before you can do anything, you have to clone the repo. Create a directory /git/speedtest with `mkdir -p git/speedtest`.

- `cd git/speedtest`
- `git clone https://github.com/pabloemma/LCWA .`
- if you need a different branch: `git fetch; git branch -v -a; git checkout branchname`
- `cd ../src`
- `./install_speedtest 2>&1 | tee install.log`

If you want to use the Ookla speedtest, you will need to install the following:

1. Install the speedtest CLI from <https://www.speedtest.net/apps/cli>
2. Once installed you need to run it once to accept the license
3. `pip install dropbox`
4. Mac: `brew install git`, Linux it is usually supplied, otherwise `sudo apt-get install git`, or `yum install git`
5. `mkdir /speedfiles` (below your home directory)
6. `mkdir git`
7. `cd git`
8. `git clone https://github.com/pabloemma/speedtest.git speedtest` (this you only do once, after this your command will be `git pull`, which will update you to the latest version)

9. You will need a dropbox account and setup an app. If you have a dropbox account, log in and then go to <https://www.dropbox.com/developers/documentation>, click on “App console” in the upper right corner and create a new app. There click on permission type and select which one you want. Once you generated the app, click on the button Generate under “Generated Access Token ”
  10. Copy into clipboard the Access token, which is very long
  11. open the file LCWA\_d.txt in your speedtest directory, remove the line and paste your access token.
  12. On dropbox also create a folder called “LCWA”, this is where the files will go.
1. install iperf3 (**not iperf or iperf2**)

### 3 Running and controlling the program

The program has currently two different ways to run; either speedtest, which is a wrapper around the Ookla speedtest or iperf, which will run an iperf test to a server on the LCWA network. The issue with the Ookla server is, that the result is depending on quite a few parameters beyond LCWA’s control. The current server is with cybermesa, and we have noticed that the server is sometimes slow. Also, since we go outside onto the internet, the results are influenced by whatever traffic is going on. The iperf test really checks the available bandwidth within the LCWA network, and is therefore a better measure of what we provide.

The program checks every 10 minutes the bandwidth, and records this in a file in the directory /home/pi/speedfiles. Every hour around xx:30 the csv file gets plotted into a pdf file and both are shipped to the corresponding dropbox. At midnight a second program collates all the different speedboxes, and creates a full 24 plot file. Also at midnight the program terminates, performs a git pull for updates, and then restarts again. In order not to blast the servers at the same time, each speedbox has a delay in the startup corresponding to the 2 digits in the name.

$$delay = XX * 25 \tag{1}$$

where the XX stands for the two digit speedboxnumber (LC04) would mean 04 as the multiplier and leads to a 100 sec delay.

### 3.1 Required directory settings

In order for me to support the program, I require the following directory structure:

/home/pi/git/speedtest is where all the components for the speedtest are located. The data files produced will be in /home/pi/speedfiles . If you choose a different configuration, you have to change the config file and you are on your own.

### 3.2 config file

Here is an example of a config file:

```
{  
  
  "Darwin" : {  
    "timeout" : "/usr/local/bin/timeout",  
    "speedtest" : "/usr/local/bin/speedtest",  
    "srcdir" : "/Users/klein/visual studio/LCWA/src/",  
    "datadir" : "/Users/klein/speedfiles/",  
    "conf_dir" : "/Users/klein/visual studio/LCWA/config/"  
  },  
  "Linux" : {  
    "timeout" : "/usr/bin/timeout",  
    "speedtest" : "/usr/bin/speedtest",  
    "srcdir" : "/home/pi/git/speedtest/src/",  
    "datadir" : "/home/git/speedfiles/",  
    "conf_dir" : "/home/pi/git/speedtest/src/config/"  
  },  
  "Control" : {  
    "runmode" : "Iperf",  
    "debug" : false,  
    "cryptofile" : "LCWA_d.txt"
```

```

},
"Iperf" : {
    "serverport" : "5201",
    "serverip" : "63.229.162.245",
    "duration" : 10,
    "numstreams" : 2,
    "blksize" : 1024,
    "latency_ip" : "65.19.14.51",
    "time_window" : 10,

    "reverse" : false
},
"Speedtest" : {
    "serverip" : "63.229.162.245",
    "serverid" : 18002,
    "time_window" : 10,
    "latency_ip" : "65.19.14.51"
}
}

```

As you can see , this is a json file (which personally I think is trash; who in their right mind develops a system, which does not allow comments) and we can look at it in some more details:

The first two block deal with the runtime environment. Since Apple and Linux have their stuff not in the same location (thank you apple) , we have to specify the locations of some critical files and directories. The next block **Control** deal;s with issues which are common to both runmodes and operating systems. **Iperf** selects the iperf test, while **Speedtest** controls space aliens (just kidding and making sure you haven't fallen asleep yet).

The last two blocks contain control parameters used for the two different runmodes.