TP 2: Monitoring Performance and QoS on the Internet

Carlos Gomes [pg47083], Miguel Gomes [pg54153], and Rui Armada [pg50737]

Universidade do Minho, Braga 4710 - 057, Portugal
Rua da Universidade, Braga, Portugal
https://www.uminho.pt/PT

Abstract. This test aims to investigate and contrast two distinct network measurement and monitoring platforms. With this in mind, we decided to examine two platforms: SamKnows, which is notable for its distinct features and novel methods of network monitoring, and Speedtest, a well-liked and extensively utilised application for determining internet connection speed.

Keywords: Quality of Service · Performance · Latency · Internet · Internet Performance · Network Measurement · Connection · Internet Connection

1 Introduction

For this second practical project, we explored the field of network measurement and monitoring and interacted with well-known technologies intended for these uses. We decided to investigate the features of Samknows[2][1] and Speedtest[1][2] among the numerous options available. With their distinct features and functionalities, these instruments are industry mainstays.

The main objective of this hands-on activity is to obtain a thorough grasp of the benefits that are built into each tool, while also examining the difficulties that they provide. Through an exploration of the nuances between Speedtest and Samknows, we want to uncover important information about how well they work, how accurate they are, and how useful they are for evaluating network properties. Furthermore, our investigation will go beyond simple assessment in an effort to pinpoint possible areas in which both tools could be improved.

We will explore the unique features of Speedtest and Samknows as we set out on our adventure, looking at their approaches, user interfaces, and underlying technology. Moreover, our examination will encompass actual situations, enabling us to put the useful consequences of employing these instruments in various network settings into perspective. We believe that our investigation will further both our own knowledge and the conversation around network monitoring technologies and how best to use them for digital connection optimization.

[1] https://speedtest.samknows.com/
[2] https://www.speedtest.net/

2 Speedtest

Ookla offers Speedtest, an online tool for assessing connection performance and speed. In October 2023, they achieved a significant milestone by carrying out 50 billion further tests [1].

2.1 Possibilities offered by Speedtest

It's possible to use the Speedtest on the web or with the Windows, Linux, and Mac apps that are available. This utility is, of course, also compatible with iPhones and Apple TVs. For people who also like to run commands in the terminal can also try Speedtest in the terminal. To put it briefly, Speedtest is essentially universally usable. In section 2.3, we'll demonstrate the use of this tool on Windows and on a terminal to get an idea of the results.

2.2 The measurement methodology in use, the performance and/or QoS metrics under control/analysis

One of the first features offered by Speedtest is simply to give you a quick overview of whether you're connected without having to make a full evaluation, and this is possible via the Network Status Feature. Basically, we just need to open the application and depending on the colour of the "GO" button, it's possible to know if either we are connected or not [1].

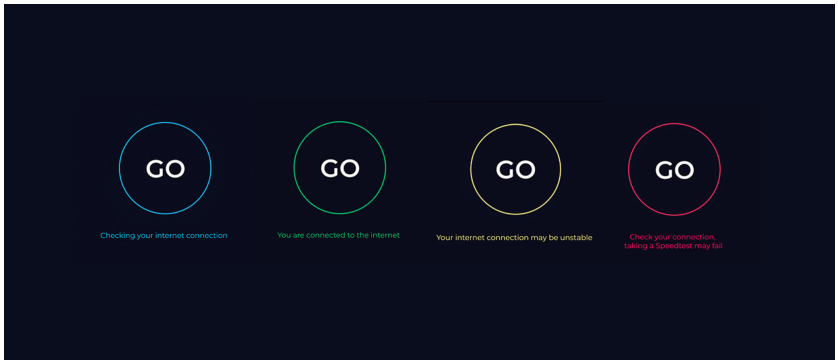

Fig. 1. Buttons Preview

As should be obvious, it's not just an aesthetic button, the moment this button is clicked, the analysis of the internet connection will begin:

TCP Test Components

Download test The client starts by establishing multiple connections to the server via port 8080 and asks the server to send an initial set of data. The client then calculates the real time speed of the transfers and adjusts the chunk size and buffer size in order to maximize the use of the internet. As it receives and requests data, the test only ends when the default time is reached.

Upload test The upload test is similar, but in this case the client sends data to the server. As with the download test, the size of the data blocks is adjusted dynamically based on the speed of the connection.

HTTP Legacy Fallback Testing

Latency Latency is measured through the response time of HTTP requests to the web server. As with the TCP test, these measurements are repeated and the lowest value is taken.

Download test In the download test, small binary files are downloaded from the server to the client to estimate the connection speed. Then, based on this result, an appropriate file size is selected for the real test which is performed is performed with cache prevention via random strings appended to each download. Throughput samples are received at up to 30 times per second and then they are aggregated into 20 slices, each slice being 5% of the samples. The remaining slices are averaged together to determine the final result [1].

Upload test To estimate connection speed, the client generates a small amount of random data and sends it to the server and based on this result, an appropriate amount of data is chosen for the upload test. The test is carried out on uniformly sized chunks that are sent to the server via POST. To maximize connection performance, up to four HTTP threads might be used. The chunks are sorted by speed, and the fastest half is used to compute the average and final result [3].

2.3 The measurement and monitoring facilities/tools offered to users

As mentioned earlier, there are lots of ways to run Speedtest, either via the application, the web or the terminal. Figure 3 shows an example of how to use speedtest from the terminal using the command $speedtest - fcsv > test.csv$. These arguments are optional, but it's just to show that if necessary it's possible to save these results in a JSON or CSV file. We even could use this and create a script to store the results in a single csv file.

Fig. 2. Speedtest in terminal, using Linux Enviroment

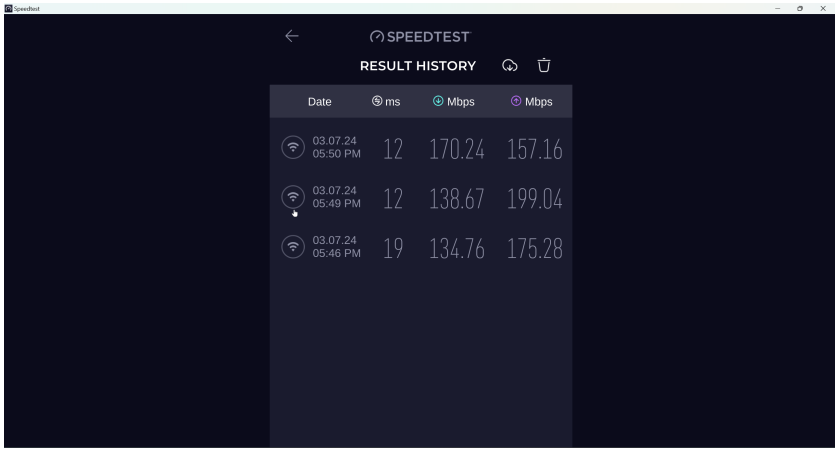

Fig. 3. Speedtest performed in Windows

2.4 Time problems associated with measurements

There are several details that can influence Speedtest results, for example, downloading a file while running the speedtest, thus causing network congestion. Or the use of different networks or even the location of the router due to the use of wifi, as there may be obstacles that could disturb the speedtest results. Not to mention the server selection that it's possible to choose from, and depending on the server we might get better or worse results.

2.5 Possible improvements in the services provided

Normally, these tests are performed to obtain the best possible results. However, it would be interesting to allow tests to be run in real-world network conditions that resemble how people use the Internet, with multiple applications, browser tabs, and devices competing for connectivity.

3 SamKnows

Cisco's SamKnows technology provides a range of tests, including latency tests, performance measurements for gaming servers, and content delivery network performance, among many others. This case will examine the method that Cisco created to gauge Internet speed on a local network [2].

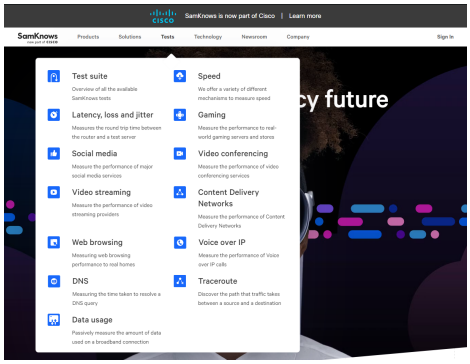

Fig. 4. Tests that SamKnows supports.

3.1 Speed tests

It's possible to perform some different kinds of internet speed measurement tests using this program. This page allows us to test the speed of TCP, UDP, WebSockets, and lightweight capacity (UDP technologies). The results of these tests vary in terms of what they reveal about the network's speed [2].

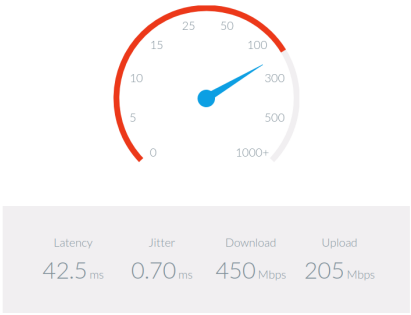

Fig. 5. Samknows speed test result.

TCP Speed Test Bits per second can be used to measure upload and download speeds with this kind of test. While the upload test creates its payload using */dev/urandom* as a non-blocking source of random content and tests its performance on the server side, the download test will obtain a random chunk of a binary payload from an HTTP server. The transfer rate limit can be adjusted to megabits for both tests, which run for 10 seconds by default. Because of this, these tests are convenient for customers who would rather have metrics based on bandwidth utilization for a set amount of time or volume.

Furthermore, there are four supported variations of this kind of test:

- A single download speed test using a TCP connection
- Download speeds for several TCP connections tested
- Upload speed test for a single TCP connection
- Upload speed test for several TCP connections

Around eight concurrent connections should be used for TCP tests with multiple connections, this number may need to be adjusted for situations involving high bandwidth and high latency [2]. By default, the program does not employ the 'cubic' TCP control congestion technique, although the HTTP test allows for its alternative adjustment.

There is a warm-up phase at the start of every test to address issues like TCP low start. The goal of this phase is to get a consistent throughput before running the intended test, which is eventually run across the same TCP connections. The warm-up ends after a minimum amount of time, 1.5 seconds by default, or a predetermined timeout, 5 seconds by default. Its data is recorded as an additional metric and is not included in the main test results.

Thus, the following metrics are recorded by the speed test client:

- The rate expressed in bits per second
- The volume of information sent
- How long the test will take
- Information regarding the warm-up phase
- The test server's IP address and hostname

Lastly, for certain hardware devices, the TCP speed test will leverage hardware-specific features. This is usually applied to devices whose CPU is not strong enough to fully utilize the WAN link when operating in software mode alone. These solutions differ in kind. In certain cases, a specific kernel driver is used to prevent needless data copying, and in other cases, a dedicated hardware co-processor is used.

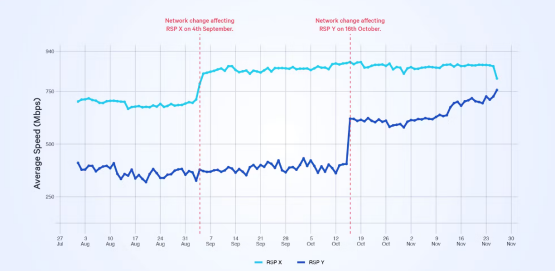

Fig. 6. ISP performance shift: After SamKnows' Fibre Max investigation in New Zealand, the download speeds of X and Y.

UDP Speed Test With support for both hardware and software acceleration options, the UDP test uses UDP at the application layer level to make an upstream and downstream measurement.

Higher speeds are achieved with hardware acceleration, which uses independent network co-processors and is available on some Broadcom and Qualcomm SOCs. This kind of test can be used in both modes in a comparable manner, notwithstanding the possible license costs and troubleshooting difficulties.

The client asks the server to do a UDP traffic test while the download is ongoing, progressively raising the transmission rate until a steady speed is attained. The client alerts the server and sends packets at full line rate during the upload process. By default, throughput is measured on the receiving end over a five-second interval.

Furthermore, a 'dual-channel' extension enables outdated devices with multiple internal interfaces to leverage hardware and software acceleration in simultaneously, eventually achieving measurements above 1Gbps on WAN lines.

The test records the following important metrics for both upstream and downstream:

- Throughput measurements
- Rate of transmission
- Bytes used in transmission
- Duration of transmission
- IP address and hostname of the server
- Packet size

Lightweight capacity test (UDP) A limited quantity of UDP packets are used in this test to assess the link's capability both upstream and downstream.

In downstream mode, the client specifies the transmission specifics as the server and client welcome one other over TCP. Refunded packet arrival times are indicated.

In upstream mode, the stream is transmitted, the client notifies the server of the impending transmission characteristics, and the server logs the arrival times. In this case, the client uses arrival times to determine throughput, enabling averaging over several windows to reduce buffering behavior.

Comparing this test to TCP tests, it takes 99% less data and completes in 100 milliseconds as opposed to 10 seconds. Nevertheless, there are compromises that reduce accuracy when compared to standard TCP or UDP testing, especially on DOCSIS[3] connections.

WebSockets speed test This test suite uses one or more concurrent TCP connections to assess upload and download speeds in bits per second. Web browsers that support HTML5 must be able to measure traffic using WebSockets and HTTP via the Fetch API.

While the upload test creates and transmits payload chunks, the download test retrieves data chunks from the server. It's crucial to point out that 10Gbps speeds have been observed in lab testing using contemporary technology and browsers.

With a transmission volume restriction that is optional, both tests have a set time. The configuration comes with eight TCP connections running simultaneously by default, chunk sizes ranging from 16KB to 1MB, and a warm-up phase to compensate for TCP's slow start. Excluding the warm-up phase data from the main results, the test results include measures of throughput, bytes transmitted, and duration time.

[3] Data Over Cable Service Interface Specification (DOCSIS) is an international telecommunications standard that permits the addition of high-bandwidth data transfer to an existing cable television (CATV) system. It is used by many cable television operators to provide cable Internet access over their existing hybrid fiber-coaxial (HFC) infrastructure.

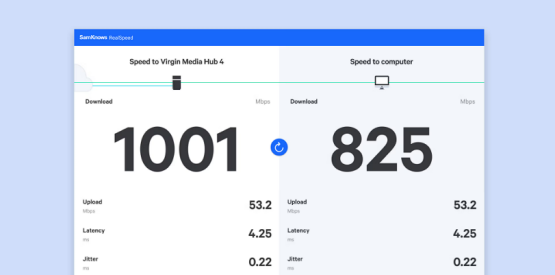

Fig. 7. RealSpeed: Web Socket speed test results from SamKnows, comparing device and router performance.

4 Discussion and Conclusion

Throughout this essay we focus more on the results analysis part and on the one hand these results obtained make sense and help to have a more real idea of the existing latency. We would have liked to have also explored more of the graphs demonstrated in the article, but due to the limits of this practical work we were unable to do so.

Overall, we think this study demonstrates the potential for innovative methods to understand and improve Internet connectivity, paving the way for future research and development in this area.

References

1. SpeedTest. "How does Speedtest measure my network speeds?" https://help.speedtest.net/hc/en-us/articles/360038679354-How-does-Speedtest-measure-my-network-speeds Last accessed on March 13, 2024.
2. Samknows. "Speed-Tests" https://www.samknows.com/tests/speed-tests Last accessed on March 11, 2024.
3. Ookla (2024). "Introducing Speedtest® Network Status" https://www.ookla.com/articles/introducing-network-status Last accessed on March 11, 2024.
4. Wikipedia contribuitors. "DOCSIS" https://en.wikipedia.org/wiki/DOCSIS. Last accessed on March 11, 2024.