

## Don't Just Test, Actively Test

I've written articles in the past discussing active testing and always get quite a bit of interest, feedback and questions.

For those of you who are new to this topic, I'll start from the beginning. As a general overview, active testing involves having a real client computer perform real commands, over your actual network to the production servers. If the product has scheduling and reporting then you can just review the results whenever you need them.

The reason why this methodology is valuable since this is the easiest way to get accurate measurements when testing. There are other tools and methods that involve simulating traffic or using computers to emulate applications which are great for modeling and testing infrastructure or equipment.

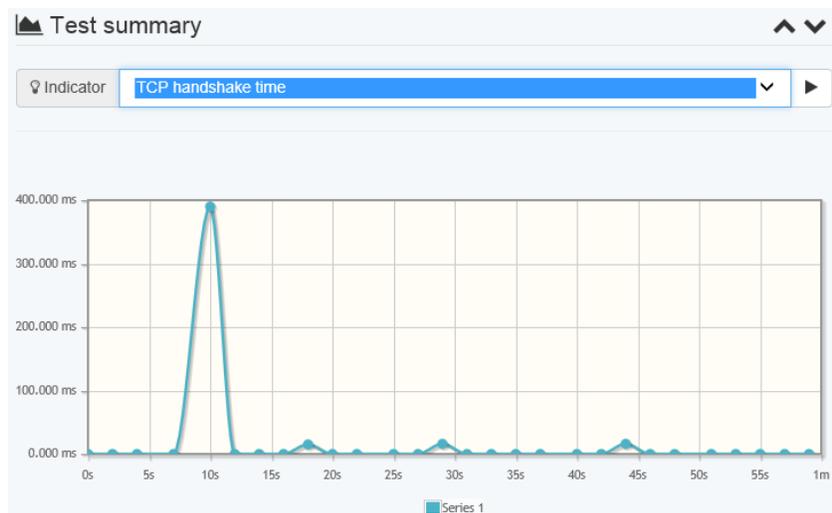
Not all products are the same and I encourage you to understand how the products work to ensure you get the one that best suits your needs. The key is to have the option to test between your agents as well as from the agent to a real server with real commands to get an accurate measurement.

Sooner or later you need to involve a tool or methodology that includes your real network and related components. Simulations and other traffic generation tools can't factor in server, client or network load which may be the root of your issue.

For this very reason I suggest you have an active testing tool in your toolbox. These tools serve many purposes such as troubleshooting, baselining and testing your equipment, infrastructure and policies.

The folks at Opale Systems ([www.opalesystems.com](http://www.opalesystems.com)) have released a freemium version of their Qual'it solution (<https://www.opalesystems.com/free-network-testing-solution>) which includes one console and 2 agents. The Qual'it product allows you to perform tests from the agent to a real server, like HTTP or between agents like TCP throughput tests.

Scheduling tests from Qual'it and the reporting are worth trying. For example the chart below is of a TCP SYN/SYN ACK during one of my tests. This would take a while to do manually not to mention how much more time it takes over longer periods.



This other image illustrates some of the thresholds used to determine if your test passes or fails.

The screenshot shows a network monitoring interface for a test pair. At the top, it says 'Pair 1' followed by status indicators: 'Success' (green), 'Service' (blue), 'HTTP' (blue), 'VMWARE' (blue), and 'Google' (blue). Below this, the test details are listed: 'Status : Success', 'Service name : Google', 'Threshold profile na... : Web application', and 'Service URL : www.google.com'. A table below shows the source and target information:

#	Name	Address	Service URL address
Source	VMWARE	10.44.10.124	www.google.com

Below the table, it says 'Thresholds results' with a plus icon. A table shows the results for three indicators:

Indicator	Result	Threshold direction	Threshold	Tolerance threshold	Result
Retransmits	0	←	50		Success
DNS resolution Average	18.6 ms	←	100 ms		Success
Total request time Average	140.4 ms	←	2000 ms		Success

This article is meant to introduce you to the product and I plan to do another article with more details. The installation is pretty straight forward and there are a few tips I can give you if you are new to the product:

- The default login and password is root
- From the console you add your agents in the Management -> Box area
- Check out Opale's youtube channel (<https://goo.gl/yDfT7d>)
- You can access the console from any browser using https://IP address
- Network pairs are tests between agents
- Service pairs are tests from the agent to a real server

I plan to write another article showing you Qual'it in a troubleshooting scenario.

Here is the youtube link of this review <https://www.youtube.com/watch?v=LurZriF32Jo>

Tony Fortunato is a network performance expert who has been designing, implementing and troubleshooting networks since 1989. His company, The Technology Firm, provides clients of all sizes with services ranging from project management, network design, consulting, troubleshooting, designing custom-designed training courses, and assisting with equipment installation. Tony blogs frequently at LoveMyTool.com, networkcomputing.com and has a popular YouTube channel.