
A Bigjar Systems, LLC White Paper
July 2024

Discover Latent Network Problems with Automated Bandwidth Testing (ABT)

Executive Overview

Everyone in IT has experienced this situation: End users discover a latent network problem when they come into the office, and IT staff must scramble to fix the problem. This white paper shows how you can use the Network Utility Knife (from Bigjar Systems, LLC) in batch mode to automatically find these problems before the users arrive, giving IT staff time to start repairs.

Introduction

Latent network problems are the bane of IT staff. Many network management systems can identify if a connection is hard down. However, because routing protocols are designed to be forgiving of packet loss, a problem that involves a noisy circuit or one that's flooded can be difficult to detect, to give some examples.

The latest version of the Network Utility Knife (NUK) has the ability to run jobs (batch mode) and one type of job tests available bandwidth. With the help of some shell scripts (included in this document and available for download) this type of job can be launched automatically with actions taken based on the throughput measured. In this way, IT staff can receive notifications when employees are not in the office, giving them more time to fix the problem.

In conclusion, the NUK can help you identify problems before users arrive in the office, giving you time to effect repairs. Nothing in the market at this price point can provide this functionality.

Now the reason the enlightened prince and the wise general conquer the enemy ... is foreknowledge.

- Sun Tzu

Automated Bandwidth Testing (ABT)

Here we describe in detail the scripts and programs involved in ABT. But first, an overview of the technology.

NUK jobs

Jobs allow for automated use of the utilities on the NUK. The included scripts submit jobs to test throughput (sample job files included). The script then waits for the output from the job, and takes action (e.g. sending an email) if necessary. This process can be repeated for different sites or different networks and for different NUK's.

Script logic

The logic for the scripts is as follows:

The **master.sh** script goes through all files in a directory and runs **bandwidthcheck.sh** passing each file as an argument. This allows for one executable to test bandwidth to multiple sites in sequence.

The **bandwidthcheck.sh** script does the following:

- Uploads the job file to the NUK specified in the ABTIP
- Waits a set period of time for the job to finish
- Downloads the output produced by the job
- Compares a value in the job output to a value specified in the job file
- If the value is less than the specified amount, then a third script is run (the script name is specified in the job file).

The **error.sh** script performs the action designated when bandwidth is too low. In the supplied script it sends an email to address specified in the ABTEMAIL variable containing debugging output from the job and the output generated by the NUK using the Unix **mail** command.

How jobs work

Jobs are launched on the NUK by uploading text files to a NUK via FTP or SFTP. The file is uploaded to `/storage/job/intake` (or `/config/storage/job/intake` if using SFTP). The specified utility is then run on the NUK and the output from this utility is stored on the NUK where it can be retrieved via FTP or HTTP.

Note: any extension is stripped from the job file and the output from the job is stored in `jobfilename.out` in the `/storage/job/output/` directory (accessible via FTP).

The format of the job file is as follows

Job file format

The text file (or job file) consists of multiple lines of the form “X=Y”. The lines define which utility is to be run, and what arguments should be passed to the utility. A sample job file is included below:

```
SCRIPT=srv-nett2.sh
PROG='iperfclient'
IPADDR='192.168.10.193'           # IP address of remote NUK
IPADDRVAR='iperfclient_ipaddr'
UOPTIONS='-w 800k -i .5 -O 1 -t 3 -J' # Options to pass to iperf3
```

There are many more possible options, but the above file will launch iperf3 in client mode and tell it to connect to 192.168.10.193 with the following options '-w 800k -i .5 -O 1 -t 3 -J'. From left to right these options have the following meanings:

-w 800k	Set the TCP Window to 800k <i>/**/</i>
-i .5	Report throughput every .5 seconds
-O 1	Omit the first second of testing from the total
-t 3	Test for 3 seconds
-J	Generate output in JSON format (<u>required</u>)

Additional lines of the form A=B can be added as long as these variables are not used internally by the NUK. We will add these lines to the job file so that a single file contains instructions both for the NUK and for the included scripts. To avoid name collisions, we will prepend ABT to each variable name.

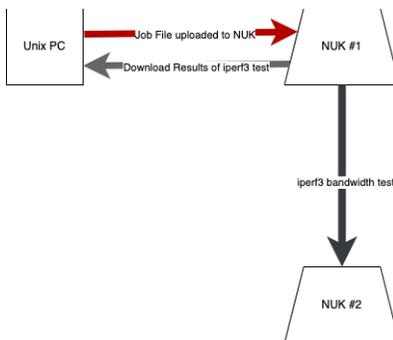
The full job file is included below:

```
SCRIPT=svr-nett2.sh
PROG='iperfclient'
IPADDR='192.168.10.193'
IPADDRVAR='iperfclient_ipaddr'
UOPTIONS='-w 800k -i .5 -O 1 -t 3 -J'
ABTIPADDR=192.168.10.203
ABTUSER=root # Username for upload. Usually root
ABTPASSWORD=password # Password for the above username
ABTFIELD=13 # Field to numerically test against.
# 13 = received bandwidth, 11 = sent bandwidth
ABTLESSTHAN=1 # Bandwidth below which an action is taken (Mbit)
ABTSCRIPT=./error.sh # Script to run if field 13 is less than expected
# An absolute path would probably be a good idea
ABTEMAIL="jmitchel@localhost" # error.sh sends email to the following address
```

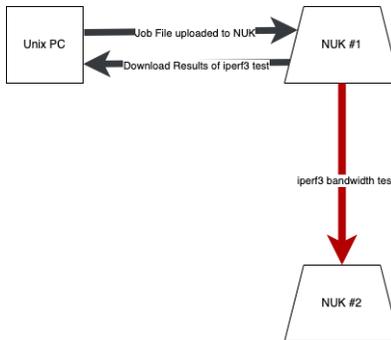
Diagrams

We know that some of you are visual learners, so here are diagrams showing the networking actions taken by the supplied scripts. In each diagram the step mentioned is shown in red.

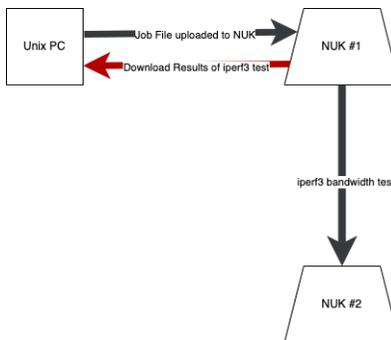
Step 1 – The job file is uploaded to the NUK



Step 2 – The iperf3 bandwidth test is performed



Step 3 – The results of the iperf3 test are retrieved



Conclusion

With the NUK and the supplied scripts, IT staff can get advance warning of problems before they impact users. This can improve productivity especially at remote offices where problems may remain undetected until users arrive for work.