NUTTCP - WHAT DO I GOT?

Any IT administrator thrives on information, the more you know about your network, systems, etc the better. It makes it easier to troubleshoot and to identify problems. Now a common problem is one of bandwidth. Whether this is internal or external bandwidth, you very often need to know just what type of pipe you have between two points and what it is capable of. A very useful tool for this is *nuttcp* (see here). You put nuttcp on each point and run the test between them to get an idea of the connection.

Getting started
First get *nuttcp* on your system. Most distro's have it as a package, but if not or if you like the source route, then go to the homepage (see here) and get the latest version. Once you have the source;

- unextract
- go into unextracted directory
- run '*make*'
- '*mv*' the compiled binary to a binary path, something like '*/usr/bin/nuttcp*'

Using it
The point of testing a network channel is that you have two endpoints, A and B. You want to test between A and B. So assuming you now have *nuttcp* on both we can begin. We start by making one of sides a server, lets use A..

```
# nuttcp -S
```

Nice and simple. This will start the server up on port 5000 so you need to be able to access that port. If needed you can set the port it listens on. Now comes the fun part, using the client (which would be B) to test the line. Lets start simple..

```
# nuttcp A
   13.7277 MB /  10.61 sec =   10.8495 Mbps 0 %TX 0 %RX
```

Fairly simple. The default test runs for 10 seconds and the result tells you;

- 13.7277 MB = the total amount of data sent
- 10.8495 Mbps = the average throughput per second
- 0 %TX = CPU usage on the transmitter/client
- 0 %RX = CPU usage on the receiver/server

Now lets try a few more things, like more verbose (-v) and per-second reports (-il)..

```
# nuttcp -v -il A
nuttcp-t: v5.5.5: socket
nuttcp-t: buflen=65536, nstream=1, port=5001 tcp -> 192.168.2.104
nuttcp-t: time limit = 10.00 seconds
nuttcp-t: connect to A with mss=1448
nuttcp-t: send window size = 8192, receive window size = 43690
nuttcp-r: v6.1.2: socket
nuttcp-r: buflen=65536, nstream=1, port=5001 tcp
nuttcp-r: interval reporting every 1.00 second
nuttcp-r: accept from 192.168.2.103
nuttcp-r: send window size = 16384, receive window size = 87380
nuttcp-r: available send window = 12288, available receive window = 65535
    1.1250 MB /   1.00 sec =    9.4241 Mbps
    1.0000 MB /   1.00 sec =    8.3912 Mbps
    1.1250 MB /   1.00 sec =    9.4306 Mbps
    1.0625 MB /   1.00 sec =    8.9156 Mbps
    1.1250 MB /   1.00 sec =    9.4403 Mbps
    0.8750 MB /   1.00 sec =    7.3353 Mbps
    1.1875 MB /   1.00 sec =    9.9644 Mbps
    1.1250 MB /   1.00 sec =    9.4399 Mbps
    1.1250 MB /   1.00 sec =    9.4403 Mbps
    1.1250 MB /   1.00 sec =    9.4304 Mbps
nuttcp-t: 11.5000 MB in 10.00 real seconds = 1177.55 KB/sec = 9.6465 Mbps
nuttcp-t: 184 I/O calls, msec/call = 55.65, calls/sec = 18.40
nuttcp-t: 0.0user 0.0sys 0:10real 0% 0i+0d 0maxrss 0+2pf 284+0csw

nuttcp-r: 11.5000 MB in 10.63 real seconds = 1107.71 KB/sec = 9.0744 Mbps
nuttcp-r: 8506 I/O calls, msec/call = 1.28, calls/sec = 800.12
nuttcp-r: 0.0user 0.0sys 0:10real 0% 0i+0d 0maxrss 0+21pf 8346+2csw
```

There are other useful flags;

- -T to change the default time used, to make the test longer
- -t to allow you to use your on input to test with via stdin
- -w to change the window size
- -h to see a listing of all the options, or just use 'man nuttcp'

Final Words
This type of tool is useful for many reasons..

- testing your external lines to see if you are getting what was sold to you
- testing lines periodically to see load and the effect it has on line speeds (graph that for a nice report)
- testing firewall throughput
- testing any type of encrypted setup

And many more. Basically the tool allows you to establish a baseline and then test against that baseline to see what effect any changes may have had. As always have fun and learn.