



www.internet2.edu

Network Measurement Security Policy: Recommendations

Disclosure/Disclaimer

This document is based on a presentation given by Matt Zekauskas, entitled Measurement Tools: Policies and Procedures. The presentation was prepared by Matt Zekauskas, using original material, and his own experience, and help from the Internet2 End-to-End Performance Initiative team.

This document was developed for use in conjunction with a Network Performance Workshop; for more information on these workshops (upcoming and past), see: <http://e2epi.internet2.edu/network-perf-wk/>.

Copyright © 2004, Internet2. All Rights Reserved, except that permission is expressly granted for others to use in noncommercial educational materials as long as attribution is given to Internet2 and the authors.

Overview: Security Policy

This document has information on:

- General security considerations
- Specific issues with recommended tools
- Abilene procedures

The focus of the section on general security considerations will be on using measurements to debug performance. This leads to specific issues about two tools that Internet2 has developed: One-Way Ping and the Bandwidth Test Controller.

Note: With regard to the section on Abilene procedures, specific policies and procedures are still in progress.

General Security Considerations

Overall, err on side of openness. Release all data unless there is a good privacy reason to keep it closed. (This mainly affects passive data; flow data and packet traces.)

Abilene uses the Abilene Observatory to collect and publish measurement data; because these collected measurements can be used to debug performance, Abilene tries to be as open as possible, and is willing to err (a tad) towards openness.

Approach

General approach:

1. Do no harm (prevent DDoS attacks)
2. Avoid being an attractive nuisance (harden machines).

The biggest issue for Abilene is not becoming or enabling a denial of service (DoS) attack. (We don't want to be the ones in the news...) On the other hand, we don't want to be paranoid. Measurement beacons are only effective if they can be used! Likewise, there is a tension between not exposing the machines so they won't be an attractive nuisance, and allowing them to be found so they won't be used.

Regarding hardening machines, a few recommendations:

1. Don't run anything you don't have to.
2. Keep up to date with security patches.
3. Perhaps run a local firewall (on the machine) if it makes sense. But see if it affects your measurement results...
4. Consider restricting logins, and where logins can occur from.
5. (If you're really good) Audit programs on the machine.

UDP / TCP Issues

Don't worry about TCP. Do worry about UDP. Do think about where most of your tests are going, and how much you're willing to test common links (like campus to GigaPoP). Will Internet2 and Commodity traffic interfere?

TCP vs. UDP

TCP tests require acknowledgement before data, so other side must collude. UDP tests do not have these requirements (except if control code forces them to do so, which is the case for the tools we talk about). TCP backs off in the presence of other flows; UDP does not. So, you can be more open with TCP, especially a TCP sink.

Privacy

User privacy is not a concern with active tests. Keeping network hidden hinders debugging. Why not release SNMP, at least border router? Also, don't totally block ping, traceroute, etc. Do watch for things that might identify users in passive measurements.

NDT

NDT is a restricted web server that primarily does TCP testing. It allows very restricted UDP sequences (only a few packets), so there is no reason to fear sticking an NDT server at the campus edge. **One-Way Ping/Bandwidth Test Controller**

Issues

More details are available via the tool-specific cookbooks (see <http://e2epi.internet2.edu/library-list.html>); for the purposes of policy, note that the tools:

- Can classify by source IP address or AES key (~password)
- Limits placed on resources they use

For example: NOC class can do anything. Peer class can do anything, but limits to 30 second total duration. Application community further limited to no UDP. **One-Way Ping-Specific Considerations**

- Anonymous tests restricted to requesting machine (so no 3rd party attacks)
- Generally very low rate.
- Can test 24x7
- Limit usually in ability to process results!
- Consider continuous testing with average rate 1/sec/path.

The good news is that there's not much traffic, so not much you need to worry about. On Abilene, we do require the requesting machine to be the target, to prevent 3rd party attacks. Also, for Abilene, the tests are in the noise, but the tests themselves (at 10/sec) generate 64 bytes/test, and it adds up quickly.

A 1/sec rate to places you care about will give you fairly fine-grained data, but not be overwhelming. This can be reduced in cases where the data rates are high.

Bandwidth Test Controller-Specific Considerations

- Generally, no anonymous usage
- Periodic tests useful, but don't want to interfere with production traffic
- Recommend TCP, not UDP (so traffic "elastic")
- To points you care about, such as your GigaPoP; 3 points in Abilene core; universities you collaborate with; and an application community

Restricting the rate of requests is something we would like to do, but it's not possible. For example, this would allow low-rate anonymous TCP requests that are time-limited, so the anonymous requests themselves cannot be attacks on our server, and won't be considered dos attacks themselves.

There is a tradeoff between testing and getting work done. It is easiest to do low-frequency TCP tests; you don't do them very often, and if you happen to coincide with another flow, it's TCP. And, usually, Reno TCP at that, so it backs off quickly. Thus, target your periodic testing to points you really care about, watch peering links, and watch universities with which you collaborate.

Resources

The resources needed by both tools include bandwidth for maximum throughput (although, UDP only with the Bandwidth Test Controller) and whether to allow an unauthenticated mode. For One-Way Ping, resources are needed for disk usage (results must be stored temporarily); for the Bandwidth Test Controller, resources are needed for test duration, determining how far in advance you can schedule, and determining how many requests are pending.

AES Keys

Just a general warning: these are really passwords that are hard to type. They are symmetric keys. More details later.

Auditing

Accepted sessions go into syslog – note that there is a lot of audit information that goes into syslog, especially when things go wrong. However, there is not enough to tell exactly who executed what for successes. Through this, you can get identities and endpoints.

Abilene Procedures

It is the Abilene goal to be an exemplar; measurements are open, tests are possible to router nodes, and throughput tests are run routinely through backbone. This is in addition to the usual testing for utilization, etc. For more information, see the “Abilene Observatory” at <http://abilene.internet2.edu/observatory>.

Machines

We currently have four machines at each router node. Here are their roles:

1. A GigE-connected high-performance tester for the Bandwidth Test Controller (nms1) with a 9000 byte MTU
2. A latency tester for One-Way Ping (nms4) with 100bT
3. A statistics collector for SNMP and flow-stats (nms3) with 100bT
4. A GigE-connected ad-hoc tester for NDT (nms2) with 1500 byte MTU

Throughput

Abilene uses the Bandwidth Test Controller for throughput. We do IPv4 and IPv6 (and they give equivalent results) for TCP and UDP; although we do UDP testing, we would

not recommend a very high rate of UDP testing for production networks. We take tests once per hour for 20 seconds each, and others test to our nodes and amongst themselves: the net result is 25% of traffic (NOT capacity) is measurement. That's a lot of measurement traffic!

Latency

Abilene uses One-Way Ping for latency. CDMA is used to synchronize NTP. We test among all router node pairs at a rate of 10/sec, for both IPv4 and IPv6. We use minimal-sized packets on a Poisson schedule. (A "poisson schedule" means that the tests are variably spaced; the idea is to take a "random sample" of times during the period, and if you sample periodically you might just miss an event that is also periodic.)

Abilene Policy

Abilene policy is still evolving. As stated earlier, we tend toward openness. We are willing to give up 10% of capacity to testing. One Gig-E limited hosts could run flat-out; so we may rethink this as usage rises. We run throughput and latency tests all the time (2/hr/pair for throughput and 10/sec/pair for latency). ***Abilene, current state***

Several Bandwidth Test Controller hosts are running with schedules at near capacity. We are currently deploying additional machines to absorb the load but, in the future, we may only allow for one periodic test point, and then only at a low rate.

Abilene Procedures

For more information on Abilene procedures, see:

<http://e2epi.internet2.edu/pipes/ami/bwctl/>. The application for testing is reviewed manually, and, if approved, AES keys distributed. As of July 2005, we haven't turned anyone down yet who has requested access.

Emergency stop

In a situation that called for an emergency stop, you can shutdown the bwctld daemon, remove the offending key (access), and restart. On Abilene, the NOC can do this for us. (In extreme cases, we could just shutdown the interface.)

Placing Machines

Finally, a few words on where the test systems should be placed. If you only have one place to put a machine, the campus edge is probably the place. If you can do more than one (or in special cases, even if you can only do one), consider important applications or servers.

Example: Test Abilene Access

First, place Bandwidth Test Controller and One-Way Ping boxes at campus edges. Then, peer with an Abilene node. Then, peer with other campuses important to you; if none obvious, then randomly take three or so sites. Run One-Way Ping once a minute to once a second and Bandwidth Test Controller at the rate appropriate for you, between once and four times a day.

For More Information...

Internet2 E2E piPEs Project

The focus of this effort is to develop an end-to-end measurement infrastructure capable of finding network problems. The tools used by this project include Bandwidth Test Controller (latency), One-Way Ping (throughput), and NDT (last mile issues). Each of these tools has a cookbook similar to this one. They can all be accessed through <http://e2epi.internet2.edu/library-list.html>.

Availability

The tools used on Abilene are available via the E2Epi website (<http://e2epi.internet2.edu/>).

Publicly-Accessible Servers

A list of publicly-accessible servers on Abilene, and other public servers that have reported themselves, is available at: <http://e2epi.internet2.edu/pipes/pmp/pmp-dir.html>. Note that this is not a complete list and more are being added when they become available. Several institutions, not listed below, run private servers.