A Tale of Two Problems:

Duplex Mismatch at SC2002

A Case Study for E2Epi

www.internet2.edu

What is E2Epi? E2Epi is Internet2's End-to-End Performance Initiative. For more information about us, see our website at: **http://e2epi.internet2.edu/.** There you'll find links to recent projects (like **E2E piPEs**) and presentations, as well as other information related to end-to-end performance issues, measurement tools, upcoming events, and related activities.

What *is* a cakebox? It is a small, inexpensive PC running Linux. The cakebox is configured such that, when you plug it into a DHCP-enabled Ethernet port and give it power, it registers its presence with an LDAP server, where it logs its current IP address so you can "find" it. You can then connect to it remotely and run a series of network utilities (like Iperf, traceroute, and pchar). Cakeboxes were developed by Internet2 to test H.323 video conferencing capabilities and have been used for a variety of other end-to-end performance tests. Instructions on "building" a cakebox are available at: http://apps.internet2.edu/cakebox/cakebox-install.html.

Bob arrived at the Baltimore Convention Center shortly after noon on Monday, the 11th of November. After experiencing several end-to-end performance problems at the Internet2 Fall Member Meeting in October, Bob wanted to test the connection ahead of time to ensure that the SC 2002 netcast, scheduled to begin the following morning, would proceed smoothly. Armed with several **cakeboxes**, his laptop, and his years of experience, Bob located the netcasting site on the upper level of the Center and immediately began running systems tests.

He first tried testing the application. The Internet2's Virtual Briefing (VB) facility in Ann Arbor, Michigan, has a VBrick set up to receive VBrick MPEG-2 traffic and then feed it into the "normal" netcast streaming technology. This allowed Bob to stream MPEG-2 traffic from the SC02 netcasting center to the Ann Arbor VBrick while "watching" the video stream as users would on the next day. Immediately, he found a problem: the video quality was poor – pixilated and jerky – indicating packet loss.

Bob then used the cakeboxes to try and locate where the loss was occurring. A cakebox was already installed at the VB facility and at the Merit gigaPoP (which connects the VB facility to Abilene); Bob installed another inside the Convention Center. He first tried 90 Mb/s UDP tests from the VB facility to the cakebox in the Convention Center. He did not see any packet loss or significant jitter, indicating that the path from the VB facility into the convention center looked fine; he repeated the test in the opposite direction to be sure, and it was also normal. This indicated that the problem was local to the Convention Center itself.

The cakebox in the Baltimore Convention Center was in the Internet2 booth, not the netcasting room. Bob had a colleague, Matt, who was working on the show floor networking staff; Bob wandered over to the conference Network Operations Center (NOC) to find him. Matt has long been a problem-solver in the community, and when Bob described the symptoms, Matt immediately suggested an Ethernet **duplex mismatch** (see page 2); Bob concurred.

Bob recalled that there was an Ethernet switch installed in the netcasting room. Because the sophisticated Ethernet switch usually used for netcasting was being used elsewhere, a relatively "cheap" switch had been placed in the netcasting room. Bob and Matt went up to the room to examine the switch. It was a true switch (not merely a *hub*, which will do only half-duplex connections), but there was no overt indication whether connections were full- or half-duplex.

They returned to the conference NOC and verified that the test netcast from the VB facility was received properly inside the NOC. They then went and found a routing engineer, Brent, and asked if they could check the setting of the building switch to which the netcasting room switch was connected. The building switch had that particular port set to full-duplex. Since the netcast switch had no adjustments, Bob and Matt asked Brent to reset the port to auto-negotiate.

Voila! They checked the port, and it had auto-negotiated a full-duplex connection. (Therefore, the switch in the netcasting room could *only* auto-negotiate.) Bob went ahead and retested by playing the stream from the VB facility on his laptop connected to the netcast switch. Everything looked fine. The quality of the audio and video was excellent. Bob and Matt believed that everything was set for the netcast the following morning.

But, *no*! The next morning, Bob arrived at 7:30 a.m. to double-check the performanceg. He tried to view the facility stream on his laptop – and received *nothing*!

What is Internet2 Detective? It is an easy-to-install software that checks to see if basic functionality is available – is there a path to an Internet2 backbone? Does multicast work? Is IPv6 available? The Internet2 Detective can save the user time by verifying that the network fulfills all the necessary requirements to support specific applications. For information about this software, see **http://detective.internet2.edu/**.

What is **duplex mismatch?** It is a common cause of end-to-end performance problems that is often manifested by a connection which works at low speed, but drops packets when high-speed flows are attempted (usually with some traffic flowing in the opposite direction). It is due to a failure in the protocol used to negotiate whether an Ethernet connection can have traffic flowing only in one direction at a time (*half-duplex*) or in both directions at the same time (*full-duplex*). It is apparent that a mismatch occurs when one machine is set at full-duplex and another at half-duplex. What is not so obvious is that:

- When auto-negotiate meets auto-negotiate, the transmission rate is full-duplex;
- When half-duplex meets auto-negotiate, the rate is half-duplex.

In both cases, no problems occur. When auto-negotiate meets full-duplex, however, auto-negotiate converts to half-duplex and **not**, *as you would expect*, to full-duplex – the auto-negotiation capability is programmed to *assume* half-duplex unless it meets another machine set at auto-negotiate.

This time, Bob started the **Internet2 Detective** on his laptop. He found that multicast was not working.

IP multicast minimizes the amount of traffic being sent over the network. When there are multiple receivers, only one multicast stream can feed them all; if the system did not use IP multicast, every receiver would require its own stream. IP multicast scales to large audiences and allows delivery of "TV quality" video to user desktops.

Bob contacted Matt again, who remembered that some of the routing engineers had found a router software bug that required disabling multicast for the duration of an overnight demo. He suspected that perhaps IP multicast had not been re-enabled; Matt checked and, indeed, that was the case. With the help of a routing engineer, multicast was quickly re-enabled, and Bob's test completed cleanly. The netcast session was a success. Total time to problem resolution: 7 hours. If either of the problems had not been discovered well before the event:

- There would have been no netcast the first day, and possibly not on the second either.
- The problem-solving process would have been further hindered by NOC staff needing to focus on demands created by the event vs. helping Bob with his problem.

Recommendations

- Set switches at auto-negotiate wherever possible – most new equipment comes with auto-negotiation switches as the default. If ANY switches are set to FULL duplex, this will cause a problem. Steve Wallace (who encounted numerous duplex mismatch cases during his tenure at the Indiana NOC) recommends: "If you remove hardwired full-duplex from the mix, you shouldn't ever have a mismatch. Modern devices that support auto will negotiate to full. Let's say you decide to hardwire ports to full, and configure the host for full. The next time someone puts a new host on that switch port, it will probably be configured (the host) for auto (that's the way they come by default), and you'll have a mismatch. "
- Tests should be run during times of typical (or worst case) load; problems may disappear during off peak times.
- Test early; it is often harder to solve problems when an event is actually running because those who might be able to assist will be distracted by other responsibilities.
- Test often – other people who have access to the network make changes or demands on the network that you cannot predict. Let those who can turn the knobs know you care and why.
  Before you make an equipment change, think of how that change may affect other users; where possible, let them know of the change or let the network administrator know so that problems caused by the change can be more easily identified and resolved. Log the changes!
- Don't immediately *assume* the network is the problem and dismiss it as "out of your hands" – test for capacity and congestion.
- Ensure you are connected to the network (**Internet2 Detective** can tell you this, among its other uses).
- Learn about various diagnostic tools that are available, such as **cakeboxes**, which are inexpensive and easy to install at key points to quickly eliminate specific paths as possible problems and the **Network Diagnostic Tool (NDT)** which diagnoses problems experienced from the desktop.
- Internet2's **E2Epi** is currently working on a design for common performance measurement points that campuses can follow, and later integrate into the **E2E piPEs** project, which will allow campus network engineers to initiate tests to points on other campuses.
- To keep informed on the latest tools and techniques for problem-solving end-to-end difficulties, join the **E2E-Perf Interest Group** via the E2Epi home page.
- Do consider, and eliminate, the most common causes of end-to-end performance problems *first* – **duplex mismatch** and connectivity. Remember that a telltale sign of duplex mismatch includes spotty transmissions (especially if noted in the audio).