Testing of Ethernet Switch

Acute Communications Corp.
Victor Yao-Tzung Wang
ytw@acutecomm.com.tw

Contents

- Why Test and Network Subsystems
- Network Test Environment Alternatives
- Philosophy and Methodology of Testing
- Test Classification
  - Functional, Negative and Stress tests
- Testing Network Subsystems
  - An Example: Ethernet Switches: ASIC/System
- Useful Information
- Summary
Why Testing Networks?

- Keeping the network up and running
- Making the network better
- The cost-effectiveness of testing
- Third-party testing

Seven Network Subsystems

- File server (S1)
- Workstation (S2)
- Networking operating system: NOS (S3)
- Application, client-server database and workstation desktop software (S4)
- Hubs, switches, bridges and routers (S5)
- Network segment (S6)
- Internetwork (S7)
Recommended Test Objectives

<table>
<thead>
<tr>
<th>Network Subsystem</th>
<th>File Server</th>
<th>Workstation</th>
<th>NOS</th>
<th>Application Client Server Database</th>
<th>Router/Hub/ Switch/Bridge</th>
<th>Network Segment</th>
<th>Internetwork</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommended Test Objectives</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Test Environment Dimensions

<table>
<thead>
<tr>
<th>Network Load</th>
<th>Real-world load</th>
<th>Real-world load</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real-world load</td>
<td>Emulated network</td>
<td>Real-world network</td>
</tr>
<tr>
<td>Emulated load</td>
<td>Emulated network</td>
<td>Real-world network</td>
</tr>
</tbody>
</table>
Emulation vs. Real-World

- Highest cost
  - More time
  - More accurate
- Lowest cost
  - Less time
  - Less accurate

Traffic Load Modeling

- What type of loads?
  - A Windows client-server application
  - A DOS word processing application
  - A packet generator

- How much load?
  - Number of users
  - Network load (%)

- Mixed-case loading

- Accelerated loading
  - Used for reliability testing
Test Philosophy

- Response time
- Feature/Functionality
- Throughput
- Acceptance
- Configuration sizing
- Reliability
- Bottleneck identification and problem isolation

Test Methodology

- An orderly system of procedures to ensure that the test results meet the test objective.
- The test results should be
  - Accurate
  - Reproducible
  - Relevant
Test Methodology Components

- Planning
- Load modeling
- Test configuration
- Data collection
- Data interpretation (Relevant results)
- Data presentation (Actionable results)

Test Classification

- **Functional Tests**
  - Verify that the DUT does what the specification says it should or must do

- **Negative Tests**
  - Verify that the DUT behaves appropriately if another device on the network is not functioning according to specifications

- **Stress Tests**
  - Verify that the DUT can perform correctly on busy networks with many devices and a high volume of network traffic
Test Setups

Conversational Testing

Multi-Interface Testing

Virtual Production Network Emulation

Multi-Interface Testing

DUT Multilayer (IP) Switch

UI
An Emulated Network (1)

A \rightarrow B \\
1 \\
1 \\
1 \\
C

\rightarrow

DUT A

\rightarrow

B \rightarrow C

TESTER

An Emulated Network (2)

A \rightarrow B \\
1 \\
1 \\
D

\rightarrow

DUT A

\rightarrow

B \rightarrow C

X \rightarrow D

TESTER
Prioritization

Routing Switch

- HI-Pri Stream
  - 75% line rate
- LO-Pri Stream
  - 75% line rate
- Overload (150%) - 10 seconds
- Streams Sink
  - Analyze the loss rate of two streams

System Test Plan (1)

- Product Overview
- Test Equipment and Test Platform
- Basic Function Test (10/100/1000 Mbps)
  - More than 30 test items
- Protocol Conformance Test
  - GVRP/GARP
  - GMRP
  - Spanning Tree Protocol
  - IGMP2
  - RIP2
  - OSPF2
  - DVMRP3
  - MOSPF
  - Self-Developed
System Test Plan (2)

- **Performance/Reliability Test**
  - Refer to IETF RFC-2330: Framework for IP Performance Metrics
  - About 15 test items
    - SMB-AST (Advanced Switch Test): Max throughput, HOL blocking, X-stream
    - Ixia Communications: Mesh peak load test
    - Self-developed: BX-stream, EBX-stream, MCAST X-stream, mixed class X-stream

- **Multivendor Interoperability Test**
  - Use different vendors’ router equipment to build real-world (tree or star Topology) operation
    - Cisco, Bay Accelar, Xylan, Ascend, etc.

- **MIB Verification Test**

Victor Y.T. Wang

System Test Plan (3)

- **IP Multicast Test**
  - Refer to IETF Draft: draft-thaler-multicast-interop-03
  - Refer to IETF RFC-2432: Terminology for IP Multicast Benchmarking

Victor Y.T. Wang
IP Multicast Test (1)

- **Multicast Speedup Index (MSI)**
  - The ratio of unicast latency \( (d) \) to multicast latency \( (D) \), i.e., \( (d/D) \).
  - In the best case, \( D = d \) \( \rightarrow \) MSI = 1
  - In the worst case, \( D = n\cdot d \) \( \rightarrow \) MSI = \( 1/n \), where \( n \) depends on the destination multicast ports

- **Multicast Latency (ML)**
  - The set of individual latencies from a single input port on the DUT to all tested ports (more than two ports) belonging to the destination multicast group

- **Group Join Delay (GJD)**
  - Time duration when an IGMP report has been issued to a DUT until the DUT starts forwarding multicast packets

- **Group Leave Delay (GLD)**
  - Time duration when an IGMP “Leave Group” message has been offered to a DUT until the DUT ceases forwarding multicast packets

IP Multicast Test (2)

- **X-Stream**
  - \[ \begin{array}{cccccccc}
  1 & 2 & 3 & 4 & 2 & 3 & 4 & \ldots \\
  2 & 3 & 4 & 1 & 3 & 4 & 1 & \ldots \\
  3 & 4 & 1 & 2 & 4 & 1 & 2 & \ldots \\
  4 & 1 & 2 & 3 & 1 & 2 & 3 & \ldots \\
  \end{array} \]

- **MCAST X-Stream (k)** /* input line load = 1/k and k=2 (typical value) */
  - \[ \begin{array}{cccccccc}
  1 & b & c & d & b & c & d & \ldots \\
  2 & c & d & a & c & d & a & \ldots \\
  3 & d & a & b & d & a & b & \ldots \\
  4 & a & b & c & a & b & c & \ldots \\
  \end{array} \]

  - Multicast groups for \( k=2 \)
    - \( a=\{1, 2\} \)
    - \( b=\{2, 3\} \)
    - \( c=\{3, 4\} \)
    - \( d=\{4, 1\} \)
**IP Multicast Test (3)**

• Mixed Class X-Stream \((k + 1)\) /* input line load = \(1/(k + 1)\) */

\[
\begin{array}{cccccccccccc}
1 & 2 & b & 3 & c & 4 & d & 2 & b & 3 & c & 4 & d & \ldots \\
2 & 3 & c & 4 & d & 1 & a & 3 & c & 4 & d & 1 & a & \ldots \\
3 & 4 & d & 1 & a & 2 & b & 4 & d & 1 & a & 2 & b & \ldots \\
4 & 1 & a & 2 & b & 3 & c & 1 & a & 2 & b & 3 & c & \ldots \\
\end{array}
\]

**Useful Information**

- **Protocol Conformance/Interoperability Lab**
  - http://www.iol.unh.edu/

- **Performance Evaluation**
  - http://www.tolly.com/

- **Testing Tools**
  - SmartBits: http://www.netcomsystems.com/
  - ANVL: http://www.midnight.com/

- **Standards**
  - RFCs: http://www.ietf.cnri.reston.va.us/
  - IEEE802.1: http://grouper.ieee.org/groups/802/1/
  - IEEE802.3: http://grouper.ieee.org/groups/802/3/
The value of testing = risk versus cost

- Risk includes
  - productivity loss
  - support cost
  - lost sales

- Cost includes
  - facility cost
  - personnel cost