Stateful Traffic Generator®
10/100/1000G, 10G & 40G Ethernet

Manufactured By
East Coast Datacom, Inc.
in collaboration with developers
Seven One Solution and NM²
Why a traffic generator?

• A traffic generator is useful for:
  – Network monitoring, analysis and performance evaluation
    • Throughput, packet loss, delay and jitter analysis in heterogeneous networks (WAN, Wired LAN, Wireless LAN, GPRS, Bluetooth, ...)
  – Testing device capabilities
    • Desktop PCs, Notebooks, Pocket PCs, Smartphones, ...
  – Testing Quality of Service (QoS) architectures
    • Queuing disciplines
    • Traffic shapers
  – Routing algorithms analysis
  – Traffic Engineering
  – Scalability and Protocol behavior analysis

• ... using synthetic but realistic application fingerprints
Hardware-based vs Software-based

• HW-based traffic generators
  – Usually very precise
  – Line Rate performance via FPGA Design
  – Expensive due to FPGA Design & Associated Hardware

• SW-based traffic generators
  – Open-source (often)
  – Easy deployability on multiple nodes (*e.g.* hundreds)
  – Easily extendable and modifiable
    • New features, statistical profiles, support for new hardware and OSs
  – Ability to test actual implementations
    • Real OSs
    • Real protocol stacks
Model-Based vs Trace Driven

• **Model-Based**
  – Uses statistical models to emulate applications traffic profiles
  – Highly flexible
  – Highly scalable

• **Trace-Driven**
  – Reproduce traffic traces stored in packet-trace files
  – Real payload at all layers
  – Accurately reproduce the behavior of the application
Why a new traffic generator?

• New features
  – Increase supported traffic patterns
  – Introduce a distributed and coordinated mode of operation
  – Introduce a Log server
  – Bridge the gap between trace-driven and model-based generation

• Improved performance
  – Increase generated bit rate via INTEL® DPDK on COTS Hardware
  – Increase received bit rate
  – Increase scalability

• Improved accuracy

• Improved features

• Achieve Full Line Rates with INTEL® DPDK Implementation

• Much Lower Cost compared to FPGA based Solutions
Some Features (1/2)

• **Distributed**
  – Multiple senders and receivers are allowed
  – Each sender can simultaneously generate multiple flows
  – Logging can happen both at sender and receiver side. Data can be sent directly to a log server

• **Can reproduce realistic traffic patterns or real traces**
  – Protocols at multiple layers: IPv4, IPv6, ICMP, TCP, UDP, SCTP, DCCP
  – Application layer protocols statistically replicated at packet level: Telnet, VoIP (G.711, G.723, G.729, Voice Activity Detection, Compressed RTP), DNS, network games, ...
  – Pcap traces

• **Reach high generation bit rates (1-40 Gbps) and packet rates (> 80 Mpps) on COTS hardware and non real time OSs**
Some Features (2/2)

- Novel transport protocols
  - DCCP, SCTP
- Both model-based and trace-driven generation
- Possibility to perform measurements with trace-driven generation
- High accuracy in the model through different kinds of models and accurate packet timing
- Highly flexible and customizable
- Intuitive web-based GUI
- High-performance generation thanks to optimized software-hardware coupling with INTEL® DPDK Development Implemented
Configurable parameters

- **Host level (specified for each Box)**
  - Log type (sent packets, received packets) and location (local or remote).
  - Logging information inserted into the packet payload (none, minimum, extended).

- **Flow level (specified for each individual flow of each Box)**
  - Duration and time to wait before start.
  - Type of application to emulate (DNS, Telnet, VoIP, Counter Strike, Quake 3).
  - Random distribution for the size and the inter departure time of the packets (Constant, Uniform, Exponential, Pareto, Cauchy, Normal, Poisson, Gamma, Weibull).
  - Seed of the random number generator.
  - Direction of the traffic (one way, round trip).
  - Inter departure time recovery mechanism.

- **Transport level (specified for each individual flow of each Box)**
  - Source and destination ports.
  - Transport protocol (TCP, UDP, SCTP, DCCP)
  - Nagle algorithm for TCP.
  - Stream ID for SCTP.
  - Congestion Control for DCCP.
  - ICMP message (specified for each individual flow of each probe)

- **Network level (specified for each individual flow of each Box)**
  - Time To Live.
  - Differentiated Services Byte (Type of Service).

- **Operating System level (specified for each individual flow of each Box)**
  - Increase priority of the generation thread.
  - Network interface to bind to.
  - Serial ports on which to rise a signal for sent and/or received packets.
Optimized software-hardware coupling

• The generator has been optimized for a particular combination of hardware CPU, NIC, etc. so to achieve high generation rates
• Direct interaction with the NIC, using special drivers
• Up to 40Gb in turbo mode (back-to-back packets)
  • TCP and UDP packets with configurable fields
• Several Gbs in standard mode with full L7 semantics
Model-based generation (1/2)

• Generates traffic on a packet-by-packet basis: a *packet-level traffic generator*

• A single traffic flow is specified through
  – Start time
  – Duration
  – Packet *Inter Departure Time (IDT)*: the time between the transmission of two subsequent packets
  – *Packet Size (PS)*: the amount of data being carried by the packet.
Model-based generation (2/2)

- It is possible to reproduce exactly the same stochastic experiment by choosing the *same seed* value for random processes.
Trace-driven generation: main features

- **Bidirectional**
  - From ITGSend to ITGRecv and vice versa

- **Synchronized**
  - Packets sent in the two directions mimic the behavior of the two parties of the application
  - Using timestamps or bytes

- **Automatic adaptation of layer-7 information**

- **Multi-flow through concurrent server**

- **Performance measure collection**

- **Uses PCAP trace format**
Accuracy

- Even at low rates, the accuracy in actually replicating the defined models can be very low in some software-based traffic generators.
Some application scenarios

- Generation of realistic background traffic for testing new application or networking environments, e.g. to understand the behavior of a new application in a real context before the official release (of both applications and network scenarios).

- Testing of different transport protocols carrying realistic traffic loads and under realistic background traffic, e.g. to understand the protocol best suited to certain application scenarios.

- Testing of Quality of Service, Traffic Engineering, and Routing architectures.

- Measurement of the performance of network architectures, devices or applications in terms of throughput, jitter, losses and delay using different traffic patterns, e.g. active measurements of link delay using Poisson probing.

- Root causes analysis over BANs
Configure the STG-10G in 4 Easy Steps
Configure the Generator

[Image of a software interface for configuring the Generator, showing tabs for Configuration, Generator, and Receiver, and settings for Experiment #1 with options to start and stop the experiment.]

Info: Be sure receiver is running before starting traffic generator.
Modify the Flow
View Available Endpoints

![Endpoint Configuration Table]

### Configuration

<table>
<thead>
<tr>
<th>Local Box Name</th>
<th>IP Address:Port number</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Box #0</td>
<td>192.168.1.100:9000</td>
<td>up</td>
</tr>
<tr>
<td></td>
<td>192.168.1.101:9001</td>
<td>up</td>
</tr>
</tbody>
</table>

**Remote Box Name**

<table>
<thead>
<tr>
<th>Remote Box Name</th>
<th>IP Address:Port Number</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Box #1</td>
<td>78.78.78.78:9000</td>
<td>✔️</td>
</tr>
<tr>
<td></td>
<td>78.78.78.78:9001</td>
<td>✗️</td>
</tr>
<tr>
<td></td>
<td>78.78.78.78:9002</td>
<td>✗️</td>
</tr>
<tr>
<td>Box #2</td>
<td>88.88.88.88:9001</td>
<td>✔️</td>
</tr>
<tr>
<td></td>
<td>88.88.88.88:9002</td>
<td>✗️</td>
</tr>
<tr>
<td>Box #3</td>
<td>98.98.98.98:90876</td>
<td>✔️</td>
</tr>
</tbody>
</table>

[View Available Endpoints Image]
View Real Time Test Results

Live results - Experiment #1

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow 1</td>
<td>Current: 1, Min: 1, Avg: 2, Max: 3</td>
<td>Current: 4, Min: 4, Avg: 5, Max: 6</td>
<td>Current: 7, Min: 7, Avg: 8, Max: 9</td>
<td>Current: 10, Min: 10, Avg: 11, Max: 12, Avg Loss Burst: 0</td>
</tr>
<tr>
<td>Flow 2</td>
<td>Current: 1, Min: 1, Avg: 2, Max: 3</td>
<td>Current: 4, Min: 4, Avg: 5, Max: 6</td>
<td>Current: 7, Min: 7, Avg: 8, Max: 9</td>
<td>Current: 10, Min: 10, Avg: 11, Max: 12, Avg Loss Burst: 0</td>
</tr>
<tr>
<td>Flow 3</td>
<td>Current: 1, Min: 1, Avg: 2, Max: 3</td>
<td>Current: 4, Min: 4, Avg: 5, Max: 6</td>
<td>Current: 7, Min: 7, Avg: 8, Max: 9</td>
<td>Current: 10, Min: 10, Avg: 11, Max: 12, Avg Loss Burst: 0</td>
</tr>
<tr>
<td>Aggregate</td>
<td>Current: 1, Min: 1, Avg: 2, Max: 3</td>
<td>Current: 4, Min: 4, Avg: 5, Max: 6</td>
<td>Current: 7, Min: 7, Avg: 8, Max: 9</td>
<td>Current: 10, Min: 10, Avg: 11, Max: 12, Avg Loss Burst: 0</td>
</tr>
</tbody>
</table>

Graphs showing trends for Delay, Jitter, Bitrate, and Packet-loss.
View Fixed End Test Results

![Image of test results dashboard]

### Results per flows - Experiment #1

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min</td>
<td>Avg</td>
<td>Max</td>
<td>Min</td>
</tr>
<tr>
<td>Flow 1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Flow 2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Flow 3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Aggregate</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

### Total Results - Experiment #1

<table>
<thead>
<tr>
<th></th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of flows</td>
<td>3</td>
</tr>
<tr>
<td>Total time [s]</td>
<td>61.311</td>
</tr>
<tr>
<td>Total packet</td>
<td>2919</td>
</tr>
<tr>
<td>Packets dropped</td>
<td>0 (0.00 %)</td>
</tr>
<tr>
<td>Minimum delay [ms]</td>
<td>0.813</td>
</tr>
<tr>
<td>Average delay [ms]</td>
<td>0.015</td>
</tr>
<tr>
<td>Maximum delay [ms]</td>
<td>0.047</td>
</tr>
<tr>
<td>Delay standard deviation [ms]</td>
<td>1.294</td>
</tr>
<tr>
<td>Average jitter [ms]</td>
<td>0.497</td>
</tr>
<tr>
<td>Average bit rate [Kbit/s]</td>
<td>23.828</td>
</tr>
<tr>
<td>Average packet rate [pkt/s]</td>
<td>45.978</td>
</tr>
<tr>
<td>Average loss-burst size [pkt]</td>
<td>0</td>
</tr>
</tbody>
</table>
Settings for Time Zone, Box ID, Upgrades