ITRINEGY NETWORK EMULATOR

INE Enterprise/Ultra Technical Specifications

General

	INE Enterprise				INE Ultra		Custom
	Model-2	**Model-4**	**Model-6**	**Model-8**	**Model-2**	**Model-4**	**Model-S**
Part Number	INE-ENTP-2	INE-ENTP-4	INE-ENTP-6	INE-ENTP-8	INE-ULTR-2	INE-ULTR-4	INE-ENTP-S INE-ENTU-S
Emulation Ports – RJ45 Copper	2	4	6	8	2	2	Custom
Emulation Ports – 10G Fiber Optic	-	-	-	-	2	4	Custom
Storage Capacity	240GB	240GB	240GB	240GB	240GB	240GB	Custom
Dimensions	1U	1U	1U	1U	1U	1U	1U – 4U
Max Packet Rate Per Second[2]	44 Million	44 Million	44 Million	44 Million	44 Million	44 Million	Custom

The following items and features are common across the INE Network Emulator family regardless of model.

Certification

- FCC / UL / CE / EN Compliant

Management & Configuration

- 1 x Management Port (RJ45 Copper)
- Web Management GUI for centralised configuration, control and monitoring of system resources
- Unlimited number of tenants/users (No limit is set on the number of devices/systems in the Emulated Network)
- API for automated and remote control of the INE
- Background emulation enables Emulator to start with one or more default emulations (impaired or unimpaired) so that it can be used in DevOps and Agile environments
- Save and share emulation scenarios centrally from the emulator
- Save emulation scenarios locally

- Control emulator resources (ports/emulations/network objects) by tenant/user
- User port security (physical and virtual ports)
- Powerful Wireshark like expression tool for packet classification, filtering and routing
- Built-in Traffic Generation
- Centralised Licensing and patch Management
- INE Drawing GUI for Windows, Mac and Linux
- Import your own network diagrams, backgrounds and icons into INE Drawing GUI
- CLI (Command Line Interface)

Ports (Virtual Ports)

	INE Enterprise				**INE Ultra**		**INE Custom**
	Model-2	**Model-4**	**Model-6**	**Model-8**	**Model-2**	**Model-4**	**Model-S**
Maximum number of Virtual Emulation Ports	253	251	249	247	251	249	Custom

- Define partitions of physical (or other virtual) ports by:
    - VLAN (layer 2) with tagging/detagging/retagging
    - IPv4 (layer 3) with ICMP/ARP/DHCP Relay support
    - IP (layer 3) with IPv4 and IPv6 support
    - Filter (layer-less) by ranges of IP src/dest, Port (application), VLANs or “Wireshark like” expressions

Traffic Generation

- Define virtual ports that generate TCP or UDP Traffic
- Specify src/dst IP addresses or ports, VLANs, mac addresses and many other fields
- Define packet size and data rate (in bps or pps)
- Use these generating ports to make internal links (circuit objects) busy
- Generate traffic that may exit the emulation to make real networks busy

Network Emulation

- 10 nanosecond timing accuracy
- Create Any Network Type (Bridged, Routed, Meshed, Star Coupled; Point to Point; Multi-Point to Point; Hybrid)
- Connect Port to Port in any configuration: Physical to Physical, Physical to Virtual, Virtual to Virtual; etc
- Sophisticated emulations available at Layer 2 and Layer 3

- Sophisticated routing including route by layer 2 and layer 3 properties e.g. IP/Subnet, IP Range, IP src/dest, Port (application) – both symmetric and asymmetric, and using "Wireshark like" classification "route" at layers 4-7
- Network Address Translation (NAT) and Port Address Translation (PAT)
- Unlimited Number of Impairments[4]
- Wireshark Packet Capture from any internal "object" (end-point, router, circuit, cloud)
- Jumbo Frame Support (up to 10 KB)
- DHCP Relay Support, multiple DHCP relays possible
- Can be used in-line, as a bridge, bridging VLANs, as a router or routed on one port (router-on-a-stick)

Real-Time Emulation Updates

- Manual (via GUI)
- Automated including looping (via GUI)
- CLI/API (Scripted)

Advanced Features

- Advanced "Wireshark like" expression matching with Just-in-time (JIT) compiler for speed
    - Ability to easily add your own protocol definitions to enhance this
- Add custom impairments to emulator kernel via easy to use python like language with examples.[5]

Bandwidth

	INE Enterprise				INE Ultra		INE Custom
	Model-2	Model-4	Model-6	Model-8	Model-2	Model-4	Model-S
Bandwidth (1bps increments)	Up to 1Gbps	Up to 1Gbps	Up to 1Gbps	Up to 1Gbps	Up to 10Gbps	Up to 10Gbps	Up to 10Gbps

- Predefined Network Types (e.g. Ethernet 10Mbps; 100Mbps; 1Gbps; E1; E3; T1; T3)
- Static Congestion
- Variable Congestion
- QoS Bandwidth (E.g. separate bandwidths for traffic types within a link)
- Cloud Bandwidth control by src/dest address, port or VLAN allowing single Cloud Object to represent a core network e.g. The Internet
- Specify emulated layer 2 header size to properly control bandwidth for non-Ethernet circuits in Bytes
- Bandwidth Queue Size in Bytes
- Symmetric and Asymmetric for circuit objects
- Limit bandwidth in router objects e.g. represent wireless cloud with maximum bandwidth capacity

Latency & Jitter

- 1 nanosecond – 10 mins[2] precision for all functions
- Fixed Delay
- Random Delay

- Gaussian Delay (Normal Distribution) with mean, standard deviation, minimum cut off (optional), maximum cut off (optional)
- Predefined delay scenarios (e.g. common intercity / international / intercontinental / satellite configurations)
- Step Delay per packet
- Step Delay per time period (Wave)
- Inter Packet Gap (enforces a minimum packet gap between random values using times to create jitter in very high-speed streams)
- Delay Sequence (specify table of base delay, jitter and durations with looping)
- Delay with Jitter (Packet Delay Variation - PDV)

Loss

- 1 in x Packets
- Random (% packets to drop; 0% - 100% in 0.00000001% increments)
- Random with Burst (Burst from 1 packet to 1 billion)
- Poisson Drop (Supply Lambda mean interval between events and a drop window duration)
- Total Drop / No Drop (for use in very speed/efficient scripting)
- Drop Packet Based on Bit Error (BER)

Bit Error (BER) / Packet Corruption

- 1 in x Errors (BER)
- 1 in x Errors (BER Random variation in x)
- 1 in x with bursts (BER random or fixed, timed, to new 1 in y rate)
- Random Packet Error (0% - 100% in 0.00000001% increments)
- Poisson Error (Supply Lambda mean interval between events and an error window duration)

Cloud Object

- A single Cloud Network Object can behave like a complete network by specifying a table or bandwidth, latency, jitter, loss and cost (TTL) values, allows easy definition of MPLS core networks, portions of the Internet etc.

QoS (Quality of Service / Class of Service and Traffic Shaping)

- Specify traffic classes based on IP src/dest, Port (application), DSCP, VLAN, and apply individuals QoS bandwidth and QoS priority. “Wireshark like” Expression matching extends this to allow very sophisticated and user defined class definitions
- Specify Bucket Class Algorithms supporting CIR, Bc, Be, Tc and traffic shaping similar to those used by CISCO, Juniper etc. Packet Classification is by IP src/dst, port, VLAN or “Wireshark like” expression.

Modification / Corruption

- Random Packet Corrupt (Packet chosen randomly between 0% - 100% in 0.00000001% increments) Unlimited number of corruptions applied to particular bytes. Corruptions can be XOR/OR/AND/ADD/SUBTRACT/OVERWRITE

Duplication

- Packet Duplication (choose packet at random 0% - 100% in 0.00000001% increments)
- Packet Duplication and Move Duplicate out of sequence

MTU and Fragmentation

- Fragment Packets if MTU is exceeded. Optionally obey or disobey don’t fragment flag (MTU from 64 bytes to 10,000 bytes, jumbo frames)

Out of Order

- Random Packet Move with Offset (0% - 100% in 0.00000001% increments) Offset 1 to 1 million packets

Reporting

- Graphs and Export to CSV
- Bit Rate (sent/received)
- Packet Rate (sent/received)
- Queue Sizes
- Average Delay
- Loss/Drops
- Corruptions
- Moves
- Queue Size
- Queue Full Packets Dropped
- Average Packet Size
- Bit Rate / Packet Rate Per Class for Cloud Object
- QoS

Filters

- Specify what traffic will be impaired, passed through unimpaired or dropped by src/dest IP, Port (Application), or/and VLAN. Or use “Wireshark like” expression matching to create sophisticated filters or user defined filters

Routing

- In addition to conventional routing (layer 3 by destination IP address) alternatively route by src and/or dest IP, Port (Application), or/and VLAN, or use “Wireshark like” expression matching route by any reasonable packet property up to layer 7.

References

1 – INE Model-S is a custom configuration available on request for specific scenarios

2 – 4 million packets per core; Model-S allows for more cores etc; other impairments may limit;

3 – Feature planned for future release

4 – Depending on traffic speed and available memory. Model-S allows for custom memory.

5 – Optional feature planned for Q1 2016

Ref: INETecComp171123

Copyright © iTrinegy Ltd 2017