QualNet®
Building Smarter Networks

QualNet 4.5
Developer Model Library

March 2008

Scalable Network Technologies, Inc.

6701 Center Drive West, Suite 520
Los Angeles, CA 90045

Phone: 310-338-3318
Fax: 310-338-7213

http://www.scalable-networks.com
http://www.qualnet.com

Scalable Network Technologies, Inc.
6701 Center Drive West, Suite 520
Los Angeles, CA 90045
Phone: 310-338-3318
Fax: 310-338-7213
http://www.scalable-networks.com
http://www.qualnet.com

Table of Contents

802.3 LAN/Ethernet

The IEEE 802.3 LAN model models a single bus LAN of either 10 Mbps/1G (Gigabit Ethernet) or 100 Mbps/10G (Gigabit Fast Ethernet) hosts. QualNet supports both half duplex and full duplex modes. Full Duplex transmission is a mode of data transfer between two stations in which data can flow in both directions at the same time. Full-Duplex Ethernet bypasses the normal CSMA/CD protocol to allow two stations to communicate over a point-to-point link. It effectively doubles the transfer rate by allowing each station to concurrently transmit and receive separate data streams. For example, a 10 Mbps full-duplex Ethernet station can transmit one 10 Mbps stream at the same time it receives a separate 10 Mbps stream. This provides an overall data transfer rate of 20 Mbps.

Command Line Configuration

To select IEEE 802.3 Ethernet as the MAC protocol, place the following entry in the scenario configuration (.config) file:

```
MAC-PROTOCOL        MAC802.3
```

802.3 LAN configuration parameters are described in Table 1.

TABLE 1. 802.3 LAN Parameters

Parameter	Description
SUBNET-PROPAGATION-DELAY <delay> or, LINK-PROPAGATION-DELAY <delay>	This parameter specifies propagation delay. This parameter is mandatory.
SUBNET-DATA-RATE <value> or, LINK-BANDWIDTH <value>	This parameter specifies subnet data rate or link bandwidth. This parameter is mandatory.
MAC802.3-MODE [HALF-DUPLEX | FULL-DUPLEX]	This parameter specifies 802.3 mode. This parameter is optional. The default value is Half-Duplex.

Half Duplex Command Configuration

Configuration of Half Duplex in subnet can be done as following:

1. Configure the subnet.
   ```
   SUBNET subnet-address {node1, node2}
   ```
2. Set the subnet MAC protocol to 802.3.
   ```
   MAC-PROTOCOL                    MAC802.3
   ```
3. Set the MAC protocol mode to Half duplex
   ```
   MAC802.3-MODE    HALF-DUPLEX
   ```
4. Set the data rate, 802.3 support 10Mbps, 100Mbps, 1Gbps, 10Gbps
   ```
   SUBNET-DATA-RATE     10000000 | 100000000 | 1000000000 | 10000000000
   ```
5. Set the propagation delay of the subnet

The propagation delay should be smaller than half of the Ethernet slot time, in order to let nodes correctly detect collisions. The Ethernet slot time is corresponding to its data rate. In QualNet, the slot time of 1000M Ethernet is 4096 nano-seconds. Thus the propagation delay should be smaller than 2048 nano-seconds.

```
SUBNET-PROPAGATION-DELAY    <time value>
```

Example:

```
SUBNET N2-192.0.2.0 {1, 2}
[N2-192.0.2.0] SUBNET-DATA-RATE          1000000000
[N2-192.0.2.0] SUBNET-PROPAGATION-DELAY 2US
[N2-192.0.2.0] MAC-PROTOCOL              MAC802.3
[N2-192.0.2.0] MAC-802.3-MODE            HALF-DUPLEX
```

Full Duplex Command Configuration:

Configuration of Full Duplex can be done by two ways:

- By using SUBNET
- By using LINK

To configure Full Duplex using SUBNET, follow these steps:

1. Configure the subnet
   ```
   SUBNET subnet-address {node1, node2}
   ```
2. Set the subnet MAC protocol to 802.3
   ```
   [subnet-address] MAC-PROTOCOL   MAC802.3
   ```
3. Set the MAC protocol mode to full duplex
   ```
   [subnet-address] MAC-802.3-MODE    FULL-DUPLEX
   ```
4. Set the data rate, 802.3 support 10Mbps, 100Mbps, 1Gbps, 10Gbps
   ```
   [subnet-address] SUBNET-DATA-RATE  10000000 | 100000000 | 1000000000 |
   10000000000
   ```
5. Set the propagation delay of the subnet
   ```
   [subnet-address] SUBNET-PROPAGATION-DELAY  <time value>
   ```

Example:

```
SUBNET N2-192.0.2.0 {1, 2}
[N2-192.0.2.0] SUBNET-DATA-RATE          1000000000
[N2-192.0.2.0] SUBNET-PROPAGATION-DELAY  1US
[N2-192.0.2.0] MAC-PROTOCOL              MAC802.3
[N2-192.0.2.0] MAC-802.3-MODE            FULL-DUPLEX
```

To configure Full Duplex using LINK, follow these steps:

1. Configure the link
   ```
   LINK link-address {node1, node2}
   ```
2. Select the MAC protocol for the
   ```
   [link-address] LINK-MAC-PROTOCOL   MAC802.3 | ABSTRACT
   ```
3. Set the duplex mode of 802.3
   ```
   [link-address] MAC-802.3-MODE   FULL-DUPLEX
   ```

4. Set the data rate of the link

```
[link-address] LINK-BANDWIDTH  10000000 | 100000000 | 1000000000 | 10000000000
```

5. Set the propagation delay

```
[link-address] LINK-PROPAGATION-DELAY  <time value>
```

Example:

```
LINK N2-192.0.2.0 {1, 2}
[N2-192.0.2.0] LINK-MAC-PROTOCOL        MAC802.3
[N2-192.0.2.0] LINK-BANDWIDTH           1000000000
[N2-192.0.2.0] LINK-PROPAGATION-DELAY   1US
[N2-192.0.2.0] MAC-802.3-MODE           FULL-DUPLEX
```

GUI Configuration

Enabling 802.3 LAN

To enable 802.3 LAN, go to **Connections > Point to point link [x] -> [y] [Address] > Point to point link Properties > Mac Protocol**. From the configurable property window, set **MAC Protocol** to **MAC802.3**, as shown in Figure 1.

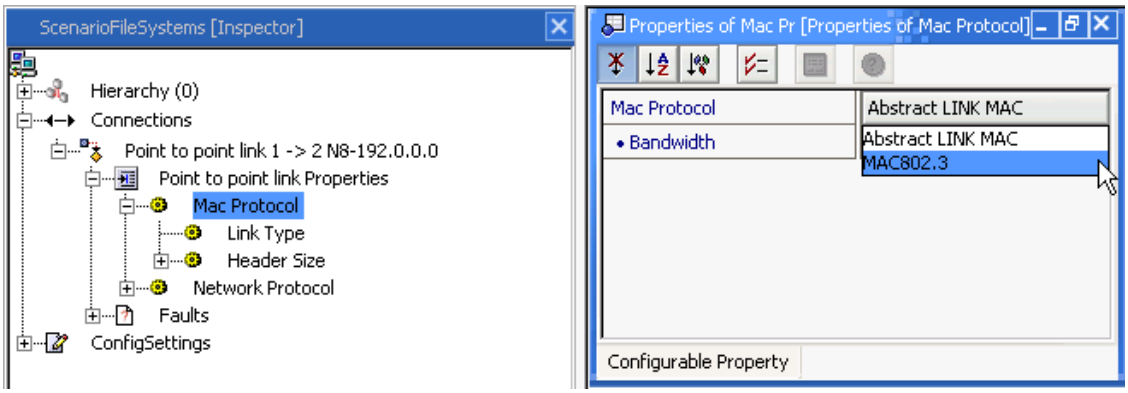


FIGURE 1. Configuring 802.3 LAN as the MAC Protocol

Configuring 802.3 LAN Parameters

When 802.3 LAN is configured, a set of its parameters is displayed. Configure the 802.3 LAN parameters as shown in Figure 2.

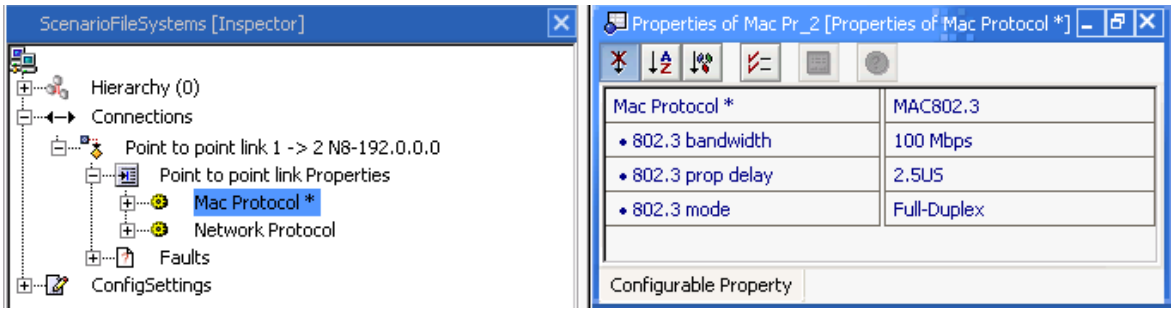


FIGURE 2. 802.3 LAN Configuration Parameters

The 802.3 LAN model requires the following three parameters:

```
SUBNET-DATA-RATE 10000000 | 100000000 | 1000000000 | 10000000000
SUBNET-PROPAGATION-DELAY 1US
MAC-802.3-MODE FULL-DUPLEX | HALF-DUPLEX
```

Note: In full-duplex mode, users must guarantee there are only 2 nodes in the subnet.

The first two parameters specify the data rate and propagation delay for the model, respectively. The third parameter specifies the mode for the 802.3 MAC. The 802.3 Ethernet model requires that these parameters be specified. When specifying 802.3, only 10Mbps (10BaseT) or 100Mbps (100BaseT) data rates are currently allowed. For 10BaseT, the propagation delay must be less than half of 51.2 micro-seconds for collision detection to work correctly. Likewise, for 100BaseT, the propagation delay must be less than half of 5.12 microseconds. For 1GBaseT, the propagation delay must be less than half of 4.096 microseconds.

Enabling Statistics Collection

To enable statistics collection, go to **ConfigSettings > Statistics > Statistics**. Enable **MAC statistics** as shown in Figure 3.

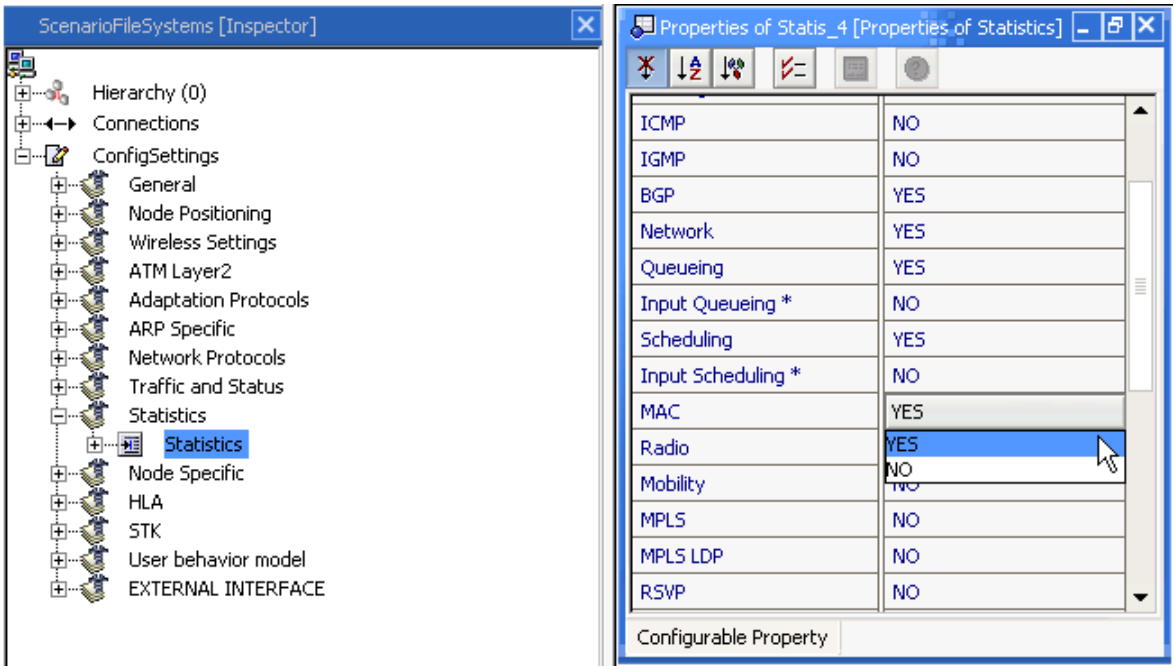


FIGURE 3. Enabling MAC Statistics

Statistics

Table 2 lists the 802.3 LAN statistics.

TABLE 2. 802.3 LAN Statistics

Statistic	Description
Half-Duplex	
Number of Frames Transmitted	Total number of frames transmitted in half duplex.
Number of Frames Received	Total number of frames received in half duplex.
Number of Retransmissions	Total number of retransmissions in half duplex.
Number of Frames Dropped	Total number of frames dropped in half duplex.
Full-Duplex	
Number of Frames Transmitted in Full Duplex	Total number of frames transmitted in full duplex.
Number of Frames Received in Full Duplex	Total number of frames received in full duplex.
Number of Frames Dropped in Full Duplex	Total number of frames dropped in full duplex.
Number of Bytes Transmitted in Full Duplex	Total number of bytes transmitted in full duplex.
Number of Bytes Received in Full Duplex	Total number of bytes received in full duplex.
Number of Bytes Dropped in Full Duplex	Total number of bytes dropped in full duplex.
Channel Utilization by Full Duplex	Calculates channel utilization by full duplex.

Abstract Satellite Model

This is an abstract model of a satellite network. Each satellite network is grouped into subnets. Each satellite subnet has exactly one satellite node and many ground nodes. The ground nodes associated with a subnet always transmit to the designated subnet satellite node. Thus, no handoffs are involved. Also, satellite nodes are bent-pipe satellites (relay data only). When the satellite node receives data from the ground nodes, it broadcasts the data to all other ground nodes in the subnet, but not to the ground node originating the data. Finally, the satellite node must not be generating any packets. Thus, the satellite node cannot run an application or routing protocol.

Note: This model represents the highest level of abstraction of a satellite model considering of a lossless linear delay process. Developers requiring more detailed simulation of satellite systems should consider using the advanced satellite addon module described in *QualNet 4.5 Satellite Model Library*.

The queuing performance of the system is entirely determined by the network layer processes. Packet loss due to data transmission errors is not modeled. Therefore, the only parameter that needs to be calculated is packet latency which is given by the following equation:

$$t_k = t_{prop} + \frac{l_k}{r}$$

where t_k is the transmission delay of frame k, t_{prop} is the configured propagation delay, l_k is the length of frame k, and r is the configured bandwidth of the satellite communication network.

Command Line Configuration

To enable the Abstract satellite model, include the following parameter in the scenario configuration (.config) file:

```
MAC-PROTOCOL        SATCOM
```

Table 3 shows the Abstract satellite configuration parameters.

TABLE 3. Abstract Satellite Model Parameters

Parameter	Description
SATCOM-SATELLITE-NODE <node ID>	Specifies which node is the satellite node. This parameter is mandatory.
SATCOM-TYPE BENT-PIPE	Specifies the SATCOM type. Currently, only the bent-pipe satellite is supported. This parameter is mandatory.
SATCOM-BANDWIDTH <bandwidth>	Specifies the satellite link bandwidth capacity.
SATCOM-PROPAGATION-DELAY <delay>	Specifies the ground-to-ground propagation delay.

Statistics

The statistics collected by the Abstract Satellite model are listed in Table 4.

TABLE 4. Abstract Satellite Statistics

Statistic	Description
Frames sent	Number of frames sent by the SATCOM MAC protocol
Frames received	Number of frames received by the SATCOM MAC protocol
Frames relayed	Number of frames received by the SATCOM MAC protocol and treated as retransmissions for bent-pipe simulation

Sample Scenario

Scenario Description

The sample scenario (see Figure 4) consists of a bent-pipe satellite system containing a satellite (node 4) and three ground stations (nodes 1 to 3).

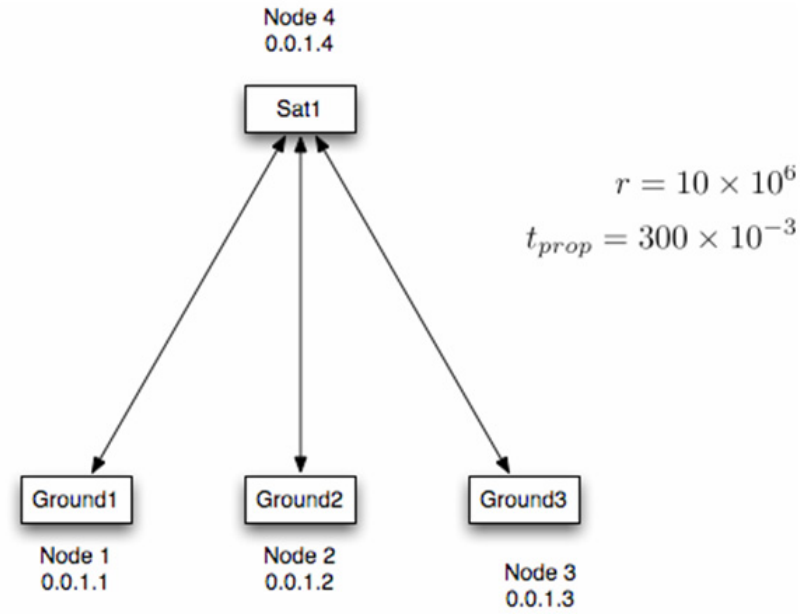


FIGURE 4. Sample SATCOM Scenario

Command Line Configuration

To configure the sample scenario, include the following lines in the scenario configuration (.config) file:

```
SUBNET N8-1.0 { 1 thru 4 }
[N8-1.0] MAC-PROTOCOL SATCOM
[N8-1.0] SATCOM-SATELLITE-NODE 4
[N8-1.0] SATCOM-TYPE BENT-PIPE
SATCOM-BANDWIDTH 100000000
SATCOM-PROPAGATION-DELAY 300MS
```

Abstract Transmission Control Protocol (Abstract TCP)

Abstract TCP is based on TCP Reno (see the TCP section of this model library). However, it simplifies and omits some features in order to improve the runtime performance of the TCP model. Because of this, it has slightly lower fidelity as a tradeoff for simulation speed.

The Abstract TCP model omits the following features: three-way hand shaking (open/close), keep alive, persistence time, variable packet size, actual data caching, and TCP options.

Command Line Configuration

To select Abstract TCP, configure as follows:

```
TCP     ABSTRACT-TCP
```

Table 5 shows the configuration parameters for Abstract TCP.

TABLE 5. Abstract TCP Parameters

Parameter	Description
TCP-DELAY-ACKS [YES | NO]	Specifies whether ACKs for data segment packets are delayed for a certain period of time to reduce the number of ACKs transmitted. If this parameter is set to NO, ACKs are sent immediately. The default value is YES.
TCP-MSS <value>	Specifies the maximum segment size (in bytes). The default value is 512.
TCP-RANDOM-DROP-PERCENT <value>	Specifies the initial random drop percent for the node. The default value is 0.0
TCP-RECEIVE-BUFFER 16384	Specifies the receive buffer size (in bytes). The receive buffer size provide an upper bound on the advertised window. **Note**: The TCP sender cannot send more data than the available space of the receive buffer at the receiver. The default value is16384.
TCP-SEND-BUFFER 16384	Specifies the send buffer size (in bytes). The send buffer size provides an upper bound on the advertised window. **Note**: The TCP sender cannot send more data than what is stored in the sender buffer. The default value is16384.
TCP-STATISTICS [YES | NO]	This parameter enables or disables TCP statistics collection. Default value is NO.
TCP-TRACE TCPDUMP-ASCII	This parameter enables TCP dump. Abstract TCP supports TCP dump in TCP Dump (ASCII) format only. The name of the dump file is tcptrace_abs.asc.

TABLE 5. Abstract TCP Parameters (Continued)

Parameter	Description
TCP-TRACE-DIRECTION [OUTPUT | INPUT | BOTH]	Specifies the type(s) of packets that are included in the dump file. INPUT : Only packets received from the network are recorded in the dump file. OUTPUT :Only packets sent to the network are recorded in the dump file. BOTH : Both sent and received packets are recorded in the dump file. The default value is BOTH.
TCP-VERIFICATION-DROP-PERCENT <value>	Specifies the drop count. This option is for use with TCP verification scenarios that test variant behavior with 0 to 4 drops. The range is 0 to 4. The default value is 0. **Note**: This parameter is for testing purposes. It is recommended that the default value be used for this parameter.

GUI Configuration

Enabling Abstract TCP

To enable Abstract TCP, go to **Hierarchy (x) > Nodes > host[x] > Node configurations > Transport Layer > TCP Variant**. From the configurable property window, set TCP Variant to **Abstract TCP** as shown in Figure 5.

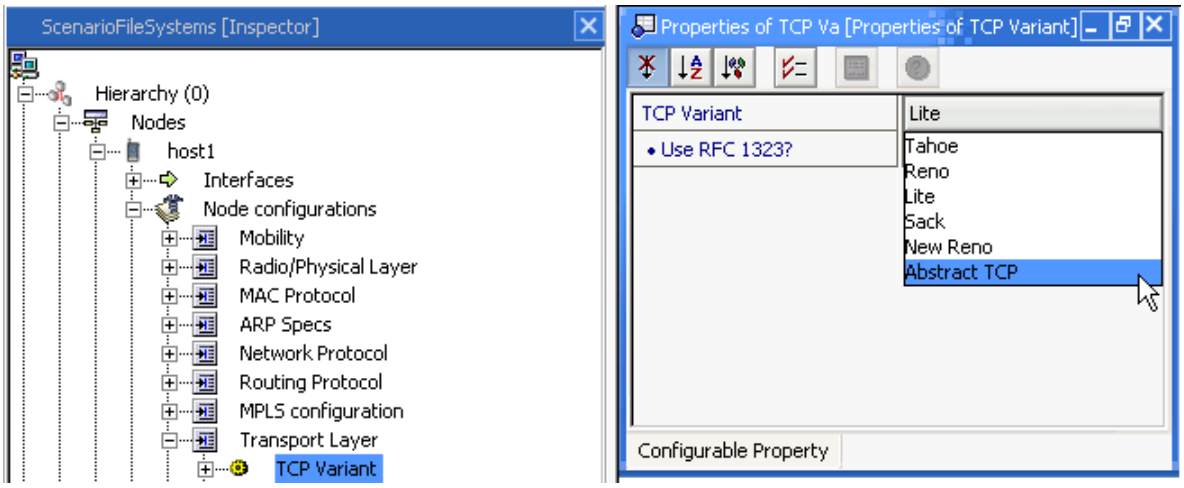


FIGURE 5. Enabling Abstract TCP

Statistics

Table 6 shows the statistics collected by Abstract TCP.

TABLE 6. Abstract TCP Statistics

Statistic	Description
ACK-only Packets Sent	Total number of pure ACK packets sent to network layer from transport layer.
Data Packets Fast Retransmitted	Total number of packets which are fast retransmitted.
Data Packets in Sequence	Total number in sequence data packets sent.
Data Packets Received	Total number of pure data packets received at transport layer.
Data Packets Retransmitted	Total number of data packets retransmitted.
Data Packets Sent	Total number of pure data packets sent to network layer from transport layer.
Duplicate ACK Packets Received	Total number of in duplicated ACK packets received.
In Sequence ACK Packets Received	Total number of in sequence ACK packets received by the server.
Packets Received that are Too Short	Total number of short packets received with packet size smaller than TCP/IP header.
Packets Sent to Network Layer	Total number of packets sent to network layer from transport layer. It is the sum of data packets and control packets and pure ACK packets. But Abstract TCP has no control packets.
Total Packets Received From Network Layer	Total number of packets received from network layer to transport layer. It is the sum of data packets and control packets and pure ACK packets. But Abstract TCP has no control packets.

Address Resolution Protocol (ARP)

ARP is a basic protocol used in almost every TCP/IP implementation. When a packet is sent from one host on a LAN to another, the device driver software does not look at the destination Network layer Protocol Address in the packet; rather, it is the MAC layer Hardware Address that determines for which interface the frame is destined. The purpose of this protocol is to translate the logical 32 bit IP address to the corresponding physical address also known as Ethernet addresses. ARP resides between the Network layer and the MAC layer in the Data link Layer. It presents a method of converting Network layer Protocol Addresses (such as IP addresses) to MAC layer Hardware Addresses (such as Ethernet addresses). ARP maintains the mapping between the IP address and the MAC address in a table called ARP cache that is essential for the efficient operation of the ARP. The entries in this table are dynamically added and removed. The default life time of dynamic entry is 20 minutes from the time it was created.

ARP is a general protocol, which can be used in any type of broadcast network. The fields in the ARP packet specify the type of the MAC address and the type of the protocol address. ARP is used with most IEEE 802.x LAN media. In particular, it is used with FDDI, Token Ring, and Fast Ethernet, in precisely the same way as it is with Ethernet.

Command Line Configuration

To configure ARP at the subnet level, use the following parameters in the config file.

```
[network address] ARP-ENABLED    YES | NO
```

ARP takes several parameters for configuration. The details and description of those parameters are given in Table 7.

TABLE 7. ARP Parameters

Parameter	Description
[<node-id> | <network address>] ARP-ENABLED	This parameter enables ARP. The default value is NO. Note: This is a mandatory parameter to enable ARP.
[<node-id> | <network address>] ARP-BUFFER-SIZE 1	This parameter specifies ARP buffer size. The default value is 1.
[<node-id> | <network address>] ARP-STATIC-CACHE-FILE ./default.arp-static	This parameter specifies the location of ARP-STATIC-CACHE FILE.
[<node-id> | <network address>] MAC-ADDRESS-CONFIG-FILE ./default.mac-address	This parameter specifies the location of MAC-ADDRESS-CONFIG FILE.
[<node-id> | <network address>] ARP-CACHE-EXPIRE-INTERVAL 20M	This parameter specifies timeout time of ARP-CACHE table. The default value is 20 minutes.
[<node-id> | <network address>] ARP-STATISTICS YES | NO	This parameter enables statistical collection of ARP. The default value is NO.

Specifying MAC Address using MAC-address-config File

You can configure the MAC address using the .mac-address file, specifying it in the .config-file:

```
MAC-ADDRESS-CONFIG-FILE    ./default.mac-address
```

Syntax for the ./default.mac-address:

```
<node id><interface-index>[<hardware-address-length>][<hardware type>]<mac
address>
```

Where:

<node id> <interface-index> is the identifier of the node.

<hardware-address-length> is the length of the hardware address.

<hardware-type> is the type of hardware used in the network. The default hardware type is ETHERNET.

<mac-address> is the hardware address and it must be specified in hex format.

For example:

```
# Mac Address Configuration File
1 0 6 ETHERNET 23:4f:5C:aa:FE:B2
2 1 6 ETHERNET 5C-AA-FE-23-4F-C2
4 0 ETHERNET 5C-BA-FE-23-4F-F2
6 0 00:4f:5C:ab:B5:C7
7 0 00:4f:5C:ab:B5:C7
```

Configuring the MAC Address at the Interface Level using the Config File

You can configure the MAC address at the interface level in .config file using the following parameters:

```
[<Interface address>] MAC-ADDRESS-TYPE <mac-address-type-string>
[<Interface address>] MAC-ADDRESS <mac-address-string>
<[Interface address>] MAC-ADDRESS-LENGTH <mac-address-length>
```

For example:

```
[192.168.100.1] MAC-ADDRESS-TYPE ETHERNET
[192.168.100.1] MAC-ADDRESS cc-cc-2e-1f-3e-dd-3e
[192.168.100.1] MAC-ADDRESS-LENGTH 6
```

Specifying ARP Static Cache Table

You can configure the ARP static cache table using the following parameter in the config file:

```
[node-id | network address] ARP-STATIC-CACHE-FILE     ./static-cache-file
```

Syntax for the default ARP cache file:

```
<node id> [interface-index] <protocol type> <protocol address> | <node-
interface> <hardware type> <hardware-address-length><hardware
address><timeout>
```

Where:

```
<node id> [interface-index] is the node where the static ARP entry is created.
<protocol type> <protocol address> | <node-interface> is the IP address of the
node whose static entry has to be inserted into the ARP cache table.
<hardware type> is the type of hardware used in the network. It can be either
ETHERNET or ATM
<hardware-address-length> is the length of the hardware-address of the node.
<hardware-address> is the 6-byte hardware address of the host whose static
entry has been inserted.
<timeout> For static ARP cache table, the time out will be 0 due to forever
existence of the entry in the table
```

For example:

```
ARP-STATIC-CACHE-FILE     ./default.arp-static

# default.arp-static
1 0 IP 3-0 ETHERNET 6 00-00-00-00-03-00 0
2 IP 2-0 ETHERNET 6 00-00-00-00-02-00 0
3 0 IP 192.0.0.1 ETHERNET 6 00-00-00-00-01-00 0
3 1 IP 192.0.1.1 ETHERNET 6 00-00-00-01-01-00 0
```

Specifying the ARP Interface Fault

ARP is enhanced to deal situations such as modifications in the network interface card. You can use the QualNet Interface fault configuration to deal such situations. To specify the interface fault, refer to the fault section of this Developer Model Library document.

GUI Configuration

Go to **Hierarchy (x) > Nodes > Subnet Nx-x.x.x.x > Subnet Properties > ARP Enabled**. From the configurable property window, set **ARP Enabled** to **Yes** as shown in Figure 6.

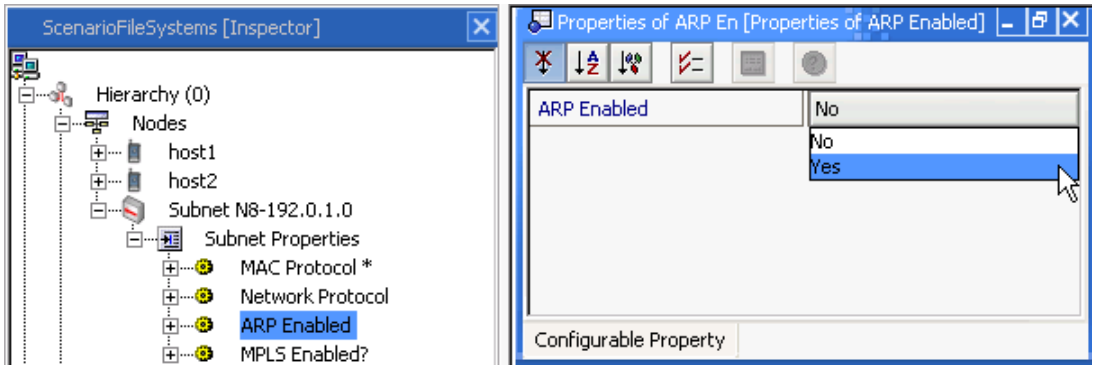


FIGURE 6. Enabling ARP at the Subnet Level

Configuring General ARP Parameters

Once ARP is enabled, a set of its parameters are displayed. Configure ARP parameters as shown in Figure 7.

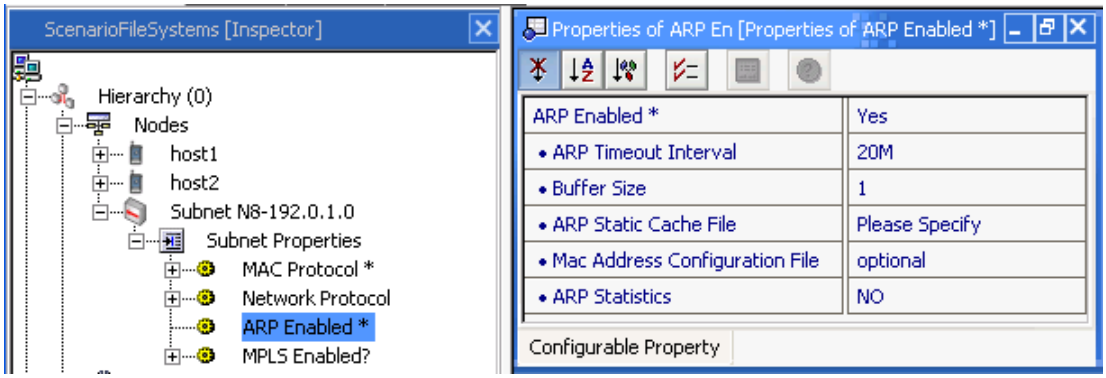


FIGURE 7. Configuring General ARP Parameters

Specifying the MAC Address

Go to **Hierarchy (x) > Nodes >Host [x] > Interfaces > interface [x] > Interface configurations > MAC Protocol > MAC Address Style**. In the Configurable Property window, set **MAC Address Style** to **Predefined** and **MAC Address Type** to **Ethernet**, as shown in Figure 8.

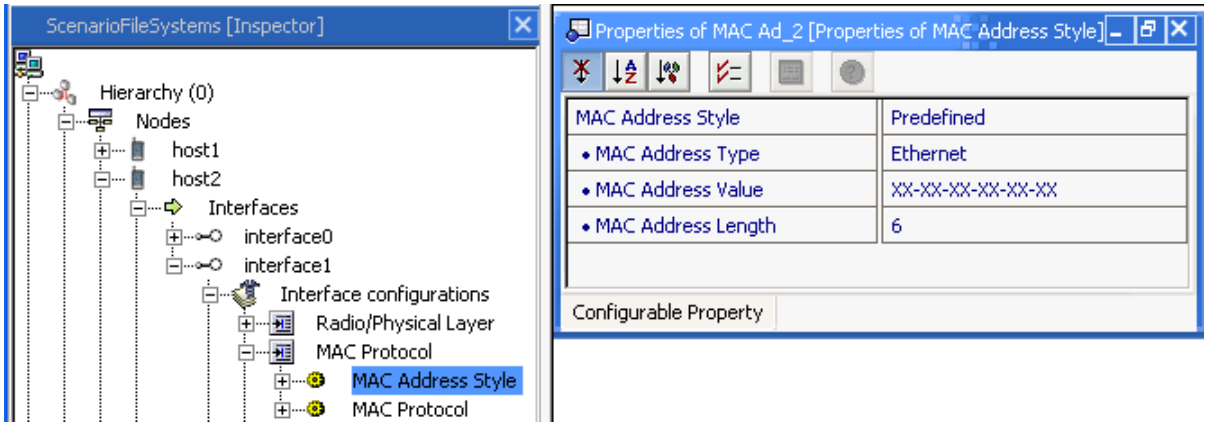


FIGURE 8. Configuring the MAC Address

Note: You can configure MAC addresses only when ARP is enabled.

Enabling Statistic Collection

To enable ARP Statistic collection, go to **Hierarchy (x) > Nodes > Host[x] > Node configurations > ARP Specs > ARP Enabled**. From the configurable property window, set **ARP Enabled** to **Yes**. Set **ARP Statistics** to **YES** as shown in Figure 9.

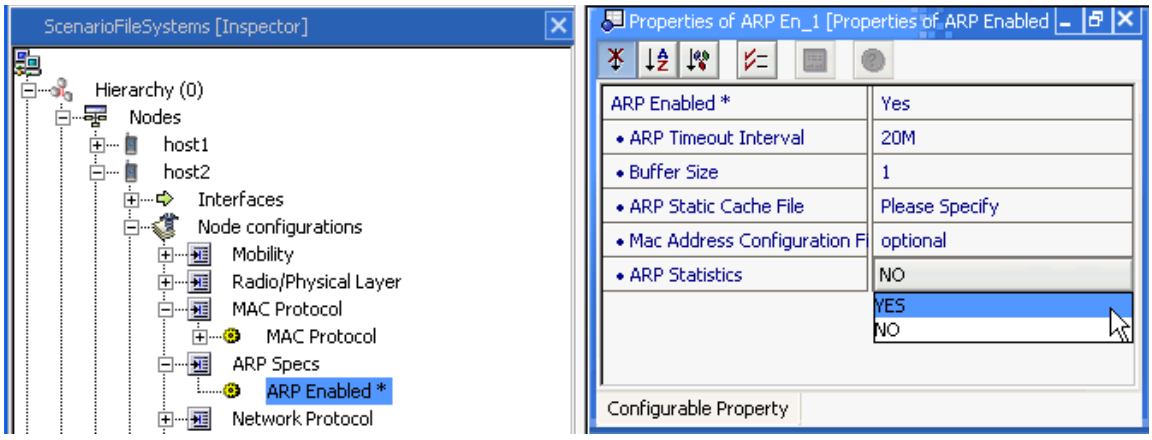


FIGURE 9. Enabling ARP Statistics

Statistics

A statistics file will be generated for the scenario, allowing you to view it and check the desired statistics. Table 8 shows this file, containing the following statistics, in addition to statistics for other layers.

TABLE 8. ARP Statistics Collected

Statistics	Description
Request Sent	Total number of ARP requests sent by the node.
Gratuitous Request Sent	Total number of gratuitous ARP requests sent by the node.
Request Received	Total number of ARP request packets received by the node.
Reply Sent	Total number of ARP reply packets sent by the node.
Reply Received	Total number of ARP reply received by the node.
Data Packet Discarded	Total number of data packets discarded because of inability to resolve Mac Address.
Cache Entry Inserted	Total number of entries inserted in the ARP cache table.
Cache Entry Updated	Total number of entries updated in the ARP cache table.
Cache Entry Agedout	Total number of entries expired due to time out.
Cache Entry Deleted	Total number of entries deleted when the interface is at fault and the entry is aged out.

Sample Scenario

Scenario Description

This is a sample of an ARP scenario. Nodes 1 to 6 are in a wired subnet. All the nodes are ARP enabled. A FTP/GENERIC application is configured between node 1 and node 2.

Topology

Figure 10 shows the topology of the sample scenario.

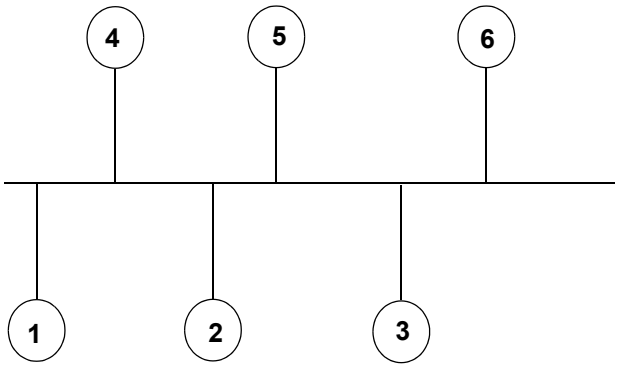


FIGURE 10. Sample ARP Scenario

Command Line Configuration

Set the parameters in the configuration file as described in the following steps.

1. Configure the Subnet.

```
SUBNET N8-192.168.100.0 {1 thru 6}
```

2. Enable ARP.

```
[N8-192.168.100.0] ARP-ENABLED YES
[N8-192.168.100.0] ARP-CACHE-EXPIRE-INTERVAL 20M
[N8-192.168.100.0] ARP-BUFFER-SIZE 1
[N8-192.168.100.0] ARP-STATISTICS YES
```

GUI Configuration

To configure ARP in the GUI, follow these steps:

1. For subnet [N8-192.168.100.0], from the configurable property window, set **ARP Enabled** to **Yes**.
2. When ARP is enabled, a set of its parameters is displayed. Configure these parameters
3. Go to **Hierarchy (x) > Nodes > Subnet Nx-x.x.x > Subnet Properties > ARP Enabled**. In the configurable property window, set **ARP Statistics** to **YES**.

Asynchronous Transfer Mode (ATM)

ATM is an International Telecommunication Union-Telecommunications Standards Section (ITU-T) standard. It is a connection-oriented cell relay protocol. Information bit streams are conveyed in small fixed-size cells (53 bytes). The ATM library in QualNet has implemented the ATM layer 2, ATM signaling, and ATM Adaptation Layer Type 5 (AAL5). It also supports interoperability between IP networks and ATM networks known as IP over ATM. You can use the ATM library to simulate pure ATM backbone networks as well as IP over ATM networks. The QualNet ATM library has implemented following features:

- Cell construction
- Cell reception and header validation
- Cell relaying, forwarding and copying
- Cell multiplexing and demultiplexing.
- Interpretation-only data cell and signaling cell
- Explicit forward congestion indication
- Point to point connection
- Connection assignment and removal
- Only CBR application is supported in a standalone ATM network
- Routing within the ATM clouds is done statically. Static route file is provided externally during configuration.
- BW is reserved for each application. Total link BW is shared among all the applications. If BW requirements exceed available BW, no virtual connection is established for that application. One virtual path is reserved for signaling.
- The virtual path setup process for each application is done dynamically, for example, using the SVC process.
- For SVC, connections are refreshed periodically and if a virtual circuit remains idle for a certain period (timeout-time), it is considered as timeout and BW is freed and can be reused.
- IP over ATM
  - IP Protocol Data Units are carried over the ATM cloud through the Gateways.
  - Gateways can be configured from the configuration file for each node present in IP clouds.
  - ATM-ARP is partially implemented for address discovery. Addresses are translated statically.
- The following applications (running at IP nodes) can go through ATM backbone: CBR, VBR, VOIP, SUPERAPPLICATION, FTP, FTP/GENERIC, HTTP, Lookup, TELNET. TRAFFIC-GEN & TRAFFIC-TRACE (without QoS parameter).

QualNet ATM library has following limitations

- QoS features are not implemented. All applications are allowed to use DEFAULT-SAAL-BW.
- Point to multiple connection is not supported
- IPv6 over ATM is not supported
- IP Multicast over ATM is not supported
- PNNI Routing is not supported
- PVC has not been implemented

Standalone ATM Backbone

ATM backbone network consists of ATM switches and ATM end system nodes. They are connected via wired point-to-point links.

ATM Adaptation Layer (AAL)

The ATM Adaptation Layer supports mapping between the ATM layer 2 and the next higher layer, and performs user-required functions such as control and management planes. It supports non-assured transfer of user data frames. The AAL type 5 is characterized by the fact that in most of the cells there is no overhead encountered. The AAL type 5 provides the capabilities to transfer the AAL-SDU from one AAL SAP to one another AAL SAP through the ATM network in the following manner:

Starting from the ATM cell stream on a single VCC (Virtual Channel connection), the only overhead the SAR sub layer uses is the payload type field in the last cell of a sequence of cells corresponding to a single PDU. A non-zero value of the AAL_Indicate field identifies the last cell in the sequence of cells indicating that the receiver can begin reassembly. The SAR sub layer reassembles the CPCS_PDU and passes it to the CPCS sub layer, which first uses the CRC-32 field to check for any errors in the received PDU. Normally, CPCS discards corrupted PDUs, but may optionally deliver them onto an SSCS. The CPCS removes the PAD and other trailer fields before passing the AAL-SDU across the AAL5-SAP. The transmit operation is the reverse of the above. Depending on the payload size (may be any integer number of octets in the range of 1 to 216-1), PAD field is chosen as a variable length so that the entire CPCS-PDU is an exact multiple of 48 and can be directly segmented into cell payloads. The length field identifies the length of the CPCS-PDU payload so that the receiver can remove the PAD field. Since 16 bits are allocated to the length field, the maximum payload length is 216-1= 65535 octets.

ATM Layer 2

ATM layer 2 is in between Physical layer and AAL layer. This is the most important layer of the ATM protocol stack. Most of the essential functionalities are performed at this layer. It receives 48 byte packets from AAL, attaches a five-byte header and sends it to the lower layer. Apart from the multiplexing and demultiplexing of cell stream, the translation of the connection identifiers (VPI / VCI) and cell switching is done at this layer at the intermediate nodes. This layer also has the mechanisms for traffic management and connection admission control. It checks for resource availability on receiving a request for connection establishment and checking the data rate offered by the user on a connection to the network.

ATM Signaling

Signaling procedure specifies the valid sequence of message exchanged between the user & the network, the rules for verifying consistency of the parameters, and the actions taken to establish and release ATM layer connections. There are various issues that govern the design of an efficient & effective signaling protocol. Such as acknowledgements, timer protection, parameter negotiation, etc.

Transmission media, being unreliable by nature, need a two-way or three-way hand shaking for a reliable communication. *Timer* is used to avoid inordinate delays in case the signaling message get lost or corrupted. *Parameter negotiation* refers to the process of fixing connection parameter through the exchange of signaling information. Two end systems can have more than one active connection simultaneously. To identify each call separately in such a scenario, you must introduce the C*all Identification process*. Generally, a call goes through three distinct phases:

1. ***Call establishment phase*:** Through the two-way or three-way handshaking process, each end system is informed about the current call & status of the other to make them ready for the data transfer process.

2. ***Data transfer phase***: The actual data transfer takes place in this phase.
3. ***Call clearing (releasing) phase:*** As the call is cleared in this phase, all resources allocated for the call are freed–up.

Command Line Configuration

Table 9 shows the configurable parameters for ATM.

TABLE 9. ATM Configuration Parameters

Parameter	Description
Configuration of ATM Layer	
ATM-LINK <network-prefix> {nodeId>,<nodeId>}	ATM supports only wired links. Each link is a point-to-point (serial) link between two nodes. These links are dedicated, error-free, and support the maximum bandwidth in both directions simultaneously. QualNet creates a network interface for each end of a link, and the ATM address is auto-assigned.
<network-prefix> ATM-LAYER2-LINK-BANDWIDTH <value>	This parameter specifies the bandwidth of the ATM Link in bits per second (bps). This parameter is mandatory. Range: The value should be greater than zero.
[<network-prefix>] ATM-LAYER2-LINK-PROPAGATION-DELAY <time>	This parameter specifies ATM Link propagation delay for wired point-to-point ATM links. This parameter is mandatory. Range: The value should be greater than zero.
ATM-SCHEDULER-STATISTICS YES | NO	To print the scheduler-specific statistics at each interface of an ATM node, use the parameter ATM-SCHEDULER-STATISTICS. The default value is NO.
ATM-LAYER2-STATISTICS YES | NO	To print the ATM Layer2 specific statistics at each interface of an ATM node, use the parameter ATM-LAYER2-STATISTICS. The default value is NO.
[nodeId] ATM-RED-MIN-THRESHOLD <value>	The number of packets in the queue that represents the lower bound at which packets can be randomly dropped or marked. The default value is 5. Range: The value should be greater than zero.
[nodeId] ATM-RED-MAX-THRESHOLD <value>	The number of packets in the queue that represents the upper bound at which packets can be randomly dropped or marked. The default value is 15. Range: The value should be greater than zero.
[nodeId] ATM-RED-MAX-PROBABILITY <value>	The maximum probability (0...1) at which a packet can be dropped (before the queue is completely full). Range: the value should be greater than zero and less than one.
[nodeId] ATM-RED-SMALL-PACKET-TRANSMISSION-TIME <value>	A sample amount of time that it would take to transmit a small packet, this is used to estimate the queue average during idle periods. The default value is 10MS.
[nodeId] ATM-QUEUE-SIZE <value>	The size of the queue. The default value is 15000.

TABLE 9. ATM Configuration Parameters (Continued)

Parameter	Description
[nodeId] SIGNALLING-STATISTICS YES | NO	SAAL is the signaling protocol in ATM layer 2. When the following variable is set to yes, ATM SAAL will show its statistics. The default value is NO.
[nodeId] ATM-CONNECTION-REFRESH-TIME <time>	This parameter specifies the refresh time of each connection. The default value is 25 minutes.
[nodeId] ATM-CONNECTION-TIMEOUT-TIME <time>	This parameter specifies timeout time of each connection. The default value is 1 minute.
[nodeId] ATM-LOGICAL-SUBNET-CONFIGURED YES	The end system ATM nodes should be placed in a logic subnet. This parameter must set to YES. The default value is NO.
[nodeId] ATM-LOGICAL-SUBNET <IPv4-subnet-prefix> {<node-ids-of-end-system-nodes>}	This parameter specifies the logical subnet.
Configuration for ATM ADAPTATION LAYER & SIGNALLING	
[<node-id-list>] ATM-NODE	This parameter specifies the ATM nodes. This parameter is mandatory.
[<node-id-list>] ATM-END-SYSTEM	Some of the nodes are defined as ATM end systems. These nodes are supposed to be edge ATM nodes which interface with other types of networks. In a standalone ATM network scenario, the ATM end system nodes are those nodes where applications can be defined. All other ATM nodes are treated as ATM-Switch. This parameter is mandatory.
[Node Id] ADAPTATION-PROTOCOL	Presently AAL5 is implemented as the ADAPTATION LAYER PROTOCOL for ATM. AAL5 should be selected as ADAPTATION LAYER protocol for ATM. This parameter is mandatory.
ATM-STATIC-ROUTE YES| NO	This parameter specifies whether of not ATM use the static route. By default it is set to NO. As ATM routes the packet using static route only this parameter should set to YES. This parameter is mandatory. The default value is No.
ATM-STATIC-ROUTE-FILE	This parameter specifies the location of the *.static.route file. Note: If ATM-STATIC-ROUTE is set to YES, it is mandatory

GUI Configuration

Configuring General ATM Parameters

To configure general ATM parameters, perform the steps described below. Note that you must select the **ATM** button in the **Devices** tab to create ATM nodes.

1. Go to **Hierarchy (x) > Nodes > Host[x]**. In the Configurable Property window, set **Node Type** to **End System** or **Switch**, as shown in Figure 11.

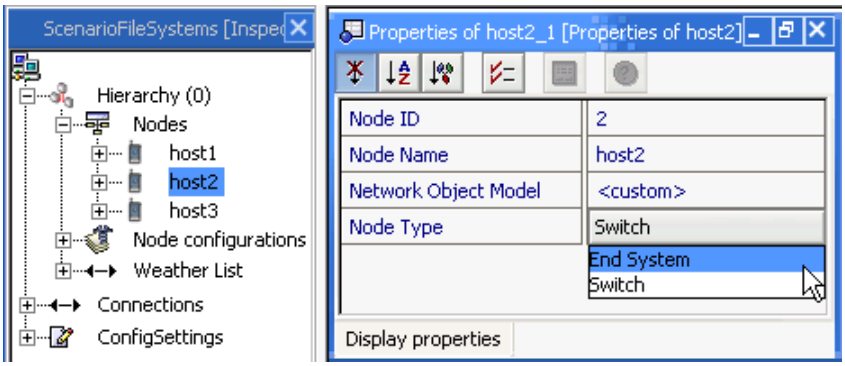


FIGURE 11. Configuring ATM Node Type

2. Go to **Hierarchy (x) > Nodes > host[x] > Interfaces > atminterface[x] > Interface configurations > ATM Layer2**. From the configurable property window, set **ATM Layer2 Link Bandwidth** and **ATM Layer2 Link Propagation Delay** shown in Figure 12.

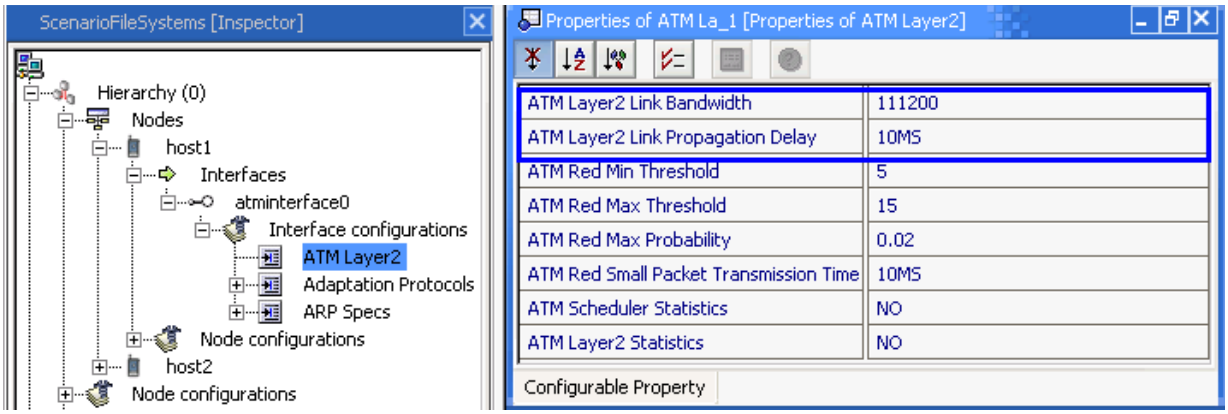


FIGURE 12. Setting ATM Link Properties

3. Go to **ConfigSettings > Adaptation Protocols > Adaptation Protocols.** From the configurable property window:
   - Set **Adaptation Protocol** to **AAL5**, as shown in Figure 13.

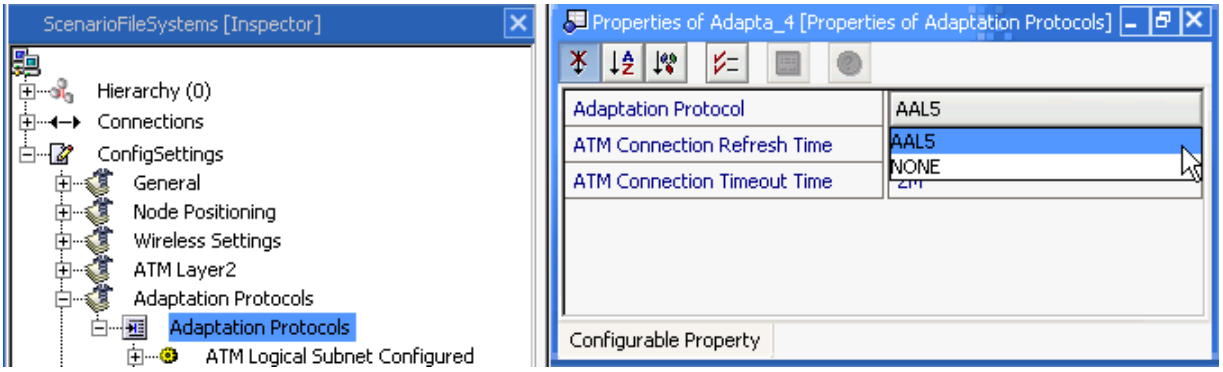


FIGURE 13. Setting Adaptation Protocol

- Set **ATM Logical Subnet Configured** to **Yes**, as shown in Figure 14.

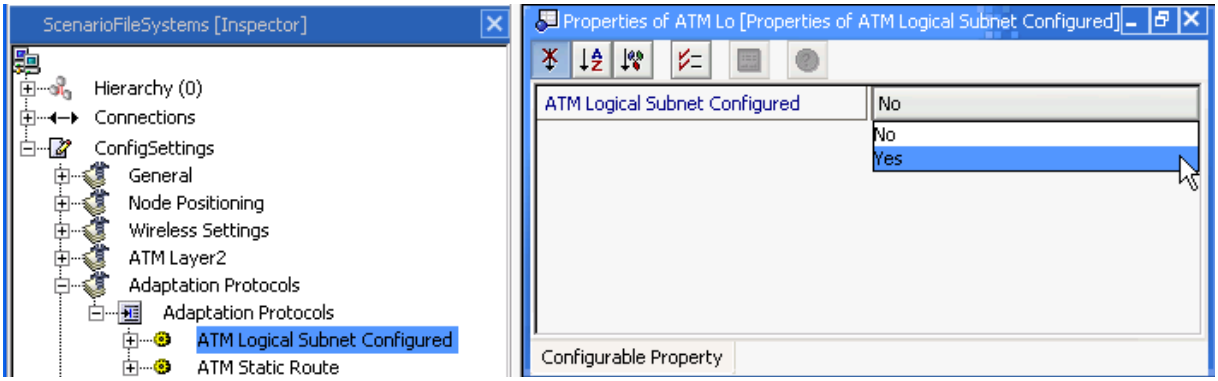


FIGURE 14. Setting Logical Subnet Configured to Yes

- When **ATM Logical Subnet Configured** is enabled, configure it as shown in Figure 15.

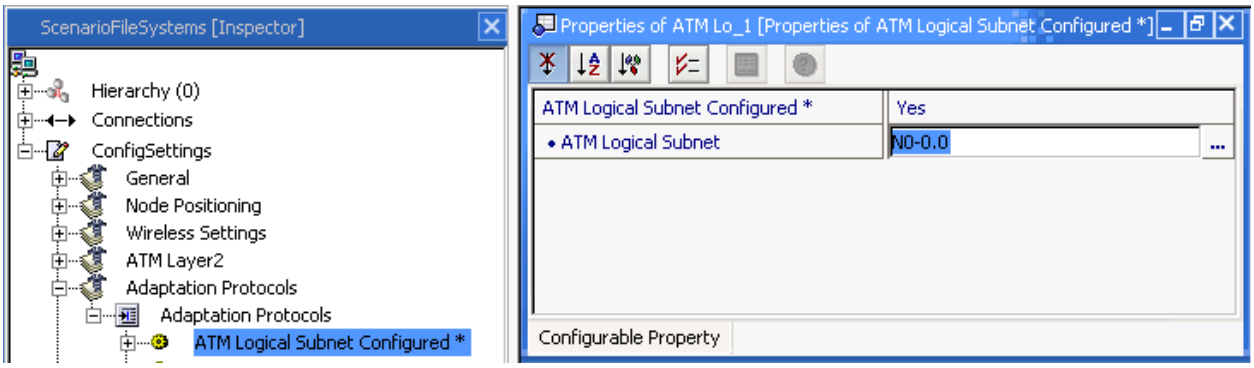


FIGURE 15. Configure ATM Logical Subnet

- Set **ATM Static Route** to **Yes** as shown in Figure 16.

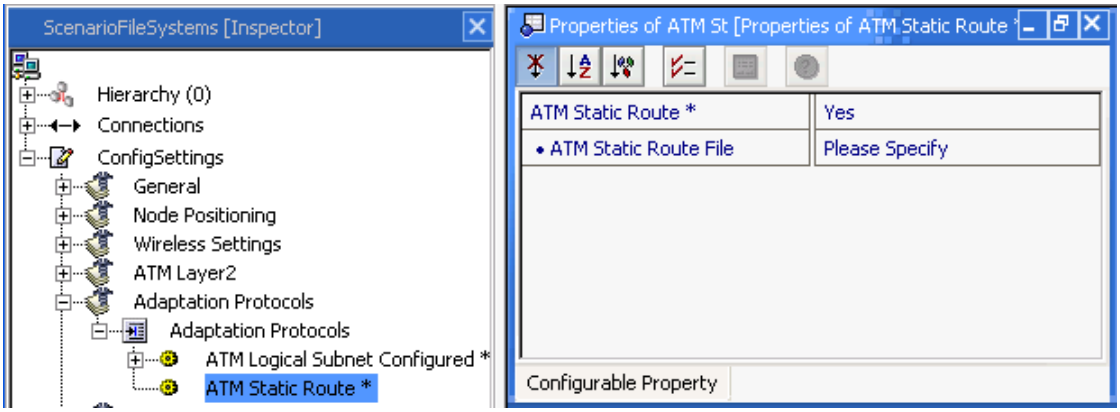


FIGURE 16. Set ATM Static Route to Yes

- When **ATM Static Route** is enabled, specify the location of the static route file.

4. Go to **Hierarchy (x) > Nodes > host[x] > Interfaces > atminterface[ID] > Interface configurations > ATM Layer2**. From the configurable property window, configure ATM RED queues parameters as shown in Figure 17.

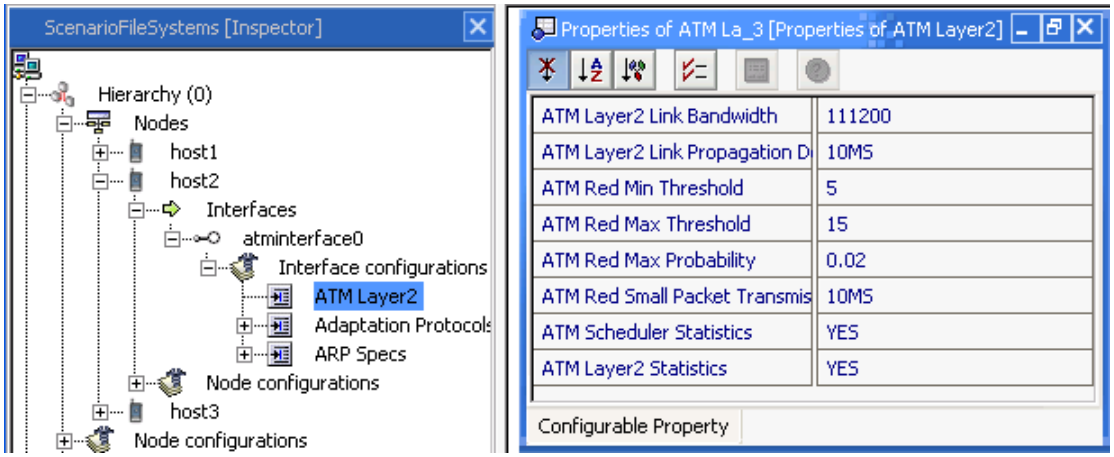


FIGURE 17. Configuring ATM RED Queues Parameters

Enabling Statistic Collection

Follow these steps to enable ATM statistic collection:

1. Go to **Hierarchy (x) > Nodes > host[x] > Interfaces > atminterface[ID] > Interface configurations > ATM Layer2**. From the configurable property window, set **ATM Scheduler Statistics** and **ATM Layer2 Statistics** to **YES** as shown in Figure 18.

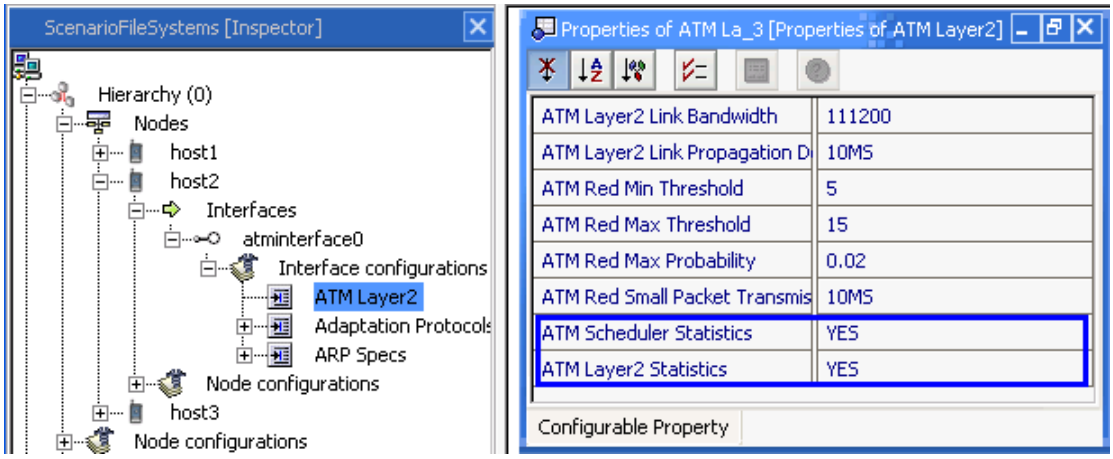


FIGURE 18. Enabling ATM Scheduler Statistics

2. Go to **Hierarchy (x) > Nodes > host[x] > Interfaces > atminterface[ID] > Interface configurations > Adaptation Protocols**. From the configurable property window, set **Adaptation Layer Statistics** and **Signalling Stats** to **YES** as shown in Figure 19

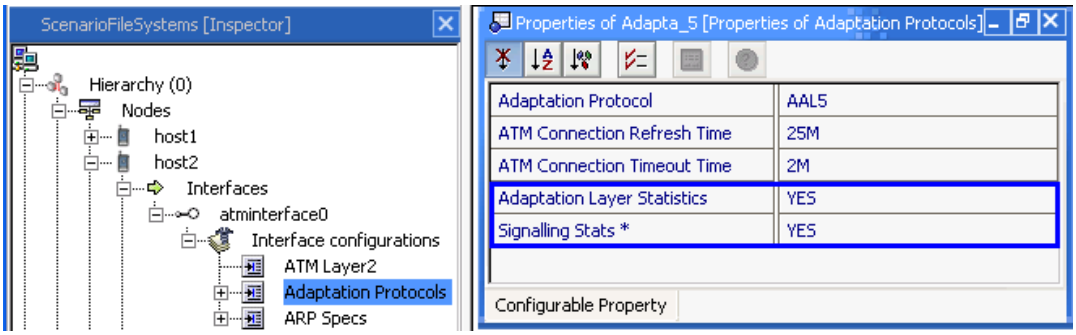


FIGURE 19. Enabling Adaptation Layer Statistics

Statistics

Table 10 shows the ATM statistics.

TABLE 10. ATM Statistics

Parameter	Description
Destination	Next destination, Node connected to the other side of this Point-to-Point link/interface.
Cell Received	Total number of cells received in the interface by ATM LAYER2.
Cell Forward	Total number of cells forwarded through the interface ATM LAYER2.

TABLE 10. ATM Statistics (Continued)

Parameter	Description
Cell Discarded for no Route	Total number of cells discarded due to unavailability of route ATM LAYER2.
Control Cell Received	Total number of control cells (signaling cell) received ATM LAYER2.
Cell Delivered to upper layer	Total number of cells delivered to upper layer (adaptation layer) ATM LAYER2.
Link Utilization	Link utilization ratio by ATM LAYER2.
Number of AAL Service Data Units Sent	Total number of AAL CPCS service data units sent from this node by AAL5.
Number of AAL Service Data Units Received	Total number of AAL CPCS service data units received in this node by AAL5.
Number of ATM Service Data Units Sent	Total number of SDUs sent to ATM layer 2 after segmentation by AAL5.
Number of ATM Service Data Units Received	Total number of SDUs received from ATM layer2 before reassembly by AAL5.
Number of ATM Service Data Units Dropped	Total number of SDUs dropped due to error, like intermediate cell drop etc. by AAL5.
Number of Packets Segmented	Packet (Data unit) segmented into number of cells by AAL5.
Number of Packets Reassembled	Total number of cells reassembled to number of Packets (Data units) by AAL5.
Number of SAAL Packets Sent	Total number of signaling packets sent from this node by AAL5.
Number of SAAL Packets Received	Total number of signaling packets received in this node by AAL5.
Number of Data Packets Sent	Total number of data packets forwarded through this node by AAL5.
Number of Data Packets Received	Total number of data packets received in this node by AAL5.
Number of Data Packets Received from IP	Total number of data packets received from IP layer by AAL5.
Number of Data Packets Forwarded to IP	Total number of data packets forwarded to IP layer by AAL5.
Number of IP Packets Discarded	Total number of erroneous data packets (multicast/control pkt) discarded. These packets are not further processed within ATM cloud by AAL5.
Number of IP Packets Dropped	Total number of data packets dropped due to no route after processing within ATM cloud by AAL5.
Number of Alert Messages Sent	Total number of alert messages sent by SAAL.
Number of Alert Messages Received	Total number of alert messages received SAAL.
Number of Connect Messages Sent	Total number of connect messages sent by SAAL.
Number of Connect Messages Received	Total number of connect messages received by SAAL.
Number of Setup Messages Sent	Total number of setup messages sent by SAAL.
Number of Setup Messages Received	Total number of setup messages received by SAAL.

Sample Scenario

Scenario Description

The scenario in Figure 20 shows an ATM scenario. Nodes 1 to 11 are ATM nodes are connected by ATM links as shown in the figure. Nodes 1, 3, 9, 10, 11 are ATM end systems. Nodes 2, 4, 5, 6, 7, 8 are ATM switches. Adaptation protocol is set to AAL5 and ATM static route is configured.

Topology

The scenario topology is shown in Figure 20.

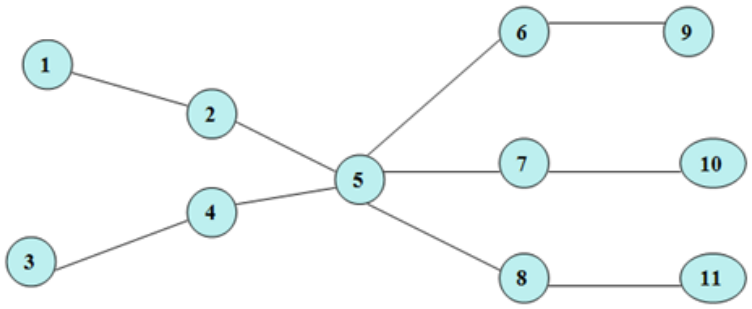


FIGURE 20. Standalone ATM Network Scenario

Command Line Configuration

Set the parameters in the configuration file as described in the following steps.

1. Configure the ATM Link:

```
ATM-LAYER2-LINK-BANDWIDTH           112000
ATM-LAYER2-LINK-PROPAGATION-DELAY   10MS

ATM-LINK ICD-91.AID-0.PTP-0 { 1, 2 }
ATM-LINK ICD-91.AID-0.PTP-1 { 2, 5 }
ATM-LINK ICD-91.AID-0.PTP-2 { 3, 4 }
ATM-LINK ICD-91.AID-0.PTP-3 { 4, 5 }
ATM-LINK ICD-91.AID-0.PTP-4 { 5, 6 }
ATM-LINK ICD-91.AID-0.PTP-5 { 5, 7 }
ATM-LINK ICD-91.AID-0.PTP-6 { 5, 8 }
ATM-LINK ICD-91.AID-0.PTP-7 { 6, 9 }
ATM-LINK ICD-91.AID-0.PTP-8 { 7, 10 }
ATM-LINK ICD-91.AID-0.PTP-9 { 8, 11 }

[1 thru 11] ATM-NODE YES
[1 3 9 10 11] ATM-END-SYSTEM YES
```

2. Configure Adaptation Layer Parameters.

```
ADAPTATION-PROTOCOL AAL5
```

ATM RED Queue is a special type of Random Early Detection (Drop) queue. This implementation marks packets; it does not drop them. ATM RED Queue uses the following setting.

```
ATM-RED-MIN-THRESHOLD     5
ATM-RED-MAX-THRESHOLD     15
ATM-RED-MAX-PROBABILITY   0.02
ATM-RED-SMALL-PACKET-TRANSMISSION-TIME   10MS
```

3. Configure Logical Subnet.

The end system nodes should be placed in a logical subnet defined as the following:

```
ATM-LOGICAL-SUBNET-CONFIGURED YES
ATM-LOGICAL-SUBNET N8-192.168.5.0 {1, 3, 9, 10, 11}
```

4. Configure ATM Static Route.

ATM routes the packet using static route only to collect the path information from the mentioned ATM-STATIC-ROUTE-FILE.

```
ATM-STATIC-ROUTE YES
ATM-STATIC-ROUTE-FILE  ./atm.route-static
```

For this sample scenario, the content of the static route file is given as below.

Note: Only routes to end system nodes are specified as the source, and destination nodes must be end system nodes.

```
1 ICD-91.AID-0.PTP-0:1 0
1 ICD-91.AID-0.PTP-2:3 0
1 ICD-91.AID-0.PTP-7:9 0
1 ICD-91.AID-0.PTP-8:10 0
1 ICD-91.AID-0.PTP-9:11 0

2 ICD-91.AID-0.PTP-0:1 0
2 ICD-91.AID-0.PTP-2:3 1
2 ICD-91.AID-0.PTP-7:9 1
2 ICD-91.AID-0.PTP-8:10 1
2 ICD-91.AID-0.PTP-9:11 1

3 ICD-91.AID-0.PTP-0:1 0
3 ICD-91.AID-0.PTP-2:3 0
3 ICD-91.AID-0.PTP-7:9 0
3 ICD-91.AID-0.PTP-8:10 0
3 ICD-91.AID-0.PTP-9:11 0

4 ICD-91.AID-0.PTP-0:1 1
4 ICD-91.AID-0.PTP-2:3 0
4 ICD-91.AID-0.PTP-7:9 1
4 ICD-91.AID-0.PTP-8:10 1
```

```
4 ICD-91.AID-0.PTP-9:11 1

5 ICD-91.AID-0.PTP-0:1 0
5 ICD-91.AID-0.PTP-2:3 1
5 ICD-91.AID-0.PTP-7:9 2
5 ICD-91.AID-0.PTP-8:10 3
5 ICD-91.AID-0.PTP-9:11 4

6 ICD-91.AID-0.PTP-0:1 0
6 ICD-91.AID-0.PTP-2:3 0
6 ICD-91.AID-0.PTP-7:9 1
6 ICD-91.AID-0.PTP-8:10 0
6 ICD-91.AID-0.PTP-9:11 0

7 ICD-91.AID-0.PTP-0:1 0
7 ICD-91.AID-0.PTP-2:3 0
7 ICD-91.AID-0.PTP-7:9 0
7 ICD-91.AID-0.PTP-8:10 1
7 ICD-91.AID-0.PTP-9:11 0

8 ICD-91.AID-0.PTP-0:1 0
8 ICD-91.AID-0.PTP-2:3 0
8 ICD-91.AID-0.PTP-7:9 0
8 ICD-91.AID-0.PTP-8:10 0
8 ICD-91.AID-0.PTP-9:11 1

9 ICD-91.AID-0.PTP-0:1 0
9 ICD-91.AID-0.PTP-2:3 0
9 ICD-91.AID-0.PTP-7:9 0
9 ICD-91.AID-0.PTP-8:10 0
9 ICD-91.AID-0.PTP-9:11 0

10 ICD-91.AID-0.PTP-0:1 0
10 ICD-91.AID-0.PTP-2:3 0
10 ICD-91.AID-0.PTP-7:9 0
10 ICD-91.AID-0.PTP-8:10 0
10 ICD-91.AID-0.PTP-9:11 0

11 ICD-91.AID-0.PTP-0:1 0
11 ICD-91.AID-0.PTP-2:3 0
11 ICD-91.AID-0.PTP-7:9 0
11 ICD-91.AID-0.PTP-8:10 0
11 ICD-91.AID-0.PTP-9:11 0
```

5. Configure Application

   Only CBR applications are supported in a standalone ATM networks. Source and destination nodes must be ATM end system nodes. For example, a sample application file defined in *.app file, is as follows:

```
CBR 1 11 1 512 1M 1M 30M
CBR 11 1 1 512 1M 1M 30M
CBR 3 9 1 512 1M 1M 30M
CBR 9 3 1 512 1M 1M 30M
```

   To transmit the packets, paths are established dynamically using SVC (for example, the translation table entry is created through the exchange of various signaling messages). Once the path is recognized, cells are transmitted through it.

6. Enable statistics collection.

```
ADAPTATION-LAYER-STATISTICS    YES
SIGNALLING-STATISTICS          YES

ATM-SCHEDULER-STATISTICS        YES
ATM-QUEUE-STATISTICS            YES
ATM-LAYER2-STATISTICS           YES
```

GUI Configuration

Follows these steps to configure ATM using the GUI:

1. For nodes 1, 3, 9, 10, 11 go to **Hierarchy (x) > Nodes > host[ID]**, and from the configurable property window, set **Node Type** to **End System**.
2. For all Links, go to **Connections > ATM Link[ID] > ATM Link Properties.** From the configurable property window, set **ATM Layer2 Link Bandwidth** to 112000 bps and **ATM Layer2 Link Propagation Delay** as 50MS.
3. For all nodes, go to **Hierarchy (x) > Nodes > host[ID] > Interfaces > atminterface[ID] > Interface configurations > Adaptation Protocols**, and configure different Adaptation Protocols Parameters.
4. For all nodes, go to **Hierarchy (x) > Nodes > host[ID] > Interfaces > atminterface[ID] > Interface configurations > ATM Layer2**, and configure ATM RED queues parameters.
5. For all nodes, go to **Hierarchy (x) > Nodes > host[ID] > Interfaces > atminterface[ID] > Interface configurations > ATM Layer2**, and set **ATM Scheduler Statistics** and **ATM Layer2 Statistics** to **YES**.
6. For all nodes go to **Hierarchy (x) > Nodes > host[ID] > Interfaces > atminterface[ID] > Interface configurations > Adaptation Protocols**, and set **Adaptation Layer Statistics** and **Signalling Stats** to **YES**.

IP over ATM

In QualNet, the interoperability between ATM backbone and IP subnets is implemented following the classical IP over ATM specifications. You can forward IP packets through ATM backbones. It introduces the concept of a logical entity called LOGICAL IP SUBNET which treats the ATM network as a number of separate IP subnets connected through routers. Membership to an LIS is defined by software configuration. Transportation of various protocol data units over an ATM network is categorized into the following subsections:

- Addressing
- Address resolution
- Data encapsulation
- Routing

Addressing

In ATM Forum Specifications the term address is used to mean locator (for example, address indicates the location of an interface). An individual ATM address is 20 octets long and identifies the location of a single ATM interface. Basically the total address consists of following parts in this format.

The Initial Domain Part (IDP) uniquely specifies an administration authority, which has the responsibility for allocating and assigning values of the Domain Specific Part (DSP). In our implementation we consider only the International Code Designator (ICD) Address format. International organizations use this format for address allocation.

Address Resolution

For operation of IP over ATM, a mechanism is used to resolve IP addresses to the corresponding ATM addresses within an ATM logical IP subnet. It uses ATMARP and InATMARP protocols for this purpose. All nodes within the LIS are configured with the unique ATM address of the ATMARP server, which is used to resolve the requests of the LIS clients to map the IP address to the ATM address. At present no ATMARP Server is configured separately, rather, a static table is provided with all ARP related information and an address table is kept at the end systems for this purpose.

Data Encapsulation

IETF describes two encapsulation methods for carrying network interconnect traffic over the ATM AAL5. These first of these two methods, LLC/SNAP Encapsulation, allows the transport network and link layer packets across an ATM connection. The second method of encapsulation, VC Multiplexing, allows the multiplexing of multiple packet types on the same connection. These methods allow you to conserve connection resource space (all data transfer between two nodes across the same connection) and to save connection latency after the first connection setup.

- LLC/SNAP Encapsulation - This encapsulation method is used for carrying multiple protocols over a single ATM VC connection. An LLC header placed in front of the carried Protocol Data Unit (PDU) identifies the packet type (IP, IPX, and AppleTalk). This approach is preferred when separate VCs for each carried protocol are either expensive or not possible (for example, the only connection type supported is PVC or charging is based on the number of VC allocated). This method is the default encapsulation method for all IP over ATM protocols.
- VC Multiplexing - In this method, each protocol is carried over a separate ATM VC, with the type of protocol identified at connection setup. With VC Based Multiplexing, the data is encapsulated into the CPCS-PDU field of AAL5 directly, so each connection can only carry one protocol. This approach is preferred when you wish to establish a large number of VCs in a fast and economical way.

Routing

Routing is another important aspect of carrying IP Protocol Data Unit over ATM. In a mixed topology of IP (connectionless), and ATM (connection oriented), paths among various routers can be created dynamically, but there is some difficulty related to QoS. An alternative is to create that path through management action so that traffic engineering can be accomplished. This leads to the concept of Gateway.

Different IP clouds are connected to ATM backbone through their respective Gateways, which act as the entry point to other (next Hop) networks (for example, ATM cloud). To reach another IP network, the egress end systems are discovered in SVC. Setup messages are forwarded to all the end systems attached in that logical network. The end system with the proper outgoing route replies back with an alert message to the desired IP. Once the path is established, data can be transferred only through that path.

Command Line Configuration

Additional parameters are needed to configure an IP over ATM network scenario.

1. Nodes that connect IP subnets with the ATM backbone must be configured as ATM end system nodes. End system nodes should be placed in a logical IP subnet.
2. The ARP module should be configured to provide the capability to translate logical IP addresses to ATM addresses of ATM interfaces.

```
ARP-ENABLED                     YES | NO
ARP-USE-BUFFER                  YES | NO
ARP-STATIC-CACHE-FILE           <static-arp-file>
ARP-STATISTICS                  YES | NO
```

   ARP static cache files are provided with the following syntax:

```
<node-id>
[<interface-index>] -optional field
<protocol-type i.e [IP]>
<protocol-address> | <node-interface>
<hardware-type [ETHRNET/ATM]>
<hardware-address>
<time-out>
```

   When <time-out> is 0, the entry won't be timed out. For IP over ATM, a timeout value of 0 must be specified. For example:

```
1    IP    192.168.5.1     ATM    23:4f:5C:aa:FE:00    0
2    IP    192.168.5.2     ATM    23:4f:5C:aa:FE:01    0
```

3. To help routing, each IP subnet must configure a default gateway that routes packets to the gateway node connecting the IP subnet to the ATM backbone.
4. The ATM backbone must be configured similarly to the previous section:

```
[<node-id>] DEFAULT-GATEWAY    <node-id or ip-address of gateway node>
```

   Some of the ATM end systems are configured as the gateway for associated IP clouds. These nodes are responsible for transferring data from IP to ATM clouds and vice versa. The gateway address may be nodeId or IP Address.

Sample Scenario

Scenario Description

This sample is an IP over ATM scenario consisting of 12 nodes, among which 3, 4, 5, 6, 7 and 8 are present in ATM cloud and the rest are in IP cloud. As node 3, 7 and 8 have interface both in IP and ATM, thus they are considered as end-systems.

Topology

Figure 21 shows the topology of the IP over ATM scenario.

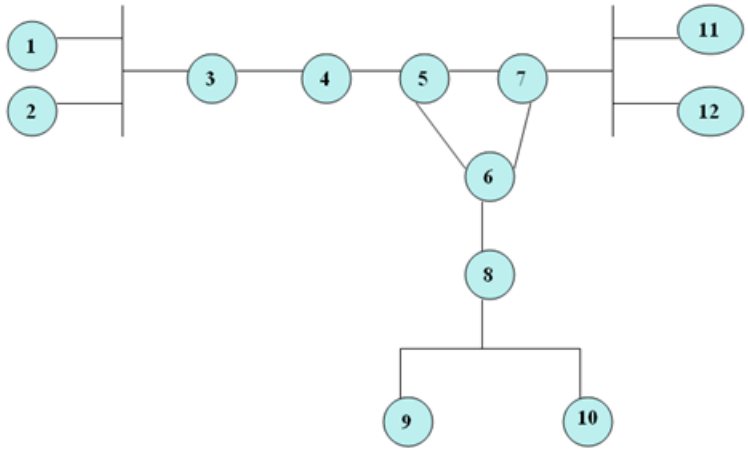


FIGURE 21. IP Over ATM Scenario

Command Line Configuration

Set the parameters in the configuration file as described in the following steps.

1. Configure the ATM part:

```
ATM-LAYER2-LINK-BANDWIDTH           1112000
ATM-LAYER2-LINK-PROPAGATION-DELAY   10MS
```

2. Configure the ATM cloud for nodes 3, 4, 5, 6, 7 & 8.

```
ATM-LINK ICD-91.AID-1.PTP-1 {3, 4}
ATM-LINK ICD-91.AID-1.PTP-2 {4, 5}
ATM-LINK ICD-91.AID-1.PTP-3 {5, 6}
ATM-LINK ICD-91.AID-1.PTP-4 {6, 8}
ATM-LINK ICD-91.AID-1.PTP-5 {6, 7}

[3 4 5 6 7 8] ATM-NODE YES
[3 7 8] ATM-END-SYSTEM YES
```

3. Configure the IP part.

```
SUBNET N8-1.0 {1 thru 3}

SUBNET N8-2.0 {7, 11, 12}

SUBNET N8-3.0 {8 thru 10}

LINK-BANDWIDTH              112000
LINK-PROPAGATION-DELAY      50MS
```

4. Configure Default-Gateway Part

```
[1 2 3] DEFAULT-GATEWAY 3
[7 11 12] DEFAULT-GATEWAY 7
[8 9 10] DEFAULT-GATEWAY 8
```

5. Configure Logical Subnet Part.

```
ATM-LOGICAL-SUBNET N8-8.0 {3, 7, 8}

ADAPTATION-PROTOCOL AAL5
ADAPTATION-LAYER-STATISTICS YES
ATM-STATIC-ROUTE YES
ATM-STATIC-ROUTE-FILE atm.route-static
ARP-ENABLED YES
ARP-USE-BUFFER YES
ARP-STATIC-CACHE-FILE atm.arp-static
ARP-STATISTICS YES

SIGNALLING-STATISTICS          YES
ATM-SCHEDULER-STATISTICS       NO
ATM-LAYER2-STATISTICS          YES
```

6. Configure Optional Parameter.

```
ATM-CONNECTION-REFRESH-TIME  15M
ATM-CONNECTION-TIMEOUT-TIME  2M

ATM-RED-MIN-THRESHOLD     5
ATM-RED-MAX-THRESHOLD     15
ATM-RED-MAX-PROBABILITY   0.02
ATM-RED-SMALL-PACKET-TRANSMISSION-TIME   10MS

NETWORK-PROTOCOL IP
MAC-PROTOCOL MAC802.3
ROUTING-PROTOCOL OSPFv2
```

All other default parameters are unaltered.

In this configuration, a CBR application is running from 1 to 12.

```
CBR 1 12 0 512 1M 20S 0M
```

While running the simulation, the following happens:

1. First, all address are assigned to all nodes in this manner:

```
Node 3 assigning Address (47.5b.1:1.3:1.3:1.0) to AtmInf 0,GenInf 1
Node 4 assigning Address (47.5b.1:1.4:2.4:2.0) to AtmInf 0,GenInf 0
Node 4 assigning Address (47.5b.1:2.4:1.4:1.0) to AtmInf 1,GenInf 1
Node 5 assigning Address (47.5b.1:2.5:2.5:2.0) to AtmInf 0,GenInf 0
Node 5 assigning Address (47.5b.1:3.5:1.5:1.0) to AtmInf 1,GenInf 1
Node 6 assigning Address (47.5b.1:3.6:2.6:2.0) to AtmInf 0,GenInf 0
Node 6 assigning Address (47.5b.1:4.6:1.6:1.0) to AtmInf 1,GenInf 1
Node 8 assigning Address (47.5b.1:4.8:2.8:2.0) to AtmInf 0,GenInf 1
Node 6 assigning Address (47.5b.1:5.6:1.6:1.0) to AtmInf 2,GenInf 2
Node 7 assigning Address (47.5b.1:5.7:2.7:2.0) to AtmInf 0,GenInf 1
```

2. Logical Subnet members are assigned with proper addresses.
3. The ARP static table and ATM forwarding table are initialized.

```
-------------------------------------------------------------------------
ProtoType   ProtoAddr   RelativeInf ExpireTime EntryTp  HwType   HwAddr
-------------------------------------------------------------------------
2048        0.0.8.2    0  0.000000000  0  19    5C-AA-FE-23-4F-C2
2048        0.0.8.3    0  0.000000000  0  19    5C-BA-FE-23-4F-F2

#3 has Fwd Table:
-----------------------------------
Destn    outIntf    nextNode
 3|1     0          4|2
 7|2     0          4|2
 8|2     1          5|2
```

4. With the first application packet, the destination is not available in the source node's forwarding table, so the packets must be routed through the default gateway. At the gateway nodes, it passes the packet to the adaptation layer, puts an LLC encapsulation, and passes it to AAL5 layer. At this point, it's assigned a VCI and VPI and checked for signaling.
5. A new entry is added in connection table.

```
# 3 has Connection Table:
Src   Dst   SPort  Dport  VCI   VPI  status lastPkt

101   203   1024   59    28839  255   1     60.000
```

6. As signaling is not yet started, check for BW availability. If it's available, various signaling messages are exchanged. Meanwhile, it searches for the edge router, which acts as an exit point from the ATM cloud. If the matching entry is found in the Address-Info table (ATM ARP Table), the exit point is known. Otherwise, using the information present in the forwarding table sets up messages sent to all end

systems and the one having a matching entry, reverts back with an Alert message. An entry is added in the translation table.

```
# 4 has Translation Table:
-----------------------------------------
 VCI      VPI     finalVCI   finalVPI outIntf
 28839     1        28839         1         2
 57727     1        57727         2         0
```

7. Each data packet is fragmented into an ATM cell and passed to the destination using the information present in the translation table. Exit point packets will be routed using normal IP routing. After each ATM-CONNECTION-REFRESH-TIME connection table is refreshed, and if the BW remains idle for more than ATM-CONNECTION-TIMEOUT-TIME, all reserved resources are freed.

At the end of the simulation, all information is noted down in the associated .stat file.

GUI Configuration

To configure the scenario in QualNet GUI, perform the steps described below. Note that you must select the **ATM** button in the **Devices** tab to create ATM nodes.

1. For nodes 1, 2, 3, 7, 8, 9, 10, 11, and 12, go to **Hierarchy (0) > Nodes > host #**. In the Configurable Property window, set **Node Type** to **End System**, as shown in Figure 22.

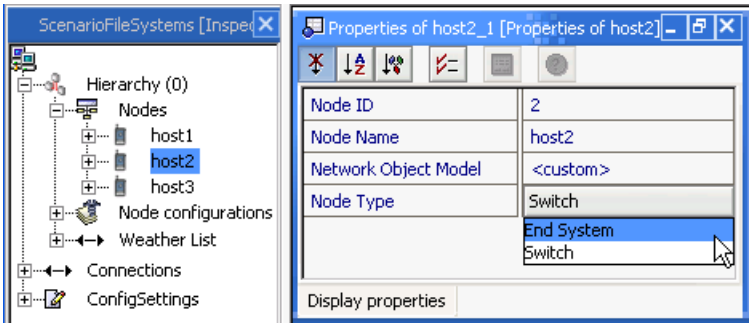


FIGURE 22. Setting the ATM Device Properties

2. For all the ATM links, go to **Hierarchy (0) > Nodes > host # > Interfaces > atminterface # > Interface configurations > ATM Layer2**. In the Configurable Property window, set **ATM Layer2 Link Bandwidth** to 112000 and **ATM Layer2 Propagation Delay** to 50MS, as shown in Figure 23.

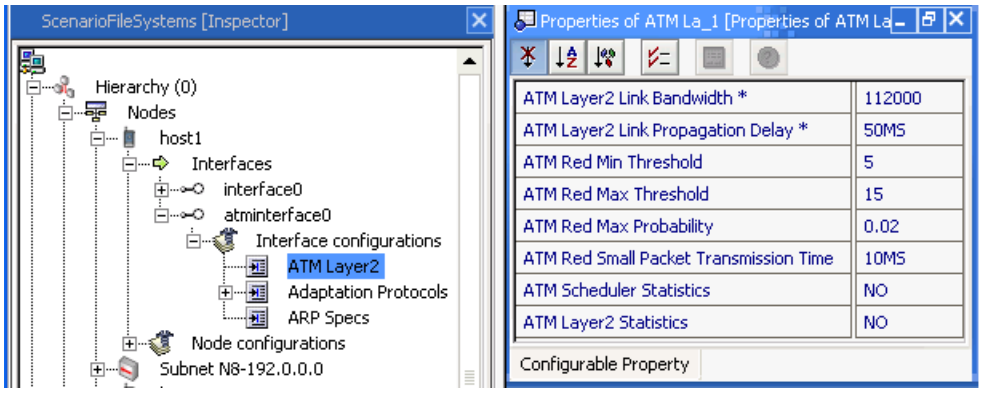


FIGURE 23. Setting the ATM Link Properties

3. For nodes 3, 4, 5, 6, 7 and 8, go to **Hierarchy (0) > Nodes > host # > Interfaces > atminterface # > Interface configurations > Adaptation Protocols**. In the Configurable Property window, set **Adaptation Protocol** to **AAL5**, as shown in Figure 24.

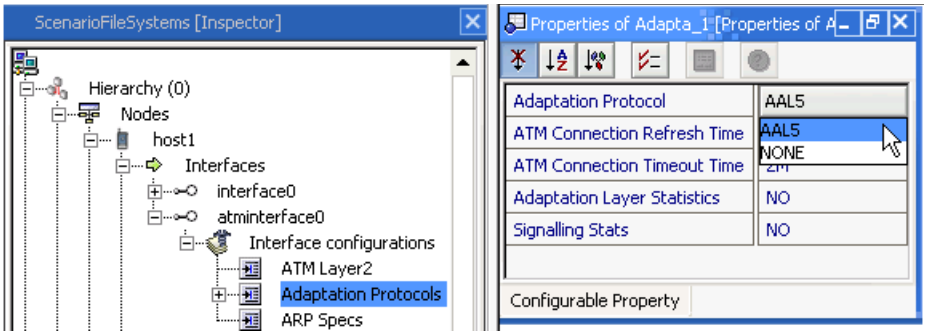


FIGURE 24. Setting Adaptation Protocol

4. Go to **ConfigSettings > Adaptation Protocols > Adaptation Protocols > ATM Logical Subnet Configured**. In the Configurable Property window, set **ATM Logical Subnet Configured** to **Yes** and configure the properties, as shown in Figure 25.

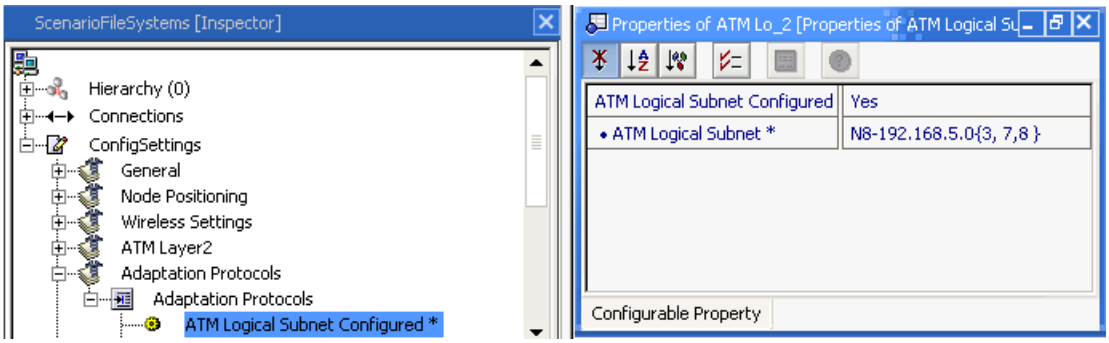


FIGURE 25. Configuring ATM Logical Subnet

5. For nodes 3, 4, 5, 6, 7 and 8, go to **Hierarchy (0) > Nodes > host # > Interfaces > atminterface # > Interface configurations > Adaptation Protocols > ATM Static Route**. In the Configurable Property window, set **ATM Static Route** to **Yes** and specify the static route file, as shown in Figure 26.

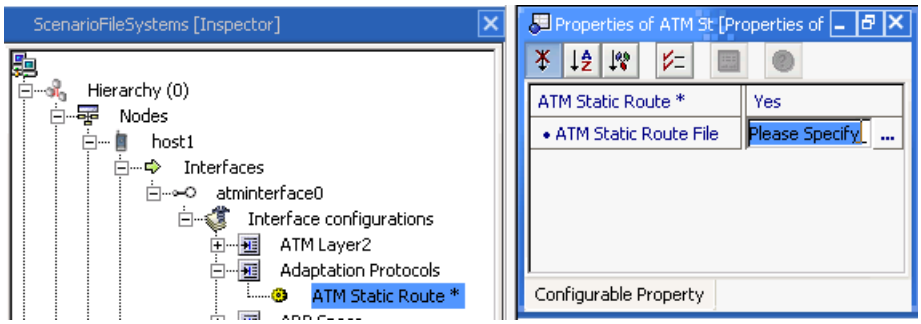


FIGURE 26. Specifying ATM Static Routes

6. Go to **ConfigSettings > ARP Specific > ARP Specs > ARP Enabled**. In the Configurable Property window, set **ARP Enabled** to **YES** and configure the properties, as shown in Figure 27.

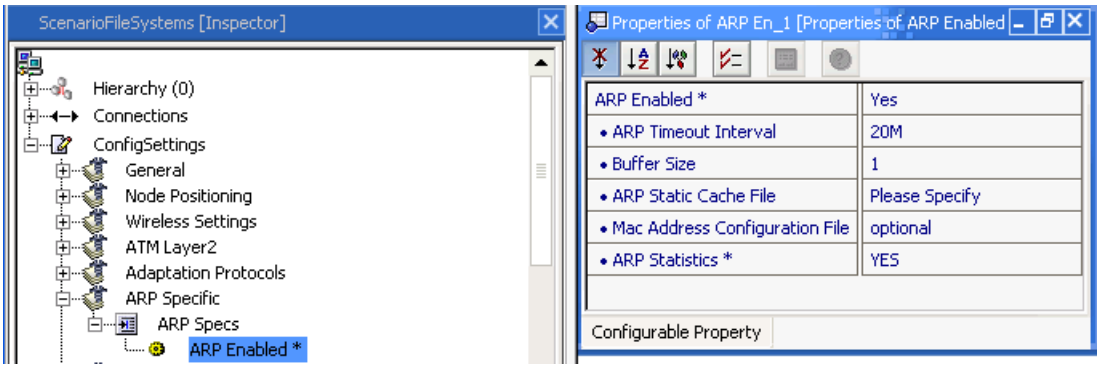


FIGURE 27. Setting ARP Parameters

7. Node 1 is a non-ATM node and only has an IP Interface. To enable node 1 to communicate with an ATM network, go to **Hierarchy (0) > Nodes > host 1 > Node configurations > Routing Protocol > Gateway Configuration**. In the Configurable Property window, set **Gateway Configuration** to **Yes** and set **Default Gateway** to the gateway's node ID or IP address, as shown in Figure 28.

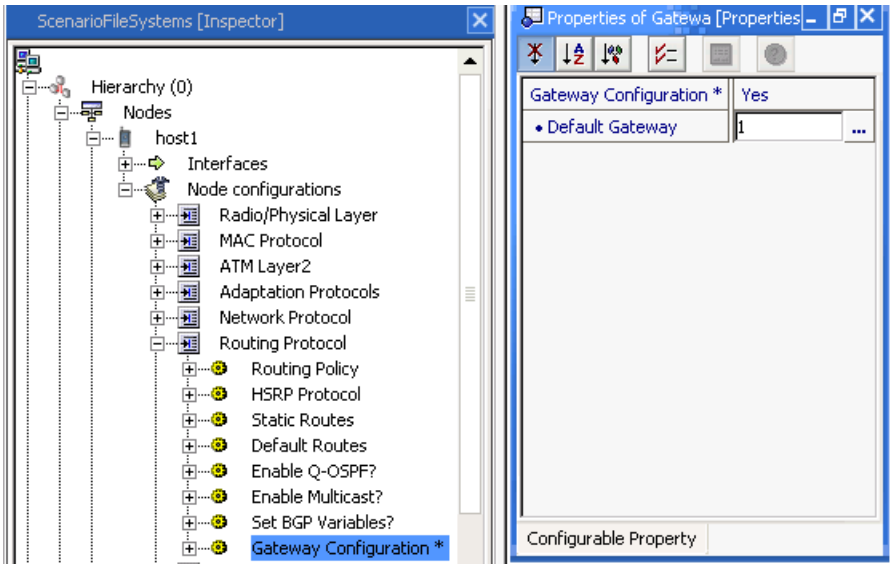


FIGURE 28. Configuring Default Gateway

8. For all nodes, go to **Hierarchy (0) > Nodes > host # > Interfaces > atminterface # > Interface configurations > ATM Layer2**. In the Configurable Property window, set **ATM Scheduler Statistics** and **ATM Layer2 Statistics** to **YES**, as shown in Figure 29.

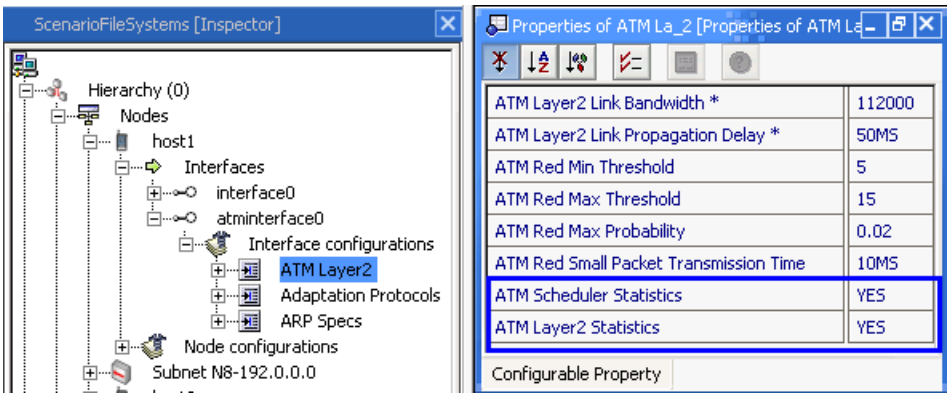


FIGURE 29. Enabling ATM Scheduler and Layer 2 Statistics

9. For all the ATM links, go to **Hierarchy (0) > Nodes > host # > Interfaces > atminterface # > Interface configurations > Adaptation Protocols**. In the Configurable Property window, set **Adaptation Layer Statistics** and **Signalling Stats** to **YES**, as shown in Figure 30.

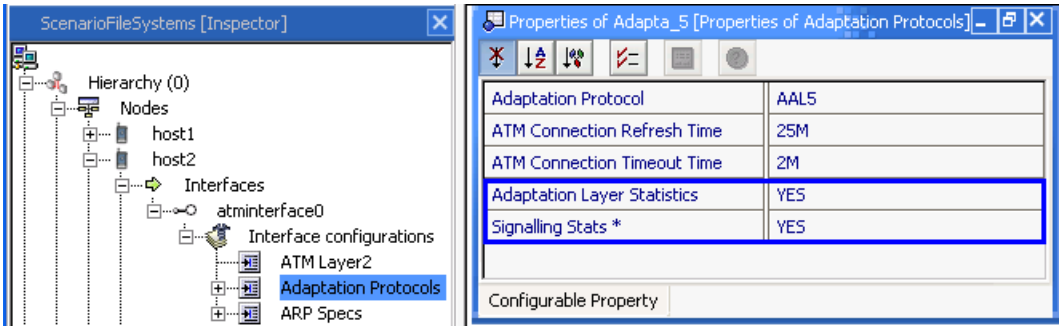


FIGURE 30. Enabling Adaptation Layer and Signalling Statistics

Bellman-Ford Routing Protocol

This is a generic Bellman-Ford (a.k.a. Ford Fulkerson) routing algorithm. This algorithm is the underlying mechanism of Routing Information Protocol (RIP), but this protocol is not RIPv2-compliant. It is a distance vector routing algorithm that uses the User Datagram Protocol (UDP) for control packet transmission.

Command Line Configuration

Specify Bellman-Ford in *.config as follows:

```
ROUTING-PROTOCOL        BELLMANFORD
```

Table 11 shows the Bellman-Ford configuration parameters.

TABLE 11. Bellman-Ford Parameters

Parameter	Description
TRACE-BELLMANFORD [YES | NO]	To enable or disable Bellman-Ford routing statistics. It's default value is Yes if TRACE-ALL parameter is set as YES otherwise traces for bellman-ford protocol would be disabled.

GUI Configuration

To enable Bellman Ford, go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Routing Protocol > Routing Policy > Routing Protocol for IPv4**. From the configurable property window, set **Routing Protocol for IPv4** to **Bellman Ford** as shown in Figure 31.

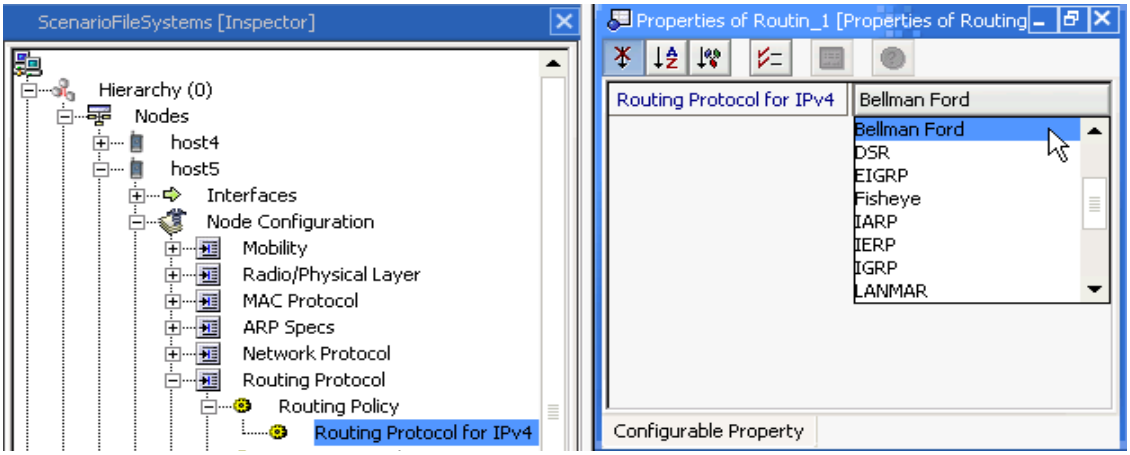


FIGURE 31. Enabling Bellman Ford Routing Protocol for a Node

Statistics

Table 12 shows the statistics collected by Bellman-Ford.

TABLE 12. Bellman-Ford Statistics

Statistic	Description
Number of periodic updates sent	Total number of periodic update messages sent.
Number of triggered updates sent	Total number of triggered update messages sent.
Number of route timeouts	Total number of route timeouts occurred.
Number of update packets received	Total number of update packets received.

Class-Based Queuing (CBQ)

CBQ is a link-sharing resource manager used to impose specific rules for the distribution of bandwidth among different agencies sharing the same physical link, and provide *assured bandwidth* service. CBQ controls the allocation of resources on a per-hop basis, as well as on an end-to-end (per flow) basis.

Command Line Configuration

Table 13 shows the parameters used to configure the CBQ Scheduler. All parameters are mandatory:

TABLE 13. CBQ Configuration Parameters

Parameter	Description
LINK-SHARING-STRUCTURE-FILE	Specifies the Link Sharing Structure File containing configuration information about weights, priorities, and link sharing between agencies. Per hop and end-to-end (per flow) link sharing and allocation of resources is specified in this file. The input format for the LINK-SHARING-STRUCTURE-FILE is as follows: <Node> <Interface> <Agency>, <descendant1>, <descendant2>, ..., <descendantN> Node - Specifies the node Id of the node for which this specification is valid, or ANY for all nodes. Interface - Specifies the IP address of the network interface for which this specification is valid, or ANY for all network interfaces on the node(s). Agency - Specifies the root agency name. descendant - If the <descendant> is a non-leaf agency, it consist of the following four fields: <Agency> <weight> <borrow> <efficient> If the <descendant> is leaf agency (or Application), it consists of the following four fields: <priority> <weight> <borrow> <efficient> priority: Specifies the priority of the agency. The default value is -1. weight: Specifies the weight for the agency. The default value is 1.0. borrow: Specifies whether the agency is allowed to borrow or not. The default value is FALSE. efficient: Specifies whether or not the agency is efficient. The default value is FALSE.
CBQ-GENERAL-SCHEDULER	Specifies the packet scheduler that CBQ uses to manage the queues. This option can be either PRR (Prioritized Round Robin) or WRR (Weighted Round Robin) Scheduling protocol.
CBQ-LINK-SHARING-GUIDELINE	Determines whether or not bandwidth is regulated by the link sharing scheduler. Bandwidth remains unregulated in the "ANCESTOR-ONLY" case when the class of traffic is under-limit, or its immediate ancestor is under-limit. Bandwidth remains unregulated in the "TOP-LEVEL" case when the class of traffic is under-limit, or at least one of its ancestors up to CBQ-TOP-LEVEL generations above, are under-limit.

TABLE 13. CBQ Configuration Parameters (Continued)

CBQ-TOP-LEVEL	This parameter determines the maximum number of generations to search, for under-limit ancestors. If the "TOPLEVEL" specified is greater than or equal to the maximum level in the link-sharing structure, then it will search up to the ROOT level.

GUI Configuration

Enabling CBQ

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Network Protocol > IP Queue Scheduler**. From the configurable property window, set **IP Queue Scheduler** as **Class Based Queuing**, as shown in Figure 32.

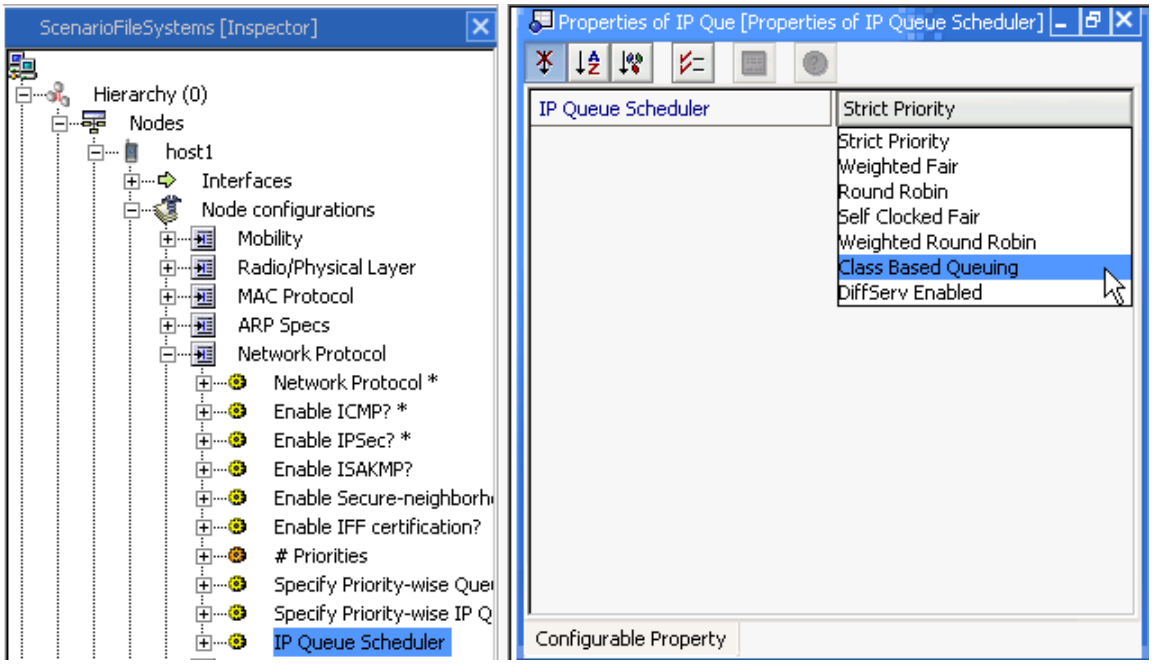


FIGURE 32. Enabling CBQ for a Node

Configuring CBQ Parameters

After enabling CBQ, configure CBQ parameters as shown in Figure 33.

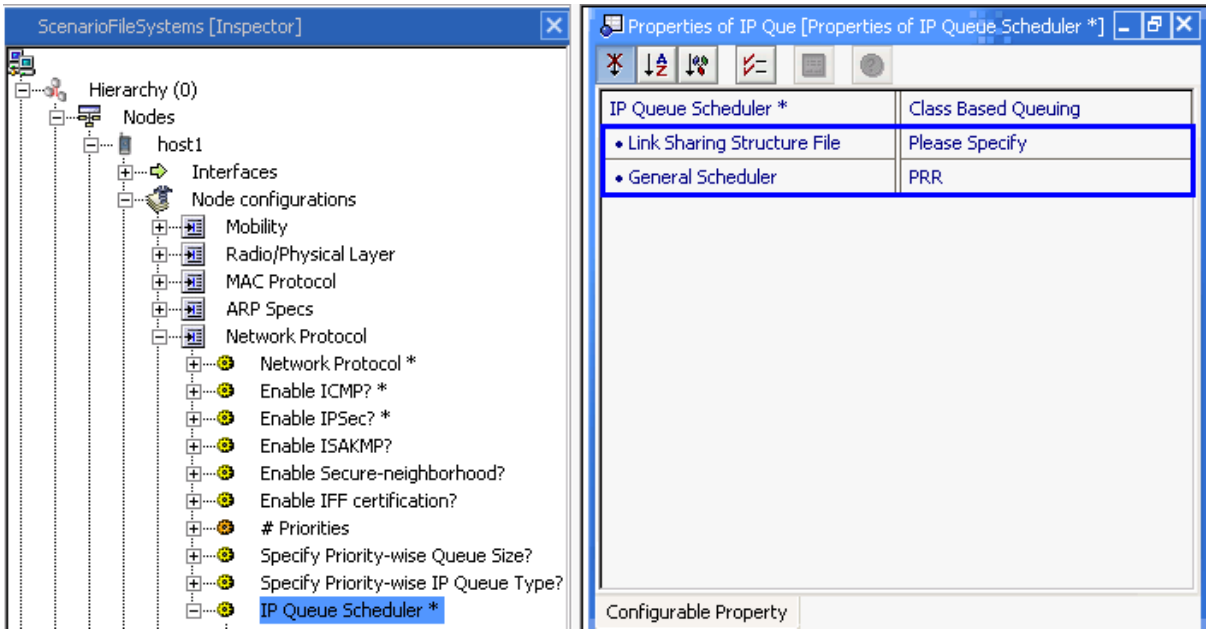


FIGURE 33. Configuring CBQ Parameters

Statistics

Table 14 shows the statistics collected for CBQ.

TABLE 14. CBQ Statistics

Statistic	Description
Packets Queued	Total number of packets queued at the interface
Packets Dequeued	Total number of packets dequeued at the interface
Number of Dequeue Requests	Total number of dequeue requests received.
Selection	Number of dequeue packets selected on the basis of General scheduler, link-sharing scheduler and forced packets.
Number of times Punished	Total number of times the packets were punished or suspended.
Max Extradelay	Maximum value of delay occurred due to packets punished/suspended.
Avg Extradelay	Average value of delay occurred due to packets punished/suspended.

Constant Bit Rate (CBR) Traffic Generator

CBR represents constant-bit-rate traffic. It is generally used to either fill in background traffic to affect the performance of other applications being analyzed, or to simulate generic multimedia traffic performance.

Command Line Configuration

To specify CBR traffic in the default.app application configuration file, make entries according to the following format:

```
CBR   <src> <dest> <items to send> <item size> <interval>
      <start time> <end time>
     [TOS <tos value> | DSCP <dscp value> | PRECEDENCE <precedence value>]
     [RSVP-TE]
```

Table 15 shows the CBR parameters and their descriptions.

TABLE 15. CBR Parameters

Parameter	Description
<src>	Specifies the client node's node identification or IP Address.
<dest>	Specifies the server node's node identification or IP Address.
<items_to_send>	Specifies the number of items to send. If it is set to 0, CBR will run until the specified <end time> or until the end of the simulation, whichever comes first.
<item_size>	Specifies the size of each item. The range is from 64 to 65023.
<interval>	Specifies the pause time between transmission of each item (inter-departure time).
<start_time>	Specifies when the transmissions should begin.
<end_time>	Specifies when the transmissions should end. If it is set to 0, CBR will run until all <items to send> is transmitted or until the end of simulation, whichever comes first.
TOS <tos value>	Specifies the contents of the 8-bit TOS value of the IP header for the packets generated. The range is from 0 to 255. This parameter is optional.
DSCP <dscp value>	Specifies the contents of the 6-bit DSCP value of the IP header for the packets generated. The range is from 0 to 63. This parameter is optional.
PRECEDENCE <precedence value>	Specifies the contents of the 3-bit Precedence value of the IP header for the packets generated. The range is from 0 to 7. This parameter is optional.
[RSVP-TE]	If the string RSVP-TE is at the end of a CBR line, this traffic generator uses RSVP-TE to transmit its packets over an MPLS network. This parameter is optional.

Notes:

1. If <items to send> and <end time> are both greater than 0, CBR will run until either <items to send> is done, <end time> is reached, or the simulation ends, whichever comes first.
2. At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP or TOS is not specified, a TOS value of 0 is assumed.

Examples

Table 16 shows examples of CBR configuration.

TABLE 16. CBR Examples

Configuration	Description
CBR 1 2 10 1460 1S 0S 600S	Node 1 sends node 2 ten items of 1460B each at the start of the simulation, and up to 600 seconds into the simulation. The inter-departure time for each item is 1 second.
CBR 1 2 0 1460 1S 0S 600S	Node 1 continuously sends to node 2 items of 1460B each at the start of the simulation up to 600 seconds into the simulation. The inter-departure time for each item is 1 second.
CBR 1 2 0 1460 1S 0S 0S	Node 1 continuously sends to node 2 items of 1460B each at the start of the simulation up to the end of the simulation. The inter-departure time for each item is 1 second.
CBR 1 2 0 1460 1S 0S 0S RSVP-TE	Node 1 continuously sends to node 2 items of 1460B each at the start of the simulation up to the end of the simulation. The inter-departure time for each item is 1 second. Here, RSVP-TE will be used to transport the data items instead of UDP.
CBR 1 192.168.0.8 10 1460 1S 5S 600S PRECEDENCE 5	Node 1 sends to node with source IP address 192.168.0.8 ten items of 1460B each 5 seconds into the simulation, and up to 600 seconds into the simulation. The inter-departure time for each item is 1 second. The Precedence value of each item is set to 5.
CBR 192.168.0.1 4 10 1460 1S 5S 600S DSCP 40	Node with source IP address 192.168.0.1 sends to node 4 ten items of 1460B each 5 seconds into the simulation, and up to 600 seconds into the simulation. The inter-departure time for each item is 1 second. The DSCP value of each item is set to 40 (express forwarding).
CBR 192.168.0.1 192.168.0.8 10 1460 1S 5S 600S TOS 175	Node with source address 192.168.0.1 sends to node with destination IP address 192.168.0.8 ten items of 1460B each 5 seconds into the simulation, and up to 600 seconds into the simulation. The inter-departure time for each item is 1 second. The TOS value of each item is set to 175.
CBR 1 192.168.0.8 10 1460 1S 5S 600S TOS 175 RSVP-TE	Node 1 sends to node with source IP address 192.168.0.8 ten items of 1460B each 5 seconds into the simulation, and up to 600 seconds into the simulation. The inter-departure time for each item is 1 second. The TOS value of each item is set to 175. Here, RSVP-TE will be used to transport the data items instead of UDP.

GUI Configuration

Configuring CBR Parameters

1. Go to **Connections > CBR [x] -> [y]**. In the configurable property window, configure the CBR properties as shown in Figure 34.

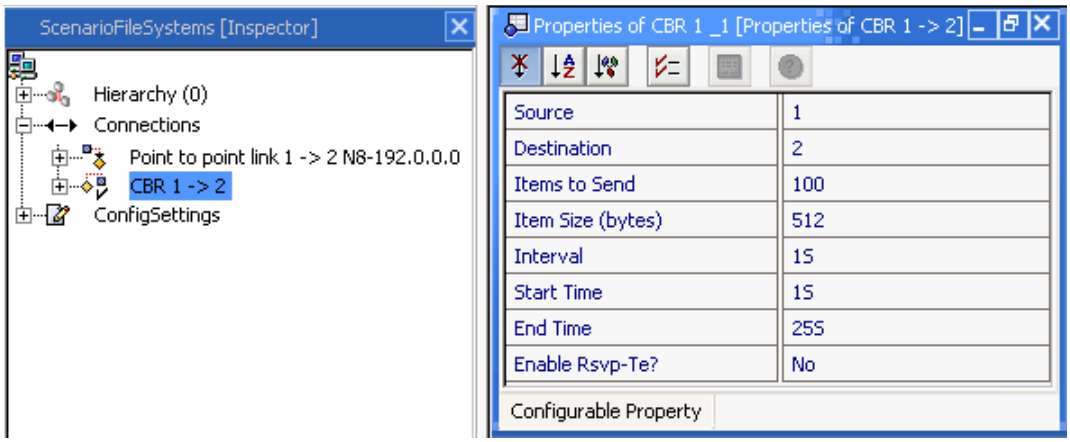


FIGURE 34. Configuring CBR Parameters

2. To set the priority, go to **Connections > CBR [x] -> [y] > Priority**. In the configurable property window, set **Priority** to the desired value from the list and set the dependent parameters for the selected priority type, as shown in Figure 35.

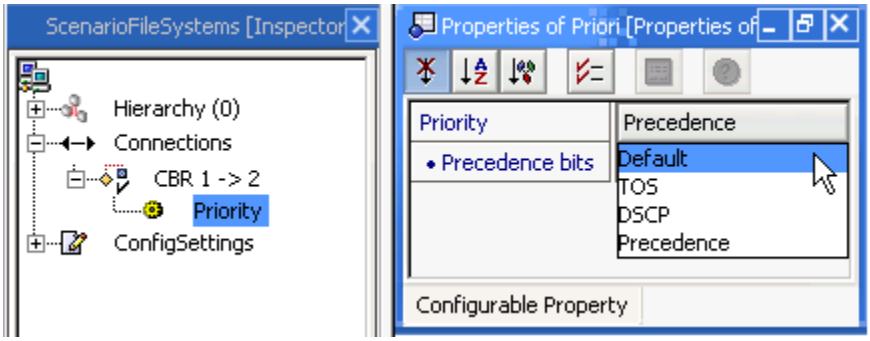


FIGURE 35. Setting Priority

Statistics

Table 17 shows the statistics collected for CBR.

TABLE 17. CBR Statistics

Statistic	Description
CBR Client	
Server address	Server address for CBR Client
First Packet Sent at (s)	Time in second when first packet was sent
Last Packet Sent at (s)	Time in second when last packet was sent

TABLE 17. CBR Statistics (Continued)

Statistic	Description
Session status	Current status of the session
Total Bytes Sent	Total number of bytes sent
Total Packets Sent	Total number of packets sent
Throughput (bits/s)	Client-side throughput
CBR Server	
Client address	Address of the client
First packet received at (s)	Time in second when first packet was received
Last packet received at (s)	Time in second when last packet was received
Session status	Current status of the session
Total Bytes Received	Total number of bytes received
Total Packets Received	Total number of packets received
Throughput (bits/s)	Server-side throughput
Average End-to-End Delay	Average delay for packet transmission between client and server.
Average Jitter	Average jitter for packet transmission between client and server.

Faults

Interface faults are one of two types, static and dynamic. Static faults have user-specified start and end times. Dynamic faults can be specified using Mean Time Before Failure (MTBF) and Mean Repair Time (MRT).

Both static and dynamic faults start with the keyword INTERFACE-FAULT followed by the interface address. The rest of the format differs.

Command Line Configuration

Static Faults

Syntax

```
INTERFACE-FAULT <interface address> <time fault occurs> <time fault is over>
[INTERFACE-CARD-FAULT <interface mac-address>]
```

For example,

```
INTERFACE-FAULT 192.0.0.2 0S 10S [INTERFACE-CARD-FAULT 5C-AA-FE-23-4F-C2]
```

Note: The optional replacement MAC address does not apply to dynamic faults.

Table 18 describes the Static Fault parameters.

TABLE 18. Static Fault Parameters

Parameter	Description
interface address	It refers to the interface address on which fault is configured. Note: This is a mandatory parameter.
time fault occurs	It refers to the time when the failure starts. Note: This is a mandatory parameter.
time fault is over	It refers to the time when the fault is over. Note: This is a mandatory parameter.
INTERFACE-CARD-FAULT <interface mac-address>	INTERFACE-CARD-FAULT and replacement is supported only when ARP is enabled and is only enabled for static faults. Note: It becomes mandatory to put interface mac-address if INTERFACE-CARD-FAULT is enabled.

Dynamic Faults

Syntax

```
INTERFACE-FAULT <interface address> <repetitions> <start time> <MTBF> <repair
time>
```

For example,

```
INTERFACE-FAULT 192.0.1.1 REPS DET 1 START DET 1S MTBF UNI 1S 2S DOWN EXP 1S
```

Note: In this example, the repetitions and start time are deterministic (keyword DET), while the Up Time is uniform (keyword UNI with two parameters) and the Down Time is exponential (keyword EXP with one parameter). The formatting of DET, UNI, and EXP should be the same as in the Traffic-Gen application.

Table 19 describes the Random Fault parameters.

TABLE 19. Random Fault Parameters

Parameters	Description
interface address	It refers to the interface address on which fault is configured. Note: It is a mandatory parameter.
repetitions	It specifies how many times the interface will fail. A value of 0 means that it will fail continuously. A number greater than 0 indicates a specific number of failures. Note: It is a mandatory parameter.
start { Deterministic | Uniform | Exponentially distributed }	The time at which the failure countdown begins. If, for example, you want the system to function correctly for 100S and then start suffering failures, the start time would be set to 100S. That doesn't mean that the first error will happen at exactly 100S, just that the first 100S will be error free. Note: It is a mandatory parameter.
MTBF { Deterministic | Uniform | Exponentially distributed }	It specifies how long the system will be up and running before suffering a failure. Note: It is a mandatory parameter.
MRT { Deterministic | Uniform | Exponentially distributed }	It specifies how long the system is down before coming back on line. Note: It is a mandatory parameter.

The Start, Up, and Down Times can be specified as Deterministic, Uniform, or Exponentially distributed. If deterministic, you must specify the amount of time for the property. If uniform, specify high and low values, and the system chooses a value in between those extremes using a uniform random distribution. If uniform, specify the mean values, and the system chooses a value from an exponential distribution with that mean.

GUI Configuration

Configuring Faults

Go to **Hierarchy (x) > Nodes > host(x) > Faults**. Right click on **Faults** and select **Add** as shown in Figure 36.

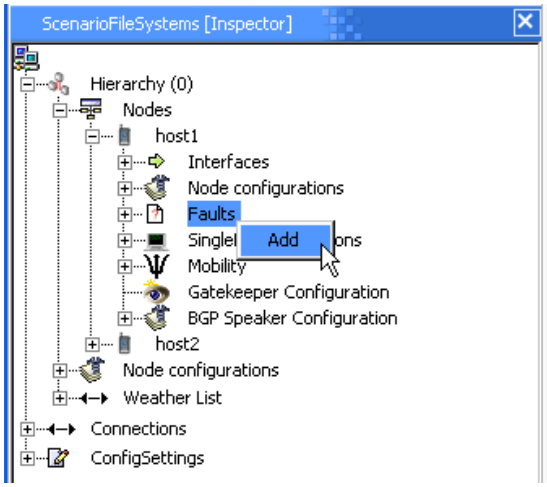


FIGURE 36. Configuring a Fault

Configuring Fault Type

When Fault is enabled, set the **Fault Type** and configure Fault parameters as shown in Figure 37.

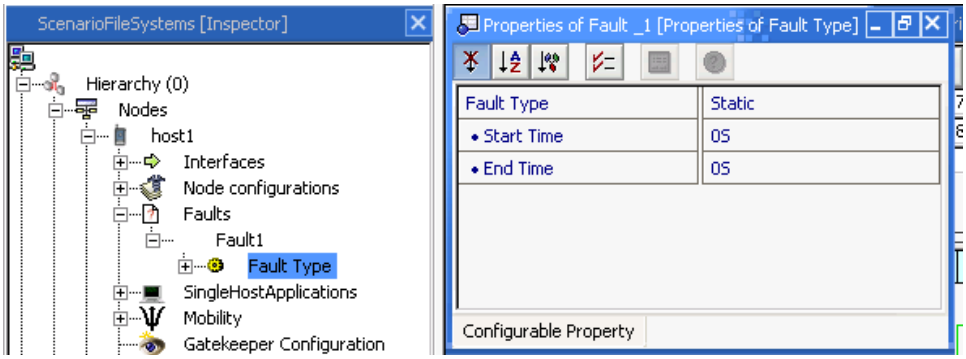


FIGURE 37. Configuring Static Fault

Enabling Card Faults

To enable a card fault, set **Specify card fault** to **Yes** as shown in Figure 38.

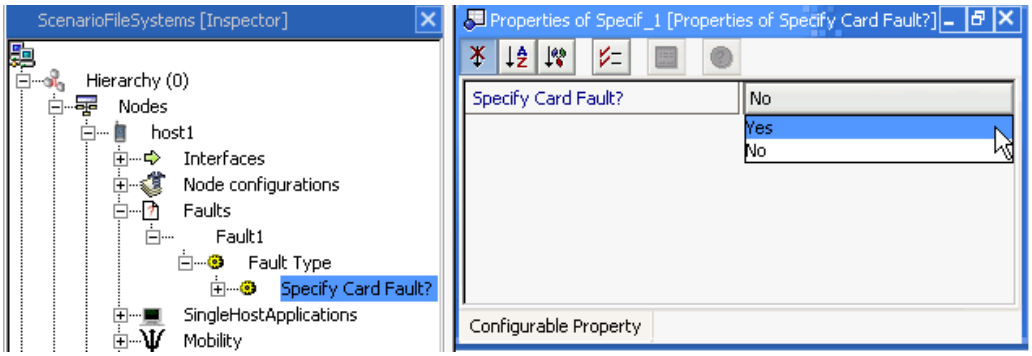


FIGURE 38. Enabling a Card Fault

Specifying an Optional MAC Address

After specifying card fault to Yes, configure the optional replacement MAC address as shown in Figure 39.

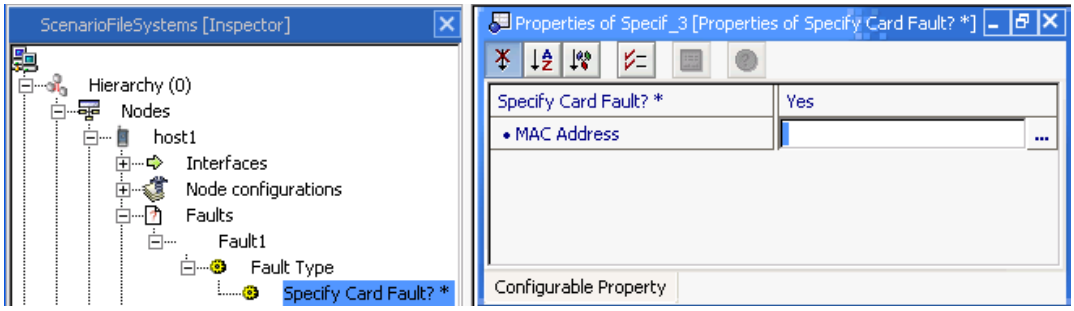


FIGURE 39. Optional MAC Address

Configuring a Random Fault

1. Set the **Fault Type** as **Random**, as shown in Figure 40.

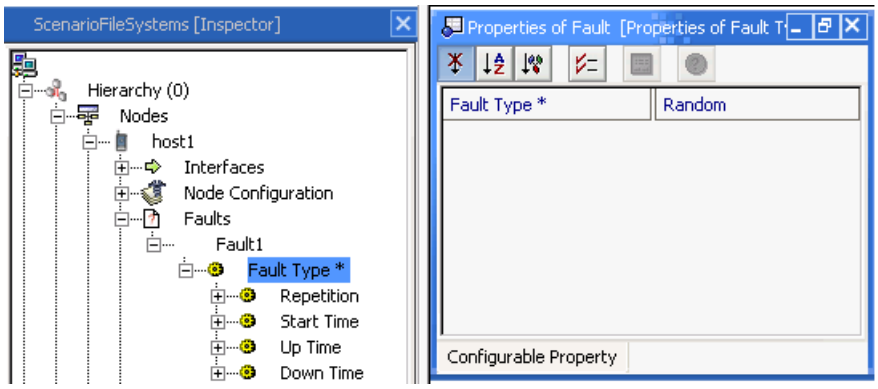


FIGURE 40. Configuring Random Fault

2. Set the dependent parameters, as shown in Figure 40.

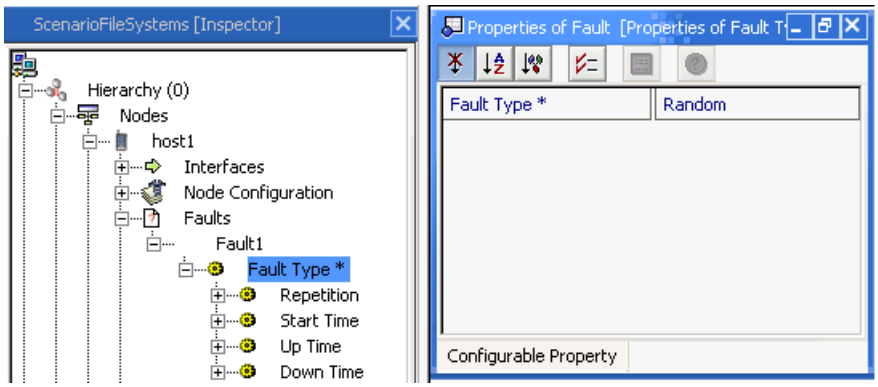


FIGURE 41. Configuring Parameters fro Random Fault Type

Statistics

There are no statistics generated for Faults.

File Transfer Protocol (FTP)

FTP represents the File Transfer Protocol server and client. The size of the items sent is taken from network traces, with distributions taken from the TCPlib library. If the number of items to send is specified as 0, then TCPlib also determines the number of items to send.

Command Line Configuration

To specify the FTP application in the application configuration (.app) file, use the following format:

```
FTP    <src> <dest> <items to send> <start time>
```

The FTP parameters are described in Table 20

TABLE 20. FTP Parameters

Parameter	Description
<src>	Client node's ID or IP Address.
<dest>	Destination node's ID or IP Address.
<items_to_send>	Number of items to send. If this is specified as 0, the number of items to send is determined by TCPlib.
<start_time>	Time when the transmissions should begin.

Note: If you want to be able to configure the size of packets, you should use FTP/Generic, rather than FTP. File Transfer Protocol/Generic represents a more configurable model of the File Transfer Protocol. The size of the items sent is not determined by network traces, instead it is user-specified.

GUI Configuration

To configure FTP parameters, go to **Connections > FTP [x] -> [y]**. In the Configurable Property window, set the FTP parameters, as shown in Figure 42.

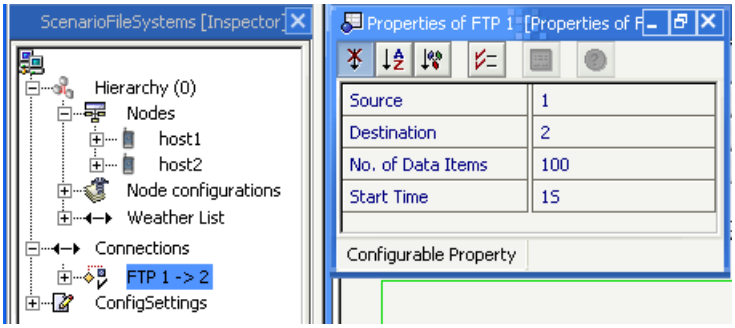


FIGURE 42. Configuring FTP Parameters

Statistics

Table 21 shows the client and server statistics collected by FTP.

TABLE 21. FTP Statistics

Statistics	Description
FTP Client	
Server Address	Specifies the server address
First Packet Sent at (s)	Specifies the time when the first packet is sent
Last Packet Sent at (s)	Specifies the time when the last packet is sent
Session Status	Specifies whether the session status is open or closed
Total Bytes Sent	Specifies the total number of bytes sent
Total Bytes Received	Specifies the total number of bytes received
Throughput (bits/s)	Specifies the total throughput of the client
FTP Server	
Client Address	Specifies the client address
First Packet Sent at (s)	Specifies the time when the first packet is sent
Last Packet Sent at (s)	Specifies the time when the last packet is sent
Session Status	Specifies whether the session status is open or closed
Total Bytes sent	Specifies the total number of bytes sent
Total Bytes Received	Specifies the total number of bytes received
Throughput (bits/s)	Specifies the total throughput of the server

File Transfer Protocol/Generic (FTP/Generic)

FTP/Generic represents a more configurable model of the File Transfer Protocol. The size of the items sent is not determined by network traces, instead it is user-specified.

Command Line Configuration

To specify FTP/Generic traffic in the application configuration (.app) file, specify the parameters according to the following format:

```
FTP/GENERIC <src> <dest> <items to send> <item size>
            <start time> <end time>
           TOS <tos value> | DSCP <dscp value> | PRECEDENCE <precedence value>]
```

Table 22 shows the parameters for the FTP/Generic model.

TABLE 22. FTP/Generic Parameters

Parameter	Function
<src>	Refers to the client node's ID or IP address. This is a mandatory parameter.
<dest>	Refers to the server node's ID or IP address. This is a mandatory parameter.
<items_to_send>	Specifies the number of items to send. If items to send is set to 0, FTP/Generic will run until the specified <end time> or until the end of the simulation, whichever comes first. This is a mandatory parameter.
<item_size>	Specifies the size of each item in bytes. This is a mandatory parameter.
<start_time>	Indicates when the transmissions should begin. This is a mandatory parameter.
<end_time>	Indicates when the transmissions should end. If end time is set to "0", FTP/Generic will run until all items to send have been sent, or until the end of the simulation, whichever comes first. This is a mandatory parameter.
[TOS <tos value>]	Specifies the contents of the 8-bit TOS value of the IP header for the packets generated. The range is from 0 to 255. Default value is 0. This entry is optional.
[DSCP <dscp value>]	Specifies the contents of the 6-bit DSCP value of the IP header for the packets generated. The range is from 0 to 63. This entry is optional.
[PRECEDENCE <precedence value>]	Specifies the contents of the 3-bit Precedence value of the IP header for the packets generated. The range is from 0 to 7. Default value is 0. This entry is optional.

Notes:

1. If <items to send> and <end time> are both greater than 0, FTP/Generic will run until either <items to send> is done, <end time> is reached, or the simulation ends, whichever comes first.
2. At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP or TOS is not specified, a TOS value of 0 is assumed.

Table 23 shows examples of the FTP/Generic configuration.

TABLE 23. FTP/Generic Examples

Configuration	Description
FTP/GENERIC 1 2 10 1460 0S 600S	Node 1 will send ten 1460-byte items to node 2, and begin these transmissions at the start of the simulation. If the 10 items are sent before the 600 simulation seconds elapse, no other items will be sent.
FTP/GENERIC 1 2 0 1460 0S 0S	Node 1 continuously sends node 2 items of 1460B each at the start of the simulation until the end of the simulation.
FTP/GENERIC 1 2 10 1460 0S 600S PRECEDENCE 4	Node 1 will send ten 1460-byte items to node 2 with IP Precedence of 4 for each data item, and begin these transmissions at the start of the simulation. If the 10 items are sent before the 600 simulation seconds elapse, no other items will be sent.
FTP/GENERIC 1 2 10 1460 0S 600S DSCP 40	Node 1 will send ten 1460-byte items to node 2 with IP DSCP of 40 for each data item, and begin these transmissions at the start of the simulation. If the 10 items are sent before the 600 simulation seconds elapse, no other items will be sent.
FTP/GENERIC 1 2 10 1460 0S 600S TOS 170	Node 1 will send ten 1460-byte items to node 2 with IP TOS of 170 for each data item, and begin these transmissions at the start of the simulation. If the 10 items are sent before the 600 simulation seconds elapse, no other items will be sent.

GUI Configuration

1. To configure FTP/Generic parameters, go to **Connections > FTP/GENERIC [x] -> [y]**. In the Configurable Property window, set the FTP/Generic parameters, as shown in Figure 43.

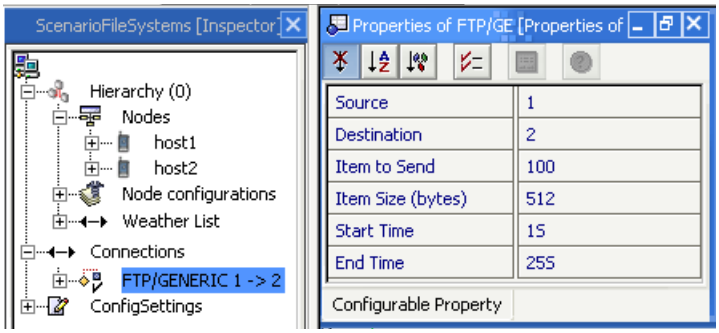


FIGURE 43. Configuring FTP/Generic Parameters

2. To set the priority, go to **Connections > FTP/GENERIC [x] -> [y] > Priority**. In the configurable property window, set **Priority** to the desired value from the list and set the dependent parameters for the selected priority type, as shown in Figure 44.

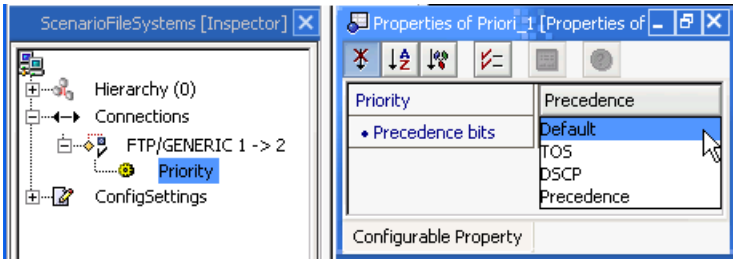


FIGURE 44. Setting Priority

Statistics

The FTP/Generic statistics are listed inTable 24.

TABLE 24. FTP/Generic Statistics

Statistic	Description
FTP/Generic Client	
Server Address	Specifies the server address
First Packet Sent at (s)	Specifies the time when the first packet is sent
Last Packet Sent at (s)	Specifies the time when the last packet is sent
Session Status	Specifies whether the session status is open or closed
Total Bytes Sent	Specifies the total number of bytes sent
Throughput (bits/s)	Specifies the total throughput of the client
FTP/Generic Server	
Client Address	Specifies the client address
First Packet Sent at (s)	Specifies the time when the first packet is sent
Last Packet Sent at (s)	Specifies the time when the last packet is sent
Session Status	Specifies whether the session status is open or closed
Total Bytes Received	Specifies the total number of bytes received
Throughput (bits/s)	Specifies the total throughput of the server

File-based Node Placement Model

In the file-based node placement model, the initial node positions are read from a node position file

Command Line Configuration

To select this model, set the parameter NODE-PLACEMENT in the configuration file as follows:

```
NODE-PLACEMENT FILE
```

The parameters for the file-based node placement model are described in Table 25

TABLE 25. File-based Node Placement Parameters

Parameter	Description
NODE-POSITION-FILE <filename>	Name of the node position file. **Note**: The same file is also used if the file-based mobility model is used. The format of the node position file is described below. This parameter is mandatory.

Format of the Node Position File

The node position file can be used for file-based mobility also, in which case it contains the node positions at different simulation times.

Each line in the node position file has the following format:

```
<nodeID> <simulation-time> <position>
```

where

- <nodeID> : Node identifier.
- <simulation-time> : Simulation time. For the initial node position, this should be 0. See Section 7.2.2 of *QualNet 4.5 User's Guide* for the format for specifying time.
- <position> : Node position. The node position is specified as the coordinates in Cartesian or Lat-Lon-Alt system, optionally followed by the orientation (azimuth and elevation). Specifying node orientation is optional and is assumed to be (0.0 0.0) when not specified. See Section 7.2.3 of *QualNet 4.5 User's Guide* for the format for specifying node positions.

Notes:

1. For each node, the node positions should be sorted (in ascending order) by simulation time.
2. Each node position specification should be on a single line by itself.
3. Comments can be entered any where in the node position file.

Example

The following lines show a segment of a node position file:

```
1 0 (35.130587432702, -116.72249286971918, 0.0) 0 0
1 10S (35.12977099236641, -116.53095393408505, 0.0) 0.0 0.0
1 20S (35.12977099236641, -116.39738452458609, 0.0)
...
1 60S (35.36132315521628, -116.2700276457615, 0.0)
1 70S (35.465648854961835, -116.26692138042432, 0.0) 0.0 0.0
2 0 (35.16793886702846, -116.72149633406089, 0.0) 0 0
2 10S (35.16897959183674, -116.58344129312579, 0.0) 30.0 0.0
2 20S (35.16938775510204, -116.4518964383234, 0.0) 30.0 45.0
...
```

Statistics

There are no statistics collected for this model.

First-In First-Out (FIFO) Queue

FIFO queue accepts packets until it is full (the packets occupying the queue total an aggregate number of bytes that prevent the next packet from being admitted). The size of the queue is specified by the IP-QUEUE-PRIORITY-QUEUE-SIZE parameter. At that point, all packets that arrive at the queue are dropped, until packets are dequeued, and space becomes available.

Command Line Configuration

To select FIFO as the network queue discipline, place the following entry in *.config:

```
IP-QUEUE-TYPE      FIFO
```

Table 26 shows the FIFO configuration parameters.

TABLE 26. FIFO Parameters

Parameter	Description
IP-QUEUE-PRIORITY-QUEUE-SIZE <queue-size>	Queue size (in bytes). The default value is 150000.

GUI Configuration

Enabling FIFO

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Network Protocol > Specify Priority-wise IP Queue Type? > IP Queue Type**. From the configurable property window, set **IP Queue Type** as **FIFO** as shown in Figure 45.

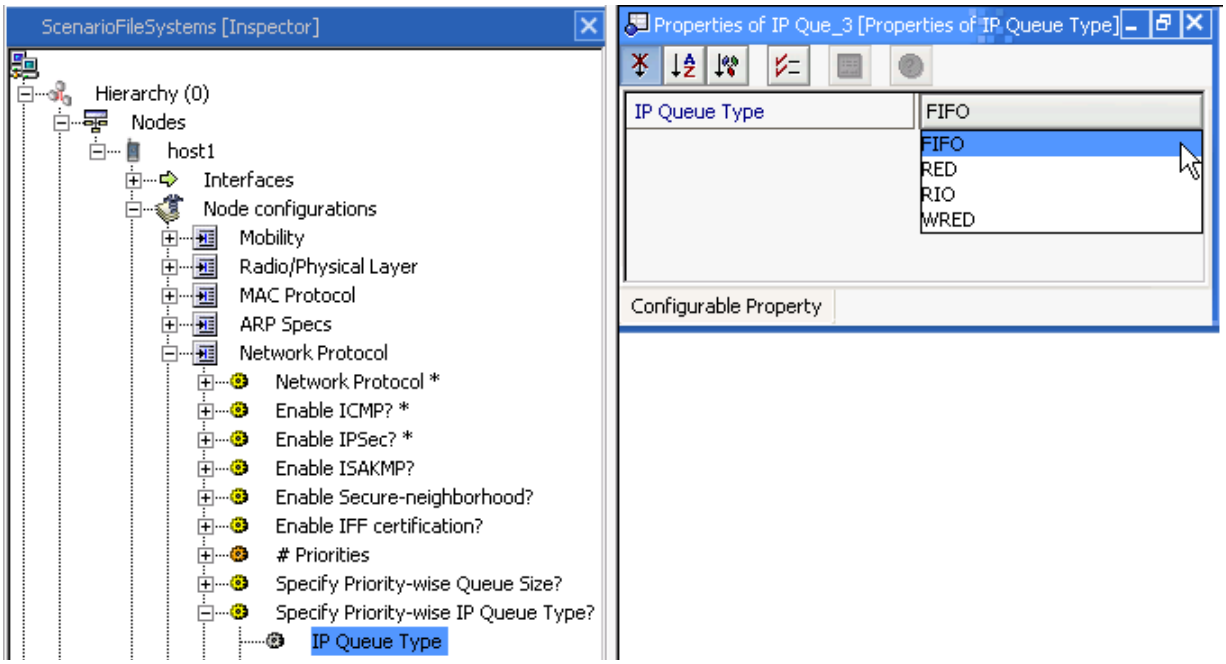


FIGURE 45. Enabling FIFO for a Node

Statistics

Table 27 shows the statistics collected by the FIFO: model.

TABLE 27. FIFO Statistics

Statistic	Description
Total Packets Queued	Total number of packets queued
Total Packets Dequeued	Total number of packets dequeued
Total Packets Dropped	Total number of packets dropped
Total Packets Dropped Forcefully	Total number of packets dropped forcefully
Average Queue Length	Average queue length in bytes
Average Time in Queue	Average time spent by packets in the queue (in seconds)
Longest Time in Queue	Longest time spent in queue by a packet (in seconds)
Peak Queue Size (bytes)	Peak queue size

Grid Node Placement Model

In the grid node placement model, the terrain is divided into a number of squares. One node is placed at each grid point, starting at the origin or the south-west corner. The size of the squares is determined by a user-specified parameter (see Table 28).

Note: This model can be used only if the number of nodes in the scenario is a square of an integer (4, 9, 16, ...).

Command Line Configuration

To select this model, set the parameter `NODE-PLACEMENT` in the configuration file as follows:

```
NODE-PLACEMENT GRID
```

The parameters for the grid node placement model are described in Table 28

TABLE 28. Grid Node Placement Parameters

Parameter Name	Description
`GRID-UNIT <unit>`	Length of a side (in meters or degrees) of a square in the grid. If Cartesian coordinate system is used, the unit is meters. If Lat-Lon-Alt system is used, the unit is degrees. This parameter is mandatory.

Statistics

There are no statistics collected for this model.

HyperText Transfer Protocol (HTTP)

HTTP simulates single-TCP connection web servers and clients. The size of HTTP items retrieved, number of items per Web page, think time, and user browsing behavior are taken from packet traces of HTTP network conversations.

Command Line Configuration

Configuring HTTP traffic in the application configuration file (.app) consists of two parts: server configuration and client configuration.

To configure an HHTP server, use the following format:

```
HTTPD   <server-ID>
```

To configure an HTTP client, use the following format:

```
HTTP  <client-ID> <num of servers> <server-1> ... <server-n> <start> <thresh>
```

Table 29 shows the HTTP parameters for clients and servers. All parameters are mandatory.

TABLE 29. HTTP Parameters

Parameter	Description
Servers	
<server-ID>	Node ID of the server pages.
Clients	
<client-ID>	Node ID of the client
<num of servers>	Number of servers.
<server-1>	Node ID of the first server.
<server-n>	Node ID of the nth server.
<start>	Start time for when the client begins requesting pages.
<thresh>	Ceiling (specified in units of time) on the amount of “think time” allowed for a client. The network-traced based amount of time modulo this threshold is used to determine think time.

GUI Configuration

In QualNet GUI, HTTP is a configured as a single host application. Perform the following steps to configure HTTP:

1. Go to **Hierarchy (x) > Nodes > host # > SingleHostApplications**. Right click on **SingleHostApplications** and select **Add** from the pop-up menu, as shown in Figure 46.

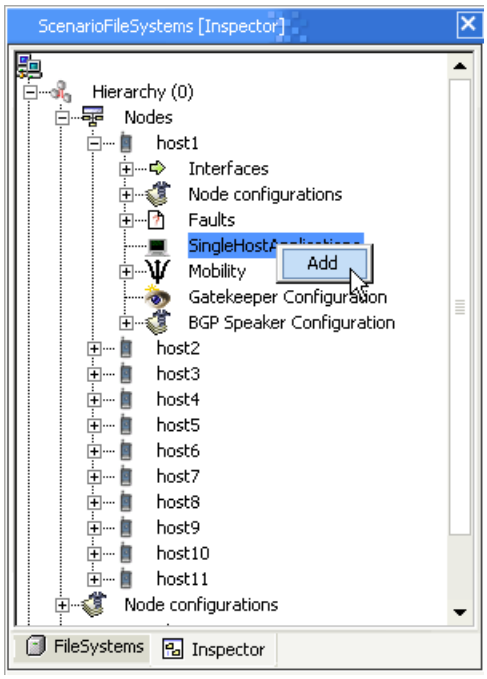


FIGURE 46. Adding a Single Host Application

2. The Single Host Application wizard is launched (see Figure 47). Select HTTP from the list and click on Finish.

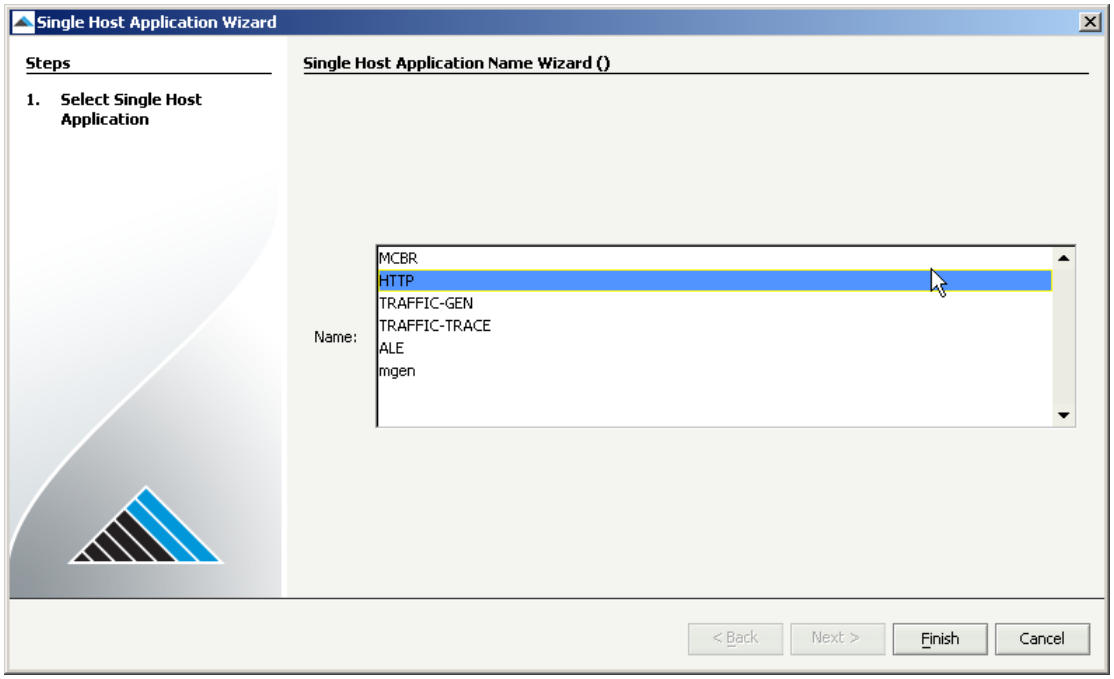


FIGURE 47. Single Host Application Wizard

3. Go to **Hierarchy (x) > Nodes > host # > SingleHostApplications > HTTP**. In the Configurable Property window, click on the value field for the parameter **List of Servers**. Click on the edit field button, as shown in Figure 48.

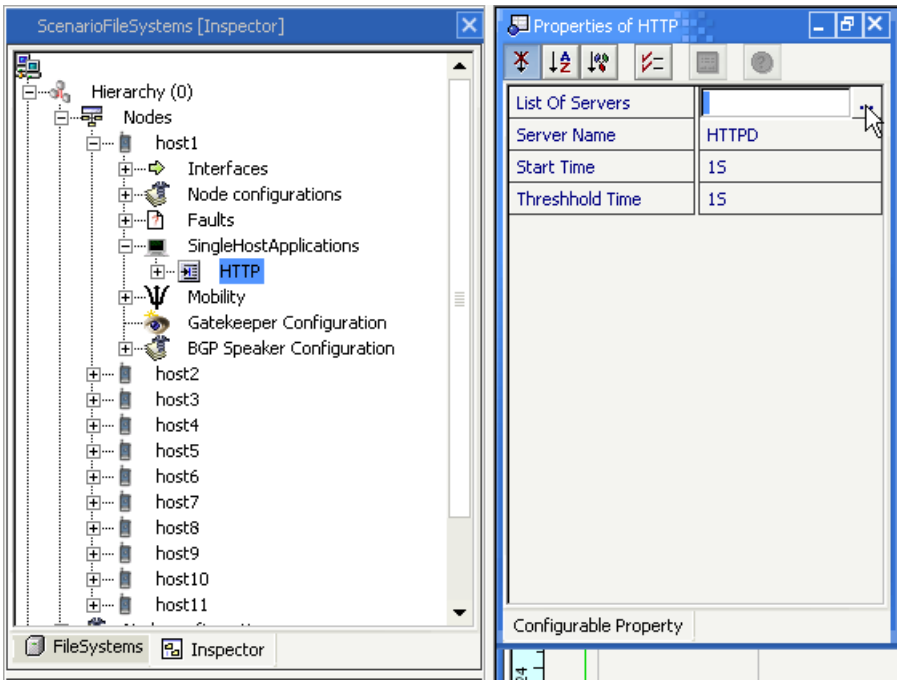


FIGURE 48. Editing List of Servers

4. This launches an editor to specify servers. Select the servers by checking the corresponding boxes and click on OK, as shown in Figure 49.

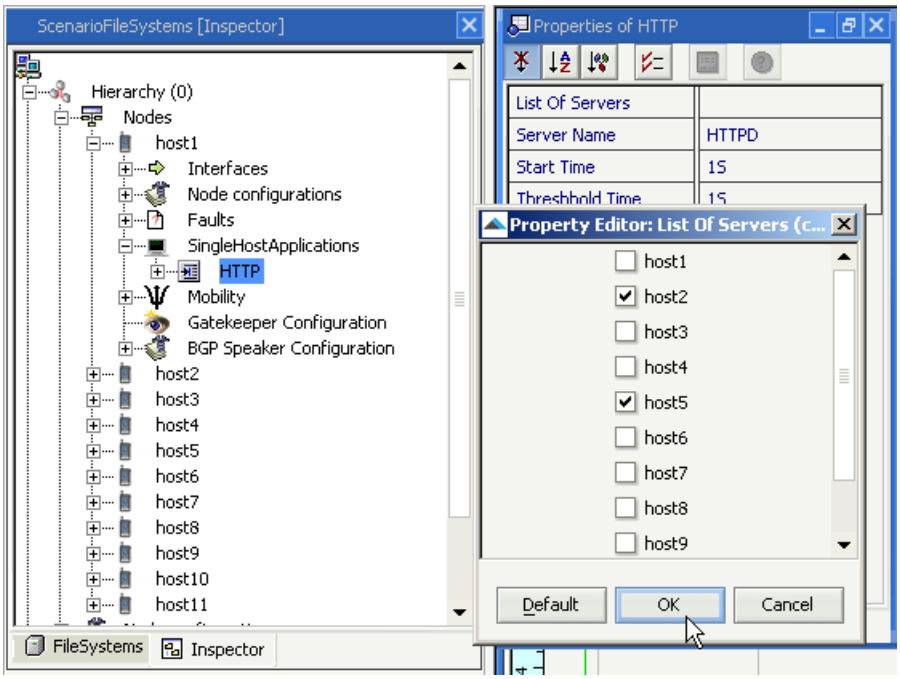


FIGURE 49. Specifying Servers

5. Configure the remaining HTTP parameters, as shown in Figure 50.

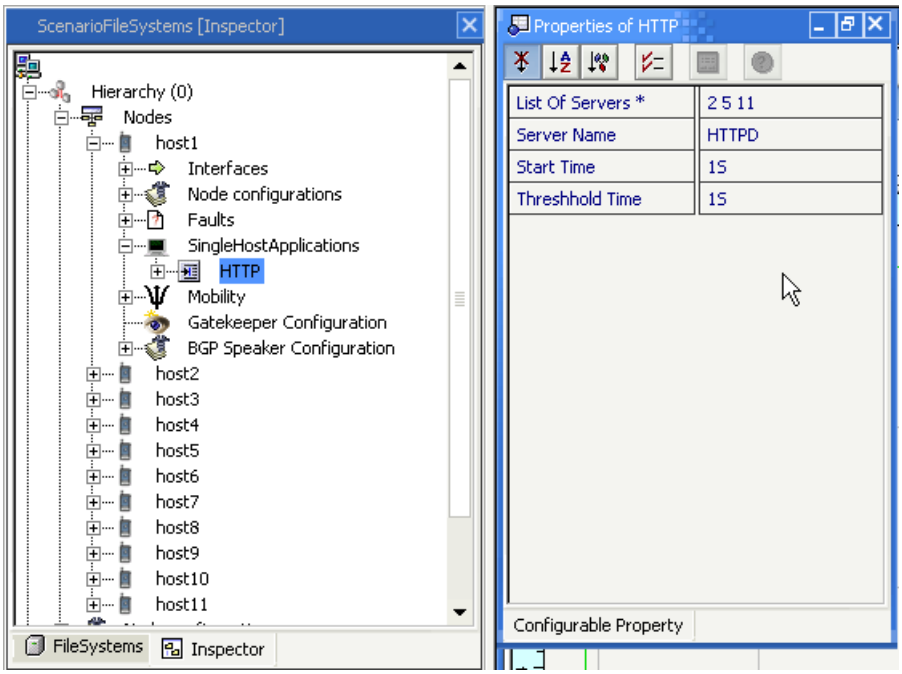


FIGURE 50. Configuring HTTP Parameters

Statistics

The HTTP statistics are listed in Table 30.

TABLE 30. HTTP Statistics

Statistic	Description
HTTP Client	
Number of Connections	Total number of connections taken by the client.
Average Connection Length (s)	The average time of connection length.
Average Number of Pages per Connection	Specifies the average number of pages per connection.
Average Number of Bytes Received per Connection	Specifies the average number of bytes received per connection.
Average Number of Bytes Sent per Connection	Specifies the average number of bytes sent per connection.
Average Page Request Time (s)	Specifies the average page request time.
Longest Page Request Time (s)	Specifies the longest page request time.
HTTP Server	
Statistics	Description
Client Address	Specifies the client address

TABLE 30. HTTP Statistics

Statistic	Description
Connection Length (s)	Specifies the length of the connection.
Pages Sent	Specifies the total number of pages sent.
Bytes Sent	Specifies the total number of bytes sent
Bytes Received	Specifies the total number of bytes received
Throughput (bits/sec)	Specifies the total throughput of the server

Sample Scenario

Scenario Description

There are HTTP servers on nodes 2, 5, 8, and 11. There is an HTTP client on node 1. This client chooses between servers 2, 5, and 11 when requesting web pages. It begins browsing 10 seconds after the start of simulation and remains idle for at most 2 minutes at a time.

Command Line Configuration

To configure the above scenario for the command line, include the following lines in the application configuration (.app) file.

```
HTTPD 2
HTTPD 5
HTTPD 8
HTTPD 11
HTTP 1 3 2 5 11 10S 120S
```

Internet Control Message Protocol (ICMP)

ICMP is an integral part of IP. It allows a router or destination host to communicate with the source, typically to report an error in IP datagram processing.

Command Line Configuration

To enable ICMP, include the following parameter in the scenario configuration (.config) file:

```
ICMP     YES
```

By default, ICMP is not enabled.

The ICMP configuration parameters are described in Table 31:.

TABLE 31. ICMP Configuration Parameters

Parameter	Description
ICMP-ROUTER-ADVERTISEMENT-MIN-INTERVAL <value>	Specifies the minimum time interval for ICMP router advertisement. The default value is 450S.
ICMP-ROUTER-ADVERTISEMENT-MAX-INTERVAL <value>	Specifies the maximum time interval for ICMP router advertisement. The default value is 600S.
ICMP-ROUTER-ADVERTISEMENT-LIFE-TIME <value>	Specifies the life time of ICMP router advertisement. The default value is 1800S.
ICMP-MAX-NUM-SOLICITATION <value>	Specifies the maximum number of solicitations. The default value is 3.
TRACE-ICMP [YES | NO]	Enables the trace of ICMP. The default value is NO.
ICMP-STATISTICS [YES | NO]	Enables or disables the ICMP statistics. The default value is NO.

GUI Configuration

IPv4 must be enabled to use ICMP. For more information, see the IPv4 section in this model library.

Enabling ICMP

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Network Protocol > Enable ICMP?**. From the configurable property window, set **Enable ICMP?** to **Yes** as shown in Figure 51.

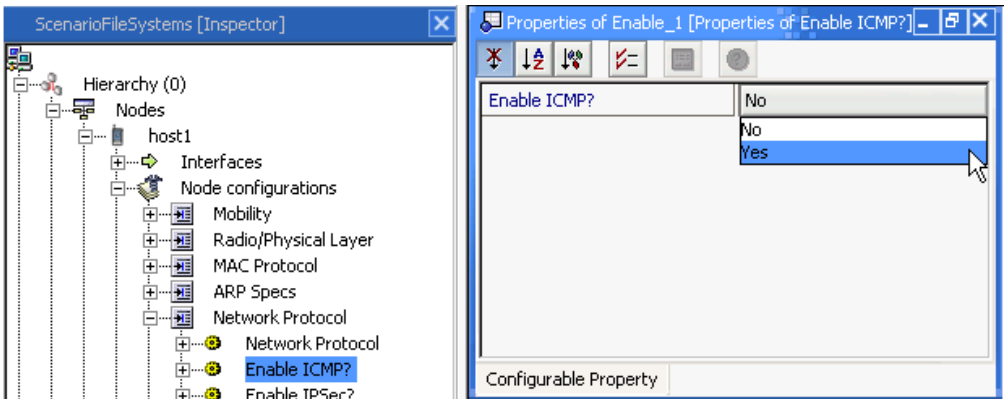


FIGURE 51. Enabling ICMP for a Node

Configuring General ICMP Parameters

When ICMP is enabled, a set of its parameters is displayed. Configure ICMP parameters as shown in Figure 52.

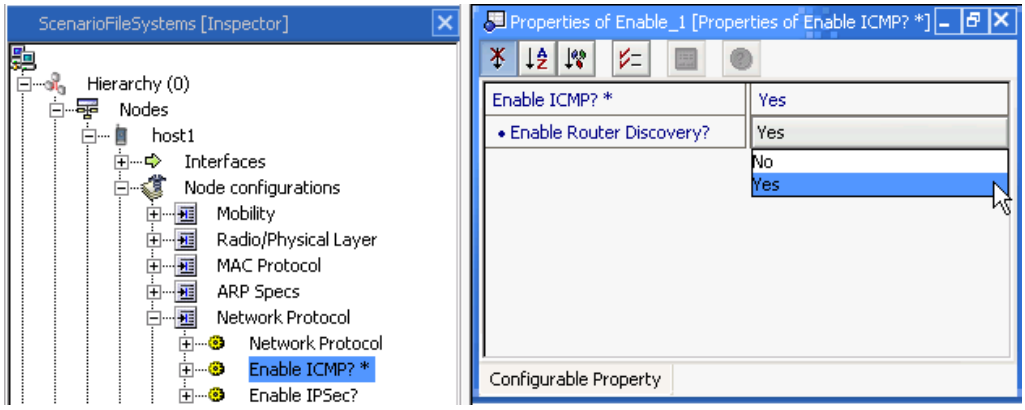


FIGURE 52. Configuring General Parameters of ICMP

Enabling Statistic Collection

Go to **ConfigSettings > Statistics > Statistics**. From the configurable property window, enable **ICMP statistics** as shown in Figure 53.

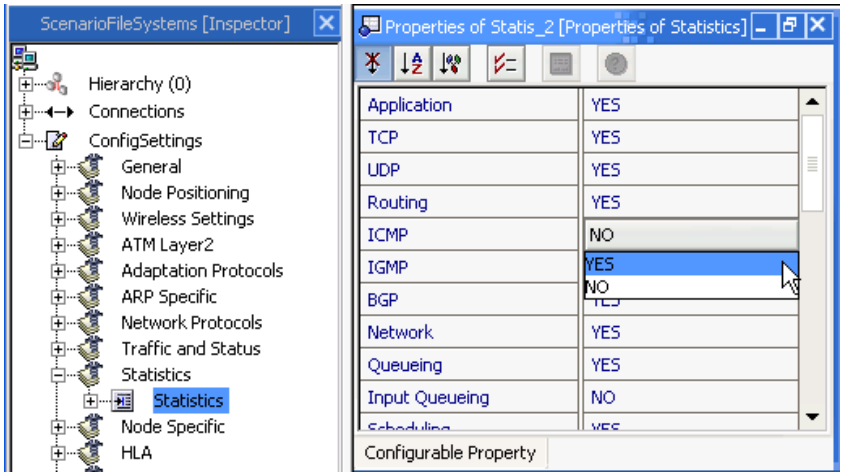


FIGURE 53. Enable Statistic Collection

Statistics

The ICMP statistics are shown in Table 32.

TABLE 32. ICMP Statistics

Statistic	Description
Advertisements Generated	Specifies the number of advertisements generated
Advertisements Received	Specifies the number of advertisements received
Solicitations Generated	Specifies the number of solicitations generated
Solicitations Received	Specifies the number of solicitations received

Internet Control Message Protocol version 6 (ICMPv6)

Internet Control Message Protocol version 6 (ICMPv6) is used by IPv6 nodes to report errors encountered in processing packets, and to perform other internet-layer functions, such as diagnostics (ICMPv6 "ping"). This protocol is an integral part of IPv6. It allows a router or destination host to communicate with the source, typically to report an error in IP datagram processing. ICMPv6 messages are grouped into two classes: error messages and informational messages.

Command Line Configurations

ICMPv6 is automatically enabled if IPv6 is specified as the network protocol.

The configuration parameter for ICMPv6 is described in Table 33.

TABLE 33. ICMPv6 Parameter

Parameter Name	Description
TRACE-ICMPV6 [YES|NO]	Specifies whether or not tracing is enabled for ICMPv6. The default value is YES.

GUI Configuration

ICMPv6 is automatically enabled if IPv6 is specified as the network protocol. See the IPv6 section of this model library for details of configuring IPv6.

Statistics

The ICMPv6 statistics generated are shown in Table 34.

TABLE 34. ICMPv6 Statistics

Statistic	Description
Bad Code	Number of ICMPv6 message whose code field is out of range.
Packet Too Short	Number of ICMPv6 message whose length is less than the minimum length of ICMPv6 header.
Bad Packet Length	Number of ICMPv6 message whose length field is not matching the actual length.
Unreachable Address	Number of Destination Unreachable Messages with code 3 (Address unreachable).

Table 35 shows the ICMPv6 Statistics generated for NDP functionality.

TABLE 35. ICMPv6 Statistics Generated for NDP Functionality

Statistic	Description
Router Solicitation Messages Sent	Number of Router Solicitation Sent by the node.
Router Advertisement Messages Sent	Number of Router Advertisement Sent by the router.

TABLE 35. ICMPv6 Statistics Generated for NDP Functionality (Continued)

Statistic	Description
Neighbor Solicitation Messages Sent	Number of Neighbor Solicitation Sent by the node.
Neighbor Advertisement Messages Sent	Number of Neighbor Advertisement Sent by the node.
Number of Redirect Messages Sent	Number of Redirect Message Sent by the router.
Router Solicitation Messages Received	Number of Router Solicitation Received by the router.
Router Advertisement Messages Received	Number of Router Advertisement Received by the node.
Neighbor Solicitation Messages Received	Number of Neighbor Solicitation Received by the node.
Neighbor Advertisement Messages Received	Number of Neighbor Advertisement Received by the node.
Bad Router Solicitation Messages	Number of Bad Router Solicitation generated by the node.
Total Number of Bad Router Advertisement messages	Number of Bad Router Advertisement generated by the router.
Total Number of Bad Neighbor Solicitation messages	Number of Bad Neighbor Solicitation generated by the node.
Total Number of Bad Neighbor Advertisement messages	Number of Bad Neighbor Advertisement generated by the node.
Total Number of Bad Redirect Messages	Number of Bad Redirect Message generated by the router.

Internet Group Management Protocol (IGMP)

IGMP is used by hosts and routers that support multicasting. IGMP is considered part of the IP layer and IGMP messages are transmitted in IP datagrams. Unlike other protocols, IGMP has a fixed size message, with no optional data.

IGMP is a multicast group management protocol. It is generally used between routers and hosts, which involve in multicast data traffic. It operates between a host sending or receiving multicast data and its directly attached router. What we mean by a directly attached router is that router which a host comes across as the first-hop router on the path to any other router outside its local network. Or, on the contrary, the last-hop router on the way to the concerned hosts. The main work of the IGMP enabled routers and hosts are to maintain group membership information over a particular interface on a router on which one or more local hosts are connected. As we can see that the work of IGMP is limited to host and router interaction it becomes very clear that there must be other multicast routing protocols present over the internet to take care of the responsibility of routing multicast data packets to their final destinations. This is accomplished by other network-layer-multicast-routing protocols such as PIM, DVMRP or MOSPF.

We know that IGMP provides a means to the host to inform its directly attached router about its group membership. This is done by the most important three IGMP messages: membership-query (general), membership-query (group-specific) and leave-group. A general-membership-query is sent by a router to all hosts on its attached interface to solicit membership information. By *interface* we mean the primary interface on an attached network. If a router has multiple physical interfaces on a single network, then this protocol needs to run on only one of them (according to rfc2236). That is to get the set of all the multicast-groups that may have been joined by one or more hosts on that interface. The routers can also determine whether a specific group has been joined by any particular host or not. The routers may query with a specific group address in the query message, to know whether any host on the interface has joined the group or not. The attached hosts on the interface in turn replies with membership-report. This carries the membership information to the querying router. This continues till there are group members on a particular interface. If there are no more reports received for a group on a particular interface, then the attached router understands that there are no more alive group members for a particular group on a particular interface. It then deletes the group information from the group list it carries with it. Thus the router no more forwards multicast data traffic for that particular group on that interface.

Command Line Configuration

IGMP can be configured using the following parameter:

```
GROUP-MANAGEMENT-PROTOCOL    IGMP
```

IGMP takes several parameters for configuration. The details and description of those parameters are given in Table 36.

TABLE 36. IGMP Configuration Parameters

Parameter	Description
IGMP-STATISTICS [YES | NO]	Enables or disables the IGMP statistics. By default IGMP statistics are disabled.
IGMP-ROUTER-LIST {Node ID List}	Specifies the nodes that will act as IGMP router in simulation. You can specify each node explicitly (separated by a comma) within the braces or, they can use the thru keyword to indicate a range of node. Even for a single node the braces must be present; or else the program prompts an error. If there is no IGMP router, you must make this line a commented line by inserting a # at the beginning of the line. All other nodes that are not in this “Node ID List” are considered IGMP Hosts.
IGMP-UNSOLICITED-REPORT-COUNT [Number greater than zero]	Specifies how many times a host will send unsolicited reports at the time of joining a group. The default value is 2, which is the Robustness Variable of the protocol..

GUI Configuration

A multicast protocol (such as DVMRP, MOSPF, or PIM) must be enabled in order to use Group Management Protocol.See the DVMRP, MOSPF, and PIM sections of *QualNet 4.5 Multimedia and Enterprise Model Library* for details of enabling a multicast protocol.

Enabling IGMP

After enabling a multicast protocol, go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Routing Protocol > Enable Multicast? > Multicast Protocol > Group Management Protocol**. From the configurable property window, set **Group Management Protocol** to **IGMP** and configure IGMP parameters as shown in Figure 54.

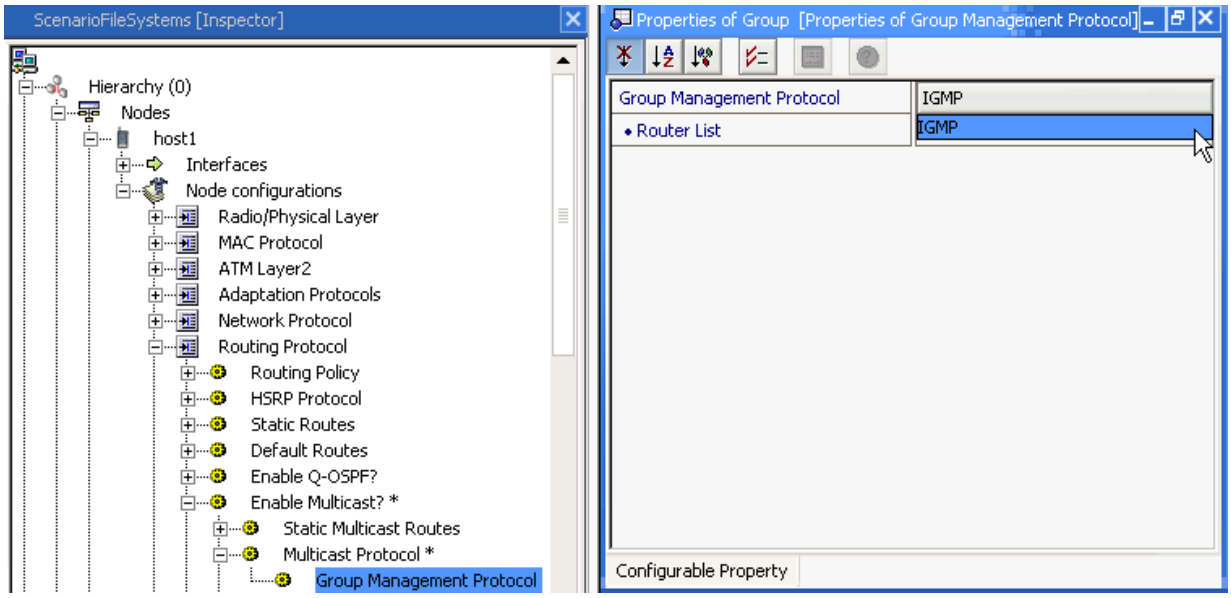


FIGURE 54. Enabling Group Management Protocol Configuration for a Node

Enabling Statistic Collection

Go to **ConfigSettings > Statistics > Statistics**. From the configurable property window, enable IGMP statistics as shown in Figure 55.

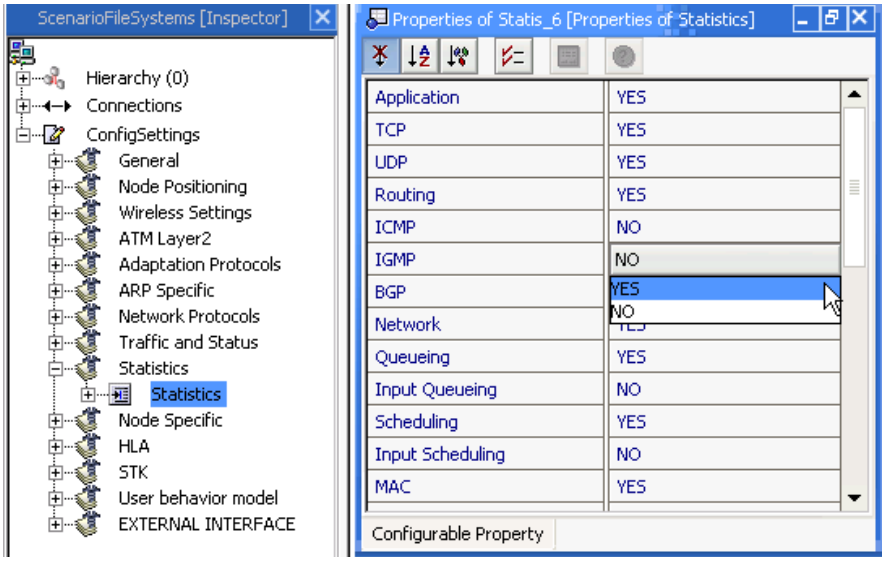


FIGURE 55. Enabling IGMP Statistics

Statistics

The IGMP statistics are shown in Table 37.

TABLE 37. IGMP Statistics

Statistic	Description
Membership created in local network	Total number of membership created in the local network
Total Group Joins	In case of host it is the number of groups the host joined through this interface, while in case of router it is the total number of different groups for which the router receives membership report through this interface
Total Group Leaves	In case of host this variable indicates whether the host leaves a previously joined group through this interface, while for router it is the total number of group that leaves this network.
Total Messages Sent	Total number of messages sent.
Total Messages Received	Total number of messages received.
Bad Messages Received	Total number of bad messages received.
Total Reports Received	Total number of reports received.
Bad Reports Received	Total number of invalid reports received.
Membership Reports Received	Total number of membership reports received.
Leave Reports Received	Total number of leave reports received.
Total Queries Received	Total number of queries received.
Bad Queries Received	Total number of invalid queries received.
General Queries Received	Total number of general queries received.
Group Specific Queries Received	Total number of group specific queries received.
Total Reports Sent	Total number of reports sent.
Membership Reports Sent	Total number of membership reports sent.
Leave Reports Sent	Total number of leave reports sent.
Total Queries Sent	Total number of queries sent.
General Queries Sent	Total number of general queries sent.
Group Specific Queries Sent	Total number of group specific queries sent.

Internet Protocol - Dual IP

IPv6 transition might become successful if IPv4/IPv6 routers/hosts maintain compatibility with the large installed base of IPv4 hosts and routers. Maintaining compatibility with IPv4 means enabling IPv6 routers/hosts to carry IPv4/IPv6 datagram over the attached IPv4 infrastructures by encapsulating the datagram within IPv4 datagram. The harmonious co-existence of IPv6 and IPv4 routers/hosts facilitates deployment of IPv6 internet among the large installed base of IPv4. IPv4 Compatibility Mechanisms include:

- Dual IP Layer: For supporting both IPv6 and IPv4 functionalities.
- Configured tunneling of IPv6 over IPv4: The IPv4 tunnel endpoint address is determined by configuration information on the encapsulating node.
- 6to4 mechanism: A mechanism for IPv6 sites to communicate with each other over the IPv4 network without explicit tunnel setup, and for them to communicate with native IPv6 domains via relay routers. Effectively it treats the wide area IPv4 networks as a unicast point-to-point link layer.

Dual-IP Layer Operation

An IPv6 node enabled with dual IP operation is provided with complete IPv4 implantation in addition to its own IPv6 layer functionalities. Such nodes are usually known as IPv6/IPv4 nodes since they have the dual ability to interoperate not only with the IPv6 nodes but also with the IPv4 nodes. While communicating with IPv4 nodes, IP4/IPv6 nodes use IPv4 packets whereas for IPv6 nodes, they use IPv6 packets.

Configured Tunneling Mechanism

Tunneling mechanism provides a way to utilize IPv4 router infrastructure to carry IPv6 traffic. An IPv4 tunnel refers to a chain of intervening IPv4 routers along the route of the IPv6 datagram. Both the starting and the ending nodes of a tunnel are essentially IPv6/IPv4 routers enabled with dual-stack implementation. The starting node of a tunnel encapsulates the IPv6 packet in IPv4 datagram that is decapsulated at the ending node of the tunnel. For each tunnel, the encapsulating node can be manually configured to store the attached IPv4 tunnel's end point address, which is associated with the tunnel-ending IPv4/IPv6 node's tunnel-connecting interface. While transmitting IPv6 datagram over the associated IPv4 tunnel, the encapsulating node encapsulates it in IPv4 datagram and route the packet to the terminating IPv4/IPv6 node using the configured tunnel end point address. The determination of which packets to tunnel is usually done via a routing table maintained by the encapsulating node. Tunnels with bi-directional trafficking ability behave as virtual point-to-point link. Any routing application for IPv6, which is running on an IPv4 tunnel end, will consider the other tunnel end point as one hop away and it will use the tunnel to send as well as receive control packets to get the routing information from other end.

Tunneling can be used in a variety of ways:

- Router-to-Router: IPv6/IPv4 routers interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans one segment of the end-to-end path that the IPv6 packet takes.
- Host-to-Router: IPv6/IPv4 hosts can tunnel IPv6 packets to an intermediary IPv6/IPv4 router that is reachable via an IPv4 infrastructure. This type of tunnel spans the first segment of the packet's end-to-end path.
- Host-to-Host: IPv6/IPv4 hosts that are interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans the entire end-to-end path of the packet.

- Router-to-Host: IPv6/IPv4 routers can tunnel IPv6 packets to their final destination IPv6/IPv4 host. This tunnel spans only the last segment of the end-to-end path.

6to4 Automatic Tunneling Mechanism

6to4 tunneling (RFC 3056 / Connection of IPv6 Domains via IPv4 Clouds) uses a simple mechanism to create automatic tunnels. Each node with a global unique IPv4 address is able to be a 6to4 tunnel endpoint (if no IPv4 firewall prohibits traffic). 6to4 tunneling is mostly not a one-to-one tunnel. Also, a special IPv6 address indicates that this node will use 6to4 tunneling for connecting the world-wide IPv6 network

Command Line Configuration

To run Dual IP as the network protocol at specific node or subnet or interface specify following in *.config file.

```
NETWORK-PROTOCOL DUAL-IP
```

Specifying tunnel configuration file

The tunnel-configuration file contains description of tunnels. Current Dual-IP implementation in QualNet supports both node-wise as well as global specification as follows:

```
[Node-id] TUNNEL-CONFIG-FILE <name of the tunnel-configuration file>
```

Configuring Tunnel

Tunnels are specified in the tunnel-configuration file. First parameter of the tunnel specification denotes the node-id on which tunnel is to be configured. The possible values of second parameter i.e. <tunnel-type> field is v4-tunnel or v6-tunnel.

Here, v4-tunnel means that this tunnel is an IPv4 tunnel, while v6-tunnel means it is an IPv6 tunnel. The next parameter specifies the starting addresses of the tunnel with address (optional not necessary) for the pseudo interface. The next parameter is the end addresses of a tunnel. The last three parameters are optional and must be configured with corresponding parameter string.

```
BORROW-FROM < interface-address of the interface from which the Routing
Protocols and other protocols are inherited>
BANDWIDTH <maximum bandwidth of the tunnel>
PROPAGATION-DELAY <end to end propagation-delay of the tunnel>
```

Tunnels should be bi-directional hence it needs to be configured at both ends.

```
# Generic Tunnel specification in the tunnel-configuration file:
<node-id> <tunnel-type> <tunnel-start-address> [optional-address] <tunnel-end-
address> [BORROW-FROM <borrowed-interface-address>] [BANDWIDTH <tunnel-
bandwidth>] [PROPAGATION-DELAY <propagation-delay>]
```

IPv4 Example

```
# IPv4 Tunnel specification in the tunnel-configuration file:
2    v4-tunnel   192.168.1.1192.168.2.2BORROW-FROM 1
5    v4-tunnel   192.168.2.2 192.168.1.1BORROW-FROM 2
```

IPv6 Example

```
# IPv6 Tunnel specification in the tunnel-configuration file:
2    v6-tunnel        2000::1     2001::2
5    v6-tunnel        2001::2     2000::1
```

Note: Optional Address in Tunnel Configuration file is mandatory if

- You enable static multicast over the tunnel, or
- You enable static routes over the tunnel and specify optional outgoing interface in static route file.

Enabling 6to4 Tunneling

You can specify whether 6to4 tunneling is enabled on an interface or not using following configuration.

```
[192.168.99.1] IPv6-ENABLE-6to4-TUNNELING YES | NO
```

Note: If you had enabled 6to4 tunneling on an interface the ipv6 stack will be disabled on that interface because 6to4 enabled interface are assumed to not having any direct connectivity from IPv6 Networks.

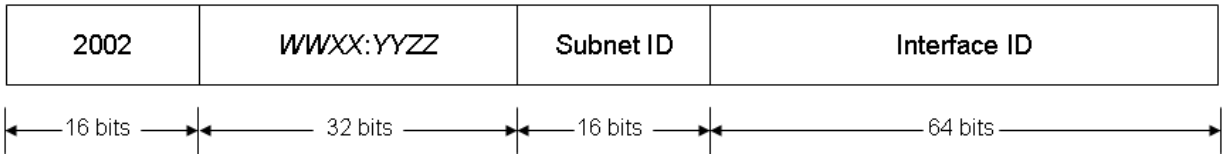
Configuring 6to4 Site

You need to generate 6to4 prefix corresponding to 6to4 enabled Interface configured above and configures the subnet with this prefix. This will configure the 6to4 site.

```
SUBNET N64-2002:c0a8:6301:0001: {1 thru 5}
LINK N64-2002:c0a8:6301:0002: {5, 6}
```

Note: You should assign different values (other than zero) to SLA field (Bit 48 to 63) of subnet prefix in above subnet and link configuration.

Structure of a 6to4 Address



Where WWXX: YYZZ is the colon-hexadecimal representation of a public IPv4 address (w.x.y.z) assigned to a subnet or host.

GUI Configuration

1. Go to **Hierarchy (x) > Nodes > host(x) > Node configuration > Network Protocol > Network Protocol**. From the configurable property window, set **Network Protocol** to **Dual-IP**, as shown in Figure 56.

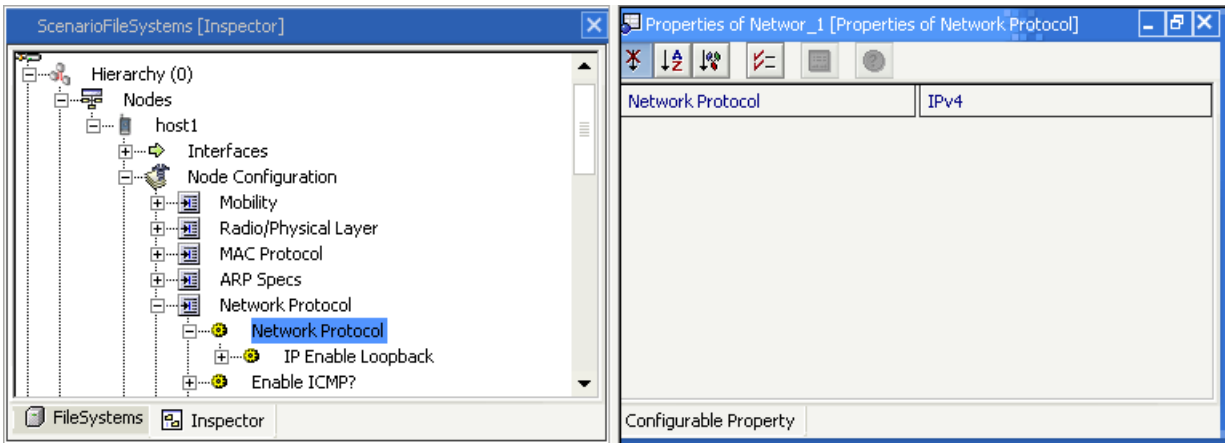


FIGURE 56. Configuring Network Protocol as Dual IP

2. When Dual IP is selected as the network protocol, specify the tunnel file, as shown in Figure 57.

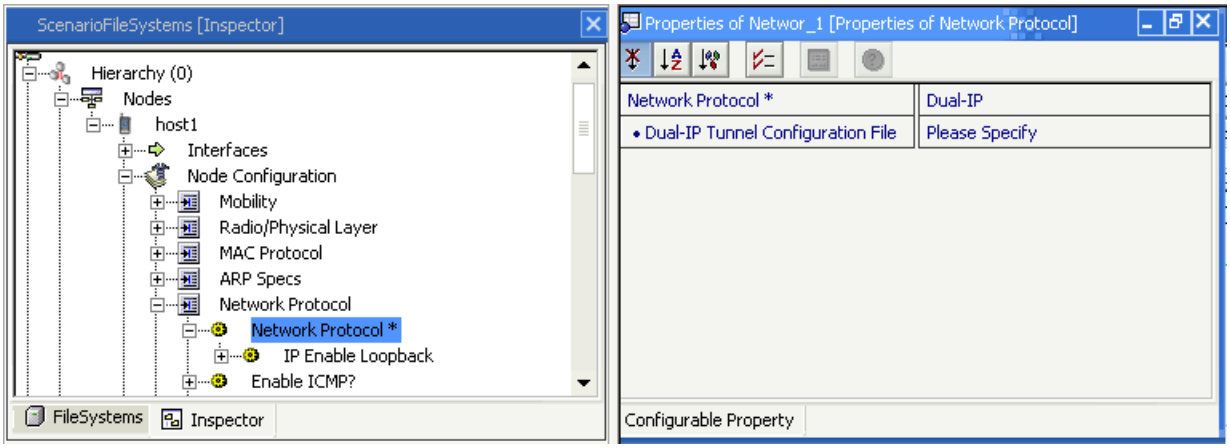


FIGURE 57. Specifying Dual IP Tunnel Configuration File

Note: If all nodes support Dual IP, then the tunnel configuration file is not needed.

Statistics

Dual IP statistics are shown in Table 38.

TABLE 38. Dual IP Statistics

Statistic	Description
Broadcast packet sent at <type of tunnel> with Endpoint <tunnel endpoint>	Number of broadcast packet encapsulated at a tunneling interface with tunnel type and tunnel end-point.
Multicast packet sent at <type of tunnel> with Endpoint <tunnel endpoint>	Number of multicast packet encapsulated at a tunneling interface with tunnel type and tunnel end-point.
Unicast packet sent at <type of tunnel> with Endpoint <tunnel endpoint>	Number of unicast packet encapsulated at a tunneling interface with tunnel type and tunnel end-point.
Broadcast packet received at <type of tunnel> with Endpoint <tunnel endpoint>	Number of broadcast packet de-capsulated at a tunneling interface with tunnel type and tunnel end-point.
Multicast packet received at <type of tunnel> with Endpoint <tunnel endpoint>	Number of multicast packet de-capsulated at a tunneling interface with tunnel type and tunnel end-point.
Unicast packet received at <type of tunnel> with Endpoint <tunnel endpoint>	Number of unicast packet de-capsulated at a tunneling interface with tunnel type and tunnel end-point.
Packet dropped at <type of tunnel> with Endpoint <tunnel endpoint> due to tunnel failure	Number of packet dropped due to tunnel failure at a tunneling interface with tunnel type and tunnel end-point.

Sample Scenario

Scenario Description

This sample is an IPv4-IPv6-IPv4 scenario. IPv4 nodes 1, 2, and 7 are connected with other IPv4 node 6 using dual IP node 3 and 5. Node 4 is an IPv6 Node. Tunnel configured is IPv6 Tunnel.

Topology

Figure 58 shows the topology of the sample scenario.

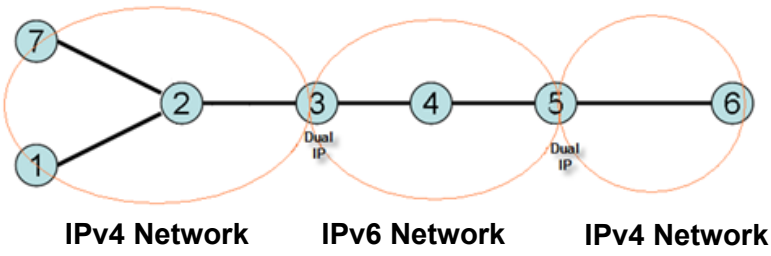


FIGURE 58. Topology of the Sample Scenario

Command Line Configuration

Enable dual IP at nodes 3 and 5 as follows:

```
[3 5] NETWORK PROTOCOL DUAL-IP
```

GUI Configuration

1. For nodes 1, 2 and 7 go to **Hierarchy(x) > Nodes > host(x) >Node Configurations > Network Protocol > Network Protocol** and from the configurable property window, select **Network Protocol** as **IPv4**.
2. For nodes 3 and 5, select **Network Protocol** as **Dual-IP** and specify **Dual-IP Tunnel Configuration file.**
3. Go to **ConfigSettings > Statistics > Statistics** and enable **Network** statistics.

Internet Protocol Security (IPSec)

IPSec is designed to provide cryptographically-based security for IPv4 and IPv6 that includes the following:

- Access Control
- Connectionless Integrity
- Data Origin Authentication
- Partial Sequential Integrity
- Confidentiality
- Traffic Flow Confidentiality

Command Line Configuration

Configuring IPSec requires specification of some parameters in the scenario configuration (.config) file and other parameters in the IPSec configuration file (which usually has the extension ".ipsec").

IPSec Parameters Specified in the Scenario Configuration (.config) File

Table 39 lists the IPSec parameters specified in the scenario configuration (.config) file.

TABLE 39. IPSec Parameters for the Scenario Configuration (.config) FIle

Parameter	Description
[node ID] IPSEC-ENABLED [YES | NO]	Enables IPSec for the node(s). The default value is NO.
IPSEC-CONFIG-FILE <filename>	Specifies the name of the IPSec configuration file. This file usually has the extension ".ipsec". Note: This is a mandatory parameter.
IPSEC-HMAC-MD5-96-DELAY <delay in clocktype per KB> IPSEC-HMAC-SHA-1-96-DELAY <delay in clocktype per KB> IPSEC-HMAC-MD5-96-DELAY <delay in clocktype per KB> IPSEC-HMAC-SHA-1-96-DELAY <delay in clocktype per KB>	Changes default delay parameters (defined in ipsec.cpp) per KB data for different algorithms. The default values for all delay parameters is 10US.
[node ID] IPSEC-DES-CBC-DELAY <delay>	Sets node-specific delay. The default value is 10US.

IPSec Parameters Specified in the IPSec Configuration (.ipsec) File

Table 40 lists the IPSec parameters specified in the IPSec configuration (.ipsec) file.

TABLE 40. IPSec Parameters for the IPSec Configuration (.ipsec) File

Parameter	Description
NODE <node id> <interface index>	Specifies the interface of the IPSec-enabled node that will be configured for IPSec.
SP <source range> <destination range> <upper layer protocol> -P <direction> <policy> Where: Source range = Nx-y.0[port number] Destination range = Nx-y.0[port number] Upper layer protocol = TCP | UDP Direction = IN | OUT Policy = DISCARD | NONE | IPSEC	Specifies an SPD entry. If IPSEC is specified it should be followed by SA names. For example: SP N2-1.0 N2-6.0 TCP -P IN IPSEC sa1
SA <name_tag> mode <dest> <security protocol> <spi> -E <encryption algorithm> "auth-key" -A <authentication algorithm> "encryp_key" Where: mode = TUNNEL | TRANSPORT dest = Tunnel End Point (Nx-y.0) security protocol = ESP (only option for now) encryption algorithm = DES-CBC authentication algorithm = HMAC-MD5-96 | HMAC-SHA-1-96	Specifies a SAD entry. For example: SA sa1 TUNNEL 6.2 ESP 12345 -E DES-CBC auth-key -A HMAC-MD5-96 encryption-key

GUI Configuration

Enabling IPSec

To enable IPSec for a particular node, perform the following steps:

1. Go to **Hierarchy (x) > Nodes > host (x) > Node configurations > Network Protocol > Enable IPSec?**. From the configurable property window, set **Enable IPsec?** to **Yes** as shown in Figure 59.

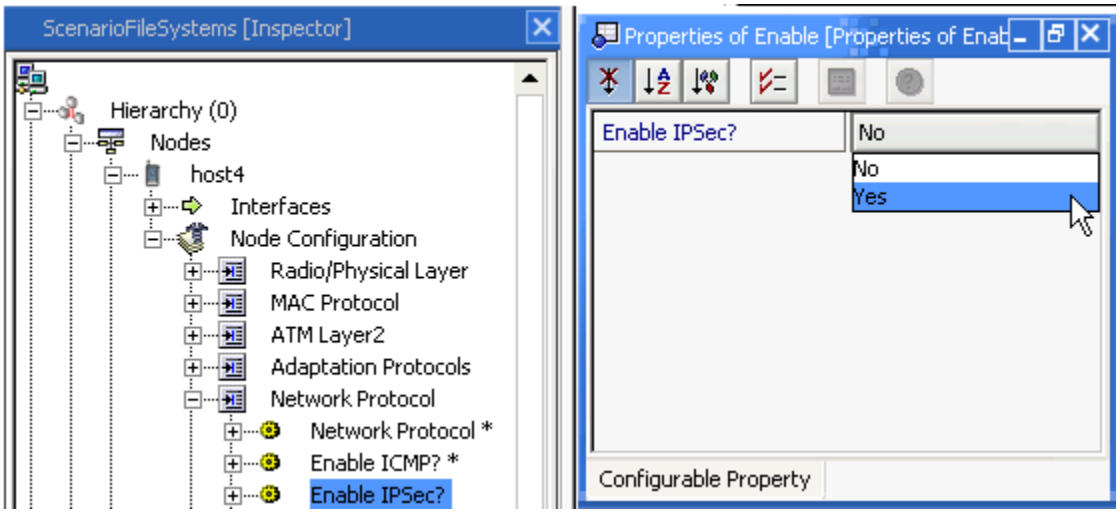


FIGURE 59. Enabling IPSec

IPSec Parameters

Once IPSec is enabled, a set of parameters is displayed. Configure IPSec parameters as shown in Figure 60.

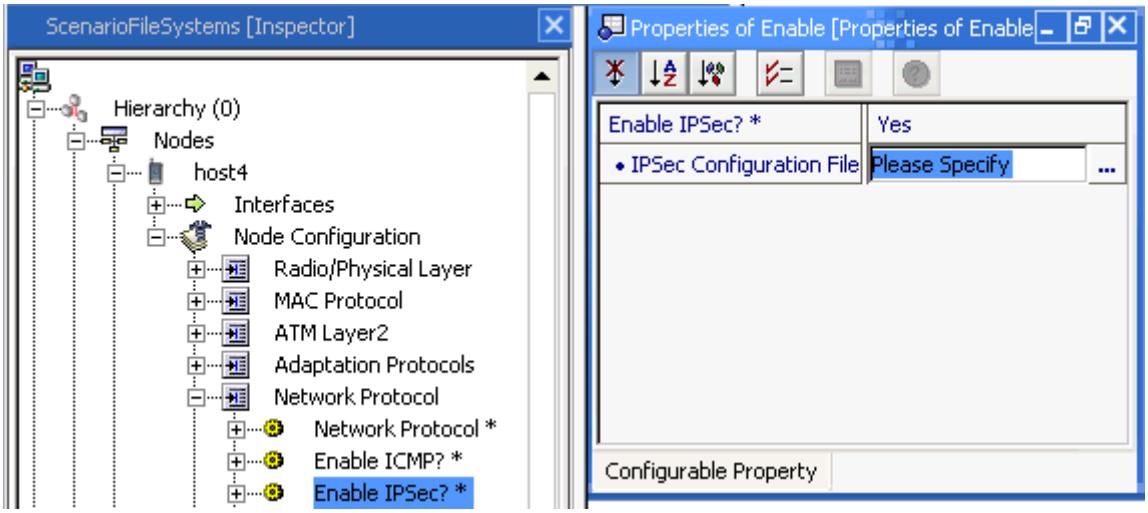


FIGURE 60. Selecting IPSec File

Configuring IPSEC Processing Delays

Set **Specify Delays for IPSec Processing?** to **IETF_RFC**. Configure the dependent parameters as shown in Figure 61.

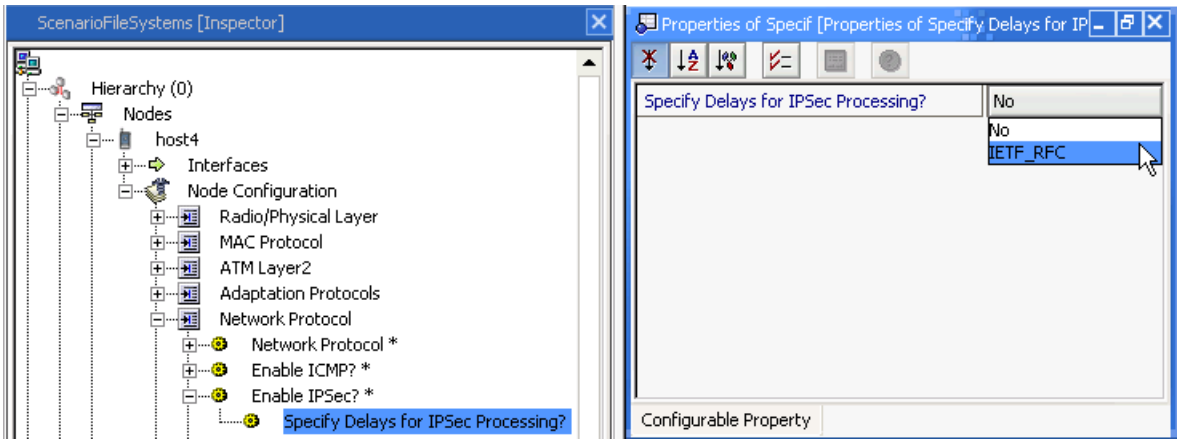


FIGURE 61. Configuring IPSEC Processing Delays

Statistics

Table 41 shows the statistics collected by IPSec.

TABLE 41. IPSec Statistics

Statistic	Description
PacketProcessed	No of IPSec packets processed at IPSec enabled inbound interface.
PacketDropped	No of IPSec packets dropped at IPSec enabled inbound interface.
TotalDelayOverhead	Total delay for inbound IPSec packet processing.
PacketProcessed	No of IPSec packets processed at IPSec enabled outbound interface.
PacketDropped	No of IPSec packets dropped at IPSec enabled outbound interface.
TotalByte Overhead	Total overhead bytes for IPsec processing.
TotalDelay Overhead	Total delay for outbound IPSec packet processing

Internet Protocol version 4 (IPv4)

IP is a protocol that provides a connectionless, best-effort delivery service of datagrams across the internet. Currently, there are two Internet Protocols: version 4 (IPv4) and version 6 (IPv6). For information on IPv6, see the section that immediately follows this section in this guide.

The Internet Protocol is designed for use in interconnected systems of packet-switched computer communication networks. The Internet Protocol provides for transmitting blocks of data called datagrams from sources to destinations, where sources and destinations are hosts identified by fixed length addresses. The Internet Protocol also provides for fragmentation and reassembly of long datagrams, if necessary, for transmission through *small packet* networks.

Command Line Configuration

Selection of this protocol is currently mandatory, and is achieved by placing the following entry in *.config:

```
NETWORK-PROTOCOL      IP
```

The configuration parameters for IPv4 are shown in Table 42.

TABLE 42. IPv4 Parameters

Parameter	Description
TRACE-IP < YES | NO >	Enables or disables the Trace of IP packets. By default, tracing of IP packets is disabled i.e. the value is NO. For example, TRACE-IP YES
IP-FRAGMENTATION-UNIT < value >	The maximum data packet size that is supported. If the size greater then the fragmentation unit comes, then the same will be fragmented before delivering the same to underlying MAC protocol. Range: 256(min) to 65535(max)
IP-FORWARDING < YES | NO >	Specifies whether a node will forward packets or not. Default value is YES. For example, IP-FORWARDING YES
DEFAULT-GATEWAY < node id >	Specifies the default gateway. Default value is 0. For example, DEFAULT-GATEWAY 8
INTERFACE-NAME	Specifies the interface name.
INTERFACE-TYPE	Interface types have been collected from two different Cisco sources for maximum accommodation following the syntax: [node-id] INTERFACE-TYPE interface-index interface-type interface-number
SCHEDULER-STATISTICS < YES | NO >	Enables or disables the scheduler statistics. For example, SCHEDULER-STATISTICS YES
SCHEDULER-GRAPH-STATISTICS <YES | NO >	Enables Scheduler Graph Statistics. For example, SCHEDULER-GRAPH-STATISTICS YES

TABLE 42. IPv4 Parameters (Continued)

Parameter	Description
SAMPLE-INTERVAL	This parameter is configured only when SCHEDULER-GRAPH-STATISTICS is configured. This becomes a mandatory parameter if Scheduler Statistics is enabled.
QUEUE-WEIGHT [priority] < value >	Specifies the queue weight for each of the priority queues. The value refers to the desired weight for that priority queue. Default value is 1.0. For example, QUEUE-WEIGHT[2] 0.2
IP-QUEUE-PRIORITY-QUEUE-SIZE <value>	Refers to the Byte Capacity of the priority queues. For example, IP-QUEUE-PRIORITY-QUEUE-SIZE 50000. This is a mandatory parameter.
IP-QUEUE-PRIORITY [priority number] <value>	Specifies the priority of the IP queue. For example, IP-QUEUE-PRIORITY [1] 9 This is a mandatory parameter.
QUEUE-STATISTICS <YES | NO >	Enables or disables the queue statistics. Default value is NO. For example, QUEUE-STATISTICS YES
BUFFER-SIZE-STATISTICS < YES | NO >	Enables or disables Buffer size stats. Default value is NO. For example, BUFFER-SIZE-STATISTICS YES
PACKETS-DROPPED-STATISTICS <YES | NO >	Enables or disables packets dropped stats. Default value is NO. For example, PACKETS-DROPPED-STATISTICS YES
INPUT-QUEUE-STATISTICS < YES | NO >	Enables or disables input queue stats. For example, INPUT-QUEUE-STATISTICS YES
INPUT-SCHEDULER-STATISTICS < YES | NO >	Enables or disables input scheduler stats. Default value is NO, For example, INPUT-SCHEDULER-STATISTICS YES
IP-QUEUE-PRIORITY-INPUT-QUEUE-SIZE <value>	Specifies the queue size of the IP priority queue. The default value is 150000
PER-HOP-BEHAVIOR-FILE	Used to select the per hop behavior file. For example, PER-HOP-BEHAVIOR-FILE default.in

GUI Configuration

Enabling IPv4

To enable IPv4, go to **Hierarchy (x) > Node configurations > Network Protocol > Network Protocol**. From the configurable property window, set Network Protocol to IPv4 as shown in Figure 62.

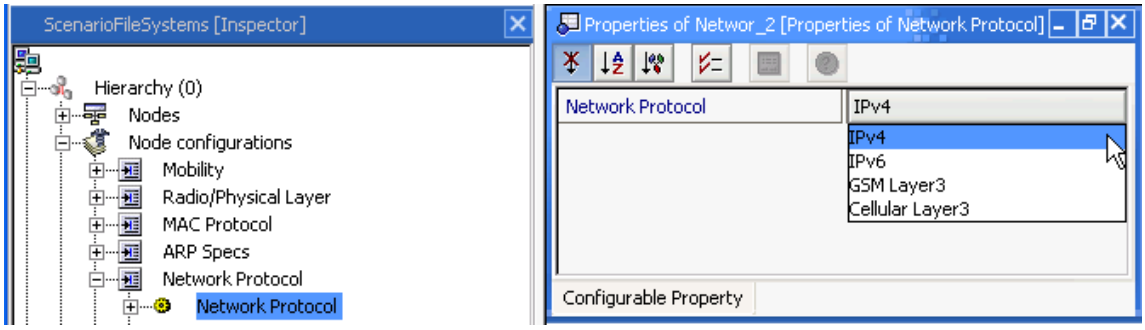


FIGURE 62. Enabling IPv4

Configuring General IPv4 Parameters

Once IPv4 is enabled, a set of parameters is displayed. Configure the IPv4 parameters as shown in Figure 63.

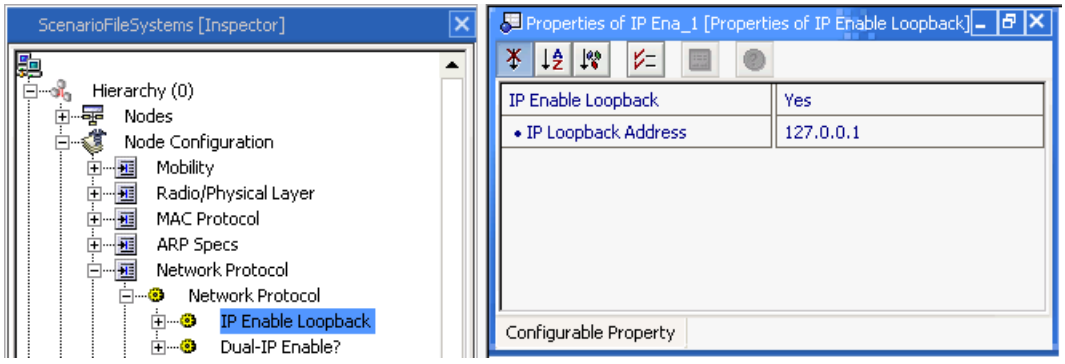


FIGURE 63. Configure General IPv4 Parameters

Configuring Gateway Configuration

Go to **Hierarchy (x) > Node configurations > Routing Protocol > Gateway Configuration**. From the configurable property window, set **Gateway Configuration** to **Yes** as shown in Figure 64.

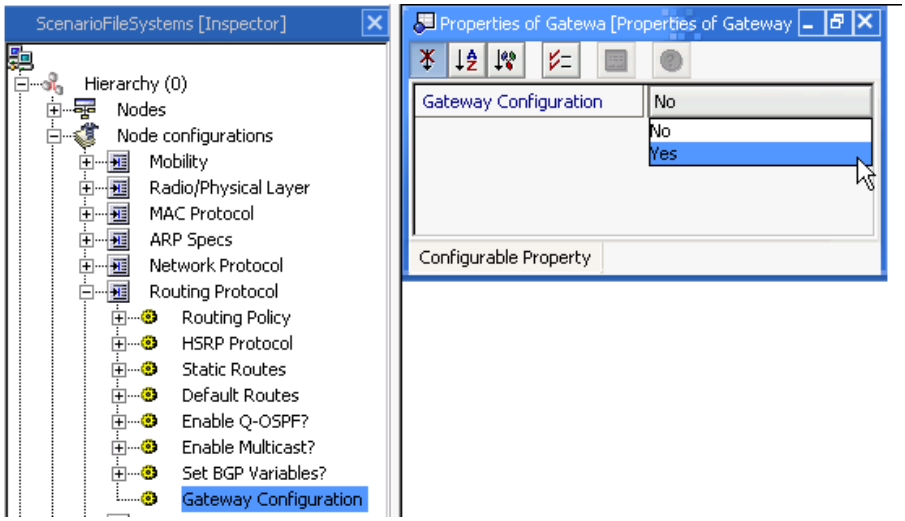


FIGURE 64. Configure Gateway Configuration

Configuring Default Gateway

Set **Gateway Configuration** to **Yes** and configure **Default Gateway** as shown in Figure 65.

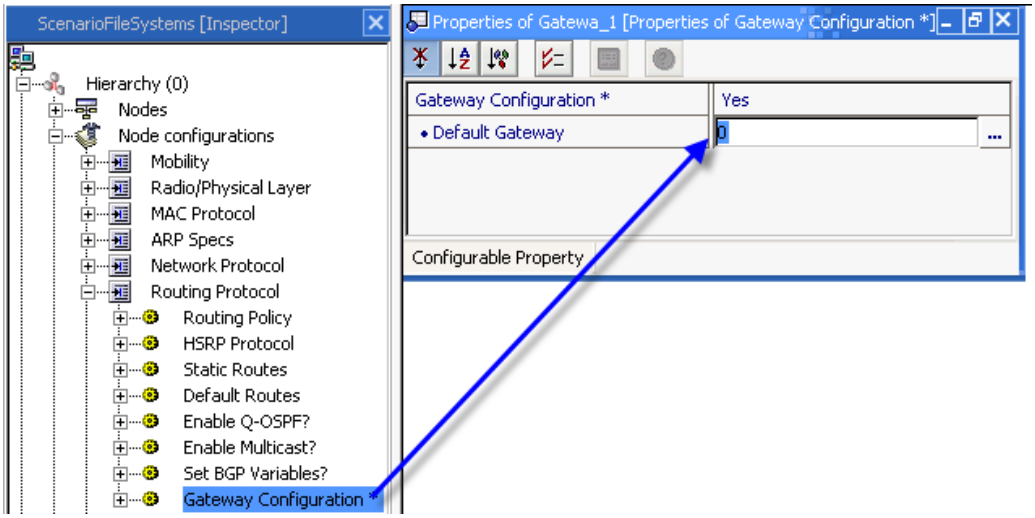


FIGURE 65. Configure Default Gateway

Enabling IP Forwarding

Go to **Hierarchy (x) > Node configurations > Routing Protocol**. From the configurable property window, set **IP Forwarding** to **Yes**, as shown in Figure 66.

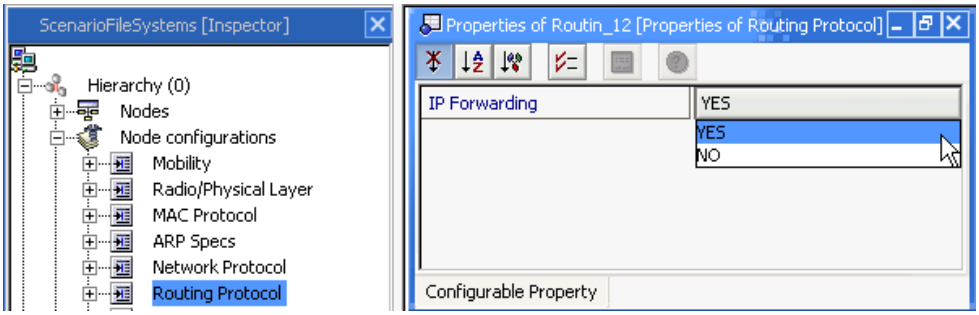


FIGURE 66. Enable IP Forwarding

Enabling Statistics

Go to **ConfigSettings > Statistics > Statistics**. Enable **Queuing**, **Input Queuing**, **Scheduling** and **Input Scheduling** statistics, as shown in Figure 67.

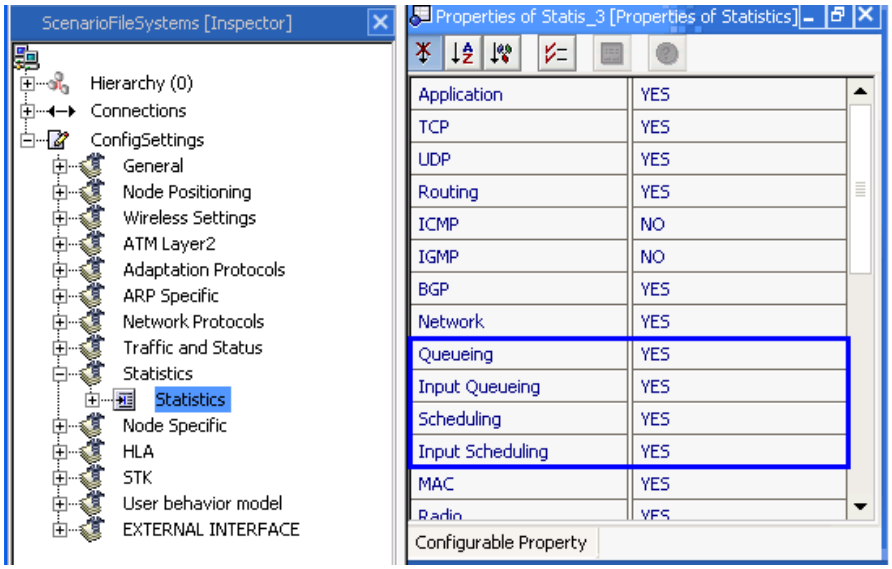


FIGURE 67. Enabling Statistics

Statistics

IPv4 statistics collected are taken from the standard MIB statistics, and listed in Table 43.

TABLE 43. IPv4 Statistics

Statistics	Description
ipInReceives	Packets received from the MAC protocol.
ipInHdrErrors	Packets dropped due to header errors.
ipInForwardDatagrams	Packets forwarded by this intermediate node towards its destination.
ipInDelivers	Packets delivered to Transport layer.
ipOutRequests	Packets sent by IP to the MAC protocol.
ipOutDiscards	Packets lost because of IP queue overflow.
ipOutNoRoutes	Packets dropped due to unreachable destination.
Packets fragmented	Number of packets fragmented
Fragments created	Number of fragments created.
Fragments received	Number of fragments received.
Fragments dropped	Number of fragments dropped.
Fragments in Buffer	Number of fragments in buffer.
Fragments reassembled	Number of fragments reassembled.
Packets created after reassembling	Number of packets created after reassembling.
ipInDelivers TTL sum	TTL sum of the packets sent to upper layers.
ipRoutePktThruGt	Number of IP datagrams routed through gateway.
Send pkt to Other Net	Number of packets sent to other networks.
Recvd pkt from Other Net	Number of packets received from other networks..
ipInDelivers TTL-based average hop count	TTL based on average hop count for packets sent to upper layers. .
ipNumDroppedDueToBackplaneLimit	Number of packets dropped due to back plane limit.
Number of packets dropped due to routing	Number of packets dropped due to routing.
ipReasmReqds	Fragments of packets received.
ipReasmOKs	Successful reassembly of fragments into a data packet.
ipReasmFails	Unsuccessful reassembly of fragments into a data packet.
IpFragOKs	Successful receptions of fragments.

Internet Protocol version 6 (IPv6)

IPv6 is the newer version of the Internet Protocol that is meant to replace IPv4. IPv4 has been used since the beginning of the Internet and has worked very well, but it has some serious limitations that IPv6 has been designed to overcome. Basic improvements of IPv6 are:

- Increased addressing space.
- IP address is expanded from 32 bits to 128 bits.
- Similar to IPv4, IPv6 also provides functionalities such as forwarding, fragmentation, automatic discovery of default routers, fastest forwarding process using destination cache, neighbor cache and prefix list, hierarchical networking, and simplified packet format.

IPv6 Addresses

IPv6 addresses are 128 bits long and are of three types. IP addresses are assigned to interfaces of the node. All interfaces must have at least one link-local unicast address. An interface may also have multiple IPv6 addresses of any type. There are three conventional forms to represent IPv6 address.

- The preferred form is x:x:x:x:x:x:x:x, where x is 16 bit hexadecimal value.
  Example: 1080:0:0:0:8:800:200C:417A
- Truncate a series of 0's with "::".
  Example: The previous address can be represented as 1080::8:800:200C:417A
- Mixed environment of IPv6 and IPv4 - x:x:x:x:x:x:d.d.d.d where 'x' is the 16 bit hexadecimal value and 'd' is the 8 bit decimal value as represented by IPv4.
  Example: 0:0:0:0:0:0:13.1.68.3

The following format is used to indicate the length of the subnet mask:

```
<IPv6-address>/<prefix-length>
```

where

`<IPv6-address>` : IPv6 address in any of the above formats
`<prefix-length>` : Number of bits in the subnet mask

IPv6 Unicast Address Types

A unicast address has two parts: subnet prefix and interface ID. Depending on the type of unicast address the length of these two types is different. There are several types of unicast address, such as:

- Global unicast address (allowed beyond the intranet)
- MSAP address (not used)
- IPX address (not used)
- site-local address (intranet specific address)
- link-local address (used for a single link)
- IPv4 capable address (0:0:0:0:0:0:<IPv4 address> or 0:0:0:0:0:FFFF:<IPv4 address>)
- Unspecified address (0:0:0:0:0:0:0:0) indicates absence of an address
- Loopback address (0:0:0:0:0:0:0:1) used to send a message to itself

QualNet N Syntax for IPv6 Addresses

Network, host, and interface addresses in IPv6 networks are represented in the format `x:x:x:x:x:x:x:x`, where each x represents a 16-bit hexadecimal value. A series of 0's can be replaced by "::".

In addition, QualNet uses a shorthand notation for IPv6 network addresses. The shorthand notation for IPv6 addresses has the following syntax:

```
N<number-of-host-bits>-<network-address>
```

where

`<number-of-host-bits>`	: Number of bits used to identify host addresses
`<network-address>`	: IPv6 address of the network, where leading zeroes may be omitted

An example of an IPv6 address in this syntax is N16-::2.0. This example specifies that 16 bits are reserved for host addresses. Hosts on this network would have IPv6 addresses from ::2:0 to ::2:FFFF, for a maximum of 65536 hosts.

QualNet TLA-NLA-SLASyntax for IPv6 Addresses

An alternate format for IPv6 addresses is:

```
<FP><TLA-ID><RES><NLA-ID><SLA-ID><Interface-ID>
```

where

`<FP>`	: 3-bit filed denoting Format Prefix for aggregate global unicast addresses, which is 001
`<TLA-ID>`	: 13-bit Top-Level Aggregation identifier
`<RES>`	: 8-bit filed reserved for future use
`<NLA-ID>`	: 24-bit Next-Level Aggregation identifier
`<SLA-ID>`	: 16-bit Site-Level Aggregation identifier
`<Interface-ID>`	: 64-bit interface identifier

QualNet supports the TLA- NLA-SLA notation for IPv6 addresses. This notation has the following syntax:

```
TLA-<tla-ID>.NLA-<nla-ID>.SLA-<sla-ID>
```

where

`<tla-ID>`	: TLA identifier
`<nla-ID>`	: NLA identifier
`<sla-ID>`	: SLA identifier

An example of an IPv6 address in this syntax is TLA-1.NLA-2.SLA-1.

Note: The TLA-NLA-SLA format is deprecated and sit is recommended that it not be used.

Creating IPv6 Networks in QualNet

IPv6 networks can be created by using the SUBNET and LINK statements, which have the following format:

```
SUBNET <Subnet Address>  {<List of Nodes>}
LINK   <Subnet Address>  {<Node 1>, <Node 2>}
```

where

`<Subnet Address>` : Address of the subnet or link in the QualNet N syntax or TLA-NLA-SLA syntax.

`<List of Nodes>` : List of node IDs in the subnet, separated by commas. A range of nodes using the keyword `thru` can also be specified.

`<Node 1>`, `<Node 2>` : Node IDs of nodes at the end points of the link.

Note: Link-Local and Site-Local addresses for IPv6 interfaces are generated automatically.

Examples:

```
SUBNET N64-2000::           {1, 2, 3}
LINK N64-2001:0000:1234:0000:0000:0000:0000:0000 {1, 2}
SUBNET TLA-1.NLA-1.SLA-1    {4, 5, 6}
LINK   TLA-1.NLA-2.SLA-1    {4, 5}
```

Command Line Configuration

To enable IPv6, specify the following parameter in the scenario configuration (.config) file:

```
NETWORK-PROTOCOL IPv6
```

The IPv6 configuration parameters are described in Table 44.

TABLE 44. IPv6 Parameters

Parameter	Description
TRACE-IPV6	Specifies whether tracing is enabled for IPv6. The default value is NO.

GUI Configuration

Enable IPv6

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Network Protocol > Network Protocol**. From the configurable property window, set **Network Protocol** to **IPv6** as shown in Figure 68.

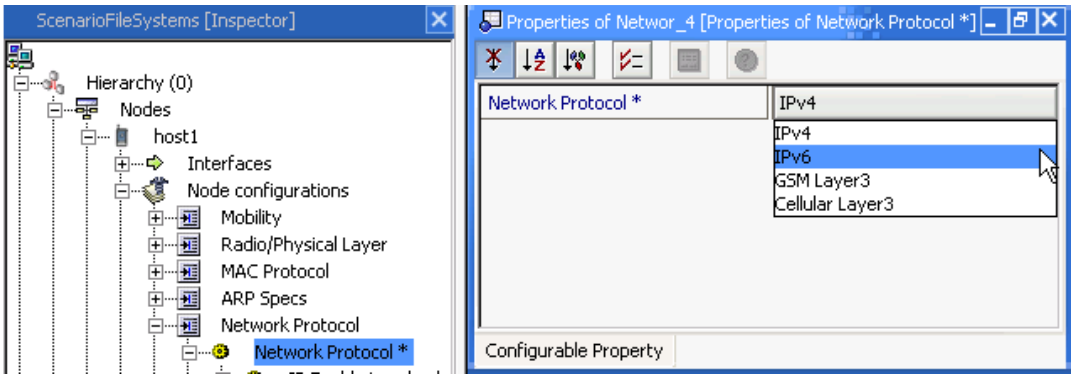


FIGURE 68. Configuring Network Protocol

Configure IPv6 Address

Go to **Hierarchy (x) > Nodes > host(x) > Interfaces > interface0**. From the configurable property window configure **IPv6 Address** as shown in Figure 69.

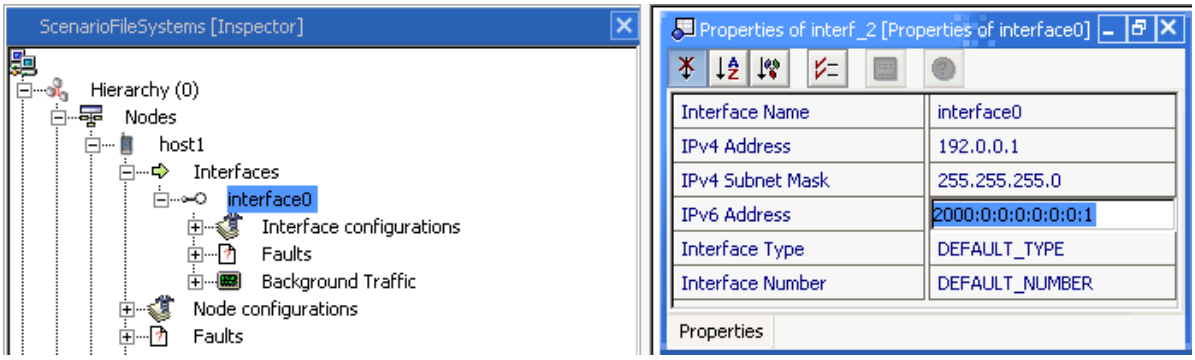


FIGURE 69. Configuring IPv6 Address

Statistics

IPv6 statistics collected are taken from the standard MIB statistics, and are explained in Table 45.

TABLE 45. IPv6 Statistics

Statistics	Description
Total packets received	Number of packets received
Datagrams delivered to upper level	Number of datagrams delivered to upper level.
Packets forwarded	Number of packets forwarded
Total ipv6 packets generated here	Number of IPv6 packets generated
Packets rcvd for unreachable dest	Number of packets received for unreachable destination
Packets discarded due to no route	Number of packets discarded due to no route

TABLE 45. IPv6 Statistics (Continued)

Statistics	Description
Packets received from bad sources	Number of packets received from bad sources
Packets output discarded	Number of packets output discarded
Improper IP version	Improper IP version number in the IP header.
Not forwarded because size greater than MTU	Number of packets not forwarded because size greater than MTU
Packet too short	Number of packets in which total packet length is less than the sum of header length and data length.
Not enough data	Number of packets in which total packet length is less than the sum of all the header length
Number of fragmented packets received	Number of fragments packets received
Number of fragmented packets dropped	Number of fragmented packets dropped
Number of fragmented packets time out	Number of fragmented packets time out
Number of Reassembled packets	Number of reassembled packets
Total output fragment created	Number of output fragment created

Neighbor Discovery Protocol

The IPv6 Neighbor Discovery protocol corresponds to a combination of the IPv4 protocols ARP, ICMP Router Discovery (RDISC), and ICMP Redirect (ICMPv4).

Nodes (hosts and routers) use Neighbor Discovery to determine the link-layer addresses for neighbors known to reside on attached links and to quickly purge cached values that become invalid. Hosts also use Neighbor Discovery to find neighboring routers that are willing to forward packets on their behalf. Finally, nodes use the protocol to actively keep track of which neighbors are reachable and which are not, and to detect changed link-layer addresses.

Neighbor discovery defines mechanisms for solving each of the following problems:

- Router Discovery: How hosts locate routers that reside on an attached link.
- Prefix Discovery: How hosts discover the set of address prefixes that define which destinations are on-link for an attached link. (Nodes use prefixes to distinguish destinations that reside on-link from those only reachable through a router.)
- Parameter Discovery: How a node learns such link parameters as the link MTU or such Internet parameters as the hop limit value to place in outgoing packets.
- Address Auto configuration: How nodes automatically configure an address for an interface.
- Address Resolution: How nodes determine the link-layer address of an on-link destination (e.g., a neighbor) given only the destination's IP address.
- Next-hop Determination: The algorithm for mapping an IP destination address into the IP address of the neighbor to which traffic for the destination should be sent. The next-hop can be a router or the destination itself.
- Neighbor Unreachability Detection: How nodes determine that a neighbor is no longer reachable. For neighbors used as routers, alternate default routers can be tried. For both routers and hosts, address resolution can be performed again.

- Duplicate Address Detection: How a node determines that an address it wishes to use is not already in use by another node.
- Redirect: How a router informs a host of a better first-hop node to reach a particular destination.

Neighbor Discovery defines five different ICMP packet types: A pair of Router Solicitation and Router Advertisement messages, a pair of Neighbor Solicitation and Neighbor Advertisements messages, and a Redirect message. The messages serve the following purpose:

- Router Solicitation: When an interface becomes enabled, hosts may send out Router Solicitations that request routers to generate Router Advertisements immediately rather than at their next scheduled time.
- Router Advertisement: Routers advertise their presence together with various link and Internet parameters either periodically, or in response to a Router Solicitation message. Router Advertisements contain prefixes that are used for on-link determination and/or address configuration, a suggested hop limit value, etc.
- Neighbor Solicitation: Sent by a node to determine the link-layer address of a neighbor, or to verify that a neighbor is still reachable via a cached link-layer address. Neighbor Solicitations are also used for Duplicate Address Detection.
- Neighbor Advertisement: A response to a Neighbor Solicitation message. A node may also send unsolicited Neighbor Advertisements to announce a link-layer address change.
- Redirect: Used by routers to inform hosts of a better first hop for a destination which has been discussed in the ICMPv6 section as well.

Table 46 lists the statistics ICMPv6 generates for NDP functionality

TABLE 46. ICMPv6 Statistics Generated for NDP Functionality

Statistic	Description
Router Solicitation Sent	Number of Router Solicitation Sent
Router Advertisement Sent	Number of Router Advertisement Sent
Neighbor Solicitation Sent	Number of Neighbor Solicitation Sent
Neighbor Advertisement Sent	Number of Neighbor Advertisement Sent
Redirect Message Sent	Number of Redirect Message Sent
Router Solicitation Received	Number of Router Solicitation Received
Router Advertisement Received	Number of Router Advertisement Received
Neighbor Solicitation Received	Number of Neighbor Solicitation Received
Neighbor Advertisement Received	Number of Neighbor Advertisement Received
Bad Router Solicitation	Number of Bad Router Solicitation
Bad Router Advertisement	Number of Bad Router Advertisement
Bad Neighbor Solicitation	Number of Bad Neighbor Solicitation
Bad Neighbor Advertisement	Number of Bad Neighbor Advertisement
Bad Redirect Message	Number of Bad Redirect Message

Logical Link Control (LLC) Protocol

Logical Link Control (LLC) is the higher of the two data link layer sub layers defined by the IEEE. The LLC sub layer handles error control, flow control, framing, and MAC-sub layer addressing. The most prevalent LLC protocol is IEEE 802.2, which includes both connectionless and connection-oriented variants.

LLC is a part of the data link layer in a protocol stack. The data link layer controls access to the network medium and defines how upper-layer data in the form of packets or datagram is inserted into frames for delivery on a particular network. The underlying physical layer then transmits the framed data as a stream of bits on the network medium.

Command Line Configuration

To enable LLC, include the following parameter in the scenario configuration (.config) file:

```
LLC-ENABLED       [YES | NO]
```

This parameter can be specified at the global and node, and interface levels. The default value of LLC-ENABLED is NO.

Note: When configuring LLC at the node level, make sure that LLC is enabled at both communicating nodes.

GUI Configuration

To enable LLC at the global level using the GUI, go to **ConfigSettings > LLC Configuration > LLC Configuration**. In the configurable property window, set **LLC Enabled** to **Yes**, as shown in Figure 70.

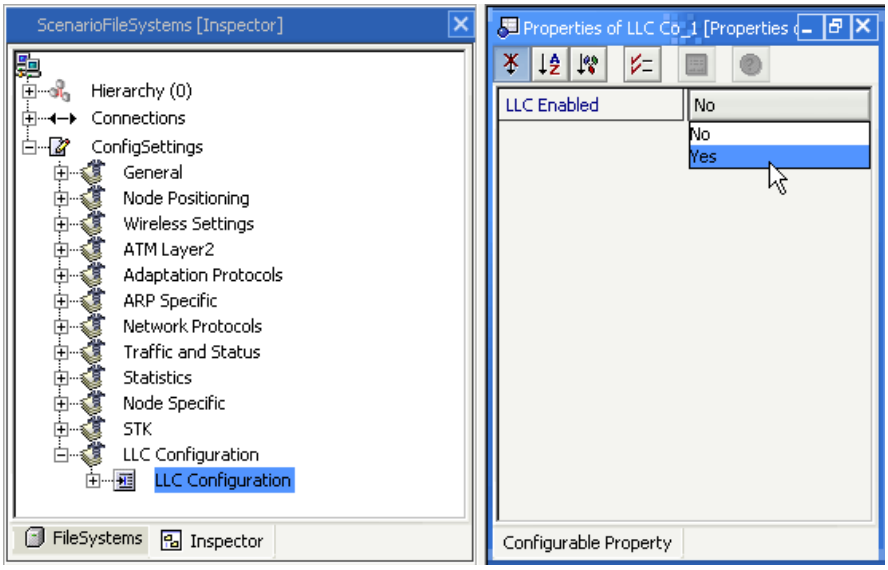


FIGURE 70. Enabling LLC

Lookup Traffic Generator

Lookup is an unreliable query/response application that can be used to simulate applications such as DNS lookup or ping.

Command Line Configuration

To use Lookup, you will need the following format:

```
LOOKUP <src> <dest> <num requests to send> <request size>
<reply size> <request interval> <reply delay>
<start time> <end time>
[TOS <tos-value> | DSCP <dscp-value> |
PRECEDENCE <precedence-value>]
```

Table 47 shows the Lookup application parameters. These parameters are mandatory:

TABLE 47. Lookup Parameters

Parameter	Function
<src>	Specifies the source node's ID or IP address.
<dest>	Specifies the destination node's node ID or IP address.
<num requests to send>	Specifies how many request packets to send. If it is set to 0, Lookup will run until the specified <end time> or until the end of the simulation, whichever comes first.
<request size>	Specifies the size of each request packet.
<reply size>	Specifies the size of each reply size. The range is from 24 to 65023.
<request interval>	Inter-departure interval between request packets. The range is from 24 to 65023
<reply delay>	Specifies the delay after receiving a request packet before responding with a reply packet.
<start time>	Specifies when the transmission should begin.
<end time>	Specifies when the transmission should end. If it is set to 0, Lookup will run until all <num requests to send> is transmitted or until the end of simulation, whichever comes first.
[TOS <tos value>]	Specifies the contents of the 8-bit TOS value of the IP header for the packets generated. The range is from 0 to 255.
[DSCP <dscp value>]	Specified the contents of the 6-bit DSCP value of the IP header for the packets generated. The range is from 0 to 63.
[PRECEDENCE <precedence value>]	Specified the contents of the 3 bit Precedence value of the IP header for the packets generated. The range is from 0 to 7.

Notes:

1. If <num requests to send> and <end time> are both greater than 0, Lookup will run until either <items to send> is done, <end time> is reached, or the simulation ends, whichever comes first.
2. At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP or TOS is not specified, a TOS value of 0 is assumed.

GUI Configuration

Configuring Lookup Parameters

1. Go to **Connections > LOOKUP [x] -> [y]**. From the configurable property window, configure the Lookup parameters as shown in Figure 71.

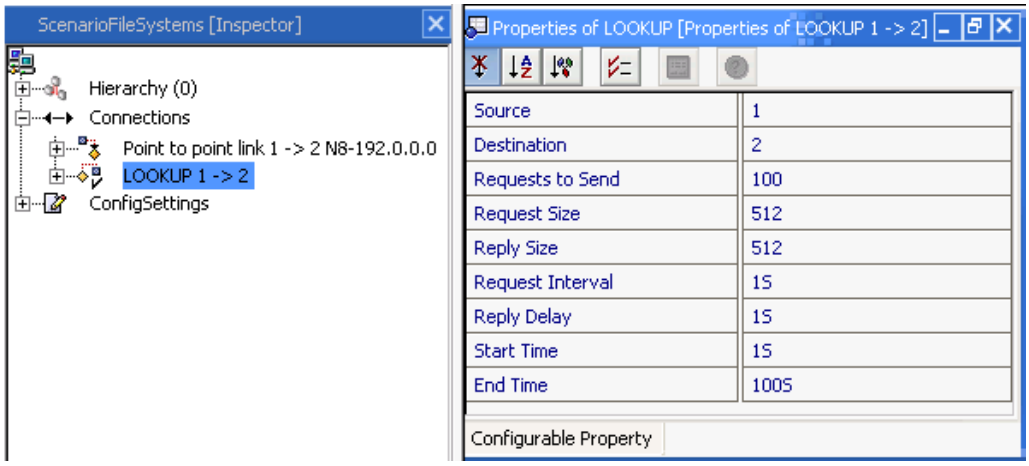


FIGURE 71. Configuring Lookup Parameters

2. To set the priority, go to **Connections > LOOKUP [x] -> [y] > Priority**. In the configurable property window, set **Priority** to the desired value from the list and set the dependent parameters for the selected priority type, as shown in Figure 35.

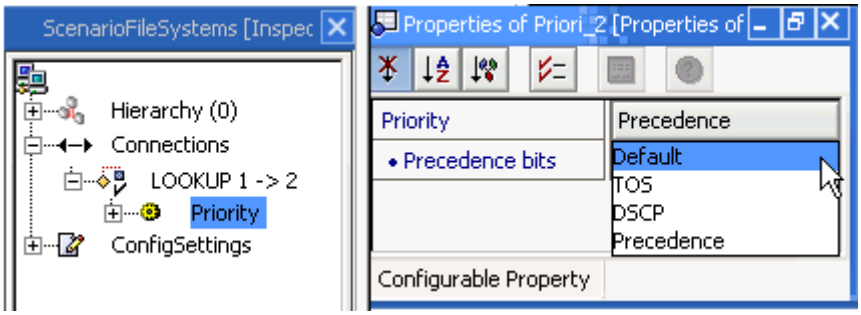


FIGURE 72. Setting Priority

Statistics

The Lookup model collects the statistics shown in Table 48.

TABLE 48. Lookup Statistics

Statistic	Description
Lookup Client	
Server address	Address of the server.
First Packet Sent at (s)	Time in second when first packet was sent.
Last Packet Sent at (s)	Time in second when last packet was sent.
Session status	Current status of the session.
Total bytes Sent	Total number of bytes sent.
Total packets Sent	Total number of packets sent.
Total bytes received	Total number of bytes received.
Total packets received	Total number of packets received.
Average round-trip delay	Average value of round-trip delay.
Lookup Server	
Client address	Address of the client.
First packet received at (s)	Time in second when first packet was received.
Last packet received at (s)	Time in second when last packet was received.
Session status	Current status of the session.
Total Bytes Received	Total number of bytes received.
Total Packets Received	Total number of packets received.
Total bytes sent	Total number of bytes sent.
Total packets sent	Total number of packets sent.

Multicast Constant Bit Rate (MCBR) Traffic Generator

MCBR operates identically to CBR, with the following exceptions:

- The destination must be a multicast address
- The type of service is not supported (TOS is always set to 0)
- RSVP-TE is not an option.
- To use MCBR, a multicast routing protocol must be configured.

Command Line Configuration

To specify MCBR traffic in the application configuration file (default.app, for example), specify the parameters according to the format in Table 49:

```
MCBR <src> <multicast destination> <items to send> <item size> <interval>
     <start time> <end time>
```

TABLE 49. MCBR Parameters

Parameter	Description
<src>	Specifies the client node's ID or IP Address.
<multicast destination>	Specifies the group multicast address to which data packets are sent.
<items to send>	Specifies the number of items to send. Note: If <items to send> is set to 0, MCBR runs until the specified <end time>, or until the end of the simulation, whichever comes first.
<item size>	Specifies the size of each item.
<interval>	Specifies the pause time between transmission of each item (inter-departure time).
<start time>	Specifies when the transmission of data packets must begin.
<end time>	Specifies when the transmission of data packets must end.

Notes:

1. If <end time> is set to 0, MCBR runs until all <items to send> are transmitted, or until the end of simulation, whichever comes first.
2. If <items to send> and <end time> are both greater than 0, MCBR runs until either <items to send> is done, <end time> is reached, or the simulation ends, whichever comes first.

Examples

1. Node 1 sends to IPv4 multicast address 225.0.0.1 ten items of 1460B each at the start of the simulation, and up to 600 seconds into the simulation. The inter-departure time for each item is 1 second.
   ```
   MCBR 1 225.0.0.1 10 1460 1S 0S 600S
   ```
2. Node 1 continuously sends to IPv4 multicast address 225.0.0.1 items of 1460B each at the start of the simulation up to 600 seconds into the simulation. The inter-departure time for each item is 1 second.
   ```
   MCBR 1 225.0.0.1 0 1460 1S 0S 600S
   ```
3. Node 1 continuously sends to IPv4 multicast address 225.0.0.1 items of 1460B each at the start of the simulation up to the end of the simulation. The inter-departure time for each item is 1 second.

```
MCBR 1 225.0.0.1 0 1460 1S 0S 0S
```

4. Node 1 continuously sends to IPv6 multicast address ff1E items of 1460B each at the start of the simulation up to the end of the simulation. The inter-departure time for each item is 1 second.

```
MCBR 1 ff1E::3 0 1460 1S 0S 0S
```

GUI Configuration

In QualNet GUI, MCBR is a configured as a single host application. Perform the following steps to configure MCBR:

1. Go to **Hierarchy (x) > Nodes > host # > SingleHostApplications**. Right click on **SingleHostApplications** and select **Add** from the pop-up menu, as shown in Figure 73.

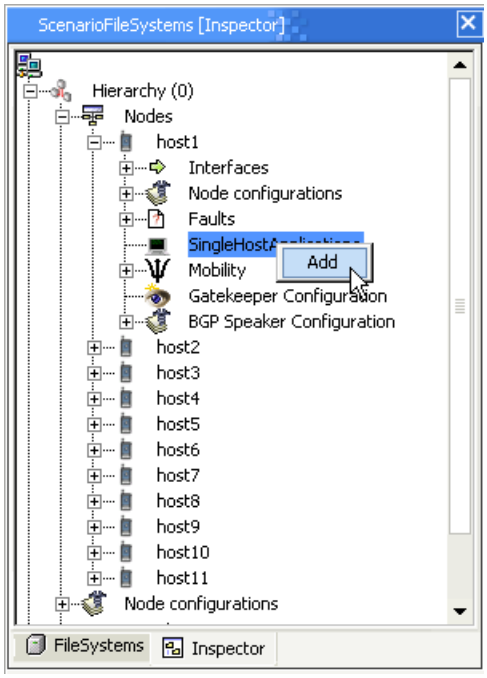


FIGURE 73. Adding a Single Host Application

2. The Single Host Application wizard is launched (see Figure 74). Select MCBR from the list and click on Finish.

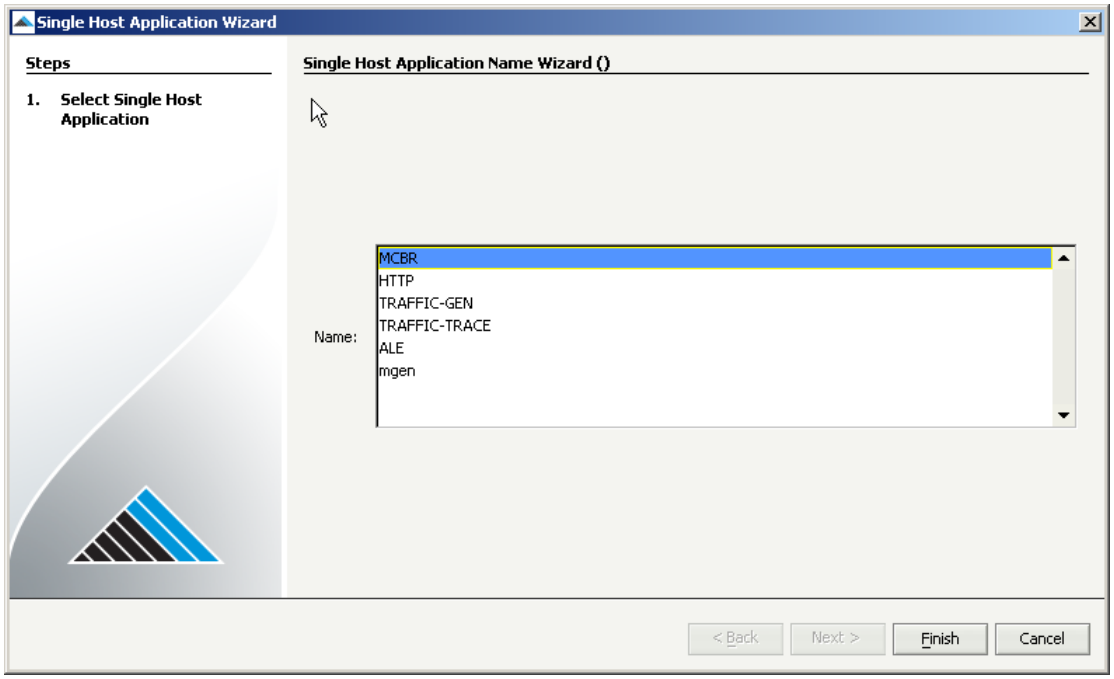


FIGURE 74. Single Host Application Wizard

3. Go to **Hierarchy (x) > Nodes > host # > SingleHostApplications > MCBR**. In the Configurable Property window, set the MCBR parameters, as shown in Figure 75.

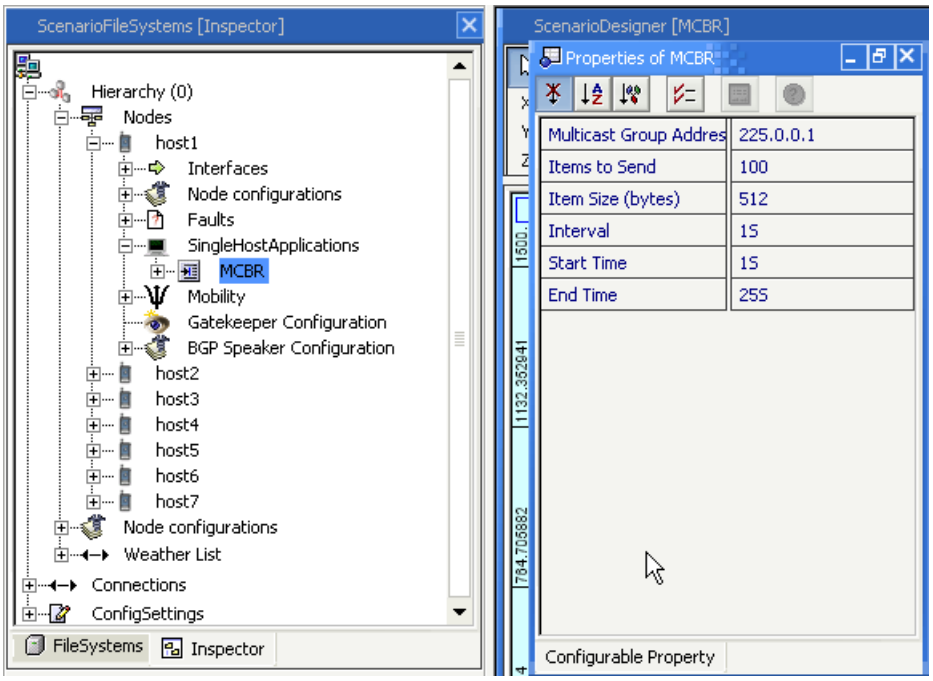


FIGURE 75. Configuring MCBR Parameters

Statistics

Table 50 shows the statistics collected by the MCBR model.

TABLE 50. MCBR Statistics

Statistic	Description
Client	
Server Address	Specifies the IP address of the server
First Packet Sent at (s)	Specifies the time at which first packet was sent
Last Packet Sent at (s)	Specifies the time at which last packet was sent
Session Status	Specifies whether the session status is open or closed
Total Bytes Sent	Total number of data bytes sent
Total Packets Sent	Total number of data packets sent
Throughput (bits/s)	Total number of throughput, from the client side
Server	
Client Address	Specifies the IP address of the client
First Packet Received at (s)	Specifies the time at which first packet was received
Last Packet Received at (s)	Specifies the time at which last packet was received
Average End-to-End Delay (s)	Specifies the average end-to-end delay time

TABLE 50. MCBR Statistics (Continued)

Statistic	Description
Session Status	Specifies whether the session status is open or closed
Total Bytes Received	Total number of data bytes received
Total Packets Received	Total number of packets received
Throughput (bits/s)	Total number of throughputs from the server side

Random Early Detection (RED) Queue

RED drops packets at the current router based on drop probabilities, rather than at a subsequent congested router, if ECN is off (default). If ECN is active, packets are marked instead of dropped, and the source of the TCP traffic will throttle back based on the notification. RED is a congestion-avoidance algorithm that calculates average queue sizes in order to address incipient congestion. The algorithm also attempts to address global synchronization as well as maintaining fairness across data flows, without bias against bursty flows.

Command Line Configuration

To select RED as the Network Queue Discipline, place the following entry in *.config:

```
IP-QUEUE-TYPE     RED
```

It requires five parameters, which can be defined using network-specific or node-specific parameterization. The following example illustrates a RED configuration where the maximum dropping probability is 1/50 (0.02 as specified by “RED-MAX-PROBABILITY”). The possibility of random early drops begins when the RED queue average algorithm determines that the queue size (number of packets in the queue, not the bytes used by those packets) average is greater than or equal to “RED-MIN-THRESHOLD” and less than “RED-MAX-THRESHOLD”. The “RED-QUEUE-WEIGHT” determines the bias towards recent or historical queue lengths in calculating the average queue size. The “RED-SMALL-PACKET-TRANSMISSION-TIME” is the time it would take to transmit a typical small packet, and this value is used to estimate the queue average during idle periods (when the queue is completely empty). Table 51 shows the RED configuration parameters.

TABLE 51. RED Parameters

Parameter	Description
RED-MIN-THRESHOLD <value>	Specifies the number of packets in the queue that represents the lower bound at which packets can be randomly dropped. Default value is 5.
RED-MAX-THRESHOLD <value>	Specifies the number of packets in the queue that represents the upper bound at which packets can be randomly dropped. Default value is 15.
RED-MAX-PROBABILITY <value>	Specifies maximum probability (0...1) value at which a packet can be dropped (before the queue is completely full). Default value is 0.02.
RED-QUEUE-WEIGHT <value>	Specifies queue weight value used to determine the bias towards recent or historical queue lengths in calculating the average. Default value is 0.002.
RED-SMALL-PACKET-TRANSMISSION-TIME <time>	Specifies sample amount of time to transmit a small packet. Used to estimate the queue average during idle periods. Default value is 10MS.

Table 52 shows a mapping between these parameters and the variable and parameter names given for the detailed algorithm for RED gateways.

TABLE 52. RED Parameters Mapped to Detailed Algorithm

QualNet Configuration File Parameter	Detailed Algorithm for RED
RED-MIN-THRESHOLD	minth
RED-MAX-THRESHOLD	maxth

TABLE 52. **RED Parameters Mapped to Detailed Algorithm (Continued)**

QualNet Configuration File Parameter	Detailed Algorithm for RED
RED-MAX-PROBABILITY	maxp
RED-QUEUE-WEIGHT	wq
RED-SMALL-PACKET-TRANSMISSION-TIME	s

Setting Node-Specific Parameters

To restrict the specification of a protocol-based parameter to a single node or set of nodes, include the IP Addresses or nodeIds of the nodes to which the parameter should apply, in brackets to the left of the parameter name, as in the following example:

```
[0.0.1.1
0.0.1.2
3
4] IP-QUEUE-TYPE RED Random Early Detection; Quality of Service (QoS) queue
management protocol.
IP-QUEUE-TYPE FIFO First-In First-Out; Queue management protocol.
```

The previous example sets the queue mechanism on the nodes that have network interfaces with IP addresses 0.0.1.1, and 0.0.1.2 to use Random Early Detection. It also sets all network interfaces on nodes with nodeIds 3 and 4, to use Random Early Detection queuing as well.

GUI Configuration

Enabling RED

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Network Protocol > Specify Priority-wise IP Queue Type? > IP Queue Type**. From the configurable property window, set **IP Queue Type** as **RED** as shown in Figure 76.

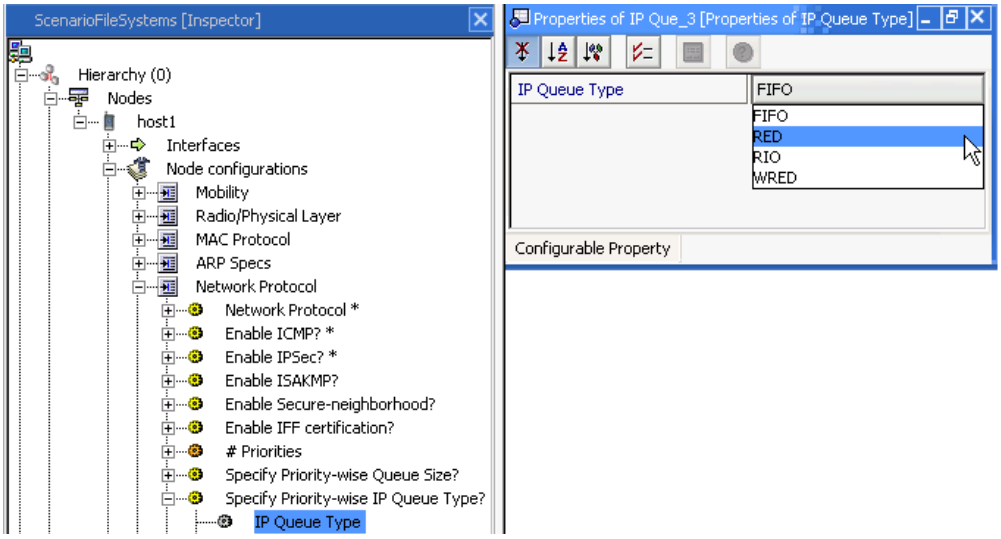


FIGURE 76. **Enabling RED for a Node**

Configuring RED Parameters

After enabling RED, configure the parameters as shown in Figure 77.

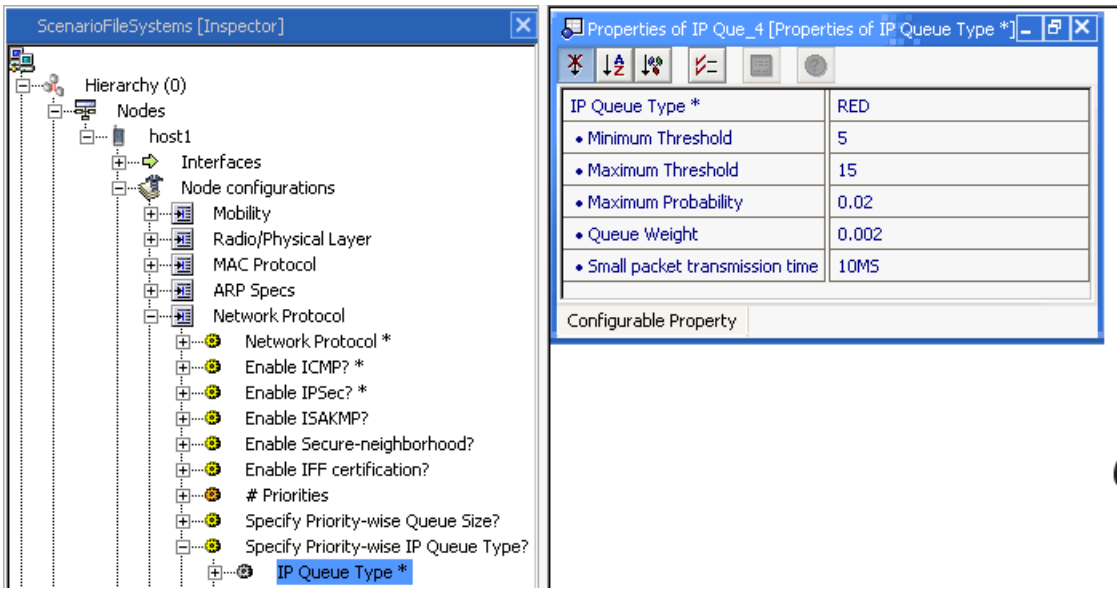


FIGURE 77. Configuring RED Parameters

Statistics

RED queues collect the statistics shown in Table 53, per priority queue, per node:

TABLE 53. RED Statistics

Statistic	Description
Total Packets Queued	Total number of packets queued
Total Packets Dequeued	Total number of packets dequeued
Total Packets Dropped	Total number of packets dropped
Average Queue Length (bytes)	Average queue length in bytes
Average Time in Queue	Average time spent by packets in the queue (in seconds)
Peak Time in Queue	Longest time spent in queue by a packet (in seconds)
Peak Queue Size (bytes)	Largest number of bytes stored in the queue at any time (the upward bound on this stat would be the queue size itself).
Total Packets Dropped Forcefully	Total packets dropped forcefully
Packets Marked ECN	Total number of packets marked by ECN capable network
Longest Time in Queue	Longest delay suffered by a packet in this queue.

Random Early Detection with In/Out (RIO) Queue

RIO is a Multilevel RED queue management protocol (MRED) queuing algorithm suitable for the AF PHB that can identify any preferentially and can punish misbehaving users. The incoming packets are marked through packet marking algorithm on the basis of Service Level Agreement (SLA) between the customer and service-provider. The making is done in two-color (DP0 and DP1) or three-color (DP0, DP1, DP2) level referring drop precedence (DP) as below:

- Green: Level Low Drop, DP0
- Yellow: Level Medium Drop, DP1
- Red: Level High Drop, DP2

RIO is a Multiple Average Multiple threshold (MAMT) variant of multilevel RED and can operate in both two and three color modes. Two simple approaches to the MAMT variant of MRED are RIO-C (RED with In/Out and Coupled average Queues) and RIO-DC (RED with In/Out and Decoupled average Queues).

RIO-C derives its name from the coupled relationship of the average queue calculation. The average queue for packets of different colors can be calculated by adding its average queue to the average queues of colors of lower drop precedence. The implication of this scheme is that the drop probability for packets with higher drop precedence is dependent on the buffer occupancy of packets having lower drop precedence. Thus, there is an inherent assumption that it is better to drop DP1 and DP2 packets for DP0 packets. Similarly, for DP1 packets, it is assumed better to drop DP2 packets. Services based on AF implementations utilizing RIO-C will be subject to this restriction.

RIO-DC (RIO with Decoupled Queues), the average queue for packets of each color is calculated independently. In this scheme, the average queue length for a color is calculated using number of packets of that color in the queue. The RED parameter settings can be chosen depending on the treatment to be given to different colors. RIO-DC appears to be the most suitable MRED-variant if the operator desires to assign weights to different colored packets within the same queue i.e. there is no relationship between the packets marked green and yellow. Thus, the RIO-DC scheme could start to emulate the WRR scheduling scheme except that a single queue is used.

Command Line Configuration

To select RIO queuing discipline for the AF PHB, place the following entry in *.config:

```
IP-QUEUE-TYPE RIO
```

TABLE 54. RIO Parameters

Parameter	Description
RIO-COUNTING-MODE [COUPLED | DECOUPLED]	Specifies the relationship of the average queue calculation. This parameter is mandatory.
RIO-COLOR-MODE [TWO-COLOR | THREE-COLOR]	Specifies the treatment to be given to RIO-COLOR-MODE. If TWO-COLOR mode is specified we need to specify RED parameter setting for Green and Yellow profile. With THREE-COLOR-MODE, we need to specify RED parameter setting for Green, Yellow and Red profile. This parameter is mandatory.

TABLE 54. RIO Parameters

Parameter	Description
GREEN-PROFILE-MIN-THRESHOLD [Number greater than zero]	Specifies the number of green profile packets in the queue that represents the lower bound at which green packets can be randomly dropped. Default value is 10.
GREEN-PROFILE-MAX-THRESHOLD [Number greater than zero]	Specifies the number of green profile packets in the queue that represents the upper bound at which green packets can be randomly dropped. Default value is 20.
GREEN-PROFILE-MAX-PROBABILITY [Number greater than 0.0 and less than 1.0]	Specifies maximum probability (0...1) value at which a packet can be dropped (before the queue is completely full). Default value is 0.02.
YELLOW-PROFILE-MIN-THRESHOLD [Number greater than zero]	Specifies the number of yellow profile packets in the queue that represents the lower bound at which green packets can be randomly dropped. Default value is 5.
YELLOW-PROFILE-MAX-THRESHOLD [Number greater than zero]	Specifies the number of yellow profile packets in the queue that represents the upper bound at which green packets can be randomly dropped. Default value is 10.
YELLOW-PROFILE-MAX-PROBABILITY [Number greater than 0.0 and less than 1.0]	Specifies maximum probability (0...1) value at which a packet can be dropped (before the queue is completely full). Default value is 0.02.
RED-PROFILE-MIN-THRESHOLD [Number greater than zero]	Specifies the number of red profile packets in the queue that represents the lower bound at which green packets can be randomly dropped. Default value is 2.
RED-PROFILE-MAX-THRESHOLD [Number greater than zero]	Specifies the number of red profile packets in the queue that represents the upper bound at which green packets can be randomly dropped. Default value is 5.
RED-PROFILE-MAX-PROBABILITY [Number greater than 0.0 and less than 1.0]	Specifies maximum probability (0...1) value at which a packet can be dropped (before the queue is completely full). Default value is 0.02.
RED-QUEUE-WEIGHT [Number greater than 0.0]	Specifies the queue weight used to determine the bias towards recent or historical queue lengths in calculating the average. Default value is 0.002.
RED-SMALL-PACKET-TRANSMISSION-TIME [Number greater than zero]	Specifies sample amount of time to transmit a small packet. Used to estimate the queue average during idle periods. Default value is 10MS.

GUI Configuration

Enabling RIO

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Network Protocol > Specify Priority-wise IP Queue Type? > IP Queue Type**. From the configurable property window, set **IP Queue Type** as **RIO**, as shown in Figure 78.

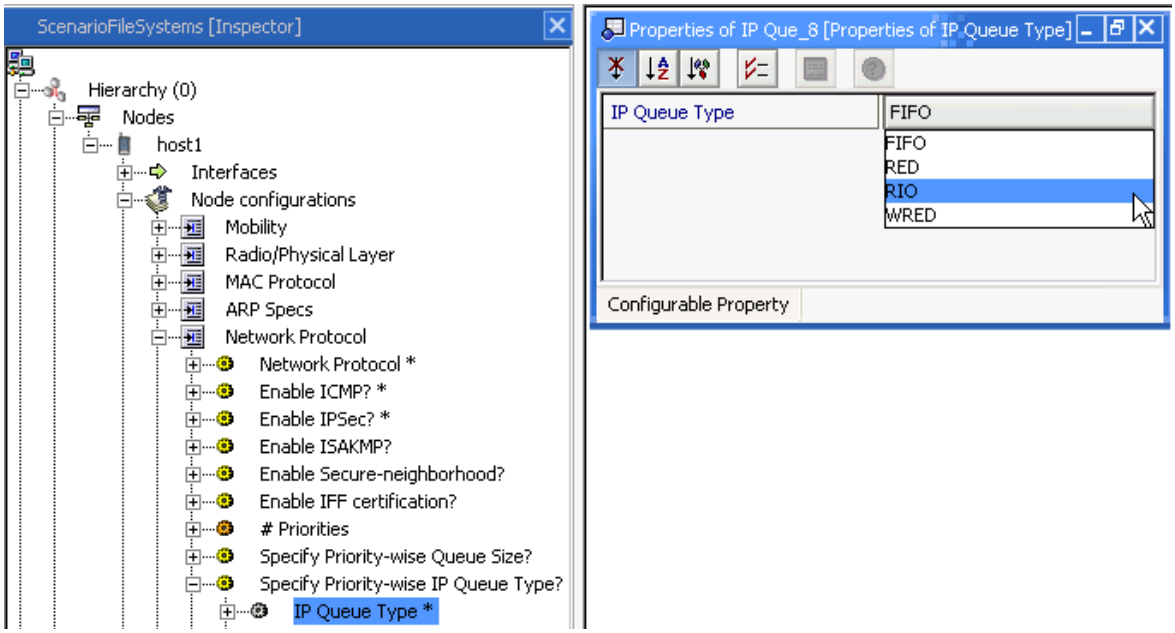


FIGURE 78. Enabling RIO for a Node

Configuring RIO Parameters

After enabling RIO, configure the parameters, as shown in Figure 79.

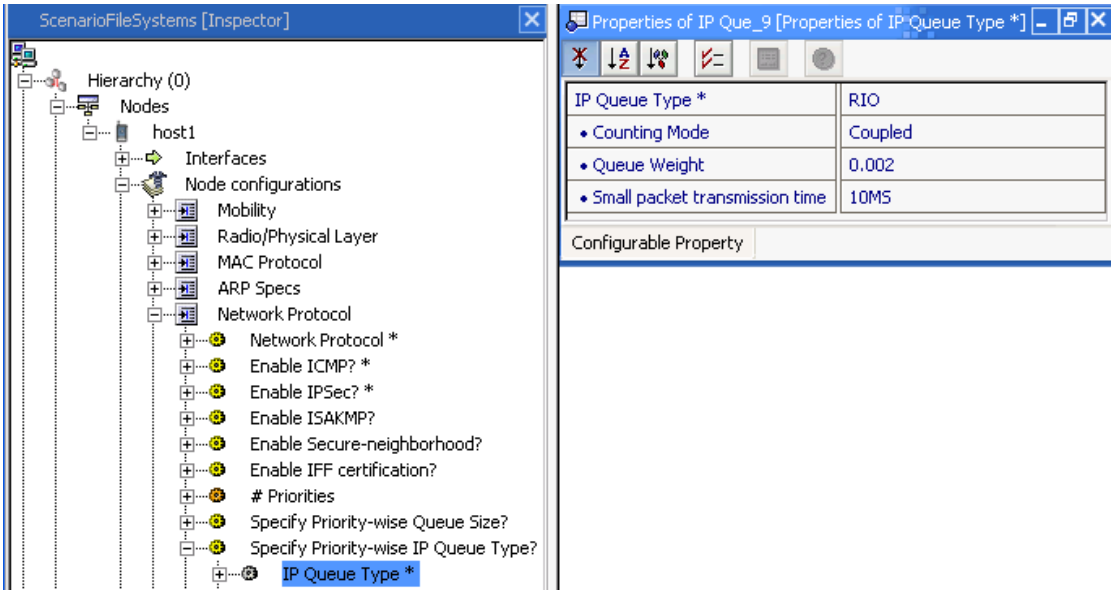


FIGURE 79. Configuring RIO Parameters

Configuring Color Mode

After enabling RIO, configure the Color Mode parameters, as shown in Figure 80.

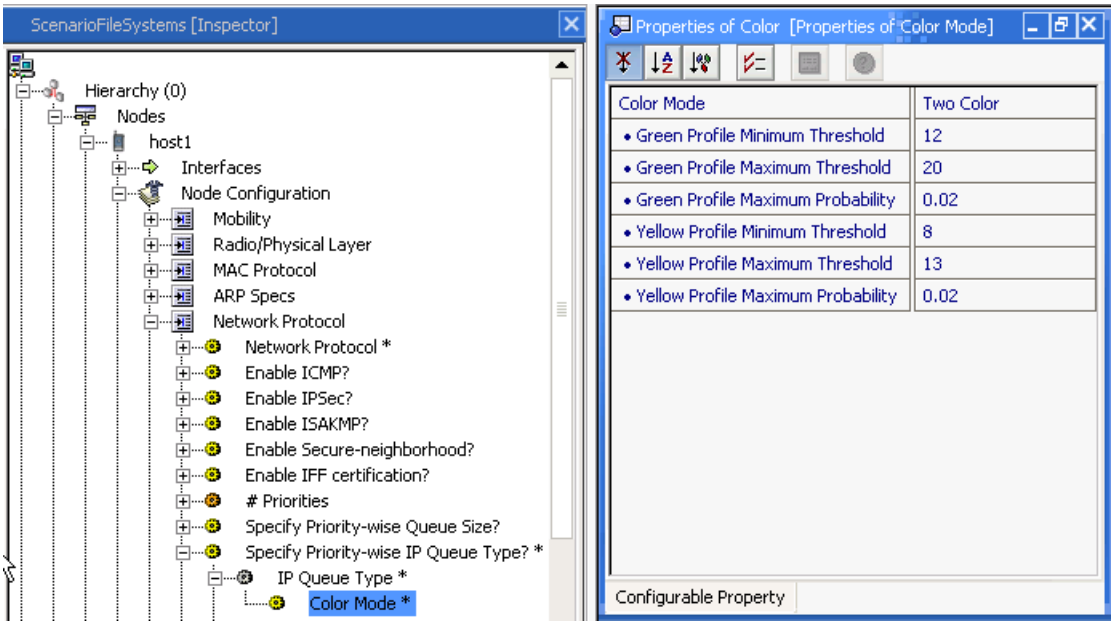


FIGURE 80. Configuring Color Mode

Statistics

RIO collects the statistics listed in Table 55, per priority queue, per node:

Note: In Table 55, <Profile> = GREEN | YELLOW | RED packet profile.

TABLE 55. RIO Statistics

Statistic	Definition
(<Profile>) Packets Queued	Total number of <profile> packets queued in this queue
(<Profile>) Packets Dequeued	Total number of <profile> packets dequeued from this queue
(<Profile>) Packets Marked ECN	Total number of <profile> packets marked by the Explicit Congestion Notification (ECN) capable network
(<Profile>) Packets Dropped	Total number of <profile> packets dropped by this queue
Total Packets Queued	Total number of packets queued in this queue
Total Packets Dequeued	Total number of packets dequeued from this queue
Total Packets Dropped	Total number of packets dropped by this queue
Total Packets Dropped Forcefully	Total number of packets dropped forcefully by this queue.
Average Queue Length (bytes)	Average of queue length reached by this queue during simulation
Average Time In Queue	Average delay suffered by the packets in this queue

TABLE 55. RIO Statistics

Longest Time in Queue	Longest delay suffered by a packet in this queue
Peak Queue Size (bytes)	Peak Queue Size reached by this queue during simulation
Packets Marked ECN	Total number of packets marked by ECN capable network

Random Node Placement Model

In the random node placement model, nodes are placed on the terrain randomly.

Command Line Configuration

To select this model, set the parameter `NODE-PLACEMENT` in the scenario configuration(.config) file as follows:

```
NODE-PLACEMENT    RANDOM
```

Statistics

There are no statistics collected for this model.

Round Robin Scheduler

The Round Robin Scheduler is an alternative scheduling discipline in QualNet. It services each priority queue, starting with the highest priority queue that contains packets, services a single packet, and moves to the next lower priority queue that contains packets, servicing a single packet from each, until each queue with packets has been serviced once. It then starts the cycle over with the highest priority queue containing packets.

Command Line Configuration

Selection of a scheduler is currently mandatory, and is achieved by placing the following entry in the *.config:

```
IP-QUEUE-SCHEDULER          ROUND-ROBIN
```

GUI Configuration

Configuring Round Robin Scheduler

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Network Protocol > IP Queue Scheduler**. From the configurable property window, set **IP Queue Scheduler** to **Round Robin** as shown in Figure 81.

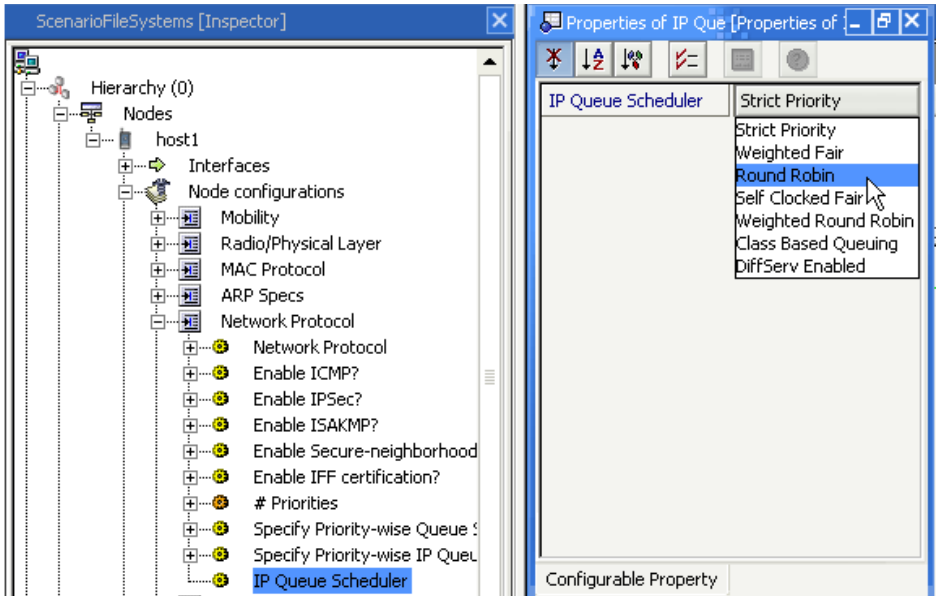


FIGURE 81. Configure Round Robin Scheduler

Statistics

The Round Robin Scheduler statistics generated are shown in Table 56:

TABLE 56. Round Robin Scheduler Statistics

Statistics	Description
Packets Queued	Total number of packets en queued in the queue
Packets Dequeued	Total number of packets de queued
Packets Dropped	Total number of packets dropped
Service Ratio received	Total number of packets de queued to total de queue requests

Routing Information Protocol/Routing Information Protocol version 2 (RIP/RIPv2)

RIP or RIPv2 are internet standard implementations of the Bellman-Ford (a.k.a. Ford Fulkerson) routing algorithm. It is a distance vector routing algorithm using the User Datagram Protocol (UDP) protocol for control packet transmission.

Command Line Configuration

RIP can be configured by placing the following parameter in *.config:

```
ROUTING-PROTOCOL RIP
```

Table 57 shows the parameters used in configuring RIP:

TABLE 57. RIP Parameters

Parameter	Description
RIP-VERSION 1|2	Appropriate version of RIP is selected based on the value. Default value is 2
SPLIT HORIZON [NO | SIMPLE | POISONED-REVERSE]	SPLIT HORIZON is a method to prevent route loops. The values are: NO - no split horizon SIMPLE - when sending route updates to a neighbor, omit those entries that learned from that neighbor (i.e. routes with that neighbor as the next hop) POISONED-REVERSE - Instead of omitting those routes as in the simple split horizon scheme, those route entries will be included in the route updates. However, the metrics of those routes will be set to infinity.

GUI Configuration

Configuration Requirements

IPv4 must be enabled in order to use RIP/RIPv2. Please refer to IPv4 section of this document.

Enabling RIP/RIPv2

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Routing Protocol > Routing Policy > Routing Protocol for IPv4**. From the configurable property window, set **Routing Protocol for IPv4** to **RIP**, as shown in Figure 82.

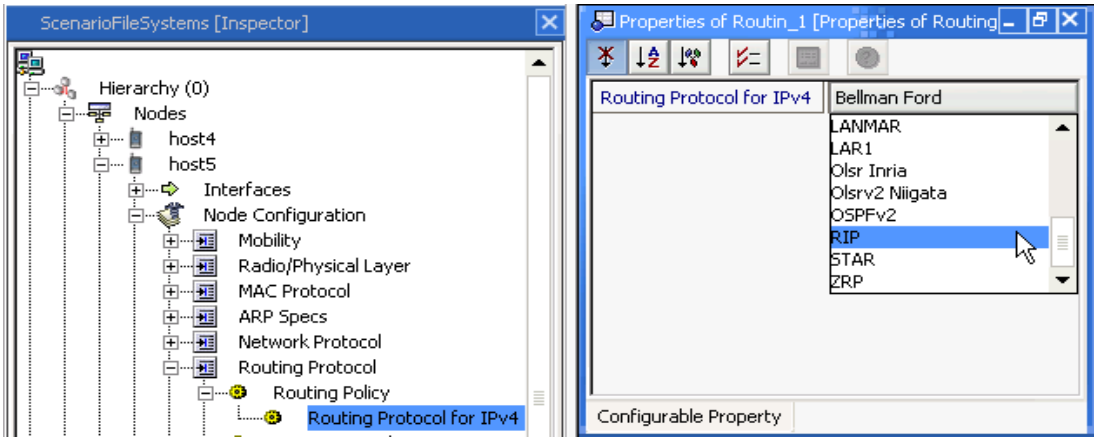


FIGURE 82. Enable RIP/RIPv2

Configuring General RIP Parameters

When RIP is enabled, a set of its parameters is displayed. Configure the RIP parameters as shown in Figure 83.

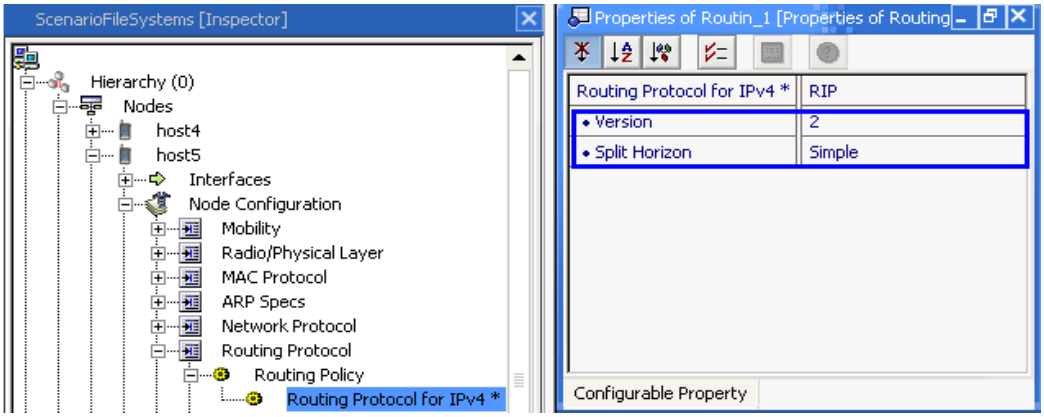


FIGURE 83. Configuring RIP/RIPv2 Parameters

Statistics

Table 58 shows the statistics collected by RIPv2:

TABLE 58. RIPv2 Statistics

Statistic	Description
Regular Update Events	Number of regular update events.
Triggered Update Events	Number of Triggered Update Events

TABLE 58. RIPv2 Statistics

Statistic	Description
Route Timeout Events	Number of Route Timeout Events
Response Packets Received	Number of Response Packets Received
Regular Update Packets Sent	Number of Regular Update Packets Sent
Triggered Update Packets Sent	Number of Triggered Update Packets Sent

Routing Information Protocol next generation (RIPng)

RIPng is a proactive *Interior Gateway Protocol* based on the distance-vector algorithm. RIPng is intended for use within the IPv6-based Internet.

As it is a distance-vector routing protocol, it forms routing tables by exchanging routing table information with each router. There are two types of updates. One is a Regular update, which is periodically sent and contains the whole routing table information. The other is a Triggered update, which is sent when a router's routing table changes and contains only those routing entities which have been modified. When a router receives a packet, it updates its routing table and if its routing table changed, it sends a triggered update to its neighbor router.

Command Line Configuration

To select RIPng as the routing protocol, specify the following parameter(s) in the scenario configuration (.config) file:

- For a dual IP-node, use the following parameter:

```
ROUTING-PROTOCOL-IPv6     RIPng
```

- For an IPv6 node, use *either* of the following parameters:

```
ROUTING-PROTOCOL          RIPng
```

or

```
ROUTING-PROTOCOL-IPv6     RIPng
```

Table 59 shows the parameters used by RIPng.

TABLE 59. RIPng Parameters

Parameters	Description
SPLIT-HORIZON NO | SIMPLE | POISONED-REVERSE	SPLIT HORIZON is a method to prevent route loops. The values are: NO - no split horizon, SIMPLE - when sending route updates to a neighbor, omit those entries that learned from that neighbor (i.e. routes with that neighbor as the next hop), POISONED-REVERSE - Instead of omitting those routes as in the simple split horizon scheme, those route entries will be included in the route updates. However, the metrics of those routes will be set to infinity. Default value is Simple.
ROUTING-STATISTICS Yes | NO	Selecting Yes means that simulation will provide the user with that statistics of the layer selected .Default is YES
TRACE-RIPNG <YES | NO>	It enables the trace of the protocol. Default value is YES

GUI Configuration

Configuration Requirements

IPv6 must be enabled in order to use RIPng. See the IPv6 section in this document for details.

Note: Routing protocols for IPv6 networks can only be configured at the subnet and interface levels when using the QualNet GUI.

Enabling RIPng

Go to **Hierarchy (x) > Nodes > host(x) > Interfaces > interface (x) > Interface Configuration > Routing Protocol > Routing Policy > Routing Protocol for IPv6**. From the configurable property window, set **Routing Protocol for IPv6** to **RIPng** as shown in Figure 84.

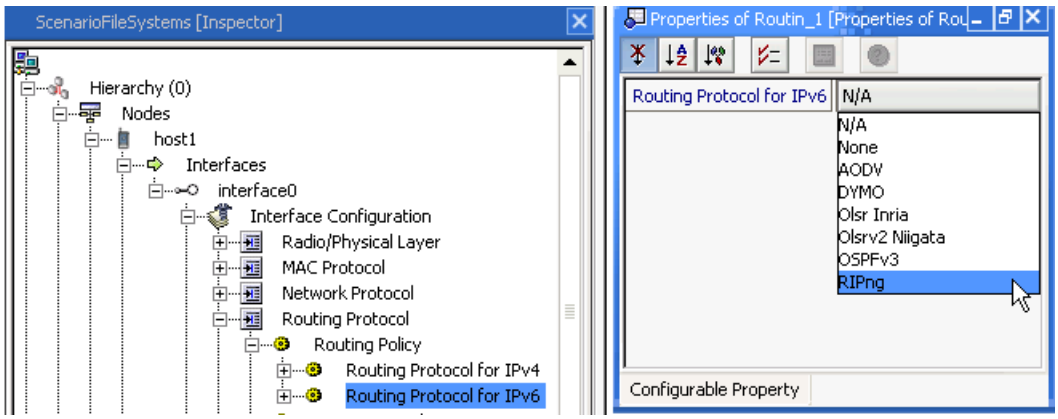


FIGURE 84. Enabling RIPng

Configuring General RIPng Parameters

When RIPng is enabled, a set of its parameters is displayed. Configure **RIPng** parameters as shown in Figure 85.

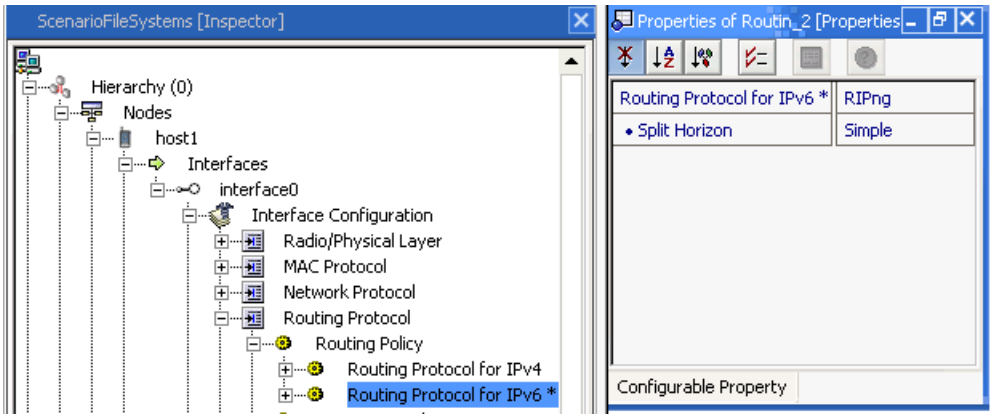


FIGURE 85. Configuring RIPng Parameters

Enabling Routing Statistics

Go to **ConfigSettings > Statistics > Statistics** and enable **Routing** statistics as shown in Figure 86.

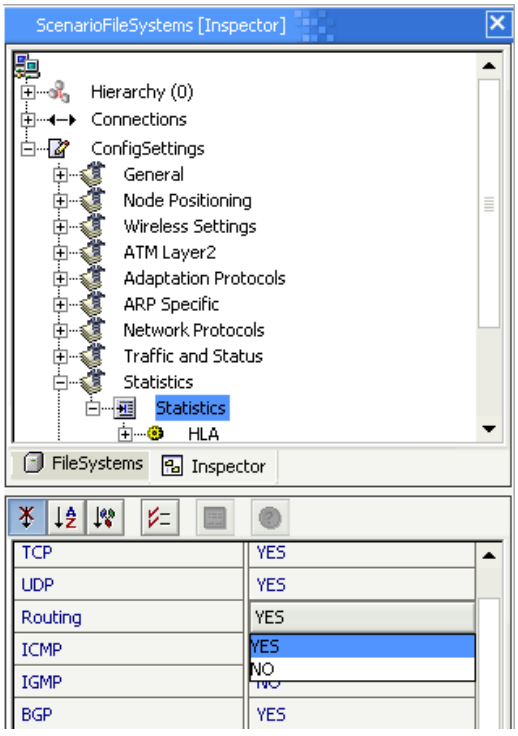


FIGURE 86. Enable Routing Statistics

Statistics

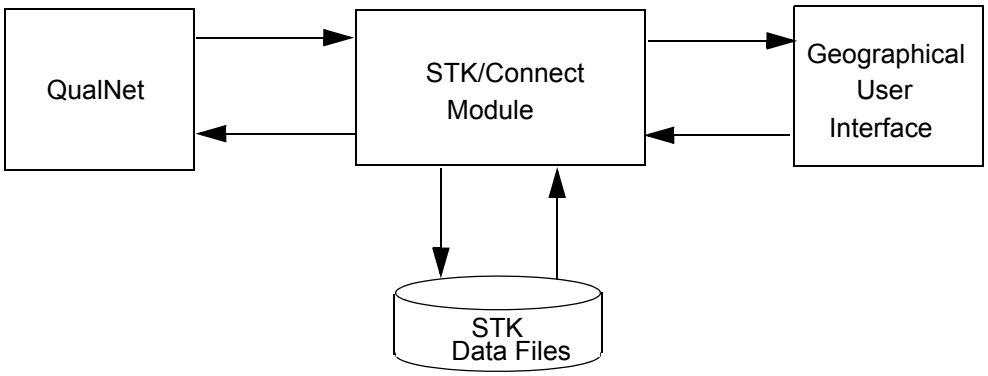
Table 60 shows the statistics collected by RIPng:

TABLE 60. RIPng Statistics Collected

Statistic	Description
RegularUpdateEvents	Number of regular update events.
TriggeredUpdateEvents	Number of triggered update events.
RouteTimeouts	Number of times route time out occurred.
ResponsesReceived	Number of response packets received.
RegularPacketSent	Number of regular updates packet sent.
TriggeredPacketSent	Number of triggered updates packet sent.

Satellite Toolkit/Connect (STK/Connect)

The QualNet/STK Integration module communicates with STK and STK/VO to visualize events in real-time. For instance, you can feed real-time telemetry data from the launch and early orbit of a mission. As a scenario, the data can be viewed in 2D or 3D to visualize the mission and assist in understanding and resolving any issues that may arise. The QualNet/STK Integration module provides a framework to interact with STK/VO through STK Connect. To create custom enhancements, you can easily add support for other STK/Connect features.



The QualNet/STK Integration module provides a way to interface QualNet with the Satellite Toolkit (STK) developed by Analytical Graphics, Inc. (AGI) and function in a client-server environment. This is achieved by using AGI's STK Connect library. The STK library contains messaging capabilities that allow third-party applications like QualNet to directly interact with STK.

QualNet/STK Integration

The QualNet/STK integration module has been tested under Windows. Linux is not supported. To compile and run the module, follow these steps:

1. Edit the file QUALNET_HOME/interfaces/stk/Makefile-windows to include the correct paths for the STK include directory and AgConnect.lib file.

   For default installation of STK 4.4, these would be:

```
ADDON_INCLUDES = $(ADDON_INCLUDES) $(STK_INCLUDES) \
                  -IC:\Progra~1\AGI\stk\4.4\Connect\Includes
ADDON_LIBRARIES = $(ADDON_LIBRARIES) \
                  C:\Progra~1\AGI\stk\4.4\bin4.4\AgConnect.lib
```

   For default installation of STK 7, these would be:

```
ADDON_INCLUDES = $(ADDON_INCLUDES) $(STK_INCLUDES) \
                 -IC:\Progra~1\AGI\STK7~1\Connect\Includes
ADDON_LIBRARIES = $(ADDON_LIBRARIES) \
                  C:\Progra~1\AGI\STK7~1\bin\AgConnect.lib
```

2. Update the environment variable path to reference AgConnect.dll.

3. In the file QUALNET_HOME/main/Makefile-addons-windows locate the following line (after the "INSERT INTERFACES HERE" comment) and uncomment it, i.e., change the line

```
#include ../addons/stk/Makefile-windows
```

to

```
include ../addons/stk/Makefile-windows
```

4. Recompile QualNet (by first "nmake clean" and then "nmake"). Use the makefile appropriate for your compiler. For example, for the VC7 compiler, use the following commands

```
nmake -f Makefile-windows-vc7 clean
nmake -f Makefile-windows-vc7
```

For more information on activating addon modules and compiling QualNet, see *QualNet 4.5 Installation Guide*.

Note: There is also an example of the QualNet/STK integration scenario located in QUALNET_HOME/addons/stk/sample directory. Start with the README file in the directory.

Command line Configuration

To enable QualNet to communicate with STK, STK-ENABLED is used. By default, this value is NO. Hence, STK is enabled by placing following entry in *.config. Table 61 shows the parameters for STK.

```
STK-ENABLED       YES
```

Note: The LATLONALT coordinate system must be used when communicating with STK**.**

TABLE 61. STK Parameters

Parameter	Description
STK-SCENARIO-NAME <scenario name>	Specifies the scenario name that will appear in STK, use STK-SCENARIO-NAME. The default value is "QualNet-STK". For example, to specify the scenario name as "TEST", use: STK-SCENARIO-NAME TEST
STK-FROMTO-VIEW < node id >	Specifies which node the STK viewpoint should use. The default value is 1. For example, to have the viewpoint set on node 7, use: STK-FROMTO-VIEW 7

TABLE 61. STK Parameters (Continued)

Parameter	Description
STK-NODE-TYPE < node type >	Specifies the STK node type by using STK-NODE-TYPE. The possible values are: STK-AIRCRAFT-NODE STK-FACILITY-NODE STK-GROUNDVEHICLE-NODE STK-SATELLITE-NODE STK-SHIP-NODE The default value is STK_GROUNDVEHICLE_NODE. For example, to have node 1 be an STK aircraft node, use: [1] STK-NODE-TYPE STK-AIRCRAFT-NODE
STK-NODE-NAME < node name >	Specifies the name of the STK node by using STK-NODE-NAME. The default value is "Node" followed by the node ID of the node. Note that the actually name that will appear in STK will include the node ID of the node appended to the end of the STK-NODE-NAME value. For example, to name node 1 as "Aircraft" (STK will show "Aircraft1"), use: [1] STK-NODE-NAME Aircraft
STK-NODE-MODEL <absolute location of 3D model file>	Specifies the 3D model to display for a node by using STK-NODE-MODEL. For example: STK-NODE-MODEL C:\Program Files\AGI\stk\4.4\STKData\VO\Models\Space\satellite.mdl
STK-NODE-MODEL-SCALE <scaling value>	If STK-NODE-MODEL is specified, the model scale can be adjusted by using STK-NODE-MODEL-SCALE. Default value is 1.0. For example, to specify a scale of 2.5, use: STK-NODE-MODEL-SCALE 2.5
STK-POSITION-UPDATE-INTERVAL <time interval>	Specifies how often to update node positions in STK by using STK-POSITION-UPDATE-INTERVAL. Default value is 60 seconds. For example, to update node positions every 10 seconds, use: STK-POSITION-UPDATE-INTERVAL 10S
STK-POSITION-OFFSET <offset value>	Specifies the offset positioning in STK.
STK-IP-STATISTICS <YES | NO>	To have STK keep track of IP stats using STK's external data feature, use STK-IP-STATISTICS. Default value is NO. For example, to enable IP statistics gathering, use: STK-IP-STATISTICS YES
STK-CBR-STATISTICS <YES | NO>	It is enabled to have STK keep track of CBR stats using STK's external data feature. Default value is NO.

Adding QualNet/STK Commands

The QualNet/STK Integration module allows QualNet to control the 3D animation capabilities of STK through STK's Connect module. STK Pro, VO and Connect modules are required. Note that currently this is a one-way communication from QualNet to STK, not from STK to QualNet (i.e., not bidirectional). Furthermore, the integration module is meant as an illustration of how QualNet can be used to communicate with STK and therefore uses only a small number of features. You must expand on this to

meet your own objectives. This section provides some reference points in facilitating further feature improvement with the QualNet/STK integration module.

Communication with STK Connect

AgConProcessSTKCmd(<connection>, <command>, <feedback>)

where:

<connection> is the connection name used to communicate with STK Connect module

<command> is the STK Connect recognized command

<feedback> is the STK Connect AgTConReturnInfo variable type

```
AgConProcessSTKCmd(partitionData->stkConnection,
                   "Animate * Start",
                   &returnInfo);// AgTConReturnInfo returnInfo;
```

Useful QualNet/STK Variables

partitionData->stkConnection

Connection name, used to communicate with STK.

partitionData->stkScenarioName

Value obtained from STK-SCENARIO-NAME.

partitionData->stkEnabled

TRUE if STK-ENABLED is YES, FALSE otherwise.

node->externalData[0]->stk.nodeType

Value obtained from STK-NODE-TYPE. Used to compare with the following enumerations:

STK_AIRCRAFT_NODE
STK_FACILITY_NODE
STK_GROUNDVEHICLE_NODE
STK_SATELLITE_NODE
STK_SHIP_NODE

node->externalData[0]->stk.nameStr

Used to reference a node in STK VO

Value is <nodeType>/<nodeName><nodeId>

<nodeType> can be:

Aircraft
Facility
GroundVehicle
Satellite
Ship

<nodeName> is obtained from STK-NODE-NAME

<nodeId> is from node->nodeId

Example: GroundVehicle/Tank1

Statistics

There are no statistics collected by STK/Connect.

Sample Scenario

The sample scenario is located in QUALNET_HOME/addons/stk/sample.

```
STK-ENABLED            YES
STK-SCENARIO-NAME      QualNet-STK-Example
STK-NODE-MODEL         C:\Program
Files\AGI\stk\4.4\STKData\VO\Models\Ground\Vehicles\groundvehicle.mdl
STK-NODE-MODEL-SCALE 0
```

1. Start STK.
2. Run QUALNET_HOME/bin/qualnet stk.config.
   See Figure 87 and Figure 88.

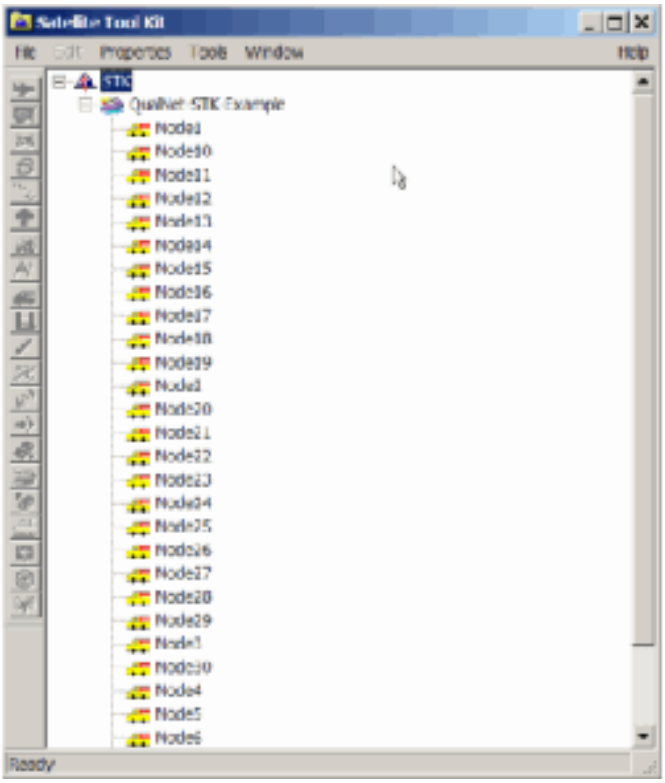


FIGURE 87. **Node Creation in STK Driven by QualNet**

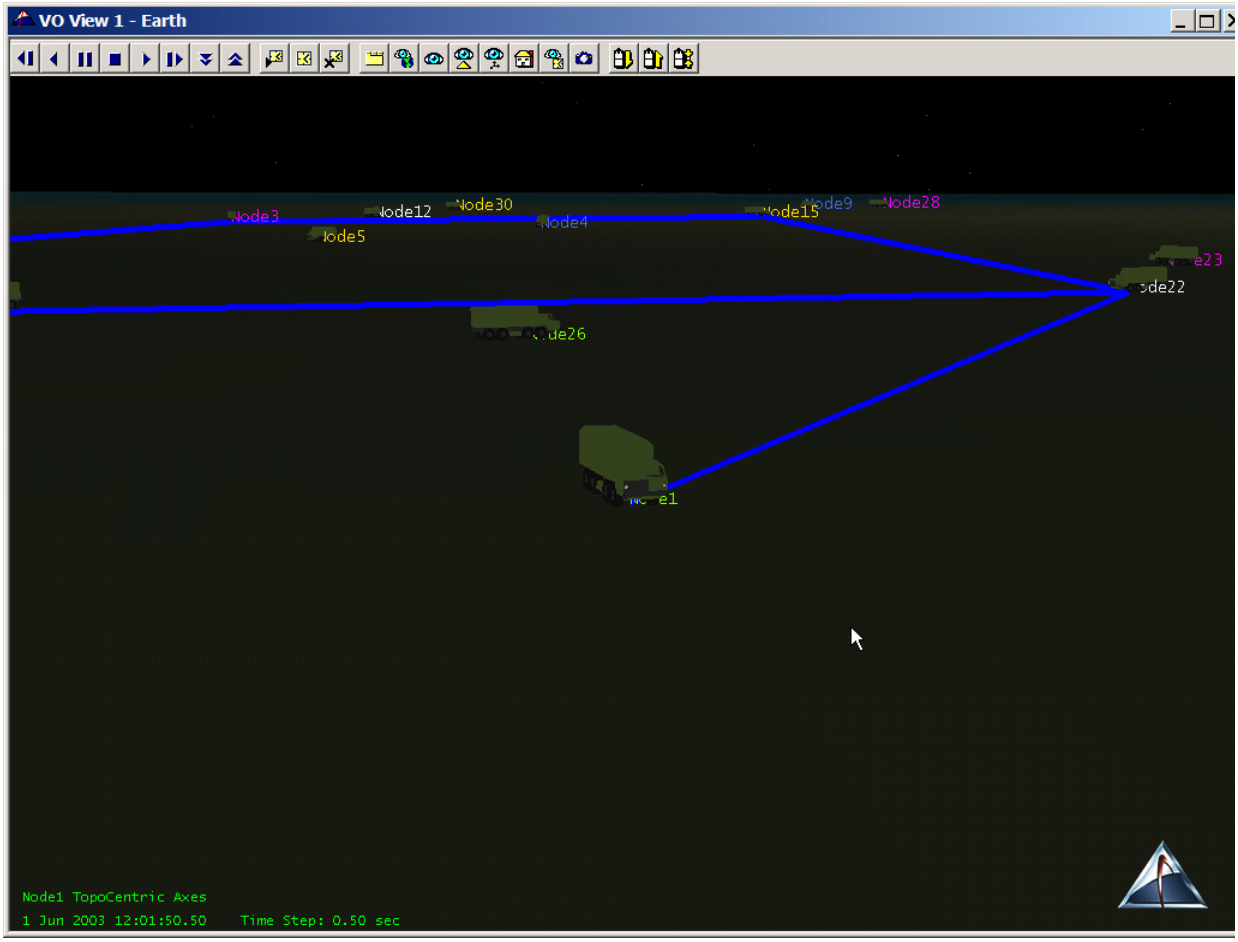


FIGURE 88. STK VO Screen Shot

Blue lines represent IP packets received.

Changing Parameters

Change 3D model for node 1:

```
[1] STK-NODE-MODEL C:\Program
Files\AGI\stk\4.4\STKData\VO\Models\Ground\Vehicles\aaa-mobile.mdl
```

Change 3D model size for node 1:

```
[1] STK-NODE-MODEL-SCALE 1
```

Change STK-POSITION-UPDATE-INTERVAL values to see impact on runtime performance (default is 60 seconds):

```
STK-POSITION-UPDATE-INTERVAL 5S
```

Self-Clocked Fair Queueing (SCFQ) Scheduler

SCFQ is an alternative scheduling discipline in QualNet. SCFQ is similar to Weighted Fair Queueing (WFQ) in its attempt to service each priority queue based on a percentage allocation of the total outgoing bandwidth of the link. However, it was discovered that calculating the Finish Time (the time at which a packet would have been serviced given a hypothetical fluid server) for WFQ was complicated/difficult. SCFQ uses a simplified method of calculating the service time, based on the transmission delay of the packet, and the finish time of the packet currently being serviced.

Command Line Configuration

Selection of a scheduler is currently mandatory, and is achieved by placing the following entry in the *.config:

```
IP-QUEUE-SCHEDULER SELF-CLOCKED-FAIR
```

Note: If no parameters are given, the weight of each queue is the ((priority value of the queue)+ 1) / (sum of all the priority_values for all the queues). Otherwise, users must specify the weight for all of the queues on that interface, such that the sum of the weights equals one, using the following parameter:

```
QUEUE-WEIGHT[priority]      <weight>
```

Where priority is the integer value of the priority assigned to each queue (numbered from 0...num_of_priority_queues-1), and the weight is a floating point number between zero and one. This parameter can also be specified using node-specific or network-specific parameters.

GUI Configuration

Configuring Self Clocked Fair Queueing Scheduler

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Network Protocol > IP Queue Scheduler**. From the configurable property window, set **IP Queue Scheduler** to **Self Clocked Fair** as shown in Figure 89.

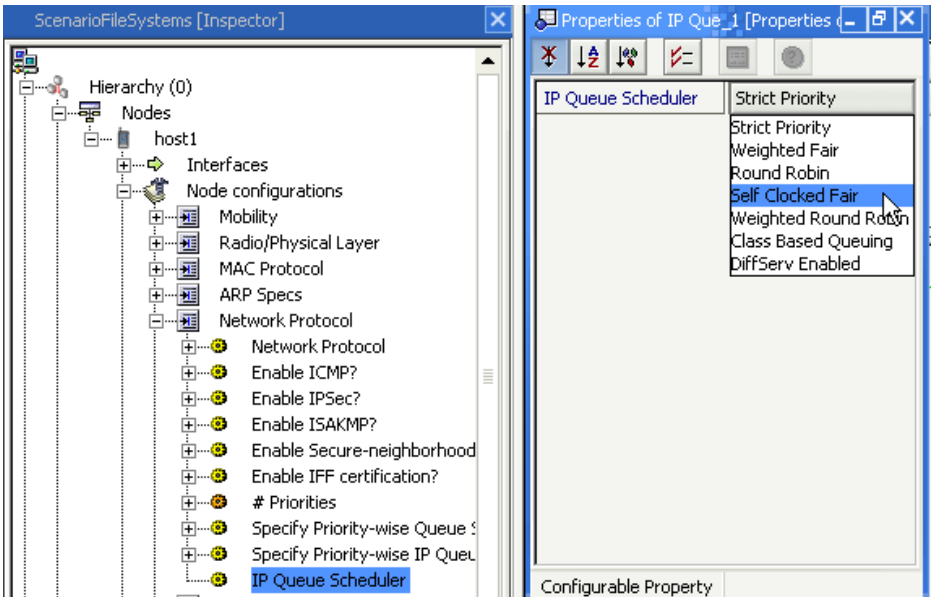


FIGURE 89. Configuring Self Clocked Fair Queueing Scheduler

Statistics

Table 62 shows the statistics generated by SCFQ:

TABLE 62. SCFQ Statistics

Statistics	Description
Packets Queued	Total number of packets en queued in the queue
Packets Dequeued	Total number of packets de queued
Packets Dropped	Total number of packets dropped
Dequeue Requests While Queue Active	Total number of dequeue requests while the queue is active.
Dequeue Requests Serviced	Total number of de queue requests services
Service Ratio received	Total number of packets de queued to total de queue requests
Total Service Received(Bytes)	Total service received in bytes.

Static and Default Routes

QualNet allows users to specify static routes that take precedence over those discovered by any routing protocols. These routes remain fixed throughout the simulation, even if network conditions (mobility, link failures) render the route useless, causing packet drops. There are also no route acquisition related overheads with static routes, such as packet transmissions, receptions, computational overhead for producing routes, or route acquisition delays that would normally be present using a traditional routing model. If no routing protocol is running on the node, and there is no static or default route to a given destination, corresponding packets will be dropped.

If a node is configured to use an on-demand routing protocol, then only routes discovered by the routing protocol are used even if static and/or default routes are enabled at the node.

If a node is configured to use a proactive routing protocol and static routes are enabled at the node, then the static routes take precedence over routes discovered by the routing protocol.

If static routes are disabled at a node and the node is not configured to use any routing protocol, then default routes are used.

Command Line Configuration

Both the static route file and the default route file have the same format. Each line in either of these files has the following syntax:

```
<source nodeId> <destination IP/Subnet Address> <nextHop IP Address>
```

To specify that static and/or default routes are enabled for a node, place the entries in the configuration file as shown in Table 63.

TABLE 63. Static and Default Routing Parameters

Parameters	Description
STATIC-ROUTE [YES | NO]	Enable this parameter to configure static route. Note: This parameter is mandatory if static route is enabled.
STATIC-ROUTE-FILE <filename>	Name of the Static route file. For example, STATIC-ROUTE-FILE default.routes-static Note: This parameter is mandatory if static route is enabled.
DEFAULT-ROUTE [YES | NO]	Enables the Default Route. Note: This parameter is mandatory if default route is enabled.
DEFAULT-ROUTE-FILE <filename>	Name of the Default route file. For example, DEFAULT-ROUTE-FILE default.routes-default Note: This parameter is mandatory if default route is enabled

IPv4

In the following example, the route for node 1 to the networks 0.0.2.0 and 0.0.3.0 uses node 1's outgoing interface identified by the IP Address 0.0.1.2.

```
1 N2-2.0 1.2
1 N2-3.0 1.2
```

IPv6

The following example specifies static routes for IPv6 network for the node 1 to networks TLA-0.NLA-0.SLA-2 and TLA-0.NLA-0.SLA-3. Use node 1's outgoing interface identified by the IPv6 Address 2000::1:0:0:0:2.

```
1 TLA-0.NLA-0.SLA-2 2000::1:0:0:0:2
1 TLA-0.NLA-0.SLA-3 2000::1:0:0:0:2
```

GUI Configuration

Enabling Static Routes

Go to **Hierarchy (x) > Nodes > host[x] > Node configurations > Routing Protocol > Static Routes**. From the configurable property window, set **Static Routes** to **Use Static Routes**.

Selecting Static Route File

Select the appropriate static route file.

Enabling Default Routes

Go to **Hierarchy (x) > Nodes > host[x] > Node configurations > Routing Protocol > Default Routes**. From the configurable property window, set **Default Route**s to **User Specified** or **System Specified**.

Selecting Default Route File

If **Default Route**s is set to **User Specified**, then specify the default route file.

Statistics

There are no statistics generated for Static and Default routes.

Static Multicast Routes

Multicast static routes are user-configured multicast routes. User can configure these routes in multicast static route file. QualNet Static Multicast model supports both IPv4 and IPv6 configuration. Following are the list of configuration parameters to configure static multicast routes.

Command Line Configuration

Table 64 lists the parameters for the static multicasting.

TABLE 64. Static Multicasting Parameters

Parameter Name	Description
MULTICAST-STATIC-ROUTE [YES | NO]	This parameter is used to enable static multicasting.
MULTICAST-STATIC-ROUTE-FILE <filename>	Specifies the name of the static multicast route file. This parameter must be specified if MULTICAST-STATIC-ROUTE is set to YES

Note: All parameters are configurable at node level only.

Format of the Static Multicast Route File

Each line of the static multicast route file has the following format:

```
<nodeID> <source-address> <multicast-address> <interface-addresses>
```

where:

<nodeID> : Node ID.
<source-address> : Source address.
<multicast-address> : Destination multicast group address.
<interface-addresses> : List of space-separated outgoing interface addresses.

Examples:

1. The following lines show a segment of a static multicast route file for IPv6:

```
1 1000:1::1  ff12::3  1000:1::1
2 1000:1::1  ff12::3  1000:2::1  1000:5::1
```

   Node 1 will forward each multicast packet from source 1000:1::1 to multicast group destination ff12::3 on outgoing interface 1000:1::1.

   Node 2 will forward each multicast packet from source 1000:1::1 to multicast group destination ff12::3 on outgoing interfaces 1000:2::1 and 1000:5::1.

2. The following lines show a segment of a static multicast route file for IPv4:

```
1 192.168.0.1 225.0.0.1 192.168.0.1
2 192.168.0.1 225.0.0.1 192.168.0.2 192.168.1.2
```

   Node 1 will forward each multicast packet from source 192.168.0.1 to multicast group destination 225.0.0.1 on outgoing interface 192.168.0.1.

   Node 2 will forward each multicast packet from source 192.168.0.1 to multicast group destination 225.0.0.1 on outgoing interfaces 192.168.0.2 and 192.168.1.2.

GUI Configuration

Enabling Multicast

Go to **Hierarchy (x) > Nodes > host(x) > Node configurations > Routing Protocol > Enable Multicast?** Set **Enable Multicast?** to **Yes** as shown in Figure 90.

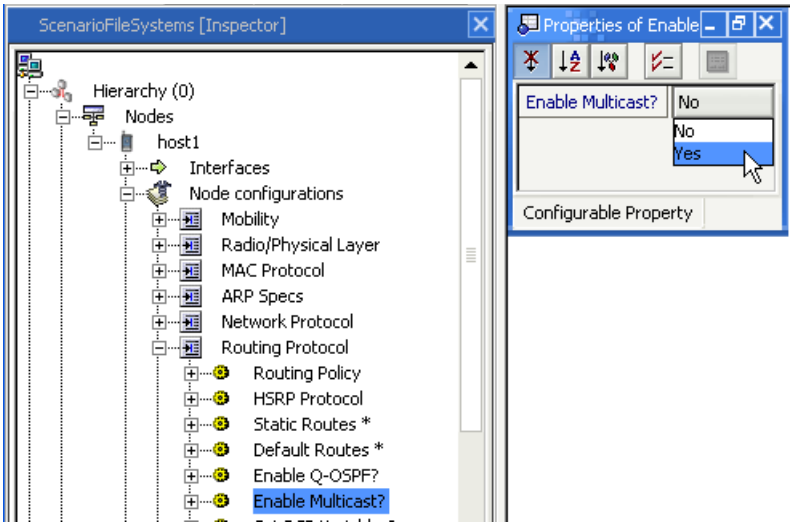


FIGURE 90. Enabling Multicast for a Node

Configuring Static Multicast Routes Parameters

Set **Static Multicast Routes** to **Use Static Multicast Routes** as shown in Figure 91.

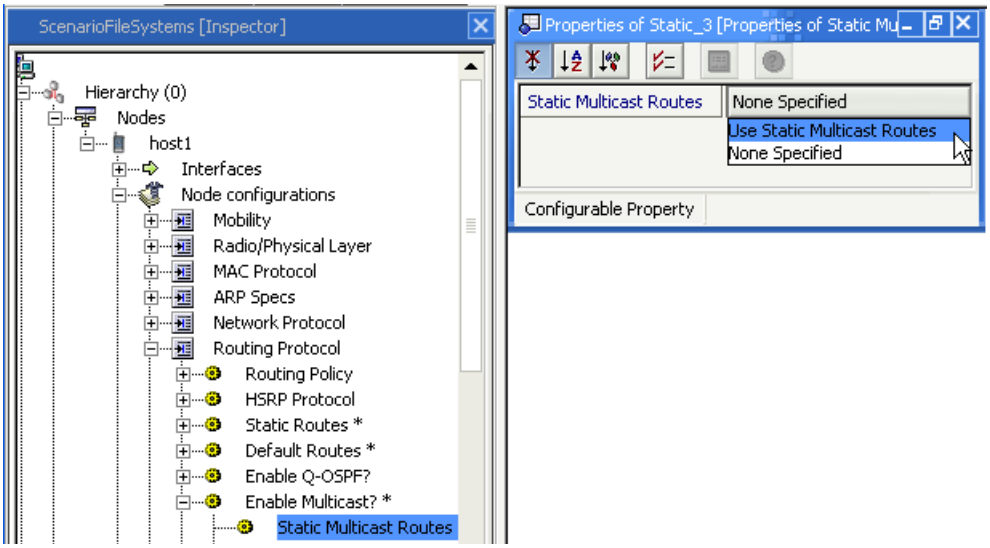


FIGURE 91. Enabling Static Multicast Routes for a Node

Specifying Static Multicast Route File

Specify the **Static Multicast Route File** as shown in Figure 92.

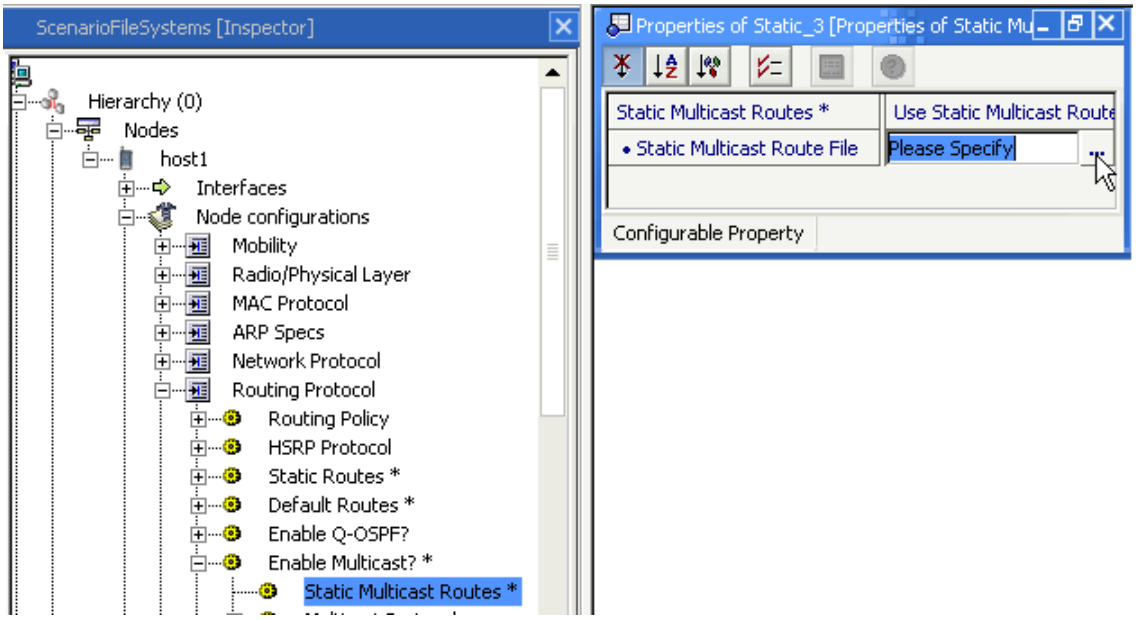


FIGURE 92. Configuring General Parameters of Static Multicast Routes for a Node

Strict Priority Scheduler

The Strict Priority Scheduler is the default scheduling discipline in QualNet. It services the highest priority queue until it is empty, and then moves to the next highest priority queue, and so on. It is possible that if there is enough high priority traffic, the lower priorities could be completely frozen out.

Command Line Configuration

Selection of a scheduler is currently mandatory, and is achieved by placing the following entry in the *.config:

```
IP-QUEUE-SCHEDULER     STRICT-PRIORITY
```

Statistics

The statistics generated by the Strict Priority Scheduler are shown in Table 65.

TABLE 65. Strict Priority Scheduler Statistics

Statistics	Description
Packets Queued	Gives the total number of packets en queued in the queue
Packets Dequeued	Gives the total number of packets de queued
Packets Dropped	Gives the total number of packets dropped

Super Application Traffic Generator

Super Application is a generic traffic generator. Super Application can simulate both UDP and TCP flows. Two-way flow (request-response) is supported for UDP based applications. Request packets travel from source to destination, whereas response packets travel from destination to source.

Command Line Configuration

To use Super Application, the following format is needed:

```
SUPER-APPLICATION <src> <dest> <keyword 1> <value 1>
<keyword 2> <value 2> .....
<keyword n> <value n>
```

Table 66 shows the parameters for Super Application.

TABLE 66. Super Application Parameters

Parameter	Description
<src>	Specifies the source node's nodeId or IP Address. This parameter is mandatory
<dest>	Specifies the destination node's nodeId or IP Address. This parameter is mandatory.
<keyword 1>	Specifies the first option name. This parameter is mandatory.
<value 1>	Specifies the first option value. This parameter is mandatory.
<keyword 2>	Specifies the second option name.
<value 2>	Specifies the second option value.
<keyword n>	Specifies nth option name.
<value n>	Specifies nth option value.

Table 67 describes the functions of required keyword pairs:

TABLE 67. Required Keyword Pairs

Keyword	Value	Function
DELIVERY-TYPE	<RELIABLE | UNRELIABLE	Whether to use TCP or UDP as the underlying transport protocol.
START-TIME	<time distribution type>	Determines when application should start sending packets.
DURATION	<time distribution type>	Duration of the application session.
REQUEST-NUM	<int distribution type>	Number of request packets to send.
REQUEST-SIZE	<int distribution type>	Size of each request packet.
REQUEST-INTERVAL	<time distribution type>	Inter-departure time between request packets.

TABLE 67. Required Keyword Pairs (Continued)

REQUEST-TOS	[TOS <tos value> | DSCP <dscp value> | PRECEDENCE <precedence value>]	The type of service a request packet receives. <tos value> is the value of the TOS bits of the IP header. <dscp value> is the value of the DSCP bits of the IP header. <precedence-value> is the contents of the PRECEDENCE bits of IP header. When no REQUEST-TOS is specified, default value of 0 is used.
REPLY-TOS	[TOS <tos value> | DSCP <dscp value> | PRECEDENCE <precedence value>]	The type of service a request packet receives. <tos value> is the value of the TOS bits of the IP header. <dscp value> is the value of the DSCP bits of the IP header. <precedence-value> is the contents of the Precedence bits of IP header. When no REQUEST-TOS is specified, default value of 0 is used.
REPLY-PROCESS	<YES | NO>	Specifies that a response is needed for each request. When REPLY-PROCESS is Yes, all reply-related keywords are required.
REPLY-NUM	<int distribution type>	The number of reply packets to send in response to a request.
REPLY-SIZE	<int distribution type>	Size of reply packets.
REPLY-PROCESS-DELAY	<time distribution type>	The wait time at the destination before sending a reply in response to a request.
REPLY-INTERDEPARTURE-DELAY	<time distribution type>	The delay between reply packets.
FRAGMENT-SIZE	<bytes>	Maximum application fragment size. Packets will be broken into multiple fragments when packet size exceeds FRAGMENT-SIZE.
SOURCE-PORT	<port number>	Port number to be used at the source. The application chooses a free source port when user does not explicitly specify a source port. User-specified ports must be unique.
DESTINATION-PORT	<port number>	Port number to be used at the destination. The application uses its default well known port when destination port is not specified by the user. User- specified ports must be unique.

```
where:
<time distribution type> ::= DET <time value> |
                             UNI <time value> <time value> |
                             EXP <time value>
<time value> ::= integer [ NS | MS | S | M | H | D ]
<int distribution type> ::= DET integer |
                            UNI integer integer |
                            EXP integer
```

Statistics

Table 68 shows the statistics collected for Super Application:

TABLE 68. Super Application Statistics

Statistic	Description
Super Application Client	
Client Address	The address of the client
Connection	Connection ID (Node numbers and ports)
First request fragment sent at (s)	Time in second when first request fragment was sent
Last request fragment sent at (s)	Time in second when last request fragment was sent
First reply fragment sent at (s)	Time in second when first reply fragment was sent (UDP)
Last reply fragment sent at (s)	Time in second when last reply fragment was sent (UDP)
First request fragment received at (s)	Time in second when first request fragment was received
Last request fragment received at (s)	Time in second when last request fragment was received
First reply fragment received at (s)	Time in second when first reply fragment was received
Last reply fragment received at (s)	Time in second when last reply fragment was received
Average end-to-end delay for request received	Average delay for request packets received
Average end-to-end delay for reply received	Average delay for reply packets received
Average jitter for request received	Average jitter for request packets received
Average jitter for replies received	Average jitter for reply packets received
Number of request bytes sent	Total number of request bytes sent
Number of request packets sent	Total number of request packets sent
Number of request fragments sent	Total number of request fragments sent
Number of reply bytes sent	Total number of reply bytes sent
Number of reply packets sent	Total number of reply packets sent
Number of reply fragments sent	Total number of reply fragments sent
Number of out of order reply fragments received (UDP)	Total number of out of order reply fragments received
Number of complete reply packets received	Total number of complete reply packets received
Total number of reply fragments received (UDP)	Total number of reply fragments received
Total reply fragment bytes received	Total number of reply fragment bytes received (UDP)
Number of out of order request fragments received	Total number of out of order request fragments received
Number of complete request packets received	Total number of complete request packets received
Total number of request fragments received	Total number of request fragments received
Total request fragment bytes received	Total number of request fragment bytes received
Throughput for requests sent (bits/s)	Throughput value for request packets sent
Throughput for requests received (bits/s)	Throughput value for request packets received
Throughput for replies sent (bits/s)	Throughput value for reply packets sent

TABLE 68. Super Application Statistics (Continued)

Statistic	Description
Throughput for replies received (bits/s) (UDP)	Throughput value for reply packets received
Super Application Server	
Server Address	The address of the server.
Connection	Connection ID (Node numbers and ports)
First request fragment sent at (s)	Time in second when first request fragment was sent
Last request fragment sent at (s)	Time in second when last request fragment was sent
First reply fragment sent at (s)	Time in second when first reply fragment was sent (UDP)
Last reply fragment sent at (s)	Time in second when last reply fragment was sent (UDP)
First request fragment received at (s)	Time in second when first request fragment was received
Last request fragment received at (s)	Time in second when last request fragment was received
First reply fragment received at (s)	Time in second when first reply fragment was received
Last reply fragment received at (s)	Time in second when last reply fragment was received
Average end-to-end delay for request received	Average delay for request packets received
Average end-to-end delay for reply received	Average delay for reply packets received
Average jitter for request received	Average jitter for request packets received
Average jitter for replies received	Average jitter for reply packets received
Number of request bytes sent	Total number of request bytes sent
Number of request packets sent	Total number of request packets sent
Number of request fragments sent	Total number of request fragments sent
Number of reply bytes sent	Total number of reply bytes sent
Number of reply packets sent	Total number of reply packets sent
Number of reply fragments sent	Total number of reply fragments sent
Number of out of order reply fragments received (UDP)	Total number of out of order reply fragments received
Number of complete reply packets received	Total number of complete reply packets received
Total number of reply fragments received (UDP)	Total number of reply fragments received
Total reply fragment bytes received	Total number of reply fragment bytes received (UDP)
Number of out of order request fragments received	Total number of out of order request fragments received
Number of complete request packets received	Total number of complete request packets received
Total number of request fragments received	Total number of request fragments received
Total request fragment bytes received	Total number of request fragment bytes received
Throughput for requests sent (bits/s)	Throughput value for request packets sent
Throughput for requests received (bits/s)	Throughput value for request packets received
Throughput for replies sent (bits/s)	Throughput value for reply packets sent
Throughput for replies received (bits/s) (UDP)	Throughput value for reply packets received

Examples

Table 69 examples of Super Application configuration.

TABLE 69. Super Application Examples

Configuration	Description
SUPER-APPLICATION 1 2 DELIVERY-TYPE RELIABLE START-TIME DET 60S DURATION DET 10S REQUEST-NUM DET 5 REQUEST-SIZE DET 1460 REQUEST-INTERVAL DET 0S REQUEST-TOS PRECEDENCE 0 REPLY-PROCESS NO	Node 1 sends to node 2 five items of 1460B each using TCP from 60 seconds simulation time till 70 seconds simulation time. If five items are sent before 10 seconds elapsed, no other items are sent. Precedence is set to 0.
SUPER-APPLICATION 1 2 DELIVERY-TYPE UNRELIABLE START-TIME DET 10S DURATION DET 21S REQUEST-NUM DET 3 REQUEST-SIZE DET 123 REQUEST-INTERVAL DET 1S REQUEST-TOS PRECEDENCE 1 REPLY-PROCESS YES FRAGMENT-SIZE 200 DESTINATION-PORT 1751 SOURCE-PORT 2345 REPLY-NUM DET 2 REPLY-SIZE DET 140 REPLY-PROCESS-DELAY DET 1S REPLY-INTERDEPARTURE-DELAY DET 0.1S	Node 1 sends to node 2 three (request) packets of 123 bytes each using UDP. Node 1 starts sending at 10 seconds into the simulation and sends for the duration of 21 seconds. Packets are sent with a 1 second interval with precedence set to 1. Node 1 uses port 2345 whereas Node 2 uses port 1751. The fragment size is bigger than packet size, so no fragmentation takes place. On receiving the request packets, Node 2 replies with 2 packets of 140 bytes each, for every request packet received. On receiving a request packet, node 2 waits 1 second before sending the first of 2 reply packets for that request packet. The second reply packet is sent 0.1 seconds after the first one. Behavior and Limitations Response is only supported with DELIVERY-TYPE UNRELIABLE. Default value for REPLY-PROCESS is NO. Replies are not processed unless this is explicitly set to Yes. Both client and server print out all possible statistics even if they are not collected. This will be changed in the next release. Not collected statistics are set to zero. Example: The client reports "Number of reply packets sent = 0" since this stat is only collected at the server. The REPLY-INTERDEPARTURE-DELAY and REQUEST-INTERVAL is applicable between packets and not fragments. If fragmentation occurs then all fragments are sent back to back. 0 second duration indicates "till end of simulation".

Telecommunications Network (TELNET)

TELNET represents the clear-text terminal server and client. The typing rate and sizes of server responses are taken from distributions created from network traces in the TCPlib library.

Command Line Configuration

To specify TELNET traffic in the application configuration file (default.app), specify the parameters according to the following format:

```
TELNET <src> <dest> <session duration> <start time>
```

Table 70 shows the parameters for TELNET. All parameters are mandatory.

TABLE 70. TELNET Parameters

Parameter	Description
<src>	Refers to the client node's nodeID or IP Address.
<dest>	Refers to the server node's nodeID or IP Address.
<session_duration>	Indicates the length of the entire session.
<start_time>	Indicates when the transmissions should begin.

Table 71 shows TELNET examples.

TABLE 71. TELNET Examples

Example	Description
TELNET 1 2 5M 50S	Node 1 initiates a 5-minute TELNET session to node 2, and begins the session 50 simulation seconds into the total simulation time
TELNET 1 2 0S 0S	Node 1 sends node 2 TELNET traffic for a duration randomly determined by TCPlib at the start of the simulation.

GUI Configuration

To configure TELNET parameters, go to **Connections > TELNET [x] -> [y]** and set the parameters, as shown in Figure 93.

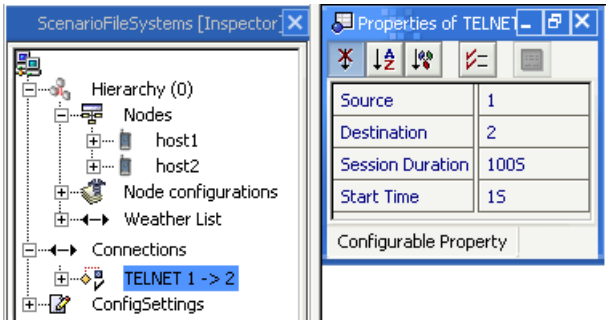


FIGURE 93. **TELNET Properties**

Statistics

The TELNET model collects the statistics shown in Table 72.

TABLE 72. **TELNET Statistics**

Statistics	Description
TELNET Client	
Server address	Specifies the server address
First Packet Sent at (s)	Specifies the time when the first packet is sent
Last Packet Sent at (s)	Specifies the time when the last packet is sent
Session Status	Specifies whether the session status is open or closed
Total Bytes Sent	Specifies the total number of bytes sent
Total Bytes Received	Specifies the total number of bytes received
Throughput (bits/s)	Specifies the total throughput of the client
TELNET Server	
Statistics	Description
Client Address	Specifies the client address
First Packet Sent at (s)	Specifies the time when the first packet is sent
Last Packet Sent at (s)	Specifies the time when the last packet is sent
Session Status	Specifies whether the session status is open or closed
Total Bytes Sent	Specifies the total number of bytes sent
Total Bytes Received	Specifies the total number of bytes received
Throughput (bits/s)	Specifies the total throughput of the server

Trace File-based Traffic Generator (Traffic-Trace)

Traffic-Trace simulates a trace file-based traffic generator. Trace files must have two columns; the first for time interval between subsequent data, and the second for each data length.

Command Line Configuration

To use Traffic-Trace, use the following format:

```
TRAFFIC-TRACE <src> <dest> <session_property> <traffic_property>
              <leaky_bucket_property> [<qos_property>[<retry_property>]]
```

Traffic-Trace parameters are defined in Table 73:

TABLE 73. TRAFFIC-TRACE Parameters

Parameter	Description
<src>	Specifies the source node (node ID or IP address). This parameter is mandatory.
<dest>	Specifies the destination node. The destination address can either be a node ID, unicast or multicast address. This parameter is mandatory.
<session_property>	Specifies the session property of a connection. Session property is divided in two parts: start time and duration of session. Both of them are controlled by random distribution. This parameter is mandatory.
<traffic_property>	Specifies how traffic is to be fed into the network. This property differs for TRAFFIC-GEN and TRAFFIC-TRACE applications. Traffic property is controlled by the trace file. This parameter is mandatory.
<leaky_bucket_property>	Specifies leaky bucket property shapes traffic by using either leaky bucket (LB) or dual leaky bucket (DLB). When shaping traffic through (D)LB, application can either delay or drop application data depending on last parameter of leaky bucket property; DROP or DELAY. NOLB should be used for no traffic shaping. This parameter is mandatory.
<qos_property>	Specifies the QoS property of a session. It will be used to find QoS path of the session and to specify QoS constraints of that session.
<retry_property>	Retries the session again if it is unable to find a QoS path in its previous try.

The format of session property is as follows:

```
<session property> ::= <start time> <duration>
<start time> ::= <time distribution type>
<duration> ::= <time distribution type>
<time distribution type> ::= DET <time value> |
                             UNI <time value>
                                 <time value> |
                             EXP <time value>
<time value> ::= integer [ NS | MS | S | M | H | D ]
```

The format of trace property is as follows:

```
<traffic property> ::= TRC <trace file name>
```

The format of leaky bucket property is as follows:

```
<leaky bucket property> ::= NOLB | LB <bucket size>
                                      <token rate>
                                      (DROP|DELAY) |
                                  DLB <bucket size>
                                      <token rate>
                                      <peak rate>
                                      (DROP|DELAY)
```

Note: If QoS property is omitted, application will inject traffic without searching for a QoS path.

Format of QoS property is as follows:

```
<qos property> ::= CONSTRAINT <bandwidth> <delay>
                   [TOS <tos-value> |
                   DSCP <dscp-value> |
                   PRECEDENCE <precedence-value>]
<bandwidth> ::= integer
<delay> ::= <time value>
<tos-value> ::= <TOS value in IP header>
<dscp-value> ::= <DSCP value in IP header>
<precedence-value> ::= <Precedence value in IP header>
```

Format of retry property is as follows:

```
<retry property> ::= RETRY-INTERVAL <interval>
<interval> ::= <time value>
```

Note: At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP or TOS is not specified, a TOS value of 0 is assumed.

GUI Configuration

Traffic-Trace can be configured as a single host application (where the destination is a multicast address) or from a source to a single destination. The steps to configure Traffic-Trace as a single host application are similar to configuring the MCBR application. See the Multicast Constant Bit Rate (MCBR) section of this model library for details.

The rest of this section describes how to configure Traffic-Trace between a source and a single destination.

Configuring General Traffic-Trace Parameters

Go to **Connections > TRAFFIC-TRACE [x] -> [y]**. Configure the Traffic-Trace parameters as shown in Figure 94.

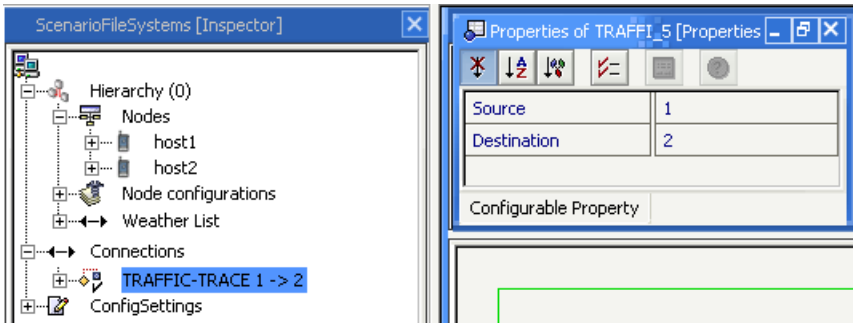


FIGURE 94. Configuring General Parameters of Traffic Trace

Configuring Start Time Parameters

Go to **Connections > TRAFFIC-TRACE [x] -> [y] > Start Time**. Configure the start time parameters as shown in Figure 95.

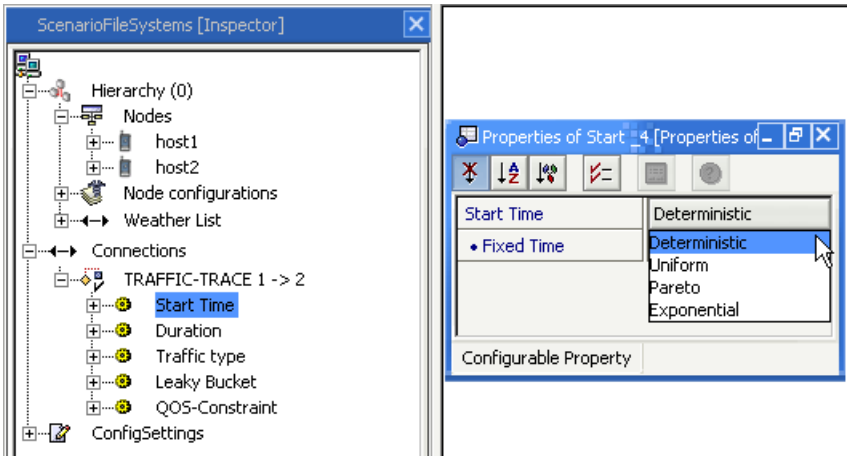


FIGURE 95. Configuring Start Time Parameters of Traffic Trace

Configuring Duration Parameters

Go to **Connections > TRAFFIC-TRACE [x] -> [y] > Duration**. Configure the duration parameters as shown in Figure 96.

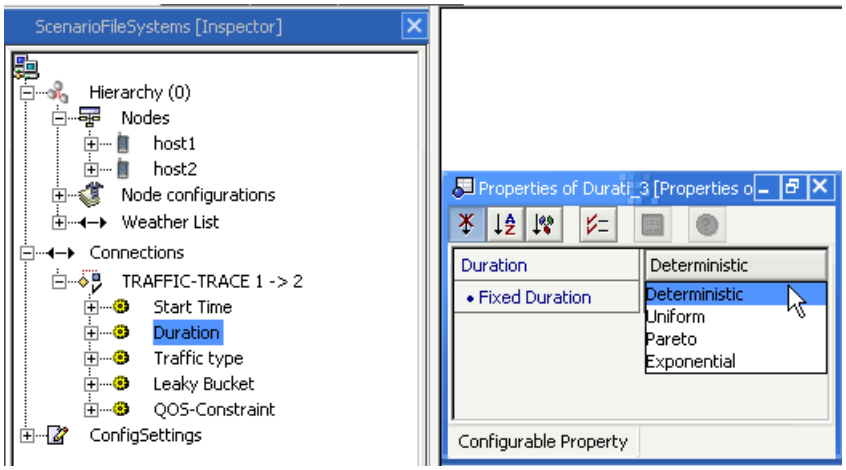


FIGURE 96. Configuring Duration Parameters of Traffic Trace

Configuring Traffic Type Parameters

Go to **Connections > TRAFFIC-TRACE [x] -> [y] > Traffic type**. Configure the traffic type parameters as shown in Figure 97.

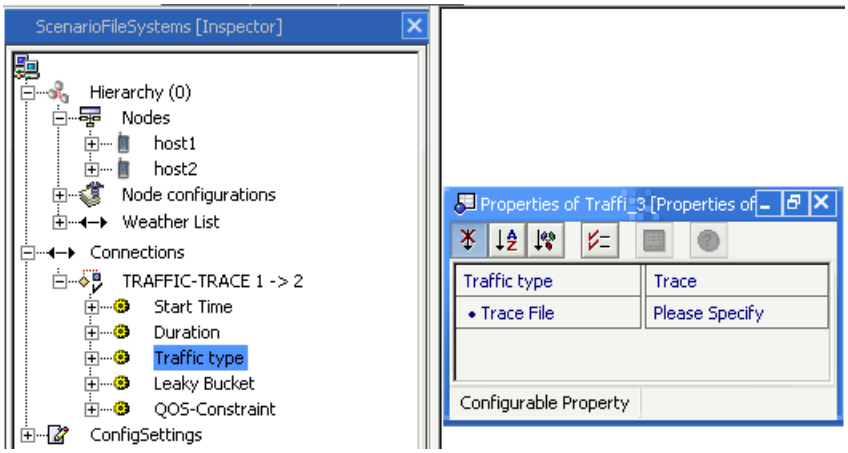


FIGURE 97. Configuring Traffic Type Parameters of Traffic Trace

Configure Leaky Bucket Parameters

Go to **Connections > TRAFFIC-TRACE [x] -> [y] > Leaky Bucket**. Configure the leaky bucket parameters as shown in Figure 98.

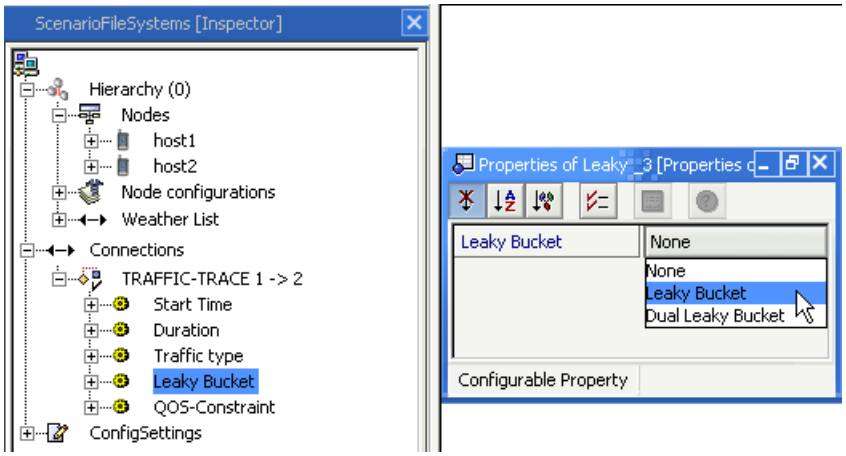


FIGURE 98. Configuring Leaky Bucket Parameters of Traffic Trace

Configure QoS-Constraint Parameters

Go to **Connections > TRAFFIC-TRACE [x] -> [y] > QoS-Constraint**. Configure the QoS constraint parameters as shown in Figure 99.

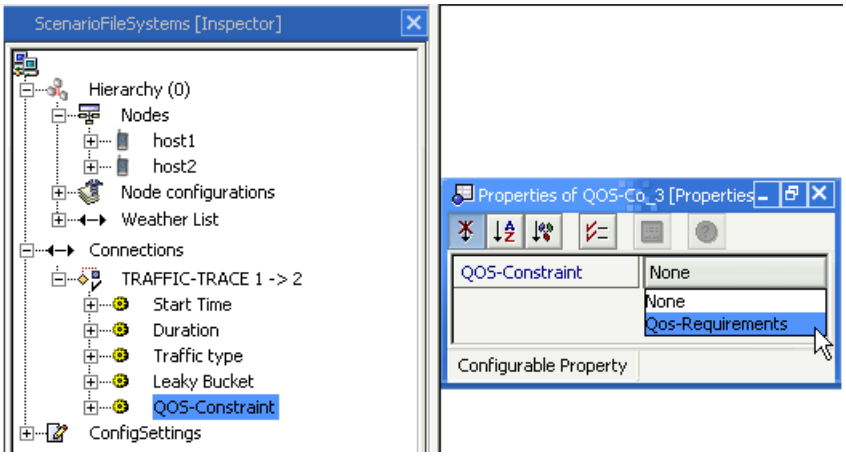


FIGURE 99. Configuring QoS-Constraint Parameters of Traffic Trace

Statistics

The TRAFFIC-TRACE model collects the statistics shown in Table 74.

TABLE 74. TRAFFIC-TRACE Statistics

Statistics	Description
TRAFFIC-TRACE Client	
Server address	Address of the server
Session start time (s)	Time in second at which session started
Session end time (s)	Time in second at which session ended
Session status	Current status of the session
Total bytes sent	Total number of bytes sent
Total data units sent	Total number of data units sent
Total data units dropped	Total number of data units dropped
Throughput (bits/s)	Client side throughput
TRAFFIC-TRACE Server	
Client address	Address of the client
Session start time	Time in second when session started
Session end time	Time in second when session ended
Session status	Current status of the session
Total bytes received	Total number of bytes received
Total data units received	Total number of data units received
Throughput (bits/s)	Server-side throughput
Average end-to-end delay	Average delay faced for packet transmission
Average jitter	Average jitter faced for packet transmission

Examples

Example 1: Video Trace File - QUALNET_HOME/scenarios/traffic-trace/soccer.trc is a trace file for a ten-minute MPG of a soccer game. To specify trace file based traffic, the keyword TRC is used, followed by the name of the trace file, soccer.trc. As in the previous example, the traffic is not shaped with a leaky bucket. Time values are expressed in seconds by default, so the start time and duration are 0S and 600S.

```
TRAFFIC-TRACE 1 2 DET 0 DET 600 TRC soccer.trc NOLB
```

The video traffic is characterized with an average rate of about 0.64 Mbps and a peak rate of a little more than 4 Mbps. Figure 100 shows the rate of the traffic and also its overall average. The average rate looks right, but the peak is much less than 4 Mbps. The reason is that the rate graph was generated with a time sliding window of size 1 second for which outgoing traffic rate is averaged. Thus, any instant peak was not clearly captured.

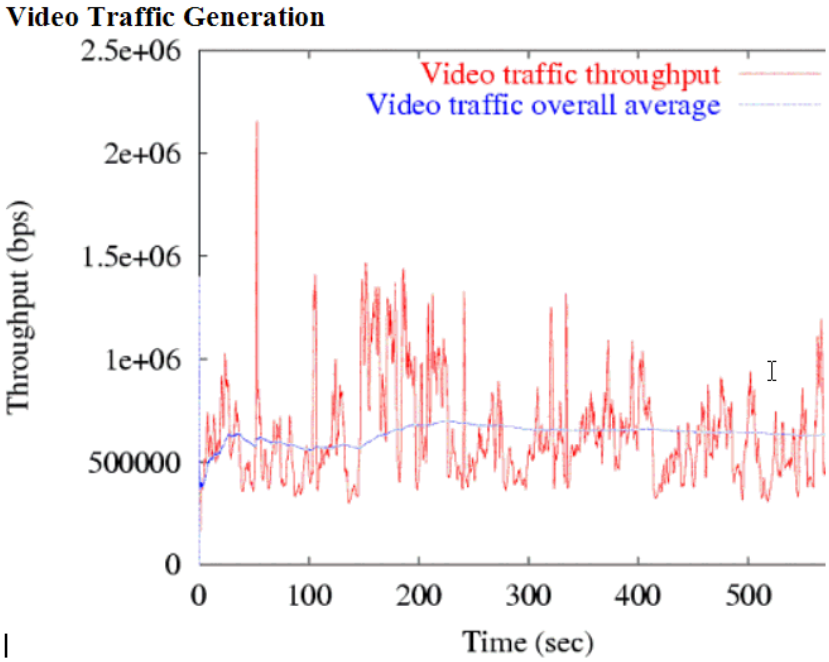


FIGURE 100. Video Traffic Generation

Example 2: Video Traffic with Leaky Bucket - This example uses a leaky bucket to regulate the traffic. Since leaky bucket regulates only burst size and average rate, we arbitrarily chose a smaller value for the average rate than the actual average rate of the video trace. This example is configured to delay packets rather than drop them.

```
TRAFFIC-TRACE 1 2 DET 0 DET 600 TRC soccer.trc
              LB 30000 500000 DELAY
```

In Figure 101, leaky bucket parameters, the bucket size in bytes and the average in bps, are specified by 30 KB and 500 Kbps, respectively. The reason for 30-KB bucket size is that since the trace file contains the biggest size data of a little less than 30 KB, the bucket size should be at least larger than that. The reason for 500-Kbps average rate is to show that the overall traffic average is suppressed by this parameter. This suppressed overall average is shown in Figure 101.

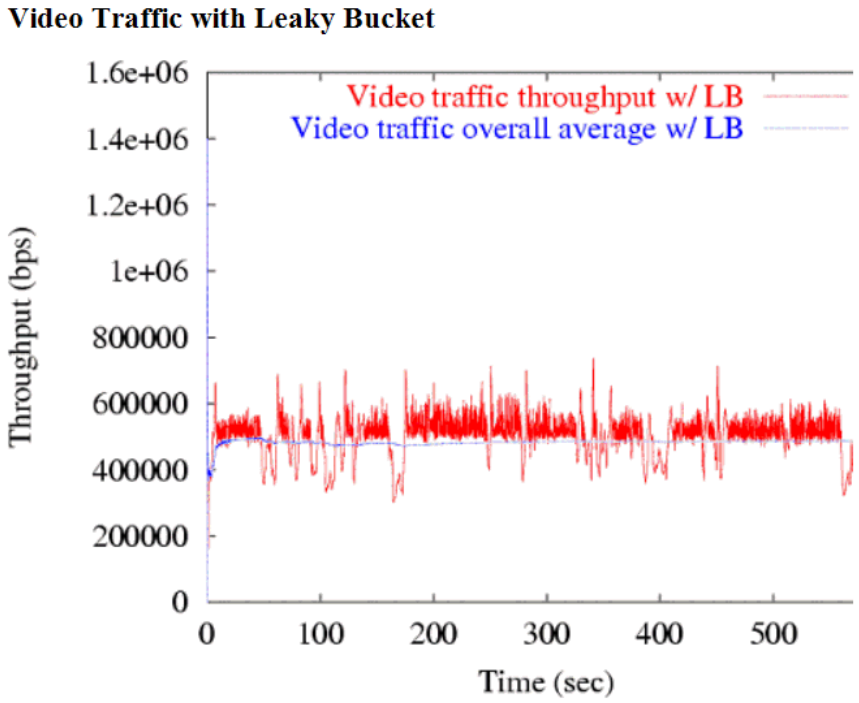


FIGURE 101. Video Traffic with Leaky Bucket

Example 3: Video Traffic with Dual Leaky Bucket

Like regular leaky bucket, dual leaky bucket controls the burst size and the overall average rate. In addition, dual leaky bucket also controls the peak rate. The parameters for the dual leaky bucket are the bucket size of 30 KB, the average rate of 800 Kbps (larger than the actual average rate of the trace), and the peak rate of 1 Mbps (much smaller than the actual peak rate of the trace). Consequently, the peak rate of the traffic is regulated by the dual leaky bucket and this is clearly shown in Figure 102.

```
TRAFFIC-TRACE 1 2 DET 0 DET 600 TRC soccer.trc
              DLB 30000 800000 1000000 DELAY
```

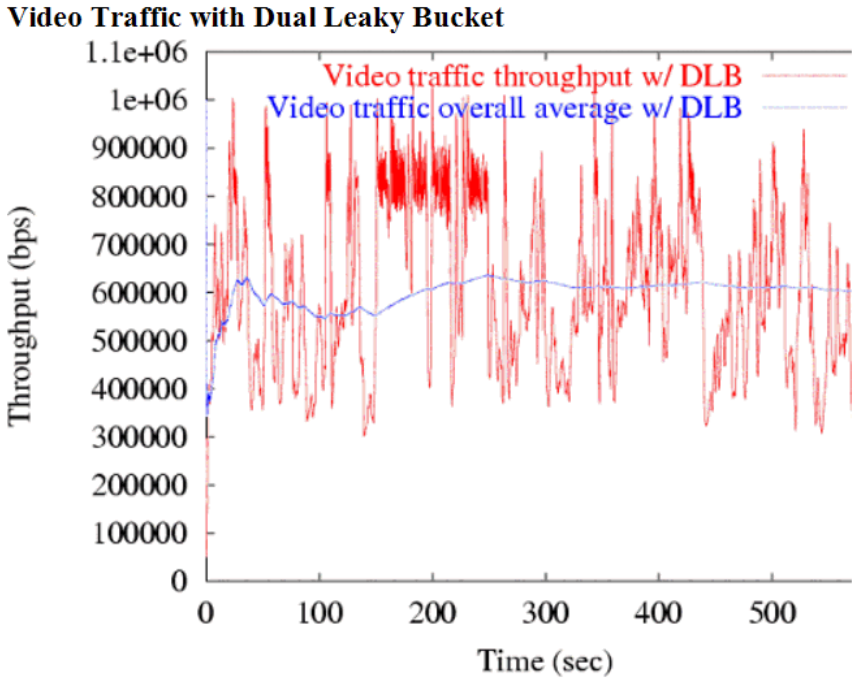


FIGURE 102. **Video Traffic with Dual Leaky Bucket**

Traffic Generator (Traffic-Gen)

Traffic-Gen simulates a random-distribution based traffic generator. Available random distribution functions are:

- Uniform (UNI)
- Exponential (EXP)
- Deterministic (DET)
- Truncated Pareto (TPD)
- Truncated Pareto with four parameters (TPD4)

These random distributions are applicable to session property and traffic property. Also, the application is capable of shaping traffic by using either leaky bucket (LB), or dual leaky bucket (DLB).

Command Line Configuration

To use Traffic-Gen, the following format is needed:

```
TRAFFIC-GEN <src> <dest> <session property>
<traffic property>
<leaky bucket property>
[<qos property> [<retry property>]]
```

Table 75 shows the Traffic-Gen parameters.

TABLE 75. Traffic-Gen Parameters

Parameter	Function
<src>	Specifies the source node (node ID or IP address). This parameter is mandatory.
<dest>	Specifies the destination node. The destination address can either be a node ID, or a unicast/multicast address. This parameter is mandatory.
<session_property>	Specifies the session property of a connection. Session property is divided in two parts: start time and duration of the session. Both of them are controlled by random distribution. This parameter is mandatory.
<traffic_property>	Specifies how traffic is to be fed into the network. This property differs for TRAFFIC-GEN and TRAFFIC-TRACE applications. Traffic property is divided into three parts: length of data segment, inter-departure time between two consecutive data segment and generation probability of a data segment. This parameter is mandatory.
<leaky_bucket_property>	Specifies the leaky bucket property shapes traffic by using either leaky bucket (LB) or dual leaky bucket (DLB). When shaping traffic through (D)LB, application can either delay or drop application data depending on last parameter of leaky bucket property; DROP or DELAY. NOLB should be used for no traffic shaping This parameter is mandatory.
<qos_property>	Used to describe the QoS property of a session. It will be used to find QoS path of the session and to specify QoS constraints of that session. Note: If QoS property is omitted, application will inject traffic without searching for a QoS path.
<retry_property>	Used to retry the session again if it is unable to find a QoS path in it's previous try.

Format of session property is as follows:

```
<session property> ::= <start time> <duration>
     <start time> ::= <time distribution type>
     <duration> ::= <time distribution type>
     <time distribution type> ::= DET <time value> |
                                  UNI <time value>
                                      <time value> |
                                  EXP <time value>
     <time value> ::= integer [ NS | MS | S | M | H | D ]
```

Format of traffic property is as follows:

```
<traffic property> ::= RND <data length> <data interval> <gen prob>
     <data length> ::= <int distribution type>
     <data interval> ::= <time distribution type>
     <gen prob> ::= float within 0 to 1.0
     <int rnd dist> ::= DET integer |
                        UNI integer integer | EXP integer
```

Format of leaky bucket property is as follows:

```
<leaky bucket property> ::= NOLB | LB <bucket size>
                                     <token rate>
                                     (DROP|DELAY) |
                                  DLB <bucket size>
                                     <token rate>
                                     <peak rate>
                                     (DROP|DELAY)
```

Format of QoS property is as follows:

```
<qos property> ::= CONSTRAINT <bandwidth> <delay>
                          [TOS <tos-value> |
                          DSCP <dscp-value> |
                          PRECEDENCE <precedence-value>]
     <bandwidth> ::= integer
     <delay> ::= <time value>
     <tos-value> ::= <TOS value in IP header>
     <dscp-value> ::= <DSCP value in IP header>
     <precedence-value> ::= <Precedence value in IP header>
```

Format of retry property is as follows:

```
<retry property> ::= RETRY-INTERVAL <interval>
<interval> ::= <time value>
```

Note: At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP or TOS is not specified, a TOS value of 0 is assumed.

Statistics

The TRAFFIC-GEN model collects the statistics shown in Table 76.

TABLE 76. Traffic-Gen Statistics

Statistics	Description
Traffic-Gen Client	
Server address	Address of the server
Session start time	Time at which session started, in seconds.
Session end time	Time at which session ended, in seconds.
Session status	Current status of the session
Total bytes sent	Total number of bytes sent.
Total data units sent	Total number of data units sent.
Total fragmented number	Total number of fragmented packets sent.
Total data units dropped	Total number of data units dropped.
Total throughput (bits/s)	Client side throughput.
Traffic-Gen Server	
Client address	Address of the client.
Session start time	Time at which the session started, in seconds.
Session end time	Time at which the session ended, in seconds.
Session status	Current status of the session.
Total bytes received	Total number of bytes received.
Total data units received	Total number of data units received.
Total throughput (bits/s)	Server side throughput
Average end-to-end delay	Average delay for packet transmission between client and server.
Average jitter	Average jitter for packet transmission between client and server.

Example 1: Voice Traffic - This example of voice traffic can be modeled with a simple two-state Markov chain. The chain consists of an ON state of exponentially distributed duration with an average of 352 ms, and an OFF state of 648 ms. In addition, the voice application uses PCM encoding which generates 160-bytes voice data every 20 ms.

```
TRAFFIC-GEN 1 2 DET 0S DET 600S RND DET 160
                DET 20MS 0.352 NOLB
```

The entry in the application file specifies traffic from node 1 to node 2, beginning at time 0, with a duration of 600 seconds. The traffic is random (as opposed to file-based) with a fixed size of 160 bytes and a fixed interval of 20 milliseconds. The problem field is a value between 0 and 1 used to determine what percent of the packets described by the given pattern will be sent. This is to model the ON and OFF states (in a somewhat less than ideal way). In this case, only about 35% of the packets will be sent. If problem is set to 1.0, the traffic is essentially identical to the CBR traffic generated by CBR 1 2 0 160 20MS 0S 600S.

Note: There is a slight difference because of how the applications schedule themselves.

The final parameter NOLB means that the traffic is not shaped by leaky bucket.

Figure 103 displays the result of this voice model. The throughput of the voice traffic and the overall average are shown. The overall average can be verified by 64 Kbps (= 160 bytes / 20 ms) * 0.352 = 23 Kbps.

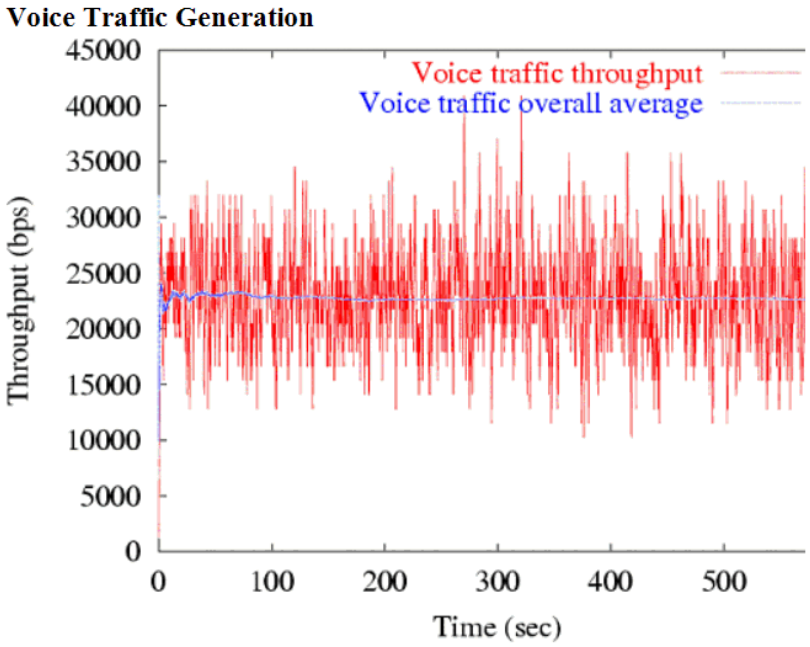


FIGURE 103. Voice Traffic Generation

Example 2: Random Traffic

Figure 104 shows an arbitrary traffic model in which the data length is exponentially distributed with an average of 160 bytes, and the inter-arrival time is also exponentially distributed with an average of 20 ms, and the data generation probability is 1. There is no traffic shaping.

```
TRAFFIC-GEN 1 2 DET 0 DET 600 RND EXP 160
EXP 20MS 1 NOLB
```

The overall average rate of this traffic can be easily computed:

```
avg = 160 bytes / 20 ms = 64 Kbps
```

This average rate is displayed in the following Random Traffic Generation graphic.

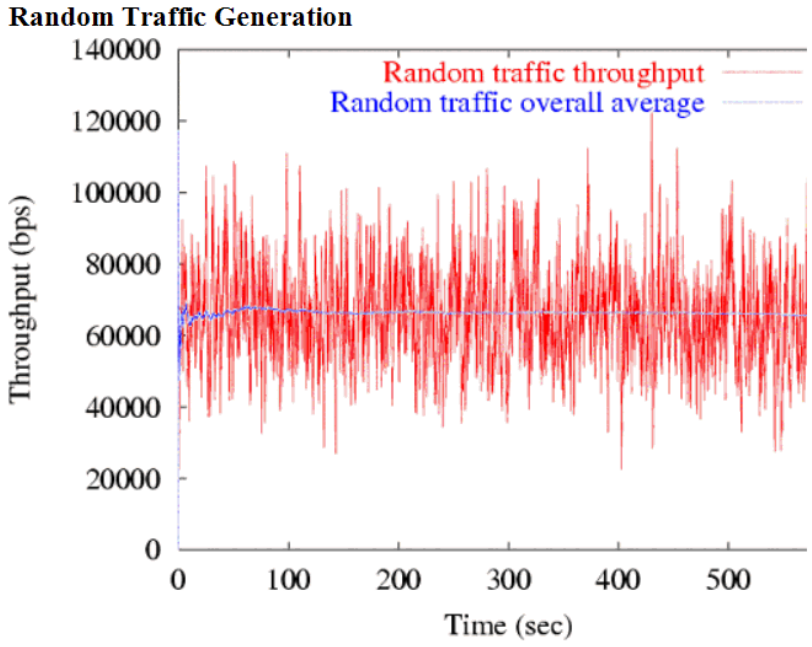


FIGURE 104. **Random Traffic Generation**

Transmission Control Protocol (TCP)

QualNet's TCP model uses code converted from the FreeBSD 2.2.2 source code implementation for TCP. The differences between the QualNet code and the FreeBSD code are improvements for simulation performance, configurability through QualNet configuration files and GUI, interface code to the simulator, and modifications to offer additional "flavors" of TCP. FreeBSD's TCP variant is known as "TCP Lite".

TCP Versions

There are several versions of TCP that you can specify via the QualNet configuration file: TCP Lite, TCP Reno, TCP SACK, and TCP Tahoe. These are modified in real systems by using compiler directives to enable sections of code, and then recompiling the kernel. Some operating systems also provide ways to dynamically modify some of the TCP parameters in the runtime. In QualNet, change them by editing the configuration file. You can either modify these parameters individually, or you can select one of the well known TCP 'flavors' that will automatically activate the appropriate parameters.

- TCP Lite is the default flavor of TCP for QualNet. TCP Lite includes the following features:
  - Slow Start
  - Congestion Avoidance
  - Fast Retransmit
  - Fast Recovery
  - Big Window & Protection Against Wrapped Sequence Numbers option (RFC 1323)
- TCP NewReno includes the following features:
  - Slow Start
  - Congestion Avoidance
  - Fast Retransmit
  - Fast Recovery with modifications
- TCP Reno includes the following features:
  - Slow Start
  - Congestion Avoidance
  - Fast Retransmit
  - Fast Recovery
- TCP Selective ACKnowledgement (TCP SACK) includes the following features:
  - Slow Start
  - Congestion Avoidance
  - Fast Retransmit
  - Fast Recovery
  - Selective Acknowledgement

- TCP Tahoe includes the following features:
  - Slow Start
  - Congestion Avoidance
  - Fast Retransmit

Command Line Configuration

To select a variant of TCP, specify the following parameter in the scenario configuration (.config) file:

```
TCP        [LITE | NEWRENO | RENO | SACK | TAHOE]
```

This parameter is optional. If it is not specified, TCP Lite is enabled by default.

Table 77 shows the TCP parameters.

TABLE 77. TCP Parameters

Parameter	Description
TCP-DELAY-ACKS [YES | NO]	Specifies whether ACKs for data segment packets are delayed for a certain period of time to reduce the number of ACKs transmitted. If this parameter is set to NO, ACKs are sent immediately. The default value is YES.
TCP-DELAY-SHORT-PACKETS-ACKS [YES | NO]	Specifies whether acknowledgement of short packets are delayed. The default value is NO.
TCP-USE-NAGLE-ALGORITHM [YES | NO]	Specifies whether Nagle algorithm is implemented. In Nagle's algorithm, a source of TCP traffic will "coalesce" small outgoing data packets into fewer and larger outgoing packets. In order to send small outgoing packets individually, disable this algorithm by changing this option to NO. The default value is YES.
TCP-USE-RFC1323 [YES | NO]	Specifies whether the RFC 1323 option is enabled. The RFC 1323 option sends window scale and timestamps in the TCP options header. When enabled, the RFC 1323 option allows the reported window size to reach a maximum 1,073,725,440 bytes. **Note**: This option cannot be selected for TCP Reno or TCP Tahoe. The default value is NO.
TCP-USE-KEEPALIVE-PROBES [YES | NO]	Specifies whether keep-alive probes are employed by TCP. To turn off these probes, change this option to NO. The default value is YES
TCP-MSS <value>	Specifies the maximum segment size (in bytes). The default value is 512.

TABLE 77. TCP Parameters (Continued)

Parameter	Description
TCP-SEND-BUFFER <value>	Specifies the send buffer size (in bytes). The send buffer size provides an upper bound on the advertised window. Note that without RFC 1323, the maximum advertised window size is 65535. Setting this parameter to a value greater than 65535 will result in wasted memory without changing the behavior. **Note**: The TCP sender cannot send more data than what is stored in the sender buffer. The default value is16384.
TCP-RECEIVE-BUFFER <value>	Specifies the receive buffer size (in bytes). The receive buffer size provide an upper bound on the advertised window. **Note**: The TCP sender cannot send more data than the available space of the receive buffer at the receiver. The default value is16384.
TCP-USE-PUSH [YES | NO]	Specifies whether the push field is set by the TCP traffic source. Setting the push field results in immediate ACK for a given packet, except for FIN segments. By setting this option to NO, the push field is not set. The default value is YES.
TCP-TRACE [NONE | TCPPEEK | TCPDUMP | TCPDUMP-ASCII]	Specifies the type of TCP dump output generated. NONE : No TCP dump output is generated. TCPPEEK : TCP dump output is generated in the TCP Peek format. TCPDUMP : TCP dump output is generated in the TCP Dump format (binary). The name of the dump file is tcptrace.dmp. TCPDUMP-ASCII: TCP dump output is generated in the TCP Dump format (ASCII). The name of the dump file is tcptrace.asc. The default value is NONE.
TCP-TRACE-WITH-DATA [YES | NO]	Specifies whether the data payload is included in TCP dump output. The default value is NO.
TCP-TRACE-DIRECTION [BOTH | INPUT | OUTPUT]	Specifies the type(s) of packets that are included int the dump file. INPUT : Only packets received from the network are recorded in the dump file. OUTPUT :Only packets sent to the network are recorded in the dump file. BOTH : Both sent and received packets are recorded in the dump file. The default value is BOTH.

TABLE 77. TCP Parameters (Continued)

Parameter	Description
TCP-RANDOM-DROP-PERCENT <value>	Specifies the initial random drop percent for the node. The default value is 0.0
TCP-VERIFICATION-DROP-COUNT <value>	Specifies the drop count. This option is for use with TCP verification scenarios that test variant behavior with 0 to 4 drops. The range is 0 to 4. The default value is 0. **Note**: This parameter is for testing purposes. It is recommended that the default value be used for this parameter.
ECN [YES | NO]	Enables Explicit Congestion Notification (ECN). ECN marks the IP header, instead of dropping packets, when network queues report congestion. ECN requires IP-QUEUE-TYPE to be Random Early Detection (RED), RED for Input Output (RIO), or Weighted Red (WRED). Furthermore, the source and destination nodes must be ECN enabled; intermediate routes do not have to be ECN enabled. The default value is NO.
TRACE-TCP [YES | NO]	Enables the packet tracing for TCP. The default value is NO.

Format of the TCP Dump (ASCII) File

if the parameter TCP-TRACE is set to TCPDUMP-ASCII, the a dump file (with the name tcptrace.dmp) is produced int the TCP Dump (ASCII) format. The format of this file is:

```
<time in "hour:minute:second">
<"IP address in dot format".port number of sender>
<"IP address in dot format".port number of receiver>
<"S" for SYN bit on; "F" for FIN bit on; "P" for PUSH bit on; "R" for RST bit on;
"." is printed if none of these bits are on>
<"initial sequence number of the packet":"initial sequence number of the packet +
packet length"(packet length)>
<"ack" for acknowledgement bit on with acknowledged sequence number>
<"win" window with value>
<remainder of this line represents the option fields with value separated by comma>:
       "nop" for no option
       "mss" for maximum segment size with value
       "wscale" for window scale with value
       "timestamp" with value
       "sackOK"
```

Note: Refer to the TCP Trace website at http://www.tcptrace.com for additional examples of the format.

GUI Configuration

Configuring TCP

Go to **Hierarchy (x) > Nodes > host (x) > Node configurations > Transport Layer**. From the configurable property window, configure the parameters as shown in Figure 105:

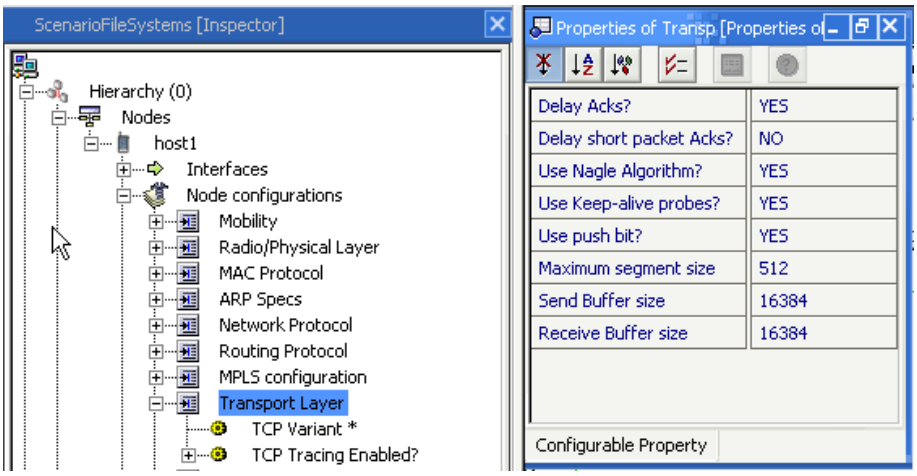


FIGURE 105. Configure TCP

Configuring TCP Variants

You can select TCP Variants as shown in Figure 106:

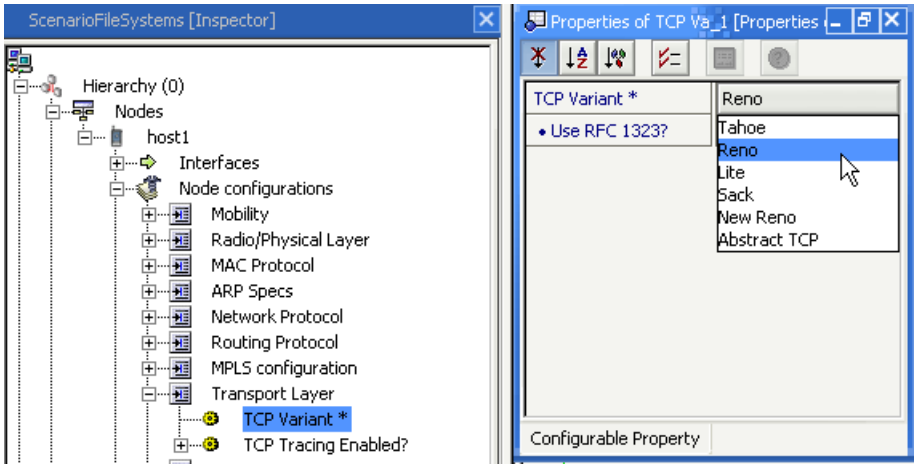


FIGURE 106. Configuring TCP Variants

Enabling TCP Tracing

To enable TCP tracing set **TCP Tracing Enabled?** to **Yes** as shown in Figure 107:

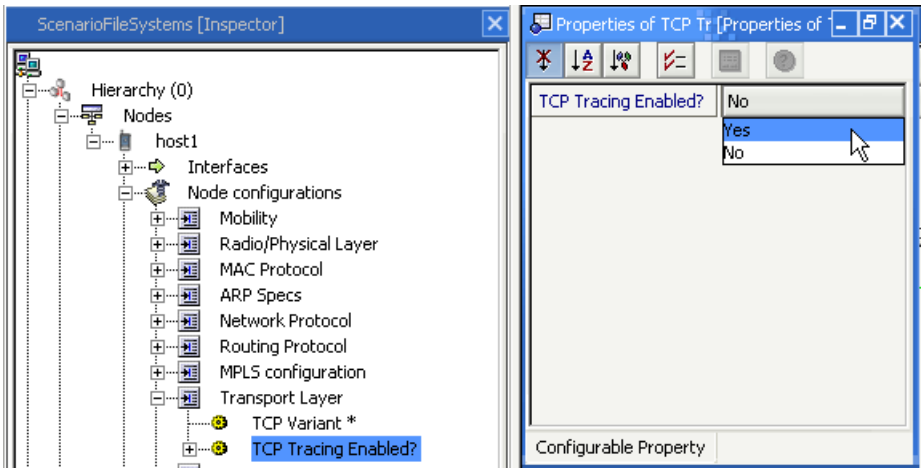


FIGURE 107. Enabling TCP tracing

Configuring TCP Tracing Properties

After TCP Tracing is enabled, configure the dependent parameters as shown in Figure 108:

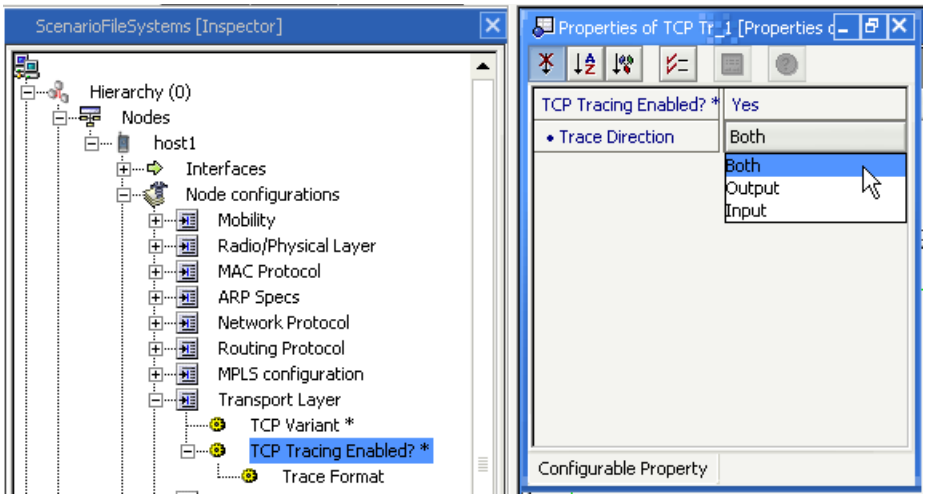


FIGURE 108. TCP Tracing Properties

Configuring TCP Trace Format

You can configure TCP Trace format as shown in Figure 109:

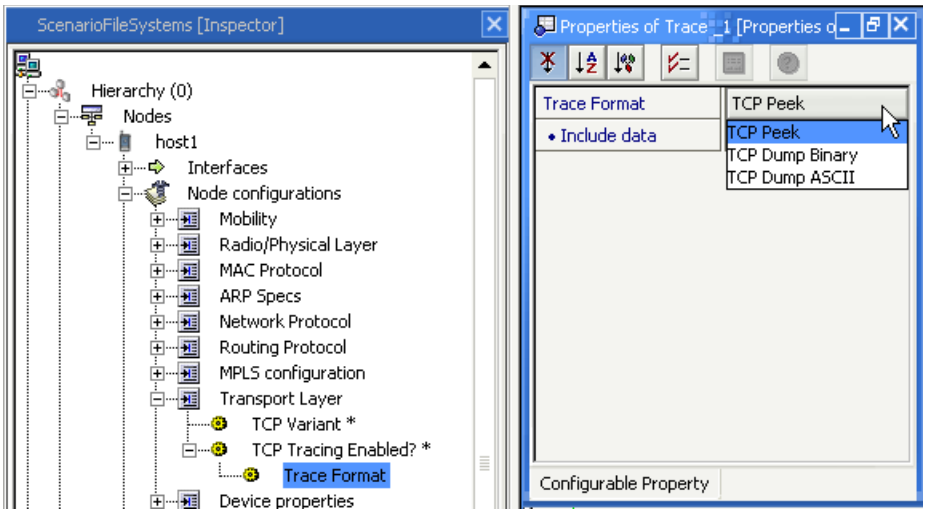


FIGURE 109. Configure Trace Direction

Enabling TCP Statistics

Go to **ConfigSettings > Statistics > Statistics**. Enable **TCP** as shown in Figure 110.

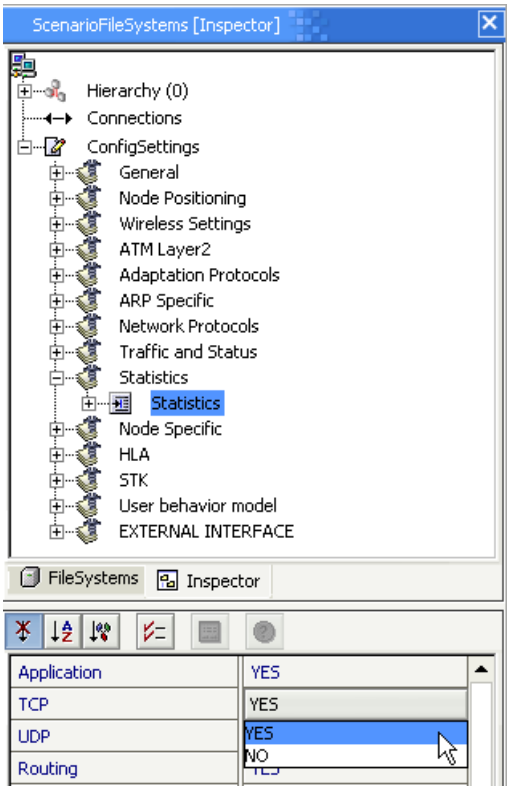


FIGURE 110. Enabling TCP Statistics

Statistics

Table 78 shows the statistics that are collected by TCP.

TABLE 78. TCP Statistics

Statistic	Description
Packets Sent to Network Layer	Total number of packets sent to the network layer.
Data Packets Sent	Total number of packets sent including retransmitted packets.
Data Packets in Sequence	Total number of packets sent, excluding retransmissions and probes.
Data Packets Retransmitted	Total number of retransmitted packets.
Data Packets Fast Retransmitted	Total number of fast retransmitted packets.
ACK-only Packets Sent	Total number of acknowledgement-only packets are sent. Piggyback acknowledgements are not sent.
Window Probes Sent	Total number of window probes sent.
Window Update-Only Packets Sent	Total number of window update packets sent.
Pure Control (SYN|FIN|RST) Packets Sent	Total number of pure control (SYN | FIN | RST) packets sent.

TABLE 78. TCP Statistics (Continued)

Statistic	Description
Total Packets Received From Network Layer	Total number of packets received from network layer.
Data Packets Received	Total number of data packets received.
Pure Control (SYN|FIN|RST) Packets Received	Total number of control packets received.
Duplicate ACK Packets Received	Total number of duplicate packets received.
In Sequence ACK Packets Received	Total number of In sequence acknowledgement packets received.
Window Probes Received	Total number of window probe packets received.
Window Update-Only Packets Received	Total number of window update packets received.
Total Packets with Errors	Total number of packets received with errors like checksum, offset, or short errors.
Packets Received with Checksum Errors	Total number of packets received with checksum errors.
Packets Received with Bad Offset	Total number of packets received with bad offset.
Packets Received that are Too Short	Total number of packets received for being too short.

TCP Dump

QualNet can produce TCPDUMP format binary- or ASCII-based packet trace files that represent the behavior of the TCP protocol during a simulation. To activate TCPDUMP binary trace file generation, place the following entry in the *.config file:

```
TCP-TRACE              TCPDUMP
```

To activate TCPDUMP ASCII trace file generation, place the following entry in the *.config file instead:

```
TCP-TRACE              TCPDUMP-ASCII
```

By default, when TCPDUMP functionality is enabled, the trace is produced for BOTH output and input packets. This value can be changed by placing one of the following three entries in the *.config file:

```
TCP-TRACE-DIRECTION    BOTH
TCP-TRACE-DIRECTION    OUTPUT
TCP-TRACE-DIRECTION    INPUT
```

Uniform Node Placement Model

In the uniform node placement model, the terrain is divided into a number of equal-sized cells, each proportional to the size and shape of the terrain. The number of cells is calculated by rounding the number of nodes to the next higher square of an integer. For example, if there are 23 nodes in a scenario, the terrain is divided into 25 cells. Beginning with the lower left cell, proceeding first along the X-axis, then along the Y-axis, one node is placed at a random position in each cell, until all nodes are placed.

Command Line Configuration

To select this model, set the parameter `NODE-PLACEMENT` in the configuration file as follows:

```
NODE-PLACEMENT UNIFORM
```

User Datagram Protocol (UDP)

UDP is a simple, low-overhead, packet-switched transport protocol that assumes that the Internet Protocol (IP) is the underlying network protocol. Applications that require UDP include CBR or routing protocols such as RIP.

Command Line Configuration

The configurable parameters for UDP are shown in Table 79.

TABLE 79. UDP Parameters

Parameter	Description
UDP-STATISTICS [YES | NO]	Enables or disables the UDP statistics. By default UDP statistics are disabled.
TRACE-UDP [YES | NO]	Enables or disables the Tracing of UDP packets. By default tracing of UDP packets is disabled.

To enable collection of the UDP statistics use the following in *.config file of the scenario.

```
UDP-STATISTICS YES
```

To enable trace of UDP Data packets, use the following in *.config file of the scenario.

```
TRACE-UDP YES
```

GUI Configuration

Enabling Statistic Collection

Go to **ConfigSettings > Statistics > Statistics** and enable **UDP** statistics as shown in Figure 111.

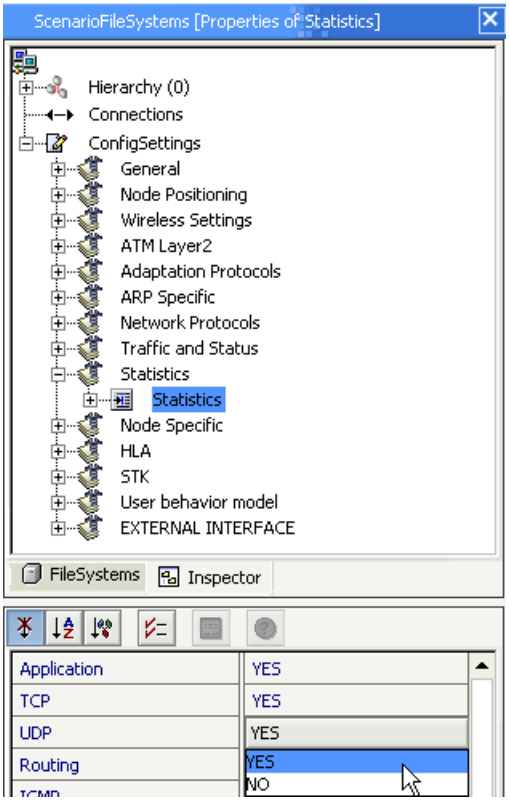


FIGURE 111. Enable Statistic Collection

Enabling Trace

Please refer to the section on Tracing Packet Headers in the *QualNet 4.5 User's Guide*.

Statistics

UDP collects the statistics shown in Table 80:

TABLE 80. UDP Statistics

Statistic	Description
Number of packets received	Total Number of packets received from applications.
Number of packets sent	Total Number of packets sent to application layer.

Variable Bit Rate (VBR) Traffic Generator

VBR is generally used to fill in background traffic in order to affect the performance of other applications being analyzed, or to simulate the performance of generic multimedia traffic.

Command Line Configuration

To specify VBR traffic in the application configuration file (default.app, for example), specify the parameters according to the following format:

```
VBR <src> <dest> <item size> <mean interval>
    <start time> <end time>
    [TOS <tos value> | DSCP <dscp value> | PRECEDENCE <precedence value>]
```

Table 81 shows the VBR parameters. These parameters are mandatory:

TABLE 81. VBR Parameters

Parameter	Description
<src>	Specifies traffic source (client) node's nodeId or IP Address.
<dest>	Specifies traffic destination ID Indicates the server node's nodeId or IP Address.
<item_size>	Specifies the size of each data unit to be sent in bytes. This is an integer value.
<Mean interval>	Calculates an exponential random number stream that determines delay between packet transmissions. This is a time value..
<start_time>	Indicates when the transmissions should begin. This is a time value.
<end_time>	Indicates when the transmissions should end. This is a time value.
TOS <tos value>	Specifies the contents of the 8-bit TOS value of the IP header for the packets generated. The range is from 0 to 255. This parameter is optional.
DSCP <dscp value>	Specifies the contents of the 6-bit DSCP value of the IP header for the packets generated. The range is from 0 to 63. This parameter is optional.
PRECEDENCE <precedence value>	Specifies the contents of the 3-bit Precedence value of the IP header for the packets generated. The range is from 0 to 7. This parameter is optional.

Note: At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP or TOS is not specified, a TOS value of 0 is assumed.

GUI Configuration

Configure VBR Parameters

Go to **Connections > VBR[x] -> [y]**. Configure the VBR parameters as shown in Figure 112.

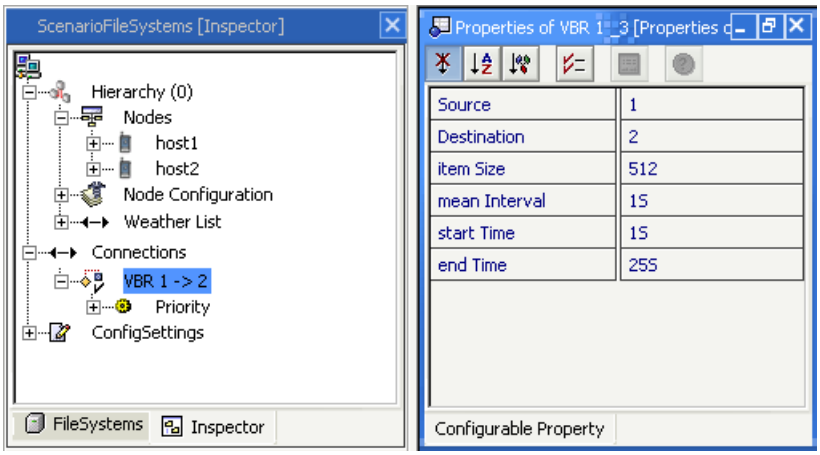


FIGURE 112. Configuring VBR Parameters

Statistics

The dynamic statistics collected for VBR are shown in Table 82. VBR keeps track of the total bytes sent and the total bytes received.

TABLE 82. VBR Statistics

Statistics	Description
Total Bytes Sent	Total number of bytes sent.
Total Bytes Received	Total number of bytes received.

Example

```
VBR 1 2 2048 1M 50S 500M
```

This example specifies that node 1 will send two-kilobyte packets at variable intervals (with the average interval time being one minute), spanning from a simulation time of 50 seconds to 500 minutes.

Note: Traffic (items with a specified size in bytes) is sent at a frequency that varies depending on an exponential distribution from the client to server node. The mean interval determines the variation. The amount of traffic received is counted by the server and reported as a statistic.

Weighted Fair Queuing (WFQ) Scheduler

WFQ Scheduler is an alternative scheduling discipline in QualNet. WFQ attempts to service each priority queue based on a percentage allocation of the total outgoing bandwidth of the link. This allocation is based on a configurable weight assigned to each queue.

Command Line Configuration

Selection of a scheduler is currently mandatory, and is achieved by placing the following entry in the *.config:

```
IP-QUEUE-SCHEDULER WEIGHTED-FAIR
```

Note: If no parameters are given, the weight of each queue is the ((priority value of the queue)+ 1) / (sum of all the priority_values for all the queues). Otherwise, users must specify the weight for all of the queues on that interface, such that the sum of the weights equals one, using the following parameter, where <weight> is between 0 and 1.

```
QUEUE-WEIGHT[priority] <weight>
```

Where priority is the integer value of the priority assigned to each queue (numbered from 0...num_of_priority_queues-1), and the weight is a floating point number between zero and one. This parameter can also be specified using node-specific or network-specific parameters.

Statistics

Table 83 shows the statistics collected by WFQ Scheduler.

TABLE 83. WFQ Scheduler

Statistics	Description
Packets Queued	Total number of packets queued in the queue
Packets Dequeued	Total number of packets dequeued
Packets Dropped	Total number of packets dropped
Dequeue Requests While Queue Active	Total number of de -queue requests while the queue is active
Dequeue Requests Serviced	Total number of de queue requests services
Service Ratio received	Total packets de queued to total de queue requests
Total Service Received(Bytes)	Total service received in bytes

Weighted RED (WRED) Queue

WRED is a multilevel RED (MRED) queuing algorithm suitable for the AF PHB that can identify any preferentially and can punish misbehaving users. The incoming packets are marked through packet marking algorithm on the basis of Service Level Agreement (SLA) between the customer and service-provider. The making is done in two-color (DP0 and DP1) or three-color (DP0, DP1, DP2) level referring drop precedence (DP) as below:

- Green: Level Low Drop, DP0
- Yellow: Level Medium Drop, DP1
- Red: Level High Drop, DP2

WRED is a Single Average Multiple threshold (SAMT) variant of multilevel RED and can operate in three-color mode. It uses three RED algorithms - destined for DP0, DP1 and DP2 packets. When congestion occurs it first drops DP2 packets, and if the congestion persists, it drops DP1 packets and finally it drops DP0 packets as a last resort. But how and when it starts dropping DP1 or DP0 packets depends on the RED parameter specification for each color. Thus, RED parameter settings in configuration file should reflect this inherent assumption that it drops DP2 packets more aggressively than dropping the DP1 packets. Again it drops DP1 packets more aggressively than dropping the DP0 packets.

Command Line Configuration

To select WRED queuing discipline for the AF PHB, place the following entry in *.config:

```
IP-QUEUE-TYPE WRED
```

Table 84 shows the WRED parameters:

TABLE 84. WRED Parameters

Parameter	Description
GREEN-PROFILE-MIN-THRESHOLD [Number greater than zero]	Specifies the number of green profile packets in the queue that represents the lower bound at which green packets can be randomly dropped. Default value is 10.
GREEN-PROFILE-MAX-THRESHOLD [Number greater than zero]	Specifies the number of yellow packets in the queue that represents the upper bound at which green packets can be randomly dropped. Default value is 20.
GREEN-PROFILE-MAX-PROBABILITY [Number greater than zero and less than 1.0]	Specifies the maximum probability (0...1) value at which a packet can be dropped (before the queue is completely full). Default value is 0.02.
YELLOW-PROFILE-MIN-THRESHOLD [Number greater than zero]	Specifies the number of yellow profile packets in the queue that represents the lower bound at which yellow packets can be randomly dropped. Default value is 5.
YELLOW-PROFILE-MAX-THRESHOLD [Number greater than zero]	Specifies the number of yellow profile packets in the queue that represents the upper bound at which yellow packets can be randomly dropped. Default value is 10.
YELLOW-PROFILE-MAX-PROBABILITY [Number greater than zero and less than 1.0]	Specifies the maximum probability (0...1) value at which a packet can be dropped (before the queue is completely full). Default value is 0.02.

TABLE 84. **WRED Parameters (Continued)**

Parameter	Description
RED-PROFILE-MIN-THRESHOLD [Number greater than zero]	Specifies the number of yellow profile packets in the queue that represents the lower bound at which red packets can be randomly dropped. Default value is 2.
RED-PROFILE-MAX-THRESHOLD [Number greater than zero]	Specifies the number of yellow profile packets in the queue that represents the upper bound at which red packets can be randomly dropped. Default value is 5.
RED-PROFILE-MAX-PROBABILITY [Number greater than zero and less than 1.0]	Specifies the maximum probability (0...1) value at which a packet can be dropped (before the queue is completely full). Default value is 0.02.
RED-QUEUE-WEIGHT	Specifies the queue weight used to determine the bias towards recent or historical queue lengths in calculating the average. Default value is 0.002.
RED-SMALL-PACKET-TRANSMISSION-TIME	Specifies the sample amount of time to transmit a small packet. Used to estimate the queue average during idle periods. Default value is 10MS.

GUI Configuration

Configuring IP Queue Type WRED

Go to **Hierarchy (x) > Node configurations > Network Protocol > Specify Priority-wise IP Queue Type? > IP Queue Type**. From the configurable property window, set **IP Queue Type** to **WRED** as shown in Figure 113.

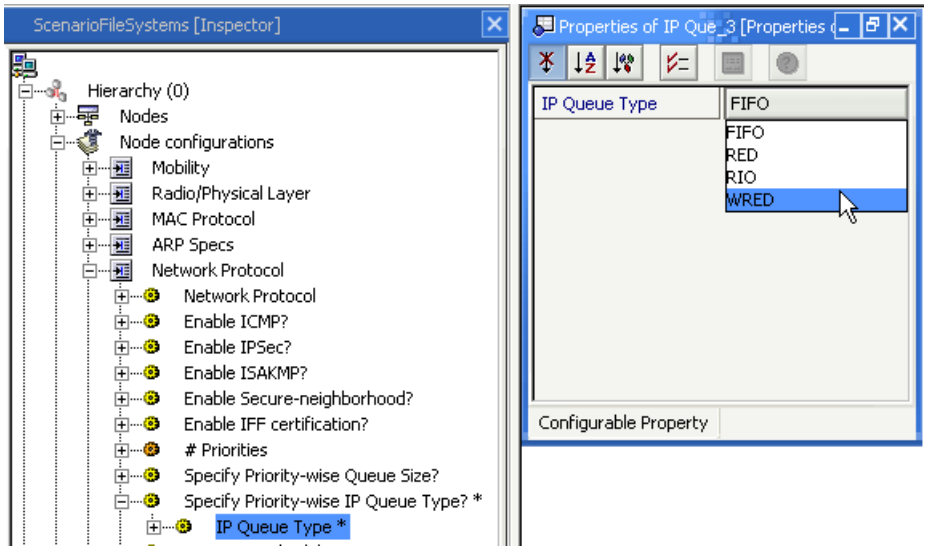


FIGURE 113. **Configuring IP Queue Type WRED**

Configuring General Parameters of WRED

When WRED is enabled, a set of its parameters is displayed. Configure the WRED parameters as shown in Figure 114.

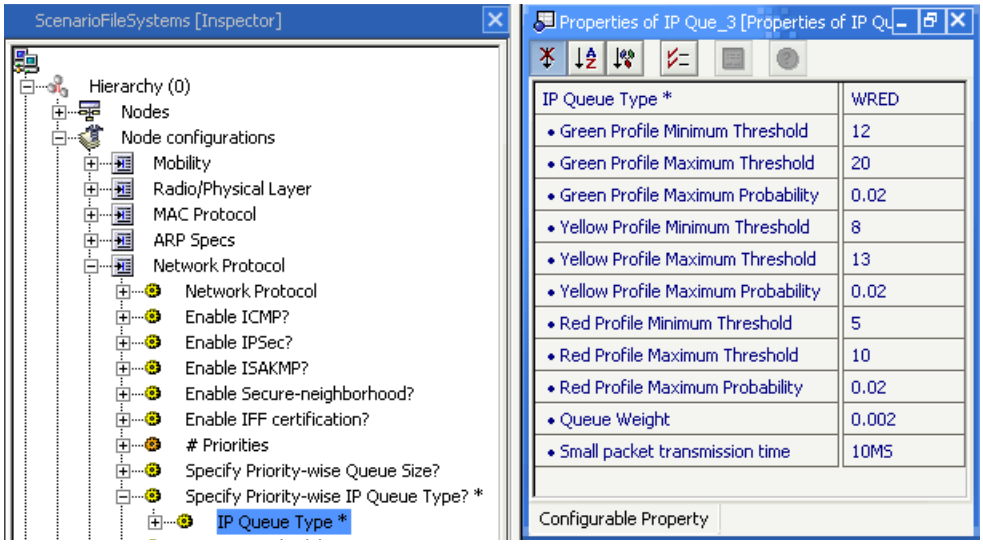


FIGURE 114. Configuring General Parameters of WRED

Statistics

WRED collects the statistics shown in Table 85, per priority queue, per node.

Note: In Table 85, <Profile> = GREEN | YELLOW | RED packet profile.

TABLE 85. WRED Statistics

Statistics	Description
(<Profile>) Packets Queued	Total number of <profile> packets queued in this queue
(<Profile>) Packets Dequeued	Total number of <profile> packets dequeued from this queue
(<Profile>) Packets Marked ECN	Total number of <profile> packets marked by the Explicit Congestion Notification (ECN) capable network
(<Profile>) Packets Dropped	Total number of <profile> packets dropped by this queue
Total Packets Queued	Total number of packets queued in this queue
Total Packets Dequeued	Total number of packets dequeued from this queue
Total Packets Dropped	Total number of packets dropped by this queue
Average Queue Length (bytes)	Average of queue length reached by this queue during simulation
Average Time In Queue	Average delay suffered by the packets in this queue
Longest Time in Queue	Longest delay suffered by a packet in this queue
Peak Queue Size (bytes)	Peak Queue Size reached by this queue during simulation

TABLE 85. **WRED Statistics**

Statistics	Description
Packets Marked ECN	Total number of packets marked by ECN capable network
Total Packets Dropped Forcefully	Total Packets Dropped Forcefully

Weighted Round Robin (WRR) Scheduler

The WRR Scheduler is an alternative scheduling discipline n QualNet. Weighted Round Robin also attempts to service each priority queue based on a percentage allocation of the total outgoing bandwidth of the link. However, it does not calculate a “Finish Time” (the time at which a packet would have been serviced given a hypothetical fluid server) for each packet, instead, it establishes a round-robin order with multiple turns for highly weighted priority queues.

Command Line Configuration

Selection of a scheduler is currently mandatory, and is achieved by placing the following entry in the *.config:

```
IP-QUEUE-SCHEDULER              WEIGHTED-ROUND-ROBIN
```

It can be specified using node-specific or network-specific parameterization.

Note: If parameters are not used, the weight of each queue is the ((priority value of the queue)+ 1) / (sum of all the priority_values for all the queues). Otherwise, users must specify the weight for all of the queues on that interface, such that the sum of the weights equals one, using the following parameter:

```
QUEUE-WEIGHT[priority]        <weight>
```

where:

priority is the integer value of the priority assigned to each queue (numbered from 0…num_of_priority_queues-1).

The weight is a floating point number between zero and one.

This parameter can also be specified using node-specific or network-specific parameterization.

Statistics

WRR generates the statistics shown in Table 86:

TABLE 86. WRR Statistics

Statistics	Description
Packets Queued	Total number of packets queued in the queue
Packets Dequeued	Total number of packets dequeued from the queue
Packets Dropped	Total number of packets dropped from the queue
Service Ratio	Total number of packets dequeued to total number of dequeue requests

Wired and Wireless Point-to-Point Links

The LINK keyword is used to describe a point-to-point link connecting exactly two nodes. It also creates the network interfaces for those nodes to connect them to the network.

The LINK keyword takes two arguments:

- The QualNet Subnet Shorthand specification of the network address and subnet mask.
- The nodeIds of the two nodes that are connected by the point-to-point link separated by commas, and surrounded by curly braces {}.

Command Line Configuration

The LINK-PHY-TYPE parameter determines whether the link is a wired link, a wireless link, or a microwave link. It has the following format:

```
LINK-PHY-TYPE   [WIRED | WIRELESS | MICROWAVE]
```

See *QualNet 4.5 Wireless Model Library* for a description of the microwave link. The parameters for the wired and wireless point-to-point link models are described below.

Table 87 shows the configuration parameters that define the properties of the link:

TABLE 87. Wired and Wireless Point to Point Links Parameters

Parameter	Description
LINK-BANDWIDTH <bandwidth>	LINK-BANDWIDTH defines the bandwidth (in bits per second) of the link. The link is assumed to be full duplex (data can be sent in both directions at the same time). This parameter is mandatory.
LINK-PROPAGATION-DELAY <delay>	LINK-PROPAGATION-DELAY specifies the time it takes a signal to travel from one node to the other node. This parameter is specified only if the link is a wired link. The delay is specified in QualNet time format. This parameter is mandatory.
LINK-HEADER-SIZE-IN-BITS <header-size>	LINK-HEADER-SIZE-IN-BITS specifies the header size (in bits). This parameter is optional and the default value of 224 bits is used if it is not specified.
MAC-LAYER-STATISTICS [YES | NO]	Specifies whether MAC Layer Statistics are enabled or disabled.

Note: If any of the parameters LINK-BANDWIDTH, LINK-PROPAGATION-DELAY, and LINK-HEADER-SIZE-IN-BITS is not qualified, then it applies to both directions. For asymmetric links, qualify the parameter with the interface address of the source node of the traffic.

Example

```
LINK N2-1.0 { 1, 2 }
LINK N2-2.0 { 2, 3 }
[N2-1.0] LINK-BANDWIDTH 1544000
[N2-1.0] LINK-PROPAGATION-DELAY 1MS
[N2-2.0] LINK-BANDWIDTH 112000
[N2-2.0] LINK-PROPAGATION-DELAY 50MS
LINK-HEADER-SIZE-IN-BITS 200
```

The first LINK statement creates a point-to-point link between nodes 1 and 2. The second LINK statement creates a point-to-point link between nodes 2 and 3. NodeId 2 is connected to both nodeId 1 and nodeId 3 by separate links, and can switch packets between them.

The first link has T1 speed, and the second link has ISDN speed. In both cases, the link is full duplex, and symmetric in propagation delay and bandwidth.

GUI Configuration

Configuring a Wired Point-to-Point Link

Go to **Connections > Point to point link [x] -> [y] [Address] > Point to point link Properties > Mac Protocol > Link Type**. Set **Link Type** as **Wired** as shown in Figure 115.

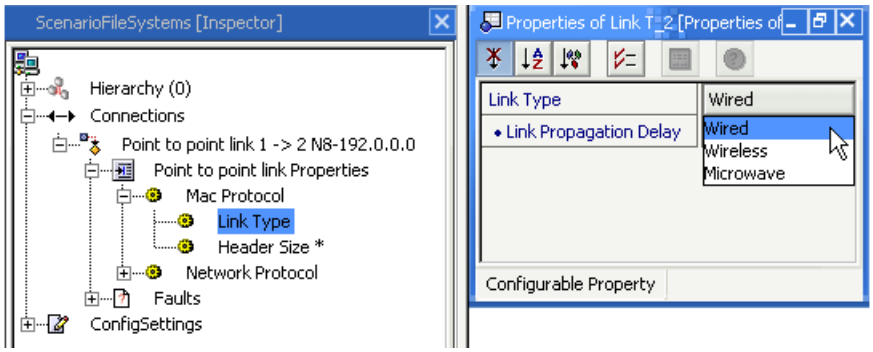


FIGURE 115. Configuring a Wired Point-to-Point Link

Configuring a Wireless Point -to-Point Link

Go to **Connections > Point to point link [x] -> [y] [Address] > Point to point link Properties > Mac Protocol > Link Type**. Set **Link Type** as **Wireless** as shown in Figure 116.

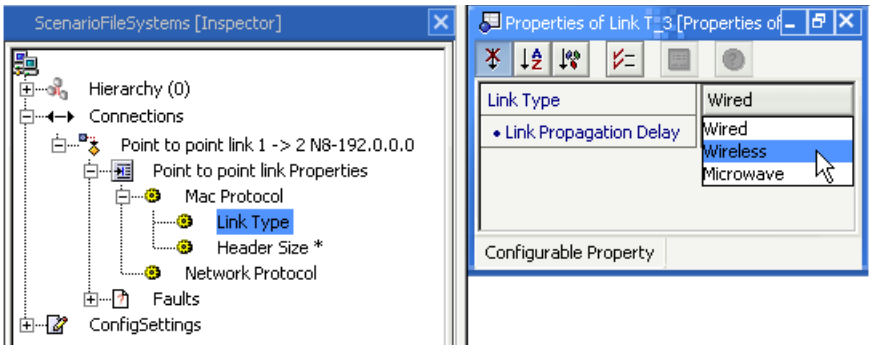


FIGURE 116. Configuring Wireless Point -to-Point Link

Configuring Header Size for Point -to-Point Link

Go to **Connections > Point to point link [x] -> [y] [Address > Point to point link Properties > Mac Protocol > Header Size**. Configure **Header Size** as shown in Figure 117.

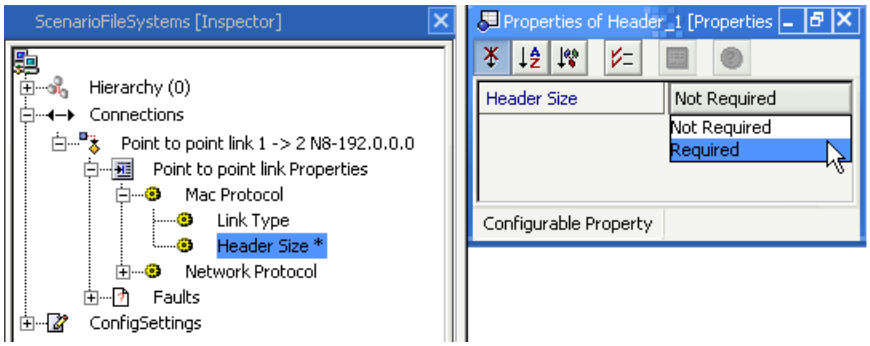


FIGURE 117. Configuring Parameter for Header Size

Configuring Network Protocol GSM Layer3 for Point -to-Point Link

Go to **Connections > Point to point link [x] -> [y] [Address] > Point to point link Properties > Mac Protocol > Network Protocol**. Configure **Network Protocol** as shown inFigure 118.

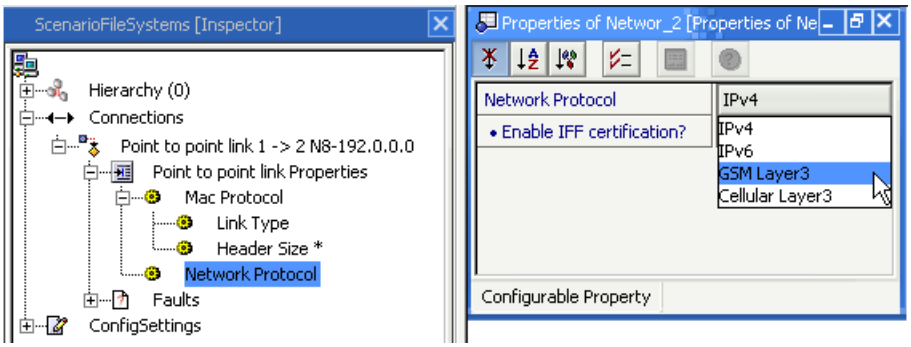


FIGURE 118. Configuring Network Protocol GSM Layer3

Statistics

Wired and Wireless Point-to-Point Links statistics are shown in Table 88.

TABLE 88. Wired and Wireless Point to Point Links Statistics

Statistics	Description
Frames sent	Total number of Frames Transmitted
Frames Received	Total number of Frames Received
Destination	Specifies the Destination
Link Utilization	Specifies the Link Utilization