

Configuring SNMP

This document describes the Livingston implementation of the Simple Network Management Protocol (SNMP) and includes the following topics:

- “Introduction” on page -1
- “The livingston.mib Definition File” on page -2
- “Setting SNMP Monitoring” on page -8
- “Setting SNMP Read and Write Community Strings” on page -9
- “Adding SNMP Read and Write Hosts” on page -9
- “Viewing SNMP Settings” on page -10

Introduction

SNMP is an application-layer protocol that allows devices to communicate management information. You can configure the PortMaster to provide network and device information via SNMP to a network management system (NMS). You must have NMS software to use SNMP.

SNMP has the following parts:

- An SNMP agent (provided in ComOS)
- An SNMP manager (not provided)
- A Management Information Base (MIB)

SNMP specifies the message format for exchanging information between the SNMP manager and an SNMP agent.

The SNMP agent returns values for MIB variables that can be changed or queried by the SNMP manager. The agent gathers information from the MIB, which resides on the target device. MIB information can include device parameters and network status. The agent is capable of responding to requests to get or set data from the manager.

Livingston products support both MIB I and MIB II variables as specified in RFC 1213, along with an enterprise-specific MIB. SNMP management can be enabled for any PortMaster. Livingston ships configuration files compatible with various network management packages along with the PMconsole software.

The livingston.mib Definition File

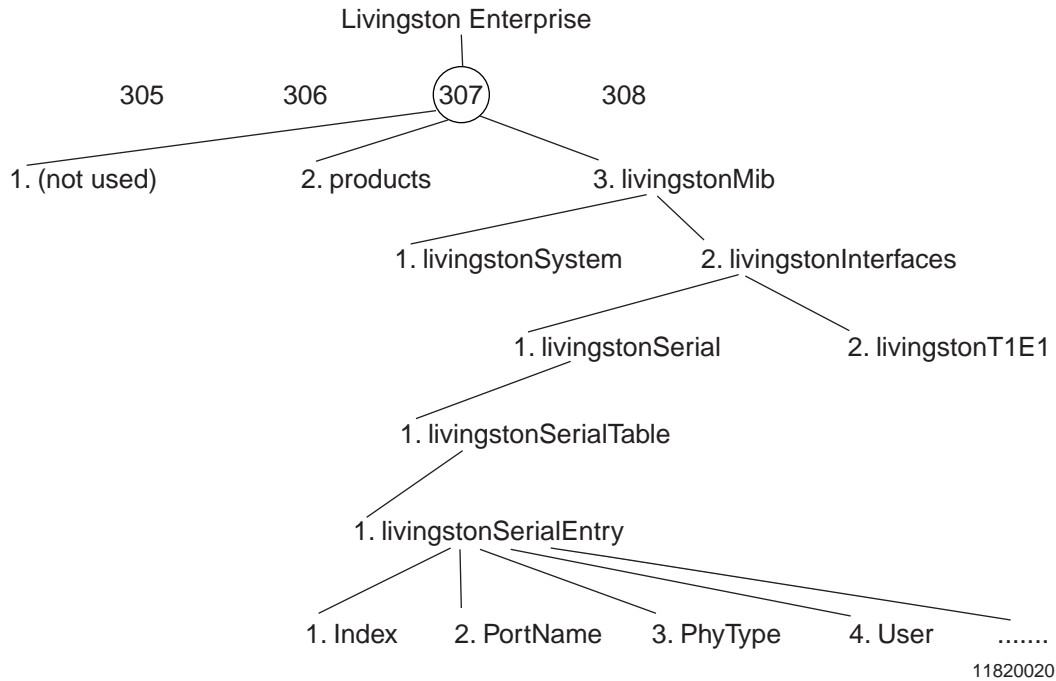
livingston.mib is the Livingston MIB definitions file SNMP tools can read and use to query SNMP agents for information about PortMaster products. The Livingston enterprise extensions to the MIB are located in the latter part of this file.

The **livingston.mib** file can be found in the SNMP directory of the ComOS software, or on the web at: **<http://www.livingston.com/Forms/one-click-dnload.html>**. To view the file with a browser, select SNMP—Livingston MIB in the Miscellaneous pop-up menu and then click the Download button.

The SNMP manager queries the agent by means of Object Identifiers, or OIDs, which are numbers associated with the objects defined in the MIB. For example, to set or return information on serial port S0 on a particular PortMaster, the request must contain an OID that uniquely identifies the port in the Livingston Enterprise name space, which is 307 (see Figure 2-1).

Note – The full OID to get to the Livingston enterprise area is 1.3.6.1.4.1.307.

Figure 2-1 livingston.mib structure.



In this example, the OID for serial port S0 is 307.3.2.1.1.2.0, which breaks out as follows:

- 307 refers to the Livingston Enterprise namespace
- 3 refers to the MIB
- 2 refers to interfaces
- 1 refers to serial interfaces
- 1 refers to the serial interfaces table
- 1 refers to an entry in the Livingston serial table (see Table 2-3)
- 2 refers to the port name
- 0 refers to the S0 port

A request containing this OID returns the port name assigned to port S0. Table 2-3, shows a partial view of the Livingston serial table, which is read top to bottom, left to right. The port name (reading down the left column) is identified first, then (reading across) the port number.

As another example, OID 307.3.2.1.1.1.5.2 returns the session identifier on port S2.

Table 2-3 Livingston Serial Table.

	S0 (0)	S1 (1)	S2 (2)	S3 (3)	S4 (4)
1	Index	Index	Index	Index	Index
2	PortName	PortName	PortName	PortName	PortName
3	PhysType	PhysType	PhysType	PhysType	PhysType
4	User	User	User	User	User
5	SessionId	SessionId	SessionId	SessionId	SessionId
6	Type	Type	Type	Type	Type
7	Direction	Direction	Direction	Direction	Direction

Livingston Serial Interfaces

Table 2-4 lists serial interface information from the Livingston Extension section of the Livingston MIB. Note that modem-specific objects apply to the PortMaster 3 only.

Table 2-4 Serial Interface Table

Object	Definition
Index	A unique value for each serial interface.
PortName	A text string containing the name of the serial interface (for example, S0, W1, and so on).
PhysType	The type of physical serial interface, distinguished according to the physical/link protocol (s) being currently used on the interface.
User	Name of the active user. Blank if not active.
SessionId	A unique session identifier that matches the RADIUS session ID.

Table 2-4 Serial Interface Table (Continued)

Object	Definition
Type	The active type of service being provided by the serial interface.
Direction	The direction in which the active session was initiated.
PortStatus	The status of the serial interface.
Started	The amount of time this session has been active.
Idle	The amount of time this session has been idle.
InSpeed	An estimate of the current inbound bandwidth in bits per second of the serial interface.
OutSpeed	An estimate of the current outbound bandwidth in bits per second of the serial interface.
ModemName (PM3 only)	A text string containing the name of the true digital modem in use by the serial interface.
IpAddress	The IP address associated with the serial interface. When used as a network type port, this is the IP address of the remote user. When used as a device or login, this is the IP address of the host to which the user is connected.
ifDescr	A text string containing information about the network interface bound to the serial interface.
InOctets	The total number of octets received on the serial interface.
OutOctets	The total number of octets transmitted on the serial interface.
QOctets	The total number of octets queued on the serial interface.
ModemStatus	The status of the modem used by the serial interface.
ModemCompression (PM-3 only)	The compression being used in the modem or by the serial interface.
ModemProtocol (PM-3 only)	The error correcting protocol being used in the modem or by the serial interface.
ModemRetrains (PM-3 only)	The number of retrains attempted by the modem attached to the serial interface.

Table 2-4 Serial Interface Table (Continued)

Object	Definition
ModemRenegotiates (PM-3 only)	The number of renegotiates attempted by the modem attached to the serial interface.

Livingston T1/E1 Interfaces

Table 2-5 lists T1/E1 interface information from the Livingston Extension section of the Livingston MIB. T1/E1 interfaces are supported only on the PortMaster 3 Integrated Access Server.

Table 2-5 T1/E1 Interfaces

Object Type	Definition
Index	A unique value for each T1/E1 interface.
PhysType	The type of interface (T1 or E1).
Function	The configured function of the interface
Status	The current operational status of the interface. Operational status states and their ID numbers include the following: <ul style="list-style-type: none">• up (1)• down (2)• loopback (3)
Framing	The configured line framing. Line framing types and their ID numbers include the following: <ul style="list-style-type: none">• esf (1)• d4 (2)• crc4 (3)• fas (4)
Encoding	The configured line signal encoding.
PCM	The configured voice modulation.
ChangeTime	The amount of time this interface has been up or down.
RecvLevel	An estimate of the current receive signal level, in DB, of the interface.

Table 2-5 T1/E1 Interfaces (Continued)

Object Type	Definition
BlueAlarms	The total number of blue alarms on the interface.
YellowAlarms	The total number of yellow alarms on the interface.
CarrierLoss	The total number of times the interface has lost carrier.
SyncLoss	The total number of times the interface has lost frame synchronizations.
BipolarErrors	The total number of frame level CRC errors detected on the interface.
CRCErrors	The total number of frame level CRC errors detected on the interface.
SyncErrors	The total number of frame synchronization errors detected on the interface.

Livingston Modem Table

Table 2-6, lists modem information from the Livingston Extension section of the Livingston MIB. Modem objects are supported only on the PortMaster 3 Integrated Access Server.

Table 2-6 Modem Table

Object Type	Definition
livingstonModemIndex	A unique value for each modem interface.
livingstonModemPortName	A textual string containing the name of the serial interface (ie. S0, S1, etc).
livingstonModemStatus	A current state of the modem.
livingstonModemProtocol	The error correcting protocol being used in the modem.
livingstonModemCompression	The compression being used in the modem interface.
livingstonModemInSpeed	An estimate of the modem interface's current inbound bandwidth in bits per second.

Table 2-6 Modem Table (Continued)

Object Type	Definition
livingstonModemOutSpeed	An estimate of the modem interface's current outbound bandwidth in bits per second.
livingstonModemInByteCount	The total number of bytes received by the modem.
livingstonModemOutByteCount	The total number of bytes transmitted by the modem.
livingstonModemRetrains	The number of retrains attempted by the modem.
livingstonModemRenegotiates	The number of renegotiates attempted by the modem.
livingstonModemCalls	The number of times a call received by the modem.
livingstonModemDetects	The number of analog calls received by the modem.
livingstonModemConnects	The number of successful calls received by the modem.

Setting SNMP Monitoring

SNMP monitoring is used to set and collect information on SNMP-capable devices. This feature is most often used to monitor network statistics such as usage and error rate.

If the SNMP monitoring is on, the PortMaster accepts SNMP queries. If this parameter is off, all SNMP queries are ignored.

To turn SNMP monitoring on or off, use the following commands:

```
Command> set snmp on|off
Command> save all
Command> reboot
```

Setting SNMP Read and Write Community Strings

Community strings allow you to control access to the MIB information on selected SNMP devices. The read and write community strings act like passwords to permit access to the SNMP agent information. The read community string must be known by any device allowed to access or read the MIB information. The default read community string is **public**. The write community string must be known by any device before information can be set on the SNMP agent. The default write community string is **private**. Community strings should be set on SNMP agents so that configuration information is not changed by unauthorized users.

To use this feature, you must set both a read community string and a write community string for your network.

To set SNMP read and write community strings, use the following command:

```
Command> set snmp readcommunity|writecommunity String
```



Note – Use of the default write community string—**private**—is strongly discouraged. Because it is the default, it is known to all users and therefore provides no security. You should use some other value for the write community string.

Adding SNMP Read and Write Hosts

PortMaster products allow you to control SNMP security by specifying the IP addresses of the hosts that are allowed to access SNMP information. The specification of read and write hosts allows another level of security beyond the community strings. If SNMP hosts are specified, each host wanting to access SNMP information must not only possess the correct community string, it must also be on the read or write host list. This additional level of security allows only authorized SNMP managers to access or change sensitive MIB information.

You can also specify a list of hosts allowed to read or write SNMP information. You can permit all hosts (not recommended because this setting reduces system security), or you can deny all hosts.

To add SNMP read and write hosts, use the following command:

```
Command> add snmp host reader|writer any|none | Ipaddress
```

To delete read and write hosts, use the following command:

```
Command> delete snmp host reader|writer /ipaddress
```

Viewing SNMP Settings

Settings for SNMP monitoring, read and write community strings, and read and write hosts are stored in the SNMP table.

To display the SNMP table, use the following command:

```
Command> show table snmp
```