Using the BTS SNMP Agent

Revised: August 10, 2011, OL-25015-01

Introduction

This chapter explains how to use the Simple Network Management Protocol (SNMP) agent.

The BTS uses a SNMP agent to communicate with a service provider's network management system (NMS). Working together, the SNMP agent and NMS monitor and control BTS components on the managed network. The NMS does most of the processing and provides the majority of memory resources. The SNMP agent collects statistical data (traffic measurements) for the following BTS components:

• AINSVC	• Element Manager	• MGCP Adapter	• SNMP
• Announcement	• H323	• POTS-Feature Server	• SUA
• Audit	• INAP	• SCCP	• TCAP
• Billing	• ISDN	• SCTP	• TSA
• Call Processing	• ISUP SGA	• SIA	• Trunk
• DQOS	• M3UA	• SIM	

Using the SNMP agent a user can do the following:

- View and change the status of individual BTS components
- View and change the status of a group of BTS components
- View SNMP trap (alarm) reports

Managing User Access to the SNMP Agent

The SNMP agent has access levels. Tasks you can perform depend both on your user group (community) and CLI security privileges.

Table 7-1   *Managing SNMP Agent Access*

Task	Sample Command
Viewing all read user groups	`show snmpconfig type=readcommunity` The default is "public". A user needs read level access to: • Collect statistics on BTS components • View status on individual BTS components • View status on a group of BTS components
Viewing all write user groups	`show snmpconfig type=writecommunity` The default is "public". A user needs write level access to: • Change settings on individual BTS components • Change settings on a group of BTS components
Adding read user groups	`add snmpconfig type=readcommunity; value=.....; key1=command_level; value1=8;` Values are ASCII strings up to 64 characters.
Adding write user groups	`add snmpconfig type=writecommunity; value=.....; key1=command_level; value1=8;` Values are ASCII strings up to 64 characters.
Deleting read user groups	`delete snmpconfig type=readcommunity; value=.....` Values are ASCII strings up to 64 characters.
Deleting write user groups	`delete snmpconfig type=writecommunity; value=.....` Values are ASCII strings up to 64 characters.

Viewing SNMP Trap Reports

The SNMP agent sends traps to the NMS; each trap maps to an EMS alarm. Alarms not mapped to a specific trap map to a generic trap. Traps show you the following, depending on information availability:

- Severity level
- Alarm ID associated with the trap
- Alarm category
- Set/Cleared flag
- Component (instance) ID
- Component type
- Details of the trap
- Time that trap was generated

Table 7-2   Receiving Trap Reports

Task	Operation
Receiving traps reports from the SNMP agent	Add an entry to SNMPTRAPDEST including the following: • NMS IP address or hostname • Port number to receive traps • Community string (not used) • Owner string (not used) • Filter Types—This specifies which subsystem events to receive: – BILLING – CALLP – CONFIG – DATABASE – MAINTENANCE – OSS – SECURITY – SIGNALING – STATISTICS – SYSTEM – AUDIT • Filter Levels—This specifies which levels of events to receive: – DEBUG – INFO – WARNING – MINOR – MAJOR – CRITICAL ⚠ **Caution** Filters with DEBUG and/or INFO traps tax BTS resources.

Viewing and Managing BTS Components

Table 7-3   *Viewing and Managing BTS Components*

Task	Operation
Viewing individual BTS components	GET/GETNEXT • Primary and secondary EMS • Primary and secondary BDMS • Primary and secondary CA • Primary and secondary POTS/Centrex/Tandem FS • Primary and secondary AIN FS GET/GETNEXT on MIB State columns • MGW • TG • Subscriber Termination • Trunk Termination • SGP • DPC • SCTP Association **Note** GET/GETNEXT on the ControlState results in one of the following: insufficient-data, not all necessary fields are set or ready-to-commit.

Table 7-3   Viewing and Managing BTS Components (continued)

Task	Operation
Changing settings on BTS components	1. SET necessary fields (Mode column, TargetState column, etc). 2. SET on the ControlState column, using **1** (commit) to change the state. SET • Primary EMS • Primary BDMS • Primary CA • Primary POTS/Centrex/Tandem FS • Primary AIN FS SETs on MIB columns • MGW • TG • Subscriber Termination • Trunk Termination • SCTP Association
Viewing groups of BTS components	GET/GETNEXT on the following branch: .iso.org.dod.internet.private.enterprises.ipcell.opticall.statusControlBulk • MGW • TG • Subscriber Termination • Trunk Termination The Status Value column displays components as follows: • ; = separates each instance of a BTS component • | = separates status fields • enumerated states are the same as the component's OAMPTable • . = separates CIC and TGN_ID

Querying the SNMP Agent

Table 7-4   Querying the SNMP Agent

Task	Operation

Querying the SNMP agent directly	1. Open the /etc/snmp/conf/snmpd.conf file.
	2. In read-community enter a single user group for read access.
	3. In managers enter the IP address or hostname of NMS to query, enter multiple addresses separated by spaces.
	Note To keep communication with the Master Agent leave the localhost entry.
	4. Restart the SNMP agent, enter:
	`/etc/init.d/S98mibiisa stop`
	`/etc/init.d/S98mibiisa start`
	5. Query the SNMP agent using the read-community and port 13230.

Querying the SNMP agent via the Master Agent	1. Open the /etc/snmp/conf/snmpd.conf file. 2. In read-community enter a single user group for read access. 3. In managers ensure the localhost is an entry. 4. Change the SNMP configuration type and value: `add snmpconfig type=SETTING; value=COUPLE_SUN_AGENT` 5. Restart the Master Agent. 6. Log in as root. `kill ``ps -ef	grep -i sad	grep -v grep	awk '{print $2}'``` 7. Query the SNMP agent using the read-community and standard port 161.
Querying the MIBs version	1. Open the /opt/BTSsnmp/etc file. 2. View the main MIB, called "opticall.mib". The SNMP Agent supports SNMPv2c operations defined in opticall.mib. opticall.mib uses variables from other MIBs: – IPCELL-TC – SNMPv2-TC – SNMPv2-SMI

Enabling NMS to Query/Poll Solaris SNMP Agent

The EMS runs two SNMP agents as follows:

- SAD (SNMP agent adapter)
- Solaris SNMP agent

The active EMS node runs the SAD process, which converts the BTS 10200 specific events/alarms into SNMP traps and sends them to the configured SNMP Trap listeners or the NMSes. The SAD process handles the SNMPWALK/GET/GETNEXT/SET on the OIDs that are defined in the opticall.mib file. The SAD process also runs on the standby EMS, but does not perform any function.

Note   The SAD process does not run on the CA nodes.

Note   The CA runs only the standard Solaris SNMP agent.

The standard Solaris SNMP agent runs on both the active and standby EMS and CA nodes. Therefore, all the four nodes generate the solaris-level traps. The name of the standard Solaris SNMP agent is **mibiisa**, which runs on port number 13230. The Solaris SNMP agent can be used to collect the sun box related statistics and/or traps. Note that the **mibiisa** supports only those OIDs (object identifiers) that are defined in the SUN MIB.

Note   The active/standby EMS and active/standby CA nodes generate the solaris-level traps, whereas only the active EMS generates BTS-specific traps and sends them to NMS. The NMS can query/poll all the four nodes to receive the generated traps.

To enable the NMS to directly query the Solaris SNMP agent for a range of OIDs specified by SUN MIBs, and receive Solaris box-level traps, do the following:

1. Open the /etc/snmp/conf/snmpd.conf file.

2. Define the read-community as "public".

3. In the "Managers" field, enter the IP address or hostname of the NMS from where the user needs to send the SNMP query. Enter multiple addresses separated by spaces, but leave the "localhost" entry as is.

4. In the Trap field, configure the IP address or hostname of the NMS where the traps have to be sent.

5. Restart the SNMP agent, enter:

```
/etc/init.d/S98mibiisa stop
/etc/init.d/S98mibiisa start
```

6. Query the SNMP agent (using SNMPGET/SNMPWALK) from the Manager using the read-community and port 13230. For example, to get the system up time, enter the following command:

```
snmpwalk -c public -p 13230 prica07 system
```

The output appears as given below:

```
system.sysDescr.0 = Sun SNMP Agent
system.sysObjectID.0 = OID: enterprises.42.2.1.1
system.sysUpTime.0 = Timeticks: (279199168) 32 days, 7:33:11.68
system.sysContact.0 = System administrator
system.sysName.0 = prica07
system.sysLocation.0 = System administrators office
system.sysServices.0 = 72
```