Teldat Router

SNMP Agent

Doc. *DM512-I* Rev. *8.40*
September, 2000

INDEX

Chapter 1

Introduction to the SNMP protocol

1. Introduction

SNMP is an OSI layer 7 (application layer) protocol for monitoring router operating characteristics.

SNMP enables network hosts to read and modify some of the settings of the router’s operating characteristics. It allows software running on a remote host to contact the router over a network and get updating information about the router on request. Since SNMP software can access most of the configuration data, you do not have to enter commands at a remote terminal.

SNMP’s basic functions include:

- Collecting information and modifying router operating characteristics on behalf of remote SNMP users.
- Sending and receiving SNMP packets via the IP protocol.

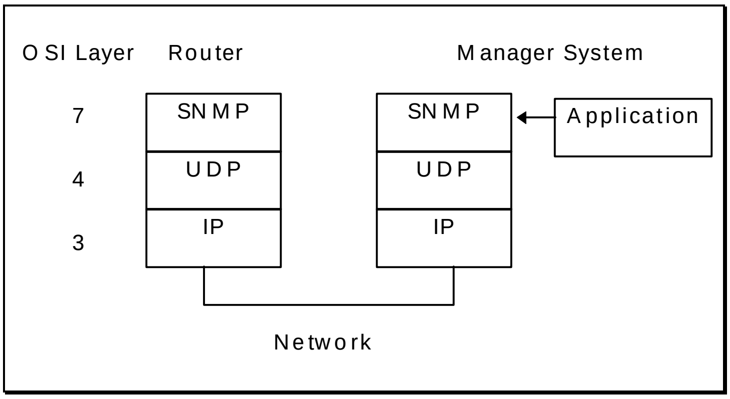

Figure 1: Protocol Layers of the SNMP Environment

The software that processes SNMP requests runs on the router and its called SNMP agent. The user program that makes SNMP requests runs on the user’s machine elsewhere in the network, not on the router, and its called SNMP manager. The SNMP agent at the router and the manager at the work station both use the UDP/IP protocol to exchange packets.

For more information about SNMP, refer to RFC 1157, *A Simple Network Management Protocol*. Refer to RFC’s 1212 and 1213 for descriptions of SNMP variables. The RFC explain how to use the protocols and formats of the packets that the protocols employ.

2. SNMP Packet Types

SNMP’s packets types reflect SNMP’s basic functions and types include the following:

- GET REQUEST packet. Travels user-to-router. Contains information requests by user software. Retrieves the exact variable requested.
- GET NEXT REQUEST packet. Travels user-to-router. Contains information requests by user software. Retrieves the next variable to the requested, following the order of the SNMP agent information tree.
- SET REQUEST packet. Travels user-to-router. Contains requests by user software to modify router operating characteristics.
- GET RESPONSE packet. Travels router-to-user. Contains the target router’s response to GET REQUEST, GET NEXT REQUEST and SET REQUEST packet, sent by the user software.
- TRAP MESSAGE packet. Travels router-to-user. Contains unsolicited information from the router. It is used to inform about the router problems or important events, like for example: “An interface on the router went down”.

3. Authentication

The entities which reside in the management stations and the network elements with which they communicate using the SNMP protocol have been named SNMP application entities. The pair formed by an SNMP agent and an arbitrary set of SNMP application entities (managers) are named SNMP community. Each SNMP community is named with a string of characters called community name or simply community.

The SNMP packets which travel between the SNMP application entities include the community name in one of their fields. In order to determine if an incoming message is a legitimate request from an authorized user, or an accidental request or a hostile attack from an unauthorized user, SNMP uses various sets of rules called authentication schema or simply authentication.

Authentication prevents unauthorized users from learning information about a router or modifying its operating characteristics. In particular, the authentication protocol ensures that both the SNMP agent and the SNMP manager ignore and discard requests from unauthorized users.

The current SNMP implementation offers an authentication schema because in each network element a permitted set of community names are defined. The community names have the following associated:

- the addresses of the managers who accept requests and the addresses of those which send alarms (traps)
- the variables the community name has access to
- the type of access that these have

To determine if an incoming message represents a legitimate request by an authorized user, or an accidental or malicious request by an unauthorized user, SNMP uses various sets of rules. Each such set of rules is called an *authentication scheme*. Authentication schemes rely on mathematical or cryptographic methods to authenticate messages.

For each SNMP community, you select an authentication scheme for users of that session. The current implementation of SNMP offers a single authentication scheme called *trivial*.

For further information on creating and using authentication schemes with SNMP, refer to RFC 1157, *A Simple Network Management Protocol*.

Chapter 2
Configuring the SNMP Agent

1. Displaying the SNMP Configuration Prompt

This chapter describes the SNMP agent configuration. After the desired options have been configured, you must save the configuration and restart the router to get the new configuration to take place. Configuration process is described with more detail in the following sections.

To access the SNMP configuration prompt, from the *Config>* prompt enter the following command

```
Config> PROTOCOL SNMP
SNMP user configuration
SNMP Config>
```

2. SNMP Configuration Commands

This section summarizes and then explains all the SNMP configuration commands. These commands allow you to specify network parameters for router interfaces that transmit SNMP packets.

Command	**Function**
? (HELP)	Lists available commands or lists the options associated with specific commands.
ADD	Adds a community to the list of SNMP communities, an IP address with mask to a community, or a subtree to a MIB view.
DELETE	Removes a community from the list of SNMP communities, an IP address with mask from a community, or a subtree to a MIB view.
SET	Sets a community’s access mode or view. A community’s access mode is one of the following: • Read and trap generation • Read, write and trap generation • Trap generation only Also allows setting of trap UDP port.
ENABLE	Enables SNMP agent and standard traps associated with named communities.
DISABLE	Disables SNMP agent and standard traps associated with named communities.
LIST	Displays the current communities, with their associated access modes, enabled traps, IP addresses, and views. Also displays all views and their associated MIB subtrees, as well as if the SNMP agent is active, and the trap UDP port.
EXIT	Return to the *Config>* prompt.

2.1. ? (HELP)

Use the **?** (HELP) command to list the commands that are available from the current promt level. You can also enter ? after a specific command name to list its options.

Syntax:

```
SNMP Config> ?
```

Example:

```
SNMP Config> ?
ADD
DELETE
SET
ENABLE
DISABLE
LIST
EXIT
SNMP Config>
```

2.2. ADD

Use the **ADD** command to add a community name to the list of SNMP communities, add an IP address to a community, or assign a portion of the MIB subtree to a view.

Syntax:

```
SNMP Config> ADD ?
COMMUNITY
ADDRESS
SUB_TREE
```

a) ADD COMMUNITY

Creates a community with the default parameters. Default parameters are: read and generation trap access mode, a MIB view of all, permitted access from all IP addresses and all disabled trap types associated to this community.

> ***NOTE: Use the SET COMMUNITY ACCESS command to assign access types to existing SNMP communities.***

Example:

```
SNMP Config> ADD COMMUNITY
Community name[]? Public
SNMP Config>
```

Community name Specifies the name of community (32 characters maximum). Special characters such as spaces, tabs, and so on, are not accepted.

b) ADD ADDRESS

Use the **ADD ADDRESS** command to add an IP address to a community. You must include the community name, the network address, and the network mask (in the standard *a.b.c.d* notation).

> ***NOTE: SNMP requests may arrive for any of the router's addresses***

You can specified more than one address for a community. To do this you must repeat the operation as many times as IP addresses you want to add.

SNMP requests will be accepted for each community if the outcome of the AND function between the IP address which originated the trap and the community network mask matches with the outcome of the AND function between the community IP address and its mask, in some of the address configured in the community. If no address is specified for the community, requests are accepted from any host. Addresses also specified which hosts are going to receive traps. If no address is specified no trap will be generated.

Example 1:

```
SNMP Config> ADD ADDRESS
Community name[]? public
IP Address [0.0.0.0]? 192.6.2.168
Mask [255.255.255.0]?
SNMP Config>
```

This operation causes that *public* community requests will be accepted if they come from any host of the 192.6.2 network, and traps are sent to the 192.6.2.168 address.

Example 2:

```
SNMP Config> ADD ADDRESS
Community name[]? public
IP Address [0.0.0.0]? 192.6.2.168
Mask [255.255.255.0]? 255.255.255.255
SNMP Config>
```

This operation causes that *public* community requests will be accepted only if they come from the 192.6.2.168 host, and traps are sent to that same host.

c) ADD SUB_TREE

Adds a portion of the MIB to a view or to create a new view. If no subtree is added, the view is the entire MIB. The **ADD SUB_TREE** command is used to manage MIB views. More than one subtree can be added to a view. To create a new view, use the **ADD SUB_TREE** command with the new view name.

To assign a view to one or more communities use the **SET COMMUNITY VIEW** command.

Example:

```
SNMP Config> ADD SUB_TREE
View name[]? mib2
MIB OID name[]? 1.3.6.1.2.1
SNMP Config>
```

View name Specify the name of the view (32 visual characters maximum). Special characters such as spaces, tabs, and so on, are not accepted.

MIB OID name Specifies the MIB Object ID for the subtree that will lead that all objects hanging on it, at the implemented MIB, will be displayed for this view.

2.3. DELETE

Use the **DELETE** command to delete:

- a specific address.
- a community and all of its addresses.
- a subtree from a view.

Syntax:

```
SNMP Config> DELETE ?
COMMUNITY
ADDRESS
SUB_TREE
```

a) DELETE COMMUNITY

Removes a community and its IP addresses.

Example:

```
SNMP Config> DELETE COMMUNITY
Community name[]? public
SNMP Config>
```

b) DELETE ADDRESS

Removes an address from a community.

Example:

```
SNMP Config> DELETE ADDRESS
Community name[]? public
IP Address [0.0.0.0]? 192.6.2.168
SNMP Config>
```

c) DELETE SUB_TREE

Removes a subtree from a view. If all subtrees are deleted, the view is also deleted and all the references to it from any community are removed.

Example:

```
SNMP Config> DELETE SUB_TREE
View name[]? mib2
MIB OID name[]? 1.3.6.1.2.1
SNMP Config>
```

2.4. SET

Use the **SET** command to assign a MIB view or the access mode to a community, or to set the SNMP UDP port numbers.

Syntax:

```
SNMP Config>  SET ?
COMMUNITY
TRAPS-PORT
TRAP-SENDING-PARAMETERS
```

a) SET COMMUNITY

Syntax:

```
SNMP Config> SET COMMUNITY ?
ACCESS
VIEW
```

SET COMMUNITY ACCESS

Assigns one of the access mode to a community. The access mode is one of the following:

READ_TRAP: Read and trap generation.

WRITE_READ_TRAP: Read-write and trap generation.

TRAP_ONLY: Trap generation.

Example:

```
SNMP Config> SET COMMUNITY ACCESS WRITE_READ_TRAP
Community name[]? Private
SNMP Config>
```

SET COMMUNITY VIEW

Assigns one MIB view to a community. You must previously create the view using the **ADD SUBTREE** command. If the *View name* is ALL, the community will have access to all the MIB.

Example:

```
SNMP Config> SET COMMUNITY VIEW
Community name[]? private
View name[]? Teldat
SNMP Config>
```

b) SET TRAPS-PORT

Specifies a UDP port number to send traps to the port number. The default value is 162, the standard port to send traps.

Example:

```
SNMP Config> SET TRAPS-PORT
UDP trap port[162]?
SNMP Config>
```

c) SET TRAP-SENDING-PARAMETERS

Permits you to configure the trap sending parameters. The sending of an SNMP trap can provoke an X.25 or ISDN call if the destination for these is on the other side of an interface of this type. For this reason it is advisable to group the traps you need to send in a buffer and sent them all together in order to reduce the number of calls carried out. The trap sending parameters which are configured from this option are:

Max time keeping traps. This is the time that a trap is stored in the buffer before being sent provided that the buffer has not reached maximum capacity. The traps are sent once the buffer is full or when the seconds indicated by this parameter have elapsed. The default value is 50 seconds.

Max number traps to keep. Size of the trap buffer to regroup. Number of traps that can be stored before being sent to their destination. In all cases the traps are sent individually, each in an UDP packet. The default value is 32 traps.

Max number of trap targets. Maximum number of trap destination. The SNMP communities can have one or various trap sending destination addresses associated. This parameter limits the number of destinations which effectively do have traps sent to them. The default value is 4 destination addresses.

Example:

```
SNMP Config> SET TRAP-SENDING-PARAMETERS
Max time keeping traps (seg)[50]?
Max number traps to keep[32]?
Max number of trap targets[4]?
SNMP Config>
```

2.5. ENABLE

Use the **ENABLE** command to enable the SNMP agent or specified traps on the router.

Syntax:

```
SNMP Config> ENABLE ?
SNMP
TRAP
DEFAULT CONFIGURATION
```

a) ENABLE SNMP

Enables SNMP.

Example:

```
SNMP Config> ENABLE SNMP
SNMP enabled
SNMP Config>
```

b) ENABLE TRAP

Enables specified traps or all traps for a community. The trap type is one of the following:

Trap type	**Description**
ALL	Enables all traps in a specified community.
COLD-START	Enables cold start traps in a specified community.
WARM-START	Enables warm start traps in a specified community.
LINK-DOWN	Enables link down traps in a specified community. A link down trap recognizes a failure in one of the router's interface. The link down trap PDU contains the name and value of the *ifIndex* instance for the affected interface as the first element of its variable-lists.
LINK-UP	Enables link up trap in a specified community. A link up trap shows that one of the router's interfaces that was fall down, has come up. The link up trap PDU contains the name and value of the *ifIndex* instance for the affected interface as the first element of its variable-lists.
AUTH-FAIL	Enables authentication failure traps in a specified community. Authentication failure traps shows that a SNMP request is not properly authenticated.
ENTERPRISE	Enables enterprise specific traps in a specified community. Enterprise specific traps recognize that some enterprise specific event has occurred. The specific-trap field identifies the particular trap that occurred. In the **Teldat Router**, the specific company traps are the ones configured as such in the Event Logging System (ELS).

Example:

```
SNMP Config> ENABLE TRAP ALL
Community name[]? private
SNMP Config>
```

c) ENABLE DEFAULT CONFIGURATION

Enables default configuration. The **ENABLE DEFAULT CONFIGURATION** command enables SNMP and originates a community called "teldat", with the following characteristics: it able to read, write, and generates traps, but does not send traps, accepts requests from any address, and has a complete MIB view. Default value of this command is Enabled.

Example:

```
SNMP Config> ENABLE DEFAULT CONFIGURATION
Default configuration is enabled
SNMP Config>
```

2.6. DISABLE

Use the **DISABLE** command to disable the SNMP agent or specified traps on the router.

Syntax:

```
SNMP Config> DISABLE ?
SNMP
TRAP
 DEFAULT CONFIGURATION
```

a) DISABLE SNMP

Disables SNMP.

Example:

```
SNMP Config> DISABLE SNMP
SNMP disabled
SNMP Config>
```

> ***NOTE: If the default configuration is enabled by default, SNMP is always enabled. This means SNMP cannot be disabled until the default configuration is disabled.***

b) DISABLE TRAP

Disables specified traps or all traps for a community. The trap type is one of the following:

Trap type	**Description**
ALL	Disables all traps in a specified community.
COLD-START	Disables cold start traps in a specified community.
WARM-START	Disables warm start traps in a specified.
LINK-DOWN	Disables link down traps in a specified community. A link down trap recognizes a failure in one of the router's interface. The link down trap PDU contains the name and value of the *ifIndex* instance for the affected interface as the first element of its variable-lists.
LINK-UP	Disables link up trap in a specified community. A link up trap shows that one of the router's interfaces that was fall down, has come up. The link up trap PDU contains the name and value of the *ifIndex* instance for the affected interface as the first element of its variable-lists.

AUTH-FAIL Disables authentication failure traps in a specified community. Authentication failure traps shows that a SNMP request is not properly authenticated.

ENTERPRISE Disables enterprise specific traps in a specified community. Enterprise specific traps recognize that some enterprise specific event has occurred. The specific-trap field identifies the particular trap that occurred. In the **Teldat Router**, the specific company traps are the ones configured as such in the Event Logging System (ELS).

Example:

```
SNMP Config> DISABLE TRAP ALL
Community name[]? Private
SNMP Config>
```

c) DISABLE DEFAULT CONFIGURATION

Disables default configuration.

Example:

```
SNMP Config> DISABLE DEFAULT CONFIGURATION
Default configuration is disabled
SNMP Config>
```

2.7. LIST

Use the **LIST** command to display the current configuration of SNMP: communities, access modes, traps, IP addresses, views, etc.

Syntax:

```
SNMP Config> LIST ?
ALL
COMMUNITY
VIEWS
TRAP-SENDING-PARAMETERS
```

a) LIST ALL

Displays all the SNMP configuration information.

Example:

```
SNMP Config> LIST ALL
Default configuration is disabled
SNMP is enabled
Trap port: 162
Max time keeping traps (sec): 50
Max number traps to keep:     32
Max number of trap targets:   4


        Community Name              IP Address       IP Mask
---------------------------------   --------------------------
public                              ALL
private                             192.6.2.168      255.255.255.255

        Community Name              Access
---------------------------------   --------------------------
public                              Read, Trap
private                             Read, Write, Trap

        Community Name              Enabled traps
---------------------------------   --------------------------
public                              None
private                             Cold Restart
                                    Warm Restart
                                    Link Down
                                    Link Up
                                    Authentication Failure
                                    Enterprise Specific

        Community name              Views
---------------------------------   --------------------------
public                              mib2
private                             teldat

        View name                   Subtree
---------------------------------   --------------------------
mib2                                1.3.6.1.2.1
teldat                              1.3.6.1.4.1.2007
SNMP Config>
```

NOTE: If the default configuration is enabled, SNMP is always enabled.

b) LIST COMMUNITY

Syntax:

```
SNMP Config> LIST COMMUNITY ?
ACCESS
ADDRESS
TRAPS
VIEW
```

LIST COMMUNITY ACCESS

Displays the access mode information for all communities.

Example:

```
SNMP Config> LIST COMMUNITY ACCESS
        Community Name              Access
---------------------------------   --------------------------
public                              Read, Trap
private                             Read, Write, Trap
SNMP Config>
```

LIST COMMUNITY ADDRESS

Displays the associated addresses information for all communities.

Example:

```
SNMP Config> LIST COMMUNITY ADDRESS
        Community Name              IP Address       IP Mask
---------------------------------   --------------------------
public                              ALL
private                             192.6.2.168      255.255.255.255
SNMP Config>
```

LIST COMMUNITY TRAPS

Displays the associated traps information for all communities.

Example:

```
SNMP Config>LIST COMMUNITY TRAPS
        Community Name              Enabled traps
---------------------------------   --------------------------
public                              None
private                             Cold Restart
                                    Warm Restart
                                    Link Down
                                    Link Up
                                    Authentication Failure
                                    Enterprise Specific
SNMP Config>
```

LIST COMMUNITY VIEW

Displays the view information associated to each community.

Example:

```
SNMP Config> LIST COMMUNITY VIEW
        Community name              Views
---------------------------------   --------------------------
public                              mib2
private                             teldat
SNMP Config>
```

c) LIST VIEW

Displays the current views for a specified SNMP community.

Example:

```
SNMP Config> LIST VIEW
        View name                    Subtree
---------------------------------   --------------------------
mib2                                1.3.6.1.2.1
teldat                              1.3.6.1.4.1.2007
SNMP Config>
```

d) LIST TRAP-SENDING-PARAMETERS

Displays the relative information on trap sending.

Example:

```
SNMP Config> LIST TRAP-SENDING-PARAMETERS
Max time keeping traps (sec): 50
Max number traps to keep:     32
Max number of trap targets:   4
SNMP Config>
```

2.8. EXIT

Use the **EXIT** command to return to the *Config>* prompt.

Syntax:

```
SNMP Config> EXIT
```

Example:

```
SNMP Config> EXIT
Config>
```

Chapter 3
Monitoring the SNMP Agent

1. Access to SNMP Monitoring Environment

To enter the SNMP monitoring environment, from the console (+) prompt, you must enter the following command:

```
+PROTOCOL SNMP
SNMP>
```

2. SNMP Monitoring Commands

Command	**Function**
? (HELP)	List available commands or lists the options associated with specific commands.
LIST	Displays the current configuration of SNMP communities, associated views, access modes, enabled traps, and IP addresses. Also displays all views and their associated MIB subtrees.
EXIT	Returns to the + prompt .

2.1. ? (HELP)

Use the **?** (HELP) command to list the commands that are available from the current prompt level. You can also enter **?** after a command to list its options.

Syntax:

```
SNMP> ?
```

Example:

```
SNMP> ?
LIST
EXIT
```

2.2. LIST

Use the **LIST** command to display the current configuration of SNMP communities, access modes, traps, IP addresses, views, etc.

Syntax:

```
SNMP> LIST ?
ALL
COMMUNITY
VIEW
```

a) *LIST ALL*

Displays the current configuration of SNMP communities.

Example:

```
SNMP> LIST ALL
SNMP>
```

b) *LIST COMMUNITY*

Syntax:

```
SNMP> LIST COMMUNITY ?
ACCESS
ADDRESS
TRAPS
VIEW
```

LIST COMMUNITY ACCESS

Displays the current configuration of access mode associated to SNMP communities.

Example:

```
SNMP> LIST COMMUNITY ACCESS
SNMP>
```

LIST COMMUNITY ADDRESS

Displays the current configuration of addresses associated to SNMP communities.

Example:

```
SNMP> LIST COMMUNITY ADDRESS
SNMP>
```

LIST COMMUNITY TRAPS

Displays the current configuration of traps associated to SNMP communities.

Example:

```
SNMP> LIST COMMUNITY TRAPS
SNMP>
```

LIST COMMUNITY VIEW

Displays the current configuration of views associated to SNMP communities.

Example:

```
SNMP> LIST COMMUNITY VIEW
SNMP>
```

c) *LIST VIEW*

Displays the views names associated to SNMP communities.

Example:

```
SNMP> LIST VIEW
SNMP>
```

2.3. EXIT

Use the **EXIT** command to return to the previous prompt level.

Syntax:

```
SNMP> EXIT
```

Example:

```
SNMP> EXIT
+
```