Teldat Router

SNMP Agent

Doc. *DM712-I* Rev. *10.00*
March, 2003

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. Therefore you can carry out centralized management of the routers which are in the network.

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.

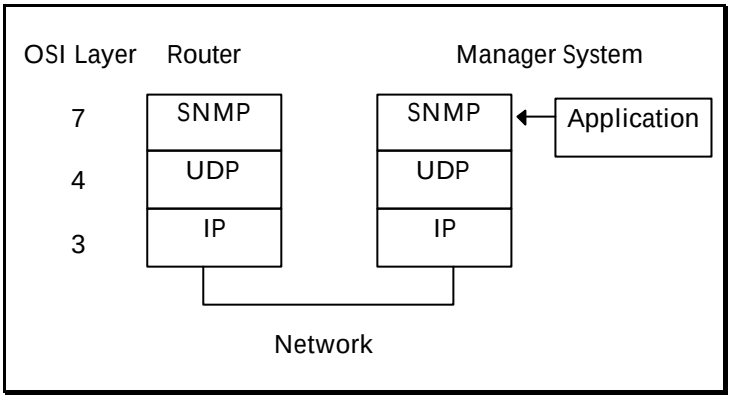

Figure 1: Protocol Layers of the SNMP Environment

The software that processes SNMP requests runs on the router and is 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 is known as 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 user software information requests. Retrieves the exact variable requested.
- GET NEXT REQUEST packet. Travels user-to-router. Contains user software information requests. Retrieves the next variable to the requested, following the order of the SNMP agent information tree.
- SET REQUEST packet. Travels user-to-router. Contains user software requests 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 provide information on router problems or important events, such as: "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 a simple authentication schema as 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

Each SNMP packet arriving at the router will be validated or dropped depending on whether it complies with the restrictions imposed by the authentication schema or not. Specifically, the accessed variable, its type of access and the source IP address of the SNMP packet must be included in those associated to the SNMP packet community name.

This authentication schema is configurable in each network element as will be explained in the following section.

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 protocol 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 environment, 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.
COMMUNITY	Permits you to add a new community to the SNMP communities list or modify the value of a parameter for an existing community: add or delete an IP address with associated mask for this community, configure the access mode, establish the community view and enable or disable distinct types of traps that the community members send to the SNMP managers.
DEFAULT-CONFIG	Enables the default configuration.
DISABLE	Disables the SNMP protocol.
ENABLE	Enables the SNMP protocol.
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, the traps destination UDP port and the values of certain parameters related with the sending of the traps.
NO	Deletes a community from the SNMP communities list and the associated IP addresses, deletes a "subtree" from a view (and the complete view if you delete the last associated "subtree"), disables the default configuration or establishes the default values for the traps destination UDP port or for the SNMP traps send parameters.
SUBTREE	Adds an MIB ("subtree") portion to a view or creates a new view.
TRAP	Configures the traps destination UDP port or certain parameters related with the SNMP traps sending.
EXIT	Return to the *Config>* prompt.

2.1. ? (HELP)

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

Syntax:

```
SNMP Config>?
```

Example:

```
SNMP Config>?
COMMUNITY           Adds a community or modifies parameters of an existing one
DEFAULT-CONFIG      Enables the default configuration
DISABLE             Disables SNMP
ENABLE              Enables SNMP
LIST                Displays SNMP configuration elements
NO                  Deletes an item, disables an option or sets default values
SUBTREE             Adds a portion of the MIB to a view or creates a view
TRAP                Sets trap UDP port or trap sending parameters
EXIT                Exits SNMP configuration menu
SNMP Config>
```

2.2. COMMUNITY

Use the **COMMUNITY** command to add a community name to the list of SNMP communities, or modify the value of an already existing community parameter: add or delete an IP address (with its mask) associated to this community, configure the access mode, set the community view and enable and disable various types of traps that the community members send to the SNMP managers.

Syntax:

```
SNMP config>COMMUNITY
Community name[]? public
default      creates a SNMP community with default values
access       sets community access
address      adds an address to a community
view         sets a view for a community
trap         enables traps of the type specified
no           deletes an address, deletes a view or disables traps
Type an option [default]?
SNMP config>
```

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

a) COMMUNITY community_name DEFAULT

Creates a community with the default parameters or establishes the said parameters for an already existing community. These are: read and trap generation access mode, view associated to all the MIB, permitted access from all IP addresses and all disabled trap types associated to this community.

NOTE: Use the COMMUNITY community_name ACCESS option to assign the SNMP communities access types. Use the COMMUNITY community_name ADDRESS option to set a determined IP addresss through which you can access a specific community. The COMMUNITY community_name VIEW option is used to limit a community view to certain "subtrees" or portions of the MIB and the COMMUNITY community_name TRAP is used to enable the types of traps that you wish the agent to send to the manager addresses configured for a community.

Example:

```
SNMP config>COMMUNITY public default
SNMP config>
```

Or:

```
SNMP config>COMMUNITY
Community name[]? public
default      creates a SNMP community with default values
access       sets community access
address      adds an address to a community
view         sets a view for a community
trap         enables traps of the type specified
no           deletes an address, deletes a view or disables traps
Type an option [default]?
SNMP config>
```

b) COMMUNITY community_name ACCESS

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

read-trap: Read and trap generation.

trap-only: Trap generation.

write-read-trap: Read-write and trap generation.

Example:

```
SNMP config>COMMUNITY public access write-read-trap
SNMP config>
```

Or:

```
SNMP config>COMMUNITY
Community name[]? public
default      creates a SNMP community with default values
access       sets community access
address      adds an address to a community
view         sets a view for a community
trap         enables traps of the type specified
no           deletes an address, deletes a view or disables traps
Type an option [default]? access
read-trap            read SNMP variables and generate traps
trap-only            only generate traps
write-read-trap      read and write SNMP variables and generate traps
Type an option [read-trap]? write-read-trap
SNMP config>
```

c) COMMUNITY community_name ADDRESS

Use the **COMMUNITY community_name ADDRESS** option 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. I.e. petitions will be accepted from any device in the subnets defined by the masks. 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>COMMUNITY public address 192.6.2.168 255.255.255.0
SNMP config>
```

Or:

```
SNMP config>COMMUNITY
Community name[]? public
default      creates a SNMP community with default values
access       sets community access
address      adds an address to a community
view         sets a view for a community
trap         enables traps of the type specified
no           deletes an address, deletes a view or disables traps
Type an option [default]? address
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>COMMUNITY public address 192.6.2.168 255.255.255.255
SNMP config>
```

Or:

```
SNMP config>COMMUNITY
Community name[]? public
default      creates a SNMP community with default values
access       sets community access
address      adds an address to a community
view         sets a view for a community
trap         enables traps of the type specified
no           deletes an address, deletes a view or disables traps
Type an option [default]? address
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.

d) COMMUNITY community_name VIEW

Assigns an MIB view to a community. The view must be previously created through the **SUBTREE** command. If the *View name* is "ALL", the community will have access to all the MIB.

Example:

```
SNMP config>COMMUNITY private view Teldat
SNMP config>
```

Or:

```
SNMP config>COMMUNITY
Community name[]? private
default      creates a SNMP community with default values
access       sets community access
address      adds an address to a community
view         sets a view for a community
trap         enables traps of the type specified
no           deletes an address, deletes a view or disables traps
Type an option [default]? view
View name[]? Teldat
SNMP config>
```

e) COMMUNITY community_name TRAP

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

Trap type	Description
ALL	Enables all types of traps in a specified community.
AUTHENTICATION-FAILURE	Enables the "authentication failure" trap in a specified community. The "authentication failure" trap indicates that an SNMP petition has not been correctly authenticated.
COLD-START	Enables cold start traps in a specified community.
ENTERPRISE-SPECIFIC	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 enterprise traps are the ones configured as such in the Event Logging System (ELS).
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 down is now 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.
WARM-START	Enables warm start traps in a specified community.

Example:

```
SNMP config>COMMUNITY private trap all
SNMP config>
```

Or:

```
SNMP config>COMMUNITY
Community name[]? private
default      creates a SNMP community with default values
access       sets community access
address      adds an address to a community
view         sets a view for a community
trap         enables traps of the type specified
no           deletes an address, deletes a view or disables traps
Type an option [default]? trap
all                          enables all trap types
authentication-failure       enables authentication failure traps
cold-start                   enables cold start traps
enterprise-specific          enables enterprise specific traps
link-down                    enables link down traps
link-up                      enables link up traps
warm-start                   enables warm start traps
Type an option [all]?
SNMP config>
```

f) COMMUNITY community_name NO

This command permits you to delete an IP address (with its mask) associated to a specific community, delete the view associated to this community or disable one trap type or disable all trap types for a community.

Syntax:

```
SNMP config>COMMUNITY
Community name[]? public
default      creates a SNMP community with default values
access       sets community access
address      adds an address to a community
view         sets a view for a community
trap         enables traps of the type specified
no           deletes an address, deletes a view or disables traps
Type an option [default]? no
address      deletes a community address
view         deletes the association of a view to a commmunity
trap         disables traps of the type specified
Type an option [address]?
IP Address [0.0.0.0]?
SNMP config>
```

COMMUNITY community_name NO ADDRESS

Deletes a community address.

Example:

```
SNMP Config>COMMUNITY public no address
IP Address [0.0.0.0]? 192.6.2.168
SNMP Config>
```

COMMUNITY community_name NO TRAP

Disables a determined trap 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.
AUTHENTICATION-FAILURE	Disables authentication failure traps in a specified community. Authentication failure traps shows that a SNMP request has not been correctly authenticated.
COLD-START	Disables cold start traps in a specified community.
ENTERPRISE-SPECIFIC	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).
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 down is now 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.
WARM-START	Disables warm start traps in a specified community.

Example:

```
SNMP config>COMMUNITY private no trap all
```

Or:

```
SNMP config>COMMUNITY
Community name[]? private
default      creates a SNMP community with default values
access       sets community access
address      adds an address to a community
view         sets a view for a community
trap         enables traps of the type specified
no           deletes an address, deletes a view or disables traps
Type an option [default]? no
address      deletes a community address
view         deletes the association of a view to a commmunity
trap         disables traps of the type specified
Type an option [address]? trap
all                          disables all traps
authentication-failure       disables authentication failure traps
cold-start                   disables cold start traps
enterprise-specific          disables enterprise specific traps
link-down                    disables link down traps
link-up                      disables link up traps
warm-start                   disables warm start traps
Type an option [all]?
SNMP config>
```

COMMUNITY community_name NO VIEW

Deletes a view assigned to a community so that this has access to all the MIB.

Example:

```
SNMP Config>COMMUNITY public NO VIEW
SNMP Config>
```

2.3. DEFAULT-CONFIG

Enables default configuration. The **DEFAULT-CONFIG** command enables SNMP and creates a community called "teldat", with the following characteristics: it has all permissions (read, write, etc.), 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>DEFAULT-CONFIG
Default configuration is enabled
SNMP config>
```

2.4. DISABLE

Disables the SNMP protocol.

Example:

```
SNMP Config>DISABLE
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.

2.5. ENABLE

Enables the SNMP protocol.

Example:

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

2.6. SUBTREE

Adds a portion of the MIB to a view or creates a new view. This command is used to configure the MIB views. More than one subtree can be added to the same view. To create a new view, use the this command with the new view name.

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

Example:

```
SNMP config>SUBTREE
View name[]? mib2
MIB OID name[]? 1.3.6.1.2.1
SNMP config>
```

View name Specify the name of the view (32 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 off it, in the implemented MIB, will be displayed for this view.

2.7. TRAP

Permits you to configure the UDP port to which traps are sent or one of the parameters used to determine sending conditions for the said traps.

Syntax:

```
SNMP config>TRAP ?
PORT                    Allows setting of trap UDP port
SENDING-PARAMETERS      Allows setting of trap sending parameters
SNMP config>
```

a) TRAP PORT

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

Example:

```
SNMP Config>TRAP PORT
UDP trap port[162]?
SNMP Config>
```

b) TRAP SENDIG-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. Additionally, it's a good idea to make sure that the address

which has been configured as the traps destination is reachable (that the call has already been established, following the previous example), so that the probability of the traps being lost along the route diminishes. However, on other occassions, you may wish to receive the traps as quickly as posible. Therefore it is convienent to minimize the number of traps saved in the buffer before being sent or the maximum time that a trap can wait until it is sent. In this particular case, it is not recommendable to check if the manager station which receives the traps is reachable. This could introduce a certain delay if you have to wait for a response to the ECHO UDP or ICMP which is sent to each configured destination from the device to see if these are accessible.

The trap sending parameters which are configured from this option are:

NUMBER	Size of the trap buffer to regroup: number of traps that can be stored before being sent to their destination.
REACHABILITY-CHECKING	Indicates if reachability checks for the manager stations configured as trap destination should be carried out before being sent.
TARGETS	Maximum number of trap destinations (SNMP managers for those that send traps).
TIME	Time that a trap is stored in the buffer before being sent (provided that the buffer does not previously reach its maximum capacity).

Syntax:

```
SNMP config>TRAP SENDING-PARAMETERS ?
NUMBER                    Maximum number of traps to keep before sending
REACHABILITY-CHECKING     Reachability checking before sending traps
TARGETS                   Maximum number of trap targets (managers)
TIME                      Max time keeping traps in buffer before sending
SNMP config>
```

TRAP SENDING-PARAMETERS NUMBER

Configures the size of the trap buffer to regroup. I.e the 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.

Example:

```
SNMP config>TRAP SENDING-PARAMETERS NUMBER
Max number traps to keep[32]?
SNMP config>
```

TRAP SENDING-PARAMETERS REACHABILITY-CHECKING

This parameter indicates if you are going to execute the reachability checking for the manager stations configured as trap destinations before sending. If this parameter is set to 0 (disabling the destination reachability check), as well as transmitting the traps without worrying whether the destination is available or not, it makes no sense to periodically send the ECHO UDP or ICMP which is used to find out which managers can be reached. These would be those that a response is received from. Therefore, disabling the check implies not transmitting the ECHO. The permited values for this variable are:

0- The traps are emited without checking if the destinations are reachable and the ECHO UDP and ECHO ICMP are not used.

1- Enables the checking, this is carried out through sending ECHO UDP.

2- Enables the checking, this is carried out through sending ECHO ICMP.

Example:

```
SNMP config>TRAP SENDING-PARAMETERS REACHABILITY-CHECKING
Check if manager is reachable before sending traps:
  0-No
  1-Yes UDP
  2-Yes ICMP[1]?
SNMP config>
```

TRAP SENDING-PARAMETERS TARGETS

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

Example:

```
SNMP config>TRAP SENDING-PARAMETERS TARGETS
Max number of trap targets[4]?
SNMP config>
```

TRAP SENDING-PARAMETERS TIME

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.

Example:

```
SNMP config>TRAP SENDING-PARAMETERS TIME
Max time keeping traps (sec)[50]?
SNMP config>
```

2.8. NO

Use the **NO** command to:

- Delete a community and all of its associated IP addresses.
- Disable the default configuration
- Delete a subtree from a view.
- Establish the default values for a traps destination port or for the parameters used to determine the sending conditions for the said traps.

Syntax:

```
SNMP config>NO ?
COMMUNITY           Removes a community and its IP addresses
DEFAULT-CONFIG      Disables the default configuration
SUBTREE             Removes a subtree from a view
TRAP                Sets default values to trap port or sending parameters
SNMP config>
```

a) NO COMMUNITY

Removes a community and its IP addresses.

Example:

```
SNMP Config>NO COMMUNITY
Community name[]? public
SNMP Config>
```

b) NO DEFAULT-CONFIG

Disables the default configuration.

Example:

```
SNMP config>NO DEFAULT-CONFIG
Default configuration is disabled
SNMP config>
```

c) NO SUBTREE

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>NO SUBTREE
View name[]? mib2
MIB OID name[]? 1.3.6.1.2.1
SNMP Config>
```

d) NO TRAP

This command permits you to set the default values for the traps destination UDP port or for the parameters used to determine the sending conditions for the said traps.

Syntax:

```
SNMP config>NO TRAP ?
PORT                    Sets default value to trap port
SENDING-PARAMETERS      Sets default values to trap sending parameters
SNMP config
```

NO TRAP PORT

Sets the default value as the UDP port number to which the traps are sent: 162, the traps sending standard port.

Example:

```
SNMP config>NO TRAP PORT
SNMP config>
```

NO TRAP SENDING-PARAMETERS

Through this command, you configure the parameters related to the traps sending to their default values.

Syntax:

```
SNMP config>NO TRAP SENDING-PARAMETERS ?
NUMBER                     Sets default value to max number of traps to keep
REACHABILITY-CHECKING      Sets reachability-checking mechanism to UDP echo
TARGETS                    Sets default value to max number of trap targets
TIME                       Sets default value to max time keeping traps
SNMP config>
```

· NO TRAP SENDING-PARAMETERS NUMBER

Configures the size of the trap buffer to regroup. I.e. the number of traps that can be stored before being sent to their destination, giving it the default value: 32 traps.

Example:

```
SNMP config>NO TRAP SENDING-PARAMETERS NUMBER
SNMP config>
```

· NO TRAP SENDING-PARAMETERS REACHABILITY-CHECKING

The **NO TRAP SENDING-PARAMETERS REACHABILITY-CHECKING** command enables the destination reachability checking before beginning to send traps. This is carried out by sending ECHO UDP.

Example:

```
SNMP config>NO TRAP SENDING-PARAMETERS REACHABILITY-CHECKING
SNMP config>
```

· NO TRAP SENDING-PARAMETERS TARGETS

Configures the maximum number of trap destinations to the default value: as a maximum, traps are sent to four destination addresses.

Example:

```
SNMP config>NO TRAP SENDING-PARAMETERS TARGETS
SNMP config>
```

· NO TRAP SENDING-PARAMETERS TIME

Sets the maximum time that a trap can be maintained in the buffer before being forwarded (if the buffer does not reach maximum capacity) to the default value: 50 seconds.

Example:

```
SNMP config>NO TRAP SENDING-PARAMETERS TIME
SNMP config>
```

2.9. 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                          Displays all the SNMP configuration information
COMMUNITY                    Displays current communities configuration
TRAP-SENDING-PARAMETERS      Displays the relative information on trap sending
VIEW                         Displays the current views configured
SNMP config>
```

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
Check if manager is reachable before sending traps: YES – UDP

        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 Start
                                  Warm Start
                                  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       Displays the access mode information for all communities
ADDRESS      Displays the associated addresses information for all communities
TRAPS        Displays the associated traps information for all communities
VIEW         Displays the view information associated to each community
SNMP config>
```

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 information on the traps associated to all communities.

Example:

```
SNMP Config>LIST COMMUNITY TRAPS
        Community Name              Enabled traps
--------------------------------   -------------------------
public                              None
private                             Cold Start
                                    Warm Start
                                    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 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
Check if manager is reachable before sending traps: YES – UDP

SNMP Config>
```

d) LIST VIEW

Displays information on the view defined in the system, with the MIB portions or "subtrees" associated to each.

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>
```

2.10. CONFIGURATION EXAMPLE

Below you will see a configuration example displayed in text mode obtained from the **SHOW CONFIG** command:

```
SNMP config>LIST ALL
Default configuration is disabled
SNMP is enabled
Trap port: 162
Max time keeping traps (seg):                        40
Max number traps to keep:                            30
Max number of trap targets:                          5
Check if manager is reachable before sending traps: YES - ICMP


        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 Start
                                   Warm Start
                                   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>
```

```
SNMP Config>SHOW CONFIG
; Showing Menu and Submenus Configuration ...
; Router C4i IPSec 1 16 Version 10.0.0CAI

   no default-config
   subtree mib2 1.3.6.1.2.1
   subtree teldat 1.3.6.1.4.1.2007
;
   community public default
   community public view mib2
;
   community private default
   community private access write-read-trap
   community private address 192.6.2.168 255.255.255.255
   community private view teldat
   community private trap all
;
   trap sending-parameters time 40
   trap sending-parameters number 30
   trap sending-parameters targets 5
   trap sending-parameters reachability-checking 2
SNMP config>
```

2.11. EXIT

Use the **EXIT** command to return to the configuration prompt.

Syntax:

```
SNMP Config>EXIT
```

Example:

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

Chapter 3
Monitoring the SNMP Agent

1. Accessing the 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	Indicates if the SNMP protocol is enabled or disabled.
	Displays the traps destination UDP port.
	Displays the communities, their access modes, enabled traps, IP addresses and associated views.
	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 at the level the router is programmed. You can also enter **?** after a specific 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 all the information for the currently active SNMP configuration.

Example:

```
SNMP>LIST ALL
SNMP is enabled
Trap port: 162

        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 Start
                                   Warm Start
                                   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>
```

b) LIST COMMUNITY

Syntax:

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

LIST COMMUNITY ACCESS

Displays information on the access mode for all the communities.

Example:

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

LIST COMMUNITY ADDRESS

Displays information on the addresses associated to all the communities.

Example:

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

LIST COMMUNITY TRAPS

Displays information on the traps associated to all the communities.

Example:

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

LIST COMMUNITY VIEW

Displays information on the view associated to each community.

Example:

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

c) LIST VIEW

Displays information on the views defined in the system.

Example:

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

2.3. EXIT

Use the **EXIT** command to return to the console prompt.

Syntax:

```
SNMP>EXIT
```

Example:

```
SNMP>EXIT
+
```