

**SeeBeyond™ eBusiness Integration Suite**

# e\*Gate Integrator SNMP Agent User's Guide

*Release 4.5.3*



The information contained in this document is subject to change and is updated periodically to reflect changes to the applicable software. Although every effort has been made to ensure the accuracy of this document, SeeBeyond Technology Corporation (SeeBeyond) assumes no responsibility for any errors that may appear herein. The software described in this document is furnished under a License Agreement and may be used or copied only in accordance with the terms of such License Agreement. Printing, copying, or reproducing this document in any fashion is prohibited except in accordance with the License Agreement. The contents of this document are designated as being confidential and proprietary; are considered to be trade secrets of SeeBeyond; and may be used only in accordance with the License Agreement, as protected and enforceable by law. SeeBeyond assumes no responsibility for the use or reliability of its software on platforms that are not supported by SeeBeyond.

e\*Gate, e\*Insight, e\*Way, e\*Xchange, e\*Xpressway, eBI, iBridge, Intelligent Bridge, IQ, SeeBeyond, and the SeeBeyond logo are trademarks and service marks of SeeBeyond Technology Corporation. All other brands or product names are trademarks of their respective companies.

© 1999–2002 by SeeBeyond Technology Corporation. All Rights Reserved. This work is protected as an unpublished work under the copyright laws.

**This work is confidential and proprietary information of SeeBeyond and must be maintained in strict confidence.**

Version 20021108092947.

# Contents

---

## Chapter 1

<b>Introduction</b>	<b>5</b>
Document Purpose and Scope	5
Intended Audience	5
Organization of Information	5
Writing Conventions	6
Supporting Documents	7
SeeBeyond Web Site	7

---

## Chapter 2

<b>Overview of SNMP Agent</b>	<b>8</b>
Using the MIB to Monitor and Control e*Gate	8
Example of messages sent using MIB commands to control e*Gate	8
System Requirements	10

---

## Chapter 3

<b>Installation</b>	<b>12</b>
Pre-Installation	12
After Installation	12
Windows 2000 and Windows NT	12
Windows Files	12
Installation Procedure	13
UNIX	15
UNIX Files	15
Installation Procedure	15

---

Chapter 4

**SNMP Agent Component Configuration** **17**

Creating an SNMP Agent Component **17**

Modifying SNMP Agent Component Properties **18**

    Configuring a Single Master SNMP Agent **19**

    Java method: eventSend() **19**

---

Chapter 5

**Modifying the Configuration** **20**

Matching Parameters **20**

Modifying the Configuration File **21**

    Configuration Parameters **21**

Sample Configuration File **22**

---

Chapter 6

**Troubleshooting** **23**

---

Appendix A

**MIB File** **24**

Description **24**

    MIB Objects **25**

    stcTraps **30**

---

Appendix B

**Password Encryption and the Configuration File** **34**

Encrypting a Password on an Existing SNMP Agent **34**

**Index** **35**

# Introduction

This chapter introduces you to this guide, its general purpose and scope, and its organization. It also provides sources of related documentation and information.

---

## 1.1 Document Purpose and Scope

This guide provides the user with a general overview of an e\*Gate system as it relates to the SNMP (Simple Network Management Protocol) Agent as it monitors and controls hosts and gateways on a network. It describes the procedures necessary to install, configure, modify a configuration, and troubleshoot the setup and system.

---

## 1.2 Intended Audience

The reader of this guide is presumed to be a system administrator with responsibility for monitoring the e\*Gate system, and should be completely familiar with the SNMP base enterprise management/monitoring system which will communicate with the e\*Gate SNMP Agent.

---

## 1.3 Organization of Information

This document is organized topically as follows:

- **Chapter 1 “Introduction”** gives a general preview of this document, its purpose, scope, and organization.
- **Chapter 2 “Overview of SNMP Agent”** discusses the general features, architecture, and system requirements of the SNMP Agent.
- **Chapter 3 “Introduction”** explains how to install the e\*Gate SNMP Agent on Windows or UNIX.
- **Chapter 4 “SNMP Agent Component Configuration”** explains how to create an SNMP Agent Component in a schema.

- **Chapter 5 “SNMP Agent Component Configuration”** instructs the user how modify the configuration file and configure the SNMP Agent’s parameters.
- **Chapter 6 “Troubleshooting”** provides the user with guidelines as to troubleshooting problems within the SNMP Agent.

In addition, there are two appendices:

- **Appendix A “MIB File”** defines for the user all the resources and attributes available for monitoring by the e\*Gate SNMP Agent.
- **Appendix B “Password Encryption and the Configuration File”** discusses updating the `stcsnmpa.conf` file with an encrypted password.

---

## 1.4 Writing Conventions

The writing conventions listed in this section are observed throughout this document.

### Hypertext Links

When you are using this guide online, cross-references are also hypertext links and appear in **blue text** as shown below. Click the **blue text** to jump to the section.

For information on these and related topics, see **“Parameter, Function, and Command Names” on page 7**.

### Command Line

Text to be typed at the command line is displayed in a special font as shown below.

```
java -jar ValidationBuilder.jar
```

Variables within a command line are set in the same font and bold italic as shown below.

```
stcregutil -rh host-name -rs schema-name -un user-name  
-up password -ef output-directory
```

### Code and Samples

Computer code and samples (including printouts) on a separate line or lines are set in Courier as shown below.

```
Configuration for BOB_Promotion
```

However, when these elements (or portions of them) or variables representing several possible elements appear within ordinary text, they are set in *italics* as shown below.

*path* and *file-name* are the path and file name specified as arguments to **-fr** in the **stcregutil** command line.

### Notes and Cautions

Points of particular interest or significance to the reader are introduced with *Note*, *Caution*, or *Important*, and the text is displayed in *italics*, for example:

**Note:** *The Actions menu is only available when a Properties window is displayed.*

## User Input

The names of items in the user interface such as icons or buttons that you click or select appear in **bold** as shown below.

Click **Apply** to save, or **OK** to save and close.

## File Names and Paths

When names of files are given in the text, they appear in **bold** as shown below.

Use a text editor to open the **ValidationBuilder.properties** file.

When file paths and drive designations are used, with or without the file name, they appear in **bold** as shown below.

In the **Open** field, type **D:\setup\setup.exe** where **D:** is your CD-ROM drive.

## Parameter, Function, and Command Names

When names of parameters, functions, and commands are given in the body of the text, they appear in **bold** as follows:

The default parameter **localhost** is normally only used for testing.

The Monk function **iq-put** places an Event into an IQ.

You can use the **stccb** utility to start the Control Broker.

---

## 1.5 Supporting Documents

The following SeeBeyond documents provide additional information about the e\*Gate Integrator SNMP Agent:

- *e\*Gate Integrator Alert Agent User's Guide*
- *e\*Gate Integrator Alert and Log File Reference Guide*
- *e\*Gate Integrator Installation Guide*
- *e\*Gate Integrator System Administration and Operations Guide*
- *e\*Gate Integrator User's Guide*
- *Monk Developer's Reference*

If necessary, refer to the appropriate Microsoft Windows or UNIX documents.

---

## 1.6 SeeBeyond Web Site

The SeeBeyond Web site is your best source for up-to-the-minute product news and technical support information. The site's URL is:

<http://www.SeeBeyond.com>

# Overview of SNMP Agent

The e\*Gate SNMP (Simple Network Management Protocol) Agent is used to monitor and control remote hosts and gateways on a network. The Agent monitors all resources within a given schema in the e\*Gate Integrator system. If you use the Agent, you can use it exclusively by itself, or you can use it in conjunction with the e\*Gate Monitor.

The Agent is designed to work with systems that use SNMP Protocol version 1. Once installed and configured, the Agent operates as a standard service (Windows 2000 or Windows NT 4) or as a daemon (UNIX) and communicates with the SNMP Manager.

You can install the Agent on any Participating Host in the e\*Gate system. Initial configuration is performed as part of the installation process. After installation, you need to create an SNMP component in your schema and modify the Notification Routing script. If you need to change the configuration parameters later, you must modify the configuration file created during installation, the properties of the SNMP component, and the Notification Routing script.

---

## 2.1 Using the MIB to Monitor and Control e\*Gate

The e\*Gate SNMP Agent communicates with the SNMP Manager through the SeeBeyond Management Information Base (MIB). Using the definitions and attributes contained in the file `stc_mib.txt`, you are able to monitor and control e\*Gate by requesting the current state of the modules in your system and sending requests to update or change the state. See “[MIB File](#)” on page 24 for a listing of the MIB objects and their possible values.

**Important:** Do not edit `stc_mib.txt`.

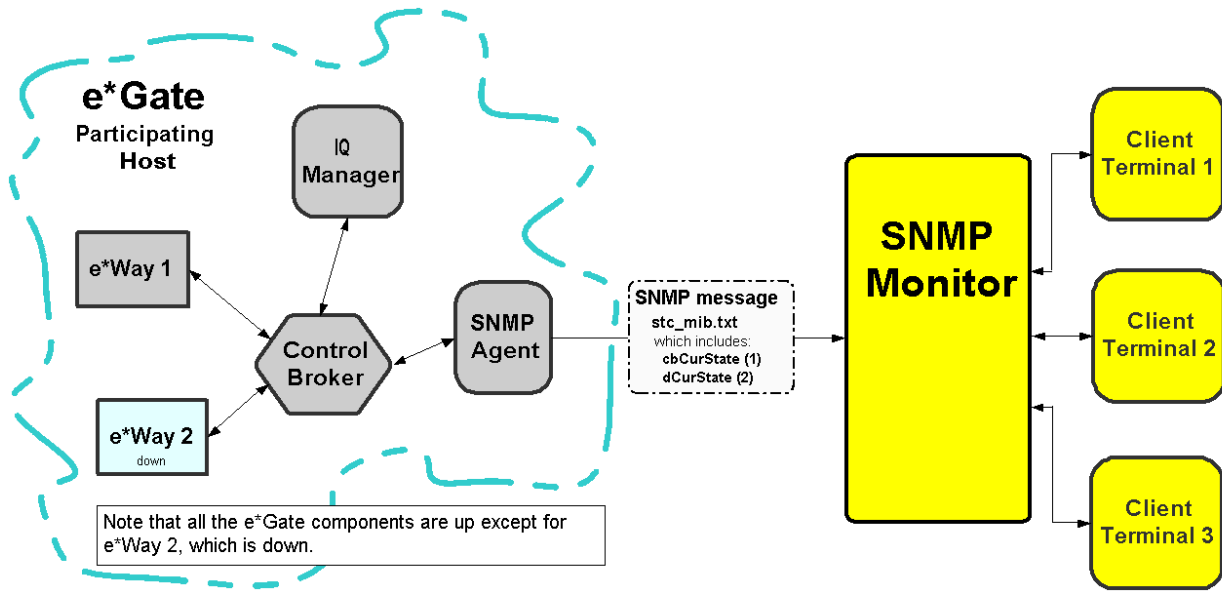
### Example of messages sent using MIB commands to control e\*Gate

In [Figure 1 on page 9](#) e\*Way 2 is down. Using the MIB, the Control Broker uses the variables found in `stc_mib.txt` to send a message (via the SNMP Agent) regarding the current state of the system. In this example, `cbCurState (1)` defines the Control Broker status as up (the variable “1” represents up), and `dCurState (2)` defines the current state of the data process (e\*Way 2) as down (the variable “2” represents down).

**Note:** `cbCurState` and `dCurState` are read only and cannot be used to change the state of the Control Broker or the data process.

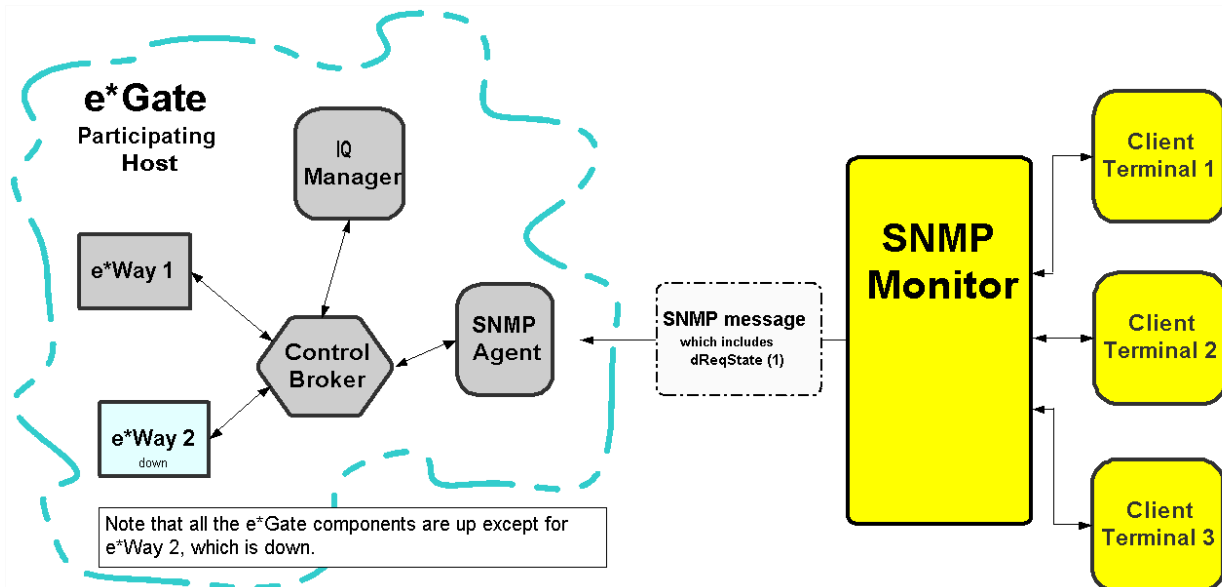


Figure 1 Control Broker Sends Message Regarding Down e\*Way



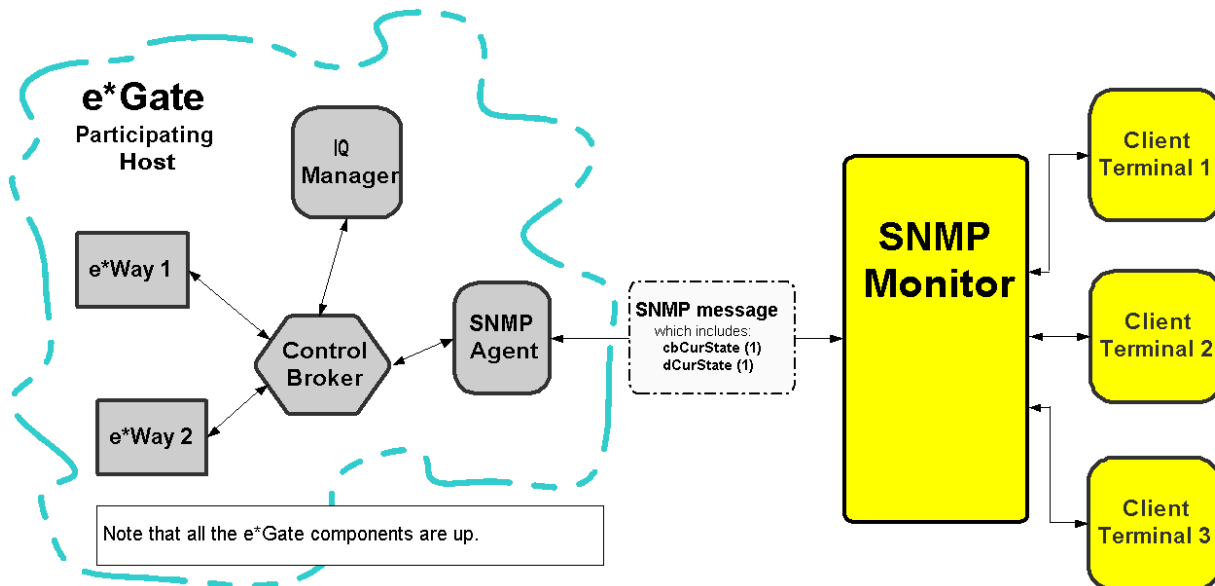
For the system to function properly, e\*Way 2 must be up. In Figure 2 the SNMP Monitor uses the variables found in `stc_mib.txt` to send a message with a request to change the state of the data process; changing the current state of e\*Way 2 from down to up. In this example, `dReqState (1)` requests that the state of e\*Way 2 be changed from down to up (the variable "1" represents up). This is accomplished with the SET command.

Figure 2 SNMP Monitor Sends Message Requesting Change to Down e\*Way



In Figure 3, e\*Way 2 is up and the system functions without any problems. Messages pass back and forth between the Control Broker (via the SNMP Agent) and the SNMP Monitor showing the status for the Control Broker and the data process. In this example, **cbCurState (1)** defines the Control Broker status as up and **dCurState (1)** defines the current state of the data process (which includes e\*Way 2) as up.

**Figure 3** Example of the System with All Components Running



## 2.2 System Requirements

The e\*Gate SNMP Agent is available on the following operating systems:

- Windows 2000, Windows 2000 SP1, and Windows 2000 SP2
- Windows NT 4 SP6a
- Windows NT 4 SP6a (Japanese)
- Solaris 2.6, 7, and 8
- Solaris 2.6, 7, and 8 (Japanese)
- AIX 4.3.3, 5.1
- HP-UX 11.0 and HP-UX 11i
- HP-UX 11.0 (Japanese)
- Compaq Tru64 UNIX V4.0F and V5.0A

To use the e\*Gate SNMP Agent, you need the following:

- An e\*Gate Participating Host, version 4.5 or later.

- The e\*Gate Registry (which need not be located on the same host as the machine where the SNMP Agent is installed).
- A TCP/IP network connection.
- At least 16 MB of free disk space and at least 8 MB of memory.

# Installation

This chapter describes how to install the e\*Gate SNMP Agent.

---

## 3.1 Pre-Installation

Before running the setup program, do the following:

- Close all Windows applications, including any anti-virus applications.
- Stop all e\*Gate services.

**Note:** *If you are installing the SNMP Agent on a system that hosts other e\*Gate components, install the e\*Gate Registry and Participating Host under a non-root account.*

### 3.1.1 After Installation

You will need to create an SNMP component in your schema and modify the Notification Routing script; see [Table 3 on page 20](#). If you need to change configuration parameters later, you must modify the configuration file created during installation, the properties of the SNMP component, and the Notification Routing script.

---

## 3.2 Windows 2000 and Windows NT

### 3.2.1 Windows Files

Table 1 lists the files that are installed on a Windows system.

**Table 1** Files Installed on a Windows System

Installation Directory	File
\bin\	stcsnmpa.exe
\bin\	stc_common.dll, javai.dll
\bin\	stc_mib.txt

**Table 1** Files Installed on a Windows System

Installation Directory	File
\bin\	stcsnmpa.conf (See the NOTE, below).
\classes\	egate.jar, stcjcs.jar
NOTE: To update the <b>stcsnmpa.conf</b> file on a previously installed SNMP Agent, see <b>“Encrypting a Password on an Existing SNMP Agent”</b> on page 34.	

### 3.2.2 Installation Procedure

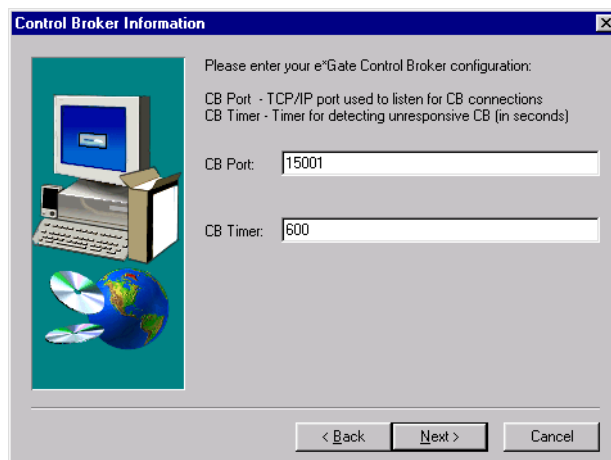
#### To install the Agent on a Windows 2000 or Windows NT system

The e\*Gate SNMP Agent is installed as an e\*Gate add-on.

- 1 To access the e\*Gate SNMP Agent installation, follow the instructions for installing add-ons in the *e\*Gate Integrator Installation Guide*.
- 2 From the list of available Agent add-ons, select **SNMP Agent**, and then click **Next** to begin the installation.
- 3 Follow the on-screen instructions.

After the files are copied, you are presented with additional screens to configure the SNMP Agent’s properties.

**Figure 4** Control Broker Information



- 4 In the **Control Broker Information** dialog box, enter the following information:

- CB Port** The TCP/IP port that the e\*Gate SNMP Agent uses to listen for Control Broker connections. This port must be a unique number not used by any other service. The default is 15001.
- CB Timer** The amount of time after which the e\*Gate SNMP Agent will consider a Control Broker unresponsive, if no interaction has been made during that time. The default is 600 seconds.

**Note:** The **CB Port** information must match the **SNMP Agent** component properties and the **SNMP** section of the **Notification Routing** script. For information on configuring the **SNMP Agent** component, see **Chapter 4**. For information on modifying the **Notification Routing** script, see the *e\*Gate Integrator Alert and Log File Reference Guide*.

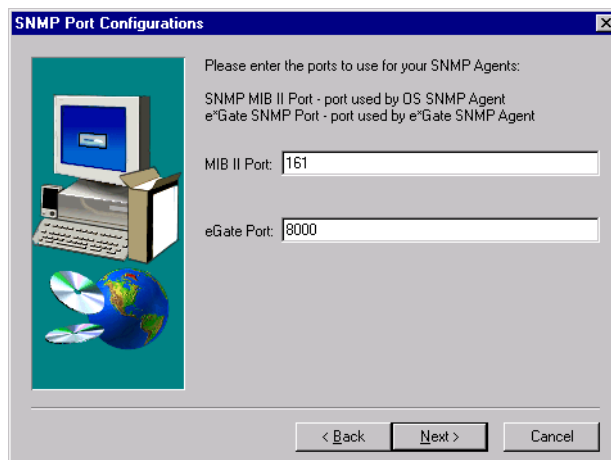
**Figure 5** Administrator Account Information



- 5 In the **Administrator Account Information** dialog box, enter the password for Administrator.

The password must be entered twice, and the entries must be identical.

**Figure 6** SNMP Port Configuration



- 6 In the **SNMP Port Configuration** dialog box, enter the following information:

**MIB II Port** The port the OS SNMP service uses to listen for requests. The e\*Gate SNMP Agent forwards all requests under the MIB II subtree to the OS SNMP service through this port. Default: 161. **Unless you have specifically changed this port number in your operating system, you should not alter the default value.**

**e\*Gate Port** The port the e\*Gate SNMP Agent will use to listen for requests from the SNMP manager. This port must be a unique number not used by any other service. Default: 8000.

7 Follow the on-screen instructions to complete the rest of the installation.

Once installed, the e\*Gate SNMP Agent runs as a standard Windows 2000 or Windows NT Service, and Control Brokers automatically connect to it through the port you specified. To monitor the status of the e\*Gate SNMP Agent, use the Task Manager and the **Services** option in the Control Panel.

## 3.3 UNIX

### 3.3.1 UNIX Files

Table 2 lists the files that are installed on a UNIX system.

**Table 2** Files Installed on a UNIX System

Installation Directory	File
/bin/	stc_mib.txt
/bin/	stcsnmp.conf
/classes/	egate.jar, stcjcs.jar
/etc/rc3.d/ (For Solaris only)	S99stcsnmpdx
/sbin/rc3.d/ (For HP only)	S99stcsnmpdx
/etc/ (For AIX only)	S99stcsnmpdx

### 3.3.2 Installation Procedure

You must be logged on as the **root** user to install the e\*Gate SNMP Agent.

To install the Agent on a UNIX system

- 1 The e\*Gate SNMP Agent is installed as an e\*Gate add-on. To access the e\*Gate SNMP Agent installation, follow the instructions for installing add-ons in the *e\*Gate Integrator Installation Guide*.
- 2 From the list of available add-ons, select **Agents**. Press **Enter** to begin installation and then follow the on-screen instructions.  
After the files are copied, enter the information to configure the SNMP Agent's properties.
- 3 Copy the **egate.jar** and the **stcjcs.jar** files from the **egate\client\classes** folder on an e\*Gate GUI Host (Windows) machine to the same location on your Participating Host machine (UNIX).

- 4 Enter the port number for the OS SNMP service. The e\*Gate SNMP Agent will forward requests to the OS SNMP service through this port. The default is 161. **Unless you have specifically changed this port number in your operating system you should not alter the default value.**

```
Please enter your SNMP MIB II port. This is the port your OS SNMP
agent will listen to for incoming requests.
SNMP II MIB Port [161]:
```

- 5 Enter the port number the e\*Gate SNMP Agent will use to listen for requests from the SNMP manager. This port must be a unique number not used by any other service. The default is 8000.

```
Please enter your e*Gate SNMP port. This is the port used by the
e*Gate SNMP agent.
e*Gate SNMP Port [8000]:
```

The software checks this port to determine if it is available. If the port is in use, you are prompted for another port number. The installation continues after an available port is located.

**Important:** *If you use NIS you must register the e\*Gate SNMP port.*

- 6 Enter the port number the e\*Gate SNMP Agent will use to listen for Control Broker connections. The default is 15001.

```
Please enter the TCP/IP port used to listen for Control Broker
connections
CB Port [15001]:
```

**Note:** *The **CB Port** information must match the SNMP component properties and the SNMP section of the Notification Routing script. For information on configuring the SNMP component, see [Chapter 4](#). For information on modifying the Notification Routing script, see the *e\*Gate Integrator Alert and Log File Reference Guide*.*

- 7 Enter the timer the e\*Gate SNMP Agent will use for detecting an unresponsive Control Broker, if no connection has been made during that time.

```
Please enter the timer for detecting unresponsive Control Broker
(in seconds)
CB Timer [600]:
```

- 8 Enter the location of the Java Runtime Environment (JRE).

```
Please enter the location of your Java Runtime Environment (JRE)
JRE Home directory [/opt/java/jre]:
```

- 9 Follow the on-screen instructions to complete the rest of the installation.

Once installed, the e\*Gate SNMP Agent runs as a daemon. Control Brokers automatically connect to it through the port you specified.



# SNMP Agent Component Configuration

This chapter describes how to configure an e\*Gate SNMP Agent component.


---

## 4.1 Creating an SNMP Agent Component

For the Control Brokers in a schema to connect to the e\*Gate SNMP Agent, you need to create an SNMP Agent component in the schema to be monitored. Although you create this component using the e\*Gate Enterprise Manager, you can put it on a remote host that has never installed the e\*Gate GUI. However, the SNMP Agent component must be in the same schema as the resources it monitors.

### To create an SNMP Agent component

**Note:** *You can install an SNMP Agent to a remote host, even one that does not have e\*Gate, simply by adding a Participating Host to the current schema and then adding an SNMP Agent to it. All components under all hosts of the same schema are monitored by the SNMP Agent.*

- 1 Log into the e\*Gate Enterprise Manager and open the schema you want to monitor.
- 2 Select the **Components** tab.
- 3 In the Navigator (left pane), open the Participating Hosts folder and either select an existing Participating Host or else create a new one.
- 4 On the Palette, click  (Create a new SNMP Agent).
- 5 In the **New Monitor SNMP Agent Component** dialog box, enter the name of the SNMP Agent component, and then click **OK**.

The SNMP Agent is now ready to be configured so that it can monitor all Participating Hosts in the schema.

**Note:** *Each schema can contain at most one SNMP Agent component.*

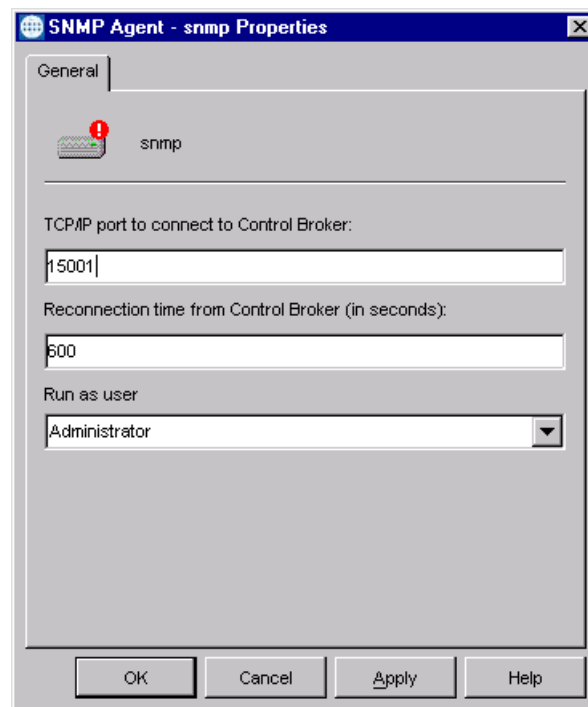
## 4.2 Modifying SNMP Agent Component Properties

Once you have created an SNMP Agent component you must modify its properties to match the configuration information in the e\*Gate SNMP Agent configuration file **stcsnmpa.conf**. For the location of this file, see [Table 1 on page 12](#) for files installed on a Windows system or [Table 2 on page 15](#) for files installed on a UNIX system.

To modify SNMP Agent component properties

- 1 Log into the e\*Gate Enterprise Manager and open the schema you want to monitor.
- 2 In the **Components** tab, use the Navigator to open the Participating Hosts folder and select the host on which you installed the e\*Gate SNMP Agent.
- 3 In the Editor (right pane), double-click the SNMP Agent component. The **SNMP Agent <agent name> Properties** dialog box opens.

**Figure 7** SNMP Agent Properties Dialog Box



- 4 In the **SNMP Agent <agent name> Properties** dialog box, enter the information that appears in the configuration file (**stcsnmpa.conf**) that was created when you installed the e\*Gate SNMP Agent. (For more information on the configuration file, see [“Modifying the Configuration File” on page 21.](#))
  - A In the **TCP/IP port to connect to Control Broker** box, enter The TCP/IP port where the SNMP Agent listens for Control Broker connections. In the **stcsnmpa.conf** file, this information is found under **CB\_PORT**.



- B In the **Reconnection time from Control Broker** (in seconds) box, type the amount of time after which the SNMP Agent will consider the Control Broker unresponsive if there has been no interaction during that time. The default is 600 (ten minutes). In the `stcsnmpa.conf` file, this information is found under **CB\_TIMER**.
  - C In the **Run as user** list, select the user under whose ID this Agent should run. Users are defined in the Enterprise Manager Users folder, not by the operating system. In the `stcsnmpa.conf` file, this information is found under **USERNAME**.
- 5 Click **OK**. The SNMP Agent component for the schema is configured.

### 4.2.1 Configuring a Single Master SNMP Agent

You can optionally use one SNMP agent to monitor all e\*Gate components—that is, all components created under a Control Broker—in all schema.

To configure a master SNMP Agent to monitor all components

*Note:* Before you begin, decide on a single set of component properties you want for the master SNMP Agent.

- 1 In the e\*Gate Enterprise Manager, click  (or, on the **File** menu, click **Open Schema**), to open a schema.
- 2 In the **Open Schema on Registry Host** window, select a schema and click **Open**.
- 3 In the **Components** tab, use the Navigator (left pane) to open the Participating Hosts folder.
- 4 If the host name where the SNMP Agent resides is not already a Participating Host for this schema, on the palette, click  (or, on the **File** menu, point to **New** and then select **Participating Host**) and add the Agent's host as a new Participating Host.
- 5 Using the steps in [“To modify SNMP Agent component properties” on page 18](#), configure the component properties to match the properties you decided on before you began this procedure, and then click **OK**.
- 6 Repeat steps 1 through 5 for all schema.

### 4.2.2 Java method: eventSend()

Information on the Java method `eventSend()` is now located in the *e\*Gate Integrator User's Guide*.

# Modifying the Configuration

This chapter describes how to configure all aspects of the e\*Gate SNMP Agent.

## 5.1 Matching Parameters

The e\*Gate SNMP Agent contains parameters which you must configure in multiple places in order for the agent to function properly. These parameters are:

- **CB Port**—The TCP/IP port where the Agent listens for Control Broker connections.
- **CB Timer**—The timer the Agent uses to detect an unresponsive Control Broker.
- **Username**—The username under which the e\*Gate SNMP Agent will run.

**Important:** Use the password you created for each user when you installed the e\*Gate system. If you do not use the same password, the SNMP Agent will not work.

The areas where these parameters must match are:

- Installation (see [Chapter 3](#)).
- The configuration file `stcsnmpa.conf`, created during installation (discussed below).
- The SNMP Agent component in the schema (see [Chapter 4](#)).
- The SNMP section in the Notification Routing script (see the *e\*Gate Integrator Alert and Log File Reference Guide*).

Table 3 lists the four parallel fields for these three parameters.

**Table 3** e\*Gate SNMP Agent Configuration Parameters

Installation	Configuration file	SNMP Component	Notification Routing script
CB Port	CB_PORT	The TCP/IP port where the SNMP Agent listens for Control Broker connections.	SNMP Agent Port Number
CB Timer	CB_TIMER	Timer for Detecting Unresponsive Control Broker (in seconds)	N/A

**Table 3** e\*Gate SNMP Agent Configuration Parameters (Continued)

Installation	Configuration file	SNMP Component	Notification Routing script
Username	USERNAME	Run as user	N/A

## 5.2 Modifying the Configuration File

The configuration file `stcsnmpa.conf` is created during the installation process. If you need to change the agent's configuration after installation you must modify this file directly.

**Note:** You must restart the e\*Gate SNMP Agent in order for changes to the configuration file to take effect.

To modify the configuration file

- 1 Shut down the e\*Gate SNMP Agent.
- 2 Open the file `\bin\stcsnmpa.conf` in any text editor.
- 3 Make the necessary changes and save the file.
- 4 Restart the e\*Gate SNMP Agent.

### 5.2.1 Configuration Parameters

The configuration file consists of the parameters described in Table 4.

**Table 4** Configuration parameters

Parameter	Possible Values	Description
DEBUG	True or False	Turns debugging on or off. To enable debugging, set to <b>true</b> . To disable debugging, set to <b>false</b> .
USERNAME	Any valid e*Gate username	The username under which the agent will run. <b>(This name MUST match the username entered for the SNMP Agent component in the e*Gate Enterprise Manager.)</b>
PASSWORD	A valid password	The password associated with the username.
CB_PORT	Any available port	The TCP/IP port where the SNMP Agent listens for Control Broker connections. <b>(This value MUST match the port entered for the SNMP Agent component in the e*Gate Enterprise Manager.)</b>

**Table 4** Configuration parameters (Continued)

Parameter	Possible Values	Description
CB_TIMER	Any interval	The number of seconds the SNMP Agent waits between notifications before assuming the Control Broker is unresponsive. When this time expires, the SNMP Agent generates a trap and sends it to the SNMP manager. <b>(This value MUST match the reconnection time entered for the SNMP Agent component in the e*Gate Enterprise Manager.)</b>
SNMP_PORT	Any available port	The port the e*Gate SNMP Agent will use to listen for requests from the SNMP manager.
SNMP_MIBII_PORT	Any available port (usually 161)	The port the OS SNMP service uses to listen for requests. The e*Gate SNMP Agent forwards all requests under the MIB II subtree to the OS SNMP service through this port.
SEPARATOR	 (the "pipe" character)	Token used to separate trap comments.

## 5.3 Sample Configuration File

A sample configuration file is reproduced below. The values in your file will vary depending on the configuration for your system.

```

DEBUG false
USERNAME Administrator
PASSWORD STC
CB_PORT 15001
CB_TIMER 600
SNMP_PORT 8000
SNMP_MIBII_PORT 161
SEPARATOR |

```

# Troubleshooting

Follow the procedure below if the e\*Gate SNMP Agent does not appear to function correctly.

- Turn on debugging. For information on debugging, see [Configuration Parameters](#) on page 21.
- Verify that e\*Gate is sending general notifications by checking whether the e\*Gate Monitor is receiving notifications. In order for the e\*Gate Monitor to receive notifications, the Control Broker must be configured to forward all notifications to the SNMP Agent. See the *e\*Gate Integrator System Administration and Operations Guide* for information on configuring the Control Broker.
- Check to see if the Control Broker log file is sending out notifications.
- If notifications are being sent to the e\*Gate SNMP Agent, check the Control Broker log for a “connected to SNMP Agent” message.
- Check the SNMP Agent log file to verify it has received notifications.
- Check to see if there is a trap for each notification in the MIB file.
- In the SNMP Trap, check for 1351 in OID. The OID is unique for SeeBeyond raised SNMP traps.
- Verify that the e\*Gate SNMP Agent is configured to forward the trap to the right Manager Station.
- Verify that the Manager Station expects the Community Name of the e\*Gate SNMP Agent Station.
- Verify that the username/password combination is correct.

# MIB File

The e\*Gate SNMP Agent communicates with the SNMP Manager through the SeeBeyond Management Information Base (MIB). The SeeBeyond MIB file `stc_mib.txt` contains the definitions of all resources and attributes available for monitoring by the e\*Gate SNMP Agent.

***Important: DO NOT MODIFY THIS FILE.***

---

## A.1 Description

The MIB table is described in two parts:

- [Table 5 on page 25](#) describes objects
- [Table 6 on page 30](#) describes the stcTraps

The location field starts one node below "egate." The complete path is as follows:

```
iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).stc(1351).  
products(1).software(1).egate(1).<append location from table below>.
```

Certain objects can be duplicated because they form part of a group of objects associated with a certain module. For example, if there is more than one Control Broker in a schema there will be more than one cbEntry node and more than one set of CB objects. The first CB will be assigned the location 1.1.1.x where x is a CB object (such as cbState). The second CB will be assigned the location 1.1.2.x, and so forth.



## A.1.1 MIB Objects

**Table 5** MIB Objects

Object	Location	Possible Values	Access	Description
Cbs	1	N/A	not accessible	The Control Broker (CB) tree.
CbTable	1.1	N/A	not accessible	A list of CB entries.
cbEntry	1.1.1	N/A	not accessible	A CB entry containing objects for a particular CB.
cbIndex	1.1.1.1	Integer	not accessible	This is a CB index.
cbName	1.1.1.2	String (0-255)	read-only	The logical name for this CB. The logical name is unique per host.
cbCurState	1.1.1.3	0=unknown 1=up 2=down	read-only	This is the state of the CB.
cbReqState	1.1.1.4	2=down	read/write	Allows you to issue a SET command that will shut down the CB. Note that you can only perform a GET operation after the parameter is SET.
cbHostName	1.1.1.5	String (0-255)	read-only	The host name for this CB.
cbLastUpdateTime	1.1.1.6	TimeTicks	read-only	The last update time for this CB.
cbStartupTime	1.1.1.7	TimeTicks	read-only	The startup time for this CB.
cbSharedDataDir	1.1.1.8	String (0-255)	read-only	The shared data directory for this CB.
cbControlPort	1.1.1.9	Integer	read-only	The CB's control port.
cbNumDRows	1.1.1.10	Integer	read-only	The number of rows of all data processes.
cbNumQRows	1.1.1.11	Integer	read-only	The number of rows of all the queues.
dTable	1.2	N/A	not accessible	A list of data processes in a particular CB. Data processes include server module, communication client, communication client proxy, e*Ways, and external processes. The cbIndex is used to identify a particular e*Gate environment.

Table 5 MIB Objects

Object	Location	Possible Values	Access	Description
dEntry	1.2.1	N/A	not accessible	A data process entry containing objects for a particular data process.
dIndex	1.2.1.1	Integer	not accessible	This is a SeeBeyond data process index.
dName	1.2.1.2	String (0-255)	read-only	Logical name of a SeeBeyond data process. This name is unique within a CB environment.
dType	1.2.1.3	0=Unidentified Element 1=BOB 2=e*Way 3=IQMgr	read-only	This object is used to differentiate between the various types of SeeBeyond data processes.
dCurState	1.2.1.4	0=Unknown 1=up 2=down 3=suspended 4=reloaded 5=removed	read-only	This is the current state of the data process.
dReqState	1.2.1.5	1=up 2=down 3=suspended 4=reloaded	read/write	This object is used to request a change of state for a data process. It is mainly used by the SET command to change the state of an existing data process. It is not meant to allow the SNMP manager to create new data processes. Note that you can only perform a GET operation after the parameter is SET.
dHostname	1.2.1.6	String (0-255)	read-only	The host name for this data module.
dLastUpdateTime	1.2.1.7	TimeTicks	read-only	The last update time of this module.
dStartupTime	1.2.1.8	TimeTicks	read-only	The startup time of this module.
dSharedDataDir	1.2.1.9	String (0-255)	read-only	The shared data directory for this module.
dControlPort	1.2.1.10	Integer	read-only	The CB control port for this data process.
dLastActionTime	1.2.1.11	TimeTicks	read-only	The last action time of this data process. <i>Reserved for future use.</i>

Table 5 MIB Objects

Object	Location	Possible Values	Access	Description
dLastActionName	1.2.1.12	String (0-255)	read-only	The last action name of this data process. <i>Reserved for future use.</i>
dLastPutTime	1.2.1.13	TimeTicks	read-only	Time of the last put operation. <i>Reserved for future use.</i>
dLastPutMsgType	1.2.1.14	TimeTicks	read-only	Time of the last put Event type. <i>Reserved for future use.</i>
dLastPutSeqNum	1.2.1.15	TimeTicks	read-only	Time of the last put sequence number. <i>Reserved for future use.</i>
dLastGetTime	1.2.1.16	TimeTicks	read-only	Time of the last get operation. <i>Reserved for future use.</i>
dLastGetMsgType	1.2.1.17	TimeTicks	read-only	Time of the last get Event type. <i>Reserved for future use.</i>
dLastGetSeqNum	1.2.1.18	TimeTicks	read-only	Time of the last get sequence number. <i>Reserved for future use.</i>
dLastMsgReceivedFromExtTime	1.2.1.19	TimeTicks	read-only	Time of the last Event received from the external. <i>Reserved for future use.</i>
dLastMsgSentFromExtTime	1.2.1.20	TimeTicks	read-only	Time of the last Event sent from the external. <i>Reserved for future use.</i>
dExtSeqNum	1.2.1.21	Integer	read-only	External sequence number. <i>Reserved for future use.</i>
dExtRetryCount	1.2.1.22	Integer	read-only	External retry count. <i>Reserved for future use.</i>
dExtState	1.2.1.23	0=Unknown 1=Communicating 2=Not Communicating	read-only	This is the state of the external process to which this data process connects. <i>Reserved for future use.</i>
qTable	1.3	N/A	not accessible	A list of queue entries in a particular CB.
qEntry	1.3.1	N/A	not accessible	A queue entry containing objects for a particular queue.
qIndex	1.3.1.1	Integer	not accessible	This is e*Gate's queue index.

Table 5 MIB Objects

Object	Location	Possible Values	Access	Description
qName	1.3.1.2	String (0-255)	read-only	The logical name of the queue. This name is unique per CB.
qCurState	1.3.1.3	0=Unknown 1=Attached 2=Detached	read-only	This is the current state of the queue.
qReqState	1.3.1.4	1=Attached 2=Detached	read/write	Allows you to issue a SET command that will alter the queue's state. Note that you can only perform a GET operation after the parameter is SET.
qHostname	1.3.1.5	String (0-255)	read-only	The host name for this queue.
qLastUpdateTime	1.3.1.6	TimeTicks	read-only	The last update time of this queue.
qStartupTime	1.3.1.7	TimeTicks	read-only	The startup time of this queue.
qNumOfRevealedMsgs	1.3.1.8	String (0-255)	read-only	The number of revealed Events for this queue. <i>Reserved for future use.</i>
qNumOfUnRevealedMsgs	1.3.1.9	Integer	read-only	The number of unrevealed Events for this queue. <i>Reserved for future use.</i>
qNumOfExpiredMsgs	1.3.1.10	Integer	read-only	The number of expired Events for this queue. <i>Reserved for future use.</i>
qNumOfJournalledMsgs	1.3.1.11	Integer	read-only	The number of journalled Events for this queue. <i>Reserved for future use.</i>
qLastTimePublished	1.3.1.12	TimeTicks	read-only	Time of the last Event published to this queue. <i>Reserved for future use.</i>
qSizeInBytes	1.3.1.13	Integer	read-only	The size of this queue. <i>Reserved for future use.</i>
qSizeMsgCount	1.3.1.14	Integer	read-only	The size of the Event count. <i>Reserved for future use.</i>
stcReleaseVer	2	String (0-255)	read-only	Version number of the MIB table (the default value is: "version 4.0").

**Table 5** MIB Objects

Object	Location	Possible Values	Access	Description
stcTraps	3	N/A	not accessible	Object identifier. The trap entries are shown on a separate table below.
moduleName	4	String (0-255)	not accessible	This object is used for a trap that has a monitor name or module name when the e*Gate SNMP Agent does not have that information.
genericComments	5	String (0-255)	not accessible	This object is used along with most of the traps as a reason for trap generation. The content will be changed as a new trap is generated, it has no significant meaning by itself.

## A.1.2 stcTraps

Table 6 MIB Table (stcTraps)

Trap Name	Location	Variables	Description
SMNPALossConnectionToCb	3.1	cbName	The e*Gate SNMP Agent detects a lost connection to a Control Broker (CB).
SMNPADetectsCbUnresponsive	3.2	cbName	The e*Gate SNMP Agent detects an unresponsive CB.
SNMPADetectsCbResponded	3.3	cbName	The e*Gate SNMP Agent detects that a CB responded after it had been unresponsive.
cbDetectsSNMPAResponded	3.4	cbName	A CB detects that an element responded.
cbConnectedToSNMPA	3.5	cbName	A CB connects to the e*Gate SNMP Agent.
cbDetectsDiskUsageAbove Threshold	3.6	cbName, genericComments	A CB detects an out-of-disk space problem. Disk sector and other useful information are concatenated in the genericComments string.
timerEvent	3.7	cbName, genericComments	A timer event in a CB is triggered to signal the CB to perform certain tasks. A genericComments string will contain the notification name and other information.
cbCantGetStatus	3.8	cbName, moduleName, genericComments	A CB cannot get status from a module.
cbCantStartModule	3.9	cbName, moduleName, genericComments	A CB cannot start a module.
cbLostConnectionToM	3.10	cbName, moduleName, genericComments	A CB lost its connection to the monitor.
cbDetectsMUnresponsive	3.11	cbName, moduleName, genericComments	A CB detects an unresponsive monitor element.
cbDetectsMResponded	3.12	cbName, moduleName, genericComments	A CB detects that a monitor element responded.
monitorConnectedToCb	3.13	cbName, moduleName, genericComments	A monitor connected to the CB. <i>(Not implemented.)</i>
userAuthenticationFailure	3.14	cbName, moduleName, genericComments	A CB failed to authenticate a module.

**Table 6** MIB Table (stcTraps)

Trap Name	Location	Variables	Description
alertDeliveryFailure	3.15	cbName, moduleName, genericComments	A CB cannot send an Alert to a certain monitor agent(s) or an agent cannot deliver the Alert. The list of alert agent(s) will be concatenated with the reason in the genericComments string.
dInputAboveThreshold	3.16	dName, genericComments	Data process input coming into an element above the threshold.
dInputBelowThreshold	3.17	dName, genericComments	Data process input coming into an element below the threshold.
dOutputAboveThreshold	3.18	dName, genericComments	Data process output from an element above the threshold.
dOutputBelowThreshold	3.19	dName, genericComments	Data process output from an element below the threshold.
dDownFatal	3.20	dName, genericComments	A data process is down for an unknown reason (for example, data process crashed or was shut down by a KILL command).
dDownControlled	3.21	dName, genericComments	A data process is shut down due to lack of memory or decided to go away.
dDownUser	3.22	dName, genericComments	A data process is shut down gracefully and status is sent to a CB.
dUp	3.23	dName	A data process is up and status is sent to a CB.
cbDetectsDUnreponsive	3.24	dName, genericComments	A data process does not respond to a CB.
cbDetectsDReponded	3.25	dName, genericComments	A CB detects a module responded after being unresponsive.
dCantConnectToExternal	3.26	dName, genericComments	A data process cannot connect to an external system. (Not implemented.)
dConnectedToExternal	3.27	dName, genericComments	A data process has connected to an external system. (Not implemented.)
dLostConnectionExternal	3.28	dName, genericComments	A data process lost connection to an external system.

**Table 6** MIB Table (stcTraps)

Trap Name	Location	Variables	Description
dNotConnectedToExternal	3.29	dName, genericComments	A data process is not connected to an external system.
dDeliveryFailure	3.30	dName, genericComments	Data process got a failure from external procedure.
unUsableMessageCantId	3.31	dName, genericComments	This notification is generated when a message fails all ID tests.
messageContentOfInterest	3.32	dName, genericComments	A message of the element matches the requirement for which the element is looking.
unQueueableMessage	3.33	dName, genericComments	A message is unqueueable.
userDefined	3.34	dName, genericComments	An Event defined by a user for a certain data process.
queueMessageExpired	3.35	qName	A message has expired.
iqLimitExceeded	3.36	qName, genericComments	A queue has exceeded its limit.
CBConnectedToRegistry	3.37	cbName	The CB has connected to the e*Gate Registry. <i>(Not implemented.)</i>
CBLostConnectionToRegistry	3.38	cbName	The CB has lost its connection to the e*Gate Registry.





# Password Encryption and the Configuration File

Password encryption is handled during the installation of the SNMP Agent (see the Windows [“Installation Procedure” on page 13](#)). However, if the SNMP Agent (version 4.5.2) was already installed without password encryption, the configuration file `stcsnmpa.conf` must be updated with an encrypted password.

---

## B.1 Encrypting a Password on an Existing SNMP Agent

There is a utility in `stcutil.jar` that allows you to encrypt the SNMP Agent password without reinstalling the SNMP Agent. The following is only an example command that you would use to encrypt your password on an existing SNMP Agent:

- 1 At the command prompt, type:

```
java -classpath c:\egate\client\classes\stcutil.jar
com.stc.eways.util.scDecrypterEncrypter encrypt myusername
mypassword
```

The example encrypts the password `mypassword` using the key `myusername` and prints out the encrypted password. Also note that your directory and path may differ from the directory and path given in this example.

- 2 After the password has been encrypted, you must update the configuration file. Specify the encrypted password as the value of `PASSWORD` in the `stcsnmpa.conf` file.

**Note:** Only users who have installed earlier versions of the SNMP Agent, have existing `stcsnmpa.conf` files already in place, and upgraded the SNMP Agent without reinstalling the product need to perform this procedure.

# Index

## A

Administrator Account Information 14

## C

CB Port 13, 16, 18, 20

CB Timer 13, 16, 19

configuration

Control Broker 13

e\*Gate port 14

Master SNMP Agent 19

MIB II port 14

SNMP Agent 18, 19

Control Broker configuration 13

conventions, writing in document 6

## E

e\*Gate

monitored by MIB 8

e\*Gate Port 15

eventSend()

location of information about 19

## I

installation

on UNIX systems 15, 16

on Windows systems 13, 15

installation files

UNIX 15

Windows 2000 and Windows NT 12, 15

## M

Master SNMP Agent 19

MIB

example of use 8

using 8

MIB file 30

MIB II Port 14, 15

MIB objects 25, 29

modifying an SNMP Agent component 18

modifying the configuration file 21, 22

## N

NIS 16

Notification Routing script 8, 20

## O

organization of information, document 5

## P

Parameters 20

PASSWORD

updating stcsnmpa.conf 34

updating value in stcsnmpa.conf 34

password encryption 13, 34

port, TCP/IP 13

properties, SNMP Agent component 18

## S

SeeBeyond SNMP Port 16

SeeBeyond Web site

technical support 7

SNMP Agent

component 19

component properties 18

SNMP Port Configuration 14

stc\_mib.txt 30

stcsnmpa.conf 18, 20, 21, 22

updating PASSWORD value 34

stcTraps 30, 32

system requirements 10

## T

TCP/IP port 13

technical support 7

timer 13

Troubleshooting 23

## U

UNIX installation 15, 16

UNIX installation files 15

Username 19

## W

Windows 2000 and Windows NT

installation 13

installation files 12, 15

Windows NT and Windows 2000

installation 15