Sentinel SuperPro 6.3 Documentation Addendum

RAINBOW
T E C H N O L O G I E S

CONFIDENTIAL INFORMATION

The Sentinel software protection system is designed to protect your software products from unauthorized use. The less information that unauthorized people have regarding your security system, the greater your protection. It is in your best interest to protect the information herein from access by unauthorized individuals.

Part Number 007-0013-001, Revision C
Sentinel SuperPro releases 6.3 and later

Revision	Action/Change	Date
C	Added details about SuperPro for Macintosh.	February 2003

Rainbow Technologies Sales Offices

United States: *http://www.rainbow.com,* Tel: +1 949 450 7300 / Toll Free: +1 800 852 8569

Australia
Tel: +61 3 9820 8900
http://www.rainbowaustralia.com.au

Germany
Tel: +49 89 32 17 98 0
http://www.de.rainbow.com

Korea
Tel: +82 31 705 8212
http://www.rainbow.com/korea

Brazil
Tel: +55 11 6121 6455
http://www.rainbow.com/brasil

Hong Kong
Tel: +852 3157 7111
http://www.rainbow.com

Mexico
Tel: +52 55 5575 1441
http://www.rainbow.com/latinamerica

China
Tel: +86 10 8266 3936
http://www.isecurity.com.cn

India
Tel: +91 11 26917538
http://www.rainbowindia.co.in

Taiwan
Tel: +886 2 6630 9388
http://www.rainbow.com/taiwan

France
Tel: +33 1 41 43 29 00
http://www.fr.rainbow.com

Japan
Tel: +81 3 5719 2731
http://www.rainbowtech.co.jp

UK
Tel: +44 (0) 1932 579200
http://www.uk.rainbow.com

International Quality Standard Certification

Rainbow Technologies, Inc. Irvine, CA facility has been issued the ISO 9001 Certification, the globally recognized standard for quality, by Det Norske Veritas as of March 2002. Certificate Number CERT-02982-2000-AQ-HOU-RABR2.

Export Considerations

Rainbow offers products that are based on encryption technology. This encryption technology is used solely to prevent unauthorized use of the protected software and for client authentication, and is not used to provide security of end user data.

The Bureau of Export Administration (BXA) in the U.S. Department of Commerce administers the export controls on Rainbow's commercial encryption products. Rules governing exports of encryption can be found in the Export Administration Regulations (EAR), 15 CFR Parts 730-774, which implements the Export Administration Act ("EAA" 50 U.S.C. App. 2401 et seq.).

Important Note: BXA requires that each entity exporting products be familiar with and comply with their obligations described in the Export Administration Regulations. Please note that the regulations are subject to change. We recommend that you obtain your own legal advice when attempting to export any product that uses encryption. In addition, some countries may restrict certain levels of encryption imported into their country. We recommend consulting legal counsel in the appropriate country or the applicable governmental agencies in the particular country.

Contents

Chapter 2 – Sentinel SuperPro 6.3 Client Library

Chapter 3 – Additional Features for Windows

Chapter 4 – Additional Features for Red Hat Linux

Chapter 5 – Additional Features for Macintosh

Chapter 6 – Platform-Independent Field Activation

Appendix A – Documentation Errata

Appendix B – Sentinel SuperPro Error Codes

Appendix C – Frequently Asked Questions

Appendix D – Changes to Sentinel SuperPro 6.2

Index

Preface

About This Addendum

This addendum contains information on the new features added since the Sentinel SuperPro 6.2 release. Because manuals have not yet been updated for the 6.2 and 6.3 releases, this addendum also contains information on features added to the 6.2 release.

We assume the reader is a Sentinel SuperPro developer who is already familiar with the Sentinel SuperPro product. If you are not already familiar with Sentinel SuperPro, please see "Accessing Online and Printed Documentation" on page xi to find out how to learn more about Sentinel SuperPro.

Conventions Used in This Addendum

Please note the following conventions concerning bold lettering, italics, syntax and more:

Convention	Meaning
Select	Use the arrow keys or mouse to select an item on a menu, a field in a window or an item in a list.
Click	Press the primary mouse button once. The primary mouse button is typically the left button, but may be reassigned to the right button.
`Courier`	Denotes syntax, prompts and code examples. If bold, denotes the text you type.

Convention	Meaning
Bold Lettering	In procedures, words in boldface type represent keystrokes, menu items, window names or mouse commands.
Italic Lettering	Words in italics represent file names and directories, or, when used in explanatory text, for emphasis.

How to Get the Most from This Addendum

The *Sentinel SuperPro 6.3 Documentation Addendum* discusses all the changes and new features introduced since the 6.1.1 and 6.2 release. It is intended to be used with the other Sentinel SuperPro user guides—see "Accessing Online and Printed Documentation" on page xi for information on these guides.

The following table explains what you can find in each chapter of this document:

Chapter/Appendix	Description
Chapter 1 – Overview and Installation	A summary of the changes in Sentinel SuperPro 6.3 and 6.2 and information on installation.
Chapter 2 – Sentinel SuperPro 6.3 Client Library	This chapter contains details on changes to the Sentinel SuperPro client library.
Chapter 3 – Additional Features for Windows	This chapter discusses the various features available on the 32-bit Windows platform, including quick automatic protection (Shell).
Chapter 4 – Additional Features for Red Hat Linux	This chapter discusses the various features available on Red Hat Linux.
Chapter 5 – Additional Features for Macintosh	This chapter discusses the various features available on Macintosh.
Chapter 6 – Platform-Independent Field Activation	This chapter discusses the platform-independence of the Field Exchange utility.
Appendix A – Documentation Errata	Corrections to the existing Sentinel SuperPro manuals are discussed in this appendix.
Appendix B – Sentinel SuperPro Error Codes	Description of the SuperPro API status codes and automatic protection (Shell) error codes.

Chapter/Appendix	Description
Appendix C – Frequently Asked Questions	This appendix discusses the most commonly asked questions by SuperPro developers and end users.
Appendix D – Changes to Sentinel SuperPro 6.2	This appendix contains details on features of the Sentinel SuperPro 6.2 release.

Accessing Online and Printed Documentation

Refer to the following documentation for more detailed information and instructions.

Note: *All the guides listed below (except the release notes) were last updated in the 6.1 software release. This addendum contains information on all features added in the subsequent releases. It must be used in addition to the documents listed below.*

Manual	What's in it?	Who should read it?
Sentinel SuperPro Developer's Guide	All the steps necessary to protect, package, and ship applications protected with Sentinel SuperPro, as well as information about the Sentinel SuperPro API calls.	Developers responsible for the overall process of protecting and shipping a Windows application.
Sentinel SuperPro Quick Start Guide for Windows	A quick tour of Sentinel SuperPro for Windows application developers.	Anyone who is new to Sentinel SuperPro or software protection and wants a quick overview of Sentinel SuperPro features.

Manual	What's in it?	Who should read it?
Sentinel SuperPro System Administrator's Guide	Instructions for installing and running the Sentinel SuperPro Server and Monitoring Tool.	End users of your protected application who are responsible for administering the application and end user license management.
Sentinel SuperPro Release Notes	Last-minute information on the current release, along with information on new features.	All Sentinel SuperPro developers.

This guide is also available online as an Adobe Portable Document Format (PDF) file. You must have the Adobe Acrobat Reader version 4 or later to view this online file. The Acrobat Reader installer is also included in the SuperPro CD.

Tip: *Check the Rainbow Technologies Web site (*www.rainbow.com*) for the most up-to-date information about Sentinel software protection products, FAQs and technical notes.*

Getting Help

Rainbow Technologies is committed to supporting Sentinel SuperPro. If you have questions, need additional assistance, or encounter a problem, please contact Rainbow Technologies Technical Support using one of the methods listed in the following table:

Rainbow Technologies Technical Support Contact Information

Rainbow Technologies Customer Connection Center (C^3)	
http://c3.rainbow.com	
Americas	
Internet	http://www.rainbow.com/support
E-mail	techsupport@rainbow.com

Rainbow Technologies Technical Support Contact Information (Continued)

United States	
Telephone	(800) 959-9954
Fax	(949) 450-7450
Europe	
E-mail	EUTechSupport@rainbow.com
France	
Telephone	0825 341000
Fax	44 (0) 1932 570743
Germany	
Telephone	01803 RAINBOW (7246269)
Fax	089 32179850
United Kingdom	
Telephone	0870 7529200
Fax	44 (0) 1932 570743
Pacific Rim	
E-mail	techsupportpacrim@rainbow.com
Australia and New Zealand	
Telephone	(61) 3 9820 8900
Fax	(61) 3 9820 8711
China	
Telephone	(86) 10 8266 3936
Fax	(86) 10 8266 3948
India	
Telephone	(91) 11 2 691 7538
Fax	(91) 11 2 684 9105
Taiwan and Southeast Asia	
Telephone	(886) 2 2570 5522
Fax	(886) 2 2570 1988

We Welcome Your Comments

To help us improve future versions of Sentinel SuperPro documentation, we want to know about any corrections, clarifications or further information you will find useful. When you contact us, please include the following information:

- The title and version of the guide you are referring to

- The version of Sentinel SuperPro you are using

- Your name, company name, job title, phone number and e-mail address

Send us e-mail at:

techpubs@rainbow.com

Or, you can write us at:

Rainbow Technologies, Inc.
50 Technology Drive
Irvine, CA 92618

Attn: Technical Publications Department

Thank you for your feedback. Keep in mind that these e-mail and mail addresses are only for documentation feedback. If you have a technical question, please contact Rainbow Technical Support using the information given in "Getting Help" on page xii.

Chapter 1

Overview and Installation

Sentinel SuperPro version 6.3 combines various enhancements and new features to provide a platform-independent, secure and *easy-to-use* system for both the SuperPro developer and end user. The addendum discusses all the changes done and explains how to use them. We also include a summary of features added for the 6.2 release.

Overview

Sentinel SuperPro pioneered the hardware key-based software protection to secure and license your applications from unauthorized use. With the ASIC-driven SuperPro key, you can enforce licensing and administer your application, ship demo applications that can be converted into full-featured software, and effect license upgrades remotely. The new features of this release add to Sentinel SuperPro's reliability, flexibility, and *ease-of-integration*.

The following sections summarize the new features:

Support for Red Hat Linux

Until now, Sentinel SuperPro offered limited support for the Linux platform, providing just the parallel port driver module and stand-alone client library. This release, on the other hand, provides substantial support including the

6.3 client library, Sentinel USB daemon, SuperPro server, Field Exchange (command-line), Monitoring Tool and Combo Installer.

See Chapter 4, "Additional Features for Red Hat Linux" for specific information.

Note: *Unless stated, all the references in this document are for the Windows version of Sentinel SuperPro.*

Support for Macintosh

This release introduces support for Macintosh platforms. You can secure your stand-alone applications using the SuperPro USB keys.

See Chapter 5, "Additional Features for Macintosh" for specific information.

The SuperPro 6.3 Client Library

This release allows an application to manage access to the SuperPro key attached on a local computer, without the aid of the SuperPro server.

Sentinel SuperPro version 6.3 brings together two different scenarios used with previous releases. In 6.0, the stand-alone client library communicated directly with the SuperPro key attached on the same system through the Sentinel System Driver. The subsequent releases (6.1 and 6.2) introduced the concept of the SuperPro server, which allowed multiple clients to access a SuperPro key across the network. Such development did allow network access, but also forced the stand-alone user to install the SuperPro server for accessing the key attached on the same system.

The 6.3 client library makes the use of the SuperPro server optional for accessing the local key. It offers various alternative access modes for both network and stand-alone communication.

See Chapter 2, "Sentinel SuperPro 6.3 Client Library" for details on the various access modes.

Platform-Independent Field Activation

With the use of identical SAFE objects (USAFE and DSAFE), the field activation process has been made platform independent. As a result, a Windows-based developer/distributor can provide licenses to customers in the field operating on any of the supported platforms. However, the field activation option is not yet available for Macintosh.

See Chapter 6, "Platform-Independent Field Activation" for more information.

Automatic Protection (Shell) Enhancements

The Toolkit's automatic protection (Shell) option has been enhanced with the following new features and changes:

- Quick automatic protection lets you protect your executables quickly and easily. It adds a protective "shell" layer around your application, but gives results almost instantly. Use the quick automatic protection hyperlink on the **Home** screen of the Toolkit to start right away!

- Now, access modes can be set even for applications protected using automatic protection. Earlier, the default access mode had to be used for the application.

- The number of steps for protecting an application using automatic protection have been reduced. The **Prototype Now** button performs the tasks, otherwise done individually in the **Prototype** and **Implementation** stage, in a single click!

- Earlier, the original copy of your unprotected application faced the risk of being overwritten if the two names—unprotected and protected file—were the same. This risk has been ruled out in this release, as the unprotected and protected file names *cannot* be the same.

The topic "Quick Automatic Protection Option in Toolkit" on page 27 illustrates the complete process of protecting an application.

Combo Installer for Your Customers

Combo Installer has been introduced to simplify the deployment of SuperPro components—Sentinel Driver, SuperPro Server and Monitoring Tool—essential for using a networked, protected application.

For Windows, the *three-in-one* installer can either be used directly by your customers or be easily integrated with your application's Windows Installer-based installation program. See the topic "Combo Installer for Windows" on page 34 for more information.

For Linux, *ready-to-use* RPMs have been provided for your use. See "Combo Installer for Linux Redistributables" on page 44 for more information.

Additional Features for Windows

The following miscellaneous changes have also been done since the 6.2 release:

- The Sentinel USB driver is Microsoft Windows Hardware Quality Labs (WHQL) certified for Windows 2000 and XP (32-bit).

- The Sentinel System Driver can now support user-configured PCI cards.

- Command-line options are now available for installing the SuperPro server on Windows NT/2000/XP.

- It is no longer necessary to run the SuperPro server while using Toolkit and stand-alone utilities (Field Exchange, Make Keys and License Generator).

- A utility has been provided that can be used to stop the SuperPro server.

- The SuperPro Toolkit has been enhanced with new features for enriched functionality and convenient navigation.

- Sentinel SuperPro is also supported on Windows XP.

- SuperPro client library interfaces have been provided for the following high-level languages in this release: .NET Object (VB.NET) Visual

Studio 7.0, .NET Object (C#) Visual Studio 7.0, MetroWerks CodeWarrior 3.0, and COM Object VC 6.0.

See "Miscellaneous Changes" on page 35 for details.

Summary of New Features Added in the 6.2 Release

The following new features were presented in the Sentinel SuperPro 6.2 release:

Support for the Novell NetWare Server

Sentinel SuperPro 6.2 introduced support for the Novell NetWare 4.2 and 5.0 servers. The Sentinel SuperPro server NetWare Loadable Module (*spn-nwsrv.nlm*) can be deployed in any Novell server to support clients using IPX/SPX protocol.

See Appendix D, "SuperPro Server on Novell" for details on requirements and configuration instructions.

API Additions to the 6.2 Client Library

In the Sentinel SuperPro 6.2 release, the RNBOsproSetProtocol and the RNBOsproSetHeartBeat APIs were introduced to allow more control over your protected application. The former lets you specify the protocol for communication between the client and SuperPro server, while the latter is for configuring the heartbeat interval of the client to maintain the license obtained.

See Appendix D, "New APIs Added to the 6.2 Client Library" for details.

NetBEUI Protocol Support

Sentinel SuperPro 6.2 provided NetBEUI support to the client library and server. See Appendix D, "Support for NetBEUI Added" for details on starting the server/service in NetBEUI mode.

SuperPro Server Icon Display Made Optional

Developers can now either show or hide the icon that appears in the System Tray when the Sentinel SuperPro server is running on Windows 9x. See Appendix D, "SuperPro Server Icon Display Made Optional" for the required registry settings.

SuperPro Server Messages Display Made Optional

Developers can now choose to hide/display the SuperPro server messages on a Windows 9x system. See Appendix D, "SuperPro Server Messages Suppression Allowed" for more information.

Installation

Installation On Windows

This section contains the system requirements and points you must remember before running the SuperPro setup program on Windows.

System Requirements

The following hardware and software requirements must be met:

Minimum Hardware Requirement

- Pentium microprocessor, P90
- VGA monitor (800 x 600 resolution recommended)
- 250 MB free hard disk space
- CD-ROM drive
- 32 MB RAM

Minimum Software Requirement

- Microsoft Windows NT 4.0 (Service Pack 6), Windows 95*, Windows 98 (Second Edition), Windows ME, Windows 2000 or Windows XP.
- The Sentinel System Driver installation requires Windows Installer version 2.0 or higher on the target system. If a lower version is found on the target system, the installer will inform the user and upgrade automatically.
- You must have Internet Explorer 4.01 or higher installed to be able to view the Sentinel SuperPro online Help file. This file cannot be viewed with any other browser, such as Netscape Communicator.

---

* On Windows 95, the developer can use the Sentinel System Driver 5.41 (parallel port driver only), SuperPro server, and client library, however, no guarantee is provided for other components. Even though a Windows 95 developer cannot use the Toolkit, applications protected on other Windows platforms using the automatic protection feature of the Toolkit can be run on Windows 95.

Important Points for Installation

You must consider the following points before you begin installation:

■ Sentinel SuperPro 6.3.x can co-exist with the previous versions of Sentinel SuperPro. However, the Sentinel System Driver and SuperPro server must be upgraded.

Note: *If you have upgraded from Windows 98 to 2000 or XP, you must first uninstall Sentinel SuperPro using its setup program included on the installation CD before installing Sentinel SuperPro 6.3 (and not through the* **Add/Remove Programs** *option in* **Control Panel***).*

■ Save and exit out of all currently running applications.

■ Remove any USB keys attached to your system before installation.

■ In order to set up Sentinel SuperPro correctly on Windows NT/2000/ XP, you *must* have Administrator privileges.

To install, insert the SuperPro CD in the appropriate drive of your workstation. The autorun feature will launch *start.exe*. If it doesn't then, navigate to the CD-drive and open *start.exe*. Click the SDK link to run the installer.

Tip: *SuperPro setup for Windows installs all the necessary components; you need not separately install the Toolkit, Sentinel System Driver, Interfaces and the SuperPro server.*

Installed Directories

The files and folders listed below will be created at the following path: *C:\Program Files\Rainbow Technologies\SuperPro\6.3* if you selected the **Complete** option (assuming C as your root drive):

Installation Directory Structure

File/Folder Name	Description
\Manuals	Contains documentation in PDF format, including the *Sentinel SuperPro Developer's Guide.*

Installation Directory Structure

File/Folder Name	Description
\Interfaces	Contains the SuperPro client library interfaces, including their documentation
\MSVCDev	Contains the header file, library and a sample program on how to use it
\Server	Contains the SuperPro server
\Monitoring Tool	Contains the SuperPro Monitoring Tool and its Help
\Tools	Contains utilities, including Field Exchange, Make Keys and License Generator
\Safe sample	Contains an example for customizing the SAFE program
\ComboInstaller	Contains the Combo Installer's *setup.exe* and MSI, *plus* MSMs and binaries of the Sentinel System Driver, SuperPro server, and Monitoring Tool.

Installation on Red Hat Linux

System Requirements

The hardware requirements are given in "Minimum Hardware Requirement" on page 7.

Software Requirements

- Red Hat Linux 7.2.0 (kernel version 2.4.7-10 smp/non-smp) or Red Hat Linux 7.3.0 (kernel version 2.4.18-3 smp/non-smp) or Red Hat Linux 8.0 (kernel version 2.4.18-14 smp/non-smp).

- The Red Hat Package Manager (RPM) version must be 4.2 or higher.

- To see the Acrobat files, you need to have Acrobat Reader version 4.0 or higher installed on your computer. Acrobat Reader version 4.05 can be installed from the Sentinel SuperPro CD.

Important Points for Installation

- To install the Sentinel SuperPro SDK, you must upgrade to Sentinel UNIX Driver version 5.50 or higher. For kernel versions other than specified, forced installation of Sentinel UNIX Driver will take place without prompts. However, no guarantee is provided for the proper functioning on those kernel versions.

- In order to set up Sentinel SuperPro, you must be a super user (root).

To install, follow the instructions given below. The software will be installed in the */opt/Rainbow Technologies* directory of your system root:

1. Insert the SuperPro CD in the appropriate drive of your computer.

2. Now, mount the CD using the `mount` command.

3. Change the directory where the installation script *install.sh* exists. Afterward, type **sh install.sh** to start the installation.

4. You will be prompted to specify one of the following options for Sentinel UNIX Driver: **1** (Only parallel driver), or **2** (Only USB daemon), or **3** (Both the parallel driver and USB daemon).

5. Soon after the Sentinel UNIX Driver installation, the SuperPro SDK installation follows. The installer will install all components automatically, and a message is shown after the installation is complete.

Note: *The License Generator utility and Developer's Toolkit are only available on Windows.*

The */opt/Rainbow Technologies* directory will contain the following subdirectories:

Directory	Files
/SUD5.50	Contains files for Sentinel UNIX Driver (parallel driver and/or USB daemon based on the option you chose during installation).
/SuperPro/6.3/Dev	Contains the source file, libraries, and a sample application.
/SuperPro/6.3/Interfaces	Contains the client library interfaces for GCC, Java, and Kylix.
/SuperPro/6.3/Manuals	Contains the Sentinel SuperPro manuals.
/SuperPro/6.3/MonitoringTool	Contains the SuperPro Monitoring Tool and Help.
/SuperPro/6.3/SafeSample	Contains source files for building the USAFE sample.
/SuperPro/6.3/Server	Contains the SuperPro server for Linux.
/SuperPro/6.3/Tools	Contains the Field Exchange and SafeCfg utility.

Installation on Macintosh

The following hardware and software requirements must be met on the developer's system in order to install the SuperPro kit:

- Power Mac G4 System with MacOS X version 10.2.x (kernel version 6.x smp/non-smp)

- 250 MB free hard disk space

- 64 MB RAM

- CD-ROM drive

Note: *In order to install and uninstall Sentinel SuperPro, you must be a super user (root) or administrator.*

To install Sentinel SuperPro, run the installer given on the CD. The Sentinel Mac Driver will be installed at */Library/Extensions,* while the following components will be installed at: */Applications/Rainbow Technologies/SuperPro/ 6.3*:

Path	Description
ILibrary	Contains the SuperPro client library.
IInterfaces/CC	Contains the interface for CC.
IManuals	Contains the Sentinel SuperPro documentation, including the developer's guide.
SSPUninstall.sh	The script to uninstall SuperPro.

Chapter 2

Sentinel SuperPro 6.3 Client Library

This chapter provides information about the various access modes introduced in this release. We also discuss the behavior of the SuperPro keys and migration requirements for the pre-6.1.x users.

The topics are arranged in the following order:

- "W hat is an Access Mode?" on page 13.
- "Highlights of the SuperPro 6.3 Client Library" on page 14.
- "Setting an Access Mode" on page 22.
- "Migration Path for the Pre-6.1.x Users" on page 25
- "Interfaces to the Client Library" on page 25.

What is an Access Mode?

Under the SuperPro software protection scheme, your application is associated with a hardware key to authorize its use. Once a correct key is found by your application, various software locks are performed to acquire, maintain and release the license. However, before all this, your application must be guided to look for the key on the network or a local computer, using a

method that provides the best performance and makes the least demand on network resources.

The 6.3 client library streamlines the path followed by a protected application in search of the key. This is determined by the *access mode* specified in RNBOsproSetContactServer API or the NSP_HOST environment variable. An access mode decides where and how your application will look for the correct key. You can use any of them to suit your objectives.

Note: *The 6.3 client library is fully-supported on the Red Hat Linux platform as on Windows. However, the Linux version of Sentinel SuperPro only supports the TCP/IP protocol.*

Note: *SuperPro on Macintosh does not provide network functionality. Therefore, the network APIs will return error when called. Refer to "Part III – The Final Steps" on page 55 for knowing the names of the APIs that cannot be used.*

Highlights of the SuperPro 6.3 Client Library

The major features of the 6.3 client library are as follows:

- **Direct-to-Driver Communication Allowed on a Local System**– Under *direct-to-driver* communication, your protected application can communicate directly with the Sentinel System Driver on a local computer—without the aid of the SuperPro server.

  The two stand-alone access modes introduced in the 6.3 client library—RNBO_STANDALONE and RNBO_SPN_DRIVER—are not dependent upon the SuperPro server for obtaining a license from the SuperPro key attached to a local computer. Further, both of them offer unlimited access to applications running locally.

  The 6.1 and 6.2 releases were based on a *client-server* architecture, and hence required all users (stand-alone or networked) to install the SuperPro server on their computer. Such a requirement disrupted the migration of pre-6.1.x clients who were not using any server to access the keys attached on a local computer. The current release, therefore,

provides a smooth migration path for pre-6.1.x clients. See the topic "Migration Path for the Pre-6.1.x Users" on page 25 for details.

Tip: *See table "6.3 Client Library Access Modes" on page 17 and figure "Access Modes Defined in Sentinel SuperPro Client Library" on page 21 to learn more about various access modes.*

- **Server-Based Communication Optional for Stand-Alone Users**– To maintain support for the 6.1 and 6.2 stand-alone clients, who used the SuperPro server for accessing a key attached to local computer., RNBO_SPN_LOCAL has been introduced.

- **Network Broadcast Maintained**–Using RNBO_SPN_BROADCAST, a broadcast can be sent over the subnet to find a SuperPro server that has a license to offer.

- **Access Modes with Combined Functionality**–The RNBO_SPN_ALL_MODES and RNBO_SPN_SERVER_MODES access modes provide *two-in-one* functionality. The license request is first sent to the key(s) attached to the local computer—followed by a network search if no license was obtained in the last attempt. Note that RNBO_SPN_ALL_MODES does not use the SuperPro server on the local computer, while RNBO_SPN_SERVER_MODES does.

- **Set a Specific SuperPro Server**–The RNBOsproSetContactServer API or the NSP_HOST variable can be set to a specific server name or IP/IPX address. The application will only look that SuperPro server for a license.

- **Deprecated Status for No-Net**–The no-net mode behaves similar to RNBO_SPN_DRIVER. Note that it is provided for backward-compatibility with the previous releases but is not guaranteed to be available in future—you should switch to RNBO_SPN_DRIVER instead.

- **No Distinction Between Stand-Alone and Network Keys**–Under *direct-to-driver* communication, a SuperPro key will allow unlimited access, and up to its hard-limit when contacted through the SuperPro server— regardless of whether the key is stand-alone or network.

License Limit for the SuperPro Keys

Scenario	Stand-Alone Key	Network Key
With SuperPro Server	1 license	Number of licenses equal to its hard limit
Direct-to-Driver	Infinite licenses	Infinite licenses

Note: *The SuperPro server can manage licenses for up to ten SuperPro keys on a system. These can be any combination of parallel and USB keys, however there is a limit of five parallel keys cascaded on a single parallel port. Similarly, the USB driver can support a maximum of 32 keys attached to a port/hub.*

6.3 Client Library Access Modes

Description	Comments
RNBO_STANDALONE (Without License Management Capability)*	
• Uses *direct-to-driver* communication—application does not use the SuperPro server for obtaining a license. • A local user is provided with unlimited access. Multiple instances of the various applications associated with the key can be run.**	• Provides a compatible migration path for the pre-6.1.x clients. • No need to ship the SuperPro server. • This mode cannot be set through the NSP_HOST environment variable.
RNBO_SPN_DRIVER	
• Uses *direct-to-driver* communication—application does not use the SuperPro server for obtaining a license. • A local user is provided with unlimited access. Multiple instances of the various applications associated with the key can be run without consuming extra licenses.** • In contrast to RNBO_STANDALONE, this mode uses a single license for all the applications being accessed locally. Therefore, if the SuperPro server is run, then it counts one license for the local user.	• Fast, convenient, and secure option for stand-alone users. • Capable of license management. Hence, recommended instead of RNBO_STANDALONE.* • Offers more controlled migration path for pre-6.1.x users vis-a-vis RNBO_STANDALONE. • No need to ship the SuperPro server.

* All modes, except RNBO_STANDALONE provide license management and take account of the licenses used.

** The unlimited access facility does not apply to a network user—each network client accessing the key will consume one license.

6.3 Client Library Access Modes (Continued)

Description	Comments
RNBO_SPN_LOCAL	
• The key attached to the local computer can be only accessed with the help of the Super-Pro server. If a key is not present locally—an error message is returned.	• A *client-server* mode of operation; similar environment to the 6.1 and 6.2 releases. • The SuperPro server is required. However, if a Red Hat Linux workstation does not have a network card configured for TCP/IP, then the SuperPro server (*spnsrvlnx*) will not work. • Allows you to control the number of clients that can be run locally. Unlike RNBO_STANDALONE and RNBO_SPN_DRIVER, no unlimited access is provided—every instance of an application uses one license.
RNBO_SPN_BROADCAST	
• When invoked, the application simply broadcasts to the network to obtain a license.	• Results may take some time as a broadcast is made to the network—searching for any SuperPro server running on the local subnet associated with the particular SuperPro key needed to fulfill the license request. This may also strain your network resources. • The SuperPro server is required.

6.3 Client Library Access Modes (Continued)

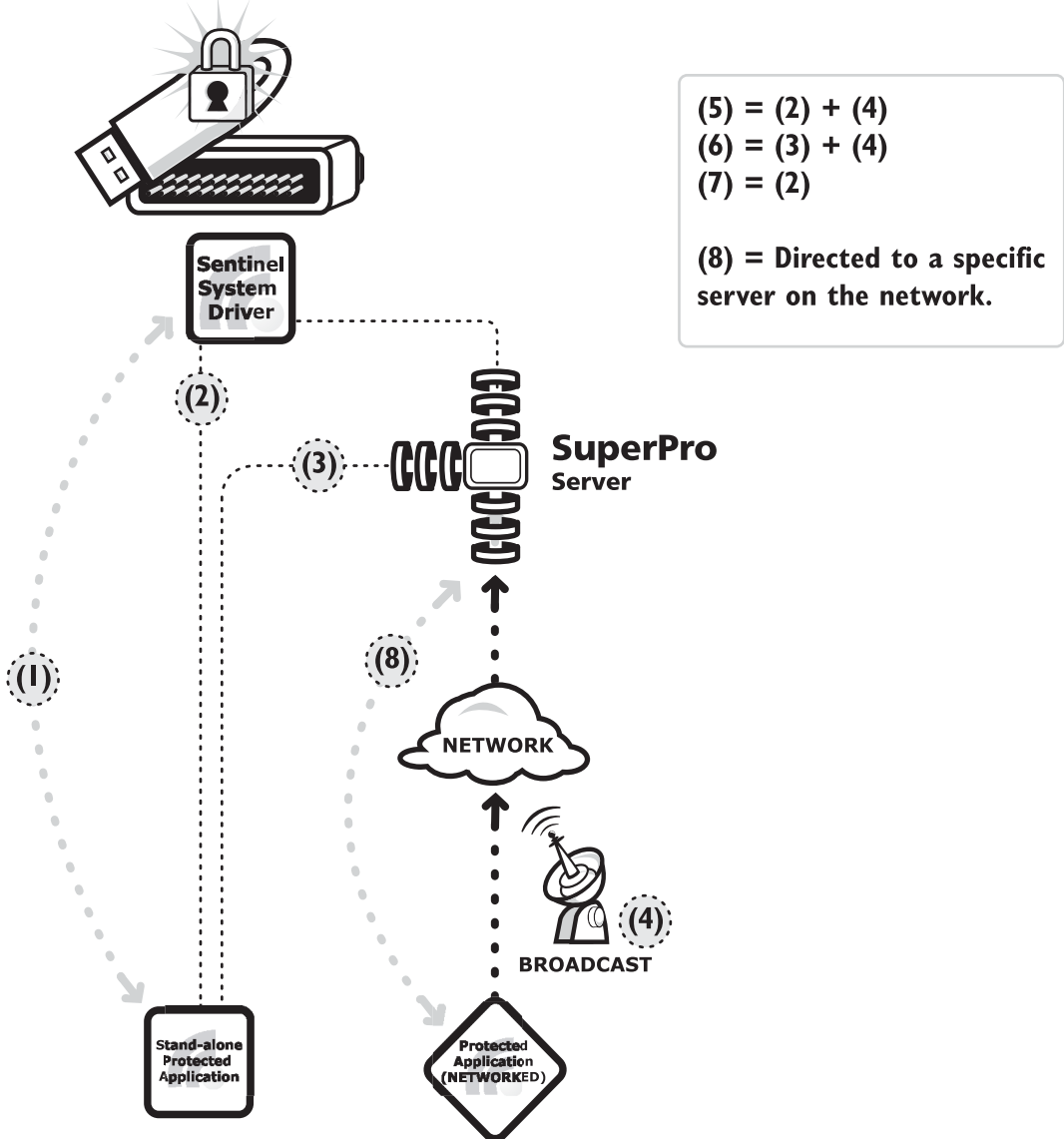| Description	Comments
RNBO_SPN_ALL_MODES	
• It performs the same function as using RNBO_SPN_DRIVER followed by the RNBO_SPN_BROADCAST mode.	• The SuperPro server is unnecessary if a license is obtained from the key(s) on a local system. However, it is essential for network operations. • Use this for maximum flexibility when you have a limited idea about the customer's setup. • Since this mode is set by default, you need not add any API functions to your source code.
RNBO_SPN_SERVER_MODES	
• It performs the same function as using the RNBO_SPN_LOCAL access mode followed by the RNBO_SPN_BROADCAST mode.	• The SuperPro server handles all the communication and manages licensing. Hence, it must be shipped along with your protected application. • Similar to the options offered in 6.1 and 6.2 releases. You may continue using this option for a set of customers familiar with this mode.
No-Net	
• The application will look only on the local system for a key.	• The no-net mode has been deprecated; switch to RNBO_SPN_DRIVER instead.

6.3 Client Library Access Modes (Continued)

Description	Comments
Servername/IPaddress/IPXaddress	
• If the function is set to a specific server (by its name or IP address and IPX address), the application will look only for the designated server. If the computer containing the selected SuperPro server is not found, or a key is not found on it, the application will not send a broadcast message to the network.	• It provides the most controlled mode of operation. Prevents the application from making a search by sending a broadcast message to the entire network. Hence, this mode saves both time and network resources used in the license acquisition process. • Since you cannot assume the server name of your customers in advance, you should alternatively allow the system administrator to set the server name using the NSP_HOST variable on each workstation.

Note: *On the system where terminal services are enabled, the SuperPro service is essential for accessing the key. Further, the Toolkit and stand-alone utilities (Make Keys, License Generator and Field Exchange) cannot be run on the terminal clients.*

Sentinel SuperPro™ Key

(5) = (2) + (4)
(6) = (3) + (4)
(7) = (2)

(8) = Directed to a specific server on the network.

Sentinel System Driver

(2)

SuperPro Server

(3)

(1)

(8)

NETWORK

(4)

BROADCAST

Stand-alone Protected Application

Protected Application (NETWORKED)

Access Modes Defined in Sentinel SuperPro Client Library

Setting an Access Mode

An access mode can be set using any of the following three options:

- **Using the Toolkit Wizard**–If you have used automatic protection (Shell), then the **Element Definition Wizard** in the **Design** stage shows you options for setting an access mode. See the topic "Using Automatic Protection" on page 28 for details.

Note: *The automatic protection option is not supported on the SuperPro Red Hat Linux release.*

- **Setting the NSP_HOST Environment Variable**–This variable can be set by your customers on their computers with values discussed in the topic "Set Access Mode Using NSP_HOST" on page 24.

- **Using the RNBOsproSetContactServer API**–Applications protected using integrated protection and custom elements require modification in the source code using the RNBOsproSetContactServer API function. Refer to the section "Using the RNBOsproSetContactServer API Function" on page 22 for details.

Using the RNBOsproSetContactServer API Function

After initializing the packet, use the RNBOsproSetContactServer API to set the desired access mode. Note that in case of *direct-to-driver* communication, the network APIs—RNBOsproGetSubLicense, RNBOsproSetProtocol and RNBOsproSetHeartBeat—will return an error if called. The protocol for broadcast is determined by the settings provided in the RNBOsproSetProtocol API. If it is not called, the broadcast will use the TCP/IP protocol by default. The RNBOsproSetProtocol API will return an INVALID_OPERATION error on Linux as only TCP/IP is supported.

Format

```
unsigned short int RNBOsproSetContactServer(
        RBP_SPRO_APIPACKET    packet,
        char                  *serverName
);
```

Parameters

packet	A pointer to the RB_SPRO_APIPACKET structure.
serverName	The *serverName* can be set to any of the following values:

1. RNBO_STANDALONE
2. RNBO_SPN_DRIVER
3. RNBO_SPN_LOCAL
4. RNBO_SPN_BROADCAST
5. RNBO_SPN_ALL_MODES
6. RNBO_SPN_SERVER_MODES
7. No-Net
8. Servername/IP Address/IPX Address

Tip: *If you set the SuperPro server name to blank, the default mode of RNBO_SPN_ALL_MODES is restored.*

Calling the RNBOsproSetContactServer API Multiple Times

You can call the RNBOsproSetContactServer API multiple times using different access modes so that the various sites are searched for in the desired order. This allows developers the flexibility to tailor their licensing policies. For example, an application has a particular search strategy for obtaining a license. It should use a particular "preferred" SuperPro server first and then try local access if this server fails. Finally, the application should do a network broadcast, if all else fails, to search for another SuperPro server.

Note: *It is not possible to use multiple FindFirstUnit calls with the same API packet once the correct key is found. Ensure that you modify the code to have different API packets for each FindFirstUnit call.*

To do this, you will first set the API to the hostname or IP address of the preferred server. If the RNBOsproFindFirstUnit API fails, you will then call the RNBOsproSetContactServer API again setting the mode to RNBO_SPN_DRIVER and retry the find first call. If that fails also, you will use the RNBO_SPN_BROADCAST mode and try one last time. In this way, the developer can loop through all the desired communication modes and possible servers in the desired order.

Set Access Mode Using NSP_HOST

You can optionally allow your customers to set an access mode through the NSP_HOST environment variable. You have the following choices available depending upon the protection type used:

If Using Automatic Protection (Shell)

Please refer to details given in step 5 of the topic "Using Automatic Protection" on page 28 for details.

If Using Integrated Protection/Custom Elements

All modes, *except*, RNBO_STANDALONE, can be set through NSP_HOST. The possible values for the variable are: `RNBO_SPN_DRIVER`, `RNBO_SPN_LOCAL`, `RNBO_SPN_BROADCAST`, `RNBO_SPN_ALL_MODES`, `RNBO_SPN_SERVER_MODES`, `NO-NET` and server host name, IP or IPX address. These strings are not case-sensitive.

You must note that an access mode set through code will always override the mode set through the environment variable. If the API is not called and no environment variable is set, the default setting is RNBO_SPN_ALL_MODES.

Tip: *To learn how to set NSP_HOST on a Windows 9x system, see the topic "Instructions for Setting the NSP_HOST Environment Variable on Windows 9x" on page 65. To set the variable on Windows NT/2000/XP, see the topic "Setting the Client Environment Variable" on page 19 of the SentinelSuperPro 6.1. System Administrator's Guide.*

Migration Path for the Pre-6.1.x Users

Having modes that allow *direct-to-driver* communication, it is now possible to extend support to pre-6.1.x users. You should use the 6.3 client library and follow the instructions given below:

1. Call the RNBOsproSetContactServer API with the `RNBO_STANDALONE` flag in your application to make it completely independent of the SuperPro server or use the `RNBO_SPN_DRIVER` flag to allow license management functionality.

2. Now, compile your application with the multi-threaded (MT) switch. Ensure that you ship Sentinel System Driver version 5.41 or higher with your protected application.

Note: *It is not possible to use multiple FindFirstUnit calls with the same API packet once the correct key is found. Ensure that you modify the code to have different API packets for each FindFirstUnit call.*

Interfaces to the Client Library

This release provides interfaces to the SuperPro client library for several development languages and environments. Each interface includes a library, source code of the example program and *.pdf* document that assists you in understanding the interface.

The table below summarizes the various interfaces available on each platform:

List of Interfaces Supported on Various Platforms

Compiler/Environment and Version(s)#	Windows	Linux	Macintosh
.NET Object (C#) Visual Studio 7.0	✔		
.NET Object (VB.NET) Visual Studio 7.0	✔		
Microsoft Visual C++ 6.0 and 5.0	✔		
Microsoft Visual Basic 6.0 and 5.0	✔		
Compaq Visual Fortran 6.6	✔		
COM Object Microsoft Visual C++ 6.0	✔		
Java Native Interface J2SEv1.3		✔	
Java Native Interface J2SEv1.4	✔		
Borland C++Builder 6.0 and 5.0	✔		
Borland Delphi 6.0 and 5.0	✔		
AutoCAD AutoCAD 2000 with ARX 16 and AutoCAD 2002 with ARX 2002	✔		
MetroWerks CodeWarrior 3.0	✔		
Borland Kylix 2.0		✔	
GNU C 2.96 (command-line only)		✔	
GNU CC compiler version 3.1			✔

Chapter 3

Additional Features for Windows

This chapter discusses automatic protection enhancements, Combo Installer, and miscellaneous changes for Sentinel SuperPro. The chapter also guides you on what to send to your customers.

The topics are arranged in the following order:

- "Quick Automatic Protection Option in Toolkit" on page 27.
- "Combo Installer for Windows" on page 34.
- "Miscellaneous Changes" on page 35.
- "What to Send to Your Customers and Distributors" on page 37.

Quick Automatic Protection Option in Toolkit

The Toolkit's automatic protection (Shell) option has been a commonly used method for protecting executables in a quick and effective way. In order to further reduce the total number of steps followed, the quick automatic protection option has been introduced. Your application can be protected and the key can be programmed with just a few clicks through a *wizard-based* interface in the Toolkit.

The automatic protection method does not require you to modify the application's source code but automatically wraps a protective shield around it. As a result, your application will not run *unless* a correct key is present throughout the entire session.
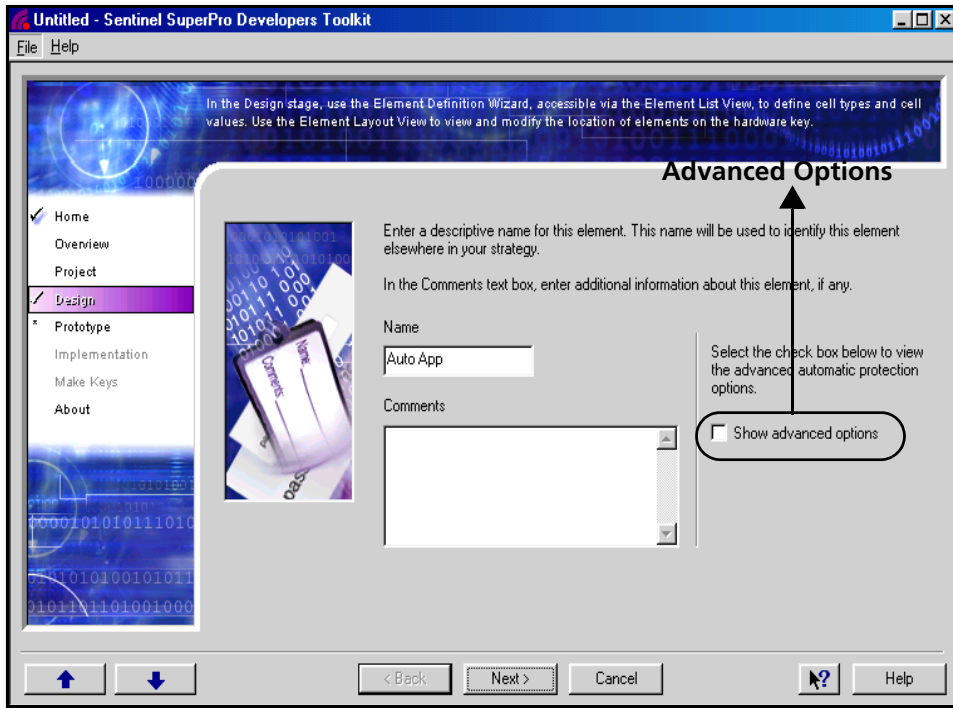
Using Automatic Protection

Protecting your application using automatic protection is so fast and easy, even someone who is new to Sentinel SuperPro will find it simple and straightforward to do. No preparation is required in advance *except* your input files. Follow the steps given below to use it:

Tip: *The steps described below also include details on various other enhancements of automatic protection (Shell) introduced in Chapter 1.*

1. Configure your Toolkit if starting the Toolkit for the first-time on the system. See "Configuring the Windows Toolkit" on page 60 for details. The **Home** screen appears after configuration.

2. Now, either use the hyperlink provided on the **Home** screen or navigate to the **Design** stage and click the **Automatic Protection** button. The **Element Definition** wizard is displayed.

3. Assign a name and comment for your application. This name will be shown in the **Element List View**.

Select Advanced Options for Additional Controls

Tip: *The default automatic protection options shown are for quickly protecting your applications. However, advanced options—such as expiration date, time control and so on—can also be added for better control. To do so, select the **Show advanced options** check box. Also see the topic "Choosing Between Advanced and Quick Automatic Protection" on page 33.*

4. The next dialog requires you to select your unprotected input file that is to be protected. Also, specify a name for your protected output file that will be shipped to your customers. The two file names cannot be identical if existing at the same location. This also keeps your original file intact. The **Overwrite existing file** check box is selected to over-write any previously existing file with same name.

5. You must also decide how you want your protected application to access the key. Choose any of the options given below:

*Tip:* *See "Highlights of the SuperPro 6.3 Client Library" on page 14 to learn more about the various access modes.*

- **Standalone**: The RNBO_STANDALONE access mode is followed.
- **Network**: This option requires the SuperPro server. The RNBO_SPN_SERVER_MODES access mode is followed, unless a specific server is set in the NSP_HOST environment variable. This is also the default mode.
- **All Modes**: The RNBO_SPN_ALL_MODES access mode is followed.

Note: *The priority of the NSP_HOST environment variable over the settings done using the **Element Definition** wizard in Toolkit is only valid with the **Network** option. The NSP_HOST variable setting on a client machine will not take effect if the mode chosen is **Standalone** or **All Modes**.*

6. You can optionally apply execution controls—the number of times your protected application will run before it expires. Select the **Limit the Number of executions** check box and select or enter a value between 0 and 65535. If you do not check this option, then your application will remain inactive on the customer's computer and can be activated only through the Field Exchange utility. Click **Next** to proceed.

7. The next dialog displays a summary of the settings given for protecting an application using quick automatic protection. You can click **Back** to edit. Note that you can still opt for the advanced options by going back to step 3.
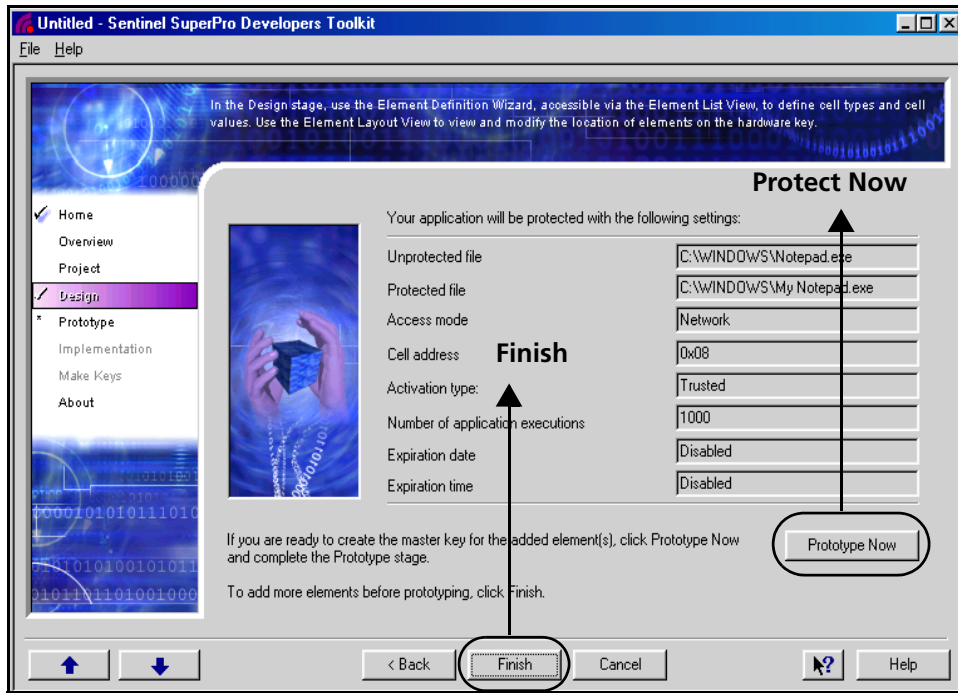
Summary of the Automatic Protection (Shell) Settings

- **Unprotected File**: This refers to the input file you chose in step 4.

- **Protected File**: The location where your protected application will be stored.

- **Access Mode**: The mode you specified in step 5.

- **Cell Address**: The protected application will occupy a minimum of four cells depending upon their availability. The Toolkit automatically allocates the appropriate cell address.

- **Activation Type**: This will always be Trusted in case of quick automatic protection.

- **Number of Application Executions**: If enabled, it shows the number of times your protected application will run.

- **Expiration Date and Expiration Time**: These options, not available under quick automatic protection, allow you to control how long the application can be run, in days, hours and/or minutes. You can select the **Show advanced options** to view them.

8. After reviewing your protection strategy, you can either click the **Finish** or **Prototype Now** button.

   Clicking **Finish** will bring you to the **Element List View**, where you find the element you just added. You can also add more elements, and edit/delete the existing ones. For adding new elements, you may also consider using any option other than the quick automatic protection. The number of applications your key can accommodate is directly related to the protection strategy specified. Afterward, you need to prototype the key in the **Prototype** stage.

   Clicking **Prototype Now** instead of **Finish** saves you from separately prototyping the master key for the elements protected using automatic protection. It does the following in this order:

   - Programs the master key.

   - Generates the query-response pairs.

   - Regenerates auto-actions for field exchange—such as activating an inactive application or disabling a counter.

   - Adds a protective layer to your application's executable file. Hence, this step also implements your automatic protection strategy, and you are no longer required to pass through the **Implementation** stage. Once the layer has been added, all the protection options you defined for the application are in place.

9. You now reach the **Make Keys** stage, where you can program as many keys as you want for all the elements included in the **Element List View**.

Editing an Element

You can edit the element added by selecting it from the **Element List View** in the **Design** stage. Now, click the **Edit Element** button and follow the steps shown by the wizard.

Note: *For both automatic and quick automatic protection, the enhanced algorithm engine is enabled, and the default activation type has been changed to Trusted.*

Deleting an Element

You can delete the element added by selecting it from the **Element List View** in the **Design** stage**.** Now, click the **Delete Element** button and click **Yes** when asked to confirm the deletion.

Choosing Between Advanced and Quick Automatic Protection

The quick automatic protection is best suited for quickly protecting your demo applications (limiting its number of executions). However, if you select the **Show advanced options** check box under the **Design** stage, the advanced automatic protection options will be shown with full-functionality and controls. The table below compares the various options available with each type:

Options in Quick Versus Advanced Automatic Protection

Option Available	Quick	Advanced
Activation Type	Trusted only.*	All four; active, static, distributed and trusted.
Demo Options	Execution Control only.	Execution control, expiration date, time control and encryption of additional files.
Means of Activation in Field	Field Exchange utility only.	Field Exchange utility and Sentinel Client Activator.

Options in Quick Versus Advanced Automatic Protection (Continued)

Option Available	Quick	Advanced
Customization of Cell Address and Error Messages	Not allowed. The appropriate cells are chosen based on their availability, and default error messages are shown at runtime.	Allowed.
Override Default Algorithm Values	Not allowed; instead the random algorithm values can be used.	Allowed.

* The activation type allowed under quick automatic protection is Trusted only. The application is inactive until activated with an activation password unless it is a demo or metered application. To activate such applications in the field, you must also ship the Field Exchange utility.

Combo Installer for Windows

Combo Installer is an integrated installer of the Sentinel System Driver, SuperPro server and Monitoring Tool. These components are required by your customers who use the network applications protected using Sentinel SuperPro. Combo Installer offers Windows Installer-based installation methods for flexible and easy installation.

The Combo Installer directory in the Sentinel SuperPro CD includes *ComboInstallWin32_Start.htm*, an online HTML document that explains how to integrate it in your application installer. The Help also contains notes for developers not using Windows Installer and will instead use the binaries included in the *\Legacy* directory.

For stand-alone applications using the *direct-to-driver* communication, you can integrate the Sentinel System Driver using the instructions given in *SentinelDriverInstall_Start.htm* available in the Sentinel System Driver CD.

Miscellaneous Changes

The following miscellaneous changes have been made to the product:

- ■ **USB Driver is WHQL Certified for Windows 2000 and XP**

    The Sentinel USB driver is Microsoft Windows Hardware Quality Labs (WHQL) certified for Windows 2000 and XP. This allows Windows 2000 and XP Logo compliance for applications that use the Sentinel SuperPro USB hardware keys.

- ■ **Support for User-Configured, Parallel Port PCI Cards**

    The Sentinel System Driver from now on supports user-configured, PCI parallel port cards. Use the Sentinel System Driver configuration utility—*SetupSysDriver.exe*—on Windows 98/ME/NT/2000/XP to configure the PCI port addresses. Subsequently, the Sentinel parallel port driver will be able to identify both the *built-in* and *user-defined* parallel ports right from startup.

- ■ **Client-Side Deployment of SuperPro Service**

    Previously, installation and uninstallation of the SuperPro service was allowed only through *loadserv.exe*. Now, the following command-line options can be used optionally on a Windows NT/2000/XP system.

Command-Line Options for SuperPro Service

Option	Description
-i	To install the service.
-u	To uninstall the service.
-l	To set the log file name.
-f	To set the error file name.

You can manually integrate them in your own installer program. For example, the spnsrvnt.exe -i -l "<path of Log file><Log File name>" -f "<path of error file><Error file name>" command will install the server and also create log and error files at

the specified path using the names given.

■ **No Need to Run SuperPro Server While Using Toolkit**

Earlier an error "Sentinel SuperPro Server is not running" was displayed whenever Toolkit or other utilities such as License Generator, Make Keys and Field Exchange were used without running the server. Now, since it is possible to directly communicate with the hardware key (product and distributor), it is no longer necessary to run the SuperPro Server while using them.

■ **SuperPro Server Shutdown Utility**

A utility *(spnsrvstop.exe)* is provided to stop the SuperPro server running on a Windows system. A custom action is also defined in the SuperPro server merge module (*SuperProNetServers.msm*) using which you can stop the server before uninstallation. See *ComboInstallWin32_Start.htm,* an online HTML document for details on the custom actions supported.

Note: *The SuperPro server shutdown utility does not work on Novell.*

■ **Changes to Toolkit's User Interface**

The Toolkit's graphical user interface has been enhanced for friendlier and enriched navigation. Apart from various enhancements in automatic protection (Shell), several modifications were made to the Toolkit, including the **Welcome** screen and the **API Explorer**. The **API Explorer** now shows a list of all the SuperPro APIs to quickly get started. An additional parameter called secret code has also been added for **Developer Configuration** under the **Project** stage. "Configuring the Windows Toolkit" on page 60 for more information about the secret code.

■ **New Interfaces to the SuperPro Client Library**

Interfaces have been provided for the following high-level languages in this release: .NET Object (VB.NET) Visual Studio 7.0, .NET Object (C#)

Visual Studio 7.0, MetroWerks CodeWarrior 3.0, and COM Object VC 6.0.

■ **Support on Windows XP**

Sentinel SuperPro is also supported on Windows XP.

What to Send to Your Customers and Distributors

Apart from sending a Sentinel SuperPro key programmed with the values expected by your application, you should provide a few other components to your customers. Use the following checklist to verify that what components are to be shipped to your customer along with your application, associated data files and SuperPro key.

If your company uses distributors to sell your products, then all the customer items listed in third column must also be passed to your distributors. In addition, your distributor will need a distributor key to manage the number of activations or updates performed *plus* all the items selected in the Distributor column.

List of Items to be Shipped to Your Windows User and Distributor

	Component	End User	Distributor
1.	Sentinel System Driver version 5.41 or later*	✔	✔
2.	SuperPro Server (Windows9x/NT)*	✔	
3.	SuperPro Monitoring Tool and its Help*	✔	
4.	*Usafe32.dll*	✔	
5.	*Dsafe32.dll*		✔
6.	*Msvcrt.dll* (mandatory for applications using automatic protection)	✔	

List of Items to be Shipped to Your Windows User and Distributor

	Component	End User	Distributor
7.	Data Protection Driver (mandatory for encryption/decryption of data files on Windows 9x)	✔	
8.	Field Exchange Utility and its Help**	✔	
9.	License Generation Utility *plus* the following files: *lang_enu.dll*, *sp_gXX.dll*, *makedll.dll* and *spcommon.dll*		✔
10.	Sentinel Client Activator**	✔	
11.	*Sentinel SuperPro System Administrator Guide* (in PDF format)	✔	
12.	If you are shipping the HTML Help for the Monitoring Tool or the Field Exchange *hhupd.exe* and *hhactivex.dll* are needed.	✔	
13.	The Sentinel SuperPro distributor's project file (*.DST*) with the protection strategy for the application you are shipping.		✔

  * For Windows platforms, you may consider shipping Combo Installer instead of components 1,2 and 3.

  ** The Sentinel Client Activator or Field Exchange Utility (*fieldexutil.exe*) is required if your application is a demo or is programmed to allow field activation. Items 4 and 5 are also associated only with the Field Exchange utility.

Chapter 4

Additional Features for Red Hat Linux

Sentinel SuperPro is now also available on the Linux Red Hat platform. This brings the popular Sentinel key based-protection to a wide community of Linux developers—allowing them to safeguard and control the use of their applications.

Until now, the Linux developers were provided with the Sentinel UNIX Driver and a stand-alone client library to allow communication with the SuperPro parallel port key in non-networked environments only. The 6.3 release goes beyond this by adding network support using the SuperPro server—multiple clients across the network can now access a SuperPro key concurrently. You can even remotely activate inactive applications or enable selective features using the Field Exchange utility. This release also introduces support for the SuperPro keys in USB form factor.

Apart from discussing these features, the following topics are also covered:

- ■ "Getting Started" on page 41. This topic will explain you the *step-by-step* process of protecting an application.

- ■ "Combo Installer for Linux Redistributables" on page 44.

- ■ "What to Send to Your Customers and Distributors" on page 45.

- ■ "Frequently Asked Questions" on page 47.

Tip: *The APIs and various programs/utilities referred within this chapter, such as the SuperPro server, Field Exchange, Monitoring Tool, are functionally similar to the Windows components, unless stated otherwise. Therefore, developers who are already familiar with the Windows version of the software will find it quick and easy to learn about the Linux version of Sentinel SuperPro. First-time developers are recommended to begin with the* Sentinel SuperPro Developer's Guide for Windows.

The major highlights of the SuperPro Linux package are as follows:

Support for SuperPro USB Keys

This release on Red Hat Linux also supports the SuperPro USB keys in addition to the parallel port keys. A daemon has been provided to allow accessing the USB keys.

Network Support Added

Using the SuperPro server, a Sentinel key can be shared by multiple clients across the subnet. The SuperPro server and Sentinel UNIX Driver must be installed on the system that has the key attached. The SuperPro server runs as a daemon and handles the entire license management. The SuperPro Monitoring Tool is also provided which tracks the SuperPro servers running on the subnet.

Note: *The SuperPro server for Red Hat Linux only supports the TCP/IP protocol.*

Field Upgrade Capabilities

Like Windows, a Linux application too can be updated remotely with the help of the Field Exchange utility. You ship the Field Exchange utility with your protected application, using which your customer generates the *locking code* of the key and passes it to you or your distributor using an e-mail message or a phone call. You then use the Windows License Generator to obtain a corresponding license code. Your customer updates the key by applying the license code through the Field Exchange utility.

Thus, you can control application use from a distance and support personalized marketing programs securely with minimum effort.

The core of the field exchange capability lies in the *SafeCfg* utility. which is configured with some vital information to produce a set of complementary objects known as *dsafe* and *usafe*. See Chapter 6, "Platform-Independent Field Activation" for details. The *usafe* shared object is shipped with the Field Exchange utility to allow this remote activation.

Warning! *Each time you modify your configuration settings, a new pair of USAFE and DSAFE objects is reproduced. However, to serve your applications already in the field, you must make a backup of objects corresponding to the existing settings.*

Getting Started

Sentinel SuperPro on Linux takes advantage of the *easy-to-use* Windows Toolkit and can be used to perform similar actions on the SuperPro key. Your actions to protect an application can be divided into four parts as described below:

Note: *The* Sentinel SuperPro Developer's Guide *explains in detail the procedures and concepts for using the Toolkit. However, due to subsequent changes in Sentinel SuperPro, some information, including the access modes, is out of date in that guide. This addendum covers in detail the changes for the 6.2 and 6.3 releases.*

Part I – Using the Windows Toolkit

Use the Windows Toolkit to design your protection strategy and program the hardware keys. You also define the field activation commands here, if you wish to allow field activation. The Toolkit also allows you to evaluate the behavior of the API functions with your programmed key prior to putting them in your source code, using the API Explorer.

Note that the automatic protection option, in which a protective layer is added around your executable, is not supported on Linux. You can instead customize a unique protection plan using integrated application protection. With this, you can take advantage of a number of protection techniques, both basic and advanced, using those that work best for your application. You can also add individual design elements to your protection strategy, known as custom elements.

The exact steps followed to protect an application are presented below:

1. On a computer where the SuperPro software for Windows is installed, point to **Programs > Rainbow Technologies > SuperPro > 6.3 > SuperPro Toolkit** from the **Start** menu.

2. If starting the Toolkit for the first-time, you will be prompted to configure the Toolkit with your developer ID, passwords and a secret code. Refer to the topic "Configuring the Windows Toolkit" on page 60.

3. Navigate to the **Design** stage and click **Integrated Protection** under **Element List View**.

4. Assign a name and comment for your application. This name will be shown in the **Element List View**. Click **Next**.

5. From the **Address** drop-down list, select the address of the cell you want the first word of the element to be placed in. If the location is unimportant to you, select **Auto** to allow the Toolkit to select a location for you.

6. Now, if you want the application to be a demo application, select the **Demo** counter check box and specify an integer value between 0 and 65535.

7. You can select the **Override default algorithm values** check box to specify your own algorithm values, although, we recommend that you use the default ones. See the instructions on page 148 of the *Sentinel SuperPro Developer's Guide* for detailed instructions.

8. In the next screen, you are required to specify an activation type for your application. You can choose any of the following activation types: active, static, trusted or distributed. Refer to page 60 of the *Sentinel SuperPro Developer's Guide* to learn more about each activation type. Note that a Linux application in the field can only be activated using the Field Exchange utility. Support for the Sentinel Client Activator is not provided for the SuperPro Linux release. Clicking **Next** will bring you back to the **Element List View**, where you find your element added in the list.

9. Now, navigate to the **Prototype** stage. Click **Go**. This will: program a master key, generate query/response pairs, define default actions for remote activation. More commands for field activation can be added in the **Implementation** stage.

10. You are automatically brought to the **Implementation** stage, where a pseudocode protection plan is generated. Although, it cannot be used directly in your source code, it gives an idea about the sequence of the various APIs to be called. It also shows you the expected responses for queries to the algorithms programmed into your keys. You may wish to copy this section from the plan for reference. Use any of the **Save** options to save the pseudocode for reference.

11. If you wish to allow remote activation for your application in the field, you must define the field activation commands. Click the **Field Activation** tab. Refer to Chapter 10, "Defining Field Activation Actions" of the *Sentinel SuperPro Developer's Guide* for exact procedure.

12. Once you are finished with the **Implementation** stage, you can proceed to make your product and distributor keys in the **Make Keys** stage. Refer to Chapter 11, "Programming Keys" of the *Sentinel SuperPro Developer's Guide* for detailed information.

13. Save your project (*.spp*) in the **Project** stage.

Part II – Working in Your Application Source Code

Afterward, you can add the SuperPro API functions to your source code. You can refer to the pseudocode obtained from the Toolkit. Interfaces are provided for GCC, Java, and Kylix to assist you in adding the APIs to code. Each interface directory consists of a document in PDF format that shows the exact syntax for each of these languages, and explains how to build the example program given.

Note: *Ensure that you also link your application with the* pthread *system library along with the SuperPro client library.*

Part III – Configuring SafeCfg

If you are allowing field activation, configure the SafeCfg utility using the instructions given in the section "Platform-Independent Field Activation" on page 57. The *libusafe.so* will be shipped along with your protected application.

Part IV – The Final Steps

Finally, you should prepare an installer for your protected application. Consider using Combo Installer for installing the end user components as explained below. The section "What to Send to Your Customers and Distributors" on page 45 provides a helpful list.

Combo Installer for Linux Redistributables

To ease the installation of the SuperPro redistributables—the parallel port driver, USB daemon, SuperPro server, and Monitoring Tool—on your customer's computer, the Combo Installer has been provided.

The */SuperProNetComboInstaller* directory on the CD contains the following files:

File	Description
combo_install.sh	A script that will install all four components.
combo_uninstall.sh	A script that will uninstall all four components.
sspmontool-6.3.1-0.i386.rpm	The SuperPro Monitoring Tool RPM.
SUD-ParallelDriver-5.50-0.i386.rpm	The Sentinel UNIX Driver (parallel port) RPM.
SUD-USBDaemon-5.50-0.i386.rpm	USB daemon RPM.
sspserver-6.3.1-0.i386.rpm	The SuperPro Server RPM.
readme.pdf	Combo Installer readme file for your customer.

The installation script does the *step-by-step* installation of each component, hence you can create your own copies of the Combo Installer CD and distribute them with your product. Otherwise, you can call the individual RPMs within your application's installation script.

What to Send to Your Customers and Distributors

Apart from sending a Sentinel SuperPro key programmed with the values expected by your application, you should provide a few other components to your customers. Use the following checklist to verify that what components are to be shipped to your customer along with your application, associated data files and SuperPro key.

If your company uses distributors to sell your products, then all the customer items listed in third column must also be passed to your distributors. In addition, your distributor will need a distributor key to manage the number of activations or updates performed *plus* all the items selected in the Distributor column.

List of Items to be Shipped to Your Linux User and Distributor

	Component	End User	Distributor
1.	Sentinel UNIX Driver version 5.50 or later and its documentation	✔	
2.	Sentinel System Driver 5.41 or later for Windows		✔
2.	SuperPro Server for Linux	✔	
3.	Monitoring Tool for Linux and its documentation	✔	
4.	*libusafe.so*	✔	
5.	*Dsafe32.dll**		✔
6.	Field Exchange Utility for Linux and its documentation	✔	
7.	SuperPro License Generator Utility for Windows		✔
8.	Sentinel Client Activator	Not Available on Linux.	
9.	Documentation You should provide sufficient information to your customers on setting the NSP_HOST variable depending on the access mode you set. Relevant information about the SuperPro server, Sentinel UNIX driver, hardware key and other redistributables may help them as well. You must look into the documentation we include with the redistributables. Distributors can be provided with documentation on topics such as, how to generate licenses using the License Generator, and so on.	✔	✔

List of Items to be Shipped to Your Linux User and Distributor (Continued)

		Component	End User	Distributor
10.		The Windows License Generator Utility and the following files: *lang_enu.dll*, *sp_gXX.dll*, *makedll.dll* and *spcommon.dll*.		✓
11.		The Sentinel SuperPro distributor's project file (*.DST*) with the protection strategy for the application you are shipping.		✓

* For platform-independent field activation, ensure that you follow the same configuration settings on Windows and Linux, like the secret code, passwords and everything else. This will allow you to have identical SAFE objects. Refer to Chapter 6, "Platform-Independent Field Activation" for more information.

Frequently Asked Questions

This section consists of a few commonly asked questions that can occur to you or your customer. If you are facing any other problem in using Sentinel SuperPro, contact Technical support using the information given in "Getting Help" on page xii.

Note: *The command-line switches given below are for bash Shell.*

Question 1. What are the various commands to use the SuperPro server?

Answer: See the table below for the relevant options:

List of Commands

Action	Switch
To run the SuperPro Server	`spnsrvlnx`
To generate the log file	`-l < log file name >`

List of Commands

Action	Switch
To generate the error file.	`-f < error file name >`
To view the Help	`-h`
To display the server messages (the default setting)	`-m 1`
To hide the server messages	`-m 0`

Question 2. What steps should I follow to stop/unload the Super-Pro server?

Answer: You must use the *loadserv* Shell script to stop the SuperPro server instead of terminating the server process. The `loadserv stop` command will stop the server, and while shutting down the server will do cleanup and log the status.

Question 3. How do I configure the PCI parallel port?

Answer: The Sentinel UNIX Driver is now capable of identifying the user-configured, non-standard PCI parallel ports. As a result, the SuperPro keys attached to PCI parallel ports can also be accessed. Follow the instructions given below to identify a port:

1. Firstly, obtain the PCI parallel port address from the */proc/pci* file. Only one address can be specified at a time.

2. Now, edit the *rc.local* file available at the */etc/rc.d/rc.local* path. The entry: `insmod -f /opt/RainbowTechnologies/SUD5.50/parallel/mdrbdr.o` should be modified to: `insmod -f /opt/RainbowTechnologies/SUD5.50/parallel/mdrbdr.o address=<address>`.

   The address of the I/O port was obtained in step 1. For example, `insmod -f /opt/RainbowTechnologies/SUD5.50/parallel/mdrbdr.o address=0xdcf8` will check the 0xdcf8 port.

3. Now, reboot the computer to apply the settings. Or, to access the key right away, unload the driver using the `rmmod mdrbdr` command—followed by reloading it using the following command: `insmod -f /opt/RainbowTechnologies/SUD5.50/parallel/mdrbdr.o address=<address>`. Now, you are ready to access the key.

Question 4. How do I set the NSP_HOST variable on Linux?

Answer: You can use the `export NSP_HOST=<Value>` command in the bash Shell to set NSP_HOST. Or, you can also make an entry in the *.bash_profile* file. For other types of Shell, you should refer to the respective man pages.

The possible values are: `RNBO_SPN_DRIVER`, `RNBO_SPN_LOCAL`, `RNBO_SPN_BROADCAST`, `RNBO_SPN_ALL_MODES`, `RNBO_SPN_SERVER_MODES`, `NO-NET` and server host name or IP address. These strings are case-sensitive.

Question 5. How do I detect if the Sentinel UNIX Driver and Super-Pro server are running on a system?

Answer: The following commands are helpful:

■ The `lsmod` command will show you if the modules required by Sentinel UNIX Driver are loaded.

   • The *mdrbdr.o* is the Sentinel parallel driver module.

   • The usbcore and usb-uhci/usb-ohci (depending on the hardware specifications) modules are needed by the USB daemon.

■ The `load_daemon.sh status` command will show the USB daemon status—whether running or stopped.

■ The `loadserv status` command will show the status of the SuperPro server.

   • `spnsrvlnx (pid nnnn nnnn nnnn nnnn) is running` output indicates that the SuperPro server is running.

   • `spnsrvlnx is stopped` output indicates that the SuperPro server is not running.

By default the SuperPro server messages are displayed. To hide them, start the SuperPro server with the `-m 0` switch,

Chapter 5

Additional Features for Macintosh

Sentinel SuperPro is now also available on the Macintosh platform. You can secure your stand-alone applications—for a single customer on a specific computer—using the SuperPro USB keys.

You program your application to send calls to the hardware key to verify its presence. If the correct hardware key is attached to the user's system, it appropriately responds back and unlocks the application. Your protected application will communicate to the Sentinel USB key using the Sentinel Mac Driver.

The entire process of protecting an application is depicted below:

Program the Sentinel USB Key Using SuperPro Toolkit on Windows — **PART - 1**

Insert SuperPro APIs into Your Application's Source Code — **PART - 2**

Prepare an Installer for Your Application. — **PART - 3**
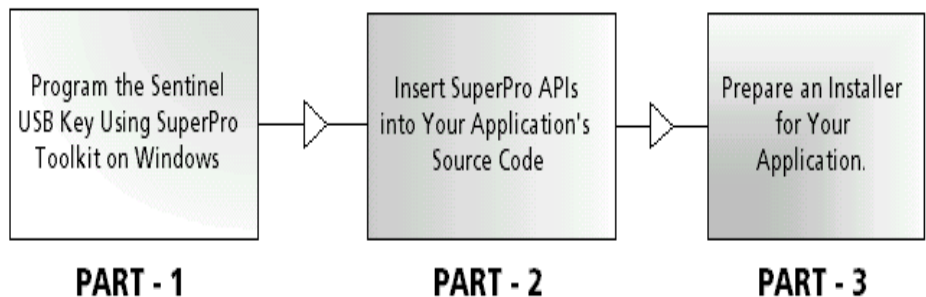
Part I – Using the Windows Toolkit

Note: *Sentinel SuperPro on Macintosh takes advantage of the easy-to-use Windows Toolkit and can be used to perform similar actions on the SuperPro key. Note that the following options: Automatic Protection (Shell), Sentinel Client Activator and Field Activation are not yet available for SuperPro on Macintosh.*

You must use the Windows Toolkit to design your protection strategy and program the hardware keys. You will be adding the software locks into your source code. Using the API Explorer, you can evaluate the behavior of the API functions with your programmed key prior to putting them in your source code.

The exact steps followed to protect an application are presented below.

1. On a computer where the SuperPro software for Windows is installed, point to **Programs > Rainbow Technologies > SuperPro > 6.3 > SuperPro Toolkit** from the **Start** menu.

2. If starting the Toolkit for the first-time, you will be prompted to configure the Toolkit with your developer ID, passwords and a secret code. Refer to the topic "Configuring the Windows Toolkit" on page 60.

3. After configuring the Toolkit, navigate to the **Design** stage and click **Integrated Protection** under **Element List View**. This will allow you to associate an algorithm with an activation type. You can instead add individual design elements (algorithms with counters and passwords, individual counters, sublicense limits, and data words that contain data you want to store on the key) to your protection strategy. These are known as *custom elements*. For more information, refer to Chapter 8, "Working With Design Elements" of the *Sentinel SuperPro Developer's Guide*.

4. Assign a name and comment for your application. This name will be shown in the **Element List View**. Click **Next**.

5. From the **Address** drop-down list, select the address of the cell you want to place the first word of the element. If the location is unimportant to you, select **Auto** to allow the Toolkit to select a location for you.

6. Now, if you want the application to be a demo application, select the **Demo** counter check box and specify an integer value between 0 and 65535. This counter value will control the number of times the application is executed (the decrement is performed by the RNBOsproDecrement function). You code your application to not run again once the counter reaches zero.

   However, upgrading from the demo versions on a customer's site is not supported on Macintosh through tools like the Sentinel Client Activator or Field Activation utility. Instead, you can write a utility that uses the RNBOsproOverwrite function to modify the counter value.

Tip: *The* Sentinel SuperPro Developer's Guide *explains in detail the APIs, procedures and concepts for using the Toolkit.*

---

7. You can select the **Override default algorithm values** check box to specify your own algorithm values, although, we recommend that you use the default ones.

8. In the next screen, you are required to specify an activation type for your application. You can specify either **Active** or **Static**. The **Trusted** and **Distributed** activation types cannot be used on Macintosh.

   Under the **Active** activation type, your application is always ready to run when the correct Sentinel key is attached.

   Under the **Static** activation type, the application is inactive until activated with an activation password. To activate an algorithm in the field, you may write your activation password utility that uses the RNBOsproActivate function once the user provides a password. Refer to page 60 of the *Sentinel SuperPro Developer's Guide* to learn more.

9. Click **Next** to return back in the **Element List View**, where you find your application or elements added.

10. Now, navigate to the **Prototype** stage. Click **Go**. This will program a master key and generate query/response pairs.

11. You are automatically brought to the **Implementation** stage, where a pseudocode protection plan is generated. Although it cannot be used directly in your source code, it gives an idea about the sequence of the various APIs to be called. It also shows you the expected responses for queries to the algorithms programmed into your keys. You may wish to copy this section from the plan for reference. Use any of the **Save** options to save the pseudocode for reference.

12. Once you are finished with the **Implementation** stage, you can proceed to make your product keys in the **Make Keys** stage. Refer to Chapter 11, "Programming Keys" of the *Sentinel SuperPro Developer's Guide* for detailed information.

13. Save your project (*.spp*) in the **Project** stage.

Part II – Working in Your Application Source Code

Afterward, you can add the SuperPro API functions to your source code. You can refer to the pseudocode obtained from the Toolkit. The CC interface is provided to assist you in adding the APIs. Since this release does not provide network functionality, the network APIs mentioned below will return an error when called:

- RNBOsproGetSubLicense
- RNBOsproSetProtocol
- RNBOsproSetContactServer
- RNBOsproGetContactServer
- RNBOsproReleaseLicense

- RNBOsproGetKeyInfo
- RNBOsproGetHardLimit
- RNBOsproEnumServer
- RNBOsproSetHeartBeat

Ensure that you also link your application with the IOKit system library along with the SuperPro client library.

Part III – The Final Steps

Finally, you should prepare an installer for your protected application. Apart from sending a Sentinel SuperPro key programmed with the values expected by your application, you must provide the Sentinel Mac Driver. You can either send the installer (*SentinelDriver1.0.0.pkg*) or integrate the driver's binary (*Sentinel.kext*) into your application's installer. You need to take care of the following points if you are including *Sentinel.kext* into your application's installer:

- Your installer must create the */Extensions* folder at the */Library* path, where the Sentinel Mac driver needs to be installed.

- In your */etc/rc* script, modify the syntax to start the daemon kextd as follows:

```
####
# Start kextd
####
if...
     ...
     ...
else
     ...
     kextd -r /Library/Extensions
```

Also, add the following syntax at the end of the rc script:

```
kextdload /Library/Extensions/Sentinel.kext
```

Tip: *Use the* `kextstat|grep Sentinel` *command to check whether the Sentinel Mac Driver is running on your system or not.*

Chapter 6

Platform-Independent Field Activation

This chapter discusses how to allow platform-independent field activation for customers regardless of the platform on which they are working. The chapter also includes the following related topics:

- "Configuring the Windows Toolkit" on page 60.

- "Configuring the SafeCfg Utility for Red Hat Linux" on page 62.

Understanding Field Activation

In field activation, you can ship an inactive (full/partial) application to your customers and provide an option to activate it remotely.

While designing your protection strategy in Toolkit, you define the field activation actions and commands that can be performed on your product keys in future. Later, your customer exchanges some information about the key (*locking code*) with you or your distributor using an e-mail message or phone call. Using that locking code, you generate a corresponding string (*license code*) to update the memory of the key.

When you generate a license code, the chosen actions and commands are encrypted into the license code. Your customer applies that license code using the Client Activator or Field Exchange utility—enabling the use of

desired features. Thus, you can control the application use from a distance and support personalized marketing programs securely with little effort.

What is Platform-Independent Field Activation?

Sentinel SuperPro version 6.3 is supported on multiple platforms. The platform-independent remote activation means that a customer on any supported platform can obtain licenses in the field from a Windows-based developer/distributor. However, the field activation option is not yet available for Macintosh.

This has been achieved by making the field exchange objects—*dsafe* and *usafe*—compatible across platforms. These objects are generated each time you change your developer information and the *secret code*. The secret code—an additional parameter introduced from 6.3 onwards—is used to create matched *dsafe/usafe* pairs.

The same secret code across various platforms ensures that compatible *dsafe/usafe* pairs are created. Hence, any *usafe* will give a locking code that can be used by *dsafe* on Windows to generate the same license string. For example, a Linux-based user can easily accept licenses from a Windows-based developer/distributor. Please refer to figure "How Do Linux and Windows Users Obtain Licenses in the Field?" on page 59.

A secret code string must be 9 to 16 characters long—consisting of at least one numeric character. For Windows, you need to submit it when you configure your Toolkit, while on Linux you use the same string to configure the SafeCfg.

Note: *The License Generator utility and Sentinel Client Activator are only available on Windows.*
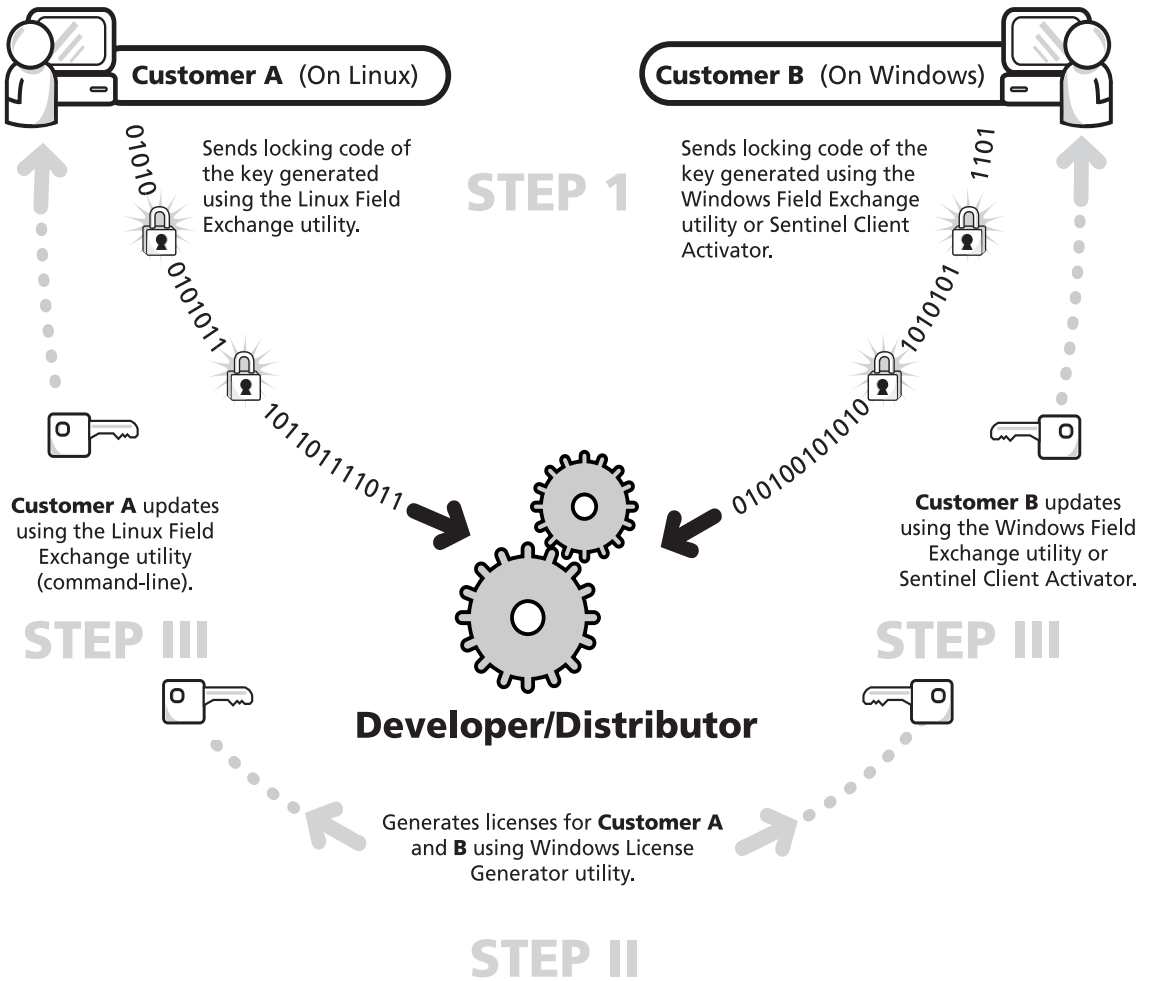
How Do Linux and Windows Users Obtain Licenses in the Field?

Configuring the Windows Toolkit

Before you can use Sentinel SuperPro, you must configure your Toolkit for your specific information, the developer ID, overwrite passwords, write password and *secret* code. You only need to enter the details for the first time as they are remembered for subsequent sessions. To configure the Toolkit, follow the steps given below:

1. Point to **Programs > Rainbow Technologies > SuperPro > 6.3 > SuperPro Toolkit** from the **Start** menu. The **Developer Configuration** dialog appears.

2. Now, you need to fill in your developer ID and passwords. These passwords and your developer ID are provided by Rainbow Technologies, and can be found on the password sheet included in your Sentinel SuperPro package. The developer ID is a unique identification code that must be given for establishing a connection with your key. Remember to keep these passwords secure, as they give the power to modify information present in your key!

3. Now, you must decide whether to include the overwrite passwords in your field exchange DLLs or not. The overwrite passwords are necessary whenever you want to change the value of a locked data word or read-only cell on the key. Also, to implement the **One-time update option** for license codes, you must include the overwrite passwords in your field exchange DLLs. Though these passwords are encrypted, Rainbow Technologies recommends against including them. This is because, if decrypted by any means, unauthorized access to the key can be gained, endangering your application's security.

   To include the overwrite passwords in the field exchange DLLs, select the **Include overwrite passwords in DSAFE.DLL and USAFE.DLL** check box. A warning message appears. Read the warning message, then click **OK**. The **One-time only license code update option** is automatically activated.

4. Now, specify a secret code in the **Secret Code** field. It must be 9 to 16 characters long string, consisting of at least one numeric character. Otherwise, you may choose to autogenerate the secret code, by clicking the **Auto Generate** button. In either case, write down the string somewhere as you will need it to update clients on all platforms.
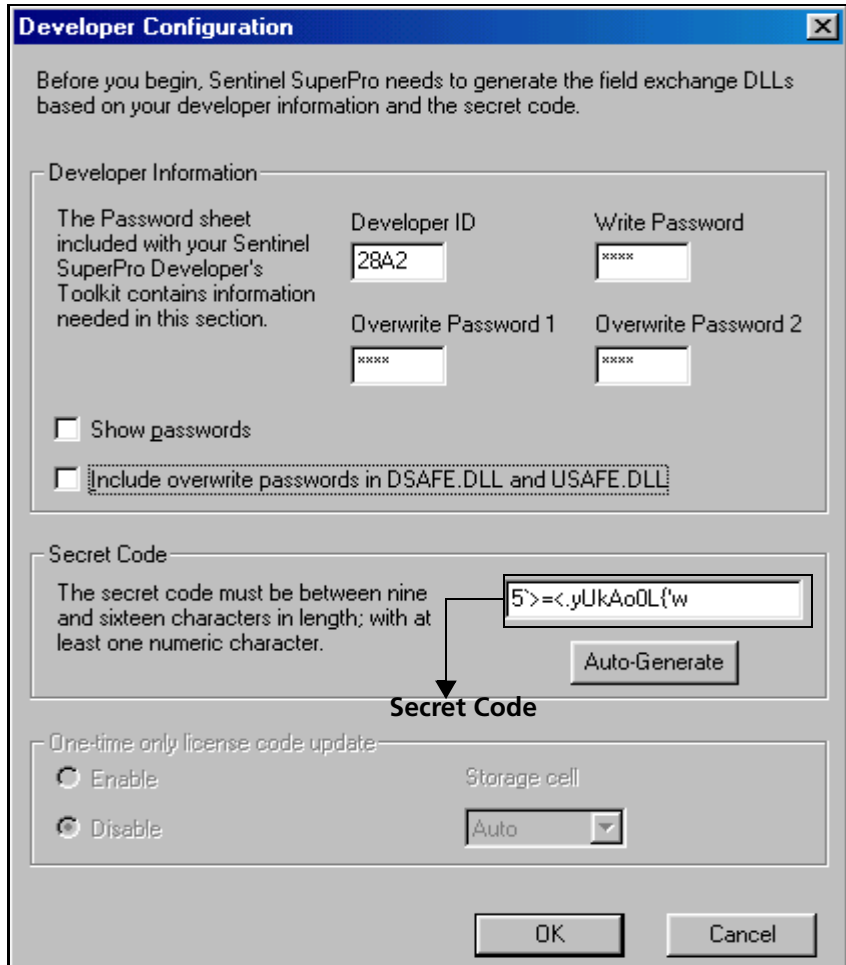
Configuring Toolkit with Developer Information and Secret Code

---

Tip: *Unlike your passwords, the secret string remains visible because you need to refer to it. To safeguard it from being viewed by an unauthorized person, you must lock your projects.*

---

5. The one-time update option allows you to prevent license codes from being applied to a key more than once. A single cell is used to determine the validity of a license code. If the value in the cell is not what is expected by the license code, the license code cannot be applied again.

   If you have included the overwrite passwords in your field exchange DLLs (see step 3), select **Enable**. A warning message appears. Read the warning message, then click **OK**. Choose a desired cell from the **Storage cell** drop-down list, which only shows the free cells available in the key.

6. Now, click **OK.** If you have already created projects with the Toolkit, be sure to have carefully read the warning message that appears and take an appropriate action.

Note: *If you do not wish to update the key in the field, then there is no need to use the same secret code while configuring SAFE objects on other platforms. However, if you do wish to do field updates, be sure to keep the same configuration settings. The SAFE objects are configured for all the information—not just the secret code—you provide while configuring the Toolkit. In any case, you cannot skip this secret code field by leaving it empty or filling with invalid data.*

Configuring the SafeCfg Utility for Red Hat Linux

SafeCfg is a command-line utility that is configured with *developer-specific* information and a secret code to generate a matching pair of DSAFE and USAFE objects. The USAFE object (*libusafe.so*) will be shipped along with your protected application.

Follow the steps given below to configure the SafeCfg utility:

1. SafeCfg is located in the */opt/RainbowTechnologies/SuperPro/6.3/Tools* path. Execute the SafeCfg utility by typing **SafeCfg** at the Shell command prompt.

2. Now, you need to fill in your developer ID and passwords in the order of appearance. Read the instructions in the section "Configuring the Windows Toolkit" on page 60 for more information. Ensure that you choose the same settings to configure the SAFE objects.

3. You will be now asked to either autogenerate the secret code or type on your own. The secret code is a parameter that enables field updates across multiple platforms. To use the autogenerated string, type **y**, or type **n** and specify a secret. A secret string must be 9 to 16 characters long, consisting of at least one numeric character.

4. You will be asked to include the overwrite passwords or not. Decide and act appropriately.

5. You may be asked to specify a one time update cell, only if you have included your overwrite passwords in step 4 earlier. Type a cell address in hexadecimal.

Tip: *You could use the **Memory View** option in Toolkit to decide which cell to select.*

6. The *libusafe.so* and *libdsafe.so* are created in your current directory.

Appendix A

Documentation Errata

This appendix contains information on corrections to the existing Sentinel SuperPro manuals and Online Help.

Sentinel SuperPro System Administrator's Guide

The *Sentinel SuperPro System Administrator's Guide* for the 6.1.1 release requires the following corrections:

Instructions for Setting the NSP_HOST Environment Variable on Windows 9x

On page 20, the instructions for setting the NSP_HOST environment variable are provided for Windows NT only. To set this variable on Windows 9x, follow the steps below:

1. Type **Set NSP_HOST=<value>** in the *autoexec.bat* file, where the value can be set to `RNBO_SPN_DRIVER`, `RNBO_SPN_LOCAL`, `RNBO_SPN_BROADCAST`, `RNBO_SPN_ALL_MODES`, `RNBO_SPN_SERVER_MODES`, or the Sentinel SuperPro server's host name, IP or IPX address. See "Set Access Mode Using NSP_HOST" on page 24 for more information.

2. After saving the changes, reboot your system. Unlike setting through the command prompt, this method frees you from resetting the variable each time the protected application is run.

Sentinel SuperPro Developer's Guide

■ On page 103, in the *Sentinel SuperPro Developer's Guide*, the RNBOsproSetContactServer has been misspelled as RNBOsproGetContactServer in the following instances.

- The third bulleted item under the "Finding a Key" heading should read as "RNBOsproSetContactServer (optional)" and not "RNBOsproGetContactServer (optional)."

- Later on this page, the fifth line from the bottom should contain RNBOsproSetContactServer instead of RNBOsproGetContactServer.

■ On page 158, the description of the **Execution Control** and **Time Control** option should read "data word" instead of the "counter cell." Similarly, the first line on page 157 should read "For execution and time control, these cells are programmed as data word," and not "For execution and time control, these cells are programmed as counters."

■ The index entry on page 383 of *Sentinel SuperPro Developer's Guide*, should read "RNBOsproOverwrite" and not "RNBOsproOverWrite."

Appendix B

Sentinel SuperPro Error Codes

This appendix enlists all the API function status codes and the error codes for automatic protection (Shell).

Note: *For Sentinel Client Activator error codes, refer to its HTML Help.*

API Status Codes

The following table describes the status codes an API function can return to your application:

API Function Status Codes

Status Code (Decimal)	Description
0	**SP_SUCCESS** The function completed successfully.

API Function Status Codes (Continued)

Status Code (Decimal)	Description
1	**SP_INVALID_FUNCTION_CODE** You specified an invalid function code. See your language's include file (for example, SPROMEPS.H) for valid API function codes. Generally, this error should not occur if you are using a Rainbow-provided interface to communicate with the driver. However, may occur when a stand-alone-only function is used in a network situation.
2	**SP_INVALID_PACKET** A checksum error was detected in the command packet, indicating an internal inconsistency. The packet record has not been initialized, or may have been tampered with. Generally, this error should not occur if you are using a Rainbow-provided interface to communicate with the driver.
3	**SP_UNIT_NOT_FOUND** Either RNBOsproFindFirstUnit or RNBOsproFindNextUnit could not find the specified Sentinel SuperPro key. Make sure you are sending the correct developer ID. This error is returned by other functions if the key has been removed.
4	**SP_ACCESS_DENIED** You attempted to perform an illegal action on a cell. For example, you may have tried to read an algorithm/hidden cell, write to a locked cell, or decrement a cell that is not a data or counter word.
5	**SP_INVALID_MEMORY_ADDRESS** You specified an invalid Sentinel SuperPro memory address. Valid addresses are 0–63 decimal (0–3F hex). Cells 00–07 are invalid for many operations. Algorithms must be referenced by the first (even) address.
6	**SP_INVALID_ACCESS_CODE** You specified an invalid access code. The access code must be 0 (read/write data), 1 (read-only data), 2 (counter) or 3 (algorithm/hidden).

API Function Status Codes (Continued)

Status Code (Decimal)	Description
7	**SP_PORT_IS_BUSY** The requested operation could not be completed because the port is busy. This can occur if there is considerable printer activity, or if a unit on the port is performing a write operation and is blocking the port. Try the function again.
8	**SP_WRITE_NOT_READY** The write or decrement could not be performed due to a momentary lack of sufficient power. Try the function again.
9	**SP_NO_PORT_FOUND** No parallel ports could be found on the local workstation or network server, or there was a problem with the protocol being used on the network.
10	**SP_ALREADY_ZERO** You tried to decrement a counter or data word that already contains the value zero. If you are using the counter to control demo application executions, this condition may occur after the corresponding algorithm has been reactivated with its activation password.
12	**SP_DRIVER_NOT_INSTALLED** The system device driver was not installed or detected. Communication to the unit was not possible. Verify the device driver is correctly installed.
13	**SP_IO_COMMUNICATIONS_ERROR** The system device driver is having problems communicating with the unit or the network server. Verify the device driver is correctly installed.
15	**SP_PACKET_TOO_SMALL** The memory allocated for the SuperPro packet is less than the required size.
16	**SP_INVALID_PARAMETER** Arguments and values passed to the SuperPro APIs are invalid.

API Function Status Codes (Continued)

Status Code (Decimal)	Description
18	**SP_VERSION_NOT_SUPPORTED** The current system device driver is outdated. Update the driver.
19	**SP_OS_NOT_SUPPORTED** The operating system or environment is not supported by the client library. Contact Technical Support for assistance.
20	**SP_QUERY_TOO_LONG** You sent a query string longer than 56 characters. Send a shorter string.
21	**SP_INVALID_COMMAND** An invalid SuperPro command was specified in the API call.
30	**SP_DRIVER_IS_BUSY** The system driver is busy. Try the function again.
31	**SP_PORT_ALLOCATION_FAILURE** Failure to allocate a parallel port through the operating system's parallel port contention handler.
32	**SP_PORT_RELEASE_FAILURE** Failure to release a previously allocated parallel port through the operating system's parallel port contention handler.
39	**SP_ACQUIRE_PORT_TIMEOUT** Failure to acquire access to a parallel port within the defined time-out.
42	**SP_SIGNAL_NOT_SUPPORTED** The particular machine does not support a signal line. For example, an attempt may have been made to use the ACK line on an NEC 9800 computer.
57	**SP_INIT_NOT_CALLED** The key is not initialized. Call the RNBOsproInitialize function before calling the function that generated this error.

API Function Status Codes (Continued)

Status Code (Decimal)	Description
58	**SP_DRIVER_TYPE_NOT_SUPPORTED** The type of driver access, either direct I/O or system driver, is not supported for the defined operating system and client library.
59	**SP_FAIL_ON_DRIVER_COMM** The client library failed on communicating with the Rainbow system driver.
60	**SP_SERVER_PROBABLY_NOT_UP** The server is not responding; the client has timed out.
61	**SP_UNKNOWN_HOST** The server host is unknown. The server is not on the network, or an invalid host name was specified.
62	**SP_SENDTO_FAILED** The client was unable to send a message to the server.
63	**SP_SOCKET_CREATION_FAILED** Client was unable to open a network socket. Make sure the TCP/IP or IPX protocol stack is properly installed on the system.
64	**SP_NORESOURCES** Could not locate enough licensing requirements. Insufficient resources (such as memory) are available to complete the request, or an error occurred in attempting to allocate needed memory.
65	**SP_BROADCAST_NOT_SUPPORTED** Broadcast is not supported by the network interface on the system.
66	**SP_BAD_SERVER_MESSAGE** Could not understand a message received from the server. An error occurred in decrypting (or decoding) a server message at the client end.

API Function Status Codes (Continued)

Status Code (Decimal)	Description
67	**SP_NO_SERVER_RUNNING** Cannot talk to server. Verify the server is running. Server on specified host may not be available for processing the request.
68	**SP_NO_NETWORK** Unable to talk to the specified host. Network communication problems encountered.
69	**SP_NO_SERVER_RESPONSE** There is no server running in the subnet, or no server in the subnet has the desired key connected.
70	**SP_NO_LICENSE_AVAILABLE** All licenses are currently in use. Server has no more licenses.
71	**SP_INVALID_LICENSE** License is no longer valid. License expired due to time-out.
72	**SP_INVALID_OPERATION** Specified operation cannot be performed. Tried to set the contact server after obtaining a license, or tried to call FindFirstUnit function on a packet that already has a license.
73	**SP_BUFFER_TOO_SMALL** The size of the buffer is not sufficient to hold the expected data.
74	**SP_INTERNAL_ERROR** An internal error, such as failure to encrypt or decrypt a message being sent or received, has occurred.
75	**SP_PACKET_ALREADY_INITIALIZED** The packet being initialized was already initialized.
76	**SP_PROTOCOL_NOT_INSTALLED** The protocol specified is not installed.
77	**SP_API_NOT_SUPPORTED** This API is not supported.

Automatic Protection (Shell) Error Codes

The table below lists the error codes for automatic protection. The error codes from 0 to 10 may be encountered during the execution of a protected application, while the rest of them may appear during the protection process. The runtime error messages can be customized in the **Design** stage if you have selected the **Show advanced options** check box.

Automatic Protection (Shell) Error Codes

Status Code (Decimal)	Description
0	**ERR_OK** Success.
1	**ERR_NO_KEY** The message appears if there is a problem accessing the key. For example, the key is not attached.
2	**ERR_LOAD_ERROR** The message appears if there is a problem loading the protected application. Try loading the application again.
3	**ERR_DATE_STOP** The message appears when the application is started and the current date is equal to or greater then the expiration date stored inside the attached key.
4	**ERR_NO_MORE_LIFE** The number of executions allowed for the application have exhausted.
5	**ERR_MEM_ERROR** Memory allocation error. Could not allocate memory to protect the application.
6	**ERR_MEM_ACCESS** Memory access error or trying access an invalid memory location.
7	**ERR_LIBLOAD** Error in loading your application's import library. The DLL required by your application could not be loaded.

Automatic Protection (Shell) Error Codes (Continued)

Status Code (Decimal)	Description
8	**ERR_BADIMPORT** Error importing library function. The imported function is not found in the specified DLL.
9	**ERR_TOO_MANY_KEYS** The message appears if there is more than one key attached. Only one hardware key is allowed.
10	**ERR_CANT_START_CA** Error opening Client Activator program. The Sentinel Client Activator could not be launched.
128	**ERR_BAD_ALGO** Invalid/inactive algorithm programmed in the key.
129	**ERR_LONG_MSG** The message text is very long. You can modify the message under the Design stage in Toolkit.
130	**ERR_READ_ERROR** Application not found or the file could not be read. Check the path specified for your application.
131	**ERR_NOT_ENOUGH_MEMORY** Not enough resources to protect the application. Try closing some of the applications running on your system.
132	**ERR_CANNOT_OPEN** Error in opening the application. Make sure that application is not read-only and the correct path is specified.
133	**ERR_WRITE_ERROR** Could not write to the application. not read only
134	**ERR_CANNOT_OVERWRITE** Could not write to the application. May be your overwrite flag is not checked.

Automatic Protection (Shell) Error Codes (Continued)

Status Code (Decimal)	Description
137	**ERR_INVALID_HEADER** Invalid file header. Ensure that the application header is in PE format.
138	**ERR_TMP_CREATE_ERROR** Cannot create a temporary file.
140	**ERR_PATH_NOT_THERE** The path for output file specified does not exist. Create a directory.
141	**ERR_OUT_OF_RANGE** Some invalid value found during binary conversion.
142	**ERR_NUMBER_EXPECTED** A numeric value was expected.
143	**ERR_NOT_WINFILE** The application is not a Windows application.
148	**ERR_BAD_FILESPEC** Invalid file name specified for the data file to be encrypted.
149	**ERR_SYNTAX** Syntax error found in the command-line option.
150	**ERR_BAD_OPTION** Invalid command-line option specified.
151	**ERR_TOO_MANY_FSPECS** Too many files specified for the encrypt additional files option.
156	**ERR_TOOMANYFILES** Too many files specified for the encrypt additional files option.
157	**ERR_BAD_FILE_INFO** This file cannot be protected. Check out the eligibility criteria for protecting an application.

Automatic Protection (Shell) Error Codes (Continued)

Status Code (Decimal)	Description
158	**ERR_NOT_WIN32FILE** Not a valid Win32 file.
159	**ERR_INVALID_MACHINE** The application is not developed on a Intel x86 platform.
160	**ERR_INVALID_SECTION** The section specified in the file header could not be found in the application.
161	**ERR_INVALID_RELOC** The relocation specified for your application could not be found, or has invalid address.
162	**ERR_NO_PESHELL** The application does not have a PE (Portable Executable) format.

Appendix C
Frequently Asked Questions

This section contains possible questions that you may have while protecting your application with SuperPro. If you have any other questions, contact Rainbow Technical Support using the information provided on page xii.

Protecting Applications Using SuperPro

Question 1. What types of applications can be protected using the automatic protection (Shell) option?

Answer: You must refer to the Appendix C - Compatible Compilers and Applications in the *Sentinel SuperPro's Developer's Guide*. The same list applies to the quick automatic protection option.

Question 2. Will my application protected with the automatic protection (Shell) option work over the Internet?

Answer: Yes, it works with all the TCP/IP supported networks, but not with any other protocol. You should choose the network access modes and ask your customers to set the NSP_HOST variable to the name of the computer where the SuperPro server is running and has the key attached.

Question 3. Can I open and edit an *.SPP* file made with an older version of Toolkit?

Answer: You can open an old SuperPro project file (.*SPP*) in the new SuperPro Toolkit. You can also program the hardware keys until the project is

saved. This is because, to save such projects you must configure the SAFE DLLs (usafe and dsafe) for the developer information including the *secret code*—an additional parameter required for platform-independent field activation. Doing so will overwrite your previous pair of DLLs.

Question 4. I do not find Sentinel Client Activator installed for automatic protection. Why is that?

Answer: Sentinel Client Activator is not installed by default. A separate installer has been included in the *\Ca22* directory of the Sentinel SuperPro installation CD.

Question 5. What are the installer products I can use with the merge modules of SuperPro redistributables?

Answer: You can use InstallShield Developer 7.02 or higher and InstallShield for Windows Installer.

SuperPro Server

Question 1. What value will be returned by the protocol member of the NSPRO_MONITOR_INFO structure when the SuperPro server is running on multiple protocols?

Answer: 3 will be returned when the SuperPro server is running on the TCP and IPX protocols.

Question 2. Which protocols does the SuperPro server support?

Answer: The SuperPro server/service will either run on NetBEUI or on TCP/IP and IPX. Therefore, the client that wishes to contact the server must have set its protocol accordingly.

SAP uses IPX as the underlying protocol for communication *except* for network broadcast.

Question 3. My Novell machine has both TCP/IP and IPX. However, my TCP client cannot access the key on this machine. Why not?

Answer: This is because the SuperPro server on the Novell platform supports only IPX.

Question 4. For customers working on Windows XP, can I supply *spnsrv9x. exe* for installing the SuperPro server?

Answer: No. You should ship the SuperPro service *(spnsrvnt.exe)* instead.

Question 5. What is the UDP port which the SuperPro server uses?

Answer: 6001.

Question 6. How do I install the SuperPro server in a Firewall protected network?

Answer: To do so, you should open the 6001 port in the Firewall for communicating with the SuperPro server using the Internet.

Question 7. I set IPX protocol in my Monitoring Tool, but I find few of my Novell servers responding to it. Why is this?

Answer: This mismatch can be because Novell 4.2 needs the Service Advertising Protocol (SAP) protocol, whereas the Novell 5.0 platform may respond to the IPX broadcast as well.

Question 8. What are the contents of the transaction/usage log file created for the Sentinel SuperPro server?

Answer: The transaction or usage log file records information about the clients accessing the server and the licenses being used. The table below provides information about its contents:

The Transaction Log File Contents

Code	Item	Description
1.	Version	Version of the Sentinel SuperPro software, say v6.3.

The Transaction Log File Contents (Continued)

Code	Item	Description
2.	Operation Code (any of the following)	The operation that took place.
	1 = issue license	License issued.
	2 = release license	License released.
	3 = time-out	Client's failure to update.
	4 = key access	Refers to the key access error such as "Key not found."
	5 = startup	Server started.
	6 = shutdown	Server shutdown.
3.	Time (in ticks)	System time in ticks such as 1020336363.
4.	Time (as string)	System time as a string such as: Tue Jan 01 00:00:01 2003.
5.	Key Index	Serial number of the key (hex value in cell number zero).
6.	Cell Number	Refers to the sublicense cell number. A value "0" indicates that only the main license (and not sublicense) is acquired.
7.	Host Name	The host name of the client machine.
8.	User Name	The user name of the client machine on the network.
9.	Client Process ID	The process ID of the client application.
10.	License Key	The hard limit of the key.
11.	License(s) in Use	The total number of licenses that are being used currently.
12.	Return Code	Operation return code.

Using the Protected Applications

Question 1. Under what conditions should I use a USB key instead of the parallel?

Answer: The universal serial bus is a low cost, plug and play, standardized interface included on virtually all personal computers built since early 1997. All future PCs are certain to have USB ports, while the widespread availability of other token interfaces remains uncertain.

We recommend using USB keys on all computers running on Windows 98, ME, 2000, XP, Linux, and Macintosh. The USB keys not only provide simple and quick installation for a user, but are designed for maximum durability and *ease-of-use*— enabling them to be inserted and removed repeatedly.

The Sentinel USB driver can support a maximum of 32 keys attached to a port/hub.

Question 2. Can I install the USB driver on my laptop that doesn't have a parallel port?

Answer: Yes. To meet the needs of such computers, a change in the installation of the Sentinel Driver has been made. As a result, it is now possible to install just the USB driver.

Question 3. Once I have acquired a license, then is it possible for my application to lose its license despite regular updates to the SuperPro server?

Answer: Yes. When the heartbeat time interval set is too low, and traffic over network is heavy, then the application can terminate even before contacting the SuperPro server. To avoid such a situation, you may increase the heartbeat interval.

Question 4. I have already installed SuperPro on my system. Why would I need Combo Installer?

Answer: Combo Installer installs the SuperPro redistributables—Sentinel System Driver, SuperPro server and Monitoring Tool—jointly on a system. These components are necessary to use the networked, protected applica-

tions. See "Combo Installer for Windows" on page 34 to learn more about various supported installation options.

Question 5. I have not installed the Data Protection Driver, but my encrypt/decrypt works fine. Why is that?

Answer. It needs to be installed only on Windows 9x computers; you might be using a Windows NT/2000/XP system. See the document available in the *\Data Protection Driver* directory of the installation CD for installation instructions.

Question 6. Can I use VMWare to access a SuperPro key across two operating systems?

Answer: No.

Appendix D

Changes to Sentinel SuperPro 6.2

This appendix contains information about the changes in Sentinel SuperPro 6.2. The topics covered in this appendix are:

- "New APIs Added to the 6.2 Client Library" on page 83.
- "SuperPro Server on Novell" on page 85.
- "Support for NetBEUI Added" on page 87.
- "SuperPro Server Icon Display Made Optional" on page 88.
- "SuperPro Server Messages Suppression Allowed" on page 88.

New APIs Added to the 6.2 Client Library

Two new network APIs, RNBOsproSetProtocol and RNBOsproSetHeartBeat, were introduced in the Sentinel SuperPro 6.2 release.

RNBOsproSetProtocol

This API is used to set the protocol between the SuperPro server and a client. It needs to be called after initializing the packet and before the RNBOsproFindFirst API. If this API is not used, then TCP/IP runs by default.

Format

```
unsigned short int RNBOsproSetProtocol (
                    RBP_SPRO_APIPACKET   packet,
                    PROTOCOL_FLAG        protocol);
```

Parameters

packet	A pointer to the RB_SPRO_APIPACKET structure.
protocol	The communication protocol between the client and the SuperPro server. Use any of the following values:

NSPRO_TCP_PROTOCOL	= 1
NSPRO_IPX_PROTOCOL	= 2
NSPRO_NETBEUI_PROTOCOL	= 4
NSPRO_SAP_PROTOCOL	= 8

Tip: *SAP (Service Advertising Protocol) is used for finding the key plugged into the computer containing the Novell server through broadcast only.*

Return Values

If successful, the function returns SP_SUCCESS(0). If an error occurs, the function returns one of the status codes listed in "API Status Codes" on page 67.

RNBOsproSetHeartBeat

This API helps you in customizing the heartbeat time interval of a client. It needs to be called after the RNBOsproFindFirst API has been called.

The time can be set to INFINITE_HEARTBEAT or from 1 minute to 30 days, as multiples of 1 second. If set as INFINITE_HEARTBEAT, then the heartbeat will have an infinite value. Hence, the SuperPro server will not release license for this client, unless the RNBOsproReleaseLicense API is called.

If the RNBOsproSetHeartBeat API is not called, then the default setting remains as 120 seconds.

Format

```
unsigned short int RNBOsproSetHeartBeat (
                  RBP_SPRO_APIPACKET   packet,
                  RB_DWORD             heartBeatValue);
```

Parameters

packet   A pointer to the RB_SPRO_APIPACKET structure.

heartBeatValue A value that represents time in seconds. The valid range is:

MAX_HEARTBEAT   = 2592000

MIN_HEARTBEAT   = 60

INFINITE_HEARTBEAT   = 0xFFFFFFFF

Return Values

If successful, the function returns SP_SUCCESS(0). If an error occurs, the function returns one of the status codes listed in "API Status Codes" on page 67.

SuperPro Server on Novell

It is now possible for a Windows client to communicate with the SuperPro server running on a Novell system.

Note: *The server has a built-in Sentinel System Driver that allows you to directly communicate with the parallel key. However, it does not support the USB version of the SuperPro keys.*

Starting SuperPro Server on Novell

You can load the SuperPro server program using either of the two options given below:

Note: *Before configuring the SuperPro server, ensure that the* WS2_32.nlm *module is loaded on the Novell file server.*

Using Command Prompt

You can run the SuperPro server manually from the command prompt as follows:

1. Copy the SuperPro server file *spnnwsrv.nlm* to the file server.

2. Attach the SuperPro key to the parallel port.

3. Now, load the SuperPro server by typing **load spnnwsrv.nlm** at the command prompt. If the server file is not available in the current directory or in the search path, then specify the full pathname, such as **load sys:\xyz\spnnwsrv.nlm**. This loads the SuperPro server from the *xyz* directory.

Using AUTOEXEC.NCF

When set through *AUTOEXEC.NCF*, the SuperPro server runs automatically every time the file server boots up. To do so:

1. Copy the SuperPro server program (*spnnwsrv.nlm*) to the file server.

2. Type **load spnnwsrv.nlm** into the *AUTOEXEC.NCF* file.

3. Attach the SuperPro key to the parallel port.

4. Now, reboot the system.

Stopping SuperPro Server on Novell

To stop the SuperPro server, type **unload spnnwsrv.nlm** at the command prompt.

Note: *IPX clients can be served by a SuperPro server running on any Windows 95/98/ME (spnsrv9x) or Windows NT/2000/XP (spnsrvnt) system. They do not require the NLM or application on the NetWare server. However, "Client for Novell Network" must be installed to allow communication between a Windows 9x client and the Novell file server. It is recommended that Windows NT clients set the IPX frametype property to **Ethernet 802.2** for successful communication with the Novell server.*

Support for NetBEUI Added

Sentinel SuperPro provides NetBEUI protocol support to the client library and server. This particularly facilitates migration of NetBEUI users of NetSentinel to Sentinel SuperPro 6.2 or higher.

Tip: *Developers migrating from Sentinel SuperPro 5.1 and NetSentinel should refer to* Chapter 16, "Migrating From Sentinel SuperPro 5.1 or NetSentinel" *of the* Sentinel SuperPro Developer's Guide.

Starting the Server in NetBEUI

To start the SuperPro server in NetBEUI mode, type `Spnsrv9x NETBEUI` at the command prompt. For starting the server in TCP/IP, no command-line arguments are required.

Note: *If the SuperPro server is set to NetBEUI, then it will only use the NetBEUI protocol and not TCP/IP or IPX.*

Starting the Service in NetBEUI

Before starting the service, ensure that the following values are present in the HKEY_LOCAL_MACHINE\SOFTWARE\Rainbow Technologies\SentinelSuperPro\ registry key:

Registry Entry for Starting the Service in NetBEUI Mode*

Key Name	Data Type	Value
Protocol	REG_SZ	NETBEUI

* The entries are not case-sensitive.

If the service has to start in TCP/IP mode, then the value of the above key will be TCPIP instead of NETBEUI. If this key name is not present in the registry, then the service will automatically start in TCP/IP mode.

SuperPro Server Icon Display Made Optional

The developers can now either show or hide the SuperPro server icon that appears in the System Tray when the server is running on Windows 9x. By default, the icon is hidden to prevent end users from stopping the Sentinel SuperPro server. The icon can be controlled through the "Show9xServerIcon" entry in the HKEY_LOCAL_MACHINE\Software\RainbowTechnologies\SentinelSuperPro registry key:

- To show the icon, change the value of Show9xServerIcon to 1.

- To hide the icon, change the value of Show9xServerIcon to 0 (default).

SuperPro Server Messages Suppression Allowed

You can choose to suppress/display the SuperPro server messages on your end user's computer in any of the following ways:

■ **Use a Command-Line Argument**–A command-line argument `spnsrv9x /msg=<options>` can be used to determine if the messages will be shown or not. In the argument, the option can be either ON or OFF (to display or hide the server messages respectively).

■ **Modify the ShowServerMessages Registry Entry**–If the command-line argument is not used, the "ShowServerMessages"

registry entry of DWORD type can be modified in the HKEY_LOCAL_MACHINE\SOFTWARE\Rainbow Technologies\SentinelSuperPro key to show or hide the messages. The installer initially sets the value to 0, but it can be changed to 1. 1 means that the messages will be displayed on the user terminal and 0 means that the messages will be hidden.

- **Modifying the Show9xServerIcon Registry Entry**–If none of the options given above are used, then the "Show9xServerIcon" registry entry of DWORD type can be modified in the HKEY_LOCAL_MACHINE\SOFTWARE\RainbowTechnologies\Sentin elSuperPro key. The entry can take a value of 1 or 0, where 1 means that the messages will be shown to the user terminal and 0 means that the messages will be hidden.

Index