User's Manual

Table of Contents

Overview

The **SecuredApp** software is a security post-processor that creates self-protecting applications to prevent piracy, malicious tampering, and access to sensitive Intellectual Property resident within the software application. The protected application can:

- At program startup, detect any modified system or application **DLL**s (dynamic link libraries)
- Minimize access to the decryption application at runtime to **protect against reverse engineering**
- Prevent attachment of debuggers to the protected application
- Detect when runtime changes have been made to the protected code
- Prevent malicious tampering of application files, statically or at runtime
- Respond to application tampering attempts by terminating execution or attempting to repair itself and continue execution if runtime tampering is detected
- Deny the ability to abort the application externally
- Restrict execution using alternate authentication to the normal system and software authentication

The SecuredApp Post Processor takes an existing executable file and associated DLLs as input and creates a protected version of the same application as output. It does not require access to source code to protect an application.

SecuredApp runs in four phases:

- The **Registration Wizard** phase integrates the application with the billing system and applies licenses and prices to the application, allowing you to control how the application is used by the consumer
- The **Using SecuredApp** phase determines the level of protection that you want to have on the application and its associated DLL files and functions
- The **Installer** phase is completed by your development team as they create an installer for the application and place the application for distribution
- The **Post-Distribution** phase occurs after the development cycle has completed and consumers can access the protected application. This phase allows you to continue to manage the license, price, and application, even after the consumer has purchased the product

Overview

You have downloaded **SecuredApp** software. This section provides the following:

- An overview of the functions performed by the SecuredApp program
- The Minimum Requirements for installation of the SecuredApp program
- Installing the SecuredApp program

- Repairing the SecuredApp program should the need arise
- Uninstalling the SecuredApp program

What SecuredApp Does

SecuredApp is sometimes referred to as a Post-Processor, because all of the encryption is integrated with your application *after* your development cycle is completed.

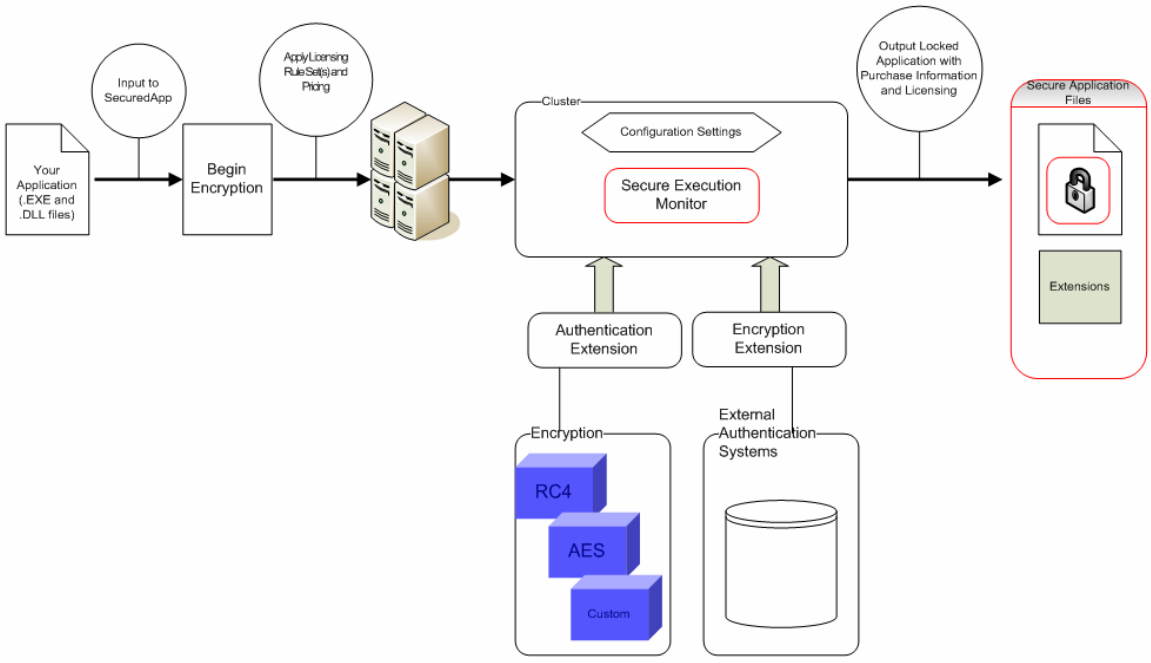
Before encryption is complete, SecuredApp will take you to the Registration Wizard so that you can configure pricing and license variables that control how your application is used by consumers.

SecuredApp then analyzes the application, encrypts selected portions of it (tailored by you), adds a runtime authenticator and decrypter (Execution Monitor), and writes the protected version of the application to a new directory.

After completing these processes, you can create an installer using the protected application files. It will launch and runs in the same way as the original unprotected application after the license is acquired.

When the protected application is launched by a consumer, the embedded SecuredApp security functions take control of the application execution, verifying the license and lifespan of the application. After receiving the keys, the application disconnects from the server and the protected application runs autonomously. The default configuration of the post-process protection does not require authentication and stores the decryption securely within the protected application.

Once SecuredApp protection process is in control of the application execution, it carefully manages the decryption and re-encryption of application functions at run-time as well as continually monitoring the environment for debuggers, virtual machines, and other processes that can be used to analyze the protected application, protecting your source code from malicious intent.



Minimum Requirements

Installing SecuredApp post-processor software requires no special knowledge of your network or computer configuration.

For SecuredApp, you need a computer with the following minimum requirements:

- 2GHz Pentium 4 or higher processor
- 512Mb of RAM or more
- 1Gb free hard disk space
- Windows 2000, XP or XPe with current service packs
- Internet connection
- Printer, if you want a paper copy of the license agreement

Note: SecuredApp supports the current Windows PE executable file format, but it does not support older versions, such as Windows 3.1 and OS/2 or DOS executable file formats. SecuredApp will notify you if a program that you are trying to protect is of an unsupported file type.

Installation

To install SecuredApp:

1. Double-click the **setup.exe** file in the directory where you downloaded the installation file.
2. Click **Next>** after the **SecuredApp** screen loads.
3. Click **Next>** after the **InstallShield Wizard** screen loads.
4. Read the **End User License Agreement (EULA)**. Print the EULA if desired. Click to select the **I accept** radio button.

   Choosing the **I do not accept the terms in the license agreement** option will not allow you to go forward with the installation. See the **License Agreement** section**License Agreement** topic for more information.
5. Click **Next>** to begin installation.
   Clicking Cancel on any dialog terminates the installation without making any changes to your computer. If you cancel, you see a confirmation dialog before installation aborts.
6. Determine which users can access the SecuredApp Post-Processor.
   - Click to select the Only for me (<user name>) option to disallow other users from the application
   - Click to select **Anyone who uses this computer (all users)** to allow all users of the computer to access the application
8. Click **Next>** to begin installation.
9. To accept this default setting, click **Next**. Installation of SecuredApp defaults to the \Program Files\SECUREDAPP directory.

If you would like to install in a different directory:

a. Click **Change** in the dialog.

b. When the Change Current Destination Folder dialog displays, navigate to the directory.

c. Click **OK**.

10. At the Ready to Install the Program window you may review your installation choices. To make any changes, click **Back**.

11. Click **Install** to complete the installation process.

When the **Wizard Completed** dialog appears, view the **readme** file by ensuring the **Show the readme file** box is checked and clicking **Finish**. The read me file details changes to the software from the previous release.

Repair

To repair SecuredApp:

1. In Windows, click **Start>Control Panel**.
2. Double-click the **Add or Remove Programs** icon.
3. Locate **SecuredApp** in the list of currently installed programs and click to highlight.
4. Click **Change**.
5. Click **Next>** when the **SecuredApp - InstallShield Wizard** window opens.
6. Click to select the **Repair** radio button.
7. Click **Next>**.
8. Verify the settings and click **Install**.
9. Click **Finish** when the wizard completes.

Uninstall

To uninstall SecuredApp:

1. In Windows, click **Start>Control Panel**.
2. Double-click the **Add or Remove Programs** icon.
3. Locate **SecuredApp** in the list of currently installed programs and click to highlight.
4. Click **Remove**.
5. Click **Yes** when asked **Are you sure you want to remove SecuredApp from your computer?**

SecuredApp is removed from your computer.

Using SecuredApp

There are five phases involved with protecting an application.

- The first phase of the SecuredApp process is the Post-Processor. In this section you will launch the application, choose the application that you wish to protect and associated functions and DLL files that you wish to encode, complete the Registration Wizard phase, and complete the encoding of the application.
- The second phase runs within the first phase. This phase is called the Registration Wizard phase. During this phase the application is registered in the Admin System and assigned licensing rights and pricing options. After completing this phase, there will be one more step to complete in the Post-Processor phase before you can move on.
- The third phase happens with your development team. You will pass the encrypted application back to your development team so that they can create an installer for the application.
- After the installer has been created, it is time for the Distribution Phase. In this phase you begin to distribute the application.
- The Post-Distribution phase is simply a reminder that you still have full control over the application, its cost, and associated licensing rules. You can make changes to an application at any time by simply logging in to the Admin System.

Launch SecuredApp

Double-click the SecuredApp icon on the desktop to start the program. Alternately, you can launch the program from the **Start** menu in Windows by choosing **All Programs>SecuredApp>SecuredApp** and clicking on **Launch SecuredApp.exe**).

Configure Input and Output Parameters

Configure the source and destination files, checkpoint preferences, and LogFile preferences:

1. Enter the path to the file to be protected (or click **Browse** to locate the file through **Windows Explorer**). Ex.: C:\DownloadedProgramFiles\test-setup.exe
2. Enter the path and file name to be created during the protection process (or click **Browse** to locate the file through **Windows Explorer**). Ex.: C:\DownloadedProgramFiles\Output\test-setup.exe
3. Click to select **Create a** checkpoint file if you wish to create a checkpoint (*.cpt) file (see glossary for further information about checkpoint files).
4. Enter the path for creating the checkpoint file, if you selected the checkbox in step 3.
5. Click to select **Create a** post-processor **log file** if desired (see glossary for further information about post-processor log files).
6. If you chose to create the log file in step 5, select the **Replace** checkbox if you wish to overwrite any old log files. Leaving the checkbox clear will append to the existing log file instead of replacing the file.

7. If you chose to create the log file in step 5, enter the path to store the log file in, or accept the default setting.

8. Click **Next>>** to move to the next screen.

Note: A checkpoint file produces the same results as using the original executable. However, if the original executable changes, the checkpoint file is obsolete. It only applies to the exact executable SecuredApp used.

DLL Selection

Application DLLs have two protection options--**Verify at Runtime** and **Protect and Verify at Runtime**. **Verify** means that SecuredApp will check the DLL upon launch of the protected application to verify that they match their original state. The **Protect** feature indicates that SecuredApp will guard the DLL file during use of the protected application. When packaging an application you must specify the DLLs that SecuredApp is to protect and/or verify in this step.

To select DLLs for verification:

1. Click **Browse** next to the DLL Name(s) box under the **DLLs to Verify at Runtime** field.

2. Navigate to the application's DLLs and choose the DLLs that you wish to verify at launch (hold down the **<Control>** key to choose multiple DLLs).

3. Click **Open**.

4. DLLs added to this field accidentally can be removed by highlighting the DLL and pressing **<Delete>** on your keyboard.

To add DLLs for protection *and* verification:

1. Click **Browse** next to the DLL Name(s) box under the **DLLs to Protect and Verify at Runtime** field.

2. Navigate to the application's DLLs and choose the DLLs that you wish to verify at launch (hold down the **<Control>** key to choose multiple DLLs).

3. Click Open.

4. DLLs added to this field accidentally can be removed by highlighting the DLL and pressing **<Delete>** on your keyboard.

Protection Level

In this step you will choose a **Protection Level** and **Function Level**.

1. Choose either **Use a Pre-defined Security Level Definition** or **Use Project-Specific Security Parameters** in the **Protection Level** heading. The default is **Maximum Security**, and **Minimum Security** is the alternate setting.

    - If you choose to use Project-Specific Parameters, click **Set/View** to choose the parameters. See the Project-Specific Security Parameters topic for further information about these parameters.

2. Choose the **Function Level Configuration** settings:

- Enter a number in the **Minimum Encryption Size (in bytes)** field. SecuredApp will not encrypt any functions smaller than the size in this box. The default setting is 500 bytes.

- **Specify functions to include or exclude**. For more information on this process, see the Function Selection topic.

3. Click **Next>>**.

Project-Specific Security Parameters

In the **Project-specific Security Parameters** window, complete the following:

1. In the **Encryption Parameters** box, specify the following:

    - **Maximum # of times to call**. This is the number of times SecuredApp will decrypt a function before leaving it decrypted in memory. The default is 3000.

    - **Maximum # of times to decrypt**. This is the number of times SecuredApp will call a function before leaving it decrypted in memory. The default is -1, which means to always re-encrypt.

    - **How often to re-encrypt**. This determines when SecuredApp re-encrypts a function that is available for re-encryption. For example, if this is set to **5**, SecuredApp will re-encrypt this function every fifth time it becomes available for re-encryption.

2. In the **Runtime Options** box, specify the following:

    - Check the **Decrypt to temporary memory** checkbox to have the protected application's encrypted code decrypted to semi-random temporary memory. (This increases security, but it may include an application performance decrease.)

    - Check the **Deny external application access** checkbox to prevent such discretionary process access as having access to the Printer dialog or being ended.

    - Check the **Warn before exit** box if you want the protected application to display a warning when it is terminating due to a detection of tampering or other conflict with the security rules associated.

    - Check the **Deny running within a virtual machine** box to deny application access when a virtual machine is launched.

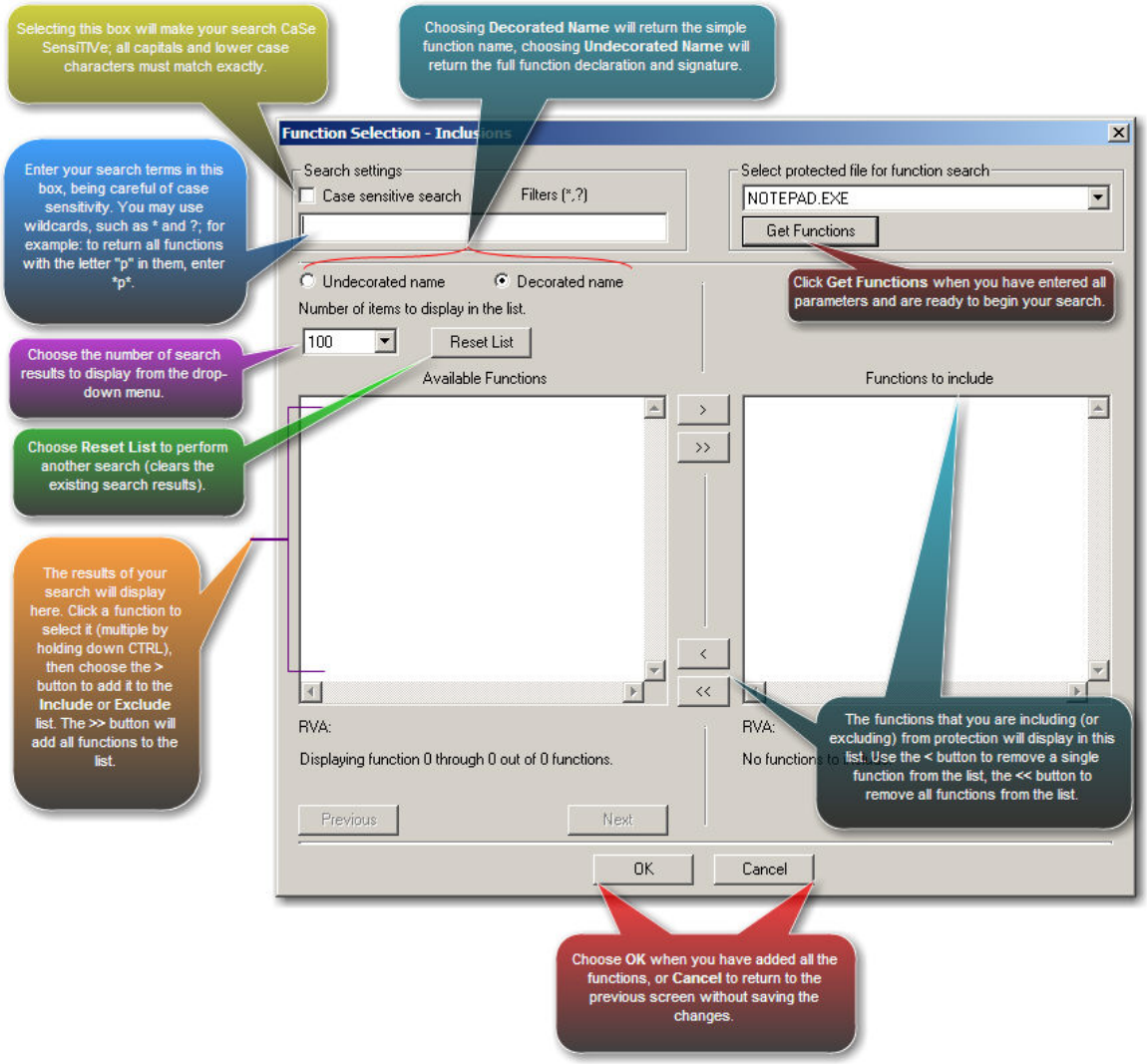
3. In the **Exceptions to Pass to the Protected Application**, SecuredApp will handle those left unchecked while passing the selected exceptions along to the protected application for handling. If the application needs to handle an exception, check that box.

    - To select all exceptions, click the **Check All** button, to un-check all exceptions, click the **Clear All** button.

4. Click **Apply** to apply the changes to the parameters, then click **Done**.

Function Selection - Inclusions and Inclusions

The **Protection Level** screen contains two buttons in the **Function Level Configuration** box—**Include** and **Exclude**. Both buttons function as a way to search for functions within the application to be protected that you want to either include or exclude.



Runtime Options

1. Choose the **Runtime Options** to use:

    - **Print the runtime statistics**. Check the Print runtime statistics checkbox to create a saMetrics.log file upon successful exit from an encrypted application. This file has a summary of the number of times each of the encrypted functions was encountered while the encrypted application was running.

    - **Multithreading support**. Turn on multithreading support only if you are trying to protect an application that is multithreaded and more than one thread may be calling a given function at the same time.

If Multithreading support is used but not needed, there may have a performance impact on the encrypted application. Likewise, if an application does utilize multithreading and multiple threads access the same protected function. This option must be chosen or the protected application is likely to crash.

2. Click **Next**.

Protection Parameters

1. Verify the **Protection Parameters** in the list.

2. To begin the **Registration Wizard** phase of protection, click **Start Protection**.

The **SecuredApp Post Processor Wizard** window pops up (this requires an Internet connection) to guide you through the Registration Wizard phase.

If you click **Stop Processing**, the operation aborts. If you have chosen to create a checkpoint file and analysis has completed, you can load the resulting .cpt file to resume operations from this point. In this case, post-processing begins.

Registration Wizard

The **Registration Wizard** phase of application protection allows you to configure the license options and pricing for the application. After the **Using SecuredApp** phase completes, the Registration Wizard log in window will appear.

To log in, enter your Username and Password and click **Log In**.

Select Wizard

The Registration Wizard allows two different types of setup wizards that you can use:

- The **Standard Setup Wizard** provides you with the basic tools to secure your application and set up pricing for sales.

- The **Advanced Setup Wizard** allows you to choose from previously stored Rule Sets or create new pricing rules as necessary.

On this screen, please:

1. Select the Sub-Account that you want to associate the protected application with from the drop-down menu

2. Select either the **Standard** or **Advanced** setup wizard by clicking to select the radio button next to the appropriate wizard

3. Click **Next** to continue.

Standard Setup Wizard

The **Standard Setup Wizard** provides you with the basic tools to secure your application and set up pricing for sales.

The first step is to choose the **Application Attributes**.

Application Attributes

Application Attributes include the Title, Description, and associated URLs for an application.

1. If you are a **Digital Store Front** client, you are asked to choose to add the application to your store front. Choose **Yes** to add the application, choose **No** if you do not wish to add the application at this time.
2. Enter a **Title** for your application (limited to 128 characters).
3. Enter a detailed **Description** of the application. This field will display to consumers.
4. *Enter the **Purchase URL** of the application that you are registering. This should be the actual location consumers visit to purchase the application, for example: http://www.myapplication.com/catalog/purchase.html.
5. *Enter the **Denial URL**. This should be the actual location of the page that you wish to send customers to when they are unable to successfully purchase the application. Please use standard formatting, for example: http://www.myapplication.com/denial.htm.
6. Enter the **License Renewal URL**. This is the page that users will be directed to when their license expires or the application is deprecated, and may offer new subscriptions, upgrades, or products. Please use standard formatting, for example: http://www.myapplication.com/renewal.htm.
7. Click **Next** to proceed.

**These items do not display when adding to your Digital Store Front.*

License Rules

In this step you will designate the **License Rules**, **Pricing**, and **Form** to use for the application.

1. Choose the pricing model that you wish to use:
    - **Buy Now**. Select the **Buy Now** option to allow consumers to purchase your application. Please choose the licensing options for the purchase:
        1. **Unlimited** will allow you to give full access to the application with no restrictions. To use this option, click to select the radio button.
        2. **Limited** allows you to give license to the user for a limited number of days or uses. To use this option, click to select the radio button, choose **days** or **uses** from the drop-down menu, and enter a value in the field.

3. Enter the Purchase Price in the **Price** field. This is the price that you wish to charge for the Buy Now license.

- **Subscription**. The **Subscription** option allows you to provide access to the application on a subscription basis. To use this option you must enter the pricing information for the subscription:

    1. **Initial Price**. Enter the price that you wish to charge for application access during the introductory period. For example; to charge $25 for the first 30 days, enter 25 in the **$** field, 00 in the **cents** field and 30 in the **days** field.

    2. **Initial Period**. Enter the time, in days, for the initial billing period. For example; to charge $25 for the first 30 days, enter 30 in the **days** field.

    3. **Recurring Price**. Enter the amount that you wish to bill for subsequent billing periods. For example; if the application is being billed at $25 for the first 30 days and then $50 a month for subsequent months, enter 50 in the **$** field, 00 in the **cents** field.

    4. **Recurring Period**. Enter the amount of time for subsequent billing periods. For example; if the application is being billed at $25 for the first 30 days and then $50 a month for subsequent months, enter 30 in the **Recurring Period** field.

    5. **Rebills**. Enter the period of time for billing to continue.

        - Choose **Indefinitely** to have the consumer billed every recurring period until cancellation or deprecation

        - Choose the **Times** option to have the system bill a certain number of times (enter the number of times to bill in the Times field for this option).

- **Free**. Choose the Free option to allow the consumer to have access to the product free of charge.

2. In the **Form Selection** section, choose the form to display in the shopping cart for the item. (To see a larger version of the form, click the form. Click the **Close** button on the larger version to close it and return to the License Rules screen.)

3. Click **Next** to proceed.

Promotion Setup

A Promotional License allows your consumers to try the software before the actual purchase. You can allow consumers to install the software and use it for a certain amount of days, or a defined amount of “launches”, depending on your needs. With this option, the consumer’s credit card will not be billed until the Promotional License expires.

To set up a **Promotional License**:

1. Select the **Yes** radio button.

2. Choose **Uses** or **Days** from the drop-down menu

3. Enter a value in the field (days or uses)

4. Click **Next** to proceed.

If you do not wish to set up a Promotional License, select the **No** radio button and click **Next**.

The Promotional License is not offered when you select **Free in the **License Rules** step.*

Completion

The **Completion** page is the last step of the Registration Wizard and allows you to add the **Download URL** and generates **Button** and **Link Code** to add to your shopping cart (if you are not using the Digital Portal you will need this code to add to your website).

1. Enter the complete URL where consumers will download the application from in the **Download URL** field. Example: *http://www.myapplication.com/downloads/application.exe*.

   Please do not use FTP or secured areas for this. Your application will be secured by the licensing protection that you have set up during this session.

2. Click **Next**.

3. Copy the codes that have been generated. This code will need to be placed in your catalog or shopping cart:

   - ***Button Code**. The button, when placed in your catalog, will read **Purchase Now**. You may replace the text *Purchase Now* in the code with alternate text if you wish, before placing the code in your catalog.

   - ***Link Code**. The button, when placed in your catalog, will read **Purchase Now**. You may replace the text *Purchase Now* in the code with alternate text if you wish, before placing the code in your catalog.

4. Enter your **Email Address** in the field provided to receive an email with the generated code. Please use standard formatting, for example: *developer@myapplication.com*.

5. Click **Finished**.

   If you do not click **Finished** on this and the next page, the registration will not be complete and the application will not show up in the Admin Area and the Post-Processor will not complete the encryption of your application.

6. The **Thank You for Registering** page loads. Click **Finished**.

7. The Internet connection closes and the Post-Processor resumes the encryption process.

The **Registration Wizard** phase is now complete. The web interface for the Registration Wizard will close, and the **SecuredApp Output** screen will now be visible again. The SecuredApp software will mark the time in the file, then continue the encryption of your application. Click **Close** to exit the SecuredApp Output window and return to the SecuredApp screen.

**This option is not available if you chose to add your product to your Digital Storefront.*

Advanced Setup Wizard

The Advanced Setup Wizard allows you to choose from previously stored Rule Sets or create new pricing rules as necessary.

Application Attributes

Application Attributes include the Title, Description, and associated URLs for an application.

1. If you are a **Digital Store Front** client, you are asked to choose to add the application to your store front. Choose **Yes** to add the application, choose **No** if you do not wish to add the application at this time.
2. Enter a **Title** for your application (limited to 128 characters).
3. Enter a detailed **Description** of the application. This field will display to consumers.
4. *Enter the **Purchase URL** of the application that you are registering. This should be the actual location consumers visit to purchase the application, for example: http://www.myapplication.com/catalog/purchase.html.
5. *Enter the **Denial URL**. This should be the actual location of the page that you wish to send customers to when they are unable to successfully purchase the application. Please use standard formatting, for example: http://www.myapplication.com/denial.htm.
6. Enter the **License Renewal URL**. This is the page that users will be directed to when their license expires or the application is deprecated, and may offer new subscriptions, upgrades, or products. Please use standard formatting, for example: http://www.myapplication.com/renewal.htm.
7. Click **Next** to proceed.

**These items do not display when adding to your Digital Store Front.*

License Rules

In this step you will designate the **License Rules** and **Pricing** to use for the application.

1. Choose one of the available options:
    - **Rule Set Listing**. Select this option to choose from a previously created Rule Set:
        1. Click to select the radio button next to **Rule Set Listing**.
        2. Click the radio button next to the Rule Set that you wish to use.

    The Rule Sets displayed can be filtered by **Rule Set Name**, **Status**, **Time**, **Uses**, **License Revocation**, **Promotion**, or **Installs** by clicking the up (ascending) or down (descending) arrows next to the column heading.

    - **New Rule Set**. Select this option to create a new Rule Set**:**
        1. Enter a descriptive **Ruleset Name** (maximum of 128 characters) This will display in the **Rule Set Listing** section after creation, so be certain to give it a name that you can recognize, for example, the name *BuyNow3launchenorevokenopromo3users can be used to describe a Rule Set for a Buy it Now license that allows 3 launches, never checks for revocation, has no promotional period, and allows three users.*
        2. Define the amount of time to allow access to the application after purchase in the **Time** field and define the time span by:

- Selecting **Unlimited** if you do not want to restrict by time
- Entering a value in the **Days** field to restrict by days (enter a value)
- Selecting **Subscription** and entering a value in the **Grace Period Days** field (choose a value from 0-6 from the drop-down menu) to allow the consumer to continue using the software after the scheduled rebill date, or until the system has been notified of subscription cancellation or failure to renew

3. Define the **Uses** parameter to restrict users to a maximum number of application launches, if desired. Choose:
    - **Unlimited** if you do not want to restrict uses
    - **Launches** to restrict a user to a maximum amount of application launches (enter a value)
    - **Hours** to restrict a user to a maximum amount of clock hours (enter a value)
4. Define the **License Revocation** parameter (the frequency that the application should check for License Revocation). Choose:
    - **Never Check** if you do not wish for the application to check for revoked licenses
    - **Launches** to check for license revocation based on application launch (enter 1 in the value field to check every time the consumer launches the software or 5 to check every fifth launch of the software, etc…)
    - **Days** to check for license revocation based on calendar days (enter 1 to check every day—the software will check on the first launch every day—or enter 5 to check every fifth day, etc....)
5. Define the **Promotion** parameter if you wish to offer a promotion (trial) for the application. Choose:
    - **None** if you do not wish to offer a promotion
    - **Launches** if you wish to offer a promotion with a maximum amount of launches (enter a value)
    - **Days** if you wish to offer a calendar day based promotion (trial)
6. Define the **Installs** parameter to restrict installations of the application. Choose:
    - **Unlimited** if you wish users to be able to install the computer for any users or any number of machines
    - **Users** if you wish to restrict the amount of users that can access the application on one machine (enter a value)

- **Computers** if you wish to restrict how many machines the application can be installed on

2. Click **Next** to proceed.

Single Purchase

This step allows you to define the pricing for a single purchase of the software and choose the look of your shopping cart.

1. Enter a purchase price for the application:

    - Select **Price** and enter a value in the fields to charge for a single purchase of the software

    - Select **Free** to offer the application free of charge

2. Select the radio button below the form that you wish to use in the **Form Selection** section. (To see a larger version of the form, click the form. Click the Close button on the larger version to close it and return to this screen.)

3. Click **Next** to proceed.

Completion

The **Completion** page is the last step of the Registration Wizard and allows you to add the **Download URL** and generates **Button** and **Link Code** to add to your shopping cart (if you are not using the Digital Portal you will need this code to add to your website).

1. Enter the complete URL where consumers will download the application from in the **Download URL** field. Example: *http://www.myapplication.com/downloads/application.exe*.

    Please do not use FTP or secured areas for this. Your application will be secured by the licensing protection that you have set up during this session.

2. Click **Next**.

3. Copy the codes that have been generated. This code will need to be placed in your catalog or shopping cart:

    - **Button Code**. The button, when placed in your catalog, will read **Purchase Now**. You may replace the text *Purchase Now* in the code with alternate text if you wish, before placing the code in your catalog.

    - **Link Code**. The button, when placed in your catalog, will read **Purchase Now**. You may replace the text *Purchase Now* in the code with alternate text if you wish, before placing the code in your catalog.

4. Enter your **Email Address** in the field provided to receive an email with the generated code. Please use standard formatting, for example: *developer@myapplication.com*.

5. Click **Finished**.

    If you do not click **Finished** on this and the next page, the registration will not be complete and the application will not show up in the Admin System and the Post-Processor will not complete the encryption of your application. .

6. The **Thank You for Registering** page loads. Click **Finished**.

The **Registration Wizard** phase is now complete. The web interface for the Registration Wizard will close, and the **SecuredApp Output** screen will now be visible again. The SecuredApp software will mark the time in the file and begin to encode the application. After the Post-Processor has completed, click **Close** to exit the SecuredApp Output window and return to the SecuredApp screen.

Re-Encrypt an Existing Application

Re-Encrypting an application allows you to encrypt new versions of an application and still keep the same exact license and payment options.

Select Application

Use the drop-down menu to select the existing application that you wish to re-encrypt. Click **Next** to proceed.

Completion

By clicking **Finished** on this page, the Internet connection will close and SecuredApp will continue with the Post-Processing of your application. The Licensing settings will remain the same as the first time you encrypted it, and the encryption parameters set up in the Post-Processor will be applied to the application. You may now proceed to the Installer phase.

Post Processing Operations

The process can take a few minutes or longer, depending on the size of the application. During this time, SecuredApp:

- Opens and analyzes the main application and any DLLs you have chosen for it to protect.
- Generates **Encryption Keys**.
- Protects all functions explicitly selected or that fall within the minimum encrypt size that you specified.
- Reads all DLLs you have chosen to verify at runtime and creates checksums for them.
- Write out the new application, along with the **Secure Execution Monitor** for it and any DLLs specified.
- Creates a **/SecuredApp** subfolder and places the protected application and any referenced application there. This can be overridden within the .vpj file.

After the application has been encrypted you can move on to the Installer Phase.

Installer Phase

After completing the encoding and registration phases, it's time to hand the packaged application back to your development team so that they can incorporate the application in the installer and begin to distribute the application to your consumers.

Distribution

Please remember as you distribute the application to be certain that the file is available in the same location that you specified as the **Download URL** in the registration phase of SecuredApp.

Post-Distribution

After the secured application has been distributed, you still have full control over the application. You can fully edit any features set up in the Registration Wizard phase of SecuredApp by signing into your Admin area. Help topics for these features are available within that Admin system.

Running a Protected Application

Consumers can launch the SecuredApp-protected application the same as the original—double-click its icon, drop a compatible file on its icon, choose it from the **Start** menu, or use a command line.

Runtime Operations

SecuredApp's post-processing protects an application both at rest and while it runs. When the application is at rest, SecuredApp keeps its functions encrypted and obfuscated. This prevents anyone from using static analysis tools to peel back layers of protection.

At runtime, the **Secure Execution Monitor** (SEM) first checks the local environment for threats. These threats consist of debugger checks, known applications, integrity checking the application, and verifying any critical DLLs it requires. If the SEM detects any threats, the application exits quietly, without giving any indication of why.

If the environment passes scrutiny, the SEM starts the application. It decrypts and verifies the integrity of the main function, then transfers control to start the application.

Appendix A: Windows Exceptions

Below is a list of the Windows Exceptions that SecuredApp can handle, as well as a brief description of each:

- **ACCESS_VIOLATION** a thread tried to read from or write to a virtual address for which it does not have the appropriate access.
- **DATATYPE_MISALIGNMENT** a thread tried to read or write data that is misaligned on hardware that does not provide alignment.
- **ARRAY_BOUNDS_EXCEEDED** a thread tried to access an array element that is out of bounds and the underlying hardware supports bounds checking.
- **FLT_DENORMAL_OPERAND o**ne of the operands in a floating-point operation is denormal. A denormal value is one that is too small to represent as a standard floating-point value.
- **FLT_DIVIDE_BY_ZERO** the thread tried to divide a floating-point value by a floating-point divisor of zero.
- **FLT_INEXACT_RESULT** the result of a floating-point operation cannot be represented exactly as a decimal fraction.

- **FLT_INVALID_OPERATION** this exception represents any floating-point exception not included in this list of exceptions.
- **FLT_OVERFLOW** the exponent of a floating-point operation is greater than the magnitude allowed by the corresponding type.
- **FLT_STACK_CHECK** the stack overflowed or underflowed as the result of a floating-point operation.
- **FLT_UNDERFLOW** the exponent of a floating-point operation is less than the magnitude allowed by the corresponding type.
- **INT_DIVIDE_BY_ZERO** the thread tried to divide an integer value by an integer divisor of zero.
- **INT_OVERFLOW** the result of an integer operation caused a carry out of the most significant bit of the result.
- **PRIV_INSTRUCTION** the thread tried to execute an instruction whose operation is not allowed in the current machine mode.
- **IN_PAGE_ERROR** the thread tried to access a page that was not present, and the system was unable to load the page. For example, this exception might occur if a network connection is lost while running a program over the network.
- **ILLEGAL_INSTRUCTION** the thread tried to execute an invalid instruction.
- **NONCONTINUABLE_EXCEPTION** the thread tried to continue execution after a non-continuable exception occurred.
- **STACK_OVERFLOW** the thread used up its stack.
- **INVALID_DISPOSITION** an exception handler returned an invalid disposition to the exception dispatcher. Programmers using a high-level language such as C should never encounter this exception.
- **INVALID_HANDLE** the thread used a handle to a kernel object that was invalid (probably because it had been closed.)
- **CONTROL_C_EXIT** the thread terminated because of CONTROL+C.