

FEITIAN



Rockey7.NET Enveloper Help

V2.0

Feitian Technologies Co., Ltd.

Website: www.FTsafes.com

Revision History:

Date	Revision	Description
Sep. 2009	V1.0	1 st release of the document
Sep. 2010	V2.0	2 nd release of the document

Software Developer's Agreement

All Products of Feitian Technologies Co., Ltd. (Feitian) including, but not limited to, evaluation copies, diskettes, CD-ROMs, hardware and documentation, and all future orders, are subject to the terms of this Agreement. If you do not agree with the terms herein, please return the evaluation package to us, postage and insurance prepaid, within seven days of their receipt, and we will reimburse you the cost of the Product, less freight and reasonable handling charges.

1. Allowable Use – You may merge and link the Software with other programs for the sole purpose of protecting those programs in accordance with the usage described in the Developer's Guide. You may make archival copies of the Software.
2. Prohibited Use – The Software or hardware or any other part of the Product may not be copied, reengineered, disassembled, decompiled, revised, enhanced or otherwise modified, except as specifically allowed in item 1. You may not reverse engineer the Software or any part of the product or attempt to discover the Software's source code. You may not use the magnetic or optical media included with the Product for the purposes of transferring or storing data that was not either an original part of the Product, or a Feitian provided enhancement or upgrade to the Product.
3. Warranty – Feitian warrants that the hardware and Software storage media are substantially free from significant defects of workmanship or materials for a time period of twelve (12) months from the date of delivery of the Product to you.
4. Breach of Warranty – In the event of breach of this warranty, Feitian's sole obligation is to replace or repair, at the discretion of Feitian, any Product free of charge. Any replaced Product becomes the property of Feitian.

Warranty claims must be made in writing to Feitian during the warranty period and within fourteen (14) days after the observation of the defect. All warranty claims must be accompanied by evidence of the defect that is deemed satisfactory by Feitian. Any Products that you return to Feitian, or a Feitian authorized distributor, must be sent with freight and insurance prepaid.

EXCEPT AS STATED ABOVE, THERE IS NO OTHER WARRANTY OR REPRESENTATION OF THE PRODUCT, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

5. Limitation of Feitian's Liability – Feitian's entire liability to you or any other party for any cause whatsoever, whether in contract or in tort, including negligence, shall not exceed the price you paid for the unit of the Product that caused the damages or are the subject of, or indirectly related to the cause of action. In no event shall Feitian be liable for any damages caused by your failure to meet your obligations, nor for any loss of data, profit or savings, or any other consequential and incidental damages, even if Feitian has been advised of the possibility of damages, or for any claim by you based on any third-party claim.

6. Termination – This Agreement shall terminate if you fail to comply with the terms herein. Items 2, 3, 4 and 5 shall survive any termination of this Agreement.

Contents

Chapter 1. Overview	1
1.1 Introduction to Rocky7.NET	1
1.1.1 ROCKEY7.NET Features.....	3
1.1.2 Supported Environments.....	4
1.2 Introduction to ROCKEY7.NET Enveloper.....	4
1.3 Encryption Process with Enveloper.....	5
1.4 Encryption Effects	5
Chapter 2. Using ROCKEY7.NET Enveloper	8
2.1 Choosing Application	9
2.2 Opening License File	11
2.3 Code Analysis	11
2.4 Setting Hardware and Binding Information	13
2.5 Running in Simulator Mode	15
2.6 Encrypting Program	16
2.7 Test Encryption	17
Chapter 3. Description Of ROCKEY7.NET Enveloper Features	18
3.1 Menu Bar	18
3.2 Toolbar	19
3.3 File Selection Area	20
3.4 List of Encrypted Files Area.....	20
3.5 Function Selection Area	20
3.6 Operational Information Area.....	20
Chapter 4. Problem Analysis	22

Chapter 1. Overview

1.1 Introduction to Rocky7.NET

ROCKEY7.NET is the ideal solution to protect .NET application against piracy. The 32bits high performance smart card chip based hardware with built-in .NET virtual machine allows .NET applet being executed within the device. Software developers can download .NET applet to the device and make it work with outer .NET application. The secure communication technique is applied to protect the data transferred between the device and host machine. Each ROCKEY.NET dongle can run multiple .NET applets simultaneously so as to protect multiple .NET application at the same time. The design of ROCKEY7.NET follows the up-to-date .NET technology. Its stable and effective performance provides excellent protection to Microsoft .NET applications.

With the advent of .NET technology, its simple, powerful and effective features are widely accepted by the users in various fields. The .NET program will be compiled into an IL (Intermediate Language) scripts and being rendered in different .NET runtime environment. The same IL scripts can be used in different platforms, which can significantly reduce the development and maintenance cost. However, the simple structure of IL language makes it too easy to be decompiled into source code. Thus the .NET program is easy to be cracked. How to protect .NET IL scripts is the most important topic of .NET application protection. Normally, the following techniques can be applied to achieve the goal:

- Anti-debugging

The protected .NET application has the feature that it cannot be debugged or audited by .NET debugger tools.

- JIT runtime decode

The instructions of .NET application will be decoded before it is rendered by JIT in runtime. The .NET application in memory is not complete.

- Function body encryption

The body of function will be encrypted with cryptographic algorithm. The .NET application is stored in cipher form.

- Keyword confusion

The keywords in the .NET application like function name, variable name will be replaced with irrerecognizable code. So even if the application is decompiled, it is not easy for cracker to understand the meaning of source code.

- Communication encryption

For the applications involving communication with hardware, the transfer datagram is encrypted and

protected against interception.

All the above methods can protect the .NET application. But they have a common feature. The protected application is completely stored in the computer, regardless it is stored in cipher or plain format. So it is easy for experienced cracker to obtain the complete original .NET application by using decryption tools. However, ROCKEY7.NET is designed to terminate the risk coming in this situation.

ROCKEY7.NET software protection solution provides a unique “function migration” method to protect the .NET application. It can download a part of the functions in original application to its built-in .NET virtual machine and execute the code in its .NET runtime environment. The internal .NET applet can cooperate with outer application to achieve the same functionality as the unprotected application. The ROCKEY7.NET built-in .NET virtual machine runtime environment makes this innovative design achievable. Worldwide, only a few manufactures can develop .NET virtual machine runtime environment, Feitian is one of them. Feitian .NET virtual machine runtime environment has been certified by Microsoft. Based on 32bits high performance smart card chip, Feitian .NET smart card provides a stable, effective platform for .NET applets being executed onboard. It is the perfect environment for .NET smart card applications. ROCKEY7.NET solution is based on Feitian .NET smart card. The Enveloper can intelligently analysis the target .NET application, migrate the core part of the application to the Feitian .NET card automatically. The protected application is not complete. A core part of the program is protected by the Feitian .NET smart card. The migration concept can guarantee that the outer application can never be fully reversely decompiled.

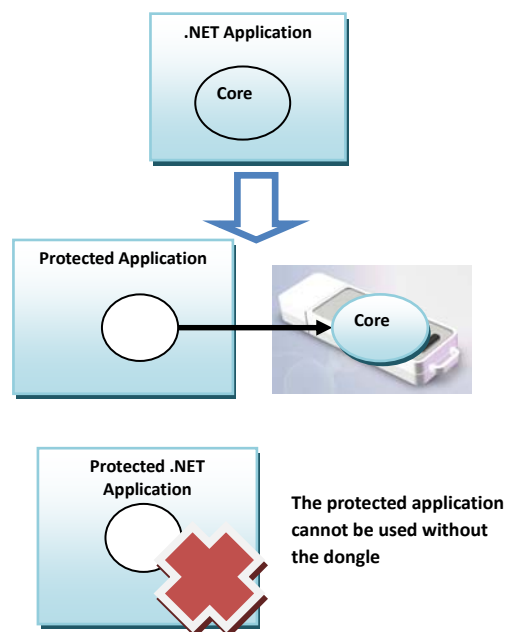


Figure 1.1 ROCKEY7.NET Software Protection Principle

ROCKEY7.NET provides many utilities to help developer maintain the entire software life cycle. The Enveloper, initialization tool, production tool, remote update tool can help developer protect and control the software remotely. Normally, there are several roles in software maintain cycle: administrator, developer, production staff,

sales staff and support staff. For each role, there are according tool to help developer perform the operation. This design can effectually separate the responsibility and task for each different role.

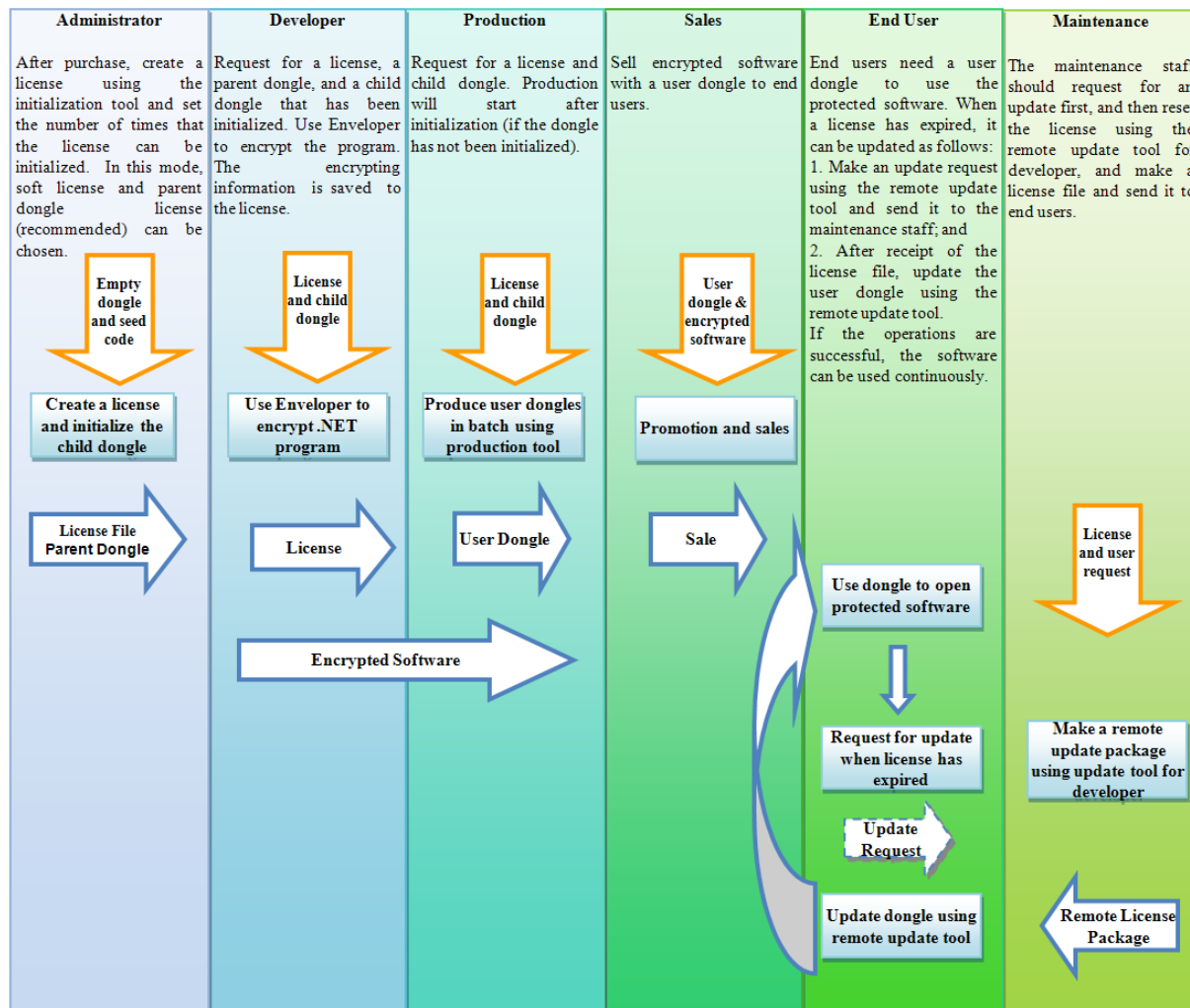


Figure 1.2 ROCKY7.NET Software Maintenance Flow

1.1.1 ROCKY7.NET Features

- 32-bit high performance smart card
- Globally unique hardware serial number
- User-defined 4-byte product ID and 8-byte management password
- No user installation of driver needed: CCID driver provided with operating system available or the driver can be installed automatically online
- Communicated packets are encrypted, preventing listening simulation software
- Counting or timing mode for user selection
- Easy and secure remote upgrade

- .NET virtual machine system with Feitian's own IPR
- Smart enveloper, no development work for encrypted product, smart and automatic analysis of .NET code functions of application, automatic migration of resulting algorithms and functions into card to run
- A complete set of easy-to-use tools for controllable license, and role and process-based management of encrypted products

1.1.2 Supported Environments

- Framework: Framework 2.0, Framework 3.0, Framework 3.5, Framework 4.0
- Supported development languages: C#, VB.NET, Delphi.NET, ASP.NET
- Supported .NET application programs: UI, console, service, control, and component programs
- Supported operating systems: Windows 2000, Windows 2003(32/64-bit), Windows XP(32/64-bit), Windows Vista(32/64-bit), Windows 2008 (32/64-bit), and Windows 7 (32/64-bit)

1.2 Introduction to ROCKEY7.NET Enveloper

The smart, secure, effective and easy to use ROCKEY7.NET Enveloper is an integral part of the ROCKEY7.NET product. It is based on the .NET virtual machine technology, and is designed to migrate the key functions of .NET programs into the smart card automatically. The protected key functions and algorithms are implemented by .NET applets in the dongle, leaving no complete program bodies on the computer. Thus, program tracing and de-compilation are made impossible. Without a valid ROCKEY7.NET product, the application program cannot run normally due to lack of its key parts.

In addition to the unique program migration mechanism of ROCKEY7.NET, the Enveloper also provides other .NET software protection means, including:

- Anti-debugging

It prevents dynamic debugging by .NET debugging programs when the encrypted program runs.

- JIT Real Time Decoding

This dynamically decodes the commands transmitted into JIT in real time, which avoids the appearance of full code body in memory.

- Encrypting Function Body

Encrypt the content of functions using an encryption algorithm, making de-compilation tools invalid when the function is off running.

- Naming Confusion

This means replaces or hides the name of functions, resulting in unknown de-compiled code.

- Communication Data Encryption

For hardware-based protection solutions, encrypt the communication data packets between the hardware and the program, preventing data from being intercepted.

Users do not need to program to use ROCKEY7.NET Enveloper for protecting .NET application programs, and have the source code of .NET programs. Encryption is processed by the Enveloper automatically, with implementation of other protection means. ROCKEY7.NET, a world-leading .NET protection solution, makes your software programs quite firm and solid.

1.3 Encryption Process with Enveloper

To protect your software with ROCKEY7.NET Enveloper:

1. De-compile the .NET source program to parse the structure of the functions of the target program.
2. Analyze the migratability of each function according to the de-compiled code to select functions to be migrated into the card.
3. List all candidate functions for users to pick core functions of the software.
4. Migrate user-selected functions into the dongle and encrypt the external program.
5. If a strong signature of the source program exists, re-deploy the signature.

The encryption process is implemented with an efficient ROCKEY7.NET encryption engine. The process of envelope encryption is perspicuous and stable.

1.4 Encryption Effects

This section illustrates the protection effects of the ROCKEY7.NET product with some .NET decompilers.

Reflector is a robust .NET decompiler which can easily restore a set of programs to source code. The following is the result of the Reflector parse of a program encrypted by ROCKEY7.NET:

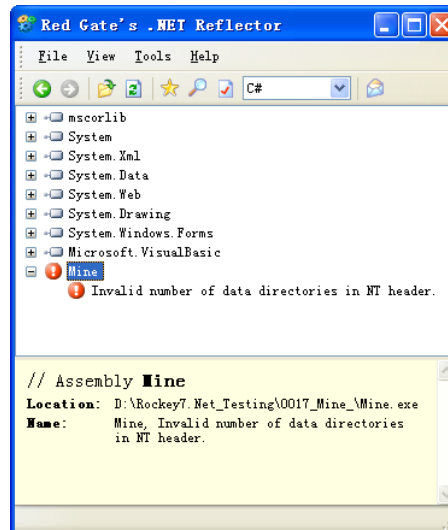


Figure 1.3 Parsing Protected Program Using Reflector

Apparently, the protected program is not properly restored.

Spices.NET is also a powerful decompiler. It can even decompile a set of programs that have been mixed by a process and automatically render the gotos of the mixture. Here, Spices.NET is used to open both the original program and the encrypted program by ROCKEY7.NET. The following is the result:

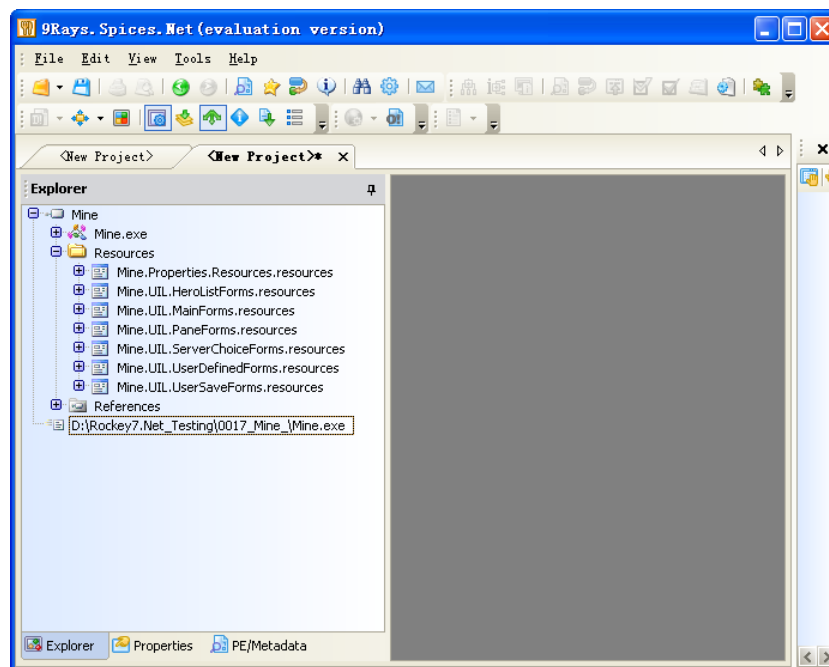


Figure 1.4 Parsing Protected Program Using Spices.NET

It can be seen that the unprotected program is fully parsed and restored by Spices.NET, but the protected program cannot be parsed, even the structure is not rendered normally.

Ilasm is a .NET decompiler available from Microsoft. It can be used to restore a .NET program into IL intermediate code.

Obviously, the keywords, such as the function names of the protected program, are not displayed (the code size is 0), because of the mixture and encryption mechanisms employed by ROCKEY7.NET.

Chapter 2. Using ROCKEY7.NET Enveloper

Run ROCKEY7.NET Enveloper. The main interface will be displayed as follows:

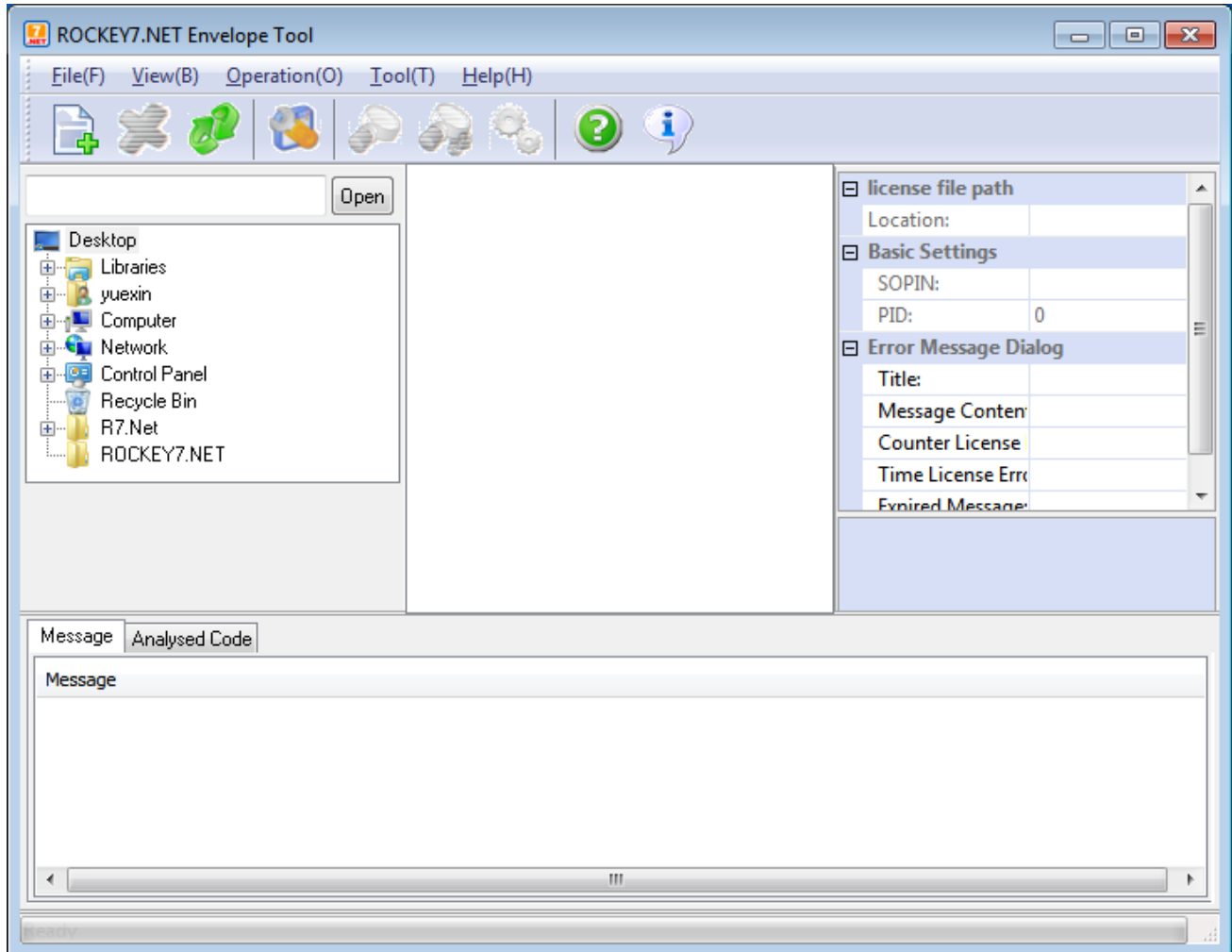


Figure 2.1 Main Interface

You can see that the main interface is divided into the following areas: the menu bar, the toolbar, the file selection area, the list of encrypted files area, the feature selection area, and the operational information area.

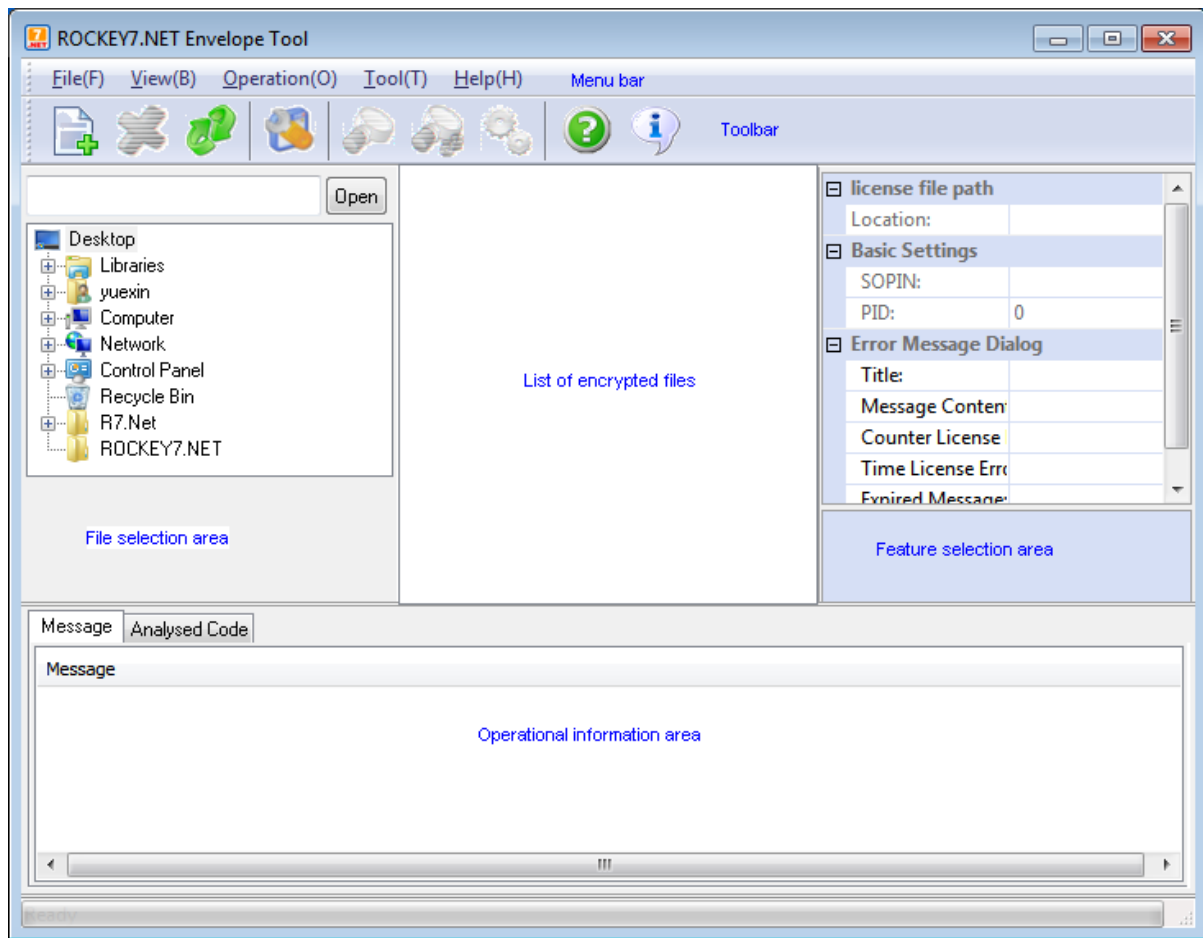


Figure 2.2 Components of Main Interface

- Menu bar: Contains the menu of functions of the Enveloper
- Toolbar: Contains the shortcut buttons of the Enveloper
- File selection area: Used for choosing a target .NET program
- List of encrypted files area: Used for listing selected .NET programs
- Feature selection area: Used for configuring error messages and the binding information for encrypted programs
- Operational information area: Users can view prompt messages during an operation on Message page and the functions that can be migrated according to code analysis for user selection on Analysed Code area

2.1 Choosing Application

Users can choose a program to encrypt. When choosing a file, only filtered files can be seen by users in the file selection area. These files include .exe and .dll files. Click Add Program on the toolbar to add the selected program

to the Enveloper. Or, users can double-click on the target program to add it.

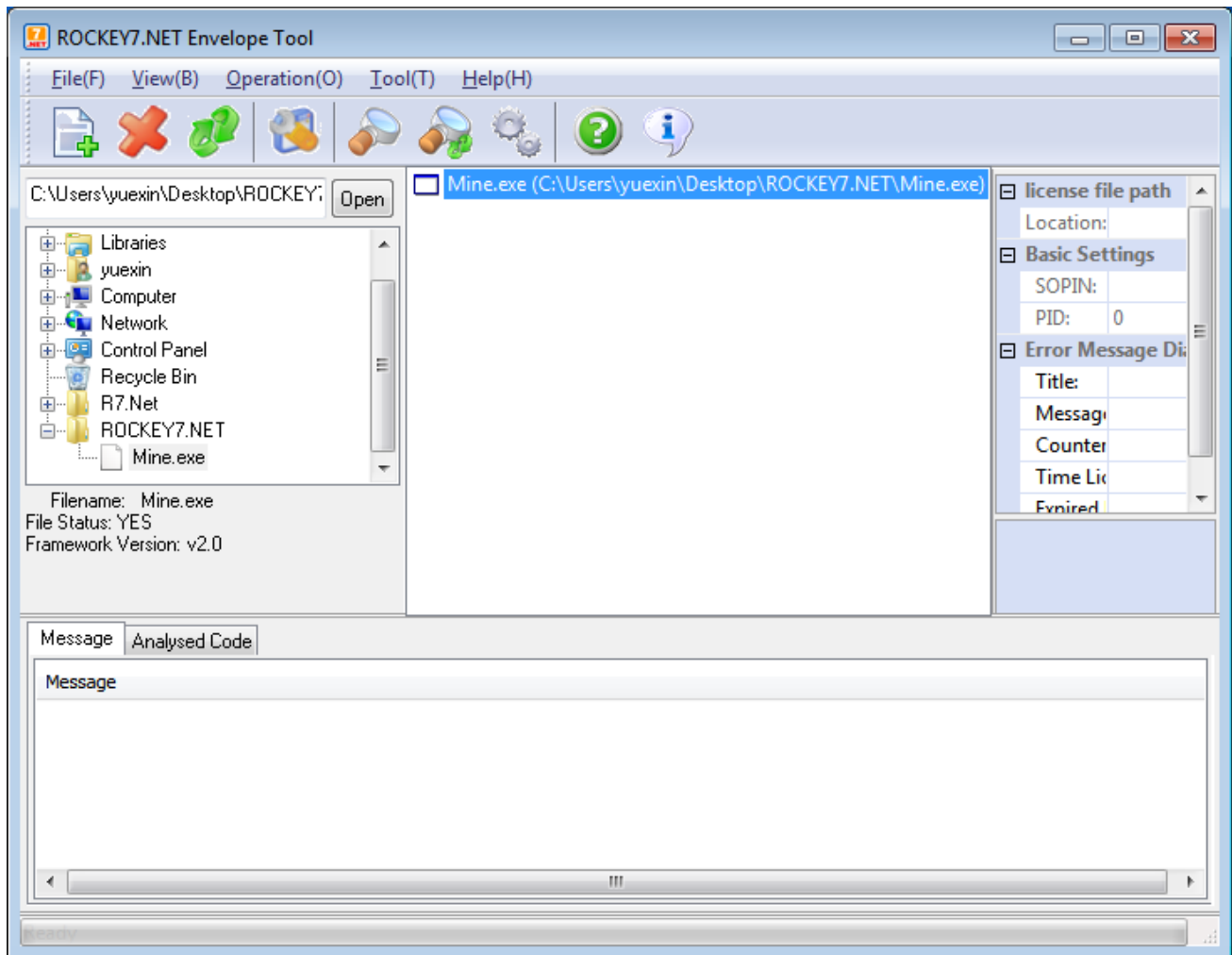


Figure 2.3 Choosing Program

After a program is selected, the Enveloper will automatically analyze the structure of the program. If it is a .NET program, its properties will be listed in the file selection area. If the selected program is not a .NET program, an error message will be displayed.

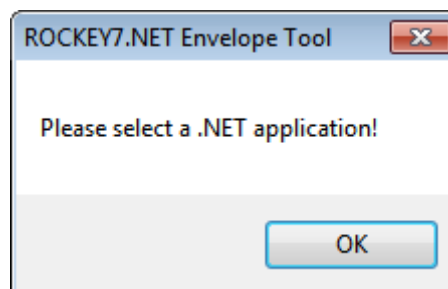


Figure 2.4 Unsupported File Format

Users can select one or more .NET programs. The program files will be listed one by one in the list of encrypted files area. If users want to cancel the encryption of a specific program, select the file and click Delete button.

2.2 Opening License File

After selecting a protected .NET program, connect a parent dongle for creation of license file and a user dongle to your computer. Choose a license file for the project. The license file will be opened then. The Enveloper will check the status of the parent dongle and the use dongle. If the status is normal, the super password for the project and the product ID will be displayed in Basic Settings area.

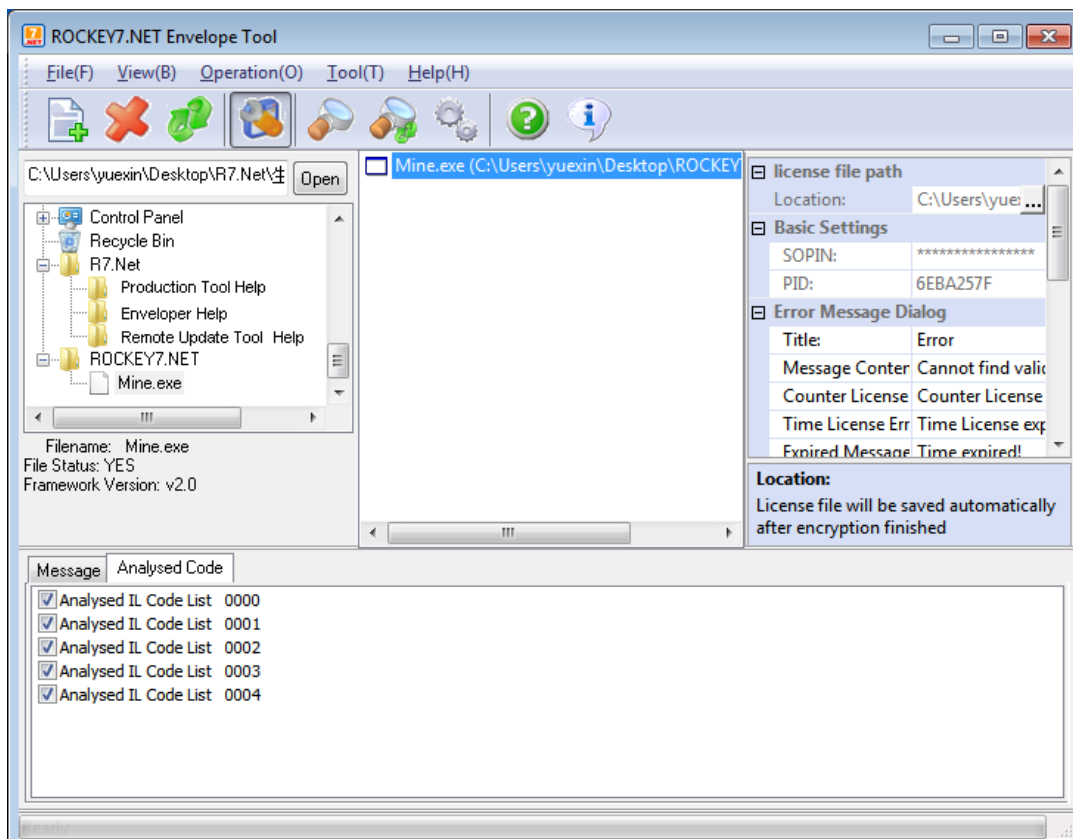


Figure 2.5 Selecting Migrated Functions

2.3 Code Analysis

The transferrable functions can be shown up by analyzing code after selecting a target program. To do so, click Code Analysis button in the toolbar.

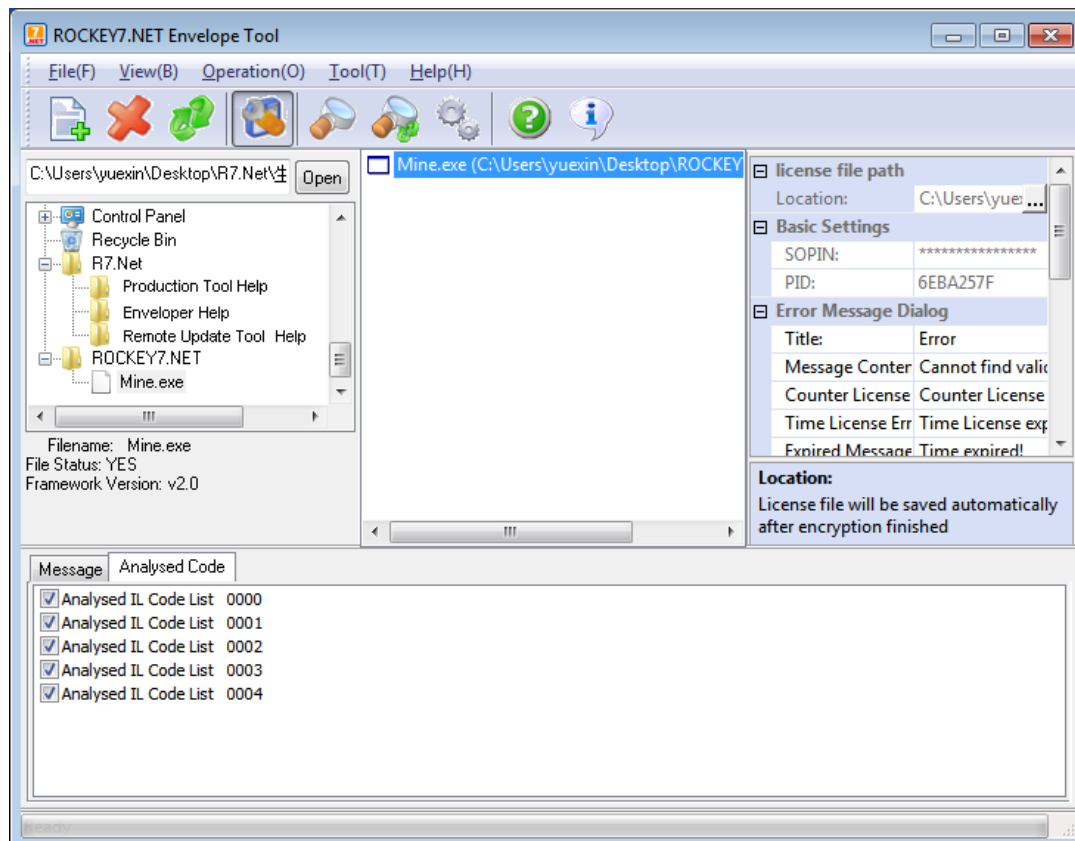


Figure 2.6 Code Analysis

After analysis, the transferrable code will be listed in the operational information area. Only some code can be transferred into the dongle. The following functions will be filtered out:

- Functions with simple structure
- Functions without any algorithm in them
- Functions in which something unsupported for .NET virtual machine is called

Users can choose the functions to transfer themselves. Or, they can let the tool to do that for them after selecting all transferrable functions. Consider the following when selecting the functions:

- The transferred functions must be vital to the program
- The transferred functions contain a complex algorithm that is hard to work out
- The transferred functions do not use much system resources and rely on the performance of hardware heavily (e.g. no complex if/else, loop, and recursive procedures in the functions)
- The functions do not output anything

2.4 Setting Hardware and Binding Information

After choosing migrated functions, users should setting the hardware and binding information of the dongle.

license file path	
Location:	C:\Users\yuexin\...
Basic Settings	
SOPIN:	*****
PID:	6EBA257F
Error Message Dialog	
Title:	Error
Message Content:	Cannot find valid de
Counter License	Counter License exp
Time License Err	Time License expire
Expired Message	Time expired!
Binding Conditions	
Hardware ID	False
Hardware ID:	FFFFFFFFFFFFFFFF
User ID	False
Start from:	1
End to:	FFFFFFFF
Time License	True
Time License	Use usable hours lic
Expire Date:	2010-08-27
Trial Period:	30
Usable Hours:	120
Counter License	True
Counter Value	100
Regular Check In	10
Location:	
License file will be saved automatically after encryption finished	

Figure 2.7 Setting Hardware Information

■ Basic Settings

Specify the SO PIN and PID (product code) of the dongle. They are generated by the ROCKEY7.NET Initialization Tool. The SO PIN is used to control the read and write permissions of the dongle. With a correct SO PIN, you can write to the dongle through the Tool, thus importing the migrated functions to it. The product code is used to identify the dongle. The encrypted program will find the dongle by the product code and work with it.

■ Error Message Settings

When the protected program cannot find the appropriate dongle, the error message will be shown to the user.

Specify the title and content of the error message dialog box.

■ Binding Information

This information is used to bind the encrypted program to a specific dongle. Only with the appropriate dongle, users can open the protected program. The binding features are:

Hardware ID (hardware serial number): Each ROCKEY7.NET dongle has a unique serial number. If the application program is bound to the serial number, only one dongle can be used to open the protected program.

User ID (user serial number): This serial number identifies different users. Software manufacturers can assign a user serial number to each of their customers. If the user serial number is used for binding, only the dongle containing a user serial number falling into the allowed range can work with the protected program.

Timer Module: This module limits the use of the protected program to a specific time scope. The Tool provides the following timing schemas:

- **Expiration Date:** Once the date is reached, the encrypted program won't work any longer.
- **Usable Days:** When the specified days elapse from the first time the protected program is executed, the encrypted program will stop running.
- **Usable Hours:** When the running time of the protected program reaches this value, the program is disabled.


Counter Module: This module limits the number of the times of the use of the protected program. Each time the program is executed successfully, the counter decreases by 1. When the value of the counter reaches 0, the program cannot be opened.

Regular Check Interval: The protected program checks for the presence of the dongle at this interval. Once it finds that the dongle has been removed, it will not work.

Users can determine the binding information of the encryption process according to their licensing modes. After setting the binding information and successful encryption, the information will be stored in the project file. In the process of production with the Production Tool, the information is written to the dongle, so that the dongle can work with the protected program.

When the license of the dongle expires, end users do not need to return their devices to the software manufacturer. Instead, a new license can be written to the dongle with the ROCKEY7.NET Remote Update Tool. For detailed information on the remote update, see the Remote Update Tool document.

2.5 Running in Simulator Mode

The function Run in Simulator Mode is helpful for users to protect their programs and evaluate the encrypted programs. Click  in the toolbar. The protected program will be started. You can use the program for some time as normal. The number of calls to each of the transferrable functions will be displayed in the operational information area. You can choose the transferred functions again according to the number and run in simulator mode until both the performance and protection strength are satisfactory.

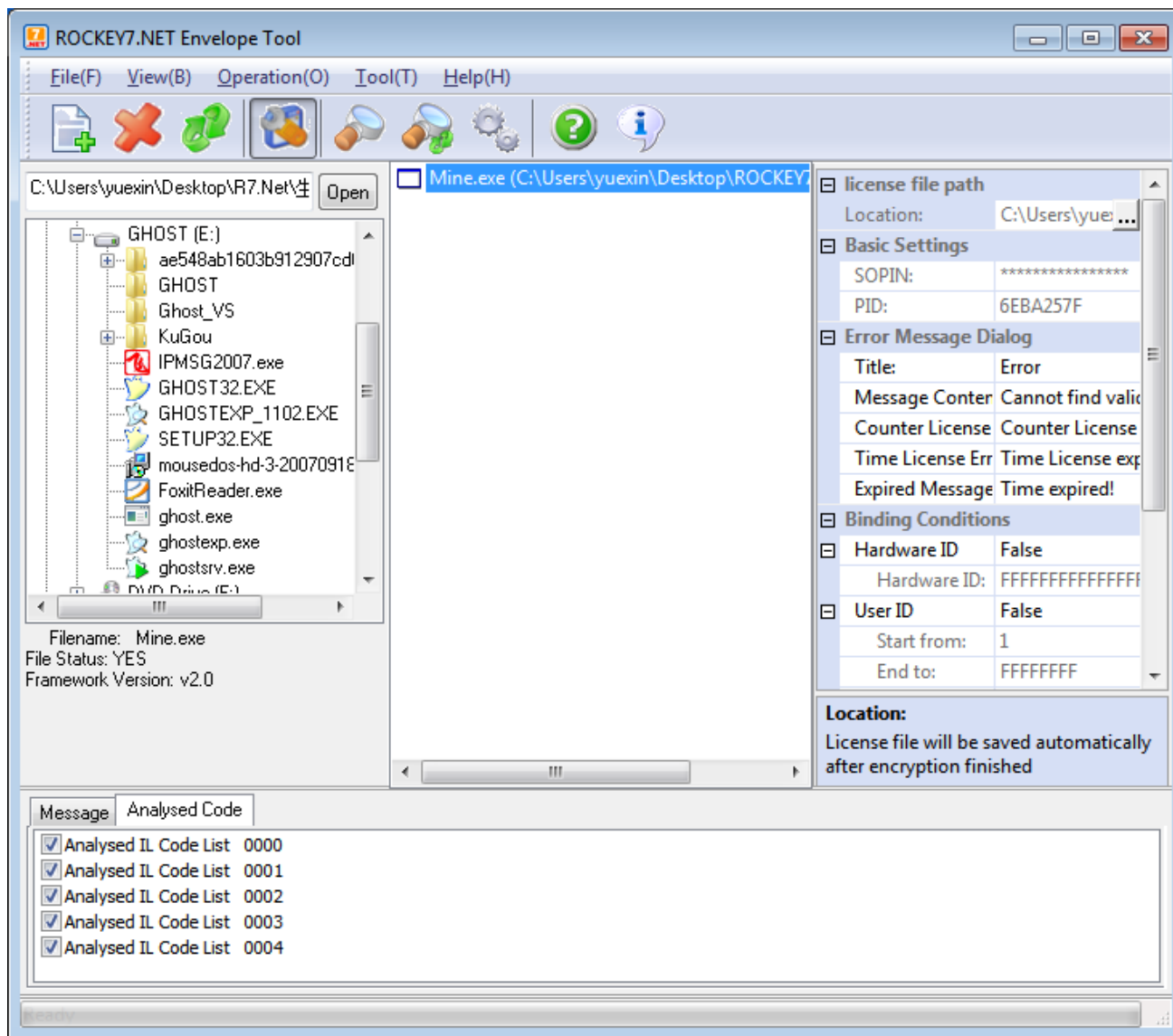


Figure 2.8 Running in Simulator Mode

According to the result of running in simulator mode, the following functions can be removed from the list of transferred functions:

- The functions that are not called

- The functions which are called many times and are not vital
- The functions that cause significant difference in the running speed of the program before and after transfer

2.6 Encrypting Program

After choosing the target program, specifying the migrated functions, and setting the hardware and binding information, users are able to protect their .NET programs by clicking on the menu bar or tool bar. Users can see the progress of encryption in the operation information area. The encryption time depends on the size and complexity of the protected program. If there are any problems in the encryption process, users can modify the settings according to the information in the operation information area. The FAQs of this document can also be used as a reference.

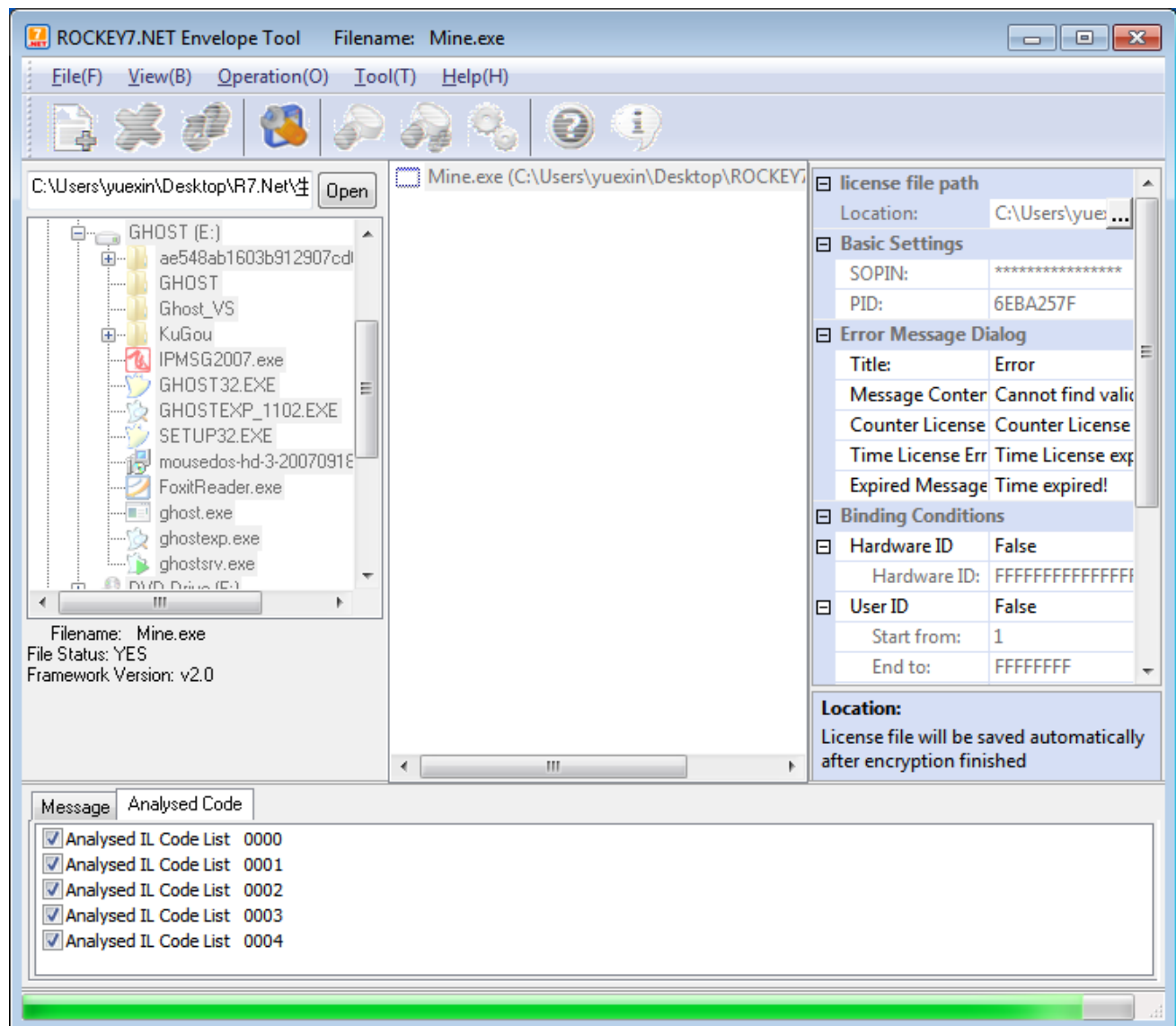


Figure 2.9 Encrypting Program

2.7 Test Encryption

After encrypting the program, users can run it to verify the effect of encryption. To do so:

- 1) Run the program without the dongle. The predefined error message will show up.

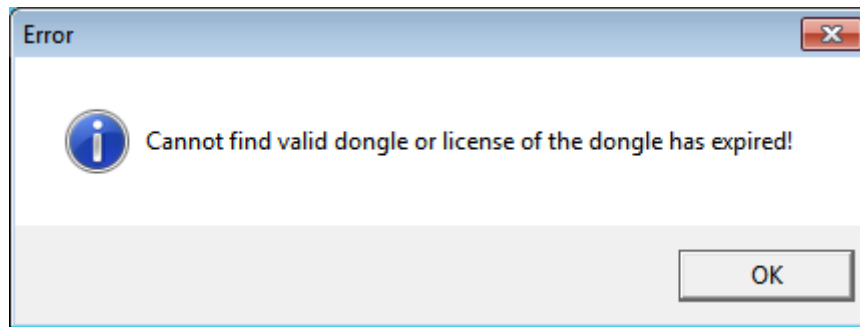


Figure 2.10 Error Message

- 2) Remove the ROCKEY7.NET device. The program won't run.
- 3) If a hardware ID has been bound, try to open the program with a different dongle. The program cannot be opened.
- 4) If a user ID has been bound, try to open the program with a dongle which contains a user serial number falling out of the binding range allowed for the program. The program cannot be opened.
- 5) If a timer module is used and the expiration date is specified, the program cannot be opened after the expiration date, even if the system clock is deliberately adjusted. Because the timer module is locked once the license has expired, modification to the system clock does not work.
- 6) If a timer module is used and the number of days is specified, the program cannot be opened the specified days after it is executed for the first time. Similarly, the program cannot be opened by modifying the system clock.
- 7) If a timer module is used and the number of hours is specified, the program cannot be opened the specified hours after it is executed for the first time. Similarly, the program cannot be opened by modifying the system clock.
- 8) If a counter module is used, users can open the program repeatedly, until the counter reaches its end. The program cannot be opened then.
- 9) If a regular check interval is specified, users can remove the dongle after the program is executed. After waiting for a time period longer than the interval, a prompt is displayed. Then, connect the dongle. The program runs again. Or users can click Retry button without inserting the dongle. After three times, the protected program will be terminated.

Chapter 3. Description Of ROCKEY7.NET Enveloper Features

3.1 Menu Bar

■ File Menu

Open License(G)	Open an existing license file.
Save License(S)	Save current configuration to a license file.
Add(A)	Add a file to the list of protected programs.
Delete(D)	Delete a file from the list of the protected programs.
Exit(X)	Exit the Tool.

■ View Menu

Tool Bar(T)	Open or close the tool bar.
Status Bar(S)	Open or close the status bar.
Advanced Bar(A)	Open or close the Advanced bar.

■ Operation Menu

Analyze(A)	Analyze a selected .NET program.
Run in Simulator Mode(S)	Simulate the running in encrypted mode.
Encrypt(E)	Encrypt a file.
Refresh Assembly	Refresh the tree view of the file selection area.

Tree(R)	
---------	--

■ Tool Menu

Resign with Strong Name(R)	Re-deploy the strong signature of the program.
----------------------------	--










Note: After encrypting the programs with a strong signature, you need to re-sign the programs using the tool. With the menu item Resign with Strong Name, you can specify an encrypted program as the target file and choose a key file for signing, and set a password for the key file and redeploy it. Thus, the encrypted program is redeployed with a strong signature.

■ Help Menu

Help(H)	Open the help documentation.
About(A)	Display version information of the Tool.

3.2 Toolbar

The toolbar includes the following icon buttons:

	Add File
	Remove File
	Refresh
	Advanced Mode
	Analyze: Pre-analyze all file added to the list of encrypted files for transferrable algorithm
	Run application in simulator mode: Simulate running of the .exe files in the list of encrypted files
	Start encryption
	Help: To open the help
	About: Display the version information of the program

3.3 File Selection Area

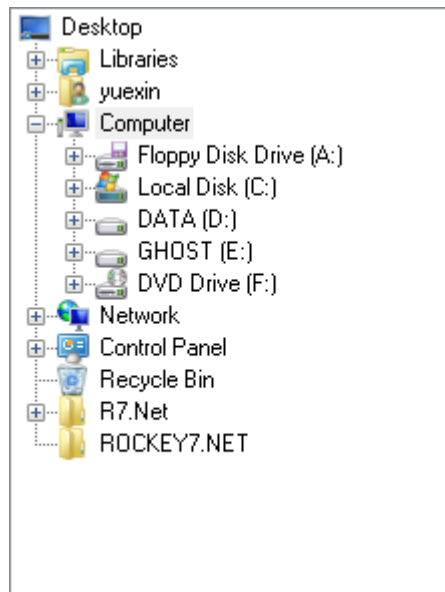


Figure 3.3 File Selection Area

This area is divided into two parts: the top part is used to display the tree view in which users can browse to choose a file. Only .exe and .dll files can be selected from the tree view. When you have selected a file, its information will be displayed in the bottom part.

3.4 List of Encrypted Files Area

The programs to be encrypted are displayed in the list of encrypted files area.

3.5 Function Selection Area

See section 2.4 for details.

3.6 Operational Information Area

The runtime status will be displayed in this area.

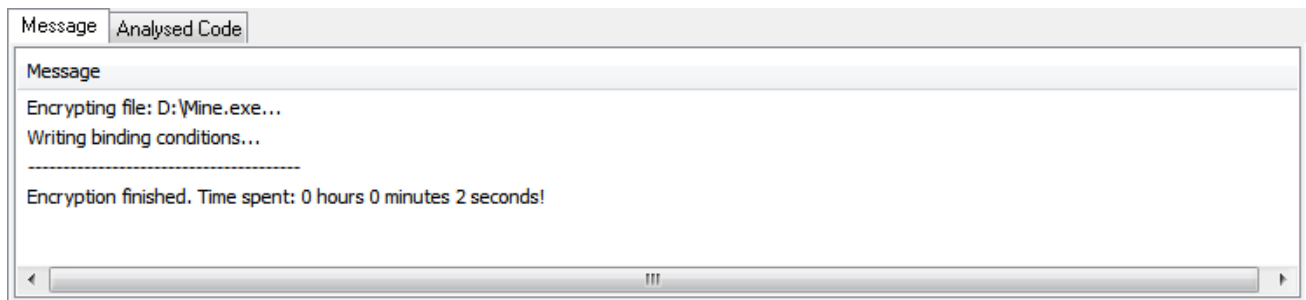


Figure 3.6 Operational Information Area

Chapter 4. Problem Analysis

- Unsupported file format. This can be displayed when you choose a file to encrypt from the tree view on the left.

Analysis: 1) Lack of Framework environment –Check if MS Framework2.0 or higher has been installed on your computer; or 2) The program you are encrypting is running – Stop the program and add the file again.

- No dongle is detected

Analysis: The ROCKEY7.NET dongle is not inserted or the hardware information of the dongle does not match the encryption settings. Please insert the valid ROCKEY7.NET dongle and check the binding conditions.

- Cannot open dongle, encryption failed

Analysis: The ROCKEY7.NET dongle is not inserted or the hardware information of the dongle does not match the encryption settings. Please insert the valid ROCKEY7.NET dongle and check the binding conditions.

- Verifying super password failed, encryption failed

Analysis: Make sure that the selected license file and the inserted dongle match.

- Cannot delete services in the dongle, encryption failed

Analysis: Please re-initialize the dongle.

- Cannot create services in the dongle, encryption failed:

Analysis: Cannot create the dongle services. This might be caused by free dongle memory space not enough. Please re-initialize the dongle.

- Writing service file in dongle failed, encryption failed

Analysis: Maybe the memory in dongle is insufficient. Please re-initialize the dongle.

- Starting dongle service failed, encryption failed

Analysis: The service file is not complete. Please re-initialize the dongle.

- No functions are obtained after clicking Analyze. Continue encryption?

Analysis: The programs do not include any algorithm and function that can run in dongle. You can go on to encrypt the program. Anti-debugging, protection based on method body, JIT image, name confusion,

and/or hardware binding will be applied.

- Writing binding information failed

Analysis: Failed to write binding information to the dongle. Please check the binding conditions.

- The encrypted program with a strong signature crashes

Analysis: Re-deploy the encrypted program.