User Manual

Virbox Protector

Version 2.0





Copyright & Trademarks

The Virbox, Virbox LM, Virbox Elite 5, **Virbox Protector** with its technical documentation is copyrighted to present by ©Beijing SenseShield Technology Co., Ltd (SenseShield). All rights reserved.

The Virbox, Virbox LM, Virbox Elite 5, Virbox Protector, are Registered Trademarks of SenseShield in China and other countries.

All products referenced throughout this document are trademarks of their respective owners.

Disclaimer

All attempts have been made to make the information in this document complete and accurate. But we cannot guarantee everything is perfect, we will correct it in next version released in case some error has been found. SenseShield is not responsible for any direct or indirect damages or loss of business resulted from inaccuracies or omissions.

The specifications contained in this document are subject to change without notice.

Documentation Improvement

Any suggestion to this manual from you are welcome, We are glad to hear any feedback from you which will help us to continuously improve the documents quality and support and serve the developer to protect software products more efficiently.

Contact

Company: Beijing Senseshield Technology Co., Ltd Address: Suite 510, Block C, Internet Innovation Center, Building 5, No.10, Xibeiwang East Road, Haidian District, Beijing China Tel: +86-10-56730936 Fax: +86-10-56730936-8007 Sales: info@senselock.com; Official Website: https://lm-global.virbox.com/ Virbox Developer Center (Virbox LM): https://developer.lm-global.virbox.com/



About this document

This document is designed to help Software Developer or Publisher to protect their Copyright or IP by protecting their software they would publish. And help the software resource supplier to protect their software resources.

Target User: The operation staff of Virbox Protector who is responsible for software copyright and IP protection.



Virbox Protector Update History:

Date	Version Number	Important Updates
2020.05	V 1.5	 Add the function to support Dotnet Core3 program protection,
		including Windows, Linux, macOS
		 Add the function to support to protect Unity3D customized program
		set.
		 Support Anti-debug plugin function to Linux, ARM-linux, Android
		platform.
2020.10	V2.0	
		 Add ARM-virtualization function to protect the ARM executable
		 Memory check function to PE and ELF program.
		 Add the compression function to DotNet DLL
		 Optimized the compression function to Dotnet executable
		 Support program protection to Java archive directly
		 Enhanced Anti-Runtrace function to ARM program.
		 Merged .vdata0 and .vadata1 segment
		 Support Control Flow Guard to PE program
		 Optimized the structure of the document



Table of Contents

1 Overview	7
1.1 Virbox Protector Introduction	7
1.2 Advanced and Secured Protection Technology	8
1.3 The program supported to be protected	8
2 Installation of Virbox Protector	
2.1 Installation	11
2.2 License mode of Virbox Protector	12
2.2.1 License Verification with cloud license (For Trial User)	12
2.2.1.1 Verify license by Virbox User License Tool	12
2.2.1.2 Sign in by the Virbox Protector Interface	13
2.2.2 License Verification with soft license	15
2.2.2.1 Use Virbox Protector in online environment	15
2.2.2.2 Use Virbox Protector in offline environment	16
2.2.3 License Verification with EI5 dongle (For official user use dongle license)	21
3 Protection Function Introduction	23
3.1 Main Menu of Virbox Protector	23
3.2 Menu Bar	23
3.2.1 File	23
3.2.2 Protect	24
3.2.3 Plug-in	25
3.2.4 Log	26
3.2.5 Setting	26
3.2.6 Help	26
3.3 File Panel and Protection Panel	26
3.3.1 File Panel	26
3.3.1 File Panel	
	27
3.3.2 Protection Panel	27
3.3.2 Protection Panel 3.3.2.1 Basic Info	27 27 27
3.3.2 Protection Panel3.3.2.1 Basic Info3.3.2.2 Function Options (function level protection)	27 27 27 27 27



3.3.2.5 Status bar	43
4 The Mechanism of software protection	
4.1 Protect the PE program (application) and DLL	45
4.2 Protect the interpreter and code resource file (Python, PHP, etc.)	46
4.3 Make the protection scheme for your software	46
5 Protection Example & Use Case	
5.1 Protect the Local Executable	49
5.1.1 Fundamental protection to software	49
5.1.1.1 Import Table Protection	49
5.1.1.2 Resources encryption	49
5.1.1.3 Additional data extension	50
5.1.1.4 Compression	50
5.1.1.5 Memory Check:	52
5.1.2 Protect the critical Functions of software with following technology.	53
5.1.2.1 Code obfuscation	53
5.1.2.2 Code Virtualization	56
5.1.2.3 Code encryption (Native)	57
5.1.3 Automatically protection to local executable files by using "Comman	d line"58
5.1.3.1 Generating & Using Map file	58
5.1.3.2 Using the SDK label API to mark the critical functions	62
5.1.3.3 Generate .ssp configuration file	65
5.1.3.4 Protect software with command line	65
5.2 Protect the .Net application	71
5.2.1 Protect the .NET application in fundamental	71
5.2.1.1 Name Obfuscation	72
5.2.1.2 Compression	73
5.2.1.3 JIT encryption	74
5.2.1.4 Remove Strong Name	75
5.2.2 Protect the critical Functions	75
5.2.2.1 Code encryption (.Net)	76
5.2.2.2 Code Obfuscation	78
5.3 Java Program protection:	80
5.3.1 Protection background and introduction	80
5.3.2 Protect to Jar archive	81
5.3.2.1 Deployment	82
5.3.3 War archive protection:	84
Convright @ 2020 Virboy All Pight Pocorved	aga F 109



5.4 Unity 3D Program Protection	20
5.4.1 Introduction	39
5.4.2 Protection Mechanism	39
5.4.3 Windows, Linux, macOS platform protection) 0
5.4.3.1 Protect with Virbox Protector GUI) 0
5.4.3.2 Using Command Line to protect the Unity3D program	€
5.4.4 Unity3D android application) 3
5.4.4.1 Protect android application with GUI) 3
5.4.4.2 Use command line to protect Unity3D apk	96
5.4.5 Unity3D program call Net dll plugin	96
5.4.6 Protection Comparison) 9
5.5 Protect Android application10)2
5.5.1 Normal apk application10)2
5.6 Protect the python based application10)2
6 Note10)5
6.1 Known Issues10)6
7 FAQ10)7
7.1 What is the difference between the soft license edition and dongle edition?	70
7.2 What is the difference between the trial edition and standard edition?	70



1 Overview

1.1 Virbox Protector Introduction

Virbox Protector, is the latest Protector and wrap tool to software developer to protect their software copyright and IP which integrated with multi encryption and protection technology: Virtualization, Obfuscation, Smart compression, Code encryption, Data and resource protection, Detecting Hardware breakpoint, Detecting Memory breakpoint, Memory Integrity Check, etc. It is the powerful protector for software developer to protect their software and critical code, algorithm without additional coding, with easy to use and effortless feature.

Virbox Protector is suitable for the following scenarios and software developers:

- Software developer has established the third party license system or self-developed license system; with Virbox Protector, Developer may enhance the security level of software and integrated with existed license system;
- The software program needs to be protected and distributed to software users without licensing to software user. Developer just need use Virbox Protector to protect the software and distribute to targeted software user.
- 3. What is the difference between Virbox Protector LM and Virbox Protector? Virbox Protector LM, a highly secured, easy to use and without code effort protection wrap tool, is one of critical component in Virbox LM solution, software developer use Virbox Protector LM to protect software and use Virbox LM (Virbox Developer Center) or Virbox Developer Utility to issue the license to the protected software and distribute the software and license to authorized software user. So, software developer may choose either Virbox Protector LM or Virbox Protector to protect software according to the software applied scenario.
- 4. Virbox Protector will only protect the software program. And it will not have impact to the software execution or lib called.

Note: Following edition of Virbox Protector available for software developers to choose: Trial Edition, Virbox Protector Standard Edition, Virbox Protector .Net Edition, Virbox Protector Unity3D Edition, Virbox Protector Java Edition, Virbox Protector Android Edition, Virbox Protector ARM-Linux Edition.

Software developer may use trial edition to test and evaluate the project first, then to select the corresponding license according to your system and software environment respectively.



You can contact Virbox team to get above edition by following way: Tel: +86-10-56730936 Fax: +86-10-56730936-8007 Sales: info@senselock.com; Or visit official site and leave message to us, we will feedback as soon as possible. https://Im-global.virbox.com/detail/virboxProtector.html

1.2 Advanced and Secured Protection Technology

- Virtualization: Code Virtualization & Secured VM function available, the native x86 code is converted into Secured Virtual Machine code and executed inside of VM; In combined with Obfuscation technology, it is effective way to defense static/dynamic analysis tools to debug, reverse engineering to your source code;
- Advanced Obfuscation: Advanced obfuscation functions supported to protect code, critical algorithm etc.
- Smart Compression: High efficiency Compression tools to developer with high performance, powerful shield to against hacker tools and effectively to prevents de-compilation of .NET, PE programs; effective to defense the crack tools and also keep small size of the program after protection.
- Code Encryption: Encrypt the function of your program and only the function is executed the function would be decrypted, with the SMC (Self-Modifying Code) technology.
- Import Table Protection: Hide the import table of the original program to protect the functions called by external program. In this way, to against the reverse engineering analysis and prevent the unpacking of the program.
- *Multi Encryption Scheme* to the selected functions, coding to be protected.

1.3 The program supported to be protected

The Operation System supported

- Windows: Windows 7 and above version
- Linux: CentOS, Ubuntu, Debian-9.4.0
- Mac: OX 10.4 and above version
- Android System (Protect Unity3D apk, .so library), 4.0 and above version supported
- ARM Linux (V7/V8 architecture)



The program/framework language

C, C++, .NET, .Net Core3, Java, Unity 3D, Unreal Engine 4, Delphi XE7 or above version, PB, BCB, C#, VB6.0, Python, Lua, Perl, R, Ruby, PHP, etc.

The plugin and framework supported

AutoCAD ARX, Revit

Development Tool supported

MATLAB, LabView

Executable file

32 bit/64 bit executable file and dynamic link library (DLL)

Elf and .so library

Resources protection supported

The software resources of the program developed based on Unity3D, UE4 engine, can be encrypted and prevent from being extracted illegally.

Below table is the type of the file supported by Virbox Protector:

File Type	System Supported	Architecture	Programming language
.NET	Windows	x86, x64	VB, C#, etc
.NET Core3	Windows, Linux, macOS	x86, x64	C#, VB.net
PE	Windows	x86, x64	C/C++, Delphi, PB、BCB, etc
Unity3D	Windows, Linux, macOS, Android	x86, x64, ARM32	C#, etc
ELF	Linux, Android	x86, x64, ARM32、ARM64	C/C++, etc
Mach-O	macOS	x64	C/C++, Objective-C, Swift
java	windows	x86, x64	Java

Software Protection & Evaluation Process:







- 1. Apply to get trial license for Virbox Protector;
- 2. Download the installation package, and verify the license for trial edition in your computer;
- Protection and Performance evaluation: protect your software or data resource with Virbox Protector to evaluate the protection scheme and performance according to the instruction of User Manual;
- 4. If you are satisfied with the protection evaluation, you can purchase a standard edition with full function.

The Limitation to trial edition:

Trial license for Virbox Protector will be valid within *30 days or 100 times usage*, the software protected by trial edition would be expired in *7* days, no limitation by standard edition;



2 Installation of Virbox Protector

2.1 Installation

For different software you want to encrypt/protect, please select the right license from Virbox. Windows Edition, .Net Edition, Unity3D Edition, Java Edition, ARM-Linux Edition, Android Edition available. Download the corresponding installation package and install on your computer.

After the installation of the Virbox Protector, you will get two software installed in your computer: Virbox Protector & Virbox User License Tool. Virbox User License Tool is the tool to verify the Virbox Protector License. You need to activate your Virbox Protector license and verify the license via Virbox User License Tool before start to protect your software/program.



Figure 2-1

The following chart shows the Virbox Protector installation path:

-bin

| -virboxprotector.exe

| --virboxprotector_con.exe

| └─dsprotector_con.exe

-example

| ---plugin

| | └─demo

| | └─src

| ∟sdk

-help

–plugin

| -anti

Copyright © 2020, Virbox, All Right Reserved.



2.2 License mode of Virbox Protector

Virbox Protector supports following license mode to software developer to choose when they apply trial and evaluate Virbox Protector performance or purchase Virbox Protector later:

Trial License: Cloud based license; which is the easiest way for software developer to apply and get the trial license quickly. The Virbox Protector's trial license can be get by providing your email, we will issue the trial license into your email, use your email and password to login the *Virbox User License Tool* to start the trial. Trial license for Virbox Protector will be valid within *30 days or 100 times usage*, the software protected by trial edition would be expired in *7* days, no limitation by standard edition;

Free trial license apply:

https://Im-global.virbox.com/detail/virboxProtector.html

For official Virbox Protector user (software developer), they can select either "Soft license" or "dongle based license" according to their requirement;

Soft license: Support for account based license both in online/offline environment;Dongle License: Use Virbox EI5 (hardware dongle, order separately) to be the License container of VirboxProtector. Developer can use Virbox Protector at designated computer which have plugged in EL5 Dongle;

All of the license modes support subscription and perpetual license for Virbox Protector.

2.2.1 License Verification with cloud license (For Trial User)

2.2.1.1 Verify license by Virbox User License Tool

Open Virbox User License Tool, sign in your account with your email (the email you provided to Virbox) to verify the Virbox Protector License, then you can open and use Virbox Protector to start the testing and evaluation

Note: After we issued the license into your email account, a password will be sent to your email. You can sign in the Virbox User License Tool with your email account. The password is the password you received in your email box.



	Region: International ~			Version: v2.2.0.38783(2.3.0.38783) FAQ
ම	Cloud Accounts	+		
-	» Offline Licenses			
Accounts	yongqiang.lu@sen	selock	License Info	
Softwares		Virbox Cloud	A User Name: User Login	Offline Licenses
Dongles		User Name:	Email or phone number	Register
Service		Password:	Characters, case sensitive, no spaces OK Cancel	Forget?
H Diagnosis			Create Offline-Bound c2d	File Import Offline-Bound d2c File
Advance				

Figure 2-2

After sign in the account, you can check the detail information of the license here showing in the picture:

	Region: China 🗸			Version: v2.2.0.38783(2.3	3.0.38783) FAQ 🌲 — 🗆 🗙
හ	Summary Available Expire	ed 1501 8 ~	All Developers	s ~	C global search Q III
Accounts		logo 555 080€ ⊘ ▲	79 logo	55555 080 10d79	55555 080 00d79 © ▲
Softwares					
Dongles					
Service					
Ø					



2.2.1.2 Sign in by the Virbox Protector Interface

You can also sign in from the Virbox Protector interface: Sign in the authorized account:



Virbox Protector 2 (64-bit) v2.0.1.12323		- 🗆 X
File Protect Plug-in Log Setting Help		
📼 🐻 🖬	iguration Protect Selected Projects Protect All Project	Authorization Info Authorized Account Login or Pluin Dongle
File/Directory	Welcome Page Authorized Account Login X	
	Account: Account Password: Remember Password Find Password	
Drag file here to protect!	Cancel Login	
	out	

Figure 2-4

You can check the license detailed information by clicking here:

File Protect Plug-in Log Setting Help Authorization Info Authorization Info Account: 150 Is Logout Open File Save Selected Configuration Protect Selected Projects Protect All Project Authorization Detail	
🗁 🗟 🗟 🧭 🧭	
Open File Save Selected Configuration Save All Configuration Protect Selected Projects Protect All Project Authorization Detail	
File/Directory Welcome Page Welcome to use Virbox Protector ! Start Open File Open Directory(Unity3D/Java) Recent Files/Directory .Net_RetroSnaker.exe PE_Tetris.exe C:/Jsers/luyongqiang/Desktop/.Net_RetroSnaker.exe angrybots5.5.3.apk out	

Figure 2-5

You can logout by clicking the "logout" button.



Virbox Protector 2 (64-bit) v2.0.1.12323		- 🗆 X
File Protect Plug-in Log Setting Help		
Open File Save Selected Configuration Save All Con	Accourt	ization Info nt: 68 Logout ization Detail
File/Directory	Sease Info 🛛 Function Options C Protection Options Resource Encryption	
Protect successfully!	Enable (It is recommended that only the resources selected by default be encrypted)	
angrybots5.5.3.apk Protect successfully! If it's an executable program,		Add File
roccession, and an excession programy	File	^
	assets/bin/Data/59027fb62f4ec427891b09546c07f7fa	×
	assets/bin/Data/ab079e63059f340088f6b9c0e209bee1	×
	assets/bin/Data/c8cc60236143f4ec59b76421156c9eda	×



2.2.2 License Verification with soft license

2.2.2.1 Use Virbox Protector in online environment

When you use the Virbox Protector in online environment, you can sign in the account that have already issued license. After you start Virbox Protector, the software license will bind to your hardware machine automatically.

	Region: International \sim	Version: v2.2.0.41922(2.3.0.41922) FAQ 🌲 — 🗆 🗙
හි	🛆 Cloud Accounts 🛛 +	
	Offline Licenses	
	» elock	
Accounts	1. 968	License Info
ŵ		A User Name: yong om(International)
Softwares		☆ Total Licenses: 8
8		Available Licenses: 2
Dongles		Expired Licenses: 6
		Refresh Set As Default Settings Logout
R		Upload Offline-Unbound c2d File
Diagnosis		
Advance		

Figure 2-7



හ	Region: International ~ Summary Available Expired	Version: v2.2.0.419 yongqiang.lu@senselock.com(International) ~	022(2.3.0.41922) FAQ ♠ All Dev€opensormal search	×
Accounts		yongqiang.u@ysenseiock.com(international) ↔	All Devejope <u>goves</u> search	
Softwares Dongles				
<u> </u>				

Figure 2-8

2.2.2.2 Use Virbox Protector in offline environment

If the Virbox Protector is used in offline environment, you need to use the following step to activate the license of your offline machine with a computer that can connect to internet (Online computer). Both computer (Online and offline computer) need to install Virbox User License Tool.

Generate c2d file on the Offline computer

Open Virbox User License Tool, click "Accounts",

Click "Offline",



Figure 2-9

Generate offline bind c2d file, and save the .c2d file.



	Region: International ~				Version: v2.2.0.34657(2.3.0.34657)	FAQ	÷ -	-
ම	Cloud Accounts	+						
	» Offline							
Accounts			License Info					
			8 User Name:	Offline				
Softwares			Total Licenses:	1				
ð			Available Licenses:	1				
			Expired Licenses:	0				
			Refresh					
J Diagnose			Create Offline-Bound c2d	File	rt Offline-Bound d2c File			
Advance								

Figure 2-10

	Region: China 🗸	Version: v2.2.0.33187(2.3.0.33187) FAQ 🌲 — 🗆 🗙
හි	Cloud Accounts +	
-	» Offline	
Accounts		License Info
1		🚖 User Name: Offine
Softwares		Info 🗶 0
S		0 Created Offline c2d file sucessfully!
Dongles		0
20		
Service		OK
m		Create Offline-Bound c2d File Import Offline-Bound d2c File
10		The sound czd File
Diagnose		
. *		
O C		

Figure 2-11

After you have created c2d file successfully. You need to copy this c2d file to the online computer.

• Create d2c file on the computer Online

Also need open Virbox User license Tool on online computer. Click "+" to login your account that have already have license.



	Region: International ~	Version: v2.2.0.34657(2.3.0.34657) FAQ 🌲 — 🗆
ම	🕒 Cloud Accounts 🛛 +	
-	Offline	
4	yongqiang.lu@senselock	
Accounts	» yongqiang.lu@senselock	License Info
Softwares		A User Name: yongqiang.lu@senselock.com(International) Virbox Cloud User Login X
\$		User Name: Email or phone number Register
Dongles		Password: Characters, case sensitive, no spaces Forget?
Service		OK Cancel gout
R		Upload Offline-Unbound c2d File
Diagnose		
A dvance		

Figure 2-12

Click "Software",

	Region: International ~			Ve	rsion: v2.2.0.34657(2.3.0.34657)	FAQ 🌲 —	
න	Summary Available Expired	All License Form \sim	All Developers ~		C gl	obal search	Q 88
Accounts	2010 9900 • @						
Softwares]					
Dongles							
Service							
K Diagnose							
రి							
Advance				Cloud	💩 Offline 🛛 💩 Offline(unbind)	Dongle	🏶 Net

Figure 2-13

Double click the license, the detail information of the license will show:



ම	Region: International ~		Version: v2.2.0.34657(2.3.0.34657) FAQ 🌲 — 🗆 🗙
Accounts	logo	Software Introduction No software description	
Softwares	2010 0900 00006	 License Details License Source: License Form: 	yongqiang.lu@senselock.com(International) Offline(bound) Bind Info Download Offline-Bound d2c File
Dongles		Dicense Status:	Available
Service		 Start Date: Expired Date: First Use Date: 	2018/07/18 00:00:00 2018/07/27 23:59:59 2018/07/18 10:52:13
<i>K</i>		Expiring Span:	0Days 0Hours 0Minutes 0Seconds
Diagnose		 Concurrence: Concurrence Type: 	
Č Advance		more	

Figure 2-14

Click "Download Offline Bound d2c file"

	Region: International \vee	Version: •	/2.2.0.34657(2.3.0.34657) FAQ 🌲 – 🗆 🗙
හි	Return		
Accounts		Software Introduction No software description	
		Download Offline-Bound d2c File	
Softwares	2010	Cloud Account: yongqiang.lu@senselock.com(International)	
	0900000000000006	Developer: 09000000000006	Bound d2c File
*		Product Name: 2010	
Dongles		c2d File Path: Browser	
Service		d2c File Path: C:\Users\liuguoquan\Desktop\sense_offline_bind_201807 Browser	
Diagnose		Open d2c file directory Start Cancel	
0		Concurrence: Concurrence Type:	
d Advance		more	

Figure 2-15

To generate a D2C file, you need to import the c2d file you generated from the offline computer in last step, Click "**Start**",



ගි	Region: International ~ Return			Version: v2.2.0.35	i139(2.3.0.35139) FAQ 🌲	- 🗆 ×
Accounts	logo	Software Descriptio No software descriptio Download Offline-Bound d2c Fi	on	×		
Softwares Dongles	2010 0900000000000	Developer: 09000000000 Product Name: 2010	i@senselock.com(International))00006 uoquan\Desktop\sense_offline_bind_2018	107 Browser	ad Offline-Bound d2c File	
Service		d2c File Path: C:\Users\liug Open d2c file directory	uoquan\Desktop\sense_offline_bind_2018	Applied Donel Cancel]	
Diagnosis Advance		 ▲ Concurrence: ▲ Concurrence Type more 	 2:		-	

Figure 2-16

You can select the path to generate d2c file, here I put it to desktop. If the file is generated successfully, it will show "**Applied Done**", such as in the picture above showed.



Figure 2-17

This is the d2c file generated.

Note:

1. Click "open d2c file directory", also will show the path of the generated file.

2. The valid time of this d2c package is 24 hours, please complete binding process in time.

Verify d2c file on the Offline computer

Now we need to copy d2c file from online computer to the computer offline and complete license verification.

Copy the **d2c** file generated from the computer (Online computer). Import it in to the offline computer.



	Region: China 🗸	Version: v2.2.0.33187(2.3.0.33187) FAQ 🌲 — 🗆 🗙
හි	Cloud Accounts +	
	» Offline	
Accounts		License Info
ŝ		A User Name: Offline
Softwares	Import Offline-Bo	ound d2c File 🗶
Dongles	d2c File Path	Browser
Service		Import Cancel
J Diagnose		Create Offline-Bound c2d File
bilgilosc		
C °		

Figure 2-18

	Region: International ~	Version: v2.2.0.35139(2.3.0.35139) FAQ 🌲 — 🗆 🗙
හ	Cloud Accounts +	
	» Soft License	
Accounts		License Info
		A User Name: Soft License
Softwares	Import Offline-Bo	ound d2c File 🔀
\$	d2c File Path:	
Dongles	C:\Users\Windov	vs-10-EN\Desktop\sense_offline_bind_20180718135448.d2{ Browser
		Import successfully!
≥ ⊙		Import Cancel
		import
K Diagnosis		Create Offline-Bound c2d File Import Offline-Bound d2c File
రి		Activate Win Go to Settings to

Figure 2-19

It will popup "Import successfully" message after you imported the file successfully.

Then you have activated the license on the offline computer successfully, and you can use it to encrypt your program now.

2.2.3 License Verification with EI5 dongle (For official user use dongle license)

If you purchased the Virbox Protector with a Virbox EI5 dongle, after installation you need to insert the dongle

Copyright © 2020, Virbox, All Right Reserved.



on your PC for license verification. Then you can use Virbox User License Tool to check the license you have subscripted. As the figure shown below:

	Region: China 🗸		Version: v2.2.0.38	783(2.3.0.38783) FAQ 🌲	- 🗆 ×
හ	Summary Available	Expired 10050000004 \sim All Devel	opers ~	C global search	Q BB
Accounts		100 sensetest hantititititit	logo se	logo sensc ⊘ ■	
Softwares					
Dongles					
Service					
Å Diagnosis					
Advance		👗 Cloud 🛛 🙆 Offline(unbou	nd) 🧔 Offline(bound)	🚳 Offline(Local) 🛛 🙀 Dong)	e 🋞 Net

Figure 2-20

You can double click that dongle icon for license detail information.

	Region: China ~		Version: v2.2.0.38783(2.3.0.387	33) FAQ	۰	-	×
ම	Return						
		Software Description					
		No software description					
Accounts	logo						
		License Details					
Softwares	100	Ø License Source:	100J04				
	sens	Ø License Container:	Dongle				
S		License Status:	Available				
Dongles		Ucerise Status.	Available				
		Start Date:					
≥ ⊙		O Expired Date:	Permanent				
Service		First Use Date:	2019/01/10 11:00:04				
		🔀 Time Span:					
K		Remain Times:					
Diagnosis							
		🔬 Concurrence:					
		🙏 Concurrence Type:	Process				
		more					
C°							
Advance							

Figure 2-21



3 Protection Function Introduction

3.1 Main Menu of Virbox Protector

The main menu shown as below: includes 3 areas:

Menu Bar: consist of: File/Protect/Plug-in/Log/Setting/Help functions;

Tool Bar: Open File/ Save Selected Configuration/Save All Configuration/Protect Selected Projects/Protect All Projects;

File/Directory Panel and Protection Panel

These functions and options in the menu, includes the Tab and Panels will be introduced and explained in this chapter.



Figure 3-1

3.2 Menu Bar

3.2.1 File

Open File: Click "*Open File*", you can select .Net, PE, ELF, MachO, Arm Linux and Android .so lib file. And load the selected files into the left panel, the File/Directory Panel.



Note: If the xxx.map file located in the same directory of the program file being protected, these xxx.map file will be loaded with the program file automatically. And the name of the functions will also be loaded and listed in the File Panel. The map file generated by VS, VC, BCB, Delphi compiler is supported at present.

Open Directory: Here you can open a whole directory, to open Unity3D directory only.

Recent/Batch Projects: Here you can reopen the recent protection project quickly, or batch file protection project. Up to 5 recent projects can be recorded.

The recent protected program would be loaded and listed into the File Panel.

If you want to save the project setting and path of multiple file, you can save those configuration into a project file **xxx.vbpsIn** by clicking "**Save Batch/project**". Then you can reload this project file for next protection.

Save Batch Projects: You can use this function to save all of path of the file, but the configuration would not be saved. If you have changed the configuration and want to save them, you need to click "save the selected configuration" or "save all of the configuration".

When you reopen the Virbox Protector, you can drag in xxx.vbpsln to open the project, the saved file and configuration would be loaded if the location of the file haven't changed.

Exit:

Close Virbox Protector and exit.

3.2.2 Protect

Parse selected project (File):

Select one or multi file which listed in the File Panel, you can parse these file by clicking "**parse selected project**" button. The file need to be parsed correctly before protection.

Parse all project

Parse all of the files in the project, no matter how much you have selected.

The purpose of parsing is to reload the configuration status you saved.

Save selected configuration

Configuration means the function options, protection options, which you selected to the protected file,

You can save the configuration of the Function options, Protection options, Message by clicking "save selected configuration"

Save All configuration

Save all of the protection configuration of the project, no matter how many file you have selected. Corresponding error report or error code will show, if the configuration is not correct and you can't save the configuration.

Protect selected project

Copyright © 2020, Virbox, All Right Reserved.



You can protect the selected file in the file list by clicking this option. If the configuration is not correct, it will remind you corresponding error report or error code.

Protect All project:

No matter how much file you selected in the file list, you can protect all the file by clicking this option.

These function are also available in tool bar, you can also use those function from the tool bar.



Figure 3-2

3.2.3 Plug-in

Open "DSProtector", which is the plugin tool for data resources protection, such as jar archive, .py file, pyc file etc.

"DSProtector"(Hereinafter referred to as DS Protector) is the plugin unit to protect the data resources provided by Virbox, software developer may use DS Protector to protect data file and encrypt related data resources together with protected software program.

Please noted that DSProtector does not support the data resources protection which from Linux and Mac system currently.

Ø DSProtector	(1.3.1.11480)		-		×		
File Settings	Help						
SSP File:							
File Filter:	ilter files when add directory, eg:txt;php;html;						
Output Directory:	Optional; we will always bakup(*bak) in the same directory						
File		Output File	Status				
<					>		
Add File	Add Directory Remove Selected Remove All		Pro	tect			
Welcome to use	DSProtecotr!						

Figure 3-3



3.2.4 Log

Show log dialog

Log dialog will show the log file when you are protecting the software. You can save the log by clicking "save", to save the log to other directory.

Open local log directory: Open the log directory.

3.2.5 Setting

Language setting:

Both Chinese and English are supported. To change the language of the interface of the software you need to restart the software. You can restart instantly or next time you open the software.

3.2.6 Help

About: It will show you the technical support email and website.

3.3 File Panel and Protection Panel

3.3.1 File Panel



Figure 3-4

After you drag the software to be protected in to the file panel, the software basic information will show in the



Basic info page.

You can select one or more software and right click the software to select the corresponding function.

- Parse (software or file)
- Save configuration
- Protect software
- Show containing folder
- Set output directory for protection
- Copy the protection option of this file to other selected files:

Select 2 more software, right click software and choose "copy the protection option of this file to other selected files", and use this setting to other software, the other software will have the same configuration.

Close project:

Right click the selected software, and choose close, you can close the program with saving the current configuration and also can exit without saving, or cancel the operation.

3.3.2 Protection Panel

- Basic info
- Function Option
- Protect Option
- Resources Encryption

3.3.2.1 Basic Info

Basic info will show you the basic information of the loaded software, File/directory path, file creation time, Last configuration Modified time, Last Accessed Time, Application Type (PE or .Net, etc).

3.3.2.2 Function Options (function level protection)

Virbox Protector supports to protect the software application to the functions level and provides several protection mode to developer select to protect the critical functions.

Add the Functions to be protected

Copyright © 2020, Virbox, All Right Reserved.



Function Option page lists all the functions in your application, you can select and protect the critical functions in your application or program in this page, you can select **No Protected**, **Virtualization**, **Code Obfuscation**, **and Code encryption** mode to protect the selected functions.

When you click the function in the software, the protection mode, function name, function address and assemble code will show in this page. And the quantity of total functions, total protected functions and the quantity of every function protection type will be shown in this page.

As shown the figure below:

This tag will list all of the function module which have been parsed (There are little difference between managed code and un-managed code), you can select the corresponding protection option.

- Managed code: The function name is "Name Space + Class Name+ Function Name"
- Un-managed code: The function name is the "va" value of the function.

Virbox Protector x64 (1.5.1.11511)		– 🗆 X
File Protect Plug-in Log Setting Help		
Open File Save Selected Configuration Save All Con	ifiguration Protect Selected Projects Protect All Project	
File/Directory		
	Seasic Info Function Options Crotection Options	
Save successfully!	Total Qty: 410 Total Protected:8	Add Functions
.Net_RetroSnaker.exe Save successfully!	Obfuscation: 1 Virtualization: 1 Encrypted:6	Add Punctions
	♥ 403EAE	×
	00 40DF90	
	40DFAF	
	40E00C	
	E 406960	
	E 40EA89	
	40ED88	
	6 40EE22	

Figure 3-5



Virbox Protector Standalone x64 (1.4.2.9340) File Protect Plug-in Log Setting Help					- 🗆 X
Copen File Save Selected Configuration Save All Con		Ö ect All Proje	ct		
File/Directory .Net_RetroSnaker.exe Protect successfully!	Base Info Function Options Total Oty: 79 Total Protected: 1	Prote	ection Options		
PE_Tetris.exe Parse successfully!	Obfuscation:0 Virtualization: 0	Encrypte	ed: 1		Add Function
apache-tomcat-9.0.8.exe Parse successfully!	❤ 🚺 食食蛇 ❤ 🧿 Program		Function Name: Address:	void Main() 40538D	
	void Main()		Ptortection Type: Note: Assembly Code	Encryption	
			40538E 5A289B	0000	call 0x0a00009b ldc.i4.0
			405394 16289C	0000	call 0x0a00009c
			405399 0A734A 40539E 06289D		newobj 0x0600004a call 0x0a00009d
			4053A3 0A		ret
E:/Test-Program/.Net_RetroSnaker.exe					.NET AnyCPU i386



Following steps will describe how to set and select options for above setting:

[Note: If xxx.map file existed in the same folder of software be protected, Virbox Protector will load this map file automatically and list functions in the main menu, currently the map file support be protected includes the map file created by VS, VC, BCB, Delphi compilers.]

Note: Usually, software developer need to balance the software execution performance and protection level before software protection. Be careful to select and protect these frequently called functions, since it will decrease software execution performance after protection/encryption.

Click "Add Function" (See picture attached), you can enter the "Add Function" window.



iration Protect Selected Projects Protect All Proj	ect		
	tection Options		
Total Qty: 79 Total Protected: 1 Obfuscation: 0 Encryp	ted: 1	Add Funct	ion
 ✓ ① 会長起 ✓ ④ Program 図 void Main() 	Function Name: void Main() Address: 40538D Ptortection Type: Obfuscation Note: Assembly Code 405393 405393 0A 405394 16289C0000 405399 0A734A0000 405398 66289D000 405393 0A	call 0x0a00009b ldc.i4.0 call 0x0a00009c newobj 0x0600004a call 0x0a00009d ret	

Figure 3-7

The **Virbox Protector** will list all of the functions used in this software in the left panel.

Click "OK" to confirm the protection and the function you selected will show up in the function list:

V Add Functions		— 🗆 X
Global Search Functions		Add Function Analysis
Total Protected: 8 / 410 Obfuscation: 1 Virtualization: 1 Encrypted: 6		
Tip: The unsupported protect mode of the functions will be known after protecting th	e program! Only show fur	nctions that were set the unsupported protection way
Name	Run Times	Protection Type Other
00 40E9E		No protected V Bytes too little(.Net<
V 40EE98		No protected V Bytes too little(.Net<
E 40EE22		Encryption -
40ED8B		Encryption 💌
E 40EA89		Encryption 🔻
6 40E9E0		Encryption 👻
6 40E00C		Encryption 👻
40DFAF		Encryption 🔻
M 40DF90		Obfuscation 🔻
<		
Total Show Functions: 410 Support Ctrl/Shift Selection and Righ-Click Menu		Cancel Ok

Figure 3-8



😏 Basic Info	Function Options	Protection Options	
Total Qty: 410	Total Protected:8		
Obfuscation: 1	Virtualization: 1	Encrypted:6	Add Functions
V 403EAE			
M 40DF90			
40DFAF			
40E00C			
6 40E9E0			
6 40EA89			×
6 40ED8B			
6 40EE22			

Figure 3-9

Select the function to be protected/encrypted with different protection mode:

Global Search Functions			Add Function Analysis
•			Add Function Analysis
Fotal Protected: 8 / 410 Obfuscation: 1 Virtualization: 1			
Fip: The unsupported protect mode of the functions will be	known after protecting the program! Only show fi	unctions that were set the u	insupported protection way
Name	Run Times	Protection Type	Other
😳 40EE9E	-	No protected	Bytes too little(.Net<10, other<
VP 40EE98	-	No protected	Bytes too little(.Net<10, other<
6 40EE22		Encryption •	
6 40ED8B	-	Encryption	
E 40EA89	-	Encryption	,
E 40E9E0	-	Encryption	,
E 40E00C	-	Encryption	
E 40DFAF	-	Encryption	
10DF90		Obfuscation	Ē
40DEB0		No protected	

Figure 3-10

Also the total protection option you selected would be counted and show in the interface:



Global Se	earch Functions			A	nalysis
Total Prot	tected: 3/138 Obfuscation: 1 Virtualization: 0 Encrypted:	2			
Tip: The u	unsupported protect mode of the functions will be known after p	rotecting the program! Only show fur	nctions that were set the u	nsupported protection way	
Name	~	Run Times	Protection Type	Other	^
	E void StartGame()	-	Encryption •	•	
	📀 void set_Width(int)		No protected	Bytes too little(.Net<10, oth	er<
	M void set_Target(Point)		Obfuscation	-	
	📀 void set_Height(int)		No protected	Bytes too little(.Net<10, oth	er<
	📀 void set_GameState(State)		No protected	Bytes too little(.Net<10, oth	er<
	😳 void set_Direction(Direction)		No protected	Bytes too little(.Net<10, oth	er<
	🐶 void set_Body(List`1 <point>)</point>		No protected	Bytes too little(.Net<10, oth	er<
	😳 void RunAgain()		No protected	•	
	🌝 void OneNewSnake()		No protected	•	
	📀 void InitTarget()		No protected		~

If the protection process failed, Virbox Protector will prompt "Some of protected functions doesn't support the protection mode you selected", you need to change to other protection mode to protect the function.

Protection Mode:

To protect the specified functions of the software, following functions protection mode can be selected: **No protected**, **Code Obfuscation**, and **Virtualization**, **Code Encryption**.

For the functions which is called frequently, select "No Protection" option, since if you protect the functions which is called frequently, it will decrease software's running performance when software is executed;

lame	Run Times	Protection Type	Other	^
💹 void set_Index(int)	1086	Obfuscation	•	
void Dispose(bool)		Encryption	•	
✓ 🧿 SnakeIO				
void SaveTaget(string)		Encryption	•	
😳 void SaveSnake(string)		No protected	•	
😳 void LoadTaget(string)		No protected	· ·	
😳 void LoadSnake(string)		No protected	•	
😳 void .ctor(Snake)	1	No protected	•	
😳 Point Convert(string)		No protected	•	
> 🧿 Snake				



Select "**Obfuscation**": Virbox Protector will translate the code instruction into a stream of

pseudo-code that neither the machine nor the human can identify it. When the pseudo-code



executed, the software will translate and interpret to restore the code into the original code. To let it be executable.

Virbox Protector support the obfuscation for x86/ARM .Net il serial instruction.

Code Protection Mechanism:

Interference the original instruction and prevent the code from being static analyzed.

The Benefit:

Prevent from anti-compiled and make it more difficult to analysis the code.

The Weakness:

Partial Impact to the execution performance.

Add Functions	;			_		×
Global Search Fur	nctions				Analysis	
Total Protected: 3						
Tip: The unsuppor	rted protect mode of the functions will be known after protecting the progra	m! Only show functi	ons that were set the uns	supported protection way		
Name	~	Run Times	Protection Type	Other		^
✓ 🚫 贪食蛇						
🗸 🧿 Snak	keLabel					
📀 -	void set_Index(int)	1086	No protected 🔹			
M	void Dispose(bool)	-	Obfuscation 🔻			
🗸 🧿 Snak	keIO					
M	void SaveTaget(string)		Obfuscation 🔻			
	void SaveSnake(string)		No protected 🔹			
	void LoadTaget(string)		No protected 🔹			
	void LoadSnake(string)		No protected 🔹			
	void .ctor(Snake)	1	No protected 🔹			~
Total Show Functi	ions: 79			Cancel	Ok	



Select "Virtualization": Virbox Protector will compiles original instructions into virtual instruction and run them in the specified virtual machine. There are certain format requirements and limitation for instructions, and some functions may not be protected;

Protection Mechanism:

Hide the original instruction, prevent the code logic from being analyzed.

The Benefit:

Highly secured protection mode, the original code logic almost can't be get by analysis.

The Weakness:

Performance impact to software execution Copyright © 2020, Virbox, All Right Reserved.



Add Functions							×
Global Search Functions				Add	Function	Analysis	;
Total Protected: 2/340 Obfuscation: 0 Virtualization: 2	Encrypted: 0						
Tip: The unsupported protect mode of the functions will be k	own after protecting the program!	Only show functi	ons that were set the uns	supported prot	ection way		
Name	Rur	Times	Protection Type	Other			^
600 40EE22			No protected 🔹				
V 40ED8B			Virtualization 👻				
V 40EA89			Virtualization 👻				
40E9E0			No protected 🔹				
🧓 40E00C			No protected 🔹				
📀 40DFAF	-		No protected 🔹				
00 40DF90	-		No protected 🔹				
00 40DEB0			No protected 🔹				
🕜 40DE 1F	-		No protected 🔹 🔻				
00 40DD57			No protected 🔹				v
Total Show Functions: 340				Cancel		Ok	

Figure 3-13

• Code Encryption:

Code encryption, it encrypt the original function of the program by SMC (Self-Modifying Code) technology and the function will be decrypted only when the program executed.

Global Search Functions				Analysis
otal Protected: 3/79 Obfuscation: 2 Virtualization:0 Encrypte	d: 1			
ip: The unsupported protect mode of the functions will be known after	er protecting the program! Only show for	unctions that were set the u	insupported protection w	ау
Name	Run Times	Protection Type	Other	
▼ 🚫 ☆食蛇				
✓ 🧿 SnakeLabel				
void set_Index(int)	1086	Obfuscation •	•	
void Dispose(bool)		Encryption -	- T	
✓ 🧿 SnakeIO				
void SaveTaget(string)		Encryption	•	
🌝 void SaveSnake(string)		No protected	•	
🐶 void LoadTaget(string)	-	No protected	•	
🕡 void LoadSnake(string)	-	No protected	•	
📀 void .ctor(Snake)	1	No protected	•	

Figure 3-14



Note: For **.Net** Programs, Support function protection options includes: **No protected, Obfuscation, Encryption**;

For Other Programs (PE or native program): Support Function protection options: No protected, Obfuscation

and Virtualization, Encryption.

Protection Mechanism:

Prevent from being unpacking, and prevent the program from being dumped directly.

The Benefit:

Almost no impact to software performance.

The Weakness:

Low Security: It is possible be decrypted and by analyzed to the protected functions.

V Add Functions				
Global Search Functions Total Protected: 3 / 79 Obfuscation: 2 Virtualization: 0 Encrypted: 1				Analysis
Tip: The unsupported protect mode of the functions will be known after protecting the progr	am! Only show funct	ions that were set the un	supported protection way	
Name	Run Times	Protection Type	Other	^
✓				
😡 void set_Index(int)	1086	Obfuscation 🔻		
E void Dispose(bool)		Encryption 💌		
SnakeIO				
void SaveTaget(string)		Encryption -		
😳 void SaveSnake(string)		No protected 🔹		
🐶 void LoadTaget(string)		No protected 🔹		
😳 void LoadSnake(string)		No protected 🔹		
😳 void .ctor(Snake)	1	No protected 🔹		~
Total Show Functions: 79			Cancel	Ok

Figure 3-15

Tips: You can click the icon as showing in the picture above to open the function.

Performance Analysis

After you selected the protection mode for the function, you can start to analysis execution performance by clicking "*Analysis*" before finalize your software protection scheme. The Analysis function will show you the software execution performance and the calling times of the protected functions when execution. After you completed analysis, the functions called times will be displayed in the middle of the panel. For how


long time you run the program you are protecting depends on your actual requirement.

Note: If the program you are analyzing is **DLL** libs, please start the main program. We currently support EXE/ELF program and DLL (Dynamic Link Library) and so library protection.

	_	Log Setting Help		Ci.				Authorizati	on Info	
File	Add Fun	ni l <u>-</u>						-		× luin D
e/Direc	Global Sear	ch Functions							Analysis	
Unit	Total Protec	tted: 2/138 Obfuscation: 1 Virt	ualization:0 Encrypted: 1	L				L		
ive suc	Tip: The uns	supported protect mode of the func	tions will be known after pro	otecting the progra	m! Only show fu	nctions that were set the	unsupported pro	tection way		nction
] .Net	Name	`			Run Times	Protection Type	Other			^
PE_1	× 🚫 🏦	食蛇								
arse su	> 🥝	State								
	~ <u>(</u>	SnakeLabel								
		🌕 void set_Index(int)				No protected	•			
		📀 void InitializeComponent()				No protected	•			
		🔟 void Dispose(bool)				Obfuscation	-			
		📀 void .ctor()				No protected	-			
		🌝 int get_Index()				No protected	Bytes too litt	le <mark>(</mark> .Net<10,	other<	
	> G	SnakeIO								
	> (6	SnakeControl								~
	Total Show	Functions: 138 Support Ctrl/SI	nift Selection and Righ-Click	Monu			Cance	J	Ok	

Figure 3-16

🕑 Virbox Protector Sta	ndalone x64 (1.4.2.9340)		- 0	\times
File Protect Plug-in	Log Setting Help			
	Fa Fa 🤝 🛇			
Open File Save Sele	😢 Add Functions		– 🗆 X	
File/Directory				
.Net_RetroSnak	Global Search Functions		Analysis	
Parse successfully!	Total Protected: 3/79 Obfuscation: 2 Virtualization: 0 Encrypted: 1		Add Function	1
	Tip: The unsupported protect mode of the functions will be known after protecting the pro-	ogram! Only show functions that were set the unsupported protect	.on way	
	Name	Run Times Protection Type Other	^	
	▶ 💦 ☆黄蛇			
	✓ ○ SnakeLabel			
	🚺 void set_Index(int)	1079 Obfuscation 💌		
	void Dispose(bool)	Encryption		
	> G SnakeIO			
	> 🧿 Snake			
	> 🚺 Properties			
	> 🧿 Program			
	> 🧿 frmGame			
	> 🧿 frmFactory		~	
	Total Show Functions: 79	Cancel	Ok	
L L				

Figure 3-17



🕑 Add Functions			—	
Global Search Functions				Analysis
Total Protected: 3/79 Obfuscation: 2 Virtualization: 0 Encrypted: 1			,	
Tip: The unsupported protect mode of the functions will be known after protecting the	program! Only show fun	ctions that were set the un	supported protection way	
Name	Run nimes	Protection Type	Other	^
M void set_Index(int)	1086	Obfuscation 🔹		
void Dispose(bool)		Encryption -		
✓ ⁽ e) SnakeIO				
void SaveTaget(string)		Encryption -		
📀 void SaveSnake(string)		No protected 🔹		
📀 void LoadTaget(string)		No protected 🔹		
📀 void LoadSnake(string)		No protected 🔹		
📀 void .ctor(Snake)	1	No protected 🔹		
😳 Point Convert(string)		No protected 🔹		
> 🧿 Snake				~
Total Show Functions: 79			Cancel	Ok

Figure 3-18

Function search:

V	Add Functions			—		×
	Slobal Search Functions Total Protected: 1/79 Obfuscation: 0 Virtualization: 0 Encrypted: 1				Analysis	
	Tip: The unsupported protect mode of the functions will be known after protecting the prog	ram! Only show funct	tions that were set the un	supported protection way		
	Name	Run Times	Protection Type	Other		
	> 🔃 会会蛇					



After you entered the keyword of the function, Virbox Protector will list the functions which contain the keyword, fuzzy query supported.

3.3.2.3 Protection Options

Protection Option setting will be different for different program, This "**Protection Option**" function has little difference to PE (local program) and .NET application due to difference of PE and .NET technology and Gaming software based Unity3D. Developer can select and setup these "**Protection Option**" in actual project.

For .NET program, following options could be set and selected:



Virbox Protector 2 (64-bit) v2.0.1.12323 Protect Plug-in Log Setting Help			- 0
	onfiguration Protect Selected Projects Protect A) I Project	Authorization Info Account: 1 368 Logout Authorization Detail
File/Directory	Basic Info Basic Info Compute Info	Protection Options	^
Protect successfully! angrybots5.5.3.apk Protect successfully! If it's an executable program,	Output File/Directory	.Net_RetroSnaker.ssp.exe	•••
Net_RetroSnaker.exe Parse successfully!	Compression		
PE_Tetris.exe	JIT Encrypt	\checkmark	
Parse successfully!	Name obfuscation	\checkmark	
	✓ Plug-in		
	✓ Anti Debug Option	\bigcirc	
	Detecting Hardware Breakpoints		
	Detecting Memory Breakpoints		
	Memory Integrity Check		

Figure 3-20

For **local program** (PE), following options could be set and selected:

Virbox Protector 2 (64-bit) v2.0.1.12323			- 🗆 X
File Protect Plug-in Log Setting Help			
- 5 5	1 🔿	(<i>J</i>	Authorization Info
	nfiguration Protect Selected Projects Protect	\checkmark	Account: 15 968 Logout Authorization Detail
	ingulation Protect selected Projects Protect	Airfojett	Authorization Detail
File/Directory			
Unity3D	😔 Basic Info 🛛 🚱 Function Options	Protection Options	
Protect successfully!	✓ Output Info		^
angrybots5.5.3.apk	Output File/Directory	PE_Tetris.ssp.exe	•••
Protect successfully! If it's an executable program,	 Protection Options 		
.Net_RetroSnaker.exe Parse successfully!	Import Table Protection		2
PE_Tetris.exe	Compression		
Parse successfully!	Memory Check		
	Resource Section Encryption		
	✓ Plug-in		
	✓ Anti Debug Option	\bigcirc	
	Detecting Hardware Breakpoints		
	Detecting Memory Breakpoints		
	Memory Integrity Check		
	Ƴ ds	\bigcirc	Open DSProtector
	Password	••••••	~
E:/Test-Program/PE_Tetris.exe	1.		PE 32-bit i386

Figure 3-21

For Unity3D file, following options will be displayed and selected:



	nfiguration Protect Selected Projects Protect All	Project		Authorization Info Account: 15010303968 Log Authorization Detail	out
File/Directory Unity3D Protect successfully1 angrybots5.5.3.aak	Basic Info Function Options Assembly jiamigou_Test_Data/Managed/Assembly-	Protection Options CSharp-firstpass.dll	Resource Encryption	+	×
Net_RetroShare.exe Parse successfully! PE Tetris.exe	jiamigou_Test_Data/Managed/Assembly- ✓ Output Info Output File/Directory				×
Parse successfully!	 ✓ Plug-in 	septement of			
	 Anti Debug Option 	\bigcirc			
	Detecting Hardware Breakpoints				
	Detecting Memory Breakpoints				
	Memory Integrity Check				



Click the "Protection Option"

- 1. **Output File**: Here, it will show the location of the protected/encrypted program. And you can change the output file path of the protected software.
- 2. Compression: Compression means to compress the application after protection and reduce size of application, it is also prevents static anti-compile the software application by hacker who use static anti-compiler tools. When you select Compression: It will keep and control the size of protected software application not too big size. Also will enhance protected software's security level after compression; for the program with big size. This function would be obvious to make the program smaller size.

The original purpose of the compression function is not compress the software size, it will encrypt the code and the data segment and hide the original import table and relocate information, and compressed the original data at same time.

Protection Mechanism:

This function will pack the original data segment with data package and compress the file, replace the original code entry with the packer code. The data segment and code segment will be retrieved when the program is executed, and relocate and to execute the program.

The Purpose:

Prevent the static anti-compile and prevent the program being patched

The Benefit:

Hide the code, data and file structure information of the program, protect the software in overall.



High efficient when the program is run, and only relatively weak performance loss when the program is loaded.

The Weakness:

When the packer code is executed, the code segment and data segment may be retrieved and be dumped.

3. JIT encryption:

.Net JIT encryption will encrypt all of the method IL instruction of the .Net Program, and only in the JIT compile process of the .Net Virtual Machine the instruction will be decrypted, This can be used to prevent static anti-compile and prevent the IL code being Dumped in memory.

JIT encryption will encrypt all of the method in default and enhance the security level of the source code after protection.

JIT encryption .support inheritance, event, reflection, recursive call which is not supported in general encryption solution.

- 4. **Name Obfuscation:** Select this Option, developer can obfuscate the program file name and transforming software program name into the pseudo code which cannot be identified by use of Static Anti-compiling Tools and then convert these Pseudo code into original program name when execute the protected software.
- 5. Anti-Debug option: you can use the following anti-debug option by clicking this button.

The anti-debug function, including Detecting Hardware Breakpoint, Detecting Memory Breakpoint, Memory Integrity Check. To prevent your program/application from being debugged by the tools, such as: ollydbg or Windbg. **The platform supported**: Windows, Linux, ARM Linux, Android so library and Android Unity3d application.

- Detecting Hardware breakpoint: when this function is enabled to the protected software, the program will stop execution if memory access breakpoint and memory write breakpoint has been detected.
- 7. Detecting memory breakpoint: This function will protect your software by exit the program if the program has been detected to be setting memory breakpoint.
- 8. Memory Integrity Check: when this function enabled, the program will be terminated execution if the memory modification has been detected (e.g.: Being modified by debugger).

For Linux, ARM Linux, Android so library and Android Unity3D program:

The anti-debug plugin function supports to detect the debug tool, and prevent the program being anti-compiled, such as: gdb, IDA, etc tool.



le/Directory	🕞 Basic Info 🛛 💀 Function Options 😳	Protection Options	
Unity3D rotect successfully!	✓ Output Info		
angrybots5.5.3.apk	Output File/Directory	PE_Tetris.ssp.exe	•••
rotect successfully! If it's an executable program, .Net_RetroSnaker.exe	 Protection Options 		
arse successfully!	Import Table Protection		2
PE_Tetris.exe	Compression		
arse successfully!	Memory Check		
	Resource Section Encryption		
	✓ Plug-in		
	 Anti Debug Option 	\bigcirc	
	Detecting Hardware Breakpoints		
	Detecting Memory Breakpoints		
	Memory Integrity Check		
	∨ ds	\bigcirc	Open DSProtector
	Password	•••••	

Figure 3-23

Following picture will show you the result, if you have enabled this function

Use GDB to debug the protected program used the anti-debug option:



Figure 3-24

Use IDA to debug the protected program used the anti-debug option:

📃 Output wir	ndow
F1FFAB1C: t	thread has started (tid=28720)
Debugger: t	thread 28720 has exited (code 0)
F1FFAB1C: t	thread has started (tid=28721)
D3515000: 1	loaded /data/app/com.KLS.LetteClear-Xgwvm2xcylu6QvChWmk7Hg==/lib/arm/libmain.so
D33C7000: 1	Loaded /system/framework/oat/arm/gson.odex
D381E3EA: g	got SIGSEGV signal (Segmentation violation) (exc.code b, tid 28663)
F204124C: g	ot SIGSEGV signal (Segmentation violation) (exc.code b, tid 28663)
D2F00000: 1	loaded /data/app/com.KLS.LetteClear-Xgwvm2xcylu6QvChWmk7Hg==/lib/arm/libmono.so
F1FFDFAA: g	got SIGABRT signal (Abort) (exc.code 6, tid 28663)
F1FFDFDC: g	got SIGABRT signal (Abort) (exc.code 6, tid 28663)





- 9. **Ds (Resource protection):** Encrypts the resource section of the protected program, DS Protector is a data resource protection tool that encrypts the data resource files of the program. When you are using this function, you need to switch on the button to "**green**".
- 10. **Password:** You can also set a password, letters and numbers are supported, but it should not be longer than 64 characters.
- 11. **DSProtector button:** You can open DSProtector by clicking this button.
- 12. **Import Table Protection:** Developer may select this option to protect "Import table" which imported to PE Program and encrypt this table, protect the function when the function is called by external program, API list has been hidden and encrypted to enhance the security level to the PE Program, recommend developer select this option. Protect the program by interference the reverse analysis.

The Protection option of "Protect Import Table" support PE program only.

Protection Mechanism:

Remove the original import table, and replace the IAT (Import Address Table) with the repair function, let the packer code take over the function call of the import function.

13. **Resource Section Encryption:** For PE program, Virbox Protector can encrypt the Resource Section in the program, to prevent the resources information from being extracted and tampered illegally.

Protection Mechanism:

When the program is enveloped, Virbox Protector will extract the resource section and encrypt, only the resources that be used externally will be protected (such as program icon, program version information). When the program executed, then these resources will be decrypted.

Note: Only local programs supported to encrypt the resource section.

14. Memory Check:

Memory check is the function implemented by Virbox Protector which is used to check the integrity of the program itself, and can be used to prevent illegal patch, memory patch and software breakpoint. What is more, memory check table and logic check is self protected to make sure the security of the software.

Memory check would be run in the program entry point, Virbox Protector loader will check every memory block to check the integrity. If verified failed, the program will exit.

If SDK label is used, every time you call VBProtectVerifyImage, the memory would be checked.



Please noted that the code encryption and Memory check can't be used at same time.

The ds option (Resource encryption) and memory check option can't be used at same time.

3.3.2.4 Resource Encryption

This option is only valid for the program of Unity3D.

You can refer the corresponding chapter for how to use this option.

Viribox Protector 2 (64-bit) v2.0.1.12323 File Protect Plug-in Log Setting Help File Save Selected Configuration Save All Confi	iguration Protect Selected Projects Protect All Project	- X Authorization Info Account: 1 Authorization Detail
File/Directory Protect successfully! angrybots5.5.3.apk Protect successfully! Protect successfully!	Basic Info Function Options Protection Options Resource Encryption Enable (It is recommended that only the resources selected by default be encrypted) Enable (It is recommended that only the resources selected by default be encrypted) File jamigou_Test_Data/globalgamemanagers jamigou_Test_Data/sharedassets0.assets jamigou_Test_Data/sharedassets0.assets.resS jamigou_Test_Data/sharedassets0.assets.resS.bak	Add File X X X X X X X X X X
E:/Test-Program/Unity3D/室内走廊游戏/室内走廊游戏/U	nity30	Unity3D

3.3.2.5 Status bar

Virbox Protector Standalone x64 (1.4.2.9340)		-		×
File Protect Plug-in Log Setting Help				_
Open File Save Selected Configuration Save All Con	figuration Protect Selected Projects Protect All Project			
File/Directory	Base Info Base Info Function Options C Protection Options			
.Net_RetroSnaker.exe				
Parse successfully!				
	File/Directory Path: E:/Test-Program/.Net_RetroSnaker.exe			
	Created Time: 2018/11/12 14:13:52			
	Last Modified Time: 2018/11/12 14:13:53			
	Last Accessed Time: 2018/11/22 14:59:15			
	Type: .NET AnyCPU i386			
	Type, inclinityceoliddo			
Friffred Decement (Net: DeterContex and		NET		or 1
E:/Test-Program/.Net_RetroSnaker.exe		INET A	AnyCPU i3	, 00

Figure 3-26

In the bottom of the window is the status bar, which will show the corresponding software full location,



software type, and hardware type of the selected software.

To Complete the Protection:

Click the button of "**Protect Selected Project**" to complete the protection process, then prompt with "**Protection Successful**" means the software protection completed. Open the directory where the protected software located, you will find the file: xxx.*ssp*.exe or xxx.ssp.dll will be listed in this directory. The executable file that has *ssp* in between filename and extension name is the software application has been protected by Virbox Protector. *Rename* this file name to be the original file name for further evaluation or distribute this protected software in future. Please keep the original software file in safety.



4 The Mechanism of software protection

4.1 Protect the PE program (application) and DLL.

Software Developer use Virbox Protector to protect the executable file and DLL library, with the functions protections Option, Protection Options, "Anti-Debug Option" plug-in feature and other Protection technology, as introduced in Chapter 2 and Chapter 3, Developer may flexible select these functions to protect the software functions, codes, critical algorithms and evaluate the software execution performance.

- <section-header><section-header><complex-block><complex-block><complex-block><complex-block><complex-block><complex-block>
- Following protection process will be implemented:

Figure 4-1

For example, you have created a .Net based C# language executable. You can use the above process to protect your software.



4.2 Protect the interpreter and code resource file (Python, PHP, etc.)

Software developer use the Virbox Protector and plug-in Unit (DS Protector) to protect the interpreter and related source code or resource files.

Software need to build an execution environment, for example, install a python environment on your PC and execute the **.py** or **.pcy** file (Or execute an **Mp3**, **Mp4** file with a media player).

- Following protection process will be implemented:
 - Use Virbox Protector to protect interpreter (Python.exe, player, etc.);
 - Use DS Protector to protect the source code or data resources (.py, media files, etc.);





4.3 Make the protection scheme for your software

When you open the Virbox Protector, you can directly drag the windows Application to the Virbox Protector to protect, as shown in the figure below:



	onfiguration Protect Selected Projects Protect Al Project		×
Fle/Directory	Welcome Page Welcome to Virbox Protector ! Start Gpen File Open Directory(Oxly support Unity20) Recent Files/Directory Cil/Vers/Nurongang/Desktop/Inery yobd5.5.3.apk Cil/Vers/Nurongang/Desktop/Iner Area Cil/Vers/Nurongang/Desktop/Java-test/demo-0.0.1.SNAPSHOT-orijar		



You can make your dedicated protection scheme and "configure" the protection options by select following Function Options and Protection Options as shown in below:

Virbox Protector Standalone x64 (1.4.2.9340)					- 🗆 ×
File Protect Plug-in Log Setting Help					
Open File Save Selected Configuration Save All Con		ct			
File/Directory	G Base Info Base Info Proteins	ection Options			
Parse successfully!	Total Qty: 79 Total Protected: 1				
	Obfuscation:0 Virtualization: 0 Encrypte	ed: 1			Add Function
	 ✓ (3) 会告号 ✓ (6) Program (8) void Main() 	Function Name: Address: Ptortection Type: Note: Assembly Code 40538E 5A289E	void Main() 40538D Obfuscation	call 0x0a0009b	
		405393 OA		ldc.i4.0	
		405394 162890	0000	call 0x0a00009c	
		405399 0A7344	40000	newobj 0x0600004a	
		40539E 06289D	00000	call 0x0a00009d	
		4053A3 OA		ret	
C://isers/luvononiano/Deskton/.Net_RetroSnaker.eve					NET AnyCPI IIi386

Figure 4-4



ation Save All Configuration Protect Selected Projects Protect All Projects	ect			
Sase Info Detion Options Options	tection Options			
Total Qty: 79 Total Protected. 1				Add Function
Obfuscation: 0 Virtualization: 0 Encrypt	ted:1			Add Function
 ✓ ③ 会会的 ✓ ④ Program (2) void Main() 	Function Name: Address: Ptortection Type: Note: Assembly Code 40538E 5A289E 405393 0A 405394 16289C 405399 0A7348	20000	call 0x0a00009b ldc.i4.0 call 0x0a00009c newobj 0x0600004a	
	40539E 06289D 4053A3 0A	0000	call 0x0a00009d	

Figure 4-5

Protect Plug-in Log Setting Help	ifguration Protect Selected Projects Protect A		Authorization Info Account: 15010303968 Logout Authorization Detail
File/Directory		Protection Options	
Protect successfully! angrybots5.5.3.apk Protect successfully! If it's an executable program,	 Output Info Output File/Directory Protection Options 	.Net_RetroSnaker.ssp.exe	•••
.Net_RetroSnaker.exe Parse successfully!	Compression		
PE_Tetris.exe	JIT Encrypt		
Parse successfully!	Name obfuscation	\checkmark	
	✓ Plug-in		
	Y Anti Debug Option	\bigcirc	
	Detecting Hardware Breakpoints		
	Detecting Memory Breakpoints		
	Memory Integrity Check		
	∽ ds	\bigcirc	Open DSProtector
	Password	••••••	



Select different technology to encrypt your function/method: Virtualization, Advanced obfuscation, Code Encryption. And also you can protect your program overall by selecting these options: Import table protection, Smart compression, Name obfuscation, Detecting Hardware Breakpoints, Detecting Memory Breakpoints, Memory Integrity Check; The resource of your program can also be protected by "Resource Section Encryption" option.

In this way to protect your software statically and dynamically.



5 Protection Example & Use Case

Before software protection:

With the Basic protection and Function level protection provided by Virbox Protector, it is not only to protect the software overall but also to protect the selected critical functions of the software and enhance software security. With the "**Performance Analysis**" feature, the protected software can get highly security and also no big impact to the software performance.

You can find all the basic protection options from the "**Protection Options**" Tag, and you can protect the functions of your software by selection from the "**Function Options**" Tag. For details information you can refer the below sample.

5.1 Protect the Local Executable

Local executable include: PE (Windows), ELF (Linux), Mach-O (macOS) format executable.

5.1.1 Fundamental protection to software

5.1.1.1 Import Table Protection

Hide the import table of the original program to protect the functions called by external program. With this way, to against the reverse engineering analysis and prevent the unpacking of the program.

The Program supported to be protected by using the "Import Table Protection":

Only PE format program/executable files supported.

Protection Mechanism:

Remove the import table of the original program, replace the Import Address Table (IAT) with stub function, and let the Virbox Protector loader take over the invoking of the import functions.

5.1.1.2 Resources encryption

Resources protection/encryption is for PE format program/encryption function, used to protect the resources and prevent the resources from being extracted illegally or tampered.

Protection Mechanism:



The resources in the PE program will be extracted and encrypted by Virbox Protector while it is protected. Those externally used resources will be decrypted in the Virbox Protector loader program when the program is executed (such as software icon and software version information).

5.1.1.3 Additional data extension

What is Additional extension data?

Additional data is the data (video or database) being combined with the original executable by the compiler or packer tool, these data will be read by the original executable when executed and will not be mapped into the memory directly.

Purpose

As the loader of software protector will modify the original program, if directly merge or join the additional extension data into the protected program, abnormal may happen when the program is executed. Hook is used in additional data extension to read the additional data normally, and encrypt the additional data to prevent the data being used illegally.

5.1.1.4 Compression

Protection Mechanism

When executed the compression of Virbox Protector, it packs the original code segment and data segment of software and compress the software, it will replace the Original Entry Point (OEP) with the Virbox Protector's code (loader). The data segment and code segment will be retrieved when the program is executed, and relocate to execute the program.

Purpose:

Prevent the static anti-compile and prevent the source program being patched

The Benefit:

- 1. Hide the code, data and file structure information of the program, protect the software in overall.
- 2. Highly efficiency when the program executed, and mini impact to performance when the program is loaded.

The Weakness:



After the Protector code (loader) executed, the code segment and data segment possible be retrieved and be dumped.

Comparison:

Before Protection:

			-										
Library function Regula	ar function	×	Data		IDA View-A	mal symbol		Hex V	iew-1	X	A		Structures
Function name		 ^	_	•		; [0000	00DC 8	BYTES: COLLA	PSED FU	NCTION _1	time.	PRESS	CTRL-NUMPAD+ TO
f sub_401000					.text:004056F4 .text:004056F4			5 U B	D O U T	TNE			
f sub_401030					.text:004056F4	;			RUUI	TINC			
f sub_401050					.text:004056F4	: Attri	butes	: librarv fu	unction	noreturn	bp-ba	ased fi	rame
f sub_401060					.text:004056F4	,		,			- P		
f sub_4010A0					.text:004056F4			public s	tart				
f sub_4010B0					.text:004056F4	start		proc nea	in				
f sub_401140					.text:004056F4								
f nullsub_10					.text:004056F4			= dword					
f sub_401170					.text:004056F4 .text:004056F4		ne	= dword = dword					
f sub_401180					.text:004056F4		Info			ptr -5Cl			
f sub_401190					.text:004056F4		11110			ptr -18h	·		
f sub_401270					.text:004056F4								
f sub_401290					.text:004056F4	;unw	ind {	//except	_handle	r3			
f sub_401320					.text:004056F4			push	ebp				
f sub_401390				- 11	.text:004056F5			mov	ebp, es				
f sub 4013A0					.text:004056F7			push	ØFFFFFF				
f sub_4013F0					.text:004056F9 .text:004056FE			push push		stru_416/ except			
f sub_401410					.text:00405703			mov		rge fs:0		Lei D	
f sub 401460				•	.text:00405709			push	eax, 10	- BC 1310			
f sub_401480				•	.text:0040570A			mov		s:0, esp			
f sub_4014D0				•	.text:00405711			sub	esp, 58	h			
f sub_401590					.text:00405714			push	ebx				
f sub_401610				- 11	.text:00405715			push	esi				
f sub_401630					.text:00405716			push	edi				
f sub_401690					.text:00405717 .text:0040571A			mov call	ds:GetV	_exc.old	_esp],	, esp	
f sub_4016C0					.text:00405720			xor	edx, ed				
f sub_4016D0				•	.text:00405722			mov	dl, ah				
f sub_401770				•	.text:00405724			mov		1E524, ed	dx		
f sub_401790					.text:0040572A			mov	ecx, ea	x			
f sub_4017C0				- 1	.text:0040572C			and	ecx, ØF				
f sub_4017E0				- 11	.text:00405732			mov		1E520, ed	cx		
f unknown libname 1					.text:00405738			shl add	ecx, 8				
f sub_401840				•	.text:0040573B .text:0040573D			mov	ecx, ed	x 1E51C, ed	~~		
f sub_401870					.text:00405743			shr	eax, 10		~		
f sub_401A50				•	.text:00405746			mov			ax		
f sub_401A70				•	.text:0040574B			push	1 -				
f sub_401A80					.text:0040574D			call	heap_	init			
f sub_401860					.text:00405752			рор	ecx				

Figure 5-1

After Protection:



nction name Segment ^ sub_489EA0		🏝 🕹 🙇 🚺 🖬	📾 📾 🚯 📌 🛪 🧩 🗃 🗙 🕨 💷 💷 No debugger 🔹
Functions window Image: Segment of the second s			
sub_489EA0 .virbox3 sub_489EB2 .virbox3 sub_489EC6 .virbox3 sub_489F12 .virbox3 sub_489F40 .virbox3 sub_489F70 .virbox3 sub_489F73 .virbox3 sub_489F83 .virbox3 sub_489F83 .virbox3 sub_489F83 .virbox3 sub_480045 .virbox3 sub_48A005 .virbox3 sub_48A096 .virbox3 sub_48A096 .virbox3 sub_48A100 .virbox3 sub_48A100 .virbox3 sub_48A499 .virbox3 sub_48A499 .virbox3 sub_48A499 .virbox3 sub_48A51B .virbox3 sub_48A58 .virbox3 sub_48A68 .virbox3 sub_48A684 .virbox3 sub_48A684	Library function 📃 Regular function	Instruction 📃 Data 📕 Unex	xplored 📒 External symbol
sub_dB9EA0.vibox3sub_dB9EB2.vibox3sub_dB9EB2.vibox3sub_dB9EEA.vibox3sub_dB9F40.vibox3sub_dB9F67.vibox3sub_dB9F67.vibox3sub_dB9F67.vibox3sub_dB9F63.vibox3sub_dB9F83.vibox3sub_dB06.vibox3sub_dBA004.vibox3sub_dBA004.vibox3sub_dBA1D0.vibox3sub_dBA499.vibox3sub_dBA499.vibox3sub_dBA51B.vibox3sub_dBA588.vibox3sub_dBA588.vibox3sub_dBA588.vibox3sub_dBA588.vibox3sub_dBA588.vibox3sub_dBA588.vibox3sub_dBA684.vibox3sub_dBA6	f Functions window		IDA View-A 🛛 🖸 Hex View-1
sub_489EB2.virbox3sub_489EC6.virbox3sub_489E64.virbox3sub_489F12.virbox3sub_489F67.virbox3sub_489F67.virbox3sub_489F67.virbox3sub_489F63.virbox3sub_489F63.virbox3sub_489F63.virbox3sub_48065.virbox3sub_48A004.virbox3sub_48A004.virbox3sub_48A096.virbox3sub_48A100.virbox3sub_48A499.virbox3sub_48A482.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A684.virbox3sub_48A684.virbox3sub_48A684.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685.virbox3sub_48A685 <td< td=""><td>Function name</td><td>Segment ^</td><td></td></td<>	Function name	Segment ^	
sub_489EC6.virbox3sub_489EA.virbox3sub_489F12.virbox3sub_489F40.virbox3sub_489F67.virbox3sub_489F63.virbox3sub_489F63.virbox3sub_48066.virbox3sub_48A096.virbox3sub_48A096.virbox3sub_48A096.virbox3sub_48A096.virbox3sub_48A096.virbox3sub_48A100.virbox3sub_48A100.virbox3sub_48A49E.virbox3sub_48A49E.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A684.virbox3sub_48A685.virbox3sub_48A685.virbox3	f sub_4B9EA0	.virbox3	
sub_489EEA.virbox3sub_489F12.virbox3sub_489F40.virbox3sub_489F67.virbox3sub_489F79.virbox3sub_489F83.virbox3sub_48966.virbox3sub_48A045.virbox3sub_48A046.virbox3sub_48A045.virbox3sub_48A046.virbox3sub_48A047.virbox3sub_48A048.virbox3sub_48A049.virbox3sub_48A100.virbox3sub_48A492.virbox3sub_48A492.virbox3sub_48A482.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3 </td <td>f sub_4B9EB2</td> <td>.virbox3</td> <td></td>	f sub_4B9EB2	.virbox3	
sub_489F12.virbox3sub_489F40.virbox3sub_489F67.virbox3sub_489F79.virbox3sub_489F83.virbox3sub_48045.virbox3sub_48A045.virbox3sub_48A096.virbox3sub_48A096.virbox3sub_48A096.virbox3sub_48A094.virbox3sub_48A095.virbox3sub_48A094.virbox3sub_48A100.virbox3sub_48A492.virbox3sub_48A492.virbox3sub_48A482.virbox3sub_48A51B.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A72F.virbox3	f sub_4B9EC6	.virbox3	
sub_489F12.virbox3sub_489F40.virbox3sub_489F67.virbox3sub_489F79.virbox3sub_489F83.virbox3sub_48045.virbox3sub_48A045.virbox3sub_48A096.virbox3sub_48A096.virbox3sub_48A096.virbox3sub_48A094.virbox3sub_48A095.virbox3sub_48A094.virbox3sub_48A100.virbox3sub_48A492.virbox3sub_48A492.virbox3sub_48A482.virbox3sub_48A51B.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A72F.virbox3	f sub_4B9EEA	.virbox3	
sub_489F67.virbox3public startsub_489F79.virbox3start proc nearsub_489F79.virbox3var_4= dword ptr -4sub_48A045.virbox3pushasub_48A004.virbox3call sub_4C1BDDsub_48A004.virbox3popasub_48A1D0.virbox3push eaxsub_48A499.virbox3push eaxsub_48A482.virbox3mov eax, offset dword_4C85A0sub_48A51B.virbox3mov eax, offset dword_4C85A0sub_48A568.virbox3mov eax, offset dword_4C85A0sub_48A58.virbox3mov eax, offset dword_4C85A0sub_48A68.virbox3mov [esp+0Ch+var_4], eaxsub_48A68.virbox3pop ebxsub_48A68.virbox3pop eaxsub_48A68.virbox3pop eaxsub_48A68.virbox3pop eaxsub_48A68.virbox3pop eaxsub_48A68.virbox3pop ebxsub_48A68.virbox3pop eaxsub_48A68.virbox3pop ebxsub_48A68.virbox3pop eaxsub_48A68.virbox3pop eaxsub_48A68.virbox3pop eaxsub_48A68.virbox3pop eaxsub_48A68.virbox3.virbox3sub_48A68.virbox3.virbox3sub_48A68.virbox3.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A72F.virbox3		.virbox3	
sub_489F67.virbox3start proc nearsub_489F83.virbox3var_4= dword ptr -4sub_48045.virbox3pushasub_48A045.virbox3call sub_4C1BDDsub_48A004.virbox3popasub_48A492.virbox3push eaxsub_48A492.virbox3mov eax, offset dword_4C85A0sub_48A58.virbox3mov eax, offset dword_4C85A0sub_48A58.virbox3mov eax, offset dword_4C85A0sub_48A58.virbox3mov eax, offset dword_4C85A0sub_48A58.virbox3mov eax, offset dword_4C85A0sub_48A68.virbox3mov eax, offset dword_4C85A0sub_48A68.virbox3mov eax, offset dword_4C85A0sub_48A68.virbox3mov eax, offset dword_4C85A0sub_48A68.virbox3mov eaxsub_48A68.virbox3pop ebxsub_48A68.virbox3pop eaxsub_48A608.virbox3pop eaxsub_48A608.virbox3pop eaxsub_48A608.virbox3pop ebxsub_48A67.virbox3pop ebxsub_48A67.virbox3pop eaxsub_48A608.virbox3pop eaxsub_48A72F.virbox3.virbox3		.virbox3	
sub_489F79.virbox3sub_489F83.virbox3sub_48045.virbox3sub_48A096.virbox3sub_48A096.virbox3sub_48A094.virbox3sub_48A100.virbox3sub_48A499.virbox3sub_48A49E.virbox3sub_48A51B.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A67.virbox3sub_48A67.virbox3sub_48A67.virbox3sub_48A67.virbox3sub_48A67.virbox3sub_48A72F.virbox3	f sub_489F67	.virbox3	
sub_48A045.virbox3sub_48A096.virbox3sub_48A004.virbox3sub_48A004.virbox3sub_48A100.virbox3sub_48A499.virbox3sub_48A492.virbox3sub_48A482.virbox3sub_48A482.virbox3sub_48A51B.virbox3sub_48A568.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A58.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A68.virbox3sub_48A684.virbox3sub_48A685 <td< td=""><td>f sub_489F79</td><td>.virbox3</td><td>start proc near</td></td<>	f sub_489F79	.virbox3	start proc near
sub_48A045.virbox3usub_48A096.virbox3pushasub_48A004.virbox3call sub_4C1BDDsub_48A100.virbox3popasub_48A499.virbox3push eaxsub_48A49E.virbox3push ebxsub_48A482.virbox3mov eax, offset dword_4C85A0sub_48A51B.virbox3mov ebx, ds:dword_4C85A0sub_48A568.virbox3mov [esp+0Ch+var_4], eaxsub_48A58.virbox3pop ebxsub_48A684.virbox3pop eaxsub_48A684.virbox3pop ebxsub_48A608.virbox3pop eaxsub_48A675.virbox3pop ebxsub_48A72F.virbox3pop eax	f sub_489FB3	.virbox3	var 4= dword ptr -4
sub_48A0D4.virbox3callsub_4C1BDDsub_48A1D0.virbox3popasub_48A499.virbox3subesp, 4sub_48A49E.virbox3pusheaxsub_48A482.virbox3moveax, offset dword_4C85A0sub_48A51B.virbox3moveax, sebxsub_48A56B.virbox3movesx, ebxsub_48A68.virbox3mov[esp+0Ch+var_4], eaxsub_48A68.virbox3popebxsub_48A68.virbox3popeaxsub_48A608.virbox3popeaxsub_48A608.virbox3popeaxsub_48A72F.virbox3popeax		.virbox3	
Jub_lab.chNHOUSsub_dBA1D0.virbox3sub_dBA499.virbox3sub_dBA499.virbox3sub_dBA492.virbox3sub_dBA51B.virbox3sub_dBA56B.virbox3sub_dBA58.virbox3sub_dBA684.virbox3sub_dBA684.virbox3sub_dBA684.virbox3sub_dBA684.virbox3sub_dBA684.virbox3sub_dBA684.virbox3sub_dBA684.virbox3sub_dBA684.virbox3sub_dBA684.virbox3sub_dBA675.virbox3sub_dBA675.virbox3sub_dBA675.virbox3sub_dBA675.virbox3sub_dBA725.virbox3	f sub_4BA096	.virbox3	
sub_dBA499.virbox3sub_esp, 4sub_dBA492.virbox3push_eaxsub_dBA49E.virbox3mov_eax, offset dword_4C85A0sub_dBA50E.virbox3mov_ebx, ds:dword_4C85A0sub_dBA56B.virbox3mov_ebx, ds:dword_4C85A0sub_dBA68F.virbox3mov_ebx, ds:dword_4C85A0sub_dBA68F.virbox3mov_ebx, ds:dword_4C85A0sub_dBA60B.virbox3pop_ebxsub_dBA60B.virbox3pop_eaxsub_dBA60B.virbox3pop_eaxsub_dBA72F.virbox3.virbox3	f sub_4BA0D4	.virbox3	
sub_48A499.virbox3pusheaxsub_48A492.virbox3moveax, offset dword_4C85A0sub_48A51B.virbox3moveax, offset dword_4C85A0sub_48A568.virbox3addeax, ebxsub_48A58.virbox3moveex, offset dword_4C85A0sub_48A58.virbox3moveex, ebxsub_48A68.virbox3popebxsub_48A68.virbox3popebxsub_48A684.virbox3popebxsub_48A608.virbox3popeaxsub_48A72F.virbox3.virbox3eax	f sub_4BA1D0	.virbox3	
sub_4BA49E.virbox3pushebxsub_4BA49E.virbox3moveax, offset dword_4C85A0sub_4BA51B.virbox3movebx, ds:dword_4C85A0sub_4BA56B.virbox3addeax, ebxsub_4BA63B.virbox3mov[esp+0Ch+var_4], eaxsub_4BA6AF.virbox3popebxsub_4BA6AF.virbox3popeaxsub_4BA6AF.virbox3popeaxsub_4BA6B4.virbox3popeaxsub_4BA6B5.virbox3popeaxsub_4BA72F.virbox3sub_4BA72Fsub_4BA72F	f sub_4BA499	.virbox3	
sub_4BA4B2.virbox3moveax, offset dword_4C85A0sub_4BA51B.virbox3movebx, ds:dword_4C85A0sub_4BA56B.virbox3addeax, ebxsub_4BA63B.virbox3mov[esp+0Ch+var_4], eaxsub_4BA63F.virbox3popebxsub_4BA684.virbox3popeaxsub_4BA6B4.virbox3popeaxsub_4BA6B4.virbox3sub_4BA6B4sub_4BA72Fsub_4BA72F.virbox3.virbox3sub_4BA72F		.virbox3	
sub_4BA51B.virbox3movebx, ds:dword_4C85A0sub_4BA56B.virbox3addeax, ebxsub_4BA5A9.virbox3mov[esp+0Ch+var_4], eaxsub_4BA63B.virbox3popebxsub_4BA6B4.virbox3popeaxsub_4BA6B4.virbox3retnsub_4BA6B4.virbox3sub_4BA6DBsub_4BA72F.virbox3sub_4BA72F		.virbox3	
sub_4BA5A9.virbox3mov[esp+0Ch+var_4], eaxsub_4BA638.virbox3popebxsub_4BA6AF.virbox3popeaxsub_4BA6B4.virbox3retnsub_4BA6DB.virbox3name endp : sp-analysis failedsub_4BA72F.virbox3	f sub_4BA51B	.virbox3	
sub_4BA63B .virbox3 pop ebx sub_4BA6AF .virbox3 pop eax sub_4BA6B4 .virbox3 retn sub_4BA6DB .virbox3 BMante endp; sp-analysis failed sub_4BA72F .virbox3	f sub_4BA56B	.virbox3	
sub_4BA6B4 .virbox3 retn sub_4BA6B4 .virbox3 retn sub_4BA6B4 .virbox3 sub_4BA72F		.virbox3	
sub_4BA6DB virbox3 sub_4BA72F virbox3		.virbox3	F-F
sub_4BA6B4 .virbox3 sub_4BA6DB .virbox3 sub_4BA72F .virbox3		.virbox3	
sub_4BA6DB .virbox3 sub_4BA72F .virbox3	f sub_4BA6B4	.virbox3	
	f sub_4BA6DB	.virbox3	
auth 4RCE01 uitheu2		.virbox3	
	f sub_4BCE01	.virbox3	



5.1.1.5 Memory Check:

- Memory check is the function implemented by Virbox Protector which is used to check the integrity of the program itself, and can be used to prevent illegal patch, memory patch and software breakpoint. What is more, memory check table and logic check is self protected to make sure the security of the software.
- 2. Memory check would be run in the program entry point, Virbox Protector loader will check every memory block to check the integrity. If verified failed, the program will exit.
- 3. If SDK label is used, every time you call *VBProtectVerifyImage*, the memory would be checked.

Instructions for Use:

Drag the PE or ELF program to Virbox Protector, the "**Memory check**" would be shown in the "Protection Option", you need to select this option to protect the program with memory check.



File/Directory	Seasic Info Function Options	Protection Options	
Parse successfully!	✓ Output Info Output File/Directory	PE_Tetris.ssp.exe	•••
	 Protection Options 		
	Import Table Protection		0
	Compression	\checkmark	
	Memory Check		
	Resource Section Encryption		
	✓ Plug-in		
	 Anti Debug Option 	\bigcirc	
	Detecting Hardware Breakpoints		
	Detecting Memory Breakpoints		
	Memory Integrity Check		
	∽ ds	\bigcirc	Open DSProtector
	Password	•••••	

5.1.2 Protect the critical Functions of software with following technology

5.1.2.1 Code obfuscation

Protection Mechanism:

Select "**Obfuscation**": **Virbox Protector** will translate the code instruction into a stream of pseudo-code that neither the machine nor the human can identify these pseudo code. When the pseudo-code executed, the software will translate and interpret to restore the code into the original code to execute. Virbox Protector support the obfuscation to **x86/arm /.net** il series instruction.

The Purpose

Obfuscate the source instruction and prevent the program from being static analyzed.

The Benefit:

Prevent from anti-compiled and make it more difficult to analysis the code.

The Weakness:

Performance impact.

Protection Comparison:

Before protection(X86):



.text:00414485	mov	eax, offset loc 414CD8
.text:004144BA	call	EH prolog
.text:004144BF	push	ecx
.text:004144C0	mov	[ebp+var 10], ecx
.text:004144C3	mov	dword ptr [ecx], offset off 416508
.text:004144C9 ; try	-16	
.text:004144C9	and	[ebp+var 4], 0
.text:004144CD	add	ecx, 4
.text:004144D0	push	ecx ; void **
.text:004144D1	call	<pre>?AfxDeleteObject@@YGXPAPAX@Z ; AfxDeleteObject(void * *)</pre>
.text:004144D6	mov	ecx, [ebp+var C]
.text:004144D9	mov	large fs:0, ecx
.text:004144E0	leave	
.text:004144E1	retn	
.text:004144E1 ; } //	starts at 414	4C9
.text:004144E1 ; } // st	tarts at 4144B	5
.text:004144E1 sub 4144E	35 endp	
.text:004144E1		

Figure 5-3

After protection(X86):

.text:004144B5		
.text:004144B5 loc_4144B5:		; CODE XREF: sub_412E0B+31p
.text:004144B5	call	sub 466CF0
.text:004144BA	out	6Ch, eax
.text:004144BC	sub	[esi+50h], ebp
.text:004144BF	call	sub 466D94
.text:004144C4	push	55
.text:004144C5	adc	dword ptr [edi-37245E23h], 64h
.text:004144C5 ;		
.text:004144CC	db ØF7	h
.text:004144CD ;		
.text:004144CD		
.text:004144CD loc 4144CD:		; CODE XREF: .text:0041453D↓j
.text:004144CD	dec	ebx
.text:004144CE	or	ebp, edi
.text:004144D0	adc	bl, bh
.text:004144D2	movsb	
.text:004144D3	cmc	
.text:004144D4	mov	[ebx-18h], edx
.text:004144D7	test	[ecx], ebp
.text:004144D9	add	eax, 1102B200h
.text:004144DE	jle	short loc 41447C
.text:004144E0	mov	cl, 10h
.text:004144E2		

Figure 5-4

ARM architecture source code:

Before protection



.text:00076E9C		
text:00076E9C ; android:		d_database_CursorWindow(_JNIEnv *)
text:00076E9C	EXPORT _ZN	7android38register_android_database_CursorWindowEP7_JNIEnv
text:00076E9C _ZN7androi	d38register_andr	oid_database_CursorWindowEP7_JNIEnv
text:00076E9C		; CODE XREF: android::register_android_databas
text:00076E9C		; DATA XREF: LOAD:0001B73010
text:00076E9C ;unwind	{	
text:00076E9C	PUSH	{R3-R7,LR}
text:00076E9E	MOV	R4, R0
text:00076EA0	LDR	R6, =(aAndroidDatabas - 0x76EA8)
text:00076EA2	LDR	R3, [R0]
text:00076EA4	ADD	R6, PC ; "android/database/CharArrayBuffer"
text:00076EA6	LDR	R2, [R3,#0×18]
text:00076EA8	MOV	R1, R6
text:00076EAA	BLX	R2
text:00076EAC	MOV	R5, R0
text:00076EAE	CBNZ	R0, loc 76EBE
text:00076EB0	LDR	R0, =(aClazzNull - 0x76EBA)
text:00076EB2	LDR	R1, =(aCursorwindow - 0x76EBC)
text:00076EB4	LDR	R2, =(aUnableToFindCl - 0x76EBE)
text:00076EB6	ADD	R0, PC ; "clazz == NULL"
text:00076EB8	ADD	R1, PC ; "CursorWindow"
text:00076EBA	ADD	R2, PC ; "Unable to find class %s"
text:00076EBC	B	loc 76EE2
text:00076EBE :		IOC_/OLL2
text:00076EBE		
text:00076EBE loc 76EBE		; CODE XREF: android::register android databas
text:00076EBE	LDR	R6, =(aData - 0x76EC8)
text:00076EC0	MOV	R1, R5
text:00076EC2	LDR	R0, [R4]
text:00076EC4	ADD	R6, PC ; "data"
text:00076EC6	LDR	R3, =(aC - 0x76ED4)
text:00076EC8	LDR.W	
		R7, [R0,#0x178]
text:00076ECC	MOV	R2, R6
text:00076ECE	MOV	R0, R4
text:00076ED0	ADD	R3, PC ; "[C"
text:00076ED2	BLX	R7
text:00076ED4	CBNZ	R0, loc_76EE8
text:00076ED6	LDR	R0, =(aResNull - 0x76EE0)
text:00076ED8	LDR	R1, =(aCursorwindow - 0x76EE2)
text:00076EDA	LDR	R2, =(aUnableToFindSt - 0x76EE4)
text:00076EDC	ADD	R0, PC ; "res == NULL"
text:00076EDE	ADD	R1, PC ; "CursorWindow"
text:00076EE0	ADD	R2, PC ; "Unable to find static field %s"
text:00076EE2		
text:00076EE2 loc_76EE2		; CODE XREF: android::register_android_databas
text:00076EE2		; android::register_android_database_CursorWir
text:00076EE2	MOV	R3, R6
text:00076EE4	BLX	android log assert

Figure 5-5

After Protection:



Figure 5-6



5.1.2.1.1 Anti-Run Trace:

Please noted that the anti-Run Trace is for the ARM architecture based program.

- Run Trace is the function provided by debugger, every time you run a command in single step, the register status of every command would be recorded, this is a common way to debug trace and anti-obfuscation to the program to be reversed.
- 2. After the code obfuscation function of Virbox Protector is used for the ARM architecture program, the instruction of the function will set some "hidden pitfall" to test the single step breakpoint, if illegal debug (wrong instruction) command is detected, it will execute some wrong instruction to interrupt the illegal debug and make the program crash and make it impossible to debug the program. In this way to enhance the security of the program you protected with code obfuscation.

Thread	Address	Instruct	ion	Result
💭 0000238D	.text:main	Memory	layout changed: 105 segments	Memory layout changed: 105 se
🗉 0000238D				X0=00000000000001X1=000
🗏 0000238D	.text:main	STP	X29, X30, [SP,#var_20]!	SP=0000007FF1123CE0
🗏 0000238D	.text:main+4	MOV	X29, SP	X29=0000007FF1123CE0
🗏 0000238D	.text:main+8	STR	W0, [X29,#0x20+argc]	
🗏 0000238D	.text:main+C	STR	X1, [X29,#0x20+argv]	
🗏 0000238D	.text:main+10	ADRP	X0,	X0=000005555603000
🗏 0000238D	.text:main+14	ADD	X0, X0,	. X0=000005555603110
🗏 0000238D	.text:main+18	BL	.puts	X30=000005555595BE4
🗏 0000238D	.plt:.puts	ADRP	X16, #puts_ptr@PAGE	X16=0000005555647000
🗏 0000238D	.plt:.puts+4	LDR	X17, [X16, #puts_ptr@PAGEOFF]	X17=00000070FE685A48
🗏 0000238D	.plt:.puts+8	ADD	X16, X16, #puts_ptr@PAGEOFF	X16=0000055556476E8
🗏 0000238D	.plt:.puts+C	BR	X17; puts	X0=0000000000000A X1=00
🗏 0000238D	.text:main+1C	ADD	X0, X29, #0x1C; argc	X0=000007FF1123CFC
🗏 0000238D	.text:main+20	LDR	X1, [X29,#0x20+argv]; argv	X1=0000007FF1123D68
0000238D	.text:main+24	BL		X30=0000055555958F0

5.1.2.2 Code Virtualization

Protection Mechanism:

Virbox Protector will compiles instructions into virtual instructions executed in the specified virtual machine.

The Purpose:

Hide the original instruction, prevent the code logic from being analyzed.

The Benefit:

Highly secured, the original code logic almost can't be identified and analyzed.

The Weakness:

Performance impact.

Please noted that X86, X64 and ARM architecture program are supported to use this function.



5.1.2.3 Code encryption (Native)

Protection Mechanism:

Encrypted the original function of the program by SMC (Self-Modifying Code) technology and only when the program is executed then the function would be decrypted.

The Purpose:

Prevent the program from **being unpacking**, and prevent the program from being dumped.

The Benefit:

No impact to software performance.

The Weakness:

It is possible to decrypt and analyze the functions.

Protection Comparison:

Before protection:

.text:004056F4 start	proc n	ear
.text:004056F4	proc n	
.text:004056F4 var 68	= dwor	d ptr -68h
.text:004056F4 lpCmdLine		d ptr -64h
.text:004056F4 var 60		d ptr -60h
.text:004056F4 StartupInfo		
.text:004056F4 ms exc	= CPPF	H RECORD ptr -18h
.text:004056F4		
.text:004056F4 ; unwind {	// exce	nt handler3
.text:004056F4	push	ebp
.text:004056F5	mov	ebp, esp
.text:004056F7	push	ØFFFFFFFh
.text:004056F9		offset stru 416AB0
.text:004056FE		offset except handler3
.text:00405703	mov	eax, large fs:0
.text:00405709	push	eax
.text:0040570A	mov	large fs:0, esp
.text:00405711	sub	esp, 58h
.text:00405714	push	ebx
.text:00405715	push	esi
.text:00405716	push	edi
.text:00405717	mov	[ebp+ms exc.old esp], esp
.text:0040571A	call	ds:GetVersion
.text:00405720	xor	edx, edx
.text:00405722	mov	dl, ah
.text:00405724	mov	dword_41E524, edx
.text:0040572A	mov	ecx, eax
.text:0040572C	and	ecx, 0FFh
.text:00405732	mov	dword_41E520, ecx
.text:00405738	shl	ecx, 8
.text:0040573B	add	ecx, edx
.text:0040573D	mov	dword_41E51C, ecx
.text:00405743	shr	eax, 10h
.text:00405746	mov	dword_41E518, eax
.text:0040574B	push	1
.text:0040574D	call	heap_init
.text:00405752	рор	ecx
.text:00405753	test	eax, eax
.text:00405755	jnz	short loc_40575F
.text:00405757	push	1Ch ; NumberOfBytesWritten
.text:00405759	call	_fast_error_exit

Figure 5-7



After Protection:

++-00405654 ··	
text:004056F4 ; text:004056F4	nuch 0
	push 0
text:004056F9 text:004056F9 :	jmp loc_420000
· · · · · ·	the set with any other and the sector and the set
text:004056FE	db 40h, 42h, 65h, 45h, 39h, 2 dup(0C6h), 14h, 92h, 9, 4Dh
text:004056FE	db 0AFh, 0AEh, 1Ch, 0B8h, 56h, 8, 4Ch, 38h, 7Dh, 0Dh, 0EAh
text:004056FE	db 0F3h, 8Ch, 19h, 0ACh, 12h, 0C8h, 62h, 4, 36h, 0Ah, 35h
text:004056FE	db 7Ah, 7Fh, 0F8h, 79h, 0E7h, 6, 62h, 4Eh, 51h, 0Ah, 0AAh
text:004056FE	db 0A4h, 0DAh, 1, 0EEh, 8Ah, 0A3h, 2Eh, 3, 0C1h, 69h, 0D0h
text:004056FE	db 0B2h, 61h, 6Dh, 75h, 4Dh, 81h, 92h, 87h, 58h, 0BCh
text:004056FE	db 69h, 33h, 0E7h, 8Bh, 0C9h, 69h, 0C5h, 0BAh, 0CDh, 0C1h
text:004056FE	db 0A7h, 45h, 66h, 8Fh, 0FFh, 71h, 0CFh, 9Dh, 40h, 0B0h
text:004056FE	db 90h, 37h, 16h, 0EEh, 7Dh, 42h, 70h, 2 dup(35h), 1Eh
text:004056FE	db 64h, 2Eh, 83h, 26h, 0D8h, 75h, 0DDh, 3Fh, 0A1h, 0E2h
text:004056FE	db 8Dh, 0A4h, 0F2h, 3, 0A4h, 17h, 5Ch, 49h, 0F8h, 27h
text:004056FE	db 76h, 90h, 0CBh, 7, 0BBh, 7, 0AEh, 89h, 61h, 98h, 9Bh
text:004056FE	db 86h, 7Fh, 70h, 0F8h, 9Eh, 5Eh, 0FAh, 0D0h, 57h, 0C7h
text:004056FE	db 0FCh, 38h, 0AEh, 64h, 0Ch, 85h, 67h, 3Ah, 0B2h, 9Bh
text:004056FE	db 7, 0F9h, 0EAh, 0ACh, 0C9h, 0BAh, 8Bh, 67h, 65h, 0D9h
text:004056FE	db 0D2h, 96h, 0CDh, 3Dh, 0D9h, 0B7h, 0FEh, 83h, 0F9h, 3Eh
text:004056FE	db 0B4h, 0E9h, 82h, 9Bh, 82h, 66h, 0CEh, 49h, 2Dh, 7Fh
text:004056FE	db 11h, 8Eh, 0F3h, 0DEh, 0FEh, 72h, 4Bh, 37h, 32h, 38h
text:004056FE	db 0ADh, 0B2h, 0EEh, 3Dh, 17h, 62h, 53h, 63h, 3Dh, 28h
text:004056FE	db 89h, 0FCh, 0Dh, 34h, 60h, 0ACh, 45h, 68h, 28h, 92h
text:004056FE	db 51h, 36h, 0D6h, 86h, 5Dh, 7Ch, 6Ch, 46h, 31h, 43h, 35h
text:004056FE	db 5Ch, 0EFh, 0F6h, 0C3h, 76h, 2Eh, 3Ch, 75h, 32h, 63h
text:004056FE	db 5Fh, 0A6h, 7, 99h, 57h, 15h, 0E0h, 0FDh, 0F5h, 4Dh
text:004056FE	db 7Bh, 0E3h, 65h, 7Fh, 1Dh, 0B7h, 8Bh, 65h, 0E8h, 0FFh
text:004056FE	db 75h, 98h, 0E8h, 3Fh, 10h, 2 dup(0)
text:004057FC	

Figure 5-8

5.1.3 Automatically protection to local executable files by using "Command line"

Virbox Protector provide 2 ways to developer to protect their local executable application:

Using GUI to "select and click" way, which is most easier way to developer to protect their application;

For Some developer who has rich experience in protection, they may more prefer to use command line to protect the critical functions in their application.

5.1.3.1 Generating & Using Map file

For the compiled application, The function will be shown and listed with the "address" when Virbox Protector parse the functions of PE program which is not easy to developer to identify and select these functions with "address", and The function will be shown and listed with function name if map file available, so using the "map" file will more convenient to developer to select the functions to be protected when use the Virbox Protector to protect their application, here we brief how to generate the "map" file with different language. Copyright © 2020, Virbox, All Right Reserved. Page 58 | 108



5.1.3.1.1 Generate Map file for BCB Program

BCB: Borland C++ Builder, here briefing how to use C++ Builder to generate the map file.

Project settings as shown below:

<u>File Edit Search View</u>	Project Run Component Databas	ie Iools Window Help (None) 💌 🔁 🖏
D D D D D D D D D D D D D Object TreeView () () () () ()	Add to Project Shift+F11 Semove from Project Import Type Library Add to Repository View Source Languages	no2 [sween Dota-Access: Dota Control: dot Control: dot Control: dot Control: dot Control: dot Control: dot Control: Web3
‱ & + +	Edit Option Source Export Makefile Add New Project Add Existing Project	
	Compile Unit Alt+F9 A Make Project2 Ctrl+F9 Build Project2 2 Information for [none] 2	
	Make All Projects	
	输 Web Deployment Options る Web Deploy	
Object Inspector	Options Shift+Ctrl+F11	12

Figure 5-9

Project Options for Project1.exe				
	Tasm CORBA CodeGuard iler Advanced Compiler C++ sed Linker Directories/Conditionals			
Linking Create debug information Use dynamic RTL	Warnings C All © Selected Warnings			
Use <u>d</u> ebug libraries Generate import library Generate lib file	PE file options Min stack size: 0x00002000			
✓ Don't generate state files Max errors: 0	Ma <u>x</u> stack size: 0x00100000 Min heap size: 0x00001000			
Map file	Mgx heap size: 0x00100000 Image base: 0x00400000			
C Segments 2 C Eublics C Detailed ↓ ♥ Show mangled names	Subsystem major: 4			
Default	OK Cancel <u>H</u> elp			

Figure 5-10

5.1.3.1.2 Generate map file for VC program

Project settings as shown below:



🧭 test - Microsoft Visual C+	+ - [a.cpp]	
Pile Edit View Insert	<u>Project</u> Build <u>T</u> ools	<u>W</u> indow <u>H</u> elp
] 🏠 🚅 🖬 🕼 ½ 🔩	Set Acti <u>v</u> e Project <u>A</u> dd To Project	> >
	D <u>e</u> pendencies	
2	<u>S</u> ettings	Alt+F7
₩ Workspace 'tes +1	Export <u>M</u> akefile	
🖻 🔄 Source Files	Insert Project into	Workspace

Figure 5-11

Figure 5-12

5.1.3.1.3 Generate map file for VS program

Project settings as shown below:

	test1 - I	Micro	soft	Visua	al Studio (Administrator)	
FILE	E EDIT	VIE	w	PRO	DJECT BUILD DEBUG TEAM TOOLS TEST ARCHITECTURE ANALYZE WINDOW HELP	
	0 - 0	17 -	2	*	Add Clas	
:8	stdafx.h		te	в÷	Class Wizard Ctrl+Shift+X	
Solution	test1			°	Add New Item Ctrl+Shift+A (Global Scope)	
		1 6	#:	* 0	Add Existing Item Shift+Alt+A	
Explorer		2	#:		Exclude From Project	
		3	#:	Ð	Show All Files	
Class View		4	v		Rescan Solution	
s ≤		5	{	ø	Set as StartUp Project	
		6		苗	Manage NuGet Packages	
Resou		7	}	۶	test1 Properties 2	
Ĕ		_				

Figure 5-13



t	est1 Property Pages			? ×
	Configuration: Active(Debug)	▼ Platform: Active(Wi	n32) 🔻	Configuration Manager
1	Common Properties	Generate Debug Info Generate Program Database File Strip Private Symbols	Yes (/DEBUG) \$(OutDir)\$(TargetName)	.pdb
	Debugging VC++ Directories _≡ ▷ C/C++	Generate Map File Map File Name Map Exports Debuggable Assembly	Yes (/MAP) S(TargetName).map Yes (/MAPINFO:EXPORTS)	
	General Input Manifest File			4
3	Debugging System Optimization			
	Embedded IDL Windows Metada [,] +	Generate Debug Info The /DEBUG option creates debuggi	ng information for the .ex	ke file or DLL.
			确定	取消 应用(A)

Figure 5-14

5.1.3.1.4 Generate map file for Delphi program

Project settings as shown below:



Figure 5-15



 Project Options for Project1.exe () Delphi Compiler Compiling Hints and Warnings Linking Output - C/C++ Resource Compiler Directories and Conditionals Build Events Forms Application Packages Runtime Packages Debugger Symbol Tables Environment Block 	Win32 - Debug) Iarget: Debug configuration - 32- Apply Save Debug information V true Save Debug information V true Save Debug information V true Save Debug information V true Save Debug information V true Save Debug information V true Save Debug information V true Save Debug information Sate Apply Viantity Concord Sate Viantity Viantity Concord Sate Viantity Viantity Viantity Viantity Sate Minimum Sate Sate Sate Sate Sate Se
	OK Cancel Help

Figure 5-16

5.1.3.1.5 Generate a map file for vb6.0 program

Add a "LINK" value to the system environment variable. The value is "/MAP". Restart the computer. This compiles and generates the exe program. The map file will not be automatically deleted, but will be retained.

🔛 System	1		- 🗆 ×
< →	Environment Variables	×	✓ Č Search Control Panel Ø
Contre	User variables for yaowb		
Device	Variable	Value	
-	OneDrive	C:\Users\yaowb\OneDrive	
💡 Remot	Path	C:\Users\yaowb\AppData\Local\Microsoft\WindowsApps;	
System	TEMP	C:\Users\yaowb\AppData\Local\Temp	Windows10
💡 Adva	New System Variable	×	
	Variable name: LINK		
_	variable name.		
_	Variable value: /MAP		
_			
	Browse Directory	Browse File OK Cancel	
ļ	Browse Directory E	Value	• • • • • • • • • • • • • • • • • • •
l			Change settings
l	Variable ComSpec DriverData	Value	Change settings
	Variable ComSpec DriverData NUMBER_OF_PROCESSORS	Value C1\WINDOWSkystem32\cmd.eve C1\WIndowsSystem32\Drivers\Drive_bata	Change settings
l	Variable ComSpec DriverData NUMBER_OF_PROCESSORS OS	Value Cr(WINDOWS\system32\cmd.exe Cr(Windows\System32\Drivers\	Change settings
l	Variable ComSpec DriverData NUMBER_OF_PROCESSORS OS Path	Value C:\WINDOWS:system32\cmd.exe C:\WINDOWS:system32\cmd.exe 4 Windows_NT C:\WINDOWS:system32;C:\WINVOWS;C:\WINDOWS:System32\Wb	Change settings
	Variable ComSpec DriverData NUMBER_OF_PROCESSORS OS Path PATHEXT	Value C1/WINDOWS/system32/cmd.exe C1/WINDOWS/system32/Drivers/Driversts 4 Windows, NT C1/WINDOWS/system32/C/WINDOWSC/WINDOWS/System32/Wb C0/WEXE, BAT; CMD, VBS, VBS, USS, WSF, WSF, MSF, MSC	Change settings
	Variable ComSpec DriverData NUMBER_OF_PROCESSORS OS Path	Value C1/WINDOWS/system32/cmd.exe C1/WINDOWS/system32/Drivers/Driversts 4 Windows, NT C1/WINDOWS/system32/C/WINDOWSC/WINDOWS/System32/Wb C0/WEXE, BAT; CMD, VBS, VBS, USS, WSF, WSF, MSF, MSC	Change settings
l	Variable ComSpec DriverData NUMBER_OF_PROCESSORS OS Path PATHEXT	Value C1/WINDOWS/system32/cmd.exe C1/WINDOWS/system32/Drivers/Driversts 4 Windows, NT C1/WINDOWS/system32/C/WINDOWSC/WINDOWS/System32/Wb C0/WEXE, BAT; CMD, VBS, VBS, USS, WSF, WSF, MSF, MSC	Change settings
	Variable ComSpec DriverData NUMBER_OF_PROCESSORS OS Path PATHEXT	Value C1\WINDOWS\system32\cmd.exe C1\WINDOWS\system32\Drives\Drives\bfs 4 Windows,NT C1\WINDOWS\system32\C1\WINDOWS\C1\WINDOWS\System32\Wb C1\WINDOWS\system32\C1\WINDOWS\C1\WINDOWS\System32\Wb COM_EXE,EAT.CMD:VBS,VB_J.St,JSE,WSF,WSF,WSF	
See als	Variable ComSpec DriverData NUMBER_OF_PROCESSORS OS Path PATHEXT	Value C1\WINDOWS\system32\cmd.exe C1\WINDOWS\system32\Drives\Drives\bfs 4 Windows,NT C1\WINDOWS\system32\C1\WINDOWS\C1\WINDOWS\System32\Wb C1\WINDOWS\system32\C1\WINDOWS\C1\WINDOWS\System32\Wb COM_EXE,EAT.CMD:VBS,VB_J.St,JSE,WSF,WSF,WSF	



5.1.3.2 Using the SDK label API to mark the critical functions

This function and related API helps developer to mark and locate the important function that need to be protected by using SDK label API. For the program is written by C++ language, if you want to find the important



function to protect it later. And only the address of the function will be shown which makes difficult to locate the function you want to protect it. So this label API will help you find the function you want to protect, and no need to find the function in thousands lines of code.

The Header file, static library and dynamic library to SDK label API can be found in the Virbox Protector SDK Kit. The software developer can load the label API statically into the function protected. Then the Virbox Protector can locate the critical code and protect those functions.

5.1.3.2.1 Using SDK Label and Protect your functions

Till now Virbox Protector supports to add these kind of label to the functions:

- VBProtectBegin: normal protection
- VBVirtualizeBegin: Virtualization protection
- VBMutateBegin: Obfuscation protection
- VBSnippetBegin: Code snippet protection (Code fragmentation)

VBProtectDecrypt: License encryption and decryption

5.1.3.2.2 Notes:

- 1. SDK label can only be loaded statically and not supported to load dynamic link lib(LoadLibrary);
- 2. The String parameter imported by VBProtectBegin, VBVirtualizeBegin, VBSnippetBegin, VBMutateBegin can't be shared with other functions.
- 3. Make sure the imported parameter to be ASCII code, then the right function name would be shown, otherwise it will show messy code and unreadable.
- Every begin will follow and match an end, use the "begin" and "end" in pair, and only one pair is allowed in one function.
- 5. If the protection mode marked in the label inconsistent with the protection mode saved in the project file. , the system will use the protection mode saved in your project file.
- 6. The code in between the "Begin" and "End" is better to more than 3 lines. To make sure the protected code will be shown in the GUI of Virbox Protector. (it will not be shown in the Virbox Protector for the instruction less 15 bytes.
- 7. SDK provides 32bit and 64bit dll, you do need to use these libs accordingly.
- 8. Not support Java, Unity3D.
- 9. Begin/end do not support nesting use.



- VBProtectDecrypt, the length of the encrypted string or buffer should be multi times of 16,
 i.e.: char g_test_string[16] = {"test_decrypt"};
- 11. VBProtectDecrypt, the input buffer and output buffer can't be the same buffer.
- 12. VBProtectDecrypt, the buffer input need to be put outside of the function, which means is a global variable. For detail how to use, please refer demo.
- 13. .Net program, is not supported by VBProtectDecrypt currently.

5.1.3.2.3 How to encrypt and decrypt the string by SDK

- 1. The encrypted string must be a constant value.
- 2. VBDecryptData also can be used to encrypt and decrypt the data, but the length and the data should be constant value.
- 3. The following type of string is supported:
 - String encryption:
 VBDecryptStringA("test_string");
 - Local static variable: static const char g_string[] = "test_string";
 - global variable: char g_test_string[] = "test_string"; const char g_test_string[] = "test_string"; static const char g_test_string[] = "test_string";

If the program to encrypt is too complicated and the data to be encrypted can't be parsed, it will report error when you protect the software, we recommend you make the program less complicate. Usually it mostly happened to the Linux program which compiled with –fpic or –fpie with O2.

The compiler may merge the same constant string to be one string, if only one of these string is encrypted, an error would be reported:

For example:

const char* a = "test_string"; const char* b = VBDecryptStringA("test_string"); printf("a = %s, b = %s\n", a, b);

For this case, messy code would be shown when you print string "a".



5.1.3.3 Generate .ssp configuration file

.ssp File is the configuration file which need to be used for protected software, here we introduce how to generate the .ssp file.

Generate ssp file manually:

Drag in the program into Virbox Protector, after you completed the configuration of software protection option, the ssp configuration file will be generated in the same path with the program protected after you clicked "Save the selected configuration" or "Save All Configuration".



Figure 5-18

5.1.3.4 Protect software with command line

Command parameter :

Command	Description (Local Dongle)	Description(Cloud Lock)	Note
filename	The files that need to be protected/encrypted		
-u3d	Protect/Encrypt the Unity3D program		/
-o output	o output The output directory of the protected/encrypted program		



5.1.3.4.1 Using Command line to protect the application in Linux Environment

5.1.3.4.1.1 Normal application protection/encryption

Command line in Linux system:

For the protection/encryption of normal application in Linux environment, here we take the Linux platform program as an example:

Use the Virbox Protector GUI tool to generate configuration files (optional)

- If a configuration file is generated, you can select the function number and protection mode in the GUI tool
- If no configuration file is generated, only the default entry function will be protected

Open a terminal in linux platform, enter the path where "virboxprotector_con" is located, and enter "virboxprotector_con" to run Virbox Protector

Help information can be viewed:





For the programs executed in different platforms, the Virbox Protector has different edition & license. You need to contact Virbox team to get the corresponding license.

Command: Path of VirboxProtector_con path of the program to be protected -o path of output file

• If the license is not verified, when you use Virbox Protector to protect program, it will prompt "Can not find the license", as shown in the figure:



<pre>sense@sense:~/Desktop/virboxprotector_1.5.0.10808/bin\$./virboxprotector_con '/h iome/sense/Desktop/virbox_test/cb_bytes_test' -o '/home/sense/Desktop/virbox_test /cb_bytes_test.ssp.vp' SenseShield Virbox [version: 1.5.0.10808] Copyright(c) SenseShield Technology Co., Ltd. All rights reserved.</pre>
loading cb_bytes_test Error (13000020): Can not find the license.

Figure 5-20

• After the license have been obtained, you can protect the program with Virbox Protector successfully, as shown in the figure:

```
sense@sense:~/Desktop/virboxprotector_1.5.0.10808/bin$ ./virboxprotector_con '/h
ome/sense/Desktop/virbox_test/cb_bytes_test' -o '/home/sense/Desktop/virbox_test
/cb_bytes_test.ssp.vp'
SenseShield Virbox [version: 1.5.0.10808]
Copyright(c) SenseShield Technology Co., Ltd. All rights reserved.
loading cb_bytes_test ...
link...
save as cb_bytes_test.ssp.vp ...
Succeed.
```

Figure 5-21

5.1.3.4.1.2 Using Command Line to protect the Unity3D program

Unity3D, as a special file type, the protection methods is different from the normal program. For the Unity3D program for Windows, Linux and macOS platforms, the entire directory of Unity3D needs to be protected; for Unity3D for the Android platform, the apk of Unity3D program needs to be protected. Here we take a Linux Unity3D as an example:

- Use the Virbox Protector GUI tool to generate configuration files (optional)
- Open a terminal window, enter the path where "virboxprotector_con" is located, and enter "virboxprotector_con" to run Virbox Protector. Help information can be viewed.
- For the programs in different platforms, the Virbox Protector need to verify the license in different platform. You need to contact Virbox team to obtain the corresponding license.

Command: Path of VirboxProtector_con path of the program to be protected -u3d –o Path of output file

If no license has been verified, when you run Virbox Protector, it will prompt "**Can not find the license**", as shown in the figure:



<pre>sense@sense:~/Desktop/virboxprotector_1.5.0.10808/bin\$./virboxprotector_con '/h ome/sense/Desktop/Particles2018.1.9f1' -u3d -o '/home/sense/Desktop/ssp.Particle s2018.1.9f1' SenseShield Virbox [version: 1.5.0.10808] Copyright(c) SenseShield Technology Co., Ltd. All rights reserved.</pre>
protect unity3d Particles2018.1.9f1 Error (13000020): Can not find the license.

Figure 5-22

After the license is verified, the program can be successfully protected by Virbox Protector, as shown in the figure:

sense@sense:~/Desktop/virboxprotector_1.5.0.10808/bin\$./virboxprotector_con '/h
ome/sense/Desktop/Particles2018.1.9f1' -u3d -o '/home/sense/Desktop/ssp.Particle
s2018.1.9f1'
SenseShield Virbox [version: 1.5.0.10808]
Copyright(c) SenseShield Technology Co., Ltd. All rights reserved.
protect unity3d Particles2018.1.9f1 ...
Succeed.



5.1.3.4.2 Using Command line to protect executables in Windows environment

5.1.3.4.2.1 Take Windows application as an example

Use the Virbox Protector GUI tool to generate configuration files (optional)

- If a configuration file is generated, you can select the quantity of protected function and protection mode in the GUI tool
- If no configuration file is generated, only the entry function on default will be protected

Open terminal in window from the start menu, input:

Virbox Protector_con.exe

To start execution.

This command can get the help info.

Command line help info:

VirboxProtector_con <filename> [-o output]

-o output : output file name.

-? : show help information.





Figure 5-24

When you use the command without license, it will have this error report: "Can not find the license".

Difference license of Virbox Protector will be required when you are protecting the program on different platform. Please contact Virbox Team for the corresponding license.

Command: path of *Virbox Protector_con.exe* the path of the program that to be protected -o path of output file.



Figure 5-25

After the license is verified, you can complete protection.



VirboxProtector_con <fi] -o output -?</fi] 		
SenseShield Virbox [vers	r Standalone\bin>virboxprotector_con.exe PE_Tetris.ex sion: 1.4.2.9353] d Technology Co., Ltd. All rights reserved.	e PE-protected.exe
loading PE_Tetris.exe link save as PE_Tetris.ssp.e; Succeed.		
D:\test\Virbox Protector	r Standalone\bin>	



5.1.3.4.2.2 Unity3D program protection:

- Use the Virbox Protector GUI tool to generate configuration files (optional)
- Open a terminal window, enter the path where "virboxprotector_con.exe" is located, and enter "virboxprotector con.exe" to run Virbox Protector. Help information can be viewed.
- For the programs of different platforms, Virbox Protector has different license. You need to contact Virbox team to obtain the license.

Command: path of *Virbox Protector_con.exe* the path of the program that to be protected –u3d -o path of output file.

If no license have been found, when you protect the software it will show:



Figure 5-27

If the license is successfully found and protect the program successfully:



Figure 5-28



5.2 Protect the .Net application

Drag the execute file you want to protect into Virbox Protector:



Figure 5-29

5.2.1 Protect the .NET application in fundamental

Protection Option Tag to .NET application as shown below:


Virbox Protector 2 (64-bit) v2.0.1.12323			- 🗆 X
File Protect Plug-in Log Setting Help			
	*	~"	Authorization Info
🗁 E E		2	Account: 15010303968 Logout
Open File Save Selected Configuration Save All Con	figuration Protect Selected Projects Protect A	All Project	Authorization Detail
File/Directory	🗢 Basic Info 🛛 🚱 Function Options	Protection Options	
Protect successfully!	✓ Output Info		^
angrybots5.5.3.apk	Output File/Directory	.Net_RetroSnaker.ssp.exe	•••
Protect successfully! If it's an executable program,	 Protection Options 		
.Net_RetroSnaker.exe	Compression	\checkmark	
Parse successfully!	JIT Encrypt		
		_	
	Name obfuscation	\checkmark	
	✓ Plug-in		
	✓ Anti Debug Option	\bigcirc	
	Detecting Hardware Breakpoints		
	Detecting Memory Breakpoints		
	Memory Integrity Check		
			`
E:/Test-Program/.Net_RetroSnaker.exe			.NET AnyCPU (i386

Figure 5-30

5.2.1.1 Name Obfuscation

Rename the .Net program method name and class name with random string, the name that exported for external call would not change.

Protection Comparison:

Before protection:





After Protection:







5.2.1.2 Compression

The main purpose of the compression is not only to compress the software, it will encrypt the code and the data segment and hide the original import table and relocate the information of the protected application, and compressed the original data size at same time.

The compression will protect the .Net program overall, in this way to protect the method being anti-compiled by DnSpy, ILSpy and .Net Reflector. This provide better compatible by using IL enveloper code.

Protection Mechanism

This function will pack the original data segment with data package and compress, replace the Original Entry Point (OEP) with the packer code. The data segment and code segment will be retrieved when the program is executed, and relocate it to execute the program.

The Purpose:

Prevent the static anti-compile and prevent the program being patched

The Benefit:

- 1. Hide the code, data and file structure information of the program, protect the software in general.
- 2. High efficiency when the program executed, and small impact to performance when the program is loaded.

The Weakness:

1. When the pack code is executed, the code segment and data segment may be retrieved and be dumped.

Protection Comparison:



Before Protection:

I ☐ xmltest (1.0.0.0)	1 // Token: 0x02000002 RID: 2
▲ ■ xmltest.exe	2 .class private auto ansi beforefieldinit xmltest.Program
Þ ≅ PE	3 extends [mscorlib]System.Object
▶ ■■ 引用	4 { 5 // Methods
▶ {} -	6 // Token: 0x06000001 RID: 1 RVA: 0x00002050 File Offset: 0x00000250
4 {} xmltest	7 .method private hidebysig static
4 🐪 Program @02000002	8 int32 Main (
▶ ■ 基类型和接口	9 string[] args
▶ 💼 派生类型	10) cil managed
♀ .ctor() : void @06000004	
ଦିଳ findNode(XmlNodeList, string) : Xml	12 // Header Size: 12 bytes
ଦ୍ନ info() : void @06000002	13 // Code Size: 610 (0x262) bytes
ଦ୍ଧ Main(string[]) : int32 @06000001	14 // LocalVarSig Token: 0x11000001 RID: 1
> 🗗 mscorlib (4.0.0.0)	15 .maxstack 5
>	16 .entrypoint 17 .locals init (



After protection:

☐ xmltest4.5 64.ssp.exe	
A 🖬 PE	22 // 所有树节点都使用十六进制编辑器来修改 PE 文件
DOS 头	
	4 // 00000000 - 0000003F DOS 头
🖬 可选头 (64 - 位)	6 // IMAGE_DOS_HEADER:
■ 章节 #0: .text	7 // 00000000 - 00000001 5A4D = e_magic
🖬 章节 #1: .vdata0	8 // 00000002 - 00000003 0090 = e_cblp
□ 章节 #2: .virbox	9 // 00000004 - 00000005 0003 = e_cp
■ 章节 #3: .vdata1	10 // 00000006 - 00000007 0000 = e_crlc
□ 章节 #4: .virbox3	11 // 00000008 - 00000009 0004 = e_cparhdr
□ 章节 #5: .reloc	12 // 0000000A - 0000000B 0000 = e_minalloc
章节 #6: .rsrc	13 // 0000000C - 0000000D FFFF = e_maxalloc



5.2.1.3 JIT encryption

- .Net JIT encryption will encrypt all of the method IL instruction of the .Net Program, and only in the JIT compile process of the .Net Virtual Machine the instruction will be decrypted, This can be used to prevent static anti-compile and prevent the IL code being Dumped in memory.
- JIT encryption will encrypt all of the method in default and enhance the security level of the source code after protection.
- JIT encryption .support inheritance, event, reflection, recursive call which is not supported in general encryption solution.

Usage Guide:

Drag the .Net program into Virbox Protector, the "Protection Option" will show "JIT encryption" option. You need to select this option if you want to use this function to protect your program.

Copyright © 2020, Virbox, All Right Reserved.



File/Directory		Protection Options	
Parse successfully!	 Output Info Output File/Directory 	.Net_RetroSnaker.ssp.exe	•••
	Protection Options	.net_redusnaker.ssp.exe	
	Compression		
	JIT Encrypt		
	Name obfuscation		
	✓ Plug-in		
	 Anti Debug Option 	\bigcirc	
	Detecting Hardware Breakpoints		
	Detecting Memory Breakpoints		
	Memory Integrity Check		
	Ƴ ds	\bigcirc	Open DSProtector
	Password	•••••	
E:/Test-Program/.Net_RetroSnaker.exe			.NET AnyCPU i386

5.2.1.4 Remove Strong Name

- 1. StrongName provides the .Net program with a mechanism which adding the label of version and label of the original author information.
- 2. StrongName can be used to help the software user to verify if the program comes from the original author and not be modified (prevent tampering).
- 3. So the software developer need to remove the strong name before protection/encryption and add the Strong name after protection/encryption again.

5.2.2 Protect the critical Functions

Function Option tag:



V Add Functions			- C X
Global Search Functions			Analysis
Total Protected: 2/79 Obfuscation: 2 Virtualization:0 E	ncrypted: 0		
Tip: The unsupported protect mode of the functions will be known	wn after protecting the program! Only show	functions that were set the unsupported pro	tection way
Name	Run Times	Protection Type Other	^
▶ 🔇 ☆食蛇			
> 🧿 SnakeLabel			
✓ 🧿 SnakeIO			
💹 void SaveTaget(string)		Obfuscation 💌	
📀 void SaveSnake(string)	-	No protected 🔹	
void LoadTaget(string)		Encryption 💌	
void LoadSnake(string)	-	Encryption 💌	
void .ctor(Snake)	1	Encryption 💌	
Point Convert(string)		Encryption 🔻	
> 🧿 Snake			~
Total Show Functions: 79		Cance	el Ok

Figure 5-35

Function Level Protection:

5.2.2.1 Code encryption (.Net)

Protection Mechanism

Code encryption is dynamic code protection technology, the original method byte code will be encrypted and decrypted only when the method is executed.

The Purpose:

Prevent the program from being unpacked and being dumped

The Benefit:

Almost no impact to software performance.

The Weakness:

The method of the program may be analyzed when decrypted to execute.

Protection Comparison:

Before protection:





Figure 5-36

After protection:



Figure 5-37

Following scenario is not supported to code encryption:

For C# program, maybe the message of "Part of the function has been set the unsupported protection technology, and please modify the protection option you used. Error code: 0xA000A000"

Following scenario is not supported to "code encryption":

- 1. The non-static method of Value type: System.ValueType (and their Inherited class)
- 2. Generic methods are not supported



- 3. C++ .net not supported
- 4. Recursive calling is not supported
- 5. Variable parameters are not supported
- 6. Default parameters are not supported

5.2.2.2 Code Obfuscation

Virbox Protector will translate the code instruction into a stream of pseudo-code that neither the machine nor the human can recognize. When the pseudo-code executed, the software will translate and interpret to restore the code into the original code. To let it executable when it is executed. Virbox Protector supports the obfuscation for x86/arm .net il serial instruct.

Protection Mechanism:

To obfuscate the original instruction and prevent from being static analyzed.

The Benefit:

Prevent from anti-compiled and make the hacker more difficult to analysis the code.

The Weakness:

Negative impact to software performance. Limited protection to software.



Before protection:

呈序集资源管理器 ▼ ×	CalBlank(int32, int32) : int32 ×
	<pre>// Token: 0x06000000 RID: 14 RVA: 0x0000259E File Offset: 0x0000079Emethod private hidebysig instance int32 CalBlank (</pre>
Solution in the second seco	20

Figure 5-38

After Protection:

	Carbiank(intsz, intsz) : intsz 🔿	
> { } loader	1 // Token: 0x0600000F RI	D: 15 RVA: 0x0002E8D0 File Offset: 0x0002B6D0
() 贪食蛇	.method private hidebys	ig
Auto @02000002	3 instance int32 CalB	lank (
AutoShop @0200008	4 int32 x,	
4 🐁 AvoidDeadAuto @02000003	5 int32 y	
▶ ■ 基类型和接口	6) cil managed	
	8 // Header Size: 12	
♀ .ctor(Snake) : void @06000007	9 // Code Size: 111 (
♀ AvoidDead() : void @06000008	10 .maxstack 9	oxor) bytes
CalBlank(int32, int32) : int32 @06(11	
♀ CalBlankSon(int32, int32) : int32 €	12 /* 0x0002B6DC 2B38	*/ IL_0000: br.s IL_003A
© CanToTarget() : bool @0600000E		
♀ Convert(Point) : Direction @06000		*/ IL_0002: stind.ref
♀ Convert(Direction) : Point @06000		*/ IL_0003: and
P InitPoints() : void @0600000D	16 17 /* 0x0002B6E0 2B11	*/ IL_0004: br.s IL_0017
ଦ୍ଧି IsBlank(int32, int32) : bool @0600		*) 1L_0004. DI.S 1L_0017
IsCanGo(Direction) : bool @06000	19 /* 0x0002B6E2 56	*/ IL 0006: stind.r4
© IsOut(int32, int32) : bool @06000	20 /* 0x0002B6E3 4B	*/ IL_0007: 1dind.u4
JudgeDirection(): void @0600001		
🤹 canEatTarget : bool @04000004		*/ IL_0008: br.s IL_0030
💁 outIndex : int32 @04000005	23 24 /* 0x0002B6E6 66	*/ IL 000A: not
outPoint : Point @04000006	24)* 0x0002B0E0 00 25 /* 0x0002B6E7 59	*/ IL_0008: sub
points : class [mscorlib]System.Cc	26	*/ IL_000D. Sab
▶ 🔩 <>c DisplayClass1 @02000005	27 /* 0x0002B6E8 03	*/ IL_000C: ldarg.1
▶ 🔩 <>c DisplayClass3 @02000006		*/ IL_000D: br.s IL_0022
• 🔩 OutParameter @02000004		
BypassAuto @02000007	30 /* 0x0002B6EB 09	*/ IL_000F: 1dloc.3
▶ 🖶 Direction @0200000C		*/ IL_0010: ldind.i2
EatTargetAuto @02000009	32 33 /* 0x0002B6ED 28100	00006 */ IL 0011: call instance int32 贪食蛇.AvoidDeadAuto::CalBlankSon
frmFactory @0200000A	(int32, int32)	obbo / In_corr. call instance intox of the more dealer and the second of
🕨 🔩 frmGame @0200000D	34 /* 0x0002B6F2 2A	*/ IL 0016: ret
Program @0200000E		
Snake @0200008		*/ IL_0017: ldc.i4.1
SnakeControl @02000011	37 /* 0x0002B6F4 2D02	*/ IL_0018: brtrue.s IL_001C



You can select the corresponding protection option according to the introduction of the function.

Then click "Protect all project" button to complete protection:

Copyright © 2020, Virbox, All Right Reserved.



Virbox Protector Standalone x64 (1.4.2.9340)			- 🗆 X
le Protect Plug-in Log Setting Help			
	6 🔗	\heartsuit	
Open File Save Selected Configuration Save All	Configuration Protect Selected Projects Prot	ect All Project	
File/Directory	😏 Base Info 🛛 💿 Function Options	Protection Options	
Net_RetroSnaker.exe Parse successfully!	✓ File		
	Output File/Directory	.Net_RetroSnaker.ssp.exe	•••
	 Protection Options 		
	Compression		
	Name obfuscation		
	✓ Plug-in		
	✓ Anti	\bigcirc	
	Anti hardware breakpoint		
	Anti memory breakpoint		
	Enable memory Check		
	✓ ds	\bigcirc	Open DSProtector
	Password	•••••	
C:/Users/luyonggiang/Desktop/.Net. RetroSnaker.e:	78		.NET AnyCPU i386

Figure 5-40

5.3 Java Program protection:

Before Java program Protection:

5.3.1 Protection background and introduction

Java program support cross platform operation which rely on the java execute in the Virtual machine environment as intermediate code, the challenge to protect java is: the de-compilation to java code is much easier compare with to de-compilation to other languages. And the decompiled code is almost compatible with the source code after optimization.

There are many Java Obfuscator available in the market to protect Java Application. The mechanism of the Java obfuscator is to obfuscate the compiled code, and makes the decompiled code difficulty to read and understand. And increasing the difficulty to reverse engineering. For the people who familiar to use the de-compilation tool. It is almost transparent. So the security level for the Java application protected by Java Obfuscator is quite limited.

After compile of the Java source code, it is easy to anti-compile the .class file which contains the class name, method name and variable name. The hacker almost can obtain the source code which is completely same with the original source code by anti-compile engineering.

Virbox support directly protect the Jar archive and War archive. Which will encrypt the byte code of every method to prevent the source code from being anti-compiled and it is quite easy to operate with Virbox Copyright © 2020, Virbox, All Right Reserved. Page 80 | 108



Protector. Currently the source code from Windows, Linux, ARM Linux platform are supported to be protected.

5.3.2 Protect to Jar archive

1. Drag the Jar archive directly into the Virbox Protector:



2. After protection, an encrypted Jar archive and sjt plugin would be generated, like the picture showing:

5	sp.Jar_protectio	on							
e	Share V	iew							
ру	Dacte	t py path ste shortcut	Move Copy to *		New item ▼ 1 Easy access ▼ 1 Folder	Properties	Belect no	ne	
C	lipboard		0	rganize	New	Open	Select		
^	→ This PC	› Local Disk	(E:) → Test-P	rogram > Java-Prog	gram-demo > ssp.Jar_pro	otection		~ Ö	Search ss
	^	Name	^		Date modified	Туре	Size		
			0.0.1-SNAPSH	IOT-ori.jar	10/19/2020 3:15 PM	JAR File	16,934 KB		
			ıx_a32.so ıx_a32hf.so		10/19/2020 3:15 PM 10/19/2020 3:15 PM	SO File SO File	447 KB 451 KB		
ents			ıx_a64.so ıx_x64.so		10/19/2020 3:15 PM 10/19/2020 3:15 PM	SO File SO File	874 KB 507 KB		
ads		📄 sjt_linu	ıx_x86.so		10/19/2020 3:15 PM	SO File	298 KB		
		🗟 sjt_win	dows_x64.dll		10/19/2020 3:15 PM	Application extens	480 KB		
		🖄 sjt_win	idows_x86.dll		10/19/2020 3:15 PM	Application extens	406 KB		
sk (C:)								
sk (D:)								
isk (E:)								



5.3.2.1 Deployment

Windows:

Add the environment variable to the computer:

1. Put the sjt library into a fixed directory, put the "sjt_windows_x64.dll" and "sjt_windows_x86.dll" to the system environment veriable.

"This PC"->"Properties"->"Advanced system settings"-> "Environment variables" ->"New..."

->"Environment variable"->"system variable", new variable: "JAVA_TOOL_OPTIONS", variable value:
"-agentpath:C:\Users\test\Desktop\sjt\sjt_windows_x64.dll"



2. After you configured the environment, you can directly run the Jar archive, or directly call the Jar archive.

Directly run the protected Jar archive

- If Java version is 64 bit version, you need to assign the "*sjt_windows_x64.dll*", If the Java version is 32 bit, you need to use "*sjt_windows_x86.dll*".
 If sjt library and the Jar archive are in the same directly, you can directly run the following command in the current Jar archive directly.
 Command: *java -agentpath:sjt_windows_x64.dll -jar ***.jar*
- If sjt library and jar archive is not in the same directly, you need to assign the full directory.
 Command: java -agentpath:C:\Users\test\Desktop\sjt\sjt_windows_x64.dll -jar ***.jar

Linux System:

Copyright © 2020, Virbox, All Right Reserved.



- Add JAVA_TOOL_OPTIONS=-agentpath:/home/sense/Desktop/sjt_so/sjt_linux_x86.so to /etc/profile environment variable.
- Input: Source /etc/profile to make the configure file take effect.
- After you have completed the environment configuration, directly run jar archive or call Jar archive.

export JAVA_HOME=/usr/local/java/jdk1.8.0_261 export JRE_HOME=\${JAVA_HOME}/jre export CLASSPATH=.:\${JAVA_HOME}/lib:\${JRE_HOME}/lib export PATH=\${JAVA_HOME}/bin:\$PATH
sense sjt
export JAVA_TOOL_OPTIONS="-agentpath:/home/sense/Desktop/sjt_so/sjt_linux_x86.so "

Directly run the protected program:

If you are using 64 bit Java, you need to use "*sjt_windows_x64.dll*", for 32 bit java please use

"sjt_windows_x86.dll";

- If the sjt library are in the same directory with Jar archive, you can run the following command in the current directory:
- Command: *java -agentpath:sjt_windows_x64.dll -jar ***.jar*
- If the sjt library is not in the same directory with the jar archive, you need to assign the complete directory:
- Command: java -agentpath:C:\Users\test\Desktop\sjt\sjt_windows_x64.dll -jar ***.jar

Please noted that after configuration of the environment variable, the Jar archive will use the environment variable even you have assigned the sjt library location.

Jar archive Protection performance compare:

(Check the encryption performance by jd-gui)

Before protection:



💀 FileUseDemo. jar 🛛 💀 FileUseDemo. jar 🎗	
FileUseDemo. jar 23 FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 23 Image: FileUseDemo. jar 24 Image: FileUseDemo. jar 25 Image: FileUseDemo. jar 25 Image: FileUseDemo. jar 25 Image: FileUseDemo. jar 26 Image: FileUseDemo. jar 27 Image: FileUseDemo. jar 28 Image: FileUseDemo. jar 28 Image: FileUseDemo. jar 28 Image: FileUseDemo. jar 28 Image: FileUseDe	Dim FileUseDemo\$1. class ※ Dim FileUseDemo\$RenameFileAction. class ※ Dim FileUseDemo. class * Dim FileUseDemo. class
	<pre>21 JPanel panel2 = new JPanel(); 22 panel2.add(new JButton(new <u>CreateFileAction())</u>); 23 panel2.add(new JButton(new <u>RenameFileAction())</u>); 24 panel2.add(new JButton(new <u>DeleteFileAction())</u>);</pre>

After protection:



5.3.3 War archive protection:

Virbox Protector will protect the class file in the War archive.

- 1. Drag War archive into Virbox Protector
- 2. After successfully encrypted, *sjt* plugin and the protected War archive would be generated.



比电脑 > 本地磁盘(C:) > 用户 > test	› 桌面 › ssp.test_war		
~ 名称 ~	修改日期	类型	大小
🗋 hello.war	2020/9/30 10:36	WAR 文件	4 KB
🗋 myhome1.war	2020/9/30 10:36	WAR 文件	26,403 KB
sample.war	2020/9/30 10:36	WAR 文件	5 KB
📄 sjt_linux_a32.so	2020/9/30 10:36	SO 文件	188 KB
📄 sjt_linux_a32hf.so	2020/9/30 10:36	SO 文件	188 KB
📄 sjt_linux_a64.so	2020/9/30 10:36	SO 文件	666 KB
📄 sjt_linux_x64.so	2020/9/30 10:36	SO 文件	218 KB
📄 sjt_linux_x86.so	2020/9/30 10:36	SO 文件	32 KB
🗟 sjt_windows_x64.dll	2020/9/30 10:36	应用程序扩展	129 KB
sjt_windows_x86.dll	2020/9/30 10:36	应用程序扩展	110 KB

Deployment:

Windows system:

Following 2 ways to configure the system, you can select one of them:

- 1. Add it to the system variable:
- You can put the sjt library into a fixed directory, and add the "sjt_windows_x64.dll" and "sjt_windows_x86.dll" to the system environment variable.
 "This PC"->"Properties"->"Advanced system settings"-> "Environment variables" -> "New..."
 "Environment variable"->"system variable", new variable: "JAVA_TOOL_OPTIONS", variable value: "-agentpath:C:\Users\test\Desktop\sjt\sjt_windows_x64.dll"





IF you use tomcat to run the protected program, you need to put encrypted archive into this location:

.\apachetomcat\webapps directory, and start tomcat service.

2. Set the setenv.bat in the tomcat directory:

Set the setenv.bat in the tomcat\bin directory:

Set the setenv.bat in the tomcat\bin directory, such as:

Create "setenv.bat" in the tomcat\bin directory, set the environment variable such as (full path):
 Set CATALINA_OPTS=%CATALINA_OPTS% -agentpath:sjt_windows_x64.dll

→ 此电	l脑 → 系统 (D:) → apache-tomcat-9.0.0.M	21 → bin			ن ب
^	名称 ^	修改日期	类型	大小	
	commons-daemon-native.tar.gz	2017/5/4 23:43	GZ 乂任	201 KB	
*	configtest.bat		Windows 批处理		
*	configtest.sh	2017/5/4 23:43	SH 文件	2 KB	
*	daemon.sh	2017/5/4 23:43	SH 文件	8 KB	
*	💿 digest.bat	2017/5/4 23:43	Windows 批处理	3 KB	
	📄 digest.sh	2017/5/4 23:43	SH 文件	2 KB	
	💿 service.bat	2017/5/4 23:43	Windows 批处理	7 KB	
	💿 setclasspath.bat	2017/5/4 23:43	Windows 批处理	4 KB	
	setclasspath.sh	2017/5/4 23:43	SH 文件	4 KB	
	💿 setenv.bat	2020/9/24 18:13	Windows 批处理	1 KB	
	💿 shutdown.bat /	2017/5/4 23:43	Windows 批处理	2 KB	
- 10	📄 shutdown.sh	2017/5/4 23:43	SH 文件	2 KB	
	\apache-tomcat-9.0.0.M21\bin\setenv.bat -	-			- 0
) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言				
🖪 📥) 🗄 🖻 💦 🔓 🕹 🕹 👘 🛅 🔁 🗲	🛍 🍢 🔍 🔫 🕼	🗟 🚍 1 📭 🖉	I 💹 🕗 🖿 🧶	🕨 🔳 🕨 🔤
📄 set	env. bat 🗵 🕨				
1	set CATALINA_OPTS=%CATALINA_OP	TS% -agentpath:C:	\Users\test\Des	ktop\sjt\sjt	windows_x64.dll

 Put the encrypted War archive into the location: .\apache-tomcat\webapps and start the tomcat service.

Linux System:

Add the library into the /etc/profile environment variable:

JAVA_TOOL_OPTIONS=-agentpath:/home/sense/Desktop/sjt_so/sjt_linux_x86.so into the /etc/profile



export JAVA_HOME=/usr/local/java/jdk1.8.0_261
export JRE_HOME=\${JAVA_HOME}/jre
export CLASSPATH=.:\${JAVA_HOME}/lib:\${JRE_HOME}/lib
export PATH=\${JAVA_HOME}/bin:\$PATH

sense_sjt
export JAVA_TOOL_OPTIONS="-agentpath:/home/sense/Desktop/sjt_so/sjt_linux_x86.so
"

- Source */etc/profile* environment variable to make it take effect.
- You can directly start the tomcat service after you have configured the environment and put the War archive in the directory: .\apache-tomcat\webapps

Setenv.sh in the tomcat directory:

 Create a new setenv.sh in the tomcat\bin directory, the full path environment variable can be set as follows: CATALINA_OPTS=%CATALINA_OPTS% -agentpath:sjt_windows_x64.dll .



Start tomcat service, you can set the CATALINA_OPTS parameter.

	onede scopped
	root@sense:/usr/local/apache-tomcat-8.5.58/bin# ./startup.sh
ł	Jsing CATALINA_BASE: /usr/local/apache-tomcat-8.5.58
	Jsing CATALINA_HOME: /usr/local/apache-tomcat-8.5.58
	Jsing CATALINA_TMPDIR: /usr/local/apache-tomcat-8.5.58/temp
	Jsing JRE_HOME: /usr/local/java/jdk1.8.0_251/jre
	Jsing CLASSPATH: /usr/local/apache-tomcat-8.5.58/bin/bootstrap.jar:/usr/lo
1	cal/apache-tomcat-8.5.58/bin/tomcat-juli.jar
	Jsing CATALINA_OPTS: -agentpath:/home/sense/Desktop/sjt_so/sjt_linux_x64.so
	JSING CATALINA_PID: /USF/LOCAL/APACNE-TOMCAT-8.5.58/TOMCAT.PID
	Tomcat started.
	<pre>coot@sense:/usr/local/apache-tomcat-8.5.58/bin#</pre>

Put the encrypted war package in the directory: .\apache-tomcat\webapps
 If the War package can be parsed correctly, the webpage can run correctly.

Please noted that: If you have configured the environment variable, the default Java running environment will use the environment variable you have set even you have assigned the *sjt* library location.



Protection performance compare:

Please noted that, currently only the class file in the war archive would be encrypted.

The following is the encrypt performance comparing by using jd-gui.exe to check the anti-compile protection to the source code:

Before protection:



After protection:





5.4 Unity 3D Program Protection

Why we need to protect the Gaming program based on Unity3D engine?

5.4.1 Introduction

The Unity 3D program mainly uses the C# and open source mono to execute the code logic and algorithm. All of the code is not compiled into the exe file and located at

{APP}\build\game_Data\Managed\Assembly-CSharp.dll (note that the program with Unity-2017 is slightly different).

And the mono execution is compatible with the Microsoft .NET Framework but the execution principle is completely different. The traditional protection to .NET Framework will be invalid to protect the mono execution. Since Assembly-CSharp.dll is neither a dynamic library in PE format nor a dynamic library in .NET, it cannot be loaded from the .NET Framework. Instead, the mono.dll read the C# script inside of Assembly-CSharp.dll. Interpreter it and execute the program.

If you protect the Unity3D program with traditional software protection tool, it would not protect the main code source. But Virbox Protector will not only protect the source code, but also protect your resources (.resS). To protect your copyright and IP.

5.4.2 Protection Mechanism

Virbox Protector protects the whole source directory of the **Unity3D** program, for the resource file, Virbox Protector will use "**Resource Encryption**" to protect it. In this way to protect your software main source code from being decompiled. And prevent your resources **(.resS)** from being extracted illegally. To protect the copyright and IP of the software developer.

Protection Mechanism:

1. Parse the Assembly-CSharp.dll script file and convert the function into IL code.

2. Encrypt the IL code, where the key is randomly generated every time and kept in the script file.

3. Link and regenerates the Assembly-CSharp.dll script file. All codes have been encrypted.

4. Process the .NET runtime library mono of Unity3D, locate the function that parses the .NET method and add hook.

5. Insert the hook code to decrypt the Assembly-CSharp.dll method, recompile to generate a new mono and replace the original dynamic Library

Copyright © 2020, Virbox, All Right Reserved.



The Purpose:

- 1. Encrypt the Unity3D script C# code, prevent reverse engineering and anti-compile;
- 2. Add the program set file: the c# program set in the managed directory which is developed by the software developer.

5.4.3 Windows, Linux, macOS platform protection

5.4.3.1 Protect with Virbox Protector GUI

Drag the whole U3D directory to the Virbox Protector

	 Virbox Protector Standalone x64 (1.4.2.9340) 		-		×
	File Protect Plug-in Log Setting Help				
	Open File Save Selected Configuration Save All Configuration	figuration Protect Selected Projects Protect Al Project			
	File/Directory	Welcome Page			
		Welcome to Virbox Protector !			
Unity3D		Start Open File Open Directory(Only support Unity3D)			
	Drag file here to protect!	Recent Files/Directory Java.exe .Net,RetroSnaker.exe Unity30 C:/Uars/luyongqiang/Deaktop/angr/bots5.5.3.apk hid.dll			
	C:/Users/luyongqiang/Desktop/java.exe		PE 64	H-bit amd6	54



The Assembly-CSharp-firstpass.dll and Assembly-CSharp.dll would be loaded in the "Protection Options" tag, you can add the customized C# program set in the /Managed directory.



Virbox Protector 2 (64-bit) v2.0.1.12323			– 🗆 X
File Protect Plug-in Log Setting Help	figuration Protect Selected Projects Protect All		Authorization Info Account: 15010303968 Logout Authorization Detail
File/Directory	Basic Info G Function Options Assembly	Protection Options Resource Encryp	otion +
Parse successfully!	 Assembly jiamigou_Test_Data/Managed/Assembly- jiamigou_Test_Data/Managed/Assembly- 		т Х Х
	 Output Info Output File/Directory 	ssp.Unity3D	•••
	 Plug-in Anti Debug Option 	0	
	Detecting Hardware Breakpoints Detecting Memory Breakpoints		
	Memory Integrity Check		
E:/Test-Program/Unity3D/室内走廊游戏/室内走廊游戏/	Jnity3D		Activa hivat ing

Figure 5-42

You can also enable the "Resource Encryption" option such as the picture such as the following shows to encrypt the resources in your Unity3D program:

Please noted that: It is recommended that only the resources selected by default will be encrypted.

e Protect Plug-in Log Setting Help		
		Authorization Info
E		Account: 15010303968 Logout
pen File Save Selected Configuration Save Al	Configuration Protect Selected Projects Protect All Project	Authorization Detail
File/Directory	Basic Info Function Options Protection Options Resource Encryption	
Parse successfully!	Enable (It is recommended that only the resources selected by default be encrypted)	Add File
	File	
	jiamigou_Test_Data/globalgamemanagers	×
	jiamigou_Test_Data/globalgamemanagers.assets	×
	jiamigou_Test_Data/level0	×
	jiamigou_Test_Data/Resources/unity default resources	×
	jiamigou_Test_Data/Resources/unity_builtin_extra	×
	jiamigou_Test_Data/sharedassets0.assets	×
	jiamigou_Test_Data/sharedassets0.assets.resS	×
	jiamigou_Test_Data/sharedassets0.assets.resS.bak	×
::/Test-Program/Unity3D/室内走廊游戏/室内走廊游	转入Unity3D	Unity30

Figure 5-43



For the configuration of "**functions to be protected**", we do not need to set, Virbox Protector will protect all of the functions in "**Assembly-CSharp.dll**" and "**Assembly-CSharp-firstpass.dll**".

After protection, Virbox Protector will generate a new folder, same directory with the original folder named ssp.xxx (xxx is the original directory name), the picture shown below.

Two more files would be generated:



And it will remind you protection successful.

"Unity3D-Test.ssp" is the configuration file you may need to use for resources protection.

"ssp.Unity3D-Test" folder is the Unity3D program after protection, you can distribute this file to the software user in future.

5.4.3.2 Using Command Line to protect the Unity3D program

Unity3D, as a special file type, the protection methods is different from the normal program. For the Unity3D program for Windows, Linux and macOS platforms, the entire directory of Unity3D needs to be protected; Here we take a Linux Unity3D as an example:

- Use the Virbox Protector GUI tool to generate configuration files (optional)
- Open a terminal window, enter the path where "virboxprotector_con" is located, and enter "virboxprotector_con" to run Virbox Protector. Help information can be viewed.
- For the programs in different platforms, the Virbox Protector need to verify the license in different platform. You need to contact Virbox team to obtain the corresponding license.

Command: Path of VirboxProtector_con path of the program to be protected -u3d –o Path of output file

If no license has been verified, when you run Virbox Protector, it will prompt "Can not find the license", as shown in the figure:



<pre>sense@sense:~/Desktop/virboxprotector_1.5.0.10808/bin\$./virboxprotector_con '/h ome/sense/Desktop/Particles2018.1.9f1' -u3d -o '/home/sense/Desktop/ssp.Particle s2018.1.9f1' SenseShield Virbox [version: 1.5.0.10808] Copyright(c) SenseShield Technology Co., Ltd. All rights reserved.</pre>
protect unity3d Particles2018.1.9f1 Error (13000020): Can not find the license.

Figure 5-45

After the license is verified, the program can be successfully protected by Virbox Protector, as shown in the figure:

sense@sense:~/Desktop/virboxprotector_1.5.0.10808/bin\$./virboxprotector_con '/h
ome/sense/Desktop/Particles2018.1.9f1' -u3d -o '/home/sense/Desktop/ssp.Particle
s2018.1.9f1'
SenseShield Virbox [version: 1.5.0.10808]
Copyright(c) SenseShield Technology Co., Ltd. All rights reserved.
protect unity3d Particles2018.1.9f1 ...
Succeed.

```
Figure 5-46
```

5.4.4 Unity3D android application

5.4.4.1 Protect android application with GUI

5.4.4.1.1 Mono format

For unity3D android application different protection process need to be used for different compile option, mono and IL2CPP.

For Mono:

1. Unzip the Android Unity3D application(apk), the library directory willshow the so library

test	› 桌面 › apktool ›	angrybots5.5.3 > lib > armeabi-v7a	ٽ ~	搜索"armeabi-v7a"
^	名称 ^	修改日期	类型	大小
	📋 libmain.so	2020/5/20 11:43	SO 文件	19 KB
	📄 libmono.so	2020/5/20 11:43	SO 文件	3,675 KB
	📄 libunity.so	2020/5/20 11:43	SO 文件	16,395 KB



2. If the libmono.so library is contained in the /lib directory, means the compile option used is mono, now you need to protect *all of the apk directory* by use of Virbox Protector:

:

Virbox Protector x64 (1.5.1.11511)			-	\times
File Protect Plug-in Log Setting Help				
	5 🛇	\bigotimes		
Open File Save Selected Configuration Save All Con	figuration Protect Selected Projects Prote	ct All Project		
File/Directory	🕞 Basic Info 🛛 🐠 Function Options	Protection Options		
rkvirbox.exe				
Parse successfully!	✓ Output Info			
angrybots5.5.3.apk	Output File/Directory	angrybots5.5.3.ssp.apk		
Protect successfully!	✓ Plug-in			
	✓ Anti Debug Option	\bigcirc		
	Check Debugger			

Figure 5-48

3. Open the "Enable" button in the "Resource Encryption" tag, then you can encrypt the resource file of the Unity3D program.

Virbox Protector 2 (64-bit) v2.0.1.12323		- 0
e Protect Plug-in Log Setting Help		
	- ~	Authorization Info
		Account: 15010303968 Logout
pen File Save Selected Configuration Save All Co	onfiguration Protect Selected Projects Protect All Project	Authorization Detail
File/Directory		
Unity3D	😔 Basic Info 🛛 😨 Function Options 😳 Protection Options 💎 Resource Encryption	
Protect successfully!	Enable (It is recommended that only the resources selected by default be encrypted)	
angrybots5.5.3.apk Parse successfully!		Add File
Parse successfully:	File	^
	assets/bin/Data/59027fb62f4ec427891b09546c07f7fa	×
	assets/bin/Data/ab079e63059f340088f6b9c0e209bee1	×
	assets/bin/Data/c8cc60236143f4ec59b76421156c9eda	×
	assets/bin/Data/e7189a48507d24912932643c3012bf1c	×
	assets/bin/Data/globalgamemanagers	×
	assets/bin/Data/globalgamemanagers.assets	×
	assets/bin/Data/level0	×
	assets/bin/Data/level2	×
	assets/bin/Data/Resources/unity_builtin_extra	×
	assets/bin/Data/sharedassets0.assets	× •
	<	>

4. After protection, a new ssp.apk would be generated, you can re-sign this ssp.apk and pack, to let it can be installed.

Drag the apk of Android Unity3D in to the Virbox Protector, after protection a new file named xxx.ssp.apk will be created, Virbox Protector will protect the **"libmono.so"** and **"Assembly-CSharp.dll"**. Before protection:





Figure 5-49

The un-protected file is named "angrybots5.5.3.apk" After protection:



Figure 5-50

After protection,

"angrybots5.5.3.ssp.apk" is the file after protection. This file can be released to software user in future.

"angrybots5.5.3.apk.ssp" is the configuration file. If you want to protect the data resources you need to use this file or this file can be deleted.

"angrybots5.5.3.apk" is the unprotected file. You can't release this file.

After the application is successfully protected, you need to sign the protected application (.ssp.apk file) and release it.

5.4.4.1.2 IL2CPP

1. Unzip the Android Unity3D apk and check the library directory:

› 桌面 › apktool › android_ila	2cpp → lib → armeabi-v7a	~ Ū	搜索"armeabi-v7a"
名称 ^	修改日期	类型	大小
📄 libdun.so	2020/4/14 15:06	SO 文件	1,734 KB
libil2cpp.so	2020/4/14 15:06	SO 文件	14,219 KB
📄 libmain.so	2020/4/14 15:06	SO 文件	351 KB
📄 libunity.so	2020/4/14 15:06	SO 文件	9,172 KB
📄 libxlua.so	2020/4/14 15:06	SO 文件	1,387 KB



- 2. If the so library libil2cpp.so is contained in the /lib directory, means the application used IL2CPP compile option. You need to protect the *so library* in the lib directory,
- 3. After protection, re-pack the application and sign the signature to the application then release it.

5.4.4.2 Use command line to protect Unity3D apk.

Virbox Protector use a different way from normal apk program to protect the Unity3D program as the particularity of Unity3D program of Android platform.

- 1. Use the Virbox Protector GUI generate ".ssp" configuration file.
- Open the Virbox Command line window, get into the "Virboxprotector_con.exe" directory, and input "virboxprotector_com.exe", you can see the help info;

Command: path of "VirboxProtector_con.exe"+ "the path of the program to be protected" + -u3d –o "the path of the program"

If the license to unity3D program can be found:



If the program can be protected:



5.4.5 Unity3D program call Net dll plugin

- 1. Firstly find the location of **the mono.dll** or **mono-2.0-bdwgc.dll** of the Unity3D compiler:
 - The mono.dll is always in the location: .\Editor\Data\Mono\EmbedRuntime directory



k电脑 → 系统 (D:) → Program Files → U	nity3d > 2018.4.24f1 >	Editor > Data > M	ono > EmbedRuntime	
~ 名称	修改日期	类型	大小	
🗎 mono.bsc	2020/7/6 10:04	Source Browser	5,571 KB	
🖄 mono.dll	2020/7/6 10:04	应用程序扩展	2,567 KB	
MonoPosixHelper.dll	2020/7/6 10:04	应用程序扩展	111 KB	

mono-2.0-bdwgc.dll is always in the location:

.\Editor\Data\MonoBleedingEdge\EmbedRuntime

比电脑 > 系统 (D:) > Program Files > Unity				
unesia / sest (D.) / Programmes / Onity	3d > 2018.4.24f1 >	Editor > Data >	MonoBleedingEdge > Emb	edRuntime
名称 ^	修改日期	英型	大小	
mono-2.0-bdwgc.dll	2020/7/6 10:02	应用程序扩展	4,827 KB	
🗟 mono-2.0-sgen.dll	2020/7/6 10:02	应用程序扩展	5,107 KB	
MonoPosixHelper.dll	2020/7/6 10:02	应用程序扩展	762 KB	

- You can create a fake Unity3D directory, the directory which need to be same with the Unity3D program directory. This is to help Virbox Protector can recognize the program successfully. Here we take the file mono-2.0-bdwgc as an example:
 - Put mono-2.0-bdwgc.dll in the directory bin\MonoBleedingEdge\EmbedRuntime

北电脑 → 本地磁盘(C:) → 用户 → test → 桌面	> > bin		
へ 名称	修改日期	类型	大小
data MonoBleedingEdge	2020/7/8 17:30 2020/7/8 17:30	文件夹 文件夹	

Put the dynamic link library to the location: bin\data\Managed such as(demo.dll)

	组织	新建	打开	选择
此电脑 > 本地磁盘(C:) > 用户	› test › 桌面 › λ	→ bin → dat	a → Managed	
へ 名称	修改日期	类型		大小
Assembly-CSharp.dll	2020/6/	28 17:32 应用	程序扩展	4 KB
demo.dll	2020/7/	8 10:28 应用	程序扩展	4 KB

`

- 3. Drag the bin directory to the Virbox Protector:
 - Set License:



			- [ı ×
			Authorization Info	
			Account: 15010303968 Logou	t
figuration Protect Selected Projects Protect Al	l Project		Authorization Detail	
🕞 Basic Info 🛛 🚱 Function Options	Protection Options	Resource Encryption		
✓ Assembly			+	
jiamigou_Test_Data/Managed/Assemb	ly-CSharp-firstpass.dll			×
ijamigou Test Data/Managed/Assemb	lv-CSharp.dll			×
✓ Output Info				
Output File/Directory	ssp.Unity3D		•••	
✓ Plug-in				
 Anti Debug Option 	\bigcirc			
Detecting Hardware Breakpoints				
Detecting Memory Breakpoints				
Memory Integrity Check				
		figuration Protect Selected Projects Protect All Project Basic Info Function Options Protection Options Assembly imaged/Assembly-CSharp-firstpass.dll jamigou_Test_Data/Managed/Assembly-CSharp-firstpass.dll jamigou_Test_Data/Managed/Assembly-CSharp.dll Output Info Output Fie/Directory ssp.Unity3D Plug-in	figuration Protect Selected Projects Protect All Project Basic Info Function Options Protection Options Resource Encryption Assembly jamigou_Test_Data/Managed/Assembly-CSharp-firstpass.dll jamigou_Test_Data/Managed/Assembly-CSharp.dll Output File/Directory ssp.Unity3D V Plug-in Output File/Directory Detecting Hardware Breakpoints Detecting Memory Breakpoints	Authorization Info Account: 15010303968 Logou Authorization Detail Basic Info Function Options Basic Info Function Options Protection Options Resource Encryption Assembly + jamigou_Test_Data/Managed/Assembly-CSharp-firstpass.dll + jamigou_Test_Data/Managed/Assembly-CSharp.dll • Output File/Directory ssp.Unity3D Plug-in - Anti Debug Option - Detecting Hardware Breakpoints - Detecting Memory Breakpoints -

• Add the program set in the "protection option":

Open File Save Selected Configuration Save All Configuration Protect :	Sected Projects Protect All Project	Authorization Info Authorized Account Login or Pluin Dongle
File/Directory PE_Tetris.exe Parse successfully! bin Parse successfully!	Basic Info Function Options Protection Options Assembly Assembly-CSharp.dll data/Managed/Assembly-CSharp.dll Output Info	Resource Encryption
		× v ð Search bin p
	Organize New folder	Date modified Type 10/23/2020 3:42 PM File folder
	Desktop MonoBleedingEdge Assembly-CSharp.dll Documents Downloads	10/23/2020 3:41 PM File folder 9/20/2016 5:43 PM Application extens 9/20/2016 5:43 PM Application extens

 After the program set is added successfully, delete the non-necessary program and reserve your dll.



Virbox Protector 2 (64-bit) v2.0.1.12323 e Protect Plug-in Log Setting Help File Save Selected Configuration Save All Configuration	iguration Protect Selected Projects Protect All Project	- X Authorization Info Account: 15010303968 Logout Authorization Detail
File/Directory	Basic Info Function Options Protection Options Resource Encryption Assembly jiamigou_Test_Data/Managed/Assembly-CSharp-firstpass.dll jiamigou_Test_Data/Managed/Assembly-CSharp.dll Output Info Output Info Output File/Directory ssp.Unity3D Plog-in	+ × ×
	Anti Debug Option Detecting Hardware Breakpoints Detecting Memory Breakpoints Memory Integrity Check	

Click "*Protect the selected protect*" to protect the program, the ssp.bin would be generated.
 This is the program after protection.

を内走 履	郇游戏													_
Shar	e Vi	ew												
Paste	🔏 Cut 🐜 Cop 💽 Pas		Move to •	Copy to •		■ Rename	New folder	Rew item ▼ T Easy access	Properties	Fdit 🖓 History	Select Select	none	n	
ipboard	1			Org	anize			New	C	Open	Sel	ect		
> T	his PC	> Local Disk	: (E:) > T	Test-Pro	gram →	Unity3D	> 室内;	走廊游戏				~	ē	Search 室内走廊游戏
	^	Name		^			Date	modified	Туре	Siz	e			
		hin					10/2	3/2020 3·42 PM	File folder					
		📙 ssp.bi	in				10/2	3/2020 3:46 PM	File folder					
		Unity:	3D				10/2	3/2020 3:38 PM	File folder					
		室内注	走廊游戏				10/2	0/2020 11:45	File folder					
		📄 bin.ss	p				10/2	3/2020 3:46 PM	SSP File		1 KB			
	ł													

- 4. Get into the ssp.bin directory, copy the
 - "mono-2.0-bdwgc.dll"

in the *ssp.bin\MonoBleedingEdge\EmbedRuntime*

directlry to "*Editor\Data\MonoBleedingEdge\EmbedRuntime*" then put the "*demo.dll*" in the directlry "*ssp.bin\data\Managed*" into project.

In this way to protect the plugin you want to protect.

5.4.6 Protection Comparison

Assembly-CSharp*.dll script file has been protected by using Virbox Protector. Before protection/encryption:



	_ 🗆 ×
帮助(H) 😋 ◎ 🚔 📽 C# 🚽 🤈 💎 🕨 启动 🔎	
Start() : void ×	
<pre>1 // Task 2 // Token 0x060000001 RID: 1 RVA: 0x00000250 File Offset: 0x000000250 3 private void Start() 4 5 thistxt.text = ""; 6 thisbtm.onClick.AddListener(delegate() 7 { 8 } Application.Quit(); 9 } thistxt.text = "GameOver"; 10 }); 11 12 12 13 14 15 15 15 15 15 15 15 15 15 15 15 15 15</pre>	
100 % + 4	F
局部变量	→ ×
	类型
	Start():void × 1 // Task: 2 // Token: 0x0000001 RID: 1 RVA: 0x00002050 File Offset: 0x00000250 3 private void Start() 4 { 5 ithistxt.text = ""; 6 thisbtn.onClick.AddListener(delegate()) 7 { 9 thistxt.text = "GameOver"; 10 }); 12 100 % -

Figure 5-52

After protection/encryption:



Figure 5-53

Protection comparison (Using the de-compiler tool: AssetStudio to decompile the resource file with and w/o protected by DS Protector.) Before protection:



File Options Model Export Filte	r Туре		
cene Hierarchy Asset List			Shader "Hidden/Internal-DeferredReflections" {
Filter			Properties {
	1		SrcBlend ("", Float) = 1
Name	Туре	Size ^	_DstBlend ("", Float) = 1
StandaloneInputModule	MonoBehaviour	96	SubShader {
StandaloneInputModule #554	MonoBehaviour	96	Pass {
Text	MonoBehaviour	264	ZWrite Off
Text #526	MonoBehaviour	268	GpuProgramID 19311
Text #555	MonoBehaviour	268	Program "vp" {
Timer	MonoBehaviour		SubProgram "d3d11 " {
WinDetector	MonoBehaviour		"//
GUI_Text Shader	Shader	16600	// Generated by Microsoft (R) D3D Shader Disassembler
Hidden_BlitCopy	Shader	3312	1/
Hidden_BlitCopyDepth	Shader	3136	11
Hidden_BlitCopyWithDepth	Shader	3416	// Input signature:
Hidden_BlitToDepth	Shader	3280	1/
Hidden_BlitToDepth_MSAA	Shader	3336	// Name Index Mask Register SysValue Format Used
Hidden_Compositing	Shader	3556	
Hidden_ConvertTexture	Shader	3396	// POSITION 0 xyzw 0 NONE float xyz // NORMAL 0 xyz 1 NONE float xyz
Hidden_CubeBlend	Shader	6184	// NORFAC 6 Xy2 I NONE FIGHT Xy2
Hidden_CubeBlur	Shader	7860	
Hidden_CubeCopy	Shader	5540	// Output signature:
Hidden_FrameDebuggerRenderTargetDisplay	Shader	17668	// orphe signaturer
Hidden_InternalClear	Shader	57568	// Name Index Mask Register SysValue Format Used
Hidden_Internal=Colored	Shader	15052	1//
Hidden_Internal=CombineDepthNormals	Shader	3788	// SV_POSITION 0 xyzw 0 POS float xyzw
Hidden_Internal=CubemapToEquirect	Shader	3236	// TEXCOORD 0 xyzw 1 NONE float xyzw
Hidden_Internal=DeferredReflections	Shader	8624	// TEXCOORD 1 xyz 2 NONE float xyz
Hidden_Internal=DeferredShading	Shader	33552	1/
Hidden_Internal=DepthNormalsTexture	Shader	32904	vs_4_0
Hidden_InternalErrorShader	Shader	13704	dcl_constantbuffer CB0[3], immediateIndexed
Hidden_Internal=Flare	Shader	3236	dcl_constantbuffer CB1[6], immediateIndexed
Hidden_Internal=GUIRoundedRect	Shader	7652	<pre>dcl_constantbuffer CB2[4], immediateIndexed dcl_constantbuffer CB3[21], immediateIndexed</pre>
Hidden_Internal-GUITexture	Shader	5696	dcl_constantbutter CBS[21], immediateindexed
Hidden_Internal=GUITextureBlit	Shader	6348	dcl input v1.xyz
Hidden_Internal-GUITextureClip	Shader	6352	dcl output siv o0.xyzw, position
Hidden_Internal-GUITextureClipText	Shader	6176	dcl output o1.xyzw
Hidden_Internal-Halo Hidden Internal-MotionVectors	Shader	3244	

Figure 5-54

After Protection:

AssetStudioGUI - no productName - 2018.2	4f1 - StandaloneWindows64	
File Options Model Export Filter Ty	e	
cene Hierarchy Asset List	PPt	r <gameobject> m_GameObject</gameobject>
Filter		<pre>int m_FileID = 0</pre>
Name	Type S. UIn	int64 m_PathID = 6 t8 m Enabled = 1
	7.4 DD4	r <monoscript> m_Script</monoscript>
MonoBehaviour #202	monopenaviour oo	int m FileID = 1
MonoBehaviour #201	MonoBehaviour 268	int64 m_PathID = 274
MonoBehaviour #200 MonoBehaviour #199	MonoBehaviour 96 MonoBehaviour 76	ing m_Name = ""
MonoBehaviour #199 MonoBehaviour #198	MonoBehaviour 76 MonoBehaviour 44	
MonoBehaviour #190 MonoBehaviour #197	MonoBehaviour 52	
MonoBehaviour #174	MonoBehaviour 32	
MonoBehaviour #173	MonoBehaviour 64	
MonoBehaviour #172	MonoBehaviour 268	
MonoBehaviour #171	MonoBehaviour 44	
MonoBehaviour #170	MonoBehaviour 40	
MonoBehaviour #169	MonoBehaviour 44	
MonoBehaviour #168	MonoBehaviour 32	
MonoBehaviour #167	MonoBehaviour 76	
MonoBehaviour #166	MonoBehaviour 44	
MonoBehaviour #165	MonoBehaviour 96	
MonoBehaviour #164	MonoBehaviour 40	
MonoBehaviour #163	MonoBehaviour 36	
MonoBehaviour #162	MonoBehaviour 44	
MonoBehaviour #161	MonoBehaviour 264	
MonoBehaviour #160	MonoBehaviour 40	
MonoBehaviour #159	MonoBehaviour 52	
	<	

Figure 5-55



5.5 Protect Android application

5.5.1 Normal apk application

For normal apk application, the application need to be unzipped and then to protect the so library in lib directory by use of Virbox Protector, such as:

> apktool > android_toutiao776 >	lib⇒ armeabi-v7a	~ Ū	搜索"armeabi-v7a"
名称 ^	修改日期	类型	大小
📄 libad.so	2020/5/13 15:10	SO 文件	668 KB
libBugly.so	2020/5/13 15:10	SO 文件	437 KB
libgetuiext3.so	2020/5/13 15:10	SO 文件	1,060 KB
libInnoSecure.so	2020/5/13 15:10	SO 文件	442 KB
libInnoSo.so	2020/5/13 15:10	SO 文件	1,849 KB
libNativeExample.so	2020/5/13 15:10	SO 文件	528 KB
libpl_droidsonroids_gif.so	2020/5/13 15:10	SO 文件	322 KB
📄 libsgmain.so	2020/5/8 16:04	SO 文件	382 KB

Figure 5-56

After protection, you need to pack all of files with signature to the apk application, then release the application.

5.6 Protect the python based application

Parser protection

 Python.exe (interpreter) file based on python protection, the detail steps are same with Windows Application, please refer the steps above. Chapter 5.1, use the default setting to encrypt the Python.exe

Function Options Setting:



Virbox Protector 2 (64-bit) v2.0.1.12323			– 🗆 X
File Protect Plug-in Log Setting Help			
Open File Save Selected Configuration Save All Con	figuration Protect Selected Projects Prote	ect All Project	Authorization Info Account: 1 ¹ ¹ 68 Logout Authorization Detail
File/Directory python.exe Parse successfully!	Basic Info Function Options Total Qty: 53 Total Protected:1 Obfuscation:0 Virtualization: 0 10001334	C Protection Options Encrypted: 1	Add Functions

Figure 5-57

Protection Options Setting:

Virbox Protector 2 (64-bit) v2.0.1.12323			– 🗆 X
File Protect Plug-in Log Setting Help			
Open File Save Selected Configuration Save All Con	figuration Protect Selected Projects Protect A	J All Project	Authorization Info Account: 15010303968 Logout Authorization Detail
File/Directory	🕞 Basic Info 🛛 🚱 Function Options	Protection Options	
python.exe Parse successfully!	✓ Output Info		^
	Output File/Directory	python.ssp.exe	•••
	 Protection Options 		
	Import Table Protection		0
	Compression		
	Memory Check		
	Resource Section Encryption		
	✓ Plug-in		
	✓ Anti Debug Option	\bigcirc	
	Detecting Hardware Breakpoints		
	Detecting Memory Breakpoints		
	Memory Integrity Check		
	✓ ds		Open DSProtector
	Paceword	·····	*
C:/Users/luyongqiang/Desktop/python.exe			

Figure 5-58

Resource Protection

For resources protection, protect the python.exe with Virbox Protector first, and then protect the .pyc,
 pyd and .py file with DSProtector.

In this example, after protect the relevant file with above process, you will get 3 file

"python.exe.ssp" is the configuration file, and when you are protecting the .py and .pyc file, you will need this Copyright © 2020, Virbox, All Right Reserved. Page 103 | 108



file.

"ssp.python.exe" is the python.exe file after protection, you need to use this file to parse the protected .py and .pyc file. (The .py and .pyc file need to run with the ssp.python.exe file). When you run the protected .py and .pyc file.

Please modify the **python.ssp.exe** to be **python.exe**, in order not to influence the existed python environment. "**python.exe**" is the file before protection.

python-test			✓ ♂ Search pythor
Name	Date modified Type	Size	
python.exe	10/25/2018 4:14 PM Application	99 KB	
python.exe.ssp	1/21/2019 11:55 AM SSP File	1 KB	
👦 python.ssp.exe	1/21/2019 11:55 AM Application	772 KB	

Figure 5-59

Open the DSProtector for .pcy and .py file protection,

_					
v		V	DSProtect	or(1.3.1.11480)	
PC	> Local Disk (E:) > Test-Program > python	File	Setting	Heln	
~	Name		Jetting.	, nep	
	Name Di	SSP	File:	E:/Test-Program/python/python.exe.ssp	
	New folder 5/	File	Filter:	Filter files when add directory, eg:txt;php;html;	
	get_license_example-bak - Copy.pyc 4/				
	get_license_example-bak.py 4/	Out	tput Directo	ry: Optional; we will always bakup(*bak) in the same directory	
	get_license_example-encryption.py 5/		File	1	Output File
	n python.exe 12		🗹 get	_license_example-bak - Copy.pyc	E:/Test-Program/python/get_license_example-bak
	python.exe.ssp 5/				
	python.ssp.exe 5/				
		<			
			Add File	Add Directory Remove Selected Remove All	

Figure 5-60

Choose the ssp configuration file created last step when you are protecting "python.exe"

Drag in the .py and .pyc file,

Click "Protect", it is showing "protect success"

The protected file name will use the original name, and the unprotected file will be with bak file name extension.

Till now the protection of **python resources** file has been completed, and you can release the protected file to Copyright © 2020, Virbox, All Right Reserved. Page 104 | 108



the customer.

6 Note

Т	уре	Unsupported scenario
		You can't use Virbox Protector to protect the software which protected by
		third party protector or wrap tools, any protected software can't be
		protected by Virbox Protector or third party Wrap tools to protect again.
Other		
		The Jar archive is not supported to be protected by Virbox Protector, The Jar
		archive can be protected by DS Protector
		The program with Self verify function can't be protected by Virbox Protector
	.NET	The program with StrongName signed can't be protected by Virbox
		Protector
		Third party runtime library is not supported to be protected by Virbox
		Protector, only Micro soft standard runtime library can be protected
		SDK label doesn't support for .Net program
		If .Net program developed by C# language contains the method that called
		by external program or contain public method, name obfuscation can't be
		used to encrypt these kind of method. Because the name obfuscation may
		change the method name and cause some of functions may not be used.
		Compression will modify the program type from AnyCPU to be PE32. So if
		you compress .Net AnyCPU program. It would be no longer to be called by
		other program.
File Type		Compression function do not support .NET dynamic link library
тпе туре		Resources protection do not support the executable converted from
		Powerpoint
		Resources protection do not supported for the program of VB6.0 language
	PE	Import Table: The symbol imported must be function and can't be import
		variable, or the program would crash when started.
		If the protected program run with load memory type, it can't be started if
		compression is used.
	ELF	Additional data is not supported for Linux program currently.
		ELF program which is compiled by -static compile option is not supported to
		be protected by Virbox Protector.
		Map file analysis is not support ELF format file.
		If all of the symbol in the program is exported, crash may happen when you
		run the program. Only required function suggested to be exported.
Protection	Code	The function parsed by Virbox protector without function name may not



Option	Encryption	supported by Virbox Protector, because of external function entry may
		exist.
		For the function instruction which too less may not be protected by Virbox
		Protector
		For ELF and Mach-O program which compiled with C or C++, the program
		may not be use obfuscation and Virtualization to protect. Because of stack
		frame issue. For such case, you may use the " code encryption " option to
	Obfuscation/	protect the application.
	Virtualization	For the function's instruction is too less, it may not be protected.
		Virtualization and code snippet is not supported for ARM architecture
		program.

6.1 Known Issues

- The protected .Net program only support Microsoft standard running lib, do not support third party running lib
- When using the command line to protect your software, the configuration file of the objective file must be exist.
- When GetField("name", bindingAttr) used in the .Net program to be protected, and if you select the "Name obfuscation" in "Function Option", the software may fail in execution, and you need to remove the obfuscation from the" Function option".
- You may fail to protect the software with code snippet, because of too less of instruction of the snippet code, maybe jump, and it can't be code ported.
- The name of the software after protection will be changed, please modify it to be the original name.
 The ARX plugin of AutoCAD can only select "remote desk service dialog message box", and now only support win7 and server2008 or above version.
- Anti-virus AVAST may cause the start failed of the protected program, it will kill the thread of the protected software when it executed.
- Program with strong signature is not supported



7.1 What is the difference between the soft license edition and dongle edition?

For Virbox Protector with soft license: the license allowed to be bind with one computer only and you may change the device up to 2 times.

For Virbox Protector Dongle based license: In addition to the software, you will also get a dongle that stored license of Virbox Protector. Any computer plugged dongle can use Virbox Protector.

7.2 What is the difference between the trial edition and standard edition?

For a Trial Edition,

- The protected software program will valid within 7 days for your internal testing and evaluation. after 7 days, when you run the protected software it will have the message pop up:" **This application is protected with trial version of Virbox Protector**". The license of trial edition Virbox Protector will be expired after 30 days or 100 times usage. For standard edition, you can protect your functions without above limitation.
- Trial edition Virbox Protector can be used to test and protect the program for: Windows, Linux, Mac, ARM Linux, Android. For standard edition, you need to purchase the corresponding license with your program you are going to protect.

No matter which software area you come from, we have experts who understand the special challenges you are facing in your industry. We will help you solve those problem with what we have. And we have helped thousands of software enterprises from different industry to Protected the software and helped them realized software monetization. And we have established special Internet sales model and deeper customer relationships with our customer. We can also do this for you.