



API Redirect on Themida

부경대학교 정보보호동아리 CERT-IS

김연재 / Yeonjae Kim

e-mail : 6l4ck3y3@gmail.com

twitter : @6l4ck3y3

순서

Themida 소개

API 리다이렉트 분석

스크립트 작성

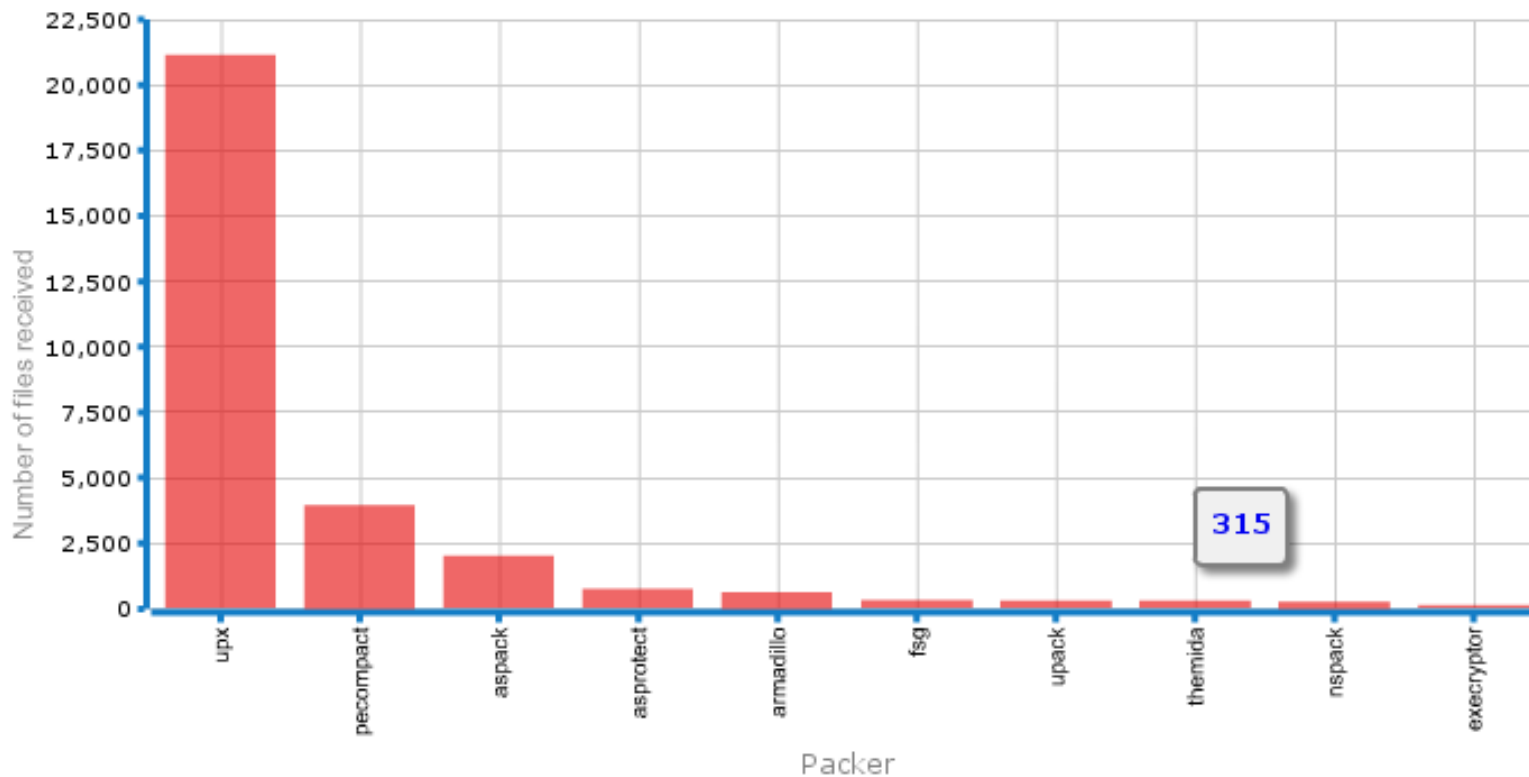
Themida 소개

Oreans
Technology

상용 프로텍터

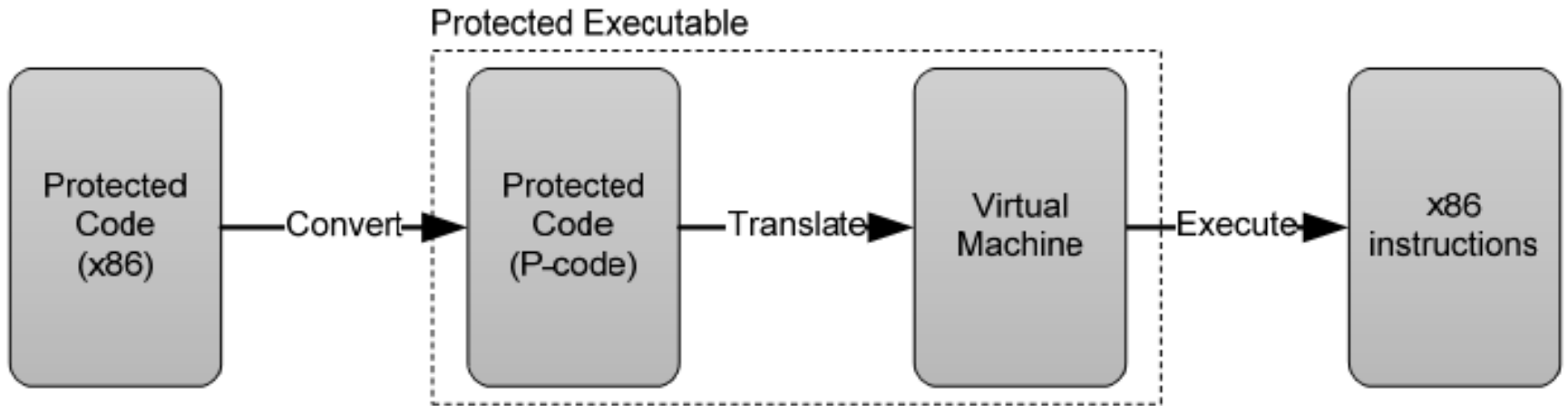


149 € = 225,747 원



<http://www.virustotal.com/stats.html>

코드 가상화



The Art of Unpacking

Themida v2.0.4.0 [Demo] | Project: +UNSAVED+

THEMIDA

Advanced Windows Software Protection System

New Open... Save Save As... Protect Help About... Demo Version

Options

- Application Information
- Protection Options
- Code Replace
- Virtual Machine
- Customized Dialogs
- Advanced Options
- XBundler

Help

- Code Replace
- SecureEngine Macros

Microsoft .NET compatible 2.0.4.0

Code Replace

Enable CodeReplace Technology

Functions to remove using the CodeReplace Technology:

- Function: 0 --- Start: 0x00401420 --- Stop: 0x00401427 --- Executions: 1
- Function: 1 --- Start: 0x00401260 --- Stop: 0x004012C6 --- Executions: 1

Buttons: Select functions from MAP file, Read functions automatically

Assembly code inside function: Force simulation before protection

```

000401260h: push  ebp
000401261h: mov   ecx, dword ptr [000405118h]
000401267h: mov   ebp, esp
000401269h: pop   ebp
00040126ah: jmp   ecx
00040126ch: lea  esi, [esi+0000000000h]
000401270h: push  ebp

```

Run with parameters

Start Simulation

API 리다이렉트

Stolen instructions from kernel32!CopyFileA

```
00D80003 MOV EDI,EDI
00D80005 PUSH EBP
00D80006 MOV EBP,ESP
00D80008 PUSH ECX
00D80009 PUSH ECX
00D8000A PUSH ESI
00D8000B PUSH DWORD PTR SS:[EBP+8]
00D8000E JMP SHORT 00D80013
00D80011 INT 20
00D80013 PUSH 7C830063 ;return EIP
00D80018 MOV EDI,EDI
00D8001A PUSH EBP
00D8001B MOV EBP,ESP
00D8001D PUSH ECX
00D8001E PUSH ECX
00D8001F PUSH ESI
00D80020 MOV EAX,DWORD PTR FS:[18]
00D80026 PUSH DWORD PTR SS:[EBP+8]
00D80029 LEA ESI,DWORD PTR DS:[EAX+BF8]
00D8002F LEA EAX,DWORD PTR SS:[EBP-8]
00D80032 PUSH EAX
00D80033 PUSH 7C80E2BF
00D80038 RETN
```

Actual kernel32!CopyFileA code

```
7C830053 MOV EDI,EDI
7C830055 PUSH EBP
7C830056 MOV EBP,ESP
7C830058 PUSH ECX
7C830059 PUSH ECX
7C83005A PUSH ESI
7C83005B PUSH DWORD PTR SS:[EBP+8]
7C83005E CALL kernel32.7C80E2A4
7C830063 MOV ESI,EAX
7C830065 TEST ESI,ESI
7C830067 JE SHORT kernel32.7C8300A6
```

The Art of Unpacking

Themida v2.0.4.0 [Demo] | Project: *UNSAVED*

THEMIDA

Advanced Windows Software Protection System

New Open... Save Save As... Protect Help About... Demo Version

Options

- Application Information
- Protection Options
- Code Replace
- Virtual Machine
- Customized Dialogs
- Advanced Options
- XBundler

Help

- Protection Options
- Protect Now
- SecureEngine Technology

Microsoft .NET compatible

2.0.4.0

Protection Options

Protection Options

Anti-Debugger Detection Advanced

Advanced API-Wrapping Level 1

- Disable
- Level 1
- Level 2

Anti-Patch None

Anti Dumpers Enable Protection

Entry Point Obfuscation Enable Protection

Resources Encryption Enable Encryption

VMWare/Virtual PC Compatible

Metamorph Security Enable Protection

Memory Guard Enable Protection

When Debugger Found Display Message

Compression

- Application
- Resources
- SecureEngine

Monitor Blockers

- Files Monitors
- Registry Monitors

Delphi/BCB Form Protection Enable Protection

API 리다이렉트 분석

정상적인 루틴과 비교

```

0040128C 90      NOP
0040128D 90      NOP
0040128E 90      NOP
0040128F 90      NOP
00401290 55      PUSH EBP
00401291 . 89E5   MOV EBP,ESP
00401293 . 83EC 18 SUB ESP,18
00401296 . C74424 0C 2000 MOV DWORD PTR SS:[ESP+C],20
0040129E . C74424 08 0030 MOV DWORD PTR SS:[ESP+8],ThemidaE.004030
004012A6 . C74424 04 1030 MOV DWORD PTR SS:[ESP+4],ThemidaE.004030
004012AE . C70424 00000000 MOV DWORD PTR SS:[ESP],0
004012B5 . E8 B6060000 CALL <JMP.&USER32.MessageBoxA>
004012BA . 83EC 10 SUB ESP,10
004012BD . B8 00000000 MOV EAX,0
004012C2 . C9     LEAVE
004012C3 . C2 1000 RETN 10
004012C6 90      NOP
004012C7 90      NOP
004012C8 90      NOP
004012C9 90      NOP
004012CA 90      NOP
004012CB 90      NOP

```

```

0040128C 90      NOP
0040128D 90      NOP
0040128E 90      NOP
0040128F 90      NOP
00401290 55      PUSH EBP
00401291 89E5   MOV EBP,ESP
00401293 83EC 18 SUB ESP,18
00401296 C74424 0C 200000 MOV DWORD PTR SS:[ESP+C],20
0040129E C74424 08 003040 MOV DWORD PTR SS:[ESP+8],ThemidaE.004030
004012A6 C74424 04 103040 MOV DWORD PTR SS:[ESP+4],ThemidaE.004030
004012AE C70424 00000000 MOV DWORD PTR SS:[ESP],0
004012B5 E8 B6060000 CALL ThemidaE.00401970
004012BA 83EC 10 SUB ESP,10
004012BD B8 00000000 MOV EAX,0
004012C2 C9     LEAVE
004012C3 C2 1000 RETN 10
004012C6 90      NOP
004012C7 90      NOP
004012C8 90      NOP
004012C9 90      NOP
004012CA 90      NOP
004012CB 90      NOP

```

리다이렉트

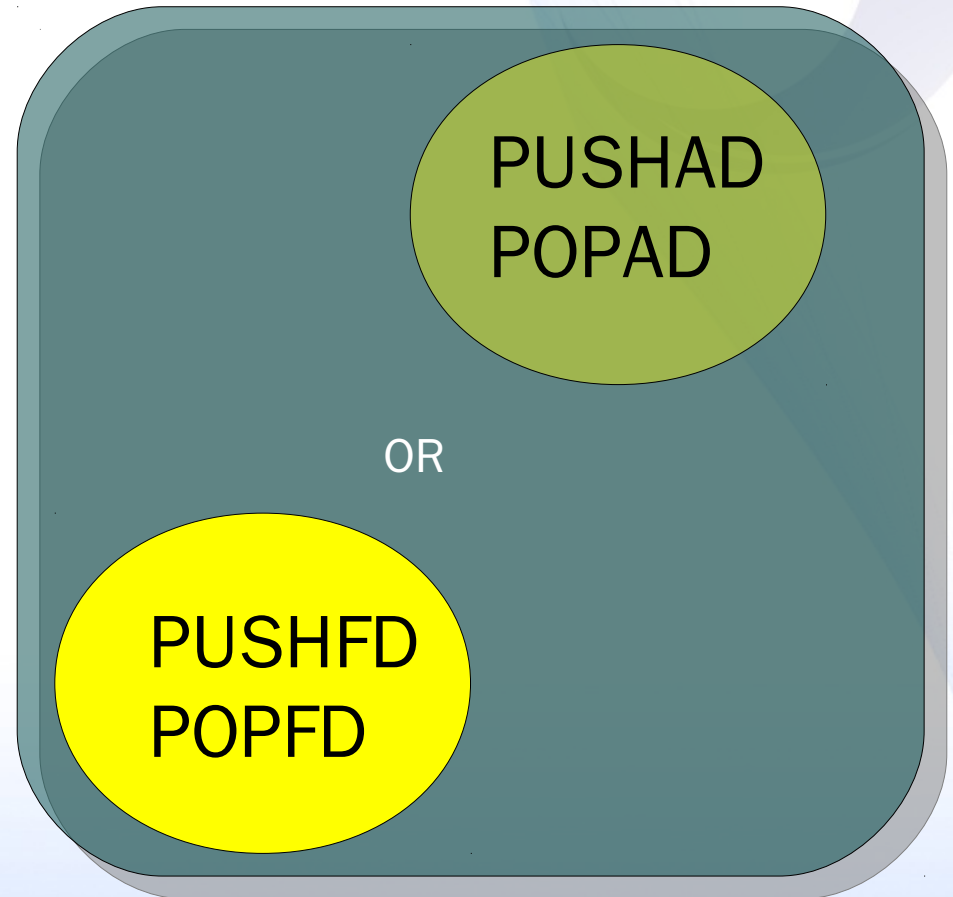
Demonstration

난독화

패턴 1

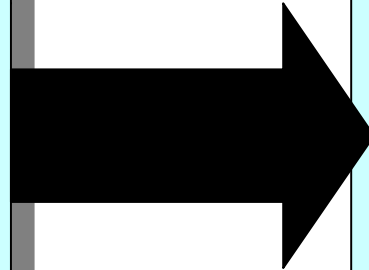
PUSH EAX
PUSH EDX
RDTSC
POP EDX
POP EAX

패턴 2



단점

```
NOP  
NOP  
NOP  
NOP  
JMP 044704CF  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
JMP 044704E5  
NOP  
NOP
```



```
PUSH EAX  
PUSH EDX  
PUSH EAX  
PUSH EDX  
JMP 044704CF  
RDTSC  
PUSHAD  
POPAD  
POP EDX  
POP EAX  
RDTSC  
JMP 044704E5  
POP EDX  
POP EAX
```

스크립트 작성

revealapi.py

```
from immllib import *
```

```
def main (args):
```

```
    imm = Debugger ()
```

```
    eip = orig_eip = imm.getRegs ()['EIP']
```

revealapi.py

```
while (eip < 0x10000000 and limit):
    opcode = imm.Disasm (eip)

    if (opcode.isJump () or opcode.isConditionalJump () or opcode.isCall ()):
        eip = opcode.getJumpAddr ()

    elif opcode.isRet ():
        return "API was NOT found. Maybe here is a local func."

    else:
        eip = eip + int (opcode.getSize ())
```

revealapi.py

Demonstration



THANK YOU !

「The Art of Unpacking」 by Mark Vincent Yason

「Design and Implementation of Virtualized Code Protection(VCP) For Anti-Reverse Engineering」 by 이용일

「GRAY HAT PYTHON」 by Justin Seitz

우리의 도전이 바로 보안입니다.