





### Data flow analysis in order to construct control flow graphs of obfuscated x86 binary codes

Jean-Yves MARION - Université de Lorraine, Loria Sylvain CECCHETTO - Cyber-Detect, Nancy

**France-Japan Cybersecurity workshop** February 25 and 26, 2021







### Data flow analysis in order to construct control flow graphs of obfuscated x86 binary codes

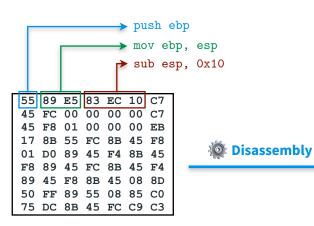
Jean-Yves MARION - Université de Lorraine, Loria Sylvain CECCHETTO - Cyber-Detect, Nancy

**France-Japan Cybersecurity workshop** February 25 and 26, 2021

## Introduction

#### **Binary analysis** Disassembly and control flow graph

[Schwarz, Debray and Andrews 2002], [Shoshitaishvili, Wang, Salls, Stephens, Polino, Dutcher, Grosen, Feng, Hauser, Kruegel and Vigna 2016] [Biondi, Rigo, Zennou and Mehrenberger 2017]



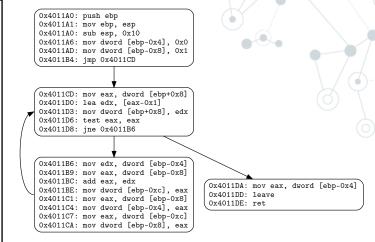
#### **Problems**

- Variable length instructions
- Mixed data in code bytes
- Indirect jumps
- ) ...

```
push ebp
mov ebp, esp
sub esp, 0x10
mov dword [ebp-0x4], 0x0
mov dword [ebp-0x8], 0x1
jmp 0x2d
mov edx, dword [ebp-0x4]
mov eax, dword [ebp-0x8]
add eax, edx
mov dword [ebp-0xc], eax
mov eax, dword [ebp-0x8]
mov dword [ebp-0x4], eax
mov eax, dword [ebp-0xc]
mov dword [ebp-0x8], eax
mov eax, dword [ebp+0x8]
lea edx, [eax-0x1]
mov dword [ebp+0x8], edx
test eax, eax
jne 0x16
mov eax, dword [ebp-0x4]
leave
ret
```

#### Example : disassembly of EB0168C3909090 code

0x0:		jmp	
0x2:	68C3909090	push	0x909090c3



0x0:	EB01	jmp 0x3
0x2:	68	;data
0x3:	C3	ret
0x4:	90	nop
0x5:	90	nop
0x6:	90	nop

# Obfuscation

[Collberg, Thomborson and Low 1997]

# Transforming a program P into a program P' so that P' works like P but is harder to understand



Build the control flow graph of a binary

0

Build the control flow graph of a binary

Malwares are protected (obfuscations)

### Build the control flow graph of a binary

Malwares are protected (obfuscations)

# Build the **control flow graph** of a **malware**

# **BOA - platform**

2

#### **Goals** Disassemble and build CFG of obfuscated binaries

EXE

#### O Main goal:

> Disassemble and build CFG of obfuscated binaries

BOA

- Others goals:
  - Detect obfuscations and doubtful behaviors
  - Create report

≣



### An approach combining the **advantages** of a **static analysis** with the **strengths** of a **dynamic analysis**?



# An approach combining the **advantages** of a **static analysis** with the **strengths** of a **dynamic analysis**?

# Symbolic execution... ... obfuscation-resistant

#### Machine state 🙆

Given by the triplet (ip,  $\sigma$ , p), with:

#### ip Instruction pointer

 $\sigma: \begin{cases} r \in \operatorname{Reg} \mapsto v \in \operatorname{Addr} \cup \{ \perp \} \\ f \in \operatorname{Flag} \mapsto v \in \{0, 1, \perp\} \\ a \in \operatorname{Addr} \mapsto v \in \operatorname{Bytes} \cup \{ \perp \} \end{cases}$  Value of CPU, flags and memory cells

#### $\perp$ : unknown value

 $p: a \in Addr \mapsto p \in \{\emptyset, R, RX, RW, RWX\}$  Memory cells permissions

Normal mode, when  $\sigma[ip]^*$ Execution 🕉 can be executed  $(ip, \sigma, p) \to (ip', \sigma', p') =_{def} \begin{cases} (ip, \sigma, p) \xrightarrow{\sigma[ip]^*} (ip', \sigma', p') \\ (ip, \sigma, p) \xrightarrow{\mathscr{O}} \mathscr{O}(ip, \sigma, p) \end{cases}$ Self-modification support : Instructions live in  $\sigma$ Else, kernel mode  $\sigma$  can be modified by  $\sigma$ [ip]\*

#### BOA In a nutshell

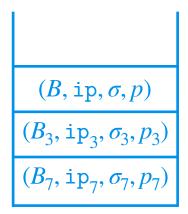
#### Data flow analysis to rescue in order to build the control flow graph of obfuscated binaries

- O Analysis approach: recursive disassembling + symbolic execution at basic block level
- Resolve indirect jumps by partial machine state computation

#### Multiple obfuscation support:

- Self-modification
- Exceptions
- Indirect jumps and call stack tampering
- Opaque predicate and dead branches
- On the fly import table construction

while  $\mathscr{A}$ :

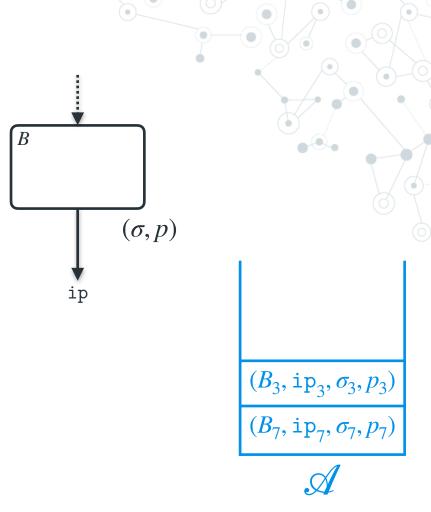


Ø



while  $\mathscr{A}$ :

// Retrieve a 4-uplet to be processed |  $(B, ip, \sigma, p) = \mathcal{A}.pop()$ 





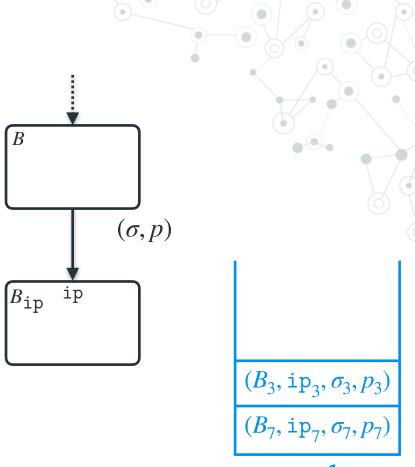
while  $\mathscr{A}$ :

// Retrieve a 4-uplet to be processed

$$| \quad (B, ip, \sigma, p) = \mathscr{A}.pop()$$

// Disassemble ip basic block

$$| B_{ip} = DisasBasicBlock(ip)$$





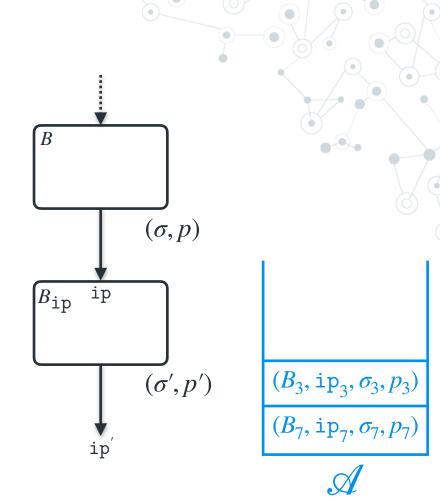
while  $\mathscr{A}$ :

// Retrieve a 4-uplet to be processed

 $(B, \text{ip}, \sigma, p) = \mathscr{A}. \text{pop}()$ 

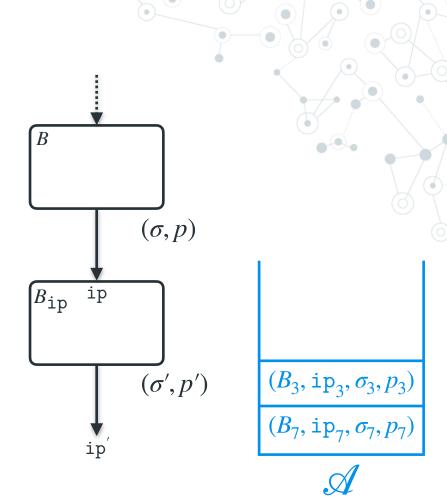
// Disassemble ip basic block

 $\begin{array}{l} B_{\texttt{ip}} = \texttt{DisasBasicBlock(ip)} \\ \texttt{// Apply symbolic execution of } B_{\texttt{ip}} \texttt{ in } \sigma \texttt{ and } p \\ \texttt{(ip}', \sigma', p') = \texttt{MachineState}(B_{\texttt{ip}}, \sigma, p) \end{array}$ 



while  $\mathscr{A}$ :

// Retrieve a 4-uplet to be processed  $(B, ip, \sigma, p) = \mathscr{A} \cdot pop()$ // Disassemble ip basic block  $B_{ip}$  = DisasBasicBlock(ip) // Apply symbolic execution of  $B_{
m ip}$  in  $\sigma$  and p $(ip', \sigma', p') = MachineState(B_{ip}, \sigma, p)$ // Add new 4-uplet in  $\mathscr{A}$ if  $ip' \neq \bot$ :  $\mathscr{A}.push((B_{ip}, ip', \sigma', p'))$ else if  $type(B_{ip}) == COND$ :  $\mathcal{A}$ .push(( $B_{ip}$ , next\_mem,  $\sigma'$ , p'))  $\mathcal{A}$ .push(( $B_{ip}, target, \sigma', p'$ ))



# **BOA - applications**

3

### **Self-modifications**

### $\odot$ Two goals

- Detect self-modification
- Continue self-modification



#### **BOA** Self modification and execution waves

- **Detect**: [Bonfante, Fernandez, Marion, Rouxel, Sabatier et Thierry 2015]
  - Keep a list of modified memory cells
  - Check every instruction before symbolic execution
- O Handle :
  - Concept of execution waves
  - Each instruction is associated to a wave



#### **BOA** Self modification and execution waves

- **Detect**: [Bonfante, Fernandez, Marion, Rouxel, Sabatier et Thierry 2015]
  - Keep a list of modified memory cells
  - Check every instruction before symbolic execution
- O Handle :
  - Concept of execution waves
  - Each instruction is associated to a wave

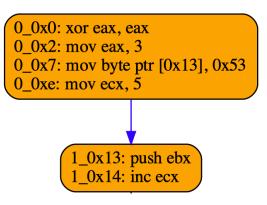


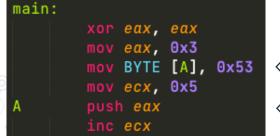
⇐ self-modification (@[A] replaced by 0x53)

 $\Leftrightarrow$  self-modified instruction

#### **BOA** Self modification and execution waves

- **Detect**: [Bonfante, Fernandez, Marion, Rouxel, Sabatier et Thierry 2015]
  - Keep a list of modified memory cells
  - Check every instruction before symbolic execution
- O Handle :
  - Concept of execution waves
  - Each instruction is associated to a wave





⇐ self-modification (@[A] replaced by 0x53)

 $\Leftrightarrow$  self-modified instruction

Type d'analyse :	Statique	Dynamique											
Type de détection :				Instructio		Page mémoire	Appel système	Table d'import					
Packer	BOA	CoDisasm [15]	<b>PinSec</b> [8]	Poly- [61] Unpack	<b>Renovo</b> [41]	PinDe- [49] monium	RePE- [45] construct	Omni- [50] Unpack	<b>Eureka</b> [66]	Bin- [20] Unpack			
ACPacker	$12/\sqrt{\times}$	$635/\checkmark/\checkmark$	-	-	-	-	-	-	-	-			
ACProtect 2.0	811/√/×	$971/\sqrt{/}$	$4/\sqrt{\times}$	-	-	-/√	-	-/√	-	-/√			
Armadillo 3.78	×/×	-	$1/\sqrt{\times}$	√/×	×/×	-	-	-/~	-/√	-/√			
Armadillo 9.64	×/×	$165/\checkmark/\checkmark$	-	-	-	-	-	-	-	-			
ASPack 2.12	$3/\sqrt{}$	$3/\sqrt{}$	$2/\sqrt{}$	$\sqrt{/\sim}$	111	-/√	-	-/√	-/√	-/√			
ASPack 2.40	$3/\sqrt{}$	$3/\sqrt{}$	-	-	-	, _	-	-	<i>.</i>	-			
EP Protector 0.3	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{}$	-	-	-	-	-	-	-			
eXPressor	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-/√	-	-	-	-/√			
FSG 2.0	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	$\sqrt{1}$	$\sqrt{\sqrt{1}}$	-/~	×/×	-/√	-/√	-/√			
JDPack 2.0	$2/\sqrt{\times}$	$3/\sqrt{}$	×/×	-	-	-	-	-	´_	-			
MEW	$2/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	$\sqrt{1}$	$\sqrt{1}$	-/√	-	-/√	-/√	-/√			
MoleBox	$2/\sqrt{\times}$	$3/\sqrt{1}$	$2/\sqrt{}$	×/×	515	-	$\sqrt{1}$	-/~	-/~	-/~			
Morphine 2.7	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$		515	$\sqrt{1}$	-	-	-/~	-/~	-			
Mystic	×/×	$4/\sqrt{\sqrt{1}}$	$1/\sqrt{}$	-	-	-	-	-	-	-			
Neolite 2.0	×/×	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	$\sqrt{1}$	-	-	-			
nPack 1.1.300	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-/√			
Obsidium 1364	$6/\sqrt{\times}$	$16/\sqrt{\sqrt{1}}$	×/×	×/×	×/×	-/√	-	-	-/√	-/√			
Packman 1.0	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	_	-	$\sqrt{1}$	-	-	-			
PE Compact 2.20	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	×/×	111	-/√	111	-	-/√	-/√			
PE Compact 3.02.2	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$		-	-	-	-	-	-	-			
PELock	$2/\sqrt{\times}$	$15/\sqrt{\times}$	$6/\sqrt{}$	-	-	-/√	-	-	-	-/√			
PESpin 1.1	$5/\sqrt{\times}$	80/√/×	×/×	-	-	-	√/×	-	-	-/√			
Petite 2.2	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	×/×	-	-	-	515	-	-	-/√			
RLPack	$2/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-/√			
Setisoft 2.7.1	×/×	$32/\sqrt{\times}$	$4/\sqrt{\times}$	-	-	-	-	-	-	-			
SVK 1.43f	$2/\sqrt{\times}$	$35/\sqrt{\times}$	×/×	-	-	-	-	-	-	-			
tElock 0.51	$5/\sqrt{\times}$	$17/\sqrt{\sqrt{1}}$	5/√/×	-	-	-	√/×	-/×	-	-/√			
Elock 0.99	$14/\sqrt{\times}$	$18 / \sqrt{\sqrt{10}}$	-	-	-	-	-	-	-	-			
Themida 1.8	3/√/×	_	×/×	×/×	$\sqrt{/\sim}$	-	√/×	-/√	-/~	-/√			
Themida 2.0.3	$3/\sqrt{\times}$	$106 / \sqrt{/}$	-	-	-	-	-	-	-	-			
Upack 0.39	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	-	-	-	-	-	-	-			
UPX 2.90	$2/\sqrt{\sqrt{1}}$	$2/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	$\sqrt{1}$	$\sqrt{1}$	-/√	$\sqrt{1}$	-/√	-/√	-			
WinUpack	$\frac{2}{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{$	$\frac{2}{3}/\sqrt{\sqrt{3}}$	$\frac{1}{2}/\sqrt{\sqrt{3}}$	$\sqrt{ \sim}$	515	-/~	×/×	-/√	-/~	-/√			
Yoda's Crypter 1.3	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	$\frac{2}{\sqrt{\times}}$	- /	-	-	-	-	-	-/~			
Yoda's Protector 1.02	$\frac{1}{2}/\sqrt{\times}$	$6/\sqrt{}$	×/×	$\sqrt{/\sim}$	$\sqrt{/\sim}$	-	-	-	-/√	-/~			

ASPack Compact New Yorking Aspace Asp

n/d/r: nombre de vagues détectées (n) / code auto-modifiant détecté (d) / récupération du code du binaire original (r)

 $\checkmark$ : réussite,  $\sim$ : réussite partielle,  $\times$ : échec, - : absence de résultat

Type d'analyse :	Statique					Dyna	mique				ACPUTE ASPack CAP CSSOF ADDA
Type de détection :				Instructio	n			Page mémoire	Appel système	Table d'import	ASPack ACProtect Metrosor New npack ACProtect Metrosor SVR Packing tElock & Protector SVR Yodaa & Protector SVR New Yoda
Packer	BOA	CoDisasm [15]	<b>PinSec</b> [8]	Poly- [61] Unpack	<b>Renovo</b> [41]	PinDe- [49] monium	RePE- [45] construct	Omni- [50] Unpack	<b>Eureka</b> [66]	Bin- [20] Unpack	Stra War Epiperetor Today Protector JDPack Wildow Mery SG P
ACPacker	$12/\sqrt{\times}$	$635/\checkmark/\checkmark$	-	-	-	-	-	-	-	-	ACPacker More and Acpack of the Acpack of th
ACProtect 2.0	811/√/×	$971/\sqrt{}$	$4/\sqrt{\times}$	-	-	-/√	-	-/√	-	-/√	ACPacker Molebox
Armadillo 3.78	×/×	-	$1/\sqrt{\times}$	√/×	×/×	-	-	-/~	-/√	-/√	rit Compare
Armadillo 9.64	×/×	$165/\checkmark/\checkmark$	-	-	-	-	-	-	-	-	
ASPack 2.12	$3/\sqrt{}$	$3/\sqrt{}$	$2/\checkmark/\checkmark$	$\sqrt{/\sim}$	11	-/√	-	-/√	-/√	-/√	
ASPack 2.40	$3/\sqrt{}$	$3/\sqrt{}$	-	-	-	-	-	-	-	·	
EP Protector 0.3	$2/\sqrt{/}$	$2/\sqrt{}$	$1/\checkmark/\checkmark$	-	-	-	-	-	-	-	•••••
eXPressor	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-/√	-	-	-	-/√	original binary : hostname.exe
FSG 2.0	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	$\sqrt{1}$	515	-/~	×/×	-/√	-/√	-/√	
JDPack 2.0	$2/\sqrt{\times}$	$3/\sqrt{}$	×/×	-	-	-	-	-	-	· -	35 pecked versions
MEW	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{}$	515	11	-/√	-	-/√	-/√	-/√	
MoleBox	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	×/×	111	-	$\sqrt{1}$	-/√	-/√	-/~	Commercia en with 0 dynamic toolo
Morphine 2.7	$2/\sqrt{\times}$	$3/\sqrt{}$	-	515	$\sqrt{/\sim}$	-	-	-/√	-/√	· -	Comparison with 9 dynamic tools
Mystic	×/×	$4/\sqrt{}$	$1/\checkmark/\checkmark$	-	-	-	-	-	-	-	
Neolite 2.0	×/×	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	$\sqrt{\sqrt{1}}$	-	-	-	
nPack 1.1.300	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-/√	
Obsidium 1364	$6/\sqrt{\times}$	$16/\sqrt{}$	×/×	×/×	$\times / \times$	-/√	-	-	-/√	-/~	
Packman 1.0	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{}$	-	-	-	$\sqrt{1}$	-	-	· -	
PE Compact 2.20	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	$\times / \times$	11	-/√	515	-	-/√	-/√	
PE Compact 3.02.2	$4/\sqrt{}$	$4/\sqrt{}$	-	-	-	-	-	-	-	-	
PELock	$2/\sqrt{\times}$	$15/\sqrt{\times}$	$6/\checkmark/\checkmark$	-	-	-/√	-	-	-	-/√	
PESpin 1.1	$5/\sqrt{\times}$	80/√/×	×/×	-	-	-	√ / ×	-	-	-/√	
Petite 2.2	$2/\sqrt{\times}$	$3/\sqrt{}$	×/×	-	-	-	111	-	-	-/~ L	
RLPack	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-/√	
Setisoft 2.7.1	×/×	$32/\sqrt{\times}$	$4/\sqrt{\times}$	-	-	-	-	-	-	-	
SVK 1.43f	$2/\sqrt{\times}$	$35/\sqrt{\times}$	×/×	-	-	-	-	-	-	-	
tElock 0.51	$5/\sqrt{\times}$	$17/\sqrt{\sqrt{1}}$	5/√/×	-	-	-	√/×	-/×	-	-/√	
tElock 0.99	$14/\sqrt{\times}$	18 / √ / √	<i>, _ ,</i>	-	-	-	-	, _	-	-	
Themida 1.8	3/√/×	-	×/×	×/×	$\sqrt{/\sim}$	-	√/×	-/√	-/~	-/√	
Themida 2.0.3	$3/\sqrt{\times}$	106 / 🗸 / 🗸	-	, _	-	-	-	-	-	-	
Upack 0.39	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	-	-	-	-	-	-	-	
UPX 2.90	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	$\sqrt{1}$	111	-/√	$\sqrt{1}$	-/√	-/√	-	
WinUpack	$\frac{3}{\sqrt{\sqrt{3}}}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	$\sqrt{1}$	515	-/~	×/×	-/√	-/√	-/√	
Yoda's Crypter 1.3	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	$\frac{3}{\sqrt{\times}}$	-	-	-	-	-	-	-/√	
Yoda's Protector 1.02	$2/\sqrt{\times}$	$6/\sqrt{}$	×/×	$\sqrt{/\sim}$	√/~				-/√	-/√	

 $\checkmark$ : réussite,  $\sim$ : réussite partielle,  $\times$ : échec, - : absence de résultat

La version exacte des packers utilisés est seulement connue pour BOA, CoDisasm et PinSec.

UPX

Type d'analyse :	Statique					Dyna	mique				ACPROTECT ASPACK TO CALL AND A APPLICATION AND A
Type de détection :				Instruction	n			Page mémoire	Appel système	Table d'import	ASPack Messor Anti-
Packer	BOA	CoDisasm [15]	<b>PinSec</b> [8]	Poly- [61] Unpack	<b>Renovo</b> [41]	PinDe- [49] monium	RePE- [45] construct	Omni- [50] Unpack	<b>Eureka</b> [66]	<b>Bin-</b> [20] <b>Unpack</b>	SVR Protector volte protector DPack With Mery SG FSG
ACPacker ACProtect 2.0	$12/\checkmark/\times$ $811/\checkmark/\times$	$\begin{array}{c} 635/\checkmark/\checkmark\\ 971/\checkmark/\checkmark\end{array}$	- 4/√/×	-	-	- -/√	-	- -/√	-	- -/√	ACPeace of Morphine Armadule UPX ACPacker JDPack UPX Molebox Moha
Armadillo 3.78 Armadillo 9.64	×/× ×/×	- 165/√/√	1/√/× -	√/× -	×/×		-	-/~	-/√ _	-/~	R Constant
ASPack 2.12	$3/\sqrt{\sqrt{1}}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	$\sqrt{/\sim}$	111	-/√	-	-/√	-/√	-/√	
ASPack 2.40	$3/\sqrt{}$	$3/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-	·	
EP Protector 0.3	$2/\sqrt{}$	$2/\sqrt{}$	$1/\checkmark/\checkmark$	-	-	-	-	-	-	-	
eXPressor	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-/√	-	-	-	-/√	original binary : hostname.exe
FSG 2.0	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	515	515	-/~	×/×	-/√	-/√	-/√	
JDPack 2.0	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	×/×	-	-	-	-	-	-	-	35 pecked versions
MEW	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{}$	515	111	-/√	-	-/√	-/√	-/√	p
MoleBox	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	×/×	111	-	$\sqrt{\sqrt{1}}$	-/~	-/~	-/√	Comparison with 9 dynamic tools
Morphine 2.7	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	-	515	√/~	-	-	-/√	-/√	-	Comparison with 9 dynamic tools
Mystic	×/×	$4/\sqrt{}$	$1/\checkmark/\checkmark$	-	-	-	-	-	-	-	
Neolite 2.0	×/×	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	$\sqrt{\sqrt{1}}$	-	-	-	Failed : 5
nPack 1.1.300	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-/√	
Obsidium 1364	$6/\sqrt{\times}$	$16/\checkmark/\checkmark$	×/×	×/×	×/×	-/√	-	-	-/√	-/√	
Packman 1.0	$2/\sqrt{/}$	$2/\sqrt{}$	$1/\sqrt{}$	-	-	-	$\sqrt{1}$	-	-	-	
PE Compact 2.20	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	×/×	111	-/√	515	-	-/√	-/√	
PE Compact 3.02.2	$4/\sqrt{}$	$4/\sqrt{}$	-	-	-	-	-	-	-	-	
PELock	$2/\sqrt{\times}$	$15/\sqrt{\times}$	$6/\checkmark/\checkmark$	-	-	-/√	-	-	-	-/√	
PESpin 1.1	$5/\sqrt{\times}$	80/√/×	×/×	-	-	-	√/×	-	-	-/√	
Petite 2.2	$2/\sqrt{\times}$	$3/\sqrt{}$	×/×	-	-	-	515	-	-	-/√ L	
RLPack	$2/\sqrt{}$	$2/\sqrt{}$	$1/\checkmark/\checkmark$	-	-	-	-	-	-	-/√	
Setisoft 2.7.1	×/×	$32/\sqrt{\times}$	$4/\sqrt{\times}$	-	-	-	-	-	-	-	
SVK 1.43f	$2/\sqrt{\times}$	$35/\sqrt{\times}$	×/×	-	-	-	-	-	-	-	
t E lock 0.51	$5/\sqrt{\times}$	$17/\sqrt{/}$	5/√/×	-	-	-	√/×	-/×	-	-/√	
t E lock 0.99	$14/\sqrt{\times}$	$18 / \checkmark / \checkmark$	-	-	-	-	-	-	-	-	
Themida 1.8	$3/\sqrt{\times}$	-	×/×	×/×	√/~	-	√/×	-/√	-/~	-/√	
Themida 2.0.3	3/√/×	$106 / \checkmark / \checkmark$	-	-	-	-	-	-	-	-	
Upack 0.39	$2/\sqrt{\times}$	$3/\sqrt{}$	$2/\checkmark/\checkmark$	-	-	-	-	-	-	-	
UPX 2.90	$2/\sqrt{/}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	11	$\sqrt{\sqrt{1}}$	-/√	$\sqrt{1}$	-/√	-/√	-	
WinUpack	$3/\sqrt{\sqrt{1}}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{/}$	√/~	111	-/~	×/×	-/√	-/~	-/√	
Yoda's Crypter 1.3	$4/\sqrt{}$	$4/\sqrt{}$	3/√ / ×	-	-	-	-	-	-	-/√	
Yoda's Protector 1.02	$2/\sqrt{\times}$	$6/\sqrt{}$	×/×	$\sqrt{/\sim}$	√/~	-	-	-	-/√	-/~	

 $\checkmark$ : réussite,  $\sim$ : réussite partielle,  $\times$ : échec, - : absence de résultat

Type d'analyse :	Statique					Dyna	mique				ACProtect
Type de détection :				Instructio	n			Page mémoire	Appel système	Table d'import	nPack A
Packer	BOA	CoDisasm [15]	<b>PinSec</b> [8]	Poly- [61] Unpack	<b>Renovo</b> [41]	PinDe- [49] monium	RePE- [45] construct	Omni- [50] Unpack	<b>Eureka</b> [66]	Bin- [20] Unpack	SVR Myst
ACPacker	$12/\sqrt{\times}$	$635/\checkmark/\checkmark$	-	-	-	-	-	-	-	-	nPac
ACProtect 2.0	811/√/×	971/√/√	$4/\sqrt{\times}$	-	-	-/√	-	-/√	-	-/√	
Armadillo 3.78	×/×	-	$1/\sqrt{\times}$	√/×	×/×	-	-	-/~	-/√	-/√	
Armadillo 9.64	×/×	$165/\checkmark/\checkmark$	-	-	-	-	-	-	-	-	
ASPack 2.12	$3/\sqrt{}$	$3/\sqrt{}$	$2/\checkmark/\checkmark$	$\sqrt{/\sim}$	11	-/√	-	-/√	-/√	-/√	
ASPack 2.40	$3/\sqrt{\sqrt{1}}$	$3/\sqrt{}$	-	-	-	-	-	-	-	- r	
EP Protector 0.3	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{}$	-	-	-	-	-	-	-	
eXPressor	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-/√	-	-	-	-/√	original
FSG 2.0	$2/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	515	515	-/~	×/×	-/√	-/√	-/√	
JDPack 2.0	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	×/×	-	-	-	-	-	-	, -	35 pecke
MEW	$2/\sqrt{}$	$2/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	$\sqrt{1}$	$\sqrt{1}$	-/√	-	-/√	-/√	-/√	oo peerie
MoleBox	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	×/×	111	-	515	-/~	-/~	-/√	<u> </u>
Morphine 2.7	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$		515	$\sqrt{ \sim}$	-	-	-/~	-/√	-	Compari
Mystic	×/×	$4/\sqrt{\sqrt{1}}$	$1/\sqrt{}$	-	-	-	-	-	-	-	
Neolite 2.0	×/×	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	$\sqrt{1}$	-	-	-	Failed :
nPack 1.1.300	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-/√	
Obsidium 1364	6/√/×	$16/\sqrt{}$	×/×	×/×	×/×	-/√	-	-	-/√	-/√	Partially
Packman 1.0	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{}$	-	-	-	$\sqrt{1}$	-	-	-	Partially
PE Compact 2.20	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	×/×	111	-/√	111	-	-/√	-/√	
PE Compact 3.02.2	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	_, . , .	_	-	-	-	-	-	-	
PELock	2/√/×	$15/\sqrt{\times}$	$6/\sqrt{}$	_	_	-/√	_	-	_	-/√	
PESpin 1.1	$5/\sqrt{\times}$	80/√/×	×/×	_	_	-	√/×	-	_	-/√	
Petite 2.2	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	×/×	_	-	-	11	-	_	-/√	
RLPack	$2/\sqrt{/}$	$2/\sqrt{\sqrt{2}}$	$1/\sqrt{\sqrt{1}}$	_	-	-	• / •	-	_	-/√	
Setisoft 2.7.1	×/×	$32/\sqrt{\times}$	$4/\sqrt{\times}$	_	-	-	_	-	_		
SVK 1.43f	2/√/×	$35/\sqrt{\times}$	×/×	_	_		_	-	_	_	
tElock 0.51	$5/\sqrt{\times}$	$17/\sqrt{/}$	5/√/×	_	_	_	√/×	-/×	_	-/√	
tElock 0.99	$14/\sqrt{\times}$	$18 / \sqrt{\sqrt{10}}$	-	_	_	_	• / ^	-/ -	_	-/ •	
Themida 1.8	$3/\sqrt{\times}$	10 / • / •	×/×	×/×	√/~		√/×	-/√	-/~	-/√	
Themida 2.0.3	$3/\sqrt{\times}$	$106 / \sqrt{/}$	^/ ^	~/~	• / · •	-	v / ^	-/ v	-//~	-/ 🗸	
Upack 0.39	$\frac{3}{\sqrt{2}}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	-	-	-	-	-	-	-	
UPX 2.90	$2/\sqrt{\chi}$ $2/\sqrt{\chi}$	$3/\sqrt{\sqrt{2}}$ $2/\sqrt{\sqrt{2}}$	$\frac{2}{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{$	- ~   ~	- ~   ~	-/√	- - 	-/√	-	-	
WinUpack	$\frac{2}{\sqrt{3}}$	$\frac{2}{\sqrt{3}}$	$\frac{1}{\sqrt{\sqrt{\sqrt{\sqrt{2}}}}}$	√ / ~ √ / ~	V   V V   V	-/√ -/√	√ /√ ×/×	-/v -/v	-/√ -/√	-/√	
Yoda's Crypter 1.3	, ,	, ,	, ,	<ul> <li>✓ / ~</li> </ul>	v / v	-/~	× / ×	-/ v	-/ 🗸	'	
	$4/\sqrt{}$	$4/\sqrt{\sqrt{1}}$	$3/\sqrt{\times}$	_	-	-	-	-	-	-/~	
Yoda's Protector 1.02	$2/\sqrt{\times}$	$6/\sqrt{}$	×/×	√/~	√/~	-	-	-	-/√	-/√	



-	original binary : hostname.exe
-	35 pecked versions
-	Comparison with 9 dynamic tools
-	Failed : 5
-	Partially unpacked : 16

 $\checkmark$ : réussite,  $\sim$ : réussite partielle,  $\times$ : échec, -: absence de résultat

Type d'analyse :	Statique					Dyna	mique				ACProtect
Type de détection :				Instructio	n			Page mémoire	Appel système	Table d'import	nPack AC
Packer	BOA	CoDisasm [15]	<b>PinSec</b> [8]	Poly- [61] Unpack	<b>Renovo</b> [41]	PinDe- [49] monium	RePE- [45] construct	Omni- [50] Unpack	<b>Eureka</b> [66]	Bin- [20] Unpack	svk Mystic
ACPacker	$12/\checkmark/\times$	$635/\checkmark/\checkmark$	-	-	-	-	-	-	-	-	nPack
ACProtect 2.0	811/√/×	$971/\checkmark/\checkmark$	$4/\sqrt{\times}$	-	-	-/√	-	-/√	-	-/√	
Armadillo 3.78	×/×	-	$1/\sqrt{\times}$	√ / ×	×/×	-	-	-/~	-/√	-/√	
Armadillo 9.64	×/×	$165/\checkmark/\checkmark$	-	-	-	-	-	-	-	-	
ASPack 2.12	$3/\checkmark/\checkmark$	$3/\checkmark/\checkmark$	$2/\checkmark/\checkmark$	√/~	11	-/√	-	-/√	-/√	-/√	
ASPack 2.40	$3/\checkmark/\checkmark$	$3/\checkmark/\checkmark$	-	-	-	-	-	-	-	- г	
EP Protector 0.3	$2/\checkmark/\checkmark$	$2/\checkmark/\checkmark$	$1/\checkmark/\checkmark$	-	-	-	-	-	-	-	
eXPressor	$2/\sqrt{}$	$2/\checkmark/\checkmark$	$1/\checkmark/\checkmark$	-	-	-/√	-	-	-	-/√	🐘 original b
FSG 2.0	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	111	$\sqrt{\sqrt{1}}$	-/~	×/×	-/√	-/√	-/√	
JDPack 2.0	$2/\sqrt{\times}$	$3/\sqrt{}$	×/×	-	-	-	-	-	-	-	35 pecked
MEW	$2/\checkmark/\checkmark$	$2/\sqrt{}$	$1/\checkmark/\checkmark$	111	11	-/√	-	-/√	-/√	-/√	
MoleBox	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	×/×	515	-	$\sqrt{1}$	-/√	-/~	-/~	Companie
Morphine 2.7	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	-	111	$\sqrt{/\sim}$	-	-	-/√	-/~	-	Comparise
Mystic	×/×	$4/\sqrt{}$	$1/\checkmark/\checkmark$	-	-	-	-	-	-	-	
Neolite 2.0	×/×	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	111	-	-	-	Failed : 5
nPack 1.1.300	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-/√	
Obsidium 1364	6/√/×	$16/\sqrt{}$	×/×	×/×	×/×	-/√	-	-	-/√	-/~	Partially
Packman 1.0	$2/\sqrt{\sqrt{1}}$	$2/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	-	-	,	515	-	, _	-	Partially
PE Compact 2.20	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	×/×	111	-/√	515	-	-/√	-/√	
PE Compact 3.02.2	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	-	-	-	, _	-	-	<i>.</i>	-	Totally un
PELock	$2/\sqrt{\times}$	$15/\sqrt{\times}$	$6/\sqrt{}$	-	-	-/√	-	-	-	-/√	
PESpin 1.1	$5/\sqrt{\times}$	80/√/×	×/×	-	-	-	√/×	-	-	-/~	
Petite 2.2	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	×/×	-	-	-	515	-	-	-/√	
RLPack	$2/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-/~	
Setisoft 2.7.1	×/×	$32/\sqrt{\times}$	$4/\sqrt{\times}$	-	-	-	-	-	-	-	
SVK 1.43f	2/√/×	35/√/×	×/×	-	-	-	-	-	-	-	
tElock 0.51	$5/\sqrt{\times}$	$17/\sqrt{}$	5/√/×	-	-	-	√/×	-/×	-	-/√	
tElock 0.99	$14/\sqrt{\times}$	$18/\sqrt{}$	-	-	-	-	-	-	-	-	
Themida 1.8	$3/\sqrt{\times}$	-	×/×	×/×	$\sqrt{/\sim}$	-	√/×	-/√	-/~	-/√	
Themida 2.0.3	$3/\sqrt{\times}$	106 /√/√	-	-	-	-	-	-	_	-	
Upack 0.39	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	-	-	-	-	-	-	-	
UPX 2.90	$2/\sqrt{/}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	$\sqrt{1}$	111	-/√	$\sqrt{1}$	-/√	-/√	_	
WinUpack	$\frac{2}{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{$	$3/\sqrt{\sqrt{3}}$	$\frac{1}{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{$	√/~	11	-/~ -/√	×/×	_/ ↓ _/√	-/√ -/√	-/√	
Yoda's Crypter 1.3	$4/\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	$\frac{2}{\sqrt{\times}}$	-			010		/*	-/v -/v	
Yoda's Protector 1.02		$6/\sqrt{}$	$\frac{3}{\times}$	√ /~	√ /~	-	-		-/√	-/v -/v	
10da 5 1 10tector 1.02	2/ 1 / ^	0/1/1	~/ ^	• //~	• //~	-	-	-	-/ <b>v</b>	-/ <b>v</b>	



35 pecked versions
 Comparison with 9 dynamic tools
 Failed : 5
 Partially unpacked : 16
 Totally unpacked : 14

n/d/r: nombre de vagues détectées (n) / code auto-modifiant détecté (d) / récupération du code du binaire original (r)

 $\checkmark$ : réussite,  $\sim$ : réussite partielle,  $\times$ : échec, -: absence de résultat

Type d'analyse :	Statique					Dyna	mique				ACTIVITE ASPack Concert Medite
Type de détection :				Instructio	n		inique	Page mémoire	Appel système	Table d'import	Acres ASPack of the store of th
rype de detection :		CoDisasm	PinSec	Poly- [61]		<b>PinDe-</b> [49]	<b>RePE-</b> [45]	<b>Omni-</b> [50]	Eureka	Bin- [20]	
Packer	BOA	[15]	[8]	Unpack	<b>Renovo</b> [41]	monium	construct	Unpack	[66]	Unpack	Mystic Mystic ACPartice IDPack UPX
ACPacker	$12/\checkmark/\times$	$635/\checkmark/\checkmark$		-	-	-	-	-		-	ACPacketer JDPack
ACProtect 2.0	811/√/×	$971/\checkmark/\checkmark$	$4/\sqrt{\times}$	-	-	-/~	-	-/ 🗸	-	-/√	ACPacker Molebox
Armadillo 3.78	×/×	-	$1/\sqrt{\times}$	√/×	×/×	-	-	-/~	-/√	-/√	- compary
Armadillo 9.64	×/×	$165/\checkmark/\checkmark$	-	-	-	-	-	-	-	-	
ASPack 2.12	$3/\checkmark/\checkmark$	$3/\checkmark/\checkmark$	$2/\checkmark/\checkmark$	√/~	$\sqrt{\sqrt{1}}$	-/√	-	-/√	-/√	-/√	
ASPack 2.40	$3/\sqrt{}$	$3/\sqrt{}$	-	-	-	-	-	-	-		
EP Protector 0.3	$2/\sqrt{}$	$2/\sqrt{}$	$1/\checkmark/\checkmark$	-	-	-	-	-	-	-	•••••
eXPressor	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-/√	-	-	-	-/√	original binary : hostname.exe
FSG 2.0	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	$\sqrt{1}$	515	-/~	×/×	-/√	-/√	-/√	
JDPack 2.0	2/√/×	$3/\sqrt{\sqrt{1}}$	×/×	-	-	,	-	-	-	·	35 pecked versions
MEW	$2/\sqrt{}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	$\sqrt{1}$	$\sqrt{1}$	-/√	-	-/√	-/√	-/√	oo peeried versions
MoleBox	2/√/×	$3/\sqrt{\sqrt{1}}$	$2/\sqrt{}$	×/×	515	-	$\sqrt{1}$	-/~	-/~	-/√	
Morphine 2.7	$2/\sqrt{\times}$	$3/\sqrt{\sqrt{1}}$		515	$\sqrt{1}$	-	-	-/~	-/~	-	Comparison with 9 dynamic tools
Mystic	×/×	$4/\sqrt{\sqrt{1}}$	$1/\sqrt{}$	-	-	-	-	-	-	-	
Neolite 2.0	×/×	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$	-	-	-	$\sqrt{1}$	-	-	_	Failed : 5
nPack 1.1.300	$2/\sqrt{}$	$2/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	-	-	-	-	-	-	-/√	
Obsidium 1364	6/√/×	$16/\sqrt{}$	×/×	×/×	×/×	-/√	-	-	-/√	-/√	Desticilly upped add 10
Packman 1.0	$2/\sqrt{}$	$2/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	-		-	$\sqrt{1}$	-	-	-	Partially unpacked : 16
PE Compact 2.20	$\frac{1}{4}\sqrt{\sqrt{1}}$	$4/\sqrt{\sqrt{1}}$	$1/\sqrt{\sqrt{1}}$	×/×	$\sqrt{1}$	-/√	515	_	-/√	-/√	
PE Compact 3.02.2	$4/\sqrt{}$	$4/\sqrt{\sqrt{1}}$			• / •	-	• / •	_	-	-	Totally unpacked : 14
PELock	2/√/×	$15/\sqrt{\times}$	$6/\sqrt{}$	_	_	-/√	_		_	-/√	<i>,</i> ,
PESpin 1.1	$5/\sqrt{\times}$	$\frac{10}{\sqrt{\times}}$	×/×	_		-/*	√/×		_	-/√	
Petite 2.2	$\frac{3}{\sqrt{x}}$	$3/\sqrt{\sqrt{1}}$	$\frac{x}{x}$	_		-	x 1 x	_	_	-/* L	
RLPack	$2/\sqrt{/}$	$2/\sqrt{}$	$1/\sqrt{\sqrt{1}}$			_	• / •			-/v	
Setisoft 2.7.1	×/×	$\frac{2}{32}/\sqrt{\times}$	$\frac{1}{\sqrt{\frac{1}{\frac{1}{\frac{1}{\frac{1}{\frac{1}{\frac{1}{1$	-	-	-	-	-	_	-/ •	
SVK 1.43f	$\frac{1}{2/\sqrt{x}}$	$32/\sqrt{\times}$ $35/\sqrt{\times}$	×/×	-	-	-	-	-	-	-	
tElock 0.51	$\frac{2}{\sqrt{x}}$	$17/\sqrt{\sqrt{1}}$	$5/\sqrt{\times}$	-	-	-	√/×	-/×	-	-/√	
tElock 0.99	$14/\sqrt{\times}$	$17/\sqrt{\sqrt{10}}$ $18/\sqrt{\sqrt{10}}$	3/ v / ×	-	-	-	v / ×	-/×	-	-/~	<b>BOA</b> :
		10 / V / V	-	-	_	-	-		-		
Themida 1.8	$3/\sqrt{\times}$	- 106 /√/√	×/×	×/×	√/~	-	$\checkmark/\times$	-/√	-/~	-/√	
Themida 2.0.3	$3/\sqrt{\times}$	/ /	-	-	-	-	-	-	-	-	100% static
Upack 0.39	2/√/×	$3/\checkmark/\checkmark$	$2/\checkmark/\checkmark$	-	_	-	-	-	-	-	TOO 20 STALL
UPX 2.90	$2/\sqrt{\sqrt{2}}$	$2/\sqrt{}$	$1/\sqrt{}$	11	$\sqrt{\sqrt{1}}$	-/~	$\sqrt{1}$	-/~	-/~	-	
WinUpack	$3/\sqrt{\sqrt{1}}$	$3/\sqrt{\sqrt{1}}$	$2/\checkmark/\checkmark$	√/~	$\sqrt{\sqrt{1}}$	-/√	×/×	-/√	-/√	-/√	
Yoda's Crypter 1.3	$4/\sqrt{}$	$4/\sqrt{}$	3/√ / ×	-	-	-	-	-	-	-/~	
Yoda's Protector 1.02	$2/\checkmark/\times$	$6/\checkmark/\checkmark$	×/×	√/~	√/~	-	-	-	-/√	-/√	

 $\checkmark$ : réussite,  $\sim$ : réussite partielle,  $\times$ : échec, - : absence de résultat

### Analysis of a real malware: Emotet





#### BOA The Emotet trojan

#### Features

- Steal password and banking information
- Network propagation
- Loader and botnet
- O Analysis of a « recent » sample
  - October 14, 2020: detected as malware by 7/63 of VirusTotal's antivirus

#### BOA Emotet: BOA analysis

#### **First wave :**

- 3 341 instructions
- No exception
- 145 RET instructions without call stack tampering
- 10 external functions address retrieved by GetProcAddress
- Construction of the second wave (after ~28 000 executed instructions):
  - Allocation of 117 043 bytes with VirtualAllocExNuma
  - Un-pack loop execution (117 042 iterations)
  - Jump on self-modified code with call ebx 🖒 enter in second wave

#### **BOA** Emotet: BOA analysis

#### **○** Second wave dump:

- « Raw » dump
- Detected as *Emotet* by 13/60 VirusTotal's antivirus

Ad-Aware	() Generic.EmotetAC.F21A24A9	ALYac	() Generic.EmotetAC.F21A24A9
Arcabit	() Generic.EmotetAC.F21A24A9	Avast	() Win32:lcedID-A [Trj]
AVG	() Win32:lcedID-A [Trj]	BitDefender	() Generic.EmotetAC.F21A24A9
DrWeb	① Trojan.Emotet.1031	Emsisoft	() Generic.EmotetAC.F21A24A9 (B)
eScan	() Generic.EmotetAC.F21A24A9	FireEye	() Generic.EmotetAC.F21A24A9
GData	() Generic.EmotetAC.F21A24A9	Gridinsoft	() Trojan.Emotet.B.sdlyf
MAX	() Malware (ai Score=86)	AegisLab	⊘ Undetected



#### Conclusion Perspectives

- Yet another tool
  - A static analysis tool...
  - ...with some features of a dynamic analysis

### Needs some improvements

- Improve SMT solvers part
- Multi-thread, multi-process, best OS simulation,...

- Bergeron, Debbabi, Desharnais, Erhioui, Lavoie et Tawbi 2001
   Static Detection of Malicious Code in Executable Programs
- Bonfante, Fernandez, Marion, Rouxel, Sabatier et Thierry 2015
   CoDisasm: Medium Scale Concatic Disassembly of Self-Modifying Binaries with Overlapping Instructions
- Biondi, Rigo, Zennou et Mehrenberger 2017
   BinCAT: purrfecting binary static analysis
- Collberg, Thomborson et Low 1997
   A Taxonomy of Obfuscating Transformations
- Djoudi et Bardin 2015
   BINSEC: Binary Code Analysis with Low-Level Regions
- Junod, Rinaldini, Wehrli et Michielin 2015
   Obfuscator-LLVM -- Software Protection for the Masses
- Kiss, Lalande, Leslous et Tong 2016
   Kharon dataset: Android malware under a microscope
- Pietrek 1997

A crash course on the depths of Win32 structured exception handling

Qiu, Su et Ma - 2014

Deobfuscate Non-Returning Calls and Call-Stack Tampering in Instruction Traces

Robin - 2017

Formal Approaches for Automatic Deobfuscation and Reverse-engineering of Protected Codes

Salwan - 2020

L'usage de l'exécution symbolique pour la déobfuscation binaire en milieu industriel

- Schwarz, Debray et Andrews 2002
   Disassembly of executable code revisited
- Shoshitaishvili, Wang, Salls, Stephens, Polino, Dutcher, Grosen, Feng, Hauser, Kruegel, Vigna 2016 SOK: (State of) The Art of War: Offensive Techniques in Binary Analysis