

南 开 大 学

《恶意代码分析与防治技术》课程实验报告

实验一



学 院 网络空间安全学院
专 业 信息安全
学 号 2112060
姓 名 孙路
班 级 信息安全 1 班

《恶意代码分析与防治技术》课程 Lab1 实验报告

一、 实验目的	3
二、 实验原理	3
三、 实验过程	3
(一) Lab1-1	3
(二) Lab1-2	9
(三) Lab1-3	13
(四) Lab1-4	16
(五) Lab1-5	22
四、 实验结论及心得体会	26

一、实验目的

使用 Lab01-01.exe、Lab01-02.exe、Lab01-03.exe、Lab01-04.exe 和 Lab01-01.dll 文件，使用第一章描述的工具和技术分析上述文件并获取关于这些文件的信息。

二、实验原理

1. VirusTotal(<http://www.virustotal.com/>) 这样的网站允许上传一个文件，并将调用多个反病毒引擎来进行扫描。VirusTotal 网站还会生成一份报告，其中提供了所有引擎对这个样本的识别情况、标识这个样本是否恶意、恶意代码名称，以及其他额外信息。

2. Strings 程序可以从一个可执行程序中搜索 ASCII 和 Unicode 字符串时，它将忽略上下文和格式，所以它将分析任何文件类型，并从整个文件中检测出可打印字符串(这也意味着，它也会识别出实际上并非真正字符串的一些字符序列)。Strings 程序搜索三个或以上连续的 ASCII 或 Unicode 字符，并以终结符结尾的可打印字符串。

3. 可以使用 PEiD 来检测加壳器的类型，或是用来链接应用程序的编译器类型。

4. 相关脱壳工具可以对使用相关技术加壳的恶意代码进行脱壳。

5. Dependency Walker 工具支持列出可执行文件的动态链接函数。

6. PEview 可以用来分析 PE 文件相关信息

7. Resource Hacker 工具可以查看资源节。

三、实验过程

(一) Lab1-1

(1) 将文件上传至 <http://www.VirusTotal.com/> 进行分析并查看报告。文件匹配到了已有的反病毒软件特征吗？

44
/ 70

44 security vendors and no sandboxes flagged this file as malicious

Reanalyze Similar More

150e42c8d4fab649bde0398867e930b86c2a599e8db83b8260393082268f2dba

Lab01-01.dll

Size

160.00 KB

Last Analysis Date

3 hours ago

DLL

pedi armadillo via-tor

Community Score

DETECTION DETAILS RELATIONS BEHAVIOR COMMUNITY 30

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label trojan:fragtor/skeeyah

Threat categories trojan

Family labels fragtor skeeyah r002c0phf20

Security vendors' analysis

Do you want to automate checks?

Alibaba	Trojan.Win32/Skeeyah.7b0bebff	ALYac	Trojan.Agent.Waski
Antiy-AVL	Trojan.Win32.BTSGeneric	Arcabit	Trojan.Fragtor.D54B1F
Avast	Win32/Malware-gen	AVG	Win32/Malware-gen
BitDefender	Gen.Variant.Fragtor.346911	BitDefender.Theta	Gen.NN.ZedialF.36662.kq4@aGkQVip
Bkav Pro	W32.AIDetect/Malware	ClamAV	Win.Malware.Agent.6369668.0
CrowdStrike Falcon	Win/malicious_confidence_100% (W)	Cylance	Unsafe
Cynet	Malicious (score: 100)	Cyren	W32/Skeeyah.AK.gen/Eldorado
DeepInstinct	MALICIOUS	Elastic	Malicious (high Confidence)
Emsisoft	Gen.Variant.Fragtor.346911 (B)	eScan	Gen.Variant.Fragtor.346911
ESET-NOD32	A Variant Of Generic.TGEWDD	GData	Gen.Variant.Fragtor.346911
Google	Detected	Ikarus	Trojan.SuspectCRC
Lionic	Trojan.Win32.Skeeyah.4lc	MAX	Malware (ai Score=100)
MaxSecure	Trojan.Malware.7164915.susgen	McAfee	Generic.RXFO-RT1290934C61DE9
McAfee-GW-Editon	Generic.RXFO-RT1290934C61DE9	Microsoft	Trojan.Win32/Skeeyah.AMTB
NANO-Antivirus	Trojan.Win32.Waski.dtkvsp	Rising	Backdoor.SkeeyahH8.12823 (CLOUD)
Sangfor Engine Zero	Trojan.Win32.Skeeyah.Vqz9	SecureAge	Malicious
Sophos	Mail/Generic.R	Symantec	ML.Attribute.HighConfidence
Trapmine	Malicious.high.ml.score	Trellix (FireEye)	Generic.mg.290934c61de9176a
TrendMicro	TROJ_GEN.R002C0PHF20	TrendMicro-HouseCall	TROJ_GEN.R002C0PHF20
VIPRE	Gen.Variant.Fragtor.346911	VariT	Trojan.Win32.X.Paxoers2_c.LCM
Webroot	W32.Gen.BT	Xcitium	Malware@#2d9e4abnce61
Yandex	Trojan.Gen.AsaHtoPtb0Qvul0	Zillya	Adware.InstallCore.Win32.1036
Acronis (Static ML)	Undetected	AhnLab-V3	Undetected
Avira (no cloud)	Undetected	Baidu	Undetected
CMC	Undetected	DrWeb	Undetected
F-Secure	Undetected	Fortinet	Undetected
Gridinsoft (no cloud)	Undetected	Jiangmin	Undetected
K7AntiVirus	Undetected	K7GW	Undetected
Kaspersky	Undetected	Malwarebytes	Undetected
Palo Alto Networks	Undetected	Panda	Undetected
QuickHeal	Undetected	SentinelOne (Static ML)	Undetected
SUPERAntiSpyware	Undetected	TACHYON	Undetected
TEHTRIS	Undetected	Tencent	Undetected
VBA32	Undetected	ViRobot	Undetected
ZoneAlarm by Check Point	Undetected	Zoner	Undetected
Avast-Mobile	Unable to process file type	BitDefenderFaix	Unable to process file type
Cybereason	Unable to process file type	Symantec Mobile Insight	Unable to process file type
Trustlook	Unable to process file type		

VirusTotal

Contact Us
Get Support
How It Works
ToS | Privacy Policy
Blog | Releases

Community

Join Community
Vote and Comment
Contributors
Top Users
Community Buzz

Tools

API Scripts
YARA
Desktop Apps
Browser Extensions
Mobile App

Premium Services

Get a demo
Intelligence
Hunting
Graph
API v3 | v2

Documentation

Searching
Reports
API v3 | v2
Use Cases

URL, IP address, domain, or file hash

54
/ 70

54 security vendors and 1 sandbox flagged this file as malicious

ReanalyzeSimilarMore

5889bd42c5bd3bf6b1389f0eee5b3cd59180e8370eb9ea838a0b327bd5fe47

Size16.00 KB

Last Analysis Date3 hours ago

EXE

Lab01-01.exe

peexechecks-disk-spacevia-fordetect-debug-environmentidlearmadillochecks-user-inputlong-sleeps

Community Score

DETECTIONDETAILSRELATIONSBEHAVIORCOMMUNITY30

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat labeltrojan:ulise/aenjarisThreat categoriestrojanFamily labelsuliseaenjarisr002c0d020

Security vendors' analysisDo you want to automate checks?

AhnLab-V3	Trojan.Win32.Agent.C957604	Alibaba	Trojan.Win32/Aenjaris.2be749b4
ALYac	Trojan.Agent.16384SS	Antiy-AVL	Trojan.Win32.TS.Generic
Arcabit	Trojan.Ulise.D1BC1E	Avast	Win32/Malware-gen
AVG	Win32/Malware-gen	Avira (no cloud)	TR/Agent.kkbv
BitDefender	Gen.Variant.Ulise.113694	Bkav Pro	W32/AIDetect/Malware
ClamAV	Win.Malware.Agent-6342616-0	CrowdStrike Falcon	Win/malicious_confidence_100% (W)
Cylance	Unsafe	Cynet	Malicious (score: 100)
Cyren	W32/Ulise.CK.gen/Eldorado	DeepInstinct	MALICIOUS
Elastic	Malicious (high Confidence)	Emsisoft	Gen.Variant.Ulise.113694 (B)
eScan	Gen.Variant.Ulise.113694	ESET-NOD32	A Variant Of Win32/Agent.WOM
F-Secure	Trojan.TR/Agent.kkbv	Fortinet	W32/Agent.WOM.tr
GData	Gen.Variant.Ulise.113694	Google	Detected
Gridinsoft (no cloud)	Trojan.Win32.Agent.oaht1	Ikarus	Trojan.SuspectCRC
Jiangmin	Trojan.Ulise.cr	K7AntiVirus	Trojan (004b6b551)
K7GW	Trojan (004b6b551)	Lionic	Trojan.Win32.Ulise.4tc
Malwarebytes	Trojan.SystemKiller	MAX	Malware (ai Score=100)
MaxSecure	Trojan.Malware.7154915.usugen	McAfee	Generic.RXAA-AAIBB7425B82141
McAfee-GW-Edison	BehavesLike.Win32.Worm.Iz	Microsoft	Trojan.Win32/Aenjaris.CT1bit
NANO-Antivirus	Trojan.Win32.Generic.thvmhd	Rising	Trojan.Agent88.B1E (TFE.5.YR.vQ5qn2...
Sangfor Engine Zero	Trojan.Win32.Aenjaris.Vfoi	SecureAge	Malicious
Sophos	Mal/Genetic.R	Symantec	Trojan.Gen.2
Tencent	Malware.Win32.Gencirc.10b-d5711	Trellix (FireEye)	Gen.Variant.Ulise.113694
TrendMicro	TROJ_GEN.R002C0D0D20	TrendMicro-HouseCall	TROJ_GEN.R002C0D0D20
VBA32	Trojan.Tiggre	VIPRE	Gen.Variant.Ulise.113694
VirIT	Trojan.Win32.Agent5.CDE	ViRobot	Trojan.Win32.Z.Agent.16384.ADZ
Webroot	W32/Malware.Gen	Xcitium	Malware@#3eb43d99afetz
Yandex	Trojan.GenAsak.Gc9XaKYsAs	Zillya	Downloader.Amonelize.Win32.3112
Acronis (Static ML)	Undetected	Baidu	Undetected
BitDefender Theta	Undetected	CMC	Undetected
DrWeb	Undetected	Kaspersky	Undetected
Palo Alto Networks	Undetected	Panda	Undetected
QuickHeal	Undetected	SentinelOne (Static ML)	Undetected
SUPERAntiSpyware	Undetected	TACHYON	Undetected
TEHTRIS	Undetected	Trapmine	Undetected
ZoneAlarm by Check Point	Undetected	Zoner	Undetected
Cybereason	Timeout	Avast-Mobile	Unable to process file type
BitDefenderFalx	Unable to process file type	Symantec Mobile Insight	Unable to process file type
Trustlook	Unable to process file type		

Virus Total

Contact Us

Get Support

How it Works

ToS | Privacy Policy

Blog | Releases

Community

Join Community

Vote and Comment

Contributors

Top Users

Community Buzz

Tools

API Scripts

YARA

Desktop Apps

Browser Extensions

Mobile App

Premium Services

Get a demo

Intelligence

Hunting

Graph

API v3 | v2

Documentation

Searching

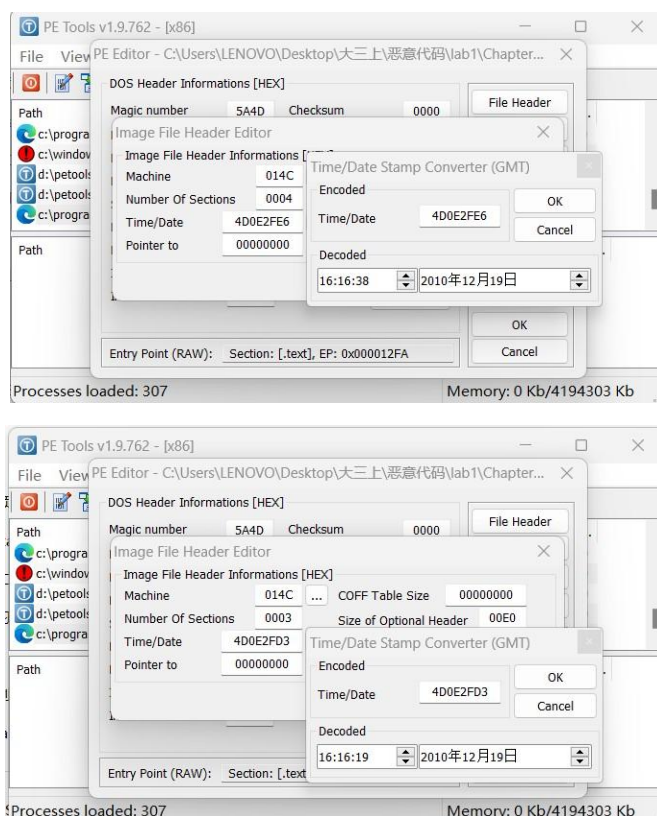
Reports

API v3 | v2

Use Cases

5

(2) 这些文件是什么时候编译的？



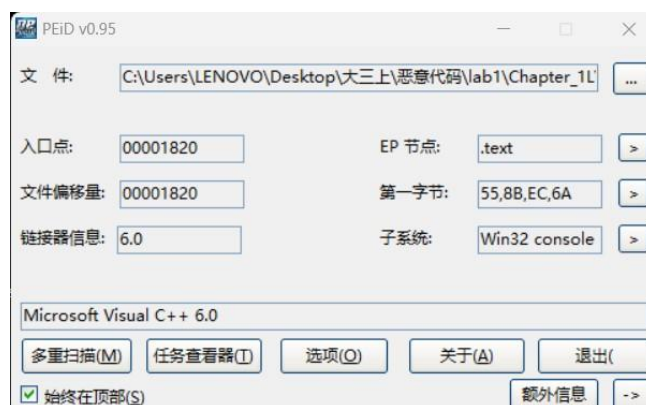
使用 PEtools 工具来打开文件。

Lab-01-01.exe 编译时间是 2010 年 12 月 19 日 16:16:19, Lab-01-01.dll 编译时间是 2010 年 12 月 19 日 16:16:38。这两个文件都是在 2010 年 12 月 19 日被编译的,两者编译时间在 1 分钟以内,基本证实了这两个文件同属一个恶意代码包。

(3) 这两个文件中是否存在迹象说明它们是否被加壳或混淆了？

如果是,这些迹象在哪里？





这两个文件的导入表数量都很少，也都有着适当大小的良好组织的文件节。

PEiD 工具标记为未加壳的代码，PEiD 正常检测出.dll 和.exe 的编译环境，EP 段是正常的.text，并且是由 Microsoft Visual C++编译的。

这两个文件并没有被加壳或混淆过的迹象。

(4) 是否有导入函数显示出了这个恶意代码是做什么的?如果是，是哪些导入函数？

	pFile	Data	Description	Value
Lab01-01.exe				
IMAGE_DOS_HEADER	00002000	00002124	Hint/Name RVA	001B CloseHandle
MS-DOS Stub Program	00002004	00002132	Hint/Name RVA	02B0 UnmapViewOfFile
IMAGE_NT_HEADERS	00002008	00002144	Hint/Name RVA	01B5 IsBadReadPtr
IMAGE_SECTION_HEADER .text	0000200C	00002154	Hint/Name RVA	01D6 MapViewOfFile
IMAGE_SECTION_HEADER .idata	00002010	00002164	Hint/Name RVA	0035 CreateFileMappingA
IMAGE_SECTION_HEADER .data	00002014	0000217A	Hint/Name RVA	0034 CreateFileA
SECTION .text	00002018	00002188	Hint/Name RVA	0090 FindClose
SECTION .idata	0000201C	00002194	Hint/Name RVA	009D FindNextFileA
IMPORT Address Table	00002020	000021A4	Hint/Name RVA	0094 FindFirstFileA
IMPORT Directory Table	00002024	000021B6	Hint/Name RVA	0028 CopyFileA
IMPORT Name Table				
IMPORT Hints/Names & DLL Names				
SECTION .data				
	00002028	00000000	End of Imports	KERNEL32.dll
	0000202C	000021D0	Hint/Name RVA	0291 malloc
	00002030	000021DA	Hint/Name RVA	0249 exit
	00002034	000021EE	Hint/Name RVA	00D3 _exit
	00002038	000021F6	Hint/Name RVA	0048 _XcptFilter
	0000203C	00002204	Hint/Name RVA	0064 _p_initenv
	00002040	00002214	Hint/Name RVA	0058 _getmainargs
	00002044	00002224	Hint/Name RVA	010F _initterm
	00002048	00002230	Hint/Name RVA	0083 _setusermatherr
	0000204C	00002244	Hint/Name RVA	009D _adjust_fdiv
	00002050	00002254	Hint/Name RVA	006A _p_commode
	00002054	00002264	Hint/Name RVA	006F _p_fmode
	00002058	00002272	Hint/Name RVA	0081 _set_app_type
	0000205C	00002284	Hint/Name RVA	00CA _except_handler3
	00002060	00002298	Hint/Name RVA	00B7 _controlfp
	00002064	000022A6	Hint/Name RVA	01C1 _stricmp
	00002068	00000000	End of Imports	MSVCRT.dll

	pFile	Data	Description	Value
Lab01-01.dll				
IMAGE_DOS_HEADER	00002000	00002116	Hint/Name RVA	0296 Sleep
MS-DOS Stub Program	00002004	0000211E	Hint/Name RVA	0044 CreateProcessA
IMAGE_NT_HEADERS	00002008	00002130	Hint/Name RVA	003F CreateMutexA
IMAGE_SECTION_HEADER .text	0000200C	00002140	Hint/Name RVA	01ED OpenMutexA
IMAGE_SECTION_HEADER .idata	00002010	00002108	Hint/Name RVA	001B CloseHandle
IMAGE_SECTION_HEADER .data	00002014	00000000	End of Imports	KERNEL32.dll
SECTION .text				
SECTION .idata				
IMPORT Address Table	00002018	0000219C	Hint/Name RVA	009D _adjust_fdiv
IMPORT Directory Table	0000201C	00002192	Hint/Name RVA	0291 malloc
IMPORT Name Table	00002020	00002186	Hint/Name RVA	010F _initterm
IMPORT Hints/Names	00002024	0000217E	Hint/Name RVA	025E free
IMAGE_EXPORT_DIRECTORY	00002028	00002168	Hint/Name RVA	02C0 strcmp
SECTION .data				
SECTION .reloc				
	0000202C	00000000	End of Imports	MSVCRT.dll
	00002030	80000017	Ordinal	0017
	00002034	80000073	Ordinal	0073
	00002038	8000000B	Ordinal	000B
	0000203C	80000004	Ordinal	0004
	00002040	80000013	Ordinal	0013
	00002044	80000016	Ordinal	0016
	00002048	80000010	Ordinal	0010
	0000204C	80000003	Ordinal	0003
	00002050	80000074	Ordinal	0074
	00002054	80000009	Ordinal	0009
	00002058	00000000	End of Imports	WS2_32.dll

使用 PView 打开两个文件。

Lab01-01.dll 导入函数有 CreateProcess（创建新进程及其主线程）、Sleep（计算机程序进程，任务或线程进入休眠）、OpenMutexA（创建互斥量）、WS2_32.dll（用来支持 internet 和网络应用程序的使用）等。这个程序很可能会创建新的进程，并且调用了 WS2_32.dll 存在的联网功能。

Lab01-01.exe 导入函数有 FindFirstFileA（搜索目录中具有与特定名称或部分名称匹配的名称的文件或子目录）、FindNextFileA（继续从先前调用 FindFirstFile、FindFirstFileEx 或 FindFirstFileTransacted 函数进行文件搜索）和 CopyFileA（将现有文件复制到新文件），CreateFileMappingA（为指定文件创建或打开命名或未命名的文件映射对象）等。

这个程序很有可能在搜索文件系统、打开修改文件、并复制文件等。

（5）是否有任何其他文件或基于主机的迹象，让你可以在受感染系统上查找？

用 IDApro 查看.exe 文件字符串

Address	Length	Type	String
.rdata:004021C2	0000000D	C	KERNEL32.dll
.rdata:004021E2	0000000B	C	MSVCRT.dll
.data:00403020	0000000D	C	kernel32.dll
.data:00403030	00000005	C	.exe
.data:00403044	00000005	C	C:*
.data:0040304C	00000021	C	C:\windows\system32\kerne132.dll
.data:0040307C	0000000D	C	Lab01-01.dll
.data:0040308C	00000021	C	C:\Windows\System32\Kernel32.dll
.data:004030B0	00000027	C	WARNING_THIS_WILL_DESTROY_YOUR_MACHINE

注意到 kernel132.dll 和 kerne132.dll 的区别（1 和 l）。

kernel132.dll 文件应该是想冒充混淆为 Windows 的系统文件 kernel32.dll。

kernel132.dll 可以作为一个基于主机的迹象来发现恶意代码感染。

（6）是否有基于网络的迹象，可以用来发现受感染机器上的这个恶意代码？

用 IDApro 查看.dll 文件字符串

Address	Length	Type	String
.rdata:1000214E	0000000D	C	KERNEL32.dll
.rdata:1000215C	0000000B	C	WS2_32.dll
.rdata:10002172	0000000B	C	MSVCRT.dll
.data:10026010	00000005	C	exec
.data:10026018	00000006	C	sleep
.data:10026020	00000006	C	hello
.data:10026028	0000000E	C	127.26.152.13
.data:10026038	00000009	C	SADFHUHF

之前已发现从 kernel32.dll 导入了 CreateProcess 和 Sleep 函数，而这两个函数普遍在后门程序中使用。exec 字符串可能是通过网络来给后门程序传送命令，让它通过 CreateProcess 函数运行一个程序。sleep 字符串可能用于命令后门程序进入休眠模式。dll 文件中还包含一个私有子网 IP 地址 127.26.152.13 的字符串。

上述内容的存在预示着极有可能休眠后门连接远程。这个 ip 地址可用于识别基于网络的恶意代码感染迹象，可用于识别恶意代码。

(7) 你猜这些文件的目的是什么？

.dll 文件可能是一个后门，.exe 文件是用来安装与运行 DLL 文件的。

(二) Lab1-2

(1) 将 Lab01-02.exe 文件上传至 <http://www.VirusTotal.com/> 进行分析并查看报告。文件匹配到了已有的反病毒软件特征吗？

56

/ 71

Community Score

66 security vendors and 1 sandbox flagged this file as malicious

ReanalyzeSimilarMore

c876a332d7dd8da331cb8eee7ab7bf32752834d4b2b54eaa362674a2a48f64a6

Size3.00 KB

Last Analysis Date9 hours ago

EXE

peexe

checks-disk-space

checks-user-input

detect-debug-environment

idle

long-sleeps

upx

via-tor

DETECTION

DETAILS

RELATIONS

BEHAVIOR

COMMUNITY30

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat labeltrojan ulise/trojanclicker

Threat categoriestrojan, downloader

Family labelsulise, trojanclicker, startpage

Security vendors' analysis

Do you want to automate checks?

AhnLab-V3	Trojan/Win32_StartPage.C26214	Alibaba	TrojanClicker/Win32/Generic.47e7b5e4
ALYac	Trojan.Startpage.3072	Antiy-AVL	Trojan/Win32_SGeneric
Arcabit	Trojan.Ser.Ulise.216	Avast	Win32/Malware-gen
AVG	Win32/Malware-gen	Avira (no cloud)	TR/Downloader.Gen
Baidu	Win32/Trojan-Clicker.Agent.ad	BitDefender	Gen.Variant.Ser.Ulise.216
BitDefenderTheta	Gen.NN.Zexaf.36662.amGtaW867f	Bkav Pro	W32/AI.Detect/Malware
ClimAV	Win.Malware.Agent-6350563-0	CrowdStrike Falcon	Win/malicious_confidence_100% (W)
Cybereason	Malicious.878404	Cylance	Unsafe
Cynet	Malicious (score: 100)	Cyren	W32/Agent.DJC.gentEldorado
DeepInstinct	MALICIOUS	DrWeb	Trojan.Click3.12740
Elastic	Malicious (moderate Confidence)	Emsisoft	Gen.Variant.Ser.Ulise.216 (B)
eScan	Gen.Variant.Ser.Ulise.216	ESET-NOD32	Win32/TrojanClicker.Agent.NVM
F-Secure	Trojan.TR/Downloader.Gen	Fortinet	W32/Agent.NVMtr
GData	Gen.Variant.Ser.Ulise.216	Google	Detected
Gridinsoft (no cloud)	Trojan.Win32.Downloader.sdfs2	Ikarus	Trojan.Win32.TrojanClicker
Jiangmin	Trojan.Generic.fmq	Lionic	Trojan.Win32.Zbot.IsXA
Malwarebytes	Trojan.Agent.LUPX	MAX	Malware (ai.Score=100)
MaxSecure	Trojan.Malware.300983.susgen	McAfee	Generic.aft
McAfee-GW-Edition	Generic.aft	NANO-Antivirus	Trojan.Win32.Click3.laups
Rising	Trojan.Clicker.Agent8.13 (CLOUD)	Sangfor Engine Zero	Suspicious.Win32.Save.a
SecureAge	Malicious	Sophos	Mai/Genenc.S
Symantec	Trojan.Horse	Tencent	Malware.Win32.Gencirc.10be33c6
Trapmine	Malicious.high.ml.score	Trellix (FireEye)	Generic.mg.836343687840da0
TrendMicro	TROJ_GEN.R002C0DHD20	TrendMicro-HouseCall	TROJ_GEN.R002C0DHD20
VBA32	Trojan.Click	VIPRE	Gen.Variant.Ser.Ulise.216
VriT	Trojan.Win32.Generic.CMEY	VRobot	Trojan.Win32.S.StartPage.3072
Webroot		Xcitem	Malware@#22epuiwh9ym
Yandex	Trojan.CL.AgentISYJ1YtE/ZV4	Zillya	Trojan.Agent.Win32.1288291
Acronis (Static ML)	Undetected	CMC	Undetected
K7AntiVirus	Undetected	K7GW	Undetected
Kaspersky	Undetected	Microsoft	Undetected
Palo Alto Networks	Undetected	Panda	Undetected
QuickHeal	Undetected	SentinelOne (Static ML)	Undetected
SUPERAntiSpyware	Undetected	TACHYON	Undetected
TEHTRIS	Undetected	ZoneAlarm by Check Point	Undetected
Zoner	Undetected	Avast-Mobile	Unable to process file type
BitDefenderFalx	Unable to process file type	Symantec Mobile Insight	Unable to process file type
Trustlook	Unable to process file type		

Virus Total

Contact Us

Get Support

How It Works

ToS | Privacy Policy

Blog | Releases

Community

Join Community

Vote and Comment

Contributors

Top Users

Community Buzz

Tools

API Scripts

YARA

Desktop Apps

Browser Extensions

Mobile App

Premium Services

Get a demo

Intelligence

Graph

API v3 | v2

Documentation

Searching

Reports

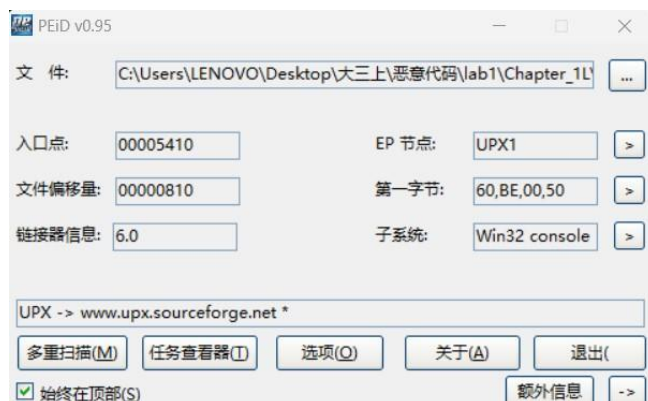
API v3 | v2

Use Cases

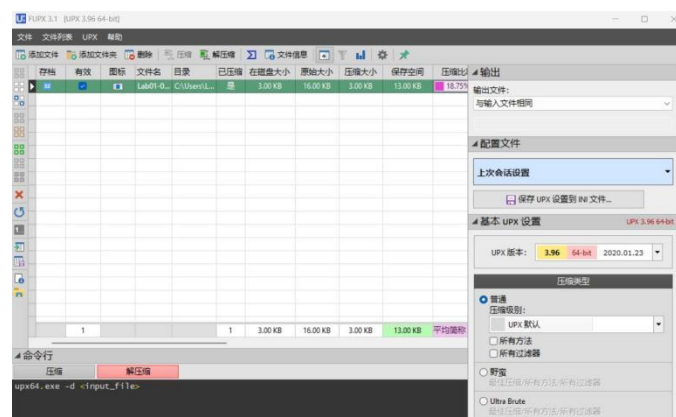
10

(2) 是否有这个文件被加壳或混淆的任何迹象?如果是这样, 这些迹象是什么?如果该文件被加壳, 请进行脱壳, 如果可能的话。

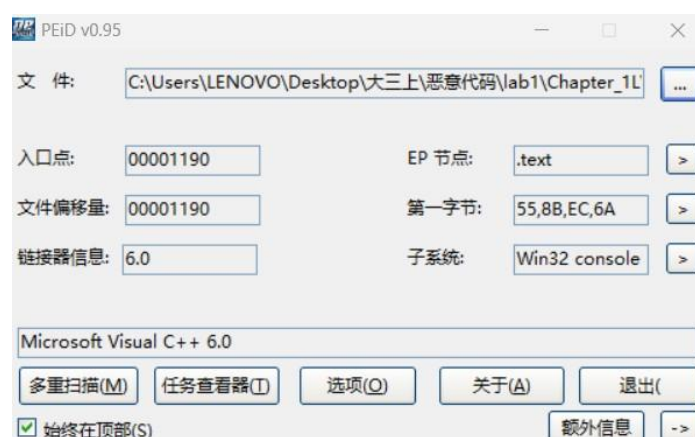
使用 PEiD 检测, 存在 UPX 壳。



使用 FUPX 进行脱壳。



使用 PEiD 检测, 已脱壳。



(3) 有没有任何导入函数能够暗示出这个程序的功能?如果是, 是哪些导入函数, 它们会告诉你什么?

rfile	Data	Description	Value
00002000	0000223C	HintName RVA	0000 CreateServiceA
00002004	0000224C	HintName RVA	0000 StartServiceCtrlDispatcherA
00002008	00002254	HintName RVA	0000 OpenSCManagerA
0000200C	00000000	End of imports	ADVAPI32.dll
00002010	0000219E	HintName RVA	0000 SystemTimeToFileTime
00002014	000021B4	HintName RVA	0000 GetModuleInformationA
00002018	000021C8	HintName RVA	0000 CreateWaitableTimerA
0000201C	000021D2	HintName RVA	0000 ExitProcess
00002020	000021EC	HintName RVA	0000 OpenMutexA
00002024	000021F8	HintName RVA	0000 SetWaitableTimer
00002028	00002204	HintName RVA	0000 WaitForSingleObject
0000202C	00002228	HintName RVA	0000 CreateMutexA
00002030	0000222E	HintName RVA	0000 CreateThread
00002034	00000000	End of imports	KERNEL32.DLL
00002038	0000227A	HintName RVA	0000 _init
0000203C	00002282	HintName RVA	0000 _xcpfilter
00002040	00002292	HintName RVA	0000 _end
00002044	00002296	HintName RVA	0000 _p_initenv
00002048	000022A6	HintName RVA	0000 _getmainargs
0000204C	000022B6	HintName RVA	0000 _initterm
00002050	000022C2	HintName RVA	0000 _setusermatherr
00002054	000022D4	HintName RVA	0000 _adjust_fiber
00002058	000022E2	HintName RVA	0000 _p__coroutine
0000205C	000022F0	HintName RVA	0000 _p__mode
00002060	000022FC	HintName RVA	0000 _wrtmp_type
00002064	0000230C	HintName RVA	0000 _except_handler3
00002068	0000231E	HintName RVA	0000 _coroutp
0000206C	00000000	End of imports	MSVCRT.dll
00002070	0000232A	HintName RVA	0000 InternetOpenUrlA
00002074	0000233C	HintName RVA	0000 InternetOpenA
00002078	00000000	End of imports	INTERNET.DLL

使用 PEView 打开已脱壳的.exe 文件，kernel32.dll 和 msvcrt.dll，而
这些函数几乎被每个程序都导入，关于恶意代码的信息不多。

advapi32.dll 的导入函数有 CreateServiceA（创建服务对象并将其添加到指定的服务控制管理器数据库）、StartServiceCtrlDispatcherA（将服务进程的 main 线程连接到服务控制管理器）、OpenSCManagerA（与指定计算机上的服务控制管理器建立连接，并打开指定的服务控制管理器数据库）函数，这三个函数一般可用于创建互斥、进程、服务。这个恶意代码很可能会创建一个服务。

从 wininet.dll 的导入函数有 InternetOpenUrlA（打开由完整的 FTP 或 HTTP URL 指定的资源）、InternetOpenA（初始化应用程序对 WinINet 函数的使用）函数，这两个函数一般可用于进行联网操作，启动一个链接。这个恶意代码很可能将会进行联网操作。

综上所述，这个恶意代码很可能会创建一个服务并进行联网操作。

（4） 哪些基于主机或基于网络的迹象可以被用来确定被这个恶意代码所感染的机器？

Address	Length	Type	String
.rdata:0040216C	0000000D	C	KERNEL32.DLL
.rdata:00402179	0000000D	C	ADVAPI32.dll
.rdata:00402186	0000000B	C	MSVCRT.dll
.rdata:00402191	0000000C	C	WININET.dll
.data:00403010	0000000B	C	MalService
.data:0040301C	0000000B	C	Malservice
.data:00403028	00000007	C	HGL345
.data:00403030	00000023	C	http://www.malwareanalysisbook.com
.data:00403054	00000016	C	Internet Explorer 8.0

用 IDApro 查看 .exe 文件字符串,看到了所创建的服务名称 Malservice。还有 www.malwareanalysisbook.com, 这可能是 InternetOpenURL 函数中所打开的 URL。

可以通过名为 Malservice 的服务,通过到 <http://www.malwareanalysisbook.com> 的网络流量,来检查被恶意代码感染的主机。

(三) Lab1-3

(1) 将 Lab01-03.exe 文件上传至 <http://www.VirusTotal.com/> 进行分析并查看报告。文件匹配到了已有的反病毒软件特征吗?

7963a582039924c70e3da2da80fd3352ebc90de7b8c4c427d484f4f050f0aec

61

71

Community Score

61 security vendors and no sandboxes flagged this file as malicious

Reanalyze

Similar

More

7963a582039924c70e3da2da80fd3352ebc90de7b8c4c427d484f4f050f0aec

Size

Last Analysis Date

Lab01-03.exe

4.64 KB

1 day ago

EXE

peexe

fsg

checks-user-input

overlay

runtime-modules

detect-debug-environment

long-sleeps

direct-cpu-clock-access

via-tor

DETECTION

DETAILS

RELATIONS

BEHAVIOR

COMMUNITY 30 +

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label

trojan.graftor.genome

Threat categories

trojan

spyware

Family labels

graftor

genome

agentb

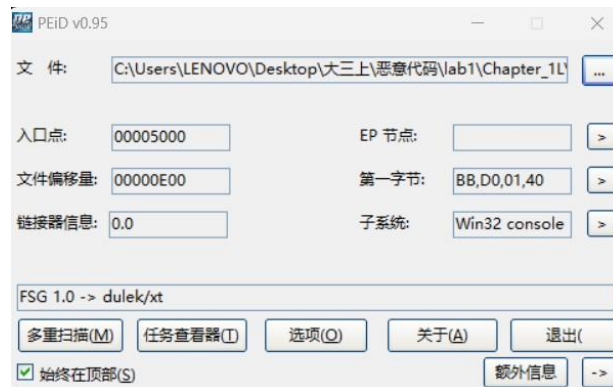
Security vendors' analysis

Do you want to automate checks?

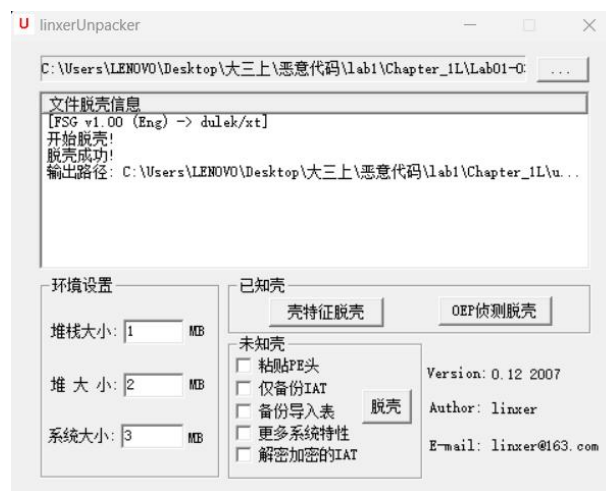
AhnLab-V3	Trojan.Win.Generic.R427327	Alibaba	TrojanClicker.Win32/Tnega.79c8a6fb
ALYac	Gen.Variant.Graftor.968808	Antiy-AVL	Trojan.Win32.S.Generic
Arcabit	Trojan.Graftor.DEC968	Avast	Win32/Malware-gen
AVG	Win32/Malware-gen	Baidu	Win32/TrojanClicker.Agent.z
BitDefender	Gen.Variant.Graftor.968808	BitDefenderTheta	Gen.NN.ZexaF.36662.ambdaODfLcf
Bkav Pro	W32/AIDetect/Malware	ClimAV	Win/Malware.Emoneg.9937593-0
CrowdStrike Falcon	Win/malicious_confidence_100% (W)	Cybereason	Malicious.431146
Cylance	Unsafe	Cynet	Malicious (score: 100)
Cyren	W32/SuspPack.DH.gen/Eldorado	DeepInstinct	MALICIOUS
DrWeb	Trojan.Click2.16518	Elastic	Malicious (high Confidence)
Emsisoft	Gen.Variant.Graftor.968808 (B)	eScan	Gen.Variant.Graftor.968808
ESET-NOD32	Win32/TrojanClicker.Agent.NVN	Fortinet	W32/WebDown.E76Atr
GData	Gen.Variant.Graftor.968808	Google	Detected
Ikarus	Trojan.Win32.Genome	Jiangmin	Trojan/Genome.bmp
K7AntiVirus	Spyware (0055e3861)	K7GW	Spyware (0055e3861)
Kaspersky	Trojan.Win32.Agentb.bquu	Lionic	Trojan.Multi.Generic.IVbD
Malwarebytes	Trojan.Agent.MWL	MAX	Malware (ai Score=100)
MaxSecure	Trojan.Malware.1728101.susgen	McAfee	GenericRXXAA-FA9C5C27494C28
McAfee-GW-Editon	BehavesLike.Win32.Dropper.xz	Microsoft	Trojan.Win32/TnegaIMSR
NANO-Antivirus	Trojan.Win32.Inor.geljo	Rising	Trojan.Proxy.Win32.Small.gs (CLASSIC)
Sangfor Engine Zero	Trojan.Win32.Clicker.Vn8a	SentinelOne (Static ML)	Static AI - Malicious PE
Sophos	Mal/Packer	Symantec	ML.Attribute.HighConfidence
TACHYON	Trojan/W32.Small.4752.C	TEHTRIS	Generic.Malware
Tencent	Malware.Win32.Gencirc.115d78c9	Trapmine	Malicious high ml score
Trellix (FireEye)	Generic.mg.9c5c27494c28ed9b	TrendMicro	TROJ_SPNR.30E214
TrendMicro-HouseCall	TROJ_SPNR.30E214	VBA32	Trojan.Wacatalc
VIPRE	Gen.Variant.Graftor.968808	VriT	Trojan.Win32.Generic.APWM
ViRobot	Trojan.Win32.Z.Genome.4752	Webroot	W32.Genome.Ssrc
Xcitium	TrojWare.Win32.Trojan.Inor.B_10@1qra8i	Yandex	Trojan.GenomeIgszR3auxbA
Zillya	Trojan.Genome.Win32.112441	ZoneAlarm by Check Point	Trojan.Win32.Agentb.bquu
Zoner	Probably Heur.Exe/HeaderL	Acronis (Static ML)	Undetected
Avira (no cloud)	Undetected	CMC	Undetected
F-Secure	Undetected	Gridinsoft (no cloud)	Undetected
Palo Alto Networks	Undetected	Panda	Undetected
QuickHeal	Undetected	SecureAge	Undetected
SUPERAntiSpyware	Undetected	Avast-Mobile	Unable to process file type
BitDefenderFalx	Unable to process file type	Symantec Mobile Insight	Unable to process file type
Trustlook	Unable to process file type		

(2) 是否有这个文件被加壳或混淆的任何迹象?如果是这样, 这些迹象是什么?如果该文件被加壳, 请进行脱壳, 如果可能的话。

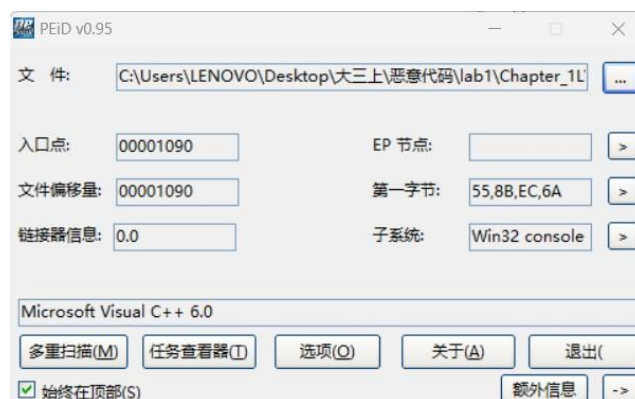
14



使用 PEiD 检测，可能存在 FSG 壳

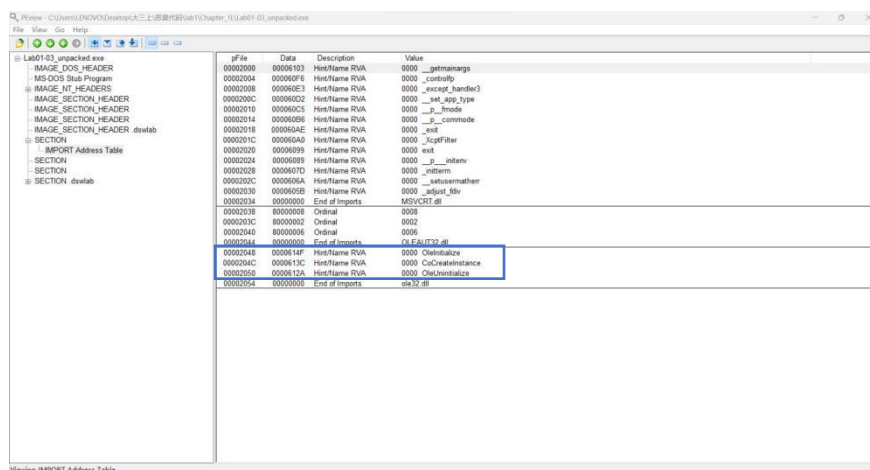


使用 linxerUnpacker 进行脱壳。



使用 PEiD 检测，已脱壳。

(3) 有没有任何导入函数能够暗示出这个程序的功能?如果是，是哪些导入函数，它们会告诉你什么?



使用 PEView 打开脱壳后的.exe 文件,发现 OleInitialize (初始化当前单元的 COM 库)、CoCreateInstance (创建并默认初始化与指定 CLSID 关联的类的单个对象)、OleUninitialize (关闭单元上的 COM 库,并释放相关对象及资源) 三个函数。

推测该程序可以通过 COM 接口访问一个网址。

(4) 有哪些基于主机或基于网络的迹象,可以被用来确定被这个恶意代码所感染的机器?

```
--p___initenv
__initterm
__setusermatherr
__adjust_fdiv
http://www.malwareanalysisbook.com/ad.html
H @
```

使用 strings.exe 查看 Lab01-03_unpacked.exe 字符串信息,可以看到一个网址,因此该恶意代码应该是利用 ole 相关组件实现了对该网址的访问,监视网络行为。

若有访问该网址的行为,则该机器应已被恶意代码感染。

(四) Lab1-4

(1) 将 Lab01-04.exe 文件上传至 <http://www.VirusTotal.com/> 进行分析并查看报告。文件匹配到了已有的反病毒软件特征吗?

0fa1498340fca6c562cfa389ad3e93395f44c726d128d7ba08579a69aaf3b126

59

71

59 security vendors and 2 sandboxes flagged this file as malicious

0fa1498340fca6c562cfa389ad3e93395f44c726d128d7ba08579a69aaf3b126

Lab01-04.exe

Size36.00 KB

Last Analysis Date6 hours ago

EXE

peexe

ide

checks-user-input

armadillo

via-tor

Community Score

DETECTION

DETAILS

RELATIONS

BEHAVIOR

COMMUNITY30+

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat labeltrojan.cerbu/gofot

Threat categories

trojan

downloader

dropper

Family labels

cerbu

gofot

didr

Security vendors' analysis

Do you want to automate checks?

Alibaba	TrojanDownloader.Win32/Gofot.7e5f679f	ALYac	Gen Variant Cerbu 64762
Anliy-AVL	Trojan[Downloader]Win32.AGeneric	Arcabit	Trojan.Cerbu.DFD0E
Avast	Win32.DropperX-gen [Drip]	AVG	Win32.DropperX-gen [Drip]
Avira (no cloud)	TR/Dldr.Small.romlh	BitDefender	Gen Variant Cerbu 64762
BitDefender Theta	AI.Packer.5911D1B71F	Bkav Pro	W32.AI.DetectMalware
ClamAV	Win.Trojan.Agent-375080	CrowdStrike Falcon	Win/malicious_confidence_100% (W)
Cybereason	Malicious.106dd2	Cylance	Unsafe
Cynet	Malicious (score: 100)	Cyren	W32/Heuristic-217EIdorado
DeepInstinct	MALICIOUS	DrWeb	Trojan.Downloader5.60705
Elastic	Malicious (high Confidence)	Emsisoft	Gen Variant Cerbu 64762 (B)
eScan	Gen Variant Cerbu 64762	ESET-NOD32	Win32/TrojanDownloader.Small.BFX
F-Secure	Trojan.TR/Dldr.Small.romlh	Fortinet	W32/Small.BFXldr.didr
GData	Gen Variant Cerbu 64762	Google	Detected
Ikarus	Backdoor.Win32.SuspectCRC	Jiangmin	Trojan/Invader.cph
K7AntiVirus	Trojan-Downloader (005663e81)	K7GW	Trojan-Downloader (005663e81)
Kaspersky	HEUR:Trojan.Win32.Gofot.gen	Lionic	Trojan.Win32.Genome.ts0c
Malwarebytes	Small.Trojan.Downloader.DDS	MAX	Malware (ai Score=100)
MaxSecure	Trojan.Malware.23478.susgen	McAfee	GenericRXXEW-DZ625AC05FD47A
McAfee-GW-Editon	GenericRXXEW-DZ625AC05FD47A	Microsoft	TrojanDownloader.Win32.SmallIMSR
NANO-Antivirus	Trojan.Win32.Kazy.cwxml	Rising	Downloader.SmallB41 (TFE 5 KpgWR ...
Sangfor Engine Zero	Suspicious.Win32.Save.ins	SecureAge	Malicious
Sophos	Mal/Generic-R	SUPERAntiSpyware	Trojan.Agent/Gen-Downloader
Symantec	ML.Attribute.HighConfidence	TACHYON	Trojan-Downloader/W32.Agent.36864.ADU
Tencent	Malware.Win32.Gencirc.10b0badc	Trapmine	Malicious.high.mf.score
Trellix (FireEye)	Generic.mg.625ac05f47adc3c	TrendMicro	Mal_DLDER
TrendMicro-HouseCall	Mal_DLDER	VBA32	BScope.Trojan.Downloader
VIPRE	Gen Variant Cerbu 64762	VriT	Trojan.Win32.Generic.BAGU
Webroot	W32.Trojan.Gen	Xcitem	Malware@#2cyf5gBq6fqyr
Yandex	Trojan.DL.SmallIo4/0V8aERQ	Zillya	Downloader.Small.Win32.47818
ZoneAlarm by Check Point	HEUR:Trojan.Win32.Gofot.gen	Acronis (Static ML)	Undetected
AhnLab-V3	Undetected	Baidu	Undetected
CMC	Undetected	Gridinsoft (no cloud)	Undetected
Palo Alto Networks	Undetected	Panda	Undetected
QuickHeal	Undetected	SentinelOne (Static ML)	Undetected
TEHTRIS	Undetected	VRobot	Undetected
Zoner	Undetected	Avast-Mobile	Unable to process file type
BitDefenderFalx	Unable to process file type	Symantec Mobile insight	Unable to process file type
Trustlook	Unable to process file type		

VirusTotal

Contact Us

Get Support

How It Works

ToS | Privacy Policy

Blog | Releases

Community

Join Community

Vote and Comment

Contributors

Top Users

Community Buzz

Tools

API Scripts

YARA

Desktop Apps

Browser Extensions

Mobile App

Premium Services

Get a demo

Intelligence

Hunting

Graph

API v3 | v2

Documentation

Searching

Reports

API v3 | v2

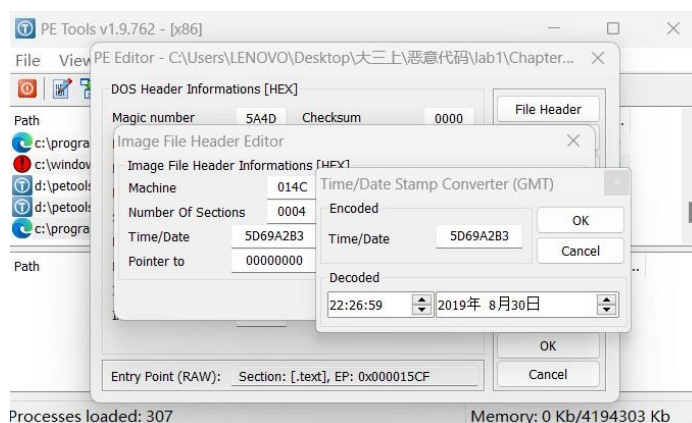
Use Cases

(2) 是否有这个文件被加壳或混淆的任何迹象?如果是这样, 这些迹象是什么?如果该文件被加壳, 请进行脱壳, 如果可能的话。



没有迹象显示这个文件是被加过壳或混淆过的。

(3) 这个文件是什么时候被编译的?



使用 PEtools 工具来打开文件。

根据文件头的信息, 这个文件是在 2019 年 8 月编译的。但推测该编译时间应该是伪造的, 仅根据当前信息还不能确定这个文件到底是什么时候被编译的。

(4) 有没有任何导入函数能够暗示出这个程序的功能?如果是, 是哪些导入函数, 它们会告诉你什么?

Offset	File	Import Name	Value
00002000	00002000	00002000	0142 OpenProcessToken
00002004	00002004	00002004	00F5 LookupPrivilegeValueA
00002008	00002008	00002008	0017 AdjustTokenPrivileges
0000200C	0000200C	0000200C	00000000
00002010	00002010	00002010	015C GetProcAddress
00002014	00002014	00002014	01C2 LoadLibraryA
00002018	00002018	00002018	02D3 WinExec
0000201C	0000201C	0000201C	02DF WriteFile
00002020	00002020	00002020	0034 CreateFileA
00002024	00002024	00002024	0295 SizeofResource
00002028	00002028	00002028	0046 CreateRemoteThread
0000202C	0000202C	0000202C	00A3 FindResourceA
00002030	00002030	00002030	0126 GetModuleHandleA
00002034	00002034	00002034	017D GetWindowsDirectoryA
00002038	00002038	00002038	01D0 MoveFileA
0000203C	0000203C	0000203C	0165 GetTempPathA
00002040	00002040	00002040	00F7 GetCurrentProcess
00002044	00002044	00002044	01EF OpenProcess
00002048	00002048	00002048	001B CloseHandle
0000204C	0000204C	0000204C	01C7 LoadResource
00002050	00002050	00002050	End of Imports
00002054	00002054	00002054	KERNEL32.dll
00002058	00002058	00002058	014E _signif
0000205C	0000205C	0000205C	00D3 _init
00002060	00002060	00002060	0048 _XcpFilter
00002064	00002064	00002064	0243 _init
00002068	00002068	00002068	0054 _getmainargs
0000206C	0000206C	0000206C	016F _initterm
00002070	00002070	00002070	0083 _setusermatherr
00002074	00002074	00002074	009D _adjust_fdiv
00002078	00002078	00002078	00A4 _J__commode
0000207C	0000207C	0000207C	006F _J__fmode
00002080	00002080	00002080	0081 _set_app_type
00002084	00002084	00002084	00CA _except_handler3
00002088	00002088	00002088	00B7 _controlfp
0000208C	0000208C	0000208C	01C1 _stricmp
00002090	00002090	00002090	End of Imports
00002094	00002094	00002094	MSVCRT.dll

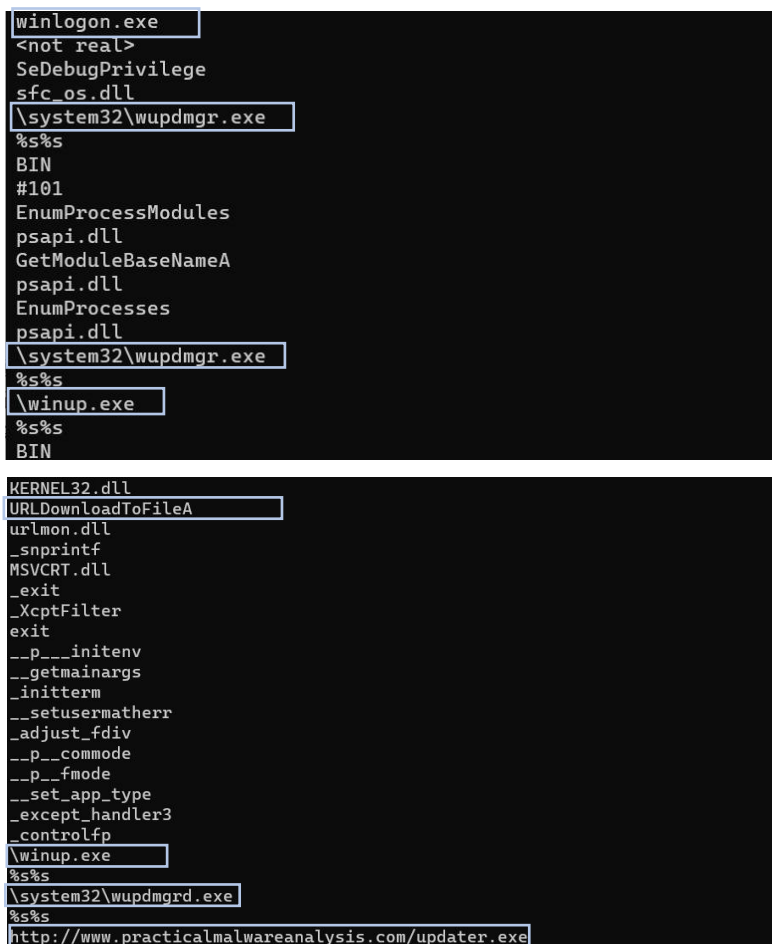
从 advapi32.dll 有 3 个导入函数，OpenProcessToken（打开与进程关联的访问令牌）、LookupPrivilegeValueA（检索本地唯一标识符（LUID）指定系统上用于本地表示指定特权名称）、AdjustTokenPrivileges（启用或禁用指定访问令牌中的特权），这 3 个函数可能和系统的权限有关。

从 kernel32.dll 中有 16 个导入函数，GetProcAddress（从指定的动态链接库 DLL 检索导出函数或变量的地址）、LoadLibraryA（将指定的模块加载到调用进程的地址空间中）、WinExec（运行指定的应用程序）、WriteFile（将数据写入指定的文件或输入/输出 I/O 设备）、CreateFileA（创建或打开文件或 I/O 设备）、SizeofResource（以字节为单位检索指定资源的大小）、CreateRemoteThread（创建在另一个进程的虚拟地址空间中运行的线程）、FindResourceA（确定指定模块中具有指定类型和名称的资源的位置）、GetModuleHandleA（检索指定模块的模块句柄）、GetWindowsDirectoryA（检索 Windows 目录的路径）、MoveFileA（移动现有文件或目录，包括其子级）、GetTempPathA（检索为临时文件指定的目录的路径）、GetCurrentProcess（检索当前进程的伪句柄）、OpenProcess（打开现有的本地进程对象）、CloseHandle（关闭打开的对象句柄）、LoadResource（检索可用于获取指向内存中指定资源第一个字节的指针的句柄）。

可以看出这个程序不仅涉及到系统权限，还涉及到文件的读写、资源的查找等。假设它试图访问使用了特殊权限进行保护的文件，这个程序从资源节中装载数据 (LoadResource、FindResource、SizeofResource)，并写一个

文件到磁盘上(CreateFile、WriteFile),接着执行一个磁盘上的文件(WinExec),并调用 GetWindowsDirectory 将文件写入到了系统目录。

(5) 有哪些基于主机或基于网络的迹象,可以被用来确定被这个恶意代码所感染的机器?



The image shows two screenshots of a command prompt window displaying the output of the strings.exe tool. The first screenshot shows the following strings: winlogon.exe, <not real>, SeDebugPrivilege, sfc_os.dll, \system32\wupdmgr.exe, %s%s, BIN, #101, EnumProcessModules, psapi.dll, GetModuleBaseNameA, psapi.dll, EnumProcesses, psapi.dll, \system32\wupdmgr.exe, %s%s, \winup.exe, %s%s, and BIN. The second screenshot shows the following strings: KERNEL32.dll, URLDownloadToFileA, urlmon.dll, _snprintf, MSVCRT.dll, _exit, _XcptFilter, _exit, __p__initenv, __getmainargs, _initterm, __setusermatherr, _adjust_fdiv, __p__commode, __p__fmode, __set_app_type, _except_handler3, _controlfp, \winup.exe, %s%s, \system32\wupdmgrd.exe, %s%s, and http://www.practicalmalwareanalysis.com/updater.exe. In both screenshots, several strings are highlighted with a yellow box.

用 strings.exe 查看 Lab01-04.exe 文件字符串,出现了很多 dll,在系统目录下出现了一些 exe 文件。

winlogon.exe 是 Windows NT 用户登陆程序,用于管理用户登录和退出;wupdmgr.exe 是大多数 Windows 系统的自动升级程序,但病毒木马也经常将自己伪装成这两个 exe 来运行。结合 GetWindowsDirectory 函数调用,表明恶意代码在 C:\Windows\System32\wupdmgr.exe 位置创建或者修改了一个文件。

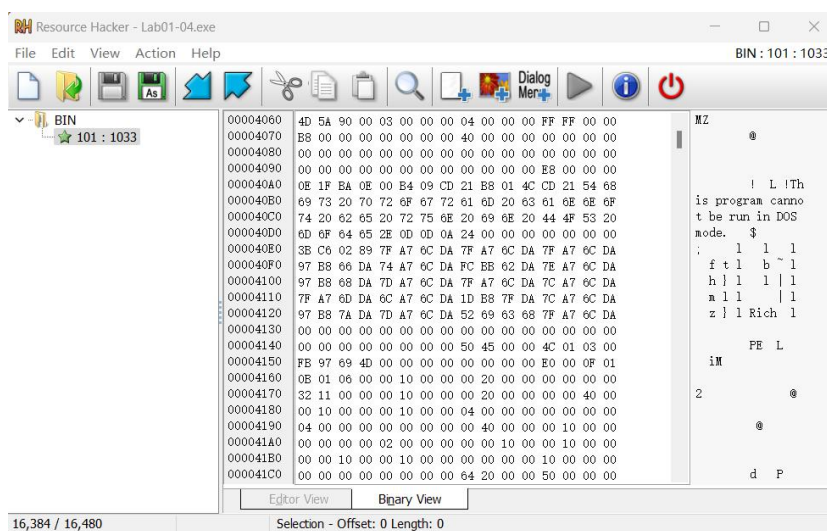
winup.exe 不是 Windows 系统自带的 exe 文件,而是至今仍被广泛使用的 infector 病毒技术的被感染文件。推测该恶意代码的资源节中存放了 winup.exe。

导入函数 URLDownloadToFileA（从互联网下载并将其保存到文件，可以与 winexec 结合使用）和一个网址

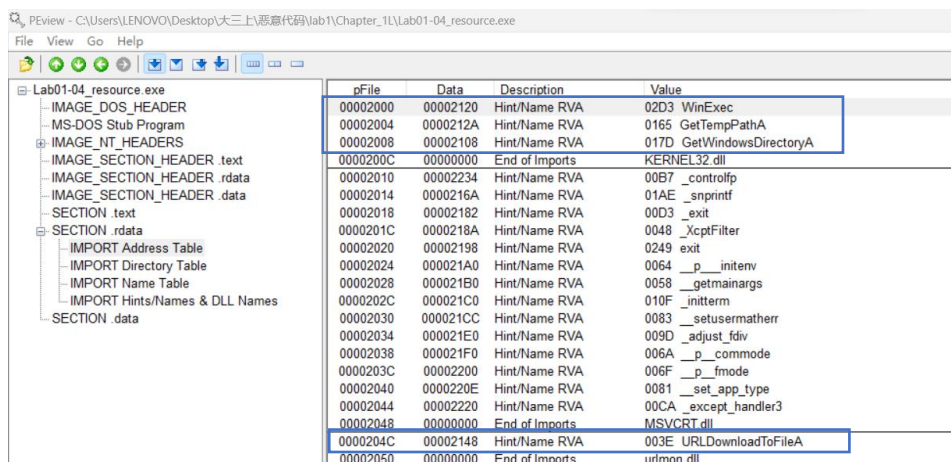
“http://www.practicalmalwareanalysis.com/updater.exe”表示这个 exe 文件将从该网址下载 updater.exe。

综上所述，如果电脑中出现了 winup.exe，在 C:\Windows\system32 文件夹下出现了 wupdmgrd.exe，则该机器被恶意代码所感染。监视网络行为，如果有向“hxxp://www.practicalmalwareanalysis.com/updater.exe”发送请求，则该机器被恶意代码所感染。

（6）这个文件在资源段中包含一个资源。使用 Resource Hacker 工具来检查资源，然后抽取资源。从资源中你能发现什么吗？



单击 Action→Save *.bin resource..., 将文件存储为.exe 格式，并在 PEvent 中打开查看导入表。



kernel32.dll 文件中有 3 个导入函数,WinExec(运行指定的应用程序)、GetTempPathA(检索为临时文件指定的目录的路径)、GetWindowsDirectoryA(检索 Windows 目录的路径)。urlmon.dll 文件中有 URLDownloadToFileA(从互联网下载并将其保存到文件中)。

综上所述,嵌入文件通过访问一些网络函数,调用了一个由恶意下载器普遍使用的函数 URLDownloadToFile 和 WinExec 函数下载了 updater.exe,并且还可能执行了下载到的文件。

(五) Lab1-5

(1) 对 Lab1 和 Lab3 的样本编写 Yara 规则。

Lab1:

1	rule Lab1
2	{
3	meta:
4	description = "rules for Lab1 "
5	date = "202x/xx/xx"
6	strings:
7	\$a = "kernel32.dll" wide ascii
8	\$b = "127.26.152.13" wide ascii
9	\$c = "http://www.malwareanalysisbook.com" wide ascii
10	\$d = "wupdmgr" wide ascii
11	condition:
12	any of them
13	}

Lab3:

1	rule Lab3
2	{
3	meta:

4	description = "rules for Lab3 "
5	date = "202x/xx/xx"
6	strings:
7	\$a = "vmx32to64" wide ascii
8	\$b = "serve.html" wide ascii
9	\$c = "http://www.malwareanalysisbook.com" wide ascii
10	\$d = "svchost" wide ascii
11	\$e = "practicalmalwareanalysis.log" wide ascii
12	condition:
13	any of them
14	}

(2) 使用自己的规则对自己电脑的 C 盘进行 Yara 引擎的扫描，记录扫描所用时间。

下载并安装好 yara、python 后，使用 pip install yara-python 命令安装 Yara Python 模块。

Python 脚本：

1	import os
2	import time
3	import yara
4	
5	# 载入 Yara 规则
6	rules = yara.compile('lab1.yara')# lab3 使用的文件更换为 lab3.yara
7	
8	# 指定要扫描的目录
9	target_directory = 'C:\\'
10	
11	start_time = time.time()

12	
13	<code>matches = []</code>
14	
15	<code># 遍历目录中的所有文件</code>
16	<code>for root, dirs, files in os.walk(target_directory):</code>
17	<code> for file in files:</code>
18	<code> file_path = os.path.join(root, file)</code>
19	<code> try:</code>
20	<code> # 执行扫描</code>
21	<code> file_matches = rules.match(file_path)</code>
22	<code> if file_matches:</code>
23	<code> for match in file_matches:</code>
24	<code> matches.append(match)</code>
25	<code> except Exception as e:</code>
26	<code> pass</code>
27	
28	<code>end_time = time.time()</code>
29	
30	<code># 输出匹配结果</code>
31	<code>if matches:</code>
32	<code> for match in matches:</code>
33	<code> print(f"匹配到规则: {match.rule}")</code>
34	<code>else:</code>
35	<code> print("未发现匹配的规则")</code>
36	
37	<code># 计算并输出扫描时间</code>
38	<code>scan_time = end_time - start_time</code>
39	<code>print(f"扫描完成, 用时: {scan_time} 秒")</code>

Lab1 扫描用时:

```
匹配到规则: Lab1  
匹配到规则: Lab1  
匹配到规则: Lab1  
匹配到规则: Lab1  
匹配到规则: Lab1  
扫描完成, 用时: 572.7772219181061 秒
```

Lab3 扫描用时:

```
匹配到规则: Lab3  
匹配到规则: Lab3  
匹配到规则: Lab3  
匹配到规则: Lab3  
扫描完成, 用时: 592.3735899925232 秒
```

(3) 讨论哪些 yara 条件执行效率高, 哪些 yara 条件执行效率低, 以及如何改进那些执行效率低的 yara 条件。

A) 执行效率高

- a) 使用静态规则
- b) 使用必要的规则, 精简规则的数目
- c) 使用较少较短的字符串并且只在必要时才进行规则匹配
- d) 条件简单, 使用较少的逻辑运算符和条件嵌套

B) 执行效率低

- a) 使用动态规则
- b) 使用规则的数量过大, 且其中存在冗余的规则
- c) 使用了大量较长的字符串进行规则匹配
- d) 使用了大量逻辑运算符、通配符 (如*和?)、较为复杂的正则表达式 (如具有多个嵌套、条件判断等) 等使用过多复杂条件的规则
- e) 扫描大型二进制文件, 文件过大

C) 如何改进

- a) 相似的规则进行合并, 减少规则数目
- b) 简化规则条件, 减少逻辑运算符等的使用
- c) 减少不必要通配符和正则表达式的使用
- d) 精确匹配范围, 减少不必要的计算

- e) 优化规则顺序，将最可能匹配的规则放在前面
- f) 缓存已匹配的结果，减少重复运算

四、实验结论及心得体会

（一）实验结论

1. Lab1-1 实验结论：通过 VirusTotal 和相关工具的分析，发现 Lab01-01.exe 和 Lab01-01.dll 是在 2010 年 12 月 19 日编译的，没有明显的加壳或混淆迹象，但存在对文件进行操作、创建新的进程并与互联网通信的迹象，可能是一个后门程序。

2. Lab1-2 实验结论：Lab01-02.exe 经过脱壳后，发现了 UPX 壳，还有与互联网通信的迹象，可能是一个恶意下载器。

3. Lab1-3 实验结论：Lab01-03.exe 存在 FSG 壳，使用脱壳工具后得到清晰代码。导入函数暗示它可以通过 COM 接口访问网址，可能涉及到下载和执行文件。

4. Lab1-4 实验结论：Lab01-04.exe 并没有加壳或混淆，但存在一些与系统权限和网络通信相关的导入函数，可能涉及到文件操作和下载。

5. Lab1-5 实验结论：编写了 Yara 规则并在 C 盘上进行了扫描，可以用于检测与恶意代码相关的特征。

（二）心得体会

1. 通过本次实验，学习了如何使用不同的工具和技术来分析恶意代码，包括 VirusTotal 网站、PEview、脱壳工具等。这些工具对于恶意代码分析非常有用，可以帮助我们识别潜在的威胁。

2. 实验中还练习了如何观察恶意代码中的迹象，例如编译时间、导入函数、字符串等，这些信息可以帮助理解恶意代码的功能和行为。

3. 编写 Yara 规则帮助巩固了课堂的知识，可以帮助我们自动化恶意代码检测的过程，并及时发现潜在的威胁。

4. 学到了优化 Yara 规则的重要性，以提高执行效率。合并相似的规则、简化条件、减少不必要的通配符和正则表达式都是优化规则的有效方法。

5. 本次实验提供了有关恶意代码分析的基本知识和技能，使我能够更好地理解和应对潜在的安全威胁。