

南开大学

恶意代码分析与防治技术课程实验报告

实验一：病毒监测与 **yara** 的使用



学 院____计网_____

专 业____信息安全_____

学 号____2111033_____

姓 名____艾明旭_____

班 级____信息安全一班_____

一、实验目的

学会使用 windows 虚拟机在隔离环境下对计算机病毒进行分析与防治。
学会使用 PEid 工具进行分析加壳与脱壳,同时提前了解 FGS 的相关知识。
学会使用 PEview 工具进行二进制文件的分析。
学会使用 IDapro 工具对文件的字符串和链接的函数进行分析。
学会使用 upx 工具对文件在 linux 和 windows 环境下进行加壳与脱壳的操作。
学会使用 yara 工具对相关的文件编写 yara 规则进行目标文件的监测。
学会优化 yara 规则,并且通过 powershell 监测到 yara 监测速度的变化。

二、实验原理

1、 实验环境

Windows10, VMWARE, Windows11

2、 实验工具

STRINGS, IDAPro,PEVIEW, YARA, upx,PEID,powershell

3、 原理

Yara 是 virustotal 发布的一个开源恶意代码查杀引擎,可以用来:
识别恶意代码
查杀恶意代码
在 <https://virustotal.github.io/yara/>上可以上传相应的恶意代码,并且生成相关的报告。

三、实验过程

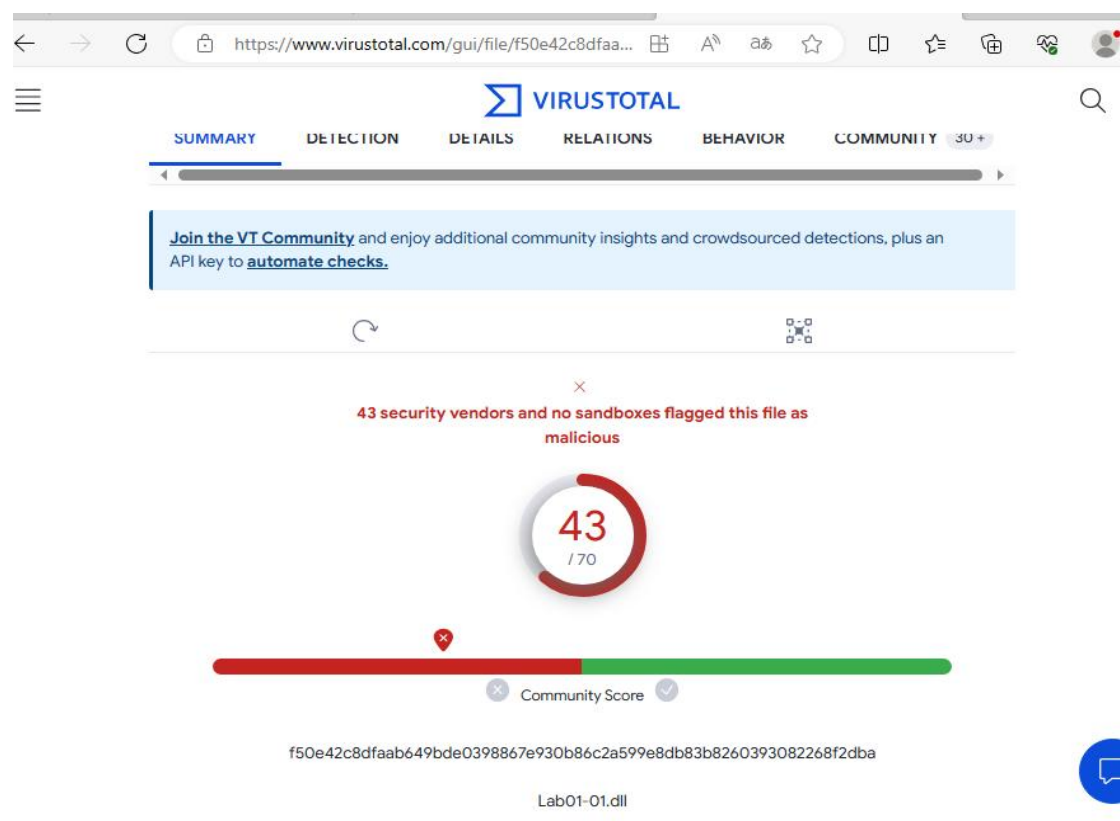
- 1、 实验为了避免病毒感染主机,在 windows 虚拟机环境当中运行,需要我们将老师提供的各种工具移动到虚拟机当中运行。

Lab 1-1

1. Upload the files to <http://www.VirusTotal.com/> and view the reports. Does either

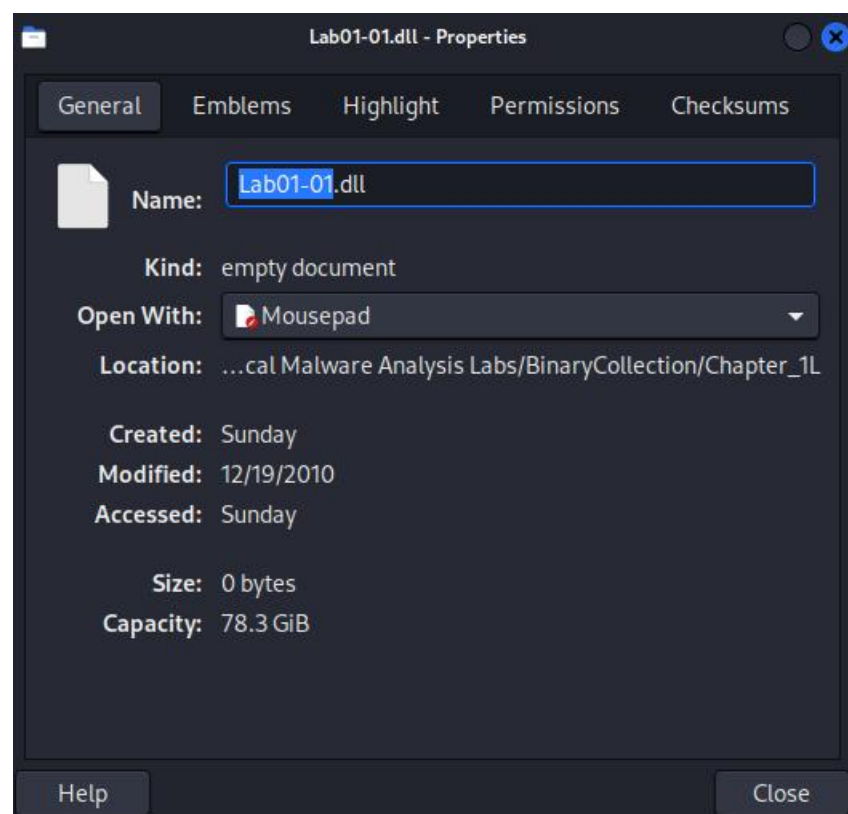
file match any existing antivirus signature?

相关的内容和分析可以在 virustotal 当中得到相应的分析和关于该病毒的信息。

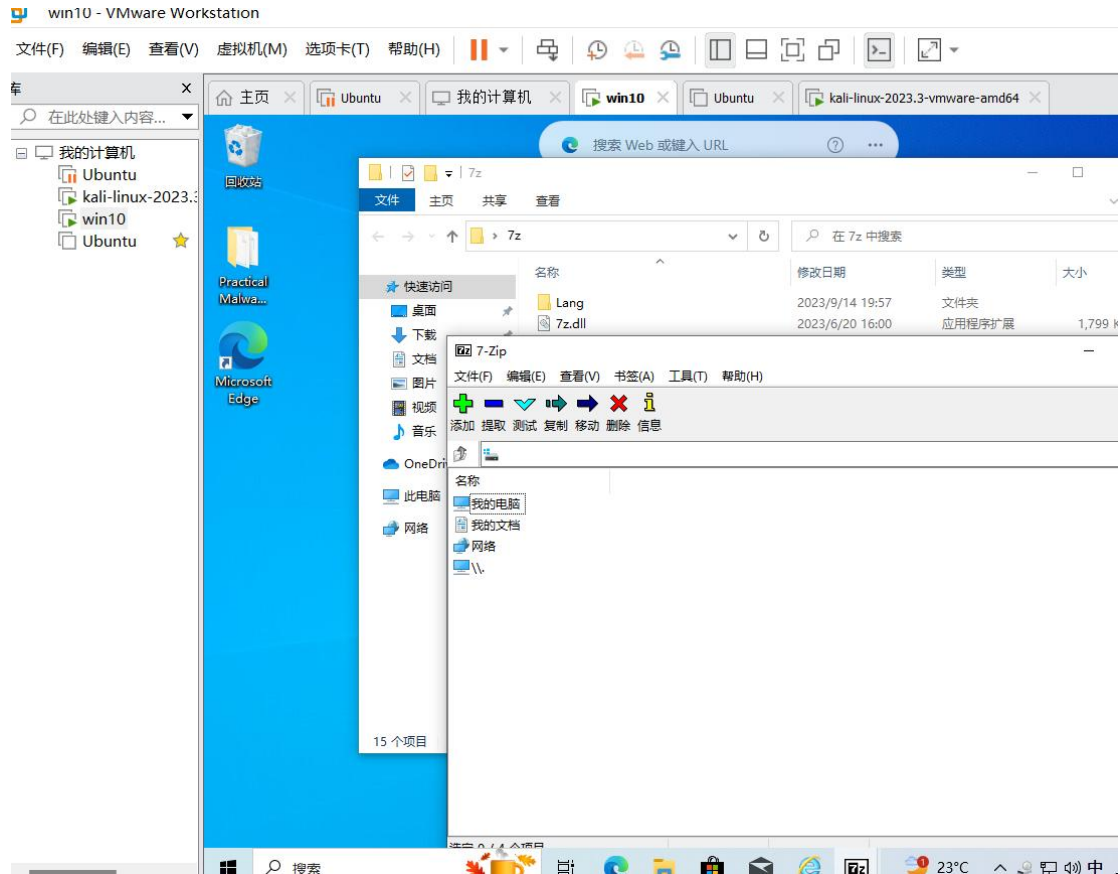


Lab01-01.exe:

Lab01-01.dll:



在 kali linux 当中可以看到其修改的日期。

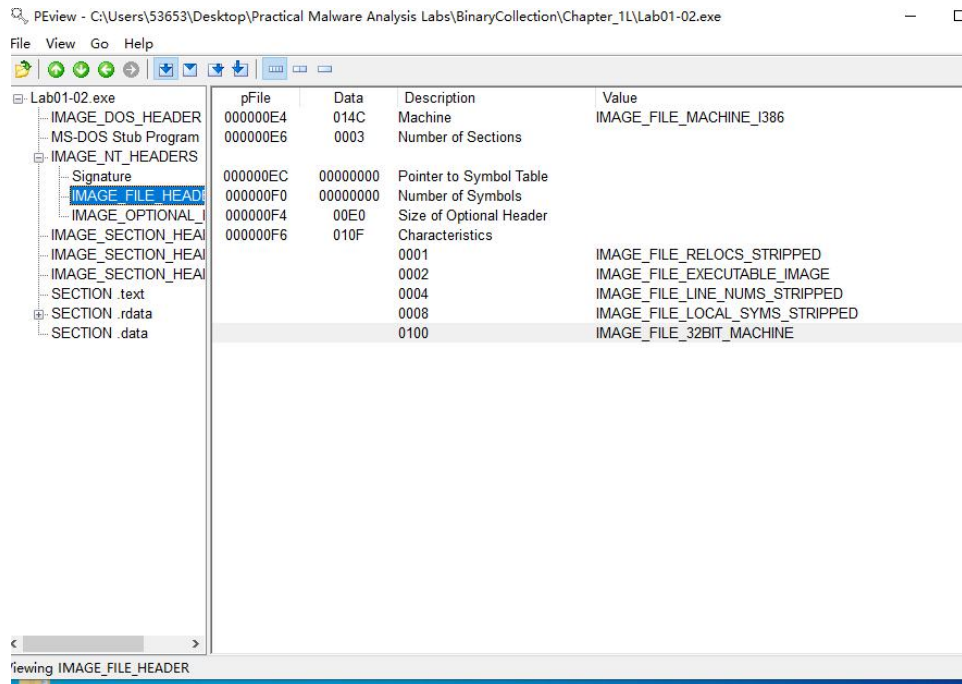


在 widows 当中，需要利用 7z 工具对相关的文件夹进行解压。

之后我尝试利用 PView 工具进行观察时间戳，我尝试在

IMAGE-HEADERS-IMAGE-FILE-HEADERS 里寻找关于时间戳的信息，但是出现报错后并没有找到时间戳的位置。

于是我重新打开了 vitustotal 的报告，找到了时间戳的位置。



在报告当中寻找其中的编译日期。

basic properties	
MD5	d41d8cd98f00b204e9800998ecf8427e
SHA-1	da39a3ee5e6b4b0d3255bfef95601890afd80709
SHA-256	e3b0c44298fc1c149afb4c8996fb92427ae41e4649b934ca495991b7852b855
SSDEEP	3::
TLSH	TNULL
File type	unknown
Magic	empty
File size	0 B (0 bytes)

History	
First Seen In The Wild	2005-09-19 10:49:09 UTC
First Submission	2006-09-18 07:26:15 UTC
Last Submission	2023-09-13 08:05:34 UTC
Last Analysis	2023-09-13 07:57:43 UTC

Names	
A91v4aybo_nnlqv4_1xo.tmp	
VPN_Lock.dat	

可以看出，这两个文件都已被一些安全软件识别为病毒，匹配到了已有的反病毒软件特征。

2. When were these files compiled?

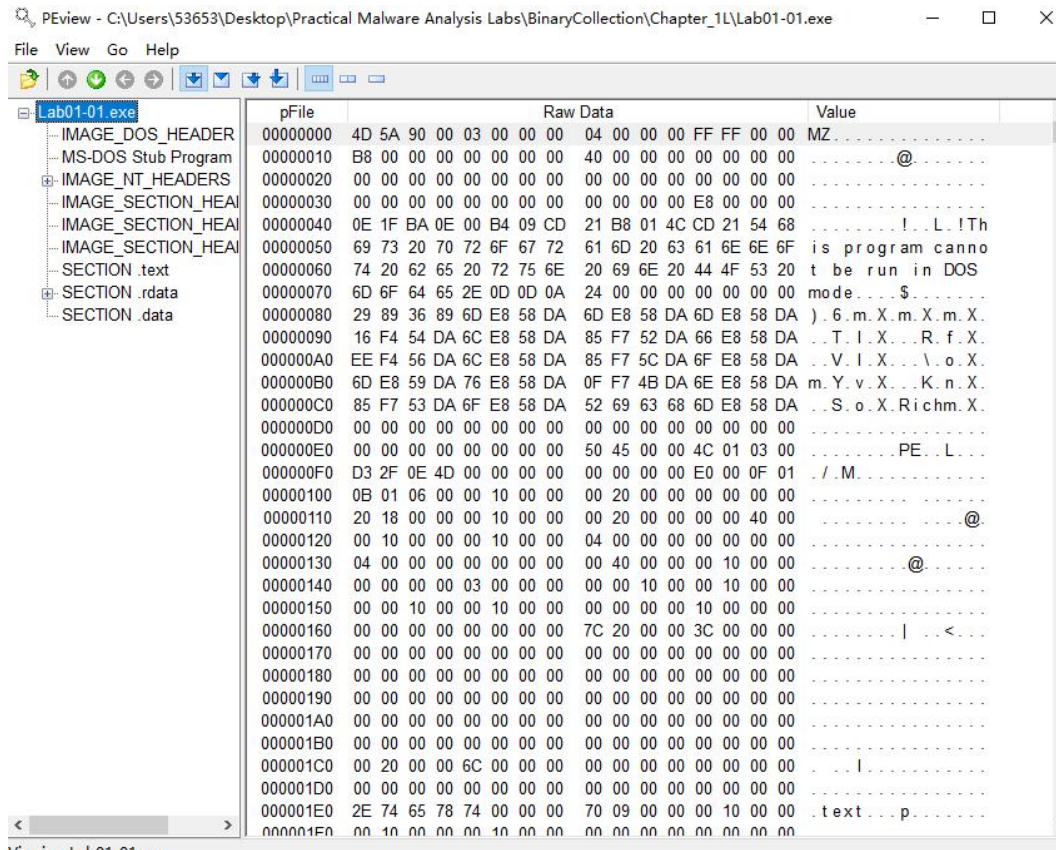
查看 Virus Total 给出的报告，可以查到 PE 头的相关信息，其中包含文件的编译时间。

SSDEEP	3::
TLSH	TNULL
File type	unknown
Magic	empty
File size	0 B (0 bytes)

History	
First Seen In The Wild	2005-09-19 10:49:09 UTC
First Submission	2006-09-18 07:26:15 UTC
Last Submission	2023-09-13 08:06:17 UTC
Last Analysis	2023-09-13 07:57:43 UTC

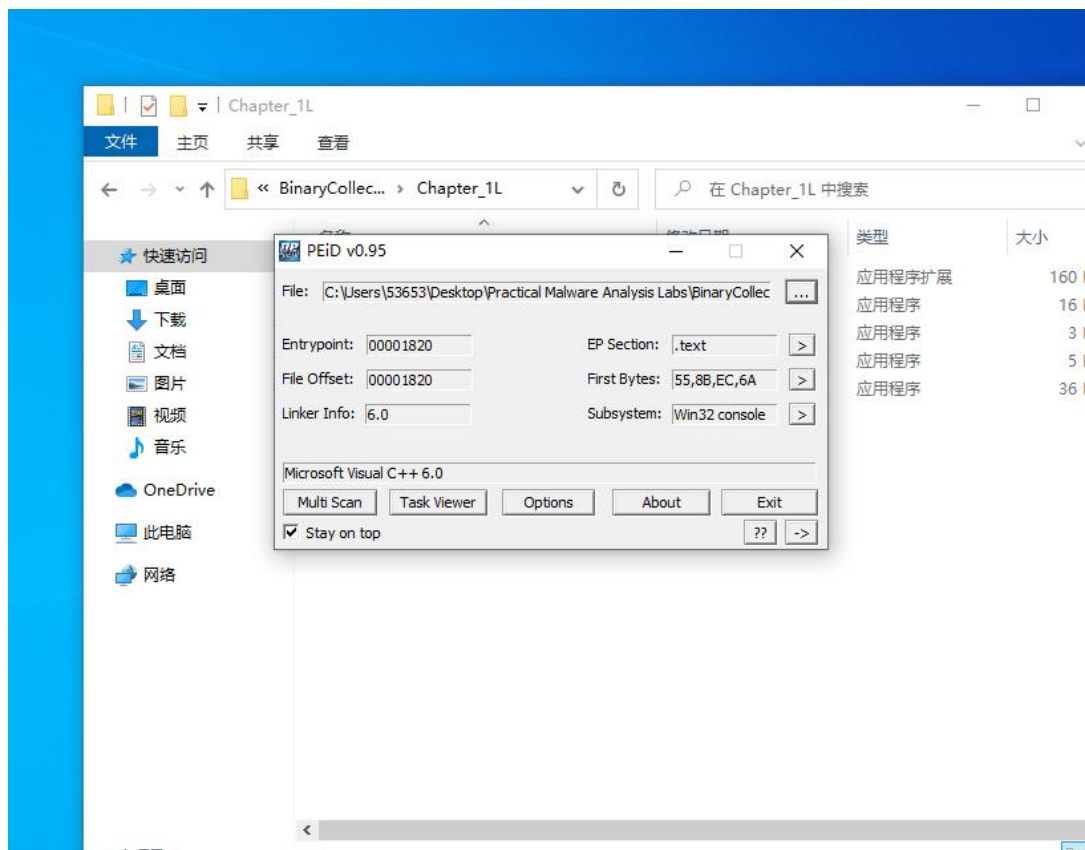
Names	
A9x1z208_1enizap_1yo.tmp	
A968l7eq_13ff8qg_17c.tmp	
A91872nqd_1skvkca_1n4.tmp	

从而知道，Lab01-01.exe 的编译时间为 2010-12-19 16:16:19；Lab01-01.dll 的编译时间为 2010-12-19 16:16:38。二者的编译时间都在 1min 之内。

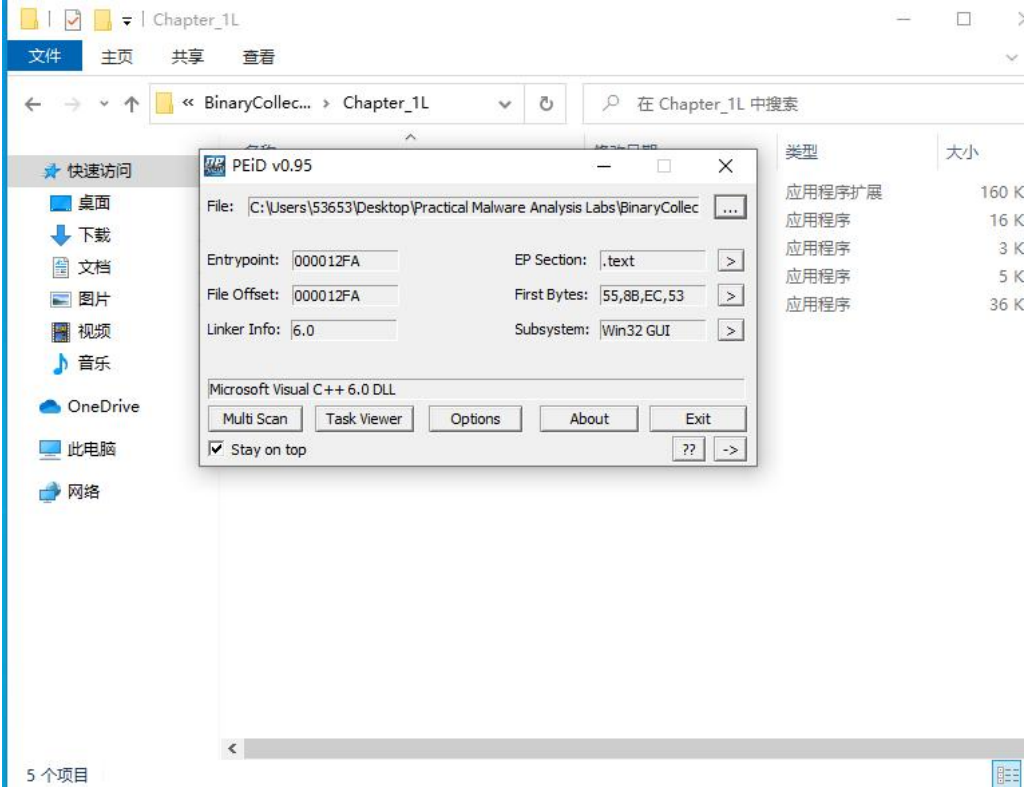


3. Are there any indications that either of these files is packed or obfuscated? If so, what are these indicators?

直接使用查壳工具 PEId 扫描文件，结果如下，可见没有加壳和混淆。



Lab01-01.exe 和 lab01-01.dll 都是没有加壳的文件，都可以直接通过 PEid 扫描到其中的很多信息。



Lab01-01.exe: Lab01-01.dll:

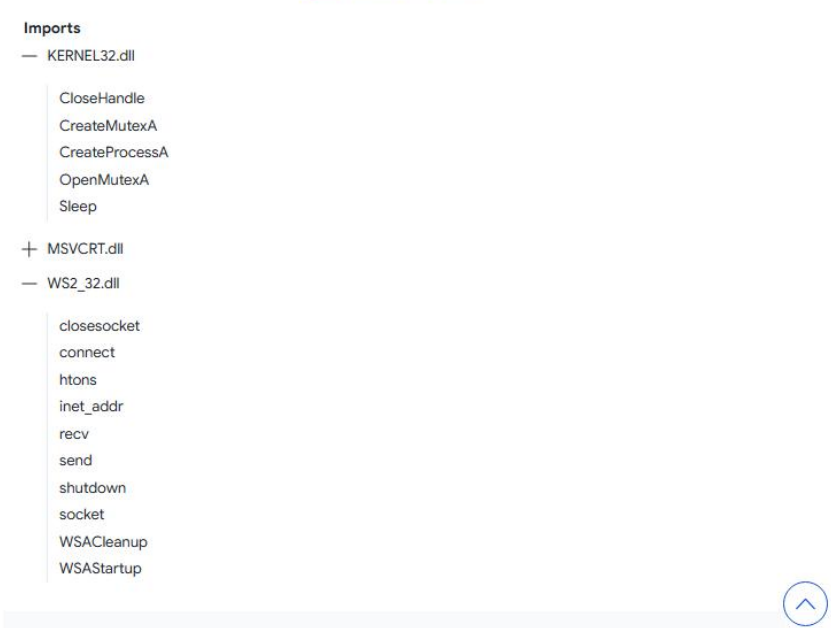
4. Do any imports hint at what this malware does? If so, which imports are they?

查看 Virus Total 给出的报告，可以知道文件中包含了哪些导入函数。

Lab01-01.exe:

其中值得关注的是 FindFirstFileA、FindNextFileA 和 CopyFileA，这三个函数配合使用，可以搜索文件系统里的所有文件并复制。文件当中只有很少的函数，说明他们可能只是一些小程序或者小漏洞。

三



Lab01-01.dll:

其中 CreateMutexA 函数用于创建一个互斥对象,可以和 OpenMutexA 一起操作一个互斥对象; CreateProcess 函数创建并启动一个新进程,如果恶意软件创建一个新的进程,新的流程需要分析; Sleep 函数可以使计算机程序进入休眠。由此可以猜想,这个 DLL 会创建一个互斥变量以保证一个资源同时只能被一个进程使用,如果这个互斥变量被锁,则其他的进程就执行 Sleep 函数等待。

而 WS2_32.dll 提供联网功能,该恶意软件很可能要执行网络相关的任务。

5. Are there any other files or host-based indicators that you could look for on infected systems?

使用 IDA 分析文件 Lab01-01.exe, 观察 Strings 窗口, 如下:

出现了 kernel32.dll, 这里企图用 1(one)混淆了 1, 说明这台主机已经被感染了。

此外,在下一行还会发现这个.exe 在运行时会导入 Lab01-01.dll,在上一题中已经分析了这个.dll 文件可能出现的恶意行为。

Address	Ordinal	Name	Library
00402000		CreateServiceA	ADVAPI32
00402004		StartServiceCtrlDispatcherA	ADVAPI32
00402008		OpenSCManagerA	ADVAPI32
00402010		SystemTimeToFileTime	KERNEL32
00402014		GetModuleFileNameA	KERNEL32
00402018		CreateWaitableTimerA	KERNEL32
0040201C		ExitProcess	KERNEL32
00402020		OpenMutexA	KERNEL32
00402024		SetWaitableTimer	KERNEL32
00402028		WaitForSingleObject	KERNEL32
0040202C		CreateMutexA	KERNEL32
00402030		CreateThread	KERNEL32
00402038		_exit	MSVCRT
0040203C		_XcptFilter	MSVCRT
00402040		exit	MSVCRT
00402044		_p__initenv	MSVCRT
00402048		_getmainargs	MSVCRT
0040204C		_initterm	MSVCRT
00402050		_setusermatherr	MSVCRT
00402054		_adjust_fdiv	MSVCRT
00402058		_p__commode	MSVCRT

6. What network-based indicators could be used to find this malware on infected machines?

使用 IDA 分析文件 Lab01-01.dll, 观察 Strings 窗口, 如下:

可以发现这里有一个 ip 地址,而在第四题分析导入函数时已经知道 WS2_32.dll 提供联网功能,该恶意软件很可能要执行网络相关的任务,因此可以推测是要与这里提到的网址联网通信。

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


```

C:\Users\53653\Desktop\strings>strings Lab01-01.dll

Strings v2.51
Copyright (C) 1999-2013 Mark Russinovich
Sysinternals - www.sysinternals.com

!This program cannot be run in DOS mode.

Rich
text
.rdata
@.data
.reloc
SUV
h8`
fj
h8`
L$xQh
h(
RVf
D$"
-(
j
IQh`

```

C:\Windows\system32\cmd.exe

```

u7WPS
u&WVS
WVS
_ ^[]
%
CloseHandle
Sleep
CreateProcessA
CreateMutexA
OpenMutexA
KERNEL32.dll
WS2_32.dll
strncmp
MSVCRT.dll
free
_initterm
malloc
_adjust_fdiv
exec
sleep
hello
127.26.152.13
SADFHUHF
/OI0[0h0p0
141G1[111
1Y2a2g2r2
3!3}3

```

C:\Users\53653\Desktop\strings>_

C:\Users\53653\Desktop\strings>

7. What would you guess is the purpose of these files?

通过以上分析可以推测，.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。

Lab 1-2

1. Upload the Lab01-02.exe file to <http://www.VirusTotal.com/> and view the reports.
Does it match any existing antivirus signature?
Lab01-02.exe:

VIRUSTOTAL

Q

SUMMARY

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](#).

56 security vendors and 1 sandbox flagged this file as malicious

56 / 71

Community Score

c876a332d7dd8da331cb8eee7ab7bf32752834d4b2b54eaa362674a2a48f64a6

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.amGfaWi867f
Bkav Pro	! W32.AIDetectMalware
ClamAV	! Win.Malware.Agent-6350563-0
CrowdStrike Falcon	! Win/malicious_confidence_100% (W)
Cybereason	! Malicious.cbcb77
Cylance	! Unsafe
Cynet	! Malicious (score: 100)
Cyren	! W32/Agent.DJC.gen!Eldorado
DeepInstinct	! MALICIOUS
DrWeb	! Trojan.Click3.12740
Elastic	! Malicious (moderate Confidence)

可以看出，这个文件都已被一些安全软件识别为病毒，匹配到了已有的反病毒软件特征。

2. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible. 使用查壳工具 PEid 扫描文件，结果如下，说明该文件已经被加壳了。

使用 Kali 的 upx -d 命令脱壳，脱壳成功！

```
kali@kali: ~/Desktop
File Actions Edit View Help
(kali@kali)-[~/Desktop]
$ upx
                                Ultimate Packer for eXecutables
                                Copyright (C) 1996 - 2020
UPX 3.96      Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 23rd 2020

Usage: upx [-123456789dlthVL] [-qvfk] [-o file] file..

Commands:
  -1      compress faster                      -9      compress better
  -d      decompress                          -l      list compressed file
  -t      test compressed file                -V      display version number
  -h      give more help                      -L      display software license

Options:
  -q      be quiet                            -v      be verbose
  -oFILE  write output to 'FILE'
  -f      force compression of suspicious files
  -k      keep backup files
file..   executables to (de)compress

Type 'upx --help' for more detailed help.

UPX comes with ABSOLUTELY NO WARRANTY; for details visit https://upx.github.io

(kali@kali)-[~/Desktop]
$
```

```
kali@kali: ~/Desktop
File Actions Edit View Help
(kali@kali)-[~/Desktop]
$ upx -d /home/Lab01-02.exe
                                Ultimate Packer for eXecutables
                                Copyright (C) 1996 - 2020
UPX 3.96      Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 23rd 2020

      File size      Ratio      Format      Name
-----
upx: /home/Lab01-02.exe: FileNotFoundException: /home/Lab01-02.exe: No such file or directory

Unpacked 0 files.

(kali@kali)-[~/Desktop]
$ upx -d /home/kali/Lab01-02.exe
                                Ultimate Packer for eXecutables
                                Copyright (C) 1996 - 2020
UPX 3.96      Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 23rd 2020

      File size      Ratio      Format      Name
-----
upx: /home/kali/Lab01-02.exe: IOException: empty file -- skipped

Unpacked 0 files.

(kali@kali)-[~/Desktop]
$
```

同样也可以使用 windows 下的 upx 软件进行脱壳，upx 脱壳的结果可以用于我们继续在 windows 系统下进行 PEid 的分析和脱壳后函数的观察。

```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.19045.3448]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\53653>cd \Desktop\upx
系统找不到指定的路径。

C:\Users\53653>cd Desktop\upx

C:\Users\53653\Desktop\upx>upx -d Lab01-02.exe
Ultimate Packer for eXecutables
Copyright (C) 1996 - 2013
UPX 3.91w Markus Oberhumer, Laszlo Molnar & John Reiser Sep 30th 2013

-----
File size      Ratio      Format      Name
-----
16384 <-      3072      18.75%      win32/pe      Lab01-02.exe

Unpacked 1 file.

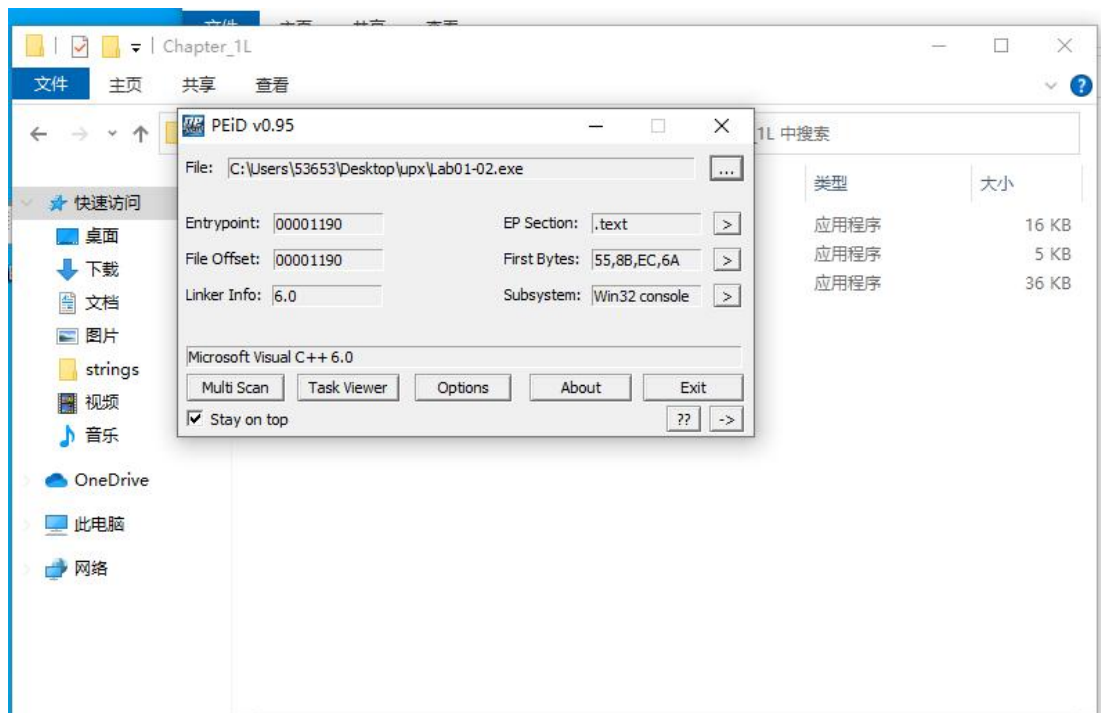
C:\Users\53653\Desktop\upx>
```

3. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?

将脱壳后的 Lab01-02.exe 文件上传到 Virus Total 网站上，得到与脱壳前不同的导入函数，如下：

Imports	
— ADVAPI32.dll	CreateServiceA
— KERNEL32.DLL	ExitProcess GetProcAddress LoadLibraryA VirtualAlloc VirtualFree VirtualProtect
+ MSVCRT.dll	
— WININET.dll	InternetOpenA

进而可以说明脱壳可以给文件的功能发生一定的变化。



推测该程序将有开启或创建服务、联网等行为。

4. What host- or network-based indicators could be used to identify this malware on infected machines?

使用 IDA 分析文件 Lab01-02.exe，观察其函数窗口和 Strings 窗口，如下：

有一些可疑字符串和可疑网址。

Address	Ordinal	Name	Library
00402000		OpenProcessToken	ADVAPI32
00402004		LookupPrivilegeValueA	ADVAPI32
00402008		AdjustTokenPrivileges	ADVAPI32
00402010		GetProcAddress	KERNEL32
00402014		LoadLibraryA	KERNEL32
00402018		WinExec	KERNEL32
0040201C		WriteFile	KERNEL32
00402020		CreateFileA	KERNEL32
00402024		SizeofResource	KERNEL32
00402028		CreateRemoteThread	KERNEL32
0040202C		FindResourceA	KERNEL32
00402030		GetModuleHandleA	KERNEL32
00402034		GetWindowsDirectoryA	KERNEL32
00402038		MoveFileA	KERNEL32
0040203C		GetTempPathA	KERNEL32
00402040		GetCurrentProcess	KERNEL32
00402044		OpenProcess	KERNEL32
00402048		CloseHandle	KERNEL32
0040204C		LoadResource	KERNEL32
00402054		_snprintf	MSVCRT
00402058		exit	MSVCRT

其中的一些可以字符串我们可以用来进行 yara 规则的编写。


Address	Length	Type	String
.rdata:0040228E	0000000D	C	KERNEL32.dll
.rdata:004022E0	0000000D	C	ADVAPI32.dll
.rdata:004022FA	0000000B	C	MSVCRT.dll
.data:0040302C	00000011	C	SeDebugPrivilege
.data:00403040	0000000B	C	sfc_os.dll
.data:0040304C	00000016	C	\\system32\\wupdmgr.exe
.data:00403064	00000005	C	%s%s
.data:00403070	00000005	C	#101
.data:00403078	00000013	C	EnumProcessModules
.data:0040308C	0000000A	C	psapi.dll
.data:00403098	00000013	C	GetModuleBaseNameA
.data:004030AC	0000000A	C	psapi.dll
.data:004030B8	0000000E	C	EnumProcesses
.data:004030C8	0000000A	C	psapi.dll
.data:004030D4	00000016	C	\\system32\\wupdmgr.exe
.data:004030EC	00000005	C	%s%s
.data:004030F4	0000000B	C	\\winup.exe
.data:00403100	00000005	C	%s%s

Lab 1-3

1. Upload the Lab01-03.exe file to <http://www.VirusTotal.com/> and view the reports. Does it match any existing antivirus signature?


Lab01-03.exe:

可以看出，这个文件都已被一些安全软件识别为病毒，匹配到了已有的反病毒软件特征。












SUMMARY
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  **trojan.graftor/genome** Threat categories trojan spywi Family labels graftor genome label

Security vendors' analysis ⓘ Do you want to automate checks?

AhnLab-V3	 Trojan/Win.Generic.R427327
Alibaba	 TrojanClicker:Win32/Tnega.79cba6fb
ALYac	 Gen:Variant.Graftor.968808
Antiy-AVL	 Trojan/Win32.SGeneric
Arcabit	 Trojan.Graftor.DEC868
Avast	 Win32:Malware-gen
AVG	 Win32:Malware-gen
Baidu	 Win32.Trojan-Clicker.Agent.z
BitDefender	 Gen:Variant.Graftor.968808

Sections

Name	Virtual Address	Virtual Size	Raw Size	Entropy	MD5	Chi2
t	4096	12288	0	0	d41d8cd98f00b204e9800998ecf8427e	-1
ta	16384	4096	652	7.36	dcbb3117347a183b93cc9e50e09abd92	580.1
a	20480	4096	512	4.51	83d2bc9613dfc4bc5c714214023f386f	27890

Imports

— KERNEL32.dll

GetProcAddress
LoadLibraryA

Overlay

entropy

offset 4748

chi2 1020

History ⓘ

Creation Time	2019-08-30 22:26:59 UTC
First Seen In The Wild	2011-07-05 18:16:16 UTC
First Submission	2011-07-06 00:05:42 UTC
Last Submission	2023-09-14 12:06:36 UTC
Last Analysis	2023-09-13 22:02:32 UTC

— ADVAPI32.dll

AdjustTokenPrivileges
LookupPrivilegeValueA
OpenProcessToken

— KERNEL32.dll

CloseHandle
CreateFileA
CreateRemoteThread
FindResourceA
GetCurrentProcess
GetModuleHandleA
GetProcAddress
GetTempPathA
GetWindowsDirectoryA
LoadLibraryA

— MSVCRT.dll

- __getmainargs
- __p__initenv
- __p__commode
- __p__fmode
- __set_app_type
- __setusermatherr
- __adjust_fdiv
- __controlfp
- __except_handler3
- __exit

2. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible. 使用查壳工具 PEid 扫描文件，结果如下，说明该文件已经被加壳了。

我们还可以搜索相应的导入表，却发现其没有明显的导入表，说明这是一个加壳后的文件。使用脱壳工具 Linxer Unpacker 进行壳特征脱壳，脱壳成功！

3. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?

将脱壳后的 Lab01-03.exe 文件上传到 Virus Total 网站上，得到与脱壳前不同的导入函数，该程序可能将有组件对象模型 COM、联网等相关操作。

4. What host- or network-based indicators could be used to identify this malware on infected machines?

使用 IDA 分析文件 Lab01-03.exe，观察 Strings 窗口，如下：可以看到一些可疑字符串。

Function name	Address	Length	Type	String
start	seg001:004040...	00000005	C	3Bt>O
sub_40501F	seg001:004040...	00000007	C	2]<,M\b1
sub_405090	seg001:004041...	00000006	C	\x1B!*G9>
sub_405092	seg001:004041...	00000008	C	ole32.vd
	seg001:004041F4	00000009	C	OLEAUT32.dll
	seg001:004042...	0000000A	C	IMSVCR71.dll
	seg001:004042...	00000007	C	_getmas
	seg001:004042...	00000007	C	P2r3Us
	seg001:0040423F	00000005	C	plvuy
	seg001:004042...	00000005	C	t)\r4p
	seg004:004051...	0000000D	C	KERNEL32.dll

Lab 1-4

1. Upload the Lab01-04.exe file to <http://www.VirusTotal.com/> and view the reports. Does it match any existing antivirus signature?

Lab01-04.exe:

可以看出，这个文件都已被一些安全软件识别为病毒，匹配到了已有的反病毒软件特征。

2. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible. 使用查壳工具 PEid 扫描文件，结果如下，说明该文件没有被加壳或混淆。

3. When were these files compiled?

查看 Virus Total 给出的报告，可以查到 PE 头的相关信息，其中包含文件的编译时间。因此这个文件的编译时间为 2019-08-30 22:26:59。（但我认为这是个不可信的时间）

4. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?

该程序可能将有操作权限、创建可执行文件并运行、联网等行为。

5. What host- or network-based indicators could be used to identify this malware on infected machines?

使用 IDA 分析文件 Lab01-04.exe，观察 Strings 窗口，如下：可以看到一些可疑字符串。其中字符串“\\system32\\wupdmgr.exe”表示这个程序会在这个位置创建或者修改文件。

Address	Ordinal	Name	Library
00402000		CloseHandle	KERNEL32
00402004		UnmapViewOfFile	KERNEL32
00402008		IsBadReadPtr	KERNEL32
0040200C		MapViewOfFile	KERNEL32
00402010		CreateFileMappingA	KERNEL32
00402014		CreateFileA	KERNEL32
00402018		FindClose	KERNEL32
0040201C		FindNextFileA	KERNEL32
00402020		FindFirstFileA	KERNEL32
00402024		CopyFileA	KERNEL32
0040202C		malloc	MSVCRT
00402030		exit	MSVCRT
00402034		_exit	MSVCRT
00402038		_XcptFilter	MSVCRT
0040203C		_p__initenv	MSVCRT
00402040		_getmainargs	MSVCRT
00402044		_initterm	MSVCRT
00402048		_setusermatherr	MSVCRT
0040204C		_adjust_fdiv	MSVCRT
00402050		_p__commode	MSVCRT
00402054		_p__fmode	MSVCRT

Line 1 of 25

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

6. This file has one resource in the resource section. Use Resource Hacker to examine that resource, and then use it to extract the resource. What can you learn from the resource?

接下来就是使用 yara 规则和 yara 程序对相应的病毒文件进行扫描

首先我们编写并利用以下的 yara 规则进行扫描。

```
rule Lab1
{
    meta:
        description = "rules for Lab1 "
        date = "202x/xx/xx"
        author = "LYT"
```

```

strings:

$a = "kernel32.dll" wide ascii

$b = "127.26.152.13" wide ascii

$c = "http://www.malwareanalysisbook.com" wide ascii

$d = "wupdmgr" wide ascii

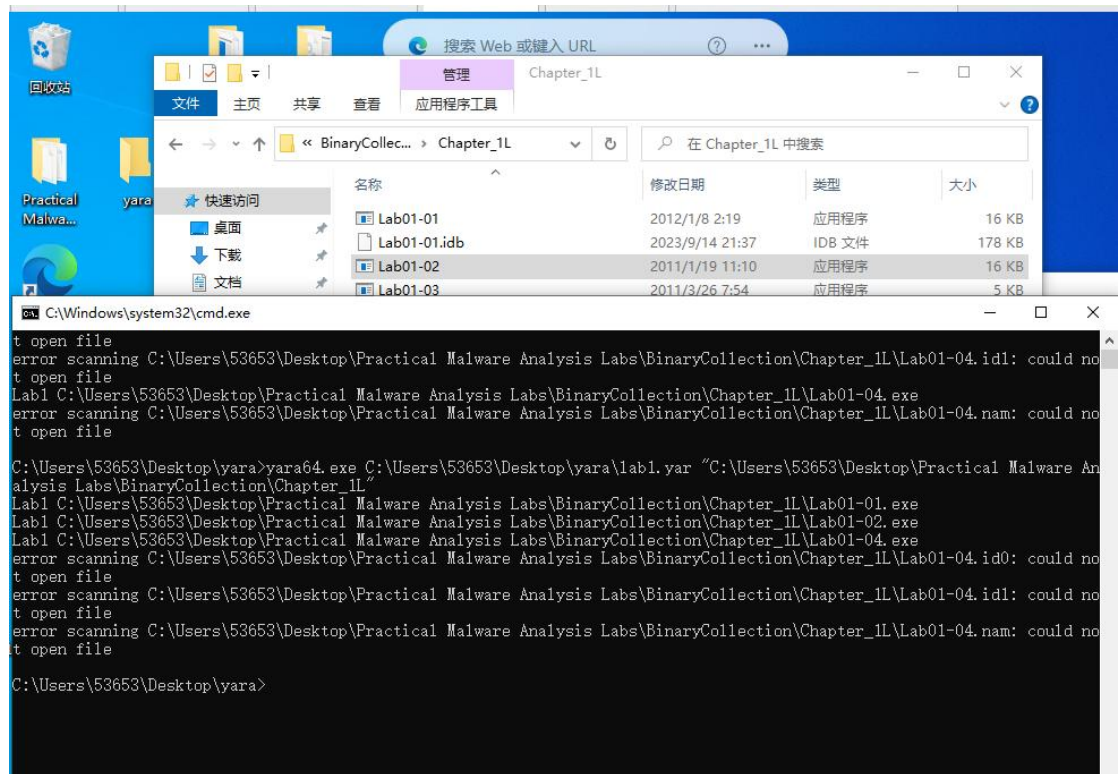
condition:

any of them

}

```

得到以下的输出结果，并且在 powershell 当中得到相应的运行时间。




```

PS C:\Users\53653\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L> measure-command {C:\Users\53653\Desktop\yara\yara64.exe C:\Users\53653\Desktop\yara\yarylouhua.yar "C:\Users\53653\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L"}
error scanning C:\Users\53653\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L\Lab01-04.id0: cannot open file
error scanning C:\Users\53653\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L\Lab01-04.id1: cannot open file
error scanning C:\Users\53653\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L\Lab01-04.name: cannot open file

Days           : 0
Hours          : 0
Minutes        : 0
Seconds        : 0
Milliseconds   : 18
Ticks          : 188707
TotalDays      : 2.1841087962963E-07
TotalHours     : 5.2418611111111E-06
TotalMinutes   : 0.00031451166666667
TotalSeconds   : 0.0188707
TotalMilliseconds : 18.8707

PS C:\Users\53653\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L>

```

可以看到运行时间从 47.1971 降低到了 18.8707，可以说明我们得到了一个比较好的结果，成功进行了相应的优化。

四、实验结论及心得体会

通过本次实验，我进行了许多病毒程序的分析和 yara 扫描的工作，我发现了很多的应用软件需要得到运用，还有许多的软件其中具有很多不同的功能。例如 PEview 可以将 PE 文件的许多问题都能展现，PEid 可以很好的扫描加壳，strings 对字符串的扫描非常精准，upx 的脱壳简单有效，yara 的扫描方便快捷，ida 的使用一直十分全面方便。

yara 规则的使用有很多的优化空间

优化字符串匹配：

1. 使用正则表达式来匹配字符串，可以提高灵活性和减少规则中的字符串数量。

对于 IP 地址和 URL 等可能存在变化的值，可以使用通配符或正则表达式部分匹配。

2. 精确定义字符串类型：

指定更具体的字符串类型，如 wide ascii 替换为 wide 或 ascii，以减少匹配的计算量。

3. 考虑添加更多具有辨别力的字符串：

根据你的需求，可以添加更多独特而与恶意行为相关的字符串。

4. 使用元数据进行分类和分组：

使用元数据来对规则进行分类和分组，以便更好地管理和组织规则集。

通过对以上的 yara 规则优化方案的使用，我们可以很好的发现 yara 规则可以有效的减少相应的程序运行时间，提高程序运行的效率。