2023-12-27

北京理工大学计算机学院

计算机病毒防治

实验八 恶意软件行为实验报告

目录

[实验准备 3](#_Toc156236777)

[一、 实验环境和所用到的软件 3](#_Toc156236778)

[二、 基本知识 3](#_Toc156236779)

[Lab 11-1 3](#_Toc156236780)

[一、 恶意软件会将哪些内容放入磁盘 3](#_Toc156236781)

[二、 恶意软件如何实现持久性 6](#_Toc156236782)

[三、 恶意软件如何窃取用户凭据的 7](#_Toc156236783)

[四、 恶意软件如何处理被盗凭据 9](#_Toc156236784)

[五、 如何使用该恶意软件在测试环境中获取用户凭据 12](#_Toc156236785)

[Lab 11-2 13](#_Toc156236786)

[一、 这个DLL恶意软件的导出是什么 13](#_Toc156236787)

[二、 使用rundll32.exe安装此恶意软件后会发生什么 13](#_Toc156236788)

[三、 为了正确地安装恶意软件，Lab11-02.ini必须位于何处 15](#_Toc156236789)

[四、 如何安装此恶意软件以实现持久性 15](#_Toc156236790)

[五、 这个恶意软件使用了什么用户空间rootkit技术 15](#_Toc156236791)

[六、 钩子代码的作用是什么 20](#_Toc156236792)

[七、 该恶意软件攻击哪个进程，为什么 21](#_Toc156236793)

[八、 .ini文件的意义是什么 21](#_Toc156236794)

[九、 如何使用Wireshark动态捕获此恶意软件的活动 21](#_Toc156236795)

[Lab 11-3 22](#_Toc156236796)

[一、 使用基本静态分析可以发现哪些有趣的分析结果 22](#_Toc156236797)

[二、 运行此恶意软件时会发生什么 23](#_Toc156236798)

[三、 Lab11-03.exe如何持久安装Lab11-03.dll 25](#_Toc156236799)

[四、 恶意软件会感染哪个Windows系统文件 28](#_Toc156236800)

[五、 Lab11-03.dll是做什么的 28](#_Toc156236801)

[六、 恶意软件将收集的数据存储在哪里 29](#_Toc156236802)

[实验总结 30](#_Toc156236803)

# 实验准备

1. 实验环境和所用到的软件

* Windows 10 x86
* Strings、PE View、Procmon、IDA Pro、Resource Hacker、WinHex、Netcat等

1. 基本知识
2. 常见的恶意代码行为：

* 下载器和启动器
* 后门
* 登录凭证窃取
* 开机自启
* 权限提升
* 用户态Rootkit

# Lab 11-1

1. 恶意软件会将哪些内容放入磁盘

首先用strings指令来查看Lab11-01.exe中包含的字符串，结果如下：

图形用户界面, 文本

描述已自动生成

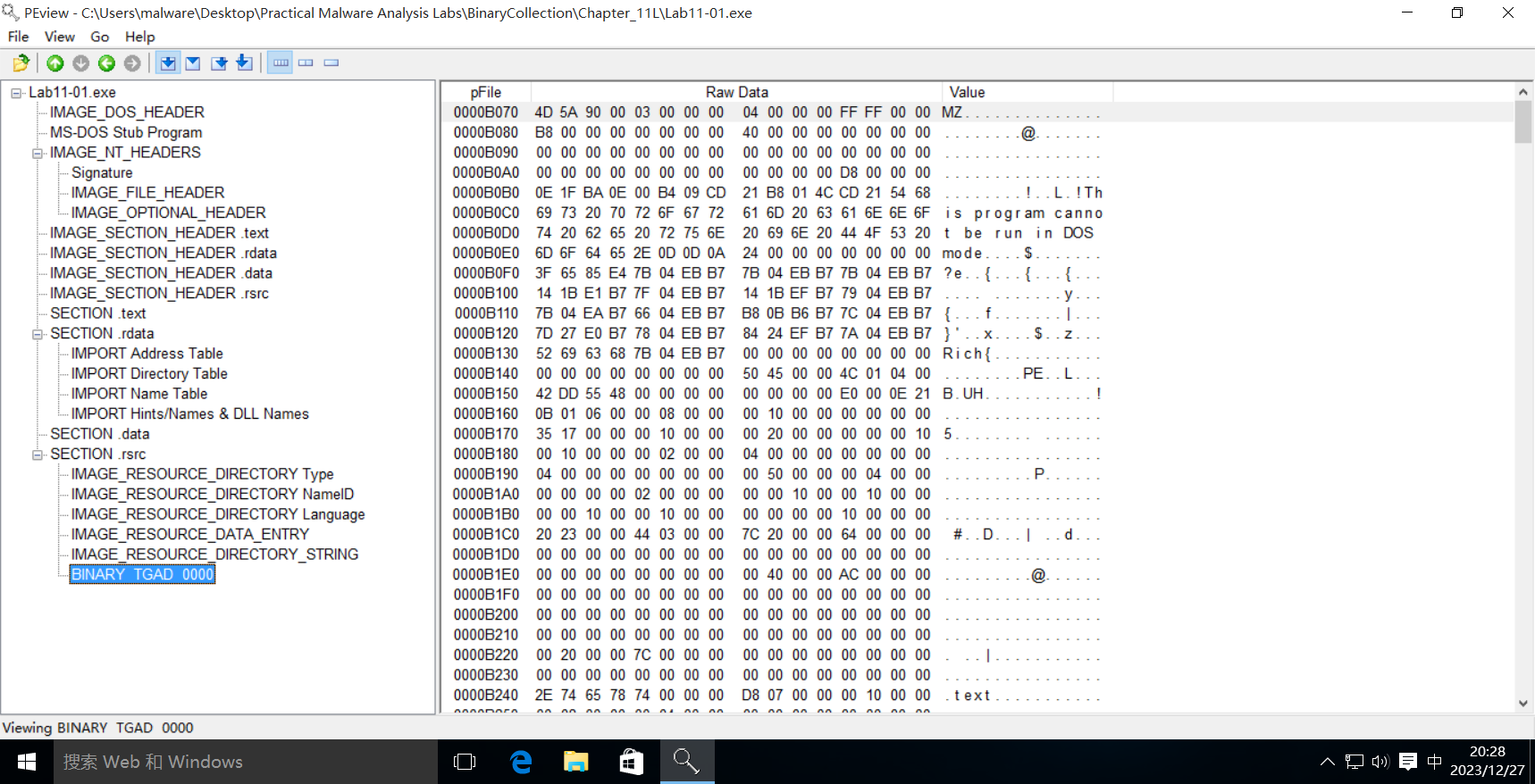
我们可以发现GinaDLL、SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon等字符串，这让我们猜测这个恶意软件可能是一个GINA拦截恶意软件。

接下来使用PE View来查看恶意软件的导入，我们可以发现用于操作注册表和提取资源部分的功能的函数。

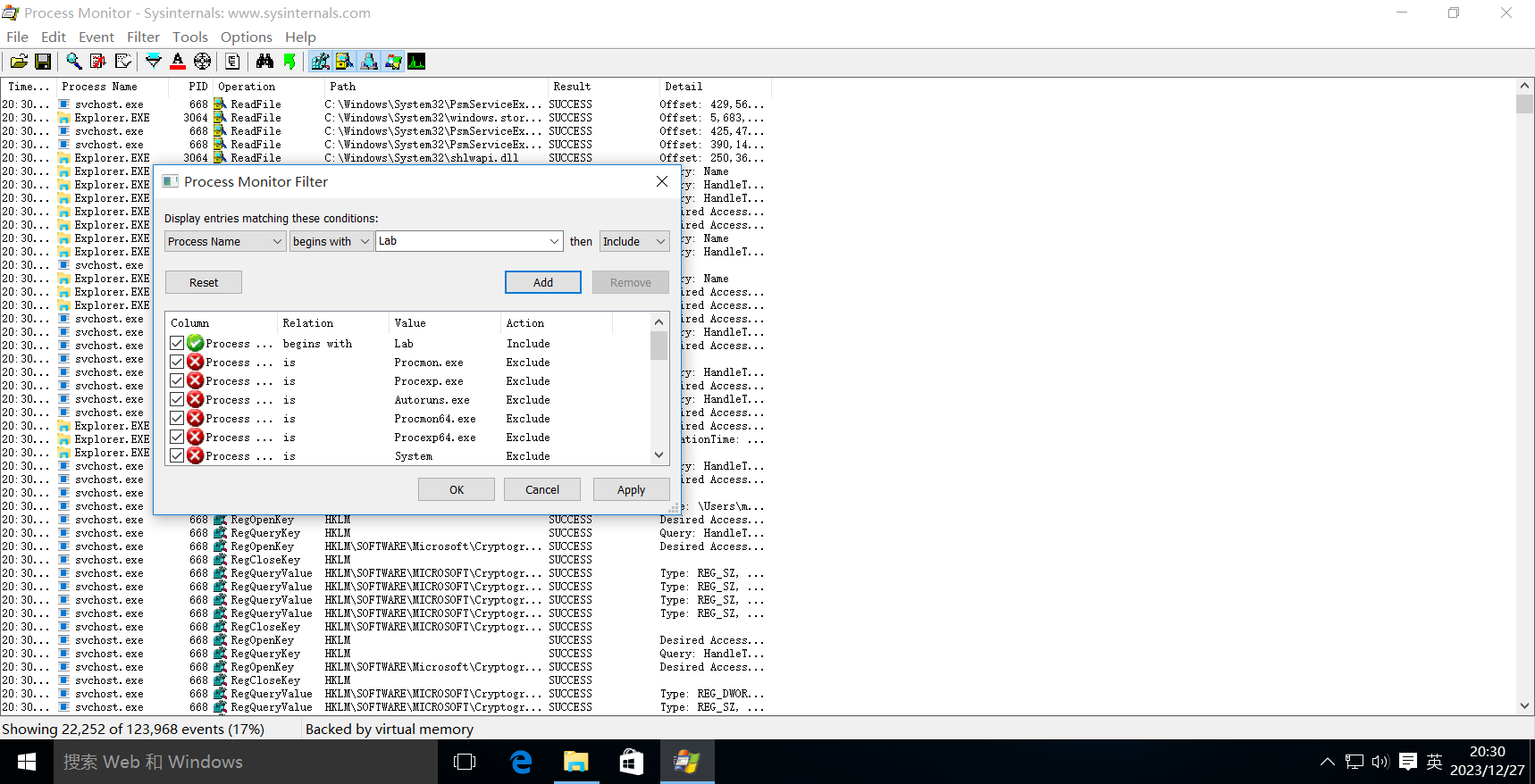
图形用户界面

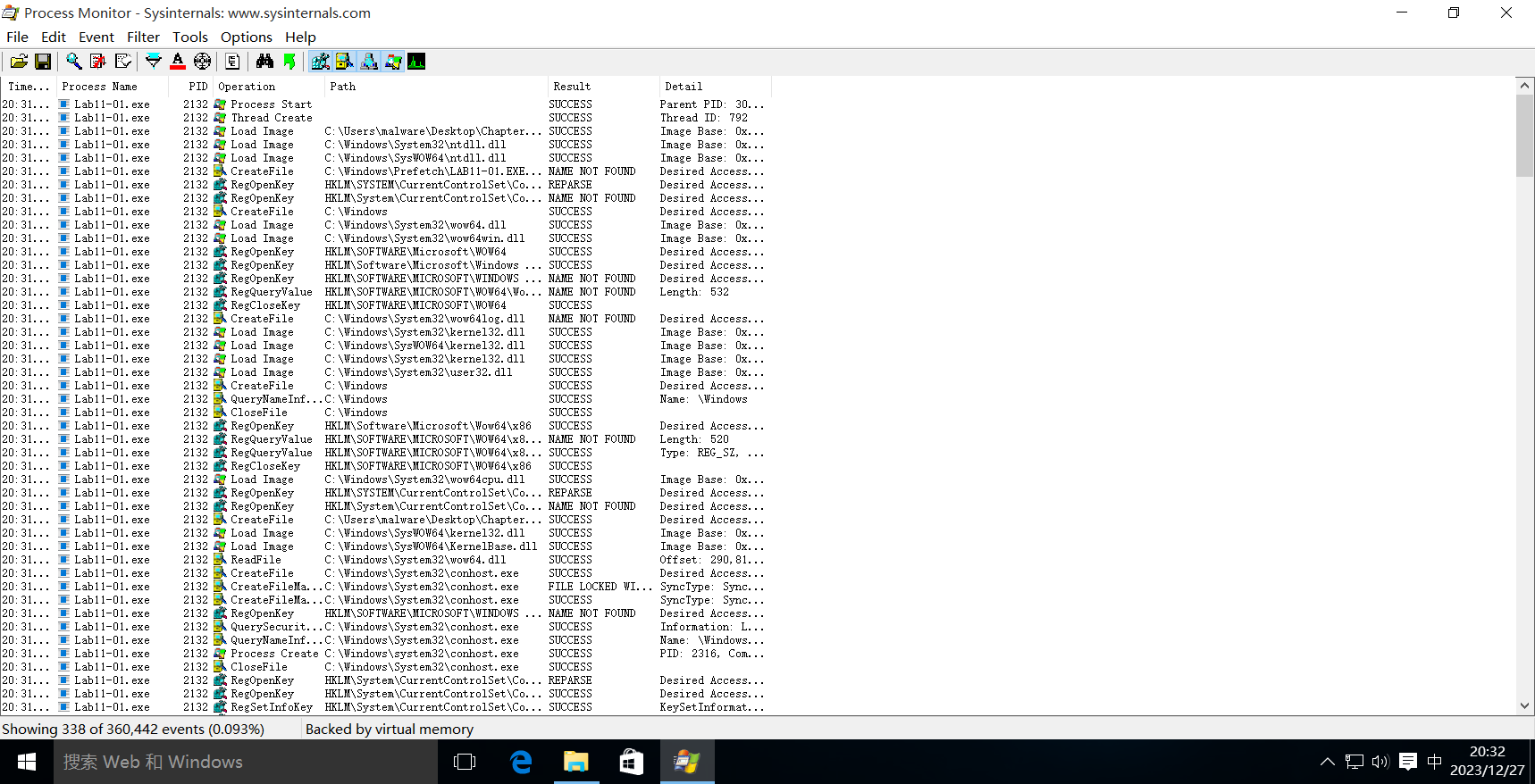
描述已自动生成

在检查PE文件格式时，我们会发现一个名为TGAD的资源部分，并且其嵌入了PE文件。

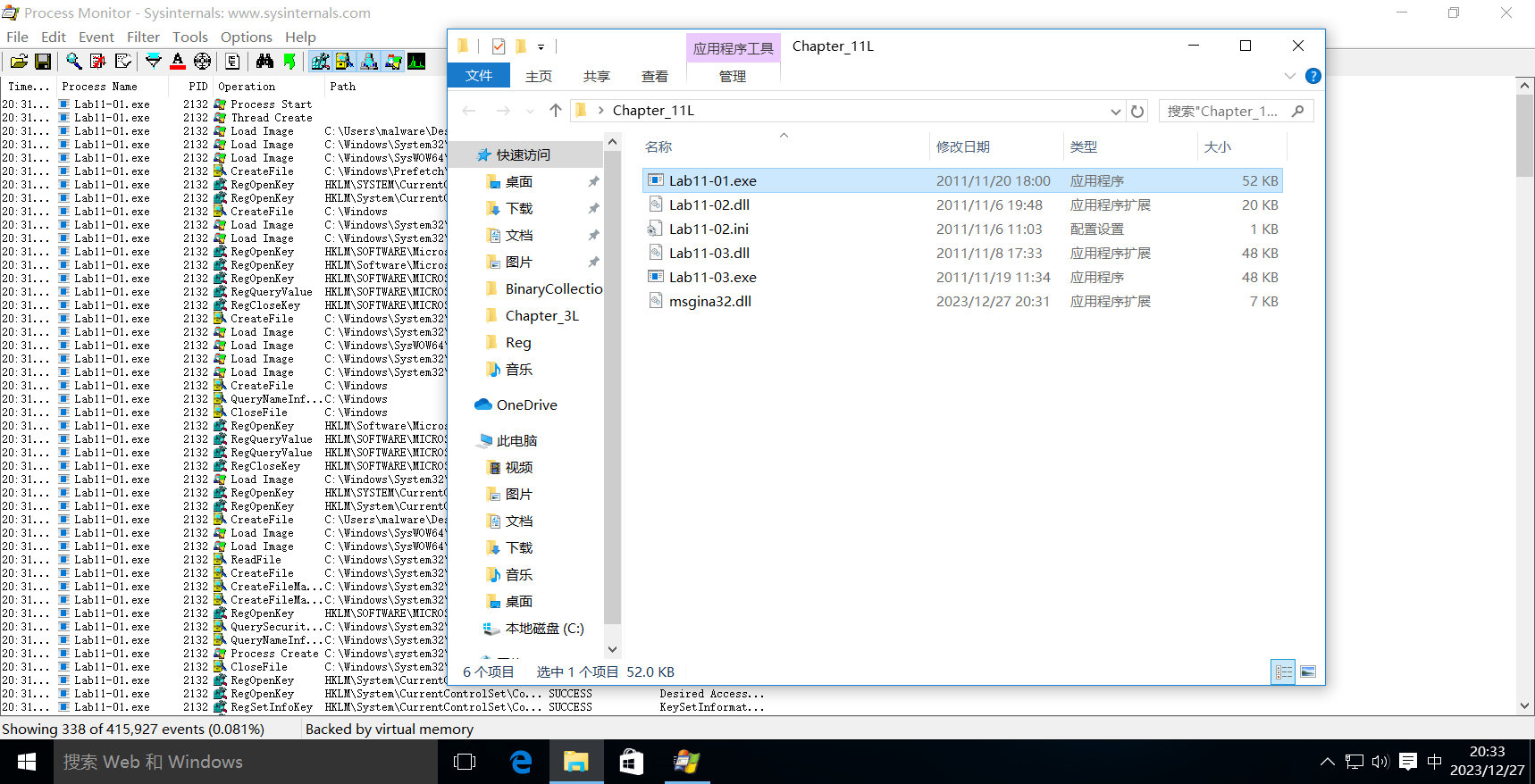


接下来我们打开procmon然后设置一个过滤：进程名为Lab11-01.exe。然后启动恶意软件。



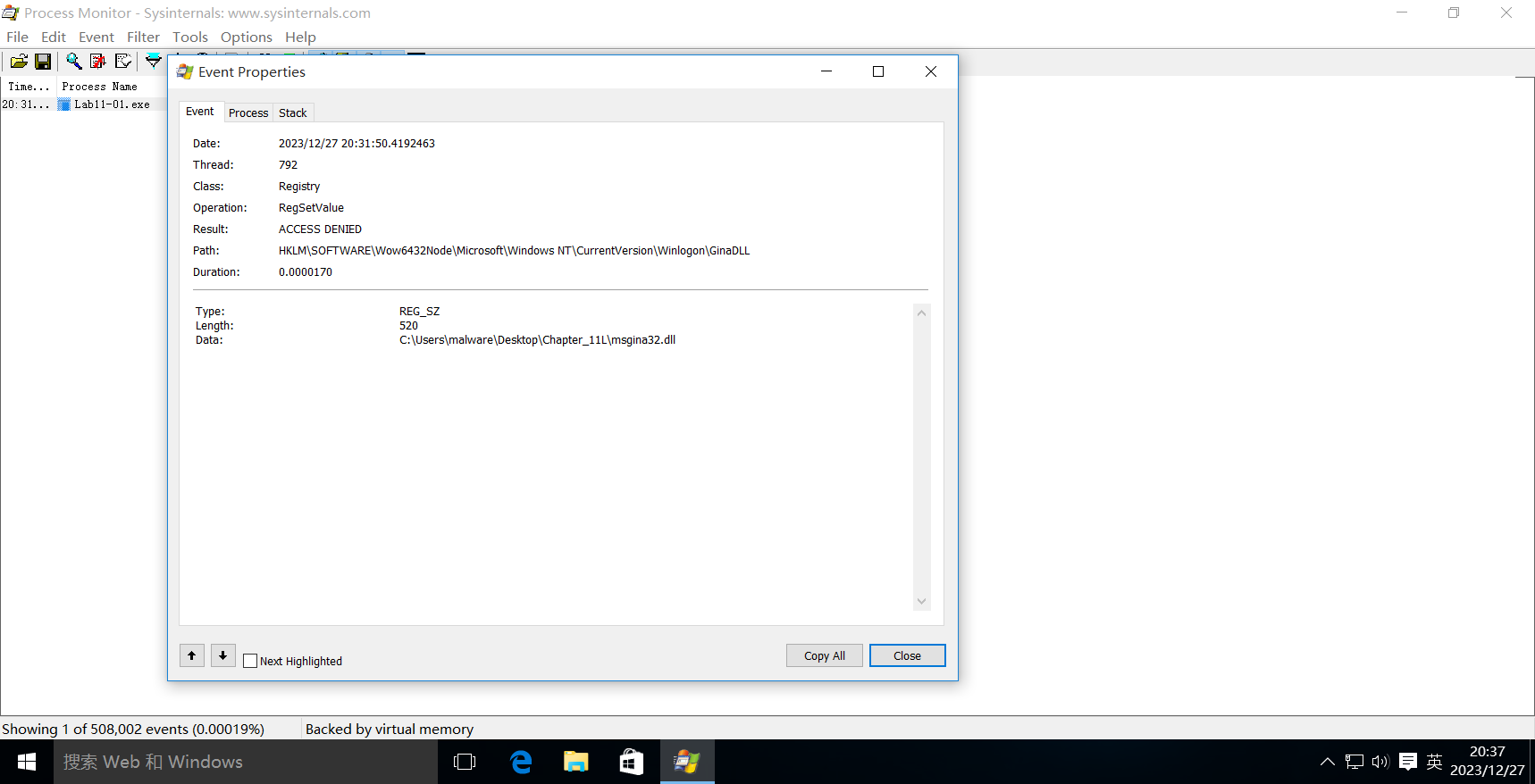


可以发现，在启动软件后，在同级目录下出现了名为msgina32.dll文件。这表明恶意软件从TGAD资源节部分抽取程序到同级目录下msgina32.dll。



1. 恶意软件如何实现持久性

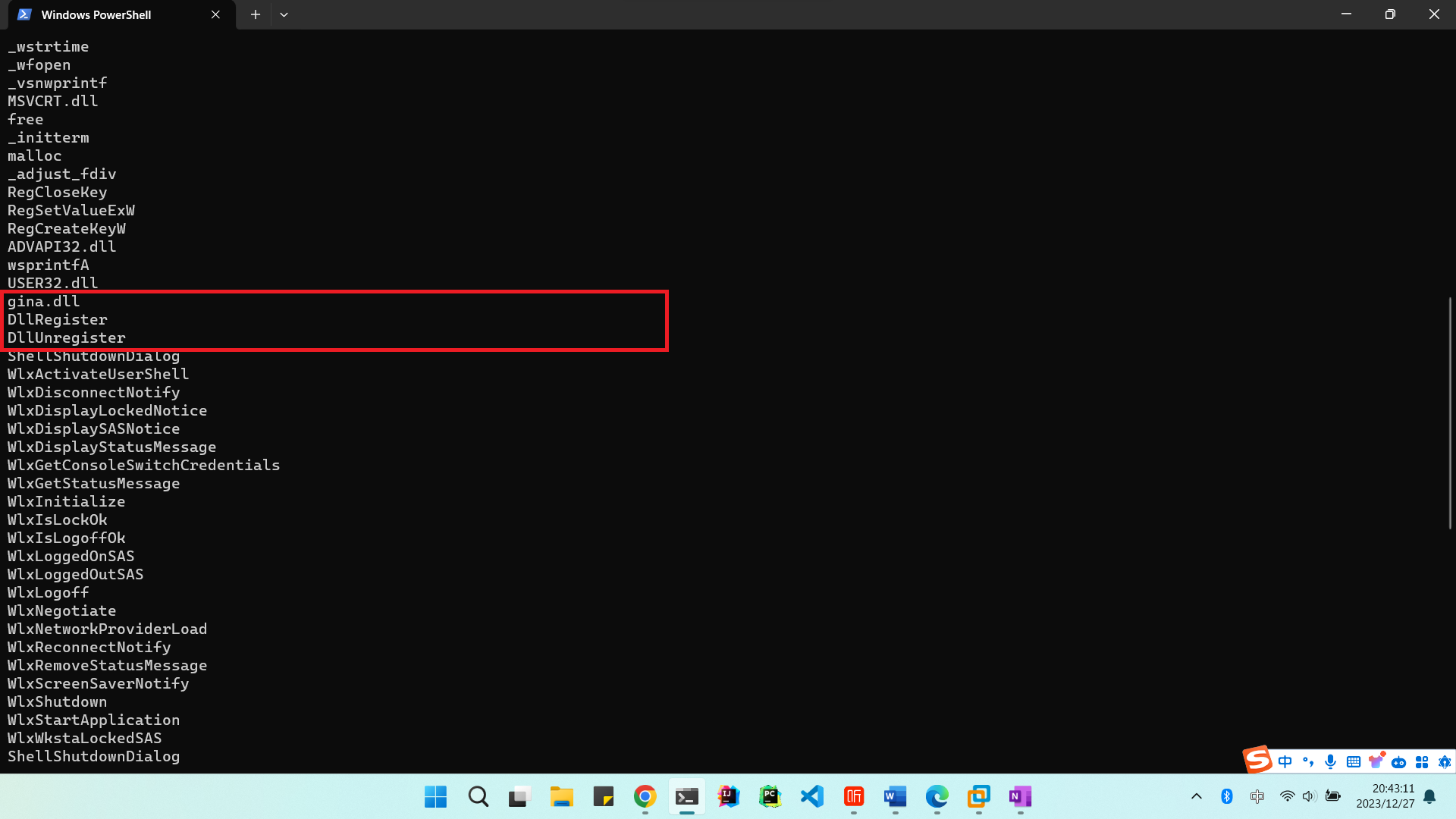
在procmon中设置过滤条件，我们可以发现恶意软件修改了注册表，在HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\GinaDLL键下插入值为msgina32.dll的路径。



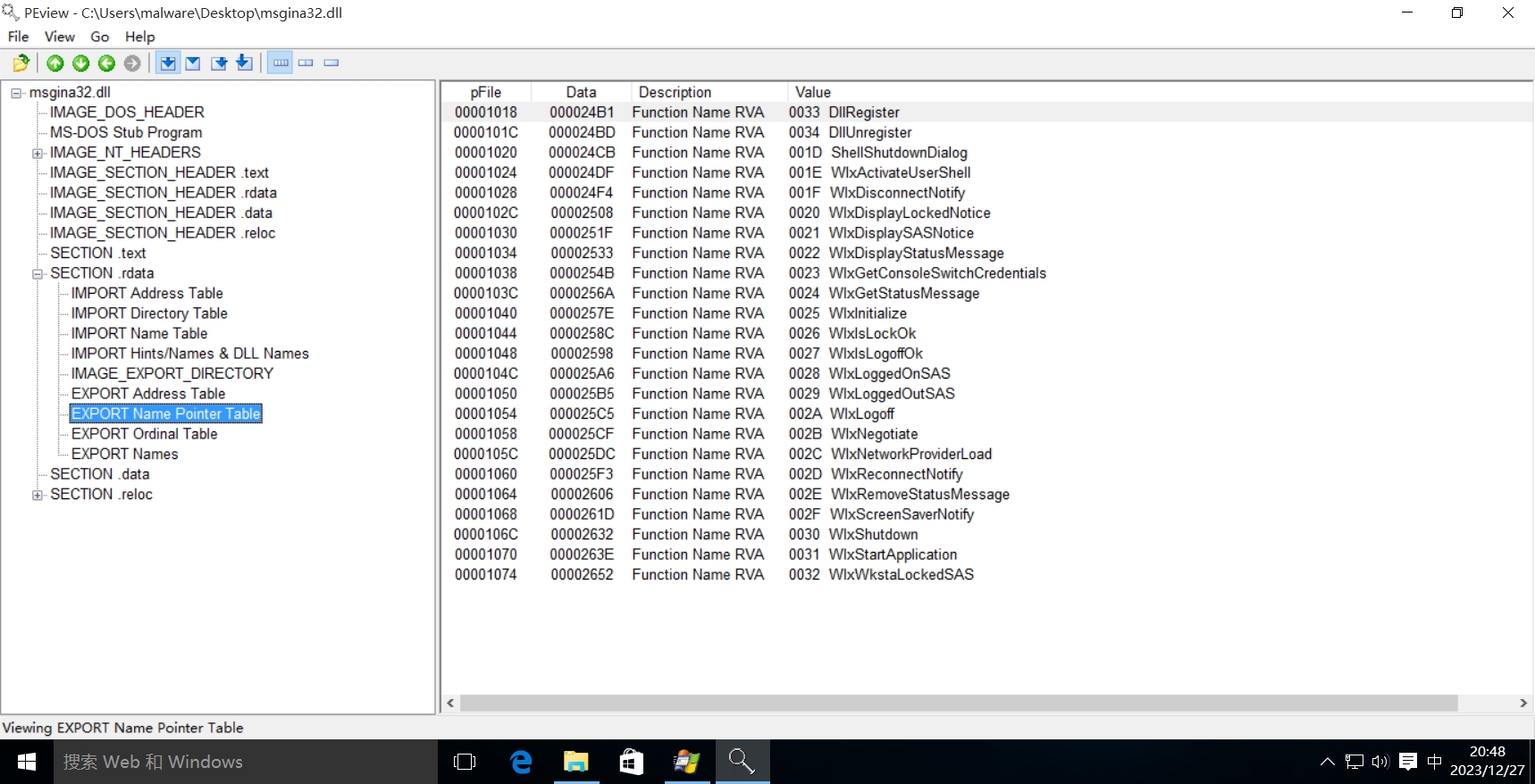
由此，恶意软件作为安装器，会将msgina32.dll通过添加到注册表从而安装为一个GINA DLL，它将会在系统启动时被Winlogon加载，从而实现开机自启。

1. 恶意软件如何窃取用户凭据的

接下来我们分析msgina32.dll文件。首先使用strings命令来查看其中包含的字符串，结果如下：



然后使用PE View来检测软件的导出，可以发现许多以Wlx开头的函数。



接下来使用IDA Pro来查看msgina32.dll。

图形用户界面, 应用程序

描述已自动生成

从IDA的反编译结果可以看到，msgina32.dll首先比较fdwReason参数和1，fdwReason参数用来指示DLL的入口点被调用的原因，而1代表DLL\_PROCESS\_ATTACH，指DLL是通过进程启动或使用LoadLibrary加载而被调用的。如果特定的DllMain在DLL\_PROCESS\_ATTACH期间被调用，GetSystemDirectoryW将被调用。恶意软件通过调用LoadLibraryW来获取msgina.dll的句柄，然后将其保存到全局变量hModule中。由于msgina32.dll正在拦截Winlogon和msgina.dll之间的通信，它必须正确地调用msgina.dll中函数，以便系统继续正常运行。

所以，恶意软件通过执行GINA拦截来窃取用户凭据。这个msgina32.dll文件可以拦截提交给系统进行身份验证的所有用户凭据。

1. 恶意软件如何处理被盗凭据

接下来分析msgina32.dll的导出函数。首先来分析WlxLoggedOnSAS函数，该函数反汇编结果如下：

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

WlxLoggeedOnSAS将参数字符串“WlxLoggedOnSAS”传递给sub\_10001000函数，该函数内部通过调用GetProcAddress函数来获取msgina.dll中的WlxLoggedOnSAS的地址，然后返回msgina32.dll的WlxLoggeedOnSAS函数中。在msgina32.dll的WlxLoggeedOnSAS的函数中，通过jmp eax来跳转到msgina.dll中的WlxLoggeedOnSAS函数。通过jmp而不是call将不会修改栈帧，这样msgina.dll的WlxLoggeedOnSAS的函数会直接返回到Winlogon。

图形用户界面, 文本, 应用程序

描述已自动生成

其他导出函数与以上分析的函数结构类似。但是WlxLoggedOutSAS函数包含了其他代码，该函数当用户登出系统时被调用。

图形用户界面

中度可信度描述已自动生成

WlxLoggedOutSAS函数通过call来调用msgina.dll中WlxLoggedOutSAS函数，并且在调用后还调用了sub\_10001570函数。

图形用户界面

中度可信度描述已自动生成

该函数中调用了vsnwprintf函数，该函数来填充格式字符串。然后调用了wfopen函数来打开msutil32.sys文件。然后调用了wstrdate函数和wstrtime函数来格式化日期和时间。最后调用fwprintf来格式化字符串。所以msutil32.sys文件用来记录被盗凭据，而不是一个驱动。

图形用户界面

中度可信度描述已自动生成

图形用户界面

中度可信度描述已自动生成

1. 如何使用该恶意软件在测试环境中获取用户凭据

综合以上信息，我们需要首先运行Lab11-01.exe来将msgina32.dll加入注册表，之后重启使得msgina32.dll被运行，然后登录和退出系统，这时恶意软件会记录下用户登录凭据。

# Lab 11-2

1. 这个DLL恶意软件的导出是什么

使用IDA Pro打开Lab11-02.dll，并切换到Exports窗口，可以看到Lab11-02.dll只有一个名为installer的导出函数。

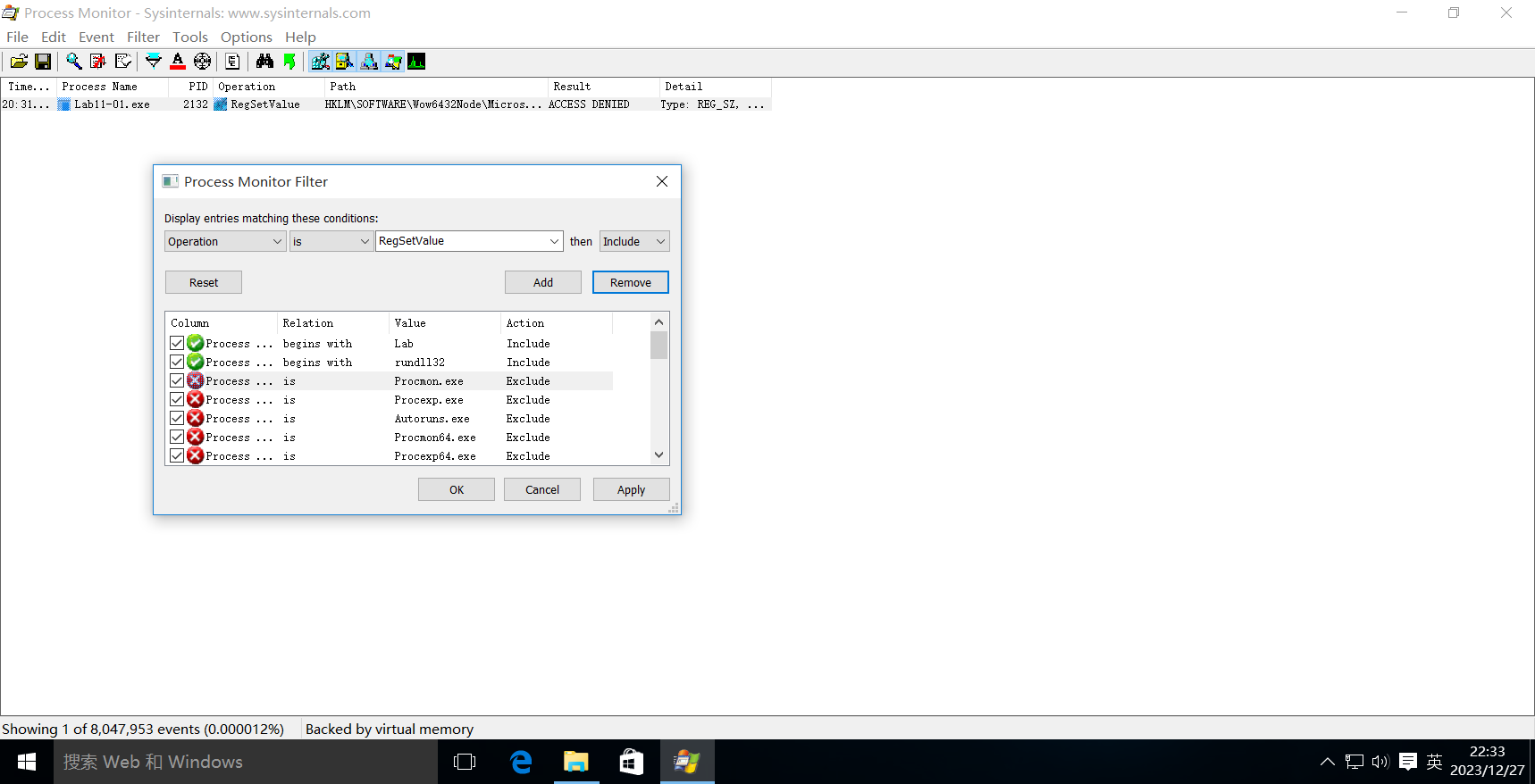
图形用户界面, 文本, 应用程序, 电子邮件

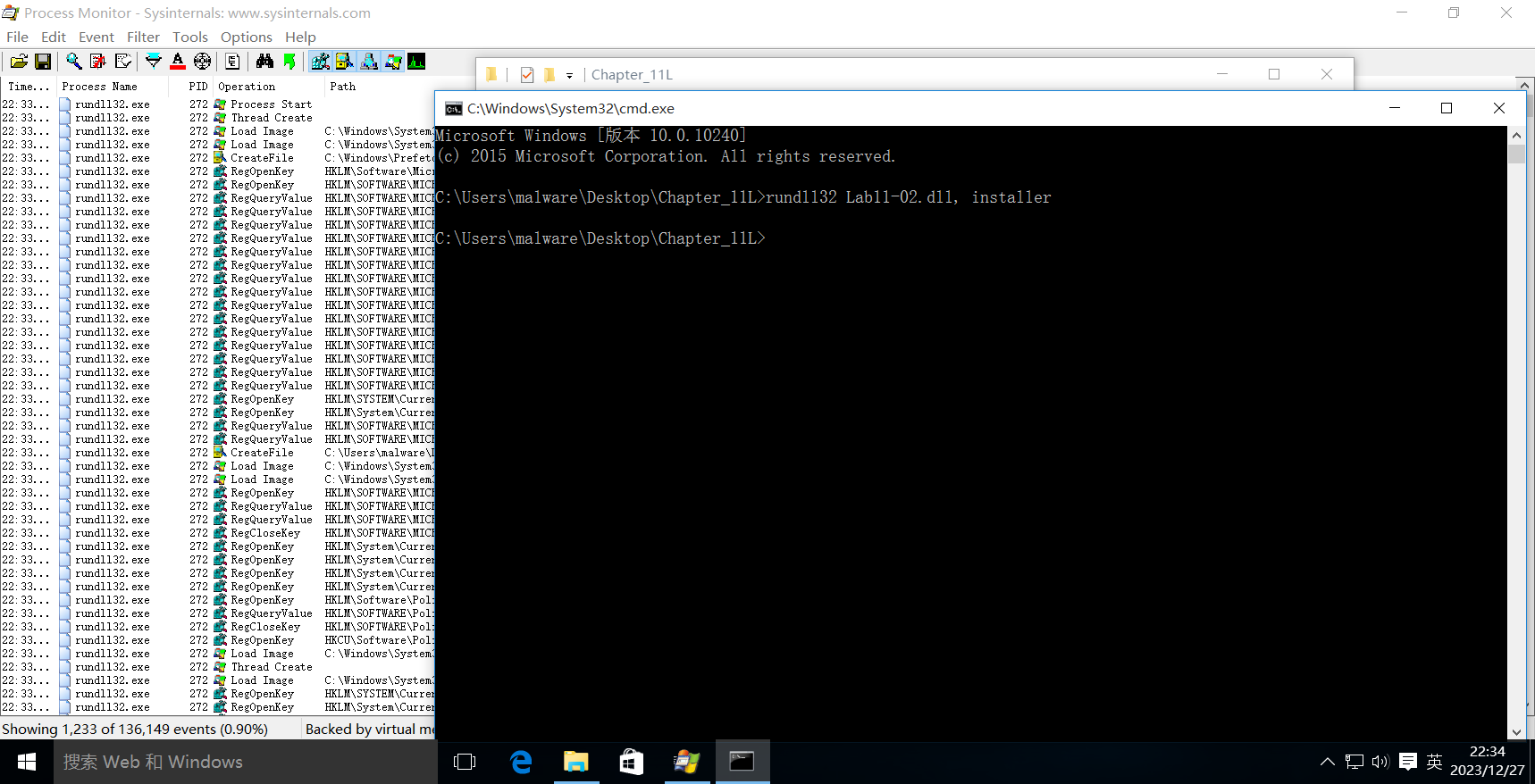
描述已自动生成

1. 使用rundll32.exe安装此恶意软件后会发生什么

打开procmon，然后在命令行中输入：rundll32.exe Lab11-02.dll, installer来运行Lab11-02.dll。

然后再procmon中设置过滤条件为：进程名rundll32.exe，结果如下：





我们可以看到恶意软件在系统目录创建了spoolvxx32.dll。经过进一步分析Lab11-02.dll和spoolvxx32.dll其实是同一个文件。

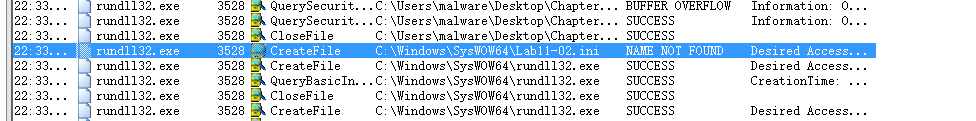


同时我们发现恶意软件将spoolvxx32.dll添加到AppInit\_DLL的列表，这会让该恶意软件被加载到使用User32.dll的每个进程中。

图片包含 应用程序

描述已自动生成

最后，我们发现恶意软件试图打开Lab11-02.ini，但是没有找到该文件。



1. 为了正确地安装恶意软件，Lab11-02.ini必须位于何处

经过以上分析，应该复制ini文件到Windows系统目录，以便恶意软件访问它。

1. 如何安装此恶意软件以实现持久性

经过以上分析，恶意软件将spoolvxx32.dll添加到AppInit\_DLL的列表，这会让该恶意软件被加载到使用User32.dll的每个进程中。

1. 这个恶意软件使用了什么用户空间rootkit技术

使用IDA Pro查看installer函数。该函数将恶意软件复制到系统目录下且更名为spoolvxx32.dll，并且将其添加到注册表项AppInit\_DLLs下。

图示

描述已自动生成

接下来查看DllMain函数。该函数首先判断DLL被调用的与原因是否为DLL\_PROCESS\_ATTACH。且该函数只有在DLL\_PROCESS\_ATTACH情况下才能继续执行。

文本

描述已自动生成

接下来恶意程序尝试将系统目录和“//Lab11-02.ini”拼接来尝试打开文件，如果打开成功，则读取其中的内容到byte\_100034A0中。

图形用户界面, 文本, 应用程序

描述已自动生成

接下来恶意函数调用了sub\_100010B3函数，该函数似乎是一个解码函数。接下来使用OllyDbg来验证我们的猜测。在0x100016CA处设置一个断点。然后可以在栈上看到函数的结果为“billy@malwareanalysisbook.com”

图形用户界面, 文本

描述已自动生成

接下来我们分析sub\_100014B6函数。

借用书上的图，我们来看一下函数的原理。

图示

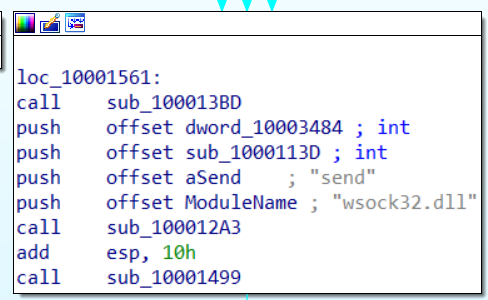
描述已自动生成

该图表明，恶意软件在send函数中插入了一段内联代码。该函数首先调用了函数sub\_10001075，该函数中又调用了GetModuleFileNameA函数，用来获取加载当前DLL的应用程序路径。

图形用户界面, 文本, 应用程序

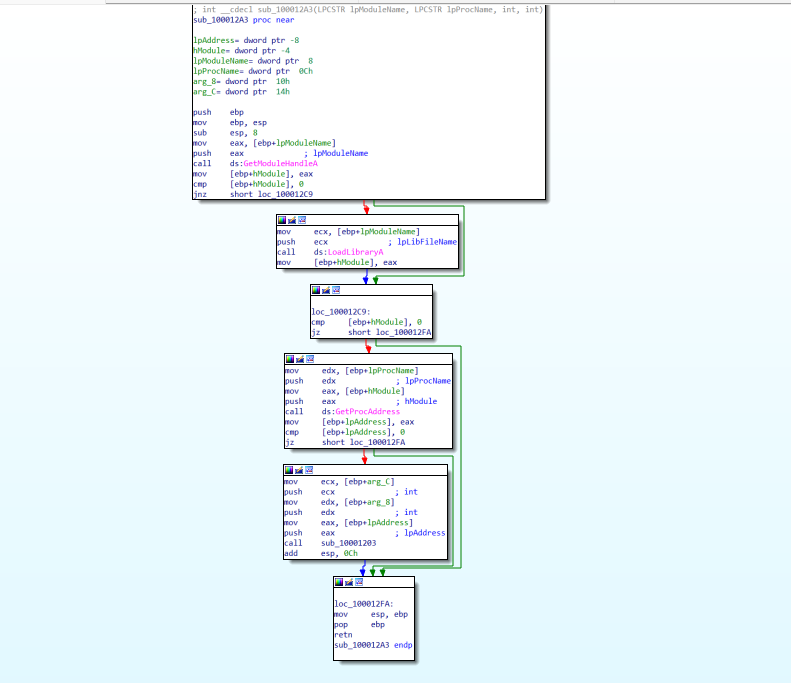
描述已自动生成

接下来，比较进程名THEBAT.EXE、OUTLOOK.EXE和MSIMN.EXE。如果当前进程名不为以上，则恶意软件退出，否则执行以下代码：



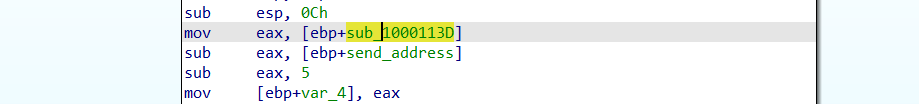
分析sub\_100013BD函数，该函数暂停当前进程的所有线程；而sub\_10001499函数恢复所有线程。这种操作在做出影响当前执行的改变时很常见，如更改内存或安装内联hook。

接下来分析sub\_100012A3函数。



该函数内部使用GetModuleHandleA来获取wscok32.dll的句柄。然后通过调用GetProcAddress来获取send函数的地址。接下来调用了sub\_10001203函数。

该函数首先计算了send函数和sub\_1000113D函数之间的偏移，然后又减去了5字节。这五字节空间为jmp sub\_1000113D指令留出空间。



接着函数分配了一段内存空间，并将代码前5字节保存起来。接着填充jmp指令，其中var\_4是偏移地址。

图片包含 图形用户界面

描述已自动生成

调用VirtualProtect函数将内存属性改为可读写执行，得以修改send函数的指令。

1. 钩子代码的作用是什么

钩子函数为sub\_1000113D，该函数首先会检查传出数据包是否是包含RCPT TO:的电子邮件，如果找到此字符串，则会添加一个包含恶意电子邮件帐户的附加RCPT TO行。

图形用户界面, 文本, 应用程序

中度可信度描述已自动生成

1. 该恶意软件攻击哪个进程，为什么

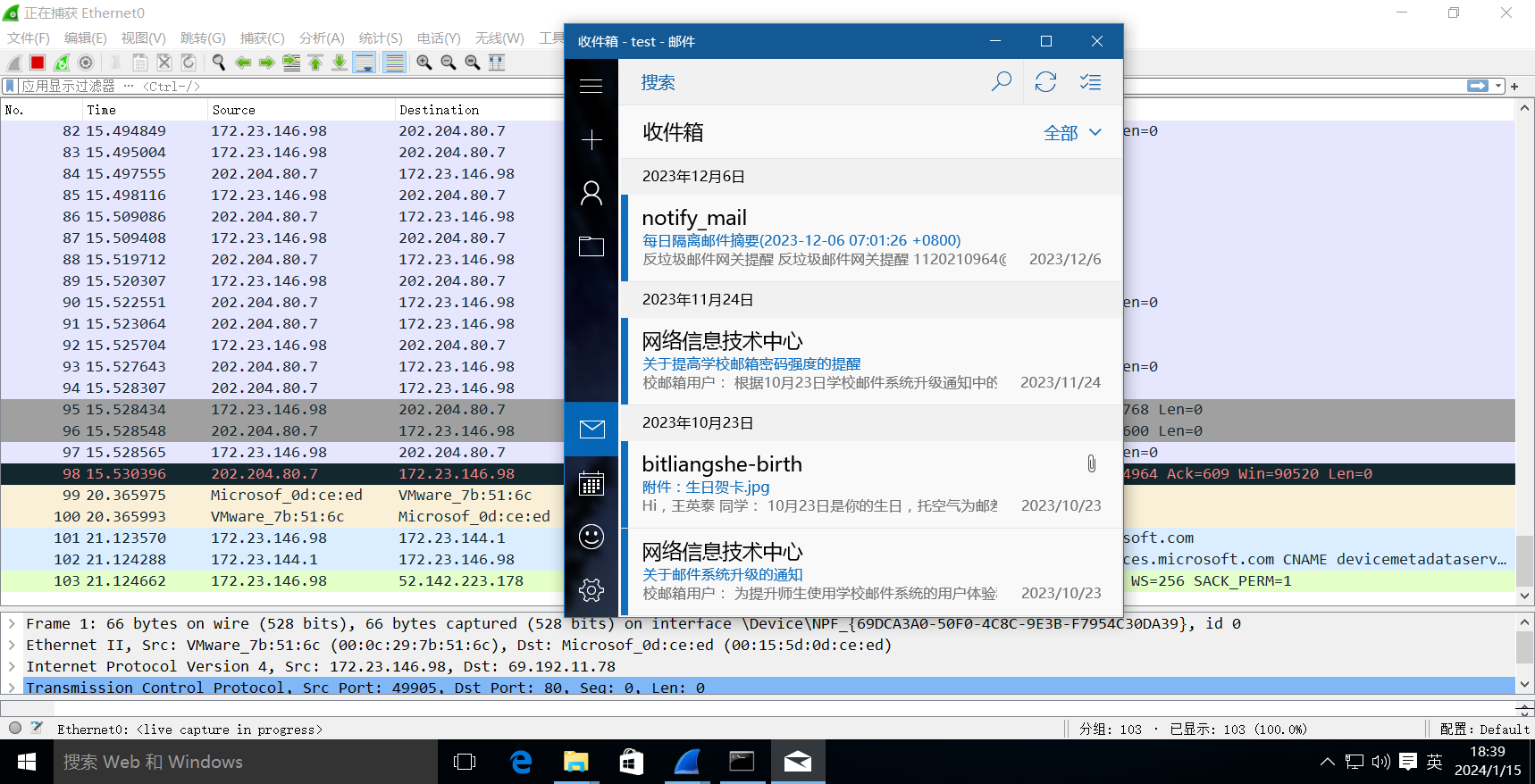
经过以上分析，该恶意软件攻击MSIMN.exe、THEBAT.exe和OUTLOOK.exe进程。因为他们是邮件客户端。恶意软件将会在他们内安装hook。

1. .ini文件的意义是什么

经过以上分析，ini文件存储的是加密后的邮箱地址。

1. 如何使用Wireshark动态捕获此恶意软件的活动

由于Windows10系统和此次被攻击的邮件客户端不同，所以无法实际体验到被攻击的效果。但是我过程其实是一样的，首先通过如下命令来运行恶意软件“rundll32.exe Lab11-02.dll,installer”，然后将Lab11-02.ini复制到C:\Windows\System32\目录下。接着启动WireShark，然后发送邮件，接着检测WireShark捕获的数据包。



# Lab 11-3

1. 使用基本静态分析可以发现哪些有趣的分析结果

使用strings指令来查看Lab11-03.exe中的字符串，可以发现其中包括“inet\_epar32.dll”、“net start cisvc” 等字符串，这暗示着这个恶意软件可能会启动系统上的CiSvc索引服务。



使用strings命令来查看Lab11-03.dll中包含的字符串。可以看到其中包含“C:\WINDOWS\System32\kernel64x.dll”、“GetAsyncKeyState”、“GetForegroundWindow”、“zzz69806582”，这暗示这个恶意软件可能是一个键盘记录器，它将键盘输入记录在kernel64x.dll中。

文本

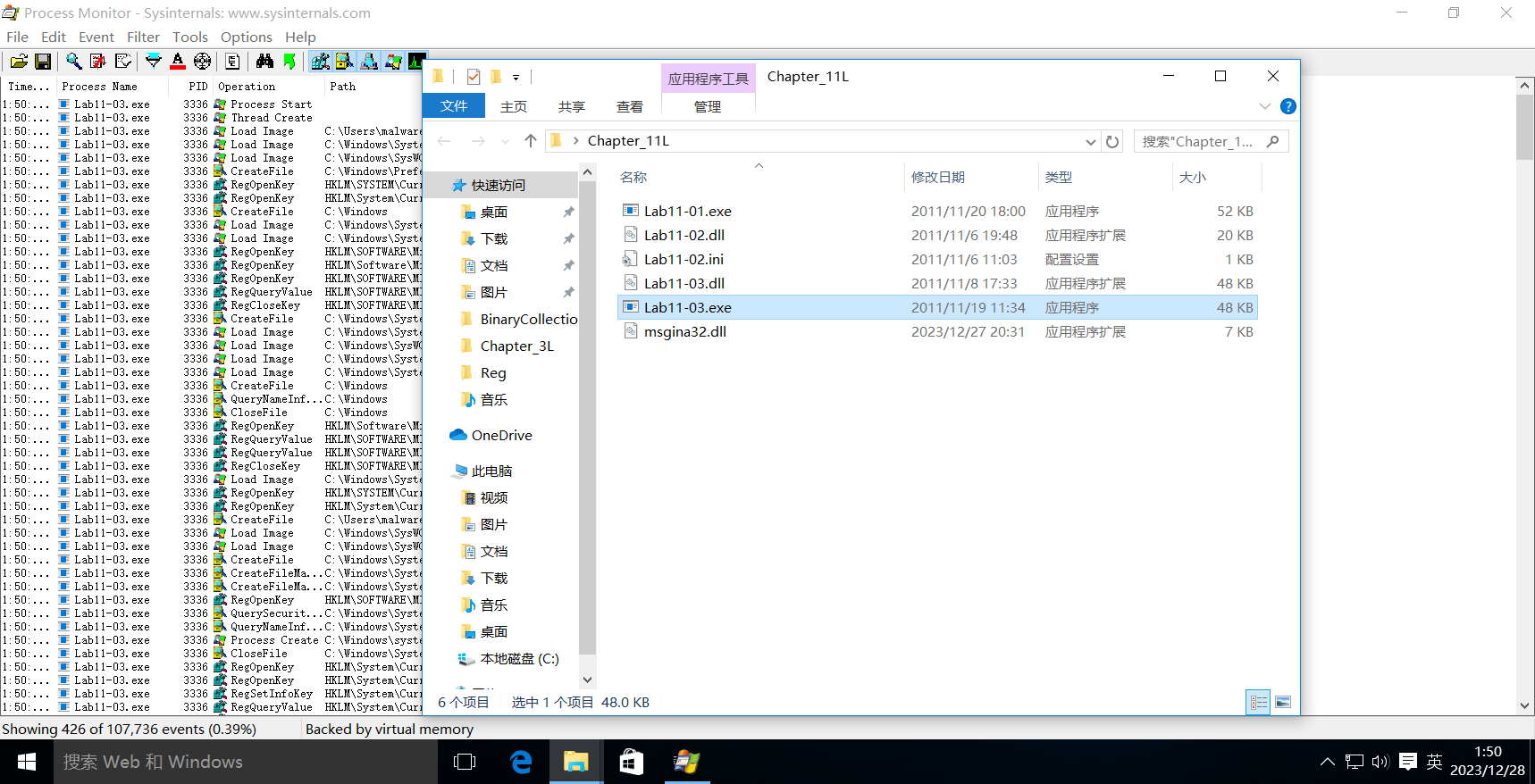
中度可信度描述已自动生成

图形用户界面, 文本, 应用程序

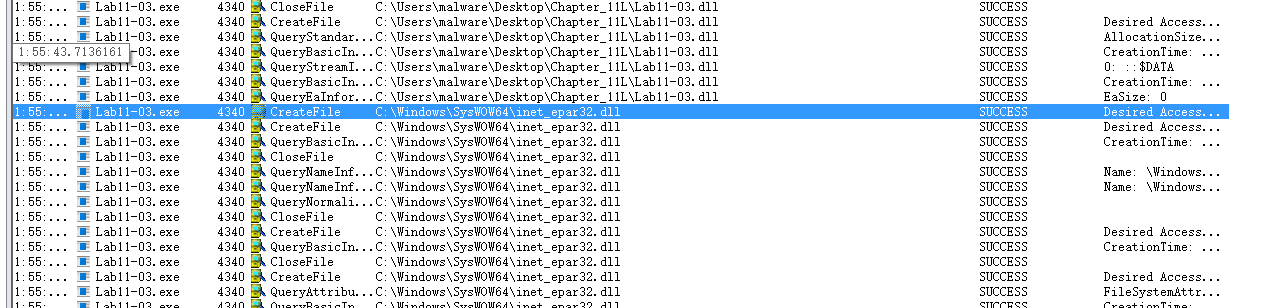
描述已自动生成

1. 运行此恶意软件时会发生什么

启动procmon，并设置过滤条件为：进程名为“Lab11-03.exe”，之后启动Lab11-03.exe，结果如下：



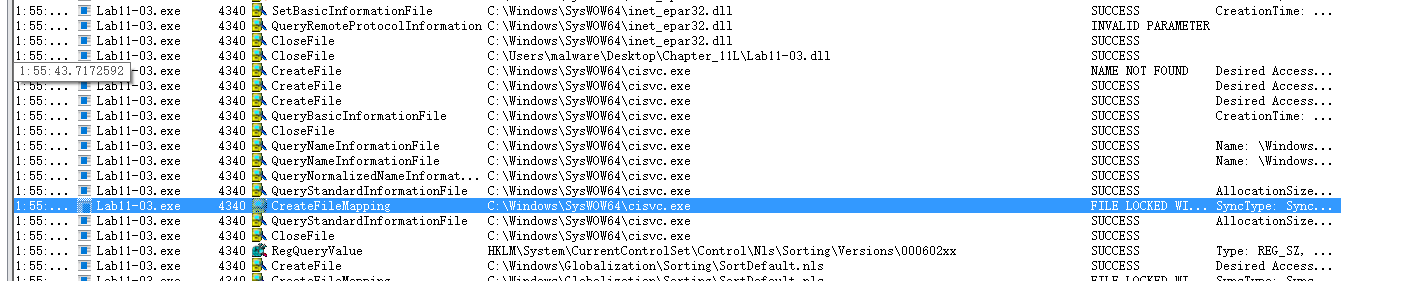
经过查看，我们发现恶意软件在系统目录下创建了inet\_epar32.dll，经过MD5检查，该文件是Lab11-03.dll的复制品。



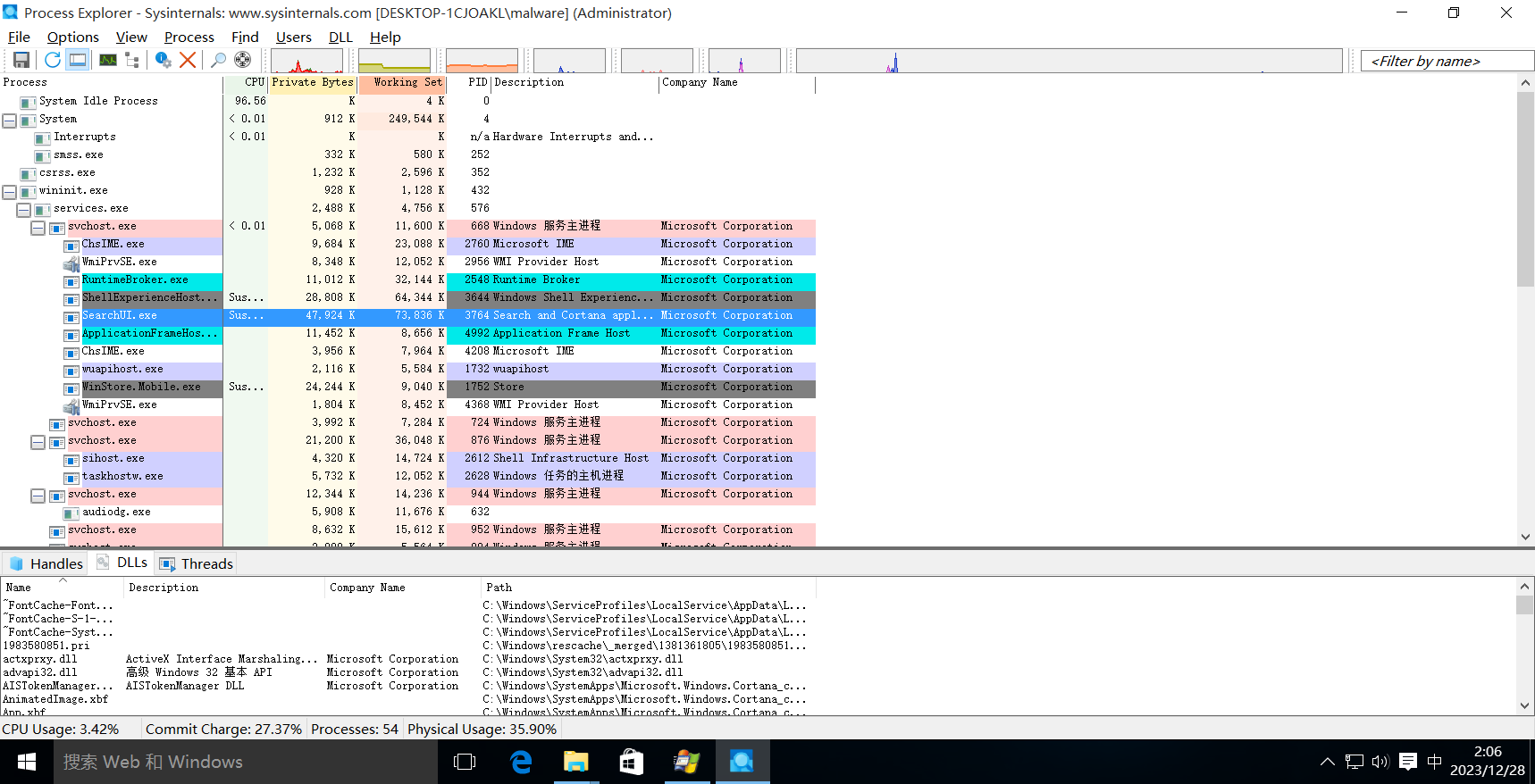
电脑萤幕的截图

描述已自动生成

恶意软件打开了cisvc.exe文件，但是我们没有看到WriteFile操作。



使用Process Explorer，我们发现cisvc.exe正在运行。并且可以发现索引服务已经启动。



由于我们猜测该恶意软件是一个键盘记录器，所以我们尝试在记事本中写入字符串，然后发现procmon中增加了创建kernel64x.dll文件的事件。

使用记事本打开kernel64x.dll，发现关于键盘输入的内容。

1. Lab11-03.exe如何持久安装Lab11-03.dll

接下来使用IDA Pro来打开Lab11-03.exe。可以看到main函数中首先将Lab11-03.dll到inet\_epar32.dll，然后将字符串“C:\WINDOWS\System32\cisvc.exe”传递给sub\_401070函数，最后调用net start cisvc命令来启动服务。



接下来我们分析sub\_401070函数。

图形用户界面, 应用程序

描述已自动生成

可以发现恶意软件调用CreateFileA、CreateFileMappingA、MapViewOfFile将cisvc.exe文件映射进了内存。任何对文件的修改通过调用UnmapViewOfFile都会被写入磁盘。所以我们没有看到WriteFile函数。

接下来通过分析以下代码发现恶意程序将byte\_409030处复制312字节到内存映射地址。

文本

描述已自动生成

byte\_409030处的原始数据通过按下“C”键来得到反汇编。这种技术通常用作进程注入。

日程表

描述已自动生成

在312字节的末尾处，可以发现两个字符串“C:\WINDOWS\System32\inet\_epar32.dll”、“zzz69806582”。这暗示这个shellcode加载这个dll，并且调用它的导出函数。

图片包含 文本

描述已自动生成

接下来比较恶意软件运行之前的cisvc.exe和运行之后的cisvc.exe。通过比较发现在PE头只有两个字节发生改变。然后加载到PEView发现两者的入口点不同。这导致了被注入的cisvc.exe在原来入口点之前执行了shellcode。

接下来分析被注入的cisvc.exe，首先先调用了LoadLibray去加载inet\_epar32.dll，然后调用GetProcAddress来获取zzz69806582函数的地址。之后调用zzz69806582。之后跳转到\_wmainCRTStartup，从而保证服务可以正常运行。

综上，恶意软件会重定位dll的入口地址来运行shellcode，这个shellcode来加载DLL。

1. 恶意软件会感染哪个Windows系统文件

从以上分析可知，恶意软件会感染cisvc.exe来加载inet\_epar32.dll，然后调用zzz69806582函数。

1. Lab11-03.dll是做什么的

接下来分析Lab11-03.dll，将其加载到IDA Pro。查看其zzz69806582函数。

文本

中度可信度描述已自动生成

之后查看StartAddress所指向的函数。

文本

描述已自动生成

首先恶意程序会创建信号量来保证一次只能运行一个恶意软件实例。接着，恶意程序会创建一个名为kernel64x.dll的文件。

图形用户界面, 文本, 应用程序

描述已自动生成

接着，调用了sub\_10001380函数。这个函数包含了一个循环，这个循环会调用“GetAsyncKeyState”、“GetForegroundWindow”、“WriteFile”。这是用户空间键盘记录器的例子。

综上，Lab11-03.dll通过它的zzz69806582来实现键盘记录的作用。

1. 恶意软件将收集的数据存储在哪里

综合以上分析，恶意软件将收集的数据放入kernel64x.dll文件中。

# 实验总结

本次实验，让我进一步巩固了之前学到的知识，尤其是IDA Pro的使用更加熟悉。然而，由于操作系统的限制，我无法实际体验实验2的攻击效果。最后，这章实际分析了一些常见的恶意软件，如键盘记录器等，其中使用的方法是我以前从来没有想过的，看了他们的方法后让我耳目一新。他们的攻击方法巧妙，让我学习到了很多。