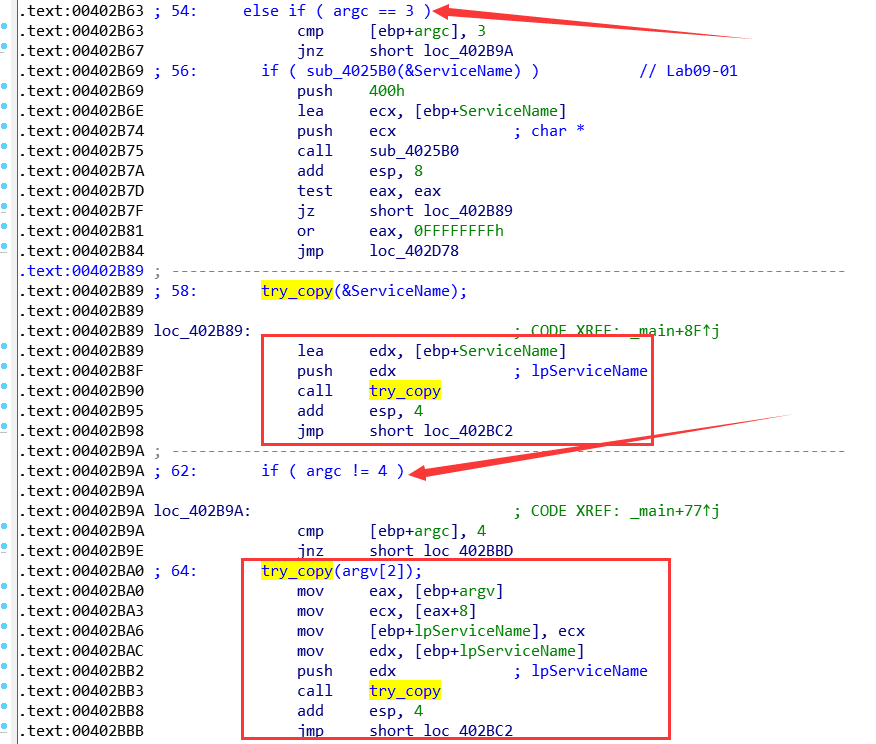
**Lab 09 – All analysis**

**1120162015 李博**

**Lab 9-1**

1. **How can you get this malware to install itself?**

通过对程序的静态分析，我发现在mian函数中有两个分支可以进入copyfile函数所在的子函数，如下图。



第一个分支进入条件为

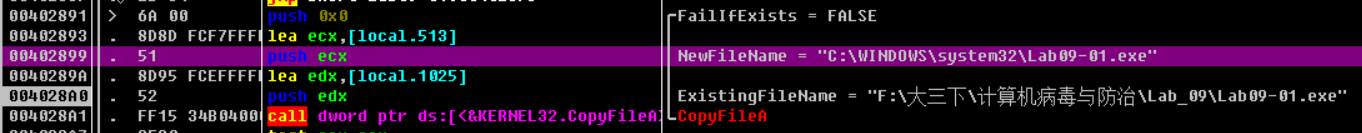
**argc==3并且argv[argc-1]==”abcd”,argv[1]==”-in”**。

该分支的作用即**安装自身**至C:\Windows\System32。

通过ollydbg中**调试->参数**来输入程序的参数



如下图，为ollydbg动态调试跟进至复制文件函数的代码处。



**2. What are the command-line options for this program? What is the pass-word requirement?**

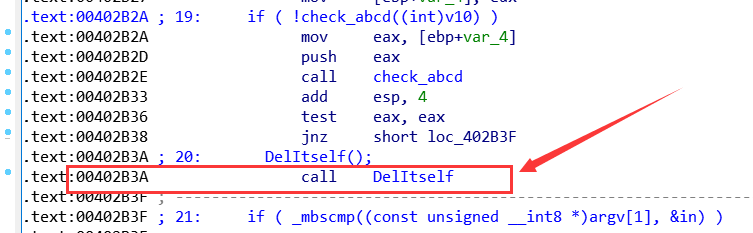
该程序的命令行参数有四种，分别为-in（安装），-re（移除），-c和-cc。

需要的password为”abcd”，需输入在命令行的最后一个参数

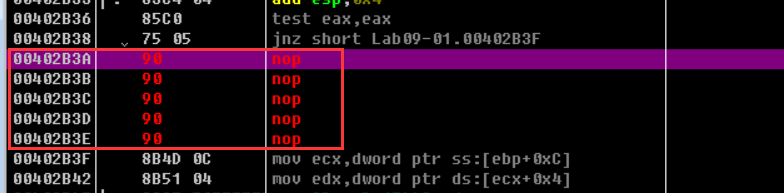
即argv[argc-1]==”abcd”。

1. **How can you use OllyDbg to permanently patch this malware, so that it doesn’t require the special command-line password?**

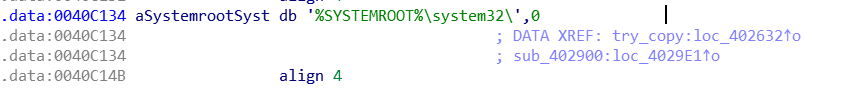
首先在IDA里定位至相应代码段，关键在**call DelItself**，位于0x 00402B3A。



由于该程序没有地址随机化，故在Ollydbg中可以直接定位至地址0x0402B3A，如下图，用nop（0x90）填充0x0402B3A-0x0402B3E之间的数据即可。



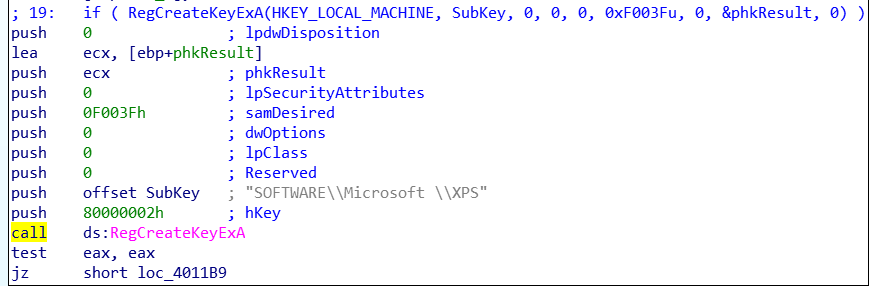
1. **What are the host-based indicators of this malware?**
2. **该字符串用于定位至系统盘**



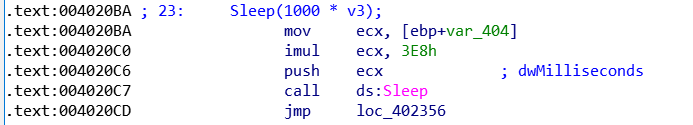
1. **该字符串用于调用命令行**



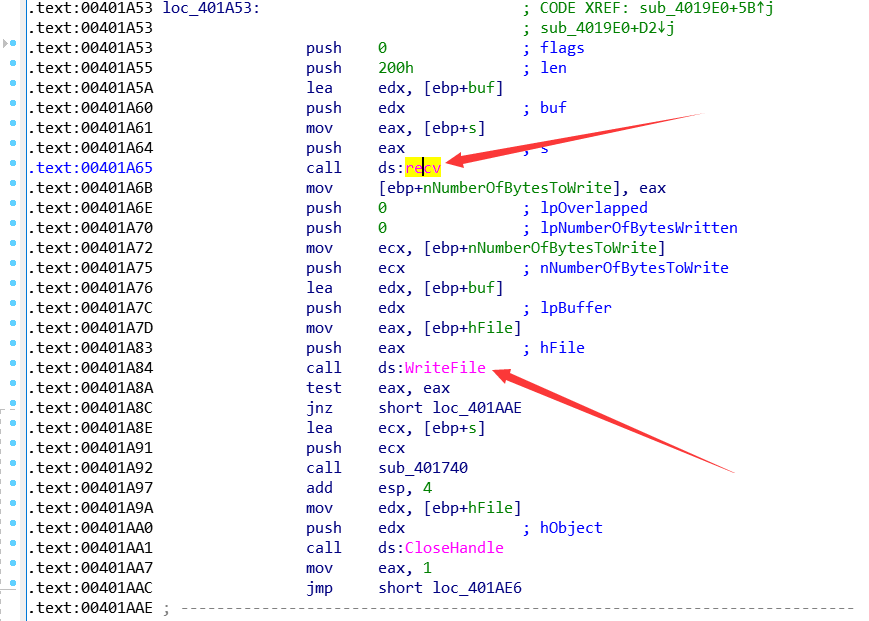
1. **创建注册表项**



1. **What are the different actions this malware can be instructed to take via the network?**
2. **休眠**



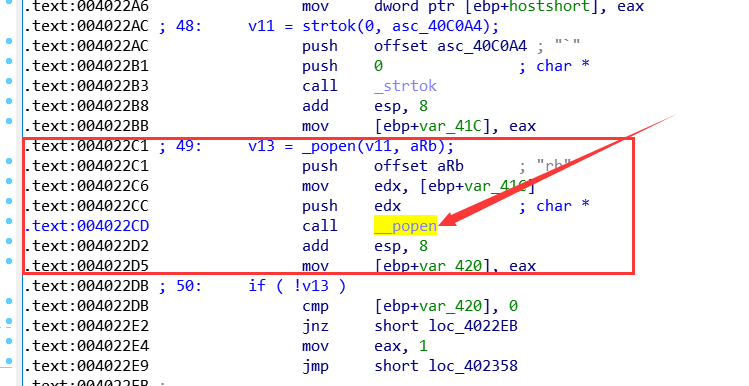
1. **接收文件**



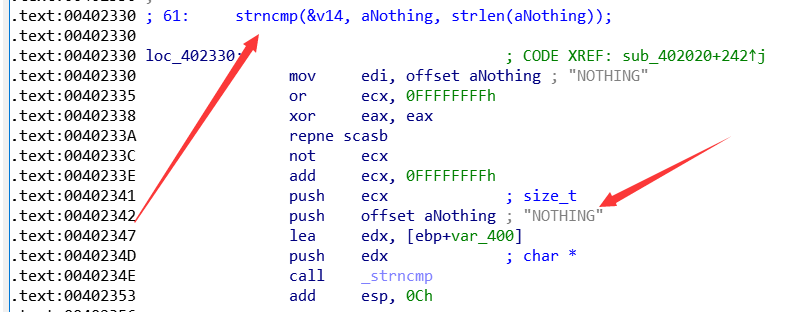
1. **上传文件**



1. **执行shell**



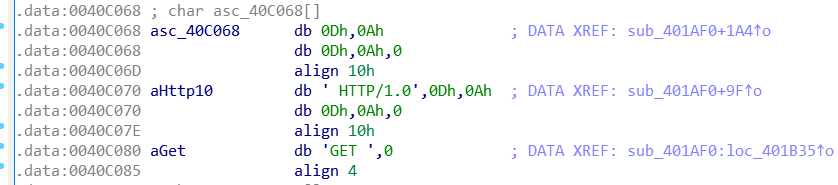
1. **无操作**



1. **Are there any useful network-based signatures for this malware?**
2. **通信的目标网址**



1. **通信方式**

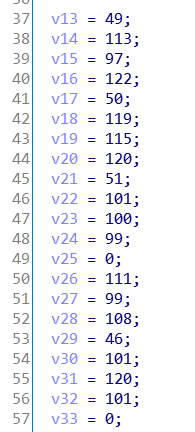
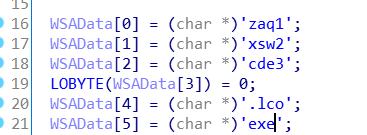


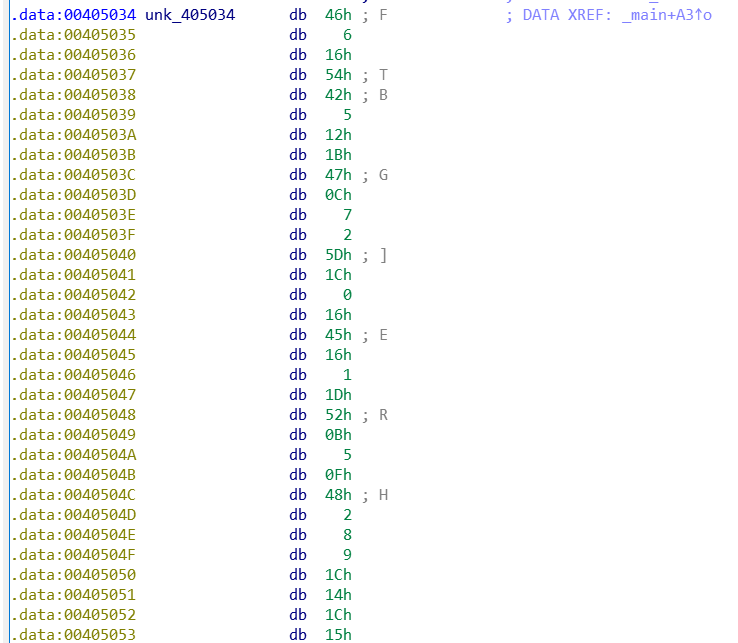




**Lab 9-2**

1. **What strings do you see statically in the binary?**
2. 在IDA中的变量上Y一下，可改变变量类型，如下图

 🡺 

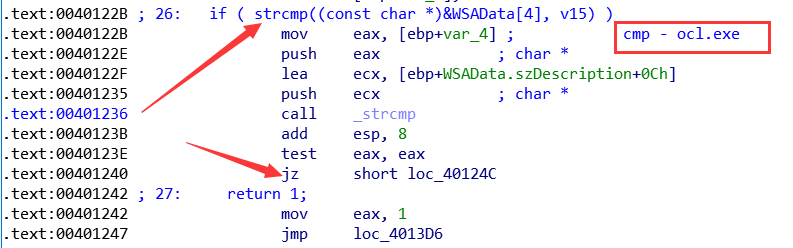


1. **What happens when you run this binary?**

什么也没发生。

1. **How can you get this sample to run its malicious payload?**

通过IDA静态分析可以发现，在main函数中，进行了文件名校验的过程，如下。



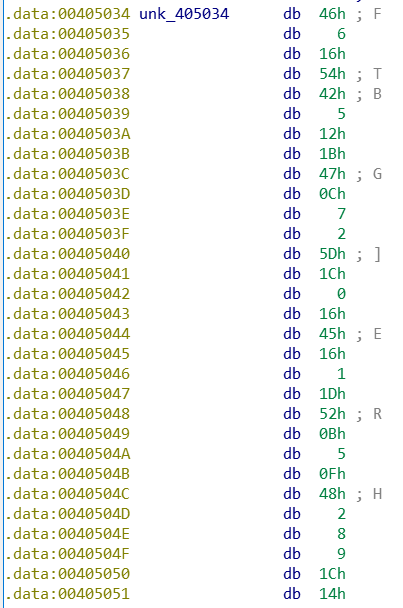
故将文件名改为**ocl.exe**即可。

1. **What is happening at 0x00401133?**

将一个字符串赋值至栈。

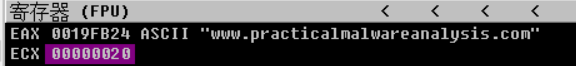
1. **What arguments are being passed to subroutine 0x00401089?**

参数一为在0x401133处放到栈上的字符串1qaz2wsx3edc

参数二为

1. **What domain name does this malware use?**

通过Ollydbg动态调试，直接在解密函数之后查看寄存器eax值如下。



1. **What encoding routine is being used to obfuscate the domain name?**

异或加密

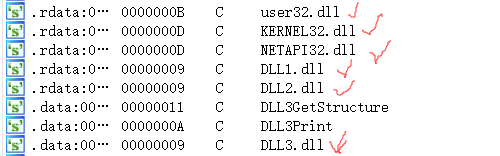


1. **What is the significance of the CreateProcessAcall at 0x0040106E?**

创建线程来启动命令行，并且通过设置showWindow标志位为0来隐藏shell。

**Lab 9-3**

1. **What DLLs are imported by Lab09-03.exe?**

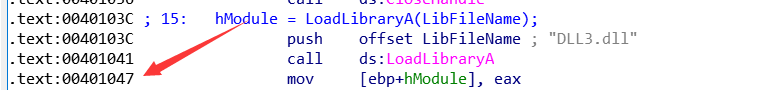


1. **What is the base address requested by DLL1.dll, DLL2.dll,and DLL3.dll?**

基地址为0x10000000

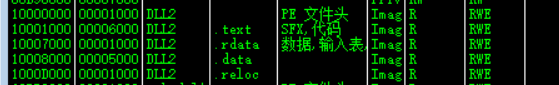
1. **When you use OllyDbg to debug Lab09-03.exe, what is the assigned based address for: DLL1.dll, DLL2.dll, and DLL3.dll?**

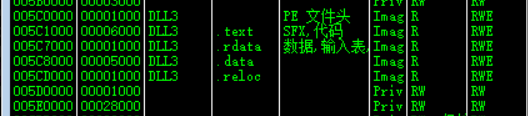
当程序运行至下图代码段时，DLL3被加载。



在Ollydbg中查看内存







1. **When Lab09-03.exe calls an import function from DLL1.dll, what does this import function do?**

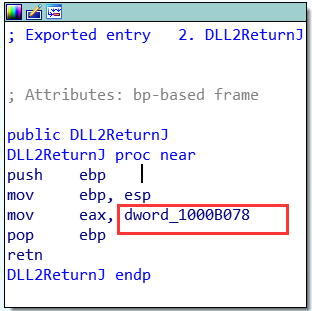
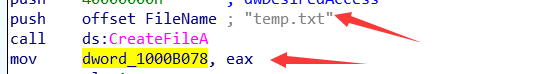
输出字符串和一个变量。

1. **When Lab09-03.exe calls WriteFile, what is the filename it writes to?**

文件名为temp.txt

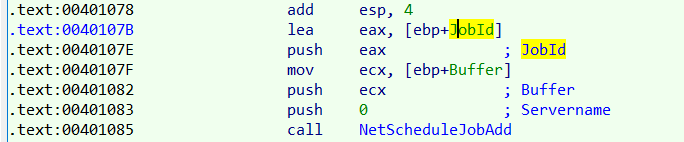
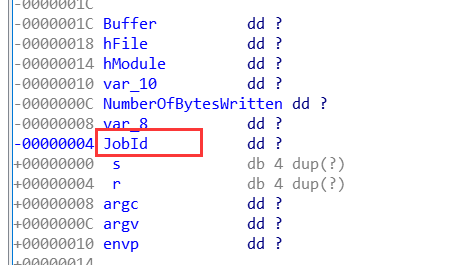


赋值过程如下



1. **When Lab09-03.exe creates a job using NetScheduleJobAdd, where does it get the data for the second parameter?**

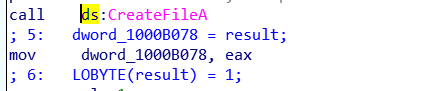
来自栈上的局部变量

1. **While running or debugging the program, you will see that it prints out three pieces of mystery data. What are the following:DLL 1 mystery data 1, DLL 2 mystery data 2, and DLL 3 mystery data 3?**
2. DLL 1 mystery data 1是GetCurrentProcessId的返回值，即进程标识符



1. DLL 2 mystery data 2是CreateFileA的返回值，即文件的句柄

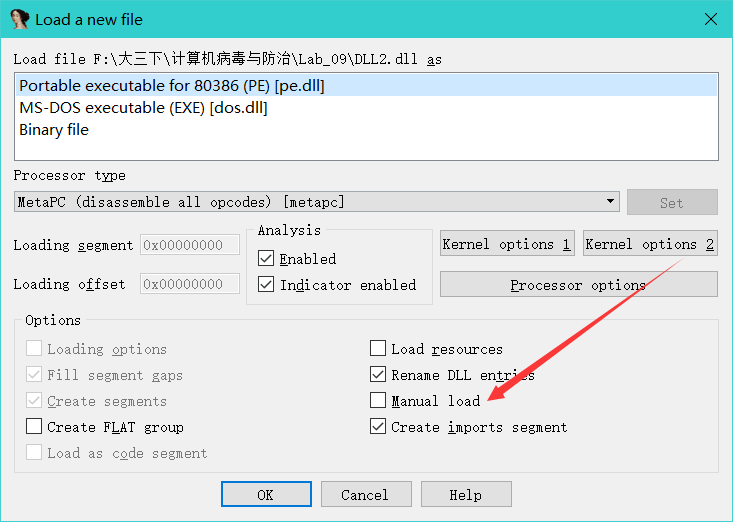


1. DLL 3 mystery data 3是字符串

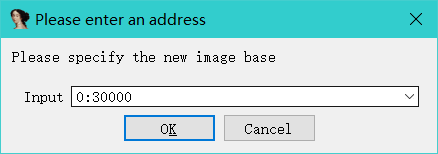
'ping [www.malwareanalysisbook.com](http://www.malwareanalysisbook.com)'的首地址

**8. How can you load DLL2.dll into IDA Pro so that it matches the load address used by OllyDbg?**

在加载dll时，选择manual load



接着将olldbg中的dll2的地址0x30000输入即可。



结果

