

**恶意代码分析与防治课程实验报告**

**实验六**

****

专 业 信息安全

学 号 2113662

姓 名 张丛

班 级 信息安全一班

1. **实验目的**

完成课本Lab6的实验内容，编写Yara规则，并尝试IDA Python的自动化分析，在此提交实验报告。

1. **实验原理**

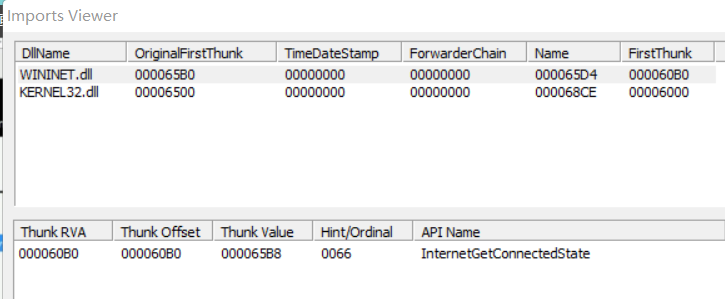
静态、动态工具的应用。

1. **实验过程**

### Lab 6-1

#### 由main函数调用的唯一子过程中发现的主要代码结构是什么？

PEiD查看导入表：



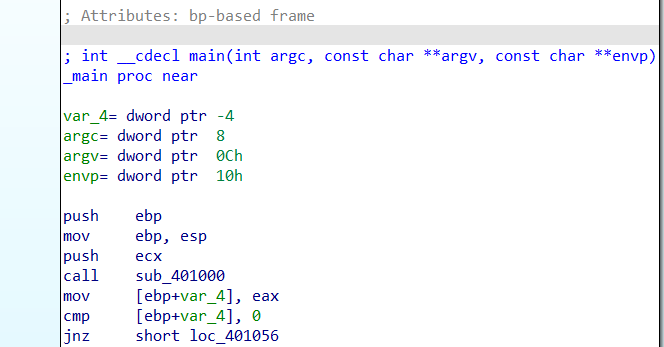
InternetGetConnectedState能获取本地系统的连接状态。

strings查看字符串：

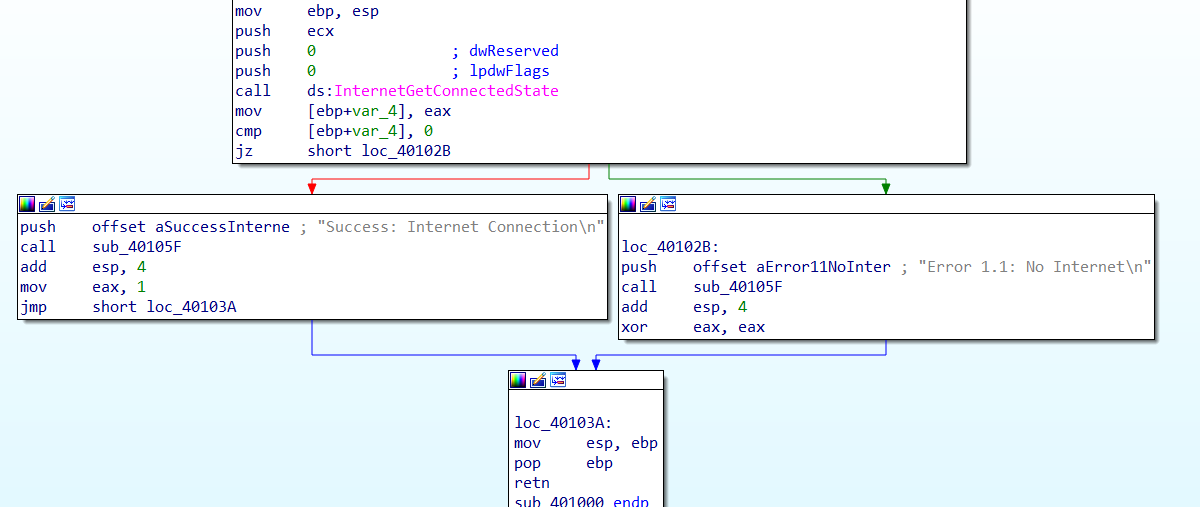


需要注意最后两个字符串，“No Internet”和”Internet Connection”可能与上面的导入函数有关。

IDA分析：



可以找到main函数唯一调用的call函数sub\_401000.



我们可以知道这是一个if语句分支结构。

#### 位于0x40105F的子过程是什么？

可以在上面的分支里找到40105F，在第一问中，无论分支走左边还是右边，都会执行40105F。

40105F把字符串压栈作为参数，可以猜测它是打印字符串的功能，即printf.

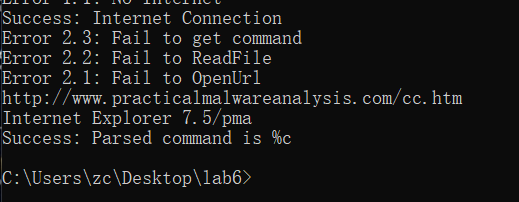
#### 这个程序的目的是什么？

程序调用了InternetGetConnectedState，如果找到可用连接，就打印"Success: Internet Connection\n"，如果没有找到，就打印 “Error 1.1: No Internet\n”。

### Lab 6-2

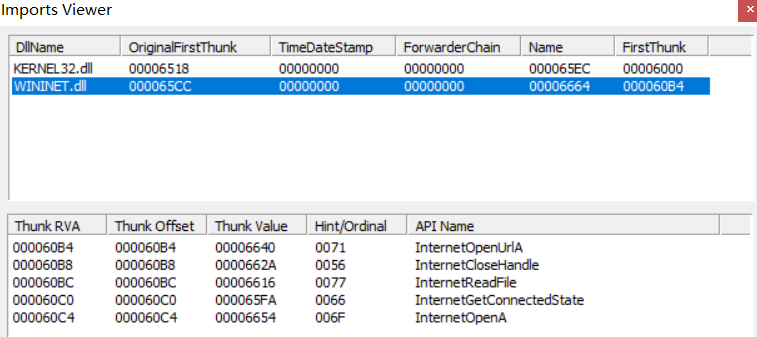
#### main函数调用的第一个子程序执行了什么操作？

stringa查看值得注意的字符串：



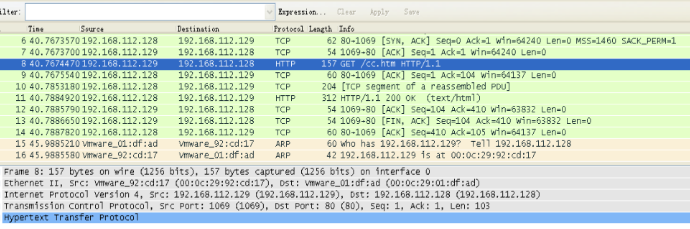
存在网址，以及表示网络状态的字符串。

查看导入表：



WININET.dll提供了通过网络实验htttp的api。

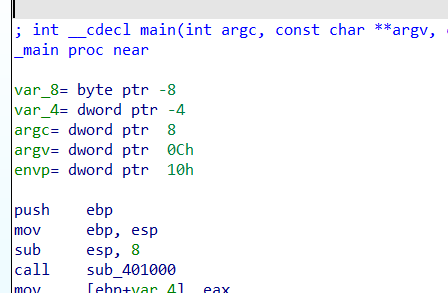
结合wireshark进行分析：



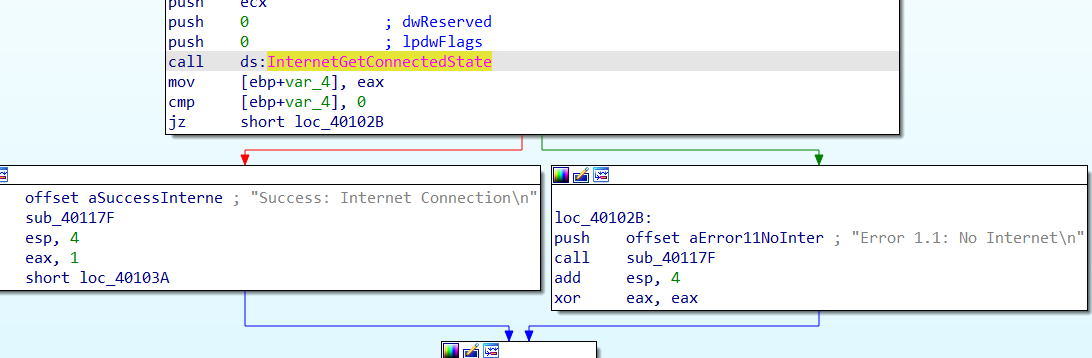
IMG_256

可以知道确实解析了上面的网址。

使用IDA:

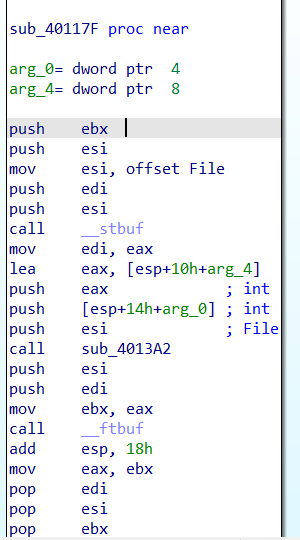


程序第一个call是sub\_401000.



可知这个子程序仍是一个if分支，调用InternetGetConnectedState，得到相应的连接状态，进入不同分支。

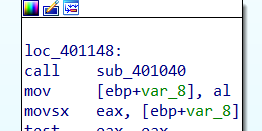
#### 位于0x40117F的子过程是什么？

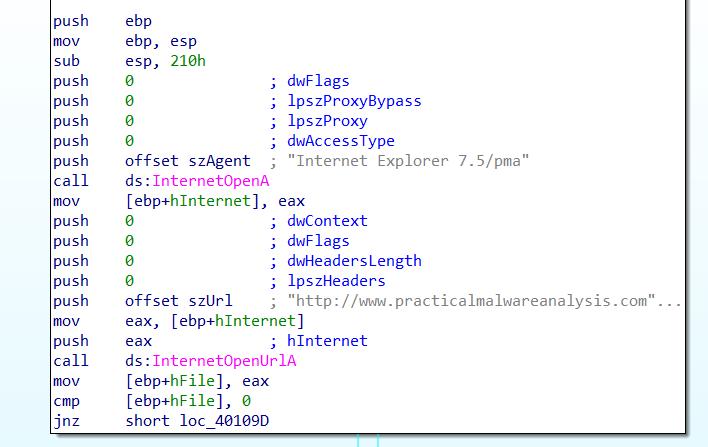


40117F有两个参数，压入了格式化字符，所以还在printf函数。

#### 被main函数调用的第二个子过程做了什么？

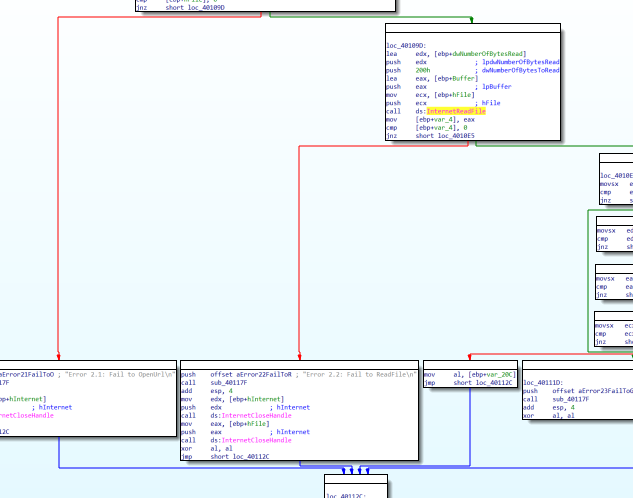
main第二个call是sub\_401040:





调用了InternetOpen和InternetOpenUrl。

下面的分支：



InternetOpenUrlA的返回值赋给hFile，判断是否为0，如果为0的话，直接关闭句柄退出了，如果不为零，判断InternetReadFile的值是否为0，然后判断得到的值的buffer，然后判断前四个字符是否是“<!–”然后打印出后面的字符，打印了html注释，否则打印出出错。

#### 在这个子过程中使用了什么类型的代码结构？

从上一题的图可以看出，代码结构是if分支里面又嵌套了一个if分支。

#### 在这个程序中有任何基于网络的指示吗？

strings查找字符串发现的网址：

<http://www.practicalmalwareanalysis.com/cc.htm>

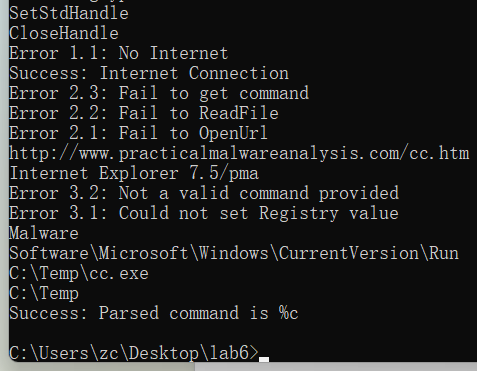
#### 这个恶意代码的目的是什么？

先判断是否有可用的网络，如果没有则停止运行，如果已连接网络则在http://www.practicalmalwareanalysis.com网址下载并解析HTML文件，然后打印字符串Success : Parsed command is加上注释正文字符，最后程序休眠60s然后终止运行。

### Lab 6-3

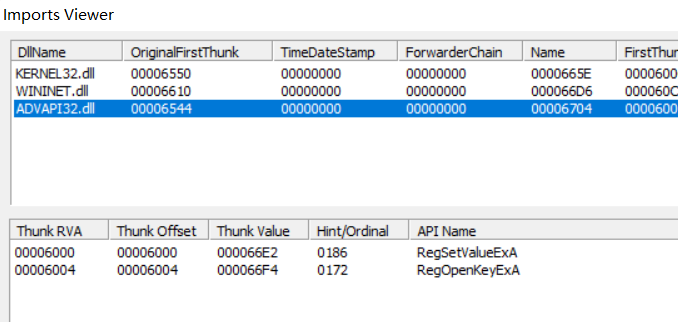
#### 比较在main函数与实验6-2的main函数的调用。从main中调用的新的函数是什么？

strings查看字符串：



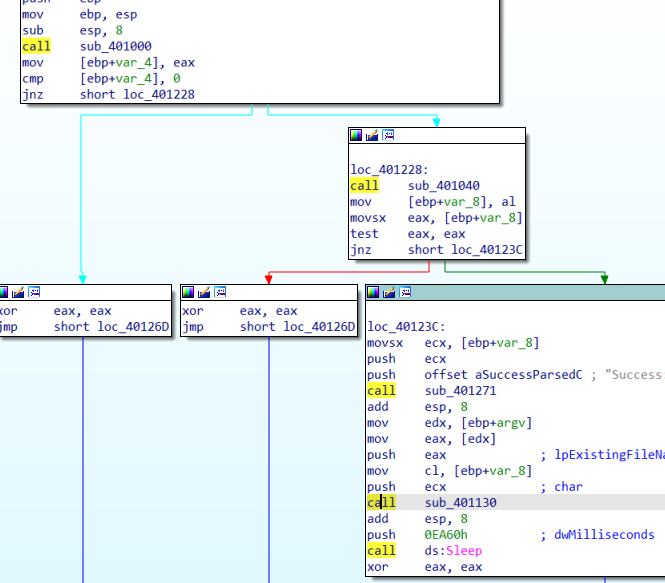
可见程序可能会修改注册表。

查看导入表：



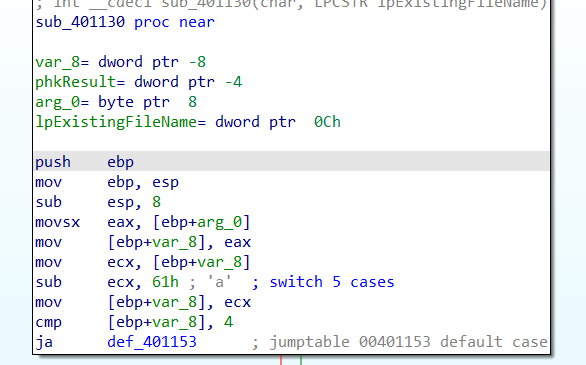
RegOpenKeyExA 函数一般与 RegSetValueExA一起用于往注册表中插入信息，在恶意代码将其自身或其他程序设置为随着系统开机就自启动以持久化运行时，通常会使用这两个函数。

IDA：



可见6-3对比6-1和6-1调用的心函数是sub\_401130。

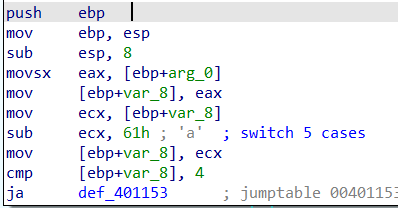
#### 这个新的函数使用的参数是什么？

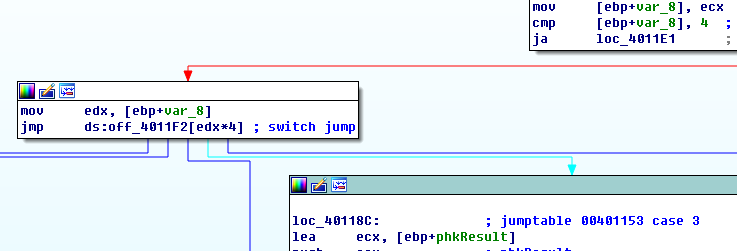


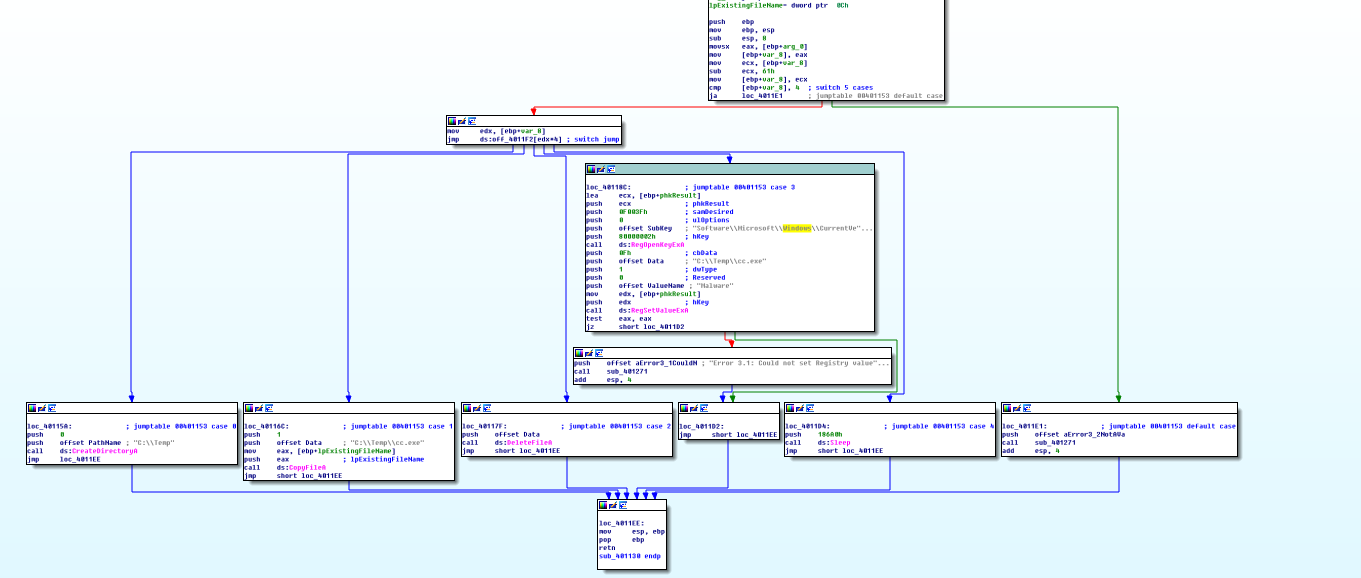
sub\_401040函数中返回的值作为了这个新函数的参数，而sub\_401040函数返回的是获取的Html中的注释的内容。

这个新的函数有两个参数，第一个是从HTML注释中解析来的指令字符，第二个参数是标准main函数中包含的参数中的argv[0]，也就是程序名本身。

#### 这个函数包含的主要代码结构是什么？







一条switch语句和一个跳转表。

#### 这个函数能够做什么？

可以根据case语句判断：

case 0：创建一个目录C:\Temp。

case 1:将自身复制到目录下并改名。“C:\Temp\cc.exe”。  
case 2:删除文件。  
case 3:创建一个自启动项。  
case 4:程序睡眠100秒钟。  
default: 打印错误，直接退出。

#### 在这个恶意代码中有什么本地特征吗？

文件路径"C:\Temp\cc.exe"

自启动项 Software\Microsoft\Windows\CurrentVersion\Run

#### 这个恶意代码的目的是什么？

程序先检查是否存在有效的Internet 连接。

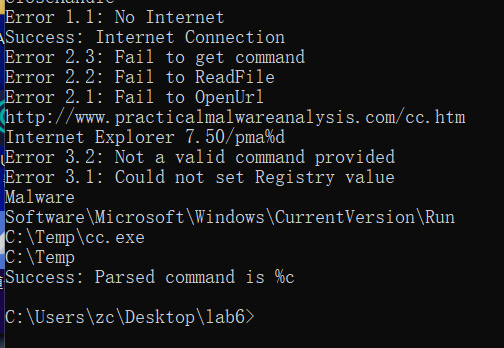
如果找不到，程序直接终止。

否则，该程序会尝试下载一个网页，该网页包含了一段以<!–开头的HTML注释。该注释的第一个字符被用于switch语句来决定程序在本地系统运行的下一步行为，包括是否删除一个文件、 创建一个目录、设置一个注册表run键、复制一个文件或者休眠100秒。

### Lab 6-4

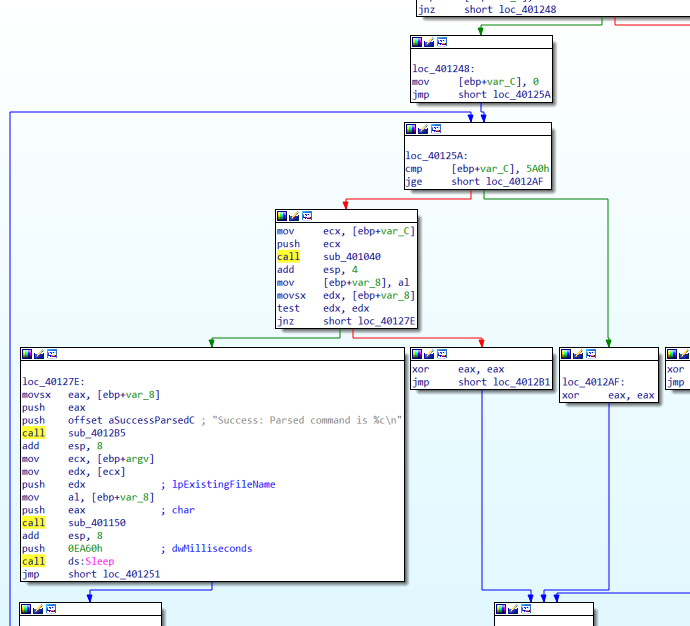
#### 在实验6-3和6-4的main函数中调用之间的区别是什么？

strings查看字符串：



会动态生成Internet Explorer 7.50/pma%d，其余类似。

IDA:



对调用的call进行分析：

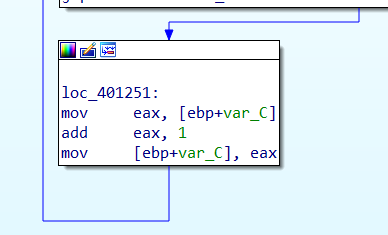
sub\_401000：检查网络连接

sub\_401040：解析HTML

sub\_401150：switch语句

sub\_4012B5：printf函数

#### 什么新的代码结构已经被添加到main中？



局部变量var\_C用于循环计数。

在1处赋值为0，在2处递增1，每次到了5就返回递增，因此这是一个循环结构。

当var\_C大于或者等于3处的5A0h时，循环停止。

#### 这个实验的解析HTML的函数和前面实验中的那些有什么区别？

sub\_401000使用一个参数，使用格式化字符串来调用sprintf函数，使得传入的参数创建用于HTTP通信的User-Agent字段。

#### 这个程序会运行多久？

1440分钟（24小时）。

#### 在这个恶意代码中有什么新的基于网络的迹象吗？

运行了一个新的User-Agent。

它的形式是 “Internet Explorer 7.50/pma%d” ，其中%d是程序已经运行了的分钟数。

#### 这个恶意代码的目的是什么？

首先检测网络连接，如果没有网络连接则停止运行；

如果网络已连接则下载HTML网页，该网页包含了注释，解析该网页；然后根据注释正文的第一个字符做出swich语句的跳转操作：

参数为‘a’时，创建C:\Temp的目录；

参数为‘b’时，进行了程序的自我复制；

参数为‘c’时，删除了C:\Temp\cc.exe；

参数为‘d’时，将程序设为开机自启动；

参数为‘e’时，程序休眠100秒。

并且将检测到网络连接后的操作循环运行1440次。

### yara规则编写

import "pe"

rule Message {

strings:

$ErrorM = "Error"

$SuccessM = "Success"

$Internet = "Internet"

condition:

$ErrorM or $SuccessM or $Internet

}

rule MalURLRequest {

strings:

$Mal = "practicalmalwareanalysis"

$Http = "http"

condition:

$Mal and $Http

}

rule EXE {

strings:

$exe = /[a-zA-Z0-9\_]+.exe/

condition:

$exe

}

rule Regedit {

strings:

$run = "Software\\Microsoft\\Windows\\CurrentVersion\\Run"

condition:

$run

}

1. **实验结论及心得体会**

进一步熟悉IDA。

复习到课上知识点，如注册表等等。