CNIT 58100 CFM: CYBERFORENSICS OF MALWARE – LAB 7

Ibrahim Waziri Jr

PhD in Information Security (CERIAS)

Lab 7

Due on: October 1st, 2014 (Week 5)

Instructor: **Associate Prof Sam Liles**

Purdue University

2014

Abstract

This lab covers the skill discussed in Chapter 7 of the text. The practice covered in this lab is all based on malware analysis and the Interactive Disassembler Professional (IDA Pro) software. The malware files used are provided as an extension of the text for practical purposes.

The lab consists of multiple questions that require short answers. Throughout this lab we used a special tool known as IDA Pro for the malware analysis.

This paper provides answers to Chapter 7 lab. The lab uses the file *Lab07-01exe, Lab07-02.exe and Lab07-03.exe.*These file are malwares and therefore could be harmful if used for non-training purposes.

The goal of this lab is to give a hands-on experience with analyzing Malicious Windows Programs (PMA).

Questions

Lab 7-1

Analyze the malware found in the file Lab07-01.exe

1. How does this program ensure that it continues running (achieves persistence) when the computer restarted?
2. Why does this program use a mutex?
3. What is a good host-based signature to use for detecting this program?
4. What is a good network-based signature for detecting this malware?
5. What is the purpose of this program?
6. When will this program finish executing?

Lab 7-2

Analyze the malware found in the file Lab07-02.exe

1. How does the program achieve persistence?
2. What is the purpose of this program?
3. When will this program finish executing?

Lab 7-3

1. How does this program achieve persistence to ensure that it continues running when the computer is restarted?
2. What are two good host-based signatures for this malware?
3. What is the purpose of this program?
4. How could you remove this malware once it is installed?

Answers

Lab 7-1:

1. The program creates a service MalService that ensures the program runs every time the computer starts.
   To do this we start by analyzing the Lab07-01.exe file using IDAPro. Viewing the imports by clicking the imports tab reveals several imports as shown in the figure below:

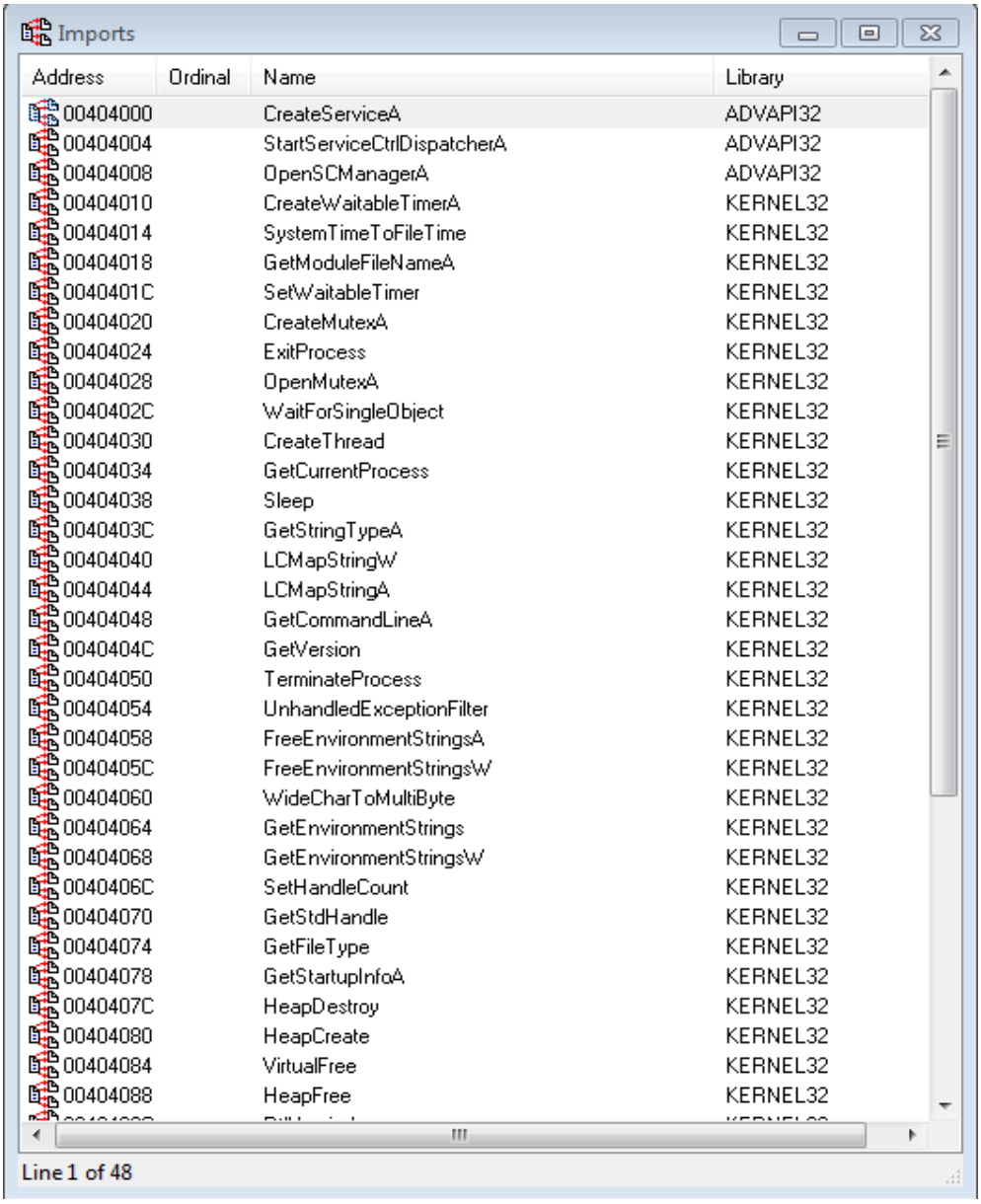

Fig 7-1a

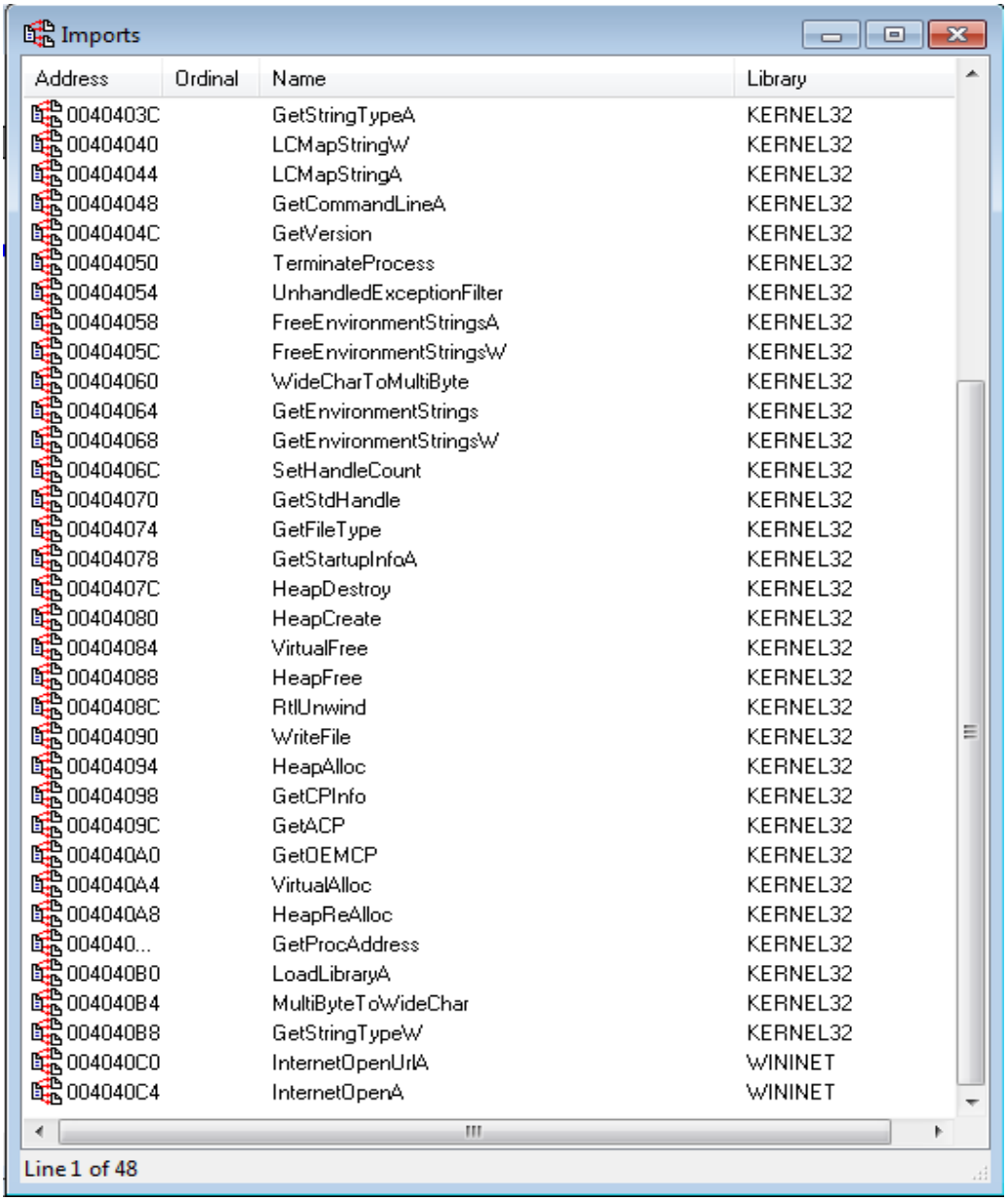

Fig 7-1b

The import OpenSCManager and CreateService indicate that this malware probably creates a service to ensure that it will run when the computer is restarted.
Another import StartServiceCtrlDispatcherA indicates that the file is actually a service. The import InternetOpen and InternetOpenUrlindicates that this program might connect to the internet to download content.

Clicking on the function tab on IDAPro reveals the functions as shown in Fig 7-1c.

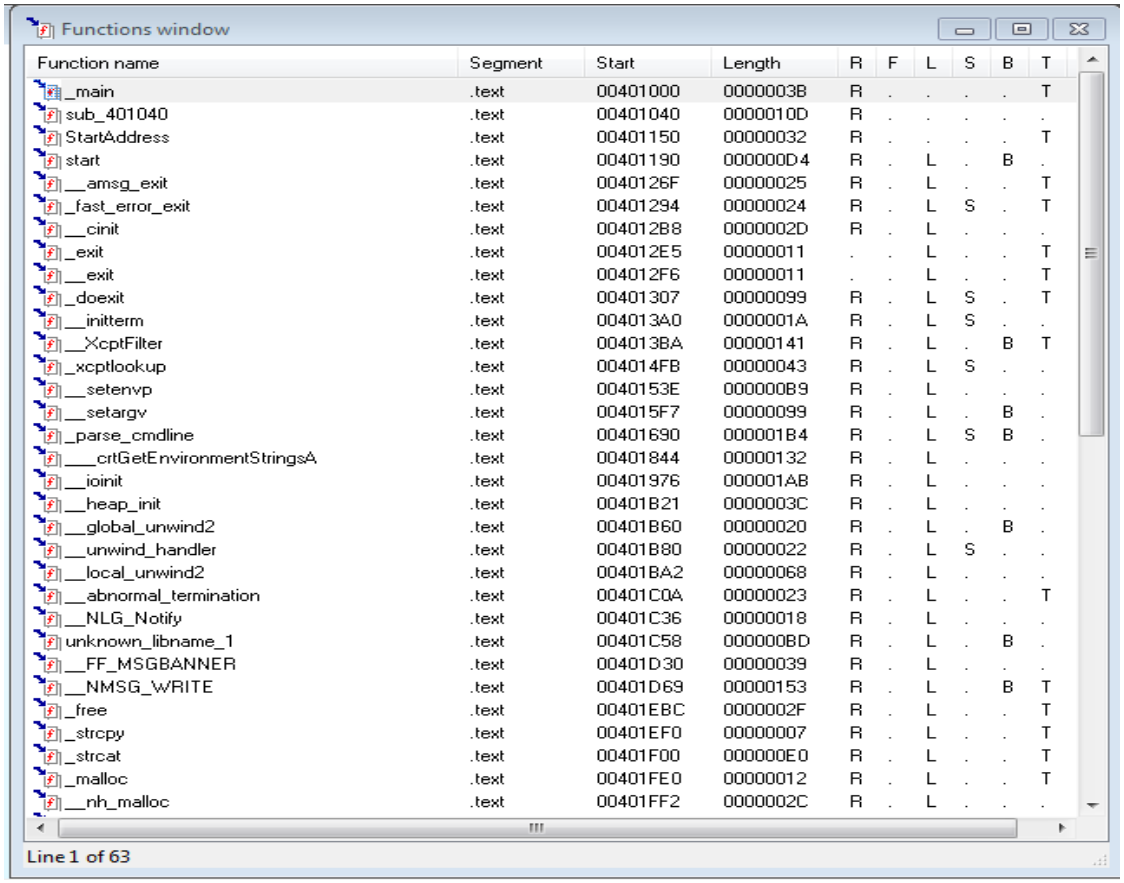

Fig 7-1c

Now viewing the _main at location 0x401000. The _main function reveals the service "MalService" as shown in Fig 7-1d below:

```
sub     esp, 10h
lea     eax, [esp+10h+ServiceStartTable]
mov     [esp+10h+ServiceStartTable.lpServiceName], offset aMalservice ; "MalService"
push    eax             ; lpServiceStartTable
mov     [esp+14h+ServiceStartTable.lpServiceProc], offset sub_401040
mov     [esp+14h+var_8], 0
mov     [esp+14h+var_4], 0
call    ds:StartServiceCtrlDispatcherA
push    0
push    0
call    sub_401040
add     esp, 18h
retn
_main endp
```

Fig 7-1d

2. The program uses a mutex to ensure that only one copy of the program is running at a time.
To verify that, we examine the sub_401040 function shown in Fig 7-1d above. The result of this function is as shown in Fig 7-2a below:

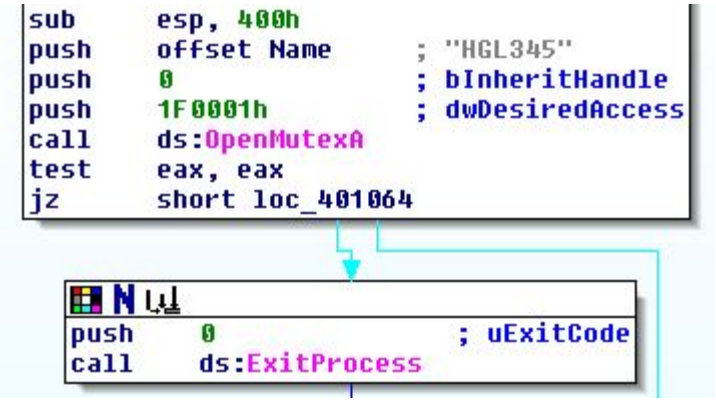

Fig 7-2a

Analyzing Fig 7-2a above, we can see that it opens a MutexA, and taking a look at Fig 7-2b below, we can see how the function creates the Mutex with the call CreateMutexA.

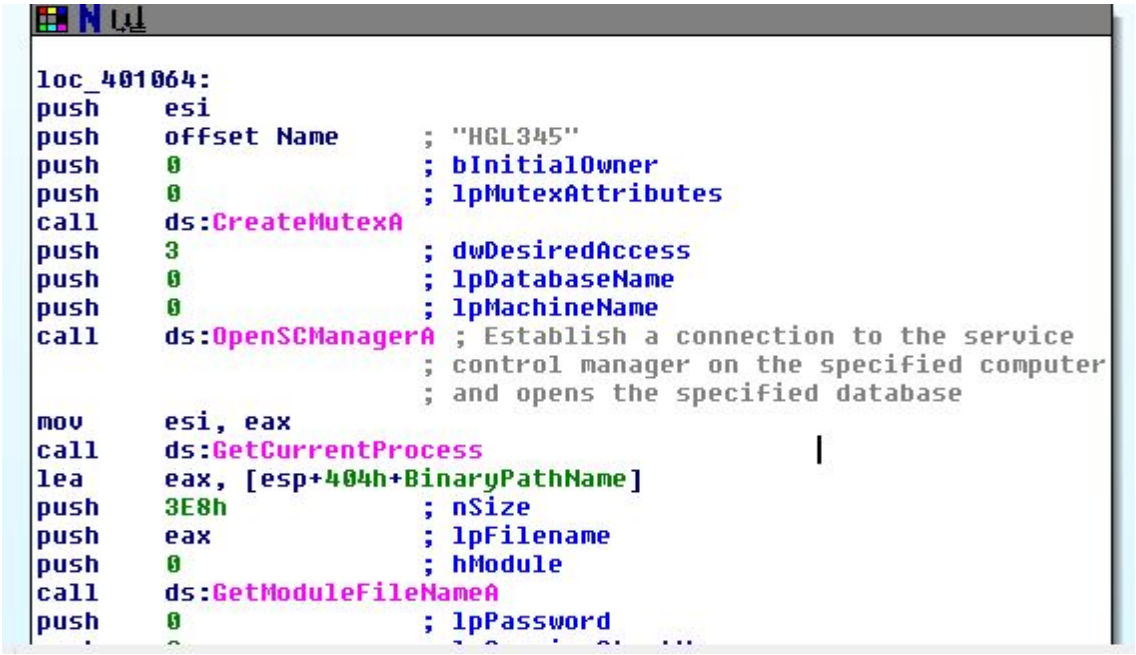

Fig 7-2b

3. A good host-based signature to use for detecting this program can be the mutex HGL345 shown in Fig 7-2b above. Another is the service MalService. The combination of these two mutex calls is designed to ensure that only one copy of this executable is running on a system at any given time. If a copy was already running, then the first call to OpenMutexA would have been successful, and the program have exited.

4. The malware uses the user-agent Internet Explorer 8.0 and communicates with www.malwareanalysisbook.com.

   From the figure shown below Fig 7-4, we can see that the malware communicates with www.malwareanalysisbook.com

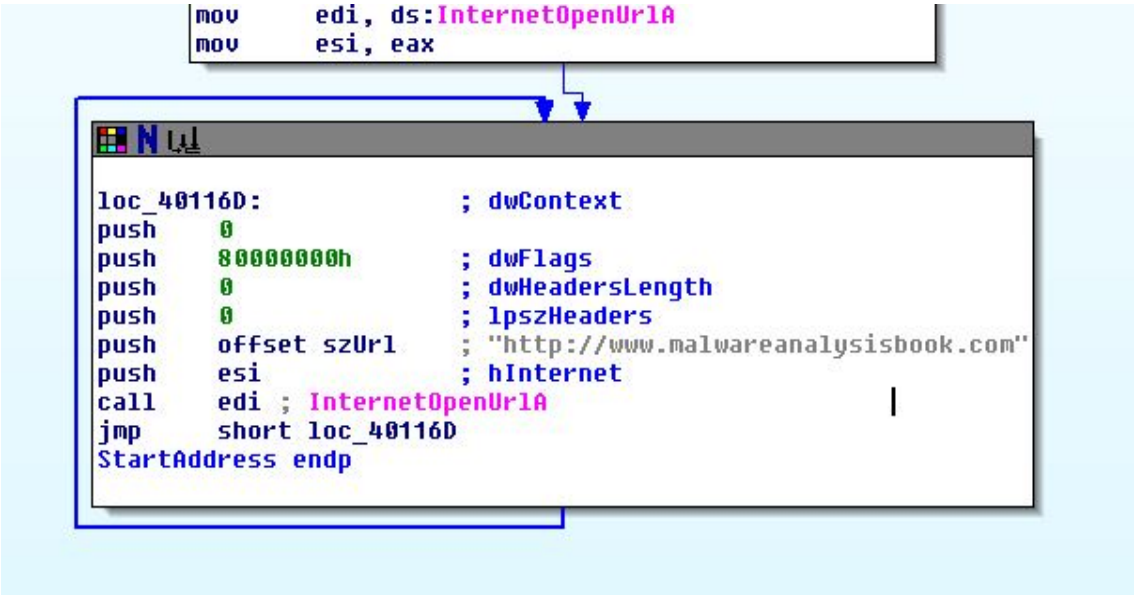

Fig 7-4

5. The program waits until midnight on January 1, 2100 and then sends many requests to http://www.malwareanalysisbook.com/, presumably to conduct a distributed denial-of-service (DDoS) attack against the site.
   To figure this out, ones needs to understand how Windows Time are structured. According to MSDN, the SYSTEMTIME structure has separate fields for the second, minute, hour, day and so on for use in specifying time. Taking a good and deep look at Fig 7-5 below, all the values are set to 0, and then the value for the year is set to 0x0834. Converting 0x0834 to decimal tells us that the year is 2100.

```
lea     eax, [esp+404h+DueTime]
mov     dword ptr [esp+404h+SystemTime.wYear], edx
lea     ecx, [esp+404h+SystemTime]
mov     dword ptr [esp+404h+SystemTime.wDayOfWeek], edx
push    eax             ; lpFileTime
mov     dword ptr [esp+408h+SystemTime.wHour], edx
push    ecx             ; lpSystemTime
mov     dword ptr [esp+40Ch+SystemTime.wSecond], edx
mov     [esp+40Ch+SystemTime.wYear], 834h
call    ds:SystemTimeToFileTime
```

Fig 7-5

6. The program will never finish.
   From Fig 7-4 above, we can see an instruction jmp short loc_40116D. Googling this tells me that it is an unconditional jump, which means that the code will never end.

Lab 7-2:

1. I couldn't find any evidence that this program achieves persistence. Because whenever I double click it, it only runs once and then exits.
2. The purpose of the program is to display a webpage, which is shown in Figure 7-6 below.

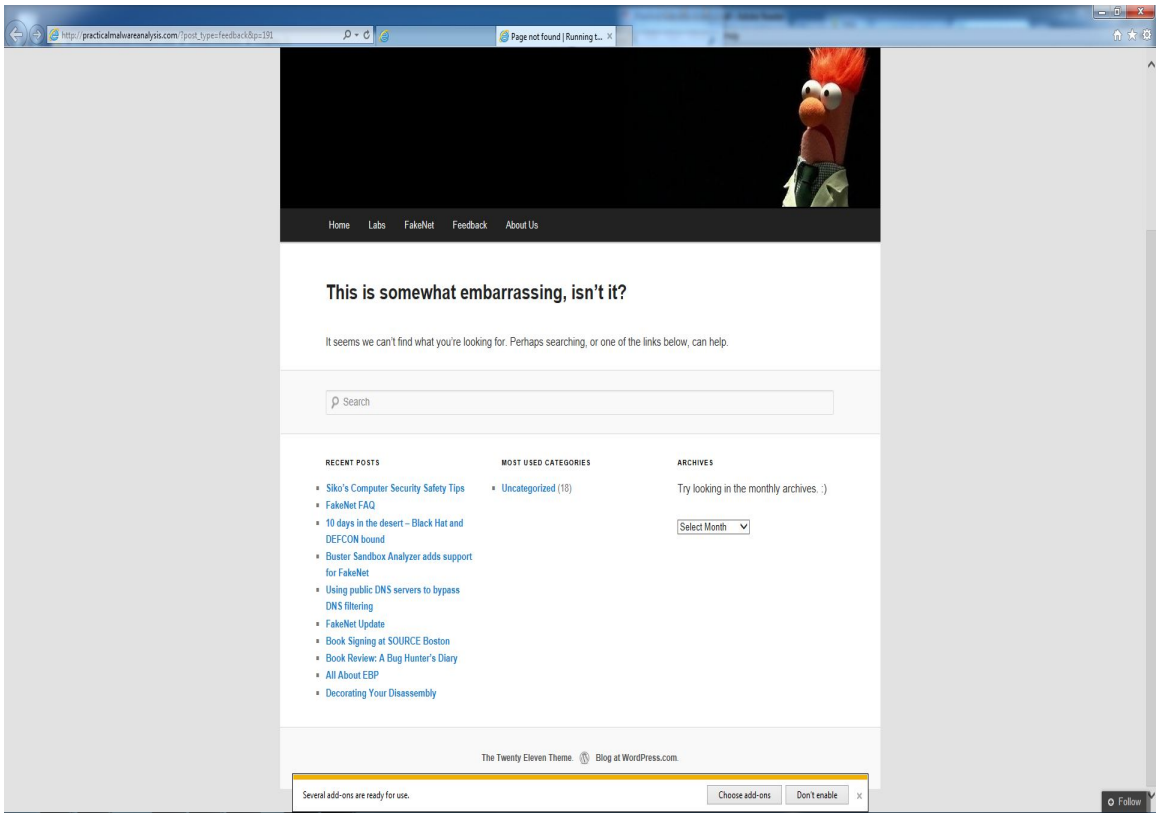

Fig 7-6

3. Without any analysis, the program automatically finishes executing when it displays the webpage. Pretty much nothing shows up again.

Lab 7-3

Whenever I run this program, my whole virtual machine fails, the first time it froze, the second it crashed. I don't think safe to run this program on my computer. So I couldn't answer it.

Problem Encountered:

While attempting to answer Lab7-3, after running the file Lab07-03.exe, I encountered some serious problems which are explained in Lab7-3 answers above.

Conclusion:

This lab covered the Windows concepts that are important to malware analysis. These concepts are elements such as processes, threads and network functionality.