Song Yang (sy540)
Xin Yang (xy213)
Zhuohang Li (zl299)

Report of Homework 9

Lab11-01:

1. What does the malware drop to disk?

We use process monitor and we saw that msgina32.dll and software.LOG are dropped on the machine.

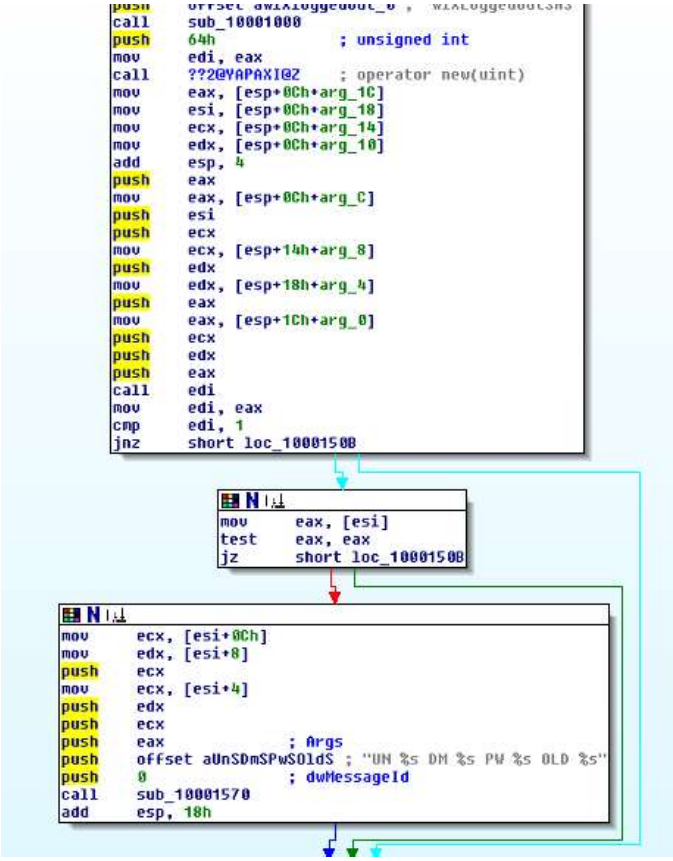
5:46:5...	Lab11-01.exe	212	RegOpenKey	HKLM\Software\Microsoft\Windows NT\CurrentVersion\Diagnostics	NAME NOT FOUND	Desired Access: R...
5:46:5...	Lab11-01.exe	212	RegOpenKey	HKLM\Software\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\ntdll.dll	NAME NOT FOUND	Desired Access: R...
5:46:5...	Lab11-01.exe	212	RegOpenKey	HKLM\Software\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\kernel32.dll	NAME NOT FOUND	Desired Access: R...
5:46:5...	Lab11-01.exe	212	ReadFile	C:\Documents and Settings\lynn\Desktop\HW9\Chapter_11L\Lab11-01.exe	SUCCESS	Offset: 4,096, Leng...
5:46:5...	Lab11-01.exe	212	ReadFile	C:\Documents and Settings\lynn\Desktop\HW9\Chapter_11L\Lab11-01.exe	SUCCESS	Offset: 32,768, Len...
5:46:5...	Lab11-01.exe	212	ReadFile	C:\WINDOWS\system32\sortkey.nls	SUCCESS	Offset: 32,768, Len...
5:46:5...	Lab11-01.exe	212	CreateFile	C:\Documents and Settings\lynn\Desktop\HW9\Chapter_11L\msgina32.dll	SUCCESS	Desired Access: G...
5:46:5...	Lab11-01.exe	212	CreateFile	C:\Documents and Settings\lynn\Desktop\HW9\Chapter_11L	SUCCESS	Desired Access: S...
5:46:5...	Lab11-01.exe	212	CloseFile	C:\Documents and Settings\lynn\Desktop\HW9\Chapter_11L	SUCCESS	
5:46:5...	Lab11-01.exe	212	WriteFile	C:\Documents and Settings\lynn\Desktop\HW9\Chapter_11L\msgina32.dll	SUCCESS	Offset: 0, Length: 4...
5:46:5...	Lab11-01.exe	212	WriteFile	C:\Documents and Settings\lynn\Desktop\HW9\Chapter_11L\msgina32.dll	SUCCESS	Offset: 4,096, Leng...
5:46:5...	Lab11-01.exe	212	CloseFile	C:\Documents and Settings\lynn\Desktop\HW9\Chapter_11L\msgina32.dll	SUCCESS	
5:46:5...	Lab11-01.exe	212	RegCreateKey	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon	SUCCESS	Desired Access: All...
5:46:5...	Lab11-01.exe	212	RegSetValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\GinaDLL	SUCCESS	Type: REG_SZ, Le...
5:46:5...	Lab11-01.exe	212	SetEndOfFileInf...	C:\WINDOWS\system32\config\software.LOG	SUCCESS	EndOfFile: 12,288
5:46:5...	Lab11-01.exe	212	SetEndOfFileInf...	C:\WINDOWS\system32\config\software.LOG	SUCCESS	EndOfFile: 12,288
5:46:5...	Lab11-01.exe	212	SetEndOfFileInf...	C:\WINDOWS\system32\config\software.LOG	SUCCESS	EndOfFile: 20,480
5:46:5...	Lab11-01.exe	212	SetEndOfFileInf...	C:\WINDOWS\system32\config\software.LOG	SUCCESS	EndOfFile: 24,576
5:46:5...	Lab11-01.exe	212	SetEndOfFileInf...	C:\WINDOWS\system32\config\software.LOG	SUCCESS	EndOfFile: 28,672

2. How does the malware achieve persistence?

We can also see that there is a process located in the register in HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\GinaDLL was added. Winlogon, the GINA, and network providers are the parts of the interactive logon model according to MSDN. In order to achieve the persistence, the malware use register to call its function each time that the user login.

3. How does the malware steal user credentials?

According to the dll file that is linked to the program whose name is msgina32.dll, we check the exports and imports, it shows that the dll might get a hook of the windows logon process. The function named WlxLoggedOutSAS is pretty dangerous, it copies the file and writes them to a local file that is stored in somewhere known by malware.



4. What does the malware do with stolen credentials?

There is a function in the dll file that is shown in the figure below.

```
push    edx             ; Dest
call    _vsnwprintf
push    offset Mode     ; Mode
push    offset Filename ; "msutil32.sys"
call    _wfopen
mov     esi, eax
add     esp, 18h
test    esi, esi
jz      loc_1000164F
```

```
lea     eax, [esp+858h+Dest]
push    edi
lea     ecx, [esp+85Ch+Buffer]
push    eax
push    ecx             ; Buffer
call    _wstrtime
add     esp, 4
lea     edx, [esp+860h+var_828]
push    eax
push    edx             ; Buffer
call    _wstrdate
add     esp, 4
push    eax
push    offset Format   ; "%s %s - %s "
push    esi             ; File
call    fwprintf
mov     edi, [esp+870h+dwMessageId]
add     esp, 14h
test    edi, edi
jz      short loc_10001637
```

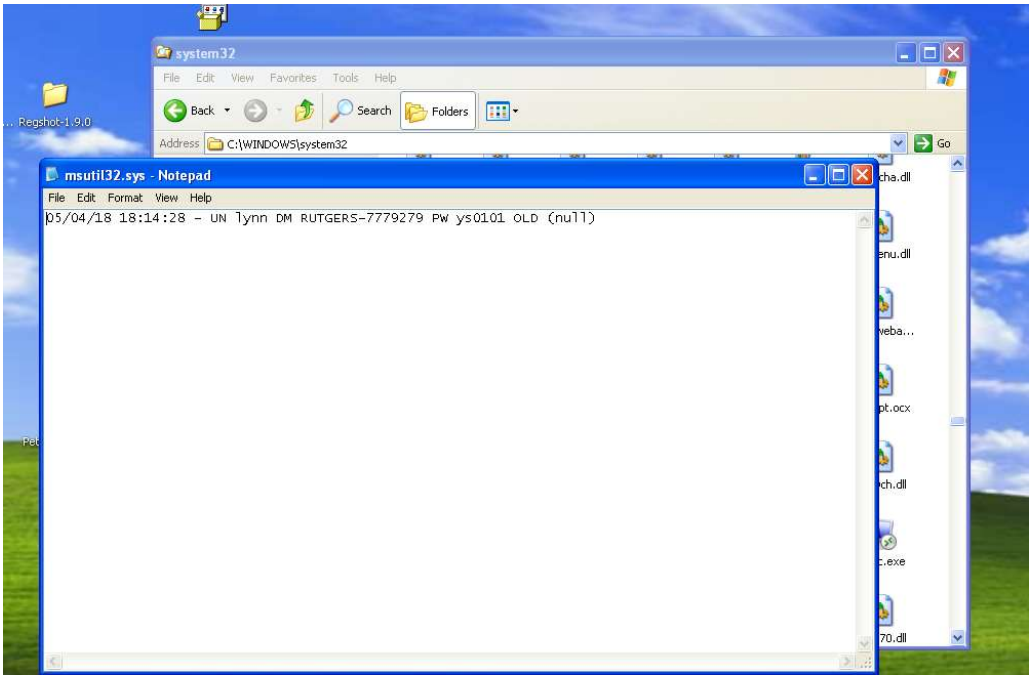
```
push    0               ; Arguments
lea     eax, [esp+860h+hMem]
push    0               ; nSize
push    eax             ; lpBuffer
push    409h            ; dwLanguageId
push    edi             ; dwMessageId
push    0               ; lpSource
push    1100h           ; dwFlags
call    ds:FormatMessageW
mov     ecx, [esp+85Ch+hMem]
push    ecx
push    edi
```

```
loc_10001637:           ; "\n"
push    offset asc_1000329C
push    esi             ; File
call    fwprintf
add     esp, 8
push    esi             ; File
call    fclose
add     esp, 4
pop     edi
```

It is trying to write to file msutil32.sys, it is a file that is located in C:/Windows/system32/ belongs to system files. The malware uses the credentials that are stolen in the last question to write file here.

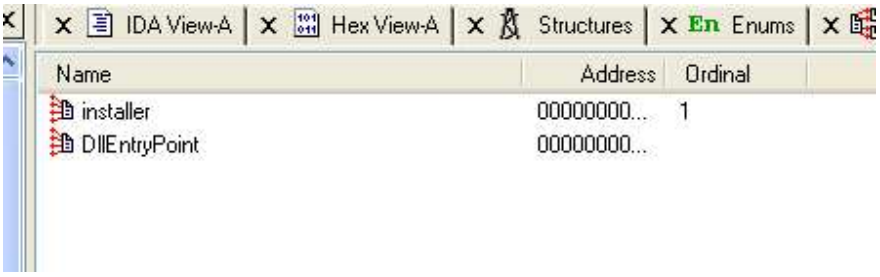
5. How can you use this malware to get user credentials from your test environment?

The malware is actually attached to logon, so after we login again after running the malware, we can get in the msutil32.sys, the password is right here!!!

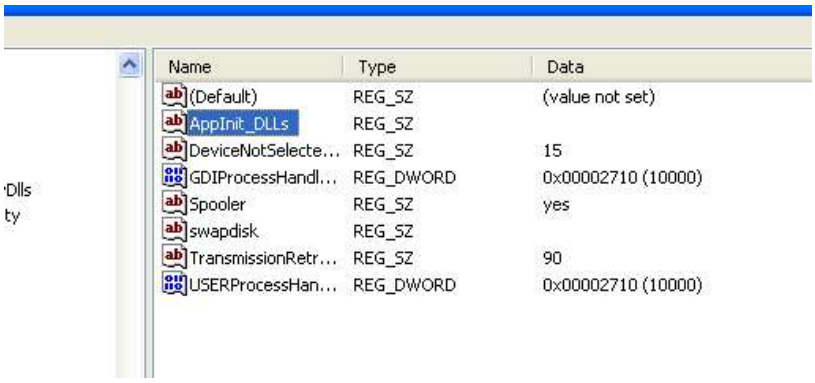


Lab11-02:

1. What are the exports for this DLL malware?



2. What happens after you attempt to install this malware using rundll32.exe?



A new key is added here in HKLM/SOFTWARE/Microsoft/Windows NT/CurrentVersion/Windows.

```
loc_10001629:           ; nSize
push    104h
push    offset ExistingFileName ; lpFilename
mov     ecx, [ebp+hModule]
push    ecx             ; hModule
call    ds:GetModuleFileNameA
push    101h            ; Size
push    0               ; Val
push    offset byte_100034A0 ; Dst
call    memset
add     esp, 0Ch
call    sub_1000105B
mov     [ebp+lpFileName], eax
push    104h            ; Count
push    offset aLab1102_ini ; "\\Lab11-02.ini"
mov     edx, [ebp+lpFileName]
push    edx             ; Dest
call    strncat
add     esp, 0Ch
push    0               ; hTemplateFile
push    80h             ; dwFlagsAndAttributes
push    3               ; dwCreationDisposition
push    0               ; lpSecurityAttributes
push    1               ; dwShareMode
push    80000000h       ; dwDesiredAccess
mov     eax, [ebp+lpFileName]
push    eax             ; lpFileName
call    ds:CreateFileA
mov     [ebp+hObject], eax
cmp     [ebp+hObject], 0FFFFFFFFh
jz      short loc_100016DE
```

It tried to open file Lab11-01.ini.

3. Where must Lab11-02.ini reside in order for the malware to install properly?

```
push    101h            ; Size
push    0               ; Val
push    offset byte_100034A0 ; Dst
call    memset
add     esp, 0Ch
call    sub_1000105B
mov     [ebp+lpFileName], eax
push    104h            ; Count
push    offset aLab1102_ini ; "\\Lab11-02.ini"
mov     edx, [ebp+lpFileName]
push    edx             ; Dest
call    strncat
add     esp, 0Ch
push    0               ; hTemplateFile
push    80h             ; dwFlagsAndAttributes
push    3               ; dwCreationDisposition
push    0               ; lpSecurityAttributes
```

```
; Attributes: bp-based frame

sub_1000105B proc near
push    ebp
mov     ebp, esp
push    104h            ; uSize
push    offset Buffer   ; lpBuffer
call    ds:GetSystemDirectoryA
mov     eax, offset Buffer
pop     ebp
retn
sub_1000105B endp
```

The malware tried to get the file Lab11-02.ini in the path that is returned by sub_1000105B, which is the system directory. So in order to install properly, the Lab11-02.ini must be in System32 folder.

4. How is this malware installed for persistence?

The malware gets installed by adding a key to register to host. According to MSDN, the appInit_DLL is for allowing an arbitrary list of DLLs to be loaded into each user-mode process on the system.

AppInit_DLLs in Windows 7 and Windows Server 2008 R2

Platform

Clients - Windows 7
Servers - Windows Server 2008 R2

Feature Impact

Severity - Low
Frequency - Low

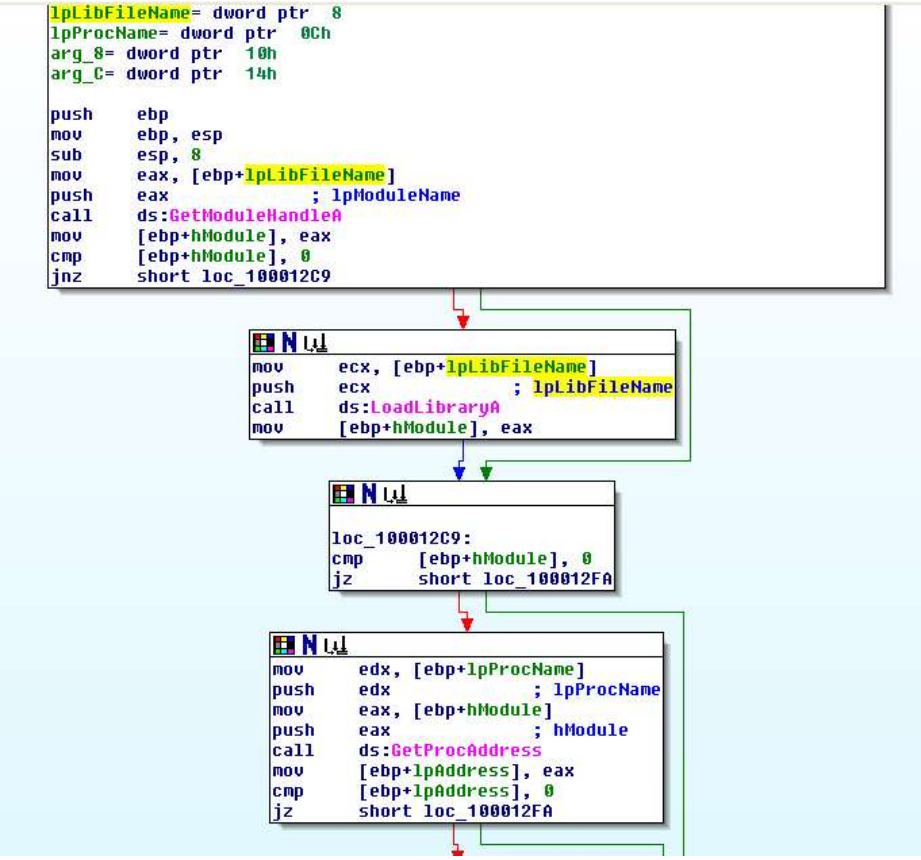
Description

AppInit_DLLs is a mechanism that allows an arbitrary list of DLLs to be loaded into each user mode process on the system. Microsoft is modifying the AppInit DLLs facility in Windows 7 and Windows Server 2008 R2 to add a new code-signing requirement. This will help improve the system reliability and performance, as well as improve visibility into the origin of software.

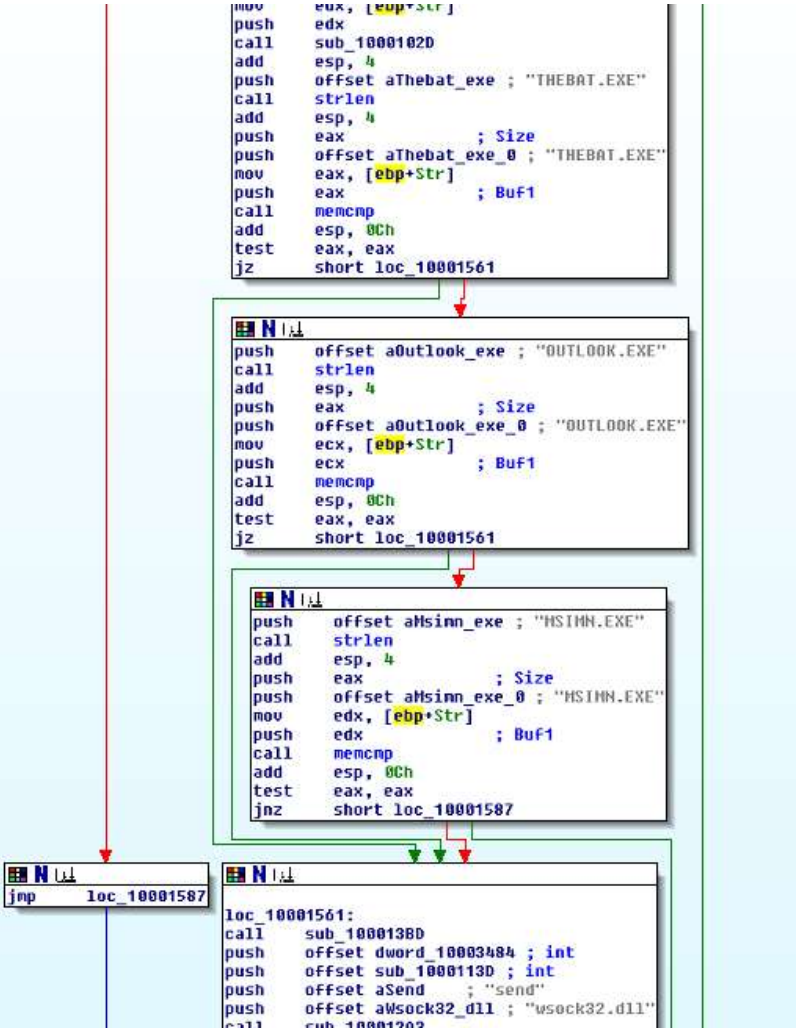
Configuration

Values stored under the HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion \Windows key in the registry determine the behavior of the AppInit_DLLs infrastructure. The table below describes these registry values:

5. What user-space rootkit technique does this malware employ?



The malware tried to get the address from a file and send them to sub-functions. And it gets offset from the file to protect the malware. It hooked to some application to get the new address of the function. The malware is hooked with 3 applications including Thebat.exe, outlook.exe, and msimn.exe. And the offset and the address should from wsock32.dll.



6. What does the hooking code do?

It may jump to a function which starts with checking if the send buffer has the string “RCPT TO”. If it does, a new buffer “RCPT TO:<billy@malwareanalysisbook.com>\r\n” would be created and send it with a send function.

```
mov     ebp, esp
sub     esp, 204h
push    offset SubStr    ; "RCPT TO:"
mov     eax, [ebp+Str]
push    eax              ; Str
call    strstr
add     esp, 8
test    eax, eax
jz      loc_100011E4
```

```
push    offset Str       ; "RCPT TO: <"
call    strlen
add     esp, 4
push    eax              ; Size
push    offset aRcptTo_1 ; "RCPT TO: <"
lea     ecx, [ebp+Dst]
push    ecx              ; Dst
call    memcpy
add     esp, 0Ch
push    101h             ; Size
push    offset byte_100034A0 ; Src
push    offset aRcptTo_2 ; "RCPT TO: <"
call    strlen
add     esp, 4
lea     edx, [ebp+eax+Dst]
push    edx              ; Dst
call    memcpy
add     esp, 0Ch
push    offset Source    ; ">\r\n"
lea     eax, [ebp+Dst]
push    eax              ; Dest
call    strcat
add     esp, 8
mov     ecx, [ebp+arg_C]
push    ecx
lea     edx, [ebp+Dst]
push    edx              ; Str
```

7. Which process(es) does this malware attack and why?

The malware may attack the user by hooking in their email client, to get the private information.

8. What is the significance of the .ini file?

The ini file is the configuration that it would be needed when installing the malware.

9. How can you dynamically capture this malware's activity with Wireshark?

```
220 mail.inetsim.org INetSim Mail Service ready.
HELO userfcc21c8345
250 mail.inetsim.org
MAIL FROM: <root@jmprsp.com>
250 2.1.0 ok
RCPT TO: <billy@malwareanalysisbook.com>
250 2.1.5 ok
RCPT TO: <user@jmprsp.com>
250 2.1.5 ok
RCPT TO: <billy@malwareanalysisbook.com>
RCPT TO: <admin@jmprsp.com>
250 2.1.5 ok
DATA
250 2.1.5 ok
Message-ID: <91EC5A5E67E942978E175CFD7F09A826@userfcc21c8345>
From: "jmprsp" <root@jmprsp.com>
To: <user@jmprsp.com>,
.<admin@jmprsp.com>
Subject: what
Date: Sat, 12 Mar 2016 16:25:38 +0800
MIME-Version: 1.0
Content-Type: multipart/alternative;
.boundary="----=_NextPart_000_0003_01D17C7B.D06F9410"
X-Priority: 3
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook Express 6.00.2900.5512
X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2900.5512

This is a multi-part message in MIME format.
```

Lab11-03:

1. What interesting analysis leads can you discover using basic static analysis?

```
; Attributes: bp-based frame

; int __cdecl main(int argc, const char **argv, const char **envp)
_main proc near

FileName= byte ptr -104h
argc= dword ptr  8
argv= dword ptr  0Ch
envp= dword ptr  10h

push    ebp
mov     ebp, esp
sub     esp, 104h
push    0               ; bFailIfExists
push    offset NewFileName ; "C:\\WINDOWS\\System32\\inet_epar32.dll"
push    offset ExistingFileName ; "Lab11-03.dll"
call    ds:CopyFileA
push    offset aCisvc_exe ; "cisvc.exe"
push    offset Format   ; "C:\\WINDOWS\\System32\\%s"
lea     eax, [ebp+FileName]
push    eax             ; Dest
call    _sprintf
add     esp, 0Ch
lea     ecx, [ebp+FileName]
push    ecx             ; lpFileName
call    sub_401070
add     esp, 4
push    offset aNetStartCisvc ; "net start cisvc"
call    sub_4013BC
add     esp, 4
xor     eax, eax
mov     esp, ebp
pop     ebp
retn
_main endp
```

The malware gets installed in the main function. It first copies the Lab11-03.dll to C:/Windows/System32. It then modifies C:/Windows/System32/cisvc.exe and executes the file starting a service with a command “net start cisvc”

```
; Exported entry   1. zzz69806582


; Attributes: bp-based frame

public zzz69806582
zzz69806582 proc near

var_4= dword ptr -4

push    ebp
mov     ebp, esp
push    ecx
push    0                       ; lpThreadId
push    0                       ; dwCreationFlags
push    0                       ; lpParameter
push    offset StartAddress ; lpStartAddress
push    0                       ; dwStackSize
push    0                       ; lpThreadAttributes
call    ds:CreateThread
mov     [ebp+var_4], eax
cmp     [ebp+var_4], 0
jz      short loc_10001566

xor     eax, eax
jmp     short loc_1000156B

loc_10001566:
mov     eax, 1

loc_1000156B:
mov     esp, ebp
pop     ebp
retn
zzz69806582 endp
```

Here is an interesting export in dll file, it creates a thread to capture the key pressing event. The function is called after creating kernel64x.dll in system32 folder to prevent the thread getting banned by the system.

```
push    0                       ; Code
call    _exit

loc_1000149D:                   ; "MZ"
push    offset Name
push    1                       ; bInitialOwner
push    0                       ; lpMutexAttributes
call    ds:CreateMutexA
mov     [ebp+var_818], eax
cmp     [ebp+var_818], 0
jnz     short loc_100014BD

loc_100014BD:                   ; hTemplateFile
push    0
push    80h                     ; dwFlagsAndAttributes
push    4                       ; dwCreationDisposition
push    0                       ; lpSecurityAttributes
push    1                       ; dwShareMode
push    0C0000000h              ; dwDesiredAccess
push    offset FileName         ; "C:\\WINDOWS\\System32\\kernel64x.dll"
call    ds:CreateFileA
mov     [ebp+hObject], eax
cmp     [ebp+hObject], 0
jnz     short loc_100014EB

jmp     short loc_10001530
jmp     short loc_10001530

loc_100014EB:                   ; dwMoveMethod
push    2
push    0                       ; lpDistanceToMoveHigh
push    0                       ; lDistanceToMove
mov     eax, [ebp+hObject]
push    eax                     ; hFile
```

2. What happens when you run this malware?

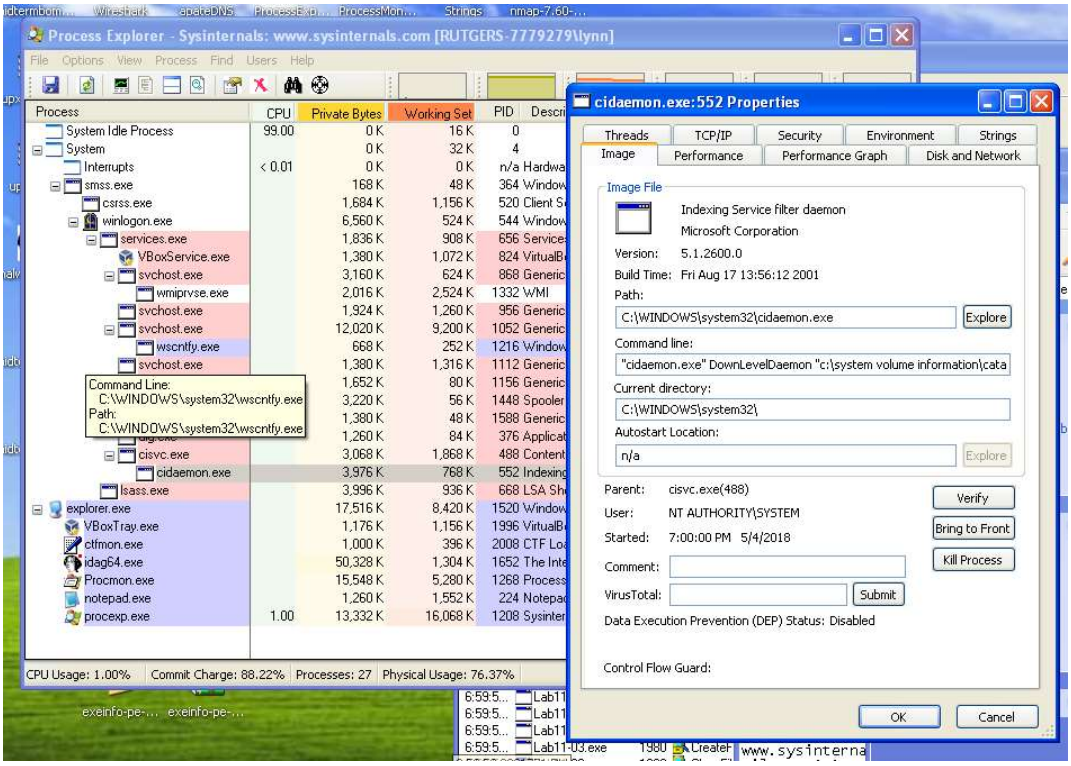
The cmd window said “the index service is running” and the malware service begin to log key pressing and save them to C:\\Windows\\System32\\kernel64x.dll.

3. How does Lab11-03.exe persistently install Lab11-03.dll ?

The malware infects cisvc.exe. The malware would installed each time the cisvc.exe runs.

4. Which Windows system file does the malware infect?

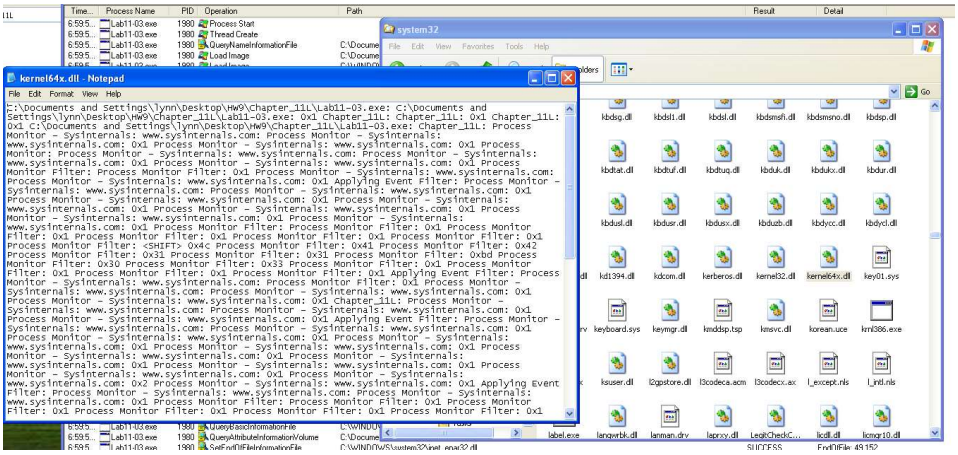
C:/Windows/System32/cisvc.exe got infected.



5. What does Lab11-03.dll do?

The malware use two functions including GetAsyncKeyState and GetForegroundWindow. The key pressing data are stored into C:/Windows/System32/kernel64x.dll.

6. Where does the malware store the data it collects?



It stored the data in kernel64x.dll in the system32 folder.