



Master of Information Security

MIS4203 – Independent Studies in Information Security

Index Number – 16770217 | Reg. Number – 2016MIS021

Study Number – 07 - Digital Forensic – II

January 19, 2019

University of Colombo School of Computing

Table of Contents

Study Number – 07 - Digital Forensic – II	1
Problem.....	3
Approach	14
Conclusion	14

Problem

Perform a forensics analysis of the memory image “is7.elf” and gather the following forensic artifacts;

Memory image “is7.elf”: Sha1 hash value: “f66dc5e898d3edf1013c86d706813071147f6ae6”
Forensics Analysis

Hash Verification:

```
[AngryBird:is7 Dilanka$ shasum 1 is7.elf
shasum: 1:
f66dc5e898d3edf1013c86d706813071147f6ae6  is7.elf
AngryBird:is7 Dilanka$ ]
```

Used tools:

1. Volatility
2. <https://packetstormsecurity.com>
3. <https://www.virustotal.com>
4. AVG Antivirus Software

- System Information
 - Operating System name and version

Command : Imageinfo

For a high level summary of the memory sample you're analyzing, use the imageinfo command. Most often this command is used to identify the operating system, service pack, and hardware architecture (32 or 64 bit)

```
[AngryBird:is7 Dilanka$ vol.py -f is7.elf imageinfo
Volatility Foundation Volatility Framework 2.5
INFO    : volatility.debug      : Determining profile based on KDBG search...
INFO    : volatility.debug      : Suggested Profile(s) : Win7SP0x86, Win7SP1x86
          AS Layer1 : IA32PagedMemoryPae (Kernel AS)
          AS Layer2 : OSXMemELF (Unnamed AS)
          AS Layer3 : FileAddressSpace (/Documents/MIS/shared_between_vms/miniprojects/19-Jan-2019/is7/is7.elf)
          PAE type : PAE
          DTB : 0x185000L
          KDBG : 0x82972c28L
          Number of Processors : 1
          Image Type (Service Pack) : 0
          KPCR for CPU 0 : 0x82973c00L
          KUSER_SHARED_DATA : 0xfffff0000L
          Image date and time : 2019-01-18 12:02:09 UTC+0000
          Image local date and time : 2019-01-18 17:32:09 +0530
[AngryBird:is7 Dilanka$ ]
```

Using suggested profiles:

```
[AngryBird:is7 Dilanka$ vol.py -f is7.elf --profile=Win7SP0x86 kdbgscan
Volatility Foundation Volatility Framework 2.5
*****
Instantiating KDBG using: Kernel AS Win7SP0x86 (6.1.7600 32bit)
Offset (V)           : 0x82972c28
Offset (P)           : 0x2972c28
KDBG owner tag check : True
Profile suggestion (KDBGHeader): Win7SP1x86
Version64           : 0x82972c00 (Major: 15, Minor: 7600)
Service Pack (CmNtCSDVersion) : 0
Build string (NtBuildLab)   : 7600.17795.x86fre.win7_gdr.15031
PsActiveProcessHead    : 0x8298aea8 (55 processes)
PsLoadedModuleList     : 0x82992810 (143 modules)
KernelBase            : 0x8284a000 (Matches MZ: True)
Major (OptionalHeader) : 6
Minor (OptionalHeader) : 1
KPCR                 : 0x82973c00 (CPU 0)

*****
Instantiating KDBG using: Kernel AS Win7SP0x86 (6.1.7600 32bit)
Offset (V)           : 0x82972c28
Offset (P)           : 0x2972c28
KDBG owner tag check : True
Profile suggestion (KDBGHeader): Win7SP0x86
Version64           : 0x82972c00 (Major: 15, Minor: 7600)
Service Pack (CmNtCSDVersion) : 0
Build string (NtBuildLab)   : 7600.17795.x86fre.win7_gdr.15031
PsActiveProcessHead    : 0x8298aea8 (55 processes)
PsLoadedModuleList     : 0x82992810 (143 modules)
KernelBase            : 0x8284a000 (Matches MZ: True)
Major (OptionalHeader) : 6
Minor (OptionalHeader) : 1
KPCR                 : 0x82973c00 (CPU 0)
```

```
[AngryBird:is7 Dilanka$ vol.py -f is7.elf --profile=Win7SP1x86 kdbgscan
Volatility Foundation Volatility Framework 2.5
*****
Instantiating KDBG using: Kernel AS Win7SP1x86 (6.1.7601 32bit)
Offset (V)           : 0x82972c28
Offset (P)           : 0x2972c28
KDBG owner tag check : True
Profile suggestion (KDBGHeader): Win7SP1x86
Version64           : 0x82972c00 (Major: 15, Minor: 7600)
Service Pack (CmNtCSDVersion) : 0
Build string (NtBuildLab)   : 7600.17795.x86fre.win7_gdr.15031
PsActiveProcessHead    : 0x8298aea8 (55 processes)
PsLoadedModuleList     : 0x82992810 (143 modules)
KernelBase            : 0x8284a000 (Matches MZ: True)
Major (OptionalHeader) : 6
Minor (OptionalHeader) : 1
KPCR                 : 0x82973c00 (CPU 0)

*****
Instantiating KDBG using: Kernel AS Win7SP1x86 (6.1.7601 32bit)
Offset (V)           : 0x82972c28
Offset (P)           : 0x2972c28
KDBG owner tag check : True
Profile suggestion (KDBGHeader): Win7SP0x86
Version64           : 0x82972c00 (Major: 15, Minor: 7600)
Service Pack (CmNtCSDVersion) : 0
Build string (NtBuildLab)   : 7600.17795.x86fre.win7_gdr.15031
PsActiveProcessHead    : 0x8298aea8 (55 processes)
PsLoadedModuleList     : 0x82992810 (143 modules)
KernelBase            : 0x8284a000 (Matches MZ: True)
Major (OptionalHeader) : 6
Minor (OptionalHeader) : 1
KPCR                 : 0x82973c00 (CPU 0)
```

- User information

By scanning registry hives using hivelist command. We can list down all the hives with ntuser.dat file related paths, then identify user account.

User account is mis-win7

Using profile : Win7SP0x86

```
[AngryBird:is7 Dilanka$ vol.py -f is7.elf --profile=Win7SP0x86 hivelist
Volatility Foundation Volatility Framework 2.5
Virtual Physical Name
-----
0x89e16788 0x27b73788 \SystemRoot\System32\Config\SECURITY
0x89e95008 0x26e76008 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0x89ecc9d0 0x260c29d0 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0x8a6379d0 0x2bfb79d0 \Device\HarddiskVolume1\Boot\BCD
0x8a7f6008 0x2c148008 \SystemRoot\System32\Config\SOFTWARE
0x9c094008 0x7641a008 \??\C:\Users\mis-win7\ntuser.dat
0x9cd9a008 0x1c05f008 \??\C:\Users\mis-win7\AppData\Local\Microsoft\Windows\UsrClass.dat
0xa4b5b9d0 0x259729d0 \??\C:\System Volume Information\Syscache.hve
0x8960c800 0x2d643800 [no name]
0x8961a2c8 0x2d68f2c8 \REGISTRY\MACHINE\SYSTEM
0x896428d8 0x2d5b98d8 \REGISTRY\MACHINE\HARDWARE
0x896d19d0 0x033279d0 \SystemRoot\System32\Config\DEFAULT
0x89e043b0 0x271c53b0 \SystemRoot\System32\Config\SAM
```

Using profile: Win7SP1x86

```
[AngryBird:is7 Dilanka$ vol.py -f is7.elf --profile=Win7SP1x86 hivelist
Volatility Foundation Volatility Framework 2.5
Virtual Physical Name
-----
0x89e16788 0x27b73788 \SystemRoot\System32\Config\SECURITY
0x89e95008 0x26e76008 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0x89ecc9d0 0x260c29d0 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0x8a6379d0 0x2bfb79d0 \Device\HarddiskVolume1\Boot\BCD
0x8a7f6008 0x2c148008 \SystemRoot\System32\Config\SOFTWARE
0x9c094008 0x7641a008 \??\C:\Users\mis-win7\ntuser.dat
0x9cd9a008 0x1c05f008 \??\C:\Users\mis-win7\AppData\Local\Microsoft\Windows\UsrClass.dat
0xa4b5b9d0 0x259729d0 \??\C:\System Volume Information\Syscache.hve
0x8960c800 0x2d643800 [no name]
0x8961a2c8 0x2d68f2c8 \REGISTRY\MACHINE\SYSTEM
0x896428d8 0x2d5b98d8 \REGISTRY\MACHINE\HARDWARE
0x896d19d0 0x033279d0 \SystemRoot\System32\Config\DEFAULT
0x89e043b0 0x271c53b0 \SystemRoot\System32\Config\SAM
```

- Start and End time

Start time can be consider as system process start time.

Offset(V)	Name	PID	PPID	Thds	Hnds	Sess	Wow64	Start	Exit
0x8483c968	System	4	0	82	569	-----	0	2019-01-17 09:16:33 UTC+0000	
0x85822428	smss.exe	272	4	2	29	-----	0	2019-01-17 09:16:33 UTC+0000	
0x85eebd40	smss.exe	340	272	0	-----	0	0	2019-01-17 09:16:39 UTC+0000	2019-01-17 09:16:43 UTC+0000
0x85eebd40	smss.exe	340	249	0	415	0	0	2019-01-17 09:16:43 UTC+0000	

- What can you conclude about the identified sockets?

TCPv4 0.0.0.0:12346

TCPv4 0.0.0.0:12345

These ports are not the common ports that applications are using. So, can be considered they are being used by the malicious activity since it is open by the malicious process's subprocess.

So, I searched for the suspected crack file and the exe file in virus databases. Found this

<https://packetstormsecurity.com/files/23678/mmcra...zip.html>

marks Java Photography MongoDB Database JavaScript WebService Multibindings · go... mac XML

packet storm
exploit the possibilities

Home Files News About Contact

mmcra...zip
Authored by Eric D Posted Nov 29, 2000

MMCRAK is a Netbus 1.6 client installer. Attempts to avoid AV software. Archive password is set to p4ssw0rd. Use at your own risk.

[tags | trojan](#) MD5 | 71c95e5d0b61dff7a6c7967c56cf826 [Download](#) | [Favorite](#) | [Comments \(0\)](#)

[Related Files](#)

Share This

[Like 0](#) [Tweet](#) [LinkedIn](#) [Reddit](#) [Digg](#) [StumbleUpon](#)

Comments [RSS](#)

No comments yet, be the first!

Using Virus total:

<https://www.virustotal.com/#/file/99e4b3b444f9ab64825dfd547c71da410c01edd.../detection>

marks Java Photography MongoDB Database JavaScript WebService Multibindings · go... mac XML hacks Arduino Cambio

Search

45 engines detected this file

Detection	Details	Relations	Behavior	Community
AegisLab	⚠️ Trojan.DOS.Crackz.blc	Anty-AVL	⚠️ Trojan[Dropper]/DOS.Crackz	
ArcaBit	⚠️ Trojan.Dropper.RSA	Avast	⚠️ Win32:Binder-DL [Drp]	
AVG	⚠️ Win32:Binder-DL [Drp]	Avira	⚠️ TR/Crackz.DOS	
BitDefender	⚠️ Trojan.Dropper.RSA	CAT-QuickHeal	⚠️ TrojanDropper.Crackz	
ClamAV	⚠️ Win.Dropper.Crackz-1	Comodo	⚠️ TrojWare.Win32.TrojanDropper.Crackz...	
CyberReason	⚠️ malicious.6ce10c	Cylance	⚠️ Unsafe	
Cyren	⚠️ Trojan!762e	DrWeb	⚠️ Trojan.Crackz.34001	
Emsisoft	⚠️ Trojan.Dropper.RSA (B)	eScan	⚠️ Trojan.Dropper.RSA	
ESET-NOD32	⚠️ Crackz	F-Prot	⚠️ Trojan!762e	
F-Secure	⚠️ Trojan.Dropper.RSA	Fortinet	⚠️ W32/Crackz.A!tr	
GData	⚠️ TrojanDropper.MMCrackZ.A	Ikarus	⚠️ Trojan.Crackz	

After that user has executed the mmcrack.exe file to open it and which cause of the start of other subprocess and infection.

- Provide event happened after the exploits (Timeline)

mftparser can be used to scans for potential Master File Table (MFT) entries in memory (using "FILE" and "BAAD" signatures) and prints out information for certain attributes, currently: \$FILE_NAME (\$FN), \$STANDARD_INFORMATION (\$SI), \$FILE_N and \$SI attributes from the \$ATTRIBUTE_LIST, \$OBJECT_ID (default output only) and resident \$DATA.

:3	2019-01-18 12:01:22 UTC+0000	[HBASE_BLOCKTimeStamp]	\SystemRoot\System32\Config\SOFTWARE
:4	2019-01-18 12:01:19 UTC+0000	[NETWORK CONNECTION]	0.0.0.0->*.*
:5	2019-01-18 12:01:19 UTC+0000	[NETWORK CONNECTION]	::0->*.*
:6	2019-01-18 12:01:15 UTC+0000	[HBASE_BLOCKTimeStamp]	\REGISTRY\MACHINE\SYSTEM
:7	2019-01-18 12:01:13 UTC+0000	[Handle (Key)]	MACHINE\CONTROLSET001\SERVICES\SHAREDACCESS\EPOCH
:8	2019-01-18 12:01:13 UTC+0000	[Handle (Key)]	MACHINE\CONTROLSET001\SERVICES\SHAREDACCESS\EPOCH
:9	2019-01-18 12:01:13 UTC+0000	[Handle (Key)]	MACHINE\CONTROLSET001\SERVICES\SHAREDACCESS\EPOCH
:0	2019-01-18 12:01:13 UTC+0000	[Handle (Key)]	MACHINE\CONTROLSET001\SERVICES\SHAREDACCESS\EPOCH
:1	2019-01-18 12:01:13 UTC+0000	[Handle (Key)]	MACHINE\CONTROLSET001\SERVICES\SHAREDACCESS\EPOCH
:2	2019-01-18 12:01:13 UTC+0000	[Handle (Key)]	MACHINE\CONTROLSET001\SERVICES\SHAREDACCESS\EPOCH
:3	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:4	2019-01-18 12:01:11 UTC+0000	[DLL LOADTIME (dll)]	apphelp.dll
:5	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:6	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:7	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:8	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:9	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:0	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:1	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:2	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:3	2019-01-18 12:01:11 UTC+0000	[DLL LOADTIME (dll)]	DEVOBJ.dll
:4	2019-01-18 12:01:11 UTC+0000	[DLL LOADTIME (dll)]	SETUPAPI.dll
:5	2019-01-18 12:01:11 UTC+0000	[DLL LOADTIME (dll)]	comctl32.dll
:6	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:7	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:8	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:9	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:0	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:1	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:2	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:3	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:4	2019-01-18 12:01:11 UTC+0000	[PROCESS]	WINI.EXE
:5	2019-01-18 12:01:11 UTC+0000	[Handle (Key)]	USER
:6	2019-01-18 12:01:11 UTC+0000	[DLL LOADTIME (dll)]	DEVOBJ.dll

- Registry changes

Listed down all the registries and find if there changes related to the malicious file.

```
vol.py -f is7.elf --profile=Win7SP1x86 userassist
```

```

REG_BINARY C:\Users\mis-win7\Downloads\mmcrackz\mmcrack.exe :
Count: 1
Focus Count: 0
Time Focused: 0.00.04.468000
Last updated: 2019-01-18 12:01:03 UTC+0000
Raw Data:
0x00000000 00 00 00 00 01 00 00 00 00 00 00 00 80 0f 00 00 ..... .
0x00000010 00 00 80 bf 00 00 80 bf 00 00 80 bf 00 00 80 bf ..... .
0x00000020 00 00 80 bf 00 00 80 bf 00 00 80 bf 00 00 80 bf ..... .
0x00000030 00 00 80 bf 00 00 80 bf ff ff ff ff 90 47 65 7c ..... Ge| .
0x00000040 25 af d4 01 00 00 00 00 ..... %..... .
-----
```

Approach

1. With the artefact provided and as it is memory dump, decided to choose appropriate tools which help to analyse and examine memory dump.
2. Initially started the analysis with the tool called Volatility.
3. Able to find information related to the infrastructure related information with the analysis by the Volatility.
 - a. System/OS information
 - b. Hardware compatibility
 - c. Processes list of the computer when taking the memory dump
 - d. Processes tree of the computer when taking the memory dump
4. Once analyse the process tree, identified suspicious processes and suspected that it is malicious activity such as Virus/Worm.
5. Then tried to find it via common virus databases.
 - a. <https://packetstormsecurity.com>
 - b. <https://www.virustotal.com>
6. Founded the it is trojan process that uses to open victims computer port to anywhere.

Conclusion

Victim user has downloaded mmcrack.zip file via chrome browser and extracted to the computer and executed the mmcrack.exe file. Which caused to open WINI.exe malicious process to be started and it has opened some ports to attacker.

***** End *****