**Exciting Digital Forensic Exercises from the Textbook**

Chapter 3: The Volatility Framework

4. Run the kdbgscan plugin against a Windows memory sample.

1. What profile does it suggest?

I ran the **kdbgscan** plugin against 9 different Windows memory smaple which can be downloaded from the textbook’s website

The results are as follows:

**root@kali:~# volatility -f /media/root/SHAD/windows/sample001.bin kdbgscan**

Volatility Foundation Volatility Framework 2.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x8054cde0

Offset (P) : 0x54cde0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP3x86

Version64 : 0x8054cdb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 3

Build string (NtBuildLab) : 2600.xpsp.080413-2111

PsActiveProcessHead : 0x80561358 (21 processes)

PsLoadedModuleList : 0x8055b1c0 (96 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x8054cde0

Offset (P) : 0x54cde0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP2x86

Version64 : 0x8054cdb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 3

Build string (NtBuildLab) : 2600.xpsp.080413-2111

PsActiveProcessHead : 0x80561358 (21 processes)

PsLoadedModuleList : 0x8055b1c0 (96 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

**root@kali:~# volatility -f /media/root/SHAD/windows/sample002.bin kdbgscan**

Volatility Foundation Volatility Framework 2.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: /media/root/SHAD/windows/sample002.bin WinXPSP2x86 (5.1.0 32bit)

Offset (P) : 0x292dc28

KDBG owner tag check : True

Profile suggestion (KDBGHeader): Win7SP1x86\_23418

Version64 : 0x292dc00 (Major: 15, Minor: 7601)

PsActiveProcessHead : 0x82945ba8

PsLoadedModuleList : 0x8294d4d0

KernelBase : 0x82804000

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: /media/root/SHAD/windows/sample002.bin WinXPSP2x86 (5.1.0 32bit)

Offset (P) : 0x292dc28

KDBG owner tag check : True

Profile suggestion (KDBGHeader): Win7SP1x86

Version64 : 0x292dc00 (Major: 15, Minor: 7601)

PsActiveProcessHead : 0x82945ba8

PsLoadedModuleList : 0x8294d4d0

KernelBase : 0x82804000

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: /media/root/SHAD/windows/sample002.bin WinXPSP2x86 (5.1.0 32bit)

Offset (P) : 0x292dc28

KDBG owner tag check : True

Profile suggestion (KDBGHeader): Win7SP0x86

Version64 : 0x292dc00 (Major: 15, Minor: 7601)

PsActiveProcessHead : 0x82945ba8

PsLoadedModuleList : 0x8294d4d0

KernelBase : 0x82804000

**root@kali:~# volatility -f /media/root/SHAD/windows/sample004.bin kdbgscan**

Volatility Foundation Volatility Framework 2.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x8054cde0

Offset (P) : 0x54cde0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP3x86

Version64 : 0x8054cdb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 3

Build string (NtBuildLab) : 2600.xpsp.080413-2111

PsActiveProcessHead : 0x80561358 (20 processes)

PsLoadedModuleList : 0x8055b1c0 (97 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x8054cde0

Offset (P) : 0x54cde0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP2x86

Version64 : 0x8054cdb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 3

Build string (NtBuildLab) : 2600.xpsp.080413-2111

PsActiveProcessHead : 0x80561358 (20 processes)

PsLoadedModuleList : 0x8055b1c0 (97 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

**root@kali:~# volatility -f /media/root/SHAD/windows/sample005.bin kdbgscan**

Volatility Foundation Volatility Framework 2.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: /media/root/SHAD/windows/sample005.bin WinXPSP2x86 (5.1.0 32bit)

Offset (P) : 0x5583d0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): Win2003SP1x86

Version64 : 0x5583a8 (Major: 15, Minor: 3790)

PsActiveProcessHead : 0x8056ede8

PsLoadedModuleList : 0x80568c08

KernelBase : 0x804de000

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: /media/root/SHAD/windows/sample005.bin WinXPSP2x86 (5.1.0 32bit)

Offset (P) : 0x5583d0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): Win2003SP2x86

Version64 : 0x5583a8 (Major: 15, Minor: 3790)

PsActiveProcessHead : 0x8056ede8

PsLoadedModuleList : 0x80568c08

KernelBase : 0x804de000

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: /media/root/SHAD/windows/sample005.bin WinXPSP2x86 (5.1.0 32bit)

Offset (P) : 0x5583d0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): Win2003SP0x86

Version64 : 0x5583a8 (Major: 15, Minor: 3790)

PsActiveProcessHead : 0x8056ede8

PsLoadedModuleList : 0x80568c08

KernelBase : 0x804de000

root@kali:~# volatility -f /media/root/SHAD/windows/sample006.bin kdbgscan

Volatility Foundation Volatility Framework 2.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x8054cde0

Offset (P) : 0x54cde0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP3x86

Version64 : 0x8054cdb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 3

Build string (NtBuildLab) : 2600.xpsp\_sp3\_gdr.090804-1435

PsActiveProcessHead : 0x80561358 (38 processes)

PsLoadedModuleList : 0x8055b1c0 (112 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x8054cde0

Offset (P) : 0x54cde0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP2x86

Version64 : 0x8054cdb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 3

Build string (NtBuildLab) : 2600.xpsp\_sp3\_gdr.090804-1435

PsActiveProcessHead : 0x80561358 (38 processes)

PsLoadedModuleList : 0x8055b1c0 (112 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

**root@kali:~# volatility -f /media/root/SHAD/windows/sample007.bin kdbgscan**

Volatility Foundation Volatility Framework 2.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x80545ae0

Offset (P) : 0x545ae0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP3x86

Version64 : 0x80545ab8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 3

Build string (NtBuildLab) : 2600.xpsp.080413-2111

PsActiveProcessHead : 0x8055a158 (31 processes)

PsLoadedModuleList : 0x80553fc0 (122 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x80545ae0

Offset (P) : 0x545ae0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP2x86

Version64 : 0x80545ab8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 3

Build string (NtBuildLab) : 2600.xpsp.080413-2111

PsActiveProcessHead : 0x8055a158 (31 processes)

PsLoadedModuleList : 0x80553fc0 (122 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

**root@kali:~# volatility -f /media/root/SHAD/windows/sample008.bin kdbgscan**

Volatility Foundation Volatility Framework 2.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x80544ce0

Offset (P) : 0x544ce0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP3x86

Version64 : 0x80544cb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 2

Build string (NtBuildLab) : 2600.xpsp\_sp2\_rtm.040803-2158

PsActiveProcessHead : 0x80559258 (25 processes)

PsLoadedModuleList : 0x805531a0 (119 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x80544ce0

Offset (P) : 0x544ce0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP2x86

Version64 : 0x80544cb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 2

Build string (NtBuildLab) : 2600.xpsp\_sp2\_rtm.040803-2158

PsActiveProcessHead : 0x80559258 (25 processes)

PsLoadedModuleList : 0x805531a0 (119 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

**root@kali:~# volatility -f /media/root/SHAD/windows/sample009.bin kdbgscan**

Volatility Foundation Volatility Framework 2.6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x80544ee0

Offset (P) : 0x544ee0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP3x86

Version64 : 0x80544eb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 2

Build string (NtBuildLab) : 2600.xpsp\_sp2\_gdr.070227-2254

PsActiveProcessHead : 0x80559458 (22 processes)

PsLoadedModuleList : 0x805533a0 (112 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Instantiating KDBG using: Kernel AS WinXPSP2x86 (5.1.0 32bit)

Offset (V) : 0x80544ee0

Offset (P) : 0x544ee0

KDBG owner tag check : True

Profile suggestion (KDBGHeader): WinXPSP2x86

Version64 : 0x80544eb8 (Major: 15, Minor: 2600)

Service Pack (CmNtCSDVersion) : 2

Build string (NtBuildLab) : 2600.xpsp\_sp2\_gdr.070227-2254

PsActiveProcessHead : 0x80559458 (22 processes)

PsLoadedModuleList : 0x805533a0 (112 modules)

KernelBase : 0x804d7000 (Matches MZ: True)

Major (OptionalHeader) : 5

Minor (OptionalHeader) : 1

KPCR : 0xffdff000 (CPU 0)

It is good to note that the **kdbgscan** plugin gives more profile information than **imageinfo** plugin. Below is an output using the **imageinfo** plugin on a Windows memory sample.

**root@kali:~# volatility -f /media/root/SHAD/windows/sample003.bin imageinfo**

Volatility Foundation Volatility Framework 2.6

INFO : volatility.debug : Determining profile based on KDBG search...

Suggested Profile(s) : WinXPSP2x86, WinXPSP3x86 (Instantiated with WinXPSP2x86)

AS Layer1 : IA32PagedMemoryPae (Kernel AS)

AS Layer2 : FileAddressSpace (/media/root/SHAD/windows/sample003.bin)

PAE type : PAE

DTB : 0x319000L

KDBG : 0x80545b60L

Number of Processors : 1

Image Type (Service Pack) : 3

KPCR for CPU 0 : 0xffdff000L

KUSER\_SHARED\_DATA : 0xffdf0000L

Image date and time : 2008-11-26 07:46:02 UTC+0000

Image local date and time : 2008-11-26 02:46:02 -0500

1. What is the virtual address of the kernel debugger data structure?

The virtual addresses of the kernel debugger datat structure for each of the memory sample is listed on the **Offset(V)** row for each of the outputs above.

1. Were any inaccurate profiles suggested? Why or why not?

No profile suggested was inaccurate. The way a suggested profile is identified as being inaccurate is if the number of processes and modules listed for a memory sample is 0. According to the textbook’s solution, one of the suggested profile was identified as being inaccurate because it had 0 processes and 0 modules.

**Chapter 5: Windows Objects and Pool Allocation**

4. Run the pslist and psscan plugins against sample003.bin.

The output when the **pslist** plugin is ran against sample003.bin

**root@kali:~# volatility -f /media/root/SHAD/windows/sample003.bin --profile=WinXPSP2x86 pslist**

Volatility Foundation Volatility Framework 2.6

Offset(V) Name PID PPID Thds Hnds Sess Wow64 Start Exit

---------- -------------------- ------ ------ ------ -------- ------ ------ ------------------------------ ------------------------------

0x819cc830 System 4 0 51 254 ------ 0

0x817e4670 smss.exe 360 4 3 19 ------ 0 2008-11-26 07:38:11 UTC+0000

0x8181bd78 csrss.exe 596 360 10 322 0 0 2008-11-26 07:38:13 UTC+0000

0x8182b100 winlogon.exe 620 360 16 503 0 0 2008-11-26 07:38:14 UTC+0000

0x8183ba78 services.exe 672 620 15 245 0 0 2008-11-26 07:38:15 UTC+0000

0x817dbc30 lsass.exe 684 620 21 347 0 0 2008-11-26 07:38:15 UTC+0000

0x81859d70 svchost.exe 844 672 19 198 0 0 2008-11-26 07:38:18 UTC+0000

0x8183d360 svchost.exe 932 672 10 229 0 0 2008-11-26 07:38:18 UTC+0000

0x818a2300 svchost.exe 1064 672 63 1308 0 0 2008-11-26 07:38:20 UTC+0000

0x817f7da0 svchost.exe 1164 672 5 77 0 0 2008-11-26 07:38:23 UTC+0000

0x8180e6f0 svchost.exe 1264 672 14 209 0 0 2008-11-26 07:38:25 UTC+0000

0x817ca478 explorer.exe 1516 1452 12 362 0 0 2008-11-26 07:38:27 UTC+0000

0x816e75e8 spoolsv.exe 1648 672 12 112 0 0 2008-11-26 07:38:28 UTC+0000

0x816af860 VMwareTray.exe 1896 1516 1 26 0 0 2008-11-26 07:38:31 UTC+0000

0x816af448 VMwareUser.exe 1904 1516 1 28 0 0 2008-11-26 07:38:31 UTC+0000

0x816a13c0 VMwareService.e 1756 672 3 45 0 0 2008-11-26 07:38:45 UTC+0000

0x816557e0 alg.exe 512 672 6 105 0 0 2008-11-26 07:38:53 UTC+0000

0x81643b28 wuauclt.exe 1372 1064 8 225 0 0 2008-11-26 07:39:38 UTC+0000

0x8164e3a8 wscntfy.exe 560 1064 1 31 0 0 2008-11-26 07:44:57 UTC+0000

The output when the **psscan** plugin is ran against sample003.bin

**root@kali:~# volatility -f /media/root/SHAD/windows/sample003.bin --profile=WinXPSP2x86 psscan**

Volatility Foundation Volatility Framework 2.6

Offset(P) Name PID PPID PDB Time created Time exited

------------------ ---------------- ------ ------ ---------- ------------------------------ ------------------------------

0x000000000181b748 alg.exe 992 660 0x08140260 2008-11-15 23:43:25 UTC+0000

0x0000000001843b28 wuauclt.exe 1372 1064 0x08140180 2008-11-26 07:39:38 UTC+0000

0x000000000184e3a8 wscntfy.exe 560 1064 0x081402a0 2008-11-26 07:44:57 UTC+0000

0x00000000018557e0 alg.exe 512 672 0x08140260 2008-11-26 07:38:53 UTC+0000

0x000000000185dda0 cmd.exe 940 1516 0x081401a0 2008-11-26 07:43:39 UTC+0000 2008-11-26 07:45:49 UTC+0000

0x00000000018a13c0 VMwareService.e 1756 672 0x08140220 2008-11-26 07:38:45 UTC+0000

0x00000000018af448 VMwareUser.exe 1904 1516 0x08140100 2008-11-26 07:38:31 UTC+0000

0x00000000018af860 VMwareTray.exe 1896 1516 0x08140200 2008-11-26 07:38:31 UTC+0000

0x00000000018e75e8 spoolsv.exe 1648 672 0x081401e0 2008-11-26 07:38:28 UTC+0000

0x00000000019456e8 csrss.exe 592 360 0x08140040 2008-11-15 23:42:56 UTC+0000

0x0000000001946020 svchost.exe 828 660 0x081400c0 2008-11-15 23:42:57 UTC+0000

0x00000000019467e0 services.exe 660 616 0x08140080 2008-11-15 23:42:56 UTC+0000

0x000000000194f658 svchost.exe 1016 660 0x08140100 2008-11-15 23:42:57 UTC+0000

0x00000000019533c8 svchost.exe 924 660 0x081400e0 2008-11-15 23:42:57 UTC+0000

0x00000000019ca478 explorer.exe 1516 1452 0x081401c0 2008-11-26 07:38:27 UTC+0000

0x00000000019dbc30 lsass.exe 684 620 0x081400a0 2008-11-26 07:38:15 UTC+0000

0x00000000019e4670 smss.exe 360 4 0x08140020 2008-11-26 07:38:11 UTC+0000

0x00000000019f7da0 svchost.exe 1164 672 0x08140140 2008-11-26 07:38:23 UTC+0000

0x0000000001a0e6f0 svchost.exe 1264 672 0x08140160 2008-11-26 07:38:25 UTC+0000

0x0000000001a1bd78 csrss.exe 596 360 0x08140040 2008-11-26 07:38:13 UTC+0000

0x0000000001a2b100 winlogon.exe 620 360 0x08140060 2008-11-26 07:38:14 UTC+0000

0x0000000001a3ba78 services.exe 672 620 0x08140080 2008-11-26 07:38:15 UTC+0000

0x0000000001a3d360 svchost.exe 932 672 0x081400e0 2008-11-26 07:38:18 UTC+0000

0x0000000001a59d70 svchost.exe 844 672 0x081400c0 2008-11-26 07:38:18 UTC+0000

0x0000000001aa2300 svchost.exe 1064 672 0x08140120 2008-11-26 07:38:20 UTC+0000

0x0000000001bcc830 System 4 0 0x00319000

1. Which process(es) are active?

All processes in the pslist output are active

1. Which process(es) have terminated?

According to the psscan output, only one process is terminated (cmd.exe pid 940)

1. Which process(es) are leftover from a previous reboot?

All of the processes that started on 2008-11-15 are leftover from a previous reboot because the latest boot was carried out on 2008-11-26.

**Chapter 6: Processes, Handles, and Tokens**

3. Run the psxview plugin against sample003.bin.

The output of the psxview plugin when ran against sample003.bin is:

**root@kali:~# volatility -f /media/root/SHAD/windows/sample003.bin --profile=WinXPSP2x86 psxview --apply-rules**

Volatility Foundation Volatility Framework 2.6

Offset(P) Name PID pslist psscan thrdproc pspcid csrss session deskthrd ExitTime

---------- -------------------- ------ ------ ------ -------- ------ ----- ------- -------- --------

0x01a2b100 winlogon.exe 620 True True True True True True True

0x01a3d360 svchost.exe 932 True True True True True True True

0x018a13c0 VMwareService.e 1756 True True True True True True True

0x018e75e8 spoolsv.exe 1648 True True True True True True True

0x019dbc30 lsass.exe 684 True True True True True True True

0x0184e3a8 wscntfy.exe 560 True True True True True False True

0x018af860 VMwareTray.exe 1896 True True True True True True True

0x01a4bc20 network\_listene 1696 False False True True True False True

0x01843b28 wuauclt.exe 1372 True True True True True True True

0x01a59d70 svchost.exe 844 True True True True True True True

0x018af448 VMwareUser.exe 1904 True True True True True True True

0x019f7da0 svchost.exe 1164 True True True True True True True

0x018557e0 alg.exe 512 True True True True True True True

0x01a3ba78 services.exe 672 True True True True True True True

0x019ca478 explorer.exe 1516 True True True True True True True

0x01a0e6f0 svchost.exe 1264 True True True True True True True

0x01aa2300 svchost.exe 1064 True True True True True True True

0x019e4670 smss.exe 360 True True True True Okay Okay Okay

0x01bcc830 System 4 True True True True Okay Okay Okay

0x01a1bd78 csrss.exe 596 True True True True Okay True True

0x01946020 svchost.exe 828 False True False False False False False

0x019533c8 svchost.exe 924 False True True False False False False

0x0185dda0 cmd.exe 940 Okay True Okay Okay Okay Okay Okay 2008-11-26 07:45:49 UTC+0000

0x019467e0 services.exe 660 False True True False False False False

0x0181b748 alg.exe 992 False True True False False False False

0x0194f658 svchost.exe 1016 False True True False False False False

0x019456e8 csrss.exe 592 False True True False Okay False False

1. Which process(es) are hidden?

When the pslist and psscan plugins were executed, the "network\_listene" process wasn’t listed. This means that it was hidden.

1. In what ways did the rootkit attempt to hide?

The **psscan** plugin finds processes that previously terminated (inactive) and processes that have been hidden or unlinked by a rootkit. The rootkit attempted to hide this process by overwriting the pool tag values.

The **pslist** plugin lists the processes of a system. This walks the doubly-linked list pointed to by PsActiveProcessHead.The rootkit was able to hide this process by unlinking it from PsActiveProcessHead.