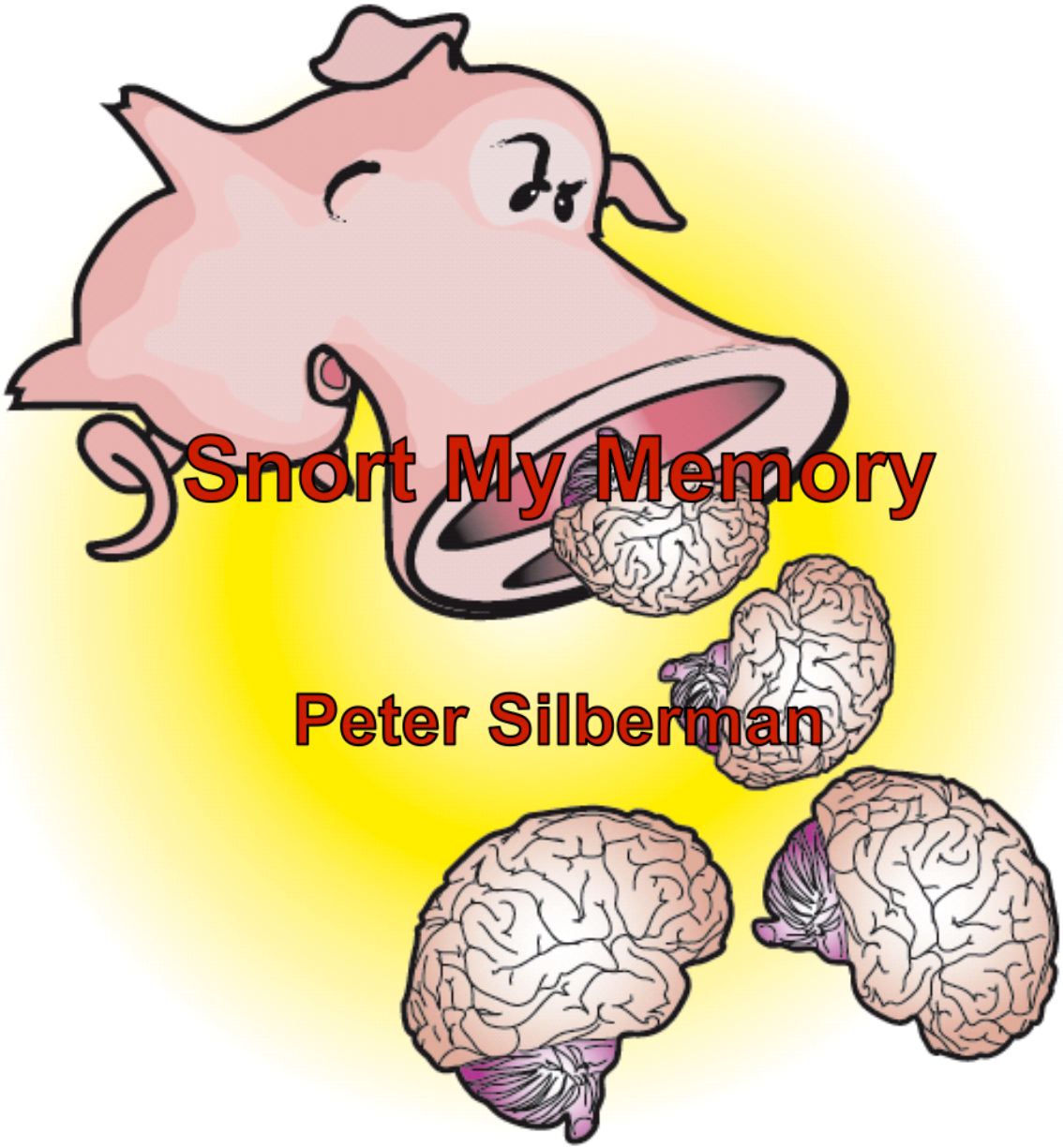

Snort My Memory
Peter Silberman

Who Am I?

- Peter Silberman
  - Researcher/Developer at MANDIANT on the product team
  - Co-Wrote (Proof of Concepts) RAIDE and FUTo
  - Co-Developed “Advanced Memory Forensics in Incident Response”
  - Wrote Audit Viewer
  - Contributor to the Uninformed Journal
  - Security Researcher:
    - Found flaws in: CA, Windows, AVG, ZoneAlarm, Kaspersky, Kerio

Talk Overview

- What is snorting memory
- Concepts critical to snorting memory
  - Accessing/using physical memory
    - Enumerating Strings
  - Snort signatures
- Discuss Snorting Memory
  - Identify malware before it goes over the wire
  - New tool MindSniffer
- Theory Mets Reality – DEMO and tool release

MANDIANT®

INTELLIGENT INFORMATION SECURITY

What is *Snorting Memory?*

- Snorting Memory: “the ability to apply a snort signature to a processes’ enumerated strings.”
- Two step process:
  1. Translate snort signatures:
     - MindSniffer
  2. Use Memoryze to enumerate each processes’ strings.
     - Snort XPath filter to strings being found in memory
       OR
     - Load XML results into Audit Viewer, apply snort signatures

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Snort My ***Memory***: Critical Concepts

- Need to understand how to access memory
  - Access memory for two purposes
    - Live Analysis
    - Dumping Memory
  - Virtual to physical address translation
- Note: Signatures work on live/dead memory

Accessing Physical Memory

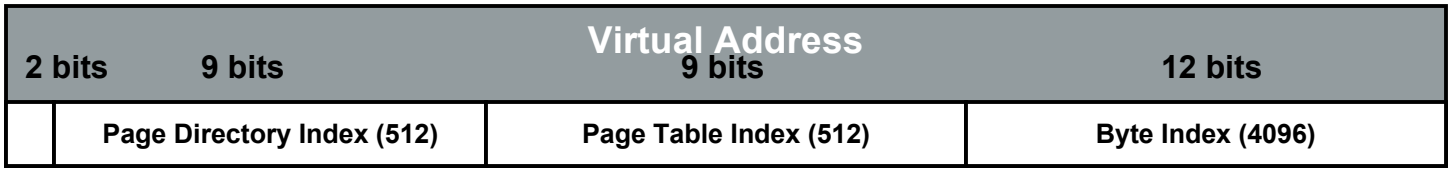
- \Device\PhysicalMemory
  - Section object exposed by Windows
    - Reading from the handle allows application to read physical memory

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Translating Virtual to Physical

- Memoryze implements its own virtual to physical address translation:
  - Every virtual address needs to be translated to a physical address within the section object
    - Non PAE:

    - PAE:



MANDIANT®
INTELLIGENT INFORMATION SECURITY

Accessing Physical Memory

- Map physical memory into buffer
- Write the buffer or parse the buffer
  - Scan buffer for some characteristic (Live Analysis)
  - Write buffer to dump (Offline)
- Problem: User-mode access is not allowed beginning in Windows 2003 SP1 http://technet.microsoft.com/en-us/library/cc787565.aspx

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Physical Memory

- Reading physical memory benefits:
  - Allows reading of process memory without debugging
    - Defeats anti debugging techniques
    - Gives a very complete picture of binaries that are packed (even themida)
    - Bypasses rootkits using dr register technique
      - http://www.invisiblethings.org/papers/chameleon_concepts.pdf

Enumerating Strings

- Find every process in physical memory:
  - Processes are represented as EPROCESS blocks in the kernel
    - Parse EPROCESS' memory sections:
      - The VadRoot within the EPROCESS
        - Tree of MMVAD entries
      - MMVAD entries contain the virtual start address and size of each memory section within a process
        - Entries can represent heap, stack, binary images
        - Helps manages process' virtual address space
      - Scan from virtual start -> virtual end
        - Convert unicode strings containing US ascii characters to ASCII strings

VAD Structure

- lkd> dt nt!_MMVAD
- **+0x000 StartingVpn : Uint4B**
- **+0x004 EndingVpn : Uint4B**
- **+0x008 Parent : Ptr32 _MMVAD**
- **+0x00c LeftChild : Ptr32 _MMVAD**
- **+0x010 RightChild : Ptr32 _MMVAD**
- **+0x014 u : __unnamed**
- **+0x018 ControlArea : Ptr32 _CONTROL_AREA**
- +0x01c FirstPrototypePte : Ptr32 _MMPTE
- +0x020 LastContiguousPte : Ptr32 _MMPTE
- +0x024 u2 : __unnamed

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Enumerating Strings

- OllyDbg's memory map view shows different sections

Address	Size	Owner	Section	Contains	Type	Access	Initial
00010000	00001000				Priv	RW	RW
00020000	00001000				Priv	RW	RW
00030000	00001000				Priv	RW	RW
0007B000	00001000				Priv	RW Gua	RW
0007C000	00004000			stack of ma	Priv	RW Gua	RW
00080000	00003000				Map	R	R
00090000	00002000				Map	R	R
000A0000	00010000				Priv	RW	RW
001A0000	00006000				Priv	RW	RW
001B0000	00003000				Map	RW	RW

- Each address range is:
  - An entry in VadRoot, represented by MMVAD structure

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Enumerating Strings

- Safe assumption: VAD tree has not been modified?
  - Tests showed modifying/removing a VAD will result in the operating system blue screening

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Snort My Memory

- What is snort?
  - Snort open source Intrusion Detection System (IDS)
  - Widely deployed and used
  - Applies signatures and rules to network traffic

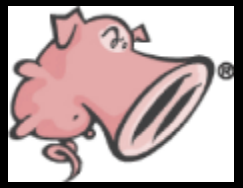
Only one more slide on snort

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Advantages to Using Snort

- Public signature/rule repository
  - Community maintains and updates signatures:
    - http://www.emergingthreats.net
  - Signatures range from policy violation, to shellcode detection
    - Most of these signatures have application in memory
- Very active community

MANDIANT®
INTELLIGENT INFORMATION SECURITY



My Memory

- Review:
  - Snort signatures applied to strings in memory
- Problem snorting memory is solving:
  - Quick identification of potential infection
- ***REMEMBER:*** Nothing found DOES NOT mean nothings there

MANDIANT®

INTELLIGENT INFORMATION SECURITY

Snort My Memory

- Why do snort signatures work in memory:
  - Snort identifies network indicators
    - Indicators generated *(usually)* by processes
    - Socket API requires strings in memory
  - Identifying strings in a process' memory space
    - Applying a given signature is very similar to how snort works
      - Potential for earlier identification

Snort My Memory

- Reasons why applying snort signatures in memory works:
  - Executables can contain strings in their .data section
  - Strings that are freed are still in memory (no garbage collection, and no zeroing of strings)
  - Malicious executables are not system files
    - Memoryze, can parse the paging file ensuring high percentage of data

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Snort My Memory

- What snort in memory can identify:
  - Malware signatures
    - Optix Pro
    - Gimmiv
    - Waledac
  - (ASCII) Payloads
    - Metasploit
  - Policy violations
    - Lots of “secret” strings in unencrypted memory
      - Passwords anyone?

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Snort My Memory

- Benefits of applying snort signatures to memory:
  - Using snorts signature set, we get continually updated signatures
  - Commercially available signatures ***should?*** have the potential to be used in memory
  - Dormant malware
  - Malware is *mostly* taking a file system up view, and not a memory down view.
    - Hide files, registry keys not strings

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Snort My Memory

- Potential problems snorting memory:
  - Strings in dynamic memory (stack or heap) may not stay around along enough to be hit by a signature
    - Timing is crucial
      - Partial signatures
  - Only ASCII characters (0x20-0x7E) can be used in signatures
  - Malware encrypting strings and zeroing them out after use
  - Some signatures will not be in memory.
  - Dead memory

MindSniffer

MindSniffer

http://www.mandiant.com/software/mms.htm

- Purpose
  - Translation from snort signature -> usable memory format
- MindSniffer:
  - python utility used in conjunction with Memoryze and or Audit Viewer.
    - What is Memoryze
    - What is Audit Viewer

MANDIANT®

INTELLIGENT INFORMATION SECURITY

MindSniffer

- MindSniffer generates:
  - An XML file that Memoryze understands and runs. The XML file will have an XPath filter in it that represents the parsed snort signature.
    OR
  - A python file that Audit Viewer loads and can apply to the result of some audit previously performed.

MANDIANT®
INTELLIGENT INFORMATION SECURITY

MindSniffer.py

- MindSniffer takes the following parameters:
  - -r - input to parse
  - -x - generate signatures as separate xpath audits
  - -p - generate python files to be used by audit viewer
  - -o - output the files to a directory
  - -n - specify or in xpath filter generation
  - -m - specify memory image to be used in xpath filter

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Memoryze

http://www.mandiant.com/software/memoryze.htm

- Memoryze
  - Will acquire memory dumps
  - List loaded drivers (including attached devices)
  - Enumerates within all processes
    - The handle table
    - The memory sections
    - The open ports
    - The strings
  - Detected dll injection
  - Acquire processes and drivers
  - Detect kernel hooks
  - When analyzing live memory, Memoryze will utilize the paging file to get a better picture of memory
  - Suports Windows 2000-2003, BETA support for Vista all 32 bit at the moment

MANDIANT®

INTELLIGENT INFORMATION SECURITY

Free Tool: Audit Viewer

http://www.mandiant.com/software/mav.htm

- Audit Viewer allows the user to quickly view complex XML output in an easily readable format. Using familiar grouping of data and search capabilities, Audit Viewer makes memory analysis quicker and more intuitive.
  - Ability to search Files, Processes, Mutants, Events, Registry Keys, and Strings using plain text or regex.
  - Ability to load multiple Memoryze result sets contained in the same directory.
  - Handle types are separated out into more abstract types representing the logical type of the handle such as Files, Directories (part of the Object Manager's namespace), Processes, Keys, Mutants, and Events.
  - And More http://blog.mandiant.com/archives/50

MANDIANT®

INTELLIGENT INFORMATION SECURITY

Snort Signatures vs. Translated Signatures

- Snort applies AND logic
  - content:"**GET** "; uricontent:"**/postcard.exe**“
- Audit Viewer uses OR
- Memoryze can use either
- OR, can result in higher false positives

MANDIANT®
INTELLIGENT INFORMATION SECURITY

MindSniffer generated XML file

- Snort Signature:
  alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET TROJAN Gimmiv.A.dll Infection"; flow: to_server,established; uricontent:"**/test**"; uricontent:"**.php**"; uricontent**:"?abc=**"; uricontent:"**?def=**"; reference:url,www.microsoft.com/security/portal/Entry.aspx?name=TrojanSpy%3aWin32%2fGimmiv.A; classtype:trojan-activity; sid:2008689; rev:2;)
- XPath Filter

<value xsi:type="xsd:string">//*[(contains(StringList, '**/test**') and contains(StringList, '**.php**') and contains(StringList, '**?abc=**') and contains(StringList, '**?def=**'))]</value>

MANDIANT®
INTELLIGENT INFORMATION SECURITY

MindSniffer generated XML file

- Snort Signature:

alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET TROJAN Hitpop.AG/Pophot.az HTTP Checkin"; flow:to_server,established; content:"**GET** "; depth:4; uricontent:"**.asp**"; nocase; uricontent:"**|3F|ver=**"; nocase; uricontent:"**|26|tgid=**"; nocase; uricontent:"**|26|address=**"; nocase; pcre:"**/address\=([0-9A-F][0-9A-F]-){5}([0-9A-F][0-9A-F])/i**"; classtype:trojan-activity; sid:2008317; rev:2;)

- XPath Filter:

<value xsi:type="xsd:string">//*[(contains(StringList, '**GET** ') and contains(StringList, '**.asp**') and contains(StringList, '**?ver=**') and contains(StringList, '**&tgid=**') and contains(StringList, '**&address=**') and matches(StringList, '**address\=([0-9A-F][0-9A-F]-){5}([0-9A-F][0-9A-F])**'"))]</value>

MANDIANT®

INTELLIGENT INFORMATION SECURITY

MindSniffer generated XML file

- Snort Signature:

  alert tcp $HOME_NET any -> $EXTERNAL_NET 1024: (msg:"ET TROJAN Perfect Keylogger FTP Initial Install Log Upload (Null obfuscated)"; flow:established,to_server; content:"**C|00|o|00|n|00|g|00|r|00|a|00|t|00|u|00|l|00|a|00|t|00|i|00|o|00|n|00|s|00|!|00| |00|P|00|e|00|r|00|f|00|e|00|c|00|t|00| |00|K|00|e|00|l|00|o|00|g|00|g|00|e|00|r|00| |00|w|00|a|00|s|00| |00|s|00|u|00|c|00|c|00|e|00|s|00|s|00|f|00|u|00|l|00|l|00|y|00| |00|i|00|n|00|s|00|t|00|a|00|l|00|l|00|e|00|d|00|**"; classtype:trojan-activity; sid:2008327; rev:1;)

- XPath Filter:

  <value xsi:type="xsd:string">//*[(contains(StringList, **'Congratulations! Perfect Kelogger was successfully installed**'))]

MANDIANT®

INTELLIGENT INFORMATION SECURITY

MindSniffer generated XML file

- Snort Signature:

alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET CURRENT_EVENTS Possible XML 0-day for Internet Explorer Exploitation Attempt (obfuscation 1)"; flow:established,from_server; content:"**|7c|XML|7c 7c|if|7c|SPAN|7c|navigator|7c|CDATA|7c|http|7c|com|7c|w2k3|7c|appVersion|7c|version|7c|nt|7c 7c|X|7c|MSIE|7c|wxp|7c|114|7c|HTML|7c|DATAFLD|7c|DATASRC|7c|DATAFORMATAS|7c|ID|7c|while|7c|2003|7c|";** classtype:web-application-attack; […]

- XPath Filter:

<value xsi:type="xsd:string">//*[(contains(StringList, **'|XML||if|SPAN|navigator|CDATA|http|com|w2k3|appVersion|version|nt||X|MSIE|wxp|114|HTML|DATAFLD|DATASRC|DATAFORMATAS|ID|while|2003|'**))]</value>

MANDIANT®

INTELLIGENT INFORMATION SECURITY

MindSniffer generated XML file (caveat)

- alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET CURRENT_EVENTS Trojan resulting from Fake MS Updates Email Login to CnC"; flow:established,to_server; content:"**HTTP/1.0|*0d 0a*|User-Agent|3a| Mozilla/4.0 (compatible\; MSIE 6.0\; Windows XP 2600.xpsp.9786-27197)|*0d 0a*|**"; […]
- //*[(contains(StringList, ' HTTP/1.0') and contains(StringList, 'User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows XP 2600.xpsp.9786-27197)'))]
  - XML Sanitization results in \r \n being broken into separate elements.
  - MindSniffer mimics this by breaking apart signatures

MANDIANT®
INTELLIGENT INFORMATION SECURITY

MindSniffer python module

- Audit Viewer now imports anything from _Signatures\ directory
- Audit Viewer calls initialize for module which registers the “plugin” with Audit Viewer
- Users can then apply signatures to processes

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Future of MindSniffer

- Better parsing of snort rules
- Better generation of translated snort rules
  - Utilize pcre flags
- **More and More and More testing**

Lessons learned

- Over the course of testing
  - Snort - Variant specific signatures
  - Memory – ***(potential)*** for generic malware signaturs per variant
    - “Optix Pro v1.33” exists 5 times in an infected processes
      - Optix Pro v\d\.\d+

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Conclusion

- This is a very small first step
- The future is very bright for writing memory specific signatures
  - There are a lot more signaturable strings in memory
  - The future for snort signatures in memory is yet to be determined.
    - Gimmicky at best, not going to solve bad ass IR

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Thanks

- Anne M
- Product Team
- Kelcey Tietjen
- Lurene Grenier
- *Michael Sutton, and Shmoocon presenter for twitter idea*

DEMO

Theory Meets Reality

MANDIANT®

INTELLIGENT INFORMATION SECURITY

DEMO

- Generate Signatures
  - Find Waledac
  - Find Gimmiv (if time permits)

Waledac Signatures

- alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET TROJAN Waledac Beacon Traffic Detected"; flow:to_server,established; content:"**POST /**"; depth:6; content:"**|0d 0a|Referer\: Mozilla|0d 0a|**"; nocase; within:50; content:"**|0d 0a|User-Agent\: Mozilla|0d 0a|**"; within:120; content:"**a=**"; nocase; within: 100; classtype:trojan-activity;reference:url,www.shadowserver.org/wiki/pmwiki.php?n=Calendar.20081231; sid:2008958; rev:1;)

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Gimmiv Signature (Time??)

- alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET TROJAN Gimmiv.A.dll Infection"; flow: to_server,established; uricontent:"**/test**"; uricontent:"**.php**"; uricontent:"**?abc=**"; uricontent:"**?def=**"; reference:url,www.microsoft.com/security/portal/Entry.aspx?name=TrojanSpy%3aWin32%2fGimmiv.A; classtype:trojan-activity; sid:2008689; rev:2;)

MANDIANT®
INTELLIGENT INFORMATION SECURITY

Questions??

- Contact:

  peter.silberman@mandiant.com
- Blog:
  http://blog.mandiant.com
- Files:

  MindSniffer - http://www.mandiant.com/software/mms.htm
  Memoryze - http://www.mandiant.com/software/memoryze.htm
  Audit Viewer - http://www.mandiant.com/software/mav.htm

MANDIANT®
INTELLIGENT INFORMATION SECURITY