YROME

Memory Forensics: A Volatility Primer

Mariano Graziano

Security Day - Lille1 University

January 2015 - Lille, France

whoami

- Ph.D student at Eurecom (France)
- Msc from Politecnico di Torino (Italy)
- Main topics: Malware analysis, Memory forensics
- "Wasted" the best years on IRC
- Interests: Exploitation techniques, *Nix Kernel hacking, CTFs

YROMEM

Outline

YROEM

- Memory forensics
- Volatility
  - Windows
  - Linux
- Virtualization Support
  - Hypervisor Structures
  - Virtual Machines Analysis
- Future Work

Memory Forensics

YROMEM

Process of capturing a copy of the system memory (RAM) to extract a number of evidences that are useful for an investigation

- Steps:
  - Take the memory dump
  - Locate raw data structures
  - Extract information (encryption keys, passwords, etc)
- New field (~2005) and very active research area

YROMEM

Pros

- Memory is smaller than hard-drives
- Every attack has a memory footprint
- Advanced samples reside only in memory

Cons

- OS diversity:
  - Data structures
  - Semantic Gap
- Memory changes:
  - Content authenticity
  - Acquisition paradox

YROMEM

Outline

YROOMEM

- Memory forensics
- Volatility
  - Windows
  - Linux
- Virtualization Support
  - Hypervisor Structures
  - Virtual Machines Analysis
- Future Work

Memory Analysis

- Retrieve specific information (processes, IP addresses, etc)
- Fill the Semantic Gap
- Require OS internals knowledge (the more, the better)

YROMEM

Existing Frameworks

YROMEM

Don't reinvent the wheel!

- Volatility (Volatility Foundation)
- Memoryze (Mandiant)
- Rekall (Google)

Framework Internals

- They all share the same concepts
- Step 1: Locating structures
  - Fixed offsets
  - Data structures walking
  - Linear scanning
- Remember the OS diversity

YROME

Interesting Structures

- Depend on the OS
- Define your "interest"
- Processes?
  - EPROCESS, KPROCESS, PEB, etc
  - task_struct, mm_struct, etc

YROOM

_EPROCESS

YROMEM

_EPROCESS:

'Pcb': 0x0, '_KPROCESS',

'ProcessLock' : 0x98, '_EX_PUSH_LOCK',

'ActiveProcessLinks' : 0xb8, → Flink && Blink

.....

'Peb' : 0x1a8, '_PEB',

'PrefetchTrace' : 0x1ac, '_EX_FAST_REF',

.....

'_KPROCESS'
'Header' : 0x0, '_DISPATCHER_HEADER',
.....
'DirectoryTableBase' : 0x18,
'LdtDescriptor' : 0x1c, '_KGDTENTRY',
.....

Interesting Process Information

- EPROCESS:
  - Creation and Exit Time
  - PID && PPID
  - Pointer to the handler table
  - VAD etc
- PEB:
  - Pointer to the Image Base Address
  - Pointer to the DLLs loaded
  - Heap Size etc

YROOM

Volatility

YROEM

- Open Source Memory analysis framework born in 2007
- Python
- Current version 2.4 (August 2014)
- http://www.volatilityfoundation.org/#!24/c12wa
- FATKit Evolution (by Petroni and Walters, DFIR Journal 2006)

Volatility 2.4

- Windows (XP, Vista, 7, 2003, 2008, 8, 8.1)
- Linux 32 and 64 bit
- MacOSX 10.5 to 10.8.3
- Android
- It works with crash dumps, hibernation files, VM snapshots, Lime format and plain raw dumps.

YROMEM

Volatility Plugins

- Volatility is highly modular
- Easy to add new features/supports
- ~160 plugins for ~25 profiles
- Several plugins for malware analysis
- `python vol.py --info`

YROMEM

Bootstrap the Analysis

- Linux: /boot/System.map-$(uname -r)
- Windows:
  - Rekall:
    - Scan the memory to find RSDS signature
    - Extract GUID and PDB filename
    - Query the Microsoft public symbols server
    - From the PDB file extracts of many symbols
  - Volatility:
    - Scan the memory to find the KDBG to locate PsActiveProcessHead (Prone to Anti-forensics)
    - Drawback: Locate KDBG:
      - XP/Vista via KPCR
      - Win8 encoded

YROMEM

Processes

YROMEM

- `Pslist`: Walk the EPROCESS objects list
- `Pstree`: Like pslist but it prints out the tree
- `Psscan`: Scan the memory for the EPROCESS signature (find hidden and terminated processes as well)

Address Translation

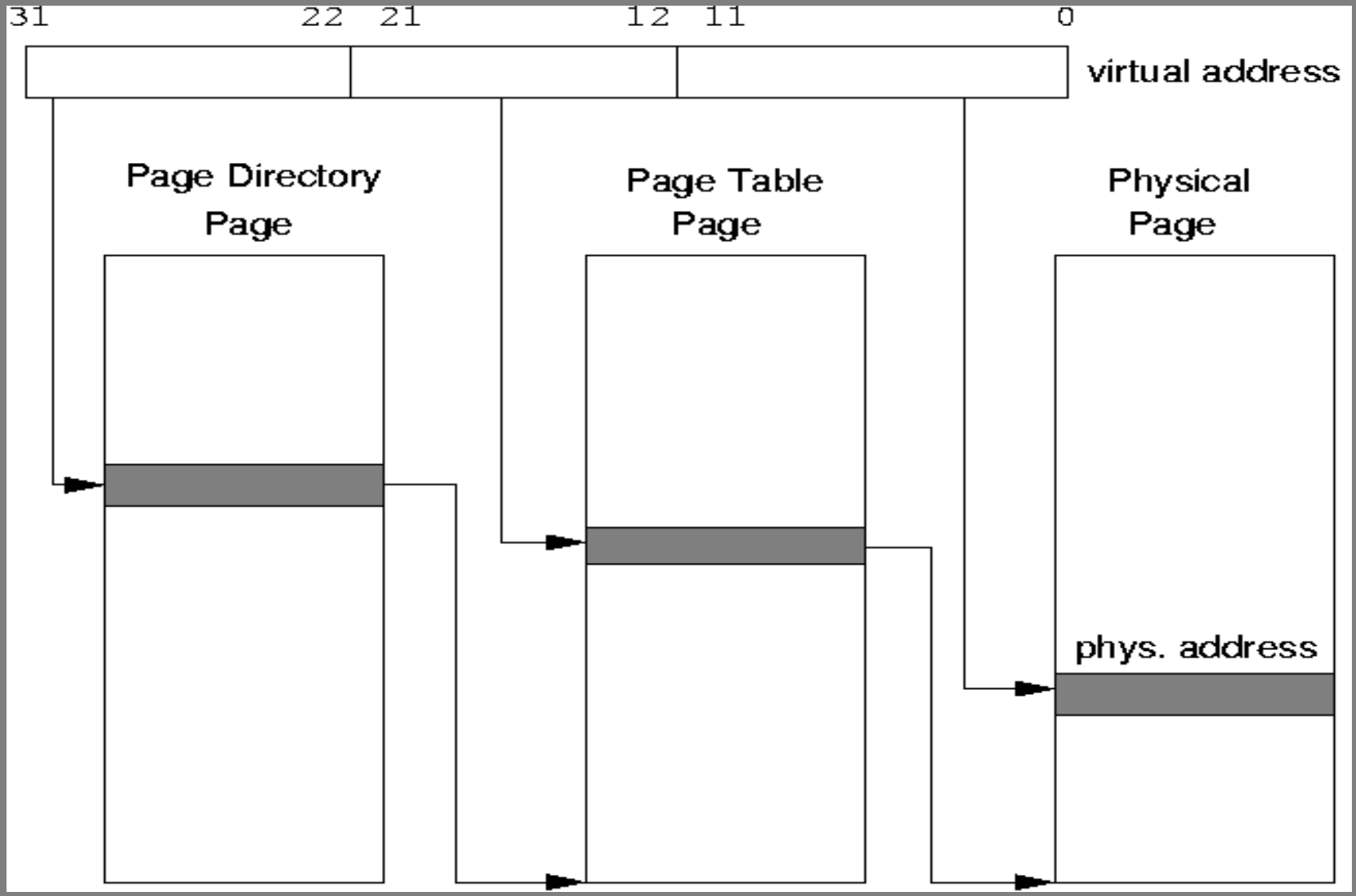
- Do you remember the Semantic Gap?
- All the pointers we have found are Virtual Addresses and we have a physical memory dump
- We need to emulate the MMU work
  - Volatility solution: Address Spaces

    (IA-32, IA-32 PAE, IA-32e, ARM, etc)

YROMEM

Address Translation IA-32

YROMEM



Outline

- Memory forensics
- Volatility
  - Windows
  - Linux
- Virtualization Support
  - Hypervisor Structures
  - Virtual Machines Analysis
- Future Work

YROOM

The problem

- Virtualization is everywhere
- No support to analyze:
  - Virtual Machines
  - Hypervisors
  - Nested configurations

YROOM

The solution

- Actaeon core:
  - VMCS layout extractor
  - HyperIs
  - Virtual Machine Introspection patch

YROOM

Warning

- Actaeon IS NOT:
  - A tool to dump the physical memory
  - A real time detector for malicious hypervisors
  - A malware detector

YROMEM

VMCS

- Virtual Machine Control Structure
- Intel VMX structure to handle VMX transistions
- Memory structure containing information for keeping the state of the system
- Fields listed in the Intel Manual but the layout is implementation specific

YROMEM

VMCS RE

YROMEM

- Simple reverse algorithm based on an Open Source hypervisor (HyperDbg):
  - VMCS fields are associated with a 32 bits value (encoding) that is used by VMREAD/VMWRITE instructions
  - The position is derived from the encoding in the processor microcode so we filled the VMCS region with 16 bit incremental numbers
  - We rebuilt the position of every field in the VMCS by associating the encoding value to the generated value

Hypervisor Discovery

YROOM

- Four heuristics on VMCS fields:
  - **REVISION_ID**: Determine the VMCS memory layout. Must match the value of MSR 0x480 (IA32 VMX_BASIC_MSR)
  - **VMX_ABORT_INDICATOR**: Must be zero. It is the second entry of the VMCS area.
  - **VMCS_LINK_POINTER**: Two consecutive words. They must be 0xFFFFFFFF
  - **HOST_CR4**: The 13th bit indicates if VMX support is enabled or not.

EPT

- Extended Page Tables
- Provide memory isolation among virtual machines
- Marked in a field in the VMCS (Secondary Based Execution Control)
- Provide an additional layer of translation (remember MMU?) transparent and in hardware
- Translation from a GPA to an HPA
- Translation has four stages (PML4, PDPT, PD, PT)

YROMEM

VMI

YROOM

- Virtual Machine Introspection via EPT
- Locate VMCS and extract the EPT pointer
- Simulate EPT translation
- Patch the Volatility core to add the EPT support

Outline

YROOMEM

- Memory forensics
- Volatility
  - Windows
  - Linux
- Virtualization Support
  - Hypervisor Structures
  - Virtual Machines Analysis
- Future Work

Actaeon

YROME

- Integration in Volatility
- x86-64 support
- Full Hyper-V support
- More testing for nested environments
- VMCS Shadowing support
- Find reliable solution to dump type-1 hypervisors

Memory Forensics

YROEM

- More research effort to enhance/ease malware analysis
- More communication among researchers
- Leverage memory forensics
- Lack of support for:
  - Net/Open/Free/BSD
  - Solaris/SPARC
  - Emulators (Qemu/Bochs/etc)
  - Containers (LXC/OpenVZ/Docker/etc)

Contact

YROME

- Mail: graziano <at> eurecom <dot> fr
- Twitter: @emd3l
- IRC: emdel/emd3l (Freenode/Efnet/W3challs)
- http://www.s3.eurecom.fr/tools/actaeon

We are looking for motivated and skilled Ph.D students. Feel free to contact me.