Breaking virtualization by any means

Jonathan Brossard
CEO – Toucan System

jonathan@
toucan-system.com
Who am I?

Security Research Engineer. Focus on low level bugs, RCE, code/binary auditing.

CEO of Toucan System (French Startup).

Previous research:
http://www.slideshare.net/endrazine

Getting in touch:
http://twitter.com/endrazine
Agenda

- Virtualization: big picture
- Attack surface analysis
- Shared Guest OS Isolation
- Attacking the host
- Privileges escalation
Virtualization: market shares

Source: Forrester Research 2009

78% of companies have production servers virtualized.

20% only have virtualized servers.
Virtualization : market shares

Source : Forrester Research 2009

VMWare is present in 98% of the companies. Microsoft virtualization products are used by 17%. Citrix/Xen is used by 10%.
In a nutshell...

- As widespread as Apache or Bind
- Proprietary software, very few builds
  (= reliable exploitation)
- You don't need a « remote » exploit : you buy a shell at the same hosting provider.
Definitions
Virtualization: Definitions

**Virtualization** is the name given to the simulation with higher level components, of lower level components.

**NOTE:** Virtualization of applications (as opposed to full Oses) is out of topic.
Virtualization: Definitions

Virtual Machine

A virtual machine (VM) is: "an efficient, isolated duplicate of a real machine".

Usage

- Cost reduction (shared hosting)
- Scalability (cloud computing)
- Run broken (old) applications
Attack surface analysis

Previous research
Privilege escalation on a guest

CVE-2009-2267 « Mishandled exception on page fault in VMware » Tavis Ormandy and Julien Tinnes
Privilege escalation on the host

VMware Tools HGFS Local Privilege Escalation Vulnerability

(http://labs.idefense.com/intelligence/vulnerabilities/display.php?id=712)
Attacking other guests

Vmare workstation guest isolation weaknesses (clipboard transfer)

http://www.securiteam.com/securitynews/5GP021FKKO.html
DoS (Host + Guests)

CVE-2007-4591 CVE-2007-4593 (bad ioctl crashing the Host+Guests)
Escape to host

Rafal Wojtczuk (Invisible things, BHUS 2008)

IDEFENSE VMware Workstation Shared Folders Directory Traversal Vulnerability (CVE-2007-1744)
Time for action
Shared Guest OS Isolation
Rebooting an alternate operating system

- Overwrite the MBR directly with autonomous offensive code
- Instrument the MBR

Optionally:
- Break boot passwords
- Attack disk encryption
- (Bootkiting, backdooring...)
Boot sequence overview

- Power supply initialize the clock
- Sends #POWERGOOD signal on bus
- CPU #RESET LINE
- POST Checks Performed with interrupts disabled
- IVT initialized
BIOS internals for keyboard management

Keyboard
- keyboard embedded 8042 PIC

Motherboard's 8259 PIC
- Up and down scan codes
- Unified key scan codes

Bios Interruption Service Routine
- read ASCII + scancode and update pointers
- function ax=0x00 or 0x01
- Int 0x16

Bios Data Area
- 0x40:0x1a: pointer to the next character in buffer
- 0x40:0x1c: pointer to last character in buffer
- 0x40:0x1e: BIOS keyboard buffer

Keyboard Interrupt Service Routine (int 0x9, IRQ 0x01)
- store ASCII + scancode in buffer and update pointers
Bruteforcing Passwords

I/O Port 0x60

ASCII code + scancode

function ax=0x00 or 0x01
int 0x16

Bootloader

I/O Port 0x64

read ASCII + scancode and update pointers

Bios Interruption Service Routine

Bios Data Area

0x40:0x1a : pointer to the next character in buffer
0x40:0x1c : pointer to last character in buffer
0x40:0x1e : Bios keyboard buffer P4ssw0rd

Motherboard's 8259 PIC

Unified key scan codes

Keyboard Interrupt Service Routine (int 0x09, IRQ 0x01)

store ASCII + scancode in buffer and update pointers
Attacking the hypervisor or host OS
Attacking the hypervisor or host OS

- VM 86 fuzzing
- ioports fuzzing
- pci fuzzing
Switching to virtual 8086 mode

- Switch to VM 86 using:
  ```
  #define __NR_vm86old 113
  #define __NR_vm86 166
  ```

- Use old school 16b interrupts to fuzz the hardware
- Note: It's (kernel) emulated. Good news! We can use it with x64 too :)


example:

Mov ah, 0x42 ; read sector from drive
Mov ch, 0x01 ; Track
Mov cl, 0x02  ; Sector
Mov dh, 0x03 ; Head
Mov dl, 0x80 ; Drive (here first HD)
Mov bx, offset buff ; es:bx is destination

Int 0x13        ; hard disk operation
Vm86 fuzzing under x64

Event Viewer

Summary page events  Number of events: 1

- Level: Error
- Date and Time: 26/06/2010 22:30:00
- Source: Hyper-V-VMMS
- Event ID: 14070
- Task Category: None

Event 14070, Hyper-V-VMMS

Virtual machine 'Ubuntu-fuzzing' (ID=CO79CB35-0249-49DE-8A5D-1F8FA50D7D37) has quit unexpectedly.

Log Name: Microsoft-Windows-Hyper-V-VMMS/Admin
Source: Hyper-V-VMMS
Logged: 26/06/2010 22:30:00
Event ID: 14070
Task Category: None
Level: Error
User: SYSTEM
Computer: WIN-M5M10P60MND
Switching to virtual 8086 mode

Limitation: Hardware unknown at BIOS Post time can't be fuzzed this way.

=> We need complementary techniques to be exhaustive.
Other techniques

- PCI fuzzing (fuzzing hot plug devices)
- Ioports fuzzing : interact with any hardware.
Ioports fuzzing:

Ioports:
outb, outw, outl, outsb, outsw, outsl,
inb, inw, inl, insb, insw, insl, outb_p,
outw_p, outl_p, inb_p, inw_p, inl_p

Problems: sequence, multiple ports ...
PCI Fuzzing

- In 16b mode: use int 0x1a
- In 32 or 64b mode: fork from pciutils :(
Escalating privileges on the host
Privilege escalation

- attacking (suid) hypervisors
- attacking kernel modules with ioctls
Thank you for coming

Questions ?