UNIVERSITY OF ILLINOIS

AT URBANA-CHAMPAIGN

WinWizard

Expanding Xen with a LibVMI Intrusion Detection Tool

illinois.edu

Main Points

- Virtual machine introspection (VMI) is key in the future of cloud security
- VMI tools, simple and easy
- VMI tools in industry

Outline

- Overview
- Xen/LibVMI
- WinWizard
- Experiment
- Performance Evaluation
- Future Work

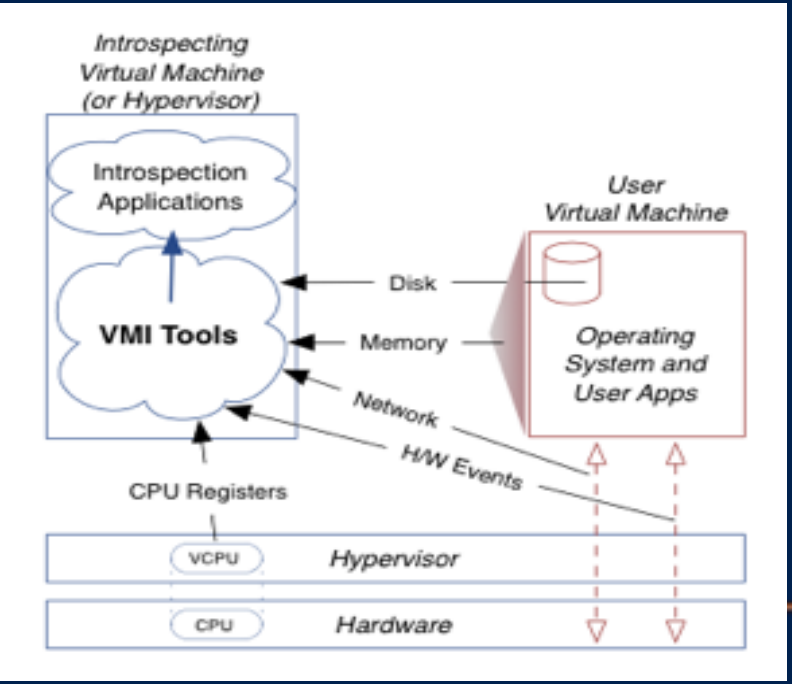
Overview

- Virtualized based security is driven by the threat of malware at the kernel level

- Enhance the security outside the machine through the use of introspection of guest machines

Xen/LibVMI

- Xen is a class of virtual machines managers which have direct access to hardware
- Allows operating systems to share conventional hardware in a safe and resource managed fashion
- LibVMI is a project aimed to provide tools to enable virtual machine introspection



WinWizard

- WinWizard is a group of two components
  - Operating system level view of currently running processes
  - Hypervisor level utilizes LIBVMI to give an hypervisor perspective of what is going on
- A comparison of both results determine whether or not something is wrong within the system

Hypervisor Level

- Hypervisor Level
  - Detect processes that have been removed from the PsActiveProcessHead list
  - Check for the manual byte-by-byte scan of physical memory, a search for the EPROCESS structure patterns

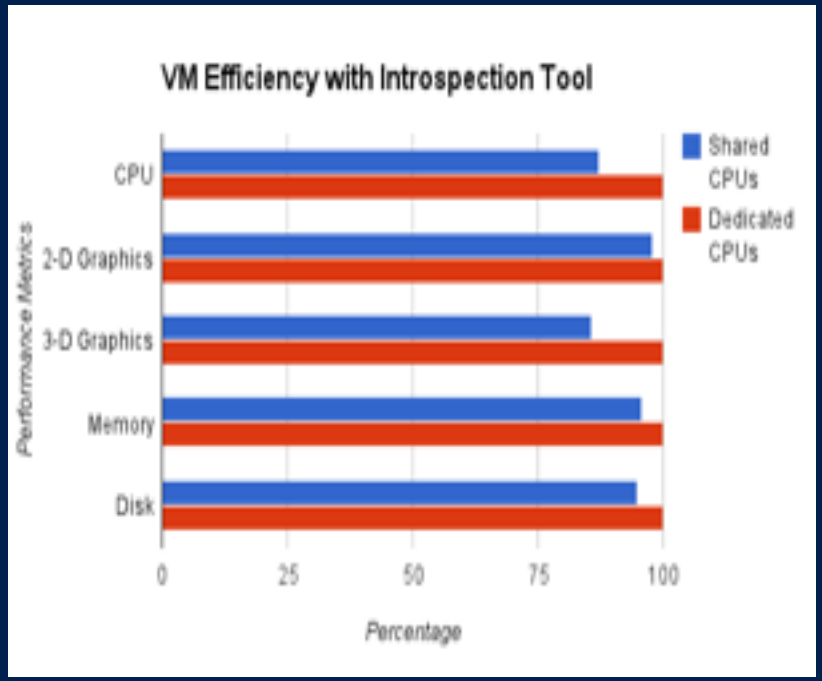
```
def isValidEprocessStruct(aByte):
    if aByte.Pcb.Header.Type != 0x03:
        return False
    if aByte.Pcb.Header.Size != 0x28:
        return False
    if aByte.ImageFileName is not valid:
        return False
    if aByte.UniqueProcessId is not valid:
        return False
    if aByte.ActiveThreads == 0:
        return False
    return True
```

The Experiment

- Two Experiments:
  - First experiment: Hacker Defender
    - Hacker Defender is a Windows operating system rootkit that allows processes, files and registry keys to be hidden from systems administration tools
  - Second experiment: Real World Simulation
    - One group was the hacker and our group were the defenders

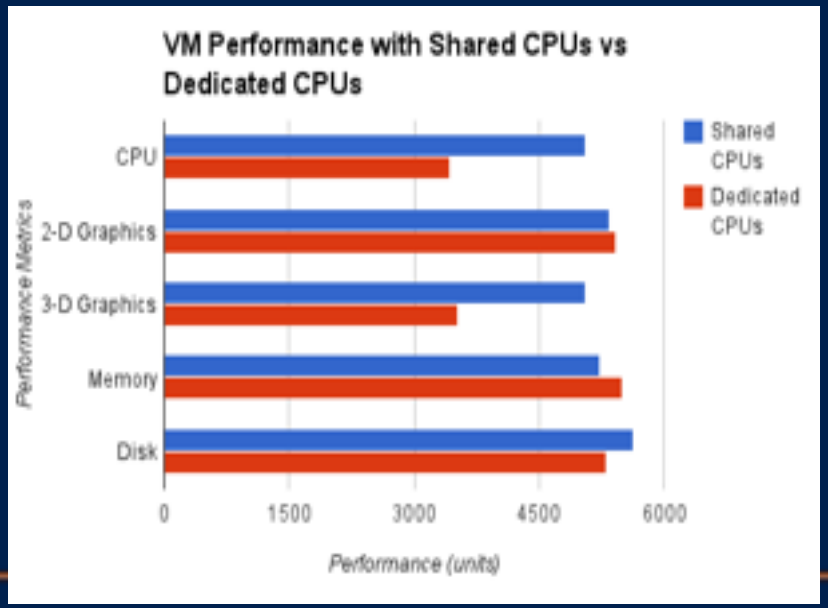
Performance Evaluation

- The tests consisted of running a VM without our introspection tool, sharing all 8 cores of the machine between the VM and dom0, and assigning the VM 4 virtual CPU's
- A performance drop of 13%



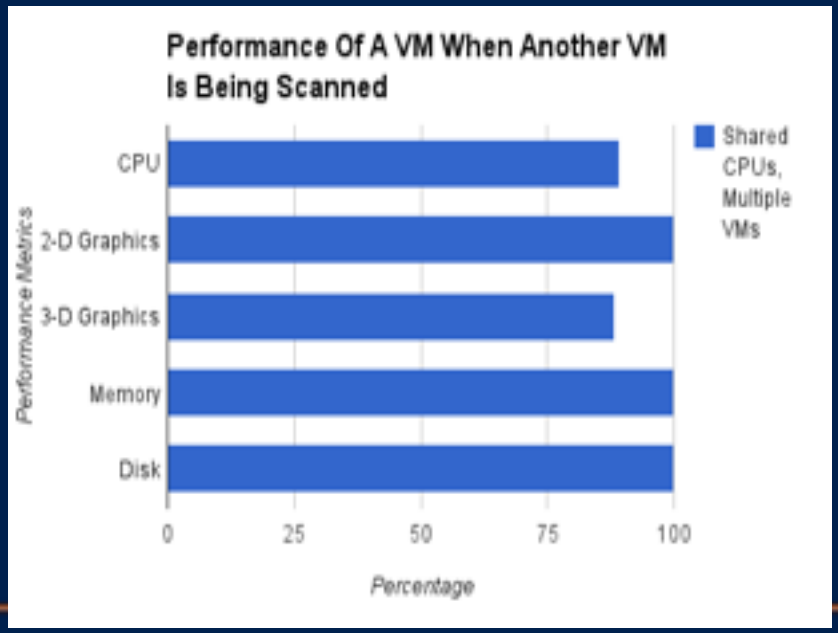
Performance Evaluation

- A dedicated core to dom0 to determine if any performance improvements
- Increasing the cores assigned to dom0 was that the introspection code was taking the majority of the execution time assigned to the dom0



Outline

- The impact on an additional guest not being scanned while a neighboring guest was having its memory scanned
- Drop of about 10% on the neighboring VM



Recap

- Virtual Machine introspection for cloud security
- VMI, tools, simple and easy
- VMI tools in industry

Future Work

- Full Set of Virtual Machine Introspection Tools
  - Signature detector
  - User program integrity detector
  - Memory access enforcer
  - NIC access enforcer
- Have both memory and network introspection tools, and use them in conjunction with each other in order to provide the best security all around