Improving the Security of Commodity Hypervisors for Cloud Computing

Anh Nguyen¹, Himanshu Raj, Shravan Rayanchu², Stefan Saroiu, and Alec Wolman

¹UIUC, ²U. of Wisconsin, and MSR
Today’s cloud computing hypervisors have very large trusted computing bases!
Hyper-V Architecture

Root VM (Dom0)

Child VM (Dom1)

Hypervisor
Hyper-V Architecture

- Root VM (Dom0)
- Child VM (Dom1)
- TCB
- Hypervisor
How Large is the TCB?

<table>
<thead>
<tr>
<th>Hypervisor</th>
<th>Lines of Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xen</td>
<td>250 KLOC</td>
</tr>
<tr>
<td>Hyper-V</td>
<td>100+ KLOC</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OS</th>
<th>Lines of Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux 2.6.32</td>
<td>11.2 MLOC</td>
</tr>
<tr>
<td>Windows Server 2008</td>
<td>50 MLOC</td>
</tr>
</tbody>
</table>

TCB of commodity hypervisors consists of tens of millions of lines of code!
Two Classes of Attacks

1. Attacks from guest VMs
   - Cases of malicious customer software already documented:
     - On Amazon EC2, customer sent spam & launch DoS

2. Physical attacks
   - Many already have access to “locked” datacenters
   - Providers are starting to outsource to 3rd-parties
     - Code offload servers deployed in coffee shops, mall areas
Two Classes of Attacks

1. Attacks from guest VMs
   - Cases of malicious customer software already documented:
     - On Amazon EC2, customer sent spam & launch DoS

2. Physical attacks
   - Many already have access to “locked” datacenters
   - Providers are starting to outsource to 3rd-parties
     - Code offload servers deployed in coffee shops, mall areas
Outline

- Motivation
- Requirements and design alternatives
- Design of Bunker-V
- Conclusions
Req#1: Hypervisors Must Accommodate Legacy OSes

<table>
<thead>
<tr>
<th>Year</th>
<th>OS Supported by Amazon EC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>RedHat</td>
</tr>
<tr>
<td>2008</td>
<td>Solaris, Oracle Linux, Win Server 2003</td>
</tr>
<tr>
<td>2009</td>
<td>Win Server 2008</td>
</tr>
<tr>
<td>Future</td>
<td>Fedora, openSUSE, Gentoo, Ubuntu, Debian</td>
</tr>
</tbody>
</table>

Future cloud computing goal: hosting home desktops
Req#2: High Performance

- Performance remains critical for cloud computing hypervisors:
  - Higher degree of multiplexing ➔ $$$

- The need for high performance is making the cloud push its computation closer to the edge
  - Cloudlets for code offload
Summary of Requirements

- Cloud hypervisors must:
  1. Be secure
  2. Accommodate legacy OSes
  3. Have high performance

- Next, we look at hypervisor alternatives
Alt#1: “Tiny” Hypervisors

- Recent project built a hypervisor in 7889 LoC!
  - Can run legacy OS!
  - Remove full-fledged OS from root VM

- Drawback: must compromise on functionality
  - Can’t run more than one VM at a time

Refs: SecVisor[SOSP’07], TrustVisor[Oakland’10]
Alt#2: Disaggregated Hypervisors

- Improves security:
  - Any exploit remains isolated in one compartment
- However, it does not reduce the size of TCB
  - We suspect TCB is even larger to include interface code among compartments

Refs: NOVA[Eurosys’10], S. Hand’s group[OASIS’04, VEE’08]
Our Approach: Bunker-V

- Bunker-V: reduce TCB’s attack surface by minimizing the interface between the TCB and guest VMs

- Even if vulnerability exists inside TCB, system remains secure as long as attacker cannot exploit it

- Re-think the design of a hypervisor for cloud scenario: eliminate unnecessary virtual devices
Outline

- Motivation
- Requirements and design alternatives
- Design of Bunker-V
- Conclusions
Virtual Devices: Interface btw. Root & Child VMs

Virtual Devices

Root VM (Dom0)

Child VM (Dom1)

Physical Devices

Hypervisor
Categories of Virtual Devices (vdev)

- **Extraneous vdevs**: not needed in the cloud
  - e.g., floppy, keyboard, mouse, monitor, serial port

- **Legacy vdevs**: not needed in the cloud, but OS cannot boot without them
  - e.g., keyboard controller, PIT, ISA bus

- **Required vdevs**: needed to run in the cloud
  - e.g., storage, NIC, PIC, PCI bus
Categories of Virtual Devices (vdev)

- **Extraneous vdevs:** not needed in the cloud
  - e.g., floppy, keyboard, mouse, monitor, serial port

- **Legacy vdevs:** not needed in the cloud, but OS cannot boot without them
  - e.g., keyboard controller, PIT, ISA bus

- **Required vdevs:** needed to run in the cloud
  - e.g., storage, NIC, PIC, PCI bus
Bunker-V’s Interface
**Challenge**: Handle Guest OS Booting

- **Approach**: delusional boot

- **Boot OS on a separate node**:
  - Separate node has legacy vdevs enabled
  - Separate node is isolated from datacenter
Conclusions

- Bunker-V improves security of hypervisors for cloud computing:
  - 79% reduction of TCB’s interfaces
  - Can run legacy OSes with high performance

- Delusional Boot: new technique for booting legacy OSes in the absence of many devices
Questions?

- ssaroiu@microsoft.com