Trusted Virtual Datacenter –
Radically simplified security management

Stefan Berger, Ramón Cáceres, Dimitrios Pendarakis,
Reiner Sailer, Ray Valdez

Secure Systems Department, T.J. Watson Research Center
Security Opportunity Prologue

- **Significant Challenges**
  - Status quo approach to IT and business security is too complex, is not measurable, will not scale
  - Lack of secure foundation for dynamic enterprise environments

- **Synergistic Strategy**
  - Leverage emerging trusted computing technologies (TCG) and commoditization of virtualization (Intel / AMD, EMC, Microsoft, IBM)
  - Near-term: stronger guarantees position security as an enabler
  - Longer-term: radically simplified IT security management
Trusted Computing and Virtualization Timeline

- IBM IMA for Linux
- MS NGSCB 1...
- IBM sHype
- IBM vTPM
- NAC
- MS Vista
- bitlocker

- TCG TPM1.1 SRTM
- TCG TPM1.2 DRTM
- AMD SVM SKINIT
- Intel LT SENTER
Virtualization Landscape at a Glance

- **Application-level (or middleware-level) virtualization**
  - E.g., Java Virtual Machine, Softricity (Microsoft SoftGrid), Thinstall

- **Operating system-level virtualization**
  - E.g., Linux VServers, Solaris Containers / Zones, Virtuozzo

- **Hypervisor-based virtualization**
  - Type 1: VMware ESX, Microsoft Viridian, Xen, PHYP, PR/SM
  - Type 2: VMware Workstation, Microsoft Virtual PC, KVM
Classic Type 1 Hypervisor

- Application
- Application
- Application
- Application
- Application
- Application
- Application
- Application
- Guest Kernel
- Guest Kernel
- Guest Kernel
- Hypervisor
- Hardware
Virtualization-based Security & Systems Management

Trusted Virtual Data Center (TVDc)

Market Analysis
Security Underwriting

Centralized IT Security management
- TVD: Grouping of VMs and resources that support common objective (customer workloads, etc.)
- Abstracting the physical infrastructure (platform independence, scalability)
- Policy-driven (consistent security configuration and management)

Distributed Enforcement
- Very strong, coarse-grain security guarantees – cannot be bypassed by VMs
- Single data center security policy across different platforms and hypervisors
- Containment (viruses, break-ins) & Trust

sailer@us.ibm.com
11/7/2007
sHype: Enabling Trusted Virtual Datacenters

**TVDc**
(manages)

- Workload Isolation + Integrity
- Radically Simplified WL-Management

**Human Resources**

**sHype**
 controIs sharing)

**Payroll**

---

**Xen VMM**
(virtualizes + isolates)

---

sailer@us.ibm.com

11/7/2007
Trusted Virtual Datacenter Simplifies Security Management

Systems View

Virtual Domain View

Red = Acct.  
Green = HR.  
Blue = Dev.

Guard-VM

Isolation

Integrity

Trust

TVDc
Trusted Virtual Data Center Value Proposition

IBM TVDc: Radically Simplified Security Management

- **Isolation Management**
- **Integrity Management**

Enforces restrictions on administration and data sharing:
- Who manages what: independent admin for Hertz and Avis accounts
- What can run together: ensure air-gaps between strongly competing workloads
- Workload and data isolation (malware confinement)

Maintains software inventory and acts as an early warning system for anomalies; detect and report:
- What is running in each VM
- If VMs/Systems are correctly configured
- If VMs are up-to-date with patches

→ TVDc reduces the risk of security exposures
→ TVDc enables consistent, policy-driven enforcement
Secure Hypervisor Architecture (sHype)

VM

Application
Application
Application
Application
Secure Services

Linux
MS Windows

Xen / sHype

Hardware

Auditing, Monitoring, Metering, …
Secure (isolated) services
e.g. Policy Management
Resource control and metering
Access control between partitions
Isolation between partitions
Attested boot and run-time
(TCG/TPM, IMA)

Sailer, Jaeger, Valdez, Cáceres, Perez, Berger, Griffin, van Doorn:
sHype Access Control Architecture (Example: Xen)

- Flexible framework: Supports Multiple Policies
- Access Control Module Implements Policy Model
- Hypervisor Security Hooks
  - mediate inter-VM communication + resource access
  - interact with ACM for access decision
- Implemented for Xen, PHYP, rHype in various stages
1. Centralized Isolation Management

- Policy authoring and management
  - Define security labels and anti-collocation rules
  - Revision-based policy management

- Labeling Systems, VMs and resources

- Label-based management
  - Restrict Admins to manage a set of security labels
  - Restrict configuration choices based on policy

= Accounting
= Human Resources
= Development

Anti-Collocation:{○,●},...
2. Distributed Isolation Enforcement at Run-time (Secure hypervisor extensions sHype/ACM)

1. Control Sharing

2. Control what a system can run

3. Enforce rules for anti-collocation

- Xen: Integrated into Open-source distribution
- PHYP Access Control Module (research prototype)
TVDc Network Isolation

1. Label VMs + VLANs

2. VMM enforces:
   VMs ↔ VLANs

3. Hardware VLAN switch enforces:
   Blades ↔ VLANs

VM1 VM2 VM3
VM4 VM5

Blade 1
Blade 2

Virtual LAN 1
Virtual LAN 2

Virtual LAN 1
Virtual LAN 2

Network Switch
Trusted Virtual Domains – Isolation and Trust

Attestation: mutually verifiable environments

Authentication: systems and workloads

Isolation: protect against attacks and limit spread of damage

Mediated Communications: transparent protection, authorization and audit
Distributed Trusted Computing Base
Putting Access Control and Integrity Measurement together

- Establish trust – enabling collaboration across multiple platforms
  - Are P1 and P2 mutually trusted (TCB)
  - Are policies A and B compatible?
  - Are policies uniformly enforceable?

Platform P1
vm1 vm2 vm3

Platform P2

McCune, Berger, Cáceres, Jaeger, Sailer:
Shamon – A System for Distributed Mandatory Access Control. 22nd ACSAC, 2006.
Trusted Platform Module (TPM)

- Trusted Computing in today’s world is largely synonymous with a use that involves the Trusted Platform Module (TPM)
- TPM is a passive storage device that has some interesting properties:
  - You cannot remove data once you’ve written it to the TPM
  - You can retrieve an aggregate of the data from the TPM that is signed by that TPM’s unique key
  - The TPM provides sealed storage
  - Storage root key protection
Integrity Measurement – Integrity & Attestation

- Provide reliable runtime integrity guarantees
  - Certificates provide identity and secure tunnel
  - But does the remote system currently satisfy security-related requirements?

- Leverage Trusted Platform Module (TPM) / Core Root of Trust for Measurement
  - Remotely attest software-stack
  - Detect cheating & compromise (load guarantees)
  - Bind sensitive data to endpoint (certificates etc.)
  - Non-intrusive / negligible overhead

- Implemented for Linux in 2003/2004
  - IBM Integrity Measurement Architecture (IMA)

Trusted Computing uses real-time attestation to establish sufficient facts about a system, such as software integrity, to interpolate from its past to its future behavior.

1. Local integrity verification
   - Does *my* system have integrity?
   - Is it save to log in and use? (Kiosk, Desktop, …)

2. Remote integrity verification
   - Does *their* system have integrity?
   - Is it save to use? (online services, …)
   - What about its users?
Integrity Measurement Architecture (IMA)

Attesting System

Verifying System

Measurements

SHA1(Boot Process)
SHA1(Kernel)
SHA1(Kernel Modules)
SHA1(Program)
SHA1(Libraries)
SHA1(Configurations)
SHA1(Structured data)
...

TPM-Signed PCR Integrity Value

Analysis

Deduce System Properties

Known Fingerprints

(1) Measurement
(2) Attestation
(3) Verification

sailer@us.ibm.com

11/7/2007
Virtual TPMs Enable VM Integrity Attestation

Measure HW, hypervisor, and critical services

Support current IMA via vTPMs (flexible, scalable)

<table>
<thead>
<tr>
<th>Stat.</th>
<th>Object</th>
<th>Measurement</th>
</tr>
</thead>
<tbody>
<tr>
<td>boot_aggregate</td>
<td>c488928f1dcbb23c14d697749fedd46181dcd2746f6</td>
<td></td>
</tr>
<tr>
<td>/bin/bash</td>
<td>6797b4a1f00b2658ac07e7d0f792515193d2</td>
<td></td>
</tr>
<tr>
<td>/lib/id-2.4.so</td>
<td>c86f8b23ebf29cf6f8e4e33b563e92e9146</td>
<td></td>
</tr>
<tr>
<td>/lib/libreadline.so.5.1</td>
<td>327724d12c843164754b04a5188dc2e04977055dd</td>
<td></td>
</tr>
<tr>
<td>/lib/libhistory.so.5.1</td>
<td>1e93017265d2a38b2f954683228c147bc17191</td>
<td></td>
</tr>
<tr>
<td>/lib/libnssการบริหาร .so.5.5</td>
<td>901ed86f6eb19094d2116b52a67119d33c9b8</td>
<td></td>
</tr>
<tr>
<td>/lib/libdl-2.4.so</td>
<td>96c0f0b6b26a7f6564e9d5f7153d7b2db416</td>
<td></td>
</tr>
<tr>
<td>/lib/libc-2.4.so</td>
<td>54945fdad59340abe673e0c3c651219a851ae4c4</td>
<td></td>
</tr>
<tr>
<td>/lib/libnss_files-2.4.so</td>
<td>e5ca37f087913b8ca48237f6949b551d54186</td>
<td></td>
</tr>
<tr>
<td>/bin/mount</td>
<td>de66f2435b05f4edc1c039d6ed33eae4c6f2</td>
<td></td>
</tr>
<tr>
<td>/bin/mknod</td>
<td>689398fa34867980377b2404956d45b44e0d6454</td>
<td></td>
</tr>
<tr>
<td>/bin/mkdir</td>
<td>6e461db4b7c6b75630be26bc397a365ef705ac611d</td>
<td></td>
</tr>
<tr>
<td>/bin/fn</td>
<td>76745b9bc6a903e6a377f7c85e85340b0eb539f2c75</td>
<td></td>
</tr>
<tr>
<td>/bin/cat</td>
<td>b2a70064651bd72ab65472096d8d65bf74b66</td>
<td></td>
</tr>
<tr>
<td>/bin/sed</td>
<td>1d62b4e75a12fb67418d7428f59eac7afab6e1d4</td>
<td></td>
</tr>
<tr>
<td>/sbin/udevd</td>
<td>c76202f190f9ed83ad94a6e0d9054686c7a0c</td>
<td></td>
</tr>
<tr>
<td>/sbin/udevttrigger</td>
<td>8ef15bcd38ec64f19e9a6463d32948be3f209d</td>
<td></td>
</tr>
<tr>
<td>/sbin/udevsettle</td>
<td>3473a0d3a8a81838b29730d0a4b4854b6567</td>
<td></td>
</tr>
<tr>
<td>/sbin/udev/create_floppy_devices</td>
<td>d5847c9c710d4afcf2302b43495c8ab82393201</td>
<td></td>
</tr>
<tr>
<td>/sbin/modprobe</td>
<td>b406cb6d82f0a49a24edf419856ac33457b75</td>
<td></td>
</tr>
<tr>
<td>ide_core</td>
<td>9ec82165f8a7d8b13b26d4664a233118be77e8</td>
<td></td>
</tr>
<tr>
<td>ide_disk</td>
<td>63e27880ba9d38b638a07e70b89dbba641207d5e</td>
<td></td>
</tr>
</tbody>
</table>
vTPM+IMA: Focus on Solving Real Problems

- **Configuration Management**
  - Configure server classes
  - Verify configuration against software stack

- **Problem Management**
  - Automatically detect and isolate real problems
  - Direct intelligence towards those real problems
  - Fix problems efficiently
  - Verify that problems no longer exist
Research Challenges around TVDc Technologies

- Controlled Sharing Between TVDc
  - Guard systems

- Integrity Measurement Architecture
  - Run-time guarantees (extend load-time guarantees)
  - Property determination and fingerprint management

- Distributed Mandatory Access Control
  - Policy composition & change management

- Virtual TPM
  - Safely migrate/save/restore the virtual root of trust
Trusted Virtual Data Center

- **TVDC is designed to achieve**
  - simplified security management
  - enterprise-level assurance

- TVDC creates confined workload domains to enable
  - independent trust and security properties

More on our department team page:

http://www.research.ibm.com/secure_systems_department

or:

Resources – TVDc building blocks freely available:

- **Integrity Measurement Architecture (IMA)**
  - Source code:  http://sourceforge.net/projects/linux-ima

- **Virtual Trusted Platform Module (vTPM)**
  - Source code in Xen:  http://www.xensource.com/xen
  - Project page:  http://www.research.ibm.com/ssd_vtpm

- **sHype Access Control Architecture**
  - Source code in Xen:  http://www.xensource.com/xen
  - Project page:  http://www.research.ibm.com/ssd_shype