Intel’s Xen Security Update

Joseph Cihula
Intel Corp.

Xen Summit
January 17-18, 2006
Current Work

- TPM virtualization (vTPM)
- Domain0 disaggregation (security domains)
- Measureable domain building
TPM virtualization (vTPM)

• **Basic support in Xen 3.0:**
  - vTPM manager
  - TPM device models and emulator
  - Off (not built) by default

• **Next steps:**
  - On (built) by default – patch sent
  - Sealing of per-virtual-TPM state/context – patch sent
  - Migration support
Domain0 Disagregation (Security Domains)

• We need a place to run security-sensitive code
  ▪ vTPM manager, domain measurement agent, TPM driver, etc.
  ▪ Should be small, trustable, and meaningfully measureable
    □ Would like it to (eventually) be a lightweight kernel
    □ Should not change frequently
  ▪ Should be extensible to finer-grained decomposition

• Start with a XenoLinux kernel
  ▪ “Domain-S0”, DOMID_DOMS0 (32751)
  ▪ -xenU kernel w/ most drivers removed
  ▪ No HW access (except for TPM)
  ▪ Privileged (SIF_PRIVILEGED | SIF_INITDOMAIN)
  ▪ Console to/from serial emergency console
  ▪ Runs completely from initrd
Boot Process w/ Security Domain

1. **GRUB loader**
   - Loads xen, DomS0, DomS0’s initrd, Dom0, Dom0’s initrd
   - Launches xen

2. **Launch DomS0**
   - Xen constructs and launches same as for Dom0 (do_createDomain(), construct_dom0())

3. **DomS0 initializes**
   - Launch TPM driver, TPM BE, xenstore, vTPM manager, measurement agent
   - Dom0 builder populates xenstore
     - Binds TPM DomS0 BE to Dom0 FE

4. **Launch Dom0**
   - Dom0 builder builds Dom0 from loaded images
     - Using libxenctrl fns
   - Launches Dom0

5. **Dom0 xend runs**
   - Migrates state from DomS0 xenstore to Dom0 xenstore
Security Domains Status

• Basic functionality is working:
  - DomS0 runs and launches Dom0 (runlevel 3)
  - vTPM FE in Dom0 talks to vTPM BE/manager in DomS0
  - xend runs and can create DomU’s

• Next steps:
  - TPM driver in DomS0
  - Make Dom0 reliable at runlevel 5
  - Finish DomU support
    - Full driver support (vTPM and network)
    - Clean shutdown
  - Verify on 64bit system
  - Test and tune
Measureable Domain Building

• **Want a measurement of initial domain state**
  - Kernel and initrd
  - Needs to be instance-independent
  - Used for measuring all domains (incl. Dom0)

• **Support de-privileging Dom0**
  - Domain building currently requires ability to map foreign pages
    - Remove this requirement by splitting domain build process
  - But don’t force domain building tools out of Dom0
    - And still permit domain layout flexibility
  - Will permit unprivileged domains creating domains

• **Support rich measurement capability**
  - Manifests, code/data/initrd separation, etc.
  - Too complex and slow to do in hypervisor
Measureable Domain Building Process

• Today:
  ▪ Dom0 tools:
    ▫ Read and parse domain config
    ▫ Create domain stub (allocation of domid, task struct)
    ▫ Allocate memory and load kernel and initrd images
    ▫ Create: phys-machine mappings, initial page table, start_info page, shared page
    ▫ Launch domain

• New:
  ▪ Dom0 tools:
    ▫ Create domain stub
    ▫ Allocate memory and load images
    ▫ Describe domain structure (code, data, etc.)
  ▪ Domain finalizer:
    ▫ Compute hash
    ▫ Create: phys-machine mappings, initial page table
    ▫ Fill-in: start_info page, shared page
    ▫ Launch domain
Measureable Domain Building Status

• **Basic functionality is working:**
  - xend modified to use new libxenctrl build fn
  - Domain finalizer is running as kernel module
  - Builds and launches x86_32 Linux domains

• **Next steps:**
  - Support revocation of Dom0 foreign mappings
    - Grant tables, etc.
  - Integrate finalizer into security domain
  - Support other domain types (EM64T, PAE, VT, BSD, etc.)
Architectural Overview

DomS0
- vTPM Mgr.
- TPM device models
- dom finalizer

XenoLinux
- TPM driver
- TPM BE
- xenbus

Xen Hypervisor
- domain creation

Dom0
- libxenctrl domain builder
- xenstore

XenoLinux
- xend
- xenbus
- TPM FE

DomN
- XenoLinux
- xenbus
- TPM FE

TPM

[Diagram showing the architectural overview of XenoLinux with connections between components like dom finalizer, vTPM Mgr., Xen Hypervisor, and xend.]
Future Plans

• I/O virtualization (VT-d)

• Hardware virtualization enhancements

• LT support