Introduction to Qubes OS

bhyvecon Tokyo 2014

@ntddk

Self-introduction

- Yuma Kurogome(@ntddk)
- Takeda Lab @ KEIO Univ.
- Researching about security in low-layer
- Participant of Security Camp '11, '13
- CTF player @ EpsilonDelta

What is Qubes OS?

What is Qubes OS?

- Secure VM developing by Invisible Things Lab
- Security by Isolation
- Open Source(GPL v2)

- Based on Xen
  - So today I don't speak about bhyve
  - Wish I could supply some inspiration for you!

Invisible Things Lab

Invisible Things Lab

- Founded by Joanna Rutkowska in 2007
  - Who forced Citrix to publish souces of XenClient
  - Published Blue Pill[SyScan'06] when she were in COSEINC
- Blue Pill
  - VT based rootkit(hypervisor)
- Previous rootkit were on Ring 0
  - Hooking System Call
  - Altering Kernel Structure
  - So we can detect it

Invisible Things Lab

- VT based rootkit were on Ring -1
  - So we can hardly detect it ***after infection***
  - For now, VT based rootkit is not serious threat

Invisible Things Lab

- They had been researched about
  - rootkit
  - SMM(System Management Mode)
  - Intel TXT(Trusted Execution Technology)

- Now they are developing Secure VM focused on mechanism of **Xen**

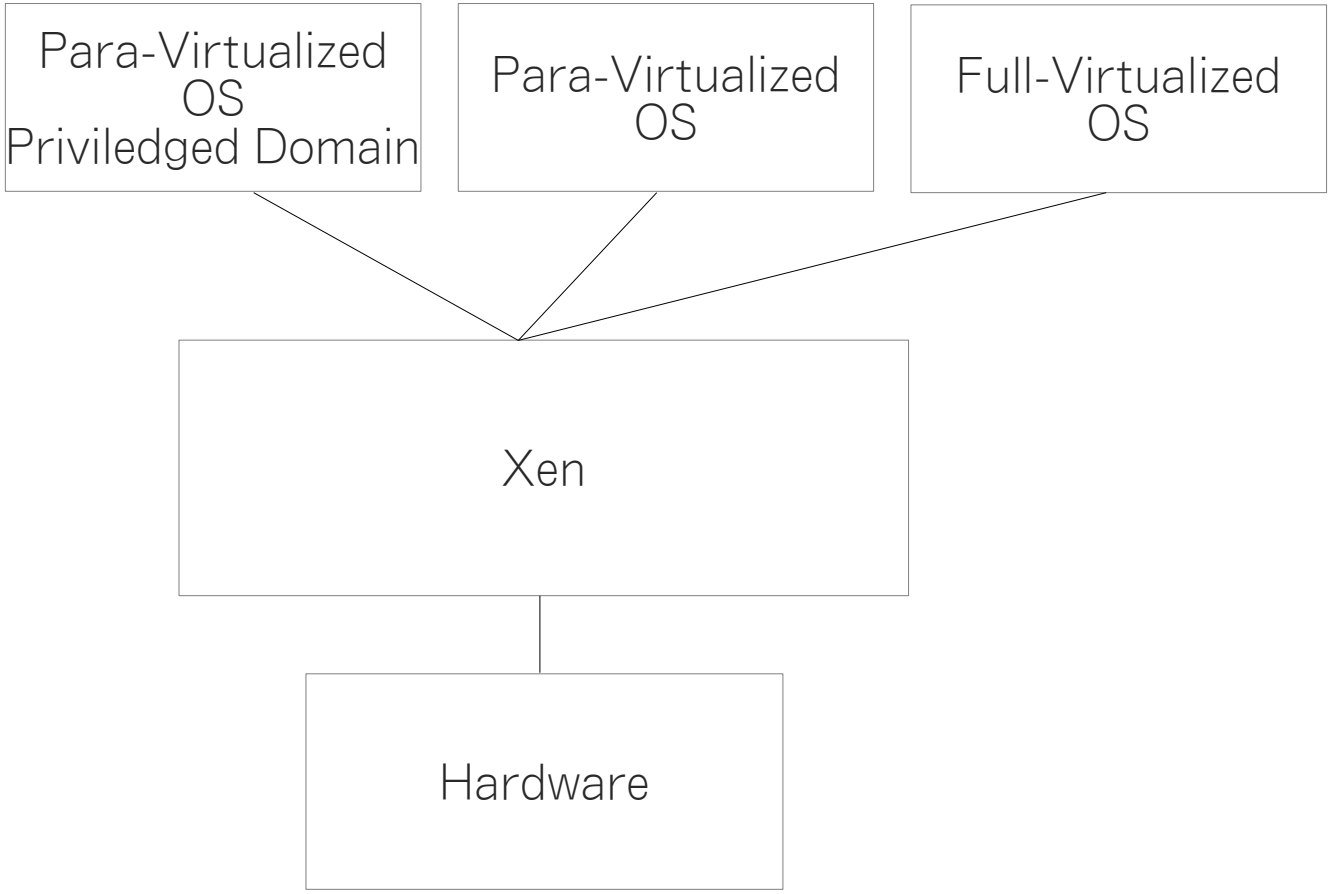
Well...
What's the difference between Xen and KVM?

Review: difference betwen Xen and KVM

- Virtualization methods
- Intrrupt
- Memory mapping

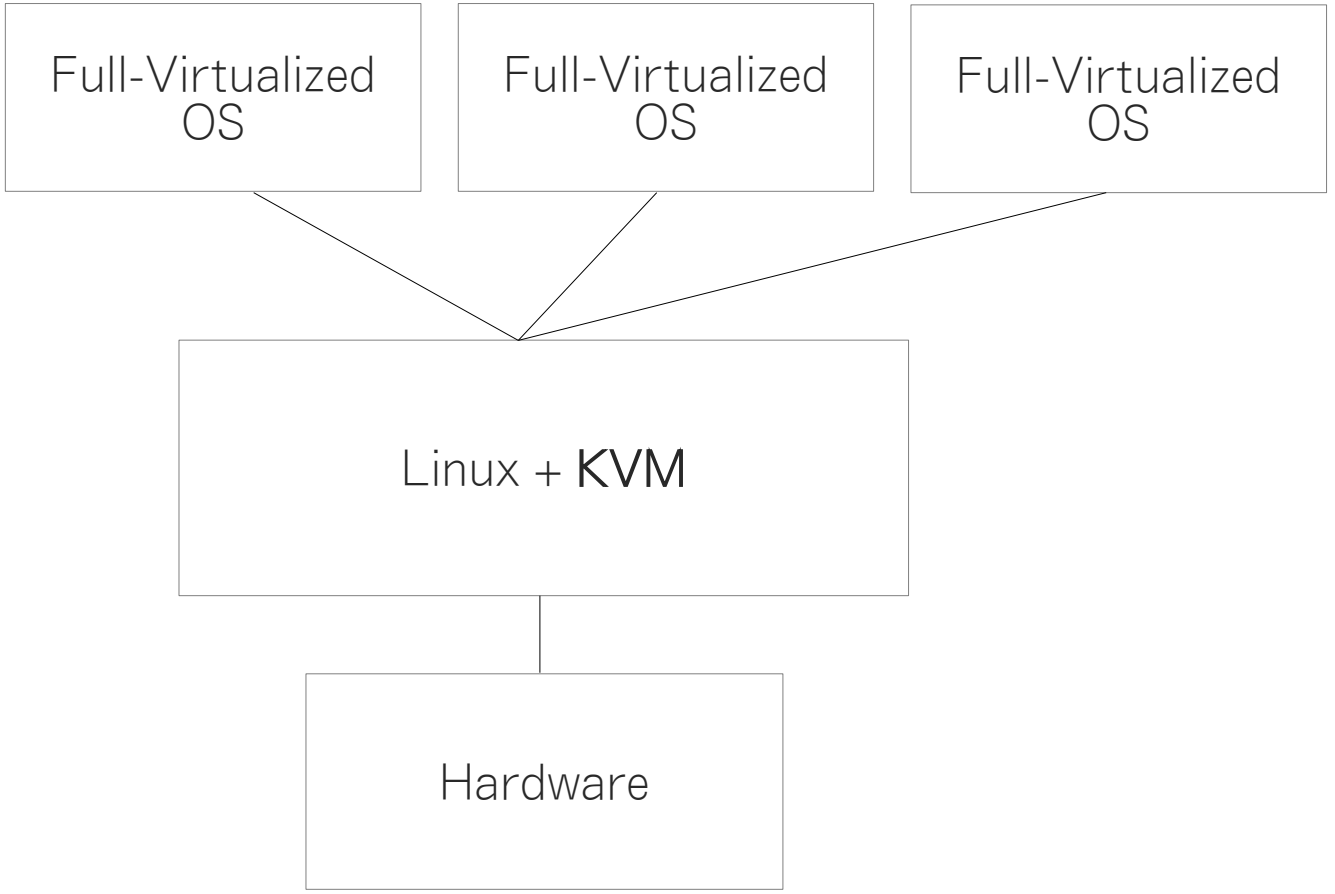
Review: difference betwen Xen and KVM

- Xen
  - Para-Virtualization
  - Full-Virtualization by Intel VT



Review: difference betwen Xen and KVM

- KVM
  - Full-Virtualization
  - Para-Virtualization by virtio



Review: difference betwen Xen and KVM

- Virtualization methods
  - Para-Virtualization
    - Modify OS for virtualized environment
    - No need of full hardware emulation
  - Full-Virtualization
    - No need of modifying OS
- Inturrupt
  - Xen uses event channnel
  - KVM uses MSI(-X)

Review: difference betwen Xen and KVM

- Memory mapping
  - KVM

    Gest-Physical memory space is part of host-virtual memory space of QEMU
  - Xen

    Mapping Gest-Physical memory space On demand

- Both use HW-assisted virtualization
  - **Intel VT**, AMD-V

Well... What is Intel VT?

Review: Intel VT

- Handling sensitive instructions
  - How to emulate it?
  - Tired to rewriting instrctuions by hand

Review: Intel VT(VMX)

1. Load some settings to VMCS
2. Set CPU to VMCS
3. VMLAUNCH → VMEntry, Enter VMX non-root mode(Guest mode)
4. Execute guest environment
5. Cause of trap → VMExit, Enter VMX root mode
6. Check VMExit reasons, emulation
7. VMRESUME → VMEntry, Enter VMX non-root mode → 4

Review: Intel VT(VMX)

- What is VMCS?
  - Virtual Machine Control Structure
    - Program Counter
    - Register
    - VM
    - What to trap

Review: Intel VT(EPT)

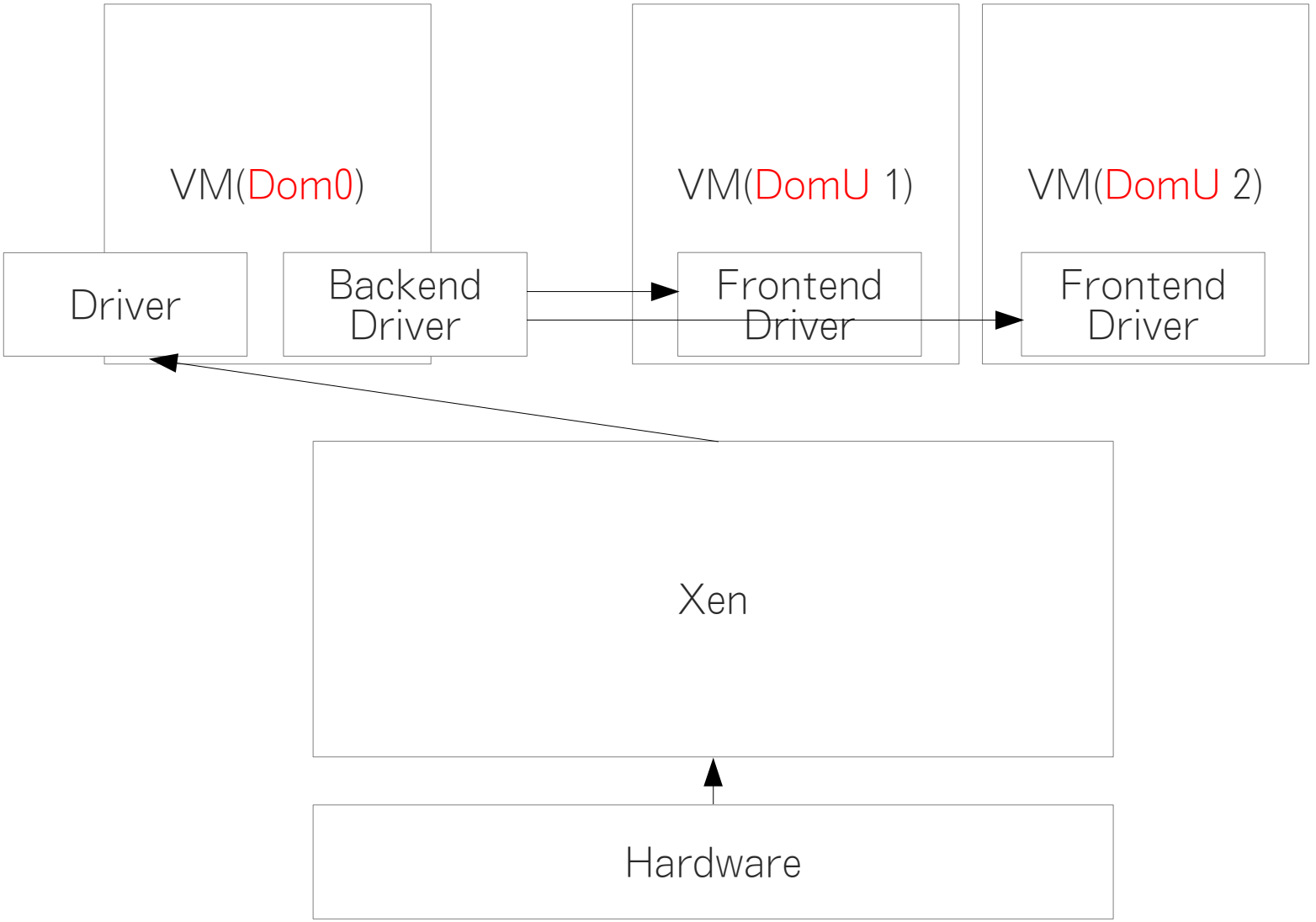
- Simplifying Paging
  - Tired to twice translation
  - Shadow Page Table
- EPT
  - Extended Page Table
  - Address translation by HW
  - Reduction of Overhead

Review: Intel VT(EPT)

- We can easily make VMM using VT! → KVM
- Xen...
  - Need of HyperCall
  - Full-Virtualization by VT

Xen Virtualization

- Xen has a Dom0(host) and some DomU(guest)



Xen Virtualization

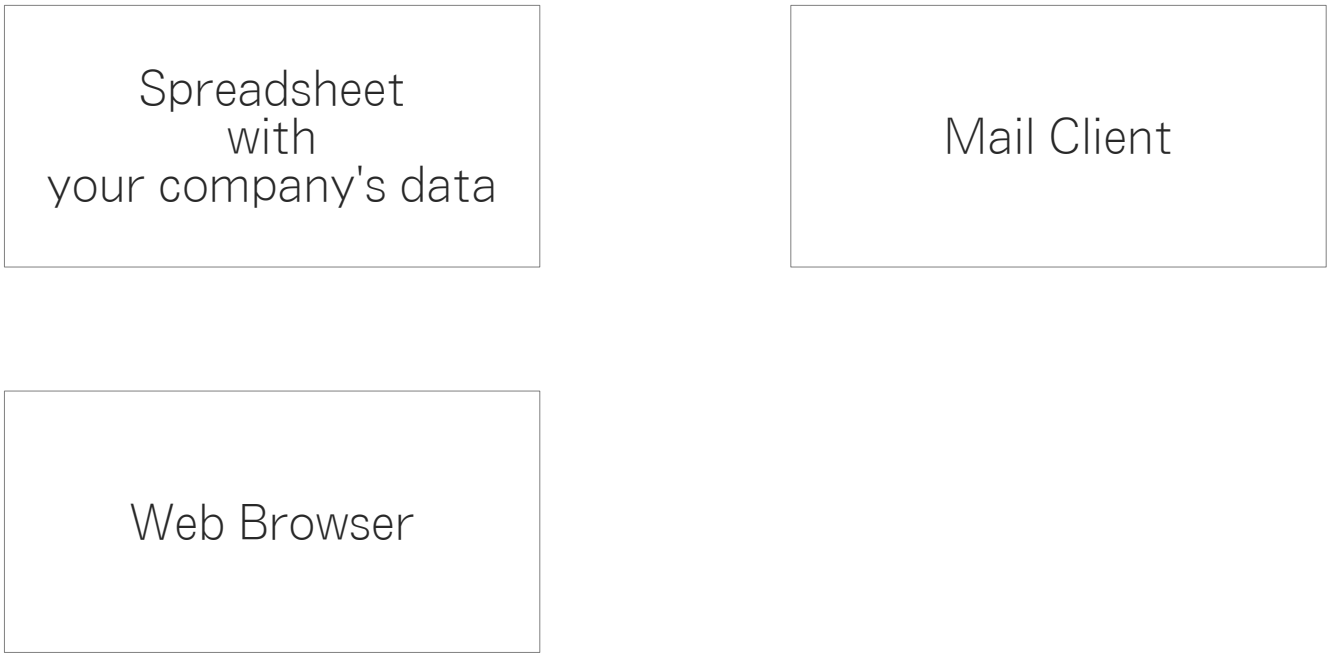
- Xen hypervisor execute Dom0 before DomU
- Dom0 manages other DomU
  - Only Privilege Domain is allowed to access all HW
  - DomU ask Dom0 to HW access via Backend/Frontend Driver

- Qubes OS apply this architecture to security

Concept of Qubes OS

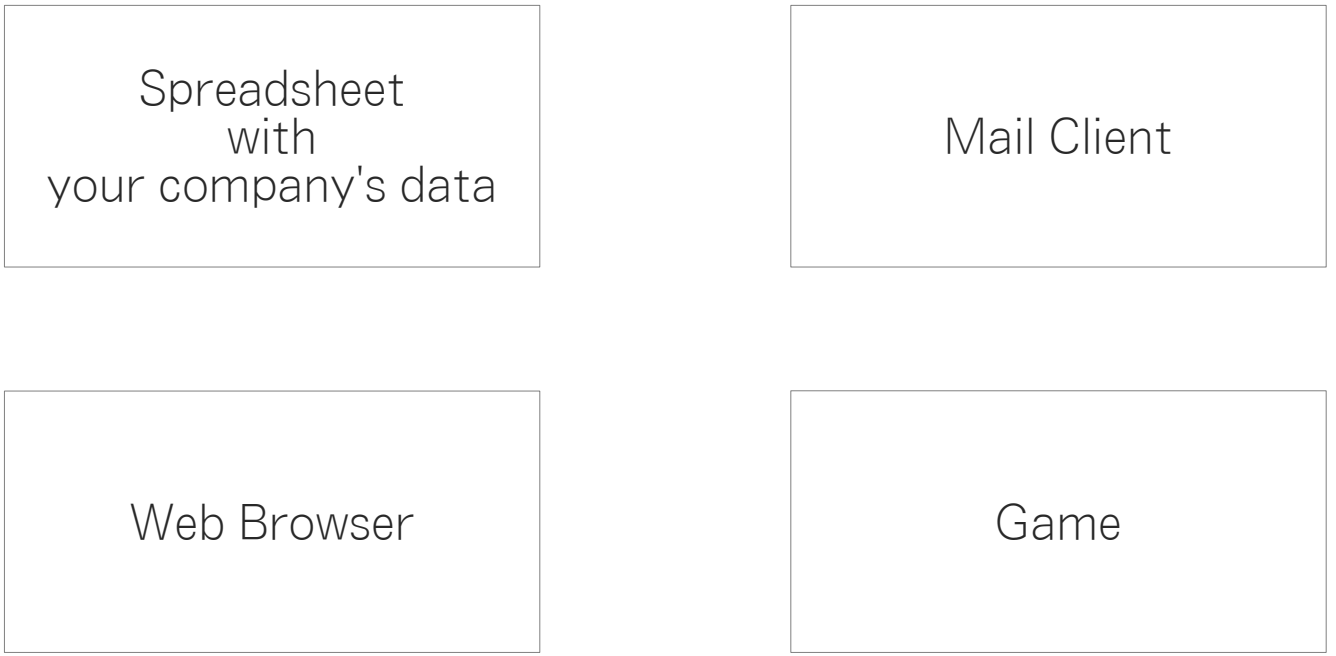
Desktop Environment

- Qubes OS want to provide strong security to desktop environment



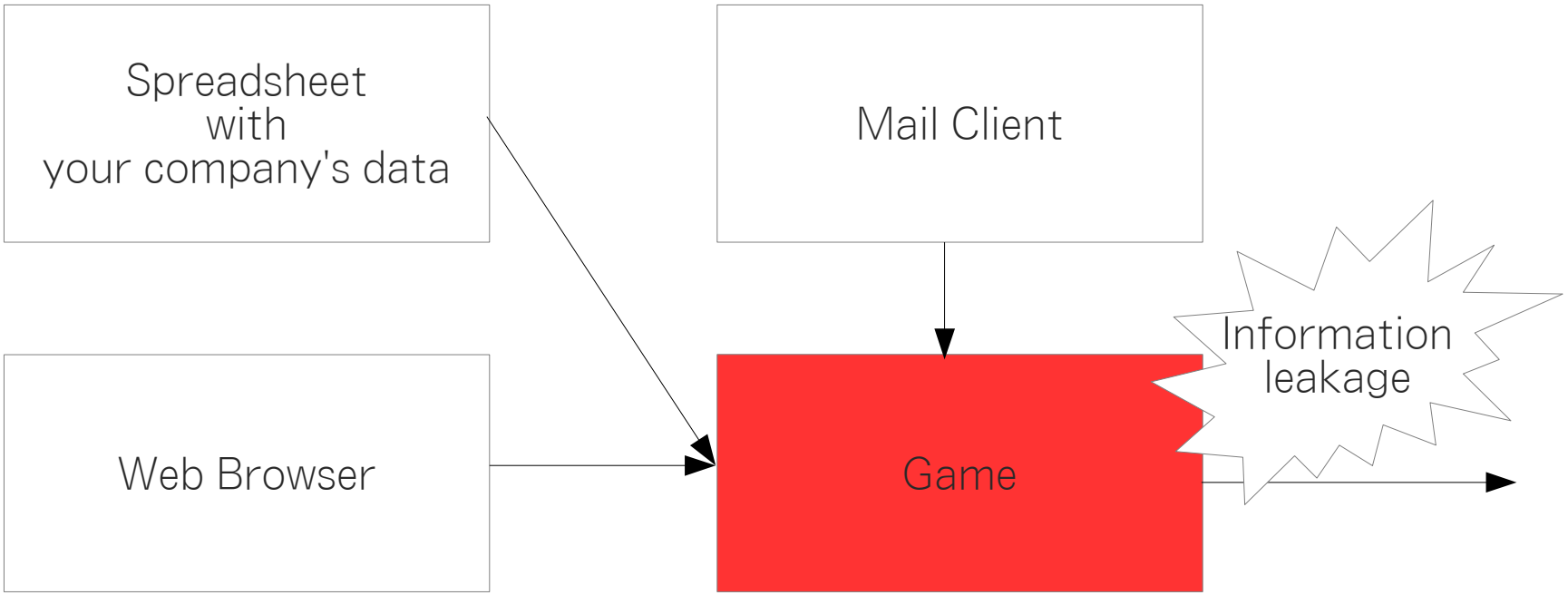
Desktop Environment

- People use different applications there



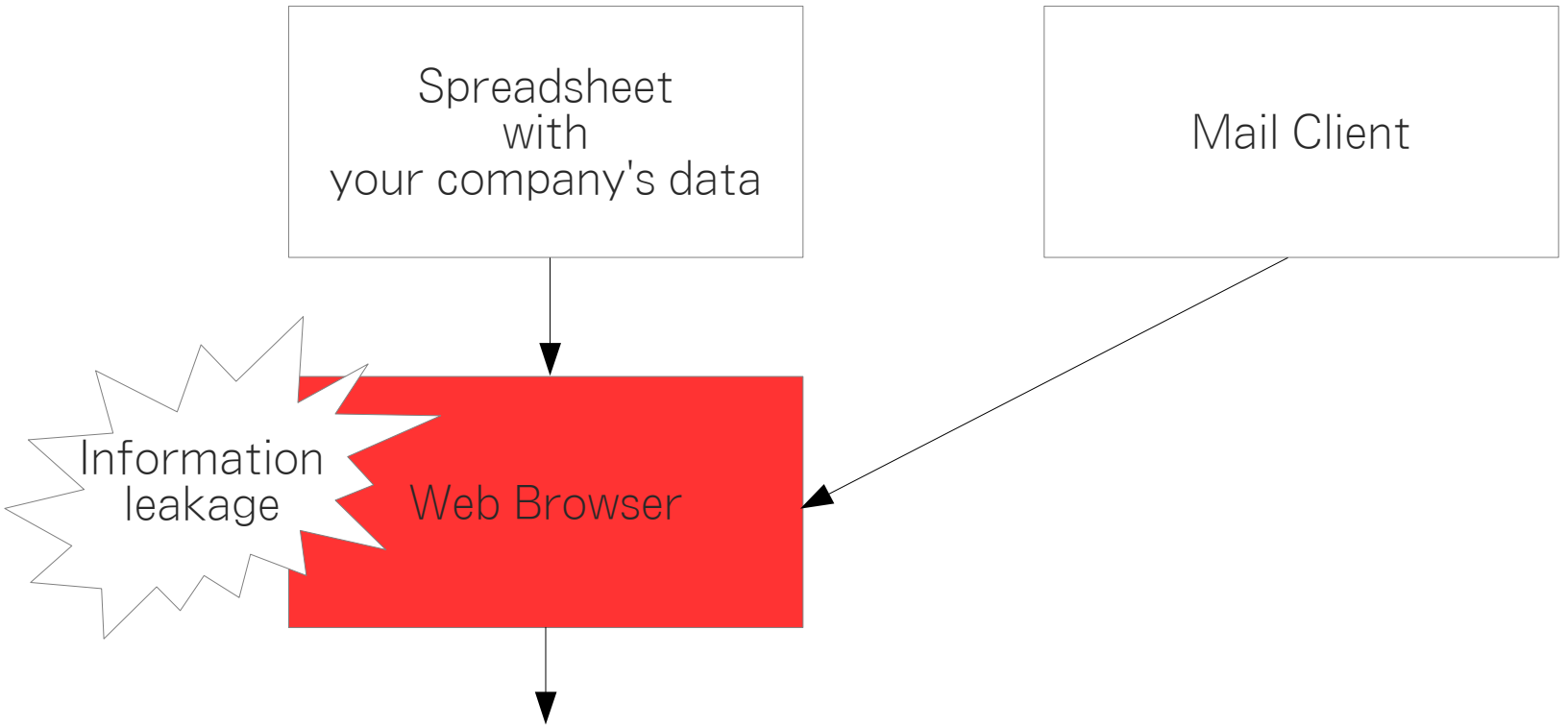
Desktop Environment

- If this game was malware?



Desktop Environment

- If the Web Browser has vulnerability?



It's Painful!

Two Approaches

- Security by Correctness
- Security by Isolation

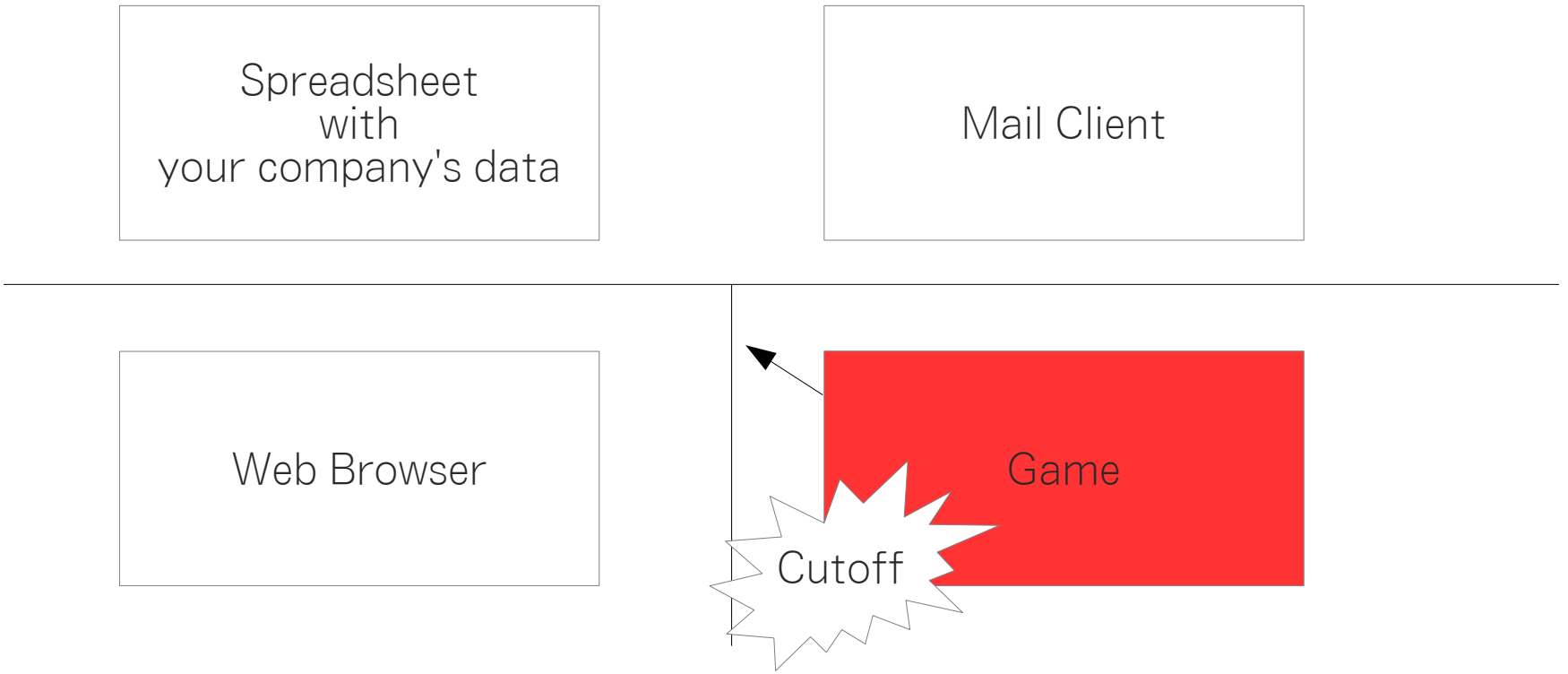
Security by Correctness

- Code Auditing
- Developers education
  - Microsoft Security Development Lifecycle
- Testing
  - Fuzzing
- “Safe” Programming Language

- It doesn't work in practice!

Security by Isolation

- We want the OS to provide isolation between various apps
- If some of them get compromised...



Security by Isolation

- We want to even “decompose” some apps...
- e.g. Web Browser
  - Internal Systems
  - Shopping
  - News
  - Googling

Security by Isolation

- Isolation provided by OSes are not enogh?
  - Address space isolation
  - User accounts isolation
  - ACL
  - Kernel/User space separation
  - chroot
  - systrace
  - SELinux
  - Secure level of BSD

- They don't work in practice!

Security by Isolation

- Monolithic kernels are buggy!
- Hundreds of 3rd-party drivers cannot be made secure!

"One bug to rule them all!"

Then, Qubes OS

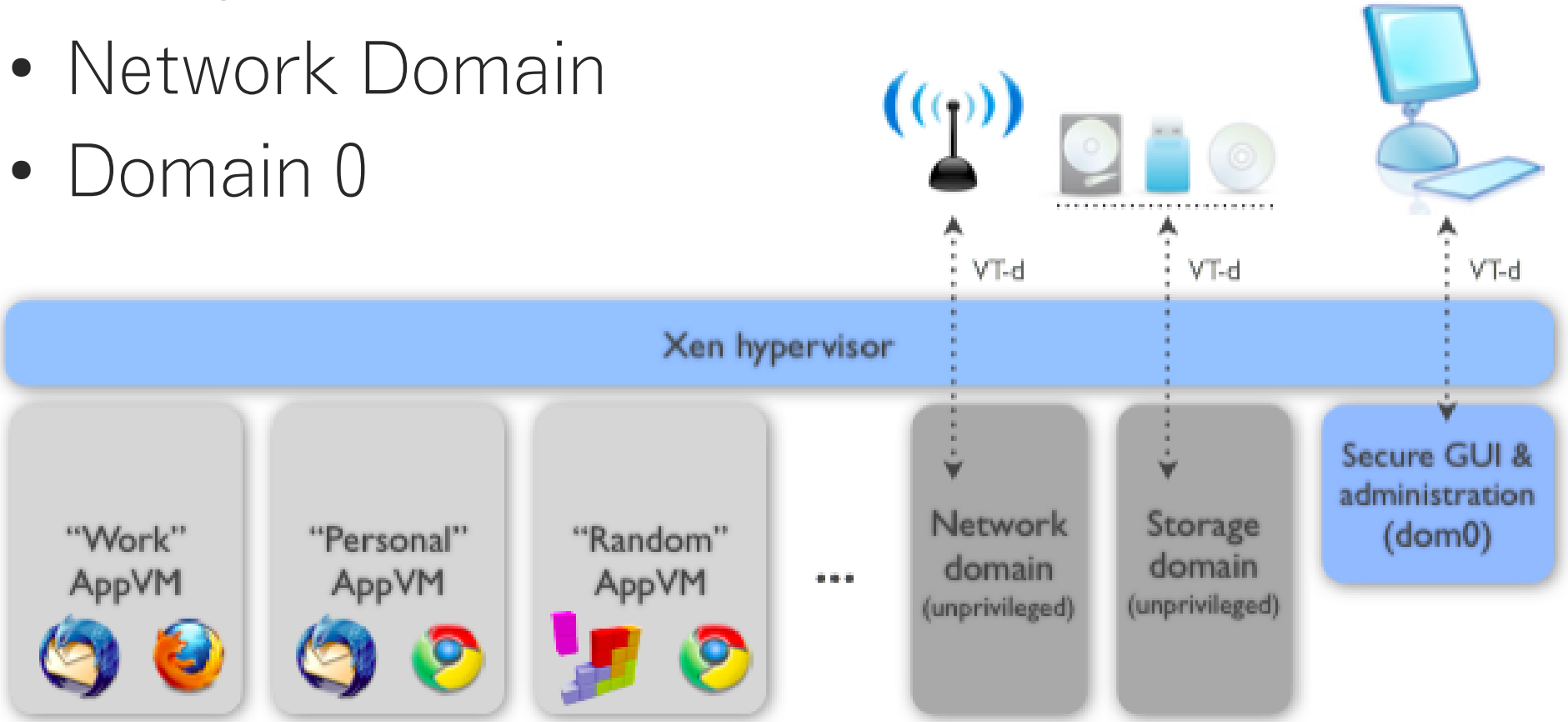
Virtualization for rescue!

Melits of virtualization

- Bug(vuln) is proportional to LOC[SOSP01]
- Linux: ten of millions LOC!
- Bare-metal hypervisor: 100k~300k LOC only!

Conceptual Diagram

- App Domain
- Strage Domain
- Network Domain
- Domain 0



Come true Isolation!!!

Dom0

- Provides secure environment and manager
- Dom0 doesn't contain Network function and Storage function
- Only 25k LOC!!!!!!!!

Strage Domain

- Non-privileged VM
- Only support Storage function

Network Domain

- Non-privileged VM
- Only support Network function

AppVM

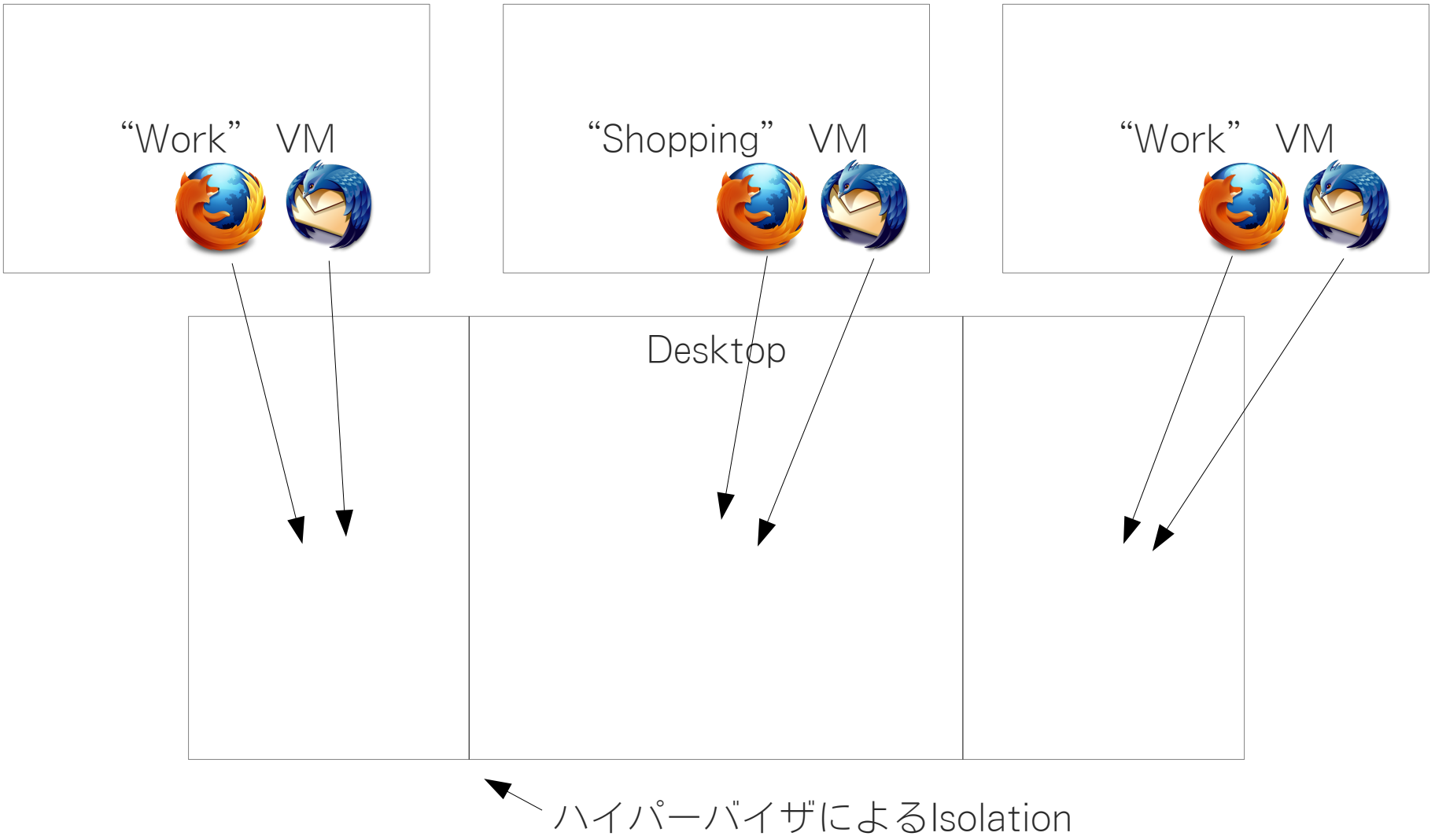
- Main Qubes building blocks(cubes)
- Hosts user applications
- We can create VM(Domain) depending on their Use
  - Work
  - Shopping
  - Personal
- Domains are isolated each other → SECURE!
- Created by Template VM(Read Only)

AppVM

- Disposable VM
  - Only supports ONE application
  - If compromised, there are no informations

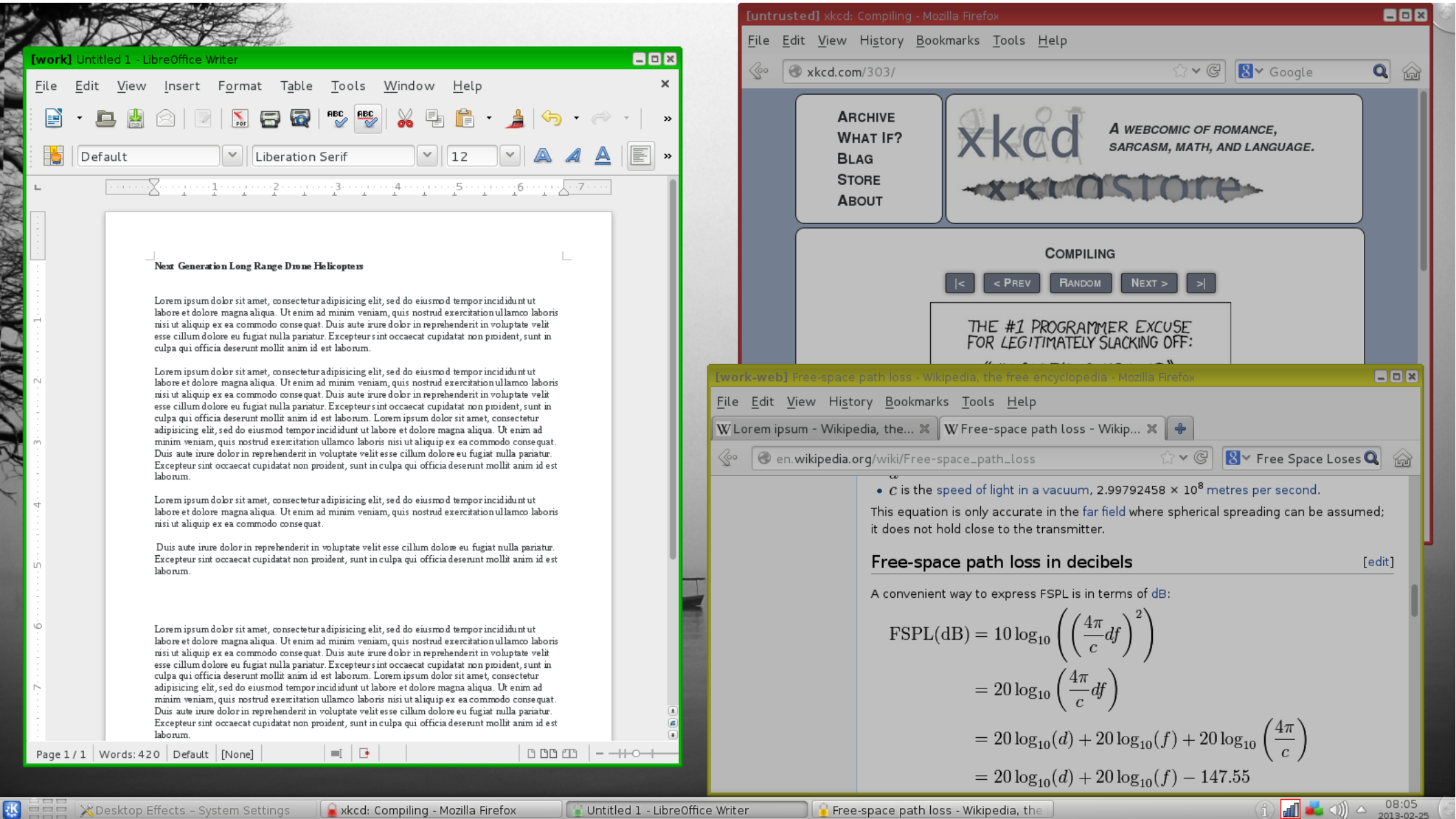
- Lightweight
  - 400MB per VM
- Centrally Updatable
- Each app gets a label (VM name + color frame) that is applied by the Window Manager running in Dom0

AppVM

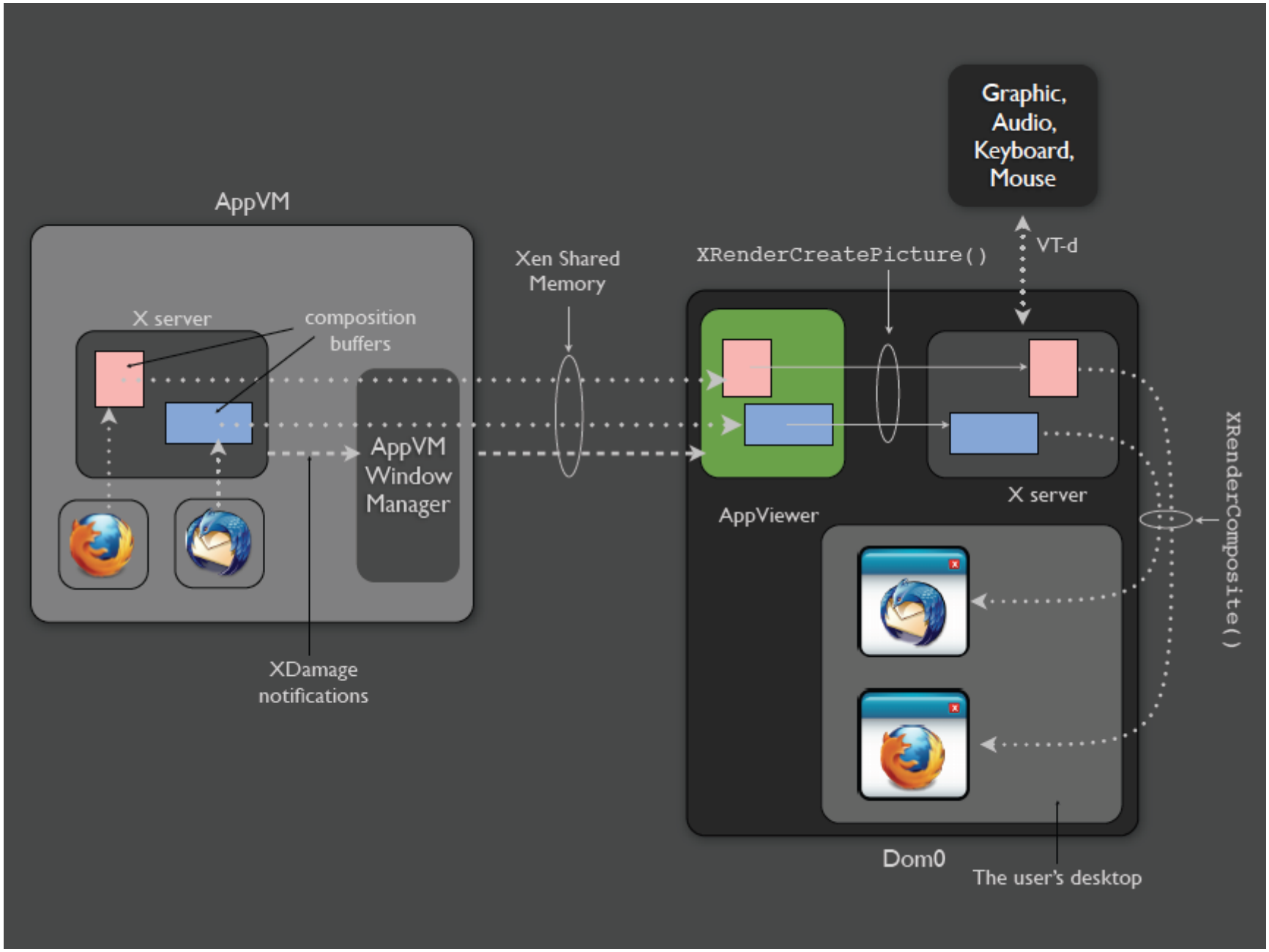


Screenshot

http://wiki.qubes-os.org/trac/attachment/wiki/QubesScreenshots/r2b2-kde-three-domains-at-work.png



GUI Virtualization



VM Protection

- Research about VM Protections
- Overshadow[ASPLO08]
  - Get context of Guest OS from VMM
  - Encrypt pages at memory access
  - Show process to not-encrypted memory
  - Need original loader
- SP3[Vee08]
  - Process memory encyption from VMM
  - Set accsess control per page
  - Has both encrypted page and not-encrypted page
    → Reduction of Overhead

VM Protection

- Qubes OS uses Intel VT-d and Intel TXT Protecting VM
- DMA Protection
  - Direct Memory Access
  - R/W memory from HW
  - No need of CPU

DMA Virtualization by Intel VT-d

1. HW → DMA Request
2. DMA Remapping Engine refers to Device Assignment Structure
3. Get Address Translation Structure

DMA Virtualization by Intel VT-d

- Prevents access from the address range other than the VM at address translation
- At early boot sequense before VT-d initialized, Intel TXT protects VM

Intel TXT

- Trust
  - All work as expected!
  - **Identity** and **Measurement**
- Establish Trust by RTM(Root of Trust for Measurement)
  - Reliable engine makes a measurement of integrity
  - Root of Trust → Chain of Trust

Intel TXT

- RTM
  - RTM cannot measures itself
- Static RTM
  - RTM is firmware
  - Building Chain of Trust from booting
- Dynamic RTM
  - RTM is GETSEC[SENTER] instruction
  - Building Chain of Trust from executing instruction
  - SENTER enable DMA protection so we can protect VM!

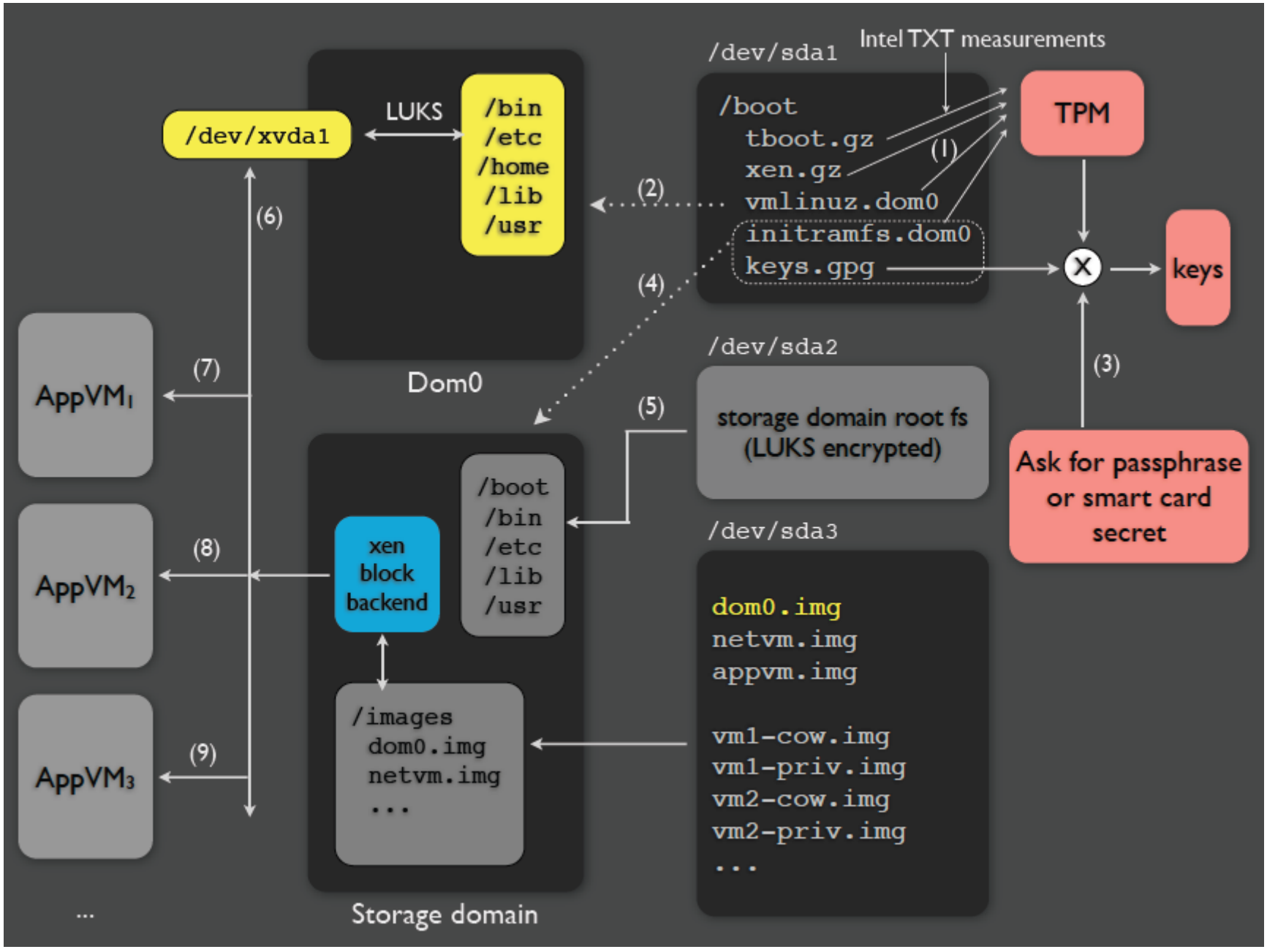
    “Kill two birds with one stone”

Intel TXT

- Intel TXT uses both SRTM and DRTM
- BIOS(chip) → (SRTM) → bootloader → (SRTM) → os → (DRTM) → hypervisor

(thx @yuzuhara)

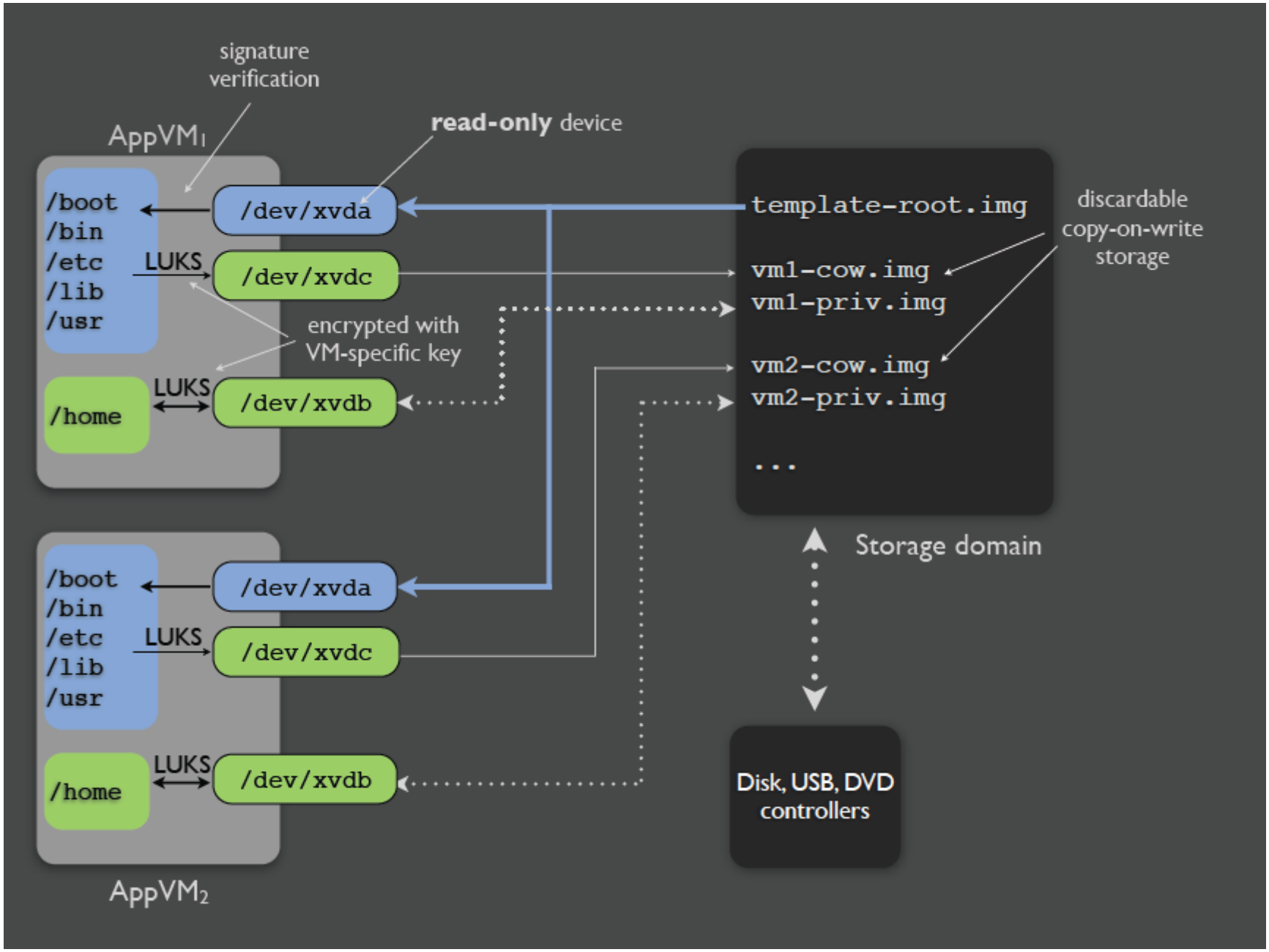
Strage


/dev/sda1
Intel TXT measurements
/boot
tboot.gz
xen.gz
vmlinuz.dom0
initramfs.dom0
keys.gpg
(1)
TPM
X
keys
/dev/xvda1
LUKS
/bin
/etc
/home
/lib
/usr
(2)
(3)
(4)
(5)
(6)
(7)
(8)
(9)
Dom0
/dev/sda2
storage domain root fs
(LUKS encrypted)
Ask for passphrase
or smart card
secret
AppVM1
AppVM2
AppVM3
...
/boot
/bin
/etc
/lib
/usr
xen
block
backend
/images
dom0.img
netvm.img
...
Storage domain
/dev/sda3
dom0.img
netvm.img
appvm.img
vm1-cow.img
vm1-priv.img
vm2-cow.img
vm2-priv.img
...

Cross-VM

- Qubes OS has some Cross-VM functions
  - Clipboard sharing
  - File transfer via virtual disk
- Cross VM vulnerability is easily targeted
- Insert rootkit at LiveMigration[BlackHat DC08]
- Cross VM Side Channel Attack[CCS12]
  - Estimate the access from another VM from response when malicious VM access physical cache continuously
  - Might steal the key

Filesystem


signature verification
read-only device
AppVM1
/boot
/bin
/etc
/lib
/usr
LUKS
/dev/xvda
/dev/xvdc
encrypted with VM-specific key
/home
LUKS
/dev/xvdb
template-root.img
vm1-cow.img
vm1-priv.img
vm2-cow.img
vm2-priv.img
...
discardable copy-on-write storage
Storage domain
/boot
/bin
/etc
/lib
/usr
LUKS
/dev/xvda
/dev/xvdc
/home
LUKS
/dev/xvdb
Disk, USB, DVD controllers
AppVM2

Summaly

- Domain oriented VM
- Creates Xen's VM per use
- Seamless operation by GUI virtualization
- DMA protection by Intel VT-d
- Strage protection by Intel TXT
- Filesystem protection by VM-specific key

See

qubes-os.org

Q&A?

Thank you!