Practical Overview of a Xen Covert Channel

Mickaël Salaün

ESIEA
<salaun@esiea-recherche.eu>

May 11, 2009

18th EICAR Annual Conference

Introduction

Virtualization

- Virtualization comes up at the 60th with *IBM CP/CMS*
- This last years more softwares comes with different methods, and now virtualization use is growing more and more

Virtualization

- Virtualization comes up at the 60th with *IBM CP/CMS*
- This last years more softwares comes with different methods, and now virtualization use is growing more and more

Common Uses

Multiples OS in an unique hardware at the same time:

- Host sharing (datacenter, computer farm)
- Mutualization (e.g. multiple application servers in one real computer)
- "Virtual" machine isolation

Isolation

Multilevel security

Why?

- Keep in a safe place critical data
- Avoid leaks
- Stay out of reach from malware...

Multilevel security

Why?

- Keep in a safe place critical data
- Avoid leaks
- Stay out of reach from malware...

Opposite Constraints

- Data isolation
- Data sharing

Compromised System

Goals

- Stay in place as long as possible
- Remain stealthy
- Use the system!

Compromised System

Goals

- Stay in place as long as possible
- Remain stealthy
- Use the system!

Needed Features

- Designed to remain hidden
- Communicate with the outside

Covert Channel

Definition

Covert channels are those that "use entities not normally viewed as data objects to transfer information from one subject to another." [Kemmerer, Richard A.]

Covert Channel

Definition

Covert channels are those that "use entities not normally viewed as data objects to transfer information from one subject to another." [Kemmerer, Richard A.]

Software Level

- Too permissive implementation
- Design bugs...

Covert Channel

Definition

Covert channels are those that "use entities not normally viewed as data objects to transfer information from one subject to another." [Kemmerer, Richard A.]

Software Level

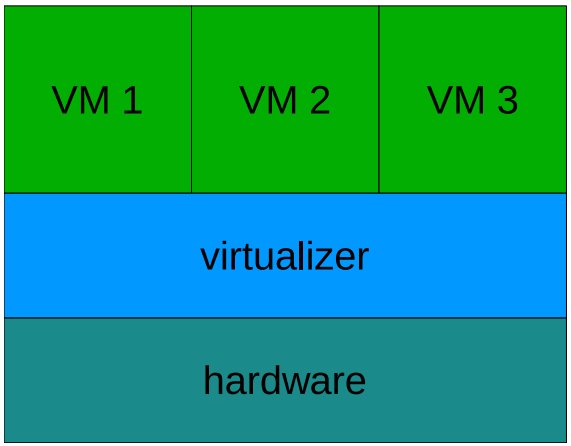
- Too permissive implementation
- Design bugs...

Hardware Level

- Device with residual memory
- Time factor (e.g. CPU time processing)

Virtualization

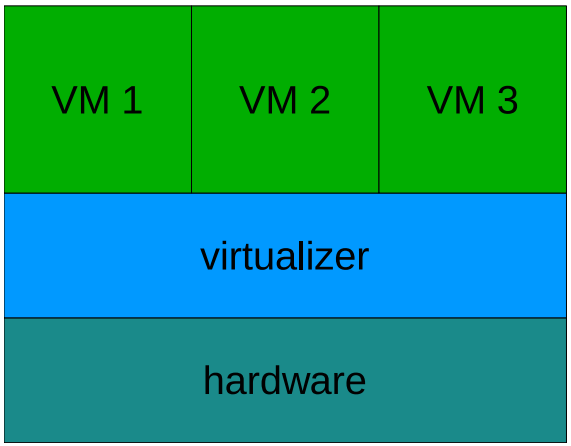
Features and Expectations



Main Goals

- Many virtual computers
- Protection between guests
- Virtualizer protection from virtual guests
- ...and protection from hardware

Features and Expectations



Main Goals

- Many virtual computers
- Protection between guests
- Virtualizer protection from virtual guests
- ...and protection from hardware

Main Problems

- Loads/devices sharing

$\Rightarrow$ Quality of service mechanism

esiea

Xen Overview

Open Source Software

- Possibility to audit the code

$\Rightarrow$ Increase trustworthy

Xen Overview

Open Source Software

- Possibility to audit the code

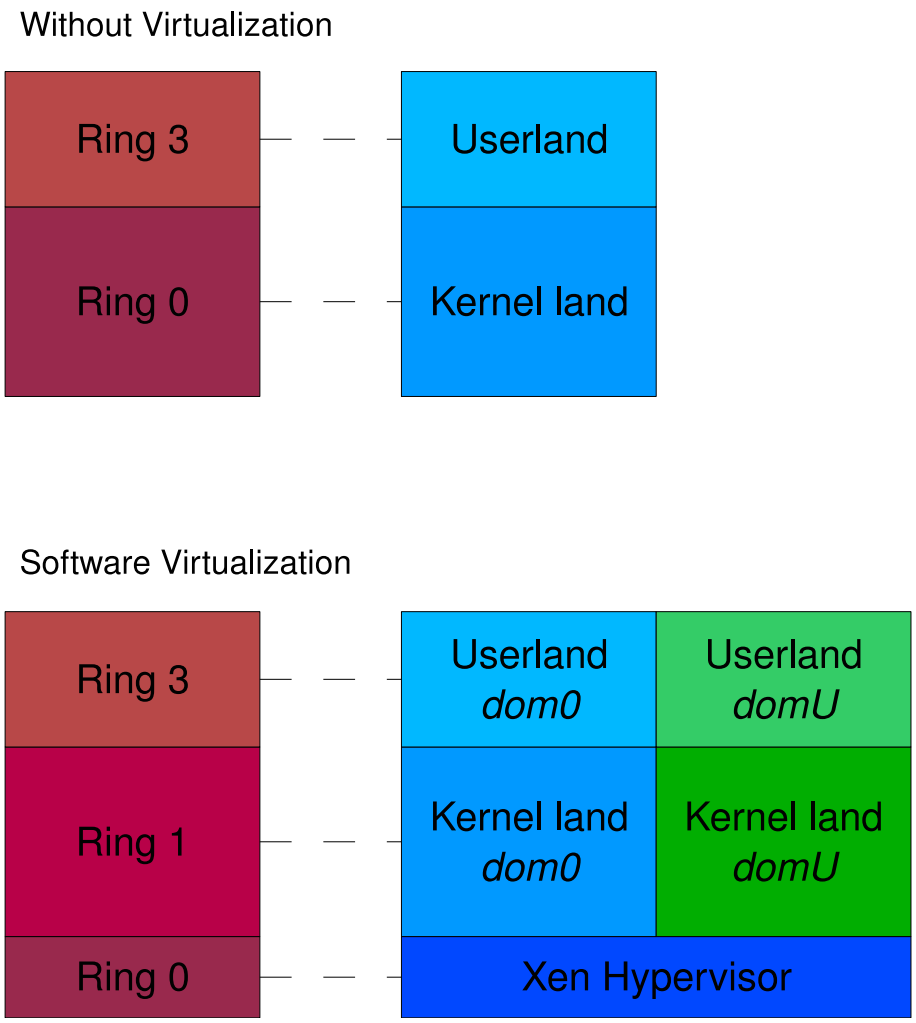
$\Rightarrow$ Increase trustworthy

Paravirtualization System

- Hypervisor:
  - Virtualizer in the lowest ring
  - Aware guests

  $\Rightarrow$ Hight performances
- Hypercalls:
  - Virtualizer "syscall"
  - Communication features (e.g. data sharing, administration)

Xen Architecture (32 bits)



Guests OS aware of

- Administrator domain (*dom0*)
- User domains (*domU*)

$\Rightarrow$ *hypercalls*

Memory Management

Hypercalls

- Memory allocation
- Data sharing
- A lot of things...

Memory Management

Hypercalls

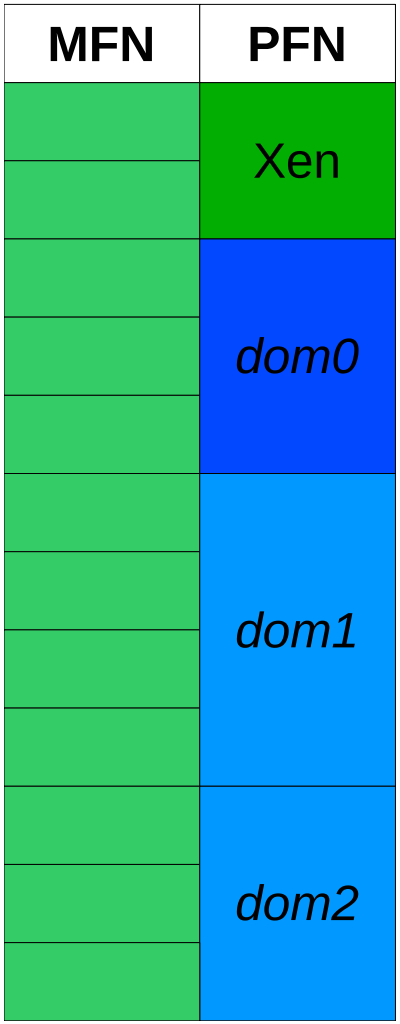
- Memory allocation
- Data sharing
- A lot of things...

Memories

- Virtual memory for userland
- Pseudo-physical memory for OS (common physical memory)
- Machine memory for hypervisor

Waterproofness

mfn2pfn:



The Pseudo-physical Transition Table

- Same table for all guests: for a performance purpose (less context switching)
- Some addresses usable for reading: guest's ones and the shared space (under control)
- Can only write in our one memory space (hopefully!)
- No entry check: the guest manage its one allocations (and mechanism) alone

esiea

XenCC

The Xen Weakness

A Design Feature

- The trick: use the shared pseudo-physical memory table

$\Rightarrow$ the PFN table can be read in most part (addresses of other guests)

The Xen Weakness

A Design Feature

- The trick: use the shared pseudo-physical memory table

$\Rightarrow$ the PFN table can be read in most part (addresses of other guests)

Covert Channel Mechanism

- Put data in place of address: virtual (useless) memory allocation with custom addresses
- Make them recognizable with a special tag: custom protocol for data exchange

Communication

Protocol Design

- Need an initial knowledge from each guest to know each other
- Possibility to create a "chat room" between accomplice guests

Communication

Protocol Design

- Need an initial knowledge from each guest to know each other
- Possibility to create a "chat room" between accomplice guests

The Header Tag

- Identifier
- Acknowledgement
- Remaining data size
- Current data size

Communication

Data Extraction

- First reading: look for the accomplice's tag in all the table and record the tag place when its found
- Next times: use the previous location to read again

Communication

Data Extraction

- First reading: look for the accomplice's tag in all the table and record the tag place when its found
- Next times: use the previous location to read again

Linux Implementation

- Need to be able to call hypercalls (kernel land)
- Easy use

$\Rightarrow$ A Linux driver (LKM: virtual device)

Use

Writing (guest 1)

```
dom1:~# echo msg dom1 > /dev/xencc
```

Use

Writing (guest 1)

```
dom1:~# echo msg dom1 > /dev/xencc
```

Reading (guest 2)

```
dom2:~# dd count=1 if=/dev/xencc
msg dom1
0+1 records in
0+1 records out
9 bytes (9 B) copied, 0.000185 s, 48.6 kB/s
```

Interesting Points

Drawbacks

- Push and pop design (no synchronisation)
- A lot of memory in saw of the data transfer
- Need to be careful with address range in use
- May not be discreet (depending of use)

Interesting Points

Drawbacks

- Push and pop design (no synchronisation)
- A lot of memory in saw of the data transfer
- Need to be careful with address range in use
- May not be discreet (depending of use)

Advantages

- Work well for an off-the-shelf Xen! ($\leq$ 450 KB/s)
- Go through the Xen security policy

$\Rightarrow$ Can be use as a new stealthy communication channel by malwares

Conclusion

Counter measures

Detection

- No public implemented solution for now
- Statistics of hypercalls usage about *mfn2pfn* table access (time)
- Look for some similarity access of guests to the table (space)

Counter measures

Detection

- No public implemented solution for now
- Statistics of hypercalls usage about *mfn2pfn* table access (time)
- Look for some similarity access of guests to the table (space)

Prevention

- For now: use the shadow page tables (lower performances)
- The better way: a *mfn2pfn* table for each guest containing only useful data

So What?

About Xen

- A great virtualization platform (new improvements: IOMMU, stub domains...)
- Some design flow regardless of the use

So What?

About Xen

- A great virtualization platform (new improvements: IOMMU, stub domains...)
- Some design flow regardless of the use

About virtualization

- No initial need of secure isolation
- A good isolation is an hardware one, but...

So What?

About Xen

- A great virtualization platform (new improvements: IOMMU, stub domains...)
- Some design flow regardless of the use

About virtualization

- No initial need of secure isolation
- A good isolation is an hardware one, but...

Reactions?

- Covert-channels seems to not be interesting for developers

$\Rightarrow$ No real reaction about this problem...

Thanks for your attention.

Questions ?

code: `http://digikod.net/public/XenCC`