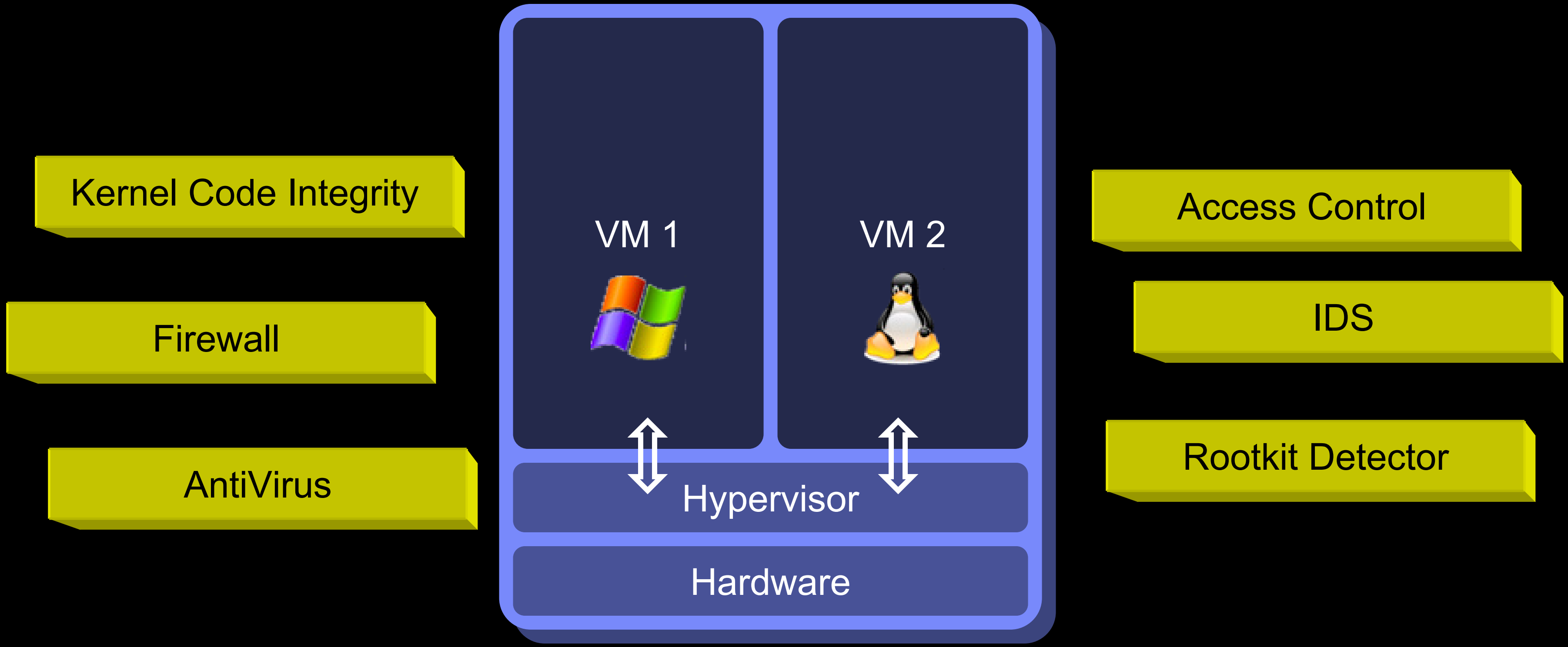
Mihai Christodorescu, Reiner Sailer, Douglas Lee Schales, Daniele Sgandurra, Diego Zamboni
IBM Research

Cloud Security Is Not (Just) Virtualization Security

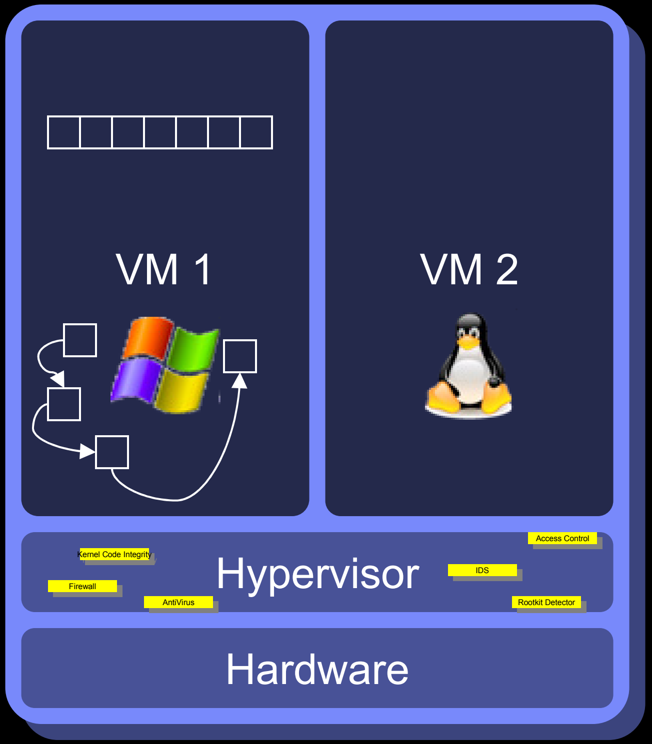
Virtualization Enables Many Security Applications

- Infrastructure clouds build on virtualization mechanisms.
- Virtualization allows for introspection into untrusted guest virtual machines (VMs).



Problem: What's Inside Matters

- Semantic gap: How do you give structure and meaning to data and code pages?
- VM lifecycle (snapshots, migration, reboots, updates) vs. security monitor lifecycle.
  - Information gap: How do you know you are monitoring the right data and code pages?

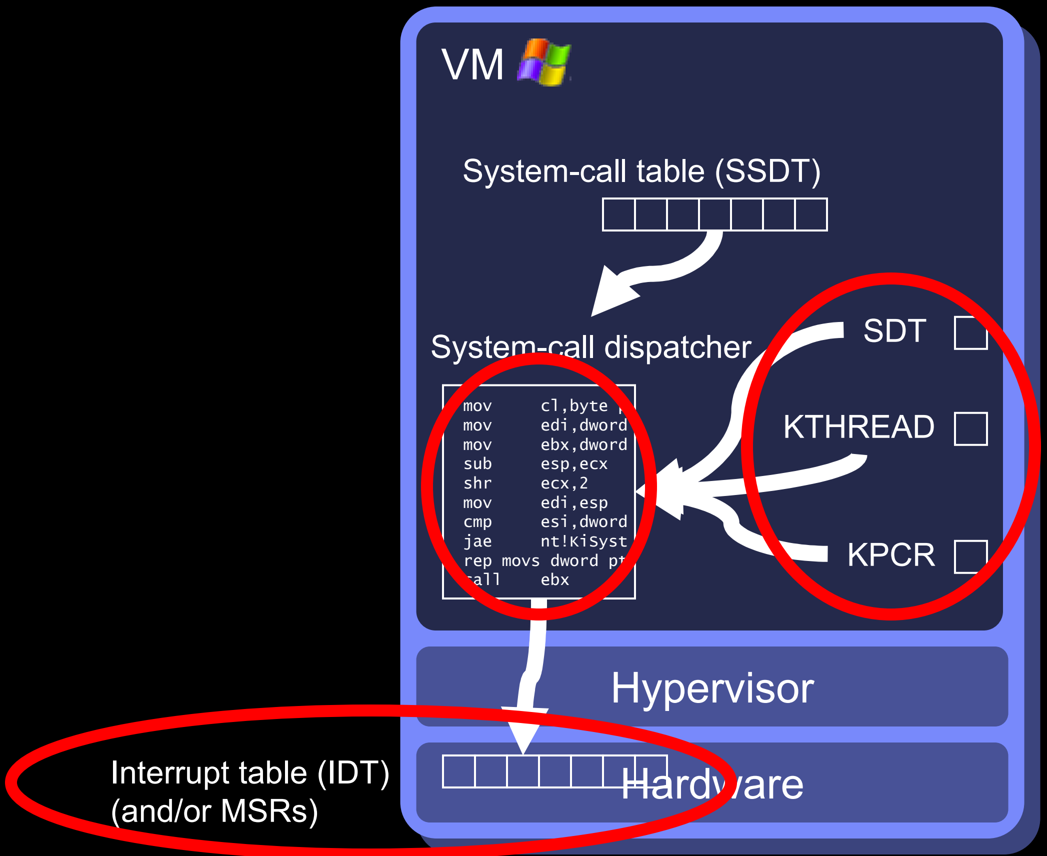


Our Contribution: Secure Introspection

- Key idea: combine discovery and integrity measurement of guest VM.
  - Implicit benefit: OS discovery is free.
  - Challenges: infected guest VM, non-cooperative guest OS.
- Result: Building block for security monitors that use introspection.

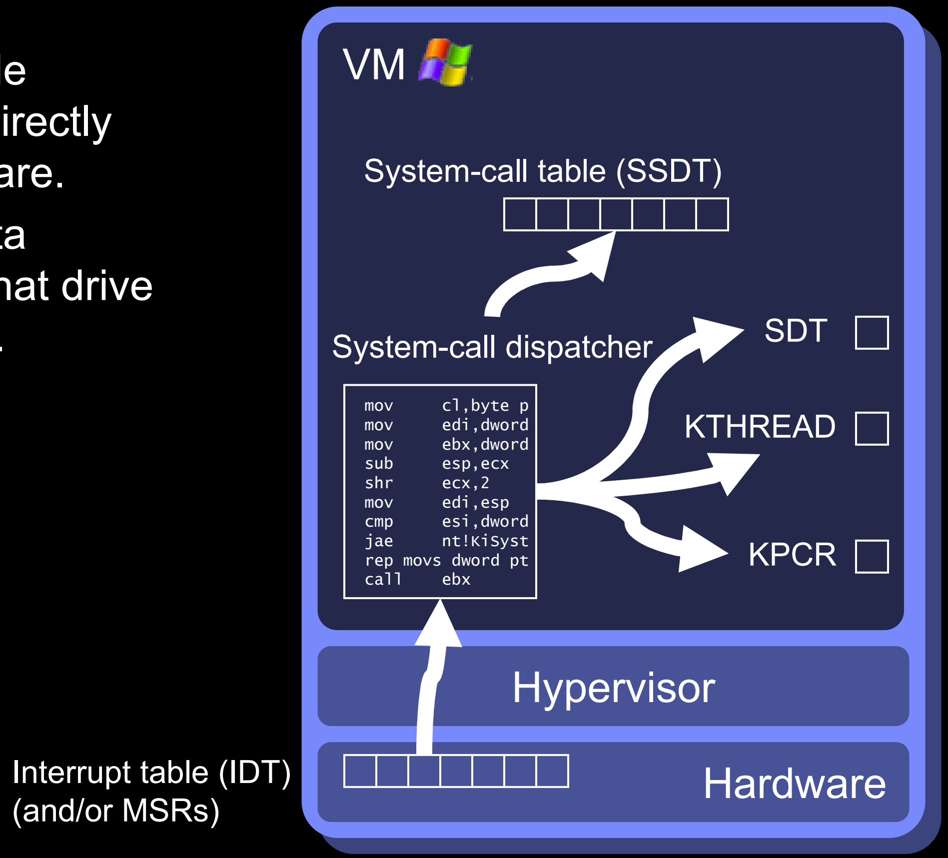
Why is Trust Needed? An Example.

- Goal: monitor the system-call table of the guest kernel.



Building Trust from Hardware State

1. Start with hardware state.
2. Explore code reachable directly from hardware.
3. Validate data structures that drive control flow.
4. Repeat as necessary.



Note:

Nothing OS-specific is used in this process.

Coverage

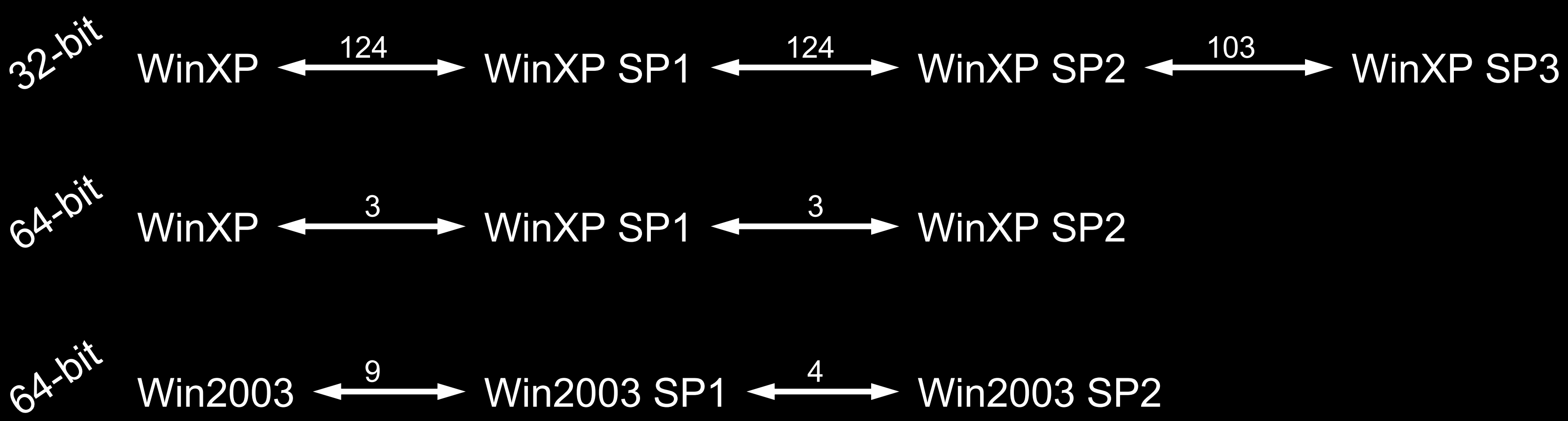
- How far can we reach into the kernel?

	Code reached
	without indirect-control flow
Windows XP	52%
Linux 2.6.24	24%

Minimal Coverage Needed for OS Identification

- If we measure the code as we traverse it, we can identify the guest OS.
  (e.g., “measure” = “hash the instruction stream in control-flow order”)

- Experimental observation:
  Code of one IDT entry is sufficient to identify the OS (and sometimes, that’s all we can use.)



Code Measurement for Validation

- Measurement of same kernel across boots made more complex by self-modifying code.
- Microsoft Windows relocates certain modules by rewriting absolute pointers in the code.
- Linux rewrites parts of its code during boot depending on the processor characteristics:
  - Replace LOCK prefixes with NOPs
- Both Windows and Linux support hot patching of the kernel:
  - Linux: ksplice available since 2.6.24
  - Windows: capability built into Windows Vista and 7

Putting Everything Together: Secure Introspection

- Information gap: How do you know you are monitoring the right data and code pages?
- A guest code page is trusted only if:
  it is reachable from hardware state via other trusted code pages
  and
  its code is valid.
- A guest control-flow data structure is trusted only if:
  it is used by trusted code
  and
  it points to trusted code pages.

Then a security monitor can reason about guest code and guest data that are trusted.

Mihai Christodorescu, Reiner Sailer, Douglas Lee Schales, Daniele Sgandurra, Diego Zamboni
IBM Research

IBM

Questions?

Cloud Security Is Not (Just) Virtualization Security