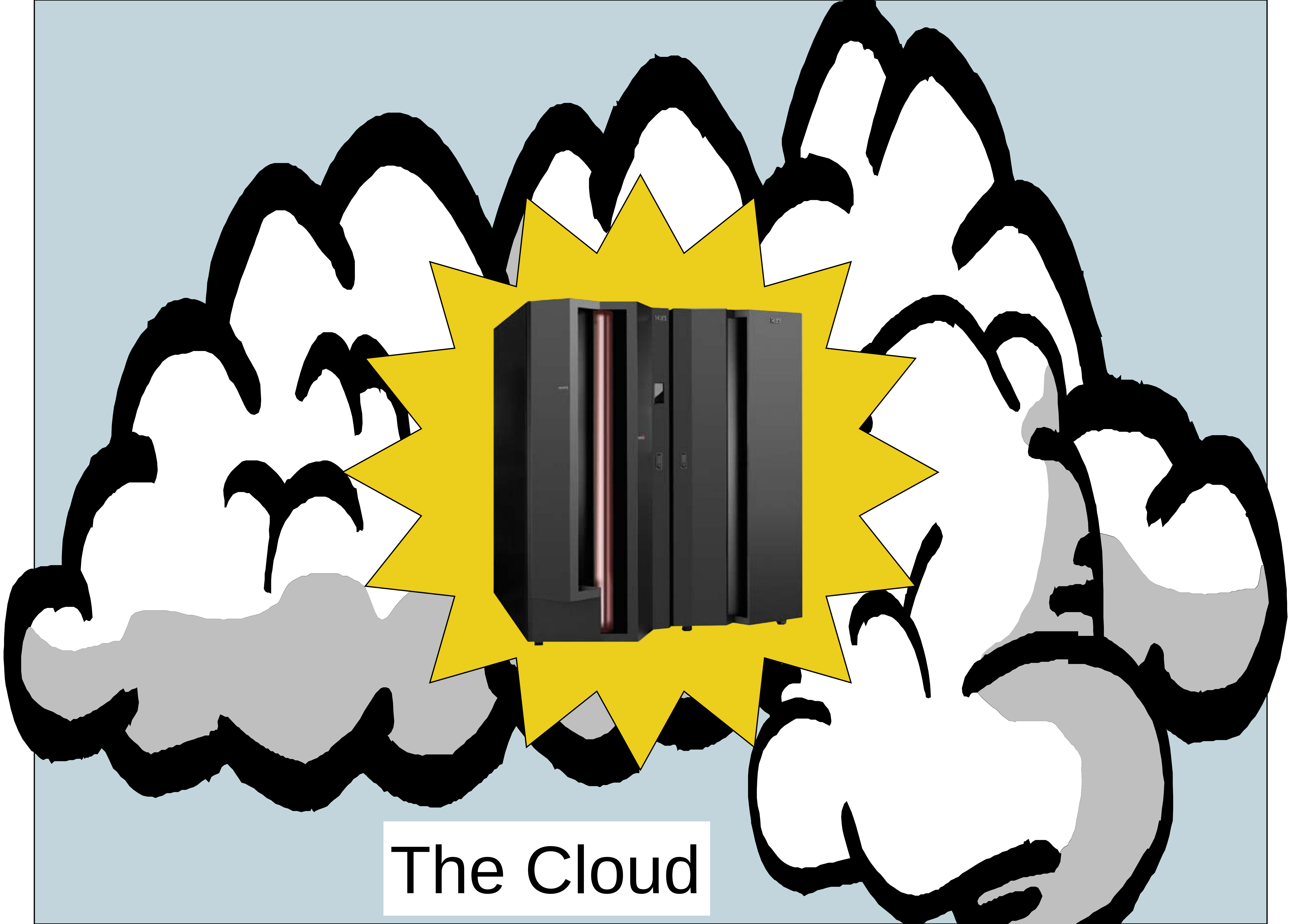
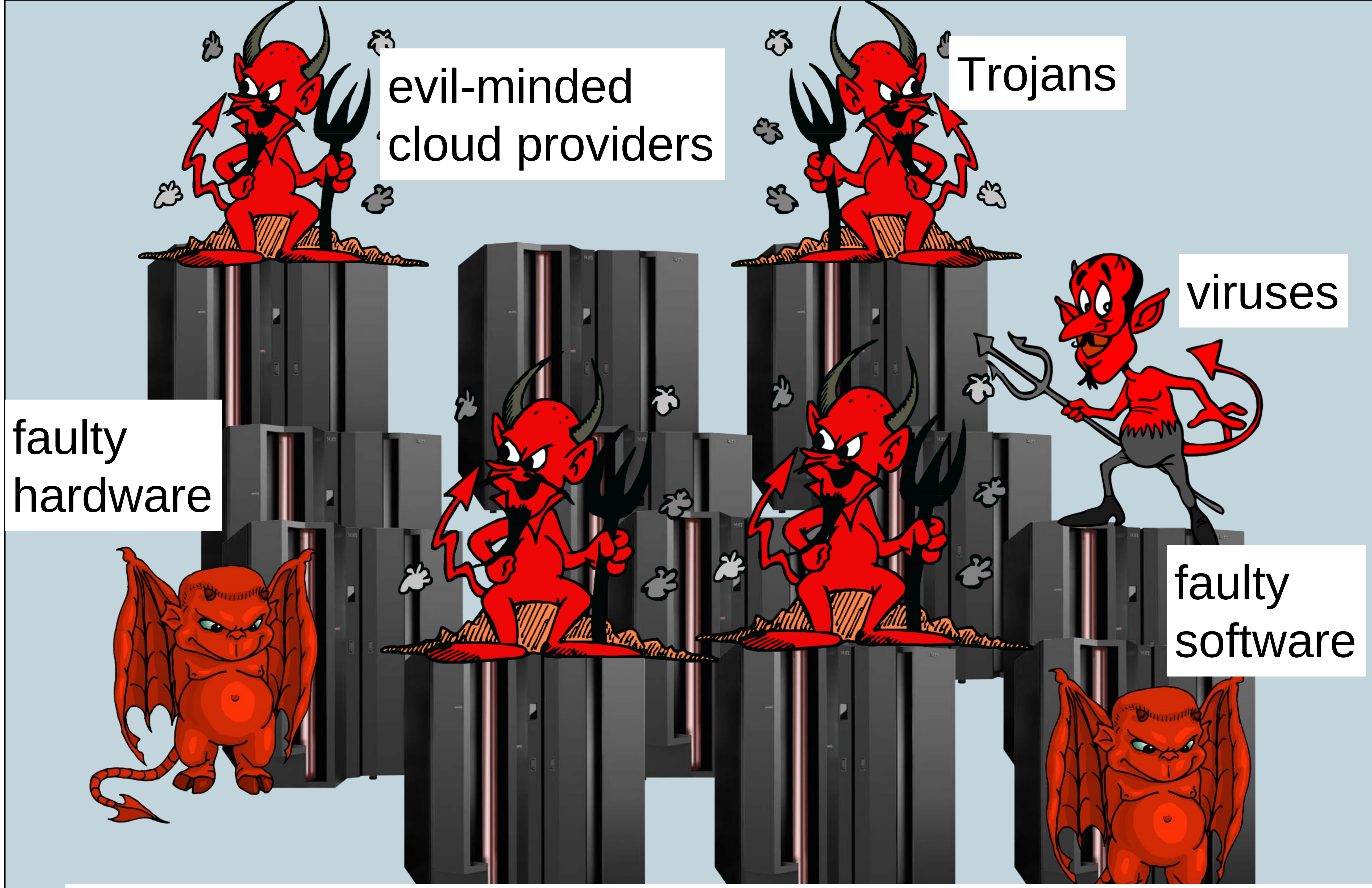
Cloud Security

Thanks to Ari Juels for parts of this deck!



The Cloud

Se rity pecialists’ vie



evil-minded cloud providers

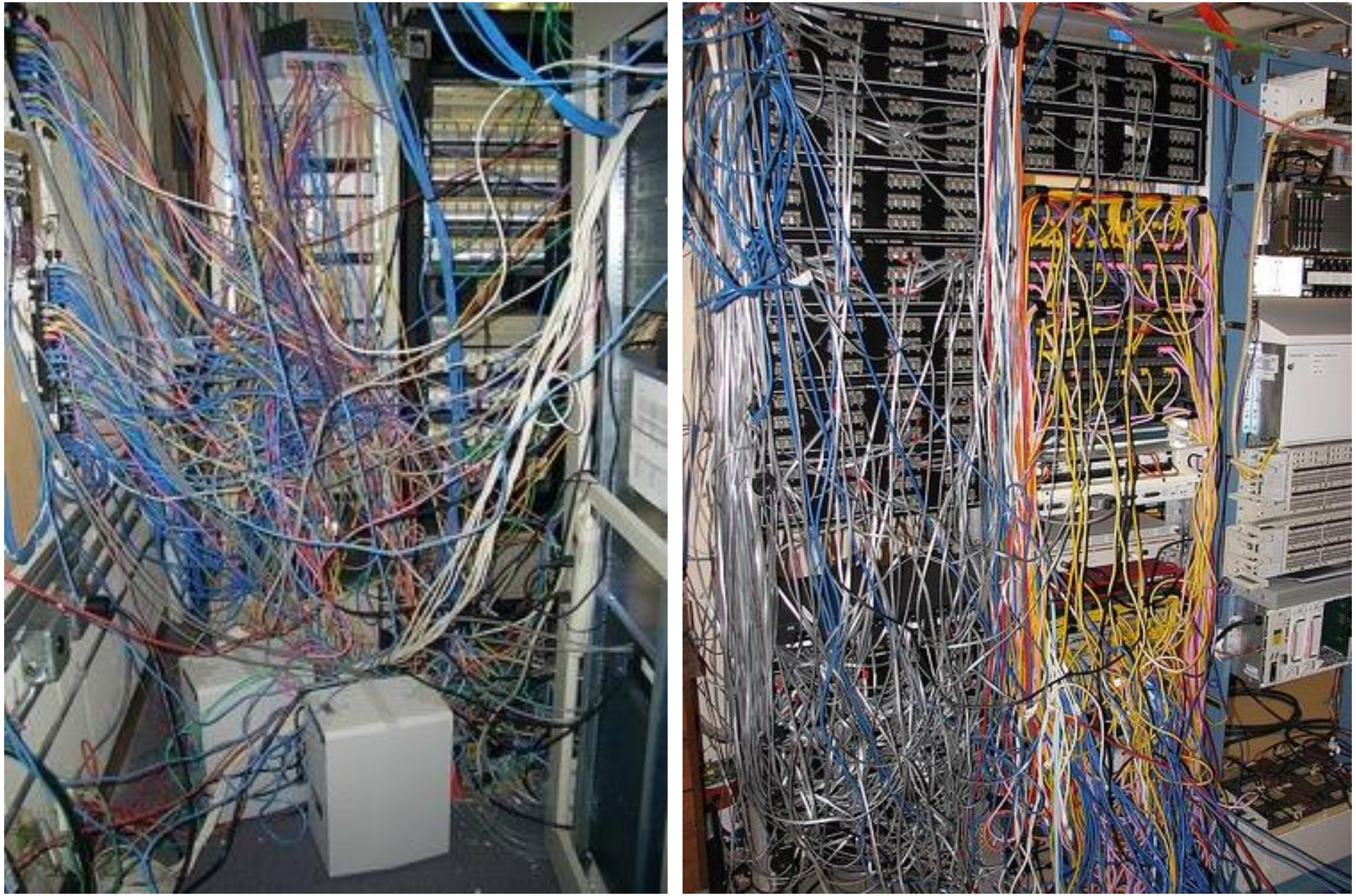
Trojans

viruses

faulty hardware

faulty software

Security specialists’ view of the Cloud



Another view of the Cloud

NIST definition of cloud computing

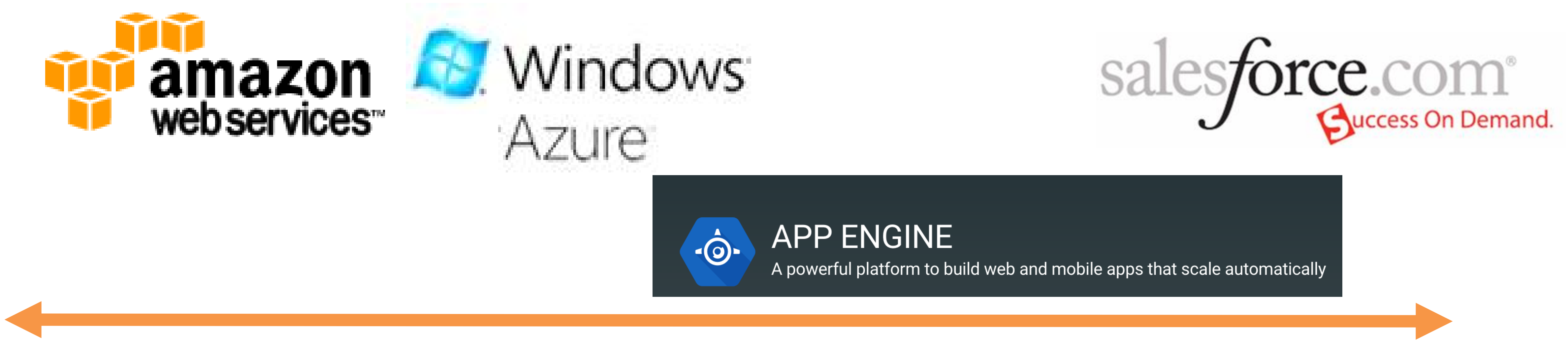
“Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.”

[P. Mell and T. Grance. "The NIST definition of cloud computing." (2011).]

Some terminology

- Cloud *tenant*: A customer, perhaps compromising multiple users, defined by its use of an exclusive virtual resource environment within the cloud.
- IaaS (Infrastructure as a Service): Tenant gets virtual machines (plus storage and network)
  - E.g., Amazon Web Services
- PaaS (Platform as a Service): Tenant gets a runtime environment / computing platform
  - E.g., Google App Engine
- SaaS (Software as a Service): Tenant gets accounts for an application
  - E.g., salesforce.com

Cloud providers



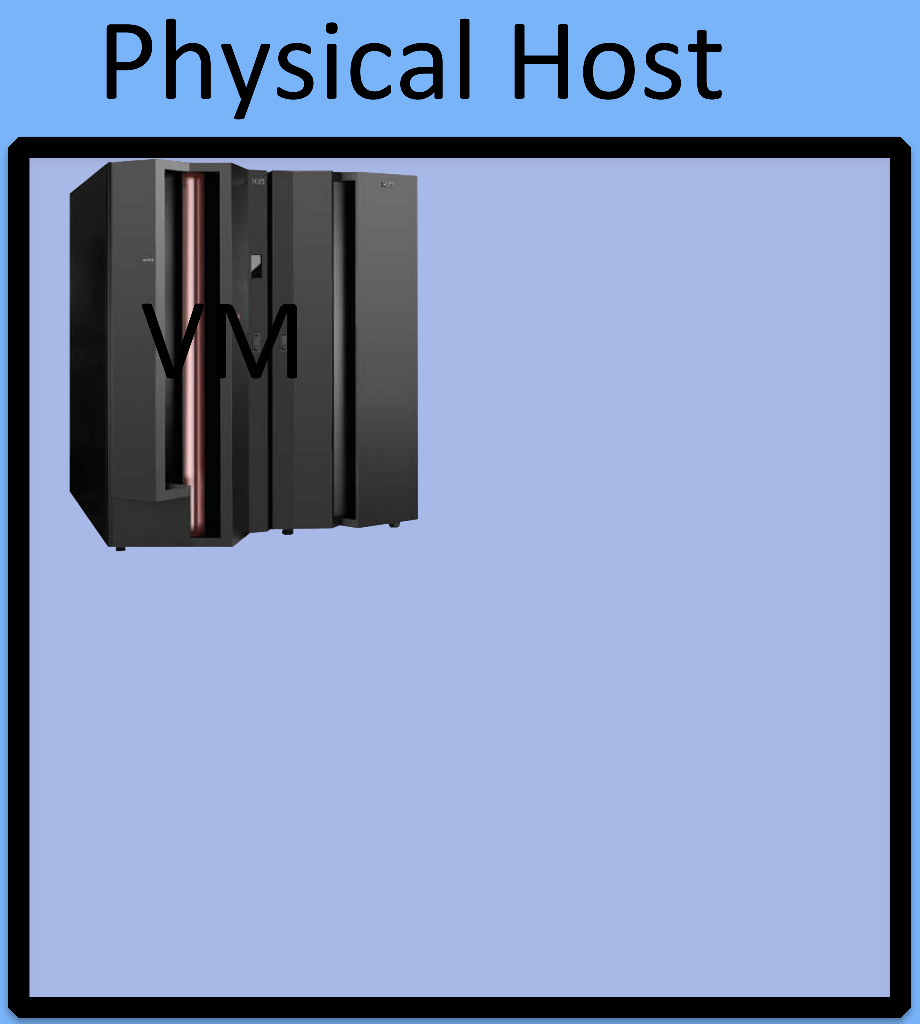
Infrastructure-as-a-service

Platform-as-a-service

Software-as-a-service

Users expect several forms of security from cloud provider, e.g.,

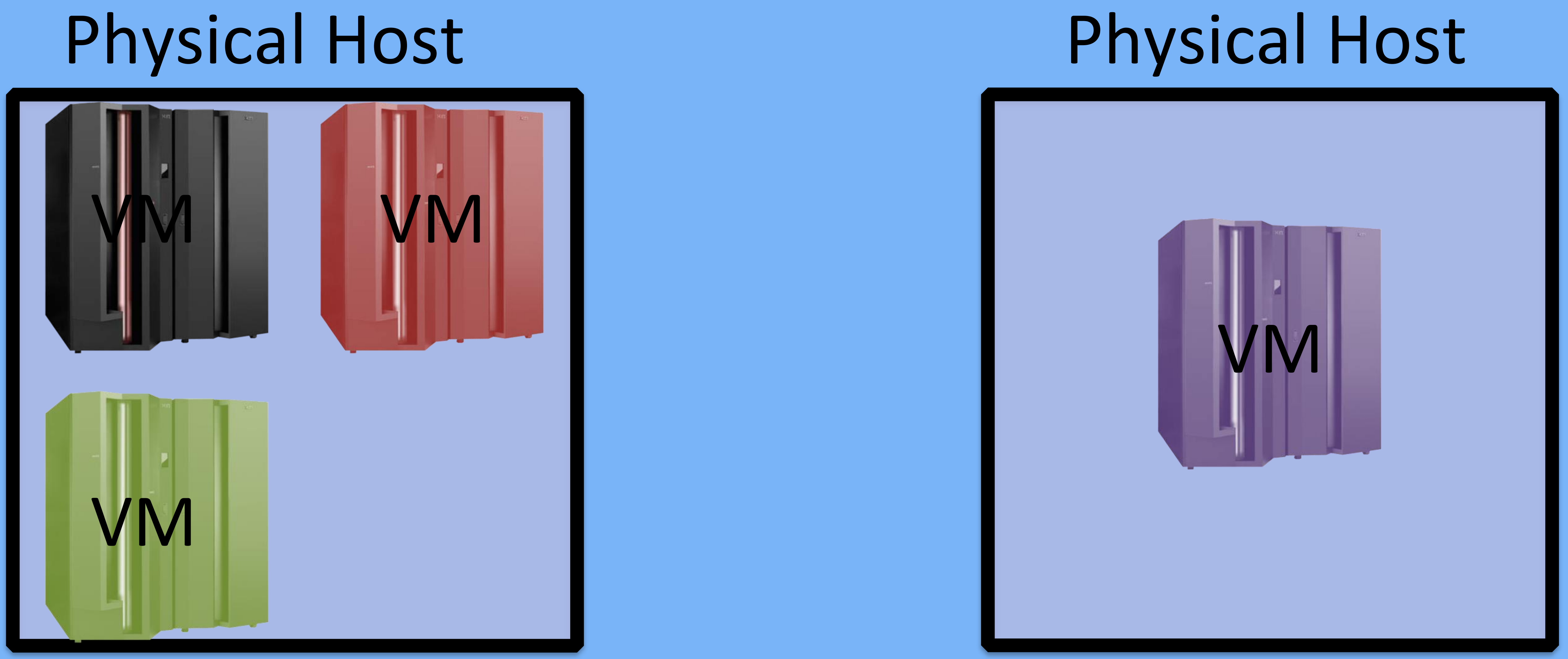
- Cloud provider should not spy on tenant data / processes
- Cloud provider should secure infrastructure from external attackers
- Cloud provider should secure infrastructure from internal attackers
  - Other tenants -> the part we'll focus on today



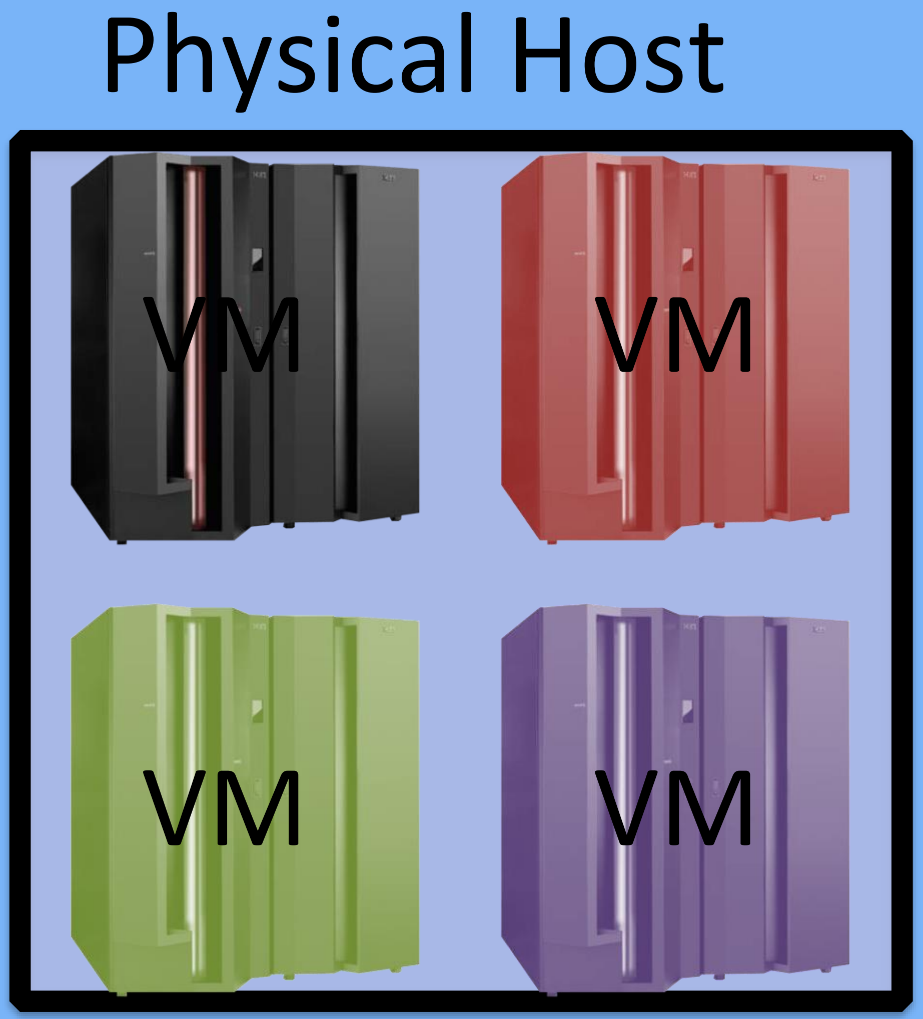
IaaS

A **virtual machine** (VM) emulates a hardware computer system / server.

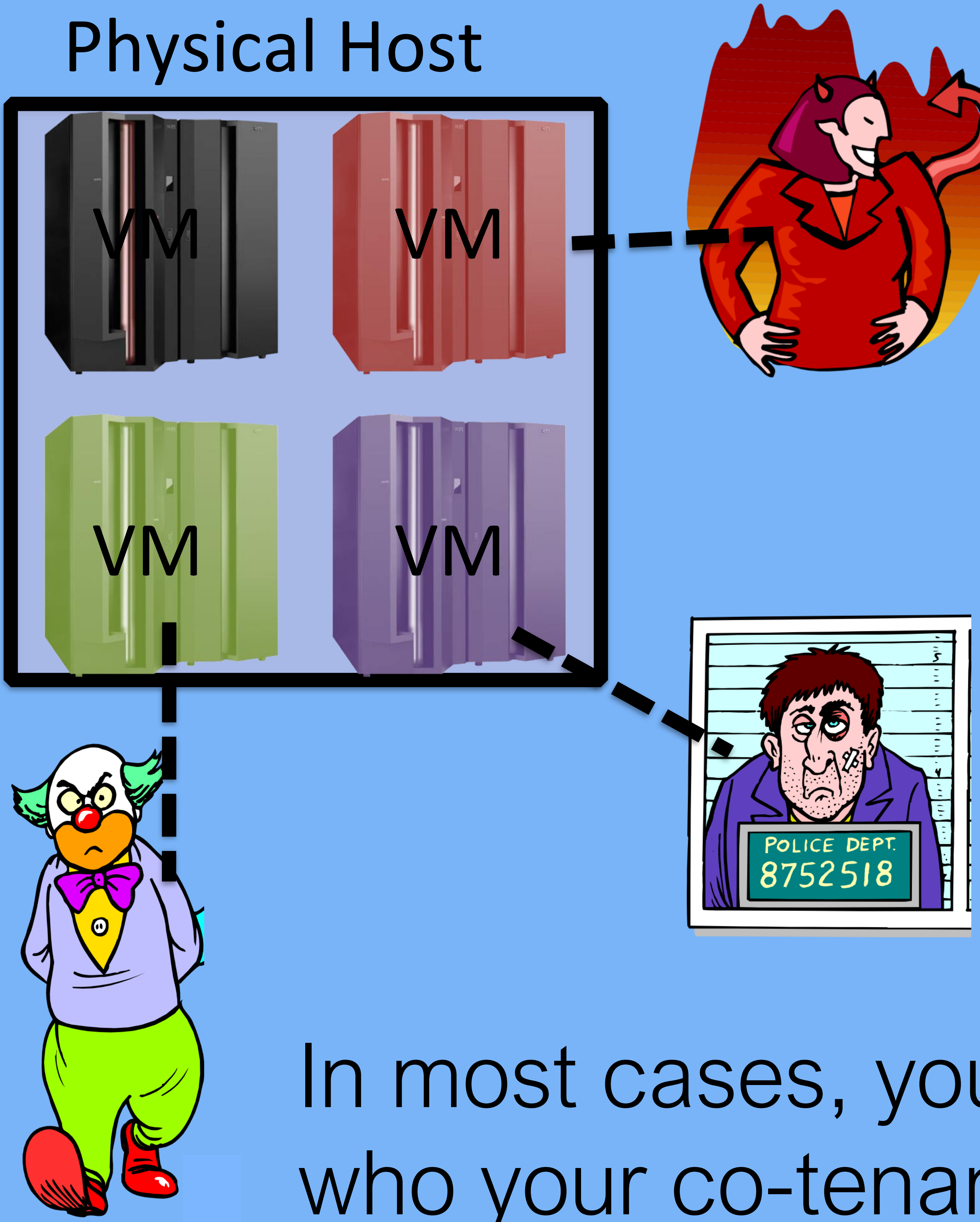
- Tenant stands up OS, applications, etc.—or uses a prerolled “machine image.”
- Tenant has illusion of exclusive computing resource ownership.



For efficient resource utilization in the cloud or in data centers, the VMs of multiple tenants may be packed into the same server / host.



Thus there may be VMs belonging to multiple tenants on the same physical server. This is called **co-residency.**



Physical Host

VM

VM

VM

VM

In most cases, you have no idea who your co-tenants are!

(Imagine an apartment building like this…)

The Larger Cloud: Multi-Tenancy

VM
VM
VM
VM
VM
VM
VM
VM
VM
VM
VM
VM
VM
VM
VM
VM
VM
VM

Main lesson today:
Sharing in the cloud
carries risks!

Reputation fate sharing

- Basic Idea: Misbehavior of some tenants can taint reputation of all.
- Example: What happens if co-tenants (not you) start sending spam?
- In 2009, Amazon EC2 tenants were abusing the ecosystem to send spam.
- Spamhaus (major spam tracking organization) blacklisted ***all*** Amazon.com EC2 IP addresses as spam originators!
- Legitimate users struggled to send e-mail.
- Amazon SES (Simple EMail Service) evolved as replacement
  - Contains throttling and reputation-based system
  - Filters ***outbound*** e-mail to prevent ***outgoing*** spam (in part to protect internal tenants)

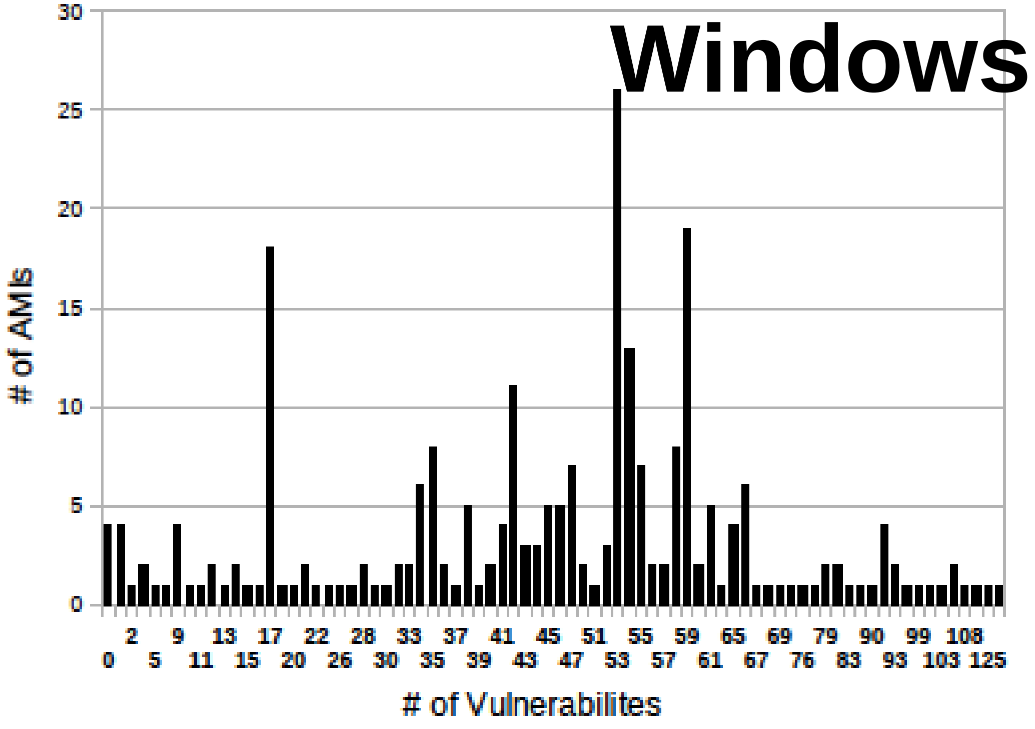
Amazon Machine Images (AMIs)

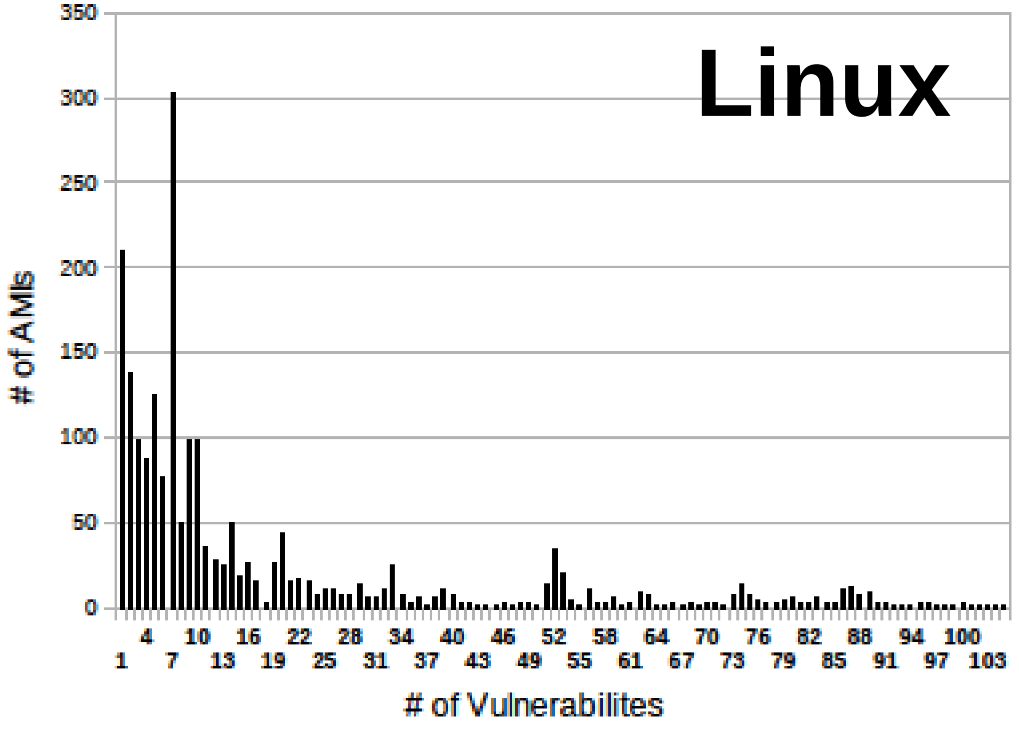
- User is wholly responsible for contents of VM
- Many public VM *images* made available in Amazon EC2 (OS + apps)
    - User-shared images
    - Provider-shared images for common needs
        - E.g., Ubuntu-based server image pre-configured with MySQL, PHP and Apache

[M. Balduzzi et al. A Security Analysis of Amazon's Elastic Compute Cloud Service, SAC, 2012.]

AMI vulnerabilities

- Large number of images have software over two years old
- Scan with Nessus revealed that
    - 98% of Windows AMIs and 58% of Linux have critical vulnerabilities!





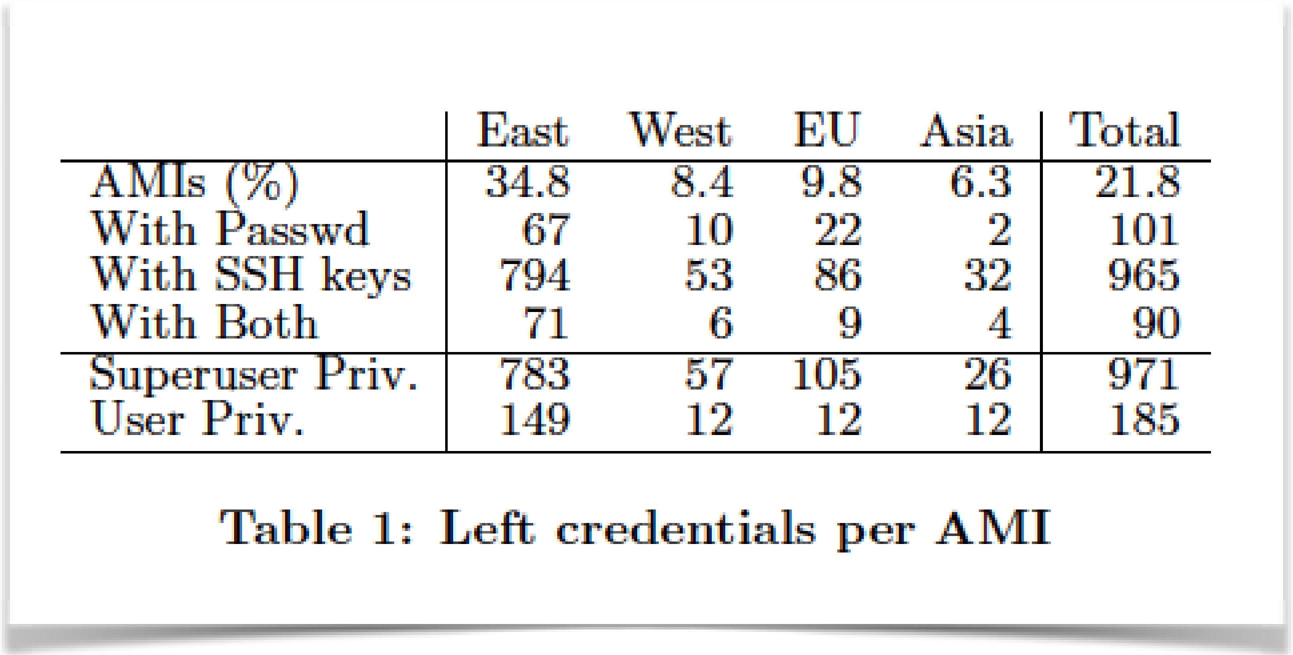
[M. Balduzzi et al. A Security Analysis of Amazon's Elastic Compute Cloud Service, SAC, 2012.]

Malware / unsolicited connections

- Two AMIs infected with malware
  - `Trojan-Spy` (variant 50112)
    - Keylogging, process monitoring, data exfiltration
  - `Trojan.Agent` (variant 173287)
    - Disappeared under reinspection
    - Seems to have become infected *while under study*
- Several Linux images sending *syslog* data to a remote host
  - Usually stored in `/var/log` and available only with administrative privileges

Leftover credentials

- Primary mechanism to connect to Linux server is SSH (Secure SHell)
- Many AMIs contain residual SSH credentials, private keys and/or passwords.
- Private keys still held by original owners
  - So original owners could SSH in!
- Passwords can be cracked!
  - Remember: Hashes viewable by anyone!
- Probably just a mistake…

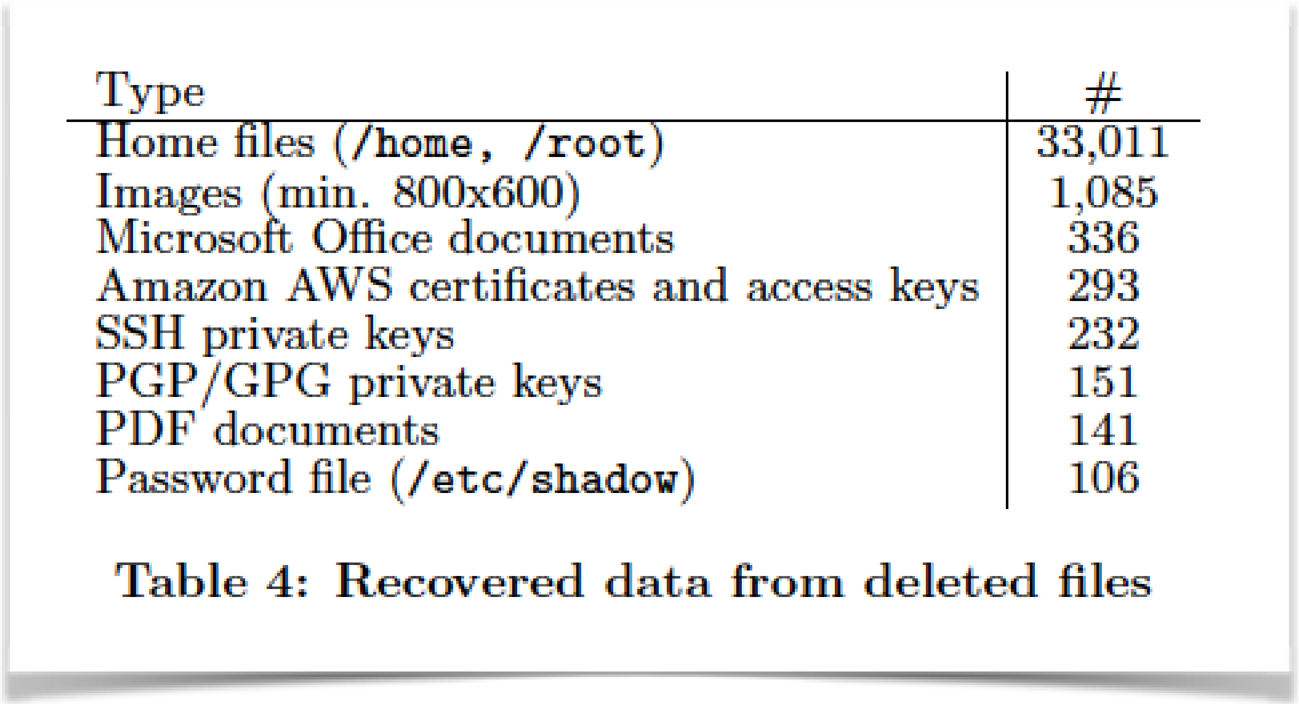


	East	West	EU	Asia	Total
AMIs (%)	34.8	8.4	9.8	6.3	21.8
With Passwd	67	10	22	2	101
With SSH keys	794	53	86	32	965
With Both	71	6	9	4	90
Superuser Priv.	783	57	105	26	971
User Priv.	149	12	12	12	185

Table 1: Left credentials per AMI

Residual private data

- 56 SSH *private* keys left in AMIs (54 unprotected)
  - Plus those in deleted files
- 187 AMIs contained 66,601 entries in `lastb` databases
  - Failed login attempts—including mistyped passwords
- 9 AMIs contained Firefox browsing history
- Of 1100 Linux AMIs, 98% contained deleted files recoverable via `extundelete`
  - From 6 to 40,000 files



Type	#
Home files (`/home, /root`)	33,011
Images (min. 800x600)	1,085
Microsoft Office documents	336
Amazon AWS certificates and access keys	293
SSH private keys	232
PGP/GPG private keys	151
PDF documents	141
Password file (`/etc/shadow`)	106

Table 4: Recovered data from deleted files

The adversarial mindset: Four key questions

1. **Security goal:** What policy or good state is meant to be enforced?
2. **Adversarial model:** Who is the adversary? What is the adversary’s space of possible actions?
3. **Mechanisms:** Are the right security mechanisms in place to achieve the security goal given the adversarial model?
4. **Incentives:** Will human factors and economics favor or disfavor the security goal?

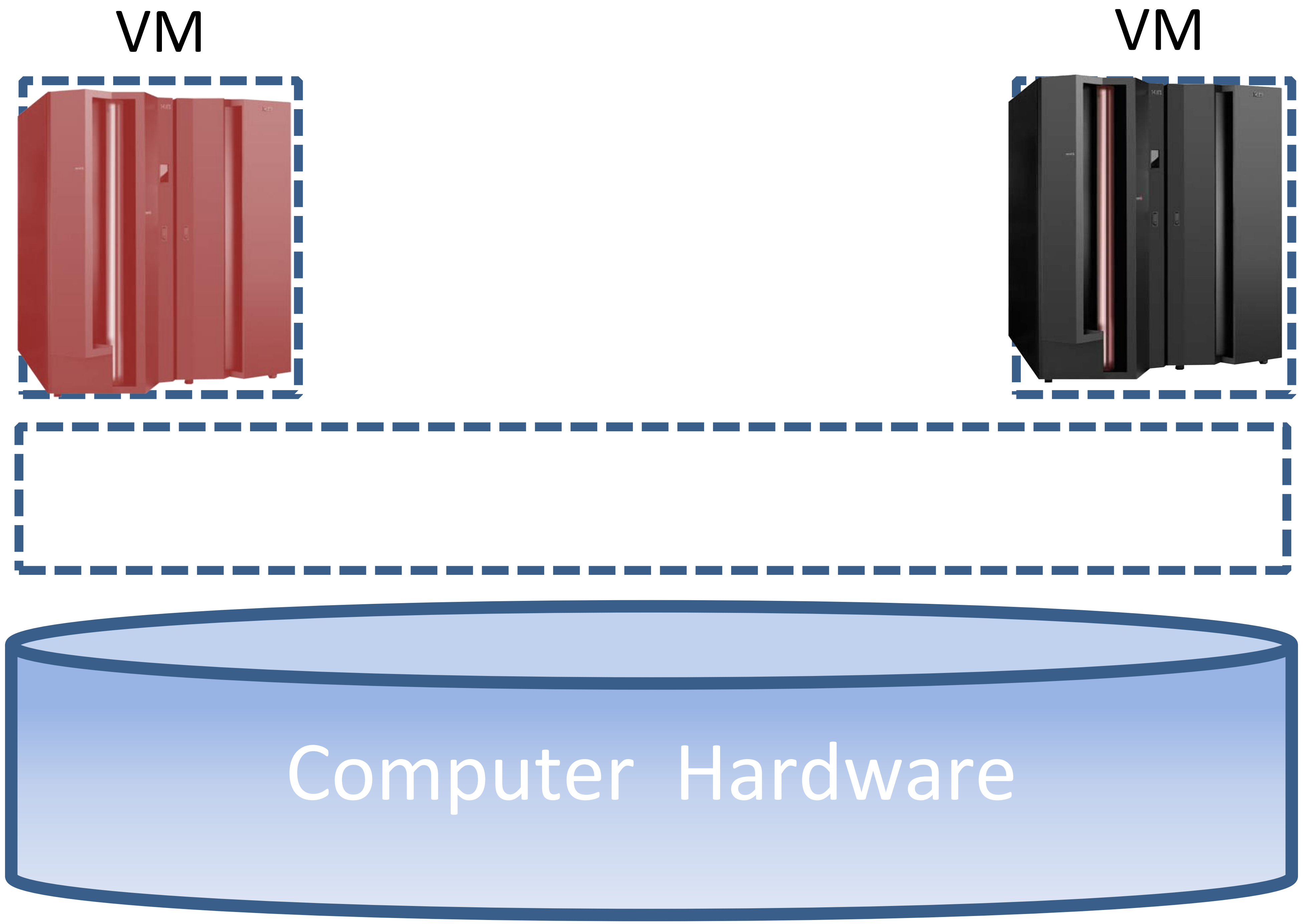
Side-channel attacks: Dangers of sharing hardware

Co-residency on a physical server

VM

VM

Co-residency on a physical server

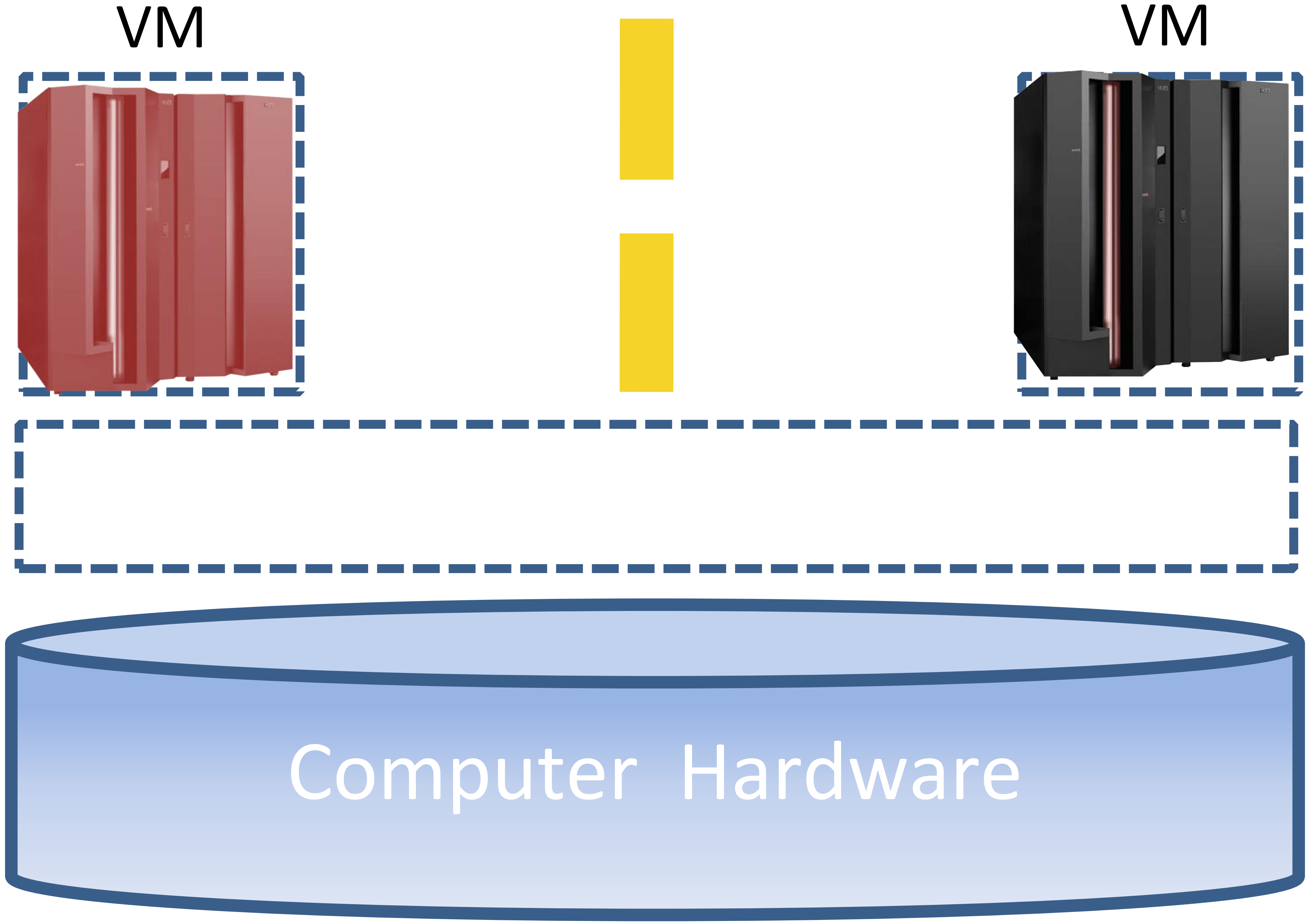


VM

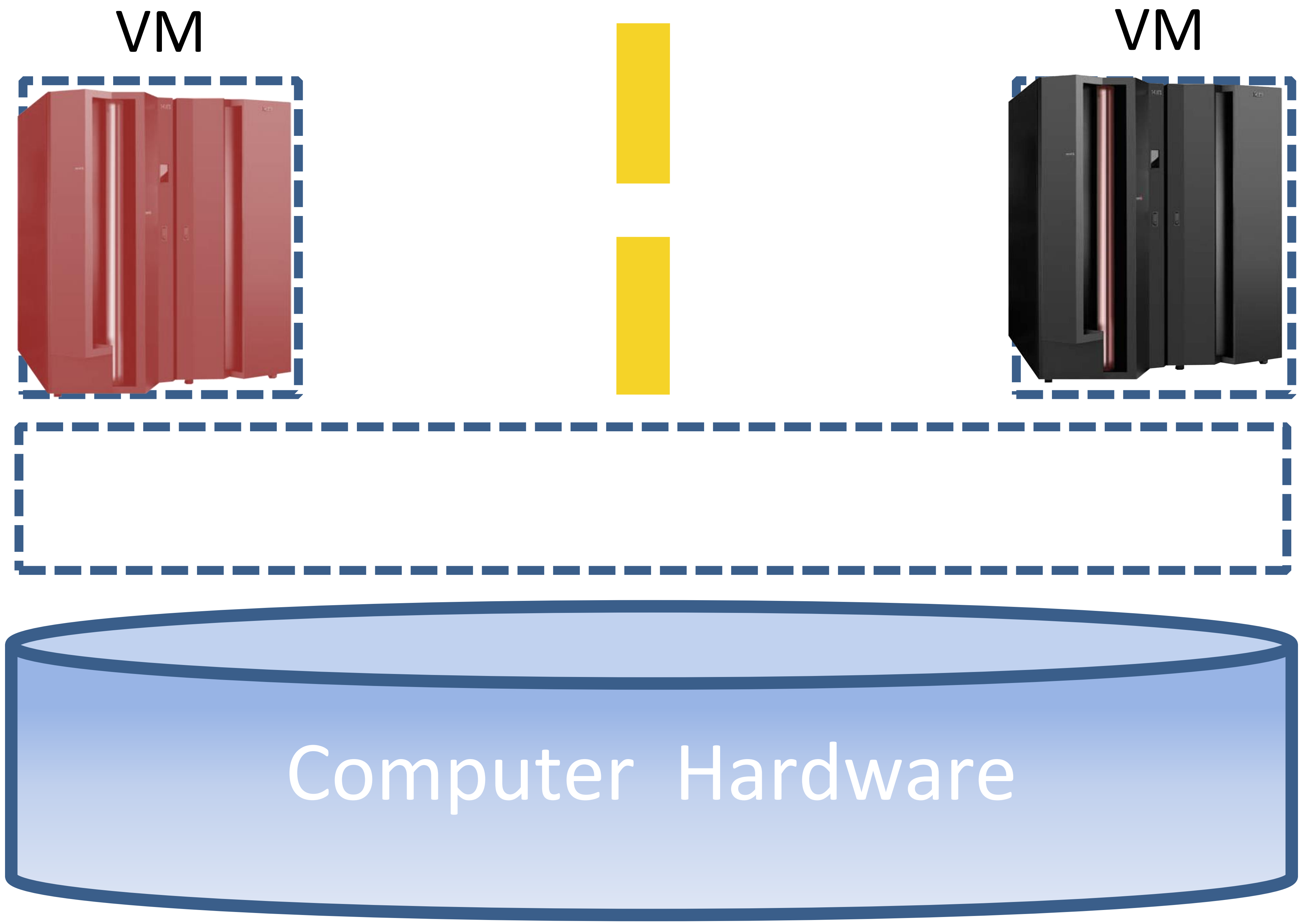
VM

Computer Hardware

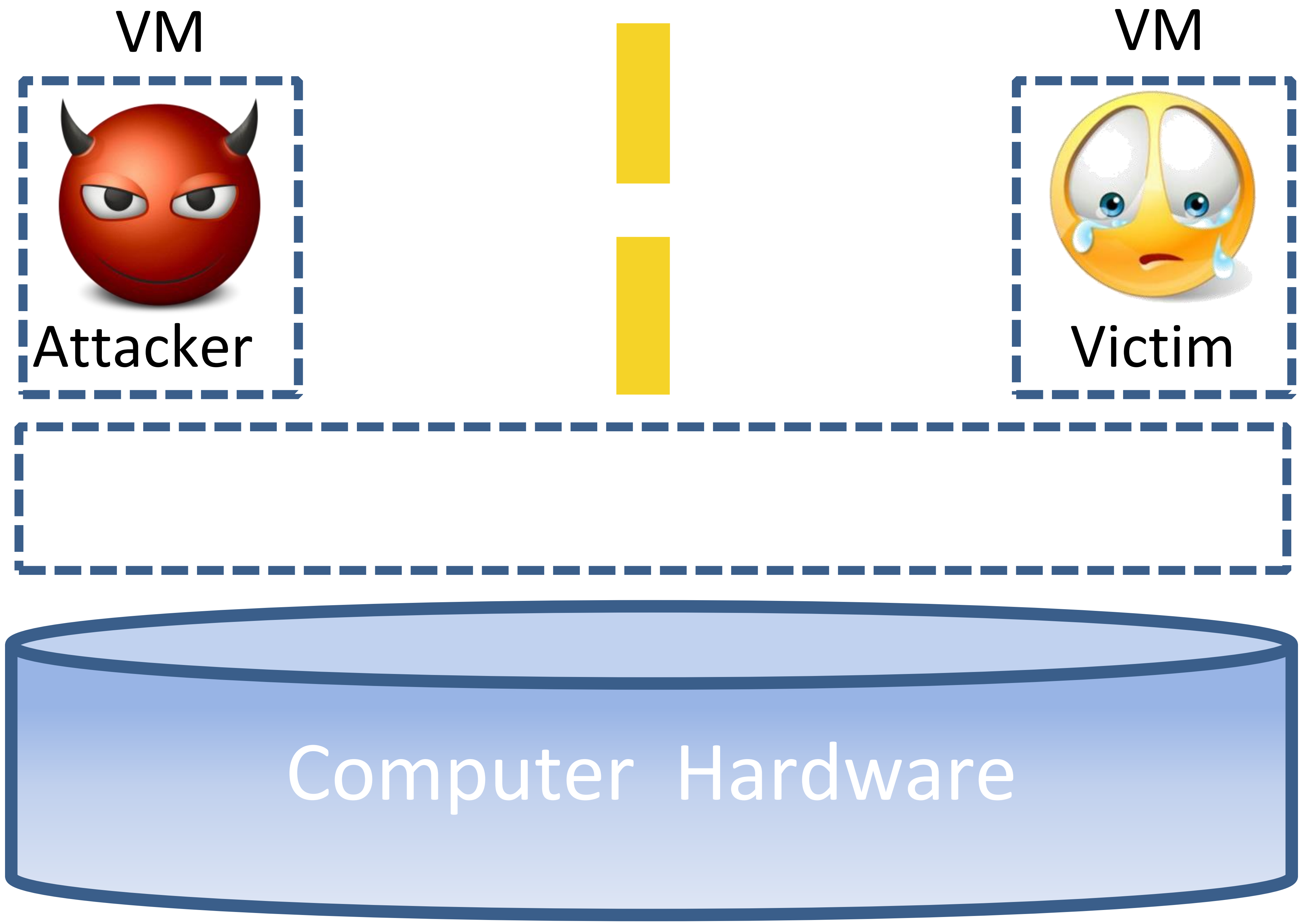
Security isolation by virtualization



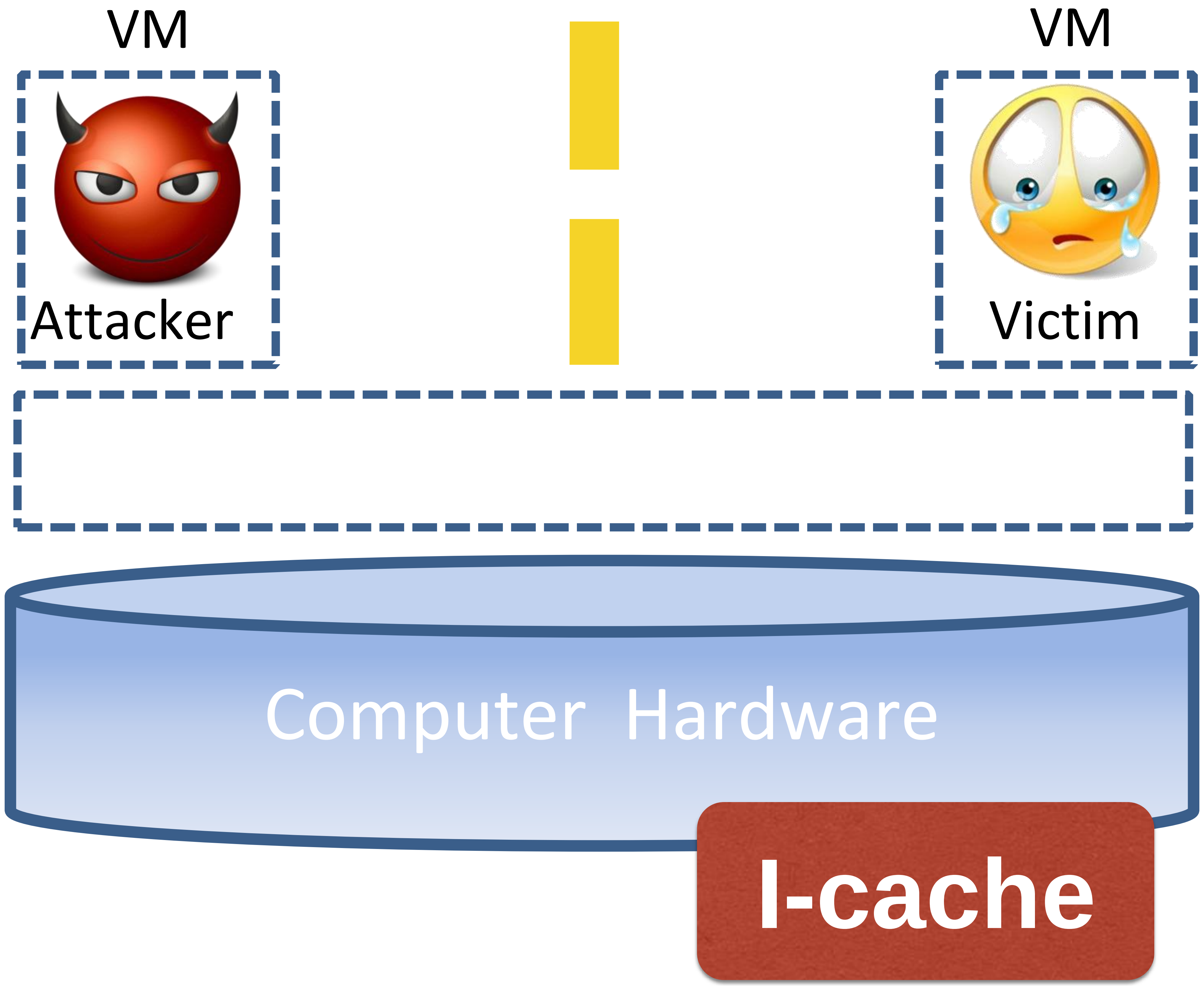
How strong is the isolation boundary?



How strong is the isolation boundary?



How strong is the isolation boundary?



VM

VM

Attacker

Victim

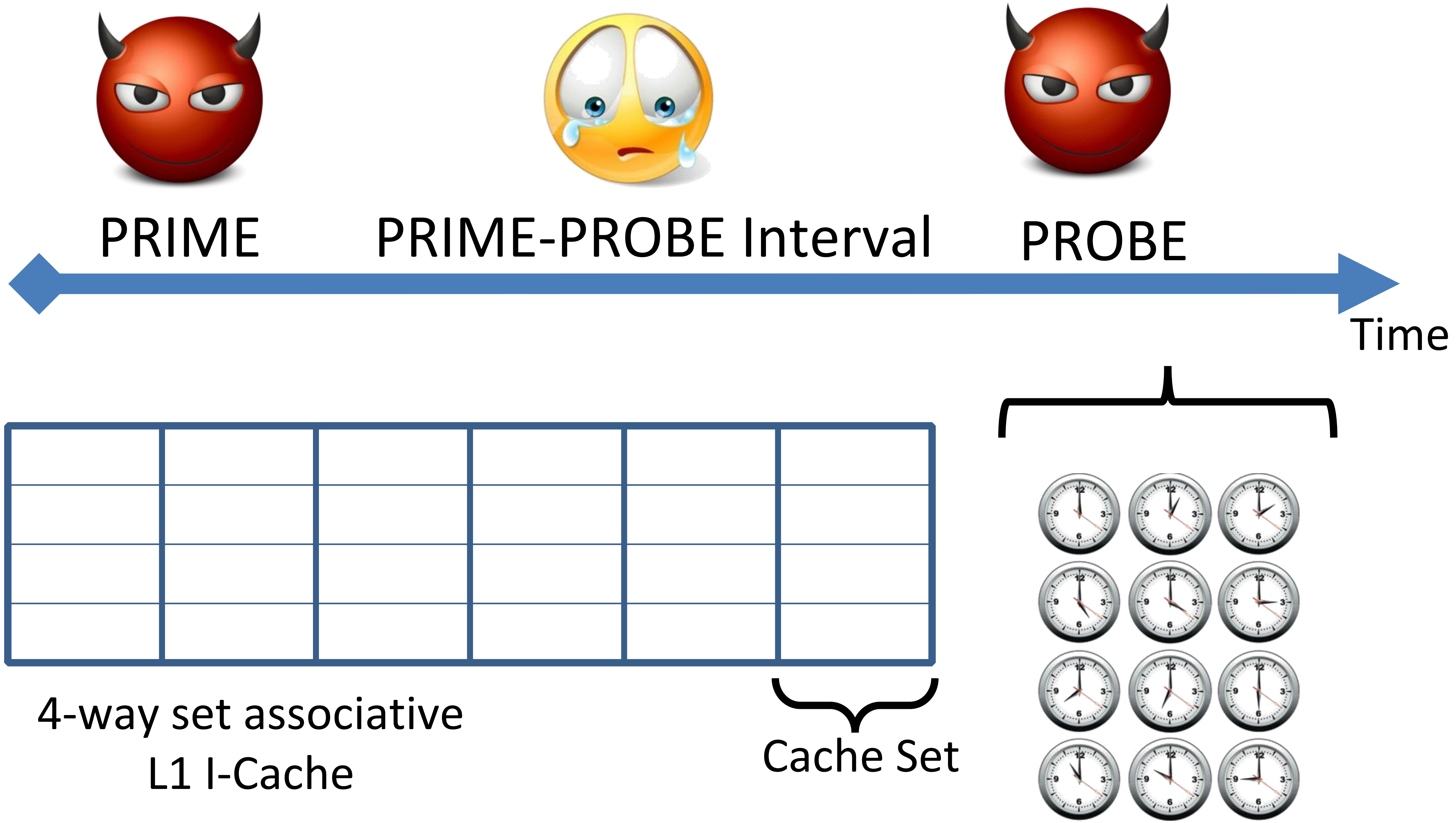
Computer Hardware

I-cache

Review of caching

- A cache is fast memory used for repeatedly fetched data / instructions.
  - Various strategies for placing and replacing cached objects
- Memory fetch results in *cache hit* when item is in cache.
  - Results in **fast retrieval**
- Memory fetch results in *cache miss* when item isn’t in cache.
  - Must seek in next level of memory; results in **slow retrieval**
- Attacker can exploit this difference to observe victim’s cache use!
  - Prime-Probe attack

Prime-Probe Protocol



PRIME

PRIME-PROBE Interval

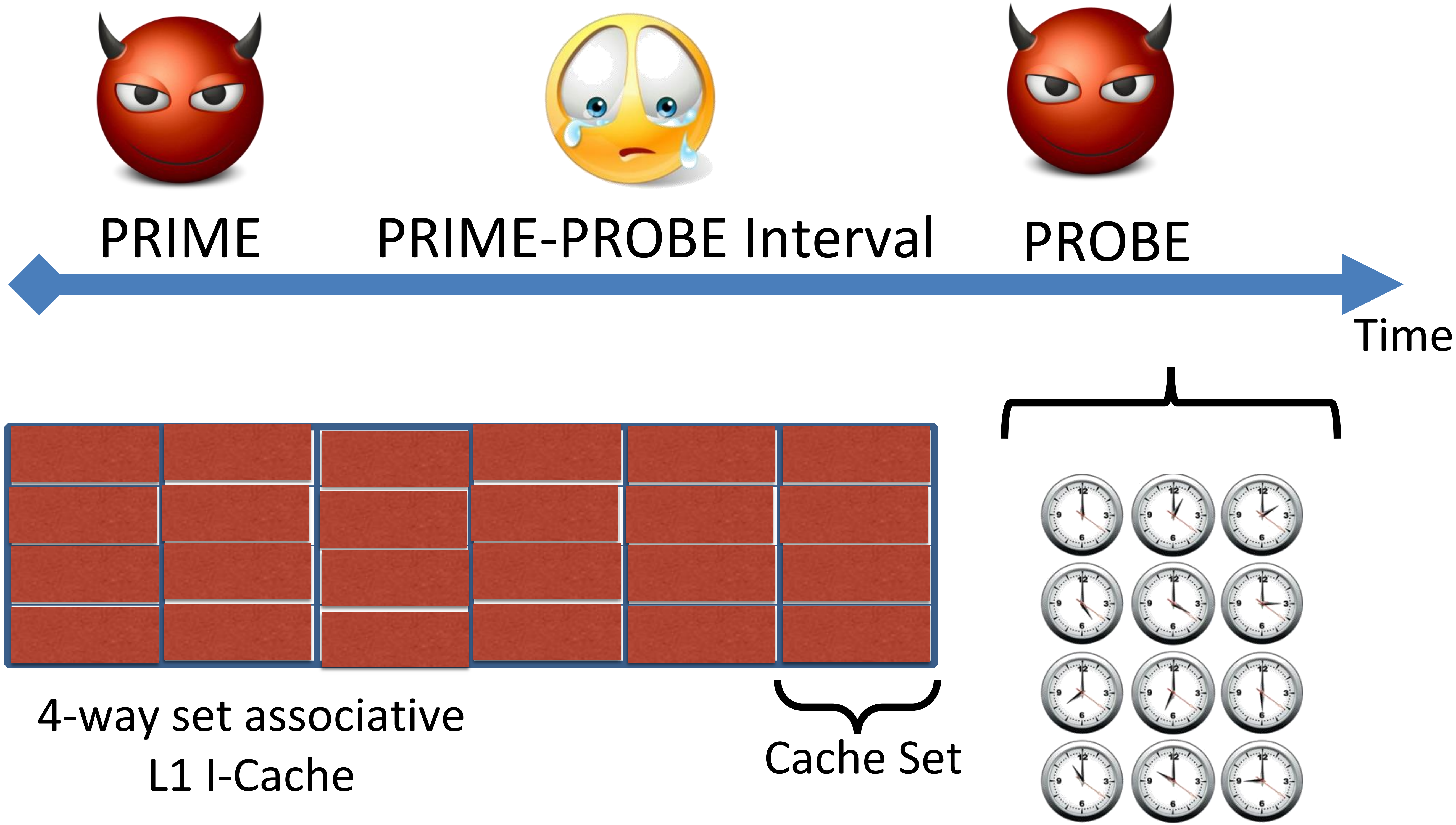
PROBE

Time

4-way set associative
L1 I-Cache

Cache Set

Prime-Probe Protocol



PRIME

PRIME-PROBE Interval

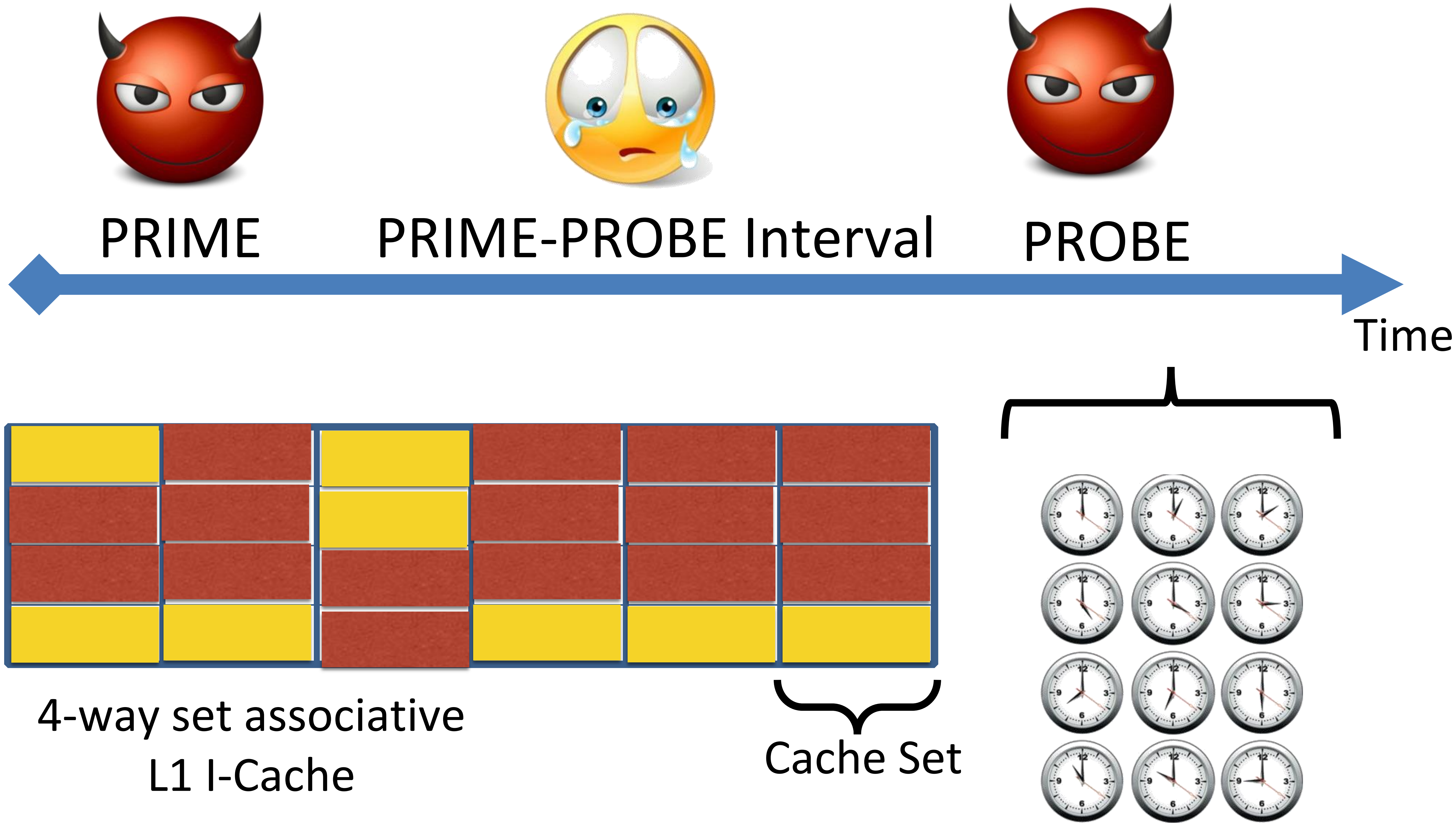
PROBE

Time

4-way set associative
L1 I-Cache

Cache Set

Prime-Probe Protocol



PRIME

PRIME-PROBE Interval

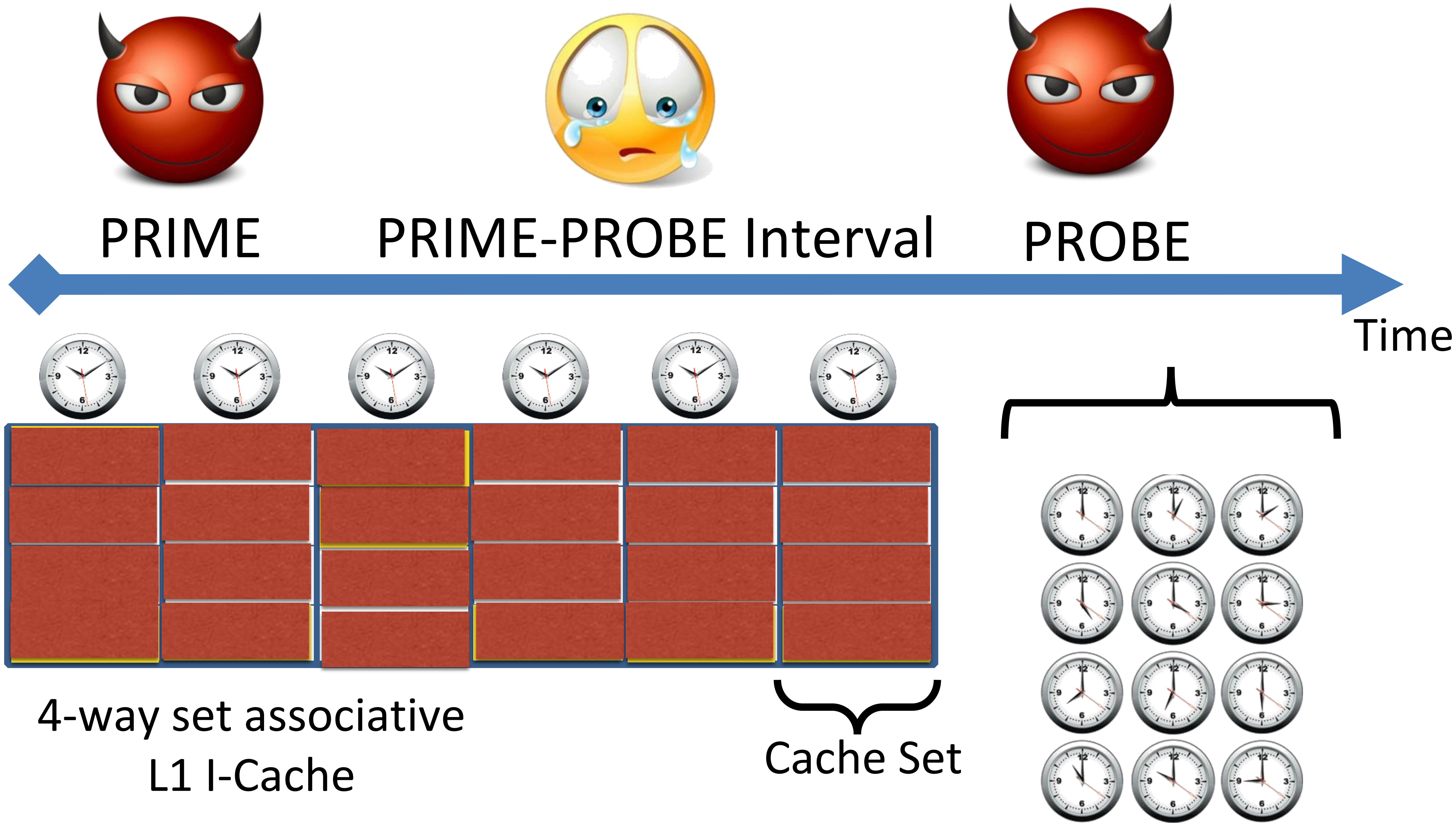
PROBE

Time

4-way set associative
L1 I-Cache

Cache Set

Prime-Probe Protocol



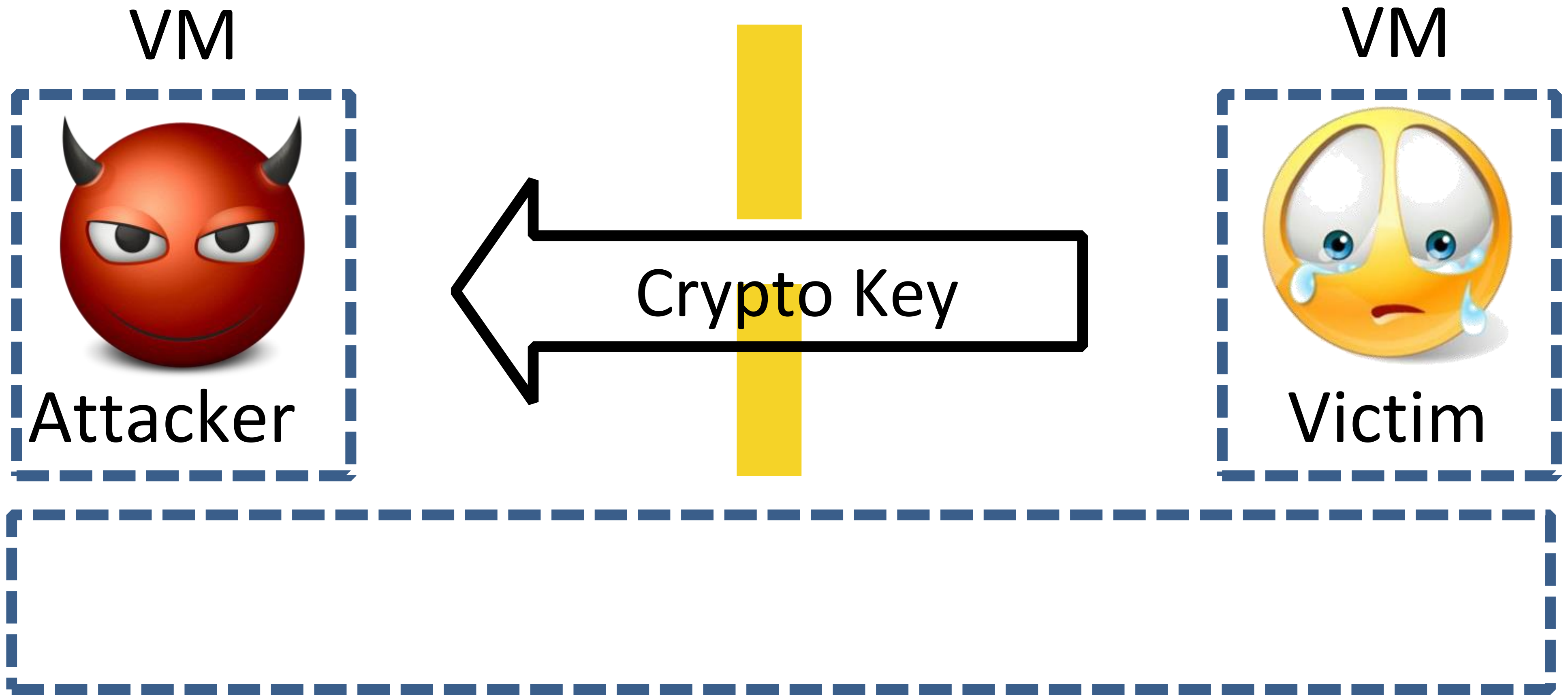
The upshot

- Adversary can use timing to learn victim activity in cache
- Adversary thus sees victim's cache footprint
  - Given knowledge of victim code, gets insight into what instructions have been executed
  - (Cache-set granularity)
- Adversary can repeatedly interrupt victim to measure footprint

Why does this matter?

- In textbook public-key crypto implementations, use of private key *SK* is performed in bitwise manner.
    - E.g., when *SK* used to decrypt message
- A ‘1’ bit in *SK* produces a different footprint in I-cache than a ‘0’ bit in *SK*.
- So when victim executes operation with private key *SK…*
    - 011001000100*…*
- An attacker can learn constituent bits and thus *SK*!

Side-channel attack



- Side-channel attacks of this type shown in lab
  - E.g., private key extraction (ElGamal) in EC2-like setup
- More powerful *flush-reload* attacks also possible
  - Yarom-Falkner '14

Why the cloud can be *good for security*

- The flip side of sharing is economies of scale.
- Some security benefits of cloud:
  - Large, specialized security team
  - Broad view of security events
    - E.g., online e-mail provider can see spam campaign
  - Ability to absorb denial-of-service attacks, e.g.,
    - In 2010 Amazon removed Wikileaks from EC2
    - “Hackivist” group Anonymous targeted Amazon in “Operation Payback”
      - Previously crashed MasterCard site and slowed Visa and PayPal
    - Amazon was virtually unaffected. Why?
    - Because holiday shopping traffic is essentially a denial-of-service attack! (Amazon has enormous spare capacity.)