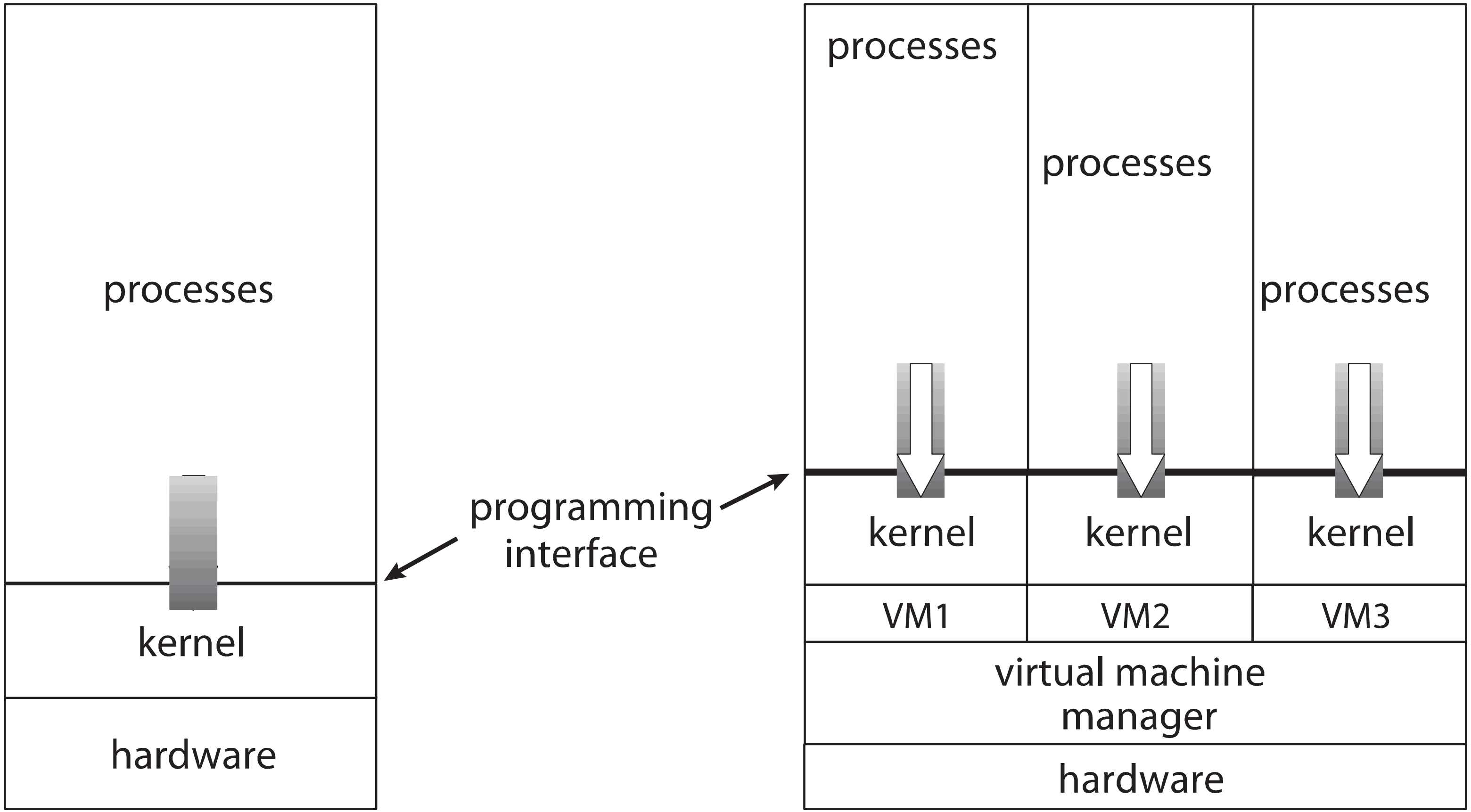
Threat Model for Cloud

Fabrizio Baiardi
f.baiardi@unipi.it

Syllabus

- Cloud Computing Introduction
  - Definitions
  - Economic Reasons
  - Service Model
  - Deployment Model
- Supporting Technologies
  - Virtualization Technology
  - Scalable Computing = Elasticity
- Security
  - New Threat Model
  - New Attacks
  - Countermeasures

System Models



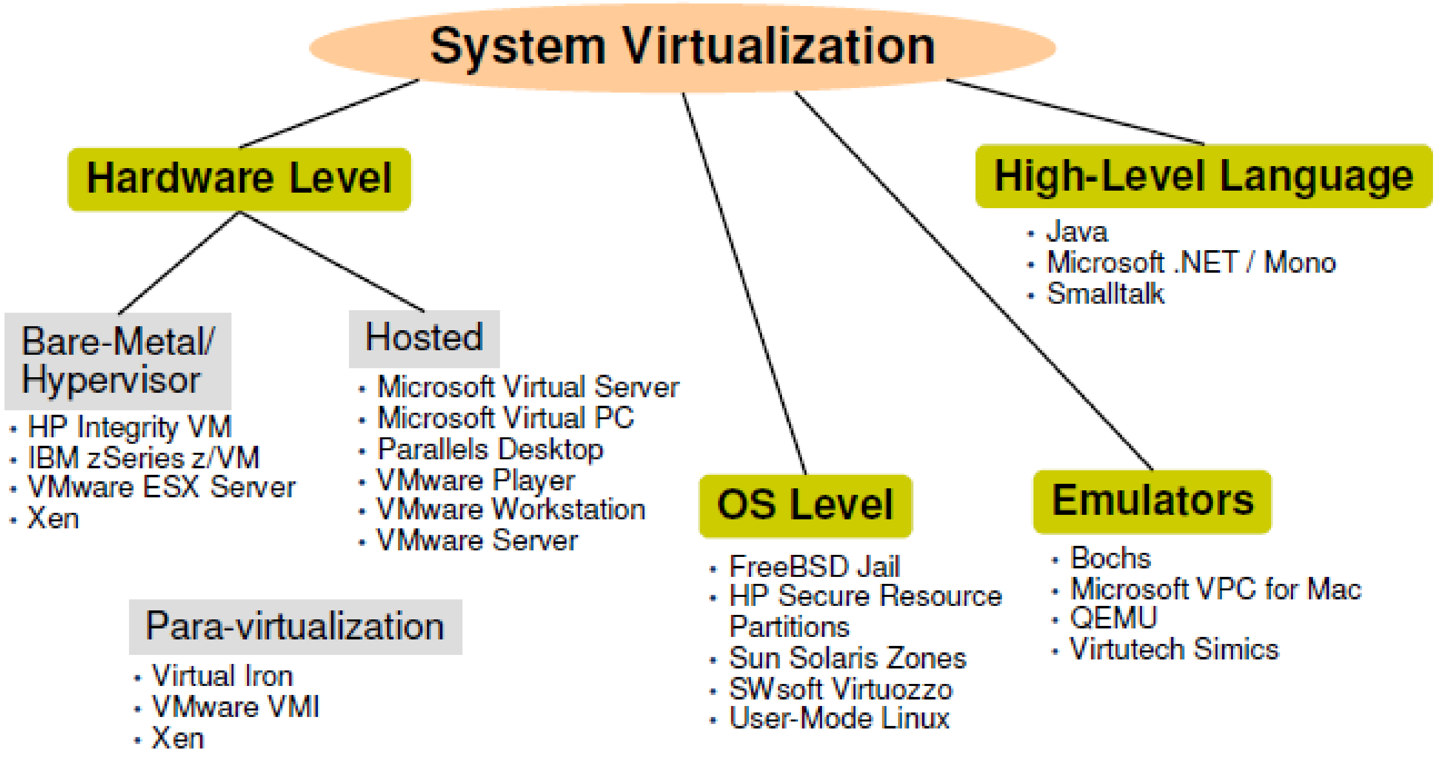
(a) Nonvirtual machine (b) Virtual machine

Implementation of VMMs

Vary greatly, with options including:

- **Type 0 hypervisors** - Hardware-based solutions that provide support for virtual machine creation and management via firmware
  - IBM LPARs and Oracle LDOMs are examples
- **Type 1 hypervisors** - Operating-system-like software built to provide virtualization
  - Including VMware ESX, Joyent SmartOS, and Citrix XenServer
- **Type 1 hypervisors** – Also includes general-purpose operating systems that provide standard functions as well as VMM functions
  - Including Microsoft Windows Server with HyperV and RedHat Linux with KVM
- **Type 2 hypervisors** - Applications that run on standard operating systems but provide VMM features to guest operating systems
  - VMware Workstation and Fusion, Parallels Desktop, and Oracle VirtualBox

Classification



System Virtualization

Hardware Level

Bare-Metal/ Hypervisor

- HP Integrity VM
- IBM zSeries z/VM
- VMware ESX Server
- Xen

Hosted

- Microsoft Virtual Server
- Microsoft Virtual PC
- Parallels Desktop
- VMware Player
- VMware Workstation
- VMware Server

Para-virtualization

- Virtual Iron
- VMware VMI
- Xen

OS Level

- FreeBSD Jail
- HP Secure Resource Partitions
- Sun Solaris Zones
- SWsoft Virtuozzo
- User-Mode Linux

Emulators

- Bochs
- Microsoft VPC for Mac
- QEMU
- Virtutech Simics

High-Level Language

- Java
- Microsoft .NET / Mono
- Smalltalk

Implementation of VMMs (cont.)

Other variations include:

- **Paravirtualization** - Technique in which the guest operating system is modified to work in cooperation with the VMM to optimize performance
- **Programming-environment virtualization** - VMMs do not virtualize real hardware but instead create an optimized virtual system
- **Emulators** – Allow applications written for one hardware environment to run on a different hardware environment, such as a different CPU
- **Application containment** - Not virtualization at all but rather provides virtualization-like features by segregating applications from the operating system, making them more secure, manageable
  - Including Oracle Solaris Zones, BSD Jails, and IBM AIX WPARs

Much variation due to breadth, depth and importance of virtualization in modern computing

Implementation of VMMs (cont.)

The Linux Containers (LXC) feature

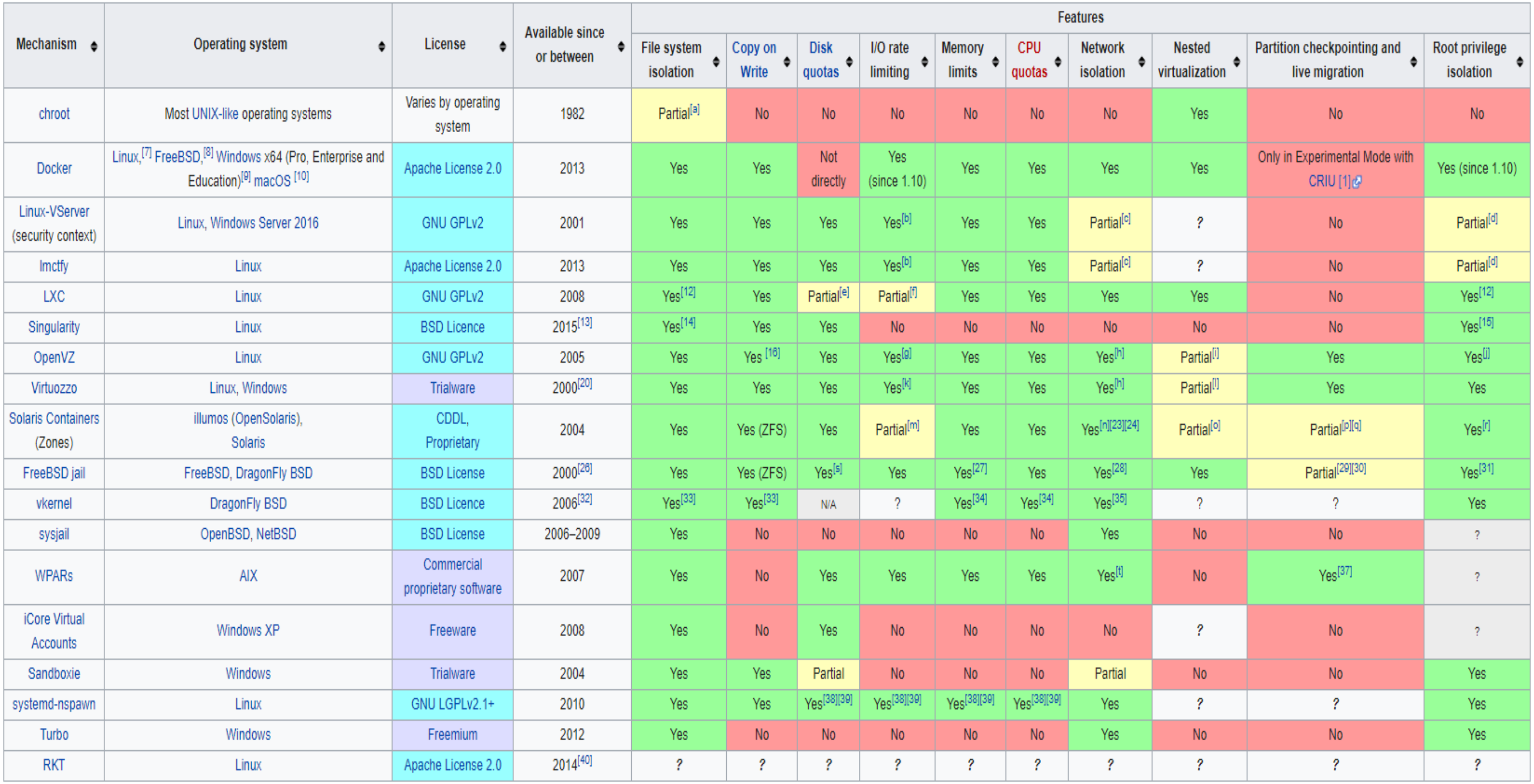
- a lightweight virtualization mechanism that does not require you to set up a virtual machine on an emulation of physical hardware.
- takes the cgroups resource management facilities as its basis and adds POSIX file capabilities to implement process and network isolation.
- You can run
  - a single application within a container (an application container) whose name space is isolated from the other processes on the system in a similar manner to a chroot jail.
  - a complete copy of the Linux operating system in a container (a system container) without the overhead of running a level-2 hypervisor such as VirtualBox.
  - the container shares the kernel with the host, so its processes and file system are completely visible from the host but from the container, you only see its file system and process space.

Container vs VMM

VMM = virtualization software. It mimica hardware features to isolate system resources, such as CPU cores, memory. A virtualized system believes not just that it has root, but that it has ring 0 (kernelmode). Hardware is either abstracted by the CPU or emulated by the hypervisor software (NICs). Because the guest believes it owns the whole system, anything that run on an physical architecture will tend to also run in the VM

Containers = rather than using virtualization, they use namespaces. Each container has every resource put in its own namespace, and it can run an independent operating system. The container init process on the container sees itself as PID 1 running as root, but the host sees it as just another non-init and non-root PID. Container share the host's kernel and can only run the same type of operating system as the host. Additionally, containers can have root processes that can do privileged actions but cannot change global kernel settings that would affect all containers.

Several solution



Mechanism	Operating system	License	Available since or between	Features									
				File system isolation	Copy on Write	Disk quotas	I/O rate limiting	Memory limits	CPU quotas	Network isolation	Nested virtualization	Partition checkpointing and live migration	Root privilege isolation
chroot	Most UNIX-like operating systems	Varies by operating system	1982	Partial[a]	No	No	No	No	No	No	Yes	No	No
Docker	Linux,[7] FreeBSD,[8] Windows x64 (Pro, Enterprise and Education)[9] macOS [10]	Apache License 2.0	2013	Yes	Yes	Not directly	Yes (since 1.10)	Yes	Yes	Yes	Yes	Only in Experimental Mode with CRIU [1]	Yes (since 1.10)
Linux-VServer (security context)	Linux, Windows Server 2016	GNU GPLv2	2001	Yes	Yes	Yes	Yes[b]	Yes	Yes	Partial[c]	?	No	Partial[d]
lmctfy	Linux	Apache License 2.0	2013	Yes	Yes	Yes	Yes[b]	Yes	Yes	Partial[c]	?	No	Partial[d]
LXC	Linux	GNU GPLv2	2008	Yes[12]	Yes	Partial[e]	Partial[f]	Yes	Yes	Yes	Yes	No	Yes[12]
Singularity	Linux	BSD Licence	2015[13]	Yes[14]	Yes	Yes	No	No	No	No	No	No	Yes[15]
OpenVZ	Linux	GNU GPLv2	2005	Yes	Yes [16]	Yes	Yes[g]	Yes	Yes	Yes[h]	Partial[i]	Yes	Yes[j]
Virtuozzo	Linux, Windows	Trialware	2000[20]	Yes	Yes	Yes	Yes[k]	Yes	Yes	Yes[h]	Partial[l]	Yes	Yes
Solaris Containers (Zones)	illumos (OpenSolaris), Solaris	CDDL, Proprietary	2004	Yes	Yes (ZFS)	Yes	Partial[m]	Yes	Yes	Yes[n][23][24]	Partial[o]	Partial[p][q]	Yes[r]
FreeBSD jail	FreeBSD, DragonFly BSD	BSD License	2000[26]	Yes	Yes (ZFS)	Yes[s]	Yes	Yes[27]	Yes	Yes[28]	Yes	Partial[29][30]	Yes[31]
vkernel	DragonFly BSD	BSD Licence	2006[32]	Yes[33]	Yes[33]	N/A	?	Yes[34]	Yes[34]	Yes[35]	?	?	Yes
sysjail	OpenBSD, NetBSD	BSD License	2006–2009	Yes	No	No	No	No	No	Yes	No	No	?
WPARs	AIX	Commercial proprietary software	2007	Yes	No	Yes	Yes	Yes	Yes	Yes[t]	No	Yes[37]	?
iCore Virtual Accounts	Windows XP	Freeware	2008	Yes	No	Yes	No	No	No	No	?	No	?
Sandboxie	Windows	Trialware	2004	Yes	Yes	Partial	No	No	No	Partial	No	No	Yes
systemd-nspawn	Linux	GNU LGPLv2.1+	2010	Yes	Yes	Yes[38][39]	Yes[38][39]	Yes[38][39]	Yes[38][39]	Yes	?	?	Yes
Turbo	Windows	Freemium	2012	Yes	No	No	No	No	No	Yes	No	No	Yes
RKT	Linux	Apache License 2.0	2014[40]	?	?	?	?	?	?	?	?	?	?

Types of Virtual Machines and Implementations

Many variations as well as HW details

- Assume VMMs take advantage of HW features
  - HW features can simplify implementation, improve performance

Whatever the type, a VM has a lifecycle

- Created by VMM
- Resources assigned to it (number of cores, amount of memory, networking details, storage details)
- In type 0 hypervisor, resources usually dedicated
- Other types dedicate or share resources, or a mix
- When no longer needed, VM can be deleted, freeing resouces simpler, faster than with a physical machine install
- Can lead to **virtual machine sprawl** with lots of VMs, history and state difficult to track and manage

Types of VMs – Type 0 Hypervisor

Old idea, under many names by HW manufacturers

- “partitions”, “domains”
- A HW feature implemented by firmware
- OS need to nothing special, VMM is in firmware
- Smaller feature set than other types
- Each guest has dedicated HW

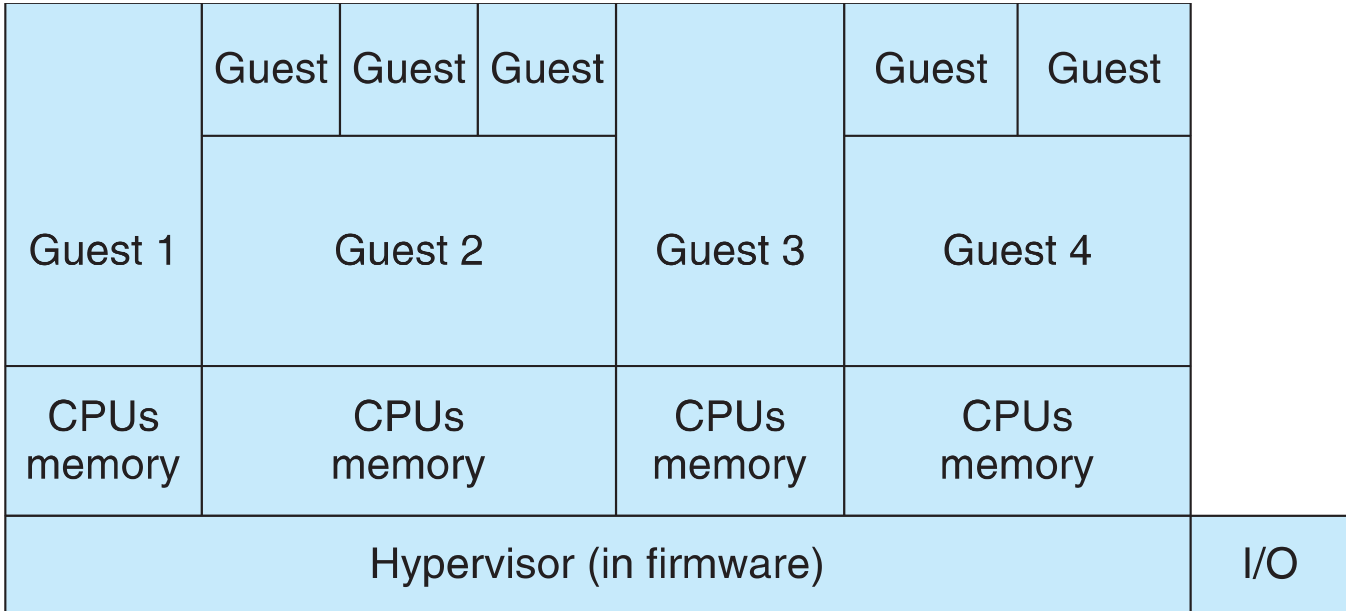
I/O a challenge as difficult to have enough devices, controllers to dedicate to each guest

Sometimes VMM implements a **control partition** running daemons that other guests communicate with for shared I/O

Can provide virtualization-within-virtualization (guest can be a VMM)

- Other types have difficulty doing this

Type 0 Hypervisor



Guest Guest Guest

Guest Guest

Guest 1

Guest 2

Guest 3

Guest 4

CPUs memory

CPUs memory

CPUs memory

CPUs memory

Hypervisor (in firmware)

I/O

Types of VMs – Type 1 Hypervisor

Commonly found in company datacenters

- In a sense becoming “datacenter operating systems”
  - Datacenter managers control and manage OSes in new, sophisticated ways by controlling the Type 1 hypervisor
  - Consolidation of multiple OSes and apps onto less HW
  - Move guests between systems to balance performance
  - Snapshots and cloning

Special purpose operating systems that run natively on HW

- Rather than providing system call interface, create run and manage guest OSes
- Can run on Type 0 hypervisors but not on other Type 1s
- Run in kernel mode
- Guests generally don’t know they are running in a VM
- Implement device drivers for host HW because no other component can
- Also provide other traditional OS services like CPU and memory management

Types of VMs – Type 1 Hypervisor (cont.)

Another variation is a general purpose OS that also provides VMM functionality

- RedHat Enterprise Linux with KVM, Windows with Hyper-V, Oracle Solaris
- Perform normal duties as well as VMM duties
- Typically less feature rich than dedicated Type 1 hypervisors

In many ways, treat guests OSes as just another process

- Albeit with special handling when guest tries to execute special instructions

Types of VMs – Type 2 Hypervisor

Less interesting from an OS perspective

- Very little OS involvement in virtualization
- VMM is simply another process, run and managed by host
  - Even the host doesn't know they are a VMM running guests
- Tend to have poorer overall performance because can't take advantage of some HW features
- But also a benefit because require no changes to host OS
  - Student could have Type 2 hypervisor on native host, run multiple guests, all on standard host OS such as Windows, Linux, MacOS

Types of VMs – Paravirtualization

Does not fit the definition of virtualization – VMM not presenting an exact duplication of underlying hardware

- But still useful!
- VMM provides services that guest must be modified to use
- Leads to increased performance
- Less needed as hardware support for VMs grows

Xen, leader in paravirtualized space, adds several techniques

- For example, clean and simple device abstractions
  - Efficient I/O
  - Good communication between guest and VMM about device I/O
  - Each device has circular buffer shared by guest and VMM via shared memory

Before the threats, the assets

- When moving your application and data to the cloud, you no longer have to protect resources (processor, memory, networks)
- Instead you have to protect your information
- Information-centric security binds security directly to information and the people who access it to ensure that they can access only the right information at the right time, when and where they need it
- All the assets you have to protect are virtual
- The cloud provider, instead, has to protect the physical assets
- No perimeter to be defended

Threat models and cloud migration: before

- They (evil, the external threat)
- Us (good ones)
- Us (the bad ones, insider threat)
- The defence is based upon
  - Preventing an attacker from entering into the system (firewall)
  - Discovering insider behaviours that violate the security policy or that have defeated the firewall (host & network IDS)
- The approach can be more detailed (e.g. defence in depth) but the distinction based upon a perimeter is always there

Threat models and cloud migration: after

- They (evil, the external threat)
- Us (good ones)
- Us (the bad ones, insider threat)
- We share the physical infrastructure, our application, our data with the evil, the cloud provider can be evil
- Every threat may become now also an insider threat
- The virtualization support has
  - to implement VMs
  - share physical resources among them
  - confine anomalous and dangerous users from good ones
- The attack surface of the cloud computing system increases at it includes
  - The VMM
  - The browser that is used to interact with the cloud.

How much control on them?

- Private cloud
  - used by a single organization eg a university, a company
  - good control
- Community cloud
  - Used by a set of organizations that share the same problems eg several hospitals
  - Acceptable control
- Public cloud
  - No control

Extended Attack Surface

- The attack surface of a system includes all the components that can be the target of an initial attack of a threat that enables the threat to reach some goals
  - Entry points
  - Exit points
  - Channels
- The notion of surface makes it possible to evaluate the percentage of a system that is exposed to threat attacks = how many initial attacks with respect to all attack
- The attack surface of a cloud system is larger than the one of an equivalent stand alone system

New classes of attacks

A system where legal user and attacker share the same architecture is the target of new attacks that

- discover and monitor the flows of information
  - among VMs, application, platforms
  - between the browser and the cloud
- discover the allocation of VMs onto physical nodes to deduce the physical resources shared among VMs =

  cloud cartography
- control the user browser to control and manipulate the cloud resources with respect to those attacks that steal info of a browser and so on, now the goal is controlling those resources that are accessed through the browser

However, there is a much larger amount of cheap processing power, Can this power be helpful for the good guys? How?

SaaS Maturity Levels

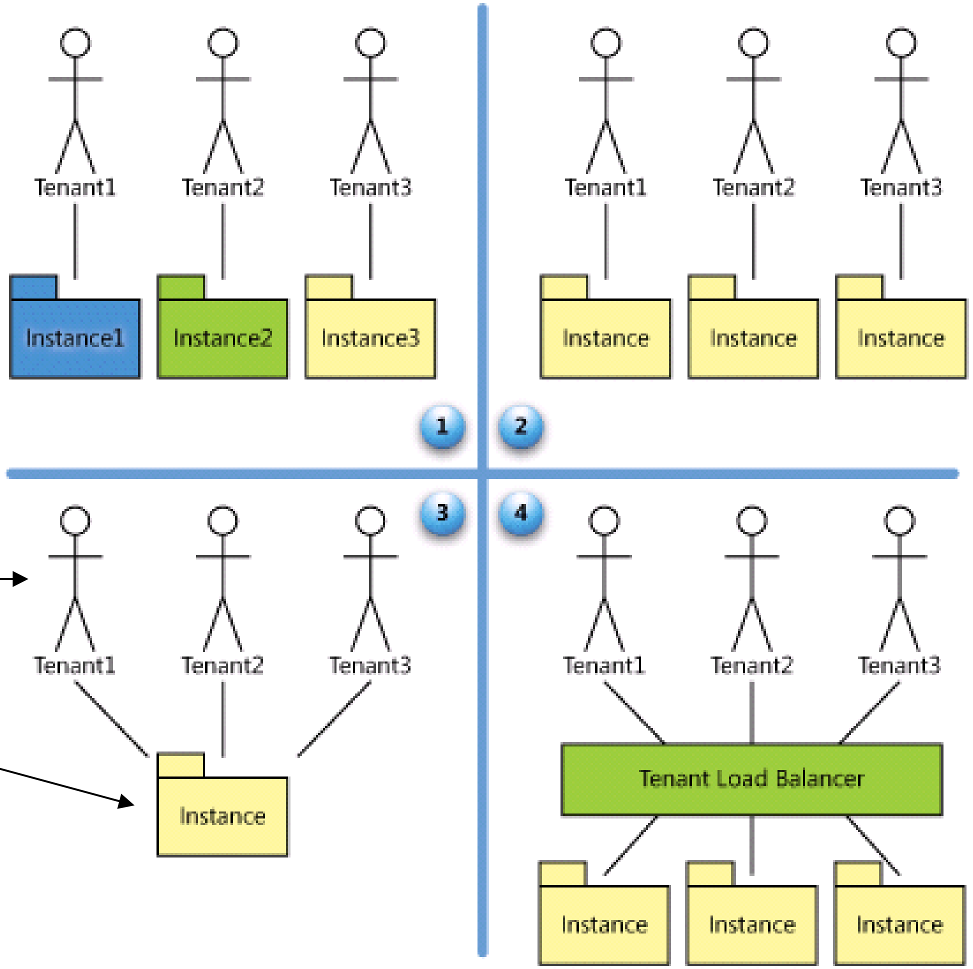
Level 1: Ad-Hoc/Custom

Level 2: Configurable

Level 3: Configurable, Multi-Tenant-Efficient

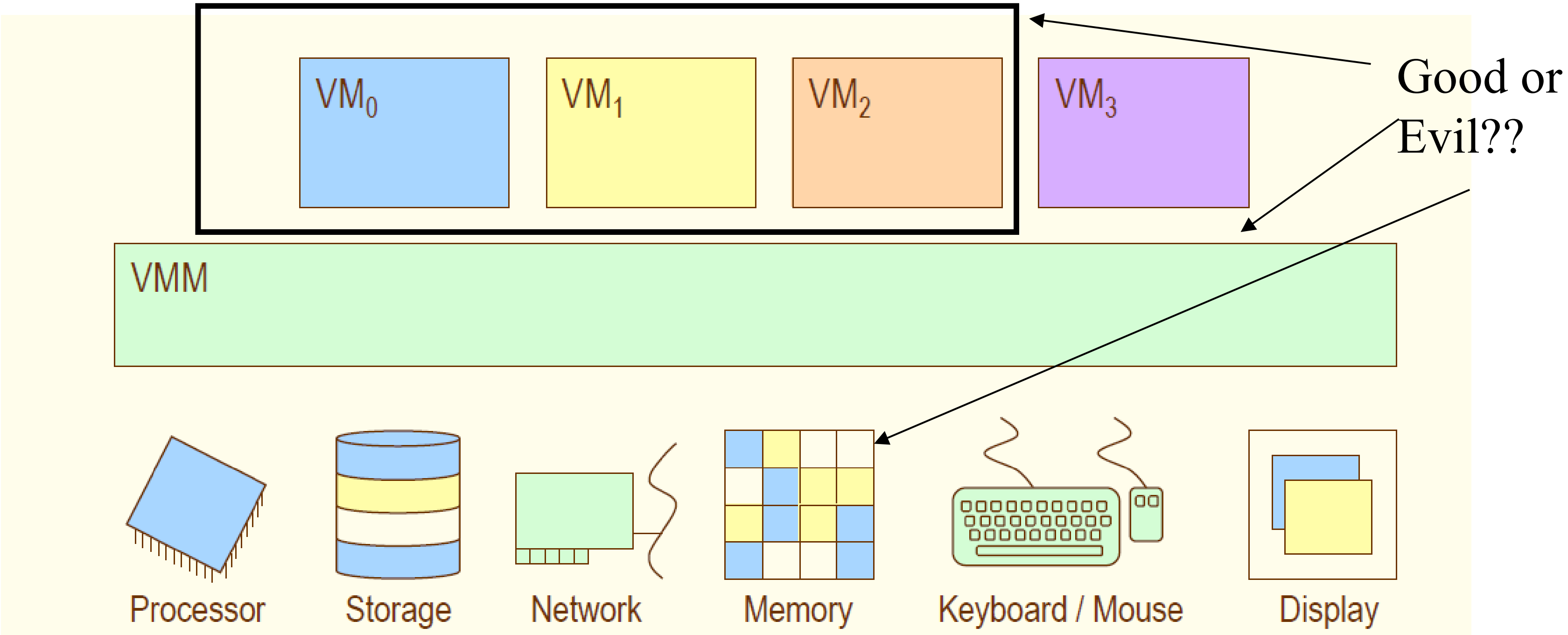
Level 4: Scalable, Configurable, Multi-Tenant-Efficient

Good or Evil?



Source: Microsoft MSDN Architecture Center

All together ????



- VMM applies all 3 sharing methods, as needed, to create illusion of platform ownership to each guest OS

New resource availables for attacker

- A cloud is an interesting target for several threat such as terrorist or organized crime
- An agent attacking a cloud can access a much larger amount of resources, know how, processing power than typical attackers
- A SME that has to face a trade off
  - Better security offered by the provider
  - More powerful attackers
- The large amount of cheap processing power that cloud systems made available simplify the implementation of brute force attacks, e.g. exhaustive key searches

The cloud provider

- It is a new threat to be considered
- Why so few papers discuss this threat ??? :-D
- The impact of a provider attack is highly critical because of the kind of access to physical and logical resources
- There are problems to be considered independently of malicious planned attack
  - Lock in with a provider in the case of SaaS
  - It is almost impossible to have some assurance that data that has been stored by a provider have been erased
- With respect to other threats the provider is known, hence we simply need to prove a misbehavior rather than detailing which misbehaviour the provider has been involved in

The cloud provider

- A Service Level Agreement (contract) has to be signed to define
  - The amount of resources that will be available
  - Largest downtime that is acceptable
  - Geographical location of the data
  - Handling of sensible data
  - Use of encryption
  - Security policy of the provider
- It is important to include only properties that can be measured and hence checked= If a decision cannot be made as the result of consuming a given metric, ask yourself why you’re tracking it
- Automated checks should be preferred to simplify the implementation and increase the number of checks that are executed for the same cost

The cloud provider

- It is important to include only those properties that can be measured and hence can be checked
-

Cloud Vulnerabilities

What follows is a long(and tedious) list of vulnerabilities

Some of them will be discussed in the following, other ones are to be remembered to check the provider since there are no new countermeasures

Authentication Vulnerabilities

Access and Autentication

- Insecure storage of cloud access credentials by customer
- Insufficient roles available
- Credentials stored on a transitory machine
- Password-based authentication may become insufficient
  - Strong or two-factor authentication for accessing cloud resources will be necessary

Authentication Vulnerabilities

- Identity of customer or billing information is not adequately verified at registration
- Delays in synchronization between cloud system components
- Multiple, unsynchronized copies of identity data are made
- Credentials are vulnerable to interception and replay
- De-provisioned credentials are still valid due to time delays in roll-out of revocation

Resource Vulnerabilities

Inaccurate Modeling of Resource Usage

- Overbooking or over-provisioning
- Failure of resource allocation algorithms due to extraordinary events (e.g., outlying news events for content delivery).
- Failure of resource allocation algorithms using job or packet classification because resources are poorly classified.
- Failures in overall resource provisioning (as opposed to temporary overloads)

No resource capping

- If there is not a flexible and configurable way for the customer and/or the cloud provider to set limits on resources, this can be problematic when resource use is unpredictable.

Inadequate Resource Provisioning and Investments in Infrastructure

- Infrastructure investments take time. If predictive models fail, the cloud provider service can fail for a long period.

Vulnerabilities

Remote Access To Management Interface

- Allows vulnerabilities in end-point machines to compromise the cloud infrastructure (single customer or CP) through, for example, weak authentication of responses and requests

Hypervisor

- Exploiting the hypervisor potentially means exploiting every VM!
- Guest to host escape: A user defeat isolation and exit from a VM
- VM hopping: After leaving a VM other are attacked
- Virtual machine-based rootkits

Isolation Vulnerabilities

Lack of Resource Isolation

- Side channel attacks
- Shared storage
- Insecure APIs
- Lack of tools to enforce resource utilization

Lack of Reputation Isolation

- Activities from one customer impact the reputation of another customer and of the cloud provider

Communication Encryption

- Reading data in transit via MITM attacks
- Poor authentication
- Acceptance of self-signed certificates

Vulnerabilities

Weak or No Encryption Data in transit

- Data held in archives and databases
- Un-mounted virtual machine images
- Forensic images and data, sensitive logs and other data at rest put customer data at risk

Unable to Process Data in Encrypted Form

Poor Encryption Key Management

- Hardware security modules (HSM) required in multiple locations
- Key management interfaces which are accessible via the public Internet
- The rapid scaling of certificate authorities issuing key pairs to new virtual machines
- Revocation of keys for decommissioned virtual machines

Vulnerabilities

Low Entropy for Random Number Generation

- The combination of standard system images, virtualization technologies and a lack of input devices means that virtual systems have much less entropy than physical RNGs

No Control of Vulnerability Assessment Process

- Restrictions on port scanning and vulnerability testing are an important vulnerability which, combined with a Acceptable Using Policies which places responsibility on the customer for securing elements of the infrastructure, is a serious security problem

Internal (Cloud) Network Probing

- Cloud customers can perform port scans and other tests on other customers within the internal network

Vulnerabilities

Co-residence Checks

- Side-channel attacks exploiting a lack of resource isolation allow attackers to determine which resources are shared by which customers

Lack of Forensic Readiness

- While the cloud has the potential to improve forensic readiness, many providers do not provide appropriate services and terms of use to enable this.

Vulnerabilities

Media Sanitization

- Shared tenancy of physical storage resources means that sensitive data may leak because data destruction policies may be impossible to implement
- Media cannot be physically destroyed because a disk is still being used by another tenant
- Customer storage cannot be located or tracked as it moves through the cloud

Service Level Agreement

- Clauses with conflicting promises to different stakeholders
- Clauses may also be in conflict with promises made by other clauses or clauses from other providers.

Vulnerabilities

Audit or Certification Not Available to Customers

- The CP cannot provide any assurance to the customer via audit certification.
- Open source hypervisors or customized versions of them (e.g., Xen) may not have Common Criteria certification, etc

Certification Schemes Not Adapted to Cloud

- Very few if any cloud-specific control, which means that security vulnerabilities are likely to be missed.

Vulnerabilities

Storage of Data in Multiple Jurisdictions

- Mirroring data for delivery by edge networks and redundant storage without real-time information available to the customer of where data is stored

Lack of Information on Jurisdictions

- Data may be stored and/or processed in high risk jurisdictions where it is vulnerable to confiscation by forced entry.

Vulnerabilities

Lack of Cloud Security Awareness

- Cloud customers and providers are not aware of the risks they could face when migrating into the cloud, particularly those risks that are generated from cloud specific threats, i.e. loss of control on data, cloud provider lock-in, exhausted resources of the cloud provider.

Lack of Vetting Processes (Personel Background Checks)

- Since there may be very high privilege roles within cloud providers, due to the scale involved, the lack or inadequate vetting of the risk profile of staff with such roles is an important vulnerability

Unclear Roles and Responsibilities

- Inadequate definition of roles and responsibilities in the cloud provider organization

Vulnerabilities

Poor Enforcement of Role Definitions

- Within the cloud provider, a failure to segregate roles may lead to excessively privileged roles which can make extremely large systems vulnerable

Need-to-know Principle Not Applied

- Poorly defined roles and responsibilities
- Parties should not be given unnecessary access to data

Inadequate Security Procedures

- Lack of physical perimeter controls (smart card authentication at entry);
- Lack of electromagnetic shielding for critical assets vulnerable to eavesdropping.
- Lack of policy or poor procedures for logs collection and retention
- Inadequate or misconfigured filtering resources

Os And Application Vulnerabilities

Mismanagement

- System or OS vulnerabilities
- Untrusted software
- Lack of - or a poor and untested - business continuity and disaster recovery plan
- Lack of - or incomplete or inaccurate - asset inventory
- Lack of - or poor or inadequate - asset classification
- Unclear asset ownership

Application Vulnerabilities and Poor Patch Management

- Bugs in the application code
- Conflicting patching procedures between provider and customer
- Application of untested patches
- Vulnerabilities in browsers
- Dormant virtual machines
- Outdated virtual machine templates

Cloud security ;-)

Information-centric security with one of

• Untrusted infrastructure	IaaS
• Untrusted Platform	PaaS
• Untrusted Software	SaaS

and a larger attack surface

Cloud Computing Threat Model

ENISA

Cloud Computing Risk Assessment

Threat Model

Risk 1:	Resource Exhaustion
Risk 2:	Customer Isolation Failure
Risk 3:	Management Interface Compromise
Risk 4:	Interception of Data in Transmission
Risk 5:	Data leakage on Upload/Download, Intra-cloud
Risk 6:	Insecure or Ineffective Deletion of Data
Risk 7:	Distributed Denial of Service (DDoS)
Risk 8:	Economic Denial of Service
Risk 9:	Loss or Compromise of Encryption Keys
Risk 10:	Malicious Probes or Scans
Risk 11:	Compromise of Service Engine/Hypervisor
Risk 12:	Conflicts between customer hardening procedures and cloud environment

Threat Model

Risk 13:	Subpoena and E-Discovery
Risk 14:	Risk from Changes of Jurisdiction
Risk 15:	Licensing Risks
Risk 16:	Network Failure
Risk 17:	Networking Management
Risk 18:	Modification of Network Traffic
Risk 19:	Privilege Escalation
Risk 20:	Social Engineering Attacks
Risk 21:	Loss or Compromise of Operation Logs
Risk 22:	Loss or compromise of Security Logs
Risk 23:	Backups Lost or Stolen
Risk 23:	Unauthorized Access to Premises, Including Physical Access to Machines and Other Facilities
Risk 25:	Theft of Computer Equipment.

Threat Model Second Version

Policy Risk

R.1 Lock-in

R.2 Loss of governance

R.3 Compliance challenges

R.4 Loss of business reputation due to co-tenant activities

R.5 Cloud service termination or failure

R.6 Cloud provider acquisition

R.7 Supply chain failure

Threat Model Second Version

Technical risks

R.8 Resource exhaustion (under or over provisioning)

R.9 Isolation failure

R.10 Cloud provider malicious insider - abuse of high privilege roles

R.11 Management interface compromise (manipulation, availability of infrastructure)

R.12 Intercepting data in transit

R.13 Data leakage on up/download, intra-cloud

R.14 Insecure or ineffective deletion of data

R.15 Distributed denial of service (DDoS)

R.16 Economic denial of service (EDOS)

R.17 Loss of encryption keys

R.18 Undertaking malicious probes or scans

R.19 Compromise service engine

R.20 Conflicts between customer hardening procedures and cloud environment

Threat Model Second Version

Legal risks

- R.21 Subpoena and e-discovery
- R.22 Risk from changes of jurisdiction
- R.23 Data protection risks
- R.24 Licensing risks

Threat Model Second Version

Risks not specific to the cloud

R.25 Network breaks

R.26 Network management (ie, congestion / non-optimal use)

R.27 Modifying network traffic

R.28 Privilege escalation

R.29 Social engineering attacks (ie, impersonation)

R.30 Loss or compromise of operational logs

R.31 Loss or compromise of security logs (manipulation of forensic investigation)

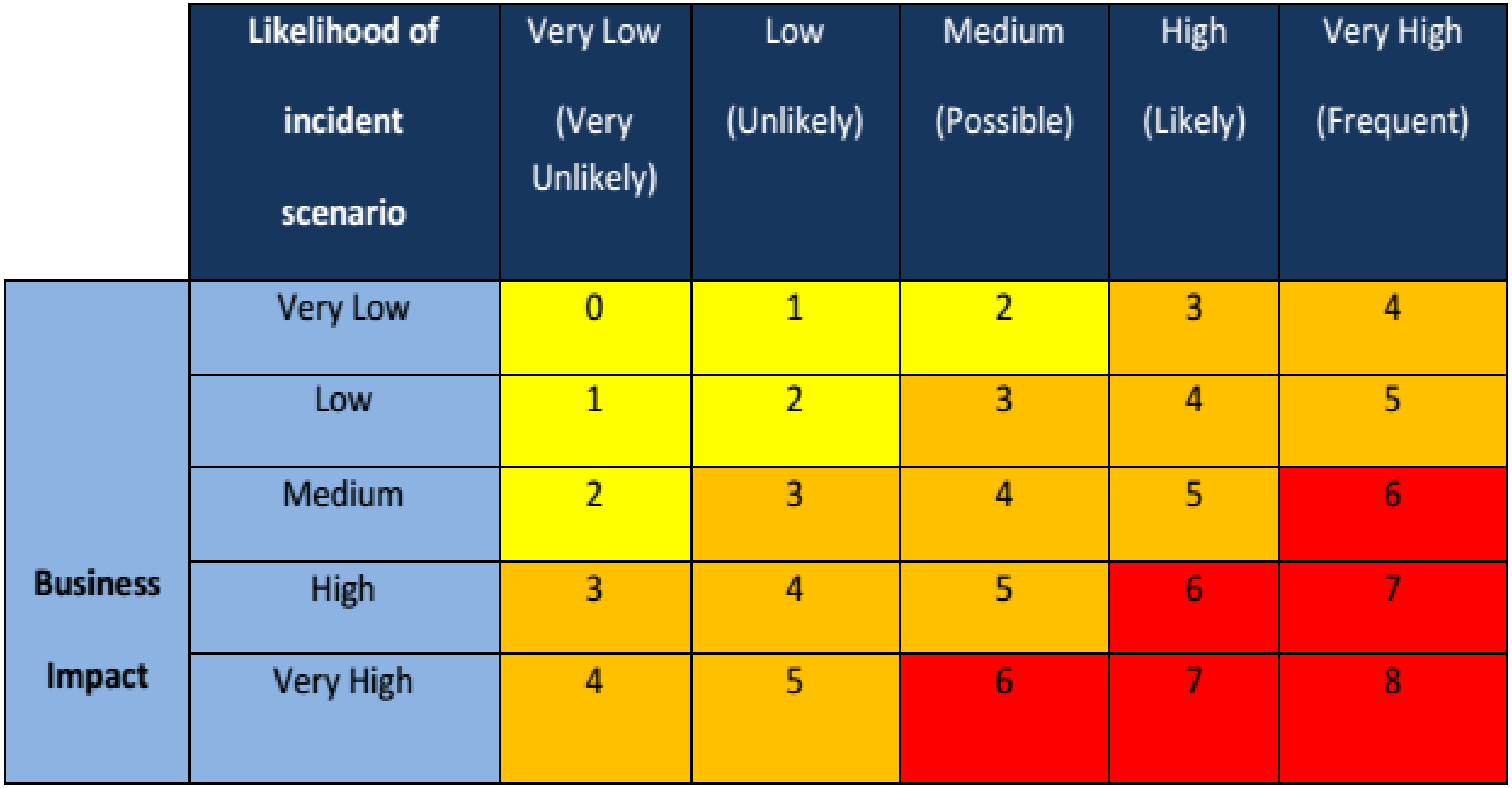
R.32 Backups lost, stolen

R.33 Unauthorized access to premises (including physical access to machines and other facilities)

R.34 Theft of computer equipment

R.35 Natural disasters

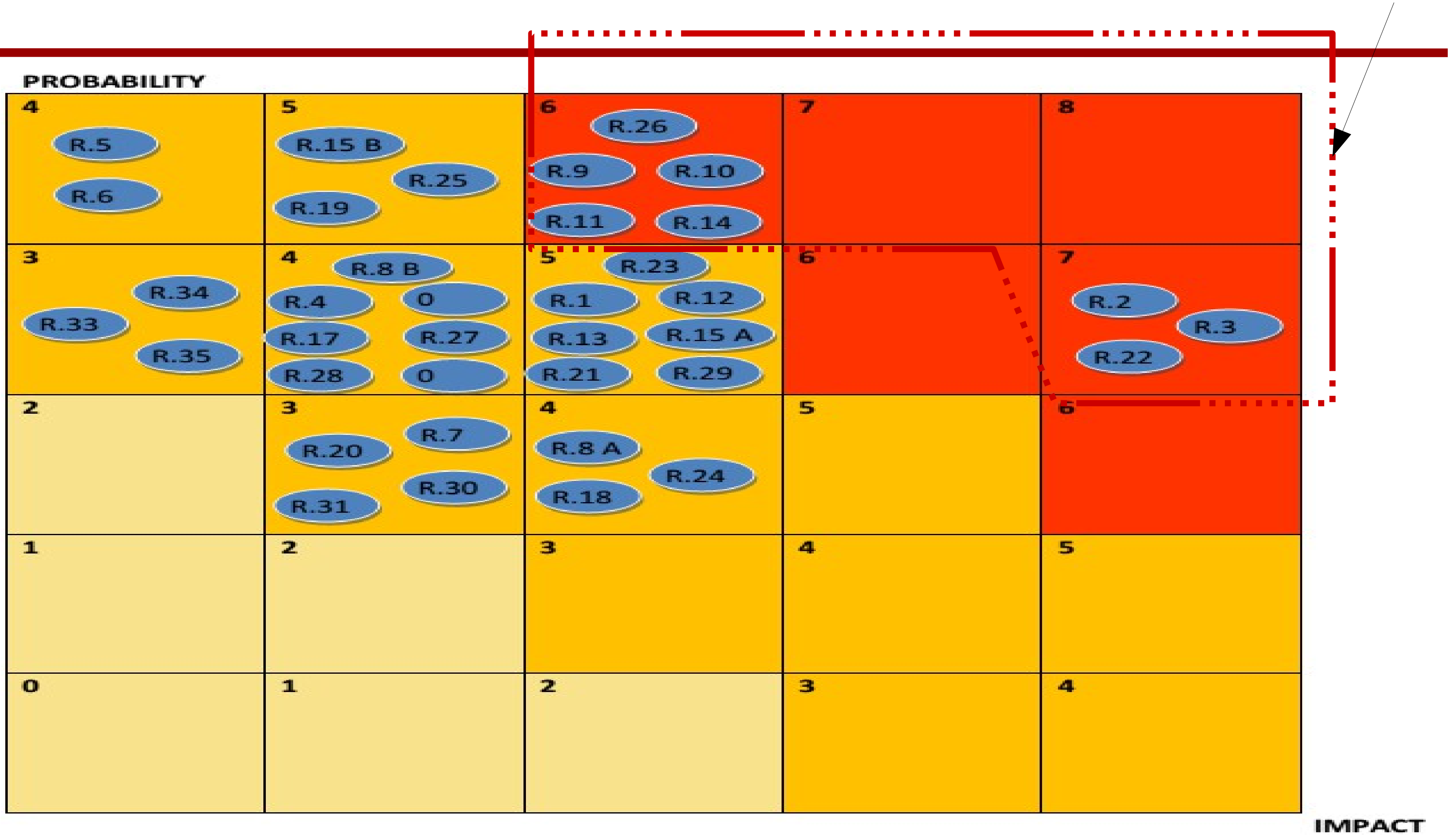
Risk Assessment



	Likelihood of incident scenario	Very Low (Very Unlikely)	Low (Unlikely)	Medium (Possible)	High (Likely)	Very High (Frequent)
Business Impact	Very Low	0	1	2	3	4
	Low	1	2	3	4	5
	Medium	2	3	4	5	6
	High	3	4	5	6	7
	Very High	4	5	6	7	8

Risk Assessment

most dangerous are related to the provider



PROBABILITY

4 R.5 R.6	5 R.15 B R.25 R.19	6 R.26 R.9 R.10 R.11 R.14	7	8
3 R.34 R.33 R.35	4 R.8 B R.4 0 R.17 R.27 R.28 0	5 R.23 R.1 R.12 R.13 R.15 A R.21 R.29	6	7 R.2 R.3 R.22
2	3 R.7 R.20 R.30 R.31	4 R.8 A R.24 R.18	5	6
1	2	3	4	5
0	1	2	3	4

IMPACT

Risk Assessment: top risks

R.2	Loss of governance
R.3	Compliance challengessolation failure
R.10	Cloud provider malicious insider - abuse of high privilege roles
R.11	Management interface compromise (manipulation, availability of infrastructure)
R.14	Insecure or ineffective deletion of data
R.22	Risk from changes of jurisdiction
R.26	Network management (ie, congestion / non-optimal use)

In the following...

- We will focus on technical risks and in some risks related to the provider
- A typical risk is the failure of isolation or the abuse of roles by the insider (working for the cloud provider)