Introduction to Cloud Computing (CS 524)

(Quiz 1)

Prof. Igor Faynberg

Student Name: **Paras Garg**

Course Section: **CS 524-A**

Quiz 1.1 –
Give one example of each a) SaaS, b) PaaS, and c) IaaS. (Try to think of your own examples rather than the ones that are given in the book.)

Solution 1.1 –

a) SaaS: *SaaS* stands for Software-as-a-Service. SaaS provides the capabilities to the customer to use provider's application running on a Cloud infrastructure. The application running are accessible from various client devices through either a thin client interface.

   Example: Microsoft Office

b) PaaS: *PaaS* stands for Platform-as-a-Service. PaaS provides capabilities to the customer to deploy onto the cloud infrastructure or acquired applications created using programming languages, libraries, services, and tools supported by the provider but does not manage or control the Cloud infrastructure.

   Example: Google App Engine

c) IaaS: *IaaS* stands for Infrastructure-as-a-Service. IaaS provides the capabilities to customer to provision processing, storage, network, and other fundamental computing resources where the customer is able to deploy and run arbitrary software, which even include operating systems and applications but does not manage or control Cloud infrastructure.

   Example: Microsoft Azure

Quiz 1.2 –
Explain the difference between the Trusted Platform Module (TPM) and the Hardware Security Module (HSM). Give one example of the use of the TPM and one example of the use of the HSM.

Solution 1.2 –
Trusted Platform Module (TPM) provides facilities for secure generation of cryptographic keys. Example: TPM offers full disk encryption of applications, such as SecureDoc, in modern Linux Kernels and BitLocker Drive. This technology can be use to protect keys used to encrypt the computer's hard disks and provide integrity authentication for a trusted boot pathway.

HSM means Hardware Security Module. Example: The AWS CloudHSM service helps you meet corporate, contractual and regulatory compliance requirements for data security by using dedicated Hardware Security Module (HSM) appliances within the AWS cloud. With CloudHSM, you control the encryption keys and cryptographic operations performed by the HSM.

Quiz 1.3 –
Explain what it means that an instruction is not virtualizable and give an example of such an instruction's behavior. Can you provide an example of an x86 instruction that is not virtualizable?

Solution 1.3 –
For an instruction is virtaulizable, Popek and Goldberg requirement must be met, but in the case when the instruction is not virtualizable these requirements are overruled. The operating system runs in modes i.e. user mode or system mode. When any privileged or sensitive instruction which is supposed to run under only system mode and the instruction traps and when executed in user or guest mode. This any attempt of utilizing or modifying any system application or resource by the user is termed as not virtualizable instructions.

Non-virtualizable instructions falls into three classes. In first class, A major problem is that the processor has only one register for each of these, which means that they need to be replicated for each virtual machine. In second class, the instructions that copy parts of the STATUS register into either general registers or memory and in third class, instructions that reference of the storage protection system, memory, or address relocation systems.

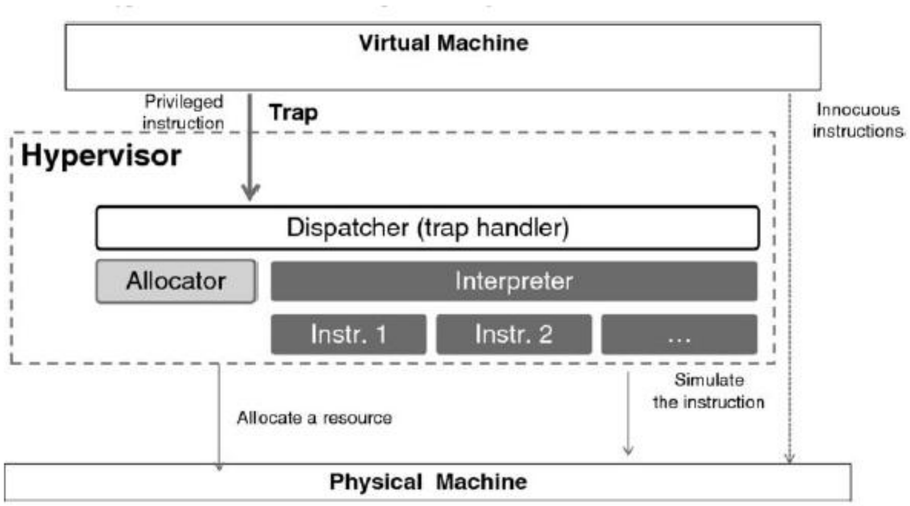
The problem with x86 instructions is that, the instructions were allowed while being executed in the user mode to transfer the value of the STATUS register to a general register. This instruction is behavior-sensitive in the Popek and Goldberg taxonomy because it allows a user program to discover which mode is running in it.

Quiz 1.4 –

You are given a machine M and a type-1 hypervisor X. Let us call this configuration (M, X). You can run a virtual machine V in this configuration on top of the hypervisor: (M, X, V). Is it possible to run X on top of X so as to obtain (M, X, X) and then (M, X, X, V)? If not, explain why not. If it is possible, explain how that would work—by drawing a scheme of interrupt processing at each level. For simplicity, assume that M has a fully-virtualizable processor.

Solution 1.4 –

No, it is not possible to run because some interrupts (such as alarm interrupts that deal with time sharing among multiple machines) are handled by the hypervisor itself, with the result that a virtual machine that has exceeded its time quantum is stopped, and the CPU is given to another virtual machine.



Quiz 1.5 –

Explain the problem that the I/O MMU solves. What problems are introduced when using the I/O MMU?

Solution 1.5 –

Input Output Memory Management Unit (I/O MMU) maps the device visible virtual address to a physical address by connecting a Direct Memory Access (DMA) to the main memory. The problems introduce when using I/O MMU are:

i. Direct Memory Access additional safety performance issues as it enables an I/O device to read and write host RAM directly without involving the CPU.
ii. Security checks or encrypting disks, writes can incur by modifying I/O requests.

Quiz 1.6 –

Explain how XEN support I/O processing in a guest operating system.

Solution 1.6 –

XEN support both paravirtualized and fully virtualized guests, respectively called PV and HVM. For guests running in HVM, XEN emulates low-level hardware and firmware components—such as graphic, network, and BIOS adapters, using techniques described in the previous section. Predictably, emulation often results in degraded performance. XEN deals with this by creating yet another mode, called PV-on-HVM (or PVHVM), in which an HVM guest is paravirtualized only partly.Xen's approach to handling physical I/O devices is straightforward and elegant. XEN creates a special environment—called a *domain*—for each guest.

Quiz 1.7 –
What is the difference between the virtual machine and a container? Provide an example of a situation where you would use a Linux container rather than a virtual machine.

Solution 1.7 –
Virtual Machine versus Container

i. Virtual Machines are used to run only a single application in an operating system, but Containers are used to run many applications.
ii. The resources (like memory, disk space and CPU utilization) available in Virtual Machine are much lesser in size as compared to Containers.
iii. Containers have more economical ways than Virtual Machines to move to the cloud.
iv. Containers are more efficient than Virtual Machines as it provides more and necessary resources for any computation or processing.
v. Virtual Machines can run more user or guest operating system and operating system have full control of the machine, whereas Contains have control of operating system user space.

Situations to use Container rather than a Virtual Machine

i. While rewriting an application based on microservices, user must use Container.
ii. Containers should not be used while writing any application from the scratch.