Operating Systems

Lecture 11b

Dr Ronald Grau School of Engineering and Informatics Spring term 2018

Previously

Security

- Terminology
- Cryptography
- Authentication
- Access Control
- Vulnerabilities
- Design

Quick recap of some security topics

Questions:

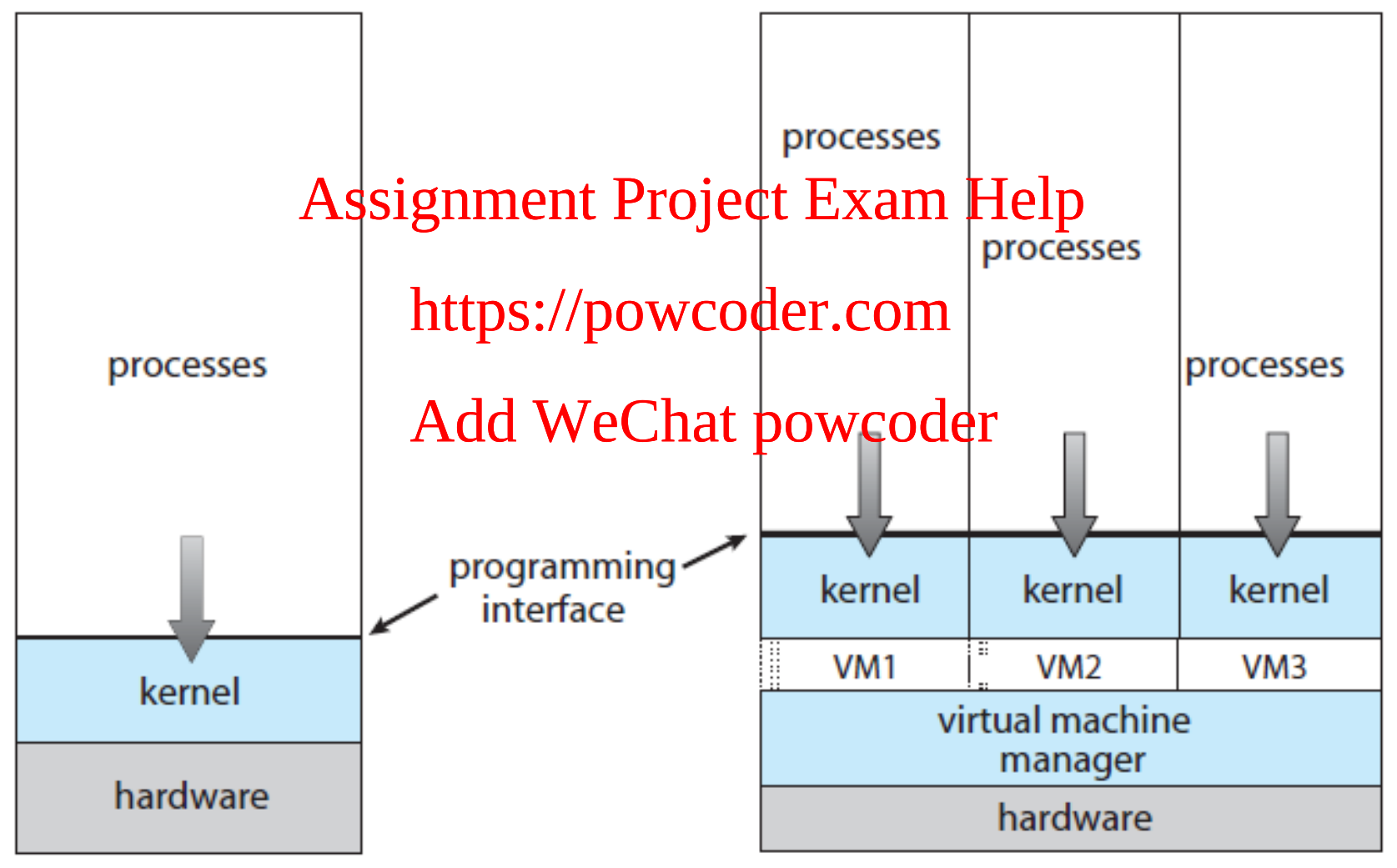
1. What are possible violations of confidentiality?
2. Ransomware poses a threat to which security property?
3. Which security mechanism is targeted by Phishing?
4. What is the principle of least privilege about?
5. What kinds of applications lend themselves to the use of symmetric cryptography?

Today

Virtualisation

- Virtual machine concepts
- Hypervisors
- Containers
- Virtualisation techniques

Virtual Machines



Virtualisation

What is the motivation behind virtualisation?

- Protection
- Flexibility
- Optimisation of hardware usage
- Easier maintenance

→ Enabler of cloud computing

Virtualisation

Requirements of virtualisation

- Provision of an environment identical to the original machine
- Only minor performance impact on applications
- Virtual machine manager is in complete control of the system

Some Terminology

Disambiguation #1

- Virtual machines for programming environments
  - Program compiled to intermediate language
  - Intermediate language executed by virtual machine
  - E.g. JVM, .NET, LLVM
- Sandboxing, Container
  - Provides protection of applications against each other
  - E.g. BSD Jails, LXC, Docker, Solaris Zones
  - Desktop/application virtualisation: e.g. Citrix, Jukebox

Some Terminology

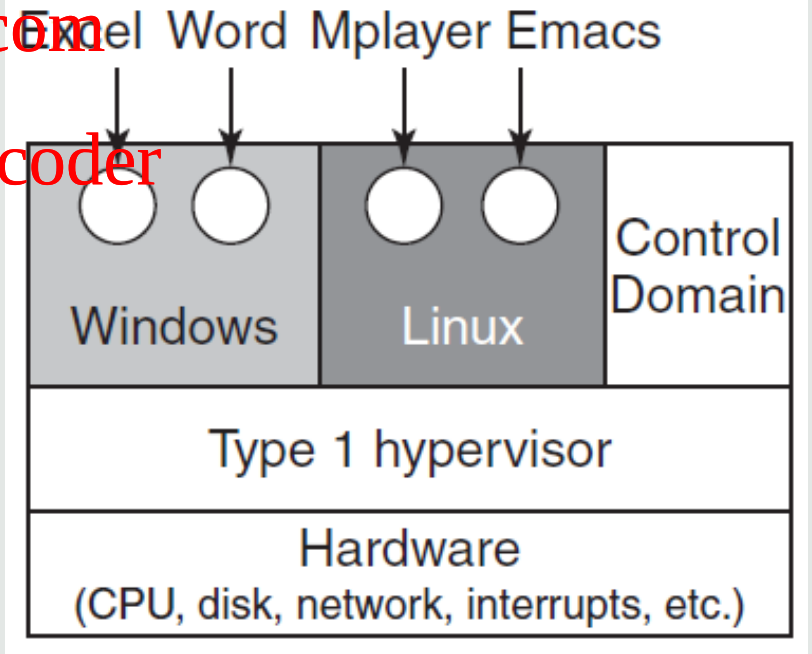
Disambiguation #2

- Emulation
  - Full simulation of HW, e.g. instruction set simulator, QEMU
  - Allows running code compiled for different CPU architectures
- Hypervisor, Virtual machine manager
  - Partial simulation sufficient to run a guest OS
  - Guest OS runs as native code

Hypervisors

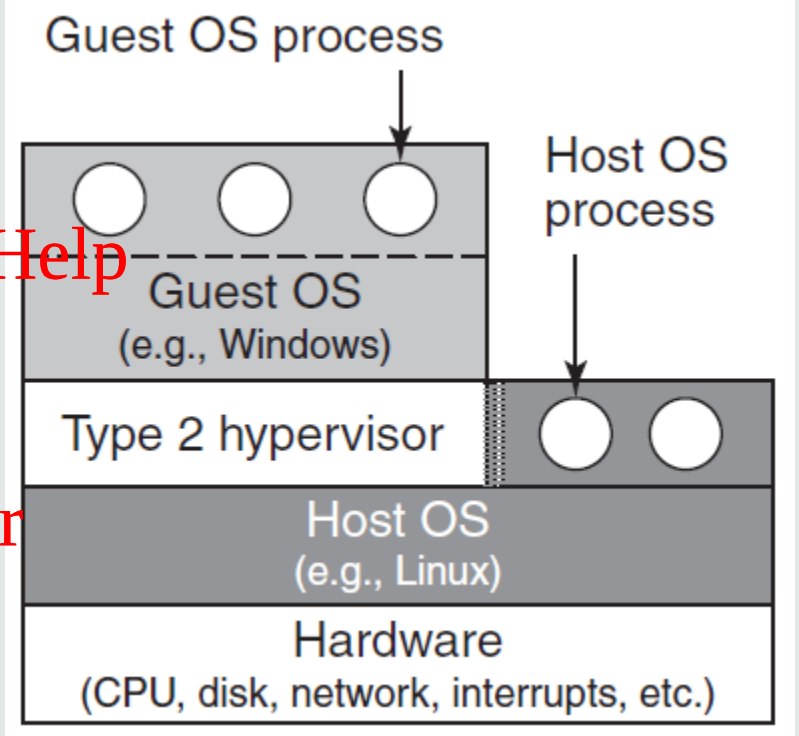
- Type 0 hypervisor
  - Hardware-based solutions, no need for a dedicated "host"-OS
    e.g. IBM LPAR (logical partitions)

- Type 1 hypervisor
  - Operating system that manages guest operating systems
  - E.g. Xen, Microsoft Hyper-V, VMWare ESX



Hypervisors

- Type 2 hypervisor
  - Application that manages guest operating systems
  - E.g. VMWare Workstation, Oracle VirtualBox
- Para-virtualisation
  - Guest operating system is aware of running on a VM
  - Performance optimisations via hypercalls: E.g. run device drivers of host instead of running guest device drivers in virtualised environment



Hypervisors

Hypervisor

- Manages resources allocated to guest OSes
- Schedules guest OSes, keeps CPU state ("Virtual CPU")

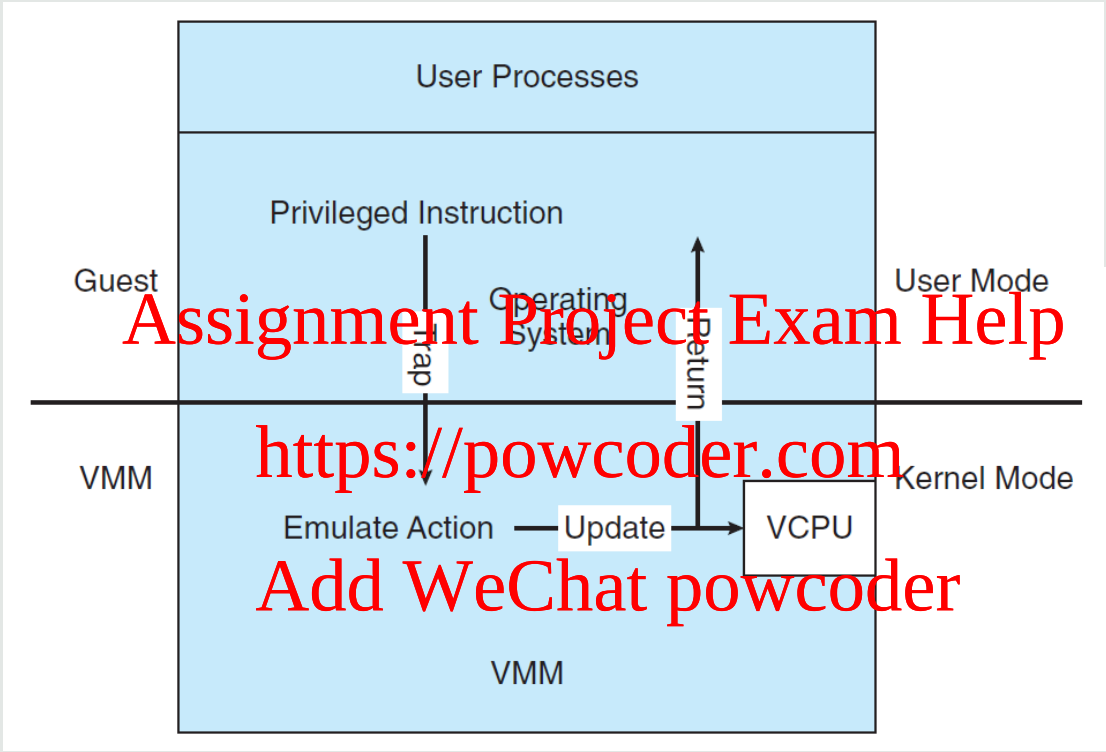
Requires more than two protection modes

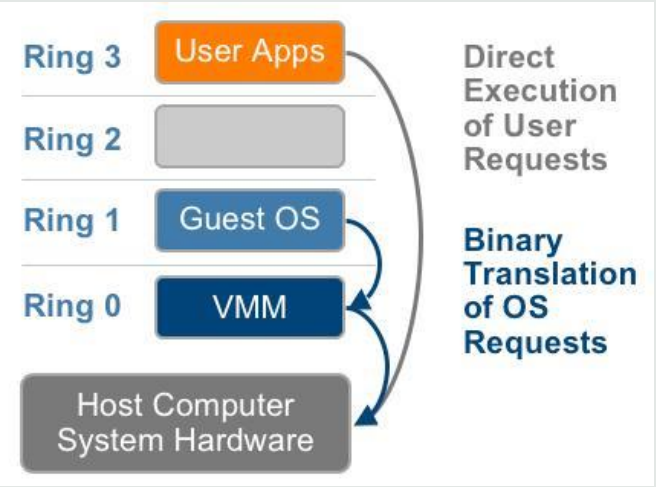
- Guest OS user mode
- Guest OS kernel mode
- Hypervisor

Guest OS kernel uses privileged instructions:

- How to ensure protection?

Trap and Emulate





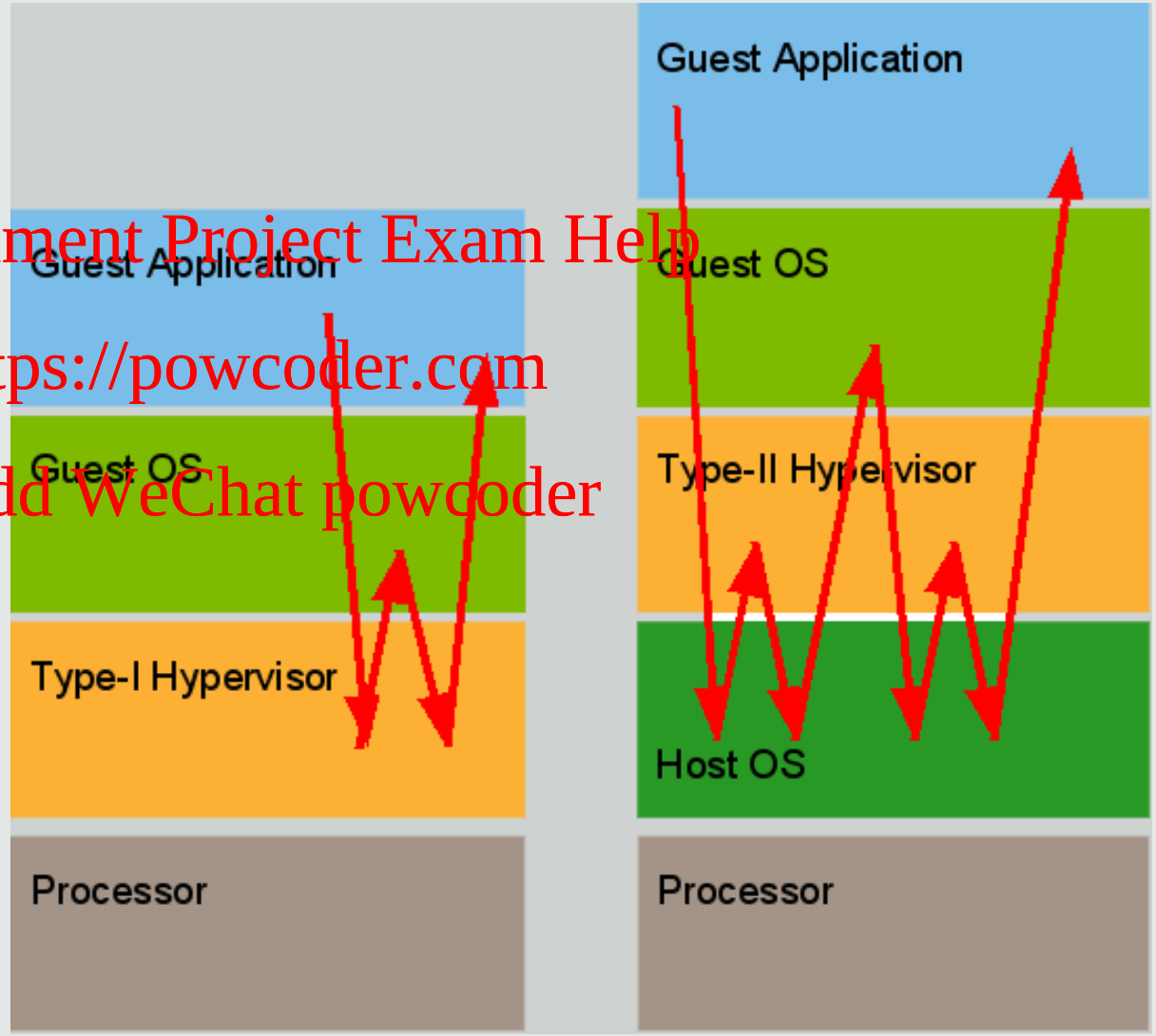
Non-privileged instructions execute on physical CPU

Guest OS kernel uses privileged instructions:

- VCPU keeps track of guest OS mode
- Privileged instructions are emulated (binary translation)

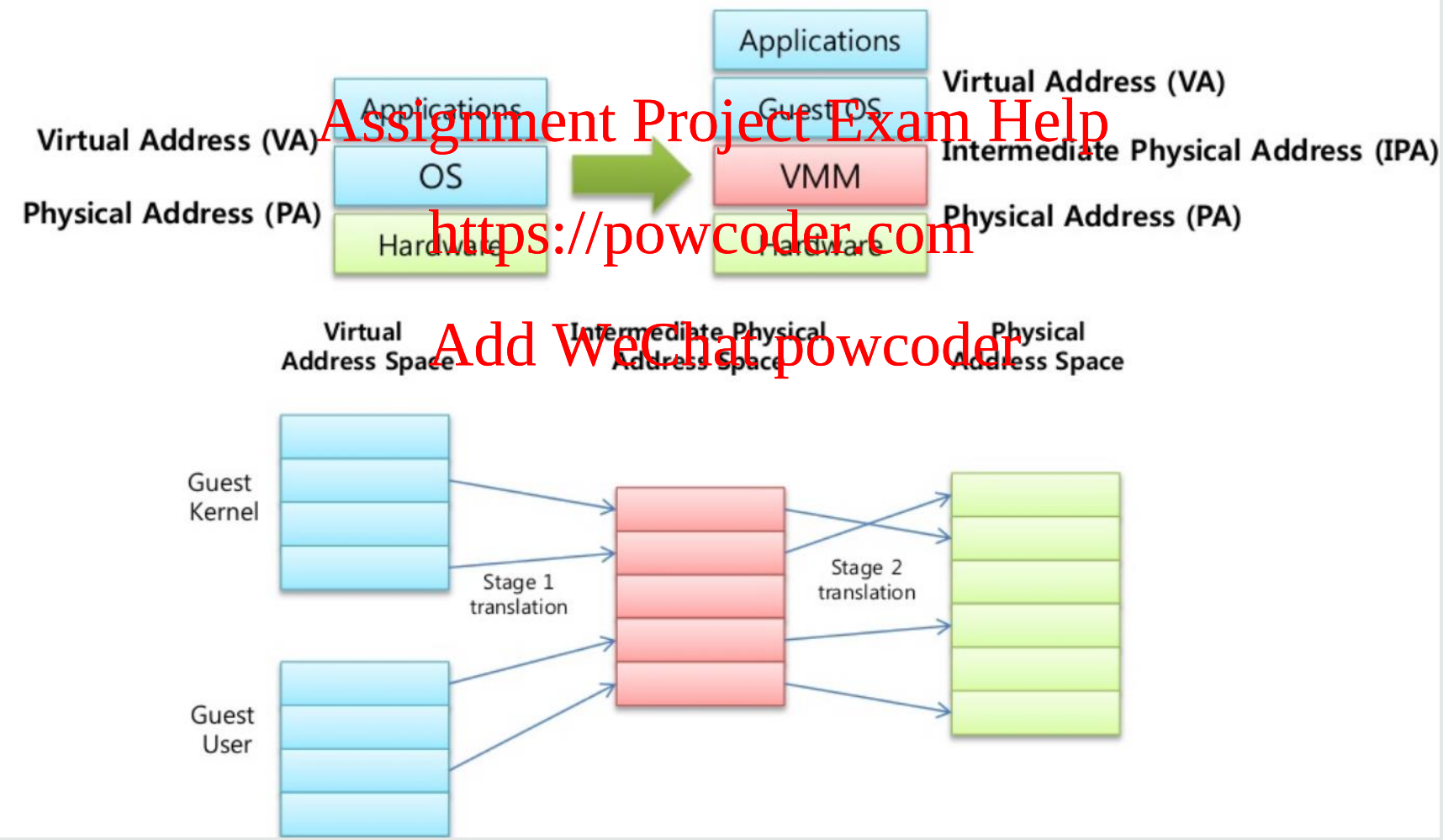
Trap and Emulate Type 1 vs 2

Type 2 Hypervisors require a kernel module



Virtualising Virtual Memory

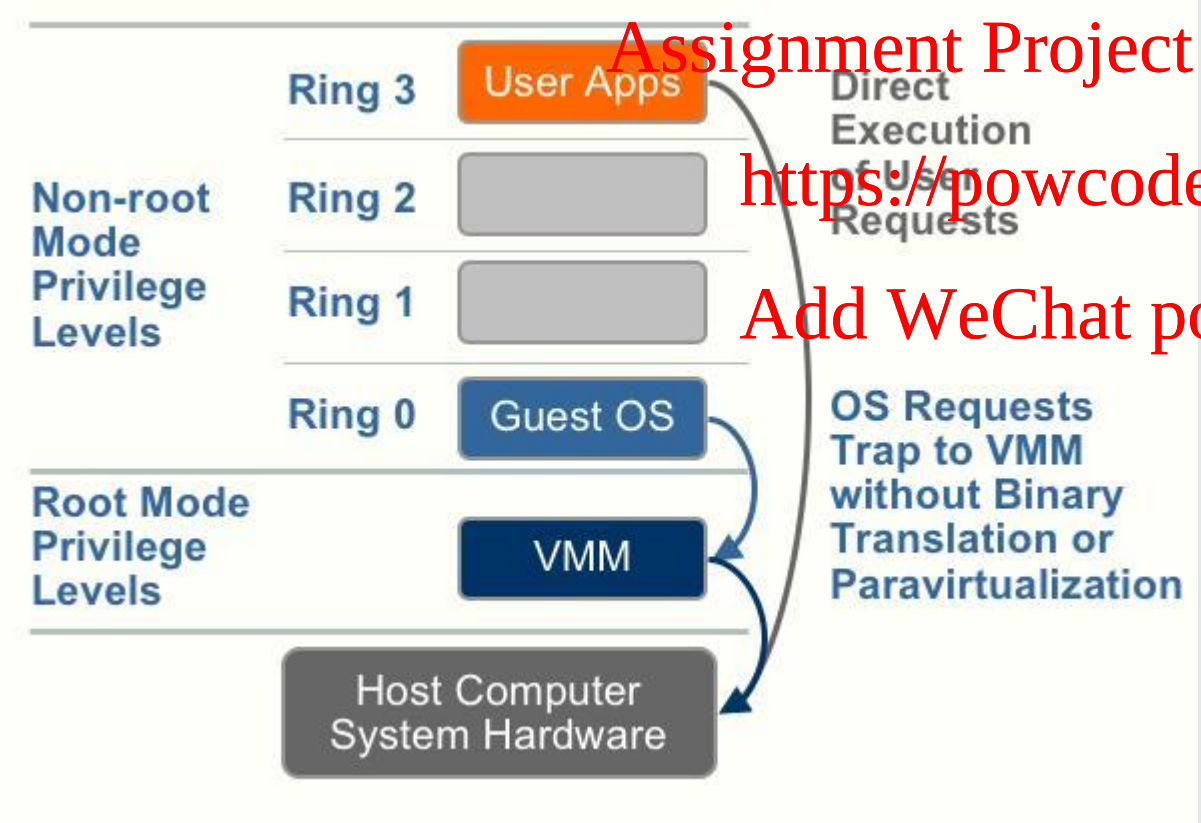
Nested Page Tables



Hardware Support

Examples: AMD SVM, Intel VT

Additional modes for host and guest system:



VCPU and nested page tables implemented in hardware

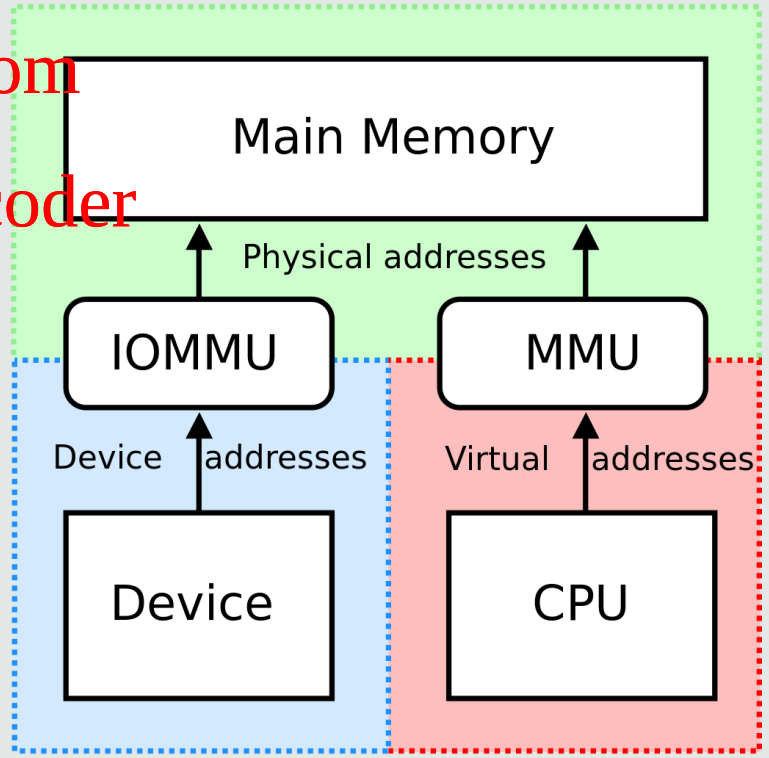
I/O Virtualisation

Problems:

- Guest OS have ISR for the same interrupt
- Guest OS programs DMA with same physical memory addresses

I/O MMU

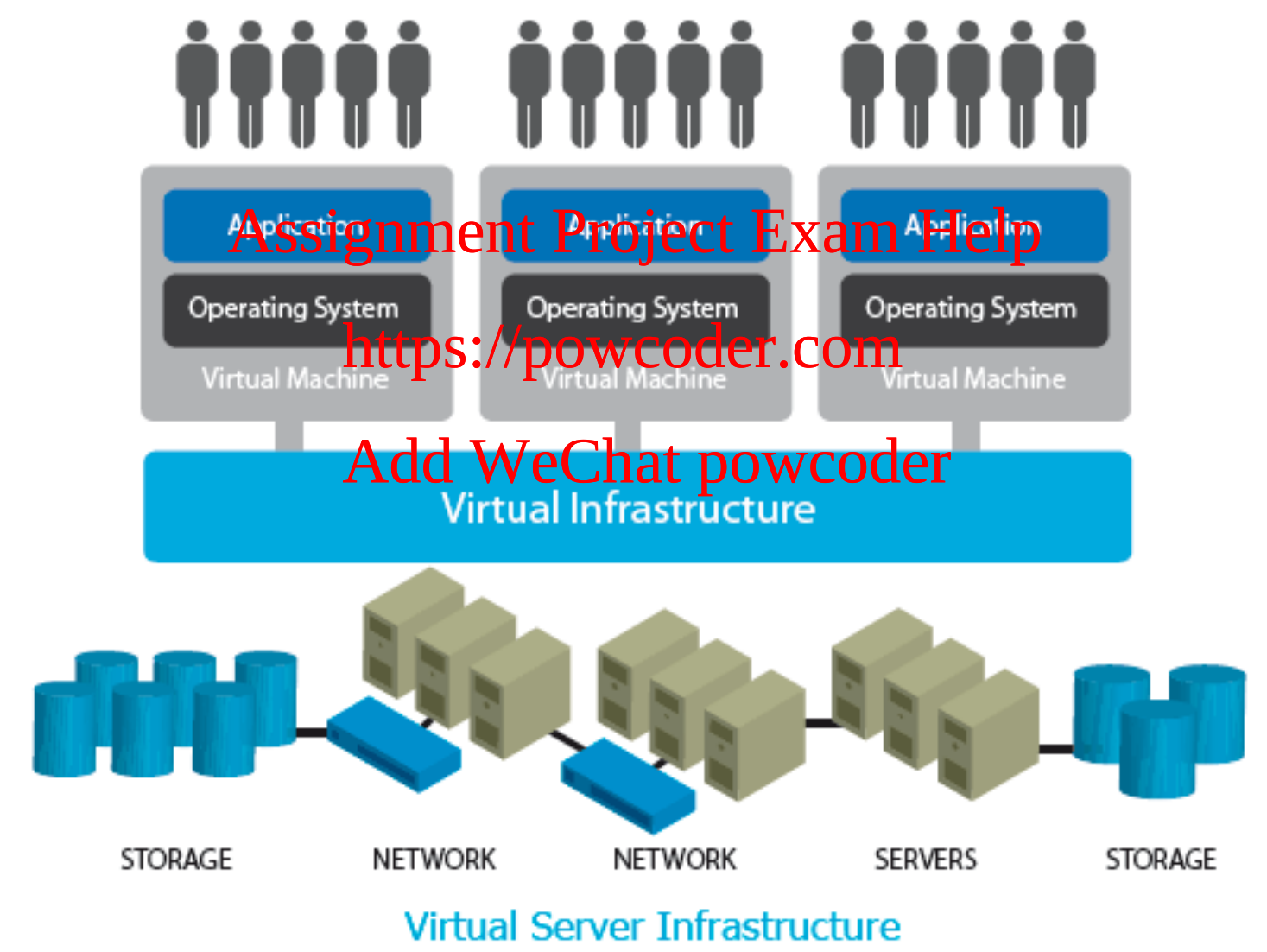
- Interrupt remapping
- Protection domains, address translation



Para-virtualisation

- Modify guest OS to "forward" I/O system calls as hypercalls to hypervisor

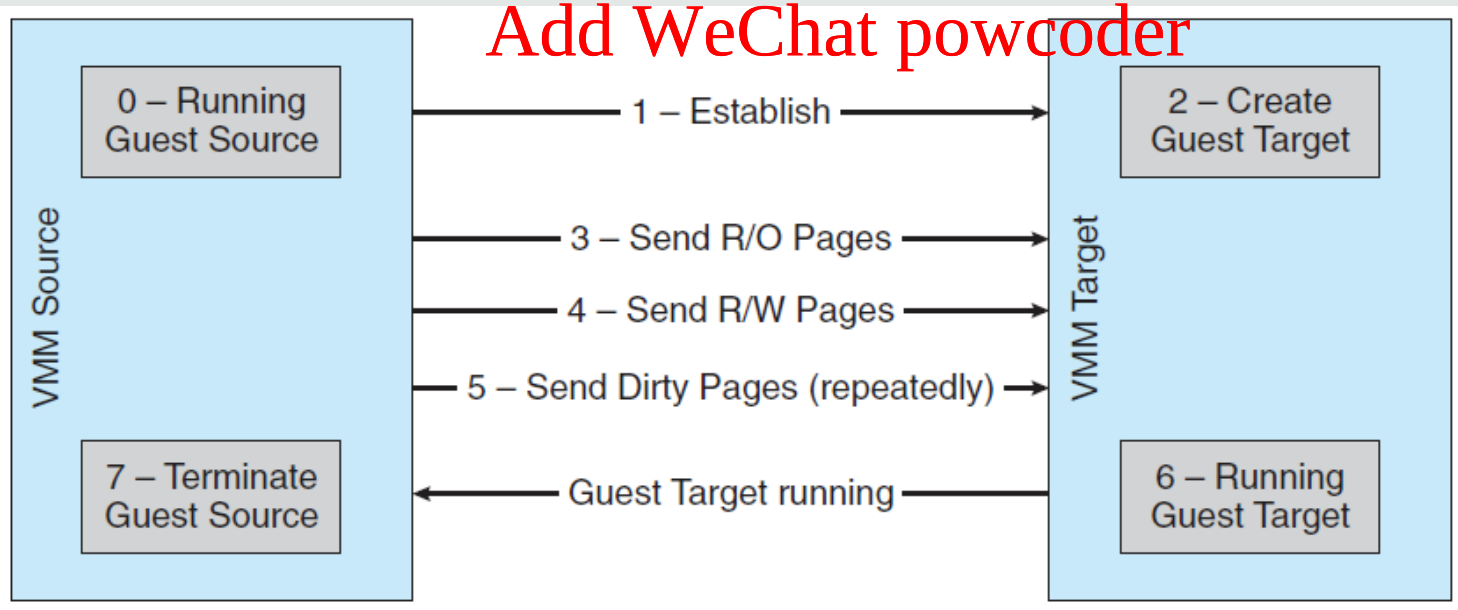
Virtual Infrastructure



Virtual Server Infrastructure

Advantages of Virtualisation

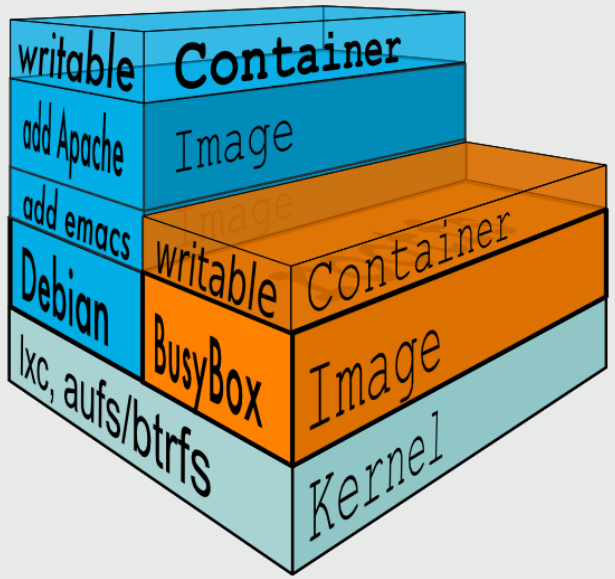
- Guest OSes are isolated (security)
- We can take snapshots of the current state of a guest OS
- Suspend execution and resume
- Cloning (reliability), templating (maintenance)
- Live migration (load balancing)



Containerisation

Package an application into a container

- Easy and rapid deployment
- Templating
- Sandboxing
- More lightweight
  - Container uses kernel of host OS
    → cannot run Windows container on Linux host!
  - Faster start-up
- Allows stacking/nesting of containers



Container Technology

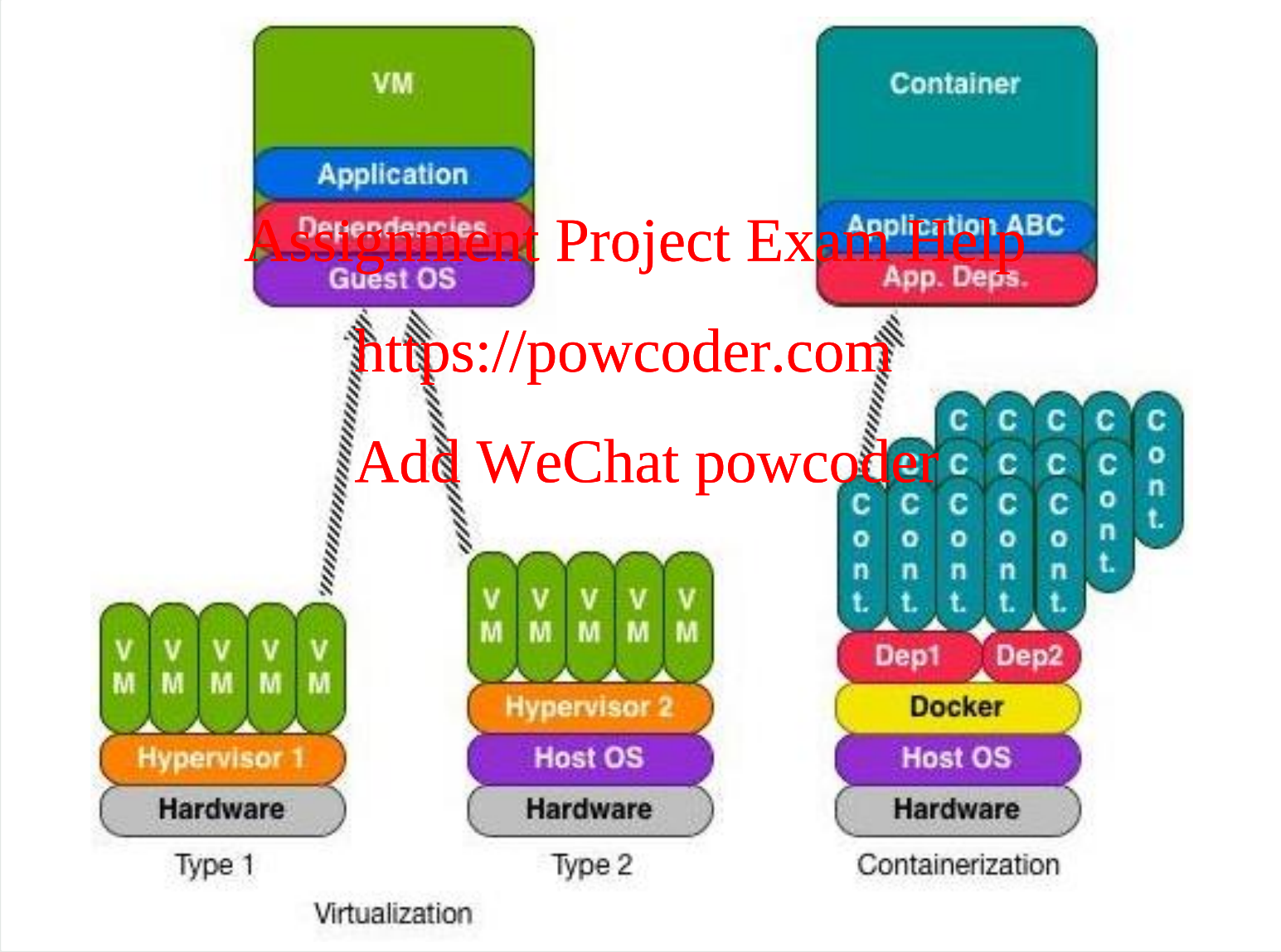
`chroot (1982)`

- Set root directory of file system for a process
  → cannot access files outside
- Idea extended to BSD Jails (2000), Solaris containers (2005), LXC (2008), Docker (2013)

Linux namespaces

- Virtualisation of system resources, e.g. users, processes, file systems, sockets
- Namespaces can be nested
  → hierarchies, isolation
- E.g. a process thinks that it is running as root although it has permissions of a less privileged user

Virtualisation vs Containerisation



Principle of Virtualisation

"All problems in computer science can be solved by another level of indirection."

(David Wheeler)

"... except for the problem of too many layers of indirection."

(Kevlin Henney)

Summary

Variety of VM concepts

- Emulators
- Hypervisors Type 0, 1, 2
- Containers
- Programming language VMs

Virtualisation techniques for

- CPU, memory, I/O

Fundamental principle of virtualisation:

- Indirection

Read

- Tanenbaum & Bos., Modern Operating Systems
  - Chapter 7
- Silberschatz et al., Operating System Concepts
  - Chapter 16

Next Lecture

- There are no labs on Monday because of the May bank holiday (Please go to the other sessions on Wednesday or Friday)
- There is no lecture on Wednesday

- Revision lecture on Friday, usual time and place.

- If you have any questions, e-mail me or post them the forum and we may take them up in the lecture, as far as time permits.