

## Lecture 12: 03/03/2020

Lecturer: Prashant Shenoy

Scribe: Gursimran Kaur

### 12.1 VIRTUALIZATION

- Emulation – Copy the whole systems from software to hardware including processor, instructions, etc. It does the binary translation of the assembly code on the actual machine, hence it is slow.
- Native Virtualization – The guest and the host OS are of the same type.
- Para Virtualization – Runs a special version of the OS, i.e., the OS has a modified API. The OS code is modified once, compiled and then run. Faster than type 2 hypervisors (see below).
- Application Level Virtualization – This reimplements a library itself, so that there's no need to virtualize a whole OS.

### 12.2 HYPERVISOR

- It is another layer of virtualization.
- **TYPE 1 HYPERVISOR:**  
Sits on top of the hardware, hence it's also called bare metal hypervisor. This acts as a kernel, so instead of an OS booting up, the hypervisor boots up. In this, the hypervisor runs in Kernel mode and the OS runs in user mode. However, the OS doesn't know that it's privileges have been limited.
  - How does the OS carry out privileged instructions in user mode?  
Whenever the OS wants to carry out a privileged instruction, it causes a trap by the hardware, this is then handled by the hypervisor and it carries out the privileged instruction.
  - Hardware support for traps –  
Hardware support is required for causing traps which decides if sensitive instructions should cause a trap or not.

- **TYPE 2 HYPERVISOR :**

This sits on top of an already existing OS. The virtualization layer runs as an application on the OS.

- How does OS carry out privileged instructions?

Sensitive instructions in guest OS are replaced with a function call/system call to the actual OS which then carries out these instructions.

- The code is scanned once before executing to replace all these sensitive instructions with system calls. This makes it slow.
- No hardware support needed.

- **NETWORKING :** Each OS has a 'virtual' NIC (Network Interface Controller). The virtual NIC are mapped to the actual NIC of the hardware, hence the actual NIC is shared.

## **12.3 MEMORY VIRTUALIZATION**

- Each OS maintains it's own 'shadow' page table as a result of virtualization.
- The hypervisor makes the changes in the actual page tables by trapping faults created by OS operations on the shadow page table and then executing those instructions in the actual page table.