# 3. VM setup

Virtualisation

One computer running multiple OS.

Life before Virtualisation:

- to run app/services we need servers
- physical computer (Servers in Datacenter)
- One service - One Server (Isolation)
- Servers are always overprovisioned
- Server Resources mostly underutilized
- Huge capital expenditure and operational expenditure

After Virtualisation

- allows one computer to run multiple OS
- Partition physical resource in virtual resource
- isolated from each other with its own isolated environment
- each virtual machine needs its own OS
- Server Virtualisation is the most common virtualisation.
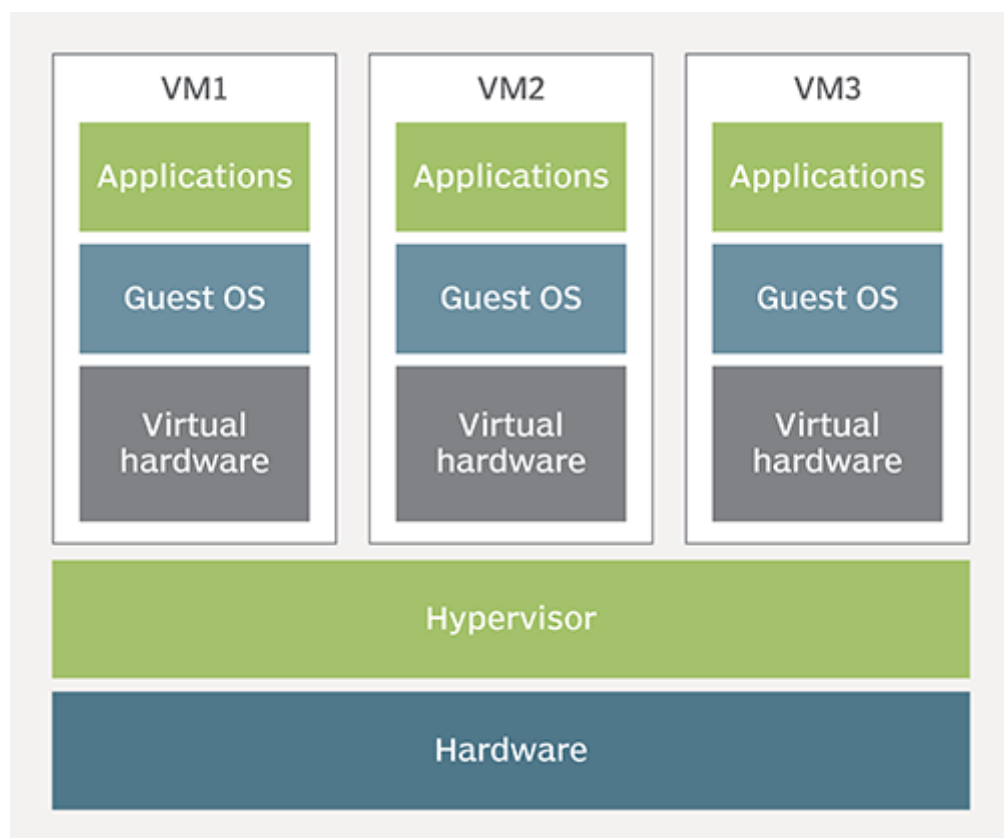
Terminologies

- Host OS

- Guest OS - OS of VM

- VM - virtual machines

- snapshot - taking a backup of a machine

- Hypervisor - tool or software that let us create virtual machines. It enables virtualisaiton
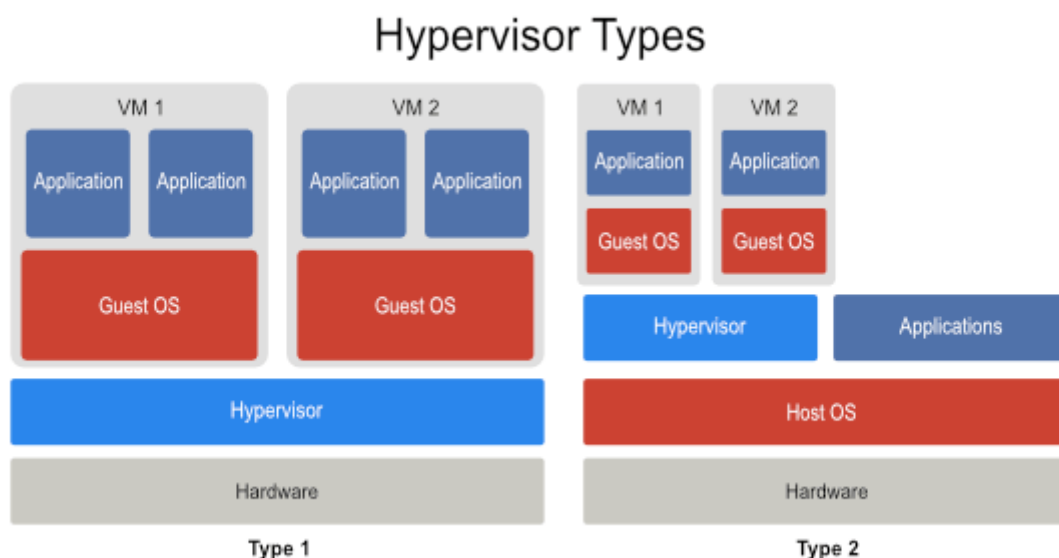
Types of Hypervisor

Type 1:

- Bare Metal

- Runs as a base OS

- Production

- eg: VMware esxi, Xen Hypervisor, HyperV

- can be clustered together to distribute VM on cluster of Hypervisor so if one goes down another can takeover

Type 2:

- Runs a software

- Learn and test purpose only

- eg: Oracle virtualbox, VMware server

Hypervisor Types

Setting VM machines

- CentOS VM (Manually)

- Ubuntu VM (Manually)

- CentOS VM (Automatically with vagrant on virtualbox)

- Ubuntu VM (Automatically with vagrant on virtualbox)

Vagrant

Contents

```
1.   Intro to Vagrant
2.   Create VM Automatically
3.   Vagrant Commands
4.   Vagrant Networking
5.   Provisioning
6.   RAM, CPU & Disk
7.   Multi VM Vagrantfile
8.   Documentation
```

What is vagrant?

Vagrant is an automation tool to manage VM life cycle. In short VM automation tool

Why VM automation tool?

- OS Installations
- Time Consuming
- Manual setup
- Tough Replication for Multi VM
- Documentations for Multi VM

Vagrant for VM's

- No OS installations
- VM setup through images

- images are easily available in vagrant cloud

- manage VM's with a file (Vagrantfile)

- VM changes automatic through Vagrantfile

- Vagrant commands to manage VM's

- Provisioning VM/Executing commands and scripts

- many more....

Vagrant architecture
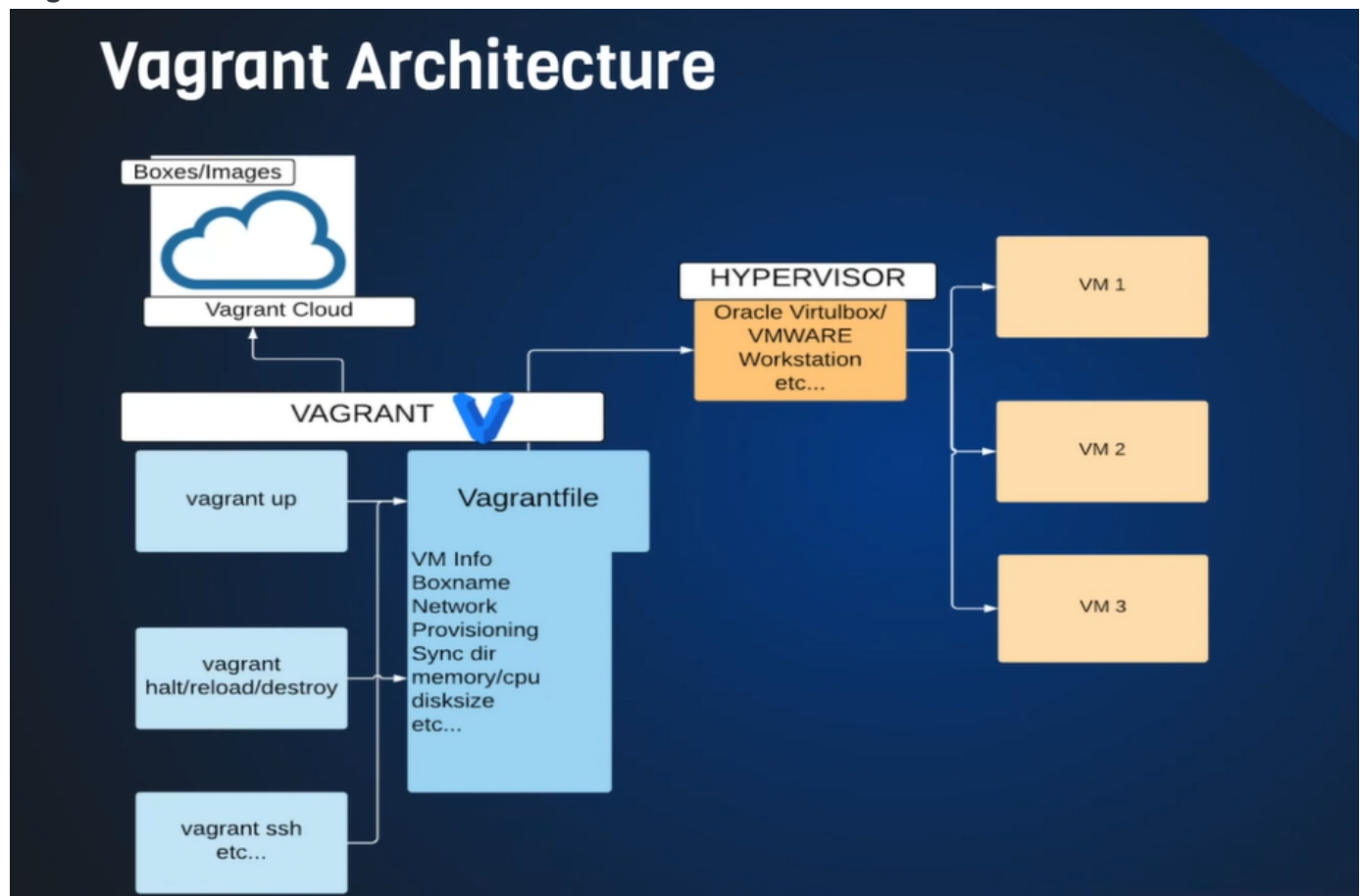
Vagrant Setup

- Virtualisation enabled in BIOS

- Vagrant tool

- Hypervisor like Oracle Virtualbox

- CLI like GIT Bash, Cygwin (Dont use command prompt)

VM setup with Vagrant

- Vagrant Box Name from https://app.vagrantup.com/boxes/search

- Project Directory (Folder/Directory at any location of your choice)

- vagrant file in project Directory

- Vagrant commands like "vagrant up"

- Login with "vagrant ssh" command for Linux VM's

Steps

1. Make a folder for vagrant vms and create subfolders inside it. (use GitBash)
2. go to vagrant cloud and copy the extension of what OS you need and paste in gitbash with the command "vagrant init geerlingguy/centos7"
3. after this you will get a VagrantFile
4. execute command "vagrant up"
5. Do the same with Ubuntu18 machine (Search on Vagrant cloud about bionic ubuntu or soemthing watch video chap 3 if confused)