UCL
Université catholique de Louvain

INGI2145 Cloud Computing

Lab 2: Introduction to Vagrant, Puppet and Docker

Muhammad Bilal

Plan for today

- Introduction to Vagrant, Puppet and Docker
  - Virtual Machines and Containers 

  - Vagrant
  - Puppet
  - Docker
- Hands on session
- Next time: Amazon Storage

Virtual Machines

- Virtual Machines are software implementation of a physical machines
- System virtual machines and Process virtual machines
- Hypervisors or VMM are used to create and run virtual machines
- Each VM runs a copy of full OS and virtual copy of all the hardware
- So …… it's SLOW!!!!

Virtual Machines: The Hypervisor

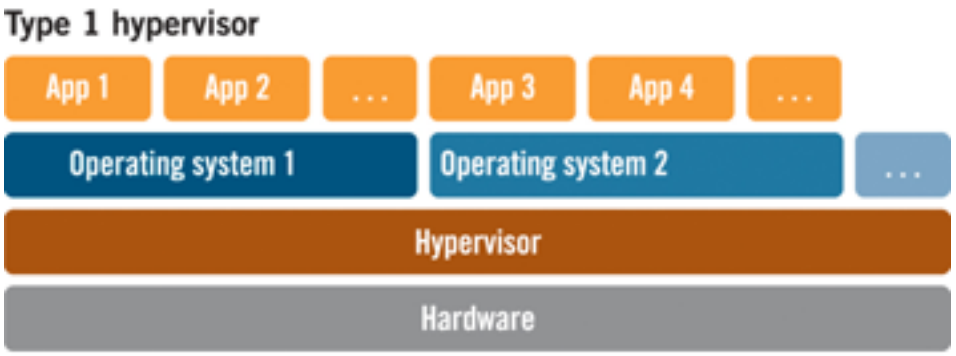


Figure 2. A Type 1 or bare-metal hypervisor sits directly on the host hardware.

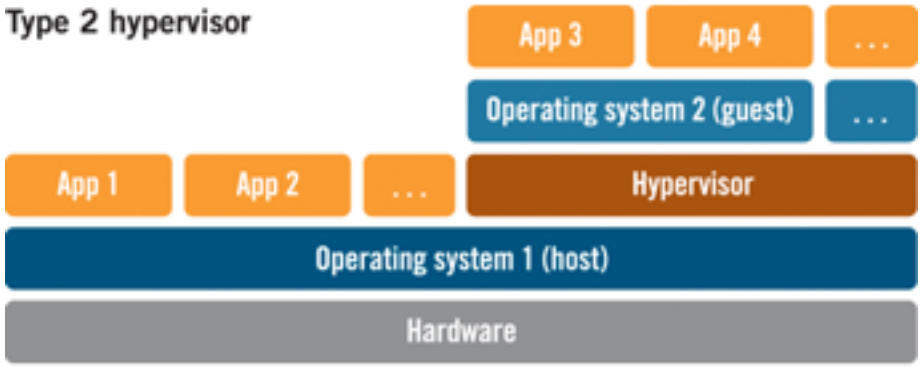


Figure 1. A Type 2 hypervisor runs as an application on a host operating system.

Containers

- Isolated execution context with isolation and limits on resource usage
- Use cgroup and namespace isolation features of Linux kernel
- Containers == abstraction of OS
- Containers != virtualized physical computer
- Think of a container as an isolated process
- Boot time ~ 1000 times faster than VMs

VM vs. Containers

- We should use containers all the time ….. right?
- Containers are fast and consume less resources but…

- Have security issues
- Only Linux based. Containers for Windows???

- Well it's in progress

Plan for today

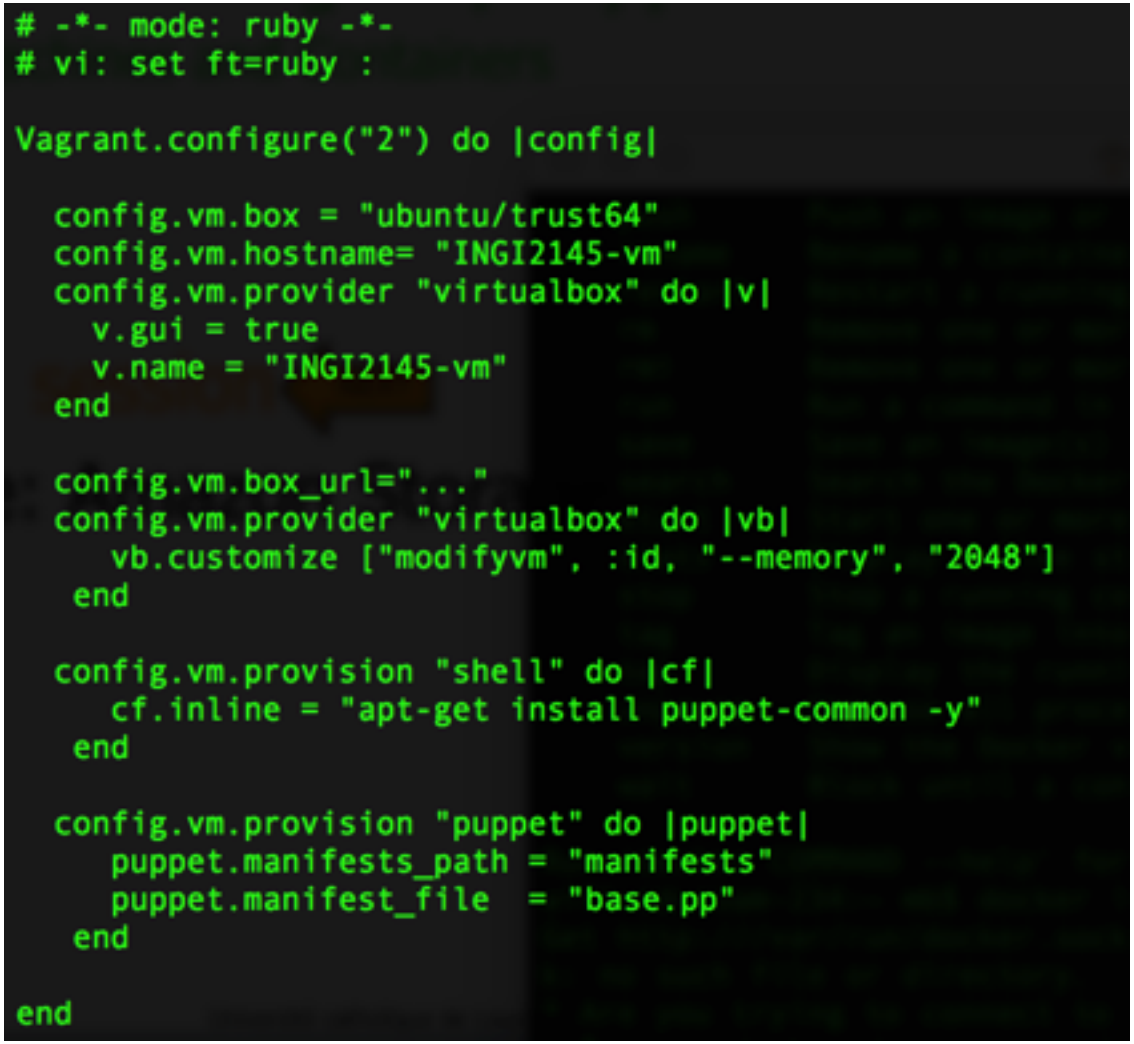
- Introduction to Vagrant, Puppet and Docker
  - Virtual Machines and Containers
  - Vagrant 

  - Docker
  - Puppet
- Hands on session
- Next time: Amazon Storage

Vagrant: An Introduction

- Creates and configures virtual development environments (using VMs)
- A high level wrapper around virtualisation and configuration management softwares
- Can be included as a part of your software project
- So, other developers can have the same development environment
- Written in Ruby

Vagrant: Example



```
# -*- mode: ruby -*-
# vi: set ft=ruby :

Vagrant.configure("2") do |config|

  config.vm.box = "ubuntu/trust64"
  config.vm.hostname= "INGI2145-vm"
  config.vm.provider "virtualbox" do |v|
    v.gui = true
    v.name = "INGI2145-vm"
  end

  config.vm.box_url="..."
  config.vm.provider "virtualbox" do |vb|
     vb.customize ["modifyvm", :id, "--memory", "2048"]
   end

  config.vm.provision "shell" do |cf|
     cf.inline = "apt-get install puppet-common -y"
   end

  config.vm.provision "puppet" do |puppet|
     puppet.manifests_path = "manifests"
     puppet.manifest_file  = "base.pp"
   end

end
```

Plan for today

- Introduction to Vagrant, Puppet and Docker
  - Virtual Machines and Containers
  - Vagrant
  - Docker 

  - Puppet
- Hands on session
- Next time: Amazon Storage

Docker: An Introduction

- An open platform for developing, shipping and running applications
- Combines lightweight container virtualisation with tools to manage and deploy applications
- Provides security and isolation
- No need for a hypervisor, so it's faster than a VM
- But …. Only recommended for single process deployment

Docker: Major Components

- Docker:

  Open source container virtualisation platform

- Docker Hub:

  SaaS platform for sharing and managing Docker containers

Docker: Architecture

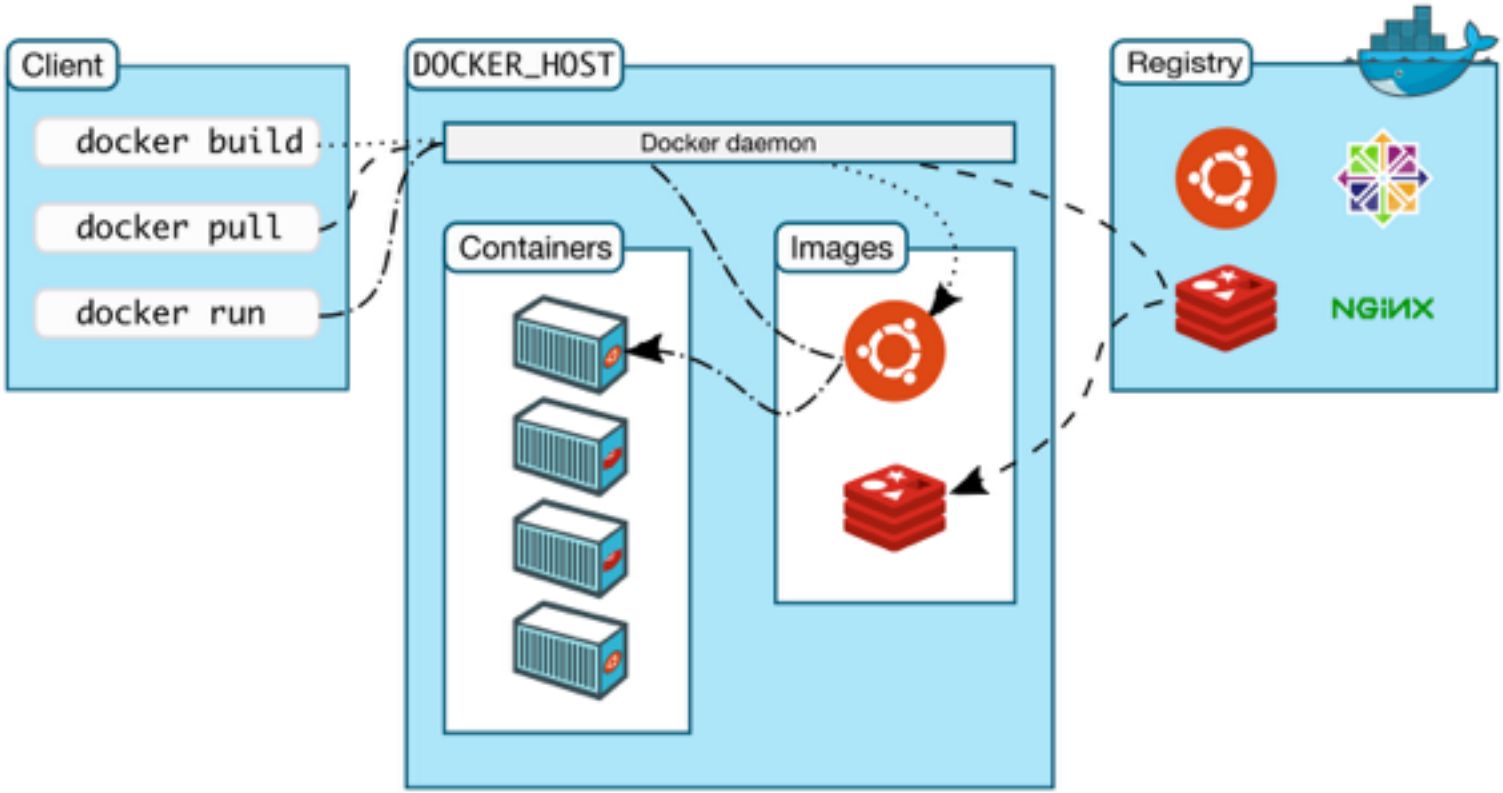


Image taken from docker.com

Docker: Internals

- Docker images
  - Read-only template used to build docker containers
- Docker registries
  - Docker registries hold images
  - Can be public or private
- Docker containers
  - Holds everything needed to run an application
  - Each container is an isolated and secure application platform

Plan for today

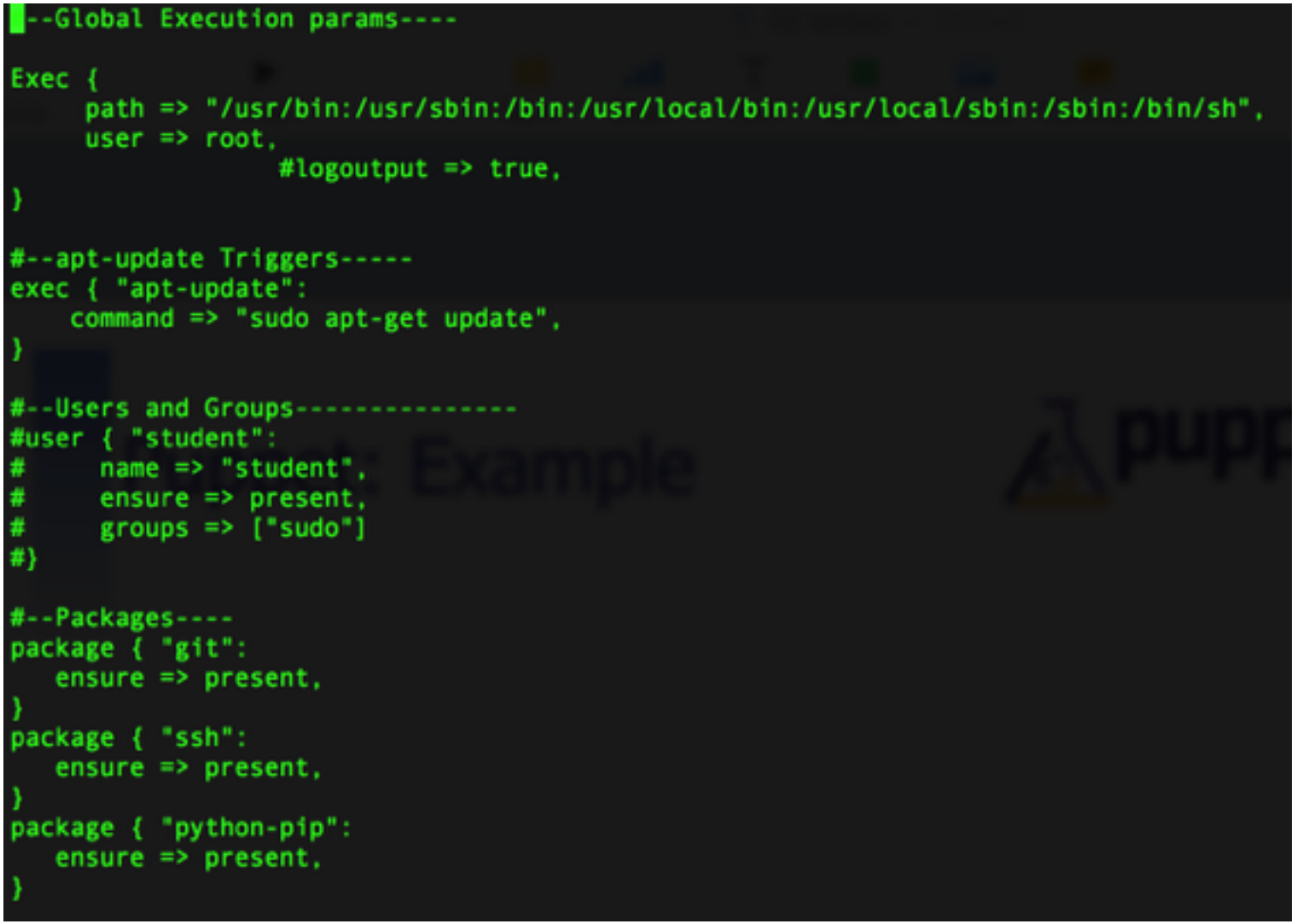
- Introduction to Vagrant, Puppet and Docker
  - Virtual Machines and Containers
  - Vagrant
  - Docker
  - Puppet 

- Hands on session
- Next time: Amazon Storage

Puppet

- Open source configuration management solution
- Has its own declarative language
- Puppet provides:
  - Provisioning physical and virtual machines
  - Orchestration and reporting
  - Early stage code development through testing
  - Product releases and updates
- In short, a system for automating system administration tasks

Puppet: Example



```
--Global Execution params----

Exec {
     path => "/usr/bin:/usr/sbin:/bin:/usr/local/bin:/usr/local/sbin:/sbin:/bin/sh",
     user => root,
                  #logoutput => true,
}

#--apt-update Triggers-----
exec { "apt-update":
    command => "sudo apt-get update",
}

#--Users and Groups---------------
#user { "student":
#     name => "student",
#     ensure => present,
#     groups => ["sudo"]
#}

#--Packages----
package { "git":
   ensure => present,
}
package { "ssh":
   ensure => present,
}
package { "python-pip":
   ensure => present,
}
```

Plan for today

- Introduction to Vagrant, Puppet and Docker
  - Virtual Machines and Containers
  - Vagrant
  - Docker
  - Puppet
- Hands on session 

- Next time: Amazon Storage

Hands on Lab

Up Next

- Introduction to Amazon Storage