

Virtualizácia a jej použitie v praxi

Boris Paráček

CESNET

13. mája 2015



- (1) Virtualizácia
- (2) Cloud
- (3) Aplikácie

Virtualizácia



- ▶ všeobecne vytvára abstrakciu zdroja za účelom
 - ▶ izolácie (bezpečnosti)
 - ▶ zdieľania (multi-tenancy)
 - ▶ plánovania (deadlock prevention, over-commitment)
 - ▶ efektívnejšieho využitia energie
 - ▶ podpory heterogénnych požiadaviek
 - ▶ ...
- ▶ napr. virtualizácia pamäte, virtualizácia hardvéru, virtualizácia na úrovni OS a aplikácií (kontajnery), ...
- ▶ *pre nás, **virtualizácia** == **HW & OS-level***

<http://bit.ly/1zTjZ8v>

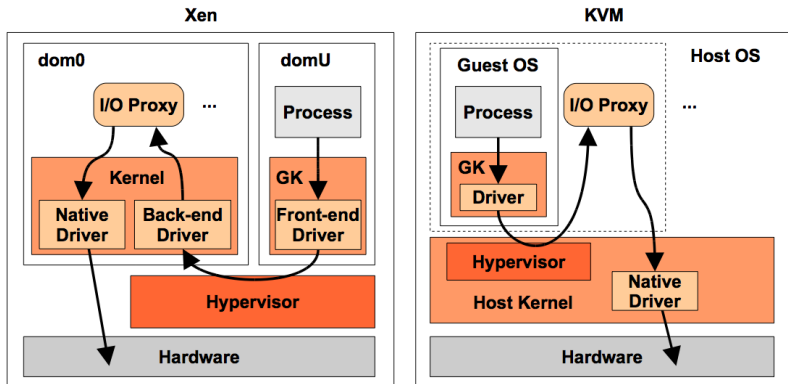
Open Source

KVM

- ▶ **K**ernel-based **V**irtual **M**achine, linuxová platforma, implementuje plnú virtualizáciu HW, modul(y) do Linux kernelu
- ▶ pre architektúru x86 s podporou rozšírení *Intel VT* alebo *AMD-V*
- ▶ postupne sa objavuje podpora ARM, PowerPC, IA64

XEN

- ▶ tzv. baremetal hypervízor, leží medzi HW a hostujúcim OS
- ▶ podporuje plnú virtualizáciu (HVM) a paravirtualizáciu(PV)
- ▶ od verzie 4.4 aj hybridnú PVH → to najlepšie z oboch (?)
- ▶ PV nevyžaduje podporu rozšírení v hardvéri hostujúceho stroja



<http://goo.gl/HbHcao>

Proprietárne/Komerčné

VirtualBox

Oracle – <https://goo.gl/EKiq6u>

vSphere

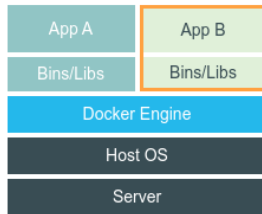
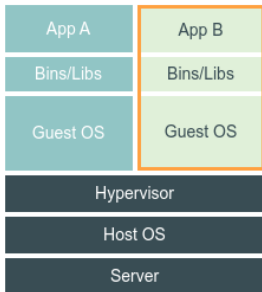
VMWare – <http://goo.gl/UjBZe5>

Parallels

Parallels, Inc. – <http://goo.gl/uDzGhN>

Hyper-V

Microsoft – <http://goo.gl/qot1TY>



- ▶ jails (BSD), zones (Solaris), OpenVZ, LXC, ...
- ▶ nie je to nový koncept, zažíva „boom“ vo svete cloudov
- ▶ abstrakcia prostredia/operačného systému pre aplikáciu
- ▶ ⇨ populárna platforma Docker (*libcontainer*), *de facto* štandard (?)



<http://www.docker.com/>



Nie je všeliek!

(výkon, správa, kompatibilita platforiem, ...)

Cloud

OpenNebula.org
The Open Source Toolkit for Cloud Computing



openstack™
CLOUD SOFTWARE

- ▶ ... ešte stále je to *buzz word!*
- ▶ všetci cloud(y) používame, či chceme alebo nechceme
- ▶ miesto s veľkým uplatnením virtualizácie (HW aj OS)
- ▶ existujú Open Source riešenia aj Ready-to-Use SP

Infrastructure

priamy prístup k virtualizovanej infraštruktúre, správa serverov a sietí

Platform

vyššia úroveň, prístup k behovým prostrediam, aplikačným serverom (DB, Web server, App Server, ...)

Service

prístup k aplikáciám a službám pre koncového užívateľa (GMail, DropBox, Twitter, ...)

Open Source

OpenStack

- ▶ framework na stavbu rozsiahlych infraštruktúr
- ▶ desiatky voliteľných komponentov, vysoká dostupnosť v návrhu
- ▶ náročné nasadenie a údržba, dobrá škálovateľnosť

OpenNebula

- ▶ cloudová platforma navrhnutá k virtualizácii datacentier
- ▶ rýchle nasadenie, jednoduchá údržba a flexibilita
- ▶ určená pre menšie a stredné IaaS infraštruktúry

Proprietárne/Komerčné

vCloud Director

VMWare – <http://goo.gl/9Ea3C8>

SoftLayer

IBM – <http://goo.gl/108Zcv>

AWS EC2

Amazon – <http://goo.gl/XhMIMO>

Microsoft Azure

Microsoft – <http://goo.gl/bzsFmQ>

Google Compute Engine

Google – <https://goo.gl/Ry1kxq>

Za moment!

Aplikácie

... alebo späť k praktickému využitiu.



Virtualizačná platforma:

- ▶ VirtualBox (Linux, Mac, Windows)
- ▶ virt-manager s KVM (Linux)

Iné:

- ▶ kvalitné pripojenie na Internet (bez dátového limitu, desiatky MB/s)
- ▶ znalosť vybraného operačného systému pre virtuálny stroj (Linux)
- ▶ skriptovanie (bash, python, ruby, ...)

Virtualbox

- ▶ Inštalácia – <https://www.virtualbox.org/>

Vagrant

- ▶ Inštalácia – <https://www.vagrantup.com/>
- ▶ Plug-iny – <https://goo.gl/dQhtTJ>

Provisioning

- ▶ <http://goo.gl/5v1TG1>
- ▶ master-less režim
- ▶ Shell, CFEngine, Salt, Docker, Chef, Puppet (+ Hiera)

```
Vagrant::Config.run do |config|
  config.vm.box = 'precise32'
  config.vm.box_url = 'http://files.vagrantup.com/precise32.box'
end
```

```
Vagrant::Config.run do |config|
  config.vm.box = 'precise32'
  config.vm.box_url = 'http://files.vagrantup.com/precise32.box'
  config.vm.host_name = 'mybox.example.com'
  config.vm.network :hostonly, '192.168.0.42'

  config.vm.customize [
    'modifyvm', :id,
    '--name', 'mybox',
    '--memory', '1024'
  ]
end
```

```
$ vagrant init
$ vagrant up
$ vagrant ssh
```

Vzorové

- ▶ HashiCorp/VagrantCloud – <https://goo.gl/h1NtYw>
- ▶ OpsCode – <http://www.vagrantbox.es/>

Veewee

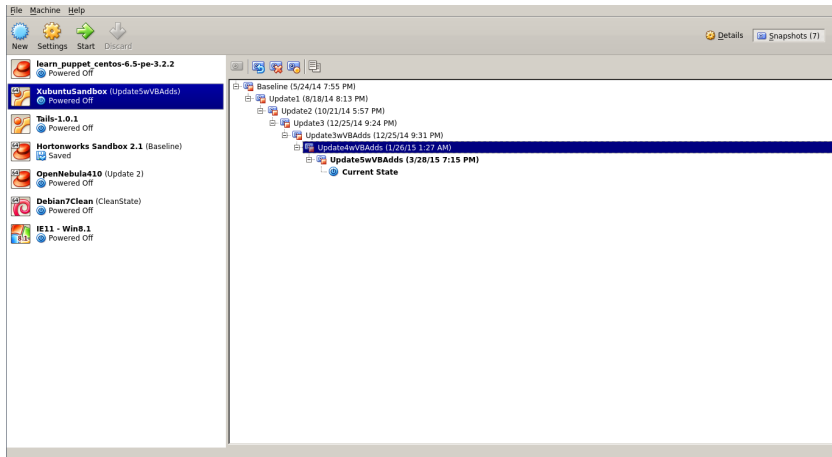
- ▶ nástroj na automatizáciu prípravy obrazov
- ▶ <https://goo.gl/nReNre>

Packer

- ▶ alternatívny nástroj, s podporou verejných cloud providerov
- ▶ <https://packer.io/>

```
{
  "provisioners": [
    {
      "type": "shell",
      "scripts": [ "scripts/base.sh" ]
    }
  ],
  "builders": [
    {
      "type": "virtualbox",
      "boot_command": [ ... ],
      "disk_size": 10140,
      "guest_os_type": "Ubuntu_64",
      "iso_md5": "af5f788aee1b32c4b2634734309cc9e9",
      "iso_url": "http://releases.ubuntu.com/12.04/ubuntu-amd64.iso",
      "ssh_username": "vagrant",
      "ssh_password": "vagrant"
    }
  ],
  "post-processors": ["vagrant"]
}
```

```
$ packer build template.json
```



Inštalácia → Snapshot → Použitie → Revert

Windows camouflage[Documentation](#)

This option makes Tails look more like Microsoft Windows XP. This may be useful in public places in order to avoid attracting suspicion.

- Activate Microsoft Windows XP Camouflage

MAC address spoofing[Documentation](#)

Spoofing MAC addresses hides the serial number of your network cards to the local networks. This can help you hide your geographical location.

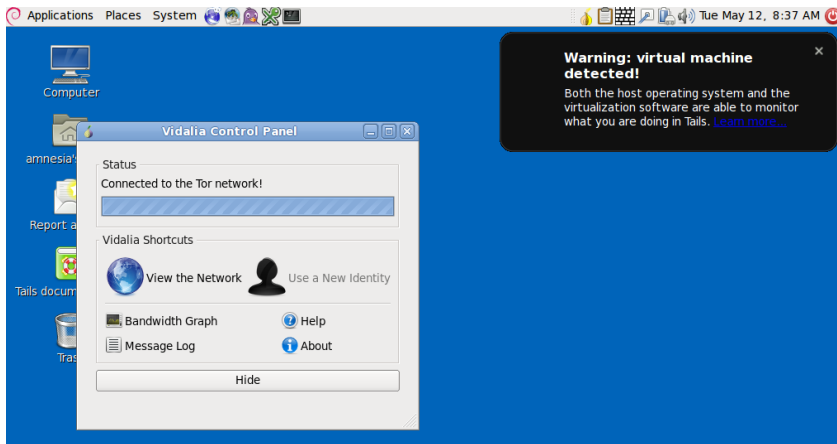
It is generally safer to spoof MAC addresses, but it might also raise suspicion or cause network connection problems.

- Spoof all MAC addresses

Network configuration[Documentation](#)

- This computer's Internet connection is clear of obstacles.
You would like to connect directly to the Tor network.

Tails – <https://tails.boum.org/>



Tails – <https://tails.boum.org/>

Všeobecné

Kitchen-Cl – <http://goo.gl/21u8Uf>

1. definovanie platforiem pre testy (YAML)
2. napísanie receptu na prípravu systému (Chef Cookbook)
3. napísanie testu (Bash Automated Testing System – *bats*)
4. converge \rightsquigarrow verify \rightsquigarrow test \rightsquigarrow destroy

Puppet moduly

Skeleton – <https://goo.gl/aLJX1L>

- ▶ „a pretty opinionated skeleton“, konvencia pred konfiguráciou
- ▶ syntax \rightsquigarrow lint \rightsquigarrow spec \rightsquigarrow acceptance
- ▶ akceptačné testy používajú *beaker*
- ▶ postavia virtualizované prostredie, skutočne aplikujú testovaný modul a overujú zmeny v systéme

```
---
driver:
  name: vagrant

provisioner:
  name: chef_solo

platforms:
  - name: ubuntu-12.04
  - name: centos-6.4

suites:
  - name: default
    run_list:
      - recipe[git::default]

# test/integration/default/bats/bc_test.bats
@test "addition using bc" {
  result="$(echo 2+2 | bc)"
  [ "$result" -eq 4 ]
}

$ kitchen list
$ kitchen test default-ubuntu-1204
```

- ▶ prostredie na automatizovanú stavbu softvérových balíkov
- ▶ prevažne pre „all-in-one“ balíky, bez obmedzení na druh projektu
- ▶ **Omnibus**
 - ▶ DSL (**D**omain-**S**pecific **L**anguage)
 - ▶ \rightsquigarrow Project DSL, definovanie metadát projektu a závislostí
 - ▶ \rightsquigarrow Software DSL, definovanie zdrojov a postupu „kompilácie“ komponentov
- ▶ kompilácia prebieha vždy pre spustenú platformu, žiadna cross-kompilácia
- ▶ integrácia s **Chef** a **Kitchen-CI** \rightsquigarrow „Build Lab“

```
name "occi-server"
maintainer "parak@cesnet.cz"
homepage "https://github.com/EGI-FCTF/rOCCI-server"

install_dir "#{default_root}/#{name}"
build_version "1.1.6"
build_iteration 1

override :rubygems, :version => '2.4.4'
override :ruby, :version => '2.1.5'

dependency "preparation"
dependency "occi-server"
dependency "version-manifest"

# add external (runtime) dependencies/services
runtime_dependency "apache2"
runtime_dependency "memcached"

exclude "**/.git"
exclude "**/bundler/git"
```

```
name "occi-server"
default_version "1.1.x"

dependency "ruby"
dependency "rubygems"
dependency "rsync"

source git: "https://github.com/EGI-FCTF/rOCCI-server"

build do
  gem "install bundler -n #{install_dir}/embedded/bin"
  gem "install passenger -n #{install_dir}/embedded/bin"
  gem "install rake -v '~> 10.3.2' -n #{install_dir}/embedded/bin"
  bundle "install --deployment --without development test"
  delete "#{install_dir}/embedded/app/rOCCI-server/vendor/bundle"
end
```

```
$ omnibus build occi-server
```

...alebo ...

```
## Chef
# Create a recipe for your project

## .kitchen.yml
# synced_folders:
#   - ['. ', '/home/vagrant/occi-server']
#
# suites:
#   - name: default
#     run_list:
#       - recipe[my::stuff]

$ kitchen list
$ kitchen test default-ubuntu-1204
```

— ??? —

- ▶ Docker – <https://www.docker.com/>
- ▶ XEN vs. KVM – <http://goo.gl/HbHcao>
- ▶ Vagrant – <https://www.vagrantup.com/>
- ▶ Packer – <https://packer.io/>
- ▶ Kitchen-Cl – <http://goo.gl/21u8Uf>
- ▶ Omnibus – <https://goo.gl/6xdVce>

– Ďakujem za pozornosť. –

parak@cesnet.cz