OSDC 2012

24th April, Nürnberg

Building Clouds with OpenNebula 3.4

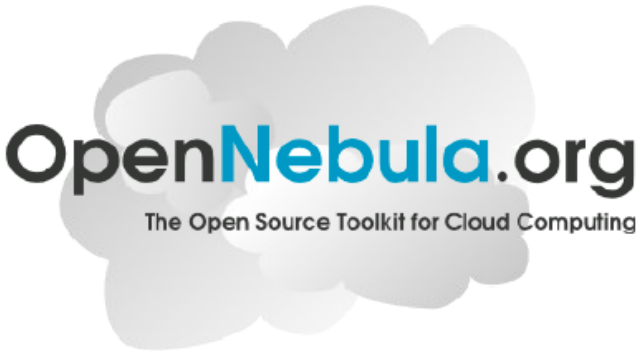
Constantino Vázquez Blanco

dsa-research.org | OpenNebula.org

Distributed Systems Architecture Research Group
Universidad Complutense de Madrid

Building Clouds with OpenNebula 3.4
Advance Usage of the Private Cloud

Constantino Vázquez Blanco
dsa-research.org | OpenNebula.org
Distributed Systems Architecture Research Group
Universidad Complutense de Madrid

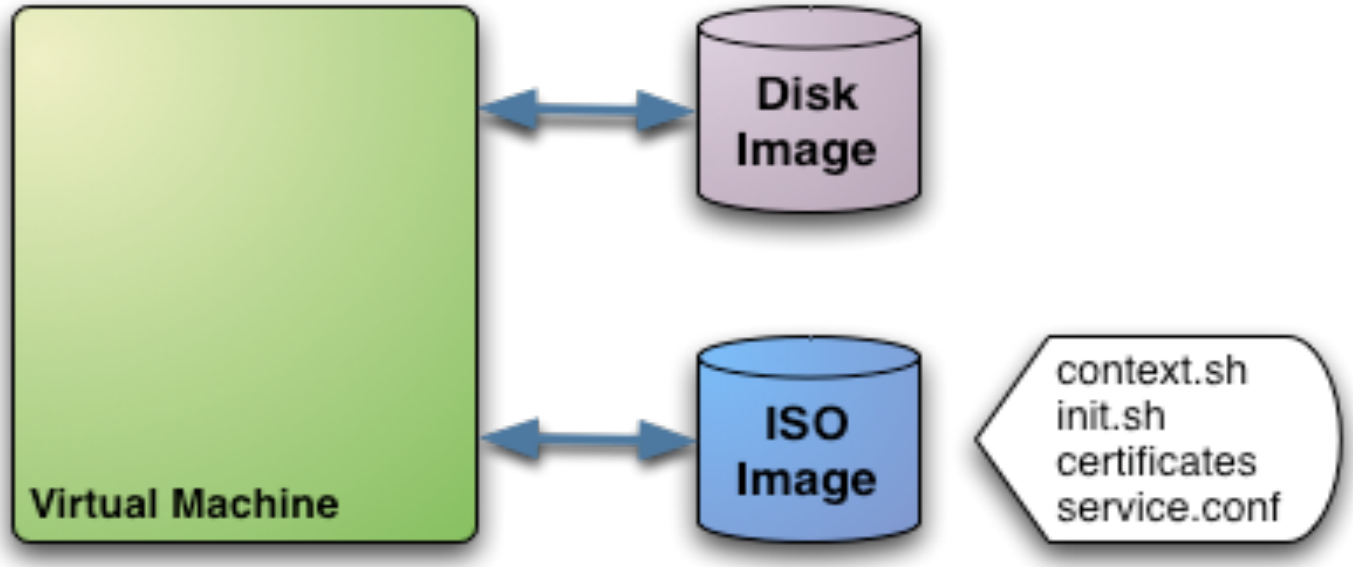


- Context for Virtual Machines
- Scheduling Virtual Machines
- Sunstone
- Groups & ACLs

Virtual Machine Context

Overview

- Block device (ISO9660) with configuration data needed at boot time
- Information includes variables and arbitrary files
- VM should be prepared to make use of context (mount + read)



Virtual Machine Context

Overview

- Context is defined in the VM template

```
#-----------------------------------
#  Context for the VM
#    values can be:
#    $<template_variable>
#    $<template_variable>[<attribute>]
#    $<template_variable>[<attribute>, <attribute2>=<value2>]
#    $<vm_id>.<context_var>
#-----------------------------------

CONTEXT = [
 var_1 = "value_1",#In context.sh as var_1="val_1" (sh syntax)
 var_n = "value_n",#In context.sh as var_N="val_N" (sh syntax)
 files = "space-separated list of paths to include in context dev",
 target= "device to attach the context device" ]
```

Virtual Machine Context

Example, create a Virtual Machine with Context

- Check the boot process of ttylinux (rc.local and vmcontext)
  - Mount context cd-rom
  - Source context.sh
  - Execute target initialization script

```
CONTEXT = [
    files       = "<path_to>/init.sh /srv/cloud/one/.ssh/id_rsa.pub",
    root_pubkey = "id_rsa.pub"
]
```

```
$ more init.sh
#!/bin/bash
. /mnt/context/context.sh
if [ -f /mnt/context/$ROOT_PUBKEY ]; then
        cat /mnt/context/$ROOT_PUBKEY >> /root/.ssh/authorized_keys
fi
```

Virtual Machine Context

Example, create a Virtual Machine with Context

- Create and define a VM with context
- Study and modify init.sh to set up hostname
- Check password-less ssh with id_rsa.pub

Scheduling Virtual Machines

Placement constraints

- Tuning the placement of VMs with the Match-making scheduler
  - First those hosts that do not meet the VM requirements are filtered out (REQUIREMENTS)
  - RANK is evaluated for the remaining hosts
  - That with the highest RANK is used for the VM
- Placement policies are specified per VM

```
#*******************************************
#           Scheduler
#*******************************************
# Use Host Monitor attributes
REQUIREMENTS = "Bool_expression_for_reqs"
RANK         = "Arith_expression_to_rank_hosts"
```

Scheduling Virtual Machines

Sample Placement Heuristics

- **Packing** (Minimize the number of cluster nodes in use)
  - **Heuristic**: Pack the VMs in the cluster nodes to reduce fragmentation
  - **Implementation**: Use those nodes with more VMs running first (RANK = RUNNING_VMS)
- **Striping Policy** (Maximize the resources available to VMs)
  - **Heuristic**: Spread the VMs in the cluster nodes
  - **Implementation**: Use those nodes with less VMs running first (RANK = "- RUNNING_VMS“)
- **Load-aware Policy** (Maximize resources)
  - **Heuristic**: Use those nodes with less load
  - **Implementation**: Use those nodes with more FREECPU first (RANK = FREECPU)

Scheduling Virtual Machines

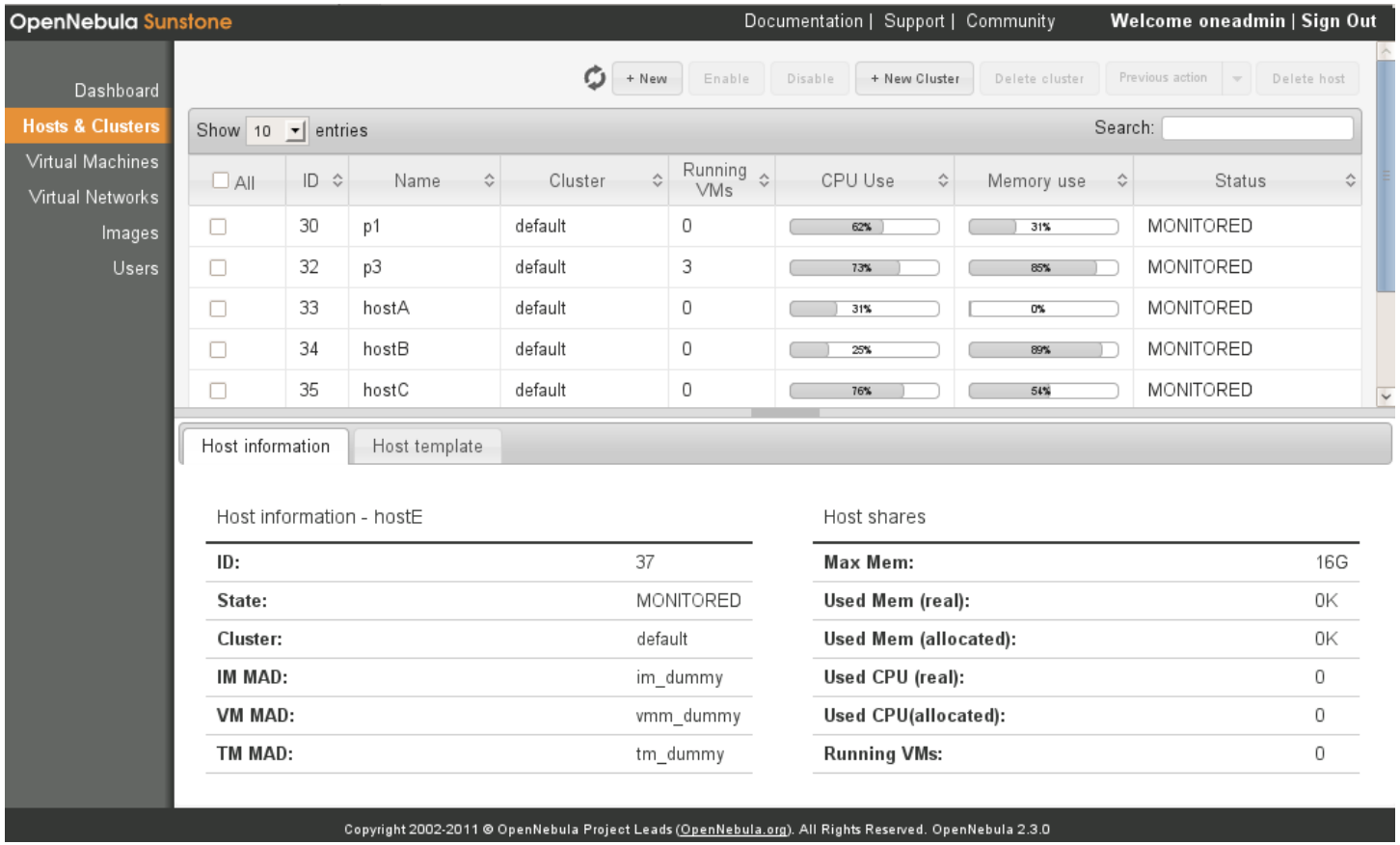
Example, guide the scheduling of the VMs

- Try VM pinning (choose a variable from onehost show) - REQUIREMENTS
- Experiment with the previous policies - RANK

Sunstone

Overview

- Web application to perform admin tasks
- Sunstone is not a public cloud user tool
- Server must have access to the XML-RPC API



Sunstone

Installation & Configuration

- Install ruby gems needed by the serve

```
# apt-get install libopenssl-ruby
# gem install json sinatra thin rack
```

- Add /var/lib/gems/1.8/bin to PATH
- Start the server as oneadmin
  - -H hostname for the server
  - -p port
  - Log information in $ONE_LOCATION/var/sunstone.log

```
$ sunstoner-server -H pcaulaXX.mydomain.com start
```

Sunstone

Example, use the GUI

- Manage the cloud (hosts, vnets, images, vms…) through sunstone

Groups & ACLs

Groups

- Administrators
- Regular Users
- Public Users
- Server Users

```
$ oneuser create otheradmin password
ID: 2

$ oneuser chgrp otheradmin oneadmin

$ oneuser list
  ID GROUP    NAME            AUTH                     PASSWORD
   0 oneadmin oneadmin        core       5baa61e4c9b93f3f06822…
   1 oneadmin serveradmin     server_c   1224ff12545a2e5dfeda4…
   2 oneadmin otheradmin      core       5baa61e4c9b93f3f06822…
```

Groups & ACLs

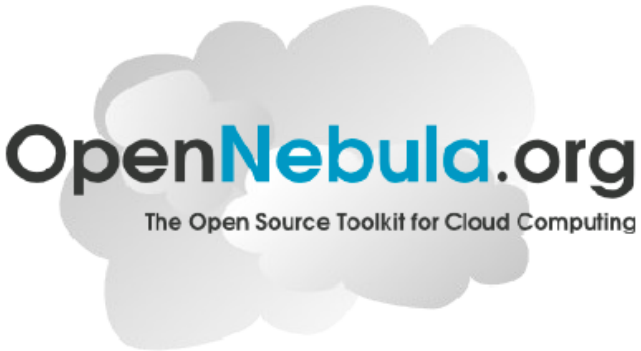
ACLs

- **USE**: Operations that do not modify the resource like listing it or using it
- **MANAGE**: Operations that modify the resource like stopping a virtual machine, changing the persistent attribute of an image or removing a lease from a network.
- **ADMIN**: Special operations that are typically limited to administrators, like updating the data of a host or deleting an user group.

```
$ onetemplate show 0
...
PERMISSIONS
OWNER           : um-
GROUP           : um-
OTHER           : u--
```

Building Clouds with OpenNebula 3.4
Advance Usage of the Private Cloud

Constantino Vázquez Blanco
dsa-research.org | OpenNebula.org
Distributed Systems Architecture Research Group
Universidad Complutense de Madrid



- Context for Virtual Machines
- Scheduling Virtual Machines
- Sunstone
- Groups & ACLs