OpenNebula, the foreman and CentOS play nice, too

Let's set up a private cloud.

Nils Domrose
Berlin, 09-May-2014
1. Cattle vs. Pets
2. Repositories, Bare-Metal Deployment OS
3. Puppet 101
4. Foreman deployment
5. OpenNebula
senior linux systems engineer at inovex GmbH

worked as a network engineer, software developer and systems engineer

using foreman for about 1 year

using bare-metal deployment for ages

life is short – let’s focus on interesting stuff
Let’s setup a private Cloud
...because everyone needs a cloud they said...

- What is it you want to achieve?
- Do you have pets or cattle?

Term “Cloud” is missing here by purpose!
Let's setup a private Cloud
...common pet Names

- build-01
- git-srv1
- lx-smb01
- scotty
- kirk
- spock
- gollum
- gandalf
- sauron
- sheldon
- leonard
- penny
- gw
- fw
- mail

(Names from popular media characters and their equivalents.)
Let’s setup a private Cloud

...common cattle Names

you
don’t
care
Let’s setup a private Cloud
…what if I can’t remember all my pet’s names?

Virtual Infrastructure

Cloud Management

oVirt

libvirt

OpenNebula

openstack

cloudstack
How to Start?
In the beginning there was bare metal...

Regardless of cattles or pets we need to get the hardware installed....
You want...

- ...long-term support
- ...HW vendor supported firmware tools für your hardware
- ...maybe benefit a bit from the RedHat Eco-System

You don‘t want...

- ...the latest greatest distro to operate your cloud on
- ...be the first to find all the errors
- ...keep hunting update issues
- ...pay for fear
You don't want to download and install an ISO....
What can pulp do for you?

- manage RPM repositories
- simple mirroring
- snapshot (branch) repositories
- create custom or mixed repositories
- manage puppet repositories (local forge)
Create a vm / setup a single system

- get pulp puppet module

https://github.com/FILIADATAGmbH/puppet-pulp

```ruby
class {'pulp': repo_enabled => true, } ->
class{'pulp::server': } ->
class{,pulp::admin':}
```

```bash
$ pulp-admin rpm repo create --repo-id=foreman --feed=http://yum.theforeman.org/releases/1.4/el6/x86_64/ --relative-url=foreman
Successfully created repository [foreman]

$ pulp-admin rpm repo sync run --repo-id=foreman
```
What you get:

- Pulp Server
- Pulp admin tools
- MongoDB (used for pulp meta data)
- qpid used as pulp internal task broker

Install the repository management (optional)

...or use the public repos if you don't need it...
You want ...

- ...it in your Datacenter!

What can the foreman do for you?

- Bare-metal deployment
- Iso based deployment
- CMDB
- Puppet ENC & dashboard
- Provision your DNS and DHCP infrastructure
- Provision your virtual datacenter or Cloud-Services
- Provide a single interface to manage your server and compute resources
- Provide an extensible platform for lifecycle management
Create a vm / setup a single system

- use puppet as early as possible via apply
- puppetize the installation later on to run on a master, too.
- The foreman installer is basically a puppet apply + hiera data

```erb
$foreman_answersfile = '/etc/foreman/foreman-installer-answers.yaml'

file {$foreman_answersfile:
  owner => 'root',
  group => 'root',
  mode => '0600',
  content => template('foreman/foreman-installer-answers.yaml.erb'),
}

exec {'foreman-installer':
  command => '/usr/bin/foreman-installer -d'
  logoutput => on_failure,
  ...
```
Bare-Metal Installer
Installing the bare-metal installer

**What you get:**
- Puppet master
- Puppet CA
- Gitolite repo
- PXE boot environment
- DHCP Server
- Foreman-Smart Proxy
- Foreman GUI
Now we need some puppet roles including the profiles for our services
Puppet 101

Puppet Module Design

Roles - Business Logic (optional)

Profiles - Technology Abstraction (optional)

Component Modules - Resource Modelling

Create roles from profiles

Create service specific profiles from component modules + config

These are puppet forge modules

Node / ENC

Hiera

optional params.pp

environment specific data

params.pp

os and module specific data

1..n

1

1

1..n

R

R

R

R

R

R

R

R

node to role classification
class profile_one {
    class {'one':
        oned => true
        sunstone => true
    }
}
Class role_one_node {
    class {'profile_base':} ->
    class {'profile_ceph ':} ->
    class {'profile_one ':
}

https://github.com/epost-dev/opennebula-puppet-module

 Might need to set:
  Encoding.default_external = Encoding::UTF_8

 Might need to set/fix defaults for $kickstart_data, $ssh_prov_keys / $ssh_pub_keys and $one_repo_enable in params.pp or via module_data or global hiera.

When we talk about open source and Cloud-Management today, we will most likely end up in an Openstack discussion.

- Openstack is an amazing Project with huge momentum
- With Openstack you can cover almost all aspects of cloud infrastructure
On the other hand Openstack is maybe moving to fast,

- Core Components are subject to change
- Manageability and clean upgrade paths just recently become a priority
- Lots of projects went live with an already outdated release
Analyze your requirements and ask yourself:

**Do I really need all the features?**

Then either stick to Openstack or consider an alternative like OpenNebula…
Installation process using the foreman

...let’s deploy some boxes

1. create new host
2. request lease
3. create DNS entries
4. request kernel & initrd
5. provision TFTP & PXE
6. create auto sign entry
7. create compute instance (optional)
8. DHCP request
9. PXE Boot
10. query unattended Resources
11. request puppet certificate
12. GET ENC & catalog, upload facts
13. notify finish

OS Repo

Smart-Proxy

DHC

DNS

Virtual / Physical Instances

virtual / physical

compute Resource

TFTP

Puppet CA

Puppet

Master

unattended Resources

FOREMAN

GET ENC & catalog, upload facts

notify finish

request puppet certificate

query unattended Resources

PXE Boot

DHCP request

create compute instance (optional)

create auto sign entry

provision TFTP & PXE

request kernel & initrd

create DNS entries

request lease

create new host
Deploying One nodes

### New Host

<table>
<thead>
<tr>
<th>Host</th>
<th>Puppet Classes</th>
<th>Network</th>
<th>Operating System</th>
<th>Parameters</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td>one01</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Location</strong></td>
<td>local</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Host Group</strong></td>
<td>myhosts</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Deploy on</strong></td>
<td>Bare Metal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Environment</strong></td>
<td>production</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Puppet CA</strong></td>
<td>foreman.local.venv.de</td>
<td></td>
<td></td>
<td></td>
<td>Use this puppet server as a CA server</td>
</tr>
<tr>
<td><strong>Puppet Master</strong></td>
<td>foreman.local.venv.de</td>
<td></td>
<td></td>
<td></td>
<td>Use this puppet server as an initial Puppet Server or to execute puppet runs</td>
</tr>
</tbody>
</table>
Deploying One nodes

Edit one01.local.venv.de

<table>
<thead>
<tr>
<th>Included Classes</th>
<th>Available Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>service_one</td>
<td>filter.classes</td>
</tr>
<tr>
<td></td>
<td>service_openstack_singlenode</td>
</tr>
<tr>
<td></td>
<td>service_realserver</td>
</tr>
<tr>
<td></td>
<td>service_test</td>
</tr>
<tr>
<td></td>
<td>service_apache</td>
</tr>
<tr>
<td></td>
<td>service_mongo_standalone</td>
</tr>
<tr>
<td></td>
<td>service_one</td>
</tr>
</tbody>
</table>
Deploying One nodes

one01.local.venv.de

Properties

<table>
<thead>
<tr>
<th>Properties</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain</td>
<td>local.venv.de</td>
</tr>
<tr>
<td>IP Address</td>
<td>192.168.122.113</td>
</tr>
<tr>
<td>MAC Address</td>
<td>52:54:00:3a:59:11</td>
</tr>
<tr>
<td>Puppet Environment</td>
<td>production</td>
</tr>
<tr>
<td>Host Architecture</td>
<td>x86_64</td>
</tr>
<tr>
<td>Operating System</td>
<td>CentOS 6.5</td>
</tr>
</tbody>
</table>
Deploying One nodes

Welcome to CentOS for x86_64

Package Installation

100%

Packages completed: 413 of 413

Installing rootfiles-8.1-6.1.el6.noarch (599 Bytes)
The basic required files for the root user’s directory

<Tab>/<Alt-Tab> between elements  |  <Space> selects  |  <F12> next screen
Accessing your new cloud

Dashboard

Virtual Machines
- Total: 6
- Active: 6
- Pending: 0
- Failed: 0

REAL CAPACITY USAGE
- CPU: 0%
- Memory: 100%

Hosts
- Total: 4
- On: 4
- Off: 0
- Error: 0

CPU
- Allocated: 35GB / 180GB (20%)
- Real: 10GB / 100GB (10%)

MEMORY
- Allocated: 3.1GB / 31.1GB (10%)
- Real: 977MB / 31.1GB (3%)

Storage
- Total: 8
- Images: 36.6GB

Users
- Total: 3
- Users: 3
- Groups: 2

Network
- Total: 1
- VNets: 1
- Used IPs: 6
Accessing your new cloud
Accessing your new cloud

<table>
<thead>
<tr>
<th>ID</th>
<th>Owner</th>
<th>Group</th>
<th>Name</th>
<th>Status</th>
<th>Host</th>
<th>IPs</th>
<th>VNC</th>
</tr>
</thead>
<tbody>
<tr>
<td>104</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-104</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.243</td>
<td></td>
</tr>
<tr>
<td>103</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-103</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.242</td>
<td></td>
</tr>
<tr>
<td>102</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-102</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.241</td>
<td></td>
</tr>
<tr>
<td>101</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-101</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.240</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-100</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.239</td>
<td></td>
</tr>
<tr>
<td>99</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-99</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.238</td>
<td></td>
</tr>
<tr>
<td>98</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-98</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.237</td>
<td></td>
</tr>
<tr>
<td>97</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-97</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.236</td>
<td></td>
</tr>
<tr>
<td>96</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-96</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.235</td>
<td></td>
</tr>
<tr>
<td>95</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-91</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.234</td>
<td></td>
</tr>
<tr>
<td>94</td>
<td>oneadmin</td>
<td>oneadmin</td>
<td>CentOS 6.3-90</td>
<td>PENDING</td>
<td></td>
<td>172.24.1.233</td>
<td></td>
</tr>
</tbody>
</table>

Total: 11
Pending: 8
Active: 3
Off: 0
Failed: 0
Accessing your new cloud

Dashboard

Virtual Machines
- 6 TOTAL
- 6 ACTIVE
- 0 PENDING
- 0 FAILED

Real Capacity Usage:
- 0% CPU
- 100% MEMORY

Hosts
- 4 TOTAL
- 4 ON
- 0 OFF
- 0 ERROR

CPU:
- 350 / 1800 (22%) Allocated
- 10 / 1000 (1%) Real

MEMORY:
- 4.3GB / 31.1GB (14%) Allocated
- 977.3MB / 31.1GB (3%) Real

Storage
- 8 IMAGES
- 36.6GB USED

Users
- 3 USERS
- 2 GROUPS

Network
- 1 VNET
- 6 SUBNETS

OpenNebula 4.6.0 by Cloudera EBS
The big picture

- **Physical Network**
- **Virtual Network**
- **Hosted**
  - EC2
  - Google Compute Engine
- **Virtual**
  - OpenStack
  - OpenNebula
  - VMware vSphere 5.0
  - oVirt
- **Real**
  - Physical Network
- **CMDB**
- **Deployment Lifecycle Mgmt.**
- **Deployment & Configuration Management**
- **Foreman REST API**
- **Jenkins**
- **(R)ex**
There are still some rough edges in the modules used

Good news – they are of minor type

Currently you cannot use OpenNebula as a compute resource due to missing fog support

You can however use OpenNebula to provision hosts on foreman for bare-metal deployment of instances – see: http://opennebula.org/opennebula-and-foreman-integration/
Contact

Nils Domrose
Senior System Engineer
Linux

inovex GmbH
Office Cologne

nils.domrose@inovex.de