foreman_provision

Infrastructure as Code

Gent, Nils Domrose
02.02.2016
About me

Head of IT @ inovex GmbH

- Systems Engineer
- Automation
- Architecture
- Security
- Network Engineering

@endyman
## The Problem

<table>
<thead>
<tr>
<th>Story</th>
<th>ToDo</th>
<th>In Progress</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build Service</td>
<td>Create Hosts in foreman</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


# The Problem

<table>
<thead>
<tr>
<th>Story</th>
<th>ToDo</th>
<th>In Progress</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build Service</td>
<td></td>
<td>Create Hosts in foreman</td>
<td></td>
</tr>
</tbody>
</table>
The Problem

### New Host

<table>
<thead>
<tr>
<th>Host</th>
<th>Puppet Classes</th>
<th>Interfaces</th>
<th>Operating System</th>
<th>Parameters</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name *</td>
<td></td>
<td></td>
<td></td>
<td>This value is used also as the host’s primary interface name.</td>
<td></td>
</tr>
<tr>
<td>Organization *</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Host Group</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Deploy on</td>
<td>Bare Metal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Environment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Puppet CA</td>
<td></td>
<td></td>
<td></td>
<td>Use this puppet server as a CA server</td>
<td></td>
</tr>
<tr>
<td>Puppet Master</td>
<td></td>
<td></td>
<td></td>
<td>Use this puppet server as an initial Puppet Server or to execute puppet runs</td>
<td></td>
</tr>
</tbody>
</table>
The Problem
The Problem

New Host

- Architecture: x86_64
- Operating system: CentOS 7.1
- Build mode: CentOS 7.1
- Media: CoreOS 766.5.0, Debian 6.0.7
- Partition table: Debian 7.6, Debian 7.7, Debian 7.8

What ever text (or ERB template) you use in here, would be used as your OS disk layout options if you want to use.
The Problem

New Host

<table>
<thead>
<tr>
<th>Host</th>
<th>Puppet Classes</th>
<th>Interfaces</th>
<th>Operating System</th>
<th>Virtual Machine</th>
<th>Parameters</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPUs</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cores per socket</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Memory (MB)</td>
<td>768</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cluster</td>
<td>ESXi Cluster</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Folder</td>
<td>VM Folder</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Guest OS</td>
<td>Other (32-bit)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCSI controller</td>
<td>LSI Logic Parallel</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Virtual H/W version</td>
<td>Default</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Start
Power ON this machine
The Problem

repeat the last steps for each system....

• 2 Loadbalancers
• 2 Webservers
• 2 Middleware Servers
• 2 Keyvalue Stores
= 8 Systemes
The Problem

... in each environment

- Dev
- Demo
- Stage
- Production
= 36 Systems*

*with smallest prod sizing
The Problem

<table>
<thead>
<tr>
<th>Story</th>
<th>ToDo</th>
<th>In Progress</th>
<th>Done</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build Service</td>
<td></td>
<td></td>
<td>Create Hosts in foreman</td>
</tr>
</tbody>
</table>
The Problem

RELYING ON HUMANS TO DO A REPETITIVE TASK IS A BAD IDEA

http://www.memgenerator.net
The Solution

```json
{}
ip: null,
environment_id: 6,
environment_name: "inovex_default",
last_report: "2015-08-27T08:02:57Z",
mac: "00:50:56:9c:80:b0",
realm_id: null,
realm_name: null,
sp_mac: null,
sp_ip: null,
sp_name: null,
domain_id: 1,
domain_name: "adm.ka-tmx.inovex.de",
architecture_id: 1,
architecture_name: "x86_64",
operatingsystem_id: 1,
operatingsystem_name: "Debian 7.8",
subnet_id: null,
subnet_name: null,
sp_subnet_id: null,
phtable_id: null,
phtable_name: null,
```
Working with the API

Ruby bindings for Forman's rest API

- 97 commits
- 3 branches

Branch: master

New pull request

domcleal Merge pull request #29 from kbrock/warning_notice

- bin
  - Yard documentation
- doc
  - Updated to latest upstream
foreman_provision
Getting Started

```sh
1 # yum install ruby-devel git gcc-c++
2 # git clone https://github.com/FILIADATAGmbH/foreman_provision.git
3 # cd foreman_provision/
4 # gem install --verbose --no-document foreman_api
```

```sh
1 # ./bin/foreman-provision -h
2 Usage: foreman-provision -c conf/config.yaml -a conf/foreman.yaml -v -d
3 -c, --config_file FILE|DIR location of the configuration file or a directory containing *.YAML files
4 -a, --auth_credentials FILE location of the foreman auth credentials YAML file
5 -v, --[no-]verbose Run verbosely
6 -l, --log_file FILE location of the log file
7 -d, --[no-]debug Run in debug mode
8 -t, --[no-]test Only show what would have been executed in no-test mode
```
Accessing the API

```
# cp conf/auth.yaml.sample conf/foreman.yaml
# cat conf/foreman.yaml
---
:base_url: https://127.0.0.1
:oath:
:consumer_key: qdoDRaienP8396KwUKRKZ2HsafxVt
:consumer_secret: 65GE3MWh2oDq8PtfwWiUC5CSPqsVpzBPw
:headers:
:foreman_user: admin
:options:
:verify ssl: 0
```
# cat conf/config.yml
---
subnets:
- name: default
  network: 10.10.30.0
  mask: 255.255.255.0
  gateway: 10.0.2.15
  from: 10.10.30.200
  to: 10.10.30.250
  dns_primary: 10.10.30.100
  dhcp_proxy: foreman.vagrant.inovex.de
  tftp_proxy: foreman.vagrant.inovex.de
  dns_proxy: foreman.vagrant.inovex.de
  domain_names:
    - vagrant.inovex.de
Provisioning Data

```yaml
:domains:
- name: vagrant.inovex.de
dns_proxy: foreman.vagrant.inovex.de
 fullname: Main Domain

:hostgroups:
- name: Test-Hosts
 architecture: x86_64
 operatingsystem: CentOS 7.0
 medium: CentOS mirror
 ptable: Kickstart default
 environment: production
 # Requires imported classes:
 #:puppetclasses:
 # - stdlib
 subnet: default
 domain: vagrant.inovex.de
 puppet_proxy: foreman.vagrant.inovex.de
 puppet_ca_proxy: foreman.vagrant.inovex.de
 root_pass: $1$HBwWOy7F$BtZTzR5x1T.nikDzxLusu1 # PleaseChangeMeSoon!
```
Provisioning Data

hosts:
- name: test1.vagrant.inovex.de
  hostgroup: Test-Hosts
  mac: 06:10:9f:dc:aa:a1
- name: test2.vagrant.inovex.de
  hostgroup: Test-Hosts
  mac: 06:10:9f:dc:aa:a2
- name: test3.vagrant.inovex.de
  hostgroup: Test-Hosts
  mac: 06:10:9f:dc:aa:a3
Dry-Run

```
# ./bin/foreman-provision --auth_credentials=conf/foreman.yaml --config_file=conf/config.yaml --verbose --test
```

```
INFO -- : Running provision in test mode
INFO -- : Would have created Foreman Provision::Subnet {::dhcp_proxy=>"foreman.vagrant.inovex.de", :dns_primary=>"10.10.30.100", :dns_proxy=>"foreman.vagr...
INFO -- : Would have created Foreman Provision::Domain {::dns_proxy=>"foreman.vagrant.inovex.de", :name=>"vagrant.inovex.de", :ensure=>"present"}
INFO -- : Would have created Foreman Provision::Hostgroup {::architecture=>"x86_64", :domain=>"vagrant.inovex.de", :environment=>"production", :medium=>"Core...
INFO -- : Would have created Foreman Provision::Host {::build=>true, :hostgroup=>"Test-Hosts", :mac=>"06:10:9f:dc:aa:a1", :name=>"test1.vagrant.inovex.de"}
```
Provisioning

```
1 # /bin/foreman-provision --auth_credentials=conf/foreman.yaml --config_file=conf/config.yaml --verbose
2 INFO : Creating ForemanProvision::Subnet "default"
3 INFO : Creating ForemanProvision::Domain "vagrant.inovex.de"
4 INFO : Creating ForemanProvision::Hostgroup "Test-Hosts"
5 INFO : Creating ForemanProvision::Host "test1"
6 INFO : Creating ForemanProvision::Host "test2"
7 INFO : Creating ForemanProvision::Host "test3"
8 INFO : Creating ForemanProvision::Host "test4"
9 INFO : Creating ForemanProvision::Host "test5"
10 INFO : Creating ForemanProvision::Host "test6"
11 INFO : Creating ForemanProvision::Host "test7"
12 INFO : Creating ForemanProvision::Host "test8"
```
### Result

#### Hosts

<table>
<thead>
<tr>
<th>Name</th>
<th>Operating system</th>
<th>Environment</th>
<th>Model</th>
<th>Host group</th>
<th>Last report</th>
</tr>
</thead>
<tbody>
<tr>
<td>test1.vagrant.inovex.de</td>
<td>CentOS 7.0</td>
<td>production</td>
<td></td>
<td>Test-Hosts</td>
<td></td>
</tr>
<tr>
<td>test2.vagrant.inovex.de</td>
<td>CentOS 7.0</td>
<td>production</td>
<td></td>
<td>Test-Hosts</td>
<td></td>
</tr>
<tr>
<td>test3.vagrant.inovex.de</td>
<td>CentOS 7.0</td>
<td>production</td>
<td></td>
<td>Test-Hosts</td>
<td></td>
</tr>
<tr>
<td>test4.vagrant.inovex.de</td>
<td>CentOS 7.0</td>
<td>production</td>
<td></td>
<td>Test-Hosts</td>
<td></td>
</tr>
<tr>
<td>test5.vagrant.inovex.de</td>
<td>CentOS 7.0</td>
<td>production</td>
<td></td>
<td>Test-Hosts</td>
<td></td>
</tr>
<tr>
<td>test6.vagrant.inovex.de</td>
<td>CentOS 7.0</td>
<td>production</td>
<td></td>
<td>Test-Hosts</td>
<td></td>
</tr>
<tr>
<td>test7.vagrant.inovex.de</td>
<td>CentOS 7.0</td>
<td>production</td>
<td></td>
<td>Test-Hosts</td>
<td></td>
</tr>
<tr>
<td>test8.vagrant.inovex.de</td>
<td>CentOS 7.0</td>
<td>production</td>
<td></td>
<td>Test-Hosts</td>
<td></td>
</tr>
</tbody>
</table>
VMware? VMware!

```yaml
---
# vSphere
:hosts:
  - :name: test11.local.venv.de
    :hostgroup: 'vsphere'
    :compute_resource: 'vsphere_local'
    :architecture: 'x86_64'
    :ptable: 'RedHat LVM'
    :domain: local.venv.de
    :subnet: 'local network'
    :operatingsystem: 'RedHat 6.5'
    :environment: 'production'
    :build: 1
    :compute_attributes:
      :cpus: 1
      :start: "1"
      :cluster: 'ESX'
    :path: '/Datencenter/TEST/prod' # from compute_resource edit screen view source
    :memory_mb: 768
    :interfaces_attributes:
      0:
        :network: 'dvportgroup-100404' # from compute_resource edit screen view source
    :volumes_attributes:
      0:
        :datastore: MY_SAN
        :name: 'Hard disk'
        :size_gb: 5
        :thin: true
    :puppetclasses:
      - stdlib
    :location: LAN
```
More Examples

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>common_params.yaml</td>
<td>refactor provision.rb, add inheritance</td>
</tr>
<tr>
<td>domains.yaml</td>
<td>refactor provision.rb, add inheritance</td>
</tr>
<tr>
<td>hostgroups.yaml</td>
<td>refactor provision.rb, add inheritance</td>
</tr>
<tr>
<td>hosts_kvm.yaml</td>
<td>refactor provision.rb, add inheritance</td>
</tr>
<tr>
<td>hosts_vsphere.yaml</td>
<td>refactor provision.rb, add inheritance</td>
</tr>
<tr>
<td>params.yaml</td>
<td>refactor provision.rb, add inheritance</td>
</tr>
<tr>
<td>proxies.yaml</td>
<td>refactor provision.rb, add inheritance</td>
</tr>
</tbody>
</table>
Limitations

- Currently Objects can only be checked or created
  - no update
  - no delete
- Fragile to changes in compute resources (fog updates)
- Issue #14 - Use apipie bindings
Possible Alternatives

- theforeman/foreman_deployments
- Terraform Provider
  https://github.com/mattwilmott/terraform-foreman
Deployment Pipeline

Links

Arnold‘s inovex Blog Post:  
https://goo.gl/hnAfu3

Source Code @ Github:  
https://github.com/FILIADATAGmbH/foreman_provision

Slidesshare:  
http://de.slideshare.net/inovex
We are hiring

Become an inovexpert

http://www.inovexperts.com
Thank You!

Nils Domrose
Head of IT

inovex GmbH
Ludwig-Erhard-Allee 6
76131 Karlsruhe