nVentory – The Source of Truth *

Christopher Nolan
Jeffrey Roberts
Darren Dao
eHarmony, Inc.
nventory

http://nventory.sourceforge.net

Currently used and maintained by employees of AT&T Interactive (yp.com) and eHarmony (including Darren Dao)

Demo: http://nventory.slacklabs.com
Infrastructure growth and complexity

Traditional

Cloud

Rapid Growth and Scaling Problems
Other Sources of Truth

Spreadsheet
Static file
DNS
Hostname
MySQL/Postgres
Custom solution
A More Automated Approach (though we went a different route)

Puppet and Chef

Enterprise solutions

Roll your own
Why We Chose nVentory

Centralized
Detailed
Metadata
Multiple access methods
  API
  CLI
  Browser
History
Automated
Open Source
DevOps?

It's essential that all individual tools be considered part of a larger toolchain that spans the entire Development to Operations lifecycle...

Damon Edwards – President DTO Solutions

Your tools should all talk to a master
Everyone should know where to go for answers
nVentory provides the foundation
You have to build the house
What/Who leverages inventory at eHarmony

Chef-solo (app config management)
Etch (system config management)
Jenkins (continuous integration and release)
Self service VMs (custom private cloud)
Monitoring and metrics (splunk)
Finance (audits)
QA, Operations, and Engineering
Got Objects?
Other Objects

- status
- hardware_profile
- operating_system
- network_interface
- ip_address
- vip
- lb_pool
- tags & graffiti
nVentory Server

• Uses MVC pattern
• Heavy usage of ActiveRecord
• For each object, there are corresponding model, view and controller for it
• RESTful API
## nVentory Server

<table>
<thead>
<tr>
<th>HTTP ACTION</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td><a href="http://mynventoryserver/nodes">http://mynventoryserver/nodes</a></td>
</tr>
<tr>
<td>POST</td>
<td><a href="http://mynventoryserver/node">http://mynventoryserver/node</a></td>
</tr>
<tr>
<td>PUT</td>
<td><a href="http://mynventoryserver/node/123">http://mynventoryserver/node/123</a></td>
</tr>
<tr>
<td>DESTROY</td>
<td><a href="http://mynventoryserver/node/123">http://mynventoryserver/node/123</a></td>
</tr>
</tbody>
</table>
nVentory Server

• How do we set it up?
  – nginx
  – thin
  – mysql
  – Configured via Chef
  – Deployed via Jenkins
nVentory Client

Ruby & Perl clients for

.NET client available for Windows as well...
nVentory Client

Main purposes

– Collect data and send back to server
– CLI to query/update data
nVentory Client

Uses various tools to gather host/hardware info

- dmidecode
- sar
- facter Ruby gem (from Puppet Labs)
- racadm
- omreport
- virsh (libvirt)
- ethtool
nVentory Client

• nv --register command
  – HTTP GET to see if the node already exists
  – HTTP POST to create node if not
  – HTTP PUT to update node if existed
What is SCALE?
The Southern California Linux Expo (SCALE) is an annual Linux, Open-Source, and Free Software conference held in Los Angeles. Now celebrating its tenth year, this community organized event will be held January 20-22, 2012 at the Hilton Los Angeles Airport hotel. SCALE offers over 100 seminars and presentation sessions, and an exhibit hall where non-profit and commercial organizations will demonstrate the latest developments in the open-source and free software realm.

Securing Communications at
How we set up the client?

– Install on each box at build time (kickstart)
– cronjob to auto-register hourly
Using nVentory Client

- `nv --name myhost`
- `nv --name myhost --fields operating_system`
- `nv --name myhost --allfields`
- `nv --get node_group[name]=hd-datanodes`
- `nv --name myhost --set status=outofservice`
Using nVentory API

```
require 'nventory'

nvclient = NVentory::Client.new

# query nodes with hadoop in their hostnames
getdata = {:objecttype => 'nodes',
           :get => {:name => 'hadoop'}}

result = nvclient.get_objects(getdata)
puts result.keys
```
Demonstration
How does eHarmony’s implementation of nVentory make my life better?

Hugely simplifies tasks that can be frustratingly time consuming

- User Management
- Actions to a group of machines
- Configuration management
- Discovery of node related details such as hardware type, location information, etc
Integration with Configuration Mngt

• Changes can be made to a single machine or a large group of machines with the same amount of effort

• Changes are visible immediately or overtime via automated methods
/etc/etchserver/trunk/source/etc/passwd/config.xml

<config>
  <file>
    <perms>644</perms>
    <source>
      <plain group="demo-orig">passwd.orig</plain>
      <plain group="demo-users">passwd.demo</plain>
    </source>
  </file>
</config>
Tasks on multiple machines

• Using the API I can write scripts that can pull all the machine in a given node group and then act on them

• NV can be used from the command line to generate lists that can be used in *for loops*
## Nodes

**New Node**

Displaying all 4 nodes

<table>
<thead>
<tr>
<th>Name</th>
<th>Aliases</th>
<th>Node Groups</th>
<th>Status</th>
<th>Hardware Profile</th>
<th>Operating System</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>View / Edit</strong></td>
<td>host1.eharmony.com</td>
<td></td>
<td>inservice</td>
<td>Dell Inc. PowerEdge M610</td>
<td>CentOS (Final) 5.5 x86_64</td>
</tr>
<tr>
<td><strong>View / Edit</strong></td>
<td>scaledemo.np.dc1.eharmony.com</td>
<td>password-orig</td>
<td></td>
<td>Red Hat KVM</td>
<td>CentOS (Final) 5.5 x86_64</td>
</tr>
<tr>
<td><strong>View / Edit</strong></td>
<td>scalestch.np.dc1.eharmony.com</td>
<td></td>
<td>setup</td>
<td>Red Hat KVM</td>
<td>CentOS (Final) 5.5 x86_64</td>
</tr>
<tr>
<td><strong>View / Edit</strong></td>
<td>scalanev.np.dc1.eharmony.com</td>
<td></td>
<td>setup</td>
<td>Red Hat KVM</td>
<td>CentOS (Final) 5.5 x86_64</td>
</tr>
</tbody>
</table>

Export CSV File

Advanced CSV Export

Permissions
scaledemo.np.dc1.eharmony.com $
nVentory

http://nventory.sourceforge.net
Demo: http://nventory.slacklabs.com
Questions?