

Orchestrate your cloud with
ansible



Who am I



- Linux consultant with an interest in openstack and cloud
- Worked as sysadm, pythonprogrammer, appl support
- RHCE, COA, RHC specialist Ansible



Scope of this presentation



- Ansible basics
- Ansible playbook as blueprint (IaC)
- Ansible on openstack



Ansible basics



- The inventory

Localhost

```
[webservers]
jupiter.cloud.net
earth.cloud.net
```

```
[db]
saturn.cloud.net
```

```
[webservers:vars]
https_port=8443
```

Inventory cont.



- Executables returning JSON
- Mix of static and dynamic sources
- `Ls <inventory dir>`
`dynscript.py*`
`ovirtinventory.py*`
`solarsystem`



SSH and sshkeys



- Remote ansible user "needs" passwordless login and passwordless sudo privs
- Remote user and private key could be given on cmdline or set in:
 - /etc/ansible/ansible.cfg
 - \$HOME/.ansible.cfg
 - \$PWD/ansible.cfg
 - \$ANSIBLE_CONFIG



What can not be done



- Ansible needs a running system to work against
- No bare metal or PXE features



What about ansible and openstack



- API-modules

https://docs.ansible.com/ansible/latest/modules/list_of_cloud_modules.html

- Clouds.yaml

<https://docs.openstack.org/python-openstackclient/pike/configuration/index.html>

- Dynamic_inventory

<https://github.com/ansible/ansible/tree/devel/contrib/inventory>



Can we deploy now ?



- Set up the deployment host (ansible host) on CentOS 7
 - centos-release-openstack-rocky
 - python-openstacksdk
 - python2-openstackclient
 - ansible



Two step process



- Step 1: Deploy servers in cloud with API
- Step 2: Deploy apps on those servers



Playbook blueprint



Localhost ansible_connection=local

[company]

<all hosts>

[web]

<all hosts running webserver>

[app1]

<host or hosts running app1>

[app2]

<host or hosts running app2>

...

Differences in the cloud



- Firewall rules
- Backup routines
- Security group rules
- Snapshot



Live Demo – Keep your fingers crossed



- Add project
- Add user
- Add roles (two)
- Security group w. rules
- Upload keypair
- Get images
- Start three servers
- Install python2 in bionic servers
- Add ansible user
- Deploy blueprint



Apply your blueprint



- Once the servers are up and prepared with ansible user you can apply your playbooks as you would have if not in the cloud

