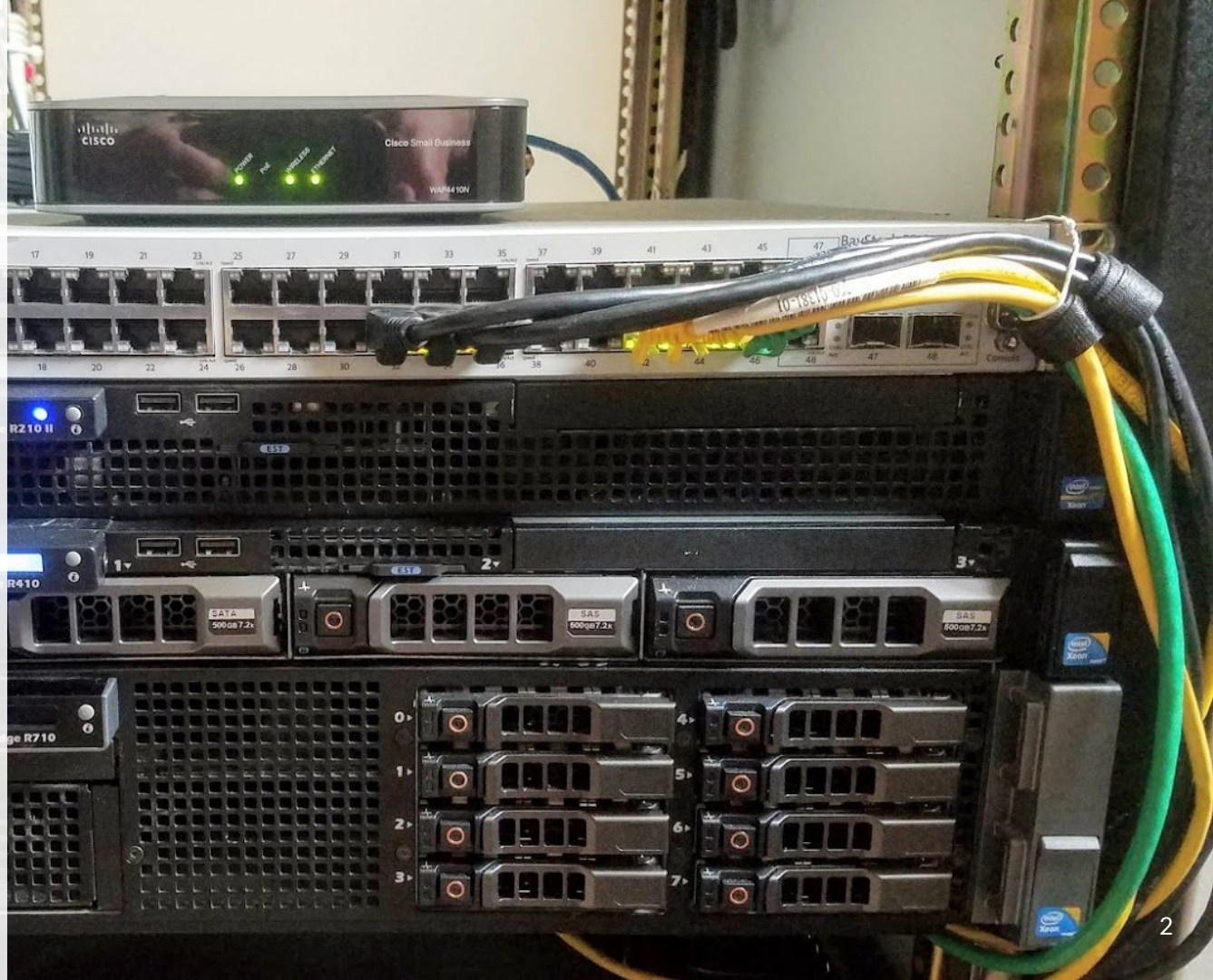


Foolproof Ansible Playbooks with Molecule

Nathaniel Beckstead

Nathaniel Beckstead

- Automation extraordinaire
- Never enough infrastructure
- Favorite ice cream flavor is mint chocolate chip



Ansible

Ansible

- Configuration management
 - Manage servers
 - Connect over SSH/WinRM
 - Deploy applications
 - Configure services
- Executes tasks
 - Make user
 - Start service
 - Download a file
 - ...
- Written in Python



YAML Ain't Markup Language

- Data serialization format
 - JSON
 - XML
- Common configuration file format

```
---  
Student-Id: 11223344  
First-Name: John  
Last-Name: Smith  
  
Phone-numbers:  
  - 281.555.7689  
  - 713.555.8967  
  - 832.555.9980  
  
Addresses:  
  - street: 123 Main St.  
    city: Houston  
    state: TX  
...
```

Ansible Inventory

- Set of **hosts** Ansible has access to
- INI format
- Groups
- Variables
- Host Variables
 - IP, Hostname
 - SSH User, Password/Keyfile

```
mail.example.com
```

```
[webservers]  
foo.example.com  
bar.example.com
```

```
[dbservers]  
one.example.com  
two.example.com  
three.example.com
```

Ansible

- **Tasks**
 - Execute single action
 - ex. Install package
- **Roles**
 - List of tasks
 - Reusable
 - Ex. Install and configure NGINX
 - Connects variables, files, templates
- **Playbooks**
 - Connect roles to hosts

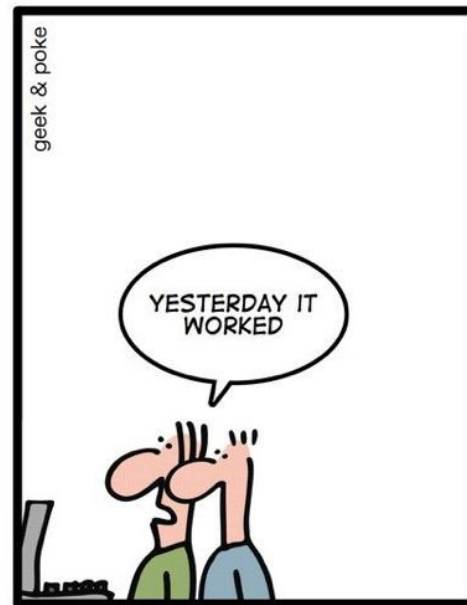
```
---  
- hosts: webservers  
  tasks:  
  - include_role:  
    name: foo_app_instance  
  vars:  
    dir: '/opt/a'  
    app_port: 5000
```

Testing

Test Automation

- Reliable
- Simultaneous
 - Test same change on different OS's, devices, browsers, etc in parallel
- Reduce time and effort
 - Available 24/7

WHEN YOU HEAR THIS:



YOU KNOW YOU'RE IN A SOFTWARE PROJECT

Testing Ansible Playbooks

“Can’t you just run the playbook to test it?” - Anonymous

Challenge: Ansible requires infrastructure

- Create base test server
 - Setup access
 - Configure the test Ansible inventory
- Run ansible-playbook commands
- Check for expected behavior
- Reset to snapshot

Molecule

Aids in the development and testing of Ansible roles.

- Created by Metacloud
 - OpenStack-based on-prem private cloud
 - Bought by Cisco
 - GitHub full of tools for Ansible and OpenStack

<https://github.com/metacloud/molecule>

Molecule - Init Role

```
pip install molecule docker-py
```

```
molecule init role -r myapp --verifier-name goss
```

```
Usage: molecule init role [OPTIONS]

Initialize a new role for use with Molecule.

Options:
  --dependency-name [galaxy]      Name of dependency to initialize. (galaxy)
  -d, --driver-name [azure|delegated|docker|ec2|gce|lxc|lxd|openstack|vagrant]
                                  Name of driver to initialize. (docker)
  --lint-name [yamllint]         Name of lint to initialize. (yamllint)
  --provisioner-name [ansible]   Name of provisioner to initialize. (ansible)
  -r, --role-name TEXT           Name of the role to create. [required]
  --verifier-name [goss|inspec|testinfra]
                                  Name of verifier to initialize. (testinfra)
  --help                          Show this message and exit.
```

Molecule - Init Role

- Creates initial Ansible files and Molecule configuration

*Can test an existing role with `init scenario`

```
httpd_webserver
├── defaults
│   └── main.yml
├── handlers
│   └── main.yml
├── meta
│   └── main.yml
├── molecule
│   └── default
│       ├── create.yml
│       ├── destroy.yml
│       ├── Dockerfile.j2
│       ├── INSTALL.rst
│       ├── molecule.yml
│       ├── playbook.yml
│       ├── prepare.yml
│       ├── tests
│       │   └── test_default.yml
│       └── verifier.yml
├── README.md
├── tasks
│   └── main.yml
├── vars
│   └── main.yml
└── .yamllint
```

Molecule

- Test
 - Style and syntax
 - That it actually works
 - Idempotence
- On Varying
 - Operating Systems
 - Distributions
 - Virtualization Providers
 - Testing Frameworks

```
test_sequence:  
- lint  
- destroy  
- dependency  
- syntax  
- create  
- prepare  
- converge  
- idempotence  
- side_effect  
- verify  
- destroy
```

Molecule - Lint

- Yamllint
 - All YAML files
- Flake8
 - Testinfra
 - PEP 8
- Ansible-lint
 - Ansible playbooks
- Strict by default
 - Warnings (trailing whitespace) will stop testing
 - Set rules to ignore
- Create the best playbooks possible

Molecule - Destroy

- Destroy any preexisting containers used by Molecule

Molecule - Dependency

- Pull in other roles
- Support for Ansible Galaxy, Gilt, or shell commands

Molecule - Syntax

- Have Ansible natively check for syntax errors

Molecule - Create

- Supports multiple providers (called **Drivers**)
 - Docker, Azure, EC2, GCE, Vagrant
- Actual creation done by the **Provisioner** (Ansible)
- Creates **Instances**

```
platforms:  
- name: redis_cnt7  
  image: centos:7  
- name: redis_ubnt  
  image: ubuntu:16.04
```

Molecule - Prepare

- Get the instance to a state you want before converging
- Base enterprise configuration

```
---  
- name: Converge  
  hosts: all  
  roles:  
    - role: redis
```

Molecule - Converge

- Run the role being tested on all instances
- Runs like a normal playbook

Molecule - Idempotence

- “An operation is idempotent if the result of performing it once is exactly the same as the result of performing it repeatedly without any intervening actions.” - Ansible Docs
- Check that there are no unexpected changes when running a playbook multiple times
- If no tasks are marked as changed, role is considered idempotent by Molecule

Molecule - Side-Effect

- “Post-Tasks” to test operability of installed services
- Test situations like high availability failover
- Disabled by default

Molecule - Verify

- Test the **actual** server state after configuration
- Supports multiple verifiers
 - Testinfra
 - Python
 - Goss
 - YAML
 - Inspec (Chef)

```
import os

import testinfra.utils.ansible_runner

testinfra_hosts = testinfra.utils.ansible_runner.AnsibleRunner(
    os.environ['MOLECULE_INVENTORY_FILE']).hosts

def test_hosts_file(host):
    f = host.file('/etc/hosts')

    assert f.exists
    assert f.user == 'root'
    assert f.group == 'root'
```

```
service:
  sshd:
    enabled: true
    running: true
user:
  sshd:
    exists: true
    uid: 74
    gid: 74
    groups:
      - sshd
    home: /var/empty/sshd
    shell: /sbin/nologin
group:
  sshd:
    exists: true
    gid: 74
process:
  sshd:
    running: true
```


Molecule - Destroy

- Destroy any containers used by Molecule
- Use `molecule test --destroy never` to keep instance up
 - Debugging

Demo

Resources

1. Ansible Hacknight Presented by Sean Newman
 - a. 10/29 7-11pm Seclab
2. Ansible Crash Course
 - a. Shoutout to Duc for starring this on GitHub
 - b. <https://github.com/blacksaildivision/ansible-tutorial>
3. Molecule GitHub Repo
 - a. <https://github.com/metacloud/molecule>
4. The Wizard: Ansible, Molecule and Test Driven Development
 - a. <https://blog.octo.com/en/the-wizard-ansible-molecule-and-test-driven-development/>
5. Digital Ocean Tutorial
 - a. <https://www.digitalocean.com/community/tutorials/how-to-test-ansible-roles-with-molecule-on-ubuntu-16-04>
6. Goss Manual
 - a. <https://github.com/aelsabbahy/goss/blob/master/docs/manual.md>