

Testing Ansible Roles w/Molecule

By Eric Anderson

Contents

- Introduction
 - About me
 - Why test Ansible roles?
- How did we test prior to Molecule?
- What is Molecule?
- Getting Started
- Using Molecule
- How do I migrate my existing roles?
- Tribulations

About Me

- Name: Eric Anderson
- Nickname: ericsysmin
- Position: Senior MTS DevOps/SRE
- Company: VMware
- Contact:
 - eric.sysmin@gmail.com
 - aeric@vmware.com
 - ericsysmin.com
 - Twitter: @ericsysmin
 - GitHub: [ericsysmin](https://github.com/ericsysmin)
- Ansible User of 6 years, Puppet prior



Why test our Ansible roles?

- Ensure that our roles are properly tested and have some kind of quality control
- Role could end up being used by thousands of people and companies around the globe
- Prevent breaking your environments
- Easy to test common cases
- Don't want hate mail because someone using your role ended up bringing down their environment
- It feels really good to have perfect scores on Ansible Galaxy
- Tests can also run on others Pull Requests

Testing prior to Molecule

- It was pretty manual
- Everyone used their own methods
- Some tests only worked on Ubuntu
- Some people wrote their own docker based testing on Travis-CI
- ex.
 - <https://www.jeffgeerling.com/blog/testing-ansible-roles-travis-ci-github>
 - <http://bertvv.github.io/notes-to-self/2015/12/11/testing-ansible-roles-with-travis-ci-part-1-centos/>
 - <http://bertvv.github.io/notes-to-self/2015/12/13/testing-ansible-roles-with-travis-ci-part-2-multi-platform-tests/>
 - <https://github.com/ericssmin/ansible-role-chrony/tree/be214cad05514a7a26cf17a5e9270e92b839866b>



ericsysmin

chrony

This role enables users to install and configure chrony on their hosts.

8276 Downloads 2 Watchers 8 Stars

8 Forks

Follow Collection

Issue Tracker

GitHub Repo

build passing

Details

Read Me

Info

Minimum Ansible Version 2.4

Installation `$ ansible-galaxy install ericsysmin.chrony`

Last Commit 14 days ago

Last Import 14 days ago

Tags **chrony** **ntp** **system** **time**

Content Score

Quality Score  5 / 5

Last scored 14 days ago. [Show Details](#)

Community Score **No Surveys** 0 / 5

Based on 0 surveys. [Show Details](#)

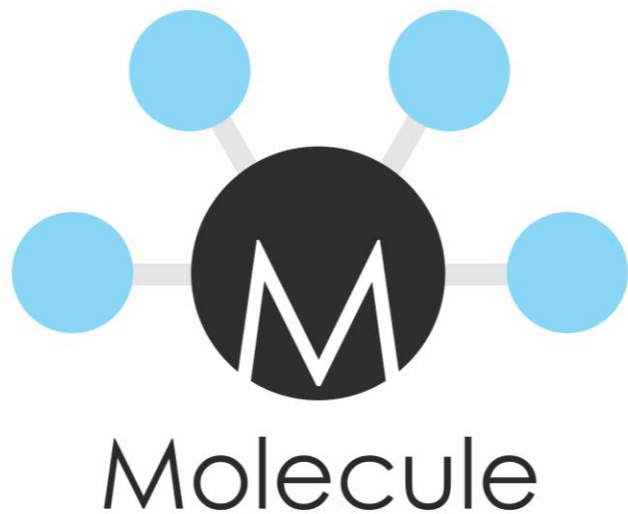
Tell us about this collection

- Quality of docs? - +
- Ease of use? - +
- Does what it promises? Y N
- Works without change? Y N
- Used in production? Y N

OS Platforms

Debian	jessie
Debian	stretch
EL	7

What is Molecule?



- Ansible based tool to help develop roles
- Includes YAML Lint, Ansible Lint and can use other lint tools
- Openly configurable
 - can modify the steps
 - can add or remove lint, providers, etc.
- Doesn't rely on being tested locally on your machine
- Runs by default with Docker provider

Getting Started

Installation

<https://molecule.readthedocs.io/en/latest/installation.html>

Getting Started

- creating a new role with Molecule:
 - `molecule init role --role-name role_name`
- deploying Molecule in an existing role:
 - `cd role_name; molecule init scenario --role-name role_name`
- testing the functionality of a role:
 - `molecule test`
- Default scenario is created in ``molecule/default`` you can create additional scenario's by specifying a ``--scenario-name`` value.
- etc. <https://molecule.readthedocs.io/en/latest/usage.html>

Molecule Scenario Configuration

```
1 ---
2 dependency:
3   name: galaxy
4 driver:
5   name: docker
6 lint:
7   name: yamllint
8   options:
9     config-file: molecule/default/.yamllint
10 platforms:
11   - name: instance
12     image: ${MOLECULE_DISTRO:-ubuntu:xenial}
13     privileged: true
14     service_manager: ${MOLECULE_SERVICE_MANAGER:-systemd}
15     volumes:
16       - /sys/fs/cgroup:/sys/fs/cgroup:ro
17     command: ${MOLECULE_COMMAND:-""}
18 provisioner:
19   name: ansible
20   lint:
21     name: ansible-lint
22     options:
23       c: molecule/default/.ansible-lint
24   playbooks:
25     converge: ${MOLECULE_PLAYBOOK:-playbook.yml}
26 scenario:
27   name: default
28 verifier:
29   name: testinfra
30   lint:
31     name: flake8
```

Dependency: Enables ability to download role dependencies during testing with a requirements role file

Driver: chooses how the instances will be created, docker, gce, ec2, etc.

Lint: you can customize your lint configuration in place, or by using a config file, this can be used to skip specific lint checks

Platforms: allows you to provide an array of docker images or systems to be created and tested against. In my example I allow use of an Environment Var to set the image, service_manager, and init command (MOLECULE_COMMAND) These I've added to allow instances running as services to be tested.

Provisioner: leave this as Ansible, it's the only supported one, you can specify the playbook for the scenario here, or allow it to be an environment variable as well which you can use in a matrix based configuration

Scenario: the name of the scenario, and you can provide overrides if desired to which steps execute.

Verifier: this allows configuration of a python based verifier to check file existence, service state, and more. It will use the lint type provided to verify the python script.

Molecule Docker Configuration

- When using the Docker provisioner we will want to have something like the following file: <https://github.com/ericysmin/ansible-role-ntp/blob/master/molecule/default/Dockerfile.j2>
- The docker file is in Jinja format and is executed by Ansible code in Molecule
- The file in the link is an example created to properly allow services on docker containers.

Travis-CI Configuration

```
1 ---
2 language: python
3 services:
4   - docker
5
6 before_install:
7   - sudo apt-get -qq update
8
9 env:
10  global:
11    - ROLE_NAME: ntp
12  matrix:
13    - MOLECULE_DISTRO: centos:7
14      MOLECULE_SERVICE_MANAGER: systemd
15      MOLECULE_COMMAND: /usr/sbin/init
16    - MOLECULE_DISTRO: fedora:27
17      MOLECULE_SERVICE_MANAGER: systemd
18      MOLECULE_COMMAND: /usr/sbin/init
19    - MOLECULE_DISTRO: fedora:28
20      MOLECULE_SERVICE_MANAGER: systemd
21      MOLECULE_COMMAND: /usr/sbin/init
22    - MOLECULE_DISTRO: ubuntu-upstart:trusty
23      MOLECULE_COMMAND: /sbin/init
24    - MOLECULE_DISTRO: ubuntu:xenial
25      MOLECULE_COMMAND: /sbin/init
26    - MOLECULE_DISTRO: ubuntu:bionic
27      MOLECULE_SERVICE_MANAGER: systemd
28      MOLECULE_COMMAND: /lib/systemd/systemd
29    - MOLECULE_DISTRO: debian:jessie
30      MOLECULE_COMMAND: /sbin/init
31    - MOLECULE_DISTRO: debian:stretch
32      MOLECULE_SERVICE_MANAGER: systemd
33      MOLECULE_COMMAND: /lib/systemd/systemd
34
35 install:
36   - pip install testinfra molecule docker
37
38 before_script:
39   - cd ../
40   - mv ansible-role- $\$ROLE\_NAME$  ericssysmin. $\$ROLE\_NAME$ 
41   - cd ericssysmin. $\$ROLE\_NAME$ 
42
43 script:
44   - molecule test
45
46 notifications:
47   webhooks: https://galaxy.ansible.com/api/v1/notifications/
```

- **services:** list of services required on the Travis-CI host
- **before_install:** Scripts to run prior to any tests on the Travis-CI host
- **env:** Define the environment variables to be provided to the execution
 - **global:** any var here will be used for all tests
 - **matrix:** The matrix defines the variables that will be used in each environment tested. This allows testing of multiple operating systems and configurations at the same time
 - In my case I provided distribution, service manager, and the proper init command to allow services to be used on docker
- **install:** We need to install some scripts on the host, so we use the install step to do so for molecule to work correctly
- **before_script:** these are commands that will need to run to properly name the folder for proper molecule execution based on an accurate role name
- **script:** this is the command that Travis-CI will run and count as the build, this is where we call Molecule
- **notifications:** this step will notify Ansible Galaxy of a success or failure of the tests

Using Molecule

<https://molecule.readthedocs.io/en/latest/usage.html>

Molecule Working Group

- Meetings
 - Weekly on Wednesday at [19:00 UTC \(ical\)](#)
 - [Meeting agenda](#)
 - [Meeting notes](#)
- Join the discussion
 - `#ansible-molecule` on Freenode
 - [Molecule mailing list](#)
- Agenda
 - Currently using the Molecule agenda [Molecule Agenda](#) for planning.



Questions?