

Ansible London

meetup

Ansible Test Driven Development Using Molecule

Pavlos Kleanthous

Whoami

DevOps Engineer

 pavlos@cuprum.io

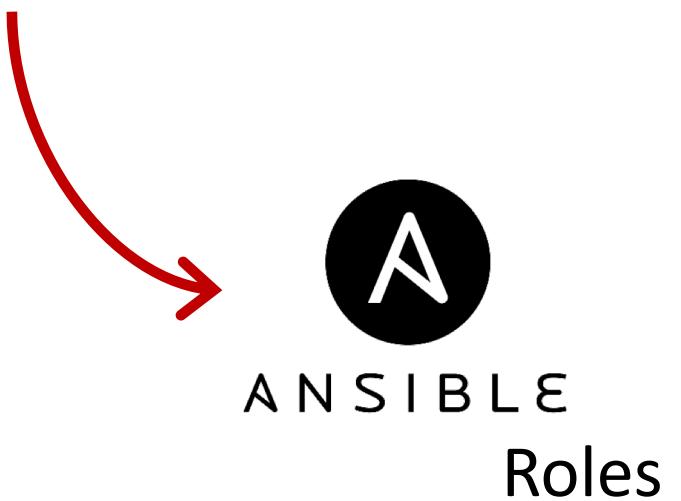
 @pkleanthous

 in/pkleanthous



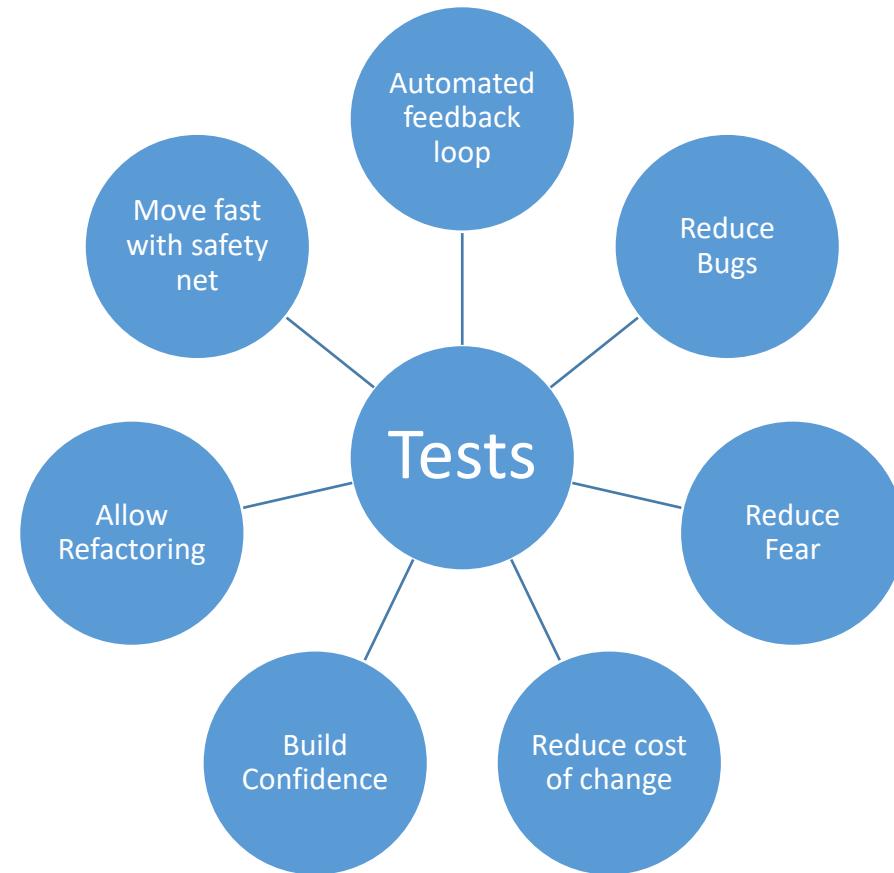
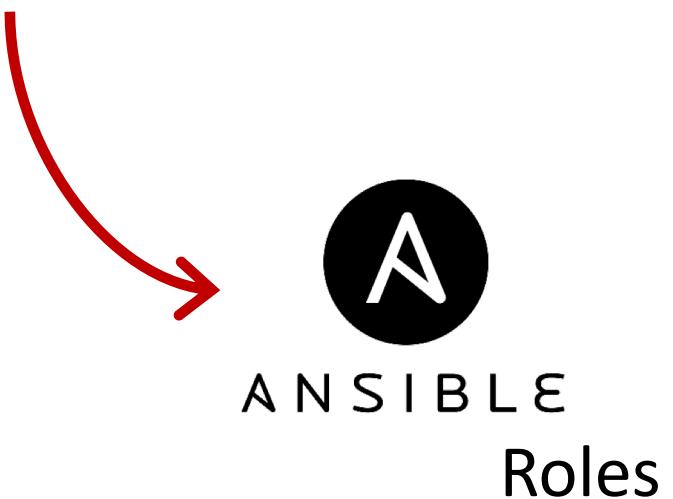
Challenge

Writing Reliable, Robust and
Maintainable infrastructure code

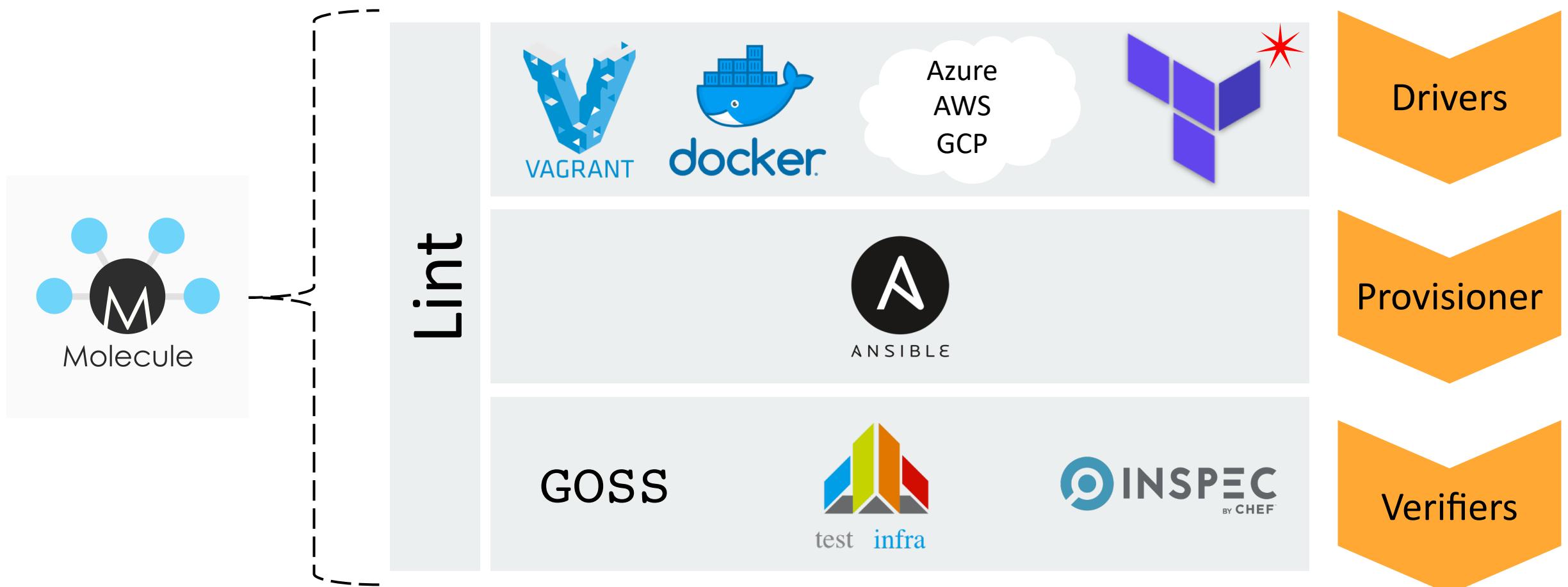


Challenge

Writing Reliable, Robust and
Maintainable infrastructure code



Molecule



```
molecule --help
Usage: molecule [OPTIONS] COMMAND [ARGS]...

   _____
  |  _  | |  _ | |  _ | |  _ | | | | | | | | |
  | | | | . | | -_ | | -_ | | -_ |
  | | | | | | | | | | | | | | | |

Molecule aids in the development and testing of Ansible roles.

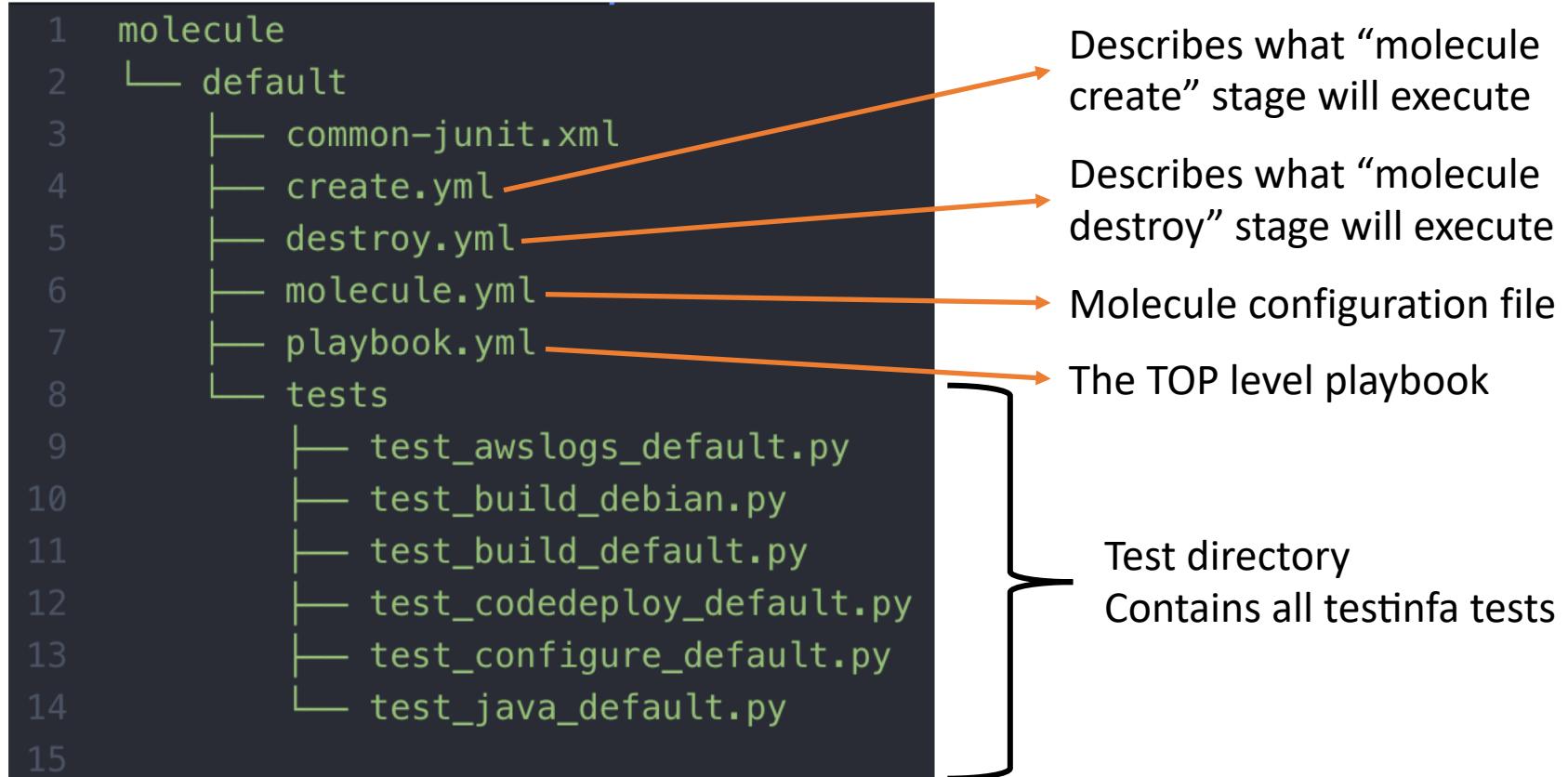
Enable autocomplete issue:

eval "$(_MOLECULE_COMPLETE=source molecule)"

Options:
--debug / --no-debug      Enable or disable debug mode. Default is disabled.
-c, --base-config TEXT   Path to a base config. If provided Molecule will
                         load this config first, and deep merge each
                         scenario's molecule.yml on top.
                         (/Users/pkleanthous/.config/molecule/config.yml)
-e, --env-file TEXT       The file to read variables from when rendering
                         molecule.yml. (.env.yml)
--version                 Show the version and exit.
--help                    Show this message and exit.

Commands:
check        Use the provisioner to perform a Dry-Run...
converge     Use the provisioner to configure instances...
create       Use the provisioner to start the instances.
dependency   Manage the role's dependencies.
destroy      Use the provisioner to destroy the instances.
idempotence Use the provisioner to configure the...
init         Initialize a new role or scenario.
lint         Lint the role.
list         Lists status of instances.
login        Log in to one instance.
matrix       List matrix of steps used to test instances.
prepare      Use the provisioner to prepare the instances...
side-effect  Use the provisioner to perform side-effects...
syntax      Use the provisioner to syntax check the role.
test         Test (lint, destroy, dependency, syntax, ...
verify      Run automated tests against instances.
```

Basic Structure of Molecule Scenario



molecule.yml

```
1  ---
2  dependency:
3    name: galaxy
4  driver:
5    - name: terraform
6      region: eu-west-2
7      profile: id1
8      vpc_id: vpc-01f0410397da7b7f1
9      subnet_id: subnet-0d5c142c85da7cbc9
10 lint:
11   name: yamllint
12 platforms:
13   - name: role-common-scenario-default
14     ami: ami-070308c631ec256eb
15     instance_type: t2.medium
16     ssh_username: ubuntu
17 provisioner:
18   name: ansible
19   options:
20     vv: true
21     become: true
22   config_options:
23     defaults:
24       forks: 10
25       stdout_callback: yaml
26   lint:
27     name: ansible-lint
28     options:
29       force-color: true
30 scenario:
31   name: default
32 verifier:
33   name: testinfra
34   options:
35     sudo: true
36     junit-xml: common-junit.xml
37     verbose: true
38   lint:
39     name: flake8
40     options:
41       benchmark: true
42
```

Use Ansible galaxy to fetch additional roles

molecule.yml

```
1  ---
2  dependency:
3    name: galaxy
4  driver:
5    - name: terraform
6      region: eu-west-2
7      profile: id1
8      vpc_id: vpc-01f0410397da7b7f1
9      subnet_id: subnet-0d5c142c85da7cbc9
10 lint:
11   name: yamllint
12 platforms:
13   - name: role-common-scenario-default
14     ami: ami-070308c631ec256eb
15     instance_type: t2.medium
16     ssh_username: ubuntu
17 provisioner:
18   name: ansible
19   options:
20     vv: true
21     become: true
22   config_options:
23     defaults:
24       forks: 10
25       stdout_callback: yaml
26   lint:
27     name: ansible-lint
28     options:
29       force-color: true
30 scenario:
31   name: default
32 verifier:
33   name: testinfra
34   options:
35     sudo: true
36     junit-xml: common-junit.xml
37     verbose: true
38   lint:
39     name: flake8
40     options:
41       benchmark: true
42
```

Define where do you want to run your code

molecule.yml

```
1  ---
2  dependency:
3      name: galaxy
4  driver:
5      - name: terraform
6          region: eu-west-2
7          profile: id1
8          vpc_id: vpc-01f0410397da7b7f1
9          subnet_id: subnet-0d5c142c85da7cbc9
10 lint:
11     name: yamllint
12 platforms:
13     - name: role-common-scenario-default
14         ami: ami-070308c631ec256eb
15         instance_type: t2.medium
16         ssh_username: ubuntu
17 provisioner:
18     name: ansible
19     options:
20         vv: true
21         become: true
22     config_options:
23         defaults:
24             forks: 10
25             stdout_callback: yaml
26     lint:
27         name: ansible-lint
28         options:
29             force-color: true
30 scenario:
31     name: default
32 verifier:
33     name: testinfra
34     options:
35         sudo: true
36         junit-xml: common-junit.xml
37         verbose: true
38     lint:
39         name: flake8
40         options:
41             benchmark: true
42
```

Define linter for the role

molecule.yml

```
1  ---
2  dependency:
3      name: galaxy
4  driver:
5      - name: terraform
6          region: eu-west-2
7          profile: id1
8          vpc_id: vpc-01f0410397da7b7f1
9          subnet_id: subnet-0d5c142c85da7cbc9
10 lint:
11     name: yamllint
12 platforms:
13     - name: role-common-scenario-default
14         ami: ami-070308c631ec256eb
15         instance_type: t2.medium
16         ssh_username: ubuntu
17 provisioner:
18     name: ansible
19     options:
20         vv: true
21         become: true
22     config_options:
23         defaults:
24             forks: 10
25             stdout_callback: yaml
26     lint:
27         name: ansible-lint
28         options:
29             force-color: true
30 scenario:
31     name: default
32 verifier:
33     name: testinfra
34     options:
35         sudo: true
36         junit-xml: common-junit.xml
37         verbose: true
38     lint:
39         name: flake8
40         options:
41             benchmark: true
42
```

Define the platform (the OS)

molecule.yml

```
1  ---
2  dependency:
3      name: galaxy
4  driver:
5      - name: terraform
6          region: eu-west-2
7          profile: id1
8          vpc_id: vpc-01f0410397da7b7f1
9          subnet_id: subnet-0d5c142c85da7cbc9
10 lint:
11     name: yamllint
12 platforms:
13     - name: role-common-scenario-default
14         ami: ami-070308c631ec256eb
15         instance_type: t2.medium
16         ssh_username: ubuntu
17 provisioner:
18     name: ansible
19     options:
20         vv: true
21         become: true
22     config_options:
23         defaults:
24             forks: 10
25             stdout_callback: yaml
26     lint:
27         name: ansible-lint
28         options:
29             force-color: true
30 scenario:
31     name: default
32 verifier:
33     name: testinfra
34     options:
35         sudo: true
36         junit-xml: common-junit.xml
37         verbose: true
38     lint:
39         name: flake8
40         options:
41             benchmark: true
42
```

Define the provisioner.
Ansible it's the only supported provisioner

molecule.yml

```
1  ---
2  dependency:
3      name: galaxy
4  driver:
5      - name: terraform
6          region: eu-west-2
7          profile: id1
8          vpc_id: vpc-01f0410397da7b7f1
9          subnet_id: subnet-0d5c142c85da7cbc9
10 lint:
11     name: yamllint
12 platforms:
13     - name: role-common-scenario-default
14         ami: ami-070308c631ec256eb
15         instance_type: t2.medium
16         ssh_username: ubuntu
17 provisioner:
18     name: ansible
19     options:
20         vv: true
21         become: true
22     config_options:
23         defaults:
24             forks: 10
25             stdout_callback: yaml
26     lint:
27         name: ansible-lint
28         options:
29             force-color: true
30 scenario:
31     name: default
32 verifier:
33     name: testinfra
34     options:
35         sudo: true
36         junit-xml: common-junit.xml
37         verbose: true
38     lint:
39         name: flake8
40         options:
41             benchmark: true
42
```

Define the Scenario.
A scenario allows Molecule test
a role in a particular way.

molecule.yml

```
1  ---
2  dependency:
3    name: galaxy
4  driver:
5    - name: terraform
6      region: eu-west-2
7      profile: id1
8      vpc_id: vpc-01f0410397da7b7f1
9      subnet_id: subnet-0d5c142c85da7cbc9
10 lint:
11   name: yamllint
12 platforms:
13   - name: role-common-scenario-default
14     ami: ami-070308c631ec256eb
15     instance_type: t2.medium
16     ssh_username: ubuntu
17 provisioner:
18   name: ansible
19   options:
20     vv: true
21     become: true
22   config_options:
23     defaults:
24       forks: 10
25       stdout_callback: yaml
26   lint:
27     name: ansible-lint
28     options:
29       force-color: true
30 scenario:
31   name: default
32 verifier:
33   name: testinfra
34   options:
35     sudo: true
36     junit-xml: common-junit.xml
37     verbose: true
38   lint:
39     name: flake8
40     options:
41       benchmark: true
42
```

Define the Verifier. Molecule handles role testing by invoking configurable verifiers. Testinfra is the default test runner.

molecule.yml

```
1  ---
2  dependency:
3    name: galaxy
4  driver:
5    - name: terraform
6      region: eu-west-2
7      profile: id1
8      vpc_id: vpc-01f0410397da7b7f1
9      subnet_id: subnet-0d5c142c85da7cbc9
10 lint:
11   name: yamllint
12 platforms:
13   - name: role-common-scenario-default
14     ami: ami-070308c631ec256eb
15     instance_type: t2.medium
16     ssh_username: ubuntu
17 provisioner:
18   name: ansible
19   options:
20     vv: true
21     become: true
22   config_options:
23     defaults:
24       forks: 10
25       stdout_callback: yaml
26   lint:
27     name: ansible-lint
28     options:
29       force-color: true
30 scenario:
31   name: default
32 verifier:
33   name: testinfra
34   options:
35     sudo: true
36     junit-xml: common-junit.xml
37     verbose: true
38   lint:
39     name: flake8
40     options:
41       benchmark: true
42
```



Summary

- Molecule framework facilitates TDD for Ansible roles
- Supports configuring a variety of compute instances (drivers)
- Integrates with Testinfra and Inspec
- Enables fast, reliable and isolated role development