LESSON 11



YAML

Ansible CI/CD

Learn it all @MyWebUniversity



A FREE MYWEBUNIVERSITY COURSE

ANSIBLE & DEVOPS

LESSON 11

Ansible in a DevOps Pipeline

Git workflow, ansible-lint, Molecule testing, CI/CD with GitHub Actions or Jenkins, environ

Git Workflow



- Branch — feature/nginx-tls
- Pull request — review + CI checks
- Merge — main

01

ansible-lint

terminal

```
$ ansible-lint site.yml
name[missing]: All tasks should be named.
roles/nginx/tasks/main.yml:12 Task/Handler: shell grep ...
fqcn[action-core]: Use FQCN for builtin module actions (copy).
roles/nginx/tasks/main.yml:20
Failed: 2 failure(s), 0 warning(s)
```

- ansible-lint checks playbooks against best practices: missing names, short module names, …
- yamllint checks the YAML style. Run both on every commit.

02

Testing Roles with Molecule

terminal

```
$ molecule test
INFO  create      ✓  rocky9, ubuntu24
INFO  converge    ✓  changed=6
INFO  idempotence ✓  changed=0
INFO  verify      ✓  nginx answering on :80
INFO  destroy     ✓
```

- Molecule tests a role in throwaway containers or virtual machines. It creates the instance, …
- A role that passes Molecule on Rocky and Ubuntu is ready to share.

03

A CI/CD Pipeline

.github/workflows/deploy.yml

```yaml
on: { push: { branches: [main] } }
jobs:
  deploy:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
      - run: pip install ansible ansible-lint
      - run: ansible-lint
      - run: echo "$VAULT_PASS" > .vault_pass
        env: { VAULT_PASS: "${{ secrets.VAULT_PASS }}" }
      - run: ansible-playbook -i inventories/staging site.yml --vault-password-file .vault_pass
```

- A pipeline runs automatically on every push: lint, test, then deploy to staging, and to …
- Store the SSH key and vault password as pipeline secrets, never in the repository. Jenkins, …

04

Environments

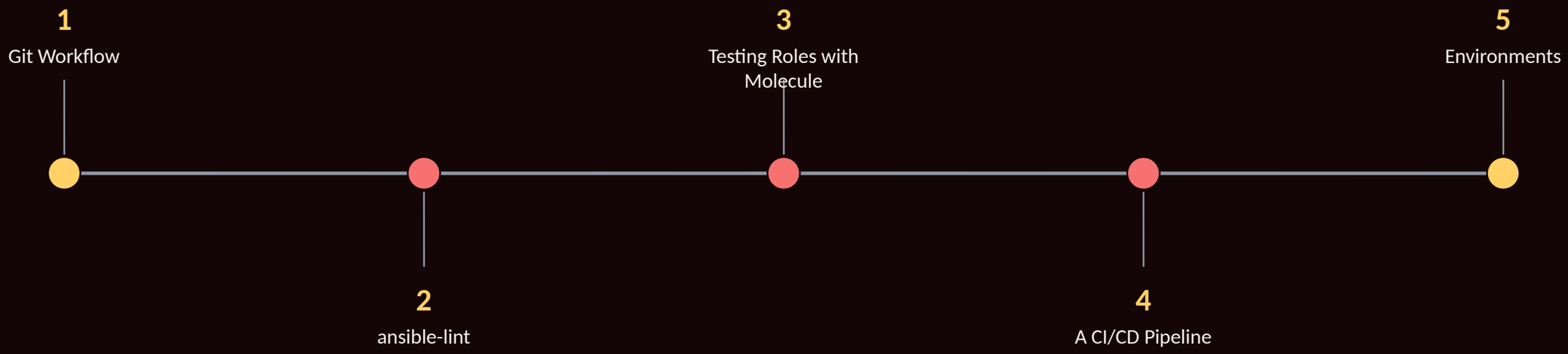
terminal

```
inventories/
├── dev/
│   ├── hosts.ini
│   └── group_vars/all.yml      # app_replicas: 1
├── staging/
│   └── hosts.ini
└── production/
    ├── hosts.ini
    └── group_vars/all.yml      # app_replicas: 4
```

- Separate inventories for development, staging and production let the same playbooks run …
- Changes always flow in one direction: dev, then staging, then production. Never fix …

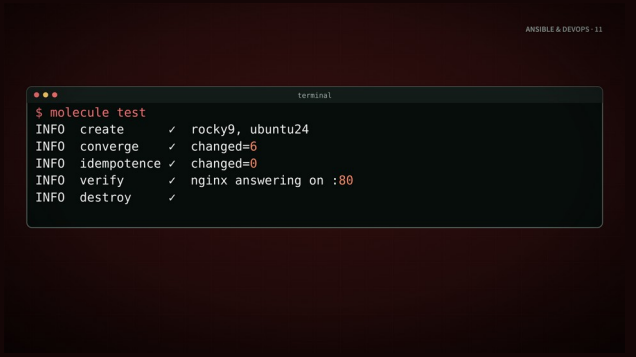
05

Lesson Map


1
Git Workflow
2
ansible-lint
3
Testing Roles with Molecule
4
A CI/CD Pipeline
5
Environments

Sources and Further Reading





Thank you for watching

MyWebUniversity.com

Free technical education since 2000 — English and Dari

@MyWebUniversity · @MyWebUniversity-Dari