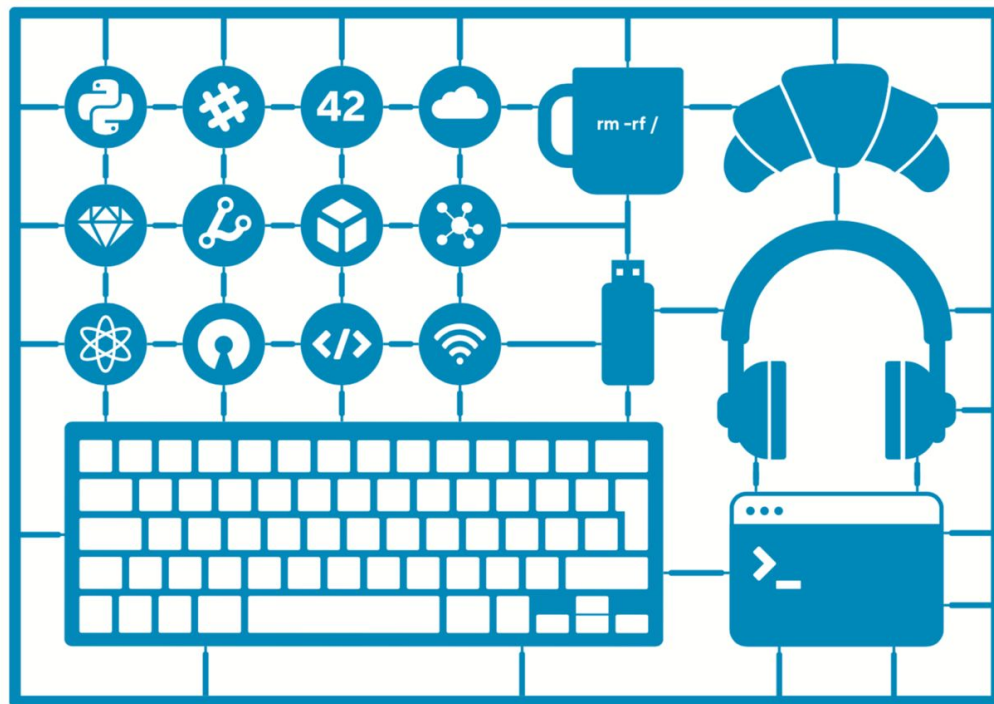


Molecule, ou comment tester ses rôles Ansible



Cyril Grosjean
@JqckB

19 Novembre 2019



Wifirst's Dev kit

ON DIT LE Wi-Fi BORDEL

- FAI depuis 2002 spécialisé dans le Wi-Fi
- Clients :
 - Étudiants (CROUS, Residhome, Nexity, ...)
 - Hôtellerie (Accor, Appart'City, ...)
 - B2B (La Poste, Blue Solution, Havas, Norauto)
 - Armées : Internet Loisir en France et en Opération Extérieure (OPEX)
- 150k bornes Wi-Fi, 20k switches, 10k serveurs
- 5ème opérateur en terme de clients connectés (400k+ clients connectés), 125 Gbit/s de trafic
- Infrastructure SI de 400 VMs, 3 OpenStacks, 3 clusters Ceph (2 Bloc et 1 Objet)



On recrute

<https://www.welcometothejungle.com/fr/companies/wifirst>

- Outil de gestion de configuration
- Agentless
- Stateless
- Idempotent
- Open Source / Red Hat (support)
- Modules en tout genre (Système, Réseaux, ...)
- Rôles disponibles communautairement (Galaxy)

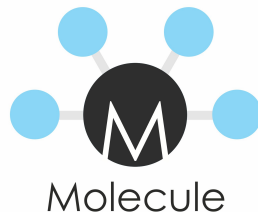
→ Utilisé en masse chez Wifirst (Déploiement de configuration clientes, Infrastructure, ...)



ANSIBLE

MOLECULE : KESAKO

- Outil de test de rôles/playbooks Ansible
- Soutenu par Ansible/Red Hat depuis Octobre 2018
- Permet d'exécuter les rôles sur un ensemble d'OS via des drivers :
 - Docker (default)
 - OpenStack
 - Vagrant
 - ...
- Exécute une série de tests :
 - Linter (YAML/Ansible/Python)
 - Déploiement du rôle
 - Idempotence du rôle (Relance le déploiement du rôle, vérifie les "changed")
 - Vérification de la configuration du rôle (TestInfra, Goss, Ansible, Inspec)



Documentation Molecule : <https://molecule.readthedocs.io/en/stable/>

Documentation TestInfra : <https://testinfra.readthedocs.io/en/latest/>

DEMO GOD, HELP ME !



Github : https://github.com/wifirst-lab/presentations/tree/master/201911_Molecule_Sysadmin_Days

MOLECULE : MOT CLÉS UTILES

En macro :

- Scénarios : Ensemble de configuration Molécule (déploiement, configuration, variables, tests, ...)
- Driver : interface permettant de sélectionner la target de déploiement des instances
OpenStack/Docker/Vagrant/AWS/GCE/Azure/Baremetal (14 drivers)
- Verifier : outil de testing (TestInfra/Goss/Inspec/Ansible)
- Converge : premier déploiement du rôle

Les fichiers de configurations :

- Obligatoire :
 - molecule.yml : rassemble les configurations du scénario
 - playbook.yml : playbook Ansible de l'étape de convergence
- Optionnel :
 - prepare.yml : playbook Ansible s'exécutant avant l'étape de convergence
 - create.yml : playbook Ansible pour la création de l'instance (spécifique à chaque driver)
 - destroy.yml : playbook Ansible pour la suppression de l'instance (spécifique à chaque driver)

MOLECULE : EXEMPLES

- Un scénario est l'équivalent d'une série de tests, exécuté sur un ensemble d'OS
 - Exemple :
 - Scénario 1 : Configurer un **Master/Slave** PostgreSQL sur CentOS 7/CentOS 8/Debian 9/Debian 10 via Docker
 - Scénario 2 : Configurer un **Standalone** PostgreSQL sur CentOS 7/CentOS 8/Debian 9/Debian 10 via Docker
 - Scénario 3 : Configurer un Standalone PostgreSQL sur CentOS 7 **via OpenStack**
- Utile pour faire de la CI sur des rôles (on utilise Gitlab-CI en interne)
 - Montée de version, changement de package, ...
 - Bloquer la nouvelle version d'un rôle non fonctionnel
- Utilisé par Tripleo (projet de déploiement OpenStack) pour valider les rôles Ansible de déploiements des nombreux services OpenStack (via Zuul)



Merci

www.wifirst.com

techblog.wifirst.net

www.linkedin.com/company/wifirst

www.twitter.com/wifirst