

Gestion de configurations : Ansible

Sommaire

Présentation	2
Mise en œuvre	2
Schéma du réseau	2
Configuration de la machine Ansible	2
Test en mode Ad-hoc	3
Création du premier playbook	3
Création du second playbook.....	4
Création du troisième playbook.....	5
Création du dernier playbook	5
Conclusion	5

Présentation

Ansible est un outil open-source de gestion de configuration écrit en python. La configuration se fait via des fichiers "Playbooks". Ansible combine le déploiement de logiciels multi-nœuds, l'exécution des tâches ad-hoc, et la gestion de configuration. Cette plate-forme a été créée par Michael DeHaan le 23 Avril 2012.

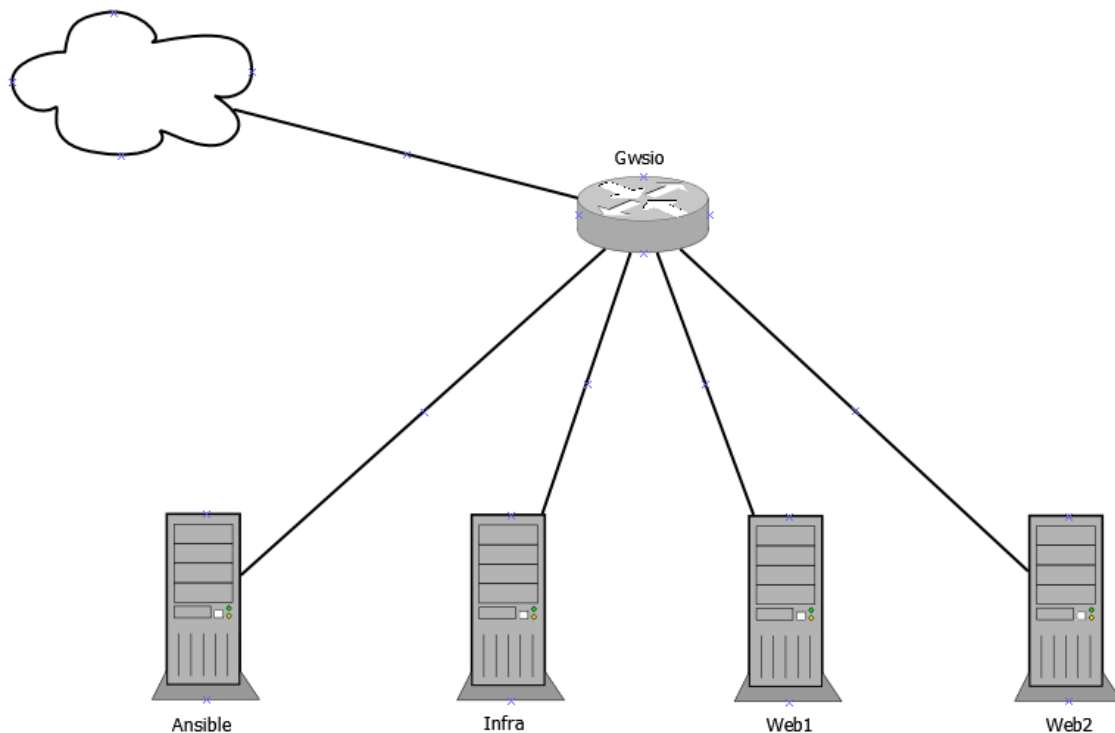
Un playbook est un fichier texte au format yaml qui va contenir la liste des différentes opérations à effectuer.

Mise en œuvre

Le but du Tp était de découvrir le fonctionnement et les avantages de la gestion de configuration à l'aide de l'outil Ansible.

Nous devons donc mettre en place 4 machines virtuelles : un serveur ansible, deux serveurs web et un serveur infra.

Schéma du réseau



Configuration de la machine Ansible

Avant de commencer il faut installer les paquets suivant : **python-dev python-yaml python-paramiko python-jinja2 git make** à l'aide de la commande **aptitude install**.

Nous allons maintenant récupérer ansible via le pip de python, il suffit de faire : **pip --proxy=http://gw :8080 install ansible**

Ansible fonctionnant exclusivement par ssh nous allons exporter la clé publique de notre serveur ansible aux autres machines. Nous allons dans un premier temps générer une clé via **ssh-keygen**. Maintenant nous allons éditer le fichier **~/.ssh/config** afin d'éviter de saisir l'adresse IP à chaque fois que nous nous connecterons en ssh. Par exemple pour la machine infra nous spécifierons :

```
host infra
  user root
  hostname 192.168.0.31
```

Nous pouvons maintenant distribuer notre clé publique aux différentes machines : **ssh-copy-id infra**
ssh-copy-id web1 ssh-copy-id web2.

Nous allons créer un répertoire ansible qui va accueillir nos fichiers. Pour se faire **mkdir /etc/ansible**
Déplacez-vous dans le répertoire et créez un fichier host qui va déterminer les groupes d'hôtes.

```
[adm]
infra

[web]
web1
web2
```

Test en mode Ad-hoc

Nous allons vérifier si notre serveur ping bien les autres machines. Pour se faire, nous allons entrer
ansible all -m ping

Création du premier playbook

Nous allons maintenant passer à la création du premier playbook qui servira à installer les paquets
apache2 php5 php5-gd sur le groupe de machine web et à mettre un index dans le répertoire
/var/www/.

Avant cela, il faut créer un dossier files dans **/etc/ansible** avant de mettre tous les fichiers de
configurations que nous allons créer.

Le premier fichier de configuration sera celui destiné au proxy : **apt.conf**

```
Acquire::http::Proxy "http://10.121.32.69:8080";
```

Le second fichier à créer sera l'index : **index.html**

```
<html>
  <head>
    <title>Sample "Hello, World" Application</title>
  </head>
</html>
```

Nous pouvons maintenant passer à la configuration du playbook intitulé **web.yml**

```
---
- hosts: web
  tasks:
    - name: Export proxy
      copy: src=/etc/ansible/files/apt.conf dest=/etc/apt/apt.conf mode=0644
    - name: Installer Apache
      apt: pkg=apache2 state=installed update_cache=true
    - name: Installer Php5
      apt: pkg=php5 state=installed
    - name: Installer Php5-gd
      apt: pkg=php5-gd state=installed
    - name: Envoi d'index
      copy: src=/etc/ansible/files/index.html dest=/var/www mode=0644
    - name: Démarrer Apache
      service: name=apache2 state=started
```

Nous pouvons maintenant tester le playbook à l'aide de la commande **ansible-playbook web.yml**

Création du second playbook

Nous voulons maintenant créer un serveur de gestion des logs (machine infra).

```
---
- hosts: infra
  tasks:
    - name: Export proxy
      copy: src=/etc/ansible/files/apt.conf dest=/etc/apt/apt.conf mode=0644

    - name: modification du fichier rsyslog.conf
      lineinfile: state=present dest=/etc/rsyslog.conf regexp='^#\$ModLoad imudp'
' line='$ModLoad imudp' backrefs=yes
    - name : modification du port udp 514
      lineinfile: state=present dest=/etc/rsyslog.conf regexp='^#\$UDPServerRun
514' line='$UDPServerRun 514' backrefs=yes
      notify:
        - restart rsyslog

  handlers:
    - name: redémarrer le service
      service: name=rsyslog stated=restarted
~
~
```

Nous allons donc spécifier aux machines du groupe web d'envoyer leurs logs sur la machine infra à l'aide d'un nouveau playbook.

```
---
- hosts: web
  tasks:
    - name: Export proxy
      copy: src=/etc/ansible/files/apt.conf dest=/etc/apt/apt.conf mode=0644

    - name: envoi des log de web1 et web2
      lineinfile: state=present dest=/etc/rsyslog.conf regexp='' line='*.* @192.
168.0.51' backrefs=yes
      notify:
        - restart rsyslog

  handlers:
    - name: restart rsyslog
      service: name=rsyslog stated=restarted
~
~
```

Création du troisième playbook

Ce playbook a pour but d'installer un serveur squid sur la machine infra.

```
---
- hosts: adm
  vars:
    proxy_port: 8080
    proxy_mem: 32
    localnet: 192.168.0.0/24

  tasks:
    - name: Export proxy
      copy: src=/etc/ansible/files/apt.conf dest=/etc/apt/apt.conf mode=0644

    - name: Installer squid3
      apt: pkg=squid3 state=installed update_cache=true

    - name: GENERATION DU SQUID CONF
      template: src=/etc/ansible/files/squid.conf.j2 dest=/etc/squid3/squid.conf
      notify:
        - restart squid3

  handlers:
    - name: restart squid3
      service: name=squid3 stated=restarted
```

Création du dernier playbook

Ce dernier playbook sert à installer un dokuwiki à partir d'une archive tar.gz

```
---
- hosts: web1
  tasks:
    - name: Export proxy
      shell: export http_proxy=http://10.121.32.69:8080

    - name: wget
      shell: export http_proxy=http://10.121.32.69:8080 && wget -O /var/tmp/dokuwiki-stable.tgz http://download.dokuwiki.org/src/dokuwiki/dokuwiki-stable.tgz

    - name: Unpack tarball
      command: tar -xvf dokuwiki-stable.tgz chdir=/var/tmp

    - name: Move dokuwiki
      shell: mv /var/tmp/dokuwiki-2014-09-29b /var/www/dokuwiki

    - name: permissions
      shell: chown -R www-data /var/www/dokuwiki/*

    - notify:
        - restart apache2

  handlers:
    - name: restart apache2
      service: name=apache2 stated=restarted
```

Conclusion

Pour conclure, nous pouvons dire qu'Ansible permet à l'administrateur système de gagner beaucoup de temps lors de l'installation d'une nouvelle machine.

En effet, les scripts sont valables sur des groupes de machines ou sur une machine spécifique ce qui fait gagner du temps.

La procédure d'installation est claire et on peut trouver des scripts adaptés à nos besoins grâce à la communauté.