

Ansible -- page ancienne



Sommaire

- [Intro](#)
- [Installation \(sur voidlinux\)](#)
 - [Quelques commandes utiles](#)
- [Annexe - Installation \(sur archlinux\)](#)
 - [Remarques importantes](#)
 - [Exemples simples](#)
 - [Comme ssh parallélisé](#)
 - [Liens, articles, etc.](#)
 - [Références](#)
 - [Voir aussi](#)

Intro

Ansible est un outil pour administrateur système de déploiement d'applications, de gestion centralisée de configuration et pour faire des actions ponctuelles sur des groupes de machines.

Installation (sur voidlinux)

Installation d'ansible sur voidlinux

```
sudo xbps-install -S ansible
mkdir -p /etc/ansible
cat > /etc/ansible/hosts <<EOF
[migration-puppet]
host1.ul.local
host2.ul.local
host3.ul.local
host4.ul.local
host5.ul.local
host6.ul.local
EOF
```

Quelques commandes utiles

Faire en sorte d'égrainer les hosts pour valider les clés des machines à ajouter aux known-hosts

```
# ssh-keyscan -f fichier_liste_de_dns_ou_ip_de_machines_une_par_ligne
# exemple avec un fichier de hosts à la *ansible*
time grep -v '^#\|^[:space:]]*\$|^[[^]]*\$' ~/ansible/hosts-def/hosts | ssh-keyscan -H -f - >> ~/.ssh/known_hosts
```

Quelques commandes utiles

```
ansible -i /etc/ansible/hosts migration-puppet -a 'grep "server = " /etc/puppet/puppet.conf'
ansible -i /etc/ansible/hosts migration-puppet -a 'sudo rm -rf /var/lib/puppet/ssl'
ansible -i /etc/ansible/hosts migration-puppet -a 'sudo puppet agent -t --server=nouveauserver.puppet.local'
```

Annexe - Installation (sur archlinux)

Installation ansible sur ArchLinux

```
yaourt -S ansible-git
# éditer le fichier PKGBUILD
# et changer comme suit :
diff --git a/PKGBUILD b/PKGBUILD
index 68dc6ce..3ec6221 100644
--- a/PKGBUILD
+++ b/PKGBUILD
@@ -34,7 +34,7 @@ package() {

  mkdir -p ${pkgdir}/usr/share/ansible
  cp ./library/* ${pkgdir}/usr/share/ansible/
  - python setup.py install -O1 --root=${pkgdir}
  + python2 setup.py install -O1 --root=${pkgdir}

  install -D docs/man/man1/ansible.1 ${pkgdir}/usr/share/man/man1/ansible.1
  install -D docs/man/man1/ansible-playbook.1 ${pkgdir}/usr/share/man/man1/ansible-playbook.1
# puis valider l'installation
```

Remarques importantes

- *Ansible* n'est utilisable que sur les machines ayant, au minimum, python 2.4 d'installé. Cela exclu donc les machines en RHEL 4.x qui n'ont que python 2.3.

- Certaines fonctionnalités, pour les machines en python 2.4, ne sont disponibles qu'à la condition d'installer, sur les machines clientes, le module *python-simplejson*.

Exemples simples

Comme ssh parallélisé

ls -al ~

```
ansible -i ~/ansible/hosts mailservers \  
-u monlogin \  
--private-key=~monlogin/.ssh/id_rsa \  
-m shell \  
-a 'HOME=$(/bin/grep monlogin /etc/passwd|cut -d: -f 6)/ ls -al $HOME'
```

- `-i ~/ansible/hosts` – Le fichier des noms/ip de machines administrées.
- `mailservers` – Le *groupe* de machines sur lesquelles faire les opérations.
- `-u monlogin` – Le login à utiliser pour se connecter en ssh sur les machines interrogées.
- `--private-key=~monlogin/.ssh/id_rsa` – Si on a besoin de donner une clé ssh particulière.
- `-m shell` – Utilisation du module *ansible shell* pour pouvoir faire des commandes à distance en masse et en parallèle sur toutes les machines.
- `-a ...` – Les arguments à passer au module utilisé (ici *shell*). Dans notre cas, il s'agit de la commande que l'on veut voir exécuter sur tout le groupe *mailservers*.

Liens, articles, etc.

- <https://news.ycombinator.com/item?id=11256819> – Concernant les interrogations concernant le grand nombre de modules sur *ansible-galaxy* : cette discussion donne quelques pistes sur le pourquoi de la chose. En gros : il y a, dans *ansible*, des problèmes d'espace de noms pour les variables et cela pose parfois des problèmes pour ce qui est de la réutilisabilité des modules poussés dans le **ansible-galaxy**. Sur ce lien, aussi, quelques autres avis de gros utilisateurs qui donnent quelques bémols à l'intérêt d'*ansible*. En résumé : gros problèmes de scalabilité pour plusieurs milliers de machines mais une très bonne lisibilité des recettes et donc une grande facilité à comprendre ce qui est fait.
- <https://communities.sas.com/t5/SAS-Communities-Library/Desired-State-is-a-Destination-not-a-Journey/ta-p/370736> – Une petite démonstration du fonctionnement d'*ansible*, avec des commandes et des explications qui donnent envie d'essayer ce logiciel.

Références

- Site web : <http://ansible.github.com/>

Voir aussi

- [Fabric - opérations sur un ensemble de machines](#)