

Gestion de configuration

Matthieu Herrb

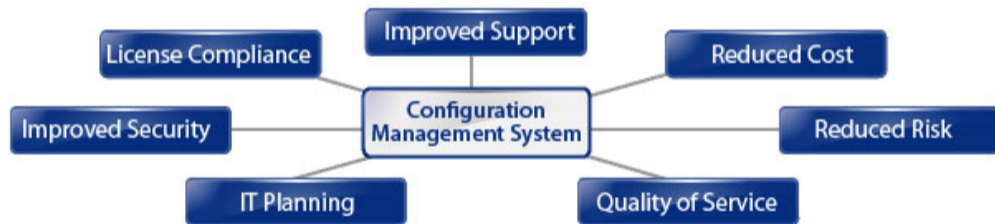


Capitoul - 17 octobre 2019

Introduction

Gestion de configuration...

- technique transversale, pas limitée à l'administration réseau ou système
- présente dans de nombreuses méthodes formelles de management (ITIL...)



« *Infrastructure as Code* »

But : automatisation de tâches répétitives pour :

- garantir l'homogénéité des configurations
- obtenir des rapports sur l'état du parc
- déployer rapidement des mises à jour
- propager des modifications de configuration
- améliore :
 - disponibilité
 - tracabilité
 - sécurité

Configuration - état

- fichiers de configuration (/etc/...)
- paquets logiciels installés

Où ?

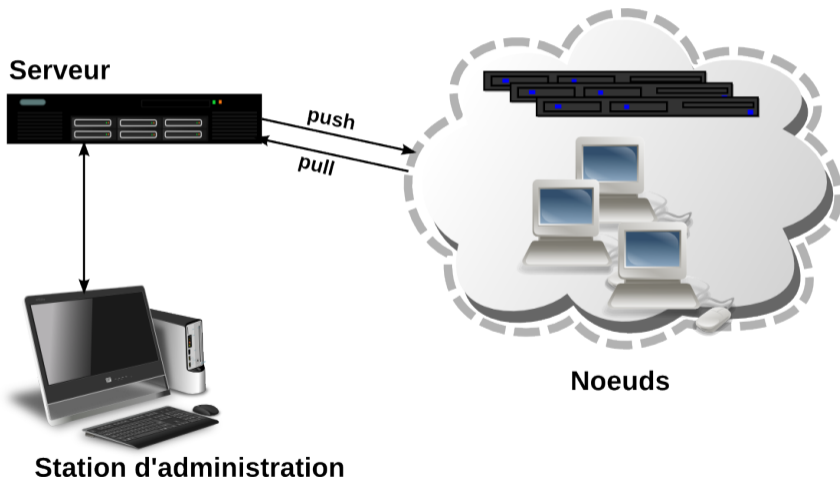
- sur des serveurs (machines virtuelles)
- sur des postes de travail, voire un parc de laptops,...
- sur des équipements réseau,

- manuelle : scripts maison
 - ssh + langages de scripts
- outils : ansible, cfengine, chef, puppet, saltstack, terraform...

Recouvrement/complémentarité avec d'autres outils :

- inventaire, suivi de problèmes (GLPI)
- déploiement (FAI, Kickstart, ...)
- gestion de virtualisation : libvirt, vSphere, OpenStack...
- gestion de containers : docker, kubernetes...

Architecture



Sur chaque système, il faut pouvoir faire :

- inventaire de l'état courant
- modification de fichiers de configuration
- installation/mise à jour de paquets
- exécution de commandes

Mode *push* ou *pull*.

Classes de machines :

Une machine est un objet qui dérive de plusieurs classes, avec possibilités de hiérarchie.

Par ex. :

- client NTP
- serveur SSH
- serveur messagerie (SMTP, IMAP,...)
- machine de tel bâtiment (pour config imprimantes par ex.)
- outils développement Java
- serveur application web (ex. Drupal), donc serveur apache

Lien avec les systèmes de gestion de version

- Notion d'environnements de production, de test
- Tags sur les configurations qui marchent à un instant t (snapshots)

⇒ utilisation de logiciels de gestion de version (git, subversion....) pour stocker les données de configuration (« recettes ») et gérer ces aspects.

- Droits d'admin (root) partout
- Le dépôt de toutes les configurations → information sensible
- Authentification forte (clés crypto) du serveur et des clients
- Mécanismes pour valider les modifications avant application
- Gestion des secrets dans les configurations (mots de passe, clés privées,...)

Systèmes de gestion de configuration :

- utiles en environnements virtualisés (+ *infonuagique*)
- utiles pour des parcs de postes de travail

Apports :

- efficacité dans le déploiement de configurations
- meilleure vue d'ensemble des systèmes
- aide pour mettre de l'ordre dans des systèmes trop confus

Difficultés potentielles :

- effort initial important
- peu d'intérêt face à un environnement en désordre qu'on ne veut pas changer