firewall name=firewall, description=Protection pour serveur
naxsi name=naxsi, description=Un type de firewall

Licence ASRALL

Exposé technique

Fully Automatic Installation

François Dupont

Florent Fillion

29 janvier 2015

Table des matières

Chapitre 1

Introduction

1.1 Objectifs

1.1.1 Définition

Fully Automatic Installation ou FAI est un logiciel libre, inspiré de son équivalent SOLARIS, JUMPSTART. Il permet d'installer et de configurer un système d'exploitation Linux sur une ou plusieurs machines, en utilisant un infrastructure client serveur, de façon rapide et automatisé.

Il est à noter que FAI n'est pas interactif contrairement à d'autres logiciels remplissant sensiblement la même fonction. Ce logiciel est également considéré comme mature (il en est à sa $4^{\text{ème}}$ itération majeure (4.0) et est développé de façon continue depuis 1999.

1.1.2 Utilité

FAI s'adresse particulièrement aux administrateurs ayant à gérer un grand parc de machine sous Linux, que ce parc soit *virtualisé* ou physique (et même des *chroot*s).

1.2 Histoire

1.3 Concepts

Cette partie pourrait avoir sa place dans le chapitre traitant de la technique à proprement parler, cependant, il nous semble essentiel d'exposer les principes fondamentaux (et simples une fois assimilés) de FAI.

Chapitre 2

Technique

Chapitre 3

Alternatives

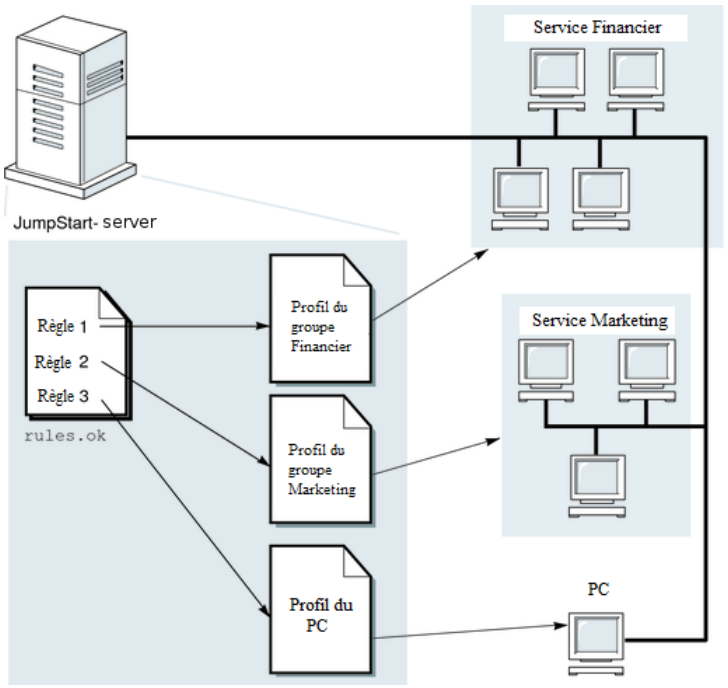
Il existe plusieurs alternatives à FAI, tels que Jumpstart, Kickstart, Rembo, ou encore Clonezilla par exemple.....

3.1 Jumpstart

Jumpstart a été créé par Sun Microsystems, qui fut ensuite racheté en 2009 par Oracle Corporation. C'est un outil permettant d'automatiser l'installation d'une ou plusieurs machines utilisant Solaris comme système d'exploitation.
La configuration de Jumpstart s'effectue à l'aide de plusieurs fichiers :

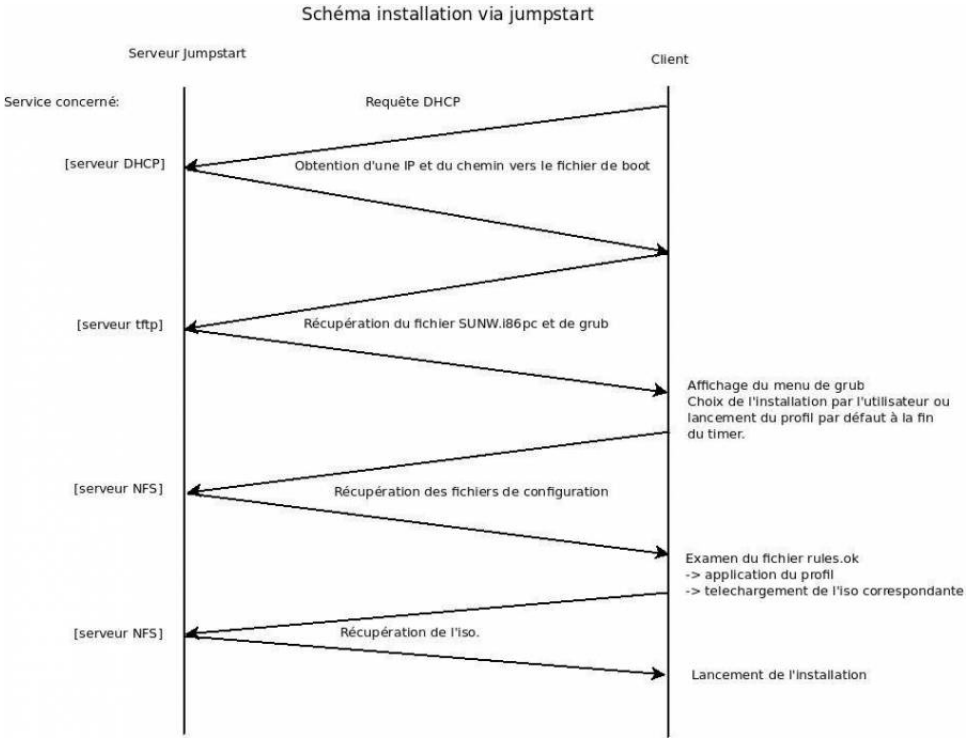
- Le fichier *profile* qui définit les partitions de disques.
- Les fichiers *rules* et *rules.ok* qui définissent les règles correspondant à un profil particulier de client à installer.
- Des scripts s'exécutant avant et après l'installation de la machine cliente.
- Le fichier *sysidcfg* qui définit les informations de configuration du client, comme par exemple son nom d'hôte, son adresse IP, etc...
- Le fichier */etc/bootparams* qui est utilisées par les machines clientes pour démarrer.
- Le fichier */etc/ethers* qui contient les mappages d'adresses MAC pour les clients.

Lors de l'installation, le programme Jumpstart va tenter de faire correspondre le sytème à installer aux règles définies dans le fichier *rules.ok*. Il lit donc ces règles, de la première à la dernière. Lorsque qu'une correspondance est établie entre un système et une règle, le programme JumpStart interrompt la lecture du fichier *rules.ok* et commence l'installation du système d'après le profil correspondant à la règle associée.



Un serveur DHCP peut également être mis en place sur le serveur Jumpstart afin d'attribuer automatiquement une configuration IP aux machines clientes.

Voici un schéma représentant brièvement les requêtes effectuées entre les clients et le serveur lors du démarrage de l'installation :



Configuration du serveur Jumpstart

Voici la liste des tâches à effectuer lors de la préparation à une installation JumpStart :

- Créer un répertoire JumpStart

- Ajouter des règles pour chaque groupe dans le fichier *rules* :
  Chaque règle définit un groupe d'après un ou plusieurs attributs système. La règle lie chaque groupe à un profil.

- Créer un profil pour chaque règle :
  Un profil est un fichier texte qui définit l'installation du logiciel Solaris, et indique par exemple le groupe de logiciels devant être installé sur un système. À chaque règle correspond un profil qui définit la procédure d'installation du logiciel Solaris sur un système. Ce profil est utilisé dès qu'une correspondance est établie entre une règle et un système déterminés.
  Généralement, on définit un profil pour chaque règle. Le même profil peut toutefois être utilisé dans plusieurs règles.

- Valider le fichier *rules* :

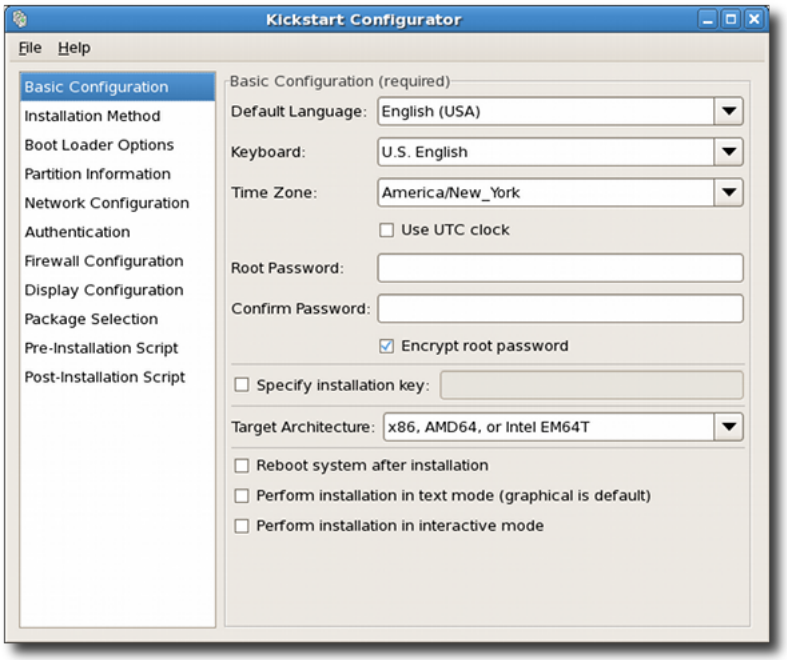
Le fichier *rules.ok* est une version générée du fichier *rules* qu'utilise Jump-Start pour détecter le système à installer avec un profil. Le fichier *rules* se valide par l'intermédiaire d'un script *check*.

3.2 Kickstart

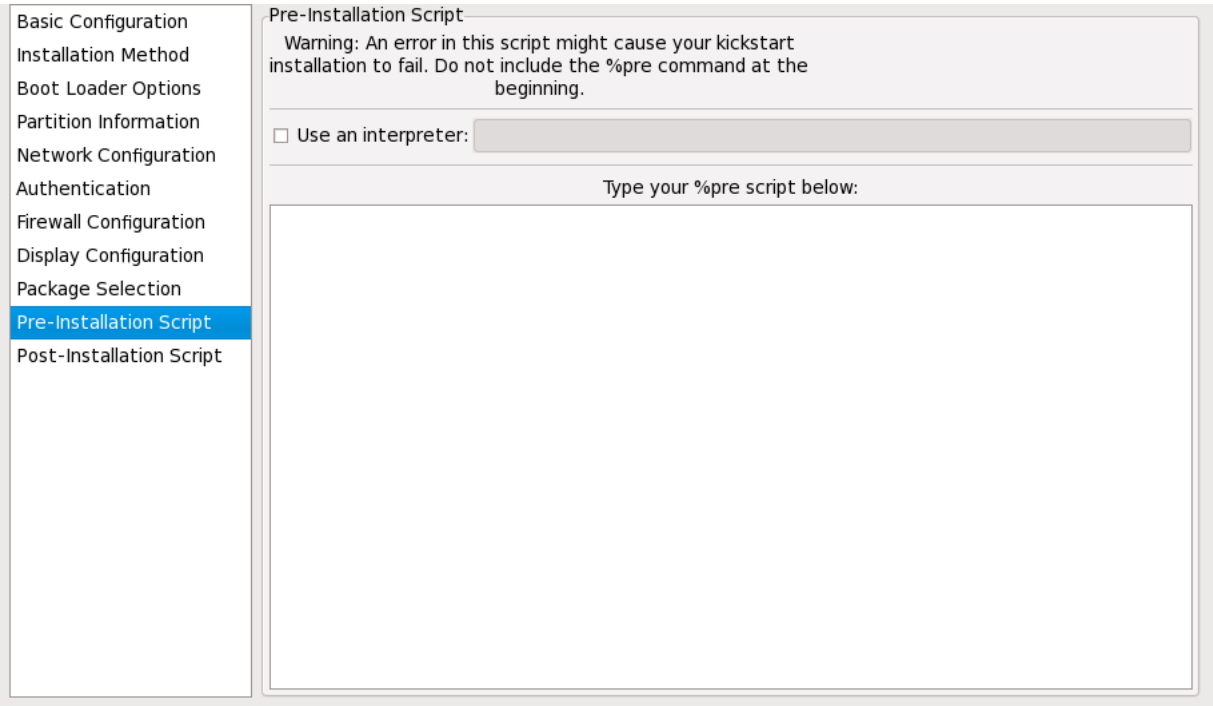
Kickstart a été créé par RedHat afin d'automatiser l'installation de Fedora et Red Hate Enterprise Linux. Sa configuration s'effectue dans un fichier kickstart *ks.cfg* unique contenant une liste d'éléments, chacun identifié par un mot-clé.
Ces éléments correspondent aux réponses à toutes les questions qui devraient normalement être posées lors d'une installation typique.
Ce fichier peut être créé manuellement en partant d'un fichier vierge puis en écrivant les directives une par une.

Cependant, et contrairement à FAI, il existe une interface utilisateur graphique, "Kickstart Configuration", permettant ainsi une configuration plus simple puisque il n'est pas nécessaire de se rappeler de la syntaxe exacte du fichier kickstart à configurer.

Il suffit donc de suivre les étapes les unes après les autres afin de configurer petit à petit le fichier en question :



Si l'on souhaite effectuer une installation plus fine et personnalisée, il est également possible de rajouter des scripts d'installations via l'interface graphique. Ces scripts peuvent être exécuté avant (Pre-Installation Script) ou après l'installation (Post-Installation Script) :



Si l'on souhaite par exemple effectuer manuellement une configuration réseau dans le fichier kickstart, il suffira d'inclure le script dans la zone de texte prévue à cet effet.
En effet, avant le début de l'installation, le fichier kickstart est analysé afin des détecter les différentes commandes à exécuter. Si une configuration réseau est détectée dans le fichier kickstart, avant que la section "Network Configuration" soit traitée, la configuration réseau sera prioritaire et prise en compte.
Pour spécifier un langage de script particulier à utiliser pour exécuter le script, il suffit de sélectionner l'option "use an interpreter" et d'entrer le langage souhaité, par exemple */usr/bin/python2.7*.

Une fois la configuration terminée, il est tout de même possible de consulter le fichier kickstart avant de le sauvegarder.

Il suffit ensuite de placer le fichier *ks.cfg* généré sur le support de démarrage souhaité, puis de booter la machine cliente sur ce même support de démarrage.

Une commande de démarrage spéciale est à taper à l'invite de démarrage. Le programme d'installation va ainsi rechercher si un fichier kickstart est présent, puis va s'en servir pour procéder à l'installation du système.

3.3 Rembo

Rembo est le successeur de BPBatch. Il a été créé par "Rembo Technology", une société suisse spécialisée dans les outils de gestion applicative et de virtualisation de serveurs, qui fut ensuite rachetée en 2006 par IBM. Rembo est un environnement de déploiement d'applications qui permet d'installer rapidemment et en même temps, n machines du même type. Il peut s'exécuter sur Windows, Linux ou encore FreeBSD et s'appuie sur un serveur DHCP. Les machines clientes doivent donc supporter PXE et auront dont exactement la même installation.

L'installation s'appuie sur l'utilisation d'une platine virtuelle, sur laquelle on sélectionne les programmes à installer. La procédure est ensuite entièrement automatique, et personnalisable à l'infini. Une machine virtuelle est créée à chaque installation puis est effacée lorsque le processus est terminé.
.....

3.4 Clonezilla

Clonezilla est une logiciel libre créé par le laboratoire de recherche NCHC pouvant être utilisé sur plusieurs systèmes d'exploitations différents tel que Linux, Windows ou encore MAC OS X par exemple. Il permet de créer une image de sauvegarde d'un disque dur ou d'une ou plusieurs partitions, pour ensuite la restaurer ou la cloner sur une (unicast) ou plusieurs machines clientes (multicast).
Il existe deux version de Clonezilla, "Clonezilla Live" et "Clonezilla Server" :

- Clonezilla Live est utilisé par l'intermédiaire d'un Live CD, (ou clé usb également). Il permet à l'utilisateur d'effectuer directement depuis la machine une sauvegarde d'un disque ou d'une partition, une restauration d'une image depuis un disque, ou encore un clonage d'une image vers un autre disque. Cepedendant, ces actions ne peuvent s'effectuer que sur une seule machine à la fois puisque le Live CD est obligatoire.

- Clonezilla Server (ou Clonezilla SE) est utilisé depuis un serveur et autorise plusieurs postes à se connecter simultanément par l'intermédiaire d'un réseau commun. Il n'y a donc pas besoin de Live CD ce qui permet d'effectuer

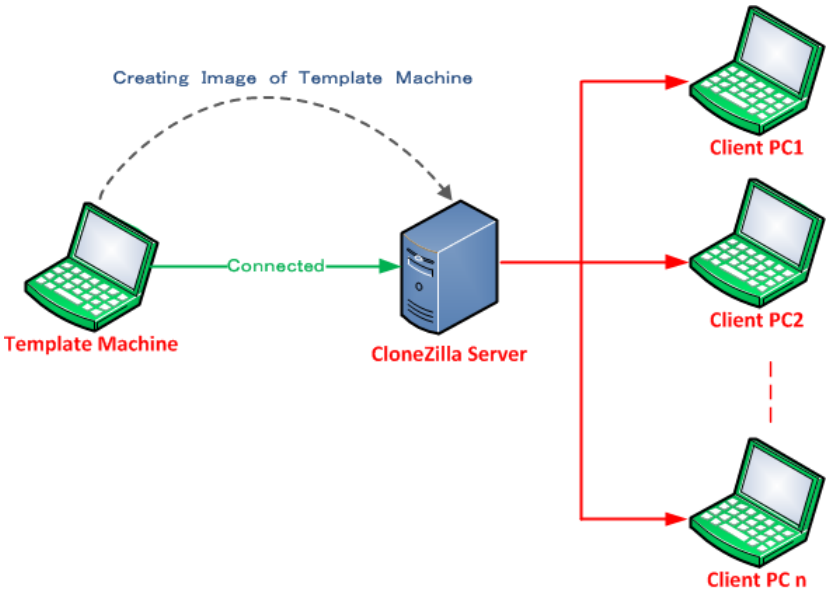
des sauvegardes/restaurations/clonages sur plusieurs machines à la fois. Un server DRBL (Diskless Remote Boot in Linux) est utilisé ce qui permet aux stations esclaves concernées d'effectuer un démarrage PXE.

Le serveur DRBL

DRBL permet de mettre à disposition un environnement système léger pour les machines clientes et intègre Clonezilla.
Il est donc utilisé, le plus souvent, pour le clonage massif de postes de travail et le déploiement rapide de salles informatiques. Il s'appuie exclusivement sur des logiciels libres tel que *partimage*, *ntfsclone* ou encore *partclone* pour cloner des disques/partitions, et utilise également *udpcast* pour transférer des données simultanément sur plusieurs destinations via le réseau (unicast/broadcast/multicast).

DRBL supporte les systèmes de fichiers ext2, ext3, ext4, reiser4, xfs, jfs, fat, ntfs, hfs+, ufs, vmfs3 ainsi que LVM2. Les bootloader syslinux et grub peuvent être installés.

Principe général de fonctionnement de Clonezilla Server



Le serveur Clonezilla a besoin d'une image type, afin qu'il puisse s'en servir par la suite comme référence pour sauvegarder, restaurer ou cloner cette même image sur les différentes machines clientes.
Lors d'un clonage par exemple, une fois l'image disponible sur le serveur, celui-ci va la copier simultanément sur les machines clientes, via le réseaux.

Chapitre 4

Conclusion

Chapitre 5

Sources

- http ://fai-project.org/
- https ://docs.oracle.com/
- https ://access.redhat.com/documentation/
- http ://http ://clonezilla.org/
- https ://www.projet-plume.org/fiche/drblclonezilla