

Déploiement PXE de postes clients Ubuntu avec le mode EFI en utilisant Kickstart, Preseed et Chef

Laurent Dang

CNRS, UMR8628

laurent.dang@math.u-psud.fr

18 mai 2017

1 Introduction

- Problématique
- PXE : installation par le réseau
- PXE : déroulement d'une installation par le réseau

2 Deuxième partie

- Configuration Kickstart pour installer les postes Ubuntu en mode EFI
- Configuration Preseed pour installer les postes Ubuntu en mode EFI

3 Troisième partie

- Chef: qu'est ce que c'est ?
- Utilisation de Chef pour déployer les logiciels sur les postes en mode EFI

Comment éviter une installation manuelle sans oublier d'installer des paquetages ? Comment faciliter la gestion, l'installation et la mise à jour d'un ensemble de machines en mode EFI quand il y en a beaucoup sans perdre trop de temps ?

Plusieurs solutions existent pour répondre à ces questions et nous allons ici parler de ce qui a été mis en place au sein du LMO (Laboratoire de Mathématiques d'Orsay)

Comment automatiser l'installation d'une nouvelle machine en mode EFI sous Ubuntu ?

Pour automatiser l'installation d'une nouvelle machine en mode EFI sous Ubuntu, il faut d'abord installer un serveur PXE, configurer un serveur DHCP et créer un répertoire tftpboot/grub contenant le fichier **grubnetx64.efi.signed**. Ce fichier est nécessaire pour démarrer le processus d'installation en PXE et en mode EFI sur une machine. Il faut ensuite alors configurer le serveur DHCP comme suit :

```
host hortensia hardware ethernet 34:17:eb:c2:f7:30;  
fixed-address xxx.yyy.zzz.ttt;filename "grub/grubnetx64.efi.signed";  
next-server nom_du_serveur_pxe.nom_du_domaine;
```

Nécessite du réseau et une infrastructure (TFTP,DHCP,DNS)

- Installation rapide
- Pas de support physique
- Plusieurs installations en parallèle possible
- Distribution parfaitement à jour dès la fin de l'installation

PXE : déroulement d'une installation par le réseau

- La nouvelle machine démarre sur le réseau (en mode PXE) et demande au serveur DHCP une adresse IP et les informations qui lui permettront de trouver son image de boot
- La nouvelle machine récupère sur le serveur TFTP son image de boot
- La nouvelle machine prend sur le serveur http le fichier Kickstart ou Preseed
- La nouvelle machine effectue son installation en récupérant les paquets sur le serveur miroir local ou distant (l'accès internet est alors indispensable dans ce dernier cas)

La machine peut alors démarrer en mode EFI et en PXE. On peut configurer et personnaliser le menu GRUB qui apparaît en modifiant le fichier `grub.conf` en abordant les deux approches suivantes :

- Première étude avec Kickstart en EFI
- Deuxième étude avec Presseseed en EFI

Fichier texte donné en paramètre à l'installateur
Plusieurs grandes parties :

- Les informations de configuration
- Les informations sur les paquets à installer
- Les scripts de pré et de post installation

Première étude avec Kickstart en EFI

Dans le cadre d'une installation Kickstart PXE en EFI, la configuration de GRUB sur le serveur est la suivante :

```
menuentry "Install Ubuntu Doct EFI"  
{  
  set gfxpayload=keep  
  linux /pxe/ubuntu_16.04/amd64/ubuntuinstaller/amd64/vmlinuz.efi  
  gfxpayload=800x600x16,800x600  
  quiet splash  
  ks= "http://nom_du_serveur_pxe/pxe/nom_du_fichier_kickstart.cfg"  
}
```

On a ajouté quelques options sur la ligne linux (sur une seule ligne). En voici le détail :

ks : où trouver le fichier de configuration Kickstart

linux : où se trouve le noyau Linux en mode EFI qu'on utilise pour effectuer l'installation

gfxpayload : remplace le paramètre vga dans les nouvelles versions de Grub dans la ligne "linux"

quiet splash : on désactive l'image ubuntu et on fait l'installation en mode texte sur l'écran

Configuration Kickstart associé à la configuration de Grub

La configuration Kickstart est celle d'une configuration standard mis part la partie partitionnement du disque :

```
clearpart --all --drives=sda  
ignoredisk --only-use=sda  
preseed partman-auto/init_automatically_partition select  
biggest_free
```

On choisit de créer une seule partition / dont la taille est maximale automatiquement.

Presseed, comment ça marche ?

Basé sur les mêmes principes que Kickstart, un fichier qui répond aux questions de l'installateur Debian (d-i). Celui-ci ouvre normalement tout un tas de boîtes de dialogues selon un scénario bien précis. Si une boîte de dialogue dispose déjà d'une réponse, elle ne sera pas présentée à l'utilisateur. **FORME : d-i module/paramètre type réponse**

- **d-i** indique qu'on s'adresse à l'installateur Debian
- **module** indique quel est le module de l'installateur Debian concerné. Par exemple un module s'occupe de la configuration réseau (netcfg), un autre du partitionnement (partman) etc
- **paramètre** indique quelle est la variable qu'on souhaite pré-remplir
- **type** indique quel est le type de la variable concernée (ex : string pour une chaîne de caractère)
- **réponse** contient la valeur affectée à la variable

Exemple : **d-i keyboard-configuration/layoutcode string fr**

Configurer le clavier, la langue et le pays de IOS :

```
d-i debconf/language string fr
d-i debian-installer/country string FR
d-i debian-installer/language string fr
d-i debian-installer/locale select fr_FR.UTF-8
d-i console-setup/ask_detect boolean false
d-i keyboard-configuration/modelcode string pc105
d-i keyboard-configuration/layoutcode string fr
d-i keyboard-configuration/variantcode string latin9
```

Configurer le proxy avec preseed :

```
d-i mirror/http/proxy string http://nom_du_serveur_proxy:3128
```

Deuxième étude avec Preseed en EFI

Dans le cadre d'une installation Preseed PXE en EFI, la configuration de GRUB sur le serveur est la suivante :

```
menuentry "Install Ubuntu Doct EFI"
```

```
{
```

```
set gfxpayload=keep
```

```
linux /pxe/ubuntu_16.04/amd64/ubuntuinstaller/amd64/vmlinuz.efi
```

```
gfxpayload=800x600x16,800x600
```

```
quiet splash
```

```
preseed/url= "http://nom_du_serveur_pxe.nom_du_domaine/pxe/  
nom_du_fichier_preseed.seed"
```

```
}
```

On utilise les options *locale=fr_FR console-keymaps-at/keymap=fr-latin9 languagechooser/language-name=French countrychooser/shortlist=FR priority=critical* qu'on rajoute à la configuration précédente pour lancer l'installation et pour choisir la langue d'installation du système. Ici on choisit le français comme langue par défaut.

La configuration Preseed sur le serveur est une configuration preseed standard où on précise les paquetages à installer. On y définit la manière de partitionner le disque pour tout rendre automatique :

```
d-i partman-lvm/device_remove_lvm boolean true  
d-i partman-lvm/confirm boolean true  
d-i partman-lvm/confirm_nooverwrite boolean true
```

On choisit de supprimer toutes les partitions LVM existantes et d'éviter la confirmation de l'effacement complet des partitions.


```
d-i partman-auto/purge_lvm_from_device boolean true  
d-i partman-auto/disk string /dev/sda  
d-i partman-auto/method string regular  
d-i partman-auto/init_automatically_partition select biggest_free
```

On choisit le partitionnement automatique des partitions et on choisit de faire une seule partition / dont la taille est maximale.

Configuration Preseed

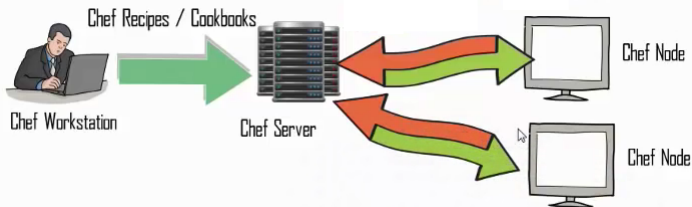
```
d-i partman-md/confirm boolean true  
d-i partman-md/confirm_nooverwrite boolean true  
d-i partman-md/device_remove_md boolean true
```

```
d-i partman-partitioning/confirm_write_new_label boolean true  
d-i partman/choose_partition select finish  
d-i partman/confirm boolean true  
d-i partman/confirm_nooverwrite boolean true
```

Pour éviter les messages de confirmation lors du partitionnement automatique on utilise les instructions ci dessus.

Chef: qu'est ce que c'est ?

Chef est un logiciel libre de gestion de configuration écrit en Ruby. Il utilise un langage dédié en pure Ruby pour l'écriture de configuration sous la forme de recettes ou de livres de recettes. On l'utilise en mode client-serveur au laboratoire pour installer les logiciels automatiquement sur les machines du parc informatique.



Exemples simples de recettes Chef

```
❶ package 'scilab'
do
  action [:install]
end

❷ for p in ["libatk1.0-0:i386",
  ...,
  "libstdc++6:i386"]
do
  package p
do
  action [:install]
end
end
```

- ❶ On installe ici le paquetage scilab sur les machines avec l'outil Chef (équivalent à un apt-get install scilab)
- ❷ équivalent à un apt-get install libatk1-0-0:i386....

- dans le fichier Kickstart
- dans le fichier Preseed

Utilisation de Chef

La clé privée validation.pem et la clé publique du serveur Chef chef-srv.domaine.fr sont nécessaires et doivent être copiés sur le client Chef dans le répertoire chef pour permettre au serveur de reconnaître le client:

```
log_location STDOUT
```

```
chef_server_url "https://chef-srv.domaine.fr/organizations/lmo"
```

```
validation_client_name "lmo-validator"
```

```
node_name "hortensia"
```

```
trusted_certs_dir "/etc/chef/trusted_certs"
```

Dans l'exemple la machine cliente Chef porte le nom "hortensia"
On doit créer tous les fichiers dans le répertoire /etc/chef/ indiqué ci dessus en utilisant la commande cat de Linux dans la configuration Kickstart.

Utilisation de Chef pour déployer les logiciels sur les postes en mode EFI

avec Preseed

Dans le fichier preseed on rajoute la ligne :

```
d-i preseed/late_command string chroot /target  
sh -c "/usr/bin/curl -o /tmp/postinstall  
http://nom_du_serveur_pxe.nom_du_domaine/pxe/postinstall  
&& /bin/bash -x /tmp/postinstall"
```

pour exécuter le script postinstall dans la configuration Preseed.

Le fichier **postinstall** contient les commandes pour copier les certificats nécessaires et pour rajouter automatiquement une machine - un noeud - sur le serveur Chef. Pour cela, on utilise la commande chef-client avec l'option -j depuis la machine cible et on l'exécute depuis le fichier /etc/rc.local qui est lu au redémarrage.

Exemple d'utilisation de Chef avec Kickstart et Preseed via un serveur proxy

On utilise ici la recette install du serveur Chef qu'on aura défini :

- Fichier `/etc/chef/first-boot.json` :

```
{ "run_list": [ "recipe[installation_logiciels::install]" ] }
```
- Fichier `/etc/rc.local` :

```
export http_proxy=http://nom_du_serveur_proxy:3128  
export https_proxy=https://nom_du_serveur_proxy:3128  
chef-client -j /etc/chef/first-boot.json
```

Exemple d'utilisation de Chef avec Kickstart et Preseed

On utilise ici la recette install du serveur Chef qu'on aura défini :

- Fichier `/etc/chef/first-boot.json` :

```
{ "run_list": [ "recipe[installation_logiciels::install]" ] }
```
- Fichier `/etc/rc.local` :

```
chef-client -j /etc/chef/first-boot.json
```

Merci à tous
Questions ?