

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

Laurent Dang

IMO, Institut de Mathématiques d'Orsay (CNRS UMR 8628)

25 janvier 2018

Contact: [laurent.dang@math.u-psud.fr](mailto:laurent.dang@math.u-psud.fr)

# Sommaire

- 1 Problématique, étude des configurations PXE en mode EFI
  - Problématique
  - PXE : automatiser l'installation d'une nouvelle machine
  - PXE : installation par le réseau
  - PXE : déroulement d'une installation par le réseau
  - Installation PXE : configuration Kickstart en mode EFI
  - Installation PXE : configuration Preseed en mode EFI
- 2 Introduction à Chef et son utilisation avec PXE
- 3 Annexes

# Problématique

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 IMO ( Institut 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/debian-installer/amd64/` contenant le fichier **bootnetx64.efi**. Ce fichier est nécessaire pour démarrer le processus d'installation en PXE et en mode EFI sur une machine.

On peut le trouver dans l'archive netboot fourni par Debian sur le Net.

Il faut ensuite alors configurer le serveur DHCP comme suit :

Exemple de configuration DHCP pour une nouvelle machine

```
host hortensia hardware ethernet 34:17:eb:c2:f7:30;  
fixed-address xxx.yyy.zzz.ttt;filename "/debian_installer/amd64/bootnetx64.efi";  
next-server nom-DNS-complet-du-serveur-PXE;
```

# PXE : installation par le réseau

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

- Installation rapide

# PXE : installation par le réseau

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

- Installation rapide
- Pas de support physique

# PXE : installation par le réseau

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

- Installation rapide
- Pas de support physique
- Plusieurs installations en parallèle possible



# PXE : installation par le réseau

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

# 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 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

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 Preseed en EFI

# Kickstart: qu'est ce que c'est et comment ça marche ?

Kickstart est un utilitaire permettant l'automatisation de l'installation sur les systèmes Redhat.



# Kickstart: qu'est ce que c'est et comment ça marche ?

Kickstart est un utilitaire permettant l'automatisation de l'installation sur les systèmes Redhat.

Fichier texte donné en paramètre à l'installateur

# Kickstart: qu'est ce que c'est et comment ça marche ?

Kickstart est un utilitaire permettant l'automatisation de l'installation sur les systèmes Redhat.

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

- Les informations de configuration

# Kickstart: qu'est ce que c'est et comment ça marche ?

Kickstart est un utilitaire permettant l'automatisation de l'installation sur les systèmes Redhat.

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

- Les informations de configuration
- Les informations sur les paquets à installer

# Kickstart: qu'est ce que c'est et comment ça marche ?

Kickstart est un utilitaire permettant l'automatisation de l'installation sur les systèmes Redhat.

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 :

## Configuration GRUB avec Kickstart

```
menuentry "Install Ubuntu Doct EFI" {  
  set gfxpayload=keep  
  linux /pxe/ubuntu_16.04/amd64/ubuntu-installer/amd64/vmlinuz.efi gfxpayload=800  
    x600x16,800x600 quiet splash ks=http://nom-complet-DNS-serveur-pxe/pxe/nom-  
    fichier-kickstart.cfg lang=fr noipv6  
  initrd /pxe/ubuntu_16.04/amd64/ubuntu-installer/amd64/initrd.gz  
}
```

# Menu GRUB avec PXE

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 :

Création d'une seule partition / de taille maximale

```
clearpart --all --initlabel
clearpart --all --drives=sda
ignoredisk --only-use=sda
preseed partman-auto/init automatically partition select biggest free
```

# Preseed: qu'est ce que c'est et comment ça marche ?

Preseed est le système d'automatisation présente dans les distributions Debian et Ubuntu. Il est plus complexe à utiliser que Kickstart.

Basé sur les mêmes principes que ce dernier, la configuration Preseed se présente sous forme d'un fichier texte 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

## FORME : d-i module/paramètre type réponse

- **d-i** indique qu'on s'adresse à l'installateur Debian

## FORME : d-i module/paramètre 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

## 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

## 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 (par exemple string pour une chaîne de caractère)

## 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 (par exemple string pour une chaîne de caractère)
- **réponse** contient la valeur affectée à la variable

# Preseed : localization, keyboard

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
```

# Preseed : configuration du proxy

Configurer le proxy avec preseed

```
d-i mirror/http/proxy string http://nom-DNS-complet-du-serveur-proxy:3128
```



# Configuration GRUB avec Preseed en EFI

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

## Configuration de GRUB avec Preseed

```
menuentry "Install Ubuntu 16.04 EFI+Chef full preseed" {  
  set gfxpayload=keep  
  linux /pxe/ubuntu_16.04/amd64/ubuntu-installer/amd64/vmlinuz.efi gfxpayload=800  
    x600x16,800x600 quiet splash preseed/url=http://nom-complet-DNS-serveur-pxe  
    /pxe/nom-du-fichier-preseed.seed  
  initrd /pxe/ubuntu_16.04/amd64/ubuntu-installer/amd64/initrd.gz  
}
```

On utilise les options suivantes :

#### Exemple de configuration

```
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.

# Exemple complet de configuration GRUB

## Exemple de fichier grub.cfg

```
menuentry "Install Ubuntu 16.04 EFI+Chef full preseed" {
  set gfxpayload=keep
  linux /pxe/ubuntu_16.04/amd64/ubuntu-installer/amd64/vmlinuz.efi gfxpayload=800
    x600x16,800x600 quiet splash locale=fr_FR console-keymaps-at/keymap=fr-
    latin9 languagechooser/language-name=French countrychooser/shortlist=FR
    priority=critical auto=true preseed/url=http://nom-complet-DNS-serveur-pxe/
    pxe/nom-du-fichier-preseed.seed
  initrd /pxe/ubuntu_16.04/amd64/ubuntu-installer/amd64/initrd.gz
}
```

# Configuration Preseed

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 :

Configuration Preseed : suppression des partitions LVM sans confirmation de l'effacement complet des partitions

```
d-i partman-lvm/device remove lvm boolean true
d-i partman-lvm/confirm boolean true
d-i partman-lvm/confirm nooverwrite boolean true
```

# Configuration Preseed

Configuration Preseed : automatisation du partitionnement avec une partition / de taille maximale

```
d-i partman-md/confirm boolean true
d-i partman-md/confirm nooverwrite boolean true
d-i partman-md/device remove md boolean true
```

# Configuration Preseed

Configuration Preseed: comment tout automatiser sans confirmation 1/2

```
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
```

### Configuration Preseed: comment tout automatiser sans confirmation 2/2

```
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
```

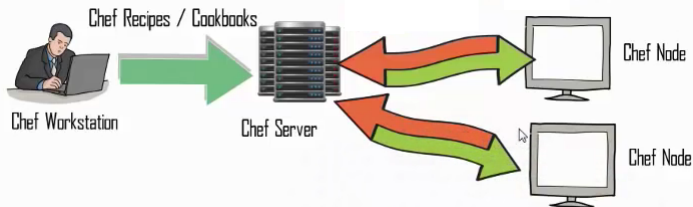
# Sommaire

- 1 Problématique, étude des configurations PXE en mode EFI
- 2 Introduction à Chef et son utilisation avec PXE
  - Chef: qu'est ce que c'est ?
  - Quelques exemples de recettes Chef
  - Comment utiliser Chef ?
  - Comment utiliser Chef avec Kickstart, Preseed ?
  - Comment utiliser Chef en utilisant un proxy ?
  - Pour en savoir plus sur Chef
- 3 Annexes



# 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....`

# Comment utiliser Chef ?

Les certificats SSL du serveur Chef sont nécessaires et doivent être copiés sur le client Chef dans le répertoire **/etc/chef/trusted\_certs** pour permettre au serveur de reconnaître le client et pour permettre la communication en SSL. Ce sont les mêmes fichiers que ceux utilisés lorsqu'on utilise la commande **knife ssl fetch** et qui sont stockés dans le même répertoire **/etc/chef/trusted\_certs** sur le référentiel Chef.

# Exemple de configuration sur un client Chef

Fichier `/etc/chef/client.rb`

```
log location STDOUT
chef server url "https://nom-DNS-complet-du-serveur-Chef/organizations/
ORGANIZATION_NAME"
validation client name "ORGANIZATION_NAME-validator"
node name hortensia
trusted_certs_dir "/etc/chef/trusted_certs"
```

# Comment installer Chef sur une nouvelle machine ?

## Installation de Chef 1/2

```
curl -L https://www.opscode.com/chef/install.sh | bash
mkdir -p /etc/chef

cat << EOF >/etc/chef/validation.pem

-----BEGIN RSA PRIVATE KEY-----
XXXXXXXXXXXXXXXXXXXXXXXXYYYYYYYYYYYYYYYYYYY
ZZZZZZZZZZZZZZZZZZZZZZZZZZZZBBBBBBBBBBBB
CCCCCCCCCCCCCCCCCCCCCTTTTTTTTTTTTTTTTTT
DDDDDDDDDDDDDDDDDDDDDDUUUUUUUUUUUUUUUUUU
VVVVVVVVVVVVVVVVVVVVVVVAAAAAAAAAAAAAAAA
-----END RSA PRIVATE KEY-----
EOF
```

# Comment installer Chef sur une nouvelle machine ?

## Installation de Chef 2/2

```
mkdir -p /etc/chef/trusted_certs

cat << EOF >/etc/chef/trusted_certs/certificat_Chef.crt

-----BEGIN CERTIFICATE-----
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABBBBBBBBBBBBBBBBBBBBBB
CCCCCCCCCCCCCCCCCCCCCCCCDDDDDDDDDDDDDDDDDDDDDDDD
YYYYYYYYYYYYYYYYYYYYYYYYYYYYEEEEEEEEEEEEEEEEEEEE
HHHHHHHHHHHHHHHHHHHHHHHHCCCCCCCCCCCCCCCCCCCC
-----END CERTIFICATE-----
EOF
```

# Comment utiliser Chef avec

- Kickstart ?

avec Kickstart

```
cat << EOF >/etc/chef/client.rb
log_location      STDOUT
chef_server_url   "https://nom_complet_DNS_du_serveur_chef/
                  organizations/ORGANIZATION_NAME"
validation_client_name "ORGANIZATION_NAME-validator"
node_name         "$HOSTNAME"
ssl_verify_mode   :verify_none
trusted_certs_dir "/etc/chef/trusted_certs"
EOF
```



avec Kickstart

```
cat << EOF >/etc/chef/first-boot.json
{"run_list":["recipe[installation_logiciels::install]"]}
EOF

chmod -R o+r /etc/chef/*
cat << EOF >/etc/rc.local
chef-client -j /etc/chef/first-boot.json
exit
EOF
```

## Explications

Dans l'exemple, la machine cliente Chef porte le nom que l'on lui attribue via la variable **\$HOSTNAME**. 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 - on peut utiliser un simple script Bash pour cela - puis on utilise la recette `install` du serveur Chef qu'on aura défini comme suit :

- Fichier `/etc/chef/first-boot.json` : `{"run_list":["recipe[installation_logiciels::install]"]}`

## Explications

Dans l'exemple, la machine cliente Chef porte le nom que l'on lui attribue via la variable **\$HOSTNAME**. 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 - on peut utiliser un simple script Bash pour cela - puis on utilise la recette `install` du serveur Chef qu'on aura défini comme suit :

- Fichier `/etc/chef/first-boot.json` : `{"run_list":["recipe[installation_logiciels::install]"]}`
- Fichier `/etc/rc.local` :  
`chef-client -j /etc/chef/first-boot.json`

# Comment utiliser Chef avec

- Preseed ?

# Utilisation de Chef avec Preseed

## avec Preseed

Dans le fichier preseed qu'on aura créé, on rajoute la ligne :

```
d-i preseed/late command string chroot /target  
    sh -c "/usr/bin/curl -o /tmp/postinstall  
http://nom-DNS-complet-du-serveur-PXE/pxe/  
postinstall  
&& /bin/bash -x /tmp/postinstall"
```

pour exécuter dans la configuration Preseed le script postinstall qui se trouve dans le répertoire pxe du serveur PXE.

### Exemple d'utilisation de Chef avec Preseed

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

- Fichier `/etc/chef/first-boot.json` : `{"run_list":["recipe[installation_logiciels::install]"]}`

### Exemple d'utilisation de Chef avec Preseed

On utilise 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`

#### Exemple de script postinstall 1/4

```
#!/bin/bash

curl -L https://www.opscode.com/chef/install.sh | bash
mkdir -p /etc/chef

cat << EOF >/etc/chef/validation.pem
-----BEGIN RSA PRIVATE KEY-----
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDDDDDDDDDDDDDDDDDDDDDDDDDD
-----END RSA PRIVATE KEY-----
EOF
```



#### Exemple de script postinstall 2/4

```
cat << EOF >/etc/chef/trusted_certs/certificat_Chef.crt
-----BEGIN CERTIFICATE-----
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABBBBBBBBBBBBBBBBBBBBBB
ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
-----END CERTIFICATE-----

EOF
```

### Exemple de script postinstall 3/4

```
cat << EOF >/etc/chef/client.rb
log_location      STDOUT
chef_server_url   "https://nom_complet_DNS_du_serveur_chef/
                  organizations/ORGANIZATION_NAME"
validation_client_name "ORGANIZATION_NAME-validator"
node_name         "$HOSTNAME"
ssl_verify_mode   :verify_none
trusted_certs_dir "/etc/chef/trusted_certs"
EOF
```

#### Exemple de script postinstall 4/4

```
cat << EOF >/etc/chef/first-boot.json
{"run_list":["recipe[installation_logiciels::install]"]}
EOF

cat << EOF >/etc/rc.local
#!/bin/sh -e
chef-client -j /etc/chef/first-boot.json
exit
EOF
```

### Le fichier postinstall: explications

Le fichier **postinstall** contient les commandes pour copier les certificats nécessaires (on utilise la commande `cat`) 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.

### Attention !

On pensera bien à remettre à jour ce fichier dans la recette **install.rb** sur le serveur Chef

### Exemple d'utilisation de Chef 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]}]}`

### Exemple d'utilisation de Chef 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`

# Pour en savoir plus sur Chef

Pour obtenir plus d'informations sur l'outil Chef, je vous renvoie ma présentation faite à Calais lors des journées Mathrice du 26 au 28 septembre 2017 : [Lien présentation](#)

# Sommaire

- 1 Problématique, étude des configurations PXE en mode EFI
- 2 Introduction à Chef et son utilisation avec PXE
- 3 Annexes



# Fichier complet kickstart

Exemple de script kickstart 1/9

```
lang fr_FR.UTF-8
langsupport fr_FR.UTF-8 --default=fr_FR.UTF-8
keyboard fr
mouse
timezone Europe/Paris
rootpw --iscrypted mot_de_passe_md5_root
user info --fullname "Service informatique" --iscrypted --
    password mot_de_passe_md5_info
```

# Fichier complet kickstart

Exemple de script kickstart 2/9

```
reboot
text
install
url --url http://fr.archive.ubuntu.com/ubuntu
bootloader --location=mbr
zerombr yes
```

# Fichier complet kickstart

Exemple de script kickstart 3/9

```
clearpart --all --drives=sda  
ignoredisk --only-use=sda
```

# Fichier complet kickstart

Exemple de script kickstart 4/9

```
preseed partman-auto/init_automatically_partition select  
    biggest_free  
preseed partman/confirm_write_new_label boolean true  
preseed partman/choose_partition select finish  
preseed partman/confirm boolean true  
preseed partman/confirm_nooverwrite boolean true  
preseed partman-md/confirm_nooverwrite boolean true  
preseed partman-lvm/confirm_nooverwrite boolean true
```

# Fichier complet kickstart

Exemple de script kickstart 5/9

```
auth --useshadow --enablemd5
network --bootproto=dhcp --device=eth0
firewall --disabled
#Do not configure the X Window System
skipx
```

# Fichier complet kickstart

Exemple de script kickstart 6/9

```
%pre
parted -s /dev/sda mklabel gpt
parted -s /dev/sda mkpart ESP fat32 1MiB 513MiB
parted -s /dev/sda set 1 boot on
partprobe
%end
```

# Fichier complet kickstart

Exemple de script kickstart 7/9

```
auth --useshadow --enablemd5
network --bootproto=dhcp --device=eth0
firewall --disabled
skipx


```
%pre
parted -s /dev/sda mklabel gpt
parted -s /dev/sda mkpart ESP fat32 1MiB 513MiB
parted -s /dev/sda set 1 boot on
partprobe

```

%end
```

# Fichier complet kickstart

Exemple de script kickstart 8/9

```
%packages --resolvedeps
@ ubuntu-desktop
@ openssh-server
@ language-pack-fr
@ language-pack-gnome-fr-base language-pack-gnome-fr
@ ntpdate at chef grub-efi-amd64
```



# Fichier complet kickstart

Exemple de script kickstart 9/9

```
%post --interpreter=bash
ntpdate ntp.nom_mon_domaine.fr
.....
```

# Fichier complet preseed

## Exemple de script preseed 1/20

```
d-i finish-install/reboot_in_progress note
d-i debian-installer/exit/reboot boolean true
d-i ubiquity/summary note
ubiquity ubiquity/reboot boolean true
unattended-upgrades unattended-upgrades/enable_auto_updates
boolean true
```

# Fichier complet preseed

Exemple de script preseed 2/20

```
d-i mirror/country string manual
d-i mirror/http/hostname string archive.ubuntu.com
d-i mirror/http/directory string /ubuntu/
d-i mirror/http/proxy string
d-i user-setup/encrypt-home boolean false
```

# Fichier complet preseed

Exemple de script preseed 3/20

```
d-i apt-setup/use_mirror boolean true
d-i apt-setup/mirror/error select Change mirror
d-i apt-setup/multiverse boolean true
d-i apt-setup/restricted boolean true
d-i apt-setup/universe boolean true
d-i apt-setup/partner boolean true
```

# Fichier complet preseed

Exemple de script preseed 4/20

```
d-i netcfg/get_hostname string install-pc-fixe
d-i netcfg/get_domain string mon_domaine.fr
d-i netcfg/enable boolean true
d-i netcfg/choose_interface select auto
d-i netcfg/disable_dhcp boolean false
d-i netcfg/dhcp_timeout string 60
```

# Fichier complet preseed

Exemple de script preseed 5/20

```
d-i netcfg/get_hostname string unassigned-hostname
d-i netcfg/get_hostname seen false
d-i netcfg/get_domain string unassigned-domain
d-i netcfg/get_domain seen false
```

# Fichier complet preseed

Exemple de script preseed 6/20

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

# Fichier complet preseed

Exemple de script preseed 7/20

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



# Fichier complet preseed

Exemple de script preseed 8/20

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

# Fichier complet preseed

Exemple de script preseed 9/20

```
# Pour passer le message de confirmation Force UEFI  
  installation :)  
# cf https://askubuntu.com/questions/827545/ubuntu-server-16-from-iso-in-uefi-preseed-file  
d-i partman-efi/non_efi_system boolean true
```

# Fichier complet preseed

Exemple de script preseed 10/20

```
### Keyboard selection ###  
d-i keyboard-configuration/layoutcode string fr  
d-i keyboard-configuration/variantcode string  
# keymap=fr  
d-i keyboard-configuration/xkb-keymap select fr  
d-i debian-installer/keymap string fr-latin9  
d-i console-setup/modelcode string pc105  
d-i console-setup/layoutcode string fr
```

# Fichier complet preseed

Exemple de script preseed 11/20

```
# Locales
d-i debian-installer/fallbacklocale select fr_FR.UTF-8
d-i debian-installer/locale select fr_FR.UTF-8
d-i debian-installer/locale string fr_FR
d-i debian-installer/language string fr
d-i debian-installer/country string FR
d-i localechooser/supported-locales multiselect fr_FR
```

# Fichier complet preseed

Exemple de script preseed 12/20

```
d-i time/zone select Europe/Paris
d-i clock-setup/utc boolean true
d-i clock-setup/ntp boolean true
d-i clock-setup/ntp-server string nom_mon_domaine.fr
d-i console-setup/ask_detect boolean false
d-i console-setup/layoutcode string fr
```

# Fichier complet preseed

Exemple de script preseed 13/20

```
d-i oem-config/enable boolean true
d-i oem-config/remove boolean true
d-i oem-config/remove_extras boolean false

d-i oem-config/install-language-support boolean true
d-i ubiquity/only-show-installable-languages boolean true
```

# Fichier complet preseed

Exemple de script preseed 14/20

```
d-i passwd/root-login boolean false
d-i passwd/user-fullname string Service informatique
d-i passwd/username string info
d-i passwd/user-password-crypted password mot_de_passe_md5
```

# Fichier complet preseed

Exemple de script preseed 15/20

```
tasksel tasksel/first multiselect ubuntu-desktop
d-i pkgssel/include string build-essential
d-i pkgssel/language-packs multiselect fr, en
d-i pkgssel/upgrade select none
d-i pkgssel/update-policy select unattended-upgrades
d-i base-installer/install-recommends boolean true
```



# Fichier complet preseed

Exemple de script preseed 16/20

```
apt-cdrom-setup apt-setup/cdrom/set-first boolean false
d-i apt-setup/non-free boolean true
d-i apt-setup/contrib boolean true
d-i apt-setup/use_mirror boolean true
d-i apt-setup/services-select multiselect security, volatile
d-i apt-setup/security_host string security.ubuntu.com
```

# Fichier complet preseed

Exemple de script preseed 17/20

```
tasksel tasksel/first multiselect standard
d-i pkgsel/upgrade select safe-upgrade
popularity-contest popularity-contest/participate boolean
true
d-i pkgsel/include string openssh-server
```

# Fichier complet preseed

Exemple de script preseed 18/20

```
d-i grub-installer/only_debian boolean true
d-i grub-installer/with_other_os boolean true
d-i grub-installer/bootdev string /dev/sda
d-i preseed/late_command string \
    in-target apt-get -y purge resolvconf;
```

# Fichier complet preseed

Exemple de script preseed 19/20

```
d-i preseed/late_command string apt-install zsh; in-target  
chsh -s /bin/zsh  
d-i preseed/late_command string apt-install openssh-server  
wget bash curl
```

# Fichier complet preseed

Exemple de script preseed 20/20

```
d-i preseed/late_command string chroot /target sh -c "/usr/  
bin/curl -o /tmp/postinstall http://nom-complet-DNS-  
serveur-pxe/pxe/postinstall && /bin/bash -x /tmp/  
postinstall"
```

# Merci de votre attention