

Cobbler: Installation automatisée CentOS

Table of Contents

- [Cobbler: Installation automatisée CentOS](#)
- [Génération d'un fichier kickstart](#)
 - [Installer et démarrer des outils connexes](#)
 - [Installation automatisée avec une image CD](#)
 - [Monter l'image initiale.](#)
 - [Copier tous les fichiers du CD-ROM dans un emplacement spécifique.](#)
 - [Supprimer tous les fichiers TRANS.TBL](#)
 - [Supprimer le fichier repodata du fichier source.](#)
 - [Adapter et régénérer les repodata](#)
 - [Personnaliser le fichier kickstart](#)
 - [Créer un répertoire spécial pour stocker les fichiers kickstart](#)
 - [Créer un menu de démarrage](#)
 - [Fabrication de miroirs de CD-ROM](#)
 - [Utiliser ftp pour extraire l'image](#)
 - [Installation par lots de systèmes d'exploitation utilisant Cobbler](#)

L'installation du système d'exploitation est généralement une opération interactive, c'est-à-dire que l'installation a besoin de quelqu'un pour configurer la langue, l'heure, le réseau ou cliquer sur l'étape suivante, dans un processus d'interaction homme-machine. Pour installer un système d'exploitation, c'est bien, mais quand il faut 10 ou 100 cela devient fastidieux.

Lorsque la configuration de l'ensemble du système d'installation est la même, l'utilisation d'un fichier kickstart permet de personnaliser à l'avance la configuration appropriée en définissant l'heure, la langue, le réseau et les autres configurations système dans celui-ci.

Il existe trois méthodes principales non interactives pour installer un système d'exploitation.

- Créer une image iso d'installation automatique (installation sur un disque), principalement en apportant quelques modifications à l'image système d'origine et en intégrant le fichier de configuration personnalisé kickstart dans l'image système d'origine pour former une nouvelle image. Ensuite, les étapes d'installation automatique sont réalisées.
- L'installation pxe. Son principe est de déployer DHCP, TFTP et d'autres services associés sur un serveur, d'y placer le fichier miroir et le fichier de configuration, puis que les autres machines obtiennent l'IP via le service DHCP, communiquent avec le serveur via ip, extraient le fichier de configuration et image à exécuter localement.
- L'installation Cobbler, qui est la deuxième encapsulation de pxe, il faut également déployer manuellement dhcp et tftp, ainsi que d'autres services connexes. Dans Cobbler, les configurations, logiciels, services, etc. liés sont intégrés, ce qui facilite la gestion.

Parmi ces trois manières, la première et la troisième sont utilisées plus fréquemment. Le premier est souvent utilisé dans de petites structures où plusieurs serveurs sont installés, tandis que le troisième est généralement utilisé dans des centaines voire des milliers de machines. En particulier, cobbler

peut également être installé sur plusieurs segments de réseau. Cet article traite principalement de la fabrication de miroirs automatisés et de serveurs Cobbler.

Génération d'un fichier kickstart

Avant d'apprendre à créer des miroirs et les utiliser dans cobbler, il faut d'abord apprendre à créer un fichier de configuration à partir du processus d'interaction personne-machine lors de l'installation du système.

Pour générer des fichiers kickstart on peut utiliser facilement un modèle pour l'adapter puis l'utiliser directement.

Installer et démarrer des outils connexes

Sur une machine équipée d'une interface graphique, installer les outils de gestion graphique puis démarrer l'interface graphique

```
# yum install system-config-kickstart
# system-config-kickstart
```

Installation automatisée avec une image CD

Lorsqu'on utilise une installation à partir d'un disque ou d'une image iso, il faut indiquer cd - rom comme source d'installation.



Cette démarche peut également être utilisé pour préparer les images d'installation qui seront utilisées par cobbler

Monter l'image initiale.

Par exemple, monter l'image système d'origine dans un répertoire, tel que /mnt/.

Copier tous les fichiers du CD-ROM dans dans un emplacement spécifique.

Par exemple dans /var/ftp/pub/fulliso/. (Le but est de récupérer l'image finale par FTP ultérieurement. Bien sûr, on va d'abord installé et ouvert le service FTP.)

```
# yum -y install vsftpd
# systemctl start vsftpd
# cp -r /mnt/* /var/ftp/pub/fulliso/
```



Il faut également copier les fichiers cachés

```
cd /mnt/
```

```
cp .discinfo .treeinfo /var/ftp/pub/fulliso/
```

Supprimer tous les fichiers TRANS.TBL

Un fichier TRANS.TBL est l'option choisie par RedHat, pour assurer la correspondances des fichiers du répertoire avec des noms MS-DOS.

```
# cd /var/ftp/pub/fulliso/
# find -name TRANS.TBL -exec rm {} \;
```

Supprimer le fichier repodata du fichier source.

Étant donné qu'on veut personnaliser l'image du CD-ROM, les repodata d'origine peuvent devenir inappropriés en raison du changement du chemin ou de la suppression de contenu. Il faut supprimer et régénérer les repodata en fonction de la nouvelle situation.

```
[root@29 /var/ftp/pub/fulliso]# rm -rf repodata/
```

Adapter et régénérer les repodata

Créer un nouveau répertoire

```
# mkdir repodata
```

Copier le contenu du répertoire repodata de l'entrepôt d'origine dans le fichier repodata de la nouvelle image iso.



Seul le fichier à la fin du XML est copié. Le fichier se terminant par x86_64-comps.xml est celui qui doit être copié.

```
# cp /mnt/repodata/d8***-x86_64-comps.xml repodata/
```

Régénérer les repodata pertinents en fonction de la nouvelle situation

```
# createrepo -g  
repodata/d87379a47bc2060f833000b9cef7f9670195fe197271d37fce5791e669265e8b-  
c7-x86_64-comps.xml ./
```

Personnaliser le fichier kickstart

Le fichier kickstart ci-dessus est un modèle de base et l'environnement d'installation réel est modifié deux fois, pour produire:

- une version mini simplifiée ks7-mini.cfg
- une version complète du bureau ks7-desktop.cfg

Version minimale

```
#platform=x86, AMD64, or Intel EM64T  
#version=DEVEL  
# Install OS instead of upgrade  
install          #Choisir Installation or Update  
# Dispositions de clavier  
keyboard 'us'  
# Mot de passe root # Le mot de passe utilisateur root peut être défini à  
l'aide de la commande openssl (voir Baidu pour plus d'informations).  
# Ou on peut directement définir le mot de passe lors de la génération  
graphique du fichier de modèle ks ci-dessus.  
rootpw --iscrypted $1$GKfMMD9.$Cgag5pWRZcCL052j1Hw4P1  
# Utiliser l'installation cdrom  
cdrom            #Une option importante est de sélectionner entre le CD-ROM ou le  
montage cobbler  
# Tout comme le fichier ks, la déclaration est l'adresse IP de  
l'installation réseau et des services associés.  
# Langue du système  
lang en_US  
# Configuration du pare-feu  
firewall --disabled  
# System authorization information  
auth --useshadow --passalgo=sha512  
# Utiliser l'installation en mode texte  
text  
# Configuration de SELinux  
selinux --disabled  
# Ne pas configurer le système X Window  
skipx  
  
# Network information
```

```
network --bootproto=dhcp --device=ens33
# Reboot after installation
reboot
# System timezone
timezone Asia/Shanghai
# System bootloader configuration
bootloader --location=mbr
# Effacer le Master Boot Record
zerombr
# Effacer la table de Partition
clearpart --all --initlabel
# Partitionnement des disques
# La partition ne peut pas être trop divisée. Elle doit être combinée avec
la situation réelle du disque dur, sinon l'installation sera incorrecte.
# Les volumes lvm peuvent être définis directement lors du partitionnement.
part / --fstype="xfs" --size=20480
part /boot --fstype="xfs" --size=500
part swap --fstype="swap" --size=4096

# Sélection des packages yum pertinents pour l'installation de la mini-
version

%packages
@^minimal
@base
-abrt-addon-ccpp
-abrt-addon-python
-abrt-cli
-abrt-console-notification
-bash-completion
-blktrace
-bridge-utils
-bzip2
-chrony
-cryptsetup
-dmraid
-dosfstools
-ethtool
-fprintd-pam
-gnupg2
-hunspell
-hunspell-en
-kmod-kvdo
-kpatch
-ledmon
-libaio
-libreport-plugin-mailx
-libstoragegmt
-lvm2
-man-pages
-man-pages-overrides
```

```
-mdadm
-mlocate
-mtr
-nano
-ntpdate
-pinfo
-plymouth
-pm-utils
-rdate
-rfkill
-rng-tools
-rsync
-scl-utils
-setuptools
-smartmontools
-sos
-sssd-client
-strace
-sysstat
-systemtap-runtime
-tcpdump
-tcsh
-teamd
-time
-unzip
-usbutils
-vdo
-vim-enhanced
-virt-what
-wget
-which
-words
-xfsdump
-xz
-yum-langpacks
-yum-utils
-zip

%end
```

Version complète avec le bureau

```
#platform=x86, AMD64, or Intel EM64T
#version=DEVEL
# Install OS instead of upgrade
install
# Keyboard layouts
keyboard 'us'
# Root password
```

```
rootpw --iscrypted $1$GKfMMD9.$Cgag5pWRZcCL052j1Hw4P1
# Use cdrom installation
cdrom
# System language
lang en_US
# Firewall configuration
firewall --disabled
# System authorization information
auth --useshadow --passalgo=sha512
# Use text mode install
text
# SELinux configuration
selinux --disabled
# Do not configure the X Window System
skipx

# Network information
network --bootproto=dhcp
# Reboot after installation
reboot
# System timezone
timezone Asia/Shanghai
# System bootloader configuration
bootloader --location=mbr
# Clear the Master Boot Record
zerombr
# Partition clearing information
clearpart --all --initlabel
# Disk partitioning information
part / --fstype="xfs" --size=20480
part /boot --fstype="xfs" --size=500
part swap --fstype="swap" --size=4096

%packages
@^graphical-server-environment
@base
@core
@desktop-debugging
@dial-up
@fonts
@gnome-desktop
@guest-agents
@guest-desktop-agents
@ha
@hardware-monitoring
@input-methods
@internet-browser
@java-platform
@kde-desktop
@large-systems
@load-balancer
```

```
@multimedia
@performance
@print-client
@virtualization-client
@virtualization-hypervisor
@virtualization-tools
@x11
chrony
kexec-tools

%end
```

Créer un répertoire spécial pour stocker les fichiers kickstart

```
# mkdir kickstart
```

Placer les fichiers KS dans le répertoire kickstart.

```
# mv ks7-desktop.cfg ks7-mini.cfg kickstart/
```

Créer un menu de démarrage

```
# cd isolinux/
# vim isolinux.cfg
```

Le contenu d'origine est inchangé. Deux étiquettes sont ajoutées à celle d'origine, qui sera affichée au démarrage. Ajouter des balises de bureau et des mini balises et spécifier l'emplacement de stockage des fichiers ks.

```
...
...
...
label desktop
    menu label Install ^desktop system
    kernel vmlinuz
    append initrd=initrd.img ks=cdrom:/kickstart/ks7-desktop.cfg
label mini
    menu label Install an m^ini system
    kernel vmlinuz
    append initrd=initrd.img ks=cdrom:/kickstart/ks7-mini.cfg
label linux
    menu label ^Install CentOS 7
    kernel vmlinuz
    append initrd=initrd.img inst.stage2=hd:LABEL=CentOS\x207\x20x86_64 quiet
```



```
label check
  menu label Test this ^media & install CentOS 7
  menu default
  kernel vmlinuz
  append initrd=initrd.img inst.stage2=hd:LABEL=CentOS\x207\x20x86_64
rd.live.check quiet

menu separator # insert an empty line

# utilities submenu
menu begin ^Troubleshooting
  menu title Troubleshooting

label vesa
  menu indent count 5
  menu label Install CentOS 7 in ^basic graphics mode
  text help
    Try this option out if you're having trouble installing
    CentOS 7.
  endtext
  kernel vmlinuz
  append initrd=initrd.img inst.stage2=hd:LABEL=CentOS\x207\x20x86_64
xdriver=vesa nomodeset quiet

label rescue
  menu indent count 5
  menu label ^Rescue a CentOS system
  text help
    If the system will not boot, this lets you access files
    and edit config files to try to get it booting again.
  endtext
  kernel vmlinuz
  append initrd=initrd.img inst.stage2=hd:LABEL=CentOS\x207\x20x86_64 rescue
quiet

label memtest
  menu label Run a ^memory test
  text help
    If your system is having issues, a problem with your
    system's memory may be the cause. Use this utility to
    see if the memory is working correctly.
  endtext
  kernel memtest

menu separator # insert an empty line

label local
  menu label Boot from ^local drive
  localboot 0xffff

menu separator # insert an empty line
```

```
menu separator # insert an empty line
```

```
label returntomain
  menu label Return to ^main menu
  menu exit
```

Fabrication de miroirs de CD-ROM

```
# mkisofs -R -J -T -v --no-emul-boot --boot-load-size 4 --boot-info-table -V
"CentOS 7.4 autoinstall" -b isolinux/isolinux.bin -c isolinux/boot.cat -o
/var/ftp/pub/auto_centos7.4.iso /var/ftp/pub/fulliso/
```

Attention: Cette commande est très longue.

- V "CentOS 7.4 autoinstall" fait référence aux détails du miroir, qui peuvent être modifiés pour fonctionner avec le même nom que la version générale
- -o/var/ftp/pub/auto_centos7.4.iso fait référence à l'emplacement de stockage de votre propre miroir. Il est conseillé de le mettre dans le répertoire de pub, puis d'utiliser ftp pour le télécharger.
- /var/ftp/pub/fulliso/ est l'emplacement du fichier source, c'est-à-dire le fichier que l'on veut mettre en miroir en tant qu'iso.



Il n'est pas nécessaire de modifier d'autres contenus, il suffit de les copier directement.

Utiliser ftp pour extraire l'image

Utiliser ftp pour extraire l'image dans Windows, puis la copier sur le disque. Lors de l'installation, choisir d'installer la version de bureau ou la version la plus petite, le reste est complètement automatisé.

Installation par lots de systèmes d'exploitation utilisant Cobbler

Quand il y a plus de machines, l'automatisation des disques est encore très fastidieuse. L'approche décrite ici permet d'utiliser le service cobbler sur un serveur pour que les autres ordinateurs puissent être installés automatiquement via pxe.

lorsqu'on utilise Cobbler, la source d'installation doit être HTTP et il faut renseigner l'adresse IP et l'emplacement du répertoire correspondants au miroir devant être utilisé, tels que http://10.1.18.29/cobbler/ks_mirror/centos7.4.

```
#platform=x86, AMD64, or Intel EM64T
```

```
#version=DEVEL
# Install OS instead of upgrade
install
# Keyboard layouts
keyboard 'us'
# Root password
rootpw --iscrypted $1$GKfMMD9.$Cgag5pWRZcCL052j1Hw4P1
# Use network installation
url --url="http://10.1.18.29/cobbler/ks_mirror/centos7.4"
# System language
lang en_US
# Firewall configuration
firewall --disabled
# System authorization information
auth --useshadow --passalgo=sha512
# Use text mode install
text
# SELinux configuration
selinux --disabled
# Do not configure the X Window System
skipx

# Network information
network --bootproto=dhcp --device=ens33
# Reboot after installation
reboot
# System timezone
timezone Asia/Shanghai
# System bootloader configuration
bootloader --location=mbr
# Clear the Master Boot Record
zerombr
# Partition clearing information
clearpart --all --initlabel
# Disk partitioning information
part / --fstype="xfs" --size=40960
part /boot --fstype="xfs" --size=500
part /usr --fstype="xfs" --size=51200
part swap --fstype="swap" --size=4096

%packages
@^minimal
@base
-abrt-addon-ccpp
-abrt-addon-python
-abrt-cli
-abrt-console-notification
-bash-completion
-blktrace
-bridge-utils
-bzip2
```

- chrony
- cryptsetup
- dmraid
- dosfstools
- ethtool
- fprintd-pam
- gnupg2
- hunspell
- hunspell-en
- kmod-kvdo
- kpatch
- ledmon
- libaio
- libreport-plugin-mailx
- libstoragemgmt
- lvm2
- man-pages
- man-pages-overrides
- mdadm
- mlocate
- mtr
- nano
- ntpdate
- pinfo
- plymouth
- pm-utils
- rdate
- rfkill
- rng-tools
- rsync
- scl-utils
- setuptools
- smartmontools
- sos
- sssd-client
- strace
- sysstat
- systemtap-runtime
- tcpdump
- tcsh
- teamd
- time
- unzip
- usbutils
- vdo
- vim-enhanced
- virt-what
- wget
- which
- words
- xfsdump

```
-xz
-yum-langpacks
-yum-utils
-zip

%end
```

From:

<http://www.ouarte.garden/> - **dwndoc**

Permanent link:

<http://www.ouarte.garden/doku.php?id=centos:cobbler-unattended-install>

Last update: **2025/02/19 10:59**

