



NOTES DE COURS

Commandes Linux de base

Table des matières

1	Afficher de l'aide.....	4
1.1	man	4
1.2	apropos	4
2	Monter et démonter un périphérique	5
2.1	mount.....	5
2.2	umount	5
3	Lister le contenu d'un répertoire (ls)	6
4	Se déplacer dans l'arborescence (cd).....	8
5	Afficher le nom du répertoire de travail (pwd)	8
6	Afficher l'arborescence (tree).....	9
7	Créer des répertoires (mkdir).....	10
8	Créer un fichier (touch)	11
9	Créer un raccourci (ln)	11
10	Copier des fichiers et des répertoires (cp)	13
10.1	Copier plusieurs fichiers à la fois.....	14
10.2	Copier en préservant les attributs	15
10.3	Copier des répertoires.....	15
11	Renommer et déplacer des fichiers (mv)	16
11.1	Renommer un fichier	16
11.2	Déplacer et renommer des fichiers	17
11.3	Déplacer et renommer des répertoires.....	18
12	Supprimer des fichiers ou des répertoires (rm)	19
12.1	Supprimer des fichiers	19
12.2	Supprimer des répertoires (rm).....	20
12.3	Supprimer des répertoires vides (rmdir).....	21
13	Rechercher des fichiers et des répertoires (find)	22
14	Consulter le contenu d'un fichier	23
14.1	cat.....	23
14.2	head	24
14.3	more	25
14.4	less	26
14.5	tail	27

15	Rechercher un mot dans un fichier (grep).....	28
16	Compression et décompression	29
16.1	gzip.....	29
16.2	gunzip	30
16.3	compress.....	30
16.4	uncompress	30
16.5	tar.....	31
17	cal	32
18	kill.....	33
19	passwd.....	33
20	uptime.....	34
21	which	34
22	who	35
23	clear.....	35
24	Arrêt et redémarrage	35
24.1	shutdown	35
24.2	reboot.....	35
24.3	init	35
25	Éditeurs de fichiers.....	36
25.1	vim	36
25.2	nano	36

1 Afficher de l'aide

1.1 man

La commande *man* recherche et affiche la page de manuel associée à un *sujet*. La recherche s'effectue de la section 1 à la section 10 et *man* s'arrête à la première page trouvée. Il est possible de préciser à quelle *section* se rapporte le *sujet* recherché.

Pour afficher de l'aide sur la commande *reboot* :

```
[root@localhost ~]# man reboot
```

1.2 apropos

La commande *apropos* recherche et affiche de l'aide sommaire.

```
[root@localhost ~]# apropos reboot
grub2-reboot (8)      - Set the default boot menu entry for
the next boot only.
halt (8)              - Halt, power-off or reboot the machine
poweroff (8)          - Halt, power-off or reboot the machine
reboot (8)            - Halt, power-off or reboot the machine
shutdown (8)          - Halt, power-off or reboot the machine
systemd-reboot.service (8) - System shutdown logic
[root@localhost boot]#
```

2 Monter et démonter un périphérique

2.1 mount

La commande *mount* permet de monter un périphérique (le rendre accessible) dans un point de montage.

Créer un répertoire qui va servir de point de montage :

```
[root@localhost ~]# mkdir /montage
```

Monter le périphérique dans le point de montage :

```
[root@localhost ~]# mount /dev/cdrom /montage/  
mount: /dev/sr0 is write-protected, mounting read-only
```

La commande *mount* seul permet d'afficher la liste des périphériques qui sont montés.

```
[root@localhost ~]# mount  
.  
.  
/dev/mapper/centos-root on / type xfs (rw,relatime,attr2,inode64,noquota)  
.  
.  
/dev/sda1 on /boot type xfs (rw,relatime,attr2,inode64,noquota)  
.  
.  
.  
/dev/sr0 on /montage type iso9660 (ro,relatime)  
[root@localhost ~]#
```

2.2 umount

La commande *umount* permet de démonter un périphérique du point de montage. Le périphérique devient inaccessible.

```
[root@localhost ~]# umount /dev/cdrom
```

3 Lister le contenu d'un répertoire (ls)

La commande `ls` liste les fichiers dans le répertoire courant. Si un ou plusieurs *noms* sont précisés, `ls` affiche les fichiers qui appartiennent aux répertoires nommés ou dont le nom correspond à l'un de ceux spécifiés (les *noms* peuvent contenir des méta-caractères).

Pour afficher le contenu du répertoire courant :

```
[root@localhost ~]# ls  
anaconda-ks.cfg
```

Pour afficher le contenu de la racine du disque :

```
[root@localhost ~]# ls /  
bin    dev    home  lib64  mnt    proc   run    srv    tmp    var  
boot   etc    lib    media  opt    root   sbin   sys    usr
```

Utiliser la commande `ls(list)` avec l'option `-l` (long).

```
[root@localhost ~]# ls -l  
total 4  
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
```

Pour afficher le contenu de la racine du disque :

```
[root@localhost ~]# ls -l /  
total 20  
lrwxrwxrwx. 1 root root 7 May 18 11:01 bin -> usr/bin  
dr-xr-xr-x. 5 root root 4096 Jun 1 19:49 boot  
drwxr-xr-x. 19 root root 3080 Jun 6 21:51 dev  
drwxr-xr-x. 77 root root 8192 Jun 6 22:10 etc  
drwxr-xr-x. 2 root root 6 Apr 11 2018 home  
lrwxrwxrwx. 1 root root 7 May 18 11:01 lib -> usr/lib  
lrwxrwxrwx. 1 root root 9 May 18 11:01 lib64 -> usr/lib64  
drwxr-xr-x. 2 root root 6 Apr 11 2018 media  
drwxr-xr-x. 2 root root 6 Apr 11 2018 mnt  
drwxr-xr-x. 2 root root 6 Apr 11 2018 opt  
dr-xr-xr-x 173 root root 0 Jun 6 21:50 proc  
dr-xr-x---. 2 root root 157 Jun 6 22:31 root  
drwxr-xr-x. 26 root root 740 Jun 6 22:28 run  
lrwxrwxrwx. 1 root root 8 May 18 11:01 sbin -> usr/sbin  
drwxr-xr-x. 2 root root 6 Apr 11 2018 srv  
dr-xr-xr-x. 13 root root 0 Jun 6 21:50 sys  
drwxrwxrwt. 9 root root 249 Jun 6 22:28 tmp  
drwxr-xr-x. 13 root root 155 May 18 11:01 usr  
drwxr-xr-x. 22 root root 4096 Jun 5 00:15 var
```

Utiliser l'option **-a** (all).

```
[root@localhost ~]# ls -la
total 32
dr-xr-x---.  2 root root  157 Jun  6 22:31 .
dr-xr-xr-x. 17 root root  224 May 18 11:05 ..
-rw-----.  1 root root 1244 May 18 11:06 anaconda-ks.cfg
-rw-----.  1 root root 1490 Jun  6 13:16 .bash_history
-rw-r--r--.  1 root root   18 Dec 28  2013 .bash_logout
-rw-r--r--.  1 root root  176 Dec 28  2013 .bash_profile
-rw-r--r--.  1 root root  176 Dec 28  2013 .bashrc
-rw-r--r--.  1 root root  100 Dec 28  2013 .cshrc
-rw-----.  1 root root  104 Jun  6 13:16 .mysql_history
-rw-r--r--.  1 root root  129 Dec 28  2013 .tcshrc
```

4 Se déplacer dans l'arborescence (cd)

La commande *cd* permet de se déplacer dans l'arborescence du système de fichiers utiliser la commande **cd** (change directory).

Se déplacer le répertoire /tmp :

```
[root@localhost ~]# cd /tmp
```

Afficher le contenu du répertoire courant (/tmp) :

```
[root@localhost tmp]# ls  
f1      f4  
f100    java  
f2      java1  
f3      systemd-private-2164956288e54a058377ab0328d2033f-  
chronyd.service-avkl72  
[root@localhost tmp]#
```

(**cd** sans argument permet de rejoindre le répertoire personnel).

```
[root@localhost tmp]# cd  
[root@localhost ~]#
```

5 Afficher le nom du répertoire de travail (pwd)

La commande *pwd* permet d'afficher le nom du répertoire courant (print working directory).

Afficher le nom du répertoire courant :

```
[root@localhost ~]# pwd  
/root
```

Se déplacer le répertoire /boot :

```
[root@localhost ~]# cd /boot
```

Afficher le nom du répertoire courant :

```
[root@localhost boot]# pwd  
/boot
```


6 Afficher l'arborescence (tree)

Cette commande permet d'afficher l'arborescence d'un répertoire.

Cette commande n'est pas installée par défaut. Pour l'installer :

```
[root@localhost ~]# yum install tree
```

Créer l'arborescence :

```
[root@localhost ~]# mkdir -p /orabec/ti/doc
```

```
[root@localhost ~]# mkdir -p /orabec/finance/2020
```

```
[root@localhost ~]# touch /orabec/ti/doc/liste_emp.txt
```

Pour afficher l'arborescence du répertoire /orabec :

```
[root@localhost ~]# tree /orabec
/orabec
├── finance
│   └── 2020
└── ti
    └── doc
        └── liste_emp.txt

4 directories, 1 file
[root@localhost ~]#
```

7 Créer des répertoires (mkdir)

La commande *mkdir* permet de créer un ou plusieurs *répertoires*. Vous devez avoir les droits d'écriture dans le répertoire père pour pouvoir créer un répertoire.

Créer les répertoires /root/java1, /root/java2, /root/java3 :

```
[root@localhost ~]# mkdir java1 java2 java3
```

Faire ls pour vérifier :

```
[root@localhost ~]# ls
doc  images  java1  java2  java3  utilisateurs
```

Créer le répertoire /root/linux/centos :

```
[root@localhost ~]# mkdir /root/linux/centos

mkdir: cannot create directory '/root/linux/centos': No such
file or directory
```

Impossible de créer le répertoire *centos* car le répertoire **parent** *linux* n'existe pas. Dans ce cas, on peut utiliser l'option *p* pour spécifier qu'il faut créer le répertoire parent s'il n'existe pas :

```
[root@localhost ~]# mkdir -p /root/linux/centos
```

8 Créer un fichier (touch)

La commande *touch* permet de créer un fichier vide ou de mettre le timestamp d'un fichier existant.

```
[root@localhost ~]# touch f1
```

```
[root@localhost ~]# touch f2 f3
```

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
-rw-r--r-- 1 root root    0 Jun  6 22:53 f1
-rw-r--r-- 1 root root    0 Jun  6 22:53 f2
-rw-r--r-- 1 root root    0 Jun  6 22:53 f3
```

9 Créer un raccourci (ln)

La commande *ln* permet de créer des liens pour des fichiers ou répertoires, permettant ainsi d'y accéder sous différents noms.

```
[root@localhost ~]# ln -s anaconda-ks.cfg toto
```

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
-rw-r--r-- 1 root root    0 Jun  6 22:58 f1
-rw-r--r-- 1 root root    0 Jun  6 22:53 f2
-rw-r--r-- 1 root root    0 Jun  6 22:53 f3
-rw-r--r-- 1 root root    0 Jun  6 22:55 f4
-rw-r--r-- 1 root root    0 Jun  6 22:58 f5
lrwxrwxrwx 1 root root   15 Jun  6 23:06 toto -> anaconda-ks.cfg
```


10 Copier des fichiers et des répertoires (cp)

La commande `cp` permet de copier un ou plusieurs fichiers ou répertoires. Si la destination est un fichier qui existe déjà, alors il sera écrasé.

Options	Description
-i	Demande une confirmation avant d'écraser un fichier existant.
-p	Garde la date de dernière modification et les droits lors de la copie.
-r	Effectue une copie récursive dans un autre <i>répertoire</i> .

Créer une copie du fichier *f3* et la nommer *f4* :

```
[root@localhost ~]# cp f3 f4
```

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
-rw-r--r-- 1 root root    0 Jun  6 22:53 f1
-rw-r--r-- 1 root root    0 Jun  6 22:53 f2
-rw-r--r-- 1 root root    0 Jun  6 22:53 f3
-rw-r--r-- 1 root root    0 Jun  6 22:55 f4
```

Utiliser l'option **-i** pour le mode interactif (confirmer l'écrasement)

```
[root@localhost ~]# cp -i f3 f1
cp: overwrite 'f1'? y
```

10.1 Copier plusieurs fichiers à la fois

```
[root@localhost ~]# cp f1 /tmp/
```

```
[root@localhost ~]# cp f2 /tmp/
```

```
[root@localhost ~]# ls -l /tmp/f*  
-rw-r--r-- 1 root root 0 Jun  6 22:59 /tmp/f1  
-rw-r--r-- 1 root root 0 Jun  6 22:59 /tmp/f2
```

```
[root@localhost ~]# cp f* /tmp/  
cp: overwrite '/tmp/f1'? y  
cp: overwrite '/tmp/f2'? y
```

```
[root@localhost ~]# ls -l  
total 4  
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg  
-rw-r--r-- 1 root root    0 Jun  6 22:58 f1  
-rw-r--r-- 1 root root    0 Jun  6 22:53 f2  
-rw-r--r-- 1 root root    0 Jun  6 22:53 f3  
-rw-r--r-- 1 root root    0 Jun  6 22:55 f4
```

10.2 Copier en préservant les attributs

L'option -p préserve les attributs (propriétaire, groupe date, permissions).

Avant la copie :

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
-rw-r--r-- 1 root root 0 Jun 6 22:58 f1
-rw-r--r-- 1 root root 0 Jun 6 22:53 f2
-rw-r--r-- 1 root root 0 Jun 6 22:53 f3
-rw-r--r-- 1 root root 0 Jun 6 22:55 f4
```

```
[root@localhost ~]# cp -p f1 f5
```

Après la copie:

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
-rw-r--r-- 1 root root 0 Jun 6 22:58 f1
-rw-r--r-- 1 root root 0 Jun 6 22:53 f2
-rw-r--r-- 1 root root 0 Jun 6 22:53 f3
-rw-r--r-- 1 root root 0 Jun 6 22:55 f4
-rw-r--r-- 1 root root 0 Jun 6 22:58 f5
```

10.3 Copier des répertoires

Copier le répertoire /boot et son contenu dans le répertoire /tmp :

```
[root@localhost boot]# cp -r /boot/ /tmp/
```

Vérifier que la copie a bien été complétée :

```
[root@localhost boot]# ls /tmp
boot  f4
f1    java
f100  java1
f2     systemd-private-2164956288e54a058377ab0328d2033f-
chronyd.service-avkl72
f3
/
```

11 Renommer et déplacer des fichiers (mv)

La commande *mv* permet de déplacer ou de renommer des fichiers et des répertoires.

11.1 Renommer un fichier

```
[root@localhost ~]# mv f5 fichier5
```

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
-rw-r--r-- 1 root root    0 Jun  6 22:58 f1
-rw-r--r-- 1 root root    0 Jun  6 22:53 f2
-rw-r--r-- 1 root root    0 Jun  6 22:53 f3
-rw-r--r-- 1 root root    0 Jun  6 22:55 f4
-rw-r--r-- 1 root root    0 Jun  6 22:58 fichier5
lrwxrwxrwx 1 root root   15 Jun  6 23:06 toto -> anaconda-
ks.cfg
```


11.2 Déplacer et renommer des fichiers

```
[root@localhost ~]# mv f3 /tmp/fichier3

[root@localhost ~]# ls -l /tmp/f*
-rw-r--r-- 1 root root 0 Jun  6 23:00 /tmp/f1
-rw-r--r-- 1 root root 0 Jun  6 23:00 /tmp/f2
-rw-r--r-- 1 root root 0 Jun  6 23:00 /tmp/f3
-rw-r--r-- 1 root root 0 Jun  6 23:00 /tmp/f4
-rw-r--r-- 1 root root 0 Jun  6 22:53 /tmp/fichier3
```

11.3 Déplacer et renommer des répertoires

Créer deux répertoires *java1* et *java2* :

```
[root@localhost ~]# mkdir java1 java2
```

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
-rw-r--r-- 1 root root    0 Jun  6 22:58 f1
-rw-r--r-- 1 root root    0 Jun  6 22:53 f2
-rw-r--r-- 1 root root    0 Jun  6 22:55 f4
-rw-r--r-- 1 root root    0 Jun  6 22:58 fichier5
drwxr-xr-x 2 root root    6 Jun  6 23:12 java1
drwxr-xr-x 2 root root    6 Jun  6 23:12 java2
lrwxrwxrwx 1 root root   15 Jun  6 23:06 toto -> anaconda-
ks.cfg
```

Renommer le répertoire *java1* en *java10* :

```
[root@localhost ~]# mv java1 java10
```

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
-rw-r--r-- 1 root root    0 Jun  6 22:58 f1
-rw-r--r-- 1 root root    0 Jun  6 22:53 f2
-rw-r--r-- 1 root root    0 Jun  6 22:55 f4
-rw-r--r-- 1 root root    0 Jun  6 22:58 fichier5
drwxr-xr-x 2 root root    6 Jun  6 23:12 java10
drwxr-xr-x 2 root root    6 Jun  6 23:12 java2
lrwxrwxrwx 1 root root   15 Jun  6 23:06 toto -> anaconda-
ks.cfg
```

Renommer et déplacer dans */tmp* le répertoire *java2* en *java20* :

```
[root@localhost ~]# mv java2 /tmp/java20
```

```
[root@localhost ~]# ls /tmp/
f1  fichier3
f2  java20
f3  systemd-private-46fabb15656f4f9fa195620c2e2cab59-
chronyd.service-7pUd2j
f4  systemd-private-46fabb15656f4f9fa195620c2e2cab59-
httpd.service-6EAMHc
```

12 Supprimer des fichiers ou des répertoires (rm)

La commande *rm* permet de supprimer un ou plusieurs *fichiers/répertoires*. Vous devez avoir les droits d'écriture dans le répertoire sans pour autant avoir des droits sur les *fichiers* eux-mêmes. Si vous n'avez pas de droits d'écriture sur les *fichiers*, un message vous demandera si vous confirmez l'opération ou non.

12.1 Supprimer des fichiers

Pour supprimer un fichier :

```
[root@localhost ~]# rm f3  
  
rm: remove regular empty file `f3'? y  
[root@localhost ~]#
```

Pour supprimer un fichier sans confirmation (option f):

```
[root@localhost ~]# rm -f f4
```

Pour supprimer un fichier avec confirmation (option i):

```
[root@localhost ~]# rm -i f5  
rm: remove regular empty file `f5'? y
```

Pour supprimer plusieurs fichiers sans confirmation :

```
[root@localhost ~]# rm -f /tmp/*.*
```

12.2 Supprimer des répertoires (rm)

Utiliser la commande **rm** avec l'option **-r** (récursive).

Créer le répertoire /root/java/ :

```
[root@localhost ~]# mkdir java/
```

Créer un fichier dans le répertoire /root/java/ :

```
[root@localhost ~]# touch java/f1
```

Il n'est possible de supprimer un répertoire avec la commande **rm** sans utiliser l'option **r** :

```
[root@localhost ~]# rm java/  
rm: cannot remove 'java/': Is a directory
```

Supprimer le répertoire /java/ avec la commande **rm** en utilisant l'option **r** :

```
[root@localhost ~]# rm -r /java/
```

On peut utiliser l'option **f** (force) pour éviter de confirmer la suppression de chaque fichier :

```
[root@localhost ~]# rm -rf /java/
```

12.3 Supprimer des répertoires vides (rmdir)

La commande *rmdir* permet de supprimer un *répertoire vide*. Si le répertoire n'est pas vide alors utiliser la commande *rm -r* peut être utilisé.

Créer le répertoire `/root/java/` :

```
[root@localhost ~]# mkdir java/
```

Créer un fichier dans le répertoire `/root/java/` :

```
[root@localhost ~]# touch java/f1
```

Impossible de supprimer le répertoire `/java/` en utilisant la commande *rmdir* car le répertoire n'est pas vide :

```
[root@localhost ~]# rmdir java/
rmdir: failed to remove `java/': Directory not empty
```

Supprimer le fichier *f1* :

```
[root@localhost ~]# rm java/f1
rm: remove regular empty file `java/f1'? y
```

Il est possible à présent de supprimer le répertoire `/java/` en utilisant la commande *rmdir* car le répertoire est vide :

```
[root@localhost ~]# rmdir java/
```

13 Rechercher des fichiers et des répertoires (find)

La commande *find* permet de recherche en fonction d'une ou plusieurs *conditions* des noms de fichiers et y effectue des opérations.

Pour recherche le fichier *passwd* à partir de la racine du disque :

```
[root@localhost ~]# find / -name passwd  
/etc/pam.d/passwd  
/etc/passwd  
/usr/bin/passwd
```

14 Consulter le contenu d'un fichier

14.1 cat

La commande `cat` permet d'afficher un ou plusieurs *fichiers* sur le canal de sortie standard.

```
[root@localhost ~]# cat anaconda-ks.cfg
#version=DEVEL
# System authorization information
auth --enableshadow --passalgo=sha512
# Use CDROM installation media
cdrom
# Use graphical install
graphical
# Run the Setup Agent on first boot
firstboot --enable
ignoredisk --only-use=sda
# Keyboard layouts
keyboard --vckeymap=us --xlayouts='us'
# System language
lang en_US.UTF-8

# Network information
network --bootproto=dhcp --device=ens32 --ipv6=auto --activate
network --hostname=localhost.localdomain

# Root password
rootpw --iscrypted
$6$cbauomSFBvt.NLfV$YgD5Ag/DJcKVnQoMYLXn9U2SOhA/oINXE/uU4O6r8qwpEur1F.PpIQc9Gwynm
LlpAJ/F0YncSaQtjpg.L2nK60
# System services
services --enabled="chronyd"
# System timezone
timezone America/New_York --isUtc
# System bootloader configuration
bootloader --append=" crashkernel=auto" --location=mbr --boot-drive=sda
autopart --type=lvm
# Partition clearing information
clearpart --none --initlabel

%packages
@^minimal
@core
chrony
kexec-tools

%end

%addon com_redhat_kdump --enable --reserve-mb='auto'

%end

%anaconda
pwpolicy root --minlen=6 --minquality=1 --notstrict --nochanges --notempty
pwpolicy user --minlen=6 --minquality=1 --notstrict --nochanges --emptyok
pwpolicy luks --minlen=6 --minquality=1 --notstrict --nochanges --notempty
%end
[root@localhost ~]#
```

14.2 head

La commande *head* permet d'afficher les premières lignes (par défaut les 10 premières) d'un ou plusieurs *fichiers* et/ou du canal d'entrée standard.

Afficher les 10 premières lignes du fichier *anaconda-ks.cfg* en utilisant la commande *head* :

```
[root@localhost ~]# head anaconda-ks.cfg
#version=DEVEL
# System authorization information
auth --enableshadow --passalgo=sha512
# Use CDROM installation media
cdrom
# Use graphical install
graphical
# Run the Setup Agent on first boot
firstboot --enable
ignoredisk --only-use=sda
[root@localhost ~]#
```


14.3 more

La commande *more* permet d'afficher les *fichiers* dans un terminal page à page. Certaines touches vont vous permettre de naviguer.

<i>Touches</i>	<i>Description</i>
RETURN	Passe à la ligne suivante.
ESPACE	Passe à la page suivante.
h	Affiche un écran d'aide.
q	Quitte.
/	Recherche dans un fichier.
:n	Passe au fichier suivant.

Afficher le contenu du fichier *anaconda-ks.cfg* en utilisant la commande *more* :

```
[root@localhost ~]# more anaconda-ks.cfg
#version=DEVEL
# System authorization information
auth --enableshadow --passalgo=sha512
# Use CDROM installation media
cdrom
# Use graphical install
graphical
# Run the Setup Agent on first boot
firstboot --enable
ignoredisk --only-use=sda
# Keyboard layouts
keyboard --vckeymap=us --xlayouts='us'
# System language
lang en_US.UTF-8

# Network information
network --bootproto=dhcp --device=ens32 --ipv6=auto --activate
network --hostname=localhost.localdomain

# Root password
rootpw --iscrypted
$6$cbauomSFBvt.NLfV$YgD5Ag/DJcKVnQoMYLXn9U2SOhA/oINXE/uU4O6r8
qwpEur1F.PpIQc9GwynmLlpAJ/F0YncSaQtjpg.L2nK60
# System services
--More-- (50%)
```

14.4 less

La commande *less* est variante très puissante de *more* qui permet de revenir en arrière en utilisant la touche *b*. En appuyant en tout moment sur la touche *v* vous appeler votre éditeur préféré pour pouvoir modifier le fichier.

Afficher le contenu du fichier *anaconda-ks.cfg* en utilisant la commande *less* :

```
[root@localhost ~]# less anaconda-ks.cfg
#version=DEVEL
# System authorization information
auth --enableshadow --passalgo=sha512
# Use CDROM installation media
cdrom
# Use graphical install
graphical
# Run the Setup Agent on first boot
firstboot --enable
ignoredisk --only-use=sda
# Keyboard layouts
keyboard --vckeymap=us --xlayouts='us'
# System language
lang en_US.UTF-8

# Network information
network --bootproto=dhcp --device=ens32 --ipv6=auto --activate
network --hostname=localhost.localdomain

# Root password
rootpw --iscrypted
$6$cbauomSFBvt.NLfV$YgD5Ag/DJcKVnQoMYLXn9U2SOhA/oINXE/uU4O6r8qwpEur1F.P
pIQc9GwynmLlpAJ/F0YncSaQtjpg.L2nK60
# System services
anaconda-ks.cfg
```

14.5tail

Affiche les dernières lignes (par défaut les 10 dernières) d'un ou plusieurs *fichiers* et/ou du canal d'entrée standard.

```
[root@localhost ~]# tail /var/log/messages

Jun  9 11:33:06 localhost NetworkManager[7075]: <info>
[1560094386.1979] dhcp4 (ens32):  nameserver '192.168.17.2'
Jun  9 11:33:06 localhost NetworkManager[7075]: <info>
[1560094386.1979] dhcp4 (ens32):  domain name 'localdomain'
Jun  9 11:33:06 localhost NetworkManager[7075]: <info>
[1560094386.1979] dhcp4 (ens32): state changed bound -> bound
Jun  9 11:33:06 localhost dbus[7069]: [system] Activating via systemd:
service name='org.freedesktop.nm_dispatcher' unit='dbus-
org.freedesktop.nm-dispatcher.service'
Jun  9 11:33:06 localhost systemd: Starting Network Manager Script
Dispatcher Service...
Jun  9 11:33:06 localhost dhclient[7148]: bound to 192.168.17.144 --
renewal in 876 seconds.
Jun  9 11:33:06 localhost dbus[7069]: [system] Successfully activated
service 'org.freedesktop.nm_dispatcher'
Jun  9 11:33:06 localhost systemd: Started Network Manager Script
Dispatcher Service.
Jun  9 11:33:06 localhost nm-dispatcher: req:1 'dhcp4-change' [ens32]:
new request (3 scripts)
Jun  9 11:33:06 localhost nm-dispatcher: req:1 'dhcp4-change' [ens32]:
start running ordered scripts...
[root@localhost ~]#
```

15 Rechercher un mot dans un fichier (grep)

La commande *grep* permet de rechercher dans un ou plusieurs *fichiers* la présence d'une *expression* régulière.

Pour rechercher le mot *end* dans le fichier *anaconda-ks.cfg* :

```
[root@localhost ~]# grep end anaconda-ks.cfg  
bootloader --append=" crashkernel=auto" --location=mbr --boot-drive=sda  
%end  
%end  
%end
```

16 Compression et décompression

16.1 gzip

La commande *gzip* permet de compresser un ou de plusieurs *fichiers*.

Pour compresser le fichier *anaconda-ks.cfg*:

```
[root@localhost ~]# gzip anaconda-ks.cfg
```

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 712 May 18 11:06 anaconda-ks.cfg.gz
```

16.2 gunzip

La commande *gunzip* permet de décompresser un ou plusieurs *fichiers* traités par *gzip* (extension *.gz*).

Pour décompresser le fichier *anaconda-ks.cfg.gz* :

```
[root@localhost ~]# gunzip anaconda-ks.cfg.gz
```

```
[root@localhost ~]# ls
anaconda-ks.cfg
```

16.3 compress

La commande *compress* permet de compresser un ou plusieurs *fichiers* (extension *.Z*).

Cette commande n'est pas installée par défaut. Pour l'installer :

```
[root@localhost ~]# yum install ncompress
```

Pour compresser le fichier en utilisant la commande *compress* :

```
[root@localhost ~]# compress anaconda-ks.cfg
```

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 921 May 18 11:06 anaconda-ks.cfg.Z
```

16.4 uncompress

La commande *uncompress* permet de décompresser un ou plusieurs *fichiers* traités par *compress* (extension *.Z*).

Pour décompresser le fichier en utilisant la commande *uncompress* :

```
[root@localhost ~]# uncompress anaconda-ks.cfg.Z
```

```
[root@localhost ~]# ls -l
total 4
-rw----- 1 root root 1244 May 18 11:06 anaconda-ks.cfg
[root@localhost ~]#
```

16.5tar

La commande *tar* permet de créer et restaurer une archive.

Les options importantes sont les suivantes :

Option	Description
c	Créer une sauvegarde.
r	Ajouter des fichiers à une sauvegarde.
t	Lister le contenu d'une sauvegarde.
u	Ajouter des fichiers nouveaux ou plus récents à une sauvegarde.
x	Extraire des fichiers d'une sauvegarde. Par défaut, tar extrait tous les fichiers contenus dans la sauvegarde.
j	Compresser avec bzip2.
z	Compresser avec gzip.

17 cal

La commande `cal` permet d'afficher le calendrier du mois en cours. Sinon elle affiche les douze mois de l'*année* précisée ou le *mois* de l'*année* précisé.

Pour afficher le calendrier du mois de février 2019 :

```
[root@localhost ~]# cal 2 2019
      February 2019
Su Mo Tu We Th Fr Sa
                1  2
 3  4  5  6  7  8  9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28
```

Pour afficher le calendrier du mois de 2020:

```
[root@localhost ~]# cal 2020
                        2020

      January          February          March
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
                1  2  3  4                1      1  2  3  4  5  6  7
 5  6  7  8  9 10 11    2  3  4  5  6  7  8    8  9 10 11 12 13 14
12 13 14 15 16 17 18    9 10 11 12 13 14 15   15 16 17 18 19 20 21
19 20 21 22 23 24 25   16 17 18 19 20 21 22   22 23 24 25 26 27 28
26 27 28 29 30 31     23 24 25 26 27 28 29   29 30 31

      April           May              June
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
                1  2  3  4                1  2      1  2  3  4  5  6
 5  6  7  8  9 10 11    3  4  5  6  7  8  9    7  8  9 10 11 12 13
12 13 14 15 16 17 18   10 11 12 13 14 15 16   14 15 16 17 18 19 20
19 20 21 22 23 24 25   17 18 19 20 21 22 23   21 22 23 24 25 26 27
26 27 28 29 30        24 25 26 27 28 29 30   28 29 30
                        31

      July            August           September
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
                1  2  3  4                1      1  2  3  4  5
 5  6  7  8  9 10 11    2  3  4  5  6  7  8    6  7  8  9 10 11 12
12 13 14 15 16 17 18    9 10 11 12 13 14 15   13 14 15 16 17 18 19
19 20 21 22 23 24 25   16 17 18 19 20 21 22   20 21 22 23 24 25 26
26 27 28 29 30 31     23 24 25 26 27 28 29   27 28 29 30
                        30 31

      October         November         December
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
                1  2  3    1  2  3  4  5  6  7    1  2  3  4  5
 4  5  6  7  8  9 10    8  9 10 11 12 13 14    6  7  8  9 10 11 12
11 12 13 14 15 16 17   15 16 17 18 19 20 21   13 14 15 16 17 18 19
18 19 20 21 22 23 24   22 23 24 25 26 27 28   20 21 22 23 24 25 26
25 26 27 28 29 30 31   29 30                27 28 29 30 31
```


18 kill

La commande *kill* permet de terminer un ou plusieurs processus référencés par leurs *IDs*.

Pour arrêter l'exécution du programme dont le pid (process id) est 1234 :

```
[root@localhost ~]# kill 1234
```

19 passwd

Permet de changer le mot de passe d'un *utilisateur*. Seul l'administrateur peut changer le mot de passe d'un autre utilisateur.

Pour changer le mot de passe de l'utilisateur *root* :

```
[root@localhost ~]# passwd
Changing password for user root.
New password:xxxxxx
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:xxxxxx
passwd: all authentication tokens updated successfully.
```

20 uptime

La commande *uptime* permet d'afficher une ligne d'information sur la machine :

- L'heure, depuis combien de temps la machine fonctionne
- Combien d'utilisateurs sont connectés
- La charge de calcul moyenné sur 1, 5 et 15 minutes

```
[root@localhost ~]# uptime
22:03:26 up 13 min, 2 users, load average:0.00, 0.14, 0.25
```

21 which

La commande *which* permet d'indiquer le chemin d'accès exact d'une commande Linux.

```
[root@localhost ~]# which reboot
/usr/sbin/reboot
```

22 who

La commande *who* affiche des informations au sujet des utilisateurs actuellement connectés. En l'absence de paramètre, *who* affiche les informations suivantes :

- Nom de l'utilisateur
- Nom du terminal
- Date et heure de connexion et nom de la machine distante ou du terminal X.

```
[root@localhost ~]# who
root      tty1      2019-06-06 21:51
root      pts/0      2019-06-06 21:52 (192.168.17.1)
```

23 clear

La commande *clear* permet d'effacer l'écran.

24 Arrêt et redémarrage

24.1 shutdown

La commande *shutdown* permet d'arrêter, de mettre hors tension ou de redémarrer la machine.

Les options importantes sont les suivantes :

Option	Description
H	Arrêter la machine.
P	Mettre hors tension la machine.
r	Redémarrer la machine.

24.2 reboot

La commande *reboot* permet de redémarrer la machine.

24.3 init

Gestionnaire de service (init).

Les options importantes sont les suivantes :

Option	Description
0	Arrêter la machine.
6	Redémarrer la machine.

25 Éditeurs de fichiers

25.1 vim

vim est un éditeur de texte proche de *vi*. Il n'est pas installé par défaut. Pour l'installer :

```
[root@localhost ~]# yum install vim
```

25.2 nano

nano est un éditeur de texte plein page. Il n'est pas installé par défaut. Pour l'installer :

```
[root@localhost ~]# yum install nano
```