

# TP9 – Manipuler les fichiers et les ré- pertoires

SAOU Rayan

## Table des matières

Table des matières .....	1
1.Examen des routes statiques.....	1

## 1. Les commandes de gestion de fichiers et de répertoires

Une fois notre utilisateur « guest » crée, nous tapons la commande ***pwd*** pour avoir accès au repertoire actuel :

```
guest@DEB13Server:~$ pwd
/home/guest
guest@DEB13Server:~$ _
```

Ici, nous sommes dans le repertoire : ***/home/guest***

Pour copier le fichier ***/etc/passwd*** en « mot\_de\_passe » nous faisons ***cp /etc/passwd mot\_de\_passe***

```
guest@DEB13Server:~$ cp /etc/passwd mot_de_passe
```

Et nous copions les fichier ***/etc/group*** et ***/etc/profile*** au repertoire courant « . » donc :

```
guest@DEB13Server:~$ cp /etc/passwd mot_de_passe
guest@DEB13Server:~$
guest@DEB13Server:~$ cp /etc/group /etc/profile .
guest@DEB13Server:~$
```

Puis nous affichons les fichiers y compris les fichiers cachés avec ***ls -a*** (*des fichiers qui doivent être présent plus tard sont ici car les captures d'écrans ont été refaites car corrompues...*)

```
guest@DEB13Server:~$ ls -a
.  ..  .bash_history  .bash_logout  .bashrc  g2  group  .local  mot_de_passe  .profile  profile  un_rep
guest@DEB13Server:~$ _
```

Puis nous créons un répertoire `un_rep` avec la commande ***mkdir*** et nous renommons le fichier `group` en `g2`. Et nous détruisons le fichier `group` avec ***rm***

```
guest@DEB13Server:~$ mkdir un_rep1
guest@DEB13Server:~$ cp group un_rep1/g2
guest@DEB13Server:~$ rm group
guest@DEB13Server:~$
```

Nous créons ensuite un alias qui permet la confirmation de suppression (`rm -i`) :

```
$ alias rm='rm -i'
```

```
guest@DEB13Server:~$ alias rm='rm -i'
guest@DEB13Server:~$ rm un_rep1/g2
rm : supprimer 'un_rep1/g2' du type regular file ? n
guest@DEB13Server:~$
```

On se déplace dans un autre répertoire (`/usr/bin`) avec la commande `cd`, puis nous faisons la commande `ls -l` + une commande (`tail` par exemple) et nous revenons dans notre répertoire par défaut :

```
guest@DEB13Server:/usr/bin$ ls -l tail
-rwxr-xr-x 1 root root 80360 4 juin 2025 tail
guest@DEB13Server:/usr/bin$ _
```

Et ensuite nous voir la catégorie d'un fichier (`/usr/bin/head` par exemple) avec la commande ***file*** :

```
guest@DEB13Server:/usr/bin$ file head
head: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.
e935f94e06a8290962, for GNU/Linux 3.2.0, stripped
guest@DEB13Server:/usr/bin$ _
```

Et la commande ***cat*** affiche d'un seul coup le contenu quand nous l'utilisons

```
guest@DEB13Server:~$ cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug enp0s3
auto enp0s3
iface enp0s3 inet dhcp
# This is an autoconfigured IPv6 interface
iface enp0s3 inet6 auto
guest@DEB13Server:~$ _
```

Et nous visualisons avec la commande *more /etc/services*

```

tcpmux      1/tcp          # TCP port service multiplexer
echo        7/tcp
echo        7/udp
discard     9/tcp          sink null
discard     9/udp          sink null
sysstat     11/tcp         users
daytime     13/tcp
daytime     13/udp
netstat     15/tcp
gotd        17/tcp          quote
chargen    19/tcp          ttytst source
chargen    19/udp          ttytst source
ftp-data    20/tcp
ftp         21/tcp
fsp         21/udp          fspd
ssh         22/tcp          # SSH Remote Login Protocol
telnet      23/tcp
smtp        25/tcp          mail
time        37/tcp          timserver
time        37/udp          timserver
whois       43/tcp          nickname
tacacs      49/tcp          # Login Host Protocol (TACACS)
tacacs      49/udp
domain      53/tcp          # Domain Name Server
domain      53/udp
bootps      67/udp
bootpc      68/udp
tftp        69/udp
gopher      70/tcp          # Internet Gopher
finger      79/tcp
http        80/tcp          www # WorldWideWeb HTTP
kerberos    88/tcp          kerberos5 krb5 kerberos-sec # Kerberos v5
kerberos    88/udp          kerberos5 krb5 kerberos-sec # Kerberos v5
iso-tsap    102/tcp         tsap # part of ISODE
acr-nema    104/tcp         dicom # Digital Imag. & Comm. 300
pop3        110/tcp         pop-3 # POP version 3
sunrpc      111/tcp         portmapper # RPC 4.0 portmapper
sunrpc      111/udp         portmapper
auth        113/tcp         authentication tap ident
nntp        119/tcp         readnews untp # USENET News Transfer Protocol
ntp         123/udp         # Network Time Protocol
--Plus-- (12%)

```

## 2. Les utilitaires

Nous utilisons **head** pour afficher les 10 premières lignes d'un fichier :

```

guest@DEB13Server:~$ head /etc/services
# Network services, Internet style
#
# Updated from https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml .
#
# New ports will be added on request if they have been officially assigned
# by IANA and used in the real-world or are needed by a debian package.
# If you need a huge list of used numbers please install the nmap package.
tcpmux      1/tcp          # TCP port service multiplexer
echo        7/tcp

```

Tandis que pour les 10 dernières lignes nous utilisons tail :

```

guest@DEB13Server:~$ tail /etc/services
sgi-cad      17004/tcp          # Cluster Admin daemon
binkp       24554/tcp          # binkp fidonet protocol
asp         27374/tcp          # Address Search Protocol
asp         27374/udp
csync2      30865/tcp          # cluster synchronization tool
dircproxy   57000/tcp          # Detachable IRC Proxy
tfido       60177/tcp          # fidonet EMSI over telnet
fido        60179/tcp          # fidonet EMSI over TCP

# Local services
guest@DEB13Server:~$

```

Et la commande pour filtrer les résultats en cherchant une chaîne est grep, donc nous allons l'utiliser pour chercher HTTP (en prenant soin décommentant l'option de couleur dans le *.bashrc* et en nous reconnectant) :

```

guest@DEB13Server:~$ grep HTTP /etc/services
http        80/tcp            www                # WorldWideWeb HTTP
https       443/udp           # HTTP/3
hkp         11371/tcp         # OpenPGP HTTP Keyserver
guest@DEB13Server:~$ _

```

Et pour ne pas tenir compte de la classe on utilise l'option -i :

```

guest@DEB13Server:~$ grep -i http /etc/services
# Updated from https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml
http        80/tcp            www                # WorldWideWeb HTTP
https       443/tcp           # http protocol over TLS/SSL
https       443/udp           # HTTP/3
http-alt    8080/tcp          webcache           # WWW caching service
hkp         11371/tcp         # OpenPGP HTTP Keyserver
guest@DEB13Server:~$

```

Pour afficher les fichiers qui commencent par http on met un '^' :

Ensuite nous affichons les trois dernières lignes d'un fichier trié :

```

guest@DEB13Server:~$ sort /etc/services | tail -3
zope          9673/tcp          # zope server
zope-ftp     8021/tcp          # zope management by ftp
zserv       346/tcp           # Zebra server
guest@DEB13Server:~$ _

```

Ensuite les pages triées, page par page, dont les lignes ne commencent pas par # :

```

acr-nema      104/tcp          dicom          # Digital Imag. & Comm. 300
afpovertcp   548/tcp          # AFP over TCP
afs3-bos     7007/udp         # basic overseer process
afs3-callback 7001/udp        # callbacks to cache managers
afs3-fileserver 7000/udp
afs3-kaserver 7004/udp         # AFS/Kerberos authentication
afs3-prserver 7002/udp         # users & groups database
afs3-rmtsys  7009/udp         # remote cache manager service
afs3-update  7008/udp         # server-to-server updater
afs3-vlserver 7003/udp        # volume location database
afs3-volser  7005/udp         # volume management server
amanda       10080/tcp        # amanda backup services
amandaidx   10082/tcp        # amanda backup services
amidxtape   10083/tcp        # amanda backup services
amqp         5672/sctp
amqp         5672/tcp
amqps        5671/tcp          # AMQP protocol over TLS/SSL
asf-rmcp     623/udp          # ASF Remote Management and Control Protocol
asp          27374/tcp        # Address Search Protocol
asp          27374/udp
auth         113/tcp          authentication tap ident
babel        6696/udp         # Babel Routing Protocol
bacula-dir   9101/tcp         # Bacula Director
bacula-fd    9102/tcp         # Bacula File Daemon
bacula-sd    9103/tcp         # Bacula Storage Daemon
obs          7000/tcp
bgp          179/tcp          # Border Gateway Protocol
bgpd         2605/tcp         # bgpd vty (zebra)
biff         512/udp          comsat
binkp        24554/tcp        # binkp fidonet protocol
bootpc      68/udp
bootps      67/udp
canna       5680/tcp         # cannaserver
cfengine     5308/tcp
chargen     19/tcp          ttytst source
chargen     19/udp          ttytst source
cisco-sccp  2000/tcp         # Cisco SCCP
clc-build-daemon 8990/tcp        # Common lisp build daemon
clearcase   371/udp          Clearcase
cmip-agent  164/tcp
cmip-agent  164/udp
cmip-man    163/tcp         # ISO mgmt over IP (CMOT)
cmip-man    163/udp
guest@DEB13Server:~$ grep -v '^#' /etc/services | sort | more_

```

Pour afficher le nombre de caractères il s'agit de la commande *wc* :

```
guest@DEB13Server:~$ wc /etc/services
365  1795 12990 /etc/services
```

Et le nombre de lignes :

```
guest@DEB13Server:~$ wc -l /etc/services
365 /etc/services
```

Nous pouvons nous connecter en root et on créer un fichier suivant à l'aide de l'éditeur *nano* et nous en affichons le contenu :

```
root@DEB13Server: ~#cat etudiants.txt
Nicolas
Denis
Souphiane
Azer
Fabien
Cedric
Tiphaine
Xavier
Antoine
David
```

Nous trions dans l'ordre alphabétique le fichier :

```
root@DEB13Server: ~#sort etudiants.txt
Antoine
Azer
Cedric
David
Denis
Fabien
Nicolas
Souphiane
Tiphaine
Xavier
```

Ensuite les prénoms dans l'ordre inverse :

```
root@DEB13Server: ~#sort -r etudiants.txt
Xavier
Tiphaine
Souphiane
Nicolas
Fabien
Denis
David
Cedric
Azer
Antoine
```

Et nous mettons le résultat du **sort** (avec l'option -o « output ») dans le fichier prénoms\_tries :

```
root@DEB13Server: ~#sort -o prenom_tries etudiants.txt
root@DEB13Server: ~#ls -l
total 8
-rw-r--r-- 1 root root 73  3 déc.  10:48 etudiants.txt
-rw-r--r-- 1 root root 73  3 déc.  10:53 prenom_tries
root@DEB13Server: ~#
```

On conserve les 3 premiers caractères du fichier :

```
root@DEB13Server: ~#cut -c -3 etudiants.txt
Nic
Den
Sou
Aze
Fab
Ced
Tip
Xav
Ant
Dav
```

On conserve ensuite les caractères entre le 2<sup>e</sup> et le 5<sup>e</sup> de chaque ligne :

```
root@DEB13Server: ~#cut -c 2-5 etudiants.txt
icol
enis
ouph
zer
abie
edri
ipha
avie
ntoi
avid
```

Et on conserve les caractères à partir du 3<sup>e</sup> :

```
root@DEB13Server: ~#cut -c 3- etudiants.txt
colas
nis
uphiane
er
bien
dric
phaine
vier
toine
vid
```

Ensuite nous modifions le fichier pour le transformer en CSV (en ajoutant la spécialité, la note et la mention séparées par une virgule.) avec l'éditeur *Nano* :

```
root@DEB13Server: ~#cat notes.csv
Antoine, SISR, 13, AB
Azer, SISR, 13, AB
Cedric, SISR, 13, AB
David, SISR, 14, AB
Denis, SLAM, 17, TB
Fabien, SISR, 13, AB
Nicolas, SISR, 17, TB
Souphiane, SLAM, 16, TB
Tiphaine, SLAM, 15, B
Xavier, SLAM, 11, M
```

Ensuite nous modifions le fichier en affichant la première colonne :

```

root@DEB13Server: ~#cut -d , -f 1 notes.csv
Antoine
Azer
Cedric
David
Denis
Fabien
Nicolas
Souphiane
Tiphaine
Xavier

```

Ensuite, nous conservons les trois premières colonnes :

```

root@DEB13Server: ~#cut -d , -f 1-3 notes.csv
Antoine, SISR, 13
Azer, SISR, 13
Cedric, SISR, 13
David, SISR, 14
Denis, SLAM, 17
Fabien, SISR, 13
Nicolas, SISR, 17
Souphiane, SLAM, 16
Tiphaine, SLAM, 15
Xavier, SLAM, 11

```

Et nous affichons finalement la valeur des champs de la ligne 1 et 3 :

```

root@DEB13Server: ~#cut -d , -f 1,3 notes.csv
Antoine, 13
Azer, 13
Cedric, 13
David, 14
Denis, 17
Fabien, 13
Nicolas, 17
Souphiane, 16
Tiphaine, 15
Xavier, 11

```