Table des matières

rclone 3
dd, ssh & split 3
Convertir timestamp dmesg 3
Serveur web light 3
umask 3
unix.txt 3
VI 3
diff remote 4
find sous AIX, le backslash ne fonctionne pas 4
Utiliser tar, gzip et cie 4
Avec 7z 4
Comment mettre de la couleur dans des scripts via les séquences escapes (xterm only) ? 4
Masquer le frappe d'un password 4
Redimensionner le nb de colonne d'un terminal 5
Cron sous Solaris 5
Faire prendre en compte un pattern spécifique par egrep 5
/etc/shadow sous Solaris 5
Substitution rapide de répertoires avec cd 5
Liste des erreurs SHELL classiques (patterns) 5
Affectation et tests sous unix 6
Affectation 6
Tests 6
Exécuter un 'find -older' 6
Récupérer plus d'infos sur un process 6
Liste des signaux Unix 6
Inverser une chaine de caractères avec sed 6
/usr/bin/rm: arg list too long 7
Modifier un binaire .exe Windows sous Unix 7
Effacer un fichier 'ineffaçable' 7
Accélerer la création d'un LV et son mirroir 8
Transformer un CSV en HTML 8
Lancer un process en background, checker sa présence, le killer si besoin 8
VI Reference 8
Shift J 9
Backup via FTP avec tar, gzip et gpg 9
Convertif un PDF en n images 9
Fichiers sensibles 9
Liste des fichiers sensibles AIX 9
Usage stand-alone 9
Usage HA-CMP 10
Liste des fichiers sensibles HP 10
Usage stand-alone 10
Usage MCServiceGuard 10
Liste des fichiers sensibles SUN 10
Usage stand-alone 10
Envoyer un mail au format HTML 10
Envoyer un mail avec une pièce jointe 10
Modification du fichier ''/etc/mail/genericstable'' 11
Rechargement de la configuration pour prise en compte par sendmail 11
Mixer 2 fichiers en colonne 11
Ports utilisés par un process 11
Débugger un process 11
Config Panasas 11
sar 11
Créer un fichier de 1 Go 11
bad interpreter: Permission denied 11
tar > 2 Go 12
Déterminer le process bloquant la suppression d'un fichier 12
Tester le débit en FTP 12
/etc/shadow 12
Bosser sans | 12
Conversion HEXA <-> Decimal 13
highlighted grep 13
Empêcher vi d'effacer l'écran lors de la sortie 13
Supprimer les espaces d'un fichier 13
Sendmail, écoute sur localhost 13

sar -d reformaté 14
Coupure SAN 14
AIX 14
HP-UX 14
Solaris (Veritas) 14
Convertir une chaine en MD5 14
ssh et background 15
Utiliser ed 15
Customiser son historique 15
Configuration de (x)inetd 15
Aligner les colonnes en shell 16
Restaurer des droits 16
Tunnel SSH 16
Tableaux / Variables 16
PrimePower 17



Astuces en vrac non triées et multi-OS (presque)

rclone

```
/usr/bin/rclone --config /root/.config/rclone/rclone.conf  copy secret:ndlp/nas-bck/win7-bck/ /ZP_backup/win7-bck/ -P --include "lv_win7B"
```

- rclone rcat

```
export TMDIR=/ZP_vDisks/tmp
```

dd, ssh & split

```
dd if=/dev/datavg/lv_win10 conv=sync status=progress |ssh nas "split -b 8G - /ZP_backup/win10-bck/lv_win10."*
```

Convertir timestamp dmesg

```
dmesg | sed -r 's#^\[([0-9]+\.[0-9]+)\](.*)#echo -n "[";echo -n $(date --date="@$(echo "$(grep btime /proc/stat|cut -d " " -f 2)+\1" | bc)" +"%c");echo -n "]";echo -n "\2"#e'
```

Serveur web light

Pour servir un fichier :

```
while true ; do nc -l 80  < test.iso ; done
```

umask

umask	fichiers	répertoires
000	666 (rw-rw-rw-)	777 (rwxrwxrwx)
002	664 (rw-rw-r–)	775 (rwxrwxr-x)
022	644 (rw-r–r–)	755 (rwxr-xr-x)
027	640 (rw-r—-)	750 (rwxr-x—)
077	600 (rw—-—-)	700 (rwx—-—-)
277	400 (r—-—-)	500 (r-x—-—-)



Ne pas confondre avec Samba où les paramètres *create mask* s'utilisent sous la forme 0775 par ex.

unix.txt

unix.zip

VI

Ouvrir le fichier à la ligne 15 :

```
vi +15 /etc/passwd
```

Ouvrir le fichier à la première occurence commençant par *toto* :

```
vi +/^toto /etc/passwd
```

diff remote

```
remsh machine1 cat /etc/passw |diff - /etc/passwd
```

find sous AIX, le backslash ne fonctionne pas

Il suffit de les doubler.

Utiliser tar, gzip et cie

Compresser un tar à la volée

```
tar cvf - files |gzip -9 > file.tar.gz
```

Transférer un tar.gz le décompresser et le détarrer sur une machine distante ? (tout ça en une ligne)

```
gzip -d < fichier.tar.gz | remsh machine "cd /path/ ;tar xvf -"
```

Avec 7z

```
7za x -so directory.tar.7z | tar xf -
```

Comment mettre de la couleur dans des scripts via les séquences escapes (xterm only) ?

En utilisant ces variables :

```
ESC='\033'
NORM=${ESC}'[0m'
BOLD=${ESC}'[1m'
UNDERSCORE=${ESC}'[4m'
BLINK=${ESC}'[5m'
REVERSE=${ESC}'[7m'
BLINK=${ESC}'[5m'
REVERSE=${ESC}'[7m'
BLACK=${ESC}'[30m'
RED=${ESC}'[31;1m'
GREEN=${ESC}'[32;1m'
YELLOW=${ESC}'[33m'
BLUE=${ESC}'[34m'
PINK=${ESC}'[35m'
MAGENTA=${ESC}'[36m'
WHITE=${ESC}'[37m'
FONDROUGE=${ESC}'[41m'
FONDVERT=${ESC}'[42m'
FONDJAUNE=${ESC}'[43m'
FONDBLEU=${ESC}'[44m'
FONDPOURPRE=${ESC}'[45m'
FONDBLEUCLAIR=${ESC}'[46m'
```

```
echo "${GREEN}Bonjour le monde ${NORM}"
echo "${FONDROUGE}${BLUE}Bonjour le monde${NORM}"
```

Sous linux :

```
echo -e "${GREEN}Bonjour le monde ${NORM}"
```

Masquer le frappe d'un password

```
stty -echo
read VAR
```

```
stty echo
```

Redimensionner le nb de colonne d'un terminal

```
stty cols 163
```

Cron sous Solaris

Ne pas mettre de ligne(s) vide(s) dans la cron.

Faire prendre en compte un pattern spécifique par egrep

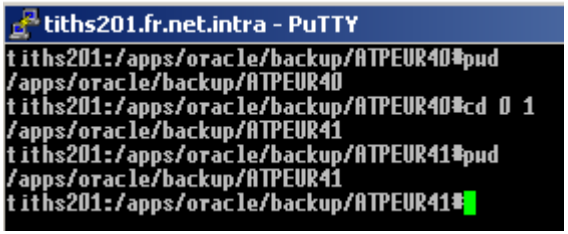
```
egrep -iv "/proc |/mnttab |/fd |swap |/mnt|/cdrom"
```

/etc/shadow sous Solaris

LK : account locked : la crontab ne marche pas

NP : mot de passe impossible mais compte actif

Substitution rapide de répertoires avec cd



Liste des erreurs SHELL classiques (patterns)

```
error
failed
aborted
permission denied
can't create
can't access
can't change
cannot access
cannot open
could not read
could not be written
no such file or directory
already exist
file exists
is a directory
not found
command garbled
No such device
Not a directory
No such device or address
Read only file system
```

```
No space left on device
syntax error
malformed
does not exist
could not be written
argument expected
Segmentation Fault
core dumped
parameter not set
```

Affectation et tests sous unix

Affectation

```
i=`echo "$i+3"|bc`
```

ou

```
((i=i+3))
```

Tests

```
if [ "$i" -le 5 -o "$i" -gt 15 ]
```

ou

```
if ((i<=5))||((i>15))
```

Exécuter un 'find -older'

```
find /tmp -name "somefile" ! -newer /somepath/somefile/
```

Récupérer plus d'infos sur un process

Avec la commande `ptree` ... (à suivre).

Liste des signaux Unix

```
_____________________________________________________________________________
   0     SIGNULL   Null            Check access to pid
   1     SIGHUP    Hangup          Terminate; can be trapped
   2     SIGINT    Interrupt       Terminate; can be trapped
   3     SIGQUIT   Quit            Terminate with core dump; can be trapped
   9     SIGKILL   Kill            Forced termination; cannot be trapped
  15     SIGTERM   Terminate       Terminate; can be trapped
  24     SIGSTOP   Stop            Pause the process; cannot be trapped
  25     SIGTSTP   Terminal stop   Pause the process; can be trapped
  26     SIGCONT   Continue        Run a stopped process
```

Inverser une chaine de caractères avec sed

```
echo "Coucou"|sed '/\n/!G;s/\(.\)\(.*\n\)/&\2\1/;//D;s/.//'
uocuoC
```

/usr/bin/rm: arg list too long

Utiliser xargs :

```
cat liste_de_fichiers|xargs rm
```

Modifier un binaire .exe Windows sous Unix

On recherche la position de la chaîne Configuration dans le binaire Putty.exe

```
machine:/home/user# strings -t d Putty.exe |grep Configuration
3 58476 %s Configuration
```

On check sa position exacte :

```
machine:/home/user# dd if=Putty.exe  bs=1 skip=358476 count=30|od -c
30+0 records in
30+0 records out
0000000   %   s       C   o   n   f   i   g   u   r   a   t   i   o   n
0000020  \0  \0  \0  \0   j     ========== ==========     c   t   r   l   _
0000036
```

Soit précisément :

```
machine:/home/user# dd if=Putty.exe  bs=1 skip=358479 count=13|od -c
13+0 records in
13+0 records out
0000000   C   o   n   f   i   g   u   r   a   t   i   o   n
```

Le mot est en position 358480 (on skip les 358479 premiers) et fait 13 caractere de long. On peut alors decouper le fichier en 3 parties :

```
machine:/home/user# dd if=Putty.exe  bs=1 count=358479 of=Putty.1
machine:/home/user# dd if=Putty.exe  bs=1 skip=358479 count=13 of=Putty.2
machine:/home/user# dd if=Putty.exe  bs=1 skip=$((358479+13)) of=Putty.3
```

On verifie de ne rien avoir perdu en route :

```
machine:/home/user# cat Putty.[1-3]|cksum
1022851399 380928
machine:/home/user# cat Putty.exe|cksum
1022851399 380928
```

On Check la taille de la futur chaîne de caractère (l'adressage ne doit pas varier ⇔ on remplace une chaîne par une autre de taille identique...) :

```
machine:/home/user# cat Putty.2|wc -c
13
machine:/home/user# echo "Hacked By JG.\c"|wc -c
13
```

On modifie la partie 2 (+check) :

```
machine:/home/user# echo "Hacked By JG.\c">Putty.2
machine:/home/user# cat Putty.[1-3]|cksum
1425792615 380928
```

On réassemble le binaire + check :

```
machine:/home/user# cat Putty.[1-3]>Putty.exe
machine:/home/user# cksum Putty.exe
1425792615 380928 Putty.exe
```

Effacer un fichier 'ineffaçable'

ls -ib : pour voir les inodes et les caractères spéciaux

```
find <chemin> -xdev -inum <inode> -exec rm -f {} \;
```

⇒ Attention, le paramètre -xdev permet de rester dans le même FS, en effet le n° d'inode n'est identique **que dans le même FS** !

Accélerer la création d'un LV et son mirroir

La méthode pour accélerer est de créer le volume avec une petite taille, de le mirrorer et ensuite de l'étendre à la taille désirée.

Transformer un CSV en HTML

```
#!/path/to/ksh

CSV=$1
CAPTION=$2
VERSION=$3

awk -F, -v CAPTION="$CAPTION" -v VERSION="$VERSION" '
BEGIN {
       print "<center>"
       print "<table border cellspacing=0 cellpadding=5>"
       print "<caption><B><FONT SIZE=4 FACE=TERMINAL>" CAPTION "</caption></B></FONT>"
       print "<caption><B><FONT SIZE=1 FACE=TERMINAL>" VERSION "</caption></B></FONT>"
       }
{
x=1
print"<tr>"
       while ( x <= NF ) {
              if ( NR ==1) print "\t <th><FONT FACE=TERMINAL SIZE=3>" $x "</FONT></th>"
              else print  "\t <td><FONT FACE=TERMINAL SIZE=2><DIV ALIGN=CENTER>" $x "</DIV></FONT></td>"
              ++x
              }
       print "</tr>"
}
END {
       print "</table>"
       print "</center>"
}' $CSV
```

Lancer un process en background, checker sa présence, le killer si besoin

Appel dsmc avec mise en place d'un timeout si la commande dsmc ne repond pas

```
    ( dsmc q ar "/apps/oracle/backup/${INSTANCE}/" -fromdate=${DateMoins7Jours} >>  ${REP_FIC_TMP}/dumpCheck.${INSTANCE}.$$ 2>&1 ) &
    DPID=$!; echo "DBA_INFO: ${PROG} DUMP PID de $INSTANCE: $DPID"

    # LANCEMENT DU KILLER / TIMEOUT
    ( touch ${REP_FIC_TMP}/timeout_dsmc.$DPID
      sleep 120
      if [ -f ${REP_FIC_TMP}/timeout_dsmc.$DPID ] ; then
          echo "$(date '+%d/%m/%Y %X') : DBA_WARNING ORACLE : ${PROG} KO TIMEOUT REACHED on $INSTANCE [PID: $DPID]" >> ${TNG}
          kill $DPID 2>/dev/null
      fi
    ) &

    # Attente de la fin du process DSMC
    wait $DPID
    [ -f ${REP_FIC_TMP}/timeout_dsmc.$DPID ] && rm ${REP_FIC_TMP}/timeout_dsmc.$DPID
```

VI Reference

Un petit PDF à récupérer :http://files.ndlp.info/bazar/info/Linux-Vi-Reference.pdf

Shift J

```
cat MACHINE_disques|awk '/disk/ {ligne=$0} /\dev\/dsk/ {print ligne" "$1" "$2}'

disk    976  0/0/14/1/0.119.36.19.0.14.6   sdisk    CLAIMED     DEVICE       EMC     SYMMETRIX
                           /dev/dsk/c96t14d6   /dev/rdsk/c96t14d6
disk    983  0/0/14/1/0.119.36.19.0.14.7   sdisk    CLAIMED     DEVICE       EMC     SYMMETRIX
                           /dev/dsk/c96t14d7   /dev/rdsk/c96t14d7
disk    130  0/0/14/1/0.119.36.19.0.15.0   sdisk    CLAIMED     DEVICE       EMC     SYMMETRIX
                           /dev/dsk/c96t15d0   /dev/rdsk/c96t15d0
disk    183  0/0/14/1/0.119.36.19.1.0.0    sdisk    CLAIMED     DEVICE       EMC     SYMMETRIX
                           /dev/dsk/c108t0d0   /dev/rdsk/c108t0d0

disk    976  0/0/14/1/0.119.36.19.0.14.6   sdisk    CLAIMED     DEVICE       EMC     SYMMETRIX /dev/dsk/c96t14d6 /dev/rdsk/c96t14d6
disk    983  0/0/14/1/0.119.36.19.0.14.7   sdisk    CLAIMED     DEVICE       EMC     SYMMETRIX /dev/dsk/c96t14d7 /dev/rdsk/c96t14d7
disk    130  0/0/14/1/0.119.36.19.0.15.0   sdisk    CLAIMED     DEVICE       EMC     SYMMETRIX /dev/dsk/c96t15d0 /dev/rdsk/c96t15d0
disk    183  0/0/14/1/0.119.36.19.1.0.0    sdisk    CLAIMED     DEVICE       EMC     SYMMETRIX /dev/dsk/c108t0d0 /dev/rdsk/c108t0d0
```

Backup via FTP avec tar, gzip et gpg

- En backup full

```
 tar cvf - /var/ --exclude="ftp" --exclude="www" |gpg -e -r ''votreclé'' - | ncftpput -c -u ''login'' -p ''password''\
dedibackup.dedibox.fr var.tar.gpg
```

- En backup incrémental

```
tar cvf - $(find /var -mtime 1 -print) |gpg -e -r ''votreclé'' - | ncftpput -c -u ''login'' -p ''password''\
dedibackup.dedibox.fr var.tar.gpg
```

note : ne fonctionne pas avec des noms de fichiers contenant des espaces

Utiliser plutôt :

```
find /home/$dir -type f -mtime 1 -print | tar cvT - |gpg -e -r 'votreclé' - | ncftpput -c -u ''login'' -p ''password''\
INC/home/$DATE/home_${dir}.tar.gpg
```

Convertif un PDF en n images

Avec Ghostscript :

```
gs -dSAFER -dBATCH -dNOPAUSE -sDEVICE=jpeg -r200 -g640x480 -sOutputFile=20min_Apple_%03d.jpeg 20min_Apple.pdf
```

Fichiers sensibles

Liste des fichiers sensibles AIX

Usage stand-alone

- /etc/filesystems
- /etc/resolv.conf
- /etc/inittab
- /etc/inetd.conf
- /etc/rc.local
- /etc/passwd
- /etc/group
- /etc/security/passwd
- /etc/security/login
- /etc/security/users
- /etc/environment
- /etc/shutdown

Pour SP :

- /etc/rc.sp

Usage HA-CMP

- /etc/hosts
- /.rhosts
- /usr/hacmp/script/*
- /usr/hacmp/exec/*

Liste des fichiers sensibles HP

Usage stand-alone

- /etc/fstab
- /etc/resolv.conf
- /etc/nswitch.conf
- /etc/inittab
- /etc/inetd.conf
- /etc/passwd
- /etc/group

Usage MCServiceGuard

- /etc/cmcluster/*

Liste des fichiers sensibles SUN

Usage stand-alone

- /etc/vfstab
- /etc/resolv.conf
- /etc/nsswitch.conf
- /etc/inittab
- /etc/inetd.conf
- /etc/passwd
- /etc/group
- /etc/hosts
- /etc/project
- /etc/pam.conf
- /.rhosts

Envoyer un mail au format HTML

Pour envoyer un mail au format HTML à partir d'Unix ou de Linux on peut, par exemple, utiliser les commandes suivantes. Par contre je n'ai testé qu'avec du HTML simple.

```
echo "<html>" >> /tmp/mail.check_fs
df -kh |grep $FS|grep "%" >> /tmp/mail.check_fs
echo "<br><br>" >> /tmp/mail.check_fs
echo "<img src="HTTP://www.domain.com/uneimage.jpg"></a>" >> /tmp/mail.check_fs
echo "<br><br>" >> /tmp/mail.check_fs
echo "<html>" >> /tmp/mail.check_fs
 {
    echo "Message-Id: <`date +%Y%m%d%H%M%S`.QAA$$@ndlp.info>"
    echo "To: toto@titi.com,tata@titi.com"
    echo "Subject: Filesystems > ? 85%"
    echo "Content-Type: text/html; charset=iso-8859-1""
    echo "Content-Transfer-Encoding: 7bit"
    cat /tmp/mail.check_fs
    uuencode /etc/hosts /etc/hosts (pour une pièce jointe)
    } | /usr/lib/sendmail -froot -t
```

Envoyer un mail avec une pièce jointe

On peut également envoyer un mail avec une pièce jointe en utilisant `uuencode`. Ci-dessous un exemple avec `mailx` et `unix2dox` pour envoyer un texte avec un piéce jointe :

```
(cat /tmp/mail_encod ; unix2dos /home/ben/reencode.log|uuencode /home/ben/reencode.log /home/ben/reencode.log)\
```

```
|mailx -s "Aucun fichier pour encodage" toto@titi.com
```

On peut aussi combiner avec du HTML.

Modification du fichier ''/etc/mail/genericstable''

On crée un ou plusieurs alias :

```
toto toto@domain.com
```

Rechargement de la configuration pour prise en compte par sendmail

- HP-UX

```
makedbm genericstable genericstable
```

- AIX

```
makemap hash genericstable < genericstable
```

Mixer 2 fichiers en colonne

```
paste fic1 fic2 > fic.out
```

Ports utilisés par un process

```
lsof -a -p 22231 -i
```

Débugger un process

```
tusc -o /tmp/sendmail.out -f -a -k -w all -p 15340
```

Config Panasas

Installer les 2 RPMs

```
/usr/local/sbin/config_panfs -r machine -a -o dir-caching=0
```

sar

```
sar | tr -s "&amp;#91; &amp;#93;" | sed 's/ /,/g' > sar.data
uuencode sar.data sar.xls | sendmail yourname@yourdomain.com
```

Créer un fichier de 1 Go

```
dd if=/dev/zero of=/apps/sybase/dump/col2/test_ben bs=1024 count=1000000
```

bad interpreter: Permission denied

Le FS a peut-être l'option noexec (linux)

tar > 2 Go

- Sur AIX (en JFS):

```
root@SomeMachine:/apps/oracle/backup/RG1P/tmp$ lsfs -q /apps/oracle/backup/RG1P
Name            Nodename   Mount Pt               VFS   Size    Options    Auto Accounting
/dev/lv_oraREG_back --         /apps/oracle/backup/RG1P jfs   102432768 rw         no   no
  (lv size: 104857600, fs size: 102432768, frag size: 4096, nbpi: 4096, compress: no, bf: false, ag: 8)
```

```
tar -cvf - /apps/registrar |split -b 2000m - /apps/oracle/backup/RG1P/tmp/mon_prefixe
```

```
cat mon_prefixe* > mon_fichier
```

Déterminer le process bloquant la suppression d'un fichier

Avec *fuser -dV*

Tester le débit en FTP

```
put "| dd if=/dev/zero bs=1024k count=50 " /dev/null
```

```
ftp> put "| dd if=/dev/zero bs=1024k count=50 " /dev/null
local: | dd if=/dev/zero bs=1024k count=50  remote: /dev/null
227 Entering Passive Mode (A,B,C,D,171,22)
150 Ok to send data.
50+0 records in
50+0 records out
226 File receive OK.
52428800 bytes sent in 4.45 secs (1.1e+04 Kbytes/sec)
```

/etc/shadow

Si vous créez un fichier */etc/shadow* sur un serveur, alors tous les passwords doivent se trouver dans */etc/shadow*. On ne peut pas avoir un tel fichier avec uniquement une entrée. Sinon plus personne ne peut se logger (vérification dans */etc/passwd* PUIS dans */etc/shadow* → si rien n'est trouvé pas de connexion possible 😕).

Bosser sans |

Dans certains cas il peut arriver que le | ne fonctionne pas (consoles foireuses, clavier mal configuré, etc). Ca rend les investigations un peu compliqué quand il faut faire des trucs du genre *ps -ef |grep titi*. En dépannage rapide on peut utiliser les fonctions basiques du named pipe.

- Vous voulez la liste des process appartenant à root et vous ne connaissez pas l'option -u (ps -fu root)

Avec un pipe, on fait :

```
ps -ef|grep ^root
```

Sans pipe

```
mkfifo MonPipe (on reconnait le device ainsi:  prw------- )
puis ps -ef > MonPipe &
grep ^root MonPipe
```

Quand c'est fini :

```
rm -f MonPipe
```

&: On lance le *ps -ef >MonPipe* en tâche de fond, car si on est sur la console, on n'a qu'une fenetre d'utilisable, sinon, on passe par un telnet classique et le probleme ne se pose pas 😉

Pour ceux qui veulent jouer un peu plus avec les pipe nommés:

```
mkfifo fifo_jgab

echo 1 >fifo_jgab &
echo 2 >fifo_jgab &
echo 3 >fifo_jgab &
```

```
cat fifo_jgab

=>
3
2
1
[1]   Done                    echo 1 >fifo_jgab
[2]-  Done                    echo 2 >fifo_jgab
[3]+  Done                    echo 3 >fifo_jgab
```

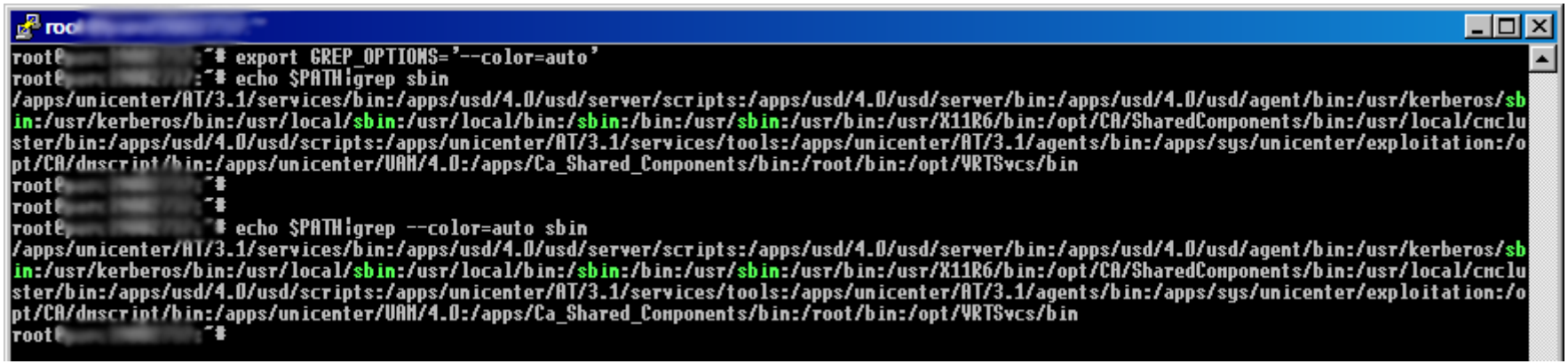
Sinon on a le code ASCII : <key>A</key><key>1</key><key>2</key><key>4</key>

Conversion HEXA <-> Decimal

```
echo "ibase=10;obase=16;1100272340"|bc

printf "%d\n" 0x9E
```

highlighted grep



Empêcher vi d'effacer l'écran lors de la sortie

Lorsqu'on sort d'un vi on n'a plus l'affichage du fichier à l'écran ce qui est pénible ;)

Il suffit d'ajouter l'entrée suivante dans le fichier *.vimrc* de l'utilisateur (on peut créer le fichier s'il n'existe pas) :

```
set t_ti= t_te=
```

Supprimer les espaces d'un fichier

```
ed -s toto.txt <<!
%s/ //g
wq
!
```

Sendmail, écoute sur localhost

Plutôt que de regénérer le fichier *sendmail.cf* on peut modifier, dans le fichier */etc/init.d/sendmail* (ou équivalent) :

```
MODE="-bd"
```

par

```
MODE="-bd -ODaemonPortOptions=Addr=localhost"
```

sar -d reformaté

```
sar -d |awk '{if (NF==1) {next};if (NF==8) {DEV=$1} else {$0=DEV$0};print}'
```

Coupure SAN

- Vérifier les chemins EMC :

```
powermt display
powermt restore (si certains disques n'apparaissent pas après reboot)
```

AIX

- Vérifier la synchro des LVs :

```
lsvg -o | grep -v rootvg | lsvg -i -l
```

- Resynchroniser

```
for i in `lsvg –o | grep –v rootvg`
do
   varyonvg  $i
done
```

HP-UX

- Vérifier la synchro des LVs :

```
vgdisplay –v <vg name> | egrep « stale|unavailable »
```

- Resynchroniser

```
vgchange -a e -q n -s </dev/vg_name> (pour un VG dans un cluster)
vgchange –a y -s </dev/vg_name>
```

Solaris (Veritas)

- Vérifier la synchro des LVs

```
vxdiskconfig
vxdisk list
```

Si *failed* :

```
vxreattach –c <device>
vxreattach <device>
```

- Resynchroniser

```
vxrecover –b <device name>
vxtask list
```

Convertir une chaine en MD5

```
WEBvm:/tmp# echo -n "toto"|md5sum
f71dbe52628a3f83a77ab494817525c6
```

ssh et background

Quand on lance plusieurs commandes distantes en ssh dans une boucle seule la première commande peut passer. Il faut appeler les commandes SSH avec **-n** :

```
ssh -n localhost ps -ef
```

Utiliser ed

- Pour lancer ed

```
ed fic.txt  (on arrive en fin de fichier)
```

- Pour connaitre le n° de ligne courante

```
n   (affiche le n° de la ligne courante et le contenu).
```

- Autres flags rigolos

```
.       affiche la ligne courante
-       aller à la ligne precedente
-3      aller 3 lignes avant
+       aller à la ligne suivante
+2      aller 2 lignes plus loin
5       aller à la ligne 5
d       supprime la ligne
a       ajoute une ligne
r       remplace la ligne courante
s/titi/toto/    remplace le 1er titi par toto de la ligne courante
s/titi/toto/g   remplace toutes les chaînes titi par toto de la ligne courante
,s/titi/toto/g  remplace dans l'ensemble du fichier titi par toto.
/toto       aller à la prochaine occurence de toto
?toto       aller à l'occurence precedente de toto


w       sauvegarde le fichier
w fic2.txt  sauvegarde sous fic2.txt
q       quitter
```

Customiser son historique

```
export HISTTIMEFORMAT="[ %d/%m/%Y %H:%M:%S ]"
```

Dans le fichier /etc/profile du coup n'importe quel shell est impacté. Après on peut voir comment le mettre. Ca donne ça :

```
1326  [ 03/02/2009 08:19:24 ] pwd
1327  [ 03/02/2009 08:19:27 ] ps aux
1328  [ 03/02/2009 08:19:32 ] commande_degueu
1329  [ 03/02/2009 08:19:36 ] ls -ltr
1330  [ 03/02/2009 08:19:49 ] make
1331  [ 03/02/2009 08:19:55 ] ls -l /etc/passwd
1332  [ 03/02/2009 08:19:58 ] history
```

On peut customiser la sortie et on peut aussi virer certaines commandes pour ne pas alourdir les fichiers histo :

```
HISTIGNORE="ls:cd:ll"
```

Configuration de (x)inetd

Pour ajouter un nouveau démon en utilisant le serveur de demon inetd (xinetd sous linux)

- 1) Ajouter les infos dans /etc/services pour faire le lien entre le demon et le port/protocole qui sera utilisé

```
mondemon        1234/tcp        #BlaBla
```

- 2.1) Modifier /etc/inetd.conf (HP-UX / AIX / Solaris)

```
 mondemon       stream tcp    nowait  root  /monchemin/monscript mondemon
```

- 2.2) Créér le fichier /etc/xinetd.d/mondemon (Linux)

```
service mondemon
{
        disable = no
        flags           = REUSE
        socket_type     = stream
        wait            = no
        user            = root
        server          = /monchemin/monscript
        server_args     = mondemon
        log_on_failure  += USERID
}
```

- 3) Refresh de inetd

service xinetd reload sur linux

```
ou kill -HUP <pid du process inetd>
```

- 4) Checks

```
netstat -an |grep 1234
tcp        0      0 0.0.0.0:1234                0.0.0.0:*                   LISTEN
=> on ecoute (LISTEN) sur le port 1234 renseigné dans /etc/services


depuis un autre serveur, lancer un telnet <monserveur> 1234
```

Aligner les colonnes en shell

```
awk '{printf "%-20s%-20s%-20s%-20s\n", $1, $2, $3, $4}' /tmp/lspv.txt
```

Restaurer des droits

```
find /etc/ -type f -exec ls -l {} \;|awk '
{ u=substr($1,2,3); gsub("\-","",u);
  g=substr($1,5,3); gsub("\-","",g);
  o=substr($1,8,3); gsub("\-","",o);
  print "chmod u="u",g="g",o="o,$NF
}' > /tmp/listefic.txt


find /etc/ -type d -exec ls -ld {} \;|awk '
{ u=substr($1,2,3); gsub("\-","",u);
  g=substr($1,5,3); gsub("\-","",g);
  o=substr($1,8,3); gsub("\-","",o);
  print "chmod u="u",g="g",o="o,$NF
}' > /tmp/listedir.txt
```

Tunnel SSH

```
ssh -L 9999:127.0.0.1:9999 serveur_de_rebond /usr/local/bin/ssh -L 9999:127.0.0.1:22 -N serveur_de_destination
```

Tableaux / Variables

```
PATHS=( "//toto"  "//titi" "//tata" )
MOUNTS=(  "/mnt/toto" "/mnt/titi" "/mnt/tata" )
for (( i = 0 ; i < ${#PATHS[@]} ; i++ ))
do
 echo ${PATHS[$i]} ${MOUNTS[$i]}
done
```

PrimePower

Accès console backup :

```
telnet host 8010
```

From:
https://unix-bck.ndlp.info/ - **Where there is a shell, there is a way**

Permanent link:
https://unix-bck.ndlp.info/doku.php/informatique:nix:tips_infos?rev=1665500952

Last update: **2022/10/11 15:09**