

Outils de partitionnement: Rlisk cfdisk pdisk Norton Partition Magic

Les types de Partitions: Partition primaire (Principale); Nbre Max = 4 /
Partition tendue; Nbre Max = 1
Kemel (noyau) : gestion du système de fichier, du processeur, de la mémoire centrale et des périphériques
Shell : interpréteur de commande (bash, ksh, csh, zsh, etc.)

Le système de fichiers:

Ext2fs: Journalisé (réparation rapide en cas d'arrêt brutal) Utile pour gros disques durs - Quotas disponibles - ACL supportés - Passage d'Ext2fs à ext3fs facile / **ReberIS/ ext2fs**

L'antériorité du système Linux

/bin contient les exécutables essentiels au système, employés par tous les utilisateurs
/boot contient les fichiers permettant à Linux de démarrer (Modules du noyau)
/dev contient les points d'entrée des périphériques
/etc contient les commandes et les fichiers nécessaires à l'administrateur du système
/home répertoire personnel des utilisateurs
/opt contient des packages d'applications supplémentaires

Commandes de Base:

ls : lister le contenu d'un répertoire
rm : supprimer un fichier
cp : copier un fichier
ln : créer un lien sur un fichier
man : afficher l'aide d'une commande
mv : déplacer un fichier
cd : changer de répertoire
mkdir : créer un répertoire
rmdir : supprimer un répertoire
pwd : afficher le répertoire courant
cat : afficher le contenu d'un fichier
file : afficher le type de contenu du fichier
locate : localiser un fichier sur le disque
adduser : ajouter un utilisateur
deluser : supprimer un utilisateur
usermod : changer les propriétés d'un compte
passwd : changer le mot de passe

cat nom_fichier affiche le contenu du fichier
cat fich1 fich2 Affiche deux fichiers à la fois
cat > nom_fichier créer un nouveau fichier s'il n'existe pas sinon son contenu sera écrasé fin de saisie par "ctrl-d"
cat fich1 > fich2 : envoie fich1 à fich2 existe il sera écrasé, sinon il sera créé
cat fich1 >> fich2 : ajoute le contenu de fich1 à celui de fich2
cat >> fich2 ajoute le texte saisi à fich2

more : visualisation par page (descente seule)
less : visualisation par page (montée et descente)
head : partie sup. d'un fichier
tail : Partie inf d'un fichier
wc : Comptage du nombre de mots, lignes ou caractères
file : Afficher le type d'un fichier
diff : Différences entre deux fichiers
sort : Tri du contenu d'un fichier
file : Afficher le type d'un fichier
diff : Différences entre deux fichiers
grep : Rechercher un mot dans un fichier

chmod : Changer les permissions
umask affiche le masque de l'utilisateur actif
umask affiche les permissions correspondantes au masque, sous forme symbolique.
umask masque fixe les permissions ultérieures de création des fichiers de l'utilisateur actif.
chgrp : Changer le groupe

ls > essa1 - envoie le résultat de ls dans essai
ls >> essa2 - envoie le résultat de ls dans essai2 avec concaténation
ls -z2 > essa3 - le message d'erreur sera envoyé à essa3
cat < essa4 - affiche
cat < essa4 >> essa1
&> redirige la sortie standard et la sortie d'erreur
2> redirige la sortie d'erreur
>> concatène la sortie standard
> redirige la sortie standard
< redirige l'entrée standard

vi nom_fichier

APPEL
vi -R nom_fichier : pour éditer un fichier en lecture seule

vi : Pour ouvrir un nouveau fichier sans nom

RENSEIGNEMENT

se **h**=n fixe la tabulation à n caractère

(par défaut = 8)

se nu Affiche les numéros de ligne

se nonu Supprime les numéros de lignes

:= affiche le numéro de la ligne courante

<ctrl>+g ou f affiche l'état du fichier

: et réécrit le fichier tel qu'il est sur le disque

DEPLACEMENT

<ctrl> + b Page précédente (back)

<ctrl> + f Page précédente (Forward)

G Dernière ligne

nG Ligne n

H Haut de l'écran (high)

M Milieu de l'écran (Middle)

L Bas de l'écran

n <space> n caractères vers la droite

0 Début de la ligne ^ Premier caractère non blanc

\$ dernier caractère de la ligne

ou k Ligne précédente

ou j Ligne suivante

ou l vers la droite

ou h vers la gauche

n <Return> n lignes vers

+ ou <Return> début de la ligne suivante / - début de la ligne précédente

RECHERCHE

/motif <return> Recherche motif vers l'avant

?motif <return> Recherche motif vers l'arrière

n Répète la dernière commande de recherche

INSERTION

i <ESC> Avant le curseur

I <ESC> Au début de la ligne courante

a <ESC> Après le curseur

A <ESC> En fin de la ligne courante

o <ESC> Au dessous de la ligne courante

O <ESC> Au dessus de la ligne courante

MODIF ETREMPLACEMENT

<ctrl>+u Annule le texte inséré de la ligne courante

<ctrl>+v Insère le caractère spécial qui suit

<ctrl>+l Insère un saut de page

r # Remplace le caractère courant par #

R <ESC> Remplace la chaîne de caractères

courante par la chaîne entre R et <ESC>

~ Bascule de Majuscule en minuscule et inversement

~ Répète la dernière commande modifiant le texte

Effacements:

x Effacement du caractère sous le curseur

3x de 3 caractères à partir de la position du curseur

dd de la ligne où se trouve le curseur

4dd de 4 lignes à partir de la ligne du curseur

Restitution:

u Restitution de la dernière modification (undo)

U La ligne courante avant modification (sauvdd)

P dernière commande d'effacement

MOUVEMENTS DE LIGNE

J joint la ligne courante à la suivante

r <Return> brise en 2 lignes à partir de la position du curseur

yy Place la ligne courante dans le tampon de travail

dd idem avec effacement de la ligne

10yy Place 10 lignes courantes dans le tampon de travail

10dd idem avec effacement

DECALAGE

>> (ou <<) décalage vers la droite (ou vers la gauche) de toute la ligne de m

positions, m étant défini par la commande set sw= m (par défaut m=8)

n> (ou <<) décalage des n lignes suivantes

: **n1, n2 >>** (ou n1, n2 <<) décalage des lignes n1 à n2

SUBSTITUTIONS

:/old/newp substitution du 1er "old" de la ligne courante par "new"

:/old/newgp Substitution de tous les old de la ligne courante par new

i, j, s,old/newp substitution des 1ers old des lignes i à j par new

i, j, s,old/newgp substitution de tous les old des lignes i à j par new

z annule la plus récente substitution réalisée

INSERTIONS DE FICHIERS

:nr nom_fichier insère le fichier nommé après la ligne n

:r nom_fichier insère le fichier nommé après la ligne courante

shell

echo, affiche son argument texte la sortie standard.

L'option e pour insérer les caractères spéciaux : \n (saut ligne) \b (retour

arrière), \t (tabulation) \a (alarme), \f (fin sans saut de ligne) *echo "Bonjour à*

tous " *echo "Bonjour à toutes "net à tous!"*

read, permet l'affectation directe par lecture de la valeur.

echo "Donnez votre nom à présent " → *read prenomnom* → *echo "Bonjour*

Sprenom \$nom"

set affiche la liste des variables actuelles ainsi que leurs valeurs

unset pour supprimer une variable (a=Bon b=jour → *echo \$a\$b* *echo \$a\$oir*

→ *echo \$a\$oir* → Bonjour)

alias: affiche la liste de tous les alias / **alias h="ls -al"** : pour remplacer la

commande « ls -al » par la

fonction shell **Sortie()** / { / clear / echo déconnexion / exit / } .

appel d'un Shell **sh nom_script ./nom_script**

variables spéciales

\$0 est le nom dans lequel le script est appelé

\$# est le nombre de paramètres passés en argument

\$* est la liste des paramètres passés en argument

\$? est le code de retour de la dernière commande (0 si la commande a été

correctement exécutée

\$1, \$2, ... \$9: représentent les arguments du script

Exemple (<<vi my_script>> création : *echo \$0* a été appelé avec \$#

paramètres *echo* qui sont \$*

Exécution (<<sh my_script a b c d>> *my_script* a été appelé avec 4 paramètres

qui sont : a b c d)

L'instruction

If commande **then** commandes **fi** (exemple : **if** **grep -i "tark"**

/etc/passwd → **then** **echo** L'utilisateur **tark** est connu du système **fi**)

If commande **then** commandes **else** commande **fi**