**453---SYSTEME D'EXPLOITATION - e-learning - Séquence 11**

**Accès au système de fichiers – 5**

**Légende :**

Explications sur fond blanc

Exercice à accomplir sur fond rouge clair

Une ou plusieurs solutions possibles pour accomplir l’exercice sur fond vert clair

Syntaxe du shell : les redirections de fichier :

< Redirection en entrée

> Redirection en sortie (création ou réécriture)

>> Redirection en sortie (création ou ajout)

« | » symbolise un « pipe », c’est-à-dire un « tube de communication » entre deux processus.

Un éditeur est un programme de traitement de texte, pourvu de fonctionnalités plus ou moins complexes.

« gedit » est un éditeur en mode graphique, disponible dans l’interface graphique du système, via le menu « Activités », puis « Afficher les applications ». Il est pourvu de fonctionnalités complexes pour assurer la mise en évidence, selon un mode de coloration, des éléments de nature semblable (commandes, instructions, variables, etc.) d’un texte de script ou de code source écrit dans un langage de programmation.

En mode texte, « nano » est un éditeur installé par défaut. Par contre, « emacs » et « joe » sont des éditeurs à installer.

**Quelques commandes :**

SORT(1)

NOM

sort - Trier les lignes de fichiers texte

SYNOPSIS

sort [OPTION]... [FICHIER]...

sort [OPTION]... --files0-from=F

DESCRIPTION

Afficher sur la sortie standard la concaténation triée de tous les FICHIERs.

HEAD(1)

NOM

head - Afficher le début des fichiers

SYNOPSIS

head [OPTION]... [FICHIER]...

DESCRIPTION

Afficher par défaut les 10 premières lignes de chaque FICHIER sur la sortie standard.

Avec plus d'un FICHIER, faire précéder chacun d'un en-tête donnant le nom du fichier.

L'entrée standard est lue quand FICHIER est omis ou quand FICHIER vaut « - ».

TAIL(1)

NOM

tail - Afficher la dernière partie de fichiers

SYNOPSIS

tail [OPTION]... [FICHIER]...

DESCRIPTION

Afficher par défaut les 10 dernières lignes de chaque FICHIER sur la sortie standard.

Lorsqu'il y a plus d'un FICHIER, faire précéder chaque groupe de lignes

d'un en-tête donnant le nom du fichier. L'entrée standard est lue

quand FICHIER est omis ou quand FICHIER vaut « - ».

GREP(1)

NOM

grep, egrep, fgrep, rgrep - Afficher les lignes correspondant à un motif donné

SYNOPSIS

grep [OPTIONS] MOTIF [FICHIER...]

grep [OPTIONS] [-e MOTIF | -f FICHIER] [FICHIER...]

DESCRIPTION

grep recherche dans les FICHIERs indiqués les lignes correspondant à un certain

MOTIF. Par défaut, grep affiche les lignes qui contiennent une correspondance

au motif. L'entrée standard est lue si FICHIER est omis ou si FICHIER vaut « - ».

Trois variantes du programme sont disponibles : egrep, fgrep et rgrep ; egrep

est identique à grep -E, fgrep est identique à grep -F et rgrep est identique à

grep -r. L'appel direct à egrep ou fgrep est déconseillé, mais est toujours possible pour

permettre à d'anciennes applications qui les utilisent de fonctionner sans modification.

CAT(1)

NOM

cat - Concaténer des fichiers et les afficher sur la sortie standard

SYNOPSIS

cat [OPTION]... [FICHIER]...

DESCRIPTION

Concaténer le(s) FICHIER(s) ou l'entrée standard, et les afficher sur la sortie standard.

APT(8)

NOM

apt - interface en ligne de commande

SYNOPSIS

apt [-h] [-o=chaîne\_de\_configuration] [-c=fichier\_de\_configuration]

[-t=publication\_cible] [-a=architecture] {list | search | show | update |

install paquet [{=numéro\_version\_paquet | /publication\_cible}]... |

remove paquet... | upgrade | full-upgrade | edit-sources |

{-v | --version} | {-h | --help}}

DESCRIPTION

apt (Advanced Package Tool) est un outil en ligne de commande pour gérer les paquets.

Il fournit une interface en ligne de commande au système de gestion de paquets. Voir

aussi apt-get(8) et apt-cache(8) pour davantage d'options en ligne de commande.

APT-GET(8)

NOM

apt-get - Utilitaire APT pour la gestion des paquets - interface en ligne de commande

SYNOPSIS

apt-get [-asqdyfmubV] [-o=chaîne\_de\_configuration] [-c=fichier\_de\_configuration]

[-t=publication\_cible] [-a=architecture] {update | upgrade | dselect-upgrade |

dist-upgrade | install paquet [{=numéro\_version\_paquet | /publication\_cible}]... |

remove paquet... | purge paquet... |

source paquet [{=numéro\_version\_paquet | /publication\_cible}]... |

build-dep paquet [{=numéro\_version\_paquet | /publication\_cible}]... |

download paquet [{=numéro\_version\_paquet | /publication\_cible}]... |

check | clean | autoclean | autoremove | {-v | --version} | {-h | --help}}

DESCRIPTION

apt-get est le programme en ligne de commande pour la gestion des paquets. Il peut être

considéré comme l'outil de base pour les autres programmes de la bibliothèque APT.

Plusieurs interfaces utilisateur existent, comme aptitude(8), synaptic(8) et wajig(1).

APTITUDE(8)

NOM

aptitude - interface évoluée pour le gestionnaire de paquets

SYNOPSIS

aptitude [<options>...] {autoclean | clean | forget-new | keep-all | update}

aptitude [<options>...] {full-upgrade | safe-upgrade} [<paquets>...]

aptitude [<options>...] {build-dep | build-depends | changelog | download |

forbid-version | hold | install | markauto | purge | reinstall | remove | show |

unhold | unmarkauto | versions} <paquets>...

aptitude extract-cache-subset <répertoire-sortie> <paquets>...

aptitude [<options>...] search <motifs>...

aptitude [<options>...] {add-user-tag | remove-user-tag} <étiquettes> <paquets>...

aptitude [<options>...] {why | why-not} [<motifs>...] <paquet>

aptitude [-S <nom-fonct>] [--autoclean-on-startup | --clean-on-startup | -i | -u]

aptitude help

DESCRIPTION

aptitude est une interface en mode texte pour le gestionnaire de paquets

de Debian GNU/Linux. Elle permet à l'utilisateur de connaître la liste des paquets et

de réaliser des tâches d'administration comme l'installation, la mise à jour ou

la suppression de paquets. Ces tâches peuvent être réalisées en mode « interactif »

ou à partir de la « ligne de commande ».

ACTIONS EN LIGNE DE COMMANDE

Le premier argument qui ne commence pas par un tiret (« - ») sera considéré comme étant

la commande que le programme doit réaliser. Si aucune commande n'est donnée,

aptitude démarrera en mode interactif.

NANO(1)

NAME

nano - Nano's ANOther editor, an enhanced free Pico clone

SYNOPSIS

nano [OPTIONS] [[+LINE,COLUMN] FILE]...

DESCRIPTION

nano is a small, free and friendly editor which aims to replace Pico, the default

editor included in the non-free Pine package. Rather than just copying Pico's look

and feel, nano also implements some missing (or disabled by default) features in

Pico, such as "search and replace" and "go to line and column number".

EMACS(1)

NAME

emacs - GNU project Emacs

SYNOPSIS

emacs [ command-line switches ] [ files ... ]

DESCRIPTION

GNU Emacs is a version of Emacs, written by the author of the original (PDP-10) Emacs,

Richard Stallman. The user functionality of GNU Emacs encompasses everything

other editors do, and it is easily extensible since its editing commands are written in Lisp.

JOE(1)

NAME

joe - Joe's Own Editor

SYNTAX

joe [global-options] [ [local-options] filename ]...

jstar [global-options] [ [local-options] filename ]...

jmacs [global-options] [ [local-options] filename ]...

rjoe [global-options] [ [local-options] filename ]...

jpico [global-options] [ [local-options] filename ]...

DESCRIPTION

joe is a powerful ASCII-text screen editor. It has a "mode-less" user interface which

is similar to many user-friendly PC editors. Users of Micro-Pro's WordStar or

Borland's "Turbo" languages will feel at home. joe is a full featured UNIX

screen-editor though, and has many features for editing programs and text.

joe also emulates several other editors.

**Un fichier :**

SERVICES(5)

NOM

services - Liste des services réseau internet

DESCRIPTION

L'emplacement du fichier services est défini par \_PATH\_SERVICES dans <netdb.h>.

Il est habituellement défini à /etc/services

services est un fichier texte ASCII fournissant une correspondance entre des noms

textuels faciles à mémoriser pour les services et les numéros de ports qui leur sont

assignés, ainsi que les types de protocoles. Chaque programme en relation avec le réseau

peut consulter ce fichier pour obtenir le numéro de port (et le protocole) pour ses services.

**LABORATOIRE – Accès au système de fichiers – 5**

**Les exercices des séquences 7 à 11 s’enchaînent.**

**Il faut donc les accomplir dans l’ordre prévu.**

**Dans la distribution Debian GNU/Linux :**

Dans la suite, le nom de l’utilisateur ordinaire sera représenté par le mot « user ».

Connectez-vous dans un terminal

Affichez le manuel de la commande sort

man sort

Affichez le fichier /etc/services trié

sort /etc/services

Affichez le fichier /etc/services trié en n’affichant qu’une seule fois les lignes multiples

sort -u /etc/services

Affichez le fichier /etc/passwd trié par ordre décroissant

sort -r /etc/passwd

Triez le fichier /etc/passwd, avec enregistrement de la sortie dans ~/f\_comptes

sort -o f\_comptes /etc/passwd

Affichez le fichier ~/f1.txt trié

sort f1.txt

sort < f1.txt

« | » symbolise un « pipe », c’est-à-dire un « tube de communication » entre deux processus.

Affichez les 5 premières lignes du fichier /etc/services trié

sort /etc/services | head -5

sort < /etc/services | head -5

Affichez les 3 dernières lignes du fichier /etc/services trié

sort /etc/services | tail -3

sort < /etc/services | tail -3

Ecrivez le fichier /etc/services trié dans le fichier ~/Tri-serv en le créant

sort /etc/services > Tri-serv

sort < /etc/services > Tri-serv

Affichez triées, page par page, les lignes du fichier /etc/services qui ne commencent pas par « # »

grep -v '^#' /etc/services | sort | more

Affichez le tri de la concaténation des fichiers ~/f1.txt, ~/f2.txt et ~/f3.txt

sort f1.txt f2.txt f3.txt

cat f1.txt f2.txt f3.txt | sort

Ecrivez le tri de la concaténation des fichiers ~/f1.txt, ~/f2.txt et ~/f3.txt dans le fichier nommé : « f5.txt », en le créant dans ~

sort f1.txt f2.txt f3.txt > f5.txt

cat f1.txt f2.txt f3.txt | sort > f5.txt

Affichez les lignes contenant la chaîne « html » de la concaténation des fichiers ~/f1.txt, ~/f2.txt et ~/f3.txt

cat f1.txt f2.txt f3.txt | grep html

Affichez les lignes contenant la chaîne « html » du tri de la concaténation des fichiers ~/f1.txt, ~/f2.txt et ~/f3.txt

sort f1.txt f2.txt f3.txt | grep html

cat f1.txt f2.txt f3.txt | sort | grep html

cat f1.txt f2.txt f3.txt | grep html | sort

Créez dans ~ le fichier nommé : « f6.txt » et écrivez-y les lignes contenant la chaîne « html » de la concaténation des fichiers ~/f1.txt, ~/f2.txt et ~/f3.txt

cat f1.txt f2.txt f3.txt | grep html > f6.txt

Créez dans ~ le fichier nommé : « f7.txt » et écrivez-y les lignes contenant la chaîne « html » du tri de la concaténation des fichiers ~/f1.txt, ~/f2.txt et ~/f3.txt

sort f1.txt f2.txt f3.txt | grep html > f7.txt

cat f1.txt f2.txt f3.txt | sort | grep html > f7.txt

cat f1.txt f2.txt f3.txt | grep html | sort > f7.txt

Un éditeur est un programme de traitement de texte, pourvu de fonctionnalités plus ou moins complexes.

« gedit » est un éditeur en mode graphique, disponible dans l’interface graphique du système, via le menu « Activités », puis « Afficher les applications ». Il est pourvu de fonctionnalités complexes pour assurer la mise en évidence, selon un mode de coloration, des éléments de nature semblable (commandes, instructions, variables, etc.) d’un texte de script ou de code source écrit dans un langage de programmation.

En mode texte, « nano » est un éditeur installé par défaut. Par contre, « emacs » et « joe » sont des éditeurs à installer.

Affichez le manuel de la commande nano

man nano

Essayez les commandes principales de l’éditeur nano

Installez l’éditeur emacs

apt-get install emacs

aptitude install emacs

Affichez le manuel de la commande emacs

man emacs

Essayez les commandes principales de l’éditeur emacs

Installez l’éditeur joe

apt-get install joe

aptitude install joe

Affichez le manuel de la commande joe

man joe

Essayez les commandes principales de l’éditeur joe

Déconnectez-vous

exit