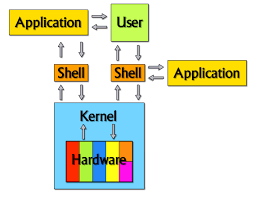
|  |  |  |  |
| --- | --- | --- | --- |
| **Logo_Lycée** | | **Systèmes d’exploitation**  **Shell Unix - BASH** | logo%20ac%20orl%E9ans%20toursDescription : Description : Description : pemDescription : Description : Description : pem |
| **TP** | **Plus loin…** |

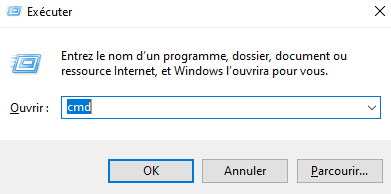
|  |  |
| --- | --- |
|  | **PRÉPARATION**  À l’aide du logiciel **FileZilla (https://urlr.me/8mDV7e), transférez** le dossier ***tp1plus*** dans votre dossier personnel sur le **Raspberry Pi.**  **Note : tp1plus** estsitué sur **le serveur NAS\_SIN** dans **TNSI/2\_Systèmes\_Exploitation !** |





**Connectez-vous** à votre compte sur le Raspberry Pi en suivant la procédure suivante :

1. **Touche Windows+R** pour ouvrir la commande "*Exécuter*". Entrez **cmd** pour lancer le **terminal** Windows.



**Correction**

1. Dans l’invite de commande de Windows, entrez :

**ssh** *votre\_identifiant\_de\_connexion****@****ip\_du\_Raspberry* puis votre\_*mot\_de\_passe*

***Objectif****: traiter un fichier texte à l’aide de la ligne de commande*



**1. Traitement des fichiers texte (complément)**

|  |  |
| --- | --- |
| ***Informations*** | Lorsque la taille d’un fichier dépasse les quelques dizaines de lignes, cela devient très compliqué de le consulter à l’aide de la commande **cat**. On utilise alors les commandes ***head*** et ***tail***. |

**Source** à consulter pour en savoir plus: https://bit.ly/3gxyiaU

|  |  |  |
| --- | --- | --- |
| **Nom** | **Description** | **Exemples** |
| **head** | Affiche les **premières lignes** d’un fichier  **-n** *valeur* : nombre de lignes affichées  **-c** *valeur* : nombre d’octets affichés | **head** /var/log/syslog*# Affiche les dix premières lignes par défaut*  **head** /var/log/syslog **-n** 2 *# Affiche les deux premières lignes* |
| **tail** | Affiche les **dernières lignes** d’un fichier  **-n** *valeur* : nombre de lignes affichées  **-c** *valeur* : nombre d’octets affichés | **tail** /var/log/syslog*# Affiche dix dernières lignes par défaut*  **tail** /var/log/syslog **-n** 2 *# Affiche les deux dernières lignes* |

|  |  |
| --- | --- |
| **Activité 1** | **Notez** ci-dessous la ligne de commande permettant d’afficher les **dix premières lignes** du fichier *actress.csv* situé dans le dossier *notes*.  */home/votredossierperso/tp1plus/imdb/elle/notes/actress.csv.*  **Remarques** : votre **répertoire courant** est la racine de votre dossier personnel. Notez la première ligne affichée*.* |

Commande : **head tp1plus/imdb/elle/notes/actress.csv** Résultat : **Ryder:Winona:89**

|  |  |
| --- | --- |
| **Activité 2** | **Notez** ci-dessous la ligne de commande permettant d’afficher les **quatre dernières lignes** du fichier *actress.csv* situé dans le dossier *notes*.  **Remarques** : votre **répertoire courant** est la racine de votre dossier personnel. Notez la dernière ligne affichée*.* |

Commande : **tail tp1plus/imdb/elle/notes/actress.csv -n 4** Résultat : **Gillan:Karen:99**

|  |  |
| --- | --- |
| ***Informations*** | La commande ***wc*** fournit des **méta-informations** sur un fichier ou sur l’entrée standard. La commande ***grep*** affiche toutes les lignes des fichiers donnés en argument, ou sur l’entrée standard, contenant une chaîne de caractères donnée en argument. |

|  |  |  |
| --- | --- | --- |
| **Nom** | **Description** | **Exemples** |
| **wc** | ***w****ord* ***c****ount* *fichier*1,…, *fichier*n affiche le nombre de lignes, de mots et d’octets contenus dans les fichiers dont les noms sont donnés en argument.  **-l** : nombre de lignes  **-w** : nombre de mots  **-c** : nombre d’octets | **wc** *rose-tatoo.txt* *# Affiche* 15 109 503 rosetatoo.txt  soit : 15 lignes 109mots 503 caractères |
| **grep** | ***g****lobal* ***r****egular* ***e****xpression* ***p****rint*. Recherche une chaîne de caractères dans des fichiers (ou depuis la console si aucun fichier n'est indiqué) | **grep** Elise prenom.txt # Affiche les lignes contenant le prénom Elise |

|  |  |
| --- | --- |
| **Activité 3** | **Notez** ci-dessous la ligne de commande permettant d’afficher le **nombre de lignes**, **mots** et **octets** du fichier *actress.csv* ainsi que le résultat renvoyé.  **Remarque** : votre **répertoire courant** est imdb. Utilisez un chemin relatif. |

Commande : **wc tp1plus/imdb/elle/notes/actress.csv**

Résultat : **72 72 1211 tp1plus/imdb/elle/notes/actress.csv, soit : 72 lignes 72mots 1211 caractères**

|  |  |
| --- | --- |
| **Activité 4** | **Notez** ci-dessous la ligne de commande permettant d’afficher les **lignes du fichier**  */home/votredossierperso/tp1plus/textes/slogan.txt*  qui contiennent le mot UNIX.  **Remarque** : votre **répertoire courant** est la racine de votre dossier personnel. Utilisez un chemin relatif. |

Commande : **grep UNIX tp1plus/textes/slogan.txt**

Résultat :

**There is nothing UNIX can’t buy *l n’y a rien qu’UNIX ne puisse acheter***

**UNIX or nothing *UNIX ou rien***

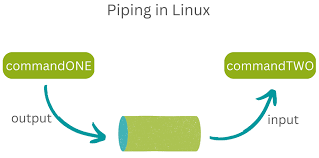
**We’re always low UNIX *Nous sommes toujours sous UNIX***

**Live free or die UNIX Vivez libre ou mourez UNIX**

|  |  |
| --- | --- |
| **Activité 5** | Notez ci-dessous la ligne de commande permettant d’afficher les **lignes du fichier**  */home/votredossierperso/tp1plus/imdb/elle/notes/actress.csv*  qui contiennent le mot Jennifer. Notez le nom des actrices.  **Remarque** : votre **répertoire courant** est imdb. Utilisez un chemin relatif. |

Commande : **grep Jennifer tp1plus/imdb/elle/notes/actress.csv** Résultat : **Connelly, Lawrence, Aniston**

**2. Branchement de commandes**



|  |  |
| --- | --- |
| ***Informations*** | Les commandes peuvent se brancher à la suite les unes des autres pour effectuer des traitements spécifiques. Le principe consiste à ce que la sortie d’une première commande soit transférée à l’entrée de la commande suivante, sans passer par un fichier intermédiaire. Un tube (**pipe** en anglais) est alors établi entre les 2 commandes. L’opérateur du tube est la barre verticale (**|**). |

Exemple : *lscpu* **|** *grep "MHz" # Affiche les détails des fréquences du CPU du Raspberry Pi*

|  |  |
| --- | --- |
| **Activité 6** | *On souhaite avoir dans un fichier, le manuel en ligne de la commande tail. Celle-ci peut être obtenue avec la commande* ***man tail****.*  **Créez** le fichier */home/votredossierperso/****tail.txt*** qui contiendra dans cet ordre :  1. Le manuel en ligne de la commande *tail*.  2. Une ligne de 3 tirets (on rappelle que le tiret '-' n'est pas le caractère souligné ou underscore '\_').  3. Le nombre de mots du manuel en ligne de la commande *tail.*  **Remarque : revoir les redirections en annexe. Notez** les lignes de commande page suivante. |

Commandes

**1. man tail > tail.txt**

**2. echo ‘---‘ >> tail.txt**

**3. man tail |wc -w >> tail.txt**

**3. Filtres simples**

|  |  |
| --- | --- |
| ***Informations*** | Un **filtre** est un moyen, comme son nom l’indique, de **simplifier**, **séparer**, **épurer** ou **clarifier** un flux de données. Un flux de données est souvent un **texte** ou une **chaîne de caractères**. Les commandes qui lisent et écrivent sur les E/S sont appelées des filtres. Les filtres sont souvent utilisés en branchement de commande (en réalisant un tube (**pipe**) avec l’aide du caractère **|**). |

**3.1 Découpage d’un fichier**

* **Les commandes *head* et *tail***

L’utilisation d’une commande ***head*** ou ***tail*** en tant que **filtre** se fait le plus souvent comme dans les exemples suivants :

cat /var/log/syslog **|** head -n 15 cat /var/log/syslog **|** tail -n 15

|  |  |
| --- | --- |
| **Activité 7** | *On souhaite voir les 24 premières lignes du magazine "elle" situé dans le fichier.*  */home/votredossierperso/tp1plus/imdb/elle/notes/actress.csv*.  **Notez** ci-dessous la ligne de commande permettant d’afficher la liste **faite par filtrage** des 24 premières lignes du fichier. |

Commande : **cat tp1plus/imdb/elle/notes/actress.csv |head -n 24**

**3.2 Modification de l’affichage d’un fichier à l’écran**

|  |  |
| --- | --- |
| ***Informations*** | **La commande *sort***  La commande **sort** permet d’afficher la sortie standard ou le contenu d’un fichier en **triant les lignes**. On peut choisir les **colonnes** qui seront les **clefs du tri**. Cette commande ne supprime pas d’information. En sortie toutes les colonnes seront présentes et le contenu de chaque ligne sera inchangé. Utilisée sans option, la commande sort trie le fichier donné en argument dans l’ordre alphabétique. |

|  |  |  |
| --- | --- | --- |
| **Nom** | **Description** | **Exemples** |
| **sort** | Affiche la sortie standard ou le contenu d’un fichier en triant les lignes  - **par défaut** : tri par ordre alphabétique sur la première colonne)  - **n** : tri numérique  - **k n** : tri sur la colonne n (**commence à 1**)  - **t** "*sep*" : change le séparateur *sep* de colonne (**espace** par **défaut**)  **- r** : inverse l’ordre de tri | **sort** tri.txt  **sort -n** tri.txt  **sort** **-nr** tri.txt  **sort -k 2** tri.txt |

|  |  |
| --- | --- |
| **Activité 8a** | **Exemples à tester** dans le dossier */home/votredossierperso/tp1plus/textes* :  cat tri.txt sort tri.txt cat tri.txt | sort cat tri.txt | sort -n |

|  |  |
| --- | --- |
| **Activité 8b** | **Notez** ci-dessous la ligne de commande permettant d’afficher le **contenu du fichier** *tri.txt* avec la troisième colonne triée par valeurs numériques et la quatrième triée par ordre alphabétique. Le résultat attendu est donné ci-dessous. |

21 Béatrice 1 AA

3 Belle 2 BB

11 Alice 2 CC

1 Bob 3 DD

Commande : **sort -k 4 tri.txt** ou **sort -nk3 tri.txt | sort -k4**

|  |  |
| --- | --- |
| **Activité 9** | *On souhaite classer les actrices par* ***ordre croissant*** *de* ***notes*** *(troisième colonne) pour le magazine "elle". Ces notes se trouvent dans le fichier :*  */home/votredossierperso/tp1plus/imdb/elle/notes/actress.csv*  *Pour cela, on propose d'utiliser la commande cat tp1plus/imdb/elle/notes/actress.csv pour afficher le contenu du fichier et se servir de la commande* ***sort*** *pour en modifier l'affichage de sortie.*  **Notez** ci-dessous la ligne de commande permettant d’afficher le fichier *actress.csv* trié. Notez les deux premières et les deux dernières lignes du fichier. |

Commande : **cat *tp1plus/imdb/*elle/notes/actress.csv | sort -nk 3 -t ':'** Résultat : **Garbo:Greta:1**

**Cusack:Joan:2**

**…**

**Gillan:Karen:99**

**Weisz:Rachel:99**

**3.3 Extraction d’information**

|  |  |
| --- | --- |
| ***Informations*** | **La commande *cut***  La commande **cut** permet d’**extraire des colonnes** d’un fichier.  ***ATTENTION****: le* ***séparateur par défaut*** *est la* ***tabulation****, ce qui est différent du caractère espace. Le séparateur peut être modifié avec l’utilisation de l’option -d.* |

|  |  |  |
| --- | --- | --- |
| **Nom** | **Description** | **Exemples** |
| **cut** | Extraction de colonnes.  **-c** n : extrait le caractère n  **-d** "*sep*" : change le séparateur *sep* de colonne (**tabulation** par **défaut**)  **-f** n : extrait la colonne n (**commence à 1**)  **-f** n,m : extrait les colonnes n et m  **-f** n-m : extrait les colonnes de n à m | **cut** **-c**1-2 *tri.txt* # Extrait les deux premiers caractères de chaque ligne du fichier tri.txt  **cut** **-f**2,4 *tri.txt* -d ' ' # Extrait les seconde et cinquième colonne. |

|  |  |
| --- | --- |
| **Activité 10a** | Testez les exemples du tableau ci-dessus. |

|  |  |
| --- | --- |
| **Activité 10b** | *On souhaite récupérer le prénom et le nom des actrices notées par le magazine "elle" sous la forme Nom:Prénom*.  **Notez** ci-dessous la ligne de commande permettant d’afficher cette liste à partir des informations du fichier *actress.csv*. |

Commande : **cat tp1plus/imdb/elle/notes/actress.csv | cut -f1,2 -d':'**

**3.4 Assemblage**

Consultez <https://urlr.me/4BtmY7> sur la commande **cat**.

|  |  |
| --- | --- |
| **Activité 11** | 1. Visualisez le contenu du fichier */home/votredossierperso/tp1plus/textes/lipsum.txt en numérotant uniquement les lignes non vides* 2. *Quelle est l’utilité de la commande paste ? (A chercher)* |

Commande : **cat -b *tp1plus/textes/lipsum.txt***

|  |  |  |
| --- | --- | --- |
| **paste** | Contrairement à la commande cut qui permet d’extraire des informations d’un fichier ou de l’entrée standard, la commande paste permet de fusionner les colonnes contenues dans plusieurs fichiers.  -d "*sep*" : change le séparateur *sep* de colonne (**tabulation** par **défaut**) | **paste** *low.txt* *high.txt*  # Affiche  1 4  2 5  3 6 |

**3.5 Modification du contenu**

Consultez <https://urlr.me/CQKzT7> sur la commande **tr.**

|  |  |
| --- | --- |
| **Activité 12** | Testez les exemples du lien. |

# **Annexes**

## Les répertoires (extrait)

|  |  |
| --- | --- |
| **Répertoire** | **Description** |
| **/** | **Le répertoire racine.** |
| **/bin** | Contient le noyau Linux et le chargeur de démarrage. |
| **/etc** | Contient tous les fichiers de configuration du système. |
| **/home** | **Chaque utilisateur dispose d’un répertoire personnel dans /home. C’est le seul endroit où il peut écrire des fichiers sans autorisation étendue.** |
| **/lib** | Contient des fichiers de bibliothèques partagées utilisées par les programmes du système principal. |
| **/var** | **On y trouve le contenu web (répertoire www), mais aussi les logs (journaux).** |

## Les commandes BASH (extrait)

**Source** : https://bit.ly/3gxyiaU

***Répertoires et fichiers***

|  |  |  |
| --- | --- | --- |
| **Nom** | **Description** | **Exemples** |
| ***cat*** *fichier* | *Affiche le contenu d’un fichier* | ***cat*** *fichier.txt* |
| ***cd*** *chemin/répertoire* | *Changement de répertoire* | ***cd*** */var/www*  ***cd*** *../ → répertoire précédent*  ***cd*** *ou* ***cd ~*** *→ compte utilisateur* |
| ***chmod*** *valeur fichier (rep)* | *Change les droits sur un fichier ou un répertoire* | ***chmod*** *770 fichier.txt* |
| ***cp*** *source… destination*  ***cp -r*** *source/ destination/* | *Copie de fichiers ou de répertoire* | ***cp*** *fichier.txt chemin/* |
| ***ls*** *répertoire* | *Affiche le contenu d’un répertoire, le répertoire courant si aucun argument* | ***ls*** *nomrépertoire*  ***ls*** *-l nomrep → avec détails*  *-h nomrep → affiche Mo ou Ko (human readable)*  *-a nomrep → affiche tous les fichiers (all)* |
| ***less*** *fichier* | *Affiche le contenu d’un fichier (défilement avec les touches flèche et page)* | *less fichier.txt* |
| ***mkdir*** *répertoire* | *Crée un répertoire* | ***mkdir*** *site* |
| ***mv*** *source… destination* | *Déplace ou renomme des fichiers* | ***mv*** *fichier1.txt … fichiern.txt chemin/*  ***mv*** *fichier.txt copie.txt* |
| ***nano*** | *Éditeur de texte* | ***nano*** *fichier.txt* |
| ***pwd*** | *Affiche le nom du répertoire courant.* |  |
| ***rm*** *fichier…*  ***rm -r*** *répertoire* | *Supprime définitivement des fichiers ou des dossiers* | ***rm*** *fichier.txt*  ***rm*** *-r nomrépertoire* |
| ***touch*** *fichier* | *Crée un fichier* | ***touch*** *fichier.txt* |
| ***tree*** | *Affiche l’arborescence d’un répertoire* |  |

***Divers***

|  |  |  |
| --- | --- | --- |
| **Nom** | **Description** | **Exemples** |
| **clear** | *Efface la console* |  |
| **exit** | *Sortie de la console* |  |
| **echo** *chaîne* | *Affiche la chaîne de caractères sur le canal de sortie standard.* | ***echo*** *"Bonjour"* |
| **history** | *Affiche l’historique des commandes entrées dans le terminal.* |  |
| ***id*** | *Affiche l’identifiant utilisateur UID et le groupe GUID.* |  |
| ***man*** *nomcommande* | *Manuel (aide en ligne)* | ***man*** *mkdir* |

**Lexique**

**BASH** : "**B**ourne **A**gain **SH**ell". Type de shell utilisé dans Debian (la version de Linux sur laquelle est basé Raspberry Pi OS).

**COMMAND LINE** : ligne spécifique sur laquelle on travaille dans l’environnement textuel.

**Console** : terminal physique avec un clavier. Les consoles étaient des terminaux connectés à un ordinateur central.

**Console virtuelle** : version virtuelle d’une console physique. Dans Linux, on accède à plusieurs consoles virtuelles en utilisant CTRL + ALT + les touches de fonction.

**SHELL** : interpréteur de ligne de commande. Entoure le noyau du système d’exploitation. On accède au noyau en passant par le shell (coquille). Le shell interprète les commandes textuelles et les transforme en code compréhensible par le noyau.

**Terminal** : programme utilisé pour accéder à l’invite de commande à partir du bureau.

## Ligne de commande et motif glob

glob

Pour exécuter une **commande** dans un terminal, on saisit un nom de commande suivi d’une liste d’arguments (**argi**) après l’invite de commande puis on valide avec la touche « Entrée ».

**nom\_de\_commande arg1 arg2 … argn**

|  |  |
| --- | --- |
| ***Informations*** | Les arguments peuvent contenir des caractères normaux, des caractères spéciaux : « **\*** », « **?** » et des ensembles de caractères entre crochets **[]**. Les expressions contenant ces caractères sont appelées ***motifs glob***.  «**\***» remplace n’importe quelle séquence de caractères  « **?** » remplace un caractère  Ce processus de substitution de motif est appelé **expansion de la ligne de commande**. |

*Exemples*

*- Photos/img*\**.jpg* permet de lister l’ensemble des fichiers du répertoire Photos commençant par img, suivi par n’importe quelle suite de caractères et dont l’extension est jpg.

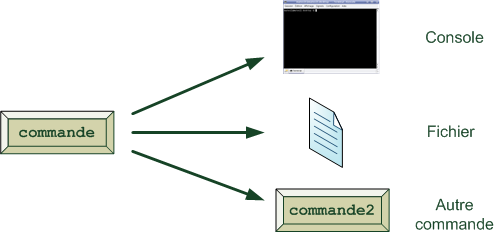
*- Photos/img*[2345]*.jpg* permet de lister l’ensemble des fichiers du répertoire Photos commençant par img, suivi par n’importe quelle suite de caractère à condition qu’un des caractères entre crochets soit présent et dont l’extension est jpg.

## Redirections

Les systèmes **POSIX** proposent à chaque programme trois fichiers spéciaux permettant de faire des entrées-sorties. L’***entrée standard***ou **stdin** (**st**andar**d** **in**put) est un fichier « virtuel » dans lequel un programme peut lire, mais pas écrire. Pour les programmes lancés dans un terminal, l’entrée standard est reliée au **clavier**. La ***sortie standard*** ou **stdout** (**st**andar**d** **ou**tput) correspond à un fichier relié à l’**affichage** dans le terminal. Le troisième fichier est la ***sortie d’erreur*** ou **stderr** (**st**andar**d** **err**or).

**Les opérateurs >, >>, 2> et |**

* L’opérateur **>** **redirige** la sortie standard d’un programme **vers un fichier**



*Fichiers spéciaux*

**[ATTENTION : si le fichier existe, il est écrasé.]**

*Exemple : ls -l* ***>*** *listerep.txt*

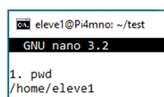
L’opérateur **>>** **redirige** à la fin du fichier pour ne pas écraser son contenu

*Exemple : ls -l* ***>>*** *listerep.txt*

* L’opérateur **2>** redirige les messages d’erreur

*Exemple : ls -l* ***2>*** *erreur.txt*

* L’opérateur **| (pipe)** effectue une **redirection** entre les commandes



*Exemple*

*Exemple : lscpu* **|** *grep "MHz" # Affiche les détails du CPU*

Exemple

Pour enregistrer dans un fichier ***log.txt*** le résultat de la commande ***pwd*** entrée au clavier et obtenir le résultat ci-contre, il suffit d’entrer la commande : **echo** *1. pwd* **>** *log.txt*.