

Documentation

Qu'est-ce que 'le Shell' :

En termes simples, le Shell est un programme qui prend les commandes du clavier et les donne au système d'exploitation pour qu'il les exécute. Autrefois, c'était la seule interface utilisateur disponible sur un système de type Unix tel que Linux. De nos jours, nous avons des interfaces utilisateur graphiques (GUI) en plus des interfaces de ligne de commande (CLI) telles que le Shell.

Sur la plupart des systèmes Linux, un programme appelé Bash (qui signifie Bourne Again SHell, une version améliorée du programme shell Unix original sh, écrit par Steve Bourne) agit comme programme Shell. En outre Bash, il existe d'autres programmes Shell disponibles pour les systèmes Linux. Ceux-ci incluent : ksh, tcsh et zsh.

Le terminal ?! :

Le terminal est un shell. On en retrouve un (ou plusieurs) dans tous les systèmes Linux ; le terminal est un programme, un émulateur de console qui nous permet de lancer des commandes en texte depuis l'interface graphique, c'est une interface de ligne de commande (CLI) par opposition à l'interface graphique utilisateur (GUI: graphical user interface) qui nous permet de passer des commandes plutôt qu'en cliquant sur des icônes de programmes, par exemple. La console, elle, est l'écran noir avec du texte qui défile rapidement lorsque l'on démarre l'ordinateur. Dites-vous que tout ce qui se passe dans votre système GNU/Linux est écrit en texte (script) quelque part dans le système de fichier. Dans Linux, tout est fichier, certains sont des fichiers exécutable (des scripts) et d'autres non. Les scripts prédéfinis permettent, entre autres, d'automatiser des opérations, des tâches, des processus, etc. Vous pouvez contrôler toutes les composantes de votre système Linux par le terminal

Le bash :

Bash (acronyme de Bourne-Again shell) est un interpréteur en ligne de commande de type script. C'est le shell Unix du projet GNU.

Fondé sur le Bourne shell, Bash lui apporte de nombreuses améliorations, provenant notamment du Korn shell et du C shell. Bash est un logiciel libre publié sous licence publique générale GNU. Il est l'interprète par défaut sur de nombreux Unix libres, notamment sur les systèmes GNU/Linux. C'était aussi le shell par défaut de Mac OS X,

Sommaire :

1 / Job 1

- .. La commande ls
- .. Application

2/ Job 2

- .. La commande cat
- .. Application

3/ job 3

- .. Le apt-get
- .. Application

4/ job 4

- .. La commande less
- .. La commande touch
- .. La commande chmoud
- .. Application

5/ Job 5

- .. Alias
- .. L'éditeur Vim
- .. Application

6/ Job 6

- .. Tar
- .. Application

7 / job 7

- .. Echo
- .. wc,grep
- .. Application

Job 1 :

Ls est l'une des commandes de base que tout utilisateur Linux devrait savoir.

La commande ls répertorie les fichiers et les répertoires dans le système de fichiers et affiche des informations détaillées sur eux.

Il fait partie du package GNU Core Utilities qui est installé sur toutes les distributions Linux.

1/ Pour afficher les manuels de n'importe quelle commande on utilise la commande (man commande souhaitée) donc pour avoir le manuel de ls on utilise la commande `Man ls`

2/ Pour afficher les fichiers cachés du home de notre utilisateur on utilise la `commande Ls -a /home`

3/ Pour les fichiers cachés plus les droits on utilise la commande `Ls -all`

Job 2 :

La commande `cat` de Linux est l'une des commandes les plus utiles que vous pouvez apprendre. Elle tire son nom du mot concaténer (concatenate en anglais) et vous permet de créer, de fusionner ou d'imprimer des fichiers dans l'écran de résultat standard ou vers un autre fichier et bien plus encore.

1/ La commande qui nous permet de lire un fichier est « cat » donc on utilise `cat .bashrc`

2/ Pour lister les 20 premières commandes on utilisera la commande `Head` pour les 20 premières c'est `Head -20 /etc/$USER/.bashrc` et pour les 10 premiers : `Head -10 /etc/$USER/.bashrc`. Le `/etc/$USER/.bashrc` est le chemin pour trouver le `.bashrc`

3/ Pour lister les 20 dernières commandes on utilisera la commande `tail` pour les 20 dernières c'est `tail -20 /etc/$USER/.bashrc` et pour les 10 derniers : `tail -10 /etc/$USER/.bashrc`. Le `/etc/$USER/.bashrc` est le chemin pour trouver le `.bashrc`

Job 3 :

Apt-get est un outil logiciel à utiliser en ligne de commande. Il permet d'effectuer l'installation et la désinstallation de paquets en provenance d'un dépôt APT.

Dans ce job on va installer le paquet 'cmatrix' pour cela on utilisera les deux commandes :

```
sudo apt-get update
```

```
sudo apt-get install cmatrix
```

Pour lancer le paquet on entre `cmatrix` dans le terminal

Pour mettre à jour le gestionnaire des paquets :

```
Sudo apt upgrade
```

Pour mettre à jour les logiciels :

```
Sudo apt Uptade
```

Pour redémarrer sa machine

`Reboot` ou `Shutdown -r now`

Pour éteindre sa machine

```
Shutdown -h now
```

Job 4 :

`Less` est une commande et utilitaire de ligne de commande qui affiche le contenu d'un fichier ou d'une sortie de commande, une page à la fois. Il est similaire à `more`, mais a des fonctionnalités plus avancées et vous permet de naviguer à la fois vers l'avant et vers l'arrière dans le fichier

La commande `touch` est une commande de base Linux qui vous permet de mettre à jour les horodatages sur les fichiers et les répertoires existants, ainsi que la création de nouveaux fichiers vides

La commande `chmod` permet de configurer l'accès aux fichiers et aux répertoires. Pour créer un groupe :

`Groupadd Plateformeurs`

On peut toujours vérifier avec `less /etc/group`

Pour ajouter des User :

`Useradd user1 et user2`

Pour ajouter un user dans un groupe :

`Usermod -a -G plateformeurs User2`

Pour vérifier `less /etc/group`

Puis créer le fichier `droits.txt`

`Touch droits.txt`

`Cp users.txt groupes.txt`

Pour vérifier `cat groupe.txt`

Pour les droits du fichier pour les user et le groupe :

`Sudo chmod g=r droits.txt`

`Sudo chmod g+w groups.txt`

Job 5 :

Les alias sont des substitutions abrégées de commandes répétitives et/ou longues à taper dans la console.

Pour créer un alias il faut taper la commande :

`Alias la='ls -la'`

`Alias update='apt-get update'`

`Alias upgrade='apt-get upgrade'`

Installer l'éditeur de texte Vim pour vérifier (Vim est un éditeur de texte pour le terminal sous GNU/Linux. Il a une stabilité exemplaire et ne cesse d'être amélioré. Malgré son austérité, c'est un outil très puissant)

`Apt-get install vim`

`vim /root/.bashrc`

Changer le propriétaire du fichier

`Sudo chmod u=wr User`

Pour ajouter à notre Le PATH

`Export Path=$Path = « path:/home/'user'/bureau`

Le Path est utilisé sur Linux pour contenir des chemins vers des programmes. Ces chemins permettent d'exécuter directement une commande dans le terminal. Si vous exécutez cette commande directement dans le terminal, le répertoire sera ajouté, mais uniquement pour la durée de la session.

Job 6 :

La commande `tar` est aussi capable d'extraire (décompresser) une archive.

```
tar -xzf GhostintheShell.tar.gz
```

Il s'agit en fait de la même commande que celle pour créer une archive, mais l'on remplace l'option `-c` par `-x` qui indique à `tar` d'extraire une archive au lieu d'en créer une.

Job 7 :

La commande `echo` est l'une des commandes les plus basiques et les plus fréquemment utilisée sous Linux. Les arguments passés à `echo` sont affichés sur la sortie standard.

`Echo` est couramment utilisé dans les scripts shell (`bash`, `Zsh` et `Ksh`) pour afficher un message ou afficher les résultats d'autres commandes.

Pour le job 7 on taper toutes les instructions demandées en une seule ligne :

```
Echo 'je suis votrefichiers texte' >une_commande.txt |wc -l /etc/apt/sources.list > nb_ligne.txt | cat  
/etc/apt/sources.list | cp /etc/apt/sources.list $USER/save_sources.txt|grep --include=*alias --include=*. -rnw
```

La commande `wc` de Linux vous permet de compter le nombre de lignes, de mots, de caractères et d'octets de chaque fichier donné ou d'entrée standard et d'imprimer le résultat.

La commande `grep` cherche la chaîne de caractères « <expression> » à l'intérieur des fichiers ou des répertoires spécifiés et affiche les lignes correspondantes.

JOB BONUS ! :

SSH, également connu sous le nom de Secure Shell, est un protocole réseau qui vous permet d'échanger des données à l'aide d'un canal très sécurisé entre deux appareils en réseau différents. SSH est l'une des méthodes sécurisées pour accéder aux systèmes distants.

À l'aide de SSH, un utilisateur peut se connecter à l'invite de commande du shell Linux et exécuter facilement des commandes. Cela donne une sensation similaire, comme si vous étiez assis juste devant votre serveur Linux et que vous tapiez des commandes sur la console. Il y a quelques années, les utilisateurs utilisaient Telnet pour accéder aux serveurs. Cependant, étant donné que telnet n'est pas très sécurisé et qu'Internet a commencé à s'ouvrir au public mondial, les serveurs connectés à Internet nécessitaient un moyen plus sécurisé d'accéder au shell.

SSH offre un cryptage pour le transfert de données qui empêche les pirates et les attaquants de pirater le mot de passe de votre serveur et les informations utilisateur. Un autre avantage de l'utilisation de SSH est qu'il vous permet de tunneler d'autres protocoles réseau.

Par exemple, si vous souhaitez transférer des fichiers en toute sécurité, vous pouvez utiliser le SSH afin de crypter les transferts FTP. Cela peut être fait avec n'importe quel type de connexion, comme VNC ou Samba, les deux sont bons pour de telles tâches.

En utilisant Secure SHell, l'utilisateur peut gérer son serveur dédié à distance, il peut également surveiller les journaux du serveur, installer / désinstaller des logiciels et des applications, il peut même démarrer ou arrêter des services, etc...

SSH est capable de reconnaître les commandes shell Unix normales, que vous pouvez utiliser pour vous connecter en tant que root sur votre serveur afin de gérer votre serveur avec tous les droits.

Depuis que SSH a été présenté aux utilisateurs de Linux, il est devenu la méthode la meilleure et la plus sécurisée pour gérer l'environnement de serveur dédié. Il est plus efficace et utile d'effectuer diverses tâches qui ne peuvent pas être effectuées avec un panneau de contrôle basé sur le Web. Par conséquent, SSH est le meilleur protocole réseau pour gérer votre serveur dédié.

(source :eukhost)

Les deux protocoles le TCP ET UDP

TCP est en mode orienté connexion et fiable, tandis que UDP est en mode non-connecté et peu fiable.

TCP nécessite plus de traitement au niveau de l'interface réseau, ce qui n'est pas le cas en UDP.

TCP utilise, handshake 3 voies, le contrôle de la congestion, le contrôle de flux et d'autres mécanismes pour assurer la transmission fiable.

UDP est principalement utilisé dans les cas où le retard de paquet est plus sérieux que la perte de paquet. (Applications en temps réel)

Différence entre SSH et Telnet

Pour changer de port `sudo nano/etc/ssh/sshd_config`

Rechercher le port 22 et le remplacer