Commandes et fichiers d'administration systèmes et réseaux sous Linux

Les commandes précédées du caractère # doivent être lancées avec le compte root. Celles précédées du caractère $ peuvent être lancées par n'importe quel utilisateur.
Les mots entre chevrons (les caractères < et >) doivent être remplacés par des valeurs spécifiques. Ceux entre crochets sont optionnels.

Commandes de configuration des interfaces réseau

`$ ifconfig`
Affiche les informations sur toutes les interfaces.
`$ ifconfig <int>`
Affiche les informations sur l'interface int.
`# ifconfig <int> <adresse> netmask <masque>`
Active l'interface int statiquement en lui donnant l'adresse IP adresse/masque.
`# ifup <int>`
Active l'interface int à partir des informations de son fichier de configuration.
`# ifdown <int>`
Désactive l'interface int.

Fichiers de configuration des interfaces

Toute interface `int` a un fichier de configuration `ifcfg-<int>` se trouvant dans le répertoire `/etc/sysconfig/network-scripts`. Ce fichier est lu à l'activation de l'interface via la commande ifup et décrit les paramètres d'activation de l'interface. Chaque ligne de ce fichier définit un de ces paramètres et est de la forme `PARAMETRE=VALEUR`. Le fichier doit au moins définir les paramètres suivants:

`DEVICE` ⇒ nom de l'interface
`ONBOOT` ⇒ `yes` si l'interface est activée au démarrage du système, `no` sinon
`BOOTPROTO` ⇒ décrit le mode d'activation de l'interface. La valeur peut être `static` si l'interface est activée statiquement, ou `dhcp` si l'interface est activée dynamiquement par l'appel à un serveur DHCP.

Si l'interface est activée statiquement les deux paramètres suivants peuvent être fournis:

`IPADDR` ⇒ adresse IP associée à l'interface
`NETMASK` ⇒ masque de réseau associé à l'interface

Commandes de tests réseau

`$ ping [<options>] <machine>`
Test d'accessibilité d'une machine distante. Envoie des messages de demande d'écho à la machine distante qui répond à chaque demande par un écho. Options courantes:

`-c <N>` ⇒ envoie N demande(s) d'écho (sinon ping ne s'arrête pas)
`-i <S>` ⇒ attend S seconde(s) entre deux envois (1 seconde par défaut)
`-b` ⇒ nécessaire si l'adresse "pinguée" est une adresse de diffusion

`$ traceroute <machine>`
Affiche la route vers la machine, c'est-à-dire la liste des routeurs sur le chemin qui mène à la machine destinaire.

Commandes de configuration de la table de routage

`$ route -n`
Affiche la table de routage.
`# route add default gw <routeur> dev <int>`
Ajoute à la table de routage une route par défaut de la forme (0.0.0.0, 0.0.0.0, routeur, int).
`# route add -net <ip> netmask <masque> gw <routeur> dev <int>`
Ajoute à la table de routage une route en remise indirecte de la forme (ip, masque, routeur, int).
`# route add -net <ip> netmask <masque> dev <int>`
Ajoute à la table de routage une route en remise directe de la forme (ip, masque, -, int).
`# route del ...`
Supprime une route de la table de routage. Il faut utiliser la commande ayant ajouté la route en remplaçant le mot add par del.

Commandes de connexion et copie de fichiers à distance

Pour accéder à une machine distante (avec ssh ou scp) le service sshd doit être démarré sur cette machine.
`$ ssh <utilisateur>@<machine>`
Ouvre sur la machine distante une connexion SSH avec le compte utilisateur indiqué.
`$ scp [<options>] <source> <destination>`
Copie de fichiers distante. La source comme la destination peuvent être des fichiers se trouvant sur la machine ou sur une machine distante. Pour désigner un fichier sur une machine distante, on utilise la syntaxe suivante:

`<utilisateur>@<machine>:<fichier>`
La copie se fait alors en utilisant le compte de l'utilisateur indiqué sur la machine distante. fichier désigne le chemin complet du fichier sur la machine distante.

Option courante:

`-r` ⇒ pour la copie de répertoire

Autres commandes réseau

`$ wget <url>`
Récupère dans le terminal le fichier à l'URL indiquée. Une URL est une adresse de la forme `http://www.site.fr/fichier`.
`$ host -t <type> <nom> [<serveur-dns>]`
Récupère auprès du serveur DNS indiqué (ou, à défaut, celui du fichier `/etc/resolv.conf`) un enregistrement concernant un nom. Les principaux types d'enregistrement sont SOA (Start Of Authority, description générale de la zone), A (une adresse IP), NS (NameServer, le nom du serveur DNS d'une zone), MX (Mail eXchanger, le serveur de messagerie d'une zone).

Commandes de gestion des services

`# systemctl <action> <nom-du-service>.service`
Effectue l'action indiquée sur le service. Les actions principales sont:

- `start` ⇒ démarre le service
- `stop` ⇒ arrête le service
- `status` ⇒ affiche des informations sur le service (démarré/arrêté, pid, ...)

Services courants: sshd, dhcpd, named (serveur de noms DNS), nfs-server, crond.

Fichiers de configuration des services

`/etc/dhcp/dhcpd.conf`	configuration du service DHCP
`/etc/named.conf`	configuration du service DNS
`/etc/exports`	répertoires exportés par le service NFS

Commandes de gestion des groupes et utilisateurs

`# useradd <utilisateur>`
Crée un utilisateur.
`# userdel <utilisateur>`
Supprime un utilisateur.
`# usermod [<options>] <utilisateur>`
Modifie les données d'un utilisateur. (Voir le manuel pour la liste des options.)
`# groupadd <nom>`
Crée un groupe.
`# groupdel <nom>`
Supprime un groupe.
`# groupmod [<options>] <nom>`
Modifie les données d'un goupe. (Voir le manuel pour la liste des options.)
`# passwd <utilisateur>`
Modifie le mot de passe d'un utilisateur.
`$ passwd`
Modifie le mot de passe de l'utilisateur connecté.

Fichiers des groupes et utilisateurs

`/etc/passwd`	Liste des utilisateurs avec leurs informations
`/etc/shadow`	Mots de passe (cryptés) des utilisateurs
`/etc/group`	Liste des groupes avec leurs informations
`/etc/gshadow`	Mots de passe (cryptés) des groupes

Commandes de gestion des processus

`$ kill <pid>`
Envoie un signal de terminaison à un processus identifié par son pid (process id).
`$ ps aux`
Affiche la liste de tous les processus en cours d'exécution avec des informations détaillées pour chaque processus (pid, consommation mémoire, heure de lancement, ...).
`$ top`
Affiche une vue en temps réel, rafraîchie toutes les secondes, de tous les processus avec des informations détaillées.

Commandes de programmation de tâches

`$ at <date-et-heure>`
Lance la programmation d'une tâche à la date et l'heure indiquées. at affiche ensuite l'identifiant de la tâche créée.
`$ atq`
Affiche la liste des tâches programmées avec at et en attente d'être exécutées.
`$ atrm <identifiant-de-tache>`
Annule une tâche programmée précédemment avec at.
`$ crontab <action>`
Programmation de tâches exécutées périodiquement. Les actions principales sont:

- `-e` ⇒ édite les règles cron présentes
- `-l` ⇒ affiche toutes les règles cron présentes
- `-r` ⇒ supprime toutes les règles existantes

Syntaxe des règles cron:
`<minute> <heure> <jour-du-mois> <mois> <jour-de-la-semaine> <commande>`
Meta-caractères pour exprimer la fréquence:

- `*` ⇒ toutes les minutes (ou heures, ou jours, ...)
- `*/N` ⇒ toutes les N minutes (ou heures, ou jours, ...)
- `X-Y` ⇒ X à Y
- `X,Y` ⇒ X et Y

Fichiers liés à la résolution DNS

`/etc/hostname`	nom de la machine
`/etc/hosts`	associations nom↔adresse IP
`/etc/nsswitch.conf`	Méthode de résolution de noms utilisée par le système
`/etc/resolv.conf`	Informations DNS (nom du serveur DNS utilisé par le système, nom de domaine de la machine, ...)