Configurer sa machine en tant que serveur

Maurice Doison <mdoison@linux62.org>

23 mars 2006

Qu'est ce qu'un serveur ?

* Un serveur est un ordinateur qui a pour but de partager des ressources.
* Une même machine peut héberger plusieurs types de services.
* Les serveurs sont généralement des machines dédiées.
* Des ordinateurs de bureau peuvent également être utilisés en tant que serveurs.
* Cette conférence donne des pistes pour faire d'une machine de bureau un serveur performant.

Prérequis

* La conférence s'appuie sur la distribution GNU/Linux Ubuntu.
* Les procédures d'installation changent peu pour d'autres distributions de GNU/Linux.
* Une partie des programmes énoncés fonctionnent sous MS-Windows, mais leur configuration est laborieuse et l'aide difficile à trouver.

Prérequis (2)

* Les programmes peuvent être installés en les téléchargeant directement sur le site officiel, mais la procédure (compilation et installation) est délicate pour les débutants.
* Sous GNU/Linux, un programme serveur se présente sous forme de *daemon*
* Un daemon est un programme qui tourne en arrière plan, sans avoir aucune interaction avec l'utilisateur

Prérequis (3)

* Les daemons se lancent automatiquement au démarrage de l'ordinateur.
* Sous Ubuntu, un daemon se lance et s'arrête avec une commande du type :

```
/etc/init.d/nom_du_daemon restart
```

* Un serveur réseau est toujours associé à un (ou plusieurs) port(s).
* Le port permet aux clients de distinguer les différents services d'un serveur.

Les points abordés

* Quelle machine, quelle configuration ?
* Etablir un serveur pour transférer des fichiers.
* Contrôler un ordinateur à distance.
* Mettre en place une suite web (Apache + PHP + MySQL)
* Régler le problème de l'ip dynamique.
* Firewall et sécurité.
* Organiser un réseau domestique.

Quelle machine, quelle configuration ?

* La plupart des services demandent peu de ressources.
* La machine doit pouvoir tenir sur la durée.
* Accès direct à Internet conseillé.
* Système GNU/Linux conseillé

Transférer des fichiers

Plusieurs solutions envisageables :

- HTTP (accès en lecture seule)
- FTP (accès en lecture/écriture)
- SSH (accès sécurisé en lecture/écriture)
- SAMBA (accès direct au disque dur) (non couvert par cette conférence)

Transfert de fichier (HTTP)

* Le protocole HTTP : protocole ‘web’.
* Taille de transfert limitée.
* Transfert unidirectionnel : du serveur vers le client.
* Accès anonyme.
* Le serveur HTTP libre le plus connu : **Apache**.

Installation et configuration d'Apache

* Site officiel :

  `http://www.apache.org`
* Package pour la plupart des distribution GNU/Linux
* Pour Ubuntu :

```
sudo aptitude install apache2
```

* Activation automatique du service web après installation.
* Répertoire 'racine' dans */var/www*
* Accès web du contenu de */var/www* par l'addresse *http ://monip*

Activation des répertoires utilisateurs

* Les répertoires utilisateurs pour apache fournissent un support web pour tous les utilisateurs.
* Tout fichier dans votre répertoire *~/public_html* sera disponnible par l'adresse *http ://monip/~nom_utilisateur*
* Permet de diffuser rapidement n'importe quel type de fichier, simplement en copiant le fichier dans *~/public_html*.

Activation des répertoires utilisateurs (2)

* Dans le fichier /etc/apache2/apache2.conf, ajouter à la fin la ligne :

```
UserDir public_html
```

* Redémarer apache :

```
sudo /etc/init.d/apache2 restart
```

* Conseil : mettre une page par défaut nommée **index.html** afin d'éviter que n'importe qui ait accès à la liste des fichiers.

Transfert de fichiers (FTP)

* FTP (File Transfert Protocol) fait pour transférer des fichiers
* Possibilité d'envoi et de réception
* Exemples de serveurs FTP : proftpd, pureftp
* Port standard du FTP : 21 TCP (et 20 TCP)

Configuration et utilisation de Proftpd

* Site officiel :

  `http://www.proftpd.org`

* Sous Ubuntu :

```
sudo aptitude install proftpd
```

* Démarage automatique du serveur.
* La connexion se fait grâce au login et au mot de passe de l'utilisateur.
* Pour ajouter un compte au serveur ftp, il suffit d'ajouter un utilisateur au système en spécifiant le répertoire personnel adéquat.

Activation de l'autentification anonyme

* Il est possible d'activer un compte anonyme
* Ce compte permet de se connecter au FTP sans login ni mot de passe
* **ATTENTION** à la sécurité !
* La configuration se fait dans le fichier */etc/proftpd.conf*
* On s'y connecte en utilisant le login *anonymous* et en n'entrant aucun mot de passe.

Activation de l'autentification anonyme

* Ajouter, à la fin du fichier :

```
<Anonymous /home/ftp>
MaxClients 5 "Nombre de clients maximum atteint
    : 5"
User ftp
Group ftp
<Limit WRITE>
DenyAll
</Limit>
</Anonymous>
```

* Redémarer le serveur ftp :

```
sudo /etc/init.d proftpd restart
```

Transfert de fichiers sécurisé par SSH

* SSH est principalement utilisé pour les connexions à distance
* Il permet aussi des transferts de fichiers
* Les données transférées sont cryptées : protocole sécurisé
* Fonctionne en émission et en réception
* Port standard de SSH : 22 TCP.

Installer et configurer OpenSSH

* Site officiel : *http ://www.openssh.com*
* Installer OpenSSH sur ubuntu :

```
sudo aptitude install openssh-server
```

* La machine cliente doit posséder un client SSH supportant le transfert de fichiers :
  - Sous GNU/Linux : openssh-client
  - Sous Windows : pscp (putty) sur `http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html`

Transférer des fichiers

* Pour transférer des fichiers :

```
scp user@addr_src:chemin user@addr_cible:chemin
```

* Si l'utilisateur (source ou cible) est le même que l'utilisateur courant, il n'a pas besoin d'apparaitre.
* Si l'addresse (source ou cible) est l'addresse locale (*localhost*), elle n'a pas besoin d'apparaitre.

ex : pour transférer un fichier *fichier.ext* vers le répertoire *documents* dans l'espace personnel de *toto* sur la machine *ailleurs* :

```
scp fichier.ext toto@ailleurs:~/documents
```

Le contrôle à distance avec SSH

* SSH est une solution permettant de contrôler à distance votre ordinateur.
* SSH est crypté, donc relativement sûr.
* L'utilisation d'SSH pour du mode texte est facile et rapide.
* SSH utilise le port 22 TCP

Le contrôle à distance avec SSH

* Voir au dessus (**Installer et configurer OpenSSH**) pour l'installation
* Une connexion à distance à SSH se fait par la commande :

```
ssh login@host
```

puis en saisissant son mot de passe
NOTE : *login* peut être ignoré s'il est le même que celui de votre session actuelle.

* Vous disposez alors d'un shell exactement comme si vous utilisiez un terminal devant votre ordinateur
* Vous ne pouvez pas lancer d'application utilisant l'interface graphique

Ouverture de session graphique à distance avec XDMCP

* XDMCP permet d'ouvrir une session graphique depuis un autre ordinateur, exactement comme si vous étiez en face de votre machine
* Utilise le port 177 UDP (paramétrable)
* Pour activer XDMCP sous Gnome sous Ubuntu, allez dans *Système*, *Administration* puis *Configurer le gestionnaire de connexion*

Ouverture de session graphique à distance avec XDMCP (2)

* Cliquez sur l'onglet *Distante* et indiquez un *Style* **Simple**.
* Fermez votre session, puis réouvrez la pour mettre à jour les modifications.

Se connecter à distance à XDMCP

* Depuis GDM (le gestionnaire de connexion), faite *Option* puis *Connexion à distance*
* Entez l'addresse IP de la machine distante, puis validez.
* Vous avez maintenant accès au gestionnaire de connexion de la machine distante, vous pouvez vous connecter.
* Attention : les performance sont médiocres via une connexion internet (réduisez la résolution !).

Mise en place d'une suite Web

* Ce que nous couvrirons :
  - Installer et configurer PHP pour Apache
  - Installer et configurer Mysql
  - Installer et utiliser PHPMyAdmin
* Ce que nous ne couvrirons pas :
  - Création de sous-domaines
  - *VirtualHost* d'Apache
  - Installation de PostgreSQL

Installer et configurer PHP

* Site officiel :

  `http://www.php.net`
* Apache doit être installé (voir plus haut)
* Installer PHP pour Apache :

```
sudo aptitude install libapache2-mod-php5
```

* Pour tester PHP, ajouter un fichier */var/www/index.php* (supprimer les éventuels autres fichiers *index.**)

Installer et configurer PHP (2)

* Editer ce fichier et mettez y la ligne :

```
<?phpinfo()?>
```

* Avec un navigateur, aller à l'addresse

  `http://localhost`

  .

* Si les informations sur PHP s'affichent, le serveur PHP est fonctionnel.

Installer et configurer MySQL

* Site officiel :

  `http://www-fr.mysql.com`
* Port standard de MySQL : 3306 TCP
* C'est PHP qui se connecte à MySQL, et pas directement l'utilisateur
* Inutile d'ouvrir le port MySQL sur le firewall !

Installer et configurer MySQL (2)

* Installer Mysql sur Ubuntu :

```
sudo aptitude install mysql-server mysql-client
    php5-mysql
```

* Définir votre mot de passe mysql pour le superutilisateur :

```
sudo mysqladmin password nouveau_mot_de_passe
```

PHPMyAdmin

* Interface web pour administrer MySQL
* Peut être utilisé par les administrateurs aussi bien que par les mainteneurs des bases de données
* Sous Ubuntu :

```
sudo aptitude install phpmyadmin
```

PHPMyAdmin (2)

* Debconf va poser quelques questions :
  - **Quels serveurs web sont à reconfigurer ?**
    Cocher 'Apache2' puis aller sur 'Ok'
  - **Faut-il redémarer Apache2 immédiatement ?**
    Répondre oui
* L'accès à MySQL se fait en tappant dans votre navigateur l'addresse :

  `http://localhost/phpmyadmin`

Créer un utilisateur

* Il est dangeureux de passer par l'utilisateur **root** pour manipuler une base de données
* PHPMyAdmin permet d'ajouter des utilisateurs à la base de données
  - Connectez vous à PHPMyAdmin avec l'utilisateur **root**
  - Aller dans l'onglet **Privilèges**
  - Cliquez sur **Ajouter un utilisateur**
  - Entrez un nom d'utilisateur, un mot de passer à confirmer.
  - Accordez toutes les permissions des sections *Données* et *Structure*
* Laissez le reste à vide et cliquez sur **Exécuter**

Créer une base de données

* Se déconnecter de **root** et se connecter avec le nouvel utilisateur
* Aller dans la section **Bases de données**, puis donnez un nom à votre base dans **Créer une base de données**
* Les informations necéssaires à PHP pour se connecter à votre base de données sont :
  - **Server :** localhost
  - **Username :** nouvel utilisateur créé
  - **Password :** son mot de passe

Vous disposez à présent d'un serveur web complet !

* Utilisation de PHP à volonté
* Connexion possible entre PHP et Mysql
* PHPMyAdmin disponnible pour organiser les bases de données
* Possibilité de créer plusieurs sites en créant des répertoires
* Site web disponnible de l'exterieur à l'addresse `http://votre-ip-internet`

L'IP dynamique

* La plupart des fournisseurs d'accès donnent une IP différente à chaque connexion.
* Impossible pour une tiers personne de retrouver d'un jour sur l'autre un serveur avec une IP dynamique.
* Il faut passer par un nom de domaine mis à jour très régulièrement

Une solution : No-ip.com

* Le site

  `http://www.no-ip.com`

  propose d'associer votre adresse ip à un nom de domaine gratuit mis à jour très régulièrement.
* Créez un compte sur le site.
* Identifiez-vous, puis créez un nom de domaine :
  - Dans le menu de gauche, cliquez sur **Add**
  - Choisissez un nom pour votre domaine, puis l'extension associée
  - Laissez le reste aux valeurs par défaut et confirmez
  - Allez dans **Manage Groups** puis dans **Add a Group**
  - Nommez votre groupe, confirmez, puis ajoutez votre nom de domaine créé à votre groupe

Mise à jour du nom de domaine

* No-ip.com propose un utilitaire pour mettre à jour automatiquement votre nom de domaine avec votre ip actuelle
* Il est disponnible soit directement sur le site officiel, soit avec certaines distributions GNU/Linux sous forme de package
* Le client fonctionne aussi bien sous GNU/Linux que sous MS-Windows
* Sous Ubuntu, faites :

```
sudo aptitude install no-ip
sudo no-ip -C
```

Mise à jour du nom de domaine (2)

* Puis entrez votre login no-ip (email) et votre mot de passe.
* Indiquez le temps (en minutes) désiré entre deux mises à jour, puis répondez 'non' à la dernière question (désirez vous lancer un programme lorsque la mise à jour est effectuée).
* Pour finir, lancez le serveur :

```
sudo /etc/init.d/no-ip start
```

* Votre nom de domaine se mettra automatiquement et régulièrement à jour !

La sécurité

* Plus une machine fournit de services différents, plus elle est susceptible d'être attaquée.
* Une attaque provient généralement d'un système automatisé programmé pour détecter les failles de sécurité.
* La sécurité vient avant tout des précautions prises par l'utilisateur, et en second lieu de programmes de sécuriété (firewall)

La sécurité (2)

* Quelques règles sont à connaitre afin d'accroître la sécurité de votre système :
  - Travaillez le moins possible avec l'utilisateur système (root).
  - Choisissez des mots de passe judicieux et variés.
  - Prenez garde à qui vous donnez vos mots de passe.
  - Faites des mises à jour régulières de votre système.

Utilisation d'un firewall

* Le firewall a pour but de réduire la quantité de connexions non désirées (ex : pirates)
* Il n'est pas un outil absolu !
* **Netfilter**, le firewall de Linux est directement intégré au noyau sous forme de modules.

Utilisation d'un firewall (2)

* **Netfilter** peut être configuré par le biais de la commande **iptables**.
* l'utilisation directe d'**iptables** est délicate et fait appel à des commandes complexes
* Afin de simplifier son utilisation, nous allons utiliser un 'frontend' pour iptables

Installer le frontend *firestarter*

* Site officiel :

  `http://www.fs-security.com/`

* Pour installer sous ubuntu (universal uniquement)

```
sudo aptitude install firestarter
```

* Par défaut, Firestarter laisse passer toutes les connexions sortantes.
* Par défaut, Firestarter bloque toutes les connexions entrantes.
* Les serveurs que vous avez installés doivent donc être autorisés explicitement.

Installer le frontend *firestarter* (2)

* Firestarter doit être lancé en mode super-utilisateur.
* Sous Gnome sous Ubuntu, firestarter ne possède aucune icône.
* Nous allons en créer un :
  - Allez dans la barre de lancement rapide (les petites icônes)
  - Cliquez sur **Ajouter au menu**
  - Cliquez sur le bouton **Lanceur d'applications personnalisé**

Installer le frontend *firestarter* (3)

* Création d'un racourci (suite) :
  - En nom et nom générique, mettez : Firestarter
  - En commande, mettez :

```
gksudo firestarter
```

  - Cliquez sur **Aucune icône** et notez dans la zone de texte en haut : */usr/share/pixmaps/firestarter.png*
  - Validez, et l'icône devrait apparaitre

Configurer *firestarter*

* Pour autoriser des connexions entrantes ou interdire des connexions sortantes, allez dans **Politique**
* Dans les connexions entrantes, cliquez droit sur la fenêtre dans **Autoriser le service** puis sur **Ajouter une règle**
* Le nom de service correspond au service que vous voulez autoriser. Cliquez sur **Ajouter** lorsque vous avez séléctionné le service dans la liste.
* Si le service désiré n'apparaît pas dans la liste, indiquez vous même le numéro de port utilisé dans **Port**.
* Cliquez sur le menu **Pare-feu** puis sur **Démarer le pare-feu** pour mettre à jour les modifications

Partager sa connexion

* En IPv4, les FAI distribuent aux particuliers des adresses IP uniques
* Un réseau domestique dispose donc d'une seule adresse IP pour plusieurs machines
* On fait alors un **réseau local**, réseau indépendant d'internet

Partager sa connexion (2)

* L'un des ordinateurs du réseau est donc connecté à deux réseaux : le réseau local et internet.
* Cette machine est la paserelle
* Pour que toutes les machines du réseau local aient accès à internet, il faut que la passerelle soit en mesure de transmettre les paquets d'un réseau à l'autre

Partager sa connexion (3)

* C'est également netfilter (et iptables) qui permettent de transférer des paquets
* *firestarter* permet, via iptables et netfilter, de faire un transfert de paquet
* Dans *firestarter*, allez sur **Edition**, **Préférences** puis *Configuration du réseau*
* Indiquez le périphérique relié à internet (ppp0 dans le cas d'un modem USB ou modem ethernet, eth0 dans le cas des freebox-like fournis par les FAI)
* Cochez la case *Autoriser le partage de la connexion internet* puis validez

Installation d'un serveur DHCP

* Le protocole DHCP délègue la tâche de la configuration du réseau à une unique machine
* Les ordinateurs du réseau n'auront plus besoin de configuration pour le réseau
* Chaque client, au démarage, va demander une addresse IP au serveur DHCP

Installation d'un serveur DHCP (2)

* Aucune configuration n'est à faire sur les ordinateurs clients (même sous Windows ! ! !)
* isc-DHCP (`http://www.isc.org/index.pl?/sw/dhcp/`) est un serveur dhcp
* Sous Ubuntu, faites :

```
sudo aptitude install dhcpd
```

Configuration du serveur dhcp

* Le serveur DHCP peut se configurer via *firestarter*
* Retournez dans la **Configuration du réseau**
* Cochez la case *Autoriser le DHCP pour le réseau local*
* Séléctionnez le bouton *Créer une nouvelle configuration DHCP*, laissez les paramètres par défaut puis confirmez.
* Le serveur est maintenant en écoute et pret à assigner les addresse IP.