

Commencer à créer son serveur debian pour un jour peut être s'auto-héberger

Ce document s'adresse à un public assez bien ciblé : les Linuxiens qui ont des activités sur le web mais qui utilisent ou subissent un prestataire pour leurs différents hébergements. L'idée de ce document c'est de commencer à réunir les outils nécessaires pour un jour envisager l'auto-hébergement.

Le titre très approximatif n'est pas un effet de style. L'auto-hébergement certains y arrivent et on peut les en féliciter, néanmoins c'est ajouter des difficultés supplémentaires dont on peut donner facilement une liste non exhaustive.

- il faut posséder une machine dédiée
- il faut entretenir la machine dédiée 365 jours de l'année pour offrir le service
- ce qui veut dire que même absent de chez vous en vacances, le serveur doit fonctionner, tout comme votre box
- selon le trafic que vous générez en mélangeant toutes les activités, c'est votre upload qui subit
- il faut avoir les connaissances pour mettre en place tous les logiciels
- les maintenir pour éviter les failles de sécurité
- analyser les logs au quotidien pour vérifier que tout va bien
- maîtriser parfaitement les notions de sécurité inhérentes à la mise en place d'un serveur.

Dès lors il est bien simple de comprendre que nous sommes nombreux à préférer payer pour éviter ce travail d'administration, il n'est pas évident dans des emplois du temps déjà très serrés de rajouter cette énorme surcharge de travail. Néanmoins, être tributaire d'un tiers n'est jamais chose aisée, et il n'est pas inintéressant à un moment donné de se lancer dans la compréhension de ces outils pour un jour imaginer pouvoir s'auto-héberger.

Quel que soit l'état d'avancement de cette documentation je ne me permettrai pas d'aller au delà du terme d'ébauche, il suffit de consulter les livres traitant du sujet pour se rendre compte de l'étendue et de la complexité. Je n'ai ici que la prétention de donner les éléments pour mettre en place un serveur qui répond à mes petits besoins et surtout à la progression de ma compréhension.

Cette documentation sera mise à jour de façon régulière.

I) Fixer l'adresse IP du serveur dans le réseau.

Votre serveur se trouve dans votre réseau domestique et par le fait, derrière votre box. Si l'on veut forwarder un port vers la machine comme le port 80 pour un site internet ou le port correspondant au ssh pour accéder à la machine de l'extérieur, il est nécessaire de fixer l'adresse IP.

Modifier le fichier `/etc/network/interfaces` et remplacer le contenu faisant allusion au DHCP par l'ensemble suivant. On peut supposer que la passerelle par défaut c'est 192.168.1.1

```
iface eth0 inet static
address 192.168.x.x
netmask 255.255.255.0
gateway 192.168.x.x
```

Une fois le changement réalisé, il suffit de faire `/etc/init.d/networking restart`, un ifconfig permet alors de vérifier l'adresse ip de la machine.

II) ssh pour communiquer avec le serveur.

1) Pour administrer sa machine

La première difficulté lorsque l'on veut monter son serveur c'est de faire abstraction de toute interface graphique. L'écran que vous allez aller monter sur le serveur pour faire l'installation de debian et bien après c'est terminé, on ne doit le mettre que pour vérifier une défaillance matérielle, tout comme le clavier. La communication avec le serveur se fait donc depuis un autre poste, le protocole qui sera utilisé est ssh, un protocole sécurisé.

Pour installer ssh : **apt-get install openssh-server**

Deux mesures de sécurité simple à mettre en oeuvre. Par défaut le port ssh est le port 22, il est donc judicieux de le changer en un autre port. Ensuite, il est indispensable d'interdire la connexion par le compte root, de cette façon il sera plus difficile pour un pirate d'accéder à votre machine, il faut d'abord trouver le mot de passe de votre compte, puis le mot de passe du root pour en prendre le contrôle.

La modification de ces paramètres se fait par le biais du fichier `/etc/ssh/sshd_config`. On modifie le port 22, et on bascule le `PermitRootLogin` de yes à no. On recharge ssh par **/etc/init.d/ssh reload**.

2) Pour échanger des fichiers

A partir du moment où l'on a un protocole sécurisé pour communiquer avec sa machine, il peut être intéressant de créer d'autres comptes pour que d'autres personnes puissent aussi communiquer avec votre machine pour par exemple échanger des fichiers. « Le soucis » du protocole ssh c'est qu'il permet à n'importe quel utilisateur de naviguer dans l'intégralité du serveur. Il existe une manière simple de restreindre entre autre un utilisateur dans son home, en utilisant le logiciel mysecureshell.

Pour installer mysecureshell dans debian, on édite le fichier `/etc/sources.list`, on ajoute à la fin du fichier les deux lignes suivantes :

```
deb http://mysecureshell.free.fr/repository/index.php/debian testing main
deb-src http://mysecureshell.free.fr/repository/index.php/debian testing main
```

puis **apt-get update && apt-get install mysecureshell**

Avant de créer des utilisateurs mysecureshell, il est nécessaire de créer des utilisateurs sur la machine : **adduser nom_du_user**

Pour créer des mots de passe complexes il est possible d'installer et d'utiliser le logiciel pwgen, **pwgen -sy** créant des mots de passe avec un mélange de caractères alphanumérique et spéciaux.

Il suffit ensuite de modifier le fichier `/etc/passwd` la ligne
`nom_du_user:x:1002:1002:,,,:/home/nom_du_user:/bin/bash`

en

```
nom_du_user:x:1002:1002:,,,:/home/nom_du_user:/bin/MySecureShell
```

Il est possible de synthétiser le adduser et la modification du fichier en une étape : **adduser -shell /bin/MySecureShell nom_du_user**

Puis le fichier `/etc/ssh/sftp_config` dans lequel on va rajouter à la fin du fichier :

```
<User nom_du_user>
Home /home/nom_du_user
</User>
```

On notera que c'est dans ce fichier dans lequel on peut modifier toutes les options de configuration du partage de fichier.

- *GlobalDownload* la valeur de la bande passante allouée à l'ensemble des utilisateurs
- *Download* la limite de vitesse pour chaque connexion
- *StayAtHome true* le fameux paramètre qui maintient l'utilisateur dans son home
- *LimitConnection* le nombre de connexions maximum simultanées pour le serveur
- *LimitConnectionByUser* le nombre de connexions maximum par utilisateur

Il est bien sûr possible en déplaçant ces paramètres dans la zone `<User nom_du_user>Home /home/nom_du_user</User>` de différencier les paramètres selon les utilisateurs.

Pour surveiller le trafic en temps réel sur la machine, on peut utiliser l'outil **iftop**.

III) Les outils pour gérer des sites internet.

1) apache et php

apt-get install apache2 php5

Si maintenant dans votre navigateur vous saisissez l'adresse de votre serveur 192.168.1.x vous obtenez un message « it works ! » issu d'un index.html placé dans le répertoire `/var/www/` c'est dans celui-ci que vous déposerez votre ou vos sites internet.

Pour que le serveur retourne le moins d'informations possibles comme l'adresse IP de la machine, la version d'apache, ou le système d'exploitation on modifiera le fichier `/etc/apache2/conf.d/security` on bascule la donnée *ServerSignature* de on à off, *ServerTokens* à Prod

Pour vérifier que les informations ont bien disparu `/etc/init.d/apache2 reload`

2) découpage en plusieurs sites

Imaginons que votre serveur personnel doive accueillir plusieurs sites, il est alors nécessaire de le « découper » de façon à ce que chaque nom de domaine différent corresponde à l'une des parties du site, c'est ce qu'on appelle un VirtualHost. Il est nécessaire de faire un fichier par nom de domaine.

Exemple de VirtualHost qui devra se trouver dans le répertoire `/etc/apache2/sites-enabled/`:

```
<VirtualHost *:80>
DocumentRoot "/var/www/nomdurepertoire"
ServerName nomdusite.com
ServerAlias www.nomdusite.com
<Directory "/var/www/nomdurepertoire">
allow from all
Options +Indexes
</Directory>
CustomLog /var/log/apache2/nomdurepertoire.log combined
</VirtualHost>
```

il est facile de vérifier si cela fonctionne. Si l'on saisit l'adresse du serveur on doit voir apparaître *nomdurepertoire* si l'on saisit dans le navigateur *nomdusite.com* on arrive dans un répertoire vierge.

3) installation de mysql la base de donnée

apt-get install mysql-server php5-mysql

Durant l'installation vous sera demandé le mot de passe root de mysql. Il peut être intéressant

d'ajouter le logiciel phpmyadmin qui est un outil graphique pour gérer de façon simple les bases de données mysql.

apt-get install phpmyadmin

Dans le cadre d'hébergement mutualisé, c'est le fournisseur d'accès qui à l'aide d'interfaces simplifiées permettent de réaliser des opérations simples comme la création d'un utilisateur, de la base de données et des privilèges. On réalise que ces opérations sont facilement réalisables depuis phpmyadmin.

Sources :

- <http://www.siteduzero.com/tutoriel-3-274694-apprenez-a-installer-un-serveur-web-sous-debian.html> qui contient des erreurs.
- <http://www.lafermeduweb.net/billet/tutorial-creeer-un-serveur-web-complet-sous-debian-1-apache-160.html#confDebian>

Remerciements :

- cep : <http://www.cepcasa.info/blog/>