

# Rapport Linux :

## CONFIGURATION D'UN SERVEUR APACHE

Ce rapport a été fait par :



Ashley GUNNESS

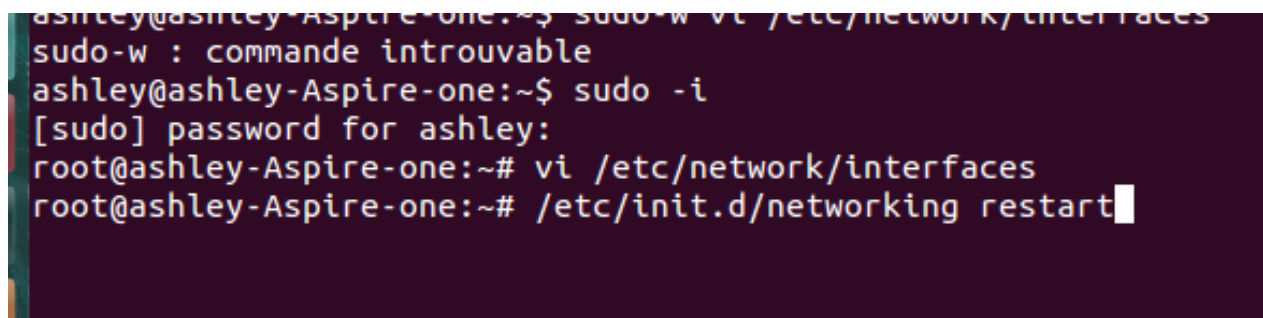
## Table des matières

1.	Configuration avant la mise en place du serveur Apache .....	2
	Avoir une adresse IP fixe.....	2
	Configurer un serveur DNS .....	3
2.	Le serveur Apache .....	8
	Configuration.....	8
	Sécurité .....	10
3.	Bilan .....	12
	Sources du projet.....	13

## 1. Configuration avant la mise en place du serveur Apache

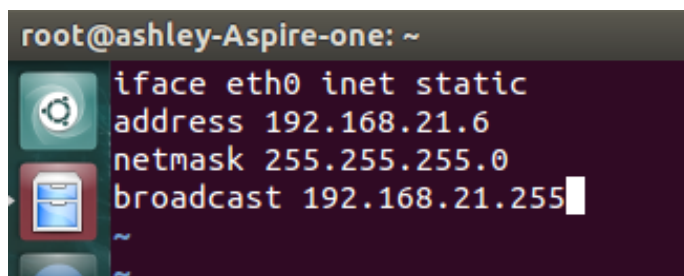
- AVOIR UNE ADRESSE IP FIXE

Avant de pouvoir configurer notre serveur Apache, il faut être sûr d'avoir déjà les configurations d'un serveur sur notre machine. Pour ce projet j'ai utilisé un notebook sous Ubuntu 14.02 LTR. Je vais d'abord attribuer une adresse IP fixe à mon notebook (c'est un numéro d'identification attribué à la machine sur le réseau), pour cela j'utilise le fichier de configuration « interfaces » :



```
ashley@ashley-Aspire-one:~$ sudo -w vi /etc/network/interfaces
sudo-w : commande introuvable
ashley@ashley-Aspire-one:~$ sudo -i
[sudo] password for ashley:
root@ashley-Aspire-one:~# vi /etc/network/interfaces
root@ashley-Aspire-one:~# /etc/init.d/networking restart
```

Sur ce screenshot on peut voir, que je suis d'abord passé en utilisateur root avec `sudo -i`, en effet on ne peut pas modifier les fichiers qui sont dans `/etc`, (à la 6) sans être root (avoir tous les droits, lecture, écriture, exécution), ensuite avec « vi » (un éditeur de texte) j'ai édité le fichier de configuration « interfaces » de cette façon :



```
root@ashley-Aspire-one: ~
iface eth0 inet static
address 192.168.21.6
netmask 255.255.255.0
broadcast 192.168.21.255
```

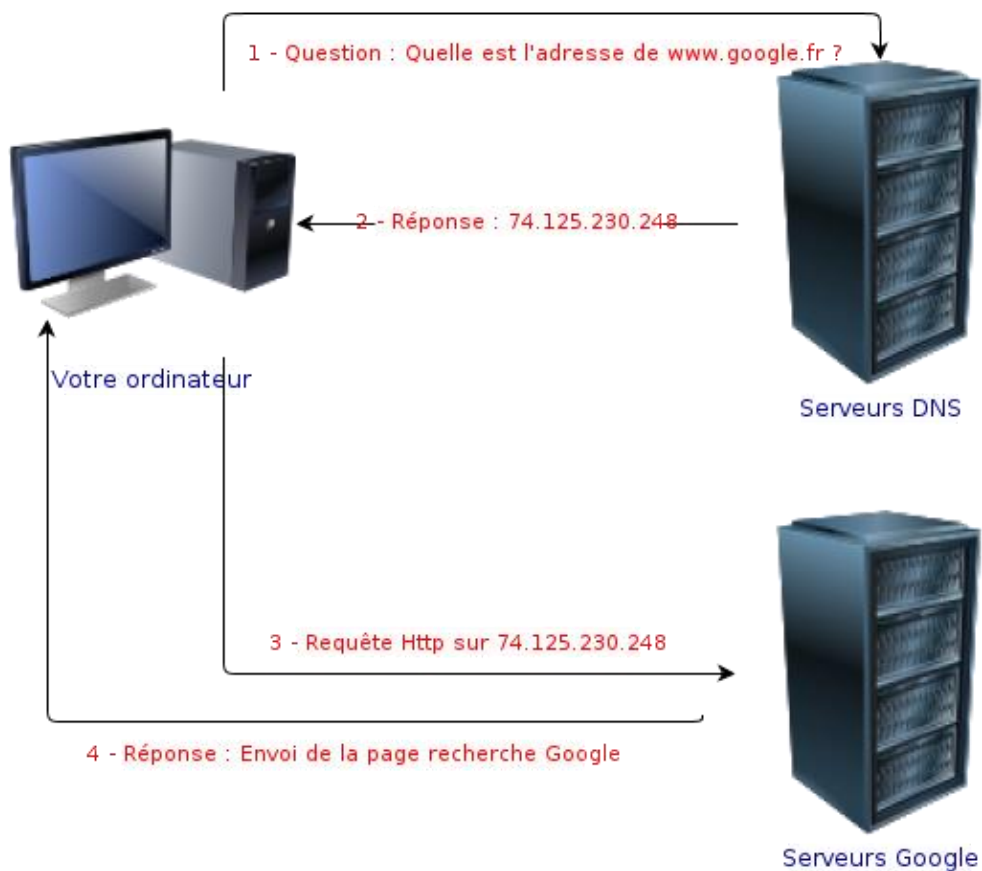
Je me suis ainsi doté d'une adresse IP fixe : 192.168.21.6 (par la suite, à cause de certains problème j'ai dû utiliser mon adresse local : 127.0.0.1).

- CONFIGURER UN SERVEUR DNS

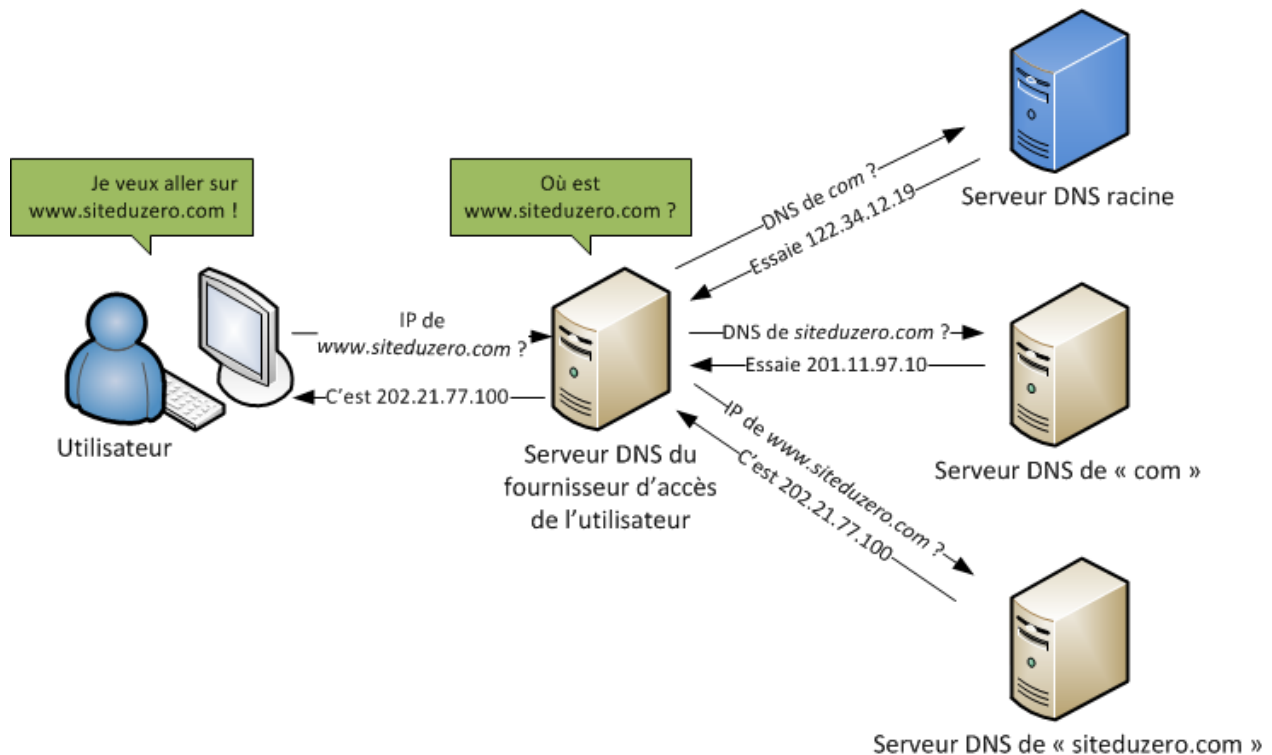
Maintenant que j'ai mon adresse IP fixe, je vais devoir configurer un serveur DNS. Mais d'abord il est important de savoir ce que c'est exactement : un serveur DNS permet de relier un nom de domaine (ex : google.fr) à une adresse IP (ex : 192.168.21.6)

Petite illustration :

### Principe d'une requête DNS



En détail :



Dans mon cas, mon notebook va jouer le rôle de serveur DNS. Je vais d'abord créer mon nom de domaine dans le fichier « `named.conf` » :

```
[sudo] password for ashley:
root@ashley-Aspire-one:~# vi /etc/network/interfaces
root@ashley-Aspire-one:~# /etc/init.d/networking restart
root@ashley-Aspire-one:~# vi /etc/named.conf
```

```
root@ashley-Aspire-one: ~
Zone « ash.go » { //c'est la zone directe,ici je configure mon nom de domaine, je l'ai appele'ash.go'
Type master;     //je donne le type "maitre" a mon nom de domaine pour indiquer
                  // que c'est le nom principal
File « /var/cache/bind/ash.go.hosts » // c'est le chemin qui permet d'accéder a la zone directe
};

Zone "21.168.192.in-addr.arpa"{ // je definis la zone inverse
Type master;     //je donne le type "maitre" comme avant
File " /var/cache/bind/ash.go.rev"; //chemin jusqu'au fichier zone inverse
};
```

Comme on peut le voir, j'ai édité le fichier avec « zone « ash.go » » en premier, c'est la zone de recherche directe, en bref le serveur DNS base sa recherche dans la base de données (fichier de zone qui contient les noms de domaines) sur la résolution nom vers adresse IP. A chaque nom de domaine il va associer une adresse IP.

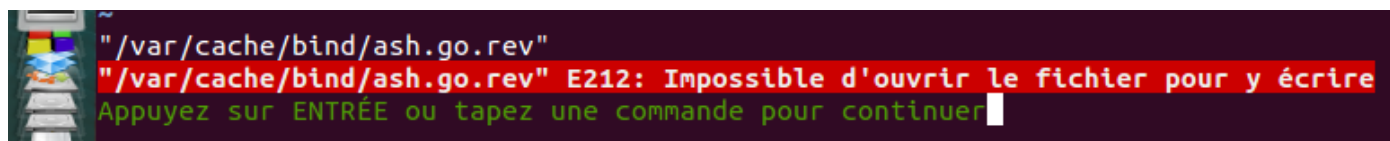
Ensuite j'ai mis « type master », qui est spécifique à BIND, un ensemble de paquets permettant la mise en place de DNS sous Debian, « type master » signifie simplement que la base des données utilisées provient d'une source locale, alors que pour un « slave » les données proviennent de l'extérieur.

« File « var/cache/bnd/ash.go.host » est le répertoire qui permet d'accéder au fichier de zone directe (la base de données).

Enfin j'ai définis la zone de recherche inverse, qui permet au serveur DNS, d'associer à l'adresse IP un nom de domaine.

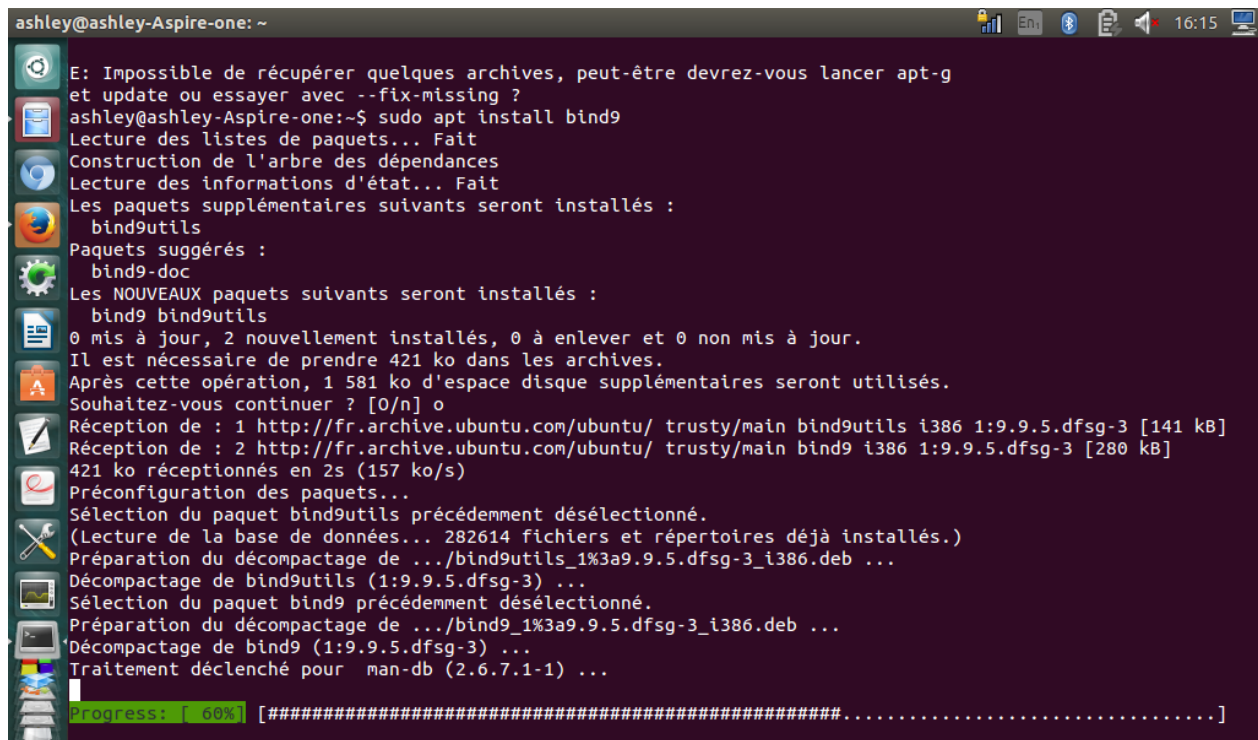
Mon serveur DNS a ainsi « autorisé » deux zones : ash.go et 21.168.192.in-addr.arpa.

On a ainsi un nom de domaine, mais il faut aussi configurer les fichiers ash.go.hosts et ash.go.rev, à ce moment j'ai rencontré une erreur :



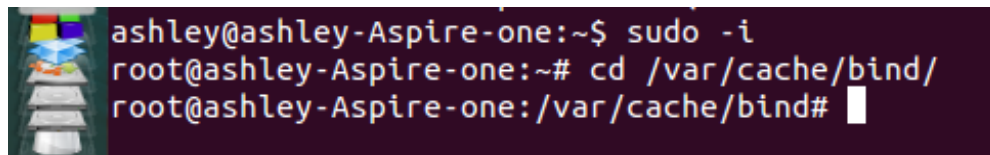
En fait le répertoire /bind n'existait pas car je n'avais pas installé bind sur mon notebook.

J'ai donc installé bind9 :



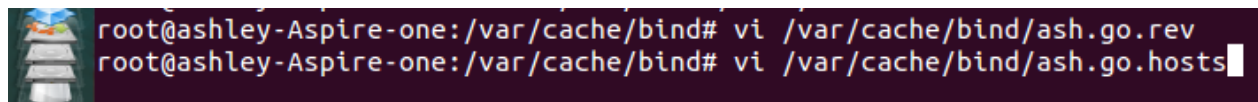
```
ashley@ashley-Aspire-one: ~  
E: Impossible de récupérer quelques archives, peut-être devriez-vous lancer apt-g  
et update ou essayer avec --fix-missing ?  
ashley@ashley-Aspire-one:~$ sudo apt install bind9  
Lecture des listes de paquets... Fait  
Construction de l'arbre des dépendances  
Lecture des informations d'état... Fait  
Les paquets supplémentaires suivants seront installés :  
  bind9utils  
Paquets suggérés :  
  bind9-doc  
Les NOUVEAUX paquets suivants seront installés :  
  bind9 bind9utils  
0 mis à jour, 2 nouvellement installés, 0 à enlever et 0 non mis à jour.  
Il est nécessaire de prendre 421 ko dans les archives.  
Après cette opération, 1 581 ko d'espace disque supplémentaires seront utilisés.  
Souhaitez-vous continuer ? [O/n] o  
Réception de : 1 http://fr.archive.ubuntu.com/ubuntu/ trusty/main bind9utils i386 1:9.9.5.dfsg-3 [141 kB]  
Réception de : 2 http://fr.archive.ubuntu.com/ubuntu/ trusty/main bind9 i386 1:9.9.5.dfsg-3 [280 kB]  
421 ko réceptionnés en 2s (157 ko/s)  
Préconfiguration des paquets...  
Sélection du paquet bind9utils précédemment désélectionné.  
(Lecture de la base de données... 282614 fichiers et répertoires déjà installés.)  
Préparation du décompactage de .../bind9utils_1%3a9.9.5.dfsg-3_i386.deb ...  
Décompactage de bind9utils (1:9.9.5.dfsg-3) ...  
Sélection du paquet bind9 précédemment désélectionné.  
Préparation du décompactage de .../bind9_1%3a9.9.5.dfsg-3_i386.deb ...  
Décompactage de bind9 (1:9.9.5.dfsg-3) ...  
Traitement déclenché pour man-db (2.6.7.1-1) ...  
Progress: [ 60%] [#####.....]
```

Et j'ai vérifié que le répertoire existait bien :



```
ashley@ashley-Aspire-one:~$ sudo -i  
root@ashley-Aspire-one:~# cd /var/cache/bind/  
root@ashley-Aspire-one:/var/cache/bind#
```

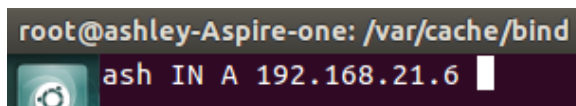
J'ai alors pu accéder aux fichiers ash.go.hosts et ash.go.rev :



```
root@ashley-Aspire-one:/var/cache/bind# vi /var/cache/bind/ash.go.rev  
root@ashley-Aspire-one:/var/cache/bind# vi /var/cache/bind/ash.go.hosts
```

Pour la configuration, j'ai mis dans le premier : 4 IN PTR ash.go

Cette commande (PTR) permet d'associer une adresse IP à un nom de domaine.

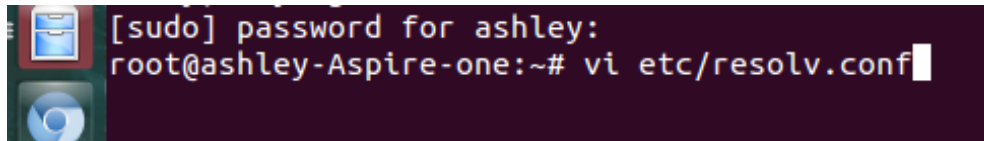


```
root@ashley-Aspire-one: /var/cache/bind  
ash IN A 192.168.21.6
```

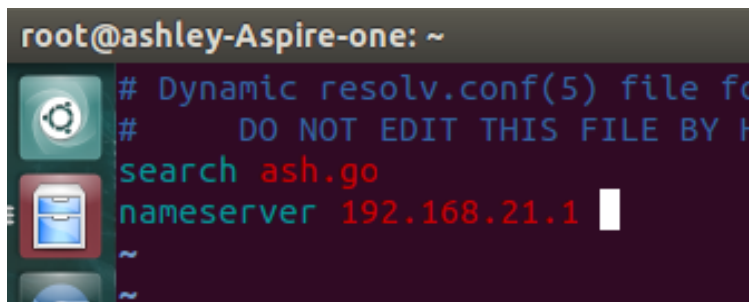
Et dans le deuxième fichier :

Ici, la commande « A » fait l'inverse de PTR et associe un nom de domaine à une adresse IP.

Maintenant, on doit aussi configurer les clients (ceux qui vont accéder au serveur), dans mon cas, j'accède au serveur à partir d'un navigateur du même ordinateur, donc je configure le fichier « resolv.conf » sur mon notebook :



```
[sudo] password for ashley:
root@ashley-Aspire-one:~# vi etc/resolv.conf
```



```
root@ashley-Aspire-one: ~
# Dynamic resolv.conf(5) file for glibc >= 2.4.
#     DO NOT EDIT THIS FILE BY HAND!!
search ash.go
nameserver 192.168.21.1
~
~
```

Ici on demande au client de chercher ash.go.

On a alors terminé de configurer le serveur DNS et on peut passer à la partie Apache.

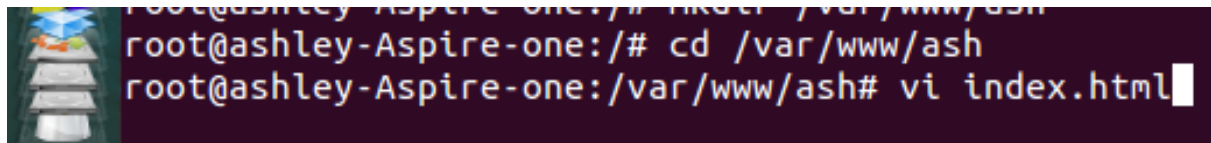


## 2. Le serveur Apache

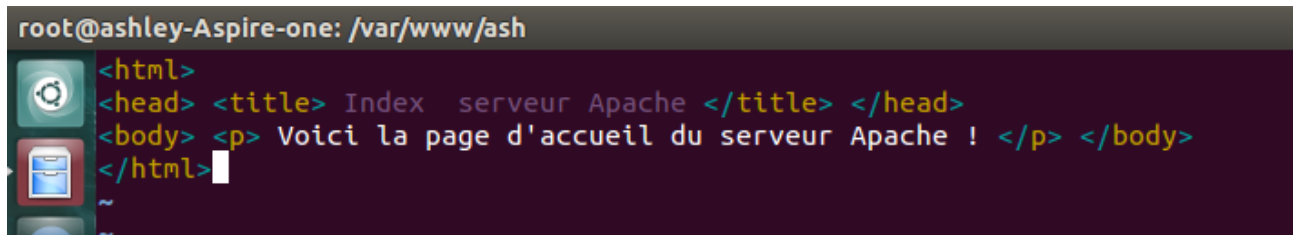
- CONFIGURATION

Avant de commencer, la configuration, il faut avoir les paquets apache, j'ai simplement fait un « `sudo apt-get install apache2` »

Ensuite j'ai créé une page html de test pour le serveur :



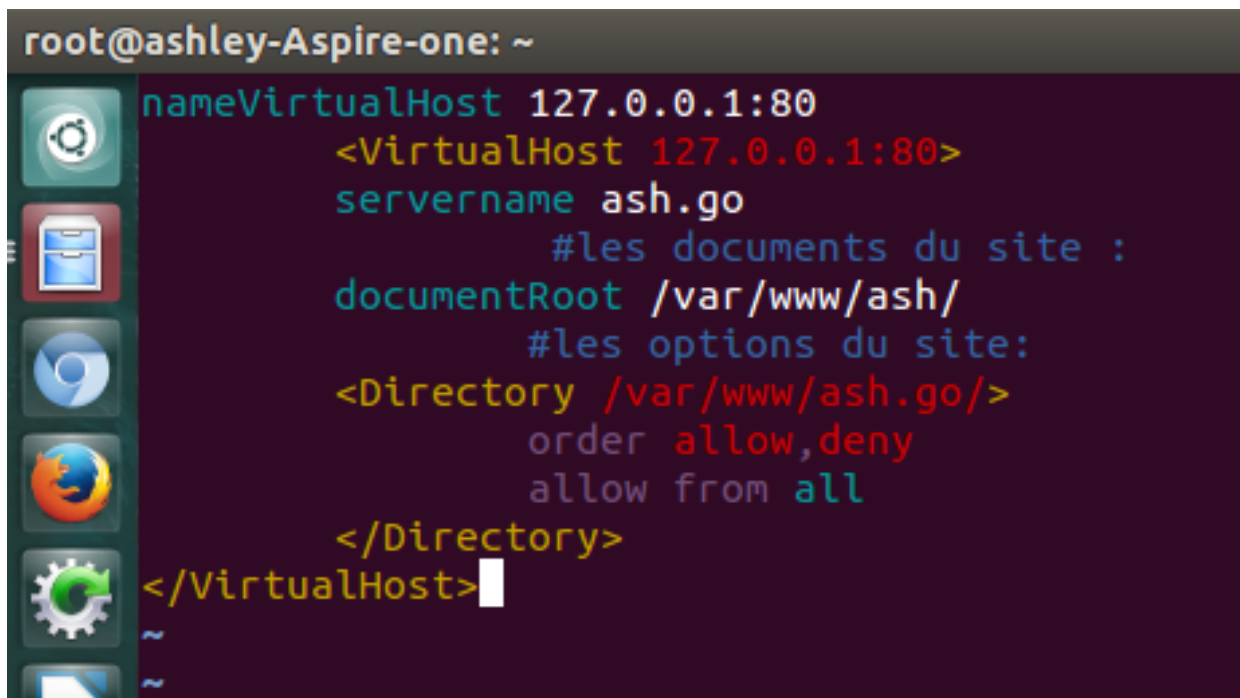
```
root@ashley-Aspire-one: /# cd /var/www/ash
root@ashley-Aspire-one: /var/www/ash# vi index.html
```



```
root@ashley-Aspire-one: /var/www/ash
<html>
<head> <title> Index  serveur Apache </title> </head>
<body> <p> Voici la page d'accueil du serveur Apache ! </p> </body>
</html>
```

Enfin j'ai pu commencer la configuration du serveur, je suis allé dans le répertoire `/etc/apache2/sites-available` et j'ai créé le fichier `ash`. Qui va contenir ma configuration :

(J'ai dû changer l'adresse IP en `127.0.0.1` à cause d'un problème de conflit)



```
root@ashley-Aspire-one: ~
nameVirtualHost 127.0.0.1:80
    <VirtualHost 127.0.0.1:80>
        servername ash.go
            #les documents du site :
        documentRoot /var/www/ash/
            #les options du site:
        <Directory /var/www/ash.go/>
            order allow,deny
            allow from all
        </Directory>
    </VirtualHost>
```

Ici, « nameVirtualHost 127.0.0.1:80 » permet d'indiquer l'adresse IP du serveur et le port qui lui est associé (80, le port réservé au Http).

« serverName ash.serv.go » indique le nom de domaine du serveur.

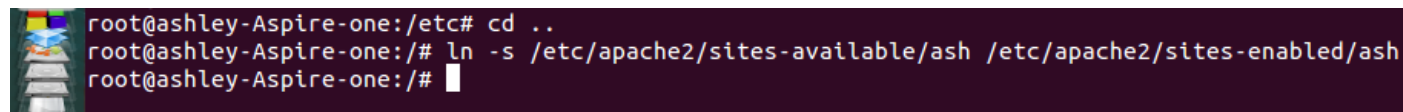
« documentRoot /var/www/ash/ » donne le chemin d'accès aux contenus des pages web.

« <Directory /var/www/sdz> » donne le chemin du répertoire apache par défaut.

« order allow,deny » indique les options pour un utilisateur, on lui donne l'accès ou on lui refuse dépendant de la configuration choisie.

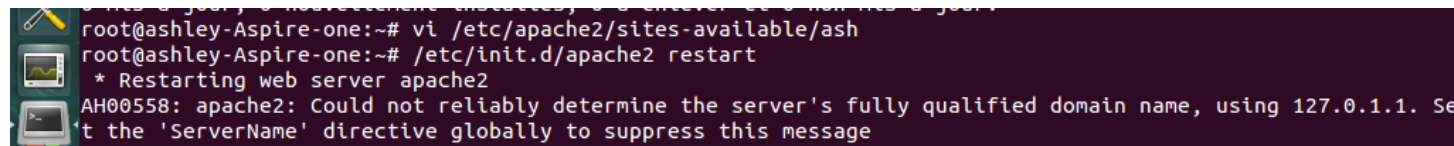
« allow from all » ici on donne l'accès au serveur tout le monde.

Après la configuration du serveur, il faut activer le domaine, pour cela j'ai créé un lien entre « /etc/apache2/sites-available/ash / » et « /etc/apache2/sites-enabled/ash », le site est alors actif pour Apache :



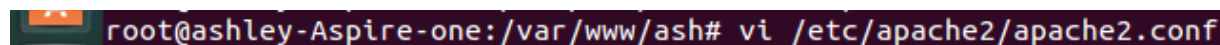
```
root@ashley-Aspire-one:/etc# cd ..
root@ashley-Aspire-one:/# ln -s /etc/apache2/sites-available/ash /etc/apache2/sites-enabled/ash
root@ashley-Aspire-one:/#
```

Pour que toute la configuration soit prise en compte, j'ai redémarré le serveur :



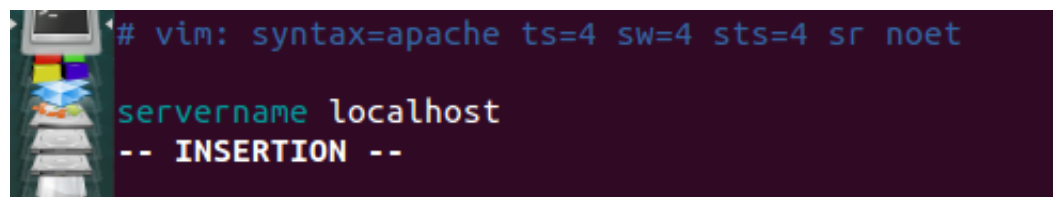
```
root@ashley-Aspire-one:~# vi /etc/apache2/sites-available/ash
root@ashley-Aspire-one:~# /etc/init.d/apache2 restart
* Restarting web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. See
'the 'ServerName' directive globally to suppress this message
```

J'ai alors rencontré l'erreur AH00558, que j'ai réglé en allant dans :



```
root@ashley-Aspire-one:/var/www/ash# vi /etc/apache2/apache2.conf
```

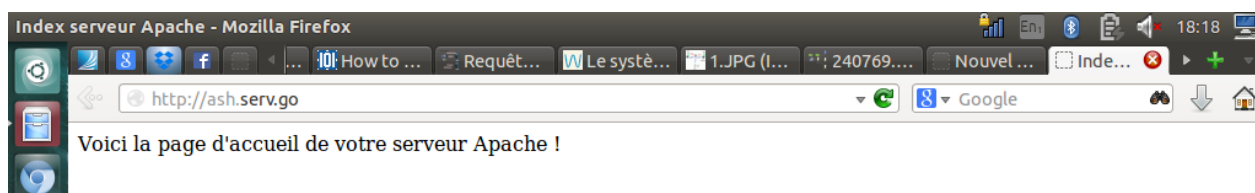
Et en ajoutant à la fin du fichier:



```
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
servername localhost
-- INSERTION --
```

Après j'ai dû faire d'autres modifications dans les fichiers de configuration précédents.

Enfin, après avoir corrigé cette erreur j'ai pu tester le serveur :



- SECURITE

Néanmoins, maintenant que mon serveur est fonctionnel, il ne faut pas oublier que j'ai mis « allow from all » dans mon fichier de configuration Apache. Tout le monde peut y accéder, il faut alors sécuriser un peu plus le serveur. Pour ça on va remodifier le fichier de configuration en y ajoutant quelques lignes :

```
root@ashley-Aspire-one: /etc/apache2
nameVirtualHost 127.0.0.1:80
<VirtualHost 127.0.0.1:80>
servername ash.go
    #les documents du site :
documentRoot /var/www/ash/
    #les options du site:
<Directory /var/www/ash.go/>
    order allow,deny
    allow from 127.0.0.2
    authname "moi"
    authtype basic
    authuserfile /var/www/ash/.htpasswd
    require valid-user
    satisfy all
    # allow from all
</Directory>
    #les logs(historiques de ips et des fichiers envoyes:
errorlog /var/log/apache2/ash.go-error_log
transferlog /var/log/apache2/ash.go-access_log
</VirtualHost>
~
~
```

Ici on peut voir que j'ai rajouté :

« allow from 127.0.0.2 » qui autorise l'accès qu'à l'ordinateur possédant cette adresse IP.

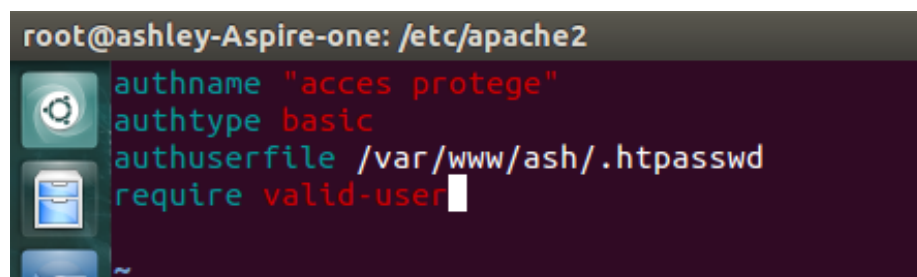
« authname « moi » » le nom de la machine qui pourra se connecter au serveur (ici « moi » est un exemple pas le vrai nom)

« authtype basic » indique que le serveur va utiliser des mot de passe dans « .htpasswd » pour identifier les utilisateurs

« authuserfile » c'est le répertoire de .htpasswd

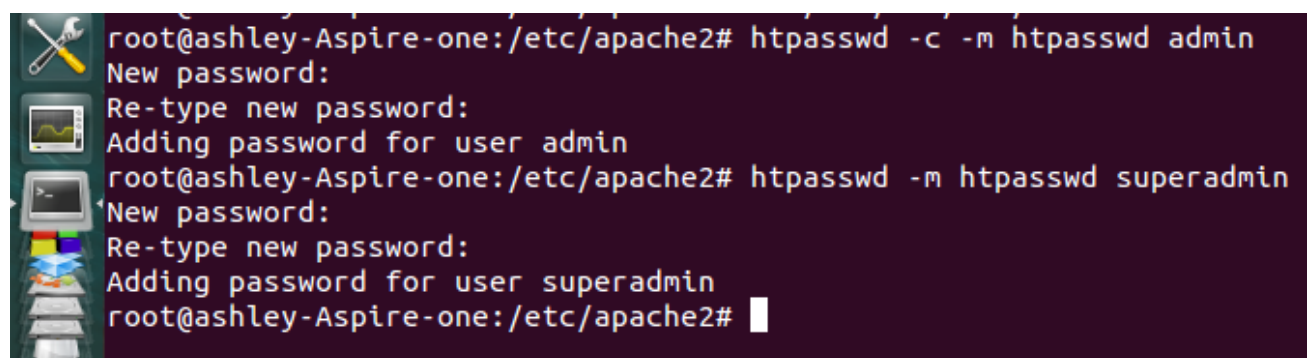
« require valid-user » l'utilisateur doit vérifier les conditions précédentes pour que la connexion s'effectue.

Je dois aussi configurer les fichiers .htaccess (dans le même répertoire que .htpasswd) et .htpasswd puisqu'ils sont utilisés dans la configuration :

A terminal window with a dark background and light-colored text. The prompt is 'root@ashley-Aspire-one: /etc/apache2'. The user has entered four lines of configuration: 'authname "acces protege"', 'authtype basic', 'authuserfile /var/www/ash/.htpasswd', and 'require valid-user'. The cursor is at the end of the last line.

```
root@ashley-Aspire-one: /etc/apache2
authname "acces protege"
authtype basic
authuserfile /var/www/ash/.htpasswd
require valid-user
```

Enfin on crée les mots de passe dans .htpasswd :

A terminal window with a dark background and light-colored text. The prompt is 'root@ashley-Aspire-one: /etc/apache2#'. The user enters 'htpasswd -c -m htpasswd admin'. The terminal prompts for a new password and then 'Re-type new password:'. The user enters a password, and the terminal says 'Adding password for user admin'. The user then enters 'htpasswd -m htpasswd superadmin'. The terminal prompts for a new password and then 'Re-type new password:'. The user enters a password, and the terminal says 'Adding password for user superadmin'. The prompt returns to 'root@ashley-Aspire-one: /etc/apache2#'.

```
root@ashley-Aspire-one: /etc/apache2# htpasswd -c -m htpasswd admin
New password:
Re-type new password:
Adding password for user admin
root@ashley-Aspire-one: /etc/apache2# htpasswd -m htpasswd superadmin
New password:
Re-type new password:
Adding password for user superadmin
root@ashley-Aspire-one: /etc/apache2#
```

Finalement on redémarre le serveur Apache, et on a alors un serveur protégé par mot de passe.

### 3. Bilan

Ce projet m'a permis de découvrir ce qu'était vraiment un serveur et son fonctionnement, en plus de me permettre d'apprendre de nouvelles fonctionnalités sous linux.

J'ai été très satisfait de réussir à mettre en place un serveur par mes propres moyens, je pensais que c'était plus difficile à faire. Finalement malgré les quelques difficultés rencontrées, j'ai pu les surmonter et réussir ce projet, néanmoins, je pense que j'aurais pu aller plus loin au niveau de la personnalisation du serveur, et de son type d'hébergement mais le temps me manquait.

- SOURCES DU PROJET

Pour ce projet je me suis inspiré de plusieurs tutoriels et des ressources d'autres sites que voici :

<http://tutorialforlinux.blogspot.fr/2013/10/solve-problem-ah00558-when-restarting.html>

<http://www.wikihow.com/Configure-Your-Website-Using-Htaccess-in-Linux-with-Apache>

<http://fr.openclassrooms.com/informatique/cours/mise-en-place-des-serveurs-apache-et-dns/configuration-ip-du-serveur>

<http://www.domaine-achat.fr/faq/content/3/48/fr/serveur-dns-definition-.html>

[http://fr.wikipedia.org/wiki/Domain\\_Name\\_System](http://fr.wikipedia.org/wiki/Domain_Name_System)

<http://fr.openclassrooms.com/informatique/exportPdf/mise-en-place-des-serveurs-apache-et-dns>

<http://www.cgsecurity.org/Articles/dns.html>

<http://fr.openclassrooms.com/informatique/cours/mise-en-place-des-serveurs-apache-et-dns/configuration-du-serveur-dns>

<http://www.developpez.net/forums/d1414132/systemes/linux/administration-systeme/serveur-maitre-serveur-slave-dns/>

<http://www.zytrax.com/books/dns/ch4/-master>

<http://fr.openclassrooms.com/informatique/cours/mise-en-place-des-serveurs-apache-et-dns/mise-en-place-du-serveur-apache>

<http://www.commentcamarche.net/contents/518-dns-systeme-de-noms-de-domaine>

[http://wiki.linux-france.org/wiki/Les\\_commandes\\_fondamentales\\_de\\_Linux](http://wiki.linux-france.org/wiki/Les_commandes_fondamentales_de_Linux)  
[Cr.C3.Ager un r.C3.Agpertoire .28mkdir.29](#)

<http://ipfixe.emule-french.org/linux/linux.htm>

[http://www.commentcamarche.net/faq/22-comment-connaître-son-adresse-ip - sous-linux](http://www.commentcamarche.net/faq/22-comment-connaître-son-adresse-ip-sous-linux)

<http://www.commentcamarche.net/forum/affich-10992159-ubuntu-probleme-de-permission>

[http://fr.wikipedia.org/wiki/Adresse\\_IP](http://fr.wikipedia.org/wiki/Adresse_IP)

<http://fr.openclassrooms.com/informatique/cours/installer-un-serveur-dns-avec-djbdns-sous-debian>