

Job 2:

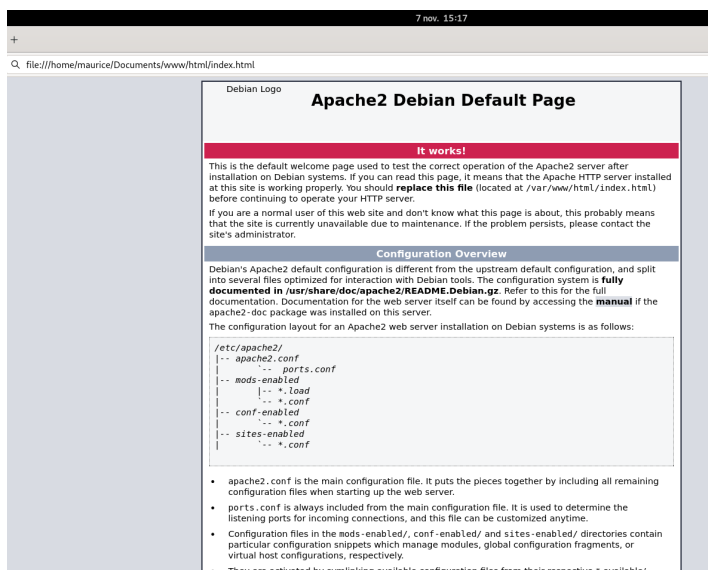
```
maurice@debian:~# sudo -s
root@debian:~# apt-get install apache2
root@debian:~# cd /var/www
root@debian:/var/www # cp /var/www home/document
```

Dans un premier temps nous devons nous mettre en root afin d'avoir l'accès à l'intégralité des droits dans le terminal de commande .

Deuxièmement il nous faut installer apache2 a l'aide de la commande *apt-get install* .

Par la suite il va falloir se placer dans le fichier www a l'aide de la commande *cd* le chemin de fichier étant */var/www* .

Et pour finir nous allons copier coller le contenu du fichier www dans le répertoire document



The screenshot shows a web browser window displaying the "Apache2 Debian Default Page". The page includes a Debian logo, a "It works!" message, and a "Configuration Overview" section. The overview explains that the default configuration is different from the upstream default and is split into several files. It lists the configuration files and their locations, and provides a list of files to be activated by symlinking.

Debian Logo

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/`

Ensuite nous allons aller dans le répertoire `document` puis dans `www` , `html` et ouvrir le fichier `index.html` afin d'obtenir la page suivante .

Job 3 :

Il existe plusieurs type de serveur web :

Serveurs DNS :

Les serveurs DNS (pour Domain Name System) sont des serveurs d'applications utilisés pour résoudre les noms de domaines des ordinateurs clients, c'est-à-dire traduire des noms conçus pour être compris de l'homme en adresses IP exploitables par une machine. Le système DNS est une base de données largement répandue qui contient des noms et d'autres serveurs DNS dont chacun peut servir à demander le nom d'un ordinateur qui, autrement, resterait inconnu. Quand un client a besoin de l'adresse d'un système, il envoie à un serveur DNS une requête DNS portant le nom de la ressource visée. Le serveur DNS répond en lui fournissant l'adresse IP nécessaire, qu'il trouvera au sein de sa table de noms.

Serveurs web :

Les serveurs web comptent parmi les catégories de serveurs les plus répandues sur le marché à l'heure actuelle. Un serveur web est un genre de serveur d'applications qui héberge des logiciels et des données que les utilisateurs vont solliciter sur internet ou sur un intranet. Ces serveurs répondent aux demandes de pages web ou d'autres services web qui proviennent de navigateurs tournant sur des ordinateurs clients. Parmi les serveurs web les plus populaires, on peut citer ceux d'Apache, de Microsoft Internet Information Services (IIS) et les serveurs Nginx.

Serveurs virtuels :

Les serveurs virtuels sont en train de conquérir le monde de l'informatique. Contrairement aux serveurs classiques, qu'on installe comme un système d'exploitation sur une machine physique, les serveurs virtuels ne peuvent tourner que sur un logiciel spécialisé appelé hyperviseur. Chaque hyperviseur peut exécuter simultanément des centaines, voire des milliers de serveurs virtuels. L'hyperviseur présente le matériel virtuel au serveur comme s'il s'agissait de matériel physique. Le serveur virtuel utilise le matériel virtuel comme s'il s'agissait de hardware normal, et l'hyperviseur confie les tâches de calcul et de stockage à la machine sur laquelle il est installé, qui est commune à tous les autres serveurs virtuels.

Serveurs de bases de données :

Le volume de données utilisées par les entreprises, les utilisateurs, et les autres services est tout bonnement colossal. Beaucoup de ces données sont stockées dans des bases de données. Celles-ci doivent être accessibles à de nombreux clients à n'importe quel moment et peuvent mobiliser une gigantesque quantité d'espace disque. À ce titre, les serveurs sont une solution idéale pour héberger des bases de données. Les serveurs de bases de données exécutent des applications de bases de données et répondent aux nombreuses requêtes des clients. Les applications les plus répandues sur ce segment sont Oracle, Microsoft SQL Server, DB2, ou encore Informix.

Job 4 :

```
maurice@debian:~# sudo -s
root@debian:~# cd /
root@debian:~# cd etc
root@debian:/etc# nano hostname
```

```
192.168.106.132 dnsproject.prepa.com
10.10.28.148 maurice
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Au cours du job 4 nous avons dû associer l'adresse IP à notre nom de domaine et donc pouvoir ping via le nom de domaine

Job 5 :

Pour obtenir un nom de domaine il faut envoyer une demande d'obtention à un organisme spécialisé dans l'hébergement de serveur , de service de messagerie comme : Amen, Gandi, Mail Club, Ovh, Ikoula, Ionos, etc. Ou sur l'Afnic (organisme officiel français servant s'occupant de la distribution du domaine .fr) de plus le nom de domaine est payant au prix d' environ 10 euros par an.

Le .com sert pour les site d'entreprise le .com étant une contraction du mot company
le .fr est le domaine de premier niveau en France
le .gouv est quant à lui réservé à l'administration française.

Job 6 :

Tout d'abord modifier hostname :

```
maurice@debian:~# sudo -s
root@debian:~# cd /
root@debian:~# cd etc
root@debian:/etc# nano hostname
```

```
10.10.28.148 maurice
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Puis il faut copier le fichier db.local pour la configuration du fichier direct :

```
GNU nano 4.8 db.dnsproject.prepa.com
$TTL 604800
@      IN      SOA     dnsproject.prepa.com. maurice.dnsproject.prepa.com. (
; Serial
604800 ; Refresh
86400  ; Retry
2419200 ; Expire
604800 ) ; Negative Cache TTL
; Serveur DNS

@      IN      NS      maurice.dnsproject.prepa.com.
maurice IN      IN      A      10.10.28.202
www     IN      CNAME   maurice.dnsproject.prepa.com.
```

Ensuite on copie ce fichier **direct** pour créer le fichier **reverse** :

```
; BIND reverse data file for broadcast zone
;
$TTL      604800
@         IN      SOA      dnsproject.prepa.com. maurice.dnsproject.prepa.com. (
; Serial
; Refresh
; Retry
; Expire
; Negative Cache TTL
);
@         IN      NS       maurice.dnsproject.prepa.com.
maurice   IN      A        10.10.28.202
28.202    IN      PTR      maurice.dnsproject.prepa.com.
```

Puis on établit les zones dans le fichier **db.conf.local** :

```
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "dnsproject.prepa.com" IN {
    type master;
    file "/etc/bind/db.dnsproject.prepa.com";
};
zone "10.10.in-addr.arpa" IN {
    type master;
    file "/etc/bind/db.192";
};
```

Ensuite on modifie le fichier **resolv.conf** :

```
This file is managed by man:systemd-resolved(8). Do not edit.
#
# This is a dynamic resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists all
# configured search domains.
#
# Run "resolvectl status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs must not access this file directly, but only through the
# symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a different way,
# replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.

nameserver 10.10.28.202
options edns0 trust-ad
search dnsproject.prepa.com
```

Enfin on configure le fichier **hosts** :

```
10.10.28.148 dnsproject.prepa.com
# The following lines are desirable for IPv6 capable hosts
::1      ip6-localhost ip6-loopback
fe00::0  ip6-localnet
ff00::0  ip6-mcastprefix
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters
```

Enfin on redémarre **bind9** et on vérifie si le serveur est opérationnel grâce à la commande **nslookup** : *remplacer raph par Maurice

```
UwU - VMware Workstation 16 Player (Non-commercial use only)
Player
root@raph: /home/rossignol

rossignol@raph:~$ sudo -s
[sudo] Mot de passe de rossignol :
root@raph:/home/rossignol# service bind9 start
root@raph:/home/rossignol# nslookup www
Server:      10.10.28.202
Address:     10.10.28.202#53

www.dnsproject.prepa.com      canonical name = raph.dnsproject.prepa.com.
Name:   raph.dnsproject.prepa.com
Address: 10.10.28.202

root@raph:/home/rossignol#
```

Tout est opérationnel donc on peut entrer le nom de domaine dans le navigateur de l'hôte :

