

# Installation et configuration d'un serveur http

## Objectifs

- L'objectif de cette activité est de créer un serveur WEB (protocole http) en installant le logiciel Apache.

## Moyens

Une machine de référence Debian et deux machines clientes Debian et Windows issues des TP précédents.



## Compte Rendu

Votre CR contiendra les réponses aux questions posées à déposer dans profsetudiants\Espace\_B1SISR\Depots\_B1\_Support\_PC à la fin de la séance.

Nom du fichier : **TP04-ServeurHttp-Partie1**.

## 1. Pré-requis

Vous devez disposer :

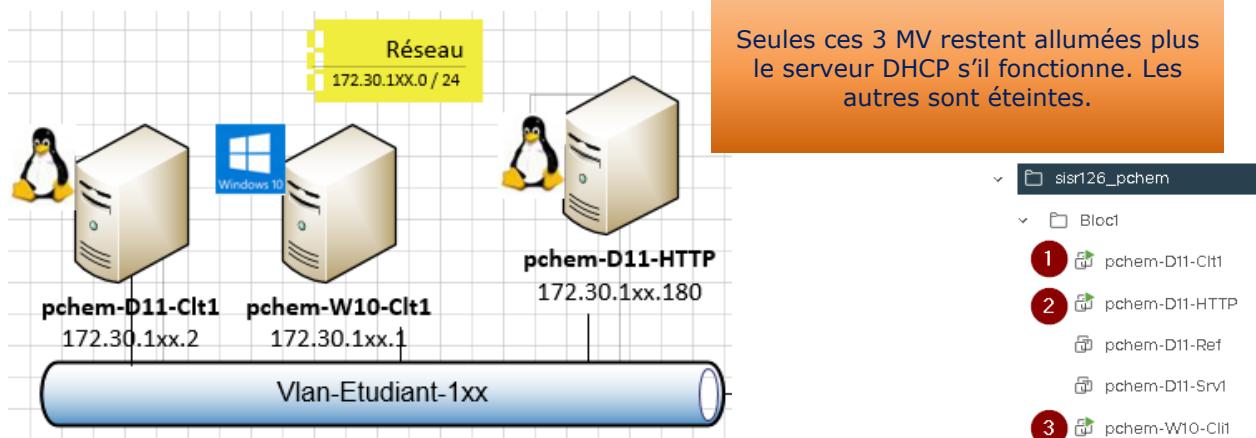
De deux **machines clientes Debian et Windows** que vous avez déjà sous le nom **xxxx-D11-Clt1 et xxxx-W10-Clt1**.

Et d'une machine **Debian de référence** déjà paramétrée. En particulier :

- VMware Tools installés ;
- Alias paramétrées (ls, ll, l, ...);
- Administration à distance avec le protocole SSH installé et paramétré ;
- Adresse IP : 172.30.1xx.222/24, passerelle : 172.30.1xx.254

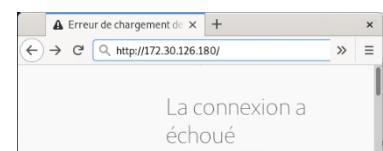
Une fois assuré(e) que tout est bien paramétré dans la machine de référence, clonez là vers la nouvelle machine **xxxx-D11-http** qui sera un serveur WEB (protocole http).

Mettez en place ce réseau de machines virtuelles en paramétrant les noms et configurations IP des systèmes d'exploitation.



1. Montrer que les configurations suivantes sont effectuées sur le serveur WEB : configurations héritées de la machine de référence (alias et accès ssh) ainsi que les nouveaux nom et configuration IPv4 ;
2. Montrer que les deux postes client et le serveur WEB communiquent (Vérifier que les adresses IP soient conformes au schéma).

Testez une connexion d'un navigateur web à partir de chaque poste client vers le serveur web. Vous devez obtenir ce résultat :



## 2. Installation du service Apache sous Linux

Le logiciel libre Apache HTTP Server (Apache) est un serveur HTTP créé et maintenu au sein de la fondation Apache. C'est le serveur HTTP le plus populaire du World Wide Web.

La version 2 d'Apache propose entre autres le support de plusieurs plates-formes (Windows, Linux et UNIX, Solaris, BSD, MAC OS X).

Nous allons utiliser l'outils APT qui permet d'installer les paquets logiciels à partir d'un site miroir de Debian. Les sites miroirs sont indiqués dans le fichier sources.list que nous étudierons ultérieurement.

Il faut installer le paquet apache2 :

```
# apt update                                → cette commande met des erreurs du type « pas de fichier Release »
# apt-get install apache2
```

```
root@pchem-D10-HTTP:~# apt-get install apache2
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libbrotli1 libcurl4 libjansson4 liblua5.2-0 ssl-cert
Paquets suggérés :
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser openssl-blacklist
Les NOUVEAUX paquets suivants seront installés :
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libbrotli1 libcurl4 libjansson4 liblua5.2-0 ssl-cert
0 mis à jour, 13 nouvellement installés, 0 à enlever et 1 non mis à jour.
Il est nécessaire de prendre 2 959 ko dans les archives.
Après cette opération, 9 659 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]
```

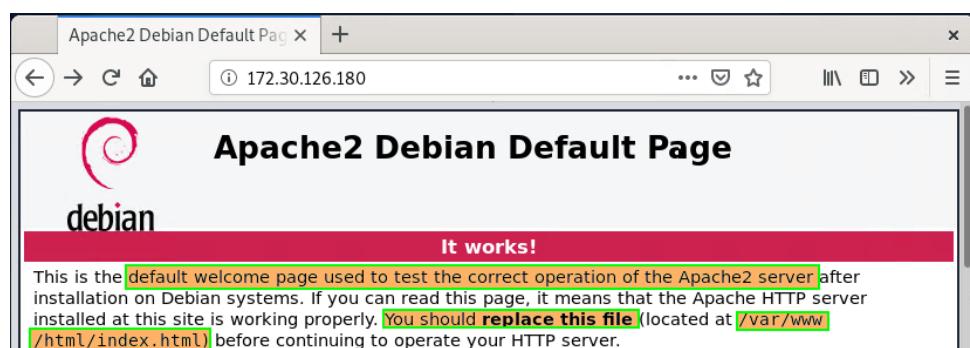
Une fois l'installation terminée, on peut vérifier l'état du serveur HTTP Apache :

```
# /etc/init.d/apache2 status
# apachectl status
# service apache2 status
```

```
root@pchem-D10-HTTP:~# systemctl status apache2
● apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Active: active (running) since Sat 2021-02-20 12:40:21 CET; 41s ago
    Docs: https://httpd.apache.org/docs/2.4/
 Main PID: 1122 (apache2)
   Tasks: 55 (limit: 2359)
  Memory: 9.0M
    CGroup: /system.slice/apache2.service
            └─1122 /usr/sbin/apache2 -k start
                  ├─1124 /usr/sbin/apache2 -k start
                  ├─1125 /usr/sbin/apache2 -k start
                  └─1126 /usr/sbin/apache2 -k start

févr. 20 12:40:21 pchem-D10-HTTP systemd[1]: Starting The Apache HTTP Server...
févr. 20 12:40:21 pchem-D10-HTTP apachectl[1111]: AH00558: apache2: Could not reliably determine t
févr. 20 12:40:21 pchem-D10-HTTP systemd[1]: Started The Apache HTTP Server.
lines 1-15/15 (END)
```

3. Montrer que votre service WEB fonctionne correctement.
4. Tester l'accès à votre service web, configuré par défaut, à partir du navigateur web des postes client. Vous devez maintenant obtenir ce résultat (attention, pas en sécurisé ou https) :



### 3. Arrêt et redémarrage du service apache

Tout au long d'un TP Apache, nous avons besoin d'arrêter et de redémarrer notre serveur http apache pour tester ses différentes fonctionnalités.

```
root@pchem-D10-HTTP:~# service apache2 status
● apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Active: active (running) since Sat 2021-02-20 15:53:00 CET; 1h 21min ago
    Docs: https://httpd.apache.org/docs/2.4/
  Process: 1859 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 1863 (apache2)
   Tasks: 55 (limit: 2359)
  Memory: 9.5M
    CGroup: /system.slice/apache2.service
            └─1863 /usr/sbin/apache2 -k start
              ├─1864 /usr/sbin/apache2 -k start
              ├─1865 /usr/sbin/apache2 -k start
```

#### Démarrage normal

Démarre normalement le processus daemon httpd (apache2 dans notre cas).

```
# Service apache2 start
```

#### Arrêter immédiatement

L'envoi du signal stop au processus parent induit chez celui-ci une tentative immédiate de tuer tous ses processus enfants. Cela peut durer plusieurs secondes. Après cela, le processus parent lui-même se termine. Toutes les requêtes en cours sont terminées, et plus aucune autre n'est traitée.

```
# Service apache2 stop
```

#### Arrêt en douceur

L'envoi du signal graceful-stop au processus parent lui fait aviser les processus enfants de s'arrêter après le traitement de leur requête en cours (ou de s'arrêter immédiatement s'ils n'ont plus de requête à traiter).

Lorsque tous les processus enfants ont terminé leurs traitements et se sont arrêtés ou lorsque le délai spécifié par la directive GracefulShutdownTimeout a été atteint, le processus parent s'arrêtera à son tour.

```
# Service apache2 graceful-stop
```

#### Redémarrer immédiatement

L'envoi du signal **restart** au processus parent lui fait tuer ses processus enfants comme pour le signal stop, mais le processus parent ne se termine pas.

Il relit ses fichiers de configuration, et réouvre ses fichiers de log. Puis il donne naissance à un nouveau jeu de processus enfants et continue de traiter les requêtes.

```
# Service apache2 restart
```

#### Quelle est la version installée ?

```
# apache2 -v
```

5. Tester ces 5 commandes et montrez leur effet à partir d'un poste client (sauf la dernière) en testant la connexion du navigateur web connectée sur le service Apache.

## 4. Fichiers de publications

Où est stocké le fichier qui contient la page d'accueil par défaut du service Apache ?

```
# /var/www/html      # Ici, j'utilise l'alias « I » qui est défini sur « ls -al »
```

Au démarrage, le service apache recherche une page par défaut. La page « **index.html** », lorsqu'elle est trouvée, est lancée automatiquement.

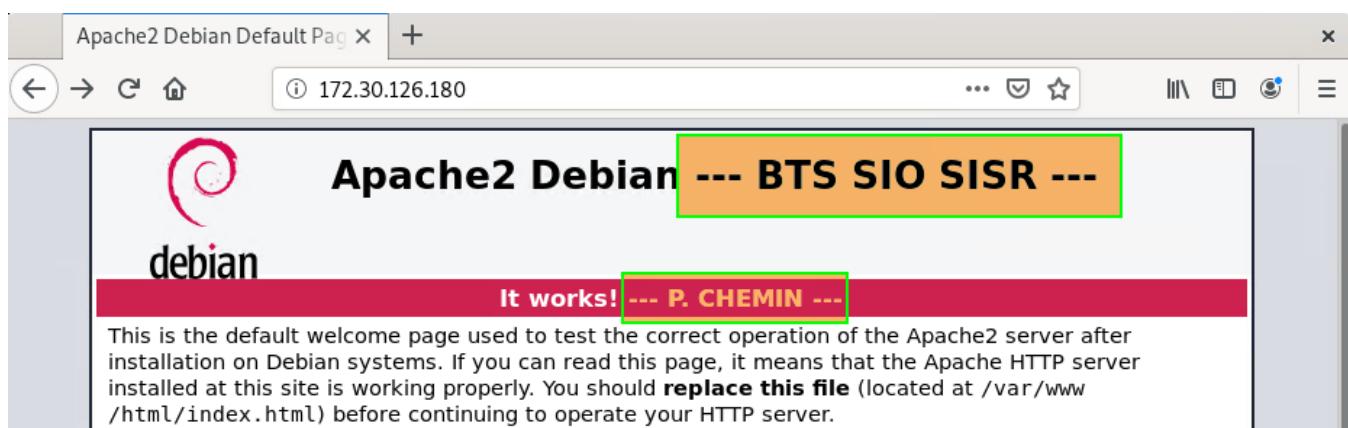
Voici un extrait de ce fichier **index.html**.

```

188 <body>
189   <div class="main_page">
190     <div class="page_header floating_element">
191       
192       <span class="floating_element">
193         Apache2 Debian Default Page
194       </span>
195     </div>
196   <!--
197     <div class="table_of_contents floating_element">
198       <div class="section_header section_header_grey">
199         TABLE OF CONTENTS
200       </div>
201       <div class="table_of_contents_item floating_element">
202         <a href="#about">About</a>
203       </div>
204       <div class="table_of_contents_item floating_element">
205         <a href="#changes">Changes</a>
206       </div>
207       <div class="table_of_contents_item floating_element">
208         <a href="#scope">Scope</a>
209       </div>
210       <div class="table_of_contents_item floating_element">
211         <a href="#files">Config files</a>
212       </div>
213   -->
214   <div class="content_section floating_element">
215
216
217     <div class="section_header section_header_red">
218       <div id="about"></div>
219       It works!
220     </div>
221     <div class="content_section_text">
222       <p>
223         This is the default welcome page used to test the correct

```

Voici mon affichage après deux modifications dans ce fichier index.html



- Effectuer les modifications nécessaires pour obtenir ce résultat avec votre nom.

Avant de modifier un fichier de configuration, vous devez le sauvegarder : montrer comment sauvegarder le fichier index.html en index.htmlSAVE. Créer ce fichier.

- Maintenant que vous avez une version de sauvegarde, supprimer tout le code du fichier index.html et remplacer le par celui-ci :

```
<html><body><h1>En fonctionnement !!! --- VotreNom B1SISR --- </h1>
<p>En construction ...</p>
</body></html>
```

## 7. Montrer le résultat sur le navigateur des postes client.

Il est également possible de visualiser la page à partir de votre serveur WEB. Pour cela, il faut installer lynx, le navigateur en mode console :

```
# apt-get install lynx
```

Puis, on accède à la page :

```
# lynx http://127.0.0.1/ ou      lynx http://172.30.126.180
```

## 8. Tester localement la nouvelle page d'accueil.

## 5. Les fichiers de configuration du service

La configuration d'Apache est située dans le répertoire /etc/apache2/.

```
root@pchem-D11-HTTP:~# cd /etc/apache2/
root@pchem-D11-HTTP:/etc/apache2# l
total 80
-rw-r--r-- 1 root root 7224 10 juil. 2021 apache2.conf
drwxr-xr-x 2 root root 4096 31 janv. 09:59 conf-available
drwxr-xr-x 2 root root 4096 31 janv. 09:59 conf-enabled
-rw-r--r-- 1 root root 1782 8 août 2020 envvars
-rw-r--r-- 1 root root 31063 8 août 2020 magic
drwxr-xr-x 2 root root 12288 31 janv. 09:59 mods-available
drwxr-xr-x 2 root root 4096 31 janv. 09:59 mods-enabled
-rw-r--r-- 1 root root 320 8 août 2020 ports.conf
drwxr-xr-x 2 root root 4096 31 janv. 09:59 sites-available
drwxr-xr-x 2 root root 4096 31 janv. 09:59 sites-enabled
root@pchem-D11-HTTP:/etc/apache2#
```

- **apache2.conf** est le fichier principal de configuration et il contient les directives de configuration.

Il se charge aussi d'inclure les autres fichiers de configuration ;

- **ports.conf** contient la directive Listen qui spécifie les adresses et les ports d'écoutes ;
- **envvars** est utilisé pour définir des variables d'environnement propres à Apache, magic est lui utilisé pour déterminer le type de contenu d'un document en regardant les quelques premiers octets de ce contenu ;
- Les répertoires **conf-available** ou **enabled** contiennent des fichiers de configuration de la police de caractères, les fichiers journaux, la sécurité, ... ;
- Les répertoires **mods-available** ou **enabled** contiennent des modules complémentaires au service apache ;
- Les répertoires **sites-available** ou **enabled** contiennent les fichiers de configuration de chaque site web de ce serveur.

Ces 3 fois 2 = 6 derniers répertoires cités, fonctionnent de la même manière. Le répertoire xxxx-available contient tous les fichiers (de configuration, module ou site) et le répertoire xxxx-enabled contient seulement ceux qui sont activés via un fichier « lien symbolique ».

## 9. Montrer les configurations et les sites activés (enabled).

L'ensemble de ces fichiers définissent une configuration par défaut qui rend le serveur HTTP Apache fonctionnel dès le départ.

L'opérateur pipe (|) permet de rediriger le résultat de la commande précédente vers la commande suivante.

### ➤ Port d'écoute du serveur ?

```
# cat /etc/apache2/ports.conf | grep -i "listen"
```

### ➤ Racine des documents web du serveur ?

```
# cat /etc/apache2/sites-enabled/000-default.conf | grep -i "root"
```

La commande grep permet de filtrer des lignes sur un motif : ici listen et root.

➤ Testez chaque commande seule

➤ Cherchez avec « man grep » ce que permet le paramètre -i

Ces commandes contiennent un opérateur de redirection (| : pipe). Elles peuvent être difficiles à maîtriser, tester chaque partie, regarder le contenu des fichiers, ... pour les comprendre.

**Conclusion :** par défaut, le serveur web écoute sur le port 80 (Listen) et renvoie les pages web depuis le répertoire /var/www/html (DocumentRoot).

**10.** Montrer la configuration du port d'écoute et de la racine des documents web de votre serveur.

**11.** Expliquer ce que permet le paramètre « -i » de la commande grep.

## 6. Les hôtes virtuels (Virtual Hosts)

**Un hôte virtuel** est une identité (supplémentaire) assumée par le serveur web.

Par défaut vous pouvez accéder au serveur web par son adresse IP (<http://172.30.126.180>). Dans ce cas, **un seul site web** est accessible sauf à avoir une adresse IP différente pour chaque site.

Le Serveur HTTP Apache 2 est capable de **gérer simultanément plusieurs arborescences Web** grâce à la notion Virtual Hosts (hôtes virtuels).

Pour pouvoir gérer plusieurs hôtes, **apache utilise le nom DNS**. Chaque nom DNS pointe sur la même adresse IP mais le service Apache redirige sur le site concerné grâce au nom DNS.

Pour cela, il faut créer les fichiers de configurations de chaque site dans le répertoire « sites-enabled » et utiliser les commandes suivantes pour activer ou désactiver un site :

```
# a2ensite nomHote : activer un hôte virtuel
# a2dissite nomHote : désactiver un hôte virtuel
```

➤ N'utilisez pas ces commandes pour l'instant, vous allez le faire ci-dessous.

### 6.1. Paramétrage de l'hôte virtuel www.b1sisr-pchem.net

On commence par désactiver la configuration par défaut sur votre installation :

On affiche le nom de la configuration par défaut

```
# ll /etc/apache2/sites-enabled/
```

Vous verrez le nom default ou 000-default.conf ou autre.

Désactiver ensuite l'hôte virtuel par défaut (en remplaçant 000-default.conf si besoin en fonction de votre configuration)

```
# a2dissite 000-default.conf
```

Chaque hôte virtuel supplémentaire est ensuite décrit par un fichier .conf placé dans le répertoire /etc/apache2/sites-available/

Ainsi, la mise en place du domaine www.b1sisr-xxxxx.net se résume à créer le fichier ci-dessous :

```
# nano /etc/apache2/sites-available/www.b1sisr-pchem.net.conf
```

```
1 <VirtualHost *:80>
2 #   Définition du nom utilisé pour le vhost
3   ServerName www.b1sisr-pchem.net
4 #   Définition d'autres noms sous lesquels le serveur répondra
5   ServerAlias www.b1sisr
6 #   Adresse électronique pour être informé des problèmes
7   ServerAdmin webmaster@localhost
8 #   Définition du répertoire racine où sont stockés les fichiers du site
9   DocumentRoot /var/www/html/www.b1sisr-pchem.net
10 </VirtualHost>
```

On va ensuite créer la racine web du serveur et une page d'accueil personnalisée :

```
# mkdir -p /var/www/html/www.b1sisr-pchem.net
# cp /var/www/html/index.html /var/www/html/www.b1sisr-pchem.net/
# nano /var/www/html/www.b1sisr-pchem.net/index.html
```

Placer le contenu suivant dans le fichier index.html :

```
<html><body><h1>Bienvenue sur www.b1sisr-pchem.net </h1>
<p>En construction ...</p>
</body></html>
```

Il faut maintenant activer le « virtual host » :

```
# a2ensite www.b1sisr-pchem.net ##### a2ensite signifie available to enable site
```

La commande a2ensite a créé un lien symbolique du fichier de sites-available/ vers sites-enabled.

**12.** Montrer le résultat de la commande a2ensite que vous venez de créer

Si un serveur DNS n'est pas fonctionnel, on assure alors la résolution de noms en local.

Il faut renseigner le fichier /etc/hosts afin d'assurer la résolution de nom (Nom -> Adresse Ip) des hôtes virtuels que l'on va créer.

➤ Effectuer cette modification sur le serveur http :

```
# nano /etc/hosts
```

```
127.0.0.1 localhost
127.0.1.1 pchem-D11-HTTP
172.30.126.80 www.b1sisr-pchem.net
172.30.126.80 www.b1sisr

# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Tester l'accès au virtual host avec lynx :

```
# lynx http://www.b1sisr-pchem.net/ # et http://www.b1sisr
```

**13.** Montrer l'affichage de la page d'accueil du site [www.b1sisr-pchem.net](http://www.b1sisr-pchem.net) et de l'alias [www.b1sisr](http://www.b1sisr) en local (lynx)

**14.** Montrer l'affichage de la page d'accueil du site [www.b1sisr-pchem.net](http://www.b1sisr-pchem.net) et de l'alias [www.b1sisr](http://www.b1sisr) à partir du navigateur du poste client Debian (plus difficile sur le client Windows pour l'instant).

## **6.2. Créer un hôte virtuel supplémentaire**

Vous allez mettre en application en autonomie la création d'un nouvel hôte virtuelle dont le nom sera [www.PrenomNom.net](http://www.PrenomNom.net) sur le port 8080.

C'est donc un nouveau site (pas d'alias) et vous devrez créer une nouvelle page d'accueil personnalisée.

**15.** Pour chaque étape du nouvel hôte virtuel, montrer le travail effectué. La dernière étape étant l'affichage de la page d'accueil à partir d'un poste client.

## **7. Résumé**

**16.** Effectuez un résumé des commandes à lancer pour

- Installer Apache ;
- Créer un hôte virtuel.

## 8. Complément sur la gestion des ports

Pour voir sur quel port le service web Apache2 écoute, on utilise pour cela la commande suivante :

Proto	Recv-Q	Send-Q	Adresse locale	Adresse distante	Etat	PID/Program name
tcp	0	0	0.0.0.0:22	0.0.0.0:*	LISTEN	451/sshd
tcp6	0	0	:::80	:::*	LISTEN	9268/apache2

La 2<sup>ème</sup> ligne montre que le port 80 est utilisé par le service apache2.

Une fois un fichier de configuration modifié, il est préférable de lancer Apache2 avec la commande « /etc/init.d/apache2 start » car La configuration Debian par défaut nécessite des variables d'environnement définies dans /etc/apache2/envvars et ne sont pas disponibles si Apache2 est démarré directement avec « service apache2 start ».

### 8.1. Ecoute sur plusieurs ports

Pour faire écouter Apache sur plusieurs ports, il suffit de mettre plusieurs lignes "**Listen**" dans le fichier "ports.conf" ...

```

1 # If you just change the port or add more ports here, you will likely also
2 # have to change the VirtualHost statement in
3 # /etc/apache2/sites-enabled/000-default.conf
4
5 Listen 80
6 Listen 8081
7 Listen 8082
8
9 <IfModule ssl_module>
10     Listen 443
11 </IfModule>
12
13 <IfModule mod_gnutls.c>
14     Listen 443
15 </IfModule>
16
17 # vim: syntax=apache ts=4 sw=4 sts=4 sr noet

```

Et ensuite de faire écouter les **virtuals hosts** sur tel ou tel port.

Site par défaut (fichier de configuration 000-default.conf) sur le port 80

```

1 <VirtualHost *:80>
2     # The ServerName directive sets the request scheme, hostname and port that
3     # the server uses to identify itself. This is used when creating
4     # redirection URLs. In the context of virtual hosts, the ServerName
5     # specifies what hostname must appear in the request's Host: header to
6     # match this virtual host. For the default virtual host (this file) this
7     # value is not decisive as it is used as a last resort host regardless.
8     # However, you must set it for any further virtual host explicitly.
9     #ServerName www.example.com
10
11     ServerAdmin webmaster@localhost
12     DocumentRoot /var/www/html
13
14     # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
15     # error, crit, alert, emerg.
16     # It is also possible to configure the loglevel for particular
17     # modules, e.g.
18     #LogLevel info ssl:warn
19
20     ErrorLog ${APACHE_LOG_DIR}/error.log
21     CustomLog ${APACHE_LOG_DIR}/access.log combined
22
23     # For most configuration files from conf-available/, which are
24     # enabled or disabled at a global level, it is possible to
25     # include a line for only one particular virtual host. For example the
26     # following line enables the CGI configuration for this host only
27     # after it has been globally disabled with "a2disconf".
28     #Include conf-available/serve-cgi-bin.conf
29 </VirtualHost>

```

Site [www.b1sisr-pchem.net](http://www.b1sisr-pchem.net) sur le port 8081

```

1 <VirtualHost www.b1sisr-pchem.net:8081>
2 # ServerName définit le nom utilisé pour le vhost.
3     ServerName www.b1sisr-pchem.net
4 # ServerAlias définit les autres sous domaines pour lesquels le serveur répondra.
5     ServerAlias www.b1sisr
6 # ServerAdmin pour spécifier un email pour être informé en cas de problème.
7     ServerAdmin webmaster@localhost
8 # DocumentRoot définit le dossier racine où seront stockés les fichiers du site.
9     DocumentRoot /var/www/html/www.b1sisr-pchem.net
10 </VirtualHost>
```

Site [www.pascal.net](http://www.pascal.net) sur le port 8082

```

1 <VirtualHost www.pascal.net:8082>
2 # ServerName définit le nom utilisé pour le vhost.
3     ServerName www.pascal.net
4 # ServerAlias définit les autres sous domaines pour lesquels le serveur répondra.
5     ServerAlias www.pascal2
6 # ServerAdmin pour spécifier un email pour être informé en cas de problème.
7     ServerAdmin webmaster@localhost
8 # DocumentRoot définit le dossier racine où seront stockés les fichiers du site.
9     DocumentRoot /var/www/html/www.pascal.net
10 </VirtualHost>
```

## 8.2. Vérifications sur le poste client.

172.30.126.180/    b1sisr-pchem.net:8081/    pascal.net/www.pascal.net/

En fonctionnement !!!! --- P. CHEMIN B1SISR

page par dÃ©faut dans /var/www/html/index.html

SITE 1111

172.30.126.180/    b1sisr-pchem.net:8081/    pascal.net/www.pascal.net/

Bienvenue sur le site www.b1sisr-pchem.net

SITE 222

172.30.126.180/    b1sisr-pchem.net:8081/    pascal.net:8082/

Site www.pascal.net

SITE 333

page /var/www/html/www.pascal.net/index.html

**17.** Effectuez les modifications pour obtenir le même résultat :

- Chaque site sur un port spécifique
- Chaque page par défaut personnalisée