



Lycée Générale et Technologie Baimbride
Brevet Technicien Supérieur
Services informatiques aux Organisations



Déploiement du service Web Apache 2.0 sous Linux Debian 12.0

Compte rendu

TP2 bloc2



Réalisé par ROSIER Matthias BTS SIO 2A

Sommaire

Compte rendu TPn°2_Bloc_n°2_SISR.....	3
Introduction.....	3
A) Contexte du TP	3
B) objectifs et enjeux.....	3
Etape 1 : Réalisation de l'infrastructure réseau	4
A) interface réseau du serveur :.....	4
B) rôle de chaque paramètre :.....	4
C) Teste de connectivité entre les vms :.....	5
D) Ajout enregistrements DNS srvweb et site :.....	5
Etape n°2 : Mise à jour des paquets du SE Linux Debian Bookworm :	8
a) rôle des lignes suivantes :.....	8
b) Deux commandes apt :.....	9
c) Modification nom de la machine :.....	9
Etape n°3 : Déploiement du service web Apache 2.0 :	10
A) teste du bon fonctionnement D'APACHE :.....	10
B) rôle des lignes avec « Include » :.....	10
C) fichiers Contenus dans <<apChe2.Conf>>:.....	11
D) répertoire dans lequel se trouve la page web par défaut apache2 :.....	12
E) Création sous-répertoire et page html :.....	12
F) rôle de Chaque paramètre :.....	14
G) tester l'accès à la page web :.....	15
H) rôle de la Commande « Ps aux grep apaChe2 »:.....	16
I) rôle de la Commande « netstat -napt » :.....	18
J) modifier le numéro de port d'exécution d'apache:	18
Etape n°4 : Contrôle d'accès au site par authentification :	20
Conclusion :	25
Fin du TP :	26

COMPTE RENDU TPN°2_BLOC_N°2_SISR

INTRODUCTION

A) CONTEXTE DU TP

Ce TP traite de l'installation et de la configuration du **service web Apache 2.0** sur un **serveur Linux Debian 12.0** pour répondre aux besoins de la mairie des Abymes. L'objectif est de migrer le site web vers une infrastructure **open-source**, tout en garantissant une **gestion simplifiée** et **sécurisée** de **l'hébergement**. En simulant un environnement réel, ce projet permet d'améliorer la flexibilité et la performance du **site**, tout en **réduisant les coûts** liés aux logiciels propriétaires.

B) OBJECTIFS ET ENJEUX

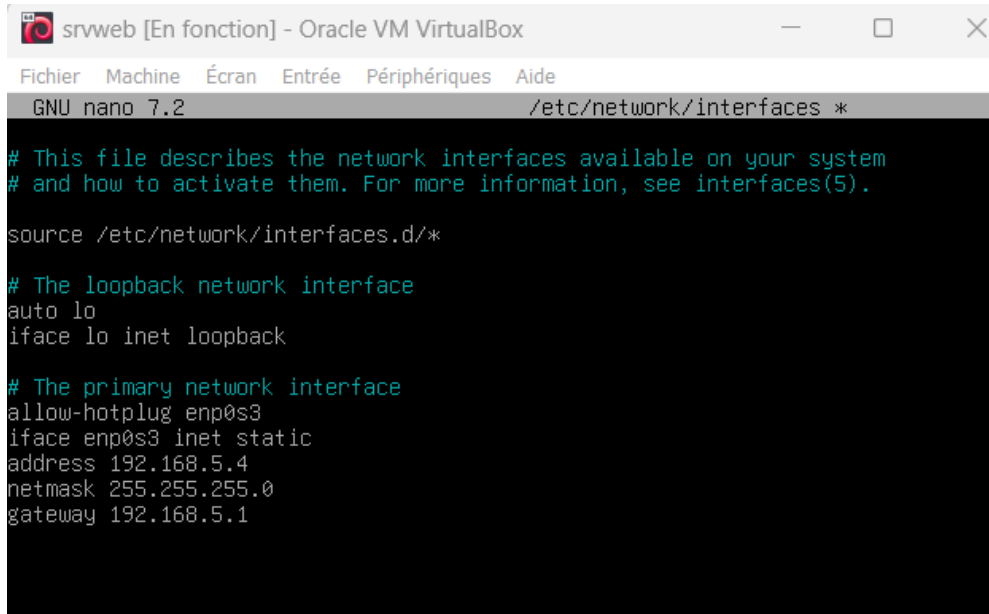
Acquérir et mettre en application les compétences suivantes :

- *Installer et configurer des éléments d'infrastructure
- *Déployer une solution d'infrastructure
- *Administrer un système
- *Tester l'intégration et l'acceptation d'une solution d'infrastructure
- *Rédiger ou mettre à jour la documentation technique et utilisateur d'une solution d'infrastructure

Condition de travail : Individuel

ETAPE 1 : RÉALISATION DE L'INFRASTRUCTURE RÉSEAU

A) INTERFACE RÉSEAU DU SERVEUR :



The screenshot shows a window titled "srvweb [En fonction] - Oracle VM VirtualBox". Inside, the "Fichier" menu is open, and the "Machine" option is selected. The main area displays the contents of the file "/etc/network/interfaces" using the nano 7.2 editor. The file contains the following configuration:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug enp0s3
iface enp0s3 inet static
address 192.168.5.4
netmask 255.255.255.0
gateway 192.168.5.1
```

B) RÔLE DE CHAQUE PARAMÈTRE :

- **auto enp0sX** : Active automatiquement l'interface réseau au démarrage.
- **iface enp0sX inet dhcp** : Configure l'interface pour obtenir une adresse IP via DHCP.
- **iface enp0sX inet static** : Configure l'interface avec une adresse IP statique.
- **address** : Attribue l'adresse IP manuellement à l'interface.
- **netmask** : Définit le masque de sous-réseau pour l'adresse IP.
- **gateway** : Définit l'adresse IP de la passerelle pour accéder à d'autres réseaux.

C) TESTE DE CONNECTIVITÉ ENTRE LES VMS :

The image displays three separate Oracle VM VirtualBox windows, each showing a command prompt with ping test results between different Virtual Machines (VMS).

- srvad [En fonction] - Oracle VM VirtualBox:** Shows a successful ping from Administrator@srvad to 192.168.5.4 (poste_client_ad) and 192.168.5.10 (srvweb). Statistics show 4 packets sent, 4 received, 0 loss, and an average time of 1ms.
- poste_client_ad [En fonction] - Oracle VM VirtualBox:** Shows a successful ping from Administrator@poste_client_ad to 192.168.5.4 (srvad) and 192.168.5.2 (srvweb). Statistics show 4 packets sent, 4 received, 0 loss, and an average time of 1ms.
- srvweb [En fonction] - Oracle VM VirtualBox:** Shows a successful ping from Administrator@srvweb to 192.168.5.10 (poste_client_ad) and 192.168.5.4 (srvad). Statistics show 4 packets sent, 4 received, 0 loss, and an average time of 1ms.

Orange labels indicate the IP addresses for each VM:

- Adresse IP srvad : 192.168.5.2
- Adresse IP poste1 : 192.168.5.10
- Adresse IP srvweb : 192.168.5.4

D) AJOUT ENREGISTREMENTS DNS SRVWEB ET SITE :

Zone de recherche direct :

The image shows the Windows DNS Manager console for the 'ville-abymes.fr' zone. The 'Zones de recherche direct' (Direct Search Zones) tab is selected, displaying a list of records. A red box highlights the records for 'srvweb', 'www2', and 'www'.

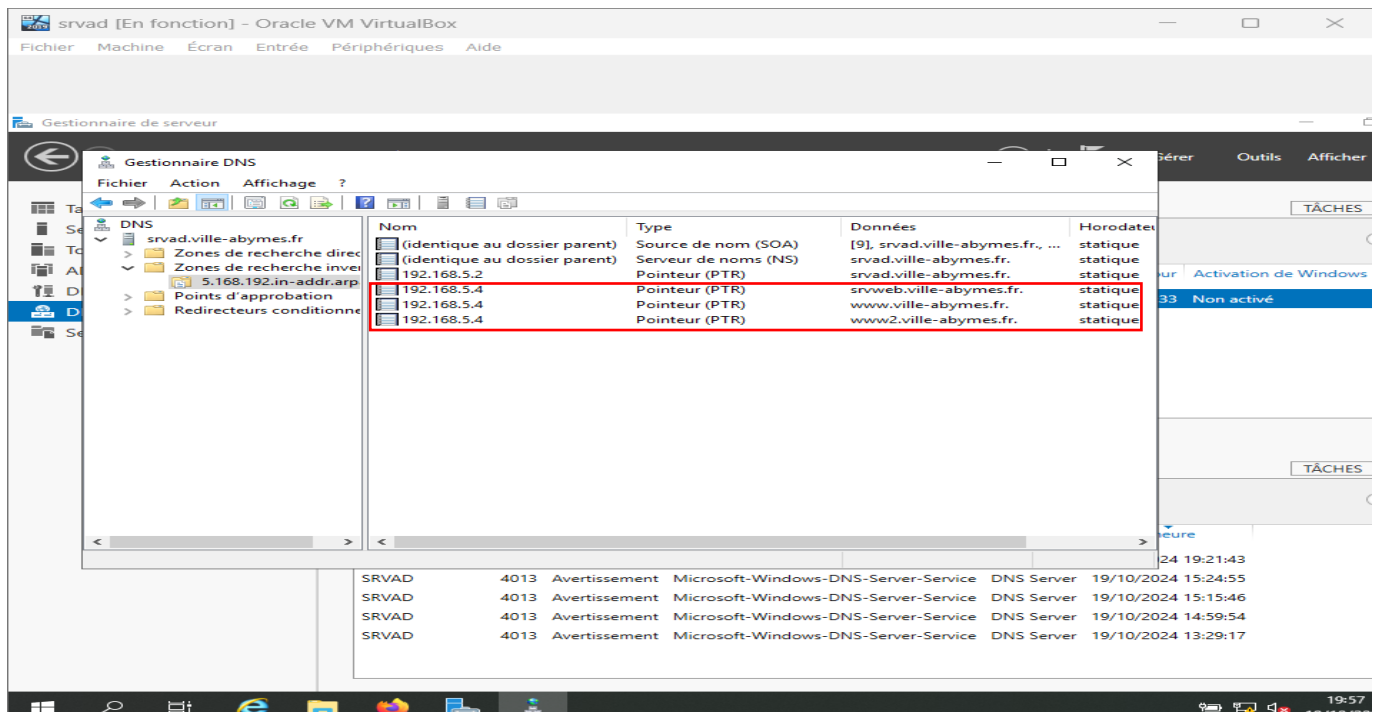
Nom	Type	Données	Horodateur
_msdcs			
_sites			
_tcp			
_udp			
DomainDnsZones			
ForestDnsZones			
(identique au dossier parent)	Source de nom (SOA)	[88], srvad.ville-abymes.fr...	statique
(identique au dossier parent)	Serveur de noms (NS)	srvad.ville-abymes.fr...	statique
(identique au dossier parent)	Hôte (A)	192.168.5.2	19/10/2024
poste1	Hôte (A)	192.168.5.10	19/10/2024
srvad	Hôte (A)	192.168.5.2	statique
srvweb	Hôte (A)	192.168.5.4	
www2	Hôte (A)	192.168.5.4	
www	Hôte (A)	192.168.5.4	

Below the table, there are three log entries for 'SRVAD' with a warning level (4013) and the message 'Microsoft-Windows-DNS-Server-Service'.

- On peut configurer les **enregistrements DNS** dans le **gestionnaire DNS du serveur Windows(srvad)** contenant active directory et le fonctionnalité DHCP et DNS. D'abord pour le **nom du seveurs (srvweb)**, puis les **alias (www/www2)** afin de permettre la résolution des **noms d'hôtes vers l'adresse IP** du serveur Linux srvweb (**192.168.5.4**).

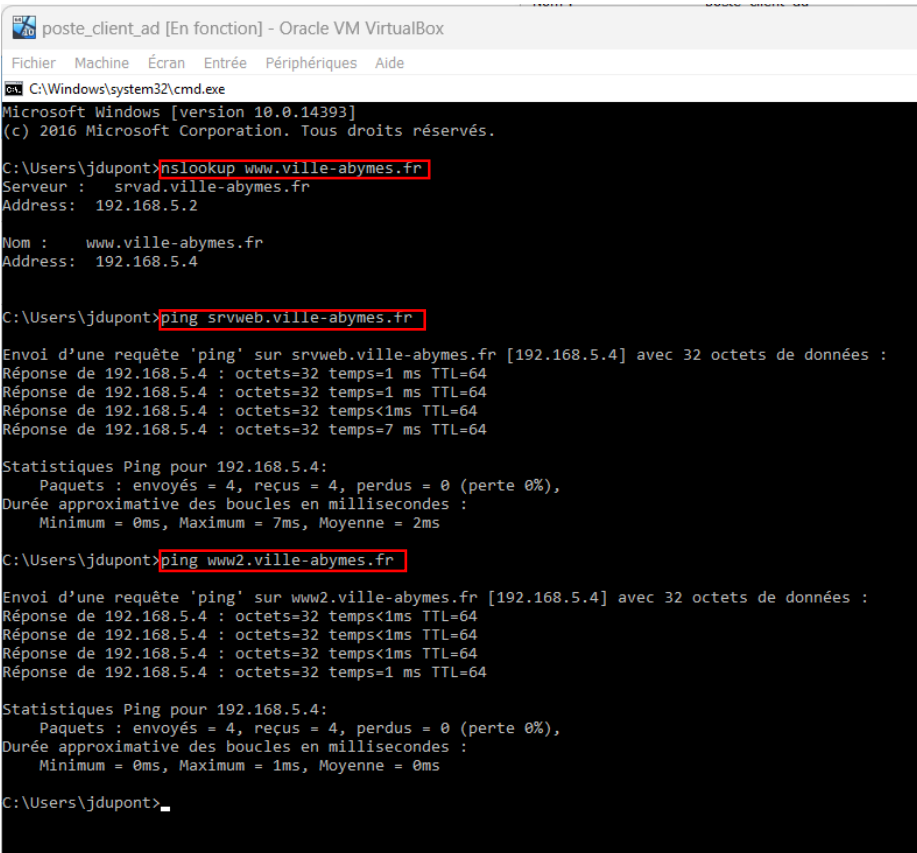
Cela permet aux utilisateurs du réseau d'accéder au serveur et aux services web hébergés via des noms faciles à mémoriser, plutôt que de devoir connaître l'adresse IP.

Zone de recherche inverse :



- Elles ont ici été créés automatiquement en cochant la case **Créer un pointeur d'enregistrement PTR associé**. L'ajout de la zone de recherche inversée (PTR) assure également la **résolution inverse**, améliorant ainsi la **cohérence des réponses DNS**.

Tests recherche direct de ping et nslookup sur le poste1(Windows) qui confirme le bon fonctionnement de cette configuration.



```
poste_client_ad [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide

C:\Windows\system32\cmd.exe
Microsoft Windows [version 10.0.14393]
(c) 2016 Microsoft Corporation. Tous droits réservés.

C:\Users\jdupont>nslookup www.ville-abymes.fr
Serveur : srvad.ville-abymes.fr
Address: 192.168.5.2

Nom : www.ville-abymes.fr
Address: 192.168.5.4

C:\Users\jdupont>ping srvweb.ville-abymes.fr

Envoi d'une requête 'ping' sur srvweb.ville-abymes.fr [192.168.5.4] avec 32 octets de données :
Réponse de 192.168.5.4 : octets=32 temps=1 ms TTL=64
Réponse de 192.168.5.4 : octets=32 temps=1 ms TTL=64
Réponse de 192.168.5.4 : octets=32 temps<1ms TTL=64
Réponse de 192.168.5.4 : octets=32 temps=7 ms TTL=64

Statistiques Ping pour 192.168.5.4:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 0ms, Maximum = 7ms, Moyenne = 2ms

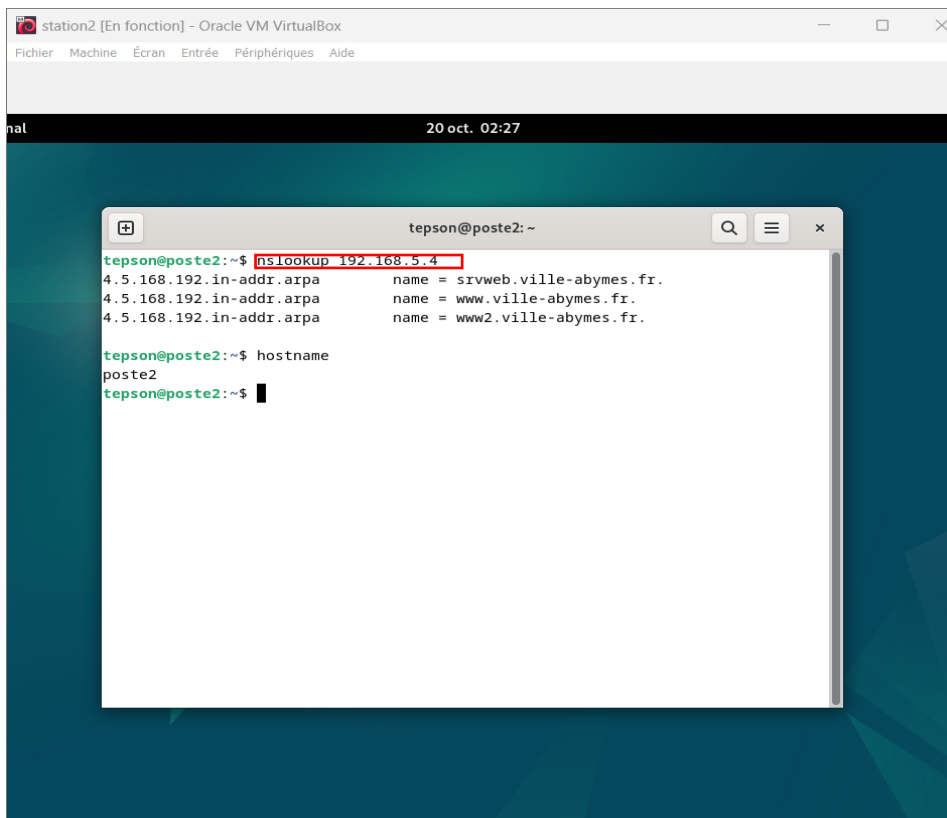
C:\Users\jdupont>ping www2.ville-abymes.fr

Envoi d'une requête 'ping' sur www2.ville-abymes.fr [192.168.5.4] avec 32 octets de données :
Réponse de 192.168.5.4 : octets=32 temps<1ms TTL=64
Réponse de 192.168.5.4 : octets=32 temps<1ms TTL=64
Réponse de 192.168.5.4 : octets=32 temps<1ms TTL=64
Réponse de 192.168.5.4 : octets=32 temps=1 ms TTL=64

Statistiques Ping pour 192.168.5.4:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 0ms, Maximum = 1ms, Moyenne = 0ms

C:\Users\jdupont>
```

Tests recherche indirect nslookup sur le poste2(Debian) qui confirme le bon fonctionnement de cette configuration.



```
station2 [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide

20 oct. 02:27

tepson@poste2: ~
tepson@poste2:~$ nslookup 192.168.5.4
4.5.168.192.in-addr.arpa    name = srvweb.ville-abymes.fr.
4.5.168.192.in-addr.arpa    name = www.ville-abymes.fr.
4.5.168.192.in-addr.arpa    name = www2.ville-abymes.fr.

tepson@poste2:~$ hostname
poste2
tepson@poste2:~$
```

ETAPE N°2 : MISE À JOUR DES PAQUETS DU SE LINUX DEBIAN BOOKWORM :

A) RÔLE DES LIGNES SUIVANTES :

```
GNU nano 7.2 /etc/apt/sources.list
deb http://deb.debian.org/debian/ bookworm main contrib non-free
deb-src http://deb.debian.org/debian/ bookworm main contrib non-free

deb http://deb.debian.org/debian-security bookworm-security main contrib non-free
deb-src http://deb.debian.org/debian-security bookworm-security main contrib non-free

# bookworm-updates, to get updates before a point release is made;
# see https://www.debian.org/doc/manuals/debian-reference/ch02.en.html#_updates_and_backports
deb http://deb.debian.org/debian/ bookworm-updates main contrib non-free
deb-src http://deb.debian.org/debian/ bookworm-updates main contrib non-free
```

Ces

lignes correspondent aux **sites miroirs** (ou **dépôts de logiciels**) qui sont des **serveurs distants** utilisés par **Debian** pour télécharger et gérer les **paquets logiciels** (applications, bibliothèques, utilitaires, etc.). Ils servent de références pour installer et mettre à jour les logiciels.

- **deb** : Utilisé pour installer des logiciels déjà compilés (binaries).
- **deb-src** : Utilisé pour obtenir les codes sources des logiciels.
- **main** : Contient les logiciels libres et approuvés par Debian.
- **contrib** : Contient les logiciels libres, mais dépendant de composants non libres.
- **non-free** : Contient les logiciels propriétaires (non libres).
- **bookworm-security** : Sert à obtenir les mises à jour de sécurité.
- **bookworm-updates** : Fournit les mises à jour générales avant la prochaine version stable.

Ces lignes permettent donc d'installer de **nouveaux logiciels**, d'obtenir les **versions** les plus récentes, et d'appliquer des **mises à jour**, y compris de **sécurité**.

B) DEUX COMMANDES APT :

- **# apt update** → Met à jour la **liste des paquets disponibles** dans les dépôts de logiciels.

Elle ne met pas à jour les logiciels eux-mêmes, mais simplement la liste des versions disponibles.

- **# apt upgrade** → Met à jour **les paquets logiciels installés** sur le système.

Elle télécharge et installe les nouvelles versions des logiciels après que la liste des paquets a été mise à jour.

C) MODIFICATION NOM DE LA MACHINE :

- Grâce à la commande << **# nano /etc/hostname** >>, on doit changer le nom de **debian** à **srvweb** :

```
GNU nano 7.2 /etc/hostname
debian
```



```
GNU nano 7.2 /etc/hostname
srvweb
```

- Redémarrer la vm avec la commande << **# sudo reboot** >> et voici les changements observables :

```
srvweb [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
Debian GNU/Linux 12 srvweb tty1
srvweb login:
Le nom n'est plus
```

```
srvweb [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
user@srvweb:~$ su
Mot de passe :
root@srvweb: /home/user#
```

ETAPE N°3 : DÉPLOIEMENT DU SERVICE WEB APACHE 2.0 :

A) TESTE DU BON FONCTIONNEMENT D'APACHE :

- Grâce à la commande on peut voir l'état du service apache si il est actif ou non << **# sytemctl status apache2** >> :

```
root@srvweb:/home/user# systemctl status apache2
• apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
  Active: active (running) since Sun 2024-10-20 01:22:35 CEST; 1h 27min ago
  Docs: https://httpd.apache.org/docs/2.4/
  Process: 480 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Main PID: 515 (apache2)
  Tasks: 55 (limit: 2315)
  Memory: 12.4M
  CPU: 700ms
  CGroup: /system.slice/apache2.service
          └─515 /usr/sbin/apache2 -k start
            └─517 /usr/sbin/apache2 -k start
              └─518 /usr/sbin/apache2 -k start

oct. 20 01:22:22 srvweb systemd[1]: Starting apache2.service - The Apache HTTP Server...
oct. 20 01:22:35 srvweb apachectl[507]: AH00557: apache2: apr_sockaddr_info_get() failed for srvweb
oct. 20 01:22:35 srvweb apachectl[507]: AH00558: apache2: Could not reliably determine the server's
oct. 20 01:22:35 srvweb systemd[1]: Started apache2.service - The Apache HTTP Server.
lines 1-18/18 (END)
```

B) RÔLE DES LIGNES AVEC « INCLUDE » :

- Les lignes commençant par <<**Include**>> permettent à Apache d'**inclure** des **fichiers de configuration supplémentaires**. Ces fichiers peuvent contenir des configurations pour des **modules**, des **sites**, ou des **fonctionnalités spécifiques**. Cela permet d'avoir un fichier de **configuration centralisé (apache2.conf)** tout en gardant des configurations plus **détaillées dans des fichiers séparés**.

Exemple :

```
# Include module configuration:
IncludeOptional mods-enabled/*.load
IncludeOptional mods-enabled/*.conf

# Include list of ports to listen on
Include ports.conf
```

```
# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf
```

C) FICHIERS CONTENUS DANS <<APACHE2.CONF>>:

- **ports.conf** : Fichier utilisé pour définir les ports d'écoute d'Apache. Il est inclus par le fichier principal et peut être modifié pour changer les ports par défaut (HTTP : 80, HTTPS : 443).
- **mods-enabled/** : Répertoire contenant les fichiers de configuration pour les modules activés. Les modules permettent d'ajouter des fonctionnalités à Apache (ex. : PHP, SSL).
- **conf-enabled/** : Répertoire contenant des fichiers de configuration généraux activés. Ce sont des configurations qui ne sont pas liées à un site spécifique, mais qui affectent le serveur dans son ensemble.
- **sites-enabled/** : Répertoire contenant les configurations des hôtes virtuels activés. Chaque site ou application web a son propre fichier de configuration dans ce répertoire.
- **Mutex file** : Fichier utilisé pour gérer les verrous mutuels dans Apache, garantissant qu'un processus à la fois peut accéder à certaines ressources partagées.
- **DefaultRuntimeDir** : Répertoire où sont stockés les fichiers temporaires et de runtime d'Apache, comme les fichiers de verrouillage ou les PID.
- **PidFile** : Fichier qui contient le PID (Process ID) du processus Apache, permettant de gérer et surveiller le serveur web.
- **ErrorLog** : Définit l'emplacement du fichier de log des erreurs d'Apache, où les erreurs du serveur sont enregistrées.
- **AccessFileName** : Nom du fichier de configuration supplémentaire, généralement .htaccess, utilisé pour définir des directives spécifiques à certains répertoires.
- **IncludeOptional mods-enabled/*.load** : Inclut les fichiers de configuration pour charger les modules activés.
- **IncludeOptional mods-enabled/*.conf** : Inclut les fichiers de configuration pour configurer les modules activés.
- **IncludeOptional conf-enabled/*.conf** : Inclut des fichiers de configuration généraux, appliqués à tout le serveur.
- **IncludeOptional sites-enabled/*.conf** : Inclut les fichiers de configuration des hôtes virtuels, activant ainsi les sites web définis.

D) RÉPERTOIRE DANS LEQUEL SE TROUVE LA PAGE WEB
PAR DÉFAUT APACHE2 :

- Le répertoire dans lequel se trouve la **page par défaut** qui s'affiche une fois que le **service web Apache** est installé est : `<< /var/www/html/ >>` plus précisément dans le fichier `<< index.html >>`.

```
root@srvweb:/var/www/html# ls
index.html
root@srvweb:/var/www/html#
```

```
GNU nano 7.2 index.html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Debian Default Page: It works</title>
    <style type="text/css" media="screen">
      * {
        margin: 0px 0px 0px 0px;
        padding: 0px 0px 0px 0px;
      }

      body, html {
        padding: 3px 3px 3px 3px;

        background-color: #D8DBE2;

        font-family: Verdana, sans-serif;
        font-size: 11pt;
        text-align: center;
      }

      div.main_page {
        position: relative;
        display: table;

        width: 800px;

        margin-bottom: 3px;
        margin-left: auto;
        margin-right: auto;
        padding: 0px 0px 0px 0px;

```

E) CRÉATION SOUS-RÉPERTOIRE ET PAGE HTML :

Création du sous-répertoire « villeabymes » :

Avec la commande << # **sudo mkdir /var/www/villeabymes** >>

Elle crée un sous-répertoire villeabymes dans le répertoire principal des sites web /var/www/. Ce répertoire servira à stocker les fichiers du site.

Création de la page d'accueil HTML :

Avec la commande << # **sudo nano /var/www/villeabymes/index.html** >>

La commande ci-dessus ouvre l'éditeur de texte **nano** pour créer la page d'accueil **index.html**. Cette page contient un message simple souhaitant la bienvenue aux visiteurs, avec le titre "Site ville des Abymes".

Création du fichier de configuration « villeabymes.conf » :

Pour se placer dans le répertoire « `/etc/apache2/sites-available/` » utilisez la commande : `<< # cd /etc/apache2/sites-available/ >>`

Copie du fichier de configuration par défaut :

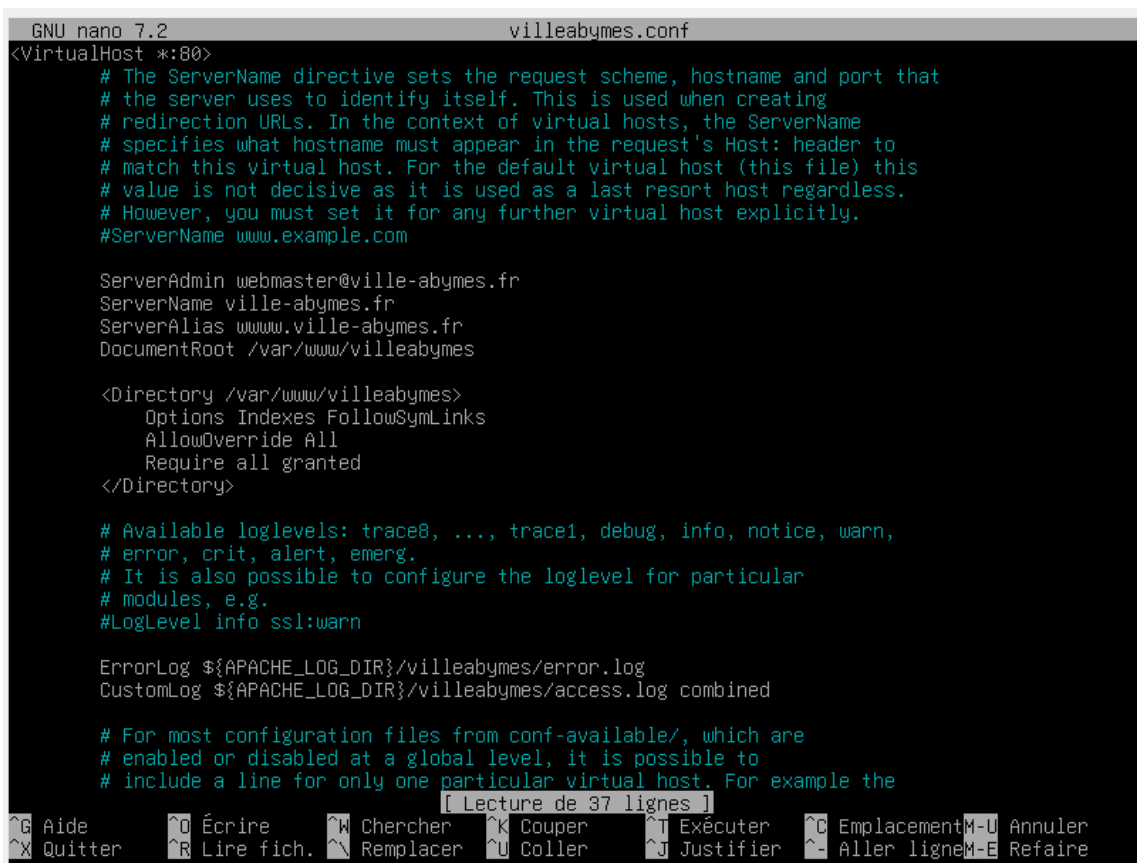
Pour créer une copie du fichier « `000-default.conf` » et la nommée « `villeabymes.conf` » utilisez la commande : `<< # cp 000-default.conf villeabymes.conf >>`

On peut vérifier que la copie a été correctement réalisée en listant les fichiers dans le répertoire à l'aide la commande : `# ls`



```
srvweb [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
root@srvweb:/etc/apache2/sites-available# ls
000-default.conf default-ssl.conf sainteloi.conf villeabymes.conf
root@srvweb:/etc/apache2/sites-available# _
```

Configuration hôte virtuel :



```
GNU nano 7.2 villeabymes.conf
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@ville-abymes.fr
    ServerName ville-abymes.fr
    ServerAlias www.ville-abymes.fr
    DocumentRoot /var/www/villeabymes

    <Directory /var/www/villeabymes>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/villeabymes/error.log
    CustomLog ${APACHE_LOG_DIR}/villeabymes/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    [ Lecture de 37 lignes ]

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C EmplacementM-U Annuler
^X Quitter   ^R Lire fich.^M Remplacer  ^U Coller    ^J Justifier  ^_ Aller ligneM-E Refaire
```

Création du répertoire de logs apache :

Grâce à la commande `<< # mkdir /var/log/apache2/villeabymes >>`, apache2 va pouvoir générer automatiquement les fichiers de logs.

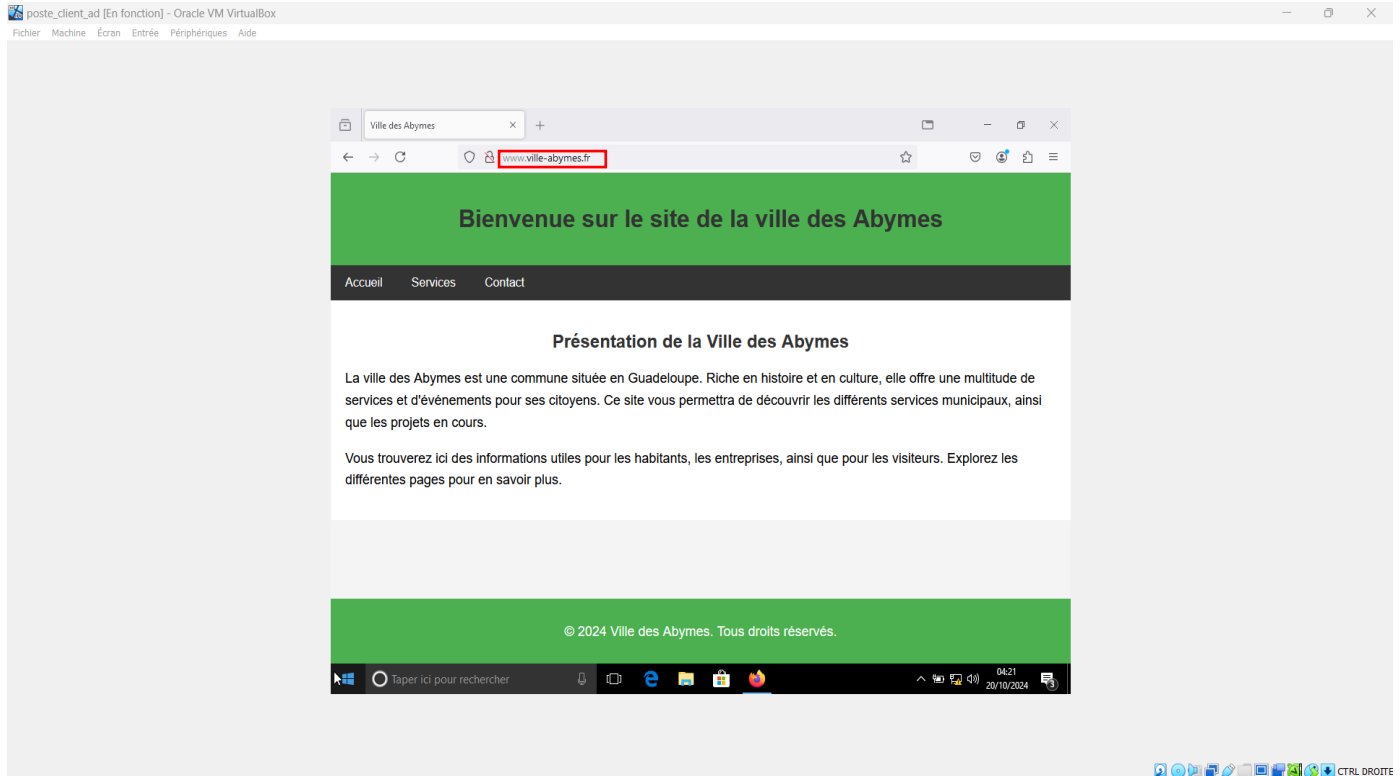
F) RÔLE DE CHAQUE PARAMÈTRE :

- **a2ensite villesabymes.conf** : Active le site villeabymes en créant un lien symbolique vers le fichier de configuration du site dans `/etc/apache2/sites-enabled/`.
Cela permet à Apache de prendre en compte la configuration de ce site.
- **a2dissite 000-default.conf** : Désactive le site par défaut d'Apache en supprimant le lien symbolique dans `/etc/apache2/sites-enabled/`.
Le site par défaut ne sera plus accessible une fois désactivé.
- **a2enmod nom_du_module** : Active un module Apache en créant un lien symbolique vers le module dans `/etc/apache2/mods-enabled/`.
Ce qui permet d'ajouter de nouvelles fonctionnalités à Apache. Exemple le module **rewrite** qui permet de réécrire les URL (utile pour rediriger des pages web).
- **a2dismod nom_du_module** : Désactive un module Apache en supprimant le lien symbolique dans `/etc/apache2/mods-enabled/`.
Désactivant ainsi les fonctionnalités associées au module.
- **systemctl [reload|restart|start|stop] apache2** :
Utilisée pour gérer le service Apache :
 - **reload** : Recharge la configuration d'Apache sans arrêter le service, appliquant ainsi les changements.
 - **restart** : Redémarre complètement Apache pour appliquer les changements.
 - **start** : Démarre Apache s'il est arrêté.
 - **stop** : Arrête le service Apache.

G) TESTER L'ACCÈS À LA PAGE WEB :

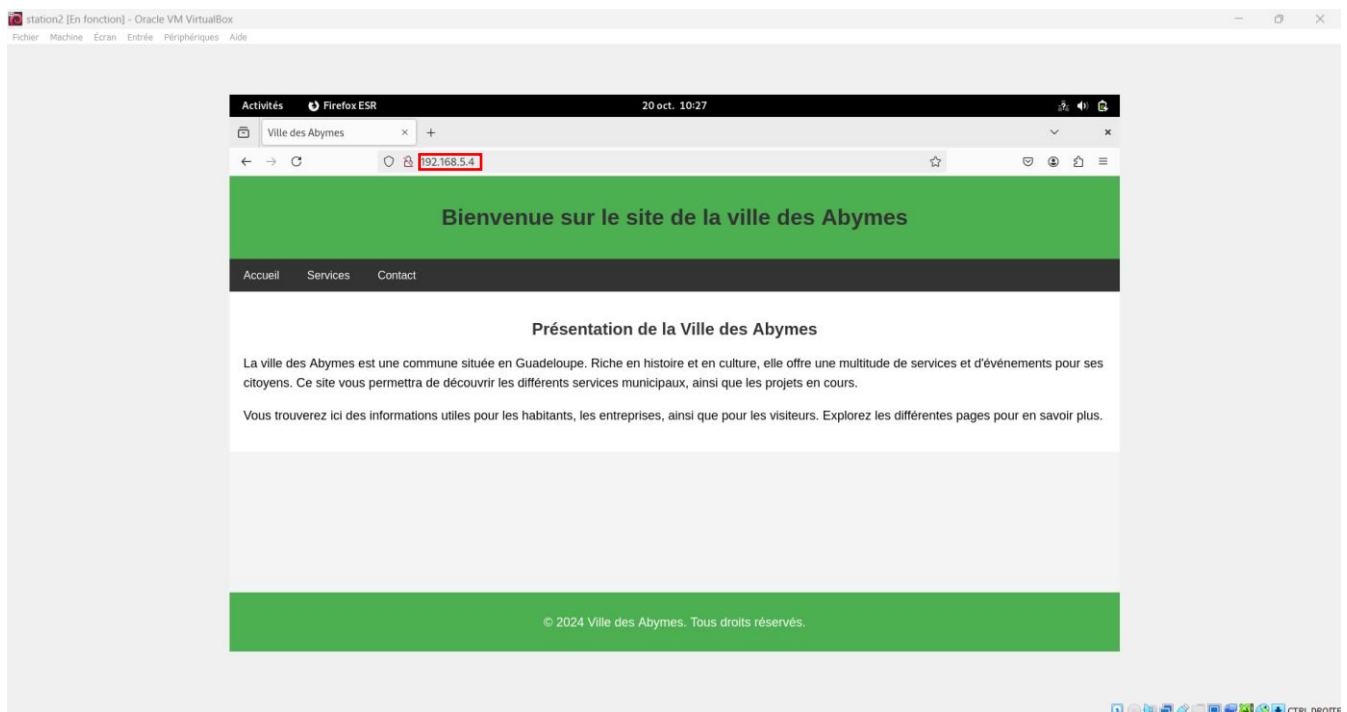
- On peut ouvrir un **navigateur** et saisir l'**URL** dans la barre d'adresse: **www.villeabymes.fr**

Teste d'accès via le nom de domaine sur le poste1 client Windows via Firefox :



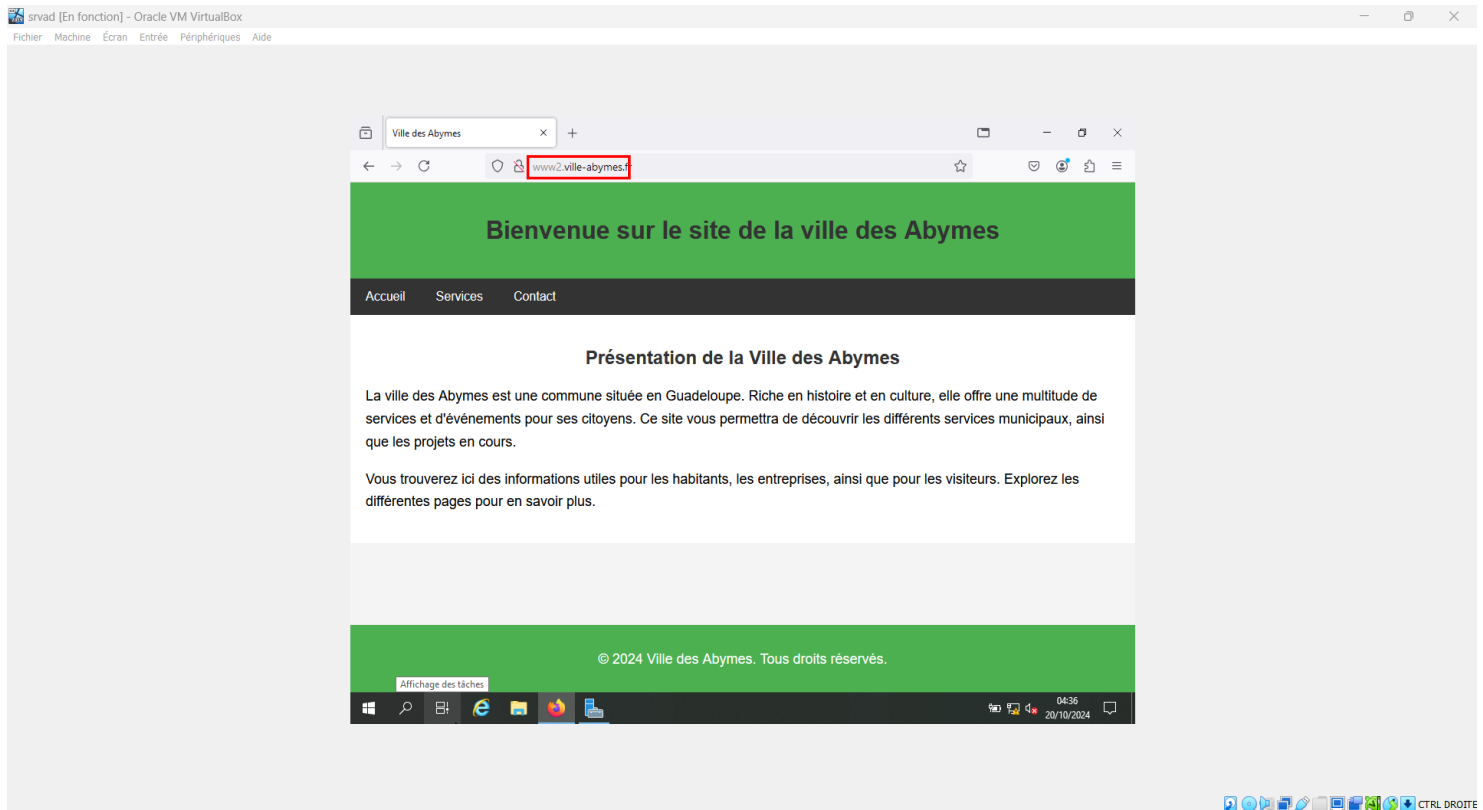
- On peut aussi ouvrir un **navigateur** et saisir l'**URL** dans la barre d'adresse : **www.192.168.5.4**

Teste d'accès via l'adresse IP sur le poste2 client Linux Debian via Firefox :



- On peut ouvrir un **navigateur** et saisir l'**URL** dans la barre d'adresse: **www2.villeabymes.fr**

Teste d'accès via un alias du nom de domaine sur le serveur Windows Active Directory via Firefox :



H) RÔLE DE LA COMMANDE « PS AUX | GREP APACHE2 »:

- **ps aux** : Cette partie de la commande liste tous les processus en cours d'exécution sur le système.
 - **ps** est l'abréviation de "**process status**", qui affiche l'état des processus.
 - **a** demande à **ps** de lister les processus de tous les utilisateurs.
 - **u** affiche des informations détaillées sur chaque processus, comme l'utilisateur, le pourcentage de CPU et de mémoire utilisé, etc.
 - **x** demande à **ps** de lister également les processus sans terminal de contrôle, comme ceux exécutés par des services en arrière-plan.
- **| grep apache2** : La commande **grep** est utilisée pour **filtrer** la sortie de **ps aux**. Ici, on cherche uniquement les lignes qui contiennent le mot "**apache2**", ce qui correspond généralement aux processus liés au **serveur web Apache**.

- **Numéro du PID des processus Apache :**
 - **Processus principal (parent) :** 15151
 - **Processus enfants :** 26102, 26103
- **Nombre de processus Apache lancés :**

Il y a **3 processus Apache** en cours d'exécution :

- **1 processus parent (PID 15151).**
- **2 processus enfants (PIDs 26102 et 26103).**

```
root@srvweb:/etc/apache2/sites-available# ps aux | grep apache2
root      15151  0.0  0.2  6652  5168 ?        Ss   05:34   0:02 /usr/sbin/apache2 -k start
www-data  26102  0.0  0.4 754036  8464 ?        Sl   08:09   0:00 /usr/sbin/apache2 -k start
www-data  26103  0.0  0.3 753900  7948 ?        Sl   08:09   0:00 /usr/sbin/apache2 -k start
root      26219  9.0  0.1  6352  2348 tty1      S+   10:43   0:00 grep apache2
root@srvweb:/etc/apache2/sites-available#
```

- **Paramètre associé dans le fichier de configuration apache2.conf :**

Le nombre de processus enfants est contrôlé par les paramètres dans le fichier de configuration d'Apache, notamment :

- **StartServers** : Définit le nombre initial de processus enfants qu'Apache démarre au lancement.
- **MaxRequestWorkers** : Définit le nombre maximum de processus ou de threads pouvant gérer des requêtes simultanées.

Ces paramètres se trouvent dans le fichier de configuration du module MPM (Multi-Processing Module), par exemple **mpm_prefork.conf** ou **mpm_worker.conf**, selon le module utilisé.

```
GNU nano 7.2                                mpm_worker.conf
# worker MPM
# StartServers: initial number of server processes to start
# MinSpareThreads: minimum number of worker threads which are kept spare
# MaxSpareThreads: maximum number of worker threads which are kept spare
# ThreadLimit: ThreadsPerChild can be changed to this maximum value during a
#               graceful restart. ThreadLimit can only be changed by stopping
#               and starting Apache.
# ThreadsPerChild: constant number of worker threads in each server process
# MaxRequestWorkers: maximum number of threads
# MaxConnectionsPerChild: maximum number of requests a server process serves

StartServers          2
MinSpareThreads       25
MaxSpareThreads       75
ThreadLimit           64
ThreadsPerChild       25
MaxRequestWorkers     150
MaxConnectionsPerChild 0
```

I) RÔLE DE LA COMMANDE « NETSTAT -NAPT » :

- **netstat -napt** : Cette commande permet d'afficher toutes les connexions réseau actives sur un système, en montrant les adresses IP, les ports utilisés, et les processus qui gèrent ces connexions. Elle est utile pour surveiller l'état du réseau et diagnostiquer les problèmes.

Informations relevées :

1. Processus gérant l'exécution du service Apache :

- Le processus qui gère l'exécution du service Apache est nommé **apache2**.

2. Numéro de port :

- Le serveur Apache écoute sur le port **80** avec la configuration **IPv6** :

- **tcp6 :::80** (IPv6)

3. État du processus :

- Le processus Apache est en attente de recevoir des connexions et est dans l'état **LISTEN**, ce qui signifie qu'il est prêt à traiter des requêtes entrantes.

```
root@srvweb:/etc/apache2/mods-available# netstat -napt
Connexions Internet actives (serveurs et établies)
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 15045/sshd: /usr/sb
tcp6 0 0 :::22 :::* LISTEN 15045/sshd: /usr/sb
tcp6 0 0 :::80 :::* LISTEN 15151/apache2
root@srvweb:/etc/apache2/mods-available# [24405.528919] watchdog: BUG: soft lockup - CPU#0 stuck for 27s! [swapper/0:0]
```

J) MODIFIER LE NUMÉRO DE PORT D'EXÉCUTION D'APACHE:

- Pour modifier le numéro de port sur lequel Apache écoute, il faut éditer le fichier :

<< **# nano /etc/apache2/ports.conf** >> et ensuite remplacer << **Listen 80** >> par le nouveau numéro de port << **Listen 8080** >>.

il faut également modifier le fichier **/etc/apache2/sites-available/villeabymes.conf** et remplacer <<**VirtualHost *:80**>> par <<**VirtualHost *:8080**>> pour que le site écoute sur le nouveau port **8080**.

Sauvegardez les changements et redémarrez Apache avec la commande :

<< # systemctl restart apache2 >>

```
GNU nano 7.2 /etc/apache2/ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>
```



```
GNU nano 7.2 /etc/apache2/ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 8080_

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>
```

```
<VirtualHost *:8080>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@ville-abymes.fr
ServerName ville-abymes.fr
ServerAlias www.ville-abymes.fr
DocumentRoot /var/www/villeabymes

<Directory /var/www/villeabymes>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/villeabymes/error.log
CustomLog ${APACHE_LOG_DIR}/villeabymes/access.log combined

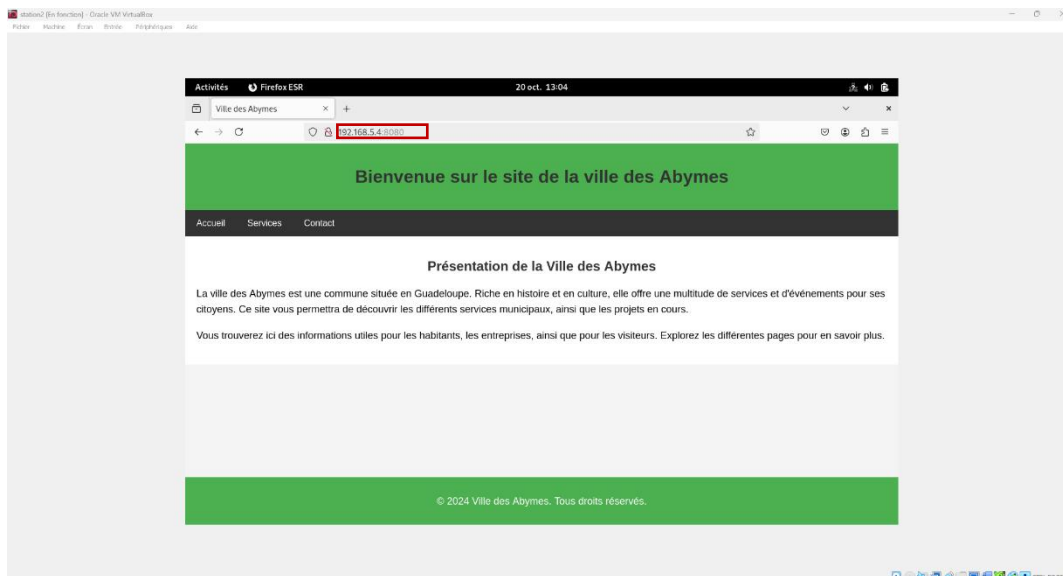
# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the

root@srvweb:/etc/apache2/mods-available# systemctl restart apache2
root@srvweb:/etc/apache2/mods-available# _
```

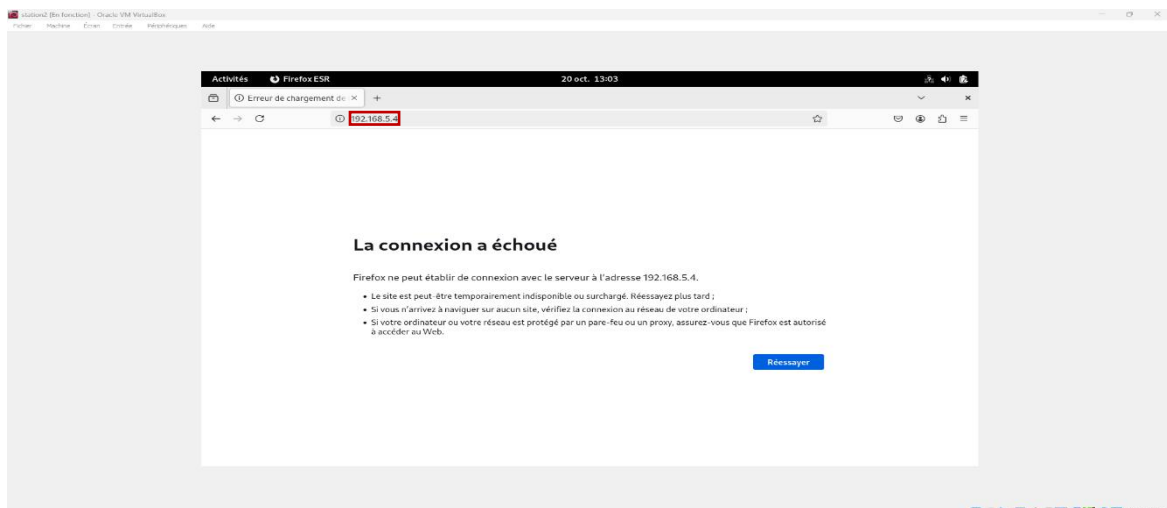
- Pour vérifier que le changement est effectif, on peut utiliser la commande `<< # netstat -napt >>` pour voir si **Apache écoute** sur le **nouveau port 8080**.

```
root@srvweb:/etc/apache2/mods-available# netstat -napt
Connexions Internet actives (serveurs et établies)
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 15045/sshd: /usr/sb
tcp6 0 0 :::8080 :::* LISTEN 26936/apache2
tcp6 0 0 :::22 :::* LISTEN 15045/sshd: /usr/sb
root@srvweb:/etc/apache2/mods-available#
```

- Alternativement, on peut essayer d'accéder à la page par défaut avec `<< http://192.168.1.4:8080 >>` depuis un **navigateur** pour confirmer qu'Apache répond sur le nouveau port.



- Maintenant si on ne met pas le `<< :8080 >>` pour accéder depuis le **navigateur** au site cela **ne passe plus avec le port 80 par défaut**.



ETAPE N°4 : CONTRÔLE D'ACCÈS AU SITE PAR AUTHENTIFICATION :

Création d'un fichier texte nommé « .htusers » dans le répertoire « /etc/apache2 » :

Avec la commande : << # touch /etc/apache2/.htusers>>

Modification du fichier << villeabymes.conf >> pour intégrer l'authentification :

Grâce à : << #nano /etc/apache2/sites-available/villeabymes.conf >>

Ajoutez ou modifiez les lignes suivantes dans le bloc <Directory> :

```
GNU nano 7.2 /etc/apache2/sites-available/villeabymes.conf *
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@ville-abymes.fr
ServerName ville-abymes.fr
ServerAlias www.ville-abymes.fr
DocumentRoot /var/www/villeabymes

<Directory /var/www/villeabymes>
    AuthType Basic
    AuthName "Accès privé -site vile des Abymes"
    AuthBasicProvider file
    AuthUserFile /etc/apache2/.htusers
    Require valid-user
</Directory>

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/villeabymes/error.log
CustomLog ${APACHE_LOG_DIR}/villeabymes/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement M-U Annuler
^X Quitter   ^R Lire fich.^N Remplacer  ^U Coller    ^J Justifier ^_ Aller ligne M-E Refaire
```

- **AuthType Basic** : Définit le type d'authentification, ici **Basic**.
- **AuthName** : Le message affiché lorsque l'utilisateur est invité à entrer ses identifiants.
- **AuthBasicProvider file** : Indique que les informations d'authentification sont stockées dans un fichier.
- **AuthUserFile /etc/apache2/.htusers** : Spécifie l'emplacement du fichier contenant les utilisateurs autorisés.
- **Require valid-user** : Seuls les utilisateurs valides, définis dans **.htusers**, peuvent accéder au site.

Création des utilisateurs avec la commande << # htpasswd >> :

```
root@srvweb:/etc/apache2# htpasswd /etc/apache2/.htusers jdupont
New password:
Re-type new password:
htpasswd: password verification error
root@srvweb:/etc/apache2# htpasswd /etc/apache2/.htusers jdupont
New password:
Re-type new password:
Adding password for user jdupont
root@srvweb:/etc/apache2# htpasswd /etc/apache2/.htusers dlaurent
New password:
Re-type new password:
Adding password for user dlaurent
root@srvweb:/etc/apache2# htpasswd /etc/apache2/.htusers eduvent
New password:
Re-type new password:
Adding password for user eduvent
root@srvweb:/etc/apache2# _
```

Les mots de passe respectifs sont :

jdupont → @jdupont&2425@

dlaurent → @dlaurent&2425@

eduvent → @eduven&2425@

Contenue du fichier << .htusers >>:

```
root@srvweb:/etc/apache2# cat /etc/apache2/.htusers
jdupont:$apr1$u86K5FRp$B9TrpABLLtA0wENEi2xFE/
dlaurent:$apr1$BNmUDmnb$08LQnJT9oSmQHSfF9CrEI.
eduvent:$apr1$epKbeK7B$5Eo1IibdPda0v42KVM3UQ0
root@srvweb:/etc/apache2#
```

- On observe que les mots de passe ne sont pas stockés en texte clair. Ils sont chiffrés (ou hashés) avec un algorithme **MD5**, identifiable par le préfixe **\$apr1\$** devant chaque chaîne de caractères associée aux utilisateurs.
- L'algorithme MD5 utilisé pour chiffrer les mots de passe est ancien et vulnérable aux attaques modernes. Cela signifie qu'il serait possible, avec suffisamment de ressources, de casser ces mots de passe.

Solution : Utilisation de l'option -B pour bcrypt

- L'option **-B** de la commande **htpasswd** permet d'utiliser l'algorithme **bcrypt**, qui est bien plus sécurisé et résistant aux attaques modernes.
- **bcrypt** est conçu pour être plus lent à calculer, ce qui rend les attaques par force brute beaucoup plus difficiles par rapport à des algorithmes comme **MD5**.
- Il intègre un **salt** (valeur aléatoire ajoutée au mot de passe) pour éviter que deux mots de passe identiques génèrent le même hash.

En utilisant **bcrypt**, cela assure que les mots de passe des utilisateurs sont mieux protégés contre les attaques par force brute et autres techniques de cassage de mot de passe. Cela résout le problème de sécurité lié à l'utilisation de **MD5**, qui est obsolète.

```
root@srvweb:/etc/apache2# htpasswd -B /etc/apache2/.htusers jdupont
New password:
Re-type new password:
Updating password for user jdupont
root@srvweb:/etc/apache2# htpasswd -B /etc/apache2/.htusers dlaurent
New password:
Re-type new password:
Updating password for user dlaurent
root@srvweb:/etc/apache2# htpasswd -B /etc/apache2/.htusers eduvent
New password:
Re-type new password:
Updating password for user eduvent
root@srvweb:/etc/apache2# cat /etc/apache2/.htusers
jdupont:$2y$05$.r0ijTiMfsMET8Qoaj3AU0kxHt0qTegnmY.kPWbrq/cWU.RLvTBZe
dlaurent:$2y$05$8aySSi0IuD.1sCXiz62QduRr8QLFYxLfF8J0p9YDh2E11oFy/gGQK
eduvent:$2y$05$y1mG6/0ufPd3jUELbhazuu8CzUawsHTJhbBs56J2/ydoyKTYc7z72
root@srvweb:/etc/apache2#
```

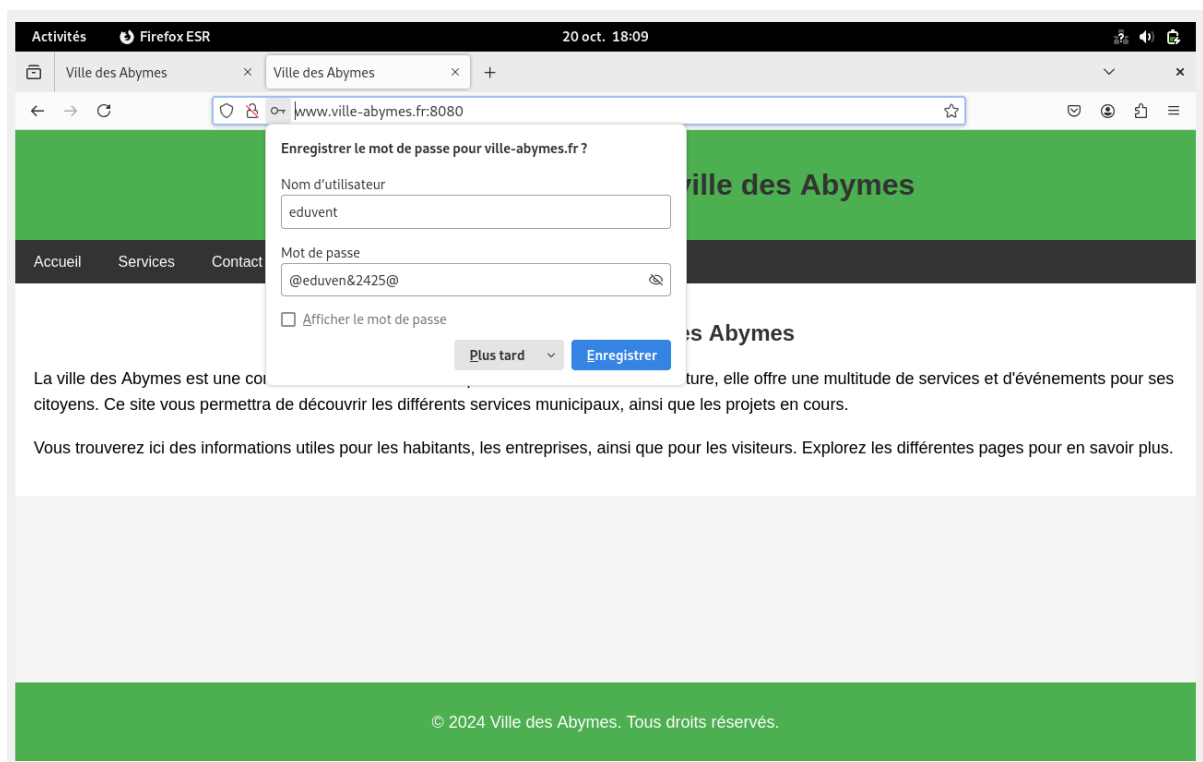
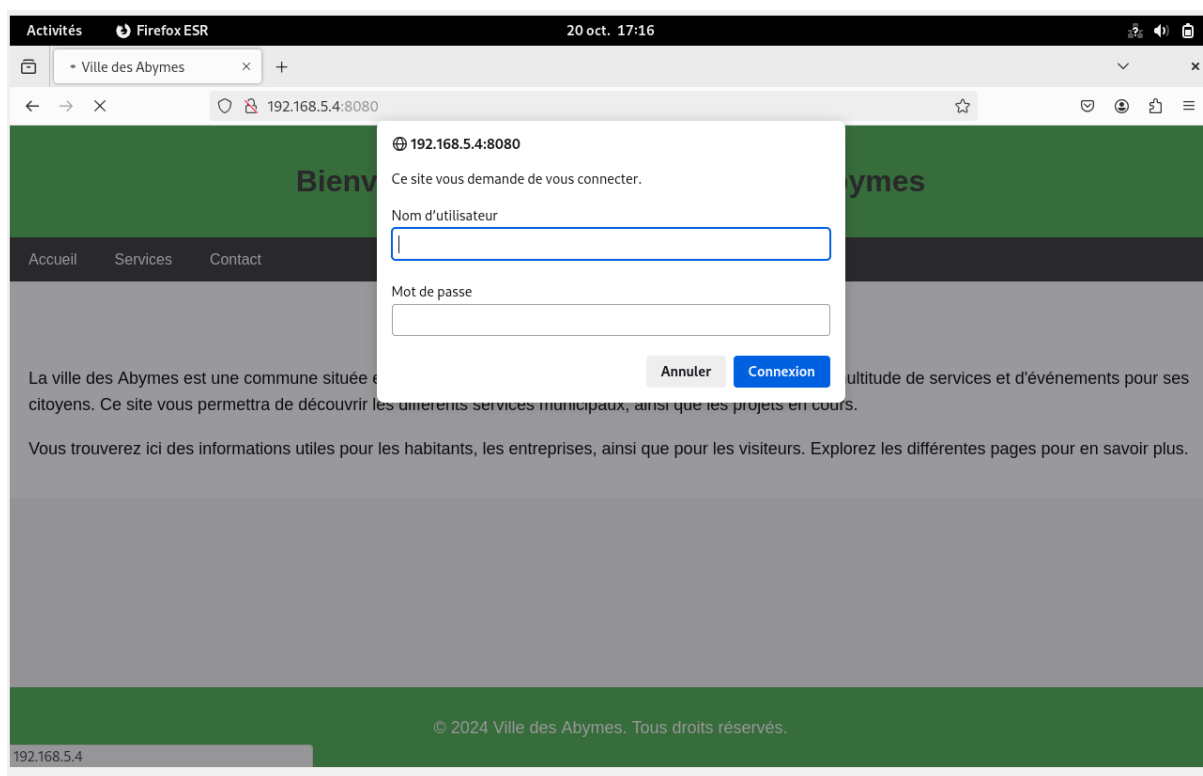
Redémarrer Apache pour prendre en compte les changements.

Avec la commande : << # **systemctl restart apache2** >>

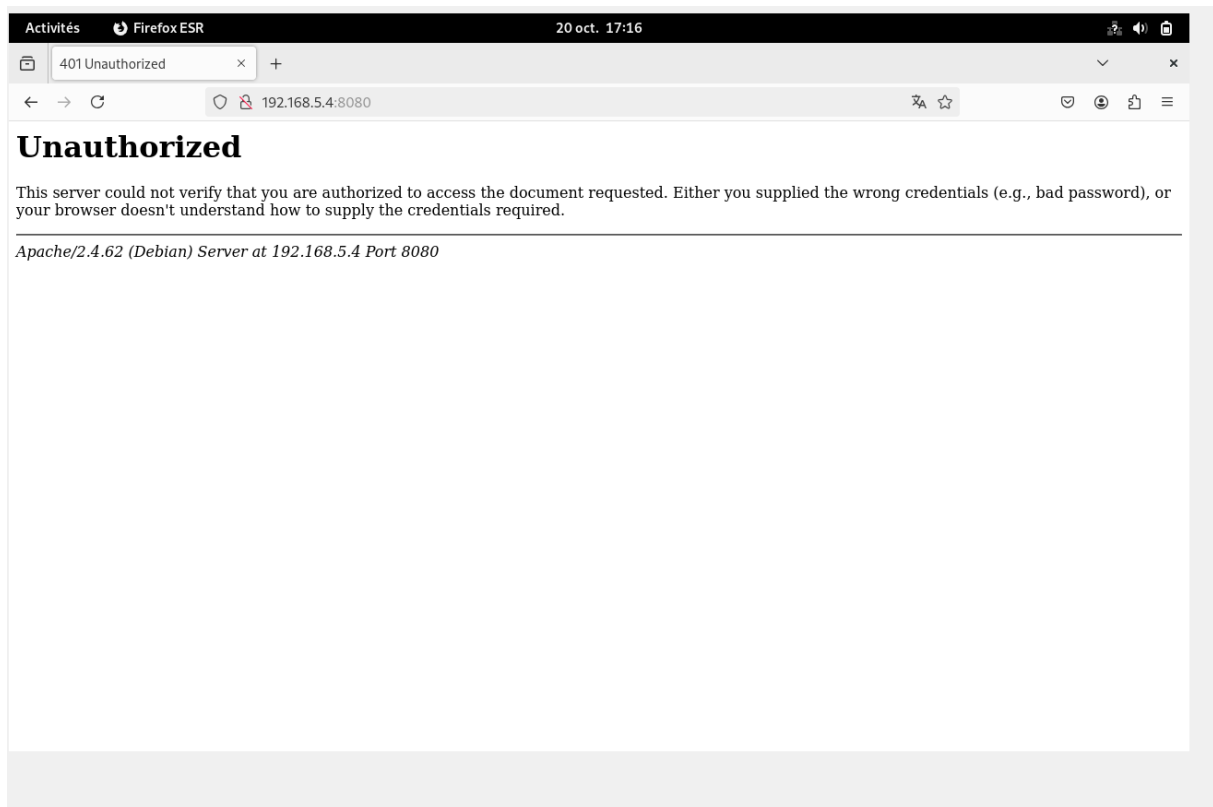
Teste de l'authentification :

Il suffit sur un poste client d'ouvrir le navigateur et de saisir une des URL permettant d'accéder au site pour moi toujours en utilisant le port 8080 et de constater qu'il faut entrer des informations de connexion pour accéder au site. Et si on ne met pas les bonnes informations l'accès est no autorisée

Teste de l'accès au site depuis le poste2 client Linux Debian :



Lorsque la connexion est réussie il est demandé si on veut enregistrer nos informations de connexion.



Au contraire si la connexion échoue on le code d'erreur 401 qui signifie que l'accès est non-autorisée.

CONCLUSION :

Ce TP sur le déploiement du service Apache 2.0 sous Linux Debian a offert une expérience pratique enrichissante dans la gestion des serveurs web sous environnement open-source. La configuration du serveur, la création des hôtes virtuels et la mise en place de l'authentification ont permis de maîtriser des compétences essentielles pour l'administration de serveurs web. Ces savoir-faire, qui sont au cœur des infrastructures modernes, resteront des acquis durables pour de futures missions professionnelles. La réussite de cette configuration, tout en garantissant la sécurité et la performance de l'hébergement, confirme l'importance de ces compétences pour tout administrateur système.

FIN DU TP :

