

Apache2

Partie A : Serveurs web virtuels

Modification DNS

Pour cette première partie nous allons devoir permettre à notre serveur web d'héberger plusieurs sites différents qui seront :

- www.m2l.org
- intranet.m2l.org
- extranet.m2l.org
- wiki.m2l.org

Mais avant de faire tout cela il va falloir que nous configurions nos DNS car créer un hôte virtuel sous apache ne provoque pas son apparition magique dans la configuration du DNS !!! comme il est dit dans la doc apache2

**Note :**

La configuration de serveurs virtuels sous Apache ne provoque pas leur apparition magique dans la configuration du DNS. Il faut que leurs noms soient définis dans le DNS, et qu'ils y soient résolus sur l'adresse IP du serveur, faute de quoi personne ne pourra visiter votre site Web. Il est possible d'ajouter des entrées dans le fichier hosts pour tests locaux, mais qui ne fonctionneront que sur la machine possédant ces entrées.

Il faut donc s'assurer que la résolution de notre DNS fonctionne correctement pour mettre en place les différents sites web. Nous allons donc modifier les fichiers de conf du DNS 1 de cette manière.

```
nano /etc/bind/db.m2l.org
```

```
$TTL      604800
@ IN SOA ns1.m2l.org. root.m2l.org (
        2023012706
        604800
        86400
        2419200
        604800)

@ IN A 10.31.80.80

; DNS Server
@ IN NS ns1.m2l.org.
ns1 IN A 10.31.80.53
@ IN NS ns2.m2l.org.
ns2 IN A 10.31.80.54
backup IN A 10.31.80.99
```

```
; Machines
www IN A 10.31.80.80
ftp IN A 10.31.80.20

; Aliases
console IN CNAME www
intranet IN CNAME www
extranet IN CNAME www
wiki IN CNAME www
```

Après avoir fait c'est modification il nous faut restart bind9 sur le dns 1 et dns 2.

```
systemctl restart bind9
```

Afin de vérifier que les modification on bien été prise en compte par les dns nous effectuons la commande :

```
dig a wiki.m2l.org @localhost
```

Nous devons obtenir une réponse comme ci-après :

```
;; ANSWER SECTION:
wiki.m2l.org.        604800 IN      CNAME  www.m2l.org.
www.m2l.org.        604800 IN      A      10.31.80.80
```

Configuration apache

Après avoir fait les modification des dns nous pouvons configurer apache2.

Nous créerons un répertoire dans le répertoire sites-available qui accueillera les fichiers de configuration pour chacun des VirtualHost

```
root@web:~# mkdir -p /home/htdocs/m2l.org/www
root@web:~# mkdir -p /home/htdocs/m2l.org/intranet
root@web:~# mkdir -p /home/htdocs/m2l.org/extranet
root@web:~# mkdir -p /home/htdocs/m2l.org/woki
root@web:~# ^Chome/htdocs/m2l.org/woki
root@web:~# mv /home/htdocs/m2l.org/woki /home/htdocs/m2l.org/wiki
```

Dans le répertoire respectif a chaque Vhost nous créons un fichier avec une extension .conf

```
nano /etc/apache2/sites-available/nameVhost.conf
```

Dans lesquels nous devons écrire se qui suit

```
<VirtualHost *:80>
    ServerName m2l.org #nom server
    ServerAlias www.m2l.org #alias de la machine
    DocumentRoot /home/htdocs/m2l.org/www #changer le chemin par le
chemin du document root du Vhost
    ErrorLog /var/log/apache2/www-error.log
    CustomLog /var/log/apache2/www-access.log combined
    <Directory /home/htdocs/m2l.org/www/>
        Require all granted
```

```
</Directory>
</VirtualHost>
```

```
<VirtualHost *:80>

    ServerName extranet.m2l.org
    DocumentRoot /home/htdocs/m2l.org/extranet

    ErrorLog /var/log/apache2/extranet-error.log
    CustomLog /var/log/apache2/extranet-access.log combined

    <Directory /home/htdocs/m2l.org/extranet/>
        Require all granted
    </Directory>
</VirtualHost>
```

```
<VirtualHost *:80>

    ServerName intranet.m2l.org
    DocumentRoot /home/htdocs/m2l.org/intranet

    ErrorLog /var/log/apache2/intranet-error.log
    CustomLog /var/log/apache2/intranet-access.log combined

    <Directory /home/htdocs/m2l.org/intranet/>
        Require all granted
        AllowOverride all
    </Directory>
</VirtualHost>
<sxh>
<VirtualHost *:80>

    ServerName wiki.m2l.org
    DocumentRoot /home/htdocs/m2l.org/wiki

    ErrorLog /var/log/apache2/wiki-error.log
    CustomLog /var/log/apache2/wiki-access.log combined

    <Directory /home/htdocs/m2l.org/wiki/>
        Require all granted
    </Directory>
</VirtualHost>
```

Après avoir créer tout les fichier de conf pour chaque Vhost, il faut les activer avec la commande :

```
a2ensite www.m2l.org #il faut renseigner le nom du Vhost a activer
après le a2ensite
```

que l'on effectue pour chaque Vhost que nous souhaitons activer puis il faut redémarrer

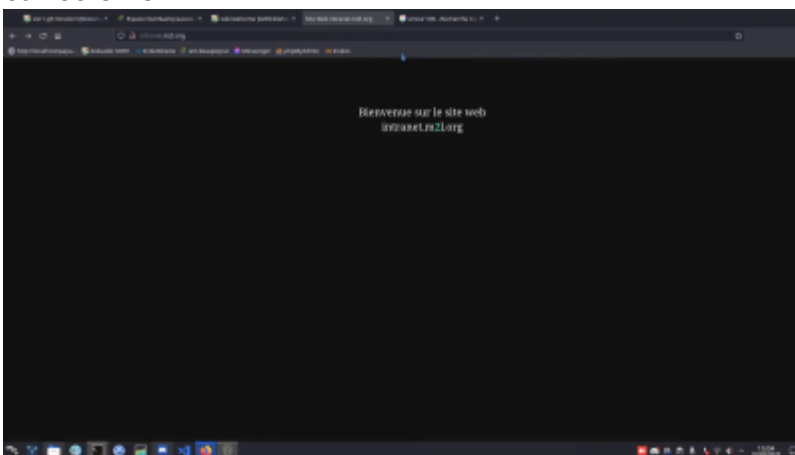
apache2 avec systemctl restart apache2.

Afin de tester si tout les Vhost fonction nous créons une petite page html avec du css pour tester. nous créons donc dans le documentRoot un fichier index.html et style.css

```
<!--index.html-->
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>Site Web www.m2l.org</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <div class="content"><br /><br /><br />Bienvenue sur le site
web<br/>www.m<em>2</em>l.org</div>
  </body>
</html>
```

```
/*style.css*/
body {background-color: #111;}
.content {
  width:100%;
  text-align: center;
  color: white;
  font-size: 25px;
}
em {
  font-weight: bold;
  font-style: normal;
  color: #44cc99;
  font-size: 25px;
}
```

Après avoir fait cela nous vérifions en allant sur un navigateur pour voir si tout fonctionne correctement.



Pour notre Vhost wiki nous installons un doku wiki dessus. Pour cela nous nous plaçons dans le répertoire où nous souhaitons l'installer.

```
cd /home/htdocs/m2l.org/wiki
```

Puis nous effectuons la commande :

```
wget https://download.dokuwiki.org/src/dokuwiki/dokuwiki-stable.tgz
```

Mais cela ne suffit pas il faut maintenant détarer l'archive avec la commande :

```
tar xzvf dokuwiki-stable.tgz
```

Comme nous souhaitons arriver dans le fichier dokuwiki nous ne le déplaçons dans le répertoire parent.

```
mv dokuwiki-2022-07-31a/ ..
```

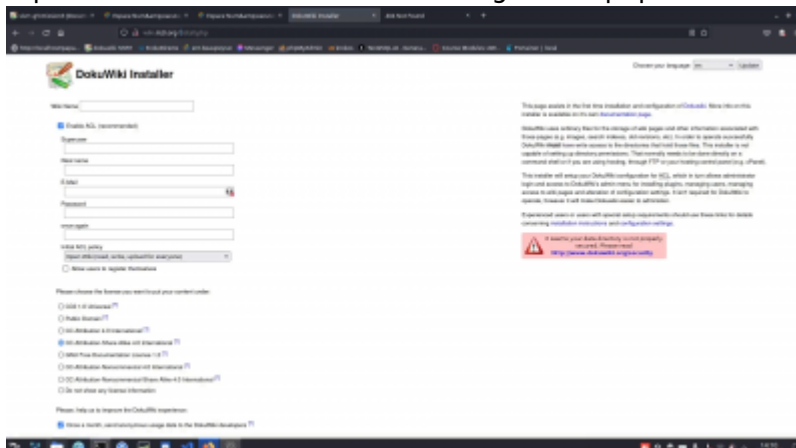
nous nous plaçons ensuite dans le répertoire parent avec `cd ..` puis nous supprimons le répertoire wiki pour le remplacer par le répertoire où se trouve le dokuwiki avec les commande suivante.

```
rm -rf wiki
mv dokuwiki-2022-07-31a/ wiki
```

Mais cela ne suffit pas il faut changer le propriétaire du répertoire pour que l'on puisse utiliser le dokuwiki correctement grâce à la commande :

```
chown -R www-data wiki #le -R permet de faire le changement de manière récursive et de changer le propriétaire de tout répertoire ou fichier se trouvant dans ce répertoire.
```

Après avoir fait cela nous nous rendons sur notre navigateur pour terminer l'installation en tapant dans la recherche `wiki.m2l.org/install.php`.



Partie B - les fichier .htaccess

Fichier .htaccess



Les fichiers .htaccess fournissent une méthode pour modifier la configuration du serveur au niveau d'un répertoire. Un fichier, contenant une ou plusieurs



directives de configuration, est placé dans un répertoire de documents particulier, et ses directives s'appliquent à ce répertoire et à tous ses sous-répertoires.

L'utilisation des fichiers .htaccess ne doit jamais être privilégiée à une configuration directe dans les fichiers d'Apache.

Pour cette partie nous allons sécuriser la connexion à l'intranet qui doit être accessible uniquement par les personnes autorisées.

Pour cela nous allons modifier le fichier de configuration d'intranet nous ajouterons ceci :

```
<Directory /home/htdocs/m2l.org/intranet/>  
    Require all granted  
    AllowOverride All  
</Directory>
```

L'instruction AllowOverride All permet au répertoire de prendre en compte les fichiers .htaccess.

Après cela on peut faire le fichier .htaccess il nous faut donc nous placer à l'endroit que l'on veut protéger. Dans notre cas on se place sur le répertoire d'intranet.

```
cd /home/htdocs/m2l.org/intranet
```

```
AuthType Basic  
AuthUserFile /var/www/accounts/.htpasswd  
AuthName "Reserved Access"  
Require valid-user
```

Après avoir fait ce fichier il faut créer les utilisateurs pouvant y accéder.

```
mkdir /var/www/accounts  
htpasswd -c /var/www/accounts/.htpasswd sio
```



il faut penser à restart apache2

```
systemctl restart apache2
```

nous pouvons également personnaliser les pages d'erreur en créant une page html pour chacune d'entre elles puis de le renseigner dans le fichier .htaccess.

```
AuthType Basic
```

```
AuthUserFile /var/www/accounts/.htpasswd
AuthName "Reserved Access"
Require valid-user
ErrorDocument 401 /401.html
ErrorDocument 403 /403.html
ErrorDocument 404 /404.html
ErrorDocument 500 /500.html
```

Partie C - Les répertoires personnels

USER_DIR



Cette directive permet à un utilisateur de publier des pages dans le répertoire public_html de son home directory. On y accède ensuite avec la requête : <http://AdresselpDuServeur/~compte/> (notez le '~').

- Configurez apache pour les comptes userdir

```
a2enmod userdir
```

- création du répertoire public_html

```
mkdir /home/sio/public_html
```

- création des page index.html et index.php

```
<!--index.html-->
<!doctype html>
<html lang="fr">
  <head>
    <meta charset="utf-8">
    <title>Site Web www.m2l.org</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <div class="content"><br /><br /><br />Bienvenue sur le site
web<br/>www.m<em>2</em>l.org</div>
  </body>
</html>
```

```
<?php
    phpinfo();
?>
```

Pour la page en php il faut réactiver php sur les compte pour que cela fonctionne.

```
nano /etc/apache2/mods-enabled/php7.4.conf
```

```
<FilesMatch ".+\.ph(ar|p|tml)$">
    SetHandler application/x-httpd-php
</FilesMatch>
<FilesMatch ".+\.phps$">
    SetHandler application/x-httpd-php-source
    # Deny access to raw php sources by default
    # To re-enable it's recommended to enable access to the files
    # only in specific virtual host or directory
    Require all denied
</FilesMatch>
# Deny access to files without filename (e.g. '.php')
<FilesMatch "^\.ph(ar|p|ps|tml)$">
    Require all denied
</FilesMatch>

# Running PHP scripts in user directories is disabled by default
#
# To re-enable PHP in user directories comment the following lines
# (from <IfModule ...> to </IfModule>.) Do NOT set it to On as it
# prevents .htaccess files from disabling it.
<IfModule mod_userdir.c>
    <Directory /home/*/public_html>
        php_admin_flag engine On #il faut passer cette valeur a on car
        elle est en off par défaut
    </Directory>
</IfModule>
```



Penser a restart apache2

From:

<https://sisr2.beaupeyrat.com/> - **Documentations SIO2 option SISR**

Permanent link:

<https://sisr2.beaupeyrat.com/doku.php?id=sisr1-g5:mission6>

Last update: **2024/03/08 14:05**



