

NFA083 – Réseau et Administration Web

Introduction à la Configuration du serveur Apache httpd

Sami Taktak

sami.taktak@cnam.fr

Centre d'Étude et De Recherche en Informatique et Communications
Conservatoire National des Arts et Métiers

Mai 2014



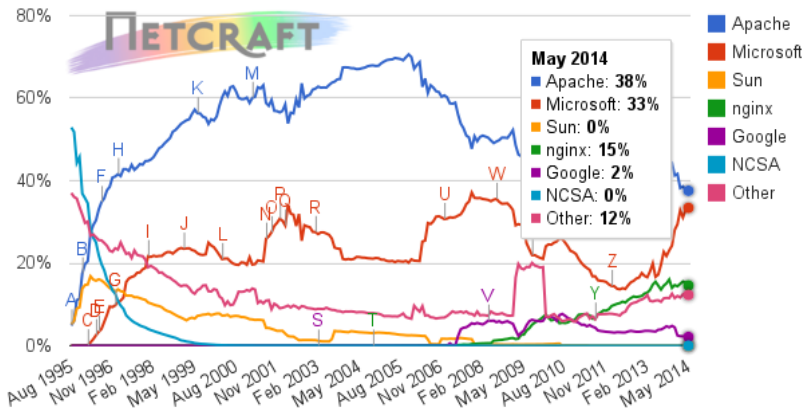
le **cnam**

Apache HTTP Server

- Serveur HTTP
- Première version publique : Avril 1995
- Maintenu par la fondation Apache
- Développé sur la base et pour remplacer NCSA httpd 1.3
- NCSA (*National Center for Supercomputing Application*)
httpd : navigateur le plus répandu à l'époque
- Devient le serveur le plus utilisé sur Internet en moins d'un an
- <http://httpd.apache.org/>

May 2014 Web Server Survey

All sites



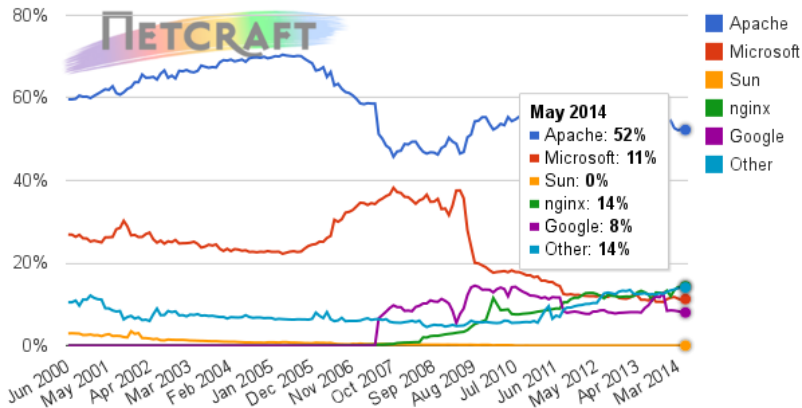
Source : <http://news.netcraft.com/archives/2014/05/07/may-2014-web-server-survey.html>



le cnam

May 2014 Web Server Survey

Active sites



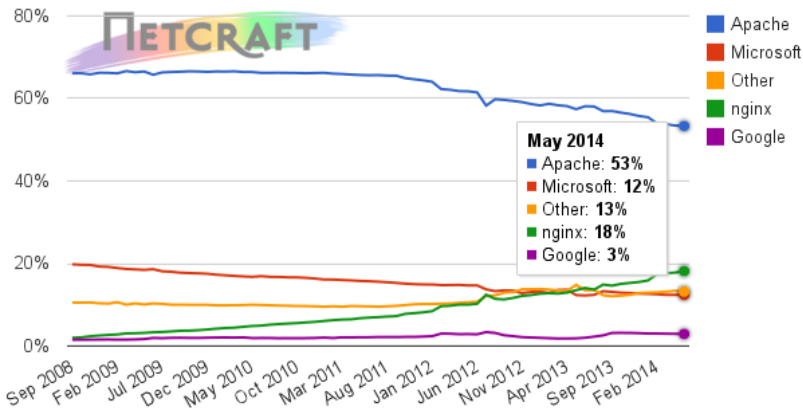
Source : <http://news.netcraft.com/archives/2014/05/07/may-2014-web-server-survey.html>



le cnam

May 2014 Web Server Survey

Top million busiest sites



Source : <http://news.netcraft.com/archives/2014/05/07/may-2014-web-server-survey.html>



le cnam

- Conception modulaire
- Support de *Common Gateway Interface* (CGI)
- Module d'interprétation de langage : Perl, PHP, Python, Ruby, ...
- Module d'extension : webdav, serveur proxy, réécriture d'URL, ...
- Fichiers de configurations hiérarchiques : chaque fichier peut être géré indépendamment des autres
- Présence de fichiers `.htaccess` permettant une configuration par répertoire

Informations Nécessaires pour la Configuration d'un Serveur Apache

De quoi a-t-on Besoin ?

- Fichiers de configuration
- Fichiers de contenu (site web)
- Gestion de droits : qui, quoi, à qui ?
- Quels services ?

Informations Nécessaires pour la Configuration d'un Serveur Apache

On doit donc fournir au serveur :

- Le nom de l'interface sur le quel il doit écouter
- Le numéro de port sur le quel il doit écouter
- Le répertoire contenant les fichiers constituant le site web
- Les options de configuration du site web
- Les droits d'accès au site web

Configuration du serveur Apache2

- située dans le répertoire `/etc/apache2`
- fichier de configuration principal : `apache2.conf`
- composé de plusieurs « sections »
- chaque section contenue dans un fichier spécifique
- fichiers spécifiques inclus dans `apache2.conf` (directive `Include`)

- Fichiers de configuration dans : `/etc/apache2/`

<code>apache2.conf</code>	<code>envvars</code>	<code>magic</code>	<code>mods-enabled/</code>	<code>sites-available/</code>
<code>conf.d/</code>	<code>httpd.conf</code>	<code>mods-available/</code>	<code>ports.conf</code>	<code>sites-enabled/</code>

- Structure des fichiers de configuration spécifique à Debian
- Généralement, un fichier de configuration unique :
`httpd.conf`
- Pour plus d'informations :
 - `man apache2`
 - fichier `/usr/share/doc/apache2.2-common/README.Debian.gz`

Structure du Fichier apache2.conf

Composé de 7 sections :

- 1 Configuration global (fichier `apache2.conf`)
- 2 Modules (fichiers dans `mods-enabled/`)
- 3 Configuration locale (fichier `httpd.conf`)
- 4 Ports et interfaces d'écoute (fichier `ports.conf`)
- 5 Format et niveau de log
- 6 Configuration spécifique (fichiers dans `conf.d/`)
- 7 Sites virtuels (fichiers dans `sites-enabled/`)



- `sites-available/` : contient les fichiers de configuration des sites web disponibles
- `sites-enabled/` : contient les fichiers de configuration des sites web activés (lien symboliques vers `sites-available/`)

```
# ls -l /etc/apache2/sites-enabled/  
lrwxrwxrwx 1 root root 26 15 avril 13:50 000-default -> ../sites-available/default
```

- a2enmod et a2dismod : commandes permettant d'activer ou de désactiver un module
 - a2enmod pour « *apache 2 enable module* »
 - a2dismod pour « *apache 2 disable module* »
- a2ensite et a2dissite : commandes permettant d'activer ou de désactiver un module
 - a2ensite pour « *apache 2 enable site* »
 - a2dissite pour « *apache 2 disable site* »

Activation et Désactivation d'un Site Virtuel

- Activation du site virtuel `monsiteweb.com`

- Activation du site virtuel `monsiteweb.com` :

```
# a2ensite monsiteweb.com
```

```
Enabling site monsiteweb.com.
```

```
To activate new configuration, you need to run:
```

```
service apache2 reload
```

- Activation du site virtuel `monsiteweb.com` :

```
# a2ensite monsiteweb.com
Enabling site monsiteweb.com.
To activate new configuration, you need to run:
    service apache2 reload

# ls -l /etc/apache2/sites-enabled/
lrwxrwxrwx 1 root root 26 15 avril 13:50 000-default ->
../sites-available/default
lrwxrwxrwx 1 root root 26 15 avril 13:50 monsiteweb.com ->
../sites-available/monsiteweb.com
```


- Activation du site virtuel `monsiteweb.com` :

```
# a2ensite monsiteweb.com
Enabling site monsiteweb.com.
To activate new configuration, you need to run:
    service apache2 reload

# ls -l /etc/apache2/sites-enabled/
lrwxrwxrwx 1 root root 26 15 avril 13:50 000-default ->
../sites-available/default
lrwxrwxrwx 1 root root 26 15 avril 13:50 monsiteweb.com ->
../sites-available/monsiteweb.com
```

- Désactivation du site virtuel `monsiteweb.com`

- Activation du site virtuel `monsiteweb.com` :

```
# a2ensite monsiteweb.com
Enabling site monsiteweb.com.
To activate new configuration, you need to run:
    service apache2 reload

# ls -l /etc/apache2/sites-enabled/
lrwxrwxrwx 1 root root 26 15 avril 13:50 000-default ->
../sites-available/default
lrwxrwxrwx 1 root root 26 15 avril 13:50 monsiteweb.com ->
../sites-available/monsiteweb.com
```

- Désactivation du site virtuel `monsiteweb.com` :

```
# a2dissite monsiteweb.com
Site monsiteweb.com disabled.
To activate new configuration, you need to run:
    service apache2 reload
```

- Activation du site virtuel `monsiteweb.com` :

```
# a2ensite monsiteweb.com
Enabling site monsiteweb.com.
To activate new configuration, you need to run:
    service apache2 reload

# ls -l /etc/apache2/sites-enabled/
lrwxrwxrwx 1 root root 26 15 avril 13:50 000-default ->
../sites-available/default
lrwxrwxrwx 1 root root 26 15 avril 13:50 monsiteweb.com ->
../sites-available/monsiteweb.com
```

- Désactivation du site virtuel `monsiteweb.com` :

```
# a2dissite monsiteweb.com
Site monsiteweb.com disabled.
To activate new configuration, you need to run:
    service apache2 reload

# ls -l /etc/apache2/sites-enabled/
lrwxrwxrwx 1 root root 26 15 avril 13:50 000-default ->
../sites-available/default
```



Structure de apache2.conf :

- Définition des variables générales
- Inclusion des fichiers de configuration des modules
- Inclusion du fichier de configuration `httpd.conf`
- Inclusion du fichier de configuration `ports.conf`
- Définition des formats de log
- Inclusion des fichiers de configuration additionnels présents dans `conf.d/`
- Inclusion des fichiers de configuration spécifiques à chaque site web



Section variables générales :

- Chemin vers les fichiers de configuration (ServerRoot), /etc/apache2 par défaut
- Fichier de verrou (LockFile)
- Fichier contenant le numéro de processus du serveur (PidFile)
- Gestion du nombre de ressources alloués au serveur (StartServers, MaxClients, ...)
- Nom de l'utilisateur (User) et du groupe (Grup) auxquels doit appartenir le serveur
- Définition du fichier gérant les droits d'accès par répertoire (AccessFileName)
- Niveau d'alerte des messages à enregistrer (LogLevel)



- définit les options globales au serveur HTTP
- donne les valeurs de configuration par défaut
- commun à tous les sites hébergés
- définit notamment :
 - le nombre de requêtes simultanées pouvant être gérées par le serveur
 - l'emplacement des fichiers de configuration

Directives de Configuration Globale

- `ServerRoot` : emplacement des fichiers de configurations et de log
ne doit pas contenir de « / » à la fin du chemin :
- `LockFile` : chemin vers le fichier verrou utiliser pour la gestion de requêtes multiples
- `PidFile` : fichier contenant le numéro de processus du serveur
- `Timeout` : temps d'attente avant expiration (en seconde)
- `KeepAlive` : connexions persistantes, plusieurs requêtes par connexion possibles
- `MaxKeepAliveRequests` : nombre maximum de requête par connexion persistante
- `KeepAliveTimeout` : temps d'inactivité avant clôture de la connexion persistante

Directives de Configuration Globale

- `User` : utilisateur auquel appartient le serveur
- `Group` : groupe auquel appartiendra le serveur
- `AccessFileName` : nom du fichier permettant d'ajouter des directives par répertoire
- `DefaultType` : type MIME par défaut lorsque le serveur n'arrive pas à déterminer le type de fichier
- `HostnameLookups` : indique si le serveur doit écrire le nom des clients ou leur adresse IP dans les logs
- `ErrorLog` : fichier par défaut pour le log des erreurs
- `LogLevel` : niveau d'alerte devant être enregistré dans le fichier `ErrorLog`

Le fichier ports.conf

ports.conf : permet de spécifier le port et le nom de domaine pour lesquels le serveur écoute

```
NameVirtualHost *:80
Listen 80
```

```
<IfModule mod_ssl.c>
    # If you add NameVirtualHost *:443 here, you will also have to change
    # the VirtualHost statement in /etc/apache2/sites-available/default-ssl
    # to <VirtualHost *:443>
    # Server Name Indication for SSL named virtual hosts is currently not
    # supported by MSIE on Windows XP.
    Listen 443
</IfModule>
```

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

Fichiers de Configuration Spécifiques à Chaque Site Web

```
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    # Possible values include: debug, info, notice, warn, error, crit,
    # alert, emerg.
    LogLevel warn
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

- Démarrer Apache 2 : `/etc/init.d/apache2 start`
- Arrêter Apache 2 : `/etc/init.d/apache2 stop`
- Redémarrer Apache 2 : `/etc/init.d/apache2 restart`

Peut aussi être géré avec `apache2ctl` :

- `apache2ctl`
`[start|stop|restart|status|fullstatus]`
- obtenir des informations sur le serveur Apache :
 - `status` : informations succinctes sur l'état du serveur
 - `fullstatus` : informations détaillées sur l'état du serveur
 - `man apache2ctl` pour plus d'informations



Hôtes Virtuels

- Virtual Host : héberger plusieurs sites web sur un même serveur web
- *IP-based* : une adresse IP par site web
- *name-based* : un nom de domaine par site web, mais une IP partagée entre les sites web
- Une directive `<VirtualHost>` par site web
- Première directive `<VirtualHost>` définit le site web par défaut
- Champ `Host` de la requête HTTP permet d'identifier l'hôte virtuel

Configuration d'Hôtes Virtuels

```
<VirtualHost *:80>  
ServerName www.domain.tld  
ServerAlias domain.tld *.domain.tld  
DocumentRoot /www/domain  
</VirtualHost>
```

```
<VirtualHost *:80>  
ServerName www.otherdomain.tld  
DocumentRoot /www/otherdomain  
</VirtualHost>
```

Configuration d'Hôtes Virtuels

- `NameVirtualHost` : design l'adresse IP et le port sur lequel le serveur va accepter des requêtes pour les hôtes virtuels, en général, `*:80` ; définit dans le fichier `/etc/apache/ports.conf`
- `<VirtualHost>` : une directive par hôte virtuel
- `ServerName` : indique le nom du serveur web virtuel
- `DocumentRoot` : indique le chemin où se trouvent les fichiers pour cet hôte
- `ServerAlias` : permet d'indiquer d'autres noms pour cet hôte

- 1 Réception d'une requête
- 2 Vérifie que l'adresse IP correspond bien à `NameVirtualHost`
- 3 Cherche une directive `<VirtualHost>` correspondant à l'adresse IP
- 4 Vérifie si `ServerName` ou `ServerAlias` correspond au nom d'hôte demandé
- 5 Si un tel hôte virtuel est trouvé, Apache utilise sa configuration
- 6 Sinon, Apache utilise la configuration du premier hôte virtuel

Conséquence :

DocumentRoot dans la section principale du fichier de configuration n'est **jamais utilisé** si l'adresse IP correspond à NameVirtualHost

⇒ utiliser le premier <VirtualHost> comme hôte par défaut

Directive <Directory>

- <Directory /un/chemin/> : permet de définir des directives s'appliquant uniquement à ce répertoire et aux fichiers et sous-répertoires qu'il contient

```
<VirtualHost *:80>
  ServerName www.domain.tld
  ServerAlias domain.tld *.domain.tld
  DocumentRoot /www/domain
  <Directory /www/domain>
    Options Indexes FollowSymlinks Multiviews
    AllowOverride None
    Require all granted
  </Directory>
</VirtualHost>
```

None	Désactive toutes les options
All	Active toutes les options SAUF Multiviews
Indexes	Permet aux utilisateurs d'avoir des index générés par le serveur si fichier index manquant (ex. index.html)
FollowSymLinks	Autorise à suivre les liens symboliques
ExecCGI	Autorise à exécuter des scripts CGI à partir de ce répertoire

<code>Includes</code>	Permet d'ajouter dynamiquement du contenu aux pages HTML (ex. date du jour)
<code>IncludesNOEXEC</code>	Permet d'ajouter dynamiquement du contenu mais empêche la commande <code>exec</code>
<code>MultiViews</code>	Autorise les vue multiples suivant un contexte
<code>SymLinksIfOwnerMatch</code>	Autorise a suivre les liens seulement si l'user ID du fichier (ou répertoire) sur lequel le lien pointe est le même que celui du lien

- définit au sein des sections :
 - `<Directory>` : porte sur des répertoires
 - `<Files>` : porte sur des fichiers
 - `<Location>` : porte sur des URLs
- *Filesystem Containers* : `<Directory>` et `<Files>` sont des directives s'appliquant au système de fichier
- *Webspace Containers* : `<Location>` change la configuration relativement à une URL
- permet de redéfinir les politiques d'accès à ces ressources :
 - `Options` : `Indexes`, `FollowSymlinks`, `Multiviews`, ...
 - `AllowOverride` : `AuthConfig`, `FileInfo`, `Options`, ...
 - `Require` : `all` (`granted|denied`), `user userid` [`userid`], `ip ip-range`, ...



Section <Directory chemin_du_répertoire> et
</Directory> :

- définit un ensemble de directives s'appliquant uniquement :
 - aux répertoires désignés
 - aux sous-répertoire qu'ils contiennent
 - **et** aux fichiers contenus dans ces répertoires.
- Peuvent apparaître dans une section :
 - configuration du serveur
 - VirtualHost

Section <Files nom_de_fichier> et </Files>

- définit un ensemble de directives s'appliquant uniquement aux fichiers désignés.
- Directives <Files> appliqués dans l'ordre après les directives
 - <Directory>
 - **et** celles des .htaccess
 - **mais avant** les directives <Location>
- Peuvent apparaître dans une section :
 - configuration du serveur
 - VirtualHost
 - Directory
 - .htaccess



```
<Directory /var/web/dir1>  
  Options +Indexes  
</Directory>
```

```
<Files private.html>  
  Require all denied  
</Files>
```

```
<Directory /var/web/dir1>  
  <Files private.html>  
    Require all denied  
  </Files>  
</Directory>
```



Section <Location URL> et </Location>

- définit un ensemble de directives s'appliquant uniquement aux URLs désignés.
- Directives <Location> appliqués dans l'ordre après les directives
 - <Directory>
 - celles des .htaccess
 - **et** <Files>
- Peuvent apparaître dans une section :
 - configuration du serveur
 - VirtualHost


```
<Location /server-status>  
    Require user admin  
</Location>
```

```
<Location /foo>  
    Require all granted  
</Location>  
<Location /foo/bar>  
    Require group admin  
</Location>
```



- `chemin_du_répertoire` : un chemin complet du système de fichier
- `nom_de_fichier` : nom de fichier uniquement (sans nom de répertoire)

Ils peuvent être donnés sous la forme :

- d'une chaîne de caractère contenant éventuellement des métacaractères (*wild-card*) dans le style des shell UNIX :
 - ? correspond à un unique caractère
 - * correspond à une séquence de caractères
 - [] définie un ensemble de caractère ([a-z] représente tous les caractères de l'alphabet en minuscule)
 - aucun métacaractère ne reconnaît « / »
- d'une expression régulière

```
<Files "cat.html">  
    # mettre ici les directives à appliquer à  
    # cat.html  
    Require all denied  
</Files>
```

Exemple

Métacaractères & Expressions régulières

```
<Files "?at.*">  
    # les directives présentes ici vont être  
    # appliquées à cat.html, bat.html, hat.php, ...  
    Require all denied  
</Files>
```

```
<Files ~ "^\.ht">  
    Require all denied  
</Files>
```

Gestion des .htaccess : directive AllowOverride

All	Gère tout ce qui est dans .htaccess
AuthConfig	Active les directives d'autorisations au niveau des répertoires
FileInfo	Active les directives contrôlant les documents (ErrorDocument, mod_mime, mod_alias, etc.)
Limit	Active les directives d'autorisation
None	Ne lit pas le fichier .htaccess
Options	Active la directive Options

Autorise/Refuse les hôtes spécifiés, les adresses IP, le nom de domaine, etc...

All granted Autorise tout le monde

All denied Refuse tout le monde

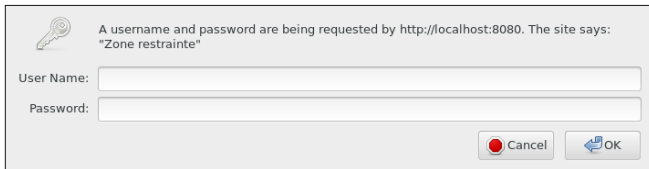
```
<VirtualHost *:80>
  ServerName www.domain.tld
  ServerAlias domain.tld *.domain.tld
  DocumentRoot /www/domain
  <Directory /var/www/htdocs>
    Options Indexes FollowSymlinks Multiviews
    AllowOverride None
    Require host domain.tld
  </Directory>
</VirtualHost>
```



Fichier .htaccess

- Permet de gérer les options du répertoire contenant le fichier .htaccess
- Permet de protéger un répertoire par mots de passe
- S'applique aussi aux sous-répertoires
- Peut être utilisé uniquement si AuthConfig a été autorisée par un AllowOverride
- Doit se trouver dans le répertoire à protéger

```
AuthUserFile /var/www/.htpasswd  
AuthName "Zone restreinte"  
AuthType Basic  
Require valid-user
```



A screenshot of a web browser authentication dialog box. The dialog has a light gray background and a thin border. In the top-left corner, there is a small icon of a key. To the right of the icon, the text reads: "A username and password are being requested by http://localhost:8080. The site says: 'Zone restreinte'". Below this text, there are two input fields. The first is labeled "User Name:" and the second is labeled "Password:". At the bottom right of the dialog, there are two buttons: "Cancel" with a red circle icon and "OK" with a blue arrow icon.

- Création du fichier de mots de passe avec ajout d'un nouvel utilisateur :
`htpasswd -c nom_fichier nom_utilisateur`
- Ajout d'un utilisateur à un fichier existant :
`htpasswd nom_fichier nom_utilisateur`
- Option `-c` permet de créer un nouveau fichier.
Supprimera le contenu d'un fichier existant !

```
# htpasswd -c /var/www/.htpasswd alice
New password:
Re-type new password:
# cat .htpasswd
alice:0sT6Iu/ERhdTk
```

Option d'Authentification

- `AuthType basic` : type d'authentification classique.
Mots de passe transmis en clair !
- `AuthName "Message d'information"` : affichera le texte comme invite dans la boîte de dialogue
- `AuthUserFile /chemin/vers/passwd` : indique où se trouvent les mots de passe
- `Require valid-user` : précise qu'il faut un compte dans le fichier de mots de passe pour accéder au répertoire

Directive Require

- Sélectionne quels utilisateurs ont accès à la ressource
- `Require user userid [userid] ...` : uniquement les utilisateurs nommé ont accès à la ressource
- `Require group group-name [group-name] ...` : uniquement les utilisateurs appartenant à un groupe nommé peuvent accéder à la ressource.
- `Require valid-user` : tout utilisateur valide peut accéder à la ressource
- Doit être accompagné des directives `AuthName` et `AuthType` ainsi que d'au moins une directive `AuthUserFile` ou `AuthGroupFile`
- Plusieurs directives `require` sont combinées avec un `OU`

Utilisation de l'Authentification dans une Directive Directory

```
<Directory /var/www/prive>  
  Options Indexes  
  AllowOverride None  
  AuthUserFile /var/www/.htpasswd  
  AuthName "Zone privée"  
  AuthType Basic  
  Require valid-user  
</Directory>
```

Directives <RequireAny> et <RequireAll>

- Permet de combiner l'usage de directives Require
- <RequireAll> : un client doit passer les restrictions sur les adresses **et** sur l'authentification
- <RequireAny> : un client doit passer les restrictions sur les adresses **ou** sur l'authentification (c'est le comportement par default)

```
<RequireAny>
```

```
Require valid-user
```

```
Require ip 192.168.1
```

```
</RequireAny>
```

Filesystem Containers vs. Webspaces Containers Lequel choisir ?

- objet appartenant au système de fichier \Rightarrow <Directory> ou <Files>
- objet n'appartenant pas au système de fichier (ie. généré dynamiquement) \Rightarrow <Location>

Important : ne pas utiliser <Location> pour restreindre l'accès à un objet du système de fichier : plusieurs URLs peuvent correspondre à un même objet du système de fichier

Configuration des Modules

- `mods-available/` : contient les fichiers de configuration des modules disponibles
- `mods-enabled/` : contient les fichiers de configuration des modules activés (lien symboliques vers `mods-available/`)

```
# ls -l /etc/apache2/mods-enabled/  
lrwxrwxrwx 1 root root 28 15 avril 13:50 alias.conf -> ../mods-available/alias.conf  
lrwxrwxrwx 1 root root 28 15 avril 13:50 alias.load -> ../mods-available/alias.load  
lrwxrwxrwx 1 root root 33 15 avril 13:50 auth_basic.load -> ../mods-available/auth_basic.load  
lrwxrwxrwx 1 root root 33 15 avril 13:50 authn_file.load -> ../mods-available/authn_file.load  
lrwxrwxrwx 1 root root 36 15 avril 13:50 authz_default.load -> ../mods-available/authz_default.load  
lrwxrwxrwx 1 root root 38 15 avril 13:50 authz_groupfile.load -> ../mods-available/authz_groupfile.load  
lrwxrwxrwx 1 root root 33 15 avril 13:50 authz_host.load -> ../mods-available/authz_host.load  
lrwxrwxrwx 1 root root 33 15 avril 13:50 authz_user.load -> ../mods-available/authz_user.load  
lrwxrwxrwx 1 root root 32 15 avril 13:50 autoindex.conf -> ../mods-available/autoindex.conf  
lrwxrwxrwx 1 root root 32 15 avril 13:50 autoindex.load -> ../mods-available/autoindex.load  
lrwxrwxrwx 1 root root 26 15 avril 13:50 cgi.load -> ../mods-available/cgi.load
```

⋮

Principaux modules activé par défaut :

- alias
- auth_basic
- autoindex
- cgi
- deflate
- dir
- env
- mime
- negotiation
- perl
- php5
- python
- reqtimeout
- status

- Ajoute un / à la fin d'une URL pointant sur un nom de répertoire

Exemple : `http://exemple.com/nom_repertoire` ⇒

`http://exemple.com/nom_repertoire/`

- Définit les fichiers servis par défaut

Exemple : si `http://exemple.com/nom_repertoire` est demandé,

`http://exemple.com/nom_repertoire/index.html` sera servit

- Permet de spécifier une URL par défaut si l'URL demandé n'existe pas

Exemple de Fichier de Configuration de Modules

Module mod_dir

2 fichiers de configurations dans /etc/apache2/mods-enabled/ :

- dir.load : permet de charger le module mod_dir

```
LoadModule dir_module /usr/lib/apache2/modules/mod_dir.so
```

- dir.conf : permet de configurer le module mod_dir

```
<IfModule mod_dir.c>  
    DirectoryIndex index.html index.cgi index.pl  
    DirectoryIndex index.php index.xhtml index.htm  
  
    FallbackResource /notfound-404.html  
</IfModule>
```



le cnam

Activation et Désactivation d'un Module

- Activation du module `mod_dav` (*Distributed Authoring and Versioning*)

Activation et Désactivation d'un Module

- Activation du module `mod_dav` (*Distributed Authoring and Versioning*) :

```
# a2enmod dav
```

```
Enabling module dav.
```

```
Run '/etc/init.d/apache2 restart' to activate new configuration!
```

Activation et Désactivation d'un Module

- Activation du module `mod_dav` (*Distributed Authoring and Versioning*) :

```
# a2enmod dav
Enabling module dav.
Run '/etc/init.d/apache2 restart' to activate new configuration!

# ls -l /etc/apache2/mods-enabled/dav*
lrwxrwxrwx 1 root root 26 25 avril 11:44 dav.load -> ../mods-available/dav.load
```

Activation et Désactivation d'un Module

- Activation du module `mod_dav` (*Distributed Authoring and Versioning*) :

```
# a2enmod dav
Enabling module dav.
Run '/etc/init.d/apache2 restart' to activate new configuration!

# ls -l /etc/apache2/mods-enabled/dav*
lrwxrwxrwx 1 root root 26 25 avril 11:44 dav.load -> ../mods-available/dav.load
```

- Désactivation du module `mod_dav`

- Activation du module `mod_dav` (*Distributed Authoring and Versioning*) :

```
# a2enmod dav
Enabling module dav.
Run '/etc/init.d/apache2 restart' to activate new configuration!

# ls -l /etc/apache2/mods-enabled/dav*
lrwxrwxrwx 1 root root 26 25 avril 11:44 dav.load -> ../mods-available/dav.load
```

- Désactivation du module `mod_dav` :

```
# a2dismod dav
Module dav disabled.
Run '/etc/init.d/apache2 restart' to activate new configuration!
```

Activation et Désactivation d'un Module

- Activation du module `mod_dav` (*Distributed Authoring and Versioning*) :

```
# a2enmod dav
Enabling module dav.
Run '/etc/init.d/apache2 restart' to activate new configuration!

# ls -l /etc/apache2/mods-enabled/dav*
lrwxrwxrwx 1 root root 26 25 avril 11:44 dav.load -> ../mods-available/dav.load
```

- Désactivation du module `mod_dav` :

```
# a2dismod dav
Module dav disabled.
Run '/etc/init.d/apache2 restart' to activate new configuration!

# ls -l /etc/apache2/mods-enabled/dav*
```


- Activation du module `mod_dav` (*Distributed Authoring and Versioning*) :

```
# a2enmod dav
Enabling module dav.
Run '/etc/init.d/apache2 restart' to activate new configuration!

# ls -l /etc/apache2/mods-enabled/dav*
lrwxrwxrwx 1 root root 26 25 avril 11:44 dav.load -> ../mods-available/dav.load
```

- Désactivation du module `mod_dav` :

```
# a2dismod dav
Module dav disabled.
Run '/etc/init.d/apache2 restart' to activate new configuration!

# ls -l /etc/apache2/mods-enabled/dav*
ls: impossible d'accéder à /etc/apache2/mods-enabled/dav*: Aucun
fichier ou dossier de ce type
```

Exemple de Configuration

/etc/apache2/apache2.conf (1/4)

```
PidFile ${APACHE_PID_FILE}
```

```
Timeout 300
```

```
KeepAlive On
```

```
MaxKeepAliveRequests 100
```

```
KeepAliveTimeout 5
```

```
User ${APACHE_RUN_USER}
```

```
Group ${APACHE_RUN_GROUP}
```

```
HostnameLookups Off
```

```
ErrorLog ${APACHE_LOG_DIR}/error.log
```

```
LogLevel warn
```



le cnam

Exemple de Configuration (Suite)

/etc/apache2/apache2.conf (2/4)

```
IncludeOptional mods-enabled/*.load  
IncludeOptional mods-enabled/*.conf
```

```
Include ports.conf
```

```
<Directory />  
    Options FollowSymLinks  
    AllowOverride None  
    Require all denied  
</Directory>
```

```
<Directory /usr/share>  
    AllowOverride None  
    Require all granted  
</Directory>
```



le cnam

Exemple de Configuration (Suite)

/etc/apache2/apache2.conf (3/4)

```
<Directory /var/www/>  
    Options Indexes FollowSymLinks  
    AllowOverride None  
    Require all granted  
</Directory>
```

```
AccessFileName .htaccess
```

```
<FilesMatch "^\.ht">  
    Require all denied  
</FilesMatch>
```



le cnam

Exemple de Configuration (Suite)

/etc/apache2/apache2.conf (4/4)

```
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O  
    \"%{Referer}i\" \"%{User-Agent}i\"" vhost_combined  
LogFormat "%h %l %u %t \"%r\" %>s %O  
    \"%{Referer}i\" \"%{User-Agent}i\"" combined  
LogFormat "%h %l %u %t \"%r\" %>s %O" common  
LogFormat "%{Referer}i -> %U" referer  
LogFormat "%{User-agent}i" agent  
  
IncludeOptional conf-enabled/*.conf  
IncludeOptional sites-enabled/*.conf
```



le cnam

Fichier mods-enabled/*.load et mods-enabled/*.conf

Exemple du module Alias

- fichier mods-enabled/alias.load

```
LoadModule alias_module /usr/lib/apache2/modules/mod_alias.so
```
- fichier mods-enabled/alias.conf

```
<IfModule alias_module>
Alias /icons/ "/usr/share/apache2/icons/"
<Directory "/usr/share/apache2/icons">
    Options Indexes MultiViews
    AllowOverride None
    Require all granted
</Directory>
</IfModule>
```

conf.d/charset :
définit l'encodage de caractères par défaut

```
AddDefaultCharset UTF-8
```

```
<Directory />  
    AllowOverride None  
    Require all denied  
</Directory>  
  
#ServerTokens Minimal  
ServerTokens OS  
#ServerTokens Full  
  
# Set to one of: On | Off | EMail  
ServerSignature On
```


Fichier conf-enabled/security.conf (Suite)

```
# Allow TRACE method
#
# Set to "extended" to also reflect the request
# body (only for testing and diagnostic purposes).
#
# Set to one of:  On | Off | extended
#
TraceEnable Off
#TraceEnable On
```

Fichier de Configuration d'Hôte Virtuel

```
<VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www/html
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/html>
        Options None
        AllowOverride None
        Require all granted
    </Directory>
```



```
<Directory /var/www/html/privé>  
    Options Indexes  
    AllowOverride None  
    AuthUserFile /var/www/passwd  
    AuthName "Zone privée"  
    AuthType Basic  
    Require valid-user  
</Directory>  
<Directory /var/www/html/privé2>  
    Options Indexes  
    AllowOverride AuthConfig  
</Directory>
```



Fichier de Configuration d'Hôte Virtuel (Suite)

```
ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/  
<Directory "/usr/lib/cgi-bin">  
    AllowOverride None  
    Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch  
    Require all granted  
</Directory>  
  
ErrorLog error.log  
# Possible values include: debug, info, notice, warn, error, crit,  
# alert, emerg.  
LogLevel warn  
CustomLog access.log combined  
</VirtualHost>
```