

4. Serveur Web

**Licence professionnelle
Université de Caen**

Jean Fromentin

<mailto:jfroment@info.unicaen.fr>

<http://www.info.unicaen.fr/~jfroment>

Serveur Web

- Le terme serveur Web désigne :

Serveur Web

- Le terme serveur Web désigne :
 - Un ordinateur sur lequel fonctionne un logiciel de serveur http ;

Serveur Web

- Le terme serveur Web désigne :
 - Un ordinateur sur lequel fonctionne un logiciel de serveur http ;
 - Un logiciel de serveur http ;

Serveur Web

- Le terme serveur Web désigne :
 - Un ordinateur sur lequel fonctionne un logiciel de serveur http ;
 - Un logiciel de serveur http ;
 - Un ensemble de serveurs permettant le fonctionnement d'applications Web.

Serveur Web

- Le terme serveur Web désigne :
 - Un ordinateur sur lequel fonctionne un logiciel de serveur http ;
 - Un logiciel de serveur http ;
 - Un ensemble de serveurs permettant le fonctionnement d'applications Web.

- En règle générale, un serveur Web ne se limite pas à un serveur http. Il est utilisé avec :

Serveur Web

- Le terme serveur Web désigne :
 - Un ordinateur sur lequel fonctionne un logiciel de serveur http ;
 - Un logiciel de serveur http ;
 - Un ensemble de serveurs permettant le fonctionnement d'applications Web.

- En règle générale, un serveur Web ne se limite pas à un serveur http. Il est utilisé avec :
 - Un serveur php,

Serveur Web

- Le terme serveur Web désigne :
 - Un ordinateur sur lequel fonctionne un logiciel de serveur http ;
 - Un logiciel de serveur http ;
 - Un ensemble de serveurs permettant le fonctionnement d'applications Web.

- En règle générale, un serveur Web ne se limite pas à un serveur http. Il est utilisé avec :
 - Un serveur php,
 - Un serveur de base de donnée.

Serveurs HTTP

- Un serveur HTTP, ou daemon HTTP, ou HTTPd, est un logiciel servant des requêtes respectant le protocole HTTP.

Serveurs HTTP

- Un serveur HTTP, ou daemon HTTP, ou HTTPd, est un logiciel servant des requêtes respectant le protocole HTTP.
- Les logiciels de serveur HTTP les plus utilisés sont :

Serveurs HTTP

- Un serveur HTTP, ou daemon HTTP, ou HTTPd, est un logiciel servant des requêtes respectant le protocole HTTP.
- Les logiciels de serveur HTTP les plus utilisés sont :
 - Apache HTTP server, de la Apache Software Foundation ;

Serveurs HTTP

- Un serveur HTTP, ou daemon HTTP, ou HTTPd, est un logiciel servant des requêtes respectant le protocole HTTP.
- Les logiciels de serveur HTTP les plus utilisés sont :
 - Apache HTTP server, de la Apache Software Foundation ;
 - Internet Information Services (IIS) de Microsoft ;

Serveurs HTTP

- Un serveur HTTP, ou daemon HTTP, ou HTTPd, est un logiciel servant des requêtes respectant le protocole HTTP.
- Les logiciels de serveur HTTP les plus utilisés sont :
 - Apache HTTP server, de la Apache Software Foundation ;
 - Internet Information Services (IIS) de Microsoft ;
 - Sun Java System Web Server de Sun Microsystems ;

Serveurs HTTP

- Un serveur HTTP, ou daemon HTTP, ou HTTPd, est un logiciel servant des requêtes respectant le protocole HTTP.
- Les logiciels de serveur HTTP les plus utilisés sont :
 - Apache HTTP server, de la Apache Software Foundation ;
 - Internet Information Services (IIS) de Microsoft ;
 - Sun Java System Web Server de Sun Microsystems ;
 - QQ.com de Tencent ;

Serveurs HTTP

- Un serveur HTTP, ou daemon HTTP, ou HTTPd, est un logiciel servant des requêtes respectant le protocole HTTP.
- Les logiciels de serveur HTTP les plus utilisés sont :
 - Apache HTTP server, de la Apache Software Foundation ;
 - Internet Information Services (IIS) de Microsoft ;
 - Sun Java System Web Server de Sun Microsystems ;
 - QQ.com de Tencent ;
 - GWS de Google ;

Serveurs HTTP

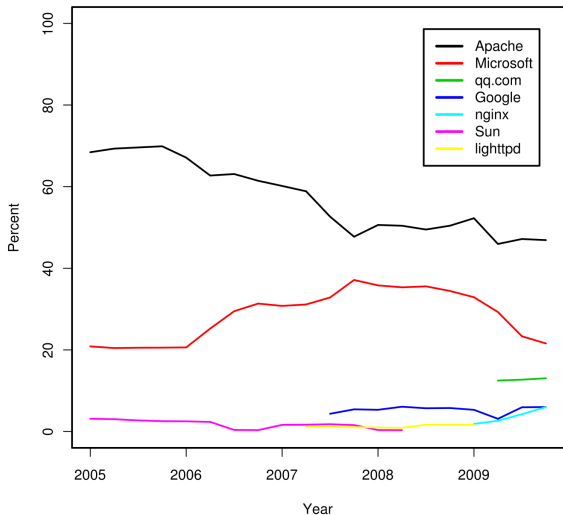
- Un serveur HTTP, ou daemon HTTP, ou HTTPd, est un logiciel servant des requêtes respectant le protocole HTTP.
- Les logiciels de serveur HTTP les plus utilisés sont :
 - Apache HTTP server, de la Apache Software Foundation ;
 - Internet Information Services (IIS) de Microsoft ;
 - Sun Java System Web Server de Sun Microsystems ;
 - QQ.com de Tencent ;
 - GWS de Google ;
 - lighttpd de Jan Kneschke ;

Serveurs HTTP

- Un serveur HTTP, ou daemon HTTP, ou HTTPd, est un logiciel servant des requêtes respectant le protocole HTTP.
- Les logiciels de serveur HTTP les plus utilisés sont :
 - Apache HTTP server, de la Apache Software Foundation ;
 - Internet Information Services (IIS) de Microsoft ;
 - Sun Java System Web Server de Sun Microsystems ;
 - QQ.com de Tencent ;
 - GWS de Google ;
 - lighttpd de Jan Kneschke ;
 - nginx d'Igor Sysoev

Serveurs HTTP

Usage share of web servers



Source: Netcraft

Serveur Apache

- Apache HTTP Server, ou Apache, est le serveur HTTP le plus populaire du Web.

Serveur Apache

- Apache HTTP Server, ou Apache, est le serveur HTTP le plus populaire du Web.



Serveur Apache

- Apache HTTP Server, ou Apache, est le serveur HTTP le plus populaire du Web.



- Quelques caractéristiques :

Serveur Apache

- Apache HTTP Server, ou Apache, est le serveur HTTP le plus populaire du Web.



- Quelques caractéristiques :
 - Il est libre, mais pas sous licence GNU GPL ;

Serveur Apache

- Apache HTTP Server, ou Apache, est le serveur HTTP le plus populaire du Web.



- Quelques caractéristiques :
 - Il est libre, mais pas sous licence GNU GPL ;
 - Il n'est pas le serveur le plus rapide ;

Serveur Apache

- Apache HTTP Server, ou Apache, est le serveur HTTP le plus populaire du Web.



- Quelques caractéristiques :
 - Il est libre, mais pas sous licence GNU GPL ;
 - Il n'est pas le serveur le plus rapide ;
 - Il est multi-plateforme ;

Serveur Apache

- Apache HTTP Server, ou Apache, est le serveur HTTP le plus populaire du Web.



- Quelques caractéristiques :
 - Il est libre, mais pas sous licence GNU GPL ;
 - Il n'est pas le serveur le plus rapide ;
 - Il est multi-plateforme ;
 - Il est écrit en C.

Configuration de Apache

- Les fichiers de configurations de Apache se trouvent dans le répertoire
`/etc/apache2`

Configuration de Apache

- Les fichiers de configurations de Apache se trouvent dans le répertoire

`/etc/apache2`

- Le fichier de configuration principal est :

`apache2.conf` (ou `httpd.conf`).

Configuration de Apache

- Les fichiers de configurations de Apache se trouvent dans le répertoire

`/etc/apache2`

- Le fichier de configuration principal est :

`apache2.conf` (ou `httpd.conf`).

- Le répertoire `conf.d` est utilisé pour ajouter des directives de configurations à l'aide de l'instruction

`Include /etc/apache2/conf.d`

écrite dans le fichier `apache2.conf`.

Configuration de Apache : les sites

- On peut héberger plusieurs sites Web, sur un même serveur Apache

Configuration de Apache : les sites

- On peut héberger plusieurs sites Web, sur un même serveur Apache
↳ c'est le cas pour votre page à l'université.

Configuration de Apache : les sites

- On peut héberger plusieurs sites Web, sur un même serveur Apache
↳ c'est le cas pour votre page à l'université.
- Pour activer les sites virtuels, on ajoute la commande
`NameVirtualHost *`

Configuration de Apache : les sites

- On peut héberger plusieurs sites Web, sur un même serveur Apache
↳ c'est le cas pour votre page à l'université.
- Pour activer les sites virtuels, on ajoute la commande
`NameVirtualHost *`
- Supposons que l'on veuille héberger les sites de John Doe et Jane Roe

Configuration de Apache : les sites

- On peut héberger plusieurs sites Web, sur un même serveur Apache
↳ c'est le cas pour votre page à l'université.
- Pour activer les sites virtuels, on ajoute la commande
`NameVirtualHost *`
- Supposons que l'on veuille héberger les sites de John Doe et Jane Roe
- On commence par créer les répertoires contenant les sites
`/var/www/jdoe.perso.info.unicaen.fr`
`/var/www/jroe.perso.info.unicaen.fr`

Configuration de Apache : les sites

- On crée les fichiers `jdoe.perso.info.unicaen.fr` et `jroe.perso.info.unicaen.fr` dans le répertoire

`sites-available`

Configuration de Apache : les sites

- On crée les fichiers `jdoe.perso.info.unicaen.fr` et `jroe.perso.info.unicaen.fr` dans le répertoire

`sites-available`

↪ ceci permet de configurer séparément Apache pour les deux sites.

Configuration de Apache : les sites

- On crée les fichiers `jdoe.perso.info.unicaen.fr` et `jroe.perso.info.unicaen.fr` dans le répertoire

`sites-available`

↪ ceci permet de configurer séparément Apache pour les deux sites.

- Pour activer les sites, il faut créer un lien symbolique dans le répertoire `sites-enabled` vers les fichiers de `sites-available`.

Configuration de Apache : les sites

- On crée les fichiers `jdoe.perso.info.unicaen.fr` et `jroe.perso.info.unicaen.fr` dans le répertoire

`sites-available`

↪ ceci permet de configurer séparément Apache pour les deux sites.

- Pour activer les sites, il faut créer un lien symbolique dans le répertoire `sites-enabled` vers les fichiers de `sites-available`.

↪ On utilise les commandes

```
a2ensite jdoe.perso.info.unicaen.fr
a2ensite jroe.perso.info.unicaen.fr
```

Configuration de Apache : les sites

- On crée les fichiers `jdoe.perso.info.unicaen.fr` et `jroe.perso.info.unicaen.fr` dans le répertoire

`sites-available`

↪ ceci permet de configurer séparément Apache pour les deux sites.

- Pour activer les sites, il faut créer un lien symbolique dans le répertoire `sites-enabled` vers les fichiers de `sites-available`.

↪ On utilise les commandes

```
a2ensite jdoe.perso.info.unicaen.fr
a2ensite jroe.perso.info.unicaen.fr
```

- Pour désactiver un site, on utilise la commande `a2dissite site`

Configuration de Apache : les modules

- Apache bénéficie d'une architecture modulaire, et chaque module additionnel peut intervenir au cours de la phase de traitement des requêtes transmises au serveur.

Configuration de Apache : les modules

- Apache bénéficie d'une architecture modulaire, et chaque module additionnel peut intervenir au cours de la phase de traitement des requêtes transmises au serveur.

↳ Exemple : php, mysql

Configuration de Apache : les modules

- Apache bénéficie d'une architecture modulaire, et chaque module additionnel peut intervenir au cours de la phase de traitement des requêtes transmises au serveur.
 - ↳ Exemple : php, mysql
- Le repertoire `mods-available` contient une serie de fichiers `.load` et `.conf`

Configuration de Apache : les modules

- Apache bénéficie d'une architecture modulaire, et chaque module additionnel peut intervenir au cours de la phase de traitement des requêtes transmises au serveur.
 - ↳ Exemple : `php`, `mysql`
- Le repertoire `mods-available` contient une serie de fichiers `.load` et `.conf`
- Le fichier `module.load` contient les instructions pour charger le module `module`

Configuration de Apache : les modules

- Apache bénéficie d'une architecture modulaire, et chaque module additionnel peut intervenir au cours de la phase de traitement des requêtes transmises au serveur.

↳ Exemple : `php`, `mysql`

- Le repertoire `mods-available` contient une serie de fichiers `.load` et `.conf`
- Le fichier `module.load` contient les instructions pour charger le module `module`
- Le fichier `module.conf` permet de configurer le module `module`.

Configuration de Apache : les modules

- Apache bénéficie d'une architecture modulaire, et chaque module additionnel peut intervenir au cours de la phase de traitement des requêtes transmises au serveur.

↳ Exemple : `php`, `mysql`

- Le repertoire `mods-available` contient une serie de fichiers `.load` et `.conf`
- Le fichier `module.load` contient les instructions pour charger le module `module`
- Le fichier `module.conf` permet de configurer le module `module`.

- L'activation d'un module est similaire à celle d'un site :

Configuration de Apache : les modules

- Apache bénéficie d'une architecture modulaire, et chaque module additionnel peut intervenir au cours de la phase de traitement des requêtes transmises au serveur.

↳ Exemple : `php`, `mysql`

- Le repertoire `mods-available` contient une serie de fichiers `.load` et `.conf`
- Le fichier `module.load` contient les instructions pour charger le module `module`
- Le fichier `module.conf` permet de configurer le module `module`.

- L'activation d'un module est similaire à celle d'un site :
 - repertoire `mods-enable`

Configuration de Apache : les modules

- Apache bénéficie d'une architecture modulaire, et chaque module additionnel peut intervenir au cours de la phase de traitement des requêtes transmises au serveur.

↳ Exemple : `php`, `mysql`

- Le repertoire `mods-available` contient une serie de fichiers `.load` et `.conf`
- Le fichier `module.load` contient les instructions pour charger le module `module`
- Le fichier `module.conf` permet de configurer le module `module`.
- L'activation d'un module est similaire à celle d'un site :
 - repertoire `mods-enable`
 - commande `a2enmod`

Configuration de Apache : les modules

- Apache bénéficie d'une architecture modulaire, et chaque module additionnel peut intervenir au cours de la phase de traitement des requêtes transmises au serveur.

↳ Exemple : `php`, `mysql`

- Le repertoire `mods-available` contient une serie de fichiers `.load` et `.conf`
- Le fichier `module.load` contient les instructions pour charger le module `module`
- Le fichier `module.conf` permet de configurer le module `module`.

- L'activation d'un module est similaire à celle d'un site :
 - repertoire `mods-enable`
 - commande `a2enmod`
 - commande `a2dismod`

Configuration de Apache : fichier `.htaccess`

- Le fichier `.htaccess` se trouvant dans le répertoire `toto` permet de configurer le comportement d'Apache lors de requêtes portant sur des fichiers de `toto`

Configuration de Apache : fichier `.htaccess`

- Le fichier `.htaccess` se trouvant dans le répertoire `toto` permet de configurer le comportement d'Apache lors de requêtes portant sur des fichiers de `toto`
- Quelle est la différence avec le fichier `apache2.conf` ?

Configuration de Apache : fichier `.htaccess`

- Le fichier `.htaccess` se trouvant dans le répertoire `toto` permet de configurer le comportement d'Apache lors de requêtes portant sur des fichiers de `toto`
- Quelle est la différence avec le fichier `apache2.conf` ?
 - ↳ les droits : cela permet à John Doe de modifier le comportement d'apache sur son site `jdoe.perso.info.unicaen.fr` sans avoir les droits d'administrer le serveur Apache.

Configuration de Apache : fichier `.htaccess`

- Le fichier `.htaccess` se trouvant dans le répertoire `toto` permet de configurer le comportement d'Apache lors de requêtes portant sur des fichiers de `toto`
- Quelle est la différence avec le fichier `apache2.conf` ?
 - ↳ les droits : cela permet à John Doe de modifier le comportement d'apache sur son site `jdoe.perso.info.unicaen.fr` sans avoir les droits d'administrer le serveur Apache.
- Est-ce que le fichier `.htaccess` permet de configurer les sous-repertoires de mon site ?

Configuration de Apache : fichier `.htaccess`

- Le fichier `.htaccess` se trouvant dans le répertoire `toto` permet de configurer le comportement d'Apache lors de requêtes portant sur des fichiers de `toto`
- Quelle est la différence avec le fichier `apache2.conf` ?
 - ↳ les droits : cela permet à John Doe de modifier le comportement d'apache sur son site `jdoe.perso.info.unicaen.fr` sans avoir les droits d'administrer le serveur Apache.
- Est-ce que le fichier `.htaccess` permet de configurer les sous-repertoires de mon site ?
 - ↳ oui

Configuration de Apache : fichier `.htaccess`

- Super, je vais mettre du `.htaccess` dans tous les répertoires de mon site!!!

Configuration de Apache : fichier `.htaccess`

- Super, je vais mettre du `.htaccess` dans tous les répertoires de mon site!!!
 - ↳ c'est déconseillé car ça réduit les performances du serveur.

Configuration de Apache : fichier .htaccess

- Super, je vais mettre du .htaccess dans tous les repertoires de mon site!!!

↪ c'est deconseillé car ça réduit les performances du serveur.

- Tout ce qu'on peut faire avec un .htaccess placé dans le repertoire toto, on peut le faire avec le fichier apache2.conf à l'aide de

```
<Directory toto>
```

```
Contenu du fichier .htaccess de toto
```

```
</Directory>
```

Configuration de Apache : fichier `.htaccess`

- Pourquoi les performances du serveurs sont réduites?

Configuration de Apache : fichier `.htaccess`

- Pourquoi les performances du serveurs sont réduites?
 - ↳ Il faut d'abord savoir qu'Apache commence à appliquer les directives des fichier `.htaccess` des repertoires parents avant celui du repertoire courant.

Configuration de Apache : fichier `.htaccess`

- Pourquoi les performances du serveurs sont réduites?
 - ↳ Il faut d'abord savoir qu'Apache commence à appliquer les directives des fichier `.htaccess` des repertoires parents avant celui du repertoire courant.
- Suppons qu'il y ait une requête dans le repertoire `/www/htdocs/example`.

Configuration de Apache : fichier `.htaccess`

- Pourquoi les performances du serveurs sont réduites?
 - ↪ Il faut d'abord savoir qu'Apache commence à appliquer les directives des fichier `.htaccess` des repertoires parents avant celui du repertoire courant.
- Supposons qu'il y ait une requête dans le repertoire `/www/htdocs/example`.
 - ↪ Apache va alors chercher un fichier `.htaccess` dans les repertoires

Configuration de Apache : fichier `.htaccess`

- Pourquoi les performances du serveurs sont réduites?
 - ↪ Il faut d'abord savoir qu'Apache commence à appliquer les directives des fichier `.htaccess` des repertoires parents avant celui du repertoire courant.
- Supposons qu'il y ait une requête dans le repertoire `/www/htdocs/example`.
 - ↪ Apache va alors chercher un fichier `.htaccess` dans les repertoires
 - `/`
 - `/www/`
 - `/www/htdocs/`
 - `/www/htdocs/example/`

Configuration de Apache : fichier `.htaccess`

- Qu'est-ce que l'on peut faire avec un fichier `.htaccess` ?

Configuration de Apache : fichier `.htaccess`

- Qu'est-ce que l'on peut faire avec un fichier `.htaccess`?
 - Gérer l'accès à certains fichiers.

Configuration de Apache : fichier `.htaccess`

- Qu'est-ce que l'on peut faire avec un fichier `.htaccess`?
 - Gérer l'accès à certains fichiers.
 - Ajouter un type Mime.

Configuration de Apache : fichier `.htaccess`

- Qu'est-ce que l'on peut faire avec un fichier `.htaccess`?
 - Gérer l'accès à certains fichiers.
 - Ajouter un type Mime.
 - Protéger l'accès à un répertoire par un mot de passe.

Configuration de Apache : fichier `.htaccess`

- Qu'est-ce que l'on peut faire avec un fichier `.htaccess`?
 - Gérer l'accès à certains fichiers.
 - Ajouter un type Mime.
 - Protéger l'accès à un répertoire par un mot de passe.
 - Protéger l'accès à un fichier par un mot de passe.

Configuration de Apache : fichier `.htaccess`

- Qu'est-ce que l'on peut faire avec un fichier `.htaccess`?
 - Gérer l'accès à certains fichiers.
 - Ajouter un type Mime.
 - Protéger l'accès à un répertoire par un mot de passe.
 - Protéger l'accès à un fichier par un mot de passe.
 - Définir des pages d'erreurs personnalisées.

Fichier `.htaccess` : Aperçu

- Un fichier `.htaccess` est composé de deux sections.

Fichier `.htaccess` : Aperçu

- Un fichier `.htaccess` est composé de deux sections.
- La première définit les chemins d'accès aux fichiers définissant les groupes d'utilisateurs et leurs mots de passe.

Fichier `.htaccess` : Aperçu

- Un fichier `.htaccess` est composé de deux sections.
- La première définit les chemins d'accès aux fichiers définissant les groupes d'utilisateurs et leurs mots de passe.
- La seconde définit les conditions d'accès à la ressources

Fichier `.htaccess` : Aperçu

- Un fichier `.htaccess` est composé de deux sections.
- La première définit les chemins d'accès aux fichiers définissant les groupes d'utilisateurs et leurs mots de passe.
- La seconde définit les conditions d'accès à la ressources
- Dans les deux sections on donne la valeur `val` à l'attribut `att` à l'aide de la ligne

`att val`

Fichier `.htaccess` : Première section

- On y définit les attributs :

Fichier `.htaccess` : Première section

- On y définit les attributs :
 - `AuthType`, qui définit le type d'authentification (`Basic`)

Fichier `.htaccess` : Première section

- On y définit les attributs :
 - `AuthType`, qui définit le type d'authentification (`Basic`)
 - `AuthName`, qui sera affiché lors de la demande d'authentification,

Fichier `.htaccess` : Première section

- On y définit les attributs :
 - `AuthType`, qui définit le type d'authentification (`Basic`)
 - `AuthName`, qui sera affiché lors de la demande d'authentification,
 - `AuthUserFile`, qui spécifie le fichier contenant les mots de passe,

Fichier `.htaccess` : Première section

- On y définit les attributs :
 - `AuthType`, qui définit le type d'authentification (`Basic`)
 - `AuthName`, qui sera affiché lors de la demande d'authentification,
 - `AuthUserFile`, qui spécifie le fichier contenant les mots de passe,
 - `AuthGroupFile` (optionnel), qui spécifie le fichier définissant les groupes,

Fichier `.htaccess` : Première section

- On y définit les attributs :
 - `AuthType`, qui définit le type d'authentification (`Basic`)
 - `AuthName`, qui sera affiché lors de la demande d'authentification,
 - `AuthUserFile`, qui spécifie le fichier contenant les mots de passe,
 - `AuthGroupFile` (optionnel), qui spécifie le fichier définissant les groupes,
- Un fichier de groupes contient des lignes de la forme

```
nom-du-groupe : liste-utilisateurs
```

Fichier `.htaccess` : Deuxième section

- La deuxième section se situe entre les balises `<Limit GET>` et `</Limit>` ou `<Files nom-fichier>` et `</Files>`

Fichier .htaccess : Deuxième section

- La deuxième section se situe entre les balises `<Limit GET>` et `</Limit>` ou `<Files nom-fichier>` et `</Files>`
- On peut remplacer GET par POST ou GET POST.

Fichier .htaccess : Deuxième section

- La deuxième section se situe entre les balises `<Limit GET>` et `</Limit>` ou `<Files nom-fichier>` et `</Files>`
- On peut remplacer GET par POST ou GET POST.
- On y définit l'attribut `Require` qui devra être satisfait pour que la ressource soit délivrée.

Fichier .htaccess : Deuxième section

- La deuxième section se situe entre les balises `<Limit GET>` et `</Limit>` ou `<Files nom-fichier>` et `</Files>`
- On peut remplacer GET par POST ou GET POST.
- On y définit l'attribut `Require` qui devra être satisfait pour que la ressource soit délivrée.
- Sa valeur peut être

Fichier .htaccess : Deuxième section

- La deuxième section se situe entre les balises `<Limit GET>` et `</Limit>` ou `<Files nom-fichier>` et `</Files>`
- On peut remplacer GET par POST ou GET POST.
- On y définit l'attribut `Require` qui devra être satisfait pour que la ressource soit délivrée.
- Sa valeur peut être
 - `user nom-utilisateur,`

Fichier .htaccess : Deuxième section

- La deuxième section se situe entre les balises `<Limit GET>` et `</Limit>` ou `<Files nom-fichier>` et `</Files>`
- On peut remplacer GET par POST ou GET POST.
- On y définit l'attribut `Require` qui devra être satisfait pour que la ressource soit délivrée.
- Sa valeur peut être
 - `user nom-utilisateur,`
 - `group nom-de-groupe,`

Fichier .htaccess : Deuxième section

- La deuxième section se situe entre les balises `<Limit GET>` et `</Limit>` ou `<Files nom-fichier>` et `</Files>`
- On peut remplacer GET par POST ou GET POST.
- On y définit l'attribut `Require` qui devra être satisfait pour que la ressource soit délivrée.
- Sa valeur peut être
 - `user nom-utilisateur,`
 - `group nom-de-groupe,`
 - `valid-user.`

Fichier .htaccess : Deuxième section

- La deuxième section se situe entre les balises `<Limit GET>` et `</Limit>` ou `<Files nom-fichier>` et `</Files>`
- On peut remplacer GET par POST ou GET POST.
- On y définit l'attribut `Require` qui devra être satisfait pour que la ressource soit délivrée.
- Sa valeur peut être
 - `user nom-utilisateur`,
 - `group nom-de-groupe`,
 - `valid-user`.

↪ On peut définir plusieurs fois cette valeur : l'une des conditions devra être satisfaite.

Fichier `.htaccess` : Fichier de mots de passe

- Le fichier de mots de passe est un fichier texte contenant sur chacune de ses lignes

```
nom-utilisateur:mot-de-passe
```

Fichier `.htaccess` : Fichier de mots de passe

- Le fichier de mots de passe est un fichier texte contenant sur chacune de ses lignes

```
nom-utilisateur:mot-de-passe
```

- Le mot de passe peut être écrit en clair ou crypté.

Fichier `.htaccess` : Fichier de mots de passe

- Le fichier de mots de passe est un fichier texte contenant sur chacune de ses lignes

```
nom-utilisateur:mot-de-passe
```

- Le mot de passe peut être écrit en clair ou crypté.
- Exemple de fichier non crypté :
JDoe:Toto14
Dupond:Dupont

Fichier `.htaccess` : Fichier de mots de passes cryptés

- Pour crypté les mots de passe, on utilise `htpasswd`

Fichier `.htaccess` : Fichier de mots de passes cryptés

- Pour crypté les mots de passe, on utilise `htpasswd`
- La commande `htpasswd -c .htpasswd JDoe`, crée le fichier `.htpasswd`, demande un mot de passe et ajoute l'entrée

```
JDoe :mot-de-passe-crypte
```

Fichier `.htaccess` : Fichier de mots de passes cryptés

- Pour crypté les mots de passe, on utilise `htpasswd`
- La commande `htpasswd -c .htpasswd JDoe`, crée le fichier `.htpasswd`, demande un mot de passe et ajoute l'entrée

```
JDoe :mot-de-passe-crypte
```

- Si le fichier `.htpasswd` est déjà créé, on utilise

```
htpasswd .htpasswd Dupond.
```

Fichier `.htaccess` : Fichier de mots de passes cryptés

- Pour crypté les mots de passe, on utilise `htpasswd`
- La commande `htpasswd -c .htpasswd JDoe`, crée le fichier `.htpasswd`, demande un mot de passe et ajoute l'entrée

```
JDoe :mot-de-passe-crypte
```

- Si le fichier `.htpasswd` est déjà créé, on utilise

```
htpasswd .htpasswd Dupond.
```

- Exemple de fichier crypté :
JDoe :KDG11aLuG27ME
Dupond :P1dSGER5LJ/tg

Fichier .htaccess : Exemple

- Fichier .htaccess

```
AuthUserFile /var/www/.htpasswd
AuthName ''Accès restreints''
AuthType Basic
```

```
<LIMIT GET POST>
Require valid-user
</LIMIT>
```

- Fichier .htpasswd

```
JDoe :KDG11aLuG27ME
Dupond :P1dSGER5LJ/tg
```

Fichier .htaccess : Exemple

- Fichier .htaccess
AuthUserFile /var/www/.htpasswd
AuthName ''Accès restreints''
AuthType Basic

```
<FILES toto.html>  
Require user JDoe  
</FILES>
```

- Fichier .htpasswd
JDoe:KDG11aLuG27ME
Dupond:P1dSGER5LJ/tg

Type MIME

- Un Internet media type, ou Type MIME encore Content-type, est un identifiant de format de données sur internet en deux parties.

Type MIME

- Un Internet media type, ou Type MIME encore Content-type, est un identifiant de format de données sur internet en deux parties.
- Un type MIME est composé d'au moins deux parties : un type et un sous-type et d'un ou plusieurs autres champs au besoin.

Type MIME

- Un Internet media type, ou Type MIME encore Content-type, est un identifiant de format de données sur internet en deux parties.
- Un type MIME est composé d'au moins deux parties : un type et un sous-type et d'un ou plusieurs autres champs au besoin.
- Par exemple, les sous-types du type text ont un champ optionnel charset indiquant le codage des caractères.

Type MIME

- Un Internet media type, ou Type MIME encore Content-type, est un identifiant de format de données sur internet en deux parties.
- Un type MIME est composé d'au moins deux parties : un type et un sous-type et d'un ou plusieurs autres champs au besoin.
- Par exemple, les sous-types du type text ont un champ optionnel charset indiquant le codage des caractères.
- Les types et les sous-types qui commencent par "x-" ne sont pas standard ; ils ne peuvent pas être enregistrés par l'IANA.

Type MIME

- Un Internet media type, ou Type MIME encore Content-type, est un identifiant de format de données sur internet en deux parties.
- Un type MIME est composé d'au moins deux parties : un type et un sous-type et d'un ou plusieurs autres champs au besoin.
- Par exemple, les sous-types du type text ont un champ optionnel charset indiquant le codage des caractères.
- Les types et les sous-types qui commencent par "x-" ne sont pas standard ; ils ne peuvent pas être enregistrés par l'IANA.
- Les sous-types qui commencent par "vnd." sont la propriété du revendeur.

Fichier `.htaccess` : Suite et fin

- A l'aide du fichier `.htaccess`, on peut associer une extension de fichier à un type MIME.

Fichier .htaccess : Suite et fin

- A l'aide du fichier .htaccess, on peut associer une extension de fichier à un type MIME.
- On utilise la commande

```
AddType (mime/type [liste d'extension])
```

Fichier .htaccess : Suite et fin

- A l'aide du fichier .htaccess, on peut associer une extension de fichier à un type MIME.

- On utilise la commande

```
AddType (mime/type [liste d'extension])
```

- Exemple : `AddType application/x-httpd-php .htm`

Fichier .htaccess : Suite et fin

- A l'aide du fichier .htaccess, on peut associer une extension de fichier à un type MIME.

- On utilise la commande

```
AddType (mime/type [liste d'extension])
```

- Exemple : `AddType application/x-httpd-php .htm`

- L'ajout de la commande

```
ErrorDocument XXX [nom de fichier ou texte]
```

permet d'afficher le texte ou charger l'url lors de l'erreur HTTP XXX

Fichier .htaccess : Suite et fin

- A l'aide du fichier .htaccess, on peut associer une extension de fichier à un type MIME.

- On utilise la commande

```
AddType (mime/type [liste d'extension])
```

- Exemple : `AddType application/x-httpd-php .htm`

- L'ajout de la commande

```
ErrorDocument XXX [nom de fichier ou texte]
```

permet d'afficher le texte ou charger l'url lors de l'erreur HTTP XXX

- Exemple : `ErrorDocument 404 Document non trouvé ;)`