



## HTTPD

---

### Rôle

---

- httpd est le programme du serveur HTTP d'Apache. Il a été conçu pour fonctionner sous forme de processus démon indépendant. Lorsqu'il est utilisé ainsi, il va créer un jeu de processus enfants ou de threads qui traiteront les requêtes.
- Le service httpd traite les requêtes http et il laisse le traitement de certaines requêtes http au WIService et faisant une réécriture des url vers le port http du wiservice (par défaut 1508) et en préfixant les url par wiapp

### Installation

---

#### Configuration du serveur

- OS : RedHat 9.x ou Rocky Linux 9.x
- Cpu : 2 CPU 2coeurs/cpu
- Ram : 4 Gb
- Hdd : 64 Gb
- Réseau/firewall
- Accès en SSH depuis chez Infodata et avoir les accès root pour faire l'installation
- Accès http et https vers le serveur web, à travers votre firewall (reverse proxy ou NAT) (remarque : le http aura une seule tâche : faire une redirection vers https)
- Accès sur le port Wiservice (TCP/1505) à partir du serveur UV
- adresse IP WAN fixe
- enregistrement DNS "{sous-domaine}.{client\_domaine}.lu" qui pointe vers l'IP WAN fixe

### httpd

---

- En principe, httpd a été installé par défaut sur le serveur linux.
- Si ce n'est pas le cas, vous pouvez suivre la [procédure d'installation de httpd](#)

### Configuration

---

Pour que service httpd fonctionne avec le WIService, il y a quelques adaptations à faire dans le fichier config de httpd.



Suivant les versions de Linux / Apache, la configuration peut être un peu différente, se référer à la documentation correspondante à la bonne version

- Vérifier que les modules indispensables sont activés dans apache
  - Dans le répertoire "**conf**" il doit exister un fichier **httpd.conf** (suivant la version, peut-être fait-il des includes d'autres fichiers)
  - c'est l'instruction *LoadModule* qui fait que apache charge les modules dont on a besoin
  - Les modules suivants (ou équivalent) doivent être actifs :

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
LoadModule proxy_http_module modules/mod_proxy_http.so

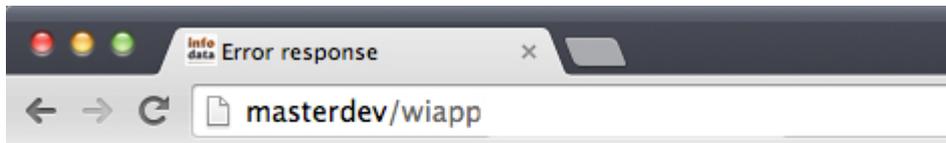
LoadModule rewrite_module modules/mod_rewrite.so
```



Sur un serveur suze, Il faut ajouter les trois modules suivants dans le paramètre **APACHE\_MODULES** du fichier `/etc/sysconfig/apache2` :

- rewrite
- proxy
- proxy\_http





## Error response

Error code 500.

Message: Unknown error : 500 [WI] Invalid application id ><.

Error code explanation: 500 = Server got itself in trouble.

### .htaccess

Pour rendre transparent le passage par une URL générique pour les requêtes qui vont dans le système, on peut configurer l'application web pour faire des redirections elle aussi.

Pour ce faire, apache autorise l'utilisation de fichier de configuration (par défaut le fichier s'appelle **.htaccess**) dans les répertoires du site web.

Exemple de fichier pour l'application websav.infodata.lu :

- [Exemple de fichier .htaccess](#)

Dans cet exemple, l'url **/websav/login** sera redirigée vers le wiservice avec une URL complétée de paramètre **/wiapp/login?w3exec...&w3hostname=...**



Pour que le service httpd d'apache prenne en compte la configuration du fichier .htaccess, la directive AllowOverride ne doit pas être à None dans le fichier httpd.conf.

Voir doc [allowoverride](#) pour connaître la configuration à mettre suivant votre utilisation.

Si vous faites des réécritures d'url, il faut ajouter les options "Options FileInfo".

Exemple :

```
#
# AllowOverride controls what directives may be placed in .htaccess
# files.
# It can be "All", "None", or any combination of the keywords:
#   Options FileInfo AuthConfig Limit
#
AllowOverride Options FileInfo
```

### Htaccess en cas de maintenance

Lors d'une maintenance, il est possible de remplacer temporairement l'htaccess pour que celui oriente vers une page de maintenance (pour l'instant, intégrée au projet). Pour ce faire, remplacer le contenu de l'htaccess par en remplaçant **#SOUS-DOMAIN#** par le sous-domaine si nécessaire:

```
# MAINTENANCE-PAGE REDIRECT
<IfModule mod_rewrite.c>
  RewriteEngine on
  RewriteBase /#SOUS-DOMAIN#/
  RewriteCond %{REQUEST_URI} !maintenance.html$ [NC]
  RewriteCond %{REQUEST_URI} !\.(jpe?g?|png|gif) [NC]
  RewriteRule .* maintenance.html [R=302,L]
</IfModule>
```



N'oubliez pas de copier le contenu de votre htaccess avant de l'écraser

## Procédure

---

### Service httpd

Lecture d'un état Normal

Pour savoir si le service httpd est démarré ou non, Il faut lancer la commande `httpd status`

Exemple s'il est déjà démarré :

```
[root@masterdev ~]# service httpd status
httpd (pid 5378) is running...
```

Exemple s'il n'est pas démarré :

```
[root@masterdev ~]# service httpd status
httpd is stopped
```

- sous centos7

```
systemctl status httpd
```

Procédure de start

Pour démarrer le service httpd d'apache, Il faut lancer la commande `httpd start`

Exemple s'il n'était pas encore démarré avant le start :

```
[root@masterdev ~]# service httpd start
Starting httpd: [ OK ]
```

Exemple s'il était déjà démarré :

```
[root@masterdev ~]# service httpd start
Starting httpd:
```

- sous centos7

```
systemctl start httpd
```

Procédure de stop

Pour arrêter le service httpd d'apache, Il faut lancer la commande `httpd stop`

Exemple d'arrêt si le service était démarré :

```
[root@masterdev ~]# service httpd status
httpd is stopped
```

- sous centos7

```
systemctl stop httpd
```

Procédure d'urgence pour un (re)start

Pour redémarrer le service httpd d'apache, Il faut lancer la commande `httpd restart`

Exemple s'il n'était pas déjà démarré avant le restart :

```
[root@masterdev ~]# service httpd restart
Stopping httpd: [ OK ]
Starting httpd: [ OK ]
```

- sous centos7

```
systemctl restart httpd
```

## Logs

### Logs d'activités

Le fichier des logs d'activités se trouve par défaut : `/etc/httpd/logs/access_log`

Le chemin de log peut être modifié dans le fichier **httpd.conf** du répertoire `httpd`

### Logs de warning/erreurs

Le fichier des logs d'activités se trouve ici `:/etc/httpd/logs/error_log`

Le chemin de log des erreur peut être modifié dans le fichier **httpd.conf** du répertoire `httpd`.

## Dépendances

---

1. au Wiservice et donc à WI.HDL. Il est donc nécessaire au bon fonctionnement des pages web et du web service