

HTTP-Serveur

Jean-François Berdjugin
IUT1, département SRC, L'Isle
d'Abeau

Références

- <http://www.apache.org/>
- http://www.lea-linux.org/cached/index/Reseau-web-apache_conf.html
- RFC 2616

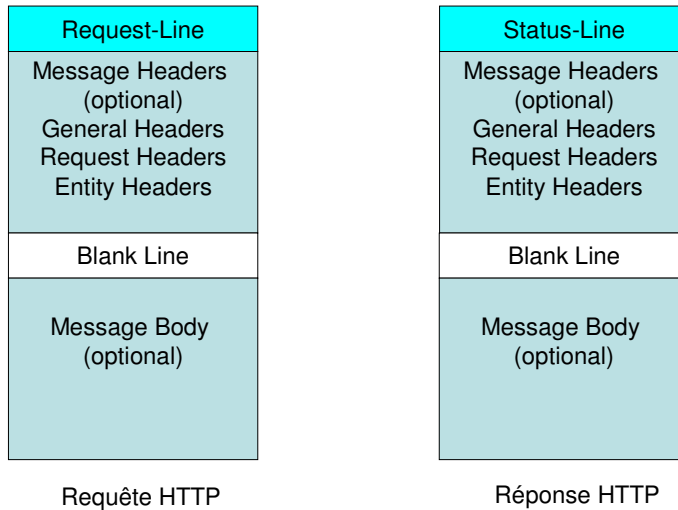
Plan

- HTTP 1.1
- Les principaux serveur
- Apache et le web dynamique

HTTP 1.1

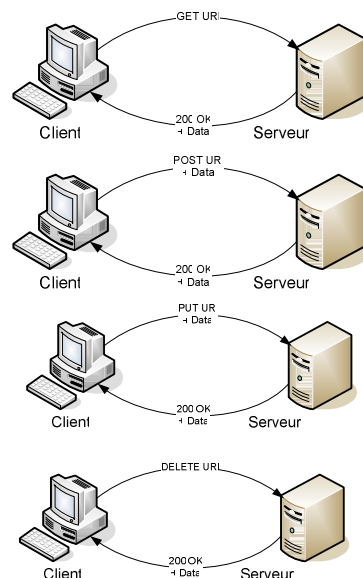
- Client/Serveur
- Persistance de la connexion
- Plusieurs sites sur la mêmes machines => Virtual Host
- Un site sur plusieurs machines => Redirection
- Contexte de session => Cookies
- Gestion des caches

Structure d'une requête et d'une réponse



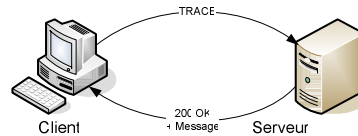
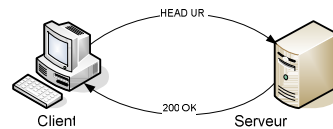
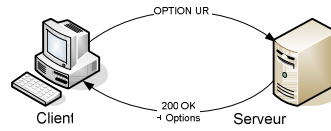
Opérations utilisateur

- GET : Récupérer une ressource
- POST : Envoi d'information à un objet (URI : script, programme)
- PUT : Envoi de données (URI : path + file)
- DELETE : Suppression d'objet



Autres opérations

- OPTIONS : découverte des capacités du serveur (URI ou *)
 - HEAD : identique à GET mais sans envoi de données
 - TRACE : traçage du chemin réseau (réexpédition de ce qui a été reçu) ?
- => Coopération de serveurs



Méthodes/Status Code

- CONNECT
- DELETE
- GET
- HEAD
- OPTIONS
- POST
- PUT
- TRACE
- 100-199 : Le serveur a bien reçu la requête mais le résultat final n'est pas disponible.
- 200-299 : Succès
- 300-399 : Redirection
- 400-499 : Erreur Client
- 500-599 : Erreur Serveur

Champs d'entête

Les méthodes des requêtes et les codes de réponses peuvent être complétés par des lignes entêtes.

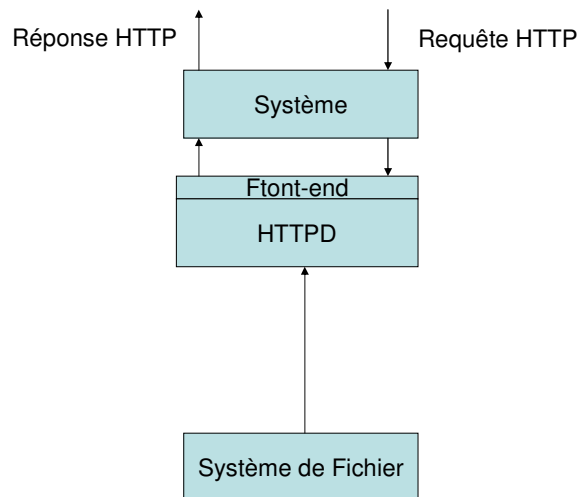
- Requête
 - > GET / HTTP/1.0
 - > Accept: */*
 - > Accept-Language: fr
 - > Proxy-Connection: Keep-Alive
 - > If-Modified-Since: Tue, 13 Apr 2004 17:45:01 GMT; length=40838
 - > User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.1.4322)
 - > Host: www.free.fr
 - > Pragma: no-cache
 - >
- Réponse
 - HTTP/1.1 304 Not Modified
 - Date: Tue, 13 Apr 2004 17:45:56 GMT
 - Server: Apache/1.3.26 (Unix) Debian GNU/Linux
 - Connection: close
 - ETag: "6929-9f86-407c271d"

Comparatif serveur

- En 2003

Apache	62,72%
Microsoft IIS	27,01%
Zeus	2,14%
- Apache libre fortement supporté sur tout type de plateforme
- IIS en standard avec le 2003 Serveur
- Zeus Serveur payant haute performance

Web Statique



Web Dynamique

- Le serveur exécute, le client reçoit
 - SSI, eXtended Server Side Include: un langage de script compromis entre CGI et PHP
 - Common Gateway Interface, FastCGI,
 - PHP (Hypertext Preprocessor)
 - Active Server Page,
 - Java Server Page
- Le serveur envoie, le client exécute
 - JavaScript
 - Applet Java,
 - Flash

SSI

```
<!--#exec cmd="ls -l"
-->
```

Donne le résultat d'un ls sur le serveur

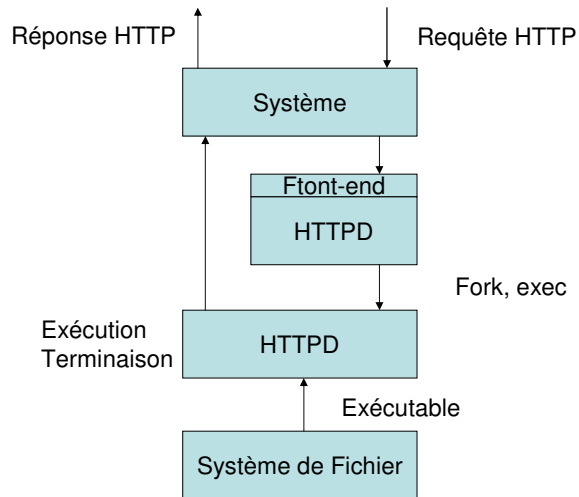
```
<!--#echo
var="DATE_LOCAL" -->
```

Donne l'heure sur le serveur

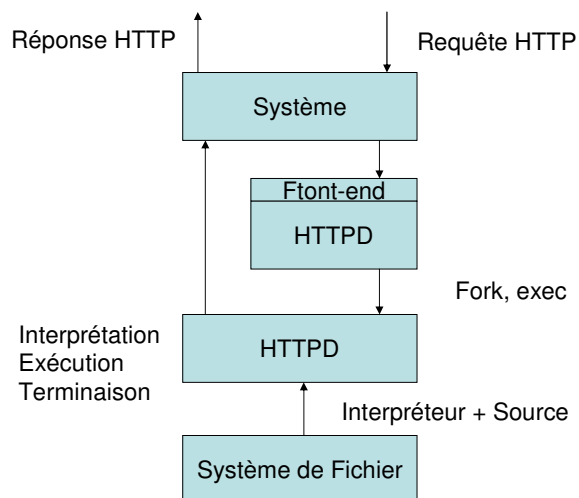
CGI

- Une passerelle entre le serveur d'information et les applications
- Permet de passer des paramètres aux requêtes : ?nom=valeur&...
- Exécution d'un programme sur le serveur puis renvoie d'une page statique.

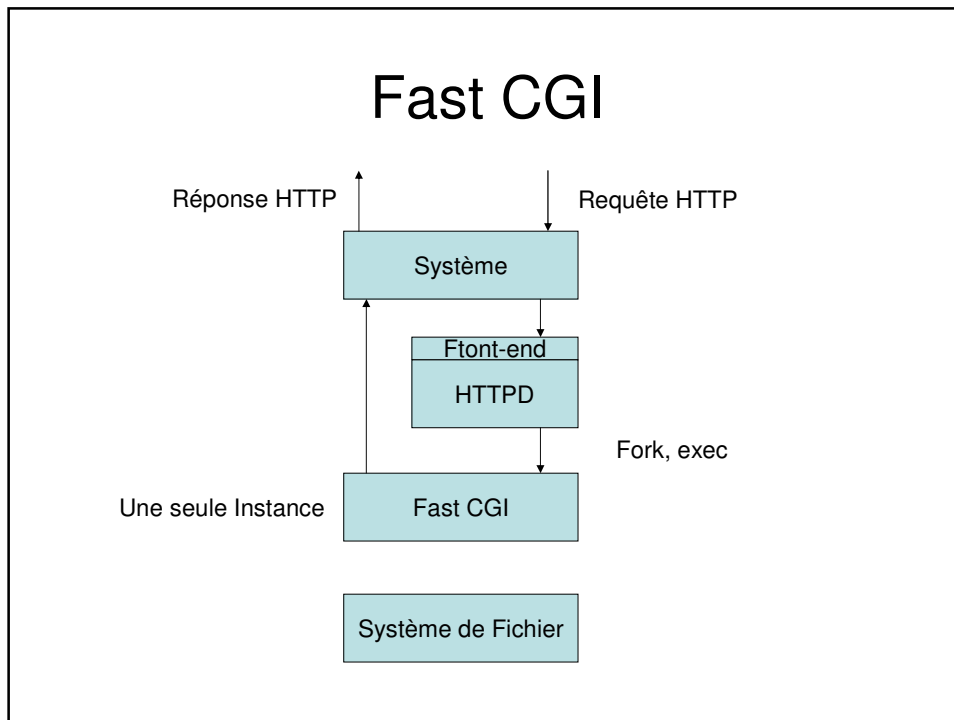
Exécution d'un CGI compilé via HTTP



Exécution d'un CGI interprété via HTTP



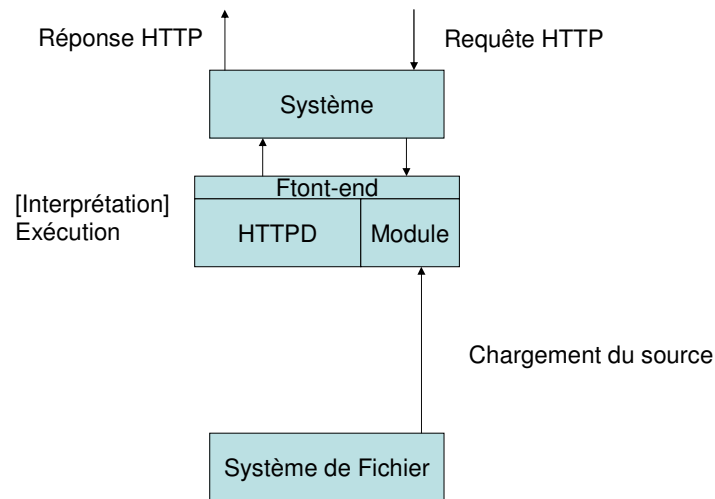
Fast CGI



Module apache

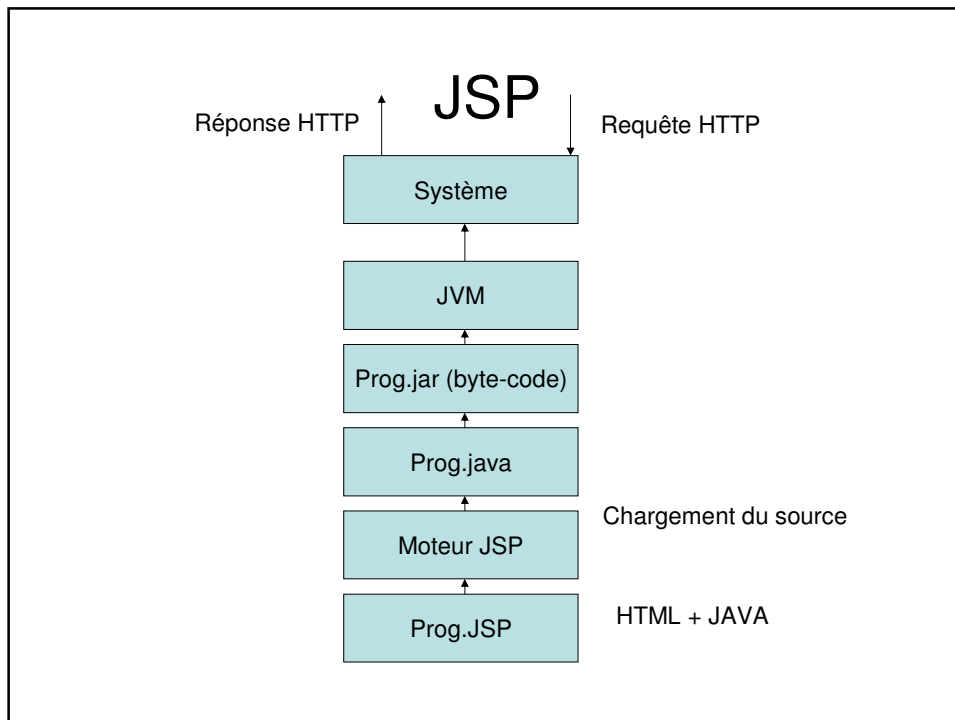
- Le serveur HTTP Apache est un programme modulaire permettant à l'administrateur de choisir les fonctionnalités qu'il souhaite activer, au moyen de modules. Les modules peuvent être intégrés dans le programme binaire httpd au moment de la compilation. Il est également possible de compiler à part des modules en tant qu'objets dynamiques partagés (Dynamic Shared Objects : DSOs)

Approche Modulaire (PHP, Perl)



JAVA

- Les servlets
 - Programme Java exécuté côté serveur
 - Transformation en byte-code avant exécution
 - Servies par un serveur dédié (ex : Tomcat)
- JSP (Java Server Pages)
 - Pages HTML avec Java embarqué
 - Transformation en Servlets



ASP/ASP.net

- Natif IIS
- Plusieurs langage : VB, C++, C#, JS, VBS, J#

Fichier Apache

- /usr/bin/httpd : le démon (binaire)
- /etc/init.d/httpd : le script de gestion du démon
 - start, stop, restart, reload, status
- /etc/httpd/conf/httpd.conf : le fichier de configuration
- /var/lib/apache/htdocs : les fichiers servis par le serveur

Rem : suivant la distribution la place de ces fichiers peut être modifiée.

Configuration d'apache

- Un fichier httpd.conf contenant des directives

```
#lancement en serveur autonome
ServerType    standalone

#Répertoire d'installation d'apache
ServerRoot    "/var/lib/apache"

#connexion persistante
KeepAlive on

#indication sur les adresses IP et les ports à utiliser
Listen    port
Listen    adresse IP
```

Httpd.conf

#permet avec la directive virtual host d'inclure des hôtes virtuels

BindAdress *

#Permet de charger des modules
LoadModule

#port par défaut du serveur
Port 80

#utilisateur et group du serveur
User nobody
Group nobody

#Adresse renvoyée au client
ServerName www.domainname

#Adresse du site web
DocumentRoot "/var/lib/apache/htdocs"

Httpd.conf

#règles particulières
pour un répertoire

<Directory xx>

Options :

AllowOverride :

Order :

Allow (ou deny):

</Directory>

- Options : None, All, Indexes, FollowSymLinks, ExecCGI, Includes, IncludesNOEXEC, Multiviews, SymLinksIfOwnerMatch
- **AllowOverride** : All, AuthConfig, FileInfo, Limit, None, Options

Httpd.conf

#exemple listing, liens symboliques, vues multiple, interdiction de réécrire les options accès depuis toute machine

```
<Directory "/var/lib/apache/htdocs">  
Options Indexes FollowSymlinks Multiviews  
AllowOverride None  
Order allow,deny  
allow from all  
</Directory>
```

#interdit aux visiteurs de voir le contenu des fichiers .ht

```
<Files ~ "\.ht">  
Order allow,deny  
Deny from all  
</Files>
```

Httpd.conf

#Alias sur des répertoires
Alias faux_nom nom_réel

#Alias sur des scripts
ScriptAlias /cgi-bin/ chemin_complet_des_cgi

#définit les extensions de fichiers
AddType type extensions

#par exemple
#AddType application/x-httpd-php .php
#AddType application/x-httpd-php-source .phps

Httpd.conf

```
#Virtuals Hosts
NameVirtualHost 80.10.20.30

<VirtualHost ip_ou_adresse_virtuelle>
  ServerAdmin
  email_webmaster@hoste_virtuel
  DocumentRoot racine_hote_virtuel
  ServerName nom_server_virtuel
  ErrorLog chmein_complet_errorlo
</VirtualHost>

<VirtualHost 80.10.20.30>
  ServerAdmin webmaster@domain1.com
  DocumentRoot /home/domain1/www
  ServerName www.domain1.com
  ErrorLog /var/log/apache/domain1-error.log
  CustomLog /var/log/apache/domain1-
  access.log combined
  ServerAlias domain1.com
</Virtualhost>

<VirtualHost 80.10.20.30>
  ServerAdmin webmaster@domain2.com
  DocumentRoot /home/domain2/www
  ServerName www.domain2.com
  ErrorLog /var/log/apache/domain2-error.log
  CustomLog /var/log/apache/domain2-
  access.log combined
  ServerAlias domain2.com
</Virtualhost>
```

.htaccess

- Un fichier qui si httpd.conf le permet redéfinit des directives pour le répertoire dans lequel il est placé ainsi que pour tous les sous-répertoires.
- Exemple :

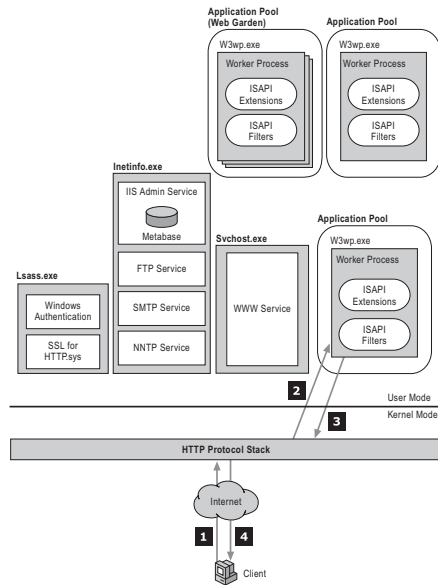
```
AuthType Basic
AuthName "By Invitation Only"
AuthUserFile /usr/local/apache/passwd/passwords
AuthGroupFile /usr/local/apache/passwd/groups
Require group GroupName
```

Avec la commande suivante pour générer le mot de passe de dpitts htpasswd
/usr/local/apache/passwd/passwords dpitts

Le fichier groupes pourra par exemple contenir : GroupName: rbowen dpitts
sungu rshersey

Rem : Require valid-user évite de gérer des groupes

IIS



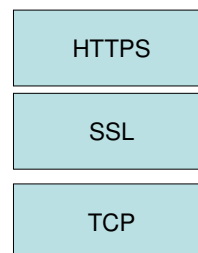
SSL

Possibilité avec HTTPS d'utiliser SSL pour sécuriser les transferts :

- Utilisation de certificat X509
- Pour le client et le serveur

=>

- Authentification
- Intégrité
- Confidentialité des données



Conclusion

- Deux standard avec apache leader en part de marché et IIS spécifique à certains produit microsoft.
- Des passerelles pour communiquer avec les applications
- Le reflet du protocole HTTP