



Apache

Apache 2.x

Dernière mise à jour : 9 septembre 2020

Version 2.0

[Christophe Vignaud](#)

Brest

ISEN | école
d'ingénieurs

Apache | 1

Plan

- Présentation
- Technologies concurrentes
- Fonctionnement du web
 - DNS/HTTP/HTML/entete
- Fonctionnement de base
- Fonctionnement évolué
 - Virtualhost
 - .htaccess
 - proxy

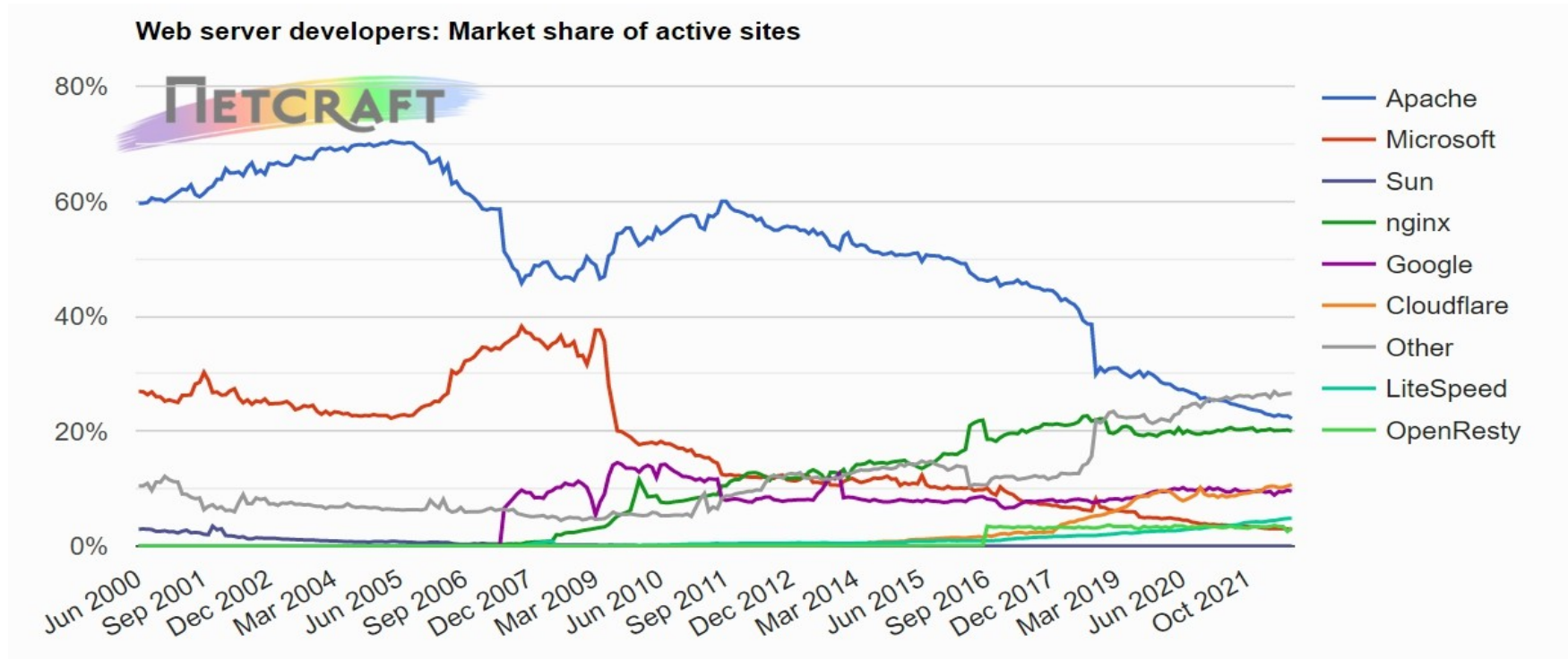


Carte d'identité

- Création en 1995 (évolution du serveur NCSA HTTPd)
- Open source
- Version actuelle => 2.4
- Serveur le plus utilisé jusqu'en 2016
- Fonctionnement à l'aide de modules (mods) et de fichiers de configuration (.conf)
- Multi-plateformes
- <https://httpd.apache.org>



Stat d'utilisation



<https://news.netcraft.com>

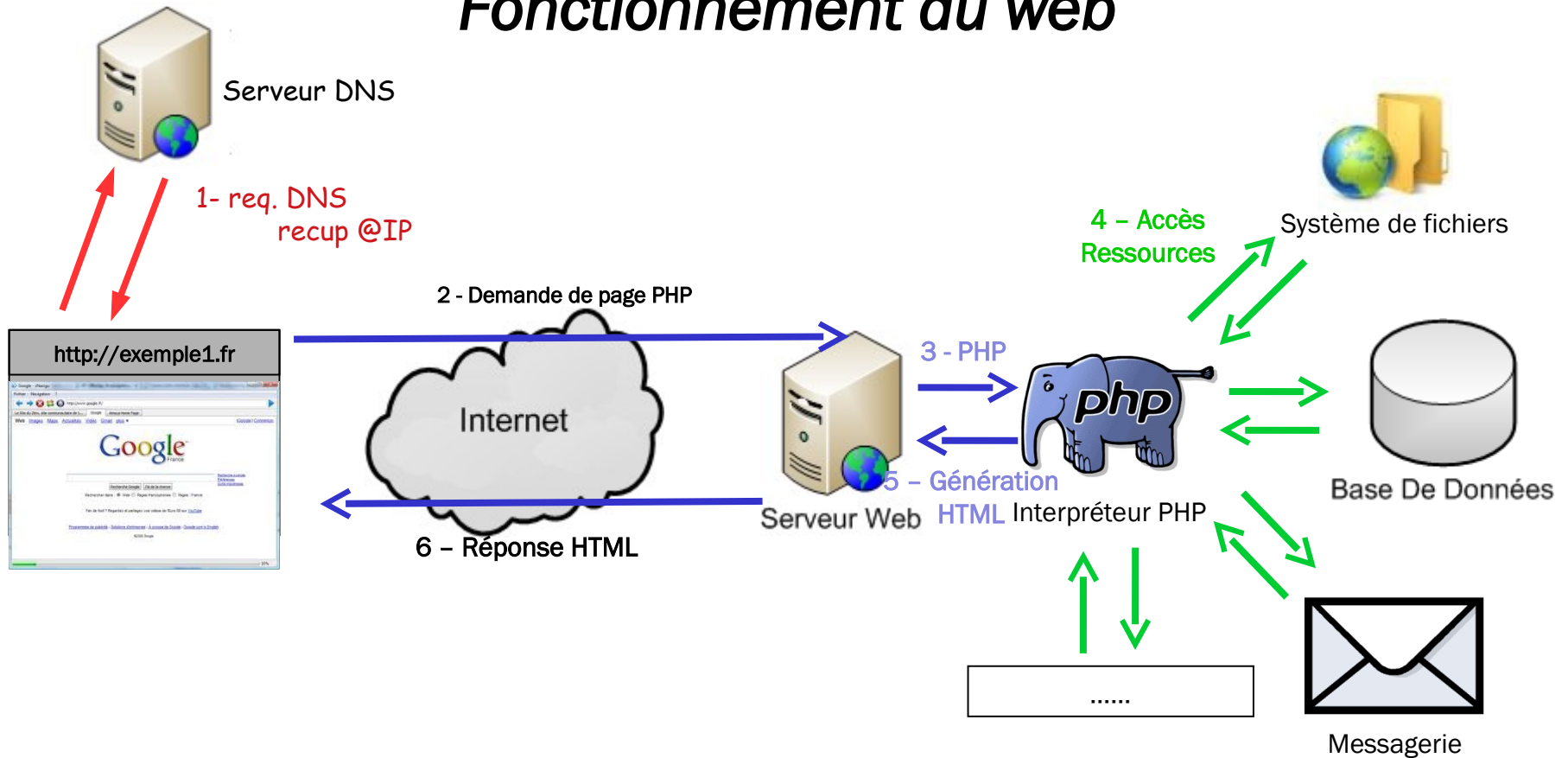


Technologies concurrentes

- Apache (version antérieure)
- Apache Tomcat (java)
- Microsoft IIS
- Sun Java System Web (SJSW)/Oracle iPlanet Web Server (OiWS)
- Nginx
- Google Web Server
- NodeJs
- autres



Fonctionnement du web



Fonctionnement

- Etape 1 => Adresse IP
 - Sur le navigateur => récupérer l'adresse IP pour l'url sélectionnée
 - Utilisation du DNS
 - Fichier hosts local (/etc/hosts, /windows/system32/drivers/etc/hosts=)
 - *Server DNS réseau local*
 - *Server DNS opérateur*

```
(base) christophe@christophe-HP-EliteBook-840-G3:~$ more /etc/hosts
127.0.0.1      localhost
127.0.0.1      monposte
127.0.1.1      christophe-HP-EliteBook-840-G3
```



Fonctionnement

- Etape 1 => Adresse IP

Détails	Identité	IPv4	IPv6	Sécurité
Vitesse de la connexion 1000 Mb/s				
Adresse IPv4 10.10.16.27				
Adresse IPv6 fe80::a04b:5ef2:9362:113f				
Adresse matérielle 98:E7:F4:2F:AB:DD				
Route par défaut 10.10.16.1				
DNS 10.10.0.53 172.16.0.3 10.20.0.53				





Fonctionnement

- Etape 2 => envoi requête HTTP
 - Protocole HTTP
 - Simple
 - Extensible
 - Sans état (mais gère les sessions)
 - S'appuie sur TCP => port 80
 - Gestion de la sécurité (origine unique)
 - Retour de code réussite/erreur (200, 404, 500, ...)





Fonctionnement

- Etape 2 => requête HTTP => envoi/réponse

▼ En-têtes de la requête (437 o)

```
GET /rh/ikvelo/ HTTP/1.1
Host: 172.16.0.18
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: fr,fr-FR;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: PHPSESSID=7unh3kqq03jftqlep5n14kjkeq
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

▼ En-têtes de la réponse (305 o)

```
HTTP/1.1 200 OK
Date: Mon, 13 Jul 2020 13:19:09 GMT
Server: Apache/2.4.25 (Debian)
Cache-Control: max-age=0, must-revalidate, private
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 1559
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
```





Fonctionnement

- Etape 3/4 => serveur web
 - Gestion dynamique
 - Des traitements
 - Des données
 - Utilisation de moteurs (PHP, java, Python, CGI, ...)
 - Génération de la sortie dans le format approprié (HTML, JSON, XML, ...)





Fonctionnement

- Etape 5/6 => envoi de la réponse
 - Création de la trame HTTP contenant la réponse
 - Dans le bon format (HTML, JSON, ...)
 - Avec les bonnes en-têtes (code erreur, ...)





Code de retour

- Le serveur web retourne un code HTTP
 - Les codes les plus courants
 - 200 : succès de la requête ;
 - 301 et 302 : redirection, respectivement permanente et temporaire ;
 - 401 : utilisateur non authentifié ;
 - 403 : accès refusé ;
 - 404 : page non trouvée ;
 - 500 et 503 : erreur serveur ;
 - 504 : le serveur n'a pas répondu.





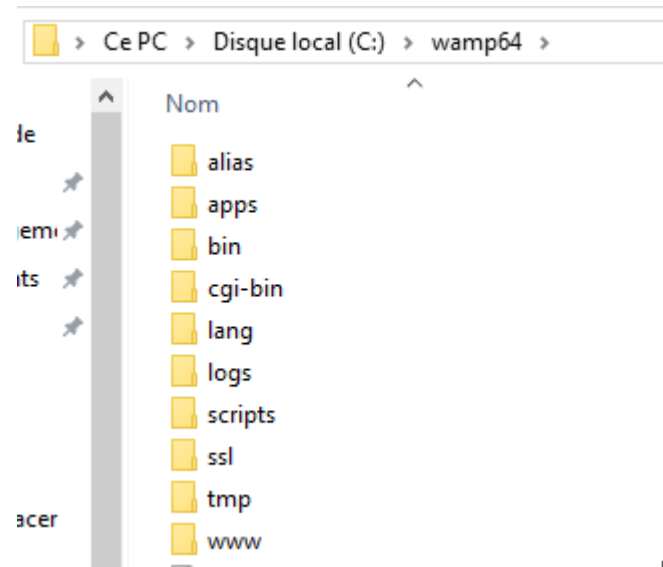
Installation

- Serveur d'entreprise sous linux
 - Nom de domaine unique
 - Multiple noms de domaines
- Serveur local sous windows (stand alone)
- Serveur local sous windows (WAMP, XAMP)



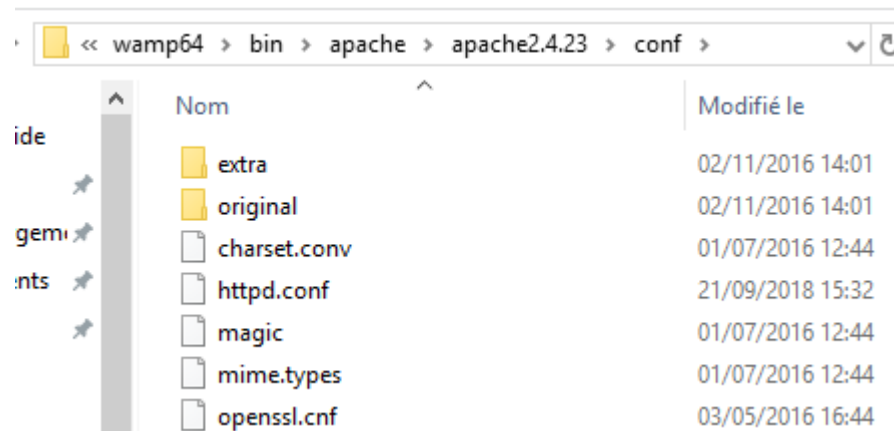
Installation avec WAMP

- De nombreuses bibliothèques à installer
- Intègre PHP, MYSQL, PhpMyAdmin, ...
- Pages web sous www



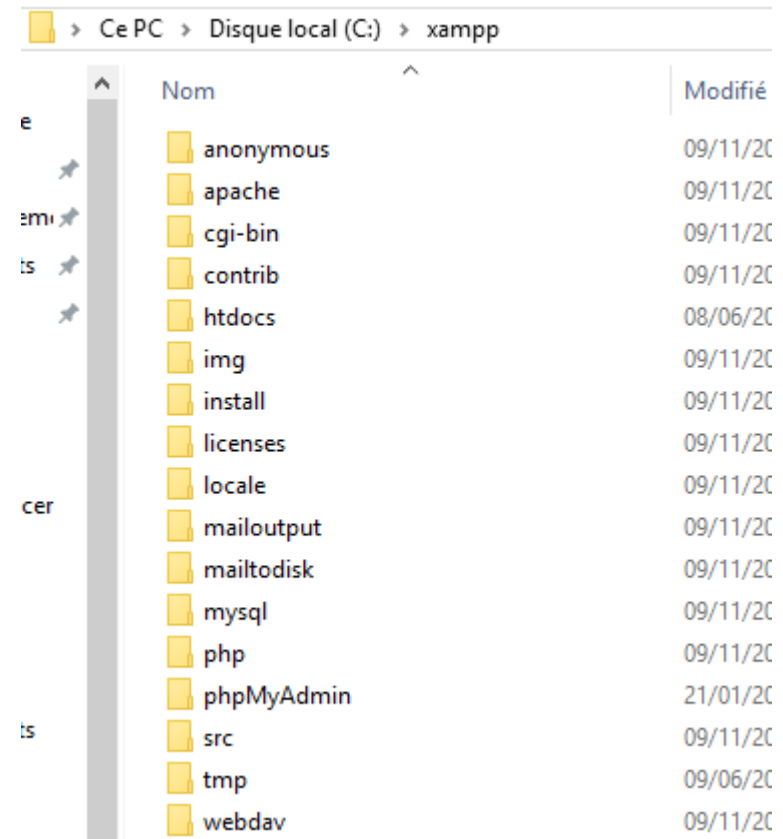
Installation avec WAMP

- Configuration sous wampxx/bin/apachexx/conf



Installation avec XAMP

- Simple à installer
- Intègre PHP, MariaDb, PhpMyAdmin, ...
- Pages web sous htdocs

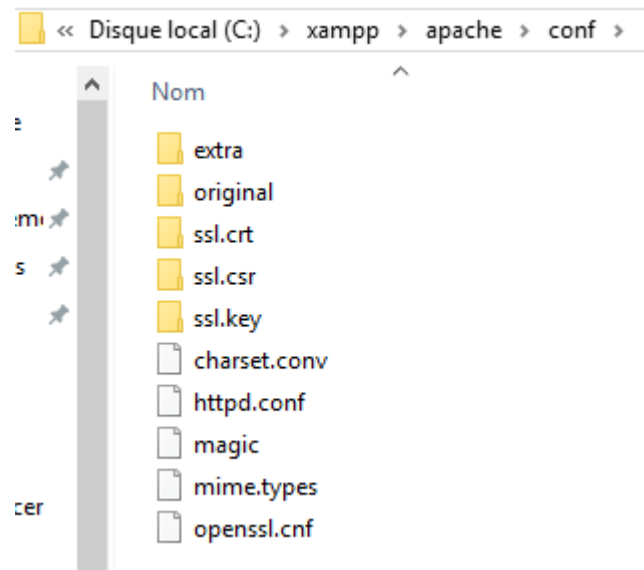


	Nom	Modifié
	anonymous	09/11/20
	apache	09/11/20
	cgi-bin	09/11/20
	contrib	09/11/20
	htdocs	08/06/20
	img	09/11/20
	install	09/11/20
	licenses	09/11/20
	locale	09/11/20
	mailoutput	09/11/20
	mailtodisk	09/11/20
	mysql	09/11/20
	php	09/11/20
	phpMyAdmin	21/01/20
	src	09/11/20
	tmp	09/06/20
	webdav	09/11/20



Installation avec XAMP

- Configuration sous xamp/apache/conf



Installation sous linux

- Installation par apt (debian)
`sudo apt install apache2`
- Installer les paquets nécessaire par appli
 - Php => libapache2-mod-php
 - Mysql => libapache2-mod-auth-mysql
 - PostGreSql => libapache2-mod-auth-pgsql
- Pages web sous `/var/www`
- Un nouveau utilisateur est crée => `www-data`



Installation sous linux

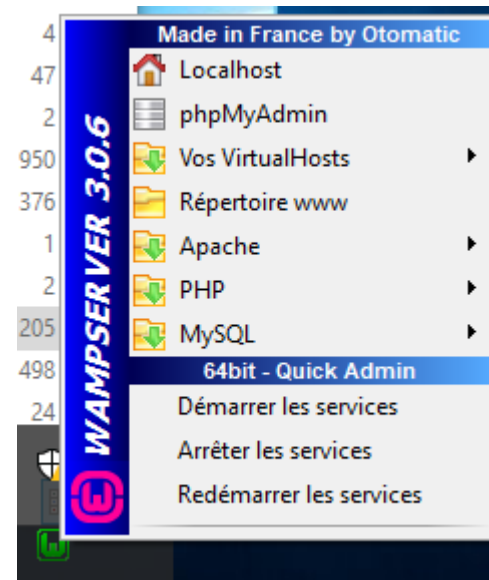
- Fichiers de conf sous /etc/apache2

```
(base) christophe@christophe-HP-EliteBook-840-G3:~$ ls -all /etc/apache2/
total 116
drwxr-xr-x  9 root root  4096 mars  23 08:49 .
drwxr-xr-x 182 root root 12288 juil.  9 11:54 ..
-rw-r--r--  1 root root  7224 juin  27  2018 apache2.conf
-rw-r--r--  1 root root  7117 avril  4  2017 apache2.conf.dpkg-old
drwxr-xr-x  2 root root  4096 mars  23 08:49 conf-available
drwxr-xr-x  2 root root  4096 sept.  1  2016 conf-enabled
-rw-r--r--  1 root root  1782 juin  27  2018 envvars
-rw-r--r--  1 root root  1853 avril  4  2017 envvars.dpkg-old
-rw-r--r--  1 root root 31063 mars  19  2016 magic
drwxr-xr-x  2 root root 16384 mai  28 09:59 mods-available
drwxr-xr-x  2 root root  4096 juin  8 16:20 mods-enabled
-rw-r--r--  1 root root   320 mars  19  2016 ports.conf
drwxr-xr-x  2 root root  4096 juin  24 22:49 sites-available
drwxr-xr-x  2 root root  4096 juin  14 18:23 sites-enabled
drwxr-xr-x  2 root root  4096 août  24  2018 ssl
```



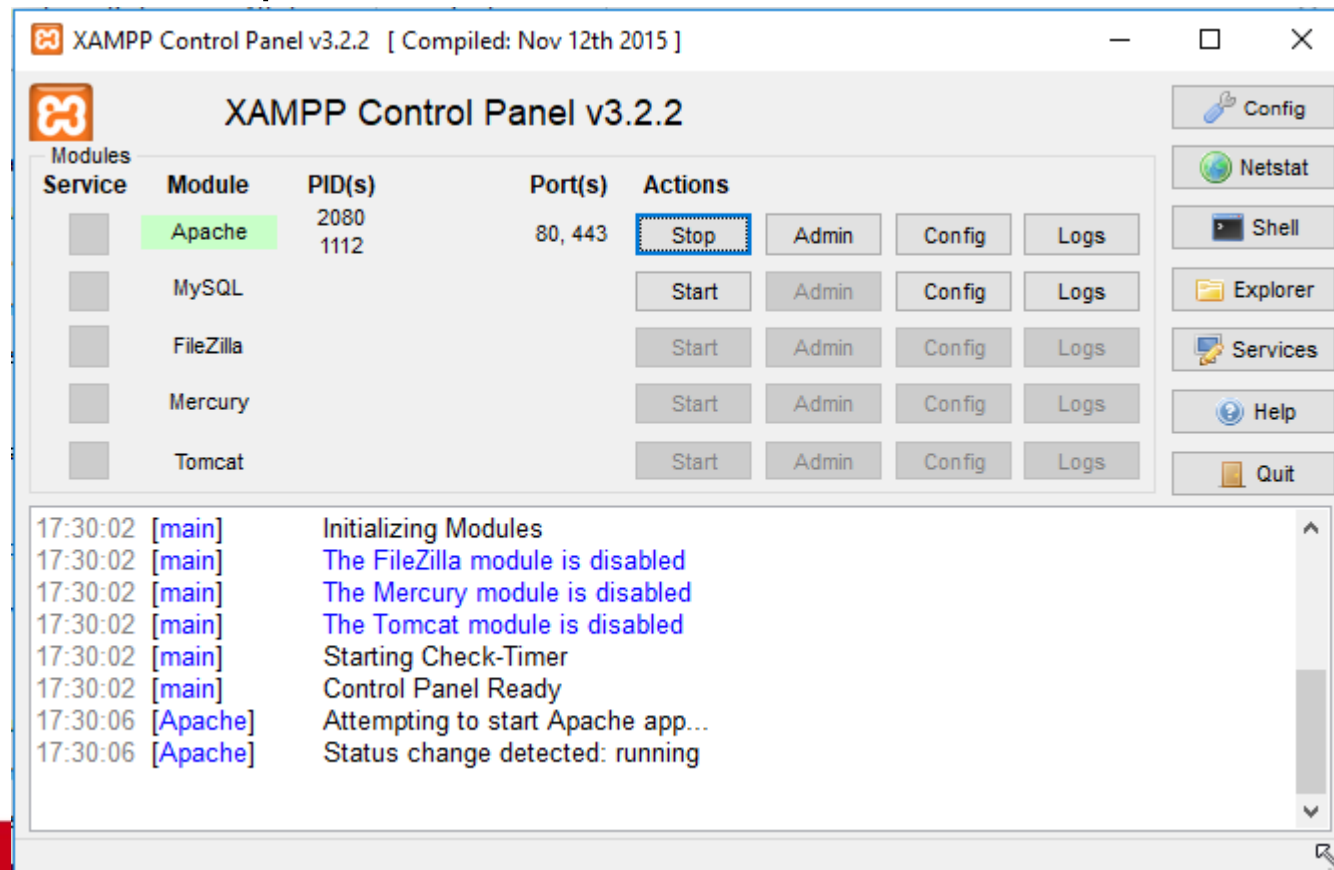
Lancer Apache avec WAMP

- Utiliser wampmanager



Lancer Apache avec XAMP

- Utiliser xamp-control



Lancer Apache avec Linux

- En ligne de commande

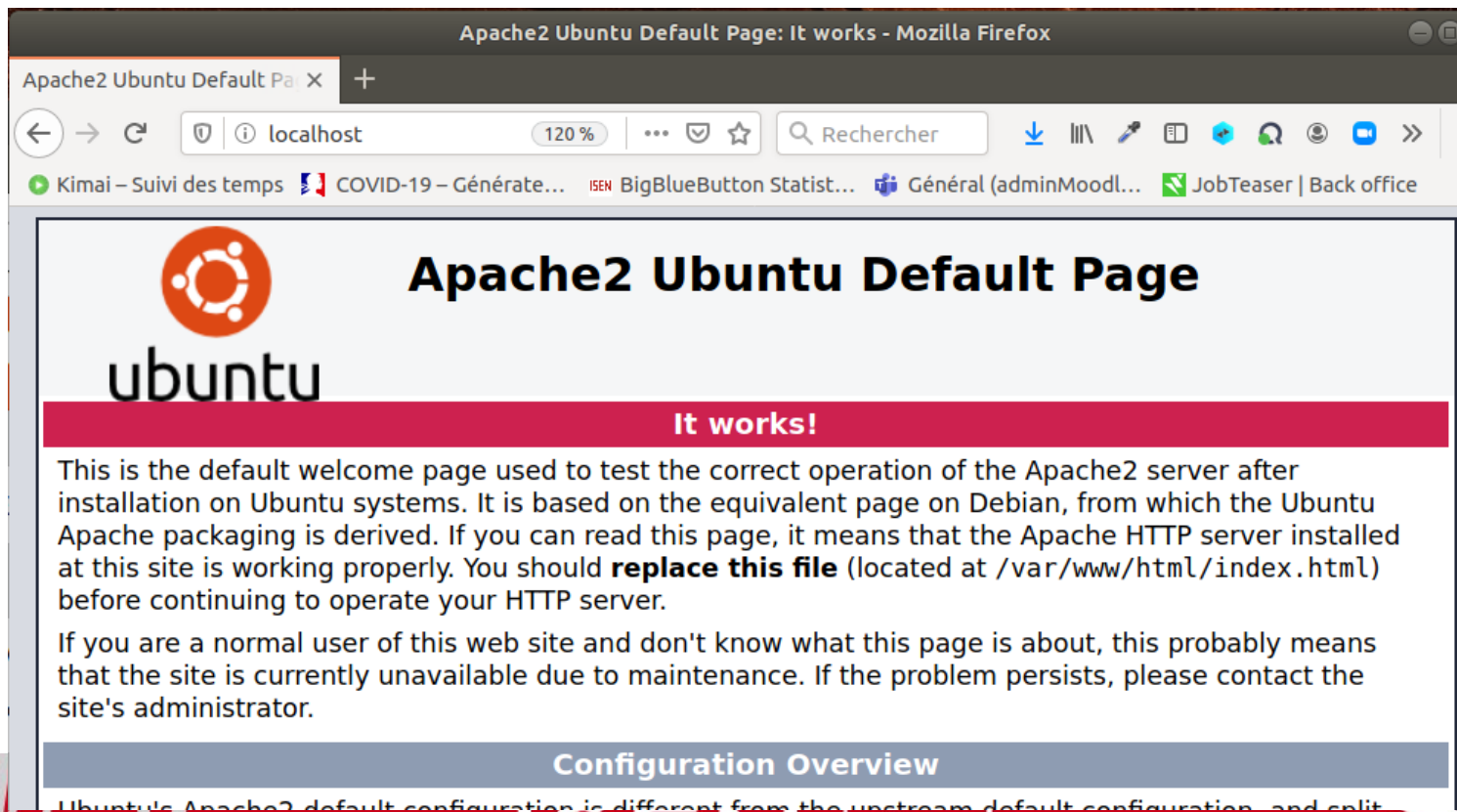
```
sudo systemctl start apache2
```

```
(base) christophe@christophe-HP-EliteBook-840-G3:~$ sudo systemctl start apache2
(base) christophe@christophe-HP-EliteBook-840-G3:~$ ps -ef | grep apache
root      1404      1  0 17:34 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1409    1404  0 17:34 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1410    1404  0 17:34 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1414    1404  0 17:34 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1416    1404  0 17:34 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  1417    1404  0 17:34 ?        00:00:00 /usr/sbin/apache2 -k start
christo+  1419  25769  0 17:34 pts/0    00:00:00 grep --color=auto apache
```



Visualisation de la page index.html

Avec un navigateur => http://localhost



Configuration

Fonctionnement sous forme

available => enabled

- Fichiers de conf
- Modules
- sites

```
(base) christophe@christophe-HP-EliteBook-840-G3:/etc/apache2$ ls -all
total 116
drwxr-xr-x  9 root root  4096 mars  23 08:49 .
drwxr-xr-x 182 root root 12288 juil. 15 11:10 ..
-rw-r--r--  1 root root  7224 juin  27  2018 apache2.conf
-rw-r--r--  1 root root  7117 avril  4  2017 apache2.conf.dpkg-old
drwxr-xr-x  2 root root  4096 mars  23 08:49 conf-available
drwxr-xr-x  2 root root  4096 sept.  1  2016 conf-enabled
-rw-r--r--  1 root root  1782 juin  27  2018 envvars
-rw-r--r--  1 root root  1853 avril  4  2017 envvars.dpkg-old
-rw-r--r--  1 root root 31063 mars  19  2016 magic
drwxr-xr-x  2 root root 16384 mai  28 09:59 mods-available
drwxr-xr-x  2 root root  4096 juin   8 16:20 mods-enabled
-rw-r--r--  1 root root   320 mars  19  2016 ports.conf
drwxr-xr-x  2 root root  4096 juin  24 22:49 sites-available
drwxr-xr-x  2 root root  4096 juin  14 18:23 sites-enabled
drwxr-xr-x  2 root root  4096 août  24  2018 ssl
```



Configuration

Fichiers de configuration de base

- http.conf (installation classique)
- **apache2.conf** (installation sous Debian)
- Utilise plusieurs fichiers de configuration par fonctionnalités et par modules

```
#  
# /etc/apache2/  
# |-- apache2.conf  
# |   |-- ports.conf  
# |   |-- mods-enabled  
# |       |-- *.load  
# |       |-- *.conf  
# |   |-- conf-enabled  
# |       |-- *.conf  
# |   |-- sites-enabled  
# |       |-- *.conf  
#  
#
```





Configuration

apache2.conf contient

- Paramétrage des log
- Paramètres de temps de sessions, de connexion, ...
- L'accès aux autres fichiers de conf
- l'accès aux fichiers de modules
- l'accès au fichiers de sites
- La configuration minimale pour `/var/www`



Configuration

ports.conf contient

- La définition des ports pour l'accès au serveur web
- 80 => accès HTTP / 443 => accès HTTPS

```
(base) christophe@christophe-HP-EliteBook-840-G3:/etc/apache2$ more ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

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





Outils

Apache comprend différents outils d'administration

- service apache2
- apachectl
- a2enmod
- a2enconf
- a2ensite



Outils

systemctl xxx apache2 permet de relancer, arreter, stopper => s'utilise avec sudo

- systemctl restart apache2
- systemctl stop apache2
- systemctl reload apache2
 - Cette fonctionnalité permet de recharger les fichiers de conf d'apache sans arrêter les sites (utile en cas de serveur mutualisé)



Outils

service apache2 xxxx permet de relancer, arreter, stopper
=> s'utilise avec sudo

- service apache2 restart
- service apache2 stop
- service apache2 reload
 - Cette fonctionnalité permet de recharger les fichiers de conf d'apache sans arrêter les sites (utile en cas de serveur mutualisé)



Outils

apachectl xxx permet de relancer, arrêter, stopper apache
=> s'utilise avec sudo

- `apachectl start`
- `apachectl restart`
- `apachectl stop`
- `apachectl configtest` (vérifie la config)
- `apachectl -h` => pour l'aide



Outils

a2enmode xxxx permet de charger et rendre utilisable un module

- Les modules se trouvent sous mods-available
- Avec la commande => un lien symbolique est réalisé sous mods-enabled



Outils

a2enconf xxxx permet de charger et rendre utilisable un fichier de configuration

- Les fichiers de configurations se trouvent sous conf-available
- Avec la commande => un lien symbolique est réalisé sous conf-enabled





Outils

a2ensite xxxx permet de charger et rendre utilisable un virtualhost

- Les fichiers de virtualhost se trouvent sous sites-available
- Avec la commande => un lien symbolique est réalisé sous sites-enabled



Directives principales

La configuration de Apache utilise des directives (de configuration)

- <https://httpd.apache.org/docs/2.4/sections.html>
- Directives évaluées au démarrage
- Directives évaluées à chaque requête
- Dans les fichiers de configuration
- Dans des fichiers .htaccess
- Les directives ont toujours une balise d'entrée et une balise de sortie



Directives évaluées au démarrage

<IfDefine>

- Le conteneur <IfDefine> contient des directives qui ne seront appliquées que si un paramètre approprié a été défini dans la ligne de commande de httpd.

```
<IfDefine ClosedForNow>  
    Redirect "/" "http://otherserver.example.com/"  
</IfDefine>
```



Directives évaluées au démarrage

<IfModule>

- Le conteneur <IfModule> contient des directives qui s'appliqueront que si un module particulier est disponible au niveau du serveur.

```
<IfModule mod_mime_magic.c>  
    MimeMagicFile "conf/magic"  
</IfModule>
```



Directives évaluées au démarrage

<IfVersion>

- Le conteneur <IfVersion> des directives qui s'appliqueront que si une version particulière du serveur s'exécute.

```
<IfVersion >= 2.4>  
  # les directives situées ici ne s'appliquent  
  # que si la version  
  # est supérieure ou égale à 2.4.0.  
</IfVersion>
```



Directives évaluées pour chaque requête

- <Directory>
- <DirectoryMatch>
- <Files>
- <FilesMatch>
- <IfModule>
- <Location>
- <LocationMatch>
- <MDomainSet>
- <Proxy>
- <ProxyMatch>
- <VirtualHost>



Directives évaluées pour chaque requête

<Directory>

- Le conteneur <Directory> contient des directives qui s'appliqueront que dans le répertoire indiqué et ses sous-répertoires.

```
<Directory "/usr/local/httpd/htdocs">  
    Options Indexes FollowSymLinks  
</Directory>
```



Directives évaluées pour chaque requête

<Directory>

- Si plusieurs conteneurs <Directory>, les directives s'appliqueront des chemins les plus court vers les plus longs.

```
<Directory "/">  
    AllowOverride None  
</Directory>
```

```
<Directory "/home">  
    AllowOverride FileInfo  
</Directory>
```

Directives évaluées pour chaque requête

<DirectoryMatch>

- Le conteneur <DirectoryMatch> contient des directives qui s'appliquent au contenu de répertoires du système de fichiers correspondant à une expression rationnelle.

```
<DirectoryMatch "^/www/ (.+/?)[0-9]{3}/">  
    # ...  
</DirectoryMatch>
```



Directives évaluées pour chaque requête

<Files>

- Le conteneur <Files> contient des directives qui s'appliquent aux fichiers indiqués.

```
<Files "?at.*">  
    # Les directives insérées ici s'appliqueront aux  
    fichiers  
    # cat.html, bat.html, hat.php, et ainsi de  
    suite.  
</Files>
```



Directives évaluées pour chaque requête

<Location>

- Le conteneur <Location> limite la portée des directives contenues aux URLs définies par l'URL.

```
<Location "/status">  
    SetHandler server-status  
    Require host example.com  
</Location>
```



Directives évaluées pour chaque requête

<IfModule>

- Le conteneur <IfModule> contient des directives qui s'exécutent si un module est actif

```
<IfModule mod_rewrite.c>  
    Options -MultiViews  
    RewriteEngine On  
</IfModule>
```



Directives évaluées pour chaque requête

<VirtualHost>

- Le conteneur <VirtualHost> contient des directives qui s'appliquent qu'à un serveur virtuel particulier (serveur mutualisé).

```
<VirtualHost *:80>  
    ServerAdmin webmaster@host.example.com  
    DocumentRoot "/www/docs/host.example.com"  
    ServerName host.example.com  
</VirtualHost>
```





Fonctionnalités

En plus des directives, apache comprend des fonctionnalités qui vont spécifier des comportements et s'utilisent à l'intérieur des directives.

<https://httpd.apache.org/docs/2.4/mod/core.html>



Fonctionnalités

AllowOverride

- Cette information indique les directives autorisées dans les fichiers .htaccess.
 - AllowOverride None => les directives dans .htaccess ne sont pas autorisées
 - AllowOverride All => toutes les directives dans .htaccess sont autorisées



Fonctionnalités

Require

- Cette information permet de vérifier si un utilisateur authentifié a l'autorisation d'accès.
 - Require all granted => L'accès est autorisé sans restriction
 - Require all denied => L'accès est systématiquement refusé.
 - Require ip 10 172.20 192.168.2
 - Require user « identifiant utilisateur »



Fonctionnalités

Options

- Cette information permet d'ajouter des options spécifique à un répertoire
 - Options FollowSymLinks => suit les liens symboliques
 - Options Indexes => utilise les fichiers index quand pas spécifiés dans l'URL (index.html, index.php, ...).



Fonctionnalités

ErrorLog

- Cette information permet de spécifier le chemin du journal d'erreur

```
ErrorLog "/var/log/httpd/error_log"
```

```
ErrorLog ${APACHE_LOG_DIR}/error-mon-site.log
```



Utilisation

Les virtualhost permettent de gérer plusieurs sites web sur le même serveur (serveur mutualisé).

Les directives peuvent être dans un seul fichier ou dans plusieurs fichiers (solution préférable).

<https://httpd.apache.org/docs/2.4/vhosts>



Utilisation

Les virtualhost permettent de gérer plusieurs sites web :

- Avec une @IP par vhost => le serveur doit disposer de plusieurs @IP
- Avec une seule @IP mais plusieurs noms de domaine (utilisation plus courante)



Vhost par nom

La directive VirtualHost est associée à une adresse IP et un port.

<VirtualHost *:80> => écoute n'importe quelle adresse du serveur sur le port 80

<VirtualHost 192.168.25.42:80> => écoute uniquement l'adresse spécifiée sur le port 80

<VirtualHost *:1080> => écoute n'importe quelle adresse du serveur sur le port 1080



Directives

ServerName (obligatoire)

- Le ServerName correspond à l'url définie au niveau du DNS et qui se trouve dans la requête HTTP.

```
<VirtualHost *:80>  
    ServerName www.example.com  
</VirtualHost>
```



Directives

ServerAlias

- Le ServerAlias permet d'accéder au serveur avec un nom de serveur différents

```
<VirtualHost *:80>  
    ServerName www.example.com  
    ServerAlias www2.example.com  
</VirtualHost>
```



Directives

DocumentRoot

- DocumentRoot détermine la racine du site (permet de structurer les arborescences quand plusieurs sites)

```
<VirtualHost *:80>  
    ServerName www.example.com  
    ServerAlias www2.example.com  
    DocumentRoot "/var/web"  
</VirtualHost>
```



Directives

Directory

```
<VirtualHost *:80>
    ServerName www.example.com
    ServerAlias www2.example.com
    DocumentRoot /var/web
    <Directory /var/web/test>
        AllowOverride None
    </Directory>
</VirtualHost>
```



Les log

- La gestion des log par VirtualHost permet
 - de spécifier les fichiers à utiliser
 - de définir le niveau de log à utiliser
 - de définir un répertoire spécifique



Les log

```
<VirtualHost *:80>
    ServerName www.example.com
    ServerAlias www2.example.com
    DocumentRoot "/var/web"
    LogLevel info
    ErrorLog ${APACHE_LOG_DIR}/error-api-aurion.log
    CustomLog ${APACHE_LOG_DIR}/access-api-aurion.log
    combined
</VirtualHost>
```





.htaccess

<http://httpd.apache.org/docs/current/howto/htaccess.html>

Apache permet de délocaliser la gestion de la configuration, au moyen de fichiers spéciaux appelés par défaut .htaccess

Chaque fichier .htaccess est placé directement dans le répertoire dont il doit gérer la configuration particulière et éventuellement protéger l'accès.



.htaccess

Syntaxe => identique aux fichiers conf de Apache (VirtualHost)

Fonctionnement => lu dynamiquement à chaque requête
=> problèmes de performances

Effet immédiat => pas de redémarrage de Apache



.htaccess

Utilisation

Pour être avec la directive utilisé , le fichiers .htaccess doivent être autorisé dans le fichier de conf avec la directive AllowOverride

AllowOverride None => pas d'utilisation de .htaccess

AllowOverride All => utilisation de .htaccess

