

Cours découverte métier DevOps

Séquence 6 : Notion de client/serveur

1. Qu'est-ce qu'un serveur ?

On appelle logiciel serveur un programme qui offre un service sur le réseau. Le serveur accepte des requêtes, les traite et renvoie le résultat au demandeur. Le terme serveur s'applique à la machine sur laquelle s'exécute le logiciel serveur.

Pour pouvoir offrir ces services en permanence, le serveur doit être sur un site avec accès permanent et s'exécuter en permanence (daemon).

2. Qu'est-ce qu'un client ?

On appelle logiciel client un programme qui utilise le service offert par un serveur. Le client envoie une requête et reçoit la réponse. Le client peut être raccordé par une liaison temporaire.

3. Qu'appelle-t-on architecture client/serveur ?

C'est la description du fonctionnement coopératif entre le serveur et le client. Les services internet sont conçus selon cette architecture.

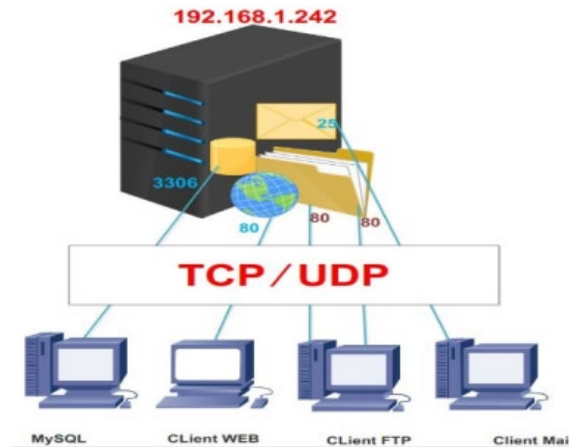
Ainsi, chaque application est composée de logiciel serveur et logiciel client. A un logiciel serveur, peut correspondre plusieurs logiciels clients développés dans différents environnements : Unix, Mac, PC...; la seule obligation est le respect du protocole de transport et le port entre les deux processus communicants.

Comment identifier un service ?

Un serveur peut offrir plusieurs services simultanément. Chacun de ces services est fourni sur un port de communication identifié par un numéro. Ce numéro identifie le service, quel que soit le site.

Exemple :

- le service HTTP est offert sur le port numéro 80,
- FTP offert sur le port 21 (contrôle) et 20 (données),
- TELNET le numéro 23, SSH offert sur le port 22,
- DNS offert sur le port 53, DHCP offert sur le port 67,
- SMTP offert sur le port 25 etc.



Pour accéder donc à un service, il faut trois éléments :

L'adresse IP du serveur, le protocole de transport (TCP/UDP) et le numéro du port du service.

Ce numéro peut être connu par le logiciel client ou peut être récupéré dans le fichier " services " en connaissant le nom du service. Sous UNIX ce fichier se trouve dans **/etc/services**.

4. Le serveur web Apache2

- **Étude de cas 1 : Sites virtuels par adresse**

Un serveur web peut gérer plusieurs sites web. Ces sites web sont appelés serveurs virtuels. Pour créer un site virtuel sous Apache, il faut vérifier que le module "mod_vhost_alias" qui permet de créer un nombre énorme de sites virtuels identifiables par leur document racine, leur adresse IP et leur numéro de port. Dans cette étude de cas nous allons montrer comment créer des serveurs virtuels ou sites virtuels ?

Sous Ubuntu, le fichier de configuration d'apache se trouve dans le répertoire **/etc/apache2.conf** et les fichiers relatifs au serveur virtuel se trouve dans les fichiers **/etc/apache2/sites-available** (dossier qui contient la liste des sites virtuels disponibles) ou **/etc/apache2/sites-enabled** (dossier qui contient la liste des sites virtuels activés). Un site virtuel par adresse, est identifié par une adresse IP. Les sites virtuels sont créés dans les directives **<VirtualHost>** **</VirtualHost>** comme suit :

```
<VirtualHost 192.168.2.122>
    ServerAdmin admin@localhost
    DocumentRoot /var/www/html/monsite
</VirtualHost>
```

Puis on active le site en utilisant l'outil **a2ensite monsite**.

```

root@latyr-ec2lt:/var/www/html# vim /etc/apache2/sites-available/monsite.conf
root@latyr-ec2lt:/var/www/html# a2ensite monsite.conf
Enabling site monsite.
To activate the new configuration, you need to run:
    systemctl reload apache2
root@latyr-ec2lt:/var/www/html# █

```

Ensuite redémarrer apache2 avec la commande :

service apache2 restart

L'exécution de cette commande # a2ensite monsite.conf crée un lien dans le dossier des sites actifs "/etc/apache2/sites-enable. Il faut ensuite créer un contenu de la page dans le dossier racine du site indiqué par le paramètre **DocumentRoot** /var/www/html/monsite :

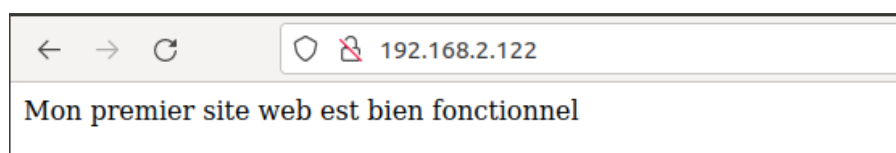
```

root@latyr-ec2lt:/var/www/html# vim monsite/index.html
root@latyr-ec2lt:/var/www/html# █

```

Le client web : le navigateur

On peut ouvrir un navigateur pour pouvoir visualiser le résultat du fichier index.html en utilisant le lien : <http://192.168.2.122>



- **Etude de cas 2 : Sites virtuels par nom**

Un serveur web peut héberger plusieurs serveurs virtuels (module mod_vhost_alias). Pour cette étude de cas, nous allons créer deux serveurs virtuels basés sur le nom respectivement monsite1.ec2lt.sn et monsite2.ec2lt.sn pointant sur une même adresse IP. Le principe de création est le même que pour le précédent. Commençons par créer les dossiers racines des sites monsite1 et monsite2

Contenu du fichier index.html pour monsite1. Nous allons taper un peu de code HTML.

Nous allons enregistrer le fichier sous index.html

```

# mkdir /var/www/html/monsite1
# vim index.html
# mkdir /var/www/html/monsite2
#vim index.html

```

Ensuite, il faut créer le fichier du site dans /etc/apache2/sites-available

```
# cd /etc/apache2/sites-available
# vim monsite1.conf
```

```
<VirtualHost *:80>
    ServerAdmin admin@localhost
    ServerName monsite1.ec2lt.sn
    DocumentRoot /var/www/html/monsite1
</VirtualHost>
```

Extrait du fichier de serveur virtuel **monsite2**

```
# cd /etc/apache2/sites-available
# vim monsite2.conf
```

```
<VirtualHost *:80>
    ServerAdmin admin@localhost
    ServerName monsite2.ec2lt.sn
    DocumentRoot /var/www/html/monsite2
</VirtualHost>
```

Activons les deux sites par **a2ensite**

```
# a2ensite monsite1.conf
# a2ensite monsite2.conf
# service apache2 restart
```

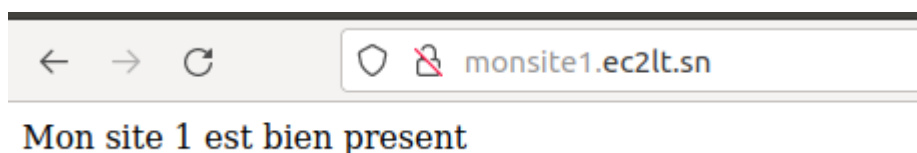
Modifions ensuite le fichier **/etc/hosts** pour pointer l'adresse vers **monsite1.ec2lt.sn** et **monsite2.ec2lt.sn** ou faire un enregistrement dans le fichier de zone du serveur DNS.

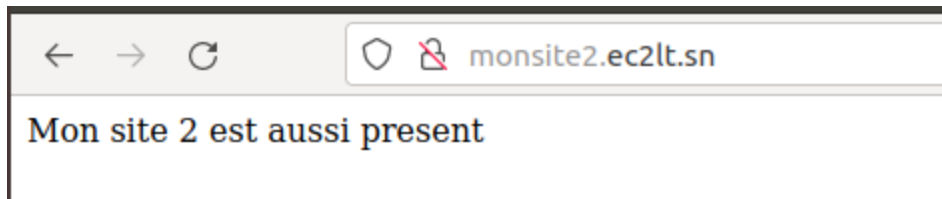
```
27.0.0.1      localhost
127.0.1.1     latyr-ec2lt

192.168.2.122 monsite1.ec2lt.sn
192.168.2.122 monsite2.ec2lt.sn
```

- **Le client web le navigateur**

Puis dans le navigateur **http://monsite1.ec2lt.sn** pour afficher la page d'accueil de **monsite1** et **http://monsite2.ec2lt.sn** pour afficher la page d'accueil de **monsite2**.





- **Etude de cas 3 : Site par dossier**

Un site par dossier n'est pas complexe à déployer. Il suffit de créer un dossier qui contiendra toutes les pages web dans le répertoire racine d'apache (**/var/www/html**). Nous allons créer un dossier appelé « **ec2lt** » auquel nous allons donner les droits 755. Dans ce dossier nous allons créer un fichier **index.html** dans lequel nous allons mettre un petit script HTML comme suit. Dans le navigateur **http://192.168.2.122/ec2lt** le contenu de la page ci-dessus doit apparaître dans la page d'accueil. De ce fait, on peut copier le dossier d'un site et placer dans le répertoire racine d'apache.

```
root@latyr-ec2lt:/var/www/html# mkdir ec2lt
root@latyr-ec2lt:/var/www/html# vim ec2lt/index.html
root@latyr-ec2lt:/var/www/html# chmod -R +x ec2lt
root@latyr-ec2lt:/var/www/html#
```

Le client web : le navigateur

Puis dans le navigateur **http://192.168.2.122/ec2lt** pour afficher la page d'accueil du site.

