

CONFIGURATION DES SERVERS : DHCP, DNS ET WEB APPACHE

I. CONFIGURATION D'UN SERVER DHCP

II. CONFIGURATION D'UN SERVER DNS

III. CONFIGURATION D'UN SERVER WEB APPACHE

Réalisé par :

OYE ESSENE Trésor



CONFIGURATION D'UN SERVER DHCP SOUS LINUX

INTRODUCTION

Le rapport actuel examine la mise en place d'un serveur DHCP sur un système Linux. Il utilise le protocole DHCP pour automatiser l'assignation des adresses IP dans un réseau. L'objectif est de simplifier la gestion des adresses IP, de minimiser les erreurs humaines et d'optimiser les performances du réseau

POINTS IMPORTANTS

- Simplifier la gestion des adresses IP.
- Réduire les erreurs humaines et à améliorer l'efficacité du réseau.

Fondements Théoriques

Dynamic Host Configuration Protocol (DHCP)

Le DHCP est un protocole réseau qui offre aux appareils la possibilité de recevoir automatiquement une configuration IP dès leur connexion au réseau. Son fonctionnement repose sur l'allocation dynamique d'adresses IP ainsi que d'autres paramètres de configuration réseau, comme la passerelle par défaut et les serveurs DNS.

Avantages du DHCP

- Automatisation : Supprime le besoin de configurer individuellement chaque appareil avec une adresse IP de manière manuelle.
- Gestion Centralisée : Facilite une administration centralisée des adresses IP dans tout le réseau.
- Réduction des Conflits : Diminue les risques de conflits liés aux adresses IP.

Objectifs de Configuration

Le but de cette configuration est de déployer un serveur DHCP afin d'assigner automatiquement des adresses IP aux clients du réseau. Les objectifs spécifiques incluent la délimitation d'une plage d'adresses IP, la configuration des paramètres réseau, et garantir une gestion efficace des adresses allouées.

Logiciels Utilisés

ISC DHCP Server a été sélectionné en raison de sa réputation de fiabilité et de flexibilité. Il est largement adopté dans les environnements Linux et offre une configuration solide du serveur DHCP.

Configuration du Serveur DHCP

• Installation du Serveur DHCP

Le processus d'installation du serveur DHCP sous Linux a été effectué en utilisant les commandes standards du gestionnaire de paquets.

```
client@client:~$ su root
Password:
root@client:/home/client# apt install isc-dhcp-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
isc-dhcp-server is already the newest version (4.4.1-2.1ubuntu5.20.04.5).
0 upgraded, 0 newly installed, 0 to remove and 290 not upgraded.
root@client:/home/client#
```

• Configuration du Fichier dhcpd.conf

Le fichier de configuration principal, /etc/dhcp/dhcpd.conf, a été modifié pour définir le sous-réseau, la plage d'adresses, et d'autres paramètres. Cela garantit une attribution cohérente des adresses IP aux clients.

```
GNU nano 4.8 /etc/dhcp/dhcpd.conf
# dhcpd.conf
#
# Sample configuration file for ISC dhcpd
#
# Attention: If /etc/ltsp/dhcpd.conf exists, that will be used as
# configuration file instead of this file.
#
# option definitions common to all supported networks...
subnet 192.168.20.0 netmask 255.255.255.0 {
    range 192.168.20.10 192.168.20.100;
    option domain-name "tresor.sn";
    option domain-name-servers 192.168.20.1;
    option routers 192.168.20.254;

    default-lease-time 86400;
    max-lease-time 172800;
}
```

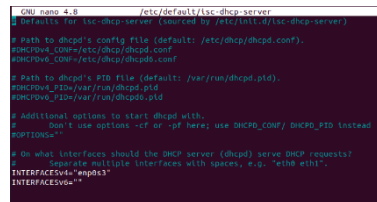
Ci-dessus , la configurer du ous-réseau et les paramètres de location

Nous avons ajouté des directives de configuration pour définir les paramètres du réseau, y compris la plage d'adresses IP à attribuer et d'autres options, afin d'assurer une configuration précise et fonctionnelle du réseau.

1. subnet: Spécifie le sous-réseau et le masque de sous-réseau.
2. range: Définit la plage d'adresses IP à attribuer aux clients DHCP.
3. option routers: Indique la passerelle par défaut.
4. option domain-name-servers: Spécifie les serveurs DNS à utiliser.
5. default-lease-time et max-lease-time: Définissent la durée de bail par défaut et maximale.

- **Configurer l'interface réseau :**

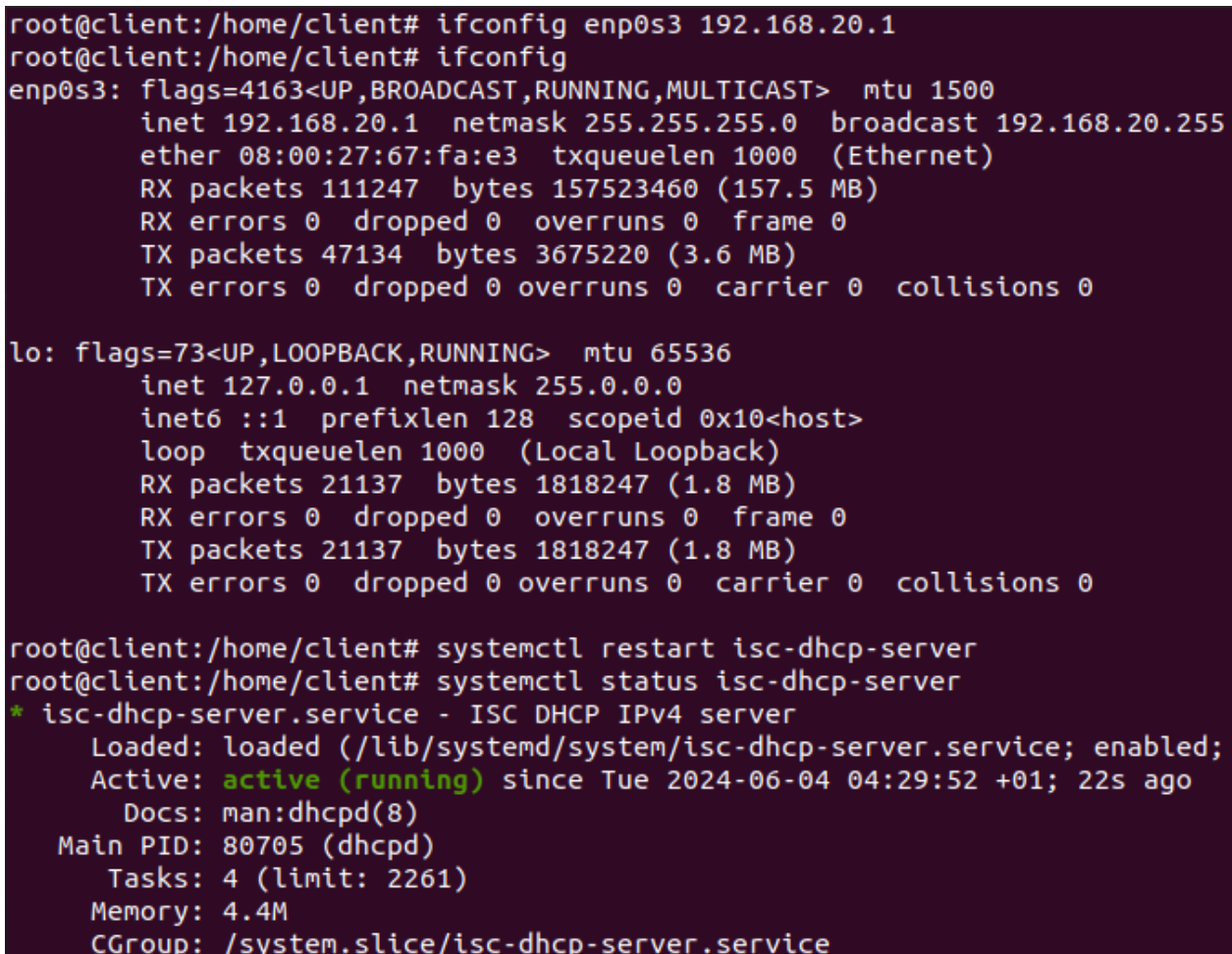
Nous avons ajouté la ligne suivante (voir image) au fichier `/etc/default/isc-dhcp-server` pour indiquer à DHCP sur quelle interface réseau il doit écouter. En spécifiant cette interface, nous définissons à quel réseau local DHCP doit distribuer les adresses IP. Cela est particulièrement utile dans les environnements où plusieurs interfaces réseau sont disponibles, car cela permet de préciser sur quelle interface le service DHCP doit fonctionner.



```
GNU nano 4.8 /etc/default/isc-dhcp-server
# Defaults for isc-dhcp-server (provided by default/isc-dhcp-server)
#
# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
DHCPD_CONF=/etc/dhcp/dhcpd.conf
# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
DHCPD_PID=/var/run/dhcpd.pid
# Additional options to start dhcpd with.
# Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
# if you need them.
#
# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
# Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACESv4="enp0s3"
INTERFACESv6=""
```

- **Redémarrer le serveur DHCP :**

Après avoir effectué les modifications, nous avons redémarré le service DHCP pour appliquer les changements.



```
root@client:/home/client# ifconfig enp0s3 192.168.20.1
root@client:/home/client# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.20.1  netmask 255.255.255.0  broadcast 192.168.20.255
    ether 08:00:27:67:fa:e3  txqueuelen 1000  (Ethernet)
    RX packets 111247  bytes 157523460 (157.5 MB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 47134  bytes 3675220 (3.6 MB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 21137  bytes 1818247 (1.8 MB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 21137  bytes 1818247 (1.8 MB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

root@client:/home/client# systemctl restart isc-dhcp-server
root@client:/home/client# systemctl status isc-dhcp-server
* isc-dhcp-server.service - ISC DHCP IPv4 server
   Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2024-06-04 04:29:52 +01; 22s ago
     Docs: man:dhcpd(8)
   Main PID: 80705 (dhcpd)
    Tasks: 4 (limit: 2261)
   Memory: 4.4M
   CGroup: /system.slice/isc-dhcp-server.service
```

Tests de Validation

Des tests approfondis ont été réalisés pour valider la configuration, notamment des simulations de connexion de clients, des vérifications d'allocation d'adresses IP, et des analyses de la stabilité du service. Ce test ci-dessous a été effectué sous une machine cliente Windows 7 Pro.

```
C:\Windows\system32\cmd.exe
Connection-specific DNS Suffix . : stevy.sn

C:\Users\stevy>ipconfig /all

Windows IP Configuration

Host Name . . . . . : stevy-PC
Primary Dns Suffix . . . . . :
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
DNS Suffix Search List. . . . . : stevy.sn

Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix . : stevy.sn
Description . . . . . : Intel(R) PRO/1000 MT Desktop Adapter
Physical Address. . . . . : 08-00-27-04-3F-05
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::b94e:deb4:a164:76d2%11(Preferred)
IPv4 Address. . . . . : 192.168.20.10(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : Sunday, May 26, 2024 1:27:46 AM
Lease Expires . . . . . : Monday, May 27, 2024 1:27:45 AM
Default Gateway . . . . . : 192.168.20.254
DHCP Server . . . . . : 192.168.20.1
DHCPv6 Iaid . . . . . : 235405351
DHCPv6 Client DUID. . . . . : 00-01-00-01-2D-E4-97-18-08-00-27-04-3F-05

DNS Servers . . . . . : 192.168.20.1
NetBIOS over Tcpip. . . . . : Enabled

Tunnel adapter isatap.stevy.sn:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . : stevy.sn
Description . . . . . : Microsoft ISATAP Adapter
Physical Address. . . . . : 00-00-00-00-00-00-00-E0
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes
```

Problèmes Rencontrés et Solutions

Un problème initial lié à la résolution DNS a été identifié et résolu en ajustant les serveurs DNS spécifiés dans la configuration DHCP.

CONCLUSION

La configuration du serveur DHCP a été menée avec succès, démontrant l'efficacité du protocole DHCP dans la gestion des adresses IP. Cette approche offre une solution automatisée et scalable pour les réseaux, améliorant la flexibilité et la facilité de gestion.



CONFIGURATION D'UN SERVER DNS LINUX

INTRODUCTION

Ce rapport examine la mise en place d'un serveur DNS sur un système Linux. le serveur DNS (Domain Name System) sert principalement à traduire les noms de domaine en adresses IP. L'objectif est de simplifier l'accès aux ressources en ligne, de minimiser les erreurs humaines liées à la mémorisation des adresses IP et Cette traduction est nécessaire pour que les utilisateurs puissent accéder aux ressources sur Internet en utilisant des noms faciles à retenir au lieu de se rappeler des adresses IP numériques..

POINTS IMPORTANTS

- Installer et configurer un serveur DNS sous Linux (BIND).
- Configurer des zones de recherche directe et inverse pour le domaine stevy.sn
- Tester la résolution de noms

Fondements Théoriques

Domain Name System (DNS)

Le DNS (Domain Name System) est un protocole réseau essentiel qui permet de traduire les noms de domaine en adresses IP, facilitant ainsi l'accès aux ressources en ligne. En utilisant des serveurs DNS, les utilisateurs peuvent saisir des noms de domaine lisibles par l'humain, tels que `www.stevy.sn`, et le DNS les convertit en adresses IP numériques nécessaires pour localiser et accéder aux sites web et autres services sur Internet.

Avantages du DNS

- **Simplification de l'accès:** Permet aux utilisateurs de se connecter à des sites web et des services en utilisant des noms de domaine simples à mémoriser, au lieu de longues adresses IP.
- **Réduction des erreurs humaines :** Diminue les erreurs liées à la mémorisation et à la saisie des adresses IP, car les utilisateurs se servent de noms de domaine.
- **Optimisation de la navigation :** Améliore l'efficacité de la navigation sur le réseau en facilitant la résolution rapide des noms de domaine.
- **Redondance et fiabilité :** Les serveurs DNS utilisent des mécanismes de redondance et de mise en cache pour assurer un accès continu et fiable aux ressources, même en cas de panne de certains serveurs DNS.
- **Scalabilité:** Le système DNS est distribué et hiérarchique, permettant une scalabilité efficace à mesure que le nombre de domaines et d'utilisateurs augmente.

Objectifs de Configuration

Le but de cette configuration est de déployer un serveur DNS afin de traduire automatiquement les noms de domaine en adresses IP pour les clients du réseau

Logiciels Utilisés

Bind9 (Berkeley Internet Name Domain version 9) a été sélectionné en raison de sa réputation de fiabilité et de flexibilité. Il est largement adopté dans les environnements Linux et offre une configuration solide du serveur DNS.

Configuration du Serveur DNS

- **Installation du Serveur DNS**

Le processus d'installation du serveur DNS sous Linux a été effectué en utilisant les commandes standards du gestionnaire de paquets.

Sur Ubuntu : **sudo apt-get install bind9**

```
root@client:/home/client# apt install bind9
Reading package lists... Done
Building dependency tree
Reading state information... Done
bind9 is already the newest version (1:9.16.48-0ubuntu0.20.04.1).
0 upgraded, 0 newly installed, 0 to remove and 290 not upgraded.
```

- **Configuration des zones :**

Le fichier de configuration principal, Ouvrir le fichier de configuration principal de BIND, généralement situé à /etc/bind/named.conf.local.

```
GNU nano 4.8 /etc/bind/named.conf.local
//
// Do any local configuration here
//
//zone direct
    zone "tresor.sn" IN {
        type master;
        file "/etc/bind/direct";
    };
//zone indirect
    zone "20.168.192.in-addr.arpa" IN {
        type master;
        file "/etc/bind/inverse";
    };

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
```

- **Configurer zone direct :**

Nous avons ajouté des directives de configuration pour définir les paramètres du réseau direct dans via la commande :

```
root@client:/home/client# cd /etc/bind
root@client:/etc/bind# nano direct
```

```
GNU nano 4.8                                direct
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      client.tresor.sn. root.client.tresor.sn. (
                                2          ; Serial
                                604800     ; Refresh
                                86400      ; Retry
                                2419200    ; Expire
                                604800 )   ; Negative Cache TTL
;
@         IN      NS       client.tresor.sn.
client   IN      A        192.168.20.1
www      IN      CNAME     client.tresor.sn.
```

Configurer zone inverse :

Nous avons ajouté des directives de configuration pour définir les paramètres du réseau direct dans via la commande :

```
root@AIMERY-STEVEY:/etc/bind# cp db.local inverse
root@AIMERY-STEVEY:/etc/bind# nano inverse
```

```
GNU nano 4.8                                inverse
;
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      client.tresor.sn. root.client.tresor.sn. (
                                2          ; Serial
                                604800     ; Refresh
                                86400      ; Retry
                                2419200    ; Expire
                                604800 )   ; Negative Cache TTL
;
@         IN      NS       client.tresor.sn.
client   IN      A        192.168.20.1
1        IN      PTR       client
```

- **Configuration du fichier resolv**

Ouvrir le fichier de zone /etc/resolv.conf.

- Ajoutez les modifications suivantes :

```

GNU nano 4.8 /etc/resolv.conf Modified
This file is managed by man:systemd-resolved(8). Do not edit.

This is a dynamic resolv.conf file for connecting local clients to the
internal DNS stub resolver of systemd-resolved. This file lists all
configured search domains.

Run "resolvectl status" to see details about the uplink DNS servers
currently in use.

Third party programs must not access this file directly, but only through the
symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a different way,
replace this symlink by a static file or a different symlink.

See man:systemd-resolved.service(8) for details about the supported modes of
operation for /etc/resolv.conf.
nameserver 192.168.20.1
search tresor.sn
options edns0 trust-ad

```

- **Redémarrer et status du service DNS :**

Après avoir effectué les modifications, nous avons redémarré le service DNS pour appliquer les changements.

```

root@client:/etc/bind# nano /etc/resolv.conf
root@client:/etc/bind# systemctl status bind9
* named.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/named.service; enabled; vendor preset:
   Active: active (running) since Tue 2024-06-04 04:48:11 +01; 15min ago
     Docs: man:named(8)
    Main PID: 80962 (named)
      Tasks: 6 (limit: 2261)
     Memory: 13.3M
    CGroup: /system.slice/named.service
            └─80962 /usr/sbin/named -f -u bind

<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>
<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>
<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>
<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>
<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>
<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>
<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>
<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>
<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>
<8A><88><86><8A><88> 04 05:02:12 client named[80962]: network unreachable>

```

Tests de Validation

Des tests approfondis ont été réalisés pour valider la configuration, notamment l'utilisation de la commande **nslookup** :

```

root@client:/home/client# nslookup www.tresor.sn
Server:          192.168.20.1
Address:         192.168.20.1#53

www.tresor.sn    canonical name = client.tresor.sn.
Name:   client.tresor.sn
Address: 192.168.20.1

```

CONCLUSION

La configuration du serveur DNS a été menée avec succès, démontrant l'efficacité du protocole DNS dans la conversion des adresses IP en nom de domaines et vice versa. Cefle approche offre une solution automatisée et scalable pour les réseaux, améliorant la flexibilité et la facilité pour les utilisateurs d'internet.



CONFIGURATION WEB
APACHE HTTP, PHP, MYSQL,
PHPMYADMIN SOUS LINUX



Ce rapport présente les étapes nécessaires pour configurer un serveur web sous Linux, utilisant Apache HTTP Server, PHP, MySQL, et PhpMyAdmin. Ces technologies sont couramment utilisées pour héberger des applications web dynamiques.

POINTS IMPORTANTS

- Installer et configurer Apache httpd.
- Installer et configurer PHP, MySQL & PhpMyAdmin
- Tester le fonctionnement du site web

Installation des composants

Mise à jour du système

Avant d'installer de nouveaux logiciels, mettons à jour notre système pour assurer que tous les paquets sont à jour : **sudo apt upgrade -y**

```
aimery@AIMERY-STEVEY:~$ sudo apt update
Hit:1 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:2 http://ma.archive.ubuntu.com/ubuntu focal InRelease
Hit:3 http://ma.archive.ubuntu.com/ubuntu focal-backports InRelease
Get:4 http://ma.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Fetched 114 kB in 1s (85.9 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
67 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

Installation d'Apache HTTP Server, php, mysql, phpmyadmin

```
aimery@AIMERY-STEVEY:~$ sudo apt install apache2 php mariadb-server phpmyadmin
[sudo] password for aimery:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils dbconfig-common dbconfig-mysql
  galera-3 gawk icc-profiles-free javascript-common libaio1
  libapache2-mod-php7.4 libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl
  libdbd-mysql-perl libdbi-perl libfcgi-perl libhtml-template-perl
  libjs-jquery libjs-openlayers libjs-sphinxdoc libjs-underscore liblua5.2-0
  libonig5 libreadline5 libsigsegv2 libsnappy1v5 libterm-readkey-perl libzip5
  mariadb-client-10.3 mariadb-client-core-10.3 mariadb-common
  mariadb-server-10.3 mariadb-server-core-10.3 php-bz2 php-common php-curl
  php-gd php-google-recaptcha php-mbstring php-mysql
  php-phpmyadmin-motranslator php-phpmyadmin-shapefile
  php-phpmyadmin-sql-parser php-phpseclib php-psr-cache php-psr-container
```

Apache est un serveur web open-source largement utilisé.

PHP est un langage de script côté serveur utilisé pour créer des pages web dynamiques.

MySQL est un système de gestion de base de données relationnelle.

PhpMyAdmin est un outil web pour gérer MySQL.

Status des applications

- Apache :

```
root@client:/home/client# systemctl restart apache2
root@client:/home/client# systemctl status apache2
* apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2024-06-04 05:12:14 +01; 10s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 81635 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 81639 (apache2)
    Tasks: 55 (limit: 2261)
   Memory: 6.9M
   CGroup: /system.slice/apache2.service
           └─81639 /usr/sbin/apache2 -k start
             └─81640 /usr/sbin/apache2 -k start
               └─81641 /usr/sbin/apache2 -k start

04 05:12:13 client systemd[1]: Starting The Apache HTTP Server: [OK]
04 05:12:14 client apachectl[81638]: AH00558: apache2: Syntax error in /etc/httpd/conf/httpd.conf: line 1: Invalid command 'Listen',
04 05:12:14 client systemd[1]: Started The Apache HTTP Server: [OK]
```

- Mysql

```
aimery@AIMERY-STEVEY:/var/www/html/test$ sudo systemctl status mysql
[sudo] password for aimery:
● mariadb.service - MariaDB 10.3.39 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2024-05-26 21:14:04 +01; 1h 6min ago
     Docs: man:mysqld(8)
           https://mariadb.com/kb/en/library/systemd/
   Main PID: 113573 (mysqld)
    Status: "Taking your SQL requests now..."
     Tasks: 30 (limit: 2545)
    Memory: 68.8M
   CGroup: /system.slice/mariadb.service
           └─113573 /usr/sbin/mysqld

May 26 21:14:03 AIMERY-STEVEY systemd[1]: Starting MariaDB 10.3.39 database server: [OK]
May 26 21:14:04 AIMERY-STEVEY systemd[1]: Started MariaDB 10.3.39 database server: [OK]
May 26 21:14:04 AIMERY-STEVEY /etc/mysql/debian-start[113608]: Upgrading MySQL to 10.3.39
May 26 21:14:04 AIMERY-STEVEY /etc/mysql/debian-start[113611]: Looking for 'mysqld' in /usr/sbin/
```

Déploiement d'un site web de démonstration

Création d'un répertoire pour votre site web et configurez Apache pour l'héberger.

```
aimery@AIMERY-STEVEY:/var/www/html$ sudo mkdir test
aimery@AIMERY-STEVEY:/var/www/html$ ls
index.html  test
aimery@AIMERY-STEVEY:/var/www/html$ cd test
```

Créez du fichier pour la page de notre site :


```
alinery@AIMERY-STEVEY:/var/www/html/test$ sudo nano index.php
```

Création de la base de données :

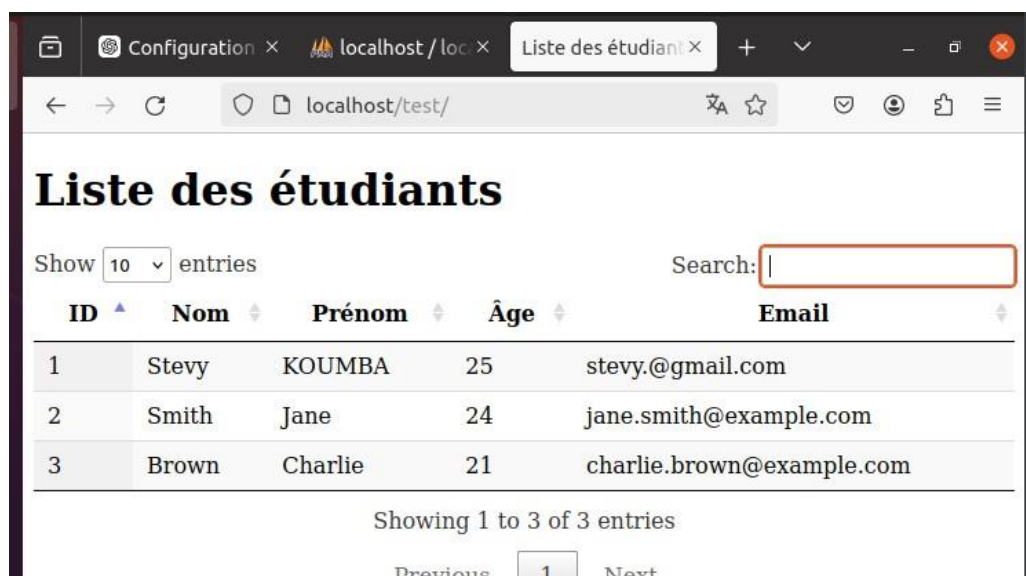
+ Options

				id	nom	prenom	age	email
<input type="checkbox"/>	Edit	Copy	Delete	1	Stevy	KOUMBA	25	stevy@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	2	Smith	Jane	24	jane.smith@example.com
<input type="checkbox"/>	Edit	Copy	Delete	3	Brown	Charlie	21	charlie.brown@example.com

☐ Check all With selected: Edit Copy Delete Export

TEST

Page d'accueil pour du site :



CONCLUSION

Ce rapport a présenté les étapes de base pour configurer un environnement de serveur web utilisant Apache, PHP, MySQL, et PhpMyAdmin sur une machine Linux. Ces outils combinés fournissent une plateforme robuste pour héberger des applications web dynamiques