



## FICHE DE SUIVI

Cette fiche est à rendre (à inclure dans votre rapport de projet) à la fin de la séance de projet encadrée. Elle doit être remplie au fur et à mesure des séances encadrées pour refléter la composition et l'état de présence des membres de votre équipe. Une version .doc est donnée sur Moodle afin de faciliter son édition électronique

## Consignes importantes

Voici quelques consignes à lire attentivement :

1. Le travail de cette SAÉ sera évalué sur la base de **votre rapport de projet à rendre** (sur Moodle lors de la toute dernière séance de projet (qu'elle soit encadrée ou en autonomie). **La date limite est donc soumise à l'emploi du temps et ne pourra être reportée.** Pour l'équité entre groupes, le volume de travail prise en considération est les heures de TP et Projet. Ceci n'empêche pas de travailler en dehors de ces séances mais ce n'est pas obligatoire.
2. La remise de votre rapport de projet (un seul fichier PDF comportant tout votre travail de projet y compris la recherche bibliographique, les configurations, captures d'écrans, et tests techniques, etc.) s'effectue à la dernière séance réglementaire dédié à votre SAÉ.
3. Habituez-vous à bien prendre notre d'une manière progressive non seulement pour les intégrer dans votre rapport de projet mais aussi pour reconfigurer votre machine virtuelle de A à Z si vous devez refaire la configuration depuis une machine virtuelle vierge.
4. Le rapport du projet (fichier PDF) doit inclure la présente fiche de suivi fournie en début du projet qui doit être remplie (état de présence des membres au fil des séances, encadrés "tâches effectuées de chaque séance" et "tâches prévisionnelles pour la séances suivante"
5. Aucun autre mode de remise ne sera accepté, aucun retard ne sera toléré. La date de l'upload/téléversement de votre fichier fait foi.
6. Assurez-vous de bien uploader/téléverser votre compte rendu dans la section de votre groupe TP (et non pas dans une section qui concerne un autre groupe)
7. Assurez-vous de noter les noms de familles de tous les membres de chaque groupe d'étudiants dans le nom du fichier téléversé et dans la page de garde de votre rapport de projet. Attention, tous les membres sont responsable pour s'assurer que le groupe a uploadé son rapport une seul fois sur Moodle.
8. Les rapports de projets (contenu et forme) seront soumis à un traitement anti-plagiat en utilisant les outils automatisés de l'université
9. Le non respects de ces consignes peut donner lieu à des pénalisations lors de votre évaluation de SAÉ.

## Déroulement et timing

Le timing accordé pour cette mission est de 2h de TD, 2h de TP, 7h de projet. Ces séances sont encadrées par un enseignant mais vous avez également 8h de projet en autonomie pour mener à bien la réalisation du projet.

La première séance de TD sera consacrée à

1. L'explication du cahier de charge avec quelques notions techniques
2. La mise en place de votre propre organisation pour mener à bien votre projet.

La première séance de TP sera consacrée à la familiarisation de votre environnement pratique (avec la machine virtuelle consacrée à la SAÉ) et le test des notions du projet

expliquées en TD. Les séances projets suivantes seront consacrées la mise en place du cahier de charge jusqu'à la fin.

Rentabilisez donc bien votre temps car ça va passer très vite ! N'hésitez pas à inclure dans votre rapport de projet **un diagramme de Gantt** pour gérer votre projet et visualiser dans le temps les diverses tâches associées et leurs répartitions entre les membres du projet. L'idéal est d'inclure 2 versions : une version du diagramme prévisionnel (au début du projet), et une version finale (affinée par le vrai déroulement du projet). N'oubliez pas d'ordonner vos tâches par priorité et bien se répartir le travail entre membres. **La rédaction du rapport du projet en fait partie et ne doit pas être négligée.**

## Composition de votre groupe

Nom et prénom	Numéro affecté	Demi Groupe TP
NEVOT Pierre	1	1C1
LE VERGE Lou	2	1C1
CHAPLAIS Ethan	3	1C1
GOGDET Maël	4	1C1

**Nom du Chef du projet** : NEVOT Pierre

**Nom du Chef-adjoint** : LE VERGE Lou

**État de Présence** :

#Etudiant	Séance 1	Séance 2	Séance 3	Séance 4	Séance 5
1	Oui	Oui	Oui	Oui	Oui
2	Oui	Oui	Oui	Oui	Oui
3	Oui	Oui	Oui	Oui	Oui
4	Oui	Oui	Oui	Oui	Oui

#Etudiant	Séance 6	Séance 7	Séance 8	Séance 9	
1	Oui	Oui	Oui	Oui	
2	Oui	Oui	Oui	Oui	
3	Oui	Oui	Oui	Oui	
4	Oui	Oui	Oui	Oui	

## Résumés des avancées de chaque séance TD, TP (encadré) ou Projet (encadré)

### Séance 1 (encadrée)

Résumé des tâches effectuées

NEVOT Pierre (Etudiant #1) :

Prendre connaissance de l'environnement pratique (VM de la SAÉ).

Vérifier le statut d'exécution du service Web (Apache) avec `systemctl status apache2` et noter l'heure exacte du dernier démarrage.

LE VERGE Lou (Etudiant #2) :

Trouver le fichier de configuration contenant la directive `DocumentRoot` et noter sa valeur. Expliquer l'utilité de la directive `DocumentRoot`.

CHAPLAIS Ethan (Etudiant #3) :

Repérer le chemin du binaire du service Web (Apache) en utilisant la commande type `-a apache2`.

GOGDET Maël (Etudiant #4) :

Identifier la version exacte du serveur Apache avec `apache2 -v` et expliquer l'importance de connaître cette version.

#### Résumé des tâches identifiées pour la séance suivante

NEVOT Pierre (Etudiant #1) :

Lister les modules installés nativement avec `apache2 -l` et comprendre le rôle de `mod_log_config.c`.

LE VERGE Lou (Etudiant #2) :

Lister tous les modules (statiques et dynamiques) avec `apache2 -M`.

CHAPLAIS Ethan (Etudiant #3) :

Différencier les commandes `apache2 -version` et `apache2 -v`, ajouter la ligne `ServerName localhost` dans `/etc/apache2/apache2.conf`.

GOGDET Maël (Etudiant #4) :

Tester l'ajout de `ServerName localhost` en redémarrant le serveur Apache.

### Séance 2 (encadrée)

#### Résumé des tâches effectuées

Respect de la prévision :

NEVOT Pierre (Etudiant #1) :

Lister les modules installés nativement avec `apache2 -l` et comprendre le rôle de `mod_log_config.c`.

LE VERGE Lou (Etudiant #2) :

Lister tous les modules (statiques et dynamiques) avec `apache2 -M`.

CHAPLAIS Ethan (Etudiant #3) :

Différencier les commandes `apache2 -version` et `apache2 -v`, ajouter la ligne `ServerName localhost` dans `/etc/apache2/apache2.conf`.

GOGDET Maël (Etudiant #4) :

Tester l'ajout de `ServerName localhost` en redémarrant le serveur Apache.

#### Résumé des tâches identifiées pour la séance suivante

NEVOT Pierre (Etudiant #1) :

Créer un dossier `private` dans `/var/www/html/` et y ajouter un fichier `index.html`.

LE VERGE Lou (Etudiant #2) :

Configurer l'accès restreint au dossier `private` avec authentification (`admin / lannion`).

CHAPLAIS Ethan (Etudiant #3) :

Tester l'accès sécurisé via navigateur et documenter le processus.

GOGDET Maël (Etudiant #4) :

Créer des liens dans la page d'accueil pour accéder à l'espace privé et vice-versa.

### Séance 3 (encadrée)

#### Résumé des tâches effectuées

NEVOT Pierre (Etudiant #1) :

Créer un dossier `private` dans `/var/www/html/` et y ajouter un fichier `index.html`.

LE VERGE Lou (Etudiant #2) :  
Configurer l'accès restreint au dossier private avec authentification (admin / lannion).

CHAPLAIS Ethan (Etudiant #3) :  
Tester l'accès sécurisé via navigateur et documenter le processus.

GOGDET Maël (Etudiant #4) :  
Créer des liens dans la page d'accueil pour accéder à l'espace privé et vice-versa.

#### Résumé des taches identifiées pour la séance suivante

NEVOT Pierre (Etudiant #1) :  
Créer un dossier secret et y ajouter un fichier phpinfo.php pour vérifier la configuration PHP.

LE VERGE Lou (Etudiant #2) :  
Accéder à phpinfo.php via navigateur pour confirmer que PHP est activé.

CHAPLAIS Ethan (Etudiant #3) :  
Vérifier la version PHP, le dossier de configuration et le fichier de configuration PHP.

GOGDET Maël (Etudiant #4) :  
Noter la valeur de l'étiquette short\_open\_tag et déterminer dans quel fichier de configuration elle peut être modifiée.

### **Séance 4 (encadrée)**

#### Résumé des taches effectuées

NEVOT Pierre (Etudiant #1) :  
Créer un dossier secret et y ajouter un fichier phpinfo.php pour vérifier la configuration PHP.

LE VERGE Lou (Etudiant #2) :  
Accéder à phpinfo.php via navigateur pour confirmer que PHP est activé.

CHAPLAIS Ethan (Etudiant #3) :  
Vérifier la version PHP, le dossier de configuration et le fichier de configuration PHP.

GOGDET Maël (Etudiant #4) :  
Noter la valeur de l'étiquette short\_open\_tag et déterminer dans quel fichier de configuration elle peut être modifiée.

#### Résumé des taches identifiées pour la séance suivante

NEVOT Pierre (Etudiant #1) :  
Activer et désactiver le module PHP en ligne de commande, tester les effets sur le serveur.

LE VERGE Lou (Etudiant #2) :  
Vérifier l'accès aux scripts PHP après désactivation et réactivation du module PHP.

CHAPLAIS Ethan (Etudiant #3) :  
Modifier le fichier de configuration pour interpréter les fichiers .html comme des fichiers PHP.

GOGDET Maël (Etudiant #4) :  
Tester l'accès à un fichier PHP sans extension et documenter les résultats.

## Séance 5 (encadrée)

### Résumé des tâches effectuées

NEVOT Pierre (Etudiant #1) :

Activer et désactiver le module PHP en ligne de commande, tester les effets sur le serveur.

LE VERGE Lou (Etudiant #2) :

Vérifier l'accès aux scripts PHP après désactivation et réactivation du module PHP.

CHAPLAIS Ethan (Etudiant #3) :

Modifier le fichier de configuration pour interpréter les fichiers .html comme des fichiers PHP.

GOGDET Maël (Etudiant #4) :

Tester l'accès à un fichier PHP sans extension et documenter les résultats.

### Résumé des tâches identifiées pour la séance suivante

NEVOT Pierre (Etudiant #1) :

Modifier la syntaxe PHP dans une page HTML (mapage.html) pour utiliser les short tags.

LE VERGE Lou (Etudiant #2) :

Tester la page et vérifier la prise en compte des modifications.

CHAPLAIS Ethan (Etudiant #3) :

Activer l'option short\_open\_tag dans le fichier de configuration PHP (php.ini).

GOGDET Maël (Etudiant #4) :

Tester de nouveau la page pour confirmer le bon fonctionnement des short tags.

## Séance 6 (encadrée)

### Résumé des tâches effectuées

NEVOT Pierre (Etudiant #1) :

Modifier la syntaxe PHP dans une page HTML (mapage.html) pour utiliser les short tags.

LE VERGE Lou (Etudiant #2) :

Tester la page et vérifier la prise en compte des modifications.

CHAPLAIS Ethan (Etudiant #3) :

Activer l'option short\_open\_tag dans le fichier de configuration PHP (php.ini).

GOGDET Maël (Etudiant #4) :

Tester de nouveau la page pour confirmer le bon fonctionnement des short tags.

### Résumé des tâches identifiées pour la séance suivante

NEVOT Pierre (Etudiant #1) :

Sécuriser le serveur MySQL avec mysql\_secure\_installation (mot de passe root : lannion).

LE VERGE Lou (Etudiant #2) :

Vérifier le statut d'exécution du service MySQL et noter les informations pertinentes.

CHAPLAIS Ethan (Etudiant #3) :

Identifier la version de MySQL avec `mysql -V`.

GOGDET Maël (Etudiant #4) :

Se connecter au serveur MySQL avec le login root et afficher les bases de données existantes.

## **Séance 7 (encadrée)**

Résumé des tâches effectuées

NEVOT Pierre (Etudiant #1) :

Sécuriser le serveur MySQL avec `mysql_secure_installation` (mot de passe root : lannion).

LE VERGE Lou (Etudiant #2) :

Vérifier le statut d'exécution du service MySQL et noter les informations pertinentes.

CHAPLAIS Ethan (Etudiant #3) :

Identifier la version de MySQL avec `mysql -V`.

GOGDET Maël (Etudiant #4) :

Se connecter au serveur MySQL avec le login root et afficher les bases de données existantes.

Résumé des tâches identifiées pour la séance suivante

NEVOT Pierre (Etudiant #1) :

Sécuriser le serveur MySQL avec `mysql_secure_installation` (mot de passe root : lannion).

Vérifier le statut d'exécution du service MySQL et noter les informations pertinentes.

Identifier la version de MySQL avec `mysql -V`.

Se connecter au serveur MySQL avec le login root et afficher les bases de données existantes.

LE VERGE Lou (Etudiant #2) :

Créer une base de données et une table via MySQL en ligne de commande.

Insérer des enregistrements dans la table, les modifier et les supprimer via des requêtes SQL.

Reproduire ces opérations avec un script PHP.

Tester le script PHP pour accéder aux bases de données et afficher les résultats.

CHAPLAIS Ethan (Etudiant #3) :

Documenter toutes les configurations effectuées et vérifier la cohérence.

Créer des scripts de sauvegarde des configurations et des bases de données.

Réaliser des tests finaux pour s'assurer du bon fonctionnement du serveur, de PHP et de MySQL.

GOGDET Maël (Etudiant #4) :

Insérer dans le code HTML d'une page nommée « `mapage.html` » un code PHP arbitraire.

Tester l'accès à la page « `mapage.html` » pour vérifier le bon fonctionnement de l'interprétation du code PHP dans du HTML.

## **Séance 8 (encadrée)**

Résumé des tâches effectuées

NEVOT Pierre (Etudiant #1) :

Sécuriser le serveur MySQL avec `mysql_secure_installation` (mot de passe root : lannion).

Vérifier le statut d'exécution du service MySQL et noter les informations pertinentes.

Identifier la version de MySQL avec `mysql -V`.

Se connecter au serveur MySQL avec le login root et afficher les bases de données

existantes.

LE VERGE Lou (Etudiant #2) :

Créer une base de données et une table via MySQL en ligne de commande.

Insérer des enregistrements dans la table, les modifier et les supprimer via des requêtes SQL.

Reproduire ces opérations avec un script PHP.

Tester le script PHP pour accéder aux bases de données et afficher les résultats.

CHAPLAIS Ethan (Etudiant #3) :

Documenter toutes les configurations effectuées et vérifier la cohérence.

Créer des scripts de sauvegarde des configurations et des bases de données.

Réaliser des tests finaux pour s'assurer du bon fonctionnement du serveur, de PHP et de MySQL.

GOGDET Maël (Etudiant #4) :

Insérer dans le code HTML d'une page nommée « mapage.html » un code PHP arbitraire.

Tester l'accès à la page « mapage.html » pour vérifier le bon fonctionnement de l'interprétation du code PHP dans du HTML.

Résumé des tâches identifiées pour la séance suivante

NEVOT Pierre (Etudiant #1) :

Préparation du compte rendu final pour avoir une cohérence sur tout le fichier.

Commencement de la partie TD.

LE VERGE Lou et CHAPLAIS Ethan et GOGDET Maël (Etudiant #2, #3 et #4) :

Aide à la préparation du compte rendu final.

## **Séance 9 (encadrée)**

Résumé des tâches effectuées

NEVOT Pierre (Etudiant #1) :

Partie TD commencé mais impossible de continuer sans l'aide d'un professeur.

Compte rendu final écrit et édité.

LE VERGE Lou et CHAPLAIS Ethan et GOGDET Maël (Etudiant #4) :

Aide à la préparation du compte rendu final.

Résumé des tâches identifiées pour la séance suivante

Aucune

# Compte rendu SAE 2.03

---

CHEF DE PROJET : NEVOT Pierre

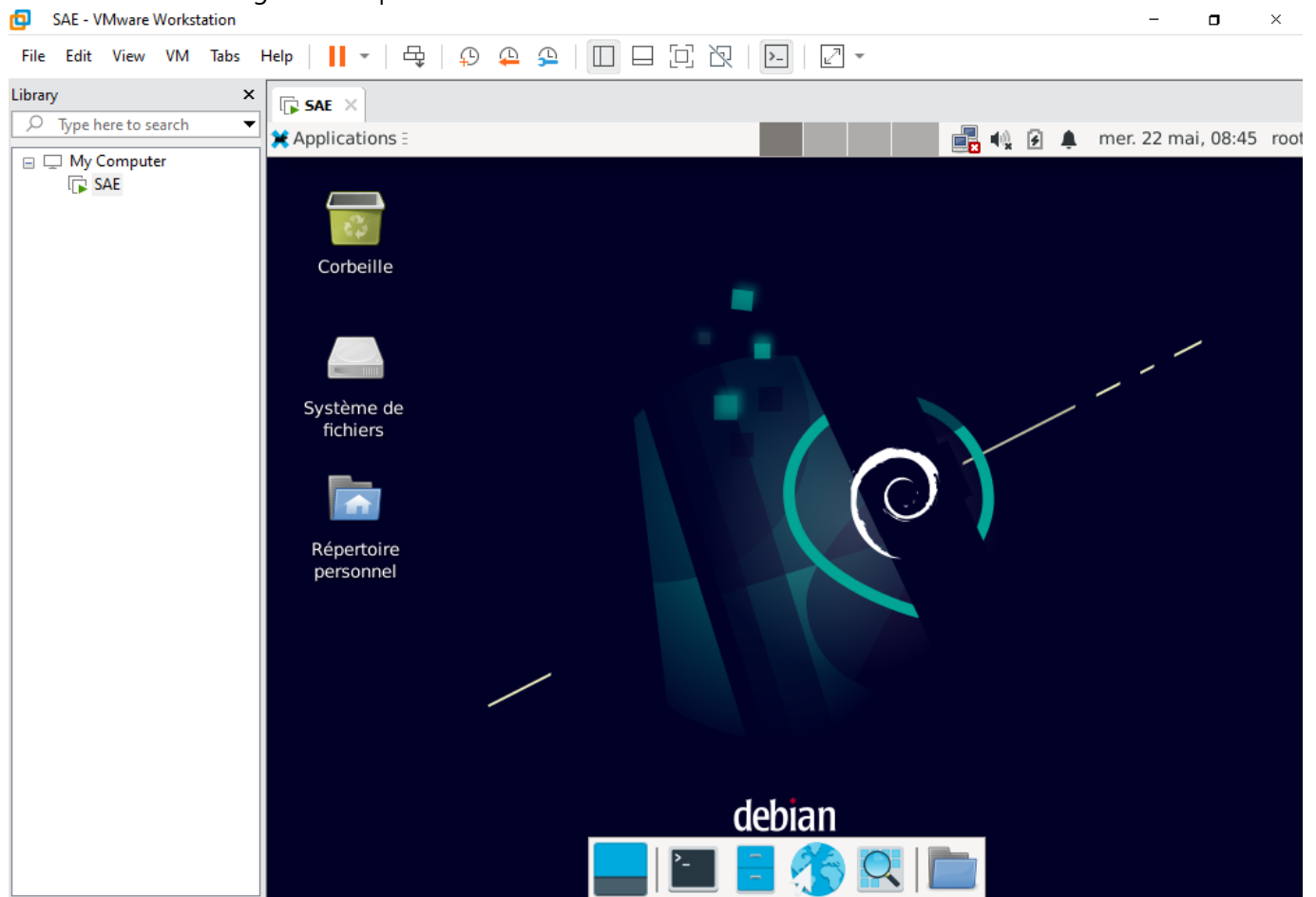
MEMBRES DE L'EQUIPE :

- NEVOT Pierre
- LE VERGE Lou
- CHAPLAIS Ethan
- GOGDET Maël

## Partie TP : Installation d'un service réseau

---

Après avoir copié les fichiers de la SAE 2.03 dans notre disque Z:. On peut lancer la machine virtuelle et se connecter avec les logins : root | lannion.



### Prise en main et le service wen avec Apache2

1. Prenez connaissance de votre environnement pratique (VM de la SAÉ).

J'obtiens l'ip locale de la machine avec la commande `ip a` et je peux voir que l'ip est `127.0.0.1/8` et que l'ipV6 est `::1/128`.



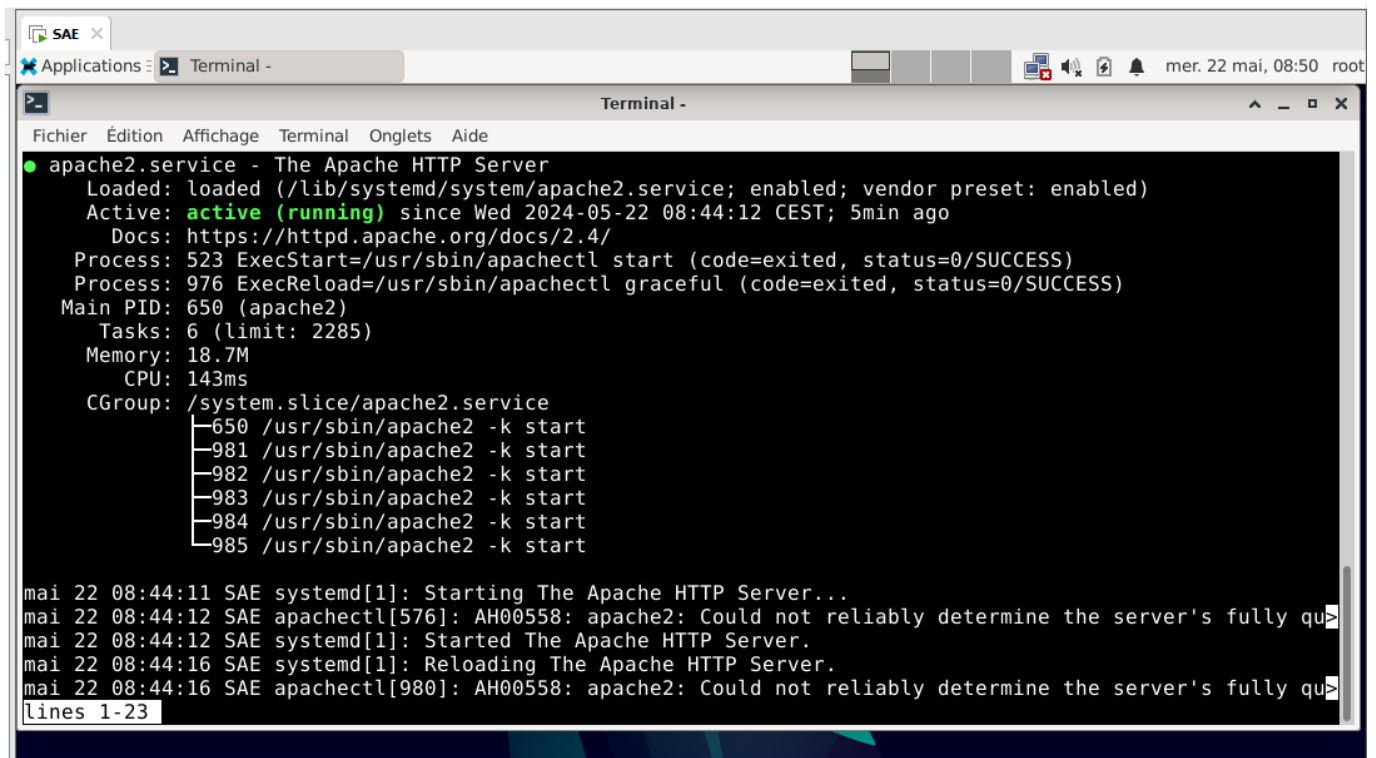
```

root@SAE:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group def
ault qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
root@SAE:~# █

```

2. Vérifiez le statut d'exécution du service Web (assuré par le paquage apache2) avec la commande : `# systemctl status apache2`.

J'exécute la commande `systemctl status apache2` et je peux voir que le service est actif et qu'il s'exécute continuellement.



```

SAE x
Applications Terminal - mer. 22 mai, 08:50 root
Terminal -
Fichier Édition Affichage Terminal Onglets Aide
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-05-22 08:44:12 CEST; 5min ago
     Docs: https://httpd.apache.org/docs/2.4/
  Process: 523 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Process: 976 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/SUCCESS)
 Main PID: 650 (apache2)
    Tasks: 6 (limit: 2285)
   Memory: 18.7M
      CPU: 143ms
   CGroup: /system.slice/apache2.service
           └─650 /usr/sbin/apache2 -k start
             └─981 /usr/sbin/apache2 -k start
               └─982 /usr/sbin/apache2 -k start
                 └─983 /usr/sbin/apache2 -k start
                   └─984 /usr/sbin/apache2 -k start
                     └─985 /usr/sbin/apache2 -k start

mai 22 08:44:11 SAE systemd[1]: Starting The Apache HTTP Server...
mai 22 08:44:12 SAE apachectl[576]: AH00558: apache2: Could not reliably determine the server's fully qu
mai 22 08:44:12 SAE systemd[1]: Started The Apache HTTP Server.
mai 22 08:44:16 SAE systemd[1]: Reloading The Apache HTTP Server.
mai 22 08:44:16 SAE apachectl[980]: AH00558: apache2: Could not reliably determine the server's fully qu
lines 1-23

```

**2.b Relevez les informations pertinentes qui montre qu'il s'exécute continuellement (ce qui est normal car c'est un serveur).**

Le principe d'un service est de s'exécuter continuellement, on peut voir que le service est actif et qu'il est en cours d'exécution.

**2.c Relevez à quelle heure exacte, le serveur a été lancé la toute dernière fois.**

Le serveur a été lancé pour la dernière fois le 22 mai à 8h44min11s. On peut en déduire qu'il a été lancé en même temps que la machine virtuelle. On peut le voir sur la capture d'écran à la ligne :

"mai 22 08:44:11 SAE systemd[1]: Starting The Apache HTTP Server..."

3. Dans quel fichier de configuration de votre serveur Web, se trouve la directive « DocumentRoot » ? Quelle est sa valeur ? Rappelez l'utilité de cette directive?

La directive `DocumentRoot` se trouve dans le fichier de configuration `/etc/apache2/sites-available/000-default.conf`. Sa valeur est `/var/www/html`. Cette directive permet de définir le répertoire racine du serveur web. C'est à dire que c'est le répertoire où le serveur web va chercher les fichiers à envoyer au client. On peut voir entourer en rouge la directive `DocumentRoot` et sa valeur `/var/www/html` sur la capture d'écran ci-dessous. Et aussi le répertoire où se situe la configuration du serveur web.

```
root@SAE:~# cd /etc/apache2/sites-available/
root@SAE:/etc/apache2/sites-available# ls
000-default.conf  default-ssl.conf
root@SAE:/etc/apache2/sites-available# cat 000-default.conf
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port
    that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header t
    # match this virtual host. For the default virtual host (this file) t
    # value is not decisive as it is used as a last resort host regardles
    s.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn
    # error, crit, alert, emerg.
```

Source : <https://httpd.apache.org/docs/2.4/fr/mod/core.html#documentroot>

4. Repérez le chemin du binaire (exécutable) du service Web (apache2) en utilisant la commande « type » et l'option « a »

J'exécute la commande `type -a apache2` et je peux voir que le binaire se trouve dans `/usr/sbin/apache2` et `/sbin/apache2`. On peut le voir sur la capture d'écran ci-dessous.

```
root@SAE:/etc/apache2/sites-available# type -a apache2
apache2 est /usr/sbin/apache2
apache2 est /sbin/apache2
```

5. Identifier la version exacte du serveur Apache. Pourquoi il est important de connaître la version exacte ?

Pour identifier la version du serveur Apache, on peut utiliser la commande `apache2 -v`. On obtient : `Server version: Apache/2.4.53 (Debian) Server built: 2022-03-14T16:28:35`. La version exacte est `2.4.53 (Debian)`. Il est important de connaître la version exacte du serveur Apache pour des raisons de sécurité. En effet, si on connaît la version exacte, on peut savoir si le serveur est à jour et si il est vulnérable à des failles de sécurité.

Source : commande `man apache2`

6. Listez les modules installés nativement (compilé dans le noyau) du le serveur avec la commande « `apache2` » et l'argument de commande « `-l` »

Pour lister les modules installés nativement, on peut utiliser la commande `apache2 -l`. On obtient :

```
Compiled in modules: core.c mod_so.c mod_watchdog.c http_core.c mod_log_config.c mod_logio.c
mod_version.c mod_unixd.c
```

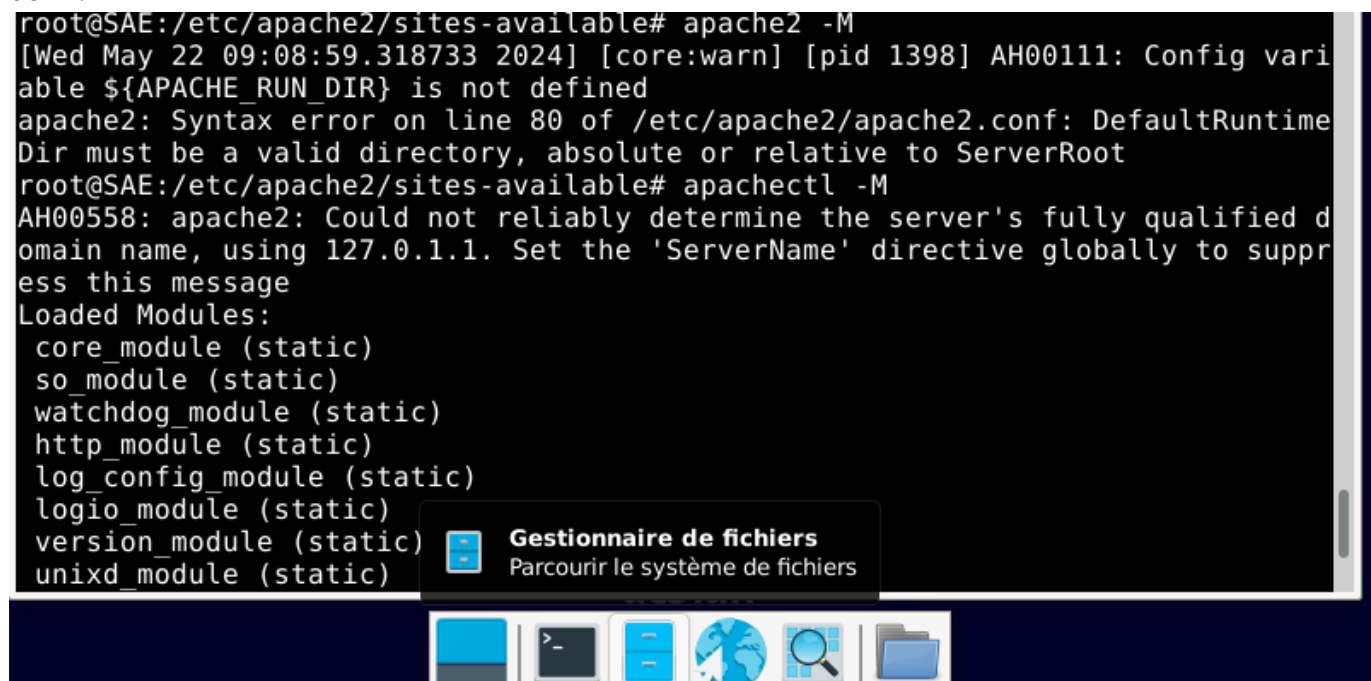
7. D'après vous, quel est le rôle du module « `mod_log_config.c` » ?

Le module `mod_log_config.c` permet de configurer les logs du serveur web. Il permet de définir le format des logs, les informations à logger, le fichier de log, etc. C'est un module très important pour la sécurité et le débogage du serveur web.

Source : [https://httpd.apache.org/docs/2.4/fr/mod/mod\\_log\\_config.html](https://httpd.apache.org/docs/2.4/fr/mod/mod_log_config.html)

8. La commande précédente ne liste pas tous les modules chargés dynamiquement par le serveur Web (comme par exemple ceux chargés avec la directive nommée « `LoadModule` »). Pour lister tous les modules (statique et dynamique), utilisez la commande : `apache2 -M` et relevez les modules affichés

Pour lister tous les modules (statique et dynamique), on peut utiliser la commande `apache2 -M`. On peut voir que j'obtiens une erreur `AH00111: Config variable ${APACHE_RUN_DIR} is not defined` `apache2: Syntax error on line 80 of /etc/apache2/apache2.conf: DefaultRuntimeDir must be a valid directory, absolute or relative to ServerRoot`. Qui veut dire que le répertoire par défaut n'est pas défini.



```
root@SAE:/etc/apache2/sites-available# apache2 -M
[Wed May 22 09:08:59.318733 2024] [core:warn] [pid 1398] AH00111: Config variable ${APACHE_RUN_DIR} is not defined
apache2: Syntax error on line 80 of /etc/apache2/apache2.conf: DefaultRuntimeDir must be a valid directory, absolute or relative to ServerRoot
root@SAE:/etc/apache2/sites-available# apachectl -M
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Loaded Modules:
  core_module (static)
  so_module (static)
  watchdog_module (static)
  http_module (static)
  log_config_module (static)
  logio_module (static)
  version_module (static)
  unixd_module (static)
```

On peut avoir après que j'exécute la commande `apachectl -M` et là j'obtiens la liste des modules chargés de manière statique et dynamique :

```
Loaded Modules: core_module (static) so_module (static) watchdog_module (static) http_module
(logic) log_config_module (static) logio_module (static) version_module (static) unixd_module (static)
access_compat_module (shared) alias_module (shared) auth_basic_module (shared) authn_core_module
```

```
(shared) authn_file_module (shared) authz_core_module (shared) authz_host_module (shared)
authz_user_module (shared) autoindex_module (shared) deflate_module (shared) dir_module (shared)
env_module (shared) filter_module (shared) mime_module (shared) mpm_prefork_module (shared)
negotiation_module (shared) php7_module (shared) reqtimeout_module (shared) setenvif_module
(shared) status_module (shared)
```

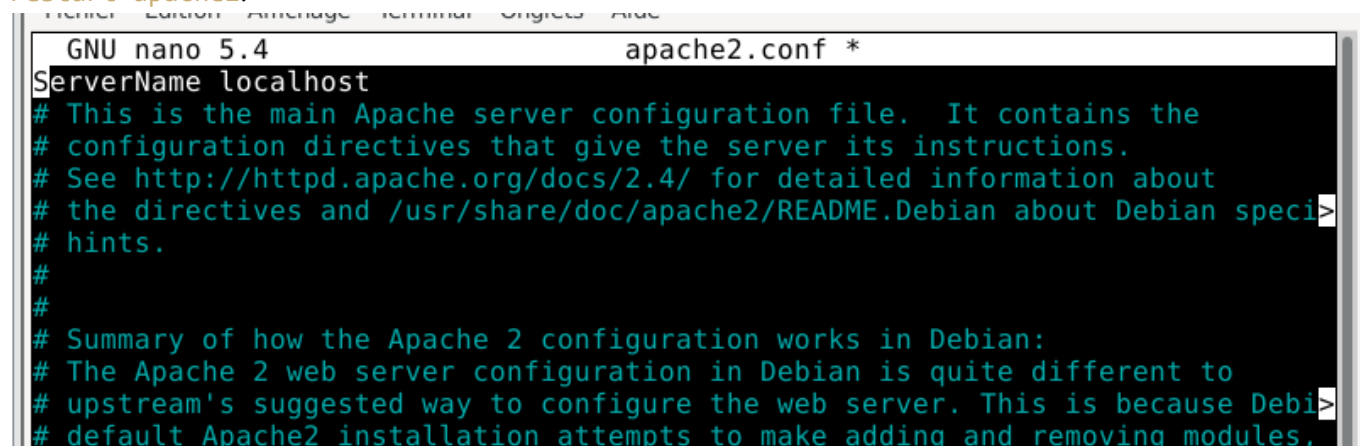
9. Notez la différence entre les commandes « `apache2 -version` » et « `apache2 -v` ». « `apache2 -v` » ne donne pas d'erreur mais « `apache2 -version` » peut le faire. S'il y a un message d'erreur, il faut le corriger.

Je n'ai aucun message d'erreur avec la commande `apache2 -v` et `apache2 -version`. On peut voir que j'obtiens la version du serveur Apache avec les deux commandes.

```
root@SAE:/etc/apache2/sites-available# apache2 -v
Server version: Apache/2.4.53 (Debian)
Server built: 2022-03-14T16:28:35
root@SAE:/etc/apache2/sites-available# apache2 -version
Server version: Apache/2.4.53 (Debian)
Server built: 2022-03-14T16:28:35
```

**9.a En fait le message d'erreur, s'il y en a, ce n'est pas une erreur grave mais c'est ce qu'on appelle « un avertissement amical » (en anglais « friendly warning ») et la réponse est dans le message d'erreur**

Je n'ai obtenu aucun message d'erreur avec les commandes `apache2 -v` et `apache2 -version`. J'ai quand même ajouté `ServerName localhost` dans le fichier `/etc/apache2/apache2.conf` pour éviter d'avoir des erreurs. Pour bien appliquer la configuration, j'ai redémarré le serveur avec la commande `systemctl restart apache2`.



```
GNU nano 5.4 apache2.conf *
ServerName localhost
# This is the main Apache server configuration file. It contains the
# configuration directives that give the server its instructions.
# See http://httpd.apache.org/docs/2.4/ for detailed information about
# the directives and /usr/share/doc/apache2/README.Debian about Debian speci
# hints.
#
#
# Summary of how the Apache 2 configuration works in Debian:
# The Apache 2 web server configuration in Debian is quite different to
# upstream's suggested way to configure the web server. This is because Debi
# default Apache2 installation attempts to make adding and removing modules,
```

10. Création d'une partie administration (« privée ») du serveur Web accessible uniquement par login/mot de passe géré directement par le serveur Web (et non pas par une base de donnée):

**10.a Créer un dossier « private » dans la racine du serveur web (/var/www/html/)**

Je crée un dossier `private` dans la racine du serveur web avec la commande `mkdir /var/www/html/private`. Et j'y accède avec la commande `cd /var/www/html/private`.

```
root@SAE:/# mkdir /var/www/html/private
root@SAE:/# cd /var/www/html/private
root@SAE:/var/www/html/private#
```

## o Créer dedans un fichier index.html. Personnaliser le contenu et le titre de cette page

Je crée un fichier `index.html` dans le dossier `private` avec la commande `touch index.html`. Et j'édite le fichier avec la commande `nano index.html`.

```
root@SAE:/var/www/html/private# touch index.html
root@SAE:/var/www/html/private# nano index.html
```

J'ajoute le code suivant :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Administration</title>
</head>
<body>
  <h1>Page privée</h1>
  <p>Cette page est privée</p>
  <a href="/">Retour à l'accueil</a>
</body>
</html>
```

Page html simple pour l'accueil :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Accueil</title>
</head>
<body>
  <h1>Page d'accueil</h1>
  <p>Bienvenue sur la page d'accueil</p>
  <a href="/private">Administration</a>
</body>
</html>
```

## 10.b Depuis la page d'accueil du serveur, créer un lien qui permet d'accéder à l'espace privé (`<a href="/private">administration</a>`) et vice-versa.\*\*

Je crée un lien qui permet d'accéder à l'espace privé depuis la page d'accueil du serveur. Pour cela, j'édite le fichier `index.html` dans la racine du serveur web avec la commande `nano /var/www/html/index.html`.

### 10.c Inspirez-vous du text TD pour n'autoriser l'accès au dossier « privé » qu'après authentification à l'utilisateur de login « admin » et de mot de passe « lannion ».

D'abord je met en place pour un mot de passe pour un utilisateur `admin` avec le mot de passe `lannion`. Pour cela, j'utilise la commande : `htpasswd -c /etc/apache2/pass admin`. On me demande de saisir le mot de passe et de le confirmer. Ensuite, je redémarre le serveur avec la commande `systemctl restart apache2`.

```
root@SAE:/var/www/html/private# cd ~
root@SAE:~# htpasswd -c /etc/apache2/pass admin
New password:
Re-type new password:
Adding password for user admin
root@SAE:~# systemctl restart apache2
root@SAE:~#
```

Ensuite, je configure le fichier de configuration du serveur web pour autoriser l'accès au dossier `private` qu'après authentification de l'utilisateur `admin` avec le mot de passe `lannion`. Pour cela, j'édite le fichier de configuration `/etc/apache2/sites-available/000-default.conf` avec la commande `nano /etc/apache2/sites-available/000-default.conf`. J'ajoute les lignes suivantes :

```
<Directory "/var/www/html/private">
    AuthType Basic
    AuthName "Veuillez saisir votre mot de login/passe"
    AuthUserFile /etc/apache2/pass
    Require valid-user
</Directory>
```

```
GNU nano 5.4 /etc/apache2/sites-available/000-default.conf *
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example t>
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

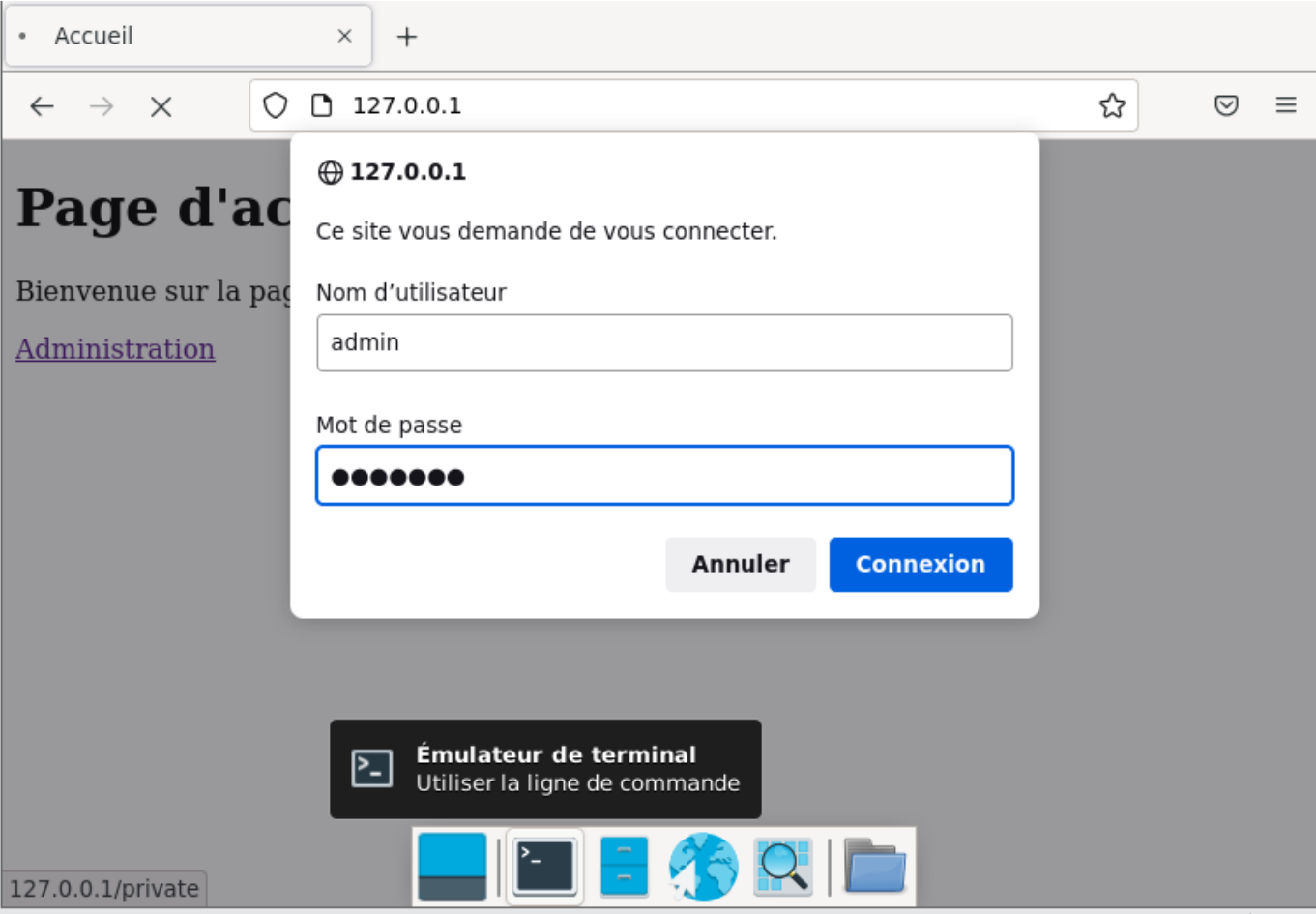
<Directory "/var/www/html/private">
    AuthType Basic
    AuthName "Veuillez saisir votre mot de login/passe"
    AuthUserFile "/etc/apache2/pass"
    Require valid-user
</Directory>

```

<sup>^</sup>G Aide    <sup>^</sup>O Écrire    <sup>^</sup>W Chercher    <sup>^</sup>K Couper    <sup>^</sup>T Exécuter  
<sup>^</sup>X Quitter    <sup>^</sup>R Lire fich.    <sup>^</sup>\ Remplacer    <sup>^</sup>U Coller    <sup>^</sup>J Justifier

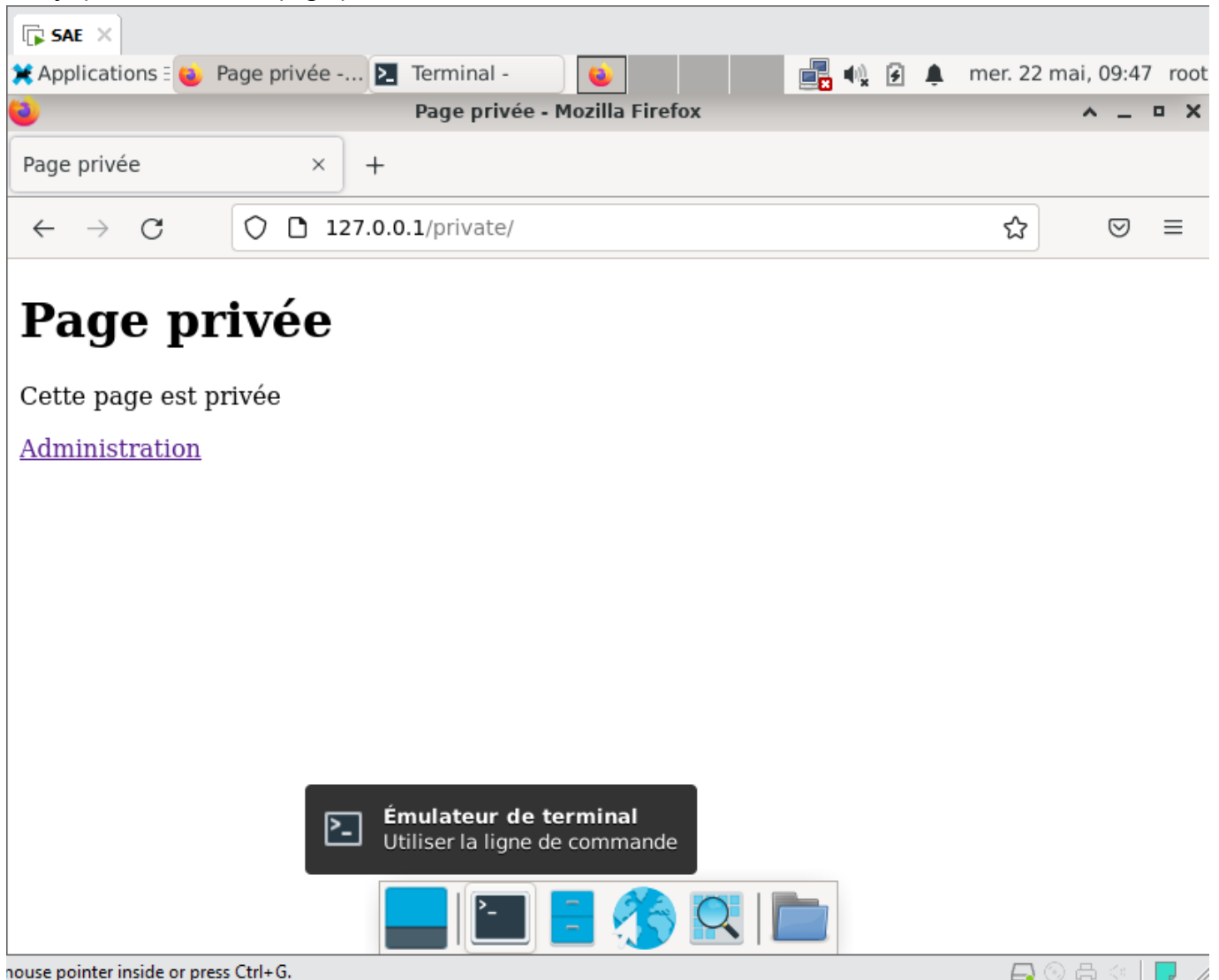
Je redémarre le serveur avec la commande `systemctl restart apache2`.

Maintant quand je clique sur le lien **Administration** depuis la page d'accueil, on me demande de saisir un login et un mot de passe. Je saisis **admin** et **lannion** et je peux accéder à la page privée.





Puis je peux accéder à la page privée.



## Le module PHP

La façon la plus simple de vérifier la configuration PHP, y compris pour voir les modules associées qui sont installés, est de créer un script php de test en utilisant la fonction `php : phpinfo()`. § Repérer le dossier des pages Web de votre serveur Apache

1. Créez dedans un dossier secret et créer dedans un fichier texte nommé : `phpinfo.php` et incluant le code suivant : `<?php phpinfo();` ?

Je crée un dossier `secret` dans la racine du serveur web avec la commande `mkdir /var/www/html/.secret`. Et j'y accède avec la commande `cd /var/www/html/.secret`.

Je crée un fichier `phpinfo.php` dans le dossier `secret` avec la commande `touch phpinfo.php`. Et j'édite le fichier avec la commande `nano phpinfo.php`.

Je rajoute le code suivant :

```
<?php
phpinfo();
?>
```



```
root@SAE:~# mkdir /var/www/html/secret
root@SAE:~# cd /var/www/html/secret
root@SAE:/var/www/html/secret# touch phpinfo.php
root@SAE:/var/www/html/secret# nano phpinfo.php
root@SAE:/var/www/html/secret# cd ..
root@SAE:/var/www/html# mv secret/ .secret
root@SAE:/var/www/html# ls
index.html  private
root@SAE:/var/www/html# cd .secret
root@SAE:/var/www/html/.secret# ls
phpinfo.php
```



## 2. Pourquoi est-il recommandé de créer un dossier secret ?

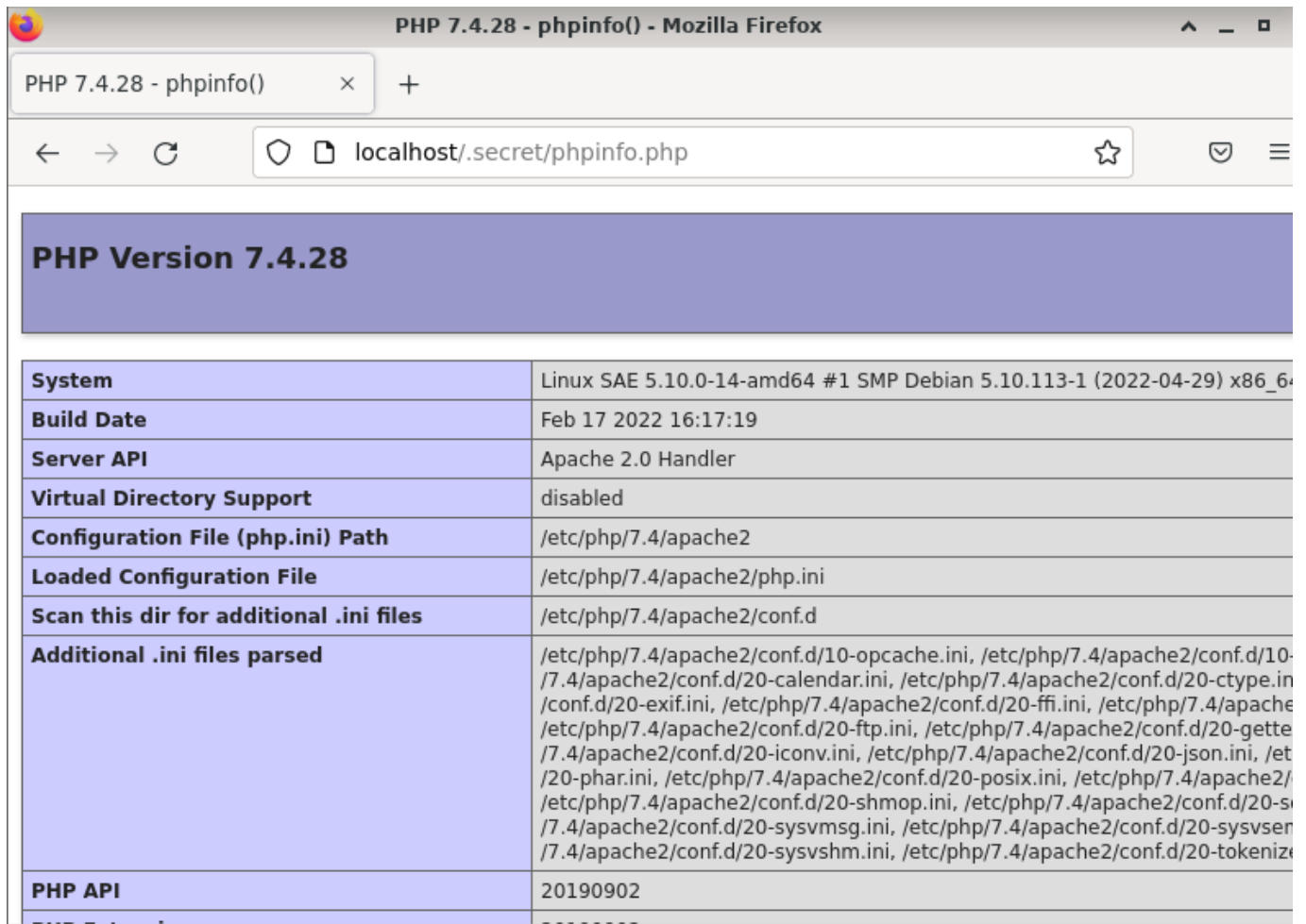
Il est recommandé de créer un dossier **secret** pour stocker les fichiers sensibles ou confidentiels. Cela permet de les protéger et de les cacher des utilisateurs non autorisés. Ou encore pour les protéger des robots qui scannent les sites web à la recherche de failles de sécurité.

## 3. Quelle est l'URL pour accéder à votre fichier php depuis un navigateur Web ?

Pour accéder au fichier **phpinfo.php** depuis un navigateur web, on peut utiliser l'url **<http://localhost/.secret/phpinfo.php>** ou en remplaçant **localhost** par l'ip de la machine virtuelle qui est local **127.0.0.1**. On peut préciser le port 80 pour le serveur web.

## 4. Accédez à cette URL. Peut-on confirmer après cet accès que le module PHP est activé par le serveur ?

Je peux confirmer que le module PHP est activé par le serveur. On peut voir que le fichier **phpinfo.php** s'exécute correctement et qu'il affiche les informations de configuration de PHP. On peut voir que le module PHP est activé par le serveur.



**PHP Version 7.4.28**

<b>System</b>	Linux SAE 5.10.0-14-amd64 #1 SMP Debian 5.10.113-1 (2022-04-29) x86_64
<b>Build Date</b>	Feb 17 2022 16:17:19
<b>Server API</b>	Apache 2.0 Handler
<b>Virtual Directory Support</b>	disabled
<b>Configuration File (php.ini) Path</b>	/etc/php/7.4/apache2
<b>Loaded Configuration File</b>	/etc/php/7.4/apache2/php.ini
<b>Scan this dir for additional .ini files</b>	/etc/php/7.4/apache2/conf.d
<b>Additional .ini files parsed</b>	/etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10- /7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.in /conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ffi.ini, /etc/php/7.4/apache /etc/php/7.4/apache2/conf.d/20-ftp.ini, /etc/php/7.4/apache2/conf.d/20-gette /7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /et /20-phar.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/ /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-si /7.4/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvser /7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokeniz
<b>PHP API</b>	20190902

5. Confirmez que le module PHP est bien activé par le serveur Web en vérifiant : (1) le contenu d'un dossier propre à Apache et (2) le contenu d'un fichier X (X est à préciser, indication : voir TD). (3) Vérifiez avec la commande « ls -l » appliqué à ce fichier X : que le fichier X n'est pas un fichier « classique »

#### 5.a Vérifiez que PHP est bien activé par le contenu d'un dossier propre à Apache

Pour vérifier que le module PHP est activé par le serveur web, on peut vérifier le contenu du dossier `/etc/apache2/mods-enabled/`. On peut voir que le module `php7.4.load` est présent dans le dossier. On peut voir que le module PHP est activé par le serveur web.

Ici le module `php7.4.load` est activé. car il se trouve dans `mods-enabled`.

```
root@SAE:/var/www/html/.secret# cd /etc/apache2/mods-enabled/
root@SAE:/etc/apache2/mods-enabled# ls
access_compat.load  authz_user.load  filter.load       php7.4.load
alias.conf          autoindex.conf  mime.conf         reqtimeout.conf
alias.load          autoindex.load  mime.load         reqtimeout.load
auth_basic.load     deflate.conf     mpm_prefork.conf setenvif.conf
authn_core.load     deflate.load     mpm_prefork.load setenvif.load
authn_file.load     dir.conf        negotiation.conf  status.conf
authz_core.load     dir.load        negotiation.load  status.load
authz_host.load     env.load        php7.4.conf
root@SAE:/etc/apache2/mods-enabled# S
```

#### 5.b Vérifiez que PHP est bien activé par le contenu d'un fichier X

Pour vérifier que le module PHP est activé par le serveur web, on peut vérifier le contenu du fichier `/etc/apache2/mods-enabled/php7.4.load`. On peut voir que le fichier est un lien symbolique vers le fichier `/etc/apache2/mods-available/php7.4.load`. On peut voir que le module PHP est activé par le serveur web.

En faisant nano sur le fichier `php7.4.load` on peut voir que c'est un lien symbolique. `LoadModule php7_module /usr/lib/apache2/modules/libphp7.4.so`

```
GNU nano 5.4                                php7.4.load *
# Conflicts: php5
# Depends: mpm_prefork
LoadModule php7_module /usr/lib/apache2/modules/libphp7.4.so
```

### 5.c Vérifiez avec la commande « `ls -l` » appliqué à ce fichier X : que le fichier X n'est pas un fichier « classique »

Pour vérifier que le fichier `php7.4.load` n'est pas un fichier classique, on peut utiliser la commande `ls -l /etc/apache2/mods-enabled/php7.4.load`. On peut voir que le fichier est un lien symbolique vers le fichier `/etc/apache2/mods-available/php7.4.load`. On peut voir que le fichier n'est pas un fichier classique.

La commande retourne : `lrwxrwxrwx 1 root root 29 4 mai 2022 /etc/apache2/mods-enabled/php7.4.load -> ../mods-available/php7.4.load`

```
root@SAE:/etc/apache2/mods-enabled# ls -l /etc/apache2/mods-enabled/php7.4.load
lrwxrwxrwx 1 root root 29 4 mai 2022 /etc/apache2/mods-enabled/php7.4.load -> ../mods-available/php7.4.load
```

6. Accédez à votre script php : `phpinfo.php` avec le navigateur et relevez les informations suivantes sur la page affichée par le serveur :

#### 6.i La version exacte du module PHP utilisé par notre serveur Web

La version exacte du module PHP utilisé par notre serveur Web est **7.4.28**.

**PHP Version 7.4.28**

#### 6.ii Le dossier de configuration du module PHP utilisé par notre serveur Web

Le dossier de configuration du module PHP utilisé par notre serveur Web est `/etc/php/7.4/apache2`.

#### 6.iii Le fichier de configuration de php pour le serveur Web

Le fichier de configuration de PHP pour le serveur Web est `/etc/php/7.4/apache2/php.ini`.

Configuration File (php.ini) Path	/etc/php/7.4/apache2
-----------------------------------	----------------------

## 6.b La valeur de l'étiquette appelée « short\_open\_tag »

La valeur de l'étiquette appelée `short_open_tag` est `Off`.

<code>short_open_tag</code>	Off
-----------------------------	-----

## 6.c À votre avis, dans quel fichier de configuration (chemin exacte) peut-on modifier la valeur de cette étiquette « short\_open\_tag »?

Pour modifier la valeur de l'étiquette `short_open_tag`, on peut modifier le fichier de configuration `/etc/php/7.4/apache2/php.ini`. On peut voir que l'étiquette `short_open_tag` est présente dans le fichier de configuration de PHP.

On peut observer l'option ci-dessous dans le fichier `php.ini`.

```
; This directive determines whether or not PHP will recognize code between
; <? and ?> tags as PHP source which should be processed as such. It is
; generally recommended that <?php and ?> should be used and that this feature
; should be disabled, as enabling it may result in issues when generating XML
; documents, however this remains supported for backward compatibility reasons.
; Note that this directive does not control the <?= shorthand tag, which can be
; used regardless of this directive.
; Default Value: On
; Development Value: Off
; Production Value: Off
; http://php.net/short-open-tag
short open tag = Off
```

## 6.d Si on devrait modifier la valeur de cette étiquette dans un fichier de configuration, faut-il recharger/relancer le serveur Web pour que la nouvelle valeur soit prise en compte ?

Si on devrait modifier la valeur de l'étiquette `short_open_tag` dans un fichier de configuration, il faut recharger le serveur web pour que la nouvelle valeur soit prise en compte. Pour recharger le serveur web, on peut utiliser la commande `systemctl reload apache2`.

Source : <https://www.php.net/manual/fr/ini.core.php#ini.short-open-tag>

## 7. À l'aide de la commande « whereis », trouvez le chemin du binaire (de l'exécutable si vous préférez) du module php

Pour trouver le chemin du binaire du module PHP, on peut utiliser la commande `whereis php`. On peut voir que le binaire se trouve dans `/usr/bin/php7.4` et `/usr/bin/php`.

```
root@SAE:/etc/apache2/mods-enabled# whereis php
php: /usr/bin/php7.4 /usr/bin/php /usr/lib/php /etc/php /usr/share/php7.4-readline /usr/share/php7.4-common /usr/share/php7.4-json /usr/share/php7.4-opcache /usr/share/man/man1/php.1.gz
```

## 7.i Confirmez, avec l'option -v du binaire, la version du module php obtenue précédemment

Pour confirmer la version du module PHP obtenue précédemment, on peut utiliser la commande `php -v`. On peut voir que la version du module PHP est `7.4.28`.

```
root@SAE:/etc/apache2/mods-enabled# php -v
PHP 7.4.28 (cli) (built: Feb 17 2022 16:17:19) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
    with Zend OPcache v7.4.28, Copyright (c), by Zend Technologies
```

## 8. Désactiver le module PHP en ligne de commande

Pour désactiver le module PHP en ligne de commande, on peut utiliser la commande `a2dismod php7.4`. On peut voir que le module PHP est désactivé par le serveur web. On peut voir que le module PHP n'est plus activé par le serveur web.

Source : <https://manpages.ubuntu.com/manpages/trusty/man8/a2enmod.8.html>

## 9. Faut-il relancer le serveur Apache pour prendre en charge la désactivation ? Si oui faites-le.

Oui, il faut redémarrer le serveur apache quand on désactive le php. Car on a le message quand on exécute la commande.

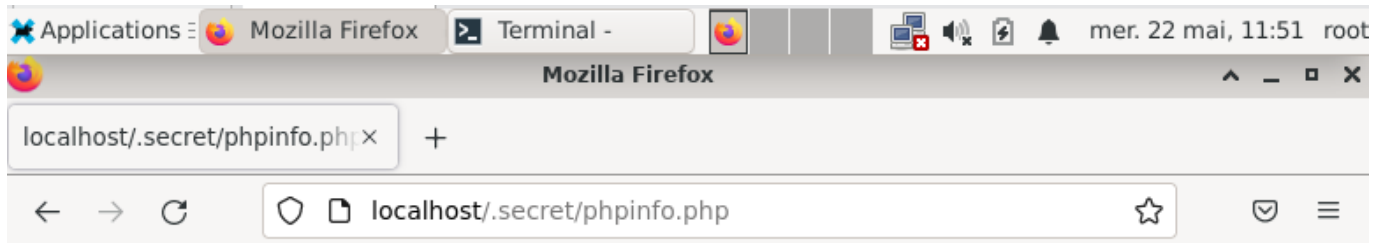
```
root@SAE:~# a2dismod php7.4
Module php7.4 disabled.
To activate the new configuration, you need to run:
    systemctl restart apache2
root@SAE:~# systemctl restart apache2
```

```
root@SAE:~# a2dismod php7.4 Module php7.4 disabled. To activate the new configuration, you need to run:
systemctl restart apache2 root@SAE:~#
```

## 10. Quelle est la conséquence de cette désactivation ? Est-elle dangereuse dans le cas où il subsiste des fichiers PHP sur notre serveur ? Effectuez deux tests : un avec un accès URL et un test de vérification du contenu d'un dossier propre à Apache (à préciser) pour montrer la désactivation du PHP.

La désactivation du module PHP entraîne l'impossibilité pour le serveur Apache d'interpréter et d'exécuter les scripts PHP. Cela peut être dangereux si des fichiers PHP sensibles restent sur le serveur, car le code PHP pourrait être exposé directement au client, révélant ainsi des informations confidentielles.

Test avec accès URL J'accède à l'URL (<http://localhost/.secret/phpinfo.php>) du fichier `phpinfo.php` après désactivation de PHP :



La page qui nous est retournée est blanche. Car le script ne peut s'exécuter.

Test de vérification du contenu d'un dossier propre à Apache On vérifie le contenu du dossier `/etc/apache2/mods-enabled/` pour confirmer que le module PHP est désactivé : `ls -l /etc/apache2/mods-enabled/`

```

root@SAE:~# systemctl restart apache2
root@SAE:~# ls -l /etc/apache2/mods-enabled/
total 0
lrwxrwxrwx 1 root root 36 3 mai 2022 access_compat.load -> ../mods-available/access_compat.load
lrwxrwxrwx 1 root root 28 3 mai 2022 alias.conf -> ../mods-available/alias.conf
lrwxrwxrwx 1 root root 28 3 mai 2022 alias.load -> ../mods-available/alias.load
lrwxrwxrwx 1 root root 33 3 mai 2022 auth_basic.load -> ../mods-available/auth_basic.load
lrwxrwxrwx 1 root root 33 3 mai 2022 authn_core.load -> ../mods-available/authn_core.load
lrwxrwxrwx 1 root root 33 3 mai 2022 authn_file.load -> ../mods-available/authn_file.load
lrwxrwxrwx 1 root root 33 3 mai 2022 authz_core.load -> ../mods-available/authz_core.load
lrwxrwxrwx 1 root root 33 3 mai 2022 authz_host.load -> ../mods-available/authz_host.load
lrwxrwxrwx 1 root root 33 3 mai 2022 authz_user.load -> ../mods-available/authz_user.load
lrwxrwxrwx 1 root root 32 3 mai 2022 autoindex.conf -> ../mods-available/autoindex.conf
lrwxrwxrwx 1 root root 32 3 mai 2022 autoindex.load -> ../mods-available/autoindex.load
lrwxrwxrwx 1 root root 30 3 mai 2022 deflate.conf -> ../mods-available/deflate.conf
lrwxrwxrwx 1 root root 30 3 mai 2022 deflate.load -> ../mods-available/deflate.load
lrwxrwxrwx 1 root root 26 3 mai 2022 dir.conf -> ../mods-available/dir.conf
lrwxrwxrwx 1 root root 26 3 mai 2022 dir.load -> ../mods-available/dir.load
lrwxrwxrwx 1 root root 26 3 mai 2022 env.load -> ../mods-available/env.load
lrwxrwxrwx 1 root root 29 3 mai 2022 filter.load -> ../mods-available/filter.load
lrwxrwxrwx 1 root root 27 3 mai 2022 mime.conf -> ../mods-available/mime.conf
lrwxrwxrwx 1 root root 27 3 mai 2022 mime.load -> ../mods-available/mime.load
lrwxrwxrwx 1 root root 34 4 mai 2022 mpm_prefork.conf -> ../mods-available/mpm_prefork.conf
lrwxrwxrwx 1 root root 34 4 mai 2022 mpm_prefork.load -> ../mods-available/mpm_prefork.load
lrwxrwxrwx 1 root root 34 3 mai 2022 negotiation.conf -> ../mods-available/negotiation.conf
lrwxrwxrwx 1 root root 34 3 mai 2022 negotiation.load -> ../mods-available/negotiation.load
lrwxrwxrwx 1 root root 33 3 mai 2022 reqtimeout.conf -> ../mods-available/reqtimeout.conf
lrwxrwxrwx 1 root root 33 3 mai 2022 reqtimeout.load -> ../mods-available/reqtimeout.load
lrwxrwxrwx 1 root root 31 3 mai 2022 setenvif.conf -> ../mods-available/setenvif.conf
lrwxrwxrwx 1 root root 31 3 mai 2022 setenvif.load -> ../mods-available/setenvif.load
lrwxrwxrwx 1 root root 29 3 mai 2022 status.conf -> ../mods-available/status.conf
lrwxrwxrwx 1 root root 29 3 mai 2022 status.load -> ../mods-available/status.load

```

Le fichier php7.4.load n'est plus être présent dans ce dossier. Cela montre que le module PHP n'est plus activé par Apache.

11. Réactiver le module PHP en ligne de commande. Mêmes questions que précédemment :

### 11.a Faut-il redémarrer le serveur ? Si oui faites le

Quand je réactive le module php avec `a2enmod php7.4`, j'obtiens :

```

root@SAE:~# a2enmod php7.4
Considering dependency mpm_prefork for php7.4:
Considering conflict mpm_event for mpm_prefork:
Considering conflict mpm_worker for mpm_prefork:
Module mpm_prefork already enabled
Considering conflict php5 for php7.4:
Enabling module php7.4.
To activate the new configuration, you need to run:
    systemctl restart apache2
root@SAE:~# systemctl restart apache2

```

root@SAE:~#

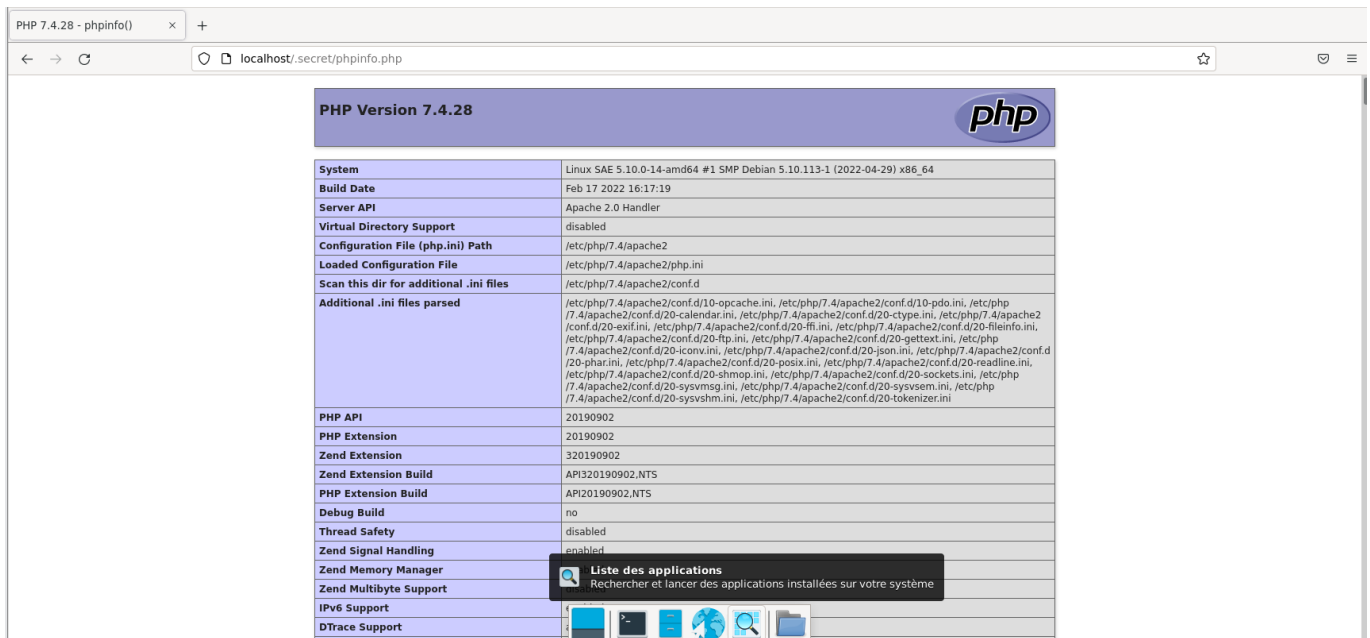
systemctl restart apache2 root@SAE:~# a2enmod php7.4 Considering dependency mpm\_prefork for php7.4: Considering conflict mpm\_event for mpm\_prefork: Considering conflict mpm\_worker for mpm\_prefork: Module mpm\_prefork already enabled Considering conflict php5 for php7.4: Enabling module php7.4. To activate the new configuration, you need to run: systemctl restart apache2 root@SAE:~#

Ici la commande nous dit qu'il faut redémarrer le serveur pour que l'activation du php soit effective.

### 11.b Effectuez les mêmes deux tests de la question précédente

Test avec accès URL J'accède à l'URL (<http://localhost/.secret/phpinfo.php>) du fichier phpinfo.php après désactivation de PHP :





Ici j'obtiens la page avec les informations du php que nous retourne le script php : phpinfo.php.

Test de vérification du contenu d'un dossier propre à Apache On vérifie le contenu du dossier /etc/apache2/mods-enabled/ pour confirmer que le module PHP est activé et présent : `ls -l /etc/apache2/mods-enabled/`

```

wxrwxrwx 1 root root 34 3 mai 2022 negotiation.load -> ../mods-available/negotiation.load
wxrwxrwx 1 root root 29 22 mai 11:52 php7.4.conf -> ../mods-available/php7.4.conf
wxrwxrwx 1 root root 29 22 mai 11:52 php7.4.load -> ../mods-available/php7.4.load
wxrwxrwx 1 root root 33 3 mai 2022 reqtimeout.conf -> ../mods-available/reqtimeout.conf
wxrwxrwx 1 root root 33 3 mai 2022 reqtimeout.load -> ../mods-available/reqtimeout.load
wxrwxrwx 1 root root 31 3 mai 2022 setenvif.conf -> ../mods-available/setenvif.conf
wxrwxrwx 1 root root 31 3 mai 2022 setenvif.load -> ../mods-available/setenvif.load
wxrwxrwx 1 root root 29 3 mai 2022 status.conf -> ../mods-available/status.conf
wxrwxrwx 1 root root 29 3 mai 2022 status.load -> ../mods-available/status.load
bt@SAE:~#

```

12. Insérer dans le code HTML d'une page nommée « mapage.html » (au début par exemple) un code PHP arbitraire. Par exemple « `<?php echo "coucou, je suis un code php dans une page HTML !" ?>` ». La page mapage.html est à créer dans la racine du serveur Web.

Je crée un fichier `mapage.html` dans la racine du serveur web avec la commande `touch`

`/var/www/html/mapage.html`. Et j'édite le fichier avec la commande `nano /var/www/html/mapage.html`. J'ajoute le code PHP arbitraire `<?php echo "coucou, je suis un code php dans une page HTML !" ?>` au début du fichier.

```

<?php echo "coucou, je suis un code php dans une page HTML !" ?>
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ma page</title>
</head>
<body>
  <h1>Ma page</h1>
  <p>Bienvenue sur ma page</p>

```



```
</body>
</html>
```

### 13. Faut-il redémarrer le serveur pour accéder à la page mapage.html ?

Non, il n'est pas nécessaire de redémarrer le serveur pour accéder à la page `mapage.html`. On peut accéder à la page directement depuis un navigateur web. Ce n'est pas une configuration du serveur web, c'est juste une page HTML avec du code PHP.

### 14. Essayez d'accéder à votre nouvelle page mapage.html, que constatez-vous ? justifiez votre réponse.

Le code PHP arbitraire `<?php echo "coucou, je suis un code php dans une page HTML !" ?>` n'est pas interprété par le serveur web. On peut voir que le code PHP est présent dans le code source de la page

```
<!--?php echo "coucou, je suis un code php dans une page HTML !" ?-->
<!DOCTYPE html>
<html lang="fr">
  <head> ... </head>
  <body>
    <h1>Ma page</h1>
    <p>Bienvenue sur ma page</p>
  </body>
</html>
```

mais commenté :

Pour qu'il soit interprété, il faut que le fichier soit en `.php` et non en `.html`.

### 15. En modifiant le fichier de configuration du module PHP, dites à apache d'interpréter les fichiers d'extensions `.html` et `.html` comme des fichiers php. Indication : voir le TD. Faites le test nécessaire pour confirmer que l'interprétation demandée est correcte.

Pour dire à Apache d'interpréter les fichiers d'extensions `.html` et `.html` comme des fichiers PHP, on peut modifier le fichier de configuration du module PHP `/etc/apache2/mods-available/php7.4.conf`. On peut ajouter les lignes suivantes à la fin du fichier :

```
<FilesMatch "\.html$">
  SetHandler application/x-httpd-php
</FilesMatch>
```

```
<FilesMatch "\.html$">
  SetHandler application/x-httpd-php
</FilesMatch>
```

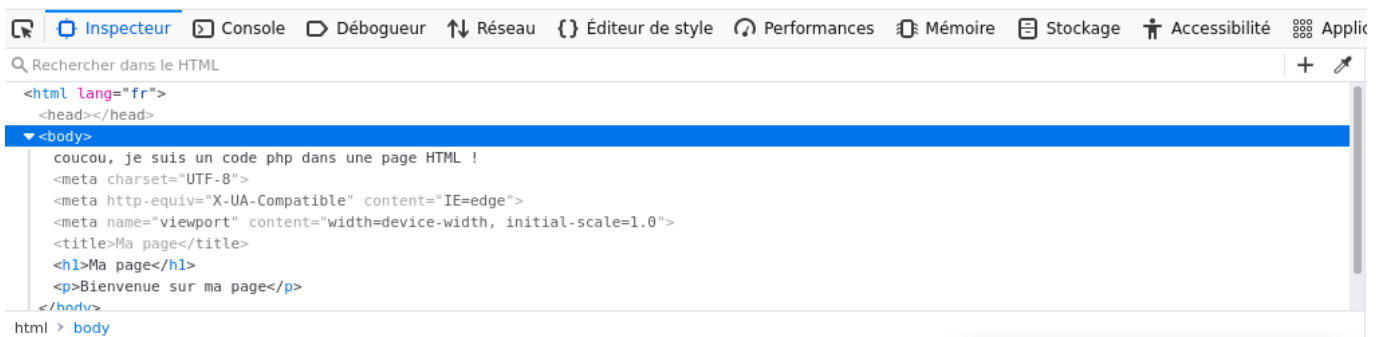
On redémarre le serveur avec la commande `systemctl restart apache2`.

Le code fonctionne maintenant sur la page `mapage.html` :

coucou, je suis un code php dans une page HTML !

## Ma page

Bienvenue sur ma page

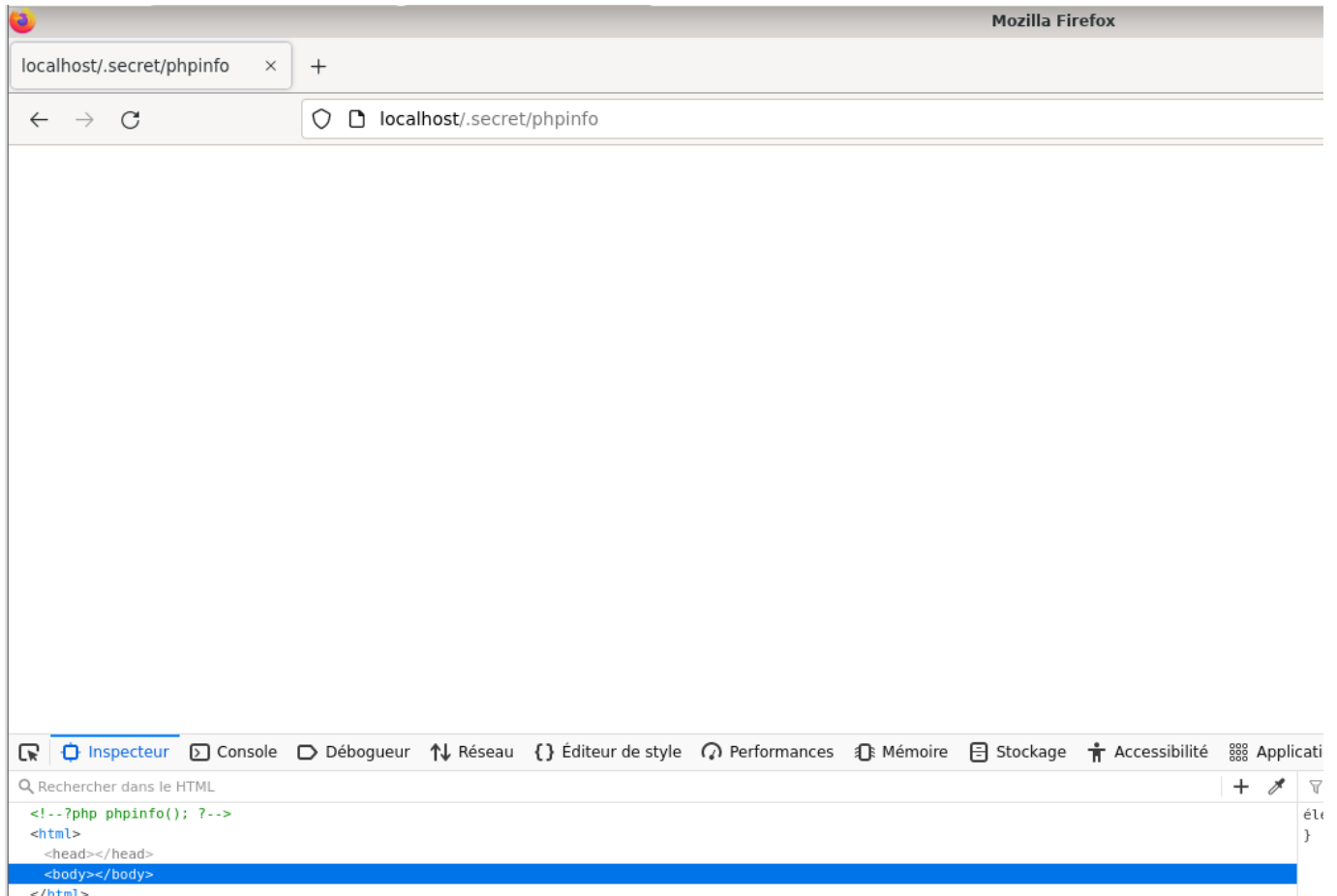


16. Dupliquer le fichier `phpinfo.php` pour créer un nouveau fichier de nom « `phpinfo` » (sans l'extension `.php`). Essayez d'accéder à ce fichier « `phpinfo` » avec votre navigateur Web. Que remarquez-vous ?

Je duplique le fichier `phpinfo.php` pour créer un nouveau fichier de nom `phpinfo` (sans l'extension `.php`) avec la commande `cp phpinfo.php phpinfo`.

```
root@SAE:/var/www/html/private# cd ../.secret
root@SAE:/var/www/html/.secret# cp phpinfo.php phpinfo
root@SAE:/var/www/html/.secret# ls
phpinfo  phpinfo.php
root@SAE:/var/www/html/.secret# S
```

Quand j'accède à la page le code php ne s'exécute pas. Il faut que le fichier soit en `.php` pour que le code soit interprété.



17. En s'inspirant du TD, configurer Apache pour traiter les fichiers sans extension comme du code PHP et confirmer votre configuration par un test.

Pour configurer Apache pour traiter les fichiers sans extension comme du code PHP, on peut ajouter les lignes suivantes dans le fichier de configuration du module PHP `/etc/apache2/mods-available/php7.4.conf` :

```
<FilesMatch "^(^\..)+$">
    SetHandler application/x-httpd-php
</FilesMatch>
```

```
# Execute php dans html
<FilesMatch "\.html$"
    SetHandler application/x-httpd-php
</FilesMatch>

# Execute php fichier sans extension
<FilesMatch "^(^\..)+$">
    SetHandler application/x-httpd-php
</FilesMatch>
```

Quand j'accède à la page `phpinfo` sans l'extension `.php`, le code PHP s'exécute correctement.

**PHP Version 7.4.28**

<b>System</b>	Linux SAE 5.10.0-14-amd64 #1 SMP Debian 5.10.113-1 (2022-04-29) x86_64
<b>Build Date</b>	Feb 17 2022 16:17:19
<b>Server API</b>	Apache 2.0 Handler
<b>Virtual Directory Support</b>	disabled
<b>Configuration File (php.ini) Path</b>	/etc/php/7.4/apache2
<b>Loaded Configuration File</b>	/etc/php/7.4/apache2/php.ini
<b>Scan this dir for additional .ini files</b>	/etc/php/7.4/apache2/conf.d
<b>Additional .ini files parsed</b>	/etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10-pdo.ini, /etc/php/7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.ini, /etc/php/7.4/apache2/conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ffi.ini, /etc/php/7.4/apache2/conf.d/20-fileinfo.ini, /etc/php/7.4/apache2/conf.d/20-ftp.ini, /etc/php/7.4/apache2/conf.d/20-gettext.ini, /etc/php/7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /etc/php/7.4/apache2/conf.d/20-phar.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/conf.d/20-readline.ini, /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-sockets.ini, /etc/php/7.4/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvsem.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokenizer.ini
<b>PHP API</b>	20190902
<b>PHP Extension</b>	20190902

Inspecteur Console Débugueur Réseau Éditeur de style Performances Mémoire Stockage Accessibilité Applications

Rechercher dans le HTML

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <body>
      <div class="center">
    </div>
  </body>
</html>
```

html > body

Filtrer les styles : hov .cls +

```
élément {
}
body {
  background-color: #fff;
  color: #222;
  font-family: sans-serif;
}
```

18. Apache maintenant interprète du code PHP dans du code HTML et c'est super ! Cependant, écrire du code avec la syntaxe `<?php echo "coucou, je suis un code php dans une page HTML !" ?>` peut être rendu plus facile avec une syntaxe simplifiée du genre `<? echo "coucou, je suis un code php dans une page HTML !" ?>` (i.e. sans préciser à chaque fois `php`).

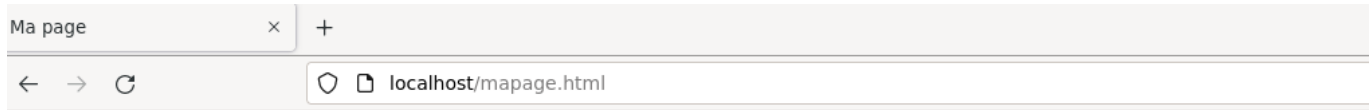
**18.a Dans votre page `mapage.html` changez la syntaxe du code php de `<?php echo "coucou, je suis un code php dans une page HTML !" ?>` vers `<? echo ... ?>`**

Je modifie la syntaxe du code PHP dans la page `mapage.html` de `<?php echo "coucou, je suis un code php dans une page HTML !" ?>` vers `<? echo "coucou, je suis un code php dans une page HTML !" ?>`.

```
GNU nano 3.4
<?echo "coucou, je suis un code php dans une page HTML !" ?>
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Ma page</title>
</head>
<body>
  <h1>Ma page</h1>
  <p>Bienvenue sur ma page</p>
</body>
</html>
```

**18.b Tester votre page depuis un navigateur (attention au cache du navigateur, il faut le vider de temps en temps ou mettez-vous en mode « vie privée stricte » pour éviter le cache pendant les manipulations). Que remarquez-vous ?**

Ici je remarque que le code php ne s'exécute pas. Il apparait dans le code source de la page.



## Ma page

Bienvenue sur ma page

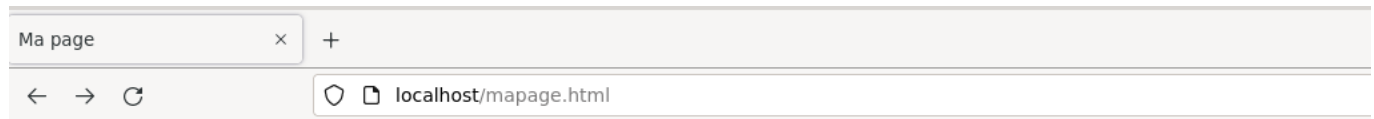


**18.c Proposez une solution et testez de nouveau pour confirmer que vous avez réussi à exécuter le code php avec une syntaxe `<? code; ?>` et non pas `<?php code; ?>`. Indication : « short\_open\_tag » !**

Pour exécuter le code PHP avec une syntaxe `<? code; ?>` et non pas `<?php code; ?>`, on peut activer l'étiquette `short_open_tag` dans le fichier de configuration de PHP `/etc/php/7.4/apache2/php.ini`. On peut modifier la valeur de l'étiquette `short_open_tag` à `On`. Puis on redémarre le serveur avec la commande `systemctl restart apache2`.

```
; This directive determines whether or not PHP will recognize code between
; <? and ?> tags as PHP source which should be processed as such. It is
; generally recommended that <?php and ?> should be used and that this feature
; should be disabled, as enabling it may result in issues when generating XML
; documents, however this remains supported for backward compatibility reasons.
; Note that this directive does not control the <?=> shorthand tag, which can be
; used regardless of this directive.
; Default Value: On
; Development Value: Off
; Production Value: Off
; http://php.net/short-open-tag
short_open_tag = On
```

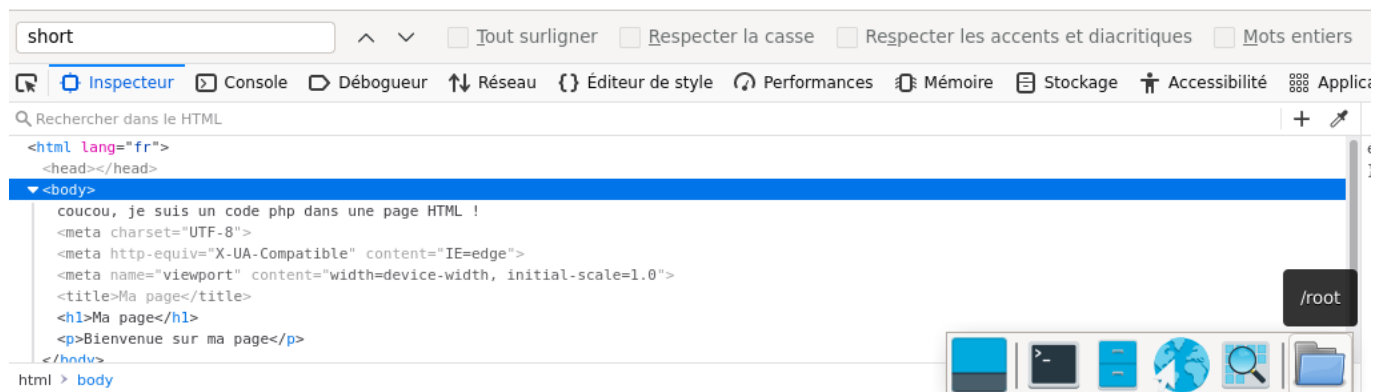
Je redémarre le serveur.



coucou, je suis un code php dans une page HTML !

## Ma page

Bienvenue sur ma page

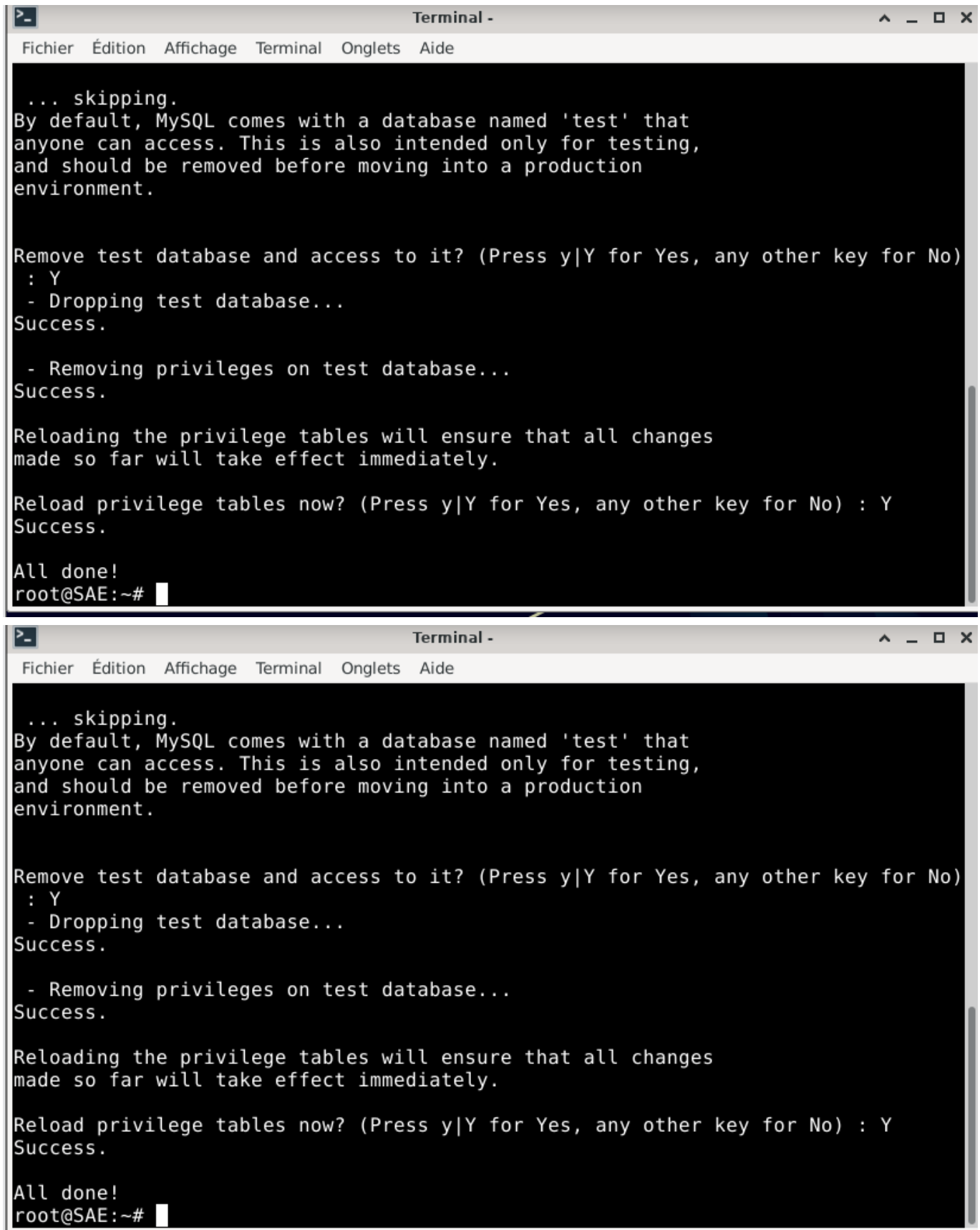


Et voilà le code php s'exécute correctement avec la syntaxe `<? echo "coucou, je suis un code php dans une page HTML !" ?>`.

## MySQL & PHP

1. Sécurisez le serveur MySQL avec la commande « `mysql_secure_installation` » (le mot de passe root du serveur MySQL doit rester : lannion)

Je lance la commande `mysql_secure_installation` pour sécuriser le serveur MySQL. Je réponds aux questions posées par le script d'installation :

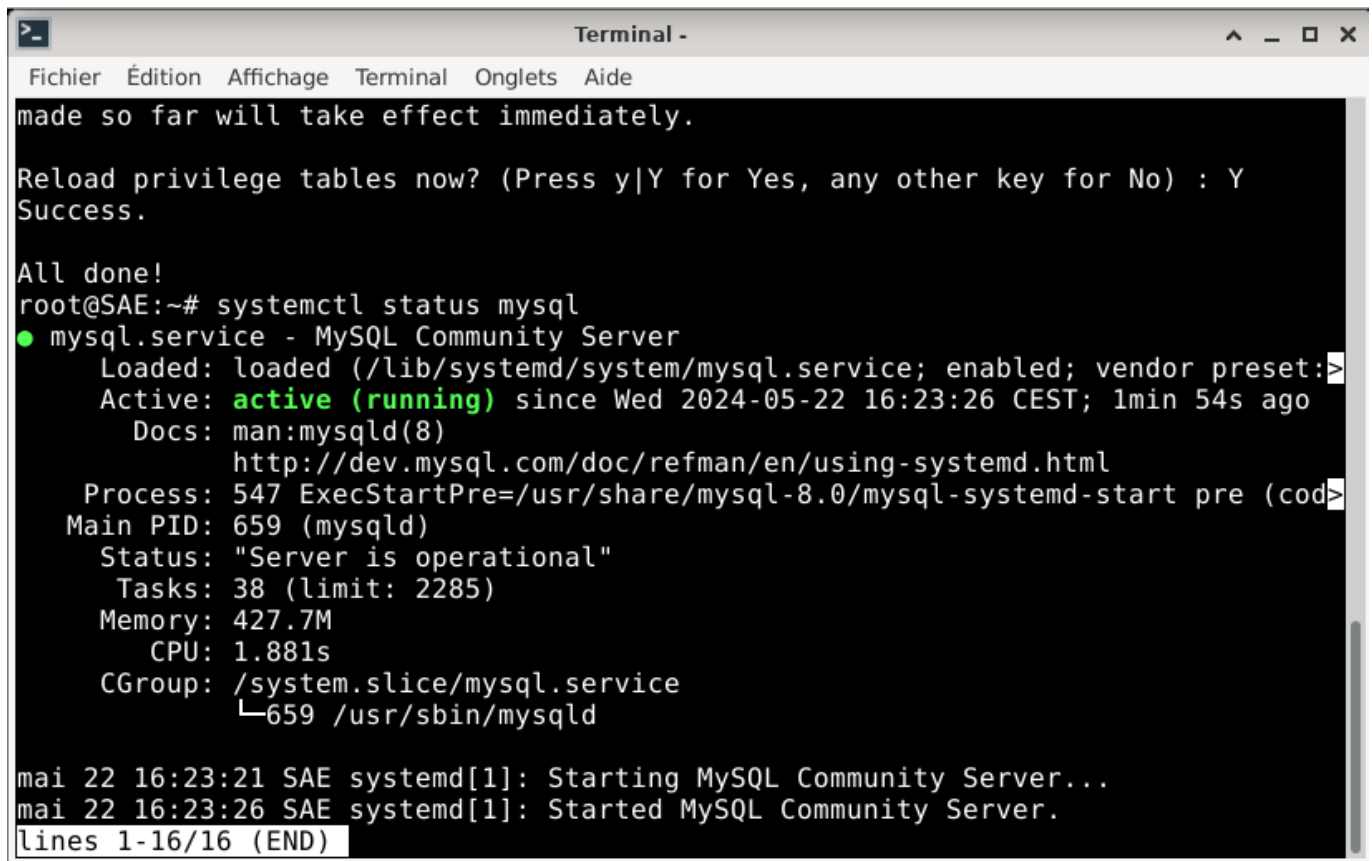


The image shows two identical screenshots of a terminal window. The terminal has a title bar 'Terminal -' and a menu bar with 'Fichier', 'Édition', 'Affichage', 'Terminal', 'Onglets', and 'Aide'. The output text is as follows:

```
> ... skipping.  
By default, MySQL comes with a database named 'test' that  
anyone can access. This is also intended only for testing,  
and should be removed before moving into a production  
environment.  
  
Remove test database and access to it? (Press y|Y for Yes, any other key for No)  
: Y  
- Dropping test database...  
Success.  
  
- Removing privileges on test database...  
Success.  
  
Reloading the privilege tables will ensure that all changes  
made so far will take effect immediately.  
  
Reload privilege tables now? (Press y|Y for Yes, any other key for No) : Y  
Success.  
  
All done!  
root@SAE:~#
```

2. Vérifier le statut d'exécution du service MySQL (assuré par le paquetage mysql-server). Et relevez les informations pertinentes qui montrent que le serveur MySQL s'exécute continuellement (ce qui est normal car c'est un serveur). Relevez à quelle heure exacte, le serveur a été lancé la toute dernière fois.

Pour vérifier le statut d'exécution du service MySQL, on peut utiliser la commande `systemctl status mysql`. On peut voir que le serveur MySQL s'exécute continuellement. On peut voir que le serveur MySQL a été lancé la dernière fois à `mai 22 16:23:21`.



```
Terminal -
Fichier Édition Affichage Terminal Onglets Aide
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : Y
Success.

All done!
root@SAE:~# systemctl status mysql
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset:
   Active: active (running) since Wed 2024-05-22 16:23:26 CEST; 1min 54s ago
     Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
   Process: 547 ExecStartPre=/usr/share/mysql-8.0/mysql-systemd-start pre (cod
   Main PID: 659 (mysqld)
     Status: "Server is operational"
    Tasks: 38 (limit: 2285)
   Memory: 427.7M
      CPU: 1.881s
   CGroup: /system.slice/mysql.service
           └─659 /usr/sbin/mysqld

mai 22 16:23:21 SAE systemd[1]: Starting MySQL Community Server...
mai 22 16:23:26 SAE systemd[1]: Started MySQL Community Server.
lines 1-16/16 (END)
```

3. Identifier la version du serveur MySQL. Attention, l'argument à utiliser pour la commande `mysql` est « `-V` » et non « `-v` »

Pour identifier la version du serveur MySQL, on peut utiliser la commande `mysql -V`. On peut voir que la version du serveur MySQL est `mysql Ver 8.0.29`.



```
root@SAE:~# mysql -V
mysql Ver 8.0.29 for Linux on x86_64 (MySQL Community Server - GPL)
root@SAE:~# S
```

4. Connectez-vous au serveur MySQL avec le login « `root` » et en utilisant la commande `mysql` et les bons arguments à trouver avec « `mysql --help` » si nécessaire. Le mot de passe configuré au préalable lors de l'installation est « `lannion` ».

Pour se connecter au serveur MySQL avec le login `root`, on peut utiliser la commande `mysql -u root -p`. On me demande de saisir le mot de passe et je saisis `lannion`. On peut voir que je suis connecté au serveur



MySQL.

```
root@SAE:~# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.29 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

5. Tout en restant connecté au serveur MySQL (en ligne de commande), afficher les bases de données existantes au niveau du serveur MySQL avec la commande (requête SQL) : « SHOW DATABASES ; »

Pour afficher les bases de données existantes au niveau du serveur MySQL, on peut utiliser la commande SQL `SHOW DATABASES;`. On peut voir que la liste des bases de données existantes sur le serveur MySQL est

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
4 rows in set (0,00 sec)

mysql> S█
```

affichée.

6. Essayer d'avoir le même résultat (i.e. la liste des bases de données existantes sur le serveur) avec un code PHP que vous créez sur le serveur (inspirez-vous du code de TD en améliorant sa sécurité de préférence). Confirmez le résultat en testant votre script avec un accès Web à votre serveur.

Je crée un script PHP nommé `showdb.php` dans la racine du serveur web avec la commande `touch /var/www/html/showdb.php`. Et j'édite le fichier avec la commande `nano /var/www/html/showdb.php`. J'ajoute le code suivant :

```
<?php
$ipserver = "localhost";
$nomutilisateur = "root";
$motdepasse = ""

if (isset($_POST["motdepasse"])) {
    $motdepasse = $_POST["motdepasse"];
```

```
}

$connexion = new mysqli($ipserver, $nomutilisateur, $motdepasse);

if ($connexion->connect_error) {
    die("Erreur de connexion : " . $connexion->connect_error);
}

$resultat = $connexion->query("SHOW DATABASES;");
if ($resultat->num_rows > 0) {
    while ($ligne = $resultat->fetch_assoc()) {
        echo $ligne["Database"] . "<br>";
    }
} else {
    echo "Aucune base de données trouvée";
}

$connexion->close();
?>
```

Source : <https://www.php.net/manual/fr/mysqli.query.php>

**6.a Sous certaines conditions, lorsqu'on essaie d'accéder à la liste des bases de données par le script php, le journal d'erreur d'apache (le /var/log/apache/error.log) peut nous donner l'erreur suivante :**

```
[Sun May 01 17:00:10.816875 2022] [php7:error] [pid 6142] [client 127.0.0.1:47936] PHP Fatal error:
Uncaught Error: Call to undefined function mysqli_connect() in /var/www/html/showdb.php:5\nStack
trace:\n#0 {main}\n thrown in /var/www/html/showdb.php on line 5
```

**6.b Donnez une explication possible à cette erreur**

Cette erreur se produit lorsque la fonction `mysqli_connect()` n'est pas définie. Cela peut se produire si le module PHP MySQLi n'est pas activé sur le serveur web. Il faut activer le module PHP MySQLi pour que la fonction `mysqli_connect()` soit définie.

**6.c Réparez cette erreur en installant le package nécessaire, referrez vous à la section « Environnement pratique : instructions importantes »**

Je dois installer les packages nécessaires pour activer le module PHP MySQLi. Tel que `dpkg -i /usr/local/src/php7.4-mysql_7.4.28-1+deb11u1_amd64.deb` et `dpkg -i /usr/local/src/php-`

mysql\_7.4+76\_all.deb.

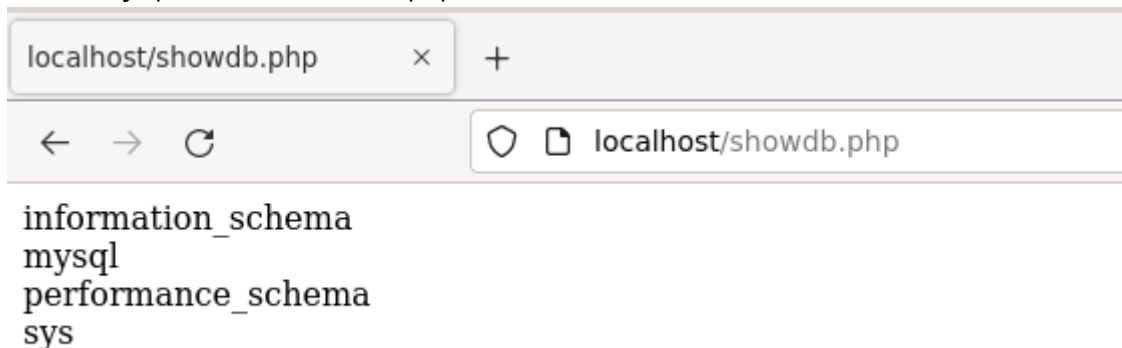
```
root@SAE:/var/www/html# dpkg -i /usr/local/src/php7.4-mysql_7.4.28-1+deb11u1_amd64.deb
Sélection du paquet php7.4-mysql précédemment désélectionné.
(Lecture de la base de données... 110643 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../php7.4-mysql_7.4.28-1+deb11u1_amd64.deb ...
Dépaquetage de php7.4-mysql (7.4.28-1+deb11u1) ...
Paramétrage de php7.4-mysql (7.4.28-1+deb11u1) ...

Creating config file /etc/php/7.4/mods-available/mysqlnd.ini with new version

Creating config file /etc/php/7.4/mods-available/mysqli.ini with new version

Creating config file /etc/php/7.4/mods-available/pdo_mysql.ini with new version
Traitement des actions différées (« triggers ») pour libapache2-mod-php7.4 (7.4.28-1+deb11u1) ...
Traitement des actions différées (« triggers ») pour php7.4-cli (7.4.28-1+deb11u1) ...
root@SAE:/var/www/html# dpkg -i /usr/local/src/php-mysql_7.4+76_all.deb
(Lecture de la base de données... 110656 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../src/php-mysql_7.4+76_all.deb ...
Dépaquetage de php-mysql (2:7.4+76) sur (2:7.4+76) ...
Paramétrage de php-mysql (2:7.4+76) ...
root@SAE:/var/www/html# S
```

Maintant je peux lancer showdb.php et voir les bases de données.



7. Tout en restant connecté au serveur MySQL en ligne de commande, créez une base de données nommée avec votre prénom (exemple « julien »). Montrer en testant en ligne de commande que la base a été bien créée.

Pour créer une base de données nommée avec mon prénom, on peut utiliser la commande SQL **CREATE DATABASE pierre**; On peut voir que la base de données **pierre** a été créée avec succès.

```
mysql> CREATE DATABASE pierre;
Query OK, 1 row affected (0,02 sec)
```

8. Toujours en ligne de commande, créez une table nommée « étudiants » avec 4 champs : un champ « id » entier qui sera clé primaire, un champ « nom », un champ « date\_naissance » et un autre champs « classement ». Montrez que votre table existe bien. Insérez-y 3 enregistrements. Affichez le contenu de ces 3 enregistrements.

```
SELECT DATABASE();
USE pierre;

CREATE TABLE etudiants (
    id INT AUTO_INCREMENT PRIMARY KEY,
    nom VARCHAR(100) NOT NULL,
    date_naissance DATE NOT NULL,
```

```

        classement INT NOT NULL
    );

    INSERT INTO etudiants (nom, date_naissance, classement) VALUES ('Pierre', '2005-10-17', 1);

    INSERT INTO etudiants (nom, date_naissance, classement) VALUES ('Paul', '2005-10-18', 2);

    INSERT INTO etudiants (nom, date_naissance, classement) VALUES ('Jacques', '2005-10-19', 3);

    SELECT * FROM etudiants;

```

```

mysql> CREATE TABLE etudiants (
->   id INT AUTO INCREMENT PRIMARY KEY,
->   nom VARCHAR(100) NOT NULL,
->   date_naissance DATE NOT NULL,
->   classement INT NOT NULL
-> ^C
mysql> INSERT INTO etudiants (nom, date_naissance, classement) VALUES ('Pierre', '2005-10-17', 1);
Query OK, 1 row affected (0,14 sec)

mysql>
mysql> INSERT INTO etudiants (nom, date_naissance, classement) VALUES ('Paul', '2005-10-18', 2);
Query OK, 1 row affected (0,01 sec)

mysql>
mysql> INSERT INTO etudiants (nom, date_naissance, classement) VALUES ('Jacques', '2005-10-19', 3);
Query OK, 1 row affected (0,01 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_pierre |
+-----+
| etudiants         |
+-----+
1 row in set (0,00 sec)

mysql> SELECT * FROM etudiants;
ERROR 1146 (42S02): Table 'pierre.etudiants' doesn't exist
mysql> SELECT * FROM etudiants;
+----+-----+-----+-----+
| id | nom   | date_naissance | classement |
+----+-----+-----+-----+
| 1  | Pierre | 2005-10-17     | 1         |
| 2  | Paul  | 2005-10-18     | 2         |
| 3  | Jacques | 2005-10-19     | 3         |
+----+-----+-----+-----+
3 rows in set (0,00 sec)

```

 **Liste des applications**  
Rechercher et lancer des applications installées sur votre système

C'est fait la table est créée et les enregistrements sont ajoutés.

9. Supprimer un enregistrement de la table ayant un id égale à une valeur connue. Montrez que ça a bien marché.

Pour supprimer un enregistrement de la table ayant un id égale à une valeur connue, on peut utiliser la commande SQL `DELETE FROM etudiants WHERE id = 2;`. On peut voir que l'enregistrement avec l'id 2 a

été supprimé avec succès.

```
mysql> SELECT * FROM etudiants;
+-----+-----+-----+-----+
| id | nom      | date_naissance | classement |
+-----+-----+-----+-----+
| 1  | Pierre   | 2005-10-17     | 1          |
| 2  | Paul     | 2005-10-18     | 2          |
| 3  | Jacques  | 2005-10-19     | 3          |
+-----+-----+-----+-----+
3 rows in set (0,00 sec)

mysql> DELETE FROM etudiants WHERE id = 2;
Query OK, 1 row affected (0,01 sec)

mysql> SELECT * FROM etudiants;
+-----+-----+-----+-----+
| id | nom      | date_naissance | classement |
+-----+-----+-----+-----+
| 1  | Pierre   | 2005-10-17     | 1          |
| 3  | Jacques  | 2005-10-19     | 3          |
+-----+-----+-----+-----+
2 rows in set (0,00 sec)
```

10. Modifier la valeur du champ « date\_naissance » dans l'enregistrement qui a le plus petite valeur du champs identifiant (i.e. « id »). La nouvelle valeur demandée est « 1er Janvier 1990 »

Pour modifier la valeur du champ `date_naissance` dans l'enregistrement qui a le plus petite valeur du champs identifiant (i.e. `id`), on peut utiliser la commande SQL :

```
UPDATE etudiants e1
JOIN (
    SELECT MIN(id) AS min_id
    FROM etudiants
) AS e2 ON e1.id = e2.min_id
SET e1.date_naissance = '1990-01-01';
```

```

mysql> UPDATE etudiants e1
-> JOIN (
->     SELECT MIN(id) AS min_id
->     FROM etudiants
-> ) AS e2 ON e1.id = e2.min_id
-> SET e1.date_naissance = '1990-01-01';
Query OK, 1 row affected (0,03 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM etudiants;
+-----+-----+-----+-----+
| id | nom      | date_naissance | classement |
+-----+-----+-----+-----+
| 4  | Pierre   | 1990-01-01     | 1          |
| 5  | Paul     | 2005-10-18     | 2          |
| 6  | Jacques  | 2005-10-19     | 3          |
+-----+-----+-----+-----+
3 rows in set (0,00 sec)

```

On peut voir que la valeur du champ `date_naissance` dans l'enregistrement qui a le plus petite valeur du champs identifiant a été modifiée avec succès.

11. Refaire les manipulations précédentes faites en ligne de commande (depuis la création de la base de données) avec cette fois-ci un script PHP.

Je crée un script PHP nommé `db.php` dans la racine du serveur web avec la commande `touch /var/www/html/db.php`. Et j'édite le fichier avec la commande `nano /var/www/html/db.php`. J'ajoute le code suivant :

```

<?php
$ipserver = "localhost";
$nomutilisateur = "root";
$motdepasse = "lannion";
$basededonnees = "pierre";

if (isset($_POST["motdepasse"])) {
    $motdepasse = $_POST["motdepasse"];
}

$connexion = new mysqli($ipserver, $nomutilisateur, $motdepasse, $basededonnees);

if ($connexion->connect_error) {
    die("Erreur de connexion : " . $connexion->connect_error);
}

$connexion->query("CREATE TABLE etudiants (
    id INT AUTO_INCREMENT PRIMARY KEY,
    nom VARCHAR(100) NOT NULL,
    date_naissance DATE NOT NULL,
    classement INT NOT NULL
);");

```

```
$connexion->query("INSERT INTO etudiants (nom, date_naissance, classement) VALUES ('Pierre', '2005-10-17', 1);");

$connexion->query("INSERT INTO etudiants (nom, date_naissance, classement) VALUES ('Paul', '2005-10-18', 2);");

$connexion->query("INSERT INTO etudiants (nom, date_naissance, classement) VALUES ('Jacques', '2005-10-19', 3);");

$resultat = $connexion->query("SELECT * FROM etudiants;");

if ($resultat->num_rows > 0) {
    while ($ligne = $resultat->fetch_assoc()) {
        echo $ligne["id"] . " " . $ligne["nom"] . " " . $ligne["date_naissance"] .
" " . $ligne["classement"] . "<br>";
    }
} else {
    echo "Aucun enregistrement trouvé";
}

$connexion->query("DELETE FROM etudiants WHERE id = 2;");

$connexion->query("UPDATE etudiants e1
JOIN (
    SELECT MIN(id) AS min_id
    FROM etudiants
) AS e2 ON e1.id = e2.min_id
SET e1.date_naissance = '1990-01-01';");

$resultat = $connexion->query("SELECT * FROM etudiants;");

if ($resultat->num_rows > 0) {
    while ($ligne = $resultat->fetch_assoc()) {
        echo $ligne["id"] . " " . $ligne["nom"] . " " . $ligne["date_naissance"] .
" " . $ligne["classement"] . "<br>";
    }
} else {
    echo "Aucun enregistrement trouvé";
}

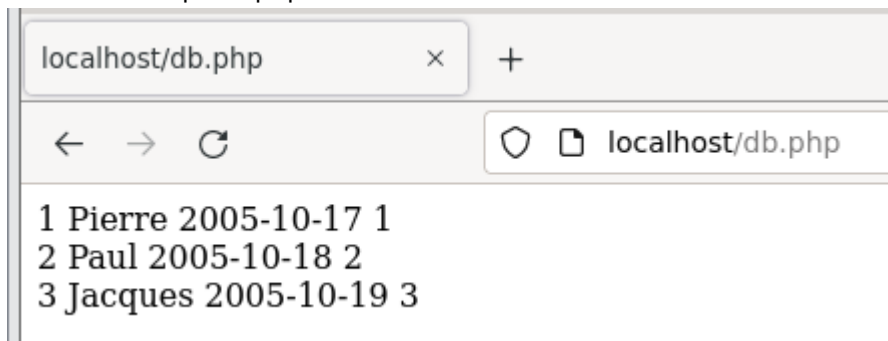
$connexion->close();
?>
```

Pour tester j'ai supprimé la table étudiants et j'ai relancé le script.

```
Database changed
mysql> DROP TABLE etudiands;
Query OK, 0 rows affected (0,04 sec)

mysql> SELECT * FROM etudiands;
ERROR 1146 (42S02): Table 'pierre.etudiands' doesn't exist
mysql> SELECT * FROM etudiands;
ERROR 1146 (42S02): Table 'pierre.etudiands' doesn't exist
mysql> █
```

Je lance le script db.php à la racine du serveur web.



```
localhost/db.php × +
← → ↻ 🔒 📄 localhost/db.php
1 Pierre 2005-10-17 1
2 Paul 2005-10-18 2
3 Jacques 2005-10-19 3
```

Je confirme que les manipulations précédentes faites en ligne de commande ont bien affecté la base de

```
Database changed
mysql> DROP TABLE etudiands;
Query OK, 0 rows affected (0,04 sec)

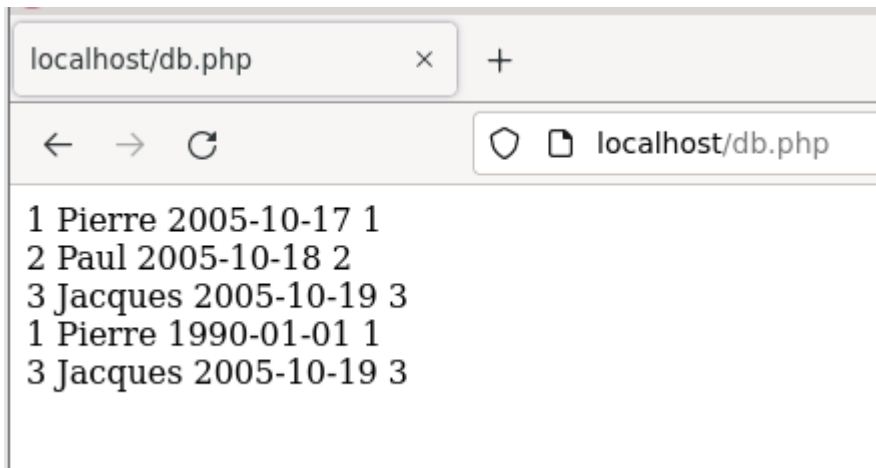
mysql> SELECT * FROM etudiands;
ERROR 1146 (42S02): Table 'pierre.etudiands' doesn't exist
mysql> SELECT * FROM etudiands;
ERROR 1146 (42S02): Table 'pierre.etudiands' doesn't exist
mysql> SELECT * FROM etudiands;
+----+-----+-----+-----+
| id | nom      | date_naissance | classement |
+----+-----+-----+-----+
| 1  | Pierre   | 2005-10-17     | 1          |
| 2  | Paul     | 2005-10-18     | 2          |
| 3  | Jacques  | 2005-10-19     | 3          |
+----+-----+-----+-----+
3 rows in set (0,00 sec)

mysql> █
```

données.

Les deux lignes du bas sont après la suppression de l'enregistrement avec l'id 2 et la modification de la date de naissance de l'enregistrement avec l'id le plus petit.





On vérifie sur la base de données.

```
mysql> SHOW TABLES;
Query OK, 0 rows affected (0,02 sec)

mysql> SELECT * FROM etudiants;
+----+-----+-----+-----+
| id | nom   | date_naissance | classement |
+----+-----+-----+-----+
| 1  | Pierre | 1990-01-01     | 1          |
| 3  | Jacques | 2005-10-19     | 3          |
+----+-----+-----+-----+
2 rows in set (0,00 sec)
```

12. Commencez maintenant la réalisation du reste de votre cahier de charge du projet : section « Cahier de charges » du fichier TD.

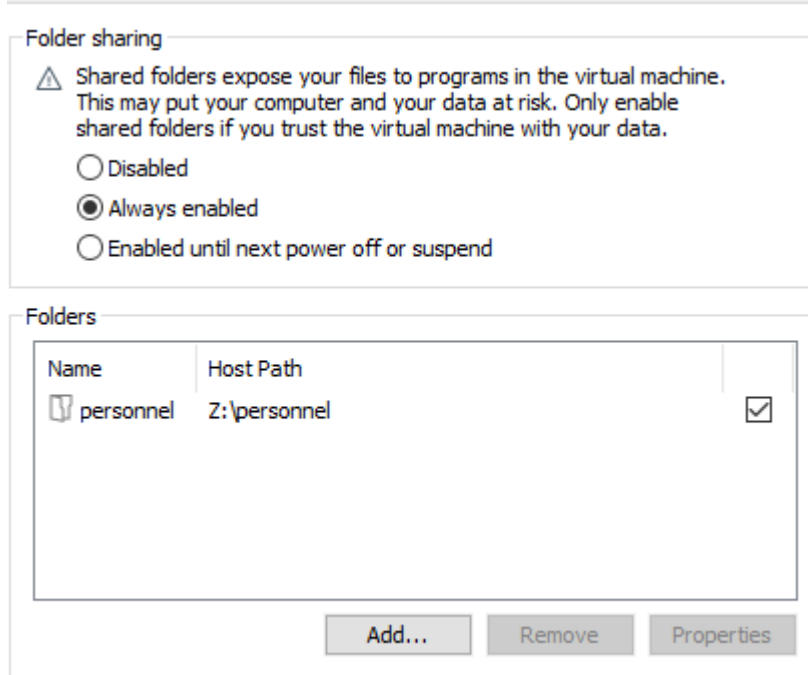
## Cahier de charges

---

1. Installer le module Apache souhaité, s'il n'est pas déjà installé.

**Exemple :** `# apt install fail2ban` Comme on a pas accès à internet depuis la machine virtuelle. On doit télécharger le module depuis le poste via le (github officiel)[<https://github.com/fail2ban/fail2ban>] et le copier sur la machine virtuelle. Je fais de même avec sa dépendance python3-systemd.

Je déplace mon fichier dans Z:\personnel puis active le partage de fichier dans les paramètres de la machine



virtuelle.

J'y accède depuis la machine virtuelle. Et l'installe à l'aide de dpkg.

```
root@SAE:/usr/src# dpkg -i /mnt/hgfs/personnel/python3-systemd_234-3+b4_amd64.deb
Sélection du paquet python3-systemd précédemment désélectionné.
(Lecture de la base de données... 110930 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../python3-systemd_234-3+b4_amd64.deb ...
Dépaquetage de python3-systemd (234-3+b4) ...
Paramétrage de python3-systemd (234-3+b4) ...
root@SAE:/usr/src# dpkg -i /mnt/hgfs/personnel/fail2ban_1.1.0-1.upstream1_all.deb
Sélection du paquet fail2ban précédemment désélectionné.
(Lecture de la base de données... 110948 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../fail2ban_1.1.0-1.upstream1_all.deb ...
Dépaquetage de fail2ban (1.1.0-1~upstream1) sur (1.1.0-1~upstream1) ...
Paramétrage de fail2ban (1.1.0-1~upstream1) ...
Traitement des actions différées (« triggers ») pour man-db (2.9.4-2) ...
```

Je télécharge ensuite le module apache2 de fail2ban. Depuis

(<https://raw.githubusercontent.com/fail2ban/fail2ban/master/config/filter.d/apache-auth.conf>)

[<https://raw.githubusercontent.com/fail2ban/fail2ban/master/config/filter.d/apache-auth.conf>]

Je le copie sur la machine virtuelle et je le place ici `/etc/fail2ban/filter.d/apache-auth.conf`

Je met à jour la configuration de fail2ban pour qu'il utilise le module apache2. Je modifie le fichier `/etc/fail2ban/jail.conf` et j'ajoute les lignes suivantes :

```
# /etc/fail2ban/jail.local
[DEFAULT]
ignoreip = 127.0.0.1/8
```

```
bantime = 600
findtime = 600
maxretry = 3

[apache-auth]
enabled = true
port = http,https
filter = apache-auth
logpath = /var/log/apache2/error.log
maxretry = 3
```

Je vérifie la syntaxe de fail2ban avec la commande `sudo fail2ban-client -d` et je redémarre le service avec `sudo systemctl restart fail2ban`.

```

Fichier  Édition  Affichage  Terminal  Onglets  Aide
able f2b-chain $hdl; done\nnft delete set inet f2b-table <addr_set>\n{ nft li
st table inet f2b-table | grep -qP '^\\s+set\\s+'; } || {\nnft delete table i
net f2b-table\n}]", ['actionflush', "{ nft flush set inet f2b-table <addr_set
> 2> /dev/null; } || {\n{ nft -a list chain inet f2b-table f2b-chain | grep -
oP '@<addr_set>\\s+.*\\s+\\Khandle\\s+(\\d+)$'; } | while read -r hdl; do\nnf
t delete rule inet f2b-table f2b-chain $hdl; done\nnft delete set inet f2b-ta
ble <addr_set>\nnft add set inet f2b-table <addr_set> \\{ type <addr_type>\\;
\\}\\nfor proto in $(echo 'tcp' | sed 's/,/ /g'); do\nnft add rule inet f2b-t
able f2b-chain $proto dport \\{ $(echo 'http,https' | sed s/:-/ /g) \\} <addr
family> saddr @<addr_set> reject\ndone\n}]", ['actioncheck', "nft list chain
inet f2b-table f2b-chain | grep -q '@<addr_set>[ \\t]'", ['actionban', 'nft
add element inet f2b-table <addr_set> \\{ <ip> \\}', ['actionunban', 'nft de
lete element inet f2b-table <addr_set> \\{ <ip> \\}', ['port', 'http,https']
, ['protocol', 'tcp'], ['chain', '<known/chain>'], ['name', 'apache-auth'], [
'actname', 'nftables'], ['table', 'f2b-table'], ['table_family', 'inet'], ['c
hain_type', 'filter'], ['chain_hook', 'input'], ['chain_priority', '-1'], ['a
ddr_type', 'ipv4_addr'], ['blocktype', 'reject'], ['nftables', 'nft'], ['addr
_set', 'addr-set-<name>'], ['addr_family', 'ip'], ['addr_family?family=inet6'
, 'ip6'], ['addr_type?family=inet6', 'ipv6_addr'], ['addr_set?family=inet6',
'addr6-set-<name>']]]
['start', 'apache-auth']
root@SAE:/mnt/hgfs/personnel#
root@SAE:/mnt/hgfs/personnel# sudo systemctl restart fail2ban
root@SAE:/mnt/hgfs/personnel#

```

Aucune erreur n'est retournée.

On voit que fail2ban à la configuration apache-auth.

```

root@SAE:/mnt/hgfs/personnel# sudo fail2ban-client status
Status
|- Number of jail:      1
`- Jail list:  apache-auth

```

J'ai ajouté à la configuration de `/etc/fail2ban/filter.d/apache-auth.conf` :

```
failregex = ^client(?:denied by server configuration|used wrong authenticat>
^user(?:`<F-USER>(?:\S*|.*)</F-USER>(?:auth(?:orig|entic)ati>
^Authorization of user <F-USER>(?:\S*|.*)</F-USER> to access .*>
^%(auth_type)suser <F-USER>(?:\S*|.*)</F-USER>: password mismat>
^%(auth_type)suser `<F-USER>(?:[^\']*|.*)</F-USER>' in realm `.+>
^%(auth_type)sinvalid nonce .* received - length is not\b
^%(auth_type)srealm mismatch - got `(?:[^\']*|.*)' but expected\b
^%(auth_type)sunknown algorithm `(?:[^\']*|.*)' received\b
^invalid qop `(?:[^\']*|.*)' received\b
^%(auth_type)sinvalid nonce .*? received - user attempted time t>
^(?:No h|H)ostname \S+ provided via SNI(?:, but no hostname prov>
^%(_apache_error_client)s user .* (authentication failure|not fo>
^%(_apache_error_client)s user .* (denied by server configuratio>
```

```
# /etc/fail2ban/filter.d/apache-auth.conf
[Definition]
failregex = ^%(_apache_error_client)s user .* (authentication failure|not
found|password mismatch)\s*$
            ^%(_apache_error_client)s user .* (denied by server configuration)\s*$
ignoreregex =
```

2. Vérifier si le module est déjà chargé ou activé par Apache. Par exemple, avec la commande : `# apachectl -t -D DUMP_MODULES`