

REPUBLIQUE DEMOCRATIE DU CONGO
MINISTERE D'EDUCATION SUPERIEUR ET UNIVERSITAIRE
UNIVERSITE DE KINSHASA



FACULTE DE SCIENCE ET TECHNOLOGIE
Mention : Mathématiques Statistiques et Informatiques

TRAVAIL PRATIQUE DE SYSTÈME D'EXPLOITATION

Sujet : Mise en place d'un serveur web Apache ou Nginx sécurisé avec
TLS (HTTPS)

PROMOTION : L2 LMD MSI

REDIGE PAR :

- **BAMPEMBE ITUKU GABRIEL**
- **NZOLO MONGA-NZOLO BENJAMIN**
- **BAKOKO IYA ERICK**
- **MBO GRADY**
- **BONCOUER NAMEGABE SAFARI**
- **SONIA**
- **YANDOLETE YAKA BENIE**
- **MUBWA MISOTO ALEX**
- **MUNGUFENI ANDAMA MARC**
- **MULANGU MULANGU**

Prof. Dr Kasengedia Motumbe Pierre

ANNEE ACADEMIQUE 2025-2026

I. INTRODUCTION

Aujourd’hui, les services web sont devenus indispensables dans presque tous les domaines : éducation, commerce, communication et administration. Chaque fois que nous ouvrons un site dans un navigateur, nous communiquons en réalité avec un serveur web qui traite notre demande et renvoie les pages affichées à l’écran. Des serveurs comme Apache ou Nginx sont largement utilisés pour assurer ce rôle dans les systèmes informatiques modernes.

Mais mettre un site en ligne ne suffit pas : il faut aussi sécuriser les échanges de données. Le protocole HTTP classique ne protège pas les informations transmises entre l’utilisateur et le serveur. C’est pour cette raison que le protocole HTTPS a été introduit. Il s’appuie sur le mécanisme de chiffrement TLS pour protéger la communication, empêcher l’interception des données et garantir une navigation plus sûre.¹

Dans le cadre du cours de Systèmes d’Exploitation , notre projet consiste à installer et configurer un serveur web Apache sécurisé avec HTTPS. Nous avons mis en place l’environnement de travail, activé les modules nécessaires à la sécurité, configuré TLS et déployé un site web de test afin de vérifier le bon fonctionnement du serveur sécurisé. Ce travail nous a permis de comprendre concrètement comment un service web est installé, configuré et protégé au niveau système.

¹ Kurose, J.F, Ross, K.W, *Computer Networking : A Top-Down Approach*, Pearson Education.

II. NOTIONS THEORIQUES

I.1 Serveur web

Un serveur web est un logiciel (et parfois aussi la machine qui l'héberge) dont le rôle est de recevoir les requêtes des clients, généralement des navigateurs web, puis de leur envoyer les pages et ressources demandées. Ces échanges se font à travers le protocole HTTP ou HTTPS. Concrètement, lorsqu'un utilisateur saisit une adresse dans son navigateur, une requête est envoyée au serveur web qui répond en retournant une page HTML, des fichiers CSS, JavaScript, des images ou d'autres contenus.

Les serveurs web constituent un élément central de l'architecture des systèmes connectés, car ils permettent la mise à disposition des applications et des services en ligne .

I.2 Apache et Nginx

Apache et Nginx sont deux des serveurs web les plus utilisés dans le monde. Apache est connu pour sa flexibilité, sa grande compatibilité et la richesse de ses modules. Il est largement utilisé dans les environnements académiques et professionnels. Nginx, de son côté, est réputé pour ses hautes performances et sa capacité à gérer un grand nombre de connexions simultanées avec une faible consommation de ressources.

Dans le cadre de ce projet, le choix s'est porté sur Apache, installé via la plateforme XAMPP, afin de simplifier le déploiement dans un environnement Windows .

I.3 HTTP et HTTPS

HTTP (HyperText Transfer Protocol) est le protocole de base utilisé pour l'échange de données sur le web. Il permet au navigateur et au serveur de communiquer. Cependant, HTTP transmet les données en clair, ce qui signifie qu'elles peuvent être interceptées par un tiers malveillant sur le réseau.

HTTPS est la version sécurisée de HTTP. Il ajoute une couche de protection grâce au chiffrement. Avec HTTPS, les données échangées entre le client et le serveur sont protégées contre l'interception et la modification.

I.4 TLS (Transport Layer Security)

TLS (Transport Layer Security) est le protocole de sécurité qui permet de chiffrer la communication entre un client et un serveur. Il garantit trois éléments principaux : la confidentialité des données, l'intégrité des messages et l'authentification du serveur via un certificat numérique.

Lorsqu'un utilisateur accède à un site en HTTPS, un processus appelé "handshake TLS" s'exécute en arrière-plan pour établir une connexion sécurisée. Dans un environnement local de test, on utilise souvent un certificat auto-signé, qui active le chiffrement mais provoque un avertissement du navigateur, car il n'est pas délivré par une autorité de certification reconnue .

III. ENVIRONNEMENT DE TRAVAIL

Pour la réalisation de ce projet, nous avons mis en place un environnement de travail local permettant d'installer, configurer et tester un serveur web sécurisé. Le choix d'un environnement local permet de faire les configurations sans dépendre d'un hébergement externe, tout en gardant un contrôle total sur le système.

Le système d'exploitation utilisé est Windows, qui a servi de plateforme principale pour l'installation et l'exécution du serveur web. Afin de simplifier le déploiement d'Apache et de ses composants, nous avons utilisé XAMPP, une distribution logicielle qui regroupe Apache, PHP et d'autres outils nécessaires au développement et aux tests de serveurs web en local².

Le serveur web utilisé dans le projet est Apache HTTP Server, démarré et administré à l'aide du panneau de contrôle XAMPP. Les configurations ont été réalisées en modifiant les fichiers de paramètres d'Apache, notamment pour activer le module SSL/TLS.

Les tests d'accès au serveur ont été effectués à l'aide d'un navigateur web moderne (Google Chrome / Microsoft Edge), en utilisant les adresses locales <http://localhost/projetSE> et <https://localhost/projetSE>. Le site de démonstration a été placé dans le dossier `htdocs` de XAMPP, qui représente le répertoire racine des fichiers servis par Apache.

L'édition des fichiers HTML et CSS du site de test a été faite avec un éditeur de code standard. Le site utilise également le framework Bootstrap via CDN pour la mise en forme de l'interface .

² Source : <https://www.apachefriends.org>

IV. INSTALATION DU SERVEUR WEB

L'installation du serveur web a été réalisée à l'aide de la plateforme XAMPP, qui fournit un environnement prêt à l'emploi pour déployer Apache sur le système d'exploitation Windows. Ce choix a été fait afin de simplifier la procédure d'installation et de réduire les risques d'erreurs de configuration manuelle. La première étape a consisté à télécharger le programme d'installation de XAMPP pour Windows depuis le site officiel <https://apachefriends.org>. Après le téléchargement, l'installation a été lancée avec les options par défaut, en conservant les composants principaux, notamment Apache. Une fois l'installation terminée, le panneau de contrôle XAMPP a été ouvert pour gérer les services. À partir du panneau de contrôle, le module Apache a été démarré en cliquant sur le bouton "Start". Lorsque le service est actif, son indicateur devient vert, ce qui confirme que le serveur web est en cours d'exécution sur la machine locale.



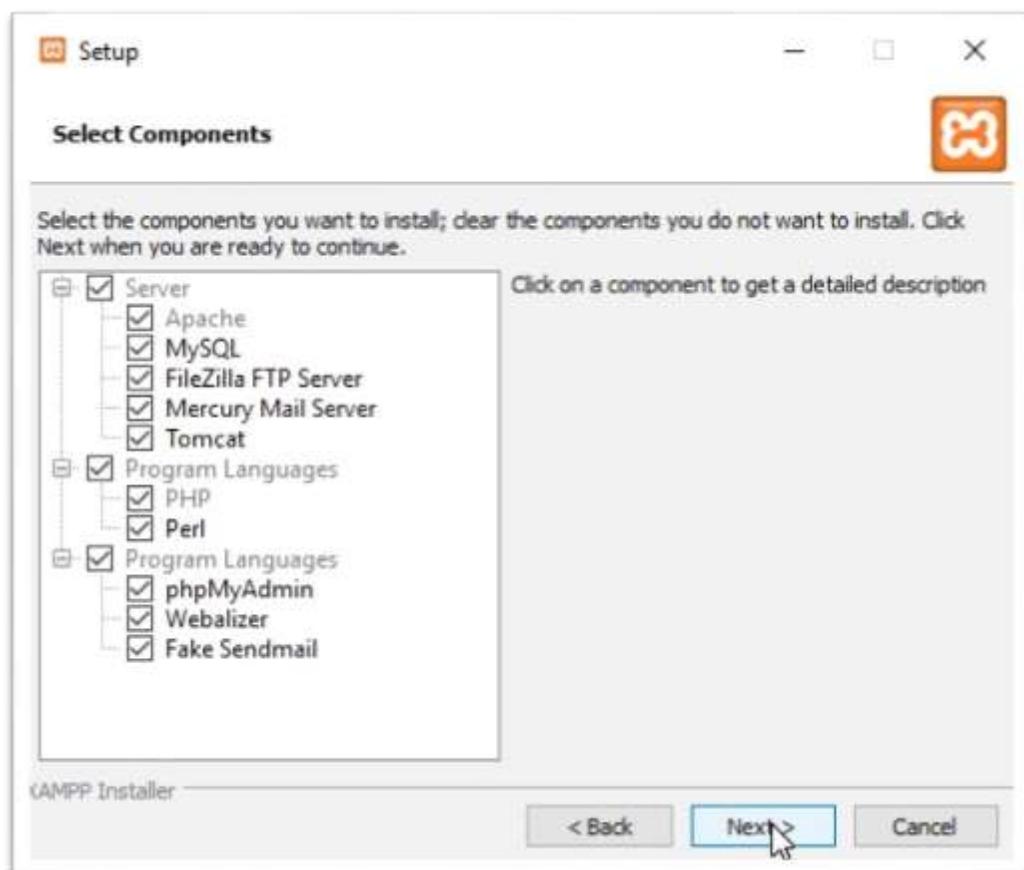
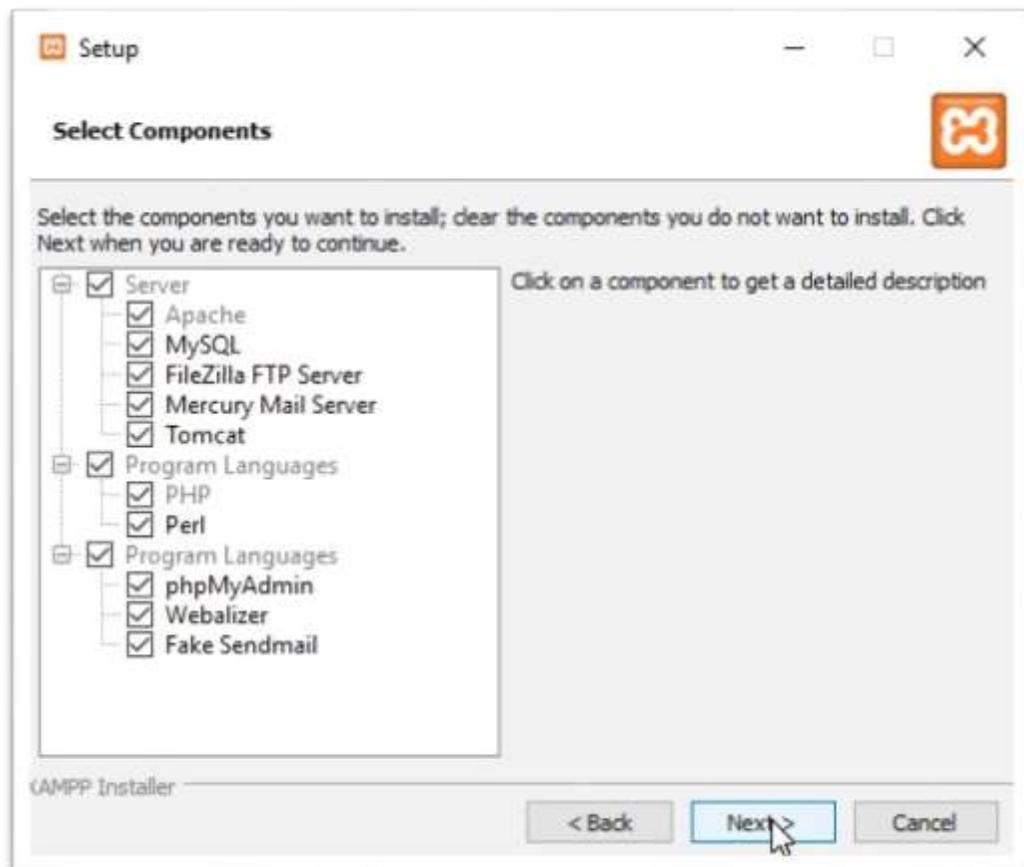
Après le démarrage d'Apache, un test a été effectué dans le navigateur en accédant à l'adresse :

<http://localhost/projetSE>



L'affichage de la page d'accueil XAMPP a confirmé que le serveur web était correctement installé et opérationnel. Cette étape a permis de vérifier que le service écoute correctement sur le port HTTP et répond aux requêtes locales. Des captures d'écran ont été réalisées montrant l'instalation de XAMPP et le panneau de contrôle avec Apache actif.





V. ACTIVATION HTTPS /TLS

Après l'installation et le démarrage du serveur Apache, l'étape suivante a consisté à activer la communication sécurisée via HTTPS. Pour cela, il est nécessaire d'activer le module SSL/TLS d'Apache et de charger le fichier de configuration dédié à la sécurité.

La configuration a été réalisée en modifiant le fichier principal des paramètres d'Apache situé dans le dossier de configuration de XAMPP. Ce fichier contient les modules que le serveur doit charger au démarrage. Nous avons recherché la ligne correspondant au module SSL et activé son chargement en supprimant le caractère de commentaire placé au début de la ligne. La même opération a été effectuée pour inclure le fichier de configuration SSL supplémentaire.

Captures :



```
# Configuration de base
User Apache
Group Apache
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule sed_module modules/mod_sed.so
LoadModule session_module modules/mod_session.so
LoadModule session_cookie_module modules/mod_session_cookie.so
LoadModule session_crypto_module modules/mod_session_crypto.so
LoadModule session_dbd_module modules/mod_session_dbd.so
LoadModule setenvif_module modules/mod_setenvif.so
LoadModule slotmem_plain_module modules/mod_slotmem_plain.so
LoadModule slotmem_shm_module modules/mod_slotmem_shm.so
LoadModule socache_shm_module modules/mod_socache_shm.so
LoadModule socache_memcache_module modules/mod_socache_memcache.so
LoadModule socache_redis_module modules/mod_socache_redis.so
LoadModule socache_stab_module modules/mod_socache_stab.so
LoadModule spelling_module modules/mod_spelling.so
LoadModule status_module modules/mod_status.so
LoadModule substitute_module modules/mod_substitute.so
LoadModule env_module modules/mod_env.so
```




```
# Configuration de base
User Apache
Group Apache
# Various default settings
Include conf/extra/httpd-default.conf
# Implements a proxy/gateway for Apache.
#include "conf/extra/httpd-proxy.conf"
# Various default settings
#include "conf/extra/httpd-default.conf"
# XAMPP settings
#include "conf/extra/httpd-xampp.conf"

# Configure mod_proxy_html to understand HTML4/HTML5
#
#  include conf/extra/proxy-html.conf
#

# Secure (SSL/TLS) connections
#Include conf/extra/httpd-ssl.conf
#
# Note: The following must be present to support
```

Après modification et enregistrement du fichier de configuration, le serveur Apache a été redémarré depuis le panneau de contrôle XAMPP afin d'appliquer les changements. Ce redémarrage est nécessaire pour que les nouveaux modules activés soient pris en compte.

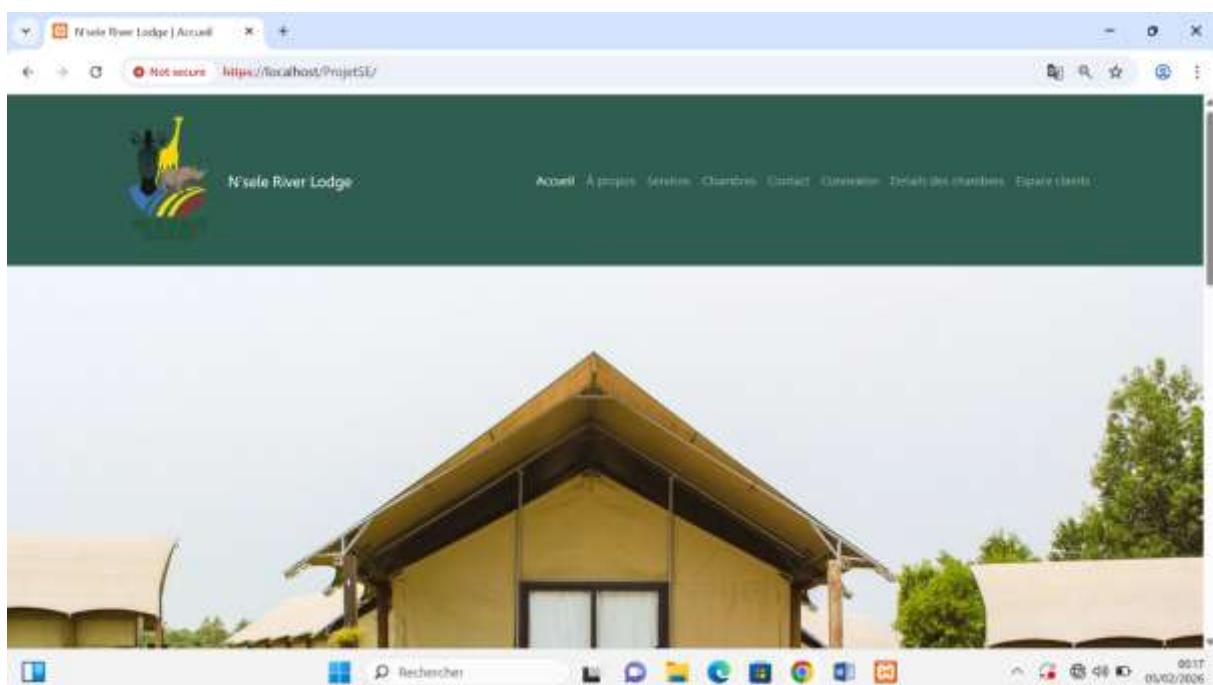
Une fois la configuration appliquée, un test a été effectué dans le navigateur en accédant à l'adresse :

<https://localhost/projetSE>

Le navigateur a affiché un avertissement de sécurité. Ce comportement est normal dans un environnement local, car le certificat utilisé est un certificat auto-signé. Cela signifie qu'il n'est pas délivré par une autorité de certification publique, mais généré localement pour les besoins de test. Malgré l'avertissement, la connexion est bien chiffrée et le protocole TLS est actif.

Cette étape confirme que le serveur web est désormais capable de servir des pages via HTTPS et que le chiffrement des communications est opérationnel.

Captures :



Bienvenue à N'sele River Lodge

Un cadre naturel et confortable pour vos séjours touristiques et de détente.

Réserver maintenant

Nos services

Hébergement

Restoration

Vue du parc

Découverte des animaux et de la nature.

VI. CONCLUSION

Ce projet a permis de mettre en pratique les notions étudiées dans le cours de Systèmes d'Exploitation , en réalisant l'installation et la configuration d'un serveur web Apache sécurisé avec HTTPS. Grâce à l'utilisation de XAMPP, nous avons pu installer rapidement le serveur, activer le module SSL/TLS et déployer un site web de test sur un environnement local.

Le projet a permis de comprendre l'importance de la sécurité dans les échanges web. L'utilisation de TLS assure le chiffrement des communications, protège les données échangées et garantit la confidentialité et l'intégrité des informations. Même si un certificat auto-signé génère un avertissement dans le navigateur, la connexion reste chiffrée et sécurisée.

Enfin, le déploiement du site de test avec Bootstrap a illustré la manière dont un serveur web distribue les fichiers et comment les ressources externes peuvent être intégrées pour enrichir l'interface. Ce travail a donc permis d'allier la théorie à la pratique et de se familiariser avec les étapes réelles de mise en place d'un serveur web sécurisé.