

BTS SIO SLAM

INSTALLATION ET CONFIGURATION D'UN CMS

Déploiement d'un LAMP pour site vitrine GSB

Établissement	Aristée (ccour.aristeecampus.org/GSB)
Période	Septembre 2025
Technologie	Debian GNU/Linux / Apache2 / MySQL (MariaDB) / PHP / WordPress
Candidat	COUR Cyrille

SOMMAIRE

1. Contexte du Projet et Environnement d'Infrastructure (GSB / Aristée)	3
2. Initialisation du Serveur Système et Déploiement de la Pile LAMP	3
3. Configuration de la Base de Données Relationnelle	4
4. Déploiement, Droits d'Accès et Hôtes Virtuels Apache	4
5. Initialisation Logicielles et Finalisation du CMS	5

1. Contexte du Projet et Environnement d'Infrastructure (GSB / Aristée)

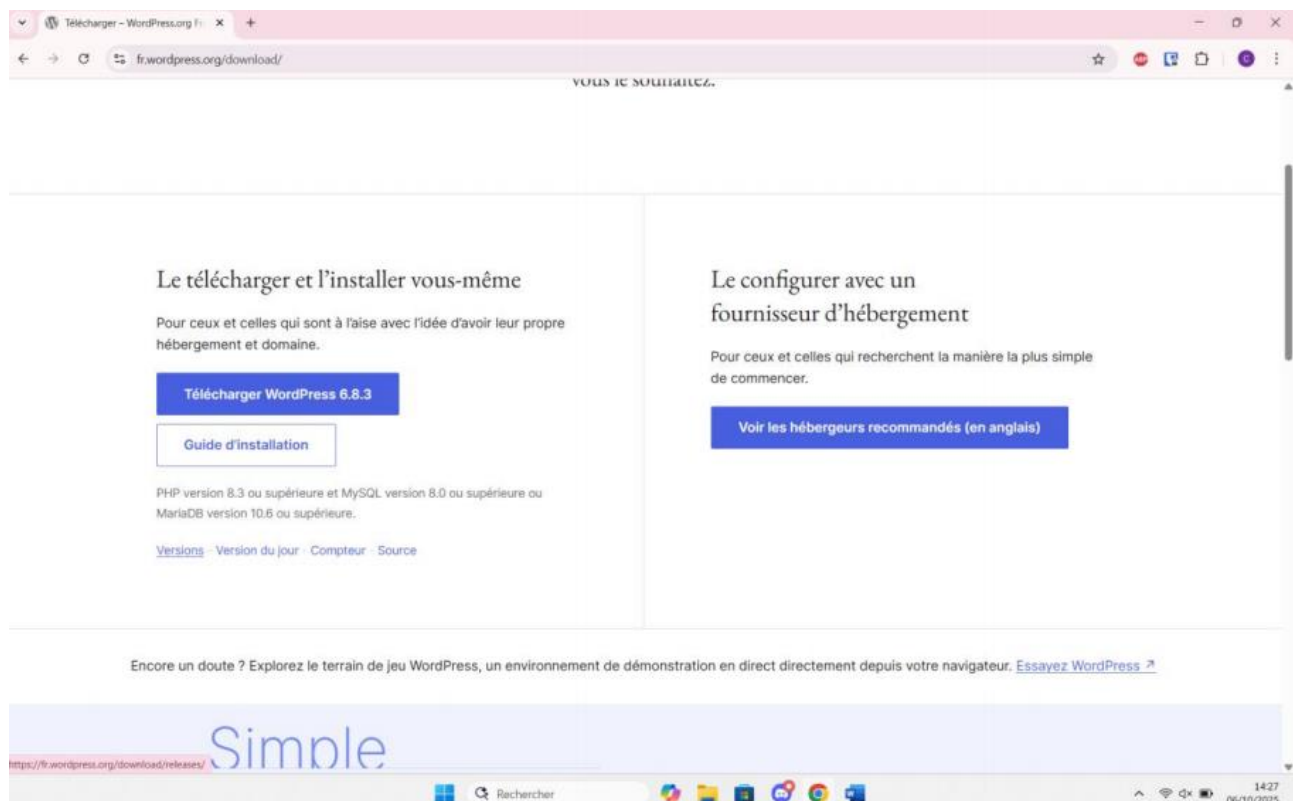
Dans le cadre des activités pratiques du BTS SIO contextualisées autour de l'organisation fictive GSB (Galaxy Swiss Bourdin), j'ai mis en place l'environnement technique nécessaire à l'hébergement du site vitrine de l'entreprise. Ce site web grand public doit présenter les services de l'organisation et est accessible à l'adresse réseau : ccour.aristeecampus.org/GSB.

Afin de garantir le contrôle total de la chaîne d'hébergement, de la sécurité du système de fichiers à la persistance des données, j'ai réalisé l'installation native d'un système de gestion de contenu (CMS) WordPress sur un serveur virtuel dédié, hébergé au sein de l'infrastructure d'Aristée.

2. Initialisation du Serveur Système et Déploiement de la Pile LAMP

Le déploiement repose sur une machine virtuelle vierge exploitant la distribution Linux Debian (sans interface graphique pour maximiser les performances et restreindre la surface d'attaque). L'infrastructure applicative utilise le socle classique LAMP :

- Mise à jour des dépôts système et des paquets logiciels : `apt update && apt upgrade -y`
- Installation du serveur HTTP Apache2 : `apt install apache2 -y`
- Installation du serveur de base de données MariaDB (MySQL) : `apt install mariadb-server -y`
- Installation de l'interpréteur PHP et de ses modules requis par le CMS (php-mysql, php-xml, php-gd...) : `apt install php libapache2-mod-php php-mysql -y`



3. Configuration de la Base de Données Relationnelle

WordPress requiert une base de données relationnelle structurée pour stocker son modèle logique (articles, pages, métadonnées, comptes utilisateurs). L'accès au SGBD a été sécurisé manuellement :

- Connexion à l'instance locale en tant que superutilisateur : `mysql -u root`
- Création de la base de données dédiée : `CREATE DATABASE gsb_wp_db;`
- Création d'un utilisateur utilisateur réseau dédié sans privilèges globaux : `CREATE USER 'gsb_admin'@'localhost' IDENTIFIED BY 'MotDePasseSecurise';`
- Attribution des privilèges d'écriture et de lecture sur la seule base GSB : `GRANT ALL PRIVILEGES ON gsb_wp_db.* TO 'gsb_admin'@'localhost';`
- Rechargement de la table des droits et déconnexion : `FLUSH PRIVILEGES; EXIT;`

❑ **Sécurité SGBD** : L'utilisation d'un compte utilisateur dédié ('gsb_admin') restreint à sa seule base de données évite qu'une faille applicative potentielle sur le CMS ne compromette l'intégralité du serveur SQL.

4. Déploiement, Droits d'Accès et Hôtes Virtuels Apache

Une fois les archives de WordPress extraites, les fichiers sources ont été déplacés vers le répertoire de production d'Apache : `/var/www/html/`.

Afin de permettre au serveur web d'écrire de manière autonome (pour l'importation de médias ou les mises à jour d'extensions), la gestion des permissions Linux a été configurée :

- Changement du propriétaire du dossier pour le processus Apache : `chown -R www-data:www-data /var/www/html/`
- Configuration des droits d'accès standardisés (755 pour les dossiers, 644 pour les fichiers).

Pour la configuration réseau de l'URL, un fichier d'hôte virtuel (VirtualHost) a été créé dans `/etc/apache2/sites-available/gsb.conf` pour mapper le nom de domaine complet. L'activation se fait par l'intermédiaire d'un lien symbolique :

- Création du lien symbolique d'activation : `a2ensite gsb.conf`
- Rechargement du démon de configuration réseau d'Apache : `systemctl reload apache2`



debian

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

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

5. Initialisation Logicielle et Finalisation du CMS

La dernière phase s'effectue depuis un navigateur web. En ciblant l'URL publique, l'assistant d'installation WordPress prend le relais pour générer le fichier de configuration persistant `wp-config.php` :

- Saisie des identifiants d'accès SQL préalablement définis (Nom de la BDD, Identifiant, Mot de passe, Hôte).
- En cas de restriction de droits en écriture sur le répertoire, création manuelle du fichier : `nano wp-config.php` et inclusion du jeton généré.
- Création du premier compte d'administration applicatif et déploiement final du site vitrine GSB.



Bienvenue sur WordPress. Avant de commencer, vous aurez besoin de connaître les éléments suivants.

1. Nom de la base de données
2. Identifiant MySQL
3. Mot de passe de base de données
4. Hôte de base de données
5. Préfixe de table (si vous souhaitez avoir plusieurs WordPress sur une même base de données)

Ces informations sont utilisées pour créer un fichier `wp-config.php`. **Si pour une raison ou pour une autre la création automatique du fichier ne fonctionne pas, ne vous inquiétez pas. Sa seule action est d'ajouter les informations de la base de données dans un fichier de configuration. Vous pouvez aussi simplement ouvrir `wp-config-sample.php` dans un éditeur de texte, y remplir vos informations et l'enregistrer sous le nom de `wp-config.php`.** Besoin d'une aide complémentaire ? [Lisez l'article du support sur wp-config.php](#).

Vous devriez normalement avoir reçu ces informations de la part de votre hébergeur. Si vous ne les avez pas, il vous faudra contacter votre hébergeur afin de continuer. Si vous avez tout le nécessaire, alors...

C'est parti !

□ **Note d'intégration BTS SIO :** Cette procédure valide la compétence d'administration système liée à la mise en production d'une application web, à la sécurisation des flux locaux, et à la gestion d'un serveur d'infrastructure de type LAMP.