



Objectif de la séquence

L'objectif de la séquence est de créer et maintenir un réseau d'entreprise dans le cloud.

Vous êtes administrateur réseau.

Votre entreprise souhaite avoir un premier retour d'expérience sur l'installation et la maintenance d'un de ces services dans le cloud.

Elle décide d'externaliser dans le cloud AWS son site web institutionnel. Celui-ci est basé sur la solution WordPress. Il vous est demandé de vous former, d'installer le CMS WordPress et d'organiser la sauvegarde de ce CMS (web et bdd)

Vous devez mettre en place le réseau de l'entreprise composé d'un serveur web et d'un serveur de base de données.

Vous devrez également mettre en place la politique de sauvegarde de l'entreprise en cas de défaillance matérielle et/ou d'attaque sur ce réseau.

Cela comprend :

- La sauvegarde régulière des informations de la base de données
- La sauvegarde des données du CMS

La séquence est découpée en plusieurs modules :

- Activité 1 – Se former au cloud, cas AWS
- Activité 2 - Créer l'infrastructure du SI dans le cloud AWS
- Activité 3 - Mise en place de la politique de sauvegarde et de restauration du réseau

Durée

- 1 séance de 5h

Livrable

- Votre compte rendu avec les résultats attendus et les réponses aux questions



I. Système d'information cible

Voici ci-dessous le système d'information que vous allez créer.

Votre VPC sera réparti dans deux "availability zone".

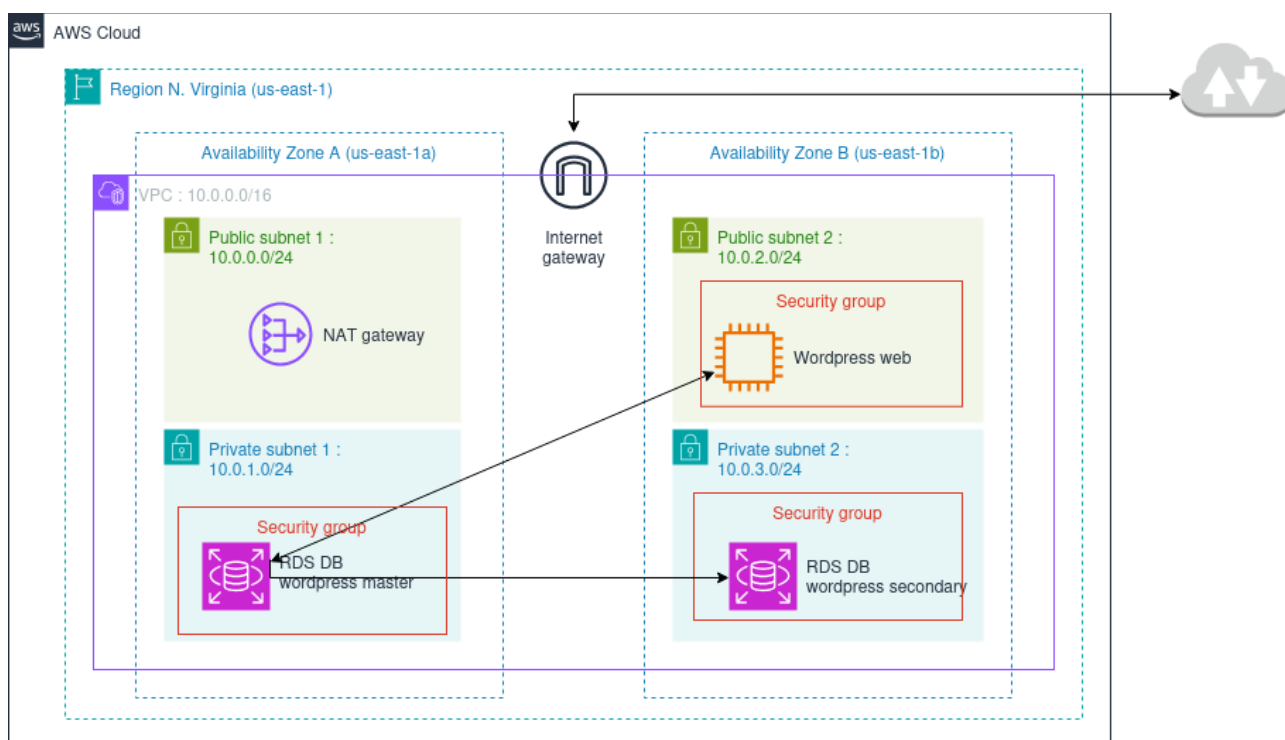
Le CMS à installer est WordPress.

Vous pouvez trouver les prérequis sur le site officiel.

requirement : <https://fr.WordPress.org/about/requirements/>

Nous allons séparer la partie applicative et la partie stockage de données sur deux serveurs différents.

- Le serveur applicatif sera installé sur une VM EC2 Debian 13
- Pour la partie stockage, nous utiliserons le service RDS d'AWS avec une base de données MySQL





II. Création de l'architecture du réseau d'entreprise, le VPC

Maintenant que vous avez suivi avec succès les modules AWS, vous allez créer l'architecture réseau cible.

Pour cela, vous allez utiliser le deuxième module AWS mis à votre disposition sur AWS academy, le learner lab.

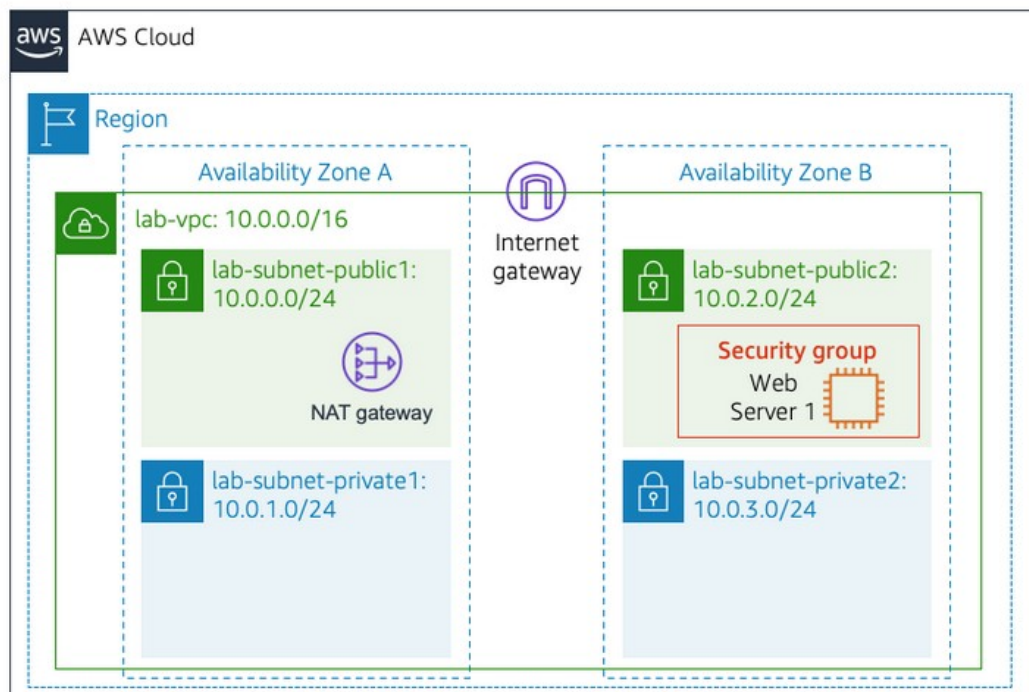
<https://awsacademy.instructure.com/login/canvas>

Le learner lab est un environnement de type sandbox pour explorer à votre guise les services d'AWS. Le budget est limité à 50\$, mais vous conserverez l'accès aux ressources AWS configuré dans votre environnement pendant la durée du lab, soit toute la séquence.

En vous aidant des connaissances du "Module AWS 5 : mise en réseau et diffusion de contenu",

Tâche1. Créer le VPC correspondant au schéma ci-après.

À la fin du lab, vous devez obtenir le schéma réseau suivant :





REMARQUES

Suivre globalement les instructions du labo.

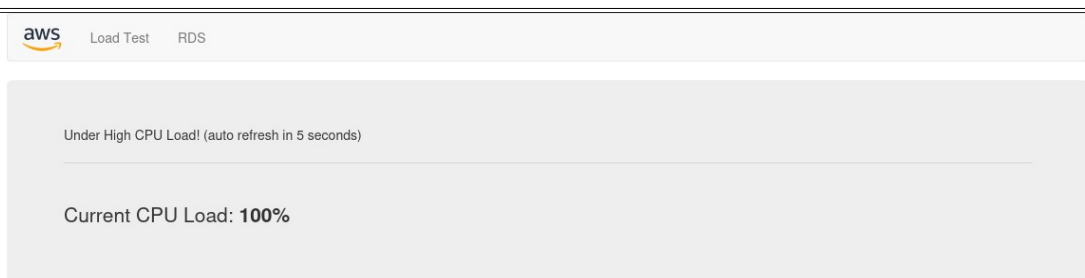
- Veillez cependant à donner un nom cohérent à votre VPC (sup-cloud par exemple)

Le lab vous fait créer une instance EC2 dans la tâche 5.

- Même si celle-ci ne correspond pas à notre cible, elle est utile pour vérifier votre travail.

Résultats attendus

- Vérifier bien que vous respectez la configuration du VPC de la figure précédente (nombre et adresses des sous-réseaux, table de routage publique et privée, groupe de sécurité)
- Si tout est bien configuré, vous devez avoir la page web correspondante au labo qui s'affiche correctement dans votre navigateur.





III. Le CMS WordPress

Le CMS WordPress est relativement facile à installer.

Le site officiel détaille la procédure d'installation

Comment installer WordPress : <https://fr.WordPress.org/support/article/how-to-install-WordPress/>

Ceci dit, dans un but de clarification, nous allons résumer les grandes étapes de l'installation.

WordPress est un CMS écrit principalement en PHP et repose sur une base de données (MySQL ou MariaDB)

- Vous allez donc avoir besoin d'un serveur pour héberger la partie applicative. C'est pour cela que vous allez installer un EC2 avec la distribution linux Debian 13.
- Sur cet EC2, il sera nécessaire d'installer un serveur Web et les modules php associés. Comme vous connaissez déjà Apache, nous allons conserver cette solution.
- Vous allez également avoir besoin d'un serveur de base de données. Vous allez utiliser le service RDS d'AWS. Comme précisé sur l'architecture cible, le serveur RDS et son replica seront dans la partie sous réseau privé pour des raisons de sécurité.

Pour la suite du module, nous allons devoir installer :

- une instance EC2 avec un serveur web Apache et les extensions php
- une instance RDS pour la base de données

III.1 Le serveur web

La mise en production du serveur web comprend les étapes suivantes :

- Provisionner l'instance EC2 dans AWS
- Installer Apache et les dépendances PHP
- Copier l'archive de WordPress sur l'instance EC2

III.1.1 Provisionner l'instance EC2

Vous pouvez détruire l'instance créée avec le VPC de la tâche précédente, nous allons utiliser une VM debian 13 dans la suite du module.



En vous aidant des connaissances du "Module 5 : mise en réseau et diffusion de contenu" et du "Module 6.1 : calcul"

Tâche2. Créer une instance EC2 dans le sous-réseau ***-public2 avec les paramètres suivants

Nom de l'instance	user WordPress web server
Type AMI	debian 13 x86_64
type d'instance	t2.micro
paire de clés	vockey
paramètres réseau	votre vpc (sup-cloud-vpc)
sous réseau	***-public2
IP attribuée automatiquement	
Sélectionner groupe de sécurité existant	web security group

- Lancer l'instance

Une fois que l'instance est provisionnée, vous pouvez passer à l'étape suivante.

Résultats attendus

- Une instance de type debian13, t2.micro est provisionnée
- L'instance est dans le VPC sup-cloud et le sous-réseau public2
- L'instance est en cours d'exécution



III.1.2 Connexion SSH

Avant de pouvoir installer Apache, vous allez devoir autoriser l'accès ssh à votre VM et vous connecter en ssh sur celle-ci.

CONFIGURATION CONNEXION SSH

Pour se connecter en ssh sur votre EC2, vous avez besoin :

- De la clé ssh vockey (learner Lab, onglet « AWS details » → Download PEM)
- De l'adresse IP publique de l'instance (Instance → DNS ou IPv4 publique)
- D'autoriser les connexions ssh sur l'instance (groupe de sécurité web, inbound port 22 TCP)
- De sécuriser la clé vockey (`chmod 600 labuser.pem`)

Tâche3. Se connecter au serveur EC2 (Sur votre PC, dans le répertoire qui contient la clé)

```
ssh -i labsuser.pem admin@ec2-54-80-140-35.compute-1.amazonaws.com
```

Résultats attendus

- La connexion ssh est opérationnelle

III.1.3 Installation d'Apache

Bien que vous ayez déjà installé Apache dans d'autres séquences, voici ci-dessous le rappel des étapes de configuration

- Installer Apache

```
sudo apt update & apt upgrade  
sudo apt install apache2
```

- Redémarrer apache2

```
sudo systemctl restart apache2
```

Tâche4. Installer apache2

Résultats attendus

- La page par défaut d'Apache est accessible sur l'IP publique de la VM

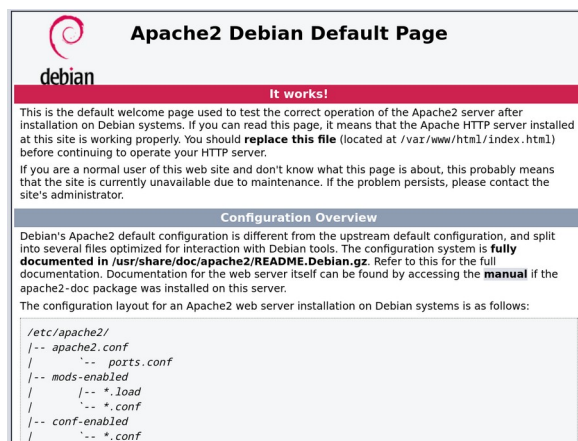


BTS CIEL_IR



Installer et maintenir un SI cloud

Activité 2



III.1.4 Installation de php

Maintenant que Apache est installé, il faut installer les extensions PHP

Ci-dessous le rappel des étapes de configuration

```
sudo apt install php libapache2-mod-php php-mysql
```

Pour vérifier que PHP est correctement installé et configuré, créer un fichier PHP de test dans le répertoire racine web d'Apache (/var/www/html par défaut)

Fichier /var/www/html/phpinfo.php

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

- Redémarrer apache2

```
sudo systemctl restart apache2
```

Tâche5. Installer PHP

Résultats attendus

- La page d'information de PHP est accessible à l'adresse <http://@IP/info.php>

**PHP Version 8.4.21**

System	Linux ip-10-0-2-183 6.12.90+deb13-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.12.90-1 (2026-05-22) x86_64
Build Date	May 8 2026 05:56:48
Build System	Linux
Build Provider	Debian
Server API	Apache 2 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.4/apache2
Loaded Configuration File	/etc/php/8.4/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.4/apache2/conf.d

Q1 : Vérifier que la version de PHP correspond aux prérequis de WordPress

III.1.5 Mise en place du Virtual host

Le serveur web est en place. Il faut maintenant installer sur celui-ci la dernière version de WordPress et un virtualHost pour l'héberger.

Ci-dessous les étapes de configuration

- Récupérer la dernière version de WordPress (https://fr.WordPress.org/latest-fr_FR.zip)
- Décompresser cette version dans le répertoire `/var/www/wordpress`
- Mettre en place un virtualHost pour héberger WordPress
- fichier `/etc/apache2/sites-available/wordpress.conf`

```
<VirtualHost *:80>
    ServerAdmin webmaster@mondomaine.com
    DocumentRoot "/var/www/wordpress"
    ServerName mondomaine.com
    ServerAlias www.mondomaine.com

    <Directory "/var/www/wordpress">
        Options Indexes FollowSymLinks
        AllowOverride None
        Require all granted
    <FilesMatch \.php$>
        SetHandler application/x-httpd-php
    </FilesMatch>
</Directory>
</VirtualHost>
```



```
</FilesMatch>
</Directory>

ErrorLog ${APACHE_LOG_DIR}/wordpress_error.log
CustomLog ${APACHE_LOG_DIR}/wordpress_access.log combined
</VirtualHost>
```

- Activer le virtual host

```
sudo a2dissite 000-default
sudo a2ensite WordPress.conf
```

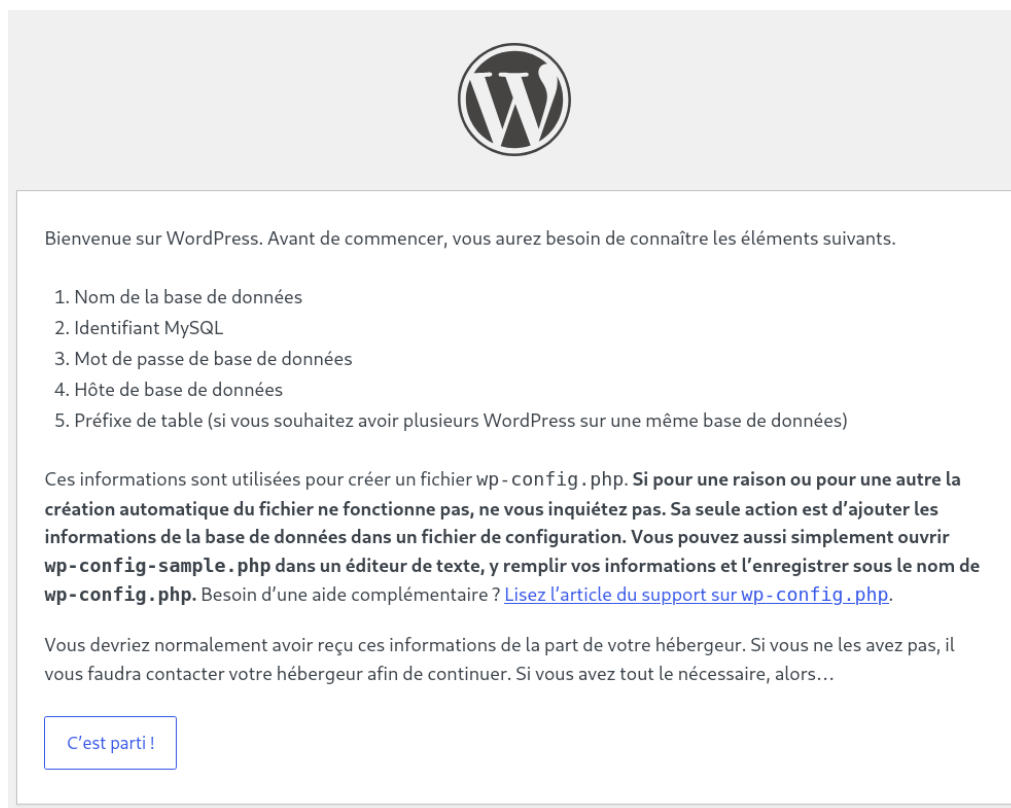
- Redémarrer apache2

```
sudo systemctl restart apache2
```

Tâche6. Installer l'application WordPress sur la VM EC2.

Résultats attendus

- La page d'accueil de WordPress est accessible sur l'IP publique de la VM





III.2 Le serveur de base de données

Maintenant que votre serveur web est opérationnel, du moins pour la partie html/php, nous allons terminer l'installation du réseau de l'entreprise.

Pour cela, il faut :

- Installer un serveur RDS dans la zone privée
- Créer la base de données sur ce serveur
- Configurer le site pour utiliser le serveur de base de données (adresse, identifiant et bdd)

III.2.1 Installer un serveur RDS dans la zone privée

S'aider des connaissances du "Module AWS 8 : bases de données"

Il faut globalement suivre les instructions du labo, mais veuillez bien à sélectionner les éléments suivants :

- Choisir maria-db
- pour l'identifiant d'instance, choisir sup-cloud-db
- **Retenez le compte administrateur et le password, vous en aurez besoin pour vous connecter à votre serveur**
- Choisir le bon VPC (sup-cloud-vpc)
- Choisir le subnet-group créé à la tâche 2 du lab

Tâche7. Créer une instance RDS dans le sous réseau ***-private1 avec les paramètres suivants

Résultats attendus

- Les serveurs de base de données (master et replica) s'exécutent dans les subnet private

III.2.2 Configurer la base de donnée

Maintenant que le serveur est opérationnel, il faut configurer le serveur de base de données.



Se connecter au RDS

Grâce au groupe de sécurité, le serveur est accessible depuis la VM EC2.

Nous allons utiliser le client mysql pour se connecter dessus et faire les opérations de création (utilisateur et base de données WordPress_db)

Ci-dessous les étapes de configuration

- Se connecter en ssh à l'instance EC2
- Installer les outils debian client de connexion à la BDD

```
sudo apt install mariadb-client
```

- Récupérer le certificat global-bundle.pem

```
wget https://truststore.pki.rds.amazonaws.com/global/global-bundle.pem
```

Pour la connexion, vous avez besoin :

- du endpoint de la bdd (exemple: sup-cloud-db.cm6gwipyisdn.us-east-1.rds.amazonaws.com)
- du nom de l'administrateur (exemple main)
- du mot de passe administrateur (exemple lab-password)
- Exemple de ligne de commande à adapter depuis votre serveur web

```
mysql -h lab-db.cqvq3pgxwbua.us-east-1.rds.amazonaws.com -u main -p \\\n--ssl-ca=global-bundle.pem --ssl-verify-server-cert
```

Tâche8. Se connecter à votre serveur RDS avec le client mysql

Résultats attendus

- Vous êtes connecté au serveur RDS

Opérations de création MySQL

Les opérations de création d'utilisateurs et de la base de données ont déjà été abordés dans une séquence précédente. Nous rappelons simplement ci-après les différentes commandes nécessaires pour la configuration.

- Créer un utilisateur ayant les droits d'accès depuis un serveur externe

```
CREATE USER 'ciel1'@'%' IDENTIFIED BY 'Ciel1_mdp';
```



- Créer une base de données pour nos utilisateurs

```
CREATE DATABASE wordpress_db;
```

- Donner les droits à l'utilisateur ciell sur la base de données

```
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, ALTER, INDEX ON  
WordPress_db.* TO 'ciell'@'%';
```

- Recharger les privilèges

```
FLUSH PRIVILEGES;
```

Tâche9. Exécuter les commandes de création d'utilisateurs, de bases de données et de tables ci-dessus

Résultats attendus

- L'utilisateur est créé avec les droits nécessaires sur la base de donnée wordpress_db

```
admin@ip-10-0-2-183: ~  
admin@ip-10-0-2-183:~$ mysql -h lab-db.cqvq3pgxwbua.us-east-1.rds.amazonaws.com -u main -p --ssl-ca=global-  
bundle.pem --ssl-verify-server-cert  
Enter password:  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MySQL connection id is 12  
Server version: 8.4.8 Source distribution  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
MySQL [(none)]> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| lab |  
| mysql |  
| performance_schema |  
| sys |  
| wordpress_db |  
+-----+  
6 rows in set (0.011 sec)  
  
MySQL [(none)]>
```



III.3 Connexion de l'application à la BDD

Maintenant que l'instance RDS est installée et configurée, il faut donner à l'application les informations nécessaires pour interagir avec la BDD.

Depuis l'interface de WordPress, configurer les données suivantes :

Vous devez saisir ci-dessous les détails de connexion à votre base de données. Si vous ne les connaissez pas, contactez votre hébergeur.

Nom de la base de données	wordpress_db2	Le nom de la base de données avec laquelle vous souhaitez utiliser WordPress.
Identifiant	ciel2	Votre identifiant MySQL.
Mot de passe	••••••••	Afficher Votre mot de passe de base de données.
Adresse de la base de données	lab-db.cqvq3pgxwbua.us-east-1.rds.amazonaws.com	Si localhost ne fonctionne pas, demandez cette information à l'hébergeur de votre site.
Préfixe des tables	wp_	Si vous souhaitez faire tourner plusieurs installations de WordPress sur une même base de données, modifiez ce réglage.

[Envoyer](#)

Tâche10. Configurer WordPress pour vous connecter à la BDD

Résultat attendu

- Le site web est opérationnel.



Blog

Bonjour tout le monde !

Bienvenue sur WordPress. Ceci est votre premier article. Modifiez-le ou supprimez-le, puis commencez à écrire !

26 mai 2026

III.4 Sécurisation HTTPS

Pour terminer cette séance, vous allez maintenant sécuriser votre serveur en https grâce à un certificat fourni par Let's encrypt.

Pour cela, il va falloir modifier votre configuration selon ce schéma général.

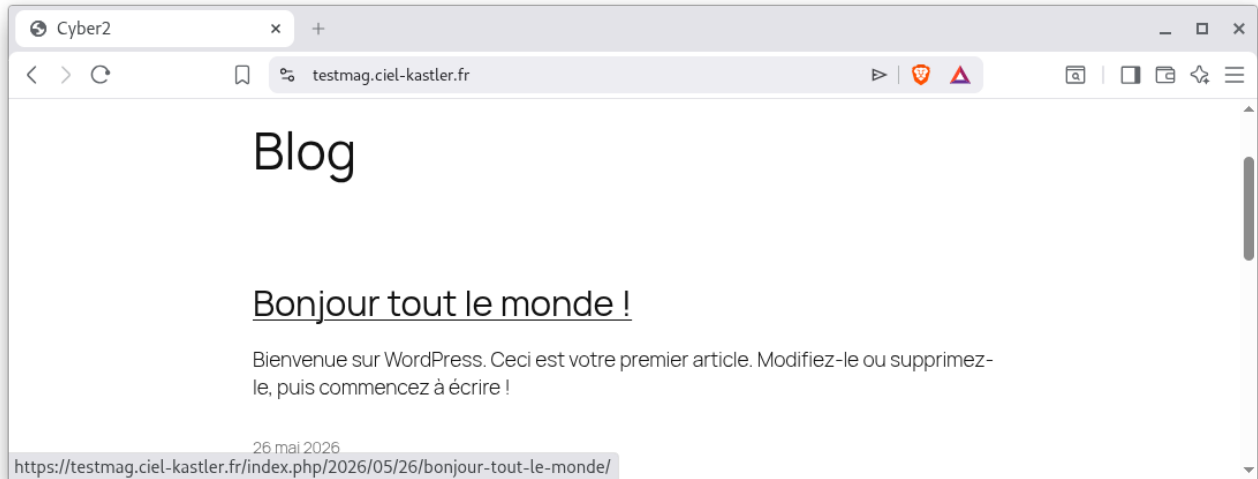
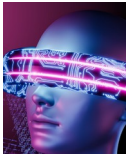
- Ajouter une règle de sécurité pour autoriser le trafic https vers WordPress
- Créer une "elastic IP" et l'attribuer à l'EC2 WordPress
- Ajouter une entrée DNS pour associer l'IP elastic à un nom de domaine (fourni par l'enseignant)
- Installer certbot sur le serveur et demander un certificat

```
sudo apt install certbot python3-certbot-apache -y  
sudo certbot --apache
```

Tâche11. Configurer la connexion https au site WordPress

Résultat attendu

- Le site web est opérationnel en https sur le nom de domaine choisi



IV. Conclusion

Bravo, vous avez déployé avec succès un site WordPress dans le cloud en utilisant AWS ! AWS incite fortement à la sécurité des applications, notamment en installant les données dans un sous réseau privé et en imposant d'avoir deux "availability zone" pour installer un RDS.

Nous avons aussi sécurisé la connexion au serveur web grâce à un certificat Let's encrypt.