

Création des instances EC2, de la base de données RDS et de bucket S3 sur le cloud aws

Procédure rédigée par : CHERIF Ahmed

Table des matières

I - Les instances et la base de données	2
I.1 – Création et lancement d’une instance EC2 dans le sous réseau public	2
I.2 – Lancer une base de données RDS dans le sous réseau privé	2
I.3 – Connecter à la base de données depuis mon instance EC2	6
I.4 – 4-\ Installer wordpress	6
II - Création d’un conteneur sur une instance EC2	9
II.1 - Le stockage des fichiers sur S3	9
II.2 - Envois des fichiers de sites vers mon instances EC2	11
II.3 - Restauration de la base de données	11
II.4 - création d’un dockerfile pour une image wordpress	12
II.5 - création d’un docker-compose.....	13

I-\ Les instances et la base de données

1-\ Création et lancement d'une instance EC2 dans le sous réseau public

Remarque :

Pour se connecter en ssh il faut changer les droits de permissions du clé privé à 600

```
chmod 600 ahmedKey.pem
```

Et après se connecte à travers cette commande :

```
ssh -i ahmedKey.pem admin@13.37.227.3
```

puisque la machine est debian l'utilisateur quand utilise pour se connecter à cette instance est « admin »

2-\ Lancer une base de données RDS dans le sous réseau privé

Create database


Choose a database creation method [Info](#)


☒ Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.


☐ Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.


Engine options


Engine type [Info](#)


☐ Amazon Aurora


☒ MySQL


☐ MariaDB


☐ PostgreSQL


☐ Oracle


☐ Microsoft SQL Server


Edition

- ☒ MySQL Community

Known issues/limitations

Review the [Known issues/limitations](#) to learn about potential compatibility issues with specific database versions.

Version

MySQL 8.0.23

Templates

Choose a sample template to meet your use case.

☐ Production
Use defaults for high availability and fast, consistent performance.

☐ Dev/Test
This instance is intended for development use outside of a production environment.

☒ Free Tier
Use our Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.

Free Tier

The Amazon RDS Free Tier is available to you for 12 months. Each calendar month, the free tier will allow you to use the Amazon RDS resources listed below for free:

- 750 hrs of Amazon RDS in a Single-AZ db.t2.micro Instance
- 20 GB of General Purpose Storage (SSD).
- 20 GB for automated backup storage and any user-initiated DB Snapshots.

[Learn more about AWS Free Tier](#)

When your free usage expires or your application use exceeds the free usage tiers, you simply pay standard, pay-as-you-go service rates as described in the [Amazon RDS Pricing page](#).

DB instance identifier [Info](#)

wordpress

▼ Credentials Settings

Type a login ID for the master user of your DB instance.

1 to 16 alphanumeric characters. First character must be a letter.

Master password [Info](#)

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), ' (single quote), " (double quote) and @ (at sign).

Mot de passe : 22601523

Connectivity



Virtual private cloud (VPC) [Info](#)

VPC that defines the virtual networking environment for this DB instance.

MainVPC (vpc-0e04100d2badf7aac) ▼

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

Subnet group [Info](#)

DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

Create new DB Subnet Group ▼

Public access [Info](#)

☐ Yes

Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

☒ No

RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group

Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

☒ Choose existing
Choose existing VPC security groups

☐ Create new
Create new VPC security group

Existing VPC security groups

Choose VPC security groups ▼

default ✕

Availability Zone [Info](#)

eu-west-3a ▼

Copier endpoint pour pouvoir se connecter à ce serveur

Changer les règles de groupes de sécurité pour laisser les VMs du VPC de se communiquer avec le SGBD.

Connectivity & security

Endpoint & port

Endpoint

wordpress.c0epnscw657t.eu-west-3.rds.amazonaws.com

Port

3306

Networking

Availability Zone

eu-west-3a

VPC

MainVPC (vpc-0e04100d2badf7aac)

Subnet group

default-vpc-0e04100d2badf7aac

Subnets

subnet-00ced15b5bd9f2d04
subnet-034de2ca2ae4e295a
subnet-0b0a9605b35725bf0
subnet-015688f94a7b73e23

Security

VPC security groups

default (sg-001ab00f9acbb4bde)

✓ Active

Publicly accessible

No

Certificate authority

rds-ca-2019

Certificate authority date

August 22, 2024, 07:08
(UTC±7:08)

Edit inbound rules [info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules [info](#)

Security group rule ID	Type info	Protocol info	Port range info	Source info	Description - optional info	
sgr-022415f2d8139179c	All TCP ▼	TCP	0 - 65535	Custom ▼ <input type="text" value="10.0.0.0/16"/>		<div>Delete</div>

Add rule

Cancel

Preview changes

Save rules

3-\ Connecter à la base de données depuis mon instance EC2

Installer le client MySQL :

```
sudo apt-get install default-mysql-client
```

Après se connecter à la base de données :

```
mysql -h wordpress.c0epnscw657t.eu-west-3.rds.amazonaws.com -u  
admin -p22601523
```

4-\ Installer wordpress

Installation de apache 2 et php :

```
sudo apt install apache2 php7.3 libapache2-mod-php7.3 php7.3-  
common php7.3-mbstring php7.3-xmlrpc php7.3-soap php7.3-gd  
php7.3-xml php7.3-intl php7.3-mysql php7.3-cli php7.3-ldap  
php7.3-zip php7.3-curl
```

Installation WordPress:

```
:~$ cd /tmp/  
  
:~$ wget -c https://wordpress.org/latest.tar.gz
```

Décompresser et déplacer le dossier WordPress

```
:~$ tar -xvzf latest.tar.gz  
  
:~$ sudo mv wordpress/ /var/www/html/  
  
:~$ sudo chown -R www-data:www-data /var/www/html/wordpress/  
  
:~$ sudo chmod 755 -R /var/www/html/wordpress/
```

Creation d'une nouvelle virtualhost pour wordpress

```
sudo nano /etc/apache2/sites-available/wordpress.conf
```

ajouter ces lignes :

```
<VirtualHost *:80>

    ServerAdmin admin@your_domain.com

    DocumentRoot /var/www/html/wordpress

    ServerName @IP public du EC2

    <Directory /var/www/html/wordpress>

        Options FollowSymlinks

        AllowOverride All

        Require all granted

    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/your-domain.com_error.log

    CustomLog ${APACHE_LOG_DIR}/your-domain.com_access.log
    combined

</VirtualHost>
```

Exécuter les commandes suivantes

```
a2ensite wordpress.conf

systemctl restart apache2

a2enmod rewrite

systemctl restart apache2
```

En dessous, cliquez sur **launch-wizard** près du mot **Security groups**.

Instance: **i-028e8bd85726b7096** Public IP: **34.242.214.97**

Description Status Checks Monitoring Tags

Instance ID	i-028e8bd85726b7096
Instance state	running
Instance type	t2.micro
Elastic IPs	
Availability zone	eu-west-1b
Security groups	launch-wizard-2 . view inbound rules . view outbound rules
Scheduled events	No scheduled events
AMI ID	amzn2-ami-hvm-2.0.20190115-x86_64-gp2 (ami-0fad7378adf284ce0)

Ajoutez l'accès HTTP comme ceci :

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
HTTP ▾	TCP	80	Custom ▾ 0.0.0.0/0	e.g. SSH for Admin Desktop ✕
SSH ▾	TCP	22	Custom ▾ 0.0.0.0/0	e.g. SSH for Admin Desktop ✕

Essayez à présent d'accéder à <http://-ip-de-votre-machine/>, et vous verrez apparaître l'écran d'installation du blog

Compléter l'installation en utilisant le navigateur web

Database Name	wordpress
Username	admin
Password	22601523
Database Host	Endpoint du rds
Table Prefix	wp_

II-\ Création d'un conteneur sur une instance EC2

Objectif :

- Envoie du backup de mon site sur l'instance EC2.
- Création d'un conteneur wordpress hébergeant mon site
- Le site doit être relié à une base de données RDS et S3 pour le stockage des fichiers

1-\ le stockage des fichiers sur S3

<https://www.wpmentor.com/wordpress-s3/>


Le premier lien qui m'a permis de créer le bucket et faire la liaison

<https://www.codeinwp.com/blog/wordpress-s3-guide/>

<https://pressable.com/knowledgebase/how-to-offload-wp-media-to-aws-s3/#overview>

Avant de prendre un backup de mon site, j'ai utilisé le plugin de wordpress WP Offload pour faire le stockage sur S3

Les médias les images vont être placés sur un bucket S3



WP Offload Media Lite for Amazon S3, DigitalOcean Spaces, and Google Cloud Storage

Copies files to Amazon S3, DigitalOcean Spaces or Google Cloud Storage as they are uploaded to the Media Library. Optionally configure Amazon CloudFro ...

By Delicious Brains

Install Now

More Details

★★★★☆ (80)

40,000+ Active Installations

Last Updated: 7 days ago

✓ Compatible with your version of WordPress

Amazon S3 > siteweb-wordpress

siteweb-wordpress Info

Publicly accessible

Objects

Properties

Permissions

Metrics

Management

Access Points

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Copy S3 URI

Copy URL

Download

Open

Delete

Actions ▾

Create folder

Upload

< 1 >

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	document/	Folder	-	-	-

Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in this bucket.

Edit

Delete

```
{
  "Version": "2008-10-17",
  "Statement": [
    {
      "Sid": "AllowPublicRead",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::siteweb-wordpress/*"
    }
  ]
}
```

2-\Envois des fichiers de sites vers mon instances EC2 :

scp -i ahmedKey.pem /root/sites/backup_base28_11_2021.sql
[admin@13.38.54.0:/tmp/](#)

```
root@serveurfichiers:~# cd keys/
root@serveurfichiers:~/keys# scp -i ahmedKey.pem /root/sites/backup_base28_11_2021.sql admin@13.38.54.0:/tmp/
backup_base28_11_2021.sql
root@serveurfichiers:~/keys# scp -i ahmedKey.pem /root/sites/site28_11_2021.tar.gz admin@13.38.54.0:/tmp/
site28_11_2021.tar.gz
root@serveurfichiers:~/keys#
```

ssh -i ahmedKey.pem [admin@13.38.54.0](#)

```
root@serveurfichiers:~/keys# ssh -i ahmedKey.pem admin@13.38.54.0
Linux ip-10-0-100-43 4.19.0-16-cloud-amd64 #1 SMP Debian 4.19.181-1 (2021-11-16)

The programs included with the Debian GNU/Linux system are free software; the
exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Nov 27 21:17:14 2021 from 89.226.123.52
admin@ip-10-0-100-43:~$ cd /tmp/
admin@ip-10-0-100-43:/tmp$ ls
backup_base28_11_2021.sql
site28_11_2021.tar.gz
systemd-private-1f7cf677dad041e2b41c0ff37ca85de4-chrony.service-9bDGxM
admin@ip-10-0-100-43:/tmp$ cd
admin@ip-10-0-100-43:~$ pwd
/home/admin
admin@ip-10-0-100-43:~$ ls
images_docker
admin@ip-10-0-100-43:~$ mkdir sites
admin@ip-10-0-100-43:~$ cd sites/
admin@ip-10-0-100-43:~/sites$ pwd
/home/admin/sites
admin@ip-10-0-100-43:~/sites$ mv /tmp/backup_base28_11_2021.sql .
admin@ip-10-0-100-43:~/sites$ mv /tmp/site28_11_2021.tar.gz .
admin@ip-10-0-100-43:~/sites$ ls
backup_base28_11_2021.sql  site28_11_2021.tar.gz
```

3-\ Restauration de la base de données

https://docs.aws.amazon.com/fr_fr/AmazonRDS/latest/UserGuide/MySQL.Procedural.Importing.NonRDSRepl.html

Pour restaurer la base de données

Accéder à la base de données

mysql -h db-wordpress.c0epnscw657t.eu-west-3.rds.amazonaws.com -u admin
-p22601523

après

source backup_base28_11_2021.sql

```
admin@ip-10-0-100-43:~/sites$ ls
backup_base28_11_2021.sql  site28_11_2021.tar.gz
admin@ip-10-0-100-43:~/sites$ mysql -h db-wordpress.c0epnscw657t.eu-west-3.rds.amazonaws.com -u admin -p22601523
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 127
Server version: 8.0.23 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)]> source backup_base28_11_2021.sql
Query OK, 0 rows affected (0.001 sec)

Query OK, 0 rows affected (0.000 sec)

Query OK, 0 rows affected (0.000 sec)
```

Appliquer des mises à jour sur la base de données :

@IP source : http://192.168.1.51

@IP destination: http://13.38.54.0

UPDATE wp_options SET option_value = replace(option_value, 'http://192.168.1.51',
'http://13.38.54.0') WHERE option_name = 'home' OR option_name = 'siteurl';

UPDATE wp_posts SET guid = replace(guid, 'http://192.168.1.51','http://13.38.54.0');

UPDATE wp_posts SET post_content = replace(post_content, 'http://192.168.1.51',
'http://13.38.54.0');

UPDATE wp_postmeta SET meta_value =
replace(meta_value,'http://192.168.1.51','http://13.38.54.0');

4-\ création d'un dockerfile pour une image wordpress

```
FROM wordpress:latest

MAINTAINER cherif ahmed <ahmedcherif3232@gmail.com>

WORKDIR /var/www/html

COPY wordpress/. .
```

Ce docker file permet de créer une image qui me permet de lancer un conteneur ou l'environnement de WordPress est installé (php ,apache2, ..) et de copier mon backup de mon site vers le /var/www/html de mon conteneur .

```
admin@ip-10-0-100-43:~/images_docker$ ls
Dockerfile  docker-compose.yaml  wordpress
admin@ip-10-0-100-43:~/images_docker$ cd wordpress/
admin@ip-10-0-100-43:~/images_docker/wordpress$ ls
index.php      wp-admin          wp-config.php    wp-links-opml.php  wp-settings.php
license.txt    wp-blog-header.php wp-content        wp-load.php        wp-signup.php
readme.html    wp-comments-post.php wp-cron.php      wp-login.php       wp-trackback.php
wp-activate.php wp-config-sample.php wp-includes      wp-mail.php        xmlrpc.php
admin@ip-10-0-100-43:~/images_docker/wordpress$
```

Pour créer l'image à partir dockerfile :

```
docker image build -t siteweb:1.0 .
```

```
admin@ip-10-0-100-43:~/images_docker$ ls
Dockerfile  docker-compose.yaml  wordpress
admin@ip-10-0-100-43:~/images_docker$ docker image build -t siteweb:1.0 .
```

5-\ création d'un docker-compose

<https://codebeautify.org/yaml-validator>

<https://devopssec.fr/article/gerez-vos-conteneurs-docker-compose>

```
version: '3.7'
services:
  wordpress:
    image: siteweb:1.0
    ports:
      - 80:80
    restart: always
    environment:
      - WORDPRESS_DB_HOST=db-wordpress.c0epnscw657t.eu-west-3.rds.amazonaws.com
      - WORDPRESS_DB_PORT=3306
      - WORDPRESS_DB_USER=admin
      - WORDPRESS_DB_PASSWORD=22601523
      - WORDPRESS_DB_NAME=wordpress
```

Ce docker-compose permet de créer le conteneur à partir de l'image créer à partir de dockerfile et faire la liaison de mon conteneur avec la base de données qui stocker sur le service rds

Pour la création des conteneurs à partir de docker-compose en tape l'instruction suivante :

```
docker-compose up -d
```

```
admin@ip-10-0-100-43:~/images_docker$ ls
Dockerfile  docker-compose.yaml  wordpress
admin@ip-10-0-100-43:~/images_docker$ nano docker-compose.yaml
admin@ip-10-0-100-43:~/images_docker$ docker-compose up -d
```