Wordpress

wordpress config for cloudflare tunnel with ssl

- [Dockerized WordPress with SSL Certificate Generation](#)
- [Setting up WordPress, MySQL, and phpMyAdmin using Docker Compose via Git](#)

Dockerized WordPress with SSL Certificate Generation

In this tutorial, we'll walk through the process of enhancing a Dockerized WordPress setup with SSL certificate generation. This will ensure secure communication between servers and clients, bolstering the security of your WordPress deployment.

Prerequisites:

1. Docker installed on your system.
2. Basic understanding of Docker Compose.

Step 1: Update Docker Compose Configuration

Open your `docker-compose.yml` file and make the following modifications:

```
version: '3.1'
services:
  wordpress:
    image: wordpress:php7.2-apache
    ports:
      - "8001:80"
    environment:
      WORDPRESS_DB_HOST: mysql
      WORDPRESS_DB_USER: root
      WORDPRESS_DB_PASSWORD:
      WORDPRESS_DB_NAME: wordpress_db
    volumes:
      - ./website:/var/www/html
      - ./ssl:/etc/apache2/ssl  # Mount SSL directory
    depends_on:
      - mysql
```

```yaml
    networks:
      - my_network

  mysql:
    image: mysql:5.7
    environment:
      MYSQL_DATABASE: wordpress_db
      MYSQL_ROOT_PASSWORD:
    ports:
      - "3306:3306"
    volumes:
      - ./data:/var/lib/mysql
    networks:
      - my_network

  phpmyadmin:
    image: phpmyadmin/phpmyadmin
    links:
      - mysql
    ports:
      - "8081:80"
    environment:
      PMA_HOST: mysql
      MYSQL_ROOT_PASSWORD:
    networks:
      - my_network

networks:
  my_network:
    driver: bridge
```

Step 2: Create SSL Certificate Generation Script

Generate a bash script named `generate_ssl.sh` with the following content:

```bash
#!/bin/bash

SSL_DIR="./ssl"
PRIVATE_KEY="$SSL_DIR/private.key"
CERTIFICATE="$SSL_DIR/certificate.crt"

# Create SSL directory if it doesn't exist
mkdir -p $SSL_DIR

# Generate SSL certificate if not already present
if [ ! -f "$PRIVATE_KEY" ] || [ ! -f "$CERTIFICATE" ]; then
    openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout "$PRIVATE_KEY" -out "$CERTIFICATE"
fi
```

Step 3: Implement SSL Certificate Generation

Execute the following commands in your terminal:

```
chmod +x generate_ssl.sh
./generate_ssl.sh
```

Step 4: Start Docker Containers

Run the following command to initiate your Docker containers:

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

Conclusion

Congratulations! You've successfully enhanced your Dockerized WordPress setup with SSL certificate generation. Your WordPress deployment now benefits from improved security through encrypted communication.

Setting up WordPress, MySQL, and phpMyAdmin using Docker Compose via Git

This guide demonstrates setting up WordPress, MySQL, and phpMyAdmin using Docker Compose version 3.1. This setup allows you to quickly create a local development environment for WordPress with a MySQL database and phpMyAdmin for database management.

Docker Compose File (docker-compose.yml)

```
version: '3.1'
services:
  wordpress:
    image: wordpress:php7.2-apache
    ports:
      - "8001:80"
    environment:
      WORDPRESS_DB_HOST: mysql
      WORDPRESS_DB_USER: root
      WORDPRESS_DB_PASSWORD:
      WORDPRESS_DB_NAME: wordpress_db
    volumes:
      - ./website:/var/www/html
      - ./ssl:/etc/apache2/ssl  # Mount SSL directory
    depends_on:
      - mysql
    networks:
      - my_network
```

```yaml
  mysql:
    image: mysql:5.7
    environment:
      MYSQL_DATABASE: wordpress_db
      MYSQL_ROOT_PASSWORD:
    ports:
      - "3306:3306"
    volumes:
      - ./data:/var/lib/mysql
    networks:
      - my_network

  phpmyadmin:
    image: phpmyadmin/phpmyadmin
    links:
      - mysql
    ports:
      - "8081:80"
    environment:
      PMA_HOST: mysql
      MYSQL_ROOT_PASSWORD:
    networks:
      - my_network

networks:
  my_network:
    driver: bridge
```

Bash Script (`setup.sh`)

```bash
#!/bin/bash

# Function to generate SSL certificates
generate_ssl_certificates() {
    SSL_DIR="./ssl"
    PRIVATE_KEY="$SSL_DIR/private.key"
    CERTIFICATE="$SSL_DIR/certificate.crt"
```

```
    # Create SSL directory if it doesn't exist
    mkdir -p $SSL_DIR

    # Generate SSL certificate if not already present
    if [ ! -f "$PRIVATE_KEY" ] || [ ! -f "$CERTIFICATE" ]; then
        openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout "$PRIVATE_KEY" -out "$CERTIFICATE"
    fi
}


# Function to start containers using Docker Compose
start_containers() {
    docker-compose up -d
}


# Main function
main() {
    generate_ssl_certificates

    # Start containers using Docker Compose
    start_containers
}


# Execute main function
main
```

Instructions for Setup

1. Ensure Docker and Docker Compose are installed on your system.
2. Clone the GitHub repository: `git clone https://github.com/j551n-ncloud/wordpress.git`
3. Navigate to the cloned directory: `cd wordpress`
4. Create the `docker-compose.yml` file with the provided content.
5. Create the `setup.sh` file with the provided content.
6. Make the `setup.sh` file executable: `chmod +x setup.sh`
7. Execute the script to start the containers: `./setup.sh`

This setup will launch WordPress accessible at `http://localhost:8001`, phpMyAdmin accessible at `http://localhost:8081`, and MySQL running internally. Additionally, SSL certificates will be generated for the Apache server hosting WordPress. You can customize ports and environment variables as per your requirements.

Please ensure to handle sensitive information like passwords securely, especially in a production environment.