

EC2 障害時の復旧ガイド（あなたの環境専用）

このドキュメントは、WordPress（Nginx + PHP-FPM）・MySQL・Flask（/kaiga）を Docker で運用しているあなたの EC2 が、

万が一障害で停止・破損した場合に確実に復旧できる状態をまとめたものです。

復旧に必要なバックアップ

1. MySQL データ（最重要）

- バックアップファイル例：db_YYYYMMDD_HHMMSS.sql

- 内容：WordPress

の投稿・固定ページ・メディア情報・ユーザ情報・設定など全データベース内容。

- 保存場所：~/wp_backups/

- 取得コマンド：./backup_db.sh

2. WordPress ファイル一式

- ディレクトリ例：wp_YYYYMMDD_HHMMSS/

- 内容：wp-content（画像・テーマ・プラグイン）、wp-config.php、WordPress コアファイル

- 取得コマンド：./backup_wp_files.sh

3. Docker 構成（docker-compose）

- /home/ubuntu/wordpress/ に保持

- GitHub 等にコピーしておけばさらに安全

4. Nginx 設定（/etc/nginx/conf.d/）

- セキュリティヘッダなど手動で修正した設定を含む

- バックアップ推奨

障害発生後の復旧手順

Step 1：新しい EC2 を起動

- OS : 元と同じ Ubuntu

- Elastic IP を割り当てる (推奨)

Step 2 : Docker / docker-compose をインストール

```
sudo apt update
```

```
sudo apt install docker.io docker-compose -y
```

Step 3 : wordpress ディレクトリを復元

- docker-compose.yml を ~/wordpress に置く

Step 4 : Docker を再構築

```
cd ~/wordpress
```

```
docker compose up -d
```

Step 5 : MySQL データベースをリストア

```
cat ~/wp_backups/db_YYYYMMDD.sql | docker exec -i wordpress-db-1 mysql -u -p
```

Step 6 : WordPress ファイルをリストア

```
docker cp ~/wp_backups/wp_YYYYMMDD/html wordpress-nginx-1:/var/www/
```

Step 7 : Nginx 設定をリストア (必要なら)

```
docker cp ~/wp_backups/nginx_conf_YYYYMMDD/conf.d wordpress-nginx-1:/etc/nginx/
```

```
docker exec wordpress-nginx-1 nginx -s reload
```

最終チェック

- WordPress 表示

- /kaiga/ 動作

- /wp-admin ログイン

- 記事・画像表示

このドキュメントの目的

- EC2 障害でも 5 ~ 10 分で復旧できる状態を作る
- Docker 構成は再構築可能
- DB とファイルがあれば完全復元できる