

Symfony – Déploiement

Générer le fichier .htaccess pour configurer la réécriture d'URL d'Apache

```
composer require symfony/apache-pack
```

Dans `public/` un fichier `.htaccess` a été ajouté pour que notre site fonctionne correctement au niveau de la réécriture d'URL.

Avec accès SSH

On récupère l'adresse d'accès SSH de notre serveur ensuite dans notre invite de commande on va écrire. (Attention ssh est accessible sur PowerShell, PuTTY ou CMD)

```
ssh nomutilisateur@ssh.cluster000.hosting.hebergeur.net
```

il va demander le mot de passe du compte, une fois ajouté, on est connecté à l'hébergement

la commande `ls` permet de voir la liste des fichiers qui existent.

La 1^{ère} chose à faire c'est d'installer composer, pour ce faire il faut aller à cette adresse :

<https://getcomposer.org/download/>

Et de copier ces lignes de commande :

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file('sha384', 'composer-setup.php') ===
'93b54496392c062774670ac18b134c3b3a95e5a5e5c8f1a9f115f203b75bf9a129d5daa8ba6a13e2cc8a1da0806388a8') {
echo 'Installer verified'; } else { echo 'Installer corrupt'; unlink('composer-setup.php'); } echo PHP_EOL;"
php composer-setup.php
php -r "unlink('composer-setup.php');"
```

Ceci va installer **composer.phar**

Pour l'utiliser dans notre système au lieu d'écrire les lignes de commande avec juste composer on va écrire `php composer.phar`

On peut créer un alias pour éviter de taper toute la ligne de commande comme ceci :

```
alias composer="php ~/composer.phar"
```

On va pouvoir maintenant utiliser le mot clé composer directement comme en local

Télécharger les sources

Dans votre dépôt git vous allez chercher votre projet et sur le bouton (vert) « clone or download » copier l'adresse URL donnée

Dans votre ligne de commande :

```
git clone https://github.com/pseudo/projet.git
```

Ceci va télécharger et installer les fichiers du projet.

Maintenant on peut aller dans notre nouveau dossier de projet `cd nomdudossier/`

Et ensuite `php ../composer.phar update` pour mettre à jour les dépendances dans le projet.

Maintenant à partir du FTP, on va récupérer le fichier `.env` et aller changer les informations

Pour la base de données :

```
DATABASE_URL=mysql://root:root@adressebasededonnee.mysql.db/nombdd
```

On peut ensuite installer notre projet en réalisant la création de la base de données si nécessaire, les migrations et fixtures (si besoin).

On va créer un lien symbolique

Lien entre le dossier `www` et notre dossier-projet/public

Pour ce faire on peut supprimer sur FTP le dossier `www` ou via ssh

- Pour sortir du dossier-projet on fait un `cd ..`
- Et taper `rm -rf www`

Pour créer le lien symbolique on va écrire `ln -s dossier-projet/public www`

`www` va devenir un lien vers le dossier-projet

Enfin il suffit de revenir dans le fichier `.env`

Pour changer la ligne `APP_ENV` en production

```
APP_ENV=prod
```

Sans accès SSH

Il faut transférer le contenu de notre dossier-projet via FTP dans le dossier WWW de notre hébergeur.

Dans le fichier .env il faut modifier les informations de notre base de données :

```
DATABASE_URL=mysql://root:root@adressebasededonnee.mysql.db:3306/nombdd
```

Et importer la base de données dans le serveur avec phpMyAdmin

Attention : désactiver la vérification des clés étrangères dans les options de l'exportation (Options spécifiques au format).

Faire pointer le nom de domaine du site vers le dossier public/

Enfin il suffit de revenir dans le fichier .env

Pour changer la ligne APP_ENV en production

```
APP_ENV=prod
```