

# Programmation avancée & Symfony

BUT Informatique - 3ème année (*R5A.05*)

## TP 8

### *Prod.* & *Notation du projet*

© Claire BOMBEAUX

*Tous droits réservés*

Version : **02/12/2025** 1559



### Objectifs du TP 8

### PROJET SYMFONY

- Finaliser les développements, améliorer certains écrans et fonctionnalités, ...
- Comprendre ce qu'est une Mise En Production (MEP)
- Savoir préparer une application Symfony en vue d'une MEP



# Sommaire

■ Objectifs du TP 8 .....	1
■ Sommaire.....	2
Introduction.....	3
1. En route vers la PROD. .....	3
1.1    Les MEP .....	3
1.2    Mettre en production un projet réalisé avec Symfony.....	5
2. Notation du TP .....	6
2.1    Avant le TP noté.....	6
2.2    Pendant le TP noté .....	7
3. Pour aller plus loin .....	7

# Introduction

*Vous pouvez profiter de ces deux heures de TP pour :*

- ❖ Finaliser vos développements, améliorer certains écrans et fonctionnalités, ...
- ❖ Préparer votre application pour la PRODUCTION (Cf. partie 1 de ce document)
- ❖ Prendre connaissance des consignes et conseils pour la notation du TP et **préparer les éléments demandés** (Cf. partie 2 de ce document)

## 1. En route vers la PROD.

### 1.1 Les MEP

La **mise en production** ou **MEP**, d'un projet est le processus qui consiste à préparer et à déployer tout ou partie d'un composant de SI (application, ...) sur un ou des environnements accessibles aux utilisateurs finaux, après les différentes phases précédentes (conception, développement, validation...). Ces environnements sont généralement appelés **environnements de production (PROD)**, même si d'autres appellations peuvent également se retrouver (par exemple : env. d'exploitation, EXPLOIT).

Concrètement, cela implique généralement plusieurs étapes, telles que :

- Gestion et planification de la date et des différentes actions. Les phases de gestion et suivi du projet s'appliquent continuellement. De plus, une MEP pouvant entraîner une **rupture de service**, certaines actions, telles que prévenir les utilisateurs, sont généralement nécessaires.
- Un **plan de retour arrière** est souvent à définir, dans le cas où la MEP se passerait « mal ».
- Préparation de(s) environnement(s) de production, respectant les prérequis techniques. Cela inclut l'installation des machines, composants, librairies, utilitaires, technos... nécessaires au fonctionnement de l'appli., en respectant les versions attendues...
- **Déploiement** : transférer le code et de le rendre actif sur les serveurs de production, de façon manuelle (par ex. via FTP) ou via des pipelines automatisés (CI/CD).
- Surveillance et tests post-déploiement : surveillance et vérification de l'application suite à la phase de déploiement, afin de détecter d'éventuels bugs, problèmes de performance ou pannes.

Après la MEP, d'autres actions sont nécessaires, par exemple :

- Surveillance, **monitoring, supervision** : surveillance de l'**application** (afin de détecter d'éventuels bugs, problèmes de performance, pannes, ...), des **services** associés + Surveillance de l'**infrastructure** (machines) de production. Utilisation d'outils de monitoring, éventuellement, avec alertes automatiques.

The screenshot shows the Icinga web interface with several panels:

- Current Incidents:** Lists various critical issues such as SSH on 'ahhhhhhhhhh' being down, Pong na Google being critical, DNS on 'DNS Server 1' being critical, and MySQL Server being critical.
- Host Problems:** Shows MySQL Server, DNS Server 2, Pong na Google, and DNS Server 1 all being down.
- Recently Recovered Services:** Lists services that have recently recovered, including Git Server, MySQL Development, and SSH on Git Server.
- Cube:** A heatmap visualization showing service status across environments (development, production, testing) and locations (berlin, london, new-york). Most services are shown as green (OK).
- Plugin Output:** Displays logs from the 'Webscraper Testing' plugin, showing it has been running for 10 hours, 41 minutes, and 14 seconds.
- Check Statistics:** Shows execution times for checks like 'random' and 'demo-icinga2'.
- Performance Data:** Displays numerical values for time, value, value\_1m, value\_5m, and uptime.
- Custom Variables:** Shows environment (testing), location (berlin), and OS (linux).
- Feature Commands:** Shows active checks.

*Icinga, solution de surveillance, de supervision open source*

By Icinga GmbH - Converted to WebP and uploaded by VulcanSphere from <https://icinga.com/products/> (direct link on <https://icinga.com/wp-content/uploads/2025/03/icinga-demo-screenshot.png>), GPLv2, <https://commons.wikimedia.org/w/index.php?curid=168630988>

- **Sauvegardes** : Des sauvegardes régulières sont nécessaires. Il s'agit de réaliser des sauvegardes, de préférence automatiquement, de l'application, mais également de la base de données / des **données**, voire de **d'autres éléments** (tels que les fichiers uploadés par les utilisateurs). Ces sauvegardes s'inscrivent généralement dans des **plans de sauvegardes** et doivent permettre la **restauration** du service si nécessaire (perte de données tolérées ou non selon cas).
- **Mises à jour** : Les mises à jour des machines, composants, librairies, utilitaires, technos... sont à suivre, gérer et réaliser.  
Cela inclut d'appliquer régulièrement les **correctifs de sécurité** et de gérer les **CVE**

Les **CVE** ou *Common Vulnerabilities and Exposures* sont des identifiants uniques attribués aux failles de sécurité découvertes dans les logiciels, les matériels ou les systèmes d'exploitation.



Les CVE sont incluses dans un référentiel public disponible ici : <https://www.cve.org>.

Dans le cadre de la directive **NIS2**, l'Europe dispose maintenant de sa propre base, appelée EU Vulnerability Database : <https://euvd.enisa.europa.eu/>

The screenshot shows the EU Vulnerability Database interface. At the top, there's a logo for enisa (European Union Vulnerability Database) and search bars for 'Search by ID' and 'Search by text'. Below the header, there are two main sections: 'Critical vulnerabilities' and 'Exploited vulnerabilities'. Each section has a table with columns: ID, Alternative ID, Exploitation, CVSS, Vendor, and Changed. The 'Critical vulnerabilities' section contains several entries, including one for Sprecher Automations SPRECON-E-C, P, and T3. The 'Exploited vulnerabilities' section contains entries for various companies like OpenPLC, Google, Oracle Corporation, and Fortinet.

Critical vulnerabilities						Exploited vulnerabilities					
ID	Alternative ID	Exploitation	CVSS	Vendor	Changed	ID	Alternative ID	Exploitation	CVSS	Vendor	Changed
EUVD-2025-200220	CVE-2025-41742	Not available	● v3.1: 9.8	Sprecher Automation	4 hours ago	EUVD-2021-13614	CVE-2021-26829	⚠️	● v3.1: 5.4	n/a	9 hours ago
Sprecher Automations SPRECON-E-C, SPRECON-E-P, SPRECON-E-T3 is vulnerable to attack by an unauthorized remote...											
EUVD-2025-200222	CVE-2025-41744	Not available	● v3.1: 9.1	Sprecher Automation	4 hours ago	EUVD-2025-197896	CVE-2025-13223 GHSA-fvx3-7348-92qj	⚠️	● v3.1: 8.8	Google	19 hours ago
Sprecher Automations SPRECON-E series uses default cryptographic keys that allow an unprivileged remote attacker to...											
EUVD-2025-44048	CVE-2025-64689	Not available	● v3.1: 9.6	JetBrains	4 hours ago	Type Confusion in V8 In Google Chrome prior to 142.0.7444.175 allowed a remote attacker to potentially exploit heap...					
This CVE ID has been rejected or withdrawn by its CVE Numbering Authority because it relates to internal functionality that is...											
EUVD-2025-200018	CVE-2025-3500	Not available	● v3.1: 9	Avast	9 hours ago	EUVD-2025-38253	CVE-2025-61757	⚠️	● v3.1: 9.8	Oracle Corporation	10 days ago
Integer Overflow or Wraparound vulnerability in Avast Antivirus (25.1.981.6) on Windows allows Privilege Escalation. This issu...											
An Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection') vulnerability [CWE-78]...											

EUVD - <https://euvd.enisa.europa.eu/>

## 1.2 Mettre en production un projet réalisé avec Symfony

Dans la mesure où nous ne disposons pas d'environnement de production dédié à notre projet, nous nous contenterons de réaliser des actions de « préparation à la MEP ».

En voici quelques-unes :

- En premier lieu, sauvegardez votre code et votre BDD
- Retirez tout code lié au débogage (dump TWIG, var\_dump PHP ...)
- Retirez toutes éventuelles failles de sécurité, type mot de passe « en dur »
- Configurez vos variables d'environnement

et passer en env. de prod.

Configuration de l'env. actif :

<https://symfony.com/doc/current/configuration.html#selecting-the-active-environment>

Obtenir la liste des variables d'environnement :

<https://symfony.com/doc/current/configuration.html#listing-environment-variables>

- Si nécessaire, en supplément, mettez à jour votre base de données

bin/console doctrine:schema:update -f

- Nettoyez le cache de Symfony

- ...



Documentation officielle Symfony - **How to Deploy a Symfony Application**

<https://symfony.com/doc/current/deployment.html>

## 2. Notation du TP

### 2.1 Avant le TP noté

Vous devez rendre :

- **Un document PDF** contenant :
  - (Vos noms, prénoms, groupes de TP...)
  - Les **captures d'écrans des principales fonctionnalités** de votre projet légendées
  - Le **MCD de votre BDD** (schéma représentant les entités, attributs et associations de votre BDD).  
Vous pouvez utiliser le *Concepteur de phpMyAdmin* pour l'obtenir
- Assurez-vous que les captures d'écran fournies correspondent à votre code source (et ce que vous présenterez en TP).
- S'agissant d'un travail en binôme, une trop forte ressemblance entre deux projets entre deux binômes différents est évidemment sanctionnable.
- **Le code source de votre application**
  - Zippé
  - sans le dossier /var
  - sans le dossier /vendor (?)
  - Les fichiers des dossiers
- **L'export de votre BDD**
  - au **format SQL**
  - en totalité c'est-à-dire la **structure et les données**
  - avec les *username* et *password* (si différent de *root*)
  - Vous pouvez utiliser les fonctionnalités d'export de *phpMyAdmin* pour le générer



Avant le TP noté, assurez-vous :

- De savoir comment activer/désactiver le *Profiler* de Symfony
- De savoir comment passer de l'environnement de dev. à celui de prod. (et réciproquement)
- De pouvoir accéder sans problème à l'IHM de votre projet, mais également à votre BDD et SGBD
- D'avoir une solution « de secours » en cas d'éventuel imprévu (panne d'ordi. ...)

## 2.2 Pendant le TP noté

- Vous présenterez, en binôme, votre réalisation.
  - Utilisation d'un seul ordinateur par binôme
  - Pendant une 10aine de minutes
- Des manipulations de votre application pourront vous être demandées (par exemple : l'ajout d'un jeu vidéo, ou la suppression d'un genre...)
- Une ou plusieurs questions « techniques » vous seront posées
  - Pour y répondre, l'utilisation de ressources (tel que le cours) ou d'Internet (sur tel ou ordi...) n'est pas autorisée.
  - Les questions sont choisies de façon aléatoire, sauf si celles-ci concernent un ou des points liés à votre projet
  - Les thèmes des questions sont variés, mais sont essentiellement liés aux TPS
  - Il n'est pas utile de connaître par cœur les commandes console (valable également pour l'examen écrit).

## 3. Pour aller plus loin

Nous aurions pu aborder et implémenter dans notre projet de nombreux autres sujets et fonctionnalités, tels que :

- Gestion des utilisateurs et de leurs droits → Gestion des connexions utilisateur, de l'authentification, des sessions, des droits utilisateurs (ACL, permissions)
- Messages flash
- Gestion de la sécurité
- Gestion des erreurs
- Testing → tests unitaires, tests fonctionnels, tests automatisés
- Translations / Internationalization
- Mailer, envoi de mails
- Scheduler (planification et exécution des tâches de manière automatique et récurrente)
- Gestion des assets, optimisation des assets, *Webpack Encore...*
- ...

Si vous êtes intéressés, n'hésitez pas à lire de la documentation associée et/ou tester la mise en place de ceux-ci (votre projet peut constituer une base suffisante pour cela).

---

Remarque : l'oral de TP ne sera réalisé que si les conditions le permettent (créneaux permettant de faire passer l'ensemble des groupes, ...). Dans le cas contraire, votre TP sera tout de même très probablement noté via étude de votre code source.