УДК 004

КОМПЛЕКСНЫЙ АНАЛИЗ РАЗВЁРТЫВАНИЯ ВЕБ-ПРИЛОЖЕНИЙ НА ОСНОВЕ ФРЕЙМВОРКА SYMFONY 7 НА ПЛАТФОРМЕ UBUNTU

♦♦♦♦♦

A COMPREHENSIVE ANALYSIS OF DEPLOYING WEB APPLICATIONS BASED ON THE SYMFONY 7 FRAMEWORK ON THE UBUNTU PLATFORM

Самсонов Игорь Юрьевич
старший преподаватель кафедры математического моделирования, компьютерных технологий и информационной безопасности,
Елецкий государственный университет им. И.А. Бунина
ig48sam@gmail.com

Андропова Ольга Юрьевна
старший преподаватель
кафедры математического моделирования, компьютерных технологий и информационной безопасности,
Елецкий государственный университет им. И.А. Бунина
olya199612@bk.ru

Александрова Людмила Николаевна
кандидат педагогических наук,
доцент,
доцент кафедры математического моделирования, компьютерных технологий
и информационной безопасности,
Елецкий государственный университет им. И.А. Бунина
alexandrovaludmila@rambler.ru

Аннотация. В статье рассматриваются методологические основы и практические подходы к развёртыванию современных веб-приложений, построенных с использованием фреймворка Symfony 7, в локальной вычислительной среде на базе операционной системы Linux (дистрибутив Ubuntu), актуальность применения которых не вызывает сомнения. Анализируются ключевые компоненты инфраструктуры – веб-сервер Apache 2, интерпретатор PHP версии не ниже 8.2, система управления зависимостями Composer, а также механизмы управления базами данных через Doctrine. Особое внимание уделено вопросам настройки прав доступа, изоляции окружения и выполнения миграций. Работа опирается на отечественные и зарубежные источники, включая официальную документацию Symfony и исследования в области DevOps-практик.

Ключевые слова: Symfony, веб-приложение, развёртывание, Apache, Linux, PHP, Doctrine, миграции, локальная среда, безопасность.

Samsonov Igor Yuryevich
Senior Lecturer at the Department of Mathematical Modeling, Computer Technologies and Information Security,
Bunin Yelets State University
ig48sam@gmail.com

Andropova Olga Yuryevna
Senior Lecturer at the Department of Mathematical Modeling, Computer Technologies and Information Security,
Bunin Yelets State University
olya199612@bk.ru

Alexandrova Lyudmila Nikolaevna
Candidate of Pedagogical Sciences,
Associate Professor,
Associate Professor of the Department of Mathematical Modeling, Computer Technologies and Information Security,
Yelets State University named After I.A. Bunin
alexandrovaludmila@rambler.ru

Annotation. This article examines the methodological foundations and practical approaches to deploying modern web applications built using the Symfony 7 framework in a local computing environment running the Linux operating system (Ubuntu distribution). Key infrastructure components are analyzed: the Apache 2 web server, PHP 8.2 or later, the Composer dependency management system, and database management mechanisms via Doctrine. Particular attention is paid to access rights configuration, environment isolation, and migrations. The work draws on domestic and international sources, including the official Symfony documentation and research on DevOps practices.

Keywords: Symfony, web application, deployment, Apache, Linux, PHP, Doctrine, migrations, local environment, security.

Введение

Современная веб-разработка характеризуется высокой степенью модульности, использованием сложных архитектурных паттернов и строгими требованиями к воспроизводимости и безопасности развёртываемых решений. Одним из наиболее зрелых и широко применяемых инструментов в экосистеме языка PHP является фреймворк Symfony, который начиная с версии 7 (выпущена в ноябре 2023 года) предъявляет повышенные требования к программной среде, в частности, к версии интерпретатора PHP (не ниже 8.2). В условиях роста числа исследований, посвящённых автоматизации жизненного цикла программного обеспечения, особую актуальность приобретают методики локального развёртывания приложений, позволяющие модели-

ровать production-среду без привлечения облачных ресурсов. Вследствие этого авторами проанализирован и представлен комплексный подход к развёртыванию Symfony-приложений на локальной машине под управлением Ubuntu – дистрибутива, доминирующего в серверных и исследовательских средах в Российской Федерации.

Актуальность исследования обусловлена несколькими факторами. Во-первых, в условиях санкционного давления и стремления к технологической независимости возрастает интерес российских разработчиков и исследовательских коллективов к open-source решениям, развёртываемым на отечественных или нейтральных платформах. Налицо потребность в локализованных, автономных и полностью контролируемых средах разработки, обеспечивающих технологическую независимость и совместимость с инфраструктурой, доступной без ограничений.

Во-вторых, локальное тестирование и отладка остаются неотъемлемой частью процесса разработки, особенно в академической среде, где ограничены ресурсы для поддержки полноценной CI/CD-инфраструктуры. Фреймворк перешёл к строгим требованиям к окружению, что делает устаревшие методы развёртывания, например, на базе PHP 7.4 или mod_php, технически неприменимыми и потенциально уязвимыми. Это требует также и пересмотра стандартизации практик локального развёртывания, особенно в образовательной и исследовательской среде, где часто используются унаследованные окружения.

В-третьих, Symfony 7 вводит ряд архитектурных улучшений, включая строгую типизацию и отказ от устаревших компонентов, что требует пересмотра традиционных подходов к конфигурированию окружения [1]. Ручная настройка окружения, хотя и сохраняет определенную ценность, всё чаще уступает место автоматизированным и воспроизводимым подходам (Docker, Ansible), соответствующим принципам Infrastructure as Code и Dev/Prod Parity. Однако внедрение этих практик невозможно без глубокого понимания базовой конфигурации, в частности, правильной настройки PHP-FPM, изоляции public/, управления переменными окружения и ACL-правами. Без этого автоматизация становится «чёрным ящиком», повышающим риски конфигурационных ошибок и уязвимостей безопасности, вопросы обеспечения которой при развёртывании даже в изолированной среде остаются критически важными, поскольку локальные инстансы нередко становятся отправной точкой для последующего развёртывания в промышленных условиях. Большая часть инцидентов в production-среде начинаются с ошибок, допущенных уже на стадии локального тестирования. Корректная настройка веб-сервера (запрет прямого доступа к *.php, кроме index.php), управление секретами через env.local и применение ACL для var/cache/, var/log/ – всё это напрямую влияет на безопасность конечного продукта [2].

Материалы и методы

При изучении заявленной проблемы в научной литературе нам встретилось весьма небольшое число публикаций, особенно на русском языке. Крайне мало таких публикаций относительно новых усовершенствованных версий Symfony в контексте развёртывания веб-приложений на платформе ubuntu.

Так, в [3] автор рассматривает реализацию компьютерно-интенсивной сетевой технологии в архитектуре Symfony платформы психологической взаимопомощи студентов. В рамках разработанной системы сначала анализируются вопросы улучшения безопасности в процессе применения алгоритмов шифрования данных пользователей. Разработчики учитывают архитектуру Symfony для систематического проектирования с применением структуры MVC без учета особенностей развертывания в разных операционных системах.

А.А. Кочнев, исследуя возможности применения фреймворка Symfony и PHP в разработке веб-приложений, по факту представляет лишь теоретический анализ самого фреймворка с точки зрения описания его возможностей, статистики применения разработчиками, перечисления преимуществ и недостатков [4].

В публикации П. Гарбарц и М. Плечавска-Войчик представлен сравнительный анализ, целью которого явилось определение того, какой из шаблонов проектирования на основе PHP (Laravel или Symfony) является лучшим и более универсальным решением для разработчика. Авторы проводят исследование на основе имеющейся документации, сравнении статических и динамических показателей, полученных в исследовательской среде для обоих тестируемых шаблонов. При этом вопросы развертыва-

ния веб-приложений в различных локальных средах, в том числе в Linux, ими также не затрагиваются [3].

Анализ научных публикаций еще раз продемонстрировал перспективность и актуальность нашего исследования, так как научных материалов за последние несколько лет нами не были обнаружены.

Так что же представляет из себя фреймворк Symfony с точки зрения применения его возможностей при проектировании и развёртывании веб-приложений, в частности, в локальной среде Linux

Symfony – это фреймворк, реализующий принципы Model-View-Controller (MVC) и основанный на компонентной архитектуре. Начиная с версии 4, он активно использует концепцию «микроядра» (microkernel), что упрощает управление зависимостями и маршрутизацией. Версия 7 продолжает эту линию, делая акцент на производительности и совместимости с современными стандартами PHP [5]. Для управления внешними библиотеками используется менеджер зависимостей Composer, который формирует автозагрузчик и каталог vendor/ на основе файла composer.json.

Важнейшим аспектом теории развёртывания является разделение окружений: dev, test, prod. В production-режиме отключается отладочная информация, включается кэширование метаданных и оптимизируется автозагрузка. Переменные окружения, включая параметры подключения к базе данных (БД) (DATABASE_URL), хранятся в файлах .env и .env.local, причём последний исключается из системы контроля версий по умолчанию, что соответствует принципам 12-factor app.

Для взаимодействия с реляционными СУБД Symfony использует Doctrine – ORM-библиотеку, реализующую паттерн Data Mapper. Управление схемой базы данных осуществляется через механизм миграций, который позволяет отслеживать изменения структуры БД во времени и обеспечивать согласованность между разработчиками и окружениями [6].

Результаты

Практическая реализация

Практическое развёртывание Symfony 7 на локальной машине с Ubuntu требует последовательного выполнения ряда шагов, направленных на создание стабильной и безопасной среды.

На первом этапе необходимо обеспечить соответствие системных требований: установка Apache 2, PHP 8.2 с расширениями intl, mbstring, xml, zip, а также Composer. В дистрибутивах Ubuntu 22.04 и выше рекомендуется использовать репозиторий ondrej/php для получения актуальных версий PHP и их расширений. После добавления репозитория устанавливается сам PHP. Однако просто установить ядро PHP недостаточно, и для полноценной работы Symfony необходимо также установить ряд ключевых PHP-расширений, которые предоставляют необходимую функциональность для работы с сетью, базами данных, XML и многими другими компонентами фреймворка. Среди этих расширений выделяются php-curl для выполнения HTTP-запросов, php-mbstring для корректной обработки многоязыковых символов, php-xml для парсинга XML, php-zip для работы с архивами, а также php-gd для обработки изображений. Отсутствие одного из этих расширений может привести к ошибкам времени выполнения или неработоспособности некоторых функций фреймворка. После установки следует активировать модуль mod_rewrite, необходимый для корректной обработки ЧПУ (человеко-понятных URL).

На втором этапе создаётся проект с помощью команды composer create-project symfony/website-skeleton, после чего формируется каталог vendor/, содержащий все зависимости. Важно подчеркнуть, что при развёртывании на «чистом» сервере без Composer именно этот каталог должен быть предварительно сформирован на машине разработчика и скопирован вместе с исходным кодом [0]. Критически важным действием, обеспечивающим воспроизводимость окружения, является включение файла composer.lock в систему контроля версий. Этот файл содержит точные версии всех установленных пакетов, включая транзитивные зависимости, гарантируя, что каждый раз, когда другой разработчик клонирует репозиторий и выполняет composer install, он получит абсолютно идентичную версию зависимостей. Без composer.lock команда composer install будет эквивалент на composer update, что может случайно обновить пакеты до новых версий, нарушая работу приложения и порождая проблемы типа «it

works on my machine». Таким образом, composer.lock является не просто дополнительным файлом, а фундаментом для обеспечения стабильности и согласованности в командной работе.

Третий этап включает настройку виртуального хоста Apache. Ключевым требованием является указание DocumentRoot на подкаталог public/проекта, что обеспечивает изоляцию внутренней структуры приложения от прямого доступа через веб. Файлы и каталоги проекта, включая конфигурацию, бизнес-логику и зависимости из vendor/, находятся вне зоны прямого доступа через веб, что предотвращает возможность утечки чувствительной информации или выполнения произвольного кода. Для корректной обработки ЧПУ (человеко-понятных URL) необходимо активировать в Apache модуль mod_rewrite и в конфигурации виртуального хоста разрешить использование файла .htaccess путем установки директивы AllowOverride All Директива AllowOverride All разрешает использование файла .htaccess для перенаправления запросов к front controller (public/index.php). Эта архитектура, основанная на принципах MVC и микроядра, где ядро фреймворка изолировано от внешнего мира, является одной из сильных сторон Symfony и требует от администратора внимательного отношения к конфигурации веб-сервера для ее полного раскрытия.

Особое внимание уделяется настройке прав доступа к каталогам var/cache/ и var/log/. В Linux-средах рекомендуется использовать Access Control Lists (ACL) для предоставления прав как пользователю веб-сервера (www-data), так и пользователю-разработчику, что предотвращает конфликты при генерации кэша через CLI и веб-интерфейс [7].

На четвёртом этапе настраивается подключение к базе данных. Для локальной разработки часто используется SQLite, не требующая отдельного сервера. Однако при моделировании production-среды предпочтение отдаётся PostgreSQL или MySQL. Параметры подключения указываются в файле env.local, после чего выполняются команды doctrine:database:create и doctrine:migrations:migrate, инициализирующие схему данных.

Завершающим шагом является проверка работоспособности приложения через браузер или CLI-инструменты (например, curl). В production-режиме обязательно выполняется команда cache:clear --env=prod для генерации оптимизированного кэша.

Помимо базовой функциональности, локальное окружение для разработки Symfony должно быть настроено на максимальную производительность, которая становится определяющим фактором при работе с крупными приложениями или при необходимости симулировать высокую нагрузку. Оптимизация производительности веб-сервера и PHP-FPM непосредственно в среде Ubuntu требует тщательного расчета и настройки параметров на основе аппаратных характеристик системы, в первую очередь объема оперативной памяти (RAM) и количества процессорных ядер (CPU). Для Apache, работающего в режиме mpm_event, ключевыми параметрами являются MaxRequestWorkers, определяющий максимальное количество одновременных соединений, и ThreadsPerChild, демонстрирующий число потоков на один рабочий процесс. Значения этих параметров, определяемые по специальным методикам, служат отправной точкой и должны быть скорректированы после проведения нагрузочного тестирования с использованием таких инструментов, как Apache Bench (ab), с последующим мониторингом использования ресурсов с помощью htop или ps_mem.py. Корректная настройка этих параметров позволяет избежать перегрузки сервера и обеспечить стабильную работу приложения под нагрузкой.

Заключение

Проведённое исследование показало, что развёртывание Symfony 7 в локальной среде Linux требует строгого соблюдения архитектурных и операционных принципов, заложенных разработчиками фреймворка. Ключевыми элементами успешной реализации являются: корректная настройка веб-сервера с изоляцией публичного каталога, управление зависимостями вне целевой системы, грамотная конфигурация переменных окружения и обеспечение безопасных прав доступа к файловой системе. Предложенный подход позволяет не только эффективно тестировать приложения на этапе разработки, но и формировать основу для последующей автоматизации развёртывания в промышленных условиях. В перспективе целесообразно исследовать интеграцию описанной методики с инструментами контейнеризации (Docker) и системами управления конфигурациями (Ansible), что повысит её применимость в распределённых исследовательских коллективах.

Литература

1. Ubuntu Community Help Wiki. PHP. 2024. – URL : https://help.ubuntu.com/community/PHP (date of application 31.10.2025).
2. Du Y. Realization of Computer Intensive Network Technology in Symfony Architecture of Student Psychological Mutual Aid Platform // International Conference on Sustainable Computing and Data Communication Systems, ICSCDS 2022 – Proceedings, Erode, April 7–9, 2022. – Erode, 2022. – P. 1106–1110. DOI: 10.1109/ICSCDS53736.2022.9760838 EDN UAURWS
3. Garbarz P. Comparative analysis of PHP frameworks on the example of Laravel and Symfony / P. Garbarz, M. Plechawska-Wójcik // Journal of Computer Sciences Institute. – 2022. – Vol. 22. – P. 18–25. DOI: 10.35784/jcsi.2781 EDN XRUWBG
4. Кочнев А.А. Применение фреймворка Symfony и PHP в разработке веб-приложений // Актуальные исследования. – 2023. – № 23-1(153). – С. 65–70. EDN PUCEES
5. Symfony. Symfony 7.0 Release Notes. 2023. – URL : https://symfony.com/releases/7.0 (date of application 31.10.2025).
6. Doctrine Project. Doctrine ORM Documentation. 2020. – URL : https://www.doctrine-project.org/projects/doctrine-orm/en/3.0/index.html (date of application 31.10.2025).
7. Symfony Documentation. Web Server Configuration. 2024. – URL : https://symfony.com/doc/current/setup/web_server_configuration.html (date of application 31.10.2025).

References

1. Ubuntu Community Help Wiki. PHP. 2024. – URL : https://help.ubuntu.com/community/PHP (date of application 31.10.2025).
2. Du Y. Realization of Computer Intensive Network Technology in Symfony Architecture of Student Psychological Mutual Aid Platform // International Conference on Sustainable Computing and Data Communication Systems, ICSCDS 2022 – Proceedings, Erode, April 7–9, 2022. – Erode, 2022. – P. 1106–1110. DOI 10.1109/ICSCDS53736.2022.9760838. – EDN UAURWS
3. Garbarz P. Comparative analysis of PHP frameworks on the example of Laravel and Symfony / P. Garbarz, M. Plechawska-Wójcik // Journal of Computer Sciences Institute. – 2022. – Vol. 22. – P. 18–25. DOI: 10.35784/jcsi.2781 EDN XRUWBG
4. Kochnev A.A. Using the Symfony framework and PHP in developing web applications // Current research. – 2023. – № 23-1(153). – P. 65–70. EDN PUCEES
5. Symfony. Symfony 7.0 Release Notes. 2023. – URL : https://symfony.com/releases/7.0 (date of application 31.10.2025).
6. Doctrine Project. Doctrine ORM Documentation. 2020. – URL : https://www.doctrine-project.org/projects/doctrine-orm/en/3.0/index.html (date of application 31.10.2025).
7. Symfony Documentation. Web Server Configuration. 2024. – URL : https://symfony.com/doc/current/setup/web_server_configuration.html (date of application 31.10.2025).