

**Макаров Кирилл Дмитриевич,**  
студент магистратуры 1 курса гр. Ф2-ИСТ-51мз,  
ФГОБУ ВО «Поволжский государственный университет  
телекоммуникаций и информатики»

Makarov Kirill Dmitrievich,  
1st year master's student of the F2-IST-51mz group,  
FGOBU in «Volga State University of Telecommunications,  
and Informatics»

**Тучкова Анна Сергеевна,** ст. преподаватель,  
ФГОБУ ВО «Поволжский государственный университет  
телекоммуникаций и информатики»

Tuchkova Anna Sergeevna, Senior Lecture, art. Lehrer,  
FGOBU in «Volga State University of Telecommunications,  
and Informatics»

**Секлетова Наталья Николаевна,** к.п.н., доцент,  
ФГОБУ ВО «Поволжский государственный университет  
телекоммуникаций и информатики»

Sekletova Natalia Nikolaevna, k.p., associate,  
FGOBU in «Volga State University of Telecommunications  
and Informatics»

## **КОНТЕЙНЕРИЗАЦИЯ ПРИЛОЖЕНИЙ: ПРЕИМУЩЕСТВА И ПЛАТФОРМЫ ДЛЯ РАЗВЁРТЫВАНИЯ MANAGING APPLICATIONS WITH CONTAINERIZATION**

**Аннотация.** В статье анализируются современные методы контейнеризации. Рассмотрены преимущества и проблемные моменты использования контейнеризации. Уделено внимание основным типам контейнеризации. Обозначены наиболее популярные платформы контейнеризации.

**Abstract.** This article analyzes modern containerization methods. It examines the advantages and challenges of using containerization. It focuses on the main types of containerization and identifies the most popular containerization platforms.

**Ключевые слова:** Контейнеризация, веб-приложение, автоматизация, масштабируемость, гибкость, портативность.

**Keywords:** Containerization, web application, automation, scalability, flexibility, portability.

Современные ИТ-компании стремятся к максимальной эффективности процессов разработки и развертывания программного обеспечения. Одним из наиболее значимых шагов в этом направлении стало активное внедрение контейнеризации, позволяющей добиться гибкости, масштабируемости и сокращения затрат на управление инфраструктурой [1].

Процесс упаковки приложения, объединяющий библиотеки, файлы, настройки в изолированные виртуальные контейнеры, называется контейнеризацией приложений. Приложение вместе с его окружением упаковывается в единый пакет и запускается затем там, где это необходимо. При переносе не возникает проблем совместимости.



Благодаря контейнеризации приложения получают непосредственно те ресурсы, которые им необходимы и работают независимо от операционной системы хоста, так как удаляется уровень операционной системы из замкнутой среды.

Итак, контейнеризация простыми словами, что это? Создается файл, который называется контейнером. Представим себе бытовой ящик, куда складываем необходимые вещи, а затем берем этот ящик и переносим в другую комнату или выезжаем с ним на природу, в другой город и т.д.

Основными типами контейнеров являются контейнеры приложений и системные контейнеры.

Контейнеры приложений подходят для неизменяемых инфраструктур, они содержат один процесс, управляющий одним приложением.

Системные контейнеры служат для размещения операционной системы, их так и называют контейнеры операционных систем.

Обычно контейнеры операционной системы используются для больших, неделимых и традиционных приложений и могут содержать в себе конфигурацию, архитектуру и инструментарий системы. Такие контейнеры часто создаются с помощью шаблонов или образов, которые точно определяют их структуру и содержимое. Таким образом, становится возможным создание контейнеров с идентичным окружением, включающим одинаковые версии пакетов и конфигураций для всех экземпляров [2].

Контейнеризация имеет ряд неоспоримых преимуществ:

переносимость, портативность, легкость, компактность, изоляция;

быстрое развертывание, легкое горизонтальное масштабирование;

высокий уровень эффективности и бесперебойная работа в различных средах;

использование микросервисной структуры;

уменьшение затрат на сервер и повышение его результативности;

сокращение периода разработки продукта;

упрощенное тестирование однотипных элементов, выявление и купирование ошибок.

Несомненно при контейнеризации компаниям потребуются дополнительные затраты на обучение специалистов и на создание дополнительных ящиков, при желании запустить большее количество контейнеров на одном хосте. При увеличении контейнеров усложняется контроль за их работой, возникает проблема с нехваткой ресурсов. Могут возникнуть проблемы в связи с коротким жизненным циклом, так как при закрытии ящика все данные в нем стираются.

Главным минусом контейнеризации является неизменяемый объем тары.

Безопасность при контейнеризации достигается за счет изоляции и управления доступом. Обязателен систематический мониторинг и регулярное обновление систем. Важно понимать, что защиты требуют и контейнеры, и инфраструктура, в которой они работают.

Контейнеризация помогает программистам разрабатывать очень гибкие и масштабируемые продукты, позволяет разворачивать их быстрее и безопаснее и обеспечивает поддержку приложений [3].

Docker и Kubernetes являются платформами контейнеризации, наиболее популярными среди прочих. Docker используется для создания и управления отдельными контейнерами, а Kubernetes, служит для автоматизации больших кластеров контейнеров.

Разумеется, технологии контейнеризации активно развиваются в части улучшения безопасности, интеграции с другими технологиями и расширения экосистемы инструментов.

#### **Список литературы:**

1. Лю, Ю., Борисов, В. И. Контейнеризация и автоматизация развертывания веб-приложений: анализ и практическая реализация [Текст]. Статья - Comp. nanotechnol., 2025, том 12, выпуск 1, с. 194-201.



**РАЗДЕЛ:** Инженерное дело, технологии и технические науки

Направление: Технические науки

---

2. Контейнеризация [Электронный ресурс] - Электронные текстовые данные. - Режим доступа: <https://cloud.obit.ru/glossarij/kontejnerizacija/>, свободный - Загл. с экрана.

3. Что такое контейнеризация [Электронный ресурс] - Электронные текстовые данные.

- Режим доступа: <https://yandex.cloud/ru/docs/glossary/containerization>, свободный. - Загл. с экрана.

