УДК 004.056.5

ИНТЕГРАЦИЯ DEVSECOPS И КОНЦЕПЦИИ ZERO TRUST ДЛЯ ПОВЫШЕНИЯ БЕЗОПАСНОСТИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В УСЛОВИЯХ ЦИФРОВОЙ ТРАНСФОРМАЦИИ

Блинов А.В. (ИТМО)
Научный руководитель – доктор технических наук, доцент Беззатеев С.В.
(ИТМО)

Введение. Современная цифровая трансформация требует от организаций не только ускорения разработки программного обеспечения, но и обеспечения высокого уровня информационной безопасности. Увеличение числа кибератак и уязвимостей, связанных с программными продуктами, подчеркивает необходимость пересмотра традиционных методов защиты [1]. На сегодняшний день концепция DevSecOps, объединяющая разработку, эксплуатацию и безопасность, становится стандартом для многих компаний [2]. Однако она часто упускает аспект динамического управления доступом, что особенно актуально в условиях увеличения удаленной работы и облачных технологий.

Концепция Zero Trust [3], предполагающая минимизацию доверия на всех уровнях сети и приложений, предлагает эффективное дополнение к DevSecOps. Анализ отечественного и зарубежного опыта показывает, что интеграция этих подходов может значительно повысить уровень защищенности программного обеспечения на всех этапах его жизненного цикла. Несмотря на это, отсутствует единая методология внедрения таких интегрированных подходов, что и составляет ключевую научную проблему.

Основная часть. Для повышения безопасности разработки ПО предлагается синергия DevSecOps и Zero Trust, которая позволяет:

- Интегрировать безопасность на каждом этапе разработки (shift-left security);
- Внедрить принципы Zero Trust для защиты инфраструктуры разработки, таких как контроль доступа на основе идентификационных данных и непрерывная аутентификация.

Ключевые элементы решения:

1. Автоматизация процессов безопасности: внедрение статического (SAST) и динамического (DAST) анализа кода на этапах CI/CD для быстрого обнаружения уязвимостей.
2. Управление доступом: использование политики минимальных привилегий (Least Privilege) для разработчиков и сервисов в сочетании с мультифакторной аутентификацией (MFA).
3. Мониторинг и аудит: внедрение инструментов анализа журналов, таких как Kubernetes Audit Logs, и централизованных систем наблюдения для раннего обнаружения аномалий.
4. Использование контейнерных технологий и Policy Engines: применение Kyverno или OPA Gatekeeper для автоматического внедрения Zero Trust политик в Kubernetes-кластерах.
5. Постоянное обучение: регулярное повышение квалификации сотрудников по вопросам безопасной разработки и применения принципов Zero Trust.

Эффективность предлагаемого подхода достигается за счет обеспечения многоуровневой защиты, масштабируемости и адаптации к динамическим условиям цифровой трансформации.

Выводы. Интеграция DevSecOps и концепции Zero Trust предоставляет новый уровень безопасности разработки ПО, позволяя выявлять и устранять уязвимости на ранних этапах, а также защищать инфраструктуру разработки от современных угроз. Практическое использование предложенного подхода целесообразно в компаниях, работающих с чувствительными данными (финансы, госучреждения, медицина).

Список использованных источников:

1. Тулеубаева А.А., Норкина А.Н. Современные проблемы информационной безопасности в разработке программного обеспечения // Угрозы и риски финансовой безопасности в контексте цифровой трансформации: Материалы VII Международной научно-практической конференции Международного сетевого института в сфере ПОД/ФТ, Москва, 24 ноября 2021 года. – Москва: Национальный исследовательский ядерный университет "МИФИ", 2021. С. 670-676.

2. Селиверстов С.Д., Мироненко Ю.В. Обзор методологии DevSecOps и ее ключевых инструментов для внедрения и обеспечения безопасной разработки ПО // Студент года 2024 – сборник статей Международного научно-исследовательского конкурса. Пенза, 2024. С. 107-111.

3. Rose S., Borchert O., Mitchell S. and Connelly S. Zero Trust Architecture // National Institute of Standards and Technology. Gaithersburg. 2020. DOI 10.6028/nist.sp.800-207