БЕЗОПАСНОСТЬ ЦИФРОВЫХ ТЕХНОЛОГИЙ. – 2025. – № 1 (116). – 28–40

МЕТОДЫ И СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ, ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

УДК 004 DOI: 10.17212/2782-2230-2025-1-28-40

ВЫХОД ЗА ПРЕДЕЛЫ СРЕДЫ КОНТЕЙНЕРИЗАЦИИ В СОВРЕМЕННЫХ IT-РЕШЕНИЯХ*

П.И. МАРТЕНС[1], А.С. БЕРЕЖНОЙ[2]

[1] *630090, РФ, г. Новосибирск, ул. Николаева, 11, ООО «Юзергент», ведущий инженер-разработчик. E-mail: regularitcat@gmail.com*
[2] *630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет, инженер кафедры защиты информации. E-mail: kaf_zi@corp.nstu.ru*

Контейнеризация решает проблему изоляций приложений, однако вносит новые риски безопасности. Угроза утечек данных и несанкционированного доступа через среду контейнеризации является актуальной проблемой для организаций, что требует применения современных подходов к защите. Контейнеры, использующие namespace и cgroups, подвержены рискам эксплуатации уязвимостей. Вектор атак может включать обход изоляции, манипуляции с контейнерными процессами, а также с ресурсами, выделенными из хостовой ОС.

Ключевые слова: контейнер, docker, runC, namespace, cgroups, вектор атаки, изоляция контейнера, CVE

ВВЕДЕНИЕ

С развитием технологий контейнеризации становится всё актуальнее вопрос обеспечения безопасности в различных средах. Контейнеры, такие как Docker, Lxc, Kubernetes, широко используются для изоляции приложений и упрощения разработки, однако они также представляют собой потенциальную угрозу для безопасности данных и систем.

В течение времени с внедрением контейнеризации всё чаще происходит эксплуатация уязвимостей в namespace и cgroups Linux, что позволяет злоумышленникам выйти за пределы виртуальной среды и получить доступ к хост-

* Статья получена 23 января 2025 г.

системам, что ставит под угрозу безопасность всей инфраструктуры. Одной из ключевых проблем является неправильная настройка контейнеров, что создает дополнительные риски для организации. Для решения этих проблем необходимо использовать методы тестирования, анализа уязвимостей и разработки эффективных механизмов защиты. Ключевую роль в этом процессе играет внедрение систем мониторинга, настройки прав доступа и постоянного контроля.

Целью работы является исследование угроз, связанных с нарушением изоляции контейнеров, а также анализ методов защиты от несанкционированного доступа.

Для достижения поставленной цели были выделены следующие задачи:

- оценить основные риски безопасности контейнеров и уязвимости, которые могут быть использованы для выхода за пределы изоляции;
- исследовать методы защиты, направленные на предотвращение эксплуатации уязвимостей контейнеров.

1. УГРОЗЫ БЕЗОПАСНОСТИ КОНТЕЙНЕРОВ

Контейнеризация, основанная на таких технологиях, как Docker, Lxc, Kubirnetis, значительно улучшает гибкость и эффективность разработки и развертывания приложений. Однако контейнеры также подвержены различным угрозам, связанным с нарушением изоляции и безопасностью данных. Одной из основных проблем является эксплуатация уязвимостей контейнеров, что может привести к несанкционированному доступу к хост-системам [1].

Для исследования угроз безопасности контейнеров был выбран сценарий, моделирующий попытки выхода за пределы контейнерной изоляции с использованием namespace и cgroups. В таком сценарии используется специальное ПО для мониторинга и анализа контейнерных систем, которое включает:

- мониторинг контейнерных процессов и их активности;
- контроль за доступом и правами контейнеров;
- анализ сетевого трафика и активностей, связанных с контейнерами;
- обнаружение попыток эксплуатации уязвимостей в изоляции контейнеров.

Контейнеры, как правило, запускаются с минимальными правами и ограниченными ресурсами, однако неправильная настройка или недостаточный контроль могут позволить злоумышленникам использовать уязвимости в сетевых или файловых системах для получения доступа к хост-системам [2].

Для повышения безопасности необходимо учитывать риски, связанные с настройкой и эксплуатацией контейнеров. Важную роль играют системы контроля и мониторинга, которые позволяют минимизировать потенциальные угрозы и своевременно обнаруживать аномальные активности [3].

2. МЕТОДЫ ЗАЩИТЫ КОНТЕЙНЕРОВ

Для эффективной защиты контейнерных технологий необходимо внедрить комплексный подход, включающий использование современных технологий изоляции, правильную настройку доступа, мониторинг контейнерных систем и защиту ресурсов. Эти методы позволяют минимизировать риски безопасности, такие как нарушение изоляции контейнеров, несанкционированный доступ и чрезмерное использование ресурсов. Рассмотрим основные методы защиты, применяемые для повышения безопасности контейнерных сред.

Namespace и cgroup в Linux являются основными механизмами изоляции и управления ресурсами контейнеров. Namespace (например, PID, NET, MNT) обеспечивают разделение процессов, сетевых интерфейсов, файловых систем и других ресурсов между контейнерами и хост-системой, что позволяет предотвратить доступ контейнеров друг к другу и к хост-системе. Каждый контейнер получает свой уникальный контекст, ограничивающий взаимодействие с другими контейнерами и хостом, что минимизирует возможности злоумышленников для выхода за пределы контейнера и доступа к важным системным данным [4].

Cgroup, в свою очередь, позволяют эффективно контролировать использование ресурсов контейнерами, таких как CPU, память, диск и сеть. Эти механизмы обеспечивают изоляцию и предотвращают доминирование одного контейнера над другими, ограничивая доступ к ресурсам хост-системы. Это особенно важно в многоконтейнерных средах, в которых важно обеспечить стабильную работу всех контейнеров, предотвращая перегрузку и нестабильность системы (рис. 1).

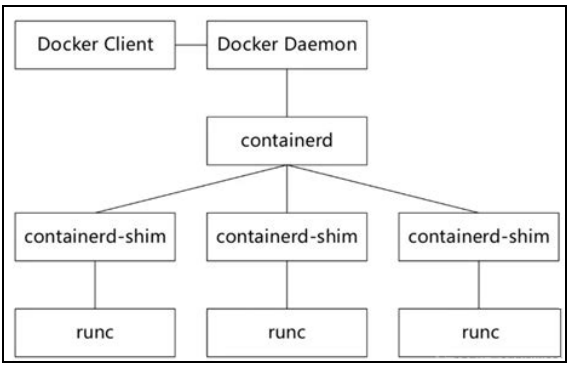

Рис. 1. Описание работы контейнера и взаимодействие с пользовательскими процессами через runC

2.1. НАСТРОЙКА ДОСТУПА И МОНИТОРИНГ

Одним из важнейших аспектов защиты контейнеров является правильная настройка прав доступа. Контейнеры должны работать с минимальными привилегиями, чтобы ограничить возможности злоумышленников в случае успешной атаки на контейнер. Важно использовать подходы, основанные на принципе наименьших привилегий, что означает предоставление контейнерам только тех прав, которые необходимы для выполнения их функций. Например, не стоит предоставлять контейнерам доступ к корневой файловой системе или возможности выполнить привилегированные операции, такие как загрузка ядра или изменение настроек системы [5].

Для обеспечения надежного контроля доступа важно внедрить механизмы аутентификации и авторизации на уровне контейнеров. Системы, такие как LDAP или интеграция с Kubernetes RBAC (Role-Based Access Control), позволяют управлять правами доступа, предоставляя доступ только уполномоченным пользователям и процессам. При этом важно регулярно обновлять и пересматривать эти права, чтобы минимизировать риски от устаревших или несанкционированных разрешений [6].

Также важную роль в защите контейнеров играет мониторинг. Современные инструменты мониторинга, такие как Prometheus, Grafana или использование SIEM (Security Information and Event Management), позволяют отслеживать деятельность контейнеров в реальном времени. Эти системы помогают оперативно обнаруживать аномальные действия и угрозы, такие как попытки выхода за пределы контейнера, использование уязвимостей или изменение конфигураций контейнеров. Они также помогают в анализе логов контейнеров, выявляя подозрительные события и предотвращая потенциальные атаки [7].

Защита ресурсов контейнеров

Защита выделенных ресурсов контейнеров из хоста является неотъемлемой частью общей стратегии безопасности. В условиях контейнеризации большое количество приложений может работать в рамках одной хост-системы, что требует эффективного управления доступом к системным ресурсам. Для этого используются механизмы контроля, такие как cgroups, которые обеспечивают ограничение потребления ресурсов контейнерами, предотвращая использование всех доступных ресурсов одним контейнером.

Контейнеры должны быть ограничены в потреблении CPU, памяти, дисков и сетевых ресурсов. Это не только предотвращает перегрузку хост-системы, но и помогает изолировать контейнеры друг от друга, так как

злоумышленники могут чрезмерно потреблять ресурсы в атаках вида DoS (Denial of Service) и других.

Также важным аспектом защиты ресурсов является управление доступом к хостовой файловой системе. Контейнеры должны работать в ограниченной файловой системе без доступа к чувствительным данным хоста. Использование технологий, таких как volume mounts (монтирование томов) и read-only файловых систем, может помочь ограничить возможность контейнера взаимодействовать с критичными частями файловой системы хоста, предотвращая возможные утечки данных и заражение хост-системы [8].

3. РАЗРАБОТКА СЦЕНАРИЯ ДЕЙСТВИЯ НАРУШИТЕЛЯ

3.1. ОПИСАНИЕ СЦЕНАРИЯ

В данном сценарии действия нарушителя показана уязвимость CVE-2019-5736, которая была обнаружена в Docker. Эта уязвимость позволяла злоумышленникам с правами доступа внутри контейнера выполнить произвольный код на хост-системе. Уязвимость возникала из-за того, что Docker неправильно обрабатывал монтирование образов, что позволяло контейнерам обращаться к устройствам хоста и выполнять команды с привилегиями хост-системы [9].

Этапы атаки

- Злоумышленник использует возможность выполнения команд в контейнере для загрузки вредоносного кода. Это может быть выполнено через слабо защищенные API, такие как Docker REST API, или через уязвимости в приложении внутри контейнера, к примеру RCE, SQLi, XXE, SSTI [10].
- Внутри контейнера атакующий использует уязвимость в Docker, которая позволяет ему записать в образ контейнера произвольный код. Он может перезаписать образ контейнера с вредоносным исполняемым кодом или использовать привилегии контейнера для взаимодействия с устройствами хоста.
- После использования уязвимости злоумышленник может нарушить изоляцию контейнера, получив доступ к хост-системе. Это дает ему возможность выполнять команды на уровне хоста, что нарушает принцип изоляции контейнеров и открывает доступ к важным данным (рис. 2).

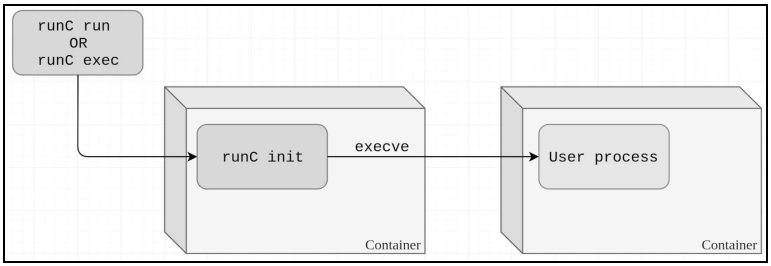

Рис. 2. runC инициируют запуск контейнера

- После получения доступа к хост-системе злоумышленник может попытаться повысить свои привилегии с помощью локальных уязвимостей, чтобы выполнить вертикальное движение и продолжить атаки на другие части сети или системы.

3.2. ПЕРЕЧЕНЬ УЯЗВИМОСТЕЙ

Исходя из описания сценария были определены следующие уязвимости и их последствия:

- CVE-2019-5736 – уязвимость позволяет атакующему перезаписать Docker контейнер, что дает ему возможность выполнить произвольный код на хосте. Для ее детектирования можно использовать инструменты мониторинга контейнеров, такие как Sysdig или Falco, которые отслеживают аномальные изменения в контейнерах и предупреждают о попытках записывать данные в контейнерные образы.
- Неверная настройка контейнеров (например, флаг -- privileged), связанные с избыточными привилегиями контейнеров, позволяют злоумышленникам обойти изоляцию. Для предотвращения и детектирования таких угроз необходимо избегать использования флага --privileged при запуске контейнеров и регулярно сканировать контейнеры на наличие неправильно настроенных прав доступа [11].

Подготовка уязвимой среды

Для тестирования данной уязвимости был использован Docker версии до 18.09.2, которая содержит уязвимость. В этой версии runC (библиотека для запуска контейнеров, используемая Docker) не выполняет корректные проверки на доступность к системным файлам и монтированию. В результате злоумышленник с правами root внутри контейнера может получить доступ к системным ресурсам хост-системы.

Запуск контейнера с правами root

Контейнер был запущен с правами root, что является базовой конфигурацией, которую можно использовать для эксплуатации уязвимости. В данном случае контейнер был настроен так, чтобы использовать привилегированные режимы, такие как --privileged, что позволило бы злоумышленнику выполнить действия на хосте.

Эксплуатация уязвимости

Используя уязвимость CVE-2019-5736, атакующий может перезаписать бинарный файл runc на хост-системе, что позволяет ему получить доступ к хосту с привилегиями root [12].

Действия злоумышленника

- Запись в /proc/self/exe (механизм в Linux, который позволяет ссылаться на исполняемый файл текущего процесса), чтобы перезаписать исполняемый файл runC на хосте. Эта операция дает злоумышленнику возможность выполнять произвольные команды на хост-системе (рис. 3).

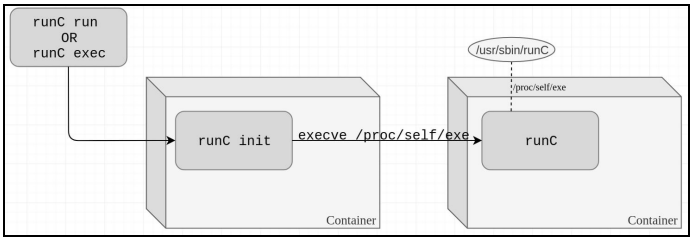

Рис. 3. Перезапись runC

- Получение контроля над хост-системой после перезаписи файла runC. Злоумышленник может выполнить произвольный код с правами root на хост-системе. Это позволяет ему получить полный контроль над системой, что эквивалентно выходу за пределы контейнера [13].

3.3. НЕЙТРАЛИЗАЦИЯ УГРОЗЫ

Для предотвращения этой уязвимости и ее последствий были предложены следующие меры.

- Основной метод нейтрализации уязвимости заключается в обновлении до версии Docker 18.09.2 или выше, в которых данная уязвимость была исправлена. В этих версиях механизмы изоляции контейнеров были улуч-

шены и была добавлена защита от попыток перезаписать важные системные файлы.

- Контейнеры должны быть настроены с минимальными правами доступа. Использование привилегированных контейнеров (--privileged) является небезопасным, поскольку дает контейнеру доступ ко всем системным ресурсам хоста. Поэтому рекомендуется избегать использования привилегированных контейнеров и применять ограничения на уровне прав доступа [14].
- Для дополнительной защиты рекомендуется использовать системы безопасности SELinux или AppArmor для контейнеров. Эти технологии позволяют ограничить привилегии контейнеров и обеспечить строгую изоляцию, даже если контейнер был скомпрометирован.
- Регулярное сканирование контейнерных образов и систем на наличие уязвимостей с использованием инструментов мониторинга и безопасности, таких как Falco или Sysdig. Эти системы могут отслеживать подозрительную активность внутри контейнеров и на хосте, что позволяет оперативно выявлять попытки эксплуатации уязвимостей [15].

3.4. АПРОБАЦИЯ СЦЕНАРИЯ

Сценарий эксплуатации уязвимости CVE-2019-5736 показал, что данная уязвимость может быть использована для получения полного контроля над хост-системой, если контейнер настроен с избыточными правами и использует уязвимые версии Docker. Применение обновлений, использование строгих политик безопасности и ограничение прав контейнеров являются основными мерами по защите от этой уязвимости.

ЗАКЛЮЧЕНИЕ

В сценарии действий злоумышленника, направленном на нарушение безопасности контейнеров, используются различные тактики и методы эксплуатации, включая использование уязвимостей в контейнерной изоляции, перезапись системных файлов через runC и выход за пределы контейнера с последующим получением root-доступа на хост-системе. Этот процесс включает использование уязвимости CVE-2019-5736 для перезаписи бинарных файлов контейнера и нарушения изоляции между контейнерами и хост-системой. Эксплуатация таких уязвимостей может быть детектирована с помощью систем мониторинга, таких как Falco и Sysdig, которые отслеживают аномальные действия и попытки перезаписать файлы или изменить права доступа.

Меры защиты от данной уязвимости включают использование последних версий Docker, в которых уязвимость была исправлена, а также отказ от использования привилегированных контейнеров и строгую настройку прав доступа. Использование дополнительных механизмов безопасности, таких как SELinux и AppArmor, помогает уменьшить риски эксплуатации уязвимостей и повысить безопасность контейнерных систем. Также важно регулярно обновлять контейнерные образы и использовать средства сканирования для обнаружения известных уязвимостей в образах.

Тестирование и апробация таких сценариев с использованием различных инструментов и подходов к мониторингу позволяют эффективно исследовать воздействие уязвимостей и выработать эффективные стратегии для их устранения. Такие подходы позволяют выявить слабые места в системе безопасности и улучшить защиту контейнерных технологий от потенциальных угроз.

СПИСОК ЛИТЕРАТУРЫ

1. *Цыбенко О.С.* Контейнерная безопасность // E-Scio. – 2023. – № 5 (80). – С. 158–165. – URL: https://cyberleninka.ru/article/n/konteynernaya-bezopasnost (дата обращения: 05.03.2025).

2. *Мельникова А.Е., Рыжков В.А.* Использование технологии контейнеризации при безопасной разработке программного обеспечения // Материалы Второго Международного научно-практического форума по экономической безопасности «VII ВСКЭБ». – М., 2021. – С. 75–83.

3. Управление информационной безопасностью организации интегрированной структуры на основе выделенного сервера с контейнерной виртуализацией / В.А. Липатников, А.А. Шевченко, А.Д. Яцкин, Е.Г. Семенова // Информационно-управляющие системы. – 2017. – № 4 (89). – С. 67–76. – DOI: 10.15217/issn1684-8853.2017.4.67.

4. *Бондаренко А.С., Зайцев К.С.* Использование систем управления контейнерами для построения распределенных облачных информационных систем с микросервисной архитектурой // Международный журнал гуманитарных и естественных наук. – 2022. – № 1-1. – С. 62–65. – DOI: 10.24412/2500-1000-2022-1-1-62-65.

5. *Голушко А.П., Жуков В.Г.* Информационная безопасность контейнеров Docker на этапе их создания и запуска // Решетневские чтения: материалы XXVI Международной научно-практической конференции, посвященной памяти генерального конструктора ракетно-космических систем академика М.Ф. Решетнева. Т. 2. – Красноярск, 2022. – С. 389–392.

6. *Липатова С.Е., Белов Ю.С.* Практики обеспечения кибербезопасности в Kubernetes // E-Scio. – 2022. – № 1 (64). – С. 489–497. – URL: https://cyberleninka.ru/article/n/praktiki-obespecheniya-kiberbezopasnosti-v-kubernetes (дата обращения: 05.03.2025).

7. *Котенко И.В., Саенко И.Б., Юсупов Р.М.* Новое поколение систем мониторинга и управления инцидентами безопасности // Информатика, телекоммуникации и управление. – 2014. – № 3 (198). – С. 7–18. – URL: https://cyberleninka.ru/article/n/novoe-pokolenie-sistem-monitoringa-i-upravleniya-intsidentami-bezopasnosti (дата обращения: 05.03.2025).

8. *Гатчин Ю.А., Теплоухова О.А.* Реализация контроля целостности образа операционной системы, загружаемого по сети на тонкий клиент // Научно-технический вестник информационных технологий, механики и оптики. – 2015. – № 6. – С. 1115–1121. – URL: https://cyberleninka.ru/article/n/realizatsiya-kontrolya-tselostnosti-obraza-operatsionnoy-sistemy-zagruzhaemogo-po-seti-na-tonkiy-klient (дата обращения: 05.03.2025).

9. Breaking out of Docker via runC – Explaining CVE-2019-5736. – URL: https://unit42.paloaltonetworks.com/breaking-docker-via-runc-explaining-cve-2019-5736/ (accessed: 05.03.2025).

10. *Шатурный М.В.* Анализ актуальных угроз и разработка подходов к защите веб-приложений // Инженерный вестник Дона. – 2024. – № 7 (115). – С. 109–120. – URL: https://cyberleninka.ru/article/n/analiz-aktualnyh-ugroz-i-razrabotka-podhodov-k-zaschite-veb-prilozheniy (дата обращения: 05.03.2025).

11. *Ганжур М.А., Дьяченко Н.В.* Мониторинг целостности файлов // Молодой исследователь Дона. – 2021. – № 2 (29). – С. 17–19. – URL: https://cyberleninka.ru/article/n/monitoring-tselostnosti-faylov (дата обращения: 05.03.2025).

12. Digital Twins and Cyber Security – solution or challenge? / D. Holmes, M. Papathanasaki, L. Maglaras, M.A. Ferrag, S. Nepal, H. Janicke // 2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM). – IEEE, 2021. – P. 1–8. – DOI: 10.1109/SEEDA-CECNSM53056.2021.9566277.

13. Уязвимость CVE-2019-5736 в runc, позволяющая получить права root на хосте. – 2019. – URL: https://habr.com/ru/companies/flant/articles/439964/ (дата обращения: 05.03.2025).

14. *Гаврилов А.* 5000 слов о защите контейнеров // Jetinfo. – 2021. – № 7–8 (307–308). – С. 44–57. – URL: https://www.jetinfo.ru/5000-slov-o-zashhite-kontejnerov/ (дата обращения: 05.03.2025).

15. National Vulnerability Database. CVE-2019-5736 Detail. – URL: https://nvd.nist.gov/vuln/detail/cve-2019-5736 (accessed: 05.03.2025).

Мартенс Павел Иванович, ведущий инженер-разработчик ООО «Юзергейт». Основное направление научных исследований – математическое моделирование в информационной безопасности. E-mail: regularitcat@gmail.com

Бережной Антон Сергеевич, ассистент кафедры защиты информации Новосибирского государственного технического университета. Область научных интересов – информационная безопасность. E-mail: kaf_zi@corp.nstu.ru

DOI: 10.17212/2782-2230-2025-1-28-40

Moving beyond containerization in today's it solutions*

P.I. Martens[1], A.S. Bereznoy[2]

[1] *LLC "Uzergent", 11 Nikolaeva Street, Novosibirsk, 630090, Russian Federation, lead development engineer. E-mail: regularitcat@gmail.com*
[2] *Novosibirsk State Technical University, 20 Karl Marx Prospekt, Novosibirsk, 630073, Russian Federation, engineer of the Department of Information Security. E-mail: kaf_zi@corp.nstu.ru*

The Containerization solves the problem of application isolations, but introduces new security risks. The threat of data leaks and unauthorized access to the containerization environment is an urgent problem for organizations, which requires the use of modern approaches to protection. Containers using namespace and cgroups are at risk of exploiting vulnerabilities. The attack vector can include bypassing isolation, manipulating container processes, as well as resources allocated from the host OS.

Keywords: container, docker, runC, namespace, cgroups, attack vector, container isolation, CVE

REFERENCES

1. Tsybenko O.S. Konteinernaya bezopasnost' [Container Security]. *E-Scio*, 2023, no. 5 (80), pp. 158–165. (In Russian). Available at: https://cyberleninka.ru/article/n/konteynernaya-bezopasnost (accessed 05.03.2025).

2. Mel'nikova A.E., Ryzhkov V.A. [Using containerization technology for secure software development]. *Materialy Vtorogo Mezhdu-narodnogo nauchno-prakticheskogo foruma po ekonomicheskoi bezopasnosti «VII VSKEB»* [Materials of the second international scientific and practical forum on economic security "VII VSKEB"]. Moscow, 2021, pp. 75–83. (In Russian).

* *Received 23 January 2025.*

3. Lipatnikov V.A., Shevchenko A.A., Yatskin A.D., Semenova E.G. Upravlenie informatsionnoi bezopasnost'yu organizatsii integrirovannoi struktury na osnove vydelennogo servera s konteinernoi virtualizatsiei [Information security management of integrated structure organization based on a dedicated server with container visualization]. *Informatsionno-upravlyayushchie sistemy = Information and Control Systems*, 2017, no. 4 (89), pp. 67–76. DOI: 10.15217/issn1684-8853.2017.4.67.

4. Bondarenko A.S., Zaytsev K.S. Ispol'zovanie sistem upravleniya konteinerami dlya postroeniya raspredelennykh oblachnykh informatsionnykh sistem s mikroservisnoi arkhitekturoi [Using container management systems to build distributed cloud information systems with microservice architecture]. *Mezhdunarodnyi zhurnal gumanitarnykh i estestvennykh nauk = International Journal of Humanities and Natural Sciences*, 2022, no. 1-1, pp. 62–65. DOI: 10.24412/2500-1000-2022-1-1-62-65.

5. Golushko A.P., Zhukov V.G. [Information security of Docker containers at the stage of creation and launch]. *Reshetnevskie chteniya* [Reshetnevsky readings]. Materials of the XXVI International Scientific and Practical Conference dedicated to the memory of the general designer of rocket and space systems, academician M.F. Reshetnev. Vol. 2. Krasnoyarsk, 2022, pp. 389–392. (In Russian).

6. Lipatova S.E., Belov Yu.S. Praktiki obespecheniya kiberbezopasnosti v Kubernetes [Practical cybersecurity facility in Kubernetes]. *E-Scio*, 2022, no. 1 (64), pp. 489–497. (In Russian). Available at: https://cyberleninka.ru/article/n/praktiki-obespecheniya-kiberbezopasnosti-v-kubernetes (accessed 05.03.2025).

7. Kotenko I.V., Saenko I.B., Yusupov R.M. Novoe pokolenie sistem monitoringa i upravleniya intsidentami bezopasnosti [New generation of security information and event management systems]. *Informatika, telekommunikatsii i upravlenie = Computing, Telecommunication and Control*, 2014, no. 3 (198), pp. 7–18. Available at: https://cyberleninka.ru/article/n/novoe-pokolenie-sistem-monitoringa-i-upravleniya-intsidentami-bezopasnosti (accessed 05.03.2025).

8. Gatchin Yu.A., Teploukhova O.A. Realizatsiya kontrolya tselostnosti obraza operatsionnoi sistemy, zagruzhaemogo po seti na tonkii klient [Integrity monitoring implementation for the operating system image loaded through a network to the thin clients]. *Nauchno-tekhnicheskii vestnik informatsionnykh tekhnologii, mekhaniki i optiki = Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2015, no. 6, pp. 1115–1121. Available at: https://cyberleninka.ru/article/n/realizatsiya-kontrolya-tselostnosti-obraza-operatsionnoy-sistemy-zagruzhaemogo-po-seti-na-tonkiy-klient (accessed 05.03.2025).

9. Breaking out of Docker via runC – Explaining CVE-2019-5736. Available at: https://unit42.paloaltonetworks.com/breaking-docker-via-runc-explaining-cve-2019-5736/ (accessed 05.03.2025).

10. Shaturniy M.V. Analiz aktual'nykh ugroz i razrabotka podkhodov k zashchite veb-prilozhenii [Analysis of current threats and development of approaches to the protection of web applications]. *Inzhenernyi vestnik Dona = Engineering Journal of Don*, 2024, no. 7 (115), pp. 109–120. Available at: https://cyberleninka.ru/article/n/analiz-aktualnyh-ugroz-i-razrabotka-podhodov-k-zaschite-veb-prilozheniy (accessed 05.03.2025).

11. Ganzhur M.A., Dyachenko N.V. Monitoring tselostnosti failov [File integrity monitoring]. *Molodoi issledovatel' Dona = Young researcher of Don*, 2021, no. 2 (29), pp. 17–19. Available at: https://cyberleninka.ru/article/n/monitoring-tselostnosti-faylov (accessed 05.03.2025).

12. Holmes D., Papathanasaki M., Maglaras L., Ferrag M.A., Nepal S., Janicke H. Digital Twins and Cyber Security – solution or challenge? *2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*. IEEE, 2021, pp. 1–8. DOI: 10.1109/SEEDA-CECNSM53056.2021.9566277.

13. *Uyazvimost' CVE-2019-5736 v runc, pozvolyayushchaya poluchit' prava root na khoste* [The vulnerability CVE-2019-5736 in the runc that allows you to gain root rights on the host]. Available at: https://habr.com/ru/companies/flant/articles/439964/ (accessed 05.03.2025).

14. Gavrilov A. 5000 slov o zashchite konteinerov [5000 words about container protection]. *Jetinfo*, 2021, no. 7–8 (307–308), pp. 44–57. (In Russian). Available at: https://www.jetinfo.ru/5000-slov-o-zashhite-kontejnerov/ (accessed 05.03.2025).

15. National Vulnerability Database. CVE-2019-5736 Detail. Available at: https://nvd.nist.gov/vuln/detail/cve-2019-5736 (accessed 05.03.2025).

Для цитирования:

Мартенс П.И., Бережной А.С. Выход за пределы среды контейнеризации в современных IT-решениях // Безопасность цифровых технологий. – 2025. – № 1 (116). – С. 28–40. – DOI: 10.17212/2782-2230-2025-1-28-40.

For citation:

Martens P.I., Bereznoy A.S. Vykhod za predely sredy konteinerizatsii v sovremennykh IT-resheniyakh [Moving beyond containerization in today's it solutions]. *Bezopasnost' tsifrovykh tekhnologii = Digital Technology Security*, 2025, no. 1 (116), pp. 28–40. DOI: 10.17212/2782-2230-2025-1-28-40.