



# Контейнеры и Kubernetes: не боимся, а используем и защищаем

Дмитрий Евдокимов

[www.luntry.ru](http://www.luntry.ru)

# Команда Luntry



- Основатель и технический директор [Luntry](#)
- Опыт в ИБ более 10 лет
- Бывший автор статей и редактор рубрик в журнале “ХАКЕР”
- Автор Telegram-канала “[k8s \(in\)security](#)”
- Автор курса “Cloud Native безопасность в Kubernetes”
- Организатор конференции [БЕКОН](#) - первая в России конференция по БЕзопасности КОнтейнеров и контейнерных сред
- Докладчик: BlackHat, HITB, ZeroNights, HackInParis, Confidence, SAS, OFFZONE, PHDays, Kazhakistan, DevOpsConf, KuberConf, VK Kubernetes Conference, HighLoad++ и др.



## Контейнеры

Отличие от  
виртуальных машин

## Оркестрация контейнеров

Kubernetes

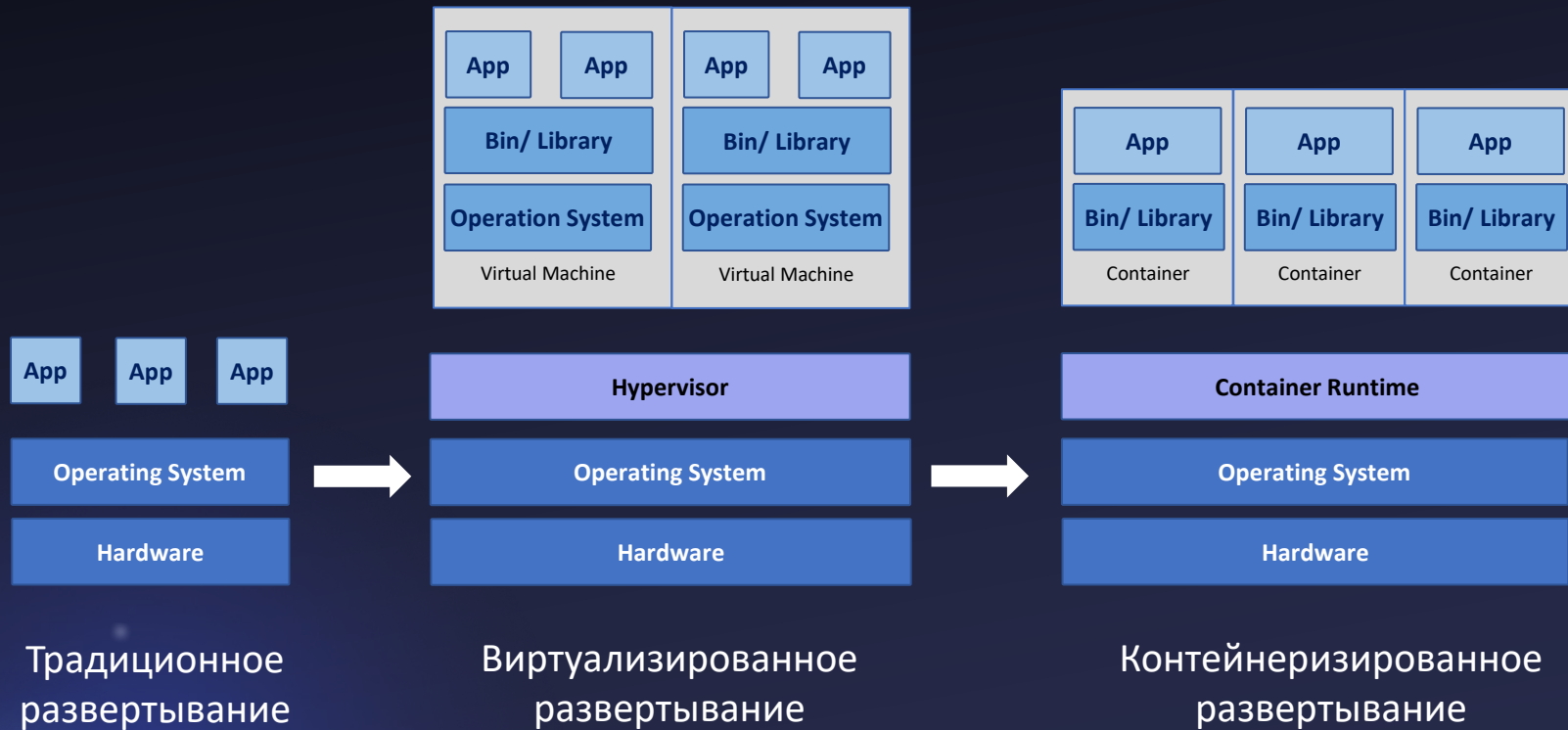
## Взгляд регуляторов на контейнерные среды

PCI DSS, 118 приказ,  
ГОСТ 57580.1-2017

# Контейнеры



# Эволюция к контейнерам



# Что такое контейнер?

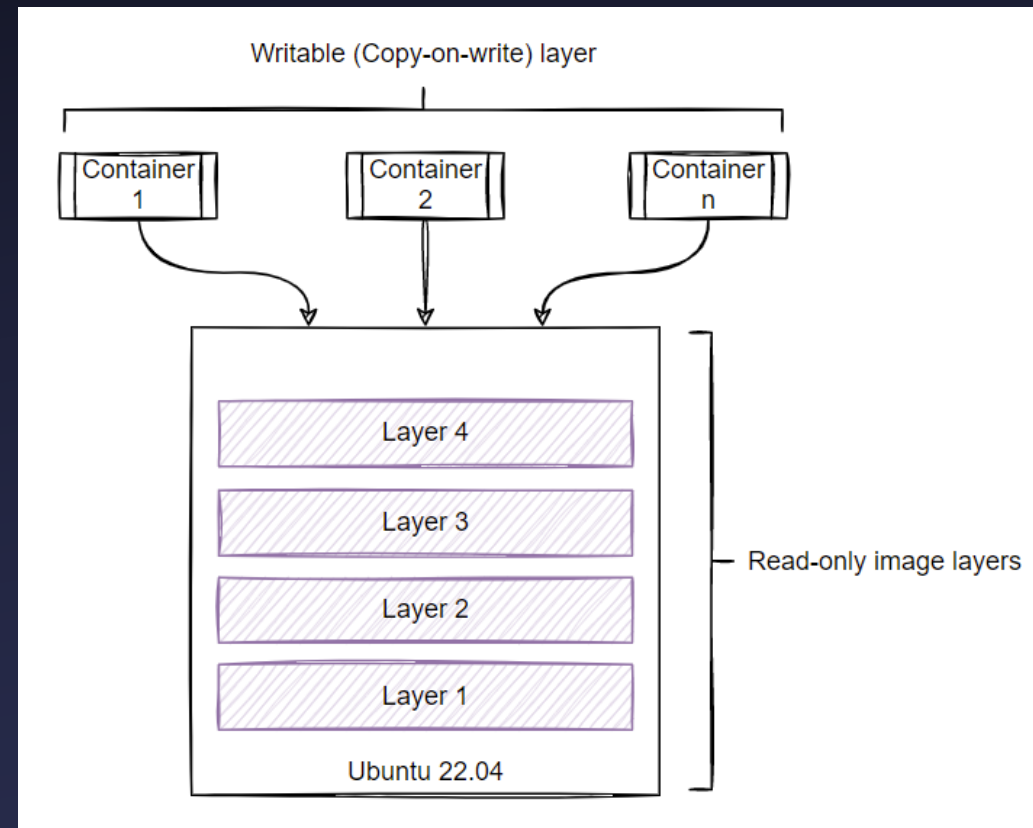
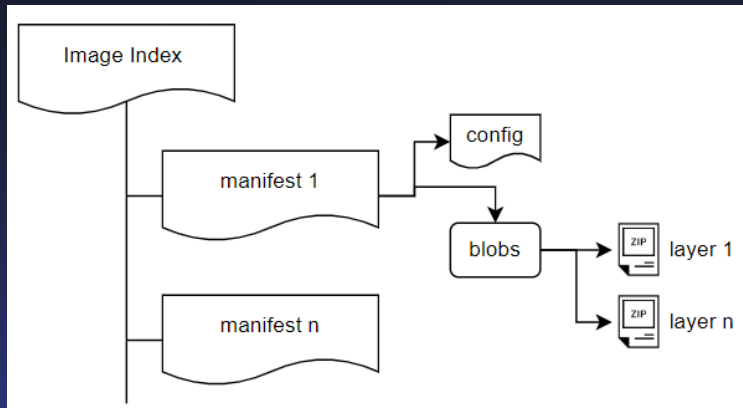
- Container это Linux process с определёнными свойствами/ограничениями
  - Что можно увидеть: namespaces (pid, user, uts, ipc, net, mnt), pivot\_root (+ image)
  - Что можно делать: Capabilities, seccomp, LSMs
  - Что можно использовать: Control group (процессор, память, устройства, ...)

```
root 2966156 0.0 0.0 110128 5932 ? SL Nov19 0:11 \_ containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1
root 2966174 0.0 0.0 1020 4 ? Ss Nov19 0:00 | \_ /pause
root 2966375 0.0 0.0 108720 6356 ? SL Nov19 0:11 \_ containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1
sadm 2966394 0.0 0.0 827512 19728 ? Ssl Nov19 0:00 | \_ node /usr/bin/nodemon /src/index.js
sadm 2966421 0.0 0.0 4460 80 ? S Nov19 0:00 | \_ sh -c node /src/index.js
sadm 2966422 0.0 0.0 967396 16596 ? SL Nov19 0:00 | \_ node /src/index.js
root 2988902 0.0 0.0 108720 5408 ? SL Nov19 0:11 \_ containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1
root 2988922 0.0 0.0 1020 4 ? Ss Nov19 0:00 | \_ /pause
root 2989066 0.0 0.0 108720 5408 ? SL Nov19 0:26 \_ containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1
root 2989099 0.0 0.0 31000 23956 ? Ss Nov19 0:42 | \_ /usr/local/bin/python /usr/local/bin/gunicorn -b :8080 --workers 1 --threads 1 --
root 2989116 0.3 0.1 142092 48964 ? SL Nov19 16:50 | \_ /usr/local/bin/python /usr/local/bin/gunicorn -b :8080 --workers 1 --threads
root 2989333 0.0 0.0 110128 5404 ? SL Nov19 0:11 \_ containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1
root 2989352 0.0 0.0 1020 4 ? Ss Nov19 0:00 | \_ /pause
root 596808 0.0 0.0 110128 6316 ? SL Nov20 0:06 \_ containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1
root 596827 0.0 0.0 1020 4 ? Ss Nov20 0:00 | \_ /pause
root 598309 0.0 0.0 110128 6224 ? SL Nov20 0:07 \_ containerd-shim -namespace moby -workdir /var/lib/containerd/io.containerd.runtime.v1
root 598334 1.4 5.5 7236340 1832196 pts/0 Ssl+ Nov20 39:39 \_ /docker-java-home/bin/java -Djava.util.logging.config.file=/opt/atlassian/conflue
root 599854 1.0 1.3 7007820 427956 pts/0 Sl+ Nov20 28:11 | \_ /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java -classpath /opt/atlassian/conf
root 701694 0.0 0.0 4288 764 ? Ss+ Nov20 0:00 \_ /bin/sh
```



# Что такое образ контейнер?

- Container Image это неизменяемый пакет файлов операционной системы, кода приложения и любых зависимостей приложения
  - Union File System
    - OverlayFS как реализация
  - OCI image спецификация



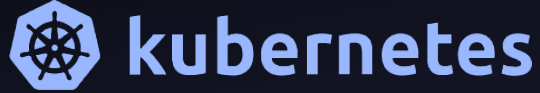
# Оркестрация контейнеров



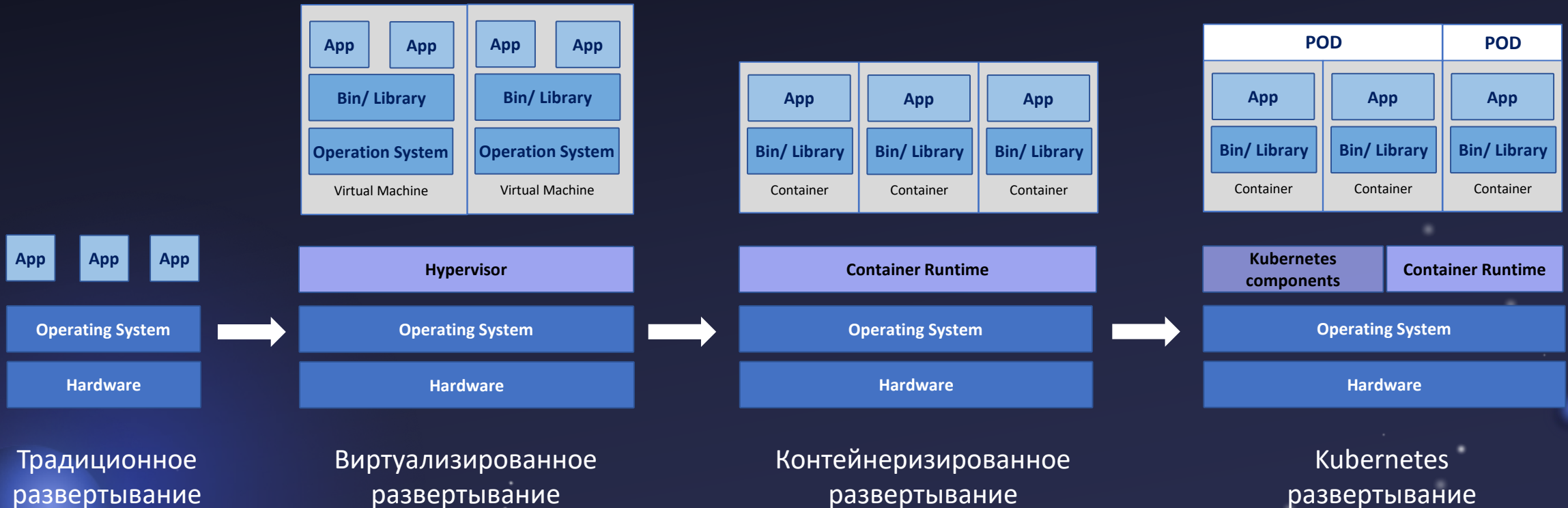


# Эволюция от контейнеров к оркестраторам контейнеров

Оркестраторы:

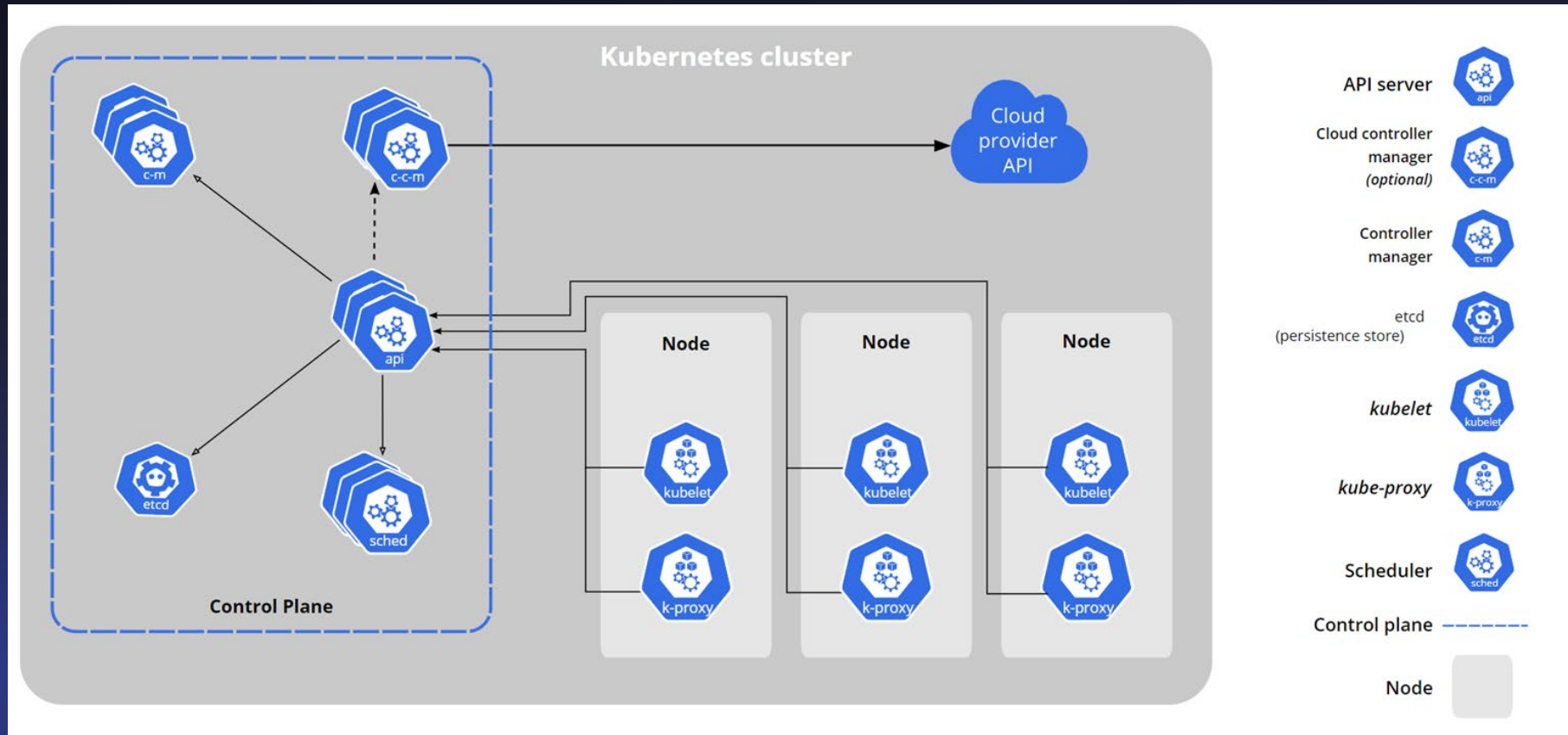


HashiCorp  
Nomad



# Что такое Kubernetes?

- Kubernetes (K8s) — это открытое программное обеспечение для автоматизации развёртывания, масштабирования и управления контейнеризированными приложениями.
- 5 бинарей



# Дистрибутивы Kubernetes

- Это фреймворк
  - OnPrem и Managed Kubernetes
  - Это ядро Linux 21 века
  - На базе Kubernetes делают свои дистрибутивы



# Взгляд регуляторов на контейнерные среды



# ГОСТ Р 57580.1-2017 от ЦБ РФ

- Примеры выполнения требований ГОСТ Р 57580.1-2017 в средах контейнерной оркестрации на базе Kubernetes

| № | ПРОЦЕСС   | ОБЪЕКТ ПРИМЕНЕНИЯ МЕР   |  |  |
|---|---|---|--|--|
|   |   | УРОВЕНЬ NODE  | УРОВЕНЬ ORCHESTRATOR   | УРОВЕНЬ POD/CONTAINER  |
| 1 | Обеспечение защиты информации при управлении доступом   | Запрет подключения к Worker Node, контроль кластером через Master Node  | Интеграция с Identity Provider (например, LDAP) для получения доступа к кластеру   | Использование RBAC для реализации принципа least privilege   |
| 2 | Обеспечение защиты вычислительных сетей   | <ul style="list-style-type: none"> <li>• Помещение кластера в защищенный сегмент</li> <li>• Реализация TLS при общении между nodes</li> </ul>           | <ul style="list-style-type: none"> <li>• Контроль запуска pods/containers только на определенных nodes</li> <li>• Использование наложенных СЗИ для контроля трафика (L3/L4 firewalling)</li> </ul> | <ul style="list-style-type: none"> <li>• Использование Network Policy</li> <li>• Использование mTLS</li> </ul>   |
| 3 | Контроль целостности и защищенности информационной инфраструктуры                             | Принятый в организации подход к реализации процесса управления уязвимостями   | Обновление версии оркестратора   | <ul style="list-style-type: none"> <li>• Подпись образов, верификация подписи перед запуском контейнера при помощи внешних решений</li> <li>• Использование внешних решения для идентификации уязвимостей в образе</li> </ul>  |
| 4 | Защита от вредоносного кода   | Использование SELinux/AppArmor (компенсирующие меры)  | Неприменимо  | Компенсирующие меры защиты: <ul style="list-style-type: none"> <li>• Использование SecurityContext</li> <li>• Использование внешних решений для защиты контейнеров в runtime</li> </ul>  |
| 5 | Предотвращение утечек информации  | Неприменимо (используется принятый в организации процесс предотвращения утечек информации)  |  |  |
| 6 | Управление инцидентами защиты информации  | Использование принятого в организации подхода к мониторингу элементов ИТ-инфраструктуры   | Разработка Audit Policy с последующим направлением журналов в SIEM для идентификации инцидентов ИБ   | <ul style="list-style-type: none"> <li>• Разработка Audit Policy с последующим направлением журналов в SIEM для идентификации инцидентов ИБ</li> <li>• Использование внешних решений, которые позволяют идентифицировать инциденты ИБ</li> <li>• Отправка журналов внешних средств защиты в SIEM (нарушение политик ИБ)</li> </ul> |
| 7 | Защита среды виртуализации  | Реализация практик, указанных в процессах [1], [2], [3], [4], [6] (согласно нумерации, используемой в таблице)  | Реализация практик, указанных в процессах [1], [2], [3], [4], [6]  | Реализация практик, указанных в процессах [1], [2], [3], [4], [6]  |
| 8 | Защита при осуществлении удаленного доступа с использованием мобильных (переносных) устройств | Неприменимо (используется принятый в организации процесс защиты при осуществлении удаленного доступа с использованием мобильных (переносных) устройств) |  |  |

|   |   |   |  |  |
|---|---|---|--|--|
| 8 | (вредоносных) устройств с использованием мобильных устройств удаленного доступа | Неприменимо (используется принятый в организации процесс защиты при осуществлении удаленного доступа с использованием мобильных (переносных) устройств) |  |  |
|---|---|---|--|--|



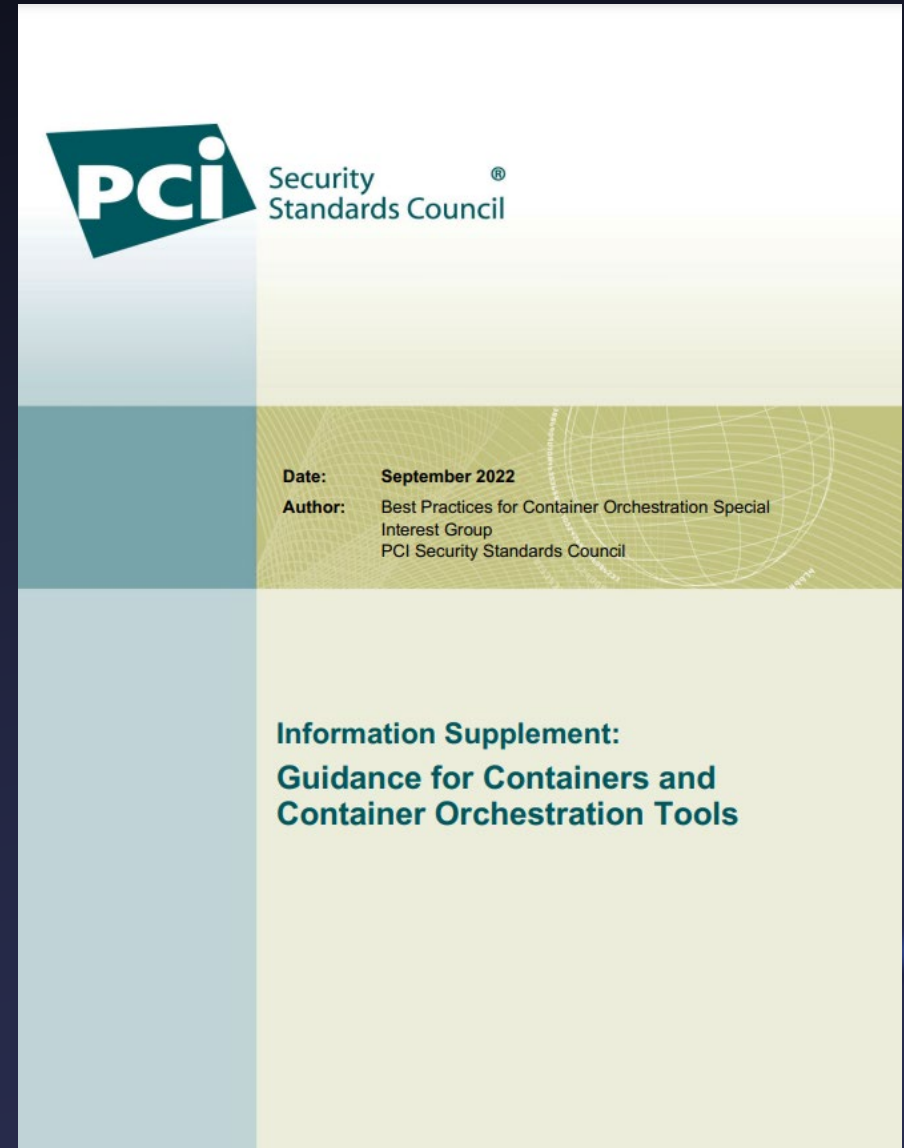
- НСПК [требует](#) соответствия PCI DSS
- Существует [руководство](#) по безопасности для контейнеров и оркестраторов контейнеров

## Стандарт PCI DSS

В ПС «Мир» для обеспечения безопасности данных карт «Мир» используется международный индустриальный стандарт PCI Data Security Standard (PCI DSS)


Этот стандарт должен применяться всеми организациями, которые хранят, обрабатывают и передают данные карт «Мир». К таким организациям относятся и торгово-сервисные предприятия, которые принимают к оплате карты «Мир».

Стандарт PCI DSS — это международный стандарт безопасности, созданный специально для защиты данных платежных карт. Он позволяет защитить организацию от инцидентов безопасности и обеспечить необходимый уровень защищенности во всей платежной системе.





## Требования по безопасности информации к средствам контейнеризации



**Требования по безопасности информации к средствам контейнеризации**

Проект

### Функциональные возможности:

формирование среды выполнения контейнеров и обеспечения выполнения их процессов

запуск контейнера и управление данным контейнером

создание образов контейнеров

распространение образов контейнеров

централизованное управление контейнерами и организацией взаимодействия между ними

### Функции безопасности:

управление доступом

идентификация и аутентификация пользователей

изоляция контейнеров

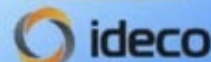
выявление уязвимостей в образах контейнеров

проверка корректности конфигурации контейнеров

контроль целостности контейнеров и их образов

централизованное управление образами контейнеров и контейнерами

регистрация событий безопасности



- Выписка из Требований по безопасности информации, утвержденных приказом ФСТЭК России от 4 июля 2022 г. N 118

- Сертифицированная ОС
- Сертифицированный оркестратор
- Сертифицированное наложенное средство безопасности

УТВЕРЖДЕНЫ  
приказом ФСТЭК России  
от 4 июля 2022 г. № 118

## **Требования по безопасности информации к средствам контейнеризации (выписка)**

1. Настоящие Требования являются обязательными требованиями в области технического регулирования к продукции, используемой в целях защиты сведений, составляющих государственную тайну или относимых к охраняемой в соответствии с законодательством Российской Федерации иной информации ограниченного доступа<sup>1</sup> (далее – требования по безопасности информации), предъявляемыми к программным средствам, обеспечивающим создание и функционирование изолированных программных сред на основе ядра хостовой операционной системы (далее – контейнеры) в информационной (автоматизированной) системе (далее – средства контейнеризации).

2. Выполнение настоящих Требований является обязательным при проведении работ по оценке соответствия (включая работы по сертификации) средств технической защиты информации и средств обеспечения безопасности информационных технологий, организуемых ФСТЭК России в пределах своих

# Заключение



- Если у вас еще нет контейнеров, то они обязательно появятся
- Так или иначе вы придёте к использованию оркестратора
- Kubernetes стандарт де-факто среди оркестраторов
- Обеспечение безопасности контейнеров и оркестратора новый вызов для ИБ
- Сегодня регуляторы уже подготовились к этому вызову

# СПАСИБО ЗА ВНИМАНИЕ!



## CONTACTS:

- Email: [de@luntry.ru](mailto:de@luntry.ru)
- Twitter: [@evdokimovds](https://twitter.com/evdokimovds)
- Tg: [@Qu3b3c](https://t.me/Qu3b3c)
- Channel: [@k8security](https://t.me/k8security)
- Site: [www.luntry.ru](http://www.luntry.ru)

