

Обеспечение безопасности контейнеров: **Обеспечение безопасности** **цепочки поставки**

В рамках данного выступления рассматривается использование сканеров уязвимостей, сравнение и примеры работы с ними в различных конфигурациях. В частности, в фокусе выступления – обеспечение безопасности контейнеров в рамках разработки, тестирования и внедрения программных решений, направленных на развитие обеспечения безопасности конвейера DevSecOps.

Содержание

1. Обеспечение безопасности цепочки поставки

1. Базовые образы
2. Библиотеки
3. Принципы обеспечения безопасности конечных образов
4. SAST (Статический анализ кода)
5. DAST (Динамический анализ кода)
6. Применение техник минимизации attack surface

1. Обеспечение безопасности контейнеров (Container Security)

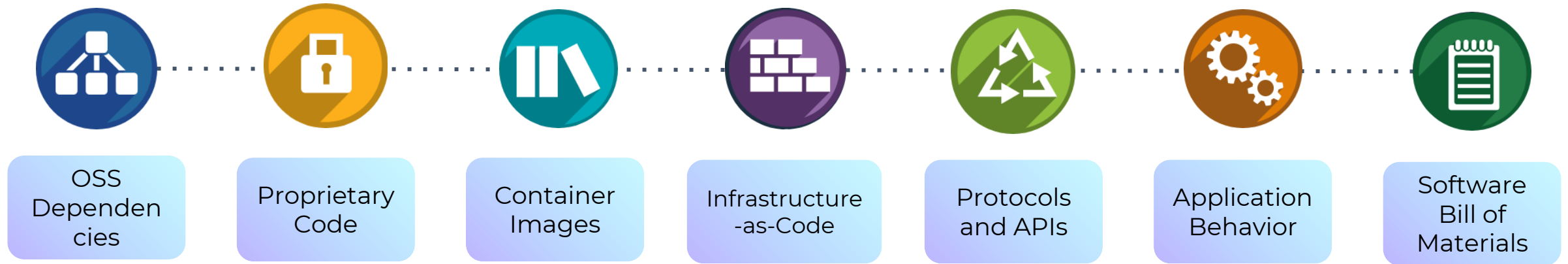
1. Принципы безопасной разработки. Практики DevSecOps
2. Использование сканеров уязвимостей, сравнение и примеры работы с ними в различных конфигурациях
3. Обеспечение безопасности контейнеров в рамках разработки, тестирования и внедрения программных решений. CVE \ CVSS.

Цепочка поставки

Цепочка поставок программного обеспечения — это все, что касается приложения или играет какую-либо роль в его разработке на протяжении всего жизненного цикла разработки программного обеспечения

Безопасность цепочки поставок программного обеспечения — это действия по обеспечению безопасности компонентов и методов, связанных с созданием и развертыванием программного обеспечения. Это включает сторонний и проприетарный код, методы и инфраструктуру развертывания, интерфейсы и протоколы, а также методы разработки и инструменты разработки. Организации несут ответственность за выполнение этих действий по обеспечению безопасности и за предоставление потребителям доказательств своих усилий по обеспечению безопасности.

Обеспечение безопасности цепочки поставки



CVE

CVE (Common Vulnerabilities and Exposures) – это база данных известных уязвимостей и дефектов безопасности:

- Система активно поддерживается центром исследований и разработок США (Federally Funded Research and Development Centers, FFRDC), которым управляет корпорация MITRE.
- Поскольку MITRE является некоммерческой организацией, CVE финансируется отделом национальной кибербезопасности США (National Cyber Security Division, NCSD).

CVE

CVE (Common Vulnerabilities and Exposures) – это база данных известных уязвимостей и дефектов безопасности:

- Система активно поддерживается центром исследований и разработок США (Federally Funded Research and Development Centers, FFRDC), которым управляет корпорация MITRE.
- Поскольку MITRE является некоммерческой организацией, CVE финансируется отделом национальной кибербезопасности США (National Cyber Security Division, NCSD).

История системы CVE

- Первоначальная концепция базы данных CVE возникла в техническом документе 1999 года под названием «[На пути к общему перечню уязвимостей](#)» (Towards a Common Enumeration of Vulnerabilities), написанном Стивеном М. Кристи и Дэвидом Э. Манном из корпорации MITRE.
- Кристи и Манн собрали рабочую группу из 19 специалистов и составили первоначальный список CVE из 321 записи. [В сентябре 1999](#) года реестр стал общедоступным. С момента запуска CVE в 1999 году различные ИБ-компании дополняли список уязвимостей. К декабрю 2000 года в инициативе участвовало 29 организаций со своими 43 ошибками.

История системы CVE

- CVE использовалась в качестве отправной точки для Национальной базы данных уязвимостей США ([National Vulnerability Database, NVD](#)) института [NIST](#).
- CVE расширяется с каждой организацией, которая присоединяется к MITRE в качестве соавтора. Полный список партнеров можно найти [на CVE.org](#).

Vulnerabilities vs Exposures

Vulnerabilities – это недостатки системы, слабые места в инфраструктуре, которые могут быть использованы киберпреступником: от неисправленного ПО до незащищенного USB-порта. Уязвимости системы могут позволить злоумышленнику:

- получить доступ к системной памяти;
- установить вредоносное ПО;
- запустить вредоносный код;
- украсть, уничтожить или изменить конфиденциальные данные.

Vulnerabilities vs Exposures

Exposures – это единичные случаи, когда система организации находится под угрозой. Простая ошибка позволяет провести кибератаку на организацию.

- Сюда можно отнести кражу конфиденциальных данных, которые затем продаются в даркнете.
- Большинство киберинцидентов вызвано раскрытием информации, а не хорошо продуманными эксплойтами.

Как определяются CVE?

Все CVE — это недостатки, но не все недостатки — CVE.

Недостаток объявляется CVE, когда он соответствует трем конкретным критериям:

- Недостаток может быть исправлен отдельно от любых других ошибок;
- Поставщик ПО признал и задокументировал уязвимость как наносящую ущерб безопасности пользователей;
- Ошибка затрагивает единственную кодовую базу.
Недостаткам, затрагивающим несколько продуктов, присваивается несколько CVE.

Как определяются CVE?

Каждой уязвимости CVE присваивается номер (CVE Identifier или CVE ID) одним из 222 центров нумерации CVE ([CVE Numbering Authorities, CNA](#)) из 34 стран.

Согласно MITRE, CNA представляют различные организации: от поставщиков ПО и open-source-проектов до поставщиков услуг по поиску ошибок и исследовательских групп.

Все эти организации имеют право назначать идентификаторы CVE и публиковать записи о них в рамках программы CVE.

На протяжении многих лет к программе CNA присоединялись предприятия из разных отраслей. Требования для вступления минимальны и не требуют контракта или денежного вноса.

CVE ID

Международный стандарт для идентификаторов CVE — это CVE-xxxx-ууууу.

- [xxxx] — год, когда уязвимость была обнаружена.
- [ууууу] — это серийный номер, присвоенный соответствующим CNA.

Почему важна программа CVE?

- База CVE была создана для упрощения обмена информацией об известных уязвимостях между организациями.
- Идентификаторы CVE дают специалисту по кибербезопасности возможность легко находить информацию о недостатках в различных авторитетных источниках, используя один и тот же идентификатор уязвимости.

Почему важна программа CVE?

- Более того, CVE является надежной базой для компании, чтобы понять необходимость инвестиций в улучшение защиты.
- Организация может быстро получить точную информацию о конкретном эксплойте из нескольких сертифицированных источников, что позволяет правильно расставить приоритеты для устранения проблемы.

Сколько существует CVE?

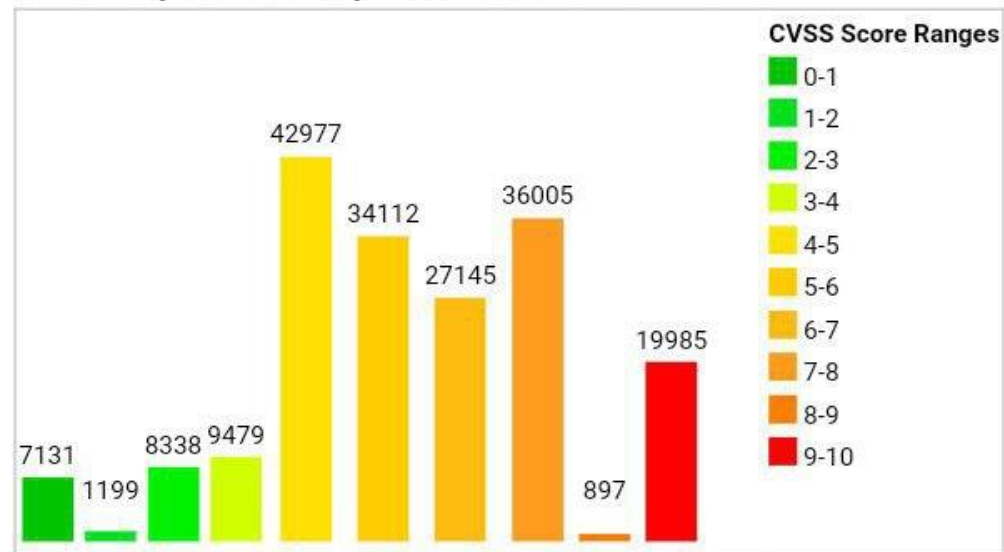
Current CVSS Score Distribution For All Vulnerabilities

Distribution of all vulnerabilities by CVSS Scores

CVSS Score	Number Of Vulnerabilities	Percentage
0-1	7131	3.80
1-2	1199	0.60
2-3	8338	4.50
3-4	9479	5.10
4-5	42977	22.90
5-6	34112	18.20
6-7	27145	14.50
7-8	36005	19.20
8-9	897	0.50
9-10	19985	10.70
Total	187268	

Weighted Average CVSS Score: **6.3**

Vulnerability Distribution By CVSS Scores



CVSS и CVE

CVSS v2.0 Ratings

Low	0.0-3.9
Medium	4.0-6.9
High	7.0-10.0

CVSS v3.0 Ratings

Low	0.1-3.9
Medium	4.0-6.9
High	7.0-8.9
Critical	9.0-10.0

CVSS

CVSS(Common Vulnerability Scoring System) — открытый стандарт для оценки степени опасности уязвимостей.

- CVSS разработал Национальный совет по инфраструктуре (National Infrastructure Advisory Council, NIAC) США.
- Также в создании и обновлении стандарта участвовали коммерческие компании, такие как Microsoft, Cisco и другие.
- Поддержкой системы занимается Форум групп безопасности и реагирования на инциденты (Forum of Incident Response and Security Teams, FIRST).

Методика оценки уязвимостей по CVSS (CVSS Score)

Согласно стандарту CVSS, уязвимости оцениваются на основании ряда метрик. Можно выделить три типа метрик:

- **Базовые метрики.**
- **Временные метрики.**
- **Метрики окружения.**

Методика оценки уязвимостей по CVSS (CVSS Score)

На основании метрик с помощью набора формул вычисляется оценка CVSS Score. Она может принимать значение от 0 до 10, где:

- 9,0–10,0 — критический уровень опасности;
- 7,0–8,9 — высокий;
- 4,0–6,9 — средний;
- 0,1–3,9 — низкий;
- 0 — опасность отсутствует.

Методика оценки уязвимостей по CVSS (CVSS Score)

Базовые метрики. Сюда относятся общие метрики, описывающие уязвимость и не зависящие от времени или конкретного окружения

Методика оценки уязвимостей по CVSS (CVSS Score)

- **Базовые метрики** делятся на две группы:
 - **Метрики эксплуатации**, описывающие, насколько уязвимость проста в эксплуатации. Например, вектор атаки: одни уязвимости можно эксплуатировать через Интернет, то есть из любой точки мира с доступом к Сети, а другие требуют физического доступа к уязвимому устройству.
 - **Метрики воздействия**, касающиеся последствий эксплуатации уязвимости для системы и хранящихся в ней данных. Например, может ли атакующий вывести систему из строя, получить доступ к конфиденциальным данным, модифицировать файлы и т. д.

Методика оценки уязвимостей по CVSS (CVSS Score)

Временные метрики описывают внешние факторы, которые могут измениться с течением времени. Например, наличие доступного эксплойта или, наоборот, патча.

Методика оценки уязвимостей по CVSS (CVSS Score)

Метрики окружения на основную оценку уязвимости никак не влияют, но позволяют определить ее опасность для конкретной IT-среды. В набор метрик окружения входят базовые метрики с поправкой на условия конкретной среды.

Методика оценки уязвимостей по CVSS (CVSS Score)

Метрики окружения

- Так, если для эксплуатации уязвимости в целом требуются минимальные привилегии, то в конкретной организации доступ к уязвимой системе могут иметь только администраторы.
- Также к метрикам окружения относятся метрики, описывающие то, насколько опасны для конкретной организации возможные последствия эксплуатации уязвимости. Например, повлияет ли на операции компании отключение сервера или у нее есть запасной сервер, на который легко переключиться в случае инцидента.

Методика оценки уязвимостей по CVSS (CVSS Score)

Для упрощения расчета CVSS Score существуют онлайн-калькуляторы CVSS.

В настоящее время для оценки уязвимостей используется версия CVSS 3.1, вышедшая в июне 2019 года:

<https://www.first.org/cvss/calculator/3.1>

CWE

The Common Weakness Enumeration (CWE) является системой классификации ошибок. Проект спонсируется MITRE и поддерживается Компьютерной командой экстренной готовности США (United States Computer Emergency Readiness Team) и Национальным отделом кибербезопасности Министерства внутренней безопасности США (National Cyber Security Division of the US Department of Homeland Security).

Под ошибками понимаются сбои и ошибки при реализации программного или аппаратного обеспечения, в проектировании, архитектуре и т.д., которые могут сделать конечный продукт уязвимым к различного рода атакам.

CWE

Основная цель CWE — предотвращать возникновение уязвимостей за счёт обучения специалистов способам избегания наиболее распространённых ошибок.

То есть в конечном итоге CWE позволяет избегать уязвимости, которым подвержено программное и аппаратное обеспечение.

CWE может помочь:

- описывать и обсуждать программные и аппаратные недостатки безопасности на общем языке;
- проверять на недостатки безопасности существующие программные и аппаратные решения;
- оценивать возможности специализированных инструментов по поиску недостатков безопасности;
- предотвращать потенциальные уязвимости в программном и аппаратном обеспечении до его доставки пользователям.

OWASP TOP 10 – Web Application Security Risks

Что такое OWASP?

- . OWASP (расшифровывается как Open Web Application Security Project) — это онлайн-сообщество, которое выпускает статьи на тему безопасности веб-приложений, а также документацию, различные инструменты и технологии.

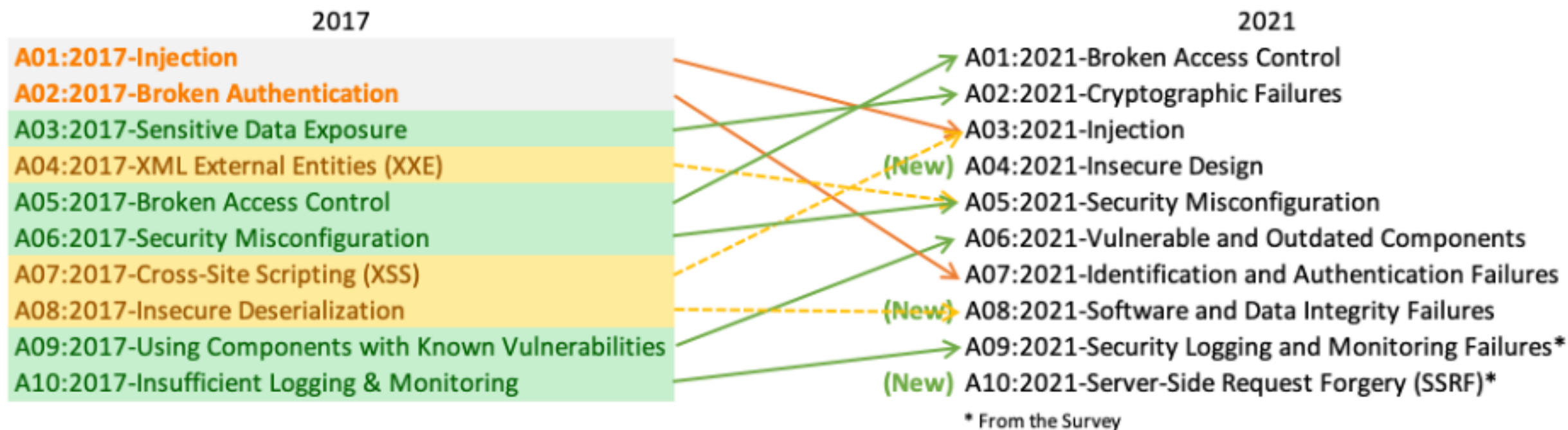
OWASP TOP 10 – Web Application Security Risks

Что такое OWASP Топ-10?

OWASP Топ-10 — это список из десяти самых распространённых на данный момент уязвимостей веб-приложений. Благодаря этому списку пользователи будут осведомлены о наиболее критичных рисках и угрозах, их последствиях и мерах противодействия. Обновляется список OWASP каждые три-четыре года :

<https://owasp.org/www-project-top-ten/>

OWASP TOP 10 – Web Application Security Risks



Статические анализаторы приложений (SAST)

Статический анализатор кода - инструмент, сообщаящий об уязвимости приложения, ориентируясь на исходные коды приложения.

Бесплатные / Open-source

- Semgrep
- ShiftLeft Scan
- Salus
- HuskyCI
- CodeQL

Коммерческие / Enterprise

- Solar AppScreener
- FortifySCA
- PT AI
- Checkmarx
- PVS-Studio
- RIPS
- Veracode Static Analysis

Динамические анализаторы приложений (DAST)

Динамический анализатор кода - инструмент, сообщаящий об уязвимости приложения, ориентируясь на ответы сервера по заданным запросам.

Бесплатные / Open-source

- OWASP ZAP
- w3af
- nikto
- nerve
- Arachni
- Nuclei

Коммерческие / Enterprise

- PortSwigger Burp Suite
- NetSparker
- Acunetix
- WebInspect
- PT AI
- Veracode Dynamic Analysis
- Tenable Web App Scanning

Сканеры Docker образов

Инструменты, направленные на поиск уязвимостей в образах контейнеров.

Бесплатные / Open-source

- Clair
- Trivy
- Anchore
- AquaMicroscanner
- Dagda
- whalescan
- gype
- syft

Коммерческие / Enterprise

- Snyk Container
- TrendMicro SmartCheck
- WhiteSource for containers
- Nexus Container

Принципы обеспечения безопасности конечных образов.

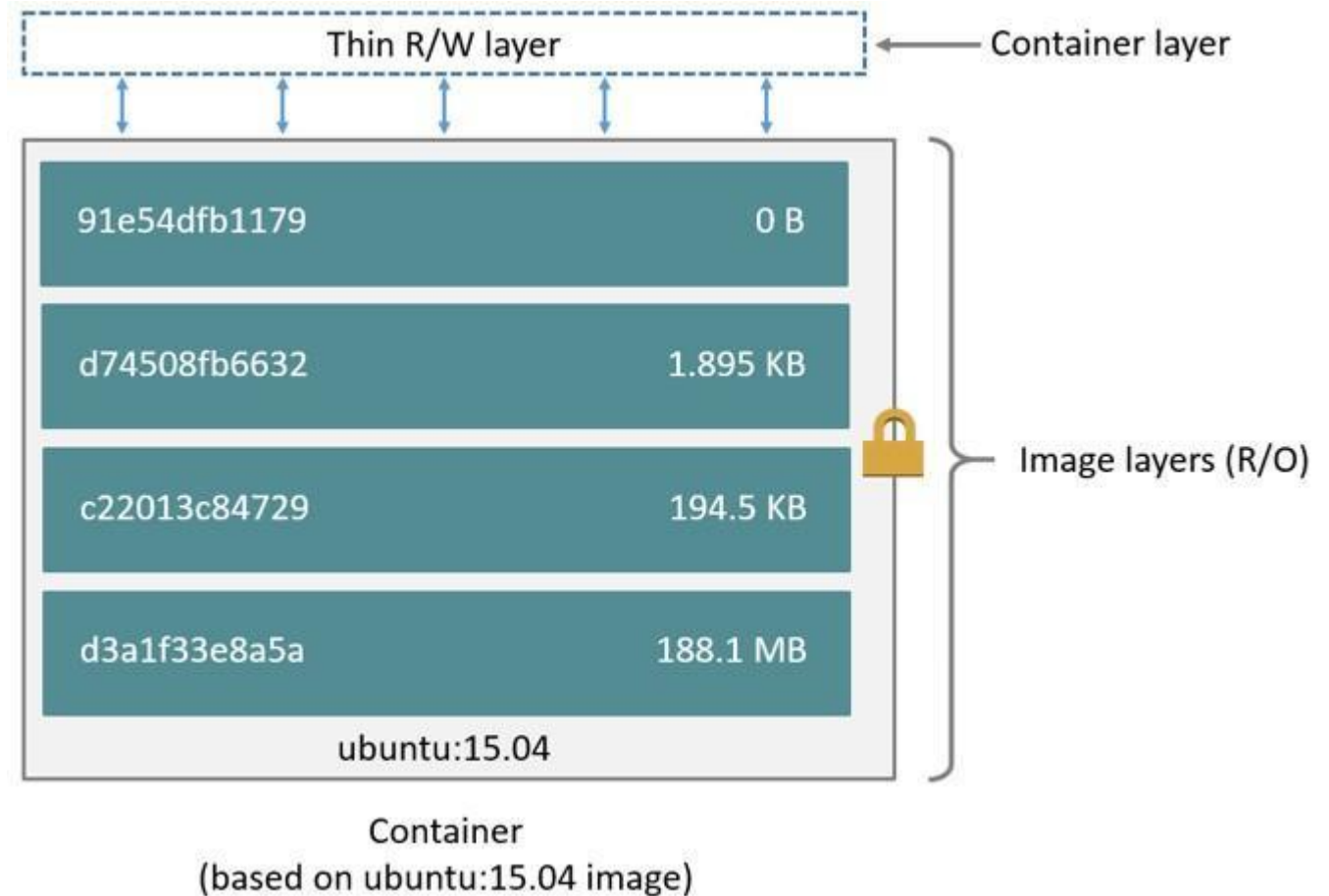
Docker Security Cheat Sheet

https://cheatsheetseries.owasp.org/cheatsheets/Docker_Security_Cheat_Sheet.html

БАЗОВЫЕ ОБРАЗЫ

- Базовый образ — это то, что является исходным слоем (или слоями) создаваемого образа. Базовый образ ещё называют родительским образом.

<https://catalog.redhat.com/software/containers/search>



Обеспечение безопасности контейнеров

Container Security Checklist: From the image to the workload

<https://github.com/krol3/container-security-checklist>



СПАСИБО ЗА ВНИМАНИЕ!