LUNTRY

БЕЗОПАСНОСТЬ КОНТЕЙНЕРОВ И KUBERNETES ДЛЯ DevSecOps СПЕЦИАЛИСТОВ

Дмитрий Евдокимов

Founder&CTO Luntry

В предыдущих ~~сериях~~ вебинарах*

01

Для CISO

- Те кто формируют направление развития

02

Для специалистов SOC

- Те кто смотрят за событиями в системе

03

Для специалистов анализа качества

- Те кто проводят ПСИ, осуществляют сертификацию ПО, разбираются с “черными ящиками”

*- слайды и видео доступно на https://luntry.ru/research

Обо мне

”

Я не верю в то, что систему можно сделать надежной и безопасной, не понимая того, как она устроена.

Основатель и технический директор Luntry

Более 15 лет опыта в ИБ

Специализация – безопасность контейнеров и Kubernetes

Автор ТГ-канала k8s(in)security

Эксперт в сфере безопасности контейнерных сред

- Организатор конференции «БеКон» по БЕзопасности КОНтейнеров
- Бывший редактор рубрик в журнале «ХАКЕР», автор серии статей
- Автор курса «Cloud Native безопасность в Kubernetes»
- Член программного комитета CFP DevOpsConf и HighLoad++

Спикер

VK Kubernetes	Confidence	ZeroNights	БеКон	HITB
DevOpsConf	HackInParis	KuberConf	BlackHat	PHDays
Kazhackstan	HighLoad++	OFFZONE	DevOops	SAS

LUNTRY

О компании Luntry

Luntry – это Комплексная Защита
на всем жизненном цикле
контейнерных приложений
и средств оркестрации на базе Kubernetes

Продукт в реестре Минцифры
https://reestr.digital.gov.ru/reestr/1057835/

Получение сертификата ФСТЭК планируется в четвертом квартале 2025 года.

LUNTRY.RU

Функциональность Luntry

Контроль всех Kubernetes-ресурсов

Контроль состояния Kubernetes-кластеров

Контроль соответствия кластера стандартам

Управление уязвимостями образов и best practice

Сетевая безопасность

Анализ прав доступа

Защита Runtime

Функциональность Luntry

Контроль всех Kubernetes-ресурсов

Контроль состояния Kubernetes-кластеров

Контроль соответствия кластера стандартам

Управление уязвимостями образов и best practice

Сетевая безопасность

Анализ прав доступа

Защита Runtime

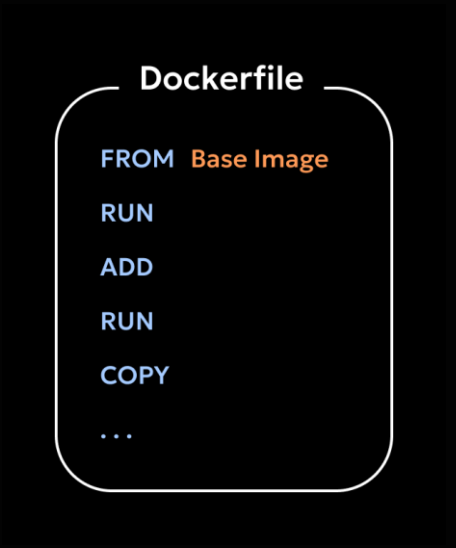
План вебинара

LUNTRY

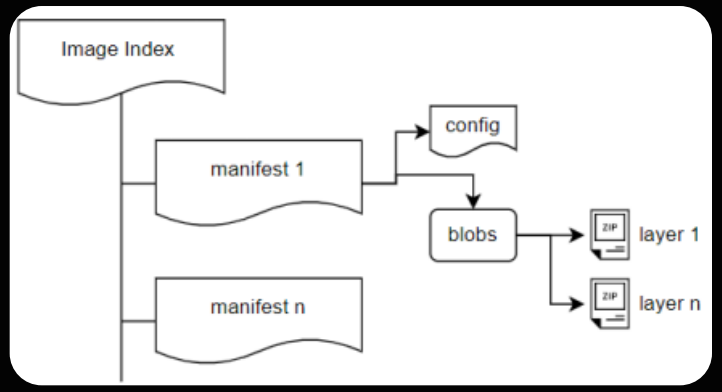
01	Теория про образы контейнеров
02	Жизненный путь образа
03	Анализ образа
04	Как Luntry помогает решить задачи, связанные с безопасностью образов
05	Выводы

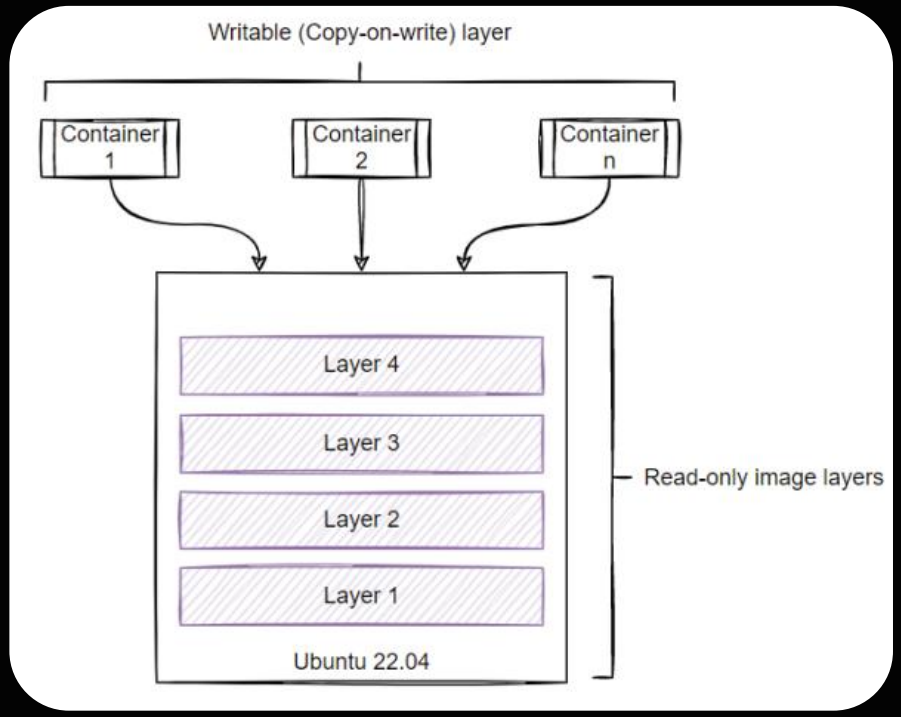
ТЕОРИЯ ПРО ОБРАЗЫ КОНТЕЙНЕРОВ

Структура образа



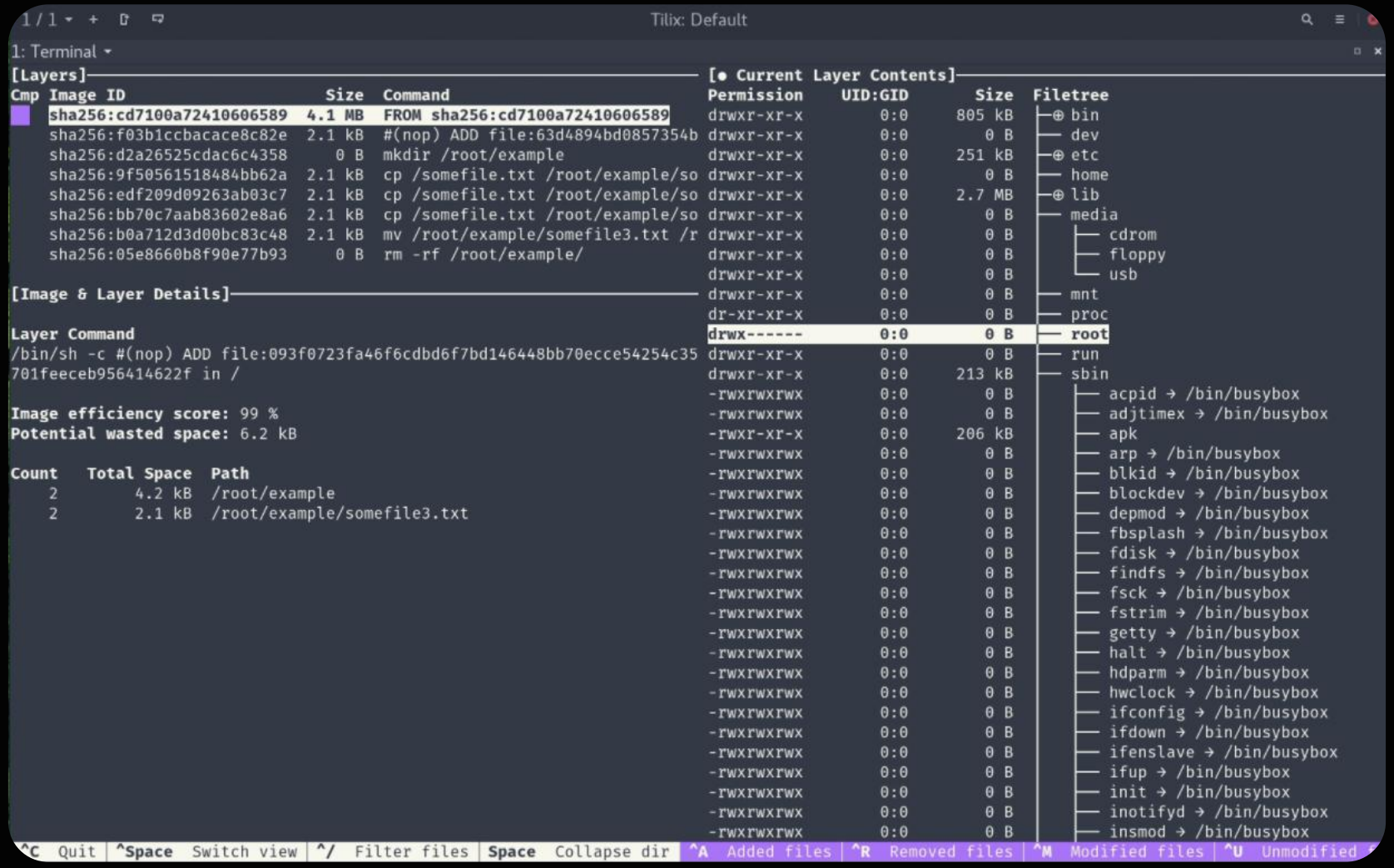
- Образ собирается из слоев описанных в Dockerfile
- Container Image это неизменяемый пакет файлов операционной системы, кода приложения и любых зависимостей приложения
  - Union File System
    - OverlayFS как реализация
  - OCI спецификация
    - OCI Image Format, OCI Distribution Specification, OCI Runtime Specification





Полезные инструменты

- Dive
- Skiff
- skopeo



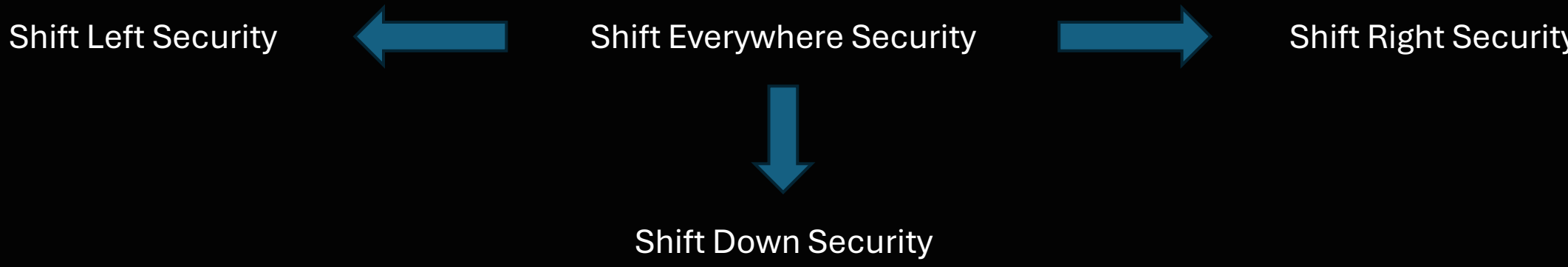
LUNTRY

ЖИЗНЕННЫЙ ПУТЬ ОБРАЗА

Жизненный путь образа

LUNTRY

- CI\CD Pipeline
  - Сборка образа
  - Анализ образа
  - Security/Quality gates
    - Возможна блокировка
  - Подпись образа
- Image registry
  - Хранение
  - Очистка/ротация
  - Анализ образов
  - Security/Quality gates
- Runtime
  - Запуск
  - Анализ образов
  - Security/Quality gates



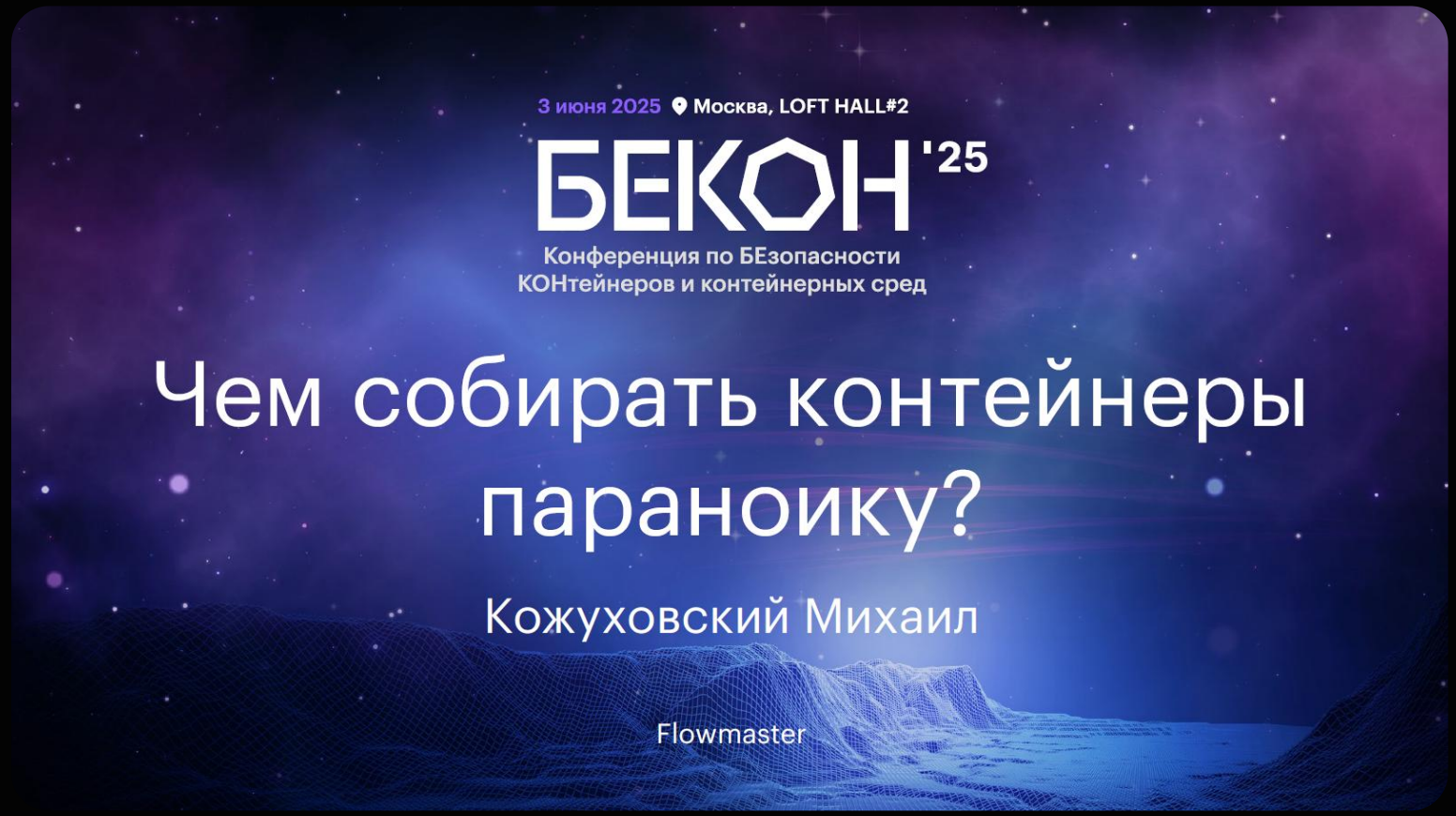
Сборка образа

- Сборка в Docker-in-Docker (DinD)
  - Плохо! Не делайте так!
- Используйте безопасные инструменты сборки
  - Не использующие Docker demon
- Используйте минималистичные базовые образы
  - Tiny/slim/minimal/distroless образы
- Используйте практику Golden image
  - Может сочетаться или дополнять концепцию минималистичных образов
- Используйте multi-stage сборку
  - Совместимо со всем выше сказанным

LUNTRY

Полезные инструменты

- Kaniko
- Buildah
- BuildKit



Доклад «Чем собирать контейнеры если вы параноик?», БеКон 2025

Домены безопасности образов

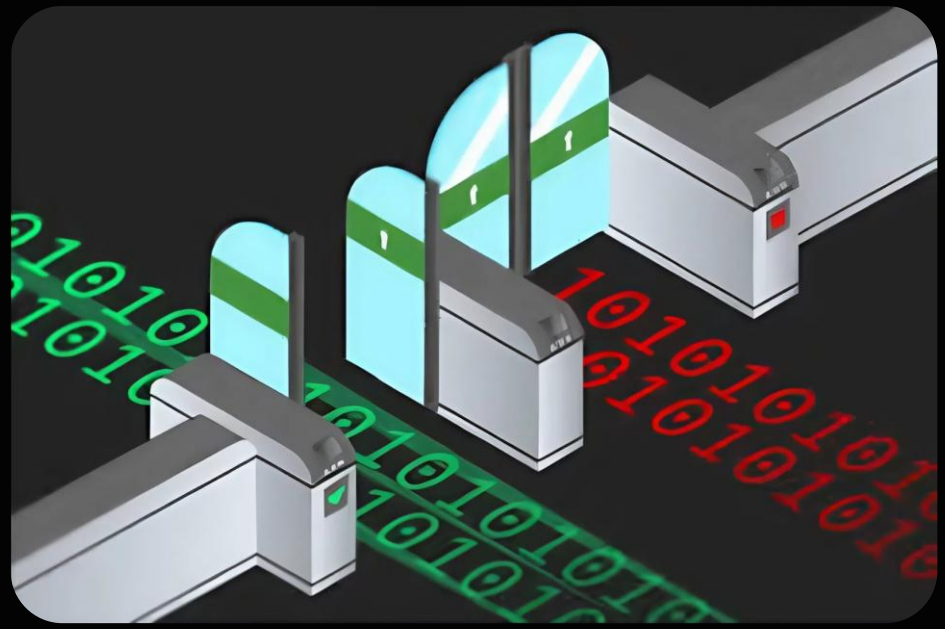
LUNTRY

5 доменов:

- Анализ Dockerfile на лучшие практики
- Поиск чувствительной информации
- Анализ на известные уязвимости
- Анализ на вредоносный код
- Компонентный анализ

Security/Quality Gates – ворота качества

- CI
  - Уведомление
  - Блокировка
- Image registry
  - Уведомление
- Runtime
  - Уведомление



Блокировка на Deploy, Runtime – опасная и плохая практика!

«Безопасность Kubernetes кластеров: вредные советы», DevOpsConf 2024

Подпись образа

LUNTRY

- Самый последний шаг в CI pipeline
- Только если все проверки прошли успешно и Security/Quality Gates не заблокировали CI pipeline
- Проверка подписи образа может быть
  - На Deploy в Admission Controller
  - В момент запуска на Node в CRI
    - Умеет CRI-O с версии 1.28
    - В Kubernetes версии 1.28 появилась новая image pull ошибка с кодом SignatureValidationFailed

«Подпись и валидация образов в Kubernetes», Вебинар Luntry

Полезные инструменты

- Sigstore
- Notary project
- Connaisseur
- In-toto
- Kyverno
  - verifyImages политики
- policy-controller

```
spec:
  validationFailureAction: Enforce
  background: false
  rules:
    - name: verify-image
      match:
        any:
        - resources:
            kinds:
              - Pod
      verifyImages:
      - imageReferences:
        - "ghcr.io/kyverno/test-verify-image*"
        mutateDigest: true
        attestors:
        - entries:
          - keys:
              publicKeys: |
                -----BEGIN PUBLIC KEY-----
                MFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAE8nXRh950IZbRj8Ra/N9sbqOPZrfM
                5/KAQN0/KjHcorm/J5yctVd7iEcnessRQjU917hmKO6JWVGHpDguIyakZA==
                -----END PUBLIC KEY-----
```

LUNTRY

Registry Staging

- Доверенные источники
- Разделение сред
- Разделение доступов
- Разделение стратегий
  - Очистки
  - Анализа

13.2 A lack of segregation between production and non-production container registries may result in insecure images deployed to the production environment.	**a.** Consider using two registries, one for production or business-critical workloads and one for development/test purposes, to assist in preventing image sprawl and the opportunity for an unmaintained or vulnerable image being accidentally pulled into a production cluster.

"Guidance for Containers and Container Orchestration Tools"
от PCI Security Standards Council

Registry Staging

Due to the use of open source components that are often pulled from public sources, organizations should create several stages of registries in their pipelines. Only authorized developers should be able to pull base images from public registries and store them in an internal registry for wide consumption within the organization. It is also advised to have separate private registries for keeping development artifacts per team or group, and finally a staging or pre-production registry for images ready for production. This enables tighter control over the provenance and security of open source components, while enabling different types of testing for stages in the CI/CD chain.

For any registry used, access control through a dedicated authentication and permission model must be implemented. Use mutually authenticated TLS for all registry connections (among other interactions within the architecture).

“Cloud Native Security Whitepaper”
от CNCF

LUNTRY

АНАЛИЗ ОБРАЗА

Анализ Dockerfile на лучшие практики

LUNTRY

- Мысли:
  - Проверяем на лучшие практики, опасные/нежелательные конструкции, эксплоиты и что угодно в соответствии с внутренними регламентами

- Проверки:
  - Docker Building best practices
  - CIS Docker Benchmarks
  - Собственные требования

- OpenSource инструменты: Dockle*, Hadolint, KICS, Checkov, Semgrep

Компонентный анализ

LUNTRY

- Мысли:
  - Инвентаризация всего стороннего внутри образа
  - Важно для идентификации уязвимых компонент, актуальности версий, лицензионной чистоты
  - Форматы SBOM: Software Package Data Exchange (SPDX), CycloneDX (CDX), Software Identification (SWID)

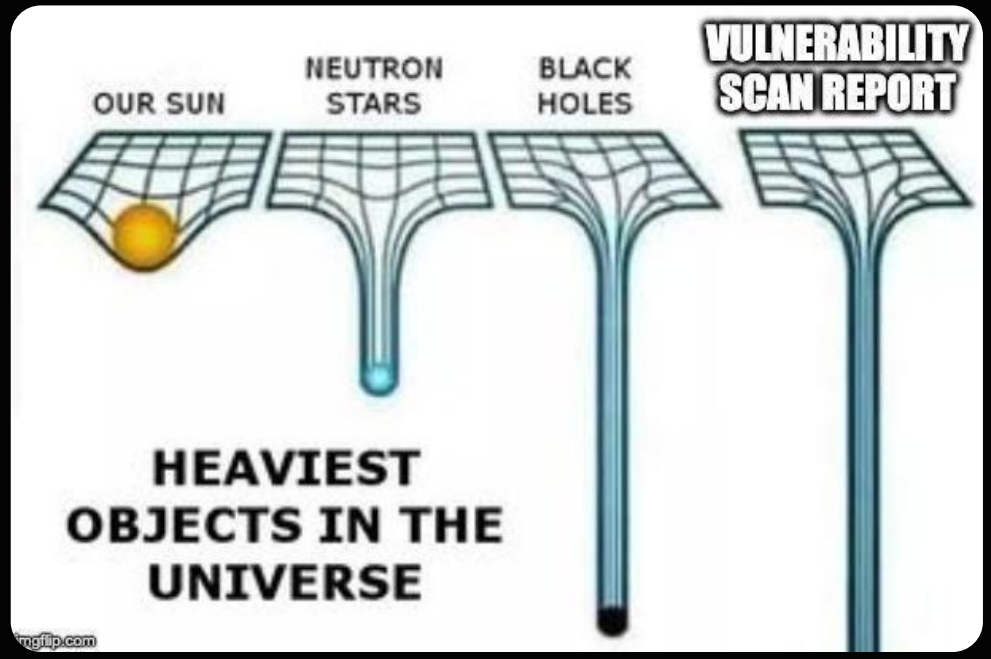
- OpenSource инструменты: syft, cdxgen, trivy

Анализ на известные уязвимости

LUNTRY

- Мысли:
  - Важная часть процесса управления уязвимостями
  - В любой момент времени всегда есть какая-то незакрытая уязвимость
  - Окружений без уязвимостей не бывает
  - Важно уметь приоритезировать уязвимости

- OpenSource инструменты:
  Trivy, Grype, Clair



Проблемы поиска известных уязвимостей

Все подходы на базе статического анализа позволяют защититься только лишь от легитимного пользователя, но не являются никакой помехой для вредоносного!

Работа с уязвимостями

- Application Security Orchestration and Correlation (ASOC)
- Экспорт результатов в CycloneDX формате в ASOC платформы
- ASOC - инструмент для работы с найденными уязвимостями
  - Агрегирует данные из разных сканеров (DAST, SAST, SCA, …)
  - Помогает делать триаж уязвимостей (работа с дубликатами тд)
  - Назначение задач на исправление, приоритезация, контроль исправлений
  - Интеграция с VCS, tasktracker, registry, сканерами
  - Формирование отчетов по уязвимостям

- Инструменты: DefectDojo, Hexway Vampy

LUNTRY

Поиск чувствительной информации

LUNTRY

- Мысли:
  - Важно сканировать в CI и блокировать при обнаружении
    - При попадании образа с секретом в registry нужно заводить инцидент и ротировать секрет!

- Подходы:
  - На основе сигнатур
  - На основе энтропии
  - Гибридный подход

- OpenSource инструменты: SecretScanner, truffleHog

Анализ на вредоносный код

LUNTRY

- Мысли:
  - Важно обнаруживать не только вредоносный код, но и код двойного назначения (GTFOBins)
  - Не тащите и не запускайте непонятные образы извне!
  - Комбинируйте с динамикой – Runtime Security

- OpenSource инструменты: Package Hunter, Malcontent, PISC

Проблемы OpenSource

- Отсутствие профессиональной поддержки и обслуживания
- Отсутствие российской специфики (ОС, feeds, сертификации)
- Необходимость высокой внутренней экспертизы для работы с данными проектами
- Зависимость от внутренних носителей знаний, которые могут уйти из компании
- Как правило отсутствие энтерпрайзных фич в бесплатных версиях
- Сложность интеграции и совместимости
- Для решения данной задачи нужен целый разнородный набор инструментов, над которым придётся писать еще обвязку и поддерживать ее
- Скрытые затраты на “бесплатное”

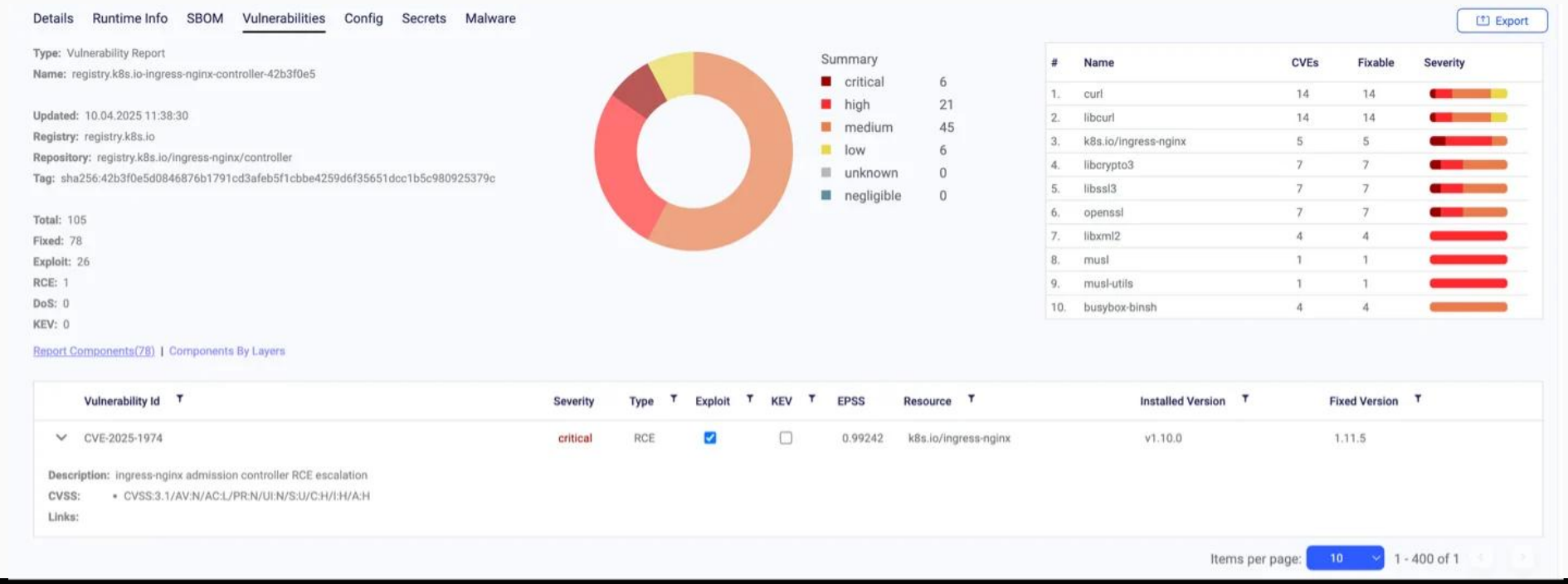
LUNTRY

LUNTRY

КАК LUNTRY ПОМОГАЕТ РЕШИТЬ ЗАДАЧИ, СВЯЗАННЫЕ С БЕЗОПАСНОСТЬЮ ОБРАЗОВ

Возможности Luntry

- Встраивание в CI\CD, Registry, Runtime
- Инвентаризация использования образов
- Компонентный анализ (SBOM)
- Сканирование на известные уязвимости
- Поддержка отечественных ОС и БДУ ФСТЭК
- Приоритизация уязвимостей CVSS, Severity, Type(RCE/DoS), Exploit, KEV, EPSS, Runtime
- Контроль соответствия лучшим практикам
- Поиск секретов
- Анализ на вредоносный код и код двойного назначения
- Поддержка внешних и собственных IoC
- Встроенная библиотека уникальных проверок
- Скорость и парализация сканирований
- Security Gates в CI\CD, Registry, Runtime



Что за сканер внутри Luntry?

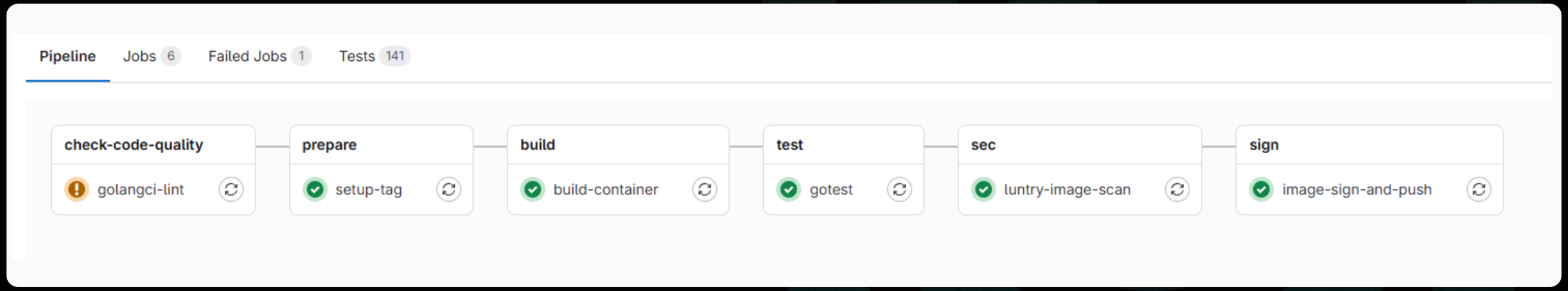
LUNTRY

- Собственная разработка
  - Для анализа Dockerfile на лучшие практики используется движок YARA
  - Для компонентного анализа используется библиотека Syft
  - Для анализа на известные уязвимости используется библиотека Grype
  - Для поиска чувствительной информации используется движок YARA
  - Для поиска вредоносного кода и кода двойного назначения используется движок YARA

- Оптимизации
- Умный параллелизм сканирований
- Высокая скорость сканирования
- Работа в закрытом контуре

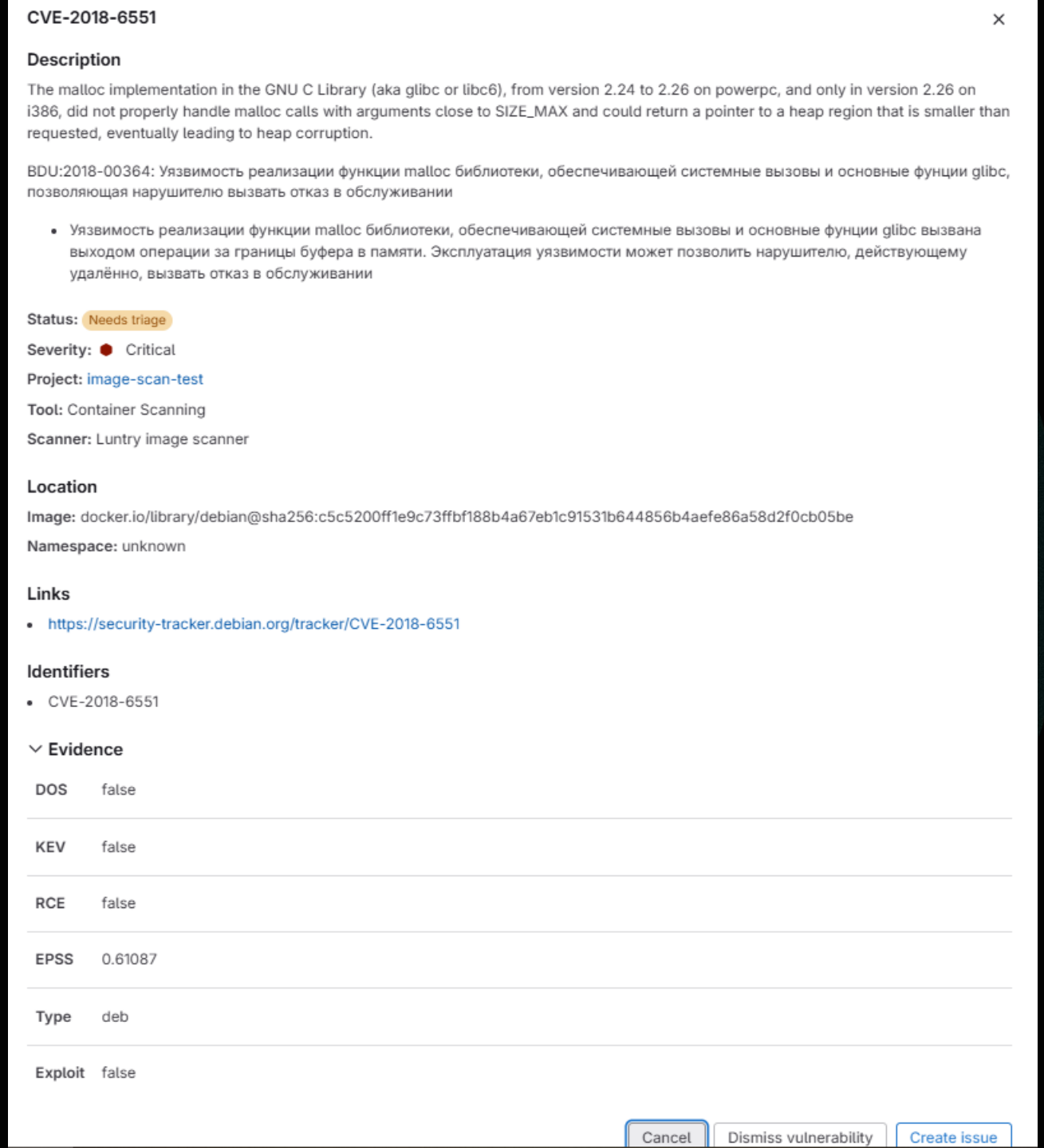
Пример встраивания в CI

LUNTRY



Интеграция с CI GitLab

- Нативная интеграция
  - Очень полезно для разработчиков



LUNTRY

Поддерживаемые feeds уязвимостей

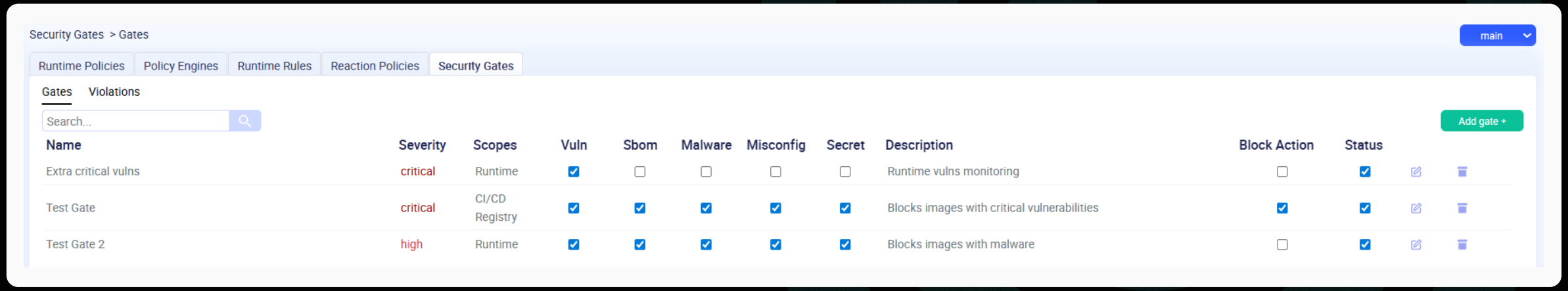
- Alpine Linux SecDB
- Amazon Linux ALAS
- Chainguard SecDB
- Debian Linux CVE Tracker
- Echo Security Advisories
- GitHub Security Advisories (GHSAs)
- MinimOS SecDB
- National Vulnerability Database (NVD)
- Oracle Linux OVAL
- RedHat Linux Security Data
- RedHat RHSAs
- SUSE Linux OVAL
- Ubuntu Linux Security
- Wolfi SecDB

- БДУ ФСТЭК
- Alt Linux
- РЕД ОС
- Astra Linux
- Sber Linux

LUNTRY

Live Demo

- Security/Quality Gates в Luntry



RoadMap Luntry

- Новый UI
- Новые режимы сканирования
- Пользовательские правила на YARA для всего
- Продвинутая приоритезация уязвимостей
- Гибкие фильтры для уязвимостей с учетом пакетов, слоев образа и т.д.
- Поддержка из коробки коммерческих TI feeds
- …

Мы всегда рады вашей обратной связи и features requests =)

LUNTRY

ВЫВОДЫ

Выводы

LUNTRY

01 Безопасность образов это не только их анализ, но и сборка и хранение

02 Важно анализировать образы по всем 5 доменам безопасности на всем жизненном цикле образа – CI\CD, Registry, Runtime

03 Security/Quality Gates на всем жизненном цикле с разными политиками облегчат работу и контроль

04 Luntry позволяет закрыть и работать со всеми аспектами безопасности образов контейнеров на всех стадиях

Полезные ссылки

- Все выступления Анатолия Карпенко ;)
  - https://luntry.ru/research

LUNTRY

LUNTRY

luntry_official
luntrysolution
luntrysolution

luntry.ru
info@luntry.ru

ДМИТРИЙ ЕВДОКИМОВ
Founder & CTO Luntry

de@luntry.ru
Qu3b3c
k8security

СПАСИБО ЗА ВНИМАНИЕ!