

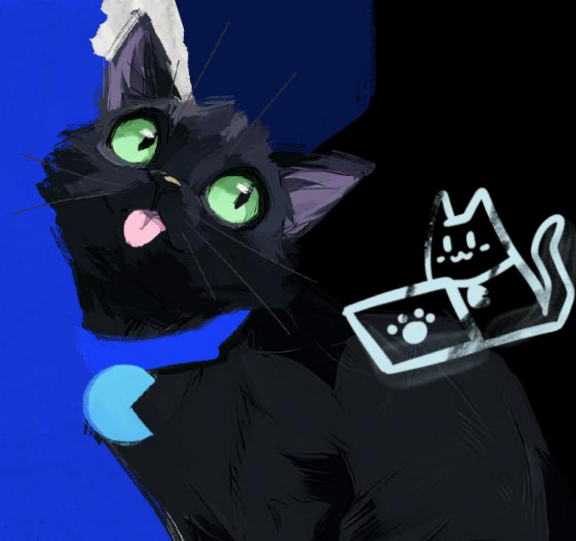
cyber
camp

LUNTRY

Путеводитель по безопасности контейнеров и Kubernetes

Дмитрий Евдокимов
Founder&CTO Luntry

27.03.2026



Обо мне

Дмитрий Евдокимов

” 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 конференций DevOpsConf и KUBER CONF

Спикер

VK Kubernetes
DevOpsConf
Kazhackstan

Confidence
HackInParis
HighLoad++

ZeroNights
KuberConf
OFFZONE

БеКон
BlackHat
DevOops

HITB
PHDays
SAS

План доклада

- 01 Подготовка к путешествию
- 02 Исследование мира безопасности контейнеров и Kubernetes
- 03 Мир глазами злоумышленников
- 04 Мир глазами защитников
- 05 Полезности
- 06 Выводы



Подготовка к путешествию

Зачем?



Зачем?

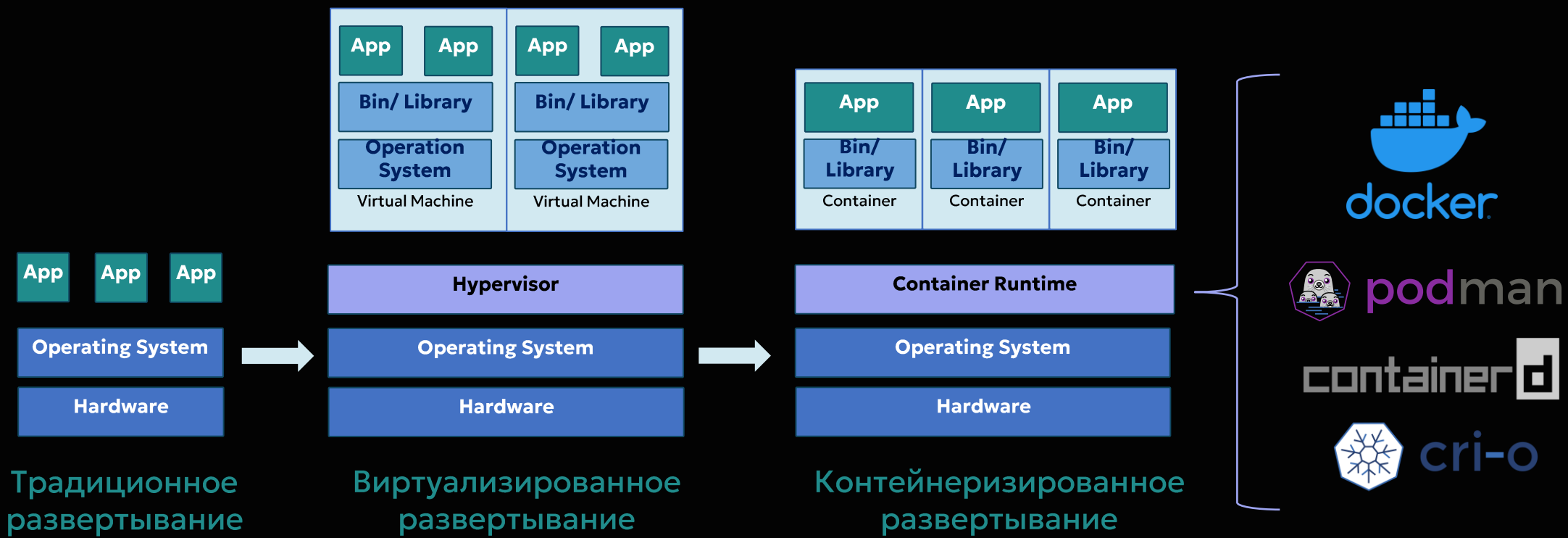
- **Контейнеризация:** Более 90% организаций уже используют контейнеры в производственных средах (production).
- **Kubernetes (K8s):** По данным ежегодного опроса CNCF за 2025 год, использование Kubernetes в production достигло 82%, закрепив за ним статус «де-факто операционной системы» для облачных вычислений и ИИ-приложений.
- **Рост экосистемы:** С момента присоединения Kubernetes к CNCF количество компаний-контрибьюторов выросло почти на 1000% (с 731 до более чем 8000).
- **Тренды 2026:** Основными направлениями развития остаются Platform Engineering, безопасность цепочек поставок программного обеспечения и интеграция ИИ-нагрузок непосредственно в K8s-кластеры.

Фундаментальное изменение в разработке

1. Улучшенная упаковка приложений
2. Изоляция приложений
3. Лёгкость и скорость по сравнению с виртуальными машинами
4. Поддержка микросервисной архитектуры
5. Упрощение CI/CD-пайплайнов
6. Декларативность и Infrastructure as Code
7. Горизонтальное масштабирование
8. Оптимизация использования ресурсов
9. Портативность между средами и облаками
10. Упрощение onboarding-а новых разработчиков
11. Быстрое восстановление и самовосстановление
12. Безопасность через ограничение поверхности атаки
13. Богатая экосистема и стандартизация
14. Поддержка blue-green и canary-деплоев

АСПЕКТ	БЕЗ КОНТЕЙНЕРОВ	С КОНТЕЙНЕРАМИ
Настройка среды	Часы / дни	Минуты
Консистентность сред	Ручная синхронизация	Гарантирована образом
Масштабирование	Ручное, медленное	Автоматическое, мгновенное
Плотность размещения	Низкая (ВМ)	Высокая
Откат версий	Рискованный	Мгновенный
Vendor lock-in	Высокий	Минимальный

Эволюция к контейнерам



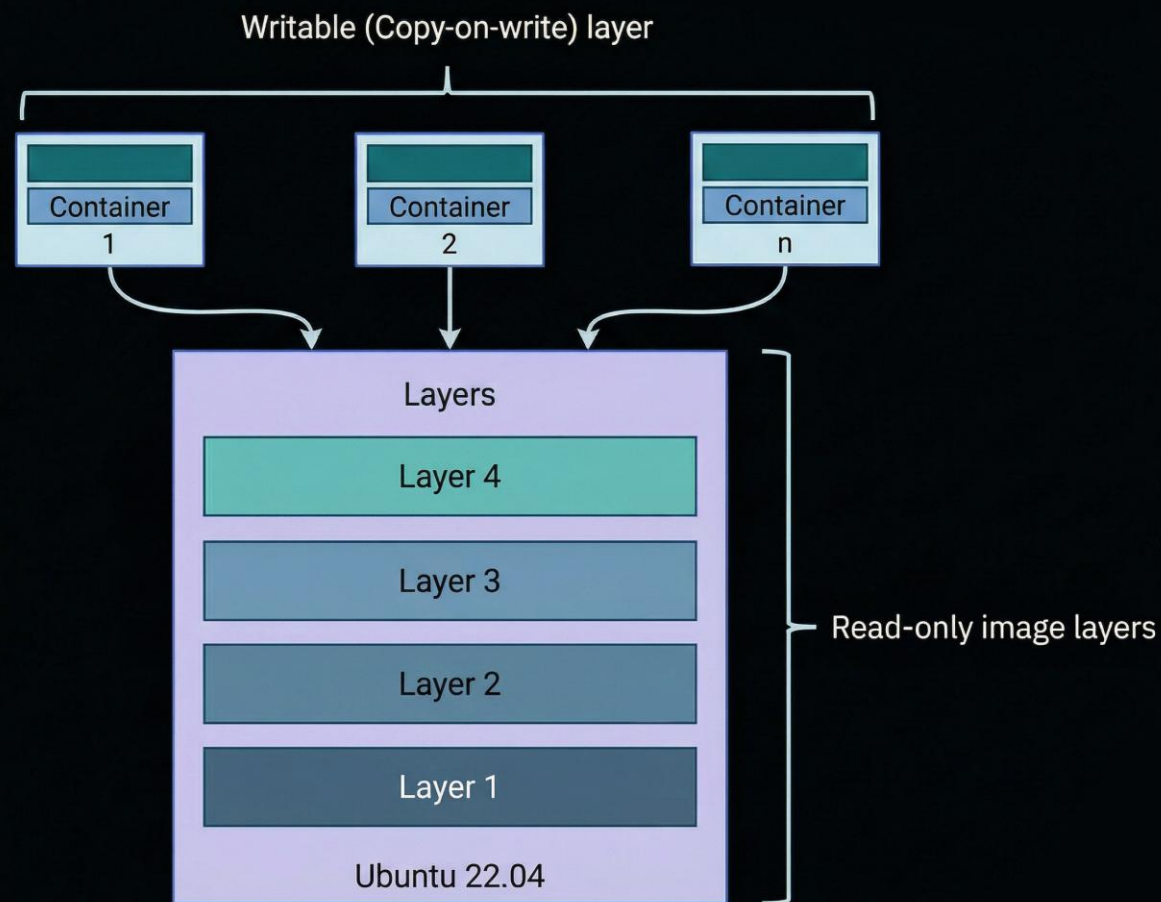
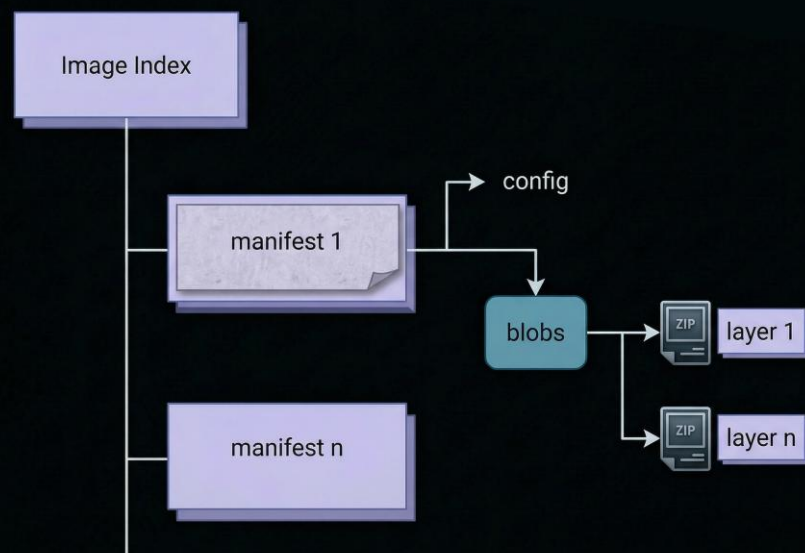
Что такое контейнер?

- 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
```

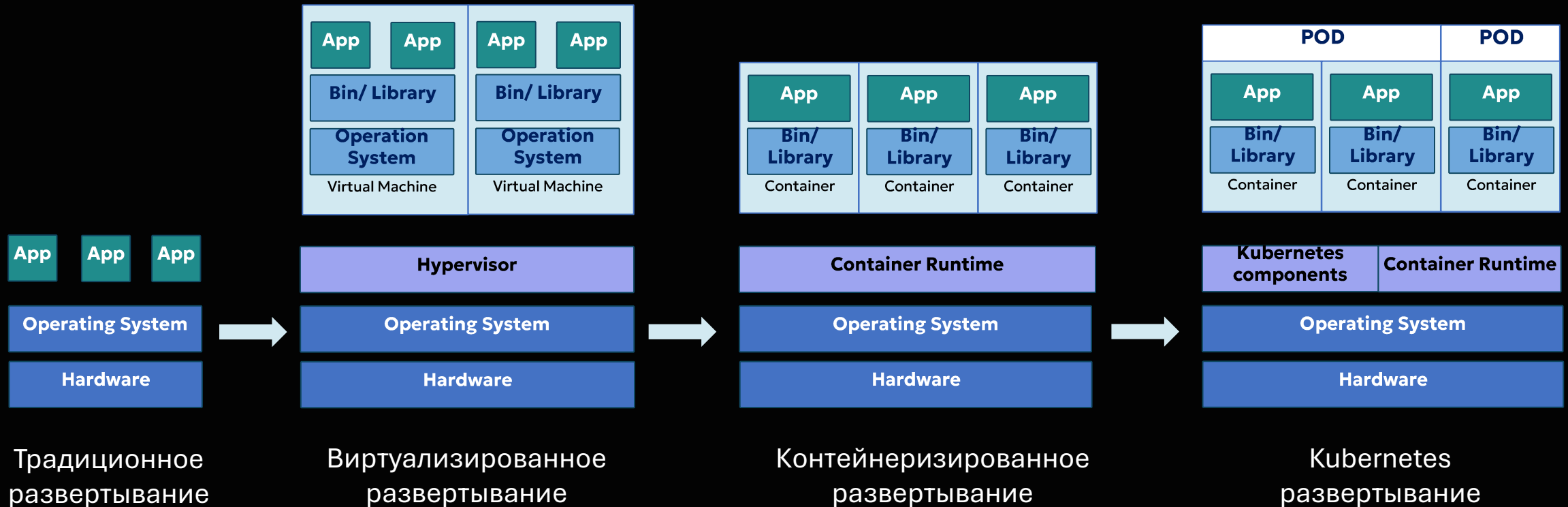
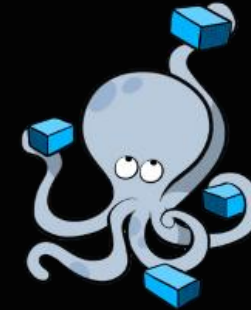
Что такое образ контейнера?

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



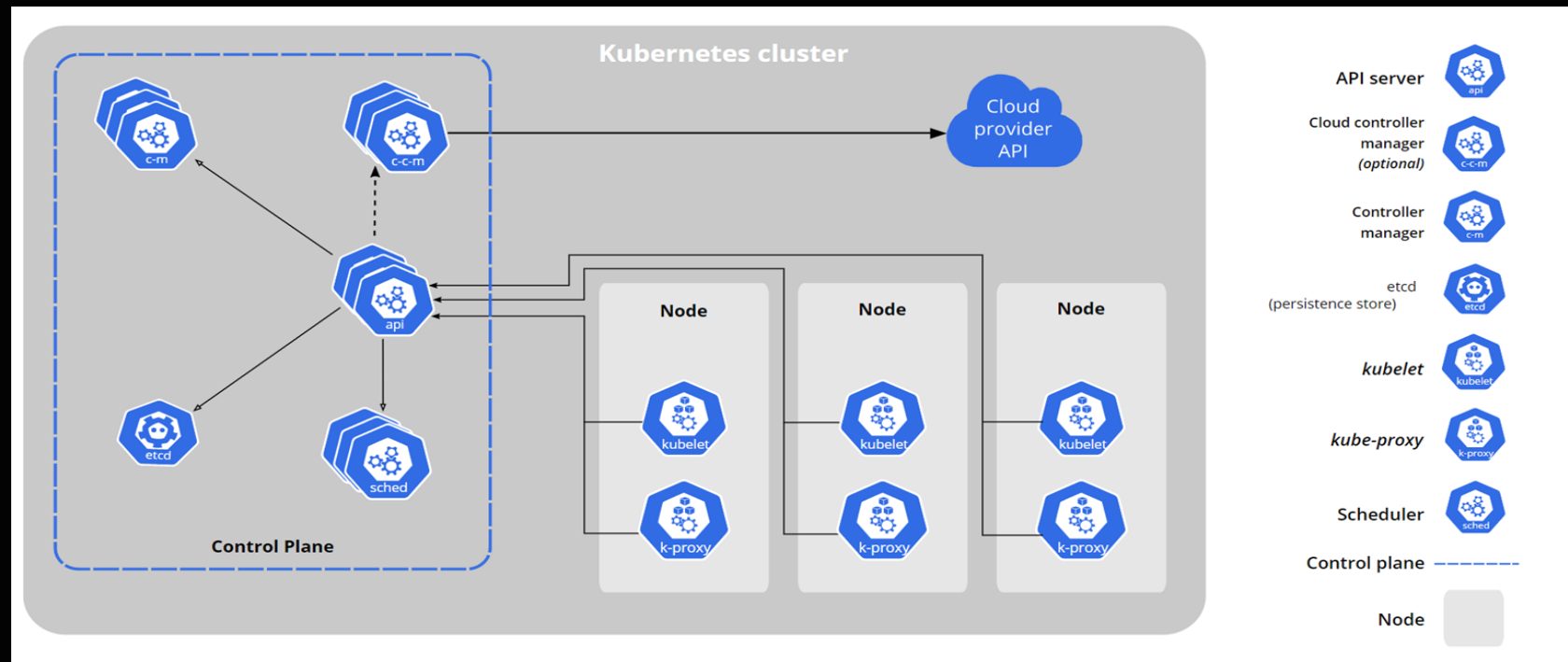
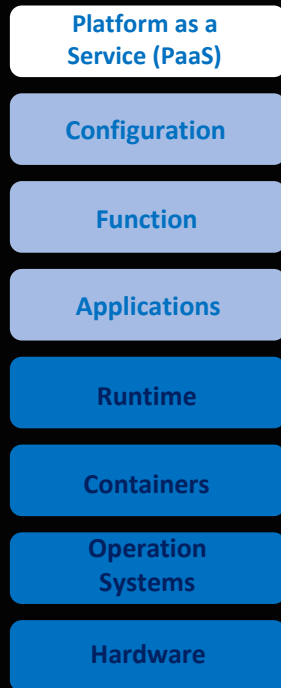
Эволюция от контейнеров к оркестраторам контейнеров

Оркестраторы:  **kubernetes**

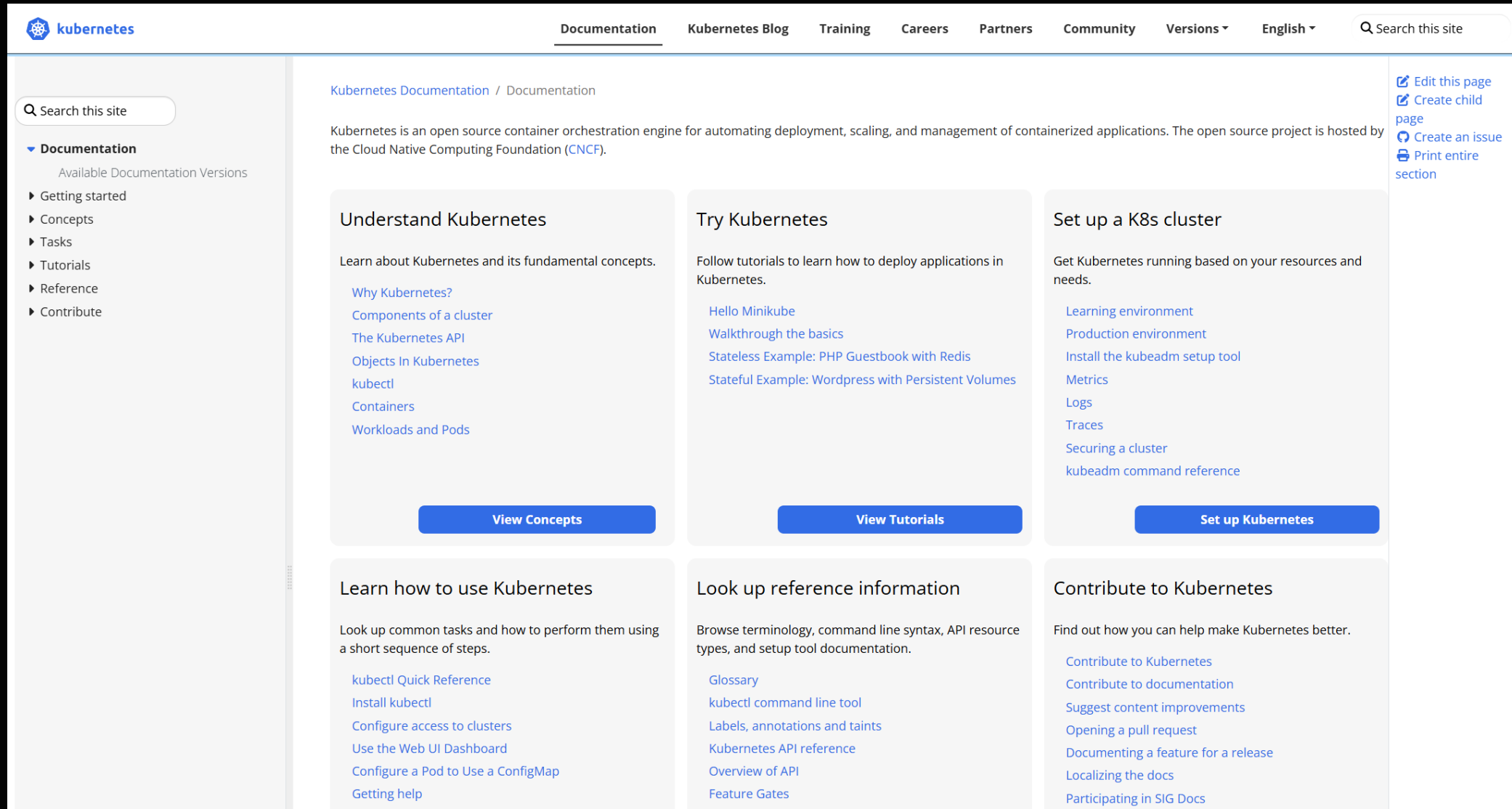


Что такое Kubernetes?

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



Официальная документация Kubernetes



The screenshot shows the Kubernetes documentation homepage. At the top, there is a navigation bar with links for Documentation, Kubernetes Blog, Training, Careers, Partners, Community, Versions, and English. A search bar is located on the right side of the navigation bar. Below the navigation bar, the main content area is divided into several sections. On the left, there is a sidebar with a search bar and a list of documentation categories. The main content area features a header section with the title 'Kubernetes Documentation / Documentation' and a brief introduction. Below this, there are three main columns of content, each with a title, a brief description, a list of links, and a button. The first column is 'Understand Kubernetes', the second is 'Try Kubernetes', and the third is 'Set up a K8s cluster'. Below these, there are three more columns: 'Learn how to use Kubernetes', 'Look up reference information', and 'Contribute to Kubernetes'. On the right side of the main content area, there are links for 'Edit this page', 'Create child page', 'Create an issue', and 'Print entire section'.

Documentation | Kubernetes Blog | Training | Careers | Partners | Community | Versions | English | Search this site

Search this site

Documentation
Available Documentation Versions

- ▶ Getting started
- ▶ Concepts
- ▶ Tasks
- ▶ Tutorials
- ▶ Reference
- ▶ Contribute

Kubernetes Documentation / Documentation

Kubernetes is an open source container orchestration engine for automating deployment, scaling, and management of containerized applications. The open source project is hosted by the Cloud Native Computing Foundation (CNCF).

Understand Kubernetes

Learn about Kubernetes and its fundamental concepts.

- [Why Kubernetes?](#)
- [Components of a cluster](#)
- [The Kubernetes API](#)
- [Objects In Kubernetes](#)
- [kubectl](#)
- [Containers](#)
- [Workloads and Pods](#)

[View Concepts](#)

Try Kubernetes

Follow tutorials to learn how to deploy applications in Kubernetes.

- [Hello Minikube](#)
- [Walkthrough the basics](#)
- [Stateless Example: PHP Guestbook with Redis](#)
- [Stateful Example: Wordpress with Persistent Volumes](#)

[View Tutorials](#)

Set up a K8s cluster

Get Kubernetes running based on your resources and needs.

- [Learning environment](#)
- [Production environment](#)
- [Install the kubeadm setup tool](#)
- [Metrics](#)
- [Logs](#)
- [Traces](#)
- [Securing a cluster](#)
- [kubeadm command reference](#)

[Set up Kubernetes](#)

Learn how to use Kubernetes

Look up common tasks and how to perform them using a short sequence of steps.

- [kubectl Quick Reference](#)
- [Install kubectl](#)
- [Configure access to clusters](#)
- [Use the Web UI Dashboard](#)
- [Configure a Pod to Use a ConfigMap](#)
- [Getting help](#)

Look up reference information

Browse terminology, command line syntax, API resource types, and setup tool documentation.

- [Glossary](#)
- [kubectl command line tool](#)
- [Labels, annotations and taints](#)
- [Kubernetes API reference](#)
- [Overview of API](#)
- [Feature Gates](#)

Contribute to Kubernetes

Find out how you can help make Kubernetes better.

- [Contribute to Kubernetes](#)
- [Contribute to documentation](#)
- [Suggest content improvements](#)
- [Opening a pull request](#)
- [Documenting a feature for a release](#)
- [Localizing the docs](#)
- [Participating in SIG Docs](#)

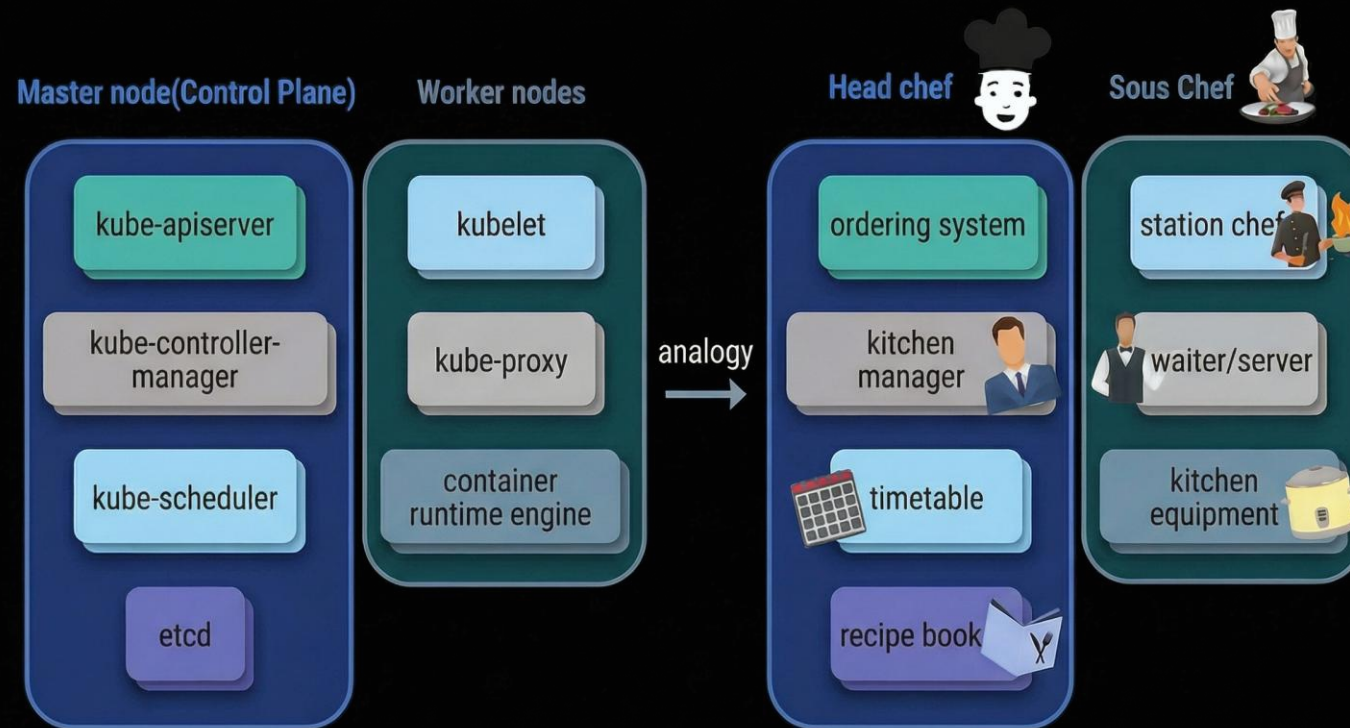
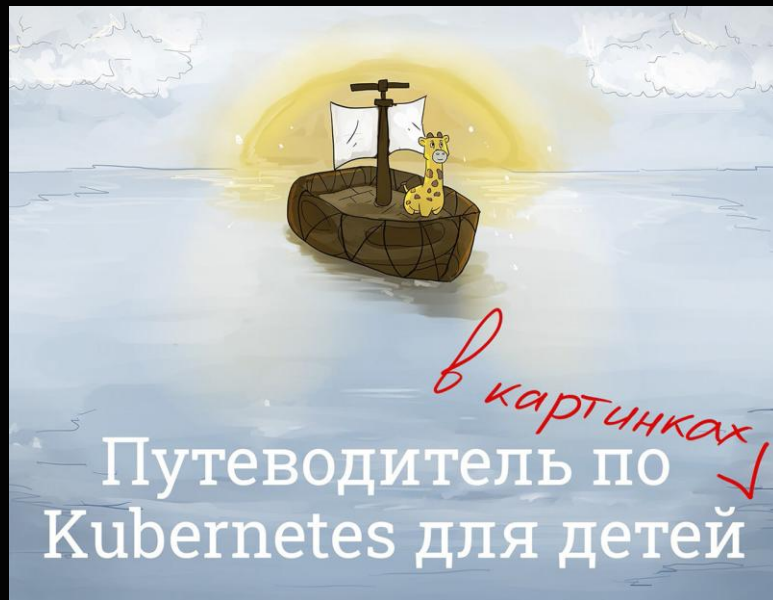
[Edit this page](#)
[Create child page](#)
[Create an issue](#)
[Print entire section](#)

Дистрибутивы Kubernetes

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



Kubernetes на простом языке

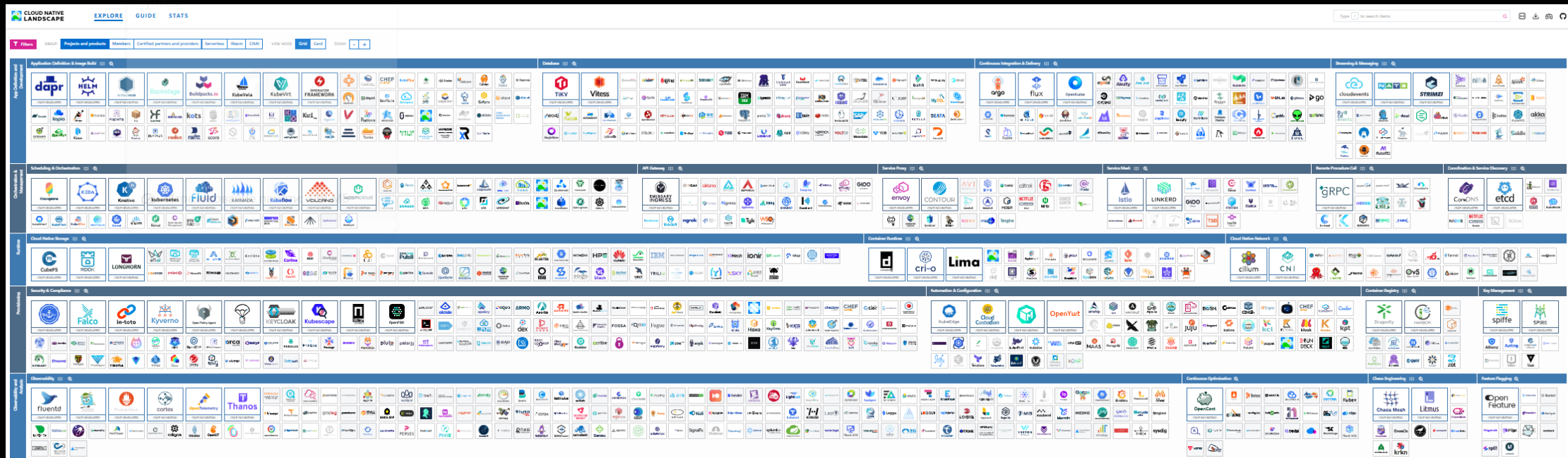


[Kubernetes в переводе на детский](#)

[Понимание архитектуры Kubernetes
через аналогию с шеф-поваром
ресторана](#)

CNCF Landscape

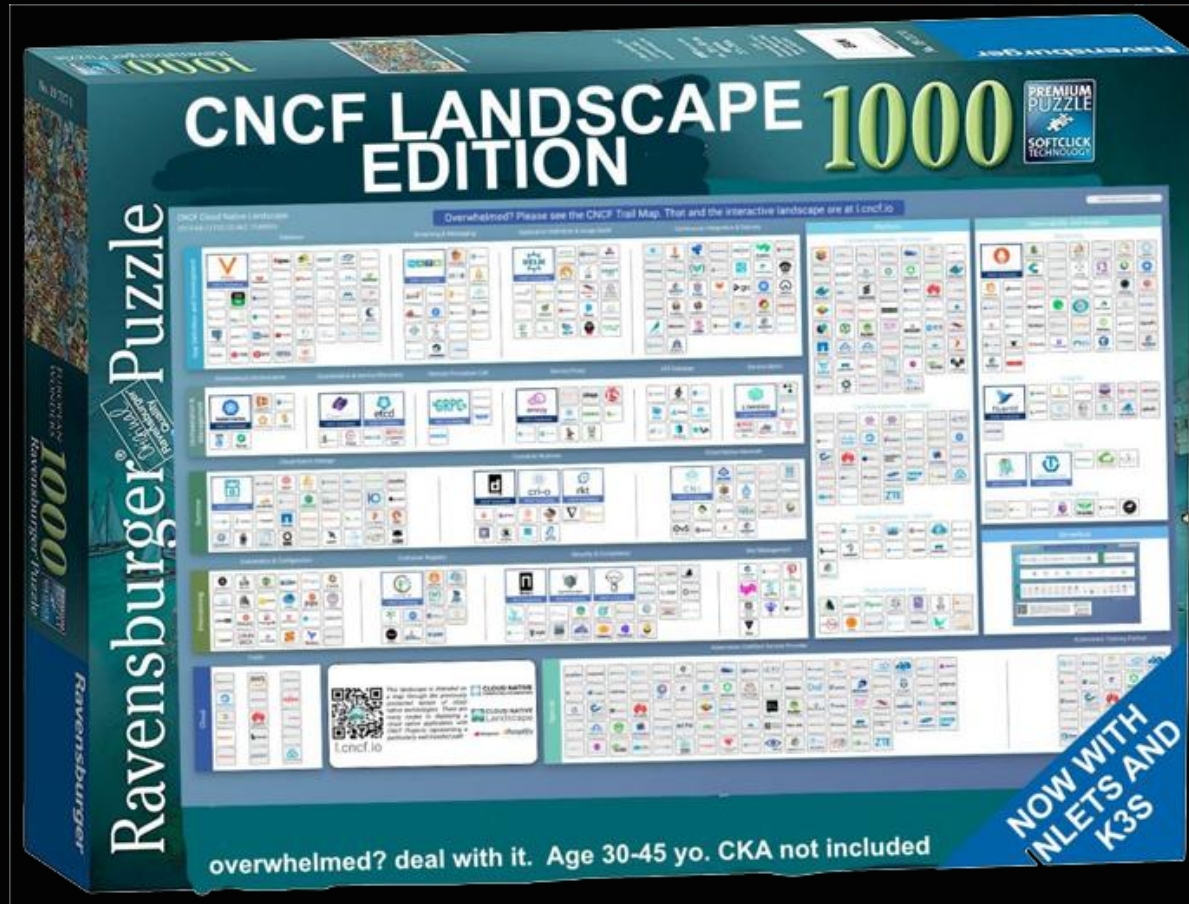
Cloud Native экосистема просто огромная и постоянно только растет!



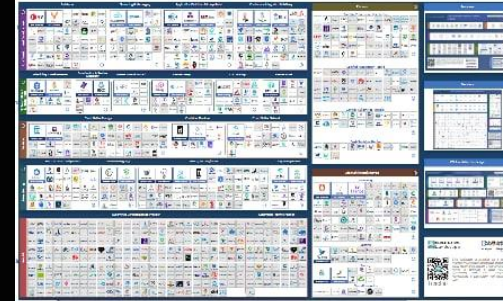
<https://landscape.cncf.io/>

CNCF Landscape

=)



Cloud Native Landscape



[Gi-hun] I'm dead.

Изучаем на практике

iximiuz Labs Catalog Learning Paths Playgrounds Docs ^{New!} Pricing More Sign in Sign up

Welcome to **iximiuz Labs**

Indie Learning Platform to Master Server Side Craft

Interactive Courses and Tutorials for Mastering Server Side Technologies

```
> Learn the skills you need for the job
> in the most hands-on way possible 🧑‍🔧
>
```

☆ Tutorial on Security

An Introduction to Docker Security

by Rory McCune

DOCKER SECURITY INTRODUCTION

☆ Course on Kubernetes

Kubernetes the (Very) Hard Way

by Márk Sági-Kazár

☆ Tutorial on Kubernetes

Kubernetes: Runtime Class

by Márk Sági-Kazár

Container runtime

<https://labs.iximiuz.com/>



Исследование мира безопасности контейнеров и Kubernetes

Cloud Native Security Whitepaper

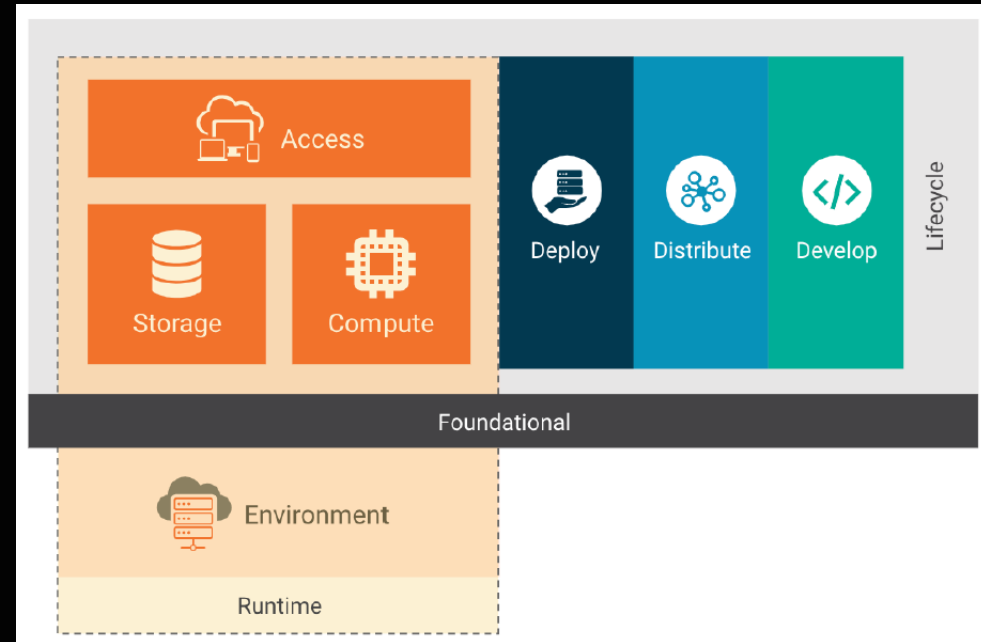
Security concerns within this landscape are complex because of the explicit focus on rapid development and deployment in addition to impracticality of traditional perimeter-based security. This complexity requires a paradigm shift to protect applications by migrating from a perimeter based approach to one where security moves closer to dynamic workloads that are identified based on attributes and metadata (e.g. labels and tags). This approach identifies and secures workloads to meet the scale needs of cloud native applications while accommodating constant flux.

Cloud native development can be modeled in distinct phases that constitute the application lifecycle: "Develop," "Distribute," "Deploy," and "Runtime." Cloud native security contrasts with traditional security approaches in that there is a tremendous opportunity to ensure that security is injected throughout these distinct phases instead of bookending the lifecycle with separately managed security informed interventions.



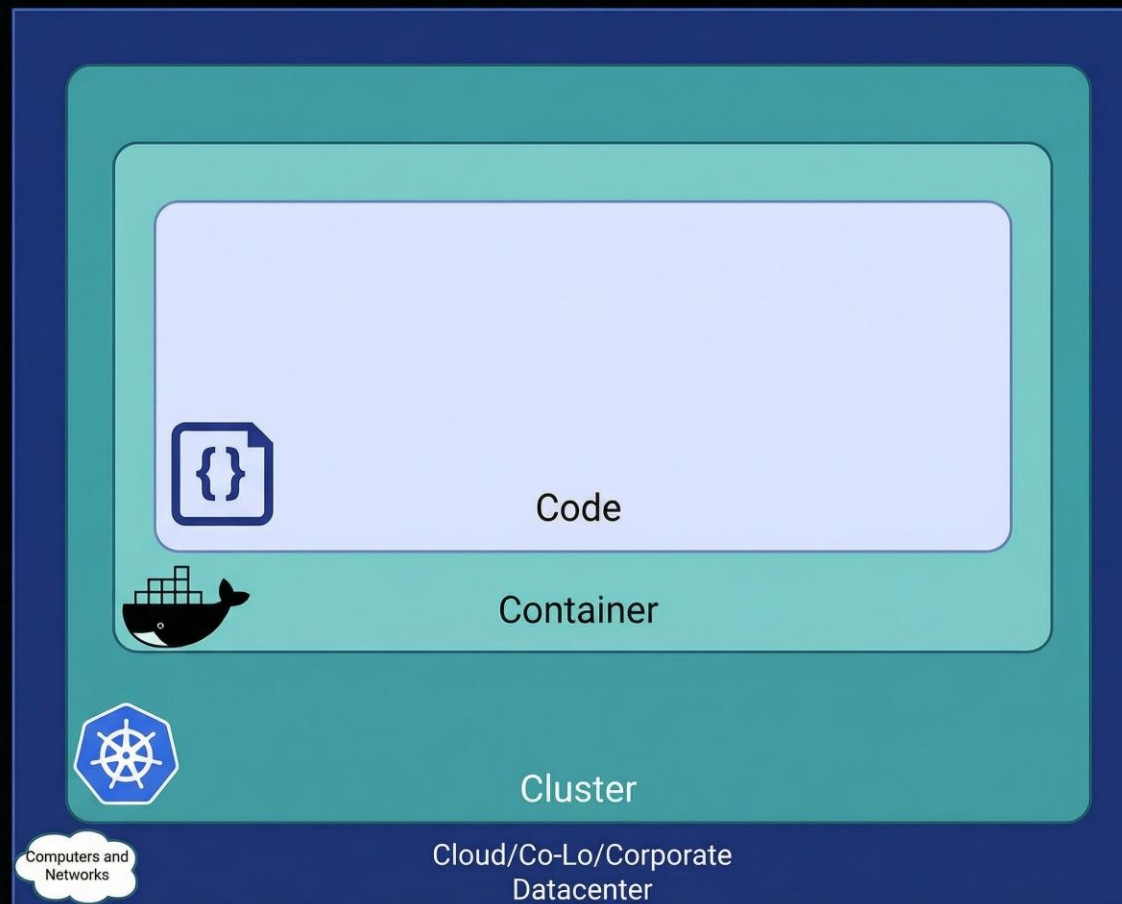
Фазы жизненного цикла

- **Develop - разработка**
 - Security как можно раньше
 - Infrastructure as Code (IaC)
- **Distribute - распространение**
 - Безопасность supply chain
 - Постоянное автоматическое сканирование и обновление артефактов
 - Подпись артефактов
- **Deploy - развертывание**
 - Проверка атрибутов рабочих нагрузок
 - Параллельное развертывание дополнительных средств
- **Runtime - выполнение**
 - Применение политик ограничения возможностей и ресурсов
 - Обнаружение, предотвращение и уведомление о несанкционированном доступе к ресурсам



Подход 4С для безопасности Cloud Native

- Cloud/Co-Lo/Corporate Datacenter
 - Управление активами
- Cluster
 - Компоненты Kubernetes
 - Kubernetes ресурсы
- Container
 - Процессы
 - Взаимодействие с файлами
 - Взаимодействие с сетью
- Code
 - Образы контейнеров



7 доменов безопасности Kubernetes

Безопасность
Kubernetes-ресурсов

Контроль состояния
Kubernetes-кластеров

Контроль соответствия
кластера стандартам

Управление безопасностью
образов контейнеров

Сетевая безопасность

Анализ прав доступа

Защита Runtime

Жизненный путь образа

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

Shift Left Security



Shift Everywhere Security



Shift Right Security



Shift Down Security

Домены безопасности образов

5 ДОМЕНОВ:

Анализ Dockerfile
на лучшие практики

Поиск чувствительной
информации

Анализ на известные
уязвимости

Анализ на вредоносный код
и код двойного назначения

Компонентный анализ



Мир глазами злоумышленников

Модели нарушителя

ВНЕШНИЙ

Попадает внутрь контейнера Pod и оттуда обращается к рядом стоящим сервисам или делает побег из контейнера на Node

ВНУТРЕННИЙ

Злоумышленник с Node или из одной сети с Kubernetes

СКОМПРОМЕТИРОВАННЫЙ РАЗРАБОТЧИК

Контролирует / формирует содержимое YAML ресурсов, образов контейнеров

Матрица угроз для контейнеров от MITRE

MATRICES

- Enterprise ^
- PRE
- Windows
- macOS
- Linux
- Cloud ▾
- Network Devices
- Containers**
- ESXi
- Mobile ▾
- ICS

Containers Matrix

Below are the tactics and techniques representing the MITRE ATT&CK® Containers platform. The techniques below are known to target containers and container orchestration systems such as Kubernetes. The Matrix contains information for the Containers platform.

[View on the ATT&CK® Navigator](#)

[Version Permalink](#)

layout: side ▾ show sub-techniques hide sub-techniques help

Initial Access 3 techniques	Execution 4 techniques	Persistence 7 techniques	Privilege Escalation 6 techniques	Defense Evasion 7 techniques	Credential Access 3 techniques	Discovery 3 techniques	Lateral Movement 1 techniques	Impact 5 techniques
Exploit Public-Facing Application	Container Administration Command	Account Manipulation (1)	Account Manipulation (1)	Build Image on Host	Brute Force (3)	Container and Resource Discovery	Use Alternate Authentication Material (1)	Data Destruction
External Remote Services	Deploy Container	Create Account (1)	Create or Modify System Process (1)	Deploy Container	Steal Application Access Token	Network Service Discovery		Endpoint Denial of Service
Valid Accounts (2)	Scheduled Task/Job (1)	Create or Modify System Process (1)	Escape to Host	Impair Defenses (1)	Unsecured Credentials (2)	Permission Groups Discovery		Inhibit System Recovery
	User Execution (1)	External Remote Services	Exploitation for Privilege Escalation	Indicator Removal				Network Denial of Service
		Implant Internal Image	Scheduled Task/Job (1)	Masquerading (2)				Resource Hijacking (2)
		Scheduled Task/Job (1)	Valid Accounts (2)	Use Alternate Authentication Material (1)				
		Valid Accounts (2)		Valid Accounts (2)				

Матрица угроз для Kubernetes от Microsoft

Microsoft Threat Matrix for Kubernetes
Search

Tactics Mitigations About

Tactics

- Initial Access >
- Execution >
- Persistence >
- Privilege Escalation >
- Defense Evasion >
- Credential Access >
- Discovery >
- Lateral Movement >
- Collection >
- Impact >

Tactics

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Impact
Using cloud credentials	Exec into container	Backdoor container	Privileged container	Clear container logs	List K8S secrets	Access Kubernetes API server	Access cloud resources	Images from a private registry	Data destruction
Compromised image in registry	bash/cmd inside container	Writable hostPath mount	Cluster-admin binding	Delete K8S events	Mount service principal	Access Kubelet API	Container service account	Collecting data from pod	Resource hijacking
Kubeconfig file	New container	Kubernetes CronJob	hostPath mount	Pod / container name similarity	Container service account	Network mapping	Cluster internal networking		Denial of service
Application vulnerability	Application exploit (RCE)	Malicious admission controller	Access cloud resources	Connect from proxy server	Application credentials in configuration files	Exposed sensitive interfaces	Application credentials in configuration files		
Exposed sensitive interfaces	SSH server running inside container	Container service account			Access managed identity credentials	Instance Metadata API	Writable hostPath mount		
	Sidecar injection	Static pods			Malicious admission controller		CoreDNS poisoning		
							ARP poisoning and IP spoofing		

Disclaimer

The purpose of the Threat Matrix for Kubernetes is to educate readers on the potential of Kubernetes-based tactics, techniques, and procedures (TTPs). It is not to teach how to weaponize or specifically abuse them.

STIF - SDLC Infrastructure Threat Framework

Component: All Components Endpoint/IDE VCS CI/CD Registry **Production/Cloud** Stage: All Stages Initial Access **Discovery and Lateral Movement** Post-Compromise Search techniques...

11 Techniques 1 Components 1 Attack Stages 46 Risks 60 Controls

Production/Cloud
Production infrastructure of the producer organization

Discovery and Lateral Movement 11

T-P002
Malicious Deployment via Compromised Pipeline
Attacker modifies CI/CD pipeline to deploy malicious code to production

RISKS

- No deployment approval gates
- Overprivileged deployment credentials
- No code signing verification
- Insufficient workflow permissions review

PROTECTIVE CONTROLS

- Deployment approval gates **SPVS**
- Code signing and verification **SPVS**
- Minimal workflow permissions **SPVS**
- Branch protection rules (VCS) **SPVS**

DETECTIVE CONTROLS

- Deployment monitoring **SPVS**

T-P003
Pivot from Self-Hosted Runner to Production Network
Attacker uses self-hosted runner network access to pivot into production

RISKS

- Self-hosted runner in production network
- Weak network segmentation
- Overprivileged runner identity
- No network monitoring

PROTECTIVE CONTROLS

- Network segmentation from SDLC **SPVS**
- Using managed runners only **SPVS**
- Runner isolation **SPVS**
- Minimal runner privileges **SPVS**

DETECTIVE CONTROLS

- Network monitoring **SPVS**

T-P004
Container Image Poisoning to Production
Attacker publishes malicious container image that gets deployed to production

RISKS

- Lack of image signing/verification
- No provenance attestation
- Automatic deployment without review
- No image scanning

PROTECTIVE CONTROLS

- Image signing and verification **SPVS**
- Provenance attestation **SPVS**
- Deployment approval gates **SPVS**

DETECTIVE CONTROLS

- Image vulnerability scanning **SPVS**
- Runtime security monitoring **SPVS**

T-P005
Infrastructure-as-Code Manipulation
Attacker modifies IaC to create backdoors in production infrastructure

RISKS

- Insufficient IaC review process
- Lack of drift detection
- Overprivileged deployment credentials

PROTECTIVE CONTROLS

- Code review for IaC changes **SPVS**
- Deployment approval gates **SPVS**
- Minimal deployment permissions **SPVS**
- CSPM/CNAPP **SPVS**

DETECTIVE CONTROLS

- Drift detection **SPVS**

T-P006
Kubernetes Service Account Token Theft
Attacker extracts Kubernetes service account tokens from CI/CD

RISKS

- Overprivileged service accounts
- Long-lived tokens

T-P007
Cloud Metadata Service Exploitation
Attacker accesses cloud metadata service to steal additional credentials

RISKS

- Metadata service not restricted
- Overprivileged IAM roles

T-P008
Kubernetes Pod Escape
Attacker escapes from container to underlying node

RISKS

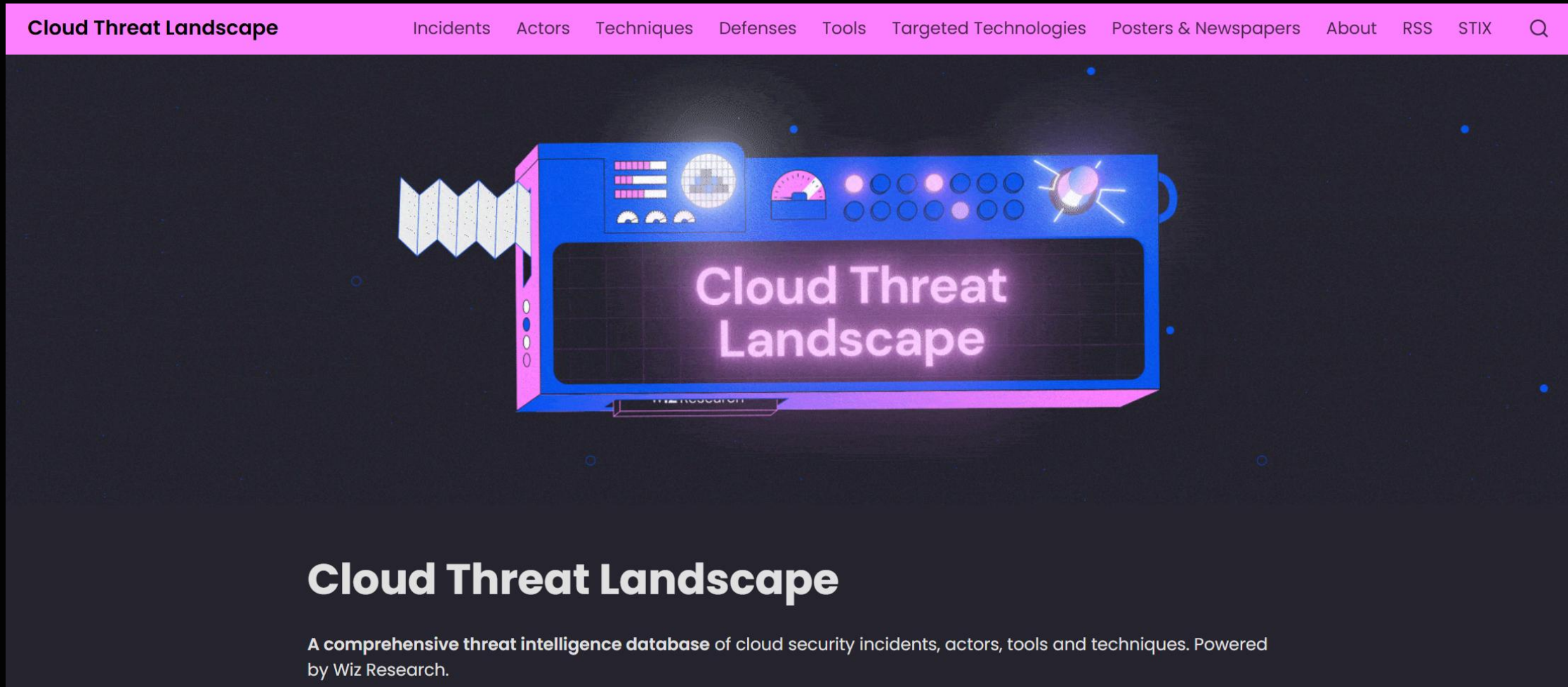
- Privileged containers allowed
- No pod security policies
- Overprivileged pod identity

T-P009
Cloud Service Lateral Movement
Attacker uses compromised cloud credentials to access other cloud services

RISKS

- Overprivileged IAM roles
- No network segmentation

Cloud Threat Landscape



Cloud Threat Landscape

Incidents Actors Techniques Defenses Tools Targeted Technologies Posters & Newspapers About RSS STIX Q

Cloud Threat Landscape

A comprehensive threat intelligence database of cloud security incidents, actors, tools and techniques. Powered by Wiz Research.

<https://threats.wiz.io/>



Мир глазами защитников

Раздел документации Kubernetes по безопасности

The screenshot shows the Kubernetes documentation page for the Security Checklist. The page is titled "Security Checklist" and is part of the "Concepts / Security" section. It provides a baseline checklist for ensuring security in Kubernetes clusters. The page includes a navigation sidebar on the left with a search bar and a list of security-related topics. The main content area contains the title, a brief description, a list of items to read, a caution section, and a section on Authentication & Authorization. The right sidebar contains links for editing the page, creating a child page, creating an issue, and printing the section, as well as a list of related topics.

Navigation: Documentation, Kubernetes Blog, Training, Careers, Partners, Community, Versions, English, Search this site

Search: Search this site

Security Checklist

Baseline checklist for ensuring security in Kubernetes clusters.

This checklist aims at providing a basic list of guidance with links to more comprehensive documentation on each topic. It does not claim to be exhaustive and is meant to evolve.

On how to read and use this document:

- The order of topics does not reflect an order of priority.
- Some checklist items are detailed in the paragraph below the list of each section.

Caution:

Checklists are **not** sufficient for attaining a good security posture on their own. A good security posture requires constant attention and improvement, but a checklist can be the first step on the never-ending journey towards security preparedness. Some of the recommendations in this checklist may be too restrictive or too lax for your specific security needs. Since Kubernetes security is not "one size fits all", each category of checklist items should be evaluated on its merits.

Authentication & Authorization

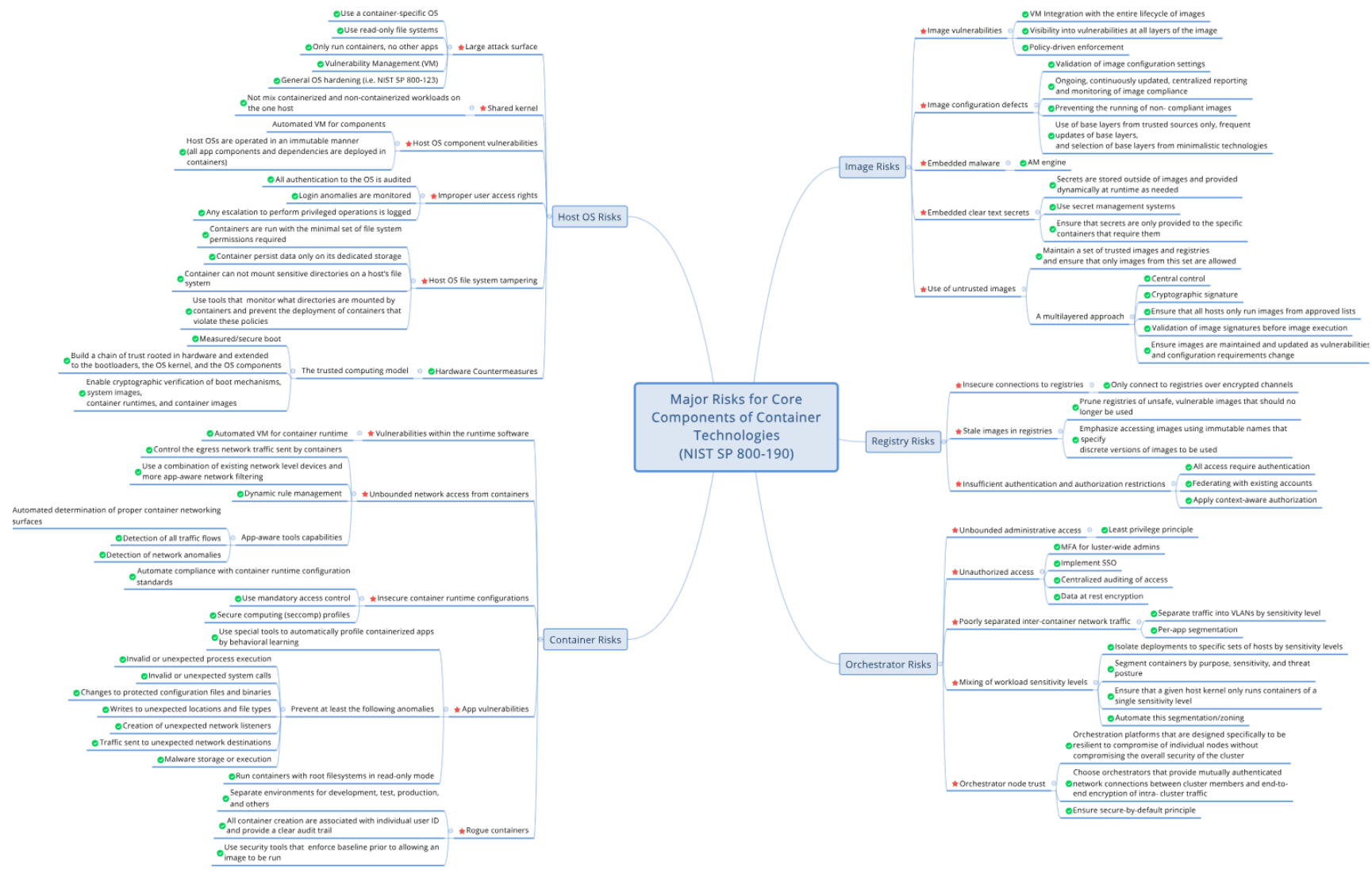
- `system:masters` group is not used for user or component authentication after bootstrapping.
- The kube-controller-manager is running with `--use-service-account-credentials` enabled.
- The root certificate is protected (either an offline CA, or a managed online CA with effective access controls).
- Intermediate and leaf certificates have an expiry date no more than 3 years in the future.
- A process exists for periodic access review, and reviews occur no more than 24 months apart.
- The [Role Based Access Control Good Practices](#) are followed for guidance related to authentication and authorization.

After bootstrapping, neither users nor components should authenticate to the Kubernetes API as `system:masters`. Similarly, running all of kube-controller-manager as `system:masters` should be avoided. In fact, `system:masters` should only be used as a break-glass mechanism, as opposed to an admin user.

Navigation (Left Sidebar): Security, Cloud Native Security, Pod Security Standards, Pod Security Admission, Service Accounts, Pod Security Policies, Security For Linux Nodes, Security For Windows Nodes, Controlling Access to the Kubernetes API, Role Based Access Control Good Practices, Good practices for Kubernetes Secrets, Multi-tenancy, Hardening Guide - Authentication Mechanisms, Hardening Guide - Scheduler Configuration, Kubernetes API Server Bypass Risks, Linux kernel security constraints for Pods and containers, Security Checklist, Application Security Checklist, Policies, Scheduling, Preemption and

Navigation (Right Sidebar): Edit this page, Create child page, Create an issue, Print entire section, Authentication & Authorization, Network security, Pod security, Enabling Seccomp, Enabling AppArmor or SELinux, Logs and auditing, Pod placement, Secrets, Images, Admission controllers, What's next

NIST SP 800-190 Application Container Security Guide



Карта безопасности для контейнерных сред

- Версия от Luntry, с учетом NIST SP 800-190 и учетом специфики Kubernetes

- Риски
- Механизмы защиты

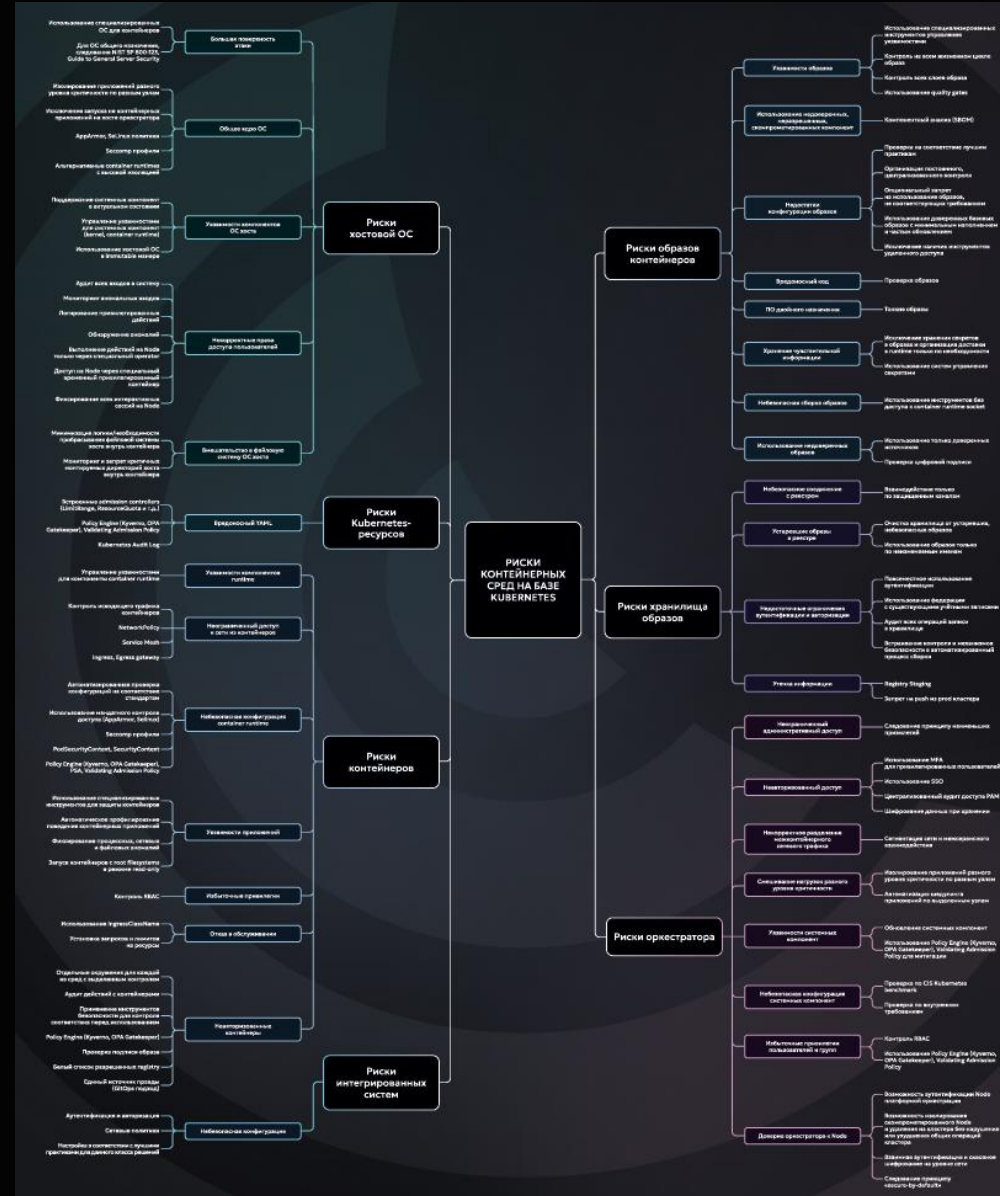
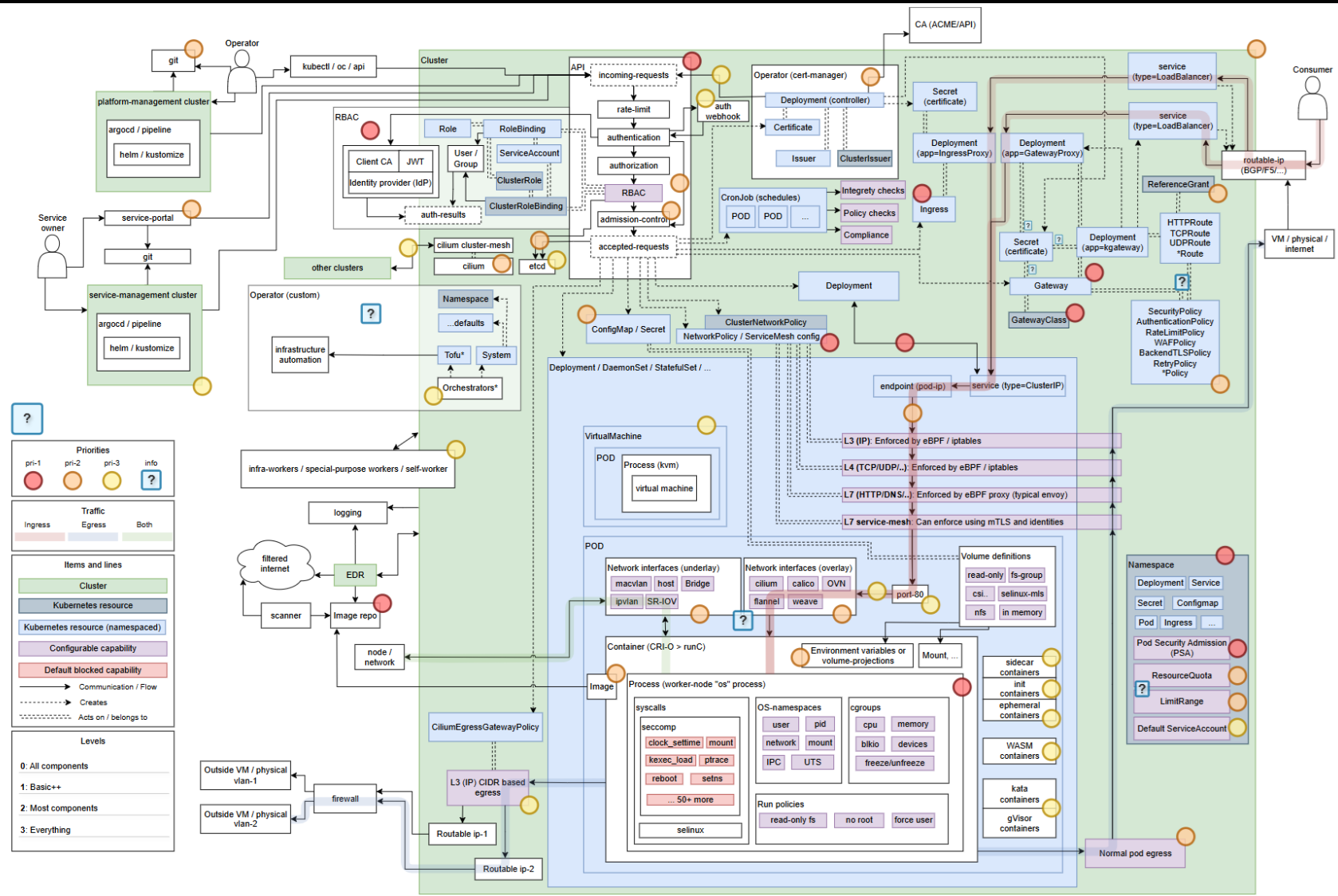


Диаграмма безопасности Kubernetes



OWASP и Kubernetes

The screenshot shows the OWASP Cheat Sheet Series website. The main content area is titled "Kubernetes Security Cheat Sheet" and includes an "Overview" section. The overview text states: "This cheat sheet provides a starting point for securing a Kubernetes cluster. It is divided into the following categories:" followed by a bulleted list of categories: "Receive Alerts for Kubernetes Updates", "INTRODUCTION: What is Kubernetes?", "Securing Kubernetes hosts", "Securing Kubernetes components", "Using the Kubernetes dashboard", "Kubernetes Security Best Practices: Build Phase", "Kubernetes Security Best Practices: Deploy Phase", and "Kubernetes Security Best Practices: Runtime Phase". Below the list, it says "For more information about Kubernetes, refer to the Appendix." There is also a section titled "Receive Alerts for Security Updates and Reporting Vulnerabilities" with a link to join the kubernetes-announce group. A table of contents is visible on the right side of the page.

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

The screenshot shows the OWASP website's "OWASP Kubernetes Top Ten" page. The page features a "Main" button and a section titled "About the Kubernetes Top 10". The text explains that when adopting Kubernetes, new risks are introduced, and the OWASP Kubernetes Top 10 is a prioritized list of these risks. Below this, there is a section titled "Top 10 Kubernetes Risks - 2025" with a note that the 2025 Top 10 Risks are now available and feedback is welcome. A bulleted list of the 10 risks is provided:

- [K01: Insecure Workload Configurations](#)
- [K02: Overly Permissive Authorization Configurations](#)
- [K03: Secrets Management Failures](#)
- [K04: Lack Of Cluster Level Policy Enforcement](#)
- [K05: Missing Network Segmentation Controls](#)
- [K06: Overly Exposed Kubernetes Components](#)
- [K07: Misconfigured And Vulnerable Cluster Components](#)
- [K08: Cluster To Cloud Lateral Movement](#)
- [K09: Broken Authentication Mechanisms](#)
- [K10: Inadequate Logging And Monitoring](#)

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

PCI DSS

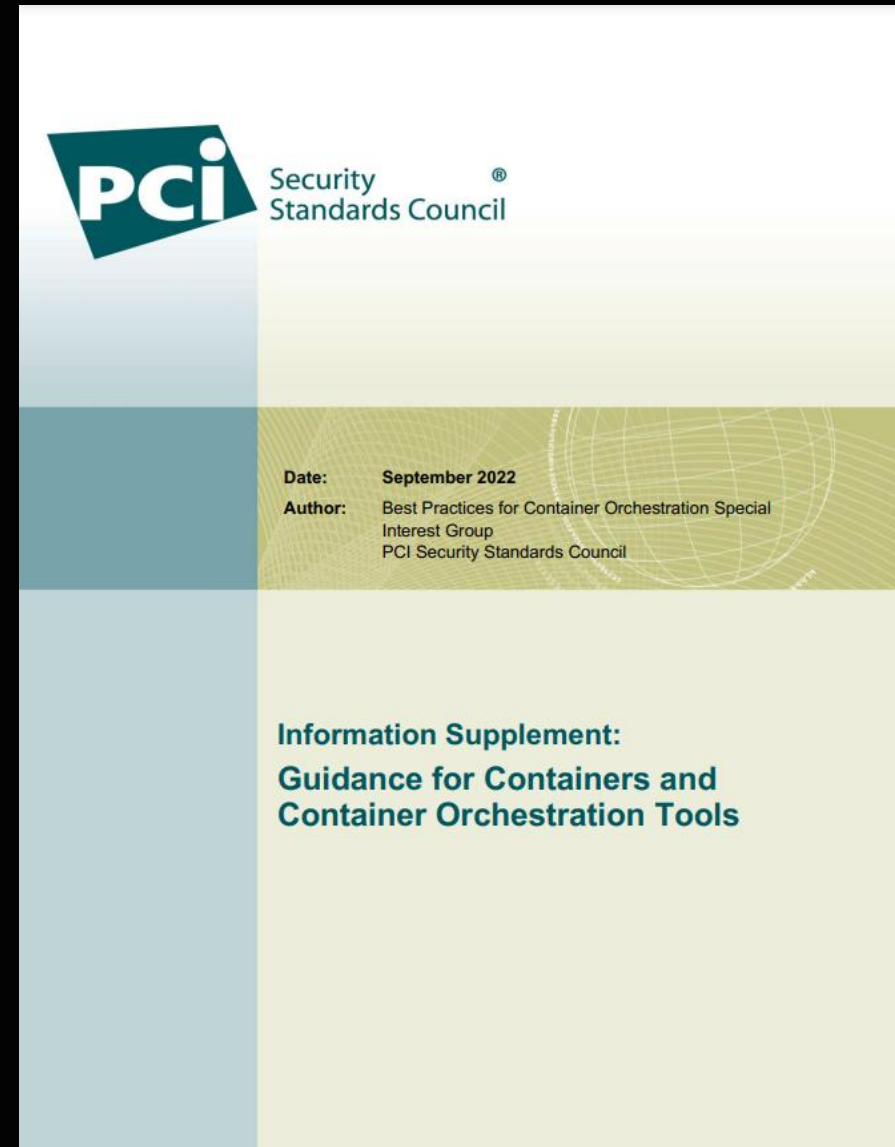
- НСПК требует соответствия PCI DSS
- Существует руководство по безопасности для контейнеров и оркестраторов контейнеров

Стандарт PCI DSS

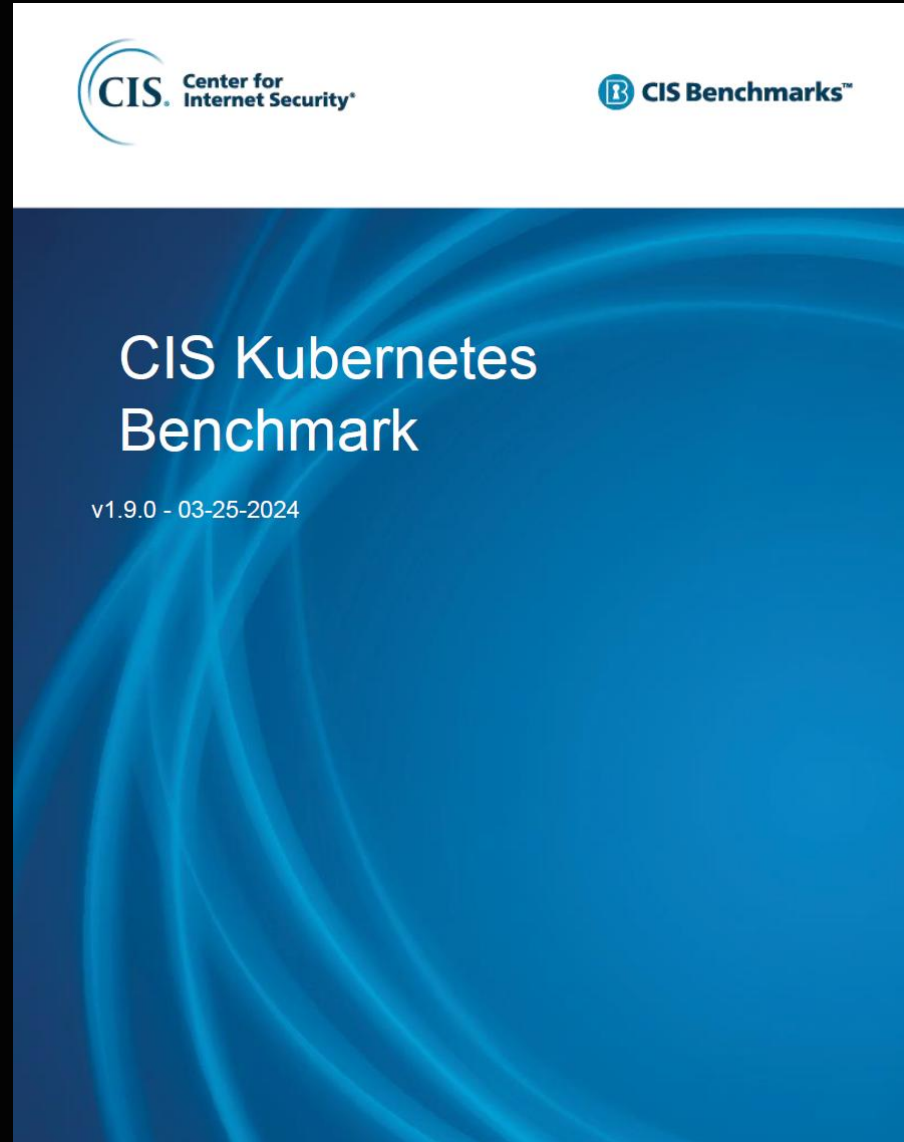
В ПС «Мир» для обеспечения безопасности данных карт «Мир» используется международный индустриальный стандарт PCI Data Security Standard (PCI DSS)

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.

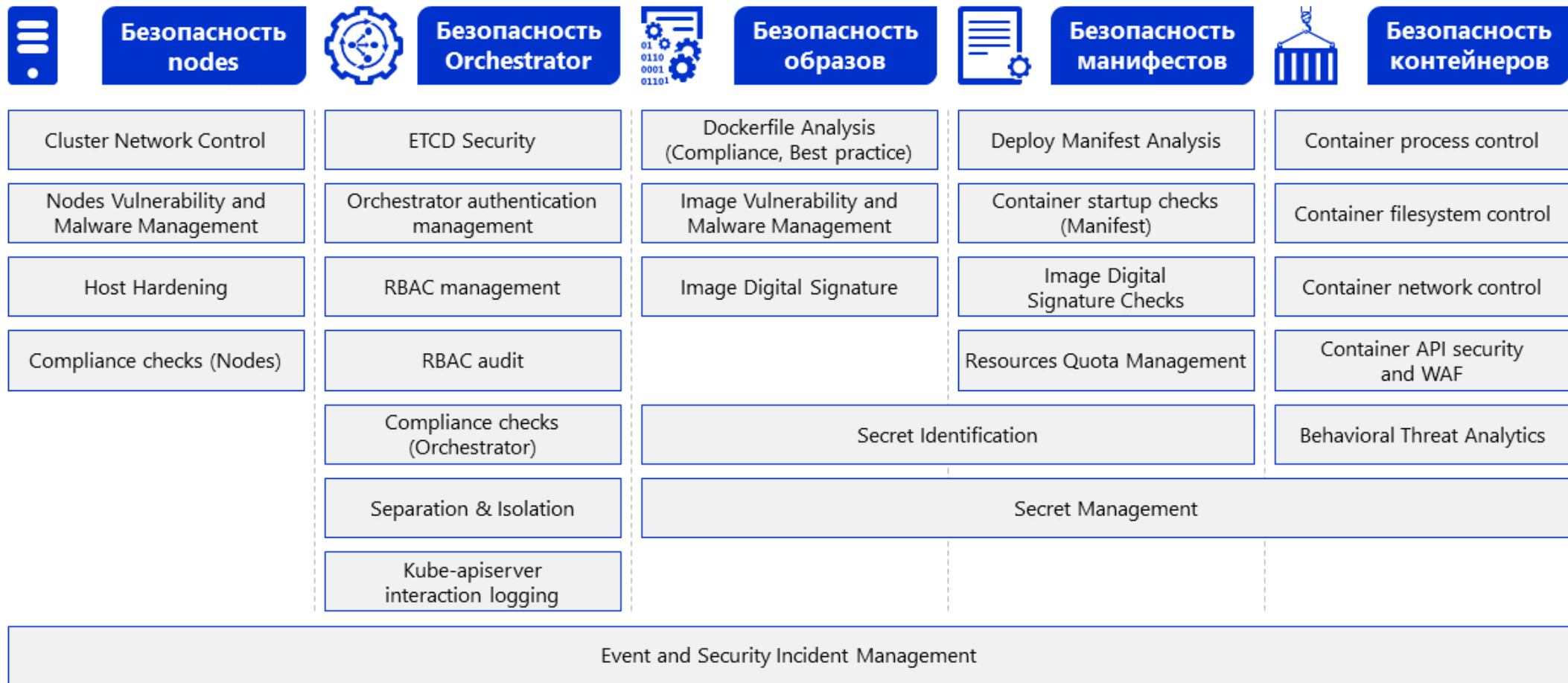


CIS Kubernetes Benchmark



<https://www.cisecurity.org/benchmark/kubernetes>

Jet Container Security Framework (JCSF)

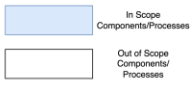


Регуляторика в РФ

- 118 приказ ФСТЭК
 - Требования по безопасности информации к средствам контейнеризации
- 117 приказ ФСТЭК
 - Пункт про защиту технологий контейнерных сред и их оркестрации
- Информационное сообщение ФСТЭК России 240/24/38
 - О повышении безопасности средств защиты информации, в состав которых разработчики включают средства контейнеризации или образы контейнеров
- Банк данных угроз безопасности информации от ФСТЭК
 - 5 УБИ посвящено контейнерам

Kubernetes для SOC

Threat Actors (TA)	
ID	Description
TA01	External attacker.
TA02	Internal attacker.

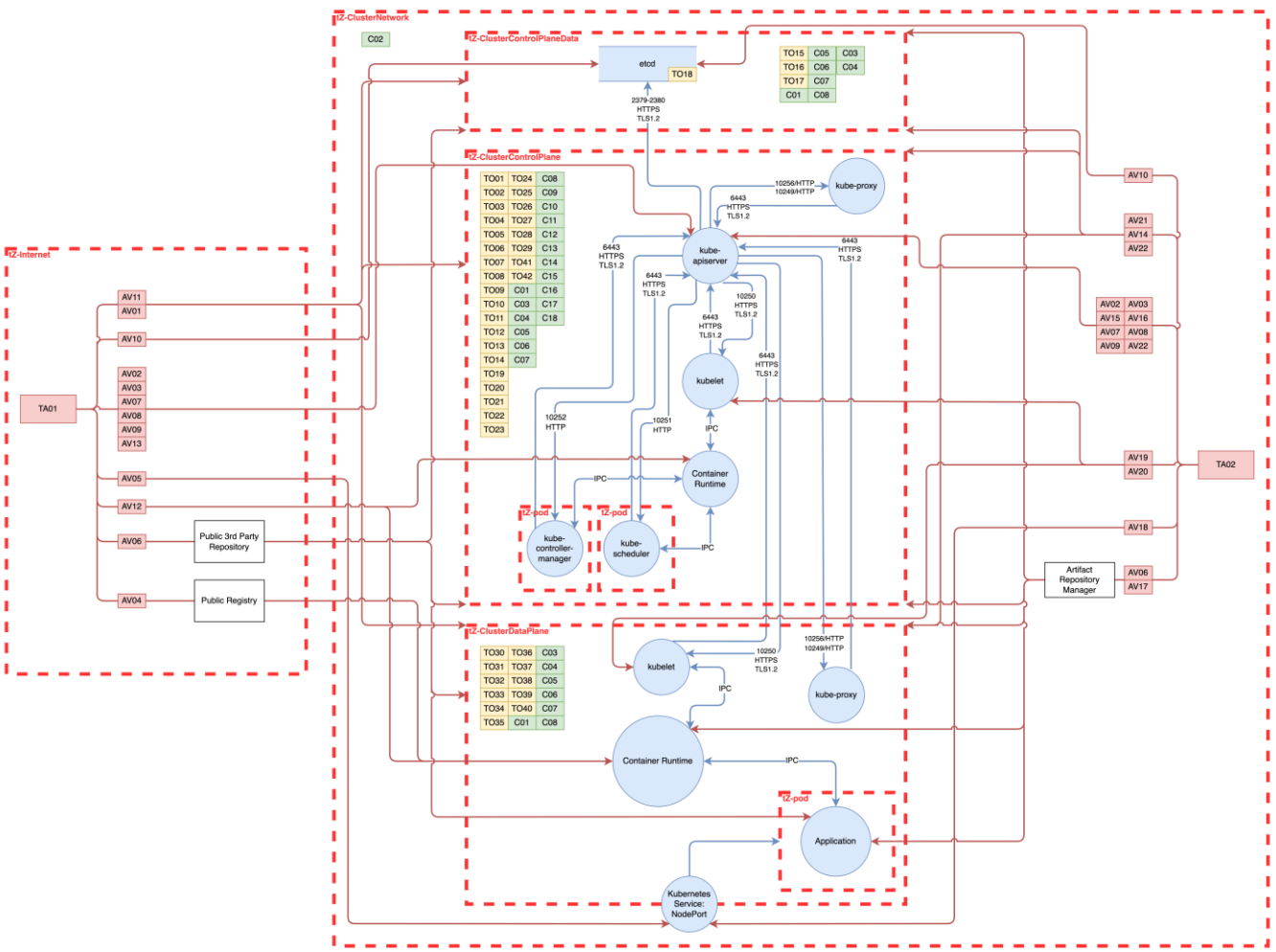


Threat Objects (TO)	
ID	Description
TO01	Cluster's CA x509 certkey pair.
TO02	kube-apiserver's server x509 certkey pair.
TO03	kube-apiserver's etcd-and-kubelet client x509 certkey pairs.
TO04	Cluster's service accounts signing x509 certkey pair.
TO05	kubernetes-admin's kubeconfig file: client x509 certkey pair embedded.
TO06	kube-scheduler's kubeconfig file: client x509 certkey pair embedded.
TO07	kube-controller-manager's kubeconfig file: client x509 certkey pair embedded.
TO08	kube-apiserver's static pod manifest file.
TO09	kube-controller-manager's static pod manifest file.
TO10	kube-scheduler's static pod manifest file.
TO11	Cluster's audit log directory.
TO12	Cluster's audit policy file.
TO13	Cluster's encryption provider config file.
TO14	x509 certificate for the CA that signed OIDC's web certificate.
TO15	etcd's CA x509 certkey pair.
TO16	etcd's server x509 certkey pair.
TO17	etcd's data directory.
TO18	Cluster's desired state.
TO19	Master node's etcd's CA x509 cert.
TO20	Master node's kubelet's kubeconfig file: client x509 certkey pair embedded.
TO21	Master node's kubelet's service file.
TO22	Master node's kubelet's binary.
TO23	Master node's kubelet's config file.
TO24	Master node's kubelet's directory.
TO25	Master node's kube-proxy's kubeconfig file: client x509 certkey pair embedded.
TO26	Master node's kube-proxy's service file.
TO27	Master node's kube-proxy's binary.
TO28	Master node's kube-proxy's config file.
TO29	Master node's CNI directories.
TO30	Worker node's kubelet's kubeconfig file: client x509 certkey pair embedded.
TO31	Worker node's kubelet's service file.
TO32	Worker node's kubelet's binary.
TO33	Worker node's kubelet's config file.
TO34	Worker node's kubelet's directory.
TO35	Worker node's cluster's CA x509 cert.
TO36	Worker node's kube-proxy's kubeconfig file: client x509 certkey pair embedded.
TO37	Worker node's kube-proxy's service file.
TO38	Worker node's kube-proxy's binary.
TO39	Worker node's kube-proxy's config file.
TO40	Worker node's CNI directories.
TO41	Cluster's admission control config file.
TO42	Cluster's authorization webhook config file.

Attack Vectors (AV)	
ID	Description
AV01	Publicly exposed host with vulnerable non-K8s endpoint(s)/service(s)/port(s).
AV02	Compromised kubeconfig file.
AV03	Compromised K8s service account token with cluster details.
AV04	Malicious container image in public registry.
AV05	Publicly exposed vulnerable K8s workload: e.g. RCE.
AV06	Malicious 3rd party dependency.
AV07	kube-apiserver vulnerability.
AV08	kube-apiserver's options configured incorrectly.
AV09	kube-apiserver's authorization configured incorrectly.
AV10	etcd's authentication configured incorrectly.
AV11	Compromised credentials of publicly exposed host.
AV12	Container runtime configured incorrectly.
AV13	Compromised OIDC token with cluster details.
AV14	Host with vulnerable non-K8s endpoint(s)/service(s)/port(s).
AV15	Legitimately obtained kubeconfig file.
AV16	Legitimately obtained K8s service account token with cluster details.
AV17	Malicious container image in private registry.
AV18	Vulnerable K8s workload: e.g. RCE.
AV19	kubelet vulnerability.
AV20	kubelet's options configured incorrectly.
AV21	Compromised credentials of host.
AV22	Legitimately obtained OIDC token with cluster details.
AV23	Legitimately obtained credentials of host.

Assumptions (AS)	
ID	Description
AS01	Fault-tolerance and high availability are not in scope for this model.
AS02	Two certificate authorities (CAs) are in scope for this model: one for the cluster and one for etcd.
AS03	Within this model, the kube-proxy is running as a service in the host and using a config file.
AS04	Container Runtime installation and related files are out of scope for this model.
AS05	Within this model, there are no business-type K8s workloads scheduled in the master node.
AS06	Within this model, K8s admission controllers include: ResourceQuota, LimitRanger, PodSecurity(Policy), and ImagePolicyWebhook.
AS07	Within this model, K8s authorisation is limited to: RBAC, Node, and Webhook.
AS08	Within this model, K8s encryption is implemented via EncryptionConfiguration and applies for data at rest only.
AS09	Within this model, K8s MutatingAdmissionController enables: disable service account token auto-mount, PodSecurityContext/ContainerSecurityContext, AppArmor, container sandboxing, and application logging consumption.
AS10	Within this model, K8s authentication is limited to: client certificates, service account tokens, and OIDC tokens.
AS11	AV02, AV03, AV07, AV08, AV09, AV10, AV12, and AV13 are being taken into account under the assumption that the hosts holding the cluster control plane's components are publicly exposing them.

Controls	
ID	Description
C01	C-Harden-Component
C02	C-Control-Traffic-Flow
C03	C-Harden-Config-Compl-Mon / C-Mon-Drift
C04	C-Evt-Log-App / C-Evt-Log-Sec / C-Evt-Log-Sys
C05	C-Mon-File-Integrity
C06	C-Mon-Log
C07	C-Mon-Health
C08	C-Vuln-Scan
C09	C-K8s-Harden-Component
C10	C-K8s-AdmissionControllers
C11	C-K8s-AuditLogs
C12	C-K8s-Authorisation
C13	C-K8s-CNI-NetworkPolicies
C14	C-K8s-MutatingAdmissionController
C15	C-K8s-ValidationAdmissionController
C16	C-K8s-Encryption
C17	C-K8s-ContainerRuntimeSecurity
C18	C-K8s-Authentication



CNCF Security Technical Advisory Group

Security Technical Advisory Group



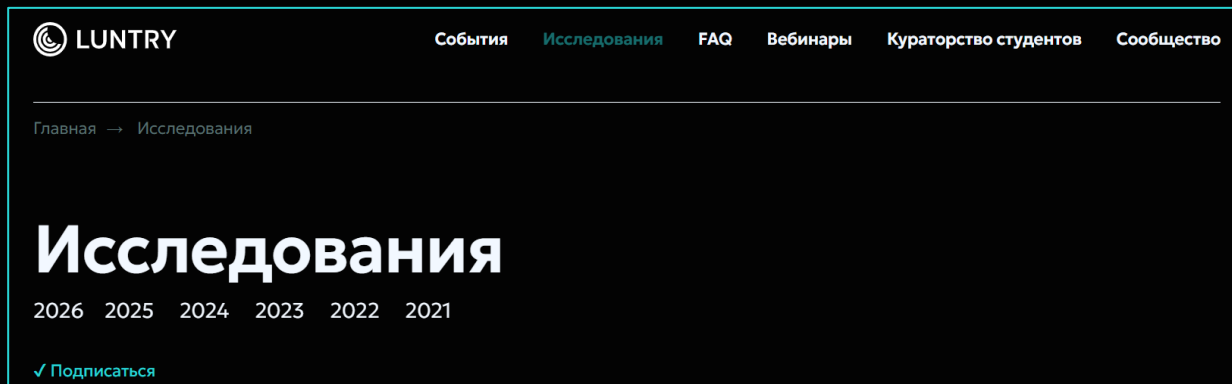
**CLOUD
NATIVE
SECURITY**



Полезности

Материалы Luntry

• Исследования



The screenshot shows the top navigation bar of the Luntry website with links for 'События', 'Исследования', 'FAQ', 'Вебинары', 'Кураторство студентов', and 'Сообщество'. Below the navigation, there is a breadcrumb trail 'Главная → Исследования'. The main heading is 'Исследования' with a sub-heading showing years from 2026 to 2021. A 'Подписаться' (Subscribe) button is visible at the bottom left.

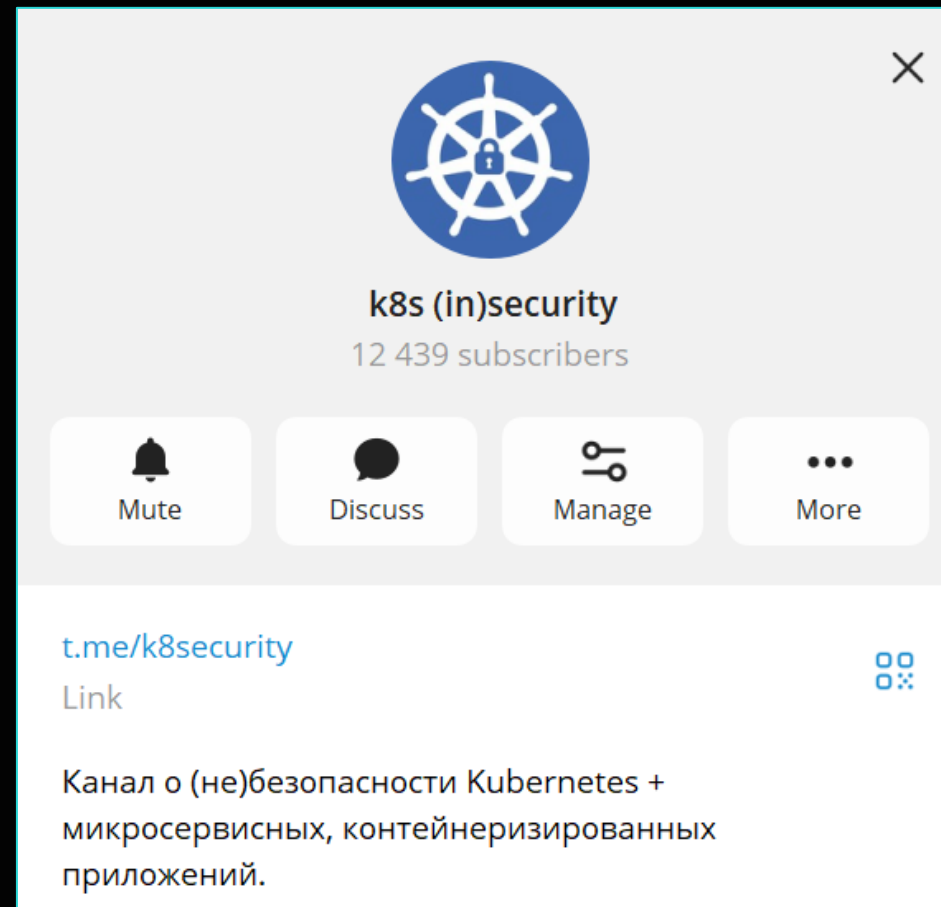
• Кураторство студентов

Кураторство и помощь в научно-исследовательской деятельности от Luntry

Команда Luntry специализируется на безопасности контейнеров и Kubernetes. Мы занимаемся разработкой, настройкой cloud-native окружений, атакой и защитой информационных систем на базе контейнеров. За это время наши специалисты накопили опыт и знания, которыми теперь готовы делиться с теми, кому интересна сфера!

Если вы еще учитесь, но уже хотите работать с актуальными задачами по контейнеризации для своей курсовой, диплома, магистерской или докторской работы или же делаете исследование в этой сфере — мы можем вам помочь.

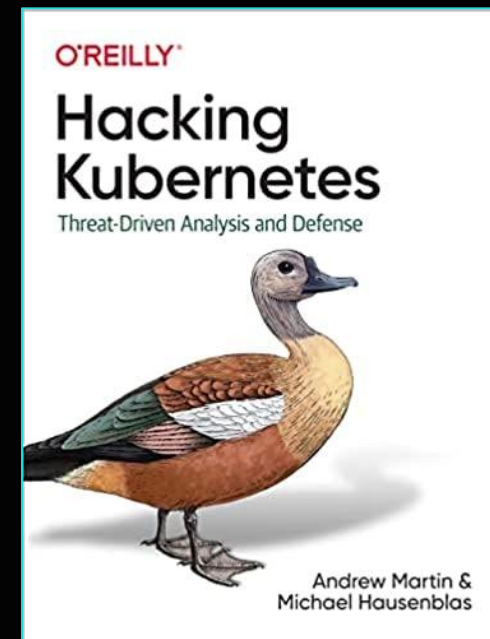
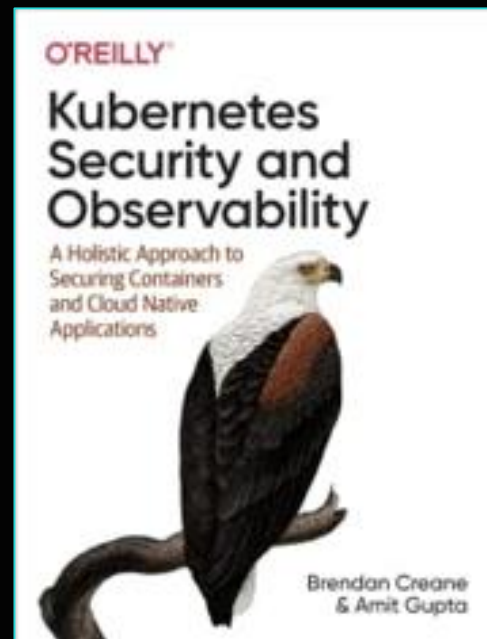
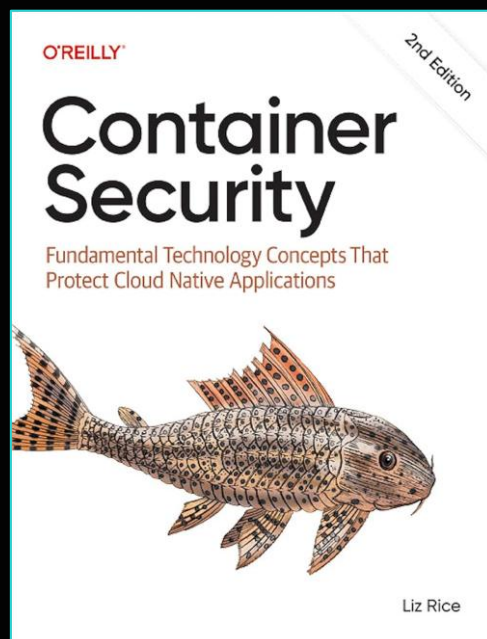
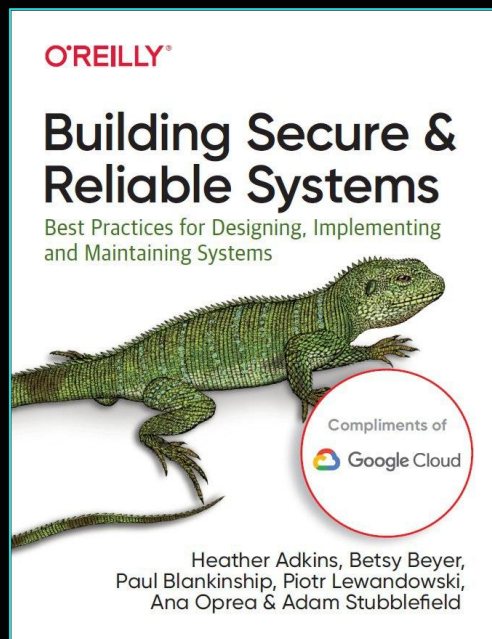
• Telegram канал



The screenshot shows the Telegram channel page for 'k8s (in)security', which has 12,439 subscribers. The channel profile picture is a blue circle containing a white ship's wheel with a padlock in the center. Below the profile information, there are four interactive buttons: 'Mute', 'Discuss', 'Manage', and 'More'. At the bottom, there is a link 't.me/k8security' and a description: 'Канал о (не)безопасности Kubernetes + микросервисных, контейнеризированных приложений.'

Полезные книги

- “Building Secure & Reliable Systems”
- “Container Security”
- “Kubernetes Security and Observability”
- “Hacking Kubernetes: Threat-Driven Analysis and Defense”



Полезные конференции

- BeKon
- Fwd:cloudsec
- DefCon Cloud Village
- Cloud Native Security Day
- KubeCon и CloudNativeCon



BEKON

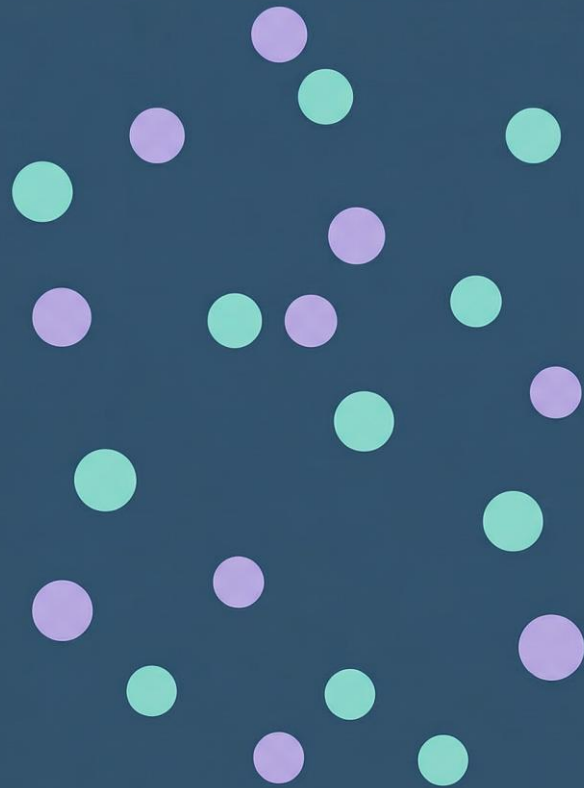


Практика и лабы

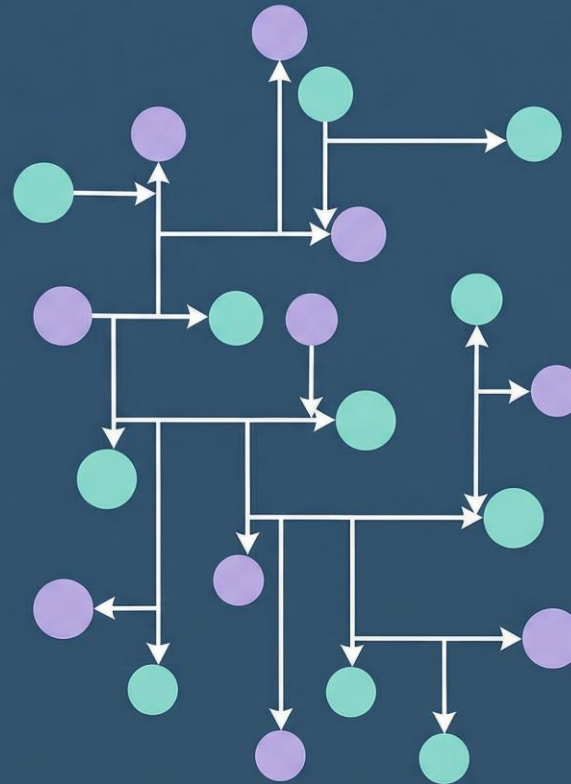
- [Docker Security: A Practical Guide](#)
- [Damn Vulnerable Kubernetes Application \(DVKA\)](#)
- [Docker Escape Labs](#)
- [K8s Best Security Practices](#)
- [Kubernetes Goat](#)
- [K8sQuest](#)
- [Breaching Bare Metal Kubernetes Clusters](#)
- [CKS](#)
- [Kubernetes LAN Party](#)
- [The Cloud Hunting Games CTF](#)
- [Awesome Cloud Security Labs](#)

Все приходит с опытом ;)

Knowledge



Experience





Выводы

Выводы

COUNTRY

01

В современном мире без контейнеров и Kubernetes уже никуда

02

Данная тема большая, серьезная и требует отдельного внимания

03

Данная тема имеет достаточно много специфики

04

На сегодняшний день это уже мейнстрим и доступно много информации, но тема быстро развивается



Дмитрий
ЕВДОКИМОВ
Founder & CTO Luntry



НА СВЯЗИ

✉ de@luntry.ru

📍 @Qu3b3c

📍 @k8security

📍 @luntry_official