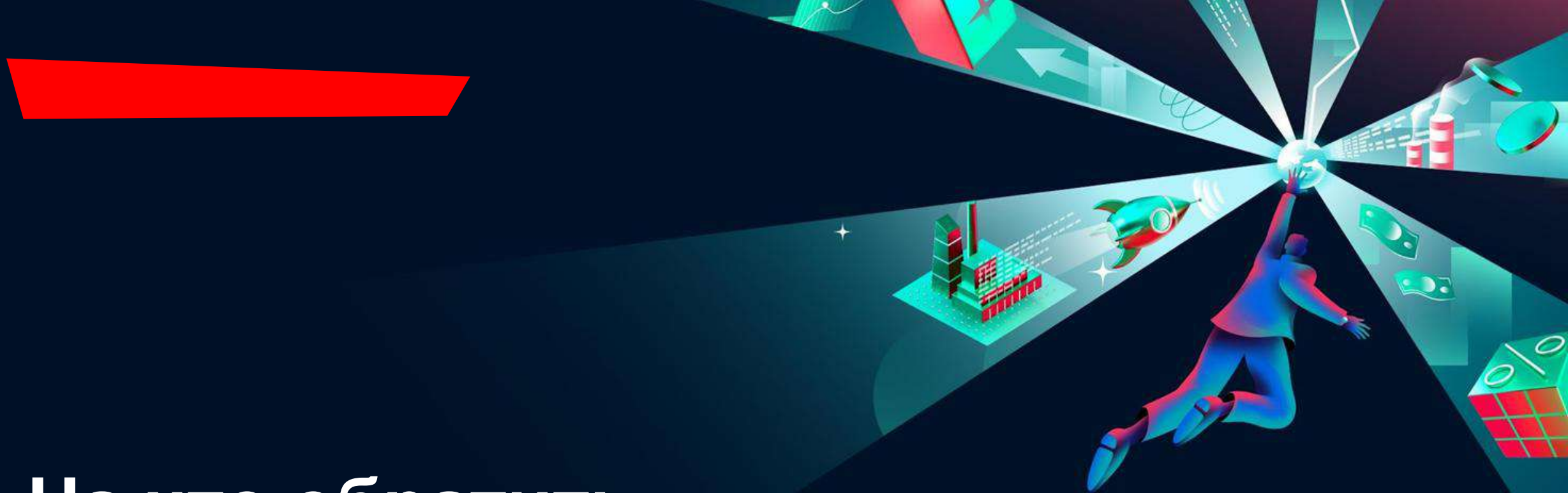
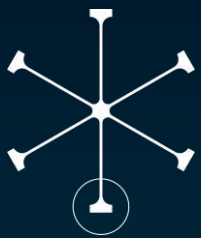


На что обратить
внимание при
комплексном подходе
к безопасной разработке.
От кода до эксплуатации





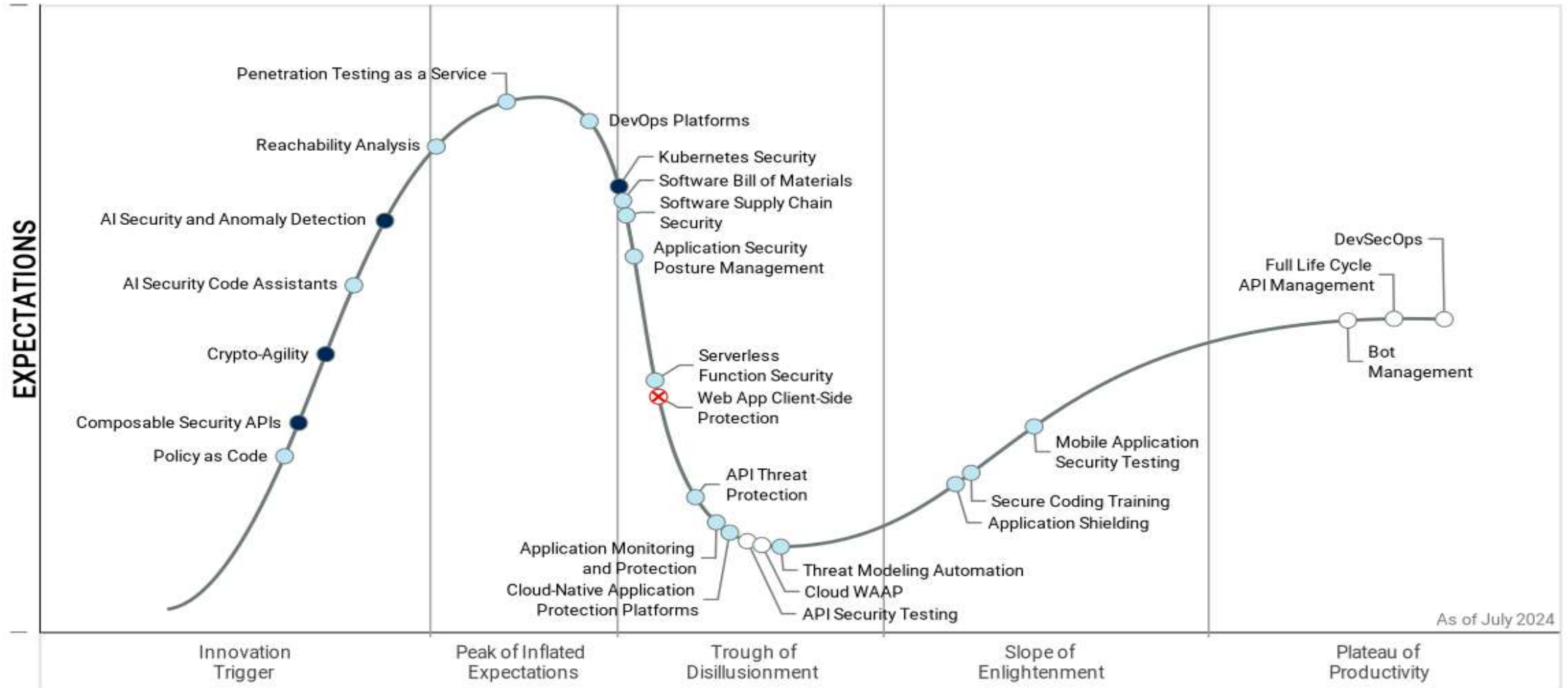
Что мы знаем о спикере?

Иван Соломатин

Коммерческий директор
CodeScoring



Hype Cycle for Application Security, 2024



As of July 2024

Plateau will be reached: ○ <2 yrs. ● 2-5 yrs. ● 5-10 yrs. ▲ >10 yrs. ⊗ Obsolete before plateau

AppSec Hype Diving

Good news first, the percentage of apps passing the OWASP Top 10 has increased 63% in 5 years (from 32% to 52%)

Now the bad news... the percentage of apps with high severity flaws has increased by 181%...

...and the average number of days to fix flaws has increased 47%.

Half of organizations have critical security debt (high severity, high exploitability)...

...and 70% of it comes from third party code and the software supply chain.

Category	Third party code	First party code
ALL SECURITY DEBT	11%	89%
CRITICAL SECURITY DEBT	70%	30%

1,658 projects scanned by Black Duck audits

97% of the codebases contained open source

70% of codebases had no original open source

On average 911 OSS components were found per application

64% of OSS components were transitioned to open source

The number of open source files in an average application has tripled in the last four years

Year	Files
2020	5,366
2022	11,858
2024	16,082

LEADING ORGANIZATIONS

- Below 43%
- Above 10% of flaws monthly
- Half of flaws in 5 weeks
- <17% of apps
- <15%

LAGGING ORGANIZATIONS

- 86% or more
- <1% of flaws monthly
- Half of flaws in over a year
- >67% of apps
- 100%

Veracode и BlackDuck, две компании, занимающиеся безопасностью ПО, одновременно выпустили свои отчеты:

- 2025 State of Software Security: A New View of Maturity
- 2025 Open Source Security and Risk Analysis Report.

#opensource #devsecops #безопаснаяразработка #appsec

```

Kubernetes Authentication
# Configure the kubectl CLI (sec540.com/1575)
# Configure access to an AWS cluster
$ aws eks update-kubeconfig --region us-west-2 --name aviata

# Configure access to an Azure cluster using the Entra ID auth plugin
$ az aks get-credentials -g acel35 -n aviata
$ kubelogin convert-kubeconfig -i azurecli

# Configure access to a Google cluster
$ gcloud container clusters get-credentials --region us-west2 --project acel35 aviata

# View Kubernetes authentication data including the user and group membership
$ kubectl auth whoami

# Check RBAC permissions to list the Kubernetes pods in the aviata namespace
$ kubectl auth can-i get pods -n aviata

Managing Kubernetes Resources
# List cluster resources
$ kubectl get nodes

# Describe an API resource and its fields
$ kubectl explain pods --recursive

# Creating or updating a resource
$ cat >> namespace.yml << EOF
apiVersion: v1
kind: Namespace
metadata:
  name: aviata
  annotations:
    acel35/owner: "aviata"
EOF
$ kubectl apply -f ./namespace.yml

# Viewing a resource in a namespace
$ kubectl describe pod -n aviata api

# Deleting a resource in a namespace
$ kubectl delete pod -n aviata api
    
```

Ассоциация "Финтех" выложила в открытый доступ материалы по безопасной разработке. Первый - матрица решений по DevSecOps, в которой присутствуют как коммерческие, так и open source компоненты с привязкой к типам и классам решаемых задач. Второй - блок-схема типичного процесса безопасной разработки (на картинке часть схемы) с отсылками на соответствующие ГОСТы ФСТЭК, требуемыми артефактами и используемым инструментарием.

#devsecops #безопаснаяразработка #appsec

А этот постер SANS про защиту Kubernetes.

#SANS #облака #devsecops

Немного общей статистики с полей

- В 2024 в РФ, в свободно-распространяемом ПО было выявлено на **30%** больше уязвимостей
- За последние 10 лет открытого кода с ошибками стало больше **в 4,5 раза**
- Лидирующие производители проприетарного ПО стали больше применять открытые компоненты в разработке
- В среднем, ПО без обновлений за один год накапливает **100** новых уязвимостей, из которых 10–15 являются эксплуатируемыми



Почему все ломается?



Ручные процессы: безопасность вне пайплайна гарантирует “бутылочное горлышко”



Чеклист безопасности в Confluence ≠ контроль на практике



Модель “безопасность в конце цикла” – поздно и дорого



Реагирование после релиза → уязвимости на проде, срочные фиксы, репутационные потери

Сырьевые и энергетические компании



- Разрабатывают большое количество разнородного ПО для автоматизации.
- Атаки могут привести к отказу критической инфраструктуры.
- Важно обеспечить проверку ПО собственной и сторонней разработки.

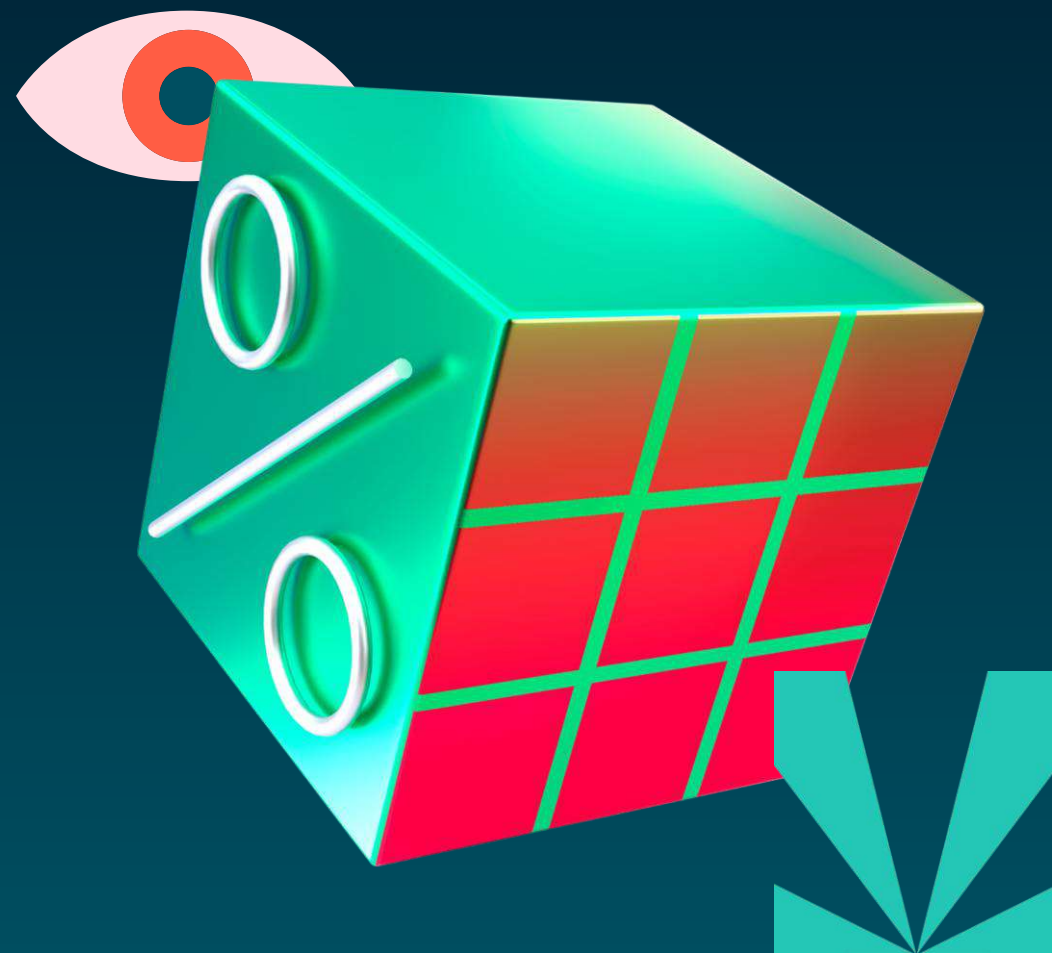
Финансовые организации



- Разрабатывают мобильные и веб-сервисы для физ. лиц и корпоративного сегмента.
- Атаки могут привести к прямым финансовым потерям.
- Важно внедрить культуру и контроль практики безопасной разработки.

Польза для безопасности

- ✦ Знание собственных и чужих продуктов
- ✦ Своевременная информированность об уязвимостях
- ✦ Автоматизация процесса защиты цепочки поставки
- ✦ Контроль за ранее выпущенными продуктами
- ✦ Понимание, что делать дальше



Проверки по этапам



Локальная среда

Pre-commit

OSA

SCA проверка в IDE
/ локальный агент

Конвейер (CI/CD)

Pre-build

Build

Post-build

OSA

Source
SCA

SCA
on build

Binary
SCA

Пост-релиз

Post-deploy

SBoM SCA

Не все ошибки критичны сразу – важно выбрать приоритет!

Делаем российскую* платформу
безопасной разработки **CodeScoring**
и помогаем защищать программные
продукты

- **5 лет** реализуем свой продукт на российском рынке
- **> 10 лет** анализируем исходные коды на качество и безопасность
- **> 50 экспертов** работает над развитием платформы
- Полностью соответствуем требованиям композиционного анализа, согласно **ГОСТ-56939-2024**
- Экспертиза компании отмечена ИСП РАН и ФСТЭК России

* Запись в едином реестре ПО №13008.

Миллионы строк, сотни и тысячи компонентов

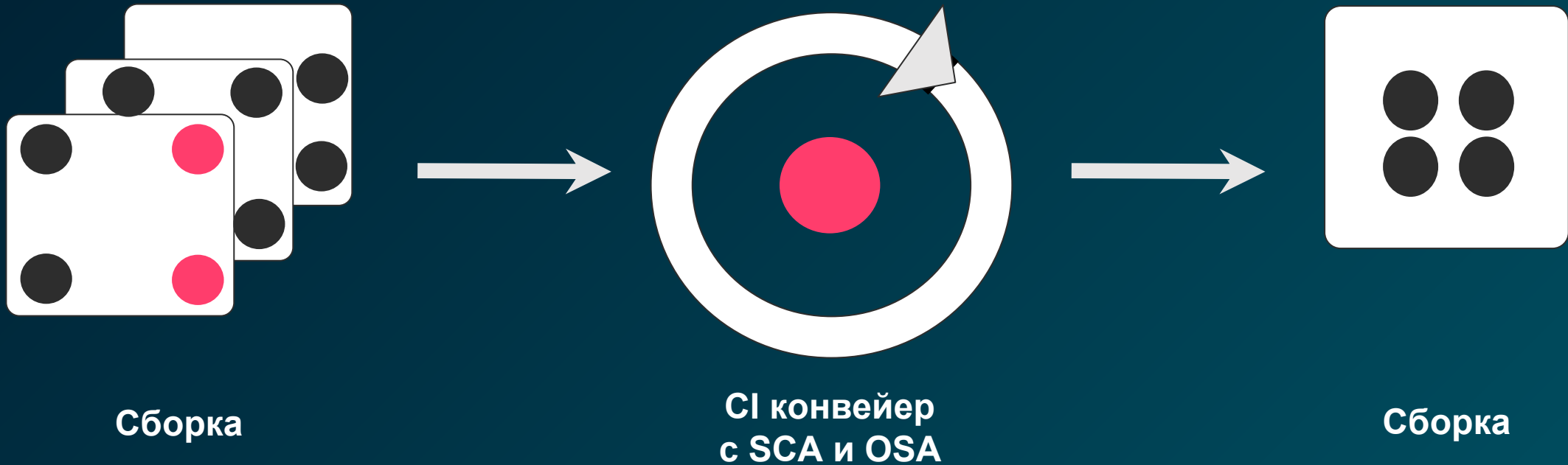


~80%
ЗАИМСТВОВАННЫХ
КОМПОНЕНТОВ
из открытых источников

~20%
собственная
разработка

CodeScoring проверяет наибольшую часть программного продукта на **наличие угроз**.

SCA и OSA: основа безопасной разработки



SCA – *Software Composition Analysis*
OSA – *Open Source Analysis*

Интеграция на всех этапах разработки



Варианты реализации CodeScoring

CodeScoring.OSA

Open Source Analysis

- защита от вредоносных компонентов, включая скомпрометированные пакеты
- политики предотвращения популярных атак на цепочку поставки
- система управления выявленными уязвимостями

CodeScoring.SCA

Software Composition Analysis

- проверка Open Source на всех этапах цикла разработки
- предоставление информации о найденных уязвимостях и лицензиях
- настройка политик безопасности по 20 критериям
- графы связей компонентов

Варианты реализации CodeScoring



CodeScoring.TQI

Teams & Quality Intelligence

- построение профилей участников разработки
- с подтвержденной компетенцией в проектах
- определение ключевых параметров технического долга
- функции для внутреннего рекрутинга
- отчетность и интеграция в SDLC

CodeScoring.Secrets

Поиск конфиденциальной информации

- управление конфигурациями инструментов поиска секретов (gitleaks)
- интерфейс разметки обнаруженных находок (TP/FP)
- привязка добавленного секрета к автору, который его добавил
- применение и дообучение ML-модели к результатам проверки

Интеграция в жизненный цикл

Менеджеры репозитория



Платформы разработки



CI/CD конвейер



Таск-менеджеры



ASPM/ASOC-системы



Среды разработки (Q2 2025)



Инфраструктура

Почтовый сервер / LDAP / API / вебхуки

Полезные материалы



Мифы и факты о цепочке поставки программного обеспечения, CyberCamp'23



Таксономия атак на цепочку поставки ПО: тренды и предпосылки новых трендов, GigaConf'24



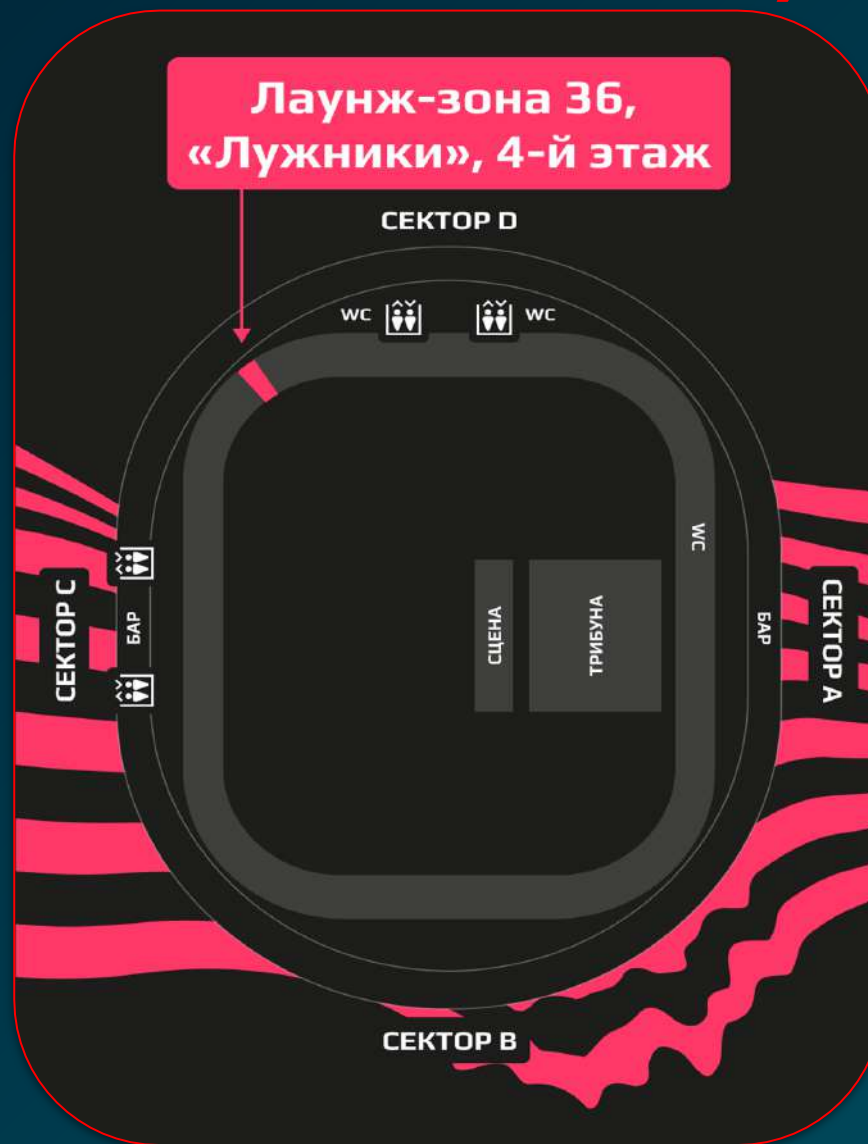
Проблема отцов и детей: аналитика и триаж транзитивных зависимостей, PHD'24



Построить SBoM, вырастить SDL-политики, воспитать культуру безопасной разработки, IT IS Conf'23



Как нас найти 22-24 мая ;)



Спасибо за внимание!

CODE
SCORING



Сайт / codescoring.ru

 Новости / [@codescoring](https://t.me/codescoring)

 Записи докладов / [@codescoring](https://www.youtube.com/@codescoring)

 Докладчик / [@soulmatin](https://t.me/soulmatin)

**POSITIVE
HACK DAYS
FEST**

от positive technologies