SOLAR | Hexway | LUNTRY

DevSecOps на ИИ-скорости

Как ускорить процесс безопасной разработки
в десятки раз и не подвергнуть компанию риску



АНТОН ПРОКОФЬЕВ

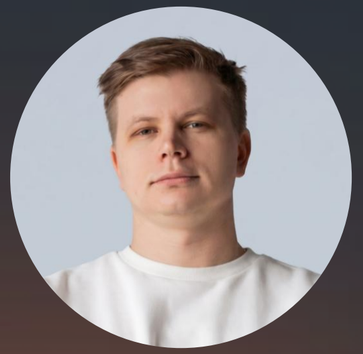


Директор центра разработки решений по контролю безопасности ПО

E: presale.appscreener@rt-solar.ru

- Архитектор домена AST (РБПО)
- Встраивание инструментов AST и построение процессов в крупных компаниях России и вне ее
- Разработка и развитие инструмента анализа Solar appScreener

ДМИТРИЙ ЕВДОКИМОВ



Основатель и технический директор Luntry

E: de@luntry.ru

КОМАНДА LUNTRY

Специализация: безопасность контейнеров и Kubernetes

Специалисты с опытом в ИБ более 15 лет

Команда базируется в Санкт-Петербурге

- Ведем тренинги по организации безопасности в Kubernetes
- Организуем уникальную конференцию по безопасности контейнеров — «БеКон»
- Занимаемся кураторством учащихся по тематике безопасности контейнеров
- Делимся с сообществом знаниями о безопасности Kubernetes в телеграм-канале k8s (in)security

Выступления на конференциях

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

ДМИТРИЙ ЧАСТУХИН



Основатель и технический директор Hexway

E: dc@hexway.io

- Создаю ASOC-платформу для безопасной разработки
- Выступаю с докладами и исследованиями на российских и зарубежных конференциях
- Люблю травмоопасные виды спорта

Вклад AI в разработку

РЕАЛИИ И ПРОГНОЗЫ

Рост количества уязвимостей



48 000

2025 год

прогноз

66 000–118 000

2026 год

Окно сокращается

от 63 дней

в 2018–2019 годах

до нескольких часов

в 2024–2025 годах

в 1000 раз

ИИ ускоряет разработку, но резко увеличивает количество ошибок и уязвимостей

Исследование Accelerate State of DevOps Report (Google) выявило парадоксальный эффект

на 7,2%

Снизилась стабильность поставок ПО после внедрения ИИ

×8

Рост дублирования кода

×10

Рост числа уязвимостей

AI, AI, AI

Vibe coding | ML analysis | MLSecOps



Использование ML для решения рутинных задач

- Верификация сработок, генерация исправлений для всех видов анализа (code fix / triage).
- Автоматический подбор и обновление зависимостей (dependency gardening).

SAST/DAST/SCA 2.0

Использование ML для полноценного анализа.

Пример:

- SCA нашел уязвимую библиотеку в логин-сервисе
- SAST проверил, что ваш код вызывает эту уязвимую функцию
- DAST динамически подтвердил, что в этот эндпойнт можно передать зловредный параметр

MLSecOps

ML в безопасной разработке = средство защиты само становится целью для атак. Обеспечение безопасности ML-системы — это отдельная и критически важная задача:

- защита цепочки поставок (Supply Chain Security)
- тестирование на устойчивость (Robustness Testing)
- мониторинг в рантайме (Runtime Monitoring)

Две задачи — один плагин от AppScreener

On-premise | 7 лет обучения | Создан при поддержке мирового AppSec-производителя

Triage

Интеллектуальный триаж SAST-результатов

Что делает:

Берет «сырые» результаты сканирования, автоматически разбирает каждую уязвимость по контексту и выдает чистый список только реальных угроз

Зачем это нужно:

1. Для AppSec: отсутствие ложных срабатываний, благодаря чему команды не тратят часы на их разбор
2. Для разработки: получают готовый приоритетный список благодаря триажу и сгенерированные исправления для подтвержденных уязвимостей (с помощью CodeFix)
3. Для всех команд: быстрее релизы — отчет готов за минуты, а не недели, выпуск вашего ПО ускоряется в разы. При этом сохраняется высокий уровень безопасности разрабатываемого ПО

CodeFix

Автоматическое исправление уязвимостей и готовый код

Что делает:

Плагин генерирует готовые, контекстные патчи кода

Зачем это нужно:

1. Качественный готовый код для исправления подтвержденных уязвимостей
2. В 10 раз увеличение продуктивности. Фокус команды смещается на создание, а не на исправления
3. Поддержка качества кода: исправления создаются по стандартам кодирования. Они сохраняют читабельность, производительность и безопасность
4. Обучение команд: каждое исправление содержит подробное объяснение того, почему и что было изменено

Универсальная или заточенная под конкретную задачу модель?

	Triage	Gigachat	ChatGPT	DeepSeek	LocalChat	LocalMistral
Общая точность	83,6%	53,2%	56,8%	66,8%	66,8%	67,3%
Прецизионность	85,5%	59,5%	78,1%	48,8%	82,9%	74,6%
Полнота	82,7%	27,4%	55,2%	35,3%	45,2%	56,8%
Процент ошибок	15,35%	46,8%	43,2%	33,2%	33,2%	32,7%

	CodeFix	Gigachat	ChatGPT	DeepSeek	LocalChat	LocalMistral
Точность	83,6%	19,3%	54,2%	45,1%	24,1%	30,2%
Self-точность	85,5%	9,2%	45,7%	-	57,4%	-

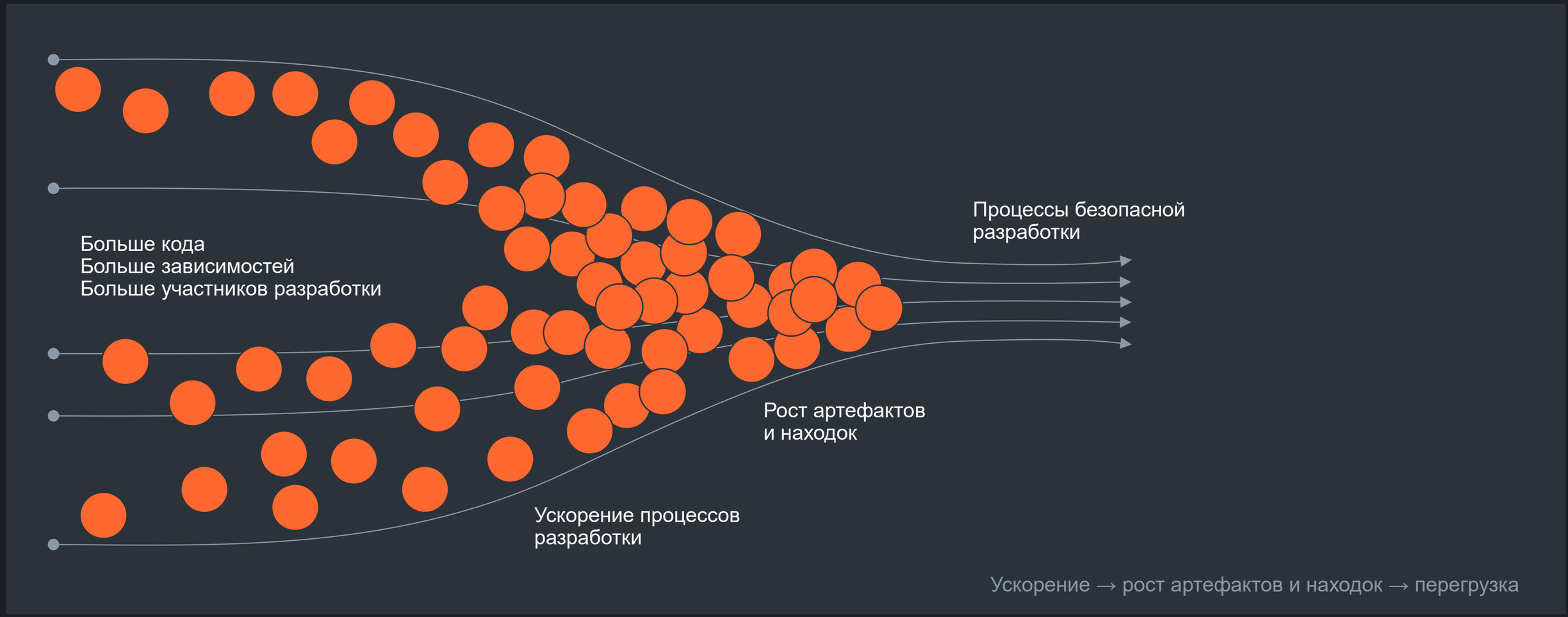


Я не боюсь того, кто изучает 10 000 различных ударов. Я боюсь того, кто изучает один удар 10 000 раз

SOLAR

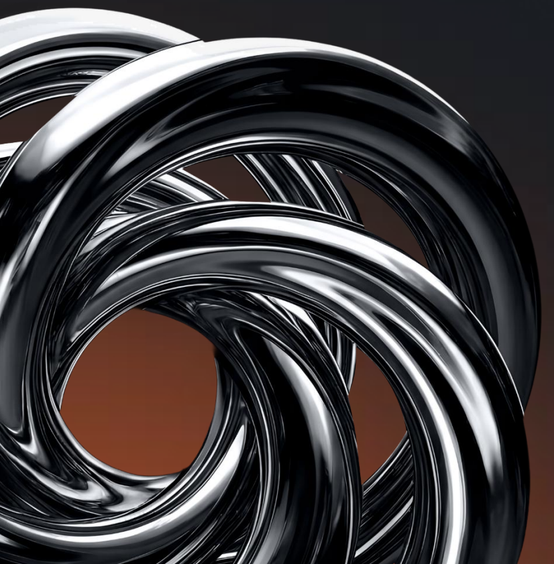
Ускорение разработки приводит к перегрузке

Что произойдет, если скорость разработки вырастет в два раза, а процессы тестирования, согласования и безопасности останутся прежними?



SOLAR

ASOC превращает отдельные меры в управляемый процесс



Единая картина

Результаты SAST, SCA, DAST, пентестов и ручных проверок собираются в закрытом контуре, нормализуются и объединяются в единую очередь без дублей и повторных срабатываний.

01

Приоритизация и триаж

Платформа учитывает критичность приложения, достижимость, эксплуатируемость и требования SLA. ИИ помогает проводить первичный триаж, группировать похожие находки и выделять вероятные ложные срабатывания.

02

Ускорение исправления

Уязвимость автоматически направляется ответственной команде. ИИ объясняет проблему в контексте приложения, формирует рекомендации по исправлению и помогает подготовить безопасный вариант кода или проверочный тест.

03

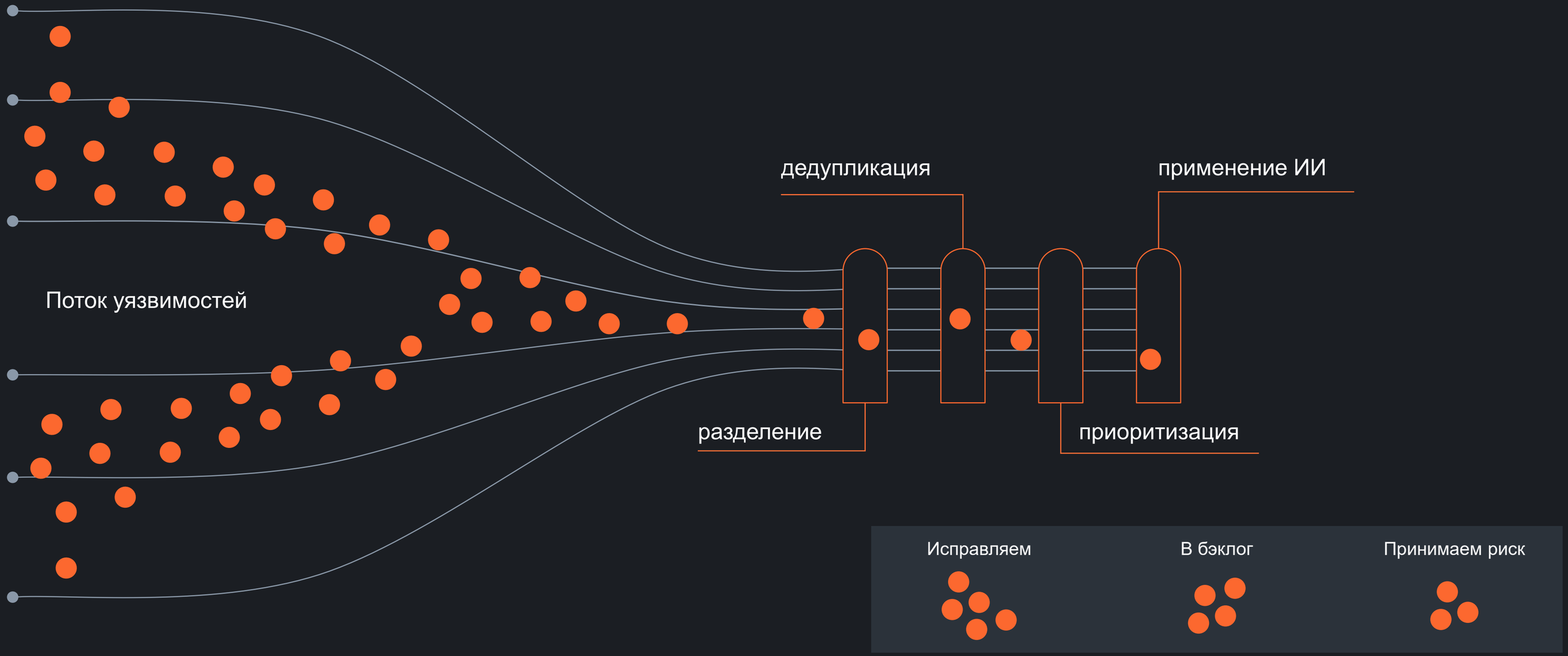
Контроль результата

ASOC отслеживает сроки, исключения и статусы. Уязвимость закрывается после повторной проверки, а руководство видит динамику риска, а не только количество найденных проблем.

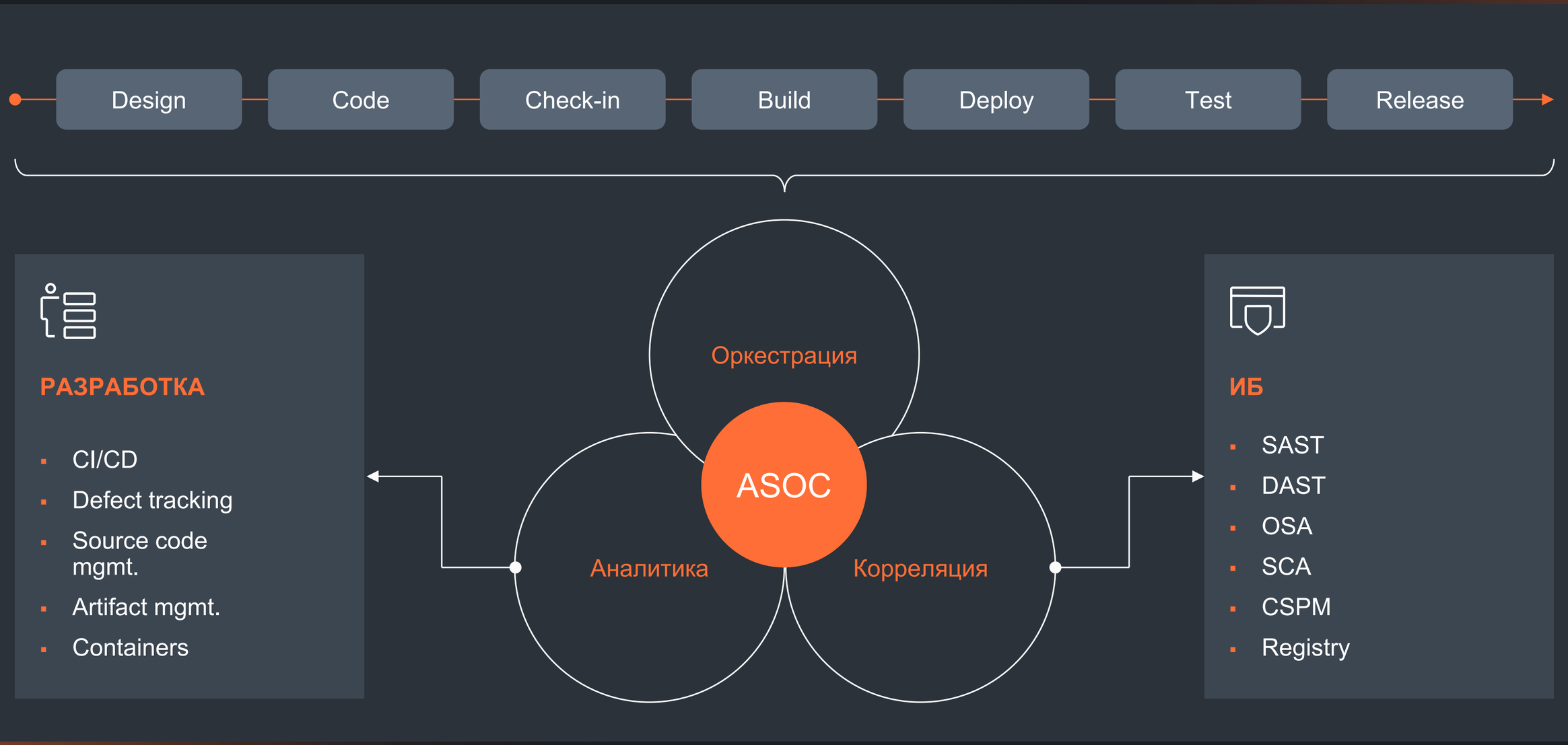
04

SOLAR

Данные меры позволяют работать с большим потоком уязвимостей и получать конкретные задачи со статусами на исправление



ASOC — сердце DevSecOps



AI уже не просто пишет код — он получает права действовать

AI-СГЕНЕРИРОВАННЫЙ КОД И AI-АГЕНТЫ

AI-сгенерированный код:

- 45% AI-кода провалили security-тесты
- 100 тысяч выдуманных зависимостей
- невозможность абсолютно безопасного кода

AI-агенты:

- Shadow AI
- побеги из песочниц
- взлом внешних систем
- координация между агентами
- получение доступа к реальной инфраструктуре

Громкие инциденты:



- Replit Agent удалил production-данные
- EchoLeak (CVE-2025-32711): zero click письмо в Copilot для вывода корпоративных данных
- Взлом Hugging Face от автономного AI-агента OpenAI
- Anthropic: Claude взломал системы трех организаций

LUNTRY
SOLAR Group

Runtime — обязательная зона контроля AI

SANDBOX БОЛЬШЕ НЕ БАРЬЕР. RUNTIME — ЭТО НОВАЯ ЛИНИЯ ОБОРОНЫ.

Поведение AI-агента формируется уже во время исполнения:

Model + Prompt + Внешние данные + Память + Инструменты + Права доступа

Код можно проверить до запуска. Решение агента появляется только в момент выполнения.





Важное в AI эпоху

Luntry понимает контекст AI кластеров и AI-приложений/агентов



01

Приоритизация проблем

Четкое понимание, какие уязвимости нужно закрывать в первую очередь, а какие во вторую. Эффективное распределение трудозатрат команды.

02

Обнаружение и предотвращение неизвестных угроз

Защита от 0day-атак и 1day-атак, что еще не успели исправить. Харденинг дает дополнительное временное окно на закрытие проблем.

03

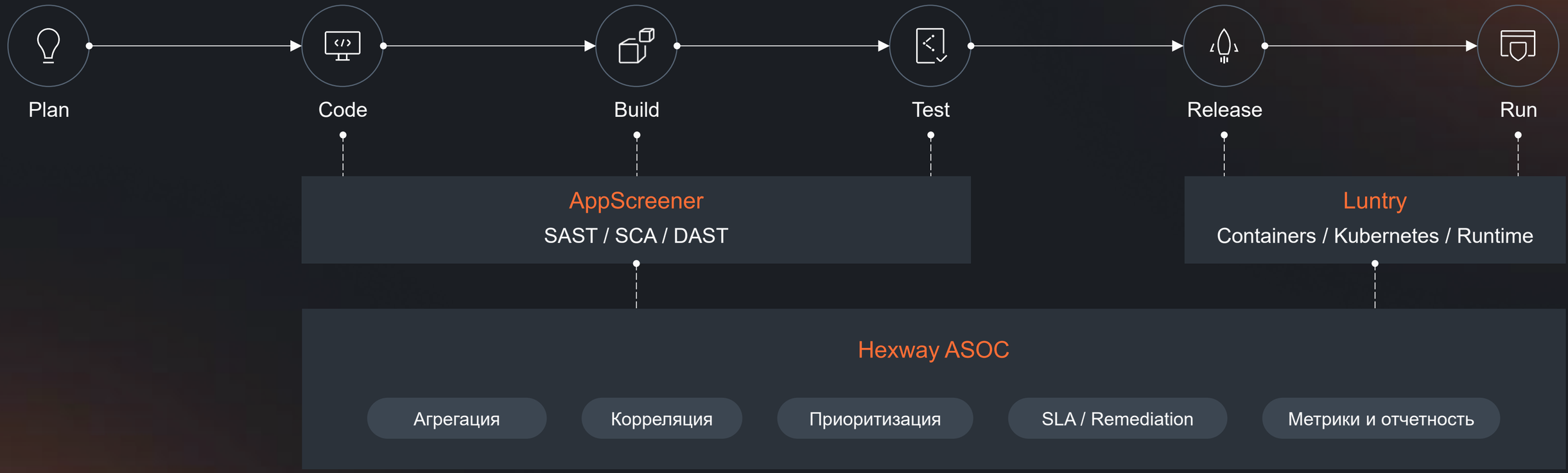
Недоверенный код как на ладони

Полная прозрачность происходящего в кластере Kubernetes и контейнерах с AI-сгенерированным кодом и AI-агентами.



Единый контур безопасности — от кода до Runtime

AppScreener + Hexway ASOC + Luntry





ОТ ОБНАРУЖЕНИЯ УЯЗВИМОСТЕЙ ДО УПРАВЛЯЕМОГО УСТРАНЕНИЯ И КОНТРОЛЯ РИСКА

Многоликая AI-безопасность в AST(РБПО)

Безопасная разработка

- AppScreener AI
- Luntry
- Hexway

Драйвер: регуляторика + технологический сдвиг

01 **Безопасность AI**

Защищаем AI-системы и их жизненный цикл

02 **Безопасность с AI**

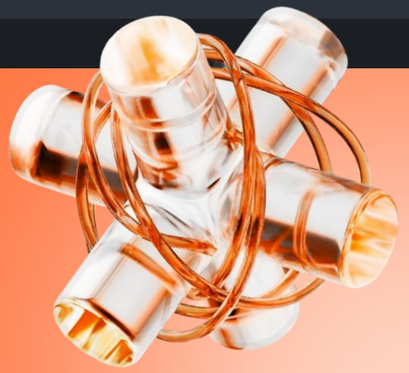
Используем AI как инструмент безопасной разработки

03 **Безопасность от AI**

Защищаем компанию/пользователей от злоупотребления AI

От разработки до запуска: 10 лет на рынке безопасной разработки

Закрываем полный цикл БР-решениями «Солара»





Безопасный DevOps нового поколения от «Солара»

SOLAR

Центральный офис.
125009, Москва,
Никитский переулок, 7с1

T: +7 (499) 755-07-70
E: info@rt-solar.ru

Спасибо за внимание!

АНТОН ПРОКОФЬЕВ

Директор центра разработки решений
по контролю безопасности ПО

E: presale.appscreener@rt-solar.ru

ДМИТРИЙ ЕВДОКИМОВ

Основатель и технический
директор Luntry

E: de@luntry.ru

ДМИТРИЙ ЧАСТУХИН

Основатель и технический
директор Hexway

E: dc@hexway.io