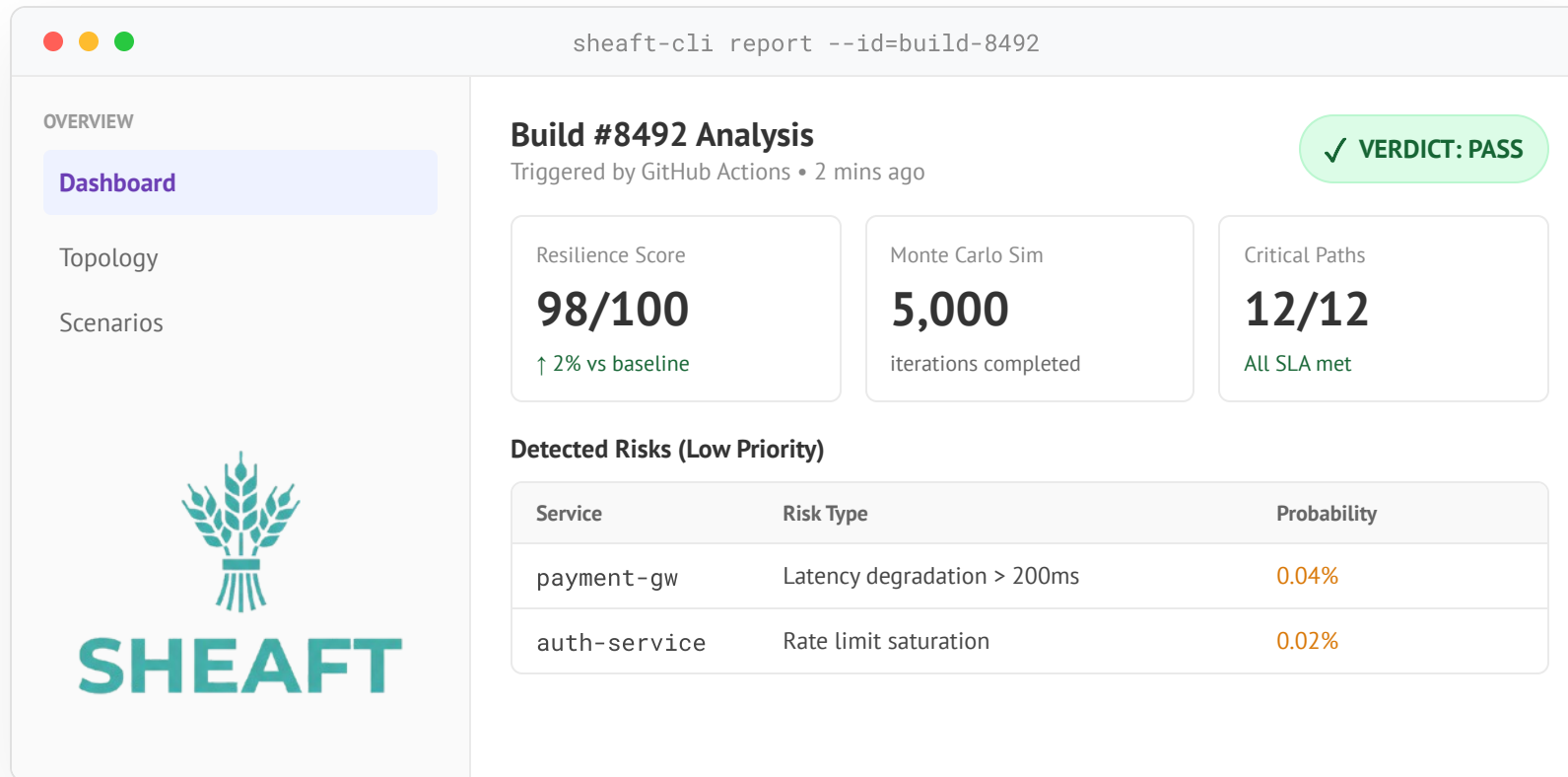


Для любых систем с сетевой структурой!

Sheaft

Виртуальные краш-тесты IT-систем до релиза

Sheaft автоматически строит цифровой двойник сервиса и его зависимостей, тестирует на нём риски отказов и объясняет, почему устойчивость низкая, до того как слабое место превратится в инцидент



Пример отчёта Sheaft

Сейчас устойчивость проверяют поздно, дорого и вручную

Риски находят после инцидента, при этом проверка перед релизом остаётся ручной и дорогой.

Что происходит сейчас

Риски видят слишком поздно

Наблюдаемость помогает разбирать уже проявившиеся проблемы, но не показывает слабые места до выпуска.

Ручной chaos engineering дорогой

Нужны отдельные серверы и редкая экспертиза, поэтому тесты не делают на каждом релизе.

Ручные схемы быстро устаревают

Зависимости меняются быстрее, чем инженеры успевают обновлять диаграммы и таблицы рисков.

Что уже видно по рынку

\$300k+

потенциальная цена часа
недоступности у Enterprise-
бизнеса

57.4%

не используют chaos engineering

23%

пишут fault-injection скрипты
сами

63%

строят инструменты надёжности
in-house

ВЫВОД 1

Рынок без
стандарта

ВЫВОД 2

Много самописного
софта

ВЫВОД 3

Скрипты и костыли
копят техдолг

**Рынок закрывает пробел кустарно, накапливая
технический долг.**

Опрос DevCrowd SRE 2025: overall n=273; вопрос о chaos engineering n=183.

Мы превращаем телеметрию в предрелизную проверку рисков

Sheaft автоматически строит цифровой двойник IT-системы, симулирует миллионы аварий и показывает, где просела устойчивость.

ШАГ 1

Телеметрия

Трејсы, логи, OTLP и конфигурации уже есть у клиента.

ШАГ 2

Цифровой двойник

Строим модель зависимостей, реплик и критических путей.

ШАГ 3

Проверка рисков

Алгоритмы проверяют миллионы виртуальных сценариев отказов за минуты.

ШАГ 4

Вердикт

Команда получает зелёный, жёлтый или красный сигнал по риску релиза.

В чём инновация

- Убираем ручной труд из регулярной проверки устойчивости.
- Проверяем не прод, а ещё не выпущенное изменение.
- Заменяем недели ручных тестов на автоматический анализ за 2 минуты.

Что подтверждает инновационность

ICSE 2026 Distinguished Paper Award

международное признание инновации как выдающейся

Заявка на патент РФ

защита технологии в России

Регистрация программы для ЭВМ

защита программного кода

Грант ФСИ

поддержка deep-tech разработки

Статус МТК

статус малой технологической компании

Sheaft проверяет устойчивость до релиза и в рантайме

2 режима работы. До деплоя видно, какие операции и зависимости могут просадить SLA-KPI. В наблюдаемости — какие пути стали рискованными реалтайм и что нужно исправить первым.

ВЫХОД 1

Отчёт по рискам

Например: Критическая зависимость от auth-service + DB Cluster и отсутствие резервирования.

ВЫХОД 2

Решение по релизу

Понятный вариант действия: можно деплоить, можно деплоить с предупреждением или сначала устранить найденные слабые места.

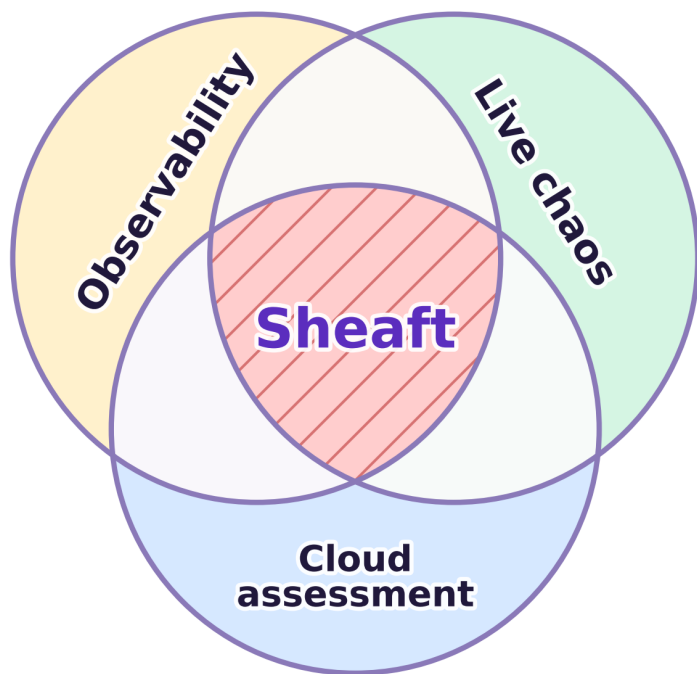
ВЫХОД 3

Сигнал в режиме наблюдаемости

Текущее состояние устойчивости можно рассчитывать "на лету" и транслировать на дашборд наблюдаемости рядом с метриками, журналами и трассировками.

Это снижает непродуктивные траты (toil) между платформой, разработкой и эксплуатацией и помогает точнее выбирать живые проверки.

Формируем новый класс решений



Синтез мониторинга, хаоса и аудита

С кем мы (не) конкурируем

КЛАСС РЕШЕНИЯ	КТО ДЕЛАЕТ	ОГРАНИЧЕНИЕ
Observability (мониторинг)	Monq, Proto, Smart Monitor, Yandex Monium, Coroot, Datadog	Дают сигнал поздно. Разбирают проявившуюся проблему, но не показывают риск до релиза.
Live Chaos (живой хаос)	Gremlin, Chaos Mesh, in-house scripts	Риск для продакшена. Не подходит как регулярная превентивная проверка каждого релиза.
Cloud Assessment (аудит)	AWS Resilience Hub, cloud-provider tools	Привязаны к вендору. Статичны, не проверяют бизнес-логику самого кода.
Virtual Chaos	Sheaft	До релиза. Без рисков. Вендор-агностик.

Готовность технологии

Технология уже оформлена в проверяемый продуктовый контур.

ПУБЛИЧНЫЙ КОНТУР

Что уже доступно

Procrustes v0.1.0 — проверяет готовность контура к анализу Bering и формирует план сбора данных.

Bering v1.0.0 — строит модель сервисов и критических операций из трассировок или явной топологии; публикует качество сигнала.

Sheaft v1.2.0 — считает границу отказоустойчивости по операциям, сертифицирует запас и блокирует регрессии релиза.

ИННОВАЦИОННОСТЬ

Валидация. Научная база

ICSE 2026, Distinguished Paper Award

Работа Model Discovery and Graph Simulation отмечена как выдающаяся в треке New Ideas and Emerging Results. Подход проверен на эталонном микросервисном стенде DeathStarBench и показывает практичность анализа зависимостей.

AINA 2026 (CORE B)

Кейс на OpenTelemetry Demo уточняет границы модели с асинхронной семантикой и показывает полезность сигнала на прикладном стеке.

ЧТО ПОД КАПОТОМ

Никаких чёрных ящиков

Детерминированный алгоритм — интерпретируемый и проверяемый расчёт по модели зависимостей, а не чёрный ящик ИИ.

Без inference-инфраструктуры — не использует нейросети, написан на Go и отрабатывает за несколько минут. Не требует дополнительной инфраструктуры.

В контуре заказчика — разворачивается локально, не требует доступа к внешней сети и не вызывает внешние API.

Пилот показывает бизнес-результат на исторических данных

Команда передаёт трассировки одного контура и историю деградаций SLA. Sheaft строит модель у себя, проверяет риски отказов и показывает, какие слабые места можно было увидеть заранее.

Вход

Что нужно от команды

один сервисный контур, где есть заметная цена деградации SLA

трассировки, OTLP или выгрузка телеметрии по этому контуру

история деградаций SLA и инцидентов за выбранный период для последующей сверки

Прогон

Что делаем мы

без интеграции принимаем выгрузку данных и уточняем границы контура

строим модель зависимостей, реплик и критических пользовательских путей

проверяем риски отказов и формируем отчёт по слабым местам до деплоя

Выход

Что на выходе

карта слабых мест и зависимостей, которые создают риск просадки SLA

объяснение причин: где нет резервирования, какие связи критичны и почему риск высокий

приоритетный список исправлений, которые стоит закрыть до релиза или следующего релизного окна

Первый шаг без интеграции: не нужен агент, CI/CD gate или подключение к продакшену.

Стейкхолдеры и результаты от внедрения Sheaft

Cui bono?

РОЛЬ	ТЕКУЩАЯ ПРОБЛЕМА	РЕЗУЛЬТАТ
SRE операционная устойчивость	Инциденты после релизов, скрытые зависимости и ручные разборы забирают время команды.	Снижение инцидентов в продакшене. Улучшение KPI подразделения (SLA). Меньше toil на устранение аварий.
Platform Engineering, DevOps релизный процесс	Нужна проверка релиза без ручных схем, экспериментов в продакшене и задержек delivery.	Проверки устойчивости в delivery-процессе. Больше успешных релизов без просадки KPI и SLA. Проблемные обновления отсекаются до продакшена.
Backend, Product критичные пользовательские пути	Зелёные тесты не показывают, какие изменения в зависимостях ударят по пользовательскому пути.	Слабые места видны до релиза. Меньше возвратов и срочных исправлений. Критичные пользовательские сценарии защищены.
Head of Engineering, CTO спонсор результата	Нужен измеримый способ снижать релизный риск без роста ручных проверок и эскалаций.	Релизный риск снижается измеримо. SLA-KPI выполняются без роста ручного контроля. Toil становится меньше.

Sheaft снижает затраты и ускоряет вывод изменений

Меньше ручной координации перед выпуском означает короче путь от изменения до релиза.

Где теряется время

- 1 Понять, какие сервисы и операции затронуты изменением.
- 2 Дождаться владельцев зависимостей, эксплуатации и архитектора.
- 3 Согласовать оценку риска по косвенным признакам.
- 4 Подготовить решение выпускать, отложить или проверить глубже.

Ручная координация удлиняет путь от изменения до релиза.

Как Sheaft ускоряет процесс

Собирает картину влияния

сервисы, операции и зависимости по данным наблюдаемости

Сузит проверку

слабые связи, отсутствие резерва и критичные пути

Даёт обоснование решения

почему выпуск можно продолжить или нужно задержать

Вердикт для выпуска

команда быстрее переходит от обсуждения к решению

Стоимость ручного этапа релиза

Считается только повторяемая координация, которую может заменить Sheaft. Без учёта предотвращённых инцидентов.

- 1 Ручной этап на релиз 1–5 ч на релиз
- 2 Релизы с таким этапом 150–300 в год
- 3 Стоимость ручного этапа релизы × часы × 2 000 ₽
- 4 Цена Sheaft для сравнения 2 млн ₽ в год

Ручной этап против цены Sheaft

Релизы	1 ч	2 ч	3 ч	4 ч	5 ч
100	10% 0,2	20% 0,4	30% 0,6	40% 0,8	50% 1,0
150	15% 0,3	30% 0,6	45% 0,9	60% 1,2	75% 1,5
200	20% 0,4	40% 0,8	60% 1,2	80% 1,6	100% 2,0
250	25% 0,5	50% 1,0	75% 1,5	100% 2,0	125% 2,5
300	30% 0,6	60% 1,2	90% 1,8	120% 2,4	150% 3,0

Контакты

Технический разбор, материалы проекта и прямой контакт с командой для следующего разговора по внедрению.

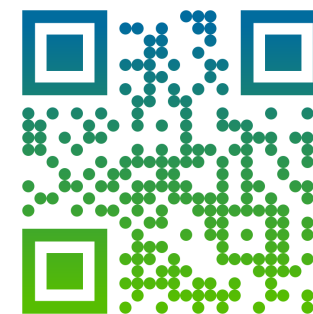


Анатолий Красновский

Основатель/Технологии
contact@mb3r-lab.org
+7 923 410-32-25



10+ лет в бэкенде и архитектуре. Эксперт по надёжности и графовым моделям. Автор научной базы проекта (кандидатская диссертация на тему моделирования надёжности распределённых систем).



Сайт проекта

Документация, GitHub и материалы
для технической проверки и
следующего разговора.