

Курс Дмитрия Якубовского по Chaos Engineering

Contents

- [Где применяется chaos engineering?](#)
- [Содержание и ожидаемые результаты курса](#)
- [Авторы курса и контакты](#)
- [Напишите нам в Telegram](#)
- [Глоссарий](#)

Тестирование надежности и отказоустойчивости информационных систем

start-chaos 0.4.6

Отказоустойчивость

Микросервисные и распределенные архитектуры позволяют создавать более сложные и масштабируемые ИТ-системы. В эксплуатационной среде неизбежны сбои, к которым одни системы готовы, а другие — нет. Проверку отказоустойчивости обеспечивает новый вид тестирования, который называется chaos engineering.

От Netflix до CNCF

Впервые chaos engineering стали применять в компании Netflix в ходе миграции видеосервиса на облачную инфраструктуру. В 2011 году Netflix выпустил в открытый доступ первый инструмент для проведения хаос-испытаний — [Chaos Monkey](#).

Лидеры направления chaos engineering в настоящее время это Amazon, Bloomberg, Netflix и Alibaba, крупнейшие провайдеры SaaS-сервисов для хаос-испытаний — американская Gremlin и европейская ChaosIQ (Великобритания/Франция).

С 2017 года инструменты chaos engineering стали официальной частью ландшафта технологий, которые [Cloud Native Computing Foundation](#) (CNCF) рекомендует для Kubernetes.

Хаос на российском рынке

В августе 2022 года на сервисе Headhunter было 1942 вакансии по DevOps, 193 по SRE и 6 по chaos engineering. Хаос-испытаниями занимается достаточно небольшой круг продвинутых команд и компаний, которые чаще всего работают с массовыми, высоконагруженными сервисами.

Кому подходит и кому не рекомендуется chaos engineering мы рассказываем [здесь](#).

Где применяется chaos engineering?

Для каких компаний подходит chaos engineering? На наш взгляд, chaos engineering наиболее полезен компаниям, которые:

- работают в сферах, где потребитель чувствителен к уровню надежности ИТ-сервисов;
- могут показать ценность надежной работы сервисов акционерам компании;
- ожидают роста требований по надежности со стороны регуляторов (например, в финансовом секторе);
- активно используют микросервисы и распределенные системы;
- имеют достаточные компетенции в проектировании отказоустойчивого ПО;
- планируют миграции на новые архитектуры или стек технологий, при этом хотят убедиться в отказоустойчивости нового решения;
- намерены совершенствовать техническую и организационную культуру компании.

Воспользоваться chaos engineering будет сложнее компаниям, у которых:

- продукт или сервис находятся в начальной стадии разработки или прототипирования, архитектура приложений будет меняться;
- нет собственной разработки, ведется только настройка и эксплуатация приобретенного ПО;
- не стоит задача создания отказоустойчивого ПО;
- не предполагается масштабирование сервиса на большое число пользователей;
- по архитектуре преобладают монолитные приложения;
- специализация преимущественно на front-end разработке;
- не документируются требования или SLA используемых систем;
- не выстроен современный релизный цикл ПО или релизы занимают длительное время;
- проводится мало других видов тестирования помимо хаос-инжиниринга;
- нет мониторинга надежности сервисов в стадии эксплуатации;
- не ведется история и разбор инцидентов в области надежности;
- надежность сервиса не критична для пользователей;
- эксплуатируются legacy и end-of-life приложения, не подлежащие рефакторингу или доработке.

Содержание и ожидаемые результаты курса

С помощью курса вы получаете доступ к современной практике проведения хаос-испытаний.

Курс позволяет сократить время на подготовку первого хаос-испытания и ускорить распространение практики chaos engineering на уровне компании.

Слушатели курса осваивают методологию испытаний, учатся настраивать и запускать инструменты для внедрения атак и участвуют в проведении реального хаос-испытания на тренировочном кластере.

Основные части курса

1. Что такое хаос-испытание? Понятие аномалии. Гипотеза испытания.
2. Типовая схема подготовки, проведения и использования результатов испытания.
3. Классификация атак по видам систем.
4. Инструментарий проведения атак. Практикум по инструментам.
5. Результаты испытания и дальнейшая работа с ними.
6. Практикум проведения хаос-испытания.

Курс проводится в течение одного дня, время занятий составляет 6 часов.

Выигрыши

По нашей оценке, курс экономит до 400 часов, которые необходимы для выбора и изучения документации по инструментам, а также на разработку и апробацию регламентов проведения хаос-испытаний.

Целевая аудитория

Курс будет полезен компаниям, которые:

- видят успешные примеры применения хаос-тестирования в своей отрасли;
- начали хаос-испытания в одной команде или проекте и хотят масштабироваться;
- планируют крупную миграцию архитектуры или смену технологического стека.

Возможные слушатели курса внутри компании:

- будущие и действующие хаос-инженеры;
- команды эксплуатации, DevOps, SRE;
- тестировщики и команды тестирования;
- разработчики;
- архитекторы;
- руководители технических подразделений и СТО.

Помимо этого курс будет полезен сотрудникам и руководителям подразделений, отвечающих за совершенствование технических и бизнес-процессов, технический аудит, анализ и управление рисками в ИТ сфере.

Адаптация курса

Возможна адаптация курса под индивидуальные запросы компании и конкретную аудиторию и уровень подготовки слушателей.

Пример трехдневного курса

🔗 День 1.	* День 2.	☐ День 3.
Внедрение хаос-испытаний в компании.	Инструментарий и правила испытаний.	Автоматизация хаос-испытаний.
Управленческий модуль	Технический модуль	Технический модуль

Авторы курса и контакты



Дмитрий Якубовский

Telegram [Дмитрий Якубовский \(@yakubovskyn\)](#)

Исполнительный директор, лидер направления Chaos Engineering в розничном блоке Сбербанка. Опыт работы в сфере Chaos Engineering с 2015 года.



Евгений Погребняк

Telegram [Евгений Погребняк \(@epogro\)](#)

Напишите нам в Telegram

1. Задайте интересующий вас вопрос по chaos engineering.
2. Расскажите о потребностях вашей компании в хаос-испытаниях.
3. Уточните содержание курса и закажите его проведение.

Глоссарий

Chaos engineering

Дисциплина и набор практик по проведению [экспериментов](#), подтверждающих способность информационных систем противостоять неблагоприятным условиям в эксплуатационной среде (адаптировано из [Principles of Chaos](#)).

CNCF

[Cloud Native Computing Foundation](#).

DevOps

Development and operations. Набор методик и инструментов, которые позволяют автоматизировать и интегрировать между собой процессы разработки и эксплуатации ПО.

K8s

[Kubernetes](#).

Principles of Chaos

Манифест методологии [chaos engineering](#), опубликован по адресу <https://principlesofchaos.org/>.

SLA

Service Layer Agreement. Соглашение о предоставлении услуг – договор между заказчиком и поставщиком, описывающий согласованный уровень качества предоставления ИТ-услуги.

SRE

Site Reliability Engineering. Набор принципов, индикаторов и практик для обеспечения надежности ИТ-систем, который [разработан и популяризируется компанией Google](#).

Атака

Создание неблагоприятных условий работы системы в ходе [хаос-испытания](#).

Гипотеза

Предположение о поведении системы при конкретном виде воздействия ([атаке](#)) на конкретный архитектурный элемент [системы](#).

Инструмент

Утилита для проведения [атаки](#) (эмуляции отказа).

Испытание

Контролируемое внесение изменений в условия работы сервиса или системы на тестовом стенде или в эксплуатационной среде под рабочей нагрузкой. Испытание проводится для подтверждения или опровержения заранее сформулированной [гипотезы](#) относительно поведения системы в неблагоприятных условиях.

Синонимы: хаос-испытание, эксперимент, хаос-эксперимент.

ПО

Программное обеспечение.

Система

Конкретная информационная система, которую мы изучаем или тестируем.

Синонимы: автоматизированная система (АС), сервис.

Точка отказа

Ситуация нештатного поведения системы, которая найдена в ходе [хаос-испытания](#).

Версия 0.4.7