



# Как мы встраивали fuzzing-тестирование, или как огрести кучу проблем одним коммитом

Косых Руслан, ведущий инженер по автоматизации тестирования

Обо мне



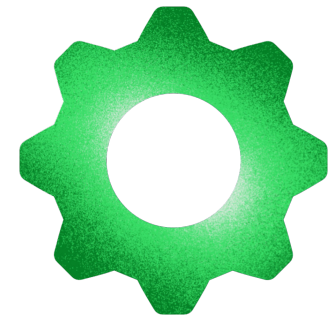
**Косых Руслан**

 @ruko12

ozon{tech

## Ведущий инженер по автоматизации тестирования Ozon Fintech

- Занимаюсь **тестовой инфрой**
- Пишу сервисы на **Go**
- Нагружаю сервисы
- Преподаю на **Route256**



# Ozon Fintech в цифрах

250+

сервисов

1 к 4

среднее соотношение  
QA к DEV

2500+

E2E API автотестов

400+

релизов в неделю (или 80+ в день,  
или ~1 релиз каждые 6 минут)

250+

прогонов E2E  
тестов в день

~10-13<sub>мин</sub>

прогон E2E-тестов  
по сервисам



# Поддерживаем высокий темп разработки!

## Немного о процессах

### 1. Единицы ручных тестов

- Мы исповедуем подход «test as code»

### 2. Разработчики сами пишут компонентные тесты

- Мы активно помогаем им

### 3. Много времени уделяем качеству кода сервисов

- Тестовое покрытие всех новых фич должны быть более 60%
- Целевые ориентиры для тестового покрытия:
  - на сервис
  - на команду разработки/департамент





# Поддерживаем высокий темп разработки!

Немного о процессах

## Разрабатываем инструменты:

- **Сбора и анализа тестового покрытия**
- **Для анализа трейсов и оптимизации запуска e2e-тестов**
  - Понимаем покрытие e2e-тестами



# Отлавливаем еще до E2E-тестов

100% покрытие тестами — миф

## «Запросы-убийцы»

- На уровне service-2-service

## Недостаточно валидации

- В новой фиче

## Прочие радости жизни

- Редкие кейсы

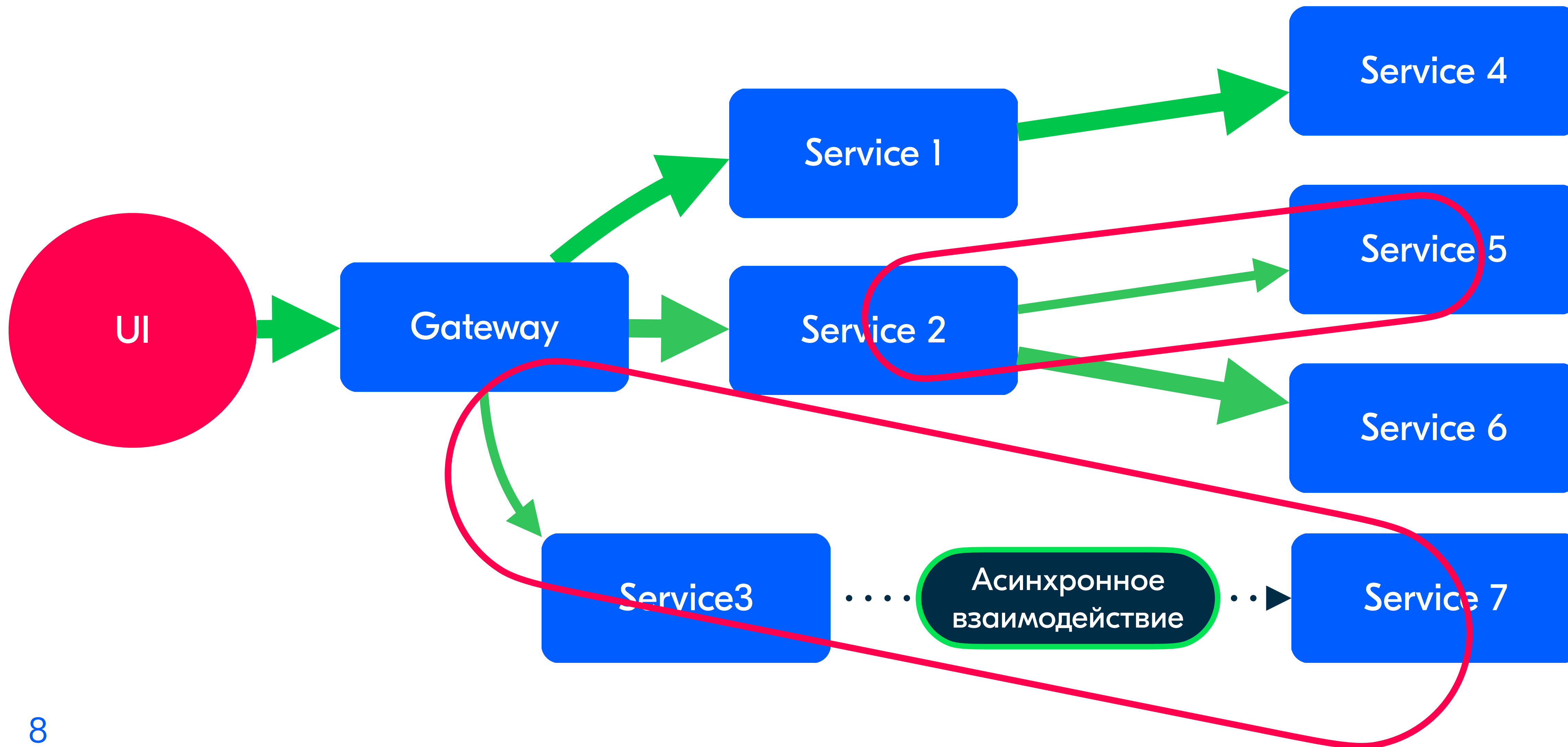


1

Какую проблему мы решаем?



# Часть API слабо задействована в e2e-тестах



# Часть API слабо задействована в e2e-тестах

Не совсем так!

## e2e-тесты — сценарные тесты

- Тяжелые тесты
- Проверяют бизнес-логику [user story]

## Компонентные тесты в сервисе

- Легче и быстрее
- Проверка валидации
- Проверка логики работы сервиса  
(интеграция с БД, но на моках)



# Опять человеческий фактор

Да скока можна!

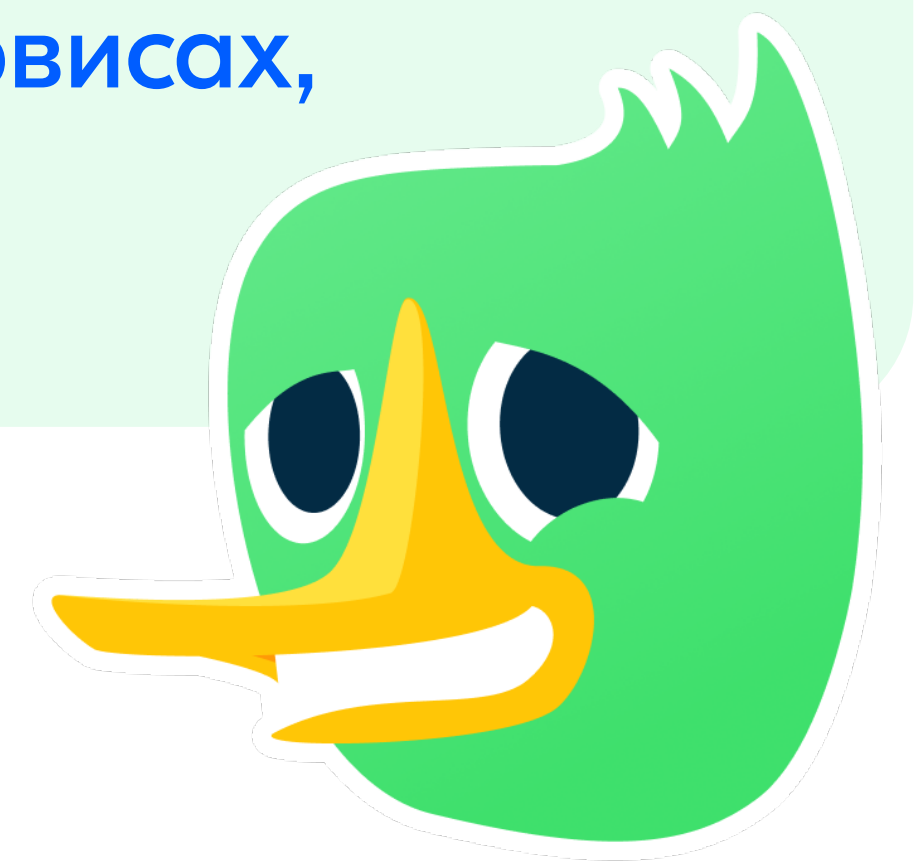
**Разрабы тоже люди!**

**«Разрабов много и за всеми не уследишь**

- ~~А так хочется!~~
- А точно эффективно и удобно масштабировать?

**На самом деле хочется:**

- Инструмент который автоматом проверит: **редкие кейсы, кейсы с невалидными данными и т.д.**
- Защиту от невалидных запросов во всех сервисах, не только в gateway





# А какую проблему мы хотим решить?

Действительно...

## Проблема:

- Есть 100500 сервисов
- У них 100500 API интерфейсов
- Покрыть автотестами за разумное время
  - ✓ сможет только семья Торетто из Форсажа



# А что в итоге хотим получить?

Хорошо, что про это вспомнили сейчас

**1. Протестить сервисы сейчас и получить инфу о существующих проблемах**

**2. Простое подключение для всех сервисов**

- Регулярные запуски

**3. Понятные отчеты для QA и для Dev**



# 2

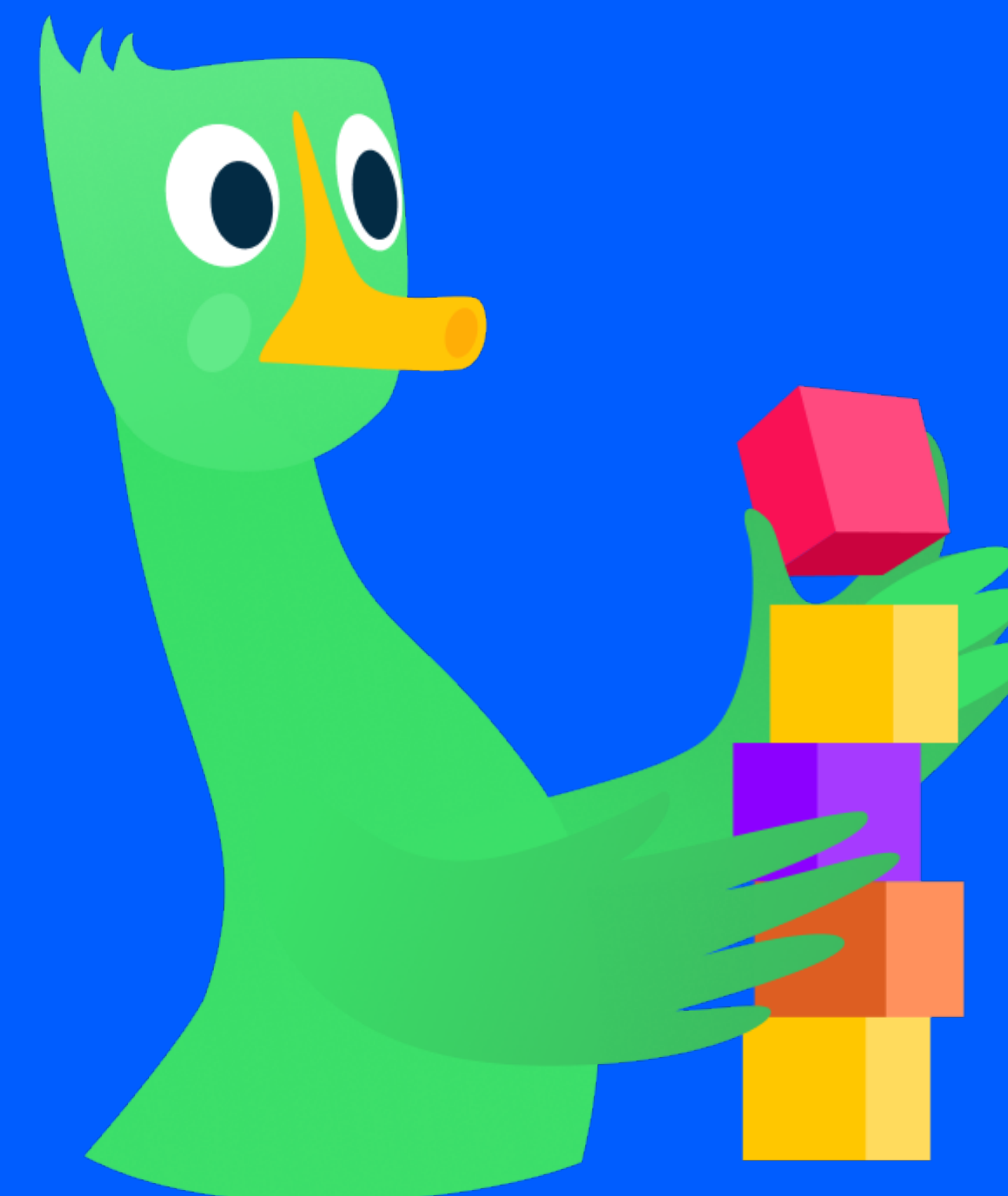
Поиски решения



# Fuzzing?!

Ищем решение...

- **Техника тестирования ПО**
  - Передача приложению неправильных, неожиданных или случайных данных
- **Выявляем**
  - Падения
  - Зависания
  - Нарушения внутренней логики
  - Нарушения проверок
  - Утечки памяти из-за входных данных







Наша статья на Хабре на тему «Что такое фаззинг и зачем он нужен?»





# Schemathesis. Property-based testing for API schemas

Specification-centric API testing tool for Open API and GraphQL-based applications

- **Суть подхода**
  - Определяем свойства, которым должен удовлетворять код
  - Определяем проверки для свойств (выполняются на большом количестве случайно сгенерированных данных)

**Пример: функция сложения двух чисел (обладает свойством коммутативности)**

 SCHEMATHESIS [Documentation](#) [Pricing](#) [Contact Us](#)

## Ensure Your API Delivers Only Quality with Schemathesis.

Catch, debug, and resolve API issues before they reach your users. Focus more on delivering value and less on troubleshooting.

Secure Your API Today!



# Schemathesis: property-based тестирование для API схем





# Простое подключение для всех сервисов

Ищем решение...

# Gitlab?!

Ищем решение...

- **Job'a находится в pipeline репозитория:**
  - Знает всю инфу о конфигурации сервиса
  - Прямой доступ до swagger-документации в репе сервиса
  - Можно запускать в main-ветке
  - Можно запускать в feature-ветке



# Понятные отчеты для QA и для DEV

Ищем решение...

# Schemathesis summary

Можно отправлять в облако на [schemathesis.io](https://schemathesis.io)

```
384 ----- PATCH /api/users/{user_id} -----
385 1. Received a response with a status code, which is not defined in the schema: 400
386 Declared status codes: 200, 404
387 Response status: 400
388 Response payload: `<html>
389 <head><title>400 Bad Request</title></head>
390 <body>
391 <center><h1>400 Bad Request</h1></center>
392 </body>
393 </html>
394 `
395 Run this cURL command to reproduce this failure:
396 curl -X PATCH -H 'X-Schemathesis-TestCaseId: 6910e195fe6e432ca51d60974b7ad23c' -H 'token: default' -d '{"first_name": "00
0", "last_name": null}' 'https://example.schemathesis.io/api/users/%00?common=0'
397 Or add this option to your command line parameters: --hypothesis-seed=54604232897874082840818088631366202584
398 ===== SUMMARY =====
399 Performed checks:
400 not_a_server_error          1703 / 1738 passed      FAILED
401 status_code_conformance    1618 / 1738 passed      FAILED
402 response_headers_conformance 1738 / 1738 passed      PASSED
403 response_schema_conformance 1728 / 1738 passed      FAILED
404 JUnit XML file: schemathesis_report.xml
405 WARNINGS:
406 - Most of the responses from `GET /api/basic` have a 401 status code. Did you specify proper API credentials?
407 - Most of the responses from `GET /api/headers` have a 401 status code. Did you specify proper API credentials?
408 Hint: You can visualize test results in Schemathesis.io by using `--report` in your CLI command.
409 ===== 22 passed, 13 failed, 4 errored in 215.95s =====
```

The screenshot shows a Schemathesis Report for 'Payments API 2.0'. It includes a table for 'Summary' with columns: Started At, Checked Endpoints, Unhealthy Endpoints, Tests, and Duration. Below it is a 'Schemathesis Checks' table with columns: Name, Failures, and Change. The report also includes a link to 'Continue to review full report at Schemathesis.io' and a list of notes at the bottom.

Specification	Schema Location	Endpoints	Commit
Open API 3.0.1	<a href="https://api.schemathesis.io/openapi.json">https://api.schemathesis.io/openapi.json</a>	44	<a href="#">8472419</a>

Started At	Checked Endpoints	Unhealthy Endpoints	Tests [1]	Duration
2022-07-23 23:00:00Z	44	2 <span style="color:red">●</span>	9000	2m 55s

Name	Failures [2]	Change [3]
Server error	2 <span style="color:red">●</span>	+2 <span style="color:blue">↑</span>
Response conformance	0 <span style="color:green">●</span>	-2 <span style="color:blue">↓</span>
Undeclared status code	0 <span style="color:green">●</span>	0

► [Other issues](#)

[Continue to review full report at Schemathesis.io](#)

- 1. Total number of test assertions for all endpoints [🔗](#)
- 2. Unique failures across all endpoints [🔗](#)
- 3. Against the `main` branch [🔗](#)

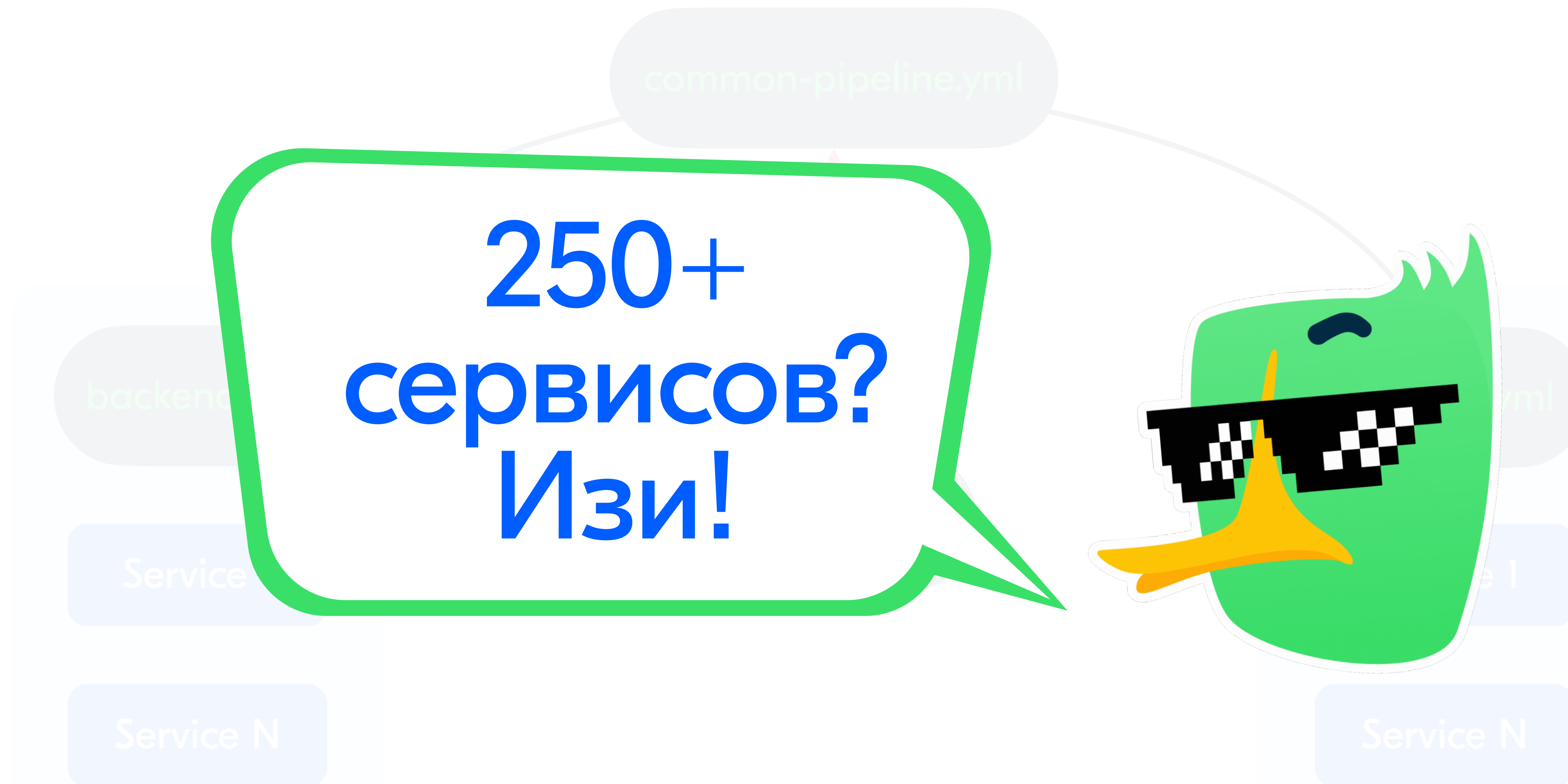


# 3

Трудности при подключении

# Gitlab include

Стиснув зубы, сжав булки раскатываем на всех!

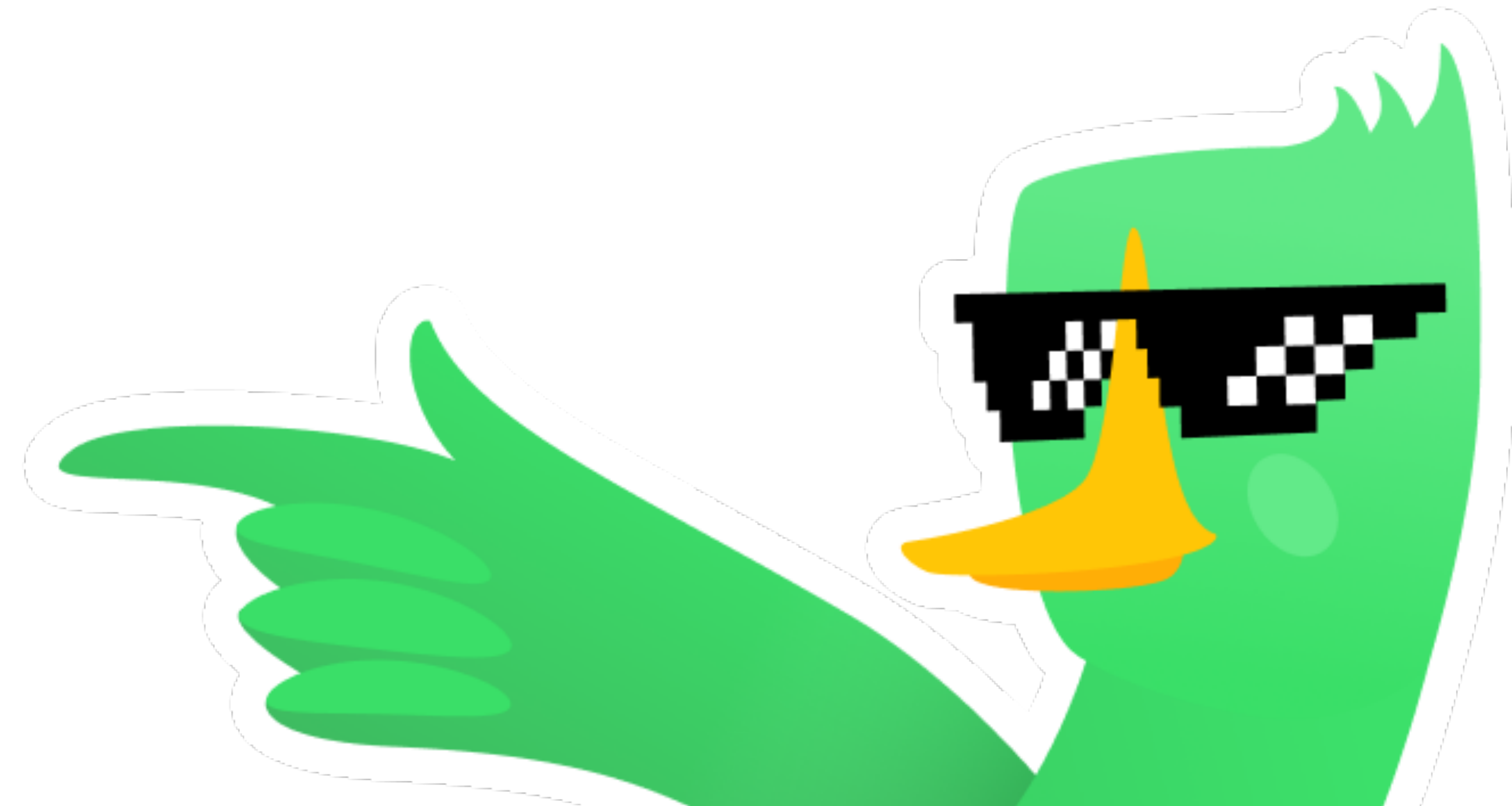


# Реакция разработчиков ... **реальность**

Повторим еще разок?

Круто! Мы уже позапустили  
и пофиксили все замечания!

- **Помогает в разборе инцидентов**
- **Запускаем для новых API интерфейсов и сервисов**



# Реакция сервисов ... **реальность**

Повторим еще разок?

- **Большой RPS от schemathesis**
  - У сервиса много API интерфейсов
- **Решение:**
  - Ограничить кол-во параллельных запросов!
  - Кастомизация для особых случаев!
    - Токены авторизации
    - Исключить некоторые API интерфейсы





4

Какие баги обнаружили?

# Нет валидации данных в запросе

500 Internal error

## Самые частые кейсы:

- Нет валидации входящего запроса
- Любая негативную ситуация — 500 Internal Error



# «Запрос-убийца»

\*играет музыка из триллера...\*

- **В БД очень много данных.**  
**В запросе передаем объединяющий ID**
  - Например, `userID`
- **А если...**
  - `userID` пустой
  - поле `userID` null

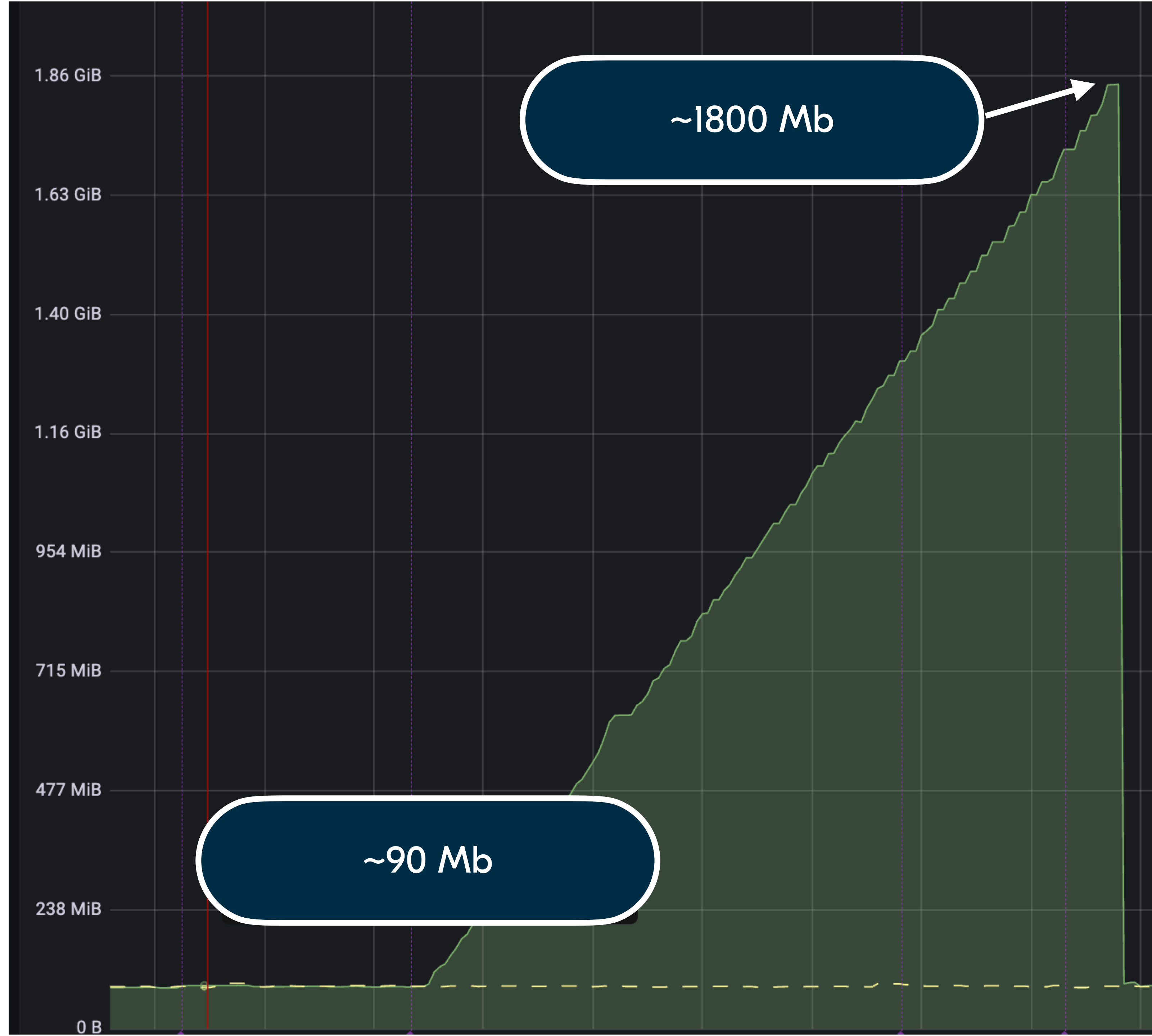


# Утечка ресурсов

Кап-кап-кап

- **Обрабатываем  
огромное  
количество  
данных...**

**И не падаем!**





5

Что дальше?

The screenshot displays the Allure test results interface. On the left is a navigation sidebar with options: Overview, Categories, Suites, Graphs, Timeline, Behaviors, and Packages. The main area shows a list of test suites under the heading 'Suites'. A table lists test cases with columns for order, name, duration, and status. The test case '#3 test\_app[GET /team/teamtree]' is highlighted in yellow and marked as failed. The right panel provides details for this failed test, including the error message 'Failed test\_app[GET /team/teamtree]', the error type 'schemathesis.exceptions.CheckFailed', and the response payload: `{"code":13, "message":"can't get root team: sql: no rows in result set", "details":[]}`. A large dark blue box with the word 'CENSORED' in white is overlaid on the bottom right of the error details. Below the error details, there are tags for 'hypothesis', categories for 'Product defects', and a severity of 'normal'.

order	name	duration	status
Status: 2 2 27 0 0	Marks: [Icons]		
tests		2 2	
amazing service		2 2	
#3	test_app[GET /team/teamtree]	11s 196ms	Failed
#17	test_app[POST /config/project/{project}/key/{key}]	10s 510ms	Failed
#21	test_app[POST /targets/approve/{targetId}]	5m 02s	Warning
#4	test_app[POST /targets/disapprove/{targetId}]	5m 09s	Warning

tests.schemathesis\_test#test\_app

**Failed** test\_app[GET /team/teamtree]

Overview History Retries

schemathesis.exceptions.CheckFailed:

- Received a response with 5xx status code: 500

-----

Response status: 500  
Response payload: `{"code":13, "message":"can't get root team: sql: no rows in result set", "details":[]}`

Run this cURL command to reproduce this response:

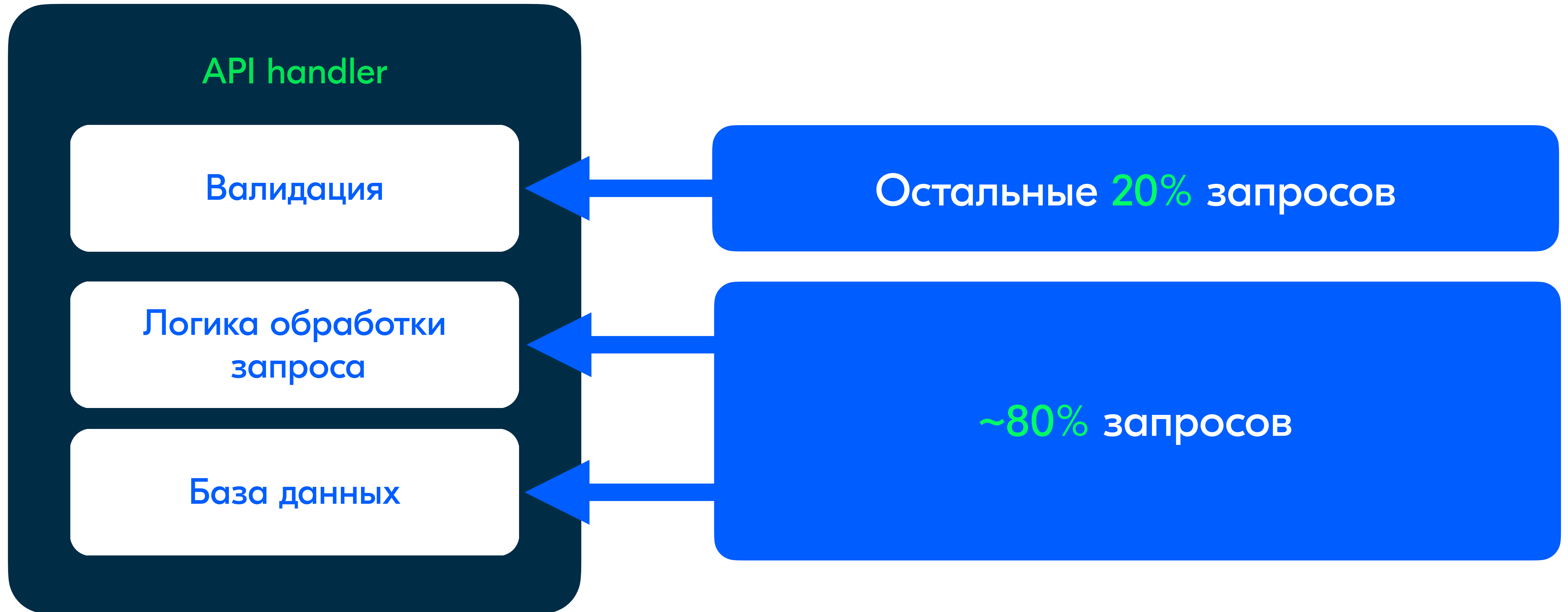
**CENSORED**

Tags: hypothesis

Categories: Product defects

Severity: normal

# Проверяем не только слой валидации





# А теперь, на Coffee Time!



ozon{tech}



@RUKO12



Schemathesis:  
property-based  
тестирование для  
API схем



Что такое  
фаззинг  
и зачем он нужен?



Пример  
подключения  
schemathesis



ozon.tech

Спасибо за внимание

Косых Руслан, ведущий инженер по автоматизации тестирования