Effective technologies

HEISENBUG 2024 Autumn

Нагрузка без стресса

Гонускус Валентина

2024, октябрь

Отправная точка

- ✓ Вы автоматизатор/разработчик в тестировании
- ✓ НТ — новая задача
- ✓ Знания теории НТ

История фейлов построения НТ процесса

АнтиСтрессовые рекомендации

1. Автоматизируй запуск замеров
2. Делегируй рутину «Робогозину»
3. Экономь время на отчетах
4. Доверяй, но проверяй результат
5. Заботься о тестовых данных
6. Keep calm и пей genetically modified чайный гриб

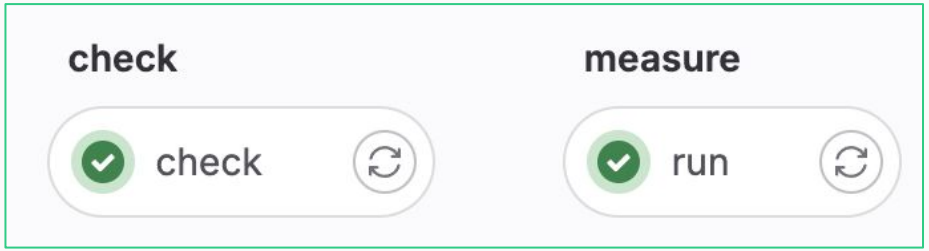
Мнение спикера может отличаться от вашего:
Обязательно напишите в комментариях свои альтернативные варианты

Автоматизируй запуск | Новичок

✓ ручной запуск

```
#!/bin/bash
./k6 run script.js
```

✓ запуск в CI/CD

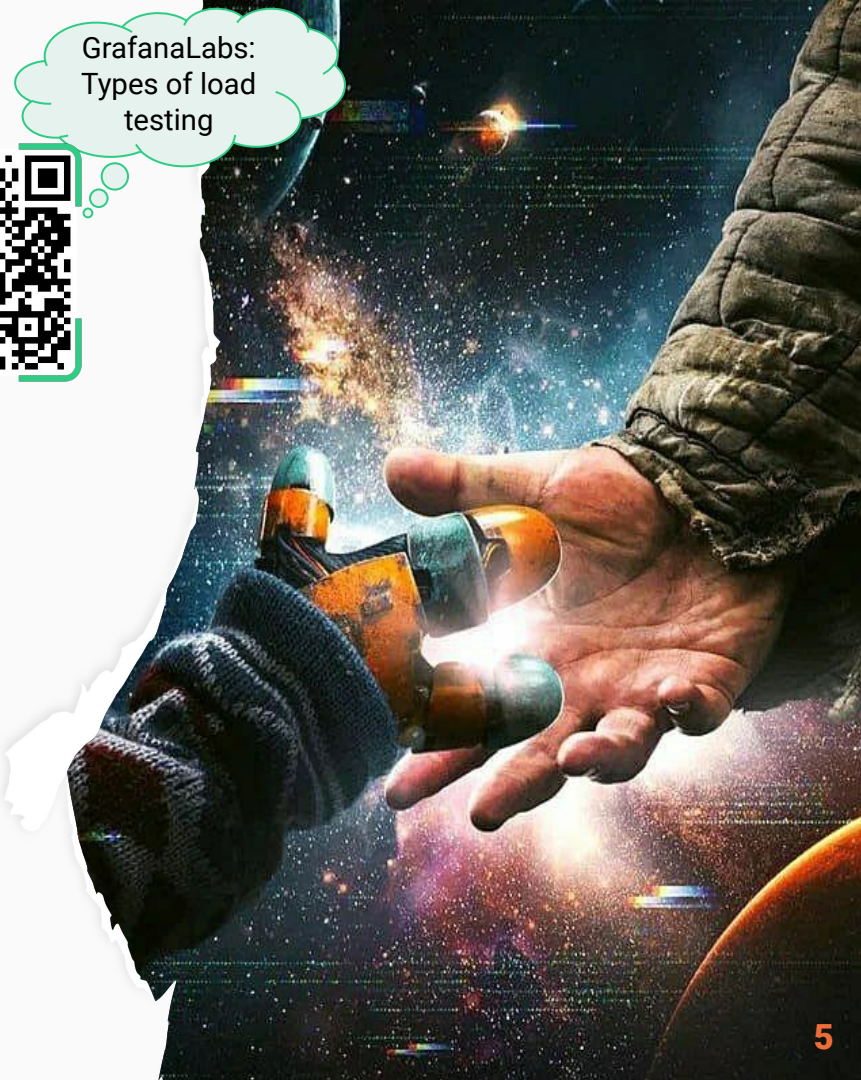


Автоматизируй запуск | На опыте

GrafanaLabs: Types of load testing

✓ конфигурируемый запуск

```
variables:
  SCRIPT:
    description: "Сценарий нагрузки"
    options:
      - "regress"
      - "service-auth"
      - "feature-N"
  CONFIG:
    description: "Профиль нагрузки"
    options:
      - "config/smoke"
      - "config/regular"
      - "config/stress"
```



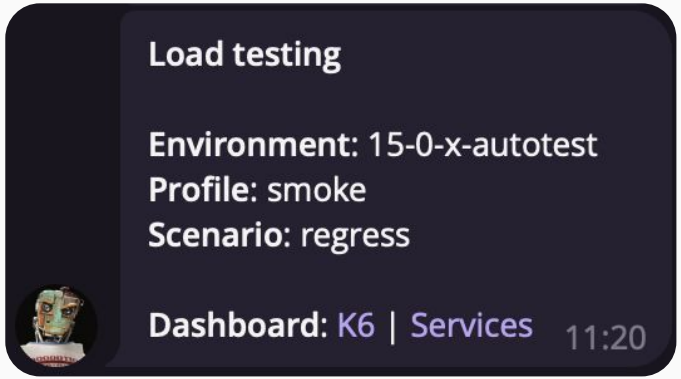
Автоматизируй запуск | Продвинутый

Что еще можно улучшить:

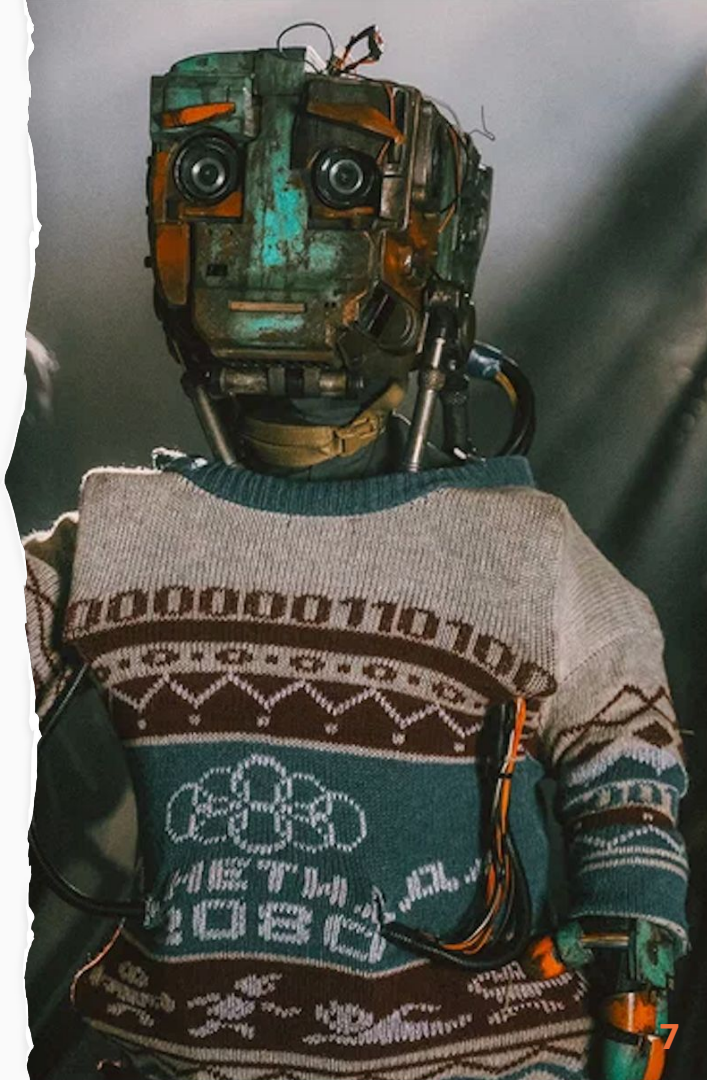
- регулярный запуск pipeline со средней нагрузкой
- НТ при динамическом масштабировании
- перспективы тестирования **выносливости, стресса и восстановления** системы/продукта

Делегируй рутину | Новичок

```
node notification.js $TG_CHAT $TG_TOKEN
```



Код нотификаций



Делегируй рутину | На опыте

Robogozin admin

🔴 Load testing interrupted

Environment: 15-0-x-autotest
Profile: smoke
Scenario: regress

Dashboard: K6 | Services 11:37

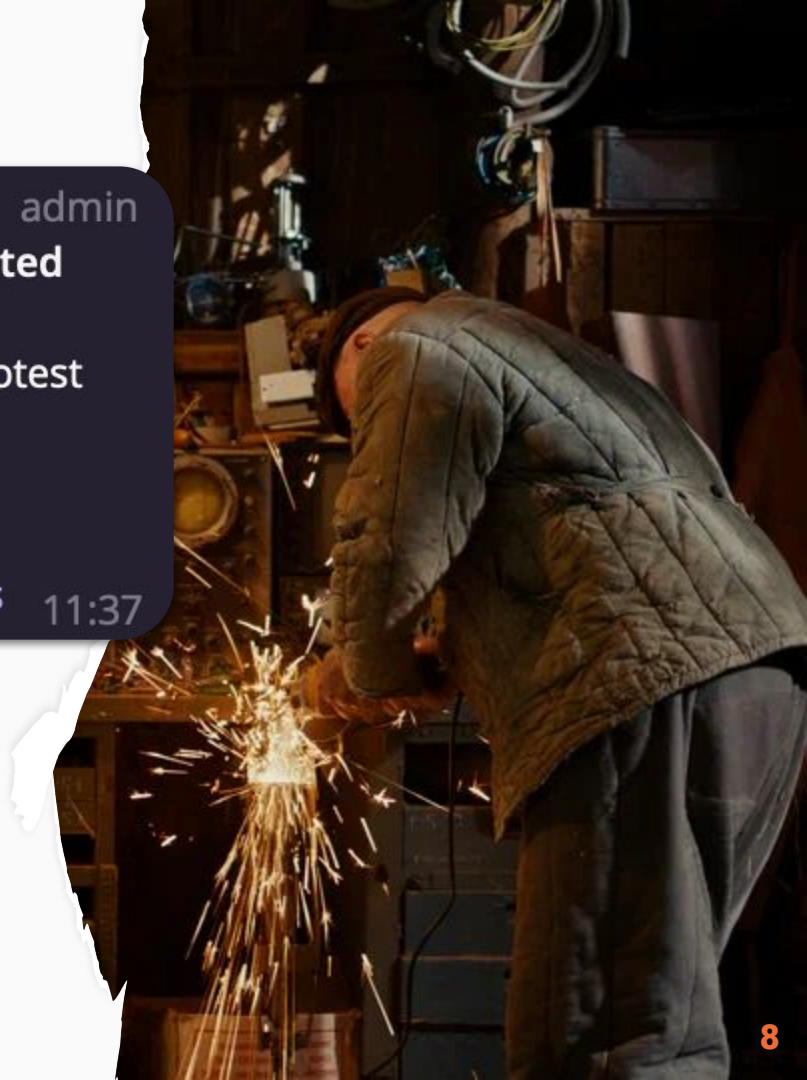
Robogozin

🟠 Load testing

Environment: 15-0-x-autot
Profile: smoke
Scenario: regress

Thresholds were crossed:
🚩 auth-api
🚩 multilang:

Dashboard: K6 | Services 11:39



Делегируй рутину | Продвинутый

Что еще можно улучшить:

- прокачай простейшего бота до chatbot-а, который по команде соберет нужную информацию

Экономь время на отчетах | Новичок

✓ стандартный настраиваемый отчет k6

```
  checks.........................: 100.00% ✓ 9        × 0
  data_received..................: 35 kB   17 kB/s
  data_sent......................: 2.6 kB  1.3 kB/s
  http_req_blocked...............: p(90)=39.55ms  p(95)=59.33ms  p(99)=75.15ms
  http_req_connecting............: p(90)=10.87ms  p(95)=16.31ms  p(99)=20.66ms
× http_req_duration..............: p(90)=412.93ms p(95)=433.2ms  p(99)=449.42ms
    { expected_response:true }...: p(90)=412.93ms p(95)=433.2ms  p(99)=449.42ms
✓ http_req_failed................: 0.00%   ✓ 0        × 6
  http_req_receiving.............: p(90)=6.92ms   p(95)=8ms      p(99)=8.86ms
  http_req_sending...............: p(90)=1.96ms   p(95)=2.8ms    p(99)=3.47ms
  http_req_tls_handshaking.......: p(90)=25.47ms  p(95)=38.21ms  p(99)=48.4ms
  http_req_waiting...............: p(90)=411.97ms p(95)=432.49ms p(99)=448.91ms
  http_reqs......................: 6       3.002202/s
  iteration_duration.............: p(90)=578.9ms  p(95)=583.23ms p(99)=586.7ms
  iterations.....................: 3       1.501101/s
  vus............................: 1       min=1      max=1
  vus_max........................: 1       min=1      max=1
```

Отчет | На опыте

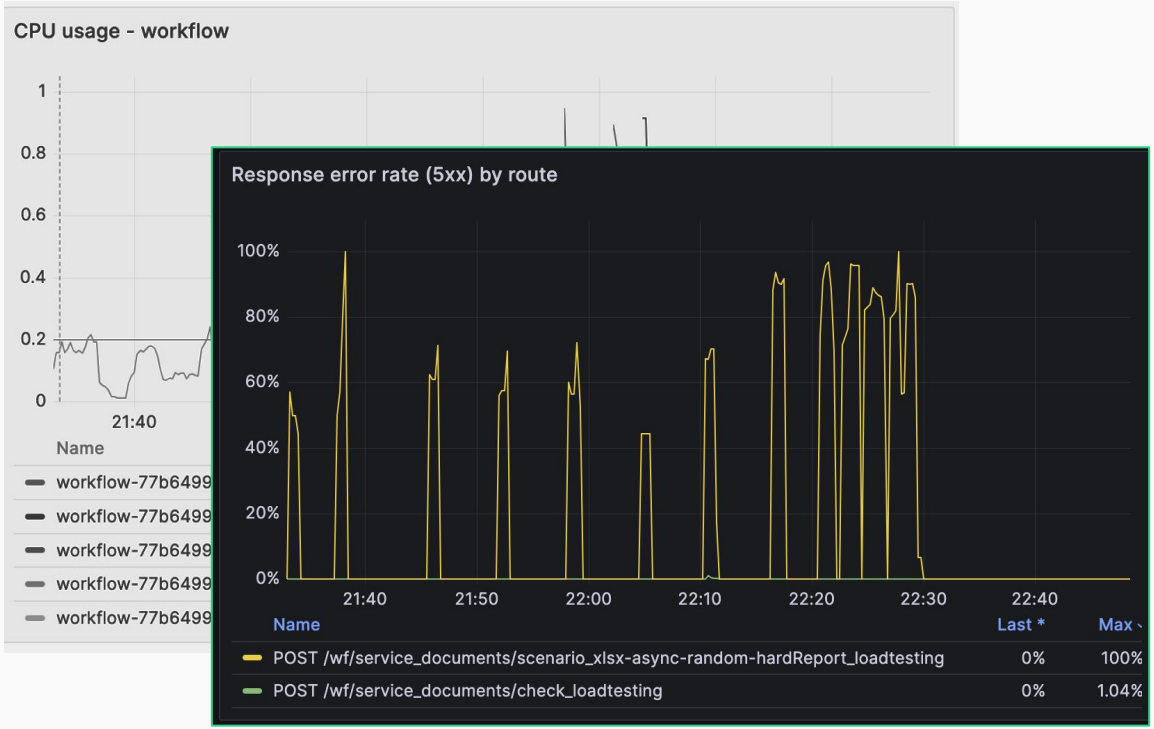
k6 → PROMETHEUS → Grafana

```
./k6 run -o experimental-prometheus-rw script.js
```

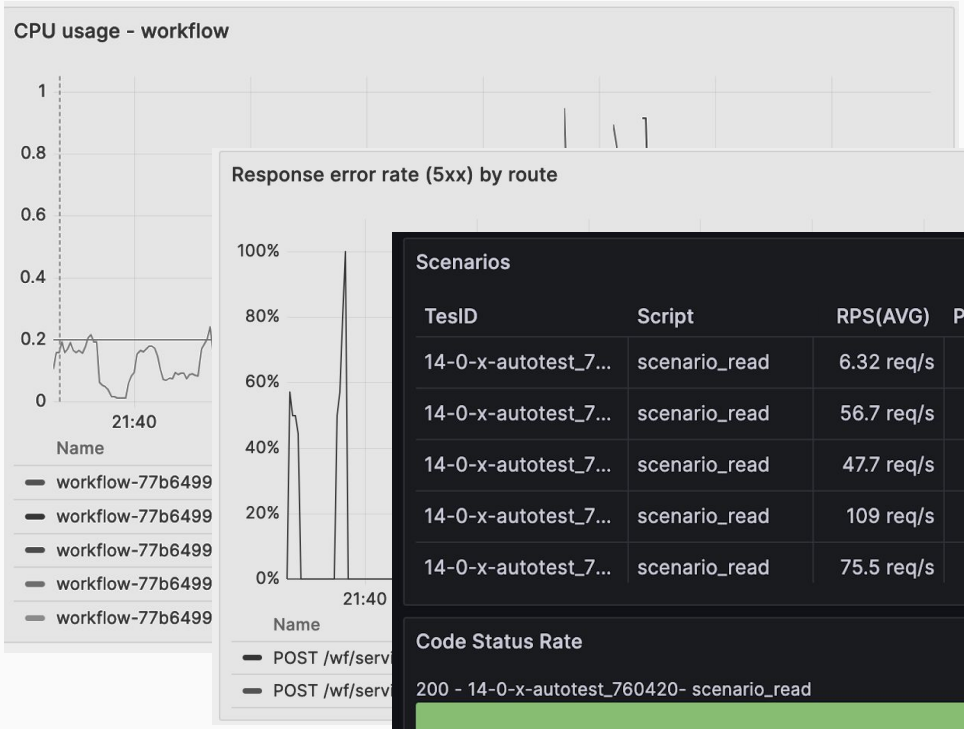
```
variables:
   K6_PROMETHEUS_RW_SERVER_URL:
    http://X.X.X.X:9090/api/v1/write
   K6_PROMETHEUS_RW_TREND_AS_NATIVE_HISTOGRAM:
    "true"
   K6_PROMETHEUS_RW_PUSH_INTERVAL:
    "1s"
```

Отчет | На опыте

Отчет | На опыте



Отчет | На опыте



Scenarios

TesID	Script	RPS(AVG)	P90 Response Time	P95 Response Time	P99 Response Time
14-0-x-autotest_7...	scenario_read	6.32 req/s	1.46 s	1.41 s	1.48 s
14-0-x-autotest_7...	scenario_read	56.7 req/s	1.70 s	1.23 s	2.02 s
14-0-x-autotest_7...	scenario_read	47.7 req/s	2.16 s	1.79 s	2.67 s
14-0-x-autotest_7...	scenario_read	109 req/s	3.15 s	2.66 s	3.75 s
14-0-x-autotest_7...	scenario_read	75.5 req/s	5.97 s	5.70 s	6.47 s

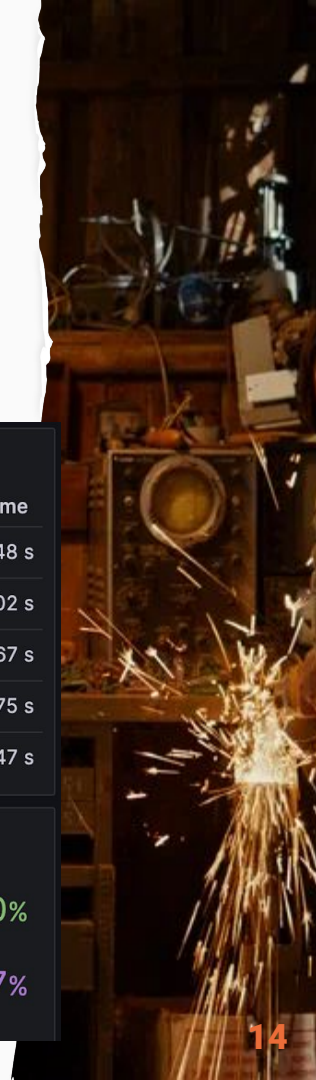
Code Status Rate

200 - 14-0-x-autotest_760420- scenario_read

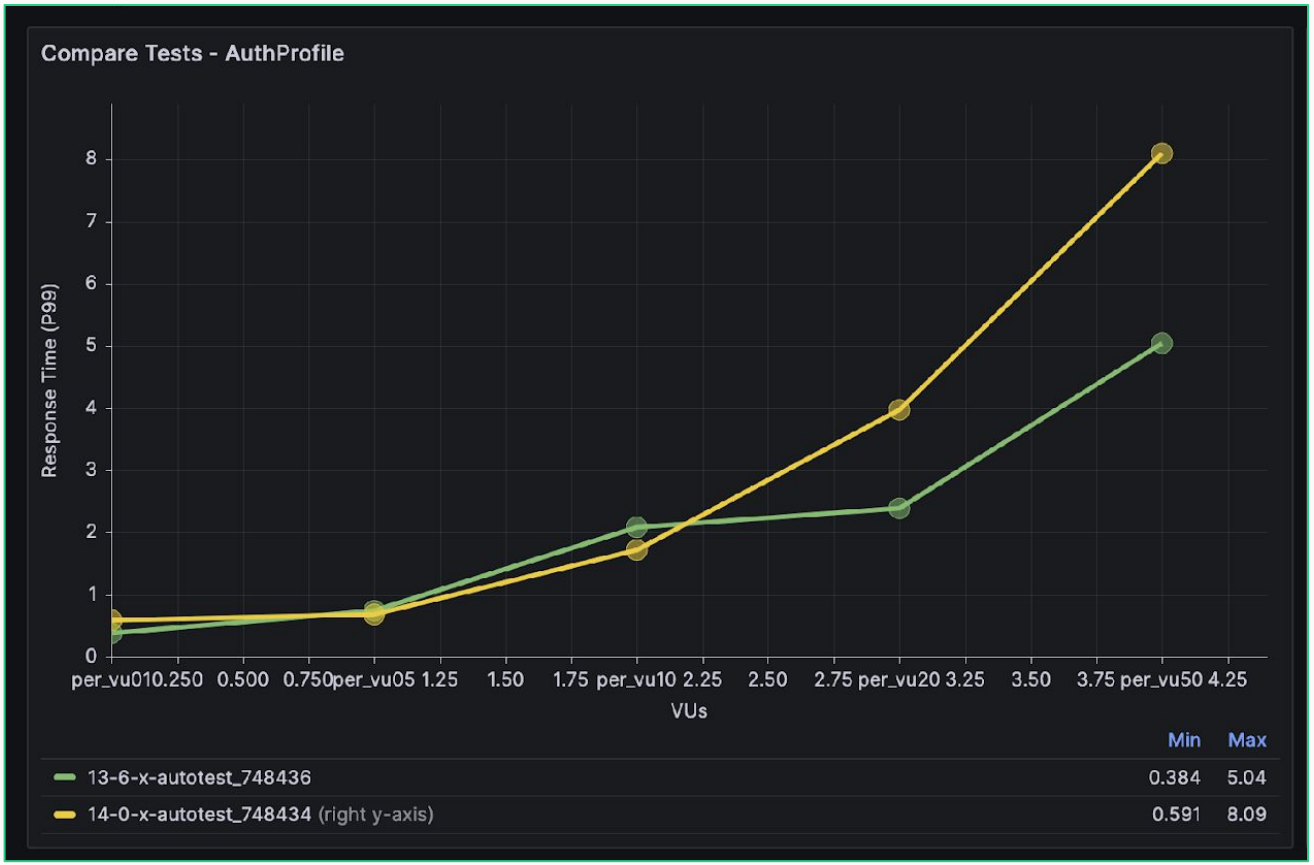
100.0%

500 - 14-0-x-autotest_760420- scenario_read

0.00367%



Отчет | Продвинутый



Пример кода dashboard

Доверяй, но проверяй результат | fail

Задача: Необходимо оценить скорость работы API

	min	avg	max
До	975.86ms	1.21s	1.32s
После	450.69ms	527.18ms	631.31ms


```
import http from "k6/http";
import { scenario } from "k6/execution";

  ...
export default function (loginArray) {
 const body = {
   login: loginArray[scenario.iterationInTest % loginArray.length],
   password: commonPassword,
 };
 http.post(`${host}/api/auth/login`, body);
}
```

Доверяй, но проверяй результат | fail

Задача: Необходимо оценить скорость работы API

	min	avg	max
До	975.86ms	1.21s	1.32s
После	450.69ms	527.18ms	631.31ms

http_req_failed: 100%

Доверяй, но проверяй результат | Новичок

```
if (
 !check(res, {"status is 200":
    (res) => res.status === 200})
) {
 fail(`Response ${url} is
    ${res.status}: ${res.body}`)
}
```

Проверки:

- код ответа
- тело
- размер ответа
- комбинации

Доверяй, но проверяй результат | На опыте

```
export const options = {
 thresholds: {
   http_req_failed: ['rate<0.01'],
   http_req_duration: ['p(95)<200'],
 },
};
```

Что еще можно улучшить:

- настроить остановку скрипта при пересечении порога (критериев успеха/провала)

Доверяй, но проверяй результат

Что еще можно улучшить:

- комбинировать проверки и пороги (с возможностью остановки скрипта)
- пороги для каждого из этапов нагрузки
- настроить разные уровни реагирования (приемлемо / тревожно / неправильно)

Заботься о тестовых данных | fail

Задача: Оценить скорость выполнения запроса на чтение данных.

	p(90)	p(95)	p(99)
Замер №1	1.44s	1.65s	1.82s
Замер №2	1.56s	1.69s	1.91s
Замер №3	1.88s	1.91s	2.01s
Замер №4	1.99s	1.95s	2.11s
Замер №5	2.01s	2.1s	2.18s

Заботься о тестовых данных | На опыте

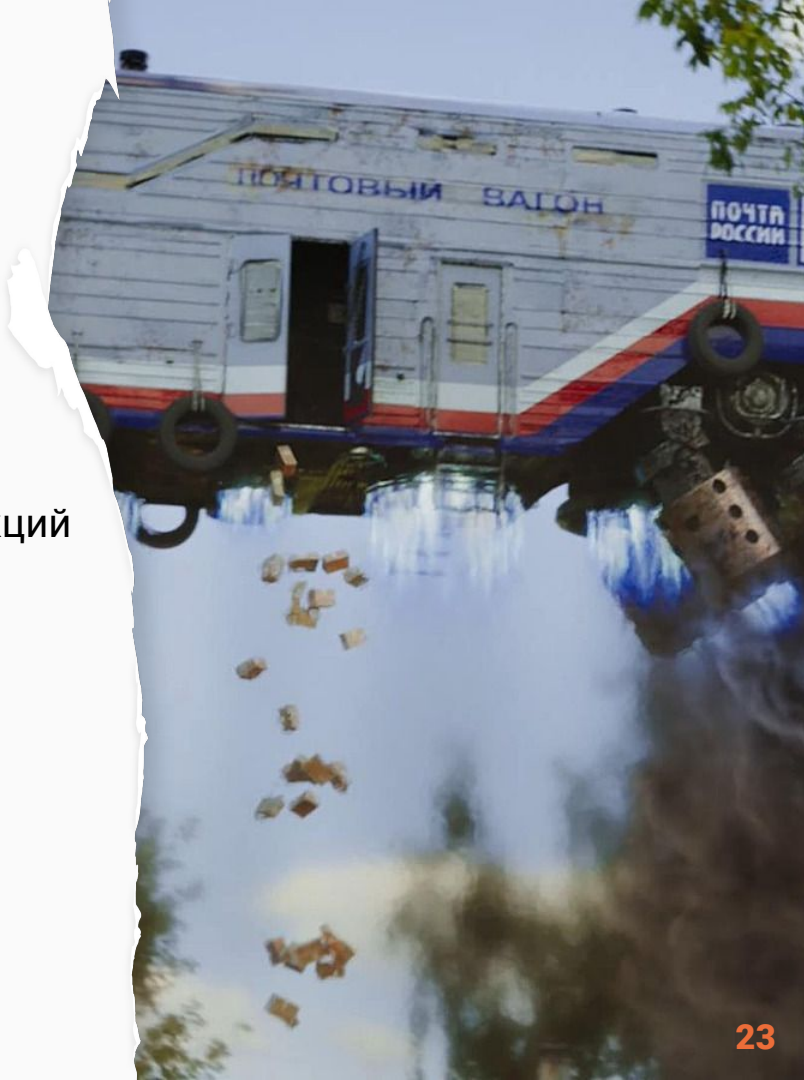
Перед стартом замеров:

- наличие и достаточность учетных записей
- наличие данных на чтение и соответствие ожидаемому объему (в т.ч. их отсутствие)
- генерация данных

Заботься о тестовых данных | Продвинутый

Что еще можно улучшить:

- проверка наличия настроек и др. конфигураций данных
- очистка отдельных таблиц/коллекций
- сброса бд в исходное состояние
- импорт бд перед замерами



АнтиСтрессовые рекомендации

1. Автоматизируй запуск замеров
2. Делегируй рутину «Робогозину»
3. Экономь время на отчетах
4. Доверяй, но проверяй результат
5. Заботься о тестовых данных
6. Keep calm и пей genetically modified чайный гриб

Мнение спикера может отличаться от вашего:
Обязательно напишите в комментариях свои альтернативные варианты

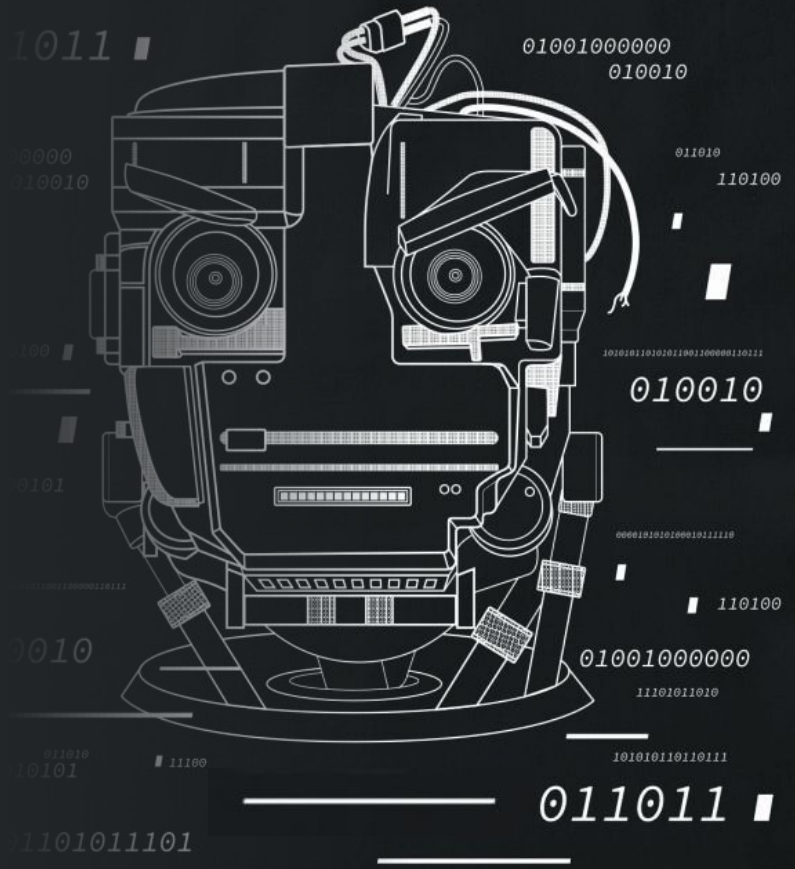
Буду рада диалогу)

@cmetikova

Узнать больше об
Effective Technologies

Рекомендую канал
QA — Load & Performance

@qa_load



При оформлении презентации использована эстетика сериала «Кибердеревня»