

Николай Ельцов

Performance & Reliability Engineer

n.eltsov.i@gmail.com · +7 965 346-98-34

Удалённо

Шесть лет занимаюсь нагрузочным тестированием и проверкой отказоустойчивости: банк, промышленная платформа на Kubernetes, WAF для API. Строю стенды и автоматизацию с нуля, проектирую профили нагрузки по данным продукта, ломаю системы под нагрузкой и довожу находки до фикса в коде или в чарте. Хочу делать то же самое для платформы частного и гибридного облака.

ОПЫТ

Wallarm · Senior Performance Engineer · контракт, удалённо

сентябрь 2025 - март 2026

Нефункциональное тестирование WAF-ноды под требования клиентов и capacity planning для них.

- Собрал платформу перф-тестов с нуля: автопровивинг стенда в AWS через Bash и AWS CLI, параметрические сценарии k6 (payload, HTTP / HTTPS, keep-alive, до 10 000 RPS), автоочистка ресурсов даже при упавшем тесте. 100+ сценариев за прогон, стенд за 10–15 минут вместо нескольких часов.
- Тесты надёжности ноды под нагрузкой: breakpoint до точки деградации по p99 и доле 5xx, spike с замером времени восстановления, soak на сутки для поиска утечек, kill процесса WAF, остановка upstream, обрыв связи с управляющим облаком через iptables, вывод ноды из-под балансировщика. Для каждого сценария фиксировал окно недоступности и вывод для клиента.
- Нашёл, что генератор нагрузки сам был узким местом: k6 упирался в CPU и терял итерации, заявленный RPS не совпадал с реальным. Добавил контроль валидности прогона и перенёс управление тестами из GitLab-пайплайнов в веб-платформу на Go и React с параметрами RPS, размера и типа трафика.
- Нагрузочные тесты в GitLab CI по расписанию и на релиз с thresholds как quality gate; результаты в Victoria Metrics с меткой версии, сравнение релизов и алерты в Grafana.
- Сделал Capacity Planner: веб-приложение для подбора инстансов AWS / GCP / Azure под прогноз нагрузки. Формулы строил на своих замерах RPS на vCPU по типу трафика и проверял прогноз реальным прогоном. Next.js, PostgreSQL, Cloud Run.

Стек: k6, Bash, AWS, Kubernetes, Helm, Go, TypeScript, Victoria Metrics, Grafana, GitLab CI, PostgreSQL.

Bimeister · Lead / Senior Performance Testing Engineer · удалённо

ноябрь 2022 - сентябрь 2025

Промышленная платформа, on-prem-инсталляции в Kubernetes у заказчиков.

- Строил профили нагрузки из логов и продуктовой аналитики заказчика: состав операций, их веса, пиковые часы. Критерии и SLA согласовывал с аналитиками до теста; по ключевому заказчику работал по ПМИ с протоколом испытаний.
- Проверял отказоустойчивость под нагрузкой: удаление подов, drain и cordon нод, масштабирование реплик, failover PostgreSQL с замером времени переключения, остановка брокера, медленная зависимость с каскадом через пул соединений, восстановление из бэкапа с проверкой RTO.
- Решил критические проблемы производительности у ключевого заказчика на его инфраструктуре с ограниченным доступом - контракт продлили.
- Инфраструктура для 4–8 параллельных потоков тестов в изолированных окружениях и хранилище результатов для автоматического сравнения релизов: цикл перф-тестов быстрее в 4 раза, релизное тестирование короче на 40%. Ночные прогоны с автоостановкой по порогу ошибок.

Стек: k6, JMeter, Kubernetes, Helm, Docker, PostgreSQL, Grafana, GitLab CI.

Банк «Открытие» · QA / Performance Engineer · Москва

март 2020 - ноябрь 2022

- По результатам нагрузки инициировал и провёл 7 production-оптимизаций - система выдержала рост с 200 до 4000 пользователей.
- Перевёл стенды нагрузочного тестирования в Kubernetes на своих Helm-чартах: новый стенд за 10 минут вместо 3 часов.
- Развернул ELK-кластер для централизованного логирования, расследовал по нему инциденты прода, где причиной была нагрузка.
- Участвовал в DR-учениях: нагрузка во время переключения на резервный контур, замер окна недоступности; отказ узла кластера приложения под нагрузкой. Тестирование по методике банка с протоколом.
- Автоматизировал 70% перф-регресса - проверка втрое быстрее.

Стек: Gatling, Locust, LoadRunner, JMeter, Kubernetes, Helm, ELK, Grafana, Python, Bash.

АО «Аэроэлектромаш» · техник → инженер · Москва

декабрь 2014 - март 2020

Испытания авиационного электрооборудования по ПМИ. Сделал мобильный диагностический стенд для расследования рекламаций, по его данным предложил три конструктивных доработки - гарантийные отказы снизились на 40%.

СВОИ ПРОЕКТЫ

- **Reliability & Load-Tools Lab** - собрал в одном месте метод и тулинг для тестов отказоустойчивости: агент инъекции отказов с автооткатом, исполнитель chaos-экспериментов (гипотеза, воздействие, откат, вердикт), генератор нагрузки с замером окна недоступности, сравнение k6 / Locust / Gatling на одном сценарии. Python 3.
- **bimAgent** - CI-агент для триажа упавших тестов через LLM: локальный Ollama как baseline, эскалация в Anthropic API на сложные случаи. TypeScript, Node 20.
- **DevBrain** - desktop-инструмент с локальной базой знаний для разработчика: RAG на ChromaDB и Ollama, MCP-сервер на 39 инструментов для Claude Code, A/B-тестирование промптов. TypeScript, Electron.

НАВЫКИ

Нагрузка и надёжность: k6 · профили нагрузки · breakpoint / spike / soak · failover и RTO · инъекция отказов · tc netem · stress-ng · Locust · Gatling · JMeter

Инфраструктура: Kubernetes · Helm · Docker · Linux · GitLab CI · AWS · GCP · Azure

Observability: Grafana · Victoria Metrics · Prometheus · ELK

Языки: Python · Go · Bash · TypeScript · SQL