Proofline

BEISPIELDATEN · BEISPIELBERICHT

Last-, API- & Security-Testbericht

Example Pay API v2 — fiktives Fintech-System

Kunde	Example Pay Ltd (fiktiv)
Umfang	48 REST-Endpunkte, Staging-Umgebung
Testarten	Last, Stress, Spike · funktional & Contract · OWASP API Top 10
Version	1.0 · Muster

Alle Systemnamen, Befunde und Zahlen in diesem Dokument sind fiktiv und als BEISPIELDATEN gekennzeichnet. Das Dokument zeigt Aufbau und Tiefe eines Proofline-Berichts – kein Kundenergebnis.

1. Management-Zusammenfassung

~560 VUs	2	9	6 / 10
max. Dauerlast innerhalb SLO	Befunde mit hohem Schweregrad	Befunde insgesamt	betroffene OWASP-API-Kategorien
BEISPIELDATEN	BEISPIELDATEN	BEISPIELDATEN	BEISPIELDATEN

Gesamtbewertung

Die API ist für den geplanten Launch-Traffic von 800 gleichzeitigen Nutzern **nicht bereit**. Zwei Probleme müssen zuerst behoben werden: eine fehlerhafte Autorisierung auf Objektebene, die Kontodaten anderer Mandanten offenlegt (F-01), und ein erschöpfter Connection-Pool, durch den die p95-Latenz bei etwa 600 virtuellen Nutzern das SLO von 400 ms überschreitet (F-02).

Die drei wichtigsten Maßnahmen

1. Mandanten-Prüfung im Accounts-Service erzwingen (F-01). Aufwand: S.
2. DB-Pool vergrößern und pgbouncer einsetzen; Re-Test bei 800 VUs (F-02). Aufwand: M.
3. Rate Limiting pro Key für Webhooks mit 429 + Retry-After (F-03). Aufwand: S.

Release-Empfehlung

Go-live, nachdem F-01 und F-02 behoben und per Re-Test bestätigt sind. Mittlere Befunde können in den ersten zwei Sprints folgen.

2. Umfang, Umgebung & Methodik

PUNKT	DETAIL
Testobjekt	Example Pay API v2 (REST, OpenAPI 3.1)
Umgebung	Staging, produktionsgleich dimensioniert (3 App-Knoten, verwaltetes PostgreSQL)
Testdaten	Synthetisch: 2 Mandanten, 5.000 Konten, 200.000 Transaktionen
Tools	`k6 · Grafana · Postman/Newman · Schemathesis · Burp Suite`
Zeitraum	10 Arbeitstage (Beispiel)
Nicht im Umfang	Mobile Apps, externer Kartenprozessor, Infrastruktur-Penetrationstest

Lastmodell

SZENARIO	PROFIL	ABNAHMEKRITERIEN
Last	Anstieg auf 200 VUs, 10 Min. halten	p95 < 400 ms · Fehler < 1 %
Stress	+100 VUs alle 2 Min. bis zum Ausfall	
Spike	200 → 800 VUs in 30 s	
Soak	300 VUs für 4 h	

Schweregrade

`Hoch` Datenoffenlegung oder SLO-Verletzung bei erwarteter Last · `Mittel` Beeinträchtigung oder mit Aufwand ausnutzbar · `Niedrig` Hygiene, geringe Auswirkung

3. Performance-Ergebnisse

Abb. 1 – Latenz vs. virtuelle Nutzer, POST /v1/payments (Stresstest) BEISPIELDATEN

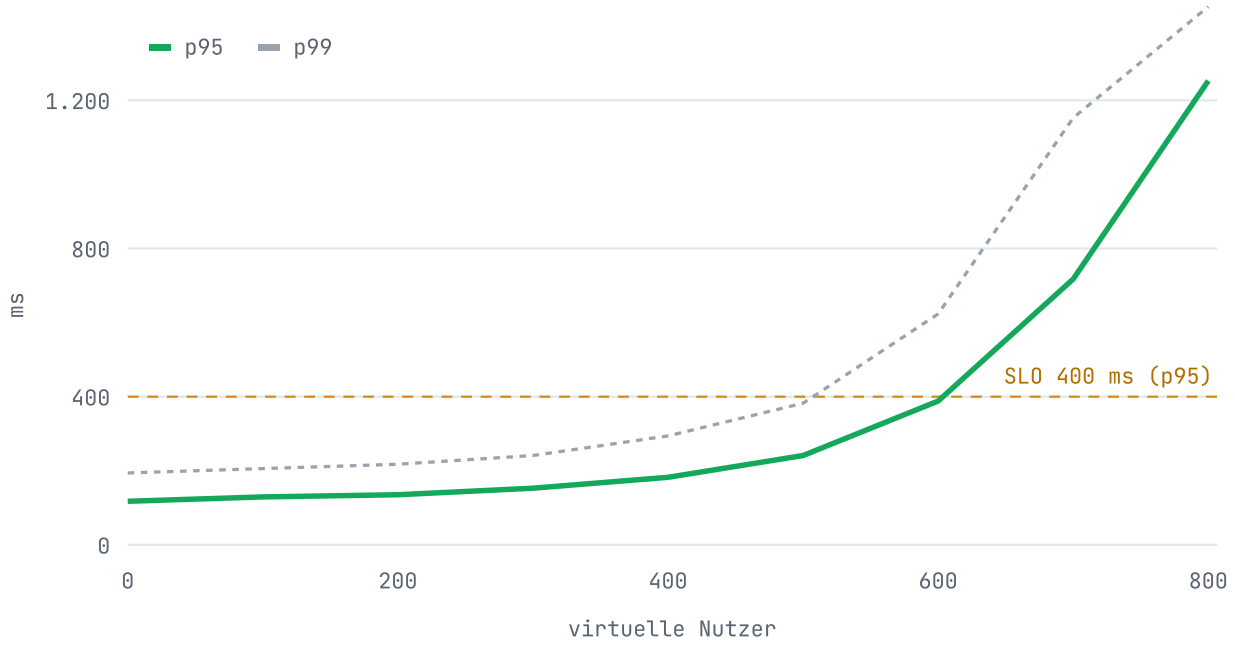
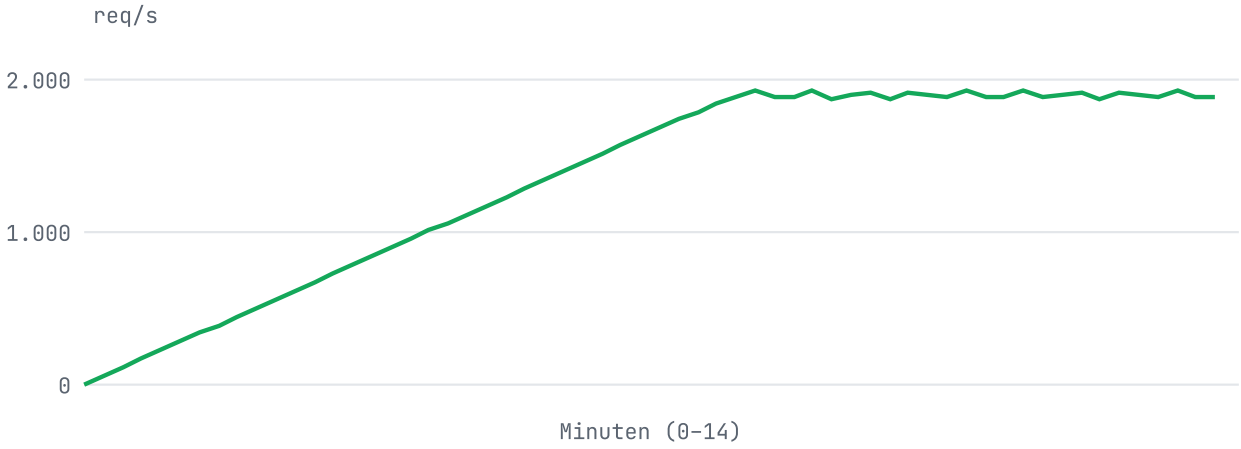


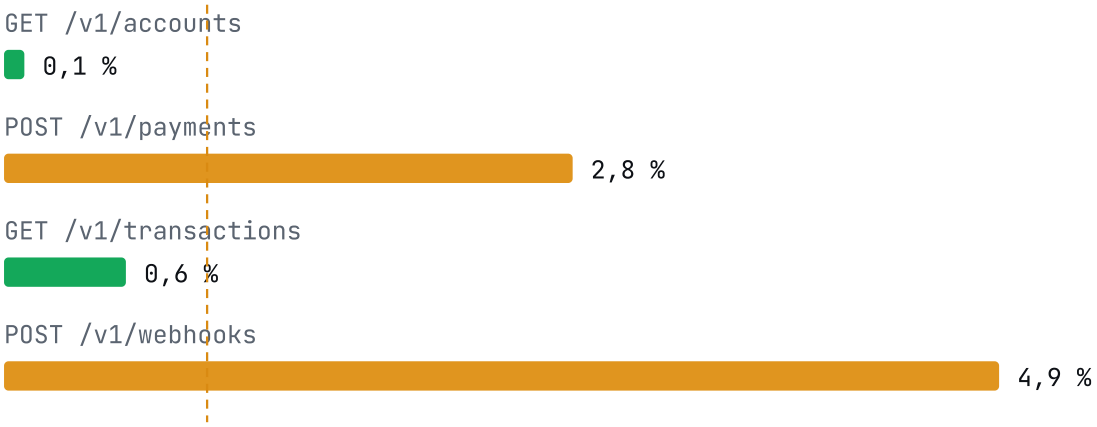
Abb. 2 – Durchsatz während des Stresstests BEISPIELDATEN



Die p95-Latenz bleibt bis ~400 VUs stabil und steigt dann stark an. Das SLO wird bei ~600 VUs verletzt (p95 390 → 720 ms). Der Durchsatz stagniert bei ~1.900 req/s – typisch für eine gesättigte Ressource, nicht für CPU-Limits (App-CPU maximal 58 %).

3. Performance-Ergebnisse (Forts.)

Abb. 3 – Fehlerrate pro Endpunkt bei 600 VUs (Schwellenwert 1 %) BEISPIELDATEN



SZENARIO	P95	P99	FEHLER	ERGEBNIS
Last (200 VUs)	135 ms	210 ms	0,1 %	Bestanden
Stress (bis Ausfall)	1.250 ms	2.030 ms	3,4 %	Fehlgeschlagen bei ~600 VUs
Spike (200→800)	1.480 ms	2.600 ms	6,2 %	Fehlgeschlagen
Soak (4 h, 300 VUs)	160 ms	260 ms	0,2 %	Bestanden, kein Speicherzuwachs

Alle Werte sind BEISPIELDATEN.

4. Befundübersicht

ID	SCHWEREGRAD	KATEGORIE	BEFUND
F-01	Hoch	API1:2023 BOLA	Mandantenübergreifender Lesezugriff auf GET /v1/accounts/{id}
F-02	Hoch	Last	DB-Connection-Pool bei ~600 VUs erschöpft
F-03	Mittel	API4:2023	Kein Rate Limit auf POST /v1/webhooks
F-04	Mittel	Contract	`amount` als String geliefert, Spezifikation: number
F-05	Mittel	API2:2023	Refresh-Tokens werden bei Passwortänderung nicht widerrufen
F-06	Mittel	Last	Idempotenz-Schlüssel bei Retry-Sturm ignoriert
F-07	Niedrig	API8:2023	Ausführliche Stacktraces in 500-Antworten
F-08	Niedrig	QA	Datumsauswahl akzeptiert 31. Februar
F-09	Niedrig	Contract	Nicht dokumentierter Status 409 bei POST /v1/payments

Abdeckung OWASP API Security Top 10 (2023)

API1 BOLA	F-01	API6 Sensible Geschäftsabläufe	F-06
API2 Broken Auth	F-05	API7 SSRF	geprüft, keine
API3 Autorisierung auf Property-Ebene	geprüft, keine	API8 Fehlkonfiguration	F-07
API4 Ressourcenverbrauch	F-03	API9 Bestandsverwaltung	F-09
API5 Autorisierung auf Funktionsebene	geprüft, keine	API10 Unsichere API-Nutzung	geprüft, keine

5. Befunddetails (Auszug)

F-01 · Hoch Mandantenübergreifender Lesezugriff auf GET /v1/accounts/{id}

Auswirkung: Jeder authentifizierte Nutzer kann durch Ändern der Konto-ID Kontostände und IBANs anderer Mandanten lesen.

```
# Als Mandant A authentifiziert
curl -s -H "Authorization: Bearer $TOKEN_TENANT_A" \
  https://api.example.test/v1/accounts/acc_B_7731 | jq '.tenant_id'
# erwartet: 404 · tatsächlich: "tenant_b"
```

Behebung: Jede Kontoabfrage über die Mandanten-ID aus dem Token einschränken; bei fremden IDs 404 zurückgeben. Negativen Contract-Test in die CI aufnehmen.

F-02 · Hoch DB-Connection-Pool bei ~600 VUs erschöpft

Beleg: Bei ~600 VUs steigt die Pool-Wartezeit auf 480 ms (p95), während die DB-CPU bei 41 % bleibt. 20 von 20 Verbindungen sind während des gesamten Plateaus belegt.

Behebung: Pool-Größe auf 50 pro Knoten erhöhen, pgbouncer im Transaction-Mode einsetzen, Stress- und Spike-Test bei 800 VUs wiederholen.

F-03 · Mittel Kein Rate Limit auf POST /v1/webhooks

Beleg: Burst von 5.000 Requests in 10 s angenommen; 4,9 % mit 5xx beantwortet, kein 429.

Behebung: Token-Bucket pro Key; 429 mit Retry-After zurückgeben.

6. Maßnahmenplan & Wiederholung der Tests

PRIORITÄT	BEFUNDE	AUFWAND	WANN
P1	F-01, F-02	S + M	Vor dem Launch
P2	F-03, F-05, F-06	S, S, M	Sprint 1
P3	F-04, F-07, F-08, F-09	S	Sprint 2

Ergebnisse in Ihrem Repository

```
tests/
  load/      checkout-load.js  stress.js  spike.js  soak.js
  api/       newman-collection.json  schemathesis.yml
  security/  bola-checks.sh  ratelimit-burst.js
.github/workflows/perf-smoke.yml
```

Wiederholen

```
k6 run -e BASE=https://staging.example.test tests/load/stress.js
newman run tests/api/newman-collection.json
schemathesis run openapi.yaml --checks all
```

Ein Re-Test der P1-Befunde nach Deployment der Fixes ist enthalten. Debriefing: 60 Minuten mit Ihrem Engineering-Team.