

Root Cause Analysis (RCA)

Service: Cities API (`/api/cities`)

IP Address: 34.198.161.179

Environment: VM-based Node.js service (non-containerized)

Date of Investigation: 14 February 2026 (00:10 - 00:20)

Investigator: SRE Team

1. Summary

Tim SRE menerima ownership atas service yang sebelumnya mengalami kegagalan dan tidak dapat diperbaiki oleh tim development.

Saat dilakukan investigasi pada VM, service berada dalam kondisi running. Untuk memastikan stabilitas dan memastikan tidak ada kegagalan aktif, dilakukan load testing menggunakan autocannon.

2. Investigation Performed

Karena keterbatasan waktu investigasi, langkah yang dilakukan berfokus pada validasi operasional service:

Load Testing

```
npx autocannon -c 10 -d 5 --renderStatusCodes  
http://localhost:3000/api/cities
```

```
ubuntu@ip-10-0-1-141:~$ npx autocannon -c 10 -d 5 --renderStatusCodes http://localhost:3000/api/cities
Running 5s test @ http://localhost:3000/api/cities
10 connections
```

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	0 ms	2 ms	6 ms	7 ms	2.13 ms	1.57 ms	35 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	10,023	10,023	15,727	15,919	13,800.8	2,492.14	10,018
Bytes/Sec	2.21 MB	2.21 MB	3.46 MB	3.5 MB	3.04 MB	548 kB	2.2 MB

Code	Count
200	69001

```
Req/Bytes counts sampled once per second.
# of samples: 5

69k requests in 5.03s, 15.2 MB read
npm notice
npm notice New minor version of npm available! 11.6.2 -> 11.10.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.10.0
npm notice To update run: npm install -g npm@11.10.0
npm notice
ubuntu@ip-10-0-1-141:~$
```

Hasil:

- 69,000+ requests dalam 5 detik
- 100% HTTP 200 responses
- Tidak ada non-2xx responses
- Average latency: ~2 ms
- Throughput: ~13k req/sec

Hasil tersebut menunjukkan:

- Service berjalan dengan normal
- Tidak terdapat error aktif
- API mampu menangani concurrent traffic tanpa kegagalan

3. Findings

1. Service dapat diakses melalui `localhost:3000`.

2. Endpoint `/api/cities` merespons secara konsisten dengan HTTP 200.
3. Tidak ditemukan indikasi kegagalan aktif berdasarkan pengujian beban ringan.
4. Environment tidak menggunakan Docker; service kemungkinan berjalan langsung sebagai Node.js process di VM.

4. Root Cause Assessment

Karena tidak ditemukan kegagalan aktif saat investigasi dilakukan, root cause insiden sebelumnya tidak dapat direproduksi atau diverifikasi secara langsung.

Berdasarkan karakteristik sistem (Node.js service di VM tanpa container isolation), kemungkinan penyebab insiden sebelumnya dapat meliputi:

- Misconfiguration environment variable
- Dependency connectivity issue (misalnya database tidak reachable)
- Service crash akibat unhandled exception
- Resource exhaustion (memory/disk)
- Service restart tanpa monitoring

Namun, saat pengujian dilakukan, service berada dalam kondisi sehat dan stabil.

5. Current Status

Service telah divalidasi dan dinyatakan:

- Running
- Stable
- Able to handle concurrent requests
- Returning consistent 2xx responses

6. Preventive Recommendations

Untuk meningkatkan reliability jangka panjang, direkomendasikan:

1. Menambahkan process manager (PM2/systemd) jika belum ada
2. Menambahkan monitoring untuk:
 - Error rate (5xx)
 - Memory & CPU usage
 - Service restart count
3. Menambahkan log retention policy
4. Jika menggunakan Docker di masa depan, tambahkan housekeeping job untuk image/cache cleanup

7. Conclusion

Pada saat takeover dan investigasi dilakukan, service dalam kondisi sehat dan stabil. Load testing menunjukkan tidak ada kegagalan aktif.

Walaupun root cause insiden sebelumnya tidak dapat direproduksi, service saat ini telah diverifikasi dan dinyatakan operasional.