

FM-KED-007 — 502 Bad Gateway Error (Application Unreachable)

Severity: S1 — Critical

Recovery Class: B — Standard Recovery

Covered by Monthly Support: Yes (known causes only)

Description

Nginx returns a **502 Bad Gateway** error because the Django application backend becomes unreachable.

In the majority of observed cases, this is caused by **Out Of Memory (OOM)** conditions on the virtual machine hosting the application worker, leading to termination of Gunicorn or equivalent application processes.

Typical Symptoms

- HTTP 502 responses from Nginx
 - Application intermittently unavailable
 - Gunicorn workers restarting or disappearing
 - Kernel logs indicating OOM events
-

Primary Root Cause

- Application requests producing excessive memory usage
- Large datasets loaded into memory
- Insufficient RAM on the worker virtual machine

When memory limits are exceeded, the operating system terminates the application process, leaving Nginx without a valid upstream.

Diagnostic Checklist

Confirm OOM Condition

```
dmesg | grep -i oom  
journalctl -k | grep -i kill
```

Check Available Memory

```
free -h
```

Verify Application Server Status

```
systemctl status gunicorn
```

Recovery Options

Apply one or more of the following, depending on constraints.

Option 1: Reduce Request Scope

- Apply stricter filters to API requests
- Limit requested date ranges
- Reduce number of portfolios, instruments, or entities per request
- Avoid bulk data retrieval in a single call

This reduces memory pressure at the application level.

Option 2: Increase RAM on Worker Virtual Machine

- Increase memory allocation on the worker VM
- Restart application services after resizing
- Verify stability under previous load

This addresses the issue at the infrastructure level.

Escalation and Unknown Issues

If the issue persists after:

- request scope reduction, and
- sufficient memory allocation

then the incident is classified as an **unknown issue**.

Such cases require investigation, profiling, or architectural analysis and are **not covered** by the standard monthly support allocation.

Preventive Notes

- Avoid unbounded API queries
 - Monitor memory usage trends
 - Define safe defaults and limits at API level
 - Prefer asynchronous processing for heavy workloads
-

Responsibility Boundary

Finmars SCSA provides best-effort diagnostics and guidance for known memory-related causes. Application design decisions and infrastructure capacity planning beyond documented scenarios require separate analysis.

Revision #1

Created 2026-01-07 14:46:29 UTC by Sergei Zhitenev

Updated 2026-01-07 14:47:07 UTC by Sergei Zhitenev