

AI-04451 GCP Btop Health Check

Owner	Ⓜ Rigel Arcayan
Tags	Project
Created time	July 22, 2026 7:40 AM

Here is an analysis and breakdown of the system resource usage shown in the screenshot, which is running btop inside a browser-based SSH terminal (Google Cloud Console's SSH-in-browser).

1. System Overview & CPU

- **Uptime:** ~39 days, 18 hours.
- **CPU Model / Speed:** Running at **2.2 GHz**.
- **Overall CPU Usage:** Extremely low at **10%**.
 - **Core 0 (C0):** 8% usage.
 - **Core 1 (C1):** 10% usage.
- **Load Average (LAV):** 0.23 (1m), 0.42 (5m), 0.53 (15m) â€” system load is very light.

2. Memory (RAM) & Swap

- **Total RAM:** 7.75 GiB
 - **Used:** 4.41 GiB (~57%)
 - **Available:** 3.33 GiB
 - **Cached:** 1.23 GiB
 - **Free:** 2.09 GiB

- **Swap Usage: 7.99 GiB Total**
 - **Used: 5.58 GiB (70%)**

⚠️ **Note:** Swap usage is elevated (**70%**). While overall RAM still has free/available capacity, heavy swap usage might indicate memory pressure spikes in the past or inactive processes paged out to swap.

3. Disks & Storage

- **Root Disk (/): 57.0 GiB Total**
 - **Used: 47.2 GiB (83%)**
 - **Status:** High usage (indicated by red bar). Disk space is running low.
- **Boot Disk (/boot): 880 MiB Total**
 - **Used: 182 MiB (21%)**

4. Network Activity

- **Interface:** ens4
- **IP Address:** 10.138.0.2 (Internal GCP VPC IP)
- **Download:**
 - **Current Rate: 397 Byte/s** (3.10 Kibps)
 - **Total Downloaded: 29.2 GiB**
- **Upload:**
 - **Current Rate: 6.85 KiB/s** (54.8 Kibps)
 - **Total Uploaded: 59.2 GiB**

5. Active Processes

Key processes currently running in the process table:

- **java (Elasticsearch):** Consuming **430 MiB** of memory.

- **python3:** Multiple instances running; one using **221 MiB**, another using **196 MiB** (bundle script/background-work).
- **otelopsc / dockerd / containerd:** Google Cloud Ops agent and Docker daemons running smoothly under **1%** CPU usage each.
- **redis-server:** Multiple instances running, using light memory footprint (~3 MB â€” 8.7 MB each).
- **htop:** The system monitor interface being viewed (using **7.3 MiB** memory).

Summary & Recommendations

1. **Root Storage Alert:** The main disk is at **83% capacity** (47.2 / 57.0 GiB). Consider cleaning up unused Docker containers/images (docker system prune) or log files to prevent running out of space.
2. **Swap Usage:** Swap usage is relatively high (**70%**). If system performance degrades, review active Java and Python process memory limits or clear swap if safe.

Htop

Here is an analysis and summary of the system performance and process activity from the provided htop terminal screenshot (accessed via Google Cloud Shell/SSH-in-browser):

1. System Summary & Metrics

- **Uptime:** 39 days, 19:01:12
- **Tasks:** 274 tasks, 1014 threads (82 kthr), 1 running
- **Load Average:** 0.38 (1m), 0.35 (5m), 0.46 (15m) â€” System load remains very light.
- **CPU Usage:**
 - Core 0: **21.9%**
 - Core 1: **20.6%**
- **Memory (RAM) Usage:** 4.11 GiB used out of 7.75 GiB (~53%).

- **Swap Usage: 5.41 GiB used** out of **8.00 GiB** (~67.6%).

2. Key Active Processes

The system is configured in a tree view showing standard system management and several background microservices:

Core System Services

- **/sbin/init (PID 1):** System init process with high cumulative CPU time (~43:23).
- **System Daemons:** systemd-journald, systemd-resolved, systemd-networkd, dbus-daemon, chronyd, polkitd.
- **Google Cloud OS Config Agent:** Multiple child instances of /usr/bin/google_osconfig_agent running cleanly.

Application Services & Containers

- **VS Code Server:** Processes for vscode-server running under user rarcayan (/home/rarcayan/.vscode-server/...).
- **iperf3 Server:** iperf3 --server --interval 0 running for network performance testing.
- **Docker / Containerd:** Multiple containerd-shim-runc-v2 instances running containerized tasks.
- **Redis Server:** redis-server *:6379 running under process user 999.
- **Node.js / pnpm Application Stack:**
 - Running under user ubuntu.
 - Executes a Node.js web/API service built with pnpm (pnpm --filter ./apps/server run start:prod).
 - Processes spawn child processes running node dist/main (likely a NestJS/Express production server instance) showing around 2.0% memory utilization each.

Key Takeaways

1. **Stable System Performance:** CPU and RAM usage are well within healthy boundaries, with a low overall system load.
2. **High Swap Utilization:** Swap usage is elevated at **67.6% (5.41 GiB)**. While active RAM still has over 3.6 GiB available, keeping an eye on swap is recommended to ensure paging isn't degrading service latency.