A black and white logo

Description automatically generated

Aerospike Health Check Report

prepared for

**BigBasket - Everest**

| Report Date: July 15, 2025  Prepared By: Debarun Dasgupta |
| --- |

**Table of Contents**

[Introduction 3](#_heading=h.bwc6bhfcgac8)

[Hardware & Configuration Summary 3](#_heading=h.8caiql2ycl71)

[Cluster Summary (Name - Everest) 3](#_heading=h.n2s551al7xcs)

[Namespace Summary 4](#_heading=h.26in1rg)

[Client Development Environment 4](#_heading=h.i4roobetmcjj)

[Use Cases 4](#_heading=h.r2qfqeqgn3ui)

[Deployment Architecture 5](#_heading=h.ipzcjnxkekk3)

[Sizing Estimate 5](#_heading=h.li888o5ww8u)

[Recommendations](#_heading=)6

[Observations from Health Check 6](#_heading=)

[Appendix A – Aerospike General Best Practices 9](#_heading=h.vx1s7mo13m3i)

# Introduction

Part of our commitment to you, as an Aerospike customer, is to provide an annual review of your Aerospike deployment (up to three clusters) to ensure that it is running optimally for your use-case(s). Any findings listed should be treated as an opportunity to improve the performance and reliability of your Aerospike environment.

The following key is used to identify key information provided:

/Users/barinder/Downloads/istock/check_15.png Configuration is within expected boundaries, no action required

/Users/barinder/Downloads/istock/exclamation_15.png A recommendation is made but does not require immediate action

/Users/barinder/Downloads/istock/x_15.png Configuration is outside expected boundaries, action is required

# Hardware & Configuration Summary

Following is the summary of the Hardware and Configuration in the Aerospike environment.

## Cluster Summary (Name - Everest)

1. Server Version : E-7.1.0.9

2. OS Version : Amazon Linux release 2023.6.20241031 (Amazon Linux) (6.1.112-124.190.amzn2023.x86\_64)

3. Cluster Size : 5 (i4i.8xlarge)

4. Devices : Total 80, per-node 16

5. Shmem Index Used : 558.023 GB

6. Device : Total 14.551 TB, 21.95% used (3.194 TB), 65.2%

available contiguous space (9.488 TB)

7. Usage (Unique Data) : Latest: 976.984 GB

8. Active Namespaces : 2 of 2

9. Usage : KVS,Batch,PIndex Query,SIndex Query,XDR Destination,Security, Index-on-shmem

## Namespace Summary

| **Namespace** | **Drives** | | **Disk** | | | **Replication Factor** | **Master Objects in millions** | **License Usage in GB** | **Status** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Total** | **Per-Node** | **Total in TB** | **% Used** | **% Available** |
| **bbcache** | 40 | 8 | 7.276 | 10.46 | 83.0 | 3 | 251.662 | 250.630 |  |
| **bigbasket** | 40 | 8 | 7.276 | 33.44 | 47.4 | 3 | 2867 | 726.353 |  |

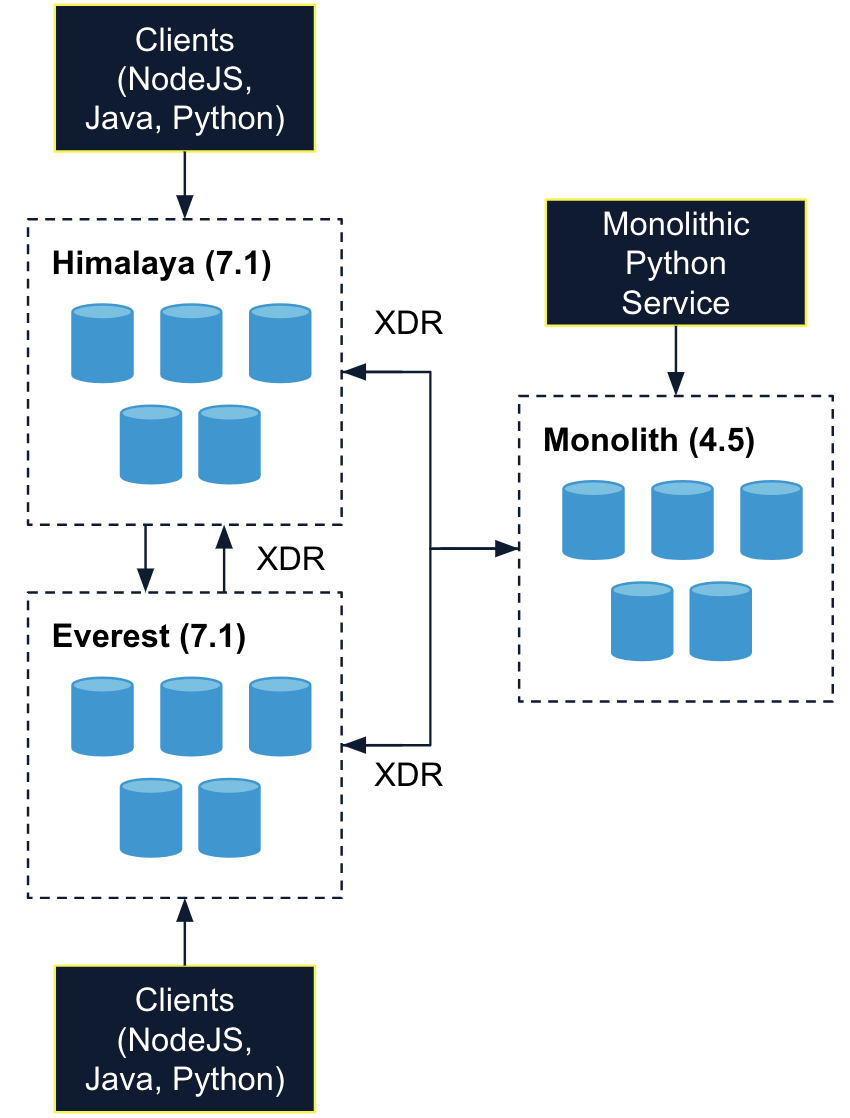
## Client Development Environment

Mix of clients like Java, NodeJS and Python are being used across multiple microservices

# Use Cases

* Product pricing information
* Catalogues of products, URLs to images, production description
* List pages, home page information
* App Metadata
* Offers

# Deployment Architecture



# Sizing Estimate

* *bbcache* namespace has 251.662 million records, occupying 10.46% of available disk
* *bigbasket* namespace is a hybrid memory namespace, having 2.867 billion records and occupying 33.44 % of total available disk space.
* Based on the analysis and assumptions of 500k TPS overall, the current configuration of the 5-node cluster should be able to handle the load.

# Recommendations

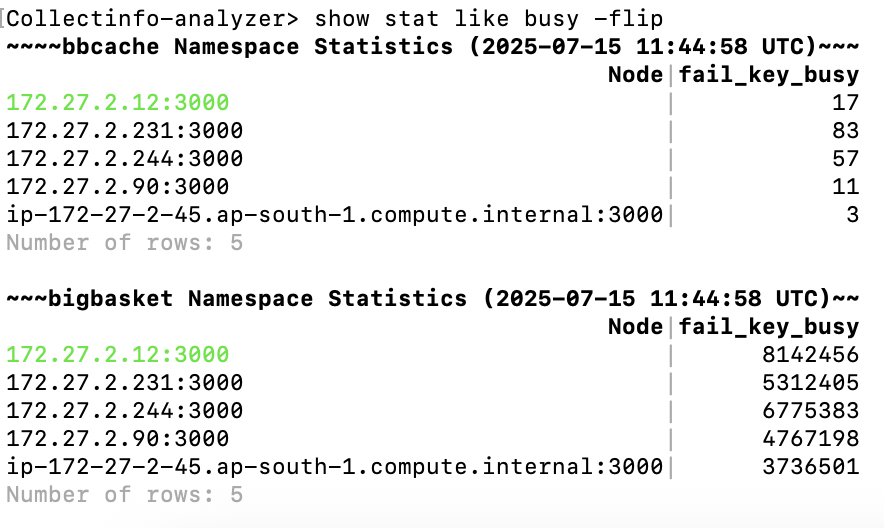
## Observations from Health Check

**Hotkeys**

High numbers of hot keys are reported on the bigbasket namespace. Please check out below approaches to identify the hot keys and some approaches around mitigating them. This is especially true if these errors keep increasing over a period of time.

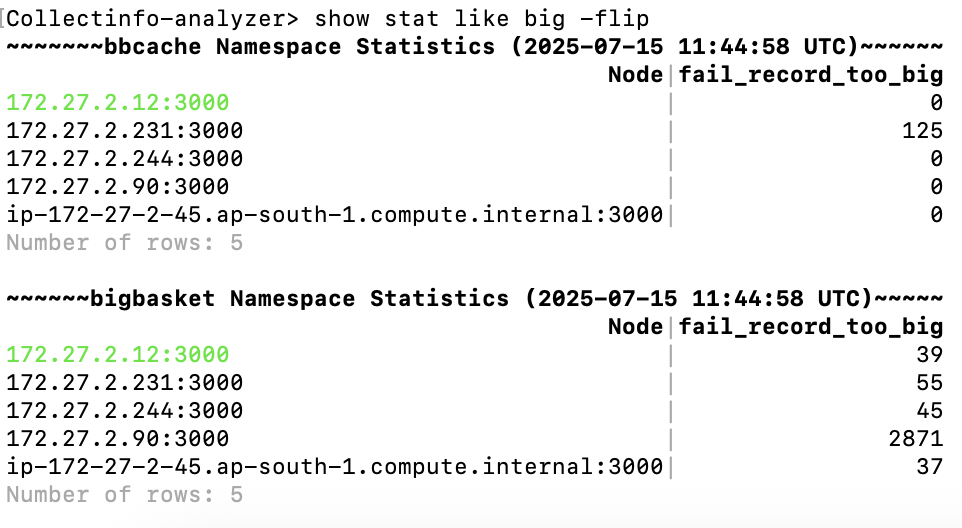
<https://discuss.aerospike.com/t/how-to-perform-hot-key-analysis/2091>

<https://support.aerospike.com/s/article/Why-does-my-client-return-Error-code-14-Hot-key>

****

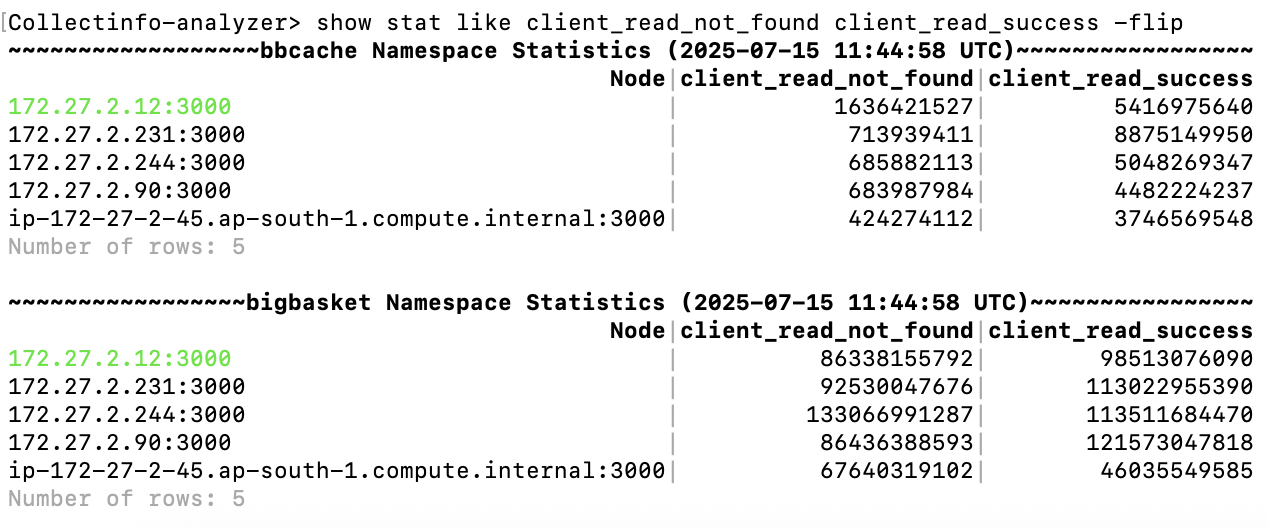
**Record Too Big Errors**

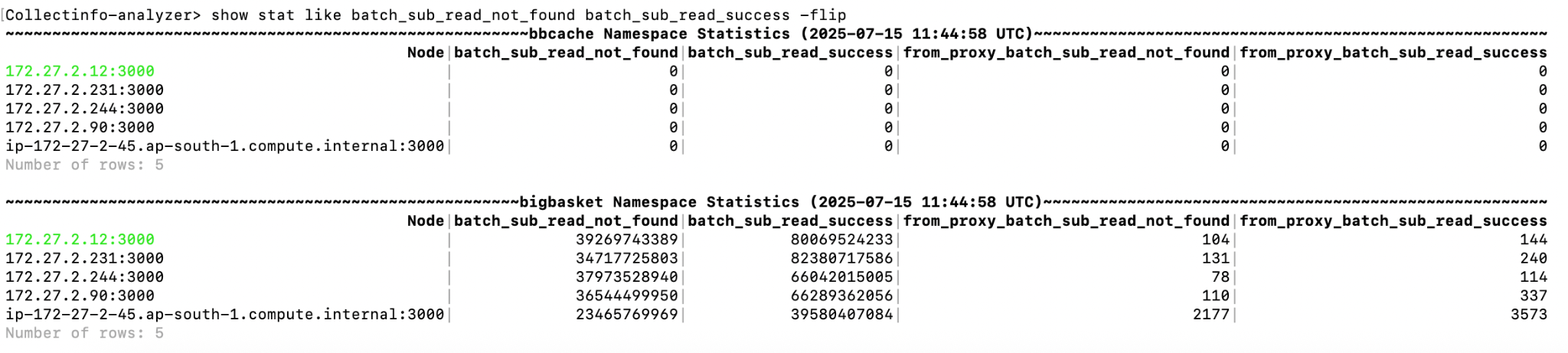
There are instances of both namespaces reporting record too big errors. This is not as high as the Himalaya cluster however it might be good to do a revisit of the application design to identify why these errors are cropping up if they keep coming up.



**High Client Reads and Batch Index Read Sub-Transactions Not Found**

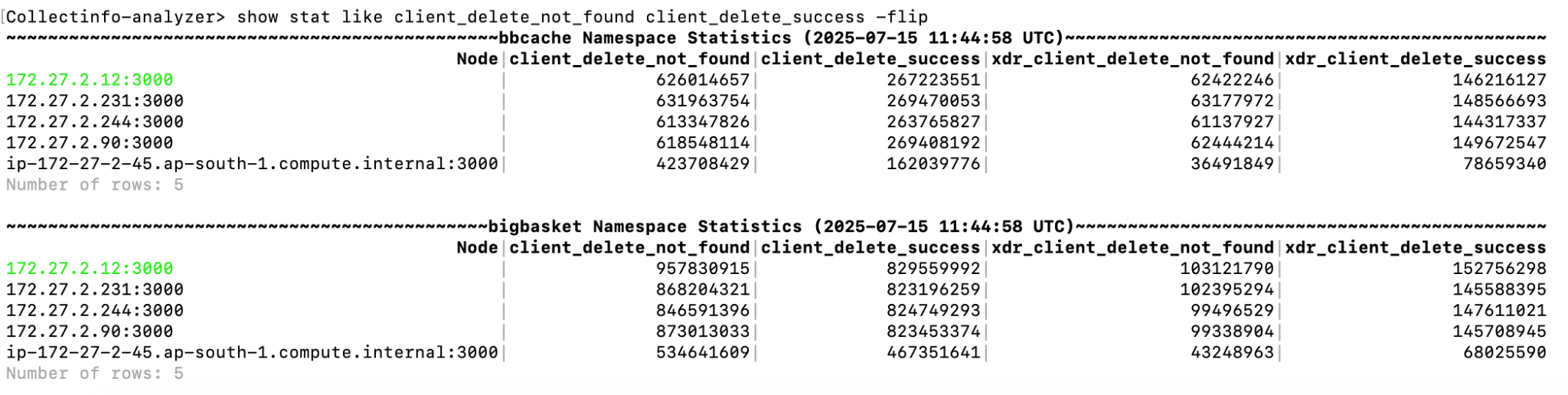
This is not necessarily an issue but might be good to review if we expect up to 50% of the reads in the *bigbasket* namespace to be on keys that do not exist. The same applies to batch read sub-transactions. Please check if an *exists* API is being used to check the existence of the key in memory and potentially reduce read load on the disk.

****

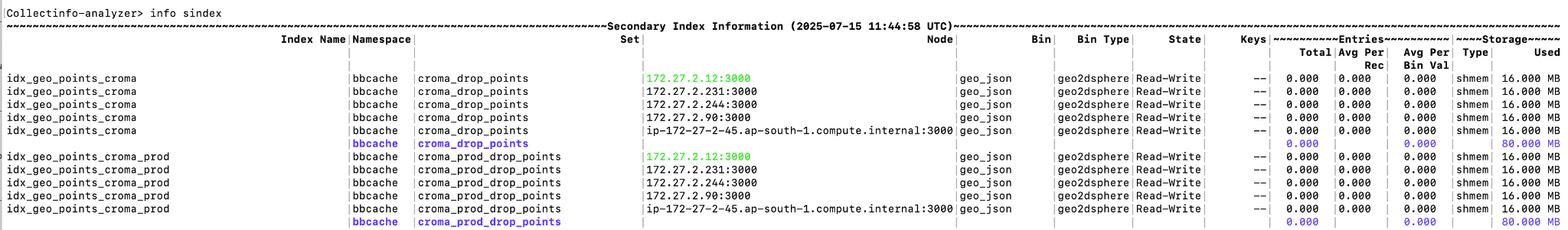
****

**High Client Delete Not Found**

Around 50% of the client deletes in the *bbcache* namespace are on keys that don't exist. This number goes up to 70% for the *bigbasket* namespace. Please review if this is expected application behaviour.

****

**Secondary Indexes with no entries**

****

There are multiple secondary indexes with no entries in them. This indicates that the secondary index is not being used. On a regular basis, it will be good to drop the secondary indexes not being used.

# Appendix A – Aerospike General Best Practices

The following are recommendations relating to best practices for Aerospike that are appropriate for this environment:

* Organize your data to match your data access patterns.
  + Identify data within records that are accessed more frequently than the rest of the data and separate this into a separate namespace.
  + Make sure that the TTL (Time To Live) associated with each record closely matches the desired (expected) lifespan of the data.
* Denormalize data whenever possible to leverage Aerospike’s phenomenal primary-key performance.
  + Create and use a compound key instead of using the equivalent of multiple joins (using relational database terminology).

Otherwise choose bins with higher cardinality / lower selectivity for secondary indexes.

* Make use of asmt, the Aerospike Shared-Memory Tool, to backup and restore the index in shared memory to speed up cold starts (<https://docs.aerospike.com/tools/asmt>).
* Leverage rack awareness to deploy a more resilient cluster. (<https://docs.aerospike.com/server/operations/configure/network/rack-aware>). Use PREFER\_RACK replica policy for lower latency reads if the risk of reading stale data is acceptable