Hazelcast Certification – Code Challenge

Benchmarking & Reports

# Introduction

This document contains the description of the benchmarking scenarios and its results.

# Benchmarking restrictions

Due to heavy restrictions in the usage of the AWS EC2 platform, the benchmarking was performed with the following configuration:

* **1 million** of credit cards
* 20 historical transactions per credit card
* 1 asynchronous backup
* 120 seconds of transaction generation process

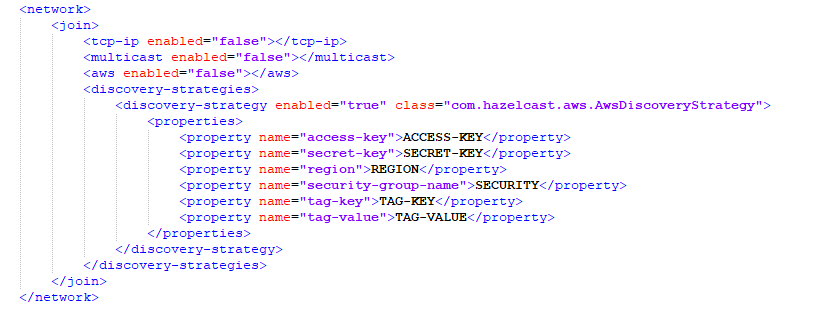
# Benchmarking setup

## Amazon EC2

For the benchmarking of the solution the Amazon Elastic Compute Cloud (Amazon EC2) environment have been used.

## Discovery configuration

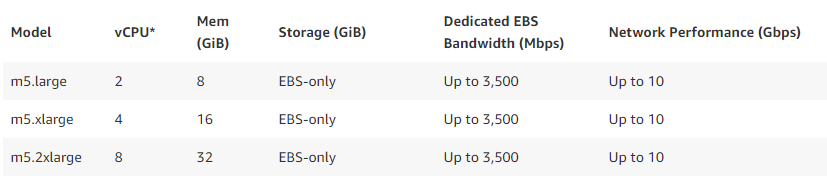
Members of the cluster and clients uses Discovery SPI as a discovery method.



1. Discovery configuration in ‘hazelcast.xml’ configuration file

## Instance types

The following instance types were used for the benchmarking of the system:



1. Technical specifications of the AWS EC2 Instance Types

# Benchmarking report

## Test 1 - m5.xlarge

Instance Type: m5.xlarge

Historical transaction loader: 1 client

Transaction generator & fraud detection server: 1 client

|  |  |  |
| --- | --- | --- |
| **INSTANCE TYPE** | **CLUSTER MEMBERS** | **TPS** |
| m5.xlarge | 6 | 4.390 |
| m5.xlarge | 4 | 3.029 |
| m5.xlarge | 2 | 73 |

1. Benchmarking results

With only two 2 nodes the system struggles and is resources available are very limited, so the TPS is very low and we got a poor performance

As we add more members to the cluster (which means more memory to store data and greater computing power) we enhance the performance of the cluster.

## Test 2 - m5.2xlarge

Instance Type: m5.2xlarge

Historical transaction loader: 1 client

Transaction generator & fraud detection server: 1 client

|  |  |  |
| --- | --- | --- |
| **INSTANCE TYPE** | **CLUSTER MEMBERS** | **TPS** |
| m5.2xlarge | 6 | 4.642 |
| m5.2xlarge | 4 | 4.750 |
| m5.2xlarge | 2 | 4.435 |

1. Benchmarking results

In this scenario, 2 members of the cluster are enough to achieve the optimal performance of the system. Adding more nodes doesn't improve the TPS.

## Test 3 - m5.large

Instance Type: m5.large

Historical transaction loader: 1 client

Transaction generator & fraud detection server: 1 client

|  |  |  |
| --- | --- | --- |
| **INSTANCE TYPE** | **CLUSTER MEMBERS** | **TPS** |
| m5.large | 6 | 4.378 |
| m5.large | 4 | OOM |
| m5.large | 2 | OOM |

1. Benchmarking results

With 6 members in the cluster, the system runs smoothly with a similar performance than previous tests, but when the cluster has 4 or less members, all members begins to fail until all nodes goes down due a lack of resources.