Sri Thripura
A Software Diagnostics Company

Sri Thripura @ IOT research labs Pvt Ltd

Problem Statement

The vehicle tracking system is flag ship product promised to track the vehicle near real time and the maximum delayed time is 30 seconds. This is delivered on SaaS model and users can monitor their fleet by integrating the API to their own interface. The application is hosted by one of private cloud dedicated server with configuration of Intel Xeon Server with 64 GB RAM and 2 TB disk space.
The problem with the application is that it stops responding causing customers lose tracking of their vehicles and the vendor not keeping the promise. The impact is very high as the issue causing loss of business and reputation.

Operational Constraints

a. Acquired the product, development team is gone
b. No Documentation of the product, installation or code available
c. Server Deployed in one of the private cloud providers in the USA
d. Developed in relatively older technologies like PHP and MariaDB

Sri Thripura

Diagnosis

Our Approach

1. Reverse Engineering of the Setup
2. Implanted Server Load monitors and Parameter Recorders
3. Reviewed the Architecture in milieu of latest technology trends
4. DB Schema and Query Review
5. Tech Stack Review

Findings

1. Server is developed in monolithic architecture
2. Single PHP script is responsible for tracking device locations, being invoked by all respective devices
3. Device location update is an aggressive and unforgiving algorithm
4. Live Tracking of vehicles through Mobile Application / API causing more burden on DB
5. OS is never configured to be application server
6. DB configuration is very rudimentary for the loads observed.

Our Plan Of Action

First 3 months

Deployment

1. HA configuration with load balancer
2. Database Parameter Optimization
3. OS Parameter Optimization

Technical Stack

1. PHP Version upgrade from 5.4 to 7.1
2. Angular JS with Web Sockets for Mobile Application and API Clients

Data Reorganization

1. Separate Configuration Data from Vehicle Data
2. Segregate data over time line

From 3 months – 6 months

Deployment

1. Enable Bin Logging
2. Configure Active - Active HA configuration

Devices

1. Combining multiple packets in case of timeouts
2. Separating recording positions from reporting
3. Time interval change: between two successful attempts.
4. Random amount of wait period if timeout occurs, increasing wait periods based on no.of timeouts

Server

1. Cached data Access Layer

Future

1. Adopt No - SQL data model for vehicle data segregated over time line
2. Relational Database
   a. Recent position tracking
   b. Configuration Data
   c. Reports and Dashboards

Envisaged Architecture