Name of the Customer: **The pioneer of electric vehicle technology in India**

About the Customer

The customer is the pioneer of electric vehicle technology in the country. Their mission is to bring tomorrow's movement, today & have a wide variety of electric vehicles and will be increasing this range even further. The products span personal and commercial segments and are designed to support the new paradigm of shared, electric and connected mobility.

Customer Challenges

- Containerization of 24 Microservices and deploy to ECS Fargate with API Gateways and WAF.
- Data movement from EBS to Cassandra; Cassandra to S3
- Aggregation of daily level Cassandra data as well as merging multiple data files into a single data file
- MQTT bridge between Azure and AWS NLB with EC2 machine

Why AWS

Amazon Web Services provides on-demand cloud computing platforms and APIs which are reliable, scalable on a metered pay-as-you-go basis. It provides the necessary features which exactly fits the customer's requirement to run their business.

The customer has applications running in their on-premise Data Centres and they decided to go with AWS since it provides low-cost migration services of the existing infrastructure to the AWS Cloud.

Along with that, AWS provides resources to aid in expansion and utilize more of their services for a long term basis with an appropriate Savings plan or reserving the capacity.

AWS is much more secure than a company hosting its website or storage. It currently has dozens of data centres across the globe which are continuously monitored and strictly maintained. The diversification of the data centres ensures that a disaster striking one region doesn't cause a permanent data loss worldwide.

Why the Customer Chose the Partner

Powerup is a cloud technology company with product & services in cloud migrations, managed cloud and AI solutions. A Premier Consulting Partner of AWS, Powerup provides product-led consulting solutions to enterprise customers on the public cloud platforms.

Powerup has gained immense popularity over the last few years and successfully executed over 150 projects in cloud transformation. The customer was convinced that Powerup would be the best option who could deliver the services and fulfil their business requirement. It stood as a market leader in the Cloud Migration sector and managed to setup a strong customer base through its successfully executed projects. This encouraged the Customer to move ahead with Powerup for assisting in the Cloud Migration Program and managing its services both in Cloud and on-premise.

Problem Statement

The customer wanted a solution which could deliver:

- Near real-time notification.
- Environment scalability for future loads.
- The availability of data for the analytics team.

Maintaining the reliability of data in Postgres and Cassandra and it's back up server.

What we proposed:

- A fully managed and scalable Infrastructure set up and configuration on AWS
- Application and services migration from Azure to AWS
- The setting of an ETL pipeline for analytics
- Setting managed data lake
  * Availability and structure of Historical data similar to Live data for analytics
- Framework for near real-time notification

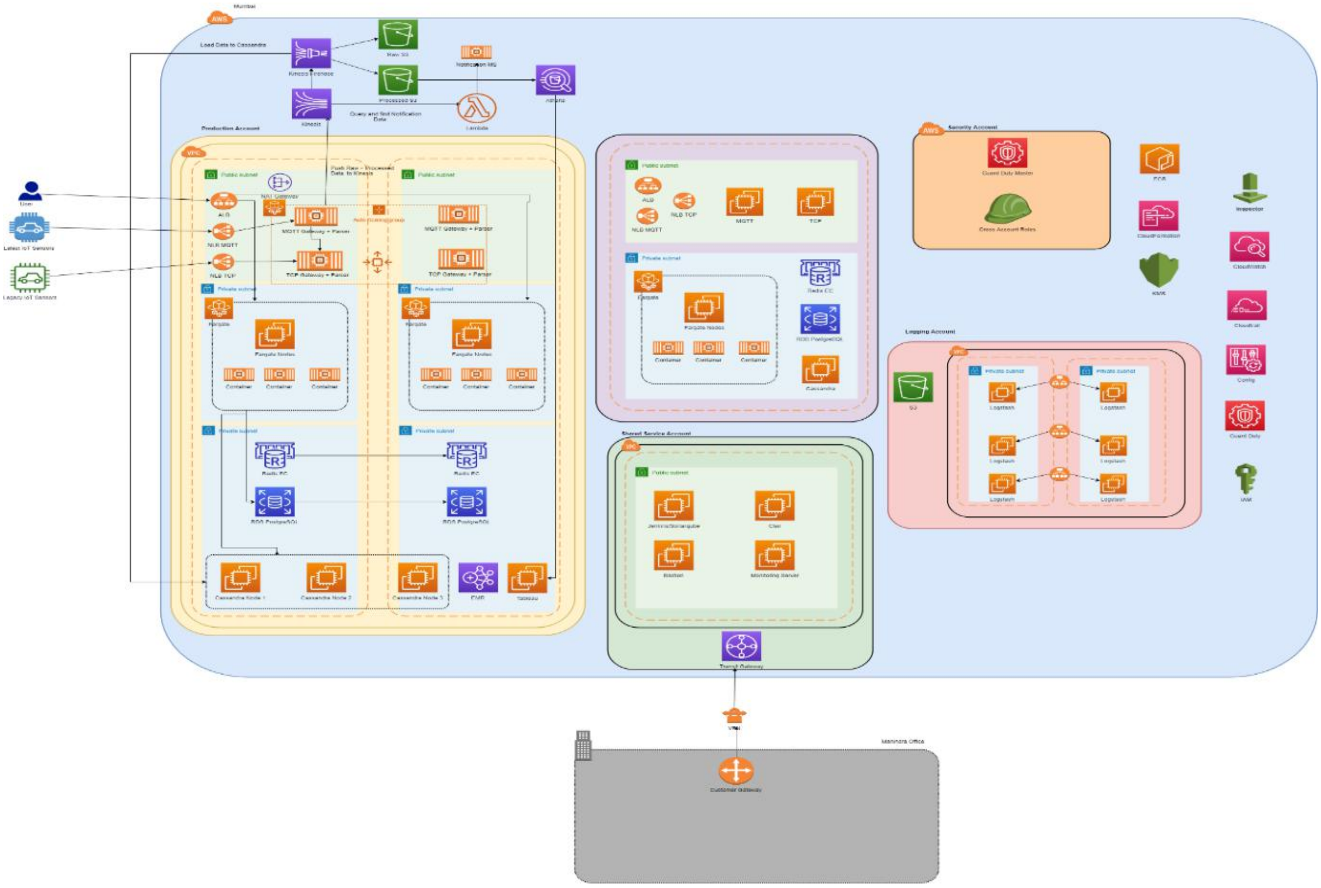
Partner Solution

- TCP & MQTT gateways running on EC2 machines and Parser application on a different EC2 instance.
- AWS Public IP addresses whitelisted on the IoT Sensor during manufacturing for the device to securely connect to AWS.
- The Gateway Server will push the raw data coming from the sensors to a Kinesis Stream.
- The Parser server will push the converted/processed data to same/another Kinesis stream.
- Lambda function will query the data in the Kinesis stream to find the fault/notification type data and will invoke the notification Microservice/ SNS to notify the customer. This reduces the current notification time from 6-8 minutes to almost near real-time.
- We will have Kinesis Firehose as a consumer reading from the Kinesis streams to push processed data to a different S3 bucket.
- Another Firehose will push the processed data to Cassandra Database and a different S3 bucket.
- AWS Glue will be used for data aggregation previously done using Spark jobs and push the data to a separate S3 bucket.
- Athena will be used to query on the S3 buckets. Standard SQL queries work with Athena and we will use Tableau to create the Dashboards.

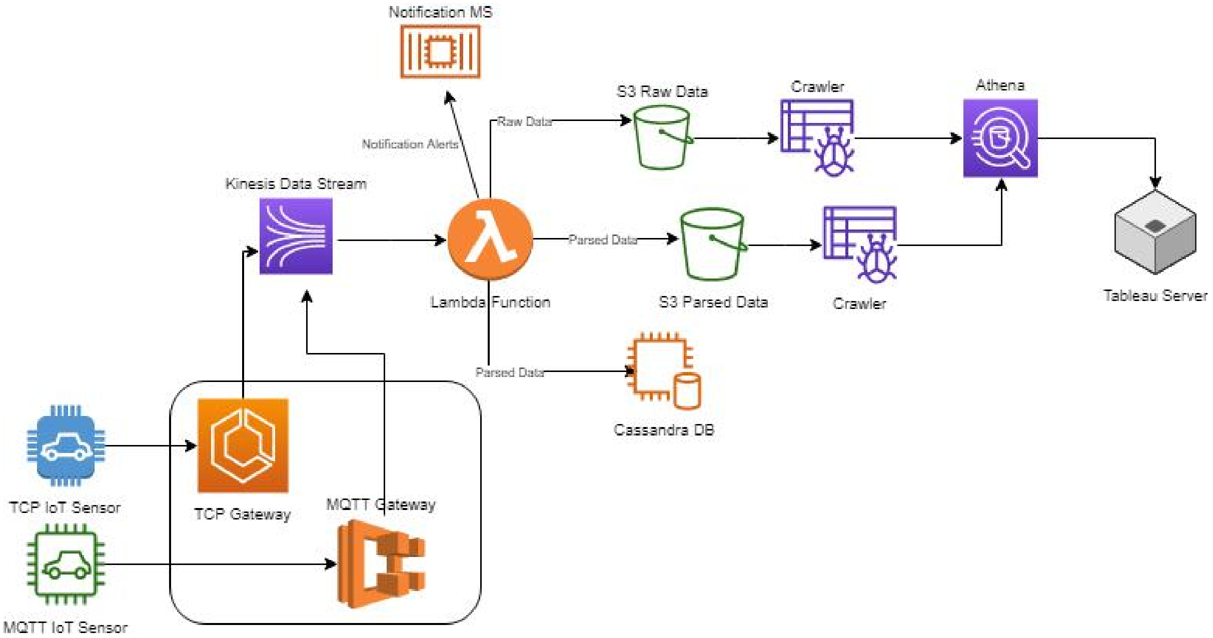
- NAT Gateway will be setup to enable internet access for servers in the private subnet.
- Application Load Balancer and WAF will be used as the ingress controller for Fargate.
- There will be 2 Network Load Balancer with a static IP which forwards the requests from IoT sensors to the API Gateways and then Parser server.
- All Docker images will be stored in Elastic Container Registry (ECR).
- AWS ECS Fargate will be used to run the Docker Containers. ECS Task Definitions will be configured for each container to be run.
- AWS ECS - Fargate will be used to deploy all the container images on the Worker Nodes. In Fargate the Control Plain & Worker Nodes are managed by AWS.
- The Scaling, HA & patching is handled by AWS. Application load balancer with WAF will be deployed as the front end to all the application Microservices.
- ALB will forward the request to the Kong API Gateway which in turn will route the requests to the microservices.
- Service level scaling will be configured in Fargate for more containers to spin up based on load.

Architecture Diagram

System Architecture



Data Ingestion Architecture



Lessons Learned

Parallel and collaborative effort from the customer and Powerup Team on containerization of the microservices.

Data-driven business decisions made by the customer team helped the movement of data easier and repetitive process.