



International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Build A Serverless Real Time Data Processing Application on AWS

Dr. Masrath Begum¹, Pratiksha U², Sushmita B³, Varshita V⁴, Vinaykumar J⁵

Assistant Professor, GNDEC Bidar, Karnataka Student, GNDEC Bidar, Karnataka
Associate Professor, VTU CPGS, Karnataka, India. masrath456@gmail.com.

ABSTRACT:

In research work uses serverless app to process real-time data streams. It builds infrastructure for a fictional ride-sharing company. In this case, This work enable operations personnel at a fictional Wild Rydes headquarters to monitor the health and status of their unicorn fleet. Each unicorn is equipped with a sensor that reports its location and vital signs. This work uses AWS to build applications to process and visualize this data in real-time. In this paper AWS Lambda is used to process real-time streams, Amazon DynamoDB to persist records in a NoSQL database, Amazon Kinesis Data Analytics to aggregate data, Amazon Kinesis Data Firehose to archive the raw data to Amazon S3, and Amazon Athena to run ad-hoc queries against the raw data.

[Serverless computing](#) allows you to build and run applications and services without thinking about servers. Serverless applications don't require you to provision, scale, and manage any servers. You can build them for nearly any type of application or backend service, and everything required to run and scale your application with high availability is handled for you. Building serverless applications means that you can focus on your core product instead of worrying about managing and operating servers or runtimes, either in the cloud or on-premises. This reduced overhead lets you reclaim time and energy that you can spent on developing great products which scale and that are reliable. This method considered a "server-less" platform / "Server-less Computing Execution Model" to build the real-time data-processing app. Architecture is based on managed services provided by AWS.

Keywords: AWS, Serverless, Cloud Computing

I. INTRODUCTION

Cloud Computing has become very popular due to the multiple benefits it provides and is being adopted by businesses worldwide. Flexibility to scale up or down as per the business needs, faster and efficient disaster recovery, subscription-based models which reduce the high cost of hardware, and flexible working for employees are some of the benefits of cloud that attracts businesses. Similar to cloud, Data Analytics is another crucial area which businesses are exploring for their growth. With the exponential rise in the amount of data available on the internet is a result of the boom in the usage of social media, mobile apps, IoT devices, sensors and so on. It has become imperative for the organisations to analyse this data to get insights into their businesses and take appropriate action

AWS provides a reliable platform for solving complex problems where cost-effective infrastructure can be built with great ease at low cost. AWS provides a wide range of managed services, including computing, storage, networking, database, analytics, application services and many more.

II. BACKGROUND STUDY

This work analysed multiple software solutions which perform analysis on data collected from the market and provide information as well as suggestions and provide better customer experience. This includes trade application providing stock price, taxi companies providing locations of nearby taxis, journey plan applications providing live updates on the different transport media and many more.

A cloud-based execution model in which the cloud provider dynamically allocates and runs the server. This is a consumption-based model where pricing is directly proportional to consumer use. AWS takes complete ownership of operational responsibilities eliminating infrastructure management and availability with higher uptime.

III. RELATED WORK

I. Mario Villamizar, Oscar Garces, "Infrastructure cost comparison of running web applications in the cloud using AWS lambda and monolithic and microservice architectures", 2016 16th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing, CC Grid 2016.

Large Internet companies like Amazon, Netflix, and LinkedIn are using the microservice architecture pattern to deploy large applications in the cloud as a set of small services that can be developed, tested, deployed, scaled, operated and upgraded independently. However, aside from gaining agility,

independent development, and scalability, infrastructure costs are a major concern for companies adopting this pattern. This paper presents a cost comparison of a web application developed and deployed using the same scalable scenarios with three different approaches: 1) a monolithic architecture, 2) a microservice architecture operated by the cloud customer, and 3) a microservice architecture operated by the cloud provider. Test results show that microservices can help reduce infrastructure costs in comparison to standard monolithic architectures. Moreover, the use of services specifically designed to deploy and scale microservices reduces infrastructure costs by 70% or more. Lastly, we also describe the challenges we faced while implementing and deploying microservice applications.

II. Hassan B. Hassan, Saman A. Barakat, Qusay I. Sarhan

" Survey on serverless computing", [Journal of Cloud Computing: Advances, Systems and Applications Volume 10 Issue 112 July 2021](https://doi.org/10.1186/s13677-021-00253-7) <https://doi.org/10.1186/s13677-021-00253-7>

Serverless computing has gained importance over the last decade as an exciting new field, owing to its large influence in reducing costs, decreasing latency, improving scalability, and eliminating server-side management, to name a few. However, to date there is a lack of in-depth survey that would help developers and researchers better understand the significance of serverless computing in different contexts. Thus, it is essential to present research evidence that has been published in this area. In this systematic survey, 275 research papers that examined serverless computing from well-known literature databases were extensively reviewed to extract useful data. Then, the obtained data were analyzed to answer several research questions regarding state-of-the-art contributions of serverless computing, its concepts, its platforms, its usage, etc. We moreover discuss the challenges that serverless computing faces nowadays and how future research could enable its implementation and usage.

III. Gimenez-Alventosa, Germán Molto, Miguel Caballer "A framework and a performance assessment for serverless MapReduce on AWS Lambda" Instituto de Instrumentación para Imagen Molecular (I3M) Centro mixto CSIC - Universitat Politècnica de Valencia, Camino de Vera s/n, 46022, Valencia, Spain

MapReduce is one of the most widely used programming models for analysing large-scale datasets, i.e. Big Data. In recent years, serverless computing and, in particular, Functions as a Service (FaaS) has surged as an execution model in which no explicit management of servers (e.g. virtual machines) is performed by the user. Instead, the Cloud provider dynamically allocates resources to the function invocations and fine-grained billing is introduced depending on the execution time and allocated memory, as exemplified by AWS Lambda. In this article, a high-performant serverless architecture has been created to execute MapReduce jobs on AWS Lambda using Amazon S3 as the storage backend. In addition, a thorough assessment has been carried out to study the suitability of AWS Lambda as a platform for the execution of High Throughput Computing jobs. The results indicate that AWS Lambda provides a convenient computing platform for general-purpose applications that fit within the constraints of the service (15 min of maximum execution time, 3008 MB of RAM and 512 MB of disk space) but it exhibits an inhomogeneous performance behaviour that may jeopardise adoption for tightly coupled computing jobs.

IV. METHODOLOGY

Serverless applications don't require you to provision, scale, and manage any servers. We can build them for nearly any type of application or backend service, and everything required to run and scale your application with high availability is handled for you. Serverless architectures can be used for many types of applications. For example, you can process transaction orders, analyze click streams, clean data, generate metrics, filter logs, analyze social media, or perform IoT device data telemetry and metering.

We will use AWS to build applications to process and visualize this data in real-time. We will use AWS Lambda to process real-time streams, Amazon DynamoDB to persist records in a NoSQL database, Amazon Kinesis Data Analytics to aggregate data, Amazon Kinesis Data Firehose to archive the raw data to Amazon S3, and Amazon Athena to run ad-hoc queries against the raw data.

Build a data stream: Create a stream in Kinesis and write to and read from the stream to track Wild Rydes unicorns on the live map. In this module you'll also create an Amazon Cognito identity pool to grant live map access to your stream. **Aggregate data:** Build a Kinesis Data Analytics application to read from the stream and aggregate metrics like unicorn health and distance traveled each minute. **Process streaming data:** Persist aggregate data from the application to a backend database stored in DynamoDB and run queries against those data.

Store & query data : Use Kinesis Data Firehose to flush the raw sensor data to an S3 bucket for archival purposes. Using Athena, you'll run SQL queries against the raw data for ad-hoc analyses

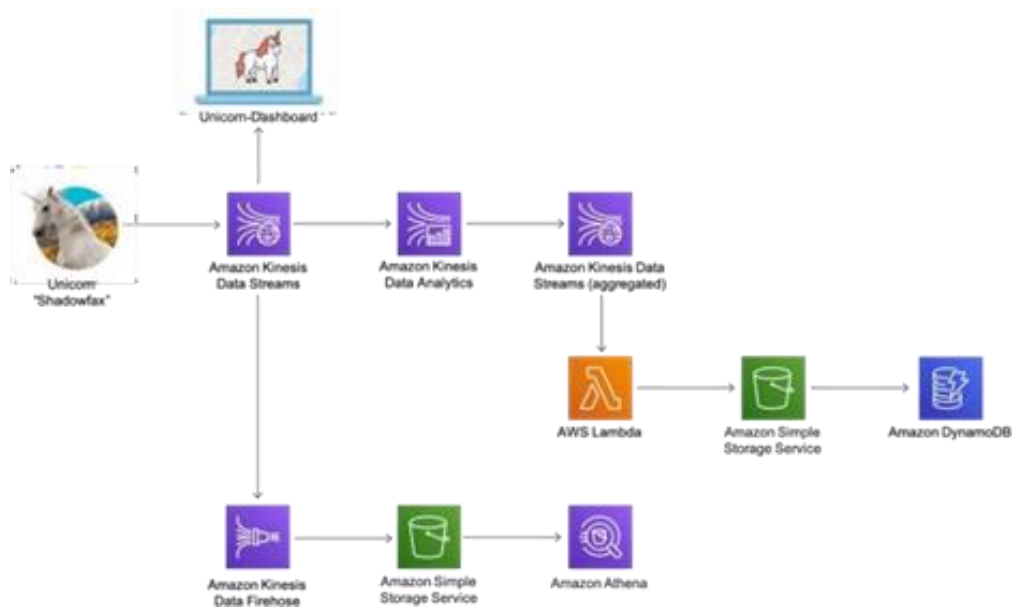


Fig 1: Architecture Diagram

IV. DESIGN

Real-time Streaming Data: Create an Amazon Kinesis stream, Produce messages into the stream, Read messages from the stream, Create an identity pool for the unicorn dashboard, Grant the unauthenticated role access to the stream, View unicorn status on the dashboard, Experiment with the producer.

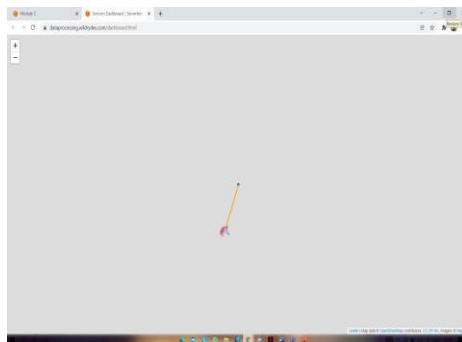


Fig 2: unicorn dashboard

Aggregate data: Create an Amazon Kinesis stream ,Create an Amazon Kinesis Data Analytics application.

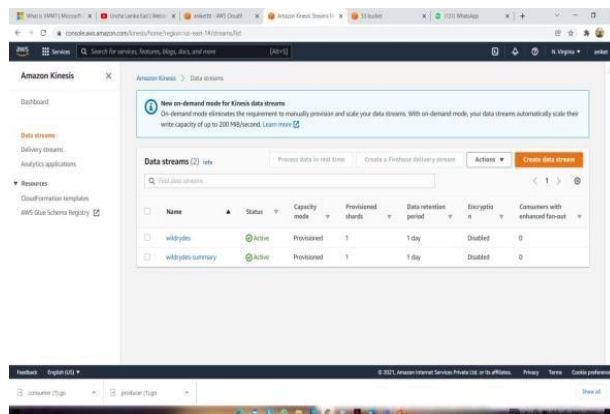


Fig 3: Amazon Kinesis stream

Process streaming data: Create an Amazon DynamoDB tables, Create an IAM role for your Lambda function ,Create a Lambda function to process the stream, Monitor the Lambda function, Query the DynamoDB table.

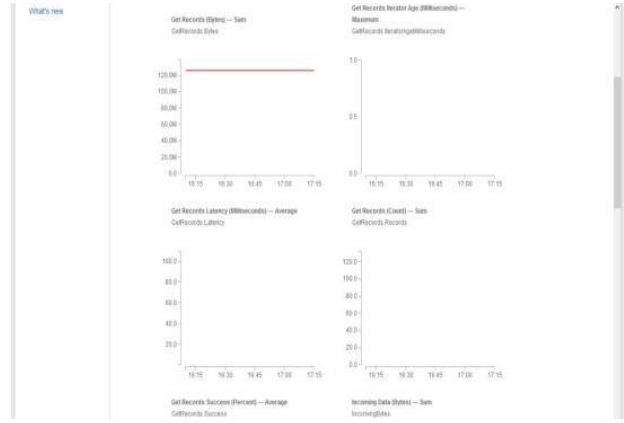


Fig 4: Monitor The Lambda Function

Store & query data: Create an Amazon S3 bucket, Create an Amazon Kinesis Data Firehose delivery stream, Create an Amazon Athena table, Explore the batched data files, Query the data files.

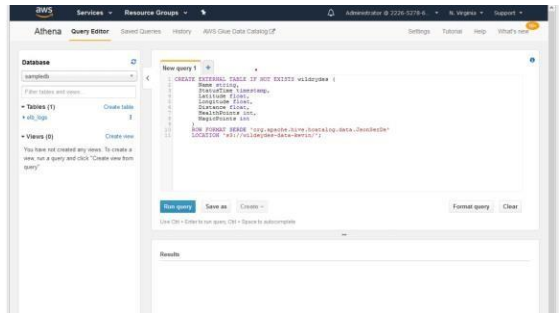


Fig 5: Create Athena Table

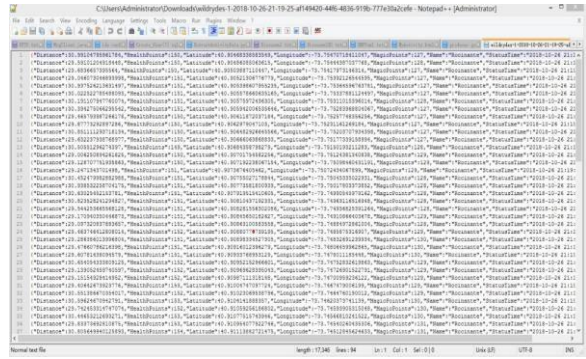


Fig 6: Explore the batched data files

Clean Up: Clean Amazon Athena, Clean Kinesis Data firehose, Clean S3, Clean Lambda, Clean DynamoDB, Clean IAM.

V. CONCLUSION

Using AWS services, we were able to create a real-time data processing application based on serverless architecture which is capable of accepting data through Kinesis data streams, processing through Kinesis Data Analytics, triggering Lambda Function and storing in DynamoDB.

The architecture can be reused for multiple data types from various data sources and formats with minor modifications. We have used all the managed services provided by AWS which led to zero infrastructure management efforts.

Capstone project has helped us in building practical expertise on AWS services like Kinesis, Lambda, Dynamo DB, Athena, S3, Identity and Access Management, Serverless Architecture and Managed Services. We have also learnt the programming language to build pseudo data producer programs. AWS CLI has helped us to connect on-premises infrastructure with cloud services.

VI. REFERENCES

Kotas, Charlotte W., Naughton III, Thomas J., and Imam, Neena. A comparison of Amazon Web Services and Microsoft Azure cloud platforms for high performance computing.

United States: N. p., 2018. Web. doi:10.1109/ICCE.2018.8326349.

K. Swedha and T. Dubey "Analysis of Web Authentication Methods Using Amazon Web Services" [2018 9th International Conference on Computing, Communication and Networking Technologies \(ICCCNT\)](#)

Giménez-Alventosa, V., Moltó, G., & Caballer, M. (2019). *A framework and a performance assessment for serverless MapReduce on AWS Lambda*. *Future Generation Computer Systems*, 97, 259–274.

G. McGrath and P. R. Brenner, "Serverless Computing: Design, Implementation, and Performance," 2017 IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW), Atlanta, GA, 2017, pp. 405-410.

H. Yoon, A. Gavrilovska, K. Schwan and J. Donahue, "Interactive Use of Cloud Services: Amazon SQS and S3," 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, Ottawa, ON, 2012, pp. 523-530.

Z. Al-Ali et al., "Making Serverless Computing More Serverless," 2018 IEEE 11th International Conference on Cloud Computing, San Francisco, CA, 2018, pp. 456-459