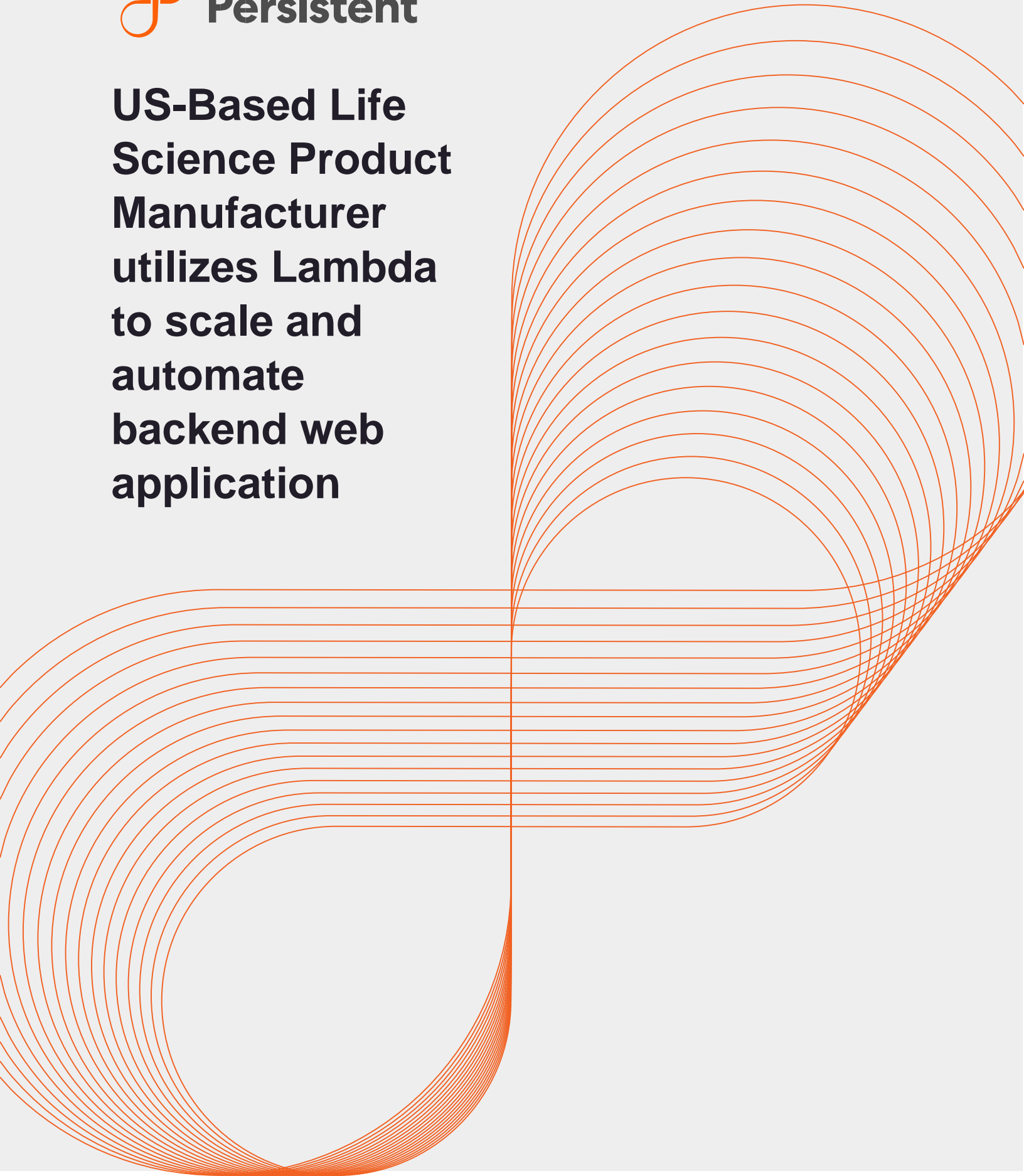




**US-Based Life  
Science Product  
Manufacturer  
utilizes Lambda  
to scale and  
automate  
backend web  
application**



## Contents

---

<b>1 About the Client .....</b>	<b>3</b>
<b>2 Problem Statement .....</b>	<b>3</b>
<b>3 What We Implemented .....</b>	<b>3</b>
<b>4 Outcomes and Benefits .....</b>	<b>5</b>

## 1 About the Client

---

The customer is a global leader in developing, manufacturing, and marketing a broad range of innovative products for the life science research and clinical diagnostic markets. With a focus on quality and customer service for over 65 years, their products advance the discovery process and improve healthcare. Their customers include university and research institutions, hospitals, public health and commercial laboratories, biotechnology, pharmaceutical, as well as applied laboratories that include food safety and environmental quality.

## 2 Problem Statement

---

The customer wanted to develop a backend web application using microservices based on AWS Lambda and API Gateway as a way to reduce operational complexity. The web application needed to move logs generated in AWS CloudWatch to Elastic Search to view on Kibana and automatically monitor Lambda functions, reporting metrics through CloudWatch. It needed to have the ability to send notification messages using AWS IoT Core to users based on their role defined in the application. Clients should also be able to send data by publishing a message on a topic and receive messages by subscribing to an AWS IoT topic. Message broker implementation is based on MQTT version 3.1 and needed Lambda to send email using SendGrid.

## 3 What We Implemented

---

### Service Overview

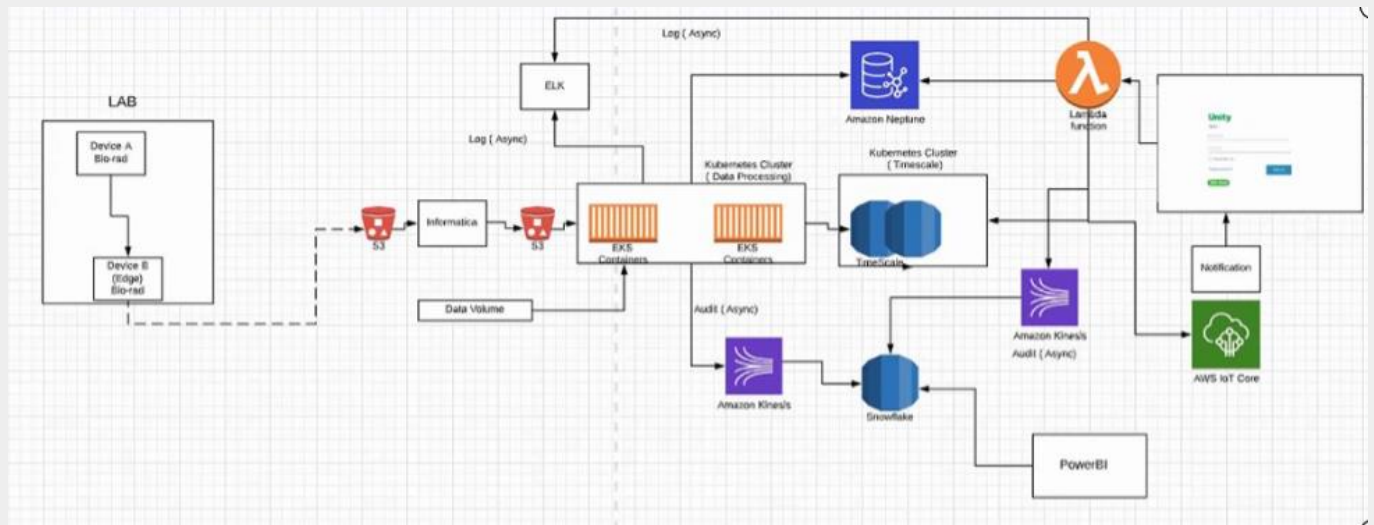
AWS Lambda was configured on CloudWatch events, which connected to the Elastic Search service, to write data to view on Kibana. AWS Elastic Search and Kibana were used to create a centralized logging system and Log Stash was used to move log files from EC2 to Elastic Search. Lambda had customized logic and was configured on an IoT Core rule event. This Lambda function connects to the AWS IoT Core service and passes messages only to authorized filtered users. The web application also utilized an AWS Lambda function to send an email using Send Grid.

AWS Lambda connects with S3 and file upload events are configured with SQS que. Golang service is configured on EC2 to poll the messages from SQS on any file uploaded to S3.

OKTA is used as the third-party user's identification for the API calls. Custom Lambda was created as API Gateway authorizer to validate an API user's token. AWS Lambda connects

with API Gateway manage multiple microservice functions. Lambda is tightly integrated with API Gateway which helps with the ability to make synchronous calls from API Gateway to Lambda and enables the creation of fully serverless applications. Terraform, Lambda, and CloudFormation are used for deployment scripts. Terraform is used for cloud agnostic deployment and AWS Lambda and CloudFormation are used for AWS only deployment.

Lambda connects with DynamoDB to manage a sequence generator microservice that is used to get an incremental account number during account creation within the customers' portal. This is automatically maintained across all configured regions of AWS using global tables. There are also Lambda functions that execute Gremlin queries against Neptune and other databases used by the web application.



## Lambda Details

22 Lambda functions were developed in total. Below are details on 5 Lambda functions used.

1. AWS Custom Authorizer for API Gateway - Lambda authorizer to validate the access tokens against OKTA
2. Creating a Rule with a Lambda Action - AWS IoT defines a rule that calls a Lambda function, passing in data from the MQTT message that triggered the rule. This allows the extract of data from the incoming message and then call another AWS or third-party service
3. Send an email using Send Grid
4. Generate a PDF in Lambda with NodeJS that creates a PDF using ghostscript
5. Microservice based on Lambda to move data generated in one Neptune database to snowflake based on Kinesis events. Power BI reporting was developed based on snowflake data.

Total estimated number of events per month expected for these Lambda functions:

Description	Values	Units
<b>Total concurrent Users</b>	500	Users
<b>Total Users</b>	50,000	Users
<b>Lambda Execution</b>	500	milli seconds
<b>Lambda Execution</b>	100	milli seconds

## Monitoring and Logging

AWS Lambda automatically monitors Lambda functions and reports metrics through CloudWatch. All errors are logged in CloudWatch and ELK and are used as the basis of log debugging and optimizing services. To help developers troubleshoot failures in a function, Lambda logs all function requests and automatically stores logs generated by code through CloudWatch Log.

## Lessons Learned

- AWS Lambda does not provide any visibility into the server infrastructure environment used to run the application code, while Amazon ECS actively exposes the servers used in the cluster as standard Amazon EC2 instances and allows/requires the user to size and scale their fleet themselves. Lambda functions must be written in one of supported the languages and are restricted in the type of actions they can perform. Amazon ECS, on the other hand, can run any container using any code that is capable of running in a container, which is almost any application that runs on a typical Linux operating system.
- Lambda using Golang as runtime proved to be better .Net core runtime Lambdas are more prone to cold start which causes performance issues.

## 4 Outcomes and Benefits

---

The web application was built on Persistent's five pillars of operational excellence; security, reliability, performance, efficiency, and cost optimization. Using Lambda allowed for a scalable web application platform for the customer. Lambda was a great approach for automating customized logic based on events of other AWS services With Lambda, you simply upload your code and let Lambda take care of everything required to run and scale the execution to meet your actual demand curve with high availability. This means, there is no administration of infrastructure needed. Lambda supports several programming languages and can be triggered from other AWS services or be called directly from any web or mobile application. One of the biggest advantages of Lambda is that you can move quickly and focus on business logic because security and scaling are managed by AWS.



## About Persistent Systems

Persistent Systems is a global company specializing in solutions, software products, and technology development services. Persistent Systems builds innovation that drives the business of our customer's enterprises with software at the core of their digital transformation. To learn more, visit <http://www.persistent.com>

### India

#### Persistent Systems Limited

Bhageerath, 402,  
Senapati Bapat Road, Pune 411 016  
Tel: +91 (20) 6703 0000  
Fax: +91 (20) 6703 009  
Email: [info@persistent.com](mailto:info@persistent.com)

### USA

#### Persistent Systems Inc

2055 Laurelwood Road, Suite 210,  
Santa Clara, CA 95054  
Tel: +1 (408) 216 7010  
Fax: +1 (408) 451 9177  
Email: [info@persistent.com](mailto:info@persistent.com)