CS3532 AWS Case Study Presentation

RV UNIVERSITY
Go, change the world
an initiative of RV EDUCATIONAL INSTITUTIONS

Deploying a Serverless Architecture for Real-Time S3 Event Notifications using AWS EventBridge, Lambda, and SNS

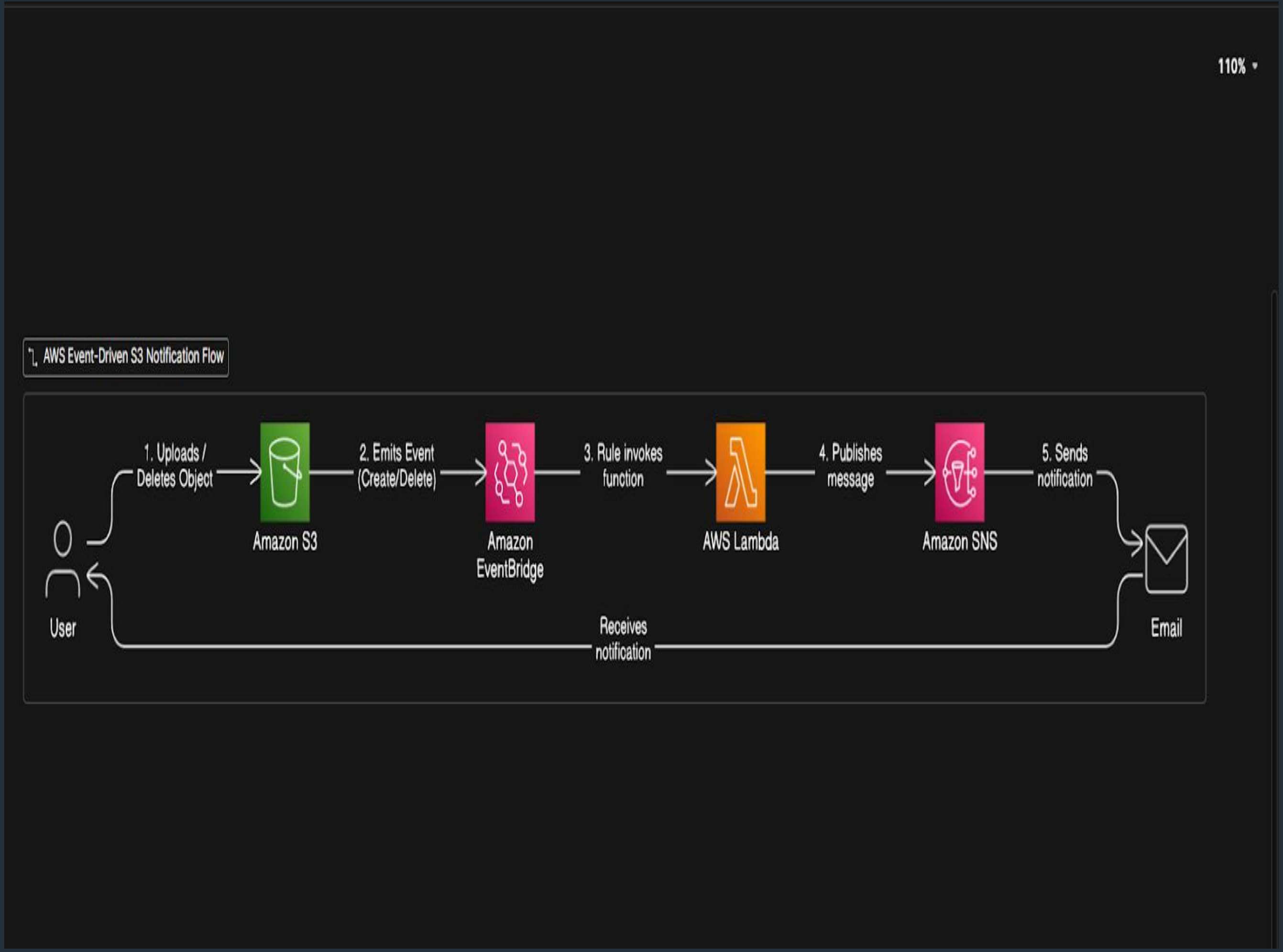
Shivanandreddy – 1RVU23CSE431
RISHEK J – 1RVU23CSE376
RISHIKESH A . R – 1RVU23CSE378

Real-Time Cloud Monitoring

- **The Problem:**
  How can we get instant, meaningful alerts about important activities in our cloud storage—like new file uploads or critical data deletions—without managing servers or constantly checking logs?
  - **The Solution: A Serverless Workflow**
    I built a fully automated system on AWS that listens for specific S3 events and sends formatted email notifications in real-time.
- **Why Serverless?**
  - **No Servers to Manage:** Zero administration overhead.
  - **Pay-per-Use:** Extremely cost-effective.
  - **Scalable:** Handles one event or a million events automatically

Architecture

1. **Key Components**
    a. Amazon S3 (Event Source)
    b. Amazon EventBridge (Event Router)
    c. AWS Lambda (Processing Logic)
    d. Amazon SNS (Notification Delivery)
    e. AWS IAM (Access Control)



Features

1. **Event-Driven Automation:**
   Automatically reacts to S3 events (upload/delete) without manual work.
2. **Real-Time Notifications:**
   Sends email alerts the moment a file is uploaded or deleted.
3. **Serverless Operation:**
   No need to manage servers — AWS manages all infrastructure.
4. **Scalable and Cost-Efficient:**
   Pay only for what you use; scales automatically.
5. **Secure Role-Based Access:**
   IAM ensures each service only has the permissions it needs.
6. **Modular Design:**
   Can easily add new features like SMS alerts, Slack integration, or database logging.

AWS Services Description

1. **Amazon S3 (Simple Storage Service):** Acts as the event source, generating notifications whenever a file is uploaded or deleted.
2. **Amazon EventBridge:** Serves as the **event bus** that receives S3 events, filters them, and routes only the required ones to AWS Lambda.
3. **AWS Lambda:** Functions as the **compute layer** that processes events, formats details, and sends messages to Amazon SNS using Python code.
4. **Amazon SNS (Simple Notification Service):** Works as the **notification system** that delivers real-time email alerts to the subscribed user.
5. **AWS IAM (Identity and Access Management)**
   Provides **secure access control**, allowing Lambda to publish notifications to SNS with least-privilege permissions.

Implementation Details

RV UNIVERSITY
Go, change the world
an initiative of RV EDUCATIONAL INSTITUTIONS

- **Amazon S3 – Event Source:**

  Created bucket **`aws-rishek-cp3-project`** in **ap-south-1 (Mumbai)** region.

  Configured **All Object Create** and **All Object Delete** event types.

  Set **Amazon EventBridge** as the event destination.

  Granted S3 permission to send events to EventBridge.

- **Amazon SNS – Notification Channel:**

  Created SNS topic **`s3ObjectCreatedTopic`** (Standard type).

  Added **Email subscription** for user notifications.

  Confirmed the subscription via email link (mandatory for message delivery).

- **AWS Lambda – Processing Logic**

  Created Lambda function **`format-s3-notification-email`** using **Python 3.9**.

  Attached IAM role with **AmazonSNSFullAccess** permission.

Implementation Details

- ❖ **Amazon EventBridge (Event Router):**

Created **rule:** NotifyOnS3ObjectCreated.

Defined **event pattern** to match only specific bucket events:"Object Created" or "Object Deleted" from aws-rishek-cp3-project.

Routes filtered events to the **Lambda function** as the target.

- ❖ **AWS IAM (Access Control):**

Ensures **secure communication** between services.

Lambda execution role attached with **AmazonSNSFullAccess**.

Enforces **least-privilege permissions** for all interactions.

Results and Conclusion

- The system successfully achieved **real-time email notifications** for every file **upload or deletion** in the Amazon S3 bucket, processed within **milliseconds** using **EventBridge, Lambda, and SNS**.
- **AWS Lambda** accurately parsed event details (bucket name, file name, timestamp, and file size for uploads) and **sent formatted alerts** through Amazon SNS to subscribed users.
- The **serverless, event-driven design** ensured **automatic scaling, zero manual effort, and pay-as-you-go cost efficiency**, with no servers to manage.
- The architecture proved to be **reliable, scalable, and easily extendable** for enterprise use cases like **audit tracking, security alerts, and monitoring workflows**.
- Overall, the project demonstrated the **power and practicality of serverless computing**, highlighting automation, simplicity, and high efficiency in modern cloud systems.

References

❖ **Amazon Web Services (2024)** – Official guides for **Amazon S3**, **EventBridge**, **Lambda**, **SNS**, and **IAM** were used to configure event notifications, routing, compute logic, messaging, and secure roles.
  🔗 https://docs.aws.amazon.com

❖ **Hendrickson et al. (2016)** – *Serverless Computation with AWS Lambda* – explained the working and efficiency of serverless execution models.

❖ **Villamizar et al. (2019)** – *Evaluation of Serverless Computing Architectures* – discussed performance and scalability in real-time cloud applications.

❖ **Gannon et al. (2017)** – *Cloud-Native Applications* – provided insights into designing modular and scalable systems on cloud platforms.

❖ **Hashizume et al. (2013)** – *Security Issues for Cloud Computing* – highlighted security best practices relevant to AWS-based workflows.

RV UNIVERSITY
Go, change the world
an initiative of RV EDUCATIONAL INSTITUTIONS

Thank you