Failover Testing & RPO/RTO Analysis Report

Project Overview

- **Project Title:** AWS Disaster Recovery Setup for a Multi-Tier Web App
- **Project Goal:** To design, build, and test an automated disaster recovery solution for a 3-tier web application, ensuring business continuity in the event of a regional failure.
- **Test Date:** September 14, 2025
- **Primary Region:** ap-south-1 (Mumbai)
- **Disaster Recovery (DR) Region:** us-east-1 (N. Virginia)

Infrastructure & Automation

- **Primary Environment (ap-south-1):** A 3-tier application was deployed, consisting of an Application Load Balancer (ALB), an Auto Scaling Group with EC2 instances, and an RDS database instance.
- **Disaster Recovery Environment (us-east-1):** The DR environment was deployed using a fully automated Infrastructure as Code approach. A CloudFormation (template.yaml) file was developed to reliably and repeatedly build the entire 3-tier stack, including all networking, security groups, compute, and database resources.

Test Scenario

A live failover test was conducted to validate the DR environment's readiness. The primary application was intentionally taken offline by deleting the listener on its Application Load Balancer, simulating a critical failure. Recovery was then validated by accessing the DR application's direct URL to confirm that the service remained operational.

Recovery Point Objective (RPO) Analysis

- **Definition:** The RPO represents the maximum acceptable amount of data loss, measured in time. It answers the question, "How much data can we afford to lose?"
- **Result:** This project's DR strategy uses an RDS snapshot-based approach. The process involves taking a snapshot of the primary database, copying it to the DR region, and restoring it. Therefore, the RPO is equivalent to the age of the most recent, successfully copied snapshot available in the us-east-1 region.

Recovery Time Objective (RTO) Analysis

- **Definition:** The RTO represents the maximum acceptable downtime for an application after a disaster. It answers the question, "How quickly can we be back online?"
- **Result:** The RTO for this manual failover test consisted of the time taken to simulate the failure and then successfully access the working DR application's URL. The end-to-end recovery of service was achieved in **under 5 minutes**, demonstrating a rapid manual recovery capability.

Challenges & Workarounds

- **Custom Domain DNS Resolution:**
  - **Challenge:** Initial testing of the custom domain (app.dr-project.com) failed with a DNS_PROBE_FINISHED_NXDOMAIN error.
  - **Root Cause Analysis:** The investigation revealed that while a Hosted Zone was correctly configured in AWS Route 53, the domain dr-project.com was not owned or registered with any domain registrar. As a result, the internet's public DNS system had no record of the domain and could not direct queries to AWS.
  - **Workaround:** To meet the project's core technical objectives, the custom domain was bypassed. All failover tests were successfully conducted using the direct DNS names of the Application Load Balancers. This approach proved the resilience of the underlying infrastructure and the success of the DR strategy.

Conclusion

The disaster recovery strategy and infrastructure were validated successfully. The project demonstrates a robust, automated method for deploying a DR environment via CloudFormation and a successful manual failover process. The challenges encountered with DNS provided a valuable learning opportunity in the prerequisites of domain management.