DNS Record Management with AWS Lambda

As part of my responsibilities during infrastructure enhancements—particularly related to Splunk component rehydration—I developed a DNS Handler Lambda function to dynamically manage DNS records in AWS Route 53. This automation ensured reliable service discovery and internal connectivity as EC2 instances were frequently rebuilt or replaced.

I implemented and integrated several key AWS components and modules:

- **Lambda Function** – Handled dynamic creation and modification of A and CNAME records in Route 53 using EC2 instance metadata.
- **Instance Tag Parsing Module** – Extracted meaningful DNS names from instance tags using standardized naming conventions.
- **EventBridge Rule** – Triggered the Lambda function in response to EC2 instance state changes (e.g., launch or terminate events).
- **IAM Role & Policies** – Created least-privilege IAM roles with policies to grant the Lambda function access to EC2, Route 53, and CloudWatch.
- **CloudWatch Logging** – Captured function execution logs for auditing, monitoring, and troubleshooting purposes.
- **Splunk Connectivity Module** – Ensured search heads, indexers, and forwarders remained connected across infrastructure refreshes.
- **Retry & Error Handling Module** – Implemented robust retry logic and error handling to minimize failure rates and ensure operational resilience.

This solution reduced manual DNS management, increased environment reliability, and ensured continuous connectivity across Splunk components, even during frequent infrastructure updates. It also deepened my hands-on experience with event-driven architectures and AWS IAM, logging, and monitoring tools.