



Infrastructure Overview

The architecture consists of several key components organized into layers, each crucial in ensuring the system's functionality and performance.

1. Client Layer

- **Client:** This is the entry point where users or applications interact with the system, sending requests and receiving responses.

2. Gateway and Load Balancing

- **Amazon API Gateway:** Acts as the front door for all client requests, routing them to the appropriate backend services. It provides security, throttling, and monitoring.
- **Classic Load Balancer:** Distributes incoming application traffic across multiple targets, such as EC2 instances, ensuring high availability and reliability.

3. Authentication and Access Control

- **AWS Identity and Access Management (IAM):** Manages access to AWS services and resources securely. It provides fine-grained access control across the system.
- **Amazon Cognito:** Handles user authentication, authorization, and user management. It authenticates users and grants access tokens for secure API interactions.

4. Business Logic Layer

- **ECS Cluster:** A highly scalable and high-performance container orchestration service. It runs microservices in containers, ensuring scalability and efficient resource utilization.

5. Microservices

- **Carrier Microservice:** Manages interactions and data integration with various insurance carriers. It routes carrier-related requests, queues carrier processing jobs, and publishes updates.
- **Policy Microservice:** Handles insurance policy management. It routes policy-related requests, queues policy processing jobs, and publishes updates.

6. Serverless Computing

- **AWS Lambda:** Executes code in response to triggers, such as HTTP requests via API Gateway, without provisioning or managing servers. It processes events and integrates with other AWS services.

7. Data Management and Processing

- **Amazon SQS (Simple Queue Service):** Manages message queues, ensuring reliable communication between decoupled microservices.
- **Amazon SNS (Simple Notification Service):** Publishes messages to subscribers, ensuring timely and reliable notifications.
- **Amazon RDS (Relational Database Service):** Manages relational databases, ensuring data integrity and supporting complex queries.
- **Amazon DynamoDB:** A NoSQL database service for storing and retrieving any amount of data, ensuring low latency and scalability.
- **Amazon Kinesis:** Streams and processes real-time data for analytics and machine learning.
- **Amazon Redshift:** A data warehousing service that allows for fast query performance on large datasets.
- **Amazon Elastic Cache:** Provides in-memory caching to improve data retrieval speeds and application performance.

8. Monitoring and Logging

- **Amazon CloudWatch:** Monitors and logs system metrics, providing insights into system performance and health.
- **Aggregator:** Aggregates responses from various microservices, processes them, and returns aggregated results to the client.

Scalability, Reliability, and Data Consistency

Scalability

- The use of **ECS Cluster** and **AWS Lambda** allows the system to scale automatically based on traffic and processing demands.
- **Amazon API Gateway** and **Classic Load Balancer** distribute incoming requests to ensure no single point of failure.
- **Amazon SQS** and **Amazon Kinesis** handle high-throughput data streams, ensuring the system can process large volumes of data in real time.

Reliability

- The architecture uses multiple Availability Zones (AZs) within AWS regions to ensure high availability and fault tolerance.
- **Amazon RDS** and **Amazon DynamoDB** provide multi-AZ deployments for database redundancy.
- **Amazon CloudWatch** continuously monitors the system, enabling quick detection and resolution of issues.

Data Consistency

- **Amazon RDS** ensures **ACID transactions** for relational data, maintaining strong consistency.
- **Amazon DynamoDB** supports eventual consistency for high scalability while offering options for strongly consistent reads.
- **Amazon Elastic Cache** provides caching for frequently accessed data, reducing latency and ensuring quick retrieval.

Cloud Services Utilization

The architecture uses various AWS cloud services to enhance scalability, reliability, and data consistency:

- Amazon API Gateway for request routing and security.
- AWS IAM and Amazon Cognito for authentication and access control.
- ECS Cluster and AWS Lambda for scalable computing.
- Amazon SQS, Amazon SNS, Amazon Kinesis for reliable messaging and data streaming.
- Amazon RDS, Amazon DynamoDB, and Amazon Redshift for robust data management.
- Amazon Elastic Cache for improved performance with in-memory caching.
- Amazon CloudWatch for comprehensive monitoring and logging.

This detailed backend architecture ensures the system is scalable, reliable, and consistent, capable of handling complex data processing and integration tasks for multiple insurance carriers.