Guidance for Game Analytics Pipeline on AWS

Architecture

This architecture diagram shows a modernized DataOps pipeline for centralized game analytics on AWS.

1. **Data Producers**
   - PC Client
   - Mobile Client
   - Game servers
   - SDK

2. **Amazon Cloud**
   - CI/CD Pipeline

3. **Data Producers**
   - Amazon Kinesis Data Streams
   - Amazon Managed Service for Apache Flink
   - Amazon API Gateway
   - Amazon DynamoDB

4. **Data Consumers**
   - Admins
   - LiveOps

5. **Data Producers**
   - Amazon CloudWatch
   - Amazon SNS

6. **Data Consumers**
   - Amazon S3
   - AWS Glue
   - Amazon Athena
   - Amazon QuickSight

7. **Data Producers**
   - Process batched telemetry data with Kinesis Data Firehose.
   - Store both raw and processed telemetry data in Amazon Simple Storage Service (Amazon S3).
   - Extract, transform, and load (ETL) stored telemetry data for analysis with AWS Glue.
   - Interactively query and analyze prepared data with Amazon Athena.
   - Visualize business intelligence (BI) data with Amazon QuickSight.
   - Deploy and operationalize the codified application using a continuous integration and continuous deployment (CI/CD) pipeline.

8. **Data Producers**
   - Provide LiveOps with BI, data visualizations, and machine learning (ML) capabilities from game telemetry data to generate key business insights.

9. **Data Consumers**
   - Send game telemetry events to the AWS Cloud from data producers.
   - Capture streaming data from the game with Amazon Kinesis Data Streams, and process data in near real-time with Amazon Kinesis Data Firehose and Amazon Managed Service for Apache Flink.
   - Provide REST API endpoints with Amazon API Gateway to register data producers. Store game configurations and API access keys in Amazon DynamoDB.
   - Capture streaming event data in Amazon Managed Service for Apache Flink. Publish custom metrics in Amazon CloudWatch.
   - Create operational dashboards and alarms from custom metrics in CloudWatch.
   - Deliver critical alarm notifications to data consumers with Amazon Simple Notification Service (Amazon SNS).
Guidance for Game Analytics Pipeline on AWS

DataOps CI/CD Pipeline

This architecture diagram shows the DataOps CI/CD pipeline for centralized game analytics on AWS.

1. Build and test the codified infrastructure using the AWS Cloud Development Kit (AWS CDK) to synthesize an AWS CloudFormation template.
2. Initiate the CI/CD pipeline when infrastructure code changes are committed to the AWS CodeCommit repository.
3. Store compiled infrastructure assets, such as a Docker container and CloudFormation templates, in Amazon Elastic Container Registry (Amazon ECR) and Amazon S3.
4. Deploy the infrastructure for integration and system testing into the quality assurance (QA) AWS account using the CloudFormation Stack.
5. Run automated testing scripts to verify that the deployed infrastructure is functional inside an AWS CodeBuild project.
6. Deploy the tested infrastructure into the Production (PROD) AWS account using the CloudFormation Stack.