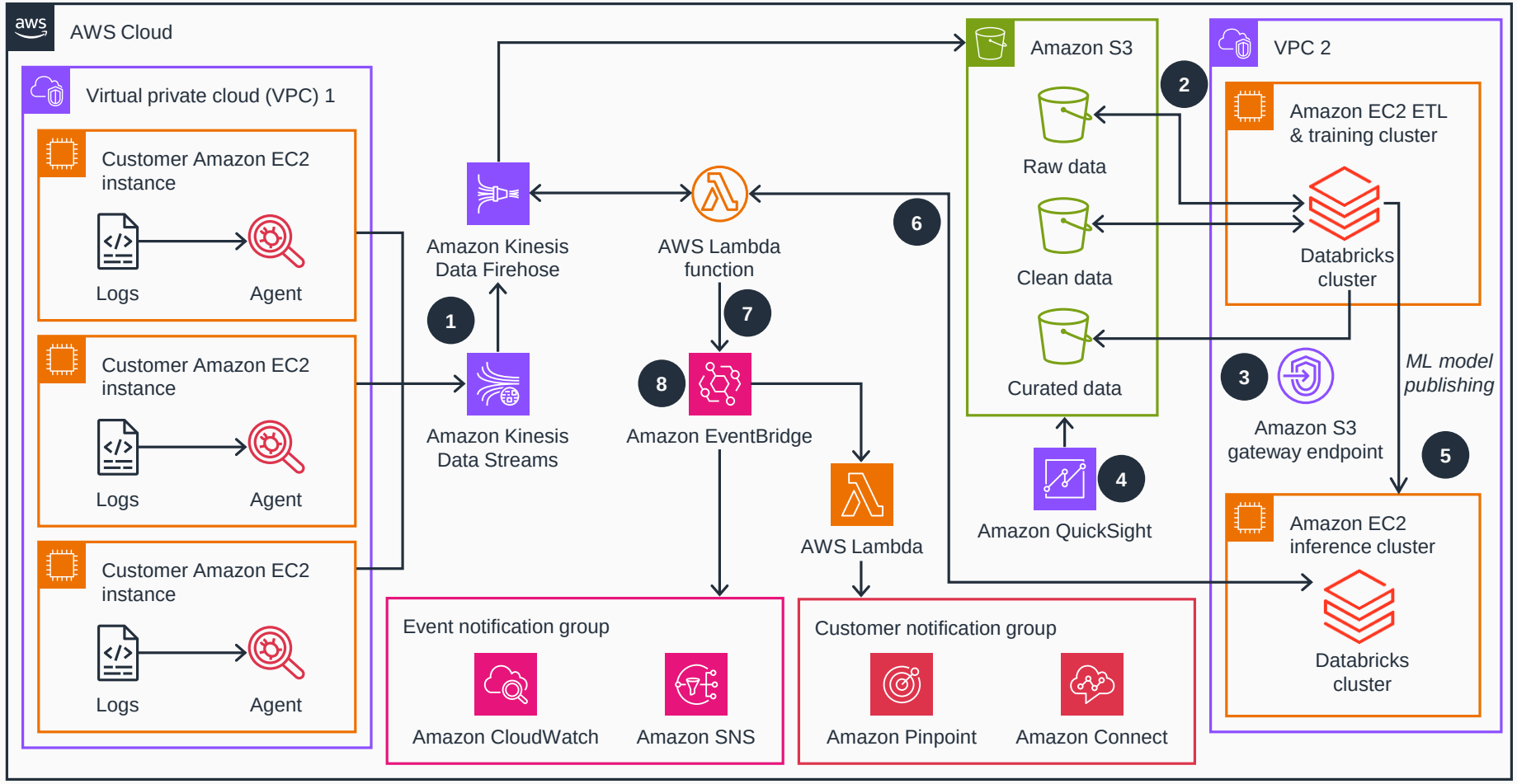
Guidance for Identification of Problematic Betting & Gaming on AWS

This architecture diagram shows an automated ML mechanism that helps protect players by predicting and notifying players and operators of problematic betting and gaming behavior in near real-time.



1. **Amazon Kinesis** agents encrypt and send data to **Amazon Kinesis Data Streams**, which forwards it to **Amazon Kinesis Data Firehose** for risk evaluation, formatting, and storage of raw data using **Amazon Simple Storage Service (Amazon S3).**

2. The Databricks Lakehouse Platform (Databricks) running on the **Amazon Elastic Compute Cloud (Amazon EC2)** extract, transform, load (ETL) and training cluster reads raw data from **Amazon S3** and transforms the data to clean data, then to curated data, writing it back to **Amazon S3** using the Delta Lake format.

3. Access to **Amazon S3** is brokered by an **Amazon S3** gateway endpoint, providing secure, reliable connectivity without requiring an internet gateway or network address translation device.

4. **Amazon QuickSight** provides dashboards that access curated data.

5. The ML model creates a risk score to predict problematic play and publishes that model to an **Amazon EC2** inference cluster.

6. **Kinesis Data Firehose** invokes an **AWS Lambda** function to send wagers through representational state transfer to the Databricks Lakehouse Platform on the **Amazon EC2** inference cluster (Databricks cluster) for risk evaluation, which returns a risk score for each player wager.

7. **Lambda** evaluates the risk score and forwards it to **Amazon EventBridge** if it exceeds a customer-configured threshold.

8. **EventBridge** sends notification events to **Amazon Simple Notification Service (Amazon SNS)** or to **Amazon Pinpoint** or **Amazon Connect** using **Lambda**. Monitoring and logging information is sent to **Amazon CloudWatch**.

Reviewed for technical accuracy October 30, 2023

AWS Reference Architecture