Monitoreo en tiempo real de transacciones con Kafka

Contexto:

Una empresa fintech requiere implementar un sistema de monitoreo en tiempo real para detectar patrones sospechosos en las transacciones de sus clientes. Actualmente, el procesamiento se realiza en modalidad batch al final del día, lo que genera demoras críticas en la detección de fraudes. Se propone el uso de Apache Kafka junto con un motor de procesamiento en streaming como Spark Structured Streaming, que permita capturar y procesar eventos a medida que ocurren.

Objetivo del flujo:

- Detectar en tiempo real patrones de fraude en transacciones financieras.

- Reducir el tiempo de detección de horas a segundos.

- Disparar alertas inmediatas a equipos de monitoreo y sistemas de bloqueo automático.

Diseño del flujo de ingesta en streaming:

1. Origen de datos:
  - API de pagos, sistemas bancarios, aplicaciones móviles y web.
  - Eventos de autenticación y cambios de perfil del cliente.

2. Kafka (plataforma de mensajería):
  - Tópicos principales: tx-auth (transacciones autorizadas) y tx-ctx (contexto: IP, dispositivo, país).
  - Particionado por accountId o cardId para mantener coherencia en ventanas.
  - Replicación (RF=3) y retención configurada para 24-72 horas.

3. Procesamiento en Spark Structured Streaming:
  - Lectura de datos desde Kafka.
  - Limpieza, normalización y enriquecimiento con listas negras de IP/BIN y límites por cliente.
  - Aplicación de reglas antifraude: transacciones de monto alto en horarios inusuales, países o comercios sospechosos, exceso de operaciones en una ventana corta de tiempo.
  - Uso de ventanas deslizantes de 5 minutos (slide cada 1 minuto) con watermarks para eventos tardíos.
  - Opcional: integración de un modelo de Machine Learning para score de riesgo.

4. Destino de resultados:
- Alertas enviadas a un tópico Kafka fraud-alerts.
- Visualización inmediata en dashboards (Grafana/Kibana).
- Almacenamiento histórico en Data Lake (S3/Delta/Iceberg).

5. Beneficios frente a procesamiento batch:
- Tiempo de reacción inmediato frente a fraudes.
- Escalabilidad gracias a Kafka y procesamiento distribuido.
- Mejor observabilidad mediante métricas y dashboards en tiempo real.
- Reducción de pérdidas económicas al bloquear fraudes casi en línea.