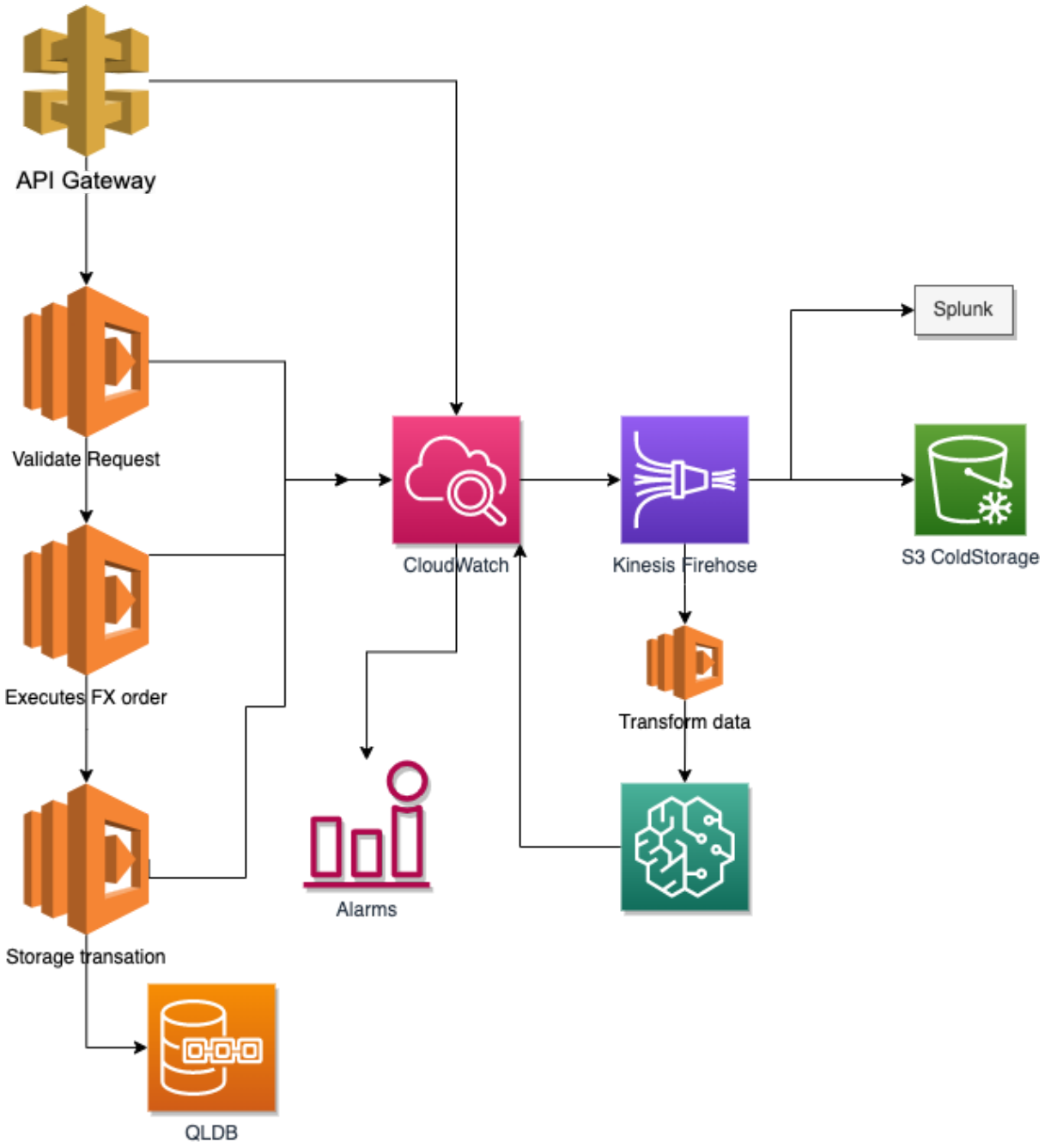
AWS Architecture diagram



Modules

- API management platforms (**API Gateway)**:
    - It receives requests from clients and routes them to the appropriate microservices. And streaming logs of all the API calls received.
- Microservices (**Lambdas**)
    - 1. Validates the request, 2. Executes the FX trade order, 3. Stores the trade data in a database.
- Store trade data (**AWS QLDB**)
    - Receives the orders and stores journal information in a verificable crypto block
- Streaming Log Collector (**Cloudwatch and Kinesis**)
    - Receives the streaming logs generated by the API Gateway and the microservices and forwards them to the log storage system and the Anomaly detection service.
- Log Storage System (**Splunk**)
    - Provides an interface to query and retrieve logs
- Anomaly Detection Service (**Sagemaker**)
    - Receives the streaming logs and runs the Machine Learning model, and generates alerts when an anomaly is detected.
- Alert Notification System: (**CloudWatch Alarms**)
    - Receives alerts generated by the Anomaly Detection Service and notifies via various channels, PagerDuty, chat, email, SMS.

Reasons

- API Gateway:
    - Easier to configure, integrate and cheaper than other alternatives as ApiGee. Evaluate other alternatives if switching cloud provider, developer testing, analytics, or monetization capabilities are required
- Lambdas:
    - Easy integration with API Gateway and CloudWatch. Easy deployment and management and automatic scaling. Evaluate other alternatives if limitation on execution time (15 min), memory or storage and if predictable cost is a requirement, migrate to container orchestration service as ECS or K8S if required more flexibility
- QLDB:
    - Easier integration with Lambdas, scalability build-in and due to the sensibility of the data the cryptographic verification of inmutable ledgers. If limitation of cost or publicity is required, conside to migrate to an open protocol ledger
- Cloudwatch as a monitoring and observability service:
    - Seamless integration with CloudWatch alarms and other AWS components as lambdas or API Gateway. If higher observability and traceability detailed is needed then consider to migrate to NewRelic or Datadog
- Kinesis:

    - Seamless integration with Clouchwatch, easier tools for visualization and transformation of log streams and cost efficiency than Spark which requires provisioned clusters. If more flexibility and control over the processing logic is required, consider to move to Apache Spark
- Splunk:
    - Rich environment for querying compared with other alternatives such as Kibana or Cloudwatch dashboards. If set up and cost are a requirement, consider to migrate to Cloudwatch
- Sagemaker:
    - I assumed as requirement that a trained model is needed in order to find anomalies, in the case of simpler algorithm are enough, then another alternatives incorporated in CloudWatch or Kinesis could be enough to reduce complexity
- S3 Glacier:
    - Splunk stores up to 30 days of data, if a longer period is required than store information in S3. S3 Glacier is optimized for infrequent access to data over cheaper cost per information saved