*Use Case 3: Central Logging in Multi-Account Environments:*

Overview

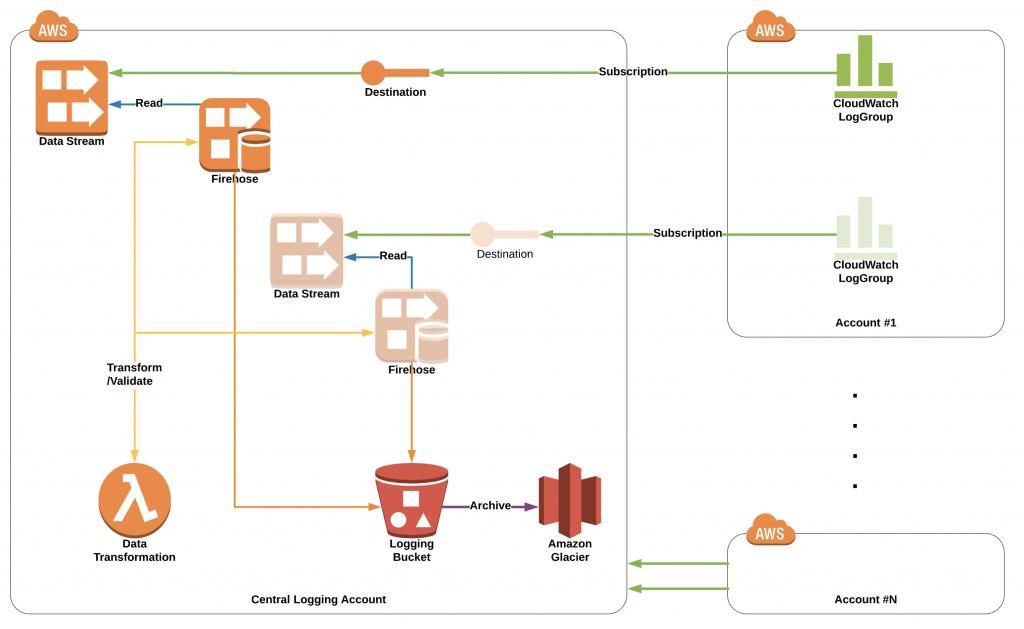
Centralized logging is often required in large enterprise environments for a number of reasons, ranging from compliance and security to analytics and application-specific needs.

I’ve seen that in a multi-account environment, whether the accounts belong to the same line of business or multiple business units, collecting logs in a central, dedicated logging account is an established best practice. It helps security teams detect malicious activities both in real-time and during incident response. It provides protection to log data in case it is accidentally or intentionally deleted. It also helps application teams correlate and analyze log data across multiple application tiers.

Application Architecture

The solution uses Amazon Kinesis Data Streams and a log destination to set up an endpoint in the logging account to receive streamed logs and uses Amazon Kinesis Data Firehose to deliver log data to the Amazon Simple Storage Solution (S3) bucket. Application accounts will subscribe to stream all (or part) of their Amazon CloudWatch logs to a defined destination in the logging account via subscription filters.

Below is a diagram illustrating how the various services work together.



The following steps are involved in setting up the central-logging solution:

* Create an Amazon S3 bucket for your central logging in the logging account
* Create an AWS Lambda function for log data transformation and decoding in logging account
* Create a central logging stack as a logging-account destination ready to receive streamed logs and deliver them to S3
* Create a subscription in application accounts to deliver logs from a specific CloudWatch log group to the logging account destination
* Create Amazon Athena tables to query and analyze log data in your logging account