

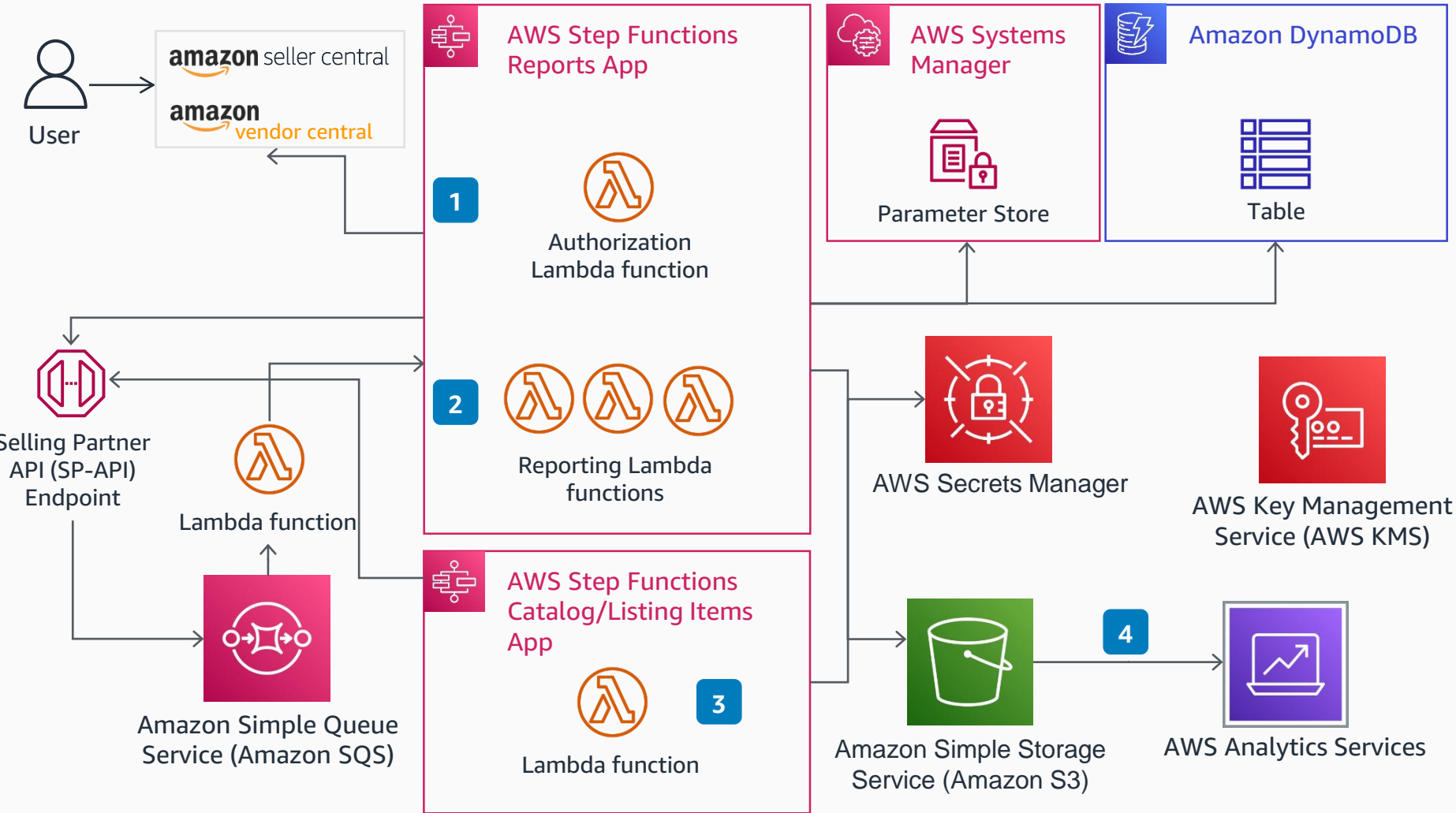
Guidance for Amazon Seller and Vendor Central Data Producer

This architecture uses serverless and managed services to help customers develop efficient and secure applications to integrate with Amazon Seller Central and Amazon Vendor Central. Customers can use this to ingest, process, and gain insights from their Amazon.com data.

Consumer Packaged Goods (CPG) companies who sell on Amazon.com need access to seller and vendor central data to manage orders, keep product catalogs updated, and to keep track of sales, shipments, and payments.

With this architecture, CPG companies can ingest data into a modern data platform to enable advanced analytics. Each of the following diagrams highlights a different aspect of this work, comprising:

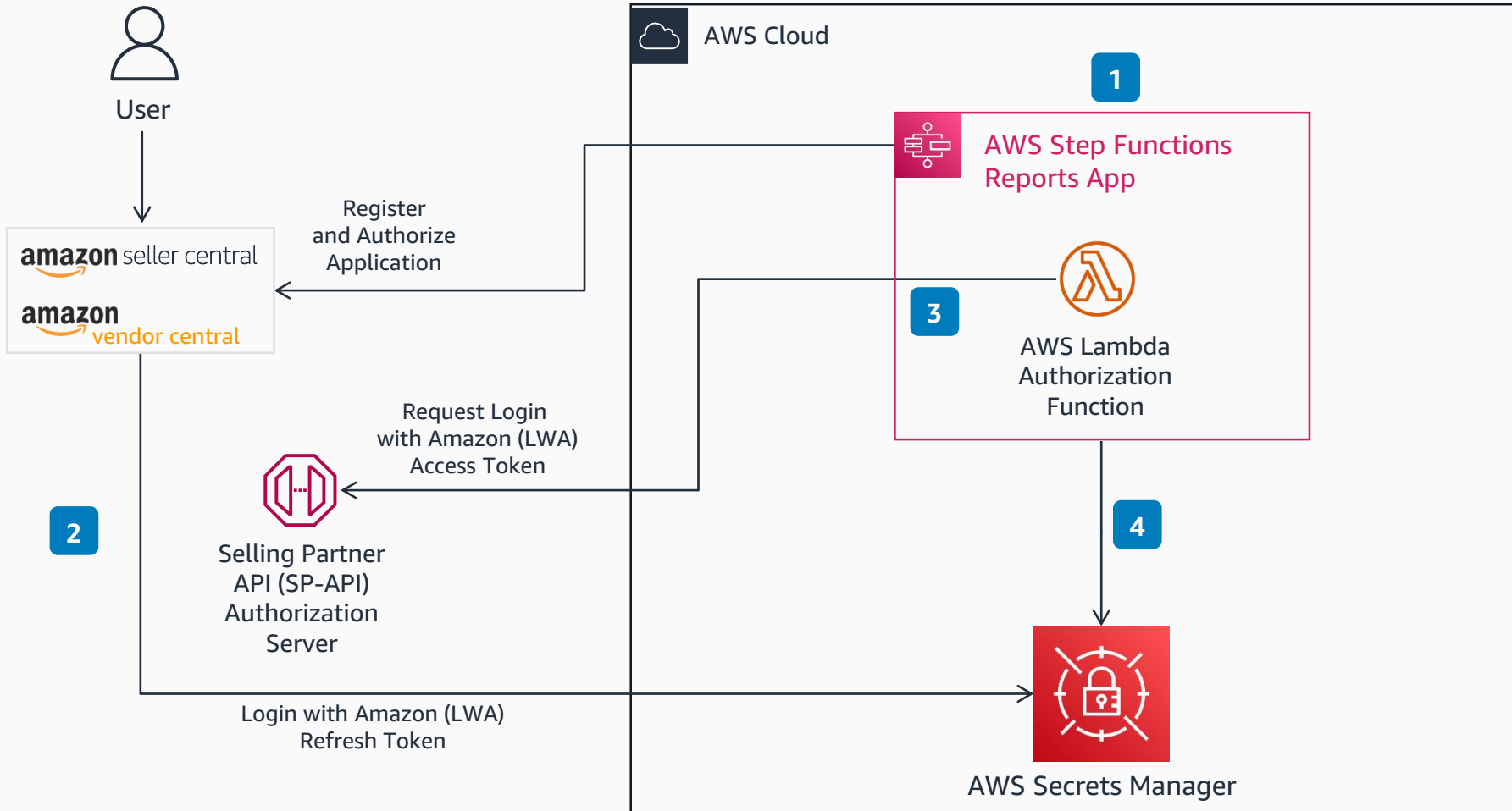
- 1 Authentication and Authorization
- 2 Serverless Reports Application
- 3 Serverless Catalog Items and Listing Items Applications
- 4 Data Storage, Movement, and Insights



Guidance for Amazon Seller and Vendor Central Data Producer

Authentication and Authorization

This architecture shows the registration process with Amazon Seller Central and Amazon Vendor Central. It includes: receiving a refresh token, secure token storage, exchanging for an access token, and using an authorized token to make API calls to the Selling Partner API (SP-API).



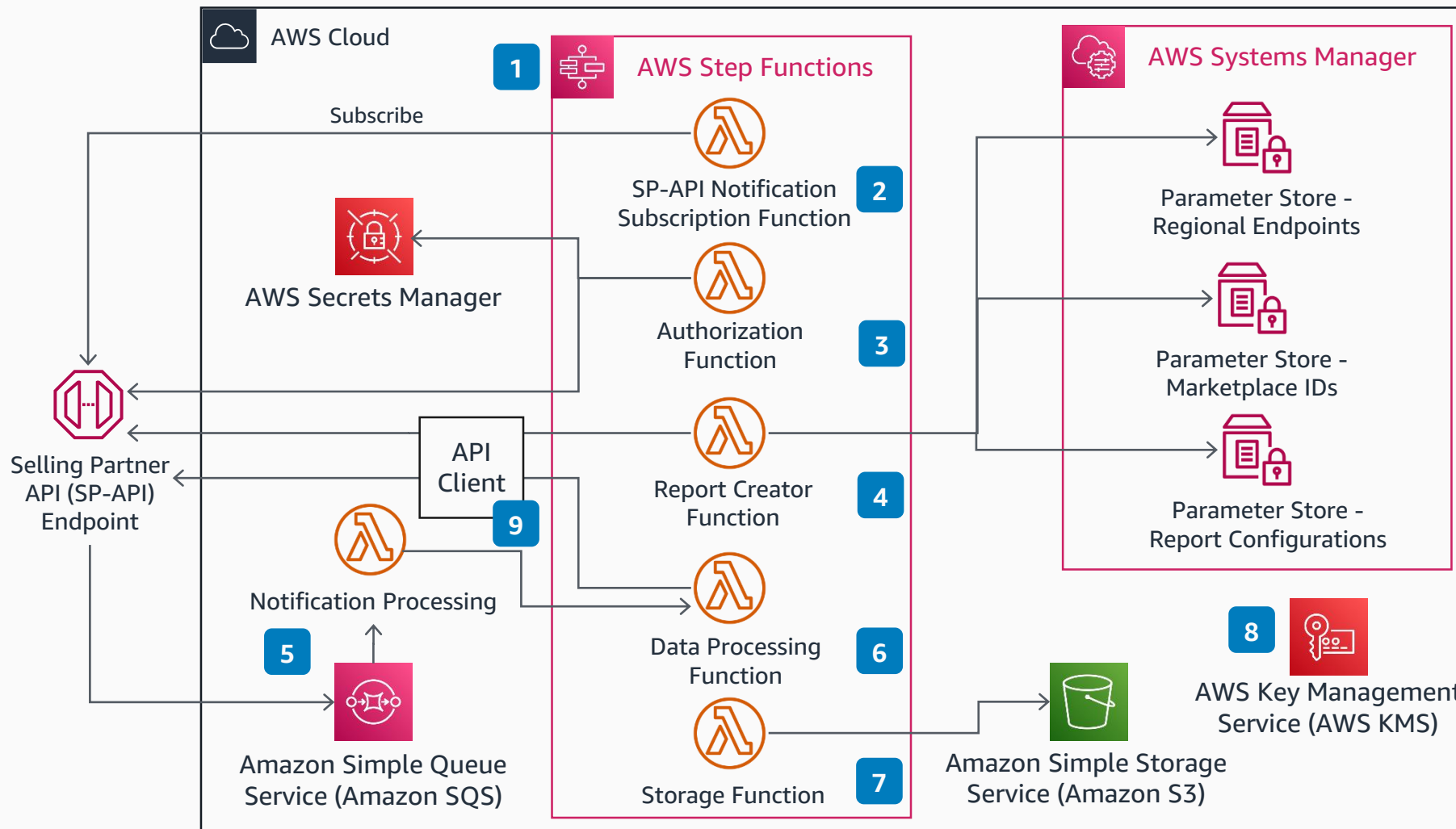
- 1** **AWS Step Functions** is used to create a serverless application to interact with the SP-API. The app is registered and authorized in Amazon Seller Central and Amazon Vendor Central.
- 2** Once authorized, you get a Login with Amazon (LWA) refresh token. The LWA refresh token is a long-lived token which you can store in **AWS Secrets Manager**.
- 3** In order to make API calls to the SP-API, your application needs an LWA access token. An **AWS Lambda** authentication function is used to first check with **Secrets Manager** to see if a valid LWA access token exists. If no valid LWA access token exists, the function retrieves an LWA refresh token from **Secrets Manager**, then exchanges the LWA refresh token for an LWA access token from the SP-API authentication server.
- 4** The LWA access token expires one hour after it is issued. To avoid having to retrieve an access token for each API call, you can cache the LWA access token in **Secrets Manager**, which can then be used for successive calls until expiry.



Guidance for Amazon Seller and Vendor Central Data Producer

Serverless Reports Application

This architecture shows how to create a serverless reports application to automatically ingest, process, and store data obtained from the Amazon Seller Central and Amazon Vendor Central Selling Partner API (SP-API).



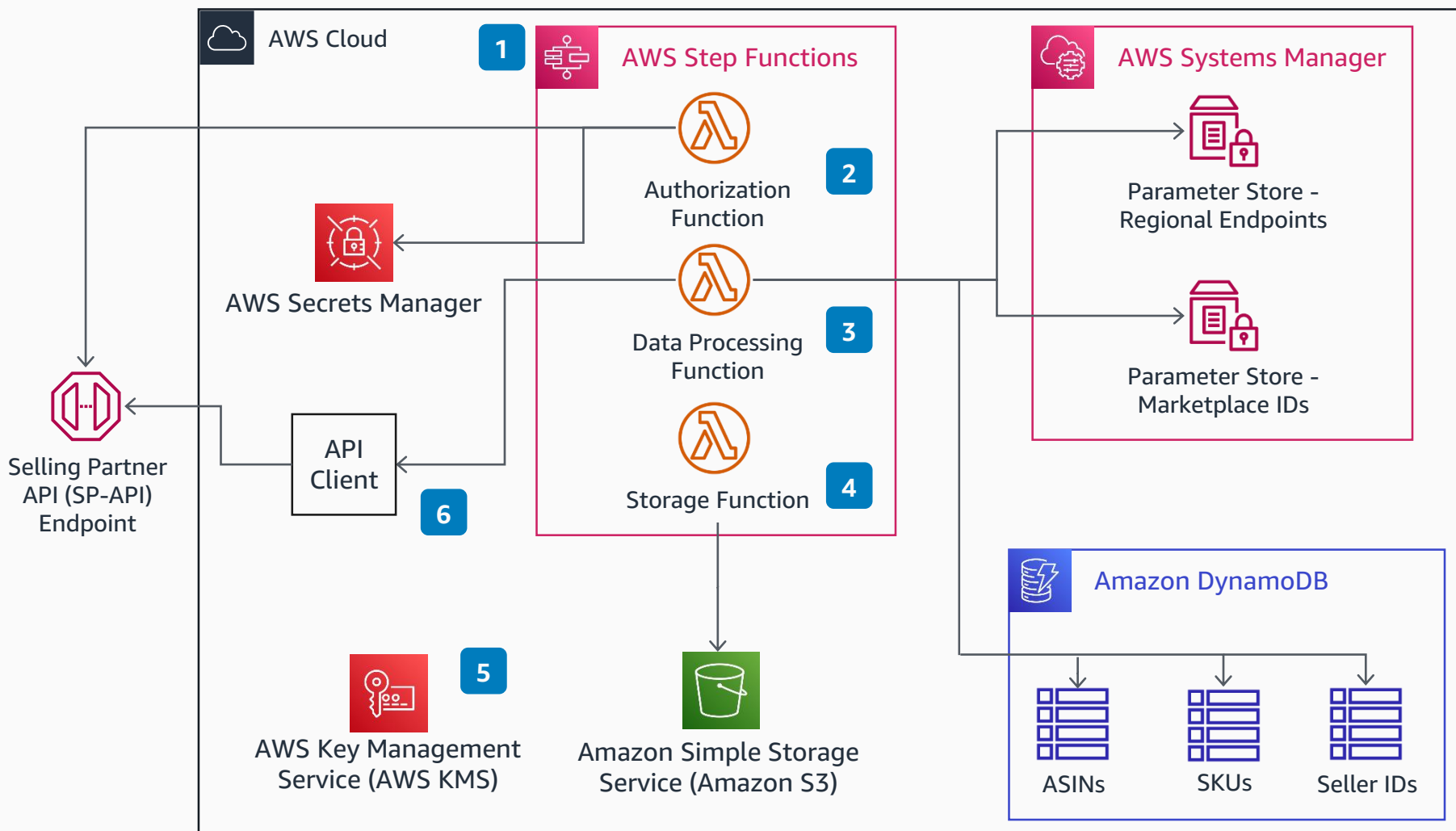
- 1 Step Functions is used as a serverless orchestration service to centrally manage the workflow for integrating with the Selling Partner API (SP-API).
- 2 The SP-API Reports API supports notifications to automate reports workflows. For this, a **Lambda** function is used to subscribe the application to the REPORT_PROCESSING_FINISHED notification type.
- 3 In order to make calls to the SP-API, an authentication **Lambda** function is used to obtain a Login with Amazon (LWA) access token as described in the previous Authentication and Authorization diagram.
- 4 The LWA access token from the authentication function is passed to a report creator **Lambda** function. This function uses regional endpoints, marketplace IDs, and report configurations data stored in **Parameter Store**, a capability of **AWS Systems Manager**, along with the LWA access token to make a createReport call to the SP-API.
- 5 The SP-API will then generate the report and upon completion, a REPORT_PROCESSING_FINISHED notification event is sent to a **Amazon Simple Queue Service** (Amazon SQS) queue, which provides information when report processing is CANCELLED, DONE, or FATAL. This triggers a **Lambda** function to process the event. If the notification event has a status of DONE, a reportDocumentId will be included.
- 6 The notification event is then passed to a data processing **Lambda** function in our **Step Functions** workflow. The data processing function uses the reportDocumentId to make a getReportDocument call to the SP-API. The SP-API returns a pre-signed URL for the location of the report document and the compression algorithm used, if the report document contents have been compressed.
- 7 This response is then passed to a storage **Lambda** function which downloads the report document, decompresses it if applicable, and stores the report document in **Amazon Simple Storage Service** (Amazon S3).
- 8 **AWS Key Management Service** (AWS KMS) is used to centrally manage encryption keys, which can be used to encrypt our secrets in **Secrets Manager**. Data is stored in **Amazon S3** and **Parameter Store**.
- 9 SP-API requests are limited using the token bucket algorithm, so an API client is recommended for rate limiting.



Guidance for Amazon Seller and Vendor Central Data Producer

Serverless Catalog Items and Listing Items Applications

This architecture shows how to create a serverless application to integrate with the Catalog Items and Listing Items APIs from the Amazon Seller Central and Amazon Vendor Central Selling Partner API.



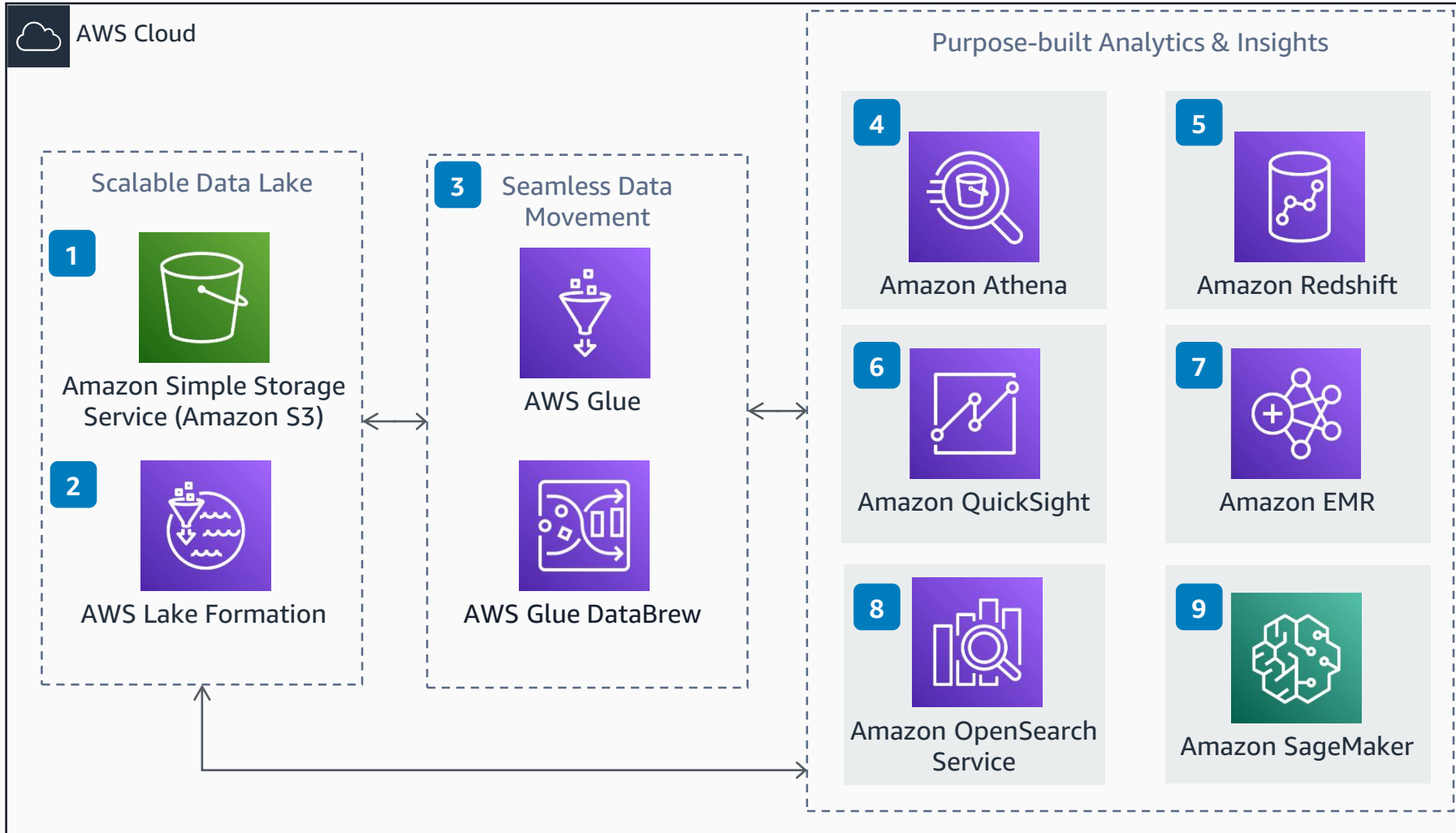
- 1** **Step Functions** is used as a serverless orchestration service to centrally manage the workflow for integrating with the Selling Partner API (SP-API).
- 2** In order to make calls to the SP-API, an authentication **Lambda** function is used to obtain a Login with Amazon (LWA) access token as described in the previous Authentication and Authorization diagram.
- 3** The LWA access token from the authentication function is passed to a data processing **Lambda** function. This function uses regional endpoints and marketplace IDs stored in **Parameter Store**, and ASINs, SKUs, and Seller IDs stored in **Amazon DynamoDB** along with the LWA access token to make an API call to the Catalog Items or Listing Items API of the SP-API.
- 4** When a response is returned, it is then passed to a storage **Lambda** function which stores the data in **Amazon S3**.
- 5** **AWS KMS** is used to centrally manage encryption keys, which can be used to encrypt our secrets in **Secrets Manager** and our data stored in **Amazon S3**, **DynamoDB**, and **Parameter Store**.
- 6** SP-API requests are limited using the token bucket algorithm, so an API client is recommended for rate limiting.



Guidance for Amazon Seller and Vendor Central Data Producer

Data Storage, Movement, and Insights

This architecture shows how to build a data analytics pipeline using a Modern Data Analytics approach to derive insights from the data.



- 1** **AWS Lake Formation** is used to build the scalable data lake, and **Amazon S3** is used as the data lake storage.
- 2** **Lake Formation** is also used to enable unified governance to centrally manage the security, access control, and audit trails.
- 3** **AWS Glue** and **AWS Glue DataBrew** are used to catalog, transform, enrich, move, and replicate data across multiple data stores and the data lake.
- 4** **Amazon Athena** enables interactive querying, analyzing, and processing capabilities.
- 5** **Amazon Redshift** is used as a Cloud Data Warehouse.
- 6** **Amazon QuickSight** provides machine learning-powered business intelligence.
- 7** **Amazon EMR** provides the cloud big data platform for processing vast amounts of data using open source tools.
- 8** **Amazon OpenSearch Service** can be used for operational analytics.
- 9** **Amazon SageMaker** can be used to build, train, and deploy machine learning models, and add intelligence to your applications.

