Guidance for Data Transfer Hub on AWS
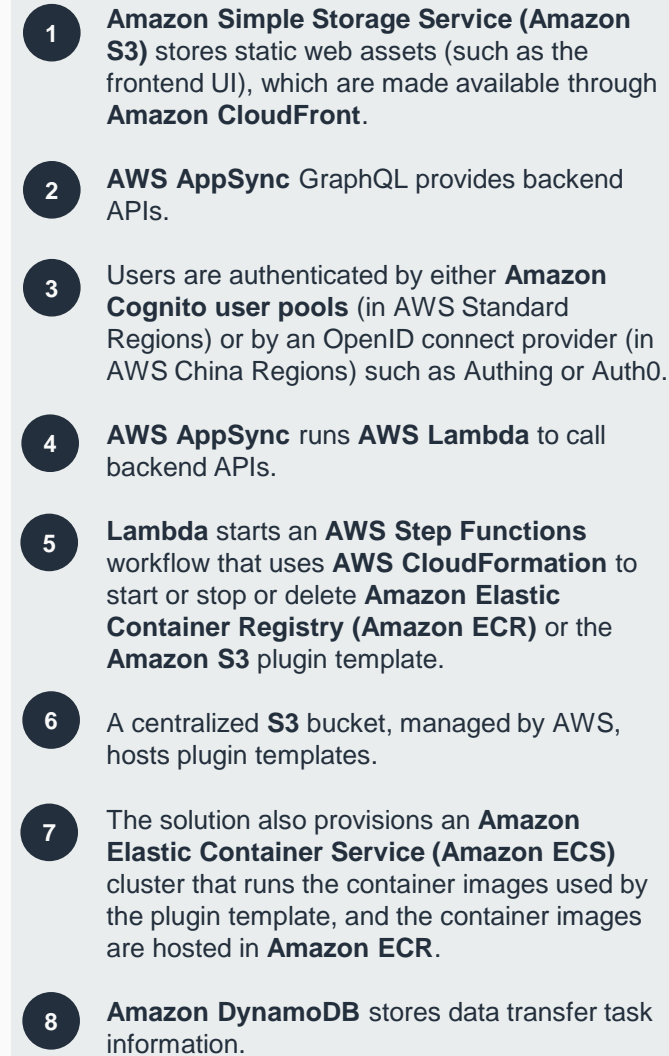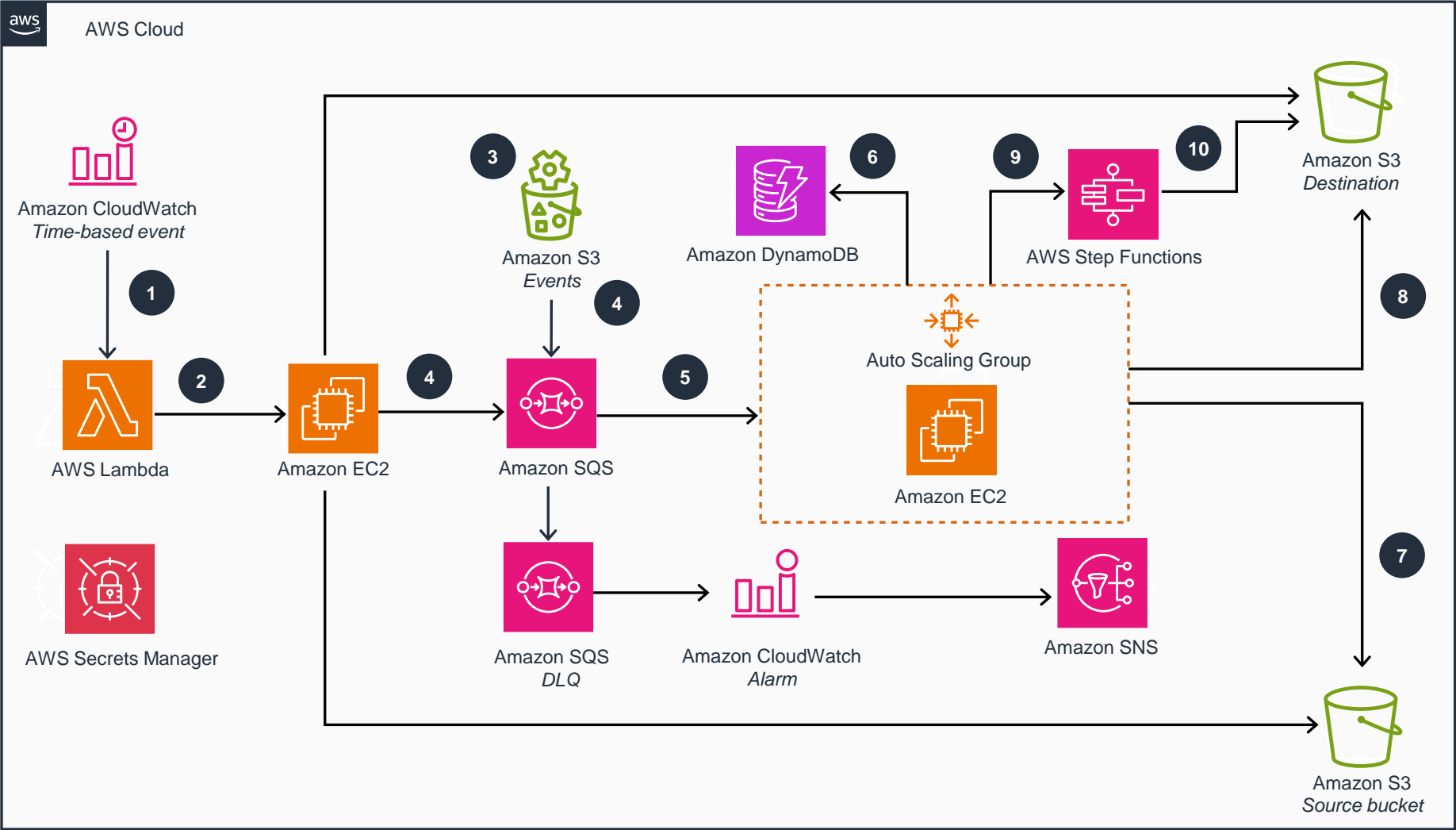
Overview architecture

This architecture diagram illustrates how to secure, scale, and track data transfer for Amazon S3 objects and Amazon ECR images.

1. **Amazon Simple Storage Service (Amazon S3)** stores static web assets (such as the frontend UI), which are made available through **Amazon CloudFront**.

2. **AWS AppSync** GraphQL provides backend APIs.

3. Users are authenticated by either **Amazon Cognito user pools** (in AWS Standard Regions) or by an OpenID connect provider (in AWS China Regions) such as Authing or Auth0.

4. **AWS AppSync** runs **AWS Lambda** to call backend APIs.

5. **Lambda** starts an **AWS Step Functions** workflow that uses **AWS CloudFormation** to start or stop or delete **Amazon Elastic Container Registry (Amazon ECR)** or the **Amazon S3** plugin template.

6. A centralized **S3** bucket, managed by AWS, hosts plugin templates.

7. The solution also provisions an **Amazon Elastic Container Service (Amazon ECS)** cluster that runs the container images used by the plugin template, and the container images are hosted in **Amazon ECR**.

8. **Amazon DynamoDB** stores data transfer task information.

AWS Reference Architecture

Guidance for Data Transfer Hub on AWS

Amazon S3 transfer option

This architecture diagram illustrates how run the Amazon S3 plugin to transfer objects from their sources into S3 buckets. This slide shows Steps 1-8.


AWS Cloud

Amazon CloudWatch
Time-based event

AWS Lambda

AWS Secrets Manager

Amazon EC2

Amazon S3
Events

Amazon SQS

Amazon SQS
DLQ

Amazon DynamoDB

Amazon CloudWatch
Alarm

Amazon SNS

Auto Scaling Group

Amazon EC2

AWS Step Functions

Amazon S3
Destination

Amazon S3
Source bucket

1. A time-based **EventBridge** rule initiates the **Lambda** function on an hourly basis.

2. **Lambda** uses the launch template to launch a data comparison job (JobFinder) in **Amazon Elastic Compute Cloud (Amazon EC2)**.

3. The job lists all the objects in the source and destination **S3** buckets and makes comparisons among objects to determine which objects should be transferred.

4. **Amazon EC2** sends a message for each object that will be transferred to **Amazon Simple Queue Service (Amazon SQS)**. **Amazon S3** event messages can also be supported for more real-time data transfer. Whenever an object is uploaded to the source bucket, the event message is sent to the same **Amazon SQS** queue.

5. A JobWorker node running in **Amazon EC2** consumes the messages in **Amazon SQS** and transfers the object from the source bucket to the destination bucket. You can use an Auto Scaling group to control the number of **EC2** instances to transfer the data based on business needs.

6. **DynamoDB** stores a record with transfer status for each object.

7. The **EC2** instance will get (download) the object from the source bucket based on the **Amazon SQS** message.

8. The **EC2** instance will put (upload) the object to the destination bucket based on the **Amazon SQS** message.

AWS Reference Architecture

Guidance for Data Transfer Hub on AWS

Amazon S3 transfer option

This architecture diagram illustrates how run the Amazon S3 plugin to transfer objects from their sources into S3 buckets. This slide shows Steps 9-10.
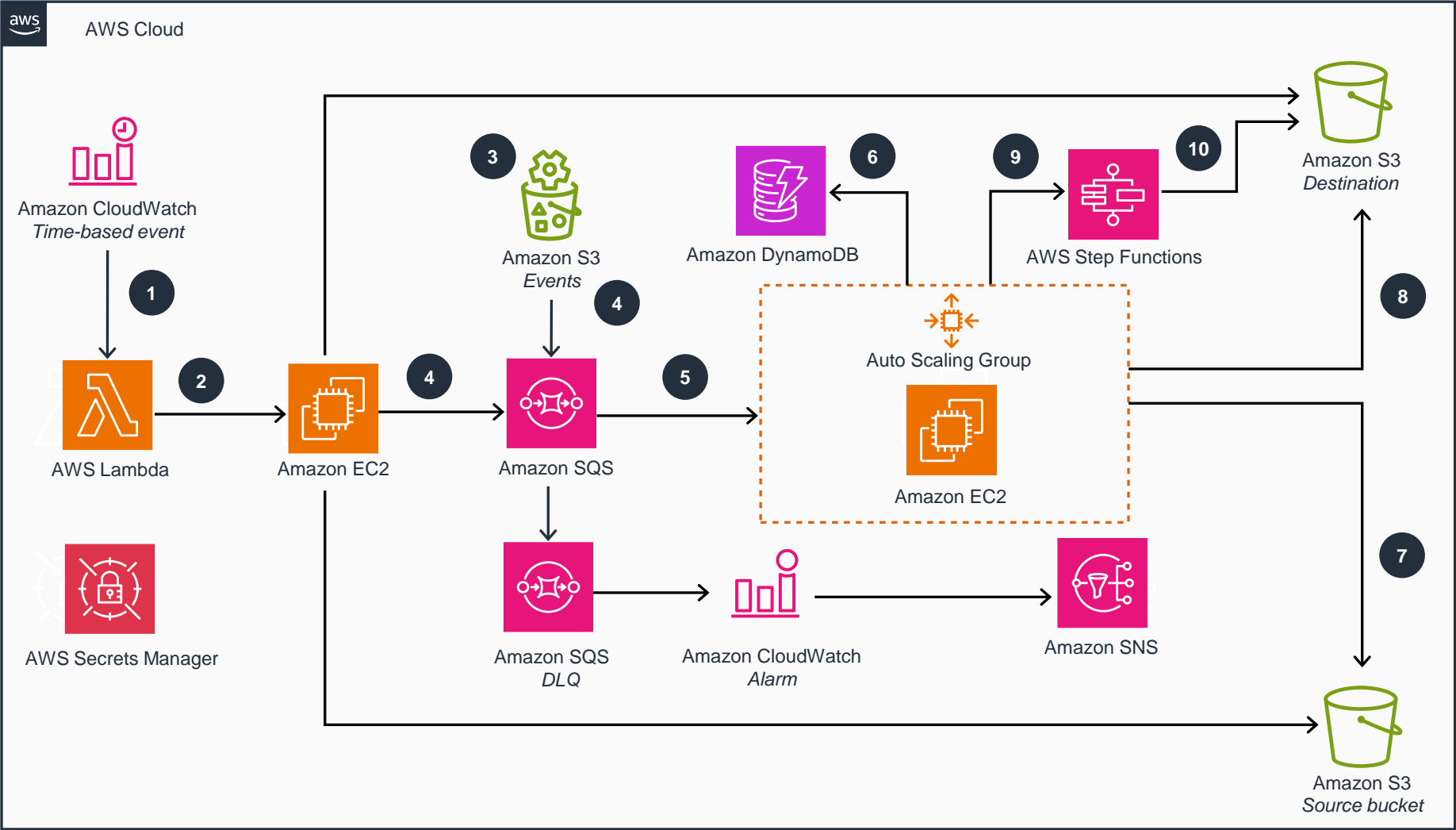

9 When the JobWorker node identifies a large file (with a default threshold of 1 GB) for the first time, a multipart upload task running in **Amazon EC2** is initiated. The corresponding UploadId is then conveyed to **Step Functions**, which invokes a scheduled recurring task. Every minute, **Step Functions** verifies the successful transmission of the distributed shards associated with the UploadId across the entire cluster

10 If all shards have been transmitted successfully, **Amazon EC2** invokes the CompleteMultipartUpload API in **Amazon S3** to finalize the consolidation of the shards. Otherwise, any invalid shards are discarded.

AWS Reference Architecture

Guidance for Data Transfer Hub on AWS

Amazon ECR transfer option - Pull method

This architecture diagram illustrates how to run the Amazon ECR plugin to transfer container images from other container registries.
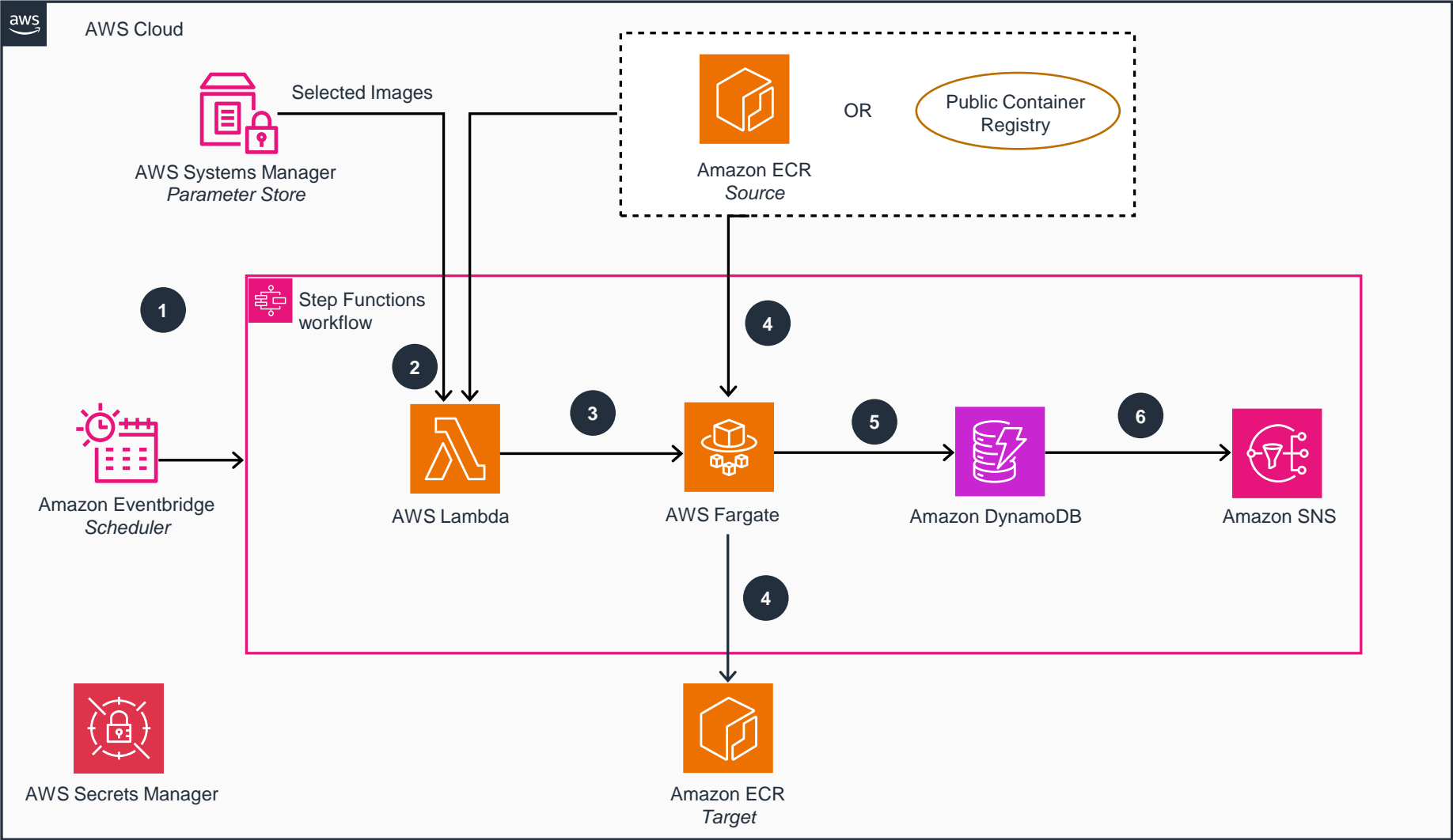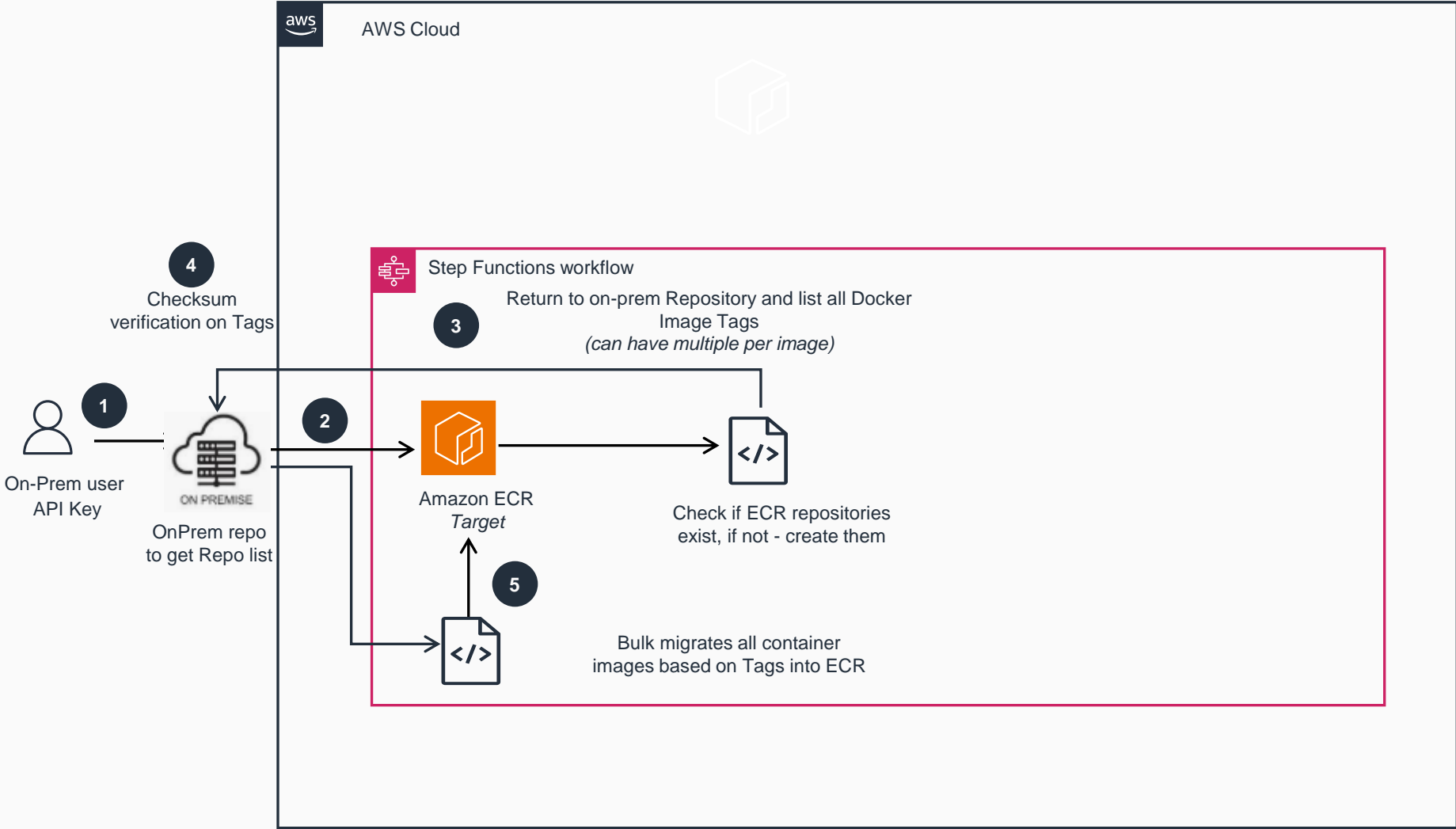

1. An **Amazon EventBridge** rule runs a **Step Functions** workflow on a regular basis (by default, it runs daily).

2. **Step Functions** invokes **Lambda** to retrieve the list of images from the source.

3. **Lambda** will either list all the repository content in the source **Amazon ECR** or get the stored image list from Parameter Store, a capability of **AWS System Manager**.

4. The transfer task will run within **AWS Fargate** in a maximum concurrency of 10. If a transfer task fails for some reason, it will automatically retry three times.

5. Each task uses skopeo to copy the images into the target **Amazon ECR** registry.

6. After the copy completes, the status (either success or fail) is logged into **DynamoDB** for tracking purposes.

AWS Reference Architecture

Guidance for Data Transfer Hub on AWS

Amazon ECR transfer option - Push method from on-prem
This architecture diagram illustrates how to bulk migrate container images from Local On-Prem repository to Amazon ECR

AWS Cloud

1 On-Prem user API Key

4 Checksum verification on Tags

OnPrem repo to get Repo list

Step Functions workflow

3 Return to on-prem Repository and list all Docker Image Tags
(can have multiple per image)

2 Amazon ECR *Target*

Check if ECR repositories exist, if not - create them

5 Bulk migrates all container images based on Tags into ECR

1 Guidance code makes an API call to on-prem repository and lists all user repositories.

2 Guidance code makes an API call to the target **Amazon ECR** using credentials configured by **AWS CLI** and checks if the list of on-prem repositories exists in ECR; if not, it creates them

3 Guidance code returns to on-prem repository and tallies all docker image tags in all repositories

4 Guidance code performs a Checksum verification of image tags in ECR: if a tag exists in ECR and the checksum matches, it is left alone.

5 Guidance code migrates all Docker container images in bulk to a target Amazon ECR

AWS Reference Architecture