
AWS
re:Invent

ANT407-R

Real-time apps with Amazon Kinesis Data Analytics and Apache Flink

Kamal Lanka
Software Development Manager
Amazon Web Services

AWS re:Invent

aws

Agenda

- Challenges of running streaming applications
- Apache Flink
- Configure and deploy an Apache Flink application using Amazon Kinesis Data Analytics

Challenges of running streaming applications

- Availability
  - Much higher requirements on the availability of the streaming applications when compared to traditional batch-based approach
- State management
  - Computations of a streaming application often rely on internal state that can be corrupted or even lost when the application fails
- Scaling
  - When the load increases, the infrastructure that supports the streaming application must scale to keep the application from becoming overloaded
  - When the load decreases, the infrastructure should scale down to remain cost effective by not provisioning more resources than are needed

Apache Flink

- Apache Flink is an open-source project that is tailored to stateful computations over unbounded and bounded datasets
  - Support for APIs (including Java and SQL) and rich time semantics
  - In-depth state management capabilities
  - Provides exactly-once processing support
  - Well suited for analyzing streaming data with low latency

Deploy and operate Apache Flink application with Kinesis Data Analytics for Java applications

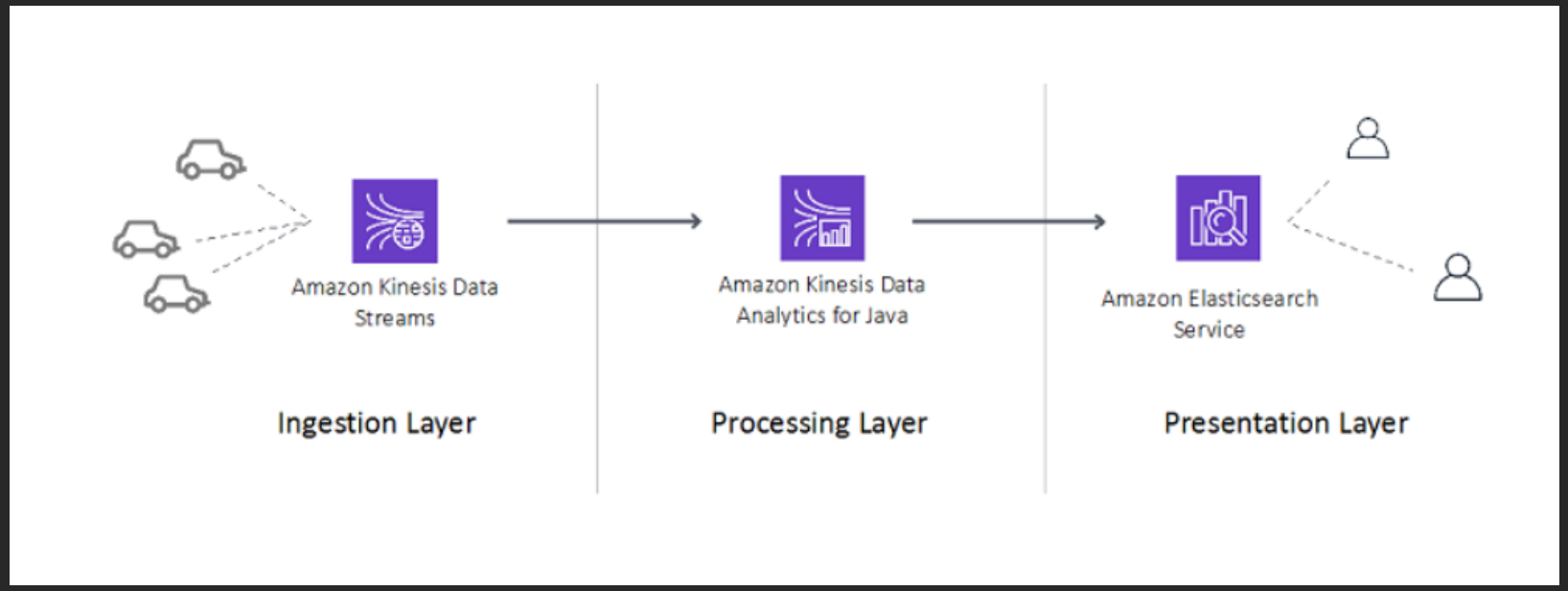
Scenario

- Analyze the telemetry data of a taxi fleet in New York City in near real time to optimize the fleet operation

Goal

- Identify areas that are currently requesting a high number of taxi rides

Architecture



- Ingestion layer: Kinesis data stream serves as a buffer that decouples the producers from the consumers
- Processing layer: Kinesis data analytics for Java processes the data in the Kinesis data stream
- Presentation layer: Persists the processed data into Amazon ES

Benefits of the architecture

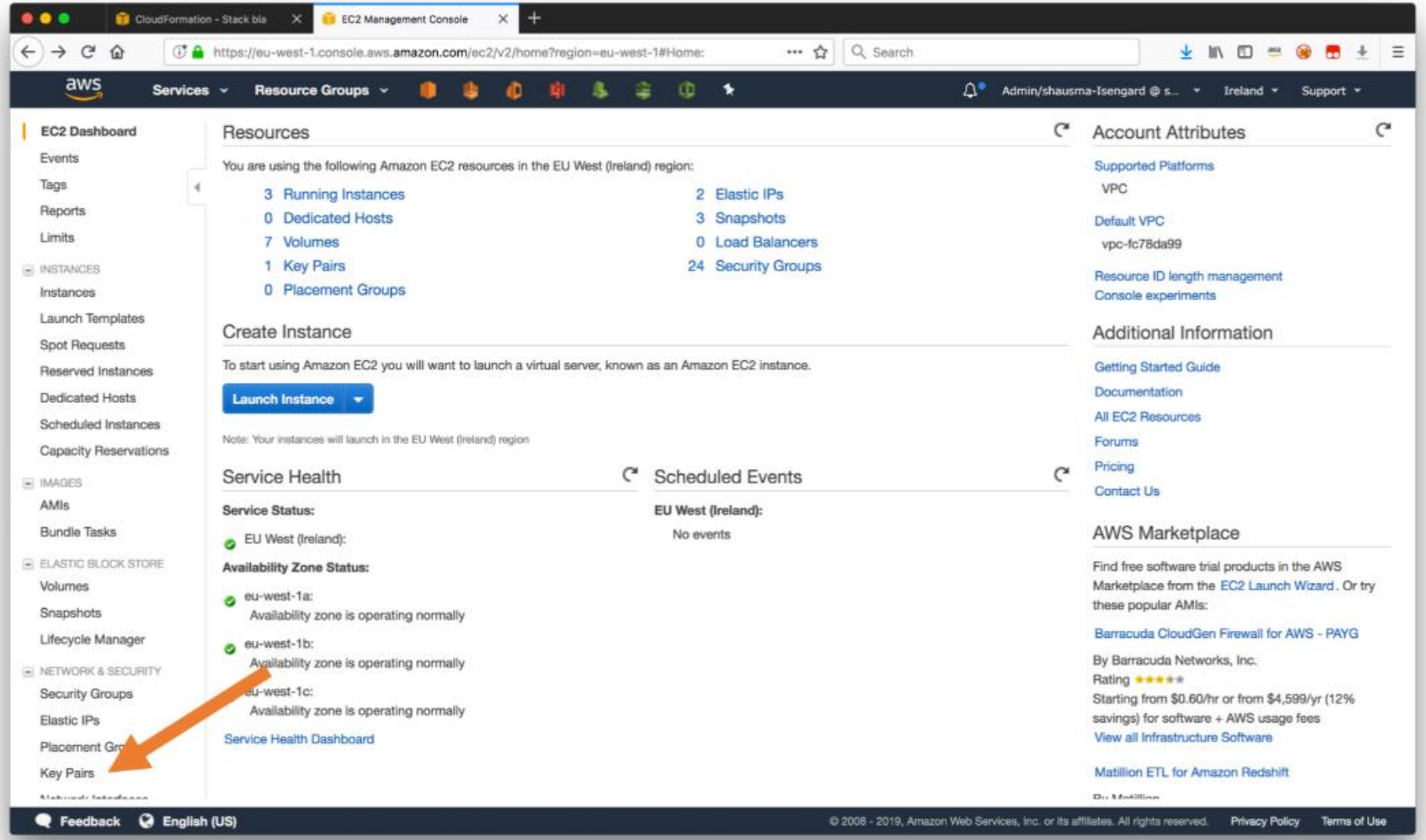
- Separating the different aspects of the architecture into ingestion, processing, and presentation nicely decouples different components
- Components can be scaled independently
- Allows you to experiment and adopt new technologies in the future

Let’s kick the tires!

Prerequisites

- Access to an AWS account with **AdminstratorAccess**

Create EC2 Key Pair



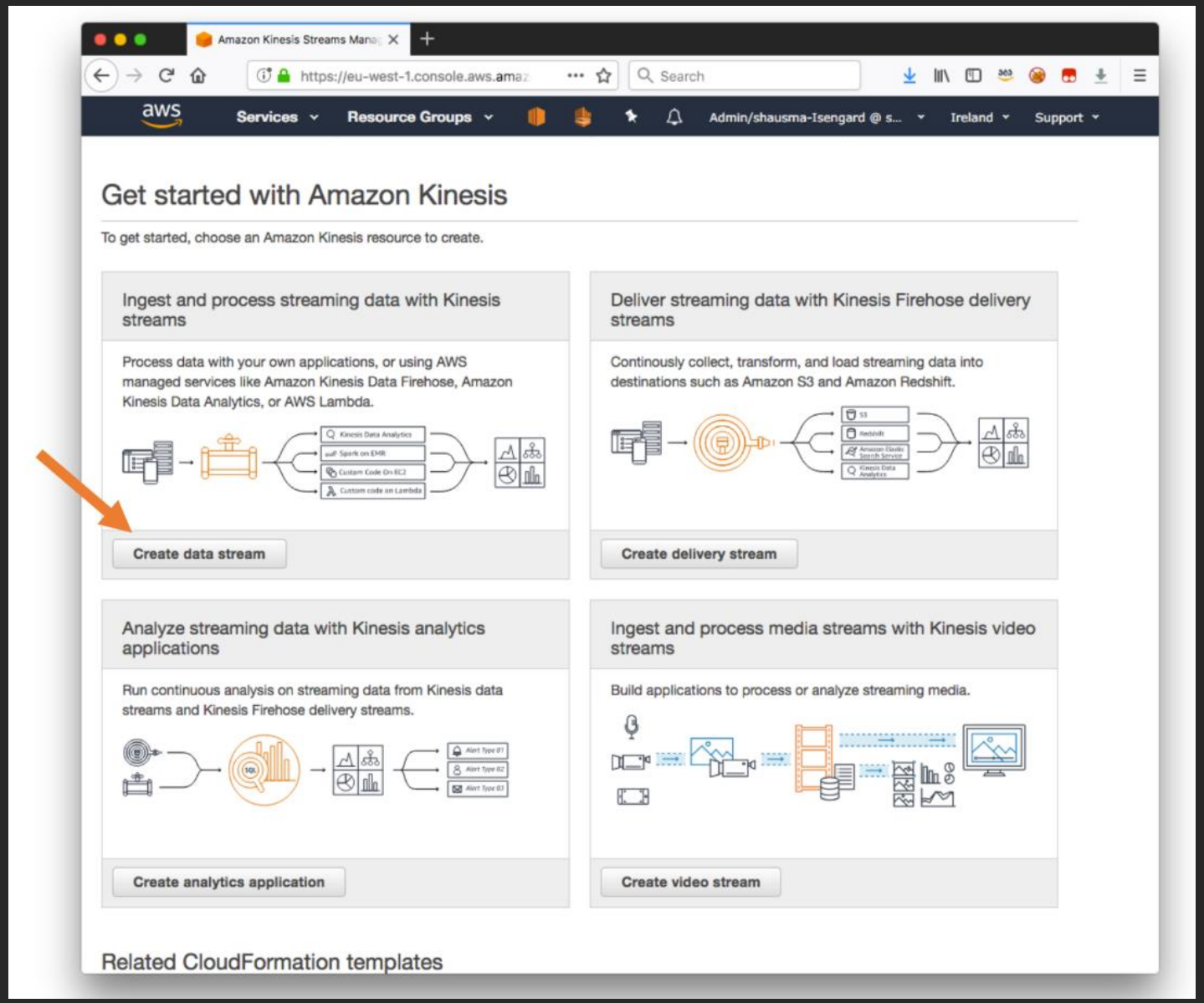
Run AWS CloudFormation

- Execute AWS CloudFormation template in your own AWS account
  - This steps will take up to 10 minutes
  - Build and generate Kinesis Data Analytics Apache Flink Jar file
  - Creates Amazon ES cluster for presentation layer
  - Provisions an EC2 instance to ingest data

- Navigate to the **Outputs** section of the CloudFormation template and take a note of the outputs. We will need them to complete the subsequent steps.

Ingestion layer

Create Kinesis Data Stream—taxi-trip-events

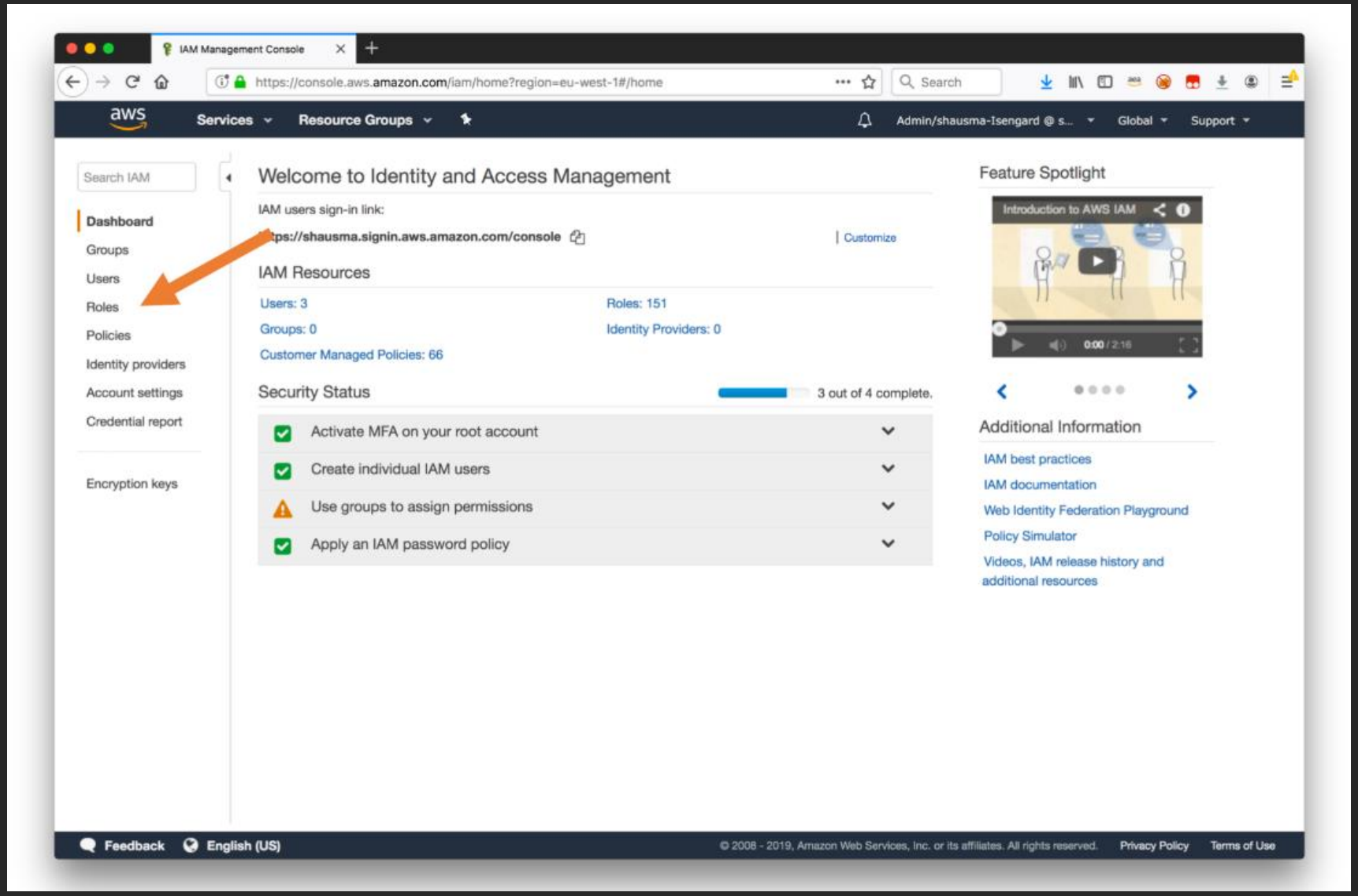


Processing layer

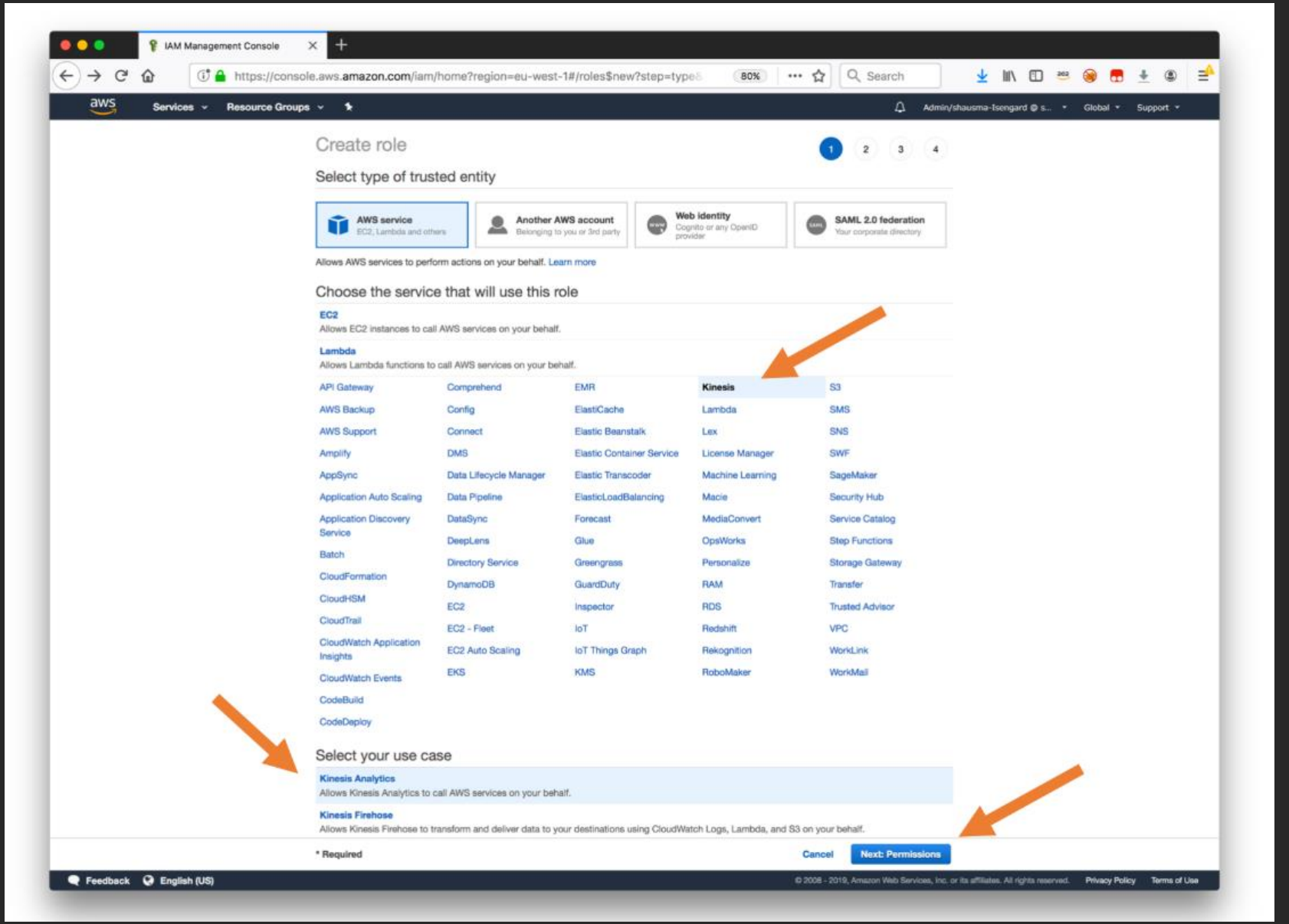
Steps

1. Create IAM role to call services
2. Create Kinesis Data Analytics application

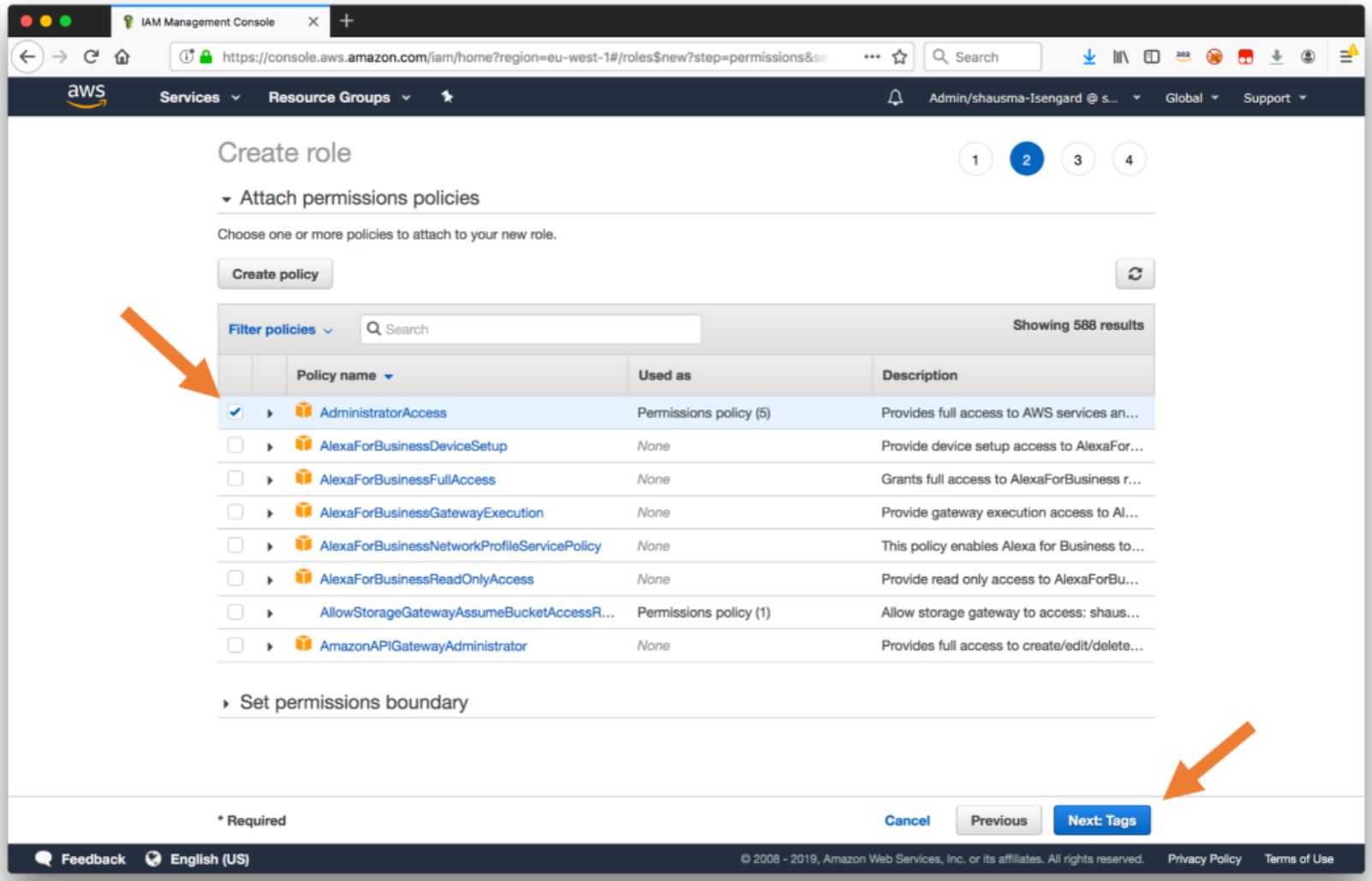
Step 1: Create IAM role to call services



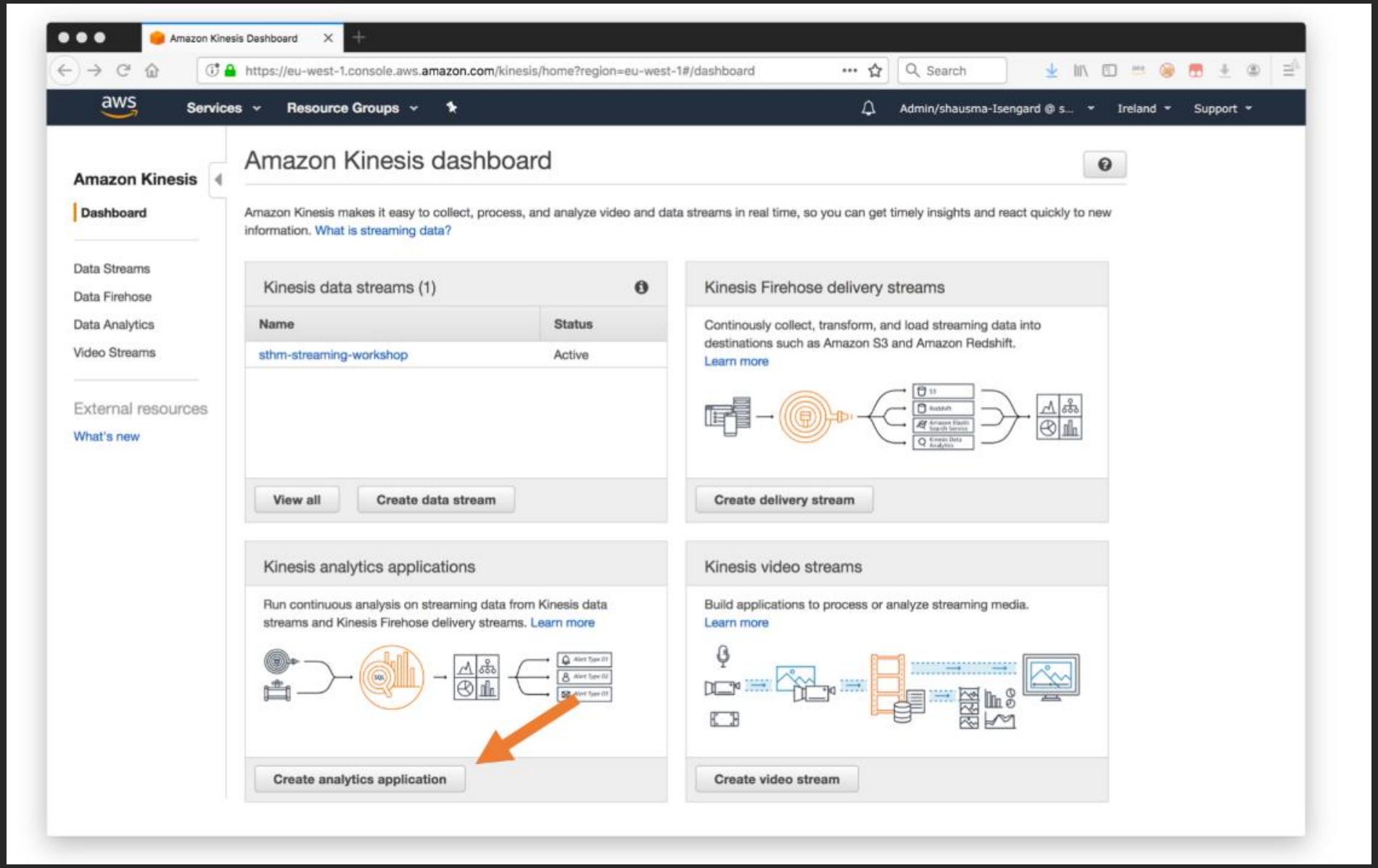
Step 1: Create IAM role to call services



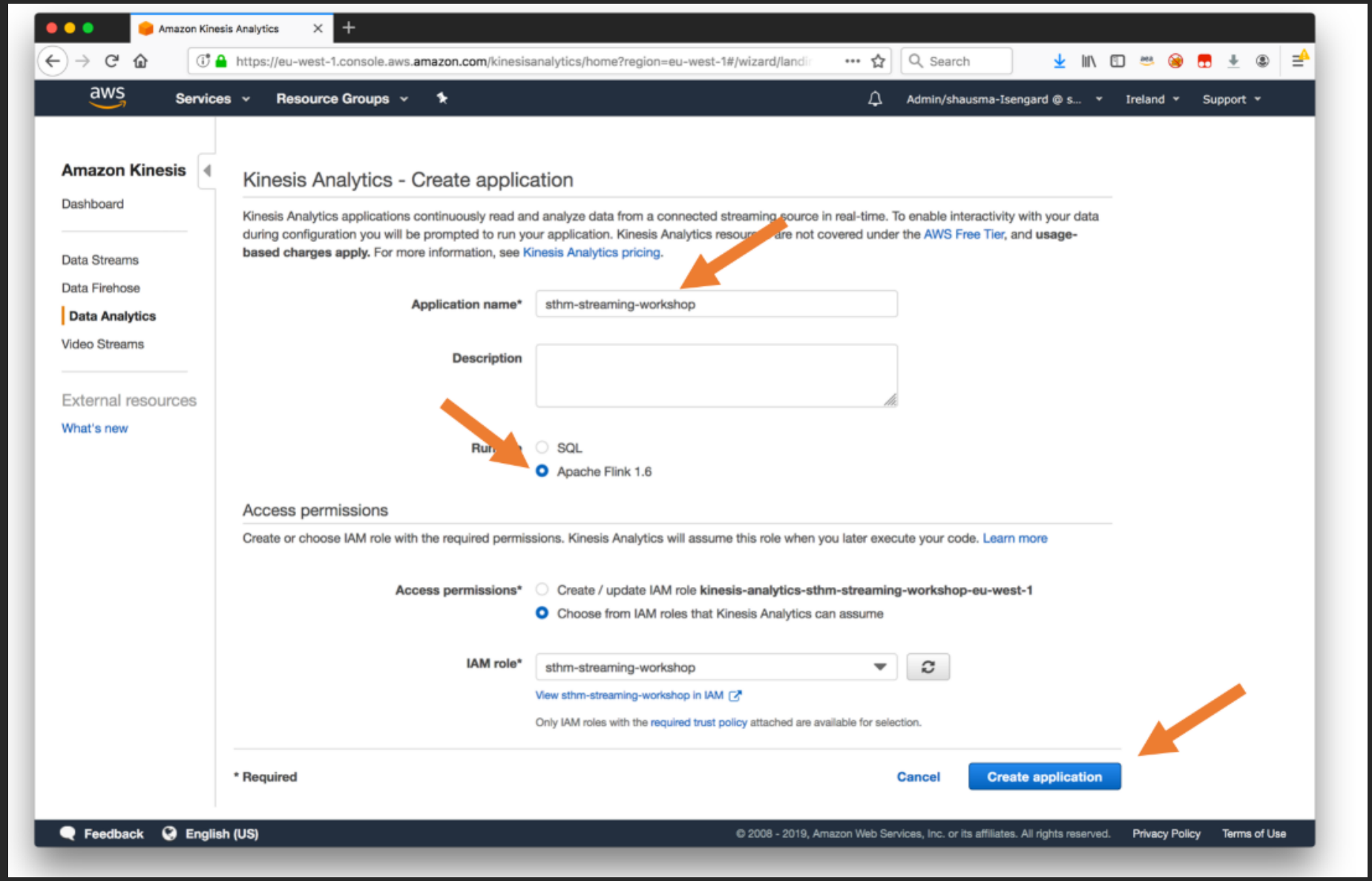
Step 1: Create IAM role to call services



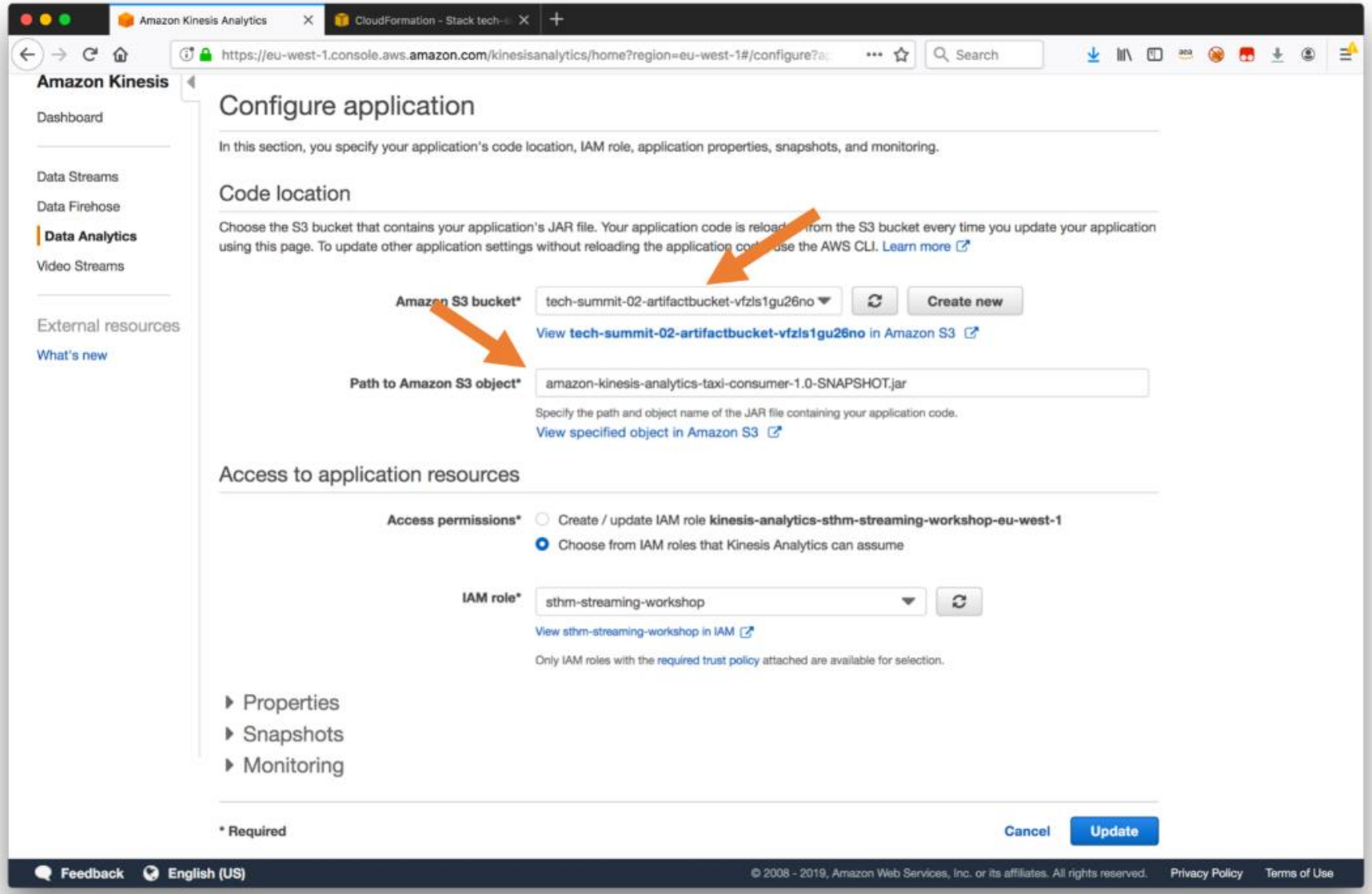
Step 2: Create Kinesis Data Analytics application



Step 2: Create Kinesis Data Analytics application



Step 2: Create Kinesis Data Analytics application



Step 2: Create Kinesis Data Analytics application

- Expand the Properties section
  - Enter Group ID : FlinkApplicationProperties
  - Add two key/value pairs:
    - InputStreamName with the name of the Kinesis stream you've created earlier, e.g., taxi-trip-events
    - ElasticsearchEndpoint with the correct Amazon ES https endpoint that can be obtained from the Output section of the CloudFormation template under ElasticsearchEndpoint

Step 2: Create Kinesis Data Analytics application

- Finally, press the blue Update button at the bottom of the page to update the properties of the application
- Once the update has completed, press Run on the resulting page and confirm that you want to run the application by choosing Run again

Ingest data

Ingest data into a Kinesis Data Stream

- Connect to the EC2 instance via browser SSH
- Once the connection has been established, start ingesting events into the Kinesis data stream by executing the jar file that has already been downloaded to the EC2 instance
  - java -jar amazon-kinesis-replay-1.0-SNAPSHOT.jar -streamRegion us-east-1 -speedup 1000 -streamName taxi-trip-events

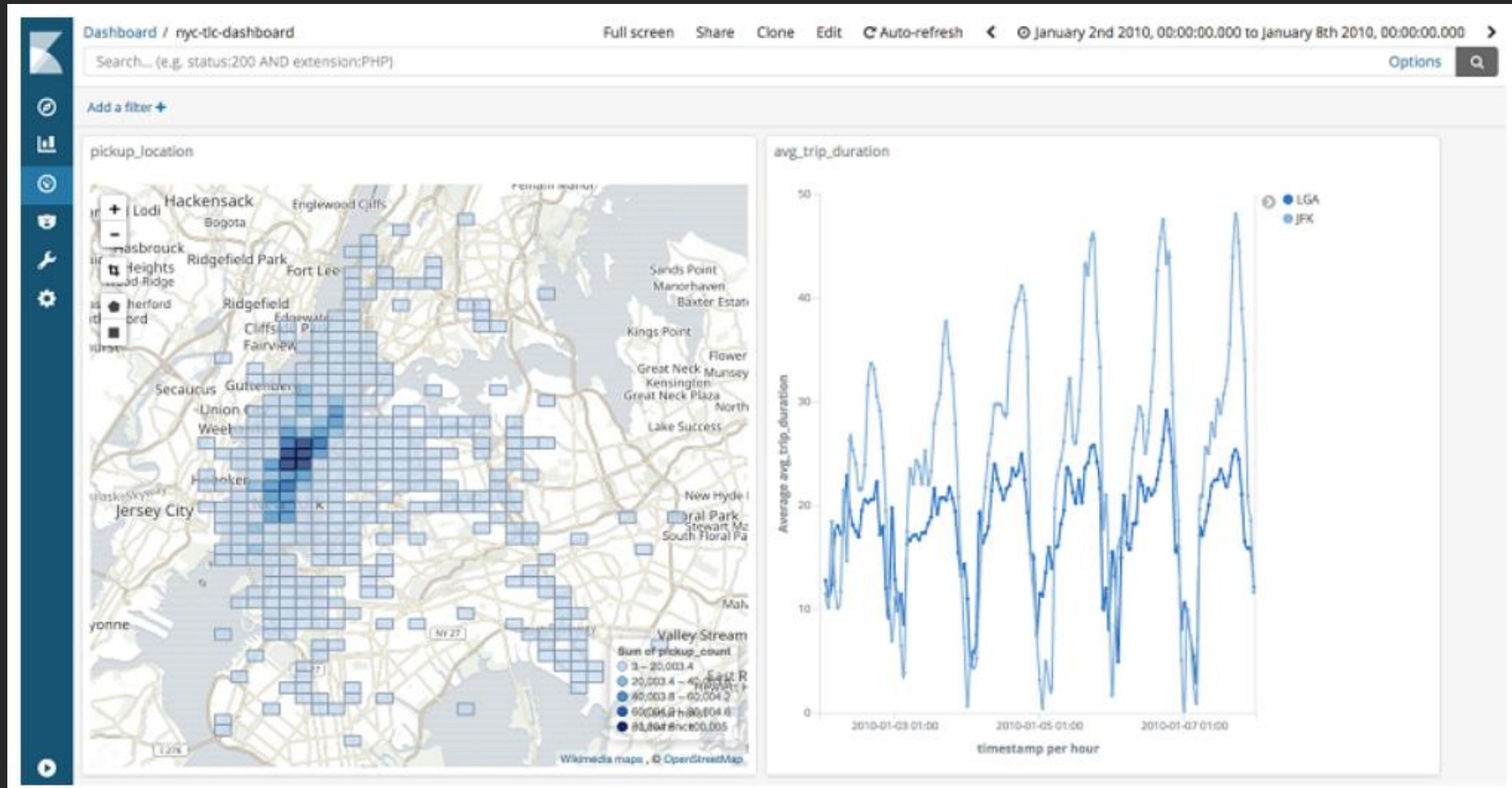
Visualizing data

AWS re:Invent

aws

Analyze the data

- Navigate to the Kibana dashboard—the URL can be obtained from the Output section of the CloudFormation template under KibanaDashboard
- The Kibana dashboard contains a heatmap and a line graph
  - The heatmap visualizes locations where taxis are currently requested
  - The line graph visualizes the average trip duration between to these two airports


Dashboard / nyc-tlc-dashboard
Full screen
Share
Clone
Edit
Auto-refresh
January 2nd 2010, 00:00:00.000 to January 8th 2010, 00:00:00.000
Search... (e.g. status:200 AND extension:PHP)
Options
Add a filter +
pickup_location
Hackensack
Englewood Cliffs
Bogota
Ridgefield Park
Fort Lee
Ridgefield
Fairview
Secaucus
Union
Jersey City
Sands Point
Kings Point
Great Neck
Great Neck Plaza
Lake Success
Valley Stream
Sum of pickup_count
Wikimedia maps , © OpenStreetMap
avg_trip_duration
LGA
JFK
Average avg_trip_duration
timestamp per hour
2010-01-03 01:00
2010-01-05 01:00
2010-01-07 01:00

Processing layer Java code walkthrough

Let's step back for a moment and review what you just did

- You created a fully managed, highly available, scalable streaming architecture
- You ingested and analyzed up to 10k events per second
  - Try to imagine what it would have taken you to build something similar from scratch

Cleanup

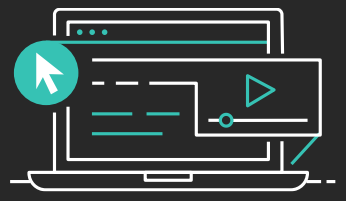
Cleanup

- Delete the Kinesis Data Analytics Java app
- Delete the Kinesis stream
- Delete the IAM role you created earlier
- Delete the CloudFormation stack. If it fails, delete the S3 bucket and delete the stack again.

Q&A

Learn big data with AWS Training and Certification

Resources created by the experts at AWS to help you build and validate data analytics skills



New free digital course, Data Analytics Fundamentals, introduces Amazon S3, Amazon Kinesis, Amazon EMR, AWS Glue, and Amazon Redshift

Classroom offerings, including Big Data on AWS, feature AWS expert instructors and hands-on labs

Validate expertise with the **AWS Certified Big Data – Specialty** exam or the new **AWS Certified Data Analytics – Specialty** beta exam

Visit aws.amazon.com/training/paths-specialty/

aws training and certification

Thank you!

AWS re:Invent

aws

Please complete the session survey in the mobile app.