Big Data on AWS

Lab Five: Visualizing Web Traffic Using Amazon Kinesis Data Streams

Table of Contents

[Lab Five: Visualizing Web Traffic Using Amazon Kinesis Data Streams 3](#_Toc509608439)

[Task 1: Start the Sample Application 3](#_Toc509608440)

[Task 2: View the Components of the Sample Application 3](#_Toc509608441)

[Data Producer 4](#_Toc509608442)

[Data Consumer: KCL 5](#_Toc509608443)

[Task 3: Create Kinesis Firehose Delivery Stream 6](#_Toc509608444)

[Task 4: Install and configure Kinesis Agent 6](#_Toc509608445)

[Task 5: Delete Sample Application 8](#_Toc509608446)

# Lab Five: Visualizing Web Traffic Using Amazon Kinesis Data Streams

In this lab, we will familiarize ourselves with Amazon Kinesis Data Streams concepts like Streams, Producers and Consumers, Kinesis Agent and Kinesis Firehose and so on.

Through this lab, you will perform the following tasks:

* Task 1: Start a sample application
* Task 2: View the components of the application
* Task 3: Create Kinesis Delivery Stream
* Task 4: Install and configure Kinesis agent
* Task 5: Delete the sample application

Note: This lab may incur nominal charges so, be sure to clean up after the lab is over.

## Task 1: Start the Sample Application

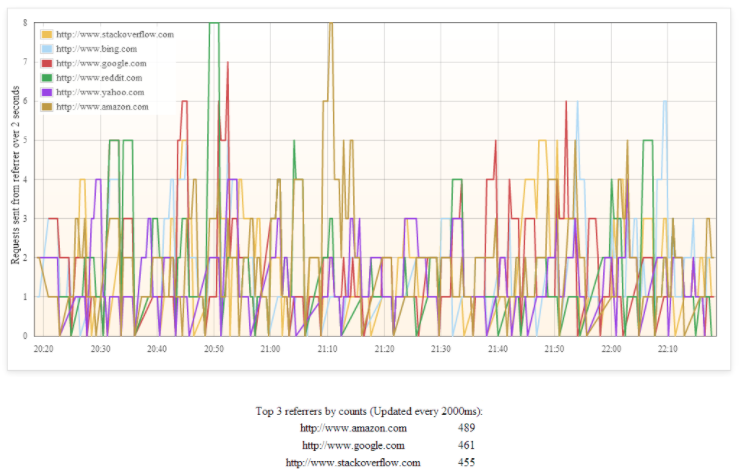
To start the application

1. Use [AWS CloudFormation template](https://console.aws.amazon.com/cloudformation/home?region=us-east-1" \l "/stacks/new?stackName=KinesisDataVisSampleApp&templateURL=https:%2F%2Fs3.amazonaws.com%2Fkinesis-demo-bucket%2Famazon-kinesis-data-visualization-sample%2Fkinesis-data-vis-sample-app.template" \t "_blank), Click **Next** on Select Template page.
2. Choose **t2.micro** on the **Specify Details** page. Choose **Next**.
3. Create tags if you want to and choose **Next**.
4. On the Review page, select I acknowledge that this template might cause AWS CloudFormation to create IAM resources, and then choose Create.
5. The stack can take several minutes to create. When the status is CREATE\_COMPLETE, continue to the next step. If the status does not update, refresh the page.

## Task 2: View the Components of the Sample Application

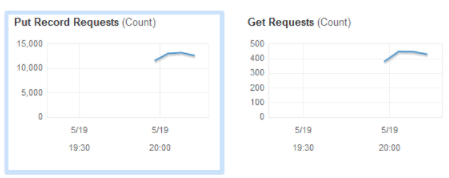
To view the stack details

1. Select the KinesisDataVisSample stack.
2. On the **Outputs** tab, choose the link in **URL**. The form of the URL should be similar to the following: <http://ec2-xx-xx-xx-xx.compute-1.amazonaws.com>. You might need to wait for a few minutes to view any meaningful data.



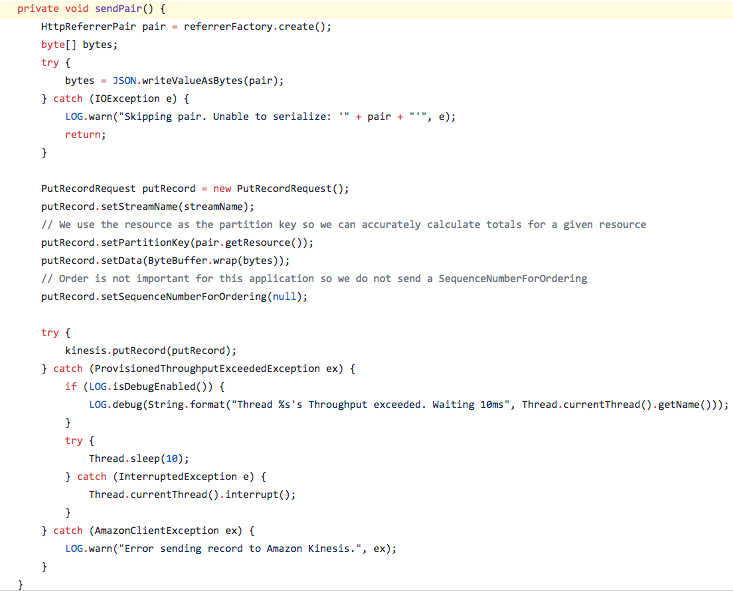
To view the stream details

1. Go to Kinesis console.
2. Select the stream whose name has the following form: KinesisDataVisSampleApp-KinesisStream-[randomString].
3. Choose the name of the stream to view the stream details.
4. The graphs display the activity of the data producer putting records into the stream and the data consumer getting data from the stream.



### Data Producer

Let’s visit [Rows 99-136](https://github.com/awslabs/amazon-kinesis-data-visualization-sample/blob/master/src/main/java/com/amazonaws/services/kinesis/samples/datavis/producer/HttpReferrerKinesisPutter.java" \l "L99-132" \t "_blank) of the HttpReferrerKinesisPutter



And [rows 85-92](https://github.com/awslabs/amazon-kinesis-data-visualization-sample/blob/d7fbcb994caad606635414b01c4eeada15c04f0b/src/main/java/com/amazonaws/services/kinesis/samples/datavis/HttpReferrerStreamWriter.java" \l "L85-92" \t "_blank) of the HttpReferrerStreamWriter



### Data Consumer: KCL

[Rows 92-98](https://github.com/awslabs/amazon-kinesis-data-visualization-sample/blob/master/src/main/java/com/amazonaws/services/kinesis/samples/datavis/HttpReferrerCounterApplication.java" \l "L92-98" \t "_blank) of the HttpReferrerCounterApplication

[Rows 104-108](https://github.com/awslabs/amazon-kinesis-data-visualization-sample/blob/master/src/main/java/com/amazonaws/services/kinesis/samples/datavis/HttpReferrerCounterApplication.java" \l "L104-108" \t "_blank) of the HttpReferrerCounterApplication

[Rows 186-203](https://github.com/awslabs/amazon-kinesis-data-visualization-sample/blob/d7fbcb994caad606635414b01c4eeada15c04f0b/src/main/java/com/amazonaws/services/kinesis/samples/datavis/kcl/CountingRecordProcessor.java" \l "L186-203" \t "_blank) of the CountingRecordProcessor

1. View the amazon DynamoDB tables starting with KinesisDataVisSampleApp\*

## Task 3: Create Kinesis Firehose Delivery Stream

1. Go to Kinesis firehose console
2. Create Delivery Stream
   1. Name: ArchivalStream
3. Click **Next**
4. On Transform records, Click Next.
5. On Select Destination,
   1. Destination: Amazon S3
   2. S3 Bucket: Create New
      1. Bucket-Name: myfirehose-[your initials]
      2. Region: Same as kinesis stack
6. Click **Next**
7. On configure settings,
   1. IAM Role: Create New
   2. Click **Allow** in the role creation tab.

This will create a role (firehose\_delivery\_role) with required policies.

1. Click Next
2. On Review, Click Create Delivery Stream

## Task 4: Install and configure Kinesis Agent

1. Figure out Public IP of the instance.
2. SSH into the instance
3. Install kinesis agent

sudo yum install –y aws-kinesis-agent

1. Modify the policy attached to the role of the ec2 instance running the agent.
2. Add the following to the JSON statement section.

{

"Action": "firehose:\*",

"Resource": "\*",

"Effect": "Allow"

}

1. Modify the kinesis agent config file to tail logs from /home/ec2-user/kinesis-data-vis-sample-app-kcl.log

sudo vi /etc/aws-kinesis/agent.json

{

"cloudwatch.emitMetrics": true,

"kinesis.endpoint": "kinesis.us-east-1.amazonaws.com",

"firehose.endpoint": "firehose.us-east-1.amazonaws.com",

"flows": [

{

"filePattern": "/home/ec2-user/kinesis-data-vis-sample-app-kcl.log",

"deliveryStream": "ArchivalStream"

}

]

}

1. Update the permissions to the logs directory so that kinesis agent user can get access.

sudo chmod -R ugo+rwx /home/ec2-user/

1. Configure the agent to start on system startup.

sudo chkconfig aws-kinesis-agent on

1. Start the agent manually

sudo service aws-kinesis-agent start

1. Verify if the agent has started successfully and sending logs to firehose stream

tail -f /var/log/aws-kinesis-agent/aws-kinesis-agent.log

Observer Data shall now be available in S3 bucket.

Congratulations!!!

## Task 5: Delete Sample Application

1. Go to CloudFormation console, choose stack and delete stack.