How to connect to NYSE Kafka cluster in PROD Env

Prerequisites:

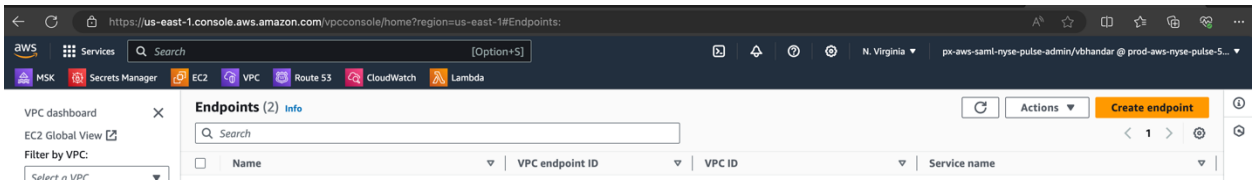
- A VPC with 3 subnets, based on the AWS region where consumer is setting up the client,

Region: US East (us-east-1)	**AZ IDs**: use1-az2, use1-az4 and use1-az6

- Access to create VPC endpoints in us-east-1 region or ap-east-1.
- Customer has provided their info to NYSE and entered their role ARN and have Pulse Kafka username/password generated by NYSE RAAMS system handy.

Steps to create VPC Endpoints:

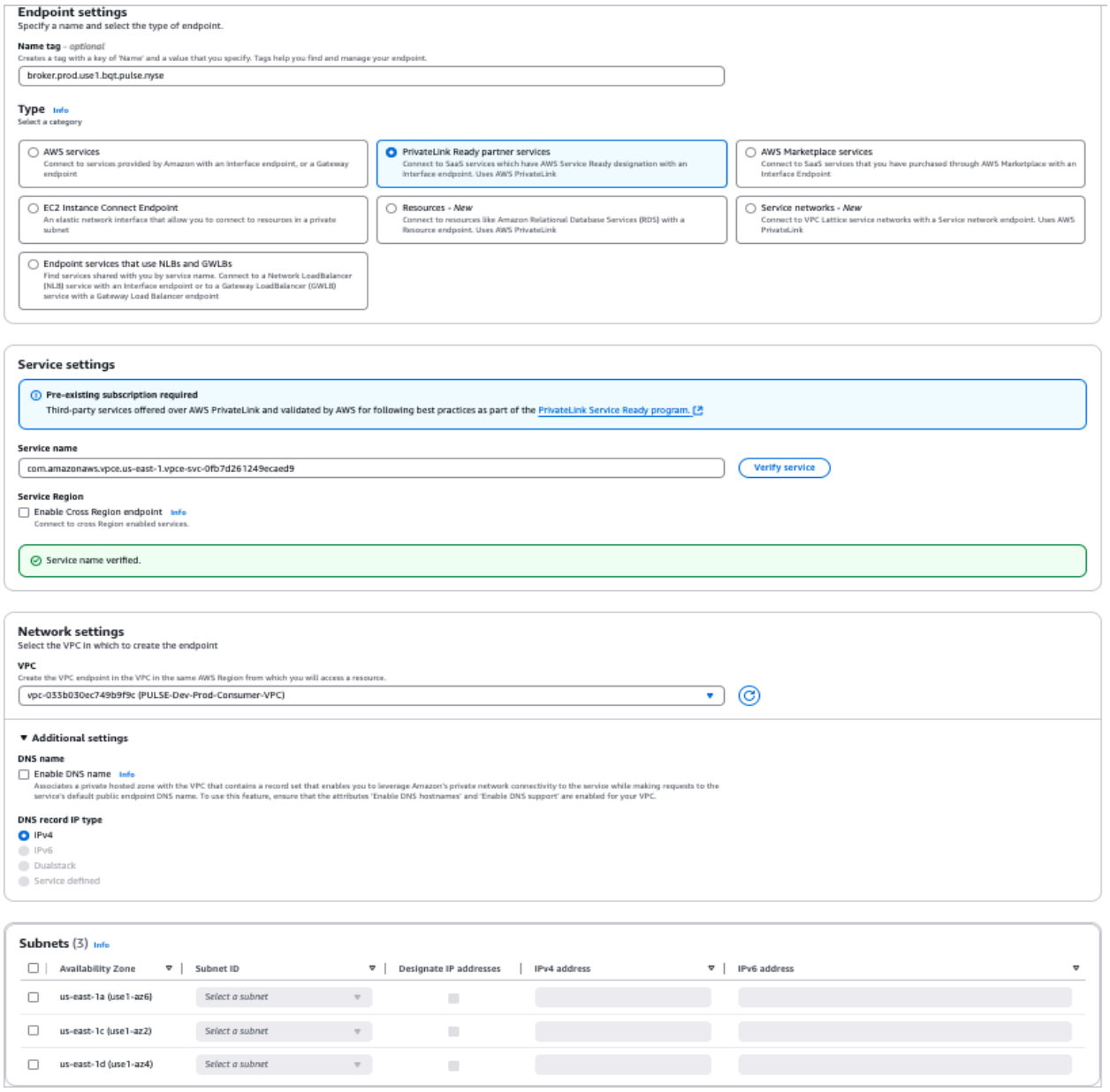
1. Once logged in to your AWS account (Use the same role to login to AWS account which has been provided to NYSE), navigate to Endpoints | VPC Console (amazon.com) and click on Create Endpoint button



2. On Next page Click on “PrivateLink Ready partner Services” radio button and enter one of the endpoint service names depending on your AWS region, as shown below. Please note your region and Availability Zone IDs will vary depending on the region.

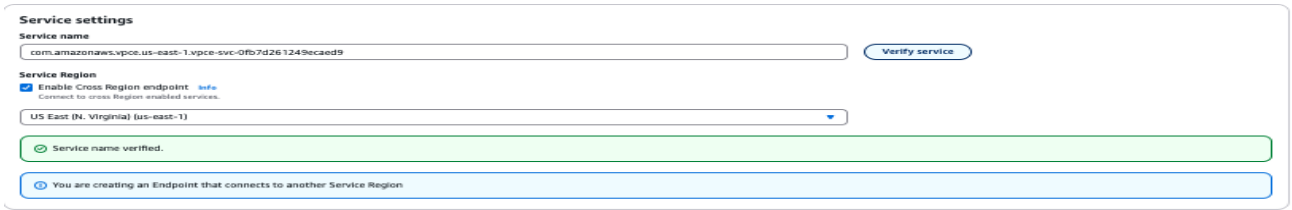
Region	Endpoint Service Name	Availability Zones	Name Tag	Port
US East 1 (us-east-1)	com.amazonaws.vpce.us-east-1.vpce-svc-0fb7d261249ecaed9	use1-az2, use1-az4 and use1-az6	broker.prod.use1.bqt.pulse.nyse	9094

Once the appropriate Service Name is entered, click on “Verify service”.



Note:

When you create endpoint from ap-east-1 (Hong Kong) region, please choose "Enable Service Region endpoint" and select "us-east-1" as a region, as shown below.



3. Once Service Name is verified, pick your VPC and subnets and click on Create endpoint. This endpoint DNS name will be referred to as broker.prod.use1.bqt.pulse.nyse in step#6 below.

4. Repeat steps 2 and 3 for three more times using following service names based on the region where you are creating the endpoints. Your AWS region and Availability Zone IDs will vary depending on the region:

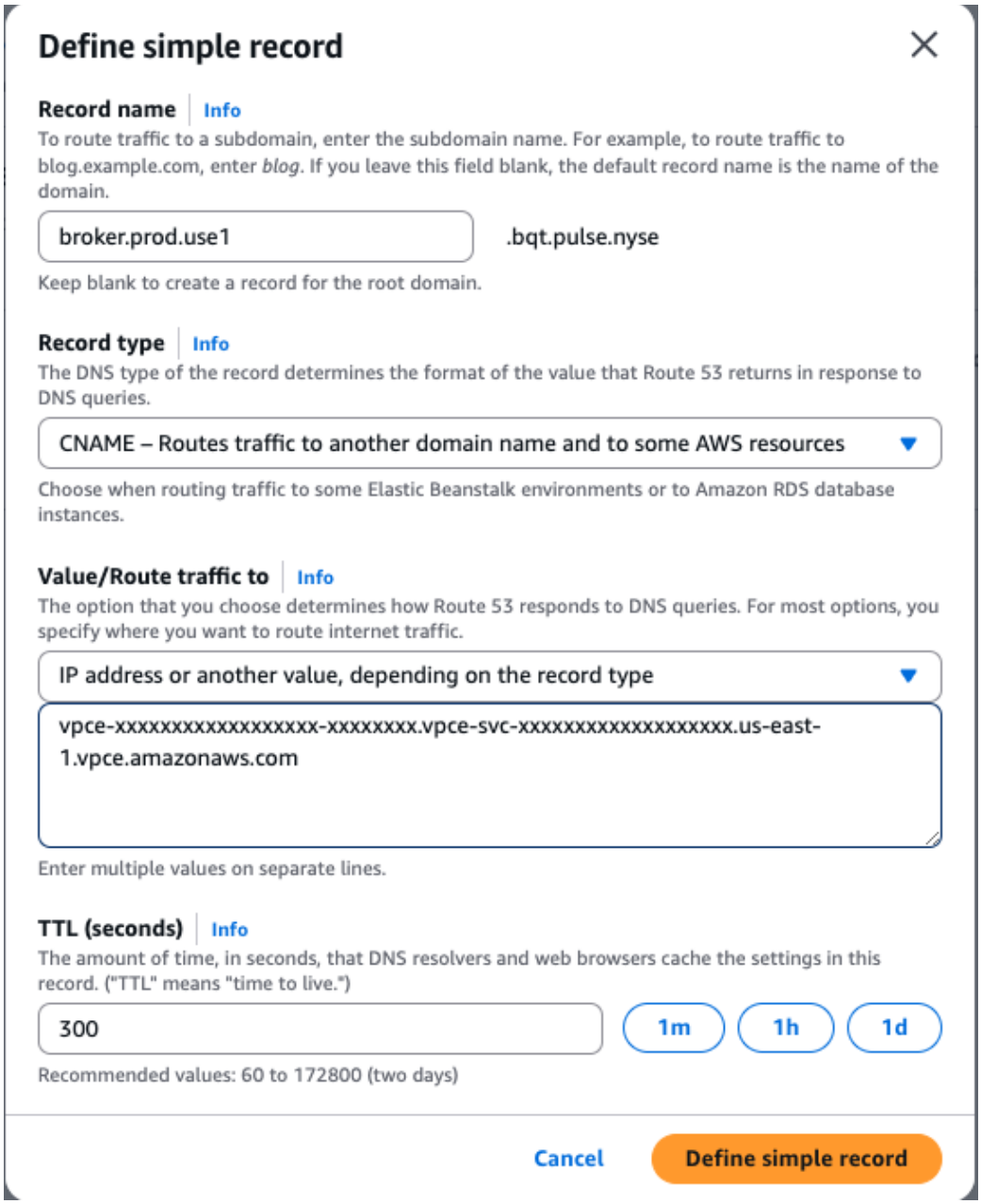
Region	**Endpoint Service Name**	**Availability Zones**	**Name Tag**	**Port**
US East (us-east-1)	com.amazonaws.vpce.us-east-1.vpce-svc-065db063a68f6c80f	use1-az2	broker1.prod.use1.bqt.pulse.nyse	9094
US East (us-east-1)	com.amazonaws.vpce.us-east-1.vpce-svc-00f6fe24081f3c61f	use1-az4	broker2.prod.use1.bqt.pulse.nyse	9094
US East (us-east-1)	com.amazonaws.vpce.us-east-1.vpce-svc-0b58442c9f5ee8ba4	use1-az6	broker3.prod.use1.bqt.pulse.nyse	9094

Please note VPC endpoint DNS names from 4 will be referred to in step number 6 below.

5. Navigate to Route 53 and create a private hosted zone named **bqt.pulse.nyse** and pick the appropriate VPC in which the endpoints were created.

6. Once hosted zone is created click on it and create 4 different records of type CNAME namely using the VPC endpoint DNS names from step # 4:

broker.prod.use1.bqt.pulse.nyse
broker1.prod.use1.bqt.pulse.nyse
broker2.prod.use1.bqt.pulse.nyse
broker3.prod.use1.bqt.pulse.nyse



7. Send an email to Dev-NYSE-Cloud@ice.com and try a library of your choice to test kafka connectivity using broker.prod.use1.bqt.pulse.nyse:9094 as the hostname and credentials shared with you.

8. Some sample libraries that you can use are:
    a. https://github.com/confluentinc/librdkafka : C++
    b. https://kafka.js.org/ : nodejs
    c. https://docs.confluent.io/kafka-clients/python/current/overview.html : python

9. To parse the messages from the stream, download the proto file and compile it using protoc command for the tech stack your system uses:

C/C++	Python	Java	Javascript
protoc --proto_path=src --cpp_out=$DEST/gen/foo.proto $SRC/bqt_cloud_streaming.proto	protoc -I=$SRC_DIR --python_out=$DST_DIR $SRC_DIR/bqt_cloud_streaming.proto	protoc -I=$SRC_DIR --java_out=$DST_DIR $SRC_DIR/bqt_cloud_streaming.proto	protoc --js_out=gen bqt_cloud_streaming.proto

10. Once you include the generated binary in your application, you can use the auto generated methods from above to deserialize the kafka message. Tutorials are in https://protobuf.dev/getting-started/
    -
11. Detailed client specifications for kafka are available here.