RBAC FOR KAFKA WITH DOCKER-COMPOSE

Configuring steps :

1) Create Token keypairs
2) Extracting the public key from token pair
3) Configure Oauth bearer authentication
4) Configure for MDS
5) Configuring Login.properties
6) Configure client
7) Login into MDS using confluent command
8) Using conlfuent iam command assign the roles to user on cluster resources
9) Request to perform action on resources of confluent

---

1. Generating the Tokenkey pairs in local using the openssl command

```
mkdir <path-to-tokenKeypair.pem> && openssl genrsa -out
<path-to-tokenKeypair.pem> 2048
```

---

2. Extract public key from tokenkeypair

```
openssl rsa -in <path-to-tokenKeypair.pem> -outform PEM -pubout -out <path-to-public.pem>
```

---

3. Configuring Oauth bearer authentication in each broker

```
kafka-1:
  image: 'confluentinc/cp-kafka:latest'
  container_name: kafka-1
  depends_on:
    - zookeeper
  ports:
    - '9094:9094'
  environment:
    KAFKA_BROKER_ID: 1
    KAFKA_ZOOKEEPER_CONNECT: zookeeper:2181
    KAFKA_ADVERTISED_LISTENERS: PLAINTEXT://localhost:9094,SASL_PLAINTEXT://kafka-1:9093,OAUTH://kafka-1:9098
    KAFKA_INTER_BROKER_LISTENER_NAME: SASL_PLAINTEXT
    KAFKA_LISTENER_SECURITY_PROTOCOL_MAP: PLAINTEXT:PLAINTEXT,SASL_PLAINTEXT:SASL_PLAINTEXT,OAUTH:SASL_PLAINTEXT
    KAFKA_OFFSETS_TOPIC_REPLICATION_FACTOR: 2
    ZOOKEEPER_SASL_ENABLED: 'false'
    #KAFKA_SECURITY_INTER_BROKER_PROTOCOL: SASL_PLAINTEXT
    KAFKA_SASL_MECHANISM_INTER_BROKER_PROTOCOL: PLAIN,OAUTHBEARER
    KAFKA_SASL_ENABLED_MECHANISMS: PLAIN,OAUTHBEARER #SCRAM-SHA-256
    KAFKA_OPTS: "-Djava.security.auth.login.config=/etc/kafka/configs/kafka_server_jaas.conf -Dzookeeper.sasl.client=false -Dkafka.plugin.path=/usr/share/java/kafka"
```

```
kafka-2:
  image: 'confluentinc/cp-kafka:latest'
  container_name: kafka-2
  depends_on:
   - zookeeper
  ports:
   - '9095:9095'
  environment:
   KAFKA_BROKER_ID: 2
   KAFKA_ZOOKEEPER_CONNECT: zookeeper:2181
   KAFKA_ADVERTISED_LISTENERS:
PLAINTEXT://localhost:9095,SASL_PLAINTEXT://kafka-1:9093,OAUTH://kafka-1:90
98
   KAFKA_INTER_BROKER_LISTENER_NAME: SASL_PLAINTEXT
   KAFKA_LISTENER_SECURITY_PROTOCOL_MAP:
PLAINTEXT:PLAINTEXT,SASL_PLAINTEXT:SASL_PLAINTEXT,OAUTH:SASL_PLAI
NTEXT
   KAFKA_OFFSETS_TOPIC_REPLICATION_FACTOR: 2
   ZOOKEEPER_SASL_ENABLED: 'false'
   #KAFKA_SECURITY_INTER_BROKER_PROTOCOL: SASL_PLAINTEXT
   KAFKA_SASL_MECHANISM_INTER_BROKER_PROTOCOL:
PLAIN,OAUTHBEARER
   KAFKA_SASL_ENABLED_MECHANISMS: PLAIN,OAUTHBEARER
#SCRAM-SHA-256
   KAFKA_OPTS:
"-Djava.security.auth.login.config=/etc/kafka/configs/kafka_server_jaas.conf
-Dzookeeper.sasl.client=false -Dkafka.plugin.path=/usr/share/java/kafka"

kafka-3:
  image: 'confluentinc/cp-kafka:latest'
  container_name: kafka-3
  depends_on:
   - zookeeper
  ports:
   - '9096:9096'
  environment:
   KAFKA_BROKER_ID: 3
   KAFKA_ZOOKEEPER_CONNECT: zookeeper:2181
   KAFKA_ADVERTISED_LISTENERS:
PLAINTEXT://localhost:9096,SASL_PLAINTEXT://kafka-1:9093,OAUTH://kafka-1:90
98
   KAFKA_INTER_BROKER_LISTENER_NAME: SASL_PLAINTEXT
   KAFKA_LISTENER_SECURITY_PROTOCOL_MAP:
PLAINTEXT:PLAINTEXT,SASL_PLAINTEXT:SASL_PLAINTEXT,OAUTH:SASL_PLAI
NTEXT
   KAFKA_OFFSETS_TOPIC_REPLICATION_FACTOR: 2
   ZOOKEEPER_SASL_ENABLED: 'false'
   #KAFKA_SECURITY_INTER_BROKER_PROTOCOL: SASL_PLAINTEXT
```

```
    KAFKA_SASL_MECHANISM_INTER_BROKER_PROTOCOL:
PLAIN,OAUTHBEARER
    KAFKA_SASL_ENABLED_MECHANISMS: PLAIN,OAUTHBEARER
#SCRAM-SHA-256
    KAFKA_OPTS:
"-Djava.security.auth.login.config=/etc/kafka/configs/kafka_server_jaas.conf
-Dzookeeper.sasl.client=false -Dkafka.plugin.path=/usr/share/java/kafka"
```

4. Configuring MDS on each broker

```
    KAFKA_AUTHORIZER_CLASS_NAME:
io.confluent.kafka.security.authorizer.ConfluentServerAuthorizer
    KAFKA_CONFLUENT_AUTHORIZER_ACCESS_RULE_PROVIDERS:
CONFLUENT
    KAFKA_SUPER_USERS: User:admin
    KAFKA_CONFLUENT_METRICS_REPORTER_BOOTSTRAP_SERVERS:
localhost:9094
    KAFKA_CONFLUENT_METADATA_SERVER_LISTENERS: http://localhost:8090
    KAFKA_CONFLUENT_METADATA_SERVER_ADVERTISED_LISTENERS:
http://localhost:8090
    KAFKA_CONFLUENT_METADATA_SERVER_AUTHENTICATION_METHOD:
BEARER
    KAFKA_CONFLUENT_METADATA_SERVER_USER_STORE: FILE
    KAFKA_CONFLUENT_METADATA_SERVER_USER_STORE_FILE_PATH:
/etc/kafka/tokens/login.properties
    KAFKA_CONFLUENT_METADATA_SERVER_TOKEN_KEY_PATH:
/etc/kafka/tokens/tokenKeyPair.pem
    KAFKA_LISTENER_NAME_OAUTH_SASL_ENABLED_MECHANISMS:
OAUTHBEARER

KAFKA_LISTENER_NAME_OAUTH_OAUTHBEARER_SASL_LOGIN_CALLBACK_
HANDLER_CLASS:
io.confluent.kafka.server.plugins.auth.token.TokenBearerServerLoginCallbackHandle
r

KAFKA_LISTENER_NAME_OAUTH_OAUTHBEARER_SASL_SERVER_CALLBAC
K_HANDLER_CLASS:
io.confluent.kafka.server.plugins.auth.token.TokenBearerValidatorCallbackHandler
    KAFKA_LISTENER_NAME_OAUTH_OAUTHBEARER_SASL_JASS_CONFIG: \
        org.apache.kafka.common.security.oauthbearer.OAuthBearerLoginModule
required \
       publicKeyPath="/etc/kafka/tokens/Public.pem";
    KAFKA_CONFLUENT_LICENSE_TOPIC_REPLICATION_FACTOR: 2
    KAFKA_CONFLUENT_METADATA_TOPIC_REPLICATION_FACTOR: 2
```

These configurations are same for all brokers

---

5. Volumes to mounting files :

```
 - /home/charan/tokenkeypair:/etc/kafka/tokens
```

In this volume mounting the token directory in local contains the

- Tokenkeypair.pem (private key)
- Publickey.pem   (public key)
- login.properties

Login.properties : simple username and passwords

Admin : admin-secret
Kafkauser : allkeys
Arun : allkeys

---

6. Configuring the client (kafkauser, Arun) :

❖ kafkauser.properties

```
security.protocol=SASL_PLAINTEXT
sasl.mechanism=OAUTHBEARER
sasl.login.callback.handler.class=io.confluent.kafka.clients.plugins.auth.token.TokenUserLoginCallbackHandler
sasl.jaas.config=org.apache.kafka.common.security.oauthbearer.OAuthBearerLoginModule required \
   metadataServerUrls="http://kafkaserver-1:8090" \
   username="kafkauser" password="allkeys" ;
```

❖ Arun.properties :

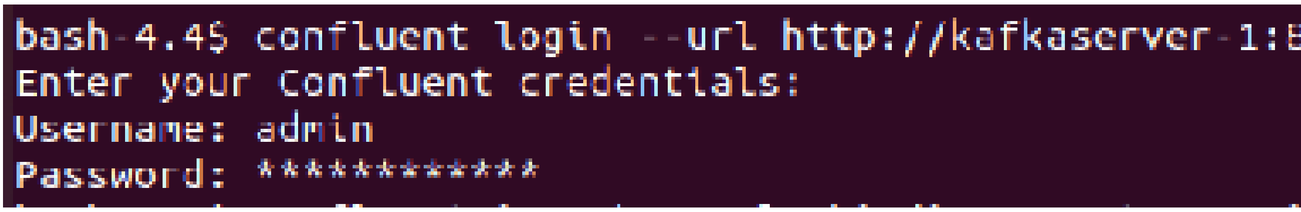
```
security.protocol=SASL_PLAINTEXT
sasl.mechanism=OAUTHBEARER
sasl.login.callback.handler.class=io.confluent.kafka.clients.plugins.auth.token.TokenUserLoginCallbackHandler
sasl.jaas.config=org.apache.kafka.common.security.oauthbearer.OAuthBearerLoginModule required \
   metadataServerUrls="http://kafkaserver-1:8090" \
   username="Arun" password="allkeys" ;
```

❖ Admin.properties

```
security.protocol=SASL_PLAINTEXT
sasl.mechanism=OAUTHBEARER
sasl.login.callback.handler.class=io.confluent.kafka.clients.plugins.auth.token.TokenUserLoginCallbackHandler
sasl.jaas.config=org.apache.kafka.common.security.oauthbearer.OAuthBearerLoginModule required \
   metadataServerUrls="http://kafkaserver-1:8090" \
   username="Admin" password="amdin-secret" ;
```

7. Login into the confluent metadata using its url as admin :

confluent login --url http://localhost:8090



```
bash-4.4$ confluent login --url http://kafkaserver-1:
Enter your Confluent credentials:
Username: admin
Password: ************
```

In above image we login the mds url with the user admin and password

By default mds listens from port 8090

---

8. Assign the roles to users :

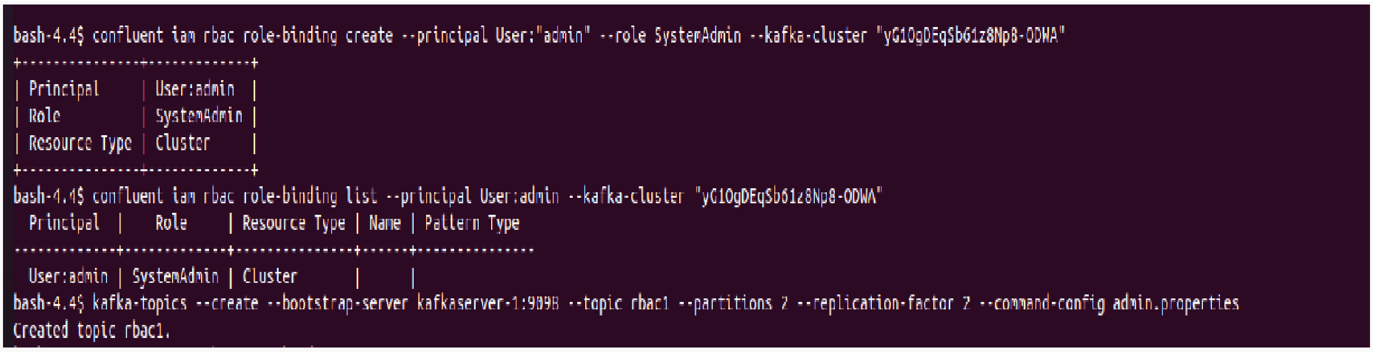
Consider two users

- Admin
- Kafkauser
- Arun

01. Asigning SystemAdmin role to the admin user :

```
confluent iam rbac role-binding create
--principal
User:"${CONFLUENT_PLATFORM_USERNAME}" --role
SystemAdmin --kafka-cluster-id
"${CLUSTER_ID}"
```

```
confluent iam rbac role-binding create
--principal User:"Admin" --role SystemAdmin
--kafka-cluster-id "NlIoEKI3TfGroRtJ85c4_Q"
```



```
bash-4.4$ confluent iam rbac role-binding create --principal User:"admin" --role SystemAdmin --kafka-cluster "yG1OgDEqSb61z8Np8-ODWA"
+--------------+------------+
| Principal     | User:admin  |
| Role          | SystemAdmin |
| Resource Type | Cluster     |
+--------------+------------+
bash-4.4$ confluent iam rbac role-binding list --principal User:admin --kafka-cluster "yG1OgDEqSb61z8Np8-ODWA"
  Principal  |    Role     | Resource Type | Name | Pattern Type
-------------+-------------+---------------+------+---------------
  User:admin | SystemAdmin | Cluster       |      |
bash-4.4$ kafka-topics --create --bootstrap-server kafkaserver-1:9098 --topic rbac1 --partitions 2 --replication-factor 2 --command-config admin.properties
Created topic rbac1.
```

<u>To list the roles for user admin</u>

```
Confluent iam rbac role-binding
list --principal User:admin
--kafka-cluster
"NlIoEKI3TfGroRtJ85c4_Q"
```

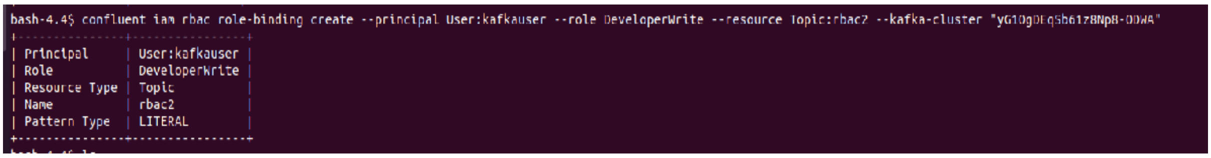
Create topic for the rabc check :

```
Kafka-topics --create_--broker-list
kafka-1 –topic rabc1 –replication-factor
3 –partitions 3
```

02. Asigning the DevoloperWrite to kafkauser user :

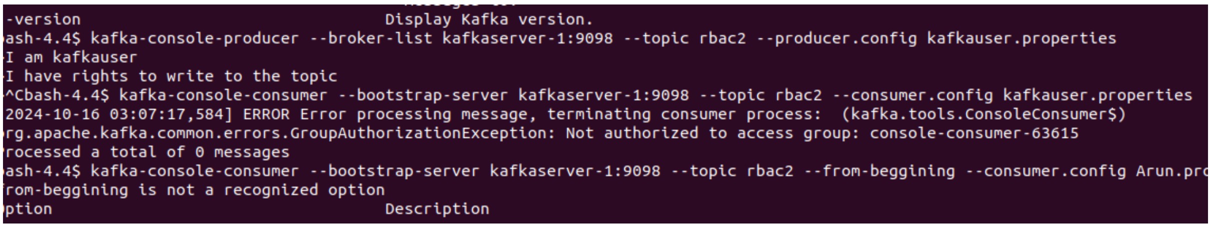
```
confluent iam rbac role-binding create --principal
User:<client_user> --role DeveloperRead --resource
Group:<consumer-group> --prefix --kafka-cluster <
kafka-cluster-id >
```

```
confluent iam rbac role-binding create --principal
User:<kafkauser> --role DeveloperWrite --resource
--kafka-cluster <NlIoEKI3TfGroRtJ85c4_Q >
```



```
bash-4.4$ confluent iam rbac role-binding create --principal User:kafkauser --role DeveloperWrite --resource Topic:rbac2 --kafka-cluster "yG1OgDEqSb61z8Np8-ODWA"
+---------------+---------------+
| Principal     | User:kafkauser |
| Role          | DeveloperWrite |
| Resource Type | Topic          |
| Name          | rbac2          |
| Pattern Type  | LITERAL        |
+---------------+---------------+
```

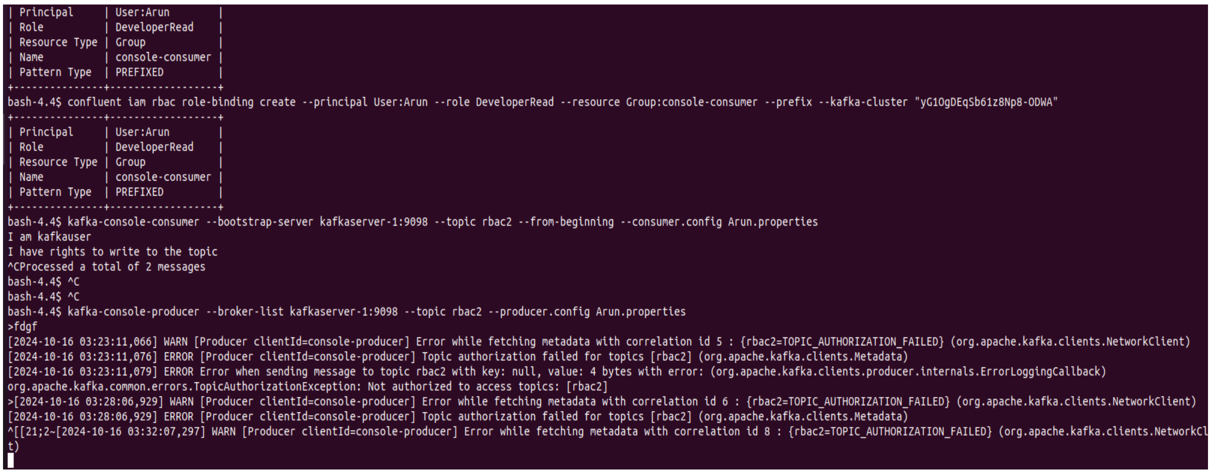
And produce some messages and try to consume messages



```
-version                                Display Kafka version.
ash-4.4$ kafka-console-producer --broker-list kafkaserver-1:9098 --topic rbac2 --producer.config kafkauser.properties
I am kafkauser
I have rights to write to the topic
^Cbash-4.4$ kafka-console-consumer --bootstrap-server kafkaserver-1:9098 --topic rbac2 --consumer.config kafkauser.properties
2024-10-16 03:07:17,584] ERROR Error processing message, terminating consumer process:  (kafka.tools.ConsoleConsumer$)
rg.apache.kafka.common.errors.GroupAuthorizationException: Not authorized to access group: console-consumer-63615
rocessed a total of 0 messages
ash-4.4$ kafka-console-consumer --bootstrap-server kafkaserver-1:9098 --topic rbac2 --from-beggining --consumer.config Arun.pro
rom-beggining is not a recognized option
ption                                   Description
```

As you have seen image kafkauser produces some messages to the topic rbac
But when try to consume messages it fails bcz kafkauser has only permission to write not read permission .

03. Asigning the DevoloperRead to Arun user :



```
| Principal     | User:Arun        |
| Role          | DeveloperRead    |
| Resource Type | Group            |
| Name          | console-consumer |
| Pattern Type  | PREFIXED         |
+---------------+------------------+
bash-4.4$ confluent iam rbac role-binding create --principal User:Arun --role DeveloperRead --resource Group:console-consumer --prefix --kafka-cluster "yG1OgDEqSb61z8Np8-ODWA"
+---------------+------------------+
| Principal     | User:Arun        |
| Role          | DeveloperRead    |
| Resource Type | Group            |
| Name          | console-consumer |
| Pattern Type  | PREFIXED         |
+---------------+------------------+
bash-4.4$ kafka-console-consumer --bootstrap-server kafkaserver-1:9098 --topic rbac2 --from-beginning --consumer.config Arun.properties
I am kafkauser
I have rights to write to the topic
^CProcessed a total of 2 messages
bash-4.4$ ^C
bash-4.4$ ^C
bash-4.4$ kafka-console-producer --broker-list kafkaserver-1:9098 --topic rbac2 --producer.config Arun.properties
>fdgf
[2024-10-16 03:23:11,066] WARN [Producer clientId=console-producer] Error while fetching metadata with correlation id 5 : {rbac2=TOPIC_AUTHORIZATION_FAILED} (org.apache.kafka.clients.NetworkClient)
[2024-10-16 03:23:11,076] ERROR [Producer clientId=console-producer] Topic authorization failed for topics [rbac2] (org.apache.kafka.clients.Metadata)
[2024-10-16 03:23:11,079] ERROR Error when sending message to topic rbac2 with key: null, value: 4 bytes with error: (org.apache.kafka.clients.producer.internals.ErrorLoggingCallback)
org.apache.kafka.common.errors.TopicAuthorizationException: Not authorized to access topics: [rbac2]
>[2024-10-16 03:28:06,929] WARN [Producer clientId=console-producer] Error while fetching metadata with correlation id 6 : {rbac2=TOPIC_AUTHORIZATION_FAILED} (org.apache.kafka.clients.NetworkClient)
[2024-10-16 03:28:06,929] ERROR [Producer clientId=console-producer] Topic authorization failed for topics [rbac2] (org.apache.kafka.clients.Metadata)
^[[21;2~[2024-10-16 03:32:07,297] WARN [Producer clientId=console-producer] Error while fetching metadata with correlation id 8 : {rbac2=TOPIC_AUTHORIZATION_FAILED} (org.apache.kafka.clients.NetworkCl
t)
```

Here we have assigned the role of DevoloperRead role to Arun user

confluent iam rbac role-binding create --principal User:<client_user> --role DeveloperRead --resource Group:<consumer-group> --prefix --kafka-cluster < kafka-cluster-id >

confluent iam rbac role-binding create --principal User:<Arun> --role DeveloperRead --resource Group:<console-consumer> --prefix --kafka-cluster <`NlIoEKI3TfGroRtJ85c4_Q` >

Try Produce some messages to topic rbac1
As see in the image when producing or writing messages to topic rbac2 It will fails bcz Arun user have only write permission Not read permission

Try to read message it will works But when he tried to produce messages it not works .