Scenario 8

Problem statement

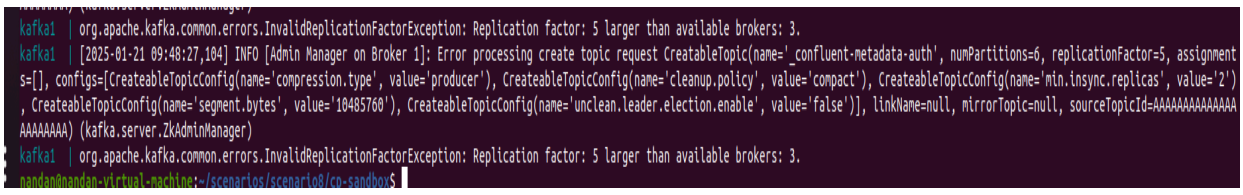
The kafka brokers are down after an upgrade to a newer version. Several properties and files were changed during the upgrade and the customer does not have an audit of the changes. Review the logs and come up with a solution to start the kafka brokers.

org.apache.kafka.common.config.ConfigException: Invalid value javax.net.ssl.SSLHandshakeException: Empty client certificate chain for configuration A client SSLEngine created with the provided settings can't connect to a server SSLEngine created with those settings.

at org.apache.kafka.common.security.ssl.SslFactory.configure(SslFactory.java:103)
at org.apache.kafka.common.network.SslChannelBuilder.configure(SslChannelBuilder.java:84)
at org.apache.kafka.common.network.ChannelBuilders.create(ChannelBuilders.java:265)
at org.apache.kafka.common.network.ChannelBuilders.serverChannelBuilder(ChannelBuilders.java:166)
at kafka.network.Processor.<init>(SocketServer.scala:1177)
at kafka.network.Acceptor.newProcessor(SocketServer.scala:1050)
at kafka.network.Acceptor.$anonfun$addProcessors$1(SocketServer.scala:1010)
at scala.collection.immutable.Range.foreachmVcsp(Range.scala:190)
at kafka.network.Acceptor.addProcessors(SocketServer.scala:1009)
at kafka.network.DataPlaneAcceptor.configure(SocketServer.scala:670)
at kafka.network.SocketServer.createDataPlaneAcceptorAndProcessors(SocketServer.scala:278)
at kafka.network.SocketServer.$anonfun$new$51(SocketServer.scala:224)
at kafka.network.SocketServer.$anonfun$new51adapted(SocketServer.scala:224)
at scala.collection.IterableOnceOps.foreach(IterableOnce.scala:575)
at scala.collection.IterableOnceOps.foreach$(IterableOnce.scala:573)
at scala.collection.AbstractIterable.foreach(Iterable.scala:933)
at kafka.network.SocketServer.<init>(SocketServer.scala:224)
at kafka.server.KafkaServer.startup(KafkaServer.scala:528)
at kafka.Kafka$.main(Kafka.scala:114)

Root cause

When tried to list the topics, it returns nothing and the following error logs can be seen from the broker.



Observation

It is observed that we have set the wrong configuration of Replication factor which is set to 5 and is larger than available brokers: 3

In the broker's properties files the confluent.metadata.topic.replication.factor is set to 5, whereas it should be ideally between 1 to 3 in our case as we have at most 3 brokers available for replication, 1 being less fault tolerant and 3 being most fault tolerant.

Solution to debug the issue

1. In all the broker's properties file set the confluent.metadata.topic.replication.factor to 3 (for high fault tolerance)

```
############################## MDS Server Settings ##############################
# Bind Metadata Service HTTP service to port 8090.
confluent.metadata.server.listeners=http://0.0.0.0:8090
confluent.metadata.server.advertised.listeners=http://kafka3:8090
# The key to encrypt the token (when you issue you a token)
confluent.metadata.server.token.key.path=/etc/kafka/mdsKey.pem
# Supported authentication methods
confluent.metadata.server.authentication.method=BEARER

confluent.metadata.topic.replication.factor=5
# confluent.security.event.logger.exporter.kafka.topic.replicas=1
```

Observation

Successfully set the value of confluent.metadata.topic.replication.factor to 3 on all three properties files of all the brokers.

Conclusion

Since the confluent.metadata.topic.replication.factor value was set to 5 which is the wrong configuration, its value can be between 1 to 3 as we have 3 available brokers at the most. After setting the confluent.metadata.topic.replication.factor value to 3 (high fault tolerance) we can successfully see that all the brokers are functioning.

→ kafka-topics --list --bootstrap-server kafka1:19092 --command-config /opt/client/client.properties

```
nandan@nandan-virtual-machine:~/scenarios/scenario8/cp-sandbox$ docker compose exec -it -u root kfkclient bash
WARN[0000] The "ADMIN_USER" variable is not set. Defaulting to a blank string.
WARN[0000] The "ADMIN_PASSWORD" variable is not set. Defaulting to a blank string.
WARN[0000] /home/nandan/scenarios/scenario8/cp-sandbox/docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[root@kfkclient appuser]# kafka-topics --list --bootstrap-server kafka1:19092 --command-config /opt/client/client.properties
__consumer_offsets
_confluent-command
_confluent-metadata-auth
_confluent-metrics
_confluent-telemetry-metrics
confluent-audit-log-events
connect-configs
connect-offsets
connect-status
[root@kfkclient appuser]#
```