Item 58

git_comments:

1. * * This exception indicates that SASL authentication has failed. The error message * in the exception indicates the actual cause of failure. * <p> * SASL authentication failures typically indicate invalid credentials, but * could also include other failures specific to the SASL mechanism used * for authentication. * </p>
2. * * If an authentication exception is encountered with connection to any broker, * fail all pending requests.
3. This is already logged as error, but propagated here to perform any clean ups.
4. * * This exception indicates that SASL authentication has failed. * On authentication failure, clients abort the operation requested and raise one * of the subclasses of this exception: * <ul> * </li>{@link SaslAuthenticationException} if SASL handshake fails with invalid credentials * or any other failure specific to the SASL mechanism used for authentication</li> * <li>{@link UnsupportedSaslMechanismException} if the SASL mechanism requested by the client * is not supported on the broker.</li> * <li>{@link IllegalSaslStateException} if an unexpected request is received on during SASL * handshake. This could be due to misconfigured security protocol.</li> * </ul>
5. * * This exception indicates unexpected requests prior to SASL authentication. * This could be due to misconfigured security, e.g. if PLAINTEXT protocol * is used to connect to a SASL endpoint.
6. * * This exception indicates that SASL authentication has failed. The error message * in the exception indicates the actual cause of failure. * <p> * SASL authentication failures typically indicate invalid credentials, but * could also include other failures specific to the SASL mechanism used * for authentication. * </p>
7. * * This exception indicates that the SASL mechanism requested by the client * is not enabled on the broker.
8. Sasl authentication error which may be one of NONE, UNSUPPORTED_SASL_MECHANISM, ILLEGAL_SASL_STATE, SASL_AUTHENTICATION_FAILED or NETWORK_EXCEPTION
9. expected exception

git_commits:

1. **summary:** KAFKA-5947; Handle authentication failure in admin client, txn producer
   message: KAFKA-5947; Handle authentication failure in admin client, txn producer 1. Raise AuthenticationException for authentication failures in admin client 2. Handle AuthenticationException as a fatal error for transactional producer 3. Add comments to authentication exceptions Author: Rajini Sivaram <rajinisivaram@googlemail.com> Reviewers: Vahid Hashemian <vahidhashemian@us.ibm.com>, Ismael Juma <ismael@juma.me.uk> Closes #3928 from rajinisivaram/KAFKA-5947-auth-failure

github_issues:

github_issues_comments:

github_pulls:

1. **title:** KAFKA-5947: Handle authentication failure in admin client, txn producer
   body: 1. Raise AuthenticationException for authentication failures in admin client 2. Handle AuthenticationException as a fatal error for transactional producer 3. Add comments to authentication exceptions

github_pulls_comments:

1. @vahidhashemian @hachikuji Can you review, please? Thanks.
2. @rajinisivaram Thanks for the PR. It looks good to me after a quick review. I like the alternate exception name @ijuma suggested.
3. @vahidhashemian @ijuma Thanks for the reviews. Have addressed the comments so far.

github_pulls_reviews:

1. I wonder if this should be called `InvalidCredentialsException`.
2. What is the reason for `hardShutdownTimeMs.get() != INVALID_SHUTDOWN_TIME`?
3. Would it be worth making this more generic and simply call it `fatalError` or something?
4. Nit: empty line.
5. Nit: empty line.
6. Nit: long line.
7. We can pass the serializers in the constructor and it's a bit more concise.
8. This is the default.
9. This is also the default, I think.
10. This and max in flight are defaults right?
11. At the moment, we have a single error code and exception for authentication failure with an error message giving more details. For SSL, this includes any handshake exception (e.g. server host name verification failure). At the moment, we are reporting only invalid credentials for SASL, but we may add others too. Since we convert SSLException and SaslException from the providers, it is simpler to maintain a more generic `AuthenticationFailed` exception rather than separate exceptions for different failures.
12. The issue I have is that `AuthenticationFailedException` is too similar to `AuthenticationException`.
13. What about `SaslAuthenticationFailedException` and `SslAuthenticationFailedException`?
14. That sounds good to me.
15. I will add without the `Failed` to be consistent with `TopicAuthorizationException` etc. So `SaslAuthenticationException` and `SslAuthenticationException`.
16. I read it a bit too quickly on my phone and I hadn't seen the "Failed" part. What you did in the end is what I had in mind.
17. We don't want to exit this background thread until a close has been requested. If we are closing and there is an authentication exception, we want to terminate immediately. If we are not closing, the thread is kept alive so that a future request after credentials are added can succeed. This is tested in the integration test.
18. Hmm... The code was changed from generic to more specific in the other PR based on review comments. So I thought this should be specific too.
19. Should we add a method that is like this one, but returns the exception instead? It seems a bit convoluted to throw a stored exception just to catch it. Not sure if that helps other cases, but at least here, it seems to make the code cleaner.
20. To add to the constructors, we need to remove the import restrictions on the package. It feels better to do them as properties instead since this test is currently in the security package. @vahidhashemian did that to avoid changing the restrictions.
21. copied and pasted earlier :-), removed now.
22. yes, removed.
23. I removed the SSL exception part from the doc. Will add `SslAuthenticationException` and the associated docs in #3918
24. OK, we can leave it like that then.
25. The serialization package is open to everyone (it's public API and at the lowest layer). So I don't think we should worry about that. It's not like we're avoiding a dependency here, we are just hiding it via a string based config (that still requires the default constructor to be present).
26. Nit: we can change the parameter to be `by name` (`=> Unit` instead of `() => Unit`) and then you don't need the noisy `() =>` in every invocation.
27. Do we want to include the exception message or stacktrace?
28. Why are we calling `handleFailure` directly instead of `fail`? If we used `fail` in both places, we could extract a method and call it twice in succession, once for `newCalls` and once for `callsToSend`.
29. Hmm, what's the impact of not exiting immediately? It seems like we would not have any responses anyway, so it should happen pretty quickly either way (and all pending requests have been cancelled). Maybe keep it simple?
30. I changed the existing method to return the exception rather than throw it.
31. @hachikuji, if you have a chance, have a look and see if this is OK.
32. Looks ok to me.

jira_issues:

1. **summary:** Handle authentication failures from transactional producer and KafkaAdminClient
   description: Follow on from KAFKA-5854 to handle authentication failures better for transactional producer API and KafkaAdminClient

jira_issues_comments:

1. GitHub user rajinisivaram opened a pull request: https://github.com/apache/kafka/pull/3928 KAFKA-5947: Handle authentication failure in admin client, txn producer 1. Raise AuthenticationException for authentication failures in admin client 2. Handle AuthenticationException as a fatal error for transactional producer 3. Add comments to authentication exceptions You can merge this pull request into a Git repository by running: $ git pull https://github.com/rajinisivaram/kafka KAFKA-5947-auth-failure Alternatively you can review and apply these changes as the patch at: https://github.com/apache/kafka/pull/3928.patch To close this pull request, make a commit to your master/trunk branch with (at least) the following in the commit message: This closes #3928 ---- commit 764f38785f83a95f840c398c9ef2f557e67fe07a Author: Rajini Sivaram <rajinisivaram@googlemail.com> Date: 2017-09-20T15:27:23Z KAFKA-5947: Handle authentication exceptions in admin client and txn producer ----
2. Github user asfgit closed the pull request at: https://github.com/apache/kafka/pull/3928