KIP-231: Improve the Required ACL of ListGroups API

Status

Current state: *Accepted*

Discussion thread: *here*

Vote thread: *here and here*

JIRA: **KAFKA-5638** - Getting issue details... STATUS

Released: 2.1.0

Please keep the discussion on the mailing list rather than commenting on the wiki (wiki discussions get unwieldy fast).

Motivation

Note: The discussion in this KIP applies to Java based (new) consumer only as the security feature is not supported by the old consumer.

From an authorization and ACL point of view, three operations (permission types) are currently defined for consumer groups: *Describe*, *Read*, *All* (a *Delete* operation is proposed by KIP-229). By default, *Read* implies *Describe*, and *All* implies all the other operations. Current consumer group related APIs and their minimum required permissions are listed in the following table:

API	Minimum Required Permission
`DescribeGroup`	*Describe* (Group)
`FindCoordinator`	*Describe* (Group)
`Heartbeat`	*Read* (Group)
`JoinGroup`	*Read* (Group)
`LeaveGroup`	*Read* (Group)
`ListGroups`	*Describe* (Cluster)
`OffsetCommit`	*Read* (Group)
`OffsetFetch`	*Describe* (Group)
`SyncGroup`	*Read* (Group)
`AddOffsetsToTxn`	*Read* (Group)
`TxnOffsetCommit`	*Read* (Group)

The anomaly is quite easy to spot in this table. All APIs require either *Read* or *Describe* operations on the *Group* resource type, except for `ListGroups`. The reason for originally choosing the *Describe (Cluster)* ACL over *Describe (Group)* is probably for admin type users who want to be able to take an inventory of all the consumer groups in the cluster (*Describe (Cluster)* could imply *Describe (Group)* for all groups in the cluster). Similar ACL setting exists for APIs such as `DescribeAcls` or `DescribeLogDirs`. However, there are some drawbacks to the current choice of ACL setting for `ListGroups` API:

1. As long as a user has a *Describe (Cluster)* ACL permission s/he can list all groups in the cluster. This exposes a security risk: either a user cannot list consumer groups in the cluster or s/he can list all of them. Listing all groups is reasonable for a cluster administrator, but for other users what they can list should be limited to what they need to list.
2. A user with *Read* access to a group can describe the group, but the same user would not see anything when listing groups (assuming there is no *Describe* access to the cluster). It makes more sense for this user to be able to list all groups s/he can already describe.

This would change the ACL requirements for `ListGroups` API in the table above (to return meaningful data) to Describe (Cluster) or Describe (Group). Not having any of these ACLs would still not cause any authorization error, but there will be no group in the response either.

API	Minimum Required Permission

`ListGroups`	*Describe* (Cluster) or *Describe* (Group)

Proposed Changes

The change proposed by this KIP is simple. An alternative ACL will be added as the minimum required permission of the *ListGroups* API: *Describe (Cluster)* would still work as before. However, a *Describe (Group)* ACL is added which gives users the ability to list groups they have this ACL on. The minimum required permissions are hard-coded in `kafka.server.KafkaApis.scala` inside each API handler method. For example, the part that enforces the minimum required permission for the `ListGroups` API currently looks like this:

```
if (!authorize(request.session, Describe, Resource.ClusterResource)) {
  sendResponseMaybeThrottle(request, requestThrottleMs =>
    request.body[ListGroupsRequest].getErrorResponse(requestThrottleMs, Errors.CLUSTER_AUTHORIZATION_FAILED.
exception))
}
```

This KIP proposes to improve this implementation:

- If the user has a *Describe (Cluster)* ACL, return all groups (this is the current behavior),
- If the user does not have *Describe (Cluster)* ACL, filter only those groups s/he has *Describe (Group)* ACL on.

Compatibility, Deprecation, and Migration Plan

- With the proposed change users who could successfully list groups before (i.e. with *Describe (Cluster)* ACL), can still do so without any change.
- Users who did not have *Describe (Cluster)* ACL, but had *Read (Group)* on some groups were not able to list those groups before. With this proposal, they can now list them. This is reasonable according to the drawback mentioned earlier. They could already describe the groups, so it only makes sense if they can list them too.

In general, *Describe (Cluster)* ACL should be given to cluster administrators only.

Rejected Alternatives

- Changing the minimum required ACL for `ListGroups` API from *Describe (Cluster)* to *Describe (Group)*. This would also work provided that cluster admins are given a wild card describe group permission so they can list all groups in the cluster. However, for the sake of backward compatibility the preference was given to the alternative, which preserves the describe cluster permission.