

[Overview](#)

[Design and Configuration](#)

[Few High Level Thoughts and Considerations](#)

[How ACL's can be configured per queue level?](#)

[Interaction with queue acls](#)

[High Level Implementation](#)

ACLs for Application Priority

Overview

Application priority feature helps certain users to run their demanding applications ahead of other lowest priority applications. Currently maximum possible cluster wide application priority can be configured by any user. Hence it is possible that everyone may try that and eventually void this feature itself. ACLs per priority could help to control the usage of priority in queue/cluster.

Design and Configuration

Few High Level Thoughts and Considerations

- Access Control Lists can be set per priority or range of priority.
- This could be configured similar to queue-mapping configuration as exists today.
- It could be better if these configuration could be pulled out to a new file.

How ACL's can be configured per queue level?

A sample configuration that can be added in capacity scheduler configuration file is given below.

Given Max cluster priority is configured as 12, generally one could configure cluster wise priority acls as follows (common for all queues)

```
<property>
<name>yarn.scheduler.capacity.priority-acls</name>
<value>user=maria.jacob max-priority=4 default-priority=3:group=webadmins max-priority=6
default-priority=4</value>
</property>
```

Also each queue (in this example it is shown for “default” queue) can override this configuration as below.

```
<property>
  <name>yarn.scheduler.capacity.default.priority-acls</name>
  <value>user=maria,jacob max-priority=4 default-priority=3:group=webadmins max-priority=5
  default-priority=3</value>
</property>
```

These configurations could be overridden by sub queues if needed.

Under each acl labels for priority, a set of users and groups can be configured and only those users/groups should be able to run that specified priority (or range of priority) applications.

Interaction with queue acls

If Queue level ACLs are configured, then this priority-based acl configuration should be a subset of that configuration – a user should have both queue-acl and priority-acl to be able to submit applications at that priority in that queue.

Note: Under both scenarios mentioned above, those applications which are not able to meet the ACL criteria could fall back to a “default priority” and continue submitting application. This “default priority” will be calculated as the upper range (or single) of configured priority range of given user acl.

High Level Implementation

At a very high level, currently PrivilegedEntity supports only Queue. There could another entity for Priority. This could be used in an AuthorizationProvider. Current providers support queue by default. This could be extended to use priority based PrivilegedEntity.

This means ConfiguredYarnAuthorizer could also support PrivilegedEntity.PRIORITY and verify UGI access per application.