

git_comments:**git_commits:**

1. **summary:** [SPARK-11314][YARN] add service API and test service for Yarn Cluster schedulers
message: [SPARK-11314][YARN] add service API and test service for Yarn Cluster schedulers This is purely the yarn/src/main and yarn/src/test bits of the YARN ATS integration: the extension model to load and run implementations of `SchedulerExtensionService` in the yarn cluster scheduler process —and to stop them afterwards. There's duplication between the two schedulers, yarn-client and yarn-cluster, at least in terms of setting everything up, because the common superclass, `YarnSchedulerBackend` is in spark-core, and the extension services need the YARN app/attempt IDs. If you look at how the the extension services are loaded, the case class `SchedulerExtensionServiceBinding` is used to pass in config info -currently just the spark context and the yarn IDs, of which one, the attemptID, will be null when running client-side. I'm passing in a case class to ensure that it would be possible in future to add extra arguments to the binding class, yet, as the method signature will not have changed, still be able to load existing services. There's no functional extension service here, just one for testing. The real tests come in the bigger pull requests. At the same time, there's no restriction of this extension service purely to the ATS history publisher. Anything else that wants to listen to the spark context and publish events could use this, and I'd also consider writing one for the YARN-913 registry service, so that the URLs of the web UI would be locatable through that (low priority; would make more sense if integrated with a REST client). There's no minicluster test. Given the test execution overhead of setting up minicluster tests, it'd probably be better to add an extension service into one of the existing tests. Author: Steve Loughran <stevel@hortonworks.com>
Closes #9182 from steveloughran/stevel/feature/SPARK-1537-service.

github_issues:**github_issues_comments:****github_pulls:**

1. **title:** [SPARK-11314] [YARN] add service API and test service for Yarn Cluster schedulers
body: This is purely the yarn/src/main and yarn/src/test bits of the YARN ATS integration: the extension model to load and run implementations of `SchedulerExtensionService` in the yarn cluster scheduler process —and to stop them afterwards. There's duplication between the two schedulers, yarn-client and yarn-cluster, at least in terms of setting everything up, because the common superclass, `YarnSchedulerBackend` is in spark-core, and the extension services need the YARN app/attempt IDs. If you look at how the the extension services are loaded, the case class `SchedulerExtensionServiceBinding` is used to pass in config info -currently just the spark context and the yarn IDs, of which one, the attemptID, will be null when running client-side. I'm passing in a case class to ensure that it would be possible in future to add extra arguments to the binding class, yet, as the method signature will not have changed, still be able to load existing services. There's no functional extension service here, just one for testing. The real tests come in the bigger pull requests. At the same time, there's no restriction of this extension service purely to the ATS history publisher. Anything else that wants to listen to the spark context and publish events could use this, and I'd also consider writing one for the YARN-913 registry service, so that the URLs of the web UI would be locatable through that (low priority; would make more sense if integrated with a REST client). There's no minicluster test. Given the test execution overhead of setting up minicluster tests, it'd probably be better to add an extension service into one of the existing tests.

github_pulls_comments:

1. ****[Test build #43985 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/43985/consoleFull)**** for PR 9182 at commit [`e796bb3`](https://github.com/apache/spark/commit/e796bb33d3128587f87529284fe45b37e1205de5). - This patch passes all tests. - This patch merges cleanly. - This patch adds the following public classes `_(experimental)_`:
`\n * `trait SchedulerExtensionService` \n * `case class SchedulerExtensionServiceBinding` \n`
2. > There's duplication between the two schedulers, ..., because the common superclass, YarnSchedulerBackend is in spark-core I'd just move that class to the yarn module. There's no reason why it has to be in spark-core.
3. -Will do. Then I can combine everything nicely.
4. Also, could you please follow the PR title convention? See <https://cwiki.apache.org/confluence/display/SPARK/Contributing+to+Spark#ContributingtoSpark-PullRequest> It would also be nice to file sub-tasks instead of piling a bunch of PRs on top of SPARK-1537.
5. ****[Test build #44368 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/44368/consoleFull)**** for PR 9182 at commit [`d018a38`](https://github.com/apache/spark/commit/d018a38c019906e26b9ab2371e5c1e6f1b2d2faa). - This patch passes all tests. - This patch merges cleanly. - This patch adds no public classes.
6. Looks OK to me, mostly just style nits. Also, needs a rebase.
7. ****[Test build #44524 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/44524/consoleFull)**** for PR 9182 at commit [`a4358d5`](https://github.com/apache/spark/commit/a4358d5b23dd2d7db706574124e4c69d1171ffb4). - This patch passes all tests. - This patch ****does not merge cleanly****. - This patch adds no public classes.
8. rebase to follow
9. ****[Test build #44537 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/44537/consoleFull)**** for PR 9182 at commit [`8171a98`](https://github.com/apache/spark/commit/8171a982ddbed930227583243b230bd35d89d047). - This patch passes all tests. - This patch merges cleanly. - This patch adds no public classes.
10. ****[Test build #44536 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/44536/consoleFull)**** for PR 9182 at commit [`32b8655`](https://github.com/apache/spark/commit/32b86557c790daed45b7b1d3e47f7beba76e91f69). - This patch passes all tests. - This patch ****does not merge cleanly****. - This patch adds no public classes.
11. ****[Test build #44593 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/44593/consoleFull)**** for PR 9182 at commit [`8a6a1f1`](https://github.com/apache/spark/commit/8a6a1f13235fd0dcc58c4106b0314098f961e67). - This patch ****fails Scala style tests****. - This patch merges cleanly. - This patch adds no public classes.
12. OK, new patch with uses String values for appId and app attempt ID in the service extension binding —but explicitly derives these from the YARN app & attempt IDs, so that services and downstream code can be confident they will be locatable in the YARN services (RM, NM, etc). Tests (here and dependent pull requests) all updated to match. I've moved the Stub app/attempt ID classes out, as they are now only needed for integration tests in the final history provider patch.
13. ****[Test build #44682 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/44682/consoleFull)**** for PR 9182 at commit [`fa6b9b1`](https://github.com/apache/spark/commit/fa6b9b11159a06bedcef0ce36e0c864cd947a5de). - This patch ****fails PySpark unit tests****. - This patch merges cleanly. - This patch adds no public classes.
14. So my whole feedback about the IDs wasn't because of the type, it was because if you could use the existing values, you wouldn't need to add all the code related to plumbing the values from the ApplicationMaster or wherever else, and you wouldn't need that `bindToYarn` method you're adding. If you really need the full attempt ID for the service you're adding later, then you still need all that code, and it doesn't matter whether you expose a String or the actual type.
15. and I've already gone and moved to strings. never mind. the existing attempt IDs are nice for humans in the web UI, and potentially in the rest, but don't let you hook up to yarn's internals. now, should I stay with String or roll back?

16. > now, should I stay with String or roll back? Up to you. I was hoping you did not need to use the full attempt id and instead be able to get rid of the code you're adding, but...
17. BTW, if you're keeping `bindToYarn` and friends, you could change `applicationId` and `applicationAttemptId` to return the values you're setting there, which also means you could have a single implementation for the two methods (instead of the current separate impls for client and cluster modes).
18. This just rolls back to the explicit app/attempt ID. it's got better guarantees of what's coming down
19. ****[Test build #45681 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/45681/consoleFull)**** for PR 9182 at commit `[810cb75]` (https://github.com/apache/spark/commit/810cb7591762fd5f325a9023690edc17b7566dda). - This patch passes all tests. - This patch ****does not merge cleanly****. - This patch adds no public classes.
20. @steveloughran could you take a look at my previous suggestion? With your current patch there are two ways that the application / attempt ID are propagated to the scheduler backend; it would be nice to consolidate them.
21. will do...ignore the latest push as its a just a rebase of the existing work prior to any other edits
22. OK, this is what I've done 1. AM doesn't propagate app/attempt IDs via system properties. It still sets them for other things to use, including `SparkContext`, but its not used to set up the `YarnClusterSchedulerBackend`. 2. `YarnSchedulerBackend` uses its `appId` and `appAttemptId` fields to generate the string values for the `applicationAttemptId()` and `applicationId()`; the subclasses have had those cut. 3. `YarnSchedulerBackend.appId` is now a protected option; its referenced in `YarnClientSchedulerBackend`, references now using `isDefined` and `get` where appropriate. This simplifies the two backend subclasses, as their parent does more. I'm not sure that the direct accessing of the `appId` is particularly elegant though
23. retest this please
24. @steveloughran I think you forgot to upload the new code?
25. (catching up: looks like the git tool I'm using decided to make up a new remote branch when I renamed the local. Fixing to track the old one and pushing up again)
26. ****[Test build #46892 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/46892/consoleFull)**** for PR 9182 at commit `[3663a91]` (https://github.com/apache/spark/commit/3663a9181a6a99e93e5c7a487111b5c105992336). - This patch passes all tests. - This patch merges cleanly. - This patch adds no public classes.
27. A few minor things left, otherwise looks ok.
28. thanks -will deal with these on wednesday.
29. ****[Test build #47072 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/47072/consoleFull)**** for PR 9182 at commit `[b9a1834]` (https://github.com/apache/spark/commit/b9a183467e529dbde79bbf1548c0c921eba18d0e). - This patch ****fails from timeout after a configured wait of `250m`****. - This patch merges cleanly. - This patch adds no public classes.
30. retest this please
31. LGTM pending tests.
32. ****[Test build #47084 has finished](https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/47084/consoleFull)**** for PR 9182 at commit `[b9a1834]` (https://github.com/apache/spark/commit/b9a183467e529dbde79bbf1548c0c921eba18d0e). - This patch passes all tests. - This patch merges cleanly. - This patch adds no public classes.
33. Merging to master.
34. It looks like some MiMa tests are now failing in the Hadoop pre-YARN builds: ```` [info] spark-core: found 1 potential binary incompatibilities (filtered 715) [error] * class org.apache.spark.scheduler.cluster.YarnSchedulerBackend#YarnSchedulerEndpoint does not have a correspondent in new version [error] filter with: ProblemFilters.exclude[MissingClassProblem] ```` https://amplab.cs.berkeley.edu/jenkins/job/Spark-Master-SBT/AMPLAB_JENKINS_BUILD_PROFILE=hadoop1.0,label=spark-test/4165/console Could you take a quick look?
35. I'll add an exclusion.

github_pulls_reviews:

1. nit: 2 spaces indent.
2. I think here we need to set `started` to `false`.
3. Do we need to try catch some exceptions like `ClassNotFound` here?
4. nit: this import can be merged with above one.
5. `bindToYarn`?
6. nit: one more empty line.
7. Can we put these fake stub classes into one file like: `SparkYarnTestHelper` or something else? That will possibly reduce the file number. Both for here `StubApplicationAttemptId` and `StubApplicationId`. If nowhere else use these classes, we could even put these classes into `ExtensionServiceIntegrationSuite.scala`. Just my thought :).
8. Here `binding` is actually duplicated with below 3 parameters, from my understanding in this code, we could choose either.
9. fixed, +lines directly below
10. good catch. will set & skip the operation if !started. This will permit reissuing of the start() call from a stopped service, but a full state machine here is probably overkill.
11. I thought about that, but consider this: when would you want failure to load your listed extension services as something not to fail on? Do you want it to quietly downgrade, vs noisily fail? maybe we could make it an option
12. OK
13. got it
14. I'm using them more in the tests in the later patches. I can (and will) move them into the test helper, but be assured, there's a lot more tests to come.
15. OK, saving binding as a field; converting the others to local vars.
16. nit: this goes before the previous import
17. When will this be not set? I assume in client mode? Could you mention that in the scaladoc above?
18. nit: indentation here is weird.
19. Hmm... `SchedulerExtensionServices` should probably mention that implementations must have an empty constructor.
20. minor: instead of another call to `map` you could add the `toList` call to the code inside the previous closure.
21. nit: move before previous import
22. nit: "client mode schedulers" is more in line with how the rest of code refers to things.
23. super minor, but maybe do a try..finally here just in case `super.stop()` throws?
24. nit: `classOf[SimpleExtensionService].getName()`
25. nit: nuke the extra empty lines.
26. correct: only the AM-side bindings have it. Will cover in the javadocs.
27. I thin I'd tried to manually edit it from Jerry's comment & got it wrong. Will try the IDE this time.
28. I can add that in the docs —or what if converted `SchedulerExtensionService` from a trait to a class, and added an empty constructor there, with the documentation on that constructor?
29. OK
30. uh oh, this looks like IDEA may be imposing a different sort order than expected, especially on `{` chars. This is what the IDE-generated order looks like, based on the instructions in the spark style guide: ```` import scala.concurrent.{ExecutionContext, Future} import scala.util.control.NonFatal import org.apache.hadoop.yarn.api.records.{ApplicationAttemptId, ApplicationId} import org.apache.spark.rpc._ import org.apache.spark.scheduler._ import org.apache.spark.scheduler.cluster.CoarseGrainedClusterMessages._ import org.apache.spark.ui.JettyUtils import org.apache.spark.util.`

- ```
{RpcUtils, ThreadUtils} import org.apache.spark.{Logging, SparkContext} ```` What's it getting wrong? as in "what will I have to review all my source files for"?
```
31. always good to be cautious. Now, should I try to be clever in that finally() and downgrade any service stop exceptions into logged events? That way, if the superclass did raise something, it wouldn't get lost by a second exception in the finally clause?
  32. whitespace line; will fix
  33. The thing it's getting wrong is that it's listing child packages (e.g. `org.apache.spark.rpc`) before the parent one (`org.apache.spark`). I'm not sure how to configure IDEs to handle that (I don't use one myself).
  34. Good question about what to do with exceptions. I think the best approach is to have `stop()` be called from a method that catches and logs exceptions. That way you kill two problems: you'll not replace any exceptions caused by `super.stop()`, and you'll make sure `stop()` is called for all existing services (which the current code doesn't do when exceptions are thrown).
  35. Still out of order.
  36. Just a suggestion in general (you don't need to change that here if you don't want to), but you can use a shorter syntax in these cases: ```` def getServices: List[SchedulerExtensionService] = services ````
  37. This class doesn't have a `toString` so this will probably look ugly.
  38. Could use the already existing `applicationId` and `applicationAttemptId` methods instead of having to add `bindToYarn` and related code you're adding in this PR? From a quick look at the code it seems like things are initialized in the right order for those to work.
  39. (update: I see that doing that would mean changing types, but do you need to expose the YARN types to the service? Is there a method to parse the String into the YARN type if needed?)
  40. OK
  41. I know what's up. It's sorting alphabetically within a group, and comes `{` after the alphabet, so child packages come first. I'll review these things by hand & will have to do the same through the other patches. Something to call out on the spark style guide maybe —it does cover the IDEA import patterns, but not this quirk
  42. string parsing has proven fairly brittle in the past; the move from single to multiple attempts broke all apps trying to do it across versions (i.e. a hadoop 2.2 parser in a 2.5 cluster). Unless you want to base-64 encode the protobuf representation, I'd avoid that.
  43. added one
  44. But do you need the parsed information? e.g. `ApplicationId` has a "cluster timestamp" and an id; I don't see much use in providing those separately to these services, the string id seems good enough in my view.
  45. For the history service: yes. What I didn't want to do was write an extension service which suited exactly one use case though. Still, there's not much you can do with an appId that doesn't involve the YARN client or AM/RPM RPC channels, and if a some extension wants to do that, it'll probably need more than just the ID structures. For other uses (e.g. grabbing the app reports, looking things up in the RM web UI, etc, those string values will work). I'll go to strings. W.r.t the existing methods, the application attempt ID that exists today is the tail of the (parsed) app attempt string, "1", "2", .... For the history stuff I do need the full attemptID for its uniqueness.
  46. Do we need this import?
  47. > For the history stuff I do need the full attemptID for its uniqueness. Couldn't you create a unique ID from app ID + current non-universally-unique attempt ID?
  48. No, because that's not the yarn attempt ID. I explicitly use that to query the YARN RM & identify attempts/apps which have finished/disappeared without closing their event stream. So I do need that attempt ID, as would any other plugin that wanted to work with the RM. They could \_probably\_ get away with it as a string. And if in future it turns out to not be the case, the use of a binding class will allow it to be added without breaking any existing plugins the way adding a new parameter would do
  49. Regarding that cluster timestamp, it came in with RM HA; its the thing that broke any 2.2 string parsing.
  50. afraid so. This patch reorders things so be consistent with the style guidelines; That's usually v. brittle with other patches, but as this diff includes a rename, patches will break anyway...
  51. nit: "on a client" -> "in client mode"
  52. super nit: stray empty line
  53. nit: if you use `\$appId` then the whole statement fits in the same line.
  54. `getConf` creates a copy of the conf; you can use `sparkContext.conf` which doesn't.
  55. nit: nuke this line and the previous one? (they're also indented wrongly.)
  56. fixed
  57. fixed
  58. nice! done.
  59. fixed. also scanning test code to see where I've done it
  60. done.
  61. What service? It's not clear what the comment refers to.
  62. nit: indented one space too far.
  63. `Utils.tryLogNonFatalError`?
  64. You can probably use `LocalSparkContext` instead of rolling your own code for cleaning this up.
  65. replacing with `/\*\* Application ID. \*/` ; the `bindToYarn()` scaladocs state what's needed.
  66. got it. IntelliJ IDEA policy games
  67. done. Will look for other uses elsewhere too
  68. will do

#### jira\_issues:

1. **summary:** Umbrella: Add a way to register long-lived services in a YARN cluster  
**description:** In a YARN cluster you can't predict where services will come up -or on what ports. The services need to work those things out as they come up and then publish them somewhere. Applications need to be able to find the service instance they are to bond to -and not any others in the cluster. Some kind of service registry -in the RM, in ZK, could do this. If the RM held the write access to the ZK nodes, it would be more secure than having apps register with ZK themselves.

#### jira\_issues\_comments:

1. You can list running apps today, but # anyone can start >1 app with the same name # The AM can only register a URL for the AM, and its own RPC service -not any other RPC ports opened, web pages served, etc. At the very least, I'd like to be able to launch an AM with a given app-type & app-name and be confident that no other YARN application with the same (type, name) was running or being started in my name. With the guarantee of a unique (user, type, name) I could use the RM as a bit of a service registry, albeit one limited in what it can register.
2. ZK backed registry looks like a good implementation option. Instead of application specific flavors of service registry implementations through ZK, it would be great to have a reusable implementation as part of YARN. For publishing services to other applications, but also within an application. For example, containers may need to find the AM after recovery (assuming that containers remain active while the AM goes down and is recovered). Encapsulating such registry in the RM would provide for a single point of configuration and secure write access (and pluggable implementation).
3. HDFS-4849 has shown value in having some per-instance unique IDs. A service registry could help here too
4. I think YARN-1143 might be relevant here: restrict the valid types and names of apps to hostname-valid strings. This would let us provide a DNS-based registry. Maybe the restriction could be applied purely to long-lived services if we felt it was too late

5. I need to write a service registry for another project that has a lot of overlap with this JIRA (multi-tenant storm). I don't have a lot of time to wait for us to all agree on the details but I would like to put the design and APIs up here for review so that hopefully the work can be mostly reused.
6. What I'm doing right now is just enumerating all instances of my app's type and verifying that the (username, instance-name) is unique : [\[https://github.com/hortonworks/hoya/blob/master/hoya-core/src/main/java/org/apache/hadoop/hoya/yarn/client/HoyaClient.java#L841\]](https://github.com/hortonworks/hoya/blob/master/hoya-core/src/main/java/org/apache/hadoop/hoya/yarn/client/HoyaClient.java#L841) That's got race condition built in to it
7. Yes it does have plenty of races. I'll try to get some detailed designs up shortly but at a high level the general idea is to have a restful web service. For the most common use case there just needs to be two interfaces. - Register/Monitor a Service - Query for Services Part of the reason we need the service registry is to securely verify that a client is talking to the real service, and no one has grabbed the service's port after it registered. To do that I want to have the concept of a verified service. For that we would need an admin interface for adding, updating, and removing verified services. The registry would provide a number of pluggable ways for services to authenticate. Part of adding a verified service would include indicating which authentication models the service can use to register and which "users" are allowed to register that service. The registry could also act like a trusted Certificate Authority. Another part of adding in a verified service would include indicating how clients could verify they are talking to the true service. This could include just publishing an application id so the client can go to the RM and get a delegation token. Another option would be having the service generate a public/private key pair. When the service registers it would get the private key and the public key would be available through the discovery interface. The plan is to also have the registry monitor the service similar to ZK. The service would heartbeat in to the registry periodically (could be on the order of mins depending on the service) after a certain period of time of inactivity the service would be removed from the registry. Perhaps we should add in an explicit unregister as well. I want to make sure that the data model it is generic enough that we could support something like a web service on the grid where each server can register itself and all of them would show up in the registry, so a service could have one or more servers that are a part of it, and each server could have some separate metadata about it. I also want to have a plug-in interface for discovery, so we could potentially make the registry look like a DNS server or an SSL Certificate Authority which would make compatibility with existing applications and clients a lot simpler.
8. Uploading a file that shows some examples of the registration service APIs. Any feedback on them is appreciated.
9. # your design reminds me a lot of bonjour, just a thought. # my use case is "ensure that no other instance of my application name exists, eg {{steve/hoya/cluster4}}", so avoiding race conditions. I'd have the server attempt to register on startup -and if it could not, fail. Implication: atomic registration by name # Hadoop now ships with the ZK JAR, for HA NN, soon the RM will use it too. This will let us assume that ZK is a live service, and make use of it.
10. +1 for REST API
11. Here's the registry proposal Sanjay and I have written. This is purely the core zookeeper-based model with API and YARN RM integration; REST access and other future extensions like out-of-cluster access will be separate JIRAs Code to follow
12. YARN 913 registry, patch 001 Features # ZK based registry YARN service running in the RM. It is currently optional as to whether this is enabled # yarn service for clients to use & register # mini ZK service for use in tests —and an optional feature of any MiniYarnCluster (credit to Twill team here) Not implemented # cleanup policies (after container exit, app attempt, app exit) # full security (The RM-based service creates the user dir, but the policies need to be set up to give the user write access under that —and nobody else)
13. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12666278/YARN-913-001.patch> against trunk revision 1dcaba9. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 24 new or modified test files. {color:red}-1 javac{color:red}. The patch appears to cause the build to fail. Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/4815/console> This message is automatically generated.
14. h3. Updated YARN service registry description This adds a {{persistence}} field to service records, enabling the records to be automatically deleted (along with all child entries) when the application, app attempt or container is terminated. h3. TLA+ service registry specification. This is my initial attempt to define the expected behaviour of a service registry built atop zookeeper. Corrections welcome.
15. Patch -002 # adds persistence policy # {{RegistryOperationsService}} implements callbacks for various RM events, and implements the setup/purge behaviour underneath. # adds a new class in the resource manager, {{RegistryService}}. This bridges from YARN to the registry by subscribing to application and container events, translating and forwarding to the {{RegistryOperationsService}} where they may trigger setup/purge operations # Hooks this up to the RM # Extends the DistributedShell by enabling it to register service records with the different persistence options. # Adds a test to verify the distributed shell does register the entries, and that the purgeable ones are purged after the application completes. This means the {{TestDistributedShell}} test is now capable of verifying that YARN applications can register themselves, that they can then be discovered, and that the RM cleans up after they terminate.
16. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12667181/YARN-913-002.patch> against trunk revision 0974f43. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 25 new or modified test files. {color:red}-1 javac{color:red}. The patch appears to cause the build to fail. Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/4844/console> This message is automatically generated.
17. not sure why the build is failing ... console output provides ~no details
18. assuming this patch builds (it does appear to locally), this patch # is in sync with trunk, including the new curator import of HADOOP-10982 # Adds security # has tests that bring up a kerberized ZK cluster to verify clients can work with it. # has the RM in charge of setting up paths and cleaning up after I don't think security is perfect ... I need to lock down the ACLs, and get the design docs from google drive into the docs as .md files.
19. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12669504/YARN-913-003.patch> against trunk revision f24ac42. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 32 new or modified test files. {color:red}-1 javac{color:red}. The patch appears to cause the build to fail. Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5004/console> This message is automatically generated.
20. Updated patch with ACL build-up for system permissions on secure clusters, testing in-VM with miniKDC, ZK and showing doAs registry creation creates an authed registry. Also: patched the build process so that precommit reports can now be found
21. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12669798/YARN-913-003.patch> against trunk revision 474f116. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 34 new or modified test files. {color:red}-1 javac{color:red}. The patch appears to cause the build to fail. Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5028/console> This message is automatically generated.
22. patch -004 with java6 generics-related compile errors fixed.
23. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12669814/YARN-913-004.patch> against trunk revision fe2f54d. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 34 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1267 javac compiler warnings (more than the trunk's current 1266 warnings). {color:red}-1 javadoc{color}. The javadoc tool appears to have generated 10 warning messages. See <https://builds.apache.org/job/PreCommit-YARN-Build/5031/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavadocWarnings.txt> for details. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:red}-1 findbugs{color}. The patch appears to introduce 11 new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:red}-1 core tests{color}. The patch failed these unit tests in hadoop-common-project/hadoop-minikdc hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests: org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell org.apache.hadoop.yarn.registry.secure.TestSecureLogins org.apache.hadoop.yarn.server.TestMiniYARNClusterRegistry {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5031/testReport/> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5031/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-yarn-server-resourcemanager.html> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5031/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-yarn-server-resourcemanager.html>

YARN-Build/5031//artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-yarn-registry.html Javac warnings:  
https://builds.apache.org/job/PreCommit-YARN-Build/5031//artifact/PreCommit-HADOOP-Build-patchprocess/diffJavacWarnings.txt Console  
output: https://builds.apache.org/job/PreCommit-YARN-Build/5031//console This message is automatically generated.

24. patch --006; fixes dshell test. No idea why the others are failing, and we still can't get the results yet

25. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12669983/YARN-913-006.patch> against trunk revision 6fe5c6b. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 34 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1267 javac compiler warnings (more than the trunk's current 1266 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:red}-1 eclipse:eclipse{color}. The patch failed to build with eclipse:eclipse. {color:green}+1 findbugs{color}. The patch does not introduce any new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:green}+1 core tests{color}. The patch passed unit tests in . {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: https://builds.apache.org/job/PreCommit-YARN-Build/5042//testReport/ Javac warnings: https://builds.apache.org/job/PreCommit-YARN-Build/5042//artifact/PreCommit-HADOOP-Build-patchprocess/diffJavacWarnings.txt Console output: https://builds.apache.org/job/PreCommit-YARN-Build/5042//console This message is automatically generated.

26. test result is spurious: the build was cancelled. test output from a full run {code} ----- T E S T S -----  
----- Running org.apache.hadoop.yarn.registry.client.binding.TestMarshalling Tests run: 5, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.463 sec - in org.apache.hadoop.yarn.registry.client.binding.TestMarshalling Running  
org.apache.hadoop.yarn.registry.client.binding.TestRegistryPathUtils Tests run: 12, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.291 sec - in  
org.apache.hadoop.yarn.registry.client.binding.TestRegistryPathUtils Running  
org.apache.hadoop.yarn.registry.client.services.TestMicroZookeeperService Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.522 sec -  
in org.apache.hadoop.yarn.registry.client.services.TestMicroZookeeperService Running  
org.apache.hadoop.yarn.registry.client.services.TestCuratorService Tests run: 20, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 2.852 sec - in  
org.apache.hadoop.yarn.registry.client.services.TestCuratorService Running org.apache.hadoop.yarn.registry.operations.TestRegistryOperations Tests  
run: 23, Failures: 0, Errors: 14, Skipped: 0, Time elapsed: 3.075 sec <<< FAILURE! - in  
org.apache.hadoop.yarn.registry.operations.TestRegistryOperations  
testLsEmptyPath(org.apache.hadoop.yarn.registry.operations.TestRegistryOperations) Time elapsed: 0.449 sec <<< ERROR! java.lang.Exception:  
Unexpected exception, expected<org.apache.hadoop.fs.PathNotFoundException> but  
was<org.apache.hadoop.yarn.registry.client.exceptions.InvalidPathnameException> at  
org.apache.zookeeper.common.PathUtils.validatePath(PathUtils.java:99) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.validateZKPath(RegistryPathUtils.java:54) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.createFullPath(RegistryPathUtils.java:90) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.createFullPath(CuratorService.java:304) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkList(CuratorService.java:696) at  
org.apache.hadoop.yarn.registry.client.services.RegistryOperationsService.list(RegistryOperationsService.java:173) at  
org.apache.hadoop.yarn.registry.operations.TestRegistryOperations.testLsEmptyPath(TestRegistryOperations.java:119)  
testDeleteNonEmpty(org.apache.hadoop.yarn.registry.operations.TestRegistryOperations) Time elapsed: 0.181 sec <<< ERROR!  
org.apache.hadoop.yarn.registry.client.exceptions.RegistryIOException: '/users/devteam/org-apache-hadoop/hdfs': Failure of existence check on  
/users/devteam/org-apache-hadoop/hdfs: org.apache.hadoop.yarn.registry.client.exceptions.InvalidPathnameException: '/registry//users/devteam/org-  
apache-hadoop/hdfs': Invalid Path "/registry//users/devteam/org-apache-hadoop/hdfs" : java.lang.IllegalArgumentException: Invalid path string  
"/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty node name specified @10: Invalid path string "/registry//users/devteam/org-  
apache-hadoop/hdfs" caused by empty node name specified @10: '/registry//users/devteam/org-apache-hadoop/hdfs': Invalid Path  
"/registry//users/devteam/org-apache-hadoop/hdfs" : java.lang.IllegalArgumentException: Invalid path string "/registry//users/devteam/org-apache-  
hadoop/hdfs" caused by empty node name specified @10: Invalid path string "/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty  
node name specified @10 at org.apache.zookeeper.common.PathUtils.validatePath(PathUtils.java:99) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.validateZKPath(RegistryPathUtils.java:54) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.createFullPath(RegistryPathUtils.java:90) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.createFullPath(CuratorService.java:304) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkStat(CuratorService.java:469) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkPathExists(CuratorService.java:511) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkSet(CuratorService.java:645) at  
org.apache.hadoop.yarn.registry.client.services.RegistryOperationsService.create(RegistryOperationsService.java:135) at  
org.apache.hadoop.yarn.registry.AbstractRegistryTest.putExampleServiceEntry(AbstractRegistryTest.java:88) at  
org.apache.hadoop.yarn.registry.AbstractRegistryTest.putExampleServiceEntry(AbstractRegistryTest.java:70) at  
org.apache.hadoop.yarn.registry.operations.TestRegistryOperations.testDeleteNonEmpty(TestRegistryOperations.java:100)  
testPutGetContainerPersistenceServiceEntry(org.apache.hadoop.yarn.registry.operations.TestRegistryOperations) Time elapsed: 0.098 sec <<<  
ERROR! org.apache.hadoop.yarn.registry.client.exceptions.RegistryIOException: '/users/devteam/org-apache-hadoop/hdfs': Failure of existence  
check on /users/devteam/org-apache-hadoop/hdfs: org.apache.hadoop.yarn.registry.client.exceptions.InvalidPathnameException:  
'/registry//users/devteam/org-apache-hadoop/hdfs': Invalid Path "/registry//users/devteam/org-apache-hadoop/hdfs" :  
java.lang.IllegalArgumentException: Invalid path string "/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty node name specified  
@10: Invalid path string "/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty node name specified @10: '/registry//users/devteam/org-  
apache-hadoop/hdfs': Invalid Path "/registry//users/devteam/org-apache-hadoop/hdfs" : java.lang.IllegalArgumentException: Invalid path string  
"/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty node name specified @10: Invalid path string "/registry//users/devteam/org-  
apache-hadoop/hdfs" caused by empty node name specified @10 at org.apache.zookeeper.common.PathUtils.validatePath(PathUtils.java:99) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.validateZKPath(RegistryPathUtils.java:54) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.createFullPath(RegistryPathUtils.java:90) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.createFullPath(CuratorService.java:304) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkStat(CuratorService.java:469) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkPathExists(CuratorService.java:511) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkSet(CuratorService.java:645) at  
org.apache.hadoop.yarn.registry.client.services.RegistryOperationsService.create(RegistryOperationsService.java:135) at  
org.apache.hadoop.yarn.registry.operations.TestRegistryOperations.testPutGetContainerPersistenceServiceEntry(TestRegistryOperations.java:450)  
testStat(org.apache.hadoop.yarn.registry.operations.TestRegistryOperations) Time elapsed: 0.087 sec <<< ERROR!  
org.apache.hadoop.yarn.registry.client.exceptions.RegistryIOException: '/users/devteam/org-apache-hadoop/hdfs': Failure of existence check on  
/users/devteam/org-apache-hadoop/hdfs: org.apache.hadoop.yarn.registry.client.exceptions.InvalidPathnameException: '/registry//users/devteam/org-  
apache-hadoop/hdfs': Invalid Path "/registry//users/devteam/org-apache-hadoop/hdfs" : java.lang.IllegalArgumentException: Invalid path string  
"/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty node name specified @10: Invalid path string "/registry//users/devteam/org-  
apache-hadoop/hdfs" caused by empty node name specified @10: '/registry//users/devteam/org-apache-hadoop/hdfs': Invalid Path  
"/registry//users/devteam/org-apache-hadoop/hdfs" : java.lang.IllegalArgumentException: Invalid path string "/registry//users/devteam/org-apache-  
hadoop/hdfs" caused by empty node name specified @10: Invalid path string "/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty  
node name specified @10 at org.apache.zookeeper.common.PathUtils.validatePath(PathUtils.java:99) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.validateZKPath(RegistryPathUtils.java:54) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.createFullPath(RegistryPathUtils.java:90) at

[illegible]

org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.createFullPath(RegistryPathUtils.java:90) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.createFullPath(CuratorService.java:304) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkDelete(CuratorService.java:671) at  
org.apache.hadoop.yarn.registry.client.services.RegistryOperationsService.delete(RegistryOperationsService.java:192) at  
org.apache.hadoop.yarn.registry.operations.TestRegistryOperations.testDeleteNonexistentEntry(TestRegistryOperations.java:68)  
testLsParent(org.apache.hadoop.yarn.registry.operations.TestRegistryOperations) Time elapsed: 0.067 sec <<< ERROR!  
org.apache.hadoop.yarn.registry.client.exceptions.RegistryIOException: `/users/devteam/org-apache-hadoop/hdfs': Failure of existence check on  
/users/devteam/org-apache-hadoop/hdfs: org.apache.hadoop.yarn.registry.client.exceptions.InvalidPathnameException: `/registry//users/devteam/org-  
apache-hadoop/hdfs': Invalid Path "/registry//users/devteam/org-apache-hadoop/hdfs" : java.lang.IllegalArgumentException: Invalid path string  
"/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty node name specified @10: Invalid path string "/registry//users/devteam/org-  
apache-hadoop/hdfs" caused by empty node name specified @10: `/registry//users/devteam/org-apache-hadoop/hdfs': Invalid Path  
"/registry//users/devteam/org-apache-hadoop/hdfs" : java.lang.IllegalArgumentException: Invalid path string "/registry//users/devteam/org-apache-  
hadoop/hdfs" caused by empty node name specified @10: Invalid path string "/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty  
node name specified @10 at org.apache.zookeeper.common.PathUtils.validatePath(PathUtils.java:99) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.validateZKPath(RegistryPathUtils.java:54) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.createFullPath(RegistryPathUtils.java:90) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.createFullPath(CuratorService.java:304) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkStat(CuratorService.java:469) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkPathExists(CuratorService.java:511) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkSet(CuratorService.java:645) at  
org.apache.hadoop.yarn.registry.client.services.RegistryOperationsService.create(RegistryOperationsService.java:135) at  
org.apache.hadoop.yarn.registry.AbstractRegistryTest.putExampleServiceEntry(AbstractRegistryTest.java:88) at  
org.apache.hadoop.yarn.registry.AbstractRegistryTest.putExampleServiceEntry(AbstractRegistryTest.java:70) at  
org.apache.hadoop.yarn.registry.operations.TestRegistryOperations.testLsParent(TestRegistryOperations.java:83)  
testStatEmptyPath(org.apache.hadoop.yarn.registry.operations.TestRegistryOperations) Time elapsed: 0.043 sec <<< ERROR! java.lang.Exception:  
Unexpected exception, expected<org.apache.hadoop.fs.PathNotFoundException> but  
was<org.apache.hadoop.yarn.registry.client.exceptions.InvalidPathnameException> at  
org.apache.zookeeper.common.PathUtils.validatePath(PathUtils.java:99) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.validateZKPath(RegistryPathUtils.java:54) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.createFullPath(RegistryPathUtils.java:90) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.createFullPath(CuratorService.java:304) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkStat(CuratorService.java:469) at  
org.apache.hadoop.yarn.registry.client.services.RegistryOperationsService.stat(RegistryOperationsService.java:156) at  
org.apache.hadoop.yarn.registry.operations.TestRegistryOperations.testStatEmptyPath(TestRegistryOperations.java:114)  
testDeleteServiceEntry(org.apache.hadoop.yarn.registry.operations.TestRegistryOperations) Time elapsed: 0.062 sec <<< ERROR!  
org.apache.hadoop.yarn.registry.client.exceptions.RegistryIOException: `/users/devteam/org-apache-hadoop/hdfs': Failure of existence check on  
/users/devteam/org-apache-hadoop/hdfs: org.apache.hadoop.yarn.registry.client.exceptions.InvalidPathnameException: `/registry//users/devteam/org-  
apache-hadoop/hdfs': Invalid Path "/registry//users/devteam/org-apache-hadoop/hdfs" : java.lang.IllegalArgumentException: Invalid path string  
"/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty node name specified @10: Invalid path string "/registry//users/devteam/org-  
apache-hadoop/hdfs" caused by empty node name specified @10: `/registry//users/devteam/org-apache-hadoop/hdfs': Invalid Path  
"/registry//users/devteam/org-apache-hadoop/hdfs" : java.lang.IllegalArgumentException: Invalid path string "/registry//users/devteam/org-apache-  
hadoop/hdfs" caused by empty node name specified @10: Invalid path string "/registry//users/devteam/org-apache-hadoop/hdfs" caused by empty  
node name specified @10 at org.apache.zookeeper.common.PathUtils.validatePath(PathUtils.java:99) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.validateZKPath(RegistryPathUtils.java:54) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.createFullPath(RegistryPathUtils.java:90) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.createFullPath(CuratorService.java:304) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkStat(CuratorService.java:469) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkPathExists(CuratorService.java:511) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkSet(CuratorService.java:645) at  
org.apache.hadoop.yarn.registry.client.services.RegistryOperationsService.create(RegistryOperationsService.java:135) at  
org.apache.hadoop.yarn.registry.AbstractRegistryTest.putExampleServiceEntry(AbstractRegistryTest.java:88) at  
org.apache.hadoop.yarn.registry.AbstractRegistryTest.putExampleServiceEntry(AbstractRegistryTest.java:70) at  
org.apache.hadoop.yarn.registry.operations.TestRegistryOperations.testDeleteServiceEntry(TestRegistryOperations.java:62)  
testDoubleMkdir(org.apache.hadoop.yarn.registry.operations.TestRegistryOperations) Time elapsed: 0.043 sec <<< ERROR!  
org.apache.hadoop.yarn.registry.client.exceptions.InvalidPathnameException: `/registry//users/devteam': Invalid Path "/registry//users/devteam" :  
java.lang.IllegalArgumentException: Invalid path string "/registry//users/devteam" caused by empty node name specified @10: Invalid path string  
"/registry//users/devteam" caused by empty node name specified @10 at org.apache.zookeeper.common.PathUtils.validatePath(PathUtils.java:99) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.validateZKPath(RegistryPathUtils.java:54) at  
org.apache.hadoop.yarn.registry.client.binding.RegistryPathUtils.createFullPath(RegistryPathUtils.java:90) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.createFullPath(CuratorService.java:304) at  
org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkMkPath(CuratorService.java:543) at  
org.apache.hadoop.yarn.registry.client.services.RegistryOperationsService.mknode(RegistryOperationsService.java:114) at  
org.apache.hadoop.yarn.registry.operations.TestRegistryOperations.testDoubleMkdir(TestRegistryOperations.java:142) Running  
org.apache.hadoop.yarn.registry.secure.TestSecureRegistry Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 5.993 sec - in  
org.apache.hadoop.yarn.registry.secure.TestSecureRegistry Running org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations Tests  
run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 5.738 sec - in org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations  
Running org.apache.hadoop.yarn.registry.secure.TestRegistrySecurityHelper Tests run: 12, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.553 sec -  
in org.apache.hadoop.yarn.registry.secure.TestRegistrySecurityHelper Running org.apache.hadoop.yarn.registry.secure.TestSecureLogins Tests run: 8,  
Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 5.371 sec <<< FAILURE! - in org.apache.hadoop.yarn.registry.secure.TestSecureLogins  
testZKinKeytab(org.apache.hadoop.yarn.registry.secure.TestSecureLogins) Time elapsed: 0.154 sec <<< ERROR! java.io.IOException: Cannot run  
program "ktutil": java.io.IOException: error=2, No such file or directory at java.lang.UNIXProcess.<init>(UNIXProcess.java:148) at  
java.lang.ProcessImpl.start(ProcessImpl.java:65) at java.lang.ProcessBuilder.start(ProcessBuilder.java:452) at  
org.apache.hadoop.util.Shell.runCommand(Shell.java:485) at org.apache.hadoop.util.Shell.run(Shell.java:455) at  
org.apache.hadoop.util.Shell\$ShellCommandExecutor.execute(Shell.java:702) at org.apache.hadoop.util.Shell.execCommand(Shell.java:791) at  
org.apache.hadoop.util.Shell.execCommand(Shell.java:774) at  
org.apache.hadoop.yarn.registry.RegistryTestHelper.ktList(RegistryTestHelper.java:321) at  
org.apache.hadoop.yarn.registry.secure.TestSecureLogins.testZKinKeytab(TestSecureLogins.java:57) Results : Tests in error:  
TestRegistryOperations.testLsEmptyPath ? Unexpected exception, expected<org.a... TestRegistryOperations.testDeleteNonEmpty:100-  
>AbstractRegistryTest.putExampleServiceEntry:70->AbstractRegistryTest.putExampleServiceEntry:88 ? RegistryIO  
TestRegistryOperations.testPutGetContainerPersistenceServiceEntry:450 ? RegistryIO TestRegistryOperations.testStat:74-  
>AbstractRegistryTest.putExampleServiceEntry:70->AbstractRegistryTest.putExampleServiceEntry:88 ? RegistryIO  
TestRegistryOperations.testMkdirNoParent:132 ? InvalidPathname `/registry//use... TestRegistryOperations.testPutGetServiceEntry:53-



- >AbstractRegistryTest.putExampleServiceEntry:88 ? RegistryIO TestRegistryOperations.testCreateComplexApplication:266 ? InvalidPathname `re... TestRegistryOperations.testResolveEmptyPath ? Unexpected exception, expected<... TestRegistryOperations.testOverwrite:215-  
 >AbstractRegistryTest.putExampleServiceEntry:70->AbstractRegistryTest.putExampleServiceEntry:88 ? RegistryIO  
 TestRegistryOperations.testDeleteNonexistentEntry:68 ? InvalidPathname `/regis... TestRegistryOperations.testLsParent:83-  
 >AbstractRegistryTest.putExampleServiceEntry:70->AbstractRegistryTest.putExampleServiceEntry:88 ? RegistryIO  
 TestRegistryOperations.testStatEmptyPath ? Unexpected exception, expected<org... TestRegistryOperations.testDeleteServiceEntry:62-  
 >AbstractRegistryTest.putExampleServiceEntry:70->AbstractRegistryTest.putExampleServiceEntry:88 ? RegistryIO  
 TestRegistryOperations.testDoubleMkdir:142 ? InvalidPathname `/registry//users... TestSecureLogins.testZKInKeytab:57-  
 >RegistryTestHelper.ktList:321 ? IO Cannot ... Tests run: 85, Failures: 0, Errors: 15, Skipped: 0 {code}
27. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12669983/YARN-913-006.patch> against trunk revision 6fe5c6b. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 34 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1267 javac compiler warnings (more than the trunk's current 1266 warnings). {color:red}-1 javadoc{color}. The javadoc tool appears to have generated 9 warning messages. See <https://builds.apache.org/job/PreCommit-YARN-Build/5041/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavadocWarnings.txt> for details. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:red}-1 findbugs{color}. The patch appears to introduce 10 new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:red}-1 core tests{color}. The patch failed these unit tests in hadoop-common-project/hadoop-minikdc hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests: org.apache.hadoop.yarn.registry.secure.TestSecureLogins org.apache.hadoop.yarn.registry.operations.TestRegistryOperations org.apache.hadoop.yarn.server.TestMiniYARNClusterRegistry {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5041/testReport/> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5041/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-yarn-server-resourcemanager.html> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5041/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-yarn-registry.html> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5041/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5041/console> This message is automatically generated.
28. Should fix Jenkins builds: javadocs, tests and findbugs, leaving only a javac somewhere.
29. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12670013/YARN-913-007.patch> against trunk revision bf27b9c. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 34 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1267 javac compiler warnings (more than the trunk's current 1266 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:red}-1 findbugs{color}. The patch appears to introduce 1 new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:green}+1 core tests{color}. The patch passed unit tests in hadoop-common-project/hadoop-minikdc hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests. {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5046/testReport/> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5046/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-yarn-server-resourcemanager.html> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5046/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5046/console> This message is automatically generated.
30. -008 patch # fixes last lurking findbug # includes the registry site documentation
31. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12670048/YARN-913-008.patch> against trunk revision 25fd69a. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 34 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1267 javac compiler warnings (more than the trunk's current 1266 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:green}+1 findbugs{color}. The patch does not introduce any new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:green}+1 core tests{color}. The patch passed unit tests in hadoop-common-project/hadoop-minikdc hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests. {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5047/testReport/> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5047/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5047/console> This message is automatically generated.
32. \* I have some concern around 'naked' zookeeper.\* config options. If another part of Hadoop uses ZK, would they be expected to use the same ZK options? e.g., what if I want a different ZK setup for a YARN registrar vs. the NN? \* Is there any risk about a user writing to a shared ZK with the RM? i.e., if a user kills the ZK used for app registry through some action, what happens to the RM and other user's bits that are running? \* Why doesn't the hostname component allow for FQDNs? \* Are we prepared for more backlash when another component requires working DNS? :) \* Is ZK actually the right thing to use here? {code} +Zookeeper has a default limit of 1MB/node. If all endpoints of a service or +component are stored in JSON attached to that node, then there is a total limit +of 1MB of all endpoint registration data. {code} If we are reducing certain fields out of fear of hitting this limit, this seems to indicate that ZK is a bad fit and/or we are using ZK incorrectly. It could be argued that ZK should contain a pointer to the actual data stored on HDFS. This allows for future growth without having to worry about blowing ZK out.
33. bq. I have some concern around 'naked' zookeeper.\* config option This something that I do think needs changing in ZK; being driven by JVM properties can work for standalone JVM servers, but not for clients. The client here sets the properties just before needed (e.g. the SASL auth details), and I was thinking of making the set-connection operation class synchronized. But...curator does some session restarting and if those JVM-wide settings are changed, there may be problems. Summary: need to fix ZK client and then have curator configure it, so the rest of us don't have to care. bq. if a user kills the ZK used for app registry through some action, what happens to the RM and other user's bits that are running # The RM isn't depending on the ZK cluster for information; it just sets up the paths for a user, and does purges of container & app lifespan parts on their completion. I've made both the setup and teardown operations async; the {{RMRegistryOperationsService}} class gets the RM event and schedules the work on its executor. If ZK is offline then these will block until the quorum is back, but it should not delay RM operations. It could block the clients and the AM starting up. # Curator supports different {{EnsembleProviders}} ... classes which provide the data needed for the client to reconnect to ZK. The code is currently only hooked up to one -the {{FixedEnsembleProvider}}, which uses a classic static ZK quorum. There's an alternative, the {{ExhibitorProvider}}, which hooks up to [Netflix Exhibitor|<https://github.com/Netflix/exhibitor/wiki>] and can do things like [[Rolling Ensemble Change|<https://github.com/Netflix/exhibitor/wiki/Rolling-Ensemble-Change>]]. This is designed for cloud deployments where a ZK server failure results in a new host coming up, with new hostname/address ... exhibitor handles the details of rebinding. I haven't added explicit support for that (straightforward) or got a test setup (harder). If you want to play with it though ... bq. Why doesn't the hostname component allow for FQDNs? do you mean in the endpoint fields? It should ... let me clarify that in the example. bq. Are we prepared for more backlash when another component requires



working DNS? The reason the initial patches here weren't building is a helper method to build up an endpoint address from an `{{InetSocketAddress}}` called `{{getHostString()}}` to get the host/FQDN, without doing any DNS work. I had to switch to `{{getHostName()}}`, which can try to do rDNS, and so rely on DNS working. bq. Is ZK the right thing to use here? # ZK gives us availability; I do plan to add a REST API later on, one that works long-haul. It's why there is deliberately no support for ephemeral nodes ... the `{{RegistryOperations}}` interface is designed to implementable by a REST client, for which there won't be any sessions to tie ephemeral nodes to. # By deliberately publishing nothing but endpoints to services, we're trying to keep the content in the store down, with the bulk data being served up by other means. In slider, we are publishing dynamically generated config files from the AM REST API; all the registry entry does is list the API + URL for that service. # I do like your idea about just sticking stuff into HDFS, S3, etc.; that's a way to share content too, including config data. It'll fit into the general category of URL formatted endpoint —maybe I should add it as an explicit address type, "filesystem"?

34. Oh, one more thing, that `{{MicroZookeeperService}}` which is used in tests? It's a YARN service-wrapped ZK microservice (based on Twill's test one), which can publish its ensemble information to registry clients running in-VM. This would make it straightforward to be deployed \*inside\* the RM ... in a small 1-2 node cluster it wouldn't be a load problem, and as the lifespan of the ZK == lifespan of RM, no worry about having a single ZK quorum outage impacting the RM. I've not put the service under the RM. Someone is free to at some point in the future.
35. I started looking at this. I have done my fair share of reviews in my life, but this is a monster patch. Can we please split this to have a realistic timeline for commit to 2.6? I propose the following sub-tasks - The addition of registry as a stand alone module. Which in itself is massive and can be further split - server side abstraction, ZK based server side implementation, client side, web-services etc. - Integration of registry into YARN - Changing YARN apps like distributed-shell - Documentation I just applied the massive patch against latest trunk and digging through it. But if you agree, we should file sub-tasks and divide up the patch. Thanks.
36. bq. Summary: need to fix ZK client and then have curator configure it, so the rest of us don't have to care. This might be a blocker then. If a client needs to talk to more than one ZK, it sounds like they are basically screwed. bq. do you mean in the endpoint fields? It should ... let me clarify that in the example. I was mainly looking at the hostname pattern: `{code} + String HOSTNAME_PATTERN = + "([a-z0-9][a-z0-9][a-z0-9\\-]*[a-z0-9])"; {code}` It doesn't appear to support periods/dots.
37. Revised patch A key change is \* support for kerberos digest (id:pass) ACLs in paths under user accounts. A user may request that all nodes/records created in a session add specific id:pass access to the nodes; these are translated into full-access-ACL entries. Why do this? It allows a user to give a long-running service the ability to manipulate part of the service registry —i.e. its own record and below— without having to worry about token expiry. It's not mandatory for long running services to do this; they can go the credential route if they want...this just makes it an option. In particular, it allows containers to add records without needing any credentials, even if the AM bootstraps the registration from one supplied at launch time. \* factory to create kerberos, anonymous and user:pass accessors to the registry \* RM automatically creates user dir with write access for the the user on app submission. \* Security model tested in \*much\* more depth. Specifically, all those different levels of access are tested to make sure that extra rights are not being granted. \* Helper methods to aid working with the registry in clients and AMs/containers. See `{{org.apache.hadoop.yarn.registry.client.binding.RegistryOperationUtils}}`
38. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12671012/YARN-913-009.patch> against trunk revision 78b452. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 36 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1266 javac compiler warnings (more than the trunk's current 1265 warnings). {color:red}-1 javadoc{color}. The javadoc tool appears to have generated 3 warning messages. See <https://builds.apache.org/job/PreCommit-YARN-Build/5104/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavadocWarnings.txt> for details. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:red}-1 findbugs{color}. The patch appears to introduce 2 new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:red}-1 core tests{color}. The patch failed these unit tests in hadoop-common-project/hadoop-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests: org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations org.apache.hadoop.yarn.server.resourcemanager.applicationsmanager.TestAMRestart The following test timeouts occurred in hadoop-common-project/hadoop-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests: org.apache.hadoop.yarn.server.resourcemanager.TestWorkPreservingRMRestart {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5104/testReport/> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5104/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-common.html> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5104/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-yarn-registry.html> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5104/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavadocWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5104/console> This message is automatically generated.
39. This patch doesn't look at why yesterday's jenkins tests failed, so if they are due to these changes, those changes won't have been fixed. Key changes are due to experience implementing a (not in this patch) read only REST view. # renamed fields in the `{{ServiceRecord}}` because Jersey ignores `{{@JsonProperty}}` annotations giving fields specific names. So no `{{yarn:id}}` `{{yarn:persistence}}` in the JSON; fields called `{{yarn_id}}` and `{{yarn_persistence}}` instead. # Specific exception `{{NoRecordException}}` to differentiate "could not resolve a node as there isn't any entry with the header used to identify service records from `{{InvalidRecordException}}`" which is only triggered on parse problems. # added a lightweight `{{list()}}` operation that only returns the child paths; the original `{{list(path) -> List<RegistryPathStatus>}}` renamed to `{{listFull}}`. There's a CLI client for this being written; it'll help validate the API & identify any further points for tuning
40. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12671298/YARN-913-010.patch> against trunk revision 9f9a222. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 36 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1266 javac compiler warnings (more than the trunk's current 1265 warnings). {color:red}-1 javadoc{color}. The javadoc tool appears to have generated 2 warning messages. See <https://builds.apache.org/job/PreCommit-YARN-Build/5131/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavadocWarnings.txt> for details. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:red}-1 findbugs{color}. The patch appears to introduce 2 new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:red}-1 core tests{color}. The patch failed these unit tests in hadoop-common-project/hadoop-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests: org.apache.hadoop.ha.TestZKFailoverControllerStress org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5131/testReport/> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5131/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-yarn-registry.html> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5131/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-common.html> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5131/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavadocWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5131/console> This message is automatically generated.

41. Distributed shell test \*appears\* to be YARN-2607, i.e. independent of this patch; HADOOP-10668 covers TestZKFailoverControllerStress intermittent failure. `{{TestSecureRMRegistryOperations}}` is a failure in the setup phase, the setup of the registry path in a `{{"zookeeper@EXAMPLE.COM".doAs()}}` clause is failing with permissions, as if the first test case has set up the path without write access. More diagnostics needed here, such as identity of user making the call, maybe start test with some diagnostics of the path `{code}`  
TestAnonReadAccess(org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations) Time elapsed: 0.099 sec <<< ERROR!  
org.apache.hadoop.service.ServiceStateException: org.apache.hadoop.fs.PathAccessDeniedException: `/registry/users / [ 1, 'world,'anyone 31, 'sas','zookeeper@EXAMPLE.COM 31, 'sas','zookeeper@EXAMPLE.COM 31, 'sas','zookeeper@EXAMPLE.COM ]': Permission denied: KeeperErrorCode = NoAuth for /registry/users at org.apache.zookeeper.KeeperException.create(KeeperException.java:113) at org.apache.zookeeper.KeeperException.create(KeeperException.java:51) at org.apache.zookeeper.ZooKeeper.create(ZooKeeper.java:783) at org.apache.curator.framework.imps.CreateBuilderImpl\$11.call(CreateBuilderImpl.java:688) at org.apache.curator.framework.imps.CreateBuilderImpl\$11.call(CreateBuilderImpl.java:672) at org.apache.curator.RetryLoop.callWithRetry(RetryLoop.java:107) at org.apache.curator.framework.imps.CreateBuilderImpl.pathInForeground(CreateBuilderImpl.java:668) at org.apache.curator.framework.imps.CreateBuilderImpl.protectedPathInForeground(CreateBuilderImpl.java:453) at org.apache.curator.framework.imps.CreateBuilderImpl.forPath(CreateBuilderImpl.java:443) at org.apache.curator.framework.imps.CreateBuilderImpl.forPath(CreateBuilderImpl.java:423) at org.apache.curator.framework.imps.CreateBuilderImpl.forPath(CreateBuilderImpl.java:44) at org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.zkMkPath(CuratorService.java:539) at org.apache.hadoop.yarn.registry.client.services.zk.CuratorService.maybeCreate(CuratorService.java:426) at org.apache.hadoop.yarn.registry.server.services.RegistryAdminService.createRootRegistryPaths(RegistryAdminService.java:201) at org.apache.hadoop.yarn.registry.server.services.RegistryAdminService.serviceStart(RegistryAdminService.java:187) at org.apache.hadoop.service.AbstractService.start(AbstractService.java:193) at org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations\$1.run(TestSecureRMRegistryOperations.java:106) at org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations\$1.run(TestSecureRMRegistryOperations.java:98) at java.security.AccessController.doPrivileged(Native Method) at javax.security.auth.Subject.doAs(Subject.java:396) at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1640) at org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations.startRMRegistryOperations(TestSecureRMRegistryOperations.java:97) at org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations.testAnonReadAccess(TestSecureRMRegistryOperations.java:130) {code}

42. Print out detailed diags (inc ACLs) on permissions problems during registry bootstrap

43. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12671457/YARN-913-011.patch> against trunk revision 662fc11. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 36 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1266 javac compiler warnings (more than the trunk's current 1265 warnings). {color:red}-1 javadoc{color}. The javadoc tool appears to have generated 2 warning messages. See <https://builds.apache.org/job/PreCommit-YARN-Build/5146/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavadocWarnings.txt> for details. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:red}-1 findbugs{color}. The patch appears to introduce 2 new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:red}-1 core tests{color}. The patch failed these unit tests in hadoop-common-project/hadoop-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests: org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5146/testReport/> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5146/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-yarn-registry.html> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5146/artifact/PreCommit-HADOOP-Build-patchprocess/newPatchFindbugsWarningshadoop-common.html> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5146/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5146/console> This message is automatically generated.

44. tightened down code, docs, javadocs, move classes around to psitions things The test for registry security failing on jenkins didn't arise last patch submission .... there's no obvious reason for that (more precisely, why it arose in the first place)

45. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12671529/YARN-913-012.patch> against trunk revision b40f433. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 36 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1266 javac compiler warnings (more than the trunk's current 1265 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:green}+1 findbugs{color}. The patch does not introduce any new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:red}-1 core tests{color}. The patch failed these unit tests in hadoop-common-project/hadoop-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests: org.apache.hadoop.ha.TestZKFailoverControllerStress org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell org.apache.hadoop.yarn.registry.secure.TestSecureRMRegistryOperations org.apache.hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesAppsModification The test build failed in hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5151/testReport/> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5151/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5151/console> This message is automatically generated.

46. patch -013; adds way more diagnostics into the exception text, to help determine why jenkins is now intermittently failing a test that works locally. (because we can't get at the test logs, just the stack traces)

47. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12671701/YARN-913-013.patch> against trunk revision 400e1bb. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 36 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1266 javac compiler warnings (more than the trunk's current 1265 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:green}+1 findbugs{color}. The patch does not introduce any new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:red}-1 core tests{color}. The patch failed these unit tests in hadoop-common-project/hadoop-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests: org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5166/testReport/> Javac warnings:

- <https://builds.apache.org/job/PreCommit-YARN-Build/5166/artifact/PreCommit-HADOOP-Build-patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5166/console> This message is automatically generated.
48. A couple of offline comments from sanjay radia # don't publish full path in {{RegistryPathStatus}} fields; it only makes moving to indirection and cross references harder in future. # don't differentiate hadoop-classic IPC from hadoop protobuf in protocol list.
49. w.r.t first comment. putting path in the registry status field \* we don't actually need to do this, not if we return the stat'd entries as a map of name:status. \* and we can pull that operation,, currently called {{listFully}} out of operations and put in {{RegistryOperationsUtils}}. this will make clear it's a separate operation; we can emphasise it's non-atomic.
50. w.r.t [~aw]'s comments: bq. If a client needs to talk to more than one ZK, it sounds like they are basically screwed. If you are grabbing binding/configs via the CLI, it's not a worry, nor if you are talking to >1 ZK quorum with the same auth policy. Its when you start tuning SASL auth and some various timeouts that this arises. This is not an issue with the registry, it's the ZK client here. bq. I was mainly looking at the hostname pattern: {code} + String HOSTNAME\_PATTERN = + "([a-z0-9][a-z0-9][a-z0-9\\-]\*[a-z0-9])"; {code} bq. It doesn't appear to support periods/dots. That's just the pattern for entries in the registry path itself; you can't give a service a name like "-#foo" as DNS won't like it. Stick whatever you want in the fields themselves. I'll javadoc that field
51. [~aw]: new constant for you+ javadocs: {code} /\*\* \* Pattern of a single entry in the registry path. : {@value}. \* <p> \* This is what constitutes a valid hostname according to current RFCs. \* Alphanumeric first two and last one digit, alphanumeric \* and hyphens allowed in between. \* <p> \* No upper limit is placed on the size of an entry. \*/ {code} Better?
52. Oh, and I renamed the field: {code} /\*\* \* Pattern of a single entry in the registry path. : {@value}. \* <p> \* This is what constitutes a valid hostname according to current RFCs. \* Alphanumeric first two and last one digit, alphanumeric \* and hyphens allowed in between. \* <p> \* No upper limit is placed on the size of an entry. \*/ String VALID\_PATH\_ENTRY\_PATTERN = "([a-z0-9][a-z0-9][a-z0-9\\-]\*[a-z0-9])"; {code}
53. Updated patch # comments & renames the {{HOSTNAME\_PATTERN}} field (for AW) # registryOperationsStatus record holds the shortname of the stat'd record, not the full path (for Sanjay) # moves the {{listFull}} operation to list then stat the children out of the core {{RegistryOperations}} API and into {{RegistryUtils}}, as it is a utility action built from the lower level operations. Migration to this across the codebase. # made that stat operation robust against child entries being deleted during the action # same for the registry purge: there may be race conditions of overlapping delete operations ... this is not an error
54. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12672203/YARN-913-014.patch> against trunk revision a469833. {color:red}-1 patch{color}. The patch command could not apply the patch. Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5191/console> This message is automatically generated.
55. parch -15; this is patch -14 rebased against trunk with a conflict fixed
56. patch -016: includes registry cli patch (-002) of YARN-2616
57. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12672406/YARN-913-016.patch> against trunk revision 875aa79. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 36 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1266 javac compiler warnings (more than the trunk's current 1265 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:red}-1 findbugs{color}. The patch appears to introduce 1 new Findbugs (version 2.0.3) warnings. {color:red}-1 release audit{color}. The applied patch generated 1 release audit warnings. {color:red}-1 core tests{color}. The patch failed these unit tests in hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests: org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell The test build failed in hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5208/testReport/> Release audit warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5208/artifact/patchprocess/patchReleaseAuditProblems.txt> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5208/artifact/patchprocess/newPatchFindbugsWarningshadoop-yarn-registry.html> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5208/artifact/patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5208/console> This message is automatically generated.
58. Failing test is still the (believed unrelated) Running org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell Tests run: 11, Failures: 1, Errors: 0, Skipped: 0, Time elapsed: 379.565 sec <<< FAILURE! - in org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell testDSRestartWithPreviousRunningContainers(org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell) Time elapsed: 38.715 sec <<< FAILURE! java.lang.AssertionError: client failed at org.junit.Assert.fail(Assert.java:88) at org.junit.Assert.assertTrue(Assert.java:41) at org.apache.hadoop.yarn.applications.distributedshell.TestDistributedShell.testDSRestartWithPreviousRunningContainers(TestDistributedShell.java:319)
59. patch -017, no changes other than resync'd to trunk
60. PDF of the -018 patch's TLA specification
61. Patch -018 # fix {{ReservationSystemTestUtil}} where the {{RMContextImpl}} constructor needed an extra argument. # TLA+ spec compiled to PDF; errors fixed
62. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12673044/YARN-913-018.patch> against trunk revision 16333b4. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 37 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1268 javac compiler warnings (more than the trunk's current 1267 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:red}-1 findbugs{color}. The patch appears to introduce 1 new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:green}+1 core tests{color}. The patch passed unit tests in hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests. {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5271/testReport/> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5271/artifact/patchprocess/newPatchFindbugsWarningshadoop-yarn-registry.html> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5271/artifact/patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5271/console> This message is automatically generated.
63. Some feedback: # rename {{RegistryOperations.create()}} to {{bind()}} # rename {{org/apache/hadoop/yarn/registry/client/services}} to {{org/apache/hadoop/yarn/registry/client/impl}} # move all ZK classes under {{org/apache/hadoop/yarn/registry/client/impl/zk}}, i.e. the current implementations of the registry client # {{RegistryOperations}} implementations to remove declaration of exceptions other than IOE. # {{RegistryOperations.resolve()}} implementation should not mention record headers in exception text: that's an implementation detail # Add README under {{org.apache.hadoop.yarn.registry.server}} to emphasize this is server-side code # Allow {{ServiceRecord}} to support arbitrary key-values # remove {{yarn\_id}} & {{yarn\_persistence}} ffields rom {{ServiceRecord}}, moving them to the set of arbitrary key-values This ensures that there isn't explicit hard-coding of the assumption "these are YARN apps" from the records.
64. Sanjay, I can do most of these, \* w.r.t. the README, we have a javadoc {{package-info.java}}, that's enough. \* I propose restricting the custom values that a service record to have to string attributes. support arbitrary JSON opens things up to people embedding entire custom JSON docs in there, which could kill the notion of having semi-standardised records that other apps can work with + published API endpoints for any extra stuff \*outside the registry\* \* I'm going to rename the yarn fields back to {{yarn:id}} and {{yarn:persistence}} if Jersey+jackson marshalls them reliably once they aren't introspection-driven. It makes the yarn-nature of them clearer.

65. TLA+ specification in sync with (forthcoming -019 patch) # uses consistent naming # service record specified as allowing arbitrary {{String|->String}} attributes; {{yarn:id}} and {{yarn:persistence}} are merely two of these
66. This patch which addresses sanjay's review comments, with the key changes being \* {{ServiceRecord}} supports arbitrary string attributes; the yarn ID and persistence are simply two of these. \* moved zk-related classes under {{/impl/zk}} package \* TLA+ spec in sync with new design I also \* moved the packaging from {{org.apache.hadoop.yarn.registry}} to {{org.apache.hadoop.registry}}. That would reduce the impact of any promotion of the registry into hadoop-common if ever desired. \* moved the registry/site documentation under yarn-site and hooked up with existing site docs. \* added the TLA toolbox generated artifacts (pdf and a toolbox dir) to {{.gitignore}}
67. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12673424/YARN-913-019.patch> against trunk revision 30d56fd. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 37 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1268 javac compiler warnings (more than the trunk's current 1267 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:red}-1 findbugs{color}. The patch appears to introduce 1 new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:green}+1 core tests{color}. The patch passed unit tests in hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests. {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5315/testReport/> Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5315/artifact/patchprocess/newPatchFindbugsWarningshadoop-yarn-registry.html> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5315/artifact/patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5315/console> This message is automatically generated.
68. javac warning is false alarm: {code} 245a246 > [WARNING] [svn, info] failed with error code 1 {code} will fix findbugs (unused assignment) warning
69. patch -020; deleted unused operation/assignment in {{RegistryUtils.statChildren()}}
70. the -021 patch is the -020 patch with all whitespace stripped, so that patch -p0 will also apply it cleanly
71. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12673460/YARN-913-020.patch> against trunk revision 9b8a35a. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 37 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1268 javac compiler warnings (more than the trunk's current 1267 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:green}+1 findbugs{color}. The patch does not introduce any new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:red}-1 core tests{color}. The test build failed in hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5317/testReport/> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5317/artifact/patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5317/console> This message is automatically generated.
72. ...build started pulling in hadoop 3.0 snapshot from repo, and stopped working.
73. {color:red}-1 overall{color}. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12673478/YARN-913-021.patch> against trunk revision 9b8a35a. {color:green}+1 @author{color}. The patch does not contain any @author tags. {color:green}+1 tests included{color}. The patch appears to include 37 new or modified test files. {color:red}-1 javac{color}. The applied patch generated 1268 javac compiler warnings (more than the trunk's current 1267 warnings). {color:green}+1 javadoc{color}. There were no new javadoc warning messages. {color:green}+1 eclipse:eclipse{color}. The patch built with eclipse:eclipse. {color:green}+1 findbugs{color}. The patch does not introduce any new Findbugs (version 2.0.3) warnings. {color:green}+1 release audit{color}. The applied patch does not increase the total number of release audit warnings. {color:green}+1 core tests{color}. The patch passed unit tests in hadoop-yarn-project/hadoop-yarn/hadoop-yarn-applications/hadoop-yarn-applications-distributedshell hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-common hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-tests. {color:green}+1 contrib tests{color}. The patch passed contrib unit tests. Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5319/testReport/> Javac warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5319/artifact/patchprocess/diffJavacWarnings.txt> Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5319/console> This message is automatically generated.
74. +1 for the core module. Someone needs to RM integration.
75. committed the core {{hadoop-yarn-registry}} module and associated diffs; YARN-2562 contains the exact patch applied to branch-2.6 & branch-2; trunk gets the same with the POM updated to version 3.0.0
76. FAILURE: Integrated in Hadoop-trunk-Commit #6214 (See [<https://builds.apache.org/job/Hadoop-trunk-Commit/6214/>]) YARN-913 service registry: YARN-2652 add hadoop-yarn-registry package under hadoop-yarn (stevel: rev 6a326711aa27e84fd4c53937afc5c41a746ec65a) \* hadoop-yarn-project/hadoop-yarn/dev-support/findbugs-exclude.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/MicroZookeeperServiceKeys.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistrySecurity.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/KerberosConfiguration.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/BindFlags.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/DeleteCompletionCallback.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryOperationsFactory.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestRegistryOperationUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/ZookeeperConfigOptions.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/RecordCuratorService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ServiceRecordHeader.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestRegistrySecurityHelper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/AbstractSecureRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/InvalidRecordException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryConstants.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/tla/yarnregistry.tla \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/BindingInformation.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoPathPermissionsException.java \* .gitignore \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryBindingSource.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/RegistryTestHelper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryOperationsService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureLogins.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-

registry/src/main/java/org/apache/hadoop/registry/client/types/ProtocolTypes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/RegistryOperationsClient.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureRMRegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/TestMicroZookeeperService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestRegistryPathUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/RegistryAdminService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/RegistryIOException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoChildrenForEphemeralsException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/AbstractRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/MicroZookeeperService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/resources/.keep \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/AbstractZKRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureRegistry.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/package-info.java \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/apt/index.apt.vm \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/Endpoint.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoRecordException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ServiceRecord.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/yarn/PersistencePolicies.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryInternalConstants.java \* hadoop-project/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/AddressTypes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/ZKPathDumper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/integration/TestRegistryRMOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/InvalidPathnameException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/RegistryPathStatus.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/CuratorService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/resources/log4j.properties \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/CuratorEventCatcher.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/AddingCompositeService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryPathUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/AuthenticationFailedException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/operations/TestRegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/integration/TestYarnPolicySelector.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/yarn/YarnRegistryAttributes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryTypeUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/SelectByYarnPersistence.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/RMRegistryOperationsService.java \* hadoop-yarn-project/hadoop-yarn/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/JsonSerDeser.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md

77. SUCCESS: Integrated in Hadoop-Yarn-trunk #706 (See [https://builds.apache.org/job/Hadoop-Yarn-trunk/706/]) YARN-913 service registry: YARN-2652 add hadoop-yarn-registry package under hadoop-yarn (stevel: rev 6a326711aa27e84fd4c53937afc5c41a746ec65a) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/apt/index.apt.vm \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoRecordException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/ZKPathDumper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/tla/yarnregistry.tla \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryOperationsService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/Endpoint.java \* hadoop-yarn-project/hadoop-yarn/pom.xml \* hadoop-yarn-project/hadoop-yarn/dev-support/findbugs-exclude.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/DeleteCompletionCallback.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestRegistryPathUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/MicroZookeeperService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/integration/TestRegistryRMOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistrySecurity.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestRegistrySecurityHelper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestMarshalling.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/CuratorEventCatcher.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/ZookeeperConfigOptions.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/yarn/PersistencePolicies.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryOperationsFactory.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/AddressTypes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureRMRegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/AbstractSecureRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/resources/.keep \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/TestMicroZookeeperService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/TestCuratorService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ProtocolTypes.java \* .gitignore \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/yarn/YarnRegistryAttributes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-

registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureLogins.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/RegistryIOException.java \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/resources/log4j.properties \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/RMRegistryOperationsService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryInternalConstants.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureRegistry.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/AddingCompositeService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/MicroZookeeperServiceKeys.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryTypeUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/integration/TestYarnPolicySelector.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/RegistryPathStatus.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/AbstractRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/RegistryAdminService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/operations/TestRegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoPathPermissionsException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/BindFlags.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/BindingInformation.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ServiceRecord.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/CuratorService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryPathUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/SelectByYarnPersistence.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/RegistryOperationsClient.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/AuthenticationFailedException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/package-info.java \* hadoop-project/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/RegistryTestHelper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestRegistryOperationUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/InvalidPathnameException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryConstants.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoChildrenForEphemeralsException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/InvalidRecordException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ServiceRecordHeader.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/AbstractZKRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/KerberosConfiguration.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/JsonSerDeser.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryBindingSource.java

78. FAILURE: Integrated in Hadoop-Hdfs-trunk #1896 (See [https://builds.apache.org/job/Hadoop-Hdfs-trunk/1896/]) YARN-913 service registry: YARN-2652 add hadoop-yarn-registry package under hadoop-yarn (stevell: rev 6a326711aa27e84fd4c53937afc5c41a746ec65a) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/TestCuratorService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/RegistryPathStatus.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/RegistryAdminService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/integration/TestYarnPolicySelector.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestRegistrySecurityHelper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoRecordException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/yarn/PersistencePolicies.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/apt/index.apt.vm \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/resources/log4j.properties \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/TestMicroZookeeperService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/RegistryTestHelper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryOperationsFactory.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/CuratorEventCatcher.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/resources/keep \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/AddingCompositeService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/BindingInformation.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/yarn/YarnRegistryAttributes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/RMRegistryOperationsService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/JsonSerDeser.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistrySecurity.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/integration/TestRegistryRMOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/InvalidPathnameException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryPathUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/KerberosConfiguration.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/ZookeeperConfigOptions.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/Endpoint.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/AbstractSecureRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryInternalConstants.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-

registry/src/main/java/org/apache/hadoop/registry/client/types/ServiceRecord.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/ZKPathDumper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ProtocolTypes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/BindFlags.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/SelectByYarnPersistence.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoChildrenForEphemeralsException.java \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/tla/yarnregistry.tla \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml \* hadoop-yarn-project/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/AddressTypes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/package-info.java \* hadoop-yarn-project/hadoop-yarn/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestRegistryPathUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryOperationsService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryBindingSource.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/InvalidRecordException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestRegistryOperationUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/AuthenticationFailedException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/AbstractZKRegistryTest.java \* hadoop-yarn-project/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ServiceRecordHeader.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoPathPermissionsException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/RegistryOperationsClient.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureRegistry.java \* hadoop-project/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestMarshalling.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/package-info.java \* .gitignore \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/CuratorService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/RegistryIOException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/AbstractRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/operations/TestRegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryConstants.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureLogins.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/MicroZookeeperService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryTypeUtils.java \* hadoop-yarn-project/hadoop-yarn/dev-support/findbugs-exclude.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureRMRegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/DeleteCompletionCallback.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/MicroZookeeperServiceKeys.java

79. FAILURE: Integrated in Hadoop-Mapreduce-trunk #1921 (See [https://builds.apache.org/job/Hadoop-Mapreduce-trunk/1921/]) YARN-913 service registry: YARN-2652 add hadoop-yarn-registry package under hadoop-yarn (stevel: rev 6a326711aa27e84fd4c53937afc5c41a746ec65a) \* hadoop-yarn-project/hadoop-yarn/dev-support/findbugs-exclude.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/JsonSerDeser.java \* hadoop-yarn-project/hadoop-yarn/pom.xml \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/operations/TestRegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/AbstractRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/InvalidPathnameException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryConstants.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/AbstractSecureRegistryTest.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/RegistryOperationsClient.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/AuthenticationFailedException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureRegistry.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/KerberosConfiguration.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoChildrenForEphemeralsException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/MicroZookeeperService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/CuratorEventCatcher.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureLogins.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/BindFlags.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/Endpoint.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/MicroZookeeperServiceKeys.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryOperationsService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/integration/TestRegistryRMOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoRecordException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/RegistryPathStatus.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryTypeUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/ZookeeperConfigOptions.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/resources/log4j.properties \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestRegistryPathUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/resources/.keep \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryBindingSource.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-



- registry/src/main/java/org/apache/hadoop/registry/client/binding/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/RegistryIOException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/InvalidRecordException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/RegistryAdminService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/TestMicroZookeeperService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/exceptions/NoPathPermissionsException.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/AddingCompositeService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/RMRegistryOperationsService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/AddressTypes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistryInternalConstants.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/SecureYarnPersistencePolicies.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestSecureRMRegistryOperations.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/secure/TestRegistrySecurityHelper.java \* .gitignore \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/integration/SelectByYarnPersistence.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/RegistrySecurity.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/services/DeleteCompletionCallback.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ProtocolTypes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/tla/yarnregistry.tla \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/RegistryTestHelper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/apt/index.apt.vm \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/CuratorService.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/server/package-info.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/binding/RegistryPathUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/api/RegistryOperationsFactory.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/BindingInformation.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ServiceRecord.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/integration/TestYarnPolicySelector.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/impl/zk/ZKPathDumper.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestMarshallingHeader.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/ServiceRecordHeader.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/client/types/yarn/YarnRegistryAttributes.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/binding/TestRegistryOperationUtils.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/AbstractZKRegistryTest.java \* hadoop-yarn-project/pom.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/client/impl/TestCuratorService.java
80. the registry uses curator; we need one that is compatible with Hadoop's own dependencies, hence HADOOP-11102
81. FAILURE: Integrated in Hadoop-trunk-Commit #7069 (See [https://builds.apache.org/job/Hadoop-trunk-Commit/7069/]) YARN-2616 [YARN-913] Add CLI client to the registry to list, view and manipulate entries. (Akshay Radia via stever) (stever: rev 362565cf5a8cbc1e7e66847649c29666d79f6938) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/cli/TestRegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/CHANGES.txt
82. FAILURE: Integrated in Hadoop-trunk-Commit #7070 (See [https://builds.apache.org/job/Hadoop-trunk-Commit/7070/]) YARN-2683. [YARN-913] registry config options: document and move to core-default. (stever) (stever: rev c3da2db48fd18c41096fe5d6d4650978fb31ae24) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-configuration.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-common-project/hadoop-common/src/main/resources/core-default.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml \* hadoop-yarn-project/CHANGES.txt
83. FAILURE: Integrated in Hadoop-Yarn-trunk-Java8 #102 (See [https://builds.apache.org/job/Hadoop-Yarn-trunk-Java8/102/]) YARN-2616 [YARN-913] Add CLI client to the registry to list, view and manipulate entries. (Akshay Radia via stever) (stever: rev 362565cf5a8cbc1e7e66847649c29666d79f6938) \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/cli/TestRegistryCli.java YARN-2683. [YARN-913] registry config options: document and move to core-default. (stever) (stever: rev c3da2db48fd18c41096fe5d6d4650978fb31ae24) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-configuration.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml \* hadoop-common-project/hadoop-common/src/main/resources/core-default.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md
84. SUCCESS: Integrated in Hadoop-Yarn-trunk #836 (See [https://builds.apache.org/job/Hadoop-Yarn-trunk/836/]) YARN-2616 [YARN-913] Add CLI client to the registry to list, view and manipulate entries. (Akshay Radia via stever) (stever: rev 362565cf5a8cbc1e7e66847649c29666d79f6938) \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/cli/TestRegistryCli.java YARN-2683. [YARN-913] registry config options: document and move to core-default. (stever) (stever: rev c3da2db48fd18c41096fe5d6d4650978fb31ae24) \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-common-project/hadoop-common/src/main/resources/core-default.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-configuration.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml
85. FAILURE: Integrated in Hadoop-Hdfs-trunk #2034 (See [https://builds.apache.org/job/Hadoop-Hdfs-trunk/2034/]) YARN-2616 [YARN-913] Add CLI client to the registry to list, view and manipulate entries. (Akshay Radia via stever) (stever: rev 362565cf5a8cbc1e7e66847649c29666d79f6938) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/cli/TestRegistryCli.java \* hadoop-yarn-project/CHANGES.txt YARN-2683. [YARN-913] registry config options: document and move to core-default. (stever) (stever: rev c3da2db48fd18c41096fe5d6d4650978fb31ae24) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-

- site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml \* hadoop-common-project/hadoop-common/src/main/resources/core-default.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-configuration.md
86. FAILURE: Integrated in Hadoop-Mapreduce-trunk-Java8 #103 (See [https://builds.apache.org/job/Hadoop-Mapreduce-trunk-Java8/103/]) YARN-2616 [YARN-913] Add CLI client to the registry to list, view and manipulate entries. (Akshay Radia via stevel) (stevel: rev 362565cf5a8cbc1e7e66847649c29666d79f6938) \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/cli/TestRegistryCli.java YARN-2683. [YARN-913] registry config options: document and move to core-default. (stevel) (stevel: rev c3da2db48fd18c41096fe5d6d4650978fb31ae24) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-configuration.md \* hadoop-common-project/hadoop-common/src/main/resources/core-default.xml \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml
87. FAILURE: Integrated in Hadoop-Mapreduce-trunk #2053 (See [https://builds.apache.org/job/Hadoop-Mapreduce-trunk/2053/]) YARN-2616 [YARN-913] Add CLI client to the registry to list, view and manipulate entries. (Akshay Radia via stevel) (stevel: rev 362565cf5a8cbc1e7e66847649c29666d79f6938) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/cli/TestRegistryCli.java \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/CHANGES.txt YARN-2683. [YARN-913] registry config options: document and move to core-default. (stevel) (stevel: rev c3da2db48fd18c41096fe5d6d4650978fb31ae24) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-common-project/hadoop-common/src/main/resources/core-default.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-configuration.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/CHANGES.txt
88. SUCCESS: Integrated in Hadoop-Hdfs-trunk-Java8 #97 (See [https://builds.apache.org/job/Hadoop-Hdfs-trunk-Java8/97/]) YARN-2616 [YARN-913] Add CLI client to the registry to list, view and manipulate entries. (Akshay Radia via stevel) (stevel: rev 362565cf5a8cbc1e7e66847649c29666d79f6938) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/main/java/org/apache/hadoop/registry/cli/RegistryCli.java \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-registry/src/test/java/org/apache/hadoop/registry/cli/TestRegistryCli.java YARN-2683. [YARN-913] registry config options: document and move to core-default. (stevel) (stevel: rev c3da2db48fd18c41096fe5d6d4650978fb31ae24) \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/using-the-yarn-service-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-security.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/yarn-registry.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-common/src/main/resources/yarn-default.xml \* hadoop-yarn-project/CHANGES.txt \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/registry-configuration.md \* hadoop-yarn-project/hadoop-yarn/hadoop-yarn-site/src/site/markdown/registry/index.md \* hadoop-common-project/hadoop-common/src/main/resources/core-default.xml