

**git\_comments:**

1. By coprocessor path wrong region will open failed, MetricsRegionWrapperImpl is already init and not close, add region close when open failed
2. there are still two task not cancel, can not cause to memory leak
3. make sure region is success close when coprocessor wrong region open failed
4. skip close

**git\_commits:**

1. **summary:** HBASE-22169 Open region failed cause memory leak  
**message:** HBASE-22169 Open region failed cause memory leak Signed-off-by: stack <stack@apache.org>

**github\_issues:****github\_issues\_comments:****github\_pulls:****github\_pulls\_comments:****github\_pulls\_reviews:****jira\_issues:**

1. **summary:** Open region failed cause memory leak  
**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}
2. **summary:** Open region failed cause memory leak  
**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at

org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

3. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

4. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at

- org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}
5. **summary:** Open region failed cause memory leak  
**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocesorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocesorHost.loadTableCoprocesors(RegionCoprocesorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocesorHost.<init>(RegionCoprocesorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}
6. **summary:** Open region failed cause memory leak  
**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocesorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocesorHost.loadTableCoprocesors(RegionCoprocesorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocesorHost.<init>(RegionCoprocesorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}
- label:** code-design
7. **summary:** Open region failed cause memory leak  
**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocesorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at

org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

8. **summary:** Open region failed cause memory leak  
**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

9. **summary:** Open region failed cause memory leak  
**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>

(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at java.lang.reflect.Constructor.newInstance(Constructor.java:423) at org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

10. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException: java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init> (DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init> (HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at java.lang.reflect.Constructor.newInstance(Constructor.java:423) at org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

11. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException: java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init> (DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init> (HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at java.lang.reflect.Constructor.newInstance(Constructor.java:423) at org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

12. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

13. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

14. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at

org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

15. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

16. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSClient.<init>  
(DFSClient.java:678) at org.apache.hadoop.hdfs.DFSClient.<init>(DFSClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at

org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

17. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocesorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocesorHost.loadTableCoprocesors(RegionCoprocesorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocesorHost.<init>(RegionCoprocesorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

18. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocesorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocesorHost.loadTableCoprocesors(RegionCoprocesorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocesorHost.<init>(RegionCoprocesorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

19. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocesorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at



org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

20. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

21. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at

org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}  
**label:** test

22. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

23. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at

org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

**label:** code-design

24. **summary:** Open region failed cause memory leak

**description:** In some cases (for example, coprocessor path is wrong) region open failed, MetricsRegionWrapperImpl is already init and not close, cause memory leak; {code:java} 2019-02-21 15:41:32,929 ERROR [RS\_OPEN\_REGION-hb-2zedsc3fxjn12dl6u-005:16020-7] regionserver.RegionCoprocessorHost(362): Failed to load coprocessor  
org.apache.kylin.storage.hbase.cube.v2.coprocessor.endpoint.CubeVisitService java.lang.IllegalArgumentException:  
java.net.UnknownHostException: emr-cluster at org.apache.hadoop.security.SecurityUtil.buildTokenService(SecurityUtil.java:378) at  
org.apache.hadoop.hdfs.NameNodeProxies.createNonHAProxy(NameNodeProxies.java:310) at  
org.apache.hadoop.hdfs.NameNodeProxies.createProxy(NameNodeProxies.java:176) at org.apache.hadoop.hdfs.DFSCClient.<init>  
(DFSCClient.java:678) at org.apache.hadoop.hdfs.DFSCClient.<init>(DFSCClient.java:619) at  
org.apache.hadoop.hdfs.DistributedFileSystem.initialize(DistributedFileSystem.java:149) at  
org.apache.hadoop.fs.FileSystem.createFileSystem(FileSystem.java:2653) at  
org.apache.hadoop.fs.FileSystem.access\$200(FileSystem.java:92) at  
org.apache.hadoop.fs.FileSystem\$Cache.getInternal(FileSystem.java:2687) at  
org.apache.hadoop.fs.FileSystem\$Cache.get(FileSystem.java:2669) at org.apache.hadoop.fs.FileSystem.get(FileSystem.java:371) at  
org.apache.hadoop.fs.Path.getFileSystem(Path.java:295) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.init(CoprocessorClassLoader.java:165) at  
org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:194) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:352) at  
org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:240) at  
org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:749) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
(HRegion.java:657) at sun.reflect.GeneratedConstructorAccessor13.newInstance(Unknown Source) at  
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:6727) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7037) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7009) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6965) at  
org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:6916) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.openRegion(OpenRegionHandler.java:362) at  
org.apache.hadoop.hbase.regionserver.handler.OpenRegionHandler.process(OpenRegionHandler.java:129){code}

**label:** code-design

**jira\_issues\_comments:**

1. | (x) \*{color:red}-1 overall{color}\* | \\\| \| Vote \| Subsystem \| Runtime \| Comment \| | {color:blue}0{color} | {color:blue} reexec {color} | {color:blue} 0m 25s{color} | {color:blue} Docker mode activated. {color} | \| \| \| \| {color:brown} Prechecks {color} | \| {color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-patterns. {color} | \| {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch does not contain any @author tags. {color} | \| {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} | {color:green} The patch appears to include 1 new or modified test files. {color} | \| \| \| \| {color:brown} branch-2.2 Compile Tests {color} | \| {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 3m 51s{color} | {color:green} branch-2.2 passed {color} | \| {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 48s{color} | {color:green} branch-2.2 passed {color} | \| {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 12s{color} | {color:green} branch-2.2 passed {color} | \| {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 3m 56s{color} | {color:green} branch has no errors when building our shaded downstream artifacts. {color} | \| {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 8s{color} | {color:green} branch-2.2 passed {color} | \| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 30s{color} | {color:green} branch-2.2 passed {color} | \| \| \| \| {color:brown} Patch Compile Tests {color} | \| {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 3m 48s{color} | {color:green} the patch passed {color} | \| {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 49s{color} | {color:green} the patch passed {color} | \| {color:green}+1{color} | {color:green} javac {color} | {color:green} 0m 49s{color} | {color:green} the patch passed {color} | \| {color:red}-1{color} | {color:red} checkstyle {color} | {color:red} 1m 9s{color} | {color:red} hbase-server: The patch generated 1 new + 242 unchanged - 0 fixed = 243 total (was 242) {color} | \| {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace issues. {color} | \| {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 3m 57s{color} | {color:green} patch has no errors when building our shaded downstream artifacts. {color} | \| {color:green}+1{color} | {color:green} hadoopcheck {color} | {color:green} 7m 56s{color} | {color:green} Patch does not cause any errors with Hadoop 2.7.4 or 3.0.0. {color} | \| {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 18s{color} | {color:green} the patch passed {color} | \| {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 30s{color} | {color:green} the patch passed {color} | \| \| \| \| {color:brown} Other Tests {color} | \| {color:red}-1{color} | {color:red} unit {color} | {color:red} 134m 22s{color} | {color:red} hbase-server in the patch failed. {color} | \| {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 18s{color} | {color:green} The patch does not generate ASF License warnings. {color} | \| {color:black}{color} | {color:black} {color} | {color:black} 168m 44s{color} | {color:black} {color} | \\\| \| Reason \| Tests \| \| Failed junit tests | hadoop.hbase.regionserver.TestOpenSeqNumUnexpectedIncrease | \\\| \| Subsystem \| Report/Notes \| \| Docker | Client=17.05.0-ce Server=17.05.0-ce base: <https://builds.apache.org/job/PreCommit-HBASE-Build/13/artifact/patchprocess/Dockerfile> | | JIRA Issue | HBASE-22169 | | JIRA Patch URL | <https://issues.apache.org/jira/secure/attachment/12965146/HBASE-22169-branch-2.2-v1.patch> | | Optional Tests | dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile | | uname | Linux 383ccd2faba2 4.4.0-138-generic #164-Ubuntu SMP Oct 2 17:16:02 UTC 2018 x86\_64 GNU/Linux | | Build tool | maven | | Personality | dev-support/hbase-personality.sh | | git revision | branch-2.2 / 4c5eec5bb3 | | maven | version: Apache Maven 3.5.4 (1edded0938998edf8bf061f1cebf3cfdeccf443fe; 2018-06-17T18:33:14Z) | | Default Java | 1.8.0\_181 | | findbugs | v3.1.11 | | checkstyle | <https://builds.apache.org/job/PreCommit-HBASE-Build/13/artifact/patchprocess/diff-checkstyle-hbase-server.txt> | | unit |

<https://builds.apache.org/job/PreCommit-HBASE-Build/13/artifact/patchprocess/patch-unit-hbase-server.txt> || Test Results |  
<https://builds.apache.org/job/PreCommit-HBASE-Build/13/testReport/> || Max. process+thread count | 4994 (vs. ulimit of 10000) ||  
modules | C: hbase-server U: hbase-server || Console output | <https://builds.apache.org/job/PreCommit-HBASE-Build/13/console> ||  
Powered by | Apache Yetus 0.9.0 <http://yetus.apache.org> | This message was automatically generated.

2. | (x) \*{color:red}-1 overall{color}\* | \\\| Vote || Subsystem || Runtime || Comment || {color:blue}0{color} | {color:blue} reexec  
{color} | {color:blue} 0m 47s{color} | {color:blue} Docker mode activated. {color} | || || || {color:brown} Prechecks {color} ||  
{color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-  
patterns. {color} | | {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch  
does not contain any @author tags. {color} | | {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} |  
{color:green} The patch appears to include 1 new or modified test files. {color} | || || || {color:brown} master Compile Tests {color} ||  
| {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 13s{color} | {color:green} master passed {color} | |  
{color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 51s{color} | {color:green} master passed {color} | |  
{color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 12s{color} | {color:green} master passed {color} | |  
{color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 31s{color} | {color:green} branch has no errors when  
building our shaded downstream artifacts. {color} | | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m  
53s{color} | {color:green} master passed {color} | | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m  
34s{color} | {color:green} master passed {color} | || || || {color:brown} Patch Compile Tests {color} || {color:green}+1{color} |  
{color:green} mvninstall {color} | {color:green} 4m 24s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} |  
{color:green} compile {color} | {color:green} 1m 4s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} |  
{color:green} javac {color} | {color:green} 1m 4s{color} | {color:green} the patch passed {color} | | {color:red}-1{color} |  
{color:red} checkstyle {color} | {color:red} 1m 13s{color} | {color:red} hbase-server: The patch generated 1 new + 235 unchanged -  
0 fixed = 236 total (was 235) {color} | | {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} |  
{color:green} The patch has no whitespace issues. {color} | | {color:green}+1{color} | {color:green} shadedjars {color} |  
{color:green} 4m 39s{color} | {color:green} patch has no errors when building our shaded downstream artifacts. {color} | |  
{color:green}+1{color} | {color:green} hadoopcheck {color} | {color:green} 9m 22s{color} | {color:green} Patch does not cause any  
errors with Hadoop 2.7.4 or 3.0.0. {color} | | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 55s{color} |  
{color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 32s{color} |  
{color:green} the patch passed {color} | || || || {color:brown} Other Tests {color} || {color:red}-1{color} | {color:red} unit {color} |  
{color:red}358m 29s{color} | {color:red} hbase-server in the patch failed. {color} | | {color:green}+1{color} | {color:green}  
asflicense {color} | {color:green} 0m 25s{color} | {color:green} The patch does not generate ASF License warnings. {color} | |  
{color:black}{color} | {color:black} {color} | {color:black}398m 37s{color} | {color:black} {color} | \\\| Reason || Tests || Failed  
junit tests | hadoop.hbase.client.TestSnapshotTemporaryDirectory || | hadoop.hbase.client.TestAsyncTableAdminApi || |  
hadoop.hbase.master.TestAssignmentManagerMetrics || | hadoop.hbase.tool.TestSecureLoadIncrementalHFiles || |  
hadoop.hbase.namespace.TestNamespaceAuditor || | hadoop.hbase.client.TestFromClientSide3 || |  
hadoop.hbase.master.procedure.TestSCPWithReplicasWithoutZKCoordinated || | hadoop.hbase.replication.TestReplicationEndpoint || |  
| hadoop.hbase.client.TestSnapshotTemporaryDirectoryWithRegionReplicas || |  
hadoop.hbase.replication.TestReplicationKillSlaveRSWithSeparateOldWALs || |  
hadoop.hbase.replication.multiwal.TestReplicationSyncUpToolWithMultipleAsyncWAL || |  
hadoop.hbase.replication.multiwal.TestReplicationSyncUpToolWithMultipleWAL || |  
hadoop.hbase.replication.TestNamespaceReplication || | hadoop.hbase.replication.TestReplicationSmallTestsSync || |  
hadoop.hbase.replication.TestReplicationSyncUpTool || | hadoop.hbase.client.TestConnectionImplementation || |  
hadoop.hbase.client.TestSnapshotDFSTemporaryDirectory || | hadoop.hbase.client.replication.TestReplicationAdminWithClusters || |  
hadoop.hbase.regionserver.TestOpenSeqNumUnexpectedIncrease || | hadoop.hbase.tool.TestLoadIncrementalHFiles || |  
hadoop.hbase.quotas.TestSpaceQuotas || | hadoop.hbase.master.procedure.TestSCPWithReplicas || |  
hadoop.hbase.client.TestCloneSnapshotFromClientNormal || | hadoop.hbase.replication.TestSyncReplicationActive || |  
hadoop.hbase.util.TestFromClientSide3WoUnsafe || | hadoop.hbase.client.TestAdmin1 || |  
hadoop.hbase.replication.TestReplicationSmallTests || | hadoop.hbase.client.TestFromClientSide || |  
hadoop.hbase.client.TestFromClientSideWithCoprocessor | \\\| Subsystem || Report/Notes || Docker | Client=17.05.0-ce  
Server=17.05.0-ce base: <https://builds.apache.org/job/PreCommit-HBASE-Build/15/artifact/patchprocess/Dockerfile> || JIRA Issue |  
HBASE-22169 | JIRA Patch URL | <https://issues.apache.org/jira/secure/attachment/12964957/HBASE-22169-master-v1.patch> ||  
Optional Tests | dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile || uname |  
Linux 5cd6833e8c77 4.4.0-138-generic #164-Ubuntu SMP Tue Oct 2 17:16:02 UTC 2018 x86\_64 GNU/Linux || Build tool | maven ||  
Personality | dev-support/hbase-personality.sh || git revision | master / b2587c874b || maven | version: Apache Maven 3.5.4  
1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z || Default Java | 1.8.0\_181 || findbugs | v3.1.11 || checkstyle  
| <https://builds.apache.org/job/PreCommit-HBASE-Build/15/artifact/patchprocess/diff-checkstyle-hbase-server.txt> || unit |  
<https://builds.apache.org/job/PreCommit-HBASE-Build/15/artifact/patchprocess/patch-unit-hbase-server.txt> || Test Results |  
<https://builds.apache.org/job/PreCommit-HBASE-Build/15/testReport/> || Max. process+thread count | 5639 (vs. ulimit of 10000) ||  
modules | C: hbase-server U: hbase-server || Console output | <https://builds.apache.org/job/PreCommit-HBASE-Build/15/console> ||  
Powered by | Apache Yetus 0.9.0 <http://yetus.apache.org> | This message was automatically generated.

3. | (x) \*{color:red}-1 overall{color}\* | \\\| Vote || Subsystem || Runtime || Comment || {color:blue}0{color} | {color:blue} reexec  
{color} | {color:blue} 0m 52s{color} | {color:blue} Docker mode activated. {color} | || || || {color:brown} Prechecks {color} ||  
{color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-  
patterns. {color} | | {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch  
does not contain any @author tags. {color} | | {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} |  
{color:green} The patch appears to include 2 new or modified test files. {color} | || || || {color:brown} master Compile Tests {color} ||  
| {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 45s{color} | {color:green} master passed {color} | |  
{color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 50s{color} | {color:green} master passed {color} | |  
{color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 13s{color} | {color:green} master passed {color} | |  
{color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 31s{color} | {color:green} branch has no errors when  
building our shaded downstream artifacts. {color} | | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m  
21s{color} | {color:green} master passed {color} | | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m  
33s{color} | {color:green} master passed {color} | || || || {color:brown} Patch Compile Tests {color} || {color:green}+1{color} |  
{color:green} mvninstall {color} | {color:green} 4m 22s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} |  
{color:green} compile {color} | {color:green} 0m 52s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} |

{color:green} javac {color} | {color:green} 0m 52s{color} | {color:green} the patch passed {color} | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 13s{color} | {color:green} the patch passed {color} | {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace issues. {color} | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 29s{color} | {color:green} patch has no errors when building our shaded downstream artifacts. {color} | {color:green}+1{color} | {color:green} hadoopcheck {color} | {color:green} 8m 57s{color} | {color:green} Patch does not cause any errors with Hadoop 2.7.4 or 3.0.0. {color} | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 35s{color} | {color:green} the patch passed {color} | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 33s{color} | {color:green} the patch passed {color} | {color:brown} Other Tests {color} || {color:red}-1{color} | {color:red} unit {color} | {color:red}305m 49s{color} | {color:red} hbase-server in the patch failed. {color} | {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 36s{color} | {color:green} The patch does not generate ASF License warnings. {color} | {color:black}{color} | {color:black}{color} | {color:black}345m 5s{color} | {color:black}{color} | \\ \\ \\ Reason || Tests || Failed junit tests |  
hadoop.hbase.client.TestSnapshotTemporaryDirectoryWithRegionReplicas ||| hadoop.hbase.client.TestFromClientSide3 |||  
hadoop.hbase.tool.TestLoadIncrementalHFiles ||| hadoop.hbase.master.TestAssignmentManagerMetrics |||  
hadoop.hbase.client.TestAdmin1 ||| hadoop.hbase.client.replication.TestReplicationAdminWithClusters |||  
hadoop.hbase.master.procedure.TestSCPWithReplicasWithoutZKCoordinated ||| hadoop.hbase.quotas.TestSpaceQuotas |||  
hadoop.hbase.client.TestFromClientSide ||| hadoop.hbase.client.TestFromClientSideWithCoprocessor |||  
hadoop.hbase.util.TestFromClientSide3WoUnsafe ||| hadoop.hbase.client.TestSnapshotDFSTemporaryDirectory |||  
hadoop.hbase.tool.TestSecureLoadIncrementalHFiles |||  
hadoop.hbase.replication.multiwal.TestReplicationSyncUpToolWithMultipleAsyncWAL |||  
hadoop.hbase.client.TestSnapshotTemporaryDirectory ||| hadoop.hbase.master.procedure.TestSCPWithReplicas || \\ \\ \\ Subsystem || Report/Notes || Docker | Client=17.05.0-ce Server=17.05.0-ce base: https://builds.apache.org/job/PreCommit-HBASE-Build/40/artifact/patchprocess/Dockerfile || JIRA Issue | HBASE-22169 || JIRA Patch URL |  
https://issues.apache.org/jira/secure/attachment/12965275/HBASE-22169-master-v2.patch || Optional Tests | dupname asflicense  
javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile || uname | Linux 0199ef358dd0 4.4.0-138-generic  
#164-Ubuntu SMP Tue Oct 2 17:16:02 UTC 2018 x86\_64 GNU/Linux || Build tool | maven || Personality | dev-support/hbase-personality.sh || git revision | master / 89ce5d17e4 || maven | version: Apache Maven 3.5.4  
(1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z) || Default Java | 1.8.0\_181 || findbugs | v3.1.11 || unit |  
https://builds.apache.org/job/PreCommit-HBASE-Build/40/artifact/patchprocess/patch-unit-hbase-server.txt || Test Results |  
https://builds.apache.org/job/PreCommit-HBASE-Build/40/testReport/ || Max. process+thread count | 5122 (vs. ulimit of 10000) ||  
modules | C: hbase-server U: hbase-server || Console output | https://builds.apache.org/job/PreCommit-HBASE-Build/40/console ||  
Powered by | Apache Yetus 0.9.0 http://yetus.apache.org | This message was automatically generated.

4. | (x) \*{color:red}-1 overall{color}\* | \\ \\ \\ Vote || Subsystem || Runtime || Comment || {color:blue}0{color} | {color:blue} reexec {color} | {color:blue} 0m 49s{color} | {color:blue} Docker mode activated. {color} | || || || {color:brown} Prechecks {color} || {color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-patterns. {color} | {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch does not contain any @author tags. {color} | {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} | {color:green} The patch appears to include 2 new or modified test files. {color} | || || || {color:brown} master Compile Tests {color} || {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 33s{color} | {color:green} master passed {color} | {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 53s{color} | {color:green} master passed {color} | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 13s{color} | {color:green} master passed {color} | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 40s{color} | {color:green} branch has no errors when building our shaded downstream artifacts. {color} | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 3m 30s{color} | {color:green} master passed {color} | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 33s{color} | {color:green} master passed {color} | || || || {color:brown} Patch Compile Tests {color} || {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 3m 57s{color} | {color:green} the patch passed {color} | {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 50s{color} | {color:green} the patch passed {color} | {color:green}+1{color} | {color:green} javac {color} | {color:green} 0m 50s{color} | {color:green} the patch passed {color} | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 8s{color} | {color:green} the patch passed {color} | {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace issues. {color} | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 17s{color} | {color:green} patch has no errors when building our shaded downstream artifacts. {color} | {color:green}+1{color} | {color:green} hadoopcheck {color} | {color:green} 9m 21s{color} | {color:green} Patch does not cause any errors with Hadoop 2.7.4 or 3.0.0. {color} | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 3m 32s{color} | {color:green} the patch passed {color} | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 31s{color} | {color:green} the patch passed {color} | || || || {color:brown} Other Tests {color} || {color:red}-1{color} | {color:red} unit {color} | {color:red}297m 1s{color} | {color:red} hbase-server in the patch failed. {color} | {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 32s{color} | {color:green} The patch does not generate ASF License warnings. {color} | {color:black}{color} | {color:black}{color} | {color:black}338m 2s{color} | {color:black}{color} | \\ \\ \\ Reason || Tests || Failed junit tests | hadoop.hbase.client.TestFromClientSideWithCoprocessor |||  
hadoop.hbase.replication.multiwal.TestReplicationSyncUpToolWithMultipleWAL |||  
hadoop.hbase.master.procedure.TestSCPWithReplicasWithoutZKCoordinated ||| hadoop.hbase.client.TestFromClientSide |||  
hadoop.hbase.util.TestFromClientSide3WoUnsafe ||| hadoop.hbase.master.TestAssignmentManagerMetrics |||  
hadoop.hbase.replication.multiwal.TestReplicationSyncUpToolWithMultipleAsyncWAL |||  
hadoop.hbase.master.procedure.TestSCPWithReplicas ||| hadoop.hbase.namespace.TestNamespaceAuditor |||  
hadoop.hbase.client.TestSnapshotTemporaryDirectoryWithRegionReplicas ||| hadoop.hbase.client.TestFromClientSide3 |||  
hadoop.hbase.client.TestAdmin1 ||| hadoop.hbase.client.TestAsyncTableRegionReplicasScan || \\ \\ \\ Subsystem || Report/Notes || Docker | Client=17.05.0-ce Server=17.05.0-ce base: https://builds.apache.org/job/PreCommit-HBASE-Build/46/artifact/patchprocess/Dockerfile || JIRA Issue | HBASE-22169 || JIRA Patch URL |  
https://issues.apache.org/jira/secure/attachment/12965333/HBASE-22169-master-v2.patch || Optional Tests | dupname asflicense  
javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile || uname | Linux 68b99a1009d7 4.4.0-139-generic  
#165-Ubuntu SMP Wed Oct 24 10:58:50 UTC 2018 x86\_64 GNU/Linux || Build tool | maven || Personality | dev-support/hbase-personality.sh || git revision | master / bd2f04dd54 || maven | version: Apache Maven 3.5.4  
(1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z) || Default Java | 1.8.0\_181 || findbugs | v3.1.11 || unit |  
https://builds.apache.org/job/PreCommit-HBASE-Build/46/artifact/patchprocess/patch-unit-hbase-server.txt || Test Results |

- <https://builds.apache.org/job/PreCommit-HBASE-Build/46/testReport/> | Max. process+thread count | 5099 (vs. ulimit of 10000) | modules | C: hbase-server U: hbase-server | Console output | <https://builds.apache.org/job/PreCommit-HBASE-Build/46/console> | Powered by | Apache Yetus 0.9.0 <http://yetus.apache.org> | This message was automatically generated.
5. | (/) \*{color:green}+1 overall{color}\* | \ \ \ \ || Vote || Subsystem || Runtime || Comment || {color:blue}0{color} | {color:blue} reexec {color} | {color:blue} 0m 17s{color} | {color:blue} Docker mode activated. {color} | | | | | {color:brown} Prechecks {color} | | {color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-patterns. {color} | | {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch does not contain any @author tags. {color} | | {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} | {color:green} The patch appears to include 2 new or modified test files. {color} | | | | | {color:brown} master Compile Tests {color} | | {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 35s{color} | {color:green} master passed {color} | | {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 52s{color} | {color:green} master passed {color} | | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 19s{color} | {color:green} master passed {color} | | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 55s{color} | {color:green} branch has no errors when building our shaded downstream artifacts. {color} | | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 55s{color} | {color:green} master passed {color} | | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 32s{color} | {color:green} master passed {color} | | | | | {color:brown} Patch Compile Tests {color} | | {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 20s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 58s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} javac {color} | {color:green} 0m 58s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 22s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace issues. {color} | | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 52s{color} | {color:green} patch has no errors when building our shaded downstream artifacts. {color} | | {color:green}+1{color} | {color:green} hadoopcheck {color} | {color:green} 8m 51s{color} | {color:green} Patch does not cause any errors with Hadoop 2.7.4 or 3.0.0. {color} | | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 3m 23s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 32s{color} | {color:green} the patch passed {color} | | | | | {color:brown} Other Tests {color} | | {color:green}+1{color} | {color:green} unit {color} | {color:green} 143m 46s{color} | {color:green} hbase-server in the patch passed. {color} | | {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 23s{color} | {color:green} The patch does not generate ASF License warnings. {color} | | {color:black}{color} | {color:black} {color} | {color:black} 184m 32s{color} | {color:black} {color} | \ \ \ \ || Subsystem || Report/Notes || Docker | Client=17.05.0-ce Server=17.05.0-ce base: <https://builds.apache.org/job/PreCommit-HBASE-Build/90/artifact/patchprocess/Dockerfile> | JIRA Issue | HBASE-22169 | JIRA Patch URL | <https://issues.apache.org/jira/secure/attachment/12966009/HBASE-22169-master-v2.patch> | Optional Tests | dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile | | uname | Linux f43795acf005 4.4.0-143-generic #169~14.04.2-Ubuntu SMP Wed Feb 13 15:00:41 UTC 2019 x86\_64 GNU/Linux | | Build tool | maven | | Personality | dev-support/hbase-personality.sh | | git revision | master / cd61bcc01e | | maven | version: Apache Maven 3.5.4 (1edded0938998edf8bf061f1cebf3cfdeccf443fe; 2018-06-17T18:33:14Z) | | Default Java | 1.8.0\_181 | | findbugs | v3.1.11 | | Test Results | <https://builds.apache.org/job/PreCommit-HBASE-Build/90/testReport/> | Max. process+thread count | 5089 (vs. ulimit of 10000) | modules | C: hbase-server U: hbase-server | Console output | <https://builds.apache.org/job/PreCommit-HBASE-Build/90/console> | Powered by | Apache Yetus 0.9.0 <http://yetus.apache.org> | This message was automatically generated.
6. **body:** [~Apache9] [~allan163] This is a case happened in our production env, and cause to memery lack;How about this patch? Thanks  
**label:** code-design
7. +1 for the patch, will commit later if on objection.
8. I tried apply this. It seemed ok on branch-2.1 though it'd fail on occasion. I ran the test on master branch and it did this: {code} -----  
----- Test set: org.apache.hadoop.hbase.regionserver.TestHRegion -----  
----- Tests run: 60, Failures: 0, Errors: 1, Skipped: 0, Time elapsed: 780.443 s <<<  
FAILURE! - in org.apache.hadoop.hbase.regionserver.TestHRegion org.apache.hadoop.hbase.regionserver.TestHRegion Time elapsed: 717.422 s <<< ERROR! org.junit.runners.model.TestTimedOutException: test timed out after 780 seconds at app/org.apache.hadoop.hbase.regionserver.TestHRegion.testOpenRegionFailedMemoryLeak(TestHRegion.java:6306) {code} Let me revert the commits on branch-2.1 and branch-2.2 for now until above is resolved. I'll put up what I cherry-picked.
9. | (x) \*{color:red}-1 overall{color}\* | \ \ \ \ || Vote || Subsystem || Runtime || Comment || {color:blue}0{color} | {color:blue} reexec {color} | {color:blue} 0m 0s{color} | {color:blue} Docker mode activated. {color} | | {color:red}-1{color} | {color:red} patch {color} | | {color:red} 0m 5s{color} | {color:red} HBASE-22169 does not apply to master. Rebase required? Wrong Branch? See <https://yetus.apache.org/documentation/in-progress/precommit-patchnames> for help. {color} | | \ \ \ \ || Subsystem || Report/Notes || JIRA Issue | HBASE-22169 | JIRA Patch URL | <https://issues.apache.org/jira/secure/attachment/12969054/0001-HBASE-22169-Open-region-failed-cause-memory-leak.patch> | | Console output | <https://builds.apache.org/job/PreCommit-HBASE-Build/348/console> | | Powered by | Apache Yetus 0.9.0 <http://yetus.apache.org> | This message was automatically generated.
10. Results for branch branch-2.2 [build #267 on builds.a.o|<https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/267/>]: (x) \* {color:red}-1 overall{color}\* ---- details (if available): (x) {color:red}-1 general checks{color} -- For more information [see general report|[https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/267//General\\_Nightly\\_Build\\_Report/](https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/267//General_Nightly_Build_Report/)] (x) {color:red}-1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report|[https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/267//JDK8\\_Nightly\\_Build\\_Report\\_\(Hadoop2\)/](https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/267//JDK8_Nightly_Build_Report_(Hadoop2)/)] (/) {color:green}+1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3) report|[https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/267//JDK8\\_Nightly\\_Build\\_Report\\_\(Hadoop3\)/](https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/267//JDK8_Nightly_Build_Report_(Hadoop3)/)] (/) {color:green}+1 source release artifact{color} -- See build output for details. (x) {color:red}-1 client integration test{color} --Failed when running client tests on top of Hadoop 2. [see log for details|<https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/267/artifact/output-integration/hadoop-2.log>]. (note that this means we didn't run on Hadoop 3)
11. Results for branch branch-2.1 [build #1154 on builds.a.o|<https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1154/>]: (x) \* {color:red}-1 overall{color}\* ---- details (if available): (x) {color:red}-1 general checks{color} -- For more information [see general report|[https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1154//General\\_Nightly\\_Build\\_Report/](https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1154//General_Nightly_Build_Report/)] (/) {color:green}+1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report|[https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1154//JDK8\\_Nightly\\_Build\\_Report\\_\(Hadoop2\)/](https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1154//JDK8_Nightly_Build_Report_(Hadoop2)/)] (/) {color:green}+1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3)

- report[https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1154/JDK8\_Nightly\_Build\_Report\_(Hadoop3)/] (/) {color:green}+1 source release artifact{color} -- See build output for details. (/) {color:green}+1 client integration test{color}
12. Results for branch branch-2 [build #1899 on builds.a.o][https://builds.apache.org/job/HBase%20Nightly/job/branch-2/1899/]: (x) \* {color:red}-1 overall{color}\* ---- details (if available): (x) {color:red}-1 general checks{color} -- For more information [see general report[https://builds.apache.org/job/HBase%20Nightly/job/branch-2/1899/General\_Nightly\_Build\_Report/] (x) {color:red}-1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report[https://builds.apache.org/job/HBase%20Nightly/job/branch-2/1899/JDK8\_Nightly\_Build\_Report\_(Hadoop2)/] (x) {color:red}-1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3) report[https://builds.apache.org/job/HBase%20Nightly/job/branch-2/1899/JDK8\_Nightly\_Build\_Report\_(Hadoop3)/] (/) {color:green}+1 source release artifact{color} -- See build output for details. (x) {color:red}-1 client integration test{color} --Failed when running client tests on top of Hadoop 2. [see log for details][https://builds.apache.org/job/HBase%20Nightly/job/branch-2/1899/artifact/output-integration/hadoop-2.log]. (note that this means we didn't run on Hadoop 3)
13. [~luffy123], can you take a look?
14. Results for branch branch-2.2 [build #280 on builds.a.o][https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/280/]: (x) \* {color:red}-1 overall{color}\* ---- details (if available): (/) {color:green}+1 general checks{color} -- For more information [see general report[https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/280/General\_Nightly\_Build\_Report/] (x) {color:red}-1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report[https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/280/JDK8\_Nightly\_Build\_Report\_(Hadoop2)/] (/) {color:green}+1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3) report[https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/280/JDK8\_Nightly\_Build\_Report\_(Hadoop3)/] (/) {color:green}+1 source release artifact{color} -- See build output for details. (x) {color:red}-1 client integration test{color} --Failed when running client tests on top of Hadoop 2. [see log for details][https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/280/artifact/output-integration/hadoop-2.log]. (note that this means we didn't run on Hadoop 3)
15. (x) \*{color:red}-1 overall{color}\* | \ \ \ \ \ Vote || Subsystem || Runtime || Comment || | {color:blue}0{color} | | {color:blue} reexec {color} | | {color:blue} 0m 38s{color} | | {color:blue} Docker mode activated. {color} | | | | | {color:brown} Prechecks {color} | | {color:green}+1{color} | | {color:green} hbaseanti {color} | | {color:green} 0m 0s{color} | | {color:green} Patch does not have any anti-patterns. {color} | | {color:green}+1{color} | | {color:green} @author {color} | | {color:green} 0m 0s{color} | | {color:green} The patch does not contain any @author tags. {color} | | {color:green}+1{color} | | {color:green} test4tests {color} | | {color:green} 0m 0s{color} | | {color:green} The patch appears to include 2 new or modified test files. {color} | | | | | {color:brown} master Compile Tests {color} | | {color:green}+1{color} | | {color:green} mvninstall {color} | | {color:green} 4m 26s{color} | | {color:green} master passed {color} | | {color:green}+1{color} | | {color:green} compile {color} | | {color:green} 0m 51s{color} | | {color:green} master passed {color} | | {color:green}+1{color} | | {color:green} checkstyle {color} | | {color:green} 1m 21s{color} | | {color:green} master passed {color} | | {color:green}+1{color} | | {color:green} shadedjars {color} | | {color:green} 4m 39s{color} | | {color:green} branch has no errors when building our shaded downstream artifacts. {color} | | {color:green}+1{color} | | {color:green} findbugs {color} | | {color:green} 3m 16s{color} | | {color:green} master passed {color} | | {color:green}+1{color} | | {color:green} javadoc {color} | | {color:green} 0m 31s{color} | | {color:green} master passed {color} | | | | | {color:brown} Patch Compile Tests {color} | | {color:green}+1{color} | | {color:green} mvninstall {color} | | {color:green} 4m 19s{color} | | {color:green} the patch passed {color} | | {color:green}+1{color} | | {color:green} compile {color} | | {color:green} 0m 56s{color} | | {color:green} the patch passed {color} | | {color:green}+1{color} | | {color:green} javac {color} | | {color:green} 0m 56s{color} | | {color:green} the patch passed {color} | | {color:red}-1{color} | | {color:red} checkstyle {color} | | {color:red} 1m 19s{color} | | {color:red} hbase-server: The patch generated 1 new + 235 unchanged - 0 fixed = 236 total (was 235) {color} | | {color:green}+1{color} | | {color:green} whitespace {color} | | {color:green} 0m 0s{color} | | {color:green} The patch has no whitespace issues. {color} | | {color:green}+1{color} | | {color:green} shadedjars {color} | | {color:green} 4m 51s{color} | | {color:green} patch has no errors when building our shaded downstream artifacts. {color} | | {color:green}+1{color} | | {color:green} hadoopcheck {color} | | {color:green} 13m 25s{color} | | {color:green} Patch does not cause any errors with Hadoop 2.8.5 2.9.2 or 3.1.2. {color} | | {color:green}+1{color} | | {color:green} findbugs {color} | | {color:green} 3m 35s{color} | | {color:green} the patch passed {color} | | {color:green}+1{color} | | {color:green} javadoc {color} | | {color:green} 0m 30s{color} | | {color:green} the patch passed {color} | | | | | {color:brown} Other Tests {color} | | {color:green}+1{color} | | {color:green} unit {color} | | {color:green} 234m 8s{color} | | {color:green} hbase-server in the patch passed. {color} | | {color:green}+1{color} | | {color:green} asflicense {color} | | {color:green} 0m 25s{color} | | {color:green} The patch does not generate ASF License warnings. {color} | | {color:black}{color} | | {color:black} {color} | | {color:black} 284m 0s{color} | | {color:black} {color} | | \ \ \ \ \ Subsystem || Report/Notes || Docker | Client=17.05.0-ce Server=17.05.0-ce base: https://builds.apache.org/job/PreCommit-HBASE-Build/508/artifact/patchprocess/Dockerfile | | JIRA Issue | HBASE-22169 | | JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12971287/HBASE-22169-master-v3.patch | | Optional Tests | dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile | | uname | Linux 4a23a76fa2f6 4.4.0-143-generic #169~14.04.2-Ubuntu SMP Wed Feb 13 15:00:41 UTC 2019 x86\_64 GNU/Linux | | Build tool | maven | | Personality | dev-support/hbase-personality.sh | | git revision | master / 03436e68bc | | maven | version: Apache Maven 3.5.4 (1edded0938998edf8bf061f1ce3cfdeccf443fe; 2018-06-17T18:33:14Z) | | Default Java | 1.8.0\_181 | | findbugs | v3.1.11 | | checkstyle | https://builds.apache.org/job/PreCommit-HBASE-Build/508/artifact/patchprocess/diff-checkstyle-hbase-server.txt | | Test Results | https://builds.apache.org/job/PreCommit-HBASE-Build/508/testReport/ | | Max. process+thread count | 4822 (vs. ulimit of 10000) | | modules | C: hbase-server U: hbase-server | | Console output | https://builds.apache.org/job/PreCommit-HBASE-Build/508/console | | Powered by | Apache Yetus 0.9.0 http://yetus.apache.org | This message was automatically generated.
16. (x) \*{color:red}-1 overall{color}\* | \ \ \ \ \ Vote || Subsystem || Runtime || Comment || | {color:blue}0{color} | | {color:blue} reexec {color} | | {color:blue} 0m 18s{color} | | {color:blue} Docker mode activated. {color} | | | | | {color:brown} Prechecks {color} | | {color:green}+1{color} | | {color:green} hbaseanti {color} | | {color:green} 0m 0s{color} | | {color:green} Patch does not have any anti-patterns. {color} | | {color:green}+1{color} | | {color:green} @author {color} | | {color:green} 0m 0s{color} | | {color:green} The patch does not contain any @author tags. {color} | | {color:green}+1{color} | | {color:green} test4tests {color} | | {color:green} 0m 0s{color} | | {color:green} The patch appears to include 2 new or modified test files. {color} | | | | | {color:brown} branch-2 Compile Tests {color} | | {color:green}+1{color} | | {color:green} mvninstall {color} | | {color:green} 4m 35s{color} | | {color:green} branch-2 passed {color} | | {color:green}+1{color} | | {color:green} compile {color} | | {color:green} 0m 52s{color} | | {color:green} branch-2 passed {color} | | {color:green}+1{color} | | {color:green} checkstyle {color} | | {color:green} 1m 18s{color} | | {color:green} branch-2 passed {color} | | {color:green}+1{color} | | {color:green} shadedjars {color} | | {color:green} 4m 15s{color} | | {color:green} branch has no errors when building our shaded downstream artifacts. {color} | | {color:green}+1{color} | | {color:green} findbugs {color} | | {color:green} 2m 51s{color} | | {color:green} branch-2 passed {color} | | {color:green}+1{color} | | {color:green} javadoc {color} | | {color:green} 0m 31s{color} | | {color:green} branch-2 passed {color} | | | | | {color:brown} Patch Compile Tests {color} | |



{color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 1s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 50s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} javac {color} | {color:green} 0m 51s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 18s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace issues. {color} | | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 24s{color} | {color:green} patch has no errors when building our shaded downstream artifacts. {color} | | {color:green}+1{color} | {color:green} hadoopcheck {color} | {color:green} 12m 58s{color} | {color:green} Patch does not cause any errors with Hadoop 2.8.5 2.9.2 or 3.1.2. {color} | | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 3m 38s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 36s{color} | {color:green} the patch passed {color} | | || || || {color:brown} Other Tests {color} || | {color:red}-1{color} | {color:red} unit {color} | {color:red} 134m 58s{color} | {color:red} hbase-server in the patch failed. {color} | | {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 22s{color} | {color:green} The patch does not generate ASF License warnings. {color} | | {color:black}{color} | {color:black} {color} | {color:black} 182m 46s{color} | | {color:black} {color} | || || || Subsystem || Report/Notes || Docker | Client=17.05.0-ce Server=17.05.0-ce base: https://builds.apache.org/job/PreCommit-HBASE-Build/513/artifact/patchprocess/Dockerfile | | JIRA Issue | HBASE-22169 | | JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12971406/HBASE-22169-branch-2-v1.patch | | Optional Tests | dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile | | uname | Linux 3d05ff45ee4b 4.4.0-143-generic #169~14.04.2-Ubuntu SMP Wed Feb 13 15:00:41 UTC 2019 x86\_64 GNU/Linux | | Build tool | maven | | Personality | dev-support/hbase-personality.sh | | git revision | branch-2 / 6978d4a1cf | | maven | version: Apache Maven 3.5.4 (1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z) | | Default Java | 1.8.0\_181 | | findbugs | v3.1.11 | | unit | https://builds.apache.org/job/PreCommit-HBASE-Build/513/artifact/patchprocess/patch-unit-hbase-server.txt | | Test Results | https://builds.apache.org/job/PreCommit-HBASE-Build/513/testReport/ | | Max. process+thread count | 4861 (vs. ulimit of 10000) | | modules | C: hbase-server U: hbase-server | | Console output | https://builds.apache.org/job/PreCommit-HBASE-Build/513/console | | Powered by | Apache Yetus 0.9.0 http://yetus.apache.org | This message was automatically generated.

17. | (x) \*{color:red}-1 overall {color}\* | || || || Vote || Subsystem || Runtime || Comment || | {color:blue}0{color} | {color:blue} reexec {color} | {color:blue} 0m 39s{color} | {color:blue} Docker mode activated. {color} | || || || {color:brown} Prechecks {color} | | {color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-patterns. {color} | | {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch does not contain any @author tags. {color} | | {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} | {color:green} The patch appears to include 2 new or modified test files. {color} | || || || {color:brown} branch-2 Compile Tests {color} | | | {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 30s{color} | {color:green} branch-2 passed {color} | | {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 52s{color} | {color:green} branch-2 passed {color} | | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 21s{color} | {color:green} branch-2 passed {color} | | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 17s{color} | {color:green} branch has no errors when building our shaded downstream artifacts. {color} | | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 55s{color} | {color:green} branch-2 passed {color} | | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 35s{color} | {color:green} branch-2 passed {color} | || || || {color:brown} Patch Compile Tests {color} | | {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 9s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 53s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} javac {color} | {color:green} 0m 53s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 16s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace issues. {color} | | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 23s{color} | {color:green} patch has no errors when building our shaded downstream artifacts. {color} | | {color:green}+1{color} | {color:green} hadoopcheck {color} | {color:green} 12m 52s{color} | {color:green} Patch does not cause any errors with Hadoop 2.8.5 2.9.2 or 3.1.2. {color} | | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 3m 9s{color} | {color:green} the patch passed {color} | | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 31s{color} | {color:green} the patch passed {color} | | || || || {color:brown} Other Tests {color} || | {color:red}-1{color} | {color:red} unit {color} | {color:red} 252m 16s{color} | {color:red} hbase-server in the patch failed. {color} | | {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 21s{color} | {color:green} The patch does not generate ASF License warnings. {color} | | {color:black}{color} | {color:black} {color} | {color:black} 299m 37s{color} | | {color:black} {color} | || || || Reason || Tests || Failed junit tests | hadoop.hbase.quotas.TestQuotaAdmin | || || Subsystem || Report/Notes || Docker | Client=17.05.0-ce Server=17.05.0-ce base: https://builds.apache.org/job/PreCommit-HBASE-Build/520/artifact/patchprocess/Dockerfile | | JIRA Issue | HBASE-22169 | | JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12971524/HBASE-22169-branch-2-v1.patch | | Optional Tests | dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile | | uname | Linux ade405237f42 4.4.0-143-generic #169~14.04.2-Ubuntu SMP Wed Feb 13 15:00:41 UTC 2019 x86\_64 GNU/Linux | | Build tool | maven | | Personality | dev-support/hbase-personality.sh | | git revision | branch-2 / 32e501df6c | | maven | version: Apache Maven 3.5.4 (1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z) | | Default Java | 1.8.0\_181 | | findbugs | v3.1.11 | | unit | https://builds.apache.org/job/PreCommit-HBASE-Build/520/artifact/patchprocess/patch-unit-hbase-server.txt | | Test Results | https://builds.apache.org/job/PreCommit-HBASE-Build/520/testReport/ | | Max. process+thread count | 4537 (vs. ulimit of 10000) | | modules | C: hbase-server U: hbase-server | | Console output | https://builds.apache.org/job/PreCommit-HBASE-Build/520/console | | Powered by | Apache Yetus 0.9.0 http://yetus.apache.org | This message was automatically generated.

18. | (/) \*{color:green}+1 overall {color}\* | || || || Vote || Subsystem || Runtime || Comment || | {color:blue}0{color} | {color:blue} reexec {color} | {color:blue} 0m 40s{color} | {color:blue} Docker mode activated. {color} | || || || {color:brown} Prechecks {color} | | {color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-patterns. {color} | | {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch does not contain any @author tags. {color} | | {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} | {color:green} The patch appears to include 2 new or modified test files. {color} | || || || {color:brown} branch-2 Compile Tests {color} | | | {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 35s{color} | {color:green} branch-2 passed {color} | | {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 55s{color} | {color:green} branch-2 passed {color} | | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 21s{color} | {color:green} branch-2 passed {color} | | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 15s{color} | {color:green} branch has no errors when building our shaded downstream artifacts. {color} | | {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 3m 14s{color} | {color:green} branch-2 passed {color} | | {color:green}+1{color} | {color:green} javadoc {color} |



{color:green} 0m 33s{color} | {color:green} branch-2 passed {color} | || || || {color:brown} Patch Compile Tests {color} || |  
{color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 14s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 55s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} javac {color} | {color:green} 0m 55s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 18s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace  
issues. {color} || {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 16s{color} | {color:green} patch has  
no errors when building our shaded downstream artifacts. {color} || {color:green}+1{color} | {color:green} hadoopcheck {color} |  
{color:green} 13m 1s{color} | {color:green} Patch does not cause any errors with Hadoop 2.8.5 2.9.2 or 3.1.2. {color} || |  
{color:green}+1{color} | {color:green} findbugs {color} | {color:green} 3m 25s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 31s{color} | {color:green} the patch passed {color} | || ||  
|| {color:brown} Other Tests {color} || | {color:green}+1{color} | {color:green} unit {color} | {color:green} 252m 51s{color} |  
{color:green} hbase-server in the patch passed. {color} || {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 25s{color} | {color:green} The patch does not generate ASF License warnings. {color} || {color:black}{color} | {color:black}  
{color} | {color:black} 301m 14s{color} | {color:black} {color} | \\\| Subsystem || Report/Notes || Docker | Client=17.05.0-ce  
Server=17.05.0-ce base: <https://builds.apache.org/job/PreCommit-HBASE-Build/521/artifact/patchprocess/Dockerfile> || JIRA Issue |  
HBASE-22169 || JIRA Patch URL | <https://issues.apache.org/jira/secure/attachment/12971524/HBASE-22169-branch-2-v1.patch> ||  
Optional Tests | dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile || uname |  
Linux 306d732e290b 4.4.0-143-generic #169~14.04.2-Ubuntu SMP Wed Feb 13 15:00:41 UTC 2019 x86\_64 GNU/Linux || Build  
tool | maven || Personality | dev-support/hbase-personality.sh || git revision | branch-2 / 32e501df6c || maven | version: Apache Maven  
3.5.4 (1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z) || Default Java | 1.8.0\_181 || findbugs | v3.1.11 || Test  
Results | <https://builds.apache.org/job/PreCommit-HBASE-Build/521/testReport/> || Max. process+thread count | 4459 (vs. ulimit of  
10000) || modules | C: hbase-server U: hbase-server || Console output | <https://builds.apache.org/job/PreCommit-HBASE-Build/521/console> || Powered by | Apache Yetus 0.9.0 <http://yetus.apache.org> | This message was automatically generated.

19. | (x) \*{color:red}-1 overall {color}\* | \\\| Vote || Subsystem || Runtime || Comment || {color:blue}0{color} | {color:blue} reexec  
{color} | {color:blue} 0m 37s{color} | {color:blue} Docker mode activated. {color} | || || || {color:brown} Prechecks {color} || |  
{color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-  
patterns. {color} || {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch  
does not contain any @author tags. {color} || {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} |  
{color:green} The patch appears to include 2 new or modified test files. {color} | || || || {color:brown} branch-2.1 Compile Tests  
{color} || | {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 10s{color} | {color:green} branch-2.1 passed  
{color} || | {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 48s{color} | {color:green} branch-2.1 passed  
{color} || | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 19s{color} | {color:green} branch-2.1 passed  
{color} || | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 7s{color} | {color:green} branch has no  
errors when building our shaded downstream artifacts. {color} || | {color:green}+1{color} | {color:green} findbugs {color} |  
{color:green} 2m 6s{color} | {color:green} branch-2.1 passed {color} || | {color:green}+1{color} | {color:green} javadoc {color} |  
{color:green} 0m 29s{color} | {color:green} branch-2.1 passed {color} | || || || {color:brown} Patch Compile Tests {color} || |  
{color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 0s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 49s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} javac {color} | {color:green} 0m 49s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 20s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace  
issues. {color} || {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 2s{color} | {color:green} patch has no  
errors when building our shaded downstream artifacts. {color} || | {color:green}+1{color} | {color:green} hadoopcheck {color} |  
{color:green} 16m 22s{color} | {color:green} Patch does not cause any errors with Hadoop 2.7.7 2.8.5 or 3.0.3 3.1.2. {color} || |  
{color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 24s{color} | {color:green} the patch passed {color} || |  
{color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 29s{color} | {color:green} the patch passed {color} | || ||  
|| {color:brown} Other Tests {color} || | {color:red}-1{color} | {color:red} unit {color} | {color:red} 199m 3s{color} | {color:red}  
hbase-server in the patch failed. {color} || {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 27s{color} |  
{color:green} The patch does not generate ASF License warnings. {color} || {color:black}{color} | {color:black} {color} |  
{color:black} 247m 11s{color} | {color:black} {color} | \\\| Reason || Tests || Failed junit tests |  
hadoop.hbase.regionserver.TestHRegion | \\\| Subsystem || Report/Notes || Docker | Client=17.05.0-ce Server=17.05.0-ce base:  
<https://builds.apache.org/job/PreCommit-HBASE-Build/522/artifact/patchprocess/Dockerfile> || JIRA Issue | HBASE-22169 || JIRA  
Patch URL | <https://issues.apache.org/jira/secure/attachment/12971554/HBASE-22169-branch-2.1-v1.patch> || Optional Tests |  
dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile || uname | Linux  
cd5b33322b5c 4.4.0-143-generic #169~14.04.2-Ubuntu SMP Wed Feb 13 15:00:41 UTC 2019 x86\_64 GNU/Linux || Build tool |  
maven || Personality | dev-support/hbase-personality.sh || git revision | branch-2.1 / c3e49fc128 || maven | version: Apache Maven  
3.5.4 (1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z) || Default Java | 1.8.0\_181 || findbugs | v3.1.11 || unit |  
<https://builds.apache.org/job/PreCommit-HBASE-Build/522/artifact/patchprocess/patch-unit-hbase-server.txt> || Test Results |  
<https://builds.apache.org/job/PreCommit-HBASE-Build/522/testReport/> || Max. process+thread count | 5274 (vs. ulimit of 10000) ||  
modules | C: hbase-server U: hbase-server || Console output | <https://builds.apache.org/job/PreCommit-HBASE-Build/522/console> || Powered by | Apache Yetus 0.9.0 <http://yetus.apache.org> | This message was automatically generated.

20. | (x) \*{color:red}-1 overall {color}\* | \\\| Vote || Subsystem || Runtime || Comment || {color:blue}0{color} | {color:blue} reexec  
{color} | {color:blue} 1m 16s{color} | {color:blue} Docker mode activated. {color} | || || || {color:brown} Prechecks {color} || |  
{color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-  
patterns. {color} || {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch  
does not contain any @author tags. {color} || {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} |  
{color:green} The patch appears to include 2 new or modified test files. {color} | || || || {color:brown} branch-2.1 Compile Tests  
{color} || | {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 6m 18s{color} | {color:green} branch-2.1 passed  
{color} || | {color:green}+1{color} | {color:green} compile {color} | {color:green} 1m 6s{color} | {color:green} branch-2.1 passed  
{color} || | {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 28s{color} | {color:green} branch-2.1 passed  
{color} || | {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 5m 1s{color} | {color:green} branch has no  
errors when building our shaded downstream artifacts. {color} || | {color:green}+1{color} | {color:green} findbugs {color} |  
{color:green} 2m 36s{color} | {color:green} branch-2.1 passed {color} || | {color:green}+1{color} | {color:green} javadoc {color} |

{color:green} 0m 45s{color} | {color:green} branch-2.1 passed {color} | || || || {color:brown} Patch Compile Tests {color} ||  
{color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 5m 19s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} compile {color} | {color:green} 1m 6s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} javac {color} | {color:green} 1m 6s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 25s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace  
issues. {color} || {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 5m 3s{color} | {color:green} patch has no  
errors when building our shaded downstream artifacts. {color} || {color:green}+1{color} | {color:green} hadoopcheck {color} |  
{color:green} 22m 38s{color} | {color:green} Patch does not cause any errors with Hadoop 2.7.7 2.8.5 or 3.0.3 3.1.2. {color} ||  
{color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 40s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 40s{color} | {color:green} the patch passed {color} | || || ||  
|| {color:brown} Other Tests {color} || {color:red}-1{color} | {color:red} unit {color} | {color:red} 187m 22s{color} | {color:red}  
hbase-server in the patch failed. {color} || {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 26s{color} |  
{color:green} The patch does not generate ASF License warnings. {color} || {color:black}{color} | {color:black} {color} |  
{color:black} 251m 1s{color} | {color:black} || || || Subsystem || Report/Notes || Docker | Client=17.05.0-ce Server=17.05.0-  
ce base: https://builds.apache.org/job/PreCommit-HBASE-Build/529/artifact/patchprocess/Dockerfile || JIRA Issue | HBASE-22169 ||  
JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12971645/HBASE-22169-branch-2.1-v1.patch || Optional Tests |  
dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile || uname | Linux 3ec81bd2fd49  
4.4.0-143-generic #169~14.04.2-Ubuntu SMP Wed Feb 13 15:00:41 UTC 2019 x86\_64 GNU/Linux || Build tool | maven ||  
Personality | dev-support/hbase-personality.sh || git revision | branch-2.1 / 1ccf37d7cb || maven | version: Apache Maven 3.5.4  
(1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z) || Default Java | 1.8.0\_181 || findbugs | v3.1.11 || unit |  
https://builds.apache.org/job/PreCommit-HBASE-Build/529/artifact/patchprocess/patch-unit-hbase-server.txt || Test Results |  
https://builds.apache.org/job/PreCommit-HBASE-Build/529/testReport/ || Max. process+thread count | 5243 (vs. ulimit of 10000) ||  
modules | C: hbase-server U: hbase-server || Console output | https://builds.apache.org/job/PreCommit-HBASE-Build/529/console ||  
Powered by | Apache Yetus 0.9.0 http://yetus.apache.org | This message was automatically generated.

21. | (x) \*{color:red}-1 overall{color}\* | || || || Vote || Subsystem || Runtime || Comment || {color:blue}0{color} | {color:blue} reexec  
{color} | {color:blue} 0m 46s{color} | {color:blue} Docker mode activated. {color} | || || || {color:brown} Prechecks {color} ||  
{color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-  
patterns. {color} || {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch  
does not contain any @author tags. {color} || {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} |  
{color:green} The patch appears to include 2 new or modified test files. {color} | || || || {color:brown} branch-2.1 Compile Tests  
{color} || {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 3m 58s{color} | {color:green} branch-2.1 passed  
{color} || {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 49s{color} | {color:green} branch-2.1 passed  
{color} || {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 11s{color} | {color:green} branch-2.1 passed  
{color} || {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 3m 52s{color} | {color:green} branch has no  
errors when building our shaded downstream artifacts. {color} || {color:green}+1{color} | {color:green} findbugs {color} |  
{color:green} 2m 10s{color} | {color:green} branch-2.1 passed {color} || {color:green}+1{color} | {color:green} javadoc {color} |  
{color:green} 0m 30s{color} | {color:green} branch-2.1 passed {color} | || || || {color:brown} Patch Compile Tests {color} ||  
{color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 3m 42s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 48s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} javac {color} | {color:green} 0m 48s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 9s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace  
issues. {color} || {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 3m 49s{color} | {color:green} patch has  
no errors when building our shaded downstream artifacts. {color} || {color:green}+1{color} | {color:green} hadoopcheck {color} |  
{color:green} 15m 27s{color} | {color:green} Patch does not cause any errors with Hadoop 2.7.7 2.8.5 or 3.0.3 3.1.2. {color} ||  
{color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 13s{color} | {color:green} the patch passed {color} ||  
{color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 29s{color} | {color:green} the patch passed {color} | || || ||  
|| {color:brown} Other Tests {color} || {color:red}-1{color} | {color:red} unit {color} | {color:red} 244m 16s{color} | {color:red}  
hbase-server in the patch failed. {color} || {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 37s{color} |  
{color:green} The patch does not generate ASF License warnings. {color} || {color:black}{color} | {color:black} {color} |  
{color:black} 290m 13s{color} | {color:black} || || || Reason || Tests || Failed junit tests |  
hadoop.hbase.TestRegionRebalancing || | hadoop.hbase.client.TestAdmin1 || || Subsystem || Report/Notes || Docker |  
Client=17.05.0-ce Server=17.05.0-ce base: https://builds.apache.org/job/PreCommit-HBASE-  
Build/532/artifact/patchprocess/Dockerfile || JIRA Issue | HBASE-22169 || JIRA Patch URL |  
https://issues.apache.org/jira/secure/attachment/12971670/HBASE-22169-branch-2.1-v1.patch || Optional Tests | dupname asflicense  
javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile || uname | Linux adea7909a1b7 4.4.0-138-generic  
#164-Ubuntu SMP Tue Oct 2 17:16:02 UTC 2018 x86\_64 GNU/Linux || Build tool | maven || Personality | dev-support/hbase-  
personality.sh || git revision | branch-2.1 / 1ccf37d7cb || maven | version: Apache Maven 3.5.4  
(1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z) || Default Java | 1.8.0\_181 || findbugs | v3.1.11 || unit |  
https://builds.apache.org/job/PreCommit-HBASE-Build/532/artifact/patchprocess/patch-unit-hbase-server.txt || Test Results |  
https://builds.apache.org/job/PreCommit-HBASE-Build/532/testReport/ || Max. process+thread count | 5380 (vs. ulimit of 10000) ||  
modules | C: hbase-server U: hbase-server || Console output | https://builds.apache.org/job/PreCommit-HBASE-Build/532/console ||  
Powered by | Apache Yetus 0.9.0 http://yetus.apache.org | This message was automatically generated.
22. | (/) \*{color:green}+1 overall{color}\* | || || || Vote || Subsystem || Runtime || Comment || {color:blue}0{color} | {color:blue} reexec  
{color} | {color:blue} 0m 31s{color} | {color:blue} Docker mode activated. {color} | || || || {color:brown} Prechecks {color} ||  
{color:green}+1{color} | {color:green} hbaseanti {color} | {color:green} 0m 0s{color} | {color:green} Patch does not have any anti-  
patterns. {color} || {color:green}+1{color} | {color:green} @author {color} | {color:green} 0m 0s{color} | {color:green} The patch  
does not contain any @author tags. {color} || {color:green}+1{color} | {color:green} test4tests {color} | {color:green} 0m 0s{color} |  
{color:green} The patch appears to include 2 new or modified test files. {color} | || || || {color:brown} branch-2.1 Compile Tests  
{color} || {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 4m 20s{color} | {color:green} branch-2.1 passed  
{color} || {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 50s{color} | {color:green} branch-2.1 passed  
{color} || {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 14s{color} | {color:green} branch-2.1 passed  
{color} || {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 4m 7s{color} | {color:green} branch has no

errors when building our shaded downstream artifacts. {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 26s{color} | {color:green} branch-2.1 passed {color} | {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 30s{color} | {color:green} branch-2.1 passed {color} | |||| {color:brown} Patch Compile Tests {color} || {color:green}+1{color} | {color:green} mvninstall {color} | {color:green} 3m 47s{color} | {color:green} the patch passed {color} || {color:green}+1{color} | {color:green} compile {color} | {color:green} 0m 51s{color} | {color:green} the patch passed {color} || {color:green}+1{color} | {color:green} javac {color} | {color:green} 0m 51s{color} | {color:green} the patch passed {color} || {color:green}+1{color} | {color:green} checkstyle {color} | {color:green} 1m 14s{color} | {color:green} the patch passed {color} || {color:green}+1{color} | {color:green} whitespace {color} | {color:green} 0m 0s{color} | {color:green} The patch has no whitespace issues. {color} || {color:green}+1{color} | {color:green} shadedjars {color} | {color:green} 3m 57s{color} | {color:green} patch has no errors when building our shaded downstream artifacts. {color} || {color:green}+1{color} | {color:green} hadoopcheck {color} | {color:green} 15m 59s{color} | {color:green} Patch does not cause any errors with Hadoop 2.7.7 2.8.5 or 3.0.3 3.1.2. {color} || {color:green}+1{color} | {color:green} findbugs {color} | {color:green} 2m 21s{color} | {color:green} the patch passed {color} || {color:green}+1{color} | {color:green} javadoc {color} | {color:green} 0m 27s{color} | {color:green} the patch passed {color} || || || || {color:brown} Other Tests {color} || {color:green}+1{color} | {color:green} unit {color} | {color:green} 132m 50s{color} | {color:green} hbase-server in the patch passed. {color} || {color:green}+1{color} | {color:green} asflicense {color} | {color:green} 0m 26s{color} | {color:green} The patch does not generate ASF License warnings. {color} || {color:black} {color} | {color:black} {color} | {color:black} 180m 18s{color} | {color:black} {color} | \\\| Subsystem || Report/Notes || Docker | Client=17.05.0-ce Server=17.05.0-ce base: https://builds.apache.org/job/PreCommit-HBASE-Build/538/artifact/patchprocess/Dockerfile || JIRA Issue | HBASE-22169 || JIRA Patch URL | https://issues.apache.org/jira/secure/attachment/12971755/HBASE-22169-branch-2.1-v1.patch || Optional Tests | dupname asflicense javac javadoc unit findbugs shadedjars hadoopcheck hbaseanti checkstyle compile || uname | Linux 2be4623fa603 4.4.0-138-generic #164-Ubuntu SMP Tue Oct 2 17:16:02 UTC 2018 x86\_64 GNU/Linux || Build tool | maven || Personality | dev-support/hbase-personality.sh || git revision | branch-2.1 / b355020224 || maven | version: Apache Maven 3.5.4 (1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T18:33:14Z) || Default Java | 1.8.0\_181 || findbugs | v3.1.11 || Test Results | https://builds.apache.org/job/PreCommit-HBASE-Build/538/testReport/ || Max. process+thread count | 5040 (vs. ulimit of 10000) || modules | C: hbase-server U: hbase-server || Console output | https://builds.apache.org/job/PreCommit-HBASE-Build/538/console || Powered by | Apache Yetus 0.9.0 http://yetus.apache.org | This message was automatically generated.

23. [~stack] I run this patch on branch-master and branch 2.1, all tests is pass; Could this patch commit now?4Y

24. Pushed on branch-2.1, 2.2, 2, and master. Thanks for the patch [~luffy123]. I tried the patch locally making sure the failing tests passed before pushing.

25. Results for branch branch-2.2 [build #400 on builds.a.o|https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/400/]: (x) \* {color:red}-1 overall{color}\* ---- details (if available): (/) {color:green}+1 general checks{color} -- For more information [see general report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/400//General\_Nightly\_Build\_Report/] (/) {color:green}+1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/400//JDK8\_Nightly\_Build\_Report\_(Hadoop2)/] (x) {color:red}-1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3) report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2.2/400//JDK8\_Nightly\_Build\_Report\_(Hadoop3)/] (/) {color:green}+1 source release artifact{color} -- See build output for details. (/) {color:green}+1 client integration test{color}

26. Results for branch master [build #1187 on builds.a.o|https://builds.apache.org/job/HBase%20Nightly/job/master/1187/]: (x) \* {color:red}-1 overall{color}\* ---- details (if available): (x) {color:red}-1 general checks{color} -- For more information [see general report|https://builds.apache.org/job/HBase%20Nightly/job/master/1187//General\_Nightly\_Build\_Report/] (x) {color:red}-1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report|https://builds.apache.org/job/HBase%20Nightly/job/master/1187//JDK8\_Nightly\_Build\_Report\_(Hadoop2)/] (x) {color:red}-1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3) report|https://builds.apache.org/job/HBase%20Nightly/job/master/1187//JDK8\_Nightly\_Build\_Report\_(Hadoop3)/] (/) {color:green}+1 source release artifact{color} -- See build output for details. (/) {color:green}+1 client integration test{color}

27. Results for branch branch-2 [build #2039 on builds.a.o|https://builds.apache.org/job/HBase%20Nightly/job/branch-2/2039/]: (x) \* {color:red}-1 overall{color}\* ---- details (if available): (x) {color:red}-1 general checks{color} -- For more information [see general report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2/2039//General\_Nightly\_Build\_Report/] (x) {color:red}-1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2/2039//JDK8\_Nightly\_Build\_Report\_(Hadoop2)/] (x) {color:red}-1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3) report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2/2039//JDK8\_Nightly\_Build\_Report\_(Hadoop3)/] (/) {color:green}+1 source release artifact{color} -- See build output for details. (/) {color:green}+1 client integration test{color}

28. Results for branch branch-2.1 [build #1319 on builds.a.o|https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1319/]: (x) \* {color:red}-1 overall{color}\* ---- details (if available): (/) {color:green}+1 general checks{color} -- For more information [see general report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1319//General\_Nightly\_Build\_Report/] (/) {color:green}+1 jdk8 hadoop2 checks{color} -- For more information [see jdk8 (hadoop2) report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1319//JDK8\_Nightly\_Build\_Report\_(Hadoop2)/] (x) {color:red}-1 jdk8 hadoop3 checks{color} -- For more information [see jdk8 (hadoop3) report|https://builds.apache.org/job/HBase%20Nightly/job/branch-2.1/1319//JDK8\_Nightly\_Build\_Report\_(Hadoop3)/] (/) {color:green}+1 source release artifact{color} -- See build output for details. (/) {color:green}+1 client integration test{color}

29. **body:** [~luffy123], [~stack] {{TestHRegion#testOpenRegionFailedMemoryLeak}} fails consistently (2/3 times) for me on 2.1.6RC0 with the following stacktrace. I think you have a race between the {{Time-Limited test}} timeout and the amount of time the test waits for metrics to propagate. It'll timeout if the amount of time spent by other test methods before this test is run eat up the overall class timeout quota. For this reason, I suggest extracting this method out into it's own dedicated class. FYI [~Apache9] . {noformat} 2019-08-21 11:15:34,239 ERROR [Time-limited test] coprocessor.CoprocessorHost(398): The coprocessor test threw java.io.IOException: Failed on local exception: java.nio.channels.ClosedByInterruptException; Host Details : local host is: "myson.local/192.168.111.14"; destination host is: "test":8020; java.io.IOException: Failed on local exception: java.nio.channels.ClosedByInterruptException; Host Details : local host is: "myson.local/192.168.111.14"; destination host is: "test":8020; at org.apache.hadoop.net.NetUtils.wrapException(NetUtils.java:773) at org.apache.hadoop.ipc.Client.call(Client.java:1480) at org.apache.hadoop.ipc.Client.call(Client.java:1413) at org.apache.hadoop.ipc.ProtobufRpcEngine\$Invoker.invoke(ProtobufRpcEngine.java:229) at com.sun.proxy.\$Proxy38.getFileInfo(Unknown Source) at org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.getFileInfo(ClientNamenodeProtocolTranslatorPB.java:776)

at sun.reflect.GeneratedMethodAccessor40.invoke(Unknown Source) at  
 sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43) at  
 java.lang.reflect.Method.invoke(Method.java:498) at  
 org.apache.hadoop.io.retry.RetryInvocationHandler.invokeMethod(RetryInvocationHandler.java:191) at  
 org.apache.hadoop.io.retry.RetryInvocationHandler.invoke(RetryInvocationHandler.java:102) at  
 com.sun.proxy.\$Proxy39.getFileInfo(Unknown Source) at org.apache.hadoop.hdfs.DFSClient.getFileInfo(DFSClient.java:2117) at  
 org.apache.hadoop.hdfs.DistributedFileSystem\$22.doCall(DistributedFileSystem.java:1305) at  
 org.apache.hadoop.hdfs.DistributedFileSystem\$22.doCall(DistributedFileSystem.java:1301) at  
 org.apache.hadoop.fs.FileSystemLinkResolver.resolve(FileSystemLinkResolver.java:81) at  
 org.apache.hadoop.hdfs.DistributedFileSystem.getFileStatus(DistributedFileSystem.java:1301) at  
 org.apache.hadoop.fs.FileSystem.isFile(FileSystem.java:1451) at  
 org.apache.hadoop.hbase.util.CoprocessorClassLoader.getClassLoader(CoprocessorClassLoader.java:250) at  
 org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:211) at  
 org.apache.hadoop.hbase.coprocessor.CoprocessorHost.load(CoprocessorHost.java:185) at  
 org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.loadTableCoprocessors(RegionCoprocessorHost.java:388) at  
 org.apache.hadoop.hbase.regionserver.RegionCoprocessorHost.<init>(RegionCoprocessorHost.java:275) at  
 org.apache.hadoop.hbase.regionserver.HRegion.<init>(HRegion.java:827) at org.apache.hadoop.hbase.regionserver.HRegion.<init>  
 (HRegion.java:716) at sun.reflect.GeneratedConstructorAccessor21.newInstance(Unknown Source) at  
 sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45) at  
 java.lang.reflect.Constructor.newInstance(Constructor.java:423) at  
 org.apache.hadoop.hbase.regionserver.HRegion.newHRegion(HRegion.java:7008) at  
 org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7218) at  
 org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7191) at  
 org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7149) at  
 org.apache.hadoop.hbase.regionserver.HRegion.openHRegion(HRegion.java:7100) at  
 org.apache.hadoop.hbase.regionserver.TestHRegion.testOpenRegionFailedMemoryLeak(TestHRegion.java:6623) {noformat}

**label:** test

30. [~ndimiduk] thanks for your suggest, I will fix it in HBASE-22894