Red Hat JBoss Data Grid 6.6

6.6.1 Release Notes

Known and resolved issues for Red Hat JBoss Data Grid 6.6.1

Red Hat JBoss Data Grid 6.6 6.6.1 Release Notes

Known and resolved issues for Red Hat JBoss Data Grid 6.6.1

Christian Huffman
Red Hat Engineering Content Services
chuffman@redhat.com

Legal Notice

Abstract

The Red Hat JBoss Data Grid 6.6.1 Release Notes list and provide descriptions for a series of bugzilla bugs. The bugs highlight issues that are known problems and resolved issues for the release.

Table of Contents

CHAPTER 1. INTRODUCTION TO RED HAT JBOSS DATA GRID 6.6.1

Welcome to Red Hat JBoss Data Grid 6.6.1. As you become familiar with the newest version of JBoss Data Grid these Release Notes provide you with information about new features, as well as known and resolved issues. Use this document in conjunction with the entire JBoss Data Grid documentation suite, available at the Red Hat Customer Service Portal's JBoss Data Grid documentation page.

Report a bug

1.1. ABOUT RED HAT JBOSS DATA GRID

Red Hat's JBoss Data Grid is an open source, distributed, in-memory key/value data store built from the Infinispan open source software project. Whether deployed in client/server mode or embedded in a Java Virtual Machine, it is built to be elastic, high performance, highly available and to scale linearly.

JBoss Data Grid is accessible for both Java and Non-Java clients. Using JBoss Data Grid, data is distributed and replicated across a manageable cluster of nodes, optionally written to disk and easily accessible using the REST, Memcached and Hot Rod protocols, or directly in process through a traditional Java Map API.

Report a bug

1.2. OVERVIEW

This document contains information about the known and resolved issues of Red Hat JBoss Data Grid version 6.6.0. Customers are requested to read this documentation prior to installing this version.

Report a bug

CHAPTER 2. PATCHING EXISTING SERVER INSTANCES

Micro releases for the JBoss Data Grid server are distributed as a patch. This is for your convenience, as the new version will not completely replace the installation. The base version to be patched is the x.y.0 release, and the patches x.y.# (micro release or cumulative patch) may be applied to that instance. The patch can be rolled back (including the configuration) if there is any issue during the patching process, and any subsequent patches may be applied in the same way.

This process is similar to the already known procedure in JBoss Enterprise Application Platform (EAP).
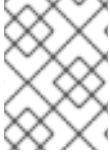
NOTE

It is strongly recommended to back up your existing installation, including all configuration files, before applying the patch.

Procedure 2.1. Applying the JBoss Data Grid 6.6.1 Patch

1. Download the patch from the Red Hat Customer Portal at https://access.redhat.com/downloads/

2. Connect to the running instance to be patched using the JBoss CLI:

   ```
   $JDG_HOME/bin/cli.sh --connect=127.0.0.1:9999
   ```

3. Ensure that there are no active connections to the server, and then apply the patch:

   ```
   patch apply /path/to/jboss-datagrid-6.6.1.CR1-server-patch.zip
   ```

4. Restart the server:

   ```
   shutdown --restart=true
   ```

All other distributions, such as the EAP modules, client's, and standalone library mode are provided as a full release. Due to this there is no possibility to patch existing archives for these distributions.

Report a bug

CHAPTER 3. SUPPORTED CONFIGURATIONS

3.1. SUPPORTED CONFIGURATIONS

For supported hardware and software configurations, see the Red Hat JBoss Data Grid Supported Configurations reference on the Customer Portal at https://access.redhat.com/site/articles/115883.

Report a bug

CHAPTER 4. COMPONENT VERSIONS

4.1. COMPONENT VERSIONS

The full list of component versions used in Red Hat JBoss Data Grid is available at the Customer Portal at https://access.redhat.com/site/articles/488833.

Report a bug

CHAPTER 5. KNOWN AND RESOLVED ISSUES

5.1. KNOWN ISSUES

BZ-1200822 - JSR-107 Support for clustered caches in HotRod implementation

When creating a new cache (which is not defined in server configuration file) in HotRod implementation of JSR-107, the cache is created as **local** only in one of the servers. This behavior requires class **org.jboss.as.controller.client.ModelControllerClient** to be present on the classpath.

As a workaround use a clustered cache defined in the server configuration file. This still requires **cacheManager.createCache(cacheName, configuration)** to be invoked before accessing the cache for the first time.

BZ-1204813 - JSR-107 Support for cacheResolverFactory annotation property

JCache annotations provides a way to define a custom **CacheResolverFactory**, used to produce **CacheResolver**; this class's purpose is to decide which cache is used for storing results of annotated methods; however, the support for specifying a **CacheResolver** is not provided yet.

As a workaround, define a CDI ManagedCacheResolver which will be used instead.

BZ-1223290 - JPA Cache Store not working properly on Weblogic

A JPA Cache Store deployed to WebLogic servers throws a **NullPointerException** after the following error message:

```
Entity manager factory name (org.infinispan.persistence.jpa) is already
registered
```

This is a known issue in Red Hat JBoss Data Grid 6.6.1, and no workaround exists at this time.

BZ-1158839 - Clustered cache with FileStore (shared=false) is inconsistent after restarting one node if entries are deleted during restart

In Red Hat JBoss Data Grid, when a node restarts, it does not automatically purge entries from its local cache store. As a result, the Administrator starting the node must change the node configuration manually to set the cache store to be purged when the node is starting. If the configuration is not changed, the cache may be inconsistent (removed entries can appear to be present).

This is a known issue in Red Hat JBoss Data Grid 6.6.1, and no workaround exists at this time.

BZ-1114080 - HR client SASL MD5 against LDAP fails

In Red Hat JBoss Data Grid, the server does not support pass-through MD5 authentication against LDAP. As a result, the Hot Rod client is unable to authenticate to the JBoss Data Grid server via MD5 is the authentication is backed by the LDAP server.

This is a known issue in Red Hat JBoss Data Grid 6.6.1 and a workaround is to use the PLAIN authentication over end-to-end SSL encryption.

BZ-1024373 - Default optimistic locking configuration leads to inconsistency

In Red Hat JBoss Data Grid, transactional caches are configured with optimistic locking by default. Concurrent **replace()** calls can return true under contention and transactions might unexpectedly commit.

Two concurrent commands, **replace(key, A, B)** and **replace(key, A, C)** may both overwrite the entry. The command which is finalized later wins, overwriting an unexpected value with new value.

This is a known issue in Red Hat JBoss Data Grid 6.6.1. As a workaround, enable write skew check and the *REPEATABLE_READ* isolation level. This results in concurrent replace operations working as expected.

BZ-1273411 - Cannot access cache with authorization enabled when using REST protocol

When authorization is configured for a cache, then any access to the cache via REST endpoint results in a security exception. A user is not able to access the cache since the security **Subject** representing the user is not properly defined, and the user cannot be authorized to access the cache.

This is a known issue in Red Hat JBoss Data Grid 6.6.1, and no workaround exists at this time.

Report a bug

5.2. RESOLVED ISSUES

BZ-1333177 - ProtobufIndexedFieldProvider may mistakenly report nested fields as unindexed

Nested indexed fields would sometimes be reported as unindexed even if the metadata says otherwise. This could cause a query to be run fully or partially unindexed, resulting in performance issues.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1340827 - ContinuousQueryListener.resultJoining notifications generated by modify operations contain the old value not the new one

ContinuousQueryListener.resultJoining notifications generated by **modify** operations contain the old value not the new one. This issue only affected queries without projections.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1322679 - NullPointerException from HotRodDecoder when topology changed

When a topology change occurs, such as a node joining the cluster, the connecting client receives **HotRodClientException** and **InvalidResponseExceptions** whenever it would execute a **get** operation.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1332272 - NullPointerException at org.infinispan.remoting.transport.jgroups.CommandAwareRpcDispatcher$FutureCollator.futureDon

A **NullPointerException** could be thrown during rebalancing. This exception was caused by **FutureCollator.futures.get()** querying the map before **FutureCollator.futures.put()** completed.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1342365 - Client doesn't retry a non-tx write operation when one of backups is suspected

When using the Hot Rod Java client with multiple backpus it was possible to have the
`Cache.remove()` method fail, without attempting a retry, if one of the backups is no longer
available.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1293575 - Rolling upgrade fails with keySet larger than 2 GB

Rolling upgrades failed if the key set was larger than 2 GB of memory. The process failed when
calling `recordKnownGlobalKeyset` because the keys could not be dumped into a single byte
array in the source cluster.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

**BZ-1324030 - ClassCastException at cache.getWithMetadata(key) in HotRod client during JDG
rolling upgrade**

When performing a rolling upgrade a `ClassCastException` was thrown on each
`getWithMetadata()` for every key.

This issue was related in the `EntryWrapping` logic when reading data from a remote store, and is
resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1357823 - Expose configuration for the SSLContext protocol in the Hot Rod client

Previously, the Hot Rod client had the `SSLContext` set with a hardcoded `TLS` protocol. This
prevented java 7 clients from using `TLSv1.2`, since it was not enabled by default.

This has been addressed in JBoss Data Grid 6.6.1 by allowing clients to specify a SSL protocol via
the `infinispan.client.hotrod.ssl_protocol` property. To use the `TLSv1.2` protocol the
following should be defined:

```
infinispan.client.hotrod.ssl_protocol=TLSv1.2
```

If no protocol is defined JBoss Data Grid defaults to using `TLS`.

BZ-1312186 - Non-threadsafe use of HashSet in AdvancedAsyncCacheLoader

A `HashSet` was created from the `AdvancedAsyncCacheLoader$process` operation, which was
then accessed by a series of tasks running in parallel, resulting in the possibility of infinite loops.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1300133 - JMX attribute evictions is always zero in Statistics and ClusterCacheStats MBeans

The `evictions` attribute of `Statistics` and `ClusterCacheStats` components of the Cache
MBean returned zero even though some eviction operations have been successfully performed. This
issue only affected statistics, not the actual eviction process.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1303483 - Clear command causing deadlock when lock time out happens

If a timeout occurred when using the **clear** command the lock it was holding was never released, resulting in a deadlock as the Hot Rod client retried the command on another node.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1309181 - Lock timeout when registering a queried class to the registry cache

When starting a series of JBoss Data Grid nodes in parallel, and having each node register a queried class to the **__cluster_registry_cache__**, lock timeout errors were printed.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1312185 - Double invalidate of invalid Hot Rod connections

When a Hot Rod operation resulted in an error, **RetryOnFailureOperation** would invalidate the connection twice. Initially in a catch block, and a second time in a finally block. This behavior resulted in the **GenericKeyedObjectPool** counts to become unreliable.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1315393 - Hot Rod client stuck sporadical if a server node is shutting down

When executing remote transactions as the server is shutting down, the client could become blocked or receive **TimeoutException** errors.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1306389 - JGRP-2014 FILE_PING destination file name can include File.separator characters

Previously it was possible for the **FILE_PING** JGroups discovery method to include a file separator, such as **/** or ****.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1, as these characters are now replaced with **-**.

BZ-1348404 - Coordinator failover is taking longer because VERIFY_SUSPECT runs twice

When using the Hot Rod Java client the time to recover takes twice as long if the coordinator is killed. This was due to **VERIFY_SUSPECT** executing twice.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1310583 - TransactionTable should ignore view changes during shutdown

During shutdown, the **TransactionTable** unregisters itself as a view change listener, but it can still receive view change notifications after it stopped the executor service. When this happens, it causes a **RejectedExecutionException** that is eventually logged by JGroups.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1314084 - protostream.javassist.NotFoundException when using ProtoSchemabuilder to register class

When configuring JBoss Data Virtualization to connect to a JBoss Data Grid remote cache using the Hot Rod EAP client modules, **ProtoSchemaBuilder** is called, resulting in a **NotFoundException** error being thrown.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1332924 - JCache remote's getCache() should retrieve existing remote caches

If a client used the JCache API together with the **remote-java-client** library a **getCache(cacheName)** would return **NULL**, even if the cache is configured within the server. This was caused by the JCache API not examining remote caches.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1332936 - If Infinispan is used as a provider for JCache using the remote approach it will not pick up the hotrod-client.properties

If an application used the JCache API together with the **infinispan-jcache-remote** library then the **CacheManager** was created with default properties, as no property files were passed into the constructor.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

BZ-1333098 - Add EXTERNAL SASL support to the client to be used with a JDG 7 server

A JBoss Data Grid 6.x node communicating to a JBoss Data Grid 7 server was unable to authenticate using the **SASL_EXTERNAL** method, as this is a new feature in JBoss Data Grid 7.

This issue is resolved for Red Hat JBoss Data Grid 6.6.1 clients. These clients may now use the server's SSL certificate **CN** chain to authenticate the user.

BZ-1339313 - Configure Hot Rod client with properties

Some properties, such as **SSL**, **FILE_STORE**, and **TRUST_STORE**, were only allowed to be defined programmatically, and could not be placed in the **hotrod-client.properties** file.

This issue is resolved for Red Hat JBoss Data Grid 6.6.1 clients, and these properties may be defined either programmatically or declaratively.

BZ-1313589 - (CVE-2016-2141) CVE-2016-2141 Authorization bypass in JGroups

It was found that JGroups did not require necessary headers for encrypt and auth protocols from new nodes joining the cluster. An attacker could use this flaw to bypass security restrictions, and use this vulnerability to send and receive messages within the cluster, leading to information disclosure, message spoofing, or further possible attacks.

This issue is resolved as of Red Hat JBoss Data Grid 6.6.1.

Report a bug

APPENDIX A. REVISION HISTORY

Revision 6.6.1-3   **Sun 11 Sep 2016**   **Christian Huffman**
   Included section on patching server.

Revision 6.6.1-2   **Fri 9 Sep 2016**   **Christian Huffman**
   Updated list of resolved issues.

Revision 6.6.1-1   **Wed 7 Sep 2016**   **Christian Huffman**
   Included set of known and resolved issues.

Revision 6.6.1-0   **Wed 7 Sep 2016**   **Christian Huffman**
   Initial Draft for 6.6.1.