Cloudera Manager 7.13.1

# Unified Cloudera Manager Release Notes

**Date published: 2024-05-15**
**Date modified: 2024-05-15**

CLOUDERA

**https://docs.cloudera.com/**

# Legal Notice

# Contents

# Cloudera Manager 7.13.2 Release Notes

This document provides a summary of the latest updates in Cloudera Manager 7.13.2 and its Cumulative hotfixes. You can review the Release Notes of Cloudera Manager 7.13.2 associated with unified Cloudera Runtime 7.3.2 (includes Cloudera Base on premises and Cloudera on cloud) for release-specific information related to new features and improvements, bug fixes, deprecated features and components, known issues, and changed features that can affect product behavior.

**Important:** Note the following information before proceeding:

- For upgrading Cloudera Manager instructions, see Upgrading Cloudera Manager 7.
- Changes to features in Cloudera Manager 7.13.2 including new features, configuration updates, or behavioral changes impact the unified Cloudera Runtime 7.3.2.
- Platform support changes in Cloudera Manager 7.13.2 also apply across the unified Cloudera Runtime 7.3.2. For more information about the supported infrastructure combinations, see Cloudera support matrix.

## What's New in Cloudera Manager

Learn about the new features and changed behavior of Cloudera Manager in Cloudera Manager 7.13.2 and its cumulative hotfixes.

You must be aware of the additional functionalities and improvements to features of Cloudera Manager in Cloudera Manager 7.13.2 and its cumulative hotfixes. Learn how the new features and improvements benefit you.

Cloudera Manager 7.13.2 introduces new features and includes all cumulative hotfixes from 7.13.1.100 through 7.13.1.700. For a comprehensive record of all updates in Cloudera Manager 7.13.1.x, see What's New in Cloudera Manager 7.13.1.x.

### Cloudera Manager 7.13.2

**Dual-Stack Support: IPv4 and IPv6 Address Visibility**

Cloudera Manager now displays both IPv4 and IPv6 addresses for hosts in clusters where nodes are configured with dual-stack networking.

**Important:** This configuration only enables IPv4 and IPv6 address visibility within the Cloudera Manager UI. The Cloudera Manager Server itself continues to operate using IPv4 networking.

For more information, see IPv6 Support and Dual-Stack Configuration.

**Python support for Cloudera Manager 7.13.2**

Cloudera Manager 7.13.2 now supports only Python 3.11 version, when used in combination with Cloudera Runtime 7.3.2, 7.3.1.500 or higher versions, and 7.1.9 SP1 CHF11 or higher versions. For more information, see Installing Python 3.

**Ubuntu 24.04 support for Cloudera Manager 7.13.2**

Starting with the Cloudera Manager 7.13.2.0 release, Cloudera Manager provides support for Ubuntu 24.04. This update ensures seamless compatibility with Ubuntu version 24.04, offering greater flexibility and platform options.

Ubuntu 24.04 supports only the Python 3.11 version in the Cloudera Manager 7.13.2 release.

**Managing Service Restarts on Ubuntu 24.04:**

Overriding needrestart Automatic Restart Behaviour

Ubuntu 24.04 (Noble Numbat) introduced a significant change to the default behavior of the need restart utility. By default, needrestart is now configured to automatically restart services any time apt updates underlying libraries (for example, libssl).

This behaviour is designed to apply security patches to running processes immediately. However, this can cause unplanned downtime for services like cloudera-scm-server.service (the Cloudera Manager Server).

To ensure service stability and to control all restarts, you can disable this automatic behaviour on all production Ubuntu 24.04 hosts. You can use multiple approaches to achieve this, as outlined in the Ubuntu Community Discourse: `needrestart` changes in Ubuntu 24.04: service restarts.

**SLES 15 SP6 support for Cloudera Manager**

Starting with the Cloudera Manager 7.13.2.0 release, Cloudera Manager provides support for SLES 15 SP6.

SLES 15 SP6 supports only the Python 3.11 version in the Cloudera Manager 7.13.2 release.

**Kafka protocol and metadata version is set automatically during upgrades**

When upgrading Kafka, Cloudera Manager now automatically sets the inter.broker.protocol.version property for ZooKeeper-based clusters and the metadata.version property for KRaft-based clusters. You no longer need to manually set these properties to the current protocol or metadata version before an upgrade. This feature is only available when upgrading to Cloudera Runtime 7.3.2 or higher.

After the upgrade, clearing these properties remains a manual task. However, in Cloudera Runtime 7.3.2 and higher, both inter.broker.protocol.version and metadata.version are now available for direct configuration in Cloudera Manager Kafka Configuration . The label names of the properties are Kafka Inter-Broker Protocol Version and Kafka Metadata Version. This means you can set or clear these properties directly from the UI, without needing to use advanced configuration snippets.

## What's new in Platform Support

You must be aware of the platform support changes for the Cloudera Manager 7.13.2 release.

### Cloudera Manager 7.13.2

**Platform Support Enhancements**

- **New OS support:** Cloudera Manager 7.13.2 now supports the following operating systems:
  - RHEL 9.6
  - Rocky Linux 9.6
  - SLES 15 SP6
  - Ubuntu 24.04

  For more information about the operating system support, see Cloudera Support Matrix.
- **New Database support**: Cloudera Manager 7.13.2 now supports the following databases:
  - MariaDB 11.4
- **New JDK Version**: None

## Release Matrix

You must be familiar with the versions of all the cumulative hotfixes for Cloudera Manager 7.13.2.

The Release build number has two parts, the [*Cloudera Manager release version number*] and the [*Cloudera Manager Build number*]. For example, in 7.13.2.100-69, 7.13.2.100 is the release version number and 69 is the build number.

### Table 1: Cloudera Manager versions certified with Cloudera Base on premises

| Cloudera Manager 7.13.2 Release Series | Cloudera Manager Release Build Number | GBN | Release Date | Download URL |
|---|---|---|---|---|
| 7.13.2.0 | 7.13.2.0-709 | 77091850 | March 31, 2026 | https://archive.cloudera.com/p/cm7/7.13.2.0/ |

## Fixed Issues

Review the list of Cloudera Manager issues that are resolved in Cloudera Manager 7.13.2 and its cumulative hotfixes.

Cloudera Manager 7.13.2 resolves issues and incorporates fixes from the cumulative hotfixes from 7.13.1.100 through 7.13.1.700. For a comprehensive record of all fixes in Cloudera Manager 7.13.1.x, see Fixed Issues 7.13.1.x.

### Cloudera Manager 7.13.2

**OPSAPS-74276: RockDB JNI library is loaded from the same place to multiple Ozone components**

7.13.2.0

By default, Ozone roles define a separate directory to load RocksDB shared library, and clean up separately from each other on the same host, unless the environment already defines the ROCKSDB_SHAREDLIB_DIR variable via a Safety valve as suggested in the workaround for OPSAPS-67650. After this change, that workaround becomes obsolete. The new directory used reside within directories used by the Cloudera Manager agent to manage the Ozone related processes.

**OPSAPS-73808: Cloudera Manager is not propagating the Storage Container Manager Block Client port to om roles**

7.13.2.0

Previously, the Ozone Managers did not start, and/or the Storage Container Managers never left safe mode because the DataNodes were not able to register with them when these port numbers were not set to their default values. This issue is now fixed. For Cloudera Runtime versions 7.3.2 or higher, Cloudera Manager correctly passes the DataNode and block client ports configured for Ozone Storage Container Managers to Ozone DataNodes and Ozone Managers, respectively. Custom ports can now be used without issues. No other action is required. For Cloudera Runtime versions lower than 7.3.2, the functionality is available through manual configuration in the ozone-site.xml safety valve parameters, as before.

**OPSAPS-75236: Excessive INFO-level logs were printed during Ozone CLI operations**

7.13.2.0

Previously, excessive INFO-level logs were printed during Ozone CLI operations and key write paths, primarily from proxy initialization and TLS configuration. This issue is now fixed by changing the log4j.rootLogger in Ozone to the correct value.

**OPSAPS-73164: Ozone's upgrade handlers were not properly added to the UpgradeHandlerRegistry**

7.13.2.0

Previously, Ozone upgrade handlers were not properly applied in certain Cloudera upgrade scenarios. This issue is fixed now.

**OPSAPS-72718: The dn-container.log is not collected in the diag bundle**

7.13.2.0

Previously, the Ozone diagnostic (diag) bundle did not collect the dn-container.log file. This issue is fixed now.

**OPSAPS-73304: Ozone Prometheus port conflict on freshly installed cluster**

7.13.2.0

Previously, the Ozone Prometheus WebUI's default port (9094) conflicted with the Kafka Broker Load Balancer Listener Port on freshly installed clusters. This issue is now fixed, and the Ozone Prometheus WebUI's default port has been changed from 9094 to 9096.

**OPSAPS-71329: The testDuplicateAndSnapshotClasses check failed**

7.13.2.0

Previously, the testDuplicateAndSnapshotClasses check failed due to the presence of the ByteBuff erPositionedReadable class in Ozone. This issue is fixed, and the ByteBufferPositionedReadable has been added to the exclude list.

**OPSAPS-71561: Ozone canary does not handle S3 secret getting revoked**

7.13.2.0

Previously, the canary did not detect when S3 credentials became invalid, resulting in repeated failures with access ID not found errors. This issue has been fixed, and if the S3 secret used by the canary is revoked or becomes invalid, the canary automatically generates and stores a new set of credentials for future runs, restoring normal operation without manual intervention.

**OPSAPS-71897: Finalize Upgrade command fails post-upgrade with `CustomKerberos setup: causing INTERNAL_ERROR with EC writes`**

7.13.2.0

Previously, finalizing an Ozone upgrade failed when a custom Kerberos principal was set to Ozone's Storage Container Manager. This issue has been fixed.

**OPSAPS-73078: Cloudera Manager is not referring to the S3 Gateway TLS enable configuration to start the S3 Gateway with secure or insecure ports**

7.13.2.0

Previously, the Ozone services started on their secure ports even though their ozone.ssl.enabled flag had been set to false. The Recon namespace du command works correctly. This issue is fixed, and the Ozone services now start on their respective secure or insecure ports correctly based on their ozone.ssl.enabled flag. The Recon namespace du command continues to work correctly.

**OPSAPS-73383: SCM principal is hardcoded in the Ozone Manager**

7.13.2.0

Ozone upgrade finalization no longer requires a Kerberos principal with "scm" as the principal's short name. This change removes the previous limitation, allowing the use of custom Kerberos principals for Cloudera Object Store powered by Apache Ozone.

**OPSAPS-71342: Setting hdds.x509.max.duration to 0 shuts down Storage Container Manager, DataNodes, and Ozone Manager**

7.13.2.0

Previously, configuring the hdds.x509.max.duration parameter to 0 or any negative value caused the Storage Container Manager (SCM), DataNodes (DNs), and Ozone Manager (OM) services to shut down, resulting in cluster-wide disruption. This issue has been fixed by adding a validator Ozon eMaxCertDurationValidator that ensures the hdds.x509.max.duration value is greater than zero and follows the ISO-8601 duration format.

**OPSAPS-76539: Cloudera Manager UI allowed adding multiple Spark 3 service instances within a single cluster**

Prevented multiple Spark 3 service instances within a single cluster by implementing a maxInsta nces flag in Cloudera Manager.

**OPSAPS-72316: Knox Gateway might crash when serving Hive and Impala clients under heavy load**

Performance issues with the PAM module affected Knox. Specifically, under heavy load, interaction with the libpam.so module might crash Knox Gateway.

This issue was fixed by adding PAM authentication caching to mitigate PAM module crashing under heavy load.

**OPSAPS-75616: Logger safety valves for Cloudera 7.1.9 were incorrect**

Logger safety valves for Cloudera 7.1.9 did not apply correctly for Knox.

Cloudera Manager now applies logger safety valves correctly for Cloudera 7.1.9.

**OPSAPS-74281: Disabled the live Spark UI when the ENCRYPT_ALL_PORTS feature flag is enabled**

Enhanced the security of Spark by implementing new default configuration settings:

1. spark.ui.enabled=false: preventing initiation of an HTTP service that can be accessed from external hosts.
2. spark.io.encryption.keySizeBits=256: the default Spark keySizeBits has been increased from 128 to 256

**OPSAPS-74346, OPSAPS-74346, OPSAPS-74316: Enhancing Spark security**

Changed the generation of the spark.yarn.historyServer.address value to use the HTTPS address when SSL/TLS is enabled. The spark3.network.crypto.enabled new configuration property is now available to enable AES-based encryption.

**OPSAPS-72254: UCL | FIPS Failed to upload Spark example jar to HDFS in cluster mode**

Fixed an issue with deploying the Spark 3 Client Advanced Configuration Snippet (Safety Valve) for spark3-conf/spark-env.sh.

For more information, see *New Cloudera Manager configuration parameter spark_pyspark_executable_path has been added to the Livy for Spark 3* in Behavioral Changes In Cloudera Manager 7.13.2.

**OPSAPS-75290, OPSAPS-74994: The yarn_enable_container_usage_aggregation job is failing with "Null real user" error on Service Monitor.**

The yarn_enable_container_usage_aggregation job is failing with "Null real user" error on Service Mnitor when the Yarn service is running on the computer cluster with Stub DFS, and when the Powerscale Service is running in the cluster with Powerscale DFS provider instead of HDFS.

To mitigate this error, Cloudera introduced the "`DFS User to Impersonate` (template name: `dfs_user_to_impersonate`)" configuration.

You must set the "`DFS User to Impersonate`" configuration to "hdfs" (recommended) or the respective File System user to resolve the **impersonation user** issue in Service Monitor.

**Important:** You must do this step, before you begin the Enable the Cluster Utilization Report task.

**OPSAPS-73372: hbase-env.sh is incorrectly copied without variable substitution to dependent projects**

7.3.2.0

The hbase-env.sh file, part of the HBase client configuration, is generated into /etc/conf/hbase and hbase-conf/hbase-env.sh in the dependent services' process directory. While variable substitution works correctly for /etc/hbase/conf/hbase-env.sh, it does not happen, at least for Omid.

This issue is fixed now. Phoenix and Omid now load environment variables from hbase-env.sh and incorporate the options specified in PHOENIX_OPTS; alternatively, if PHOENIX_OPTS is undefined, they utilize the options from HBASE_OPTS.

**OPSAPS-74862: Unable to set HBase RPC mTLS key for clients in Cloudera Manager**

7.3.2.0

If client-side TLS for HBase RPC is enabled, the server mTLS setting is also set to NEED by default, requiring authentication from the client. This setting is defined by `hbase.server.netty.tls.client.auth.mode`. However, Cloudera Manager does

not add a client mTLS key to the client HBase configuration, so HBase clients using the gateway configuration do not work by default.

This issue is fixed now. HBase Client mTLS setting is now disabled by default to allow clients to connect remotely without presenting a valid certificate during the TLS handshake. This makes it easier for clients to establish encrypted connectivity.

**OPSAPS-76258: The Deploy client configuration and refresh operation fails after a CDH upgrade**

7.3.2.0

After upgrading Cloudera Manager to version 7.13.2 and CDH to version 7.3.2-1 with OpenJDK 17.0.11, the cluster fails to complete the deploy client configuration and refresh operation. Although the deploy client configuration step succeeds, the refresh step fails.

This issue is fixed now. The system is updated to resolve a Null Pointer Exception (NPE) that occurs while running the refresh configuration command. You can run the deploy client configuration and refresh command as usual.

**OPSAPS-71576: Default value for fe_service_threads increased to improve concurrency**

The default value for the fe_service_threads setting was 64. Starting with Cloudera Runtime 7.13.2, the default value is 128.

This has now been fixed.

**OPSAPS-74019/OPSAPS-72739: Query execution stability with temporary directories**

Queries failed with an execution error when using a compression library. This happened because the system attempted to use /tmp as a temporary folder for script execution, which was not permitted by default for this library, leading to query failures.

This issue was resolved by configuring Hive to use a different default temporary folder, /var/lib/hive, instead of /tmp.

**OPSAPS-74044: Setting the catalog topic mode when disabling the local catalog**

Previously, unchecking the local_catalog_enabled checkbox in the Impala configuration page did not correctly trigger the necessary evaluators to set the catalog topic mode to full or disable the local catalog in `impalad`.

This issue is resolved by adding an evaluator that takes effect when the local_catalog_enabled checkbox is unchecked. This action now correctly sets --catalog_topic_mode=full for `catalogd` and `impalad`, and --use_local_catalog=false for `impalad`.

**OPSAPS-72905: Missing MemoryUsage counter in Impala Query Profile**

Previously, the MemoryUsage counter was missing from the Impala Query Profile in Cloudera Manager. This issue caused the memory_aggregate_peak metric to display incorrect values.

This issue is resolved by including the MemoryUsage counter in the Cloudera Manager Impala Query Profile. This ensures that the memory_aggregate_peak and memory_accrual metrics provide accurate data.

**OPSAPS-73880: Impala thrift definition update**

Previously, the Thrift files under if/impala/ were outdated, which could lead to compatibility issues with newer versions of Impala.

This issue is resolved by updating the Thrift files.

**OPSAPS-74044: Setting the catalog topic mode when disabling local catalog**

Previously, unchecking the local_catalog_enabled checkbox did not correctly trigger the necessary evaluators to set the catalog topic mode to full and disable the local catalog in `impalad`.

This issue is resolved by adding an evaluator that takes effect when the local_catalog_enabled checkbox is unchecked. For Cloudera Runtime 7.3.1 and higher, this action now sets --catalog_topic_mode=full for `catalogd` and `impalad`, and --use_local_catalog=false for `impalad`.

**OPSAPS-76290: HMS Metastore schema setup timeout**

Previously, the Create Hive Metastore database tables command frequently failed because the context preparation consumed most of the allocated 150-second timeout, leaving insufficient time for schema initialization.

This issue is resolved by increasing the default timeout for this task to 600 seconds to ensure successful and clean execution.

**OPSAPS-72998: Missing Hive Metastore event API charts**

Previously, charts for Hive Metastore (HMS) event APIs, including get_next_notification, get_current_notificationEventId, and fire_listener_event, were missing from the Cloudera Manager Charts Library.

This issue is now resolved by adding new charts for HMS-related metrics and connection pools to the user interface.

**OPSAPS-72930: Tez client configuration during upgrade**

Previously, the Tez client configuration was not automatically deployed during the upgrade process.

This issue is now fixed by including the Tez client configuration deployment as a step in the upgrade process.

**OPSAPS-60161: Hive Metastore canary test failures**

Previously, the cloudera_manager_metastore_canary_test failed in environments with multiple Hive Metastore (HMS) nodes.

This issue is now fixed by adding an ifExists field to the catalog drop request, ensuring the process completes successfully even if the catalog was already removed.

**Apache Jira**: HIVE-28443

**OPSAPS-75843: Hive external table replication fails when Zookeeper has a non-default service name**

Previously, Hive external table replication policies failed when Zookeeper was configured with a customized principal or non-default service name. This issue is now fixed. You can successfully use a customized principal by adding the -Dzookeeper.sasl.client.username = *[*** ADD CUSTOMIZED PRINCIPAL *** ]* key-value pair in the Cloudera Manager Clusters Hive service Configuration Client Java Configuration Options (hive_client_java_opts) property.

**OPSAPS-70834: Multiple instances of Atlas replication policy are running at the same time**

Previously, multiple instances of an Atlas replication policy were running at the same time, which was incorrect. This issue is now fixed.

**OPSAPS-70681: Atlas client configuration at policy level**

Previously, you could set Atlas client-related properties at the cluster level which was not efficient. This issue is now fixed. You can now configure these properties at replication policy level using the Cloudera Manager API. For example, you can set the following properties:

```
"atlasClientAdvanceConfigs": {
"atlas.client.connectTimeoutMSecs": "12345",
"atlas.client.readTimeoutMSecs": "12345" }
```

**OPSAPS-70713: Error is displayed when running Atlas replication policy if source or target clusters use Dell EMC Isilon storage**

Previously, you could not create an Atlas replication policy between clusters if one or both the clusters used Dell EMC Isilon storage. This issue is now fixed.

**OPSAPS-71220: Replication History page displays incorrect status for Atlas replication**

Previously, when you ran Hive external table or Iceberg replication policies that included replicating Atlas metadata (also called composite replication), the **Replication Policies** page displayed success even if one of the replications failed. For example, if during the Iceberg replication policy run, the Atlas metadata replication failed, the page displayed the **Successful** status, which was incorrect. This issue is now fixed.

**OPSAPS-75080, OPSAPS-75125: Replication policies history page displays half the count of history than expected for composite replication**

Previously, the **Replication Policy History** page for a composite replication policy displayed half the number of job runs. The composite replication policies include Hive external table or Iceberg replication policies that also migrated Atlas metadata. This issue is now fixed. The page displays all the job runs.

**OPSAPS-74864: Iceberg composite replication policy displays all the options in the history list**

Previously, during an Iceberg composite replication policy job run, when Atlas replication failed but Iceberg replication continued, the **Replication Policies** page displayed all the available options in the **History** list, which was incorrect. This issue is now fixed.

**OPSAPS-76077, OPSAPS-75926: Hive external metadata-only replication of Ozone backed tables fails for virtual views**

Previously, Hive on Ozone external metadata replication failed if the input regex matched any virtual views. This issue is now fixed. The virtual views are replicated by default.

If you do not want to replicate the virtual views, add the DISALLOW_VIRTUAL_VIEWS_FOR_OZONE=true key-value pair in the Cloudera Manager Clusters Hive service Configuration hive_replication_env_safety_valve property.

**OPSAPS-73218, OPSAPS-73219: Dry run for Ozone replication policies does not work as expected**

Previously, the **Dry Run** action for Ozone replication policies failed and led to data loss. This issue is now fixed. The **Dry Run** action is no longer available for Ozone replication policies when the Listing type is Incremental only or Incremental with fallback to full file listing.

**OPSAPS-71067: Wrong interval sent from the Replication Manager UI after Ozone replication policy submit or edit process**

Previously, when you edited the existing Ozone replication policies, the schedule frequency changed unexpectedly. This issue is now fixed.

**OPSAPS-74203: Incorrect parameters are displayed for HBase Snapshot operations**

Previously, incorrect parameters were displayed for HBase Snapshot operations on the **Snapshot Policies** page and in Cloudera Manager Server logs. The UI now properly interpolates the tableName and snapshotName into i18n message strings to display the correct parameters.

**OPSAPS-70822: Hive external table replication policy could not be saved on the 'Edit Hive External Table Replication Policy' window**

Previously, Replication Manager did not save the changes as expected when you clicked Save Policy after you edited a Hive replication policy using the Actions Edit Configuration option for the replication policy on the **Replication Policies** page. This issue is fixed.

**OPSAPS-72276: Cannot edit Ozone replication policy if the MapReduce service is stale**

Previously, you could not edit an Ozone replication policy in Replication Manager if the MapReduce service did not load completely. This issue is fixed.

**OPSAPS-71596, OPSAPS-69782: Exception appears if the peer Cloudera Manager's API version is higher than the local cluster's API version**

HBase replication using HBase replication policies in CDP Public Cloud Replication Manager between two Data Hubs/COD clusters succeed as expected when all the following conditions are true:

- The destination Data Hub/COD cluster's Cloudera Manager version is 7.9.0-h7 through 7.9.0-h9 or 7.11.0-h2 through 7.11.0-h4, or 7.12.0.0.
- The source Data Hub/COD cluster's Cloudera Manager major version is higher than the destination cluster's Cloudera Manager major version.
- The Initial Snapshot option is chosen during the HBase replication policy creation process and/or the source cluster is already participating in another HBase replication setup as a source or destination with a third cluster.

**OPSAPS-71424: The 'configuration sanity check' step ignores the replication advanced configuration snippet values during the Ozone replication policy job run**

Previously, the OBS-to-OBS Ozone replication policy jobs failed when the S3 property values for fs.s3a.endpoint, fs.s3a.secret.key, and fs.s3a.access.key were empty in Ozone Service Advanced Configuration Snippet (Safety Valve) for ozone-conf/ozone-site.xml even when these properties were defined in Ozone Replication Advanced Configuration Snippet (Safety Valve) for core-site.xml. This issue is fixed.

**OPSAPS-75136, OPSAPS-75187, OPSAPS-75245, OPSAPS-75449: Kerberos ticket validation fails during HDFS replication**

Previously, Kerberos ticket validation failed during the HDFS replication policy run. This issue is now fixed because Kerberos ticket validation now checks the current cached tickets by utilizing the Kerby Credential Cache. This improvement also prevents a round-trip authentication request to the Key Distribution Center (KDC).

**OPSAPS-74314, OPSAPS-74636: HBase snapshot export always runs with the default client configuration**

Previously, when multiple HBase services existed in a cluster, the HBase export process used the default client configuration. This issue is now resolved because the export process prioritizes the correct HBase replication client configurations based on the set CLASSPATH value in the snapshot-hbase.sh file.

**OPSAPS-73217, OPSAPS-74665, OPSAPS-75303, OPSAPS-75444: Snapshot retention after incremental Ozone replication dry run**

Previously, the dry run process for the incremental Ozone replication policy did not delete the snapshot it created after the replication process was complete. This issue is now fixed. For information about this issue, see the corresponding Knowledge article: Technical Service Bulletin 2025-835: Dry run of incremental Ozone replication can cause failure to replicate some changes in Cloudera Replication Manager.

**OPSAPS-73138, OPSAPS-72435: Ozone OBS-to-OBS replication policies created incorrect directories in the target cluster**

Ozone OBS-to-OBS replication policies created incorrect directories in the target cluster even when no such directories existed on the source cluster. This issue is now resolved.

**OPSAPS-72447, CDPD-76705: Ozone incremental replication fails to copy renamed directory**

Ozone incremental replication using Ozone replication policies succeed but might fail to synchronize the nested renames for FSO buckets. When a directory and its contents are renamed between the replication runs, the outer level rename synced but did not synchronize the contents with the previous name. This issue is fixed now.

**OPSAPS-74082: Ozone FSO to FSO replication failed on link buckets**

Previously, the Ozone replication policies for FSO to FSO buckets failed for link buckets if the link bucket was not in the s3v volume. This issue is now resolved.

**OPSAPS-74040: Ozone OBS replication fails due to pre-filelisting check failure**

During OBS-to-OBS Ozone replication, if the source bucket is a linked bucket, the replication failed during the \Run Pre-Filelisting Check step, and the Source bucket is a linked bucket, however the bucket it points to is also a link error message appeared, even when the source bucket directly links to a regular, non-linked bucket. The issue is now fixed.

The Ozone OBS-to-OBS replication no longer fails when the source or the target bucket is a linked bucket because the linked bucket resides in the s3v volume and refers to another bucket in **s3v** or any other volume.

**OPSAPS-73906, OPSAPS-73737, OPSAPS-73655, OPSAPS-74061: Cloud replication no longer fails after the delegation token is issued**

Previously, the replication policies were failing during incremental replication job runs if you chose the Advanced Setting Delete Policy Delete permanently option during the replication

policy creation process. You can now configure com.cloudera.enterprise.distcp.skip-delegation-token-on-cloud-replication to false in the Cloudera Manager Clusters HDFS service Configuration HDFS Replication Advanced Configuration Snippet (Safety Valve) for core-site.xml advanced configuration snippet to ensure that the HDFS and Hive external table replication policies replicating from an on-premises cluster to cloud do not fail. When the advanced configuration snippet is set to false, the MapReduce client process obtains the delegation tokens explicitly before it submits the MapReduce job for the replication policy. By default, the advanced configuration snippet is set to true.

**OPSAPS-73142: The required configuration from replication safety valve is not accessed**

An Ozone replication policy with Incremental with fallback to full file listing option failed with Pre-Filelisting Check Failed with Error: target bucket has layout OBS, but [fs.s3a.endpoint, fs.s3a.secret.key, fs.s3a.access.key] properties are missing from the target Ozone service core-site.xml config error because the required configuration was not available in the required folders. To mitigate this issue, the required configuration parameters are now added automatically to the required folders during the Ozone replication policy run.

**OPSAPS-72756: The runOzoneCommand API endpoint fails during the Ozone replication policy run**

The `/clusters/{clusterName}/runOzoneCommand` Cloudera Manager API endpoint fails when the API is called with the `getOzoneBucketInfo` command. In this scenario, the Ozone replication policy runs also fail if the following conditions are true:

- The source Cloudera Manager version is 7.11.3 CHF11 or 7.11.3 CHF12.
- The target Cloudera Manager is version 7.11.3 through 7.11.3 CHF10 or 7.13.0.0 or later where the feature flag API_OZONE_REPLICATION_USING_PROXY_USER is disabled.

This issue is fixed now.

**OPSAPS-72468: Subsequent Ozone OBS-to-OBS replication policy do not skip replicated files during replication**

Replication Manager now skips the replicated files during subsequent Ozone replication policy runs after you add the following key-value pairs in Cloudera Manager Clusters Ozone service Configuration Ozone Replication Advanced Configuration Snippet (Safety Valve) for core-site.xml :

- com.cloudera.enterprise.distcp.ozone-schedules-with-unsafe-equality-check = *[***ENTER COMMA-SEPARATED LIST OF OZONE REPLICATION POLICIES' ID OR ENTER ALL TO APPLY TO ALL OZONE REPLICATION POLICIES***]* –- The advanced snippet skips the already replicated files when the relative file path, file name, and file size are equal and ignores the modification times.

  

  **Caution:** Usage of this advanced snippet might lead to data loss. For example, if you modified a file on the source or target cluster and the file size remains the same, the advanced snippet ignores the file during the replication run.
- com.cloudera.enterprise.distcp.require-source-before-target-modtime-in-unsafe-equality-check = *[***ENTER TRUE OR FALSE***]* –- When you add both the key-value pairs, the subsequent Ozone replication policy runs skip replicating files when the matching file on the target has the same relative file path, file name, file size and the source file's modification time is less or equal to the target file modification time.

**OPSAPS-67498: The Replication Policies page takes a long time to load**

Previously, the Cloudera Manager Replication Manager Replication Policies took a long time to load. This issue is resolved.

To ensure the page loads faster, new query parameters have been added to the internal policies that fetch the REST APIs for the page which improves pagination. Replication Manager also caches internal API responses to speed up the page load.

**OPSAPS-69622: Cannot view the correct number of files copied for Ozone replication policies**

The last run of an Ozone replication policy does not show the correct number of the files copied during the policy run when you load the Cloudera Manager Replication Manager Replication Policies page after the Ozone replication policy run completes successfully. This issue is fixed now.

**OPSAPS-70848: Hive external table replication policies succeed when the source cluster uses Dell EMC Isilon storage**

During the Hive external table replication policy run, the replication policy failed at the Hive Rep lication Export step. This issue is fixed now.

**OPSAPS-70909: Use specified users instead of "hive" for Ozone replication-related commands**

Starting from Cloudera Manager 7.11.3 CHF15, Ozone commands executed by Ozone replication policies are run by impersonating the users that you specify in the Run as Username and Run on Peer as Username fields in the **Create Ozone replication policy** wizard. The bucket access for OBS-to-OBS replication depends on the user with the access key specified in the fs.s3a.access.key property. When the source and target clusters are secure, and Ranger is enabled for Ozone, specific permissions are required for Ozone replication to replicate Ozone data using Ozone replication policies.

**OPSAPS-71093: Validation on source for Ranger replication policy fails**

The Cloudera Manager page would be logged out automatically when you created a Ranger replication policy. This is because the source cluster did not support the `getUsersFromRanger` or `getPoliciesFromRanger` API requests. The issue is fixed now. The required validation on the source completes successfully as expected.

**OPSAPS-72559: Incorrect error messages appear for Hive ACID replication policies**

Replication Manager now shows correct error messages for every Hive ACID replication policy run on the Cloudera Manager Replication Manager Replication Policies Actions Show History page as expected.

**OPSAPS-71544, OPSAPS-75166, OPSAPS-75182: Ranger replication policies failed for custom username**

Previously, when you used a custom username or Kerberos principal in the Ranger replication policy, the policy failed during the transformation step if the custom Ranger process user was set in Cloudera Manager. This issue is now fixed.

**OPSAPS-72509: Hive metadata transfer to GCS fails with ClassNotFoundException**

Hive external table replication policies from an on-premises cluster to cloud failed during the Tran sfer Metadata Files step when the target is on Google Cloud and the source Cloudera Manager version is 7.11.3 CHF7, 7.11.3 CHF8, 7.11.3 CHF9, 7.11.3 CHF9.1, 7.11.3 CHF10, or 7.11.3 CHF11. This issue is fixed.

**OPSAPS-72446, OPSAPS-71565, OPSAPS-71566, OPSAPS-73405,OPSAPS-72860: Replication policy runs when the source or target cluster becomes available after it recovers from temporary node failures**

Hive replication policies and HBase replication policies can now recover from a temporary node failure on the source or target clusters to continue the replication policy job run. Alternatively, you can also rerun the failed or aborted policies manually. To ensure that the RemoteCmdWork daemon continues to poll even in case of network failures or if the Cloudera Manager goes down, you can set the remote_cmd_network_failure_max_poll_count = *[*** ENTER REMOTE EXECUTOR MAX POLL COUNT***]* parameter on the target Cloudera Manager Administration Settings page.

The actual timeout is provided by a piecewise constant function that is a step function with the following breakpoints: 1 through 11 is 5 seconds, 12 through 17 is 1 minute, 18 through 35 is 2 minutes, 36 through 53 is 5 minutes, 54 through 74 is 8 minutes, 75 through 104 is 15 minutes, and so on. Therefore when you enter 1, the polling continues for 5 seconds after the Cloudera Manager goes down or after a network failure. Similarly when you set it to 75, the polling continues for 15 minutes.

To ensure Replication Manager attempts to recover the RemoteCmdWork daemon on the target cluster, ensure that you set the retry value in the target Cloudera Manager Administration Settings remote_cmd_max_recovery_count parameter, or set it to 0 to turn off the feature. By default,

Replication Manager attempts to recover the command twice after the target cluster goes down temporarily. This issue is now fixed.

**OPSAPS-74279, OPSAPS-72439, OPSAPS-74265: HDFS and Hive external tables replication policies failed when using custom krb5.conf files**

HDFS and Hive external tables replication policies failed when using custom krb5.conf files. This is because the custom krb5.conf was not propagated to the required files. To mitigate this issue, complete the instructions provided in Step 13 in Using a custom Kerberos configuration path.

**OPSAPS-72978: The getUsersFromRanger API parameter truncates the user list after 200 items**

The Cloudera Manager API endpoint `v58/clusters/[***CLUSTER***]/services/[***SERVICE***]/commands/getUsersFromRanger` API endpoint no longer truncates the list of returned users at 200 items.

**OPSAPS-73602, OPSAPS-74360: HDFS replication policies to cloud failed with HTTP 400 error**

The HDFS replication policies to cloud were failing after you edited the replication policies in the Cloudera Manager Replication Manager UI . This issue is fixed.

**OPSAPS-72804: For recurring policies, the interval is overwritten to 1 after the replication policy is edited**

Previously, when you edited an Atlas, Iceberg, Ozone, or a Ranger replication policy that had a recurring schedule on the **Replication Manager UI**, the **Edit Replication Policy** modal window appeared as expected. However, the frequency of the policy was reset to run at 1 unit where the unit depended on what you configured in the replication policy. For example, if you configured the replication policy to run every four hours, it was reset to one hour when you edited the replication policy. This issue is fixed.

**OPSAPS-72214: Cannot create a Ranger replication policy if the source and target cluster names are not the same**

You could not create a Ranger replication policy if the source cluster and target cluster names were not the same. This issue is fixed.

**OPSAPS-71853: The Replication Policies page does not load the replication policies' history**

When the sourceService is null for a Hive ACID replication policy, the Cloudera Manager Replication Manager UI failed to load the existing replication policies' history details and the current state of the replication policies on the **Replication Policies** page. This issue is now fixed.

**OPSAPS-71256: The "Create Ranger replication policy" action shows 'TypeError' if no peer exists**

When you clicked target Cloudera Manager Replication Manager Replication Policies Create Replication Policy Ranger replication policy option, the TypeError: Cannot read properties of undefined error appeared. This issue is fixed now.

**OPSAPS-71459: Commands continue to run after Cloudera Manager restart**

Some remote replication commands continue to run endlessly even after a Cloudera Manager restart operation. This issue is fixed

**OPSAPS-72573: Monitoring for Kudu tablet sizes and replica counts**

Previously, Cloudera Manager lacked integrated monitoring for Kudu tablet sizes and replica counts, making it difficult to track on-disk footprints or identify excessively large tablets.

This issue is now resolved. A new health test has been added to monitor the number of tablet replicas per server. If a server exceeds 2000 tablet replicas, its health status now automatically changes to a WARN state.

**OPSAPS-75602: Issue with RANGER_C719 CSD becoming stale after upgrading Cloudera Manager**

Fixed an issue where the RANGER_C719 CSD could become stale after upgrading Cloudera Manager from 7.13.1.600 with Cloudera 7.1.9 to 7.13.2.0 by fixing the following:

- OPSAPS-73498: Added Cloudera Manager side ranger-trino integration changes.

- OPSAPS-73152: Improved Ranger Admin Diagnostic collection command from Cloudera Manager scripts.

**OPSAPS-75556: After upgrade from 7.1.9 to 7.3.2.0 dataset field type is set to boolean in solr managed-schema**

Fixed an issue where, after upgrading from Cloudera 7.1.9 to 7.3.2, the datasets field in the ranger_audits Solr collection schema was incorrectly set to the boolean type instead of key_lower_case with multiValued="true". This schema mismatch caused Ranger Admin to fail to load the Access Audit page on upgraded clusters. The upgrade process now updates the ranger_audits Solr schema so that the datasets field is created with the correct type and behaves consistently with fresh 7.3.2 deployments.

**OPSAPS-71619: Removed the mandatory validation for ranger.ldap.user.dnpattern**

Previously, when LDAP was configured as the external authentication type for Ranger Admin, the ranger.ldap.user.dnpattern parameter was mandatory. If it was not set, the Ranger Admin service failed to start, even though this parameter is rarely required and is ignored when LDAP bind DN/password and user search parameters are configured. This has been fixed by removing the mandatory validation for ranger.ldap.user.dnpattern, so the parameter is now optional and the service can start without requiring a dummy value.

**OPSAPS-69156: Fixed an issue with Java add-opens/add-modules/add-exports options**

Cloudera Manager components now consistently use the --add-opens=, --add-modules=, and --add-exports= syntax for Java options. This avoids cases where options passed via JAVA_TOO L_OPTIONS could be rejected (for example when using --add-opens or --add-exports without =), improving compatibility across different Java runtimes.

**OPSAPS-67197: Ranger RMS server shows as healthy without service being accessible**

Previously, Cloudera Manager reported the Ranger RMS server as healthy based only on the RMS process (PID), even when the RMS web service was not fully initialized and the service was inaccessible. The health check logic has been updated to use a Cloudera Manager web alert that verifies the Ranger RMS RMS web endpoint instead of relying solely on the PID. This allows Cloudera Manager to more accurately detect when RMS is not accessible and helps users identify RMS availability issues faster.

**OPSAPS-72766: Ranger KMS tomcat context update**

Updated the default Tomcat context for Ranger KMS from /kms to / by changing the ranger.contextName property in ranger-kms-site.xml. This aligns the Ranger KMS context path with Cloudera configuration and simplifies access and integration.

**OPSAPS-74083: JAAS configuration for Ranger services connecting to ZooKeeper with strict SASL enforcement**

When ZooKeeper was configured with strict SASL enforcement, Ranger Admin, Ranger Tagsync, and Ranger RAZ could not establish SASL-secured connections because no JAAS configuration was defined. This has been fixed by introducing a dedicated JAAS configuration file for these Ranger services and adding a Java option to reference this file, enabling successful SASL authentication with ZooKeeper.

**OPSAPS-74063: RANGER_RMS CSD changes for supporting multiple storage types**

Fixed an issue in the Ranger RMS CSD configuration for the 7.13.2.0 release to support multiple storage types by adding S3- and Ozone-specific HMS source service properties and updating the supported URI schemes and default HMS source service type.

**OPSAPS-74517: Service users denied access to Kafka topics**

Service users were denied access to internal Kafka Connect topics (connect-status, connect-secrets, connect-offsets, and connect-configs), generating a large number of access-denied audit entries for the streamsrepmgr and atlas service users. The default “connect internal - topic” policy for Kafka has been updated to include these service users, ensuring they can access the required internal topics and preventing further denied-access audit noise.

**OPSAPS-72249: Oozie database dump fails on JDK 17**

Previously, the Oozie database dump and load commands could not be executed from Cloudera Manager when using JDK 17. This issue is fixed now.

**OPSAPS-72767: The Install Oozie ShareLib command fails on FIPS and FedRAMP clusters**

Previously, the Install Oozie ShareLib command could not be executed on FIPS and FedRAMP clusters. This issue is fixed now.

**OPSAPS-75667: Oozie failed to start due to an insufficient minimum Java heap size setting**

Previously, the minimum heap size for Oozie was set to 256 MB, which could lead to out-of-memory errors during startup. This issue is fixed now, and the minimum heap size has now been increased to 1 GB to ensure reliable Oozie service startup and operation.

**OPSAPS-70948: The HTTPFS java option parameters set through Cloudera Manager are not being picked up**

Previously, the HDFS HTTPFS did not get all Java related options from Cloudera Manager. This issue is fixed now.

**OPSAPS-75733: Services are not enabled for Oozie on PostUpgrade**

During upgrades from Cloudera Manager 7.1.x to 7.3.x, the removal of 7.2.x upgrade handlers (as part of OPSAPS-74572) caused required service dependencies for Oozie to be unset. This issue is fixed, and restores the necessary dependency setup for Oozie during upgrades to 7.3.x, ensuring that all required services are properly enabled and configured post-upgrade.

# Known Issues

You must be aware of the known issues and limitations, the areas of impact, and workaround in Cloudera Manager 7.13.2 and its cumulative hotfixes.

## Known issues identified in Cloudera Manager 7.13.2

**OPSAPS-77052: Ozone DataNode decommission command stuck for more than 4 hours**

7.13.2

Ozone DataNode decommissioning can appear stuck in Cloudera Manager while the actual decommissioning is successful. This occurs due to a bug in the monitoring script, where a loose grep expression causes the script to wait indefinitely.

Administrators can manually monitor the DataNode decommission state using the Storage Container Manager (SCM) Web UI or CLI. Once all desired DataNodes are confirmed as decommissioned, the decommission command in Cloudera Manager can be safely aborted.

**OPSAPS-76455: The Stop command failed on Ozone S3 Gateway service**

7.13.2

When a restart is performed on the Ozone S3 Gateway, the java.lang.IllegalStateException: Sin gleton not set for STATIC_INSTANCE exception occurs. This exception originates from the JBoss Weld bootstrap process when the CDI registry fails to reinitialize the static singleton provider correctly during the restart cycle.

Avoid the restart operation. Instead, perform a manual stop followed by a start.

**OPSAPS-76062: Ozone Replication Configuration Override**

7.13.2

When the ozone.replication property is exposed in Cloudera Manager Ozone configurations and assigned a default value, it unintentionally overrides the bucket-level replication configuration as a client-side setting—even if the user does not intend to set client-side configurations.

None

**OPSAPS-76845: Last Page button on the Bucket Browser tab is not functional**

7.13.2

In Cloudera Manager UI, the Last Page button in the Bucket Browser tab does not function as expected. When users click the Last Page button, the UI refreshes the current page instead of navigating to the last page of buckets. The Next button continues to work as intended, allowing users to move forward one page at a time. This issue is particularly noticeable when there are a large number of buckets, as users must navigate page by page to reach the end.

There is no direct workaround to enable the Last Page button. However, to reduce the number of navigation steps, users can increase the number of buckets displayed per page in the UI settings. This adjustment allows more buckets to be viewed at once, minimizing the number of pages to navigate.

**OPSAPS-74844: Service Monitor fails to connect to multiple clusters with distinct custom Kerberos principals**

7.13.2

The Cloudera Manager Service Monitor does not support unique Kerberos principal configurations across multiple clusters. The following limitations apply to the Service Monitor:

- You cannot apply different custom Kerberos settings to different clusters managed by the same Service Monitor.
- You cannot connect to multiple clusters simultaneously if those clusters require distinct custom Kerberos principals.

None

**OPSAPS-76330: Missing request/process context ID in Atlas logs after migration from log4j2 to logback using CM default pattern**

7.13.2

After upgrading Apache Atlas logging from log4j2 to logback, and using the default logback pattern provided by Cloudera Manager, Atlas logs no longer consistently include the request/process context ID (for example, etp<timestamp>-<pid> - <uuid>). This results in a regression compared to earlier behavior with log4j2, where the context ID was consistently present for request-scoped operations. The missing context information makes troubleshooting and tracing individual requests more difficult.

None

**OPSAPS-75366: The Knox Gateway gateway.log.gz file in the support bundle is corrupt**

7.13.2

When you collect diagnostic data, the Knox Gateway gateway.log.gz file under logs/*[***HOST NAME***]*/ in the downloaded bundle might have 0-byte length. The file does not contain the Knox Gateway logs even when gateway.log on the host has content.

In the diagnostic bundle, open the service-diagnostics/*[***CLUSTER NAME***]*/*[***KNOX SERVICE NAME***]*/ folder. The Knox Gateway role diagnostics archive there includes the gateway logs (for example gateway.log and gateway-audit.log).

**OPSAPS-75684: Spark fails due to Zookeeper Custom Kerberos Principal issue**

Incorrect Zookeeper principal configuration and missing JVM property setup leads to SASL authentication failures.

When a custom Zookeeper principal is used, add the -Dzookeeper.sasl.client.username=[***USERNAME***] JVM argument to spark.*.defaultJavaOptions or spark.*.extraJavaOptions in spark-defaults.conf.

**OPSAPS-75443: Hive Metastore Server fails to start after memory reallocation**

7.13.2

After executing the `/api/v57/hosts/reallocateMemory` API, the Hive Metastore Server (HMS) might fail to start with a "Not enough space" memory error. This issue occurs even after the heap size is set to 8GB and typically appears following an Atlas Server memory failure. The HMS service remains in a stopped state because it cannot allocate the required memory resources.

None

**OPSAPS-76683: Hive system database creation**

7.13.2

The Hive system database creation blocks the upgrade process. When upgrading, the process is interrupted or blocked during the creation of the Hive system database.

None

**OPSAPS-73421: Hive Metastore performance logging**

7.13.2

The performance logger does not function as expected in the Hive Metastore. Performance logging (Perflogger) fails to record entries in the Hive Metastore (HMS) logs, even when the "Enable Performance Logging" flag is enabled in the Hive service configurations. The required logger is not correctly added to the loggers list in the logging properties.

None

**OPSAPS-73237: Hive default heap sizes on Data Hubs**

7.13.2

The default Java heap sizes for Hive Metastore (HMS) and HiveServer2 (HS2) are too large for certain Data Hub configurations. On Data Hub clusters, such as those with a 64 GB environment, the default Java heap size for Hive Metastore and HiveServer2 is automatically configured to 16 GB. This high allocation can lead to memory overcommitment and leave insufficient memory for other cluster processes.

You can manually reduce the Java heap size for Hive services to 8 GB or lower in the Cloudera Manager configuration.

**OPSAPS-75673: Wrong enablement of Ranger RMS Database Full Sync command**

7.1.8, 7.1.9, 7.1.9 SP1, 7.2.18, 7.3.1, 7.3.2.0

The `Ranger RMS Database Full Sync` command should be enabled only when all RMS server instances are stopped. This is required to ensure that the RMS database synchronizes correctly without introducing conflicts or data corruption. However, when HA (High Availability) is enabled on the cluster, the command becomes available from Cloudera Manager Ranger RMS Actions drop-down, even though only one Ranger RMS instance is stopped while the others are still running.

None.

**OPSAPS-73684: Service startup failures during High Availability deployment**

7.13.2

During High Availability (HA) cluster deployments, Impala services can fail to start due to dependent services not having fully started.

Retry of HA deployment might succeed in some cases.

**OPSAPS-76959: Stale alternatives temporary files cause client configuration deployment failure**

7.13.2

An interrupted Cloudera Manager upgrade or Agent restart can leave stale temporary files in the /var/lib/alternatives/ directory. These leftover files prevent subsequent Deploy Client Configuration tasks from completing, as the update-alternatives command fails when it encounters an existing .new state file.

During a Cloudera Manager upgrade or configuration activation, a race condition or a forced service restart (such as a SIGTERM/Exit Code -15) can interrupt the update-alternatives process. This interruption leaves behind a stale temporary file, typically named /var/lib/alternatives/<alt_name>.new.

When you later attempt to Deploy Client Configuration, the Agent tries to run update-alternatives --install. The command fails because the operating system detects the pre-existing .new file and returns a non-zero exit code (usually Exit Code 2). Cloudera Manager then reports a command failure similar to: "client configuration ... exited with 2 and expected 0"

This issue is typically host-specific rather than cluster-wide.

If a deployment fails due to a stale alternatives state, manually clear the temporary files on the affected host and retry the deployment from the Cloudera Manager UI.

1. Verify no active alternatives processes: SSH into the affected host and ensure no other instances of the alternatives tool are currently running:

   ```
   pgrep -af 'update-alternatives|alternatives'
   ```

   If the command returns no active processes, proceed to Step 2.
2. Identify the stale temporary files:List the temporary files to confirm which entries are blocked:

   ```
   ls -l /var/lib/alternatives/*.new
   ```

   To check a specific service (for example, HDFS or Hive), use:

   ```
   ls -l /var/lib/alternatives/<alt_name>*
   ```
3. Back up and remove the stale .new files: Create a backup of the stale file in a temporary directory before deleting it:

   ```
   sudo cp -a /var/lib/alternatives/<alt_name>.new /var/tmp/<al
   t_name>.new.$(date +%s).bak
   sudo rm -f /var/lib/alternatives/<alt_name>.new
   ```
4. Verify the alternatives state: Confirm the current status of the link:

   ```
   /usr/sbin/update-alternatives --display <alt_name>
   ```
5. Retry the operation: Return to the Cloudera Manager UI and re-run Deploy Client Configuration for the affected service.

**OPSAPS-76960: Missing symbolic links due to Agent restart race condition during Cloudera Manager upgrade**

7.13.2

During a Cloudera Manager upgrade, a race condition might prevent the creation of critical parcel symbolic links (symlinks). If the cloudera-scm-agent restarts while an update-alternatives process is active, the system terminates the task (Exit Code -15). This leaves symlinks unconfigured and causes dependent services, such as Solr, to fail.

When you upgrade Cloudera Manager, the cloudera-manager-agent package update triggers the cloudera-scm-agent service to execute parcel activation tasks. If an automated script or a user issues a systemctl restart cloudera-scm-agent command while update-alternatives is running, the operating system sends a SIGTERM (Signal 15) to the Agent and its child processes.

This forceful termination interrupts the creation of essential symbolic links, including:

- /var/lib/hadoop-hdfs/ozone-filesystem-hadoop3.jar
- /etc/alternatives/ozone-filesystem-hadoop3.jar

The Agent logs this failure as Exit Code: -15 in /var/log/cloudera-scm-agent/cloudera-scm-agent.log. Because these links are missing, dependent services cannot locate necessary libraries and fail to start.

If services fail to start after an upgrade due to missing alternatives or symbolic links (symlinks), manually complete the interrupted activation steps on the affected host or use the Cloudera Manager UI to reconcile the state.

Option 1: Manual fix through CLI

1. Verify the missing symlink: SSH into the affected host machine and check the status of the failing JAR or symlink. For example:

   ```
   /usr/sbin/update-alternatives --display ozone-filesystem-had
   oop3.jar
   ```

   If the output shows the link is missing or broken, proceed to Step 2.
2. Manually run the interrupted command:

   Search the Agent log at /var/log/cloudera-scm-agent/cloudera-scm-agent.log for Exit Code: -15 to locate the failed update-alternatives command immediately preceding that error. Copy the full path to the parcel library from that log entry.

   For the ozone-filesystem-hadoop3.jar failure, run the following installation command manually as the root user.:

   ```
   sudo /usr/sbin/update-alternatives --install /var/lib/hadoop
   -hdfs/ozone-filesystem-hadoop3.jar ozone-filesystem-hadoop3.
   jar /opt/cloudera/parcels/<CDH-VERSION-PATH>/lib/hadoop-ozone/
   ozone-filesystem-hadoop3.jar 5
   ```

   **Important:** Replace <CDH-VERSION-PATH> with the specific parcel version path found in your log file.
3. Verify the fix: Run the display command again to ensure the link now correctly points to the new parcel directory:

   ```
   /usr/sbin/update-alternatives --display ozone-filesystem-had
   oop3.jar
   ```
4. Restart Services: After you verify the fix, restart the failing service (such as Solr) through the Cloudera Manager UI.

Option 2: Cloudera Manager UI (Automatic Fix)

Instead of manual CLI intervention, you can force Cloudera Manager to recreate all symbolic links:

1. Navigate to the Hosts Parcels page in the Cloudera Manager UI.
2. Select the affected parcel and click Activate again. This process declaratively identifies and recreates any missing symbolic links across all hosts the cluster.

**CDPD-99248: Ozone upgrade finalization might fail in Cloudera Manager**

7.13.2

When finalizing an Ozone upgrade for the first time through Cloudera Manager, the finalization command might report a failure in the standard error (stderr) log, even though the process continues to run on the Storage Container Manager (SCM).

During the initial Ozone upgrade finalization, Cloudera Manager might return the following error message:

```
Invalid response from Storage Container Manager.
```

```
Current finalization status is: FINALIZATION_IN_PROGRESS
```

This error occurs because Cloudera Manager fails to parse the interim status response from the SCM. Despite the "failed" status in the Cloudera Manager interface, the SCM continues the finalization process in the background.

If you encounter this error, do not restart the finalization command. Instead, manually verify the progress directly on the SCM by following these steps:

1. Check the Status: Run the following command from the command line to monitor the actual SCM finalization state:

   ```
   ozone admin scm finalizationstatus
   ```

2. Wait for Completion: Monitor the output until it indicates that finalization is complete.
3. Verify in Cloudera Manager: Once the CLI command confirms a successful finalization, you can safely ignore the previous failure message in Cloudera Manager and proceed with your post-upgrade tasks.

**OPSAPS-76363: Knox gateway database connection properties are not populated automatically when Oracle is the cluster database**

7.13.2

When you run the Knox gateway in high availability (HA) and use JWT token features, only MySQL and PostgreSQL are supported for the Knox gateway token database; Oracle is not supported. Cloudera Manager populates the Knox gateway database connection properties automatically only when you use MySQL or PostgreSQL as the Knox gateway database. If you use Oracle instead, these properties are not populated automatically.

Without those settings, JWT tokens might not behave consistently across Knox gateway instances.

Use MySQL or PostgreSQL for the Knox gateway database when you run Knox in HA with JWT token features.

**OPSAPS-76116: Knox gateway database configuration might not be retained after upgrading a Cloudera Base on premises cluster**

7.13.2

Knox database configuration properties knox_gateway_database_name, knox_gateway_databas e_host, knox_gateway_database_user, and knox_gateway_database_password might not be retained after you upgrade a Cloudera Base on premises cluster.

None

**OPSAPS-76528: Ozone services enforce IPv4 in DUAL_STACK configuration**

7.13.2

In Cloudera Base on premises 7.3.2.0, Ozone must be able to operate correctly in a DUAL-STACK environment, but you will not have control over whether Ozone services communicate in IPv4-only mode or in dual-stack mode.

None

**OPSAPS-75735: Systemd disables Cgroup v2 Controllers written by Cloudera Manager Agent**

When you enable Cgroup v2, Cloudera Manager services might fail to start because specific controllers (such as cpu, memory, or pids) are missing or not enabled at the root.

Although the Cloudera Manager Agent writes controllers to the root subtree during startup, syst emd (the cgroup manager for most modern Linux distributions) frequently disables controllers from the delegation tree if a service does not actively use them. On specific Linux distributions, syst emd's strict enforcement of this cleanup prevents the Cloudera Manager Agent from maintaining the necessary environment for service sub-processes.

If you encounter an error stating that a controller is "missing" or "not enabled at root," follow these steps to restore and persist the controllers.

1. Restart the Cloudera Manager Agent on the affected hosts to force it to rewrite the controllers to the root subtree.

   ```
   sudo systemctl restart cloudera-scm-agent
   ```

   Try starting the affected services again. If the issue persists, proceed to the next step.
2. Configure Systemd Delegation by performing the following steps:

   Explicitly instruct systemd to delegate cgroup controllers to the Cloudera Manager Agent process. This ensures the controllers remain available at the root level regardless of systemd's cleanup policies.

   a. Open the unit file: Use a text editor to open the Cloudera Manager Agent service unit file: /usr/lib/systemd/system/cloudera-scm-agent.service
   b. Add the Delegate parameter: Locate the [Service] section. Add Delegate=yes to the configuration.

      Before:

      ```
      [Service]
      Type=simple

      TasksMax=infinity
      ```

      After:

      ```
      [Service]
      Type=simple
      Delegate=yes
      TasksMax=infinity
      ```
3. Save the changes to the file.
4. Reload and restart: Run the following commands in order to apply the new configuration:

   ```
   sudo systemctl daemon-reload
   sudo systemctl restart cloudera-scm-agent
   ```

**OPSAPS-76314: Cloudera Management Service restart fails on large clusters due to Cloudera Manager Descriptor Fetch Timeout**

On large-scale deployments, the Cloudera Management Service might fail to start or restart correctly with the following error:

```
2026-01-03 04:43:05,600 INFO com.cloudera.cmf.BasicScmProxy: Aut
henticated to SCM.
2026-01-03 04:43:16,074 WARN com.cloudera.cmf.BasicScmProxy: Time
d out while fetching the SCM descriptor. This can happen on larg
e clusters. Timeout can be increased by configuring Descriptor F
etch Timeout under Administration > Settings.
2026-01-03 04:43:16,075 WARN com.cloudera.cmf.eventcatcher.ser
ver.EventCatcherService: No descriptor fetched from https://ip-1
0-129-36-226.iopscloud.cloudera.com:7183 on after 1 tries, sleep
ing for 2 secs.
```

This occurs because the Cloudera Manager Descriptor Fetch Timeout defaults to 10 seconds, which is often insufficient for the Cloudera Manager Server to generate and transmit the full cluster descriptor to Cloudera Management Service roles like the Event Server or Host Monitor in high-scale environments.

Reaching this timeout causes the service to log a warning and fail initialization. Consequently, the Event Catcher enters a loop, unable to retrieve the necessary configuration.

f you encounter service startup failures on large clusters, manually increase the fetch timeout through the Cloudera Manager Admin Console:

1. Log in to the Cloudera Manager Admin Console.
2. Navigate to  Administration Settings .
3. Search for the parameter: Cloudera Manager Descriptor Fetch Timeout.
4. Increase the value from the default 10 seconds to 60 seconds.
5. Click Save Changes.
6. Restart the Cloudera Management Service.

**OPSAPS-75899: HDFS directory creation fails on JDK 11 or higher when LDAP or Active Directory integrated clusters using the hadoop.security.group.mapping property.**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500, and 7.13.1.600

Due to this issue, many critical Cloudera Manager operations fail to complete, such as:

- Install Oozie ShareLib ( Oozie Actions Install Oozie ShareLib )
- Install YARN MapReduce Framework JARs ( YARN  Actions Install YARN MapReduce Framework JARs )

You must perform the following workaround steps to manually add specific Java modules to the HDFS service script on all nodes in the cluster:

1. Navigate to the directory /opt/cloudera/cm-agent/service/hdfs/.
2. Open the hdfs.sh file for editing.
3. Locate the JAVA17_ADDITIONAL_JVM_ARGS variable.
4. Append the flags (--add-exports=java.naming/com.sun.jndi.ldap=ALL-UNNAMED --add-opens=java.naming/com.sun.jndi.ldap=ALL-UNNAMED) to the end of the existing list, and update JAVA17_ADDITIONAL_JVM_ARGS variable to include the following flags:

```
Change this:
JAVA17_ADDITIONAL_JVM_ARGS="--add-opens=java.base/java.lang
=ALL-UNNAMED --add-opens=java.management/com.sun.jmx.mbeanse
rver=ALL-UNNAMED --add-exports=java.management/com.sun.jmx.m
beanserver=ALL-UNNAMED --add-exports=java.base/sun.net.dns=A
LL-UNNAMED --add-exports=java.base/sun.net.util=ALL-UNNAMED"
To this:
JAVA17_ADDITIONAL_JVM_ARGS="--add-opens=java.base/java.lang=AL
L-UNNAMED --add-opens=java.management/com.sun.jmx.mbeanserve
r=ALL-UNNAMED --add-exports=java.management/com.sun.jmx.mbea
nserver=ALL-UNNAMED --add-exports=java.base/sun.net.dns=ALL-
UNNAMED --add-exports=java.base/sun.net.util=ALL-UNNAMED --a
dd-exports=java.naming/com.sun.jndi.ldap=ALL-UNNAMED --add-o
pens=java.naming/com.sun.jndi.ldap=ALL-UNNAMED"
```

**OPSAPS-74066, OPSAPS-74547: DataHub high memory consumption on Hiveserver load for JDK 17**

7.13.2

In upgraded DataHub deployments, HiveServer might fail to start due to memory overallocation. This occurs because Cloudera Manager does not account for memory already assigned to Management Service roles when allocating memory for cluster roles. This issue is fixed in fresh installations of Cloudera Manager 7.13.1.500. The updated algorithm now correctly reallocates memory across all roles during cluster setup.

To resolve this issue, use the following API to manually trigger the Cloudera Manager memory allocation algorithm on the host where both HiveServer and management roles are running, and restart cluster to apply updated memory configs.:

```
API Endpoint: POST /api/v57/hosts/reallocateMemory
```

Include the host name (the host where HiveServer and management roles run) in the API request body. This ensures that memory assignments are recomputed correctly, taking all roles on the host into account.

**Important:**

Cloudera Manager always assigns at least the minimum requested memory for every role, even if this results in allocating more memory than the host's available physical memory.

After running the above API, if it still has overallocation and can't reduce memory allocation further for any role on the node, it will still lead to failures. For example, if the Roles are not starting due to high memory requests, then manually reduce the heap size configuration for the affected role.

**OPSAPS-74668: ozone.snapshot.deep.cleaning.enabled and ozone.snapshot.ordered.deletion.enabled configs are missing with Cloudera 7.1.9 SP1 CHF and Cloudera Manager 7.13.1**

7.13.2

Two Ozone Manager configs are missing while using Cloudera 7.1.9 SP1 CHF after upgrading Cloudera Manager version from 7.11.3 to 7.13.1.400.

If you are using Cloudera 7.1.9 SP1 CHF, before upgrading Cloudera Manager version from 7.11.3 to 7.13.1.400, add the following configs to Ozone Manager Advanced Configuration Snippet (Safety Valve) for ozone-conf/ozone-site.xml so that Ozone Manager does not miss important config after the Cloudera Manager upgrade:

```
<property>
<name>ozone.snapshot.deep.cleaning.enabled</name>
<value>false</value>
</property>

<property>
<name>ozone.snapshot.ordered.deletion.enabled</name>
<value>true</value>
</property>
```

**OPSAPS-73038: False-positive port conflict error message displayed in Cloudera Manager**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500

Cloudera Manager might display a false-positive error message: Port conflict detected: 8443 (Gateway Health HTTP Port) is also used by: Knox Gateway during cluster installations. The warning does not cause actual installation failures.

None

**OPSAPS-74950: Ozone replication policies fail for Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters using Cloudera Manager 7.13.1.400**

7.13.1.400

7.13.1.500, 7.13.2.0

Ozone replication policies for Ozone linked buckets fail when the Cloudera Private Cloud Base 7.1.9 SP1 CHF11 source or target clusters use Cloudera Manager 7.13.1.400.

Use Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters with Cloudera Manager 7.13.1.500.

**OPSAPS-72439: HDFS and Hive external tables replication policies fail when using custom “krb5.conf” files for Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters using Cloudera Manager 7.13.1.400**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500, , 7.13.2.0

The issue appears when the custom krb5.conf file is not propagated to the required files, and you are using Cloudera Private Cloud Base 7.1.9 SP1 CHF11 source or target clusters with Cloudera Manager 7.13.1.400.

Use Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters with Cloudera Manager 7.13.1.500, and complete the instructions in step 13 in Using a custom Kerberos configuration path.

**OPSAPS-71459: Commands continue to run after Cloudera Manager restart**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500, 7.13.2.0

If you are using Cloudera Private Cloud Base 7.1.9 SP1 CHF11 source or target clusters with Cloudera Manager 7.13.1.400, remote replication commands continue to run endlessly even after a Cloudera Manager restart operation.

Use Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters with Cloudera Manager 7.13.1.500.

**OPSAPS-73158, OPSAPS-74206: HDFS replication policies fail when the policies prefetch the expired Kerberos ticket from the 'sourceTicketCache' file for Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters using Cloudera Manager 7.13.1.400**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500, 7.13.2.0

If you are using Cloudera Private Cloud Base 7.1.9 SP1 CHF11 source or target clusters with Cloudera Manager 7.13.1.400, Replication Manager pre-fetches the Kerberos ticket from the sourceTicketCache file for the replication policies. Issues appear when the file contains an expired Kerberos ticket.

Use Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters with Cloudera Manager 7.13.1.500.

**OPSAPS-73405, OPSAPS-71565, OPSAPS-72860, OPSAPS-72859: Replication policies fail even after the source or target cluster becomes available after it recovers from temporary node failures for Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters using Cloudera Manager 7.13.1.400**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500, 7.13.2.0

If you are using Cloudera Private Cloud Base 7.1.9 SP1 CHF11 source or target clusters with Cloudera Manager 7.13.1.400, Hive replication policies and HBase replication policies fail even after the source or target cluster recovers from a temporary node failure.

Use Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters with Cloudera Manager 7.13.1.500.

**OPSAPS-73655, OPSAPS-73737: Cloud replication fails even after the delegation token is issued for Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters using Cloudera Manager 7.13.1.400**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500, 7.13.2.0

If you are using Cloudera Private Cloud Base 7.1.9 SP1 CHF11 source or target clusters with Cloudera Manager 7.13.1.400, the replication policies fail during an incremental replication run if you chose the  Advanced Setting Delete Policy Delete permanently  option during the replication policy creation process.

Use Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters with Cloudera Manager 7.13.1.500.

**OPSAPS-74040, OPSAPS-74058: Ozone OBS replication fails due to pre-filelisting check failure for Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters using Cloudera Manager 7.13.1.400**

7.13.1.400

7.13.1.500, 7.13.2.0

If you are using Cloudera Private Cloud Base 7.1.9 SP1 CHF11 source or target clusters with Cloudera Manager 7.13.1.400 and the source bucket is a linked bucket, then the replication fails during the `Run Pre-Filelisting Check` step for OBS-to-OBS Ozone replication, and the error message Source bucket is a linked bucket, however the bucket it points to is also a link appears. This issue appears even when the source bucket is directly linked to a regular, non-linked bucket.

Use Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters with Cloudera Manager 7.13.1.500.

**OPSAPS-73602, OPSAPS-74353: HDFS replication policies to cloud fails with HTTP 400 error for Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters using Cloudera Manager 7.13.1.400**

7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500, 7.13.2.0

If you are using Cloudera Private Cloud Base 7.1.9 SP1 CHF11 source or target clusters with Cloudera Manager 7.13.1.400, the HDFS replication policies to cloud fail after you edit the replication policies in the Cloudera Manager Replication Manager UI .

Use Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters with Cloudera Manager 7.13.1.500.

**OPSAPS-73645, OPSAPS-73847: Ozone bucket browser does not show the volume buckets for Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters using Cloudera Manager 7.13.1.400**

7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500, 7.13.2.0

If you are using Cloudera Private Cloud Base 7.1.9 SP1 CHF11 source or target clusters with Cloudera Manager 7.13.1.400, the volume buckets do not appear if the number of volumes exceed 26, when you click on Next Page on the Cloudera Manager Clusters *OZONE SERVICE* Bucket Browser page and then on a volume name.

Use Cloudera Private Cloud Base 7.1.9 SP1 CHF11 clusters with Cloudera Manager 7.13.1.500.

**RELENG-27000: Proper link for bigtop-detect-javahome is missing when using CDP Private Cloud Base 7.1.9 SP1 CHF5 with Cloudera Manager 7.13.1 CHF3.**

When using Cloudera Manager 7.13.1 CHF3 with CDP Private Cloud Base 7.1.9 SP1 CHF5 results in inappropriate bigtop-detect-javahome link.

Create a link under /opt/cloudera/parcels/CDH/bin/bigtop-detect-javahome that points to /opt/cloudera/parcels/CDH/lib/bigtop-utils/bigtop-detect-javahome. For example:

```
ln -s /opt/cloudera/parcels/CDH/lib/bigtop-utils/bigtop-detect-j
avahome /opt/cloudera/parcels/CDH/bin/bigtop-detect-javahome
```

**CDPD-79725: Hive fails to start after Datahub restart due to high memory usage**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

After restarting the Cloudera Data hub, the services appears to be down in the Cloudera Manager UI. The Cloudera Management Console reports a node failure error for the master node.

The issue is caused by high memory usage due to the G1 garbage collector on Java 17, leading to insufficient memory issues and thereby moving the Cloudera clusters to an error state.

Starting with Cloudera 7.3.1.0, Java 17 is the default runtime instead of Java 8, and its memory management increases memory usage, potentially affecting system performance. Clusters might report error states, and logs might show insufficient memory exceptions.

To mitigate this issue and prevent startup failures after a Datahub restart, you can perform either of the following actions, or both:

- Reduce the Java heap size for affected services to prevent nodes from exceeding the available memory.
- Increase physical memory for on cloud or on-premises instances running the affected services.

**OPSAPS-74370: Knox's Save Alias - IDBroker command fails due to missing variable declaration**

7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

Users trying to create IDBroker aliases through the Cloudera Manager UI face issues in Cloudera Manager 7.13.1 using CDP 7.1.9.

The alias(es) can be created using the Knox CLI:

1. ssh to Knox host.
2. export KNOX_GATEWAY_DATA_DIR="/var/lib/knox/idbroker/data"; export KNOX_GATEWAY_CONF_DIR="/var/lib/knox/idbroker/conf"
3. /opt/cloudera/parcels/CDH/lib/knox/bin/knoxcli.sh create-alias <ALIAS_NAME> --cluster <CLUSTER_NAME> --value <ALIAS_VALUE>
4. Verify the addition using /opt/cloudera/parcels/CDH/lib/knox/bin/knoxcli.sh list-alias --cluster <CLUSTER_NAME>

For HA deployments, users must do it on every Knox hosts (whereas the Save Alias command applies the change to all hosts automatically).

**OPSAPS-71669: The Continue option is disabled on the Static Service Pools Review page, affecting the functionality of Static Service Pools**

7.13.1

7.13.1.100

The minimum and maximum I/O weight values for Cgroup v2 were incorrectly set to 100 and 1000, respectively, in Cloudera Manager 7.13.1.0. According to official Cgroup v2 documentation, the valid range should be 1 to 10,000. Due to this incorrect configuration range, the Continue option on the **Static Service Pools Review** page was disabled, preventing users from proceeding with pool configuration.

This issue might occur on clusters running Cloudera Manager 7.13.1.0 with Cgroup v2 resource management when configuring or reviewing Static Service Pools. After upgrading to Cloudera Manager 7.13.1.100 CHF-1, this issue no longer occurs.

None

**OPSAPS-75290, OPSAPS-74994: The yarn_enable_container_usage_aggregation job is failing with “Null real user” error on Service Monitor.**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, and 7.13.1.500

The yarn_enable_container_usage_aggregation job is failing with "Null real user" error on Service Mnitor when the Yarn service is running on the computer cluster with Stub DFS, and when the Powerscale Service is running in the cluster with Powerscale DFS provider instead of HDFS.

None.

**OPSAPS-71581: Cloudera Manager Agent's append_properties function fails with the realpath: invalid option -- 'u' error when executed from service control scripts.**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, and 7.13.1.500

Errors appear on the standard error (stderr) log of Cloudera Data Platform (CDP) services when you are attempting to trigger the cloudera-config.sh script. The error log contains the following message: realpath: invalid option -- 'u'. This is caused by an incorrectly placed command-line flag in the script, which prevents some service configurations from loading correctly.

To resolve this issue temporarily, you must perform the following workaround steps on each agent node in the base cluster::

1. Navigate to the directory /opt/cloudera/cm-agent/service/common/.

2. Open the cloudera-config.sh file for editing.
3. Locate the two lines that execute the python scripts such as append_properties.py and get_property.py.
4. In both lines, remove the -u flag or change its position to after python to the end of the line:

```
Change this:
value=$(python -u "${GET_PROPERTY_PY_DIR}"/get_property.py
"${1}" "${2}")
To this:
value=$(python "${GET_PROPERTY_PY_DIR}"/get_property.py "${1}"
 "${2}" -u)
```

```
Change this:
python -u "${APPEND_PROPERTIES_PY_DIR}"/append_properties.py
 "${1}" "${2}"
To this:
python "${APPEND_PROPERTIES_PY_DIR}"/append_properties.py "
${1}" "${2}" -u
```

5. After saving the changes on all agent nodes, restart the entire cluster for the new configuration to take effect.
6. Verify the fix by checking the stderr.log on a few service instances to ensure the realpath: invalid option -- 'u' error no longer appears.

**OPSAPS-71878: Ozone fails to restart during cluster restart and displays the error message: Service has only 0 Storage Container Manager roles running instead of minimum required 1.**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, and 7.13.1.500

1. You must open Cloudera Manager on the second browser and restart the Ozone service separately.
2. After the Ozone service restarts, you can resume the cluster restart from the first browser.

**ENGESC-30503, OPSAPS-74868: Cloudera Manager limited support for custom external repository requiring basic authentication**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, and 7.13.1.400

Current Cloudera Manager does not support custom external repository with basic authentication (the Cloudera Manager Wizard supports either HTTP (non-secured) repositories or usage of Cloudera https://archive.cloudera.com only). In case customers want to use a custom external repository with basic authentication, they might get errors.

The assumption is that you can access the external custom repository (such as Nexus or JFrog, or others) using LDAP credentials. In case an applicative user is used to fetch the external content (as done in Data Services with the docker imager repository), the customer should ensure that this applicative user is located under the user's base search path where the real users are being retrieved during LDAP authentication check (so the external repository will find it and will allow it to gain access for fetching the files).

Once done, you can use the current custom URL fields in the Cloudera Manager Wizard and enter the URL for the RPMs or parcels/other files in the format of "https://USERNAME:PASSWOR D@server.example.com/XX".

While using the password, you are advised to use only the printable ASCII character range (excluding space), whereas in case of a special character (not letter/number) it can be replaced with HEX value (For example, you can replace Aa1234$ with Aa1234%24 as '%24' is translated into $ sign).

**OPSAPS-72164: Proxy Settings and Telemetry Publisher in Cloudera Manager**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, and 7.13.1.400

In Cloudera Manager 7.13.1, the PROXY settings for the Telemetry Publisher (TP) are not functioning as expected. This may impact the Telemetry Publisher's ability to communicate through a configured proxy.

You must upgrade to Cloudera Manager 7.13.1 CHF5 (7.13.1.500) and higher.

**OPSAPS-60726: Newly saved parcel URLs are not showing up in the parcels page in the Cloudera Manager HA cluster.**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

To safely manage parcels in a Cloudera Manager HA environment, follow these steps:

1. Shutdown the Passive Cloudera Manager Server.
2. Add and manage the parcel as usual, as described in Install Parcels.
3. Restart the Passive Cloudera Manager server after parcel operations are complete.

**OPSAPS-74341: NodeManagers might fail to start during the cluster restart after the Cloudera Manager 7.13.1.x upgrade**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500

Cgroup v2 support is enabled in CDP 7.1.9 SP1 CHF5 and higher versions. However, if the user upgrades from Cloudera Manager 7.11.3.x to Cloudera Manager 7.13.1.x, and the environment is using cgroup v2, the NodeManagers might fail to start during the cluster restart after the Cloudera Manager 7.13.1.x upgrade.

To resolve this issue temporarily, you must perform the following steps:

1. Go to the YARN service page on the Cloudera Manager UI.
2. Navigate to the Configuration tab.
3. Search for NodeManager Advanced Configuration Snippet (Safety Valve) for yarn-site.xml.
4. Add the following entry:
   a. Add yarn.nodemanager.linux-container-executor.cgroups.v2.enabled=true
5. Restart the Nodemanagers. Nodemanagers restart successfully.

**OPSAPS-73546: Service Monitor fails to perform Canary tests on HMS / HBASE / ZooKeeper due to missing dependencies**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

Due to a missing dependency caused by an incomplete build and packaging in certain OS releases, the HMS (Hive Metastore) Canary health test fails, logging a ClassNotFoundException in the Service Monitor log. This problem relates to all deliveries using runtime cluster version 7.1.x or 7.2.x, while the Cloudera Manager version is 7.13.1.x and the OS is NOT RHEL8.

In case your OS is either RHEL 9 or SLES 15 or Ubuntu 2004 or Ubuntu 2204 and if you install the Cloudera Manager 7.13.1.x version, then create a symbolic link using root user privileges on the node that host the Service Monitor service (cloudera-scm-firehose) at /opt/cloudera/cm/lib/cdh71/cdh71-hive-client-7.13.1-shaded.jar, pointing to /opt/cloudera/cm/lib/cdh7/cdh7-hive-client-7.13.1-shaded.jar.

**Note:** The above example relates to Cloudera Base on premises releases. In case your cluster is on Cloud, use "cdh72" instead of "cdh71" in the above symbolic link.

Restart the Service Monitor service post the change. This will allow the Service Monitor to perform Canary testing correctly on the HMS (Hive Metastore) service.

**OPSAPS-72706, OPSAPS-73188: Hive queries fail after upgrading Cloudera Manager from 7.11.2 to 7.11.3 or later**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500

Upgrading Cloudera Manager from version 7.11.2 or earlier to 7.11.3 or later causes Hive queries to fail due to JDK17 restrictions. Some JDK8 options are deprecated, leading to inaccessible classes and exceptions:

```
java.lang.reflect.InaccessibleObjectException: Unable to make fi
eld private volatile java.lang.String java.net.URI.string access
ible
```

To resolve this issue:

1. In Cloudera Manager, go to Tez Configuration
2. Append the following values to both tez.am.launch.cmd-opts and tez.task.launch.cmd-opts:

```
--add-opens=java.base/java.net=ALL-UNNAMED
--add-opens=java.base/java.util=ALL-UNNAMED
--add-opens=java.base/java.util.concurrent.atomic=ALL-UNNAMED
--add-opens=java.base/java.util.regex=ALL-UNNAMED
--add-opens=java.base/java.lang=ALL-UNNAMED
--add-opens=java.base/java.time=ALL-UNNAMED
--add-opens=java.base/java.io=ALL-UNNAMED
--add-opens=java.base/java.nio=ALL-UNNAMED
```

3. Save and restart

**OPSAPS-72998: Missing charts for HMS event APIs**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

Charts for HMS event APIs (get_next_notification, get_current_notificationEventId, and fire_listener_event) are missing in Cloudera Manager Hive Hive Metastore Instance Charts Library API

Monitor HMS event activity using Hive Metastore logs.

**OPSAPS-72270: Start ECS command fails on uncordon nodes step**

7.13.1, 7.13.1.100, 7.13.1.200

7.13.1.300

In an ECS HA cluster, the server node restarts during the start up. This may cause the uncordon step to fail.

To resolve this issue temporarily, you must perform the following steps:

1. Run the following command on the same node to verify whether the kube-apiserver is ready:

```
kubectl get pods -n kube-system | grep kube-apiserver
```

2. Resume the command from the Cloudera Manager UI.

**OPSAPS-73225: Cloudera Manager Agent reporting inactive/failed processes in Heartbeat request**

7.13.1, 7.13.1.100, 7.13.1.200

7.13.1.300

As part of introducing Cloudera Manager 7.13.x, some changes were done to the Cloudera Manager logging, eventually causing Cloudera Manager Agent to report on inactive/stale processes during Heartbeat request.

As a result, the Cloudera Manager servers logs are getting filled rapidly with these notifications though they do not have impact on service.

In addition, with adding the support for the Cloudera Observability feature, some additional messages were added to the logging of the server. However, in case the customer did not purchase the Cloudera Observability feature, or the telemetry monitoring is not being used, these messages

(which appears as "TELEMETRY_ALTUS_ACCOUNT is not configured for Otelcol" are filling the server logs and preventing proper follow-up on the server activities).

This will be fixed in a later release by moving these log notifications to DEBUG level so they don't appear on the Cloudera Manager server logs. Until that fix, perform the following workaround to filter out these messages.

On each of the Cloudera Manager servers, update with root credentials the file /etc/cloudera-scm-server/log4j.properties and add the following lines at the end of the file:

```
# === Custom Appender with Filters ===
log4j.appender.filteredlog=org.apache.log4j.ConsoleAppender
log4j.appender.filteredlog.layout=org.apache.log4j.PatternLayout
log4j.appender.filteredlog.layout.ConversionPattern=%d{ISO8601}
%p %c: %m%n
# === Filter #1: Drop warning ===
log4j.appender.filteredlog.filter.1=org.apache.log4j.varia.Str
ingMatchFilter
log4j.appender.filteredlog.filter.1.StringToMatch=Received Proces
s Heartbeat for unknown (or duplicate) process.
log4j.appender.filteredlog.filter.1.AcceptOnMatch=false
# === Filter #2: Drop telemetry config warning ===
log4j.appender.filteredlog.filter.2=org.apache.log4j.varia.String
MatchFilter
log4j.appender.filteredlog.filter.2.StringToMatch=TELEMETRY_ALTU
S_ACCOUNT is not configured for Otelcol
log4j.appender.filteredlog.filter.2.AcceptOnMatch=false
# === Accept all other messages ===
log4j.appender.filteredlog.filter.3=org.apache.log4j.varia.Accep
tAllFilter
# === Specific logger for AgentProtocolImpl ===
log4j.logger.com.cloudera.server.cmf.AgentProtocolImpl=WARN, filt
eredlog
log4j.additivity.com.cloudera.server.cmf.AgentProtocolImpl=false
# === Specific logger for BaseMonitorConfigsEvaluator ===
log4j.logger.com.cloudera.cmf.service.config.BaseMonitorConfigsE
valuator=WARN, filteredlog
log4j.additivity.com.cloudera.cmf.service.config.BaseMonitorConfi
gsEvaluator=false
```

Once done, restart the Cloudera Manager server(s) for the updated configuration to be picked.

**OPSAPS-73211: Cloudera Manager 7.13.1 does not clean up Python Path impacting Hue to start**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

When you upgrade from Cloudera Manager 7.7.1 or lower versions to Cloudera Manager 7.13.1 or higher versions with CDP Private Cloud Base 7.1.7.x Hue does not start because Cloudera Manager forces Hue to start with Python 3.8, and Hue needs Python 2.7.

The reason for this issue is because Cloudera Manager does not clean up the Python Path at any time, so when Hue tries to start the Python Path points to 3.8, which is not supported in CDP Private Cloud Base 7.1.7.x version by Hue.

To resolve this issue temporarily, you must perform the following steps:

1. Locate the hue.sh in /opt/cloudera/cm-agent/service/hue/.
2. Add the following line after export HADOOP_CONF_DIR=$CONF_DIR/hadoop-conf:

```
export PYTHONPATH=/opt/cloudera/parcels/CDH/lib/hue/build/env/
lib64/python2.7/site-packages
```

**OPSAPS-73011: Wrong parameter in the /etc/default/cloudera-scm-server file**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300

7.13.1.400

In case the Cloudera Manager needs to be installed in High Availability (2 nodes or more as explained here), the parameter CMF_SERVER_ARGS in the /etc/default/cloudera-scm-server file is missing the word "export" before it (on the file there is only CMF_SERVER_ARGS= and not expo rt CMF_SERVER_ARGS=), so the parameter cannot be utilized correctly.

Edit the /etc/default/cloudera-scm-server file with root credentials and add the word "export" before the parameter CMF_SERVER_ARGS=.

**OPSAPS-60346: Upgrading Cloudera Manager Agent triggers cert rotation in Auto-TLS use case 1**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

Upgrading Cloudera Manager Agent nodes from the Cloudera Manager UI wizard as part of a Cloudera Manager upgrade causes the host to get new certificates, which becomes disruptive.

The issue happens with use case 1 and Cloudera Manager DB is because Cloudera Manager always regenerates the host cert as part of the host install or host upgrade step. However, with use case 3, Cloudera Manager does not regenerate the cert as it comes from the user.

Currently, there are three following possible workarounds:

- Rotate all CMCA certs again using the `generateCmca` API command, and using the "location" argument to specify a directory on disk. This will revert to the old behavior of storing the certs on disk instead of the DB.

  **Important:** This approach is not recommended if Cloudera Manager is in HA mode or you plan to enable HA in the future.

- Switch to Auto-TLS Use Case 3 (Customer CA-signed Certificates).
- Manual upgrade of Cloudera Manager Agents, instead of upgrading from Cloudera Manager GUI.

**OPSAPS-72447, CDPD-76705: Ozone incremental replication fails to copy renamed directory**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300

7.13.1.400, 7.13.2.0

Ozone incremental replication using Ozone replication policies succeed but might fail to sync nested renames for FSO buckets.

When a directory and its contents are renamed between the replication runs, the outer level rename synced but did not sync the contents with the previous name.

None

**OPSAPS-72756:The runOzoneCommand API endpoint fails during the Ozone replication policy run**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300

7.13.1.400, 7.13.2.0

The `/clusters/{clusterName}/runOzoneCommand` Cloudera Manager API endpoint fails when the API is called with the `getOzoneBucketInfo` command. In this scenario, the Ozone replication policy runs also fail if the following conditions are true:

- The source Cloudera Manager version is 7.11.3 CHF11 or 7.11.3 CHF12.
- The target Cloudera Manager is version 7.11.3 through 7.11.3 CHF10 or 7.13.0.0 or later where the feature flag API_OZONE_REPLICATION_USING_PROXY_USER is disabled.

Choose one of the following methods as a workaround:

- Upgrade the target Cloudera Manager before you upgrade the source Cloudera Manager for 7.11.3 CHF12 version only.
- Pause all replication policies, upgrade source Cloudera Manager, upgrade destination Cloudera Manager, and unpause the replication policies.

- Upgrade source Cloudera Manager, upgrade target Cloudera Manager, and rerun the failed Ozone replication policies between the source and target clusters.

**OPSAPS-65377: Cloudera Manager - Host Inspector not finding Psycopg2 on Ubuntu 20 or Redhat 8.x when Psycopg2 version 2.9.3 is installed.**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300

7.13.1.400

Host Inspector fails with Psycopg2 version error while upgrading to Cloudera Manager 7.13.1.x versions. When you run the Host Inspector, you get an error Not finding Psycopg2, even though it is installed on all hosts.

None

**OPSAPS-68340: Zeppelin paragraph execution fails with the User not allowed to impersonate error.**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

Starting from Cloudera Manager 7.11.3, Cloudera Manager auto-configures the livy_admin_users configuration when Livy is run for the first time. If you add Zeppelin or Knox services later to the existing cluster and do not manually update the service user, the User not allowed to impersonate error is displayed.

If you add Zeppelin or Knox services later to the existing cluster, you must manually add the respective service user to the livy_admin_users configuration in the Livy configuration page.

**OPSAPS-72804: For recurring replication policies, the interval is overwritten to 1 after the replication policy is edited**

7.13.1

7.13.1.100, 7.13.2.0

When you edit an Atlas, Iceberg, Ozone, or a Ranger replication policy that has a recurring schedule on the Replication Manager UI, the Edit Replication Policy modal window appears as expected. However, the frequency of the policy is reset to run at “1” unit where the unit depends on what you have set in the replication policy. For example, if you have set the replication policy to run every four hours, it is reset to one hour when you edit the replication policy.

After you edit the replication policy as required, you must ensure that you manually set the frequency to the original scheduled frequency, and then save the replication policy.

**OPSAPS-69342: Access issues identified in MariaDB 10.6 were causing discrepancies in High Availability (HA) mode**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

MariaDB 10.6, by default, includes the property require_secure_transport=ON in the configuration file (/etc/my.cnf), which is absent in MariaDB 10.4. This setting prohibits non-TLS connections, leading to access issues. This problem is observed in High Availability (HA) mode, where certain operations may not be using the same connection.

To resolve the issue temporarily, you can either comment out or disable the line require_secure_t ransport in the configuration file located at /etc/my.cnf.

**CDPD-53160: Incorrect job run status appears for subsequent Hive ACID replication policy runs after the replication policy fails**

7.13.1, 7.13.1.100, 7.13.1.200

7.13.1.300, 7.13.2.0

When a Hive ACID replication policy run fails with the **FAILED_ADMIN** status, the subsequent Hive ACID replication policy runs show **SKIPPED** instead of **FAILED_ADMIN** status on the Cloudera Manager Replication Manager Replication Policies Actions Show History page which is incorrect. It is recommended that you check Hive ACID replication policy runs if multiple subsequent policy runs show the **SKIPPED** status.

None

**CDPQE-36126: Iceberg replication fails when source and target clusters use different nameservice names**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

When you run an Iceberg replication policy between clusters where the source and target clusters use different nameservice names, the replication policy fails.

Perform the following steps to mitigate the issue, note that in the following steps the source nameservice is assumed to be ns1 and target cluster nameservice is assumed to be ns2:

1. Go to the Cloudera Manager Replication Replication Replication Policies page.
2. Click Actions Edit for the required Iceberg replication policy.
3. Go to the **Advanced** tab on the **Edit Iceberg Replication Policy** modal window.
4. Enter the mapreduce.job.hdfs-servers.token-renewal.exclude = ns1, ns2 key value pair for Advanced Configuration Snippet (Safety Valve) for source hdfs-site.xml and Advanced Configuration Snippet (Safety Valve) for destination hdfs-site.xml fields.
5. Save the changes.
6. Click Actions Run Now to run the replication policy.

**CDPD-53185: Clear REPL_TXN_MAP table on target cluster when deleting a Hive ACID replication policy**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300

7.13.1.400, 7.13.2.0

The entry in REPL_TXN_MAP table on the target cluster is retained when the following conditions are true:

1. A Hive ACID replication policy is replicating a transaction that requires multiple replication cycles to complete.
2. The replication policy and databases used in it get deleted on the source and target cluster even before the transaction is completely replicated.

In this scenario, if you create a database using the same name as the deleted database on the source cluster, and then use the same name for the new Hive ACID replication policy to replicate the database, the replicated database on the target cluster is tagged as ‘database incompatible’. This happens after the housekeeper thread process (that runs every 11 days for an entry) deletes the retained entry.

Create another Hive ACID replication policy with a different name for the new database

**DMX-3973: Ozone replication policy with linked bucket as destination fails intermittently**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

When you create an Ozone replication policy using a linked/non-linked source cluster bucket and a linked target bucket, the replication policy fails during the "Trigger a OZONE replication job on one of the available OZONE roles" step.

None

**OPSAPS-68143:Ozone replication policy fails for empty source OBS bucket**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

An Ozone incremental replication policy for an OBS bucket fails during the “Run File Listing on Peer cluster” step when the source bucket is empty.

None

**OPSAPS-71592: Replication Manager does not read the default value of “ozone_replication_core_site_safety_valve” during Ozone replication policy run**

7.13.1

7.13.1.100, 7.13.2

During the Ozone replication policy run, Replication Manager does not read the value in the ozone_replication_core_site_safety_valve advanced configuration snippet if it is configured with the default value.

To mitigate this issue, you can use one of the following methods:

- Remove some or all the properties in ozone_replication_core_site_safety_valve, and move them to ozone-conf/ozone-site.xml_service_safety_valve.
- Add a dummy property with no value in ozone_replication_core_site_safety_valve. For example, add <property><name>dummy_property</name><value></value></property>, save the changes, and run the Ozone replication policy.

**OPSAPS-71897: Finalize Upgrade command fails after upgrading the cluster with CustomKerberos setup causing INTERNAL_ERROR with EC writes.**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

After the UI `FinalizeCommand` fails, you must manually run the finalize commands through the Ozone Admin CLI:

1. `kinit with the scm custom kerberos principal`
2. `ozone admin scm finalizeupgrade`
3. `ozone admin scm finalizationstatus`

**OPSAPS-72204: HMS compaction configuration not updated through Cloudera Manager UI**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

The hive.compactor.initiator.on checkbox in Cloudera Manager UI for Hive Metastore (HMS) does not reflect the actual configuration value in cloud deployments. The default value is false, causing the compactor to not run.

To update the hive.compactor.initiator.on value:

1. In the Cloudera Manager, go to  Hive Configuration
2. Add the value for hive.compactor.initiator.on to true in the "Hive Service Advanced Configuration Snippet (Safety Valve) for hive-site.xml"
3. Save the changes and Restart.

Once applied, the compaction process will run as expected.

**OPSAPS-70702: Ranger replication policies fail if the clusters do not use AutoTLS**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

Ranger replication policies fail during the Exporting services, policies and roles from Ranger r emote step.

- Log in to the Ranger Admin host(s) on the source cluster.
- Identify the Cloudera Manager agent PEM file using the `# cat /etc/cloudera-scm-agent/config.ini | grep -i client_cert_file` command. For example, the file might reside in client_cert_file=/myTLSpath/cm_server-cert.pem location.
- Create the path for the new PEM file using the `# mkdir -p /var/lib/cloudera-scm-agent/agent-cert/` command.
- Copy the client_cert_file from config.ini as cm-auto-global_cacerts.pem file using the `# cp /myTLSpath/cm_server-cert.pem /var/lib/cloudera-scm-agent/agent-cert/cm-auto-global_cacerts.pem` command.
- Change the ownership to 644 using the `# chmod 644 /var/lib/cloudera-scm-agent/agent-cert/cm-auto-global_cacerts.pem` command.
- Resume the Ranger replication policy in Replication Manager.

**Note:** Ensure that you change /myTLSpath/cm_server-cert.pem to the actual PEM file location defined in config.ini under client_cert_file.

**OPSAPS-71424: The configuration sanity check step ignores during the replication advanced configuration snippet values during the Ozone replication policy job run**

7.13.1

7.13.1.100, 7.13.2.0

The OBS-to-OBS Ozone replication policy jobs fail if the S3 property values for fs.s3a.endpoint, fs.s3a.secret.key, and fs.s3a.access.key are empty in Ozone Service Advanced Configuration Sni ppet (Safety Valve) for ozone-conf/ozone-site.xml even though you defined the properties in Ozone Replication Advanced Configuration Snippet (Safety Valve) for core-site.xml.

Ensure that the S3 property values for fs.s3a.endpoint, fs.s3a.secret.key, and fs.s3a.access.key contains at least a dummy value in Ozone Service Advanced Configuration Snippet (Safety Val ve) for ozone-conf/ozone-site.xml.

Additionally, you must ensure that you do not update the property values in Ozone Replication Ad vanced Configuration Snippet (Safety Valve) for core-site.xml for Ozone replication jobs. This is because the values in this advanced configuration snippet overrides the property values in core-site.xml and not the ozone-site.xml file.

Different property values in Ozone Service Advanced Configuration Snippet (Safety Valve) for ozone-conf/ozone-site.xml and Ozone Replication Advanced Configuration Snippet (Safety Valve) for core-site.xml result in a nondeterministic behavior where the replication job picks up either value during the job run which leads to incorrect results or replication job failure.

**OPSAPS-71403: Ozone replication policy creation wizard shows "Listing Type" field in source Cloudera Private Cloud Base versions lower than 7.1.9**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

When the source Cloudera Private Cloud Base cluster version is lower than 7.1.9 and the Cloudera Manager version is 7.11.3, the Ozone replication policy creation wizard shows Listing Type and its options. These options are not available in Cloudera Private Cloud Base 7.1.8.x versions.

**OPSAPS-71659: Ranger replication policy fails because of incorrect source to destination service name mapping**

7.13.1

7.13.1.100, 7.13.2.0

Ranger replication policy fails because of incorrect source to destination service name mapping format during the transform step.

If the service names are different in the source and target, then you can perform the following steps to resolve the issue:

1. SSH to the host on which the Ranger Admin role is running.
2. Find the ranger-replication.sh file.
3. Create a backup copy of the file.
4. Locate substituteEnv SOURCE_DESTINATION_RANGER_SERVICE_NAME_MAPPING ${RANGER_REPL_SERVICE_NAME_MAPPING} in the file.
5. Modify it to substituteEnv SOURCE_DESTINATION_RANGER_SERVICE_NAME_MAPPING '"${RANGER_REPL_SERVICE_NAME_MAPPING//\"}'"
6. Save the file.
7. Rerun the Ranger replication policy.

**OPSAPS-69782: HBase COD-COD replication from 7.3.1 to 7.2.18 fails during the "create adhoc snapshot" step**

7.13.1

7.13.1.100, 7.13.2.0

An HBase replication policy replicating from 7.3.1 COD to 7.2.18 COD cluster that has ‘Perform Initial Snapshot” enabled fails during the snapshot creation step in Cloudera Replication Manager.

**OPSAPS-71414: Permission denied for Ozone replication policy jobs if the source and target bucket names are identical**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

The OBS-to-OBS Ozone replication policy job fails with the com.amazonaws.services.s3.model.AmazonS3Exception: Forbidden or Permission denied error when the bucket names on the source and target clusters are identical and the job uses S3 delegation tokens. Note that the Ozone replication jobs use the delegation tokens when the S3 connector service is enabled in the cluster.

You can use one of the following workarounds to mitigate the issue:

- Use different bucket names on the source and target clusters.
- Set the fs.s3a.delegation.token.binding property to an empty value in ozone_replication_core_site_safety_valve to disable the delegation tokens for Ozone replication policy jobs.

**OPSAPS-71256: The “Create Ranger replication policy” action shows 'TypeError' if no peer exists**

7.13.1

7.13.1.100, 7.13.2.0

When you click target Cloudera Manager Replication Manager Replication Policies Create Replication Policy Ranger replication policy , the TypeError: Cannot read properties of undefined error appears.

**OPSAPS-71067: Wrong interval sent from the Replication Manager UI after Ozone replication policy submit or edit process.**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.2.0

When you edit the existing Ozone replication policies, the schedule frequency changes unexpectedly.

**OPSAPS-70848: Hive external table replication policies fail if the source cluster is using Dell EMC Isilon storage**

7.13.1

7.13.1.100, 7.13.2.0

During the Hive external table replication policy run, the replication policy fails at the Hive Replication Export step. This issue is resolved.

**OPSAPS-71005: RemoteCmdWork uses a singlethreaded executor**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

Replication Manager runs the remote commands for a replication policy through a single-thread executor.

**OPSAPS-59553: SMM's bootstrap server config should be updated based on Kafka's listeners**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

SMM does not show any metrics for Kafka or Kafka Connect when multiple listeners are set in Kafka.

Workaround: SMM cannot identify multiple listeners and still points to bootstrap server using the default broker port (9093 for SASL_SSL). You need to override the bootstrap server URL by performing the following steps:

1. In Cloudera Manager, go to SMM Configuration Streams Messaging Manager Rest Admin Server Advanced Configuration Snippet (Safety Valve)

2. Override bootstrap server URL (hostname:port as set in the listeners for broker) for streams-messaging-manager.yaml.
3. Save your changes.
4. Restart SMM.

**OPSAPS-69317: Kafka Connect Rolling Restart Check fails if SSL Client authentication is required**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

The rolling restart action does not work in Kafka Connect when the ssl.client.auth option is set to required. The health check fails with a timeout which blocks restarting the subsequent Kafka Connect instances.

You can set ssl.client.auth to requested instead of required and initiate a rolling restart again. Alternatively, you can perform the rolling restart manually by restarting the Kafka Connect instances one-by-one and checking periodically whether the service endpoint is available before starting the next one.

**OPSAPS-70971: Schema Registry does not have permissions to use Atlas after an upgrade**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

Following an upgrade, Schema Registry might not have the required permissions in Ranger to access Atlas. As a result, Schema Registry's integration with Atlas might not function in secure clusters where Ranger authorization is enabled.

1. Access the Ranger Console (Ranger Admin web UI).
2. Click the cm_atlas resource-based service.
3. Add the schemaregistry user to the all - * policies.
4. Click  Manage Service Edit Service .
5. Add the schemaregistry user to the default.policy.users property.

**OPSAPS-59597: SMM UI logs are not supported by Cloudera Manager**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

Cloudera Manager does not display a Log Files menu for SMM UI role (and SMM UI logs cannot be displayed in the Cloudera Manager UI) because the logging type used by SMM UI is not supported by Cloudera Manager.

View the SMM UI logs on the host.

**OPSAPS-72298: Impala metadata replication is mandatory and UDF functions parameters are not mapped to the destination**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

Impala metadata replication is enabled by default but the legacy Impala C/C++ UDF's (user-defined functions) are not replicated as expected during the Hive external table replication policy run.

Edit the location of the UDF functions after the replication run is complete. To accomplish this task, you can edit the “path of the UDF function” to map it to the new cluster address, or you can use a script.

**OPSAPS-70713: Error appears when running Atlas replication policy if source or target clusters use Dell EMC Isilon storage**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

You cannot create an Atlas replication policy between clusters if one or both the clusters use Dell EMC Isilon storage.

None

**OPSAPS-72468: Subsequent Ozone OBS-to-OBS replication policy do not skip replicated files during replication**

7.13.1

7.13.1.100

The first Ozone replication policy run is a bootstrap run. Sometimes, the subsequent runs might also be bootstrap jobs if the incremental replication fails and the job runs fall back to bootstrap replication. In this scenario, the bootstrap replication jobs might replicate the files that were already replicated because the modification time is different for a file on the source and the target cluster.

None

**OPSAPS-72470: Hive ACID replication policies fail when target cluster uses Dell EMC Isilon storage and supports JDK17**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.2.0

Hive ACID replication policies fail if the target cluster is deployed with Dell EMC Isilon storage and also supports JDK17.

None

**OPSAPS-73138, OPSAPS-72435: Ozone OBS-to-OBS replication policies create directories in the target cluster even when no such directories exist on the source cluster**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

Ozone OBS-to-OBS replication uses Hadoop S3A connector to access data on the OBS buckets. Depending on the runtime version and settings in the clusters:

- directory marker keys (associated to the parent directories) appear in the destination bucket even when it is not available in the source bucket.
- delete requests of non-existing keys to the destination storage are submitted which result in `Key delete failed` messages to appear in the Ozone Manager log.

The OBS buckets are flat namespaces with independent keys, and the character '/' has no special significance in the key names. Whereas in FSO buckets, each bucket is a hierarchical namespace with filesystem-like semantics, where the '/' separated components become the path in the hierarchy. The S3A connector provides filesystem-like semantics over object stores where the connector mimics the directory behaviour, that is, it creates and optionally deletes the "empty directory markers". These markers get created when the S3A connector creates an empty directory. Depending on the runtime (S3A connector) version and settings, these markers are deleted when a descendant path is created and is not deleted.

Empty directory marker creation is inherent to S3A connector. Empty directory marker deletion behavior can be adjusted using the `fs.s3a.directory.marker.retention` = keep or delete key-value pair. For information about configuring the key-value pair, see Controlling the S3A Directory Marker Behavior.

**OPSAPS-73655: Cloud replication fails after the delegation token is issued**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400

7.13.1.500, 7.13.2.0

HDFS and Hive external table replication policies from an on-premises cluster to cloud fail when the following conditions are true:

1. You choose the Advanced Options Delete Policy Delete Permanently option during the replication policy creation process.
2. Incremental replication is in progress, that is the source paths of the replication are snapshottable directories and the bootstrap replication run is complete.

None

**OPSAPS-75090: Ozone replication policies fail without source proxy user**

7.13.1, 7.13.1.100, 7.13.1.200, 7.13.1.300, 7.13.1.400, 7.13.1.500

An Ozone replication policy with an empty Run on Peer as Username field (The default value for this field is empty) fails with the "java.io.IOException: Error acquiring writer for listing file "ofs://<service id>/user/om/.cm/distcp-staging/<timestamp>/fileList.seq": bucket name 'om' is too short, valid length is 3-63 characters". error message.

If you do not have a source proxy user name to specify in the Run on Peer as Username field, you can enter om as the default user for the replication on the source cluster.

**OPSAPS-75994: Intermittent HBase replication failure because of missing result file**

7.13.2

HBase replication policies fail intermittently during the Check if source tables exist step with the java.lang.IllegalArgumentException: argument "src" is null error message.

Delete and recreate the failed HBase replication policy.

**OPSAPS-72125: The arguments field size exceeds the limit for Hive external table replications**

7.13.2

When replicating a large number of Hive external tables using table filters to target clusters that use PostgreSQL for the Cloudera Manager database, the arguments field of the Hive Data Replication command might exceed the column limit. By default, the arguments column limit in the COMMANDS table is 1,048,676 characters. If the command exceeds the limit, Cloudera Manager cannot persist the command to the database.

Perform the following steps to mitigate this issue:

1. When using table filters, split the policy into multiple chunks so that the Hive Data Replication command created by the chunked policies can be persisted.
2. Increase the arguments column size of the COMMANDS table in the target Cloudera Manager database using the `ALTER TABLE COMMANDS ALTER COLUMN arguments TYPE character varying(10485760);` command. The maximum varchar column size is 10,485,760.

**OPSAPS-73362: Temporary Ozone snapshots are not deleted automatically**

7.13.2

Temporary snapshots used by Ozone incremental data replication for checking the target side changes are not deleted automatically in some error modes.

Currently, the temporary snapshots are generated and reside in the cm-tmp-*[***RANDOM_UUID *** ]* target bucket. These snapshots are deleted immediately after a snapshot-diff calculation. You can delete the snapshots manually only when no replication policy involving this bucket is actively running.

**OPSAPS-73254, OPSAPS-73252: Editing a replication policy can set the user name to an empty string**

7.13.2

On an unsecure (non-Kerberos) cluster, creating or editing a replication policy with an empty Run as Username or Run on Peer as Username field might cause the replication jobs to fail.

Use the Cloudera Manager API to update the fields to contain a null value instead of an empty string.

**DMX-4681: Iceberg replication synchronization step fails for the database created at a custom location without an Ozone key**

7.13.2

The synchronization step of the Iceberg replication command fails during bootstrap replication if you created the database in an Ozone bucket without providing the Ozone key name. The policy fails even if you have configured the Location Mapping field to map the correct Ozone buckets.

- For existing databases or tables that you created without keys, enter the location mapping of the source and target om service IDs in the Location Mapping field in the Iceberg replication policy. For example, ofs://srcomid, ofs://tgtomid.
- For new databases and tables, ensure that you provide a key when you create the database. For example, `CREATE DATABASE db1 LOCATION ‘ofs://omid/volume1/bucket1/db1.db';` and `CREATE EXTERNAL TABLE tb1 (id int) STORED BY ICEBERG LOCATION 'ofs://omid/volume1/bucket1/tb1’`.

**OPSAPS-76854: Cannot edit existing Iceberg replication policies after upgrade**

7.13.2

You cannot edit the existing Iceberg replication policies in Replication Manager UI after you upgrade from Cloudera Manager 7.11.3 or 7.13.1 to 7.13.2.0.

You can use the Cloudera Manager API to view the policy details. To edit the replication policy, use the Cloudera Manager API, or delete and recreate the policy.

**CDPD-63922, CDPD-95711: Atlas replication policies fail when the number of databases and tables exceed 100,000**

7.13.2

When a composite replication policy targets more than 100,000 entities, for example, 100 databases containing 1,000 tables each, the following issues occur:

- A Iceberg replication policy with Atlas metadata migration – The replication policy fails for both bootstrap and incremental jobs. However, the Cloudera Manager Replication Replication Policies page displays the replication policy **Status** as **Successful**, and the Atlas UI does not represent the expected entities.
- A Hive external replication policy with Atlas metadata migration — The replication policy fails for 400 GB (40,000 entities), the **Replication Policies** page displays the replication policy **Status** as **Failed**, and the Atlas UI becomes unresponsive.

**OPSAPS-74398: Ozone and HDFS replication policies might fail when you use different destination proxy user and source proxy user**

7.13.2

HDFS on-premises to on-premises replication fails when the following conditions are true:

- You configure different Run As Username and Run on Peer as Username during the replication policy creation process.
- The user configured in Run As Username does not have the permission to access the source path on the source HDFS.

Ozone replication fails when the following conditions are true:

- FSO-to-FSO replication or an OBS-to-OBS replication with Incremental with fallback to full file listing or Incremental only replication type.
- You configured different Run As Username and Run on Peer as Username during the replication policy creation process.
- The user configured in Run As Username does not have the permission to access the source bucket on the source Ozone.

Provide the same permissions to the user configured in Run As Username as the permissions of Run on Peer as Username on the source cluster.

**OPSAPS-75361: Multiple policies do not start simultaneously**

7.13.2

When multiple Atlas replication policies are scheduled to start at the same time, some policies might fail to initiate. For example, you schedule to run seven Atlas replication policies to run simultaneously, only three might start successfully. The remaining policies are not triggered, remain

in a **None** state, and do not recur, which results in incomplete replication. The **Replication Policies** page displays **None** for these policies.

Do not schedule multiple Atlas replication policies to start at the same time. To avoid this issue, Replication Manager also ensures that a two-minute gap between two Atlas replication policies creation process is maintained to avoid this issue.

**OPSAPS-76832, OPSAPS-70771: Running replication policy runs must not allow you to download the performance reports**

7.13.2

During a replication policy run, the A server error has occurred. See Cloudera Manager server log for details error message appears on the Replication Manager UI, and the Cloudera Manager log shows java.lang.IllegalStateException: Command has no result data when you click:

- Performance Reports Performance Summary *OR* Performance Reports Performance Full on the **Replication Policies** page.
- Download CSV on the **Replication History** page to download any report.

This is because the Replication Manager UI shows the performance report links as enabled and clickable which is incorrect. You can download the reports only after the replication job run is complete.

None

**OPSAPS-76099: Incremental Iceberg replication time exceeds the bootstrap duration**

7.13.2

The incremental Iceberg replication takes a longer time to complete as compared to bootstrap replication for Iceberg replication policies.

None

**OPSAPS-75848: Composite Iceberg and Atlas replication duration takes 10xto 15x times more duration as compared to standalone Atlas replication**

7.13.2

The Atlas replication takes up to 15x the time when run using the composite Iceberg replication as compared to standalone Atlas replication, though the Iceberg data gets replicated in the expected time.

None

**OPSAPS-75853: The history entries display "Partial success" for successful composite replication for Iceberg and Atlas**

7.13.2

The Cloudera Manager Replication Replication History page for a composite Iceberg replication policy displays **Partial success** even when both the Atlas and Iceberg replications were successful which is incorrect.

None

## Behavioral Changes

You can review the changes in certain features or functionalities of Cloudera Manager that have resulted in a change in behavior from the previously released version to this version of Cloudera Manager 7.13.2.

Cloudera Manager 7.13.2 introduces functional adjustments, behavioral updates, and includes all cumulative hotfixes from 7.13.1.100 through 7.13.1.700. For a comprehensive record of all functional adjustments in Cloudera Manager 7.13.1.x, see Behavioral Changes 7.13.1.x.

## Cloudera Manager 7.13.2

**Summary: Enhanced the security of Spark by implementing new default configuration settings**

**Previous behavior:**

1. spark.ui.enabled=true: possible initiation of an HTTP service that can be accessed from external hosts.
2. spark.io.encryption.keySizeBits=128: the default Spark keySizeBits

**New behavior:**

1. spark.ui.enabled=false: preventing initiation of an HTTP service that can be accessed from external hosts.
2. spark.io.encryption.keySizeBits=256: the default Spark keySizeBits has been increased from 128 to 256

**Summary: Cloudera Manager 7.13.2 introduces changes to its underlying SAML library, impacting Service Provider (SP) metadata signing, response binding options, and message signing algorithms.**

**Previous behavior:**

- Metadata Signing: The SAML library supported the signing of the Service Provider's (Cloudera Manager) metadata.
- Response Binding: SAML Response Binding supported only POST & ARTIFACT, with the default set to POST.
- Message Signing Algorithm: You could select the Message Signing Algorithm from a list: RSA_SHA1, RSA_SHA256, RSA_SHA384, RSA_SHA512.

**New behavior:**

- Metadata Signing: The SAML library no longer supports the signing of the Service Provider's (Cloudera Manager) metadata.
- Response Binding: SAML Response Binding will now support only POST & REDIRECT, with the default set to POST.
- Message Signing Algorithm: The Message Signing Algorithm defaults to RSA_SHA256 (which is secure), and Cloudera Manager will provide no options to select from the UI.

**Summary: When Cloudera Manager runs in FIPS-enabled mode, it does not use the JVM's default cace rts truststore, as it is in a non-FIPS JKS format.**

**Previous behavior:**

In standard (non-FIPS) mode, Cloudera Manager relies on the JVM's default cacerts truststore, which is in JKS format. This allows Cloudera Manager to automatically trust the default CA certificates shipped with the JDK and establish secure connections to external endpoints such as S3 or Cloudera repositories.

**New behavior:**

In FIPS mode, only FIPS-compliant keystore formats (such as BCFKS) are permitted. Because the default cacerts is in JKS format, Cloudera Manager skips loading it to prevent startup failures and instead uses the Cloudera Manager-provided truststore exclusively. This ensures consistent and reliable operation under FIPS compliance, but means that Cloudera Manager does not automatically trust certificates present only in the JVM's default cacerts.

As a result, connections to certain external endpoints (for example, public cloud services, S3, Cloudera download URLs) might fail unless their certificates are also present in the Cloudera Manager-provided truststore.

To preserve the default JVM truststore behavior in FIPS mode, convert the JKS-formatted cacerts to BCFKS using keytool. For more information, see Preserving the default JVM truststore behavior in FIPS mode.

**Summary: Removed the hbase.secure.rpc.engine configuration property**

**Previous behavior:**

The hbase.secure.rpc.engine configuration property is obsolete because HBase performs replication securely by default.

**New behavior:**

The hbase.secure.rpc.engine configuration property is removed from the Cloudera Manager because it is obsolete and no longer needed.

**The default value of the Region Mover Threads configuration is updated**

**Previous behavior:**

The current default value for the configuration HBase Configuration Region Mover Threads is set to 1

**New behavior:**

As only a single thread is utilized for region movement, the overall operation exhibits poor performance. The default value is updated 20, which is proposed to leverage 20 region mover threads and is expected to yield substantial performance improvements.

For more information see Rolling Restart.

**HBase BucketCache configuration names and descriptions are incorrect in Cloudera Manager**

**Previous behavior:**

The display names and descriptions of the BucketCache configuration names are as follows:

- hbase.rs.evictblocksonclose = HBase region server cache data blocks on read (Cache data blocks on read)
- hbase.block.data.cacheonread = HBase region server evict blocks of a file cache when the file is closed (Evict all blocks of a given file from the block cache when the file is closed)

**New behavior:**

The display names and descriptions are corrected in Cloudera Manager for the following configuration items:

- hbase.rs.evictblocksonclose = HBase RegionServer evict blocks on closed (Evict all blocks of a given file from the block cache when the region is closed)
- hbase.block.data.cacheonread = HBase RegionServer cache data blocks on read (Cache data blocks on read)

**HBOSS related configuration items are removed from Cloudera Manager**

**Previous behavior:**

Cloudera Manager contains the following HBOSS related configuration items.

- fs.hboss.fs.s3a.impl
- fs.hboss.sync.impl
- fs.s3a.connection.maximum
- fs.s3a.threads.max

**New behavior:**

The above items are removed from HBase Cloudera Manager configuration because HBOSS support is deprecated.

**The BucketCache related configuration items are dynamically configurable in Cloudera Manager**

**Previous behavior:**

The following HBase BucketCache related configuration items were previously not dynamically configurable in Cloudera Manager, and users had to configure them using the advanced configuration snippet (safety valve).

- hbase.bucketcache.acceptfactor

- hbase.bucketcache.minfactor

**New behavior:**

You can now configure these items dynamically in Cloudera Manager. The change takes effect immediately without requiring an HBase service restart.

**HBase supports Log4J2 for logging**

**Previous behavior:**

HBase used Log4j 1.x for its logging framework.

**New behavior:**

HBase now uses Log4j2 due to its enhanced performance, improved features, and ongoing security support.

**Summary: Allowing configurable File Descriptor (FD) limits in supervisord**

**Previous behavior:**

Each service role ran with the File Descriptor (FD) limit you explicitly configured in Cloudera Manager, or with the default limit if you set none.

**New behavior:**

The new behavior allows administrators to increase the FD limit globally across all managed services by modifying the relevant supervisord service file or system.conf, regardless of individual role settings.

Configuration Rules: The FD limit you define in /usr/lib/systemd/system/cloudera-scm-supervisord .service or /etc/systemd/system.conf (depending on the Cloudera Manager version) applies to the supervisord daemon itself. supervisord inherits this limit for service roles only when you configure no explicit FD limit for that role.

**Important:** A role's effective FD limit is exactly the value you configure for it in Cloudera Manager. The supervisord or system-wide FD setting only takes effect when a role does not have its own configured FD limit.

**Summary: Apache Ranger policies for Kafka topics and the consumer group used by Atlas asynchronous import**

**Previous behavior:**

Cloudera Manager did not add default Apache Ranger policies that grant the atlas user permission to create, edit, or delete Kafka topics with the ATLAS_IMPORT_ prefix or to access the atlas_import consumer group.

**New behavior:**

Cloudera Manager adds default Apache Ranger policies so the atlas user can create, configure, alter, publish, and delete Kafka topics with the ATLAS_IMPORT_ prefix and can use the atlas_import consumer group for Atlas asynchronous import.

**Summary: Removed EventCounter from Hadoop**

**Previous behavior:**

The EventCounter functionality was deprecated and later removed by Apache Hadoop.

**New behavior:**

The occurrences of EventCounter have been removed from the Java code and log4j configuration files.

**Summary: Component-level custom Java home configuration removed for Kafka, Schema Registry, SMM, and SRM**

**Previous behavior:**

You could configure a component-specific Java home for Kafka, Schema Registry, Streams Messaging Manager (SMM), and Streams Replication Manager (SRM).

**New behavior:**

The component-level custom Java home configuration options are removed for Kafka, Schema Registry, SMM, and SRM. These services now use the host-level java_home configuration. If you previously set a component-specific Java home for any of these services, verify the host-level java_home setting after upgrading.

| In a Cloudera Manager 7.13.2.0 Cluster | Effective FD Limit |
|---|---|
| You configure HDFS roles with an FD limit of 72K. | They run with 72K. |
| You configure Ozone roles with an FD limit of 48K. | They run with 48K. |
| The supervisord service file sets an FD limit of 64K. | supervisord itself uses 64K. |
| You do not configure Hive roles explicitly. | Hive inherits the 64K limit from supervisord. |

If you want to increase the File Descriptor (FD) limit across all services managed by Cloudera Manager, you should perform the following steps:

**Option A: If Cloudera Manager version is 7.13.2 or higher**

1. Go to /usr/lib/systemd/system/cloudera-scm-supervisord.service.
2. Make the change that is prescribed in the cloudera-scm-supervisord.service file to increase the FD limit across services.
3. Then run systemctl daemon-reload followed by systemctl restart cloudera-scm-supervisord.

**Option B: If Cloudera Manager version is lower than 7.13.2**

1. Go to /etc/systemd/system.conf.
2. This file will contain #DefaultLimitNOFILE=. You need to uncomment this line (if it is commented) and set an FD value of your choice.
3. Then run systemctl daemon-reload followed by systemctl restart cloudera-scm-supervisord.

After you implement this change, the FD limit will apply across all services managed by Cloudera Manager.

# Cloudera Manager API Compatibility Matrix

This document provides the Cloudera Manager API Compatibility Matrix to ensure seamless communication between automated clients and the Cloudera Manager server. This matrix defines the required Python environments and minimum Cloudera Manager versions necessary for specific API client versions and REST endpoints to function correctly.

To maintain system stability and avoid breaking changes, Cloudera Manager enforces strict versioning for its API clients and REST interfaces. This compatibility framework allows administrators to upgrade Cloudera Manager while understanding the impact on existing automation scripts and custom integrations.

- Python Support: Each cm-client version requires a specific Python runtime to execute successfully.
- Version Alignment: Upgrading to a newer REST API version (for example, v57 to v58) typically requires a corresponding minimum Cloudera Manager backend version to support the newer features and data structures.
- Backward Compatibility: While newer Cloudera Manager versions generally support older API clients, using an outdated cm-client with a modern Cloudera Manager server might limit access to the latest platform capabilities.

## Cloudera Manager Client Compatibility

The following table identifies the supported Python environments and the minimum Cloudera Manager version required for each cm-client release.

| cm-client Version | Supported Python Versions | Minimum Cloudera Manager / Runtime Versions Supported |
|---|---|---|
| 58.0.2 | 3.11 | 7.13.2 / 7.3.2.0 |

### REST API Version Support

This table maps the REST API endpoint versions to the minimum Cloudera Manager version required to host them.

| REST API Version | Minimum Cloudera Manager Version Supported |
|---|---|
| v58 | 7.13.2.0 |
| v57 | 7.13.1.0 |
| v54 | 7.11.3.0 |

## Cloudera Manager API Migration Notes

This document outlines the transition of the Cloudera Manager REST API from Swagger 2 to OpenAPI 3.0 starting with Cloudera Manager version 7.13.2.0. It details necessary code changes for Java and Python clients, including data type shifts and authentication changes, ensuring your integrations remain compatible while leveraging the improved performance and strict typing of the new specification.

### Overview and Compatibility

The Cloudera Manager server traditionally provides its public REST API through Swagger 2 specifications.

Starting with Cloudera Manager 7.13.2.0, the REST API moves to the OpenAPI 3.0 specification because Java 8 is no longer supported. The OpenAPI specification introduces differences that require changes in the updated REST API implementation.

- Client Compatibility: Older API clients continue to work with the Cloudera Manager 7.13.2.0 server. You only need to migrate your code if you require access to new endpoints introduced in Cloudera Manager 7.13.2.0.
- Python Support: This version officially removes support for Python 2.x.
- Data Type Changes: A key change in OpenAPI 3.0 is the removal of the file type. Because code generators require this type to create client code, this change affects all API endpoints that currently reference file.

Cloudera offers pre-built client bindings in Java and Python for the REST API. Use the following steps to migrate your code to the new API clients.

### Java Client Updates

**Numerical Data Type Shift: BigDecimal to Long**

In the previous client, many method signatures use the BigDecimal type for numerical arguments. Because OpenAPI 3.0 specifications require precise type definitions, the current client uses Long in place of BigDecimal.

**AuditsResourceApi Change: Stream Audits**

The AuditsResourceApi.streamAudits method now returns a String instead of a File. This change follows OpenAPI 3.0 compliance rules and might impact performance.

Previous Signature:

```
java.io.File streamAudits(String endTime, BigDecimal maxResults,
 String query, BigDecimal resultOffset, String startTime)
```

Current Signature:

```
String streamAudits(String endTime, BigDecimal maxResults, String
 query, BigDecimal resultOffset, String startTime)
```

Example 1: Use the new type directly.

```
AuditsResourceApi auditsApi = new AuditsResourceApi();
String auditsJson = auditsApi.streamAudits("now", BigDecimal.valu
eOf(100), null,
    BigDecimal.valueOf(0), null);
// deserialize auditsJson ...
```

Example 2: Save to File like the previous client.

```
AuditsResourceApi auditsApi = new AuditsResourceApi();
Response response = auditsApi
    .streamAuditsCall("now", BigDecimal.valueOf(100), null,
        BigDecimal.valueOf(0), null, null, null)
    .execute();
File file = Configuration.getDefaultApiClient().downloadFileFr
omResponse(response);
```

**Important:**

The API implementation differs because of OpenAPI 3.0 compliance. Performance results might vary from previous release versions.

**ControlPlanesResourceApi Change: Get Manifest JSON**

The ControlPlanesResourceApi.getManifestJson method now returns a String instead of a File. This update follows OpenAPI 3.0 compliance rules.

Previous Signature:

```
java.io.File getManifestJson(ApiRemoteRepoUrl remoteRepoUrl)
```

Current Signature:

```
String getManifestJson(ApiRemoteRepoUrl remoteRepoUrl)
```

Example: Save to File

Follow the streamAudits example to save the response data to a file. Use the getManifestJsonCall method to execute the call and download the file from the response.

**Important:**

The API implementation differs because of OpenAPI 3.0 compliance. Performance results might vary from previous release versions.

**ControlPlanesResourceApi Change: Get Log Content**

The ControlPlanesResourceApi.getLogContent method now returns a String and accepts a Long type for the command ID. This update follows OpenAPI 3.0 compliance rules.

Previous Signature:

```
java.io.File getLogContent(java.math.BigDecimal commandId)
```

Current Signature:

```
String getLogContent(Long commandId)
```

Example: Save to File

Use the getLogContentCall method to execute the call and save the response data to a file.

```
ControlPlanesResourceApi controlPlanesApi = new ControlPlanesRes
ourceApi();
Response response = controlPlanesApi
    .getLogContentCall(123L, null, null)
    .execute();
File file = Configuration.getDefaultApiClient().downloadFileFr
omResponse(response);
```

**Important:**

The API implementation differs because of OpenAPI 3.0 compliance. Performance results might vary from previous release versions.

## Python Client Updates

### Python Client Setup and Configuration

The current client code no longer accepts a URL as an argument for the cm_client.ApiClient constructor. You must use a Configuration object to set the host and configure Basic Authentication.

Follow these steps to configure the connection and use Basic Authentication for all subsequent requests.

Example: Configure the Connection

```
configuration = cm_client.Configuration()
configuration.host = 'https://cm_server_host:7183/api/v58'
configuration.username = 'admin'
configuration.password = 'admin'

# Encode and set Basic Auth header
auth_str = base64.b64encode(f"{configuration.username}:{configura
tion.password}".encode()).decode("utf-8")
api_client = cm_client.ApiClient(configuration)
api_client.default_headers["Authorization"] = f"Basic {auth_str}"
```

### Python ApiException Import Change

Previous Python clients import ApiException within ApiClient automatically. The current client no longer performs this import. To fix this, update your code with one of the following options:

Option 1: Add a Direct Import

Add this line to your code: import cm_client.rest.ApiException

Option 2: Use the rest Namespace

If your code already imports cm_client, update all ApiException references to: rest.ApiException

### Persistent File Support for Legacy APIs

Cloudera simplified the transition for APIs that previously returned a 'file', meaning you do not need to make any changes.

The new compatibility layer automatically handles file returns so your current code continues to work without any manual updates.

To prevent widespread breaks for users following the removal of file from OpenAPI, Cloudera built a code layer. This layer allows you to use client APIs exactly as you did in the past. For specific APIs, a transparent adapter handles the task of reading response data and writing it to a file.

This ensures the caller avoids any modifications. The logic within this adapter replicates the exact process found in the previous client code.

**Important:** If you view Python documentation (for example, through the help function), the text might show a return type of str even though the actual output remains a file.

**Persistent Text Support for Legacy APIs**

Cloudera provides a new internal adapter that automatically processes UTF-8 text responses to ensure current code remains compatible without any manual updates.

For the Simplified API Transition, you do not need to make any changes to APIs that provide text content.

Cloudera maintains several API endpoints that return a 'text/plain' content type. During a call, the previous client performs a UTF-8 decode automatically. New clients show an issue; they return a string literal with the representation of the bytes.

To bypass this, a user must call the API with _preload_content=False and process the response manually, or add a line to evaluate the string and then call decode. Cloudera now uses an adapter to maintain compatibility for these APIs. This adapter performs the evaluation and UTF-8 decoding automatically.

## Fixed Common Vulnerabilities and Exposures in Cloudera Manager 7.13.2

Common Vulnerabilities and Exposures (CVE) that are fixed in Cloudera Manager 7.13.2.

### Cloudera Manager 7.13.2.0

| CVEs | Package Name |
|---|---|
| CVE-2024-53990 | async-http-client |
| CVE-2018-10237 | Guava |
| CVE-2024-47561 | Apache avro |
| CVE-2023-39410 | Apache avro |
| CVE-2012-6708 | jquery |
| CVE-2015-9251 | jquery |
| CVE-2019-11358 | jquery |
| CVE-2020-11022 | jquery |
| CVE-2020-11023 | jquery |
| CVE-2020-7656 | jquery |
| CVE-2011-4969 | jquery |
| CVE-2023-44487 | Netty |
| CVE-2024-21634 | ion-java |
| CVE-2025-8916 | Bouncycastle |
| CVE-2024-23114 | Apache Camel |
| CVE-2024-22369 | Apache Camel |
| CVE-2024-22371 | Apache Camel |
| CVE-2025-27636 | Apache Camel |
| CVE-2025-29891 | Apache Camel |
| CVE-2023-34442 | Apache Camel |

| CVEs | Package Name |
|---|---|
| CVE-2024-25710 | Commons-compress |
| CVE-2024-26308 | Commons-compress |
| CVE-2025-48976 | Commons-fileupload |
| CVE-2012-5783 | Commons-httpclient |
| CVE-2020-13956 | Commons-httpclient |
| CVE-2024-47554 | Commons-io |
| CVE-2025-23184 | Apache CXF |
| CVE-2020-36843 | Eddsa |
| CVE-2025-5878 | Esapi |
| CVE-2024-13009 | Eclipse Jetty |
| CVE-2024-8184 | Eclipse Jetty |
| CVE-2017-7536 | Hibernate-validator |
| CVE-2025-35036 | Hibernate-validator |
| CVE-2019-10219 | Hibernate-validator |
| CVE-2023-1932 | Hibernate-validator |
| CVE-2020-10693 | Hibernate-validator |
| CVE-2024-36114 | Aircompressor |
| CVE-2018-1000873 | Jackson Databind |
| CVE-2017-15095 | Jackson Databind |
| CVE-2017-17485 | Jackson Databind |
| CVE-2017-7525 | Jackson Databind |
| CVE-2018-11307 | Jackson Databind |
| CVE-2018-14718 | Jackson Databind |
| CVE-2018-14719 | Jackson Databind |
| CVE-2018-7489 | Jackson Databind |
| CVE-2019-14379 | Jackson Databind |
| CVE-2019-14540 | Jackson Databind |
| CVE-2019-14892 | Jackson Databind |
| CVE-2019-16335 | Jackson Databind |
| CVE-2019-16942 | Jackson Databind |
| CVE-2019-16943 | Jackson Databind |
| CVE-2019-17267 | Jackson Databind |
| CVE-2019-17531 | Jackson Databind |
| CVE-2019-20330 | Jackson Databind |
| CVE-2020-8840 | Jackson Databind |
| CVE-2020-9547 | Jackson Databind |
| CVE-2020-9548 | Jackson Databind |
| CVE-2020-10673 | Jackson Databind |
| CVE-2018-5968 | Jackson Databind |

| CVEs | Package Name |
|---|---|
| CVE-2020-10650 | Jackson Databind |
| CVE-2020-24616 | Jackson Databind |
| CVE-2020-24750 | Jackson Databind |
| CVE-2020-35490 | Jackson Databind |
| CVE-2020-35491 | Jackson Databind |
| CVE-2020-36179 | Jackson Databind |
| CVE-2020-36180 | Jackson Databind |
| CVE-2020-36181 | Jackson Databind |
| CVE-2020-36182 | Jackson Databind |
| CVE-2020-36183 | Jackson Databind |
| CVE-2020-36184 | Jackson Databind |
| CVE-2020-36185 | Jackson Databind |
| CVE-2020-36186 | Jackson Databind |
| CVE-2020-36187 | Jackson Databind |
| CVE-2020-36188 | Jackson Databind |
| CVE-2020-36189 | Jackson Databind |
| CVE-2021-20190 | Jackson Databind |
| CVE-2018-12022 | Jackson Databind |
| CVE-2019-12086 | Jackson Databind |
| CVE-2019-14439 | Jackson Databind |
| CVE-2020-36518 | Jackson Databind |
| CVE-2022-42003 | Jackson Databind |
| CVE-2022-42004 | Jackson Databind |
| CVE-2019-12384 | Jackson Databind |
| CVE-2019-12814 | Jackson Databind |
| CVE-2019-10172 | Jackson-mapper-asl |
| CVE-2024-22201 | Eclipse Jetty |
| CVE-2024-9823 | Eclipse Jetty |
| CVE-2025-59340 | jinjava |
| CVE-2026-25526 | jinjava |
| CVE-2020-13949 | Apache Thrift |
| CVE-2018-1320 | Apache Thrift |
| CVE-2019-0205 | Apache Thrift |
| CVE-2019-0210 | Apache Thrift |
| CVE-2018-11798 | Apache Thrift |
| CVE-2025-68161 | log4j-core |
| CVE-2025-12183 | lz4-java |
| CVE-2025-66566 | lz4-java |
| CVE-2025-31672 | Apache Poi |

| CVEs | Package Name |
|---|---|
| CVE-2022-3510 | protobuf-java |
| CVE-2010-5312 | jquery-ui |
| CVE-2016-7103 | jquery-ui |
| CVE-2021-41182 | jquery-ui |
| CVE-2021-41183 | jquery-ui |
| CVE-2021-41184 | jquery-ui |
| CVE-2022-31160 | jquery-ui |
| CVE-2024-38820 | Spring Framework |
| CVE-2024-38808 | Spring Framework |
| CVE-2025-41249 | Spring Framework |
| CVE-2025-41242 | Spring Framework |
| CVE-2025-22228 | Spring Security |
| CVE-2024-38821 | Spring Security |
| CVE-2024-38809 | Spring Framework |
| CVE-2024-38828 | Spring Framework |
| CVE-2024-38816 | Spring Framework |
| CVE-2023-48795 | sshj |
| CVE-2020-26870 | DOMPurify |
| CVE-2021-40690 | xmlsec |
| CVE-2023-44483 | xmlsec |
| CVE-2015-1796 | xmltooling |
| CVE-2024-47072 | xstream |

# Deprecation notices in Cloudera Manager 7.13.2

Certain features and functionalities have been removed or deprecated in Cloudera Manager 7.13.2. You must review these items to understand whether you must modify your existing configuration. You can also learn about the features that will be removed or deprecated in the future release to plan for the required changes.

## Terminology

Items in this section are designated as follows:

**Deprecated**

Technology that Cloudera is removing in a future Cloudera Manager release. Marking an item as deprecated gives you time to plan for removal in a future Cloudera Manager release.

**Moving**

Technology that Cloudera is moving from a future Cloudera Manager release and is making available through an alternative Cloudera offering or subscription. Marking an item as moving gives you time to plan for removal in a future Cloudera Manager release and plan for the alternative Cloudera offering or subscription for the technology.

**Removed**

Technology that Cloudera has removed from Cloudera Manager and is no longer available or supported as of this release. Take note of technology marked as removed since it can potentially affect your upgrade plans.

## Deprecation Notices for Cloudera Manager

Certain features and functionality are deprecated or removed in Cloudera Manager 7.13.2. You must review these changes along with the information about the features in Cloudera Manager that will be removed or deprecated in a future release.

## Platform, OS, and Environment Support

Cloudera Manager 7.13.2 removes support for several operating systems, databases, Python versions, and Java Development Kits (JDKs). To ensure a successful upgrade, verify that your environment meets these new requirements.

### Database Support

Starting with Cloudera Manager 7.13.2, Cloudera no longer supports the following databases:

- PostgreSQL 13
- Oracle 19c (including 19c RAC)

### Operating System

Starting with Cloudera Manager 7.13.2, Cloudera no longer supports the following operating systems:

- SLES 15 SP4
- Ubuntu 20.04

### Python Support

Cloudera Manager 7.13.2 limits Python compatibility to modern versions, and now only supports Python 3.11. Consequently, Cloudera has:

- Removed support for Python 3.8
- Removed support for Python 3.9
- Removed support for Python 3.10

### JDK Support

To improve performance and security, Cloudera Manager 7.13.2 consolidates Java support, and now only supports JDK 17. Consequently, Cloudera has:

- Removed support for Azul JDK 8, Open JDK 8, and Oracle JDK 8
- Removed support for Azul JDK 11, Open JDK 11, and Oracle JDK 11