MySQL 8.4 Release Notes

Abstract

This document contains release notes for the changes in MySQL 8.4. For information about changes in a different version of MySQL, see the release notes for that version.

For additional MySQL 8.4 documentation, see the MySQL 8.4 Reference Manual, which includes an overview of features added in MySQL 8.4 (What Is New in MySQL 8.4 since MySQL 8.0), and discussion of upgrade issues that you may encounter while upgrading.

MySQL platform support evolves over time; please refer to https://www.mysql.com/support/supportedplatforms/database.html for the latest updates.

Updates to these notes occur as new product features are added, so that everybody can follow the development process. If a recent version is listed here that you cannot find on the download page (https://dev.mysql.com/downloads/), the version has not yet been released.

The documentation included in source and binary distributions may not be fully up to date with respect to release note entries because integration of the documentation occurs at release build time. For the most up-to-date release notes, please refer to the online documentation instead.

For legal information, see the Legal Notices.

For help with using MySQL, please visit the MySQL Forums, where you can discuss your issues with other MySQL users.

Document generated on: 2026-04-28 (revision: 31150)

Table of Contents

Preface and Legal Notices 1
Changes in MySQL 8.4.9 (2026-04-21) 3
Changes in MySQL 8.4.8 (2026-01-20) 6
Changes in MySQL 8.4.7 (2025-10-21) 8
Changes in MySQL 8.4.6 (2025-07-22) 10
Changes in MySQL 8.4.5 (2025-04-15) 15
Changes in MySQL 8.4.4 (2025-01-21) 21
Changes in MySQL 8.4.3 (2024-10-15) 27
Changes in MySQL 8.4.2 (2024-07-23) 35
Changes in MySQL 8.4.1 (2024-07-01) 35
Changes in MySQL 8.4.0 (2024-04-30) 42
Changes in MySQL 8.3.0 (2024-01-16) 62
Changes in MySQL 8.2.0 (2023-10-25) 75
Changes in MySQL 8.1.0 (2023-07-18) 89

Preface and Legal Notices

This document contains release notes for the changes in MySQL 8.4.

Legal Notices

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Changes in MySQL 8.4.9 (2026-04-21)

Note

These release notes were created with the assistance of MySQL HeatWave GenAI.

- Atomic DDL Notes
- Audit Log Notes
- Compilation Notes
- SQL Function and Operator Notes
- InnoDB Notes
- JSON Notes
- Optimizer Notes
- Packaging Notes

- Performance Schema Notes
- Thread Pool Notes
- Bugs Fixed

Atomic DDL Notes

- It was not possible to drop columns, on a table with virtual columns, using `LOCK=NONE`. (Bug #83557, Bug #24962142)

Audit Log Notes

- Fixed an issue related to processing certain gzip files. The MySQL Server has been updated to relax the `.gz` header checks, allowing ordinary gzip files to be processed. Errors were returned similar to the following:

  ```
  Invalid audit log file content
  ```

  (Bug #38980223)

Compilation Notes

- The included `zlib` library has been upgraded to version 1.3.2. (Bug #38987448)

SQL Function and Operator Notes

- Fixed an issue relating to the `DEFAULT()` function. (Bug #39057054)

InnoDB Notes

- Memory usage during FTS index construction for large tables has been optimized. (Bug #39040226)
- Fixed an issue relating to multi-value indexes. (Bug #39040128)
- Fixed an issue in the parallel reader. (Bug #39033858)
- `dict_sdi_create_idx_in_mem` function added columns in a different order than expected, with the `compressed_len` and `uncompressed_len` fields swapped. (Bug #38810801)
- Under certain circumstances, running `CREATE INDEX` with a high value for `--innodb_parallel_read_threads` could cause the disk space to fill up, leading to disk space exhaustion. (Bug #38370155)
- Fixed an issue related to `TRUNCATE TABLE`. (Bug #38169053)
- Under certain circumstances, when calculating the maximum possible index record size, an assertion failure could occur. (Bug #85060, Bug #25579578)

JSON Notes

- Fixed an issue relating to processing of JSON files. (Bug #39000847)

Optimizer Notes

- Fixed an issue relating to optimized prepared DELETE and UPDATE statements. (Bug #39071552)

- Fixed an issue relating to Batched Key Access path and partitioning. (Bug #38947039)
- Fixed an issue relating to query management. (Bug #38928287)
- Fixed an issue relating to memory management. (Bug #38573278)
- Queries using semi join with materialization may return incorrect results due to a missing condition.

  Our thanks to Jingqi Tian and the team at Alibaba. (Bug #38110792)
- Fixed an issue relating to the Optimizer's SQL planner. (Bug #35634700)
- When errors and warnings occured during background histogram updates, the conditions in the diagnostic area were not cleared, leading to future background histogram updates emitted the same stale conditions to the error log.

  Our thanks to Tony Chen and the team at Amazon for the contribution. (Bug #119922, Bug #38983545)
- A previous fix changed range analysis for non-binary string collations, when a string constant in a range predicate was longer than the indexed column, no index range was formed, unless the collation was known to be safe. This resulted in queries, which should have used efficient range plans, using index or table scans.

  Our thanks to Yakir Gibraltar for the contribution. (Bug #118009, Bug #37849917)

  References: This issue is a regression of: Bug #35169384.
- A histogram generated by `ANALYZE TABLE UPDATE HISTOGRAM ON col_name` was not be accepted by `ANALYZE TABLE UPDATE HISTOGRAM ON col_name USING DATA ...` if the histogram was built on a `bigint` column with values outside the `uint32` range.

  Our thanks to Tianfeng Li for the contribution. (Bug #116611, Bug #37269033)
- In an explain expanded query, join order hints were not printed with a valid syntax.

  Our thanks to Kaiwang Chen for the contribution. (Bug #116084, Bug #37053765)
- Conditions on const tables and on previously joined tables were not consistently taken into account when building ranges in `get_quick_record_count()`. This could lead to suboptimal or incorrect range estimates, especially when range predicates depended on values from const or earlier-joined tables.

  Our thanks to Xingyu Yang for the contribution. (Bug #112737, Bug #35912840)

Packaging Notes

- Building with PGO (Profile Guided Optimization) is now also supported for RPM builds on SLE/openSUSE and Fedora, by adding `--define=with_pgo 1` to the rpmbuild command line. (Bug #38915967)
- For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated to version 3.5.5. For more information, see OpenSSL 3.5 Series Release Notes. (Bug #38867109, WL #17209)

Performance Schema Notes

- It is now possible to enable or disable Telemetry meters from the command line or configuration file using the `performance-schema-meter` parameter, on server startup.

See Server Meters. (Bug #36698082)

Thread Pool Notes

- Under high load, when using Thread Pool, connections could hang. (Bug #38693319)

Bugs Fixed

- Fixed an issue relating to Information Schema privileges. (Bug #35243416, Bug #37387633)
- The SQL parser consumed a large amount of memory when parsing very large queries with many large `IN` clauses.

  As of this release, the SQL parser's memory management has been improved. (Bug #105004, Bug #33390851)

Changes in MySQL 8.4.8 (2026-01-20)

Note

These release notes were created with the assistance of MySQL HeatWave GenAI.

- Compilation Notes
- Configuration Notes
- InnoDB Notes
- Optimizer Notes
- Packaging Notes
- Pluggable Authentication
- Security Notes
- Bugs Fixed

Compilation Notes

- The cmake macro `MYSQL_CHECK_PROTOBUF` did not properly handle multiple calls when using system protobuf. Errors were returned similar to the following:

  ```
  ADD_LIBRARY cannot create imported target
  "ext::libprotobuf" because another target with the same
  name already exists.
  ```

  (Bug #38417986)

Configuration Notes

- The `mysqlx_port` server variable was not persisted in the `my.ini` file when set through the **Type and Networking** configuration page. (Bug #38728671)

InnoDB Notes

- **InnoDB:** InnoDB redo logging error messages now include the current LSN and redo log capacity information. A new warning code `ER_IB_WRN_REDO_DISABLED_INFO` is added which includes the current LSN. A new error code `ER_IB_MSG_LOG_WRITER_WAIT_ON_NEW_LOG_FILE_INFO` is added to include the current log capacity and the current log capacity used. The `MONITOR` output has been enhanced to include redo log capacity details and current log capacity.

  The warning `ER_IB_WRN_REDO_DISABLED` was removed, as was the error `ER_IB_MSG_LOG_WRITER_WAIT_ON_NEW_LOG_FILE`. (Bug #37645185)

- Fixed an issue related to bulk inserts. (Bug #38208188)

Optimizer Notes

- Fixed an issue relating to Common Table Expressions (CTE). (Bug #38573285)
- Fixed an issue related to executing specific SQL queries. (Bug #38448700)
- Fixed an issue relating to `SHOW CREATE TABLE`. (Bug #38298692)
- Queries which used REGEXP took longer to execute as prepared statements than as direct queries. (Bug #114056, Bug #36326728)

Packaging Notes

- For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated to version 3.0.18. For more information, see OpenSSL 3.0 Series Release Notes. (Bug #38632932)

Pluggable Authentication

- Connecting to a MySQL server with a non-existing user now consistently returns an "Access denied for user" error message, regardless of the username length or MySQL version. Previously, errors were returned similar to the following:

  ```
  ERROR 1045 (28000): Access denied for user
  'foo'@'localhost' (using password: NO), ERROR
  1524 (HY000): Plugin 'mysql_native_password' is not
  loaded
  ```

  (Bug #36527984)

Security Notes

- Fixed an issue relating to authentication. (Bug #118447, Bug #38077617)

Bugs Fixed

- **InnoDB:** Under certain circumstances, when using the `que_eval_sql` interface, a race condition could occur. (Bug #118705, Bug #38310595)
- Running SET PERSIST on a system variable after an upgrade can result in duplicate variable entries across different sections in `mysqld-auto.cnf`. (Bug #38680162)
- Binary log purged before persisted `binlog_expire_*` options were loaded. (Bug #38554467)

- When 2 of 6 instances in a geographically dispersed InnoDB Cluster lost connectivity, the primary server became unresponsive, causing the `COMMIT` and `INSERT` operations to become unresponsive as well. (Bug #38380392)
- `mysqldump`'s `--order-by-primary` option sorted data by every index on the table, instead of just sorting by the primary key. (Bug #38284832)
- Fixed several issues relating to Thread Pool blocking connections from closing. (Bug #38170188, Bug #36782728, Bug #38549372)
- GTID gaps were generated because of the `replica- skip-errors` option. (Bug #28590993)
- If server was started on Windows platforms with --skip-grant-tables, privileges were not checked and remote connections were not permitted. As a result, the server might not start. (Bug #118805, Bug #38328780)

Changes in MySQL 8.4.7 (2025-10-21)

Note

These release notes were created with the assistance of MySQL HeatWave GenAI.

- Audit Log Notes
- Compilation Notes
- SQL Function and Operator Notes
- InnoDB Notes
- Optimizer Notes
- Packaging Notes
- Performance Schema Notes
- Bugs Fixed

Audit Log Notes

- Fixed an issue related to audit log strategy. (Bug #38223011)
- Improved performance related to adding multiple `general_sql_command.str` fields to an Enterprise Audit filter. (Bug #38053242)
- Fixed an issue related to `audit_log_read` performance. (Bug #36281295, Bug #38275868)

Compilation Notes

- OpenSSL 3 is now used on Solaris. (Bug #38193362)
- Upgraded the bundled googletest library to version 1.17. (Bug #38152017)
- Upgraded the bundled libbacktrace library to the latest version (April 2025). (Bug #38151970)
- Warnings raised by Clang 20 for non-trivially copyable types, deprecated literal operators, and incorrect `main` function declarations are no longer generated. (Bug #37785251)

SQL Function and Operator Notes

- Fixed an issue related to executing certain UPDATE statements. (Bug #37590580)

  References: This issue is a regression of: Bug #31562881.

InnoDB Notes

- Creating a thread for parallel scan may fail, resulting in assertion failures when it fell back to single-thread mode. Errors were returned similar to the following:

  ```
  Assertion failure: row0pread.h:306:active >= n_threads
  ```

  (Bug #38325137)

- Under certain circumstances, a virtual index rollback could fail on 32-bit builds of MySQL Server. (Bug #38167527)

- If the buffer pool was very large, it could lead to a large number of chunks required by each buffer pool instance, which could fail if the operating system was unable to allocate the required memory.

  As of this release, this allocation is checked to ensure it is allocated correctly. (Bug #37994397)

- Users could encounter an assertion failure due to a data size and bounds mismatch. Errors were returned similar to the following:

  ```
  Assertion failure: ddl0file-reader.cc:193:m_ptr + data_size <
  m_bounds.second
  ```

  (Bug #37882398)

  References: This issue is a regression of: Bug #37233273.

- Fixed an issue relating to modifying the internal FTS configuration. (Bug #37792010)

- Fixed an issue related to virtual indexes. (Bug #37602657)

- If the row size exceeded the maximum allowed size due to `innodb_strict_mode=OFF`, confusing warning messages similar to "Cannot add field" could appear in certain situations, such as selecting from a table or dropping a column.

  Error messages are improved for this scenario. (Bug #37003342, Bug #36768046, Bug #36867372)

Optimizer Notes

- Fixed an issue relating to SQL queries involving correlated scalar subqueries and JSON tables. (Bug #36684370)

Packaging Notes

- For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated to version 3.0.17. For more information, see OpenSSL 3.0 Series Release Notes. (Bug #38457056)

- MySQL Server RPM installation on Fedora 42 could fail due to conflicts with MariaDB 11.8 packages. Errors were returned similar to the following:

```
file /usr/bin/mysqld conflicts between mysql-community-server-8.4.7
and mariadb11.8-server-3:11.8.3, file
/usr/share/man/man1/mysql.1.gz conflicts between mysql-community-
client-8.4.7 and mariadb11.8, file
/usr/lib/systemd/system/mysqld.service conflicts between mysql-
community-server-8.4.7 and mariadb11.8-server
```

(Bug #38384844)

- The RPM specification and DEB control files were missing several requirements, making it impossible to build those formats. (Bug #38310603)

Performance Schema Notes

- Connections waiting on the connection control plugin were not displayed in the `performance_schema.processlist` table. (Bug #38043683)

Bugs Fixed

- Logical clock handling for `CREATE ... SELECT` statements has been improved to ensure accurate logical clock values in the replica relay log. (Bug #38383106)
- The server sometimes processed table references incorrectly. (Bug #38001000)
- `PARTITION BY ... (DEFAULT (column))` was not always handled correctly. (Bug #111234, Bug #35451459)

  References: This issue is a regression of: Bug #33142135.

Changes in MySQL 8.4.6 (2025-07-22)

Note

These release notes were created with the assistance of MySQL HeatWave GenAI.

- Compilation Notes
- Configuration Notes
- InnoDB Notes
- Installation Notes
- Packaging Notes
- Functionality Added or Changed
- Bugs Fixed

Compilation Notes

- **macOS:** It is now possible to compile the server on MacOS using `-DWITH_KERBEROS`.
- **Solaris:** Clang and GCC now can be used for compiling MySQL on Solaris. (Bug #30562248)

- Upgraded the bundled `libcurl` library to version 8.14.1. (Bug #38042758)
- MySQL Server now supports CMake 4, ensuring compatibility with future CMake versions where support for versions prior to 3.10 is expected to be discontinued. (Bug #38027636)
- The included ICU library has been upgraded to version 77-1. (Bug #37870791)
- The included `zstd` library has been upgraded to version 1.5.7. (Bug #37869972)
- Disabled `clang::musttail` when building with GCC 15. (Bug #37776018)
- Worked around an issue with list handling in certain older versions of CMake. (Bug #37709169)
- Some compiler features tests did not pass when building with `-fprofile-use`. (Bug #37707556)
- The linker tried to use the empty Cmake variable `${ICU_LIBRARIES}`, even though the correct library (`ext::icu`) was already linked elsewhere. (Bug #36524167)
- On MacOS, silenced deprecation warnings generated by Xcode 14; this includes suggestions to use `snprintf(3)` instead of `sprintf(3)`, and warnings about possible loss of precision when converting from 64-bit to 32-bit integers. (Bug #34776172)

Configuration Notes

- **Microsoft Windows:** An in-place upgrade of MySQL Server using MySQL Configurator failed when a Windows service name other than the default had been used. (Bug #37917039)
- **Microsoft Windows:** When upgrading a server from 8.0 to a higher series, MySQL Configurator did not persist custom server settings in the `my.ini` file. (Bug #37481548)
- **Microsoft Windows:** When upgrading a MySQL Server using MySQL Configurator, the process hung in the "Starting the server and upgrading system tables" step if a custom error log name was used. (Bug #37463478)

InnoDB Notes

- Fixed an issue relating to importing tables. (Bug #37621360)

Installation Notes

- Debian packages for installing MySQL can now be run by users other than root. This helps prevent issues for Debian or Ubuntu systems that require rootless installations. (Bug #37765153)
- RPM and Yum repository installation are now supported for Red Hat Enterprise Linux and Oracle Linux 10. (Bug #37592019)

Packaging Notes

- On Fedora 24, Oracle MySQL RPMs could not be installed due to package conflicts when MariaDB was already installed on the system. With this fix, the conflicts are not properly handled to allow MySQL Server to be installed successfully. (Bug #37798784)

Functionality Added or Changed

- **Important Change:** Added the `mysql` client `--commands` option, which enables or disables most `mysql` client commands.

This option is enabled by default. To disable it, start the client with `--commands=OFF` or `--skip-commands`.

For a complete list of all commands affected by this option, and additional information, see mysql Client Options. (WL #16949)

References: See also: Bug #36416568, Bug #38066040.

- **Group Replication:** Added the error `ER_GRP_RPL_APPLIER_THD_KILLED`, to distinguish when the applier thread has been terminated using SQL KILL, rather than stopped by an error. (Bug #37764717)
- Binary packages that include `curl` rather than linking to the system `curl` library have been upgraded to use `curl` 8.14.1. (Bug #37389565)

Bugs Fixed

- **Important Note; Group Replication:** The Group Communication System (GCS) handles Group Replication communication between members, and keeps track of the group membership and connections between all group members. Membership tracking includes the membership's current and previous two iterations. When a member leaves, the remaining members in the group keep a record of the departing member but stop communicating with it until it returns. For example: The group contains members M1, M2, and M3. M3 leaves the group; M1 and M2 stop communicating with M3. When a new member (M4) joins, it learns the previous iterations of this group's membership and attempts to communicate with all servers, including those from previous iterations (in this case, M3).

  In the event that some previous servers were gone and did not return, the new member continuously tried to connect to the missing servers; in some conditions, these ongoing connection attempts could introduce network latency affecting group member communication.

  In order to avoid this issue, connections to servers that belong to iterations of the group membership are now stopped after 5 minutes, which should be sufficient time to re-establish valid connections without imposing a continuing impact on group communication. (Bug #37704514)

- **Performance:** Redundant conditions in some queries optimized away in MySQL 8.0 were no longer removed in later versions, leading to a significant drop in the performance of such queries. (Bug #117907, Bug #37808260)

  References: This issue is a regression of: Bug #30112096.

- **NDB Cluster:** Following an upgrade from NDB 8.0 to NDB 8.4, all data nodes in the cluster underwent an unexpected simultaneous restart. This occurred when the transaction coordinator had no scan state, leading to protocol timeout; the resulting misalignment in protocol states caused data nodes to shut down unexpectedly. This is fixed by extending existing handling of an unexpected `SCAN_NEXTREQ` signal to cover the case when the scan is already stateless. (Bug #37994985)

  References: This issue is a regression of: Bug #37022901.

- **NDB Cluster:** Concurrent `ALTER TABLE` statements could cause delays of up to 100 * `TransactionDeadlockDetectionTimeout` before failing with a lock wait timeout when a client held a conflicting row lock, stalling the accompanying `get_commit_count()` call. The retry mechanism has been adjusted to identify locking issues sooner. (Bug #37955025)

- **InnoDB:** When rebuilding a primary key, the server sometimes encountered issues when duplicates were present, potentially leading to the server stopping unexpectedly.

  Our thanks to Xizhe Zhang and the team at Alibaba for the contribution. (Bug #37822992)

- **InnoDB:** Fixed an issue relating to dropping columns that were part of an index. (Bug #37726881)
- **InnoDB:** MySQL Community Edition binaries included unnecessary OpenTelemetry symbols, due to unprotected static arrays. (Bug #37689163)
- **InnoDB:** The double write buffer was unnecessarily large. When calculating the number of segments per file, one extra segment was always added, whereas an extra segment should be added only if the number of `Double_write` instances is not divisible by the number of `dblwr` files. (Bug #37684656)
- **InnoDB:** Fixed an issue relating to DELETE operations. (Bug #37478594)
- **InnoDB:** Creating a secondary index on a `VARCHAR` column could allocate more memory than configured, with the amount allocated being directly related to the value of `innodb_ddl_buffer_size`, leading to errors similar to `ERROR 1136 (21S01): Column count doesn't match value count at row 1`. (Bug #37233273)
- **InnoDB:** Fixed an issue relating to indexing spatial datatype columns. (Bug #36682518)
- **InnoDB:** The `temptable` handler did not terminate cleanly during server shutdown, potentially causing the server to terminate unexpectedly. This issue has been addressed by implementing a mechanism to track and properly clean up temptable objects associated with each thread. (Bug #36538081)
- **InnoDB:** A long semaphore wait crash could occur when a redo log consumer lagged behind after a failed MySQL Enterprise Backup incremental backup, preventing the redo log writer thread from advancing. Error messages were returned similar to the following:

  ```
  [Warning] [MY-013934] [InnoDB] Redo log writer is waiting for MEB redo
  log consumer which is currently reading LSN=23335640211468 preventing
  reclamation of subsequent portion of the redo log. Consider
  increasing innodb_redo_log_capacity.
  ```

  (Bug #36330455)
- **InnoDB:** Fixed an issue relating to range queries on tables. (Bug #31360522)

  References: See also: Bug #38063122.
- **Partitioning:** Truncating a partition was rejected with a duplicate entry error when the partition ID exceeded `INT_MAX`, preventing creation of new partitioned tables. To mitigate this issue, the `Table_partition_values_pk` class constructor now uses `ulonglong` instead of `int` for the object ID. (Bug #35912852)
- **Replication:** When using replication in a chain, `CREATE TABLE ... AS SELECT` sometimes resulted in inconsistent entries in the binary log, potentially causing replication to break on downstream servers. Error messages related to this issue included errors resulting from missing `START TRANSACTION` statements in the log. (Bug #37986380)
- **Replication:** During semisync replication, when the length of the binary log suffix exceeded six digits (`.999999`), so that the next log file became—for example—`mysql-bin.1000000`, the replication protocol unexpectedly changed from semisynchronous to asynchronous.

  Our thanks to Wuhao Cao and Karry Zhang and the team at Alibaba for the contribution. (Bug #115861, Bug #113813, Bug #37024069, Bug #36246779)
- **Group Replication:** The `Gr_empty_consensus_proposals_count` system status variable was not updated as expected. (Bug #37937927)
- **Group Replication:** In an unstable network environment, a Group Replication InnoDB Cluster with `group_replication_paxos_single_leader=ON` experienced several long-running

transactions that became stuck in the `waiting for handler commit` state. As a consequence, `group_replication_set_as_primary()` was forced to wait, which in turn blocked other incoming queries and rendered the cluster unwritable.

The issue involving long-running transactions stuck in `waiting for handler commit` occurred as follows: During an intermittent network partition, a secondary node incorrectly assumed the leader role due to outdated or inaccurate membership information. This resulted in conflicts in synode number allocation, causing transactions originating from the primary node to remain incomplete.

We fix this by making sure that a secondary node always reflect the latest, accurate state before pushing the view message to Paxos. This ensures that outdated or inconsistent membership information does not lead to conflicts in leadership or synode number allocation. (Bug #37764970)

References: See also: Bug #117424, Bug #37237959, Bug #37645674.

- **NDB Cluster APIs:** Excluding a `VARCHAR` column from an event definition resulted in an `Invalid schema object version` error. (Bug #37766391)

  References: See also: Bug #31848270.

- RPM installations on Fedora 24 could not be completed because conflicting packages were pulled from upstream. This fix adds the proper obsoletes to block the conflicting packages. (Bug #37976913)

- Some triggers which called stored routines did not always execute correctly. (Bug #37915445)

- A view using a CTE which contained a subquery was not always handled correctly. (Bug #37832605)

- Installing Oracle packages for MySQL on an Ubuntu 25.04 system was not possible where MySQL was already installed using the system's APT repositories. (Bug #37804480)

- Improved a previous fix for an issue in which client connections were not always terminated correctly during shutdown. (Bug #37755594)

  References: This issue is a regression of: Bug #35854919.

- The Fedora 42 RPM installation packages have been adapted to accommodate the merged `/usr/bin` and `/usr/sbin` directories in Fedora 42 and later. (Bug #37737658)

- Performance of fulltext searches using `InnoDB`, particularly for phrase searches, has been improved. The efficiency of `doc_id` matching has been enhanced. (Bug #37682648)

- Queries against the `performance_schema.keyring_keys` table caused issues when the `keyring_okv` plugin (see Using the keyring_okv KMIP Plugin) was installed but not configured correctly. (Bug #37655299)

- The fix for Bug #30875669 was not actually included in the code for the `mysql` client when the bug was closed. The changes are now implemented. (Bug #37572191)

  References: See also: Bug #30875669.

- An init file having one or more single lines, each containing multiple SQL statements, sometimes gave rise to errors during initialization. (Bug #37559598)

- Some sequences of `CREATE TABLE` and `DROP TABLE` statements were not handled correctly. (Bug #37534068)

  References: This issue is a regression of: Bug #35721121.

- `UPDATE ... SET ...` could not always be rolled back successfully. (Bug #37489167)

- Multiple `DEFAULT` column expressions in `CREATE TABLE` statements were not always handled correctly. (Bug #37436220)
- Queries having a correlated subquery which performed aggregation were sometimes (incorrectly) rejected with duplicate key errors during execution.

  This issue was introduced by a previous fix which removed the restoration of the original reference slice (`set_ref_item_slice(REF_SLICE_SAVED_BASE)`) during `JOIN::cleanup()` execution under the assumption that this was not necessary. As a result, temporary table field references from previous executions were not cleared, leading to attempts to insert duplicate keys into temporary tables triggering the error `Can't write; duplicate key in table`.

  We fix this by restoring the original reference slice during cleanup, ensuring that any stale references are discarded. (Bug #37415167)

  References: See also: Bug #32141711. This issue is a regression of: Bug #35856247.
- Unquoted semicolon characters (`;`) within comments were not always flagged as errors, in spite of the fact that they are not allowed. (Bug #37117875)

  References: See also: Bug #38063286.
- A query using a nested aggregate function which contained a subquery was not always properly handled. (Bug #36421727)
- When attempting to transform a subquery to a derived table, certain cases were not always considered. (Bug #36421710)

  References: This issue is a regression of: Bug #36921175.
- With `optimizer_switch` set to `subquery_to_derived=on`, some queries with `ROLLUP` were not handled properly. (Bug #36314993)
- Setting `max_join_size` led to improper processing of some nested queries. (Bug #35625769)
- The transform of a scalar subquery into a join with a derived table where the subquery is in the `SELECT` list and the containing query is implicitly grouped should be allowed, but was rejected when the `subquery_to_derived` optimizer switch was enabled. (Bug #35150438)
- An error in `include/assert_grep.inc` could lead to erroneous results from any file that included it.

  Out thanks to Ke Yu for the contribution. (Bug #116239, Bug #37105430, Bug #37675340)
- The query rewrite plugin (see The Rewriter Query Rewrite Plugin) did not work properly when the server was run with `autocommit=OFF`. (Bug #115437, Bug #36784795)

Changes in MySQL 8.4.5 (2025-04-15)

- Account Management Notes
- Audit Log Notes
- Compilation Notes
- SQL Function and Operator Notes
- INFORMATION_SCHEMA Notes
- InnoDB Notes

- Functionality Added or Changed
- Bugs Fixed

Account Management Notes

- It was possible in some cases to grant a user the `EXECUTE` privilege, but subsequently to be unable to revoke it from the same user. (Bug #37570206)

Audit Log Notes

- `<COMMAND_CLASS>` was not populated for `<NAME>Execute</NAME>`.

  For more information, see Logging Specific Event Classes. (Bug #36686351)

Compilation Notes

- **Group Replication:** The OpenSSL Engine interface is deprecated, and is no longer being included in OpenSSL v3 main packages by some Linux distributions, including Fedora.

  To avoid build issues, the usage of the OpenSSL Engine interface by the Group Communication System (GCS) is now restricted to OpenSSL versions previous to 1.1. (Bug #37475769)

- **Linux:** Use /usr/bin/gcc (GCC 14.2.1) when building the server on Oracle Linux 10. (Bug #37616148)
- Upgraded the bundled Curl library to version 8.12.1. (Bug #37633587)
- Abseil could not be built on FreeBSD. (Bug #37611924)
- The bundled version of `opentelemetry-cpp` was upgraded to version 1.19.0. (Bug #37506554)
- In order to use `xxhash` functions independently from the `lz4` library (bundled or source), we compiled `xxhash.c` into our own binaries, which required using a great many CMake directives. Instead, we now build an interface library for `xxhash`, and link with that wherever such functions are used. (Bug #37417386)
- Use xxHash-0.8.2 from GitHub rather than the version bundled with lz4. (Bug #37387318)

SQL Function and Operator Notes

- **Important Change:** When an SQL function is improved from one release to the next, it may throw SQL errors in situations in which it previously did not. If this happens in a table's constraints, default expressions, partitioning expressions, or virtual columns, the table could not be opened. This prevented both analyzing the problem (using, for example, `SHOW CREATE TABLE`) and addressing it (such as with an `ALTER TABLE ... DROP ...` statement).

  Now, on server upgrade, we scan the data dictionary for tables that use any of the features just mentioned. We then try to open such tables, and if we fail to do so, we alert the user. This patch addresses this. The `--check-table-functions` server option introduced in this release helps to address this problem by making it possible to specify the server's behavior when encountering an error with such a function. Set this option to `WARN` in order to log a warning for each table which the server could not open; setting it to `ABORT` also logs these warnings as `WARN`, but aborts the server upgrade if any issues were found.

  `ABORT` is the default; this enables the user to fix the issue using the older version of the server before upgrading to the new one. `WARN` flags the issues, but allows the user to continue in interactive mode while addressing the problem. (Bug #36890891)

References: See also: Bug #37009318. This issue is a regression of: Bug #98950, Bug #98951, Bug #31031886, Bug #31031888.

INFORMATION_SCHEMA Notes

- Fixed a performance issue in the `PROCESSLIST` table. (Bug #36778475)

InnoDB Notes

- **InnoDB:** InnoDB now uses simplified APIs for saving and restoring level one progress during parallel scans. (Bug #37517125)

Functionality Added or Changed

- **Important Change:** For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated to version 3.0.16. For more information, see OpenSSL 3.0 Series Release Notes and OpenSSL Security Advisory (11th February 2025). (Bug #36033684)
- **InnoDB:** To improve debugging, the `buf_page_t` and `buf_block_t` structure's metadata is now printed to the error log. (Bug #35115629)

  References: See also: Bug #35115601.
- Increased the historical 1024-byte limit when printing the current query during signal handling to 1073741824 (1024 * 1024 * 1024). (Bug #37603354)

Bugs Fixed

- **InnoDB:** Fixed a potential memory leak in several places in the innobase code. (Bug #37403052)
- **InnoDB:** Under certain circumstances, MySQL could crash during shutdown due to pages which were still fixed or dirty. Errors similar to the following were logged:

  ```
  [ERROR] [MY-011908] [InnoDB] [FATAL] Page [page id: space=46, page number=75] still fixed or dirty
  [ERROR] [MY-013183] [InnoDB] Assertion failure: buf0buf.cc:5889:ib::fatal triggered thread 139963705668
  ```

  (Bug #37391519)

  References: See also: Bug #35115601.
- **InnoDB:** `CHECK TABLE` for spatial indexes did not verify the MBR against the geometry MBR stored in the clustered index record. This could result in incorrect behavior of spatial indexes.

  As of this release, `CHECK TABLE EXTENDED` verifies the MBR matches the MBR stored in the clustered index record. (Bug #37359538)
- **InnoDB:** Fixed an issue relating to pessimistic row update.

  Our thanks to Mengchu Shi and the team at Alibaba for the contribution. (Bug #37292404)
- **InnoDB:** The `CHECK TABLE` operation could incorrectly report corruption in spatial indexes. (Bug #37286473)
- **InnoDB:** Fixed an issue relating to InnoDB redo log recovery. (Bug #37061960)
- **InnoDB:** Fixed an issue relating to reading `index_id` values. (Bug #36993445, Bug #37709706)

- **InnoDB:** Fixed an issue relating to `lower_case_table_names`. (Bug #32288105)
- **InnoDB:** Partition table indexes were not checked when retrieving a record count while that table's definition was being altered by another client session. The record count was executed without error.

  As of this release, the index is checked to ensure it is usable when retrieving a record count. (Bug #117459, Bug #37617773)
- **InnoDB:** Refactored code related to BPR_PCUR_* positioning for restore operations. (Bug #117259, Bug #37505746)

  References: This issue is a regression of: Bug #37318367.
- **InnoDB:** Changes made to `innodb_spin_wait_delay` in MySQL 8.0.30 negatively impacted performance. (Bug #116463, Bug #37212019)
- **InnoDB:** Under certain circumstances, using `ALTER TABLE` with `INPLACE` to modify the size of a column could result in an index which exceeds the valid size limit (767 bytes). This occurred for tables with a row format of `Redundant` or `Compact` and the row format was not explicitly defined in the table creation.

  As of this release, a validation is performed and an error returned by any ALTER TABLE, INPLACE operation which will result in an invalid index size. (Bug #116353, Bug #37168132)
- **InnoDB:** Fixed a memory leak in the `Clone_persist_gtid` thread.

  Our thanks to Baolin Huang and the team at Alibaba for the contribution. (Bug #107991, Bug #34454572)
- **Partitioning:** When inserting `NOW()` into a column not part of the partition key of a partitioned table, all partitions were retrieved, and no pruning occurred. (Bug #37397306)
- **Replication:** When the log sanitizer analyzes relay log files, it first searches for a starting point (such as a rotation event or transaction end), but in some cases, it was possible for a binary log file containing a needed GTID to be deleted as having no relevant data; this resulted in the point-in-time recovery process hanging indefinitely while waiting for the missing GTID to be applied. Now in such cases, the analysis skips parsing of transaction boundaries until the start point is established. (Bug #37635908)
- **Replication:** In a source-replica setup, the replica encountered irregular failures of `UPDATE` and `DELETE` statements with `ER_KEY_NOT_FOUND` errors on the same table. (The replica's binary log and GTID records showed that the row required was committed, and had not been deleted or updated.) This occurred on the replica when the row-matching algorithm used was `HASH_SCAN` and two rows in the same table had the same CRC32 value.

  In the event of such a CRC32 collision, finding a matching CRC32 in the hash table does not guarantee that the correct row is being updated, so the algorithm iterates over any multiple entries having the same CRC32, and compares the full record for each of them in a loop; the problem occurred due to the fact that the logic to exit this loop was incorrect. This logic has now been corrected. (Bug #37462058, Bug #37731216)
- **Replication:** The `asynchronous_connection_failover_delete_source()` function did not always perform as expected in all cases. (Bug #36479088)
- **Replication:** In some cases, the `asynchronous_connection_failover_add_source()` function did not perform as expected. (Bug #36479083)
- **Replication:** In some cases, `MASTER_POS_WAIT()` did not perform as expected. (Bug #36421684, Bug #37709187)

- **Replication:** The `asynchronous_connection_failover_add_managed()` function in some cases did not produce the expected result. (Bug #34648589)

- **Replication:** When the server was under a heavy write load, the binary log position for `gtid_executed` as shown in the Performance Schema `log_status` table did not match that of the gtid shown in the binary log file.

  We fix this by increasing the scope of the lock on the `log_status` table when querying it to ensure that transactions in the commit pipeline are completed. This ensures that a query against the `log_status` table waits until `gtid_executed` has been fully updated, thereby guaranteeing consistency with its position in the binary log. (Bug #102175, Bug #32442772)

- **Group Replication:** When a secondary joined the group, it might happen that all group members started to grow the value of the column `COUNT_TRANSACTIONS_ROWS_VALIDATING` column of the Performance Schema `replication_group_member_stats` table indefinitely. This impacted memory consumption in all group members, eventually leading to thrashing if not mitigated by restarting the secondary group member that triggered the behavior, or in some cases, by restarting the whole group.

  Analysis pointed to issues with the Group Replication start operation, which checks whether there are partial transactions on the `group_replication_applier` channel from previous group participation; if any are found, this channel is stopped after applying all complete transactions and its relay logs purged, and then the channel is restarted. After this, distributed recovery is performed, applying any missing data from group members.

  The issues arose when the Group Replication pipeline operation for stopping the `group_replication_applier` channel incorrectly stopped a periodic task from the certifier module, which caused some periodic internal operations not to take place. One of these tasks was the periodic sending of the committed transactions; this omission prevented garbage collection for certification, which in turn caused a continuous increase in `COUNT_TRANSACTIONS_ROWS_VALIDATING` in the Performance Schema `replication_group_member_stats` table.

  To solve this problem, we have taken steps to ensure that the pipeline operation for stopping the `group_replication_applier` channel no longer interferes with the certifier module, which also stops spurious values from being added for `COUNT_TRANSACTIONS_ROWS_VALIDATING`. (Bug #37613510)

- **Group Replication:** When running Group Replication, some transactions may not have write sets, as with empty transactions with GTID_NEXT specified, or DDL statements. For such transactions, Group Replication cannot check conflicts; thus, it is not known whether they can be applied in parallel, and for this reason, Group Replication follows a pessimistic approach, and runs them sequentially, potentially leading to an impact on performance.

  While DDL must be applied sequentially, there is no actual reason to force such behavior for empty transactions, so this fix makes it possible for empty transactions to be applied concurrently with other nondependent transactions. (Bug #37597512, Bug #37569333)

- The `fprintf_string()` function in `mysqldump` did not use the actual quote character for string escaping. (Bug #37607195)

- `EXPLAIN` did not always handle subqueries correctly. (Bug #37560280)

- If a demangled function name exceeded 512 bytes in a stack trace, the function name was truncated and a newline was not printed.

  As of this release, long strings, such as filenames and demangled functions, are written directly to the output. (Bug #37543598)

- `mysqldump` did not escape certain special characters properly in its output. With this fix, `mysqldump` now follows the rules as described in String Literals. (Bug #37540722, Bug #37709163)
- Some operations on tables having functional indexes were not handled properly. (Bug #37523857)
- Attempting to install an unknown component using `INSTALL COMPONENT` was not always handled correctly. (Bug #37437317)
- The Audit Log plugin did not handle errors correctly when writing JSON output.

  See MySQL Enterprise Audit, for more information. (Bug #37370439)
- An `UPDATE` subsequent to an `INSERT` affecting a table which had a `BEFORE INSERT` trigger was sometimes rejected with a null value error when the `INSERT` had set a `NOT NULL` column to `NULL`, even though this should have been allowed by the server `sql_mode` in effect. (Bug #37337527)
- In some cases, components could not reuse the same connection for running multiple queries. (Bug #37286895)
- Improved error handling for stored routines. (Bug #37193011)
- Stored routines were not always invoked correctly in prepared statements. (Bug #37077424, Bug #37292797)
- Increased the size of `SEL_ROOT::elements` from `uint16` to `size_t`. (Bug #36610878)
- Removed an issue with multibyte UTF8 handling. (Bug #36593253)
- An `ORDER BY` containing an aggregation was not always handled correctly. (Bug #36593244)
- An optimizer hint was ignored, unexpectedly requiring the use of `FORCE INDEX`, when querying a view that included a `UNION`. For more information, see Optimizer Hints. (Bug #36536936)
- Some subselects were not handled correctly. (Bug #36421690)
- An invalid DDL statement in certain cases was not always rejected as expected. (Bug #35721121)
- Improved the internal function `append_identifier()`. (Bug #35633084)
- Normally, a view with an unused window definition should be updatable, but when it contained a subquery, it was marked as not updatable. At update time, the window was eliminated, but this was too late to allow an update to be performed.

  We fix this by testing mergeability, by checking the presence of window functions, rather than that of window definitions; this allows the view to be updatable, and the problematic `UPDATE` to succeed. (Bug #35507777)
- In some cases, `SET` did not perform correctly in prepared statements. (Bug #35308309)
- This fix addresses the following issues:
  - `Query_expression::is_set_operation()` was not always executed properly.
  - Some sequences of DML statements could lead to an unplanned exit.
  - Some nested subselects were not always handled correctly.

  (Bug #34361287, Bug #35889583, Bug #35996409, Bug #36404149, Bug #36863048, Bug #37611264)

- On Debian, `dh_strip_nondeterminism` is no longer executed on zip and gzip files within the packages. (Bug #33791880)
- Removed an issue relating to invalid UTF8 values. (Bug #27618273, Bug #37709687)
- Addressed an issue relating to an invalid identifier. (Bug #22958632, Bug #37709664)
- A negative impact in performance was observed when using a multivalued index with `ORDER BY DESC` and `LIMIT` in a query, where the value specified by `LIMIT` was greater than the number of rows actually in the result. (Bug #117085, Bug #37436310)

  References: This issue is a regression of: Bug #104897, Bug #33334911.
- If one client session had a uncommitted transaction that caused a `DROP TABLE` statement in another client session to be blocked, a third client session hung when trying to issue a `USE DATABASE` statement. (Bug #115706, Bug #36892499)

Changes in MySQL 8.4.4 (2025-01-21)

- Account Management Notes
- C API Notes
- Character Set Support
- Compilation Notes
- Component Notes
- Firewall Notes
- Installation Notes
- Optimizer Notes
- Performance Schema Notes
- Functionality Added or Changed
- Bugs Fixed

Account Management Notes

- The database cache was not flushed properly following execution of `DROP USER`. (Bug #37132323)
- Failed password validation was not always handled correctly. (Bug #37041439)

C API Notes

- Process memory usage grew when the `libmysqlclient` API user tried to cache and reuse a prepared statement, preparing it once and then calling either of `mysql_stmt_bind_param()` or `mysql_stmt_bind_named_param()` followed by `mysql_stmt_execute()`, repeatedly without calling the matching `mysql_stmt_close()`, or calling it in the distant future (on application exit, for example).

  We solve this by introducing a separate `MEM_ROOT` object for storing the bind parameters array, which object can be cleared (deallocating the memory) on each call to `mysql_stmt_bind_param()` or `mysql_stmt_bind_named_param()`. (Bug #37202066)

Character Set Support

- Zero rows resulted (where one row was expected) when selecting from a view created with its connection and client character set to `latin1`, where a query on the view used `utf8` as its connection and client character set, the view contained literal values with non-ASCII characters, and the query performed a condition pushdown into a `UNION` of query blocks of the view.

  This problem was related to a previous issue which fixed an error for a similar problem: The problem in that case was properly considering the character set of the view definition when pushing down conditions contained in the view to inner query blocks, but the fix implemented at that time did not take into account the possibility that the view might contain non-ASCII characters.

  This meant that the condition to be pushed down was written to a text string with the wrong character set. We fix this oversight by ensuring that the string is created with the correct character set. (Bug #37111452)

  References: See also: Bug #36246859.

Compilation Notes

- **macOS:** Removed obsolete CMake code from MacOS builds. (Bug #37258036)
- **macOS:** It is now possible to build MySQL using the Homebrew version of Clang. (Bug #37256912)
- **macOS:** Removed warnings of the form `ld: warning: ignoring duplicate libraries` and warnings specific to `xcodebuild`. (Bug #37065301)
- **Microsoft Windows:** Disabled windows compiler warnings C26445 and C26821 in `cmake/msvc_cppcheck.cmake`. Both of these relate to MSVC substitution of `gsl::span` for `std::span`, which is not used for MySQL. (Bug #37158156)
- **Solaris:** The minimum required version of GCC to build MySQL on Solaris has been raised to 11.4. See Source Installation Prerequisites, for more information. (Bug #37256600)
- Starting with CMake 3.26, CMake writes the file `CMakeFiles/CMakeConfigureLog.yaml`, which supersedes `CMakeError.log`. References to `CMakeError.log` have therefore been removed. (Bug #37305289)
- Implemented the standards-compliant `my_char_traits<unsigned char>` for use as a drop-in replacement for `std::char_traits<unsigned char>`, which was deprecated in Clang 18 and removed in Clang 19. (Bug #37273525, Bug #37785339)
- Removed a **`maybe-uninitialized`** error found in `sql/item.cc` when building MySQL with GCC 14. (Bug #37157201)
- The version of `libedit` used to compile MySQL was upgraded to 20240808-3.1. (Bug #37101293)
- The server could not be built on Ubuntu 22.04 using Clang 13. (Bug #37075154)
- Removed an error in `mysql_prepare_create_table()` (in the file `sql/sql_table.cc`) found when compiling MySQL with XCode 16. (Bug #37068527)
- MySQL could not be compiled using Clang 19. (Bug #37014761)
- `#include <chrono>` was missing from `plugin/group_replication/libmysqlgcs/src/bindings/xcom/xcom/task.cc` even though `std::chrono::duration_cast()` was referenced in this file. (Bug #116779, Bug #37329617)

- The server could not be built on Fedora 40 (and possibly other Linux platforms) using cmake 3.11, due to an issue with TIRPC. (Bug #116164, Bug #37080195)

Component Notes

- `INSTALL COMPONENT` issued concurrently with a `SET PERSIST` which used a subquery could sometimes lead to an unplanned exit of the server. (Bug #36559078)

  References: See also: Bug #35647759.

Firewall Notes

- In some cases, after performing an upgrade, stored procedures relating to MySQL Enterprise Firewall were not processed correctly. (Bug #36084822)

Installation Notes

- When upgrading from MySQL 5.7 to a later MySQL release series, the system-created `mysql.sys` and `mysql.session` accounts are now modified to use the `caching_sha2_password` authentication plugin instead of the `mysql_native_password` plugin, which is deprecated in MySQL 8.0, and removed in MySQL 9.0. (Bug #36608636)

Optimizer Notes

- Pushing down a condition which had an aggregate function in a `WHERE` clause caused the aggregate function to be evaluated when it should not have been. (Bug #36421735)

Performance Schema Notes

- If a user other than root ran `START REPLICA`, `PERFORMANCE_SCHEMA.PROCESSLIST` assigned that user's name to the newly spawned foreground replication threads instead of `system user`.

  As of this release, `system user` is assigned to all foreground system threads. (Bug #37357562)

- Under certain circumstances, a metadata lock can be upgraded or downgraded to a different `LOCK_TYPE`. This change was not reflected in the `PERFORMANCE_SCHEMA.METADATA_LOCKS` table.

  Our thanks to George Ma and the Alibaba team for the contribution. (Bug #116625, Bug #37271768)

Functionality Added or Changed

- Binary packages that include `curl` rather than linking to the system `curl` library have been upgraded to use `curl` 8.11.1. (Bug #37389565)

Bugs Fixed

- **Incompatible Change:** Corruption occurred in a spatial index when an update of a geometry with a minimal change in the minimum bounding rectangle (MBR) was followed by a delete operation.

  When upgrading to this release, it is recommended that you drop any spatial indexes beforehand, then re-create them after the upgrade is complete. Alternatively, you can drop and re-create such indexes immediately following the upgrade, but before making use of any of the tables in which they occur. You should also be aware that downgrading to any previous version reintroduces the original problem described previously.

  For more information, see Creating Spatial Indexes. (Bug #36452528)

- **InnoDB:** Concurrently truncating a table while querying the Performance Schema sometimes cause MySQL to halt unexpectedly. (Bug #37271715)
- **InnoDB:** It was possible for an `ALTER TABLE` operation using the `INPLACE` algorithm on a table containing both a spatial index and an auto-increment column to cause corruption or, in debug builds, to trigger a debug assert. This was due to the auto-increment column value being overwritten in the old records of the spatial index while the new record was prepared. (Bug #37189985)
- **InnoDB:** Certain IO buffer serializations triggered an assertion in debug builds that caused the system to hang. (Bug #37139618)
- **InnoDB:** Improved `InnoDB` start up time. (Bug #36880863)

  References: This issue is a regression of: Bug #36808732.
- **InnoDB:** An assertion failure was raised when creating a `FULLTEXT` index on a table with an `FTS_DOC_ID` value greater than 4294967295. (Bug #36879147)

  References: See also: Bug #37387224.
- **InnoDB:** Dropping a primary key, and then adding a new `AUTO_INCREMENT` column as a primary key in descending order using the `INPLACE` algorithm failed.

  Our thanks to Shaohua Wang and the team at Alibaba for the contribution. (Bug #36658450)
- **InnoDB:** Extending a user tablespace produces file extension redo log records (`MLOG_FILE_EXTEND`), but they were not produced when extending the system tablespace. (Bug #36511673)
- **InnoDB:** A `DELETE` operation on a table with a self referential foreign key and full-text index could have triggered an assertion. (Bug #36234681)
- **InnoDB:** When clearing an AHI index from all buffer pool pages, the block state would potentially change to BUF_BLOCK_MEMORY before acquiring the block mutex thus causing an unexpected halt. (Bug #35037114)
- **InnoDB:** Common prefix compression for redo log inserts (`MLOG_REC_INSERT`) was disabled but is now enabled when the versions match. (Bug #34946626)

  References: This issue is a regression of: Bug #13899.
- **InnoDB:** Virtual column information for a row containing an externally stored `BLOB` was not always logged during an `UPDATE` operation, which sometimes resulted in an `Index PRIMARY is corrupted` error. (Bug #34574604)
- **InnoDB:** `ON DELETE CASCADE` with generated columns containing secondary indexes sometimes failed, due to virtual column templates not being initialized before deletion.

  Our thanks to Rahul Malik for the contribution. (Bug #33691659)
- **InnoDB:** An update operation attempted to update a virtual column while building an update node for a child table, but should not have done so since foreign key constraints cannot reference virtual columns. (Bug #33327093)
- **InnoDB:** It was possible for `ALTER TABLE`, which rebuilds `InnoDB` tables using the `INPLACE` algorithm, to be rejected with a duplicate key error due to a non-duplicate record being inserted concurrently while the rebuild was paused to release a page latch.

  Our thanks to Dmitry Lenev and the team at Percona for contributing to this fix. (Bug #115511, Bug #36808088)

- **InnoDB:** The check enforcing the rule that `ALGORITHM=INSTANT` cannot be used on a column referenced by a foreign key constraint from another table did not inspect the last field of said constraint. (Bug #115457, Bug #36796094)

- **InnoDB:** CPU usage statistics did not account for a processor count over 128, which could degrade performance on these larger systems. (Bug #115399, Bug #36765223)

- **InnoDB:** Executing `ALTER TABLE` with `ADD COLUMN` or `DROP COLUMN` against an empty table now uses the `INPLACE` algorithm by default instead of `INSTANT`. This change means the row version is no longer incremented for these simple operations. (Bug #113051, Bug #36004394)

- **InnoDB:** An `ALTER TABLE` operation that rebuilt an `InnoDB` table using the `INPLACE` algorithm potentially led to losing a row of data if a purge occurred concurrently on the altered table that contained a clustered or spatial index.

  Our thanks to Dmitry Lenev and the team at Percona for contributing to this fix. (Bug #110706, Bug #113812, Bug #115608, Bug #116764, Bug #35303494, Bug #36261480, Bug #36846567, Bug #37318367)

- **InnoDB:** Queries with a descending primary key and the `index_merge` optimization sometimes yielded incorrect results such as missing rows. (Bug #106207, Bug #33767814)

- **Replication:** In an InnoDB ClusterSet setup, when `autocommit` was set to `OFF` on all nodes in the cluster, a controlled switchover using MySQL Shell was rejected with Error 1105 (`Unknown error`).

  To fix this, we now force a new transaction in `channel_change_source_connection_auto_failover()` whenever `autocommit=OFF` to prevent table access deadlocks when an info repository transaction is executed after changing `SOURCE_CONNECTION_AUTO_FAILOVER`. (Bug #37173907)

- **Replication:** While large transactions were being received and applied, and a request to stop the replication channel was made using `STOP REPLICA`, MySQL did not do so properly, and subsequently did not process any channel commands. In addition, the server shutdown process did not complete gracefully, and required either the MySQL process to be killed or the host system to be restarted. (Bug #115966, Bug #37008345)

- **Replication:** The log message written when a replica reconnects to the source (when, for example, it is stopped and restarted by issuing `STOP REPLICA` followed by `START REPLICA`) `While initializing dump thread for replica with UUID` *`uuid`*`, found a zombie dump thread with the same UUID. Source is killing the zombie dump thread(`*`thread_id`*`)` has been improved to `Upon reconnection with the replica, while initializing the dump thread for UUID` *`uuid`*`, an existing dump thread with the same UUID was detected. The source is terminating the previous dump thread (`*`thread_id`*`), which is normal and expected`. (Bug #84358, Bug #25330090)

- **Group Replication:** Removed a potential race condition between the internal functions `cs::apply::Commit_order_queue::front()` and `cs::apply::Commit_order_queue::remove()`. (Bug #37223451)

  References: See also: Bug #35206392.

- **Group Replication:** When the primary node unexpectedly left the group and quickly attempted to rejoin, the member which had been elected to remove other, faulty members tried to expel or remove the faulty node but could not do so due to lack of a majority. When the expelled or removed node was the primary, this left the cluster without a primary, resulting in an unusable state. (Bug #36991859)

  References: See also: Bug #37181867.

- **Group Replication:** In some cases, adding a new secondary caused existing secondaries to lag, leading to a deadlock which persisted with no more writes possible until the primary was restarted.

  This deadlock occurred between the ticket manager, which ensures that transactions are committed on the correct side of a view change (before or after the view change), and the commit order manager on the inbound replication channel, which ensures that transactions are committed in the same order in which they are received, when these two managers required different orders. This meant that, in some cases, adding a new secondary caused the group primary to be unable to do writes.

  We solve this issue by ignoring the commit order manager ordering and enforcing the ticket manager ordering for non-conflicting transactions when such a deadlock occurs. A consequence of this is that `replica_preserve_commit_order` may not be strictly honored near a `View_change_log_event`. In other words, `replica_preserve_commit_order` no longer provides a strict guarantee on an inbound channel on a Group Replication primary. `replica_preserve_commit_order` still ensures that transactions are ordered correctly, with the only exception being non-conflicting transactions around view change log events. (Bug #35206392)

  References: See also: Bug #37223451.

- **Group Replication:** Improved garbage collection in Group Replication by eliminating non-essential calls to `is_subset_not_equals()`. (Bug #110673, Bug #35286974)

- **Group Replication:** Removing a group member from a group in which all members were running the same version of MySQL, upgrading it to a later version (from a later release series), and then requesting it to rejoin the group caused the upgraded group member to hang in the recovering state.

- The errors `ER_DD_UPDATE_DATADIR_FLAG_FAIL`, `ER_IB_MSG_FIL_STATE_MOVED_PREV_OR_HAS_DATADIR`, `ER_RPL_KILL_OLD_DUMP_THREAD_ENCOUNTERED`, and `ER_RPL_MTA_ALLOW_COMMIT_OUT_OF_ORDER` were originally defined in MySQL 8.0, but were subsequently assigned different error code numbers (but with the same names) in MySQL 8.4. The numbers assigned in MySQL 8.0 now apply to MySQL 8.0 only; in the MySQL 8.4 and later release series, only the numbers assigned in MySQL 8.4 are used. (Bug #37284176)

- Added missing checks for `NULL` input arguments in `mle::validate()` and `validate_session_options()`. (Bug #37267887)

- In `sql/item_cmpfunc.cc`, `Item_bool_func2::resolve_type()` made an unchecked call to `Item_bool_func::resolve_type()`; the call to `Item_bool_func::resolve_type()` ignored its return value, and execution continued even in case of an error. (Bug #37143289)

- AppArmor denied access to `/proc/$pid/task/$thread_id/mem`, a file required to generate a stack trace. (Bug #37063288)

  References: See also: Bug #37387034.

- A filter condition in a subquery was sometimes ignored when a query used the `index_subquery` join type for subquery execution, and the subquery table used materialization in the execution plan. The derived table access path replaced the filter condition, resulting a final plan without the filter layer. To fix this, in such cases, we now add the derived table access path along with the filter access path instead of replacing the latter. (Bug #36918913)

- Some `UNION` operations similar to `a UNION b UNION c ...` consumed excessive memory. To help keep this from happening, we now flatten equal set operations at the parsing level, before contextualization occurs, which should result in reduced resource usage by such operations. (Bug #36652610)

- Improved the internal function `my_print_help()`. (Bug #36615714)

  References: See also: Bug #37387224.

- Removed incorrect code from `Acl_cache`. (Bug #36608160)

- A subquery containing an aggregate function `WITH ROLLUP` which was part of a row value comparator was not always processed correctly. (Bug #36593235)

  References: See also: Bug #37387180. This issue is a regression of: Bug #30969045, Bug #30921780, Bug #26227613, Bug #29134467, Bug #30967158.

- It was possible for errors raised when persisting variables not to be reported correctly. (Bug #36574732)

- Some subqueries using `WITH ROLLUP` were not always processed correctly. (Bug #36421704)

- `MyISAM` block length calculations were not always performed correctly. (Bug #36347992)

- Fixed an issue relating to FTS and concurrent DDL or DML. (Bug #34633727)

- `DROP VIEW` *`name`* was rejected with `ER_BAD_TABLE_ERROR` if there existed a table with the same *`name`*. (Bug #33200087)

- Incorrect results were returned by some queries that used hash antijoins when the hash table did not fit in the join buffer and spilled to disk. (The query triggering the issue actually specified `LEFT JOIN`, but this was transformed internally from a left outer join to an antijoin.)

  The problem was that some rows in the probe table were skipped when writing the probe rows to chunk files, the skipped rows being those that had `NULL` in part of the join key. Such rows can be skipped for inner joins and semijoins, as they are known to have no match in the build table, but for outer joins and antijoins, rows in the probe table which have no matching row in the build table should be part of the join result, so they must be included in the chunk files.

  We already preserved these rows in the chunk files for outer joins. This fix extends the logic used for that purpose so that it also applies for antijoins. (Bug #116334, Bug #37161583)

- In MySQL 8.0 and later, queries of the form `SELECT DISTINCT ... FROM t1 WHERE NOT IN(SELECT ...)` were transformed into an antijoin if possible, causing the optimizer not to choose a group skip scan for table t1 whereas it would have been chosen in MySQL 5.7. This resulted in a performance degradation for such queries. Group skip scan is not chosen, since the query is now no longer a single-table query following the antijoin transformation, and this access method is enabled only for single table queries. The same behavior can be seen for queries which are transformed into semijoins as well. In such cases, group skip scan access method can still be used if the access method is used only for duplicate removal (that is, with `DISTINCT` or `GROUP BY`, but without aggregate functions).

  To fix this, we enable group skip scan when there is only one table in the original query, irrespective of the number of semijoin tables present after internal transformations as long as the query contains no aggregate functions. (Bug #112362, Bug #35842412)

- The `mysql` client did not allow using '#' or '--' inside an optimizer hint comment.

  Our thanks to Kaiwang Chen for the contribution. (Bug #98521, Bug #30875669)

Changes in MySQL 8.4.3 (2024-10-15)

- C API Notes

- Compilation Notes
- Configuration Notes
- SQL Function and Operator Notes
- JavaScript Programs
- Performance Schema Notes
- sys Schema Notes
- Thread Pool Notes
- Functionality Added or Changed
- Bugs Fixed

C API Notes

- The asynchronous interface used unsafe static local variables to store state information. (Bug #115703, Bug #36891894)

Compilation Notes

- MySQL did not compile on Fedora 41. (Bug #37046924)
- MySQL did not compile on Ubuntu 24.10. (Bug #37042308)
- Added `CONTRIBUTING.md` and `SECURITY.md` files to the MySQL sources to conform to Oracle's Open Source guidelines. (Bug #36998165)
- Aligned `CMAKE_MINIMUM_REQUIRED` with the correct required CMake version (3.14.6) as well as CMake policies for third-party libraries used by MySQL. (Bug #36978193)
- Binary packages that include curl rather than linking to the system curl library have been upgraded to use curl 8.9.1. Important issues fixed in curl version 8.9.1 are described at https://curl.se/docs/security.html. (Bug #36967379, Bug #36955197)
- The included `zlib` library has been upgraded from version 1.2.13 to version 1.3.1. (Bug #36950863)
- CMake options `BUILD_SHARED_LIBS` and `CMAKE_SKIP_INSTALL_ALL_DEPENDENCY` are now set to `OFF` in the top level of the build. (Bug #36930664)
- Pulling in the most recent CMake packages for gRPC and Protobuf caused the builds for these libraries to fail. Fixed by rewriting the package lookup to work for both older and newer CMake packages. (Bug #36905657)
- Upgraded the bundled `libcurl` library to version 8.9.0. (Bug #36886877)
- The bundled `lz4` library was upgraded to version 1.10.0. (Bug #36886747)
- The `-DWITHOUT_SERVER` option for CMake did not work on Enterprise Linux 7. (Bug #36824515)
- Linking with `mold` did not work on Enterprise Linux 9. (Bug #36818902)

  References: See also: Bug #34099162.

- The version of `libfido` used with MySQL was upgraded to 1.15.0. (Bug #36752604)
- The RE2 regular expression library is now bundled with the MySQL sources. (Bug #36729026)
- The version of `clang-format` used for the MySQL codebase was upgraded from 10 to 15. (Bug #36500268)
- The bundled version of TI-RPC was upgraded to 1.3.5. (Bug #115698, Bug #36886602)
- It was not possible to build MySQL using Protobuf 22 or later.

  Our thanks to Gordon Wang for the contribution. (Bug #115163, Bug #36678092)

Configuration Notes

- **Microsoft Windows:** On Windows, *MySQL Configurator* revert button functionality did not always revert to the original value. (Bug #36702176)
- **Microsoft Windows:** On Windows, the *MySQL Configurator* data directory file browsing mechanism required the `ProgramData` folder but can now function outside of that folder. (Bug #36702072)
- **Microsoft Windows:** On Windows, clicking the **[X]** close button on a *MySQL Configurator* wizard's page now yields a confirmation popup if the wizard is busy executing an operation. (Bug #36671317)
- **Microsoft Windows:** On Windows, *MySQL Configurator* no longer defines a custom `server_type` variable in the generated MySQL Server configuration file. This information is now stored in the `configurator_settings.xml` file. (Bug #36670309)
- **Microsoft Windows:** On Windows, the `Removing Windows Firewall` step in *MySQL Configurator* would fail if the `my.ini` file was missing a `mysqlx_port` definition. (Bug #36666260)
- **Microsoft Windows:** On Windows, *MySQL Configurator* no longer performs unnecessary background operations after clicking **Next** if the selected options did not change. (Bug #36395605)
- **Microsoft Windows:** On Windows, if *MySQL Configurator* failed to find a valid `my.ini` or `my.cnf` file from the *MySQL Server Installations* page, then clicking the **Browse** button disabled the **Next** button even when the selected file was valid. (Bug #36395569)
- **Microsoft Windows:** On Windows, *MySQL Configurator* now only shows the removal steps if the associated MySQL Server was previously configured. (Bug #36395417)
- **Microsoft Windows:** On Windows, MySQL Configurator described an empty password as strong rather than weak. (Bug #35533726)

SQL Function and Operator Notes

- The output from `DATABASE()` was truncated when this function was used as part of a `UNION` query. (Bug #36871927)

JavaScript Programs

- Employing a user variable in a DML statement within a stored procedure that used the Statement handle interface sometimes led to an unplanned server exit when any statement was executed following execution of such a stored procedure. To address this issue, we now reset the memory used to the store the user variable and its value to the statement execution memory root after executing a sub-statement from the Statement handle interface. (Bug #36892945)

Performance Schema Notes

- Fixed issues relating to the OTLP exporter. (Bug #36792180, Bug #36783070)
- Running `SELECT * FROM sys.innodb_lock_waits;` on an instance which was under heavy load affected the performance of the server.

  As of this release, `SELECT * FROM sys.innodb_lock_waits;` fetches only 2 locks for each wait, instead of scanning all locks twice for each wait.

  As part of this fix, primary keys were added to `DATA_LOCKS` and `DATA_LOCK_WAITS`. (Bug #100537, Bug #31763497)

sys Schema Notes

- The performance of the `innodb_lock_waits` view is improved in this release. (Bug #36337708)

Thread Pool Notes

- Some concurrent connection attempts were not handled correctly. (Bug #36625082)

Functionality Added or Changed

- **Important Change:** For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated to version 3.0.15. For more information, see OpenSSL 3.0 Series Release Notes and OpenSSL Security Advisory [3rd September 2024]. (Bug #37021075)
- **Performance; Replication:** The data structure used in tracking binary log transaction dependencies has been changed from `Tree` to `ankerl::unordered_dense::map`, which uses approximately 60% less space, and which should thus contribute to better dependency tracking performance. (Bug #37008442, Bug #37529256)
- Added the `--system-command` option for the `mysql` client, which enables or disables the `system` client command.

  This option is enabled by default. To disable it, use `--system-command=OFF` or `--skip-system-command`, which causes the `system` command to be rejected with an error. (Bug #36377685, WL #16482)

  References: See also: Bug #36248967.
- The MySQL server now supports an alternative syntax for the `GROUP BY` clause `ROLLUP` modifier. Consider the following query:

```
SELECT
    IF(GROUPING(year), 'All years', year) AS year,
    IF(GROUPING(country), 'All countries', country) AS country,
    IF(GROUPING(product), 'All products', product) AS product,
    SUM(profit) AS profit
FROM sales
GROUP BY year, country, product WITH ROLLUP;
```

  Using the alternative syntax, the same query can be written like this:

```
SELECT
    IF(GROUPING(year), 'All years', year) AS year,
    IF(GROUPING(country), 'All countries', country) AS country,
    IF(GROUPING(product), 'All products', product) AS product,
```

```
    SUM(profit) AS profit
FROM sales
GROUP BY ROLLUP (year, country, product);
```

Either version of the query produces the same result.

For more information, see GROUP BY Modifiers, as well, as SELECT Statement. (WL #15843)

Bugs Fixed

- **Performance; InnoDB:** Several functions internal to `InnoDB`, which were defined as inline in MySQL 8.0.28, were found to be no longer inline in MySQL 8.0.33, due in part to refactoring which accompanied improvements made in MySQL 8.0.30 to improve the `InnoDB` adaptive hash index. This had an adverse effect on queries using joins on `InnoDB` tables. (Bug #111538, Bug #35531293)

  References: This issue is a regression of: Bug #81814, Bug #16739204, Bug #23584861.

- **InnoDB:** InnoDB did not allow updating a `REDUNDANT` table column that was altered with `NULL` as the default value using the `INSTANT` algorithm. Instead, MySQL unexpectedly halted. (Bug #36840107)

- **InnoDB:** When redo log capacity was reduced with `log_files_truncate`, under rare circumstances the `file.end_lsn` equaled `log_sys->write_lsn` thus disallowing future redo log writes. (Bug #36730830)

- **InnoDB:** It was possible for the MySQL server to halt unexpectedly when executing a `DELETE` or `UPDATE` statement after a column was dropped using the `INSTANT` algorithm. (Bug #36723117)

- **InnoDB:** FTS index optimization would sometimes not function correctly with tokens equal in collation order but different in terms of bytes. (Bug #36652127)

- **InnoDB:** Replaced `std::this_thread::sleep_for(std::chrono::seconds(0))` usage with `std::this_thread::yield()` to prevent it from being optimized out while compiled. (Bug #36522343)

- **InnoDB:** The upgrade process unexpectedly halted when the database contained a full-text index created in MySQL 5.7 with a user-created `FTS_DOC_ID` column as the primary key.

  Our thanks to Huaxiong Song and the team at Alibaba for contributing to this fix. (Bug #36496164)

- **InnoDB:** Disabled the optimizer for indexes with range conditions on multiple-value virtual columns. (Bug #36341532)

- **InnoDB:** Redesigned the performance schema `data_locks` and `data_lock_waits` tables so that querying them does not require an exclusive global mutex on the transaction or lock system. It now iterates over buckets of hash tables that hold the locks to only latch the actively processed shard, when previously it iterated over the transactions. This also improves the iteration logic complexity in terms of speed and memory to decrease the impact of these queries on the rest of the system.

  Note that the query result might show an incomplete list of transaction locks if it committed, started, or otherwise changed the set of owned locks in-between visiting two buckets. This differs from previous behavior which always showed a consistent snapshot of locks held by individual transactions, although two different transactions could have been presented at different moments. In other words, the new approach gives a consistent view of a single wait queue to show conflicting locks with a waiting lock because they are always in the same bucket, while the old approach could miss some of them because they belonged to other transactions. The old approach would always show all the other locks held by a reported transaction but could miss locks of other transactions even if they were conflicting. (Bug #36302624)

- **InnoDB:** On Windows, fixed a doublewrite buffer regression that slowed file access, and refactored `FILE_FLAG_OVERLAPPED` flag usage for opening files. (Bug #36259487)
- **InnoDB:** A schema mismatch was possible when a table was imported with a different `sql_mode` than when it was created, because strict mode influences the number of nullable columns in a table's secondary index. Now the secondary index trees are also checked for corruption after a tablespace is imported. (Bug #35799038)
- **InnoDB:** Given a table with an FTS index, when its tablespace is discarded the corresponding FTS tables are also dropped. When performing an operation that cleared out these tables, the code incorrectly assumed that the FTS tables must also exist if the FTS indexes were present in the table metadata. (Bug #35343458)
- **InnoDB:** An uninitialized buffer was being written to a temporary file when checking if the system supports a different sector size for FusionIO. This check is made when `innodb_flush_method` is set to `O_DIRECT` or `O_DIRECT_NO_FSYNC`. (Bug #115229, Bug #36705034)
- **InnoDB:** Table rebuild operations involving secondary indexes required greater file I/O for `InnoDB` temporary files as compared with MySQL 8.0.26, which degraded query performance. (Bug #114465, Bug #36444172)
- **InnoDB:** Adding indexes with the parallel index builder was much slower with the Performance Schema enabled due to issues in the `Alter_stage` class. (Bug #113505, Bug #36163502)
- **InnoDB:** Fixed an `innodb.parallel_read_kill` related unit test to better account for recent optimizer changes, and fixed an `innodb.ddl_kill` unit test and its associated macro usage.

  Our thanks to Dmitry Lenev and the team at Percona for contributing to these fixes. (Bug #113002, Bug #115416, Bug #35992036, Bug #36764973)

  References: See also: Bug #112767.
- **Partitioning:** `ALTER TABLE` did not always work correctly with partitioned tables. (Bug #36677952)
- **Group Replication:** Under certain conditions, all secondaries shut down unexpectedly following a brief period of network inactivity on the primary host. (Bug #35642087)

  References: See also: Bug #32673639, Bug #34565855.
- **Group Replication:** Memory aggregated by threads as reported did not account for all `memory/sql/Gtid_set::Interval_chunk` freed by other threads from what was allocated, leading to a incorrect ever-increasing consumption of resources by Group Replication thread `thread/group_rpl/THD_applier_module_receiver`.

  Global memory as reported by `performance_schema.memory_summary_global_by_event_name` was not affected by this issue. (Bug #34819861)
- **Group Replication:** Removed a potential memory leak from `xcom\network\xcom_network_provider_native_lib.cc`. (Bug #115162, Bug #36673883)
- The server did not always handle connections correctly when running with both the thread pool and audit log plugins. (Bug #37039422)
- Packages for Debian-based systems are now built with `WITH_ZLIB=system`, and disable `MYSQL_MAINTAINER_MODE` for debug builds. (Bug #37038213)
- Updated the Kerberos library bundled with commercial builds to version 1.21.3. For more information, see the Kerberos 5 1.21.x Release Notes. (Bug #37034600)

- Removed a heap-use-after-free warning in `regexp::Regexp_facade::~Regexp_facade()`. (Bug #36867806)
- While dumping tablespaces, `mysqldump` did not properly escape certain SQL statements in its output. In addition, the dump now encloses the following identifiers within backticks: `LOGFILE GROUP`, `TABLESPACE`, and `ENGINE`. (Bug #36816986)
- Eliminated unnecessary copying in `StoreLinkedImmutableStringFromTableBuffers()`, improving the performance of some hash joins by 5 to 10 percent. (Bug #36805260)
- A previous fix for an issue in `sql/sql_executor.cc` checked for `const_item()` rather than `const_for_execution()`. (Bug #36804785)

  References: This issue is a regression of: Bug #34951115.
- When executing an index range scan using `IndexRangeScanIterator` the record buffer was never set due to the fact that the data type used for `IndexRangeScanIterator::m_expected_rows` had been changed from double to boolean during refactoring. This unintended change has been reverted.

  In addition, when the record buffer was enabled for index range scans a problem arose with multi-valued indexes used for covering these scans. (This is possible only when they are used as part of index merge scans, since index merge scans force covering index scans.) The source of the problem was that the implementation of `Field_typed_array::key_cmp()` needs the value of the generated column for the indexed expression, and this column is not available in the multi-valued index, so the storage engine cannot safely evaluate the end range condition when filling the record buffer for a covering scan. To fix this, we now disable the record buffer when multi-valued indexes are used for covering index range scans. (Bug #36775910)

  References: See also: Bug #36341532.
- Certain triggers and stored procedures were not handled correctly. (Bug #36775910)
- Some errors raised when referencing external tables were not handled correctly. (Bug #36758378)
- On Debian, stopped stripping `libmysqlclient.a` to increase LTO build compatibility. (Bug #36737581)
- The internal function `my_convert_internal()` sometimes returned a pointer rather than the intended value. (Bug #36684463)

  References: This issue is a regression of: Bug #36479091.
- Removed a memory issue in the server component. (Bug #36600205)
- Fixed a memory leak in the `mysql` client. (Bug #36600203)
- The `AES_ENCRYPT()` function did not always return a valid result. (Bug #36593265)
- In some cases a join involving a very large number of rows and many tables was not processed correctly. (Bug #36562979)
- Preparation of table value constructors did not track the number of hidden columns added for expressions in an `ORDER BY` clause. This could cause problems later in the resolution process, possibly leading to an unplanned server exit.

  We fix this by counting the number of hidden items in table value constructors in the same way as this is done in other query blocks. (Bug #36560156)

  References: This issue is a regression of: Bug #35785452.

- `DROP INDEX` with the addition of a `FULLTEXT` index in the same transaction sometimes led to an unplanned server exit. (Bug #36559642)
- Some combinations of optimizer hints did not function correctly. (Bug #36492114)
- Following the fix for a previous issue, a `const` item that is part of a `GROUP BY` and is not found in the select list is not added as a hidden item to the field list, but this was not taken into consideration while checking for replacements for expressions involving `ROLLUP` operations while creating a temporary table. We fix this by checking whether the item that is not found in the field list is a `const` item, and if so, we use the same item as a replacement. (Bug #36444257)

  References: This issue is a regression of: Bug #34951115.
- Removed assertions found during testing of the data masking functions `gen_rnd_pan()` and `mask_ssn()`. (Bug #36397869, Bug #36398272)

  References: See also: Bug #36398221.
- Fixed an issue relating to FTS comparisons.

  Our thanks to Shaohua Wang and the team at Alibaba for the contribution. (Bug #36210202)
- Some `CREATE TABLE ... SELECT` statements were not always processed correctly. (Bug #36138460)
- RelWithDebInfo builds are no longer compiled with the "-g1" flag, thus increasing the available debugging information and generated file sizes. (Bug #36111629)

  References: See also: Bug #33664929.
- Corrected potential misbehavior when the server was run with `--skip-grant-tables`. (Bug #36043213)
- Fixed a regression in an earlier fix for a problem with references to columns from tables of outer query blocks in the `ON` condition of a join. (Bug #35854686)

  References: This issue is a regression of: Bug #96946, Bug #30350696.
- In certain rare cases, a `CREATE TABLE` statement involving the creation of a table with foreign keys was not processed correctly. (Bug #35553557, Bug #36350852)
- Fixed an issue relating to `SHOW INDEX` and generated columns. (Bug #35497623)
- In some cases, table DDL in prepared statements was not handled correctly. (Bug #35221658)
- A left join with an impossible condition as part of an `ON` clause was not optimized as in MySQL 5.7, so that, in later versions, the query executed more quickly without the impossible condition than with it. An example of such a query, impossible condition included, is `SELECT * FROM t1 JOIN t2 ON t1.c1=t2.c1 AND 1=2`. (Bug #34668756)
- The XML function `ExtractValue()` did not handle arguments containing accented characters correctly. (Bug #117778, Bug #36236440)
- Changed the SLES 15 target platform from openSUSE 15.5 to 15.6, which also means using GCC 13 instead of GCC 12 and building against the system's OpenSSL 3.x instead of OpenSSL 1.1.x. (Bug #115535, Bug #36934913)
- Fixed a typo in the Debian package description.

  Our thanks to Henning Pöttker for the contribution. (Bug #115363, Bug #36749142)

- `MYSQL_TYPE_JSON` was missing from the description of binary resultsets in the MySQL source code documentation.

  Our thanks to Daniël van Eeden for the contribution. (Bug #115360, Bug #36740656)

- With `prefer_ordering_index=off`, a query with no reference key reverted to scanning and sorting the full table even though it should have been possible to avoid the sort.

  Our thanks to Daniel Nichter for the contribution. (Bug #113699, Bug #36213938)

- A query using a greater-than (`>`) or less-than (`<`) comparison with a multi-valued index executed much more slowly than the same query using an equality (`=`) comparison with the same index. (Bug #104897, Bug #33334911)

Changes in MySQL 8.4.2 (2024-07-23)

Bugs Fixed

- **InnoDB:** In some cases, following the creation of a very large number of tables (8001 or more), the server could not be restarted successfully. (Bug #36808732)

  References: This issue is a regression of: Bug #33398681.

- **InnoDB:** Improved tablespace file scan performance at startup. (Bug #110402, Bug #35200385)

- **Group Replication:** Running a `CREATE TABLE ... SELECT` statement on a source coming from an asynchronous channel to Group Replication led to errors on the replica. (Bug #36784284)

Changes in MySQL 8.4.1 (2024-07-01)

Important

This release is no longer available for download. It was removed due to a critical issue that could stop the server from restarting following the creation of a very large number of tables (8001 or more). Please upgrade to MySQL 8.4.2 instead.

- Audit Log Notes
- Authentication Notes
- C API Notes
- Compilation Notes
- Component Notes
- Configuration Notes
- Connection Management Notes
- Data Dictionary Notes
- Performance Schema Notes
- Pluggable Authentication
- Thread Pool Notes
- X Plugin Notes

- Bugs Fixed

Audit Log Notes

- Audit log pruning did not function after removing or renaming a file from the audit log. Now pruning continues in such cases, but with a warning printed in the error log stating that it was not possible to delete the missing audit log file. (Bug #35902913)
- MySQL now calls `plugin->deinit()` with a valid plugin struct as an argument regardless of the plugin's type.

  Our thanks to Martin Alderete for the contribution.

Authentication Notes

- Improved log messages to provide clear reasons for `Access denied` errors when using the `authentication_ldap_sasl` plugin without proxying. (Bug #35317691)
- MySQL LDAP SASL authentication, when used with the GSSAPI method to access an OpenLDAP server, was rejected with the MySQL server error `Plugin authentication_ldap_sasl reported: 'LDAP authentication failed or group retrieval failed: LDAP error: Invalid DN syntax'`, because OpenLDAP did not recognize the root DN used. (Bug #32631511)

C API Notes

- C API applications stalled while receiving results for server side prepared statements.

Compilation Notes

- **macOS:** The Xcode version of `zlib` was removed from the default list of system libraries to use when configuring with `-DWITH_SYSTEM_LIBS=ON`. (Bug #36537593)
- **Microsoft Windows:** The `BUILD_ALL` target did not work when compiling on Windows. (Bug #36424619)
- **Microsoft Windows:** Excessive RAM usage led to disk swapping when compiling MySQL on Windows using Ninja. (Bug #36399256)
- Maintainer mode is now disabled when building the debug version of the server for `.deb` packages. (Bug #36619757)
- Upgraded the bundled `googletest` and `googlemock` sources to version 1.14.0. (Bug #36562482)
- Added a missing dependency on `GenError`. (Bug #36551721)
- When compiling on Fedora 38, `grep -E` is now used in place of `egrep`. (Bug #36507549)
- It is now possible on Linux systems to build MySQL using a bundled `tcmalloc` library that is provided with the source by specifying `-DWITH_TCMALLOC=BUNDLED`. This is supported on Linux only. (Bug #36313839)
- The bundled `tcmalloc()` is now used when building MySQL on Enterprise Linux 8. (Bug #114844, Bug #35674008)
- Removed warnings raised in `sql/statement/ed_connection.cc` when building on Ubuntu 23.04. (Bug #114436, Bug #36428465)
- Linux `aarch64` platform binaries are now built using `patchelf --page-size=65536` for compatibility with systems using either 4k or 64k for the page size. (Bug #114233, Bug #36393794)

Component Notes

- The values for component options set using the `--loose` prefix were not read when the component was installed. (Bug #28341329)

Configuration Notes

- **Microsoft Windows:** On Windows, *MySQL Configurator* was updated to support in-place upgrades as per Upgrade Paths. (Bug #36685422)
- **Microsoft Windows:** For MSI installations on Windows, *MySQL Configurator* now automatically upgrades MySQL 8.4 LTS installations without user intervention. (WL #16274)

Connection Management Notes

- The `conn_delay/Waiting in connection_control plugin` stage was not reset after a delay introduced by the connection control plugin which resulted in incorrect monitoring information. (Bug #35205358)

Data Dictionary Notes

- Attempting to upgrade a `MyISAM` table containing a mix of regular columns and generated columns from MySQL 5.7 to 8.0 or later led to table corruption. (Bug #105301, Bug #33503328)

Performance Schema Notes

- **Group Replication:** The following tables did not contain data on replication channels which did not have a configured hostname, such as Group Replication recovery channels:

  - `REPLICATION_CONNECTION_STATUS`
  - `REPLICATION_CONNECTION_CONFIGURATION`
  - `REPLICATION_APPLIER_CONFIGURATION`
  - `REPLICATION_APPLIER_STATUS`
  - `REPLICATION_APPLIER_STATUS_BY_COORDINATOR`
  - `REPLICATION_APPLIER_STATUS_BY_WORKER`

  As of this release, these tables contain data for partially configured Group Replication channels. (Bug #36018242)

- Under certain conditions, a race condition could result in the amount of RAM used by `TABLE_HANDLES` increasing to a maximum of 9GB. (Bug #36170903)
- The `PROCESSLIST_INFO` column of `THREADS` was not updated when executing a prepared statement.

  Thanks to Daniel Lenski and Amazon for the contribution. (Bug #104121, Bug #33057164)

Pluggable Authentication

- The deprecation warning issued when authenticating with the `mysql_native_password` plugin is now issued only once. (Bug #35792948)

Thread Pool Notes

- Connecting to a thread group that had no connection handler threads stalled. We fix this by making sure that connection handler threads terminate only if there is at least one connection thread left. (Bug #36550125)
- Previous refactoring incorrectly removed the connection locking performed when processing incoming connection requests, which ledto a race condition between the thread adding new connections and the connection handler thread processing them. This appeared to cause a situation in which connection requests might be ignored and not processed, so that the connection attempt appeared to hang.

  We fix this by taking the connection before processing the queue, and releasing it before waking or creating new threads. (Bug #36548687)
- It was possible to set the `thread_pool_longrun_trx_limit` system variable to values outside its stated range.

  In addition, settings for this variable were not reflected in the output of SHOW VARIABLES or SELECT. (Bug #36347102, Bug #36371145)
- `SET PERSIST_ONLY` did not work correctly with `thread_pool_max_transactions_limit`. (Bug #35019884)

X Plugin Notes

- The system variable `caching_sha2_password_digest_rounds` could not be set to a non-default value using X Protocol. (Bug #36402455)
- An outdated link to the MySQL documentation in the `mysql_function_names` unit test source file has been updated.

  Our thanks to Minha Jeong for the contribution. (Bug #113500, Bug #36137217)

Bugs Fixed

- **InnoDB:** MySQL unexpectedly halted on an `UPDATE` after an `ALTER TABLE` operation. (Bug #36571091)

  References: This issue is a regression of: Bug #35183686.
- **InnoDB:** The log index size calculation now accounts for column order changes. (Bug #36526369)

  References: This issue is a regression of: Bug #35183686.
- **InnoDB:** File system operations performed by `InnoDB` now consistently `fsync` the parent directory when performing directory altering tasks. (Bug #36174938)
- **InnoDB:** In debug builds, setting the `innodb_interpreter_output` debug variable would cause the server to unexpectedly halt. This is now a read-only variable. (Bug #36041032)
- **InnoDB:** Improved `os_innodb_umask` handling, and made it read-only. (Bug #35932118)

  References: This issue is a regression of: Bug #29472125.
- **InnoDB:** For tables created with an index on a column that was too wide for the redundant row format (allowed before MySQL 5.7.35), an in-place upgrade silently imported the table but it was not accessible,

which interfered with making backups. Now all operations that involve using the invalid index are rejected with `ER_INDEX_CORRUPT` until the index is dropped. An `ER_IB_INDEX_PART_TOO_LONG` error is also reported in the error log. (Bug #35869747)

References: See also: Bug #34826861.

- **InnoDB:** An `InnoDB` assertion error referencing an invalid column index was triggered when the column index was valid. (Bug #34800754)
- **InnoDB:** With an empty `XA` transaction, shutting the server down after an `XA START` would cause the server to halt unexpectedly. (Bug #32416819)
- **InnoDB:** Shutting down the replication applier or binlog applier while processing an empty XA transaction caused the system to unexpectedly halt. (Bug #32416819)
- **InnoDB:** Removed unnecessary heap usage in the `Validate_files::check()` function.

  Our thanks to Huaxiong Song for the contribution. (Bug #115041, Bug #36626203)
- **InnoDB:** If a partition table was read with `innodb_parallel_read_threads=1`, read performance greatly decreased from any table after 256 reads. InnoDB behaved as if it reached the maximum capacity of parallel read threads despite not using any.

  Our thanks to Ke Yu for the contribution. (Bug #114154, Bug #36347408)
- **InnoDB:** The result from a spatial index containing a column with a spatial reference identifier (SRID) attribute was empty. In addition, using `FORCE INDEX` to force a covering index scan on a spatial index led to an assertion. (Bug #112676, Bug #114200, Bug #35894664, Bug #36361834)
- **InnoDB:** `SELECT ... GROUP BY` queries were at least twice as slow with the TempTable engine than the Memory engine. (Bug #107700, Bug #34338001)
- **Replication:** If a source contained a stored, generated column populated by a JSON function and `binlog_row_image` was set to `MINIMAL`, any subsequent update or deletion on the underlying column failed with the following error:

  ```
  Invalid JSON text in argument 1 to function json_extract: 'The document is empty.'
  ```

  The replica attempted to re-evaluate the generated column and failed with that error because the underlying column was unavailable. As of this release, stored, generated columns are not re-evaluated when the underlying columns are unavailable. (Bug #36515172)
- **Replication:** When running GTID-based replication with `relay_log_space_limit` enabled, a restart of the auto positioning protocol sometimes resulted in an infinite loop, leading to a deadlock in replication. This was because `relay_log_space_limit` was not honored, not only for transactions whose size exceed this limit, but when the replica could not purge previous logs as well.

  To fix this issue, we make the following changes:

  - The receiver respects `relay_log_space_limit` as set by the user, unless a transaction received by the receiver cannot fit into the purged relay log. Before queuing the received transaction, receiver now checks whether scheduling a full transaction is possible. If not, the receiver performs the following actions:
    - Sets the flag indicating that receiver is waiting
    - Rotates the relay log

    - Waits until it is notified that relay log purge was executed and that the applier has purged all available relay logs; after this, the receiver may queue a transaction without checking the limit again

  - Before moving to the next file, the coordinator checks whether the receiver is waiting for available relay log space. If so, the coordinator forcibly purges the applied logs, including the current relay log file. To purge the current relay log file safely, the coordinator must do the following:

    - Synchronize all of its workers before moving to the next file

    - Forcibly update group positions, which is necessary to allow current purging of the relay log

    - Update the variable read by the receiver which contains relay log filename to which applier was moved

    These operations are allowed because we know that receiver waits at a transaction boundary and rotates the relay log before waiting.

  (Bug #36507020)

- **Replication:** Worker jobs now contain information about the relay log file which initiated the transaction, instead of using the default defined by `relay_log`. (Bug #36395631)

- **Replication:** Handling an incident while transactions were being committed to the binary log caused MySQL to wait indefinitely. (Bug #35671897)

- **Group Replication:** Removed a memory leak from `/xcom/gcs_xcom_networking.cc`. (Bug #36532199)

- **Group Replication:** Under certain circumstances, if a primary's host experienced network inactivity of 20 seconds or more, the secondaries could stop unexpectedly. (Bug #36306144)

- **Group Replication:** Under certain circumstances, if garbage collection occurred just before a relay log rotation, it could cause the applier to stop applying new transactions on the secondary members.

  This was caused by garbage collection incrementing the relay log's `last_committed` and `sequence_number`, creating a gap in the recorded `sequence_number` after the log rotation. The applier was unaffected if the gap occurred anywhere else in the relay log.

  As of this release, only `last_committed` is updated during garbage collection. (Bug #36280130, Bug #36446250)

- **JSON:** Added missing checks for error handling to `NULLIF()`, `COALESCE()`, and the shift (`>>`) operator. (Bug #113668, Bug #35513196, Bug #36198403)

  References: See also: Bug #31358416.

- **MySQL NDB ClusterJ:** Running the ClusterJ test suite resulted in an error message saying a number of threads did not exist. That was due to some wrong handling of threads and connections, which was corrected by this patch. (Bug #36086735)

- On macOS, reinstated the MySQL preference pane's ability to load a custom `my.cnf` configuration file. This includes loading it before initializing a new data directory. (Bug #36630493)

- On Fedora 40, there were conflicts when installing MySQL 8.4.0 community server on a system with native `mysql-server` packages present. (Bug #36575524)

- Averages of certain numbers were not always computed correctly. (Bug #36563773)

- The following files in `strings` contained incorrect license information:

  - `mb_wc.h`
  - `ctype-uca.cc`
  - `ctype-ucs2.cc`
  - `ctype-utf8.cc`
  - `dtoa.cc`
  - `strxmov.cc`
  - `strxnmov.cc`

  (Bug #36506181)

- In certain unusual cases, the `UpdateXML()` function did not process all of its arguments correctly. (Bug #36479091)

- Explaining a query which used `FORCE INDEX` on a spatial index containing a column with SRID attributes led to an unplanned exit. (Bug #36418426)

- Added the `ER_LOG_PARTITION_PREFIX_KEY_NOT_SUPPORTED` error definition, which references functionality added in MySQL 8.4.0. (Bug #36350938)

- Updated `BuildRequire` rules to align with versions now required for CMake and Bison. (Bug #36343254)

- Some prepared statements were not reprepared correctly. (Bug #36267792, Bug #35712413)

  References: This issue is a regression of: Bug #34929930.

- When incrementing the reference count for an expression, underlying expressions within this expression are not looked at. While removing an expression, after decrementing the reference count, even the underlying expressions were examined, which led to unintentional deletion of the underlying expressions. This issue manifested in `Item_ref::real_item()` as well as in an assert in `sql/item.h`. We fix this by not looking at the underlying expression unless the current expression contains the only remaining reference. (Bug #36204344, Bug #36356279)

- Under certain conditions, `EXPLAIN FORMAT=JSON FOR CONNECTION` sometimes led to an unplanned exit. (Bug #36189820)

- Some `CREATE USER` statements were not handled correctly. (Bug #36022885)

- In certain cases, a lateral join was not handled correctly. (Bug #35945239)

  References: See also: Bug #107700, Bug #34338001. This issue is a regression of: Bug #32644631.

- For a `SELECT` with `ORDER BY` and `LIMIT`, the optimizer first chose a full table scan with a very expensive cost, then performed another check and used the `perform_order_index` type of path, but this was not reflected by the cost in the optimizer plan. (Bug #35930969)

- Client connections were not alway terminated correctly during shutdown. (Bug #35854919)

- Executing `mysqldump` on a replica would insert the `FLUSH TABLES` operation, an operation that writes to the binary log. Now `FLUSH LOCAL TABLES` is inserted instead to prevent GTID related issues during replication due to these binary log changes.

The workaround was to set the `--source-data` option to 1 or 2. (Bug #35665076)

References: This issue is a regression of: Bug #33630199.

- All internal ACL bitmask variables are now explicitly 32 bits (`uint32_t`). (Bug #35507223)
- It was not possible to add a functional index on `FIND_IN_SET()`. (Bug #35352161)
- Running two concurrent `OPTIMIZE TABLE` statements on the same table with fulltext indexes and `innodb_optimize_fulltext_only` enabled sometimes caused the server to exit. (Bug #34929814)
- The `gen_range()` function as implemented by the (deprecated) data masking plugin did not always return the correct result.

  This issue affected the data masking plugin only, and did not affect the data masking component which supersedes it. (Bug #34163992)
- In some circumstances, such when DDL operations were performed on a very large number of tables, the error log was flooded with warnings from background histogram updates; the offending warning was concerning a failure to acquire metadata locks on a table.

  To remedy this problem we now throttle messages written to the error log from background histogram update operations, the rate being capped at one message per minute, which should suffice for the user to identify potential problems with background histogram updates. In addition, we downgrade all error events that occur during background histogram updates from errors to warnings. (Bug #114845, Bug #36574298)
- On macOS, the DMG now installs the keyring component instead of installing keyring plugin functionality that was removed in MySQL Server 8.4.0.

  The 8.4.0 workaround was to disable the "Keyring Data File" option in the preference pane or to manually remove the two keyring lines from the launchd plist. (Bug #114836, Bug #36577944)
- Fixed an erroneous comment in `include/my_command.h`.

  Our thanks to Sho Nakazono for the contribution. (Bug #114507, Bug #36455468)
- It was possible for a deterministic stored function to return an incorrect result when the function used `JOIN ON` inside the `return` statement. If the query needed to be reprepared due to a table metadata caused by, for example, `FLUSH TABLES` between two executions, the `ON` clause was sometimes lost. (Bug #114235, Bug #36379879)
- Added the missing `mysql-community-libs-compat` package for the EL8 and EL9 platforms. (Bug #112949, Bug #35975348)

Changes in MySQL 8.4.0 (2024-04-30)

- Audit Log Notes
- C API Notes
- Character Set Support
- Compilation Notes
- Configuration Notes

- Deprecation and Removal Notes
- Firewall Notes
- INFORMATION_SCHEMA Notes
- Installation Notes
- Keyring Notes
- Optimizer Notes
- Performance Schema Notes
- Server Administration
- Thread Pool Notes
- Functionality Added or Changed
- Bugs Fixed

Audit Log Notes

- Invoking `audit_api_message_emit_udf()` with arguments of mixed types could lead to an unplanned shutdown of the server. (Bug #36301441)
- Audit log filtering by type, using error as the type, did not work correctly. (Bug #36142157)
- Following an unplanned shutdown and restart, the file that was in use by the server for writing at the time may be broken or otherwise unreadable. The Audit Log plugin log message indicating unreadability of the file was shown as an error; now instead this message is treated as a warning. (Bug #36118809)

C API Notes

- **Important Change:** The following MySQL C API functions, removed in MySQL 8.3, have been reimplemented and restored in MySQL 8.4.0:
  - `mysql_kill()`: In place of `COM_PROCESS_KILL` (removed), this function has been reimplemented such that it uses `mysql_real_query()` to execute `KILL`.
  - `mysql_list_fields()`: Restored as previously implemented, along with `COM_FIELD_LIST`.
  - `mysql_list_processes()`: Reimplemented using `mysql_real_query()` to execute `SHOW PROCESSLIST`, in place of `COM_PROCESS_INFO` (removed).
  - `mysql_refresh()`: Reimplemented using `mysql_real_query()` to execute `FLUSH` statements in place of `COM_REFRESH`, which was removed in MySQL 8.3.
  - `mysql_reload()`
  - `mysql_shutdown()`: Reimplemented using mysql_real_query() to execute a shutdown command rather than `COM_SHUTDOWN`, removed in MySQL 8.3.
  - `mysql_ssl_set()`
  - `mysql_stmt_bind_param()`

  The functions just listed are supported for the lifetime of the MySQL 8.4 series. (WL #16221)

- **Microsoft Windows:** Third party DLL files on which MySQL plugins depend are located, when installed, in the same directory as the MySQL executables. The default Windows behavior is to look for dependences in the same directory as the current executable, which is not appropriate for clients using `libmysql.dll` outside of the installation directory.

  We fix this by that changing the default behavior of MySQL clients so that the loader looks for dependencies in the directory of current module (the executable or `libmysql.dll`). In addition, since `libsasl.dll` expects to load all its required dependencies from a directory of its won, SASL plugins are now located in a dedicated subdirectory. (Bug #36006295)

Character Set Support

- When the `character_set_server` system variable was set using `SET PERSIST` or `SET GLOBAL`, it did not take effect for new client sessions or for a client establishing a connection to the server after the server was restarted. The only workaround was to set the corresponding command-line option when starting the server.

  To fix this, we now make sure that, at the time of server restart, the configuration data is read in the correct order so that the variable setting takes effect as expected. (Bug #35529604)

Compilation Notes

- The `libevent` library has been removed. (Bug #36357190)
- Added the `libcno` library. (Bug #36357181)
- Some of the files in `extra/libbacktrace` contained incorrect licensing information, copyright information, or both. (Bug #36118772)
- Warnings for unused variables are turned into compilation errors when compiling with `-DMYSQL_MAINTAINER_MODE=1`. To avoid this use `-DMYSQL_MAINTAINER_MODE=0` to disable such errors. (Bug #113662, Bug #36198423)

Configuration Notes

- **Microsoft Windows:** On Windows, *MySQL Configurator* incorrectly altered the configuration settings after the **Back** and **Next** buttons were used. (Bug #36156577)
- **Microsoft Windows:** On Windows, *MySQL Configurator* no longer opens when removing a MySQL Server that was not configured. (Bug #35709927)
- **Microsoft Windows:** On Windows, MySQL Configurator stopped adding the `default_authentication_plugin` variable to the generated `my.ini` file, a variable removed in MySQL Server 8.4. It also removes it when upgrading an installation to MySQL 8.4. Note that the replacement variable `authentication_policy` is not set by *MySQL Configurator*. (WL #16137)
- **Microsoft Windows:** On Windows, MySQL Configurator no longer allows upgrading from MySQL 5.7 or earlier, when before it allowed the upgrade to execute after stating that it was not officially supported. (WL #16138)

Deprecation and Removal Notes

- **Important Change:** The deprecated `mysql_native_password` authentication plugin is now disabled by default. It can be enabled by starting MySQL with the new `--mysql-native-password=ON` server option, or by adding `mysql_native_password=ON` to the `[mysqld]` section of your MySQL configuration file.

For more information, see Native Pluggable Authentication. (Bug #36337893)

- **Partitioning:** Silent omission of columns with index prefixes as part of a table's partitioning key was deprecated in MySQL 8.0.21, and generated a warning. In this release, the use of any such columns in the proposed partitioning key is now expressly disallowed, and causes the `CREATE TABLE` or `ALTER TABLE` statement in which it occurs to be rejected with an error.

  For more information, see Column index prefixes not supported for key partitioning, and KEY Partitioning. (WL #16054)

  References: See also: Bug #31100205.

- **Replication:** Syntax for a number of features relating to MySQL Replication that was deprecated in previous versions of MySQL has now been removed. These features include aspects of SQL statement syntax as well as several system status variables in the MySQL server. These changes are detailed following.

  SQL statements removed. The following SQL statements have been removed (replacements in brackets): `START SLAVE` (`START REPLICA`); `STOP SLAVE` (`STOP REPLICA`); `SHOW SLAVE STATUS` (`SHOW REPLICA STATUS`); `SHOW SLAVE HOSTS` (`SHOW REPLICAS`); `RESET SLAVE` (`RESET REPLICA`); `CHANGE MASTER TO` (`CHANGE REPLICATION SOURCE TO`); `RESET MASTER` (`RESET BINARY LOGS AND GTIDS`); `SHOW MASTER STATUS` (`SHOW BINARY LOG STATUS`); `PURGE MASTER LOGS` (`PURGE BINARY LOGS`); and `SHOW MASTER LOGS` (`SHOW BINARY LOGS`).

  The statements just listed have also been removed from all MySQL test programs and files, and elsewhere, where used internally.

  Statement options removed. The following options formerly supported by `CHANGE REPLICATION SOURCE TO` and `START REPLICA` have been removed and are no longer accepted by the server. They are listed here for each of these statements, with their replacements in brackets:

  - `CHANGE REPLICATION SOURCE TO` options removed:

    `MASTER_AUTO_POSITION` (`SOURCE_AUTO_POSITION`), `MASTER_HOST` (`SOURCE_HOST`), `MASTER_BIND` (`SOURCE_BIND`), `MASTER_USER` (`SOURCE_USER`), `MASTER_PASSWORD` (`SOURCE_PASSWORD`), `MASTER_PORT` (`SOURCE_PORT`), `MASTER_CONNECT_RETRY` (`SOURCE_CONNECT_RETRY`), `MASTER_RETRY_COUNT` (`SOURCE_RETRY_COUNT`), `MASTER_DELAY` (`SOURCE_DELAY`), `MASTER_SSL` (`SOURCE_SSL`), `MASTER_SSL_CA` (`SOURCE_SSL_CA`), `MASTER_SSL_CAPATH` (`SOURCE_SSL_CAPATH`), `MASTER_SSL_CIPHER` (`SOURCE_SSL_CIPHER`), `MASTER_SSL_CRL` (`SOURCE_SSL_CRL`), `MASTER_SSL_CRLPATH` (`SOURCE_SSL_CRLPATH`), `MASTER_SSL_KEY` (`SOURCE_SSL_KEY`), `MASTER_SSL_VERIFY_SERVER_CERT` (`SOURCE_SSL_VERIFY_SERVER_CERT`), `MASTER_TLS_VERSION` (`SOURCE_TLS_VERSION`), `MASTER_TLS_CIPHERSUITES` (`SOURCE_TLS_CIPHERSUITES`), `MASTER_SSL_CERT` (`SOURCE_SSL_CERT`), `MASTER_PUBLIC_KEY_PATH` (`SOURCE_PUBLIC_KEY_PATH`), `GET_MASTER_PUBLIC_KEY` (`GET_SOURCE_PUBLIC_KEY`), `MASTER_HEARTBEAT_PERIOD` (`SOURCE_HEARTBEAT_PERIOD`), `MASTER_COMPRESSION_ALGORITHMS` (`SOURCE_COMPRESSION_ALGORITHMS`), `MASTER_ZSTD_COMPRESSION_LEVEL` (`SOURCE_ZSTD_COMPRESSION_LEVEL`), `MASTER_LOG_FILE` (`SOURCE_LOG_FILE`), and `MASTER_LOG_POS` (`SOURCE_LOG_POS`).

  - `START REPLICA` options removed: `MASTER_LOG_FILE` (`SOURCE_LOG_FILE`) and `MASTER_LOG_POS` (`SOURCE_LOG_POS`).

  Status variables removed. Also as part of this work, the following system status variables have been removed from the server, and no longer appear in the output of statements such as `SHOW`

STATUS. These variables are listed here, with their replacements in brackets: Com_slave_start (Com_replica_start); Com_slave_stop (Com_replica_stop); Com_show_slave_status (Com_show_replica_status); Com_show_slave_hosts (Com_show_replicas); Com_show_master_status (Com_show_binary_log_status); and Com_change_master (Com_change_replication_source).

See also Com_xxx Variables. (WL #15831, WL #16063, WL #16069, WL #16086, WL #16087, WL #16088, WL #16089, WL #16090)

- **Group Replication:** The group_replication_allow_local_lower_version_join system variable is now deprecated, and setting it raises a warning (ER_WARN_DEPRECATED_SYNTAX_NO_REPLACEMENT).

  You should expect this variable to be removed in a future version of MySQL. Since there is no longer any reason to allow incompatible members to join a group, no replacement for this functionality is planned. (WL #16018)

- A number of server options and variables supported in previous versions of MySQL have been removed in this release. Attempting to set any of them in MySQL 8.4 raises an error. These options and variables are listed here:

  - binlog_transaction_dependency_tracking: Deprecated in MySQL 8.0.35 and MySQL 8.2.0.

    There are no plans to replace this variable or its functionality, which has been made internal to the server: Now, when multithreaded replicas are in use, the source mysqld uses always writesets to generate dependency information for the binary log; this has the same effect as setting binlog_transaction_dependency_tracking to WRITESET in previous versions of MySQL.

  - group_replication_recovery_complete_at: Deprecated in MySQL 8.0.34.

    Beginning with this release, the policy applied during the distributed recovery process is always to mark a new member online only after it has received, certified, and applied all transactions that took place before it joined the group; this is equivalent to setting

`group_replication_recovery_complete_at` to `TRANSACTIONS_APPLIED` in previous versions of MySQL.

- `avoid_temporal_upgrade` and `show_old_temporals`: Both deprecated in MySQL 5.6. Each of these variables no longer had any effect, and has been removed. There are no plans to replace either variable.
- `--no-dd-upgrade`: Deprecated in MySQL 8.0.16, now removed. Use `--upgrade=NONE` instead.
- `--old` and `--new`: Both deprecated in MySQL 8.0.35 and MySQL 8.2.0, and now removed.
- `--language`: This option was deprecated in MySQL 5.5, and has now been removed.
- The `--ssl`, `--skip-ssl`, and `--admin-ssl` server options, as well as the `have_ssl` and `have_openssl` server system variables, were all deprecated in MySQL 8.0.26, and are all removed in this release. Use `--tls-version` and `--admin-tls-version` instead.
- `default_authentication_plugin`: Deprecated in MySQL 8.0.27, and now removed. Use `authentication_policy` instead.

  You should also be aware that the syntax for setting the `authentication_policy` variable has changed; see its description in the Manual for more information.

(Bug #36337893, WL #9677, WL #13965, WL #15461, WL #15839, WL #16056, WL #16058, WL #16059, WL #16095)

- In cases where an aliased table was referenced in `EXPLAIN` output, the table name was qualified with a schema name, which was not necessary and could result in confusion. These schema qualifications have been removed from the output. (Bug #36053664)
- The unused `INFORMATION_SCHEMA.TABLESPACES` table, deprecated in MySQL 8.0.22, has now been removed.

  For `InnoDB` tables, the Information Schema `INNODB_TABLESPACES` and `INNODB_DATAFILES` tables provide tablespace metadata. (WL #14065)
- `LOW_PRIORITY` used with `LOCK TABLES ... WRITE` had had no effect since MySQL 5.5, and was deprecated in MySQL 5.6. It is removed in this release; including `LOW_PRIORITY` in `LOCK TABLES` now causes a syntax error. (WL #16057)
- Support for use of the `AUTO_INCREMENT` modifier with `FLOAT` and `DOUBLE` columns was deprecated in MySQL 8.0, and is now removed. Attempting to use these together in `CREATE TABLE` and `ALTER TABLE` statements now causes an `Incorrect column specifier for column` error (`ER_WRONG_FIELD_SPEC`).

  

  Important

  Prior to upgrading to this release, you *must* alter any table having a `FLOAT ... AUTO_INCREMENT` or `DOUBLE ... AUTO_INCREMENT` column so that it no longer uses either of these. Otherwise, the table cannot be upgraded.

  (WL #13103)
- The `mysql_ssl_rsa_setup` utility, which was deprecated in MySQL 8.0.34, is removed in this release. For MySQL distributions compiled using OpenSSL, the MySQL server can perform automatic generation of missing SSL and RSA files at startup. For more inforamtion, Creating SSL and RSA Certificates and Keys using MySQL. (WL #16205)

- This release removes support for the `ENGINE` clause from the following SQL statements:

  - `DROP TABLESPACE` (all variants)
  - `ALTER TABLESPACE ... DROP DATAFILE`
  - All other variants of `ALTER TABLESPACE`, with the two exceptions listed here:

    1. `ALTER TABLESPACE ... ADD DATAFILE ENGINE={NDB|NDBCLUSTER}`
    2. `ALTER UNDO TABLESPACE ... SET {ACTIVE|INACTIVE} ENGINE=INNODB`

  Other than in the exceptional cases listed previously, use of the `ENGINE` clause with `ALTER TABLESPACE` or `DROP TABLESPACE` causes the statement to be rejected with an error.

  `ENGINE` clauses for the `ALTER TABLESPACE` and `DROP TABLESPACE` statements were deprecated in MySQL 8.0. (WL #16055)

- The `SET_USER_ID` privilege, deprecated in MySQL 8.2.0, has been removed in this release, and its use in `GRANT` statements now causes a syntax error. Use the `SET_ANY_DEFINER` and `ALLOW_NONEXISTENT_DEFINER` privileges instead. (WL #15875)

- Removed the `mysql_upgrade` utility, which was deprecated in MySQL 8.0.16. (WL #16096)

- Removed the deprecated `mysqlpump` utility along with its associated `lz4_decompress` and `zlib_decompress` helper utilities. Instead, use `mysqldump` or the MySQL Shell dump utilities. (WL #16096)

- The following plugins have been removed. They are noted in the list provided here, along with any server system variables, CMake options, and other features associated with them which have also been removed:

  - `authentication_fido`, `authentication_fido_client`: Use `authentication_webauthn` instead; see WebAuthn Pluggable Authentication.

    The `authentication_fido_rp_id` server system variable, `mysql` client `--fido-register-factor` option, and the `-DWITH_FIDO` CMake option have also been removed.

  - `keyring_file`: Use `component_keyring_file` instead; see Using the component_keyring_file File-Based Keyring Component.

    The `keyring_file_data` system variable has also been removed. In addition, the CMake options `-DINSTALL_MYSQLKEYRINGDIR` and `-DWITH_KEYRING_TEST` have been removed.

  - `keyring_encrypted_file`: Use `component_keyring_encrypted_file` instead; see Using the component_keyring_encrypted_file Encrypted File-Based Keyring Component.

    The `keyring_encrypted_file_data` and `keyring_encrypted_file_password` server system variables have also been removed.

  - `keyring_oci`: Use `component_keyring_oci` instead; see Using the Oracle Cloud Infrastructure Vault Keyring Component.

    The following server system variables have also been removed: `keyring_oci_ca_certificate`, `keyring_oci_compartment`, `keyring_oci_encryption_endpoint`, `keyring_oci_key_file`, `keyring_oci_key_fingerprint`, `keyring_oci_management_endpoint`, `keyring_oci_master_key`,

`keyring_oci_secrets_endpoint`, `keyring_oci_tenancy`, `keyring_oci_user`, `keyring_oci_vaults_endpoint`, and `keyring_oci_virtual_vault`.

- `openssl_udf`: Use the MySQL Enterprise Encryption component instead; see MySQL Enterprise Encryption.

(WL #15937, WL #15938, WL #15939, WL #15941, WL #16140)

- Support for weak encryption ciphers has been removed. This means that, when configuring encrypted connections, MySQL no longer allows specifying any cipher that does not meet the following conditions:

  - Proper TLS version (TLS v1.2 or TLSv1.3, as appropriate)
  - Forward secrecy
  - SHA2 in cipher, certificate, or both
  - AES in GCM or any other AEAD algorithms or modes

  This has implications for setting the system variables `ssl_cipher`, `admin_ssl_cipher`, `tls_ciphersuites`, and `admin_tls_ciphersuites`. See the descriptions of these variables for their permitted values.

  You should be aware that `libmysqlclient` is not affected in this change, and continues to support ciphers that do not satisfy its conditions so that it can continue to connect to previous versions of MySQL. (WL #15801)

- The use of non-unique or partial keys as foreign keys is deprecated in MySQL. Beginning with this release, you must explicitly enable such nonstandard keys in one of the ways listed here:

  - Set `restrict_fk_on_non_standard_key` (added in this release) to `OFF`.
  - Start the server with the `--skip-restrict-fk-on-non-standard-key` option (also new in this release).

  The `restrict_fk_on_non_standard_key` server system variable is `ON` by default. This means that any attempt to use a nonstandard key as a foreign key in a `CREATE TABLE` or `ALTER TABLE` statement is rejected with the error `ER_FK_NO_INDEX_PARENT`; setting it to `ON` allows such statements to run, but they raise `ER_WARN_DEPRECATED_NON_STANDARD_KEY` as a warning.

  Upgrades to MySQL 8.4 releases from MySQL 8.0 are supported even if the old database contains one or more foreign keys referring to non-unique or partial keys. As part of the upgrade process, the server prints a list of warning messages with the names of those foreign keys referring to nonstandard keys.

  See the description of `restrict_fk_on_non_standard_key` for more information. (WL #15699)

  References: See also: Bug #30615520, Bug #97836.

Firewall Notes

- Following an upgrade, some MySQL Firewall stored procedures were not updated as expected. (Bug #36084930)
- Several enhancements have been made in the stored procedures provided by MySQL Enterprise Firewall. These improvements are listed here:

- Stored procedures provided by MySQL Enterprise Firewall are now transactional. When an error occurs during execution of a firewall stored procedure, an error is reported, and all changes made by the stored procedure up to that point in time are rolled back.
- Firewall stored procedures now avoid performing unnecessary combinations of `DELETE` plus `INSERT` statements, as well as those of `INSERT IGNORE` plus `UPDATE` operations, making them faster and more efficient.
- User-based stored procedures and UDFs, previously deprecated, now raise a deprecation warning, such that calling either of `sp_set_firewall_mode()` or `sp_reload_firewall_rules()` now generates such a warning. See Firewall Account Profile Stored Procedures, as well as Migrating Account Profiles to Group Profiles, for more information.

(WL #15790)

INFORMATION_SCHEMA Notes

- Fixed a potential race condition in the `PROCESSLIST` table. (Bug #35509371)

Installation Notes

- As part of the installation process, a file in JSON format named `mysql_upgrade_history` is now created in the server's data directory, or updated if it already exists. Information contained in this file includes the following items, among others:

  The day and time of the installation

  The MySQL server version installed

  The maturity level of the release (LTS or Innovation)

  The `mysql_upgrade_info` file was deprecated in MySQL 8.0.17, and is no longer used; its presence is now checked for, and if this file is found, it is removed as part of the installation process. (WL #16039)

  References: See also: Bug #95165, Bug #29702060.

Keyring Notes

- Migration from a keyring component to a keyring plugin is now supported. To perform such a migration, use the `--keyring-migration-from-component` server option introduced in this release, setting `--keyring-migration-source` to the name of the source component, and `--keyring-migration-destination` the name of the target plugin.

  See Key Migration Using a Migration Server, for more information. (WL #16017)

Optimizer Notes

- This release adds support for automatic updates of histograms. When this feature is enabled for a given histogram, the histogram is updated whenever `ANALYZE TABLE` is run on the parent table. Automatic recalculation of persistent statistics by `InnoDB` also updates the histogram when automatic updates are enabled.

  Automatic histogram updates use the same number of buckets as the histogram was originally specified with, if any.

To enable automatic histogram updates, include the `AUTO UPDATE` option (added in this release) for the `ANALYZE TABLE` statement. To disable it, include `MANUAL UPDATE` instead. `MANUAL UPDATE` (no automatic updates) is the default if neither option is specified. When upgrading to this release, existing histograms are treated as though they had been created using `MANUAL UPDATE`.

For more information, see Histogram Statistics Analysis. See also Configuring Persistent Optimizer Statistics Parameters. (Bug #36053241, WL #15786)

- The multi-range read (MRR) optimization did not perform as well as in previous releases. (Bug #113711, Bug #36220640)

- In this release, we lift the requirement for transforming a correlated scalar subquery to a derived table that an operand of the equality expression which does not contain an outer reference must be a simple column reference.

  This means that inner columns can be contained in deterministic expressions, as shown here:

  ```
  func1(.., funcN(.., inner-column-a, ..), inner-column-b) = outside-expression

  inner-column-a + inner-column-b = outside-expression
  ```

  For example, the following query is now supported for optimization:

  ```
  SELECT * FROM t1
    WHERE ( SELECT func(t2.a) FROM t2
              WHERE func(t2.a) = t1.a ) > 0;
  ```

  The inner operand cannot contain outer column references; likewise, the outer operand cannot contain inner column references. In addition, the inner operand cannot contain a subquery.

  If the transformed subquery has explicit grouping, functional dependency analysis may be excessively pessimistic, resulting in an error such as `ERROR 1055 (42000): Expression #2 of SELECT list is not in GROUP BY clause and contains nonaggregated column ...`. For `InnoDB`, the transform is disabled by default, in which case such queries pass without error (but are not transformed).

  For more information, see Correlated Subqueries. (WL #15540)

Performance Schema Notes

- User variables assigned decimal values were rounded up in the `user_variables_by_thread` table. (Bug #35781732)

Server Administration

- **Important Change:** This release adds a privilege which is specific to the use of the `FLUSH PRIVILEGES` statements. Unlike the existing `RELOAD` privilege, the new `FLUSH_PRIVILEGES` privilege applies only to `FLUSH PRIVILEGES` statements. This privilege is global in scope, and is applicable to users and roles.

  The `RELOAD` privilege continues to be supported in this capacity to provide backwards compatibility; users having this privilege can still execute `FLUSH PRIVILEGES` statements following an upgrade. As part of upgrading to a MySQL 8.4 release, a check is performed to see whether there are any users having the `FLUSH_PRIVILEGES` privilege; if there are none, users having the `RELOAD` privilege are automatically assigned the new privilege as well. (WL #16044)

- **Important Change:** This release adds a new `OPTIMIZE_LOCAL_TABLE` privilege. Users must have this privilege to execute `OPTIMIZE LOCAL TABLE` and `OPTIMIZE NO_WRITE_TO_BINLOG TABLE` statements.

  When upgrading from a previous releases, users already having the `SYSTEM_USER` privilege are automatically granted the `OPTIMIZE_LOCAL_TABLE` privilege. (WL #15819)

Thread Pool Notes

- **Important Change:** Previously, when the limit defined by `thread_pool_max_transactions_limit` was reached, new connections or transactions on existing connections sometimes appeared to hang until one or more of the existing transactions were completed. This release introduces a way to mitigate this issue in many cases by setting an upper limit `thread_pool_longrun_trx_limit` (added in this release) on the length of time during which the number of ongoing transactions is allowed to match the maximum number of thread pool transactions specified by `thread_pool_max_transactions_limit`; once this limit is reached, the upper limit on the number of transactions is suspended for the thread group.

  When the number of long-running transactions decreases appreciably, `thread_pool_max_transactions_limit` can be (and is) enforced again. See the description of the `thread_pool_longrun_trx_limit` server system variable for more information about how this is determined. (WL #16132)

- The Performance Schema `tp_connections` thread pool plugin table contained no entries for connections that were in the `admin` group. (Bug #36296830)

Functionality Added or Changed

- **Important Change; Group Replication:** MySQL 8.0 performs special handling for group members whose version is 8.0.17 or earlier. This special handling is removed in the current release.

  Users of MySQL 8.0 are encouraged to upgrade all instances to the latest 8.0 release prior to upgrading to MySQL 8.4. (Bug #36314222)

- **Important Change; Group Replication:** In-place downgrades of servers within groups are supported within the MySQL 8.4 LTS series. For example, a member of a group running MySQL 8.4.2 can be downgraded to MySQL 8.4.0.

  Similarly, cross-version group membership is also supported within the 8.4 release series. For example, a server running MySQL 8.4.0 can join a group all of whose members currently run MySQL 8.4.2, as can a server running MySQL 8.4.3.

  For more information, see Upgrading Group Replication. (Bug #35918034)

  References: See also: Bug #35397276.

- **Important Change; Group Replication:** The default values of two server system variables relating to Group Replication have changed:

  - The default value of the `group_replication_consistency` system variable is now `BEFORE_ON_PRIMARY_FAILOVER`; previously, this was `EVENTUAL`.

  - The default value of the `group_replication_exit_state_action` system variable is now `OFFLINE_MODE`; previously, this was `READ_ONLY`.

For more information, see the descriptions of the variables listed, as well as Configuring Transaction Consistency Guarantees, and Responses to Failure Detection and Network Partitioning. (WL #15712, WL #15713)

- **Important Change; Group Replication:** When issued with `group_replication_consistency` set to `BEFORE_ON_PRIMARY_FAILOVER`, the MySQL `KILL` statement now ignores any consistency guarantees, with any interrupted transactions now being rolled back.

- **Important Change:** For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated to version 3.0.13. Issues fixed in OpenSSL version 3.0.13 are described at https://openssl-library.org/news/openssl-3.0-notes/. (Bug #36261675)

- **Important Change:** Upgrading from MySQL 5.7 to MySQL 8.4 is not supported; the code and behavior was updated to reflect this. Upgrade MySQL 5.7 to 8.0 before proceeding to 8.4. (WL #15924)

- **InnoDB:** Progress messages are now logged periodically during long-running rollbacks as informational note level error messages, initially as `ER_IB_LONG_ROLLBACK_FULL` (which appends transaction information) followed by successive `ER_IB_LONG_ROLLBACK`. (WL #15822)

- **InnoDB:** Changed the default values for the following `InnoDB` configuration options: `innodb_adaptive_hash_index`, `innodb_buffer_pool_in_core_file`, `innodb_buffer_pool_instances`, `innodb_change_buffering`, `innodb_doublewrite_files`, `innodb_doublewrite_pages`, `innodb_flush_method`, `innodb_io_capacity`, `innodb_io_capacity_max`, `innodb_log_buffer_size`, `innodb_numa_interleave`, `innodb_page_cleaners`, `innodb_parallel_read_threads`, `innodb_purge_threads`, `innodb_read_io_threads`, `innodb_use_fdatasync`, `temptable_max_ram`, `temptable_max_mmap`, and `temptable_use_mmap`.

  The settings affected by the `--innodb-dedicated-server` startup option have also changed.

  For a list of the new default values, see What Is New in MySQL 8.4 since MySQL 8.0. See also Enabling Automatic InnoDB Configuration for a Dedicated MySQL Server. (WL #16179)

- **Packaging:** Added support for Fedora 40 and Ubuntu 24.04.

- **Replication:** It is now possible to recover the relay log with any incomplete transactions removed. The relay log is now sanitized when the server is started with `--relay-log-recovery=OFF` (the default). This means that, on startup, the server removes all of the following items:

  - Incomplete transactions
  - Relay log files containing incomplete transactions or parts of incomplete transactions only
  - References in the relay log index file to any relay log files removed

  For more information, see the description of the `relay_log_recovery` server system variable.

- **Group Replication:** When a member rejoining a group has transactions to apply on the `group_replication_applier` channel from previous participation in the group, those transactions are applied when the member rejoins before connections to a donor during distributed recovery.

  This backlog of transactions to apply can be monitored using the `performance_schema.replication_applier_status_by_worker` table, but there was no

information about it in the error log, which could lead to the mistaken impression that the server was stalled.

Now in such cases, one of the messages `Distributed recovery will wait until the transactions ... contained on the group_replication_applier channel are applied` or `Distributed recovery finished applying the transactions ... contained on the group_replication_applier channel` is also written to the error log, as appropriate. (Bug #36229998)

- **Group Replication:** As of this release, distributed recovery using the clone plugin is permitted between different releases in the same LTS series. (Bug #35992145)

- **Group Replication:** MySQL Group Replication now supports preemptive certification information garbage collection when running in single-primary mode. This feature can be enabled using the `group_replication_preemptive_garbage_collection` system variable added in this release; when enabled, only the write sets for those transactions that have not yet been committed are kept, which can save time and memory consumption. `group_replication_preemptive_garbage_collection_rows_threshold` (also introduced in in this release) sets a lower bound on the number of certification rows needed to trigger preemptive garbage collection when the feature is enabled; the default value is 100000.

  The value of `group_replication_preemptive_garbage_collection` can be changed only when Group Replication is not running, and has no effect on a group running in multi-primary mode. When this system variable is enabled, it is not possible to change between multi-primary mode and single-primary mode (see Changing the Group Mode). For help with obtaining information about memory consumed by the garbage collection process, see Monitoring Group Replication Memory Usage with Performance Schema Memory Instrumentation. (WL #15923)

- **Microsoft Windows:** MySQL Windows binary files (`.exe` and `.dll` files) now display additional information when their properties are viewed. (Bug #36379291)

- Clone plugin version requirements have been relaxed to allow cloning between different point releases in the same series. In other words, only the major and minor version numbers must match when previously the release number also had to match.

  For example, cloning of MySQL 8.4.0 to (a future) MySQL 8.4.14, or from MySQL 8.4.14 to 8.4.0, is now supported.

  For more information, see The Clone Plugin. (Bug #36293529, WL #15989)

- When using the iterator-based format for `EXPLAIN FORMAT=JSON` (that is, when `explain_json_format_version` is `2`), the output now contains a `query_type` field identifying the type of statement (select, insert, delete, and so on). (Bug #36134568)

Bugs Fixed

- **Important Change; Replication:** The `TRANSACTION_GTID_TAG` privilege is now required to set the `gtid_executed` server system variable. (Bug #36201133)

- **Important Change:** The Robin Hood hashing library has been replaced with `unordered_dense`. (Bug #36158022)

- **InnoDB; Microsoft Windows:** Improved redo log performance on Windows by opening redo log files in overlapped mode. (Bug #36154818)

  References: This issue is a regression of: Bug #12527.

- **InnoDB:** The log writer calls functions that temporarily release `log.writer_mutex`; when `innodb_log_writer_threads=OFF`, this potentially led to other threads writing to the redo log in between these times. (Bug #36425219)

- **InnoDB:** Some FTS operations on tables with FTS indexes led to inconsistent results. For example, if the server terminated while synchronizing the FTS cache or when synchronization occurred concurrently with another FTS operation.

  Our thanks to Yin Peng and the Tencent team for the contribution. (Bug #36347647)

- **InnoDB:** When creating an index on a table containing data, `valgrind` occasionally reported reads of uninitialized memory from `ddl::Builder::bulk_add_row()`. (Bug #36342792)

- **InnoDB:** On Windows, keeping a file open without a shared write lock and attempting to acquire the `fil_shard` mutex caused a deadlock with another thread that had acquired the `fil_shard` mutex and was attempting to access the same file. (Bug #36159317)

  References: See also: Bug #32808809.

- **InnoDB:** Fixed a potential redo log rotation issue that could emit a "Found existing redo log files, but at least one is missing" error during recovery. (Bug #36124625)

- **InnoDB:** Found and fixed an assertion failure related to full-text indexes. (Bug #35836581)

- **InnoDB:** Added a log buffer check to the fil_tablespace_redo_* functions for them to better handle corrupt redo logs. (Bug #35676721)

- **InnoDB:** Improved buffer handling during the tablespace deletion process, a situation that could have potentially caused an assertion failure. (Bug #35676106, Bug #36343647)

- **InnoDB:** The redo log would potentially not log a column order change with instant DDL, which could cause an incorrect log replay during recovery. (Bug #35183686)

- **InnoDB:** Results for `SHOW ENGINE INNODB STATUS` showed an invalid value (`NULL`) as the name of the first thread in the `File IO` section. (Bug #113819, Bug #36118112)

  References: This issue is a regression of: Bug #34992157.

- **InnoDB:** With `innodb_parallel_read_threads` set to a value greater than 1, `InnoDB` unnecessarily disabled read-ahead heuristics which resulted in stalls when pages were not already in the buffer pool. (Bug #113482, Bug #36142806)

- **InnoDB:** Importing a tablespace had a hard limit of 128 characters for the imported column names, which did not properly account for variable-length encodings. It's now set to 64 * the maximum length of a multi-byte characters.

  Our thanks to Lee Adria for the contribution. (Bug #113208, Bug #36047803)

- **InnoDB:** Running a query that used a unique hash index with the TempTable storage engine could take significantly more time compared to running the query with the MEMORY engine.

  Our thanks to xiaoyang chen for the contribution. (Bug #113178, Bug #36037224, Bug #36224958)

- **InnoDB:** The redo log consumer could not advance if capacity was full and another thread was executing USER-related operations such as `CREATE USER`. This also blocked new connections, which potentially prevented the workaround solution of increasing `innodb_redo_log_capacity` size. (Bug #112608, Bug #36004840)

- **InnoDB:** In debug builds, there was an assertion failure in InnoDB's background when a transaction it wanted to acquire an MDL lock on was no longer active.

  This fix is based on a patch from Genze Wu with Alibaba, thank you for the contribution. (Bug #112424, Bug #35835864)

  References: This issue is a regression of: Bug #33700835.

- **InnoDB:** The MySQL truncate undo operation (purge thread) did not remove the `undo_{space_number}_trunc.log` file when attempting to truncate the undo tablespace. (Bug #112262, Bug #35784192)

- **InnoDB:** With `innodb_parallel_read_threads` set to a value greater than 1, InnoDB would unnecessarily request asynchronous reads which required more synchronization during I/O completion and created a bottleneck due to the limited number of available threads (`innodb_read_io_threads`) for handling I/O operations. Now this performs synchronous instead of asynchronous reads. (Bug #112137, Bug #35740866)

- **InnoDB:** A trx would unexpectedly halt after encountering an incorrect trx->in_innodb value.

  Our thanks to Shaohua Wang for the contribution. (Bug #110652, Bug #35277407)

- **InnoDB:** Fixed performance issues related to querying the `data_lock` and `data_lock_waits` tables when thousands of read-only transactions were present. (Bug #109539, Bug #34951273)

- **InnoDB:** MySQL no longer ignores the optimizer hint to use a secondary index scan, which instead forced a clustered (parallel) index scan. (Bug #100597, Bug #112767, Bug #31791868, Bug #35952353)

- **Replication:** `diagnostics.sql` prevented upgrades to MySQL 8.4.0 from earlier versions of MySQL when restoring from data containing old replication terminology such as `SHOW SLAVE STATUS`. (Bug #36323066)

- **Replication:** `Gtid_tagged_log_event` encoded the correct value only when the original commit timestamp was equal to the immediate commit timestamp, instead of only when they were different. (Bug #36312880)

- **Replication:** In certain cases, the `gtid_next` server system variable accepted an invalid value, displayed an invalid value after setting it (even to a legal value), or both. (Bug #36308318)

- **Replication:** The replication receiver thread did not report errors when a replication channel was configured with an unknown network namespace. The receiver thread stopped when such an error occurred but no reason for the halt was shown or logged. (Bug #36054355)

- **Replication:** With `binlog_format=ROW` and `gtid_mode=OFF`, deadlocks were sometimes reported among workers contending for the auto-increment lock when applier concurrency was high on the replica. (Bug #35851009)

- **Replication:** In certain cases, the SQL thread terminated with error `MY-001755` (`ER_MTA_CANT_PARALLEL`) when executed with the parallel applier. (Bug #35431274)

- **Replication:** Failure of `XA COMMIT` of a prepared transaction could result in transaction rollback. (Bug #33650776)

- **Replication:** The replication receiver thread stopped with an error if the replication source server sent a heartbeat event containing a binary log file position that was above the 4GB offset, due to the large size of the binary log file. A new heartbeat event (Heartbeat_log_event_v2, log event type 41) that handles the larger value correctly has been added for use in this situation. (Bug #29913991)

- **Replication:** When the server printed an `ER_REPLICA_HEARTBEAT_FAILURE` error message, it did not respect the length of the master log file name, leading to it print unrelated data. (Bug #29913928)

- **Group Replication:** Problems arose when members `M1` and `M2` were in a group, with `M1` using `u1` as its recovery user and `M2` using `u2` as its own recovery user, and both users `u1` and `u2` existing on `M1` and `M2` with all necessary privileges, and when a new member `M3` joined the group using `u2` as its recovery user. `M3` knew only of user `u2`, but did not know of user `u1`, leading `START GROUP REPLICATION` on `M2` to be rejected since `M1` was unable to connect to `M2`. This also generated a new `view_id` listing the group members as `M1` and `M2`, but `M1` nevertheless continued trying to connect to `M3`, with `M1` logging `Error in establishing mysql connection` and M3 logging `Access denied` errors for the connection attempts from `M1`.

  By design, XCom stores the last three known configurations, including references to physical connections shared among all past and present configurations. This is done to facilitate quick reconnections by nodes rejoining the group, explicitly or implicitly, and that were already present in any of those configurations.

  A side effect of this was that we might keep attempting to connect to a node that was currently not in the group. To solve this problem, we inhibit error logging if the node is not in the current configuration, in order to avoid false negatives which might lead a DBA or an operator to think mistakenly that there is a problem in the system. (Bug #36210988)

  References: See also: Bug #32592027.

- **Group Replication:** Improved handling of GTID sets. (Bug #36093405)

- **Group Replication:** Two cases were found in which a member exited the group and moved to the `ERROR` state, but did not honor the action specified by `group_replication_exit_state_action`; these are listed here:

  - When an error occurred while enabling `super_read_only`

  - When member join recovery was not possible, due to missing binary logs and clone groups on group members

  Example: When the value of `group_replication_exit_state_action` was `OFFLINE_MODE` and one of these events took place, offline mode was not enabled as expected. (Bug #36076308)

- **Group Replication:** After successfully setting a new primary, `group_replication_set_as_primary()` in some cases waited indefinitely for the operation to complete. (Bug #36059098)

- **Group Replication:** For errors affecting transactions with `AFTER` (`ER_GRP_RPL_TRX_WAIT_FOR_GROUP_PREPARE_FAILED`), the message that was written to the error log referenced a session ID instead of the UUID. (Bug #35953196)

- **Group Replication:** A group running group replication with a primary `i1` and two secondaries `i2` and `i3` started to have intermittent issues because of high memory usage on the primary. The secondaries began reporting the primary as unreachable then reachable again, and the primary began reporting the secondaries as intermittently reachable then reachable as well. Following a period of such instability, the secondaries expelled the original primary (`i1`) and elected a new one (`i2`).

  Under these conditions, queries against the `performance_schema.replication_group_members` table on the former primary (`i1`) reported `i1` as `ONLINE` and `PRIMARY`, `i2` as `ONLINE` and `SECONDARY`,

and `i3` as `ONLINE` and `SECONDARY` for an extended period of time (12 hours or more) until the `mysqld` process was restarted on `i1`.

The problems observed were found to have begun on the original primary (`i1`) when one of the secondaries was overloaded and began intermittently leaving and joining the group, its connections being dropped and recreated repeatedly on the primary server. During the reconnection process, the primary hung when trying to create the connection, thus blocking the single XCom thread. This was traced to the invocation of `SSL_connect()` on the XCom communication stack, which changed in MySQL 8.0.27 from asynchronous to synchronous form. When a node was overloaded, it might not respond to the `SSL_connect()` call, leaving the connecting end blocked indefinitely.

To fix this, we now connect in a way that is non-blocking, and that returns in case of a timeout, leaving the retry attempts to the caller—in this specific case, the XCom thread when trying to reconnect to another node. (Bug #34348094, Bug #36047891)

References: See also: Bug #37587252.

- **JSON:** `JOIN` and `GROUP BY` handled some `JSON` column values differently. (Bug #101048, Bug #31969607)
- **MySQL NDB ClusterJ:** The `setLimits()` method can now be chained to `deletePersistentAll()` to limit the number of items to delete. See the description of `deletePersistentAll()` for details. (Bug #36049906)
- The `strings` and `strings_shared` library files declared but did not supply the function `mysql::collation::find_by_id()`. (Bug #36353447)
- Raised the minimum required version of CMake to build MySQL from 3.5.1 to 3.14.6. (Bug #36338366)

  References: See also: Bug #35553331.
- Configuration of the backtrace library was performed too early in the build process, and the library itself was built with an incomplete set of compiler flags, differing in both these respects from the rest of the server. (Bug #36292247)
- `SET GLOBAL offline_mode=ON` did not always perform correctly when issued under high loads. (Bug #36275182)

  References: See also: Bug #36405894.
- Upgraded `curl` to version 8.6.0. (Bug #36267545)
- Added a new error message for the case when a timeout is detected in `net_read_raw_loop()` rather than in the thread pool code. This includes information about the conditions triggering the timeout. This is an error-level message if the timeout occurs earlier than indicated by `wait_timeout`. (Bug #36250895)

  References: See also: Bug #34857147.
- `mysqldump` did not always interpret the server version correctly. (Bug #36248967)

  References: See also: Bug #36405879.
- Condition pushdown to a view was rejected with a collation mismatch if the view was created with a different character set than the character set used when querying the view. (Bug #36246859)
- Improved the SQL grammar in `sql/sql_yacc.yy` by removing four shift-reduce conflicts which were not needed. (Bug #36221823)

- Use of the deprecated `exec_program()` command has been replaced by `execute_process()` to provide compatibility with CMake 3.28.1 and later. (Bug #36220656)
- The MLE component was added to the minimal RPM build. (Bug #36210740)
- Some queries using `NULLIF()` and `EXCEPT` raised an assertion in `set_typelib()`. (Bug #36151537)

  References: See also: Bug #33045412.
- Certain queries raised an assertion in `EstimateDeleteRowsCost()`. (Bug #36130806)

  References: This issue is a regression of: Bug #35590128.
- A query of the form `SELECT 1 FROM t WHERE CAST(a AS UNSIGNED INTEGER) = 1 AND a = (SELECT 1 FROM t)` led to an assertion in `item_func.cc`. (Bug #36128964)
- When selecting two empty strings that were combined with `UNION` as in `SELECT '' AS a UNION SELECT '' AS b`, the type of the resulting data was `CHAR(0)` instead of `VARCHAR(0)`.

  We fix this by removing an exception that was made for strings of length 0. (Bug #36112585)
- Upgraded the `protobuf` library to version 25.1. (Bug #36108397)
- For building Enterprise Linux RPMs, the build scripts now point to a newer `strip` command (under `/opt/rh/gcc-toolset-12`), and they now check that the corresponding `dwz` tool is available. Previously this was only implemented for EL8. (Bug #36090069)
- We now look for `gcc-ar` and `gcc-ranlib` when building on Oracle Linux with link-time optimization. (Bug #36089900)
- Use `sa_sigaction` rather than `sh_handler` for catching fatal signals, which allows the signal handler to output more information when handling `SIGSEGV` or `SIGFPE` signals. (Bug #36082110)
- The MySQL client was unable to authenticate with mysql_native_password to old MySQL Server versions that don't support pluggable authentication, such as MySQL 5.0.15. (Bug #36066161)
- Improved the messages written to the log during a server downgrade. (Bug #36053108)
- Keyring component error logging now supplies more information than previously when the component is unable to initialize. (Bug #36037172)
- Set `_ITERATOR_DEBUG_LEVEL` to `0` when compiling debug builds on Windows using Clang. (Bug #36032501)
- When performing a rollup on an `ENUM` or `SET` column, an assertion was raised in `sql/item_sum.cc` during resolution when type information for neither of these types could be found. (Bug #36028294)

  References: See also: Bug #33045412.
- When a Common Table Expression (CTE) contained an `INTERSECT` or `EXCEPT` set operation, the second use of the same CTE in a subsequent join returned a wrong result. (Bug #36002215)
- Killing a query, while it was evaluating an uncorrelated subquery containing a hash join during optimization, led to an assert in `sql/sql_select.cc`. (Bug #35991384)
- The server sometimes terminated unexpectedly in response to a specific query. (Bug #35957627)
- A rollup query with a window function such as `COUNT()` in the select list, which was also part of an `ORDER BY`, led to an unexpected shutdown of the server. (Bug #35947358)

References: This issue is a regression of: Bug #33069747.

- Improved view and trigger definer handling by view and table DDL. (Bug #35942937)
- The server did not always return metadata to the client correctly for certain queries. (Bug #35904044, Bug #36521376)
- Found and fixed an assertion failure at `handler::ha_index_end()` in `handler.cc`. (Bug #35877600)
- For a query such as `SELECT DISTINCT t1.x, t2.x FROM t AS t1, t AS t2 WHERE t1.pk = t2.x`, where `t1.pk = t2.x` and `pk` is the primary key, there is a functional dependency `t2.x->t1.x`. This means that some candidate plans grouped on `{t2.x, t1.x}` and others on `{t1.x}`, which were both valid but yielded different row estimates for two sets of fields, since this did not take functional dependencies into account.

  Now we ensure that we perform a single calculation of the number of distinct rows, and use that number for all plan candidates. (Bug #35855573)
- When running queries against a table with a multi-value index, the server sometimes exited unexpectedly, often while executing a complex `SELECT` query which used this index. (Bug #35789759)
- Improved code in `sql/item_subselect.cc`. (Bug #35733778, Bug #35738531, Bug #35779012)
- Some aggregations of window functions were not handled correctly. (Bug #35560806)
- `CREATE USER IF EXISTS` was not always logged correctly. (Bug #35530823)
- The server did not disallow subqueries in partition expressions properly. These are invalid, and should cause a syntax error. (Bug #35476172)
- Events created within stored programs were not always handled correctly. (Bug #35395333, Bug #36402968, Bug #37918920)

  References: This issue is a regression of: Bug #17809, Bug #11745618.
- Upgraded the minimum Boost version used to 1.84.0. (Bug #35259498)
- Some `RANK() ... OVER()` queries raised an assertion in `sql/sql_executor.cc`. (Bug #35228083)
- When successive `ALTER TABLE ... ALGORITHM=COPY` statements were issued within 10 seconds of one another, the `n_rows` value became 0. (Bug #35127747)
- Removed a memory leak observed while running `authentication_kerberos` under Valgrind. (Bug #34482788, Bug #36570929)
- A query using `MAX(column)` gave different results before and after an index was added to the column. (Bug #34057432)
- Some queries that used the `LEAD()` or `LAG()` window functions on a column of type `SET` or `ENUM` hit an assertion during resolution. The same assertion was hit in some queries using the `LEAST()` or `GREATEST()` function on a `SET` or `ENUM` column. (Bug #33045412)
- When adding a `HAVING` condition to a temporary table, it is expected that all the fields in the `HAVING` condition are already replaced with the temporary table fields, but for a query which had an expression involving the internal `Item_row` type in the `HAVING` clause, constant expressions were not getting cached, so that the `HAVING` clause still held references to the fields from the underlying tables. (Bug #30112096)

- In queries that materialized rows in a temporary table before performing hash join or streaming aggregation, data was sometimes copied twice from the temporary table to the join buffer or aggregation buffer. While this did not cause any wrong results, it led to inefficient use of buffer space with a possible negative impact on performance.

  This was due to the internal `WalkTablesUnderAccessPath()` function visiting tables in `MATERIALIZE` access paths twice: first when it saw the `MATERIALIZE` access path itself, and then again when it visited the `table_path` member of the MATERIALIZE access path.

  We fix this by not visiting the table when seeing the `MATERIALIZE` path, and doing so only when seeing the `table_path` below `MATERIALIZE`. (Bug #113647, Bug #36190386)

- Updated the URL used for downloading the Boost C++ libraries. (Bug #113576, Bug #36164514)

- In the debug server, an intersection comparing columns of different types sometimes triggered an assert in `sql/item.cc`. (Bug #113385, Bug #36094867)

- A transform could be semantically invalid when the selected item in the subquery tested for `NULL`; the left outer join with a grouped derived table might in such cases produce `NULL` while the original subquery might not. To prevent this from happening, we now bar such subqueries from being transformed. (Bug #113318, Bug #36070542)

- The fix for a previous issue, first addressed in MySQL 8.0.30, was incomplete.

  Our thanks to Hao Lu for the contribution. (Bug #113174, Bug #36035044)

  References: This issue is a regression of: Bug #110801, Bug #35328028.

- On s390x, we now compile the FMA test with `-O2` to avoid overoptimization.

  Our thanks to Jonathan Albrecht for the contribution. (Bug #113096, Bug #36016140)

- Although s390x is a big-endian platform, the little-endian ICU data directory was used for compiling.

  Our thanks to Jonathan Albrecht for the contribution. (Bug #113095, Bug #36016141)

- `SET SESSION optimizer_switch = 'hash_set_operations=off'` after preparing a statement led to an assertion in `sql/sql_select.cc` when trying to execute the same prepared statement. (Bug #112919, Bug #35970620)

- The server now uses `ER_NO_REFERENCED_ROW_2` or `ER_ROW_IS_REFERENCED_2` for foreign key errors whether error details are displayed, or not. In addition, we now display parent and child table details in error messages when the user has the proper grants. (Bug #112589, Bug #35868410)

- Incorrect results were sometimes obtained from a query that used a group by loose index scan. (Bug #112541, Bug #35854362)

- An assertion in `sql/sql_derived.cc` that checked whether a referenced item in an `Item_ref` object had consistent outer reference information failed when the reference was of type `OUTER_REF`. For objects of type `Item_outer_ref`, dependency information was set for the `Item_outer_ref` object and the original expression that this reference points to, but an intermediate reference object between the `Item_outer_ref` and the original expression did not contain this information. (Bug #112478, Bug #35846847)

- An assertion failed in debug builds when inserting data with a zero-length column, such as `CHAR(0)` or `BINARY(0)`, into a table. Now, a less strict assertion more accurately fails only if it detects that a non-zero number of bytes copied from a source is identical to the target. (Bug #111450, Bug #35507763)

- MySQL did not build correctly using the `musl` version of `libc`.

  Our thanks to Sam James for the contribution. (Bug #110808, Bug #35330950)

- Using a default string histogram on a `TEXT` column raised an assertion due to a collation mismatch when comparing histograms bucket values with the string returned by `REVERSE(1)`. (Bug #110527, Bug #35227319)

- A `VALUES` statement in a correlated lateral or (other) dependent subquery yielded an incorrect result. (Bug #109252, Bug #110076, Bug #34852090, Bug #35087820)

Changes in MySQL 8.3.0 (2024-01-16)

- Audit Log Notes
- Authentication Notes
- Compilation Notes
- Component Notes
- Configuration Notes
- Deprecation and Removal Notes
- Replication with GTIDs
- INFORMATION_SCHEMA Notes
- MySQL Enterprise Notes
- Optimizer Notes
- Packaging Notes
- Performance Schema Notes
- Pluggable Authentication
- Thread Pool Notes
- Functionality Added or Changed
- Bugs Fixed

Audit Log Notes

- In some cases, calling `audit_log_read( audit_log_read_bookmark() )` led to an `Out of memory` error. (Bug #35957453)

Authentication Notes

- **Microsoft Windows:** The server-side `authentication_ldap_sasl` plugin which performs SASL-based LDAP authentication is now supported on Windows platforms. This means that Windows client programs can now use GSSAPI with Kerberos for authentication using the `authentication_ldap_sasl_client` plugin.

  See SASL-Based LDAP Authentication (Without Proxying), for more information. (WL #14056)

Compilation Notes

- **Microsoft Windows:** MySQL did not compile correctly using Visual Studio 2022. (Bug #35967676)
- Improved the `-DWITH_ZLIB=system` check. (Bug #35968195)
- The minimum version of Clang required for compiling MySQL was raised from Clang 10 to Clang 12. (Bug #35868054)
- On macOS, the OpenTelemetry component would not compile with the system `protobuf` library and instead required the bundled version by using the `-DWITH_PROTOBUF=bundled` CMake option. (Bug #35821812)
- Removed compiler warnings relating to code signatures when using XCode 14 or later. (Bug #35818055)
- On Windows, enabled the `__cplusplus` macro when compiling with Visual Studio. (Bug #35808500)
- MySQL now bundles the Boost C++ libraries and has removed support for using local or external sources. Now only the bundled Boost libraries are used when compiling MySQL.

  This change also removes the CMake build options `WITH_BOOST`, `DOWNLOAD_BOOST`, and `DOWNLOAD_BOOST_TIMEOUT`. (Bug #35805629)
- Removed references to unsupported platforms, such as SUSE 12, EL6, and Ubuntu 16/18.

  Additional changes: added `WITH_ZLIB` to the `WITH_SYSTEM_LIBS` CMake option, which was removed from it in MySQL 8.0.30, and set `WITH_FIDO` to `none` if all authentication plugins are disabled. (Bug #35795161)

  References: See also: Bug #34015600.
- Changed the default standard C++ language version needed to compile MySQL from C++17 to C++20. (Bug #35781622)
- Added alternate OpenSSL system package support on EL8 by passing `openssl3-fips` to `-DWITH_SSL` CMake option, which already supported `openssl3`. Authentication plugins such as LDAP and Kerberos are disabled since they do not support these alternative versions of OpenSSL. (Bug #35638881)
- On Linux, added `mold` (version 2 and higher) linker support. This adds a new `-DWITH_LD=mold|lld` option; by default, this is empty, which causes the standard linker to be used. This option is not supported on Enterprise Linux, which must use the `ld` linker.

  This change also removes the `-DUSE_LD_LLD` CMake option in favor of passing in `lld` to the new option. (Bug #34099162)
- Enabled "-Wdocumentation" in MYSQL_MAINTAINER_MODE when building with Clang, and corrected the inaccurate documentation comments it revealed. (Bug #31037561)
- Improved the cycle timer for the s390x architecture.

  Our thanks to Jonathan Albrecht for the contribution. (Bug #112845, Bug #35949958)
- For compiling on Linux, changed the `no-error=deprecated-declarations` flag to `no-deprecated-declarations` for the OpenSSL 3 library.

  Our thanks to karry zhang for the contribution. (Bug #112209, Bug #35755328)
- The server did not build correctly on Ubuntu 23.10.

Component Notes

- The MySQL Enterprise Data Masking and De-Identification component now includes the ability to flush the data on the secondary or replica into memory. This can be done in either of the ways described here:

  - A flush can be performed manually, using the `masking_dictionaries_flush()` function which is added in this release.

  - The component can be configured to flush the memory periodically, leveraging the Scheduler component, by setting the new `component_masking.dictionaries_flush_interval_seconds` system variable to an appropriate value.

  For more information, see the descriptions of these items as well as MySQL Enterprise Data Masking and De-Identification. (WL #15738)

- MySQL Enterprise Edition now supports collecting server metric data in the OpenTelemetry format using the `component_telemetry` component. This data is then forwarded to a configurable endpoint where it can be used by any OpenTelemetry-compatible system.

  

  Note

  Telemetry metrics are now supported by MySQL Enterprise Edition on Linux, Windows, and MacOS platforms. In previous versions Telemetry was only supported on Linux.

  See Telemetry. (WL #15200)

Configuration Notes

- **Microsoft Windows:** On Windows, the *MySQL Configurator* option to 'Update the Start menu link' did not function. (Bug #35848493)

- **Microsoft Windows:** On Windows, *MySQL Configurator* can now perform in-place upgrades. For example, MySQL Server 8.3.0 can replace an existing MySQL Server 8.2.0 installation that includes updating and renaming the data directory. (WL #15882)

Deprecation and Removal Notes

- **InnoDB:** The `--innodb` and `--skip-innodb` server options were deprecated in MySQL 5.6 and have had no effect since that time. These options have now been removed.

  

  Note

  The `InnoDB` storage engine is always enabled, and it is not possible to disable it.

  (WL #10909)

- **InnoDB:** Removed the `InnoDB` memcached plugin, which was deprecated in MySQL 8.0.22. (WL #15116)

- **Replication:** A number of server options and variables relating to MySQL Replication which were deprecated in previous versions of MySQL, have been removed from MySQL 8.4. Attempting to use any of these now causes the server to raise a syntax error. The effected options and variables are listed here:

- `--slave-rows-search-algorithms`: The algorithm used by the replication applier to look up table rows when applying updates or deletes is now always `HASH_SCAN,INDEX_SCAN`, and is no longer configurable by the user.
- `log_bin_use_v1_events`: This allowed source servers running MySQL 5.7 and newer to replicate to earlier versions of MySQL which are no longer supported or maintained.
- `--relay-log-info-file`, `--relay-log-info-repository`, `--master-info-file`, `--master-info-repository`: The use of files for the applier metadata repository and the connection metadata repository has been superseded by crash-safe tables, and is no longer supported. See Replication Metadata Repositories.
- `transaction_write_set_extraction`
- `group_replication_ip_whitelist`: Use `group_replication_ip_allowlist` instead.
- `group_replication_primary_member`: No longer needed; check the `MEMBER_ROLE` column of the Performance Schema `replication_group_members` table instead.

(WL #11007, WL #12899, WL #12927, WL #13161, WL #13959, WL #13951, WL #15861)

- **Replication:** When global transaction identifiers (GTIDs) are used for replication, transactions that have already been applied are automatically ignored, which means that `IGNORE_SERVER_IDS` is not compatible with GTID mode; the use of this option together with GTID-based replication was deprecated in MySQL 8.0, and is no longer allowed in MySQL 8.3. If `gtid_mode` is `ON`, `CHANGE REPLICATION SOURCE TO` with a non-empty `IGNORE_SERVER_IDS` list is now rejected with an error. Likewise, if any existing replication channel was created with a list of server IDs to be ignored, `SET gtid_mode=ON` is also rejected. Before starting GTID-based replication, check for and clear any ignored server ID lists on the servers involved; you can do this by checking the output from `SHOW REPLICA STATUS`. In such cases, you can clear the list by issuing `CHANGE REPLICATION SOURCE TO` with an empty list of server IDs, like this:

  ```
  CHANGE REPLICATION SOURCE TO IGNORE_SERVER_IDS = ();
  ```

  For more information, see CHANGE REPLICATION SOURCE TO Statement, and Restrictions on Replication with GTIDs. (WL #11005)

- **Group Replication:** Group Replication recovery no longer uses `View_change_log_event` for marking changes in group membership in the binary log; instead, when all members of a group are MySQL version 8.3.0 or later, members share compressed recovery metadata, and no such event is logged when a new member joins the group. The recovery metadata includes the GCS view ID, `GTID_SET` of certified transactions, and certification information.

  As part of this work, the `group_replication_view_change_uuid` system variable is now deprecated, and you should expect its removal in a future version of MySQL. No replacement or alternative for this variable is planned. (WL #14595)

- **API:** The following deprecated C API functions have been removed:

  - `mysql_kill()`: Use `mysql_real_query()` or `mysql_query()` to execute a `KILL` statement instead.

  - `mysql_list_fields()`: Use `mysql_real_query()` or `mysql_query()` to execute `SELECT * FROM table LIMIT 0` instead. (This is roughly equivalent to executing the statement in the `mysql` client after starting it with the `--column-type-info` option.)

    The `char *def` member the `MYSQL_FIELD` structure was used by this function only, and has also been removed.

  - `mysql_list_processes()`: Use `mysql_real_query()` or `mysql_query()` to execute a `SHOW PROCESSLIST` statement instead.

  - `mysql_refresh()`: Use `mysql_real_query()` or `mysql_query()` to execute a `FLUSH` statement instead.

    Due to the removal of this function, the `mysqladmin flush-threads` command is also removed.

  - `mysql_reload()`: Use `mysql_real_query()` or `mysql_query()` to execute a `FLUSH PRIVILEGES` statement instead.

  - `mysql_shutdown()`: Use `mysql_real_query()` or `mysql_query()` to execute a `SHUTDOWN` statement instead.

  - `mysql_ssl_set()`: Use `mysql_options()` with the appropriate options to specify certificate and key files, encryption ciphers, and so on.

  `mysql_shutdown()` was deprecated in MySQL 8.0; `mysql_set_ssl()` was deprecated in MySQL 8.0.35 and MySQL 8.2.0; the remainder of the functions listed were deprecated in MySQL 5.7.11.

  Due to these changes, the MySQL C API library version is raised from 22.1 to 23.0. (Bug #36029117, WL #10911, WL #11092, WL #13448)

- The compiler now emits a deprecation warning for the `mysql_stmt_bind_param()` C API function, which was deprecated in MySQL 8.2.0 and superseded by `mysql_stmt_bind_named_param()`. (Bug #35819974)

- The `--character-set-client-handshake` and `--old-style-user-limits` server options were formerly used for compatibility with very old versions of MySQL which are no longer supported or maintained. Since they no longer serve any useful purpose, both options have been removed. (WL #13221, WL #13229)

- The `FLUSH HOSTS` statement, deprecated in MySQL 8.0.23, has been removed. To clear the host cache, truncate the Performance Schema `host_cache` table or use `mysqladmin flush-hosts` instead. (WL #14330)

Replication with GTIDs

- This release extends the format of global transaction identifiers (GTIDs) used in MySQL Replication and Group Replication with tags to enable identification of groups of transactions. This enhancement makes it possible to assign a unique name to the GTIDs of a specific group of transactions. For example, transactions containing data operations can easily be distinguished from those arising from administrative operations simply by comparing their GTIDs.

The new GTID format is `UUID:<TAG>:NUMBER`, in which <TAG> is an arbitrary string up to 8 characters long. This is enabled by setting the value of the `gtid_next` system variable to `AUTOMATIC:<TAG>`. This tag persists for all transactions originating in the current session (unless changed using `SET gtid_next`), and is applied at commit time for such transactions, or, when using Group Replication, at certification time. It is also possible to set `gtid_next` to `<UUID>:<TAG>:NUMBER` to set the UUID of a single transaction to an arbitrary value, along with assigning it a custom tag. In both cases, the user is responsible for making sure that the tag is unique to a given replication topology.

The original `UUID:NUMBER` format for GTIDs continues to be supported unchanged, as implemented in previous versions of MySQL; changes to existing replication setups using GTIDs are not required.

Setting `gtid_next` to `AUTOMATIC:<TAG>` or `<UUID>:<TAG>:NUMBER` requires a new `TRANSACTION_GTID_TAG` privilege which is added in this release; this is true both on the originating server as well as for the `PRIVILEGE_CHECKS_APPLIER` for the replica applier thread. This also means that an administrator can now restrict the use of `SET @gtid_next=AUTOMATIC:<TAG>` or `SET @gtid_next=<UUID>:<TAG>:NUMBER` to a desired set of MySQL users or roles so that that only those users related to a given data or operational domain can commit new transactions with assigned tags.

Interaction with upgrades: When upgrading from a previous version of MySQL to MySQL 8.4, the `TRANSACTION_GTID_TAG` privilege is granted automatically to any user accounts or roles which already have the `BINLOG_ADMIN` privilege.

The built-in functions `GTID_SUBSET()`, `GTID_SUBTRACT()`, and `WAIT_FOR_EXECUTED_GTID_SET()` are compatible with tagged GTIDs.

For more information, see Changing GTID Mode on Online Servers. See also the descriptions of the `gtid_next` system variable and the `TRANSACTION_GTID_TAG` privilege. (WL #15294)

INFORMATION_SCHEMA Notes

- When running `OPTIMIZE TABLE` on a table with a `FULLTEXT` index, the index was not fully optimized at the expected time, and thus an extra pass was required. (Bug #35502793)
- This release implements the following two status variables for tracking the use of the deprecated `INFORMATION_SCHEMA.PROCESSLIST` table:
  - `Deprecated_use_i_s_processlist_count` provides a count of the number of references to the `PROCESSLIST` table in queries since the server was last started.
  - `Deprecated_use_i_s_processlist_last_timestamp` stores the time the `PROCESSLIST` table was last accessed. This is a timestamp value (number of microseconds since the Unix Epoch).

  (WL #16085)

MySQL Enterprise Notes

- Data-masking components now support specifying a dedicated schema to store the related internal `masking_dictionaries` table. Previously, the `mysql` system schema provided the only storage option. The new `component_masking.masking_database` read-only variable enables setting and persisting an alternative schema name at server startup. (WL #15733)

Optimizer Notes

- The hashing algorithm employed yielded poor performance when using a `HASH` field to check for uniqueness. (Bug #109548, Bug #34959356)

Packaging Notes

- **Important Change:** The GnuPG build key (`A8D3785C`) used to sign MySQL downloadable packages has been updated. The previous GnuPG build key (`3A79BD29`) expired on 2023-12-14. For information about verifying the integrity and authenticity of MySQL downloadable packages using GnuPG signature checking, or to obtain a copy of our public GnuPG build key, see Signature Checking Using GnuPG.

  Due to the GnuPG key update, systems configured to use `repo.mysql.com` may report a signature verification error when upgrading to MySQL 8.0.36 and higher or to MySQL 8.3.0 and higher using `apt` or `yum`. Use one of the following methods to resolve this issue:

  1. Manually reinstall the MySQL APT or YUM repository setup package from https://dev.mysql.com/downloads/.
  2. Download the MySQL GnuPG public key and add it your system GPG keyring.

Performance Schema Notes

- **Group Replication:** The `memory/group_rpl/Gcs_message_data::m_buffer` key in the Performance Schema `setup_instruments` table sometimes reported negatives values when running in single-primary mode. (Bug #35940509)
- Executing `SELECT * from performance_schema.data_locks` on a server under heavy load could cause MySQL to consume too much memory and close unexpectedly.

  As of this release, memory used executing such a query is now instrumented with `memory/performance_schema/data_container`, enabling you to observe memory consumption. (Bug #35240825)
- Executing `SELECT * from performance_schema.data_locks` on a server under heavy load could cause a deadlock in InnoDB. (Bug #35068461)

  References: See also: Bug #35240825.
- When executing a stored program, the Performance Schema instrumentation caused some unnecessary overhead.

  As of this release, all stored procedure micro instructions (`statement/sp/%`), except `statement/sp/stmt`, are disabled by default. (Bug #27934653)
- The performance of the Performance Schema statement instrumentation has been improved. Specifically, collecting `MESSAGE_TEXT` data is now more efficient. (Bug #112621, Bug #35916912)
- Under certain circumstances, under load, the Performance Schema could return the same row twice, despite having a unique key defined. This issue could occur for query digests, user names, host names, and account names (user@host). (Bug #110555, Bug #35239372)
- Removed a potential issue with `performance_schema.table_handles` found through code analysis, but never reported by users. (Bug #108501, Bug #34606682)

Pluggable Authentication

- Beginning with this release, the behavior of the `AUTHENTICATION_PAM_LOG` environment variable used in debugging the PAM authentication plugin is changed as follows:
  - Setting `AUTHENTICATION_PAM_LOG` to an arbitrary value (except as noted in the next item) no longer includes passwords in its diagnostic messages.

- To include passwords in the diagnostic messages, set `AUTHENTICATION_PAM_LOG=PAM_LOG_WITH_SECRET_INFO`.

For more information, see PAM Authentication Debugging. (Bug #74313, Bug #20042010)

Thread Pool Notes

- Added additional thread pool connection information to the MySQL Performance Schema, as follows:
  - Added a `tp_connections` table, with information about each thread pool connection.
  - Added the following columns to the `tp_thread_state` table: `TIME_OF_ATTACH`, `MARKED_STALLED`, `STATE`, `EVENT_COUNT`, `ACCUMULATED_EVENT_TIME`, `EXEC_COUNT`, and `ACCUMULATED_EXEC_TIME`
  - Added the following columns to the `tp_thread_group_state` table: `EFFECTIVE_MAX_TRANSACTIONS_LIMIT`, `NUM_QUERY_THREADS`, `TIME_OF_LAST_THREAD_CREATION`, `NUM_CONNECT_HANDLER_THREAD_IN_SLEEP`, `THREADS_BOUND_TO_TRANSACTION`, `QUERY_THREADS_COUNT`, and `TIME_OF_EARLIEST_CON_EXPIRE`.

  For more information about these tables, see Performance Schema Thread Pool Tables. For information about the Thread Pool plugin, see MySQL Enterprise Thread Pool. (WL #15515)

Functionality Added or Changed

- **Important Change; Replication:** The `binlog_transaction_dependency_tracking` server system variable was deprecated in MySQL 8.2.0. In preparation for the eventual removal of this variable, its default value is now `WRITESET`. There are no plans to replace this variable or its functionality, which is expected later to be made internal to the server. (WL #15861)
- **Important Change:** For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated to version 3.0.12. Issues fixed in OpenSSL version 3.0.12 are described at https://openssl-library.org/news/openssl-3.0-notes/. (Bug #36033684)
- **Performance; NDB Cluster:** The `NDB` binary log injector uses arena allocation. In some cases, when handling schema changes and keeping track of current state of the binary log, the arena for this was forced by changing thread local pointers, thus attempting to try and catch all allocations performed during epoch processing. At the end of the epoch, those pointers were reset, arena memory was released, and the arena structures discarded; this released the memory, but also required setting it up again for the next epoch. The thread local pointer changes also introduced the risk of memory being allocated wrongly when activating functionality in different subsystems. In this release, we make the following improvements to this functionality:
  - Changes to thread local pointers are removed, and replaced by explicit arguments to provide the arena used for allocation during the epoch.
  - Reuse of the arena for next epoch, thus avoiding the need to set it up again.

  These changes are internal only but should provide a noticeable improvememnt by saving on memory release and re-allocation over successive epochs. (WL #15002)
- **Group Replication:** `group_replication_set_as_primary()` now waits for DDL and DCL statements to complete before a new primary is elected. This includes these statements: `ALTER DATABASE`, `ALTER FUNCTION`, `ALTER INSTANCE`, `ALTER PROCEDURE`, `ALTER SERVER`, `ALTER`

TABLESPACE, ALTER USER, ALTER VIEW, CREATE DATABASE, CREATE FUNCTION, CREATE PROCEDURE, CREATE ROLE, CREATE SERVER, CREATE SPATIAL REFERENCE SYSTEM, CREATE TABLESPACE, CREATE TRIGGER, CREATE USER, CREATE VIEW, DROP DATABASE, DROP FUNCTION, DROP PROCEDURE, DROP ROLE, DROP SERVER, DROP SPATIAL REFERENCE SYSTEM, DROP TABLESPACE, DROP TRIGGER, DROP USER, DROP VIEW, GRANT, RENAME TABLE, and REVOKE.

These are in addition to those statements added in MySQL 8.1 or otherwise already supported in this regard. For more information, including a listing of all such statements supported in MySQL 8.3, see the description of the `group_replication_set_as_primary()` function, in the MySQL 8.4 Manual. (WL #15752)

- Replaced some of the bit functions in `include/my_bit.h` and `sql/join_optimizer/bit_utils.h` with standard library functions such as `std::popcount()` and `std::has_single_bit()` that were added in C++20. Functions replaced include `is_single_bit()`, `my_clear_highest_bit()`, `my_bit_log2()`, `my_round_up_to_next_power()`, `my_bit()`, and `my_count_bits()`, among others. (Bug #35813111)

- Added the `explain_json_format_version` system variable to determine the version of the JSON output format used by `EXPLAIN FORMAT=JSON` statements. `1` (the default) causes Version 1 to be used; this is the same JSON output format used for all such statements in previous versions of MySQL. Setting this variable to `2` causes the server to use the Version 2 format, also introduced in this release, for `EXPLAIN FORMAT=JSON` output. The Version 2 format is based on access paths, and is intended to provide better compatibility with future versions of the MySQL Optimizer.

  For more information and examples, see Obtaining Execution Plan Information. (WL #15684)

Bugs Fixed

- **Important Change; Replication:** When `binlog_transaction_dependency_tracking` (deprecated) was set to `WRITESET` or `WRITESET_SESSION`, while `binlog_format` was `MIXED`, dependencies were not tracked for statement parts of a transaction, and were not properly calculated, leading to collisions betweens transactions on MTA replicas.

  Because using the writeset information for conflict detection in this way can lead to false negatives, we now limit the usage of writesets for conflict checks to row-based logging only (`binlog_format=ROW`). (Bug #35931702)

- **InnoDB:** The InnoDB table statistics were not updated by bulk loaded statements. (Bug #35889669)

- **InnoDB:** LHS and RHS comparison values were added to their associated error handlers. (Bug #35814266)

- **InnoDB:** Improved recv_apply_hashed_log_recs() to better handle the DB_CORRUPT error code. (Bug #35595808)

- **InnoDB:** The hash function used by the adaptive hash index (AHI) was improved to increase performance. (Bug #35449386)

- **InnoDB:** If change buffer entries are present during startup, a disabled `innodb_validate_tablespace_paths` option will no longer be enforced and instead the MySQL server will proceed to validate all tablespaces. Otherwise, secondary indexes could end up corrupted. (Bug #35208990)

- **InnoDB:** During concurrent DDL and DML operations, DDL could fail if the online log grew too large. Buffer handling was improved to prevent this issue. (Bug #35115601)

- **InnoDB:** When clearing an AHI index from all buffer pool pages, the block state would potentially change to BUF_BLOCK_MEMORY before acquiring the block mutex thus causing an unexpected halt. (Bug #35037114)
- **InnoDB:** The `Innodb_row_lock_current_waits` variable could show a non-zero value when there were no row locks.

  Our thanks to Bin Wang for the contribution. (Bug #112532, Bug #35849707)
- **InnoDB:** Removed redundant code related to handling FIL_PAGE_LSN during recovery.

  Our thanks to Alexi Xing for the contribution. (Bug #108731, Bug #34687854)
- **Replication:** It was possible in some cases when an error occurred while applying an event for the error reported to apply to the previous event rather than to the current one. (Bug #35945223)
- **Replication:** Enabling `binlog_transaction_compression` on the source suppressed `ER_RPL_MTA_STATISTICS` messages on the replica. (Bug #35923771)
- **Replication:** An issue with calculating the current number of bytes used for `Log_event` events in Performance Schema memory instrumentation made it appear as though the `sql/replica_sql` thread on the replica grew endlessly and never decreased in size. (Bug #35546877)
- **Replication:** Stopping replication while replicating `CREATE TABLE AS SELECT` caused the server to exit. (Bug #33934013)
- **Group Replication:** Following a major outage, when all instances were reachable, rebooting the cluster hung and failed rejoining instances.

  This happened in such cases because the old secondaries retained old credentials to connect to the nodes, leading to a partial connection, in which, when a second server tried to rejoin the group, it was able to contact the primary and send an `add_node()` command to it, but could not be contacted in turn by the primary. The solution, is to validate that, when receiving an `add_node()`, we have established bi-directional communication with the new node. (Bug #35763950)

  References: See also: Bug #34898318.
- **Group Replication:** Removed a potential memory leak in `xcom_tcp_server_startup()`. (Bug #35594709)
- **Group Replication:** The monitor I/O thread for a managed replication channel (channel with asynchronous connection failover mechanism enabled) might fail for any of several reasons for which the error log entry lacked any root cause and reported only the following: `[Warning] [MY-013684] [Repl] The Monitor IO thread failed to detect if the source belongs to the group majority on the source (host:127.0.0.1 port:3310 network_namespace:) for channel 'read_replica_replication'`. (Bug #35082638)
- **Group Replication:** A lock instituted by `Transaction_monitor_thread::start()` in `plugin/group_replication/src/group_actions/group_actions_transaction_controller.cc` was never explicitly released. (Bug #35064211)
- **Group Replication:** The timeout set by `group_replication_set_as_primary()` did not terminate DDL statements. (Bug #35042876)
- **Group Replication:** A forced `START GROUP_REPLICATION` while a replication channel was in an error state could lead to an unplanned server exit. (Bug #34724344)
- **Group Replication:** Removed a possible memory leak in `plugin/group_replication/src/certifier.cc`. (Bug #110518, Bug #35226747)

- **macOS:** The `secure_file_priv` system variable did not always work correctly on Mac OS. (Bug #30274493)
- **Microsoft Windows:** Improved exception handing to ensure that the output is correct in debug mode. (Bug #35944853)
- **JSON:** Added missing checks for error handling to `NULLIF()`, `COALESCE()`, and the shift (`>>`) operator. (Bug #113668, Bug #35513196, Bug #36198403)

  References: See also: Bug #31358416.
- **JSON:** `JOIN` and `GROUP BY` handled some `JSON` column values differently. (Bug #101048, Bug #31969607)
- For building Enterprise Linux 8 RPMs, the build scripts now point to a newer `strip` command (under `/opt/rh/gcc-toolset-12`), and they now check that the corresponding `dwz` tool is available. (Bug #36086236)
- `BIO_get_mem_data()` used an incorrect argument type.

  Our thanks to Samuel Chiang for the contribution. (Bug #35950004)
- Removed an assertion failure at `CreateIteratorFromAccessPath()` in `access_path.cc`. (Bug #35944739)

  References: This issue is a regression of: Bug #32788576.
- During shutdown, `InnoDB` purging of threads can sometimes (depending on the volume of data) take a considerable amount of time to complete its work; this could leave users with the impression that the shutdown process had hung. This fix adds adds an externally visible `systemd` notification indicating that this step is in progress. (Bug #35902058)
- In some cases, calling a loadable function installed by an improperly initialized plugin caused an unplanned shutdown. (Bug #35889261)
- When the `MYSQL_FIREWALL` plugin failed to initialize properly during server startup, this led to errors in the server log and sometimes unplanned termination of `mysqld`. (Bug #35853298)
- Some nested queries with `GROUP BY` were not handled correctly. (Bug #35846402, Bug #35945822)

  References: This issue is a regression of: Bug #32918400.
- The access path for `dbug_dump` in debug mode used the wrong value.

  Our thanks to Tianfeng Li for the contribution. (Bug #35835858)
- Equality propagation replaced a non-nullable (primary key) column inside a `CAST()` with a nullable column without changing the nullability of the `CAST` expression. (Bug #35829972)

  References: See also: Bug #34898903.
- MySQL could not be built using `-DPROTOBUF_BUILD_SHARED_LIBS=OFF`. (Bug #35827217)
- `mysqld -D` printed some of its error logging information to `stdout`. (Bug #35810857, Bug #35830459)
- Removed an assert in `HashJoinIterator::Init()`, in `hash_join_iterator.cc`. (Bug #35789589)
- Removed an assertion seen when a table value constructor with an `ORDER BY` clause was used as an `IN` or `EXISTS` subquery. (Bug #35785452)

- In limited cases, passing data to the `MD5()` encryption function could halt the server. (Bug #35764496)
- Some subselects from views were not always handled correctly. (Bug #35738548)
- While performing an operation such as the bulk renaming of many tables, simultaneously executing a data definition statement similar to `CREATE TABLE ... SELECT` could stop the server unexpectedly. (Bug #35735937)
- Improved handler state resource allocation. (Bug #35713711)
- `UPDATE HISTOGRAM` did not behave as expected in all cases.

  UPDATE HISTOGRAM did not behave as expected in all cases. (Bug #35710404)
- `EXPLAIN ANALYZE` did not always produce the expected result. (Bug #35710383)
- An error occurred during subquery resolution. (Bug #35710373)

  References: This issue is a regression of: Bug #35184353.
- Some row subqueries were not always handled correctly. (Bug #35710218)
- Refreshing of used table information is now postponed to the start of the next execution, just after tables have been opened, when we know that all table objects are in a proper state. (Bug #35710213, Bug #36030073)
- Some `HAVING` queries did not produce expected results. (Bug #35710183)
- Removed an assertion in `hash_join_iterator.cc`. (Bug #35703114)

  References: This issue is a regression of: Bug #34940000.
- Removed an assert in `composite_iterators.cc`. (Bug #35686098)
- Some recursive CTEs did not function as expected. (Bug #35654240)
- The combination of `INSTALL COMPONENT` and `SET GLOBAL` with a subquery could cause the server to exit. (Bug #35647759)
- Some queries using `OVER (PARTITION ...)` were not always executed successfully. (Bug #35627798)
- Some subqueries with `ROLLUP` were not always handled correctly. (Bug #35621842, Bug #35804794)
- When performing `EXPLAIN FORMAT=TREE` for certain queries, the cost of reading the first row for an `Append` access path was lower than for the first child input access path, which should not be possible. (Bug #35590128)
- Removed an assertion in `IsBitSet()`, in `sql/join_optimizer/join_optimizer.cc`. (Bug #35590108)
- Removed the `CPACK_COMPONENT_GROUP_INFO_DISPLAY_NAME` configuration option from the Windows installation MSI interface. Now the `INFO_BIN` and `INFO_SRC` files are always installed. (Bug #35529968)
- When attempting a subquery-to-derived transformation, an `ER_FIELD_IN_GROUPING_NOT_GROUP_BY` error was not thrown, even when a field in the projection list was neither part of the `GROUP BY`, nor was functionally dependent on one. (Bug #35508108)
- Some queries using windowing functions were not always handled correctly. (Bug #35471471)

- In debug builds, a case-altered column name could cause the server to exit. (Bug #35449266)
- MySQL Server installation packages contained two copies of the `INFO_SRC` file. (Bug #35400142)
- Setting a user variable inside the argument of a window function, which in turn was evaluated using the window frame buffer, led to an assert. Setting of user variables inside expressions is already deprecated; this fix merely avoids the assert. (Bug #35390341, Bug #36008133)

  References: This issue is a regression of: Bug #32644631, Bug #32802301.
- Removed an assert in `Item_typecast_signed::val_int()`. (Bug #35049440)
- Improved the output from `my_print_stacktrace()`, which prints the stack trace when MySQL terminates unexpectedly. (Bug #34904177, Bug #36027494)
- The error message parser in `utilities/comp_err.cc` did not handle escaped apostrophe characters correctly. (Bug #34637697)
- After privileges had been granted to a role, revoking from the role a privilege which had not been granted to it explicitly appeared to succeed, and no error or warning was returned. (Bug #34063709)
- Some `SELECT DISTINCT` queries were not always handled correctly. (Bug #33725447)
- Removed an assertion failure in `sql/field.cc`. (Bug #112503, Bug #35846221)
- Removed an assertion in `sql/sql_resolver.cc`. (Bug #112498, Bug #35846192)
- Sme queries having the form `SELECT AVG(...) OVER (PARTITION BY ...)` were not always handled correctly. (Bug #112460, Bug #35710179, Bug #35845413)
- Upgrading MySQL using an official MySQL Yum or SUSE repository always enables the MySQL service. Now it enables the service only after installing, and preserves (and does not edit) the existing value while upgrading. (Bug #112382, Bug #35823558)
- `MEM_ROOT::AllocBlock()` did not satisfy the condition `minimum_length > wanted_length`, due to a mismatch between block size and `mininum_length`. A successful return (`false`) from `ForceNewBlock()` is expected to have a new block supporting `minimum_length`, but this assumption failed when `mininum_length` was larger; thus `AllocBlock()` did not conform to its contract.

  This fix is based on a contribution from Kaiwang Chen. (Bug #112304, Bug #35793743)
- The same query returned different results using the `NO_BNL` optimizer hint. (Bug #112296, Bug #35788971)
- In debug builds, using the `NTH_VALUE()` window function with an empty logical range produced an assertion error. (Bug #111562, Bug #35537311)
- Ordering a nested block where the `ORDER BY` contained a window function raised an assert in `sql/sql_executor.cc`. (Bug #111306, Bug #34856256, Bug #35471522)
- Several functions did not have a default data-type assignment for their arguments, which could affect prepared statements by raising an assertion error in debug builds. These default types now are assigned:

  - `DOUBLE` for `format_bytes()` and `format_pico_time()`
  - `BIGINT` for `ps_thread_id()`

  (Bug #110677, Bug #35287798)

- A `VALUES` statement in a correlated lateral or (other) dependent subquery yielded an incorrect result. (Bug #109252, Bug #110076, Bug #34852090, Bug #35087820)
- If the argument to a window function contained a subquery, the access path of that subquery was printed twice by `EXPLAIN FORMAT=TREE`. We fix this by ensuring that we ignore duplicate paths in such cases. (Bug #103609, Bug #32855925)
- In some cases, selecting from a view leaked a small amount of memory. (Bug #103133, Bug #32764586)
- A `WHERE` condition using the `IN()` operator with a table value constructor returned all rows. An example of an affected query would be `SELECT * FROM t1 WHERE a IN (VALUES ROW(5), ROW(55))`. (Bug #98268, Bug #30775369)
- The `--wait` command-line option did not function with the `mysql` client. (Bug #31422, Bug #11747227)

Changes in MySQL 8.2.0 (2023-10-25)

- Audit Log Notes
- Authentication Notes
- C API Notes
- Compilation Notes
- Configuration Notes
- Deprecation and Removal Notes
- Firewall Notes
- SQL Function and Operator Notes
- Parallel Event Execution (Multithreaded Replica)
- Optimizer Notes
- Packaging Notes
- Performance Schema Notes
- Security Notes
- Server Administration
- SQL Syntax Notes
- Functionality Added or Changed
- Bugs Fixed

Audit Log Notes

- Added the new `audit_log_filter_uninstall.sql` script to simplify removing MySQL Enterprise Audit. For information about this plugin, see MySQL Enterprise Audit. (Bug #35611072)

Authentication Notes

- The `mysql_native_password` plugin, which was deprecated previously, now is no longer mandatory and can be disabled at server startup. For information about this plugin, see Native Pluggable

Authentication. For general information about pluggable authentication and other available authentication plugins, see Pluggable Authentication and Authentication Plugins.

Caution

If the server-side `mysql_native_password` authentication method is disabled, older client accounts that use the client-side plugin (prior to MySQL 5.5 and possibly MySQL 5.6) are no longer able to connect and the server writes an "Access denied" message to the error log. These connections are no longer possible because, although the protocol remains compatible, the authentication method supported by the client is missing.

(WL #15667)

- MySQL Enterprise Edition now supports authentication to MySQL Server using devices such as smart cards, security keys, and biometric readers in a WebAuthn context. The new WebAuthn authentication method is based on the FIDO and FIDO2 standards. It uses a pair of plugins, `authentication_webauthn` on the server side and `authentication_webauthn_client` on the client side. The server-side WebAuthn authentication plugin is included only in MySQL Enterprise Edition distributions. For additional information, see WebAuthn Pluggable Authentication.

  The pluggable FIDO authentication method is deprecated with the introduction of WebAuthn authentication. (WL #15006)

C API Notes

- The client library now supports the use of query attributes in prepared statements with the introduction of `mysql_stmt_bind_named_param()`, a new C API prepared statement function. `mysql_stmt_bind_named_param()` replaces the now deprecated `mysql_stmt_bind_param()` function. Unlike `mysql_stmt_bind_param()`, `mysql_stmt_bind_named_param()` enables binding both unnamed and named parameters of a prepared statement. For an overview of the interface, see C API Prepared Statement Interface. (Bug #35435138, WL #15803)

- Earlier distributions, such as MySQL 5.1 and MySQL 5.0, included servers that did not advertise pluggable authentication. When a newer client using any pluggable authentication method attempted to connect to one of these older servers, the server emitted a "bad handshake" error message. This fix now ensures that client-side authentication (specifically, the `--default-auth` option) works properly for all servers, including those that do not support pluggable authentication. In addition, `mysql_native_password` now replaces `caching_sha2_password` as the default authentication method when the server does not support pluggable authentication. (Bug #90994, Bug #28082093)

- The `mysql_ssl_set()` C API function is deprecated and subject to removal in a future MySQL release. There are equivalent `mysql_options()` TLS options for all `mysql_ssl_set()` parameters. (WL #11091)

Compilation Notes

- **Microsoft Windows:** Updated code to compile with the latest MSVC 2022 version, v17.7.2. (Bug #35737379)

- Removed the MYSQL_PARSE_ARGUMENTS macro and the obsolete `support-files/compiler_warnings.sup` file. (Bug #35661316)

- Setting the `WITH_CURL` CMake option to `bundled` now uses the bundled curl distribution in `extra/curl/`. (Bug #35649213)

- Removed CMake configuration definitions related to the 32-bit architecture as 32-bit is not supported as of MySQL 8.1.0. (Bug #35621081)
- Fixed RPATH definitions in the CMake code for telemetry_client and component_telemetry so they'll build properly with debug mode enabled. (Bug #35598485)
- Increased the minimum required CMake version from 3.5.1 to 3.14.6. (Bug #35553331)
- Very long path names of `keyring_encrypted_file` source files were shortened to conform with the 256-character limit established by Windows compilers. (Bug #35493420)
- Improved ASAN for clang support on Windows. (Bug #35468711)
- Compiling on macOS looked explicitly for `openssl@1.1` but now looks for the generic `openssl` symlink instead, to allow for `openssl@3`. (Bug #35468370)
- Fixed the `WITH_DEVELOPER_ENTITLEMENTS` macOS CMake option; it did not activate due to a typo. (Bug #35374026)
- Added experimental C++20 compiler support for Linux. Usage requires GCC 10+ or Clang 11+. (Bug #35362952, Bug #35667284)
- Simplified the way OpenSSL is imported by utilizing the standard CMake OpenSSL::SSL and OpenSSL::Crypto libraries. The "system" type uses FIND_PACKAGE, alternative system types use pkg-config, and custom paths utilize FIND_PATH/FIND_FILE/FIND_LIBRARY. (Bug #35140672)
- Fixed a string concatenation warning produced when compiling with Clang 12. (Bug #111614, Bug #35549962)
- Made additional improvements to `WITH_ZLIB` functionality.

  Thanks to Nikolai Kostrigin for the contribution. (Bug #111549, Bug #35534309)
- Building with WITH_PROTOBUF=system failed with Protobuf 22 or newer due to Protobuf 22 adding the Abseil dependency. (Bug #111469, Bug #111623, Bug #35546459, Bug #35550389)
- Building with `WITH_ZLIB`="system" would break the MySQL build as it failed to find ZLIB.

  Our thanks to Meng-Hsiu Chiang and the team at Amazon for contributing to this fix. (Bug #111467, Bug #35511210)

  References: This issue is a regression of: Bug #35057542.
- On EL7 aarch64-based platforms, fixed an issue related to how fetching the CPU cache line size returned 0 that caused the MySQL server to unexpectedly halt. (Bug #110752, Bug #35479763)

  References: See also: Bug #107081, Bug #34095278.
- The C++ Standardization Committee's Library Working Group recently resolved an issue (LWG-3865 *Sorting a range of pairs*) which changes how the comparison operators are defined for `std::pair`. This fix updates the equality operator used in two files in `sql/auth` to align with this change.

  Based on a suggestion by the Microsoft Visual Studio team. (Bug #110254, Bug #35137978)

Configuration Notes

- **Microsoft Windows:** On Windows, *MySQL Configurator* no longer adds the deprecated sync-relay-log-info system variable to the generated `my.ini`. (Bug #35500581)

- **Microsoft Windows:** On Windows, *MySQL Configurator* showed "Next" instead of "Finish" on the "Apply Configuration" wizard page to finalize the configuration process. (Bug #112069, Bug #35709559)

Deprecation and Removal Notes

- **Incompatible Change:** The `WAIT_UNTIL_SQL_THREAD_AFTER_GTIDS()` SQL function, deprecated in MySQL 8.0.18, has been removed; attempting to invoke it now causes a syntax error. Use `WAIT_FOR_EXECUTED_GTID_SET()` instead. (WL #13180)

- **Replication:** The `binlog_transaction_dependency_tracking` server system variable is now deprecated, and subject to removal in a future version of MySQL. Setting or retrieving the value of this variable now triggers a warning, as does using the equivalent startup option `--binlog-transaction-dependency-tracking`. Applications depending on this variable (or option) should begin migrating away from it as soon as possible. No replacement for this variable or its functionality is planned or needed. (WL #13964)

- The `expire_logs_days` server system variable, which was deprecated in MySQL 8.0, has now been removed, and is no longer supported. This means that, beginning with this release, any attempt to get or set this variable at runtime results in an error. Trying to start the server while employing the equivalent startup option (`--expire-logs-days`) is also now rejected with an error.

  Use `binlog_expire_logs_seconds` rather than `expire_logs_days`; this variable continues to be supported, and allows you to specify expiration periods in units other than an integral number of days. (WL #11006)

- The deprecated server startup options `--abort-slave-event-count` and `--disconnect-slave-event-count`, deprecated in MySQL 8.0, have been removed in this release. Attempting to start `mysqld` with either of these options now results in an error.

  These options were formerly used in testing; no replacement for them is planned. (WL #14854)

- The `old` and `new` server system variables are now deprecated, and a warning is now issued whenever either of these variables is set or read. Expect these variables are to be removed in a future version of MySQL; applications which rely on them should migrate away from such dependencies as soon as possible. (WL #8609)

- The `--character-set-client-handshake` server option, originally intended for use with upgrades from very old versions of MySQL, is now deprecated, and a warning is issued whenever it is used. You should expect this option to be removed in a future version of MySQL; applications depending on this option should begin migration away from it as soon as possible. (WL #13220)

- Two changes in this release affect granting of privileges in MySQL. One of these changes impacts database grants; the other concerns host names used in grants. These changes are listed here:

  1. **Database-level grants.** The use of the characters `%` and `_` as wildcards in database-level grants is now deprecated, and you should expect for this wildcard functionality to be removed in a future MySQL release. The intent is for these characters always to be treated as literals, as they are already whenever the value of the `partial_revokes` server system variable is `ON`.

     For example, with `partial_revokes` set to `ON`, `GRANT SELECT ON db_.* TO user1` allows `user1` to select from any table in a database named `db_`, but does *not* allow `user1` to select from tables in databases `db1`, `db2`, `dbx`, `dby`, and so on. You should expect this always to be the case once support for wildcards in database names in grants is removed.

  2. **Host name '%' in grants.** The treatment of `%` by the server as a synonym for `localhost` when checking privileges is now also deprecated and thus also subject to removal in a future version of MySQL.

Currently, privileges granted to `'user1'@'%'` are also granted to `'user1'@'localhost'`. This automatic assignment is now deprecated; once the behavior is removed, it will be necessary always to grant privileges to `'user1'@'localhost'` explicitly, using the `localhost` host name.

Note

In both of the cases listed, no warnings or errors are raised in the user session or logs, since doing so could very quickly produce a great many such messages.

(WL #14280, WL #15676)

- `INFORMATION_SCHEMA.PROCESSLIST` is deprecated and subject to removal in a future MySQL release. As such, the implementation of `SHOW PROCESSLIST` which uses that table is also deprecated.

  It is recommended to use the Performance Schema implementation of `SHOW PROCESSLIST` instead. As such, `performance_schema_show_processlist` is also deprecated and subject to removal in a future MySQL release. (WL #15915)

- The `SET_USER_ID` privilege is deprecated and is subject to removal in a future version of MySQL. When `SET_USER_ID` is granted using `GRANT`, a SQL warning informs you of the deprecation status. It is now superseded by these new privileges:

  - `SET_ANY_DEFINER` for definer object creation.
  - `ALLOW_NONEXISTENT_DEFINER` for orphan object protection.

  Both of the new privileges are required to produce orphaned SQL objects using `CREATE PROCEDURE`, `CREATE FUNCTION`, `CREATE TRIGGER`, `CREATE EVENT`, or `CREATE VIEW`.

  During an upgrade of MySQL Server, these privileges coexist as follows:

  - If none of the user accounts has `SET_ANY_DEFINER` or `ALLOW_NONEXISTENT_DEFINER` granted, then all user accounts with `SET_USER_ID` granted are also granted `SET_ANY_DEFINER` and `ALLOW_NONEXISTENT_DEFINER`. The `SET_USER_ID` privilege remains granted.
  - The existing assignment of `SET_USER_ID` to all `SUPER` holders, if none of the existing accounts has it, changes to grant `SET_ANY_DEFINER` and `ALLOW_NONEXISTENT_DEFINER` *only if* none of the user accounts has `SET_ANY_DEFINER`, `ALLOW_NONEXISTENT_DEFINER`, or `SET_USER_ID`.

  At runtime, when the ACL tables are read (as with `FLUSH PRIVILEGES`, a server startup, and so on) and a `SET_USER_ID` grant is sourced, a warning message is added to the error log detailing the account `SET_USER_ID` receiving the grant. (WL #15874)

- Support for the `TLS_AES_128_CCM_8_SHA256` ciphersuite now is deprecated and subject to removal in a future version of MySQL. Any attempt to set it using either the `tls_ciphersuites` or `admin_tls_ciphersuites` system variable now returns a warning. By default, both system variables accept these ciphersuites:

  - `TLS_AES_128_GCM_SHA256`
  - `TLS_AES_256_GCM_SHA384`
  - `TLS_CHACHA20_POLY1305_SHA256`

- `TLS_AES_128_CCM_SHA256`

Several previously supported ciphers now are deprecated and subject to removal in a future version of MySQL. If the deprecated ciphers are specified using either the `ssl_cipher` or `admin_ssl_cipher` system variable, then a warning now is issued. By default, both system variables accept these ciphers:

- `ECDHE-ECDSA-AES128-GCM-SHA256`
- `ECDHE-ECDSA-AES256-GCM-SHA384`
- `ECDHE-RSA-AES128-GCM-SHA256`
- `ECDHE-RSA-AES256-GCM-SHA384`
- `ECDHE-ECDSA-CHACHA20-POLY1305`
- `ECDHE-RSA-CHACHA20-POLY1305`
- `ECDHE-ECDSA-AES256-CCM`
- `ECDHE-ECDSA-AES128-CCM`
- `DHE-RSA-AES128-GCM-SHA256`
- `DHE-RSA-AES256-GCM-SHA384`
- `DHE-RSA-AES256-CCM`
- `DHE-RSA-AES128-CCM`
- `DHE-RSA-CHACHA20-POLY1305`

(WL #15800)

Firewall Notes

- MySQL Enterprise Firewall now permits its memory cache to be reloaded periodically with data stored in the firewall tables. Previous implementations reloaded the cache only at server startup or when the server-side plugin was reinstalled. The new `mysql_firewall_reload_interval_seconds` system variable sets up the schedule to use to reload table data at runtime, or it disables the reload capability (default). For an overview, see Scheduling Firewall Cache Reloads. (WL #15696)

- MySQL Enterprise Firewall previously stored internal tables, functions, and stored procedures in the `mysql` system database. Now, the new `mysql_firewall_database` server system variable enables specifying a custom schema to use with the existing installation scripts and at server startup.

  A new script, `uninstall_firewall.sql`, now simplifies removing the firewall (see Installing or Uninstalling MySQL Enterprise Firewall). (WL #15655)

SQL Function and Operator Notes

- The `STR_TO_DATE()` function did not perform complete range checking on the string to be converted, so that it was possible to pass to it a string which would yield an invalid date, such as `'2021-11-31'`. (Bug #108782, Bug #34704094)

Parallel Event Execution (Multithreaded Replica)

- The `START REPLICA` statement's `SQL_AFTER_GTIDS` option is now fully compatible with the multi-threaded applier.

  Previously, when MTA was enabled (that is, whenever `replica_parallel_workers` was set greater than 1), and the user attempted to use this option, the statement raised the warning `ER_MTS_FEATURE_IS_NOT_SUPPORTED`, and the replica was switched to single-threaded mode. Now this is no longer an issue, and a replica which needs to catch up with missing transactions can do so while taking advantage of the performance benefits of multithreading.

  See START REPLICA Statement, and the documentation for the `replica_parallel_workers` system variable, for more information. (WL #15755)

Optimizer Notes

- Performance of statements using the set operations `EXCEPT` and `INTERSECT` is improved using a new hash table optimization which is enabled automatically for such statements. To cause the optimizer to revert to the temporary table optimization used in previous versions of MySQL, set the `hash_set_operations` optimizer switch introduced in this release to `off`. The amount of memory allocated for this optimization can be controlled by setting the value of the `set_operations_buffer_size` server system variable; increasing the buffer size can further improve execution times of some statements using these operations.

  For more information, see Switchable Optimizations. (WL #15257)

Packaging Notes

- On Windows, the MSI package definition files were updated to work with the Windows Installer XML (WiX) toolset version 4. Note that they can no longer be used with previous versions of the toolset. (Bug #35613791)
- The bundled `libedit` library was upgraded to version 20221030-3.1. (Bug #35489173)

Performance Schema Notes

- The `SUM_ROWS_EXAMINED` column of the Performance Schema `events_statements_summary_by_digest` table did not provide a correct count of rows for Index Merge. (Bug #35616015)
- Under certain circumstances, if an out of memory condition occurred during Performance Schema initialization, the server closed unexpectedly during the cleanup process. (Bug #111860, Bug #35635853)
- The Performance Schema Server Telemetry Metrics service is added in this release. An interface which provides plugins and components a way to query telemetry meters (metric groups), metrics, and metric measurements, in order to periodically export these measurements using Open Telemetry protocol.

  For more information on this interface, see the *Server telemetry metrics service* section in the MySQL Source Code documentation.

  The following were added:

  - Status variables:
    - `Telemetry_metrics_supported`

    - `Performance_schema_meter_lost`
    - `Performance_schema_metric_lost`
  - System variables:
    - `performance_schema_max_meter_classes`
    - `performance_schema_max_metric_classes`
  - Performance Schema tables:
    - `setup_metrics`
    - `setup_meters`

  The following was changed:

  - The default value of `performance_schema_max_rwlock_classes` has changed from 60 to 100 to better accomodate the growing number of rwlocks.

  See Telemetry. (WL #15199)

Security Notes

- Binary packages that include curl rather than linking to the system curl library have been upgraded to use curl 8.4.0. Important issues fixed in curl version 8.4.0 are described at https://curl.se/docs/security.html. (Bug #35897778)

  References: See also: Bug #35709229.

Server Administration

- Added the following types of messages to the server startup and shutdown processes as noted in this list:
  - Start and end messages for server initialization when the server is started with `--initialize` or `--initialize-insecure`; these are in addition to and distinct from those shown during normal server startup and shutdown.
  - Start and end messages for `InnoDB` initialization.
  - Start and end messages for init file execution during server initialization.
  - Start and end messages for for execution of compiled-in statements during server initialization.
  - Start and end mesages for crash recovery during server startup (if crash recovery occurs).
  - Start and end messages for initialization of dynamic plugins during server startup.
  - Start and end messages for compoenents initialization step (apparent during server startup).
  - Messages for shutdown of replica threads, as well as graceful and forceful shutdown of connection threads, during server shutdown.
  - Start and end messages for shutdown of plugins and components during server shutdown.

- Exit code (return value) information with shutdown messages during initialization or shutdown.

In addition, if the server was built using `WITH_SYSTEMD`, the server now includes every `systemd` message in the error log (see The Error Log). (WL #15400)

SQL Syntax Notes

- **Replication:** This release continues work done previously releases to update terminology used in features relating to MySQL Replication. MySQL 8.2 deprecates a number of SQL statements, including `RESET MASTER` (use `RESET BINARY LOGS AND GTIDS` instead), `SHOW MASTER STATUS` (use `SHOW BINARY LOG STATUS`), `SHOW MASTER LOGS` (use `SHOW BINARY LOGS`), and `PURGE MASTER LOGS` (use `PURGE BINARY LOGS`). In addition, the `DISABLE ON SLAVE` option for `CREATE EVENT` and `ALTER EVENT` is now deprecated, and is superseded by `DISABLE ON REPLICA`.

  Related changes include the following:

  - `mysqldump` adds an option `--output-as-version` which defines the level of terminology used in the dump for statements relating to replicas and events, making it possible to create dumps that are compatible with previous versions of MySQL that accept only terminology that is now deprecated. Possible values for this option are `SERVER`, `BEFORE_8_0_23`, and `BEFORE_8_2_0`.

    The default is `SERVER`, which causes `mysqldump` to obtain the server version and output commands that are compatible with that version; this means that, if the server version is previous to 8.2.0, the output contains deprecated `DISABLE ON SLAVE` terminology for events, and if it is previous to 8.0.23, the output also uses the deprecated `SLAVE` and `MASTER` terminology for statements used on replicas. If `--output-as-version` is set to `BEFORE_8_2_0`, `SHOW CREATE EVENT` shows how the event would have been created in a server of a version previous to 8.2.0. If the option is set to `BEFORE_8_0_23`, the dump also uses deprecated statements for replicas such as `START SLAVE` and `CHANGE MASTER TO`.

    This change affects the output of `--events`, `--dump-replica`, `--source-data`, `--apply-replica-statements`, and `--include-source-host-port`.

  - A new value `BEFORE_8_2_0` is added for the `terminology_use_previous` server system variable. Setting the variable to this value causes the server to print `DISABLE ON SLAVE` (now deprecated) instead of `DISABLE ON REPLICA` in the output of `SHOW CREATE EVENT`. This is also now done when `terminology_use_previous` is `BEFORE_8_0_26`, in addition to those effects it already had previously.

  - The term `SLAVESIDE_DISABLED` is now deprecated, and no longer used in event descriptions such as in the Information Schema `EVENTS` table; `REPLICA_SIDE_DISABLED` is now shown instead. This can be overridden using `terminology_use_previous`.

  - The `Com_show_master_status` system status variable is renamed to `Com_show_binary_log_status`. The previous name is deprecated but remains supported for backwards compatibility.

  (WL #14190)

- `EXPLAIN` now supports a `FOR SCHEMA` or `FOR DATABASE` option which causes the statement to be analyzed as if it had been run in the database specified by the option. The database must exist and the user must have the necessary privileges to access it.

  The syntax is as shown here, where *`stmt`* is an explainable statement:

```
EXPLAIN [options] FOR SCHEMA schema_name stmt
```

This causes *`stmt`* to be run as if *`schema_name`* were the current database.

This option is incompatible with `FOR CONNECTION`; the two cannot be used together in the same `EXPLAIN` statement.

For more information and examples, see Obtaining Execution Plan Information. (WL #15785)

Functionality Added or Changed

- **Incompatible Change:** Upgraded the bundled libfido2 library to libfido2 v1.13.0. This libfido2 version requires OpenSSL 1.1.1 or higher, which means the authentication_fido and authentication_fido_client authentication plugins are no longer available on the following platforms: Enterprise Linux 6 and 7, Solaris 11, and SLES 12. (Bug #35685515)
- **Important Change:** For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated to version 3.0.10. Issues fixed in OpenSSL version 3.0.10 are described at https://openssl-library.org/news/openssl-3.0-notes/. (Bug #35702863, Bug #35732474)
- **Important Change:** Added support for Debian 12, Fedora 39, macOS 14, and Ubuntu 23.10. For macOS, the macOS 13 binaries were tested on macOS 14.
- For `mysqldump`: added an `--ignore-views` option to skip table views in the generated dump file.

  Our thanks to Meta for the contribution. (Bug #30284943, WL #15662)
- For `mysqldump`: added `--init-command` and `--init-command-add` options to allow executing SQL statements after connecting or reconnecting to the MySQL server.

  Our thanks to Meta for the contribution. (Bug #27411227, WL #15662)
- For `mysql`: added an `--init-command-add` option that adds an additional SQL statement to execute after connecting or reconnecting to the MySQL server. It's similar to the `--init-command` option.

  Our thanks to Meta for the contribution. (Bug #27411227, WL #15662)
- Added a new `--no-login-paths` command-line option that disables login paths from being processed. It was added to each MySQL client that has the `--login-path` option. (WL #12429)

Bugs Fixed

- **Performance:** `records_in_range` performed an excessive number of disk reads for insert operations.

  Our thanks to Facebook for the contribution. (Bug #109595, Bug #34976138)
- **InnoDB:** If a MySQL table in a system schema had an INSTANT ADD column that was added before 8.0.29 (they are not allowed as of that version), and after MySQL was upgraded to a version greater than 8.0.29, DMLs on these tables would result in the server unexpectedly closing.

  Our thanks to Richard Dang for the contribution. (Bug #35625510, Bug #35981565, Bug #36180360)
- **InnoDB:** Improved `innodb_doublewrite` related error messages. (Bug #35482724)
- **InnoDB:** Fixed processing of single character tokens by a FTS parser plugin.

  Our thanks to Shaohua Wang for the contribution. (Bug #35432973)

- **InnoDB:** Improved flush handling, to help make sure flushing is correctly and uniformly handled throughout the code. (Bug #35385801)
- **InnoDB:** While innodb_redo_log_capacity_update changed log capacity and waited for the next governor thread iteration to finish, the governor thread was not always aware of the change if it was already running in parallel. The log_files_wait_until_next_governor_iteration routine in this case just waited for the current iteration to finish. Now it waits for a full iteration to complete. (Bug #35145188)
- **InnoDB:** If an index creation attempt was interrupted by 'kill query', InnoDB would not always attempt to drop the index being created if another thread referenced the table; and when the other reference was held by the stats background thread. This would raise an assertion error. (Bug #34823462)

  References: This issue is a regression of: Bug #33399379.
- **InnoDB:** B-tree validation could exceed the maximum wait time for large tables with debug mode enabled. (Bug #34411425)
- **InnoDB:** On Windows, the redo log file was not flushed even with `innodb_flush_log_at_trx_commit=1`. (Bug #112078, Bug #35713721)
- **InnoDB:** Fixed a potential transaction rollback issue stemming from the ALTER TABLE EXCHANGE PARTITIONS statement. (Bug #110869, Bug #35352168)

  References: This issue is a regression of: Bug #33700835.
- **InnoDB:** When creating full-text indexes spanning multiple columns, index creation could fail with an error similar to "ERROR 1062 (23000): Duplicate entry 'NULL-NULL'." (Bug #109242, Bug #33542939, Bug #34846823)
- **InnoDB:** The last detected deadlock section of the engine status log was only showing 1024 characters for the combined thread and query information. Fixing by removing the printed query string limit. (Bug #80927, Bug #23036096)
- **Replication:** Memory used by binary log compression was not instrumented. (Bug #35290223)
- **Replication:** The server did not honor the setting for the `binlog_transaction_compression_level_zstd` server system variable. (Bug #34833913)
- **Replication:** The first and last transaction timestamps for the relay log showed incorrect values when querying the `performance_schema.binary_log_transaction_compression_stats` table. This was due to direct use of `my_getsystime()` to populate the `first_transaction_timestamp` and `last_transaction_timestamp` columns in `binary_log_transaction_compression_stats`; this function returns the timestamp value multiplied by 10, leading to the incorrect column values. We fix this by dividing the value returned by `my_getsystime()` by 10 before propagating it. (Bug #32022794)
- **Replication:** For large transactions (greater than 4GB) and very small values of `MASTER_HEARTBEAT_PERIOD`, it was possible for the heartbeat event to be sent before binary log rotation could complete, causing `RESET REPLICA` and similar statements affecting the replica to hang. (Bug #111149, Bug #35431274)
- **Replication:** To detect dependencies and conflicts among transactions, MySQL Replication uses writesets which are extracted from keys that are `NOT NULL` (primary keys by definition, as well as unique and foreign keys which are `NOT NULL`) whenever `binlog_format` is `ROW` and `transaction_write_set_extraction` is enabled (the default in both cases). In the case of multi-column unique keys using only the leading parts of column values (see Column Prefix Key Parts), the writesets were not properly generated, which sometimes led to spurious duplicate key errors; this was due to the fact that the whole values of referenced key columns were added to the writeset in

such cases, even though the keys themselves referenced only the column prefix. (Bug #111027, Bug #35404584)

- **Group Replication:** Removed a memory leak found in the internal function call_gr_incoming_connection_cb(). (Bug #111529, Bug #35526671)
- Following a change made in MySQL 8.0.23, in some cases where `GROUP_CONCAT()` had previously returned `NULL`, it did so no longer. (Bug #35730982)

  References: This issue is a regression of: Bug #32053547.
- Conversion of some subquery predicates to antijoins was not handled correctly. (Bug #35710378)

  References: This issue is a regression of: Bug #35184353.
- When no rows matched the `WHERE` condition of a query, the results of the query as prepared (using `PREPARE`) differed from those of the same query when run directly. (Bug #35689804)
- It was not possible to pass configuration options to the telemetry_client from the command line.

  Also, the telemetry_client configuration options `trace` and `help` were renamed to `otel-trace` and `otel-help`. (Bug #35616866)
- A subquery in the inner expression of a semijoin caused the containing query to be rejected with `Internal error: Key not found`. (Bug #35535934)

  References: See also: Bug #31401468, Bug #34453026.
- Removed an assertion in strings/ctype-ucs2.cc. (Bug #35512282)
- The transformation of correlated scalar subqueries to a join with a derived table added a redundant `ANY_VALUE()` wrapper to a `GROUP BY` column. (Bug #35507109)
- Some queries using `INTERSECT` were not always processed correctly. (Bug #35504625)

  References: See also: Bug #35362424.
- In some cases, the `QUOTE()` function returned binary data rather than the expected `NULL`. (Bug #35499232)
- The server did not always apply the subquery-to-derived transformation correctly to correlated subqueries. (Bug #35497723)
- On Fedora, fixed MySQL Cluster 8.1.0 packages to not reference non-Cluster MySQL packages as dependencies. (Bug #35495002)
- With AddressSanitizer enabled (compiled with -WITH_ASAN=true), ASAN complained and caused MySQL to unexpectedly halt if a MySQL client in interactive mode encountered a dollar sign as the first character. (Bug #35493484)

  References: This issue is a regression of: Bug #35303407.
- For the `REPEAT()` function we may evaluate the count argument during resolution if it is determined to be constant, but due to query rewriting introducing cached values, it was possible for the argument to be evaluated differently at resolution time and at run time.

  We fix this by performing count evaluation for `REPEAT()` in the same way as we do for `RPAD()` and `LPAD()`, using an unsigned integer rather than a signed integer for the value. (Bug #35489153)

- Previously, executing `audit_log_rotate()` manually could cause the function to appear unresponsive when the output performance was low, the workload on the server was heavy, and the `audit_log_strategy` system variable had a write strategy of `ASYNCHRONOUS` or `PERFORMANCE`. (Bug #35397216)

- In some cases the `QUOTE()` function returned `NULL` even though it had previously been resolved as non-nullable. (Bug #35381715)

  References: See also: Bug #35145246, Bug #35195181.

- When `sql_mode` included `PAD_CHAR_TO_FULL_LENGTH`, a `GRANT EXECUTE ON PROCEDURE` statement worked after it was first issued, but following `FLUSH PRIVILEGES` or a restart of the MySQL server, the user named in the `GRANT` statement could no longer execute the procedure. (Bug #35380295)

- When a backslash character was used to escape the '_' wildcard character in a database name pattern inside the `GRANT` statement, the `SHOW DATABASES` and `USE` *`database_name`* queries did not return the expected results. (Bug #35338567)

- A previous fix in MySQL 8.0.30 for a stored program that was not executed correctly following the first invocation did not cover the case where it contained a `SET` statement.

  Our thanks to Hao Lu for the contribution. (Bug #35328028)

  References: This issue is a regression of: Bug #33754993.

- Fixed or removed a number of improper references in the data dictionary and stored procedure code. (Bug #35325895, Bug #35325909, Bug #35325920, Bug #35325935)

- The MySQL command-line client with `--comments=on` was unable to properly process multi-line comments prior to MySQL built-in commands. (Bug #35290350)

- The internal `Item_typecast_year` class did not have its own `print()` member function override, which meant that some of the definitions shown for views, tables, or both could be malformed. (Bug #35244286)

- For a query with a derived condition pushdown where a column in the condition needs to be replaced, a matching item could not found, even when known to be present, when the replacement item was wrapped in a `ROLLUP` while the matching item was not. (Bug #35211828)

  References: This issue is a regression of: Bug #33349994.

- An issue with stored functions was found to be a regression from work done previously to improve internal functions that are used in copying values between columns. (Bug #35150382)

  References: This issue is a regression of: Bug #32742537.

- When a condition that needed to be pushed down to a derived table had one outer-referenced field and another local to the query block, an assertion which checks for consistent dependency information for referenced items did not hold good. This was because, after collecting information related to a field in an expression, dependency and context were not cleared before collecting information for the next field in the expression.

  We fix this by clearing the dependency and context information before looking into a new field in the expression. (Bug #35102220)

- Updated the Kerberos library bundled with commercial builds to version 1.21.2. (Bug #35001935)

- If the probe input for a hash join is empty, we do not need to read the build input, because we already know that the result will be empty. We now therefore read from the probe input first when the join is a left join or an antijoin; for these join types we must read the probe input even if the build input is empty, although the converse is not true. (Bug #34940000)
- Some inserts on tables with triggers were not always handled correctly. (Bug #34920120)

  References: See also: Bug #35178672, Bug #35195079.
- Some `DESCRIBE` statements using `FORMAT = JSON` were not always processed correctly, and sometimes raised assertions in debug builds. (Bug #34909766)
- Some complex queries using multiple common table expressions were not always handled correctly. (Bug #34900334)

  References: This issue is a regression of: Bug #34377854.
- Some queries with window functions were not pushed down correctly. (Bug #34778435)
- Row estimates for a `SELECT DISTINCT` query were the same as for an identical `SELECT` with no `DISTINCT` modifier. This distorted cost estimates, which could lead to choosing inefficient query plans. (Bug #34762651)
- Handled an unexpected condition which could arise during join optimization. (Bug #31401468)

  References: See also: Bug #34350945.
- Execution of a prepared statement containing one or more common table expressions led to an assertion in debug builds. (Bug #111955, Bug #35671595)
- For Enterprise Linux, changed the krb5-devel build requirement from commercial to all builds. (Bug #111596, Bug #35546449)
- `EXPLAIN FORMAT=TREE` lost the subquery in a hash join.

  Our thanks to Wen He and the Tencent team for the contribution. (Bug #111564, Bug #35537921)
- In some cases, an unexpected server exit followed a partitioning function error. (Bug #111443, Bug #35507164)
- When pushing a condition down to a derived table, we clone the condition, and, if the underlying field is a view reference (that is, a field from a merged derived table), we strip off the view reference and clone the expression that it references. When the underlying expression is a constant expression from a table that is on the inner side of an outer join, it cannot be treated as a normal constant because of the need to generate `NULL` values. When we stripped off the view reference, this information was lost, leading to wrong results.

  We fix this by avoiding condition pushdown for such cases. (Bug #111355, Bug #35634714)
- `CREATE ROLE` and `DROP ROLE` statements were not handled correctly when checking for orphaned SQL objects. (Bug #111303, Bug #35471453)
- A join on a `BINARY` column with a `VARBINARY` column of the same size, having matching values equal in size to that of the `BINARY` column in both columns, did not produce any matching rows. (Bug #111290, Bug #35467555)
- A case was found that was not handled by work done in MySQL 8.0.24 for supporting transforms of correlated scalar subqueries: When the scalar subquery is grouped in addition to being correlated, the transformation needs to check that—*for each partition of the result set, as partitioned by the inner*

expression (columns) being added to the group by—there is at most one row in the derived table so constructed. (Bug #111189, Bug #35473657)

- For a user with no roles granted to it, any `SET ROLE` statement caused MySQL to forget all permissions associated directly with that user account until the session ongoing was terminated and a new one started. For example, a `SHOW TABLES` statement that succeeded previously was rejected following `SET ROLE ALL`, `SET ROLE NONE`, or `SET ROLE DEFAULT` because the user's privileges were no longer recognized. (Bug #110997, Bug #35386565)
- `Clone_Snapshot::extend_and_flush_files()` always created files of type `OS_CLONE_DATA_FILE`. This function uses `flush_redo()` to handle redo log files, which must be of type `OS_CLONE_LOG_FILE`, which could sometimes lead to errors in `os_file_set_size()`.

  This problem is now fixed by ensuring that the type is set correctly when creating the new file.

  Our thanks to Tencent for the contribution. (Bug #110569, Bug #35240055)
- MySQL Server did not initialize correctly, using `--initialize` or `--initialize-insecure`, if `autocommit` was set to `OFF`. The `sys` schema was not present.

  As of this release, non-default values of `autocommit` are ignored for this scenario. (Bug #110535, Bug #35254025)
- The low limit heuristic did not work well for `ORDER BY DESC` due to choosing the wrong index. (Bug #107626, Bug #34306497)

Changes in MySQL 8.1.0 (2023-07-18)

- Account Management Notes
- Audit Log Notes
- Binary Logging
- C API Notes
- Compilation Notes
- Component Notes
- Configuration Notes
- Deprecation and Removal Notes
- Logging Notes
- Performance Schema Notes
- Spatial Data Support
- SQL Syntax Notes
- Functionality Added or Changed
- Bugs Fixed

Account Management Notes

- A new password-validation system variable now permits the configuration and enforcement of a minimum number of characters that users must change when attempting to replace their own MySQL

account passwords. This new verification setting is a percentage of the total characters in the current password. For example, if `validate_password.changed_characters_percentage` has a value of 50, at least half of the characters in the replacement account password must not be present in the current password, or the password is rejected.

This new capability is one several that provide DBAs more complete control over password management. For more information, see Password Management. (WL #15751)

Audit Log Notes

- In MySQL 8.0.33, the `audit_log` plugin added support for choosing which database to use to store JSON filter tables. It is now possible to specify an alternative to the default system database, `mysql`, when running the plugin installation script. Use the `audit_log_database` server system variable (or `-D database_name`) on the command line together with the alternative database name, for example:

  ```
  $> mysql -u root -D database_name -p < audit_log_filter_linux_install.sql
  ```

  For additional information about using `audit_log` plugin installation scripts, see Installing or Uninstalling MySQL Enterprise Audit. (Bug #35252268)

- The new `Audit_log_direct_writes` system variable is added to count direct writes into the audit file.

- MySQL Enterprise Audit allocates a temporary buffer to hold data that forms a single event written into the log file. The audit plugin buffers every query that arrives into the audit log. While this is effective for short queries, the server is not always capable of allocating extra memory to hold a long query. Now, the `audit_log` plugin is optimized not to use a temporary buffer when JSON-format logging is used. (WL #15403)

- MySQL Enterprise Audit now supports using the scheduler component to configure and execute a recurring task to flush the in-memory cache. For setup instructions, see Enabling the Audit Log Flush Task. (WL #15567)

Binary Logging

- Several functions now are added to the `libmysqlclient.so` shared library that enable developers to access a MySQL server binary log: `mysql_binlog_open()`, `mysql_binlog_fetch()`, and `mysql_binlog_close()`.

  Our thanks to Yura Sorokin for the contribution. (Bug #110658, Bug #35282154)

C API Notes

- Added the new `mysql_reset_connection_nonblocking()` C API function. It is the counterpart to the `mysql_reset_connection()` synchronous function, for use by applications that require asynchronous communication with the server. Our thanks to Meta for the contribution. (Bug #32202058, WL #15633)

- The new `mysql_get_connect_nonblocking_stage()` C API function permits applications to monitor the progress of asynchronous connections for the purpose of taking appropriate actions based on the progress. Our thanks to Meta for the contribution. (Bug #32202053, WL #15651)

- In the calling function, `len` is initialized to 0 and never changed if `net->vio` is null. This fix adds a check of `net` before dereferencing `vio`.

  Our thanks to Meta for the contribution. (Bug #30809590)

- A variable in the `async` client was uninitialized in certain code paths. It is fixed by always initializing the variable.

  Our thanks to Meta for the contribution. (Bug #30809568)

Compilation Notes

- **Microsoft Windows:** For Windows, improved `MSVC_CPPCHECK` support; and check for MSVC warnings similar to "maintainer" mode. For example, check after all third party configurations are complete. (Bug #35283401)

  References: See also: Bug #34828882.

- **Microsoft Windows:** For Windows builds, improved WIN_DEBUG_NO_INLINE=1 support; usage would exceed the library limit of 65535 objects. (Bug #35259704)

- Upgraded the bundled robin-hood-hashing from v3.8.1 to v3.11.5. (Bug #35448980)

- Removed the unused `extra/libcbor/doc/` directory as `extra/libcbor/doc/source/requirements.txt` inspired bogus pull requests on GitHub. (Bug #35433370)

- Updated the bundled ICU files from version 69.1 to version 73 for the `icu-data-files` package. (Bug #35353708)

- ZSTD sources bundled in the source tree were upgraded to ZSTD 1.5.5 from 1.5.0. (Bug #35353698)

- For SUSE-based systems, changed the default GCC version from version 9 to 12; which is the default compiler on these platforms. (Bug #35341000)

- MySQL did not compile correctly with GCC 12. (Bug #35327995)

- Initialize the internal `MEM_ROOT` class memory with garbage using the TRASH macro to make easier to reproduce bugs caused by reading initialized memory allocated from MEM_ROOT. (Bug #35277644)

- Fixed ODR violations due to multiple different instances of YYSTYPE and other symbols generated by Bison. This includes Bison implementation changes, such as replacing the --name-prefix argument on the Bison command line with api.prefix definitions. (Bug #35232738)

- We now determine stack direction at runtime rather than at configure time. (Bug #35181008)

- Added the `OPTIMIZE_SANITIZER_BUILDS` CMake option that adds `-O1 -fno-inline` to sanitizer builds. It defaults to ON. (Bug #35158758)

- Changed the minimum Bison version requirement from v2.1 to v3.0.4. For macOS, this may require installing Bison via a package manager such as Homebrew. (Bug #35154645, Bug #35191333)

- On Windows, the default for the `MSVC_CPPCHECK` CMake option has changed from OFF to ON. (Bug #35067705)

- MySQL now sets LANG=C in the environment when executing `readelf` to avoid problems with non-ASCII output.

  Our thanks to Kento Takeuchi for the contribution. (Bug #111190, Bug #35442825)

- On macOS, MySQL would not compile if rapidjson was installed via Homebrew. The workaround was to `brew unlink rapidjson`. (Bug #110736, Bug #35311140)

  References: This issue is a regression of: Bug #35006191.

- MySQL would not build with `-DWITH_ZLIB=system`; it'd complain about not finding the system zlib library despite finding it. (Bug #110727, Bug #110745, Bug #35307674, Bug #35312227)

Component Notes

- MySQL Enterprise Edition now supports collecting server trace data in the OpenTelemetry format using the `component_telemetry` component. This data is then forwarded to a configurable endpoint where it can be used by any OpenTelemetry-compatible system.

  

  Note

  Telemetry traces are supported by MySQL Enterprise Edition on Linux platforms only.

  See Telemetry. (WL #15198)

Configuration Notes

- **Microsoft Windows:** On Windows, a new *MySQL Configurator* application was added to help configure a MySQL server. It replaces the *MySQL Installer* application that installed and configured MySQL products in previous versions of MySQL. *MySQL Configurator* (`mysql_configurator.exe`) is included with both the MSI and Zip archive packages. (Bug #35461041)

Deprecation and Removal Notes

- **Important Change:** Since MySQL provides other means of performing database dumps and backups with the same or additional functionality, including `mysqldump` and MySQL Shell Utilities, the `mysqlpump` client utility program has become redundant, and is now deprecated. Invocation of this program now produces a warning. You should keep in mind that `mysqlpump` is subject to removal in a future version of MySQL, and move applications depending on it to another solution, such as those mentioned previously. (WL #15652)

- **Replication:** The `sync_relay_log_info` server system variable is deprecated in this release, and getting or setting this variable or its equivalent startup option `--sync-relay-log-info` now raises a warning.

  Expect this variable to be removed in a future version of MySQL; applications which make use of it should be rewritten not to depend on it before this happens. (Bug #35367005, WL #13968)

- **Replication:** The `binlog_format` server system variable is now deprecated, and subject to removal in a future version of MySQL. The functionality associated with this variable, that of changing the binary logging format, is also deprecated.

  The implication of this change is that, when `binlog_format` is removed, only row-based binary logging, already the default in MySQL 8.0, will be supported by the MySQL server. For this reason, new installations should use only row-based binary logging, and existing ones using the statement-based or mixed logging format should be migrated to the row-based format. See Replication Formats, for more information.

  The system variables `log_bin_trust_function_creators` and `log_statements_unsafe_for_binlog`, being useful only in the context of statement-based logging, are now also deprecated, and are thus also subject to removal in a future release of MySQL.

  Setting or selecting the values of any of the variables just mentioned now raises a warning. (WL #13966, WL #15669)

- **Group Replication:** The `group_replication_recovery_complete_at` server system variable is now deprecated, and setting it produces a warning. You should expect its removal in a future release of MySQL. (WL #15460)

- The `mysql_native_password` authentication plugin now is deprecated and subject to removal in a future version of MySQL. `CREATE USER`, `ALTER USER`, and `SET PASSWORD` operations now insert a deprecation warning into the server error log if an account attempts to authenticate using `mysql_native_password` as an authentication method. (Bug #35336317)

- Previously, if the `audit_log` plugin was installed without the accompanying audit tables and functions needed for rule-based filtering, the plugin operated in legacy filtering mode. Now, legacy filtering mode is deprecated. New deprecation warnings are emitted for legacy audit log filtering system variables. These deprecated variables are either read-only or dynamic.

  (Read-only) `audit_log_policy` now writes a warning message to the MySQL server error log during server startup when the value is not `ALL` (default value).

  (Dynamic) `audit_log_include_accounts`, `audit_log_exclude_accounts`, `audit_log_statement_policy`, and `audit_log_connection_policy`. Dynamic variables print a warning message based on usage:

  - Passing in a non-NULL value to `audit_log_include_accounts` or `audit_log_exclude_accounts` during MySQL server startup now writes a warning message to the server error log.
  - Passing in a non-default value to `audit_log_statement_policy` or `audit_log_connection_policy` during MySQL server startup now writes a warning message to the server error log. `ALL` is the default value for both variables.
  - Changing an existing value using `SET` syntax during a MySQL client session now writes a warning message to the client log.
  - Persisting a variable using `SET PERSIST` syntax during a MySQL client session now writes a warning message to the client log.

  (WL #11248)

- The use of the dollar sign (`$`) as the initial character of an unquoted identifier was deprecated in MySQL 8.0.32. In this release, the use of an unquoted identifier starting with the dollar sign and containing one or more dollar signs in addition to the first one generates a syntax error. Quoted identifiers, and unquoted identifiers that start with a dollar sign but contain no additional occurrences of this character, are not affected by this change. Use of an unquoted identifier with a leading dollar sign that is otherwise permitted continues to raise a warning.

  For more information, see Schema Object Names. (WL #15254)

  References: See also: Bug #34684193.

- MySQL enables control of FIPS mode on the server side and the client side using a system variable and client option. Application programs can use the `MYSQL_OPT_SSL_FIPS_MODE` option to `mysql_options()` to enable FIPS mode on the client. Alternatively, it is possible to handle FIPS mode directly through OpenSSL configuration files rather than using the current server-side system variable and client-side options. When MySQL is compiled using OpenSSL 3.0, and an OpenSSL library and

FIPS Object Module are available at runtime, the server reads the OpenSSL configuration file and respects the preference to use a FIPS provider, if one is set. OpenSSL 3.0 is certified for use with FIPS.

To favor the OpenSSL alternative, the `ssl_fips_mode` server system variable, `--ssl-fips-mode` client option, and the `MYSQL_OPT_SSL_FIPS_MODE` option now are deprecated and subject to removal in a future version of MySQL. A deprecation warning prints to standard error output when an application uses the `MYSQL_OPT_SSL_FIPS_MODE` option or when a client user specifies the `--ssl-fips-mode` option on the command line, through option files, or both.

Prior to being deprecated, the `ssl_fips_mode` server-side system variable was dynamically settable. It is now a read-only variable (accepts `SET PERSIST_ONLY`, but not `SET PERSIST` or `SET GLOBAL`). When specified on the command line or in the `mysqld-auto.cnf` option file (with `SET PERSIST_ONLY`) a deprecation warning prints to the server error log. (WL #15631)

- The `mysql_ssl_rsa_setup` program originally provided a simple way for community users to generate certificates manually, if OpenSSL was installed on the system. Now, `mysql_ssl_rsa_setup` is deprecated because MySQL Community Edition no longer supports using yaSSL as the SSL library, and source distributions no longer include yaSSL. Instead, use MySQL server to generate missing SSL and RSA files automatically at startup (see Automatic SSL and RSA File Generation). (WL #15668)

- The `keyring_file` and `keyring_encrypted_file` plugins now are deprecated. These keyring plugins are superseded by the `component_keyring_file` and `component_keyring_encrypted_file` components. For a concise comparison of keyring components and plugins, see Keyring Components Versus Keyring Plugins. (WL #15659)

- Previously, the MySQL server processed a version-specific comment without regard as to whether any whitespace followed the MySQL version number contained within it. For example, the comments `/*!80034KEY_BLOCK_SIZE=1024*/` and `/*!80034 KEY_BLOCK_SIZE=1024*/` were handled identically. Beginning with this release, when the next character following the version number in such a comment is neither a whitespace character nor the end of the comment, the server issues a warning: `Immediately starting the version comment after the version number is deprecated and may change behavior in a future release. Please insert a whitespace character after the version number`.

  You should expect the whitespace requirement for version-specific comments to become strictly enforced in a future version of MySQL.

  See Comments, for more information. (WL #15686)

- The MySQL client library currently supports performing an automatic reconnection to the server if it finds that the connection is down and an application attempts to send a statement to the server to be executed. Now, this feature is deprecated and subject to removal in a future release of MySQL.

  The related `MYSQL_OPT_RECONNECT` option is still available but it is also deprecated. C API functions `mysql_get_option()` and `mysql_options()` now write a deprecation warning to the standard error output when an application specifies `MYSQL_OPT_RECONNECT`. (WL #15766)

Logging Notes

- To aid in troubleshooting in the event of an excessively long server shutdown, this release introduces a number of new messages that are written to the MySQL error log during shutdown, including those listed here:

  - Startup and shutdown log messages for the MySQL server, including when it has been started with `--initialize`

- Log messages showing start and end of shutdown phases for plugins
- Log messages showing start and end of shutdown phases for components
- Start-of-phase and end-of-phase log messages for connection closing phases
- Log messages showing the number and IDs of threads still alive after being forceably disconnected, and potentially causing a wait

See The Error Log, for more information. (WL #15369)

Performance Schema Notes

- The type used for the Performance Schema `clone_status` table's `gtid_executed` column has been changed from `VARCHAR(4096)` to `LONGTEXT`. (Bug #109171, Bug #34828542)

Spatial Data Support

- The EPSG data set containing spatial reference system data for spatial calculations has been upgraded from version 9.3 to version 9.7. (Bug #28615740)

SQL Syntax Notes

- **JSON:** It is now possible to capture `EXPLAIN FORMAT=JSON` output in a user variable using a syntax extension added in this release. `EXPLAIN FORMAT=JSON INTO var_name stmt` works with any explainable statement *`stmt`* to store the output in the user variable *`var_name`*, where it can be retrieved for later use in analysis. This value is a valid JSON document and can be inspected and manipulated with MySQL JSON functions such as `JSON_EXTRACT()`. (See JSON Functions.)

  The `INTO` clause is supported only with `FORMAT=JSON`; the value of the `explain_format` system variable has no effect on this requirement. If the statement cannot be executed (due to, for instance, a syntax error), the user variable is not updated.

  `INTO` is not supported for `EXPLAIN ANALYZE` or `EXPLAIN FOR CONNECTION`.

  For additional information and examples, see Obtaining Execution Plan Information. (Bug #35362996, WL #15588, WL #15606)

- `CURRENT_USER()` can now be used as a default value for `VARCHAR` and `TEXT` columns in `CREATE TABLE` and `ALTER TABLE ... ADD COLUMN` statements.

  The functions `SESSION_USER()`, `USER()`, and `SYSTEM_USER()` are also supported in all of the cases just mentioned. By way of example, the following sequence of statements now works similarly to what is shown here, with the precise output dependent on your environment:

```
mysql> SELECT CURRENT_USER();
+-------------------+
| CURRENT_USER()    |
+-------------------+
|  sakila@localhost |
+-------------------+
1 row in set (0.00 sec)

mysql> CREATE TABLE t (
    >  c1 VARCHAR(288) DEFAULT (USER()),
    >  c2 VARCHAR(288) DEFAULT (CURRENT_USER()),
    >  c3 VARCHAR(288) DEFAULT (SESSION_USER()),
```

```
    >  c4 VARCHAR(288) DEFAULT (SYSTEM_USER())
    > );
Query OK, 0 rows affected (0.04 sec)

mysql> INSERT INTO t VALUES ROW();
Query OK, 1 row affected (0.01 sec)

mysql> TABLE t;
+-------------------+-------------------+-------------------+-------------------+
| c1                | c2                | c3                | c4                |
+-------------------+-------------------+-------------------+-------------------+
|  sakila@localhost |  sakila@localhost |  sakila@localhost |  sakila@localhost |
+-------------------+-------------------+-------------------+-------------------+
1 row in set (0.00 sec)
```

When used in this way, these functions are also included in the output of `SHOW CREATE TABLE` and `SHOW COLUMNS`, and referenced in the `COLUMN_DEFAULT` column of the Information Schema `COLUMNS` table where applicable.

If you need to insure that values having the maximum possible length can be stored in such a column, you should make sure that the column can accommodate at least 288 characters (255 for the user name and 32 for the host name, plus 1 for the separator `@`). For this reason—while it is possible to use one of these functions as the default for a `CHAR` column, it is not recommended due to the risk of errors or truncation of values. (Bug #17809, Bug #11745618)

Functionality Added or Changed

- **Important Change; Replication:** The default value for the `SOURCE_RETRY_COUNT` option of the `CHANGE REPLICATION SOURCE TO` statement has been changed to 10. This means that, using the default values for this option and for `SOURCE_CONNECT_RETRY` (60), the replica waits 60 seconds between reconnection attempts, and keeps attempting to reconnect at this rate for 10 minutes before timing out and failing over.

  This change also applies to the default value of the `--master-retry-count` server option. You should note that this option is deprecated and therefore subject to removal in a future MySQL release. Use `SOURCE_RETRY_COUNT` with `CHANGE REPLICATION SOURCE TO`, instead.

  See CHANGE REPLICATION SOURCE TO Statement, as well as Asynchronous Connection Failover for Sources, for further information. (WL #15702)

- **Important Change:** For platforms on which OpenSSL libraries are bundled, the linked OpenSSL library for MySQL Server has been updated from OpenSSL 1.1.1 to OpenSSL 3.0. The exact version is now 3.0.9. More information on changes from 1.1.1 to 3.0 can be found at https://www.openssl.org/docs/man3.0/man7/migration_guide.html. (Bug #35475140, WL #15614)

- **Important Change:** MySQL version numbers used in versioned comments now support a major version consisting of one or two digits (previously, only a single digit was supported for this value). See Comments, for more information about how this change affects handling of version-specific comments in MySQL. (WL #15687)

- **Important Change:** Dropped support for Enterprise Linux 6 (and associated glibc 2.12 generic), SUSE 12, Debian 10, MacOS 12, Ubuntu 18.04 and 20.04, Windows 10 and Server 2012R2; and 32-bit versions are no longer built.

- **Replication:** When running in debug mode, `mysqlbinlog` now prints all `Rows_log_event` flags (and not only `STMT_END_F`), and now asserts with `UNKNOWN_FLAG(0x`*N*`)` if it encounters an invalid flag.

  Our thanks to Meta for this contribution. (Bug #33172581)

- **Group Replication:** Any statement that fetches values of system status variables fetches them all, and acquires a read lock on them all as well. This includes statements such as `SHOW STATUS LIKE 'Uptime'` and `SELECT * FROM performance_schema.global_status WHERE VARIABLE_NAME='Uptime'`. In addition, the following operations all acquire a write lock on the status variables:

  - `START GROUP_REPLICATION` and `STOP GROUP_REPLICATION` statements
  - Setting `group_replication_force_members` or `group_replication_message_cache_size`
  - Invoking `group_replication_get_write_concurrency()` or `group_replication_set_communication_protocol()`
  - Automatic rejoin
  - Change of primary with `group_replication_single_primary_mode` enabled

  This meant that a `SHOW STATUS` started after one of the operations just listed was required to wait until the operation was complete before returning.

  Now in such cases, the statement fetching status variables immediately returns their cached values instead of waiting. (Bug #35373030)

  References: See also: Bug #35312441.

- **Group Replication:** Before it elects a new primary, `group_replication_set_as_primary()` waits for all transactions to finish, including all DML operations that are currently being processed. In this release, this function now also waits for all ongoing DDL statements, such as `ALTER TABLE`, to complete as well.

  Listed here are all operations now considered to be DDL statements by `group_replication_set_as_primary()`:

  - `ALTER TABLE`
  - `ANALYZE TABLE`
  - `CACHE INDEX`
  - `CHECK TABLE`
  - `CREATE INDEX`
  - `CREATE TABLE`
  - `DROP INDEX`
  - `LOAD INDEX`
  - `OPTIMIZE TABLE`
  - `REPAIR TABLE`
  - `TRUNCATE TABLE`
  - `DROP TABLE`

  This also includes any open cursors (see Cursors).

For more information, see the description of the `group_replication_set_as_primary()` function, in the MySQL 8.4 Manual. (Bug #34664197, WL #15497)

- **Group Replication:** For better diagnosis and troubleshooting of network instabilities, MySQL Group replication adds a number of variables in this release providing network, control message, and data message statistics for each member of Group Replication. This makes it possible to observe directly the time spent in each of several steps in Group Replication operations.

  Group Replication also adds a new `MEMBER_FAILURE_SUSPICIONS_COUNT` column to the Performance Schema `replication_group_communication_information` table, which shows how many times each group member has been seen as suspect by the local node. For example, in a group with three members, the value of this column should look something like this:

  ```
  {
    "d57da302-e404-4395-83b5-ff7cf9b7e055": 0,
    "6ace9d39-a093-4fe0-b24d-bacbaa34c339": 10,
    "9689c7c5-c71c-402a-a3a1-2f57bfc2ca62": 0
  }
  ```

  These enhancements also help pinpoint how much time and network resources are consumed by user-initiated or background operations, which can then be correlated with overall performance.

  See Group Replication Status Variables, for more information. (WL #13849)

  References: See also: Bug #34279841.

- Binary packages that include curl rather than linking to the system curl library have been upgraded to use curl 8.1.2. (Bug #35329529)

- MySQL now implements client-side Server Name Indication (SNI), which is an extension to the TLS protocol. Client applications can pass a server name to the `libmysqlclient` C API library with the new `MYSQL_OPT_TLS_SNI_SERVERNAME` option for mysql_options(). Similarly, each MySQL client program now includes a `--tls-sni-servername` command option to pass in a name. The new `Tls_sni_server_name` server status variable indicates the name if one is set for the session. Our thanks to Meta for the contribution. (Bug #33176362, WL #14839)

- Comments in the `mysql` client are now enabled by default. To disable them, start `mysql` with the `--skip-comments` option.

  Our thanks to Daniël van Eeden for the contribution. (Bug #109972, Bug #35061087, WL #15597)

- Implemented a `SHOW PARSE_TREE` statement in debug builds to display the JSON-formatted parse tree for a `SELECT` statement. This statement is not supported in release builds, and is available only in debug builds, or by compiling the server using `-DWITH_SHOW_PARSE_TREE`. (WL #15426)

- Previously, invalid SSL server and CA certificates were not identified as problematic until after the server started or after an invalid certificate was loaded at runtime. Now, the new `tls-certificates-enforced-validation` system variable permits a DBA to enforce certificate validation at server startup or when using the `ALTER INSTANCE RELOAD TLS` statement to reload certificates at runtime. With enforcement enabled, discovering an invalid certificate halts server invocation at startup, prevents loading invalid certificates at runtime, and emits warnings. For more information, see Configuring Certificate Validation Enforcement. (WL #13470)

- New server system variables now control the amount of time MySQL accounts that connect to a MySQL server using LDAP pluggable authentication must wait when the LDAP server is down or unresponsive. The default timeout is 30 seconds for the following simple and SASL-based LDAP authentication variables:

  - `authentication_ldap_simple_connect_timeout`
  - `authentication_ldap_simple_response_timeout`
  - `authentication_ldap_sasl_connect_timeout`
  - `authentication_ldap_sasl_response_timeout`

  Connection and response timeouts are configurable through the system variables on Linux platforms only. For more information, see Setting Timeouts for LDAP Pluggable Authentication. (WL #14757)

- Previously, MySQL Server generated and emitted activity-monitoring events through plugin APIs. Now, the server emits events using component APIs. At the same time, to provide backward compatibility with plugins that use audit plugin APIs (such as `audit_log`, `MYSQL_FIREWALL`, `CONNECTION_CONTROL`, `Rewriter`, and so on), the server also implements an intermediate layer that generates required events through plugin APIs. Some of the related error messages may have an `EVENT_TRACKING_` prefix, rather than the current `MYSQL_AUDIT_` prefix. (WL #12652)

Bugs Fixed

- **Incompatible Change; Replication:** Setting server variables equal to SQL `NULL` as options on the command line should not be possible and is not supported. Beginning with this release, setting any of these to `NULL` is disallowed, and attempting to do is rejected with an error.

  The following variables are excepted from this restriction: `admin_ssl_ca`, `admin_ssl_capath`, `admin_ssl_cert`, `admin_ssl_cipher`, `admin_tls_ciphersuites`, `admin_ssl_key`, `admin_ssl_crl`, `admin_ssl_crlpath`, `basedir`, `character_sets_dir`, `ft_stopword_file`, `group_replication_recovery_tls_ciphersuites`, `init_file`, `lc_messages_dir`, `plugin_dir`, `relay_log`, `relay_log_info_file`, `replica_load_tmpdir`, `ssl_ca`, `ssl_capath`, `ssl_cert`, `ssl_cipher`, `ssl_crl`, `ssl_crlpath`, `ssl_key`, `socket`, `tls_ciphersuites`, and `tmpdir`.

  See Server System Variables, for more information. (Bug #109387, Bug #34897517)

- **Important Change:** The default value of the `connection_memory_chunk_size` server system variable, when introduced in MySQL 8.0.28, was mistakenly set at 8912. This fix changes the default to 8192, which is the value originally intended. (Bug #35218020)

- **NDB Cluster:** The fix for a previous issue introduced a slight possibility of unequal string values comparing as equal, if any Unicode 9.0 collations were in use, and the collation hash methods calculated identical hash keys for two unequal strings. (Bug #35168702)

  References: See also: Bug #27522732. This issue is a regression of: Bug #30884622.

- **InnoDB:** Errors occurred when the size of bulk load data was larger than the InnoDB page. (Bug #35332046, Bug #110813)

- **InnoDB:** Purging a large number of system threads sometimes led to congestion. (Bug #35289390, Bug #110685)

- **InnoDB:** Errors occurred when `innodb_thread_concurrency` was set to 999. (Bug #34925101)

- **InnoDB:** Changes in hashing functions made in MySQL 8.0.30 had an adverse effect on performance. (Bug #34870256)

- **InnoDB:** Errors occurred with character sets `ucs2`, `utf16`, and `utf32`. (Bug #34790366)

- **InnoDB:** The rules used for aggregating entries in the redo log have been improved. (Bug #34752625, Bug #108944)
- **InnoDB:** The recovery process yielded contradictory warning and error messages in read-only mode when the redo log was not empty. (Bug #34506094, Bug #108177)
- **InnoDB:** In some cases, tablespace deletion caused issues with the buffer pool. (Bug #34330475, Bug #107689)
- **InnoDB:** `idb2sdi` sometimes reported a partial page at the end of an `.ibd` file, which was ignored. This was caused by a multiplication overflow.

  As part of this fix, we now ensure that `ibd2sdi` checks the size of tablespace files, such that stops with an error if the size of the tablespace file reaches or exceeds the maximum allowed for its page size. See InnoDB Limits, for more information.

  Our thanks to Facebook for the contribution. (Bug #33172685, Bug #104474)
- **InnoDB:** An issue with unsigned integer underflow disabled the heap memory limit, which led to exhaustion of the buffer pool. (Bug #27238364)
- **Packaging; Group Replication:** The group replication plugin from the Generic Linux packages did not load on some platforms that lacked a compatible version of `tirpc`. (Bug #35323208)
- **Replication:** Changes in `session_track_gtids` were not always propagated correctly. (Bug #35401212)
- **Replication:** By design, all DDL operations (including binary log operations such as purging the binary log) acquire a shared lock on the `BACKUP_LOCK` object, which helps to prevent simultaneous backup and DDL operations. For binary log operations, we checked whether any locks existed on `BACKUP_LOCK` but did not check the types of any such locks. This caused problems due to the fact that binary log operations should be prevented only when an exclusive lock is held on the `BACKUP_LOCK` object, that is, only when a backup is actually in progress, and backups should be prevented when purging the binary log.

  Now in such cases, instead of checking for locks held on the `BACKUP_LOCK` object, we acquire a shared lock on `BACKUP_LOCK` while purging the binary log. (Bug #35342521)
- **Replication:** In all cases except one, when `mysqlbinlog` encountered an error while reading an event, it wrote an error message and returned a nonzero exit code, the exception being for the active binary log file (or any binary log where the format_description_log_event had the `LOG_EVENT_BINLOG_IN_USE_F` flag set), in which case it did not write a message, and returned exit code 0, thus hiding the error.

  Now `mysqlbinlog` suppresses only those errors which are related to truncated events, and when doing so, prints a comment rather than an error message. This fix also improves the help text for the `--force-if-open` option. (Bug #35083373)
- **Replication:** Compressed binary log event handling was improved. (Bug #33666652)
- **Replication:** A transaction consisting of events each smaller than 1 GiB, but whose total size was larger than 1 GiB, and where compression did not make it smaller than 1 GiB, was still written to the binary log as one event bigger than 1 GiB. This made the binary log unusable; in effect, it was corrupted since neither the server nor other tools such as `mysqlbinlog` could read it.

  Now, when the compressed data grows larger than 1 GiB, we fall back to processing the transaction without any compression. (Bug #33588473)

- **Replication:** The multithreaded applier wrote messages similar to `Multi-threaded slave: Coordinator has waited 312251 times hitting slave_pending_jobs_size_max; current event size = 8176` into the error log, although they did not belong there. (Bug #32587480)

- **Replication:** Executing either of the statements `FLUSH BINARY LOGS` or `SET GLOBAL binlog_checksum = CRC32` after setting the session transaction access mode to `READ ONLY` led to an unplanned shutdown. Execution of either of these statements causes rotation of the binary log; before doing so, it is necessary to update the `mysql.gtid_executed` table, but this was rejected due to the session transaction access mode being `READ ONLY`.

  We fix this by allowing the binary log rotation to proceed by ignoring `READ ONLY` access mode, as when the server is running in read-only or super-read-only mode. (Bug #109894, Bug #35041573)

- **Group Replication:** In a group replication setup, when there was a source of transactions other than the applier channel, the following sequence of events was possible:

  1. Several transactions being applied locally were already certified, and so were associated with a ticket, which we refer to as Ticket 2, but had not yet been committed. These could be local or nonlocal transactions.

  2. A view is created with Ticket 3, and must wait on transactions from Ticket 2.

  3. The view change (VC1) entered the GR applier channel applier and waited for the ticket to change to 3.

  4. Another group change, and another view change (VC2), occurred while the transactions from Ticket 2 were still completing.

  This gave rise to the following issue: There was a window wherein the last transaction from Ticket 2 had already marked itself as being executed but had not yet popped the ticket; VC2 popped the ticket instead but never notified any of the participants. This meant that VC1 continued to wait indefinitely for the ticket to change, and with the additional effect that the worker could not be killed.

  We fix this by checking for the need to break each second so that this loop is responsive to changes in the loop condition; we also register a new stage, so that the loop is more responsive to kill signals. (Bug #35392640)

  References: See also: Bug #35206392, Bug #35374425.

- **Group Replication:** Executing `SET GLOBAL group_replication_force_members = host:port` and `SHOW STATUS LIKE 'group_replication_primary_member'` on the host in parallel sometimes led to a timeout while waiting for a new view. (Bug #35312441)

- **Group Replication:** Removed a memory leak discovered in `Network_provider_manager::open_xcom_connection()`. (Bug #34991101)

- **Group Replication:** When a group action was sent to the group and the connection was killed on the coordinator, group members were in different states, with members which received the coordinated action waiting for the member that executed it, and the member which started execution having nothing to process, which caused problems with coordination of the group.

  Now in such cases, we prevent this issue from occurring by causing group actions to wait until all members have completed the action. (Bug #34815537)

- **Group Replication:** Cleanup of resources used by OpenSSL connections created indirectly by group replication was not carried out as expected at all times. We fix this by adding cleanup functionality that can be called at any time such connections are created by group replication. (Bug #34727136)

- **Group Replication:** In some cases, the MySQL server continued to accept connections intended for group replication even after the group replication plugin had commenced shutdown. (Bug #34398622)

- **JSON:** When the result of `JSON_VALUE()` was an empty string and was assigned to a user variable, the user variable could in some cases be set to `NULL` instead, as shown here:

```
mysql> SELECT JSON_VALUE('{"fname": "Joe", "lname": ""}', '$.lname') INTO @myvar;
Query OK, 1 row affected (0.01 sec)

mysql> SELECT @myvar = '', @myvar IS NULL;
+-------------+----------------+
| @myvar = '' | @myvar IS NULL |
+-------------+----------------+
|        NULL |              1 |
+-------------+----------------+
1 row in set (0.00 sec)
```

  With this fix, the query just shown now returns `(1, 0)`, as expected. (Bug #35206138)

- **JSON:** Some JSON schemas were not always processed correctly by `JSON_SCHEMA_VALID()`. (Bug #109296, Bug #34867398)

- Some combinations of regular expression functions and arithmetic functions were not always evaluated correctly. (Bug #35462660)

- In rare cases, MySQL server could exit rather than emit an error message as expected. (Bug #35442407)

- The internal resource-group enhancement added in MySQL 8.0.31 and refactored in MySQL 8.0.32 is now reverted. (Bug #35434219)

  References: Reverted patches: Bug #34702833.

- An in-place upgrade from MySQL 5.7 to MySQL 8.0, without a server restart, could result in unexpected errors when executing queries on tables. This fix eliminates the need to restart the server between the upgrade and queries. (Bug #35410528)

- A fix in MySQL 8.0.33 made a change for `ORDER BY` items already resolved so as not to resolve them again (as is usually the case when a derived table is merged), but this did not handle the case in which an `ORDER BY` item was itself a reference. (Bug #35410465)

  References: This issue is a regression of: Bug #34890862.

- Changes in `session_track_gtids` were not always handled correctly. (Bug #35401212)

- Some pointers were not always released following statement execution. (Bug #35395965)

- In `Item_func_min_max::cmp_datetimes()`, it was sometimes possible to set `null_value` when the current item was not actually nullable. (Bug #35380560, Bug #35492532)

- Some instances of subqueries within stored routines were not always handled correctly. (Bug #35377192)

- Fortified parsing of the network packet data sent by the server to the client. (Bug #35374491)

- Some queries using `INTERSECT` were not always processed correctly. (Bug #35362424)

- A `SELECT` statement within a prepared statement unexpectedly returned different results on successive executions. (Bug #35340987, Bug #35846585, Bug #35846873)

  References: This issue is a regression of: Bug #35060385.

- Encryption enhancements now strengthen compliance and remove the use of deprecated APIs. (Bug #35339886)

- When a column reference given by table name and column name was looked up in the function `find_item_in_list()`, we ignored that the item searched for might not have a table name, as it was not yet resolved. We fix this by making an explicit check for a null table name in the sought-after item. (Bug #35338776)

- Deprecated the `lz4_decompress` and `zlib_decompress` command-line utilities that exist to support the deprecated `mysqlpump` command-line utility. (Bug #35328235)

- Certain queries using `NULLIF()` led to an assertion. The issue was found to originate in `Item_func_nullif::resolve_type_inner()`, where, if the original data type was a temporal type, the type was adjusted to a string type but the result type was not also adjusted accordingly, which could later lead to later inconsistencies. This is fixed by setting the result type in such cases to `STRING_RESULT`. (Bug #35323398)

- On Linux, the `mysql` client's `ssl_session_data_print` command now saves files with an 0600 absolute mode (permissions) instead of the default 0644; when passing in the optional *filename* parameter. (Bug #35304195)

- Queries using `LIKE '%...%'` ran more poorly than in previous versions of MySQL. (Bug #35296563)

- We calculate the cost of `MATERIALIZE` paths by adding the cost of materialization to the sum of the cost of the child paths. If the number of output rows is undefined for a child, we ignore that child, as we assume that the cost of that child is then also undefined. If the child was an `AGGREGATE` path with implicit grouping, the number of output rows could be set to 1, even when the cost was undefined. We fix this by checking in such cases whether the cost of the child is actually defined, and—if it is not—skipping it. (Bug #35240913)

  References: See also: Bug #33834146, Bug #34302461.

- In `Bounded_queue::push()`, when `Key_generator::make_sortkey()` returns `UINT_MAX` (error), then no key has been produced; now when this occurs, we no longer update the internal queue.

  As part of this fix, `push()` now returns true on error. (Bug #35237721)

- The `authentication_oci` plugin is fixed to allow federated and provisioned users to connect to a DB System as a mapped Proxy User using an ephemeral key-pair generated through the OCI CLI. (Bug #35232697)

- Some queries using common table expressions were not always processed correctly. (Bug #35231475)

- The internal function `compare_pair_for_nulls()` did not always set an explicit return value. (Bug #35217471)

- Removed the clang-tidy checks that clash with the MySQL coding style. (Bug #35208735)

- Some subqueries using `EXISTS` in both the inner and outer parts of the query were not handled correctly. (Bug #35201901)

- Rotated audit log files now always reset the ID value of the bookmark to zero, rather than continuing the value from the previous file. (Bug #35200070)

- Errors were not always propagated correctly when evaluating items to be sorted by filesort. (Bug #35195181)

  References: See also: Bug #35145246.

- In certain cases, `UNIX_TIMESTAMP()` was evaluated prematurely. (Bug #35174730)

- When attempting to transform a scalar subquery to a derived table, we saw the top level query is implicitly grouped, so we moved the grouping into a first derived table. If, after this, we did not perform the original transformation, the initial transform had still been carried out, which should have been valid, but we neglected to look at join conditions in subqueries when substituting reference fields. In such cases we also did not descend into any subqueries other than derived table subqueries. (Bug #35170671)

- The fix for a previous issue with `ROLLUP` led to a premature server exit in debug builds. (Bug #35168639)

  References: This issue is a regression of: Bug #33830659.

- Simplified the implementation of `Item_func_make_set::val_str()` to make sure that we never try to reuse any of the input arguments, always using the local string buffer instead. (Bug #35154335, Bug #35158340)

- The transform of a scalar subquery into a join with a derived table where the subquery is in the `SELECT` list and the containing query is implicitly grouped should be allowed, but was rejected when the `subquery_to_derived` optimizer switch was enabled. (Bug #35150438)

- When transforming subqueries to a join with derived tables, with the containing query being grouped, we created an extra derived table in which to do the grouping. This process moved the initial select list items from the containing query into the extra derived table, replacing all of the original select list items (other than subqueries, which get their own derived tables) with columns from the extra derived table.

  This logic did not handle `DEFAULT` correctly due to the manner in which default values were modelled internally. This fix adds support for `DEFAULT(`*`expression`*`)` in queries undergoing the transform previously mentioned. This fix also solves an issue with item names in metadata whereby two occurrences of the same column in the select list were given the same item name as a result of this same transform. (Bug #35150085, Bug #35101169)

- A query of the form `SELECT * FROM t1 WHERE (SELECT a FROM t2 WHERE t2.a=t1.a + ABS(t2.b)) > 0` should be rejected with `Subquery returns more than 1 row`, but when the `subquery_to_derived` optimization was enabled, the transform was erroneously applied and the query returned an incorrect result. (Bug #35101630)

- Handling of certain potentially conflicting `GRANT` statements has been improved. (Bug #35089304)

- A query using both `MEMBER OF()` and `ORDER BY DESC` returned only a partial result set following the creation of a multi-valued index on a JSON column. This is similar to an issue fixed in MySQL 8.0.30, but with the addition of the `ORDER BY DESC` clause to the prblematic query. (Bug #35012146)

  References: See also: Bug #106621, Bug #33917625.

- The debug server asserted on certain operations involving `DECIMAL` values. (Bug #34973932)

- The nullability of `ANY` subqueries was sometimes incorrect because the nullability of the left operand was not taken into account. We fix this by marking an `ANY` subquery as nullable whenever the left operand is nullable. (Bug #34940790)

- All instances of adding and replacing expressions in the select list when transforming subqueries to use derived tables and joins have been changed so that their reference counts are maintained properly. (Bug #34927110)

- Aggregation of item type from multiple arguments required processing in multiple internal functions; this has been simplified such that it is now performed in one function only. This should improve the efficiency of this process, which is used for expressions that are the results of set operations, and those that are output from the `CASE` operator (and the associated functions `COALESCE()` and `IF()`), as well as `LEAD()` and `LAG()`. (Bug #34847836)

- Index Merge (see Index Merge Optimization) should favor ROR-union plans (that is, using RowID Ordered Retrieval) over sort-union plans if they have similar costs, since sort-union requires additionally sorting of the rows by row ID whereas ROR-union does not.

  For each part of a `WHERE` clause containing an `OR` condition, the range optimizer gets the best range scan possible and uses all these range scans to build an index merge scan (that is, a sort-union scan). If it finds that all the best range scans are also ROR-scans, the range optimizer always proposes a ROR-union scan because it is always cheaper than a sort-union scan. Problems arose when the best range scan for any one part of an `OR` condition is not a ROR-scan, in which case, the range optimizer always chose sort-union. This was true even in cases, where it might be advantageous to choose a ROR-scan (even though it might not be the best range scan to handle one part of the OR condition), since this would eleminate any need to sort the rows by row ID.

  Now, in such cases, when determining the best range scan, the range optimizer also detects whether there is any possible ROR-scan, and uses this information to see whether each part of the `OR` condition has at least one possible ROR-scan. If so, we rerun the range optimizer to obtain the best ROR-scan for handling each part of the `OR` condition, and to make a ROR-union path. We then compare this cost with the cost of a sort-union when proposing the final plan. (Bug #34826692, Bug #35302794)

- Selecting from a view sometimes raised the error `Illegal mix of collations ... for operation '='` when the collation used in the table or tables from which the view definition selected did not match the current session value of `collation_connection`. (Bug #34801210)

- If a view (`v1`) accessed another view (`v2`), and if `v2` was recreated, then `SHOW COLUMNS FROM v1` reported an invalid view error. This issue occurred when the user was granted privileges to all resources (`*.*`), but not table-level or column-level privileges. It is fixed by removing the condition that caused an omission of the proper table-level check. (Bug #34467659)

- `ANALYZE TABLE` with `UPDATE HISTOGRAM` or `DROP HISTOGRAM` invalidated the `TABLE_SHARE`, which meant that subsequent queries were required to wait for all queries then running to terminate before the old `TABLE_SHARE` could be freed and a new one initialized with the updated collection of histograms for the table. This could introduce long waits, as queries issued after the `TABLE_SHARE` was invalidated had to wait for any existing long-running queries that referenced the old `TABLE_SHARE` to terminate.

  This fix changes the behavior of the histogram commands to mark tables for reopening instead of invalidating the `TABLE_SHARE`. Instead of having a single set of table histograms cached on the `TABLE_SHARE`, we now maintain a collection of reference-counted sets of table histograms on the share. When the histograms on a given table are modified, we now insert a new snapshot of the set of histograms into the collection on the `TABLE_SHARE` and mark it current. When a table object is opened, it acquires a pointer to the current snapshot of the set of histograms for the table from the share, and when the table object is closed it releases its pointer back to the share.

  By using multiple reference-counted versions of histogram statistics for a table we avoid the potential wait for synchronization of all queries on the table around the reinitialization of the `TABLE_SHARE` when histograms are updated or dropped. (Bug #34288890, Bug #35419418)

- Valid MySQL commands (`use` and `status`) and C API functions (`mysql_refresh`, `mysql_stat`, `mysql_dump_debug_info`, `mysql_ping`, `mysql_set_server_option`, `mysql_list_processes`, and `mysql_reset_connection`) could write an error message to the audit log, even though running the command or calling the function emitted no such error. (Bug #33966181)

- Increased the maximum fixed array size to 8192 instead of 512. This fixes an issue with `mysqladmin` extended status requests, which can exceed 512 entries.

  Our thanks to Meta for the contribution. (Bug #30810617)

- The `mysqldump --column-statistics` option attempted to select from `information_schema.column_statistics` against MySQL versions before 8.0.2, but this now generates the `warning column statistics not supported by the server` and sets the option to false.

  Our thanks to Meta for the contribution. (Bug #28782417)

- The function used by MySQL to get the length of a directory name was enhanced. (Bug #28047376)

- Executing a query with an implicit aggregation should return exactly one row, unless the query has a `HAVING` clause that filters out the row, but a query with a `HAVING` clause which evaluated to `FALSE` sometimes ignored this, and returned a row regardless. (Bug #14272020)

- Queries using `DISTINCT` treated 0 and -0 differently. (Bug #117845, Bug #34361437, Bug #37779495)

- `EXPLAIN` and `EXPLAIN ANALYZE` did not produce the same output for covering index scans. (Bug #117666, Bug #34527697, Bug #37691057)

- Some complex queries using multiple common table expressions were not always handled correctly. (Bug #112021, Bug #35284734, Bug #35694546)

  References: See also: Bug #111994, Bug #35686058. This issue is a regression of: Bug #34377854.

- The presence of an unused window function in a query, along with an `ORDER BY` that could have been eliminated, led to an unplanned server exit. (Bug #111585, Bug #35168639, Bug #35204224, Bug #35545377)

  References: This issue is a regression of: Bug #35118579.

- `ORDER BY RANDOM_BYTES()` had no effect on query output. (Bug #111252, Bug #35148945, Bug #35457136)

- Improved the `mysql` client's *status* output; the Protocol row now includes the compression algorithm and zstd level.

  Our thanks to Daniël van Eeden for the contribution. (Bug #110950, Bug #35369870)

- The MySQL source code documentation was missing the following information about C API protocols: `zstd_compression_level` is only sent when `CLIENT_ZSTD_COMPRESSION_ALGORITHM` is set.

  Our thanks to Daniël van Eeden for the contribution. (Bug #110939, Bug #35365351)

- In certain cases, `VALUES ROW()` did not handle expressions which evaluated to `NULL` correctly. (Bug #110925, Bug #35363550)

- The `QUOTE()` function returned unexpected results with columns selected from a table having the `utf16` character set. (Bug #110672, Bug #35286970)

- Fixed an issue which could occur when loading user-defined functions. (Bug #110576, Bug #35242734)

- Concurrent execution of `FLUSH STATUS`, `COM_CHANGE_USER`, and `SELECT FROM I_S.PROCESSLIST` could result in a deadlock. A similar issue was observed for concurrent execution of `COM_STATISTICS`, `COM_CHANGE_USER`, and `SHOW PROCESSLIST`.

  Our thanks to Dmitry Lenev for the contribution. (Bug #110494, Bug #35218030)

- The `mysqldump` utility could generate invalid INSERT statements for generated columns. (Bug #110462, Bug #35208605)

- For `mysqldump`: usage would unexpectedly halt when used against tables with functional indexes. (Bug #110452, Bug #35205310)

- An impossible `WHERE` similar to `WHERE int_col = 05687.3E-84` was not always handled correctly. (Bug #110434, Bug #35200367)

- The loading and unloading of UCA character sets has been rewritten to improve memory handling when cycling through initialization and deinitialization. (Bug #109540, Bug #110836, Bug #34969838, Bug #35341006)

- During optimization, range-select tree creation uses logic which differs based on the left-hand side of the `IN()` predicate. For a field item, each value on the right-hand side is added to an OR tree to create the necessary expression. In the case of a row item comparison (example: `WHERE (a,b) IN ((n1,m1), (n2, m2), ...)`), an expression in disjunctive normal form (DNF) is needed. A DNF expression is created by adding an `AND` tree with column values to an `OR` tree for each set of RHS values, but instead the `OR` tree was added to the `AND` tree causing the tree merge to require exponential time due to $O(n^2)$ runtime complexity. (Bug #108963, Bug #34758905)

- When using `SELECT` to create a table and the statement has an expression of type `GEOMETRY`, MySQL could generate an empty string as the column value by default. To resolve this issue, MySQL no longer generates default values for columns of type `GEOMETRY` under these circumstances. Our thanks to Tencent for the contribution. (Bug #107996, Bug #34426943)

- Removed an assertion encountered when creating fields of type `YEAR` for temporary tables holding results of `UNION` operations. (Bug #107826, Bug #34370933, Bug #35282236)

- For index skip scans, the first range read set an end-of-range value to indicate the end of the first range, but the next range read did not clear the stale end-of-range value and applies this stale value to the current range. Since the indicated end-of-range boundary had already been crossed in the previous range read, this caused the reads to stop, causing multiple rows to be missed in the result.

  We fix this by making sure in such cases that the old end-of-range value is cleared. (Bug #107460, Bug #34235624, Bug #34982949)