MySQL Enterprise Backup 8.0 Release Notes

Abstract

This document lists the changes to the MySQL Enterprise Backup 8.0 product, beginning with the most recent release. Each release section covers added or changed functionality, bug fixes, and known issues, if applicable. For information about changes in a different MySQL Enterprise Backup series, see the release notes for that series.

For additional MySQL Enterprise Backup 8.0 documentation, see the MySQL Enterprise Backup User's Guide (Version 8.0.20).

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: 2020-04-30 (revision: 20217)

Table of Contents

Preface and Legal Notices .......................................................... 1
Changes in MySQL Enterprise Backup 8.0.20 (2020-04-27, General Availability) ........................................ 2
Changes in MySQL Enterprise Backup 8.0.19 (2020-01-13, General Availability) ........................................ 4
Changes in MySQL Enterprise Backup 8.0.18 (2019-10-14, General Availability) ........................................ 7
Changes in MySQL Enterprise Backup 8.0.17 (2019-07-22, General Availability) ........................................ 9
Changes in MySQL Enterprise Backup 8.0.16 (2019-04-25, General Availability) ........................................ 11
Changes in MySQL Enterprise Backup 8.0.15 (2019-02-01, General Availability) ........................................ 13
Changes in MySQL Enterprise Backup 8.0.14 (2019-01-21, General Availability) ........................................ 13
Changes in MySQL Enterprise Backup 8.0.13 (2018-10-22, General Availability) ........................................ 14
Changes in MySQL Enterprise Backup 8.0.12 (2018-07-27, General Availability) ........................................ 15
Changes in MySQL Enterprise Backup 8.0.11 (2018-04-19, General Availability) ........................................ 16
Index ................................................................................. 18

Preface and Legal Notices

This document lists the changes to the MySQL Enterprise Backup 8.0 product, beginning with the most recent release.

Legal Notices

Copyright © 2003, 2020, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication,
disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

This documentation is NOT distributed under a GPL license. Use of this documentation is subject to the following terms:

You may create a printed copy of this documentation solely for your own personal use. Conversion to other formats is allowed as long as the actual content is not altered or edited in any way. You shall not publish or distribute this documentation in any form or on any media, except if you distribute the documentation in a manner similar to how Oracle disseminates it (that is, electronically for download on a Web site with the software) or on a CD-ROM or similar medium, provided however that the documentation is disseminated together with the software on the same medium. Any other use, such as any dissemination of printed copies or use of this documentation, in whole or in part, in another publication, requires the prior written consent from an authorized representative of Oracle. Oracle and/or its affiliates reserve any and all rights to this documentation not expressly granted above.

**Documentation Accessibility**

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at https://www.oracle.com/corporate/accessibility/.

**Access to Oracle Support**

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit https://www.oracle.com/corporate/accessibility/learning-support.html#support-tab.

**Changes in MySQL Enterprise Backup 8.0.20 (2020-04-27, General Availability)**

MySQL Enterprise Backup 8.0.20 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.20. For earlier versions of MySQL 8.0, use the MySQL Enterprise Backup version with the same version number as the server. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.
Functionality Added or Changed

• Functionality Added or Changed

The `tablespace_tracker` file has been simplified: it now contains only two fields for each external tablespace: `server_file_path` and `space_id`. `mysqlbackup` no longer relies on the file for information on the `backup_file_path` and the tablespace type, which means that users no longer need to update the `tablespace_tracker` file when they move a directory backup to a new location.

• Table-Level Recovery (TLR) is a new feature of MySQL Enterprise Backup that allows selective restores of tables or schemas from full backups; see Table-Level Recovery (TLR) for details.

• The legacy option `--include` is now deprecated. A deprecation warning is now issued by `mysqlbackup` whenever the option is used. The `--include-tables` and `--exclude-tables` options should be used instead for partial backups and restores.

Bugs Fixed

• A backup failed with `ERROR: Bad table space file header` when the server had more than one system tablespace files. It was because `mysqlbackup` looked for the tablespace file header at the wrong place, and this patch corrects the problem. (Bug #30983009)

• During an incremental backup, `mysqlbackup` simply repeated the Server Repository Options when trying to print the Backup Configuration Options in its output. (Bug #30948251)

• Backups failed when the server used a keyring plugin and its `sql_mode` was set to `ANSI_QUOTES`. It was because `mysqlbackup` used the wrong kind of quotes in the situation when querying the server, and that has been fixed by this patch. (Bug #30920140)

• An incremental optimistic image backup failed when the server was started with a non-default `innodb_data_file_path` value containing more than one InnoDB system tablespace file. It was because `mysqlbackup` could not handle the situation in which the two different files had the same space ID, and this patch fixes the problem. (Bug #30914039)

• A partial restore of a TTS backup failed when the file path specified with the `--datadir` option contained extra slashes (/) when compared with the data directory file path the server was started with. With this fix, such extra slashes for the `--datadir` option are ignored. (Bug #30914039)

• During a backup operation, if any tablespace's encryption status was changed (for example, from encrypted to unencrypted or vice versa, and even if the table was eventually changed back to its original encryption status), `mysqlbackup` reported success, but it quit unexpectedly during the restore operation of the backup because of its inconsistency. With this fix, the encryption statuses of tablespaces are properly tracked throughout the backup operation, so that the tables are consistently backed up. (Bug #30599476)

• When the `--src-entry` option was used with the `extract` command, a trailing slash in its value (for example, in `foo/`) was ignored, so that instead of extracting from the backup only those directories whose names ended with the value (for example, `datadir/foo/`), `mysqlbackup` also extracted all files whose paths contained the value (for example, `datadir/bar/foo.sdi`). With this fix, the trailing slash is honored, and it only causes folders whose names end with the value to be extracted.

It is also clarified in the documentation that the value of the `--src-entry` option is actually used to match any files or non-empty folders that contain the value in their names, and a trailing slash is interpreted as described in the last paragraph. (Bug #30461403)

• When there was a user-created `mysql.backup_progress` table on a server that was being backed up, `mysqlbackup` finished the backup successfully, but also printed error messages and recorded a backup failure in the `backup_history` table. With this fix, the backup is finished as normal with a warning. (Bug #30351172)
• The binary log basename appeared as an empty string in the progress report of a `copy-back-and-apply-log` operation. (Bug #29936558)

• When a data tablespace had the same name as an undo tablespace on the server, a compressed backup containing the tablespace could be created by `mysqlbackup`, but the backup could not be restored due to the filename conflict. With this fix, the backup fails in the situation. (Bug #29881640)

• A backup failed when it involved encrypted InnoDB tables and the `--skip-unused-pages` option was used. (Bug #29861298)

• When a compressed backup was created with the `backup-and-apply-log` command and then restored using the `copy-back-and-apply-log` command, the redo log were missing from the restored server, causing an InnoDB error when the server was started. (Bug #29851603)

• A backup failed when the `--skip-unused-pages` and `--optimistic-busy-tables` options were used together. (Bug #29840923)

• When the server to be backed up has `super_read_only=ON`, `mysqlbackup` gave the warning that the backup operation could not be logged even if the `--no-history-logging` option has already been used with the backup command. This patch removes the unnecessary warning. (Bug #29742011)

• A `backup-and-apply-log` operation failed for a TTS backup if the `--compress` option was used. (Bug #29639871)

• An `extract` operation for the file `meta/backup_variables.txt` failed with `mysqlbackup` complaining that the value of the option `--src-entry` did not match any table in the backup. With this fix, `mysqlbackup` no longer throws an error in the situation, but prints the message “The src-entry 'backup_variables.txt' is by default extracted to backup-dir”. (Bug #29519710)

• During a backup operation, `mysqlbackup` printed messages regarding the encryption keyring even though the server did not utilize InnoDB table encryption. With this patch, `mysqlbackup` stops printing such messages in the situation. (Bug #29151380)

Changes in MySQL Enterprise Backup 8.0.19 (2020-01-13, General Availability)

MySQL Enterprise Backup 8.0.19 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.19. For earlier versions of MySQL 8.0, use the MySQL Enterprise Backup version with the same version number as the server. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.

• Functionality Added or Changed

• Bugs Fixed

Functionality Added or Changed

• **Important Change:** For the `backup-to-image`, `extract`, `list-image`, and `copy-back-and-apply-log` commands, any relative path specified with `--backup-image` is now taken to be relative to the current directory in which the command is run. (Bug #29943103)

• **Important Change:** The storage engine for the `mysql.backup_progress` table on a backed-up server has switched from CSV to InnoDB. Also, an auto-increment primary key `id` column plus a composite index on the table’s `backup_id` and `current_timestamp` columns have been added. When working with a Group Replication setup, `mysqlbackup` now makes the backup_progress table available to all members of the server group by making sure that the table is updated on a primary node during each `mysqlbackup` operation.

When MySQL Enterprise Backup 8.0.19 or later tries to perform its first full backup on a database, it automatically checks the format of the `mysql.backup_progress` table. If it detects that the
table is in the old format (which means the server has been upgraded from 8.0.18 or earlier and has been backed up by MySQL Enterprise Backup before), it attempts to perform an update on the table automatically. Grant these privileges required for the table upgrade to the mysqlbackup user on the server:

```
GRANT ALTER ON mysql.backup_progress TO 'mysqlbackup'@'localhost';
GRANT CREATE, INSERT, DROP ON mysql.backup_progress_old TO 'mysqlbackup'@'localhost';
GRANT CREATE, INSERT, DROP, ALTER ON mysql.backup_progress_new TO 'mysqlbackup'@'localhost';
```

See Backup Progress Table Update for details. (Bug #29882544, Bug #28695518)

- **Important Change:** Due to the added protection for the backups against inconsistency caused by parallel DDL operations (see discussions on the new feature below), the user by which mysqlbackup connects to the MySQL server must now be granted the SELECT ON *.* privilege; see Grant MySQL Privileges to Backup Administrator.

- The logging for backup restore has been improved: at the steps for setting the sizes of the log files, the names of the log files are now included. (Bug #30380310)

- mysqlbackup now prints a stack trace after being terminated by a signal. (Bug #30042338)

- When mysqlbackup fails to connect to a server, the warning returned by mysqlbackup now includes the hostname and port number for TCP connections, and the socket information for socket connections. This is particularly helpful for a Group Replication setup, for which mysqlbackup might attempt to connect to more than one host. (Bug #30040027)

- If a binary log file could not be opened during a non-incremental backup, the backup would still be completed, but the backup_history table would indicate the backup had failed. With this fix, backups of all types fail if any relevant binary log files could not be opened, and a proper error is thrown. (Bug #29882381)

- mysqlbackup now includes the configuration files auto.cnf and mysqld-auto.cnf from a server in its backup (except for a TTS backup). They are restored to the target server’s data directory as backup-auto.cnf and backup-mysqld-auto.cnf respectively. To use those files to configure your restored server, rename them to their original names before starting the server. (Bug #27121423, Bug #30033486)

- Master key rotation for binary log encryption on the server in between a full and an incremental backup, as well as between two incremental backups performed by mysqlbackup, is now supported. During an incremental backup, mysqlbackup now records encryption information for all the encrypted binary log files (including those already backed up in earlier full or incremental backups) unless the --skip-binlog option is used, in which case a warning is given that the older binary log files might become unrestorable.

Also, the --skip-binlog option now makes binary log to be skipped not just for the current backup operation, but also for all subsequent incremental backups that are based on the current backup.

- In the past, if DDL operations took place when a backup was in progress, the backup might become inconsistent. It is now safe to have DDL operations (CREATE TABLE, RENAME TABLE, DROP TABLE, ALTER TABLE, and operations that map to ALTER TABLE like CREATE INDEX) happening on the server in parallel with a backup operation as long as:

  - The tables involved exist in their own tablespaces, instead of being in the system tablespace or some general tablespaces.
  - These server features have not been applied to the tables involved:
    - Data-at-rest encryption
    - Page-level compression
    - Full-text indexing
The backup is not taken with the following `mysqlbackup` features:

- Optimistic backup
- Transportable tablespace (TTS)
- Redo log archiving
- Incremental backups with-redo-log-only
- Incremental backup using page-tracking

**Bugs Fixed**

- `mysqlbackup` reported that an `extract` operation succeeded even if `--src-entry` specified a file that did not exist in the backup. This fix adds a validation for the `--src-entry` value and makes `mysqlbackup` throw an error if the validation fails. (Bug #30461566)

- During an `extract` operation, the extracted file contents did not go into `stdout` as expected when `--dst-entry=-`, but into a file named “-” instead. (Bug #30451238)

- Sometimes, after receiving a signal 6 due to some errors, `mysqlbackup` quit without printing any error messages. (Bug #30423128)

- `mysqlbackup` sometimes hung during a backup when redo log archiving had been enabled. It was due to the way `mysqlbackup` switched between reading the redo log files and the redo log archive, which has been corrected by this fix. (Bug #30387689)

- If a tablespace on a server was deleted in the middle of a backup process and then restored before the end of the process so that the backup included the table and was successful, `mysqlbackup` still reported in the `mysql.backup_history` table that the backup failed. (Bug #30340161)

- At restart, a restored server sometimes gave the warning `Doublewrite page ### for {space: ###, page_no:###} could not be restored`. This happened because the doublewrite buffer, being restored from the backed up server, contained pages that were no longer relevant. With this fix, the doublewrite buffer in the backup was cleared during the backup process, so that it is no longer restored. (Bug #30286862)

- When OpenSSL 1.1.1 was used for connecting `mysqlbackup` to the server and the `--tls-version` option was not specified, TLSv1.3 was used, but the output of `mysqlbackup` indicated it was using TLSv1.2. (Bug #30268505)

- Backups for a server using the `keyring_file` or `keyring_encrypted_file` plugin failed with the error `Opening of file master_keyring_kef failed` if the keyring file was located in the server’s data directory. This was because in that case, the plugin returned a path for the keyring file that was relative to the data directory, with which `mysqlbackup` could not locate the file. With this fix, the plugin returns a full path of the file to `mysqlbackup`. (Bug #30238406)

- `mysqlbackup` quit unexpectedly when the `validate` command was issued without specifying any command options. With this fix, `mysqlbackup` quits gracefully in the situation by throwing a proper error. (Bug #30204114)

- The default value of the `--page-reread-time` option was 0 millisecond, instead of 100 milliseconds as documented in the manual. (Bug #30036877)

- When a `copy-back-and-apply-log` operation was applied on a prepared backup, the warning `Apply-log operation has already been done on that backup` appeared twice in the output of `mysqlbackup`. (Bug #29941423)

- A backup sometimes failed with `mysqlbackup` reporting that an undo log file looked corrupted when the system variable `innodb_undo_log_encrypt` had been set to `ON` on the server. It was because the encryption information had not yet propagated to the undo log file header when the file...
MySQL Enterprise Backup 8.0.18 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.18. For earlier versions of MySQL 8.0, use the MySQL Enterprise Backup version with the same version number as the server. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.

**Functionality Added or Changed**

- Close to the end of a backup, `mysqlbackup` applied the `FLUSH TABLES tbl_name [, tbl_name] ... WITH READ LOCK` statement on all non-InnoDB tables. With this fix, the statement is not applied to tables that are not included in a partial backup, thus avoiding unnecessary locks for some tables. (Bug #29873048)

- The International Components for Unicode (ICU) library for regular expressions, used by MySQL Server 8.0, has now replaced the RE2 library as the library used by MySQL Enterprise Backup for handling regular expressions.

  This change also removes a memory leak associated with the use of the Partial Backup and Restore Options. (Bug #27374240, Bug #29840790)

- The `image-to-backup-dir` command is now an alias for the `extract` command.

- Two new options, `--compression-algorithms` and `--zstd-compression-level`, have been introduced for configuring compression for server connections. See Connection Options and Command Options for Connection Compression for details.

- MySQL Enterprise Backup now supports a faster way to create incremental backups by using the page tracking functionality on MySQL Servers. To use this new feature, set `--incremental=page-track`. See Incremental Backup Using Page Tracking for details.

- The `--uncompress` option is now supported for the `extract` operation: Files from a compressed single-file backup can now be extracted and uncompressed with a single command.

**Bugs Fixed**

- Backups on a Group Replication primary node failed when `mysqlbackup` tried to upgrade the `mysql.backup_history` table on the node (see Backup History Table Update for details),
and there was a problem causing the node to switch to `super_read_only` mode. With this fix, `mysqlbackup` checks if the node is in `super_read_only` mode during the `backup_history` table upgrade and, if so, skips the steps in the upgrade process that caused the problem. (Bug #30065637)

- After a partial backup created with the `--only-innodb` option was restored, the server failed to start due to an assertion error if the `clone plugin` or `thread pool plugin` was loaded during the server startup. It was because the backup created with the `--only-innodb` option did not include the performance schema, and the plugin was looking for the data folder for the performance schema when the restored server was being started. This fix prevents the problem by having `mysqlbackup` create an empty `performance_schema` folder under the backup's data directory when creating backups with `--only-innodb`. (Bug #29999075)

- Wrong value for the `--safe-slave-backup-timeout` option was printed when the `--help` option was used with `mysqlbackup`, if `--safe-slave-backup-timeout` was set with an unsigned integer. (Bug #29994968)

- A backup failed for a server containing encrypted InnoDB tables if the server was started with `--skip-grant-tables`. It was because the server only accepted connection through a Unix socket in that case, but `mysqlbackup` did not pass the value of `--keyring-migration-socket` to the keyring migration server; this fix makes `mysqlbackup` pass the option whenever it connects to a server to be backed up with a socket. (Bug #29954367)

- When restoring a full backup of a slave server in a replication setup, `mysqlbackup` gave warnings for missing relay log information file and master information file even if those files were never used on the backed-up server. This fix removes the unnecessary warnings. (Bug #29941160)

- When the `--compress-method` option was used at an `apply-log` operation, the operation failed, and the backup was corrupted. With this fix, `mysqlbackup` throws an error and quits whenever any compression options are used with `apply-log`. (Bug #29941117)

- At the end of any apply-log operation, `mysqlbackup` printed the message `INFO: Backup was originally taken with the --include regexp option` whenever a partial backup option that made use of regular expressions was used during the backup. With this fix, under the situation, `mysqlbackup` simply mentions in the log message that the backup is a partial one. (Bug #29872975)

- When a base backup was restored with `--skip-relaylog` but a subsequent incremental image backup restore did not use the option, the relay log files were copied from the incremental image backup onto the server. With this fix, an incremental image restore, with or without using the `--skip-relaylog` option, does not restore the relay log files whenever its base backup did not restore the relay log. (Bug #29864964)

- A memory leak that occurred with backups involving encrypted redo log has been removed. (Bug #29841265)

- A memory leak caused by the use of the `--optimistic-time` option has been removed. (Bug #29841031)

- When a base backup was restored with `--skip-binlog` but a subsequent incremental image backup restore did not use the option, the binary log files were copied from the incremental image backup onto the server, and the server could not be started because of the incomplete binary log. With this fix, an incremental image restore, with or without using the `--skip-binlog` option, does not restore the binary log files whenever its base backup did not restore the binary log. (Bug #29802632)

- `mysqlbackup` hung when performing an incremental or compressed image backup when the `--limit-memory` option was set to a low value. This fix prevents the problem by adjusting the way data buffers are allocated in the situation. (Bug #29773223)

- When restoring an incremental backup created with the `--skip-final-rescan` option, the external tablespaces on the target server got deleted. It was due to the missing entries for the
external tablespaces in the incremental backup's `tablespace_tracker` file, and this fix corrects the issue. (Bug #29513642)

- Attempts to extract binary log files from a compressed backup failed with a `No such file or directory` error when the size of the binary log files on the backed up server was greater than 16MB. (Bug #28787312)

### Changes in MySQL Enterprise Backup 8.0.17 (2019-07-22, General Availability)

MySQL Enterprise Backup 8.0.17 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.17. For earlier versions of MySQL 8.0, use the MySQL Enterprise Backup version with the same version number as the server. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.

- **Functionality Added or Changed**
- **Bugs Fixed**

#### Functionality Added or Changed

- Before the current release, when backing up a server that used the `keyring_okv` plugin for InnoDB table encryption, `mysqlbackup` must not be run by a sudo user of its operating system. This restriction has now been removed. (Bug #29020232)

- The `--datadir` option is no longer required for restoring a TTS backup. If the option is specified and its value does not match with that of the target server, the restore will be aborted. (Bug #28546760)

- The `--incremental-base` option now accepts a new value, `history:last_full_backup`, which makes it easy to create a differential backup. See the description of `--incremental-base` for details.

- To avoid `mysqlbackup` failing to catch up with the growing redo log during a backup operation and missing redo log data, `mysqlbackup` now utilizes redo log archiving, a new feature available on MySQL Server 8.0.17. Redo log archiving can be disabled using the new `mysqlbackup` option `--no-redo-log-archive`. See [Backing up Using Redo Log Archiving](#) for details.

- `mysqlbackup` now supports encrypted InnoDB redo logs. The encrypted redo tablespaces are handled the same way as the encrypted tablespaces for InnoDB tables. See [Working with Encrypted InnoDB Tablespaces](#) for details.

#### Bugs Fixed

- A backup failed when the value of the server's system variable `innodb_undo_directory` contained in itself the file path for the server's data directory. It was due to a mishandling of the file path prefix of the undo tablespace directory by `mysqlbackup`, which has been corrected by this fix. (Bug #29849566)

- Restore of an incremental backup failed if its base full backup had been restored with the `--skip-binlog` option. (Bug #29757701)

- When the binary log on the server was more than one level below the data directory on the directory tree, `mysqlbackup` failed to copy the binary log into a backup. This was due to an error on parsing the path of the binary log directory, which has been corrected by this fix. (Bug #29710251)

- If a relative path was used with the `backup_innodb_data_home_dir` option when backing up a server, the whole directory specified by the option was being copied into the target server's data directory during a restore of the backup. Not only was that not the expected behavior of `mysqlbackup`, but it also made subsequent backups of the server failed when the same argument for `backup_innodb_data_home_dir` was used again. (Bug #29613025)
• During a backup operation, when a table or database name contains a slash (/), `mysqlbackup` always treated the corresponding tablespace as an external tablespace; if that was not actually the case, restore for the backup was going to fail. With this fix, `mysqlbackup` checks if the tablespace is really external and then handles it appropriately. (Bug #29472939)

• External undo tablespaces were missing after a restore of a backup directory extracted from a backup image using the `image-to-backup-dir` command. It was because of the mishandling of the tracker file for external tablespaces by the `image-to-backup-dir` command, which has been corrected by this fix. (Bug #29401027)

• While no upgrade path exists between MySQL Enterprise Backup 4.1 and 8.0, an attempt to update the `mysqlbackup` package from release 4.1.3 to 8.0.16 on Ubuntu failed with the complaint that the update tried to overwrite the installation directory for `mysqlbackup`. With this fix, package conflict information has been added so that at the same attempt, the old package is uninstalled (with the user's consent) before the new package is installed. (Bug #29314267)

• When using MySQL Enterprise Backup 8.0 to back up MySQL Server 5.7, an error was thrown, and the error message suggested a wrong version of MySQL Enterprise Backup to be used for the Server. With this fix, the appropriate version of MySQL Enterprise Backup is suggested. (Bug #29195233)

• When backing up a server that used the `keyring_okv` plugin for InnoDB table encryption, if the `--host`, `--user`, and `--port` options were not specified with the `mysqlbackup` command via the command line or a configuration file, the backup failed. It was because in that case, `mysqlbackup` had no values for those options it could use to connect to the server that took care of keyring operations. With this fix, default values are now set, so that `mysqlbackup` connects to the server on `localhost` as `root` and on port `3306` for keyring operations when those options are not specified. (Bug #29015923)

• A `copy-back-and-apply-log` operation for a compressed backup created using the `--backup_innodb_data_home_dir` option with a relative file path terminated with signal 6. (Bug #28967141)

• `mysqlbackup` hung during a restore operation when the backup contained more than a hundred InnoDB tablespaces. (Bug #28884254, Bug #29674585)

• A restore operation for a compressed backup failed with an `unexpected end of file` error when the backup was created using `--compress-method=zlib` and the `innodb_page_size` was smaller than 16KB. (Bug #28623215)

• A backup created on an EL7 platform containing InnoDB tables encrypted with MySQL Enterprise Transparent Data Encryption (TDE) could not be restored to a server on a Solaris platform. It was because in this case, the source and the target platforms of the backup used different byte ordering formats, causing difficulties in loading the encryption key from the backup. This fix prevents the issue by adding detection and conversion utilities for different system architectures. (Bug #28569367)

• Using the `--uncompress` option for restoring a backup not created with the `--compress` option caused the operation to fail with the error `No such file or directory`. With this fix, the proper error is thrown in the situation. (Bug #28334690)

• A backup failed with the error `Log scan was only able to reach...` when there was a large amount of DML activities occurring in parallel on the server that was being backed up. (Bug #27555969)

• During the InnoDB buffer pool dump in a backup operation, `mysqlbackup` sometimes reported failure for the dump while it was actually still in progress. The fix prevents the problem by improving the way `mysqlbackup` checks for the status of the dump. (Bug #27185901)
Changes in MySQL Enterprise Backup 8.0.16 (2019-04-25, General Availability)

MySQL Enterprise Backup 8.0.16 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.16. For earlier versions of MySQL 8.0, use the MySQL Enterprise Backup version with the same version number as the server. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.

- Functionality Added or Changed
- Bugs Fixed

Functionality Added or Changed

- `mysqlbackup` now supports encrypted InnoDB undo logs. The encrypted undo tablespaces are handled the same way as the encrypted tablespaces for InnoDB tables. See Working with Encrypted InnoDB Tablespaces for details.

- Near the end of the backup process, instead of locking the whole server instance for a brief period of time, `mysqlbackup` now applies these locks consecutively:
  1. A backup lock on the server instance, which blocks DDLs (except those on user-created temporary tables), but not DMLs on InnoDB tables.
  2. A `FLUSH TABLES tbl_name [, tbl_name] ... WITH READ LOCK` operation on all non-InnoDB tables, for copying the relevant ones among them into the backup. This step is skipped if no user-created non-InnoDB tables exist.
  3. A brief blocking of logging activities on the server, for collecting logging-related information.

See The Backup Process for details. The removal of the lock on the whole server instance reduces disruption to the database service by the backup operation.

**Important**

The change requires that the `BACKUP_ADMIN` and `SELECT` privileges on all tables be granted to the user by which `mysqlbackup` connects to the server (the `BACKUP_ADMIN` privilege is automatically granted to users with the `RELOAD` privilege when an in-place upgrade to MySQL Server 8.0 from an earlier version is performed).

- `mysqlbackup` now supports dynamic changes to undo tablespaces on the server being backed up. During a restore, the default undo tablespaces, as well as any non-default undo tablespaces resided in the backed-up server's data directory, are restored to the location pointed to by the `mysqlbackup` option `--innodb_undo_directory`. Non-default, external undo tablespaces are restored to the locations they were found on the backed-up server. See undo log files for details.

- In addition to the requirement that the target data directory for a restore specified by the `--datadir` option must be non-existent or empty, `mysqlbackup` now enforces the same rule for the `--innodb_data_home_dir`, `--innodb_log_group_home_dir`, and `--innodb_undo_directory` options (the `--force` option cannot be used to override the requirement on the three options).

Bugs Fixed

- Zip packages of `mysqlbackup` contained duplicate files, which have now been removed. (Bug #29497272, Bug #94683)
MySQL Enterprise Backup 8.0 Release Notes

- `mysqlbackup` might quit unexpectedly if it lost its connection to the server at the middle of a backup operation. With this fix, `mysqlbackup` exits gracefully in the situation after throwing the appropriate errors. (Bug #29376006)

- Restore of an incremental backup failed if, on the server, some binary log files had been purged in between the times the incremental backup and its base backup were made. (Bug #29306026)

- A backup failed for `mysqlbackup` if the path given by the `--backup-dir` option was of the Universal Naming Convention (UNC) format, as `mysqlbackup` failed to create the backup directory then. (Bug #29190803)

- A `mysqlbackup` operation failed when the `backup-image` option was supplied to `mysqlbackup` in a configuration file instead of on the command line. (Bug #29157495)

- A restore operation for a TTS backup failed if the backed-up server has ANSI_QUOTES as one of its SQL modes, as specified in its system variable `sql-mode`. (Bug #28979134)

- `mysqlbackup` quit unexpectedly when the `--password` option was used twice, with no argument given at the second time, in a `mysqlbackup` command that was itself invalid aside from the use of the `--password` option. (Bug #28894102)

- When a compressed incremental folder backup was restored, the binary log files created in between the times of the base and the incremental backups were not copied onto the target server. (Bug #28773998)

- A restore operation for a database containing encrypted InnoDB tables failed without returning a proper error message when the the `--encrypt-password` option was not used in the `mysqlbackup` command. (Bug #28773077)

- After restoring an incremental backup taken from a MySQL Community Server with encrypted InnoDB tables, the keyring file of the restored server became corrupted, so the server could not be started. (Bug #28422191)

- A restore operation could corrupt a backup when, by mistake, a user specified the source directory to become the target directory for restoring some files (for example, specifying what was the backup's `--backup_innodb_data_home_dir` value as the restore's `--innodb_data_home_dir` value). This fix prevents the problem by having `mysqlbackup` throw an error when the command options make the source and target file paths the same for any file copying during a restore. (Bug #28376873)

- While MySQL Server interprets the system variable setting `--innodb_checksum_algorithm=0` to mean `--innodb_checksum_algorithm=crc32`, a `mysqlbackup` operation (except for backup) failed when `--innodb_checksum_algorithm=0` was set as a configuration option on the backed up server. With this fix, `mysqlbackup` now takes `--innodb_checksum_algorithm=0` as valid and interprets it as `--innodb_checksum_algorithm=crc32`. (Bug #28295519)

- A restore operation failed with the error that the binary log index file could not be opened if the binary log base name for the backed-up server was a substring of the word “index”. (Bug #28127023)

- `mysqlbackup` tried to connect to a remote host specified by the `--host` option, while it was supposed to ignore the option (see Connection Options for details). With this fix, the option is now ignored.

```
Note
As a side-effect of this change, on Unix-like platforms, `mysqlbackup` command that used the `--host` option now needs to use the `--protocol=TCP` option to indicate explicitly that `mysqlbackup` is to connect to localhost using TCP/IP.
```

(Bug #25911987)
• When the option --no-locking was used during a backup operation, the backup sometimes failed with mysqlbackup complaining that the highest LSN was larger in a copied page than on the backed-up server. It was because mysqlbackup did not perform a log flushing before copying the redo log when the option was used. With this fix, log flushing was always performed to prevent the error. (Bug #25412655)

Changes in MySQL Enterprise Backup 8.0.15 (2019-02-01, General Availability)
MySQL Enterprise Backup 8.0.15 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.15. For earlier versions of MySQL 8.0, use the MySQL Enterprise Backup version with the same version number as the server. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.

This release contains no functional changes and is published to align version number with the MySQL Server 8.0.15 release.

Changes in MySQL Enterprise Backup 8.0.14 (2019-01-21, General Availability)
MySQL Enterprise Backup 8.0.14 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.14. For earlier versions of MySQL 8.0, use the MySQL Enterprise Backup version with the same version number as the server. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.

• Functionality Added or Changed

• Bugs Fixed

Functionality Added or Changed

• mysqlbackup now supports encrypted binary and relay log. See descriptions for the undo log files for details.

• mysqlbackup now supports the --ssl-fips-mode option, which controls whether mysqlbackup operates in FIPS mode. See FIPS Support for details.

Bugs Fixed

• When working with a Group Replication cluster, mysqlbackup quit unexpectedly near the end of a backup operation when, in order to write to the backup_history table, it tried to connect with an unencrypted connection to one of the nodes on which the backup user had not logged on before. It was because, as a user created with the caching_sha2_password plugin (enabled by default on MySQL 8.0 servers), the backup user must log on with an encrypted connection when it connects to the server for the first time; the attempt to log on thus failed, and mysqlbackup could not handle the failure. With this fix, at such failures, mysqlbackup quits gracefully with the warning that the backup operation is finished without updates to the backup history. (Bug #28893180)

• An apply-incremental-backup operation failed with an error (RDR1 ERROR: Unable to remove relaylog files from full backup) when the incremental backup was created with the --compress option. (Bug #28366241)

• mysqlbackup quit unexpectedly during an apply-incremental-backup operation if the backed up server had been started using relative paths for --datadir and --log-bin. (Bug #28334521)

• Attempts to restore a backup of a MySQL 5.7 Server to a MySQL 8.0 Server resulted in a strange error message (Server_version is not obtained). With this fix, mysqlbackup now indicates that the operation is not supported. For related information, see Restoring a Backup with a Database Upgrade or Downgrade. (Bug #27952379)
MySQL Enterprise Backup 8.0 Release Notes

• After restoring an incremental folder backup and putting its binary log at a specified location different from that for the base backup, the older binary log files of the base backup were not removed by mysqlbackup. (Bug #27890472)

• mysqlbackup quit unexpectedly when backing up a MySQL Server of release 8.0.12 or later for the first time if the ALTER privilege on the mysql_backup_history_new table had not been granted to the MySQL user with which mysqlbackup connected to the server. With this fix, mysqlbackup quits gracefully in the situation after throwing the proper error.

Also, the CREATE, INSERT, and DROP privileges on mysql.backup_history_old and CREATE, INSERT, DROP, and ALTER privileges on mysql.backup_history_new are now required only for backing up for the first time a MySQL Server that has been upgraded from 8.0.11 or earlier and has been backed up by MySQL Enterprise Backup before. (Bug #27879530, Bug #28546256)

• Partial backups sometimes failed because full-text index files had their file names matched by the regular expression provided by the --include-tables option, and the files were then handled as ordinary tablespace files by mysqlbackup. With this fix, mysqlbackup excludes any full-text index files from backups. (Bug #25044900)

Changes in MySQL Enterprise Backup 8.0.13 (2018-10-22, General Availability)

MySQL Enterprise Backup 8.0.13 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.13. For earlier versions of MySQL 8.0, use the MySQL Enterprise Backup version with the same version number as the server. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.

• Functionality Added or Changed

• mysqlbackup now supports backup compression (the use of the --compress and --uncompress options) for incremental backups (except for incremental backups created with the --incremental-with-redo-log-only option).

• mysqlbackup now supports transparent page compression for InnoDB tables. The support is enabled by setting the mysqlbackup option --compress-method=punch-hole; see description for the option for details.

• Bugs Fixed

• Restoring an incremental backup image using the copy-back-and-apply-log command failed with mysqlbackup complaining that the server repository configuration (including, e.g., value of innodb_data_file_path) was unknown for the target server. With this fix, mysqlbackup gets the required information from the backup-my.cnf file already restored with the base backup of the incremental backup. (Bug #28411028)

References: This issue is a regression of: Bug #27429244.

• mysqlbackup hung when a backup operation failed due to a full disk. With this fix, mysqlbackup quits gracefully in the situation by throwing an error. (Bug #28399821)

• During an --apply-incremental-backup operation, mysqlbackup attempted to delete the binary log of the backed-up, running server. (Bug #28377502)

• On FreeBSD platforms, using the --show-progress option did not make mysqlbackup print progress reports. (Bug #28350122)
• A `mysqlbackup` operation on an image stored on an OpenStack cloud storage service sometimes failed with a segmentation fault or a bad URL error. It was because of a race condition caused by an uninitiated variable, which has been eliminated by this fix. (Bug #28189239, Bug #28183729)

• Backups for databases with encrypted InnoDB tables failed when the `--compress` option was used. (Bug #28177466)

• The Windows version of MySQL Enterprise Backup did not display its build ID when invoked. (Bug #27916702)

• A `mysqlbackup` operation on an image stored on an OpenStack cloud storage service failed with a `401 Unauthorized` error when the operation took a long time and the authentication token for the cloud access expired. With this fix, a separate thread in `mysqlbackup` requests a new token from the OpenStack cloud service in that situation, so that the operation can continue. (Bug #27893174)

• When the `--show-progress=table` option was used, `mysqlbackup` gave a warning in the error log on an aborted connection to the server near the end of the operation. It was because the connection to the server for writing to the `backup_progress` table had remained open. With this fix, the connection is properly closed after the `mysqlbackup` operation is finished. (Bug #27647283)

• When an incremental backup was restored without using the `--log-bin` option, the binary log was not restored to its original location on the backed up server, but to the location specified by `--log-bin` earlier during the restore of the base backup. The same occurred for relay logs of incremental backups for slaves when the `--relay-log` option was not used. (Bug #27545745)

• If, when a backup was in progress and `mysqlbackup` was reading the binary log (or the relay log) index file and the server tried to modify the index file (because, for example, a log flush or log purge just took place), the binary logging (or relay logging) stopped; the server also quit unexpectedly on Windows platforms. It was because `mysqlbackup` did not handle well the file sharing violation. With this fix, `mysqlbackup` now reads the index file using the local file system API, which handles the file sharing violation gracefully; also, `mysqlbackup` now copies the index file into its buffer and then closes it, instead of keeping it open for long, so the server can modify or delete the index file more easily. (Bug #22914974, Bug #26047119)

Changes in MySQL Enterprise Backup 8.0.12 (2018-07-27, General Availability)

MySQL Enterprise Backup 8.0.12 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.12. For earlier versions of MySQL 8.0, use the MySQL Enterprise Backup version with the same version number as the server. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.

• Functionality Added or Changed

• Bugs Fixed

Functionality Added or Changed

• Important Change: Starting from release 8.0.12, the storage engine of the `mysql.backup_history` table on a backed-up server has switched from CSV to InnoDB, and a new column for server UUIDs has been added to the table. See Backup History Table Update for the new user privileges required by `mysqlbackup` due to this change.

• Important Change: When working with a Group Replication setup, `mysqlbackup` now makes the backup history available to all members of the server group by making sure that the `backup_history` table is updated on a primary node after each `mysqlbackup` operation. See Using MySQL Enterprise Backup with Group Replication for details.

With the implementation of this feature, the new user privilege of SELECT on `performance_schema.replication_group_members` is now required by `mysqlbackup` to
work with any server, even when it does not belong to a Group Replication setup. See Grant MySQL Privileges to Backup Administrator for details.

- Version information for mysqlbackup is now printed to the stdout instead of the stderr stream when the --version or --help option is used. (Bug #27253989)

- OAuth is now supported for Oracle Cloud Storage client authentication. Two new options, --cloud-storage-url and --cloud-oauth-token, have been introduced for the purpose. See Cloud Storage Options for details.

**Bugs Fixed**

- Backups for a server failed when it had ANSI_QUOTES in its values for sql_mode. (Bug #27939774)

- The maximum value that could be set for the --safe-slave-backup-timeout option was 2700 (seconds), which automatically replaced any larger value. With this fix, there is no longer a maximum limit, even though a high value is not recommended; see the description of --safe-slave-backup-timeout for details. (Bug #27883020)

- Restoring an incrementation backup on top of a data directory restored using a compressed backup failed. It was because mysqlbackup did not set is_compressed=0 in the backup_variables.txt file inside the restored data directory. (Bug #27787988)

- When mysqlbackup performed sanity checks on InnoDB tablespaces and found a space ID mismatch for an FSP header and a page header, the name of the problematic tablespace was not given in the error report. (Bug #27752703)

- If an ALTER TABLE statement was executed on the server before an incremental backup was taken, a server restored with the backup on which the incremental backup was applied (using the apply-incremental-backup command) could not be started, as the .ibd file of the altered table was missing in the restored data. (Bug #27735134)

- After a server has been restored using an incremental backup created with the --incremental-with-redo-log-only option, it could not be started. (Bug #27722525)

- mysqlbackup issued a warning whenever the number of files specified in the system variable innodb_data_file_path of the server to be backed up exceeded 100. With this fix, a warning is issued only if the number of InnoDB data files to be opened exceeds the number specified by the system variable innodb_open_files. (Bug #27701402)

- Backups failed for a server that had been started with a value for --innodb_log_file_size different from the one the server was initialized with. (Bug #27571663)

- An apply-incremental-backup operation failed when individual tablespaces with relative file paths were involved. (Bug #27278876)

- In a Group Replication setting for MySQL servers, when changes were made to one group member and a backup was taken on another, the relay log for the replication applier was missing from the backup, so that the restored server could not be started. (Bug #25534078)

- When a compressed backup was restored with the --innodb_data_home_dir option pointing to a restore location outside of the data directory, .ibd files were still being copied into the data directory, causing an exception to be thrown at the attempt to start the restored server. (Bug #24626986)

**Changes in MySQL Enterprise Backup 8.0.11 (2018-04-19, General Availability)**

MySQL Enterprise Backup 8.0.11 is the latest release for MySQL Enterprise Backup. It only supports MySQL Server 8.0.11. For MySQL server 5.7, please use MySQL Enterprise Backup 4.1, and for MySQL Server 5.6 and 5.5, please use MySQL Enterprise Backup 3.12.
Functionality Added or Changed

- Offline backups are no longer supported by `mysqlbackup`. As a result, a number of options used for offline backup operations have been removed. See What's New in MySQL Enterprise Backup 8.0? for details. (Bug #27429244)

- The server option `--secure-auth`, deprecated since MySQL 5.7.5, is no longer supported by `mysqlbackup`. (Bug #27265328)

- Servers’ use of the `keyring_encrypted_file` and `keyring_aws` plugins is now supported. See Working with Encrypted InnoDB Tablespaces for details. (Bug #27127898)

- Information on the executed GTIDs is now included in the `mysqlbackup` output and the backup log when the backed-up server has GTIDs enabled. (Bug #25978803)

- The binary log for a backed-up server, instead of being restored always to the data directory on the target server, is now restored by default to the same location it was found on the backed-up server. It can also be restored to a different location specified with the new `--log-bin` option. (Bug #25141738, Bug #83927)

- The relay log for a backed-up slave server, instead of being restored always to the data directory on the target slave server, is now restored by default to the same location it was found on the backed-up slave server. It can also be restored to a different location specified with the new `--relay-log` option. (Bug #25141738, Bug #83927)

- The `backup_history` table now includes a `server_uuid` column, which stores the value of the `server_uuid` of the backed up server.

- The options `--ssl` and `--ssl-verify-server-cert`, already deprecated in MySQL Enterprise Backup 4.1, have now been removed. Use the `--ssl-mode` option instead to configure the security mode of your connection to the server.

- MySQL Enterprise Firewall is now supported.

- A new option, `--tls-version`, specifies the protocols `mysqlbackup` permits for encrypted connections to MySQL servers.

- A file now tracks information of external tablespaces for a backup or restore in JSON format. See description for `tablespace_tracker` in Types of Files in a Backup for details.

- `mysqlbackup` could not restore the auto increment values in tables and the corruptions flags for indexes onto a server. The tasks are now made possible by having `mysqlbackup` copying onto the target server blocks of redo logs that cover the duration from the latest checkpoint up to the backup end time, so that the restored server can, during the recovery phase at its first start, restore the auto increment values and the corruption flags using those blocks.

- The buffer size for cloud transfers can now be specified using the new `--cloud-buffer-size` option. See Cloud Storage Options for details.

- HTTP Basic Authentication and non-chunked transfer are now supported for backup and restore using OpenStack Swift-compatible object storage services. Two new options, `--cloud-basicauth-url` and `--cloud-chunked-transfer`, have been introduced for these purposes. See Cloud Storage Options for details.

Bugs Fixed

- After restoring a full backup, if the following restore of an incremental backup changed the restore location of the undo log, either `mysqlbackup` hung, or the restored server failed to start. With this
fix, `mysqlbackup` quits with a proper error (“Undo tablespace in the base backup not found”) in the situation.

Users should make sure the undo log location does not change between successive restores of a full and an incremental backups, or of two incremental backups. (Bug #27530916)

- When restoring an incremental backup containing encrypted InnoDB tables to a MySQL Community Server, the password provided to `mysqlbackup` with the `--encrypt-password` option was never validated, so that when the wrong password was given, the restore still succeeded, but the restored server could not be started. With this fix, `mysqlbackup` throws an error and stops the restore if the password is wrong. (Bug #27483449)

- `mysqlbackup` failed to backup to an Amazon S3 cloud storage. (Bug #27231229)

- An `apply-incremental-backup` operation corrupted the non-InnoDB files in its target backup when the sizes of those files are smaller in the incremental backup than in the target backup. (Bug #27001934)

- `mysqlbackup` could not establish a connection to the server with a Unix socket using the option `--protocol=SOCKET`. (Bug #26977679)

- A number of memory leaks were observed when running `mysqlbackup`. They have now been fixed. (Bug #26495834, Bug #26373259, Bug #26093563, Bug #26423820, Bug #26497245)

- After applying an incremental backup created with the `--incremental-with-redo-log-only` option to a full backup, the full backup's binary log became corrupted. (Bug #26403452)

- On macOS, `mysqlbackup` failed to determine the relay log file name correctly and thus could not back up the relay log for a slave server. (Bug #25574605)

- `mysqlbackup` only accepts values for `--ssl-mode` in upper case. With this fix, the option's value has become case insensitive. (Bug #25548088)

Index

Symbols

- `--backup-image`, 11
- `--compress`, 2
- `--compress-method`, 7
- `--dst-entry`, 4
- `--help`, 15
- `--include`, 2, 7
- `--include-tables`, 13
- `--incremental-base`, 9
- `--incremental-with-redo-log-only`, 15
- `--innodb-undo-directory`, 9
- `--innodb_checksum_algorithm`, 11
- `--innodb_log_file_size`, 15
- `--limit-memory`, 7
- `--no-locking`, 11
- `--only-innodb`, 7
- `--optimistic-busy-tables`, 2
- `--optimistic-time`, 7
- `--page-reread-time`, 4
- `--safe-slave-backup-timeout`, 7
- `--show-progress`, 14
- `--skip-binlog`, 7
- `--skip-final-rescan`, 7
- `--skip-relaylog`, 7
--skip-unused-pages, 2
--sll-fips-mode, 13
--src-entry, 2
--uncompress, 7, 9, 15
--use-tts, 2
--version, 15

A
ALTER TABLE, 15
Amazon S3, 16
ANSI_QUOTES, 11, 15
apply-incremental-backup, 13, 15, 16
apply-log, 7
auto increment values, 16

B
backticks, 4
backup-and-apply-log, 2
backup-to-image, 4
backupdir, 4
backup_history table, 4, 15
backup_innodb_data_home_dir, 9
binary log, 4, 7, 9, 11, 14, 16

C
cclone plugin, 7
ccloud operations, 14
ccloud-basicauth-url, 16
ccloud-buffer-size, 16
ccloud-chunked-transfer, 16
ccloud-oauth-token, 15
ccloud-storage-url, 15
ccompressed backups, 15
compressed backup, 2, 7, 9
compressed incremental backup, 14
compression, 14, 15
cconfiguration file, 11
copy-back-and-apply-log, 2, 4, 9
corruption flags for indexes, 16
coy-back-and-apply-log, 2

D
differential backup, 9
disk full, 14
doublewrite buffer, 4

E
enencrypted binary log, 4
enencrypted InnoDB tables, 9, 14, 16
enencrypted redo logs, 9
enencrypted tables, 2
enencrypted undo log, 4
enencrypted undo logs, 11
everwarded tablespace, 16
extract, 2, 4, 7
MySQL Enterprise Backup 8.0 Release Notes

F
failed connection, 4
FLSUH TABLES, 4

G
Group Replication, 13, 15
GTID, 16

I
image-to-backup-dir, 7
Important Change, 4, 15
incremental backup, 2, 4, 7, 9, 13, 14, 15, 16
incremental-with-redo-log-only, 16
InnoDB cluster, 13
InnoDB table encryption, 7
innodb_data_home, 11
innodb_log_group_home_dir, 11
innodb_open_files, 15
innodb_undo_directory, 11
innodb_undo_log_encrypt, 4
innodeb_data_file_path, 14
innodn_data_file_path, 15
innodn_data_home_dir, 15
installing, 9

K
keyring, 2
keyring_encrypted_file plugin, 4
keyring_file plugin, 4
keyring_file_data, 4
keyring_okv, 9

L
locks, 11
losing connection, 11

M
master info file, 7
MySQL Enterprise Firewall, 16

N
no-history-logging, 2
non-InnoDB tables, 7

O
OAuth, 15
offline backup, 16
OpenStack, 14

P
packaging, 9, 11
page tracking, 7
parakkkek DDL, 4
partial backup, 7
partial backups, 7
partial restore, 2
privileges, 4, 13
progress report, 2
PURGE BINARY LOG, 11

R
redo log archiving, 4, 9
redo log encryption, 7
relay log, 14, 16
relaylog info file, 7
restore, 4

S
secure-auth, 16
server upgrade, 13
server_uuid, 16
signal, 4
signal 6, 4
sql_mode, 11, 15
src-entry, 4
ssl-mode, 16
stack trace, 4

T
tablespace ID check, 15
tablespace_tracker, 16
tablespace_tracker file, 2
TLS version, 4
tls-version, 16
transparent page compression, 14

U
UNC paths, 11
undo log, 9, 16
undo tablespace, 2, 4
undo tablespaces, 11

V
validate, 4
versions, 9

W
warnings, 4