
MySQL Enterprise Backup 3.10 Release Notes

Abstract

This document lists the changes to the MySQL Enterprise Backup 3.10 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 3.10 documentation, see the [MySQL Enterprise Backup User's Guide \(Version 3.10.2\)](#).

For legal information, see the [Legal Notices](#).

Document generated on: 2015-07-03 (revision: 6424)

Table of Contents

| | |
|--|---|
| Preface and Legal Notices | 1 |
| Changes in MySQL Enterprise Backup 3.10.2 (2014-07-01) | 2 |
| Changes in MySQL Enterprise Backup 3.10.1 (2014-04-16) | 3 |
| Changes in MySQL Enterprise Backup 3.10.0 (2014-03-10) | 4 |

Preface and Legal Notices

This document lists the changes to the MySQL Enterprise Backup 3.10 product, beginning with the most recent release.

Legal Notices

Copyright © 2003, 2014, 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 software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software 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 which

may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. MySQL is a trademark of Oracle Corporation and/or its affiliates, and shall not be used without Oracle's express written authorization. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on 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. 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.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this material is subject to the terms and conditions of your Oracle Software License and Service Agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle or as specifically provided below. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

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.

For more information on the terms of this license, or for details on how the MySQL documentation is built and produced, please visit [MySQL Contact & Questions](#).

For additional licensing information, including licenses for third-party libraries used by MySQL products, see [Preface and Legal Notices](#).

For help with using MySQL, please visit either the [MySQL Forums](#) or [MySQL Mailing Lists](#) where you can discuss your issues with other MySQL users.

For additional documentation on MySQL products, including translations of the documentation into other languages, and downloadable versions in variety of formats, including HTML and PDF formats, see the [MySQL Documentation Library](#).

Changes in MySQL Enterprise Backup 3.10.2 (2014-07-01)

- [Security Notes \[3\]](#)
- [Functionality Added or Changed \[3\]](#)
- [Bugs Fixed \[3\]](#)

Security Notes

- **Security Fix:** The linked OpenSSL library for MySQL Enterprise Backup 3.10 has been updated from version 1.0.1g to version 1.0.1h. Versions of OpenSSL prior to and including 1.0.1g are reported to be vulnerable to [CVE-2014-0224](#). (CVE-2014-0224)

Functionality Added or Changed

- MySQL Enterprise Backup now supports creation and restoration of single-file backups using a cloud storage. See [Cloud Storage Options](#) for details.

Bugs Fixed

- After a table backed up using the transportable tablespace option (`--use-tts`) was restored to a server, queries on the table did not make use of its indexes. That was because the cardinalities of the indexes were not properly updated after the table's restoration. This fix adds an `ANALYZE TABLE` step towards the end of the restoration process for tables backed up with the `--use-tts` option, in order to update the indexes' cardinalities. (Bug #18682317)
- When cloning a slave for a GTID-enabled server using MySQL Enterprise Backup, the `backup_gtid_executed.sql` script created and stored in the backup directory was not copied onto the slave by the `copy-back-and-apply-log` operation. This fix has the script copied into the data directory of the slave. (Bug #18674861)
- The maximum number of memory buffers that could be created for a `mysqlbackup` operation was hard-coded to be 100, making it impossible to set the number of buffers to a larger value using the `number-of-buffers` option. This fix removes the hard-coded maximum number for buffers. (Bug #18560870)
- `mysqlbackup` threw an error if a table was dropped when the backup process was running. With this fix, the dropped table is ignored (as it does not need to be restored) and `mysqlbackup` finishes without throwing an error. (Bug #18358912, Bug #71865)
- A segmentation fault occurred when a backup image created from a backup directory was restored using the `copy-back-and-apply-log` subcommand. It was because `copy-back-and-apply-log` was not able to extract `backup-my.cnf` from the image and get the value for `innodb_data_file_path`. (Bug #18242586)
- After an `apply-log` operation was performed on a compressed backup (with the `--uncompress` and `--apply-log` options), when a `copy-back-and-apply-log` was applied on the backup, the restored data was inconsistent. That was because the first operation did not delete the compressed, `.ibz` backup file and did not mark the data as uncompressed at the end of the operation. The subsequent `copy-back-and-apply-log` operation then acted on the still existing, raw, compressed file, but thought that an `apply-log` operation had already been performed on it. This fix makes `mysqlbackup` delete the compressed, raw backup file once decompression and `apply-log` are finished and properly mark the backup as uncompressed and up-to-date. (Bug #18005786, Bug #18005732)
- After an incremental backup was applied to a full backup, a second incremental would fail if the same incremental backup directory was used and if the `--incremental-base=dir:directory_path` option was pointing to the full backup's directory. This was because MySQL Enterprise Backup checked the end LSN in the full backup directory against the end LSN in the MySQL history table (which might not have been updated yet) and failed the process when there was a mismatch. This fix removes that check, so user in the described situation can proceed with creating more incremental backups. (Bug #16249018)

Changes in MySQL Enterprise Backup 3.10.1 (2014-04-16)

Heartbleed Bug

- **Security Fix:** MySQL Enterprise Backup 3.10.1 has been updated to use [OpenSSL](#) version 1.0.1g, which has been publicly reported as not vulnerable to [CVE-2014-0160](#). Please see [Oracle Note #1645479.1](#) for further details.

Changes in MySQL Enterprise Backup 3.10.0 (2014-03-10)

- [Functionality Added or Changed](#) [4]
- [Bugs Fixed](#) [4]

Functionality Added or Changed

- When using the `--skip-unused-pages` option for backup operations, MySQL Enterprise Backup now displays, besides the number of pages of data skipped, the total amount of memory saved by using the option. (Bug #15915315)
- The [compression](#) feature of MySQL Enterprise Backup has been enhanced by the addition of two compression algorithms: the LZ4 method (the default for MySQL Enterprise Backup 3.10 and after) and the LZMA method. Because the LZ4 algorithm, though faster, produces larger files than the ZLIB algorithm used in MySQL Enterprise Backup 3.9 and earlier, users of MySQL Enterprise Backup 3.10 will find an increase in the size of the compressed files if they use the default values for the compression options.
- When the `apply-log` or `apply-incremental-backup` operation was performed repeatedly on a backup without using the `--force` option, an error message was thrown. From MySQL Enterprise Backup release 3.10.0 onward, the same action just causes a message to be returned, saying the operation has already been performed.
- In order to enhance security for backed up data, MySQL Enterprise Backup now provides encryption function for single-file backups. See [Encryption for Backups](#) and [Encryption Options](#) for details.
- Backups created with the `--use-tts` option can now be restored in a single step using the `copy-back-and-apply-log` subcommand.
- MySQL Enterprise Backup 3.10 introduces two new options for partial backup: `--include-tables` and `--exclude-tables`. The new options are intended for replacing the older options of `--include`, `--databases`, `--databases-list-file`, and `--only-innodb-with-frm`, which will be deprecated in the near future.
- MySQL Enterprise Backup can now [validate](#) data integrity of a backup directory as well as a backup image file. The validation function in 3.10 has also become more robust, as it tries to verify the checksum value of every data page.

Bugs Fixed

- The `--help` option of the `mysqlbackup` command did not display information correctly. (Bug #18335871)
- `apply-incremental-backup` might fail with a “probable data corruption on page” error if the InnoDB tables being backed up were created in Barracuda format and with the `ROW_FORMAT=COMPRESSED` clause. (Bug #18240566)
- `mysqlbackup` could not read the value of `innodb_data_file_path` from the server when it was more than 1024-character long. It was because `mysqlbackup` could not read the value from the configuration files and relied on the `SHOW VARIABLES` command (which reads no more than 1024 characters) to access the parameter. With this fix, `innodb_data_file_path` can now be read from the configuration files. (Bug #18038409)

- When a compressed backup was restored using the `copy-back-and-apply-log` subcommand, any InnoDB tables that were created on the server as compressed tables (by using the `ROW_FORMAT=COMPRESSED` option, the `KEY_BLOCK_SIZE=` option, or both) did not get restored, because they had not been further compressed by `mysqlbackup` when the backup was created, and the table files bear the `.ibd` instead of the `.ibz` extension, causing them to be ignored when `mysqlbackup`, with the `--uncompress` option, tried to restore the compressed backup. With this fix, InnoDB tables that are not compressed by `mysqlbackup` are now also saved with the `.ibz` extension, so they will be restored just like tables compressed by `mysqlbackup`.

**Note**

This fix only works for compressed backups created with MySQL Enterprise Backup 3.10.0 and after. For backups created by earlier versions, the workaround is to perform the restore in two steps: instead of using `copy-back-and-apply-log`, perform an `apply-log` first, followed by a `copy-back`.

(Bug #17992297)

- When a database was initialized with `--innodb-file-per-table=0` and had a fixed-size system tablespace, all non-InnoDB files backed up had zero size. (Bug #17793020)
- A backup taken with both the `--use-tts=with-full-locking` and `--skip-unused-pages` options could not be restored. This was because with `--use-tts=with-full-locking` used, the `apply-log` operation was always skipped (the command did nothing to the backup), while the expansion of the unused pages was part of the `apply-log` operation. This fix separates the expansion from the `apply-log` operation, so that the backup can be restored. (Bug #17764917)
- After a server restoration from an incremental backup with the `copy-back-and-apply-log` subcommand, `mysqlbackup` returned a success code even after a file renaming failed during the restoration process. The restored server thus failed to start with an assertion error. This fix makes sure a proper error is thrown when a file renaming fails during a restoration. (Bug #17539568)
- In a replication setup, when a backup was performed on a master, the modifications of the tables `mysql.backup_history` and `mysql.backup_progress` were propagated to the slaves, causing a wrong backup status for the slaves to be registered. This fix makes MySQL Backup Enterprise disable binary logging by setting `set sql_log_bin=0` during a backup on the master, so that the backup information will not get replicated to the slaves. (Bug #17449449)
- When `copy-back-and-apply-log` was used to restore a backup created using the `--skip-unused-pages` option (which is not a supported use case of `copy-back-and-apply-log`), MySQL Enterprise Backup crashed. This fix makes MySQL Enterprise Backup throw an error instead. (Bug #17281069, Bug #17955913)
- When the `--messages-logdir` option was used and the `--trace` option was set to “2” or “3” for an `apply-log` operation, MySQL Enterprise Backup created an extra log file. (Bug #17271318)
- MySQL Enterprise Backup continued to run even after reporting that `--messages-logdir` specified a non-existing directory. This fix makes `mysqlbackup` exit gracefully upon the error. (Bug #17268391)
- Because the default value for the parameter `innodb_data_file_path` had been changed to “ibdata1:12M:autoextend” since MySQL 5.6.7, during a backup restoration, if the value of the parameter was not specified in a configuration file or in the command, `mysqlbackup` would still use its default value of “ibdata1:10M:autoextend”, which would cause the restoration to fail. With this fix, `mysqlbackup` will use the value of `innodb_data_file_path` as found in the backup's `backup-my.cnf` file, but will also issue a warning that, depending on the configuration of the target server, the user might need to specify a value for the parameter. (Bug #16547531)

- With binlogging enabled, MySQL does not necessarily flush the redo log buffer on commit. The behaviour might cause MySQL Enterprise Backup to take an inconsistent backup, with some of the transactions (those still in the redo log buffer when the backup was being taken) possibly missing from the backup. With this fix, MySQL Enterprise Backup maintains the backup's consistency by performing a `FLUSH ENGINE LOGS` on the database before it records the binlog position and copies the redo log entries into the backup. (Bug #16401736)
- When the `--only-innodb-with-frm` option was used, MySQL Enterprise Backup failed to copy `.par` and `.frm` files during backups. (Bug #15915315)