

---

# MySQL Router 8.4 Release Notes

## Abstract

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

For additional MySQL Router documentation, see <https://dev.mysql.com/doc/mysql-router/en/>.

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: 2025-07-02 (revision: 30207)

## Table of Contents

Preface and Legal Notices .....	1
Changes in MySQL Router 8.4.5 (2025-04-15, LTS Release) .....	3
Changes in MySQL Router 8.4.4 (2025-01-21, LTS Release) .....	3
Changes in MySQL Router 8.4.3 (2024-10-15, LTS Release) .....	4
Changes in MySQL Router 8.4.2 (2024-07-23, LTS Release) .....	4
Changes in MySQL Router 8.4.1 (2024-07-01, LTS Release) .....	4
Changes in MySQL Router 8.4.0 (2024-04-30, LTS Release) .....	5
Changes in MySQL Router 8.3.0 (2024-01-16, Innovation Release) .....	7
Changes in MySQL Router 8.2.0 (2023-10-25, Innovation Release) .....	8
Changes in MySQL Router 8.1.0 (2023-07-18, Innovation Release) .....	10

## Preface and Legal Notices

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

### Legal Notices

Copyright © 2006, 2025, Oracle and/or its affiliates.

#### License Restrictions

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.

#### Warranty Disclaimer

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.

## **Restricted Rights Notice**

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), 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 embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

## **Hazardous Applications Notice**

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.

## **Trademark Notice**

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

Intel and Intel Inside 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, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

## **Third-Party Content, Products, and Services Disclaimer**

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.

## **Use of This Documentation**

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

<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 Router 8.4.5 (2025-04-15, LTS Release)

### Bugs Fixed

- Dollar-quoted strings were not supported. A syntax error was returned. (Bug #37538219)
- The `http_server` plugin would throw an uncaught exception and terminate when given an invalid URL. Now the exception is caught with a 400 HTTP (Bad Request) error returned to the client. (Bug #37499075)
- If a client without the SESSION TRACK capability, or without enabling `trackSessionState` (MySQL Connector/J), connected to the Read/Write splitting port of MySQL Router, updates could fail with an error similar to the following:

```
Buffer length is less than expected payload length.
```

(Bug #36374232)

- Empty and commented out statements were not allowed with Read-Write splitting enabled. (Bug #116952, Bug #37391353)
- Enabling the `ANSI_QUOTES sql_mode` was not allowed with Read-Write splitting enabled, and queries with it enabled would log an 'Unknown column' warning. (Bug #116950, Bug #37391347)

## Changes in MySQL Router 8.4.4 (2025-01-21, LTS Release)

### Bugs Fixed

- **Important Change:** MySQL Router no longer supports MySQL Server versions from a newer series. The metadata is checked and discarded for servers from a newer series, a check that executes during bootstrap and during periodic metadata refresh operations. A newer version in the same series is supported. For example, MySQL Router 8.4.4 supports MySQL Server 8.4.5 but not MySQL Server 9.0.0. (Bug #36041256)
- Because building with `CMAKE_BUILD_TYPE=Debug` disables optimizations, `std::vector::resize()` attempted to initialize every byte before each read from an encrypted socket. Memory usage was improved to account for this. (Bug #37385923)
- Accessing the `/routes/{name}/connections` REST API endpoint while also benchmarking queries through the router could cause the router to unexpectedly halt. (Bug #37251508)

## Changes in MySQL Router 8.4.3 (2024-10-15, LTS Release)

### Bugs Fixed

- With connection sharing enabled, changing the password of an active connection yielded a vague "Lost connection" error instead of indicating that reauthentication failed. (Bug #36981282)
- Updated `<cstring>` and `<iostream>` header usage by adding or removing their inclusion as needed, and annotated the functions utilized from each. (Bug #36946579, Bug #36951336)
- Replaced most uses of `std::enable_if<>` and `std::enable_if_t<>` with the `requires` keyword introduced in C++20, to limit the visibility of methods. This improves the error messages and stack traces. (Bug #36939919)
- On macOS, AddressSanitizer (ASan) discovered a potential issue with connection pooling that could cause Router to unexpectedly halt. (Bug #36935988)
- Improved the dependency injection mechanism (DIM) to increase performance in a multithreaded environment. DIM is no longer controlled by a single mutex. (Bug #36846616)
- In some cases, the LogFilter would execute when a log message wasn't logged. (Bug #36841009)
- Improved HTTP request handling to better scale for a large number of parallel connections. (Bug #36796808)
- A TLS shutdown was not executed when Read-Write splitting was used and a prepared statement needed to be prepared on the Write node but the current connection was on the Read node. (Bug #36715733)
- Connections in the connection pool were closed without waiting for the server to close the connection which affected the TLS session reuse cache. This affected connection sharing and Read-Write splitting as both let the connection pool close their connections. (Bug #36715372)
- Greatly improved HTTP/2 compatibility for the REST API. (Bug #36697876)
- With Read-Write splitting enabled and after the PRIMARY switched its role to SECONDARY, all write statements would fail with an error indicating that the MySQL server is running in `super_read_only` mode. Now it drops the client connection like it does without Read-Write splitting. (Bug #114594, Bug #36591958)

## Changes in MySQL Router 8.4.2 (2024-07-23, LTS Release)

This release contains no functional changes and is published to align its version number with that of the MySQL Server 8.4.2 release.

## Changes in MySQL Router 8.4.1 (2024-07-01, LTS Release)



### Important

This release is no longer available for download. It was removed due to a dependency on a version of the server which is also no longer available for download. See [MySQL 8.4.1 Release Notes](#). Please upgrade to MySQL Router 8.4.2 instead.

### Bugs Fixed

- System variable changes were not propagated when read-write splitting was enabled. (Bug #36674083)
- Current schema changes were not propagated to shared connections when both read-write splitting and connection sharing were enabled. (Bug #36663727)

- MySQL Router always updates the `last_check_in` in the Metadata Schema when it starts, even if the Cluster is `INVALIDATED`, leading to an errant transaction. As of this release, MySQL Router does not update `last_check_in` if the Cluster is `INVALIDATED`. (Bug #36563430)
- Increased the maximum `router_id` value from 999999 to 4294967295 ( $2^{32}-1$ ). This also affects how the internal metadata\_cache `user` name is defined. (Bug #36393211)

## Changes in MySQL Router 8.4.0 (2024-04-30, LTS Release)

- [Deprecation and Removal Notes](#)
- [Functionality Added or Changed](#)
- [Bugs Fixed](#)

### Deprecation and Removal Notes

- The configuration option `bootstrap_server_addresses`, deprecated in MySQL Router 8.0.14, is removed in this release.  
  
If this option is present in a configuration file, MySQL Router generates an error. (WL #15867)
- The configuration option `unreachable_destination_refresh_interval`, deprecated in MySQL Router 8.0.32, is removed in this release.  
  
If this option is present in a configuration file, MySQL Router generates an error. (WL #15869)
- As of this release, version 1.x of MySQL InnoDB Cluster Metadata is no longer supported. It is recommended to always use the latest version of MySQL Shell and MySQL Router.  
  
See [Upgrading MySQL Router](#) and [Upgrade Metadata Schema](#). (WL #15868)
- The `allow_primary_reads` parameter of the `destinations` URI is removed in this release. (WL #15872)
- The configuration option `mode`, deprecated in MySQL Router 8.0.4, is removed in this release. (WL #15877)

### Functionality Added or Changed

- As of this release, MySQL Router exposes its configuration in the Cluster metadata for all routers bootstrapped against it. This information is stored as JSON in the Cluster metadata schema and can be accessed by the MySQL Shell operation, `object.routerOptions` for Cluster, ClusterSet, and ReplicaSets. (WL #15649)
- As of this release, server connections which were not explicitly closed, but remain idle longer than the defined `connection-sharing-delay`, are no longer placed in the connection pool. Instead, the connections remain open and available for use by new client connections.  
  
It is no longer necessary to enable the connection pool to enable connection sharing. Therefore `max_idle_server_connections` can be zero if connection sharing is enabled. (WL #15742)

### Bugs Fixed

- If a hostname was not resolved, due to a DNS failure, MySQL Router did not check if that host became available again, later. (Bug #36246652)
- MySQL Router's bootstrap process checks the `mysql.user` table for the unsupported `mysql_native_password` authentication plugin. If the bootstrap user had no access to the table, the following error was returned:

```
Failed checking the Router account authentication plugin: Error executing MySQL query "select
```

```
'user'": SELECT command denied to user 'user'@'host'
for table 'user' (1142)
```

As of this release, this error is not returned. (Bug #36225456)

- If MySQL Router was bootstrapped without the `--account` parameter, a new metadata user was added using the following query:

```
CREATE USER IF NOT EXISTS username@hostname IDENTIFIED BY 'password'
```

If the server used an authentication plugin which does not accept single-factor or password-based authentication, the account creation failed. As of this release, the `caching_sha2_password` plugin is used in this situation:

```
CREATE USER IF NOT EXISTS username@hostname IDENTIFIED WITH `caching_sha2_password` BY 'password'
```

MySQL Router used `default_authentication_plugin` to determine the authentication plugin to use. This variable is now deprecated. MySQL Router uses `caching_sha2_password`. (Bug #36220663, Bug #36245132)

- The following error was displayed if MySQL Router was closed before the metadata cache started:

```
Error: routing:_: Metadata Cache not initialized
```

(Bug #36151125)

- It was not possible to connect to MySQL through MySQL Router, using the Node.js MySQL driver, if MySQL Router was running with both `client_ssl_mode` and `server_ssl_mode` disabled. Authentication failed without a useful error message.

As of this release, MySQL Router returns an error explaining that the Node.js MySQL driver does not support the requested authentication protocol. (Bug #36105279)

- If an incoming port is opened and closed by a TCP connection, such as a load balancer or a service-monitoring utility performing a health check, the connection counts towards the limit defined by `max_connect_errors`, even though the connection was not established. This could lead to MySQL Router closing the incoming port when the error limit is reached.

As of this release, `max_connect_errors` is not incremented if the Router's incoming port is opened and closed without establishing a connection. (Bug #36104070)

- If a client sent read-only statements to a destination which was not configured as `super_read_only`, using the Read-Write splitting port, MySQL Router redirected the statement to the primary. The statement did not return the correct result. (Bug #36042078)
- It was not possible to build MySQL Router on 32-bit Linux platforms. (Bug #36040942)
- Queries with trailing semi-colons (;) were treated as multi-statements, which is not permitted if MySQL Router is configured for Read-Write Splitting. (Bug #36036725)
- The following error did not provide enough information for troubleshooting:

```
timestamp routing ERROR [code] connecting to backend failed: Connection timed out (generic:110)
```

As of this release, it contains the following details:

- Route name and client source.
- `resolve()` errors.

- Hostname, IP addresses, and errors for each connection attempt.
- The amount of time spent on the connection attempt.

(Bug #35503245)

## Changes in MySQL Router 8.3.0 (2024-01-16, Innovation Release)

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

### Functionality Added or Changed

- The default value of `--server-ssl-mode` is changed from `AS_CLIENT` to `PREFERRED`. (WL #15966)
- MySQL Router now supports client TLS certificates.

The following configuration options are introduced:

- `server_ssl_key`
- `server_ssl_cert`
- `client_ssl_ca`
- `client_ssl_capath`
- `client_ssl_crl`
- `client_ssl_crlpath`
- `router_require_enforce`

(WL #14304)

### Bugs Fixed

- If a client sent read-only statements to a destination which was not configured as `super_read_only`, using the Read-Write splitting port, MySQL Router redirected the statement to the primary. The statement did not return the correct result. (Bug #36042078)
- Queries with trailing semi-colons (;) were treated as multi-statements, which is not permitted if MySQL Router is configured for Read-Write Splitting. (Bug #36036725)
- It was not possible to connect to the read-write splitting port with Connector/J. The following error was returned:

```
Unknown character set ''
```

(Bug #35968017)

- If MySQL Router was bootstrapped over an existing configuration, the existing user's credentials were retrieved from the keyring and reused, even if the user used the deprecated authentication plugin, `mysql_native_password`.

As of this release, MySQL Router checks the user's authentication method and, if it is set to `mysql_native_password`, changes it to the Cluster's default if there are no other `user@host` entries for the MySQL Router user. If there is more than one such entry, a warning is returned, advising the user to manually correct the authentication configuration. (Bug #35943415)



- If MySQL Router was bootstrapped against an InnoDB Cluster, without using an existing account or using the `--force-password-verification` parameter, the auto-created metadata user was created using the `mysql_native_password` plugin. This plugin was deprecated in MySQL Server 8.1 and is subject to removal in a future release.

As of this release, the metadata user is created using the default authorization plugin defined on the InnoDB Cluster. (Bug #35737357)

- In the Docker image for MySQL Router 8.2.0, the read-write splitting port, 6450, was not exposed by default. (Bug #112900, Bug #35964936)

## Changes in MySQL Router 8.2.0 (2023-10-25, Innovation Release)

- [Deprecation and Removal Notes](#)
- [Functionality Added or Changed](#)
- [Bugs Fixed](#)

### Deprecation and Removal Notes

- The `allow_primary_reads` parameter of the `destinations` URI is now deprecated and subject to removal in a future version of MySQL Router. Use the `PRIMARY_AND_SECONDARY` parameter instead. For example:

```
[routing:example]
destinations=metadata-cache://cache-name/default?role=PRIMARY_AND_SECONDARY
```

(WL #15871)

- Metadata Schema v1.x support is now deprecated and subject to removal in a future version of MySQL Router. Connecting to a Cluster which uses this version now triggers a warning stating that the metadata version is deprecated and the Cluster metadata should be upgraded. (WL #15876)

### Functionality Added or Changed

- MySQL Router now recognizes accounts with empty password sent from PHP over the classic MySQL protocol. (Bug #35463338)
- MySQL Router can be configured to route traffic to a partition of a Cluster which does not have quorum. This option is configured in the Cluster metadata, using the `unreachable_quorum_allowed_traffic` option of `cluster.setRoutingOption()` in MySQL Shell's AdminAPI.

See [Routing Options](#). (WL #15841)

- MySQL Router supports Read-Write splitting. This configuration enables you to direct all read traffic to read-only instances, and all write traffic to read/write instances.

The following `router` configuration options are added:

- `access_mode`
- `wait_for_my_writes`
- `wait_for_my_writes_timeout`

Read-Write splitting is enabled by default. A new section, `[router:read_write_split]`, is added automatically to the bootstrap configuration. To disable this configuration, you must bootstrap with `--disable-rw-split`.



See [Read/Write Splitting](#). (WL #12794)

## Bugs Fixed

- Queries larger than 16MB were not properly handled if Connection Sharing was enabled. (Bug #35769702)
- Unclosed comments, `/*` without a corresponding closing `/*`, caused MySQL Router to close unexpectedly if Connection Sharing was enabled. (Bug #35769610)
- Authentication failed for third-party clients using non-SSL connections, with caching-sha2-password and the following SSL options configured:
  - `client_ssl_mode` set to `PREFERRED` or `PASSTHROUGH`
  - `server_ssl_mode=AS_CLIENT`

An error similar to the following was returned:

```
Couldn't read RSA public key from server
```

(Bug #35737521)

- MySQL Router closed unexpectedly when attempting to connect to a ClusterSet but only had access to members without quorum. (Bug #35705590)
- If the directory used for bootstrapping contained a symlink, the keyring's master key could not be located. An error similar to the following was returned:

```
Error: Master key for keyring at '/bootstrapPath/data/keyring'  
could not be read
```

(Bug #35630329)

- Under certain circumstances, MySQL Router did not shutdown when closed with SIGTERM or SIGINT. It was possible for a connection to be in an intermediate state and block the shutdown until that remaining connection was closed by the user. (Bug #35574557)
- If an invalid client greeting was received, the following message was logged:

```
loop(): Input too short
```

This message was not helpful to the user. As of this release, an error message is returned to the client and no error message is logged if the packet is invalid. (Bug #35523018)

- MySQL Router maintained the router-server connection until the transaction was complete, even when the client-router connection had closed. This could result in a `max-connection` error.

As of this release, MySQL Router checks the client-router connection status while waiting for the transaction response. (Bug #35515899)

- Routing on named sockets did not resume after Cluster recovery. Error 2002 was logged. (Bug #35503286)
- MySQL Router error log did not contain MySQL Router version information. (Bug #35503191)
- Server-side TLS session were not reused after multiple consecutive authentication failures due to invalid authentication data. (Bug #35499025)
- It was not possible to connect to MySQL Router with PHP and an account with an empty password using a command similar to the following:

```
php -r 'mysqli_real_connect(mysqli_init(), "127.0.0.1", "empty", "");'
```

Authentication failed with the following error:

```
PHP Warning: mysqli_real_connect(): Premature end of data
(mysqlnd_wireprotocol.c:703) in Command line code on line 1
PHP Warning: mysqli_real_connect(): AUTH_RESPONSE packet 1 bytes
shorter than expected in Command line code on line 1
```

(Bug #35493871)

- Connection sharing was not possible for clients which do not support session trackers, such as PHP's `mysqlnd` or Python's `pymysql`. (Bug #35468897)
- Connection sharing was disabled if a change-user operation occurred over a plaintext connection. (Bug #35467047)
- If a client aborted a TLS handshake due to a certificate which cannot be verified, the following was incorrectly logged as an ERROR by MySQL Router:

```
ERROR ... classic::loop() processor failed:
error:0A000418:SSL routines::tlsv1 alert unknown ca
(tls_err:167773208)
```

As of this release, the connection is closed without a `processor failed` error and an INFO message is logged explaining why the TLS handshake failed. (Bug #35443773)

- MySQL Router logs did not contain the correct information for classic MySQL protocol connections. For TCP/IP, the address information was missing, while UNIX socket connections were missing the amount of data transferred. (Bug #35431001)
- MySQL Router did not reject ambiguous port configuration. For example, if both `bind_address` and `bind_port` were specified with different port numbers. (Bug #34972789)
- MySQL Router disconnected all connections to a Cluster if the user added a new instance to an existing Cluster with only one remaining member. This occurred because the new instance was present in the Group Replication metadata but not yet in the Cluster's metadata. MySQL Router assumed there was no quorum and disconnected. It was possible to reconnect after the new instance was represented in the Cluster's metadata.

As of this release, MySQL Router relies only on the Group Replication metadata for quorum reporting. (Bug #33989165)

- Under certain circumstances, on Microsoft Windows platforms, it was not possible to run MySQL Router as a service. An error message stating the MySQL Router configuration file could not be found was logged even though the configuration file was present and correct. (Bug #33301070)

## Changes in MySQL Router 8.1.0 (2023-07-18, Innovation Release)

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

### Functionality Added or Changed

- TLS sessions from client to router and router to server can now be cached and resumed when needed. This shortens the connection handshake, saving time and resources.

The following configuration options were added:

- Client TLS session cache:
  - `client_ssl_session_cache_mode`: Enables or disables the cache for client-router TLS sessions.
  - `client_ssl_session_cache_size`: Defines the maximum number of sessions cached.
  - `client_ssl_session_cache_timeout`: Defines the maximum amount of time, in seconds, a session remains in the cache. If the timeout is reached, and the session is not reused, the session is removed from the cache and the connection is closed.
- Server TLS Session Cache:
  - `server_ssl_session_cache_mode`: Enables or disables the cache for router-server sessions.
  - `server_ssl_session_cache_size`: Defines the maximum number of sessions cached.
  - `server_ssl_session_cache_timeout`: Defines the maximum amount of time, in seconds, a session remains in the cache. If the timeout is reached, and the session is not reused, the session is removed from the cache and the connection is closed.

The TLS session caches are enabled by default. (WL #15573)

- The `status_update_frequency` changes introduced for ClusterSets in MySQL Router 8.0.31 are extended to apply to Clusters and ReplicaSets, but only for Metadata Schema version 2.2.0 or higher. (WL #15599)
- If MySQL Router connects to a server and the connection is refused with a transient error, such as `max-connections reached`, MySQL Router now retries the connection after a number of seconds defined in the new configuration option `connect_retry_timeout`. (WL #15721)
- MySQL Router supports InnoDB Cluster Read Replicas.

MySQL Router reads the values defined in the metadata field, `v2_router_options.router_options.read_only_targets`, to retrieve routing information for read-only traffic.

`v2_router_options.router_options.read_only_targets` is populated by the AdminAPI method `cluster.setRoutingOption()` which sets the routing policy to one of the following values with the `read_only_targets` option:

- `all`: all Read Replicas and Secondary cluster members are used for read-only traffic.
- `read_replicas`: only Read Replicas are used for read-only traffic.
- `secondaries`: only Secondary cluster members are used for read-only traffic.



#### Note

If `read_only_targets` is not present, or set to a value other than `all`, `read_replicas`, or `secondaries`, MySQL Router defaults to `secondaries`.

A warning is logged if a value other than `all`, `read_replicas`, or `secondaries` is used.

(WL #15086)

- MySQL Router now supports tracing of statements as they are processed by MySQL Router from client to server and the response to the client. The trace is returned as JSON.

This enables debugging, testing, application connection comparisons, and so on.

See [MySQL Router Set Trace](#). (WL #15582)

## Bugs Fixed

- Deprecated ciphers were removed from the default cipher lists of `client_ssl_cipher` and `server_ssl_cipher`. Triple-DES ciphers and DSS ciphers are no longer permitted. (Bug #35489217)
- It was not possible to run two MySQL Routers as services on the same Microsoft Windows host. (Bug #35453506)
- MySQL Router failed to start if configured to use ECC certificates. The following error was displayed: `no RSA certificate`. (Bug #35317484)
- Connections were not shared if the client enabled `CLIENT_MULTI_STATEMENTS` with `COM_SET_OPTION`. (Bug #35263986)
- `SHOW WARNINGS` did not behave as expected if connection sharing was enabled. It did not return all the status-flags, such as `autocommit`. (Bug #35213196)
- Client authentication failed if the client authenticated with a password and a default authentication method other than `caching_sha2_password`. The error message included: `using password: NO`. (Bug #35195287)
- Connection sharing was disabled if a change-user operation occurred, resetting the user's stored password. As of this release, if a change-user operation is called, the existing connection is updated with the new credentials. (Bug #35189721)
- The client's character set was not restored when their connection was restored from the connection pool. (Bug #35184765)