Percona ProxySQL version 1 and Admin tool Documentation

None

Percona Technical Documentation Team

None

Table of contents

1. ProxySQL, proxysql-admin, and percona-scheduler-admin documentation

This documentation is for the latest release: Percona ProxySQL admin tools 2.7.1-1 (Release notes).

ProxySQL is a tool that performs like a proxy between *Percona XtraDB Cluster* and your client application. *ProxySQL* manages a connection pool, which caches your connections and keeps the connections open for future requests. *ProxySQL* is designed to run continuously without being restarted.

Without a connection pool, each SQL request opens a connections to the remote node. When the SQL request is complete, the connection is closed. A new one is opened on the next SQL request.

ProxySQL maintains the connection pool. The pool allows a certain number of connections to remain open. A connection is reused or closed if not reused within a specific time. You connect to the proxy and the tool forwards your requests to the cluster.

ProxySQL runs as a daemon watched by a monitoring process which can restart *ProxySQL* in case of an unexpected exit to minimize downtime. The daemon accepts incoming traffic from *MySQL* clients and forwards the traffic to backend *MySQL* servers.

The configuration options include runtime parameters, server grouping, and traffic-related parameters. Many of the settings can be done at runtime using queries that are similar to SQL statements.

The ProxySQL documentation provides information on installing and running ProxySQL and the ProxySQL admin tools.

ProxySQL is available from the Percona software repositories with the following:

- ProxySQL 1.x.x downloads include:
    - ProxySQL Admin 1.x.x does not natively support *Percona XtraDB Cluster* and requires custom `Bash` scripts to track the status of a *Percona XtraDB Cluster*.

2024-10-31

2. Get help from Percona

Our documentation guides are packed with information, but they can’t cover everything you need to know about Percona ProxySQL admin tools. They also won’t cover every scenario you might come across. Don’t be afraid to try things out and ask questions when you get stuck.

2.1 Percona’s Community Forum

Be a part of a space where you can tap into a wealth of knowledge from other database enthusiasts and experts who work with Percona’s software every day. While our service is entirely free, keep in mind that response times can vary depending on the complexity of the question. You are engaging with people who genuinely love solving database challenges.

We recommend visiting our Community Forum. It’s an excellent place for discussions, technical insights, and support around Percona database software. If you’re new and feeling a bit unsure, our FAQ and Guide for New Users ease you in.

If you have thoughts, feedback, or ideas, the community team would like to hear from you at Any ideas on how to make the forum better?. We’re always excited to connect and improve everyone’s experience.

2.2 Percona experts

Percona experts bring years of experience in tackling tough database performance issues and design challenges. We understand your challenges when managing complex database environments. That’s why we offer various services to help you simplify your operations and achieve your goals.

Service	Description
24/7 Expert Support	Our dedicated team of database experts is available 24/7 to assist you with any database issues. We provide flexible support plans tailored to your specific needs.
Hands-On Database Management	Our managed services team can take over the day-to-day management of your database infrastructure, freeing up your time to focus on other priorities.
Expert Consulting	Our experienced consultants provide guidance on database topics like architecture design, migration planning, performance optimization, and security best practices.
Comprehensive Training	Our training programs help your team develop skills to manage databases effectively, offering virtual and in-person courses.

We’re here to help you every step of the way. Whether you need a quick fix or a long-term partnership, we’re ready to provide your expertise and support.

2025-01-21

3. ProxySQL 1.x.x and the proxysql-admin

3.1 Use ProxySQL 1.x.x with ProxySQL Admin

ProxySQL version 1.4.x does not natively support *Percona XtraDB Cluster* and proxysql-admin and requires custom bash scripts to keep track of *Percona XtraDB Cluster* status: proxysql_galera_checker and proxysql_node_monitor.

3.1.1 Automatic configuration

The `proxysql` package from *Percona* includes the *proxysql-admin* tool for configuring *Percona XtraDB Cluster* nodes with *ProxySQL*.

> **Note**
>
> The proxysql-admin script is specially developed by *Percona* to automate the *ProxySQL* configuration. Bug reports and feature proposals are welcome in the proxysql-admin issue tracking system.
>
> The proxysql-admin tool can only be used for the *initial ProxySQL* configuration.

To view usage information, run `proxysql-admin` without any options:

```
Usage: [ options ]
Options:
--config-file=<config-file>        Read login credentials from a configuration file
                        (command line options override any configuration file login credentials)
--proxysql-datadir=<datadir>       Specify the proxysql data directory location
--proxysql-username=user_name      ProxySQL service username
--proxysql-password[=password]     ProxySQL service password
--proxysql-port=port_num           ProxySQL service port number
--proxysql-hostname=host_name      ProxySQL service hostname
--cluster-username=user_name       Percona XtraDB Cluster node username
--cluster-password[=password]      Percona XtraDB Cluster node password
--cluster-port=port_num            Percona XtraDB Cluster node port number
--cluster-hostname=host_name       Percona XtraDB Cluster node hostname
--cluster-app-username=user_name   Percona XtraDB Cluster node application username
--cluster-app-password[=password]  Percona XtraDB Cluster node application password
--without-cluster-app-user         Configure Percona XtraDB Cluster without application user
--monitor-username=user_name       Username for monitoring Percona XtraDB Cluster nodes through ProxySQL
--monitor-password[=password]      Password for monitoring Percona XtraDB Cluster nodes through ProxySQL
--use-existing-monitor-password    Do not prompt for a new monitor password if one is provided.
--node-check-interval=3000         Interval for monitoring node checker script (in milliseconds)
                                (default: 3000)
--mode=[loadbal|singlewrite]       ProxySQL read/write configuration mode
                                    currently supporting: 'loadbal' and 'singlewrite'
                                    (default: 'singlewrite')
--write-node=host_name:port        Writer node to accept write statments.
                                    This option is supported only when using --mode=singlewrite
                                    Can accept comma-delimited list with the first listed being
                                    the highest priority.
--include-slaves=host_name:port    Add specified slave node(s) to ProxySQL, these nodes will go
                                into the reader hostgroup and will only be put into
                                the writer hostgroup if all cluster nodes are down (this
                                depends on the value of --use-slave-as-writer).
                                Slaves must be read only.  Can accept a comma-delimited list.
                                If used, make sure 'read_only=1' is in the slave's my.cnf
--use-slave-as-writer=<yes/no>     If this value is 'yes', then a slave may be used as a writer
                                if the entire cluster is down. If 'no', then a slave
                                will not be used as a writer. This option is required
                                if '--include-slaves' is used.
--writer-is-reader=<value>         Defines if the writer node also accepts writes.
                                Possible values are 'always', 'never', and 'ondemand'.
                                'ondemand' means that the writer node only accepts reads
                                if there are no other readers.
                                (default: 'ondemand')
--max-connections=<NUMBER>         Value for max_connections in the mysql_servers table.
```

```
                                This value is the maximum number of connections that
                                ProxySQL will open to the backend servers.
                                (default: 1000)
--debug                            Enables additional debug logging.
--help                             Displays this help text.

These options are the possible operations for proxysql-admin.
You must provide one of the options.
--adduser                          Adds the Percona XtraDB Cluster application user to the ProxySQL database
--disable, -d                      Remove any Percona XtraDB Cluster configurations from ProxySQL
--enable, -e                       Auto-configure Percona XtraDB Cluster nodes into ProxySQL
--quick-demo                       Setup a quick demo with no authentication
--syncusers                        Sync user accounts currently configured in MySQL to ProxySQL
                                May be used with --enable.
                                (deletes ProxySQL users not in MySQL)
--sync-multi-cluster-users         Sync user accounts currently configured in MySQL to ProxySQL
                                May be used with --enable.
                                (doesn't delete ProxySQL users not in MySQL)
--version, -v                      Print version info
```

Note

The *Percona XtraDB Cluster* nodes and *ProxySQL* must be available before using the proxysql-admin tool. For security purposes, change the default user settings in the *ProxySQL* configuration file.

3.1.2 Prepare a configuration file

We recommend providing the connection and authentication information in the *ProxySQL* configuration file (/etc/proxysql-admin.cnf), instead of specifying this information on the command line.

By default, the configuration file contains the following:

```
# proxysql admin interface credentials.
export PROXYSQL_DATADIR='/var/lib/proxysql'
export PROXYSQL_USERNAME='admin'
export PROXYSQL_PASSWORD='admin'
export PROXYSQL_HOSTNAME='localhost'
export PROXYSQL_PORT='6032'

# PXC admin credentials for connecting to pxc-cluster-node.
export CLUSTER_USERNAME='admin'
export CLUSTER_PASSWORD='admin'
export CLUSTER_HOSTNAME='localhost'
export CLUSTER_PORT='3306'

# proxysql monitoring user. proxysql admin script will create this user in pxc to monitor pxc-nodes.
export MONITOR_USERNAME='monitor'
export MONITOR_PASSWORD='monit0r'

# Application user to connect to pxc-node through proxysql
export CLUSTER_APP_USERNAME='proxysql_user'
export CLUSTER_APP_PASSWORD='passw0rd'

# *ProxySQL* read/write hostgroup
export WRITE_HOSTGROUP_ID='10'
export READ_HOSTGROUP_ID='11'

# *ProxySQL* read/write configuration mode.
export MODE="singlewrite"

# Writer-is-reader configuration
export WRITER_IS_READER="ondemand"

# max_connections default (used only when INSERTing a new mysql_servers entry)
export MAX_CONNECTIONS="1000"
```

Note

We recommend that you change the default *ProxySQL* credentials before running *ProxySQL* in production. Make sure that you provide *ProxySQL* location and credentials in the configuration file. See Do not use the default credentials. Provide superuser credentials for one of the *Percona XtraDB Cluster* nodes. The `proxysql-admin` script will detect other nodes in the cluster automatically.

3.1.3 Enabling ProxySQL

Use the `--enable` option to automatically configure a *Percona XtraDB Cluster* node into *ProxySQL*. The `proxysql-admin` tool will do the following:

- Add a *Percona XtraDB Cluster* node into the *ProxySQL* database
- Add the `proxysql_galera_checker` monitoring script into the ProxySQL scheduler table if it is not available. This script checks for desynced nodes and temporarily deactivates them. It also calls the `proxysql_node_monitor` script, which checks cluster node membership and re-configures *ProxySQL* if the membership changes.
- Create two new *Percona XtraDB Cluster* users with the `USAGE` privilege on the node and add them to *ProxySQL* configuration if they are not already configured. *ProxySQL* uses one user for monitoring cluster nodes, and the other one is used for communicating with the cluster. Make sure to use superuser credentials from cluster to set up the default users.

Warning

Running more than one copy of `proxysql_galera_check` in the same runtime environment simultaneously is not supported and may lead to undefined behavior. To avoid this problem, Galera process identification prevents a duplicate script execution in most cases. However, in some rare cases, it may be possible to circumvent this check if you run more than one copy of `proxysql_galera_check`.

The following example shows how to add a *Percona XtraDB Cluster* node using the *ProxySQL* configuration file with all necessary connection and authentication information:

```
$ proxysql-admin --config-file=/etc/proxysql-admin.cnf --enable
```

Expected output

```
This script will assist with configuring ProxySQL for use with
Percona XtraDB Cluster (currently, only PXC in combination with ProxySQL is supported)

ProxySQL read/write configuration mode is singlewrite

Configuring the ProxySQL monitoring user.  ProxySQL monitor user name as per
command line/config-file is monitor

User 'monitor'@'127.%' has been added with USAGE privileges

Configuring the Percona XtraDB Cluster application user to connect through ProxySQL
Percona XtraDB Cluster application user name as per command line/config-file is proxysql_user

Percona XtraDB Cluster application user 'proxysql_user'@'127.%' has been added with ALL privileges. This user is created for testing purposes.
Adding the Percona XtraDB Cluster server nodes to ProxySQL

Write node info

+-----------+--------------+-------+--------+
| hostname  | hostgroup_id | port  | weight |
+-----------+--------------+-------+--------+
| 127.0.0.1 | 10           | 26100 | 1000   |
+-----------+--------------+-------+--------+

ProxySQL configuration completed!

ProxySQL has been successfully configured to use with Percona XtraDB Cluster

You can use the following login credentials to connect your application through ProxySQL

$ mysql --user=proxysql_user -p --host=localhost --port=6033 --protocol=tcp
```

```
mysql> select hostgroup_id,hostname,port,status,comment from mysql_servers;
```

Expected output

```
+--------------+-----------+-------+--------+---------+
| hostgroup_id | hostname  | port  | status | comment |
+--------------+-----------+-------+--------+---------+
| 11           | 127.0.0.1 | 25400 | ONLINE | READ    |
| 10           | 127.0.0.1 | 25000 | ONLINE | WRITE   |
| 11           | 127.0.0.1 | 25100 | ONLINE | READ    |
| 11           | 127.0.0.1 | 25200 | ONLINE | READ    |
| 11           | 127.0.0.1 | 25300 | ONLINE | READ    |
+--------------+-----------+-------+--------+---------+
5 rows in set (0.00 sec)
```

3.1.4 Disabling ProxySQL

Use the `--disable` option to remove a *Percona XtraDB Cluster* node's configuration from *ProxySQL*. The `proxysql-admin` tool will do the following:

- Remove *Percona XtraDB Cluster* node from the *ProxySQL* database
- Stop the *ProxySQL* monitoring daemon for this node
- Remove the application user for this cluster
- Remove any query rules set up for this cluster

The following example shows how to disable *ProxySQL* and remove the *Percona XtraDB Cluster* node:

```
$ proxysql-admin --config-file=/etc/proxysql-admin.cnf --disable
*ProxySQL* configuration removed!
```

3.1.5 Additional options

The following extra options can be used:

- `--adduser`

  Add *Percona XtraDB Cluster* application user to *ProxySQL* database.

  ```
  $ proxysql-admin --config-file=/etc/proxysql-admin.cnf --adduser
  ```

  Expected output

  ```
  Adding Percona XtraDB Cluster application user to *ProxySQL* database
  Enter Percona XtraDB Cluster application user name: cluster_user
  Enter Percona XtraDB Cluster application user password: cluster_passw0Rd
  Added Percona XtraDB Cluster application user to *ProxySQL* database!
  ```

- `--syncusers`

Sync user accounts currently configured in *Percona XtraDB Cluster* to *ProxySQL* database except for users with no password and the `admin` user.

Warning

This option also deletes users that are not in *Percona XtraDB Cluster* from *ProxySQL* database.

- `--sync-multi-cluster-users`

This option works in the same way as `--syncusers`, but it does not delete *ProxySQL* users that are not present in the Percona XtraDB Cluster. It is to be used when syncing proxysql instances that manage multiple clusters.

- `--node-check-interval`

This option configures the interval for monitoring via the `proxysql_galera_checker` script (in milliseconds).

```
$ proxysql-admin --config-file=/etc/proxysql-admin.cnf \
--node-check-interval=5000 --enable
```

- `--mode`

Set the read/write mode for *Percona XtraDB Cluster* nodes in *ProxySQL* database, based on the hostgroup.

Supported modes are `loadbal` and `singlewrite`.

- `singlewrite` is the default mode, it will accept writes only on one single node (based on the info you provide in `--write-node`). The remaining nodes will accept only read statements.

  Servers can be separated by commas, for example:

  ```
  10.0.0.51:3306,10.0.0.52:3306
  ```

  In the previous example, `10.0.0.51:3306` will be in the writer hostgroup if it is ONLINE. If it is OFFLINE, then `10.0.0.52:3306` will go into the writer hostgroup. And if that node also goes down, then one of the remaining nodes will be randomly chosen for the writer hostgroup. The configuration file is deleted when `--disable` is used.

- `singlewrite` mode setup:

  ```
  $ sudo grep "MODE" /etc/proxysql-admin.cnf
  export MODE="singlewrite"
  $ sudo proxysql-admin --config-file=/etc/proxysql-admin.cnf --write-node=127.0.0.1:25000 --enable
  *ProxySQL* read/write configuration mode is singlewrite
  [..]
  *ProxySQL* configuration completed!
  ```

  To check the configuration you can run:

  ```
  mysql> SELECT hostgroup_id,hostname,port,status,comment FROM mysql_servers;
  +--------------+-----------+-------+--------+---------+
  | hostgroup_id | hostname  | port  | status | comment |
  +--------------+-----------+-------+--------+---------+
  | 11           | 127.0.0.1 | 25400 | ONLINE | READ    |
  | 10           | 127.0.0.1 | 25000 | ONLINE | WRITE   |
  | 11           | 127.0.0.1 | 25100 | ONLINE | READ    |
  | 11           | 127.0.0.1 | 25200 | ONLINE | READ    |
  | 11           | 127.0.0.1 | 25300 | ONLINE | READ    |
  +--------------+-----------+-------+--------+---------+
  5 rows in set (0.00 sec)
  ```

- The `loadbal` mode uses a set of evenly weighted read/write nodes.

  `loadbal` mode setup:

  ```
  $ sudo proxysql-admin --config-file=/etc/proxysql-admin.cnf --mode=loadbal --enable

  This script will assist with configuring *ProxySQL* (currently only Percona XtraDB cluster in combination with *ProxySQL* is supported)

  *ProxySQL* read/write configuration mode is loadbal
  [..]
  *ProxySQL* has been successfully configured to use with Percona XtraDB Cluster

  You can use the following login credentials to connect your application through *ProxySQL*

  mysql --user=proxysql_user --password=***  --host=127.0.0.1 --port=6033 --protocol=tcp
  ```

  ```
  mysql> SELECT hostgroup_id,hostname,port,status,comment FROM mysql_servers;
  +--------------+-----------+-------+--------+-----------+
  | hostgroup_id | hostname  | port  | status | comment   |
  +--------------+-----------+-------+--------+-----------+
  | 10           | 127.0.0.1 | 25400 | ONLINE | READWRITE |
  | 10           | 127.0.0.1 | 25000 | ONLINE | READWRITE |
  | 10           | 127.0.0.1 | 25100 | ONLINE | READWRITE |
  | 10           | 127.0.0.1 | 25200 | ONLINE | READWRITE |
  | 10           | 127.0.0.1 | 25300 | ONLINE | READWRITE |
  +--------------+-----------+-------+--------+-----------+
  5 rows in set (0.01 sec)
  ```

- `--quick-demo`

  This option is used to setup dummy *ProxySQL* configuration.

  ```
  $ sudo  proxysql-admin  --enable --quick-demo
  ```

Expected output

```
You have selected the dry test run mode. WARNING: This will create a test user (with all privileges) in the Percona XtraDB Cluster & *ProxySQL* installations.

You may want to delete this user after you complete your testing!

Would you like to proceed with '--quick-demo' [y/n] ? y

Setting up proxysql test configuration!

Do you want to use the default *ProxySQL* credentials (admin:admin:6032:127.0.0.1) [y/n] ? y
Do you want to use the default Percona XtraDB Cluster credentials (root::3306:127.0.0.1) [y/n] ? n

Enter the Percona XtraDB Cluster username (super user): root
Enter the Percona XtraDB Cluster user password:
Enter the Percona XtraDB Cluster port: 25100
Enter the Percona XtraDB Cluster hostname: localhost


*ProxySQL* read/write configuration mode is singlewrite

Configuring *ProxySQL* monitoring user..

User 'monitor'@'127.%' has been added with USAGE privilege

Configuring the Percona XtraDB Cluster application user to connect through *ProxySQL*

Percona XtraDB Cluster application user 'pxc_test_user'@'127.%' has been added with ALL privileges, this user is created for testing purposes

Adding the Percona XtraDB Cluster server nodes to *ProxySQL*

*ProxySQL* configuration completed!

*ProxySQL* has been successfully configured to use with Percona XtraDB Cluster

You can use the following login credentials to connect your application through *ProxySQL*

mysql --user=pxc_test_user  --host=127.0.0.1 --port=6033 --protocol=tcp
```

- `--include-slaves=host_name:port`

This option helps to include specified slave node(s) to *ProxySQL* database. These nodes will go into the reader hostgroup and will only be put into the writer hostgroup if all cluster nodes are down. Slaves must be read only. Can accept comma-delimited list. If this is used, make sure `read_only=1` is included into the slave’s `my.cnf` configuration file.

Note

With `loadbal` mode slave hosts only accept read/write requests when all cluster nodes are down.

3.1.6 proxysql_status script

There is a simple script to dump *ProxySQL* configuration and statistics:

```
Usage:

proxysql-status admin admin 127.0.0.1 6032
```

2022-12-07

3.2

3.2.1 Release Notes for Version 1

ProxySQL 1.4.16 and proxysql-admin (2020-02-11)

- **Installation**: https://www.percona.com/doc/percona-xtradb-cluster/LATEST/howtos/proxysql-v1.html#installing-proxysql-v1

ProxySQL release, released by ProxySQL, is now available for download in the Percona repository along with an updated version of **Percona**'s **proxysql-admin** tool.

ProxySQL is a high-performance proxy, currently for MySQL, and database servers in the MySQL ecosystem (like **Percona Server for MySQL** and MariaDB). It acts as an intermediary for client requests seeking resources from the database. René Cannaò created *ProxySQL* for DBAs as a means of solving complex replication topology issues.

The *ProxySQL* 1.4.16 source and binary packages are available from the Percona download page for ProxySQL include **ProxySQL Admin** – a tool developed by **Percona** to configure **Percona XtraDB Cluster** nodes into *ProxySQL*. Docker images are available as well. You can download the original ProxySQL from GitHub. *ProxySQL* offers the ProxySQL documentation.

BUGS FIXED

- PSQLADM-219: The ProxySQL scheduler was handling the `pxc_maint_mode` variable incorrectly. As a result, open connections were closed immediately. This bug has been fixed and now the ProxySQL scheduler only sets the node status to `OFFLINE_SOFT`. This prevents opening new connections and lets the already established connections finish their work. It is up to the user to decide when it is safe to start the node maintenance.

ProxySQL is available under Open Source license GPLv3.

ProxySQL 1.4.12 and proxysql-admin (2018-11-13)

- **Installation**: https://www.percona.com/doc/percona-xtradb-cluster/LATEST/howtos/proxysql-v1.html#installing-proxysql-v1

ProxySQL release, released by ProxySQL, is now available for download in the Percona repository along with an updated version of **Percona**'s **proxysql-admin** tool.

ProxySQL is a high-performance proxy, currently for MySQL, and database servers in the MySQL ecosystem (like **Percona Server for MySQL** and MariaDB). It acts as an intermediary for client requests seeking resources from the database. René Cannaò created *ProxySQL* for DBAs as a means of solving complex replication topology issues.

The *ProxySQL* 1.4.12 source and binary packages are available from the Percona download page for ProxySQL include **ProxySQL Admin** – a tool developed by **Percona** to configure **Percona XtraDB Cluster** nodes into *ProxySQL*. Docker images are available as well. You can download the original ProxySQL from GitHub. *ProxySQL* offers the ProxySQL documentation.

IMPROVEMENTS

- PSQLADM-68: Scripts are now compatible with *Percona XtraDB Cluster (PXC)* hosts using IPv6
- PSQLADM-107: In include-slaves, a slave would always be moved into the write hostgroup even if the whole cluster went down. A new option –use-slave-as-writer specifies whether or not the slave is added to the write hostgroup.

BUGS FIXED

- PSQLADM-110: In some cases, pattern cluster hostname did not work with proxysql-admin.
- PSQLADM-104: proxysql-admin testsuite bug fixes.
- PSQLADM-113: `proxysql_galera_checker` assumed that parameters were given in the long format.
- PSQLADM-114: In some cases, *ProxySQL* could not be started
- PSQLADM-115: `proxysql_node_monitor` could fail with more than one command in the ProxySQL scheduler.
- PSQLADM-116: In some cases, the ProxySQL scheduler was reloading servers on every run
- PSQLADM-117: The `–syncusers` option did not work when enabling cluster
- PSQLADM-125: The check-is-galera-checker-running function was not preventing multiple instances of the script from running.

Other bugs fixed: PSQLADM-112, PSQLADM-120

ProxySQL is available under Open Source license GPLv3.

ProxySQL 1.4.8 and proxysql-admin (2018-05-22)

- **Installation**

https://www.percona.com/doc/percona-xtradb-cluster/LATEST/howtos/proxysql-v1.html#installing-proxysql-v1

ProxySQL release, released by ProxySQL, is now available for download in the Percona repository along with an updated version of **Percona**’s **proxysql-admin** tool.

ProxySQL is a high-performance proxy, currently for MySQL, and database servers in the MySQL ecosystem (like **Percona Server for MySQL** and MariaDB). It acts as an intermediary for client requests seeking resources from the database. René Cannaò created *ProxySQL* for DBAs as a means of solving complex replication topology issues.

The *ProxySQL* 1.4.8 source and binary packages are available from the Percona download page for ProxySQL include ProxySQL Admin – a tool developed by **Percona** to configure **Percona XtraDB Cluster** nodes into *ProxySQL*. Docker images are available as well. You can download the original ProxySQL from GitHub. *ProxySQL* offers the ProxySQL documentation.

USABILITY IMPROVEMENT

- PSQLADM-84: Now proxysql_status tool dumps host_priority and /etc/proxysql-admin.cnf. Also output format was changed.

OTHER IMPROVEMENTS AND BUG FIXES

- PSQLADM-66: The `–syncusers` option now makes ProxySQL Admin to update the user’s password in *ProxySQL* database if there is any password difference between *ProxySQL* user and MySQL user.
- PSQLADM-45: it was unclear from the help screen, that `–config-file` option requires an argument.
- PSQLADM-48: `${PROXYSQL_DATADIR}/${CLUSTER_NAME}`_mode file was not created at ProxySQL Admin upgrade (1.4.5 or before to 1.4.6 onwards).
- PSQLADM-52: The *proxysql_galera_checker* script was not checking empty query rules.
- PSQLADM-54: `proxysql_node_monitor` did not change OFFLINE_HARD status properly for the coming back online nodes.

ProxySQL is available under Open Source license GPLv3.

ProxySQL 1.4.7 and proxysql-admin (2018-04-16)

- **Installation**: https://www.percona.com/doc/percona-xtradb-cluster/LATEST/howtos/proxysql-v1.html#installing-proxysql-v1

ProxySQL release, released by ProxySQL, is now available for download in the Percona repository along with an updated version of **Percona**'s **proxysql-admin** tool.

ProxySQL is a high-performance proxy, currently for MySQL, and database servers in the MySQL ecosystem (like **Percona Server for MySQL** and MariaDB). It acts as an intermediary for client requests seeking resources from the database. René Cannaò created *ProxySQL* for DBAs as a means of solving complex replication topology issues.

The *ProxySQL* 1.4.7 source and binary packages available from the Percona download page for ProxySQL include ProxySQL Admin – a tool developed by **Percona** to configure **Percona XtraDB Cluster** nodes into *ProxySQL*. Docker images are available as well. You can download the original ProxySQL from GitHub. *ProxySQL* offers the ProxySQL documentation.

USABILITY IMPROVEMENTS

- Added `proxysql_status` tool to dump *ProxySQL* configuration and statistics.

BUG FIXES

- PSQLADM-2: `proxysql_galera_checker` script didn't check if another instance of itself is already running. While running more then one copy of `proxysql_galera_checker` in the same runtime environment at the same time is still not supported, the introduced fix is able to prevent duplicate script execution in most cases.
- PSQLADM-40: ProxySQL scheduler generated a lot of `proxysql_galera_checker` and `proxysql_node_monitor` processes in case of wrong ProxySQL credentials in `/etc/proxysql-admin.cnf` file.
- PSQLADM-41: Timeout error handling was improved with clear messages.
- PSQLADM-42: An inconsistency of the date format in *ProxySQL* and scripts was fixed.
- PSQLADM-43: `proxysql_galera_checker` didn't take into account the possibility of special characters presence in mysql-monitor_password.
- PSQLADM-44: `proxysql_galera_checker` generated unclear errors in the `/etc/proxysql.log` file if wrong credentials where passed.
- PSQLADM-46: `proxysql_node_monitor` script incorrectly split the hostname and the port number in URLs containing hyphen character.

ProxySQL is available under Open Source license GPLv3.

ProxySQL 1.4.6 and proxysql-admin (2018-03-12)

- **Installation**: https://www.percona.com/doc/percona-xtradb-cluster/LATEST/howtos/proxysql-v1.html#installing-proxysql-v1

ProxySQL release, released by ProxySQL, is now available for download in the Percona repository along with an updated version of **Percona**'s **proxysql-admin** tool.

ProxySQL is a high-performance proxy, currently for MySQL, and database servers in the MySQL ecosystem (like **Percona Server for MySQL** and MariaDB). It acts as an intermediary for client requests seeking resources from the database. René Cannaò created *ProxySQL* for DBAs as a means of solving complex replication topology issues.

The *ProxySQL* 1.4.6 source and binary packages available from the Percona download page for ProxySQL include ProxySQL Admin – a tool developed by **Percona** to configure **Percona XtraDB Cluster** nodes into *ProxySQL*.

Docker images are available as well. You can download the original ProxySQL from GitHub. *ProxySQL* offers the ProxySQL documentation.

USABILITY IMPROVEMENTS

- PSQLADM-32: Now, proxysql-admin script can configure multiple clusters in *ProxySQL*, when there are unique cluster names specified by the wsrep*cluster_name option, and the /etc/proxysql-admin.cnf configuration contains different _ProxySQL* READ/WRITE hostgroup and different application user for each cluster. Currently multiple clusters support is not compatible with host priority feature, which works only with a single cluster.
- PSQLADM-81: The new version substantially increases the number of test cases in the ProxySQL Admin test-suite.

BUG FIXES

- PSQLADM-35: proxysql_galera_checker monitoring script was unable to discover new writer nodes.
- PSQLADM-36: upgrade to *ProxySQL* 1.4.6 from the previous version was broken.
- PSQLADM-79: Fixed by properly quoting the MONITOR_USERNAME environment variable in the admin script query.

ProxySQL is available under Open Source license GPLv3.

ProxySQL 1.4.5 and proxysql-admin (2018-02-15)

- **Installation**:https://www.percona.com/doc/percona-xtradb-cluster/LATEST/howtos/proxysql-v1.html#installing-proxysql-v1

ProxySQL 1.4.5, released by ProxySQL, is now available for download in the Percona repository along with an updated version of **Percona**’s **proxysql-admin** tool.

ProxySQL is a high-performance proxy, currently for MySQL, and database servers in the *MySQL* ecosystem (like **Percona Server for MySQL** and *MariaDB*). It acts as an intermediary for client requests seeking resources from the database. René Cannaò created *ProxySQL* for DBAs as a means of solving complex replication topology issues.

The *ProxySQL* 1.4.5 source and binary packages are available from the Percona download page for ProxySQL include ProxySQL Admin – a tool developed by **Percona** to configure **Percona XtraDB Cluster** nodes into *ProxySQL*. Docker images are available as well. You can download the original ProxySQL from GitHub. *ProxySQL* offers the ProxySQL documentation.

- PSQLADM-6: If the cluster node goes offline, the proxysql*node_monitor script now sets the node status as OFFLINE_HARD, and does not remove it from the _ProxySQL* database. Also, logging is consistent regardless of the cluster node online status.
- PSQLADM-30: Validation was added for the host priority file.
- PSQLADM-33: Added –proxysql-datadir option to run the proxysql-admin script with a custom *ProxySQL* data directory.
- Also, BATS test suite was added for the proxysql-admin testing.

BUG FIXES

- PSQLADM-5: *Percona XtraDB Cluster (PXC)* mode specified with proxysql-admin with use of `–mode` parameter was not persistent.
- PSQLADM-8: *ProxySQL* High CPU load took place when mysqld was hanging.

ProxySQL is available under Open Source license GPLv3.

ProxySQL 1.4.4 and proxysql-admin (2018-01-18)

- **Installation**: https://www.percona.com/doc/percona-xtradb-cluster/LATEST/howtos/proxysql-v1.html#installing-proxysql-v1

ProxySQL 1.4.4, released by ProxySQL, is now available for download in the Percona repository along with an updated version of **Percona**'s **proxysql-admin** tool.

ProxySQL is a high-performance proxy, currently for MySQL, and database servers in the MySQL ecosystem (like **Percona Server for MySQL** and MariaDB). It acts as an intermediary for client requests seeking resources from the database. René Cannaò created *ProxySQL* for DBAs as a means of solving complex replication topology issues.

The *ProxySQL* 1.4.4 source and binary packages are available from the Percona download page for ProxySQL include ProxySQL Admin – a tool developed by **Percona** to configure **Percona XtraDB Cluster** nodes into *ProxySQL*. Docker images are available as well. You can download the original ProxySQL from GitHub. *ProxySQL* offers the ProxySQL documentation.

THIS RELEASE FIXES THE FOLLOWING BUGS IN PROXYSQL ADMIN

- PXC-892: proxysql-admin was unable to recognize IP address of localhost.
- PXC-893: proxysql-admin couldn't interpret passwords with special characters correctly, such as '$'
- PSQLADM-3: proxysql_node_monitor script had writer/reader hostgroup conflict issue.
- PQA-155: Runtime table was not updated in case of any changes in Percona XtraDB Cluster membership.
- BLD-853: *ProxySQL* logrotate script did not work properly, producing empty /etc/proxysql.log after logrotate.

ProxySQL is available under Open Source license GPLv3.

2024-03-07

3.3 Install ProxySQL 1.x.x

If that is what you used to install PXC or any other **Percona** software, run the corresponding command:

On Debian or Ubuntu:

```
$ sudo apt install proxysql
```

On Red Hat Enterprise Linux or CentOS:

```
$ sudo yum install proxysql
```

To start *ProxySQL*, run the following command:

```
$ sudo service proxysql start
```

3.3.1 Do not use the default credentials

⚠ **Warning**

Do not run ProxySQL with default credentials in production.

Before starting the proxysql service, you can change the defaults in `/etc/proxysql.cnf` by changing the `admin_credentials` variable. For more information, see ProxySQL global variables.

2022-12-07

3.4 Install ProxySQL 1.X from a binary tarball

Installing *ProxySQL* from a tarball is an alternative method if the recommended method, using either the apt or the yum package manager, is not applicable in your environment.

1. In Download ProxySQL 1.x page, select the *ProxySQL* version, and select **Linux - Generic** in the *Software* field. Download the appropriate package for your platform.
2. Extract the files from the archive and change to the directory that contains the extracted files.

```
# Extract the files (assuming you have changed to the download destination directory)
$ tar xzf proxysql-VERSION-Linux-PLATFORM-ARCHITECTURE*.tar.gz
# Change to the directory that contains the extracted files
$ cd proxysql-VERSION-Linux-PLATFORM-ARCHITECTURE
```

3. Create a directory to store the *ProxySQL* data:

```
$ mkdir /home/user/data
```

4. Update the value of the `datadir` in the configuration file to point to the data directory you have created.

```
datadir="/home/user/data"
```

5. Set the other options, as needed.
6. Start *ProxySQL* with the `-c` option to pass the configuration file you have updated:

```
$ /home/user/path-to-extracted-dir/usr/bin/proxysql \
-c /home/user/path-to-extracted-dir/etc/proxysql.cnf
```

2022-12-07

3.5 Configure ProxySQL 1.x.x

This tutorial describes how to configure *ProxySQL* with three *Percona XtraDB Cluster* nodes.

Node	Host name	IP address
Node 1	pxc1	192.168.70.61
Node 2	pxc2	192.168.70.62
Node 3	pxc3	192.168.70.63
Node 4	proxysql	192.168.70.64

ProxySQL can be configured either using *etc/proxysql-admin.cnf* or using the admin interface. The admin interface can change the configuration dynamically and there is no need to restart the proxy.

To connect to the ProxySQL admin interface, use the `mysql` client. You can either connect to the admin interface from a *Percona XtraDB Cluster* node that already has the `mysql` client installed (*Node 1*, *Node 2*, *Node 3*) or install the client on *Node 4* and connect locally. For this tutorial, install *Percona XtraDB Cluster* on Node 4:

On Debian-derived distributions **On Red Hat-derived distributions**

```
[root@proxysql ~]# apt install percona-xtradb-cluster-client
```

```
[root@proxysql ~]# yum install percona-xtradb-cluster-client
```

To connect to the admin interface, use the credentials, host name, and port specified in the ProxySQL global variables.

Do not use the default credentials in production.

The following example shows how to connect to the ProxySQL admin interface with the default credentials:

```
root@proxysql:~# mysql -u admin -padmin -h 127.0.0.1 -P 6032
```

Expected output

```
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 5.1.30 (ProxySQL Admin Module)

Copyright (c) 2009-2022 Percona LLC and/or its affiliates
Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql@proxysql>
```

To see the *ProxySQL* databases and tables use the following commands:

```
mysql@proxysql> SHOW DATABASES;
```

Expected output

```
+-----+---------+-------------------------------+
| seq | name    | file                          |
+-----+---------+-------------------------------+
| 0   | main    |                               |
| 2   | disk    | /var/lib/proxysql/proxysql.db |
| 3   | stats   |                               |
| 4   | monitor |                               |
+-----+---------+-------------------------------+
4 rows in set (0.00 sec)
```

```
mysql@proxysql> SHOW TABLES;
```

Expected output

```
+--------------------------------------+
| tables                               |
+--------------------------------------+
| global_variables                     |
| mysql_collations                     |
| mysql_query_rules                    |
| mysql_replication_hostgroups         |
| mysql_servers                        |
| mysql_users                          |
| runtime_global_variables             |
| runtime_mysql_query_rules            |
| runtime_mysql_replication_hostgroups |
| runtime_mysql_servers                |
| runtime_scheduler                    |
| scheduler                            |
+--------------------------------------+
12 rows in set (0.00 sec)
```

ProxySQL has three areas where the configuration can reside:

- MEMORY (your current working place)
- RUNTIME (the production settings)
- DISK (durable configuration saved in a SQLite database)

When you change a parameter, you change it in the MEMORY area. This method allows you to test the changes before pushing the change to production (RUNTIME) or to disk.

3.5.1 Add cluster nodes to ProxySQL

To configure the backend *Percona XtraDB Cluster* nodes in *ProxySQL*, insert the corresponding records into the `mysql_servers` table.

ProxySQL uses the concept of hostgroups to group cluster nodes. This approach enables balancing the load in a cluster by routing different types of traffic to different groups.

There are many ways you can configure hostgroups (for example, master and slaves, read and write load, etc.) and a node can be a member of multiple hostgroups.

This example adds three *Percona XtraDB Cluster* nodes to the default hostgroup (`0`), which receives both write and read traffic:

```
mysql@proxysql> INSERT INTO mysql_servers(hostgroup_id, hostname, port) VALUES (0,'192.168.70.61',3306);
mysql@proxysql> INSERT INTO mysql_servers(hostgroup_id, hostname, port) VALUES (0,'192.168.70.62',3306);
mysql@proxysql> INSERT INTO mysql_servers(hostgroup_id, hostname, port) VALUES (0,'192.168.70.63',3306);
```

To see the nodes:

```
mysql@proxysql> SELECT * FROM mysql_servers;
```

Expected output

hostgroup_id	hostname	port	status	weight	compression	max
192.168.70.61	3306	ONLINE	1	0	1000	0
192.168.70.62	3306	ONLINE	1	0	1000	0
192.168.70.63	3306	ONLINE	1	0	1000	0

3.5.2 Create a ProxySQL monitoring user

To enable monitoring of *Percona XtraDB Cluster* nodes in *ProxySQL*, create a user with the `USAGE` privilege on any node in the cluster and configure the user in ProxySQL.

The following example shows how to add a monitoring user on Node 2:

```
mysql@pxc2> CREATE USER 'proxysql'@'%' IDENTIFIED BY 'ProxySQLPa55';
mysql@pxc2> GRANT USAGE ON *.* TO 'proxysql'@'%';
```

The following example shows how to configure this user on the ProxySQL node:

```
mysql@proxysql> UPDATE global_variables SET variable_value='proxysql'
              WHERE variable_name='mysql-monitor_username';
mysql@proxysql> UPDATE global_variables SET variable_value='ProxySQLPa55'
              WHERE variable_name='mysql-monitor_password';
```

To load this configuration at runtime, issue a `LOAD` command. Issue a `SAVE` command to save these changes to disk, this operation ensures that the changes persist after ProxySQL shuts down.

```
mysql@proxysql> LOAD MYSQL VARIABLES TO RUNTIME;
mysql@proxysql> SAVE MYSQL VARIABLES TO DISK;
```

Check the monitoring logs to ensure that monitoring is enabled:

```
mysql@proxysql> SELECT * FROM monitor.mysql_server_connect_log ORDER BY time_start_us DESC LIMIT 6;
```

Expected output

```
+---------------+------+------------------+----------------------+---------------+
| hostname      | port | time_start_us    | connect_success_time | connect_error |
+---------------+------+------------------+----------------------+---------------+
| 192.168.70.61 | 3306 | 1469635762434625 | 1695                 | NULL          |
| 192.168.70.62 | 3306 | 1469635762434625 | 1779                 | NULL          |
| 192.168.70.63 | 3306 | 1469635762434625 | 1627                 | NULL          |
| 192.168.70.61 | 3306 | 1469635642434517 | 1557                 | NULL          |
| 192.168.70.62 | 3306 | 1469635642434517 | 2737                 | NULL          |
| 192.168.70.63 | 3306 | 1469635642434517 | 1447                 | NULL          |
+---------------+------+------------------+----------------------+---------------+
6 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM monitor.mysql_server_ping_log ORDER BY time_start_us DESC LIMIT 6;
```

Expected output

```
+---------------+------+------------------+-------------------+------------+
| hostname      | port | time_start_us    | ping_success_time | ping_error |
+---------------+------+------------------+-------------------+------------+
| 192.168.70.61 | 3306 | 1469635762416190 | 948               | NULL       |
| 192.168.70.62 | 3306 | 1469635762416190 | 803               | NULL       |
| 192.168.70.63 | 3306 | 1469635762416190 | 711               | NULL       |
| 192.168.70.61 | 3306 | 1469635702416062 | 783               | NULL       |
| 192.168.70.62 | 3306 | 1469635702416062 | 631               | NULL       |
| 192.168.70.63 | 3306 | 1469635702416062 | 542               | NULL       |
+---------------+------+------------------+-------------------+------------+
6 rows in set (0.00 sec)
```

The previous examples show that *ProxySQL* is able to connect and ping the nodes you added.

To enable monitoring of these nodes, load them at runtime:

```
mysql@proxysql> LOAD MYSQL SERVERS TO RUNTIME;
```

3.5.3 Create a ProxySQL client user

ProxySQL must have users that can access backend nodes to manage connections.

To add a user, insert credentials into `mysql_users` table:

```
mysql@proxysql> INSERT INTO mysql_users (username,password) VALUES ('sbuser','sbpass');
```

Expected output

```
Query OK, 1 row affected (0.00 sec)
```

Note

ProxySQL currently does not encrypt passwords.

Load the user into runtime space and save these changes to disk to ensure that the user account persists after ProxySQL shuts down:

```
mysql@proxysql> LOAD MYSQL USERS TO RUNTIME;
mysql@proxysql> SAVE MYSQL USERS TO DISK;
```

To confirm that the user has been set up correctly, you can try to log in:

```
root@proxysql:~# mysql -u sbuser -psbpass -h 127.0.0.1 -P 6033
```

Expected output

```
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 1491
Server version: 5.1.30 (ProxySQL)

Copyright (c) 2009-2016 Percona LLC and/or its affiliates
Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

To provide read/write access to the cluster for ProxySQL, add this user on one of the *Percona XtraDB Cluster* nodes:

```
mysql@pxc3> CREATE USER 'sbuser'@'192.168.70.64' IDENTIFIED BY 'sbpass';
```

Expected output

```
Query OK, 0 rows affected (0.01 sec)
```

```
mysql@pxc3> GRANT ALL ON *.* TO 'sbuser'@'192.168.70.64';
```

Expected output

```
Query OK, 0 rows affected (0.00 sec)
```

3.5.4 Add Galera support in ProxySQL 1.x.x

ProxySQL 2.x.x supports monitoring the status *Percona XtraDB Cluster* nodes. ProxySQL 1.x.x can’t detect a node which isn’t in `Synced` state. To monitor the status of *Percona XtraDB Cluster* nodes in ProxySQL 1.x.x, use the script proxysql_galera_checker.

To use this script, load it into ProxySQL scheduler.

The following example shows how you can load the script for default ProxySQL 1.x.x configuration:

```
mysql> INSERT INTO scheduler (active,interval_ms,filename,arg1,comment)
VALUES (1,10000,'/usr/bin/proxysql_galera_checker','--config-file=/etc/proxysql-admin.cnf
--write-hg=10 --read-hg=11 --writer-count=1 --mode=singlewrite
--priority=192.168.100.20:3306,192.168.100.40:3306,192.168.100.10:3306,192.168.100.30:3306
--log=/var/lib/proxysql/cluster_one_proxysql_galera_check.log','cluster_one');
```

This scheduler script accepts the following options in the `arg1` argument:

Option	Name	Required	Description
`--config-file`	Configuration file	Yes	Specify `proxysql-admin` conifiguration file
`--write-hg`	HOSTGROUP WRITERS	No	Specify ProxySQL write hostgroup
`--read-hg`	HOSTGROUP READERS	No	Specify ProxySQL read hostgroup
`--writer-count`	NUMBER WRITERS	No	Specify the write node count. The options are: 0 for `loadbal` mode and `1` for `singlewrite` mode.
`--mode`	MODE	No	Specify ProxySQL read/write configuration mode
`--priority`	WRITER PRIORITY	No	Specify write notes priority
`--log`	LOG FILE	No	Specify `proxysql_galera_checker` log file

Note

Specify the cluster name in comment column.

To load the scheduler changes into the runtime space:

```
mysql@proxysql> LOAD SCHEDULER TO RUNTIME;
```

To make sure that the script has been loaded, review the runtime_scheduler table:

```
mysql@proxysql> SELECT * FROM scheduler\G
```

Expected output

```
*************************** 1. row ***************************
        id: 1
    active: 1
interval_ms: 10000
filename: /bin/proxysql_galera_checker
    arg1: --config-file=/etc/proxysql-admin.cnf --write-hg=10 --read-hg=11
            --writer-count=1 --mode=singlewrite
            --priority=192.168.100.20:3306,192.168.100.40:3306,192.168.100.10:3306,192.168.100.30:3306
            --log=/var/lib/proxysql/cluster_one_proxysql_galera_check.log
    arg2: NULL
    arg3: NULL
    arg4: NULL
    arg5: NULL
    comment: cluster_one
1 row in set (0.00 sec)
```

Review the status of available nodes:

```
mysql@proxysql> SELECT hostgroup_id,hostname,port,status FROM mysql_servers;
```

Expected output

```
+--------------+---------------+------+--------+
| hostgroup_id | hostname      | port | status |
+--------------+---------------+------+--------+
| 0            | 192.168.70.61 | 3306 | ONLINE |
| 0            | 192.168.70.62 | 3306 | ONLINE |
| 0            | 192.168.70.63 | 3306 | ONLINE |
+--------------+---------------+------+--------+
3 rows in set (0.00 sec)
```

Each node can have the following status:

ONLINE

```
Backend node is fully operational.
```

SHUNNED

```
backend node is temporarily taken out of use, because either too many
connection errors happened in a short time, or replication lag exceeded the
allowed threshold.
```

OFFLINE_SOFT

```
New incoming connections aren't accepted, while existing connections are kept
until they become inactive. In other words, connections are kept in use until
the current transaction is completed.  This allows to gracefully detach a
backend node.
```

OFFLINE_HARD

```
Existing connections are dropped, and new incoming connections aren't
accepted.  This is equivalent to deleting the node from a hostgroup, or
temporarily taking it out of the hostgroup for maintenance.
```

3.5.5 Test the cluster with sysbench

Sysbench is designed to run CPU, memory and I/O test and has the option to run Online Transaction Processing (OLTP) workloads on a MySQL database. Install `Sysbench` from Percona software repositories:

On Debian-derived distributions   **On Red Hat-derived distributions**

```
root@proxysql:~#> apt install sysbench

[root@proxysql ~]#> yum install sysbench
```

`Sysbench` requires the ProxySQL client user credentials from Create a ProxySQL client user.

1. Create a database on one of the *Percona XtraDB Cluster* nodes. Use this database for testing.

```
mysql@pxc1> CREATE DATABASE sbtest;
```

2. Populate the table with data for the benchmark on the ProxySQL node:

```
root@proxysql:~#> sysbench --report-interval=5 --num-threads=4 \
--num-requests=0 --max-time=20 \
--test=/usr/share/doc/sysbench/tests/db/oltp.lua \
--mysql-user='sbuser' --mysql-password='sbpass' \
--oltp-table-size=10000 --mysql-host=127.0.0.1 --mysql-port=6033 \
prepare
```

3. Run the benchmark on the ProxySQL node:

```
root@proxysql:~#> sysbench --report-interval=5 --num-threads=4 \
--num-requests=0 --max-time=20 \
--test=/usr/share/doc/sysbench/tests/db/oltp.lua \
--mysql-user='sbuser' --mysql-password='sbpass' \
--oltp-table-size=10000 --mysql-host=127.0.0.1 --mysql-port=6033 \
run
```

ProxySQL stores collected data in the `stats` schema:

```
mysql@proxysql> SHOW TABLES FROM stats;
```

Expected output

```
+--------------------------------+
| tables                         |
+--------------------------------+
| stats_mysql_query_rules        |
| stats_mysql_commands_counters  |
| stats_mysql_processlist        |
| stats_mysql_connection_pool    |
| stats_mysql_query_digest       |
| stats_mysql_query_digest_reset |
| stats_mysql_global             |
+--------------------------------+
```

For example, to see the number of commands that run on the cluster:

3.5.6 Automatic fail-over

ProxySQL automatically detects if a node isn't available or if the node isn't synced with the cluster.

You can check the status of all available nodes by running:

```
mysql@proxysql> SELECT hostgroup_id,hostname,port,status FROM mysql_servers;
```

Expected output

```
+--------------+---------------+------+--------+
| hostgroup_id | hostname      | port | status |
+--------------+---------------+------+--------+
| 0            | 192.168.70.61 | 3306 | ONLINE |
| 0            | 192.168.70.62 | 3306 | ONLINE |
| 0            | 192.168.70.63 | 3306 | ONLINE |
+--------------+---------------+------+--------+
3 rows in set (0.00 sec)
```

To test problem detection and fail-over mechanism, shut down Node 3:

```
root@pxc3:~# service mysql stop
```

ProxySQL detects that the node is down and updates the node’s status to `OFFLINE_SOFT`:

```
mysql@proxysql> SELECT hostgroup_id,hostname,port,status FROM mysql_servers;
```

Expected output

```
+--------------+---------------+------+--------------+
| hostgroup_id | hostname      | port | status       |
+--------------+---------------+------+--------------+
| 0            | 192.168.70.61 | 3306 | ONLINE       |
| 0            | 192.168.70.62 | 3306 | ONLINE       |
| 0            | 192.168.70.63 | 3306 | OFFLINE_SOFT |
+--------------+---------------+------+--------------+
3 rows in set (0.00 sec)
```

Start Node 3 again:

```
root@pxc3:~#> service mysql start
```

The script detects the change and marks the node as `ONLINE`:

```
mysql@proxysql> SELECT hostgroup_id,hostname,port,status FROM mysql_servers;
```

Expected output

```
+--------------+---------------+------+--------+
| hostgroup_id | hostname      | port | status |
+--------------+---------------+------+--------+
| 0            | 192.168.70.61 | 3306 | ONLINE |
| 0            | 192.168.70.62 | 3306 | ONLINE |
| 0            | 192.168.70.63 | 3306 | ONLINE |
+--------------+---------------+------+--------+
3 rows in set (0.00 sec)
```

3.5.7 Assisted maintenance mode

For maintenance, identify that node, update its status in ProxySQL to `OFFLINE_SOFT`, wait for ProxySQL to divert traffic from this node, and then initiate the shutdown or perform maintenance tasks. *Percona XtraDB Cluster* includes a special *maintenance mode* for nodes that enables you to take a node down without adjusting *ProxySQL* manually. This mode is controlled by the `pxc_maint_mode` variable, which is monitored by *ProxySQL* and can be set to one of the following values:

- `DISABLED`: This is the default state that tells ProxySQL to route traffic to the node as usual.
- `SHUTDOWN`: This state is set automatically when you initiate node shutdown.
- `MAINTENANCE`: You can manually change to this state if you need to perform maintenance on a node without shutting it down.

2022-12-07

4. Reference

4.1 Trademark policy

This Trademark Policy is to ensure that users of Percona-branded products or services know that what they receive has really been developed, approved, tested and maintained by Percona. Trademarks help to prevent confusion in the marketplace, by distinguishing one company’s or person’s products and services from another’s.

Percona owns a number of marks, including but not limited to Percona, XtraDB, Percona XtraDB, XtraBackup, Percona XtraBackup, Percona Server, and Percona Live, plus the distinctive visual icons and logos associated with these marks. Both the unregistered and registered marks of Percona are protected.

Use of any Percona trademark in the name, URL, or other identifying characteristic of any product, service, website, or other use is not permitted without Percona’s written permission with the following three limited exceptions.

First, you may use the appropriate Percona mark when making a nominative fair use reference to a bona fide Percona product.

Second, when Percona has released a product under a version of the GNU General Public License (“GPL”), you may use the appropriate Percona mark when distributing a verbatim copy of that product in accordance with the terms and conditions of the GPL.

Third, you may use the appropriate Percona mark to refer to a distribution of GPL-released Percona software that has been modified with minor changes for the sole purpose of allowing the software to operate on an operating system or hardware platform for which Percona has not yet released the software, provided that those third party changes do not affect the behavior, functionality, features, design or performance of the software. Users who acquire this Percona-branded software receive substantially exact implementations of the Percona software.

Percona reserves the right to revoke this authorization at any time in its sole discretion. For example, if Percona believes that your modification is beyond the scope of the limited license granted in this Policy or that your use of the Percona mark is detrimental to Percona, Percona will revoke this authorization. Upon revocation, you must immediately cease using the applicable Percona mark. If you do not immediately cease using the Percona mark upon revocation, Percona may take action to protect its rights and interests in the Percona mark. Percona does not grant any license to use any Percona mark for any other modified versions of Percona software; such use will require our prior written permission.

Neither trademark law nor any of the exceptions set forth in this Trademark Policy permit you to truncate, modify or otherwise use any Percona mark as part of your own brand. For example, if XYZ creates a modified version of the Percona Server, XYZ may not brand that modification as “XYZ Percona Server” or “Percona XYZ Server”, even if that modification otherwise complies with the third exception noted above.

In all cases, you must comply with applicable law, the underlying license, and this Trademark Policy, as amended from time to time. For instance, any mention of Percona trademarks should include the full trademarked name, with proper spelling and capitalization, along with attribution of ownership to Percona Inc. For example, the full proper name for XtraBackup is Percona XtraBackup. However, it is acceptable to omit the word “Percona” for brevity on the second and subsequent uses, where such omission does not cause confusion.

In the event of doubt as to any of the conditions or exceptions outlined in this Trademark Policy, please contact trademarks@percona.com for assistance and we will do our very best to be helpful.

2023-06-26

4.2 Copyright and licensing information

4.2.1 Documentation licensing

2024-02-20