Percona Monitoring and Management Documentation

2.42.0 (Jun 11, 2024)

Percona Technical Documentation Team

Percona LLC and/or its affiliates ©2024

Table of contents

1. Welcome 6
   - 1.1 Start here 7
   - 1.2 Read more 8
   - 1.3 Get expert help 8
2. Get started with PMM 10
   - 2.1 Install PMM 10
   - 2.2 Connect database 11
   - 2.3 Check database monitoring results 13
   - 2.4 Next steps 13
   - 2.5 Get expert help 13
3. Setting up 14
   - 3.1 Setting up 14
   - 3.2 Server 15
   - 3.3 Client 56
4. Get started 126
   - 4.1 Get Started 126
   - 4.2 User Interface 126
   - 4.3 Percona Alerting 134
   - 4.4 Backup and Restore 145
   - 4.5 Roles and permissions 162
   - 4.6 Query Analytics 182
   - 4.7 Advisors 206
5. How to 220
   - 5.1 How to 220
   - 5.2 Configure 220
   - 5.3 Manage users 228
   - 5.4 Upgrade 232
   - 5.5 Secure 234
   - 5.6 Optimize 236
   - 5.7 Set up PMM in High Availability (HA) mode 237
   - 5.8 Annotate 250
   - 5.9 Share dashboards and panels 251
   - 5.10 Extend Metrics 253
   - 5.11 Resolve issues 254
   - 5.12 Export PMM data with PMM Dump 259

Percona LLC and/or its affiliates ©2024

5.13 Integrate with Percona Platform 260
6. Details 265
6.1 Details 265
6.2 Architecture 265
6.3 UI components 271
6.4 PMM components and versions 275
6.5 Data handling in PMM 276
6.6 Develop Advisor checks 277
6.7 PMM Inventory 286
6.8 Dashboards 293
6.9 Commands 431
6.10 Query Analytics under the hood 455
6.11 API 458
6.12 VictoriaMetrics 462
6.13 ClickHouse 464
6.14 PostgreSQL 465
6.15 Glossary 467
7. DBaaS 469
7.1 Introduction to Database as a service (DBaaS) 469
7.2 DBaaS architecture 469
7.3 Managing Database clusters 473
7.4 DBaaS backup and restore 476
8. FAQ 479
8.1 How can I contact the developers? 479
8.2 What are the minimum system requirements? 479
8.3 How can I upgrade from version 1? 479
8.4 How to control data retention? 479
8.5 How are PMM Server logs rotated? 479
8.6 What privileges are required to monitor a MySQL instance? 480
8.7 Can I monitor multiple service instances? 480
8.8 Can I rename instances? 480
8.9 Can I add an AWS RDS MySQL or Aurora MySQL instance from a non-default AWS partition? 480
8.10 What resolution is used for metrics? 481
8.11 How do I set up Alerting? 481
8.12 How do I use a custom Prometheus configuration file? 482
8.13 How to troubleshoot an Update? 482
8.14 What are my login credentials when I try to connect to a Prometheus Exporter? 482
8.15 How to provision PMM Server with non-default admin password? 482

Percona LLC and/or its affiliates ©2024

8.16 How to change the PMM password for a default admin user? 483
8.17 How to use a non-default listen-port for pmm-admin? 483
8.18 How does PMM handle personal and confidential data? 483
8.19 Why am I getting a “User already exists” error when logging back into PMM? 483
8.20 Get expert help 484
9. Release notes 485
9.1 Release Notes 485
9.2 Percona Monitoring and Management 2.42.0 486
9.3 Percona Monitoring and Management 2.41.2 492
9.4 Percona Monitoring and Management 2.41.1 499
9.5 Percona Monitoring and Management 2.41.0 500
9.6 Percona Monitoring and Management 2.40.1 503
9.7 Percona Monitoring and Management 2.40.0 504
9.8 Percona Monitoring and Management 2.39.0 510
9.9 Percona Monitoring and Management 2.38.1 512
9.10 Percona Monitoring and Management 2.38.0 512
9.11 Percona Monitoring and Management 2.37.1 520
9.12 Percona Monitoring and Management 2.37.0 521
9.13 Percona Monitoring and Management 2.36.0 525
9.14 Percona Monitoring and Management 2.35.0 528
9.15 Percona Monitoring and Management 2.34.0 532
9.16 Percona Monitoring and Management 2.33.0 535
9.17 Percona Monitoring and Management 2.32.0 538
9.18 Percona Monitoring and Management 2.31.0 543
9.19 Percona Monitoring and Management 2.30.0 547
9.20 Percona Monitoring and Management 2.29.1 550
9.21 Percona Monitoring and Management 2.29.0 551
9.22 Percona Monitoring and Management 2.28.0 557
9.23 Percona Monitoring and Management 2.27.0 559
9.24 Percona Monitoring and Management 2.26.0 563
9.25 Percona Monitoring and Management 2.25.0 565
9.26 Percona Monitoring and Management 2.24.0 567
9.27 Percona Monitoring and Management 2.23.0 569
9.28 Percona Monitoring and Management 2.22.0 571
9.29 Percona Monitoring and Management 2.21.0 573
9.30 Percona Monitoring and Management 2.20.0 574
9.31 Percona Monitoring and Management 2.19.0 576
9.32 Percona Monitoring and Management 2.18.0 578

Percona LLC and/or its affiliates ©2024

9.33 Percona Monitoring and Management 2.17.0 580
9.34 Percona Monitoring and Management 2.16.0 582
9.35 Percona Monitoring and Management 2.15.1 585
9.36 Percona Monitoring and Management 2.15.0 585
9.37 Percona Monitoring and Management 2.14.0 588
9.38 Percona Monitoring and Management 2.13.0 590
9.39 Percona Monitoring and Management 2.12.0 592
9.40 Percona Monitoring and Management 2.11.1 594
9.41 Percona Monitoring and Management 2.11.0 594
9.42 Percona Monitoring and Management 2.10.1 595
9.43 Percona Monitoring and Management 2.10.0 596
9.44 Percona Monitoring and Management 2.9.1 597
9.45 Percona Monitoring and Management 2.9.0 598
9.46 Percona Monitoring and Management 2.8.0 599
9.47 Percona Monitoring and Management 2.7.0 600
9.48 Percona Monitoring and Management 2.6.1 601
9.49 Percona Monitoring and Management 2.6.0 602
9.50 Percona Monitoring and Management 2.5.0 603
9.51 Percona Monitoring and Management 2.4.0 604
9.52 Percona Monitoring and Management 2.3.0 606
9.53 Percona Monitoring and Management 2.2.2 607
9.54 Percona Monitoring and Management 2.2.1 608
9.55 Percona Monitoring and Management 2.2.0 609
9.56 Percona Monitoring and Management 2.1.0 611
9.57 Percona Monitoring and Management 2.0.1 613
9.58 Percona Monitoring and Management 2.0.0 614
10. Reference 616
10.1 Copyright and licensing information 616
10.2 Trademark policy 616

Percona LLC and/or its affiliates ©2024

1. Welcome

Percona Monitoring and Management (PMM) is an open-source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

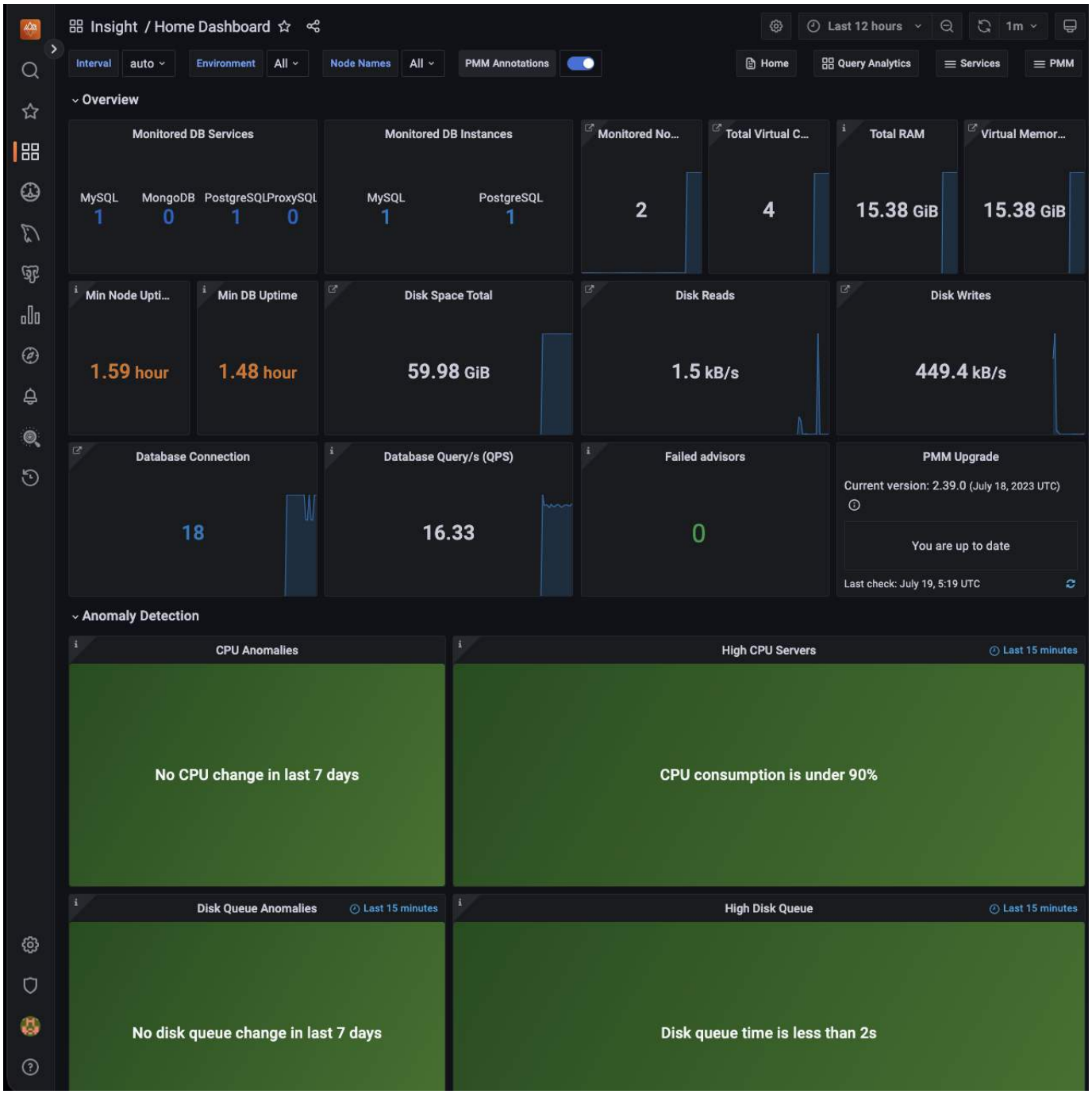
It allows you to observe the health of your database systems, explore new patterns in their behavior, troubleshoot them and perform database management operations no matter where they are located on-prem or in the cloud.

- PMM **collects** thousands of out-of-the-box performance **metrics** from databases and their hosts.
- The PMM web UI **visualizes data** in dashboards.
- Additional features include advisors for database health assessments.

This is the documentation for the latest release, **PMM 2.41.2** (Release Notes).

Here's how the home page looks on our free, live demo system.

Percona LLC and/or its affiliates ©2024



PMM runs in the cloud, on-prem, or across hybrid platforms. It's supported by our legendary expertise in open source databases, and by a vibrant developer and user community.

A minimal PMM set-up comprises one server and a client agent on every system you want to monitor.

1.1 Start here

- An easy install script, which you download, make executable and run. The script installs Docker and runs PMM Server as a container.
- The Quickstart install guide shows how to run PMM Server as a Docker container, and how to install PMM Client on Ubuntu or Red Hat Linux hosts.
- Setting Up explains in detail how to set up PMM Server, clients, and how to add services.

Percona LLC and/or its affiliates ©2024

1.2 Read more

1.2.1 Links to popular sections

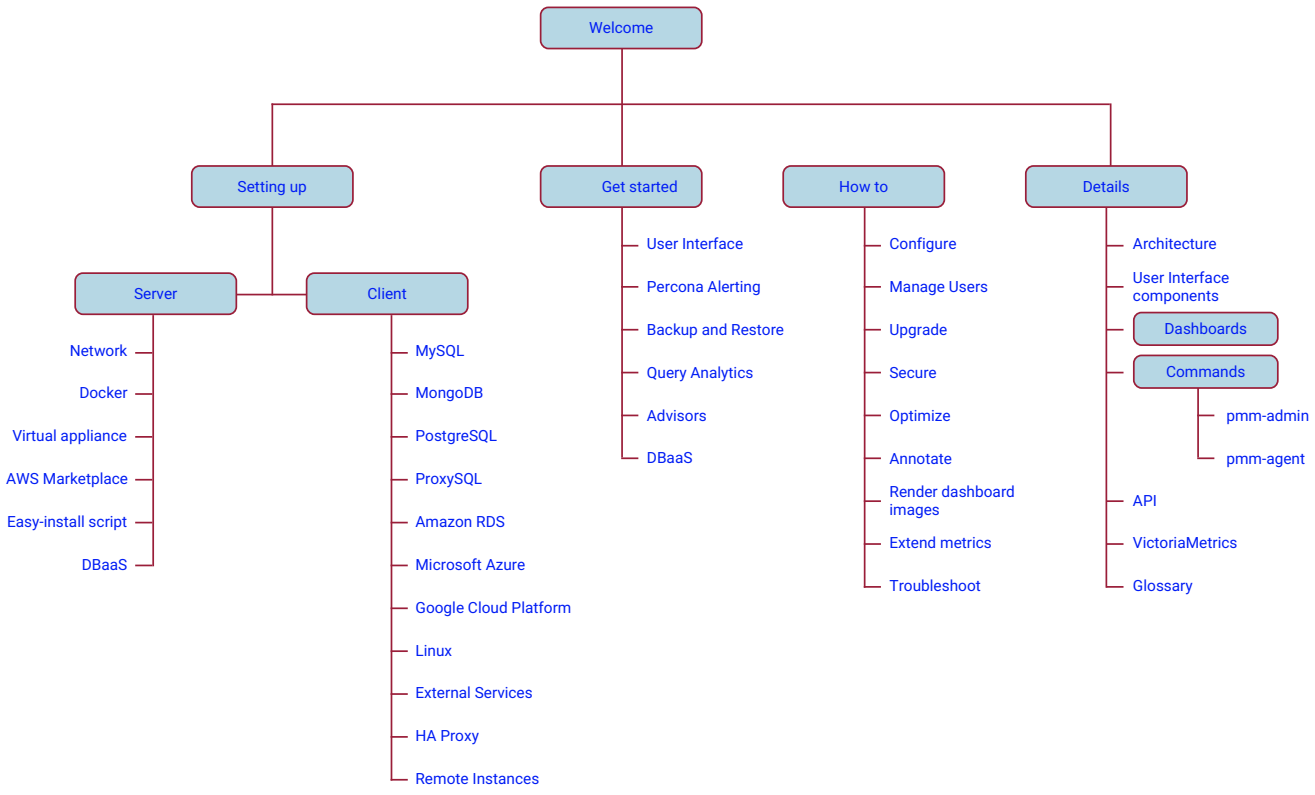
For System Administrators

- Setting up
- How to configure
- How to upgrade
- pmm-admin
- Architecture

For Users

- User interface
- Using Query Analytics
- Using Percona Alerting
- Dashboards reference

Full section map (click to show/hide)



1.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2023-08-14

Percona LLC and/or its affiliates ©2024

2. Get started with PMM

To get up and running with Percona Monitoring and Management (PMM) in no time, install PMM on Bare Metal/ Virtual using the Easy-install script for Docker.

This is the simplest and most efficient way to install PMM.

Alternative installation options

For alternative setups, explore the additional installation options detailed in the **Setting up** chapter:

- Deploy on Podman
- Deploy based on a Docker image
- Deploy on Virtual Appliance
- Deploy on Kubernetes via Helm
- Run a PMM instance hosted at AWS Marketplace

Prerequisites

Before you start installing PMM, verify that your system meets the compatibility requirements.

Verify system compatibility

- **Disk**: Approximately 1 GB of storage per monitored database node with data retention set to one week. By default, retention is 30 days.
- **Memory**: A minimum of 2 GB per monitored database node. The increase in memory usage is not proportional to the number of nodes. For example, the data from 20 nodes should be easily handled with 16 GB.
- **Ports**: Your system's firewall should allow TCP traffic on port 443.

2.1 Install PMM

The Easy-install script only runs on Linux-compatible systems. To use it, run the command with `sudo` privileges or as `root`:

0. Download and install PMM using `cURL` or `wget`:

   cURL wget

   ```
   curl -fsSL https://www.percona.com/get/pmm | /bin/bash

   wget -qO - https://www.percona.com/get/pmm | /bin/bash
   ```

0. After the installation is complete, log into PMM with the default `admin:admin` credentials.

What's happening under the hood?

This script does the following:

- Installs Docker if it is not installed on your system.
- Stops and renames any currently running PMM Docker container from `pmm-server` to `pmm-server-{timestamp}`. This old `pmm-server` container is not a recoverable backup.
- Pulls and runs the latest PMM Docker image.

Percona LLC and/or its affiliates ©2024

2.2 Connect database

Once PMM is set up, choose the database or the application that you want it to monitor:

Percona LLC and/or its affiliates ©2024

MySQL PostgreSQL MongoDB ProxySQL HAProxy

To connect a self-hosted MySQL database:

0. Create database account for PMM using the following command example. This creates a database user with name `pmm`, password `<your_password>`, and the necessary permissions:

```
CREATE USER 'pmm'@'127.0.0.1' IDENTIFIED BY '<your_password>' WITH MAX_USER_CONNECTIONS 10;
GRANT SELECT, PROCESS, REPLICATION CLIENT, RELOAD, BACKUP_ADMIN ON *.* TO 'pmm'@'127.0.0.1';
```

0. To optimize server-side resources, install PMM Client via Package Manager on the database node:

   Debian-based Red Hat-based

   Install the following with `root` permission:

   0. Install the Percona Release Tool. If this is already, make sure to update it to the latest version:

```
wget https://repo.percona.com/apt/percona-release_latest.generic_all.deb
dpkg -i percona-release_latest.generic_all.deb
```

   0. Install the PMM Client package:

```
apt update
apt install -y pmm2-client
```

   Install the following with `root` permission:

   0. Install percona-release tool. If this is already installed, update percona-release to the latest version.

```
yum install -y https://repo.percona.com/yum/percona-release-latest.noarch.rpm
```

   0. Install the PMM Client package:

```
yum install -y pmm2-client
```

0. Register PMM Client:

```
pmm-admin config --server-insecure-tls --server-url=https://admin:admin@X.X.X.X:443
```

0. Add the MySQL database using Performance Schema:

```
pmm-admin add mysql --query-source=perfschema --username=pmm --password=<your_password>
```

Alternative database connection workflows

While the default instructions above focus on connecting a self-hosted MySQL database, PMM offers the flexibility to connect to various MySQL databases, including AWS RDS, Azure MySQL or Google Cloud MySQL.

The PMM Client installation also comes with options: in addition to the installation via Package Manager described above, you can also install it as a Docker container or as a binary package. Explore alternative PMM Client installation options for more information.

Additionally, if direct access to the database node isn't available, opt to Add remote instance via User Interface instead.

To connect a PostgreSQL database:

0. Create a PMM-specific user for monitoring:

```
CREATE USER pmm WITH SUPERUSER ENCRYPTED PASSWORD '<your_password>';
```

0. Ensure that PMM can log in locally as this user to the PostgreSQL instance. To enable this, edit the `pg_hba.conf` file. If not already enabled by an existing rule, add:

```
local   all             pmm                                     md5
```


Percona LLC and/or its affiliates ©2024

2.3 Check database monitoring results

After installing PMM and connecting the database, go to the database's Instance Summary dashboard. This shows essential information about your database performance and an overview of your environment.

For more information, see PMM Dashboards.

2.4 Next steps

- Configure PMM via the interface
- Manage users in PMM
- Set up roles and permissions
- Back up and restore data in PMM

2.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-12-20

Percona LLC and/or its affiliates ©2024

3. Setting up

3.1 Setting up

There are three stages to installing and setting up PMM.

Summary

1. Set up a PMM Server.
2. Set up PMM Client(s).
3. Add services.

3.1.1 Set up PMM Server

Install and run at least one PMM Server.

Choose from:

Use	Benefits	Drawbacks
Docker	1. Quick. 2. Simple.	1. Docker installation required. 2. Additional network configuration required.
Podman	1. Quick. 2. Simple. 3. Rootless.	1. Podman installation required.
Helm	1. Quick. 2. Simple. 3. Cloud.	1. Requires running Kubernetes cluster.
Virtual appliance	1. Easily import into Hypervisor of your choice	1. More system resources compared to Docker footprint.
Amazon AWS	1. Wizard-driven install.	1. Non-free solution (infrastructure costs).

3.1.2 Set up PMM Client

Install and run PMM Client on every node where there is a service you want to monitor.

The choices:

- With Docker;
- Natively, installed from:
  - Linux package (installed with `apt`, `apt-get`, `dnf`, `yum`);
  - Binary package (a downloaded `.tar.gz` file).

Binary is only way to install PMM client without root permissions

3.1.3 Add services

On each PMM Client, you configure then add to PMM Server's inventory the node or service you want to monitor.

Percona LLC and/or its affiliates ©2024

How you do this depends on the type of service. You can monitor:

- MySQL (and variants: Percona Server for MySQL, Percona XtraDB Cluster, MariaDB);
- MongoDB;
- PostgreSQL;
- ProxySQL;
- Amazon RDS;
- Microsoft Azure;
- Google Cloud Platform (MySQL and PostgreSQL);
- Linux;
- External services;
- HAProxy;
- Remote instances.

3.1.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-03-15

3.2 Server

3.2.1 Set up PMM Server

1. Check system requirements.

   Disk

   Approximately 1 GB of storage per monitored database node with data retention set to one week. By default, retention is 30 days.

   > **Tip**
   >
   > Disable table statistics to decrease the VictoriaMetrics database size.

   Memory

   A minimum of 2 GB per monitored database node. The increase in memory usage is not proportional to the number of nodes. For example, data from 20 nodes should be easily handled with 16 GB.

   Architecture

   Your CPU must support the `SSE4.2` instruction set, a requirement of ClickHouse, a third-party column-oriented database used by Query Analytics. If your CPU is lacking this instruction set you won't be able to use Query Analytics. Additionally, since PMM 2.38.0, your CPU and any virtualization layer in use must support x86-64-v2 or your container may not start.

2. Configure your network.

Percona LLC and/or its affiliates ©2024

3. Decide how you want to run PMM Server. Choose from:
    - Docker;
    - Podman;
    - Helm;
    - Virtual appliance;
    - Amazon AWS;
    - Use the easy install script.
4. Authenticating using API keys.

    While adding clients to the PMM server, you use the `admin` user. However, if you change the password for the admin user from the PMM UI, then the clients will not be able to access PMM. Also, due to multiple unsuccessful login attempts Grafana will lock out the `admin` user. The solution is to use API key for authentication. You can use API keys as a replacement for basic authentication.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-08-16

3.2.2 Network

Ports

This is a list of ports used by the various components of PMM.

For PMM to work correctly, your system's firewall should allow TCP traffic on these ports (UDP is not needed).

Ports to expose:

PMM component	TCP port	Direction	Description
PMM Server	80	both	HTTP server, used for gRPC over HTTP and web interface (**insecure**, use with caution).
PMM Server	443	both	HTTPS server, used for gRPC over HTTPS and web interface (secure, use of SSL certificates is highly encouraged).

Other ports:

PMM component	TCP port	Direction	Description
PMM Server	7771	both	gRPC, used for communication between `pmm-agent`, `pmm-admin`.
PMM Server	7772	out	HTTP1 server, used for older links like `logs.zip`.
PMM Server	7773	out	Debugging.

Percona LLC and/or its affiliates ©2024

PMM component	TCP port	Direction	Description
`pmm-agent`	7777	out	Default `pmm-agent` listen port.
`vm-agent`	8428	both	VictoriaMetrics port.
`pmm-agent`	42000 - 51999	in	Default range for `pmm-agent` connected agents.

Important

Depending on your architecture other ports may also need to be exposed. - For `pmm-agent`, the default listen port is 7777.
- The default port range for `pmm-agent` is large by default to accomodate any architecture size but it can be modified using the `--ports-min` and `--ports-max` flags, or modifying the configuration file. In network constraint environments, the range can be reduced to a minimum by allocating at least one port per agent monitored. Learn more about available settings for `pmm-agent` in Percona PMM-Agent documentation.

Network configuration for locked-down environments

For computers in a locked-down corporate environment without direct access to the Internet, make sure to enable access to Percona Platform services following the instructions in the Percona Platform documentation.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-03-21

3.2.3 Docker

How to run PMM Server with Docker based on our Docker image.

The tags used here are for the current release. Other tags are available.

See also

Easy-install script

Before you start

- Install Docker 1.12.6 or higher.
- For PMM 2.38.0 or greater, ensure your CPU (and any virtualization layer you may be using) supports `x86-64-v2`

Percona LLC and/or its affiliates ©2024

Run

Summary

- Pull the Docker image.
- Copy it to create a persistent data container.
- Run the image.
- Open the PMM UI in a browser.

You can store data from your PMM in:

1. Docker volume (Preffered method)
2. Data container
3. Host directory

RUN DOCKER WITH VOLUME

1. Pull the image.

```
docker pull percona/pmm-server:2
```

2. Create a volume:

```
docker volume create pmm-data
```

3. Run the image:

```
docker run --detach --restart always \
--publish 443:443 \
-v pmm-data:/srv \
--name pmm-server \
percona/pmm-server:2
```

4. Change the password for the default `admin` user.
   - For PMM versions 2.27.0 and later:

```
docker exec -t pmm-server change-admin-password <new_password>
```

   - For PMM versions prior to 2.27.0:

```
docker exec -t pmm-server bash -c 'grafana-cli --homepath /usr/share/grafana --configOverrides cfg:default.paths.data=/srv/grafana admin reset-admin-password newpass'
```

5. Visit `https://localhost:443` to see the PMM user interface in a web browser. (If you are accessing the docker host remotely, replace `localhost` with the IP or server name of the host.)

RUN DOCKER WITH DATA CONTAINER

1. Create a persistent data container.

```
docker create --volume /srv \
--name pmm-data \
percona/pmm-server:2 /bin/true
```


Percona LLC and/or its affiliates ©2024

Important

PMM Server expects the data volume to be `/srv`. Using any other value will result in **data loss** when upgrading.

To check server and data container mount points:

```
docker inspect pmm-data | grep Destination && \
docker inspect pmm-server | grep Destination
```

2. Run the image.

```
docker run --detach --restart always \
--publish 443:443 \
--volumes-from pmm-data \
--name pmm-server \
percona/pmm-server:2
```

3. Change the password for the default `admin` user.

   - For PMM versions 2.27.0 and later:

```
docker exec -t pmm-server change-admin-password <new_password>
```

   - For PMM versions prior to 2.27.0:

```
docker exec -t pmm-server bash -c 'grafana-cli --homepath /usr/share/grafana --configOverrides cfg:default.paths.data=/srv/grafana admin reset-admin-password newpass'
```

4. Visit `https://localhost:443` to see the PMM user interface in a web browser. (If you are accessing the docker host remotely, replace `localhost` with the IP or server name of the host.)

RUN DOCKER WITH THE HOST DIRECTORY

Availability

This feature is available starting with PMM 2.29.0.

1. Pull the image.

```
docker pull percona/pmm-server:2
```

2. Run the image.

```
export DATA_DIR=$HOME/srv
docker run -v $DATA_DIR/srv:/srv -d --restart always --publish 80:80 --publish 443:443 --name pmm-server percona/pmm-server:2
```

`DATA_DIR` is a directory where you want to store the state for PMM.

3. Visit `https://localhost:443` to see the PMM user interface in a web browser. (If you are accessing the docker host remotely, replace `localhost` with the IP or server name of the host.)

MIGRATE FROM DATA CONTAINER TO HOST DIRECTORY/VOLUME

To migrate your PMM from data container to host directory or volume run the following command:

```
docker cp <containerId>:/srv /target/host/directory
```


Percona LLC and/or its affiliates ©2024

Backup

Summary

- Stop and rename the `pmm-server` container.
- Take a local copy of the `pmm-data` container's `/srv` directory.

Important

Grafana plugins have been moved to the data volume `/srv` since the 2.23.0 version. So if you are upgrading PMM from any version before 2.23.0 and have installed additional plugins then plugins should be installed again after the upgrade.

To check used grafana plugins:

```
docker exec -it pmm-server ls /var/lib/grafana/plugins
```

1. Stop the container.

   ```
   docker stop pmm-server
   ```

2. Move the image.

   ```
   docker rename pmm-server pmm-server-backup
   ```

3. Create a subdirectory (e.g., `pmm-data-backup`) and move to it.

   ```
   mkdir pmm-data-backup && cd pmm-data-backup
   ```

4. Backup the data.

   ```
   docker cp pmm-data:/srv .
   ```

Upgrade

Summary

- Stop the running container.
- Backup (rename) the container and copy data.
- Pull the latest Docker image.
- Run it.

Important

Downgrades are not possible. To go back to using a previous version you must have created a backup of it before upgrading.

Percona LLC and/or its affiliates ©2024

Tip

To see what release you are running, use the *PMM Upgrade* panel on the *Home Dashboard*, or run:

```
docker exec -it pmm-server \
curl -ku admin:admin https://localhost/v1/version
```

(If you are accessing the docker host remotely, replace `localhost` with the IP or server name of the host.)

1. Stop the container.

   ```
   docker stop pmm-server
   ```

2. Perform a backup.
3. Pull the latest image.

   ```
   docker pull percona/pmm-server:2
   ```

4. Rename the original container

   ```
   docker rename pmm-server pmm-server-old
   ```

5. Run it.

   ```
   docker run \
   --detach \
   --restart always \
   --publish 443:443 \
   --volumes-from pmm-data \
   --name pmm-server \
   percona/pmm-server:2
   ```

Restore

Summary

- Stop and remove the container.
- Restore (rename) the backup container.
- Restore saved data to the data container.
- Restore permissions to the data.

Important

You must have a backup to restore from.

1. Stop the container.

   ```
   docker stop pmm-server
   ```

2. Remove it.

   ```
   docker rm pmm-server
   ```


Percona LLC and/or its affiliates ©2024

3. Revert to the saved image.

```
docker rename pmm-server-backup pmm-server
```

4. Change directory to the backup directory (e.g. `pmm-data-backup`).
5. Remove Victoria Metrics data folder.

```
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 rm -r /srv/victoriametrics/data
```

6. Copy the data.

```
docker cp srv pmm-data:/
```

7. Restore permissions.

```
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 chown -R root:root /srv && \
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 chown -R pmm:pmm /srv/alertmanager && \
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 chown -R root:pmm /srv/clickhouse && \
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 chown -R grafana:grafana /srv/grafana && \
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 chown -R pmm:pmm /srv/logs && \
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 chown -R postgres:postgres /srv/postgres14 && \
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 chown -R pmm:pmm /srv/prometheus && \
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 chown -R pmm:pmm /srv/victoriametrics && \
docker run --rm --volumes-from pmm-data -it percona/pmm-server:2 chown -R postgres:postgres /srv/logs/postgresql14.log
```

8. Start the image.

```
docker start pmm-server
```

Remove

Summary

- Stop the container.
- Remove (delete) both the server and data containers.
- Remove (delete) both images.

Caution

These steps delete the PMM Server Docker image and any accumulated PMM metrics data.

1. Stop pmm-server container.

```
docker stop pmm-server
```

2. Remove containers.

```
docker rm pmm-server pmm-data
```

3. Remove the image.

```
docker rmi $(docker images | grep "percona/pmm-server" | awk {'print $3'})
```


Percona LLC and/or its affiliates ©2024

Environment variables

Use the following Docker container environment variables (with `-e var=value`) to set PMM Server parameters.

Variable	Description
`DISABLE_UPDATES`	Disables a periodic check for new PMM versions as well as ability to apply upgrades using the UI
`DISABLE_TELEMETRY`	Disable built-in telemetry and disable STT if telemetry is disabled.
`METRICS_RESOLUTION`	High metrics resolution in seconds.
`METRICS_RESOLUTION_HR`	High metrics resolution (same as above).
`METRICS_RESOLUTION_MR`	Medium metrics resolution in seconds.
`METRICS_RESOLUTION_LR`	Low metrics resolution in seconds.
`DATA_RETENTION`	The number of days to keep time-series data. **N.B.** This must be set in a format supported by `time.ParseDuration` and represent the complete number of days. The supported units are `ns`, `us` (or `µs`), `ms`, `s`, `m`, and `h`. The value must be a multiple of 24, e.g., for 90 days 2160h (90 * 24).
`ENABLE_VM_CACHE`	Enable cache in VM.
`DISABLE_ALERTING`	Disables built-in Percona Alerting, which is enabled by default.
`ENABLE_AZUREDISCOVER`	Enable support for discovery of Azure databases.
`DISABLE_BACKUP_MANAGEMENT`	Disables Backup Management, which is enabled by default.
`ENABLE_DBAAS`	Enable DBaaS features.
`PMM_DEBUG`	Enables a more verbose log level.
`PMM_TRACE`	Enables a more verbose log level including trace-back information.
`PMM_PUBLIC_ADDRESS`	External IP address or the DNS name on which PMM server is running.

The following variables are also supported but values passed are not verified by PMM. If any other variable is found, it will be considered invalid and the server won't start.

Variable	Description
`_`, `HOME`, `HOSTNAME`, `LANG`, `PATH`, `PWD`, `SHLVL`, `TERM`	Default environment variables.
`GF_*`	Grafana environment variables.
`VM_*`	VictoriaMetrics' environment variables.
Note that environment variables inherit their names from the command line flags. To find out which variables are available to you, see the full list of CLI command flags.	

| `SUPERVISOR_` | `supervisord` environment variables. | `KUBERNETES_` | Kubernetes environment variables. | `MONITORING_` | Kubernetes monitoring environment variables. | `PERCONA_TEST_` | Unknown variable but won't prevent the server starting. | `PERCONA_TEST_DBAAS` | Deprecated. Use `ENABLE_DBAAS`.

Percona LLC and/or its affiliates ©2024

Preview environment variables

> **Warning**
>
> The `PERCONA_TEST_*` environment variables are experimental and subject to change. It is recommended that you use these variables for testing purposes only and not on production.

Variable	Description
`PERCONA_TEST_SAAS_HOST`	SaaS server hostname.
`PERCONA_TEST_PMM_CLICKHOUSE_ADDR`	Name of the host and port of the external ClickHouse database instance.
`PERCONA_TEST_PMM_CLICKHOUSE_DATABASE`	Database name of the external ClickHouse database instance.
`PERCONA_TEST_PMM_CLICKHOUSE_POOL_SIZE`	The maximum number of threads in the current connection thread pool. This value cannot be bigger than max_thread_pool_size.
`PERCONA_TEST_PMM_CLICKHOUSE_BLOCK_SIZE`	The number of rows to load from tables in one block for this connection.

Tips

- To Disable the Home Dashboard *PMM Upgrade* panel you can either add `-e DISABLE_UPDATES=true` to the `docker run` command (for the life of the container) or navigate to *PMM → PMM Settings → Advanced Settings* and disable "Check for Updates" (can be turned back on by any admin in the UI).
- Eliminate browser certificate warnings by configuring a trusted certificate.
- You can optionally enable an (insecure) HTTP connection by adding `--publish 80:80` to the `docker run` command. However, running PMM insecure is not recommended. You should also note that PMM Client *requires* TLS to communicate with the server, only working on a secure port.

ISOLATED HOSTS

If the host where you will run PMM Server has no internet connection, you can download the Docker image on a separate (internet-connected) host and securely copy it.

1. On an internet-connected host, download the Docker image and its checksum file.

   ```
   wget https://downloads.percona.com/downloads/pmm2/2.41.2/docker/pmm-server-2.41.2.docker
   wget https://downloads.percona.com/downloads/pmm2/2.41.2/docker/pmm-server-2.41.2.sha256sum
   ```

2. Copy both files to where you will run PMM Server.
3. Open a terminal on the PMM Server host.
4. (Optional) Check the Docker image file integrity.

   ```
   shasum -ca 256 pmm-server-2.41.2.sha256sum
   ```

5. Load the image.

   ```
   docker load -i pmm-server-2.41.2.docker
   ```

6. Run the container as if your image is already pulled using your desired method for a storage volume (you can step over any docker pull commands as the image has been pre-staged).

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2024-03-21

3.2.4 Podman

How to run PMM Server with Podman on our Docker image

The tags used here are for the current release (PMM 2.33.0). Other tags are available.

See also

Docker

Podman is an open-source project available on most Linux platforms and resides on GitHub. Podman is a daemonless container engine for developing, managing, and running Open Container Initiative (OCI) containers and container images on your Linux System.

Non-privileged users could run containers under the control of Podman.

It could be just aliased (`alias docker=podman`) with docker and work with the same way. All instructions from Docker section also apply here.

Percona recommends running PMM as a non-privileged user and running it as part of the SystemD service provided. SystemD service ensures that the service is running and maintains logs and other management features (start, stop, etc.).

Before you start

- Install Podman.
- Configure rootless Podman.

Run as non-privileged user to start PMM

Availability

This feature is available starting with PMM 2.29.0.

Summary

- Install.
- Configure.
- Enable and Start.
- Open the PMM UI in a browser.

1. Install.

Percona LLC and/or its affiliates ©2024

Create `~/.config/systemd/user/pmm-server.service` file:

```
mkdir -p ~/.config/systemd/user/
cat << "EOF" > ~/.config/systemd/user/pmm-server.service
[Unit]
Description=pmm-server
Wants=network-online.target
After=network-online.target
After=nss-user-lookup.target nss-lookup.target
After=time-sync.target

[Service]
Type=simple

# set environment for this unit
Environment=PMM_PUBLIC_PORT=8443
Environment=PMM_VOLUME_NAME=%N
Environment=PMM_TAG=2.33.0
Environment=PMM_IMAGE=docker.io/percona/pmm-server
Environment=PMM_ENV_FILE=%h/.config/pmm-server/pmm-server.env

# optional env file that could override previous env settings for this unit
EnvironmentFile=-%h/.config/pmm-server/env

ExecStart=/usr/bin/podman run --rm --replace=true --name=%N -p ${PMM_PUBLIC_PORT}:443/tcp --ulimit=host --volume=${PMM_VOLUME_NAME}:/srv --env-file=$
{PMM_ENV_FILE} --health-cmd=none --health-interval=disable ${PMM_IMAGE}:${PMM_TAG}
ExecStop=/usr/bin/podman stop -t 10 %N
Restart=on-failure
RestartSec=20

[Install]
Alias=%N
WantedBy=default.target

EOF
```

Create `~/.config/pmm-server/pmm-server.env` file:

```
mkdir -p ~/.config/pmm-server/
cat << "EOF" > ~/.config/pmm-server/pmm-server.env
# env file passed to the container
# full list of environment variables:
# https://www.percona.com/doc/percona-monitoring-and-management/2.x/setting-up/server/docker.html#environment-variables

# keep updates disabled
# do image replacement instead (update the tag and restart the service)
DISABLE_UPDATES=1

# Enable DBaaS feature
#ENABLE_DBAAS=1
EOF
```

2. Configure.

   There are 2 configuration files: 1. `~/.config/pmm-server/pmm-server.env` defines environment variables for PMM Server (PMM parameters like DBaaS feature and etc) 2. `~/.config/pmm-server/env` defines environment variables for SystemD service (image tags, repo and etc)

   SystemD service passes the environment parameters from the `pmm-server.env` file (in `~/.config/pmm-server/pmm-server.env`) to PMM. For more information about container environment variables, check Docker Environment.

   SystemD service uses some environment variables that could be customized if needed:

   ```
   Environment=PMM_PUBLIC_PORT=8443
   Environment=PMM_VOLUME_NAME=%N
   Environment=PMM_TAG=2.33.0
   Environment=PMM_IMAGE=docker.io/percona/pmm-server
   ```

   You can override the environment variables by defining them in the file `~/.config/pmm-server/env`. For example, to override the path to a custom registry `~/.config/pmm-server/env`:

   ```
   mkdir -p ~/.config/pmm-server/
   cat << "EOF" > ~/.config/pmm-server/env
   PMM_TAG=2.31.0
   ```


Percona LLC and/or its affiliates ©2024

```
PMM_IMAGE=docker.io/percona/pmm-server
PMM_PUBLIC_PORT=8443
EOF
```

Important

Ensure that you modify PMM_TAG in `~/.config/pmm-server/env` and update it regularly as Percona cannot update it. It needs to be done by you.

3. Enable and Start.

```
systemctl --user enable --now pmm-server
```

4. Visit `https://localhost:8443` to see the PMM user interface in a web browser. (If you are accessing host remotely, replace `localhost` with the IP or server name of the host.)

Backup

Summary

- Stop PMM server.
- Backup the data.

Important

Grafana plugins have been moved to the data volume `/srv` since the 2.23.0 version. So if you are upgrading PMM from any version before 2.23.0 and have installed additional plugins then plugins should be installed again after the upgrade. To check used grafana plugins: `podman exec -it pmm-server ls /var/lib/grafana/plugins`

1. Stop PMM server.

```
systemctl --user stop pmm-server
```

2. Backup the data.

```
podman volume export pmm-server --output pmm-server-backup.tar
```

Important

If you changed the default name to `PMM_VOLUME_NAME` environment variable, use that name after `export` instead of `pmm-server` (which is the default volume name).

Percona LLC and/or its affiliates ©2024

Upgrade

Summary

- Perform a backup.
- Update PMM tag.
- Pre-pull image.
- Run it.

Important

You cannot downgrade. To go to a previous version, you must create a backup before upgrading.

Tip

To see the current release running on your system, use the *PMM Upgrade* panel on the *Home Dashboard*, or run:

```
podman exec -it pmm-server \
curl -ku admin:admin https://localhost/v1/version
```

(If you are accessing the podman host remotely, replace `localhost` with the IP or server name of the host.)

1. Perform a backup.
2. Update PMM tag.

   Edit `~/.config/pmm-server/env` and create/update with a new tag from latest release:

   ```
   sed -i "s/PMM_TAG=.*/PMM_TAG=2.33.0/g" ~/.config/pmm-server/env
   ```

3. Pre-pull image for faster restart.

   ```
   source ~/.config/pmm-server/env
   podman pull ${PMM_IMAGE}:${PMM_TAG}
   ```

4. Run PMM.

   ```
   systemctl --user restart pmm-server
   ```

Restore

Summary

- Stop PMM server.
- Run PMM on the previous image.
- Restore the volume.
- Start PMM Server.

Percona LLC and/or its affiliates ©2024

Important

You must have a backup to restore from. You need to perform restore only if you have issues with upgrade or with the data.

1. Stop PMM server.

```
systemctl --user stop pmm-server
```

2. Run PMM on the previous image.

   Edit `~/.config/pmm-server/env` file:

```
sed -i "s/PMM_TAG=.*/PMM_TAG=2.31.0/g" ~/.config/pmm-server/env
```

   Important

   X.Y.Z (2.31.0) is the version you used before upgrade and you made Backup with it

3. Restore the volume.

```
podman volume import pmm-server pmm-server-backup.tar
```

4. Start PMM Server.

```
systemctl --user start pmm-server
```

Remove

Summary

- Stop PMM server.
- Remove (delete) volume.
- Remove (delete) images.

Caution

These steps delete the PMM Server Docker image and the associated PMM metrics data.

1. Stop PMM server.

```
systemctl --user stop pmm-server
```

2. Remove volume.

```
podman volume rm --force pmm-server
```

3. Remove the PMM images.

```
podman rmi $(podman images | grep "pmm-server" | awk {'print $3'})
```


Percona LLC and/or its affiliates ©2024

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-12-14

3.2.5 Helm

Helm is the package manager for Kubernetes. Percona Helm charts can be found in percona/percona-helm-charts repository on Github.

Before you start

- Install Helm following its official installation instructions.
- Kubernetes cluster that Helm supports

> Helm v3 is needed to run the following steps.

Refer to Kubernetes Supported versions and Helm Version Support Policy to find the supported versions.

PMM should be platform-agnostic, but it requires escalated privileges inside a container. It is necessary to have a `root` user inside the PMM container. Thus, PMM would not work for Kubernetes Platforms such as OpenShift or others that have hardened Security Context Constraints, for example:

- Security context constraints (SCCs)
- Managing security context constraints

Kubernetes platforms offer a different set of capabilities. To use PMM in production, you would need backups and, thus storage driver that supports snapshots. Consult your provider for Kubernetes and Cloud storage capabilities.

Locality and Availability

You should not run the PMM monitoring server along with the monitored database clusters and services on the same system.

Please ensure proper locality either by physically separating workloads in Kubernetes clusters or running separate Kubernetes clusters for the databases and monitoring workloads.

You can physically separate workloads by properly configuring Kubernetes nodes, affinity rules, label selections, etc.

Also, ensure that the Kubernetes cluster has high availability so that in case of a node failure, the monitoring service will be running and capturing the required data.

Use Helm to install PMM server on Kubernetes clusters

> **Availability**
>
> This feature is available starting with PMM 2.29.0.

Percona LLC and/or its affiliates ©2024

Summary

- Setup PMM admin password
- Install
- Configuration parameters
- PMM environment variables
- PMM SSL certificates
- Backup
- Upgrade
- Restore
- Uninstall

SETUP PMM ADMIN PASSWORD

Create Kubernetes secret with PMM admin password:

```
cat <<EOF | kubectl create -f -
apiVersion: v1
kind: Secret
metadata:
  name: pmm-secret
  labels:
    app.kubernetes.io/name: pmm
type: Opaque
data:
# base64 encoded password
# encode some password: `echo -n "admin" | base64`
  PMM_ADMIN_PASSWORD: YWRtaW4=
EOF
```

To get admin password execute:

```
kubectl get secret pmm-secret -o jsonpath='{.data.PMM_ADMIN_PASSWORD}' | base64 --decode
```

INSTALL

To install the chart with the release name `pmm`:

```
helm repo add percona https://percona.github.io/percona-helm-charts/
helm install pmm \
--set secret.create=false \
--set secret.name=pmm-secret \
percona/pmm
```

The command deploys PMM on the Kubernetes cluster in the default configuration and specified secret. The Parameters section lists the parameters that can be configured during installation.

Tip

List all releases using `helm list`.

PARAMETERS

The list of Parameters is subject to change from release to release. Check the Parameters section of the PMM Helm Chart.

Percona LLC and/or its affiliates ©2024

Tip

You can list the default parameters values.yaml or get them from chart definition: `helm show values percona/pmm`

Specify each parameter using the `--set key=value[,key=value]` or `--set-string key=value[,key=value]` arguments to `helm install`. For example,

```
helm install pmm \
--set secret.create=false --set secret.name=pmm-secret \
--set-string pmmEnv.DISABLE_UPDATES="1" \
--set service.type="NodePort" \
--set storage.storageClassName="linode-block-storage-retain" \
    percona/pmm
```

The command above installs PMM, configuring the service network type as `NodePort` and setting the storage class to `linode-block-storage-retain` for persistent storage on LKE.```

Important

Once this chart is deployed, it is impossible to change the application's access credentials, such as password, using Helm. To change these application credentials after deployment, delete any persistent volumes (PVs) used by the chart and re-deploy it, or use the application's built-in administrative tools (if available)

Alternatively, a YAML file that specifies the values for the above parameters can be provided while installing the chart. For example:

```
helm show values percona/pmm > values.yaml

#change needed parameters in values.yaml, you need `yq` tool pre-installed
yq -i e '.secret.create |= false' values.yaml

helm install pmm -f values.yaml percona/pmm
```

PMM ENVIRONMENT VARIABLES

In case you want to add extra environment variables (useful for advanced operations like custom init scripts), you can use the `pmmEnv` property.

```
pmmEnv:
  DISABLE_UPDATES: "1"
```

PMM SSL CERTIFICATES

PMM ships with self signed SSL certificates to provide secure connection between client and server (check here).

You will see the warning when connecting to PMM. To further increase security, you should provide your certificates and add values of credentials to the fields of the `cert` section:

```
certs:
  name: pmm-certs
  files:
    certificate.crt: <content>
    certificate.key: <content>
    ca-certs.pem: <content>
    dhparam.pem: <content>
```

Another approach to set up TLS certificates is to use the Ingress controller, see TLS. PMM helm chart supports Ingress. See PMM network configuration.

Percona LLC and/or its affiliates ©2024

Backup

PMM helm chart uses PersistentVolume and PersistentVolumeClaim to allocate storage in the Kubernetes cluster.

Volumes could be pre-provisioned and dynamic. PMM chart supports both and exposes it through PMM storage configuration.

Backups for the PMM server currently support only storage layer backups and thus require StorageClass and VolumeSnapshotClass.

Validate the correct configuration by using these commands:

```
kubectl get sc
kubectl get volumesnapshotclass
```

Storage

Storage configuration is Hardware and Cloud specific. There could be additional costs associated with Volume Snapshots. Check the documentation for your Cloud or for your Kubernetes cluster.

Before taking a VolumeSnapshot, stop the PMM server. In this step, we will stop PMM (scale to 0 pods), take a snapshot, wait until the snapshot completes, then start PMM server (scale to 1 pod):

```
kubectl scale statefulset pmm --replicas=0
kubectl wait --for=jsonpath='{.status.replicas}'=0 statefulset pmm

cat <<EOF | kubectl create -f -
apiVersion: snapshot.storage.k8s.io/v1
kind: VolumeSnapshot
metadata:
  name: before-v2.34.0-upgrade
  labels:
    app.kubernetes.io/name: pmm
spec:
  volumeSnapshotClassName: csi-hostpath-snapclass
  source:
    persistentVolumeClaimName: pmm-storage-pmm-0
EOF

kubectl wait --for=jsonpath='{.status.readyToUse}'=true VolumeSnapshot/before-v2.34.0-upgrade
kubectl scale statefulset pmm --replicas=1
```

Output:

```
statefulset.apps/pmm scaled
statefulset.apps/pmm condition met
volumesnapshot.snapshot.storage.k8s.io/before-v2.34.0-upgrade created
volumesnapshot.snapshot.storage.k8s.io/before-v2.34.0-upgrade condition met
statefulset.apps/pmm scaled
```

PMM scale

Only one replica set is currently supported.

You can view available snapshots by executing the following command:

```
kubectl get volumesnapshot
```

UPGRADES

Percona will release a new chart updating its containers if a new version of the main container is available, there are any significant changes, or critical vulnerabilities exist.

Percona LLC and/or its affiliates ©2024

By default the UI update feature is disabled and should not be enabled. Do not modify that parameter when customizing the `values.yaml` file:

```
pmmEnv:
  DISABLE_UPDATES: "1"
```

Before updating the helm chart, it is recommended to pre-pull the image on the node where PMM is running, as the PMM images could be large and could take time to download.

Update PMM as follows:

```
helm repo update percona
helm upgrade pmm -f values.yaml percona/pmm
```

This will check updates in the repo and upgrade deployment if the updates are available.

Restore

The version of the PMM server should be greater than or equal to the version in a snapshot. To restore from the snapshot, delete the old deployment first:

```
helm uninstall pmm
```

And then use snapshot configuration to start the PMM server again with the correct version and correct storage configuration:

```
helm install pmm \
--set image.tag="2.34.0" \
--set storage.name="pmm-storage-old" \
--set storage.dataSource.name="before-v2.34.0-upgrade" \
--set storage.dataSource.kind="VolumeSnapshot" \
--set storage.dataSource.apiGroup="snapshot.storage.k8s.io" \
--set secret.create=false \
--set secret.name=pmm-secret \
percona/pmm
```

Here, we created a new `pmm-storage-old` PVC with data from the snapshot. So, there are a couple of PV and PVCs available in a cluster.

```
$ kubectl get pvc
NAME                   STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS      AGE
pmm-storage-old-pmm-0   Bound    pvc-70e5d2eb-570f-4087-9515-edf2f051666d   10Gi       RWO            csi-hostpath-sc   3s
pmm-storage-pmm-0       Bound    pvc-9dbd9160-e4c5-47a7-bd90-bff36fc1463e   10Gi       RWO            csi-hostpath-sc   89m

$ kubectl get pv
NAME                                       CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                              STORAGECLASS      REASON   AGE
pvc-70e5d2eb-570f-4087-9515-edf2f051666d   10Gi       RWO            Delete           Bound    default/pmm-storage-old-pmm-0      csi-hostpath-sc            4m50s
pvc-9dbd9160-e4c5-47a7-bd90-bff36fc1463e   10Gi       RWO            Delete           Bound    default/pmm-storage-pmm-0          csi-hostpath-sc            93m
```

Delete unneeded PVC when you are sure you don't need them.

UNINSTALL

To uninstall `pmm` deployment:

```
helm uninstall pmm
```

This command takes a release name and uninstalls the release.

It removes all resources associated with the last release of the chart as well as the release history.

Helm will not delete PVC, PV, and any snapshots. Those need to be deleted manually.

Percona LLC and/or its affiliates ©2024

Also, delete PMM `Secret` if no longer required:

```
kubectl delete secret pmm-secret
```

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-05-20

3.2.6 Virtual Appliance

How to run PMM Server as a virtual machine.

Summary

- Download and verify the latest OVF file.
- Import it.
- Reconfigure network.
- Start the VM and get IP.
- Log into PMM UI.
- (Optional) Change VM root password.
- (Optional) Set up SSH.
- (Optional) Set up static IP.

Most steps can be done with either a user interface or on the command line, but some steps can only be done in one or the other. Sections are labelled **UI** for user interface or **CLI** for command line instructions.

Terminology

- *Host* is the desktop or server machine running the hypervisor.
- *Hypervisor* is software (e.g. VirtualBox, VMware) that runs the guest OS as a virtual machine.
- *Guest* is the CentOS virtual machine that runs PMM Server.

OVA file details

Item	Value
Download page	https://www.percona.com/downloads/pmm2/2.41.2/ova
File name	`pmm-server-2.41.2.ova`
VM name	`PMM2-Server-2024-03-20-N` (`N`=build number)

Percona LLC and/or its affiliates ©2024

VM specifications

Component	Value
OS	CentOS 7.9 (64-bit)
CPU	1
Base memory	4096 MB
Disks	LVM, 2 physical volumes
Disk 1 (`sda`)	VMDK (SCSI, 40 GB)
Disk 2 (`sdb`)	VMDK (SCSI, 400 GB)

Users

Default Username	Default password
`root`	`percona`
`admin`	`admin`

Download

UI

1. Open a web browser.
2. Visit the PMM Server download page.
3. Choose a *Version* or use the default (the latest).
4. Click the link for `pmm-server-2.41.2.ova` to download it. Note where your browser saves it.
5. Right-click the link for `pmm-server-2.41.2.sha256sum` and save it in the same place as the `.ova` file.
6. (Optional) Verify.

CLI

Download the latest PMM Server OVA and checksum files.

```
wget https://www.percona.com/downloads/pmm2/2.41.2/ova/pmm-server-2.41.2.ova
wget https://www.percona.com/downloads/pmm2/2.41.2/ova/pmm-server-2.41.2.sha256sum
```

Verify

CLI

Verify the checksum of the downloaded .ova file.

```
shasum -ca 256 pmm-server-2.41.2.sha256sum
```

VMware

IMPORT

UI

1. Select *File → Import*.
2. Click *Choose file…*.
3. Navigate to the downloaded `.ova` file and select it.

Percona LLC and/or its affiliates ©2024

4. Click *Open*.
5. Click *Continue*.
6. In the *Save as* dialog:

   a. (Optional) Change the directory or file name.

   b. Click *Save*.
7. Choose one of:
   - (Optional) Click *Finish*. This starts the virtual machine.
   - (Recommended) Click *Customize Settings*. This opens the VM's settings page without starting the machine.

CLI

1. Install `ovftool`. (You need to register.)
2. Import and convert the OVA file. (`ovftool` can't change CPU or memory settings during import, but it can set the default interface.)

   Choose one of:

   - Download and import the OVA file.

     ```
     ovftool --name="PMM Server" --net:NAT=Wi-Fi \
     https://www.percona.com/downloads/pmm2/2.41.2/ova/pmm-server-2.41.2.ova \
     pmm-server-2.41.2.vmx
     ```

   - Import an already-downloaded OVA file.

     ```
     ovftool --name="PMM Server" --net:NAT=WiFi \
     pmm-server-2.41.2.ova \
     pmm-server.vmx
     ```

RECONFIGURE INTERFACE

When using the command line, the interface is remapped during import.

UI

1. If started, shut down the virtual machine.
2. In the VMware main window, select the imported virtual machine.
3. Click *Virtual Machine → Settings…*.
4. Click *Network Adapter*.
5. In the *Bridged Networking* section, select *Autodetect*.
6. Close the settings window.

START GUEST AND GET IP ADDRESS

UI

1. In the VMware main window, select the imported virtual machine.
2. Click the play button or select *Virtual Machine → Start Up*.
3. When the instance has been booted, note the IP address in the guest console.

CLI/UI

1. Start the virtual machine in GUI mode. (There's no way to redirect a VMware VM's console to the host.)

Percona LLC and/or its affiliates ©2024

```
vmrun -gu root -gp percona start \
pmm-server.vmx gui
```

2. When the instance has been booted, note the IP address in the guest console.
3. (Optional) Stop and restart the instance in headless mode.

```
vmrun stop pmm-server.vmx
vmrun -gu root -gp percona start \
pmm-server.vmx nogui
```

VirtualBox

IMPORT

UI

1. Select *File → Import appliance…*.
2. In the *File* field, type the path to the downloaded `.ova` file, or click the folder icon to navigate and open it.
3. Click *Continue*.
4. On the *Appliance settings* page, review the settings and click *Import*.
5. Click *Start*.
6. When the guest has booted, note the IP address in the guest console.

CLI

1. Open a terminal and change the directory to where the downloaded `.ova` file is.
2. (Optional) Do a ‘dry run’ import to see what values will be used.

```
VBoxManage import pmm-server-2.41.2.ova --dry-run
```

3. Import the image. Choose one of:

   - With the default settings.

   ```
   VBoxManage import pmm-server-2.41.2.ova
   ```

   - With custom settings (in this example, Name: “PMM Server”, CPUs: 2, RAM: 8192 MB).

   ```
   VBoxManage import --vsys 0 --vmname "PMM Server" \
   --cpus 2 --memory 8192 pmm-server-2.41.2.ova
   ```

INTERFACE

UI

1. Click *Settings*.
2. Click *Network*.
3. In the *Adapter 1* field, click *Attached to* and change to *Bridged Adapter*.
4. In the *Name* field, select your host’s active network interface (e.g. `en0: Wi-Fi (Wireless)`).
5. Click *OK*.

CLI

1. Show the list of available bridge interfaces.

Percona LLC and/or its affiliates ©2024

```
VBoxManage list bridgedifs
```

2. Find the name of the active interface you want to bridge to (one with *Status: Up* and a valid IP address). Example: `en0: Wi-Fi (Wireless)`
3. Bridge the virtual machine's first interface (`nic1`) to the host's `en0` ethernet adapter.

```
VBoxManage modifyvm 'PMM Server' \
--nic1 bridged --bridgeadapter1 'en0: Wi-Fi (Wireless)'
```

4. Redirect the console output into a host file.

```
VBoxManage modifyvm 'PMM Server' \
--uart1 0x3F8 4 --uartmode1 file /tmp/pmm-server-console.log
```

GET IP

UI

1. Select the *PMM Server* virtual machine in the list.
2. Click *Start*.
3. When the guest has booted, note the IP address in the guest console.

CLI

1. Start the guest.

```
VBoxManage startvm --type headless 'PMM Server'
```

2. (Optional) Watch the log file.

```
tail -f /tmp/pmm-server-console.log
```

3. Wait for one minute for the server to boot up.
4. Choose one of:
   - Read the IP address from the tailed log file.
   - Extract the IP address from the log file.

```
grep -e "^IP:" /tmp/pmm-server-console.log | cut -f2 -d' '
```

5. (Optional) Stop the guest:

```
VBoxManage controlvm "PMM Server" poweroff
```

Log into user interface

UI

1. Open a web browser and visit the guest IP address.
2. The PMM login screen appears.
3. Enter the default username and password in the relevant fields and click *Log in*.
   - username: `admin`
   - password: `admin`
4. (Recommended) Follow the prompts to change the default password.

Percona LLC and/or its affiliates ©2024

You also can change the default password through SSH by using the `change-admin-password` command.

1. The PMM Home Dashboard appears.

(Optional) Change root password

UI

1. Start the virtual machine in GUI mode.
2. Log in with the default superuser credentials:
   - Username: `root`
   - Password: `percona`
3. Follow the prompts to change the password.

(Optional) Set up SSH

UI/CLI

1. Create a key pair for the `admin` user.

   ```
   ssh-keygen -f admin
   ```

2. Log into the PMM user interface.
3. Select *PMM → PMM Settings → SSH Key*.
4. Copy and paste the contents of the `admin.pub` file into the *SSH Key* field.
5. Click *Apply SSH Key*. (This copies the public key to `/home/admin/.ssh/authorized_keys` in the guest).
6. Log in via SSH (`N.N.N.N` is the guest IP address).

   ```
   ssh -i admin admin@N.N.N.N
   ```

(Optional) Set up static IP

When the guest OS starts, it will get an IP address from the hypervisor's DHCP server. This IP can change each time the guest OS is restarted. Setting a static IP for the guest OS avoids having to check the IP address whenever the guest is restarted.

CLI

1. Start the virtual machine in non-headless (GUI) mode.
2. Log in as `root`.
3. Edit `/etc/sysconfig/network-scripts/ifcfg-eth0`
4. Change the value of `BOOTPROTO`:

   ```
   BOOTPROTO=none
   ```

5. Add these values:

   ```
   IPADDR=192.168.1.123 # replace with the desired static IP address
   NETMASK=255.255.255.0 # replace with the netmask for your IP address
   GATEWAY=192.168.1.1 # replace with the network gateway for your IP address
   ```


Percona LLC and/or its affiliates ©2024

```
PEERDNS=no
DNS1=192.168.1.53 # replace with your DNS server IP
```

6. Restart the interface.

```
ifdown eth0 && ifup eth0
```

7. Check the IP.

```
ip addr show eth0
```

8. Preserve the network configuration across reboots.

```
echo "network: {config: disabled}" > /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg
```

Remove

UI

1. Stop the virtual machine: select *Close* → *Power Off*.
2. Remove the virtual machine: select *Remove* → *Delete all files*.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

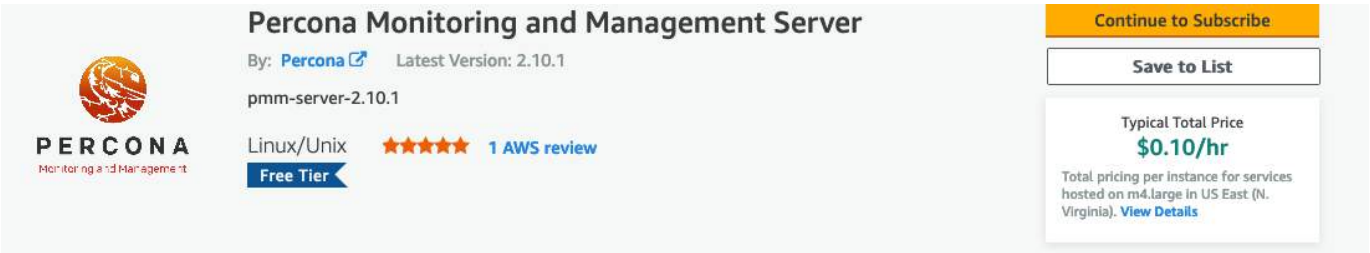
Community Forum Get a Percona Expert

2023-02-23

3.2.7 AWS Marketplace

You can run an instance of PMM Server hosted at AWS Marketplace.

Assuming that you have an AWS (Amazon Web Services) account, locate *Percona Monitoring and Management Server* in AWS Marketplace or use this link.



Selecting a region and instance type in the *Pricing Information* section will give you an estimate of the costs involved. This is only an indication of costs. You will choose regions and instance types in later steps.

Percona Monitoring and Management Server is provided at no cost, but you may need to pay for infrastructure costs.

Percona LLC and/or its affiliates ©2024

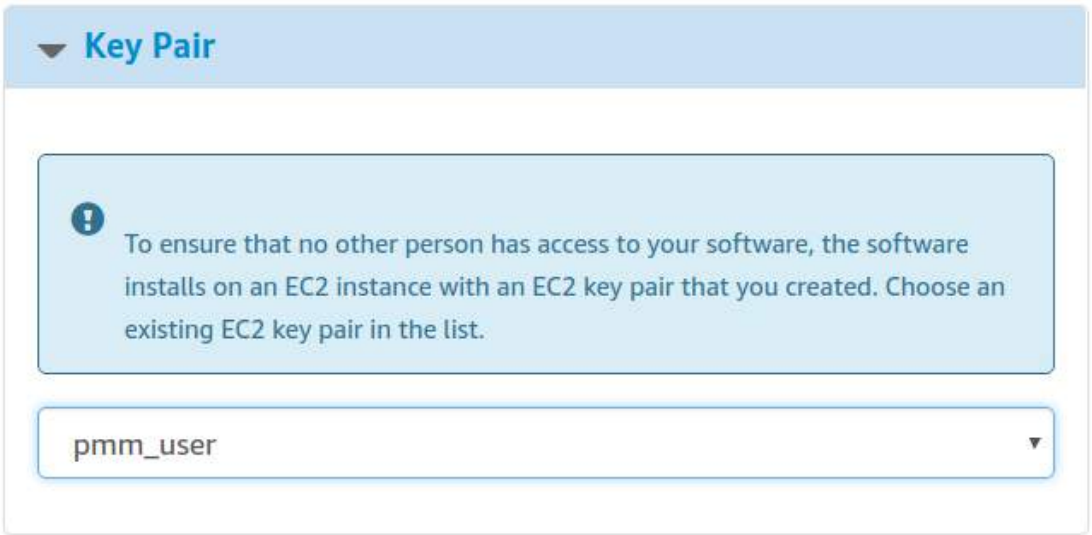
Disk space consumed by PMM Server depends on the number of hosts being monitored. Although each environment will be unique, you can consider the data consumption figures for the PMM Demo web site which consumes approximately 230 MB per host per day, or approximately 6.9 GB per host at the default 30 day retention period.

For more information, see our blog post How much disk space should I allocate for Percona Monitoring and Management?.

1. Click *Continue to Subscribe*.
2. **Subscribe to this software**: Check the terms and conditions and click *Continue to Configuration*.
3. **Configure this software**:
   a. Select a value for *Software Version*. (The latest is 2.41.2.)
   b. Select a region. (You can change this in the next step.)
   c. Click *Continue to Launch*.
4. **Launch this software**:
   a. **Choose Action**: Select a launch option. *Launch from Website* is a quick way to make your instance ready. For more control, choose *Launch through EC2*.
   b. **EC2 Instance Type**: Select an instance type.
   c. **VPC Settings**: Choose or create a VPC (virtual private cloud).
   d. **Subnet Settings**: Choose or create a subnet.
   e. **Security Group Settings**: Choose a security group or click *Create New Based On Seller Settings
   f. **Key Pair Settings**: Choose or create a key pair.
   g. Click *Launch*.

Limiting Access to the instance: security group and a key pair

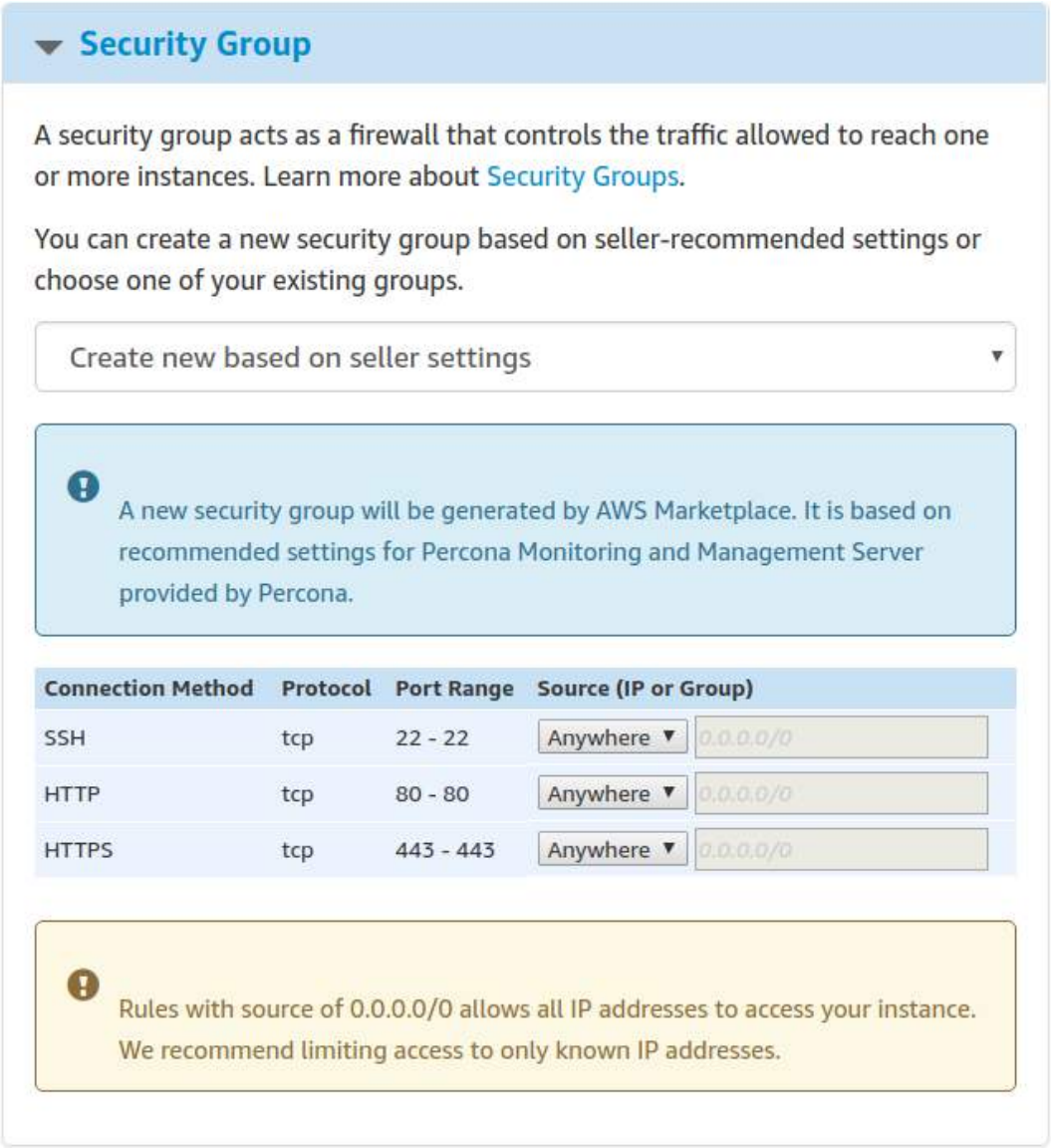
In the *Security Group* section, which acts like a firewall, you may use the preselected option `Create new based on seller settings` to create a security group with recommended settings. In the *Key Pair* select an already set up EC2 key pair to limit access to your instance.



Percona LLC and/or its affiliates ©2024

Important

The security group should allow communication via the the following ports: *22*, *80*, and *443*. PMM should also be able to access port *3306* on the RDS that uses the instance.

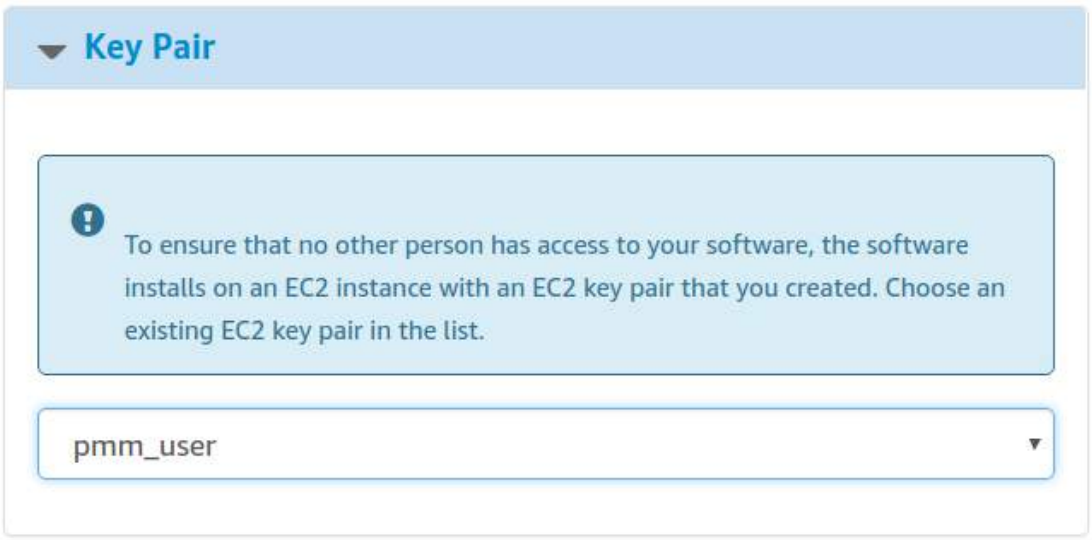


Applying settings

Scroll up to the top of the page to view your settings. Then, click the *Launch with 1 click* button to continue and adjust your settings in the EC2 console.

Your instance settings are summarized in a special area. Click the Launch with 1 click button to continue.

Percona LLC and/or its affiliates ©2024

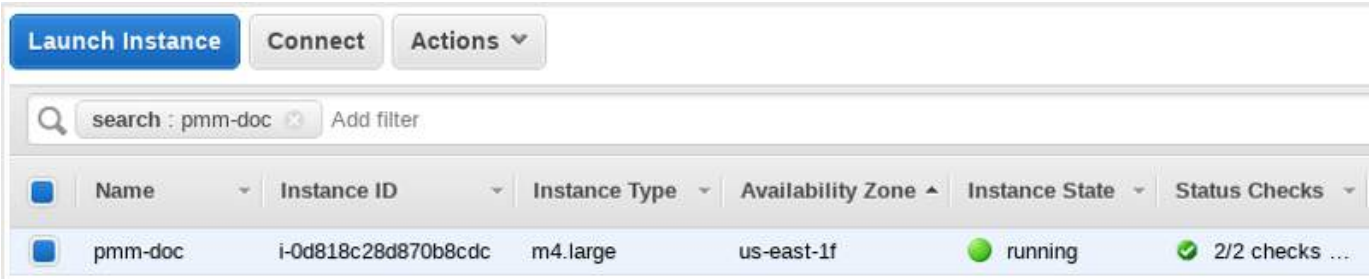


The *Launch with 1 click* button may alternatively be titled as *Accept Software Terms & Launch with 1-Click*.

Adjusting instance settings in the EC2 Console

Your clicking the *Launch with 1 click* button, deploys your instance. To continue setting up your instance, run the EC2 console. It is available as a link at the top of the page that opens after you click the *Launch with 1 click* button.

Your instance appears in the EC2 console in a table that lists all instances available to you. When a new instance is only created, it has no name. Make sure that you give it a name to distinguish it from other instances managed via the EC2 console.



Running the instance

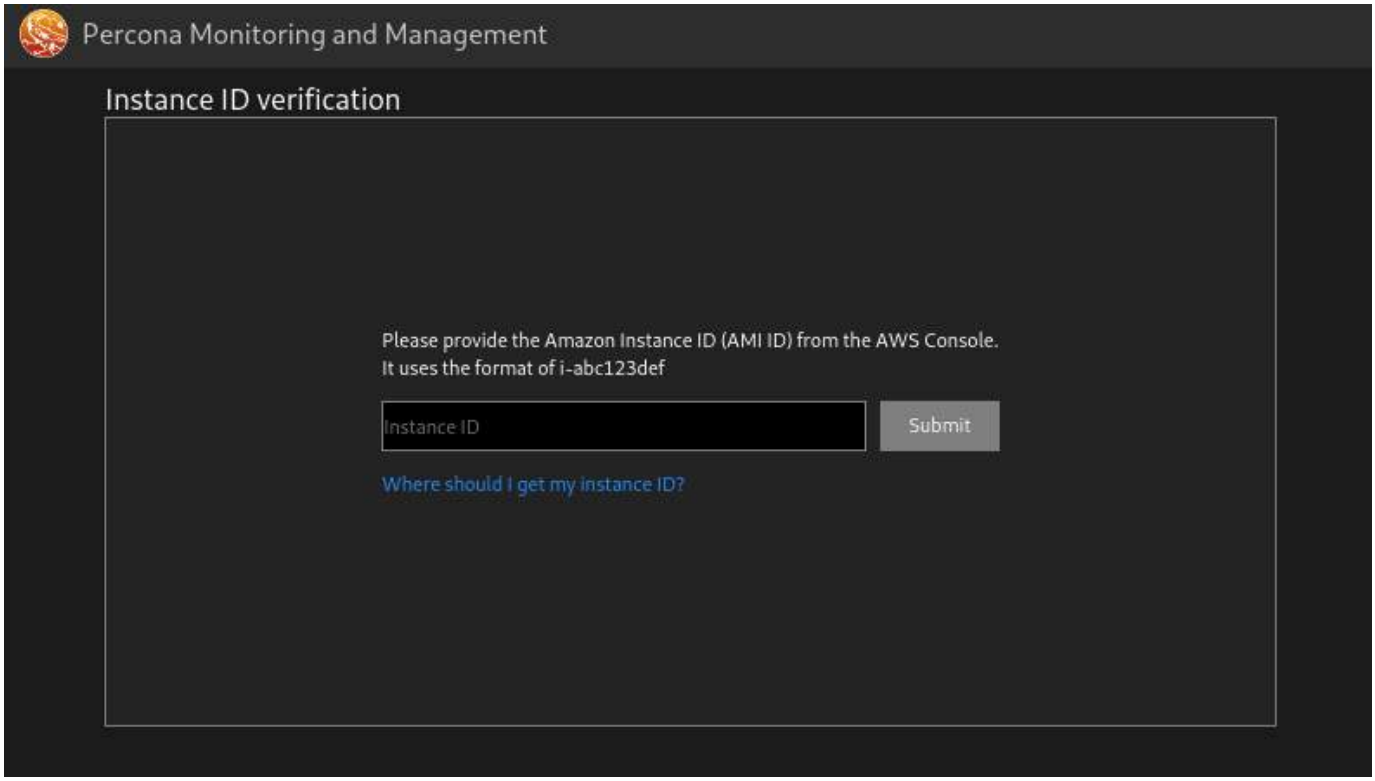
After you add your new instance ,it will take some time to initialize it. When the AWS console reports that the instance is now in a running state, you may continue with configuration of PMM Server.

When started the next time after rebooting, your instance may acquire another IP address. You may choose to set up an elastic IP to avoid this problem.

With your instance selected, open its IP address in a web browser. The IP address appears in the *IPv4 Public IP* column or as value of the *Public IP* field at the top of the *Properties* panel.

Percona LLC and/or its affiliates ©2024

To run the instance, copy and paste its public IP address into the location bar of your browser. In the *Percona Monitoring and Management* welcome page that opens, enter the instance ID.



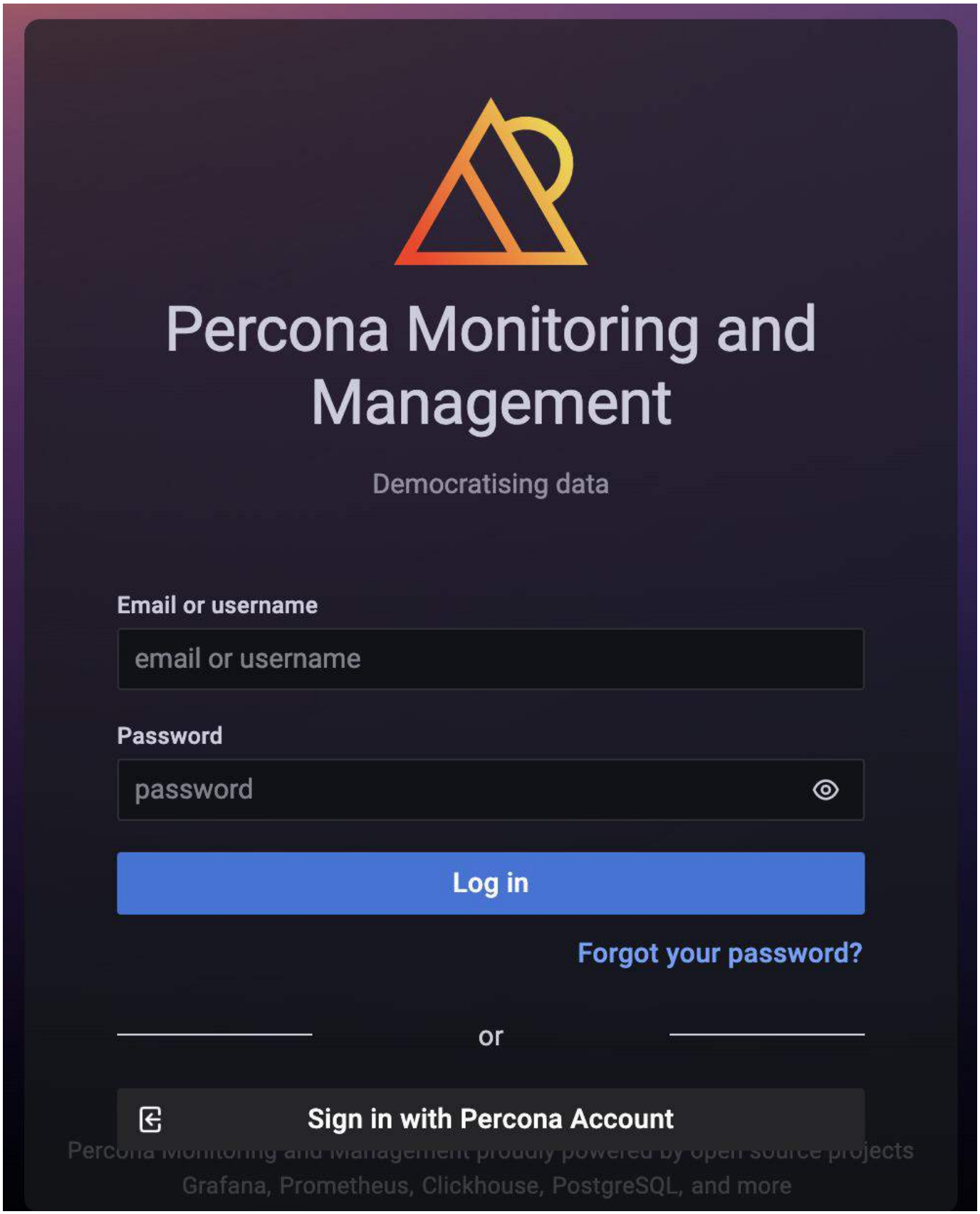
You can copy the instance ID from the *Properties* panel of your instance, select the *Description* tab back in the EC2 console. Click the *Copy* button next to the *Instance ID* field. This button appears as soon as you hover the cursor of your mouse over the ID.

Hover the cursor over the instance ID for the Copy button to appear.

Paste the instance in the *Instance ID* field of the *Percona Monitoring and Management* welcome page and click *Submit*.

PMM Server provides user access control, and therefore you will need user credentials to access it:

Percona LLC and/or its affiliates ©2024

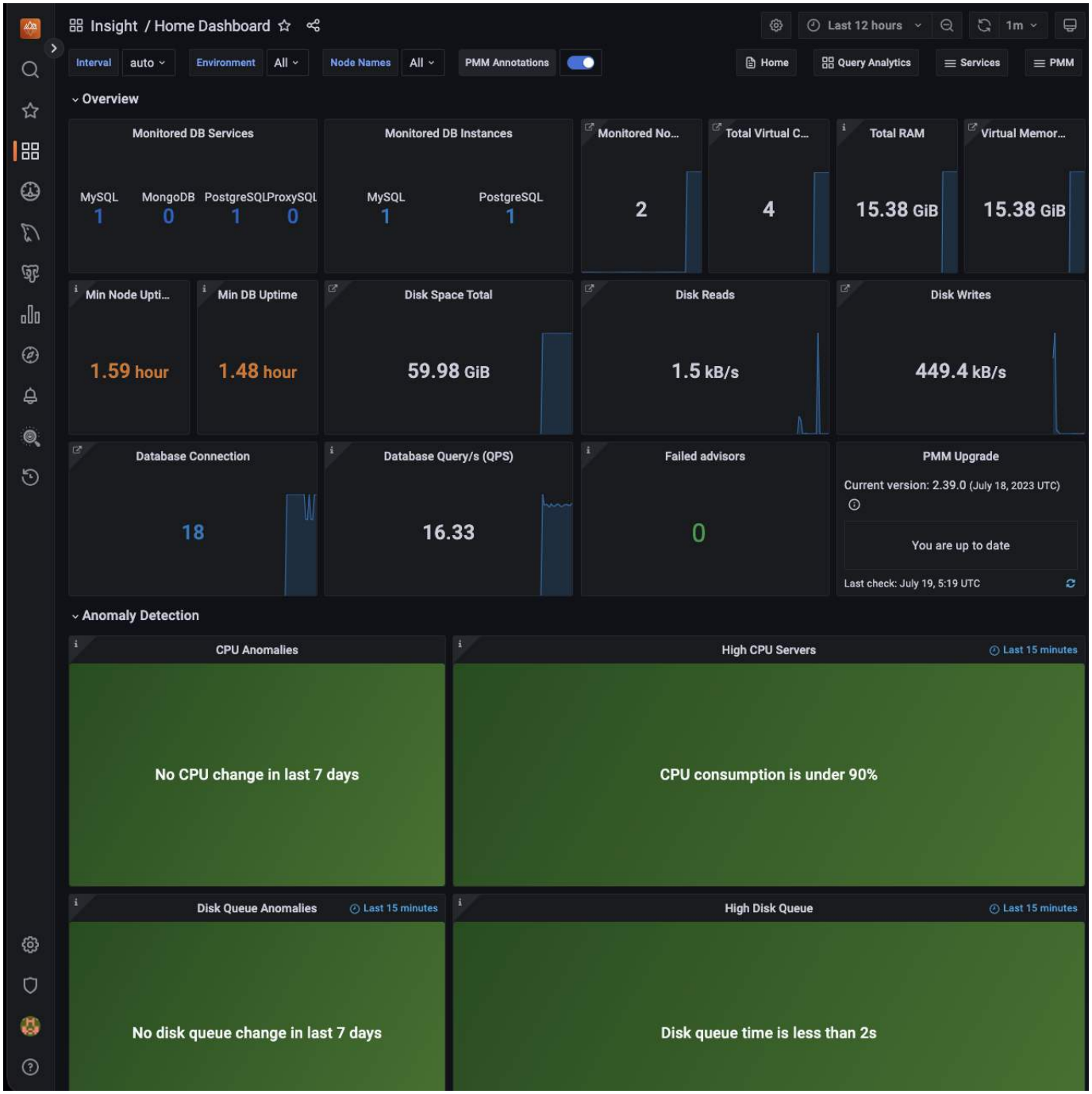


- Default user name: `admin`
- Default password: `admin`

You will be prompted to change the default password every time you log in.

Percona LLC and/or its affiliates ©2024

The PMM Server is now ready and the home page opens.



You are creating a username and password that will be used for two purposes:

1. authentication as a user to PMM - the credentials to log in to PMM.
2. authentication between PMM Server and PMM Clients - you will re-use these credentials on another host when configuring PMM Client for the first time on a server, for example (DO NOT RUN ON THIS PMM SERVER YOU JUST CREATED):

```
pmm-admin config --server-insecure-tls --server-url=https://admin:admin@<IP Address>:443
```

For instructions about how to access your instances by using an SSH client, see Connecting to Your Linux Instance Using SSH

Percona LLC and/or its affiliates ©2024

Make sure to replace the user name `ec2-user` used in this document with `admin`.

Resizing the EBS Volume

Your AWS instance comes with a predefined size which can become a limitation. To make more disk space available to your instance, you need to increase the size of the EBS volume as needed and then your instance will reconfigure itself to use the new size.

The procedure of resizing EBS volumes is described in the Amazon documentation: Modifying the Size, IOPS, or Type of an EBS Volume on Linux.

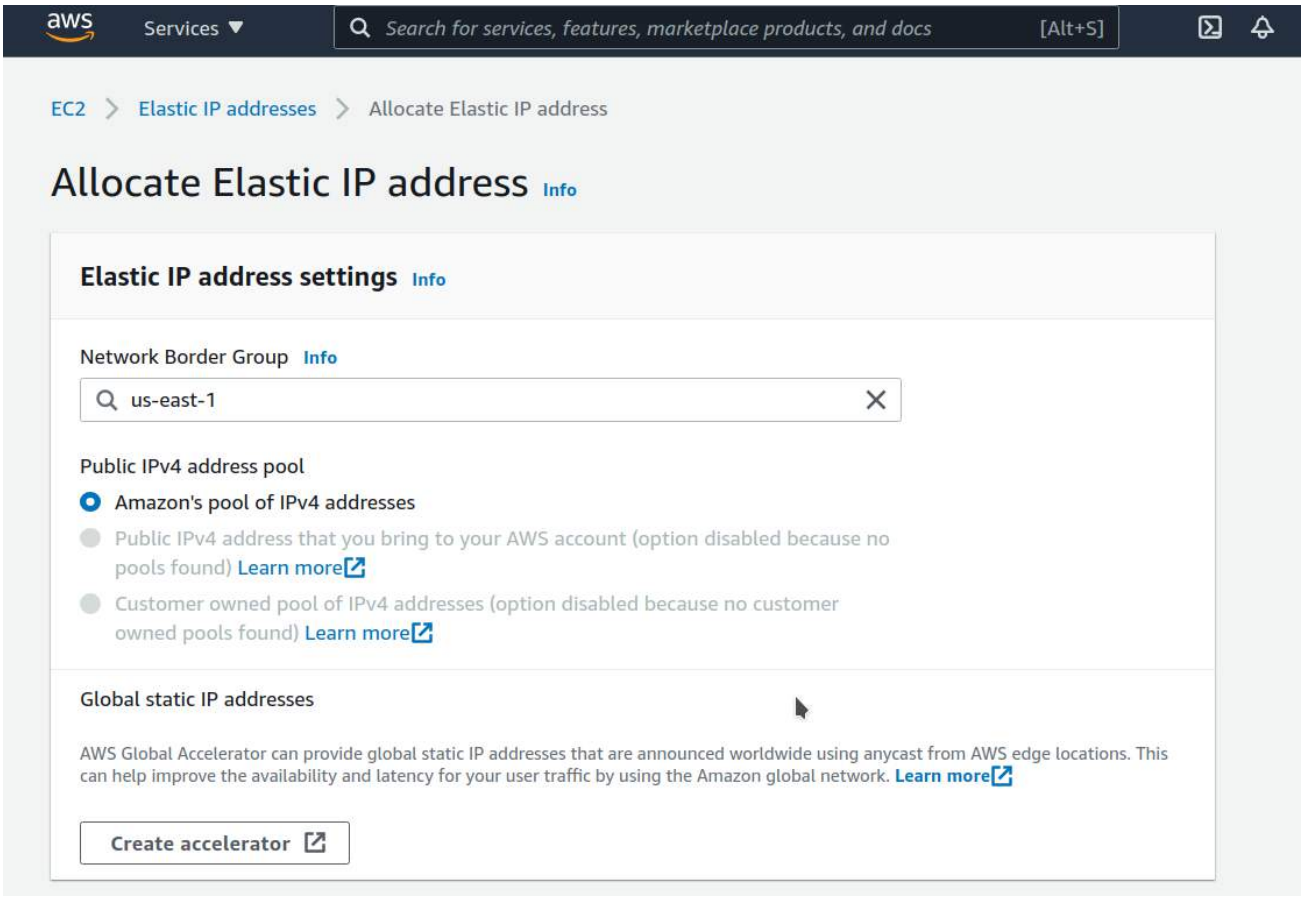
After the EBS volume is updated, PMM Server instance will auto-detect changes in approximately 5 minutes or less and will reconfigure itself for the updated conditions.

Upgrading PMM Server on AWS

CHANGE PUBLIC IP ADDRESS

To assign a public IP address for an Amazon EC2 instance, follow these steps:

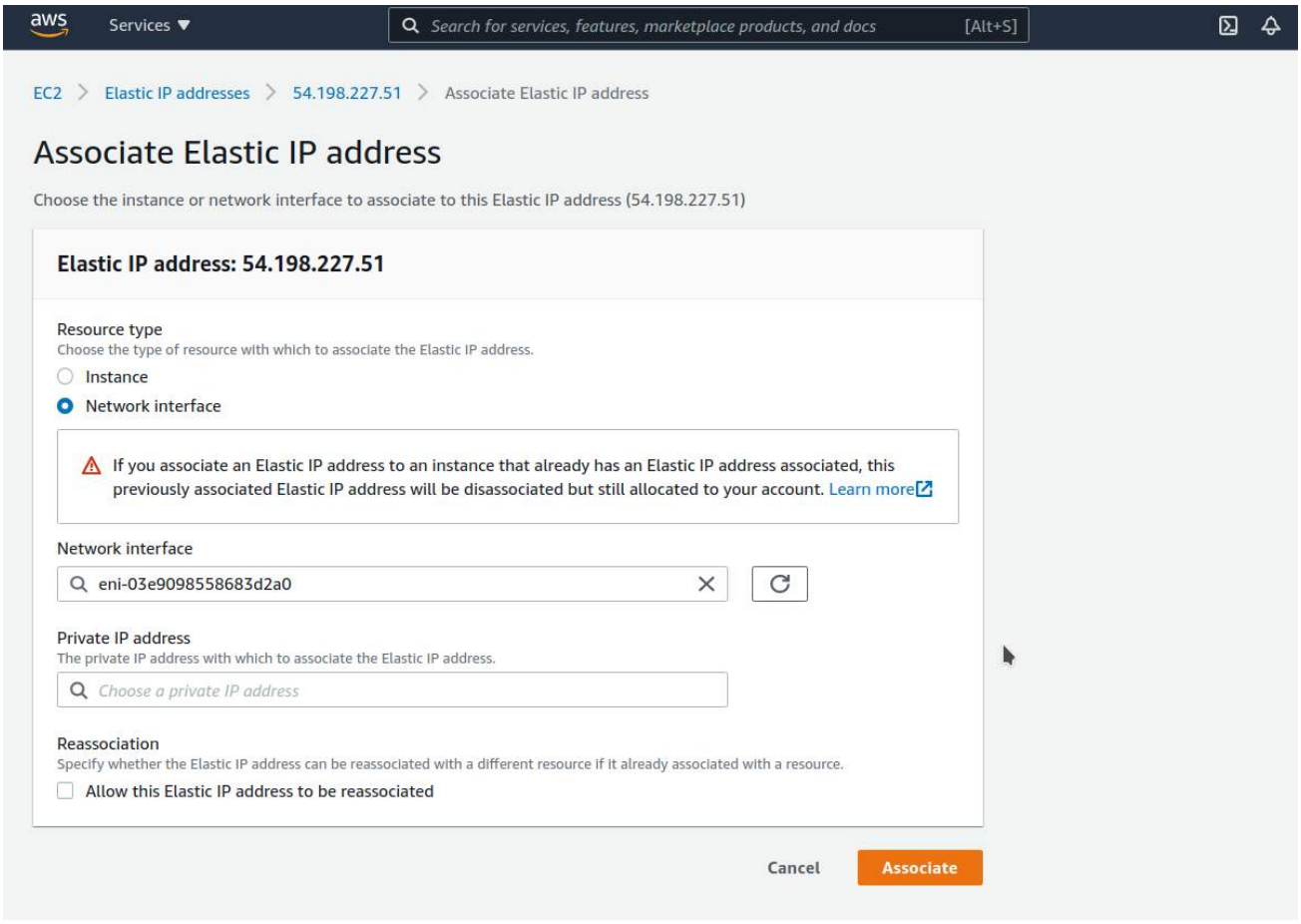
1. Allocate Elastic IP address



2. Associate Elastic IP address with a Network interface ID of your EC2 instance

   If you associate an Elastic IP address to an instance that already has an Elastic IP address associated, this previously associated Elastic IP address will be disassociated but still allocated to your account.

Percona LLC and/or its affiliates ©2024

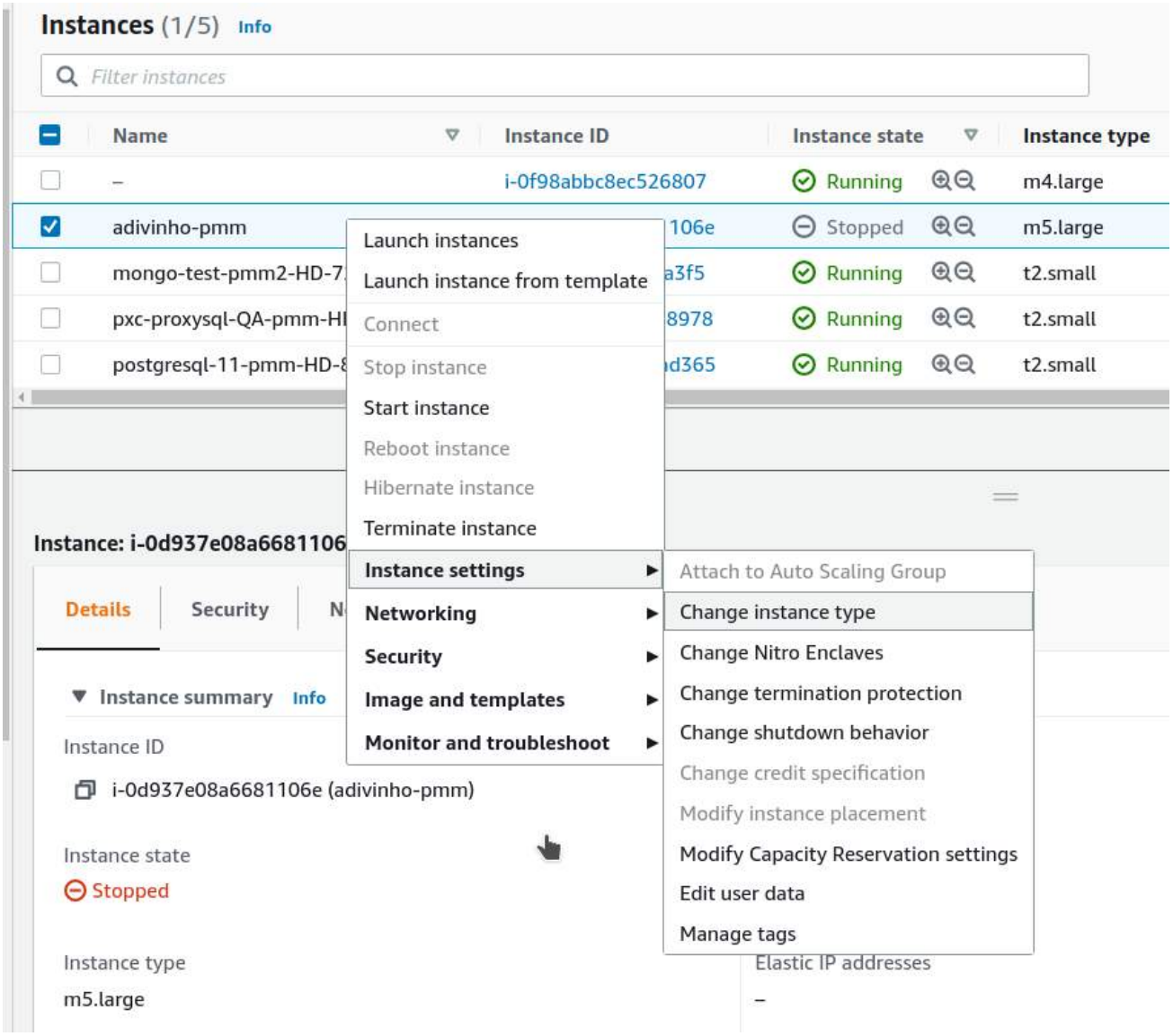


UPGRADING EC2 INSTANCE CLASS

Upgrading to a larger EC2 instance class is supported by PMM provided you follow the instructions from the AWS manual. The PMM AMI image uses a distinct EBS volume for the PMM data volume which permits independent resizing of the EC2 instance without impacting the EBS volume.

1. Open the Amazon EC2 console.
2. In the navigation pane, choose PMM Server Instances.
3. Select the instance and choose Actions, Instance state, Stop instance.
4. In the Change instance type dialog box, select the instance type that you want.

Percona LLC and/or its affiliates ©2024



5. Choose Apply to accept the new settings and start the stopped instance.

EXPANDING THE PMM DATA EBS VOLUME

The PMM data volume is mounted as an XFS formatted volume on top of an LVM volume. There are two ways to increase this volume size:

1. Add a new disk via EC2 console or API, and expand the LVM volume to include the new disk volume.
2. Expand existing EBS volume and grow the LVM volume.

EXPAND EXISTING EBS VOLUME

To expand the existing EBS volume for increased capacity, follow these steps.

1. Expand the disk from AWS Console/CLI to the desired capacity.
2. Login to the PMM EC2 instance and verify that the disk capacity has increased. For example, if you have expanded your disk from 16G to 32G, `dmesg` output should look like below:

```
[  535.994494] xvdb: detected capacity change from 17179869184 to 34359738368
```


Percona LLC and/or its affiliates ©2024

3. You can check information about volume groups and logical volumes with the `vgs` and `lvs` commands:

```
vgs
```

```
VG     #PV #LV #SN Attr   VSize  VFree
DataVG   1   2   0 wz--n- <16.00g    0
```

```
lvs
```

```
LV       VG     Attr       LSize   Pool Origin Data%  Meta% Move Log Cpy%Sync Convert
DataLV   DataVG Vwi-aotz-- <12.80g ThinPool        1.74
ThinPool DataVG twi-aotz--  15.96g 1.39  1.29
```

4. Now we can use the `lsblk` command to see that our disk size has been identified by the kernel correctly, but LVM2 is not yet aware of the new size. We can use `pvresize` to make sure the PV device reflects the new size. Once `pvresize` is executed, we can see that the VG has the new free space available.

```
lsblk | grep xvdb
```

```
xvdb                    202:16 0 32G 0 disk
```

```
pvscan
```

```
PV /dev/xvdb   VG DataVG    lvm2 [<16.00 GiB / 0    free]
Total: 1 [<16.00 GiB] / in use: 1 [<16.00 GiB] / in no VG: 0 [0   ]
```

```
pvresize /dev/xvdb
```

```
Physical volume "/dev/xvdb" changed
1 physical volume(s) resized / 0 physical volume(s) not resized
```

```
pvs
```

```
PV         VG     Fmt  Attr PSize   PFree
/dev/xvdb  DataVG lvm2 a--  <32.00g 16.00g
```

5. We then extend our logical volume. Since the PMM image uses thin provisioning, we need to extend both the pool and the volume:

```
lvs
```

```
LV       VG     Attr       LSize   Pool    Origin Data%  Meta% Move Log Cpy%Sync Convert
DataLV   DataVG Vwi-aotz-- <12.80g ThinPool        1.77
ThinPool DataVG twi-aotz--  15.96g                 1.42   1.32
```

```
lvextend /dev/mapper/DataVG-ThinPool -l 100%VG
```

```
Size of logical volume DataVG/ThinPool_tdata changed from 16.00 GiB (4096 extents) to 31.96 GiB (8183 extents).
Logical volume DataVG/ThinPool_tdata successfully resized.
```

```
lvs
```

```
LV       VG     Attr       LSize   Pool    Origin Data%  Meta% Move Log Cpy%Sync Convert
DataLV   DataVG Vwi-aotz-- <12.80g ThinPool        1.77
ThinPool DataVG twi-aotz--  31.96g                 0.71   1.71
```

6. Once the pool and volumes have been extended, we need to now extend the thin volume to consume the newly available space. In this example we've grown available space to almost 32GB, and already consumed 12GB, so we're extending an additional 19GB:

Percona LLC and/or its affiliates ©2024

```
lvs
```

```
LV       VG     Attr       LSize   Pool    Origin Data%  Meta% Move Log Cpy%Sync Convert
DataLV   DataVG Vwi-aotz-- <12.80g ThinPool        1.77
ThinPool DataVG twi-aotz--  31.96g                 0.71   1.71
```

```
lvextend /dev/mapper/DataVG-DataLV -L +19G
```

```
Size of logical volume DataVG/DataLV changed from <12.80 GiB (3276 extents) to <31.80 GiB (8140 extents).
Logical volume DataVG/DataLV successfully resized.
```

```
lvs
```

```
LV       VG     Attr       LSize   Pool    Origin Data%  Meta% Move Log Cpy%Sync Convert
DataLV   DataVG Vwi-aotz-- <31.80g ThinPool        0.71
ThinPool DataVG twi-aotz--  31.96g                 0.71   1.71
```

7. We then expand the XFS file system to reflect the new size using `xfs_growfs`, and confirm the file system is accurate using the `df` command.

```
df -h /srv
```

```
Filesystem                Size Used Avail Use% Mounted on
/dev/mapper/DataVG-DataLV   13G 249M   13G   2% /srv
```

```
xfs_growfs /srv
```

```
meta-data=/dev/mapper/DataVG-DataLV isize=512    agcount=103, agsize=32752 blks
         =                          sectsz=512   attr=2, projid32bit=1
         =                          crc=1        finobt=0 spinodes=0
data     =                          bsize=4096   blocks=3354624, imaxpct=25
         =                          sunit=16     swidth=16 blks
naming   =version 2                 bsize=4096   ascii-ci=0 ftype=1
log      =internal                  bsize=4096   blocks=768, version=2
         =                          sectsz=512   sunit=16 blks, lazy-count=1
realtime =none                      extsz=4096   blocks=0, rtextents=0
data blocks changed from 3354624 to 8335360
```

```
df -h /srv
```

```
Filesystem               Size Used Avail Use% Mounted on
/dev/mapper/DataVG-DataLV   32G 254M   32G   1% /srv
```

EXPAND THE AMAZON EBS ROOT VOLUME

1. Expand the disk from AWS Console/CLI to the desired capacity.
2. Login to the PMM EC2 instance and verify that the disk capacity has increased. For example, if you have expanded disk from 8G to 10G, `dmesg` output should look like below:

```
# dmesg | grep "capacity change"
[63175.044762] nvme0n1: detected capacity change from 8589934592 to 10737418240
```

3. Use the `lsblk` command to see that our disk size has been identified by the kernel correctly, but LVM2 is not yet aware of the new size.

```
# lsblk
NAME                    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
nvme0n1                 259:1    0    10G  0 disk
└─nvme0n1p1               259:2    0     8G  0 part /
...
```


Percona LLC and/or its affiliates ©2024

4. For volumes that have a partition, such as the root volume shown in the previous step, use the `growpart` command to extend the partition.

```
# growpart /dev/nvme0n1 1
CHANGED: partition=1 start=2048 old: size=16775168 end=16777216 new: size=20969439 end=20971487
```

5. To verify that the partition reflects the increased volume size, use the `lsblk` command again.

```
# lsblk
NAME                    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
nvme0n1                 259:1    0    10G  0 disk
└─nvme0n1p1               259:2    0    10G  0 part /
...
```

6. Extend the XFS file system on the root volume by `xfs_growfs` command. I

```
# xfs_growfs -d /
meta-data=/dev/nvme0n1p1         isize=512    agcount=4, agsize=524224 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0 spinodes=0
data     =                       bsize=4096   blocks=2096896, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=1
log      =internal               bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 2096896 to 2621120
```

7. Verify that file system reflects the increased volume size

```
# df -hT /
Filesystem     Type  Size  Used Avail Use% Mounted on
/dev/nvme0n1p1 xfs    10G  5,6G  4,5G  56% /
```

Backup PMM Server

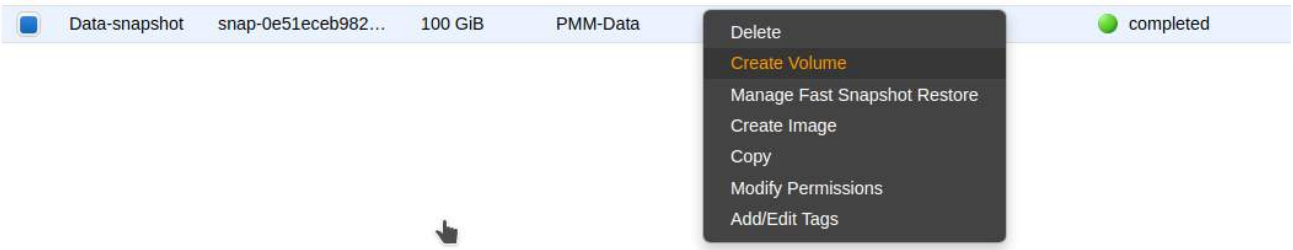
All data are stored in the `/srv` partition, so it's enough to back the PMM data volume. You can create a point-in-time snapshot of the volume and use it for data backup.

The procedure of creating a snapshot is described in the Amazon documentation: Create Amazon EBS snapshots

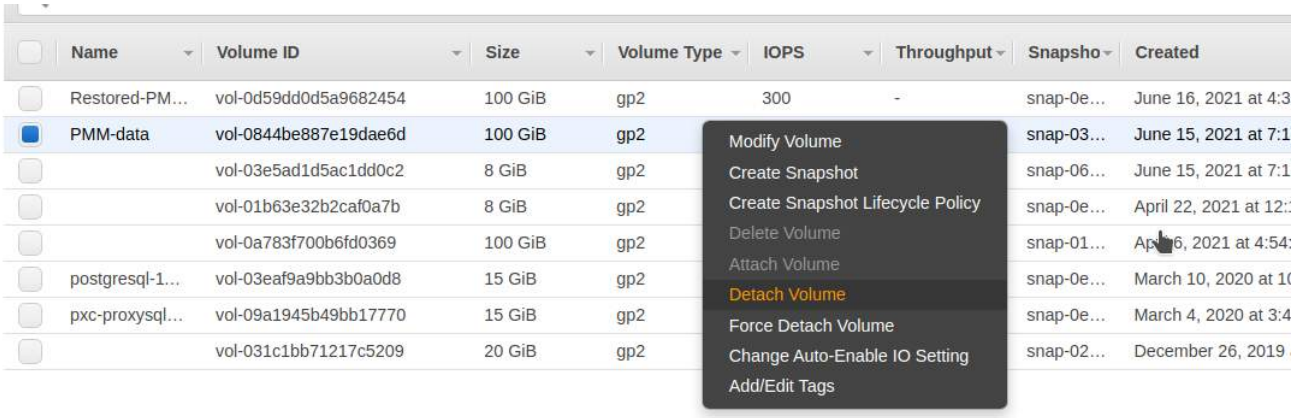
Restore PMM Server from a backup

1. Create a new volume by using the latest snapshot of the PMM data volume.

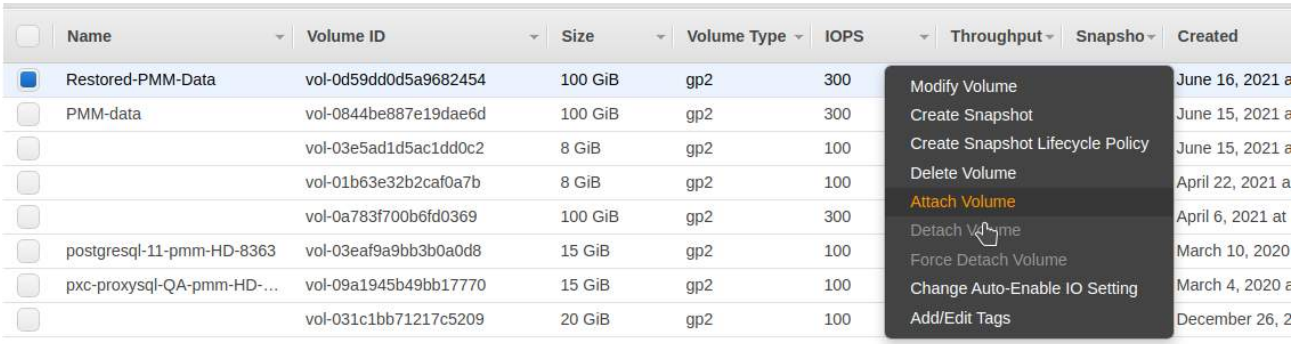
Percona LLC and/or its affiliates ©2024



2. Stop the PMM Server instance.
3. Detach the current PMM data volume.



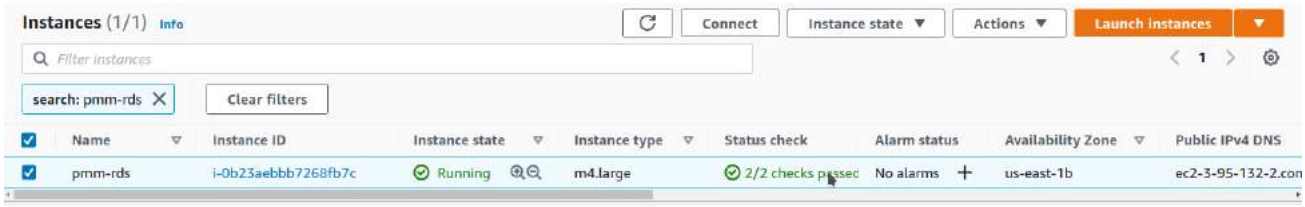
4. Attach the new volume.



5. Start the PMM Server instance.

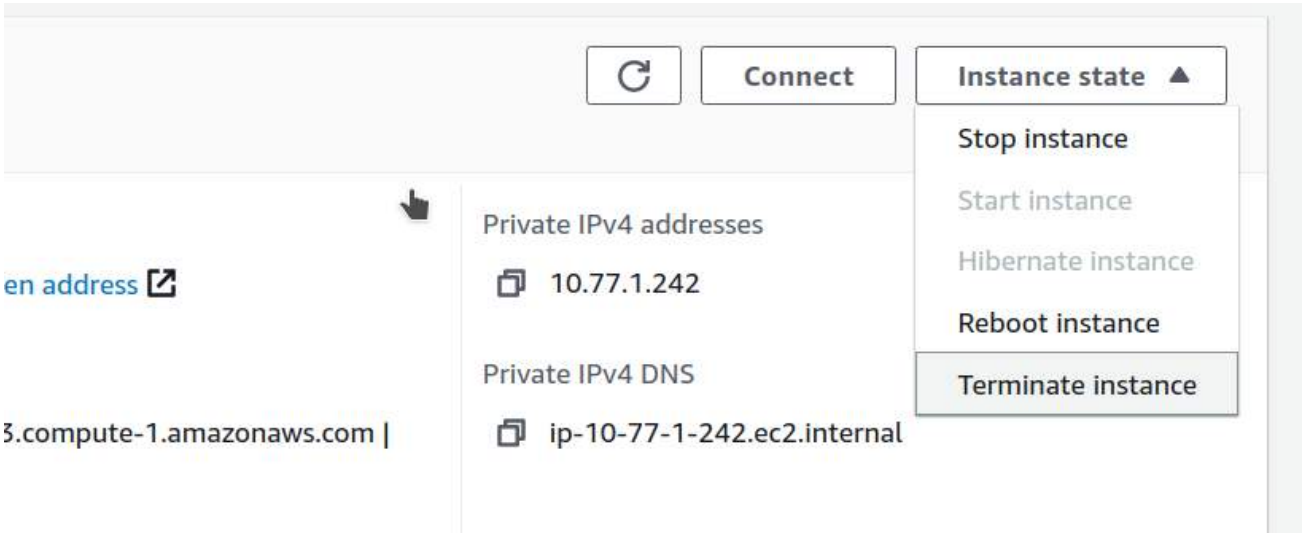
Remove PMM Server

1. Find the instance in the EC2 Console

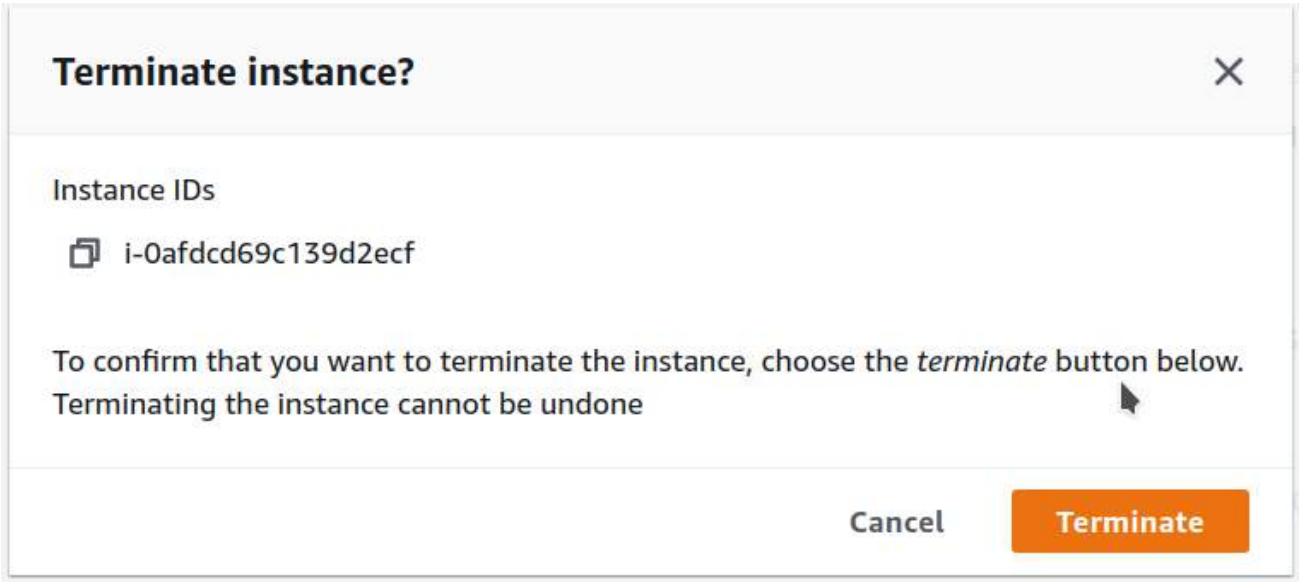


2. Select “Instance state” menu and “Terminate instance”

Percona LLC and/or its affiliates ©2024



3. Confirm termination operation



See also

Improving Percona Monitoring and Management EC2 Instance Resilience Using CloudWatch Alarm Actions

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-02-23

Percona LLC and/or its affiliates ©2024

3.2.8 Easy-install script

> **Caution**
>
> You can download and check `get-pmm.sh` before running it from our github:

Linux or macOS

Using `curl`:

```
curl -fsSL https://www.percona.com/get/pmm | /bin/bash
```

Using `wget`:

```
wget -O - https://www.percona.com/get/pmm | /bin/bash
```

This script:

- Installs Docker if it is not already installed on your system.
- Stops and renames any currently running PMM Server Docker container from `pmm-server` to `pmm-server-{timestamp}`. This old pmm-server container is not a recoverable backup.
- Pulls and runs the latest PMM Server Docker image.
- Can run in Interactive mode to change the default settings:

```
curl -fsSLO https://www.percona.com/get/pmm (or wget https://www.percona.com/get/pmm)
chmod +x pmm
./pmm --interactive
```

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-06-28

3.3 Client

3.3.1 Set up PMM Client

There are different ways to install PMM Client on a node and register it with PMM Server. Choose from:

- Docker: Run PMM Client as a Docker container.
- Package manager:
  - On Debian or Red Hat Linux, install `percona-release` and use a Linux package manager (`apt`/`dnf`) to install PMM Client.
  - On Debian or Red Hat, download `.deb`/`.rpm` PMM Client packages and manually install them.

Percona LLC and/or its affiliates ©2024

Binary is only way to install PMM client without root permissions

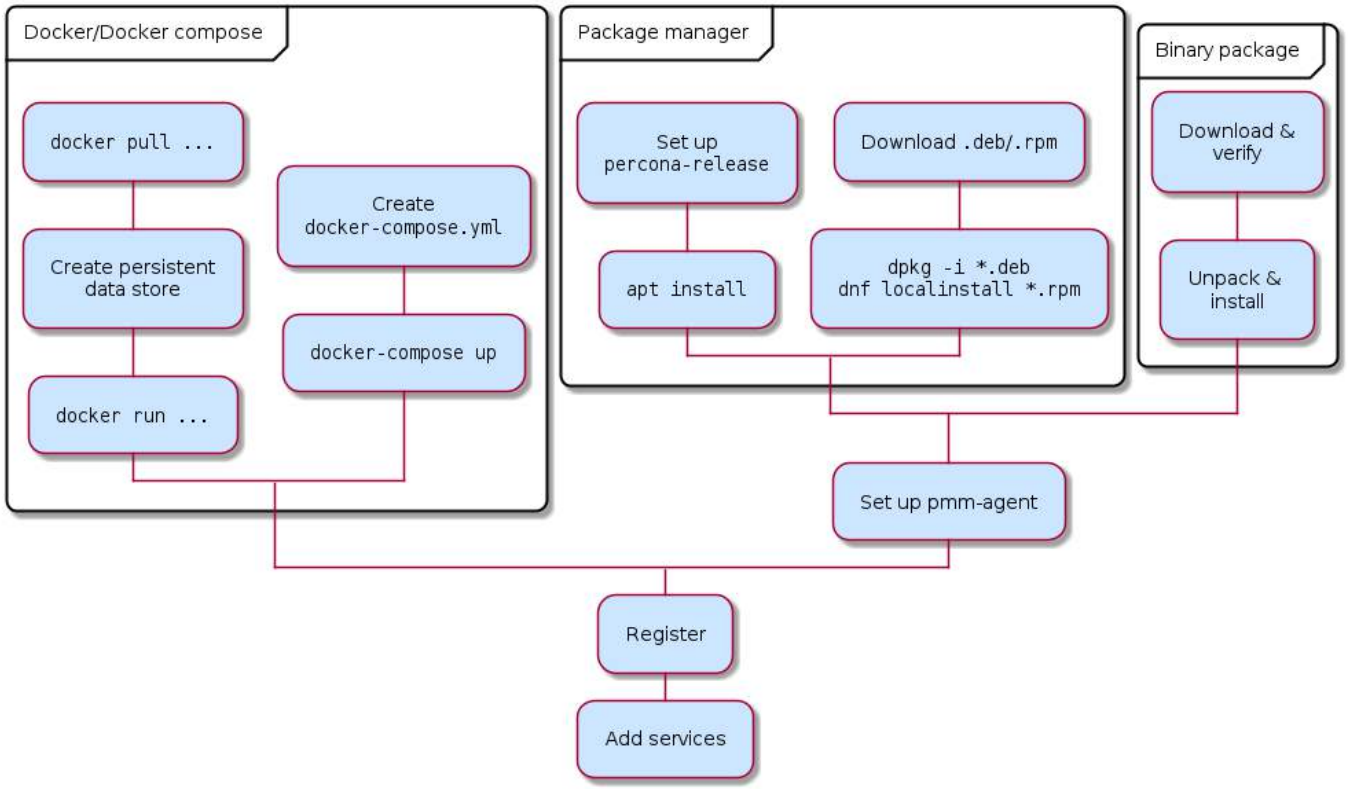
- Binary package: For other Linux distributions, download and unpack generic PMM Client Linux binaries.

When you have installed PMM Client, you must:

- Register the node with PMM Server.
- Configure and add services according to type.

If you need to, you can unregister, remove services or remove PMM Client.

Here's an overview of the choices.



Before you start

- Set up PMM Server with a known IP address accessible from the client node.
- You have superuser (root) access on the client host.
- You have superuser access to any database servers that you want to monitor.
- These Linux packages are installed: `curl`, `gnupg`, `sudo`, `wget`.
- If using it, install Docker.

Percona LLC and/or its affiliates ©2024

- System requirements:
  - Operating system – PMM Client runs on any modern 64-bit Linux distribution. It is tested on supported versions of Debian, Ubuntu, CentOS, and Red Hat Enterprise Linux. (See Percona software support life cycle).
  - Disk – A minimum of 100 MB of storage is required for installing the PMM Client package.

  With a good connection to PMM Server, additional storage is not required. However, the client needs to store any collected data that it cannot dispatch immediately, so additional storage may be required if the connection is unstable or the throughput is low. VMagent uses 1 GB of disk space for cache during a network outage. QAN, on the other hand, uses RAM to store cache.

Install

DOCKER

The PMM Client Docker image is a convenient way to run PMM Client as a preconfigured Docker container.

1. Pull the PMM Client docker image.

   ```
   docker pull \
   percona/pmm-client:2
   ```

2. Use the image as a template to create a persistent data store that preserves local data when the image is updated.

   ```
   docker create \
   --volume /srv \
   --name pmm-client-data \
   percona/pmm-client:2 /bin/true
   ```

3. Run the container to start PMM Agent in setup mode. Set `X.X.X.X` to the IP address of your PMM Server. (Do not use the `docker --detach` option as PMM agent only logs to the console.)

   ```
   PMM_SERVER=X.X.X.X:443
   docker run \
   --rm \
   --name pmm-client \
   -e PMM_AGENT_SERVER_ADDRESS=${PMM_SERVER} \
   -e PMM_AGENT_SERVER_USERNAME=admin \
   -e PMM_AGENT_SERVER_PASSWORD=admin \
   -e PMM_AGENT_SERVER_INSECURE_TLS=1 \
   -e PMM_AGENT_SETUP=1 \
   -e PMM_AGENT_CONFIG_FILE=config/pmm-agent.yaml \
   --volumes-from pmm-client-data \
   percona/pmm-client:2
   ```

Tips

You can find a complete list of compatible environment variables here.

1. Check status.

   ```
   docker exec pmm-client \
   pmm-admin status
   ```

   In the PMM user interface you will also see an increase in the number of monitored nodes.

You can now add services with `pmm-admin` by prefixing commands with `docker exec pmm-client`.

Percona LLC and/or its affiliates ©2024

Tips

- Adjust host firewall and routing rules to allow Docker communications. (Read more)
- For help: `docker run --rm percona/pmm-client:2 --help`

In the GUI.

- Select *PMM Dashboards* → *System (Node)* → *Node Overview*.
- In the *Node Names* menu, select the new node.
- Change the time range to see data.

Danger

`pmm-agent.yaml` contains sensitive credentials and should not be shared.

PACKAGE MANAGER

Tip

If you have used `percona-release` before, disable and re-enable the repository:

```
percona-release disable all
percona-release enable original release
```

Debian-based

1. Configure repositories.

```
wget https://repo.percona.com/apt/percona-release_latest.generic_all.deb
dpkg -i percona-release_latest.generic_all.deb
```

2. Install the PMM Client package.

Root permissions

```
apt update
apt install -y pmm2-client
```

3. Check.

```
pmm-admin --version
```

4. Register the node.

Red Hat-based

1. Configure repositories.

```
yum install -y https://repo.percona.com/yum/percona-release-latest.noarch.rpm
```

2. Install the PMM Client package.

```
yum install -y pmm2-client
```

3. Check.

Percona LLC and/or its affiliates ©2024

```
pmm-admin --version
```

4. Register the node.

PACKAGE MANAGER – MANUAL DOWNLOAD

1. Visit the Percona Monitoring and Management 2 download page.
2. Under *Version:*, select the one you want (usually the latest).
3. Under *Software:*, select the item matching your software platform.
4. Click to download the package file:
   - For Debian, Ubuntu: `.deb`
   - For Red Hat, CentOS, Oracle Linux: `.rpm`

(Alternatively, copy the link and use `wget` to download it.)

Here are the download page links for each supported platform.

- [Debian 9 (Stretch)]
- Debian 10 (Buster)
- Debian 11 (Bullseye)
- Red Hat/CentOS/Oracle 7
- Red Hat/CentOS/Oracle 8
- [Ubuntu 18.04 (Bionic Beaver)]
- Ubuntu 20.04 (Focal Fossa)
- Ubuntu 22.04 (Jammy Jellyfish)

Debian-based

```
dpkg -i *.deb
```

Red Hat-based

```
dnf localinstall *.rpm
```

BINARY PACKAGE

1. Download the PMM Client package:

```
wget https://downloads.percona.com/downloads/pmm2/2.41.2/binary/tarball/pmm2-client-2.41.2.tar.gz
```

2. Download the PMM Client package checksum file:

```
wget https://downloads.percona.com/downloads/pmm2/2.41.2/binary/tarball/pmm2-client-2.41.2.tar.gz.sha256sum
```

3. Verify the download.

```
sha256sum -c pmm2-client-2.41.2.tar.gz.sha256sum
```

4. Unpack the package and move into the directory.

```
tar xfz pmm2-client-2.41.2.tar.gz && cd pmm2-client-2.41.2
```

5. Choose one of these two commands (depends on your permissions):

Percona LLC and/or its affiliates ©2024

Without root permissions

```
export PMM_DIR=YOURPATH
```

where YOURPATH replace with you real path, where you have required access.

With root permissions

```
export PMM_DIR=/usr/local/percona/pmm2
```

6. Run the installer.

Root permissions (if you skipped step 5 for non root users)

```
./install_tarball
```

7. Change the path.

```
PATH=$PATH:$PMM_DIR/bin
```

8. Set up the agent (pick the command for you depending on permissions)

Root permissions

```
pmm-agent setup --config-file=/usr/local/percona/pmm2/config/pmm-agent.yaml --server-address=192.168.1.123 --server-insecure-tls --server-username=admin --server-password=admin
```

Non root users

```
pmm-agent setup --config-file=${PMM_DIR}/config/pmm-agent.yaml --server-address=192.168.1.123 --server-insecure-tls --server-username=admin --server-password=admin --paths-tempdir=${PMM_DIR}/tmp --paths-base=${PMM_DIR}
```

9. Run the agent.

```
pmm-agent --config-file=${PMM_DIR}/config/pmm-agent.yaml
```

10. Open a new terminal and check.

```
pmm-admin status
```

!!! hint PMM-Agent can be updated from tarball:

a. Download tar.gz with pmm2-client.

b. Extract it.

c. Run ./install_tarball script with the “-u” flag.

The configuration file will be overwritten if you do not provide the “-u” flag while the pmm-agent is updated.

Register

Register your client node with PMM Server.

Percona LLC and/or its affiliates ©2024

```
pmm-admin config --server-insecure-tls --server-url=https://admin:admin@X.X.X.X:443
```

- `X.X.X.X` is the address of your PMM Server.
- `443` is the default port number.
- `admin/admin` is the default PMM username and password. This is the same account you use to log into the PMM user interface, which you had the option to change when first logging in.

Important

Clients *must* be registered with the PMM Server using a secure channel. If you use http as your server URL, PMM will try to connect via https on port 443. If a TLS connection can't be established you will get an error and you must use https along with the appropriate secure port.

EXAMPLES

Register on PMM Server with IP address `192.168.33.14` using the default `admin/admin` username and password, a node with IP address `192.168.33.23`, type `generic`, and name `mynode`.

```
pmm-admin config --server-insecure-tls --server-url=https://admin:admin@192.168.33.14:443 192.168.33.23 generic mynode
```

Add services

You must configure and adding services according to the service type.

- MySQL (and variants Percona Server for MySQL, Percona XtraDB Cluster, MariaDB)
- MongoDB
- PostgreSQL
- ProxySQL
- Amazon RDS
- Microsoft Azure
- Google Cloud Platform (MySQL and PostgreSQL)
- Linux
- External services
- HAProxy
- Remote instances

Tip

To change the parameters of a previously-added service, remove the service and re-add it with new parameters.

Remove

How to remove (uninstall) PMM Client.

DOCKER

Caution

These steps delete the PMM Client Docker image and client services configuration data.

Percona LLC and/or its affiliates ©2024

1. Stop pmm-client container.

```
docker stop pmm-client
```

2. Remove containers.

```
docker rm pmm-client
```

3. Remove the image.

```
docker rmi $(docker images | grep "percona/pmm-client" | awk {'print $3'})
```

4. Remove the volume.

```
docker volume rm pmm-client-data
```

PACKAGE MANAGER

Debian-based distributions

1. Uninstall the PMM Client package.

```
apt remove -y pmm2-client
```

2. Remove the Percona repository

```
dpkg -r percona-release
```

Red Hat-based distributions

1. Uninstall the PMM Client package.

```
yum remove -y pmm2-client
```

2. Remove the Percona repository

```
yum remove -y percona-release
```

Unregister

How to unregister PMM Client from PMM Server.

```
pmm-admin unregister --force
```

All services monitored by this node will be removed from monitoring.

Remove services

You must specify the service type and service name to remove services from monitoring.

```
pmm-admin remove <service-type> <service-name>
```

`service-type`

One of `mysql`, `mongodb`, `postgresql`, `proxysql`, `haproxy`, `external`.

Percona LLC and/or its affiliates ©2024

See also

- Percona release
- PMM Client architecture

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-03-21

3.3.2 MySQL

How to set up PMM to monitor a MySQL or MySQL-based database instance.

PMM Client collects metrics from MySQL, Percona Server for MySQL, Percona XtraDB Cluster, and MariaDB. (Amazon RDS is also supported and explained in a separate section.)

Summary

- Create PMM account and set permissions.
- Choose a data source:
  - Slow query log, or,
  - Performance Schema.
- Configure:
  - Query response time,
  - Tablestats,
  - User statistics.
- Add service.
- Check the service.

Before you start

Check that:

- PMM Server is installed and running with a known IP address accessible from the client node.
- PMM Client is installed and the node is registered with PMM Server.
- You have superuser (root) access on the client host.

Create a database account for PMM

It is good practice to use a non-superuser account to connect PMM Client to the monitored database instance. This example creates a database user with name `pmm`, password `pass`, and the necessary permissions.

On MySQL 8.0

Percona LLC and/or its affiliates ©2024

```
CREATE USER 'pmm'@'127.0.0.1' IDENTIFIED BY 'pass' WITH MAX_USER_CONNECTIONS 10;
GRANT SELECT, PROCESS, REPLICATION CLIENT, RELOAD, BACKUP_ADMIN ON *.* TO 'pmm'@'127.0.0.1';
```

On MySQL 5.7

```
CREATE USER 'pmm'@'127.0.0.1' IDENTIFIED BY 'pass' WITH MAX_USER_CONNECTIONS 10;
GRANT SELECT, PROCESS, REPLICATION CLIENT, RELOAD ON *.* TO 'pmm'@'127.0.0.1';
```

Choose and configure a source

Decide which source of metrics to use, and configure your database server for it. The choices are Slow query log and Performance Schema.

While you can use both at the same time we recommend using only one–there is some overlap in the data reported, and each incurs a small performance penalty. The choice depends on the version and variant of your MySQL instance, and how much detail you want to see.

Here are the benefits and drawbacks of *Slow query log* and *Performance Schema* metrics sources.

	Benefits	Drawbacks
Slow query log	1. More detail. 2. Lower resource impact (with query sampling feature in Percona Server for MySQL).	1. PMM Client must be on same host as database server or have access to slow query log. 2. Log files grow and must be actively managed.
Performance Schema	1. Faster parsing. 2. Enabled by default on newer versions of MySQL.	1. Less detail.

DATA SOURCE RECOMMENDATIONS

Database server	Versions	Recommended source
MySQL	8.0+	Performance Schema
MariaDB	10.0+	Performance Schema
Percona Server for MySQL	8.0+	Slow query log
Percona XtraDB Cluster	8.0+	Slow query log

SLOW QUERY LOG

This section covers how to configure a MySQL-based database server to use the *slow query log* as a source of metrics.

APPLICABLE VERSIONS

Server	Versions
MySQL	5.1-5.5
MariaDB	10.1.2+
Percona Server for MySQL	5.7.10+, 8.0.12+
Percona XtraDB Cluster	5.6, 5.7, 8.0

The *slow query log* records the details of queries that take more than a certain amount of time to complete. With the database server configured to write this information to a file rather than a table, PMM Client parses the file and sends aggregated data to PMM Server via the Query Analytics part of PMM Agent.

Percona LLC and/or its affiliates ©2024

SETTINGS

Variable	Value	Description
`slow_query_log`	ON	Enables the slow query log.
`log_output`	`'FILE'`	Ensures the log is sent to a file. (This is the default on MariaDB.)
`long_query_time`	0	The slow query threshold in seconds. In heavily-loaded applications, many quick queries can affect performance more than a few slow ones. Setting this value to `0` ensures all queries are captured.
`log_slow_admin_statements`	ON	Includes the logging of slow administrative statements.
`log_slow_slave_statements`	ON	Enables logging for queries that have taken more than `long_query_time` seconds to execute on the replica.

Examples

- Configuration file.

```
slow_query_log=ON
log_output=FILE
long_query_time=0
log_slow_admin_statements=ON
log_slow_slave_statements=ON
```

- Session.

```
SET GLOBAL slow_query_log = 1;
SET GLOBAL log_output = 'FILE';
SET GLOBAL long_query_time = 0;
SET GLOBAL log_slow_admin_statements = 1;
SET GLOBAL log_slow_slave_statements = 1;
```

Slow query log – extended

Some MySQL-based database servers support extended slow query log variables.

Applicable versions

Server	Versions
Percona Server for MySQL	5.7.10+, 8.0.12+
Percona XtraDB Cluster	5.6, 5.7, 8.0
MariaDB	10.0

Settings

Variable	Value	Description
`log_slow_rate_limit`	100	Defines the rate of queries captured by the *slow query log*. A good rule of thumb is 100 queries logged per second. For example, if your Percona Server instance processes 10,000 queries per second, you should set `log_slow_rate_limit` to `100` and capture every 100th query for the *slow query log*. Depending on the amount of traffic, logging could become aggressive and resource consuming. This variable throttles the level of intensity of the data capture without compromising information.
`log_slow_rate_type`	‘query’	Set so that it applies to queries, rather than sessions.
`slow_query_log_always_write_time`	1	Specifies which queries should ignore sampling. With query sampling this ensures that queries with longer execution time will always be captured by the slow query log, avoiding the possibility that infrequent slow queries might not get captured at all.

Percona LLC and/or its affiliates ©2024

Variable	Value	Description
`log_slow_verbosity`	'full'	Ensures that all information about each captured query is stored in the slow query log.
`slow_query_log_use_global_control`	'all'	Configure the slow query log during runtime and apply these settings to existing connections. (By default, slow query log settings apply only to new sessions.)

Examples

- Configuration file (Percona Server for MySQL, Percona XtraDB Cluster).

```
log_slow_rate_limit=100
log_slow_rate_type='query'
slow_query_log_always_write_time=1
log_slow_verbosity='full'
slow_query_log_use_global_control='all'
```

- Configuration file (MariaDB).

```
log_slow_rate_limit=100
```

- Session (Percona Server for MySQL, Percona XtraDB Cluster).

```
SET GLOBAL log_slow_rate_limit = 100;
SET GLOBAL log_slow_rate_type = 'query';
SET GLOBAL slow_query_log_always_write_time = 1;
SET GLOBAL log_slow_verbosity = 'full';
SET GLOBAL slow_query_log_use_global_control = 'all';
```

Slow query log rotation

Slow query log files can grow quickly and must be managed.

When adding a service with the command line use the `pmm-admin` option `--size-slow-logs` to set at what size the slow query log file is rotated. (The size is specified as a number with a suffix. See `pmm-admin add mysql`.)

When the limit is reached, PMM Client will:

- remove the previous `.old` slow log file,
- rename the current file by adding the suffix `.old`,
- execute the MySQL command `FLUSH LOGS`.

Only one `.old` file is kept. Older ones are deleted.

You can manage log rotation yourself, for example, with `logrotate`. If you do, you can disable PMM Client's log rotation by providing a negative value to `--size-slow-logs` option when adding a service with `pmm-admin add`.

PERFORMANCE SCHEMA

This section covers how to configure a MySQL-based database server to use *Performance Schema* as a source of metrics.

Applicable versions

Server	Versions
Percona Server for MySQL	5.6, 5.7, 8.0
Percona XtraDB Cluster	5.6, 5.7, 8.0
MariaDB	10.3+

Percona LLC and/or its affiliates ©2024

PMM's *MySQL Performance Schema Details* dashboard charts the various `performance_schema` metrics.

To use *Performance Schema*, set the variables below.

Important

Make sure to restart pmm-agent after making any changes to MySQL perfschema.

Variable	Value	Description
`performance_schema`	`ON`	Enables *Performance Schema* metrics. This is the default in MySQL 5.6.6 and higher.
`performance-schema-instrument`	`'statement/%=ON'`	Configures Performance Schema instruments.
`performance-schema-consumer-statements-digest`	`ON`	Configures the `statements-digest` consumer.
`innodb_monitor_enable`	all	Enables InnoDB metrics counters.

Important

When dealing with long queries, increasing the value of the variable performance_schema_max_digest_length will avoid having query examples truncated.

Examples

- Configuration file.

```
performance_schema=ON
performance-schema-instrument='statement/%=ON'
performance-schema-consumer-statements-digest=ON
innodb_monitor_enable=all
```

- Session.

  (`performance_schema` cannot be set in a session and must be set at server start-up.)

```
UPDATE performance_schema.setup_consumers
SET ENABLED = 'YES' WHERE NAME LIKE '%statements%';
SET GLOBAL innodb_monitor_enable = all;
```

MariaDB 10.5.7 or lower

There is no *Explain* or *Example* data shown by default in Query Analytics when monitoring MariaDB instances version 10.5.7 or lower. A workaround is to set this variable.

Percona LLC and/or its affiliates ©2024

Variable	Value	Description
performance_schema.setup_instruments	'statement/%'	List of instrumented object classes.

- Session.

```
UPDATE performance_schema.setup_instruments SET ENABLED = 'YES', TIMED = 'YES' WHERE NAME LIKE 'statement/%';
UPDATE performance_schema.setup_consumers SET ENABLED = 'YES' WHERE NAME LIKE '%statements%';
```

- Transactions

  MariaDB doesn't implement queries history for transactions. All queries executed within a transaction won't have query examples since PMM relies on the `performance_schema.events_statements_history` to grab the query example but that table won't have any query executed as part of a transaction.

  This behavior is because MariaDB doesn't implement these consumers:

```
events_transactions_current
events_transactions_history
events_transactions_history_long
```

Query response time

Query time distribution is a chart in the *Details* tab of Query Analytics showing the proportion of query time spent on various activities. It is enabled with the `query_response_time_stats` variable and associated plugins.

APPLICABLE VERSIONS

Server	Versions
Percona Server for MySQL	5.7 (**not** Percona Server for MySQL 8.0.)
MariaDB	10.0.4

Set this variable to see query time distribution charts.

Variable	Value	Description
query_response_time_stats	ON	Report *query response time distributions*. (Requires plugin installation. See below.)

- Configuration file.

```
query_response_time_stats=ON
```


Percona LLC and/or its affiliates ©2024

You must also install the plugins.

- Session.

  a. Check that `/usr/lib/mysql/plugin/query_response_time.so` exists.

  b. Install the plugins and activate.

  For MariaDB 10.3:

```
INSTALL PLUGIN QUERY_RESPONSE_TIME_AUDIT SONAME 'query_response_time.so';
INSTALL PLUGIN QUERY_RESPONSE_TIME SONAME 'query_response_time.so';
SET GLOBAL query_response_time_stats = ON;
```

  For Percona Server for MySQL 5.7:

```
INSTALL PLUGIN QUERY_RESPONSE_TIME_AUDIT SONAME 'query_response_time.so';
INSTALL PLUGIN QUERY_RESPONSE_TIME SONAME 'query_response_time.so';
INSTALL PLUGIN QUERY_RESPONSE_TIME_READ SONAME 'query_response_time.so';
INSTALL PLUGIN QUERY_RESPONSE_TIME_WRITE SONAME 'query_response_time.so';
SET GLOBAL query_response_time_stats = ON;
```

Tablestats

Some table metrics are automatically disabled when the number of tables exceeds a default limit of 1000 tables. This prevents PMM Client from affecting the performance of your database server.

The limit can be changed when adding a service on the command line with the two `pmm-admin` options:

`pmm-admin` option	Description
`--disable-tablestats`	Disables tablestats collection when the default limit is reached.
`--disable-tablestats-limit=N`	Sets the number of tables (`N`) for which tablestats collection is disabled. 0 means no limit. A negative number means tablestats is completely disabled (for any number of tables).

User statistics

APPLICABLE VERSIONS

User activity, individual table and index access details are shown on the MySQL User Details dashboard when the `userstat` variable is set.

Server	Versions
Percona Server for MySQL	5.6, 5.7, 8.0
Percona XtraDB Cluster	5.6, 5.7, 8.0
MariaDB	5.2.0+

EXAMPLES

- Configuration file.

```
userstat=ON
```

- Session.

```
SET GLOBAL userstat = ON;
```


Percona LLC and/or its affiliates ©2024

Add service

When you have configured your database server, you can add a MySQL service with the user interface or on the command line.

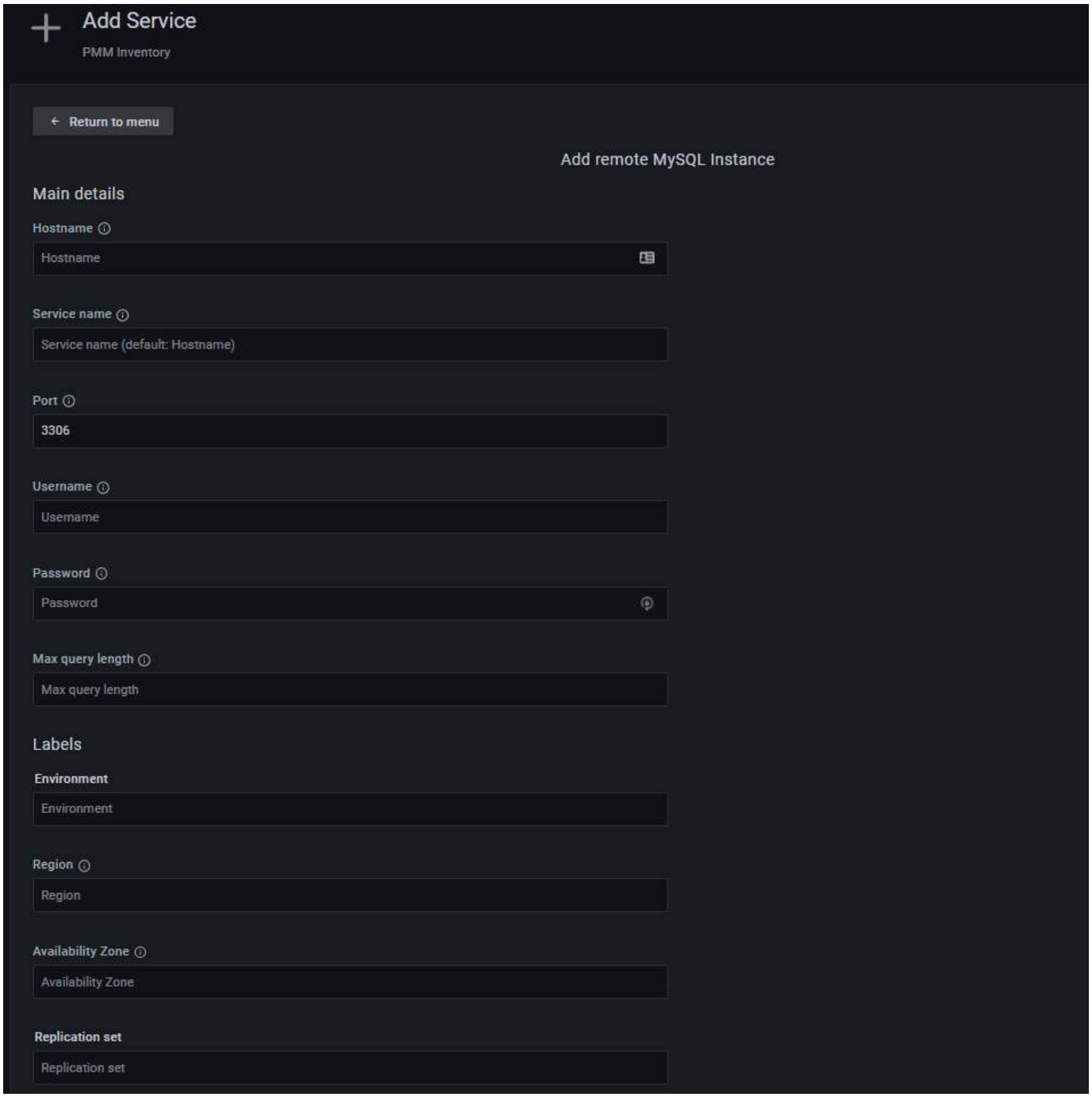
When adding a service with the command line, you must use the `pmm-admin --query-source=SOURCE` option to match the source you've chosen and configured the database server for.

With the PMM user interface, you select *Use performance schema*, or deselect it to use *slow query log*.

WITH THE USER INTERFACE

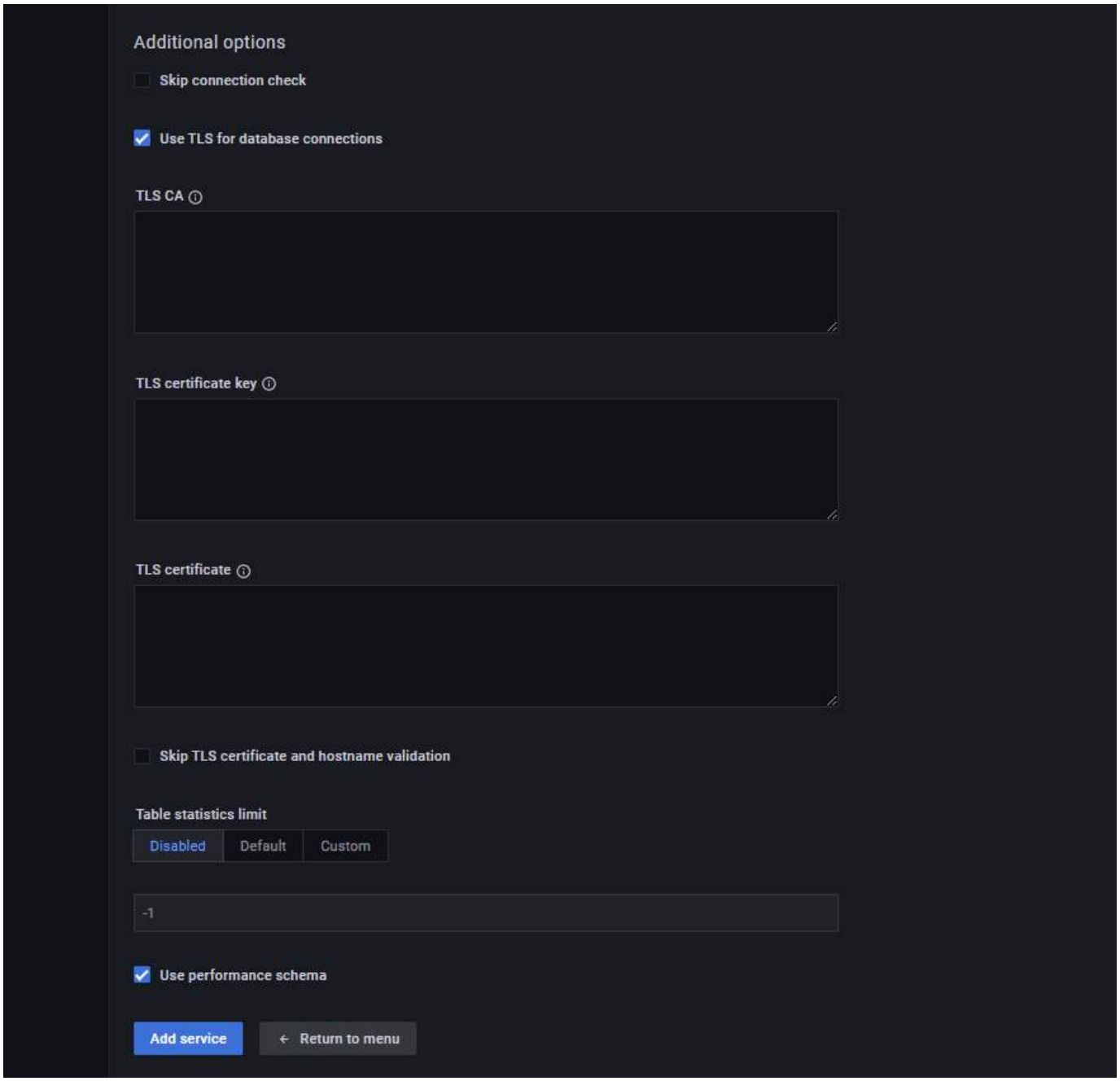
1. Select *Configuration* → *Inventory* → ⊕ *Add Service*.
2. Select *MySQL – Add a remote instance*.
3. Enter or select values for the fields.
4. Click *Add service*.

Percona LLC and/or its affiliates ©2024



If your MySQL instance is configured to use TLS, click on the *Use TLS for database connections* check box and fill in your TLS certificates and key.

Percona LLC and/or its affiliates ©2024



ON THE COMMAND LINE

Add the database server as a service using one of these example commands. If successful, PMM Client will print `MySQL Service added` with the service's ID and name. Use the `--environment` and `-custom-labels` options to set tags for the service to help identify them.

EXAMPLES

TLS connection

```
pmm-admin add mysql --username=user --password=pass --tls --tls-skip-verify --tls-ca=pathtoca.pem --tls-cert=pathtocert.pem --tls-key=pathtocertkey.pem --
server-url=http://admin:admin@127.0.0.1 --query-source=perfschema name localhost:3306
```

Slow query log

Default query source (`slowlog`), service name (`{node name}-mysql`), and service address/port (`127.0.0.1:3306`), with database server account `pmm` and password `pass`.

Percona LLC and/or its affiliates ©2024

```
pmm-admin add mysql --username=pmm --password=pass
```

Slow query log source and log size limit (1 gigabyte), service name (`MYSQL_SERVICE`) and service address/port (`191.168.1.123:3306`).

```
pmm-admin add mysql --query-source=slowlog --size-slow-logs=1GiB --username=pmm --password=pass MYSQL_SERVICE 192.168.1.123:3306
```

Slow query log source, disabled log management (use `logrotate` or some other log management tool), service name (`MYSQL_SERVICE`) and service address/port (`191.168.1.123:3306`).

```
pmm-admin add mysql --query-source=slowlog --size-slow-logs=-1GiB --username=pmm --password=pass MYSQL_SERVICE 192.168.1.123:3306
```

Default query source (`slowlog`), service name (`{node}-mysql`), connect via socket.

```
pmm-admin add mysql --username=pmm --password=pass --socket=/var/run/mysqld/mysqld.sock
```

Performance Schema

Performance schema query source, service name (`MYSQL_SERVICE`) and default service address/port (`127.0.0.1:3306`).

```
pmm-admin add mysql --query-source=perfschema --username=pmm --password=pass MYSQL_SERVICE
```

Performance schema query source, service name (`MYSQL_SERVICE`) and default service address/port (`127.0.0.1:3306`) specified with flags.

```
pmm-admin add mysql --query-source=perfschema --username=pmm --password=pass --service-name=MYSQL_SERVICE--host=127.0.0.1 --port=3306
```

Identifying services

Default query source (`slowlog`), environment labeled `test`, custom labels setting `source` to `slowlog`. (This example uses positional parameters for service name and service address.)

```
pmm-admin add mysql --environment=test --custom-labels='source=slowlog'  --username=root --password=password --query-source=slowlog MySQLSlowLog localhost:3306
```

Check the service

PMM USER INTERFACE

1. Select *Configuration* → *Inventory*.
2. In the *Services* tab, verify the *Service name*, *Addresses*, and any other relevant information in the form.
3. In the *Options* column, expand the *Details* section and check that the Agents are using the desired data source.

COMMAND LINE

Look for your service in the output of this command.

```
pmm-admin inventory list services --service-type=mysql
```

CHECK DATA

1. Open the *MySQL Instance Summary* dashboard.
2. Set the *Service Name* to the newly-added service.

Percona Server for MySQL, MariaDB

If query response time plugin was installed, check for data in the *MySQL Query Response Time Details* dashboard or select a query in *PMM Query Analytics* to see the *Query time distribution* bar.

Percona LLC and/or its affiliates ©2024

Percona XtraDB Cluster

Open the *PXC/Galera Cluster Summary* dashboard.

See also

- Percona Server for MySQL – Slow Query Log Extended
- Percona Server for MySQL – User Statistics
- MariaDB – Slow Query Log Overview
- MariaDB – Slow Query Log Extended Statistics
- MariaDB – User Statistics
- Percona Blog – PERFORMANCE_SCHEMA vs Slow Query Log
- Percona Blog – MySQL's INNODB_METRICS table
- Percona Blog – Rotating MySQL Slow Logs Safely
- Percona Blog – Impact of logging on MySQL's performance
- Percona Blog – Running Custom MySQL Queries in Percona Monitoring and Management

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-05-24

3.3.3 MongoDB

How to set up PMM to monitor a MongoDB or Percona Server for MongoDB database instance.

Before you start

Check that:

- PMM Server is installed and running with a known IP address or hostname accessible from the client node.
- PMM Client is installed and the node is registered with PMM Server.
- You have superuser (root) access on the client host.
- You have `adminUserAnyDatabase` or superuser role privilege to any database servers that you want to monitor.
- Your MongoDB server is version 4.0 or higher.

Create PMM account and set permissions

We recommend using a dedicated account to connect PMM Client to the monitored database instance.

Run the example codes below in a `mongo` session to:

- create custom roles with the privileges required for creating/restoring backups and working with Query Analytics (QAN)
- create/update a database user with these roles above, plus the built-in `clusterMonitor` role

Percona LLC and/or its affiliates ©2024

Values for username (`user`) and password (`pwd`) are examples. Replace them before using these code snippets.

CREATE ROLES WITH PRIVILEGES FOR BACKUPS AND QAN

```
db.getSiblingDB("admin").createRole({
  "role": "explainRole",
  "privileges": [
     {
        "resource": {
           "db": "",
           "collection": ""
        },
        "actions": [
           "collStats",
           "dbHash",
           "dbStats",
           "find",
           "listIndexes",
           "listCollections"
        ]
     },
     {
        "resource": {
           "db": "",
           "collection": "system.profile"
        },
        "actions": [
           "dbStats",
           "collStats",
           "indexStats"
        ]
     },
     {
        "resource": {
           "db": "",
           "collection": "system.version"
        },
        "actions": [
           "find"
        ]
     }
  ],
  "roles": []
})
```

```
db.getSiblingDB("admin").createRole({
   "role": "pbmAnyAction",
   "privileges": [{
       "resource": {
           "anyResource": true
       },
       "actions": [
           "anyAction"
       ]
   }],
   "roles": []
});
```

CREATE/UPDATE USER AND ASSIGN CREATED ROLES

```
db.getSiblingDB("admin").createUser({
   user: "pmm",
   pwd: "pmm",
   roles: [
       { role: "explainRole", db: "admin" },
       { role: "clusterMonitor", db: "admin" },
       { role: "read", db: "local" },
       { "db" : "admin", "role" : "readWrite", "collection": "" },
       { "db" : "admin", "role" : "backup" },
       { "db" : "admin", "role" : "clusterMonitor" },
       { "db" : "admin", "role" : "restore" },
       { "db" : "admin", "role" : "pbmAnyAction" }
   ]
})
```


Percona LLC and/or its affiliates ©2024

PERMISSIONS FOR ADVANCED METRICS

To fetch advanced metrics, use the following to provide additional privileges to an existing PMM user:

```
{
resource : {
    db : "",
    collection : "system.profile"
    },
actions : [
    "collStats",
    "dbStats",
    "indexStats"
   ]
}
```

Profiling

To use PMM Query Analytics, you must turn on MongoDB's profiling feature.

You can set profiling:

- permanently, by editing the MongoDB configuration file and restarting the database instance (recommended);
- when starting MongoDB, by passing arguments to `mongod` on the command line;
- until the next database instance restart, by running a command in a `mongo` session.

Important

Profiling is disabled by default as it may negatively impact the performance of the database server under specific circumstances, such as when busy servers are profiling all queries.

SET PROFILING IN THE CONFIGURATION FILE

1. Edit the configuration file (usually `/etc/mongod.conf`).
2. Create or add this to the `operationProfiling` section. (Read more.)

```
operationProfiling:
  mode: all
  slowOpThresholdMs: 200
  rateLimit: 100 # (Only available with Percona Server for MongoDB.)
```

Important

This is a YAML file. Indentation matters.

3. Restart the `mongod` service. (Example for `systemd`.)

```
systemctl restart mongod
```

SET PROFILING ON THE COMMAND LINE

Percona LLC and/or its affiliates ©2024

```
mongod --dbpath=DATABASEDIR --profile 2 --slowms 200 --rateLimit 100
```

- `--dbpath`: The path to database files (usually `/var/lib/mongo`).
- `--profile`: The MongoDB profiling level. A value of `2` tells the server to collect profiling data for *all* operations. To lower the load on the server, use a value of `1` to only record slow operations.
- `--slowms`: An operation is classified as *slow* if it runs for longer than this number of milliseconds.
- `--rateLimit`: (Only available with Percona Server for MongoDB.) The sample rate of profiled queries. A value of `100` means sample every 100th fast query. (Read more.)

> **Caution**
>
> Smaller values improve accuracy but can adversly affect the performance of your server.

SET PROFILING IN A `MONGO` SESSION

In a `mongo` session, the profiler should be enabled **per** database. For example, to enable the profiler in the `testdb`, run this:

```
> use testdb
> db.setProfilingLevel(2, {slowms: 0})
```

> **Important**
>
> If you have already added the MongoDB service to PMM, make sure to restart the PMM agent service after adjusting the profiling level.

Add service

After configuring your database server, you can add a MongoDB service either through the user interface or via the command line.

> **Important**

To monitor MongoDB sharded clusters, PMM requires access to all cluster components. Make sure to add all the config servers, shards, and at least 1-2 mongos routers. Otherwise, PMM will not be able to correctly collect metrics and populate dashboards. Keep in mind that adding all mongos routers may cause excessive overhead.

ON THE COMMAND LINE

Use `pmm-admin` to add the database server as a service using one of these example commands.

When successful, PMM Client will print `MongoDB Service added` with the service's ID and name. Use the `--environment` and `-custom-labels` options to set tags for the service to help identify them.

Percona LLC and/or its affiliates ©2024

Tips

- When adding nodes to a sharded cluster, ensure to add each node separately using the `--cluster mycluster` option. This allows the MongoDB Cluster Summary dashboard to populate correctly.
- You can also use the `--replication-set` option to specify a replication set, altough they are automatically detected. For instance, you can use `--replication-set config` for your config servers; `--replication-set rs1` for your servers in the first replica set, `--replication-set rs2` for your servers in the second replica set, and so on.
- When connecting to an Atlas instance, keep in mind that direct connections are not supported. Make sure to use the `--direct-connection=false` option with `pmm-admin`. Otherwise, replicaset status functionality will not work as expected, and the MongoDB Overview dashboard widget will show invalid values.

EXAMPLES

```
pmm-admin add mongodb \
--username=pmm_mongodb --password=password \
--query-source=profiler --cluster=mycluster
```

```
pmm-admin add mongodb \
--username=pmm_mongodb --password=password \
--service-name=mymongosvc --host=127.0.0.1 --port=27017 \
--enable-all-collectors
```

Connecting via SSL/TLS

```
pmm-admin add mongodb --tls \
--tls-certificate-key-file=PATHTOCER \
--tls-certificate-key-file-password=IFPASSWORDTOCERTISSET \
--tls-ca-file=PATHTOCACERT \
--authentication-mechanism=AUTHENTICATION-MECHANISM \
--authentication-database=AUTHENTICATION-DATABASE \
--cluster=mycluster \
--enable-all-collectors
```

where:

- `PATHTOCERT`: Path to TLS certificate file.
- `IFPASSWORDTOCERTISSET`: Password for TLS certificate file.
- `PATHTOCACERT`: Path to certificate authority file.
- `AUTHENTICATION-MECHANISM`: Authentication mechanism. Default is empty. Use `MONGODB-X509` for SSL certificates.
- `AUTHENTICATION-DATABASE`: Authentication database. Default is empty. Use `$external` for SSL certificates.

WITH THE PMM INTERFACE

Use this option when you don't have direct access to the underlying host to install pmm-agent locally.

1. Select *Configuration* → *Inventory*.
2. Select *MongoDB – Add a remote instance*.
3. Enter or select values for the fields.
4. Click *Add service*.

Check the service

ON THE COMMAND LINE

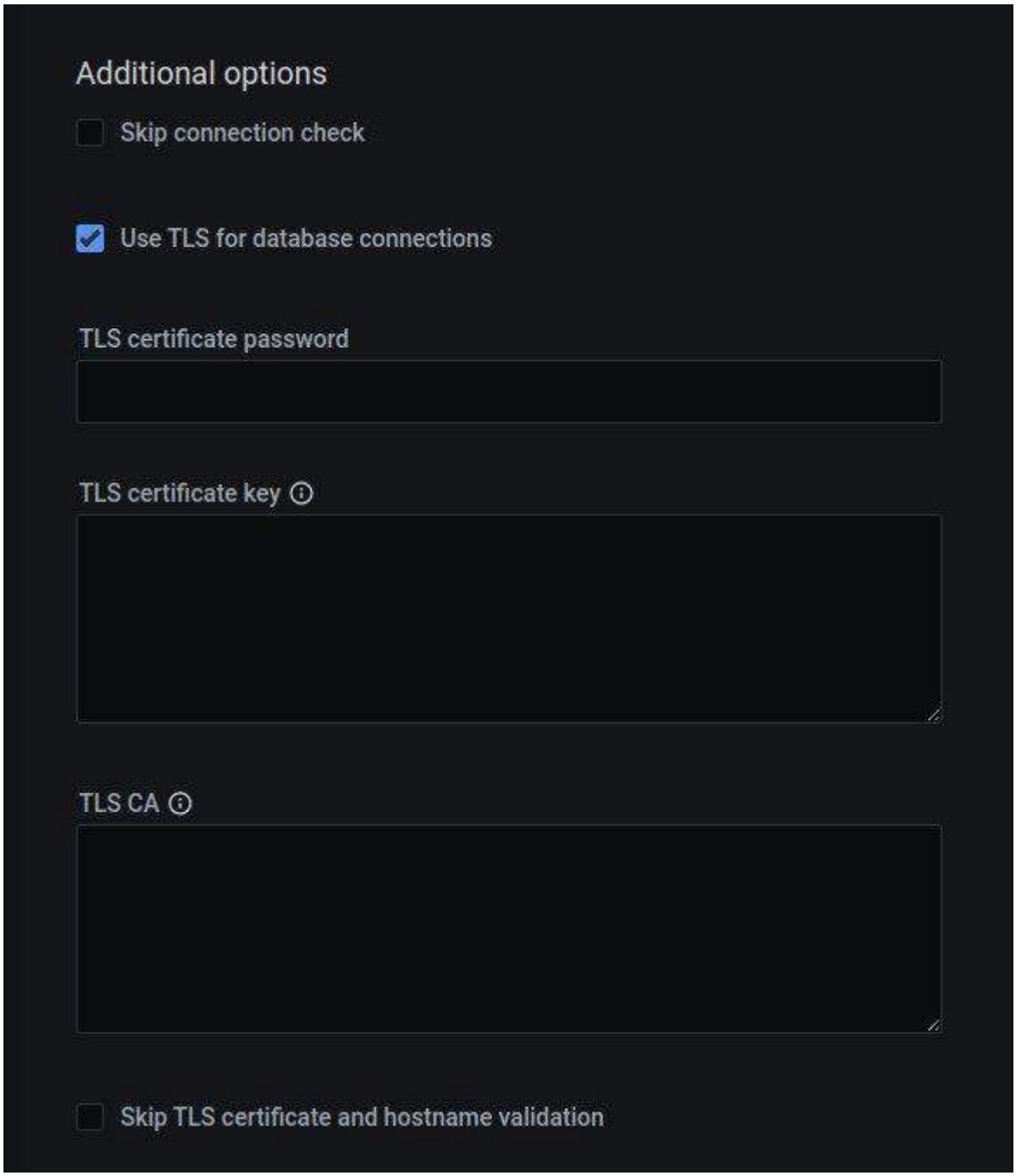
Look for your service in the output of this command.

```
pmm-admin inventory list services --service-type=mongodb
```


Percona LLC and/or its affiliates ©2024

WITH THE PMM INTERFACE

1. Select *Configuration* → *Inventory*.
2. In the *Services* tab, verify the *Service name*, *Addresses*, and any other relevant values used when adding the service.
3. In the *Options* column, expand the *Details* section and check that the Agents are using the desired data source.
4. If your MongoDB instance is configured to use TLS, click on the **Use TLS for database connection** check box and fill in TLS certificates and keys. If you use TLS, the authentication mechanism is automatically set to `MONGODB-X509`.



CHECK DATA

1. Open the *MongoDB Instances Overview* dashboard.
2. Set the *Service Name* to the newly-added service.

Percona LLC and/or its affiliates ©2024

Query Analytics

1. Open *PMM Query Analytics*.
2. In the *Filters* panel:
    a. Under *Service Name*, select your service.
    b. Under *Service Type* select `mongodb`.

Remove service

ON THE COMMAND LINE

```
pmm-admin remove mongodb SERVICE_NAME
```

- `SERVICE_NAME`: The name the service was added as. (Find it with `pmm-admin list`.)

WITH THE PMM INTERFACE

Use this option only top remove agents installed through the PMM interface.

1. Select *Configuration* → *Inventory*.
2. In the first column, click the tick box for the service you want to remove.
3. Click *Delete*.
4. On the *Confirm action* dialog window:
    a. (Optional) Select *Force mode* to also delete associated agents.
    b. Click *Proceed*.

See also

- `pmm-admin add mongodb`
- Troubleshooting connection difficulties

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-05-27

3.3.4 PostgreSQL

How to set up PMM to monitor a PostgreSQL or Percona Distribution for PostgreSQL database instance.

Percona LLC and/or its affiliates ©2024

Summary

- Create PMM account and set permissions.
- Choose, install and configure an extension:
  - `pg_stat_statements`, or,
  - `pg_stat_monitor`.
- Add service.
- Check the service.

Before you start

Check that:

- PMM Server is installed and running with a known IP address accessible from the client node.
- PMM Client is installed and the node is registered with PMM Server.
- You have superuser (root) access on the client host.
- You have superuser access to any database servers that you want to monitor.

(PMM follows PostgreSQL's end-of-life policy. For specific details on supported platforms and versions, see Percona's Software Platform Lifecycle page.)

Create a database account for PMM

We recommend creating a PMM database account that can connect to the `postgres` database with the `SUPERUSER` role.

1. Create a user. This example uses `pmm`. (Replace `******` with a strong password of your choice.)

   ```
   CREATE USER pmm WITH SUPERUSER ENCRYPTED PASSWORD '*******';
   ```

   If your database runs on Amazon RDS / Aurora PostgreSQL, The SUPERUSER cannot be assigned.
   So we have to create the user first and then grant the `rds_superuser` role to it.

   ```
   CREATE USER pmm WITH ENCRYPTED PASSWORD '*******';
   GRANT rds_superuser TO pmm;
   ```

   Optionally, you can also set up a connection limit (only if the user is not a SUPERUSER):

   ```
   ALTER USER pmm CONNECTION LIMIT 10;
   ```

2. PMM must be able to log in locally as this user to the PostgreSQL instance. To enable this, edit the `pg_hba.conf` file. If not already enabled by an existing rule, add:

   ```
   local   all             pmm                                     md5
   # TYPE  DATABASE        USER        ADDRESS                     METHOD
   ```

   (Ignore the second line. It is a comment to show field alignment.)

3. Reload the configuration:

   ```
   su - postgres
   psql -c "select pg_reload_conf()"
   ```

4. Check local login.

   ```
   psql postgres pmm -c "\conninfo"
   ```

5. Enter the password for the `pmm` user when prompted.

Percona LLC and/or its affiliates ©2024

Choose and configure an extension

Decide which database extension to use, and configure your database server for it. The choices are:

1. `pg_stat_statements`, the original extension created by PostgreSQL, part of the `postgresql-contrib` package available on Linux.
2. `pg_stat_monitor` is a new extension created by Percona. `pg_stat_monitor` has all the features of `pg_stat_statements` but adds *bucket-based data aggregation*, provides more accurate data, and can expose Query Examples.

Here are the benefits and drawbacks of each.

	Benefits	Drawbacks
`pg_stat_statements`	1. Part of official `postgresql-contrib` package.	1. No aggregated statistics or histograms 2. No query examples 3. No query examples 4. No plan execution information
`pg_stat_monitor`	1. Builds on `pg_stat_monitor` features. 2. Bucket-based aggregation.	

For a more detailed comparison of extensions, see Comparison with pg_stat_statements in the `pg_stat_monitor` documentation.

Bucket-based data aggregation

`pg_stat_monitor` collects statistics and aggregates data in a data collection unit called a *bucket*. These are linked together to form a *bucket chain*.

You can specify:

- the number of buckets (the length of the chain);
- how much space is available for all buckets;
- a time limit for each bucket's data collection (the *bucket expiry*).

When a bucket's expiration time is reached, accumulated statistics are reset and data is stored in the next available bucket in the chain.

When all buckets in the chain have been used, the first bucket is reused and its contents are overwritten.

If a bucket fills before its expiration time is reached, data is discarded.

Percona LLC and/or its affiliates ©2024

PG_STAT_STATEMENTS

Install

- Debian/Ubuntu

  Root permissions

  ```
  apt install -y postgresql-contrib
  ```

- Red Hat/CentOS

  Root permissions

  ```
  yum install -y postgresql-contrib
  ```

Configure

1. Add these lines to your `postgresql.conf` file:

   ```
   shared_preload_libraries = 'pg_stat_statements'
   track_activity_query_size = 2048 # Increase tracked query string size
   pg_stat_statements.track = all   # Track all statements including nested
   track_io_timing = on             # Capture read/write stats
   ```

2. Restart the database server. After the restart, the extension starts capturing statistics from every database.
3. Install the extension.

   ```
   psql postgres postgres -c "CREATE EXTENSION pg_stat_statements SCHEMA public"
   ```

   This command creates the view where you can access the collected statistics.

We recommend that you create the extension for the `postgres` database. In this case, you receive access to the statistics collected from every database.

You can now add the service.

PG_STAT_MONITOR

`pg_stat_monitor` has been tested with:

- PostgreSQL versions 11, 12, 13, 14, 15.
- Percona Distribution for PostgreSQL versions 11, 12, 13, 14, 15.

Install

- If you use *Percona Distribution for PostgreSQL*, you can install the extension with your Linux package manager. See Installing Percona Distribution for PostgreSQL.
- If you use *PostgreSQL* you can install by downloading and compiling the source code. See Installing `pg_stat_monitor`.

Configure

1. Set or change the value for `shared_preload_library`.

Percona LLC and/or its affiliates ©2024

In your `postgresql.conf` file:

```
shared_preload_libraries = 'pg_stat_monitor'
```

Caution

If you use both `pg_stat_statements` and `pg_stat_monitor`, set `pg_stat_monitor` **after** `pg_stat_statements`:

```
shared_preload_libraries = 'pg_stat_statements, pg_stat_monitor'
```

2. Set configuration values.

   In your `postgresql.conf` file:

   ```
   pg_stat_monitor.pgsm_query_max_len = 2048
   ```

   Caution

   It is important to set maximal length of query to 2048 characters or more for PMM to work properly.

   You can get a list of other available settings with `SELECT * FROM pg_stat_monitor_settings;`.

   Other important parameters are:

   ```
   pg_stat_monitor.pgsm_normalized_query
   ```

   and

   ```
   pg_stat_monitor.pgsm_enable_query_plan
   ```

   If the value for `pg_stat_monitor.pgsm_normalized_query` is set to 1, the actual query values are replaced by placeholders. If the value is 0, the examples are given in QAN. Examples can be found in QAN details tab example.

   If `pg_stat_monitor.pgsm_enable_query_plan` is enabled, the query plans are captured and will be available in the `Plan` tab on the Query Analytics dashboard.

   See `pg_stat_monitor` online documentation for details about available parameters.

3. Start or restart your PostgreSQL instance. The extension starts capturing statistics from every database.
4. In a `psql` session:

   ```
   CREATE EXTENSION pg_stat_monitor;
   ```

   This command creates the view where you can access the collected statistics.

   We recommend that you create the extension for the `postgres` database. In this case, you receive the access to the statistics, collected from every database.

5. Check the version.

   ```
   SELECT pg_stat_monitor_version();
   ```

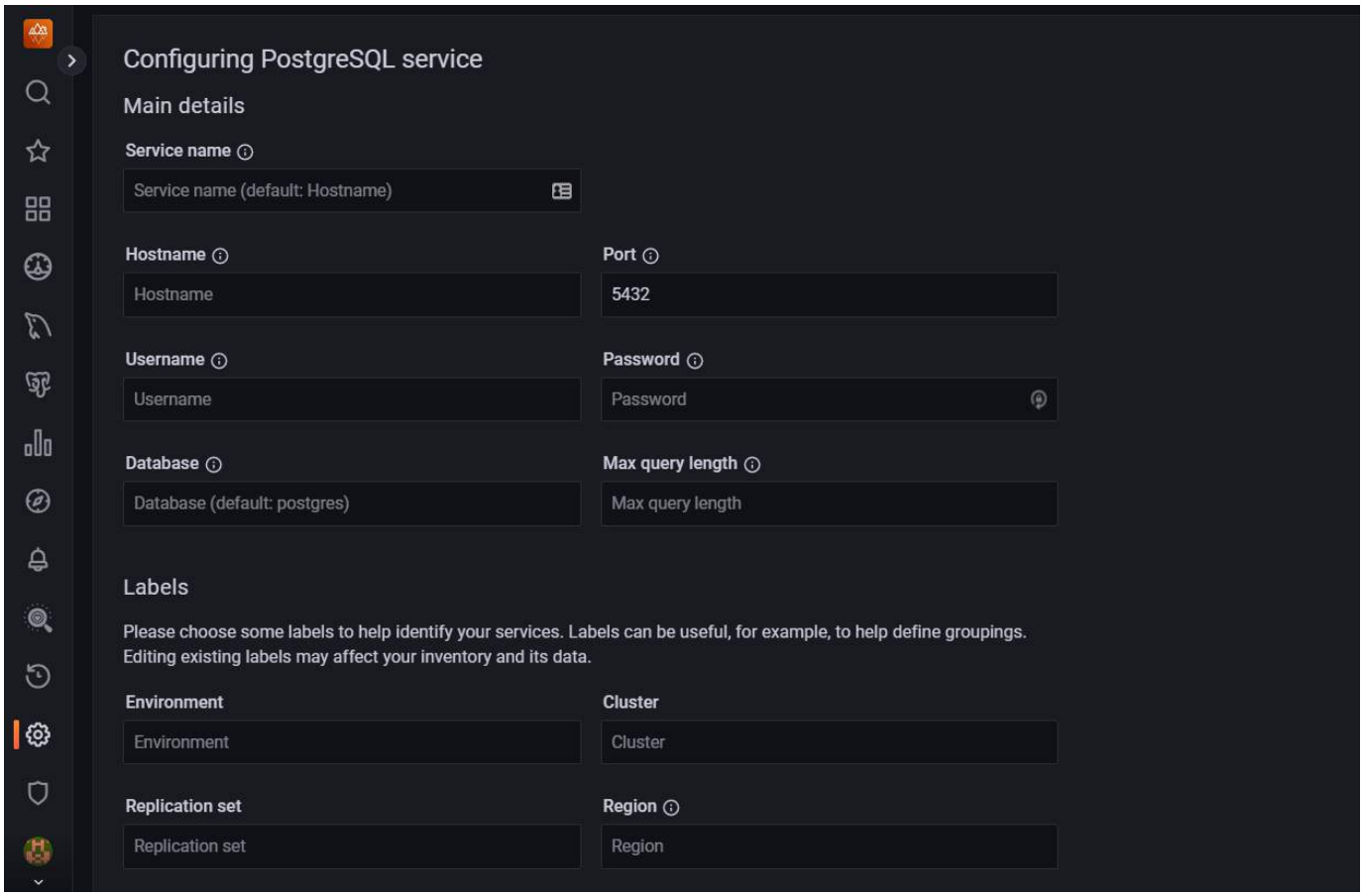

Percona LLC and/or its affiliates ©2024

Add service

When you have configured your database server, you can add a PostgreSQL service with the user interface or on the command line.

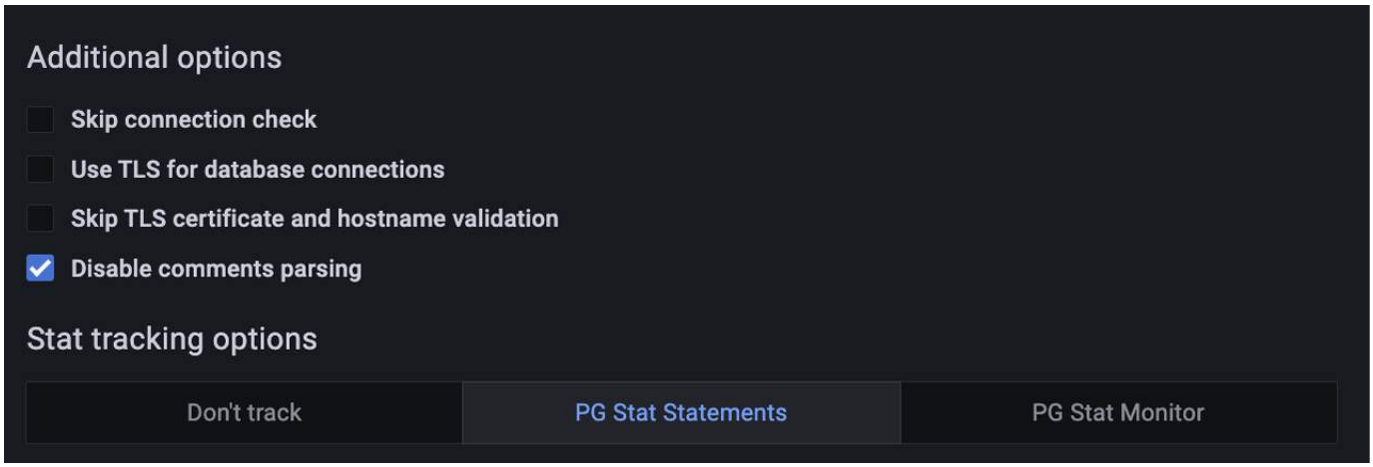
WITH THE USER INTERFACE

1. Select *Configuration* → *Inventory* → ⊕ *Add Service*.
2. Select *PostgreSQL – Add a remote instance*.
3. Enter or select values for the fields.
4. Click *Add service*.



If your PostgreSQL instance is configured to use TLS, click on the *Use TLS for database connections* check box and fill in your TLS certificates and key.

Percona LLC and/or its affiliates ©2024



Note

For TLS connection to work SSL needs to be configured in your PostgreSQL instance. Make sure SSL is enabled in the server configuration file `postgresql.conf`, and that hosts are allowed to connect in the client authentication configuration file `pg_hba.conf`. (See PostgreSQL documentation on Secure TCP/IP Connections with SSL.)

AUTO-DISCOVERY LIMIT

Starting with PMM 2.41.0 you can set a limit for **Auto-discovery** in PostgreSQL, a feature that dynamically discovers all databases in your PostgreSQL instance.

Limiting **Auto-discovery** reduces connections and prevents high CPU and RAM usage caused by multiple databases.

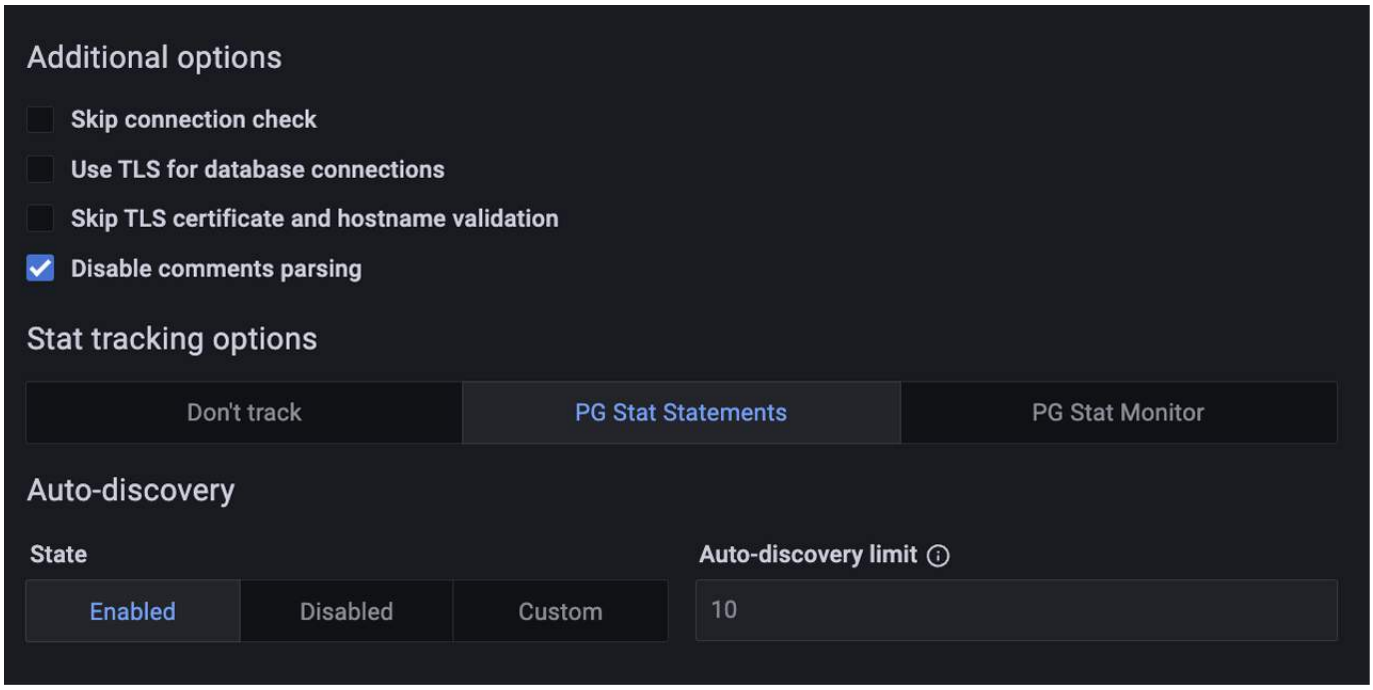
Caution

Limiting auto-discovery may result in fewer metrics being captured from the non-primary databases. Ensure that you set the limit appropriately:

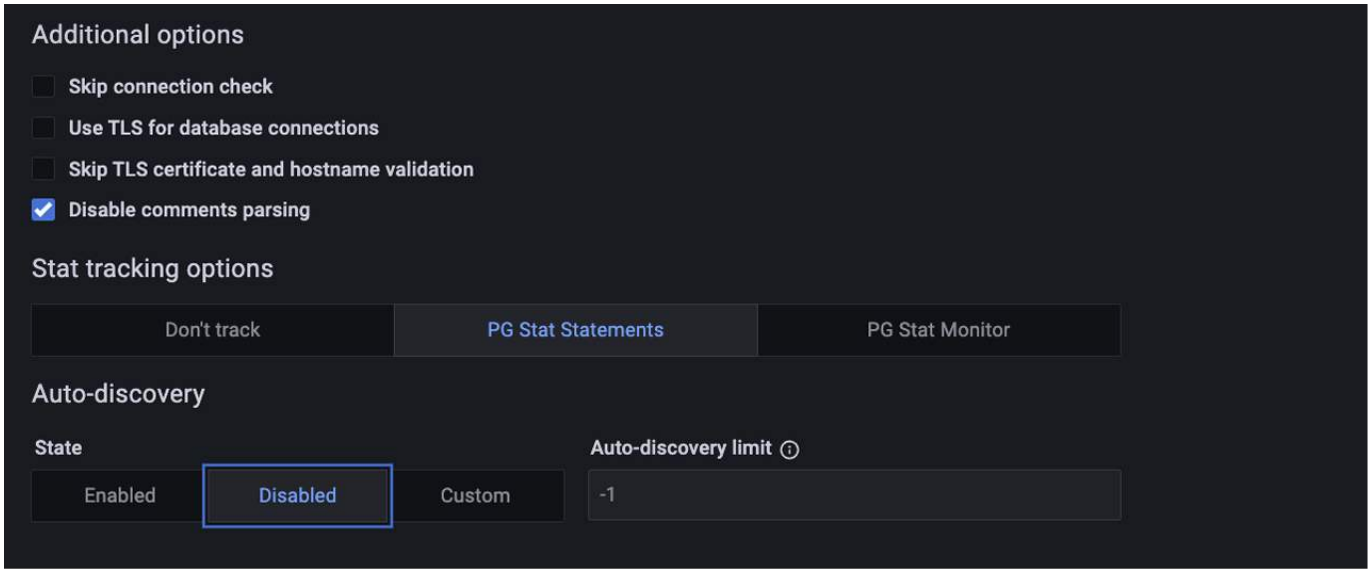
- Setting a high limit may impact performance adversely.
- Setting a low limit might result in some missing metrics due to Auto-discovery being disabled.

By default, **Auto-discovery** is enabled (server defined with a limit 10).

Percona LLC and/or its affiliates ©2024

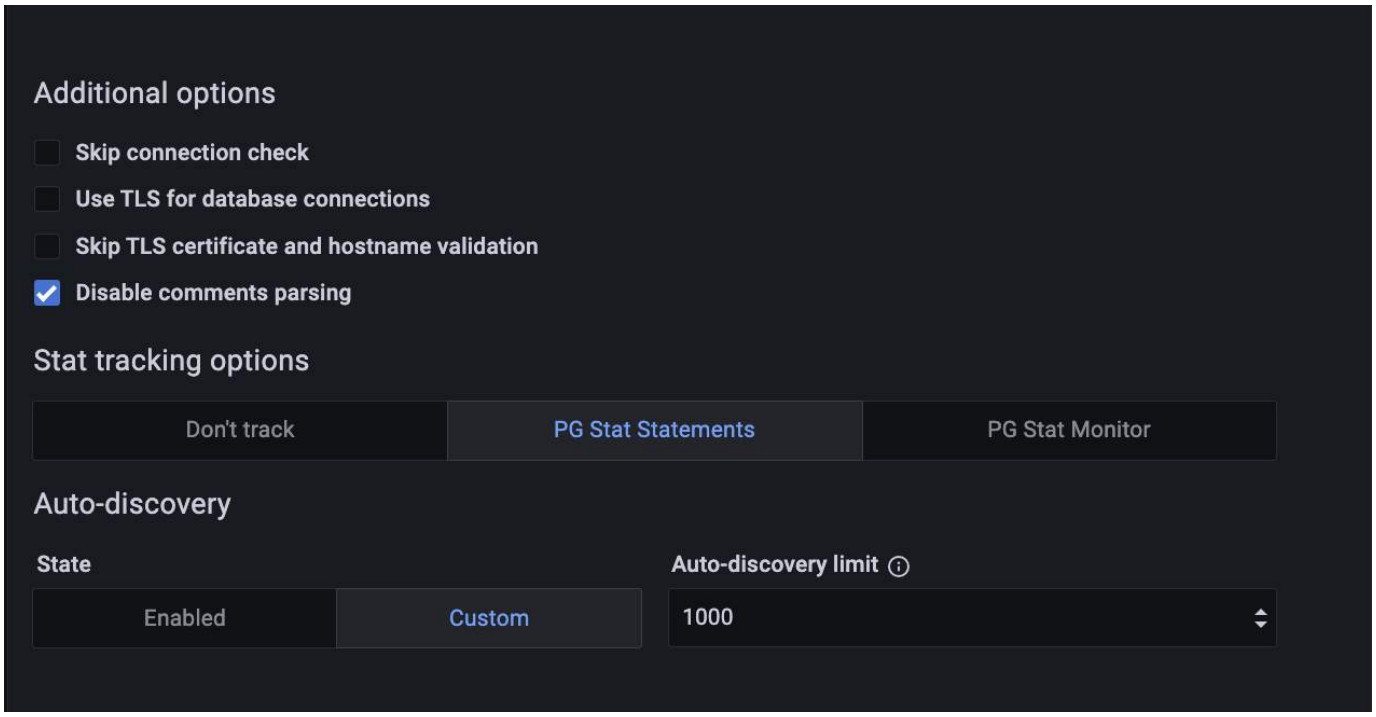


When you select **Disabled**, the **Auto-discovery limit** will be set to `-1`.



For a custom value, select **Custom** and enter or choose your preferred value from the **Auto-discovery limit** field.

Percona LLC and/or its affiliates ©2024



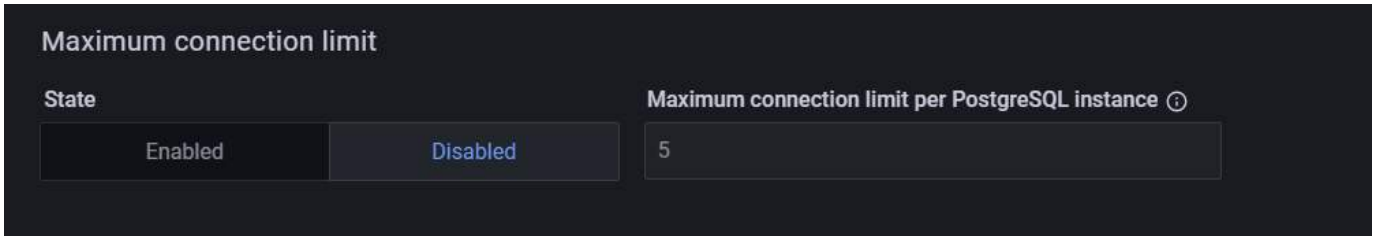
MAXIMUM CONNECTION LIMIT

Starting with PMM 2.42.0, you can set a maximum limit on the number of connections that the PostgreSQL exporter can open to the same PostgreSQL instance.

By setting a maximum connection limit, you can prevent excessive connections during concurrent operations, and ensure that connections are closed promptly to avoid idle connections.

When adjusting the maximum number of connections, consider the following:

- higher values might be needed for larger or busier instances.
- setting the limit too high can impact performance.
- if no limit is specified or the option is disabled, the server will manage the connection limits automatically.



ON THE COMMAND LINE

Add the database server as a service using one of these example commands. If successful, PMM Client will print `PostgreSQL Service added` with the service's ID and name. Use the `--environment` and `-custom-labels` options to set tags for the service to help identify them.

EXAMPLES

Add instance with default service name (`<node>-postgresql`).

```
pmm-admin add postgresql \
--username=pmm \
--password=password \
```


Percona LLC and/or its affiliates ©2024

```
--server-url=https://admin:admin@X.X.X.X:443 \
--server-insecure-tls
```

- `<user name>`: The PostgreSQL PMM user
- `<password>`: The PostgreSQL user credentials.

The service name will be automatically chosen.

Add instance with specified service name.

```
pmm-admin add postgresql \
--username=pmm \
--password=password \
--server-url=https://admin:admin@X.X.X.X:443 \
--server-insecure-tls \
--service-name=SERVICE-NAME
```

Add instance to connect with a UNIX socket.

```
pmm-admin add postgresql --socket=/var/run/postgresql
```

where: - `SOCKET`: directory containing the socket

Connecting via SSL/TLS

```
pmm-admin add postgresql --tls \
--tls-cert-file=PATHTOCERT \
--tls-ca-file=PATHTOCACERT \
--tls-key-file=PATHTOKEY \
--host=HOST \
--port=PORT \
--username=USER \
--service-name=SERVICE-NAME
```

where:

- `PATHTOCERT`: Path to client certificate file.
- `PATHTOCACERT`: Path to certificate authority file.
- `PATHTOKEY`: Path to client key file.
- `HOST`: Instance hostname or IP.
- `PORT`: PostgreSQL service port number.
- `USER`: Database user allowed to connect via TLS. Should match the common name (CN) used in the client certificate.
- `SERVICE`: Name to give to the service within PMM.

Automatic discovery limit via CLI

Starting with PMM 2.41.0, there is a new flag in `pmm-admin` to limit Auto-discovery:

`--auto-discovery-limit=XXX`

- If number of databases > Auto-discovery limit, then auto discovery is **OFF**
- If number of databases <= Auto-discovery limit, then auto discovery is **ON**
- If the Auto-discovery limit is not defined, it takes the default value, which is 0 (server defined with limit 10), and Auto-discovery is **ON**(if you do not have more than 10 databases).
- If Auto-discovery limit < 0 then auto discovery is **OFF**.

Example:

If you set the limit to 10 and your PostgreSQL instance has 11 databases, automatic discovery will be disabled.

Percona LLC and/or its affiliates ©2024

```
pmm-admin add postgresql --username="pmm-agent" --password="pmm-agent-password" --auto-discovery-limit=10
```

Check the service

CHECK SERVICE – PMM USER INTERFACE

1. Select *Configuration* → *Inventory*.
2. In the *Services* tab, verify the *Service name*, *Address* and any other relevant details.
3. In the *Options* column, expand the *Details* section and check that the Agents are using the desired data source.

CHECK SERVICE – COMMAND LINE

Look for your service in the output of this command.

```
pmm-admin inventory list services
```

If using Docker, use `docker exec pmm-client pmm-admin inventory list services`

CHECK DATA

1. Open the *PostgreSQL Instance Summary* dashboard.
2. Set the *Service Name* to the newly-added service.

RUNNING CUSTOM QUERIES

The PostgreSQL exporter can run custom queries to add new metrics not provided by default.
Those custom queries must be defined in the `/usr/local/percona/pmm2/collectors/custom-queries/postgresql` in the same host where the exporter is running. There are 3 directories inside it: - high-resolution/ - every 5 seconds - medium-resolution/ - every 10 seconds - low-resolution/ - every 60 seconds

Depending on the desired resolution for your custom queries, you can place a file with the queries definition. The file is a yaml where each query can have these fields:

```
query_name:
   query: the query definition
   master: boolean to specify if the query should be executed only in the master
   metrics:
     - metric name:
         usage: GAUGE, LABEL, COUNTER, MAPPEDMETRIC or DURATION
         description: a human readable description
```

Example

```
pg_postmaster_uptime:
   query: "select extract(epoch from current_timestamp - pg_postmaster_start_time()) as seconds"
   master: true
   metrics:
     - seconds:
         usage: "GAUGE"
         description: "Service uptime"
```

Check the see also section for a more detailed description on MySQL custom queries with more examples about how to use custom queries in dashboards.

Percona LLC and/or its affiliates ©2024

See also

- `pmm-admin` man page for `pmm-admin add postgresql`
- Configuring Percona Repositories with percona-release
- Percona Blog – Running Custom MySQL Queries in Percona Monitoring and Management

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-06-11

3.3.5 ProxySQL

Use the `proxysql` alias to enable ProxySQL performance metrics monitoring.

USAGE

```
pmm-admin add proxysql --username=pmm --password=pmm
```

where `username` and `password` are credentials for the administration interface of the monitored ProxySQL instance. You should configure a read-only account for monitoring using the `admin-stats_credentials` variable in ProxySQL

Additionally, two positional arguments can be appended to the command line flags: a service name to be used by PMM, and a service address. If not specified, they are substituted automatically as `<node>-proxysql` and `127.0.0.1:6032`.

The output of this command may look as follows:

```
pmm-admin add proxysql --username=pmm --password=pmm
```

```
ProxySQL Service added.
Service ID  : /service_id/f69df379-6584-4db5-a896-f35ae8c97573
Service name: ubuntu-proxysql
```

Beside positional arguments shown above you can specify service name and service address with the following flags: `--service-name`, and `--host` (the hostname or IP address of the service) and `--port` (the port number of the service), or `--socket` (the UNIX socket path). If both flag and positional argument are present, flag gains higher priority. Here is the previous example modified to use these flags for both host/port or socket connections:

```
pmm-admin add proxysql --username=pmm --password=pmm --service-name=my-new-proxysql --host=127.0.0.1 --port=6032
pmm-admin add proxysql --username=pmm --password=pmm --service-name=my-new-proxysql --socket=/tmp/proxysql_admin.sock
```

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2022-02-04

3.3.6 Amazon RDS

Required settings

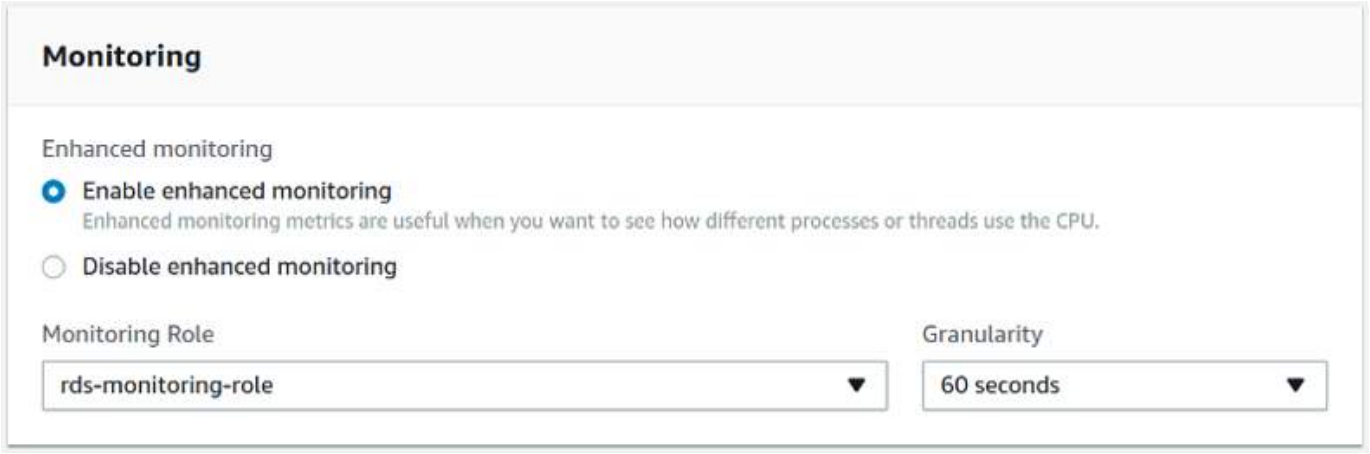
It is possible to use PMM for monitoring Amazon RDS. In this case, the PMM Client is not installed on the host where the database server is deployed. By using the PMM web interface, you connect to the Amazon RDS DB instance. You only need to provide the IAM user access key (or assign an IAM role) and PMM discovers the Amazon RDS DB instances available for monitoring.

First of all, ensure that there is the minimal latency between PMM Server and the Amazon RDS instance.

Network connectivity can become an issue for VictoriaMetrics to scrape metrics with 1 second resolution. We strongly suggest that you run PMM Server on AWS (Amazon Web Services) in the same availability zone as Amazon RDS instances.

It is crucial that *enhanced monitoring* be enabled for the Amazon RDS DB instances you intend to monitor.

Set the *Enable Enhanced Monitoring* option in the settings of your Amazon RDS DB instance.



Creating an IAM user with permission to access Amazon RDS DB instances

It is recommended that you use an IAM user account to access Amazon RDS DB instances instead of using your AWS account. This measure improves security as the permissions of an IAM user account can be limited so that this account only grants access to your Amazon RDS DB instances. On the other hand, you use your AWS account to access all AWS services.

The procedure for creating IAM user accounts is well described in the Amazon RDS documentation. This section only goes through the essential steps and points out the steps required for using Amazon RDS with Percona Monitoring and Management.

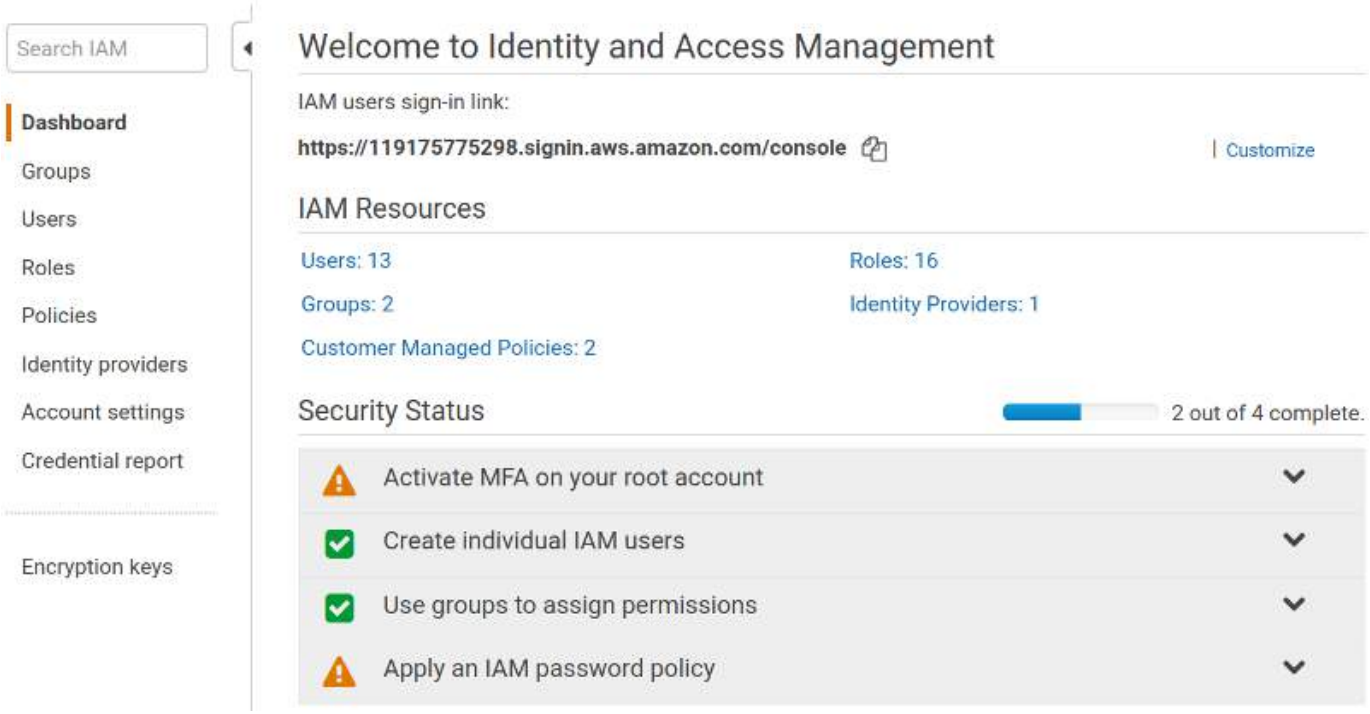
The first step is to define a policy which will hold all the necessary permissions. Then, you need to associate this policy with the IAM user or group. In this section, we will create a new user for this purpose.

Creating a policy

A policy defines how AWS services can be accessed. Once defined it can be associated with an existing user or group.

To define a new policy use the IAM page at AWS.

Percona LLC and/or its affiliates ©2024

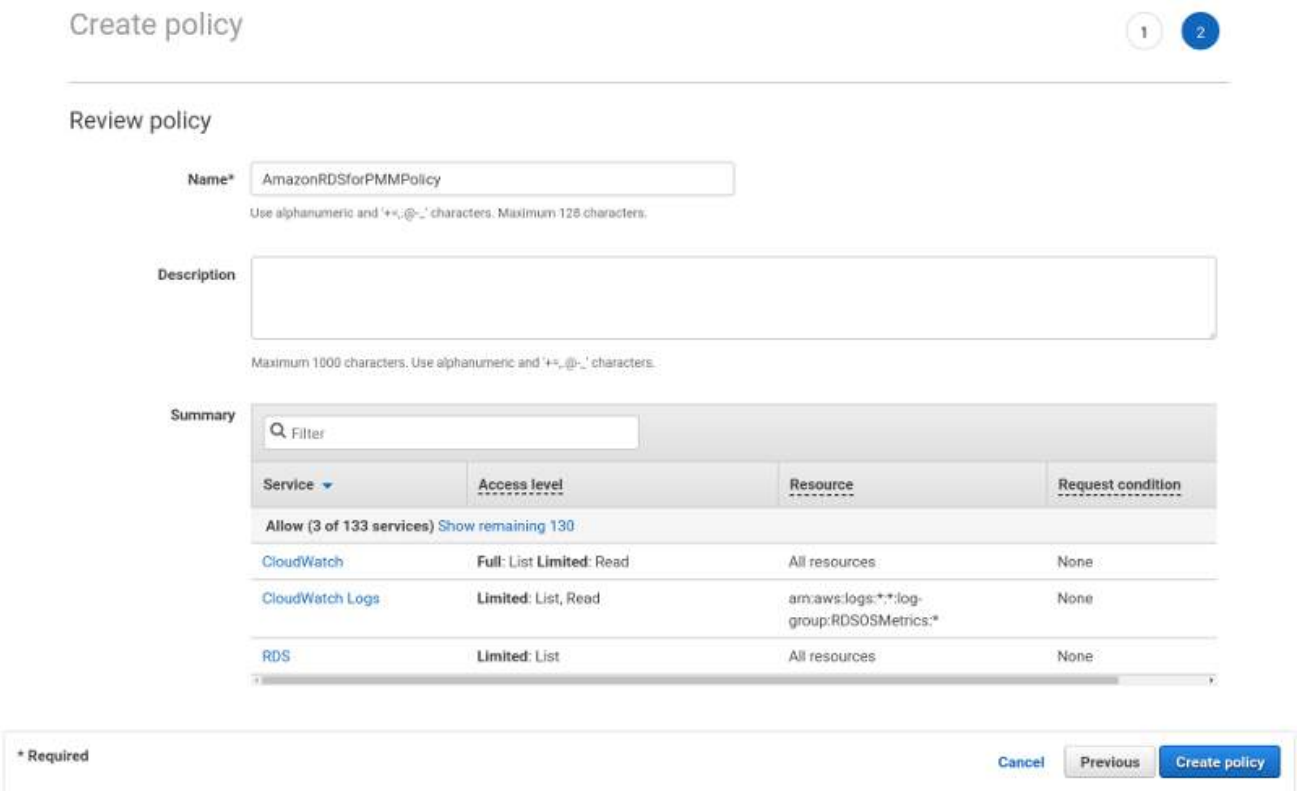


1. Select the *Policies* option on the navigation panel and click the *Create policy* button.
2. On the *Create policy* page, select the JSON tab and replace the existing contents with the following JSON document.

```
{ "Version": "2012-10-17",
  "Statement": [{ "Sid": "Stmt1508404837000",
              "Effect": "Allow",
              "Action": [ "rds:DescribeDBInstances",
                          "cloudwatch:GetMetricStatistics",
                          "cloudwatch:ListMetrics"],
                          "Resource": ["*"] },
             { "Sid": "Stmt1508410723001",
               "Effect": "Allow",
               "Action": [ "logs:DescribeLogStreams",
                           "logs:GetLogEvents",
                           "logs:FilterLogEvents" ],
                           "Resource": [ "arn:aws:logs:*:*:log-group:RDSOSMetrics:*" ]}
          ]
}
```

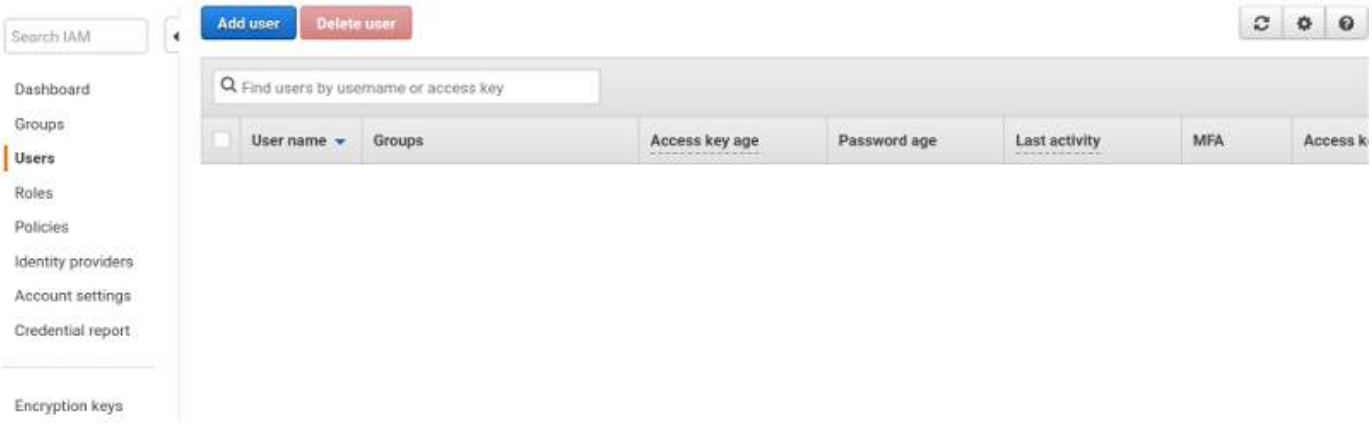
3. Click *Review policy* and set a name to your policy, such as `AmazonRDSforPMMPolicy`. Then, click the *Create policy* button.

Percona LLC and/or its affiliates ©2024



Creating an IAM user

Policies are attached to existing IAM users or groups. To create a new IAM user, select *Users* on the Identity and Access Management page at AWS. Then click *Add user* and complete the following steps:



1. On the *Add user* page, set the user name and select the *Programmatic access* option under *Select AWS access type*. Set a custom password and then proceed to permissions by clicking the *Permissions* button.
2. On the *Set permissions* page, add the new user to one or more groups if necessary. Then, click *Review*.
3. On the *Add user* page, click *Create user*.

Creating an access key for an IAM user

To discover an Amazon RDS DB instance in PMM, you either need to use the access key and secret access key of an existing IAM user or an IAM role. To create an access key for use with PMM, open the IAM console and click *Users* on the navigation pane. Then, select your IAM user.

Percona LLC and/or its affiliates ©2024

To create the access key, open the *Security credentials* tab and click the *Create access key* button. The system automatically generates a new access key ID and a secret access key that you can provide on the *PMM Add Instance* dashboard to have your Amazon RDS DB instances discovered.

In case, the PMM Server and Amazon RDS DB instance were created by using the same AWS account, you do not need create the access key ID and secret access key manually. PMM retrieves this information automatically and attempts to discover your Amazon RDS DB instances.

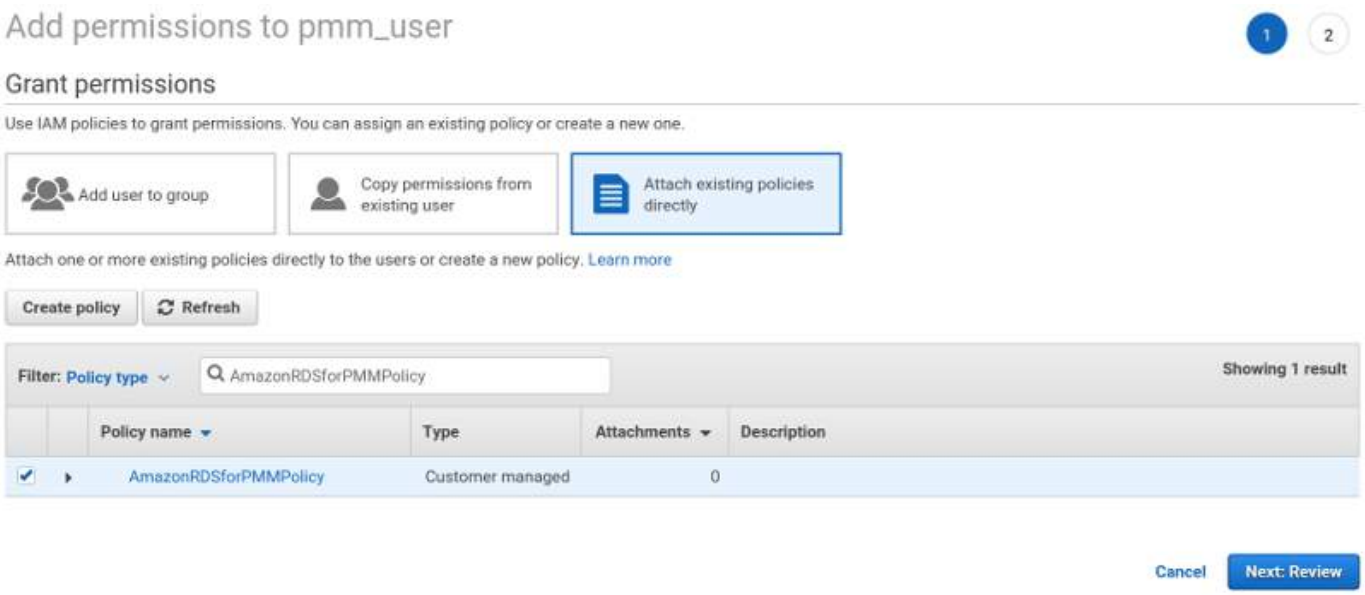
Attaching a policy to an IAM user

The last step before you are ready to create an Amazon RDS DB instance is to attach the policy with the required permissions to the IAM user.

First, make sure that the Identity and Access Management page is open and open *Users*. Then, locate and open the IAM user that you plan to use with Amazon RDS DB instances. Complete the following steps, to apply the policy:

1. On the *Permissions* tab, click the *Add permissions* button.
2. On the *Add permissions* page, click *Attach existing policies directly*.
3. Using the *Filter*, locate the policy with the required permissions (such as `AmazonRDSforPMMPolicy`).
4. Select a check-box next to the name of the policy and click *Review*.
5. The selected policy appears on the *Permissions summary* page. Click *Add permissions*.

The `AmazonRDSforPMMPolicy` is now added to your IAM user.



Creating an IAM role

Instead of creating an IAM user you can create an IAM role for a service, to discover Amazon RDS DB instances automatically without the need for access and secret keys. (But this only works if you are running PMM through AWS.)

To create an IAM role open the IAM console and click *Roles* on the navigation pane.

1. Click the *Create role* button.
2. Select *AWS service* and select *EC2* for the use case.
3. Click the *Next: Permissions* button.

Percona LLC and/or its affiliates ©2024

4. Find the policy created previously and select it.
5. Click the *Next: Tags* button.
6. (Optional) Add a metadata tag to the role.
7. Click the *Next: Review* button.
8. Fill the role name and description.
9. Click the *Create role* button

After the role is created EC2 instances running PMM will have permissions to discover RDS DB instances.

> It's also possible to create an IAM role to delegate permissions to an IAM user or to add permissions to a user belonging to another AWS account. See the official AWS documentation on creating IAM roles.

Setting up the Amazon RDS DB Instance

Query Analytics requires Configuring Performance Schema as the query source, because the slow query log is stored on the AWS (Amazon Web Services) side, and QAN agent is not able to read it. Enable the `performance_schema` option under `Parameter Groups` in Amazon RDS.

> **Important**
>
> Enabling Performance Schema on T2 instances is not recommended because it can easily run the T2 instance out of memory.

When adding a monitoring instance for Amazon RDS, specify a unique name to distinguish it from the local instance. If you do not specify a name, it will use the client's host name.

Create the `pmm` user with the following privileges on the Amazon RDS instance that you want to monitor:

```
CREATE USER 'pmm'@'%' IDENTIFIED BY 'pass';
GRANT SELECT, PROCESS, REPLICATION CLIENT ON *.* TO 'pmm'@'%';
ALTER USER 'pmm'@'%' WITH MAX_USER_CONNECTIONS 10;
GRANT SELECT, UPDATE, DELETE, DROP ON performance_schema.* TO 'pmm'@'%';
```

Adding an Amazon RDS, Aurora or Remote Instance

The preferred method of adding an Amazon RDS database instance to PMM is via the *Configuration* → *PMM Inventory* → ⊕ *Add Instance* menu option.

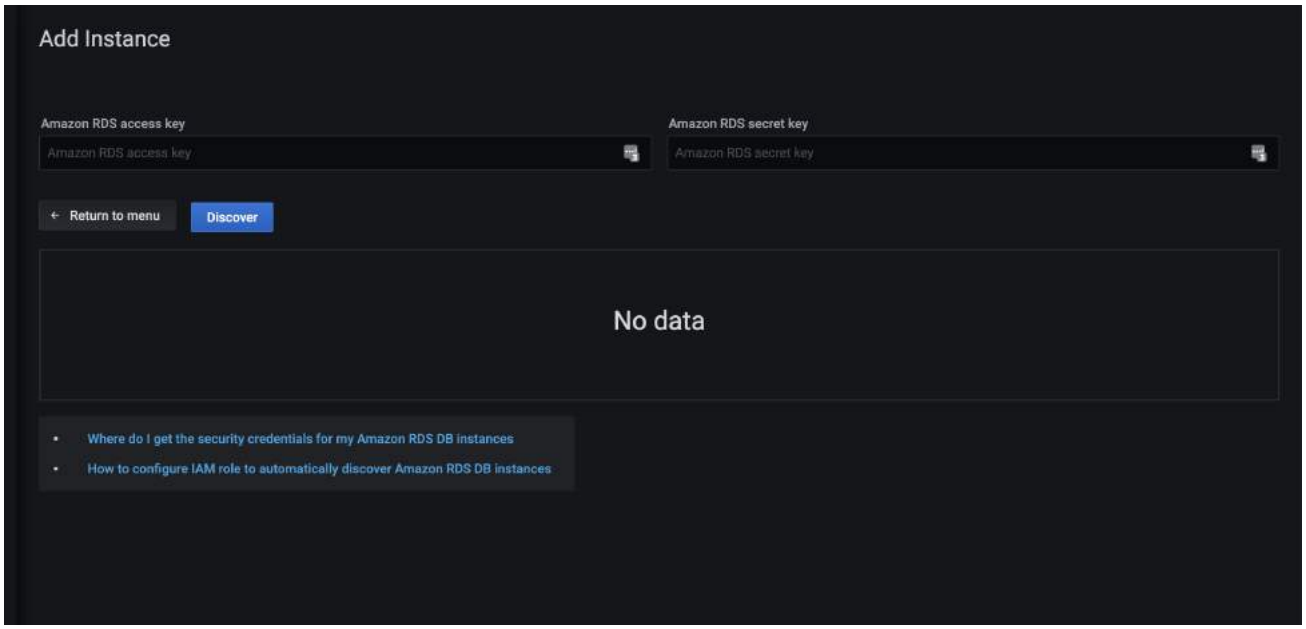
> **Important**
>
> It may take longer for PMM to discover Amazon RDS instances in the `creating` state. You must wait a bit longer until PMM discovers these instances.

This method supports Amazon RDS database instances that use Amazon Aurora, MySQL, or MariaDB engines, as well as any remote PostgreSQL, ProxySQL, MySQL and MongoDB instances.

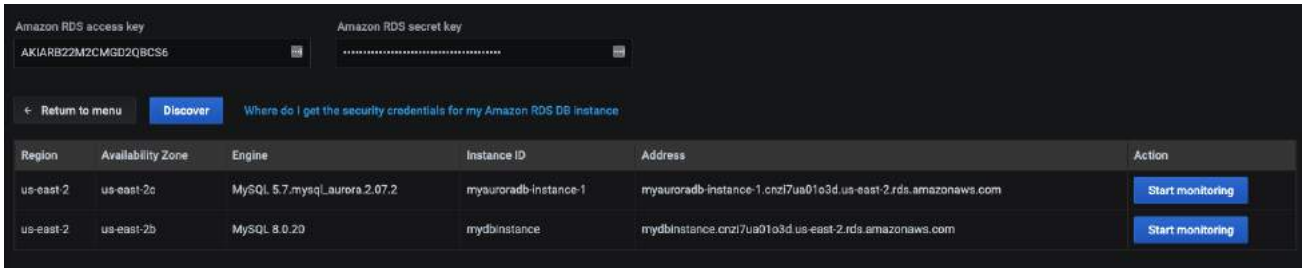
The following steps are needed to add an Amazon RDS database instance to PMM:

1. In the PMM web interface, go to *Configuration* → *PMM Inventory* → ⊕ *Add Instance*.
2. Select *Amazon RDS – Add a remote instance*.

Percona LLC and/or its affiliates ©2024

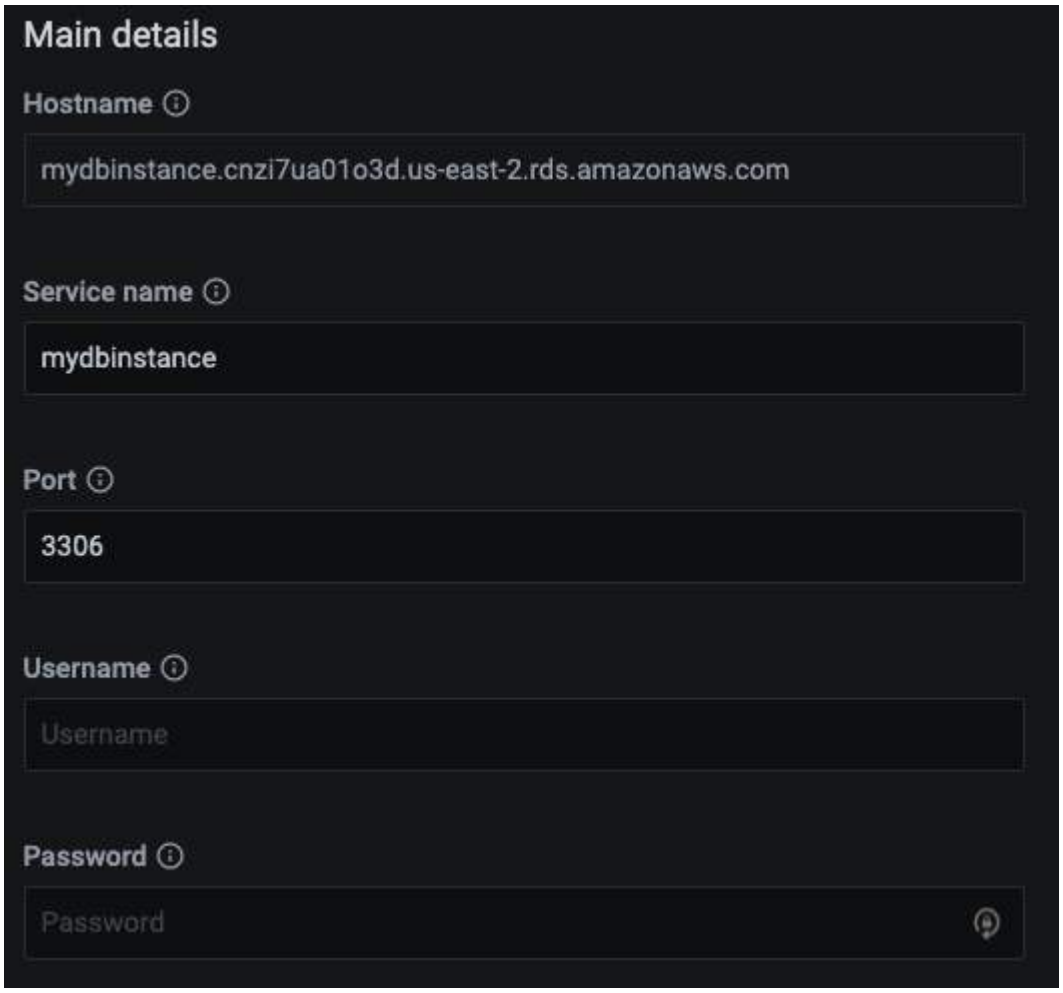


3. Enter the access key ID and the secret access key of your IAM user or leave these fields empty if an IAM role was created.
4. Click the *Discover* button for PMM to retrieve the available Amazon RDS instances.



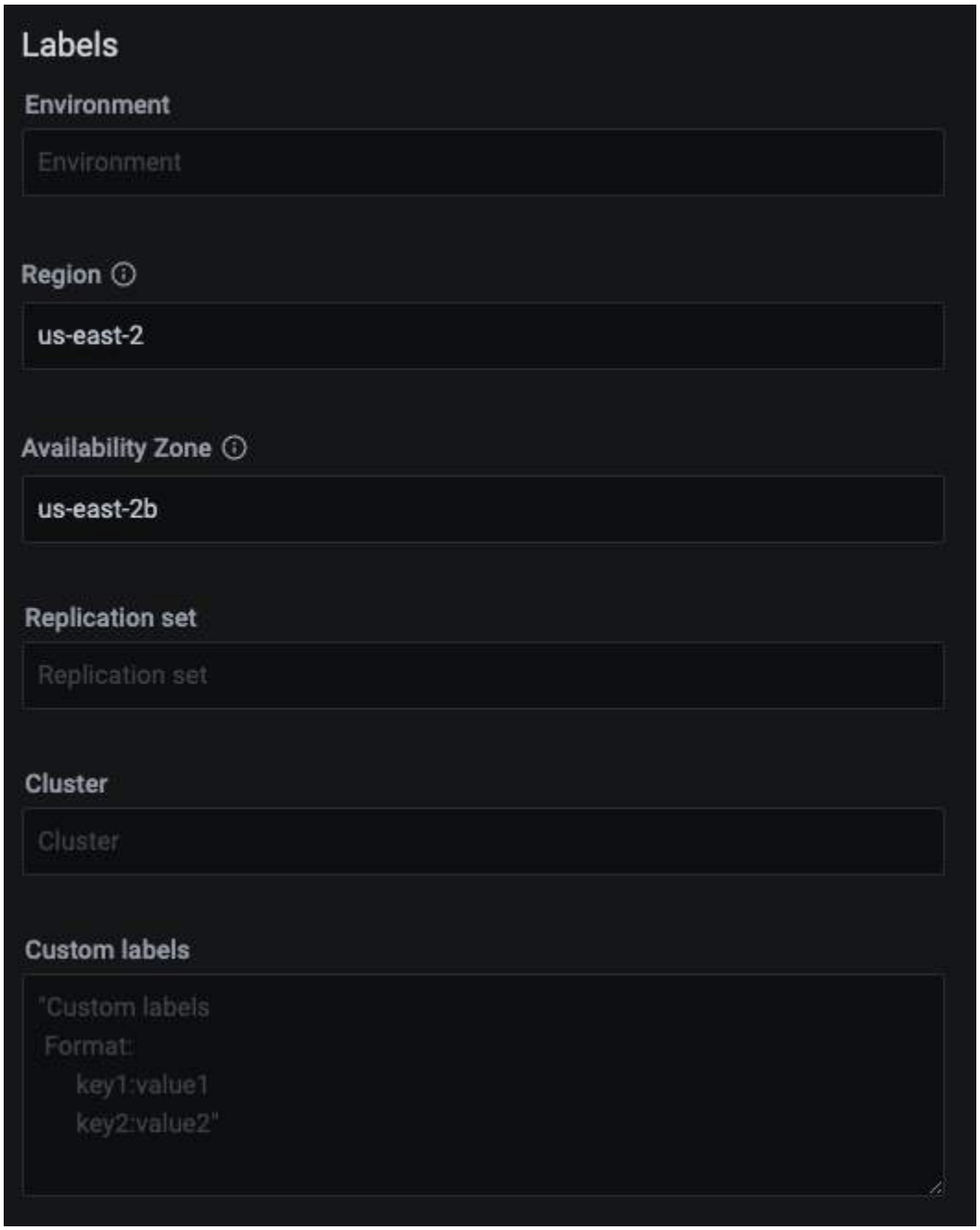
5. For the instance that you would like to monitor, select the *Start monitoring* button.
6. You will see a new page with the number of fields. The list is divided into the following groups: *Main details*, *RDS database*, *Labels*, and *Additional options*. Some already known data, such as already entered *AWS access key*, are filled in automatically, and some fields are optional.

Percona LLC and/or its affiliates ©2024



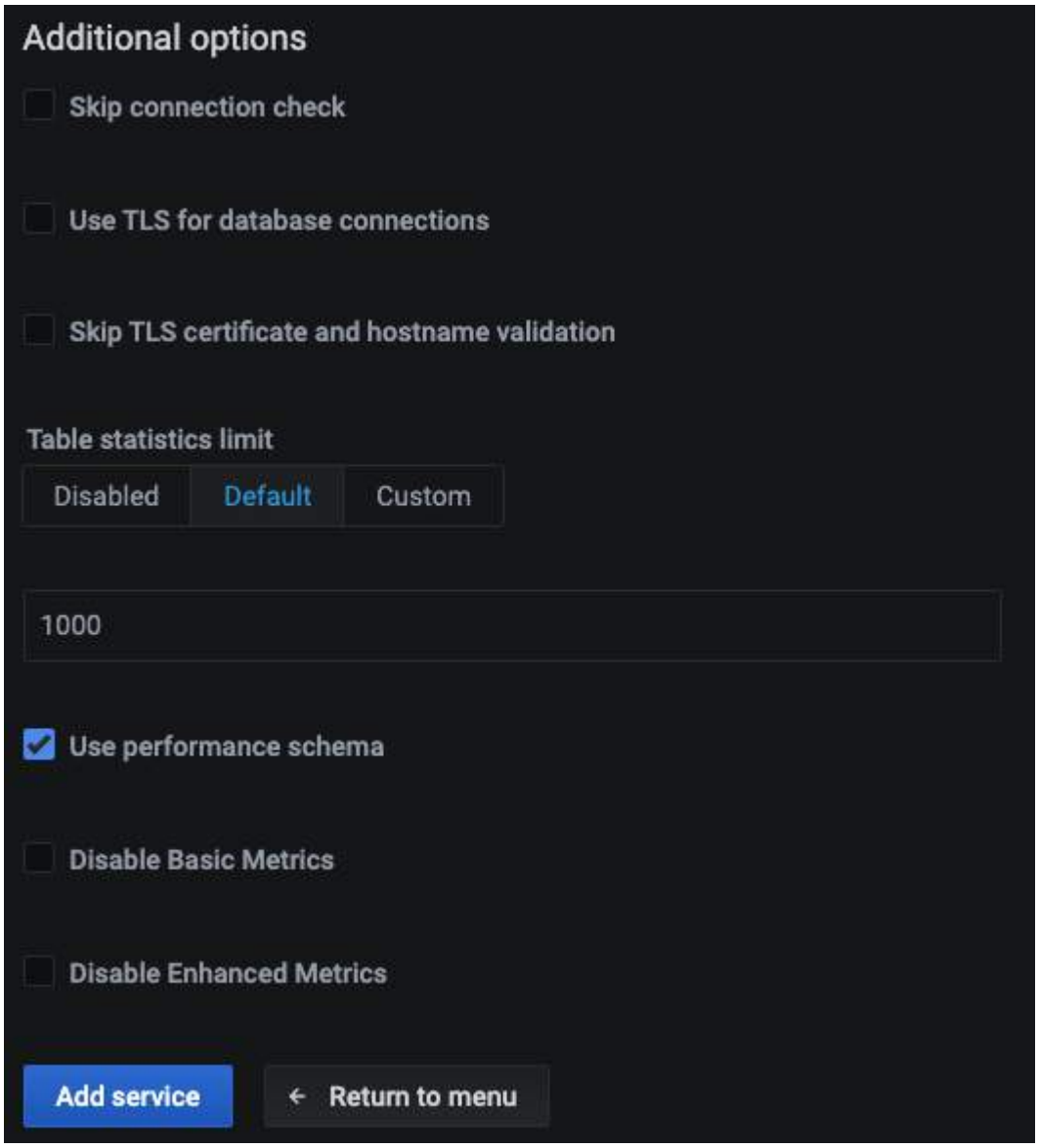
The *Main details* section allows you to specify the DNS hostname of your instance, the service name to use within PMM, the port your service is listening on, and the database user name and password.

Percona LLC and/or its affiliates ©2024



The *Labels* section allows you to specify labels for the environment, the AWS region and availability zone to be used, the Replication set and Cluster names and also it allows you to set the list of custom labels in a key:value format.

Percona LLC and/or its affiliates ©2024



The *Additional options* section contains specific flags which allow you to tune the RDS monitoring. They can allow you to skip connection check, to use TLS for the database connection, not to validate the TLS certificate and the hostname, as well as to disable basic and/or enhanced metrics collection for the RDS instance to reduce costs.

Also this section contains a database-specific flag, which would allow Query Analytics for the selected remote database:

- when adding some remote MySQL, AWS RDS MySQL or Aurora MySQL instance, you will be able to choose using performance schema for the database monitoring;
- when adding a PostgreSQL instance, you will be able to activate using `pg_stat_statements` extension;
- when adding a MongoDB instance, you will be able to choose using Query Analytics MongoDB profiler.

7. Finally press the *Add service* button to start monitoring your instance.

Adding an Amazon RDS PostgreSQL instance

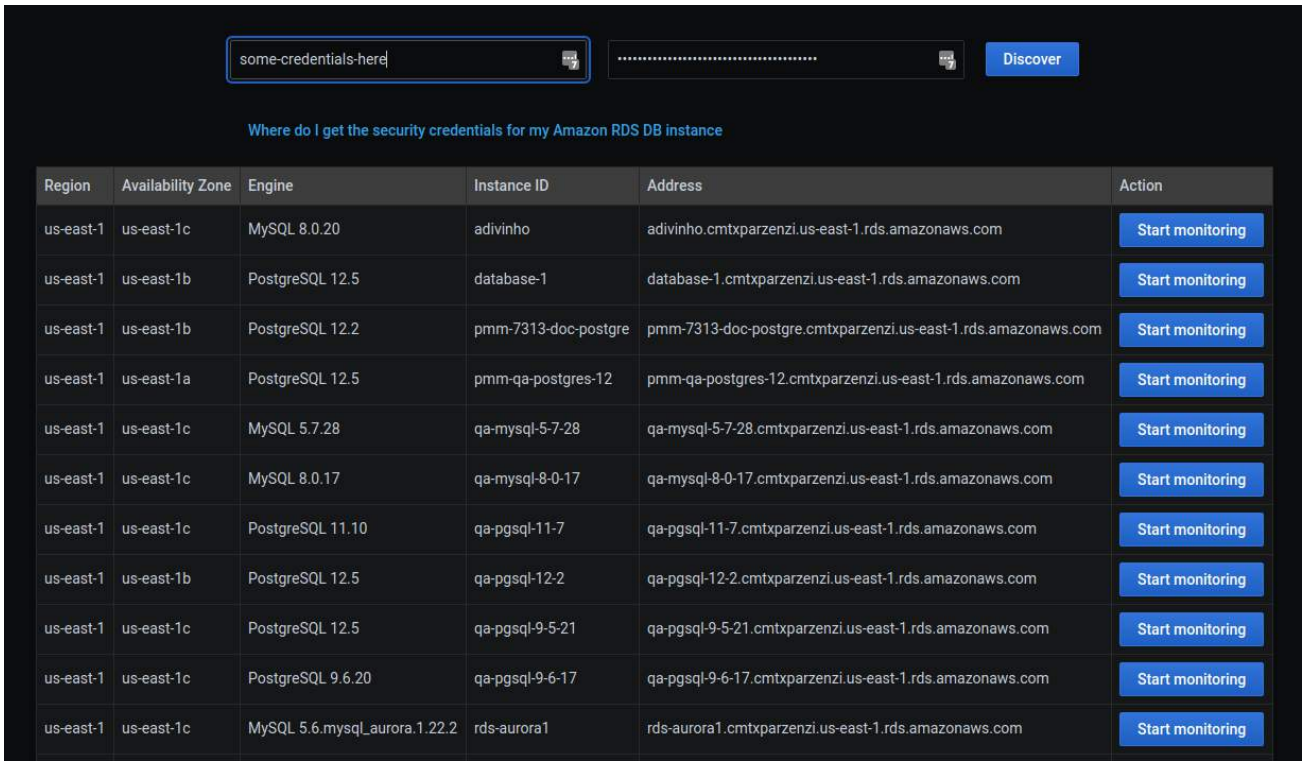
For PostgreSQL, use the same method described above.

1. In the PMM web interface, go to *Configuration* → *PMM Inventory* → ⊕ *Add Instance*..

Percona LLC and/or its affiliates ©2024

2. Select *Amazon RDS – Add a remote instance*.

   At the moment of writing this guide, the Add button doesn't mention PostgreSQL but the discovery function already supports it.

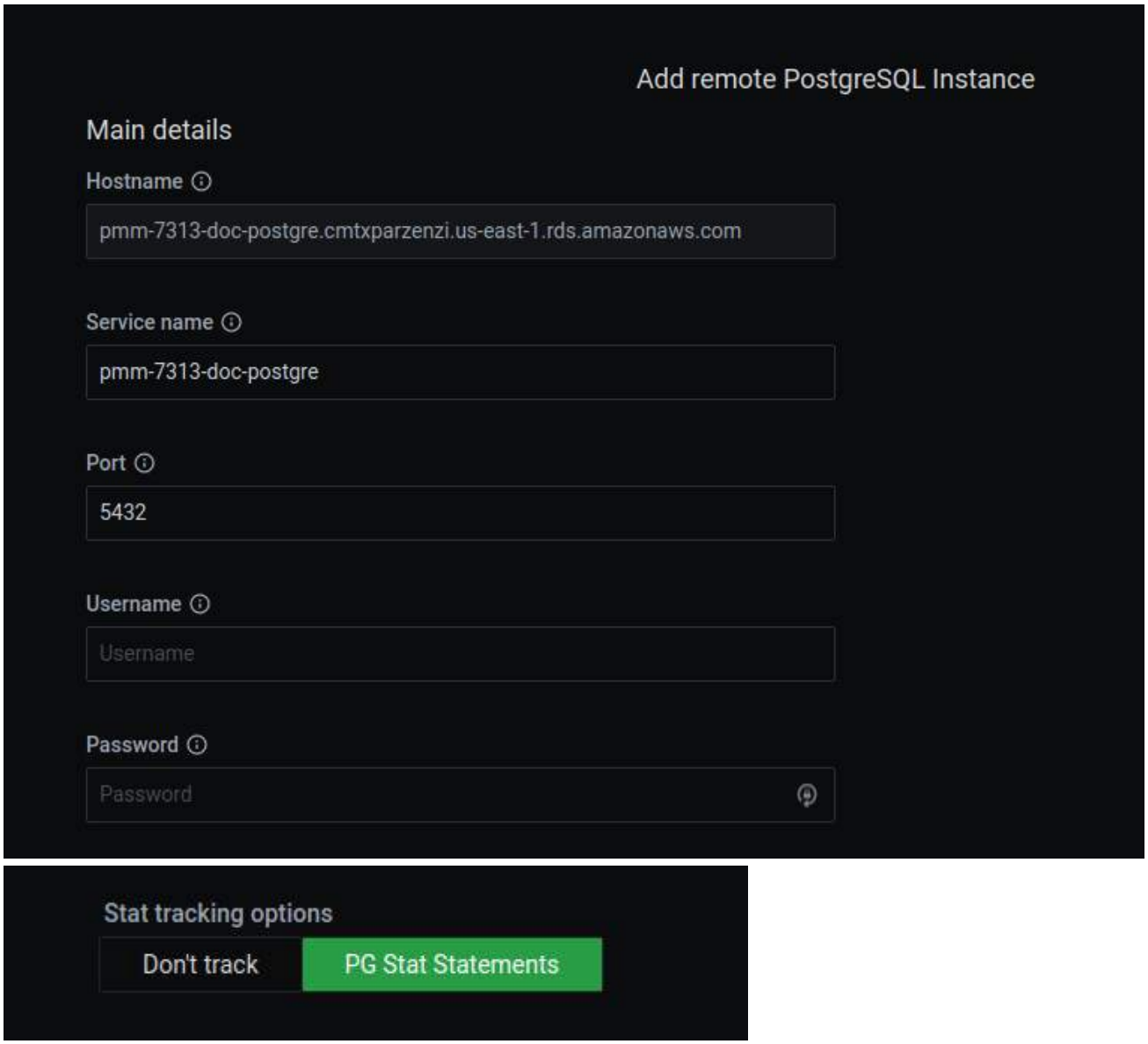


3. Follow steps 4 to 6 as in the previous section. Fill the form and remember to select `PG Stat Statement` to enable Query Analytics.

   To get queries for Query Analytics, you need to enable `pg_stat_statements` in `postgres` database of your instance by running:

```
CREATE EXTENSION pg_stat_statements SCHEMA public;
```


Percona LLC and/or its affiliates ©2024



Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

3.3.7 Microsoft Azure

> **Caution**
>
> Microsoft Azure functionality is currently in technical preview and is subject to change.

Percona LLC and/or its affiliates ©2024

Activate Microsoft Azure

The Microsoft Azure feature is turned off by default. To turn it on:

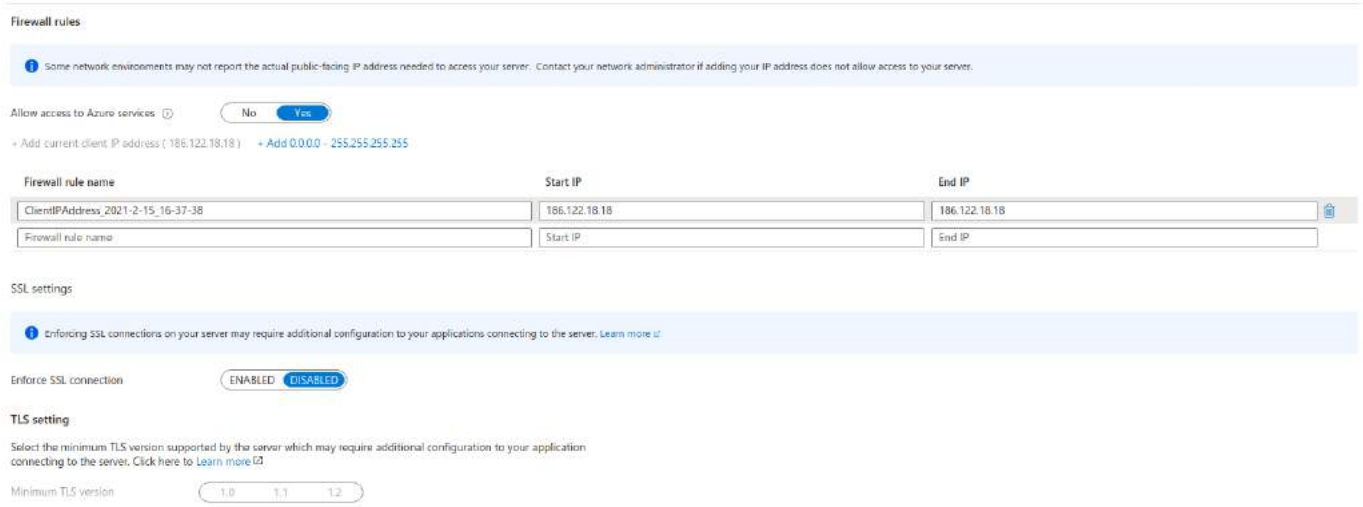
1. Go to *Configuration* → *Settings* → *Advanced Settings*.
2. Click the toggle in the *Technical preview features* section of the page.

Required settings

It is possible to use PMM for monitoring Azure database instances like other remote instances. In this case, the PMM Client is not installed on the host where the database server is deployed. By using the PMM web interface, you connect to the Azure DB instance. Discovery is not yet implemented in PMM but it is possible to add known instances by providing the connection parameters.

First of all, ensure that there is the minimal latency between PMM Server and the Azure instance.

Second, add a firewall rule to enable access from PMM Client like this:



Setting up a MySQL instance

Query Analytics requires you to configure *Performance Schema* as the query source, because the slow query log is stored on the Azure side, and QAN agent is not able to read it. Enable the `performance_schema` option under `Parameter Groups` in Azure MySQL databases.

When adding a monitoring instance for Azure, specify a unique name to distinguish it from the local MySQL instance. If you do not specify a name, it will use the client's host name.

Create the `pmm` user with the following privileges on the Azure MySQL database instance that you want to monitor:

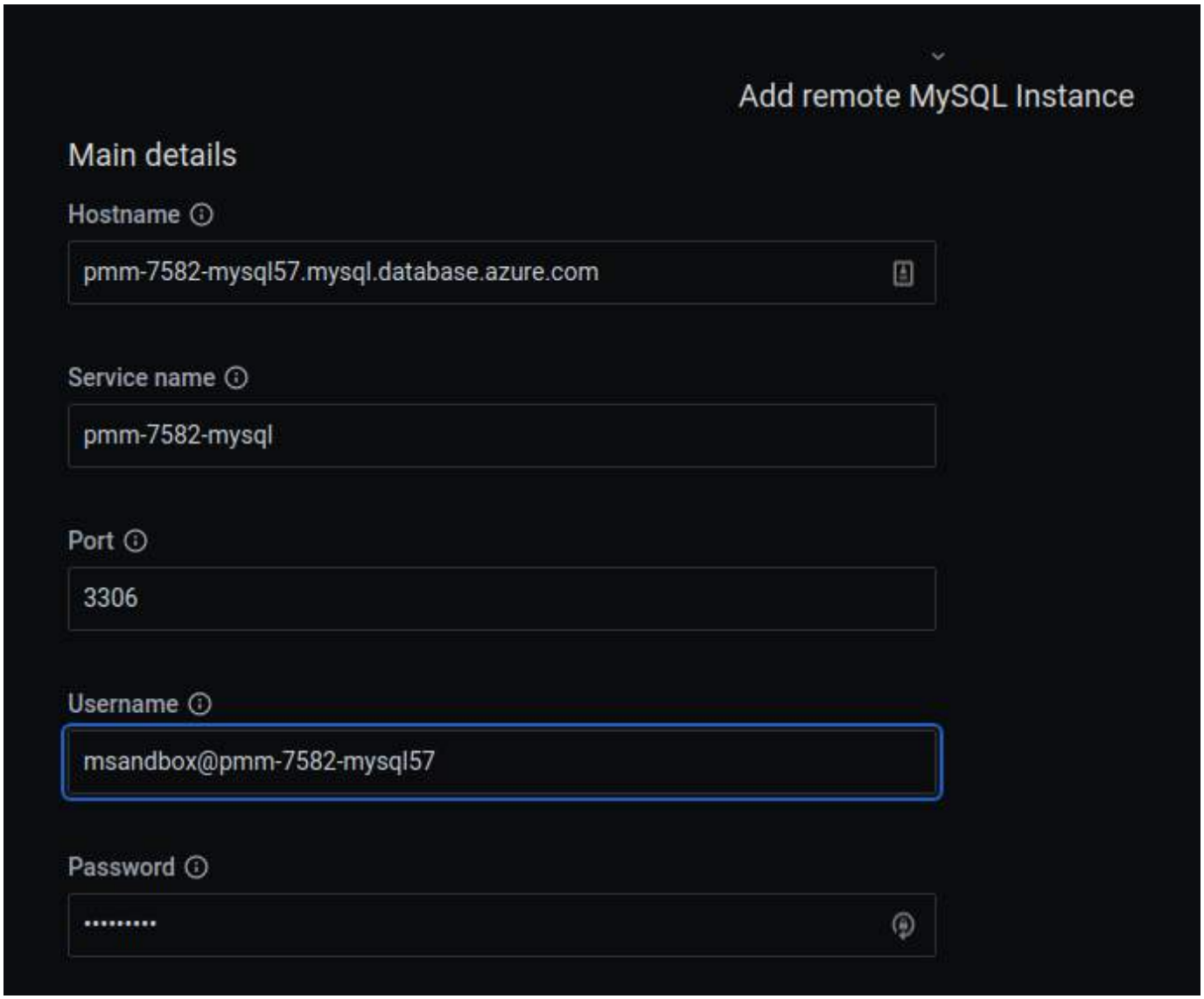
```
CREATE USER 'pmm'@'%' IDENTIFIED BY 'pass';
GRANT SELECT, PROCESS, REPLICATION CLIENT ON *.* TO 'pmm'@'%';
ALTER USER 'pmm'@'%' WITH MAX_USER_CONNECTIONS 10;
```

Adding an Azure Instance

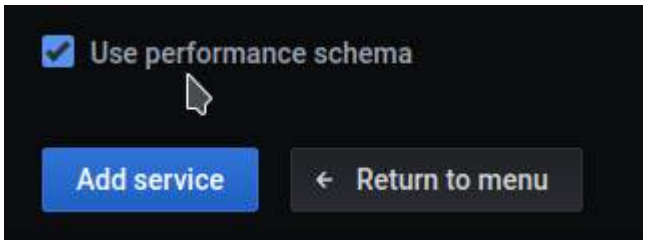
Follow the instructions for remotes instances explained here, Azure MySQL databases are similar to AWS RDS databases.

Example:

Percona LLC and/or its affiliates ©2024



and be sure to set *Performance Schema* as the query collection method for Query Analytics.



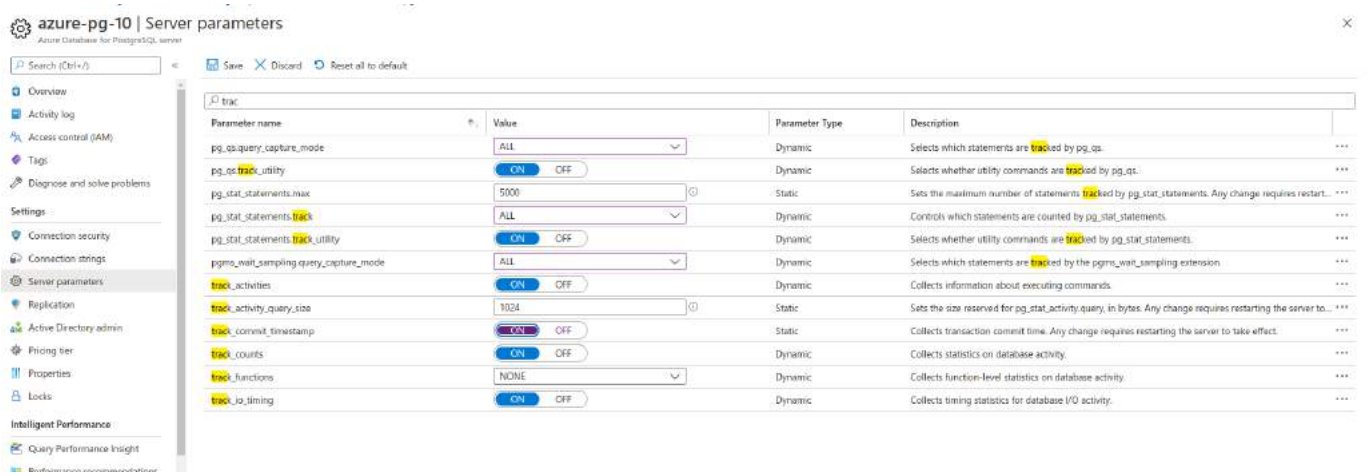
MARIADB

MariaDB up to version 10.2 works out of the box but starting with MariaDB 10.3 instrumentation is disabled by default and cannot be enabled since there is no SUPER role in Azure-MariaDB. So, it is not possible to run the required queries to enable instrumentation. Monitoring will work but Query Analytics won't receive any query data.

POSTGRESQL

For PostgreSQL follow the same methods used for MySQL and MariaDB and enable `track_io_timing` in the instance configuration to enable Query Analytics.

Percona LLC and/or its affiliates ©2024



For Query Analytics, set the server parameter:

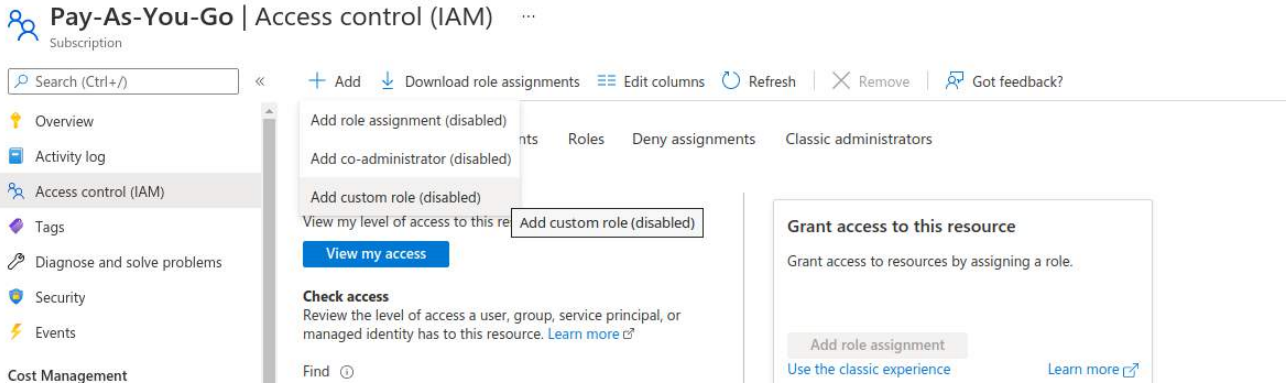
```
pg_stat_statements.track = all
```

To discover databases on Azure

You need to get the Client ID, Client Secret, Tenant ID and Subscription ID.

GET THE SUBSCRIPTION ID

1. Search **Subscriptions**, click on your subscription name

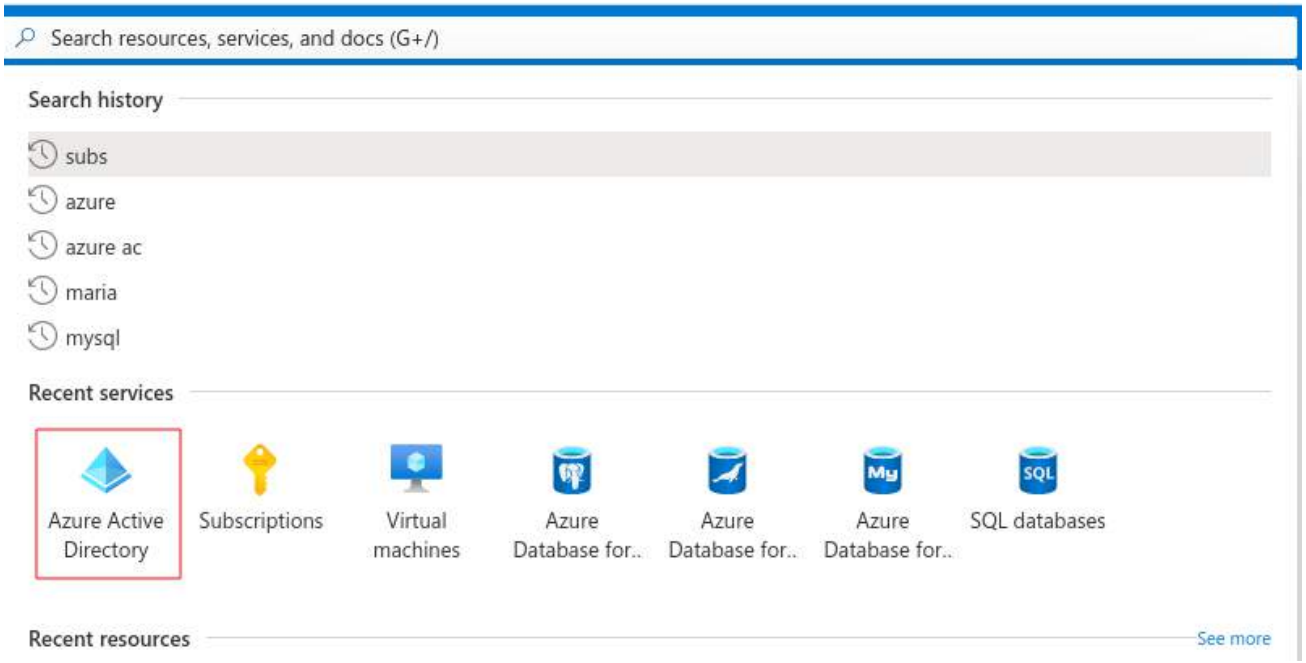


2. Copy the subscription ID

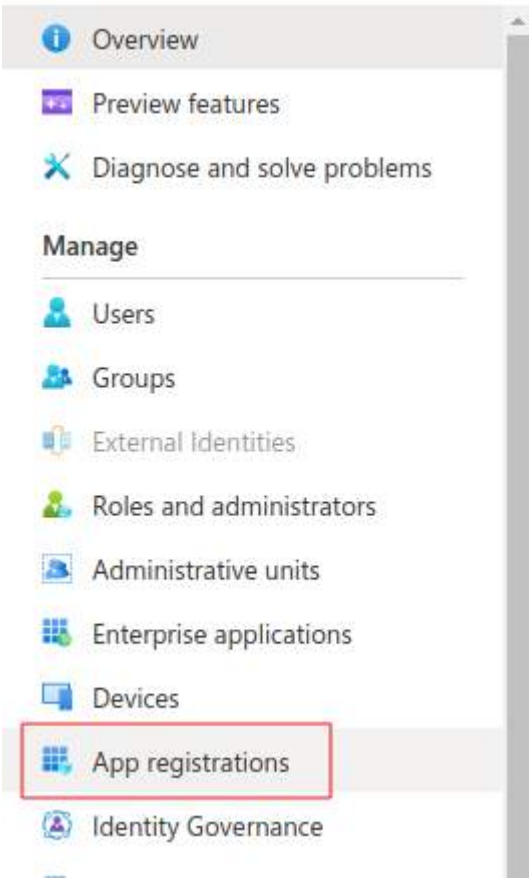
Percona LLC and/or its affiliates ©2024

CREATE A NEW APPLICATION TO GET THE TENANT ID, CLIENT ID AND THE CLIENT SECRET.

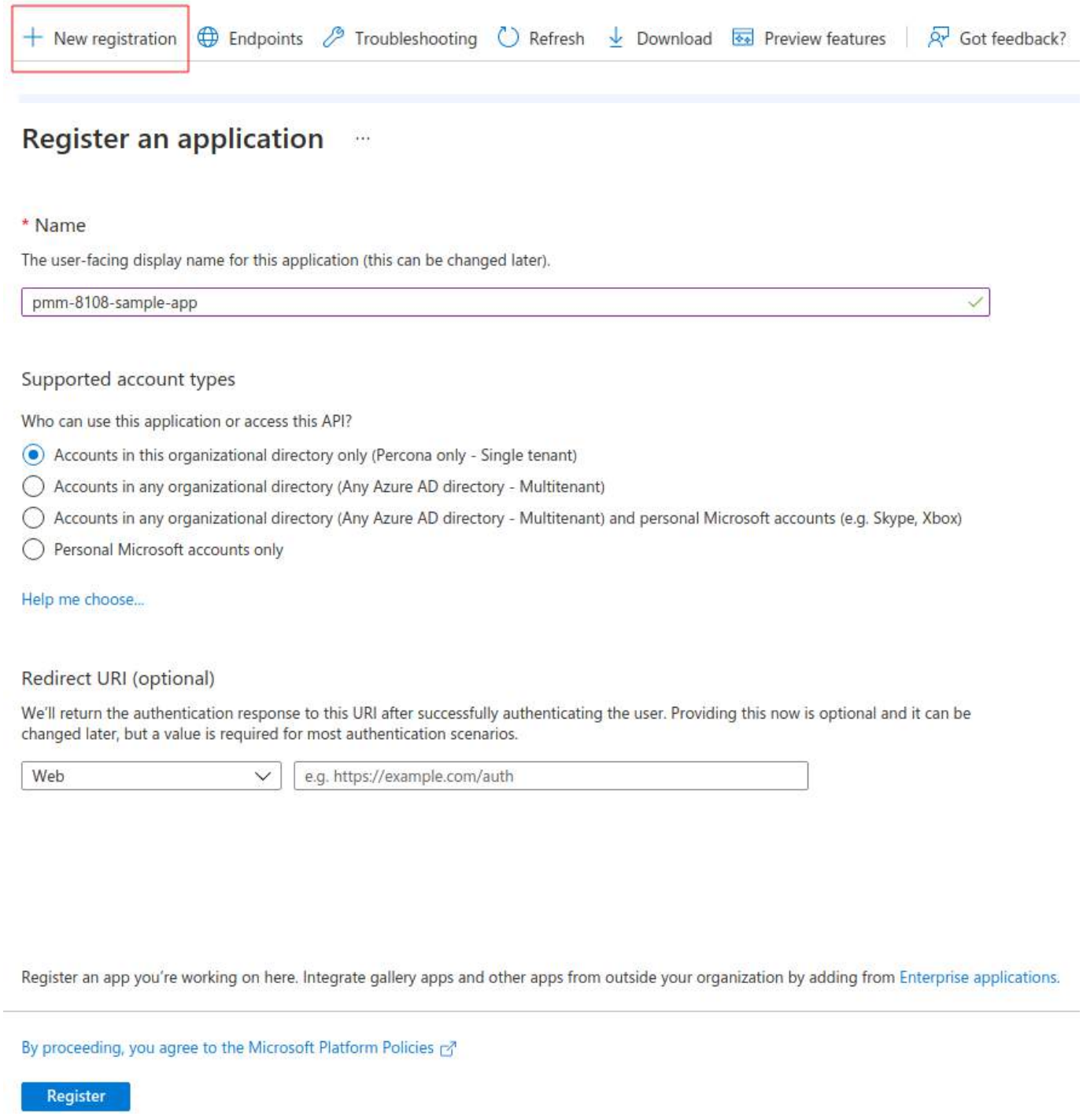
1. Search for **Azure Active Directory**



2. Register a new application:



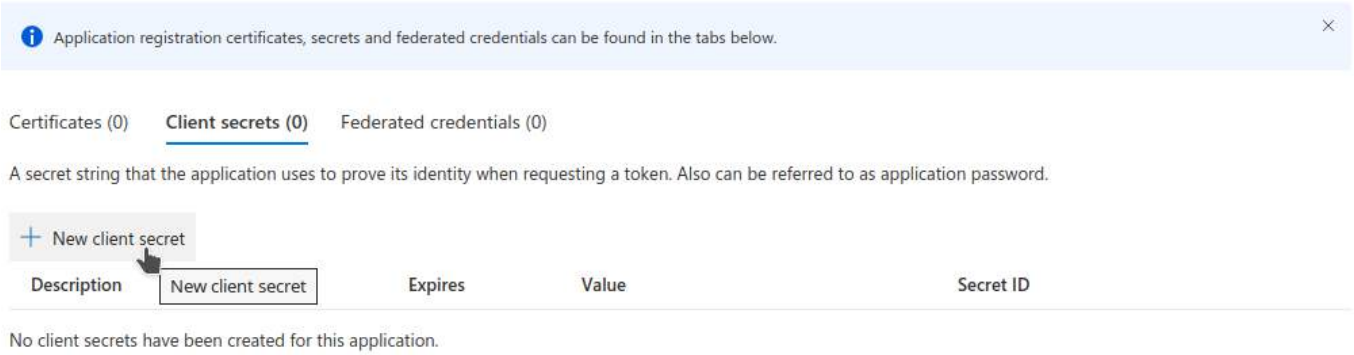
Percona LLC and/or its affiliates ©2024

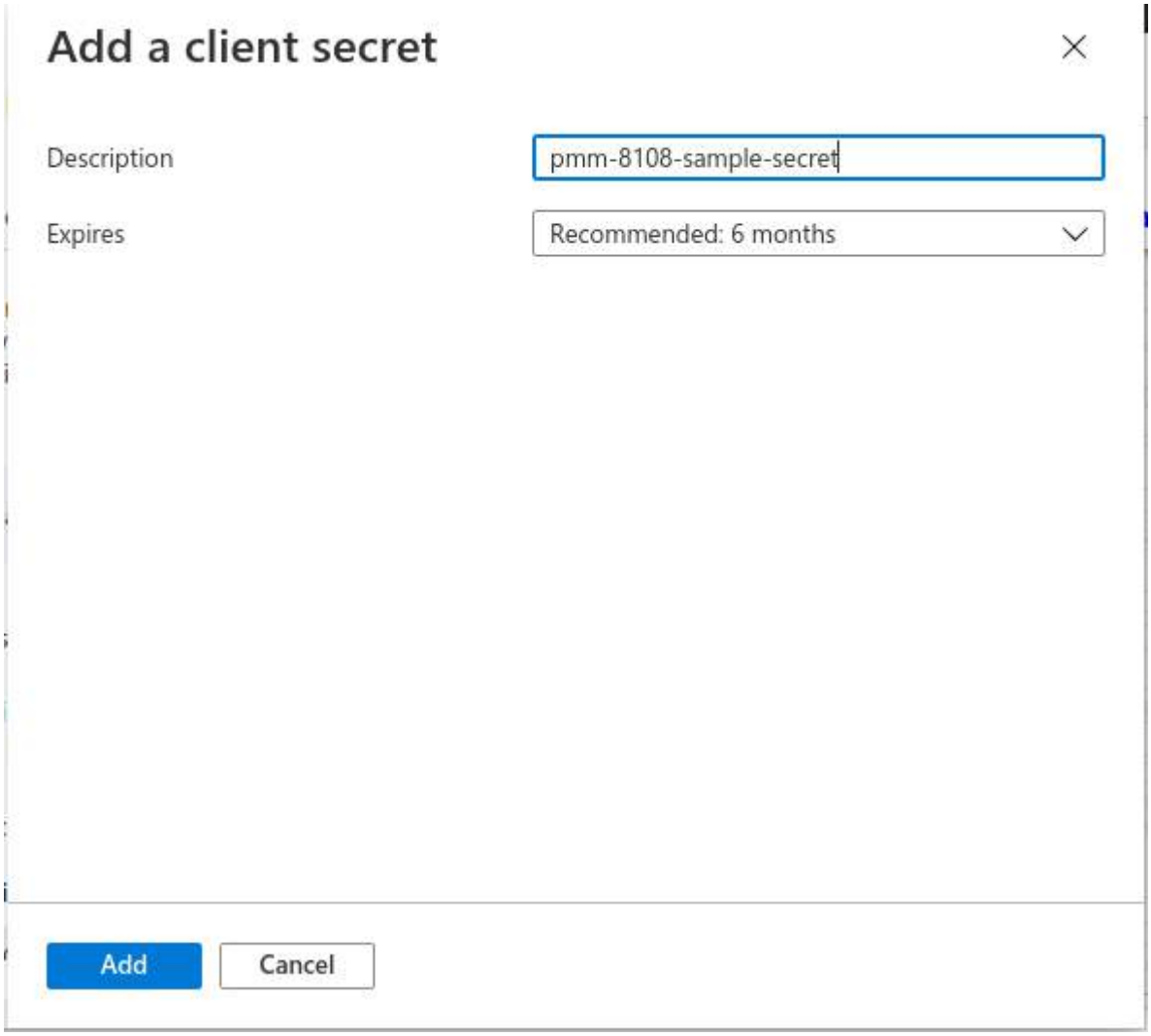


3. At this point you can copy the client and tenant IDs.

4. Create an application secret.

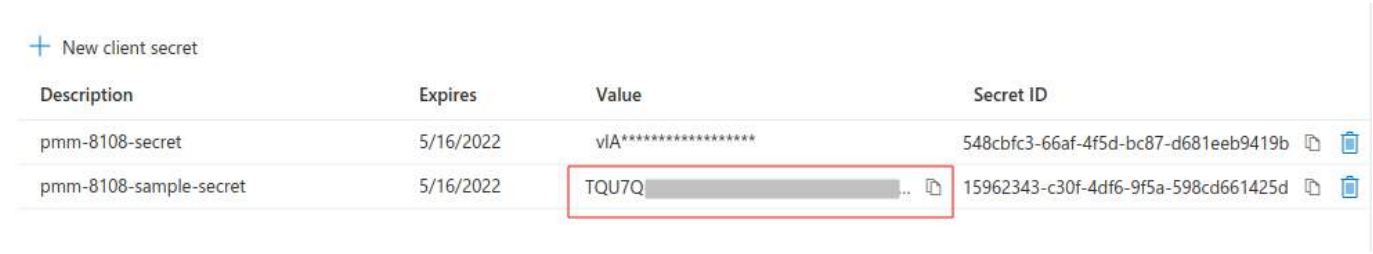
Percona LLC and/or its affiliates ©2024





5. Copy the **value** of the application secret. Once you leave this page you won't be able to see the secret again and you will have to generate a new one.

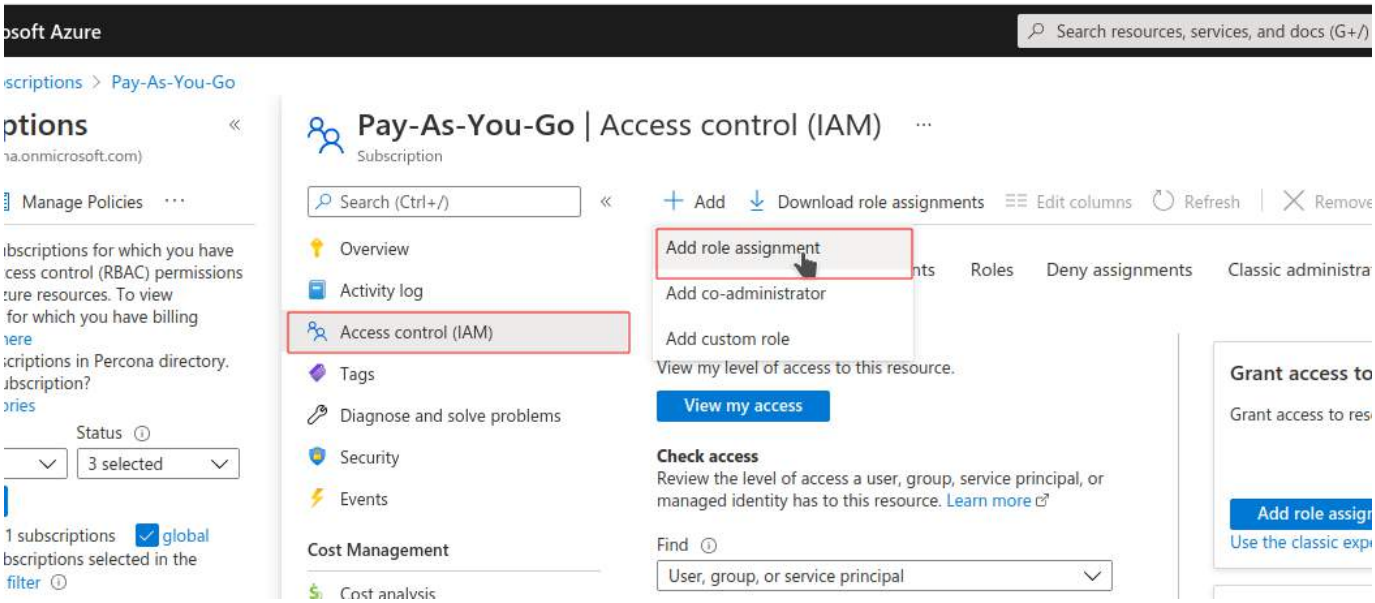
Percona LLC and/or its affiliates ©2024

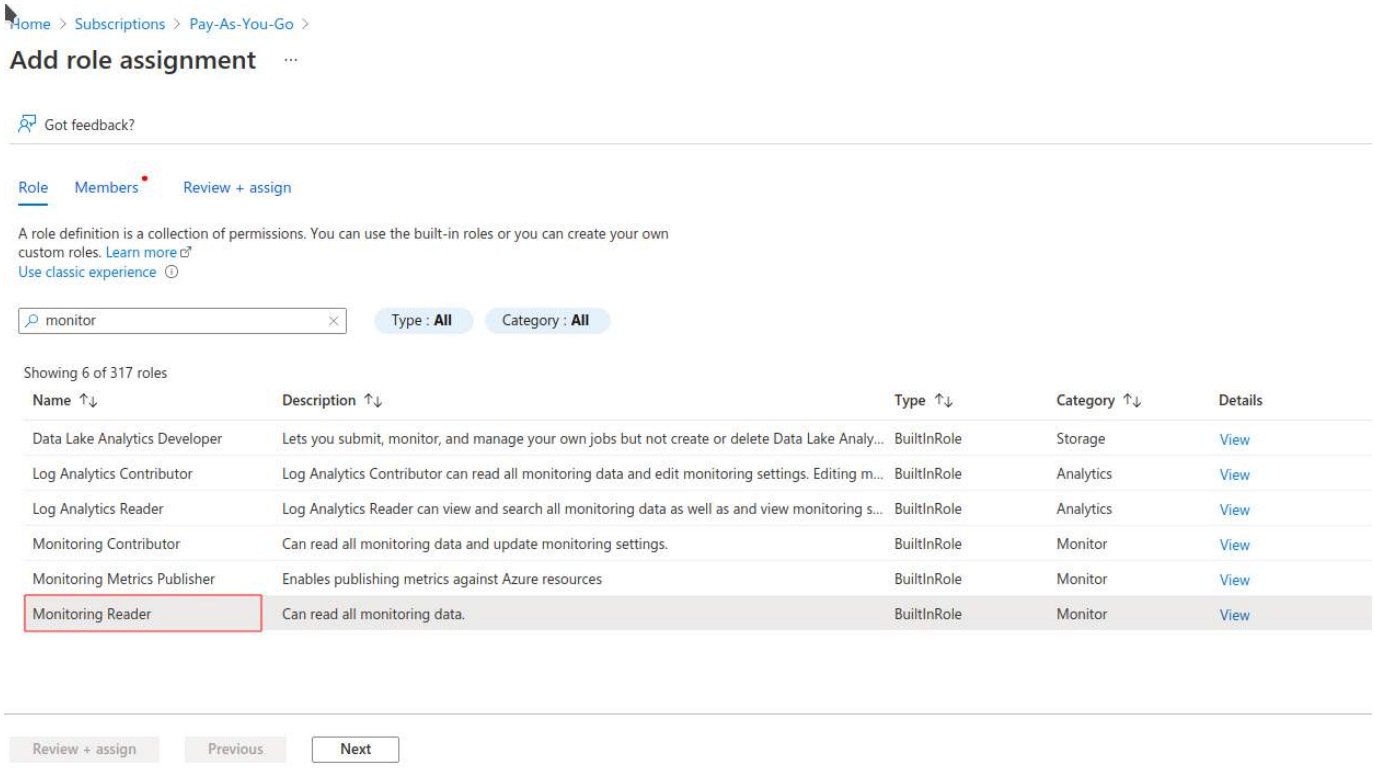


6. Give API access permissions to your application.

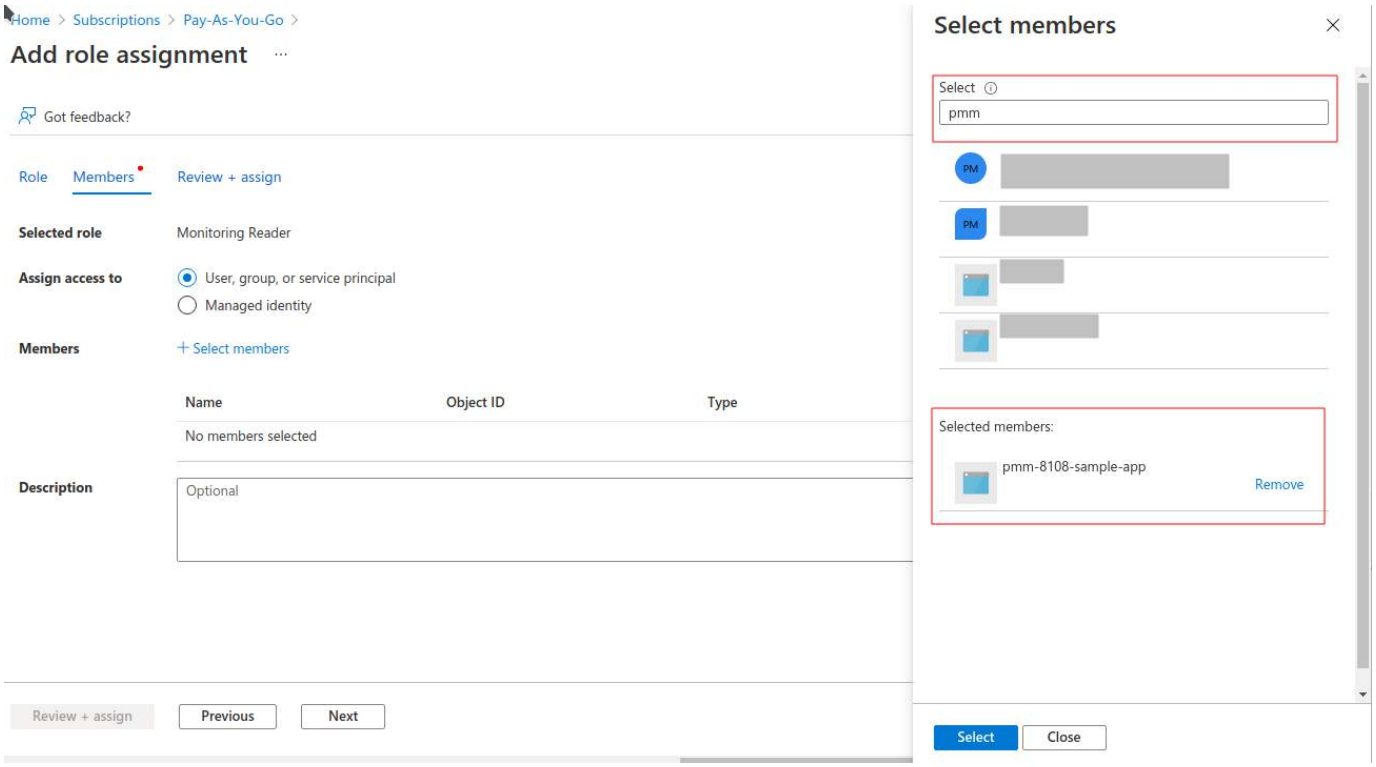
```
6.1. Search for **Subscriptions** like in step 1.

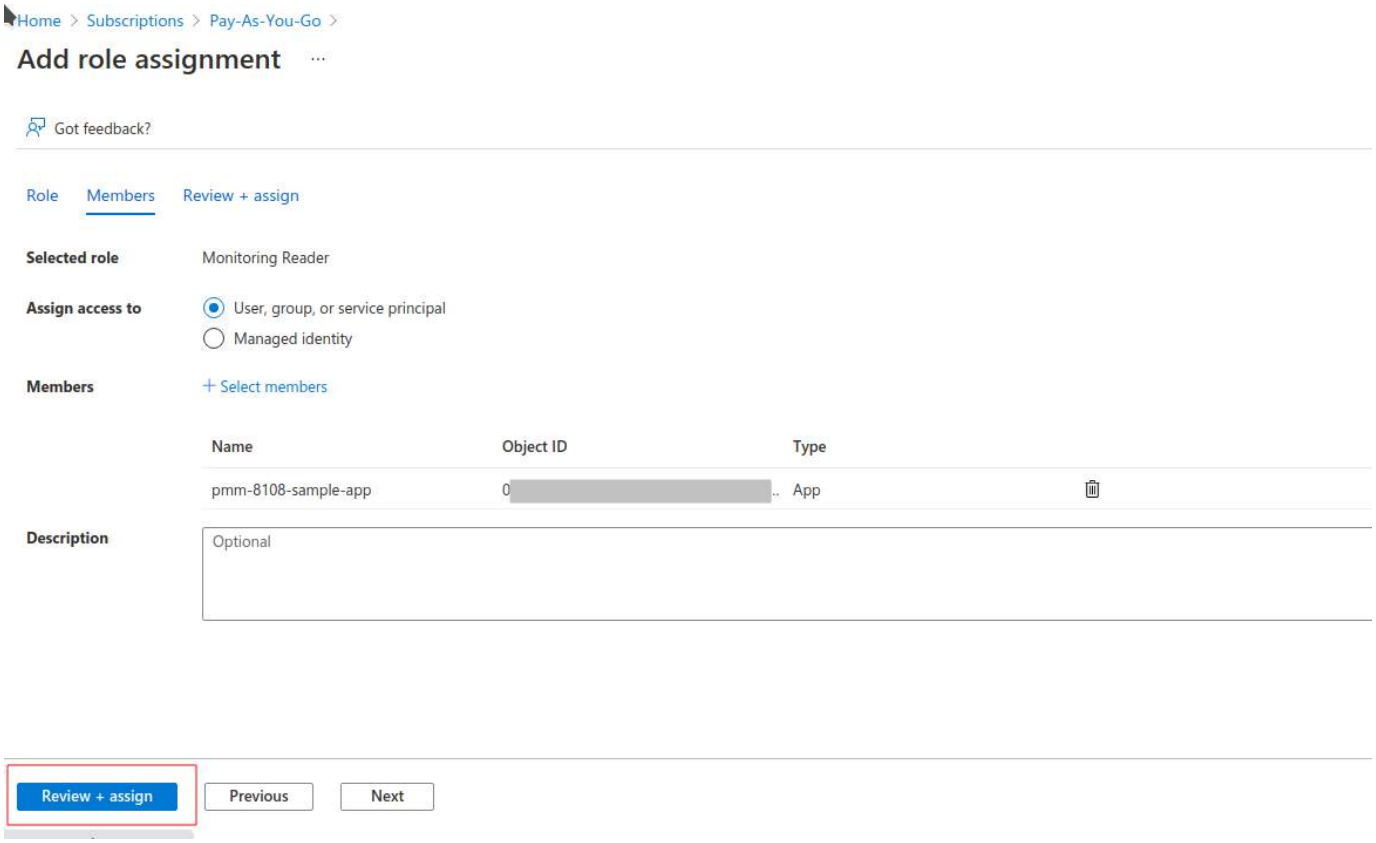
6.2. Select your application and grant **Monitor Reader** permissions. This might require you to have admin permissions in your Azure account.
```





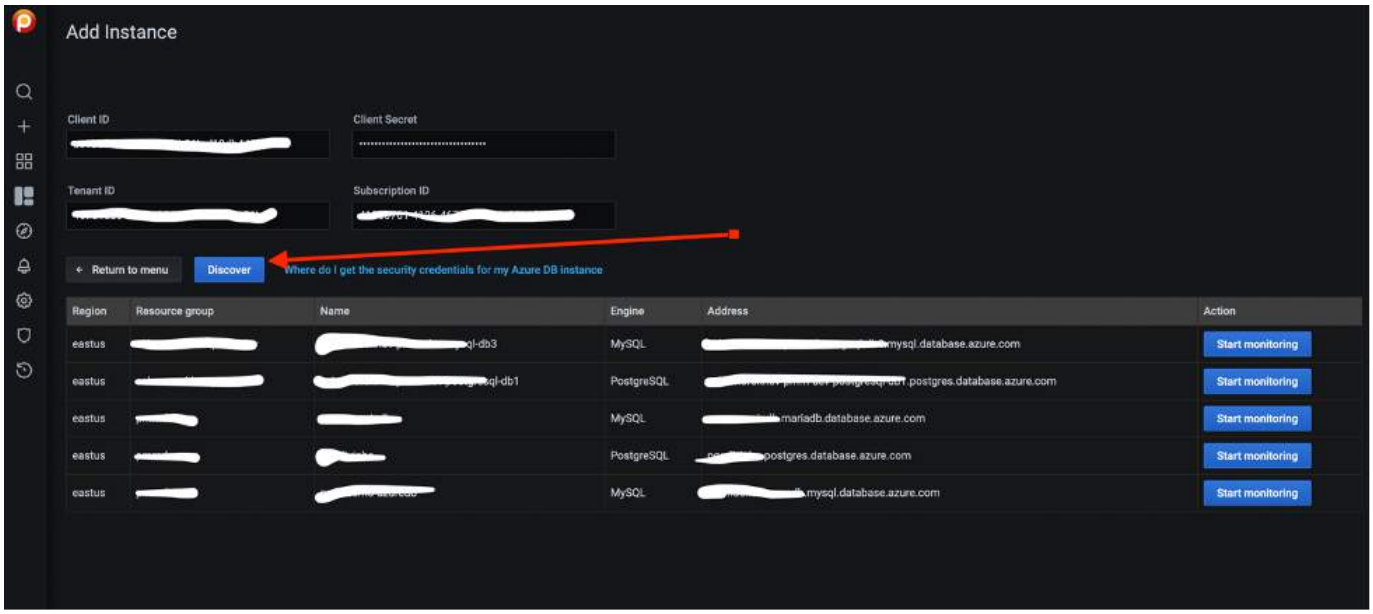
Percona LLC and/or its affiliates ©2024





When you fill in all fields press the *Discover* button and you will see a list of available databases for monitoring.

Percona LLC and/or its affiliates ©2024



You can monitor 6 types of databases:

- `Microsoft.DBforMySQL/servers`
- `Microsoft.DBforMySQL/flexibleServers`
- `Microsoft.DBforMariaDB/servers`
- `Microsoft.DBforPostgreSQL/servers`
- `Microsoft.DBforPostgreSQL/flexibleServers`
- `Microsoft.DBforPostgreSQL/serversv2`

You can find more details on how to create DB on Azure at:

- https://docs.microsoft.com/en-us/azure/postgresql/
- https://docs.microsoft.com/en-us/azure/mysql/

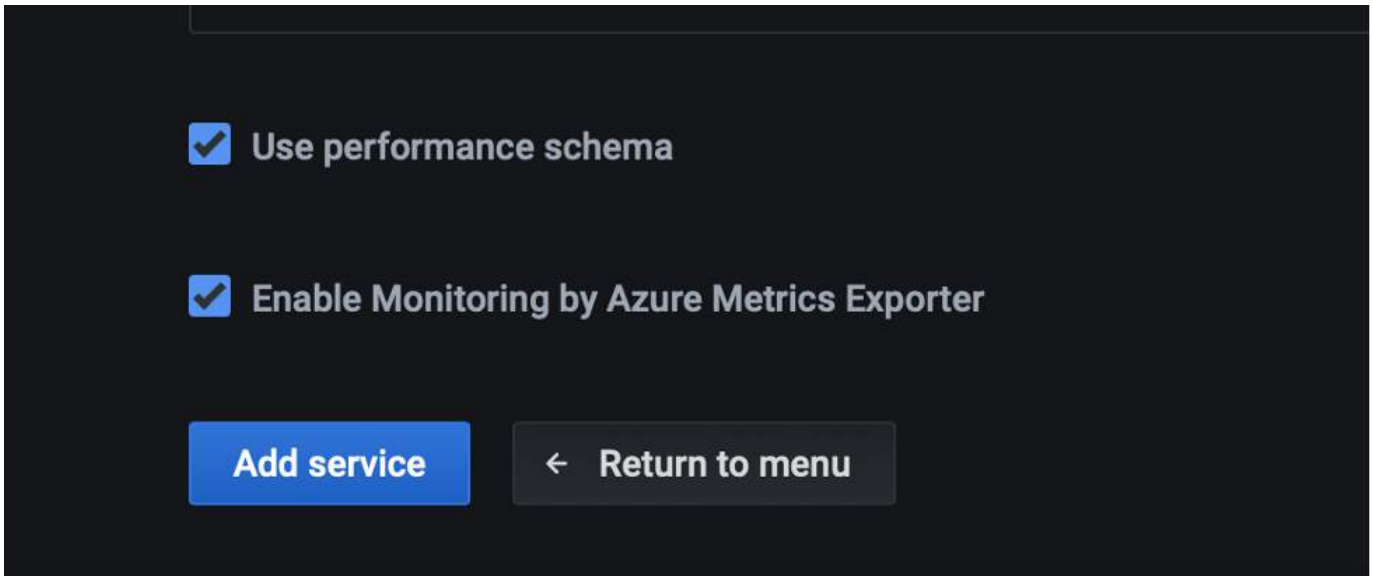
Tip

You must set `pg_stat_statements.track = all` in your PostgreSQL Server settings to use PMM Query Analytics. (Read more.)

In the list of databases on the Discovery page click *Start Monitoring* to add the selected Azure Database to PMM.

Fill in all required fields and click *Add service*.

Percona LLC and/or its affiliates ©2024



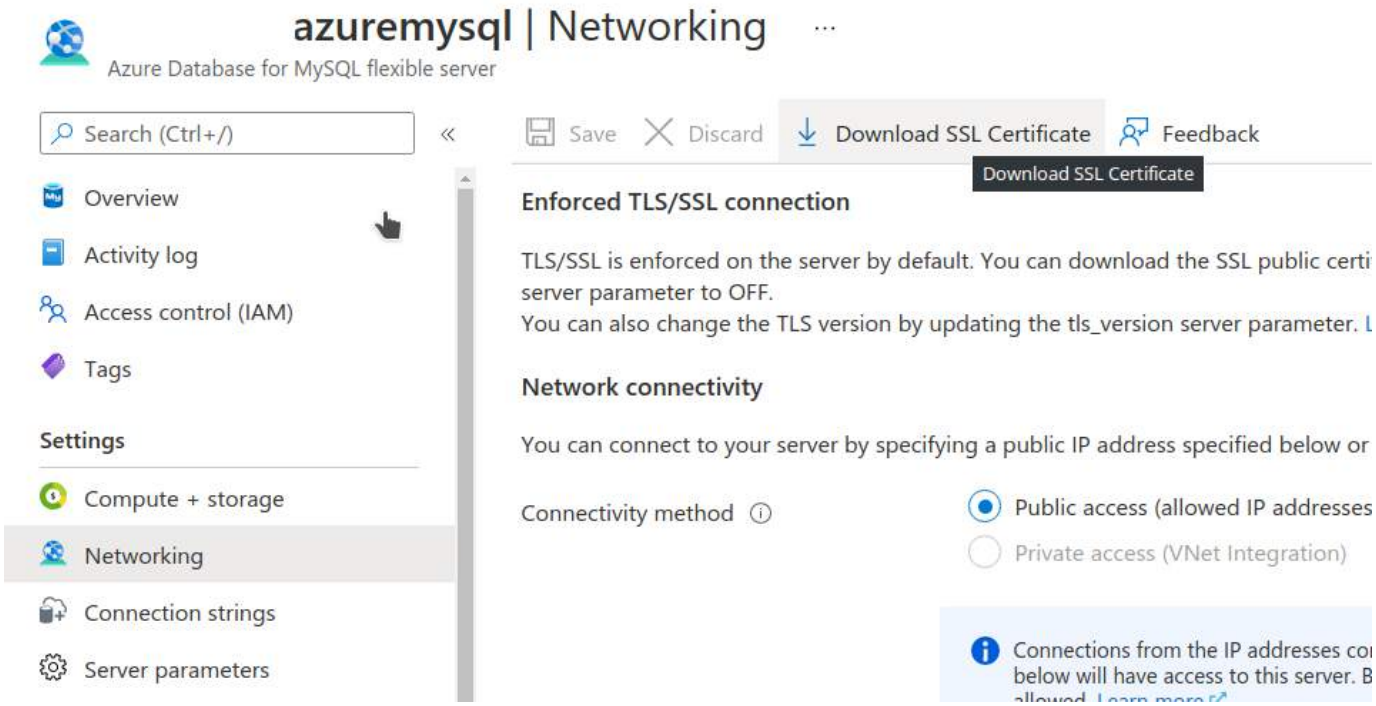
PMM can use 3 exporters to collect metrics:

- Azure Metrics Exporter – collect “system” metrics related to DB.
  - `node_cpu_average`
  - `azure_resource_info`
  - `node_filesystem_size_bytes`
  - `azure_memory_percent_average`
  - `azure_storage_percent_average`
  - `azure_storage_used_bytes_average`
  - `node_network_receive_bytes_total`
  - `node_network_transmit_bytes_total`
- `mysql_exporter` or `postgres_exporter` – to collect database related metrics.
- PMM Agent to collect queries related metrics using `pg_stat_statements` for PostgreSQL or Performance Schema for MySQL (MariaDB)

ADDING AN AZURE INSTANCE ON PMM-CLIENT SIDE

TLS/SSL is enforced on the server by default. So please download the certificate needed to communicate over SSL with your Azure Database. It can be done on Networking tab for your Azure Database instance.

Percona LLC and/or its affiliates ©2024



Also enforced TLS/SSL connection option can be disabled on server side.

Command for adding an azure database service for monitoring without TLS/SSL.

```
pmm-admin add mysql --username=azureuser --password=secure --host=azuremysql.mysql.database.azure.com --service-name=azure1 --query-source=perfschema
```

Downloaded certificate is named `DigiCertGlobalRootCA.crt.pem`.

An example of the command for adding an Azure database service for monitoring with TLS/SSL would be:

```
pmm-admin add mysql --username=azureuser --password=secure --host=azuremysql.mysql.database.azure.com --service-name=azure1 --query-source=perfschema --tls --
tls-ca=DigiCertGlobalRootCA.crt.pem --tls-cert=client-cert.pem --tls-key=client-key.pem --tls-skip-verify
```

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

3.3.8 Google Cloud Platform

PMM can monitor MySQL or PostgreSQL instances hosted on the Google Cloud Platform.

The connection can be direct, or indirect using Cloud SQL Proxy.

MySQL

1. Set up a MySQL instance on Google Cloud.

Percona LLC and/or its affiliates ©2024

2. The database server must be accessible by PMM Client. If PMM Client is not also hosted on GCP, you will need to add a network interface with a public interface.
3. Configure *Performance Schema* on the MySQL server. Using the GCP console's *Cloud Shell* or your own `gcloud` installation, run:

```
gcloud sql instances patch <instance_name> --database-flags performance_schema=on
```

4. Log into the PMM user interface.
5. Select *Configuration* → *PMM Inventory* → ⊕ *Add Instance*.
6. Click *MySQL Add a remote instance*.
7. Fill in the details for the remote MySQL instance.
   - Ensure *Use performance schema* is selected.
8. Click *Add service*.
9. Check for values in the *MySQL Instance Overview* dashboard and in *Query Analytics*.

PostgreSQL

1. Set up a PostgreSQL instance on Google Cloud.
2. The database server must be accessible by PMM Client. If PMM Client is not also hosted on GCP, you will need to add a network interface with a public interface.
3. Configure `pg_stat_statements`. Open an interactive SQL session with your GCP PostgreSQL server and run:

```
CREATE EXTENSION pg_stat_statements;
```

4. Log into the PMM user interface.
5. Select *Configuration* → *PMM Inventory* → ⊕ *Add Instance*.
6. Select *PostgreSQL Add a remote instance*.
7. Fill in the details for the remote PostgreSQL instance:
   - In *Stat tracking options*, select *PG Stat Statements*.
8. Click *Add service*.
9. Check for values in the *PostgreSQL Instance Overview* dashboard and *Query Analytics*.

Cloud SQL Proxy

MYSQL

1. Create instance on GCP.
2. Note connection as `<project_id>:<zone>:<db_instance_name>`.
3. Enable Admin API and download the JSON credential file.
4. Enable *Performance Schema*.

Percona LLC and/or its affiliates ©2024

5. Run Cloud SQL Proxy (runs on PMM Client node).

    - As a Docker container:

    ```
    docker run -d \
    -v ~/path/to/admin-api-file.json:/config \
    -p 127.0.0.1:3306:3306 \
    gcr.io/cloudsql-docker/gce-proxy:1.19.1 \
    /cloud_sql_proxy \
    -instances=example-project-NNNN:us-central1:mysql-for-pmm=tcp:0.0.0.0:3306 \
    -credential_file=/config
    ```

    - On Linux:

    ```
    wget https://dl.google.com/cloudsql/cloud_sql_proxy.linux.amd64 -O cloud_sql_proxy
    chmod +x cloud_sql_proxy
    ./cloud_sql_proxy -instances=example-project-NNNN:us-central1:mysql-for-pmm=tcp:3306 \
    -credential_file=/path/to/credential-file.json
    ```

6. Add instance.

    ```
    pmm-admin add mysql --host=127.0.0.1 --port=3306 \
    --username=root --password=secret \
    --service-name=MySQLGCP --query-source=perfschema
    ```

POSTGRESQL

1. Create instance on GCP.
2. Note connection as `<project_id>:<zone>:<db_instance_name>`.
3. Enable Admin API and download the JSON credential file.
4. Run Cloud SQL Proxy.

    ```
    ./cloud_sql_proxy -instances=example-project-NNNN:us-central1:pg-for-pmm=tcp:5432 \
    -credential_file=/path/to/credential-file.json
    ```

5. Log into PostgreSQL.
6. Load extension:

    ```
    CREATE EXTENSION pg_stat_statements;
    ```

7. Add service:

    ```
    pmm-admin add postgresql --host=127.0.0.1 --port=5432 \
    --username="postgres" --password=secret --service-name=PGGCP
    ```

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

Percona LLC and/or its affiliates ©2024

3.3.9 Linux

Adding general system metrics service

PMM collects Linux metrics automatically starting from the moment when you have configured your node for monitoring with `pmm-admin config`.

> **Get expert help**
> If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.
>
> Community Forum Get a Percona Expert

2020-12-10

3.3.10 External Services

Adding general external services

You can collect metrics from an external (custom) exporter on a node when:

- there is already a PMM Agent instance running and,
- this node has been configured using the `pmm-admin config` command.

USAGE

```
pmm-admin add external --service-name=<service-name> --listen-port=<listen-port> --metrics-path=<metrics-path> --scheme=<scheme>
```

```
pmm-admin add external-serverless --external-name=<external-service-name> --host=<hostname> --listen-port=<listen-port> --metrics-path=<metrics-path> --
scheme=<scheme>
```

Getting data from external exporters

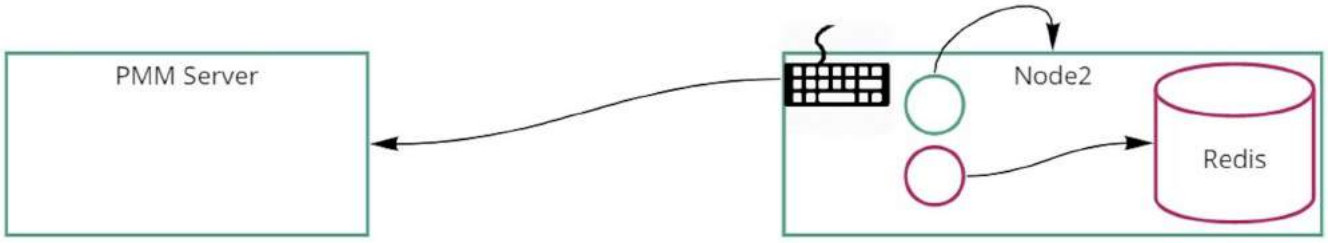
There two ways to get metrics from other exporters:

- `external` will collect metrics from the exporter that is run on the same host as PMM Client's connection to it by a port. (See more details with `pmm-admin add external --help`.)
- `external-serverless` is useful for collecting metrics from cloud services. You need a host and port number to add it to PMM Server. (See more details with `pmm-admin add external-serverless --help`.)

Here are the differences between `external` and `external-serverless` types.

Connection schema of external exporter:

Percona LLC and/or its affiliates ©2024



Connection schema of external serverless exporter:

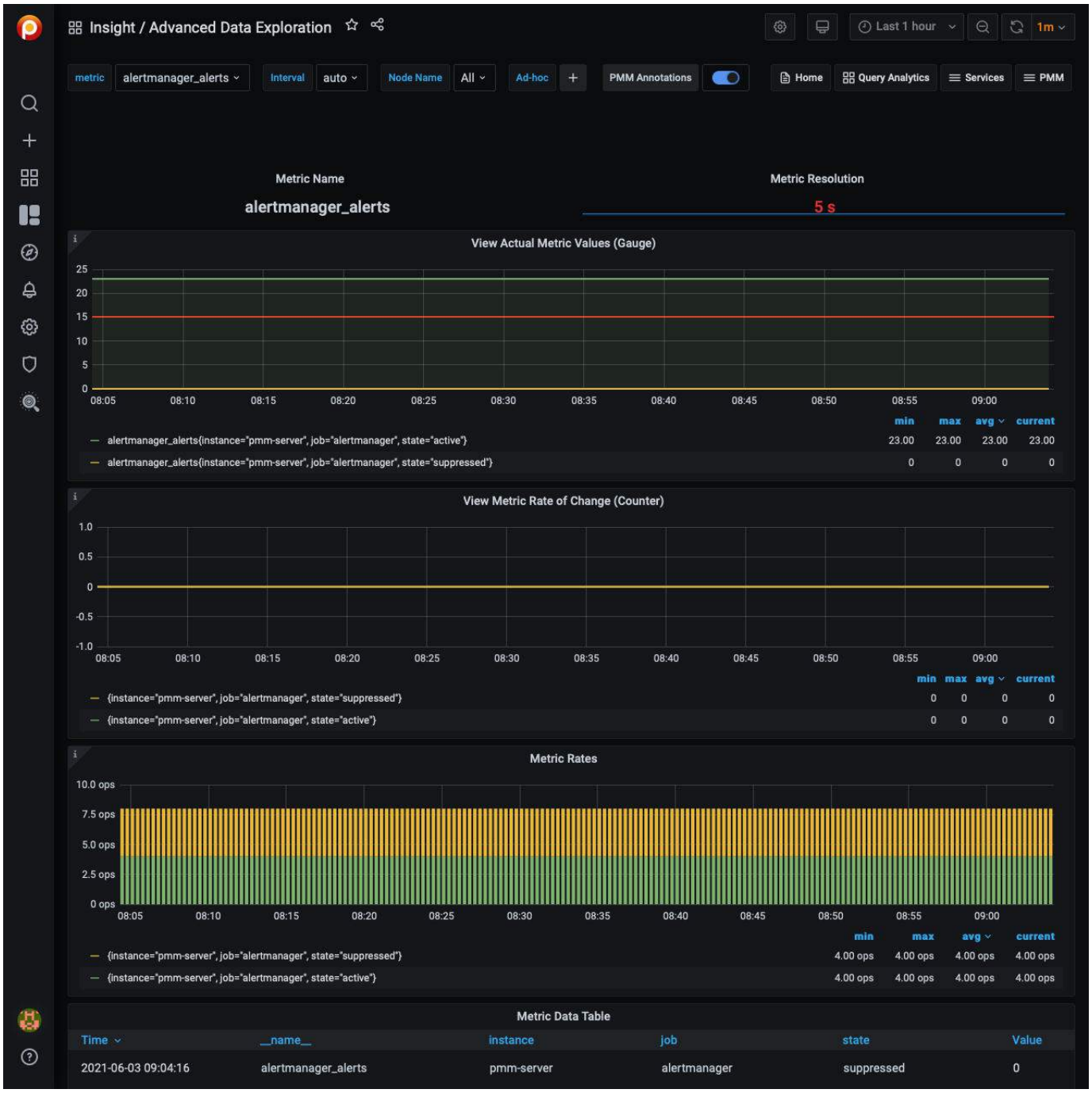
How I can add something not supported by PMM

PMM can collect any metrics in Open metrics or Prometheus exposition format. You must specify the host and port of these metrics using the `pmm-admin add external` or `pmm-admin add external-serverless` commands.

From this point, PMM will collect and store available metrics.

To browse and visualize collected metrics as a first step, we can look at the Advanced Data Exploration dashboard and select informative services and metrics.

Percona LLC and/or its affiliates ©2024



Another way is to create a new Grafana Dashboard to PMM as needed.

One more way is to search for an already created dashboard at https://grafana.com/grafana/dashboards for the added exporter and import it into PMM.

THIRD-PARTY EXPORTERS

You can find more exporters on the official Prometheus page.

CUSTOM EXPORTER

You can write a custom external exporter or extend your application to expose metrics in Prometheus format.

For more details see https://prometheus.io/docs/instrumenting/writing_exporters/.

Percona LLC and/or its affiliates ©2024

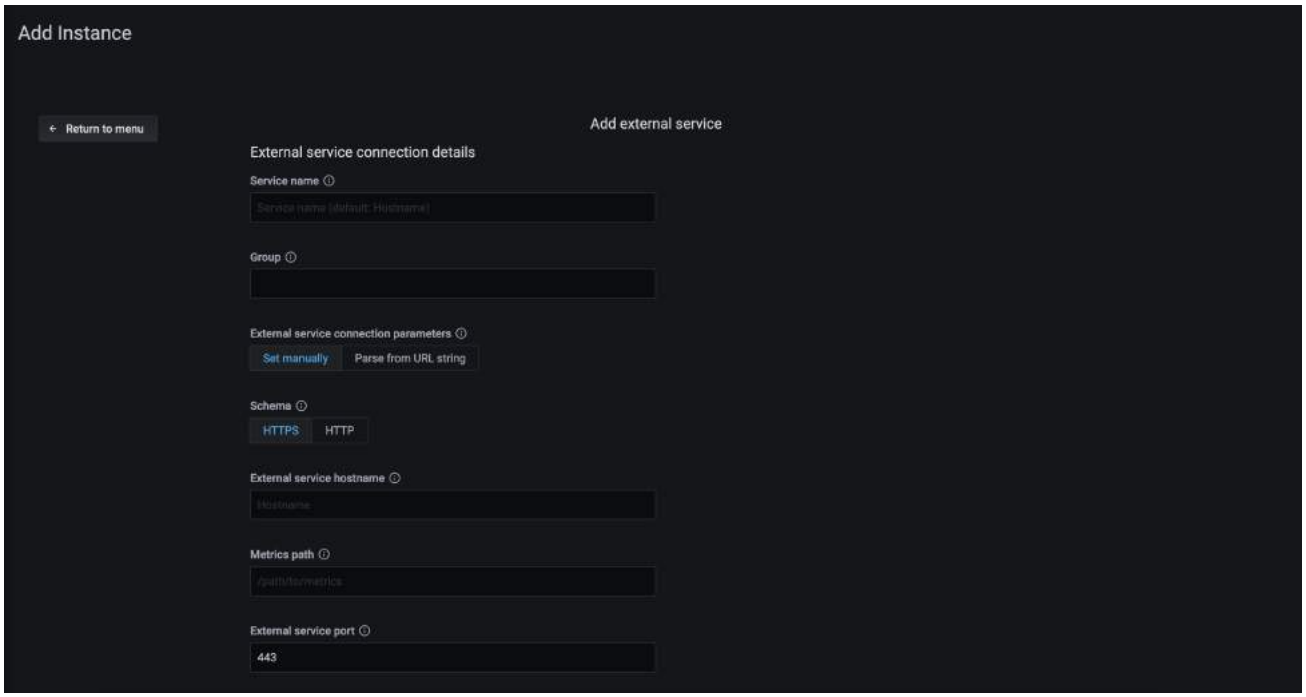
EXAMPLES

```
root@mysql1:~# pmm-admin add external --group=processes  --listen-port=9256
External Service added.
Service ID  : /service_id/6485f4fd-745b-4dfb-8b72-328e300f8b50
Service name: mysql1-processes
Group       : processes
```

- Add an exporter running on local port 9256 to the group called `processes`.
- Use the group and host names to automatically generate a service name.
- Use the default scheme and metrics path.

Adding an External service via UI

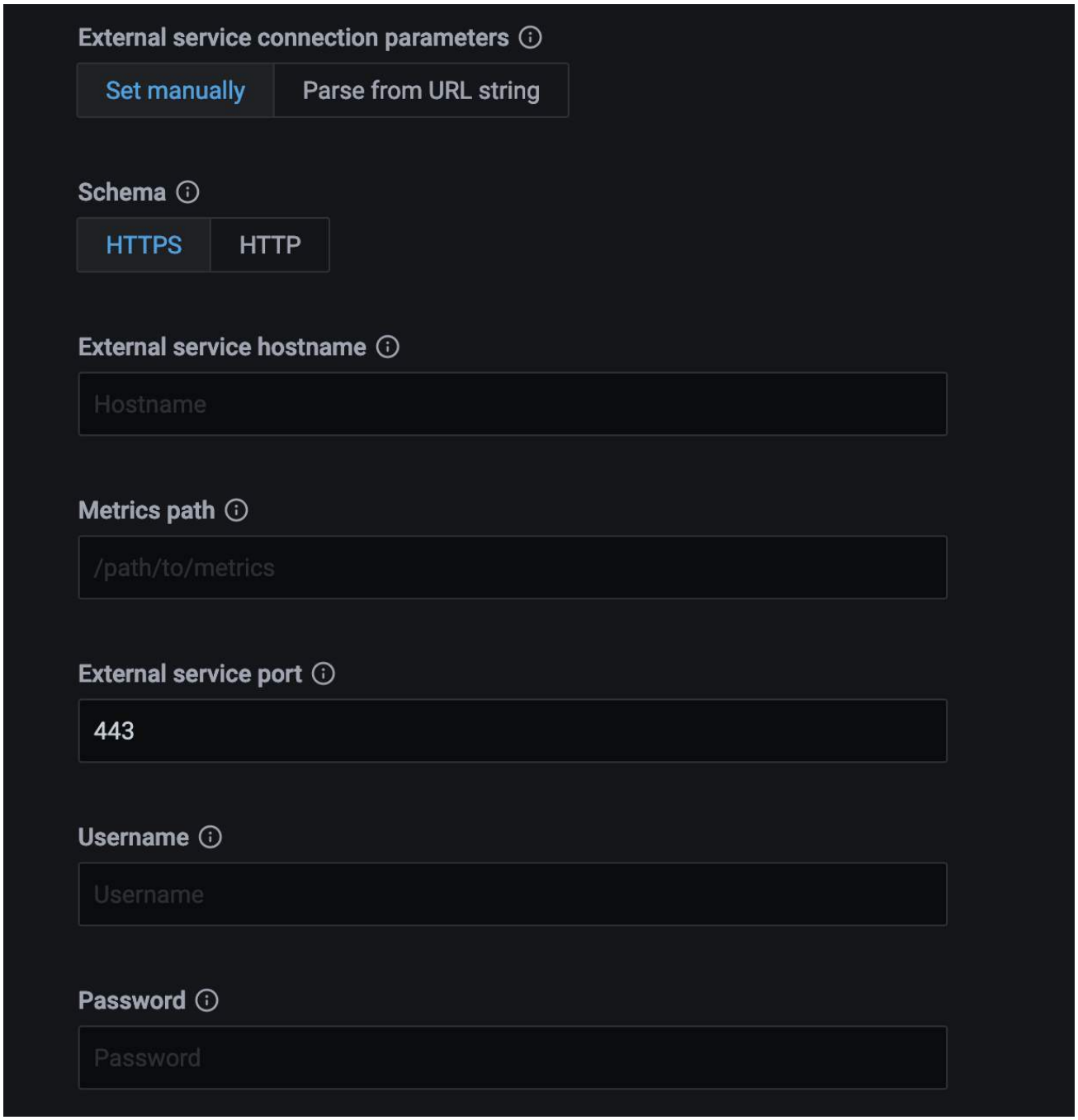
1. In the PMM web interface, go to *Configuration* → *PMM Inventory* → ⊕ *Add Instance*.
2. Select *External Service – Add a remote instance*.



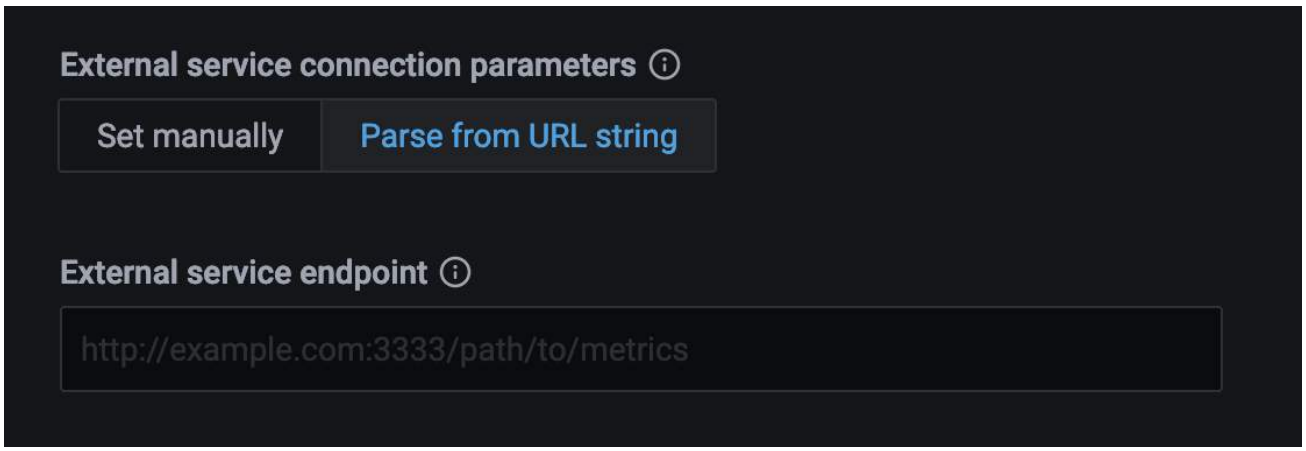
3. Fill the form and set the external service endpoint.

   The endpoint can be set manually:

Percona LLC and/or its affiliates ©2024



or by parsing required data from a URL string, in which case you only need to pass a valid URL.



Percona LLC and/or its affiliates ©2024

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

3.3.11 HAProxy

Adding HAProxy services

You can collect metrics from HAProxy on a node when:

- There is already a configured haproxy instance.
- See How to configure HAProxy.
- After HAProxy is running (default address http://localhost:8404/metrics) you can add it to PMM.
- Use the `haproxy` alias to enable HAProxy metrics monitoring.
- There is already a PMM Agent instance running.
- This node has been configured using the `pmm-admin config` command.

USAGE

```
pmm-admin add haproxy --listen-port=8404
```

where `listen-port` is the port number where HAProxy running. (This is the only required flag.)

The output of this command should look as follows:

```
HAProxy Service added.
Service ID  : /service_id/c481183f-70a2-443f-91e5-cae5cecd06a2
Service name: Ubuntu-haproxy
```

Additionally, one positional argument can be appended to the command line flags: a service name to be used by PMM. If not specified, they are substituted automatically as `<node>-haproxy`.

During adding here is connection check (can be skipped by flag `--skip-connection-check`). If HAProxy doesn't run properly on the given port then you will see an error message:

```
Connection check failed: Get "http://127.0.0.1:8404/metrics": dial tcp 127.0.0.1:8404: connect: connection refused.
```

Beside positional argument shown above you can specify service name with the following flags: `--username`, `--password`, `--metrics-path` (path for scraping metrics, default: /metrics) and `--scheme` (http or https). Here are some examples:

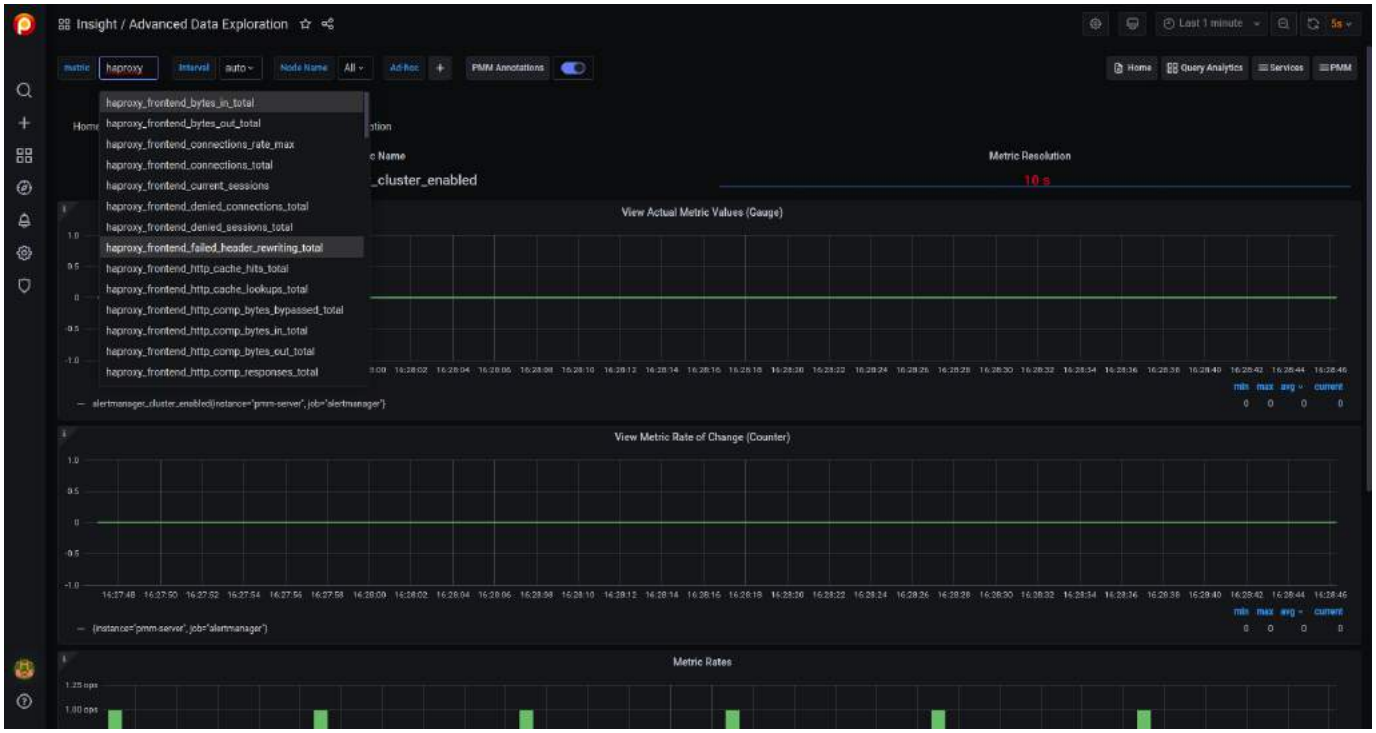
```
pmm-admin add haproxy --listen-port=8404 --username=pmm --password=pmm new-haproxy
pmm-admin add haproxy --listen-port=8404 --metrics-path=/prom-metrics --scheme=https
```

Here you can check list of all available flags: pmm-admin.

You can also add HAProxy by UI in Grafana: Select *Configuration* → *PMM Inventory* → ⊕ *Add Instance*.

Percona LLC and/or its affiliates ©2024

HAProxy data is visible in the *Advanced Data Exploration* dashboard:



Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

3.3.12 Remote instances

Recommended settings

When monitoring remote instances including RDS and Google instances, network latency might affect the scrape process and throw timeout errors. For this reason, it is recommended to lower the metrics resolution.

Starting with PMM 2.18, the scrape timeout has been updated according to the following rules:

- For resolutions <= 2 seconds, scrape timeout is 1 second.
- For resolutions <= 10 seconds, timeout is set to resolution minus 1 second. For example, for 10 second resolution, timeout will be set at 9 seconds.
- For lower resolutions (values > 10 seconds), the scrape timeout is set to 90% of the resolution time. For example, for 60 second resolution, the scrape timeout will be set to 54 seconds.

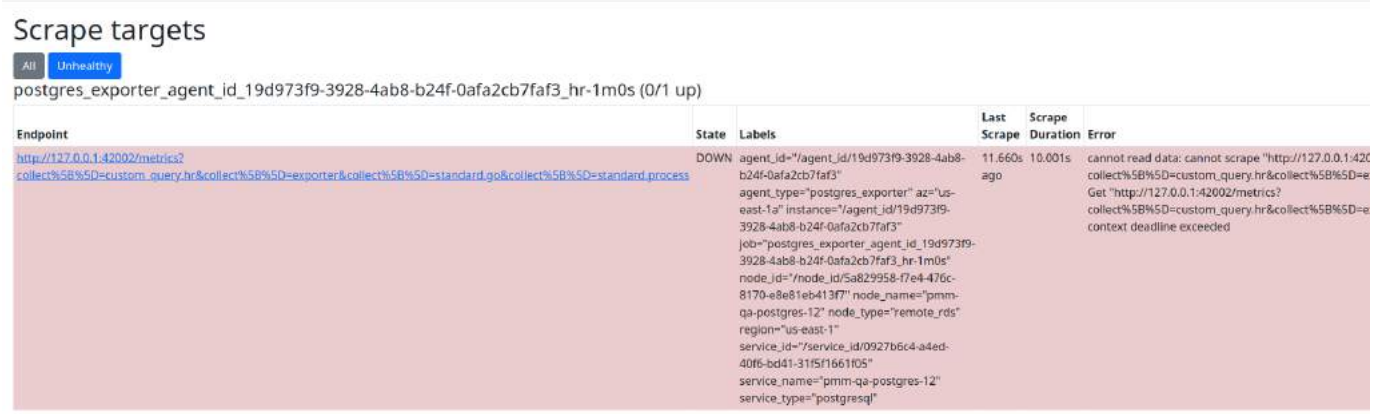
Percona LLC and/or its affiliates ©2024

How to check for scrape timeouts

Sometimes it is hard to check if you are using the correct values to scrape or if there some other reason why there is no data in a dashboard even when the instance has been added correctly and the agent is running.

One additional step you can do is to check for scrape target statuses. Browse to `http://<your-pmm-server-address>/prometheus/targets` and then click on the Unhealthy button.

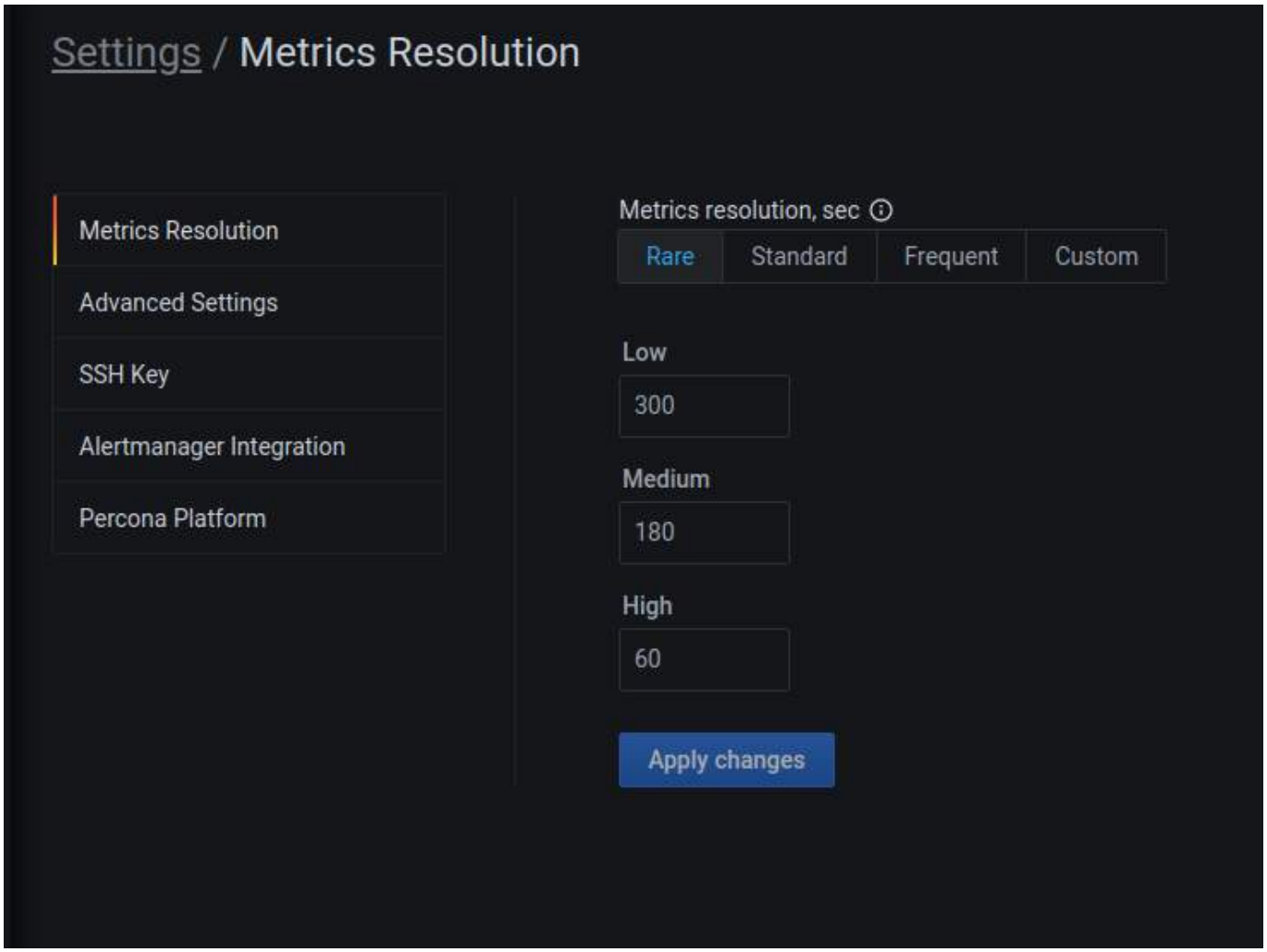
The page will show only agents having issues while scrapping and the scrape result including the error messages.



In the example here, there is a message that says: *context deadline exceeded* and the scrape duration column says the scrape took 10 seconds; this means that the exporter didn't respond in the 10 seconds the scrape process was allowed to run due to the configured metric resolutions and their timeouts.

In this case, we can lower the metric resolutions increasing these values as shown in the image below.

Percona LLC and/or its affiliates ©2024



Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-29

Percona LLC and/or its affiliates ©2024

4. Get started

4.1 Get Started

- User Interface
  - Using the web-based user interface.
  - Finding dashboards.
  - Rendering dashboard images.
  - Viewing graph details.
  - Annotating events.
- Percona Alerting
- Backup and Restore
- Query Analytics, a specialized dashboard for detailed query analysis.
- Advisors: Enabling and seeing the results of database Advisor checks.
- DBaaS: Configuration for DBaaS and UI for creating Database clusters.

These features are currently only available for PMM Admin users:

- Backup
- DBaaS
- Percona Alerting
- Advisors

To use these features you must be logged in as a PMM Admin user and activate the features.

If you are logged in as a user that has a Viewer or Editor role you'll see an 'insufficient access' message when trying to use these features.

4.1.1 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-02-23

4.2 User Interface

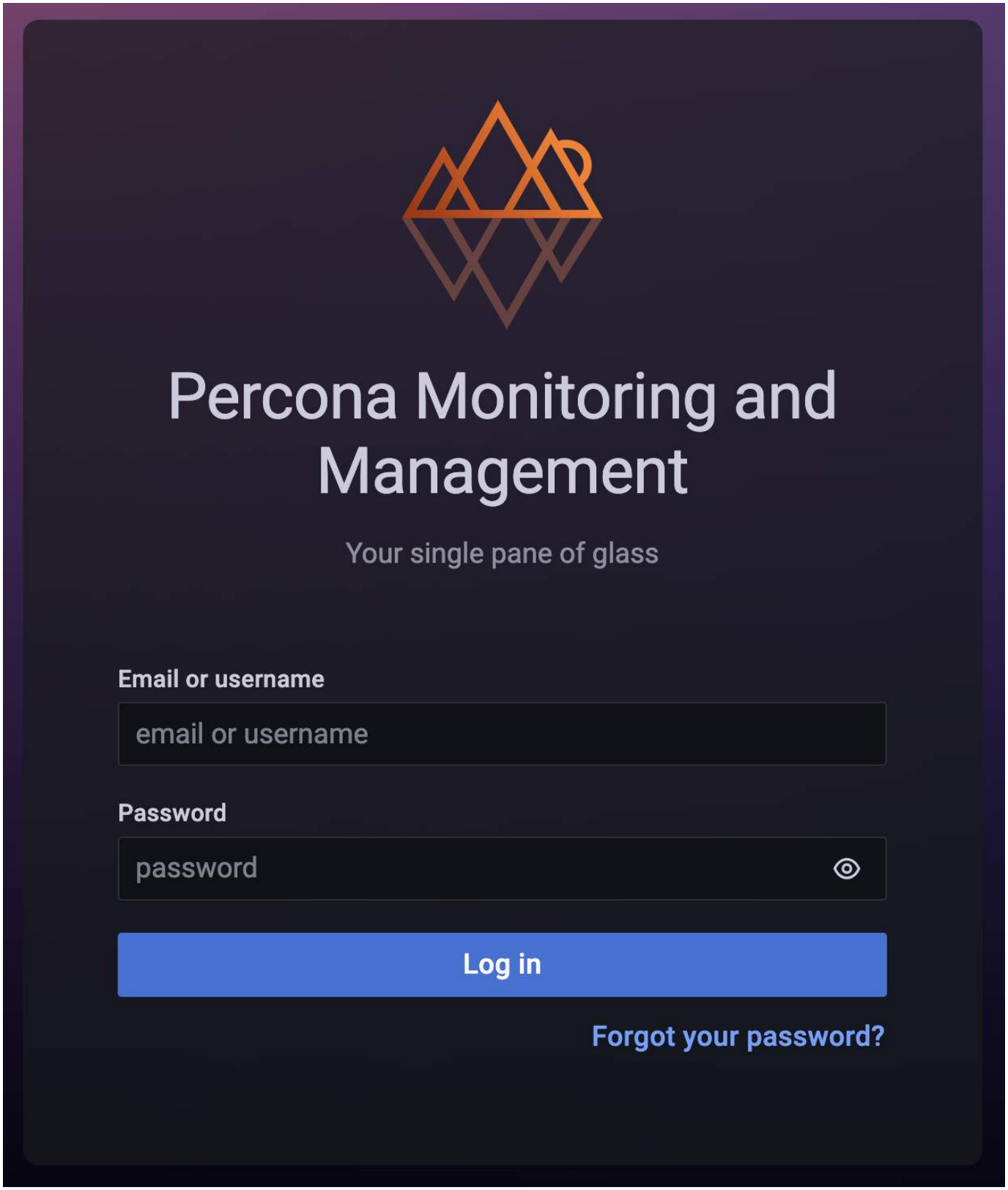
How to log in, how the user interface is laid out, and what the controls do.

PMM's user interface is a browser application based on Grafana.

Percona LLC and/or its affiliates ©2024

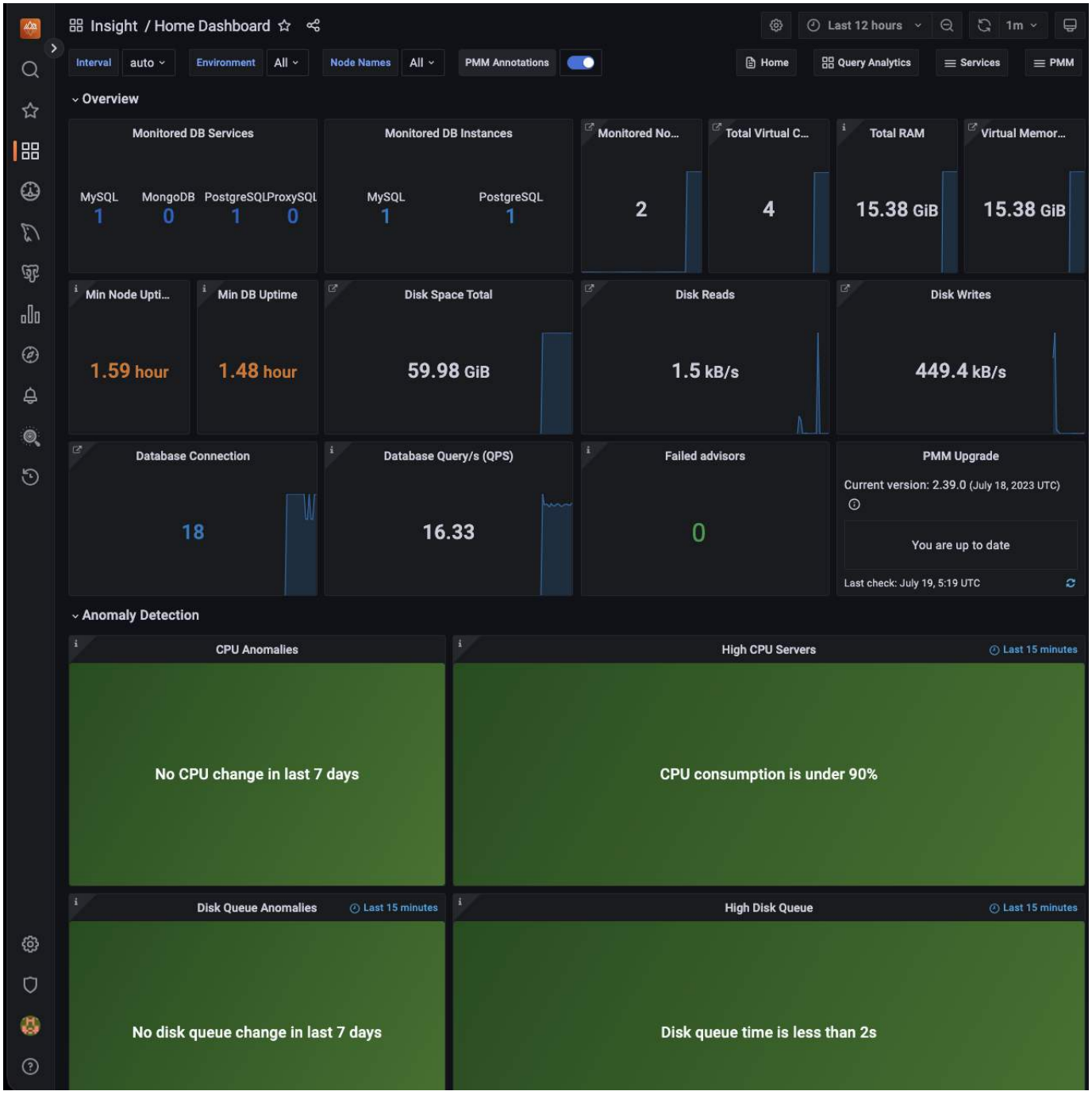
4.2.1 Logging in

1. Start a web browser and in the address bar enter the server name or IP address of the PMM server host.
2. The page loads showing the PMM login screen.



Percona LLC and/or its affiliates ©2024

1. Enter the username and password given to you by your system administrator. The defaults are:
2. Username: `admin`
3. Password: `admin`
4. Click *Log in*.
5. If this is your first time logging in, you'll be asked to set a new password. (We recommend you do.)
6. Either enter a new password in both fields and click *Submit*, or,
7. click *Skip* to use the default password.
8. The PMM Home dashboard loads.



Percona LLC and/or its affiliates ©2024

4.2.2 Dashboards

The interface is a collection of web pages called *dashboards*.

Dashboards are grouped into *folders*. You can customize these, by renaming them or creating new ones.

The area inside dashboards is populated by *panels*. Some are in collapsible panel groups. A panel can show a value, a graph, a chart, or a visual representation of a set.

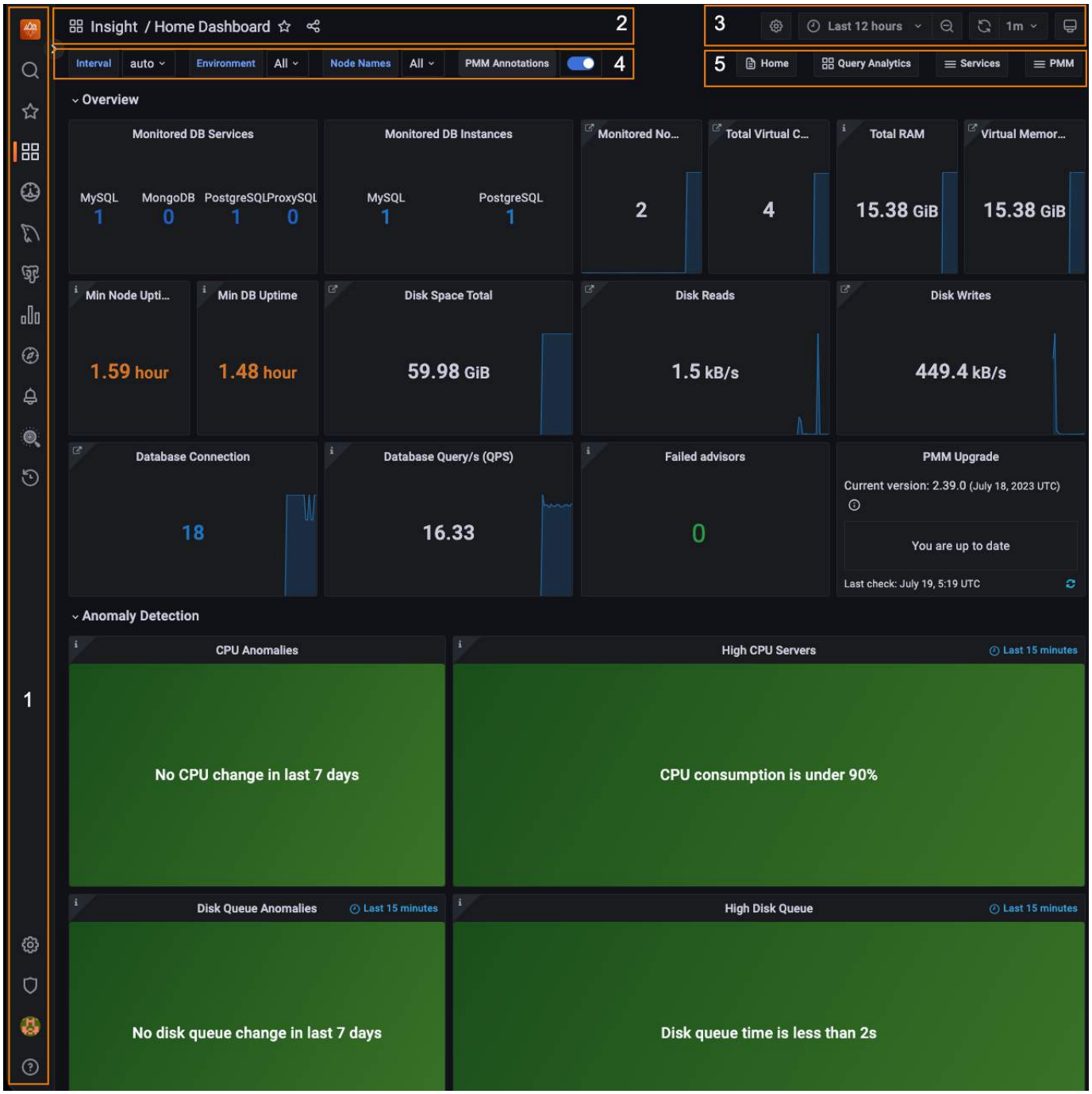
4.2.3 Controls

These menus and controls appear on all dashboards:

1. Main menu (also *Grafana menu*, *side menu*).
2. Navigation bar.
3. View controls.
4. View selectors (with dynamic contents).
5. Shortcut menu (with dynamic contents).

(For details see UI Components.)

Percona LLC and/or its affiliates ©2024



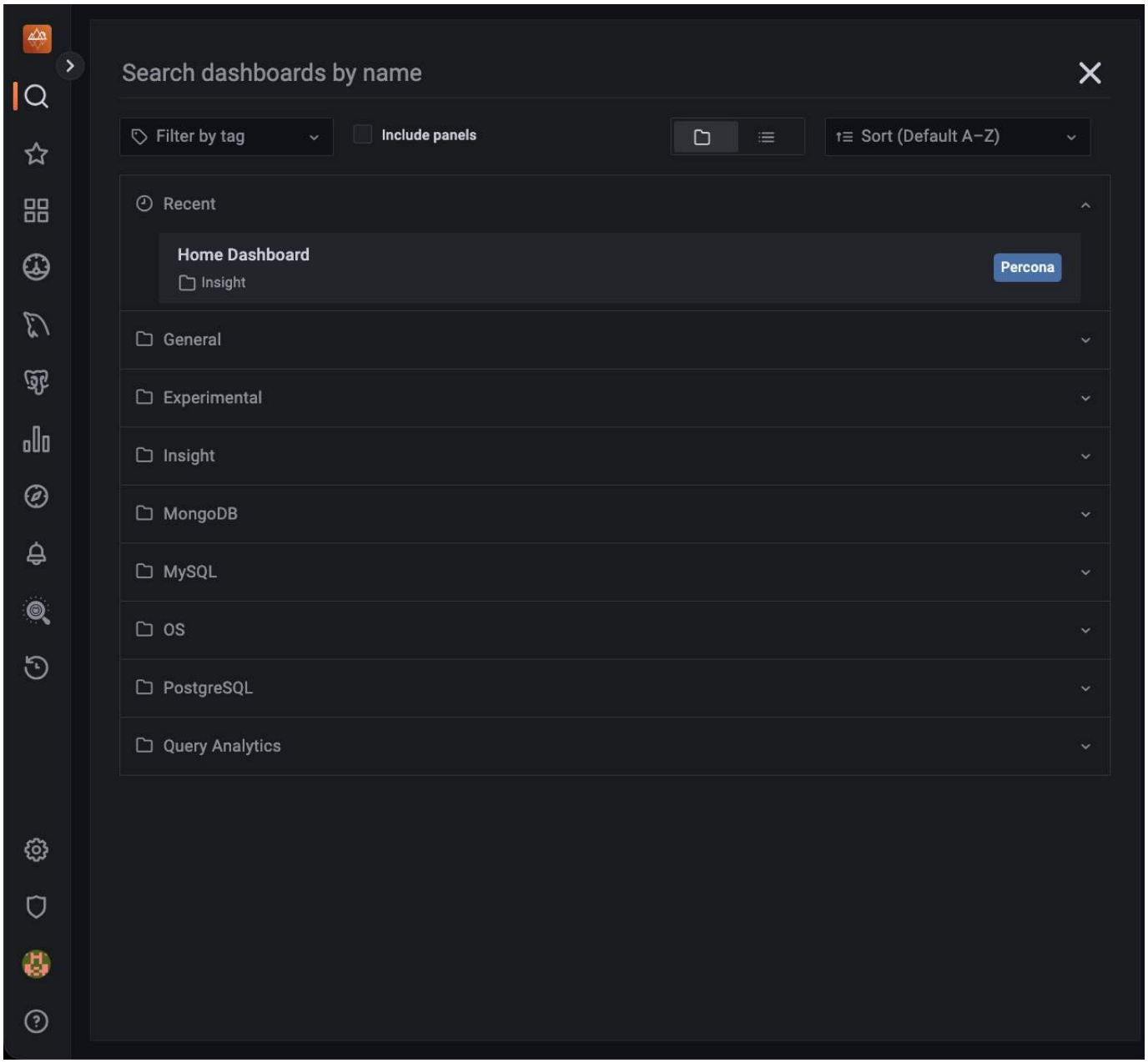
4.2.4 Navigation

Search for a dashboard by name

There are two ways to open the dashboard search page. (Each takes you to the same search screen.)

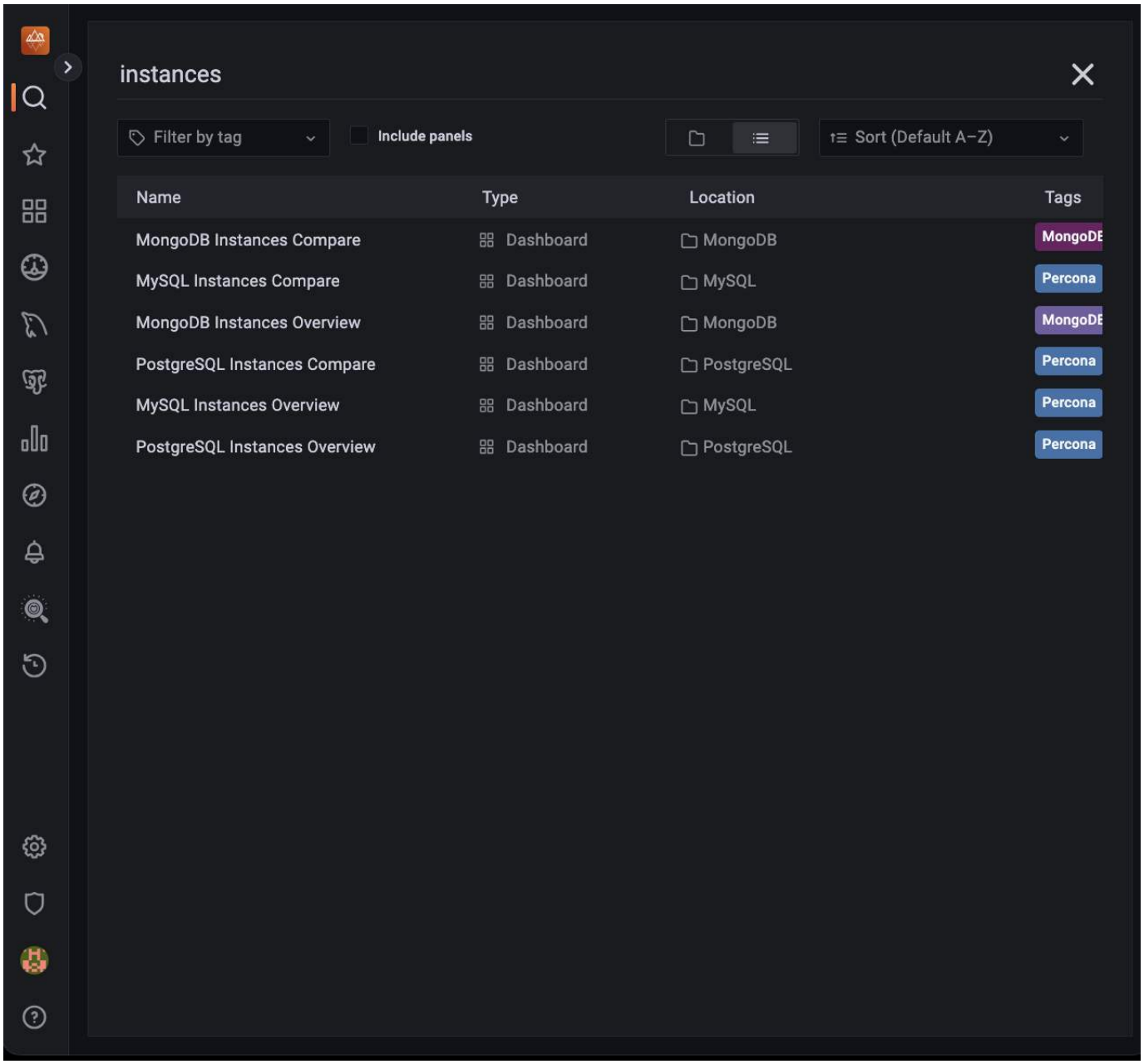
- Click the icon in the main menu.
- Click the dashboard name in the navigation bar (top row, to the right of the icon). (To search within the current folder, click the folder name instead of the dashboard name.)

Percona LLC and/or its affiliates ©2024



1. Click *Search dashboards by name* and begin typing any part of the dashboard name (in this example, *“Instances”*).

Percona LLC and/or its affiliates ©2024



1. Click one of the search results to go to that dashboard. Change the search text to refine the list.
2. To abandon the search, click the icon at the end of the search bar.

Open a dashboard with the menu

In the main menu, the *PMM Dashboards* icon reveals a submenu containing links to all PMM dashboards grouped by service type. (This menu will replace the shortcut menu which has links to commonly-used dashboards.)

4.2.5 Panels

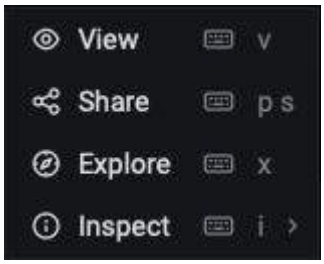
Charts, graphs and set-based panels reveal extra information when the mouse is moved over them.

Some panels have an information icon **i** in the top left corner. Mouse over this to reveal panel information.

Percona LLC and/or its affiliates ©2024

Panel menu

At the top of each panel and to the right of the panel name is the *panel menu*.



Tip

The menu is hidden until you mouse over it. Look for the symbol in the title bar of a panel.

Item	Description
View	Open the panel in full window mode.
Share	Share the panel's link or image.
Explore	Run PromQL queries.
Inspect	See the panel's data or definition.
More	(Only charts and graphs) Additional options.

View

The *View* menu items opens panels in full-window mode. This is useful for graphs with several metrics.

Exit a panel's full window mode by pressing *Escape* or clicking the left arrow next to the dashboard name.

See also

- How to render dashboard images
- How to annotate special events

4.2.6 Timezones

By default Grafana uses the timezone from your web browser. However, you can change this setting.

Set user timezone

1. On the left menu, hover your cursor over your avatar and then click *Preferences*.
2. Click to select an option in the *Timezone list*.
3. Click *Save*.

4.2.7 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2023-08-14

4.3 Percona Alerting

Percona Alerting is the new Alerting feature introduced in PMM 2.31. This replaces the Integrated Alerting feature available in previous versions.

Alerting notifies of important or unusual activity in your database environments so that you can identify and resolve problems quickly. When something needs your attention, Percona Alerting can be configured to automatically send you a notification through your specified contact points.

PMM 2.31 introduced Percona Alerting which replaces Integrated Alerting in previous PMM versions. In addition to full feature parity, Percona Alerting includes additional benefits like Grafana-based alert rules and a unified, easy-to-use alerting command center on the **Alerting** page.

Percona Alerting is enabled by default in the PMM Settings. This feature adds the **Percona templated alerts** option on the **Alerting** page.

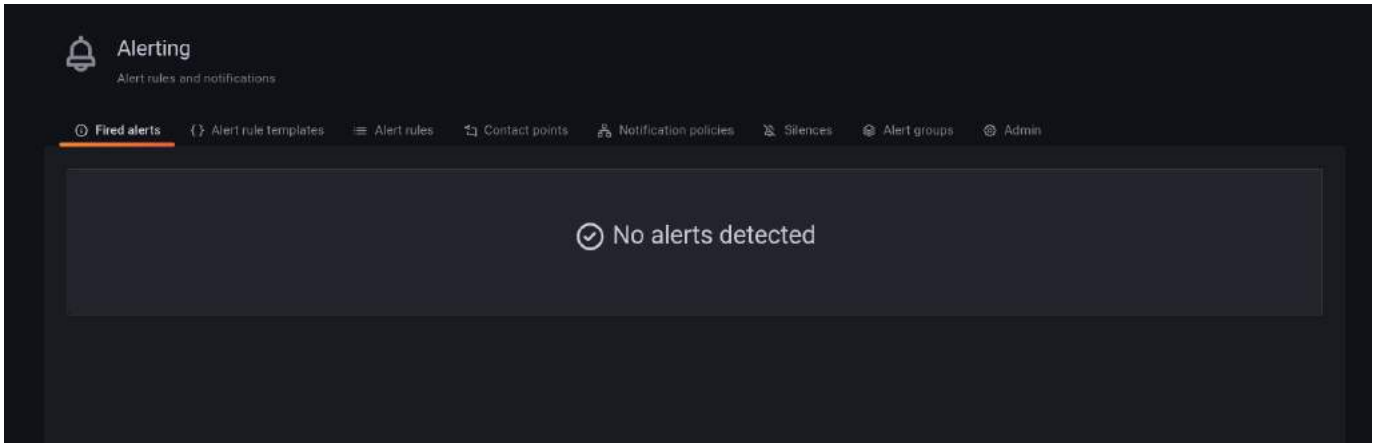
4.3.1 Alert types

Percona Alerting is powered by Grafana infrastructure. It leverages Grafana's advanced alerting capabilities and provides pre-configured Alert Rule Templates that simplify creating powerful alerting rules.

Depending on the datasources that you want to query, and the complexity of your required evaluation criteria, Percona Alerting enables you to create the following types of alerts:

- **Percona templated alerts**: alerts based on a set of Percona-supplied templates with common events and expressions for alerting. If you need custom expressions on which to base your alert rules, you can also create your own templates.
- **Grafana managed alerts**: alerts that handle complex conditions and can span multiple different data sources like SQL, Prometheus, InfluxDB, etc. These alerts are stored and executed by Grafana.

The Alerting page contains are split into eight tabs: Fired Alerts, Alert Rules, Alert Rule Templates, Contact Points, Notification Policies, Silences, Alert Groups and Admin.

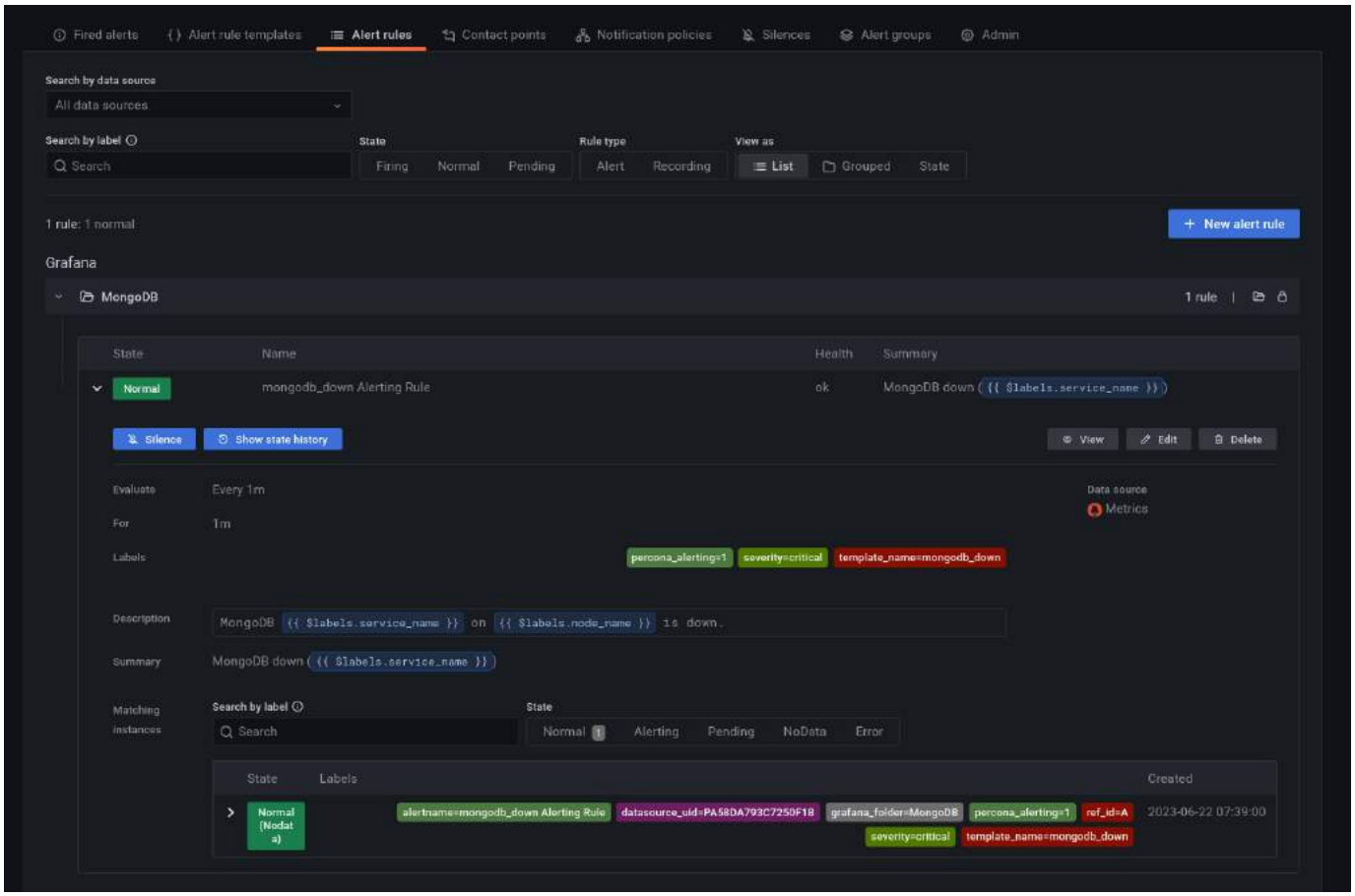


Percona LLC and/or its affiliates ©2024

4.3.2 Alert rules

Alert rules describe the circumstances under which you want to be alerted. The evaluation criteria that you define determine whether an alert will fire.

An alert rule consists of one or more queries and expressions, a condition, the frequency of evaluation, and the duration over which the condition is met. For example, you might configure an alert to fire and trigger a notification when MongoDB is down.



An alert rule can be in three possible states:

- Normal: Everything is working correctly and the conditions specified in the rule has not been met. This is the default state for newly created rules.
- Pending: The conditions specified in the alert rule has been met, but for a time that is less than the configured duration.
- Firing: Both the conditions and the duration specified in the alert rule have both been met.

It takes at least one evaluation cycle for an alert rule to transition from one state to another (e.g., from `Normal` to `Pending`).

4.3.3 Alert rules templates

PMM provides a set of Alert Rule templates with common events and expressions for alerting. These templates can be used as a basis for creating Alert Rules. You can also create your own templates if you need custom expressions.

Percona LLC and/or its affiliates ©2024

You can check the alert templates available for your account under **Alerting > Alert rule templates** tab. PMM lists here the following types of templates:

- Built-in templates, available out-of-the-box with PMM.
- Templates downloaded from Percona Platform.
- Custom templates created or uploaded on the **Alerting page > Alert Templates** tab. You can also store your custom template files in your `/srv/alerting/templates` directory and PMM will load them during startup.

Create alert rules from alert rule templates

This section focuses on creating an alert rule based on PMM templates. For information on working with the other alert types, check the Grafana documentation on Grafana Labs.

Provision alert resources

Before creating PMM alert rules, configure the required alert resources:

1. Go to **Configuration > PMM Settings** and ensure that the **Alerting** option is enabled. This is enabled by default starting with PMM 2.31. However, if you have disabled it, the **Alerting** page displays only Grafana-managed alert rules. This means that you will not be able to create alerts based on PMM templates.
2. Go to **Dashboards > Browse** and check the folders available for storing alert rules. If none of the available folders are relevant for your future alert rules, click **New > New Folder** and create a custom one.
3. Go to **Alerting > Alert Rule Templates** and check the default PMM templates. If none of the templates include a relevant expression for the type of alerts that you want to create, click **Add** to create a custom template instead.

Configure alert templates

Alerts templates are YAML files that provide the source framework for alert rules. Alert templates contain general template details and an alert expression defined in MetricsQL. This query language is backward compatible with PromQL.

Create custom templates

If none of the default PMM templates contain a relevant expression for the alert rule that you need, you can create a custom template instead.

You can base multiple alert rules on the same template. For example, you can create a `pmm_node_high_cpu_load` template that can be used as the source for alert rules for production versus staging, warning versus critical, etc.

Percona LLC and/or its affiliates ©2024

Template format

When creating custom templates, make sure to use the required template format below:

- **name** (required): uniquely identifies template. Spaces and special characters are not allowed.
- **version** (required): defines the template format version.
- **summary** (required): a template description.
- **expr** (required): a MetricsQL query string with parameter placeholders.
- **params**: contains parameter definitions required for the query. Each parameter has a name, type, and summary. It also may have a unit, available range, and default value.
  - **name** (required): the name of the parameter. Spaces and special characters are not allowed.
  - **summary** (required): a short description of what this parameter represents.
  - **unit** (optional): PMM currently supports either s (seconds) or % (percentage).
  - **type** (required): PMM currently supports the `float` type. `string`, `bool`, and other types will be available in a future release.
  - **range** (optional): defines the boundaries for the value of a float parameter
- **value** (optional): default parameter value. Value strings must not include any of these special characters: `< > ! @ # $ % ^ & * ( ) _ / \ ' + - =` (space)
- **for** (required): specifies the duration of time that the expression must be met before the alert will be fired
- **severity** (required): specifies default alert severity level
- **labels** (optional): are additional labels to be added to generated alerts
- **annotations** (optional): are additional annotations to be added to generated alerts.

Template example

```
---
templates:
  - name: pmm_node_high_cpu_load
    version: 1
    summary: Node high CPU load
    expr: |-
      (1 - avg by(node_name) (rate(node_cpu_seconds_total{mode="idle"}[5m])))
      * 100
      > bool [[ .threshold ]]
    params:
      - name: threshold
        summary: A percentage from configured maximum
        unit: "%"
        type: float
        range: [0, 100]
        value: 80
    for: 5m
    severity: warning
    annotations:
      summary: Node high CPU load ({{ $labels.node_name }})
      description: |-
        {{ $labels.node_name }} CPU load is more than [[ .threshold ]]%.
```

Test alert expressions

If you want to create custom templates, you can test the MetricsQL expressions for your custom template in the **Explore** section of PMM. Here you can also query any PMM internal database.

To test expressions for custom templates:

1. On the side menu in PMM, choose **Explore > Metrics**.

Percona LLC and/or its affiliates ©2024

2. Enter your expression in the **Metrics** field and click **Run query**.

For example, to check the CPU usage, Go to **Explore > Metrics** in your PMM dashboard and run the query expression below:

```
(1 - avg by(node_name) (rate(node_cpu_seconds_total{mode="idle"}[5m]))) * 100
```

Note that to paste the query above, **Explore** must be in `Code` mode, and not in `Builder` mode.

Add an alert rule

After provisioning the resources required for creating Percona templated alerts, you are now ready to create your alert rule:

1. Go to **Alerting > Alert Rules**, and click **New alert rule**.
2. On the **Create alert rule** page, select the **Percona templated alert** option. If you want to learn about creating Grafana alerts instead, check our Grafana's documentation.
3. In the **Template details** section, choose the template on which you want to base the new alert rule. This automatically populates the **Name**, **Duration**, and **Severity** fields with information from the template. You can change these values if you want to override the default specifications in the template.
4. In the **Filters** field, specify if you want the alert rule to apply only to specific services or nodes. For example: `service_name=ps5.7`. When creating alert rule filters, consider the following:
   - Filters use conjunction semantics. This means that if you add more than one filter, PMM will combine their conditions to search for matches: filter 1 AND filter 2 AND filter 3.
   - **Label** must be an exact match. You can find a complete list of labels using the **Explore** menu in PMM.
5. From the **Folder** drop-down menu, select the location where you want to store the rule.
6. Click **Save and Exit** to close the page and go to the **Alert Rules** tab where you can review, edit and silence your new alert.

Percona LLC and/or its affiliates ©2024

4.3.4 Contact points

Contact points specify where Percona Alerting should deliver notifications for alerts. PMM can be configured via a Notification policy to send a notification to specified contact points whenever an alert is fired.

Depending on the severity of an alert, you might want to send different alert notifications to different channels. For example, you can deliver common notifications via a Slack channel, but send an email notification for potentially critical issues.

Percona Alerting uses email as the default contact point but you can choose from a variety of other contact points, including Slack, Webhooks, PagerDuty, and more.

Before Percona Alerting can send out email notifications via email contact points, you will need to:

1. Configure Email (SMTP) server settings.
2. Configure a contact point to define the email delivery options

Contact points with invalid settings show a **No Attempts** status under **Alerting > Contact points**.

Configure Email (SMTP) server settings

To use SMTP with a PMM Docker installation:

1. Create an `.env` file and populate it with your SMTP credentials (and other environment variables) as follows:

   ```
   GF_SMTP_ENABLED=true
   GF_SMTP_HOST=smtp.gmail.com:587
   GF_SMTP_USER=email@domain.com
   GF_SMTP_PASSWORD=<YOUR_SMTP_PASSWORD>
   GF_SMTP_SKIP_VERIFY=false
   GF_SMTP_FROM_ADDRESS=email@domain.com
   GF_SMTP_FROM_NAME=Percona Alerting
   ```

   Below is a summary of each environment variable above:

   - `GF_SMTP_ENABLED`: When true, enables Grafana to send emails.
   - `GF_SMTP_HOST`: Host address of your SMTP server.
   - `GF_SMTP_USER`: Username for SMTP authentication.
   - `GF_SMTP_PASSWORD`: Password for SMTP authentication
   - `GF_SMTP_SKIP_VERIFY`: When true, verifies SSL for the SMTP server.
   - `GF_SMTP_FROM_ADDRESS`: Email address to be used when sending out emails.
   - `GF_SMTP_FROM_NAME`: Name to be used when sending out emails.

   NB: If you are using your Gmail's SMTP credentials as shown above, you will have to generate an app password and fill it in as the value of your $GF_SMTP_PASSWORD variable.

2. Pass in the `.env` file to Docker run using the `--env-file` flag:

   ```
   docker run --env-file=.env -p 443:443 -p 80:80 percona/pmm-server:2
   ```

   This command starts a docker container and will keep running as long as the container is also running. Stopping the command (e.g with Ctrl+C) will stop the container hence, subsequent commands should be run in a new terminal.

Restore SMTP settings following an upgrade

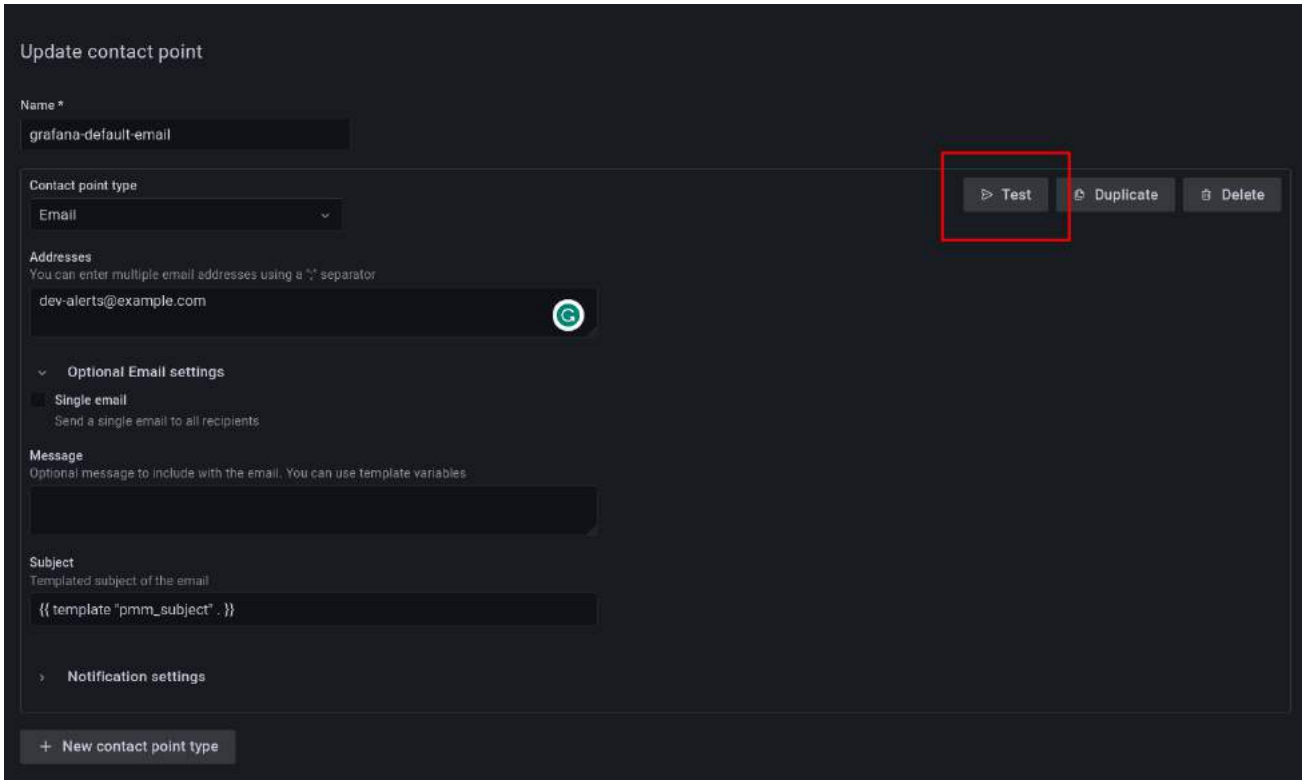
If you configured PMM to use SMTP settings via environment variables, you do not need to do anything after an upgrade as your settings will be transferred.

Percona LLC and/or its affiliates ©2024

Configure an Email contact point

After configuring the SMTP settings, specify email delivery options for an Email contact point:

1. Go to **Alerting > Contact points**.
2. Click the edit button next to the **grafana-default-email** to update PMM's default Email contact point, or click **New contact point** to create a custom one.
3. Enter a contact point name, and add the email addresses for the recipients of the email notifications.
4. Expand **Optional settings** and fill in any other relevant settings:
   - Enable the **Single email** option to send a single email to the recipients containing alerts that are firing. For example, if an alert fires for three nodes, this would send only one email listing all three alerts.
   - Add an optional message to include with the email notifications.
   - Edit the email subject for the notifications. The default subject line uses the following format FIRING: *number of alerts firing for the alert rule*.
5. If you do not want to be notified when an alert resolves, expand **Notification settings**, and tick the **Disable Resolved Message** checkbox.
6. If you want your contact point to notify via multiple channels, for example, both via Email and Teams, click **New contact point type** and fill out additional contact point type details.
7. Click the **Test** button to send a test email and make sure your contact point works as expected.



8. Click the **Save contact point** button at the bottom of the page. The contact point is now listed under **Alerting > Contact points**.

CREATE ADDITIONAL CONTACT POINTS

In addition to Email contact points, you can add a variety of other contact points, including Slack, email, webhooks, PagerDuty, and more.

Follow the steps above to create additional contact points. Different contact points require different configuration information. For example, for Slack, PMM requires the recipient information, the API token and the webhook URL, which you can get from your Slack administrator.

Percona LLC and/or its affiliates ©2024

Notification policies

Notification policies determine how notifications (triggered by alerts) are routed to contact points by defining where, when, and how to send notifications.

For example, you might specify a limit for the number of times a notification is sent during a certain period. This helps ensure that you don't spam your Slack channel with too many notifications about the same issue.

ROOT NOTIFICATION POLICY

Percona Alerting comes pre-configured with a Notification Root Policy, which is the default notification policy. It uses the **grafana-default-email** contact point and is applied to all alerts that don't have a custom notification policy assigned to them.

HOW MATCHING WORKS

Policies can have one or more child policies. An alert matches if the alert's labels match all the **Matching Labels** specified on the policy.

Alerts that don't match any specific policies are handled by the root policy. The root policy also handles any alert rules for which the assigned custom notification policy has been deleted, to ensure notifications for existing alerts continue to be delivered.

EDIT THE ROOT NOTIFICATION POLICY

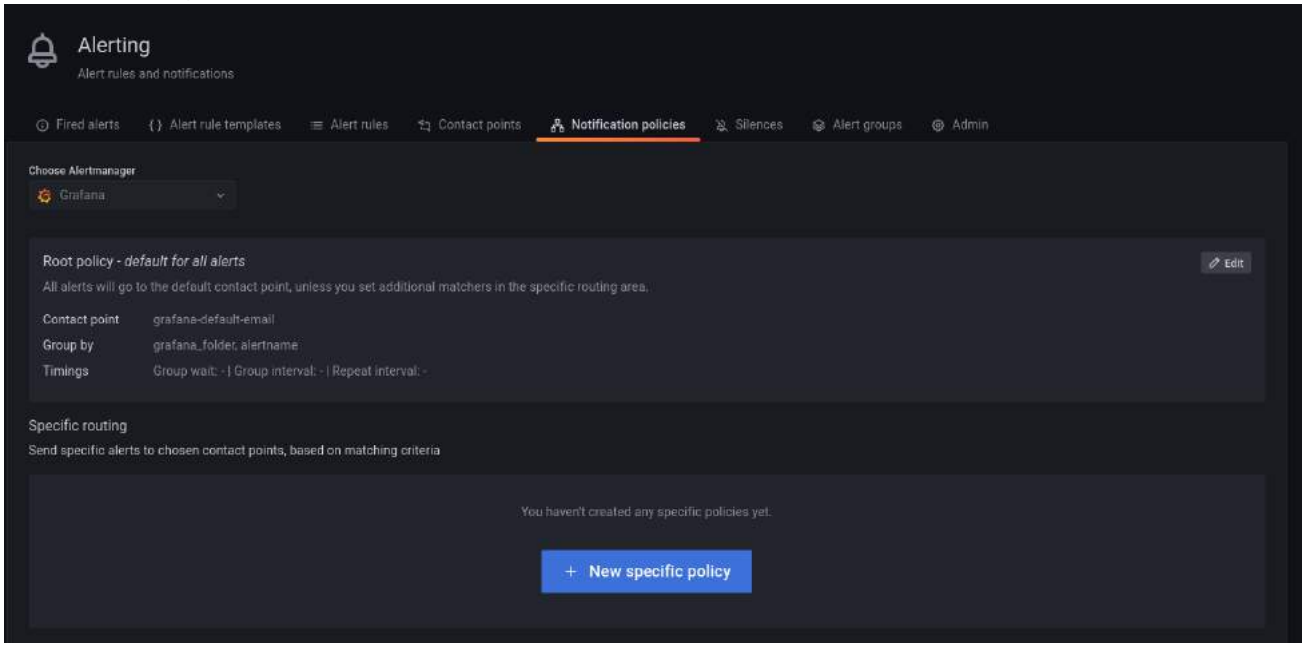
1. Go to **Alerting > Notification policies** tab.
2. Click **Edit** on the top right of the root policy box.
3. Choose whether to keep the default Email contact point, select a new available contact point or create a new one.
4. In the **Group by** field, specify how alert rules should be processed into notifications. If multiple alerts are matched for this policy, they will be grouped based on the labels you specify, and a notification will be sent per group.
5. Expand the **Timing options** section and specify how notification wait times should be processed. These are short pauses the system can take to efficiently process multiple sets of alerts for notifications:
6. **Group wait**: The default is to wait 30 seconds to buffer alerts of the same group before sending a notification initially.
7. **Group interval**: The default is to wait five minutes before sending a batch of new alerts after the first notification was sent.
8. **Repeat interval**: The default is to wait four hours before resending an alert.
9. Click **Save** to save your changes.

CREATE A NEW NOTIFICATION POLICY

To create a new notification policy:

Percona LLC and/or its affiliates ©2024

1. Go to **Alerting > Notification policies** tab.



2. Click **New specific policy**.
3. The **Matching labels** section defines the rules for matching alert labels. The matching label is a combination of label name, operator and label value, where the label name is any valid label in your environment. For example: `node_name`, `cluster`, etc. A policy will match an alert if the alert's labels match all the matching labels specified on the policy. If there are no matchers, **the policy will handle all the alert instances**. For example, you could add a **node_name=pmm-server** matcher to send out notifications only for this node.
4. Select an existing contact point for the policy.
5. Enable **Continue matching subsequent sibling nodes** to continue matching subsequent siblings of the policy after an alert matched the parent policy. This can be useful, for example, when you want to send notifications to a catch-all contact point as well as to one of more specific contact points handled by subsequent policies.
6. Toggle **Override grouping** if you do not want to use root policy grouping.
7. Toggle **Override general timings** to specify how often you want to wait until the initial notification is sent for a new group. When this is disabled, PMM uses root policy group timings instead.
8. Add a mute timing if you want to mute notifications or this policy for a specific, regular interval. For example, you can create a mute to suppress trivial notifications during weekends. Mute timings are different from silences in the sense that they are recurring, while silences have a fixed start and end time.

> **Important**
>
> Time specified in mute timing must be in UTC and military format i.e. 14:00 not 2:00 PM.

4.3.5 Silence alerts

Create a silence when you want to suppress/stop alerts and their associated notifications for a very specific amount of time. Silences default to today's current date and have a default duration of two hours.

You can also schedule a silence for a future date and time. This is referred to as a `Pending` silence, which can be observed on the Silences page.

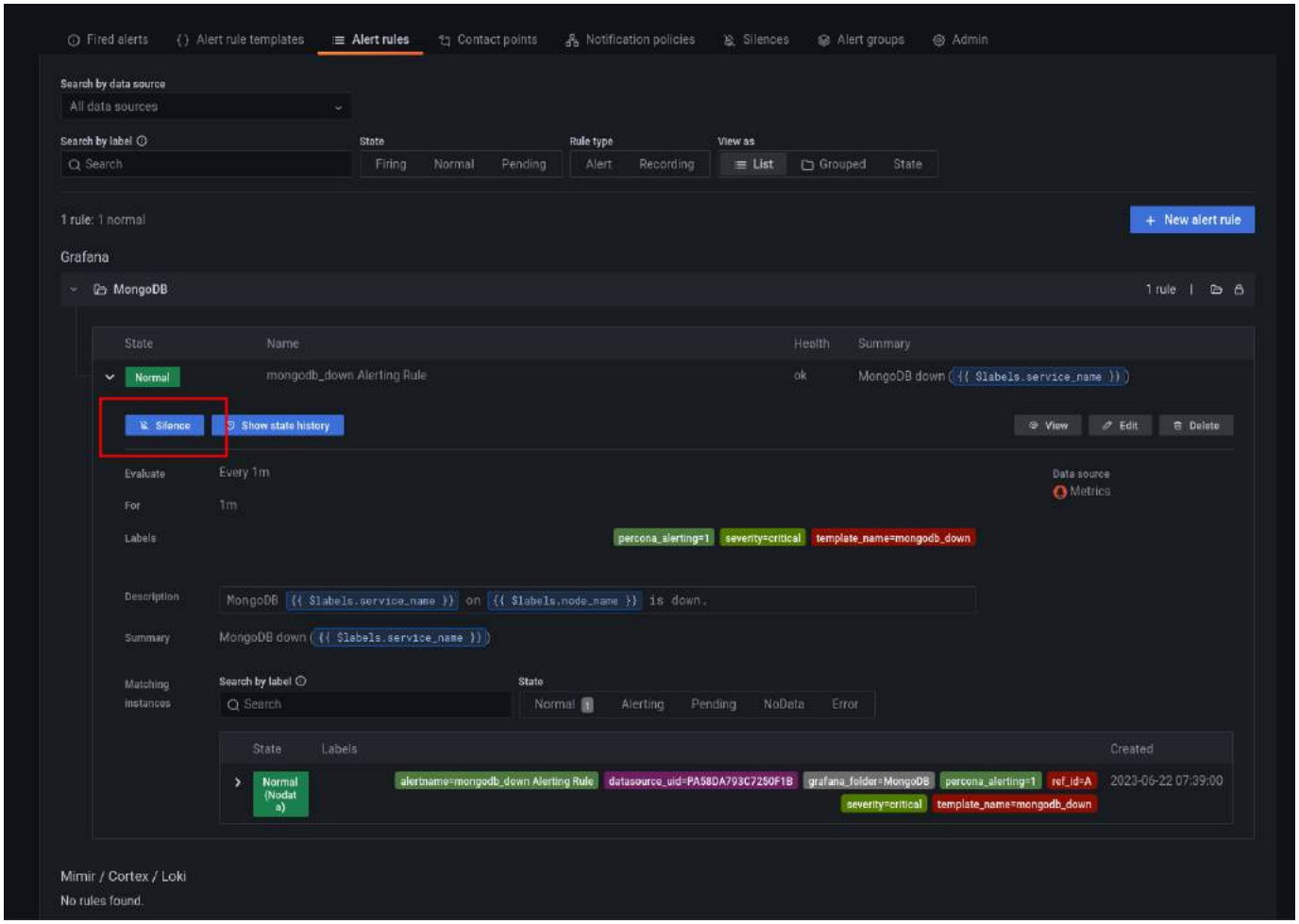
Percona LLC and/or its affiliates ©2024

During a silence, PMM continues to track metrics but does not trigger alerts or send notifications to any specified contact points. Once the silence expires alerts and notifications will resume.

Silenced alerts are still recorded under **Alerting > Fired Alerts** so that you can review them later. Silenced alerts show up as **Surpressed** and are disabled for as long as it's specified in the **Silence Duration**, or until you remove a silence.

USING SILENCES

You can silence an alert from the **Fired alerts** page or from the **Alert rules** page by expanding the Alert Rule and clicking the *Silence* button.



You can also create a silence from the **Silences** page, but here you would also need to define labels that match the alert that you want to silence.

To create a new silence from the **Silences** page:

1. Click the **New Silence** button.
2. Select the start and end date to indicate when the silence should go into effect and expire.
3. Optionally, update the duration to alter the time for the end of silence in the previous step to correspond to the start plus the duration.
4. Enter one or more matching labels by filling out the **Name** and **Value** fields. Matchers determine which rules the silence will apply to. Note that all labels specified here must be matched by an alert for it to be silenced.
5. Enter any additional comments you would like about this silence - by default, the date the silence was created is placed here.

Percona LLC and/or its affiliates ©2024

6. Review the affected alert instances that will be silenced.
7. Click **Submit** to create the silence.

For more information on working with silences, see About alerting silences in the Grafana documentation.

Alerting compatibility

TEMPLATE COMPATIBILITY WITH PREVIOUS PMM VERSIONS

If you have used Integrated Alerting in previous PMM versions, your custom alert rule templates will be automatically migrated to PMM 2.31. After upgrading to this new version, you will find all your alert templates under **Alerting > Alert Templates**.

If you have any templates available in the `/srv/ia/templates` folder, make sure to transfer them to `/srv/alerting/templates` as PMM 2.31 and later will look for custom templates in this location.

If you are upgrading from PMM 2.25 and earlier, alert templates will not be automatically migrated. This is because PMM 2.26.0 introduced significant changes to the core structure of rule templates.

In this scenario, you will need to manually recreate any custom rule templates that you want to transfer to PMM 2.26.0 or later.

TEMPLATE COMPATIBILITY WITH OTHER ALERTING TOOLS

If you have existing YAML alert templates that you want to leverage in Percona Alerting:

1. Go to **Alerting > Alert Rule Templates** tab and click **Add** at the top right-hand side of the table.
2. Click **Add** and upload a local .yaml file that contains the definition of one or more alert templates. Alert templates added in bulk will be displayed individually on **Alert rule templates** page.

MIGRATE ALERT RULES

Alert rules created with Integrated Alerting in PMM 2.30 and earlier are not automatically migrated to Percona Alerting.

After upgrading to PMM 2.31, make sure to manually migrate any alert rules that you want to transfer to PMM 2.31 using the Integrated Alerting Migration Script.

Script commands

The default command for migrating rules is:

```
python3 ia_migration.py -u admin -p admin
```

To see all the available options, check the scrip help using `ia_migration.py -h`

Script prerequisites

- Python version 3.x, which you can download from Python Downloads centre.
- Requests library, which you can install with the following command: `pip3 install requests`.

> **Important**
>
> The script sets all migrated alert rules to Active. Make sure to silence any alerts that should not be firing.

For more information about the script and advanced migration options, check out the help information embedded in the script.

Percona LLC and/or its affiliates ©2024

Disable Percona Alerting

If for some reason you want to disable PMM Alert templates and keep only Grafana-managed alerts:

1. Go to **Configuration > PMM Settings**.
2. Disable the **Alerting** option. The **Alerting** page will now display only Grafana-managed alert rules.

4.3.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-12-12

4.4 Backup and Restore

4.4.1 Back up and restore

Losing your data can destroy your business. This is why backing up data is critical for all database operations. Even more important than backing up data, is the ability to restore it in the event of data loss. PMM enables you to do all this with zero downtime and minimal performance impact.

Currently, PMM provides Backup and Restore functionality to work with:

- MongoDB (Generally Available)
- MySQL (in Technical Preview)

Enable the **Backup Management** option in PMM's Advanced Settings to activate the **Backup** page from where you can:

- Create and restore MongoDB and MySQL backups
- Automate backup scheduling
- Set retention policies
- Monitor your backup and restore activity

Supported setups

For MySQL databases, you can create and restore on-demand and scheduled physical backups. For MongoDB, you can create and restore physical, logical and Point-in-Time-Recovery (PITR) backups, both on-demand and scheduled.

SHARDED MONGODB CLUSTER CONFIGURATIONS

PMM 2.38 added support for creating backups of sharded MongoDB clusters. However, the restoring process is not handled end-to-end, and requires you to manually restore the artifacts using the CLI in Percona Backup for MongoDB.

For information on restoring sharded backups, check the PBM documentation

Percona LLC and/or its affiliates ©2024

Start here

To learn how to create and restore backups, check out subtopics below:

- Prepare a storage location
- MongoDB backups
- MySQL backups

Additional resources

Here are some external resources for learning more about databases backups:

- Amazon Web Services S3
- Percona Backup for MongoDB
- PERCONA_QPRESS
- PERCONA_XBCLOUD
- PERCONA_XBSTREAM
- PERCONA_XTRABACKUP
- oplog slices
- Percona Server for MongoDB
- MongoDB Replication

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-12-20

4.4.2 Prepare a storage location

Prepare a storage location as a backup destination for creating and storing your backup artifacts.

PMM supports the following types of storage:

- **Amazon S3-compatible**: enables you to use not only AWS S3, but also other storage solutions that support S3 API, like min.io.
- **Local storage**: currently only available for MongoDB backups.

Prepare a location for local backups

If you prefer storing your MongoDB backup artifacts on a remote filesystem, make sure that you've mounted the remote folder to all the mongoDB nodes on the same path, and that PBM tool has Write permissions on the path you define.

For more information, see the Percona Backup for MongoDB (PBM) documentation.

Percona LLC and/or its affiliates ©2024

Prepare a location for S3-compatible storage

If you want to store backup artifacts in S3-compatible storage, you can use Amazon S3, Min.io or any other storage solution with S3-compatible API.

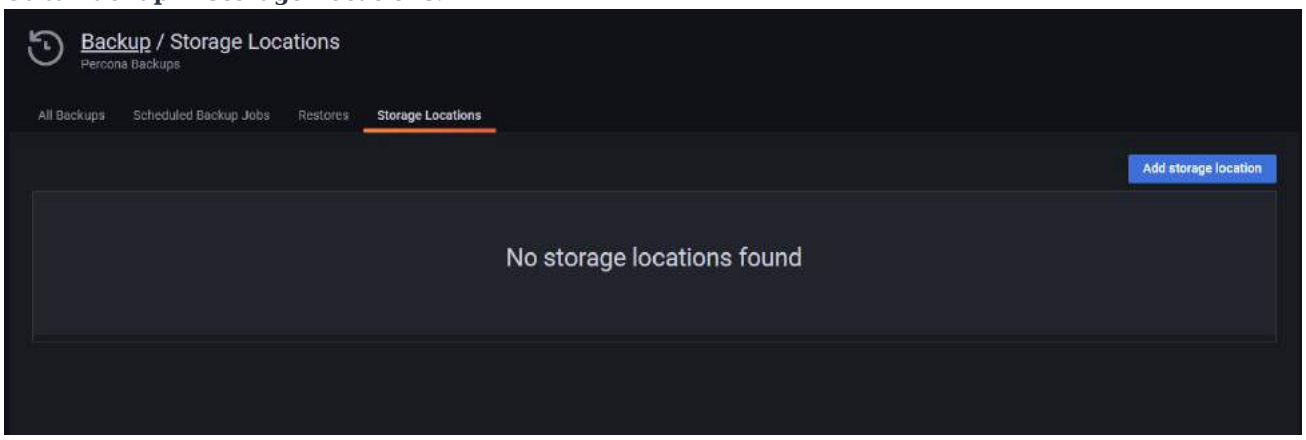
Before creating a cloud storage location for our future backups, make sure you have your S3-compatible storage ready. In addition to bucket location details, you will also need to ensure proper S3 permissions.

The general minimum permissions are **LIST/PUT/GET/DELETE**. A sample IAM policy is:

```
```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": "arn:aws:s3:::pmm-backup-testing"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:PutObjectAcl",
                "s3:GetObject",
                "s3:GetObjectAcl",
                "s3:DeleteObject"
            ],
            "Resource": "arn:aws:s3:::pmm-backup-testing/*"
        }
    ]
}
```
```

Create the storage location

1. Go to **Backup > Storage Locations**:



2. Click **Add storage location** and fill in a name and description for this new location.
3. Choose the type of storage location you are creating:
    - **S3**: Specify the S3-compatible backup location endpoint (URL), bucket name, and connection details.
    - **Local Client**: specify the path on your local client for files to be backed up to.

Percona LLC and/or its affiliates ©2024

Important

If your S3 endpoint is using a custom x509 certificate, PMM Server and all PMM Clients must be able to verify it. For this, append the CA certificate to the file containing the list of root CA certificates. These are stored on */etc/ssl/cert.pem* file inside the pmm-server container. On the Client side, it is usually */etc/ssl/certs/ca-bundle.crt*

1. Optionally, for S3-compatible storages, you can click **Test** to check the connection.
2. Click **Add** to create the location.

Specific target directories for backups

During backup creation, PMM enables you to set a specific folder within the local or S3-compatible location that you prepared following the instructions above.
Organizing backups in folders not only makes it easier to group backups for an entire cluster, but also improves PMM-PBM (Percona Backup for MongoDB) integration workflows.

The **Folder** field on the Create Backup pages is automatically populated with the value of the cluster label. You can change this default folder from PMM's Advanced Settings, but make sure you understand how your custom folder will impact PBM integration workflows.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-05-20

4.4.3 MongoDB backups

Supported setups for MongoDB backups

PMM supports the following actions for MongoDB backups.

REPLICA SET SETUPS

- Storing backups on Amazon S3-compatible object storage, and on mounted filesystem
- Creating and restoring Logical snapshot backups
- Creating and restoring Physical snapshot backups
- Creating logical PITR backups both locally and on S3-compatible object storage. Restoring logical PITR backups from S3-compatible object storage.

Percona LLC and/or its affiliates ©2024

SHARDED CLUSTERS

Backups of sharded clusters is supported starting with PMM 2.38. However, restoring for sharded cluster configurations is only supported from the CLI, and is handled via Percona Backup for MongoDB.

- Storing backups on Amazon S3-compatible object storage, and on mounted filesystem
- Creating Logical snapshot backups
- Creating Physical snapshot backups
- Creating logical PITR backups both locally and on S3-compatible object storage

For a detailed overview of the supported setups for MongoDB, check out the Support matrix.

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-07-03

MongoDB backup prerequisites

Before creating MongoDB backups, make sure to:

1. Check that **Backup Management** is enabled and the Backup option is available on the side menu. If Backup Management has been disabled on your instance, go to **Configuration > PMM Settings > Advanced Settings**, re-enable **Backup Management** then click **Apply changes**.
2. Prepare and create a storage location for your backups.
3. Check that PMM Client is installed and running on all MongoDB nodes in the cluster.
4. Check that Percona Backup for MongoDB (PBM) is installed and `pbm-agent` is running on all MongoDB nodes in the replica set. Make sure to configure the MongoDB connection URI for pbm-agent on all nodes.
5. Check that installed **mongod** binary is added to **PATH** variable of the user under which PMM client is running, and that **mongod** is controlled as a service by **systemctl**. PMM only works with a single **mongod** installed on a node.
6. Check that your MongoDB Services are managed as clusters in PMM. Go to **PMM Inventory > Services** page, expand the **Details** section on the **Options** column, and make sure that all the services in the table specify a cluster name. Services that do not specify a cluster name should be removed and re-added using commands like the following:

   ```
   pmm-admin add mongodb \
      --username=pmm_mongodb --password=password \
      query-source=profiler --cluster=mycluster
   ```

7. Check that MongoDB nodes are members of replica set.
8. Check that you set the required permissions for creating and restoring MongoDB backups.
9. Verify the MongoDB supported configurations and limitations.

Percona LLC and/or its affiliates ©2024

Important

Use `pbm` in manual mode only for restoring sharded cluster backups or other operations that can only be completed via the PBM CLI! Since PMM takes care of the PBM configuration, any unnecessary manual intervention can break the state.

PMM 2.32 and later require PBM 2.0.1 or newer.

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

Create MongoDB on-demand and scheduled backups

Before creating a backup, make sure to check the MongoDB backup prerequisites.

To schedule or create an on-demand backup, check the instructions below. If you want to create a Point-in-time-recovery (PITR) backup instead, see Create MongoDB PITR backups.

1. Go to **Backup > All Backups**.
2. Click **Create Backup**.
3. In the **Create Scheduled backup** window, select whether you want to create an **On Demand** or a **Schedule Backup**.
4. Enter a unique name for the backup.
5. Choose the service to back up from the **Service name** drop-down menu. This automatically populates the **DB Technology** field.
6. Select whether you want to create a **Physical** or **Logical** backup of your data, depending on your use case and requirements.
7. Choose a storage location for the backup. MongoDB supports both Amazon S3-compatible and local storage. If no options are available here, see the Create a storage location topic.
8. Specify the backup type, the schedule, and a retention policy for your backup:
   - **Backup Type**: select **Full**. If you want to create a PITR backup instead, see the Create MongoDB PITR backups topic
   - **Schedule**: if you're creating a scheduled backup, configure its frequency and start time.

Important

Percona LLC and/or its affiliates ©2024

Make sure that the schedule you specify here does not create overlapping jobs or overhead on the production environment. Also, check that your specified schedule does not overlap with production hours.

- **Retention**: this option is only available for snapshot backups stored on S3-compatible storage. If you want to keep an unlimited number of backup artifacts, type `0`.
- Expand **Advanced Settings** to specify the settings for retrying the backup in case of any issues. You can either let PMM retry the backup again (**Auto**), or do it again yourself (**Manual**). Auto-retry mode enables you to select up to ten retries and an interval of up to eight hours between retries.
- In the **Folder** field, check the target directory available for the specified service and location. By default, this field is prefilled with the cluster label to ensure that all the backups for a cluster are stored in the same directory. If the field is not automatically populated, the service you have specified is not member of a cluster and should be re-added using the following set of commands:

```
pmm-admin add mongodb \
   --username=pmm_mongodb --password=password \
   query-source=profiler --cluster=mycluster
```

Important

Unless you are using verified custom workflows, make sure to keep the default **Folder** value coming from the cluster name. Editing this field will impact PMM-PBM integration workflows.

a. To start creating the backup artifact, click **Backup** or **Schedule** at the top of the window, depending on whether you are creating a scheduled or an on-demand backup.

b. Go to the **All Backups** tab, and check the **Status** column. An animated ellipsis indicator ●●● shows that a backup is currently being created.

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-07-03

Create MongoDB PITR backups

Point-in-Time Recovery (PITR) restores databases up to a specific moment in time. PITR includes restoring the data from a backup snapshot and replaying all events that occurred to this data up to a specified moment from oplog slices.

Point-in-Time Recovery helps you prevent data loss during a disaster such as crashed database, accidental data deletion or drop of tables, or unwanted update of multiple fields instead of a single one.

COMPATIBILITY WITH PERCONA BACKUP FOR MONGODB

PMM introduced the option to create PITR Backups for MongoDB in version 2.23, as part of the larger Backup Management feature. This implementation in PMM uses Percona Backup for MongoDB (pbm) behind the scenes.

Percona LLC and/or its affiliates ©2024

Percona Backup for MongoDB is a distributed, low-impact solution for achieving consistent backups of MongoDB sharded clusters and replica sets. Starting with PMM 2.32, restoring PITR backups is available for backups based on pbm ≤ 2.0.1. To restore PITR backups, make sure you have pbm ≥ 2.0.1 installed.

Percona Backup for MongoDB supports Percona Server for MongoDB and MongoDB Community ≤ 3.6, with MongoDB Replication enabled. For more information, see the Percona Backup for MongoDB documentation.

HOW DOES IT WORK?

When point-in-time recovery (PITR) is enabled, pbm-agent periodically saves consecutive slices of the oplog.

To start saving oplog, PBM requires a backup snapshot. Such snapshots are created when you activate a PITR-scheduled task in PMM.

Since PBM saves oplog slices and streams them into your storage between scheduled task runs, scheduling frequent PITR backups is not necessary. You can use the available oplog slices in your storage to restore a backup to any moment between snapshots.

Before creating a backup, make sure to check the MongoDB backup prerequisites.

1. Go to **Backup > All Backups**.
2. Click **Create Backup**.
3. Select the **Schedule Backup** option in the **Create Scheduled backup** window.
4. Enter a unique name for this backup.
5. Choose the service to back up from the **Service name** drop-down menu. This automatically populates the **DB Technology** field.
6. Select **Logical** as this is the only data model that currently supports PITR backups.
7. Choose a storage location for the backup. MongoDB supports both Amazon S3-compatible and local storage. However, restoring from local storage is not supported yet. If no options are available here, see the Create a storage location topic.
8. Specify the backup type and the schedule for your backup:
   - **Backup Type**: select the **PITR** option.
   - **Schedule**: configure the frequency and the start time for this backup.

   > **Important**
   >
   > Make sure that the schedule you specify here does not create overlapping jobs or overhead on the production environment. Also, check that your specified schedule does not overlap with production hours.

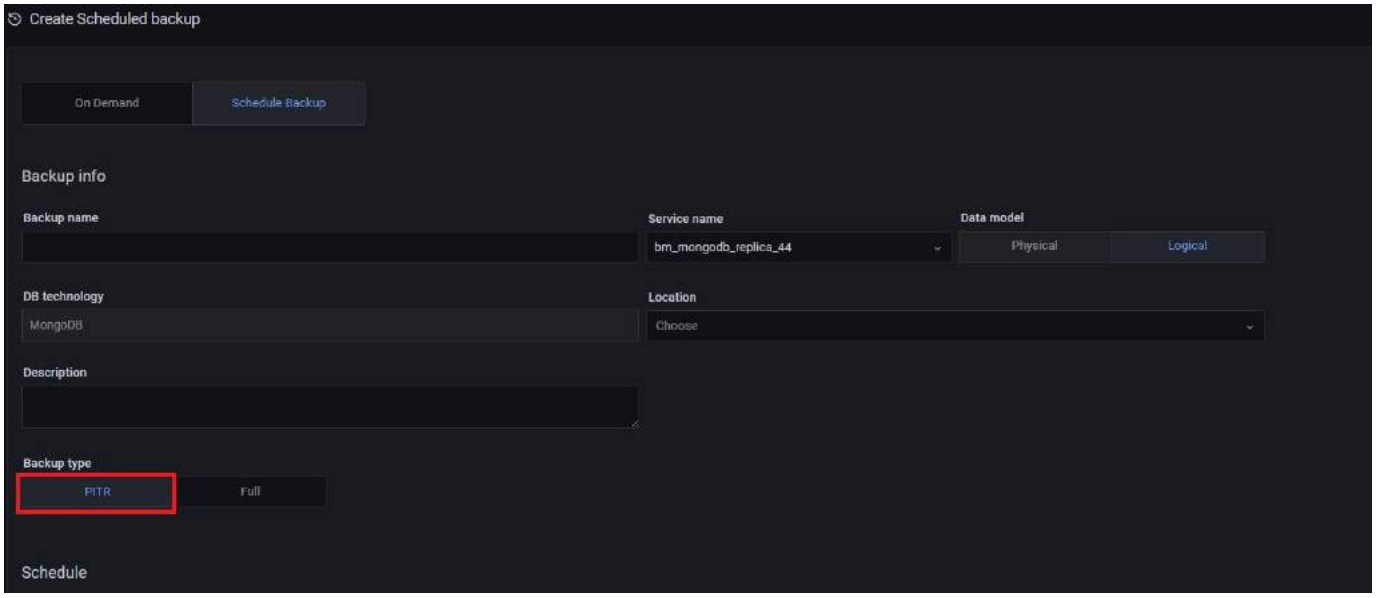
   - **Retention**: this option is not available for PITR backups. Currently, retention policies can only be specified for Snapshot backups stored on Amazon S3-compatible storage.
9. Expand **Advanced Settings** to specify the settings for retrying the backup in case of any issues. You can either let PMM retry the backup again (**Auto**), or do it again yourself (**Manual**).
   Auto-retry mode enables you to select up to ten retries and an interval of up to eight hours between retries.
10. In the **Folder** field, check the target directory available for the specified service and location. By default, this field comes prefilled with the cluster label to ensure that all the backups for a cluster are stored in the same directory. If the field is not automatically populated, the service you have specified is not member of a cluster and should be re-added using the following set of commands:

```
pmm-admin add mongodb \
   --username=pmm_mongodb --password=password \
   query-source=profiler --cluster=mycluster
```


Percona LLC and/or its affiliates ©2024

> **Important**
>
> Unless you are using verified custom workflows, make sure to keep the default **Folder** value coming from the cluster name. Editing this field will impact PMM-PBM integration workflows.

11. Click **Schedule** to start creating the backup artifact.
12. Go to the **All Backups** tab, and check the **Status** column. An animated ellipsis indicator ●●● shows that a backup is currently being created.

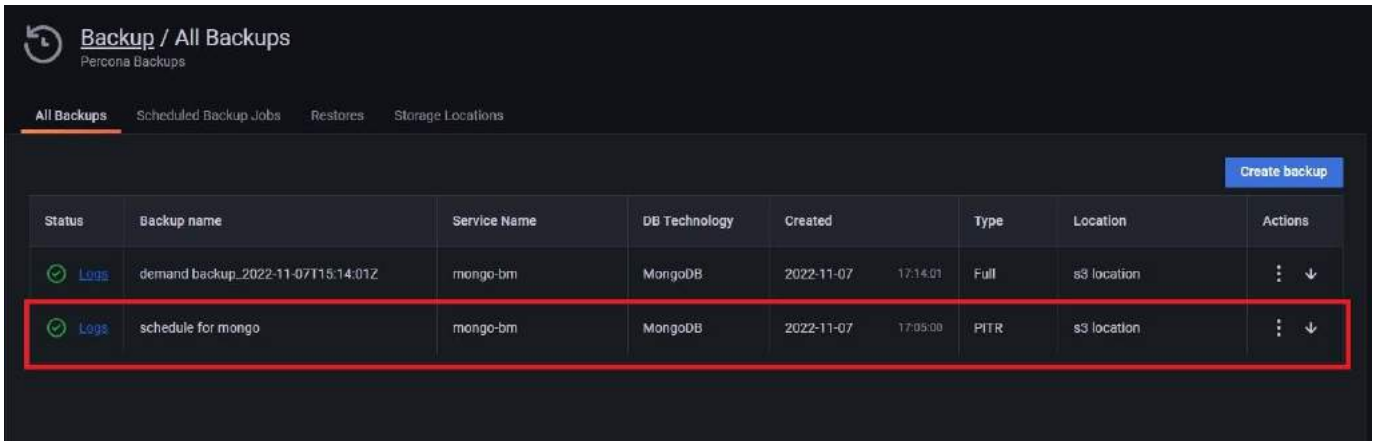


FAILED BACKUP ALERTS

If you want to be notified of any MongoDB backups that fail, you can create an alert based on the Backup Failed alert template. For information on working with alert templates, see the Percona Alerting topic.

PITR ARTIFACTS

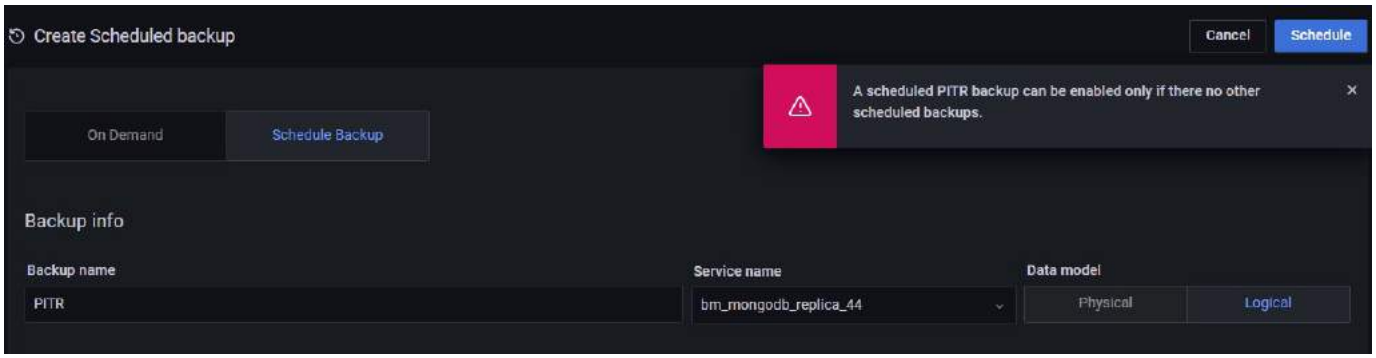
The PITR oplog is available a few minutes (10 by default) after your PITR job has run for the first time. To see the corresponding PITR artifact, check out the list under **Backup > All Backups**.



PITR AND OTHER SCHEDULED BACKUPS

Make sure to disable any other scheduled backup jobs before creating a PITR backup. PMM displays an error message if you try to enable PITR while other scheduled backup jobs are active:

Percona LLC and/or its affiliates ©2024



This constraint applies at the service level. You can still have PITR enabled for one service while having regular scheduled backup jobs for other services.

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-10-05

Restore a MongoDB backup

MongoDB backups can only be restored to the same service they were created from.

To restore a backup:

1. Go to **Backup → All backups** and find the backup that you want to restore.
2. Click the arrow in the **Actions** column to check all the information for the backup, then click **Restore from backup**. This opens the **Restore from backup** dialog, with the **Same service** option automatically preselected. This is because, currently, MongoDB backups can only be restored to a service with identical properties.
3. If you are restoring a PITR backup, select the point for the date and time that you want to restore the database to.
4. Click **Restore** then go to the **Restores** tab to check the status of the restored backup.

Important

During restoring, PMM disables all the scheduled backup tasks for the current service. Remember to re-enable them manually after the restore.

RESTORE TO A NEW CLUSTER MANUALLY

1. Install MongoDB and Percona Backup for MongoDB. Pay attention to the versions. To minimize potential incompatibility, use the same versions that were used for taking backups. For instructions, see the PBM install documentation.

Percona LLC and/or its affiliates ©2024

2. Configure your environment:

   - to restore to a new environment with the same replica set name, make sure that the replica set name in your new destination cluster use the same name as that in the cluster that was backed up.
     For more information, see Restoring a backup into a new-environment in the PBM documentation.

   - to restore **logical backups** to a new environment that has a different replica set name, configure the name mapping between the source and target environments.
     For the new environment, you can either set the **PBM_REPLSET_REMAPPING** environment variable for pbm CLI, or use the `--replset-remapping` flag for PBM commands.

     The mapping format is `<rsTarget>=<rsSource>`.

     For example:

     ```
     $ export PBM_REPLSET_REMAPPING="targetRS=sourceRS"
     ```

     OR

     ```
     $ pbm restore <timestamp> --replset-remapping="targetRS=sourceRS"
     ```

     For more information, see Restoring into a replica set with a different name in the PBM documentation.

3. Make sure that Percona Backup for MongoDB configuration in the new environment points to the remote storage defined for the original environment, including the authentication credentials for object stores.

   The easiest way to configure it is to create a config file, called, for example, `pbm_config.yaml`.

   For this, you can either copy the config from the source host or create a new one.

   To redirect the config output from the existing environment, use:

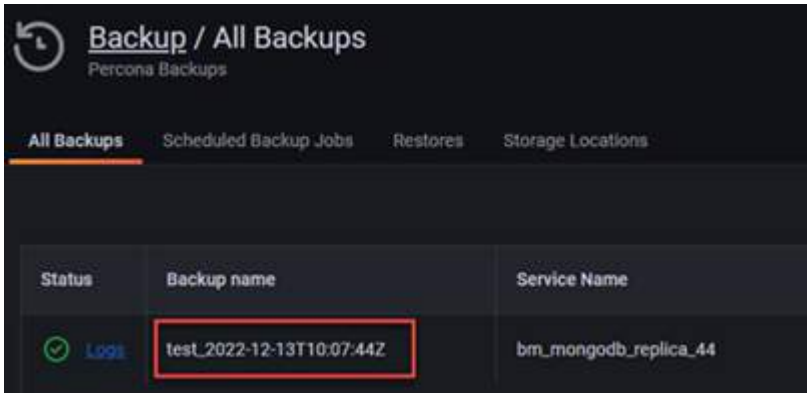
   ```
   pbm config >> pbm_config.yaml
   ```

   then copy the resulting file to the new environment.

   Here's an example of config file content for AWS S3-compatible storage:

   ```
   storage:
     type: s3
     s3:
       region: us-west-2
       bucket: pbm-test-bucket
       prefix: backup_name_from_pmm
       credentials:
         access-key-id: <your-access-key-id-here>
         secret-access-key: <your-secret-key-here>
   ```

   The prefix name is the artifact name that appears in the **Backup name** column, under **Backup > All Backups** page:

   To implement the config, use the following command:

   ```
   pbm config --file pbm_config.yaml
   ```


Percona LLC and/or its affiliates ©2024

For more information, see **Restoring a backup into a new-environment** in the PBM documentation.

4. Run `pbm list` to check if pbm is ready to perform the restore procedure.
5. Once all the backups made from the original environment are available, run the restore command:

   - For snapshot backups:

     a) run the following command:

     ```
     pbm list
       Backup snapshots: 2022-11-23T19:40:06Z [restore_to_time: 2021-01-13T15:53:40Z]
     ```

     b) provide the timestamp of the backup to the `pbm` command:

     `pbm restore 2022-11-23T19:40:06Z`

     For more information, see Restore a backup topic in the PBM documentation.

   - For PITR backups:

     a) run the following command:

     ```
     pbm list
     Backup snapshots:
       2022-11-23T19:40:06Z <logical> [restore_to_time: 2022-11-23T19:40:25Z]
       2022-11-23T19:45:07Z <logical> [restore_to_time: 2022-11-23T19:45:22Z]
     PITR <on>:
       2022-11-23T19:40:26Z - 2022-11-23T19:45:22Z
     ```

     b) provide the timestamp from one of the PITR ranges to the `pbm` command:

     `pbm restore --time="2022-11-23T19:40:26`

   For more information, see the Point-in-time Recovery topic in the PBM documentation.

6. Check the progress of the restore operation, using one of the commands below:

   - For logical restores: `pbm describe-restore <restore_name>`
   - For physical restores: `pbm describe-restore --config=/path/to/pbm_config.yaml <restore_name>`

   Required arguments:

   - PBM generates the `<restore_name>` information after you start the restoring.
   - The *pbm_config.yaml* file required for physical restores is the PBM config file that you provided for step 3.

Important

Make sure not to run pbm backup from the new environment while the Percona Backup for MongoDB config is pointing to the remote storage location of the original environment.

Restoring from a sharded cluster

Sharded cluster backups are supported starting with PMM 2.38 and PMM handles the backup process end-to-end. However, restoring such artifacts is currently possible only via the CLI, using Percona Backup for MongoDB.

For information on restoring sharded backups, check the PBM documentation

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2024-01-11

MongoDB Backup and Restore support matrix

Creating and restoring MongoDB backups in PMM currently has the following limitations and requirements:

- Physical backups and restores are supported only for **Percona Server for MongoDB**.
- Physical restores are not supported for deployments with arbiter nodes. For more information, see the Percona Backup for MongoDB documentation.
- Creating backups for sharded clusters was included in PMM 2.38 and is available straight from the UI. However, restoring these backup artifacts is only possible via the CLI, using Percona Backup for MongoDB. For information on restoring sharded backups, check the PBM documentation.
- Retention policy is supported only for snapshot types of scheduled backups and for the S3-compatible storage type.
- Before restoring, make sure to prevent clients from accessing the database.

SUPPORT MATRIX

BACKUP: LOGICAL

Full or PITR	Storage type (S3 or Local)	Support level
PITR	S3	**Full**
PITR	Local	**Full**
Full	S3	**Full**
Full	Local	**Full**

BACKUP: PHYSICAL

Full or PITR	Storage type (S3 or Local)	Support level
PITR	S3	**No**
PITR	Local	**No**
Full	S3	**Full**
Full	Local	**Full**

RESTORE: LOGICAL

Full or PITR	Storage type (S3 or Local)	Support level
PITR	S3	**Full**
PITR	Local	**No**
Full	S3	**Full**
Full	Local	**Full**

RESTORE: PHYSICAL

Full or PITR	Storage type (S3 or Local)	Support level
PITR	S3	**No**
PITR	Local	**No**
Full	S3	**Partial***

Percona LLC and/or its affiliates ©2024

Full or PITR	Storage type (S3 or Local)	Support level
Full	Local	**Partial***

* Partial support for non-containerized deployments and NO support for containerized deployments.

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-08-14

4.4.4 MySQL backups

Supported setups for MySQL backups

Important

MySQL backup functionality is still in Technical Preview.

PMM supports MySQL database server for:

- Creating and restoring physical backups
- Storing backups to Amazon S3-compatible object storage

BACKING UP MYSQL DATABASES HOSTED IN DOCKER CONTAINER

To ensure PMM can correctly backup and restore databases from a MySQL Docker container, make sure that the container is compatible with systemd.

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-03-30

MySQL backup prerequisites

Before creating MySQL backups, make sure to:

1. Check that **Backup Management** is enabled and the Backup option is available on the side menu. If Backup Managemt has been disabled on your instance, go to **Configuration > PMM Settings > Advanced Settings**, re-enable **Backup Management** then click **Apply changes**. !!! caution alert alert-warning

Percona LLC and/or its affiliates ©2024

"Important" If PMM Server runs as a Docker container, enable backup features at container creation time by adding `-e ENABLE_BACKUP_MANAGEMENT=1` to your `docker run` command.

2. Check that the PMM Client is installed and running on the node where the backup will be performed.
3. To enable Xtrabackup for MySQL 8.0+, check that pmm-agent connects to MySQL with a user that has BACKUP_ADMIN privilege.
4. Check that there is only one MySQL instance running on the node.
5. Verify that MySQL is running:
    - as a service via `systemd`;
    - with the name `mysql` or `mysqld` (to confirm, use `systemctl status mysql` or `systemctl status mysqld` respectively);
    - from a `mysql` system user account.
6. Make sure that there is a `mysql` system group.
7. Check that MySQL is using the `/var/lib/mysql` directory for database storage.
8. Make sure that `pmm-agent` has read/write permissions to the `/var/lib/mysql` directory.
9. Check that the latest versions of the following packages are installed and included in the `$PATH` environment variable:
    - `xtrabackup`, which includes:
        - `xbcloud`
        - `xbstream`
    - [`qpress`][PERCONA_QPRESS].

Important

Make sure that the versions of xtrabackup, xbcloud, xbstream, and qpress are fully compatible with the currently installed version of MySQL on the system.

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-06-04

Create a MySQL backup

To create a backup:

1. Go to **Backup > All Backups**.
2. Click **Create Backup**.
3. Specify the type of backup that you want to create: **On Demand** or **Schedule Backup**.
4. Enter a unique name for this backup.

Percona LLC and/or its affiliates ©2024

5. Choose the service to back up from the Service name drop-down menu. This automatically populates the **DB Technology** field and selects the **Physical** data model, as this is the only model available for MySQL backups.
6. Choose a storage location for the backup. MySQL currently only supports storing backups to Amazon S3. If no options are available here, see the Create a storage location topic section above.
7. If you're creating scheduled backups, also specify the backup type, the schedule, and a retention policy for your backup:
   - **Backup Type**: currently, PMM only supports **Full** backup types for MySQL.
   - **Schedule**: configure the frequency and the start time for this backup.

   > **Important**
   >
   > Make sure that the schedule you specify here does not create overlapping jobs or overhead on the production environment. Also check that your specified schedule does not overlap with production hours.

   - **Retention**: this option is only available for Snapshot backups stored on S3-compatible object storage. If you want to keep an unlimited number of backup artifacts, type `0`.
8. Leave the **Folder** field as is. This field is relevant for MongoDB backups to ensure compatibility with PBM wokflows and comes prefilled with the cluster label.
9. Expand **Advanced Settings** to specify the settings for retrying the backup in case of any issues. You can either let PMM retry the backup again (**Auto**), or do it again yourself (**Manual**). Auto retry mode enables you to select up to ten retries and an interval of up to eight hours between retries.
10. To start creating the backup artifact, click **Backup** or **Schedule** at the top of the window, depending on whether you are creating a scheduled or an on-demand backup.
11. Go to the **All Backups** tab, and check the **Status** column. An animated ellipsis indicator ●●● shows that a backup is currently being created.

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-07-03

Restore a MySQL backup

RESTORE COMPATIBILITY

MySQL backups can be restored to the same service it was created from, or to a compatible one.

To restore a backup:

1. Go to **Backup > All backups** and find the backup that you want to restore.
2. Click the three dots in the **Actions** column to check all the information for the backup, then click **Restore from backup**.
3. In the **Restore from backup** dialog, select **Same service** to restore to a service with identical properties or **Compatible services** to restore to a compatible service.
4. Select one of the available service names from the drop-down menu.

Percona LLC and/or its affiliates ©2024

5. Check the values, then click **Restore**.
6. Go to the **Restores** tab to check the status of the restored backup.

During restoring, PMM disables all the scheduled backup tasks for the current service. Remember to re-enable them manually after the restore.

GET EXPERT HELP

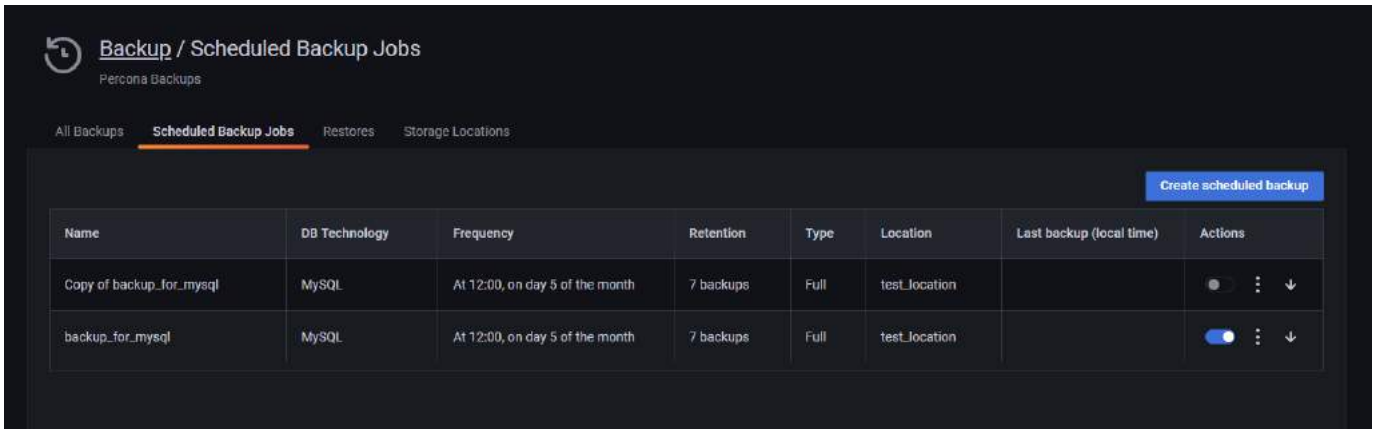
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

4.4.5 Edit a scheduled backup

1. Go to **Backup > Scheduled Backup Jobs**.
2. In the **Actions** column:
   - Click the switch to enable or disable the backup.
   - Click to edit, delete or create a copy (disabled by default) of the backup schedule.



Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

4.4.6 Delete a backup

You can only delete backup artifacts stored on Amazon S3-compatible. Local backups must be removed manually.

Percona LLC and/or its affiliates ©2024

To delete a backup:

1. Go to **Backup > All Backups** and find the row with the backup you want to delete.
2. Click the arrow in the **Actions** column to check all the information for the backup, then click **Delete backup**.
3. In the **Delete backup artifact** dialog box, enable **Delete from storage** if you also want to delete the actual backup content besides just the backup register.
4. Click **Delete**.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

4.5 Roles and permissions

4.5.1 Roles and permissions

Roles are the sets of permissions and configurations that determine which metrics a user can access.

Each PMM user is associated with a role that includes permissions. Permissions determine the privileges that a user has in PMM.

By creating roles, you can specify which data can be queried based on specific label criteria, for instance, allowing the QA team to view data related to test environments.

4.5.2 About Access Control

Caution

PMM Access Control is currently in technical preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

Access control in PMM allows you to manage who has access to individual Prometheus (Victoria Metrics) metrics based on **labels**. Thus, access management provides a standardized way of granting, changing, and revoking access to metrics based on the role assigned to the users.

The following topics are covered as part of access control:

- Configure access control
- Labels for access control
- Create access roles
- Use case

Percona LLC and/or its affiliates ©2024

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-02-23

4.5.3 Configure access control

You can configure access control in PMM as follows:

- Docker
- User Interface

Configure access control using Docker

To configure access roles in a `pmm-server` docker container, pass an additional environment variable `ENABLE_RBAC=1` when starting the container.

```
docker run ⋯ -e ENABLE_RBAC=1
```

For compose add an additional variable:

```
services:
  pmm-server:
    ⋯
    environment:
      ⋯
      ENABLE_RBAC=1
```

Configure access control from the UI

To configure access control from the UI, do the following:

From the main menu, navigate to *Configuration* → *Settings* → *Advanced Settings* → *Access Control* and click toggle.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

Percona LLC and/or its affiliates ©2024

4.5.4 Labels for access control

Label-based access control in PMM allows you to manage who has access to metrics based on labels. By creating roles, you can specify which data can be queried based on specific label criteria, for instance, allowing the QA team to view data related to test environments.

With Label-based access control, you can associate multiple labels with a role, ensuring only data from series that match your defined labels is returned.

Standard vs custom labels

PMM supports standard as well as custom labels. PMM automatically assigns standard labels. You can also set standard labels when an object (Node, Service, or Agent) is created. Custom labels are assigned and updated only by a user.

Examples

Label Type	Object	**Label name **	Example
Standard	Node	node_id	/node_id/123
	Service	service_type	- mysql, mongodb, postgresql etc.
Custom	Node, Service, Agent	Any string matching regular expression: [a-zA-Z_][a-zA-Z0-9_]*. Also, it cannot start with two underscores.	owner=”joe” _rack=”12345”

Adding labels

In PMM, labels enable you to identify, categorize, and organize your monitored services and resources. Labels also simplify filtering, grouping, and analyzing data.

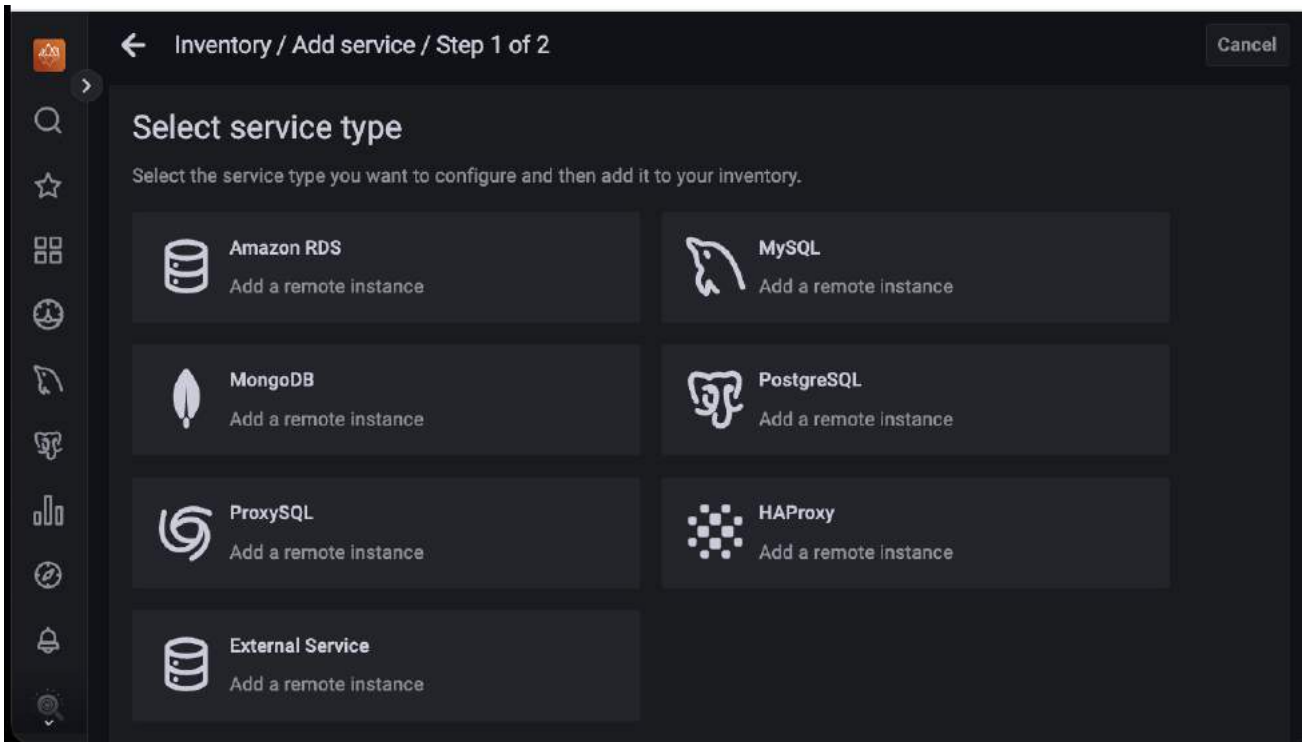
While adding a service for monitoring in PMM, you can add custom or standard labels.

Using PMM UI

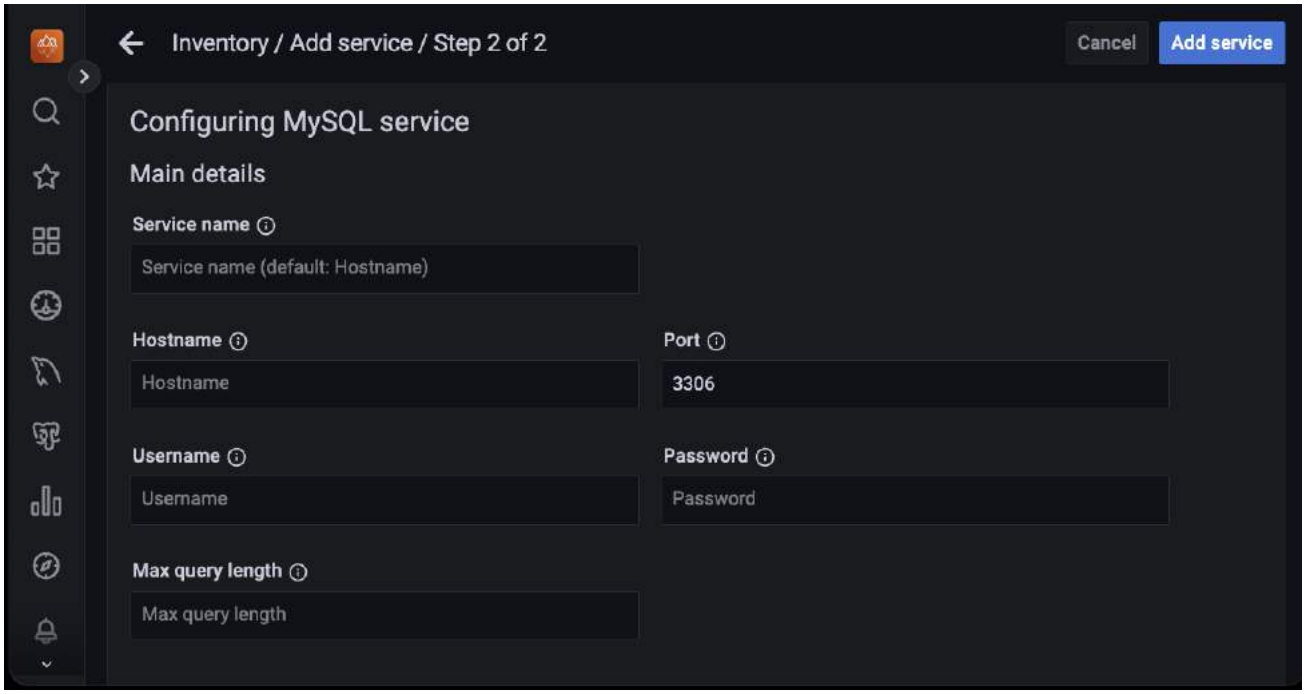
You can set the labels using the User interface as follows:

1. From the *Main* menu, navigate to *Configuration → Add Service*.
2. Select the service you want to add to PMM for monitoring. The page to add the service opens.

Percona LLC and/or its affiliates ©2024

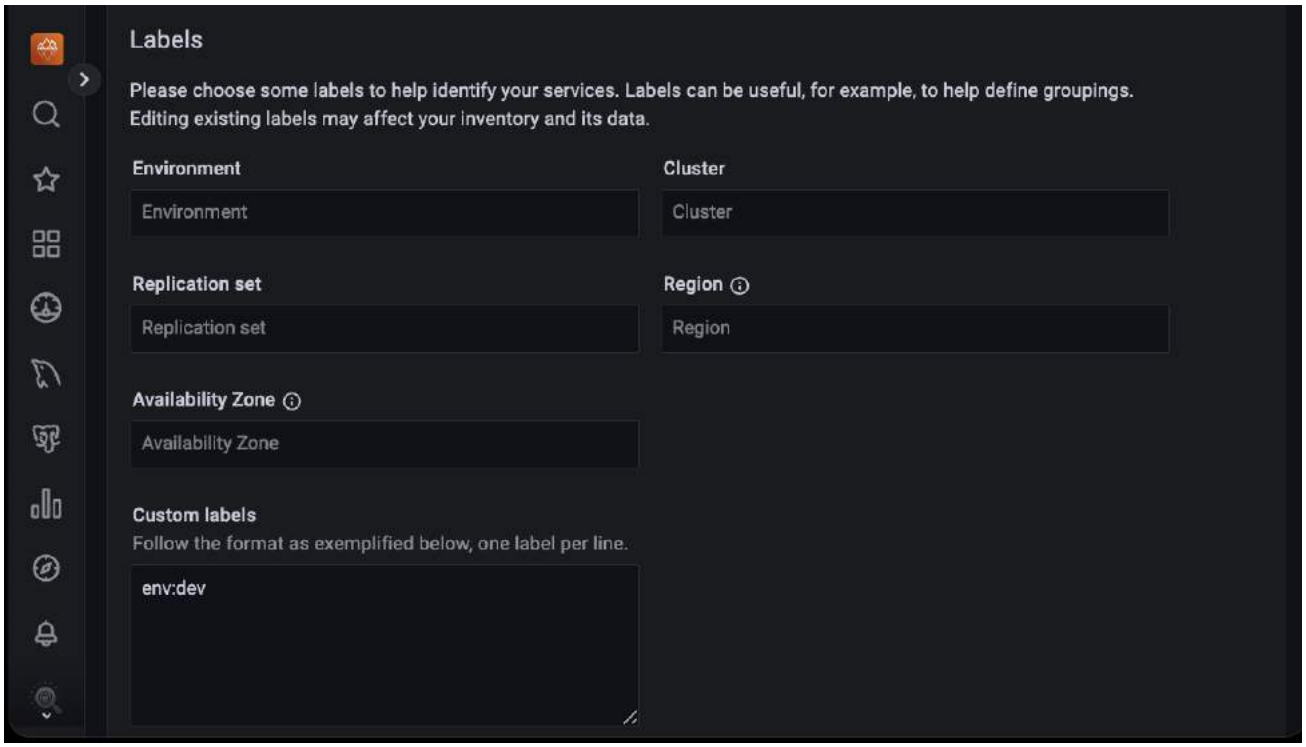


3. Enter the details such as **Hostname, Service name, Port, Username, Password, Max query length**.



4. (Optional) Enter the **Labels** for the service being monitored.

Percona LLC and/or its affiliates ©2024



5. (Optional) Check the **Additional Options** that you want to include, such as:

    - **Skip connection check**: Do not check the connection to the database.
    - **Use TLS for database connections**: Enabling TLS (Transport Layer Security) for database connections is an important security measure to protect the confidentiality and integrity of data transmitted between the PMM Server and the monitored database instances.
    - **Skip TLS certificate and hostname validation**: For certain purposes, like debugging and testing, it may be necessary to bypass TLS certificate and hostname validation.
    - **Table statistics limit**: Do not collect table statistics if the number of tables in a database exceeds this limit (defaults to 1000).
    - **Disable comments parsing**: Filter out comments in log files or data streams to isolate valid data.
    - **Use performance schema**: Use Performance Schema instead of Slow Query Log (default) for monitoring and diagnosing performance issues in your database.

You can also edit the labels for a service with the PMM user interface. For details on editing labels for a service, see Editing labels for a service

Using pmm-admin

You can also assign labels using pmm-admin.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2024-01-11

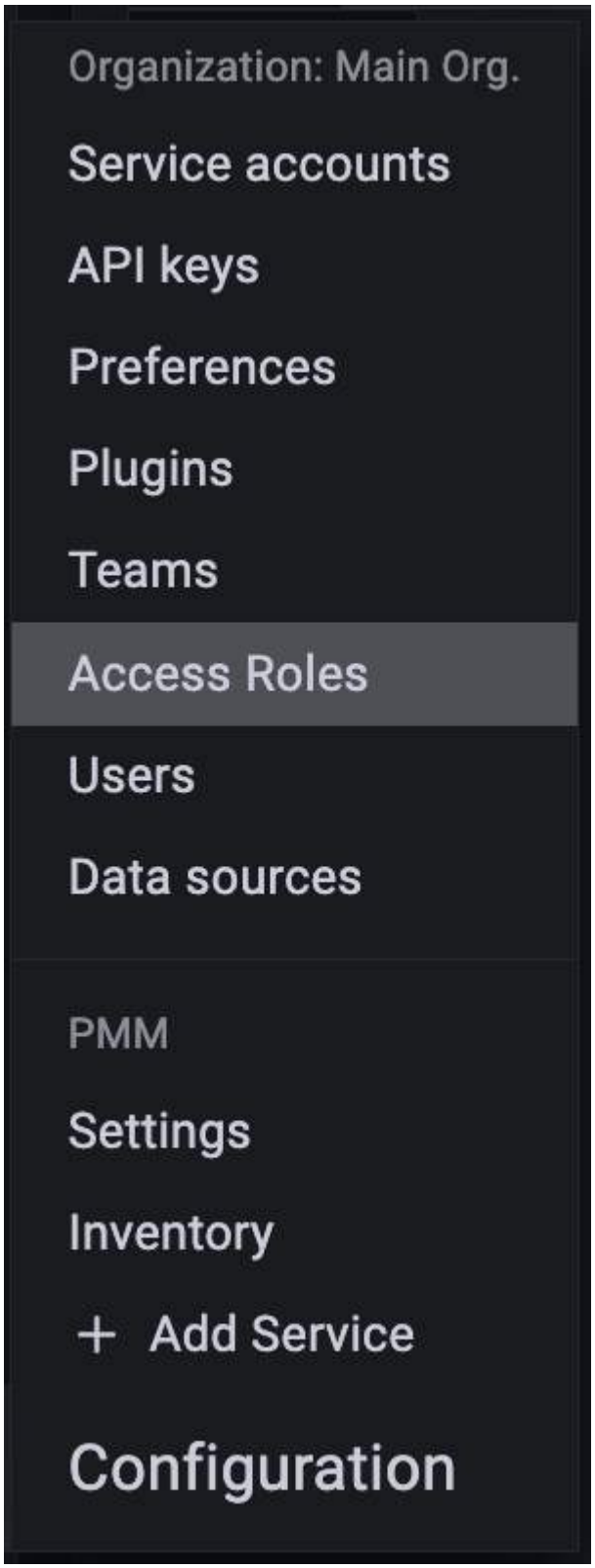
4.5.5 Create access roles

Roles are a vital part of Access control. Roles provide users with access to specific, role-based metrics.

To create access roles in PMM, do the following:

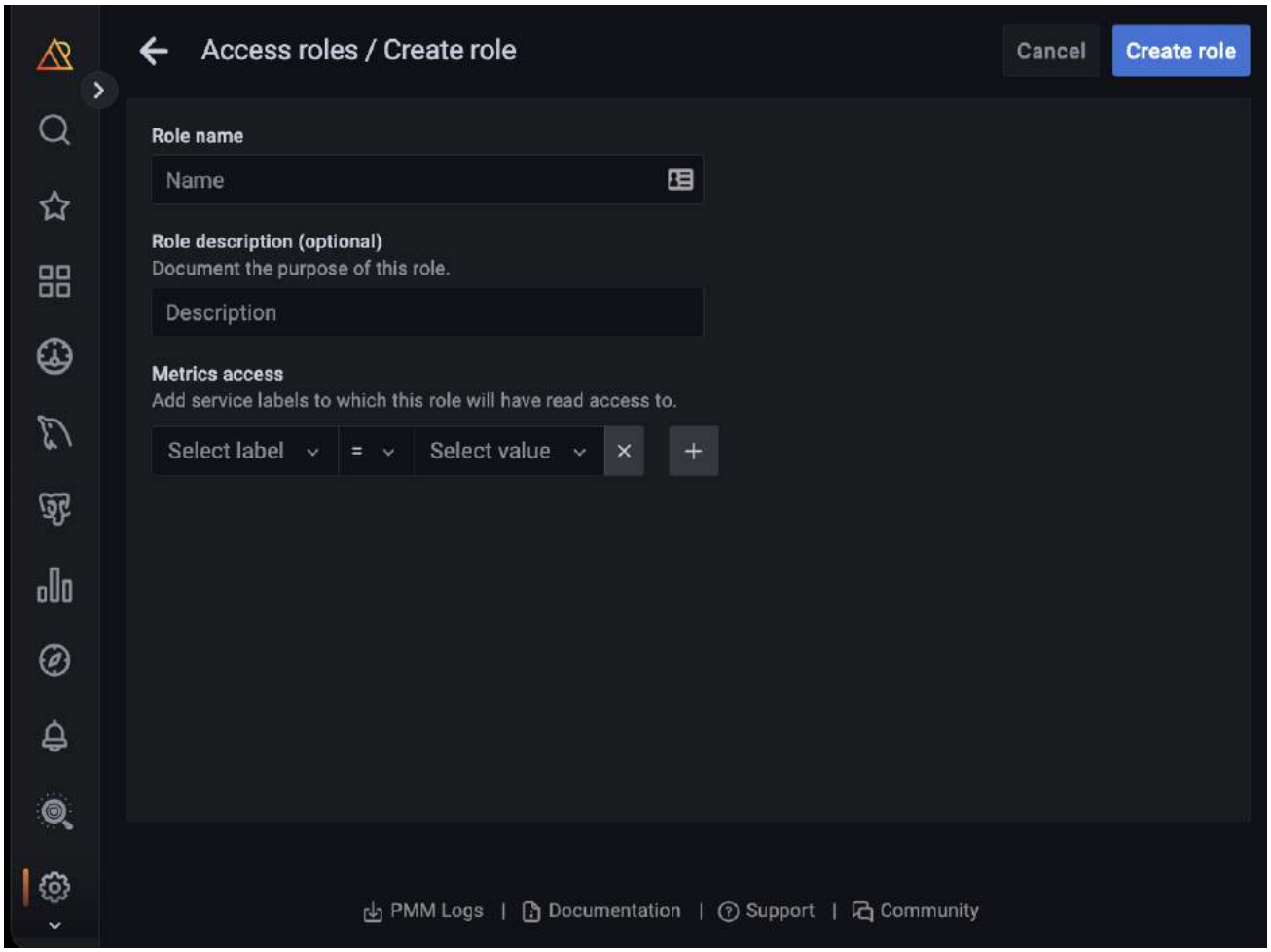
1. From the *Main* menu, navigate to *Configuration → Access Roles*. *Access Roles* tab opens.

Percona LLC and/or its affiliates ©2024



2. Click *Create*. Create role page opens.
3. Enter the Role name and Role description.

Percona LLC and/or its affiliates ©2024



4. Select the following from the drop-downs for metrics access:
   - Label
   - Operator
   - Value of the label.

   If you want to add more than one label for a role, click + and select the values from the drop-down.

   For information on how the Prometheus selectors work, see Prometheus selectors.
5. Click *Create* role.

Note

To create roles, you must have admin privileges. For more information, see Manage users.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2024-01-11

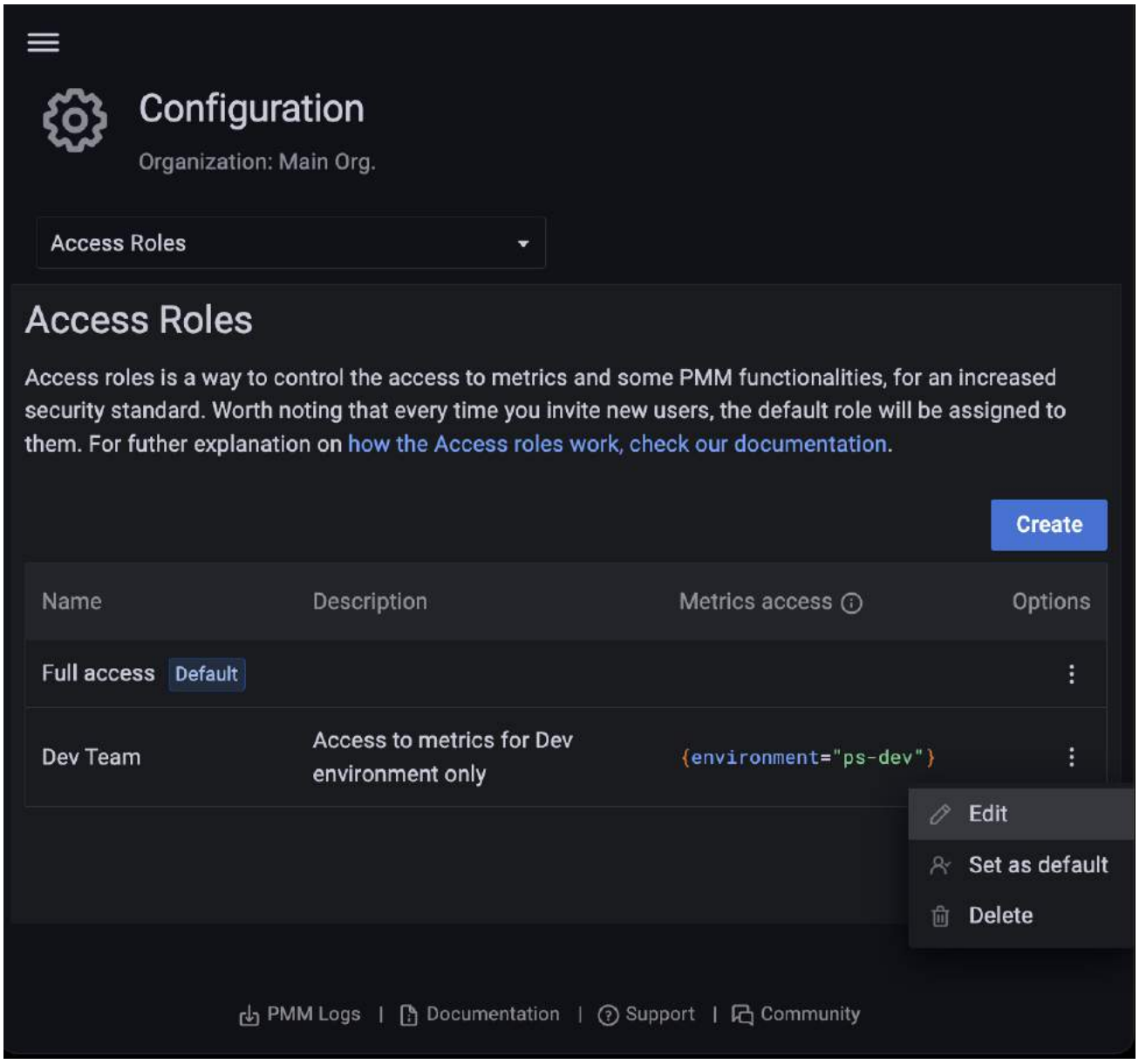
4.5.6 Manage access roles

You can manage roles in PMM by editing or deleting a role.

Edit roles

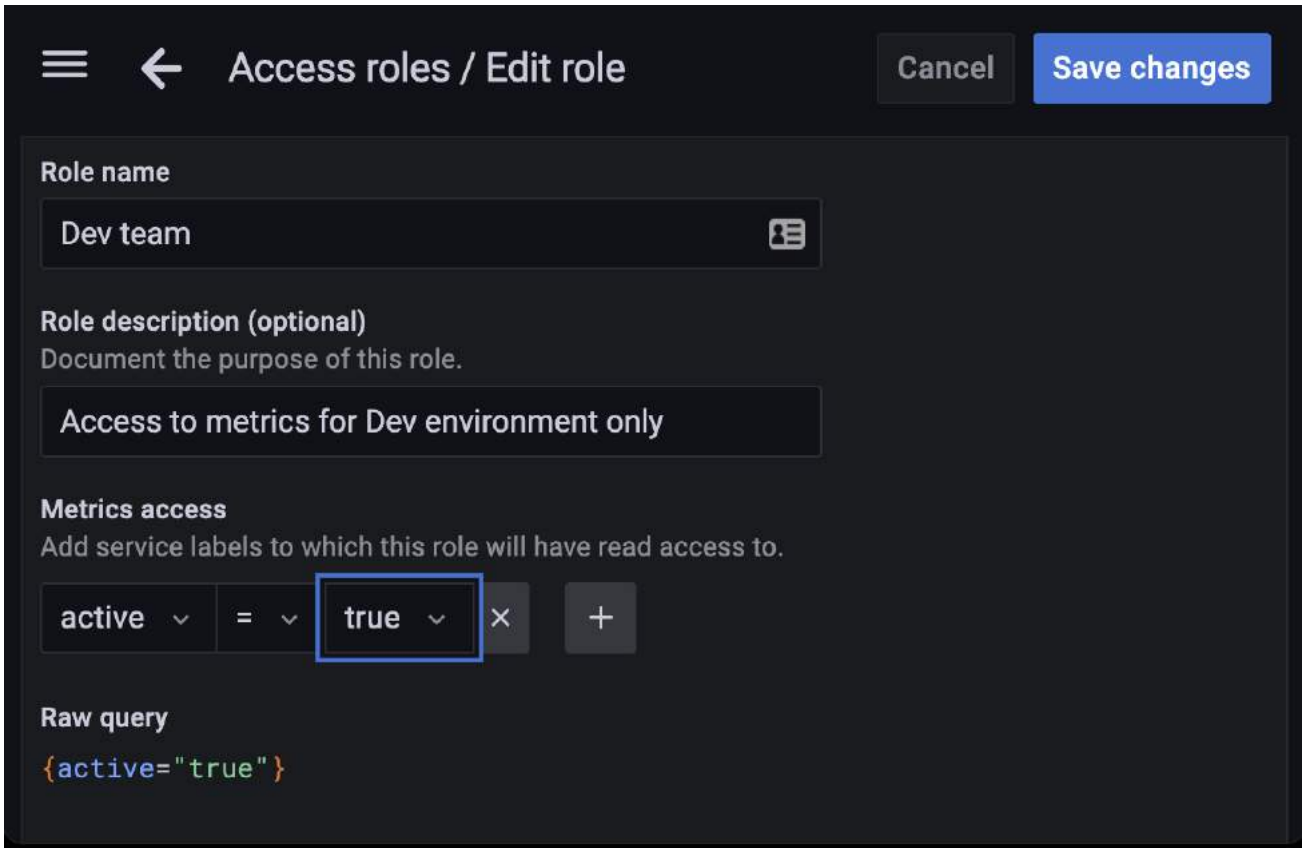
To edit access roles, do the following:

1. From the *Main* menu, navigate to *Configuration* → *Access Roles*. The *Access Roles* tab opens.
2. On the role you want to edit, click the *ellipsis (three vertical dots)* > *edit role* in the *Options* column. The *Edit* role page opens.



3. Make the required changes to the role.

Percona LLC and/or its affiliates ©2024



4. Click Save Changes.

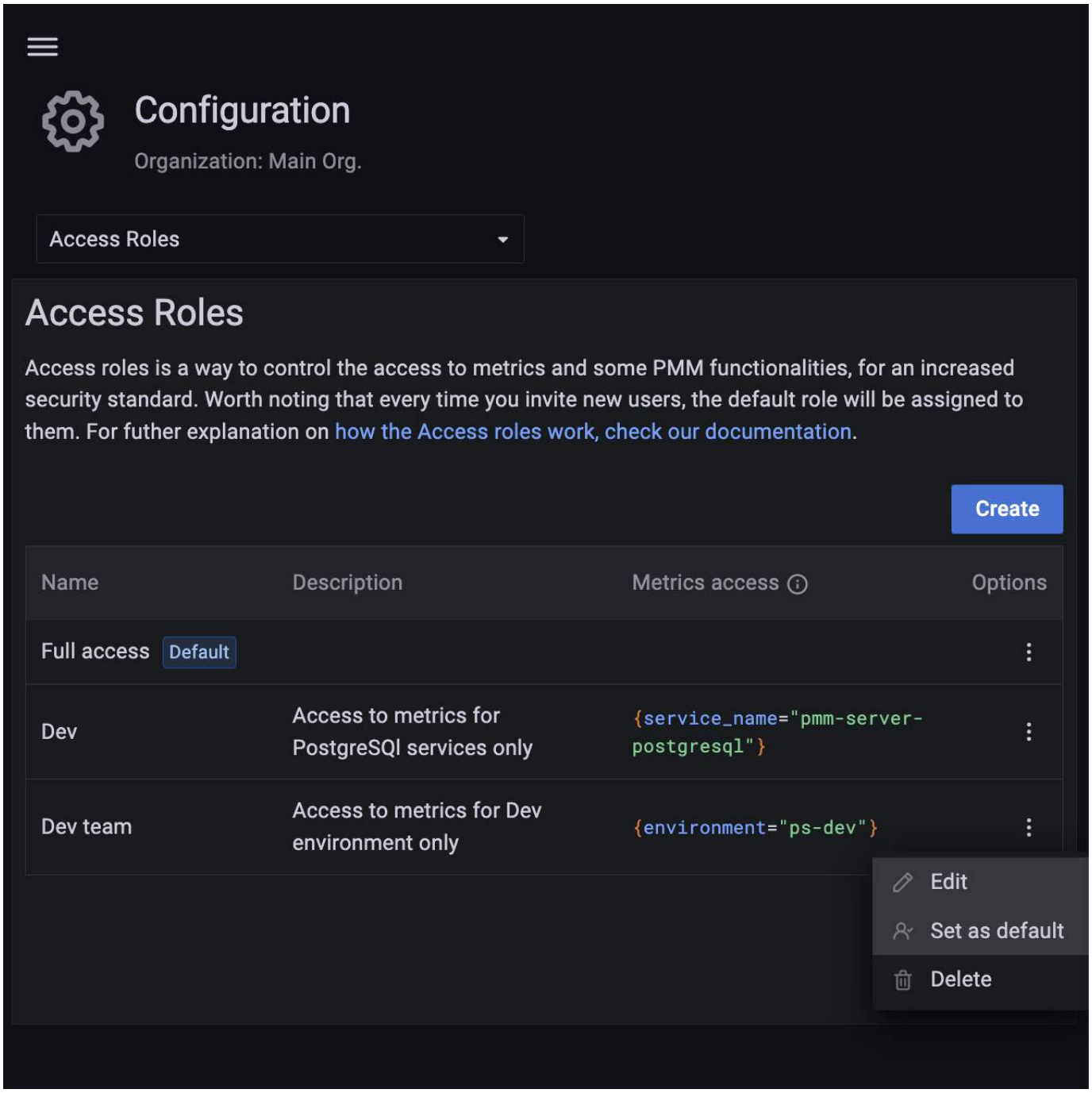
Set a role as default

When a user signs in to PMM for the first time and the user has no role assigned, the user is automatically assigned the *Default* role. For administrators, the default role provides a convenient way to configure default permissions for new users.

To set a role as default, do the following:

1. From the *Main* menu, navigate to *Configuration → Access Roles*. The *Access Roles* tab opens.
2. On the role you want to set as default, click the *ellipsis (three vertical dots) → set as default* in the *Options* column.

Percona LLC and/or its affiliates ©2024

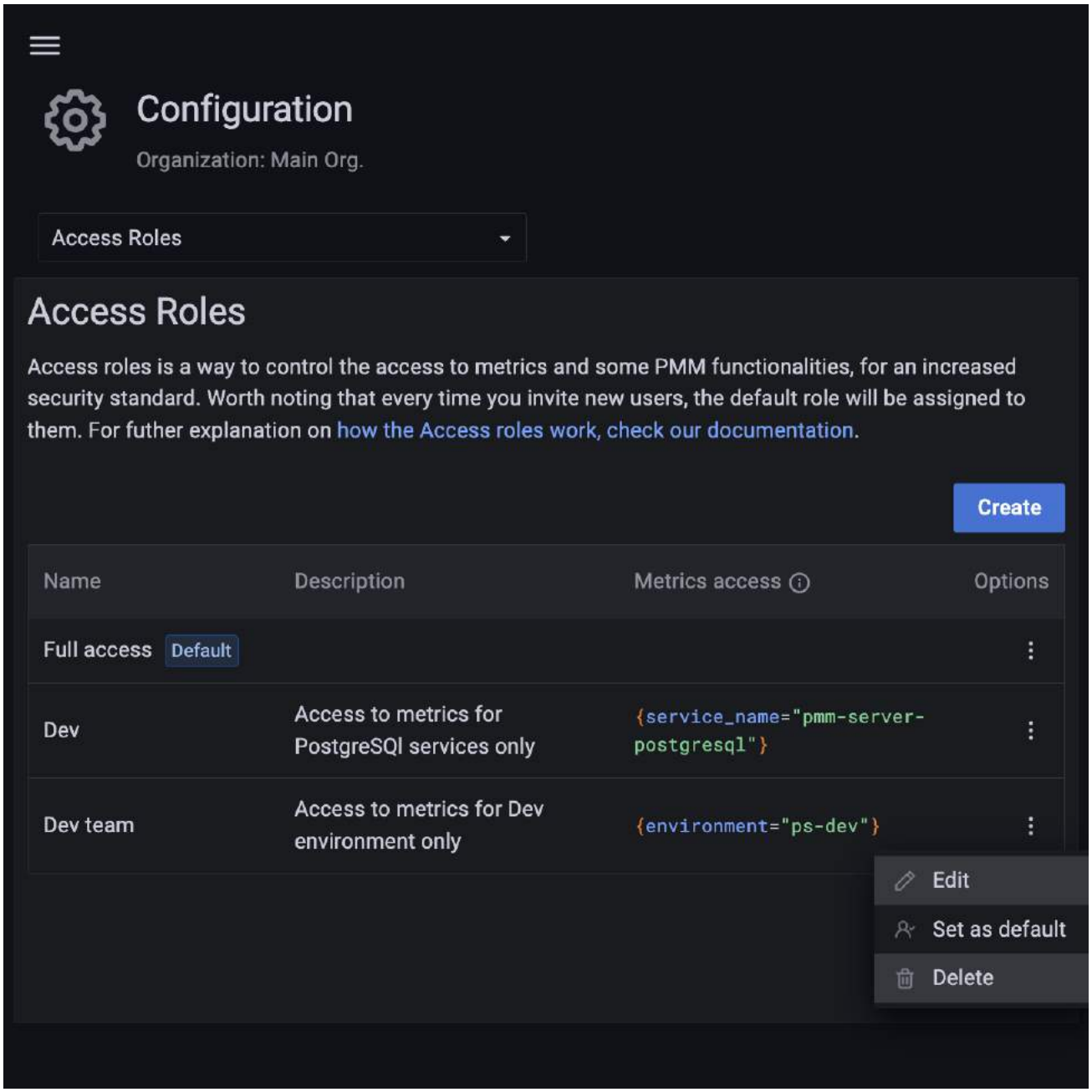


Remove roles

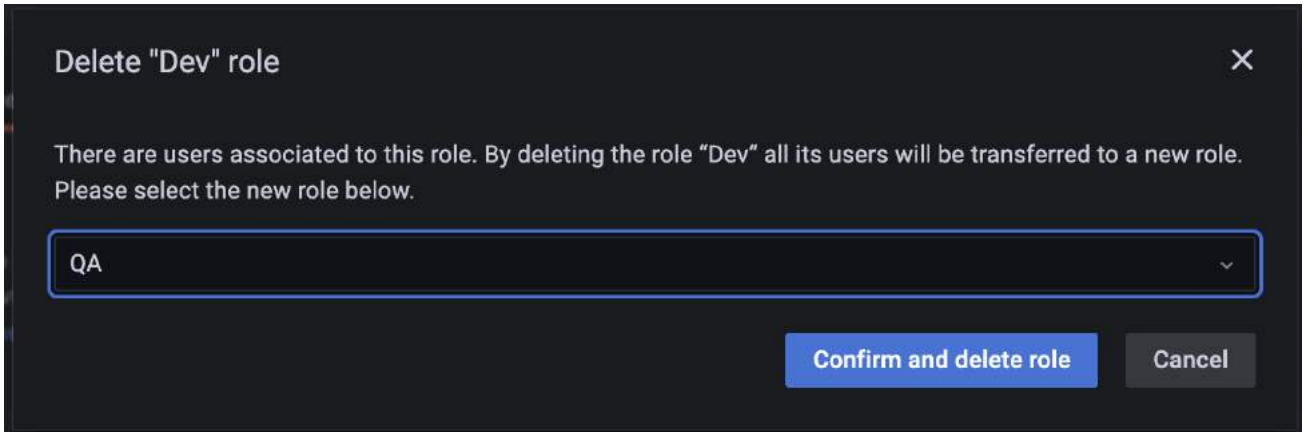
To remove access roles, do the following:

1. From the *Main* menu, navigate to *Configuration → Access Roles*. The *Access Roles* tab opens.
2. On the role you want to remove, click the *ellipsis (three vertical dots) → Delete* in the *Options* column. Delete role pop-up opens.

Percona LLC and/or its affiliates ©2024



3. Starting with **PMM 2.36.0**, if the role that you want to delete is already assigned to a user, you will see a drop-down with replacement roles. Select the replacement role and the selected role will be assigned to the user.



Percona LLC and/or its affiliates ©2024

4. Click *Confirm* and delete the role.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

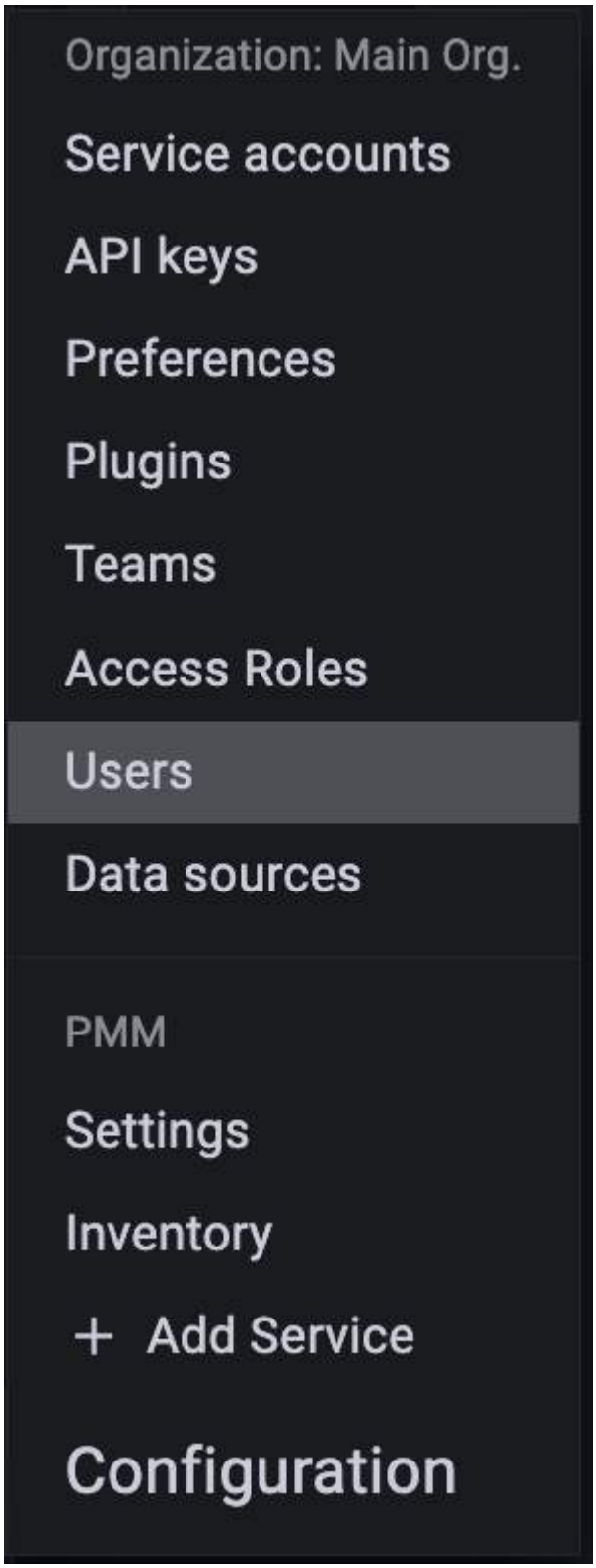
2024-01-11

4.5.7 Assign roles to users

To assign access roles to users, do the following:

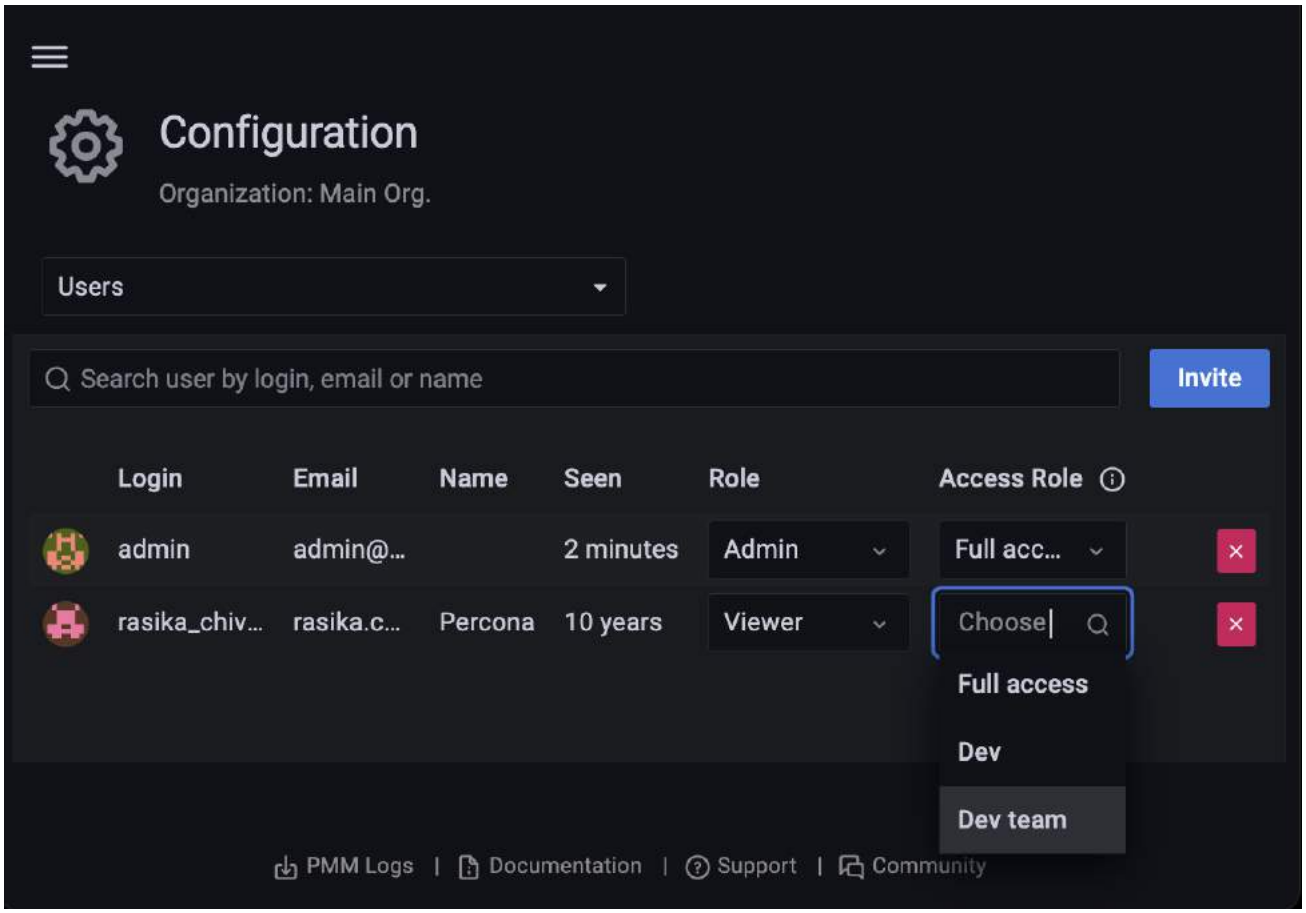
1. From the *Main* menu, navigate to *Configuration* → *Users*. *Users* tab opens.

Percona LLC and/or its affiliates ©2024



2. Select the *Access Role* you want to assign to a user from the dropdown. You can assign several roles to a user.

Percona LLC and/or its affiliates ©2024



Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

4.5.8 Use Case

Use case 1

This use case demonstrates the following scenario:

Labels

- Environments: **prod** and **qa**
- Projects: **shop** and **bank**

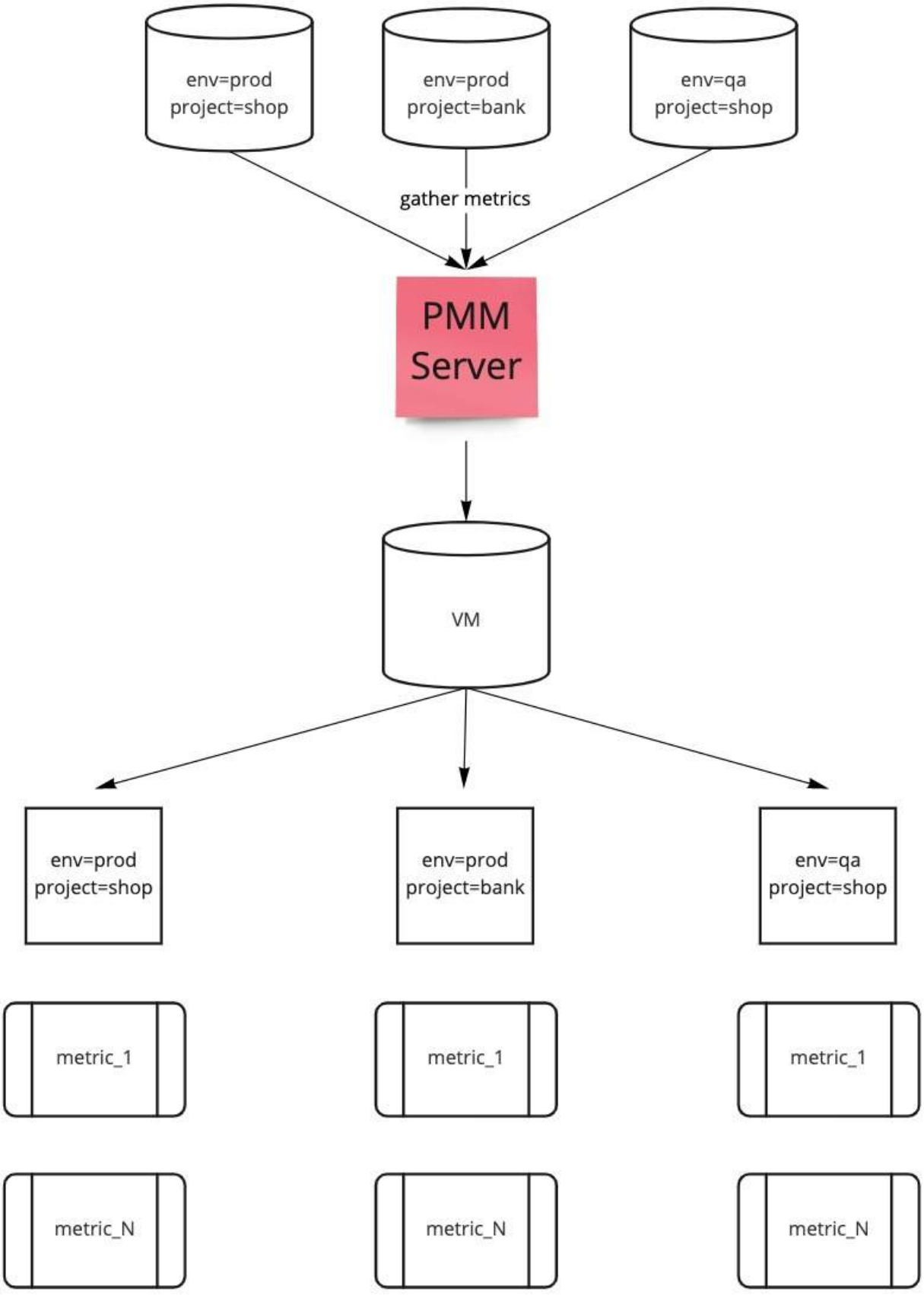
Roles

- Roles: Admin, Dev and QA

Percona LLC and/or its affiliates ©2024

An overview of the infrastructure can be seen in the diagram below. PMM monitors several services. The metrics that are stored in VictoriaMetrics have the appropriate labels.

Percona LLC and/or its affiliates ©2024



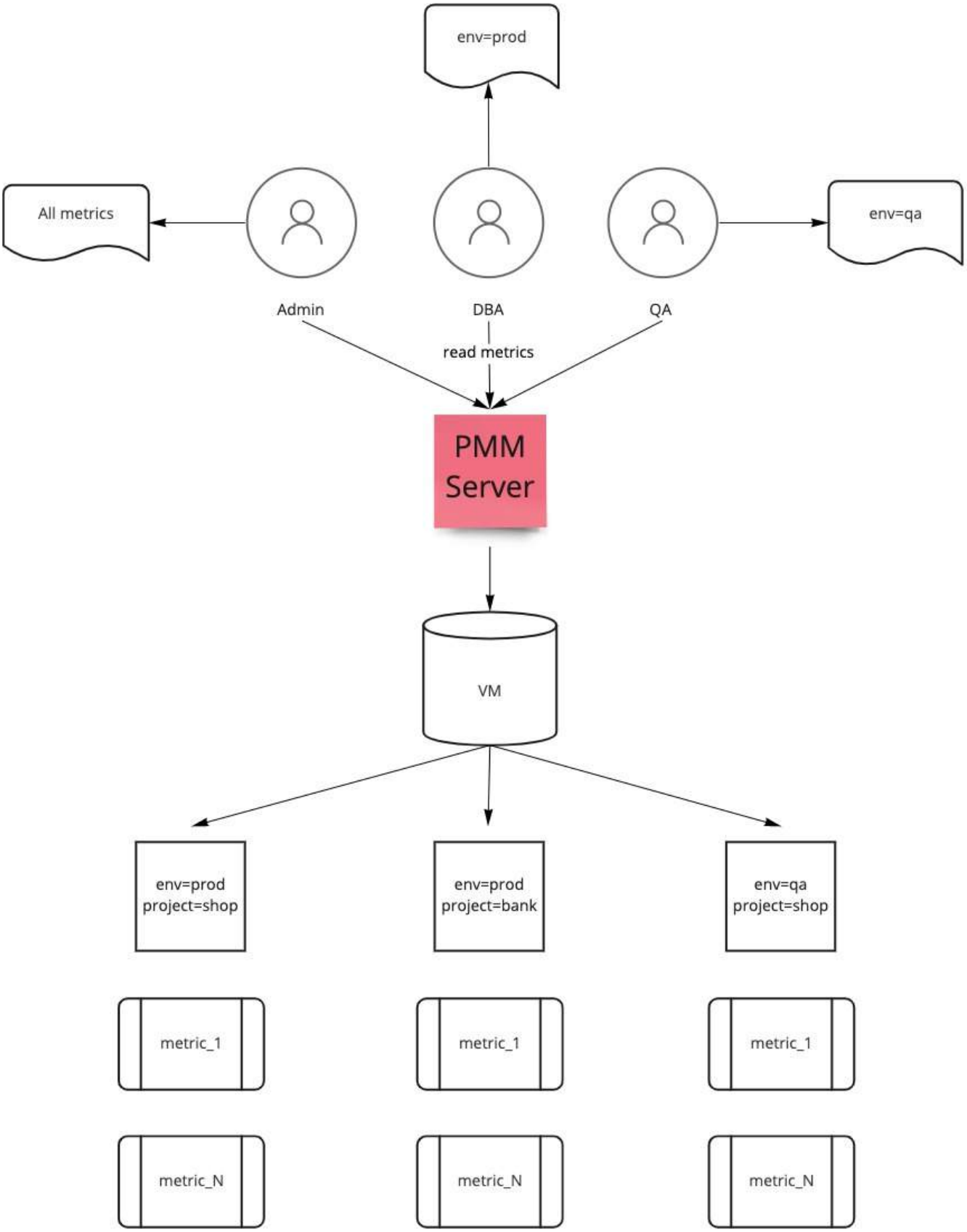
Percona LLC and/or its affiliates ©2024

This diagram shows several roles within a company structure that have access to PMM, as well as the permissions they should be granted:

Percona LLC and/or its affiliates ©2024

- Admin role - has access to all the metrics
- DBA role - has access to all metrics within **env=prod** only
- QA role - has access to all metrics within **env=qa** only

Percona LLC and/or its affiliates ©2024



Percona LLC and/or its affiliates ©2024

Use case 2

The use case demonstrates the following scenario:

Labels

- Environments: prod and dev
- Services: postgresql and mysql

Roles

- role_postresql
- role_mysql

	Role assigned	Labels applied to the role	Accessible Metrics
User 1	role_postresql	dev, service_name=postgresql	The metrics for service postgresql will be accessible.
User 2	role_mysql	prod, service_name=mysql	The metrics for service mysql will be accessible.
User 3	role_postgresql and role_mysql	dev, service_name=postgresql and prod, service_name=mysql	The metrics for both the services mysql and postresql will be accessible.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

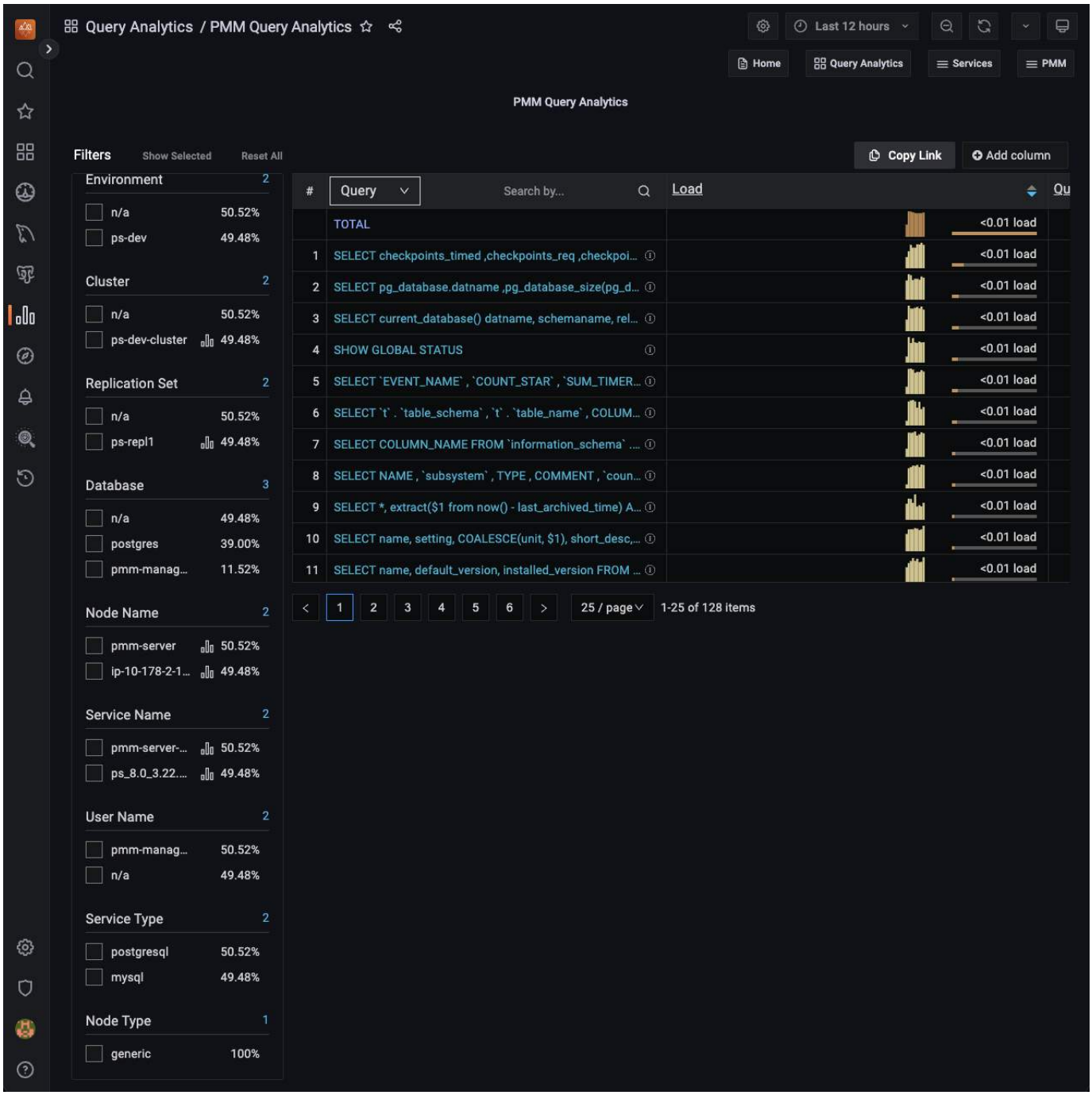
Community Forum Get a Percona Expert

2023-02-23

4.6 Query Analytics

The *Query Analytics* dashboard shows how queries are executed and where they spend their time. It helps you analyze database queries over time, optimize database performance, and find and remedy the source of problems.

Percona LLC and/or its affiliates ©2024



Query Analytics supports MySQL, MongoDB and PostgreSQL. The minimum requirements for MySQL are:

- MySQL 5.1 or later (if using the slow query log).
- MySQL 5.6.9 or later (if using Performance Schema).

Query Analytics displays metrics in both visual and numeric form. Performance-related characteristics appear as plotted graphics with summaries.

The dashboard contains three panels:

- the Filters Panel;
- the Overview Panel;
- the Details Panel.

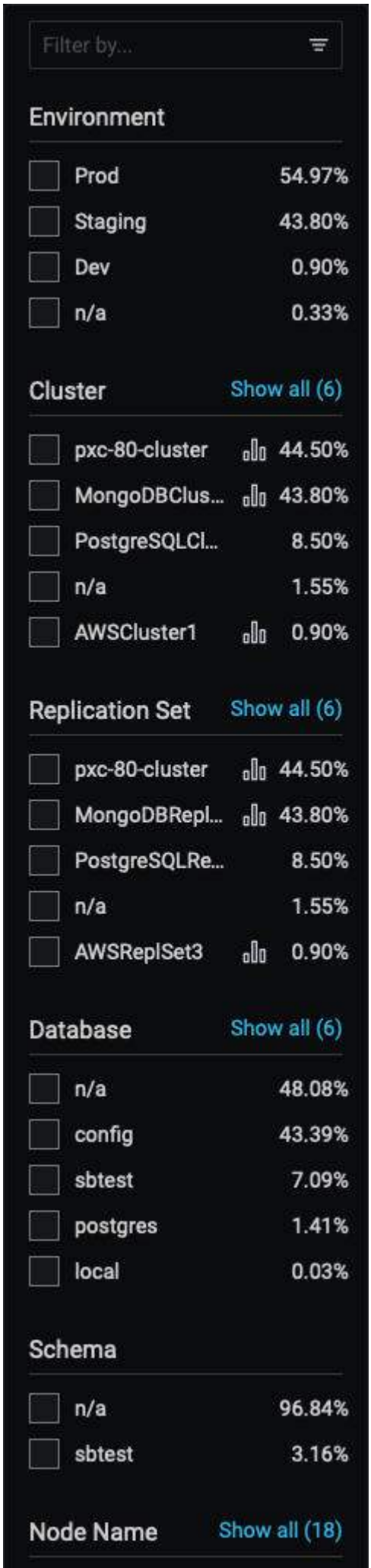
Percona LLC and/or its affiliates ©2024

Query Analytics data retrieval may experience delays due to network conditions. As a result, a small amount of data (up to 1 hour) will be buffered in the memory and reported when the connection is restored.

Percona LLC and/or its affiliates ©2024

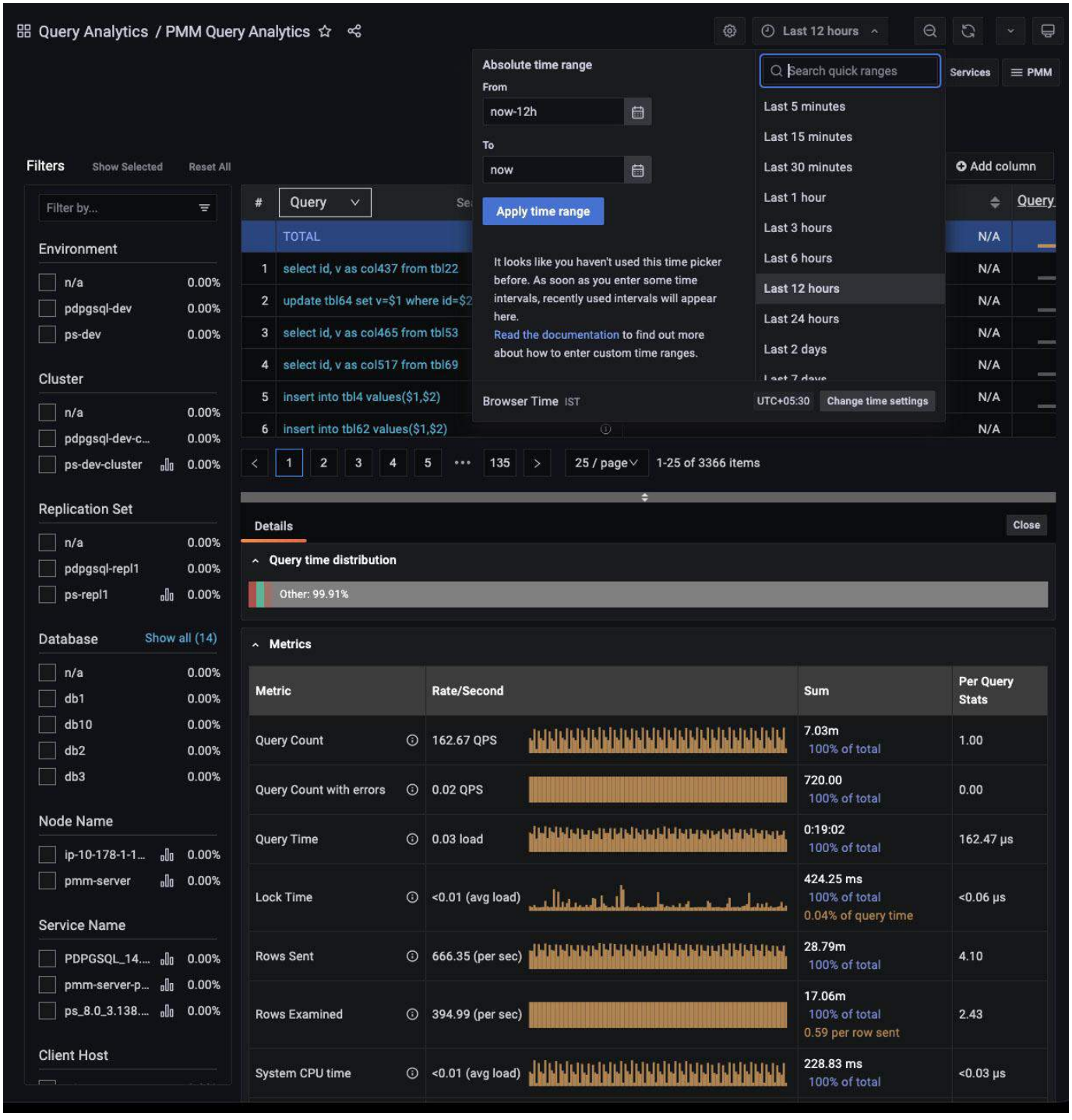
4.6.1 Filters Panel

Percona LLC and/or its affiliates ©2024



Percona LLC and/or its affiliates ©2024

- The *Filter* panel on the left hand side of the dashboard lists the filters grouped by category. It also shows the percentage of the main metrics (explained below). If you select a different metric, the percentages on the left panel will change as per this metric. When you select a metric, it reduces the overview list as per the matching filter.
- The first five of each category are shown. If there are more, the list is expanded by clicking *Show all* beside the category name, and collapsed again with *Show top 5*.
- Applying a filter may make other filters inapplicable. These become grayed out and inactive.
- Click the chart symbol to navigate directly to an item's associated dashboard.
- Separately, the global *Time range* setting filters results by time, either your choice of *Absolute time range*, or one of the predefined *Relative time ranges*.



Percona LLC and/or its affiliates ©2024

Custom filter groups

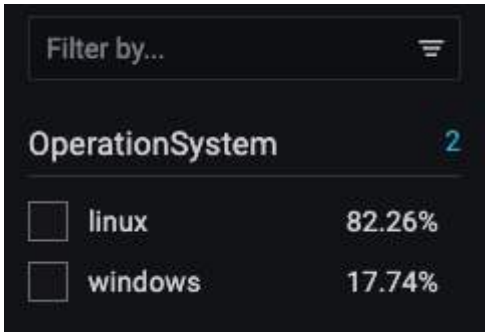
> **Important/Caution**
>
> This feature is still in Technical Preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

Starting with PMM 2.38.0, you can filter queries by custom filter groups based on key=value pairs separated from query comments. By default, this feature is disabled.

SUPPORTED TECHNOLOGIES AND AGENTS

- MySQL (`perfschema`, `slowlog`),
- PostgreSQL (`pg_stat_statements`, `pg_stat_monitor`)

Example



In the image above we have tagged queries running databases on Windows using the following comment:

```
comment: /* OperationSystem='windows' */.
```

Queries from the database running on Linux are tagged with:

```
/* OperationSystem='linux' */.
```

All types of comments and multicomments are supported (`/* */`, `--`, `# etc`).

So the queries are as follows:

```
SELECT * /* OperationSystem='windows' */ FROM city;
SELECT city /* OperationSystem='linux' */ FROM world;
```

In the output, you can see another custom group in the `OperationSystem` filter. Use this to easily filter by any custom key or value.

ENABLING CUSTOM FILTER GROUPS

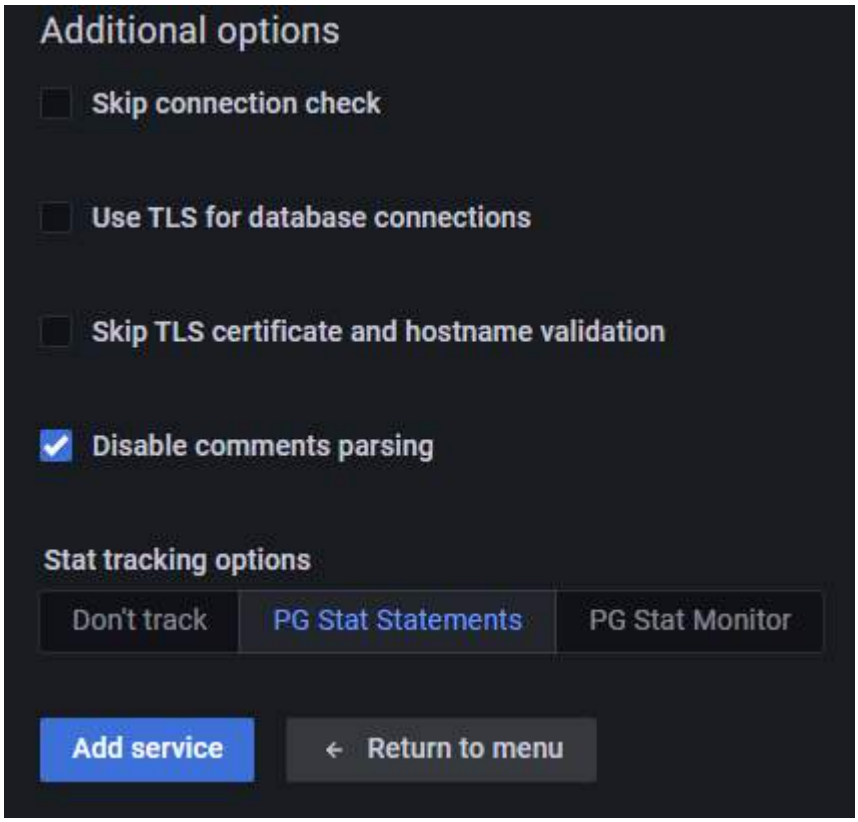
- *CLI*: While adding a service through CLI use the flag `comments-parsing`. Possible values are `on/off`.

  Example for adding MySQL with comments parsing on:

  ```
  pmm-admin add mysql --username=root --password=root-password --comments-parsing="on"
  ```

- *UI*: While adding a service through the UI you will see new checkbox to `enable/disable` comments parsing for current service.

Percona LLC and/or its affiliates ©2024



MySQL CLI

- If you are using official MySQL CLI to trigger queries, start mysql with `--comments` flag. Otherwise comments will not be parsed.
- In case of PGSM (`pg_stat_monitor`), set the DB variable `pgsm_extract_comments=yes`

4.6.2 Overview Panel

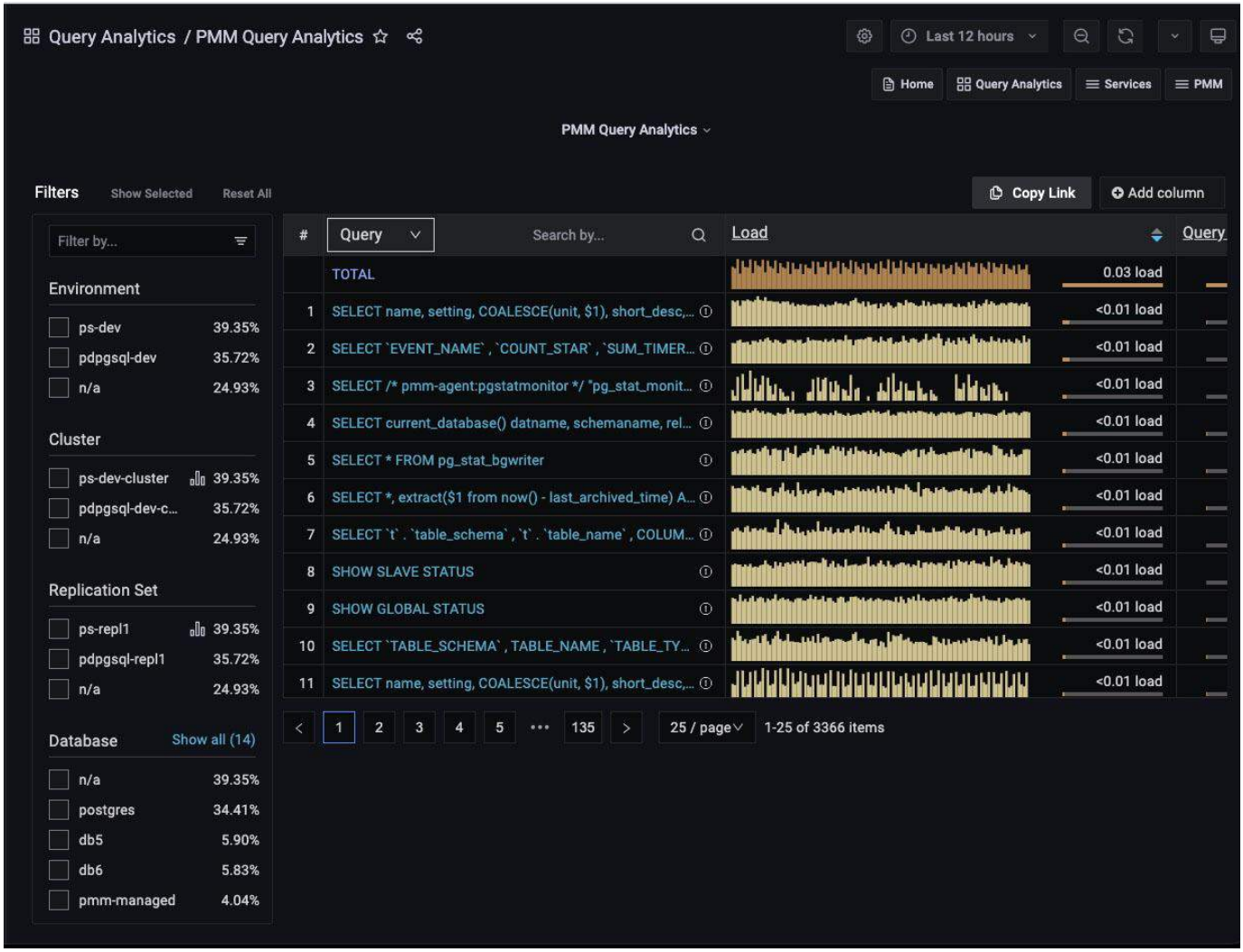
The **Overview** panel is located above the dashboard, on the right side of the **Filters** panel.

Percona LLC and/or its affiliates ©2024

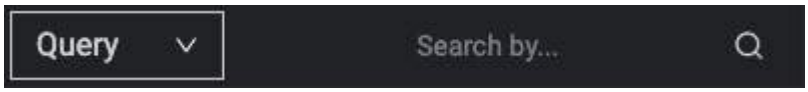
Each row of the table represents the metrics for a chosen object type, one of:

- Query;
- Service Name;
- Database;
- Schema;
- User Name;
- Client Host.

At the top of the second column is the *dimension* menu. Use this to choose the object type.



On the right side of the dimension column is the *Dimension Search* bar.



Enter a string and press Enter to limit the view to queries containing only the specified keywords.

Delete the search text and press Enter to see the full list again.

Percona LLC and/or its affiliates ©2024

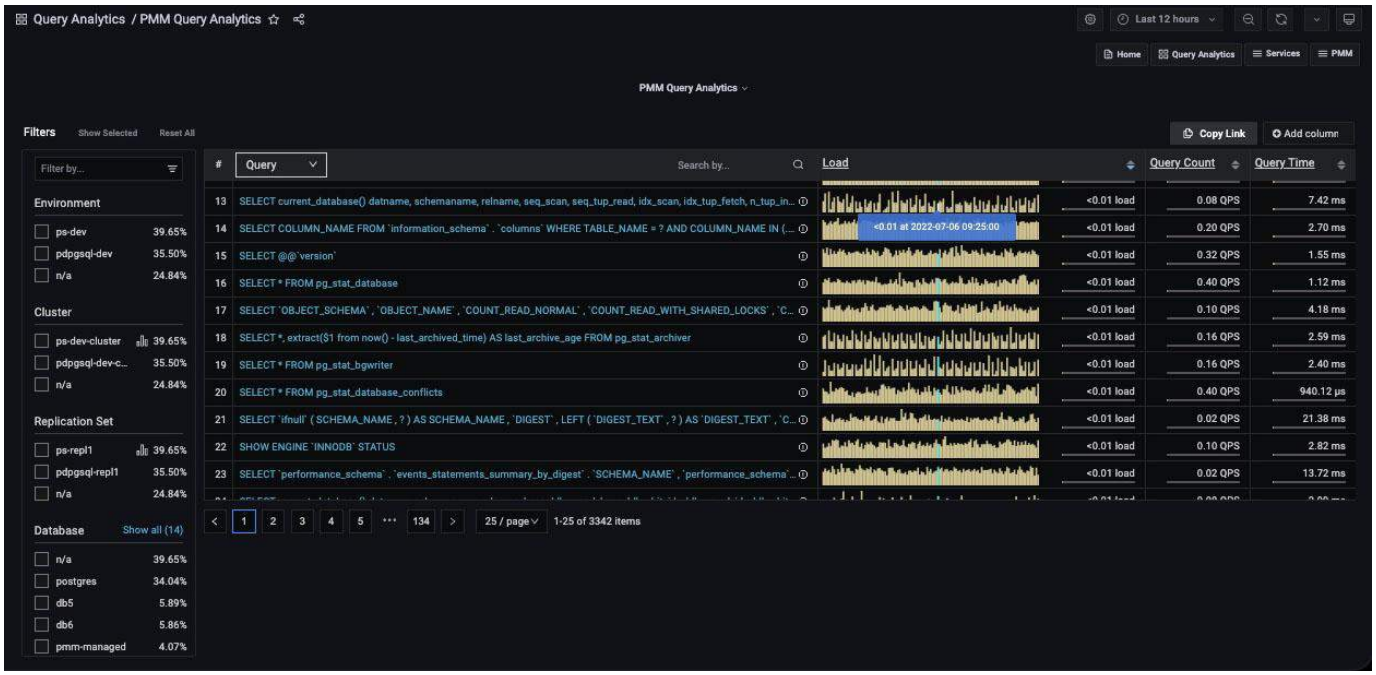
Columns

- The first column is the object's identifier. For *Query*, it is the query's Fingerprint.
- The second column is the *Main metric*, containing a reduced graphical representation of the metric over time, called a *sparkline*, and a horizontal meter, filled to reflect a percentage of the total value.
- Additional values are revealed as mouse-over tool-tips.

Tool-tips

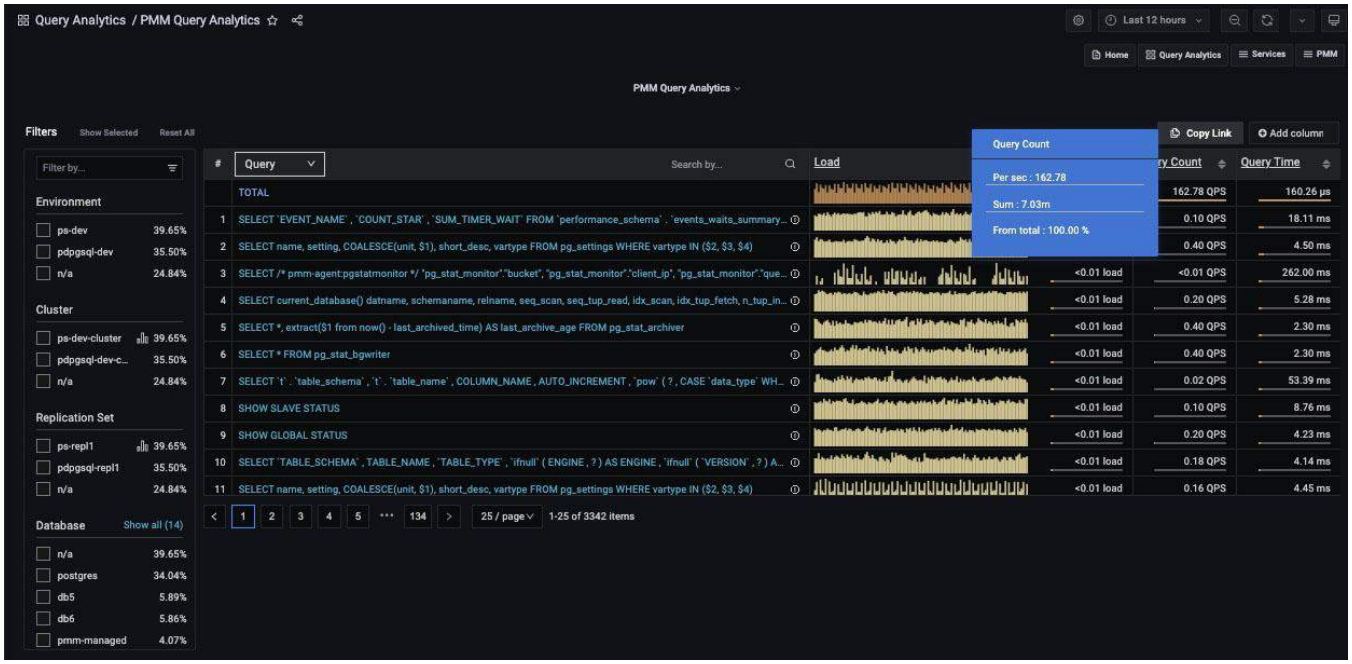
- For the *Query* dimension, hovering over the information icon ℹ reveals the query ID and its example.
- Hovering on a column header reveals an informative tool-tip for that column.
- Hovering on the main metric sparkline highlights the data point and a tooltip shows the data value under the cursor.

Percona LLC and/or its affiliates ©2024

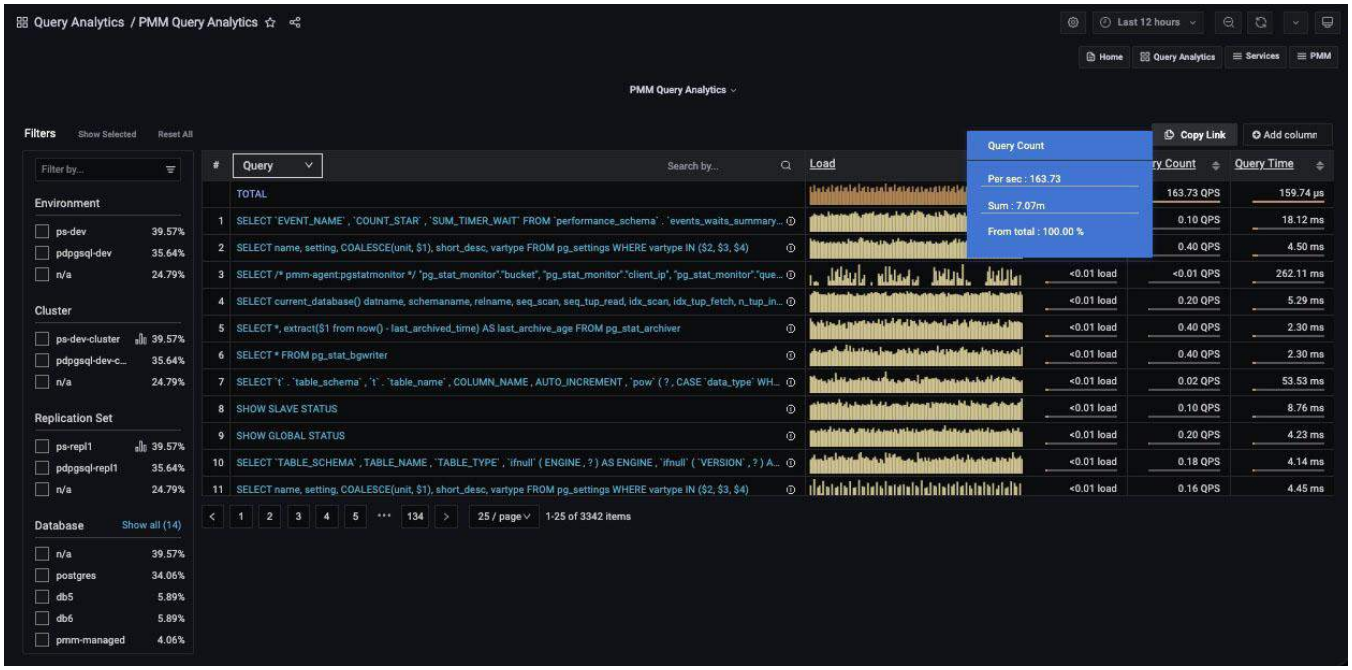


Percona LLC and/or its affiliates ©2024

- Hovering on the main metric meter reveals the percentage of the total, and other details specific to the main metric.

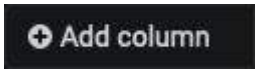


- Hovering on column values reveals more details on the value. The contents depends on the type of value.



Adding and removing columns

- Metrics columns are added with the *Add column* button.



- When clicked, a text field and list of available metrics are revealed. Select a metric or enter a search string to reduce the list. Selecting a metric adds it to the panel.
- A metric column is removed by clicking on the column heading and selecting *Remove column*.

Percona LLC and/or its affiliates ©2024

- The value plotted in the *main metric* column can be changed by clicking a metric column heading and selecting *Swap with main metric*.

Sorting

- The entire list is sorted by one of the columns.
- Click either the up or down caret to sort the list by that column's ascending or descending values.

Pagination

- The pagination device lets you move forwards or backwards through pages, jump to a specific page, and choose how many items are listed per page.

- Queries are grouped into pages of 25, 50 or 100 items.

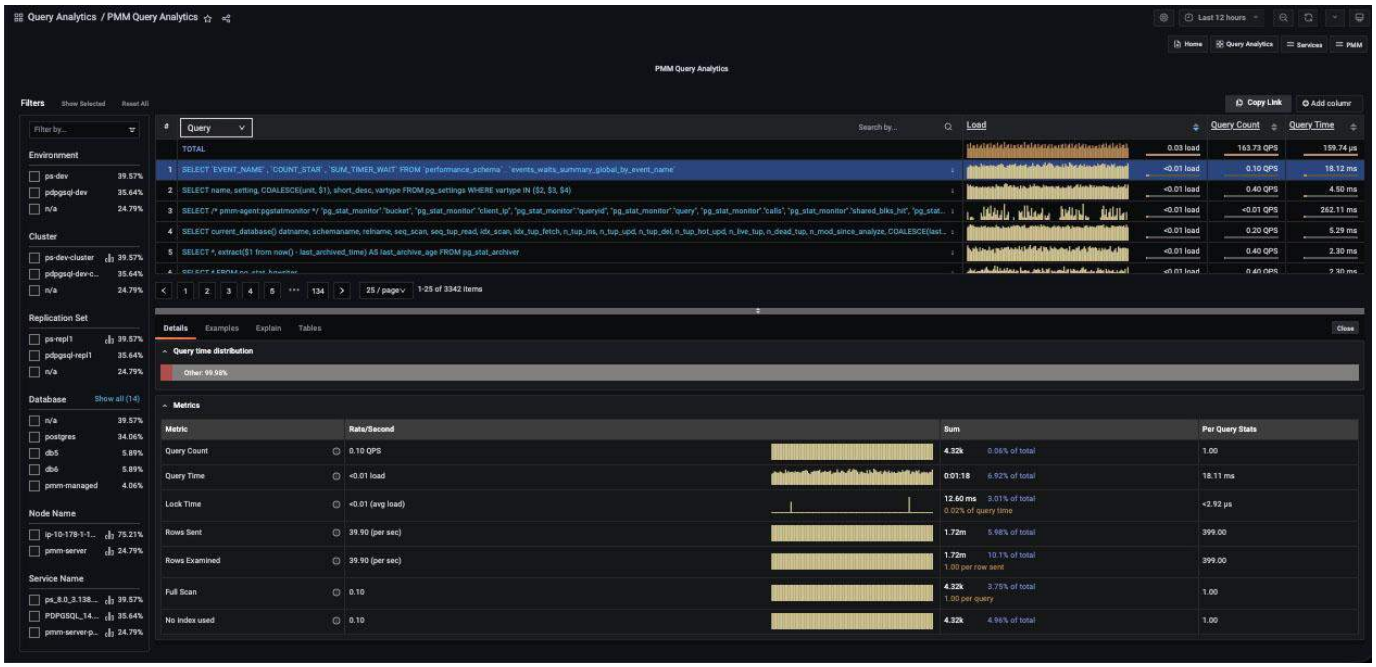
4.6.3 Details Panel

- Selecting an item in the Overview panel opens the *Details panel* with a Details Tab.
- If the dimension is *Query*, the panel also contains the Examples Tab, Explain Tab, and Tables Tab.

Details Tab

The *Details* tab contains a *Query time distribution* bar (only for MySQL databases) and a set of *Metrics* in collapsible subpanels.

Percona LLC and/or its affiliates ©2024



Percona LLC and/or its affiliates ©2024

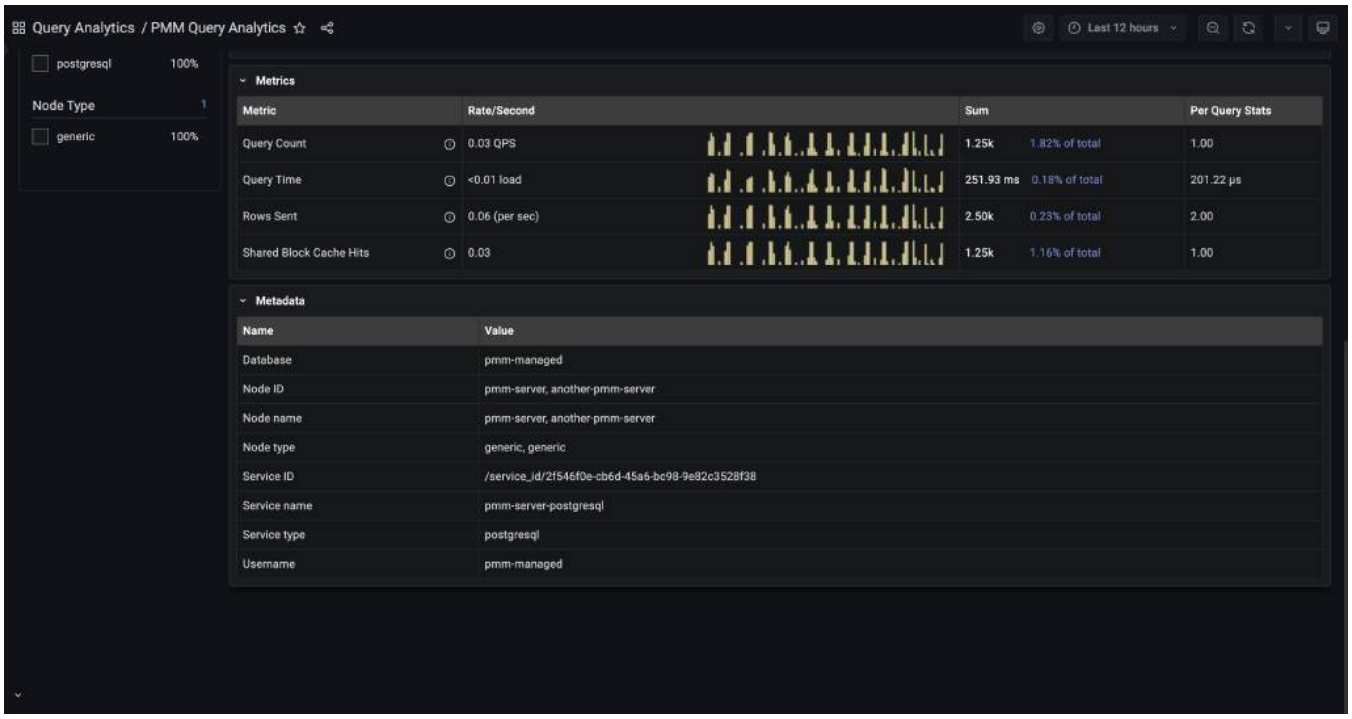
- The *Query time distribution* bar shows a query's total time made up of colored segments, each segment representing the proportion of time spent on a named activity.
  - `query_time`: Statement execution time.
  - `lock_time`: Time to acquire locks.
  - `blk_read_time`: Total time the statement spent reading blocks (if `track_io_timing` is enabled, otherwise zero).
  - `blk_write_time`: Total time the statement spent writing blocks (if `track_io_timing` is enabled, otherwise zero).
  - `innodb_io_r_wait`: Time for InnoDB to read the data from storage.
  - `innodb_queue_wait`: Time the query spent either waiting to enter the InnoDB queue, or in it pending execution.
  - `innodb_rec_lock_wait`: Time the query waited for row locks.
  - `other`: Remaining uncategorized query time.
- *Metrics* is a table with headings:
  - *Metric*: The Metric name, with a question-mark tool-tip that reveals a description of the metric on mouse-over;
  - *Rate/Second*: A sparkline chart of real-time values per unit time;
  - *Sum*: A summation of the metric for the selected query, and the percentage of the total;
  - *Per Query Stats*: The value of the metric per query.
- Each row in the table is a metric. The contents depends on the chosen dimension.
- *Metadata* table:

  PMM 2.39.0 now includes a metadata table for QAN to address the issue of identifying problematic queries. This feature allows you easily identify the services, nodes, and labels associated with your identified queries.

  The **Metadata** table shows the following information:

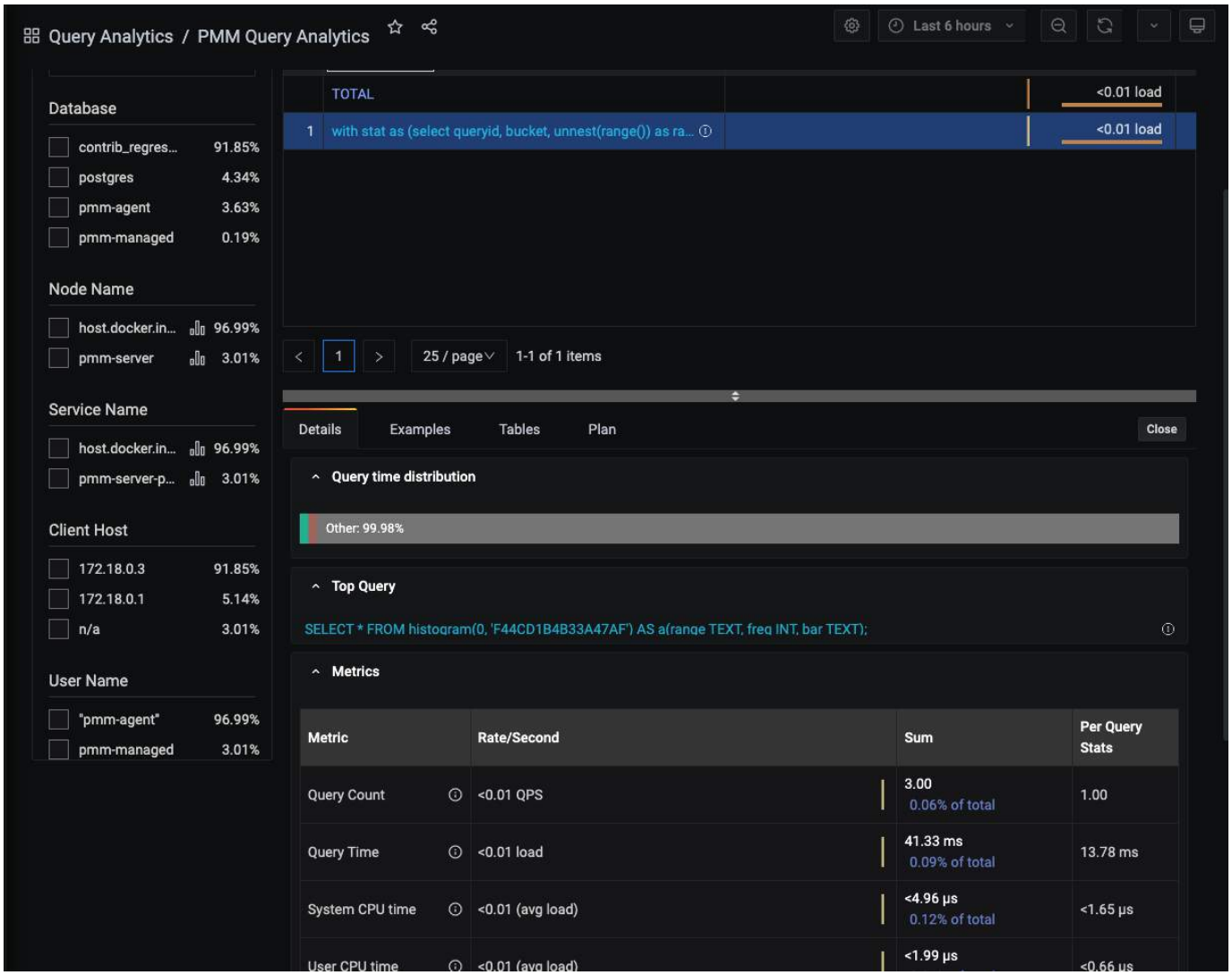
  - *Database*: The database being monitored.
  - *Environment*: Environment being used such as Dev, Staging, Production, etc.
  - *Node Name*: Name or identifier of a node.
  - *Node type*: Type of the node.
  - *User name*: User name of database being monitored.
  - *Service Name*: The name or identifier of the service.
  - *Service type*: The service which is being monitored. For example, MySQL, PostgreSQl, etc.

Percona LLC and/or its affiliates ©2024



For PostgreSQL queries (when using `pg_stat_monitor`) the top query will also be shown in the details section if the query was called by an outer query.

Percona LLC and/or its affiliates ©2024



Other useful metrics (when using *pg_stat_monitor*) to monitor PostgreSQL Server performance are Histograms. *Histograms* provide more explicit information about number of queries for fingerprint (`queryid`). Ranges are from 0 seconds up to 100 seconds.

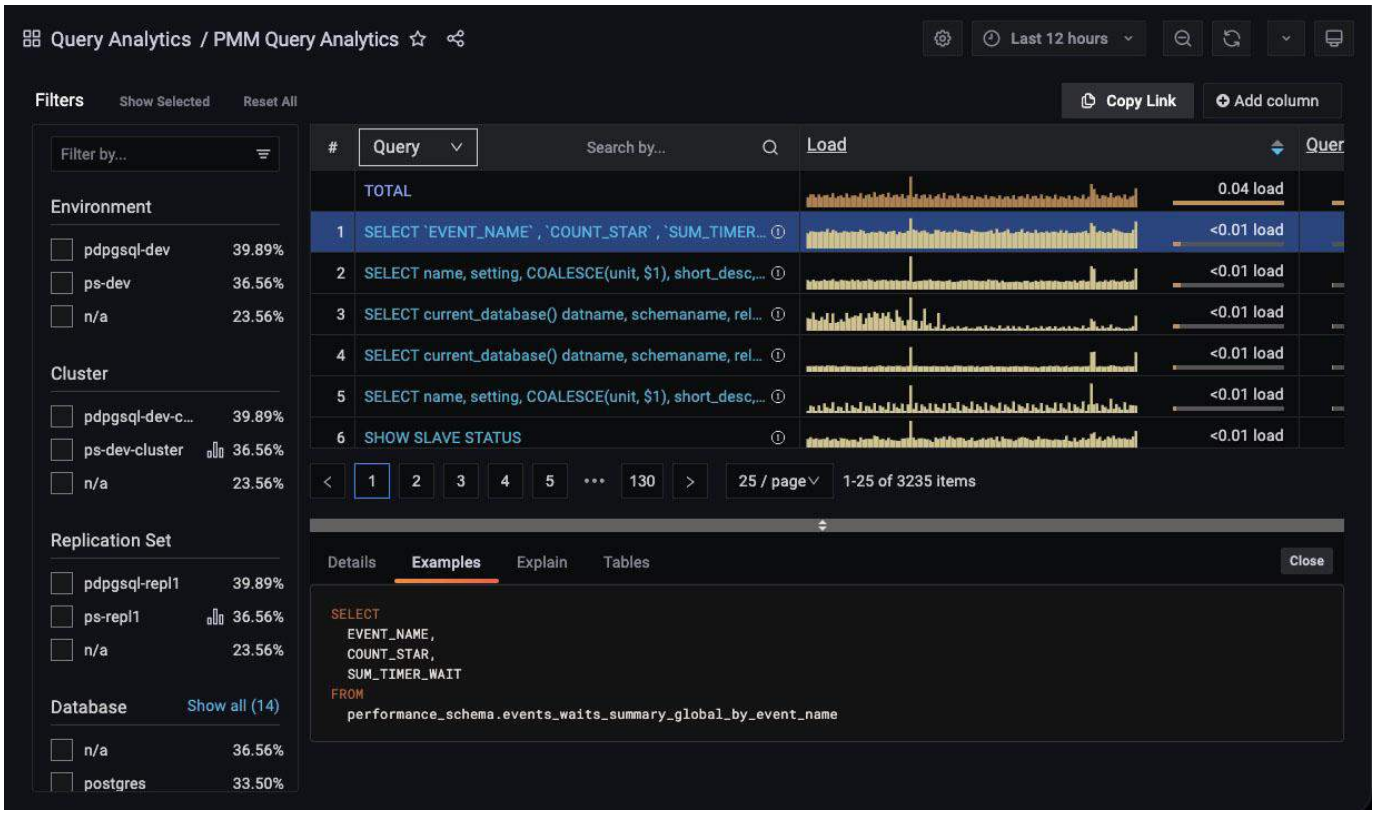
Percona LLC and/or its affiliates ©2024

Here is picture of *histogram* in graph:

Examples Tab

(For *Query* dimension.)

The *Examples* tab shows an example of the selected query's fingerprint or table element.



Query example and fingerprint can be truncated to 1024 long to reduce space usage. In this case, the query explains section will not work.

Percona LLC and/or its affiliates ©2024

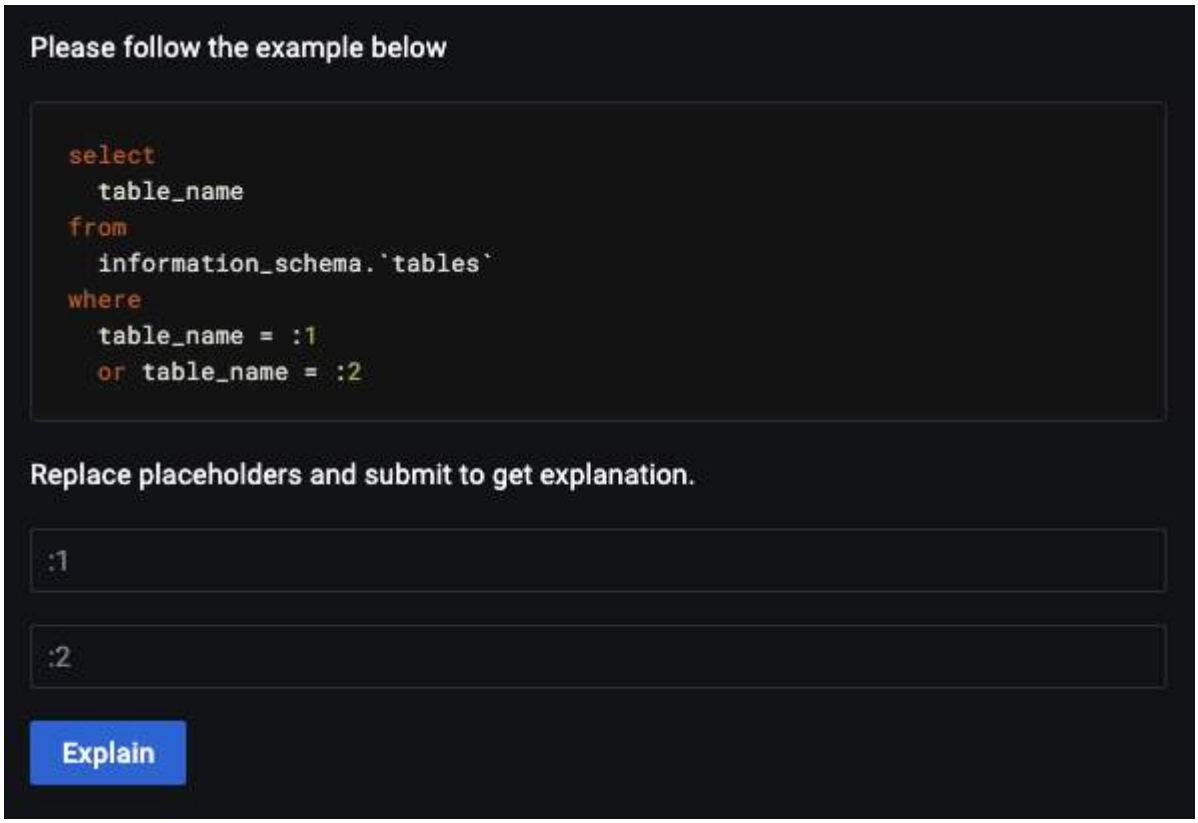
Explain Tab

(For *Query* dimension.)

The *Explain* tab shows the `explain` output for the selected query, in Classic or JSON formats.

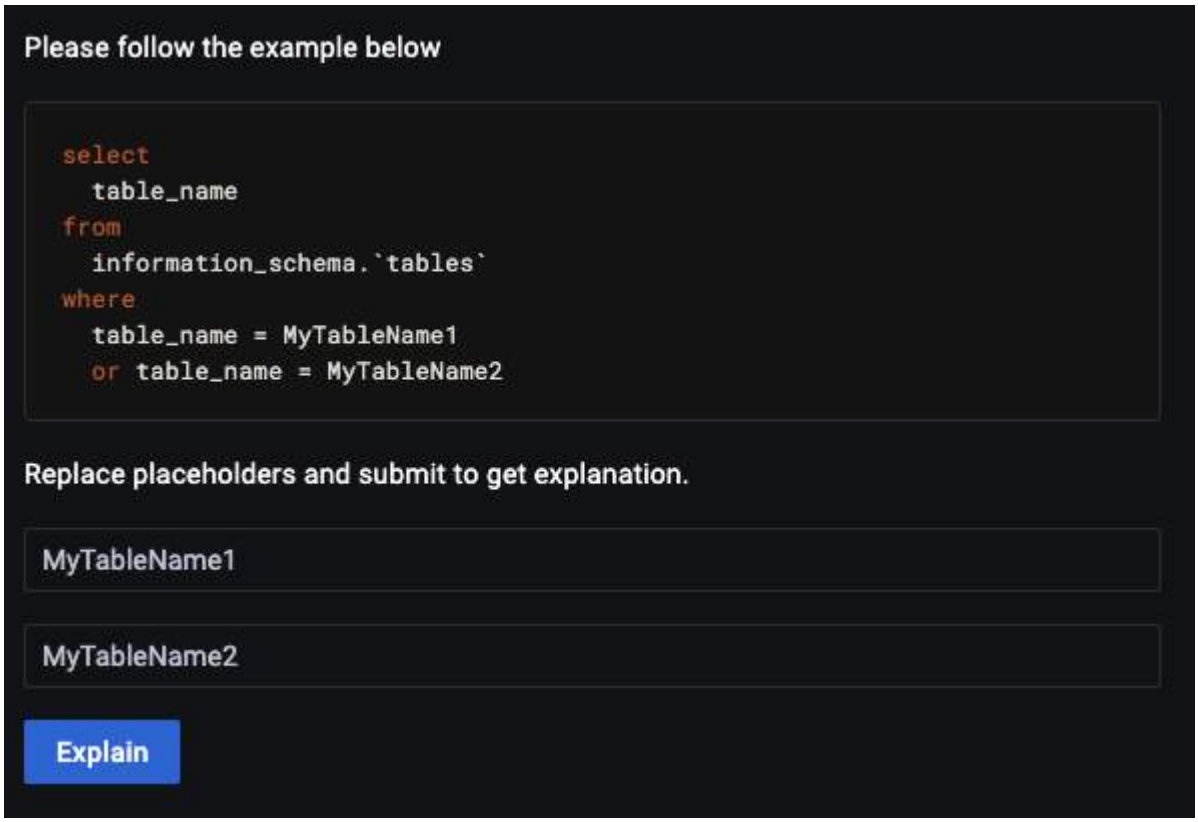
- MySQL: Classic and JSON.
- MongoDB: JSON only.
- PostgreSQL: Supported by pg_stat_monitor (PGSM), not by pg_stat_statements (PGSS).

Starting with PMM 2.33.0, for MySQL, the *Explain* tab is supported without the *Examples* enabled. If a query in the *Explain* tab contains sensitive data, placeholders will replace them. Before you can run Explain, you must specify the values for these placeholders. This image illustrates the query with placeholders.



Below is an illustration of the same query using values instead of placeholders.

Percona LLC and/or its affiliates ©2024



The image shown above illustrates a query with two placeholders. Therefore, you must enter the correct values in both fields. After filling in these values, click *Explain* to get the results like in the previous PMM versions without data leaks. You will get result like in previous PMM versions. This method of `explain` prevents data leak.

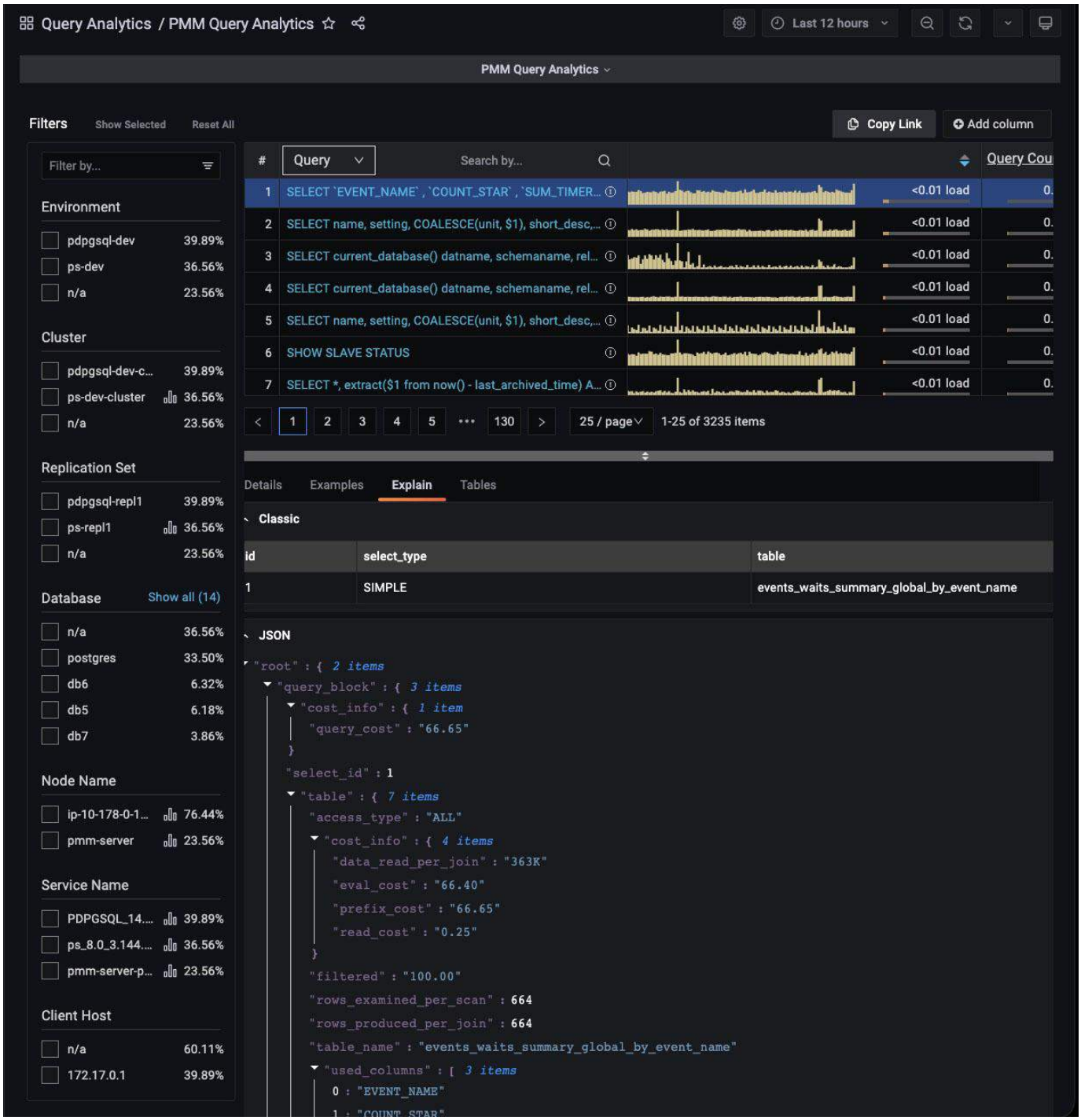
'Explain' for MongoDB

To run Explain you need the same permissions as for executing the original query. For example, to run explain on `updates` you need update permissions.

Example: Grant the `explainRole` with update permissions.

```
db.grantPrivilegesToRole( "explainRole", [ { resource: { db: "", collection: "" }, actions: [ "update" ] } ])
```


Percona LLC and/or its affiliates ©2024

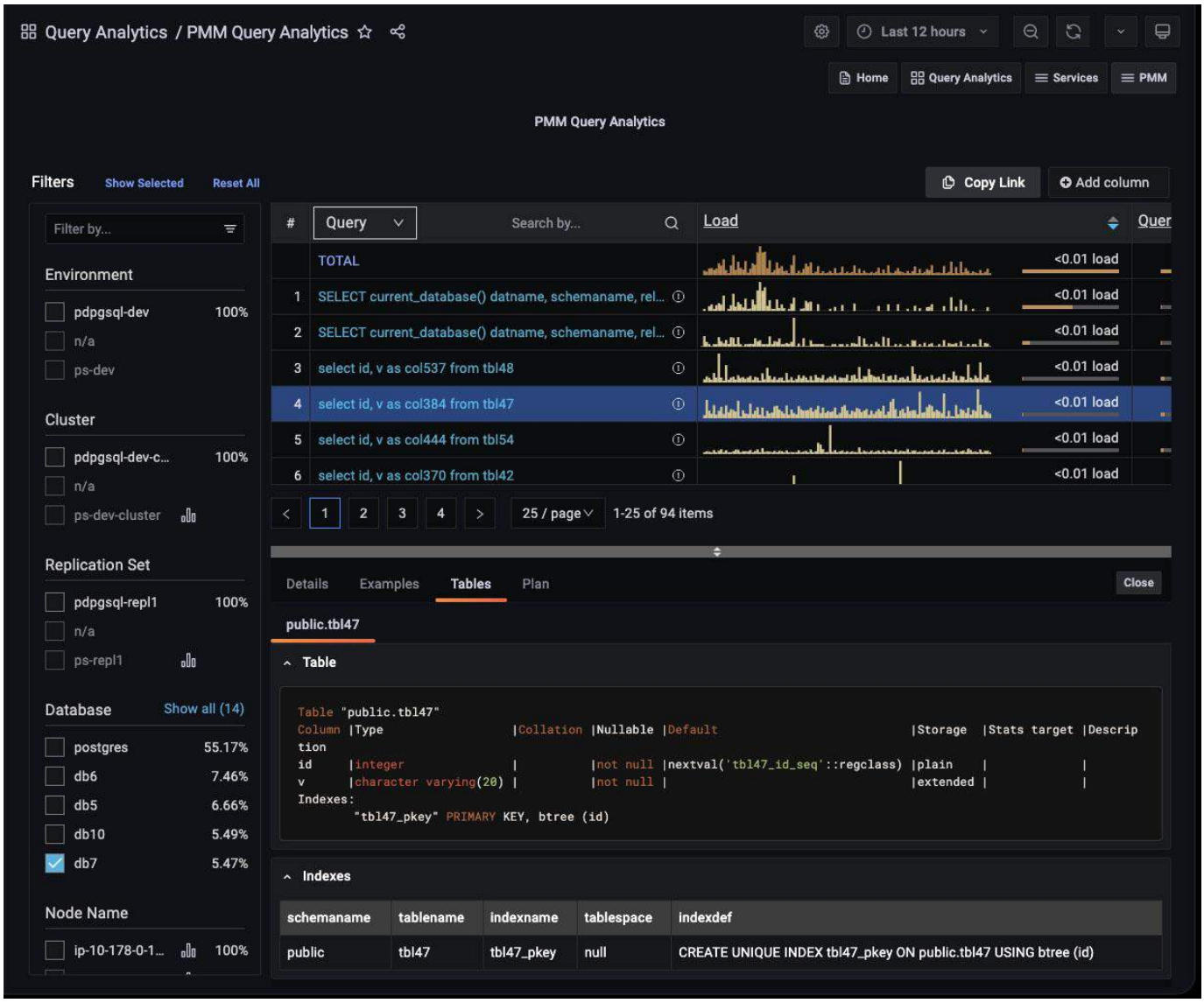


Tables Tab

(For *Query* dimension.)

The *Tables* tab shows information on the tables and indexes involved in the selected query.

Percona LLC and/or its affiliates ©2024

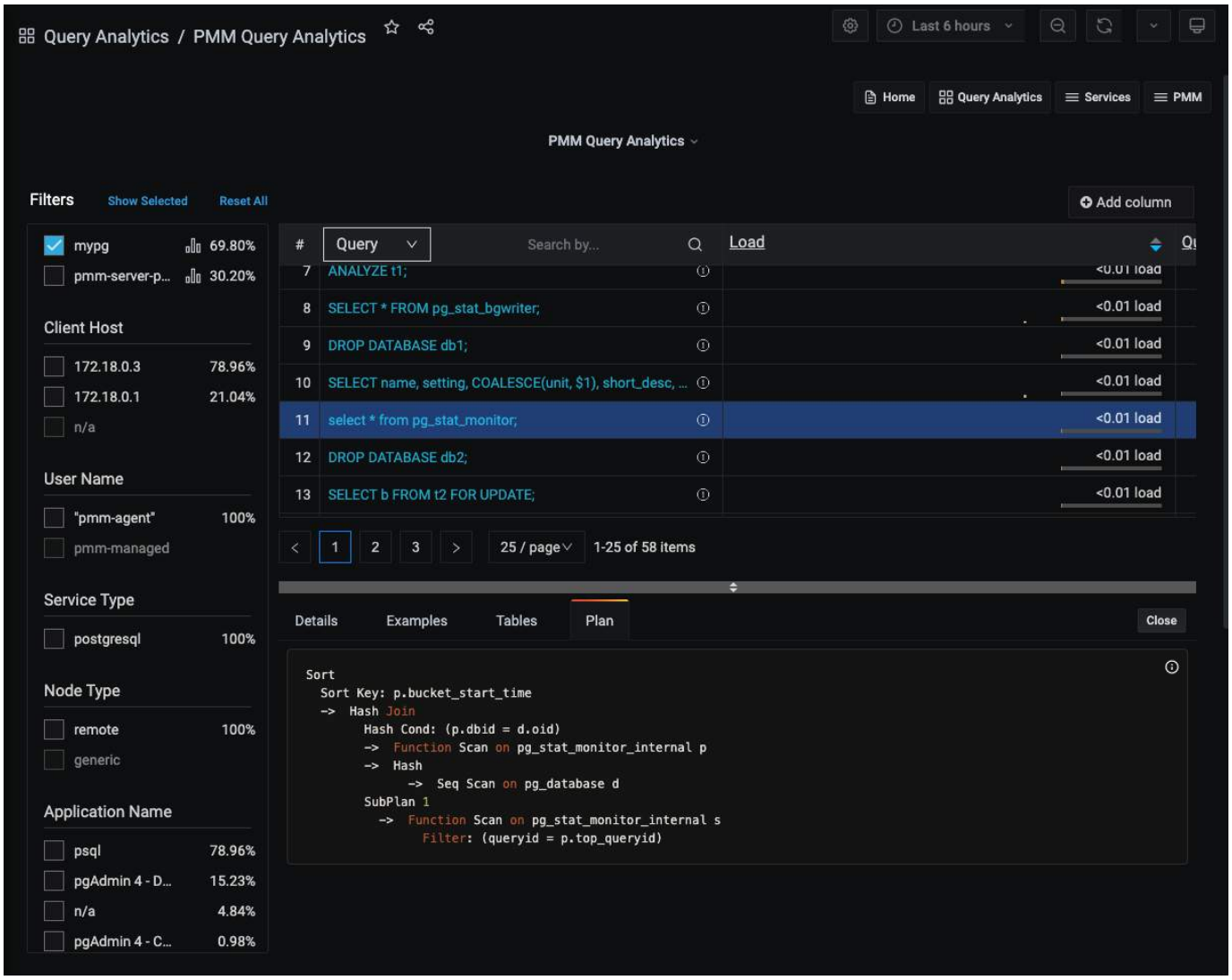


Plan Tab

(For *Query* dimension.)

The *Plan* tab shows the plan for PostgreSQL queries (only available when using *pg_stat_monitor*).

Percona LLC and/or its affiliates ©2024



4.6.4 Query Analytics (QAN) for PostgreSQL

QAN offers support for two types of query sources: `pg_stat_monitor` (PGSM) and `pg_stat_statements` (PGSS). While PGSS used to be the default query source, recent PMM versions have transitioned to PGSM. If PGSM is unavailable on your system, QAN will seamlessly fall back to PGSS and you will be notified of this in pmm-admin.

Percona LLC and/or its affiliates ©2024

PMM has integrated support for pg_stat_monitor 2.0 (PGSM 2.0) into QAN starting with version 2.36. This robust tool enhances PostgreSQL query performance monitoring by providing access to the latest improvements and fixes included in PGSM2, including:

- Improved internal architecture that results in fewer lock acquisitions and increases performance by approximately 20%.
- Support for PostgreSQL 15
- Enhanced consistency with `pg_stat_statements` so that the `pg_stat_monitor` view has identical column names, columns, and data types as `pg_stat_statements` for every major version of PostgreSQL from versions 11 to 15.
- A bucket status indication (done vs. current) eliminates the need for the tool to evaluate bucket status and facilitates accurate data display.
- The generation of a unique ID for a query makes it easier to monitor query planning, execution, and performance regardless of version, database, user, or schema.
- It has backward compatibility with your historical data (data collected by older PMM and PGSM)
- More detailed histogram ranges
- Security improvements
- All previous PGSM versions are still supported

Postgres 15

You will probably need to grant more permissions to the user in PostgreSQL 15. An error message will appear in the `pmm-agent` log if more permissions are required. This behavior is pertaining to PostgreSQL and not PMM.

You can use this query:

```
CREATE USER pmm WITH SUPERUSER ENCRYPTED PASSWORD 'USERNAME';
```

where `USERNAME` should be replaced by the user.

4.6.5 Query Analytics for MongoDB

MongoDB is conceptually different from relational database management systems, such as MySQL and MariaDB.

Relational database management systems store data in tables that represent single entities. Complex objects are represented by linking tables.

In contrast, MongoDB uses the concept of a document where all essential information for a complex object is stored in one place.

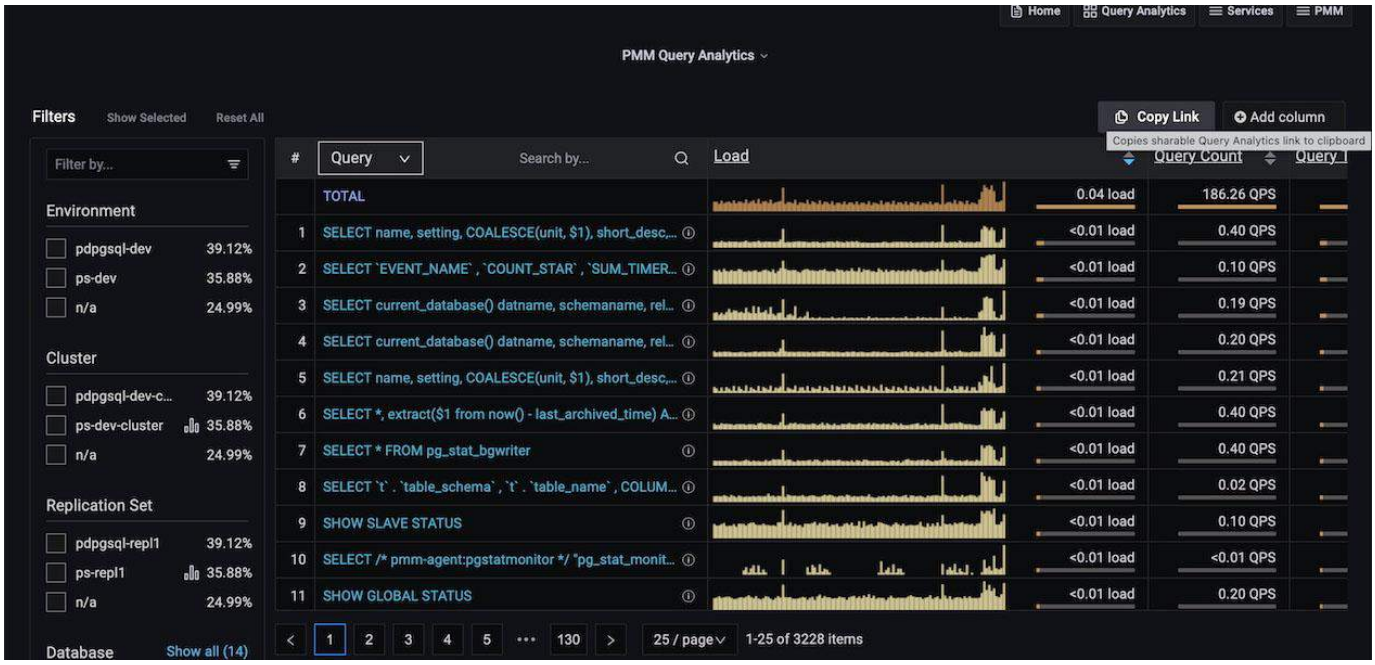
Query Analytics can monitor MongoDB queries. Although MongoDB is not a relational database management system, you analyze its databases and collections in the same interface using the same tools.

4.6.6 Sharing a link for Query Analytics

To share a link for Query Analytics, use *Copy Link*. It copies the link to the clipboard with all the relevant information such as selected query, table page, selected filters, details tab, and time range. Thus, when you open the link, it will display the exact information.

Ensure that you use *Copy Link* to copy the link instead of using the browser address bar or the standard Grafana functionality (to share a dashboard). Otherwise, Query Analytics might not display the exact information that existed while sharing the link. By default, Grafana uses a relative time range and not an absolute range, so it will have a different timestamp when this link is opened.

Percona LLC and/or its affiliates ©2024



4.6.7 QAN under the hood

To understand how metrics are calculated and to learn more details about QAN, see

QAN under the hood.

4.6.8 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-06-03

4.7 Advisors

4.7.1 Working with Advisors

Percona Advisors provide automated insights and recommendations within Percona Monitoring and Management. These proactive insights help you uncover problems before they become larger issues: security risks, misconfigurations, poor performance, etc.

Advisors are grouped by category: Security, Configuration, Performance and Query. Each Advisor category offers a set of automated checks, which investigate a specific range of possible issues. The list of Advisor checks available for your instance depends on whether your instance is connected to Percona Platform, and on your current subscription plan.

Percona LLC and/or its affiliates ©2024

Prerequisites for accessing Advisor checks

All checks are hosted on Percona Platform. PMM Server automatically downloads them from here when the **Advisors** and **Telemetry** options are enabled in PMM under **Configuration > Settings > Advanced Settings**. Both these options are enabled by default.

Highest security for your databases

Percona Platform communicates with PMM via secure channels, using the highest standards for privacy and security. Before downloading and running Advisor checks on your database, PMM verifies the content and integrity of all Advisor checks to confirm that every component originated from Percona Platform and that no one has altered them since the checks were digitally signed.

ADVISOR CHECK TIERS AND PLATFORM ENTITLEMENTS

Depending on the entitlements available for your Percona Account, the set of Advisor checks that PMM can download from the Percona Platform differs in terms of complexity and functionality.

If your PMM instance is not connected to Percona Platform, PMM can only use the default Advisor checks. As soon as you connect your PMM instance to Percona Platform, has access to additional checks, available only for registered PMM instances.

If you are a Percona customer with a Percona Customer Portal account, you also get access to Standard/Premium Advisor checks, which offer more advanced database health information.

To see the complete list of available checks, see the Advisor Checks for PMM topic in the Percona Platform documentation.

Enable/Disable

To download the checks available for your Percona Account, the Advisors and Telemetry options have to be enabled under **Configuration > Settings > Advanced Settings**.

These options are enabled by default so that PMM can run automatic Advisor checks in the background. However, you can disable them at any time if you do not need to check the health and performance of your connected databases.

Automatic checks

Advisor checks can be executed manually or automatically. By default, PMM runs all the checks available for your PMM instances every 24 hours.

CHANGE RUN INTERVAL FOR AUTOMATIC ADVISORS

You can change the standard 24-hour interval to a custom frequency for each Advisor check:

- *Rare interval* - 78 hours
- *Standard interval* (default) - 24 hours
- *Frequent interval* - 4 hours

To change the frequency of an automatic check:

1. Click **Advisors**.
2. Select the Advisor tab that contains the check for which you want to change the frequency.
3. Expand the relevant Advisor and scroll through the list to find your check. Alternatively, use the **Filter** section at the top of the table to search checks by Name, Description, Status, or Interval.

Percona LLC and/or its affiliates ©2024

> **Tip**
>
> If you need to share filtered Advisor results with your team members, send them the PMM URL. This saves your search criteria and results.

a. Click the **Interval** icon in the **Actions** column, next to the check you want to update.

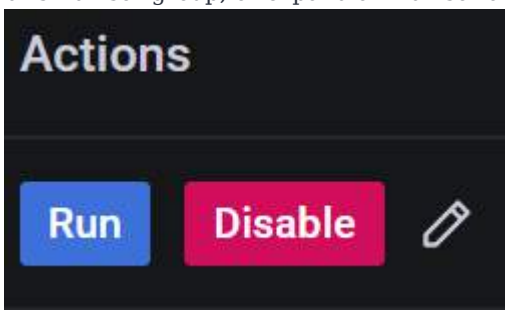
b. Chose an interval and click **Save**.

Manual checks

In addition to the automatic checks that run every 24 hours, you can also run checks manually, for ad-hoc assessments of your database health and performance.

To run checks manually:

1. Click **Advisors** on the main menu.
2. Select the Advisor tab that contains the checks which you want to run manually.
3. Click **Run checks** to run all the available checks for this Advisor group, or expand an Advisor and click **Run** next to each check that you want to run individually.



Advisor checks results

The results are sent to PMM Server where you can review any failed checks on the Home dashboard. The summary count of failed checks is classified as:

- **Critical**, which also includes checks tagged as **Alert** and **Emergency**
- **Error**
- **Warning**
- **Notice**, which also includes checks tagged as **Info** and **Debug**

To see more details about the available checks and any checks that failed, click the *Advisors* icon on the main menu.

Check results data *always* remains on the PMM Server. This is not related to anonymous data sent for Telemetry purposes.

Percona LLC and/or its affiliates ©2024

Create your own Advisors

PMM Advisors offer a set of checks that can detect common security threats, performance degradation, data loss and data corruption.

Developers can create custom checks to cover additional use cases, relevant to specific database infrastructure. For more information, see Develop Advisor checks.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

4.7.2 Advisors details

List of database Advisors

Percona Monitoring and Management (PMM) offers four categories of database Advisors to help you improve database performance: Configuration, Performance, Query and Security Advisors.

Each Advisor includes a set of automated checks, which investigate a specific range of possible issues and areas of improvement: security threats, non-compliance issues, performance degradation, query and index optimization strategies etc.

This page presents the complete list of database Advisors along with the corresponding subscription tier for which they are available.

You can also access this list through the **Advisor checks for PMM** section in the Percona Portal documentation, as the Advisors are hosted on the Percona Platform. PMM Server automatically downloads them from this source when the Advisors and Telemetry options are enabled in PMM under **Configuration > Settings > Advanced Settings**. Both options are enabled by default.

CONFIGURATION ADVISORS

Advisor Name	Description	Subscription	Database Technology
Version Configuration	Notifies of newly released database versions to streamline database maintenance and ensure the most up-to-date performance.	All Users	MySQL, MongoDB, PostgreSQL
Generic Configuration	Provides basic recommendations for improving your database configuration.	All Users	MySQL, MongoDB, PostgreSQL
Resources Configuration	Watches your database and gives you recommendations for efficient management of resources like binaries architecture, CPU number versus DB Configuration, etc.	All Users	MySQL, MongoDB
Connection Configuration	Provides recommendations on configuring database connection parameters for improving database performance.	Customers only	MySQL, MongoDB, PostgreSQL
			MySQL, MongoDB

Percona LLC and/or its affiliates ©2024

Advisor Name	Description	Subscription	Database Technology
Replication Configuration	Provides recommendations for scalable replication in database clusters.	Customers only	
InnoDB Configuration	Advises on configuring InnoDB optimization for high performance.	Customers only	MySQL
Vacuum Configuration	Provides recommendations on optimizing Vacuum operations.	Customers only	PostgreSQL

PERFORMANCE ADVISORS

Advisor Name	Description	Subscription	Database Technology
Generic Performance	Provides basic database configuration recommendations for high-performance query execution.	All Users	MongoDB, PostgreSQL
Vacuum Performance	Helps improve the efficiency and execution speed of database Vacuum commands.	Customers only	PostgreSQL
Replication Performance	Checks efficient replication usage of your database.	Customers only	MongoDB, PostgreSQL

SECURITY ADVISORS

Advisor Name	Description	Subscription	Database Technology
CVE Security	Informs you of any database versions affected by CVE.	All Users	MongoDB, PostgreSQL
Configuration Security	Checks your database configuration to ensure that security best practices are correctly implemented.	All Users	MySQL, MongoDB, PostgreSQL
Authentication Security	Ensures that all database authentication parameters are configured securely.	Customers only	MySQL, MongoDB, PostgreSQL
Replication Security	Helps safeguard data replication by assessing security risks and providing recommendations for improving protection.	Customers only	MySQL
Connection Security	Helps identify security issues on network connections and provides recommendations for enhancing security.	Customers only	MySQL, MongoDB

QUERY ADVISORS

Advisor Name	Description	Subscription	Database Technology
Index Query	Provides query and index optimization strategies for peak database performance.	Customers only	MySQL, MongoDB, PostgreSQL
Schema Design Query	Helps create efficient database schemas by analyzing queries and offering suggestions for optimization.	All Users	MySQL

List of checks

Every Advisor consists of one or more Advisor checks. We have listed the checks and their details here.

Percona LLC and/or its affiliates ©2024

MONGODB

Advisor	Check Name	Description	Summary
Connection Configuration	mongodb_connection_sudden_spike	Warns about any significant increase in the number of connections exceeding 50% of the recent or typical connection count.	MongoDB Sudden Increase in Connection Count
Connection Configuration	mongodb_connections	Returns the current number of connections as an informational notice when connection counts exceed 5000.	MongoDB High Connections
Generic Configuration	mongo_cache_size	Warns when Mongo wiredtiger cache size is greater than the default 50%.	Mongo Storage Cache
Generic Configuration	mongodb_active_vs_available_connections	Warns if the ratio between active and available connections is higher than 75%.	MongoDB Active vs Available Connections
Generic Configuration	mongodb_journal	Warns if the journal is disabled.	MongoDB Journal
Generic Configuration	mongodb_loglevel	Warns if MongoDB is not using the default Log level.	MongoDB Non-Default Log Level
Generic Configuration	mongodb_read_tickets	Warns if MongoDB is using more than 128 read tickets.	MongoDB Read Tickets
Generic Configuration	mongodb_write_tickets	Warns if MongoDB is using more than 128 write tickets.	MongoDB Write Tickets
Generic Configuration	mongodb_write_tickets_runtime	Warns if MongoDB is using more than 128 write tickets during runtime.	MongoDB - Configuration Write Ticket Check
Replication Configuration	mongodb_psa_architecture_check	Raises an error if the replicaSet is utilizing a PSA (Primary-Secondary-Arbiter) architecture.	MongoDB PSA Architecture
Replication Configuration	mongodb_replicaset_topology	Warns if the Replica Set has less than three data-bearing nodes.	MongoDB Replica Set Topology
Resources Configuration	mongodb_collection_fragmented	Warns if the storage size exceeds the data size of a collection, indicating potential fragmentation. This suggests the need for compaction or an	MongoDB Collections Fragmented

Percona LLC and/or its affiliates ©2024

Advisor	Check Name	Description	Summary
		initial sync to reclaim disk space.	
Resources Configuration	mongodb_cpucores	Warns if the number of CPU cores does not meet the minimum recommended requirements according to best practices.	MongoDB CPU Cores
Resources Configuration	mongodb_dbpath_mount	Warns if dbpath does not have a dedicated mount point.	MongoDB - Separate Mount Point Other Than “/” Partition for dbpath.
Resources Configuration	mongodb_fcv_check	Warns if there is a mismatch between the MongoDB version and the internal FCV (Feature Compatibility Version) parameter setting.	MongoDB - FCV Mismatch
Resources Configuration	mongodb_maxsessions	Warns if MongoDB is configured with a maxSessions value other than the default value of 1000000.	MongoDB maxSessions
Resources Configuration	mongodb_swap_allocation	Warns if there is no swap memory allocated to your instance.	MongoDB - Allocate Swap Memory
Resources Configuration	mongodb_taskexecutor	Warns if the count of MongoDB TaskExecutorPoolSize exceeds the number of available CPU cores.	MongoDB TaskExecutorPoolSize High
Resources Configuration	mongodb_xfs_ftype	Warns if dbpath is not using the XFS filesystem type.	MongoDB - XFS
Version Configuration	mongodb_EOL	Raises an error or a warning if your current PSMDB or MongoDB version has reached or is nearing its End-of-Life (EOL) status.	MongoDB Version EOL
Version Configuration	mongodb_unsupported_version	Raises an error if your current PSMDB or MongoDB version is not supported.	MongoDB Unsupported Version
Version Configuration	mongodb_version	Provides information on current MongoDB or Percona Server for	MongoDB Version Check

Percona LLC and/or its affiliates ©2024

Advisor	Check Name	Description	Summary
		MongoDB versions used in your environment. It also offers details on other available minor or major versions that you may consider for upgrades.	
Generic Performance	mongodb_multiple_services	Warns if multiple mongod services are detected running on a single node.	MongoDB - Multiple mongod Services
Replication Performance	mongodb_chunk_imbalance	Warns if the distribution of chunks across shards is imbalanced.	MongoDB Sharding - Chunk Imbalance Across Shards
Replication Performance	mongodb_oplog_size_recommendation	Warns if the oplog window is below a 24-hour period and provides a recommended oplog size based on your instance.	MongoDB - Oplog Recovery Window is Low
Replication Performance	mongodb_replication_lag	Warns if the replica set member lags behind the primary by more than 10 seconds.	MongoDB Replication Lag
Index Query	mongodb_shard_collection_inconsistent_indexes	Warns if there are inconsistent indexes across shards for sharded collections. Missing or inconsistent indexes across shards can have a negative impact on performance.	MongoDB Sharding - Inconsistent Indexes Across Shards
Index Query	mongodb_unused_index	Warns if there are unused indexes on any database collection in your instance. This requires enabling the “indexStats” collector.	MongoDB - Unused Indexes
Authentication Security	mongodb_auth	Warns if MongoDB authentication is disabled.	MongoDB Authentication
Authentication Security	mongodb_localhost_auth_bypass	Warns if MongoDB localhost bypass is enabled.	MongoDB localhost authentication bypass enabled
Configuration Security	mongodb_authmech_scramsha256	Warns if MongoDB is not using the default SHA-256 hashing	MongoDB Security AuthMech Check

Percona LLC and/or its affiliates ©2024

Advisor	Check Name	Description	Summary
		function as its SCRAM authentication method.	
Connection Security	mongodb_bindip	Warns if the MongoDB network binding is not set as Recommended.	MonogDB IP Bindings
CVE Security	mongodb_cve_version	Shows an error if MongoDB or Percona Server for MongoDB version is older than the latest version containing CVE (Common Vulnerabilities and Exposures) fixes.	MongoDB CVE Version

MYSQL

Advisor	Check Name	Description	Summary
Connection Configuration	mysql_configuration_max_connections_usage	Checks the MySQL max_connections configuration option to ensure maximum utilization is achieved.	Check Max Connections Usage
Generic Configuration	mysql_automatic_sp_privileges_enabled	Checks if the automatic_sp_privileges configuration is ON.	Checks if automatic_sp_privi configuration is ON
Generic Configuration	mysql_config_binlog_retention_period	Checks whether binlogs are being rotated too frequently, which is not recommended, except in very specific cases.	Binlogs Retention Check
Generic Configuration	mysql_config_binlog_row_image	Advises when to set binlog_row_image=FULL.	Binlogs Raw Image Not Set to FULL
Generic Configuration	mysql_config_binlogs_checksummed	Advises when to set binlog_checksum=CRC32 to improve consistency and reliability.	Server is Not Configured to Enfo Data Integrity
Generic Configuration	mysql_config_general_log	Checks whether the general log is enabled.	General Log is Ena
Generic Configuration	mysql_config_log_bin	Checks whether the binlog is enabled or disabled.	Binary Log is disab
Generic Configuration	mysql_config_sql_mode	Checks whether the server has specific values configured in sql_mode to ensure maximum data integrity.	Server is Not Configured to Enfo Data Integrity
Generic Configuration	mysql_config_tmp_table_size_limit	Checks whether the size of temporary tables exceeds the size of heap tables.	Temp Table Size is Larger Than Heap Table Size
Generic Configuration	mysql_configuration_log_verbosity	Checks whether warnings are being printed on the log.	Check Log Verbosi
Generic Configuration	mysql_test_database	Notifies if there are database named ‘test’ or ‘test_%’.	MySQL Test Datab

Percona LLC and/or its affiliates ©2024

Advisor	Check Name	Description	Summary
Generic Configuration	mysql_timezone	Verifies whether the time zone is correctly loaded.	MySQL configurati check
InnoDB Configuration	innodb_redo_logs_not_sized_correctly	Reviews the InnoDB redo log size and provides suggestions if it is configured too low.	InnoDB Redo Log S is Not Configured Correctly.
InnoDB Configuration	mysql_ahi_efficiency_performance_basic_check	Checks the efficiency and effectiveness of InnoDB's Adaptive Hash Index (AHI).	InnoDB Adaptive H Index (AHI) Efficie
InnoDB Configuration	mysql_config_innodb_redolog_disabled	Warns when the MySQL InnoDB Redo log is set to OFF, which poses a significant security risk and compromises data integrity. The MySQL InnoDB Redo log is a crucial component for maintaining the ACID (Atomicity, Consistency, Isolation, Durability) properties in MySQL databases.	Redo Log is Disabl This Instance
InnoDB Configuration	mysql_configuration_innodb_file_format	Verifies whether InnoDB is configured with the recommended file format.	MySQL InnoDB Fil Format
InnoDB Configuration	mysql_configuration_innodb_file_maxlimit	Checks whether InnoDB is configured with the recommended auto-extend settings.	InnoDB Tablespace Size Has a Maximu Limit.
InnoDB Configuration	mysql_configuration_innodb_file_per_table_not_enabled	Warns when innodb_file_per_table is not enabled.	innodb_file_per_tab Not Enabled
InnoDB Configuration	mysql_configuration_innodb_flush_method	Checks whether InnoDB is configured with the recommended flush method.	MySQL InnoDB Flu Method
InnoDB Configuration	mysql_configuration_innodb_strict_mode	Warns about password lifetime.	InnoDB strict mode
Replication Configuration	mysql_config_relay_log_purge	Identifies whether a replica node has relay-logs purge set.	Automatic Relay Lo Purging is OFF
Replication Configuration	mysql_config_replication_bp1	Identifies whether a replica node is in read-only mode and if *checksum* is enabled.	Checks Basic Best Practices When Se Replica Node.
Replication Configuration	mysql_config_slave_parallel_workers	Identifies whether replication is single-threaded.	Replication is Sing Threaded
Replication Configuration	mysql_config_sync_binlog	Checks whether the binlog is synchronized before a transaction is committed.	Sync Binlog Disabl
Replication Configuration	mysql_log_replica_updates	Checks if a replica is safely logging replicated transactions.	MySQL Configurat Check
Replication Configuration	replica_running_skipping_errors_or_idempotent_mode	Reviews replication status to check if it is configured to skip errors or if the slave_exec_mode is set to be *idempotent*.	Replica is skipping errors or slave_exec_mode is Idempotent.

Percona LLC and/or its affiliates ©2024

Advisor	Check Name	Description	Summary
Resources Configuration	mysql_32binary_on_64system	Notifies if version_compile_machine equals i686.	Check if Binaries a 32 Bits
Version Configuration	mysql_unsupported_version_check	Warns against an unsupported Mysql version.	Checks Mysql Vers
Version Configuration	mysql_version	Warns if MySQL, Percona Server for MySQL, or MariaDB version is not the latest available one.	MySQL Version
Version Configuration	mysql_version_eol_57	Checks if the server version is EOL.	End Of Life Server Version (5.7).
Index Query	mysql_performance_temp_ondisk_table_high	Warns if there are too many on-disk temporary tables being created due to unoptimized query execution.	Too Many on Disk Temporary Tables
Index Query	mysql_tables_without_pk	Checks tables without primary keys.	MySQL check for a table without Prim Key
Schema Design Query	mysql_indexes_larger	Check all the tables to see if any have indexes larger than data. This indicates a sub-optimal schema and should be reviewed.	Tables With Index S Larger Than Data
Authentication Security	mysql_automatic_expired_password	Warns if the MySQL parameter for automatic password expiry is not active.	MySQL Automatic Expired Password
Authentication Security	mysql_security_anonymous_user	Verifies if anonymous users are present, as this would contradict security best practices.	Anonymous User (Y Must Remove Any Anonymous User)
Authentication Security	mysql_security_open_to_world_host	Checks whether host definitions are set as '%' since this is overly permissive and could pose security risks.	UserS Have Host Definition '%' Whic Too Open
Authentication Security	mysql_security_root_not_local	Checks whether the root user has a host definition that is not set to 127.0.0.1 or localhost.	Root User Can Con From Non-local Location
Authentication Security	mysql_security_user_ssl	Reports users who are not using a secure SSL protocol to connect.	Users Not Using Secure SSL
Authentication Security	mysql_security_user_super_not_local	Reports users with super privileges who are not connecting from the local host or the host is not fully restricted (e.g., 192.168.%).	Users have Super privileges With Re and Too Open Acce
Authentication Security	mysql_security_user_without_password	Reports users without passwords.	Users Without Password
Configuration Security	mysql_config_local_infile	Checks if the "LOAD DATA INFILE" functionality is active.	Load Data in File Active
Configuration Security	mysql_configuration_secure_file_priv_empty	Warns when secure_file_priv is empty as this enables users with FILE privilege to create	secure_file_priv is Empty

Percona LLC and/or its affiliates ©2024

Advisor	Check Name	Description	Summary
		files at any location where MySQL server has Write permission.	
Configuration Security	mysql_password_expiry	Checks if MySQL user passwords are expired or expiring within the next 30 days.	Check MySQL Use Password Expiry
Configuration Security	mysql_require_secure_transport	Checks the status of *mysql_secure_transport_only*.	MySQL configurati check
Configuration Security	mysql_security_password_lifetime	Warns about password lifetime.	InnoDB Password Lifetime
Configuration Security	mysql_security_password_policy	Checks for password policy.	MySQL Security Cl for Password
Connection Security	mysql_private_networks_only	Notifies about MySQL accounts that are allowed to connect from public networks.	MySQL Users With Granted Public Networks Access
Replication Security	mysql_replication_grants	Checks if replication is configured on a node without user grants.	MySQL Security Cl for Replication Use
Replication Security	mysql_security_replication_grants_mixed	Checks if replication privileges are mixed with more elevated privileges.	Replication Privileg

POSTGRESQL

Advisor	Check Name	Description
Connection Configuration	postgresql_max_connections_1	Notifies if the *max_connections* configuration option is set to a high value (above 300). PostgreSQL doesn't cope well with having many connections even if they are idle. The recommended value is below 300.
Generic Configuration	postgresql_archiver_failing_1	Verifies if the archiver has failed.
Generic Configuration	postgresql_fsync_1	Returns an error if the *fsync* configuration option is set to OFF, as this can lead to database corruptions.
Generic Configuration	postgresql_log_checkpoints_1	Notifies if the *log_checkpoints* configuration option is not enabled. It is recommended to enable the logging of checkpoint information, as that provides a lot of useful information with almost no drawbacks.
Generic Configuration	postgresql_logging_recommendation_checks	Verifies whether the recommended minimum logging features are enabled.
Generic Configuration	postgresql_wal_retention_check	Checks if there are too many WAL files retained in the WAL directory.
Vacuum Configuration	postgresql_log_autovacuum_min_duration_1	Notifies if the *log_autovacuum_min_duration configuration* option is set to -1 (disabled). It is recommended to enable the logging of autovacuum run information, as it provides a lot of useful information with almost no drawbacks.
Vacuum Configuration	postgresql_table_autovac_settings	Returns tables where autovacuum parameters are specified along with the corresponding autovacuum settings.

Percona LLC and/or its affiliates ©2024

Advisor	Check Name	Description
Vacuum Configuration	postgresql_txid_wraparound_approaching	Verifies the age of databases and alerts if the transaction ID wraparound issue is nearing.
Vacuum Configuration	postgresql_vacuum_sanity_check	This performs a quick check of some vacuum parameters.
Version Configuration	postgresql_eol_check	Checks if the currently installed PostgreSQL version has reached its EOL and is no longer supported.
Version Configuration	postgresql_extension_check	Lists outdated extensions with newer versions available.
Version Configuration	postgresql_unsupported_check	Verifies if the currently installed version is supported by Percona.
Version Configuration	postgresql_version_check	Checks if the currently installed version is outdated for its release level.
Generic Performance	postgresql_cache_hit_ratio_1	Checks the hit ratio of one or more databases and raises a complaint when they are too low.
Generic Performance	postgresql_config_changes_need_restart_1	Warns if there are any settings or configurations that have been changed and require a server restart or reload.
Generic Performance	postgresql_tmpfiles_check	Reports the number of temporary files and the number of bytes written to disk since the last statistics reset.
Replication Performance	postgresql_stale_replication_slot_1	Warns if there is a stale replication slot. Stale replication slots will lead to WAL file accumulation and can result in a database server outage.
Vacuum Performance	postgresql_table_bloat_bytes	Verifies the size of the table bloat in bytes across all databases and raises alerts accordingly.
Vacuum Performance	postgresql_table_bloat_in_percentage	Verifies the size of the table bloat in the percentage of the total table size and alerts accordingly.
Index Query	postgresql_number_of_index_check	Lists relations with more than ten indexes.
Index Query	postgresql_sequential_scan_check	Checks for tables with excessive sequential scans.
Index Query	postgresql_unused_index_check	Lists relations with indexes that have not been used since the statistics were last reset.
Authentication Security	postgresql_super_role	Notifies if there are users with Superuser role.
Configuration Security	postgresql_expiring_passwd_check	Checks for passwords that are expiring and displays the time left before they expire.
CVE Security	postgresql_cve_check	Checks if the currently installed version has reported security vulnerabilities.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2024-04-01

Percona LLC and/or its affiliates ©2024

5. How to

5.1 How to

- Configure via the PMM Settings page.
- Manage users via the PMM Users page.
- Upgrade PMM Server via the user interface.
- Secure your PMM installation.
- Optimize the performance of your PMM installation.
- Set up PMM in HA mode.
- Annotate charts to mark significant events.
- Share dashboards and panels to save or share.
- Extend Metrics with textfile collector.
- Troubleshoot

5.1.1 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

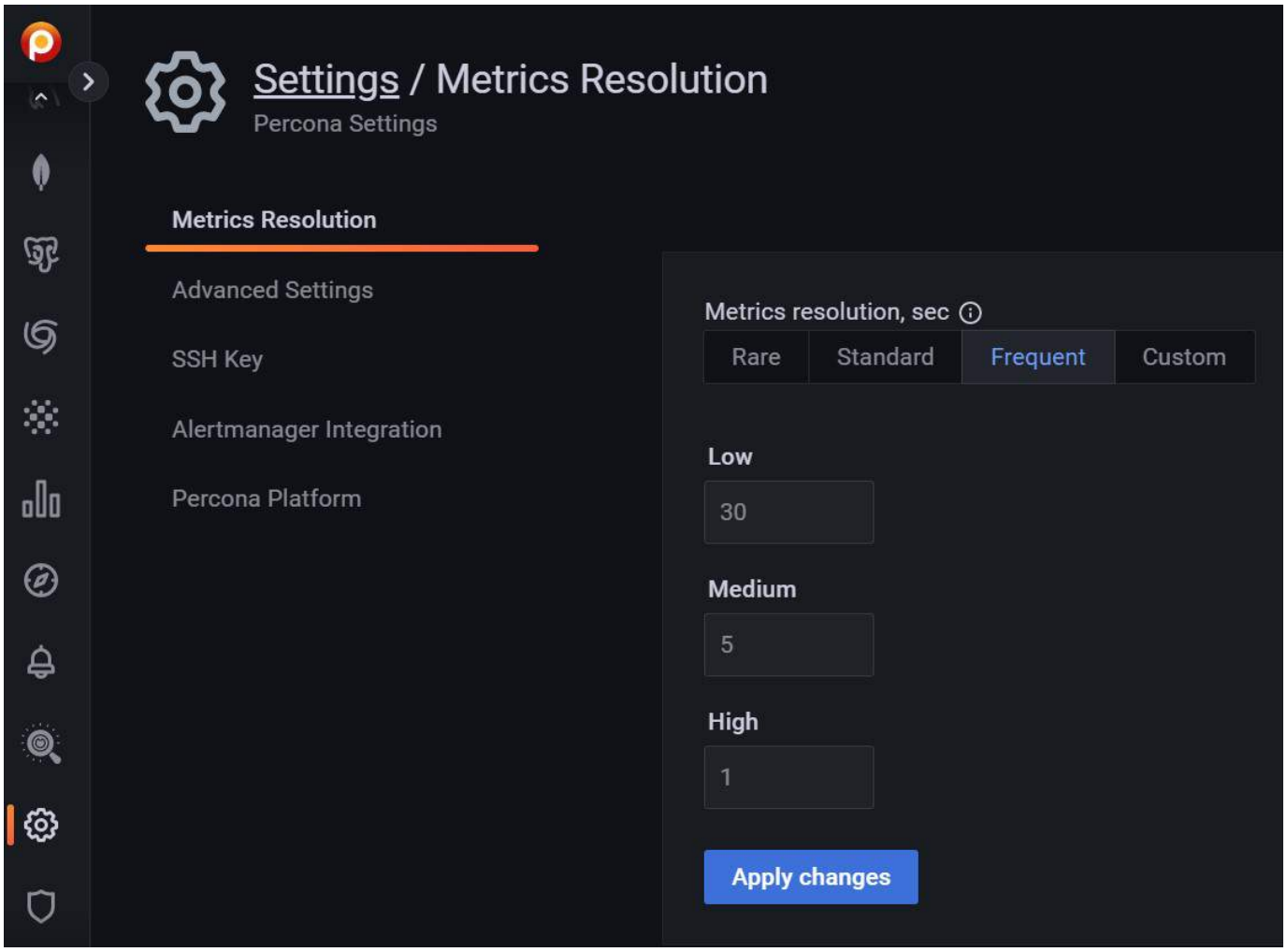
2023-12-12

5.2 Configure

The **Settings** page is where you configure PMM.

Open the **Settings** page from the main menu with **Configuration →** **Settings**. The page opens with the **Metrics Resolution** settings tab selected.

Percona LLC and/or its affiliates ©2024



On the left are the selector tabs:

- Configure
- Metrics resolution
  - Configure metrics resolution per-service
- Advanced Settings
  - Data Retention
  - Telemetry
  - Check for updates
  - Advisors
  - Percona Alerting
  - Backup Management
  - Public Address
  - Database as a Service (DBaaS)
  - Microsoft Azure monitoring
- SSH Key
- Alertmanager integration
- Percona Platform

Percona LLC and/or its affiliates ©2024

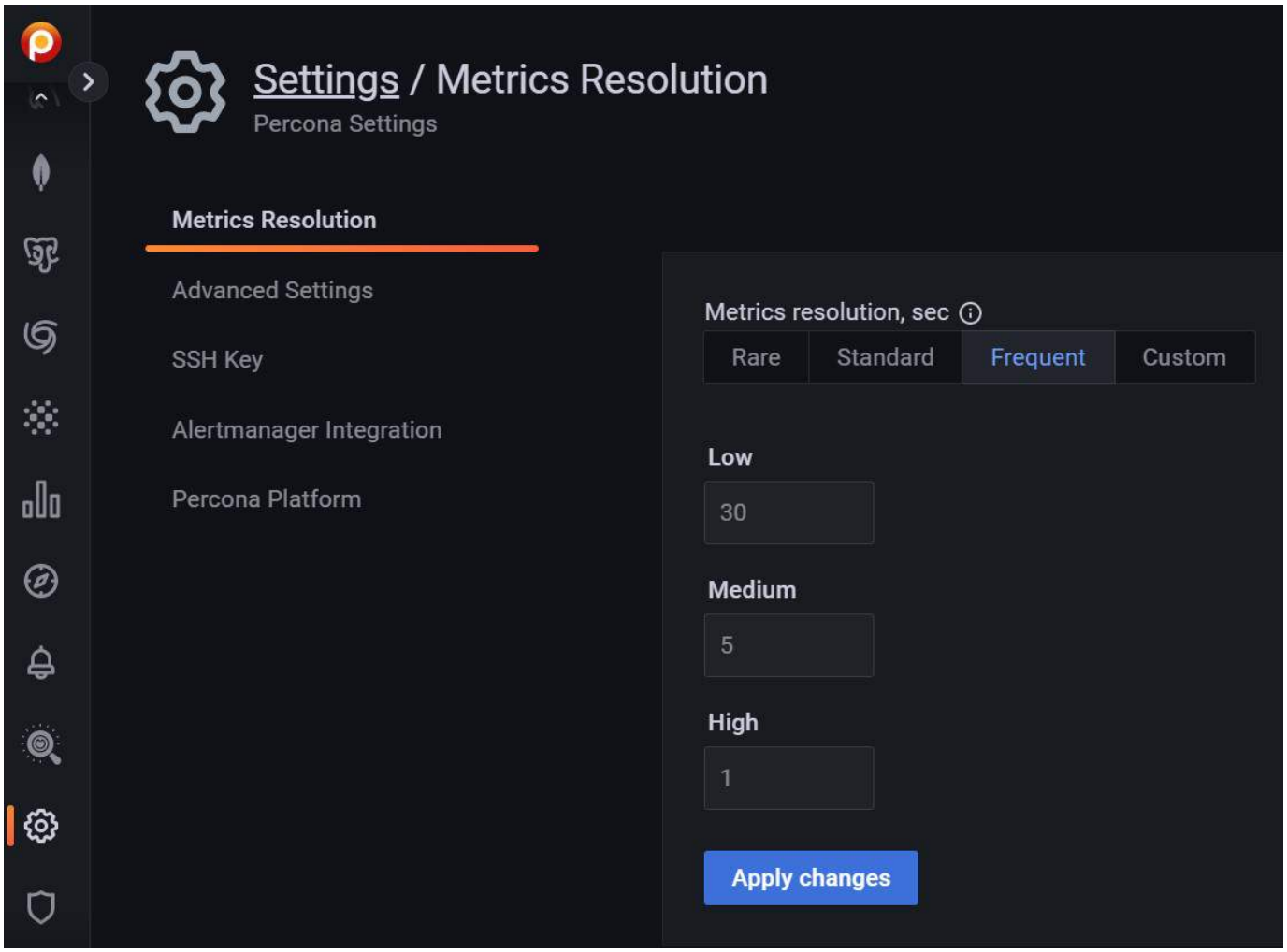
Tip

Click *Apply changes* to save any changes made here.

5.2.1 Metrics resolution

Metrics are collected at three intervals representing low, medium and high resolutions.

The *Metrics Resolution* settings tab contains a radio button with three fixed presets (*Rare*, *Standard* and *Frequent*) and one editable custom preset (*Custom*).



Each preset is a group of low, medium and high resolutions. The values are in seconds.

Time intervals and resolutions

Short time intervals are *high* resolution metrics. Longer time intervals are *low* resolution. So:

- A low-resolution interval *increases* the time between collection, resulting in low-resolution metrics and lower disk usage.
- A high-resolution interval *decreases* the time between collection, resulting in high-resolution metrics and higher disk usage.

The default values (in seconds) for the fixed presets and their resolution names are:

Percona LLC and/or its affiliates ©2024

Editable?	Preset	Low	Medium	High
No	Rare	300	180	60
No	Standard	60	10	5
No	Frequent	30	5	1
Yes	Custom (defaults)	60	10	5

Values for the *Custom* preset can be entered as values, or changed with the arrows.

If there is poor network connectivity between PMM Server and PMM Client, or between PMM Client and the database server being monitored, scraping every second may not be possible when the network latency is greater than 1 second.

Configure metrics resolution per-service

While changing the metrics resolution in the settings tab applies to all services, you can also configure metrics resolutions on a per-service basis by setting the resolution settings for each exporter individually via the API.

Customizing the resolution settings for individual services enables you to fine-tune your PMM setup to balance data granularity and resource consumption.

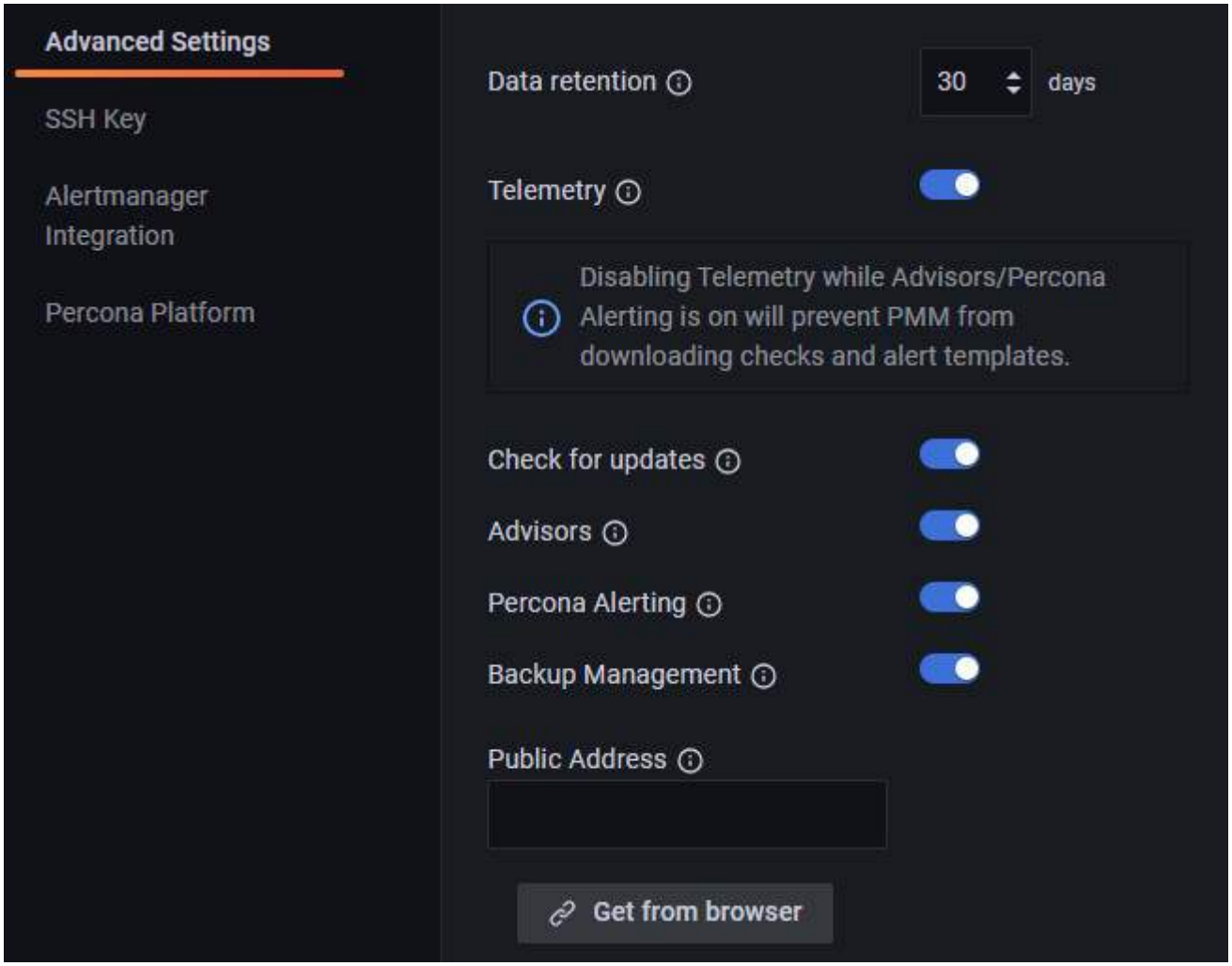
This enables you to:

- Allocate resources efficiently by focusing on high-resolution data for key services
- Reduce storage requirements by adjusting resolution for less critical components
- Align your monitoring setup with the specific needs of your environment

For information on enabling this feature via API, see the **Change Postgres Exporter endpoint** in the API documentation.

Percona LLC and/or its affiliates ©2024

5.2.2 Advanced Settings



Data Retention

Data retention specifies how long data is stored by PMM Server. By default, time-series data is stored for 30 days. You can adjust the data retention time to balance your system's available disk space with your metrics history requirements.

Telemetry

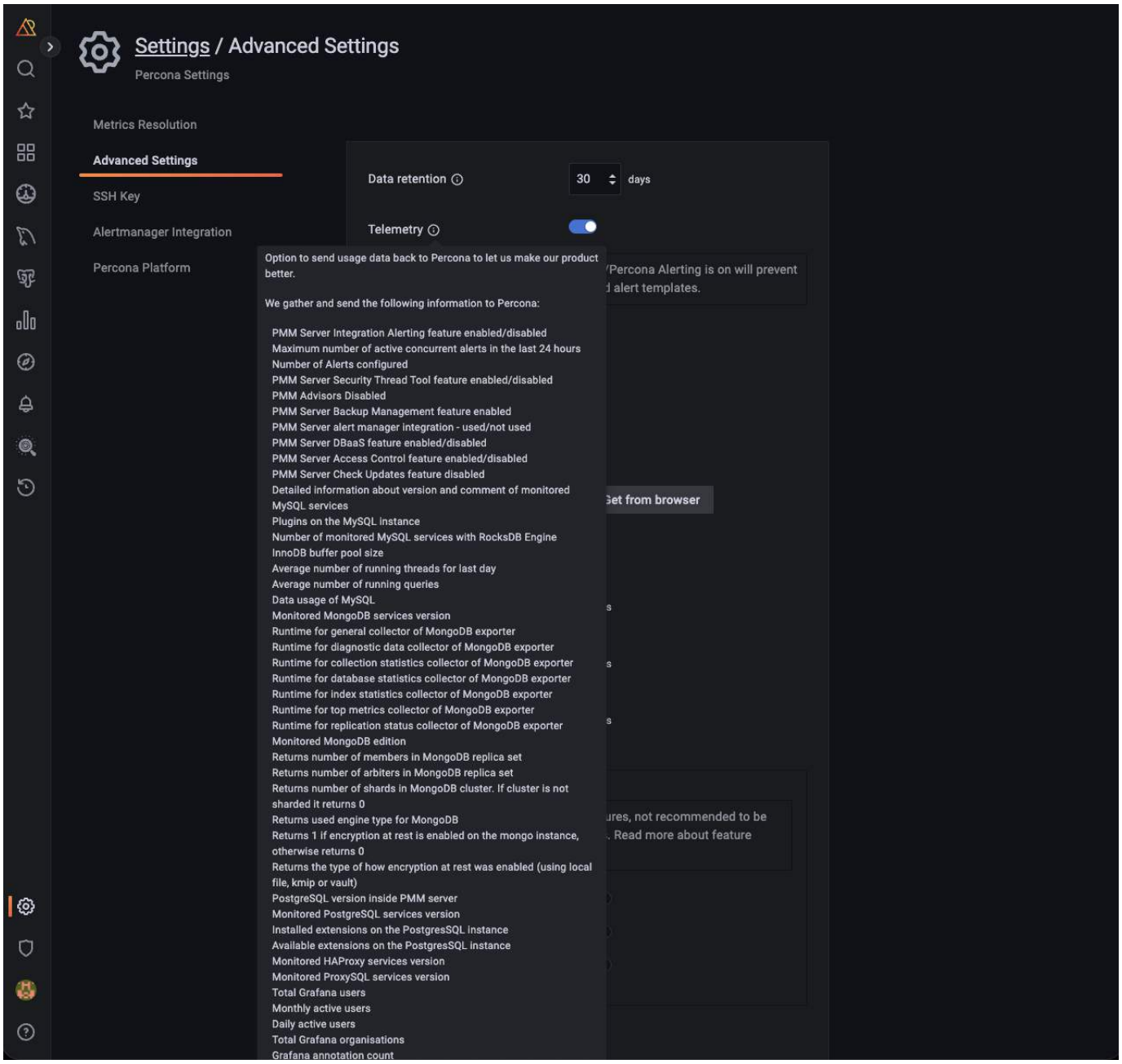
The *Telemetry* switch enables gathering and sending basic **anonymous** data to Percona, which helps us to determine where to focus the development and what is the uptake for each release of PMM. Specifically, gathering this information helps determine if we need to release patches to legacy versions beyond support, determine when supporting a particular version is no longer necessary, and understand the best frequency of releases.

PMM Telemetry is based on data collected by various PMM components and stored inside PMM server

> When PMM is installed, telemetry is not sent immediately. Before the first telemetry report is generated, PMM provides users with a 24-hour grace period to disable telemetry.

To see the metrics being collected by telemetry, from the main menu navigate to *Configuration → Settings → Advanced Settings → Telemetry* and hover over the exclamation mark.

Percona LLC and/or its affiliates ©2024



We do not gather anything that can identify your system, but consider the following:

1. The Country Code is evaluated from the submitting IP address before being discarded.
2. We do create an “instance ID” - a random string generated using UUID v4. This instance ID is generated to distinguish new instances from existing ones, for figuring out instance upgrades.

The first telemetry reporting of a new PMM Server instance is delayed by 24 hours to allow enough time to disable the service for those that do not wish to share any information.

The landing page for this service, check.percona.com, explains what this service is.

Grafana’s anonymous usage statistics is not managed by PMM. To activate it, you must change the PMM Server container configuration after each update.

As well as via the *PMM Settings* page, you can also disable telemetry with the `-e DISABLE_TELEMETRY=1` option in your docker run statement for the PMM Server.

Percona LLC and/or its affiliates ©2024

For information on the various config parameters for telemetry, see the config file.

Check for updates

When active, PMM will automatically check for updates and put a notification in the home page *Updates* dashboard if any are available.

Advisors

Advisors are sets of checks grouped by functionality that run a range of database health checks on a registered PMM instance.

The findings are reported on the **Advisors > Advisor Insights** page, and an overview is displayed on the Home dashboard.

The Advisors option is enabled by default. Checks are re-fetched and rerun at intervals.

See Working with Advisor checks.

Percona Alerting

Enables Percona Alerting and reveals the **Percona templated alerts** option on the **Alerting** page.

Backup Management

Enables Backup Management option and reveals the **Backup** page from where you can:

- Create and restore MongoDB and MySQL backups
- Automate backup scheduling
- Set retention policies
- Monitor your backup and restore activity

Public Address

The address or hostname PMM Server will be accessible at. Click **Get from browser** to have your browser detect and populate this field automatically.

Database as a Service (DBaaS)

> **Caution**
>
> DBaaS functionality is a technical preview that must be turned on with a server feature flag. See DBaaS.

Enables/disables DBaaS features on this server.

> **Important**
>
> Deactivating DBaaS ***does not*** suspend or remove running DB clusters.

Percona LLC and/or its affiliates ©2024

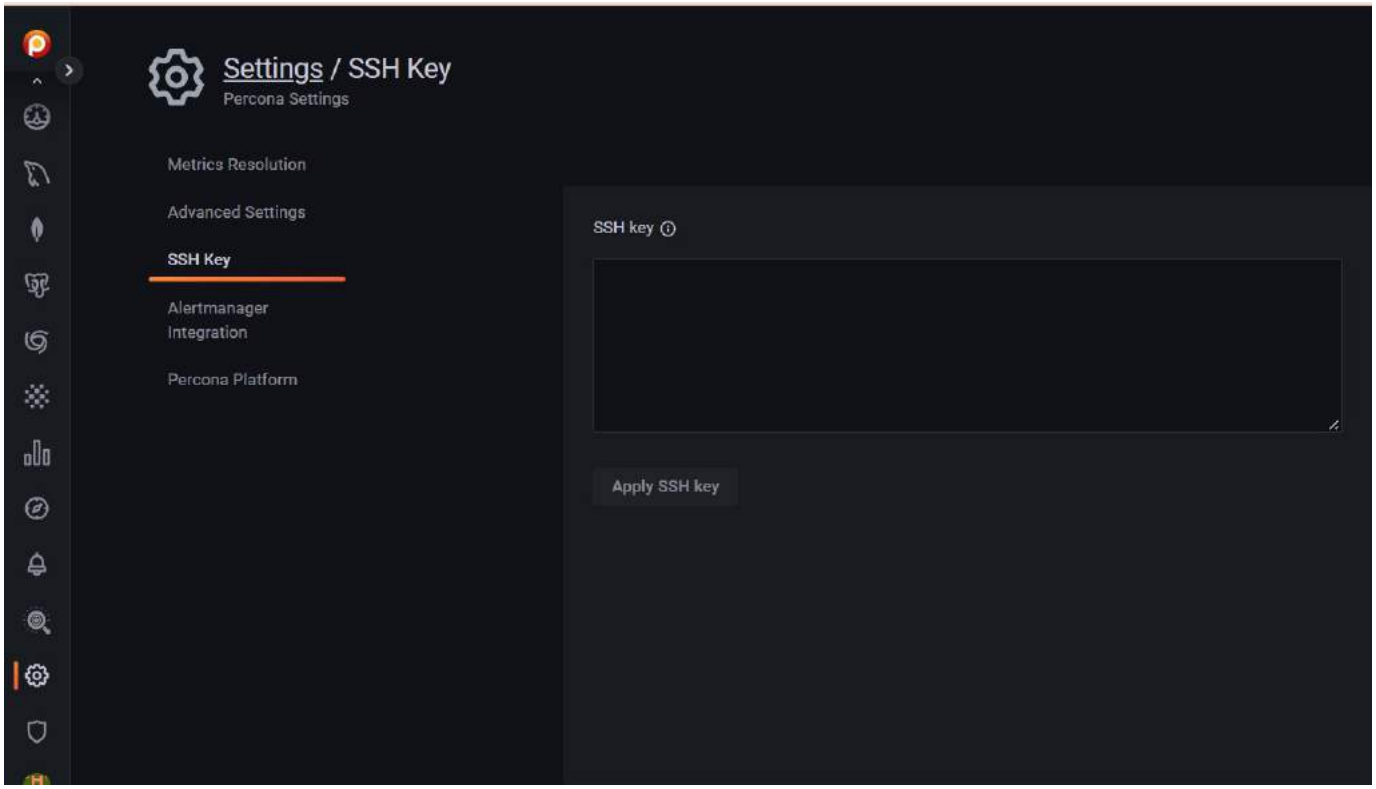
Microsoft Azure monitoring

> **Caution**
>
> This is a technical preview feature.

Activates Microsoft Azure monitoring.

5.2.3 SSH Key

This section lets you upload your public SSH key to access the PMM Server via SSH (for example, when accessing PMM Server as a virtual appliance).



Enter your **public key** in the *SSH Key* field and click *Apply SSH Key*.

5.2.4 Alertmanager integration

Alertmanager manages alerts, de-duplicating, grouping, and routing them to the appropriate receiver or display component.

This section lets you configure how VictoriaMetrics integrates with an external Alertmanager.

> **Tip**
>
> If possible, use Integrated Alerting instead of Alertmanager.

- The *Alertmanager URL* field should contain the URL of the Alertmanager which would serve your PMM alerts.
- The *Prometheus Alerting rules* field is used to specify alerting rules in the YAML configuration format.

Percona LLC and/or its affiliates ©2024

Fill in both fields and click the *Apply Alertmanager settings* button to proceed.

5.2.5 Percona Platform

This is where you connect your PMM server to your Percona Platform Account.

To learn how to connect your PMM servers to Percona Platform and leverage Platform services that boost the monitoring capabilities of your PMM installations, see Integrate PMM with Percona Platform.

> Your Percona Platform Account is separate from your PMM User account.

5.2.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

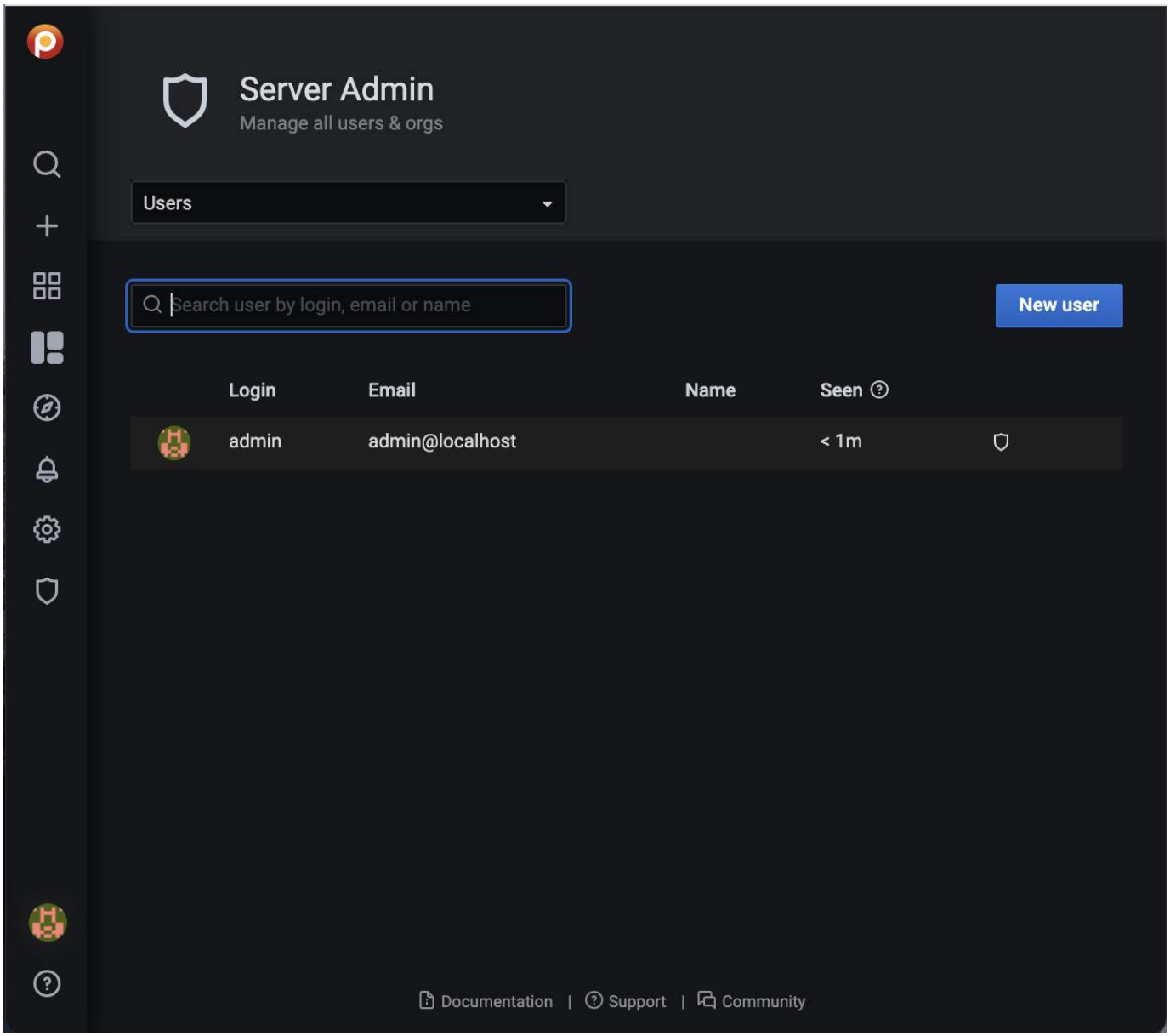
2024-06-11

5.3 Manage users

This topic explains user management in PMM.

You can manage users from the main menu by navigating to *Server Admin* → *Users* page.

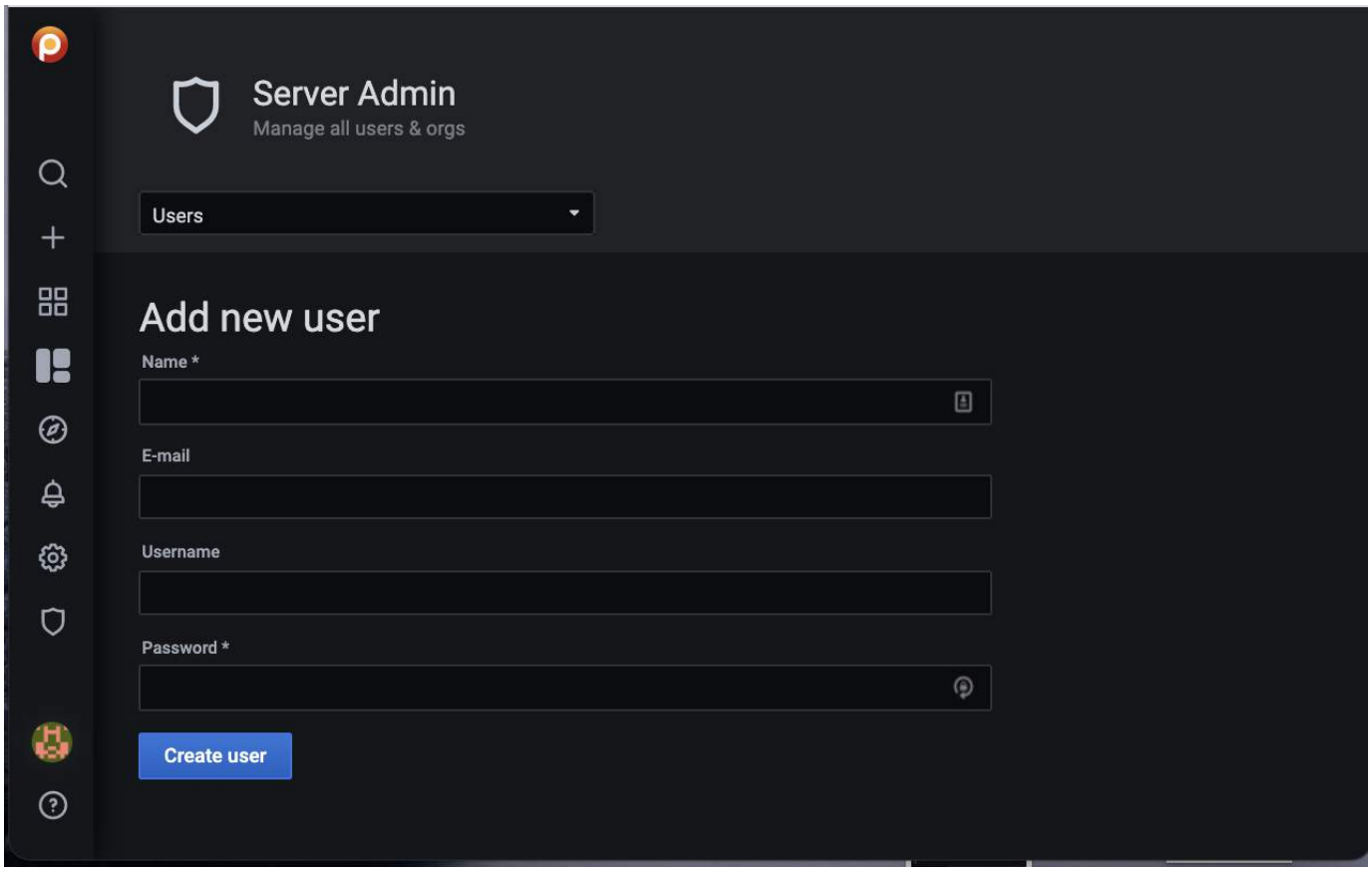
Percona LLC and/or its affiliates ©2024



5.3.1 Add users

You can add a user in PMM from *User → New user* tab.

Percona LLC and/or its affiliates ©2024



To add a new user in PMM:

1. On the *Users* tab, click *New user*.
2. On the *Add new user* dialog box, enter the following:
    - Name
    - email address or username (if this is an existing grafana user)
    - Username
    - Password
3. Click *create user*.

5.3.2 Edit users

You can edit users by changing the information or settings for an individual user account.

> **Important**
>
> After changing the default admin password for the PMM server, register the pmm-agent using the same credentials and add the services again. Otherwise, PMM will cease to monitor the service/nodes.

Grant or Revoke admin privileges

You can grant or revoke admin access to a user as follows:

1. On the *Users* tab, click the user account you want to edit.
2. To grant or revoke the privileges, click the user. User information dialog box opens.

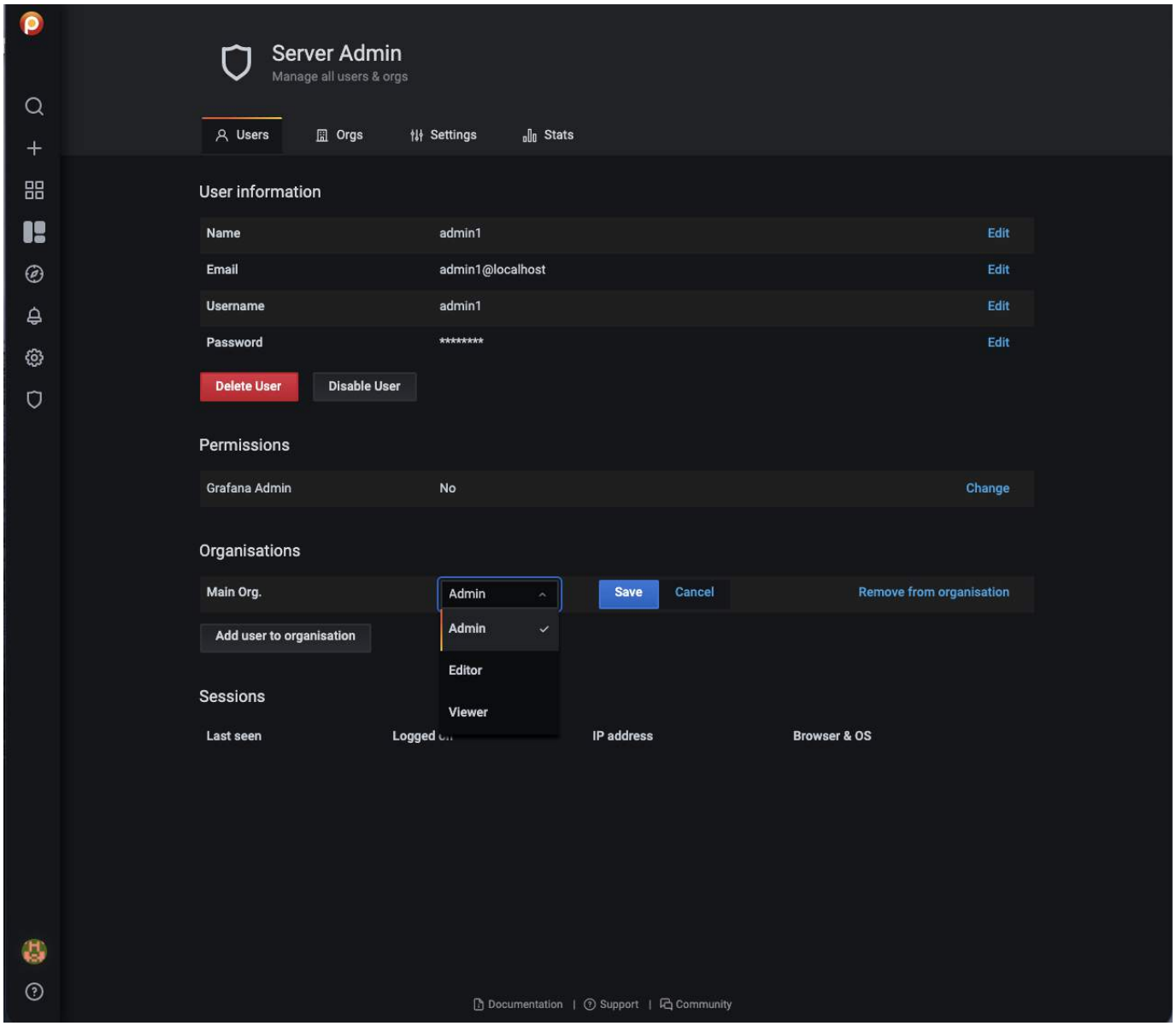
Percona LLC and/or its affiliates ©2024

3. In the *Permissions* section, click *Change* and then select *Yes/No*, depending on whether you want to provide admin access or not.
4. Click *Change*.

> **Important**
>
> After connecting your PMM instance to the Percona Platform, when you log in using your Percona account, you will be granted the *Viewer* access. For *Admin* access, log in to PMM as an admin, and change the permissions for this user.

Change organization role

You can change the organization role assigned to your user account.



To change the role:

1. On the *Users* tab, click the user for whom you want to change the role.
2. In the *Organisations* section, click *Change role*.

Percona LLC and/or its affiliates ©2024

3. Select the role from the drop-down and click *save*.

The following are the privileges for the various roles:

- Admin - Managing data sources, teams, and users within an organization.
- Editor - Creating and editing dashboards.
- Viewer - Viewing dashboards.

For detailed information on the privileges for these roles and the different tasks that they can perform, refer to: Grafana organization roles.

5.3.3 Delete Users

You can delete a user in PMM as follows:

1. On the *User* tab, click the user you want to delete.
2. Click *Delete user*.

5.3.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-18

5.4 Upgrade

5.4.1 Plan the upgrade order

Upgrade PMM Server first

Make sure to upgrade the PMM Server before you upgrade the PMM Client.

Ensure that the PMM Server version is higher than or equal to the PMM Client version. Otherwise, there might be configuration issues, thus leading to failure in the client-server communication as PMM Server might not be able to identify all the parameters in the configuration.

For example, for a PMM Server version 2.25.0, the PMM Client version should be 2.25.0 or 2.24.0. If the PMM Client version is 2.26.0, PMM might not work as expected.

Staged upgrade approach

When upgrading PMM from older versions (2.30.0 and below), we recommend following a staged approach: first, upgrade to version 2.33.0, and then proceed to the latest version.

This sequential upgrading process ensures that PMM's internal components are migrated and updated correctly.

Percona LLC and/or its affiliates ©2024

5.4.2 Update the Server

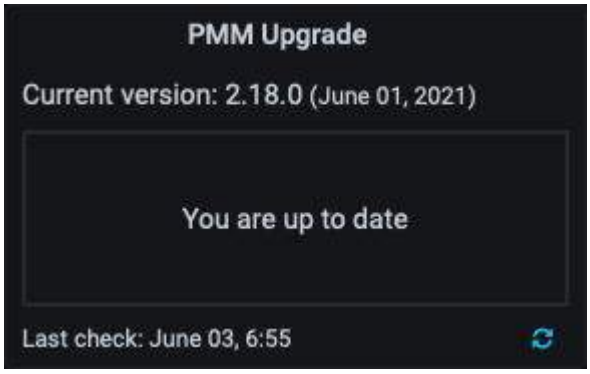
Known issues for older versions

- Upgrading to PMM 2.32.0 from older versions fails. We recommend upgrading directly to2.33 or latest version. For more information, see the troubleshooting topic.
- PMM versions prior to 2.33.0 may not show the latest versions available with instances created from the AWS marketplace in specific environments, including AWS. For solution, see the troubleshooting section.

Client and server components are installed and updated separately.

PMM Server can run natively, as a Docker image, a virtual appliance, or an AWS cloud instance. Each has its own installation and update steps.

The preferred and simplest way to update PMM Server is with the *PMM Upgrade* panel on the Home page.



The panel shows:

- the current server version and release date;
- whether the server is up to date;
- the last time a check was made for updates.

Click the refresh button to manually check for updates.

If one is available, click the update button to update to the version indicated.

See also

PMM Server Docker upgrade

5.4.3 Updating a PMM-Agent

PMM-Agent can be updated from tarball:

1. Download `tar.gz` with `pmm2-client`.
2. Extract it.
3. Run `./install_tarball` script with the `-u` flag.

Percona LLC and/or its affiliates ©2024

Important

The configuration file will be overwritten if you do not provide the `-u` flag while the `pmm-agent` is updated.

5.4.4 Upgrade from PMM 1

Because of the significant architectural changes between PMM1 and PMM2, there is no direct upgrade path. The approach to making the switch from PMM version 1 to 2 is a gradual transition, outlined in this blog post.

In short, it involves first standing up a new PMM2 server on a new host and connecting clients to it. As new data is reported to the PMM2 server, old metrics will age with the retention period (30 days, by default), at which point you'll be able to shut down your existing PMM1 server.

Any alerts configured through the Grafana UI will have to be recreated due to the target dashboard id's not matching between PMM1 and PMM2. In this instance we recommend moving to Alertmanager recipes in PMM2 for alerting which, for the time being, requires a separate Alertmanager instance. We are working on integrating this natively into PMM2 Server and expect to support your existing Alertmanager rules.

5.4.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-19

5.5 Secure

By Default, PMM ships with a self-signed certificate to enable usage out of the box. While this does enable users to have encrypted connections between clients (database clients and web/API clients) and the PMM server, it shouldn't be considered a properly secured connection. Taking the following precautions will ensure that you are truly secure:

- SSL encryption with trusted certificates to secure traffic between clients and server;
- Grafana HTTPS secure cookies

5.5.1 SSL encryption

Valid and trusted SSL certificates are needed to encrypt traffic between the client and server. Certificates can be purchased online from various sources, or some organizations generate their own trusted certificates. Regardless of which path you choose for enabling maximum security, the process to secure PMM consists of the following components:

1. Staging the files in the proper locations:
   - You can directly mount to a local directory containing the required certificates or
   - You can copy the files to the appropriate directory in your Container|AMI|OVF
2. Restarting PMM

Percona LLC and/or its affiliates ©2024

3. Ensuring the client(s) trust the certificate issuer (Ubuntu | RedHat can get you started but this is somewhat OS specific)

With our Docker, OVF and AMI images, certificates are stored in `/srv/nginx` and our self-signed certificates are staged there by default.

Mounting certificates

For container-based installation, if your certificates are in a directory called `/etc/pmm-certs` on the container host, run the following to mount that directory in the proper location so that PMM can find it when the container starts:

```
docker run -d -p 443:443 --volumes-from pmm-data \
  --name pmm-server -v /etc/pmm-certs:/srv/nginx \
  --restart always percona/pmm-server:2
```

- All certificates must be owned by root. You can do this with: `chown 0:0 /etc/pmm-certs/*`
- The mounted certificate directory (`/etc/pmm-certs` in this example) must contain the files named `certificate.crt`, `certificate.key`, `ca-certs.pem`, and `dhparam.pem`.
- For SSL encryption, the container should publish on port 443 instead of 80.

Copying certificates

If PMM Server is running as a Docker image, use `docker cp` to copy certificates. This example copies certificate files from the current working directory to a running PMM Server docker container.

```
docker cp certificate.crt pmm-server:/srv/nginx/certificate.crt
docker cp certificate.key pmm-server:/srv/nginx/certificate.key
docker cp ca-certs.pem pmm-server:/srv/nginx/ca-certs.pem
docker cp dhparam.pem pmm-server:/srv/nginx/dhparam.pem
docker exec -it pmm-server chown root.root /srv/nginx/*
```

Use trusted SSL when connecting PMM Client to PMM Server

For the new trusted certificates to take effect, you'll just need to restart the PMM server (or advanced users can restart just nginx from a shell: supervisorctl restart nginx).

You can now register clients to the PMM Server using the following:

```
pmm-admin config --server-url=https://<user>:<password>@<server IP>
```

Remember

Your client machine(s) must trust the issuer of the certificate, or you will still see "untrusted connections" messages when accessing the web interface. Thus, your client will need the `--server-insecure-tls` parameter when running the `pmm-admin config` command. Follow the instructions on your operating system to install the issuer certificate (ca-certs.pem).

In case of pmm-client running in the container, mount certificates to `/etc/pki/tls/certs`:

```
PMM_SERVER=X.X.X.X:443
docker run \
--rm \
--name pmm-client \
-e PMM_AGENT_SERVER_ADDRESS=${PMM_SERVER} \
-e PMM_AGENT_SERVER_USERNAME=admin \
-e PMM_AGENT_SERVER_PASSWORD=admin \
-e PMM_AGENT_SETUP=1 \
-e PMM_AGENT_CONFIG_FILE=config/pmm-agent.yaml \
-v /your_directory_with/certs:/etc/pki/tls/certs \
```


Percona LLC and/or its affiliates ©2024

```
--volumes-from pmm-client-data \
percona/pmm-client:2
```

5.5.2 Grafana HTTPS secure cookies

To enable:

1. Start a shell within the Docker container.

   ```
   docker exec -it pmm-server bash
   ```

2. Edit `/etc/grafana/grafana.ini`.
3. Enable `cookie_secure` and set the value to `true`.
4. Restart Grafana.

   ```
   supervisorctl restart grafana
   ```

5.5.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-04-12

5.6 Optimize

5.6.1 Improving PMM Performance with Table Statistics Options

If a MySQL instance has a lot of schemas or tables, there are two options to help improve the performance of PMM when adding instances with `pmm-admin add`:

- `--disable-tablestats`, or,
- `--disable-tablestats-limit`.

Important

- These settings are only for adding an instance. To change them, you must remove and re-add the instances.
- Only one of these options can be used when adding an instance.

5.6.2 Disable per-table statistics for an instance

When adding an instance with `pmm-admin add`, the `--disable-tablestats` option disables table statistics collection when there are more than the default number (1000) of tables in the instance.

USAGE

```
pmm-admin add mysql --disable-tablestats
```


Percona LLC and/or its affiliates ©2024

5.6.3 Change the number of tables beyond which per-table statistics is disabled

When adding an instance with `pmm-admin add`, the `--disable-tablestats-limit` option changes the number of tables (from the default of 1000) beyond which per-table statistics collection is disabled.

USAGE

```
pmm-admin add mysql --disable-tablestats-limit=<LIMIT>
```

EXAMPLE

Add a MySQL instance, disabling per-table statistics collection when the number of tables in the instance reaches 2000.

```
pmm-admin add mysql --disable-tablestats-limit=2000
```

5.6.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-07-29

5.7 Set up PMM in High Availability (HA) mode

High Availability (HA) is a critical aspect of any monitoring system, as it ensures that your monitoring infrastructure remains resilient and continues to function seamlessly, even if one or more instances encounter issues. HA implements redundant systems that are ready to take over to minimize downtime and maintain continuous visibility into the performance and health of PMM.

5.7.1 HA options PMM

Since HA can add complexity, before considering HA for PMM, keep in mind that:

- Critical systems requiring immediate response benefit from sub-second failover HA, while less critical applications with some tolerance for downtime (seconds or minutes) have more flexibility.
- PMM itself has a one-minute minimum alerting interval, so even with perfect HA, the fastest you'll know about an issue is one minute after it occurs.
- Consider your specific uptime needs, performance requirements, and potential data loss you can tolerate, while also keeping in mind PMM's limitations.

1. Simple Docker restart with data caching

The most straightforward approach to increase availability in PMM is to launch the PMM Server within Docker using the `--restart=always` flag. See Setting up PMM Server with Docker for more information.

This ensures that the PMM Server automatically restarts if a minor issue occurs. Additionally, PMM's data caching feature stores data locally on the PMM Client when the connection to the PMM Server is interrupted.

Percona LLC and/or its affiliates ©2024

Once the connection is restored, the cached data is transferred to the PMM Server, ensuring no data loss during the restart process.

This option is suitable for scenarios where the primary concern is the ability to investigate potential issues later. However, it's important to note that this approach is limited by the underlying physical infrastructure. If the failure stems from a hardware issue, automatic recovery might be challenging.

2. Leverage Kubernetes for enhanced isolation

If you are running PMM in a Kubernetes (K8s) environment, PMM offers a Helm chart that facilitates running PMM with enhanced isolation. See Setting up PMM Server with Docker.

In this setup, even if the physical infrastructure encounters a problem, K8s automatically handles failover, migrating the PMM instance to a healthy node.

While restarts within K8s can take up to several minutes (depending on your infrastructure configuration), PMM's data caching ensures that information is preserved during this transition. Alerts will still be triggered to keep you informed about any issues that started during PMM's restart and continue after PMM is back.

3. Fully-clustered PMM in Kubernetes (coming Q3/2024)

If you have a large deployment with numerous instances and distributed locations, you might find that a fully clustered PMM setup in Kubernetes is better suited to your needs. We are actively developing this solution, which is slated for release in Q3/2024, to cater specifically to users managing extensive and complex monitoring environments.

This option will provide a comprehensive HA solution, including clustered database setups (ClickHouse, VictoriaMetrics, and PostgreSQL). In this setup, multiple PMM instances will be configured, with one being the leader and the others as followers.

Leader election will be managed using the Raft consensus algorithm, ensuring a smooth transition of the leader role if the current leader fails. The architecture will consist of:

- Multiple PMM instances for redundancy
- Clustered PostgreSQL for storing metadata and configuration data
- Clustered ClickHouse for storing query performance metrics (Query Analytics)
- Clustered VictoriaMetrics for storing operational metrics from monitored databases and hosts
- HAProxy for managing and directing network traffic to the current leader PMM instance

4. Manual setup for HA

> **Important**
>
> Manual setup for HA is feature is currently in Technical Preview. Early adopters are advised to use this feature for testing purposes only as it is subject to change.

If none of the above options work for your specific use case, consider setting up PMM in HA mode manually by following the steps below.

Percona LLC and/or its affiliates ©2024

To enable communication and coordination among the PMM Server instances, two key protocols are used:

- **Gossip protocol**: Enables PMM servers to discover and share information about their states. It is used for managing the PMM server list and failure detection, ensuring that all instances are aware of the current state of the cluster.
- **Raft protocol**: Ensures that PMM servers agree on a leader and that logs are replicated among all machines to maintain data consistency.

These protocols work in tandem to ensure that the PMM Server instances can effectively store and manage the data collected from your monitored databases and systems.

In an HA configuration, three PMM Server instances are configured: one as the leader and the others as followers. The leader server handles all client requests. If the leader fails, the followers take over, minimizing downtime.

To eliminate single points of failure and provide better service level agreements (SLAs), the critical services typically bundled with PMM Server are extracted and set up as separate, clustered instances:

- ClickHouse: A clustered setup of ClickHouse is used to store Query Analytics (QAN) metrics. This ensures that QAN data remains highly available and can be accessed even if one of the ClickHouse nodes fails.
- VictoriaMetrics: A clustered setup of VictoriaMetrics is used to store Prometheus metrics. This provides a highly available and scalable solution for storing and querying metrics data.
- PostgreSQL: A clustered setup of PostgreSQL is used to store PMM data, such as inventory and settings. This ensures that PMM's configuration and metadata remain highly available and can be accessed by all PMM Server instances.

PREREQUISITES

You will need the following before you can begin the deployment:

- Docker installed and configured on your system. If you haven't installed Docker, you can follow **this guide**.

> **Note**
>
> - The sections below provide instructions for setting up the services on both the same and separate instances. However, it is not recommended to run the services on a single machine for production purposes. This approach is only recommended for the development environment.
> - It is recommended to use clustered versions of PosgreSQL, Victoriametrics, Clickhouse, etc., instead of standalone versions when setting up the services.

To set up PMM in HA mode manually:

STEP 1: DEFINE ENVIRONMENT VARIABLES

Before you start with the setup, define the necessary environment variables on each instance where the services will be running. These variables will be used in subsequent commands.

For all IP addresses, use the format `17.10.1.x`, and for all usernames and passwords, use a string format like `example`.

Variable	Description
`CH_HOST_IP`	The IP address of the instance where the ClickHouse service is running or the desired IP address for the ClickHouse container within the Docker network, depending on your setup.Example: `17.10.1.2`
`VM_HOST_IP`	The IP address of the instance where the VictoriaMetrics service is running or the desired IP address for the VictoriaMetrics container within the Docker network, depending on your setup.Example: `17.10.1.3`
`PG_HOST_IP`	The IP address of the instance where the PostgreSQL service is running or the desired IP address for the PostgreSQL container within the Docker network, depending on your setup. Example: `17.10.1.4`

Percona LLC and/or its affiliates ©2024

Variable	Description
`PG_USERNAME`	The username for your PostgreSQL server. Example: `pmmuser`
`PG_PASSWORD`	The password for your PostgreSQL server. Example: `pgpassword`
`GF_USERNAME`	The username for your Grafana database user.Example: `gfuser`
`GF_PASSWORD`	The password for your Grafana database user.Example: `gfpassword`
`PMM_ACTIVE_IP`	The IP address of the instance where the active PMM server is running or the desired IP address for your active PMM server container within the Docker network, depending on your setup.Example: `17.10.1.5`
`PMM_ACTIVE_NODE_ID`	The unique ID for your active PMM server node.Example: `pmm-server-active`
`PMM_PASSIVE_IP`	The IP address of the instance where the first passive PMM server is running or the desired IP address for your first passive PMM server container within the Docker network, depending on your setup. Example: `17.10.1.6`
`PMM_PASSIVE_NODE_ID`	The unique ID for your first passive PMM server node.Example: `pmm-server-passive`
`PMM_PASSIVE2_IP`	The IP address of the instance where the second passive PMM server is running or the desired IP address for your second passive PMM server container within the Docker network, depending on your setup.Example: `17.10.1.7`
`PMM_PASSIVE2_NODE_ID`	The unique ID for your second passive PMM server node.Example: `pmm-server-passive2`
`PMM_DOCKER_IMAGE`	The specific PMM Server Docker image for this guide.Example: `percona/pmm-server:2`

Expected output

```
export CH_HOST_IP=17.10.1.2
export VM_HOST_IP=17.10.1.3
export PG_HOST_IP=17.10.1.4
export PG_USERNAME=pmmuser
export PG_PASSWORD=pgpassword
export GF_USERNAME=gfuser
export GF_PASSWORD=gfpassword
export PMM_ACTIVE_IP=17.10.1.5
export PMM_ACTIVE_NODE_ID=pmm-server-active
export PMM_PASSIVE_IP=17.10.1.6
export PMM_PASSIVE_NODE_ID=pmm-server-passive
export PMM_PASSIVE2_IP=17.10.1.7
export PMM_PASSIVE2_NODE_ID=pmm-server-passive2
export PMM_DOCKER_IMAGE=percona/pmm-server:2
```

Note

Ensure that you have all the environment variables from Step 1 set in each instance where you run these commands.

STEP 2: CREATE DOCKER NETWORK (OPTIONAL)

1. Set up a Docker network for PMM services if you plan to run all the services on the same instance. As a result of this Docker network, your containers will be able to communicate with each other, which is essential for the High Availability (HA) mode to function properly in PMM. This step may be optional if you run your services on separate instances.
2. Run the following command to create a Docker network:

```
docker network create pmm-network --subnet=17.10.1.0/16
```


Percona LLC and/or its affiliates ©2024

STEP 3: SET UP CLICKHOUSE

ClickHouse is an open-source column-oriented database management system. In PMM, ClickHouse stores Query Analytics (QAN) metrics, which provide detailed information about your queries.

To set up ClickHouse:

1. Pull the ClickHouse Docker image.

```
docker pull clickhouse/clickhouse-server:23.8.2.7-alpine
```

2. Create a Docker volume for ClickHouse data.

```
docker volume create ch_data
```

3. Run the ClickHouse container.

Run services on same instance **Run services on a seperate instance**

```
docker run -d \
--name ch \
--network pmm-network \
--ip ${CH_HOST_IP} \
-p 9000:9000 \
-v ch_data:/var/lib/clickhouse \
clickhouse/clickhouse-server:23.8.2.7-alpine
```

```
docker run -d \
--name ch \
-p 9000:9000 \
-v ch_data:/var/lib/clickhouse \
clickhouse/clickhouse-server:23.8.2.7-alpine
```

Note

- If you run the services on the same instance, the `--network` and `--ip` flags assign a specific IP address to the container within the Docker network created in the previous step. This IP address is referenced in subsequent steps as the ClickHouse service address.
- The `--network` and `--ip` flags are not required if the services are running on separate instances since ClickHouse will bind to the default network interface.

STEP 4: SET UP VICTORIAMETRICS

VictoriaMetrics provides a long-term storage solution for your time-series data. In PMM, it is used to store Prometheus metrics.

To set up VictoriaMetrics:

1. Pull the VictoriaMetrics Docker image.

```
docker pull victoriametrics/victoria-metrics:v1.93.4
```

2. Create a Docker volume for VictoriaMetrics data.

```
docker volume create vm_data
```

3. Run the VictoriaMetrics container.

Percona LLC and/or its affiliates ©2024

You can either run all the services on the same instance or a separate instance.

Run services on same instance **Run services on a seperate instance**

```
docker run -d \
--name vm \
--network pmm-network \
--ip ${VM_HOST_IP} \
-p 8428:8428 \
-p 8089:8089 \
-p 8089:8089/udp \
-p 2003:2003 \
-p 2003:2003/udp \
-p 4242:4242 \
-v vm_data:/storage \
victoriametrics/victoria-metrics:v1.93.4 \
--storageDataPath=/storage \
--graphiteListenAddr=:2003 \
--opentsdbListenAddr=:4242 \
--httpListenAddr=:8428 \
--influxListenAddr=:8089
```

```
docker run -d \
--name vm \
-p 8428:8428 \
-p 8089:8089 \
-p 8089:8089/udp \
-p 2003:2003 \
-p 2003:2003/udp \
-p 4242:4242 \
-v vm_data:/storage \
victoriametrics/victoria-metrics:v1.93.4 \
--storageDataPath=/storage \
--graphiteListenAddr=:2003 \
--opentsdbListenAddr=:4242 \
--httpListenAddr=:8428 \
--influxListenAddr=:8089
```

Note

- If you run the services on the same instance, the `--network` and `--ip` flags are used to assign a specific IP address to the container within the Docker network created in Step 2. This IP address is referenced in subsequent steps as the VictoriaMetrics service address.
- The `--network` and `--ip` flags are not required if the services are running on separate instances, as VictoriaMetrics will bind to the default network interface.

STEP 5: SET UP POSTGRESQL

PostgreSQL is a powerful, open-source object-relational database system. In PMM, it's used to store data related to inventory, settings, and other features.

To set up PostgreSQL:

1. Pull the Postgres Docker image.

   ```
   docker pull postgres:14
   ```

2. Create a Docker volume for Postgres data:

   ```
   docker volume create pg_data
   ```

3. Create a directory to store init SQL queries:

   ```
   mkdir -p /path/to/queries
   ```

   Replace `/path/to/queries` with the path where you want to store your `init` SQL queries.

Percona LLC and/or its affiliates ©2024

4. Create an `init.sql.template` file in newly created directory with the following content:

```
CREATE DATABASE "pmm-managed";
CREATE USER <YOUR_PG_USERNAME> WITH ENCRYPTED PASSWORD '<YOUR_PG_PASSWORD>';
GRANT ALL PRIVILEGES ON DATABASE "pmm-managed" TO <YOUR_PG_USERNAME>;
CREATE DATABASE grafana;
CREATE USER <YOUR_GF_USERNAME> WITH ENCRYPTED PASSWORD '<YOUR_GF_PASSWORD>';
GRANT ALL PRIVILEGES ON DATABASE grafana TO <YOUR_GF_USERNAME>;

\c pmm-managed

CREATE EXTENSION IF NOT EXISTS pg_stat_statements;
```

5. Use **`sed`** to replace the placeholders with the environment variables and write the output to **`init.sql`**.

```
sed -e 's/<YOUR_PG_USERNAME>/'"$PG_USERNAME"'/g' \
    -e 's/<YOUR_PG_PASSWORD>/'"$PG_PASSWORD"'/g' \
    -e 's/<YOUR_GF_USERNAME>/'"$GF_USERNAME"'/g' \
    -e 's/<YOUR_GF_PASSWORD>/'"$GF_PASSWORD"'/g' \
    init.sql.template > init.sql
```

6. Run the PostgreSQL container.

   You can either run all the services on the same instance or on a seperate instance.

> **Note**
>
> It is recommended to use absolute paths instead of relative paths for volume mounts.

Run services on same instance **Run services on a seperate instance**

```
docker run -d \
  --name pg \
  --network pmm-network \
  --ip ${PG_HOST_IP} \
  -p 5432:5432 \
  -e POSTGRES_PASSWORD=${PG_PASSWORD} \
  -v /path/to/queries:/docker-entrypoint-initdb.d/ \
  -v pg_data:/var/lib/postgresql/data \
  postgres:14 \
  postgres -c shared_preload_libraries=pg_stat_statements
```

```
docker run -d \
  --name pg \
  -p 5432:5432 \
  -e POSTGRES_PASSWORD=${PG_PASSWORD} \
  -v /path/to/queries:/docker-entrypoint-initdb.d \
  -v pg_data:/var/lib/postgresql/data \
  postgres:14 \
  postgres -c shared_preload_libraries=pg_stat_statements
```

   Replace **`/path/to/queries`** with the path to your **`init.sql`** file. This command mounts the **`init.sql`** file to the **`docker-entrypoint-initdb.d`** directory, which is automatically executed upon container startup.

> **Note**
>
> - If you run the services on the same instance, the `--network` and `--ip` flags are used to assign a specific IP address to the container within the Docker network created in Step 2. This IP address is referenced in subsequent steps as the PostgreSQL service address.
> - The `--network` and `--ip` flags are not required if the services are running on separate instances, as PostgreSQL will bind to the default network interface.

Percona LLC and/or its affiliates ©2024

STEP 6: RUNNING PMM SERVICES

The PMM server orchestrates the collection, storage, and visualization of metrics. In our high-availability setup, we'll have one active PMM server and two passive PMM servers.

1. Pull the PMM Server Docker image:

```
docker pull ${PMM_DOCKER_IMAGE}
```

2. Create a Docker volume for PMM-Server data:

```
docker volume create pmm-server-active_data
docker volume create pmm-server-passive_data
docker volume create pmm-server-passive-2_data
```

3. Run the active PMM managed server. This server will serve as the primary monitoring server.

Percona LLC and/or its affiliates ©2024

You can either run all the services on the same instance or a separate instance.

Run services on same instance **Run services on a seperate instance**

```
docker run -d \
--name ${PMM_ACTIVE_NODE_ID} \
--hostname ${PMM_ACTIVE_NODE_ID} \
--network pmm-network \
--ip ${PMM_ACTIVE_IP} \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_CLICKHOUSE=1 \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_POSTGRES=1 \
-e PERCONA_TEST_PMM_CLICKHOUSE_ADDR=${CH_HOST_IP}:9000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_DATABASE=pmm \
-e PERCONA_TEST_PMM_CLICKHOUSE_BLOCK_SIZE=10000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_POOL_SIZE=2 \
-e PERCONA_TEST_POSTGRES_ADDR=${PG_HOST_IP}:5432 \
-e PERCONA_TEST_POSTGRES_USERNAME=${PG_USERNAME} \
-e PERCONA_TEST_POSTGRES_DBPASSWORD=${PG_PASSWORD} \
-e GF_DATABASE_URL=postgres://${GF_USERNAME}:${GF_PASSWORD}@${PG_HOST_IP}:5432/grafana \
-e PMM_VM_URL=http://${VM_HOST_IP}:8428 \
-e PMM_TEST_HA_ENABLE=1 \
-e PMM_TEST_HA_BOOTSTRAP=1 \
-e PMM_TEST_HA_NODE_ID=${PMM_ACTIVE_NODE_ID} \
-e PMM_TEST_HA_ADVERTISE_ADDRESS=${PMM_ACTIVE_IP} \
-e PMM_TEST_HA_GOSSIP_PORT=9096 \
-e PMM_TEST_HA_RAFT_PORT=9097 \
-e PMM_TEST_HA_GRAFANA_GOSSIP_PORT=9094 \
-e PMM_TEST_HA_PEERS=${PMM_ACTIVE_IP},${PMM_PASSIVE_IP},${PMM_PASSIVE2_IP} \
-v pmm-server-active_data:/srv \
${PMM_DOCKER_IMAGE}

docker run -d \
--name ${PMM_ACTIVE_NODE_ID} \
-p 80:80 \
-p 443:443 \
-p 9094:9094 \
-p 9096:9096 \
-p 9094:9094/udp \
-p 9096:9096/udp \
-p 9097:9097 \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_CLICKHOUSE=1 \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_POSTGRES=1 \
-e PERCONA_TEST_PMM_CLICKHOUSE_ADDR=${CH_HOST_IP}:9000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_DATABASE=pmm \
-e PERCONA_TEST_PMM_CLICKHOUSE_BLOCK_SIZE=10000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_POOL_SIZE=2 \
-e PERCONA_TEST_POSTGRES_ADDR=${PG_HOST_IP}:5432 \
-e PERCONA_TEST_POSTGRES_USERNAME=${PG_USERNAME} \
-e PERCONA_TEST_POSTGRES_DBPASSWORD=${PG_PASSWORD} \
-e GF_DATABASE_URL=postgres://${GF_USERNAME}:${GF_PASSWORD}@${PG_HOST_IP}:5432/grafana \
-e PMM_VM_URL=http://${VM_HOST_IP}:8428 \
-e PMM_TEST_HA_ENABLE=1 \
-e PMM_TEST_HA_BOOTSTRAP=1 \
-e PMM_TEST_HA_NODE_ID=${PMM_ACTIVE_NODE_ID} \
-e PMM_TEST_HA_ADVERTISE_ADDRESS=${PMM_ACTIVE_IP} \
-e PMM_TEST_HA_GOSSIP_PORT=9096 \
-e PMM_TEST_HA_RAFT_PORT=9097 \
-e PMM_TEST_HA_GRAFANA_GOSSIP_PORT=9094 \
-e PMM_TEST_HA_PEERS=${PMM_ACTIVE_IP},${PMM_PASSIVE_IP},${PMM_PASSIVE2_IP} \
-v pmm-server-active_data:/srv \
${PMM_DOCKER_IMAGE}
```

4. Run the first passive PMM managed server. This server will act as a standby server, ready to take over if the active server fails.

Percona LLC and/or its affiliates ©2024

You can either run all the services on the same instance or a separate instance.

Run services on same instance **Run services on a seperate instance**

```
docker run -d \
--name ${PMM_PASSIVE_NODE_ID} \
--hostname ${PMM_PASSIVE_NODE_ID} \
--network pmm-network \
--ip ${PMM_PASSIVE_IP} \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_CLICKHOUSE=1 \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_POSTGRES=1 \
-e PERCONA_TEST_PMM_CLICKHOUSE_ADDR=${CH_HOST_IP}:9000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_DATABASE=pmm \
-e PERCONA_TEST_PMM_CLICKHOUSE_BLOCK_SIZE=10000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_POOL_SIZE=2 \
-e PERCONA_TEST_POSTGRES_ADDR=${PG_HOST_IP}:5432 \
-e PERCONA_TEST_POSTGRES_USERNAME=${PG_USERNAME} \
-e PERCONA_TEST_POSTGRES_DBPASSWORD=${PG_PASSWORD} \
-e GF_DATABASE_URL=postgres://${GF_USERNAME}:${GF_PASSWORD}@${PG_HOST_IP}:5432/grafana \
-e PMM_VM_URL=http://${VM_HOST_IP}:8428 \
-e PMM_TEST_HA_ENABLE=1 \
-e PMM_TEST_HA_BOOTSTRAP=0 \
-e PMM_TEST_HA_NODE_ID=${PMM_PASSIVE_NODE_ID} \
-e PMM_TEST_HA_ADVERTISE_ADDRESS=${PMM_PASSIVE_IP} \
-e PMM_TEST_HA_GOSSIP_PORT=9096 \
-e PMM_TEST_HA_RAFT_PORT=9097 \
-e PMM_TEST_HA_GRAFANA_GOSSIP_PORT=9094 \
-e PMM_TEST_HA_PEERS=${PMM_ACTIVE_IP},${PMM_PASSIVE_IP},${PMM_PASSIVE2_IP} \
-v pmm-server-passive_data:/srv \
${PMM_DOCKER_IMAGE}

docker run -d \
--name ${PMM_PASSIVE_NODE_ID} \
-p 80:80 \
-p 443:443 \
-p 9094:9094 \
-p 9096:9096 \
-p 9094:9094/udp \
-p 9096:9096/udp \
-p 9097:9097 \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_CLICKHOUSE=1 \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_POSTGRES=1 \
-e PERCONA_TEST_PMM_CLICKHOUSE_ADDR=${CH_HOST_IP}:9000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_DATABASE=pmm \
-e PERCONA_TEST_PMM_CLICKHOUSE_BLOCK_SIZE=10000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_POOL_SIZE=2 \
-e PERCONA_TEST_POSTGRES_ADDR=${PG_HOST_IP}:5432 \
-e PERCONA_TEST_POSTGRES_USERNAME=${PG_USERNAME} \
-e PERCONA_TEST_POSTGRES_DBPASSWORD=${PG_PASSWORD} \
-e GF_DATABASE_URL=postgres://${GF_USERNAME}:${GF_PASSWORD}@${PG_HOST_IP}:5432/grafana \
-e PMM_VM_URL=http://${VM_HOST_IP}:8428 \
-e PMM_TEST_HA_ENABLE=1 \
-e PMM_TEST_HA_BOOTSTRAP=0 \
-e PMM_TEST_HA_NODE_ID=${PMM_PASSIVE_NODE_ID} \
-e PMM_TEST_HA_ADVERTISE_ADDRESS=${PMM_PASSIVE_IP} \
-e PMM_TEST_HA_GOSSIP_PORT=9096 \
-e PMM_TEST_HA_RAFT_PORT=9097 \
-e PMM_TEST_HA_GRAFANA_GOSSIP_PORT=9094 \
-e PMM_TEST_HA_PEERS=${PMM_ACTIVE_IP},${PMM_PASSIVE_IP},${PMM_PASSIVE2_IP} \
-v pmm-server-passive_data:/srv \
${PMM_DOCKER_IMAGE}
```

5. Run the second passive PMM managed server. Like the first passive server, this server will also act as a standby server.

Percona LLC and/or its affiliates ©2024

You can either run all the services on the same instance or a separate instance.

Run services on same instance **Run services on a seperate instance**

```
docker run -d \
--name ${PMM_PASSIVE2_NODE_ID} \
--hostname ${PMM_PASSIVE2_NODE_ID} \
--network pmm-network \
--ip ${PMM_PASSIVE2_IP} \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_CLICKHOUSE=1 \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_POSTGRES=1 \
-e PERCONA_TEST_PMM_CLICKHOUSE_ADDR=${CH_HOST_IP}:9000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_DATABASE=pmm \
-e PERCONA_TEST_PMM_CLICKHOUSE_BLOCK_SIZE=10000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_POOL_SIZE=2 \
-e PERCONA_TEST_POSTGRES_ADDR=${PG_HOST_IP}:5432 \
-e PERCONA_TEST_POSTGRES_USERNAME=${PG_USERNAME} \
-e PERCONA_TEST_POSTGRES_DBPASSWORD=${PG_PASSWORD} \
-e GF_DATABASE_URL=postgres://${GF_USERNAME}:${GF_PASSWORD}@${PG_HOST_IP}:5432/grafana \
-e PMM_VM_URL=http://${VM_HOST_IP}:8428 \
-e PMM_TEST_HA_ENABLE=1 \
-e PMM_TEST_HA_BOOTSTRAP=0 \
-e PMM_TEST_HA_NODE_ID=${PMM_PASSIVE2_NODE_ID} \
-e PMM_TEST_HA_ADVERTISE_ADDRESS=${PMM_PASSIVE2_IP} \
-e PMM_TEST_HA_GOSSIP_PORT=9096 \
-e PMM_TEST_HA_RAFT_PORT=9097 \
-e PMM_TEST_HA_GRAFANA_GOSSIP_PORT=9094 \
-e PMM_TEST_HA_PEERS=${PMM_ACTIVE_IP},${PMM_PASSIVE_IP},${PMM_PASSIVE2_IP} \
-v pmm-server-passive-2_data:/srv \
${PMM_DOCKER_IMAGE}

docker run -d \
--name ${PMM_PASSIVE2_NODE_ID} \
-p 80:80 \
-p 443:443 \
-p 9094:9094 \
-p 9096:9096 \
-p 9094:9094/udp \
-p 9096:9096/udp \
-p 9097:9097 \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_CLICKHOUSE=1 \
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_POSTGRES=1 \
-e PERCONA_TEST_PMM_CLICKHOUSE_ADDR=${CH_HOST_IP}:9000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_DATABASE=pmm \
-e PERCONA_TEST_PMM_CLICKHOUSE_BLOCK_SIZE=10000 \
-e PERCONA_TEST_PMM_CLICKHOUSE_POOL_SIZE=2 \
-e PERCONA_TEST_POSTGRES_ADDR=${PG_HOST_IP}:5432 \
-e PERCONA_TEST_POSTGRES_USERNAME=${PG_USERNAME} \
-e PERCONA_TEST_POSTGRES_DBPASSWORD=${PG_PASSWORD} \
-e GF_DATABASE_URL=postgres://${GF_USERNAME}:${GF_PASSWORD}@${PG_HOST_IP}:5432/grafana \
-e PMM_VM_URL=http://${VM_HOST_IP}:8428 \
-e PMM_TEST_HA_ENABLE=1 \
-e PMM_TEST_HA_BOOTSTRAP=0 \
-e PMM_TEST_HA_NODE_ID=${PMM_PASSIVE2_NODE_ID} \
-e PMM_TEST_HA_ADVERTISE_ADDRESS=${PMM_PASSIVE2_IP} \
-e PMM_TEST_HA_GOSSIP_PORT=9096 \
-e PMM_TEST_HA_RAFT_PORT=9097 \
-e PMM_TEST_HA_GRAFANA_GOSSIP_PORT=9094 \
-e PMM_TEST_HA_PEERS=${PMM_ACTIVE_IP},${PMM_PASSIVE_IP},${PMM_PASSIVE2_IP} \
-v /srv/pmm-data:/srv \
${PMM_DOCKER_IMAGE}
```

Note

- Ensure to set the environment variables from Step 1 in each instance where you run these commands.
- If you run the service on the same instance, remove the `-p` flags.
- If you run the service on a separate instance, remove the `--network` and `--ip` flags.

STEP 7: RUNNING HAPROXY

HAProxy provides high availability for your PMM setup by directing traffic to the current leader server via the `/v1/leaderHealthCheck` endpoint.

Percona LLC and/or its affiliates ©2024

1. Pull the HAProxy Docker image.

```
docker pull haproxy:2.4.2-alpine
```

2. Create a directory to store the SSL certificate.

```
mkdir -p /path/to/certs
```

   Replace `/path/to/certs` with the path where you want to store your SSL certificates.

3. Navigate to this directory and generate a new private key.

```
openssl genrsa -out pmm.key 2048
```

   This command generates a 2048-bit RSA private key and saves it to a file named `pmm.key`.

4. Using the private key, generate a self-signed certificate.

```
openssl req -new -x509 -key pmm.key -out pmm.crt -days 365
```

   Enter country, state, organization name, etc. when asked. Use `-days 365` option for 365-day certificate validity.

5. Copy your SSL certificate and private key to the directory you created in step 2. Ensure that the certificate file is named `pmm.crt` and the private key file is named `pmm.key`.

   Concatenate these two files to create a PEM file:

```
cat pmm.crt pmm.key > pmm.pem
```

6. Create a directory to store HA Proxy configuration.

```
mkdir -p /path/to/haproxy-config
```

   Replace `/path/to/haproxy-config` with the path where you want to store your HAProxy configuration.

7. Create an HAProxy configuration file named `haproxy.cfg.template` in that directory. This configuration tells HAProxy to use the `/v1/leaderHealthCheck` endpoint of each PMM server to identify the leader.

```
global
    log stdout    local0 debug
    log stdout    local1 info
    log stdout    local2 info
    daemon

defaults
    log     global
    mode    http
    option  httplog
    option  dontlognull
    timeout connect 5000
    timeout client  50000
    timeout server  50000

frontend http_front
    bind *:80
    default_backend http_back

frontend https_front
    bind *:443 ssl crt /etc/haproxy/certs/pmm.pem
    default_backend https_back

backend http_back
    option httpchk
    http-check send meth POST uri /v1/leaderHealthCheck ver HTTP/1.1 hdr Host www
    http-check expect status 200
    server pmm-server-active-http PMM_ACTIVE_IP:80 check
    server pmm-server-passive-http PMM_PASSIVE_IP:80 check backup
    server pmm-server-passive-2-http PMM_PASSIVE2_IP:80 check backup

backend https_back
    option httpchk
```


Percona LLC and/or its affiliates ©2024

```
    http-check send meth POST uri /v1/leaderHealthCheck ver HTTP/1.1 hdr Host www
    http-check expect status 200
    server pmm-server-active-https PMM_ACTIVE_IP:443 check ssl verify none
    server pmm-server-passive-https PMM_PASSIVE_IP:443 check ssl verify none
    server pmm-server-passive-2-https PMM_PASSIVE2_IP:443 check ssl verify none
```

8. Before starting the HAProxy container, use `sed` to replace the placeholders in `haproxy.cfg.template` with the environment variables, and write the output to `haproxy.cfg`.

```
sed -e "s/PMM_ACTIVE_IP/$PMM_ACTIVE_IP/g" \
    -e "s/PMM_PASSIVE_IP/$PMM_PASSIVE_IP/g" \
    -e "s/PMM_PASSIVE2_IP/$PMM_PASSIVE2_IP/g" \
    /path/to/haproxy.cfg.template > /path/to/haproxy.cfg
```

9. Run the HAProxy container.

```
docker run -d \
  --name haproxy \
  --network pmm-network \
  -p 80:80 \
  -p 443:443 \
  -v /path/to/haproxy-config:/usr/local/etc/haproxy \
  -v /path/to/certs:/etc/haproxy/certs \
  haproxy:2.4.2-alpine
```

   Replace `/path/to/haproxy-config` with the path to the `haproxy.cfg` file you created in step 6, and `/path/to/certs` with the path to the directory containing the SSL certificate and private key.

Note

- It is recommended to use absolute paths instead of relative paths for volume mounts.
- If you're running services on separate instances, you can remove the `--network` flag.

HAProxy is now configured to redirect traffic to the leader PMM managed server. This ensures highly reliable service by redirecting requests to the remainder of the servers in the event that the leader server goes down.

STEP 8: ACCESS PMM

You can access the PMM web interface via HAProxy once all the components are set up and configured:

1. Access the PMM services by navigating to `https://<HAProxy_IP>` in your web browser. Replace `<HAProxy_IP>` with the IP address or hostname of the machine running the HAProxy container.
2. You should now see the PMM login screen. Log in using the default credentials, unless you changed them during setup.
3. You can use the PMM web interface to monitor your database infrastructure, analyze metrics, and perform various database management tasks.

When you register PMM Clients, you must use the HAProxy IP address (or hostname) rather than the PMM Server address once your PMM environment has been set up in high-availability (HA) mode. Even if one PMM server becomes unavailable, clients will still be able to communicate with the servers.

You have now successfully set up PMM in HA mode using Docker containers. Your PMM environment is more resilient to failures and can continue providing monitoring services if any of the instances fail.

Note

Ensure that all containers are running and accessible. You can use `docker ps` to check the status of your Docker containers. If a container is not running, you can view its logs using the command `docker logs <container_name>` to investigate the issue.

Percona LLC and/or its affiliates ©2024

5.7.2 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

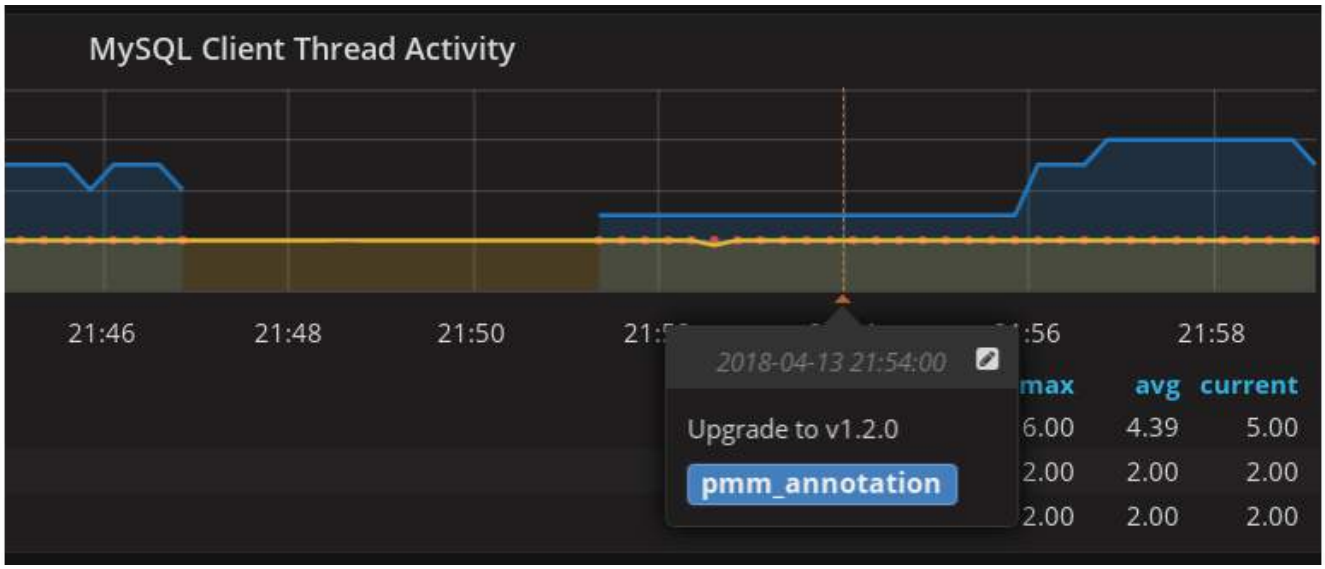
2024-06-11

5.8 Annotate

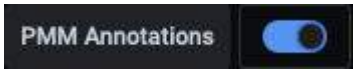
Annotations mark a moment in time. They are useful for marking system changes or other significant application events. They can be set globally or for specific nodes or services.

You create them on the command line with the `pmm-admin annotate` command.

Annotations show as a vertical dashed line on a dashboard graph. Reveal the annotation text by mousing over the caret indicator below the line.



You turn annotations on or off with the *PMM Annotations* switch in the second row menu bar.



5.8.1 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

5.9 Share dashboards and panels

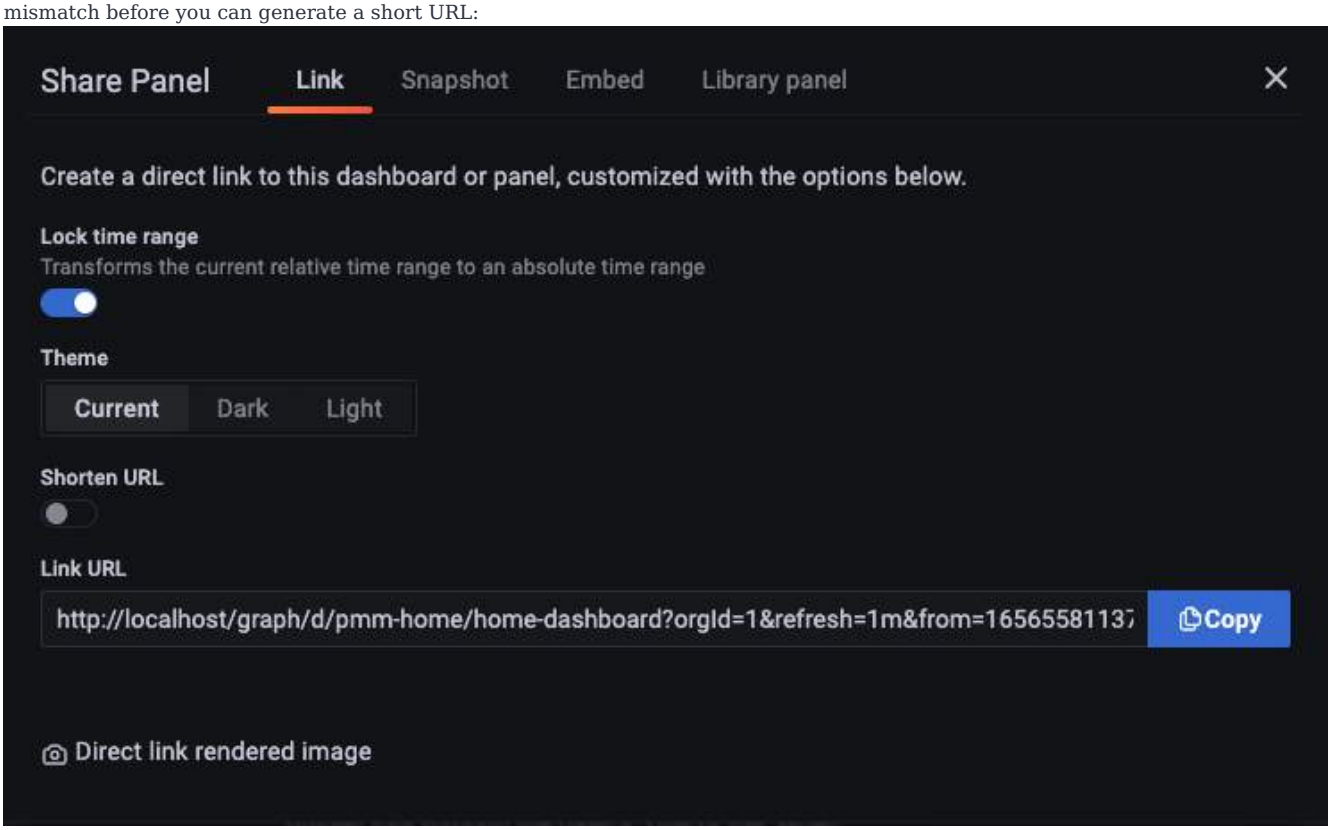
When you need to share a dashboard with your team members, you can either send them a direct link to the dashboard, or render and send the dashboard as a .PNG image.

5.9.1 Share as direct link

1. Go to the dashboard that you want to share.
2. Click at the top of the dashboard to display the panel menu.
3. Select **Share** to reveal the **Share Panel** and either:
4. copy and send the full URL for the dashboard, OR
5. toggle the **Short URL** option to generate a simple link with a unique identifier

Tip

If your current domain is different than the one specified in the Grafana .INI configuration file, PMM will ask you to correct this mismatch before you can generate a short URL:

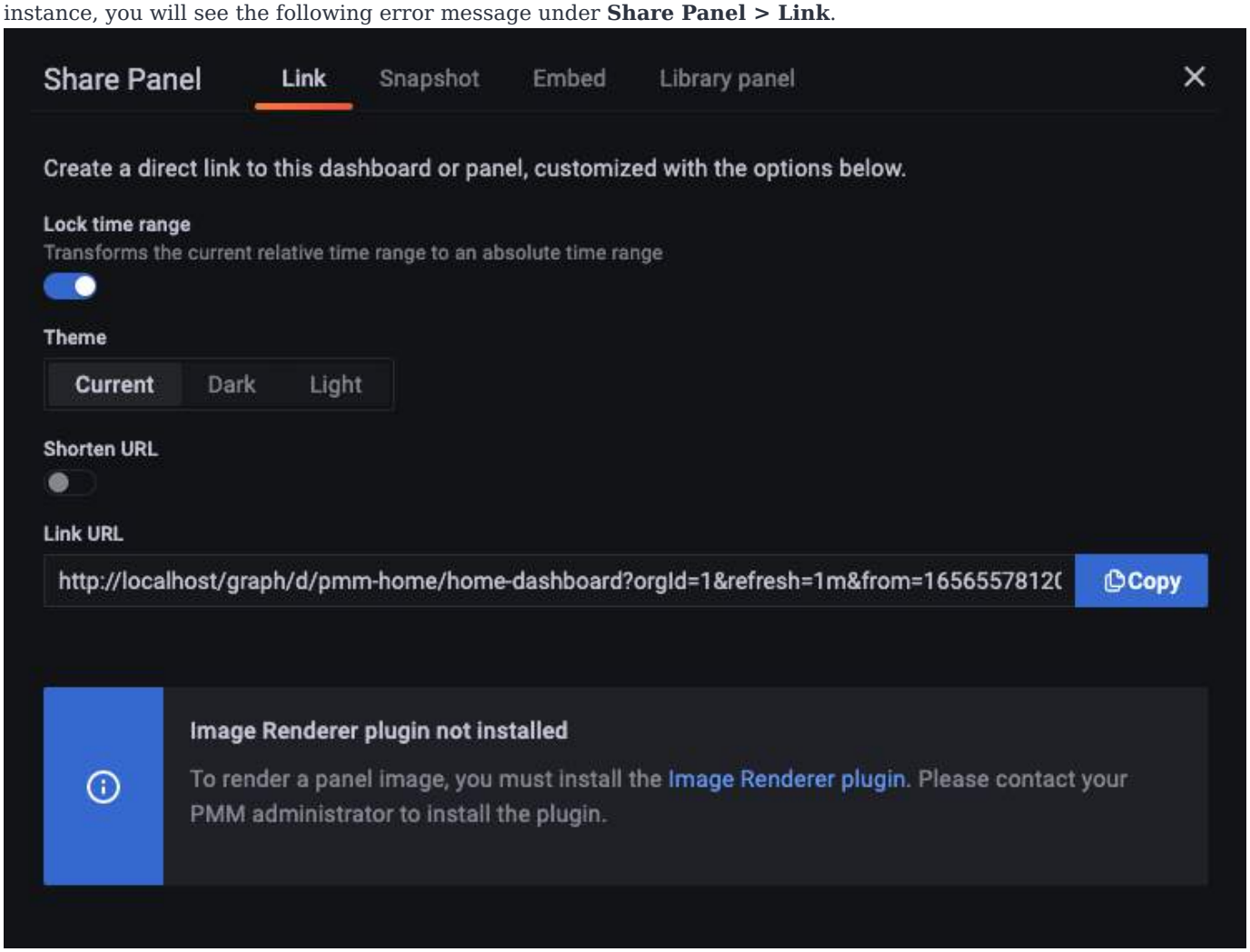


To fix this

Percona LLC and/or its affiliates ©2024

5.9.2 Share as a PNG file

Rendering images requires the Image Renderer plug-in. If your PMM Admin has not installed this for your PMM instance, you will see the following error message under **Share Panel > Link**.



To install the dependencies:

1. Connect to your PMM Server Docker container.

```
docker exec -it pmm-server bash
```

2. Install Grafana plug-ins.

```
grafana-cli plugins install grafana-image-renderer
```

3. Restart Grafana.

```
supervisorctl restart grafana
```

4. Install libraries.

```
yum install -y libXcomposite libXdamage libXtst cups libXScrnSaver pango \
atk adwaita-cursor-theme adwaita-icon-theme at at-spi2-atk at-spi2-core \
cairo-gobject colord-libs dconf desktop-file-utils ed emacs-filesystem \
gdk-pixbuf2 glib-networking gnutls gsettings-desktop-schemas \
gtk-update-icon-cache gtk3 hicolor-icon-theme jasper-libs json-glib \
libappindicator-gtk3 libdbusmenu libdbusmenu-gtk3 libepoxy \
liberation-fonts liberation-narrow-fonts liberation-sans-fonts \
liberation-serif-fonts libgusb libindicator-gtk3 libmodman libproxy \
```


Percona LLC and/or its affiliates ©2024

```
libsoup libwayland-cursor libwayland-egl libxkbcommon m4 mailx nettle \
patch psmisc redhat-lsb-core redhat-lsb-submod-security rest spax time \
trousers xdg-utils xkeyboard-config alsa-lib
```

To render the image:

1. Go to the dashboard that you want to share.
2. Click at the top of the dashboard to display the panel menu.
3. Select **Share** to reveal the **Share Panel**.
4. Click **Direct link rendered image**. This opens a new browser tab.
5. Wait for the image to be rendered, then use your browser's Image Save function to download the image.

5.9.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-07-19

5.10 Extend Metrics

When you need a metric that's not present in the default list of `node_exporter` metrics you may be able to use the `textfile` collector. The textfile collector allows exporting of statistics from batch jobs. It can also be used to export static metrics, such as what role a machine has.

5.10.1 Enable the textfile collector

The collector is enabled by default. The following folders are used for different resolutions:

Resolution	Folder
High	`/usr/local/percona/pmm2/collectors/textfile-collector/high-resolution`
Medium	`/usr/local/percona/pmm2/collectors/textfile-collector/medium-resolution`
Low	`/usr/local/percona/pmm2/collectors/textfile-collector/low-resolution`

The exporter parses all files in these directories that match the filename wildcard expression `*.prom` using a simple text-based exposition format. Metrics are stored on the PMM Server-side with additional labels related to this Node.

5.10.2 Examples of shell commands for custom metrics

To statically set roles for a machine using labels:

Percona LLC and/or its affiliates ©2024

```
echo 'node_role{role="my_monitored_server_1"} 1' > /usr/local/percona/pmm2/collectors/textfile-collector/low-resolution/node_role.prom
```

Here's an example of a `cron` job that automatically pushes logged-in users:

```
$ cat /etc/cron.d/loggedin_users
*/1 * * * *     root    /usr/bin/who | /usr/bin/wc -l | sed -ne 's/^/node_loggedin_users /p' > /usr/local/percona/pmm2/collectors/textfile-collector/high-resolution/node_users.prom
```

5.10.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-23

5.11 Resolve issues

This section describes solutions to common problems and scenarios you might encounter while using PMM.

5.11.1 Troubleshooting checklist

The following questions might help you identify the origin of the problem while using Percona Monitoring and Management:

1. Are you using the latest PMM version?
2. Did you check the known issues section in the Release Notes for that particular PMM release?
3. Are you receiving any error messages?
4. Do the logs contain any messages about the problem? See Message logs and Trace logs for more information.

Percona LLC and/or its affiliates ©2024

5. Does the problem occur while configuring PMM, such as:
    - Does the problem occur while you configure a specific function?
    - Does the problem occur when you perform a particular task?
6. Are you using the recommended authentication method?
7. Does your system's firewall allow TCP traffic on the ports used by PMM?
8. Have you allocated enough disk space for installing PMM? If not, check the disk allocation space.
9. Are you using a Technical Preview feature? Technical Preview features are not production-ready and should only be used in testing environments. For more information, see the relevant Release Notes.
10. For installing the PMM client, are you using a package other than a binary package without root permissions?
11. Is your PMM Server installed and running with a known IP address accessible from the client node?
12. Is the PMM Client installed, and is the node registered with PMM Server?
13. Is PMM-client configured correctly and has access to the config file?
14. For monitoring MongoDB, do you have adminUserAnyDatabase or superuser role privilege to any database servers you want to monitor?
15. For monitoring Amazon RDS using PMM, is there too much latency between PMM Server and the Amazon RDS instance?
16. Have you upgraded the PMM Server before you upgraded the PMM Client? If yes, there might be configuration issues, thus leading to failure in the client-server communication, as PMM Server might not be able to identify all the parameters in the configuration.
17. Is the PMM Server version higher than or equal to the PMM Client version? Otherwise, there might be configuration issues, thus leading to failure in the client-server communication, as PMM Server might not be able to identify all the parameters in the configuration.

5.11.2 Troubleshooting areas

Upgrade issues

PMM SERVER NOT UPDATING CORRECTLY

If the PMM server wasn't updated correctly, or if you have concerns about the release, you can force the update process in 2 ways:

1. From the UI - Home panel: click the Alt key on the reload icon in the Update panel to make the Update Button visible even if you are on the same version as available for update. Pressing this button will force the system to rerun the update so that any broken or not installed components can be installed. In this case, you'll go through the usual update process with update logs and successful messages at the end.
2. By API call (if UI not available): You can call the Update API directly with:

    ```
    curl --user admin:admin --request POST 'http://PMM_SERVER/v1/Updates/Start'
    ```

    Replace `admin:admin` with your username/password, and replace `PMM_SERVER` with your server address.

    > You will not see the logs using this method.

    Refresh The Home page in 2-5 minutes, and you should see that PMM was updated.
3. Upgrade PMM server using Docker.

Percona LLC and/or its affiliates ©2024

PMM SERVER NOT SHOWING LATEST VERSIONS AVAILABLE WITH THE INSTANCES CREATED FROM AWS

For PMM versions prior to 2.33.0, in specific environments, including AWS, some EPEL repository mirrors did not respond within the time limit defined by `pmm-update` (currently set to 30 seconds). It was causing supervisord to kill pmm-update-checker, which determines if a newer PMM Server is available for upgrade.

Solution

Log in to the PMM Server and run the following command as a root user:

```
$ yum-config-manager --setopt=epel.timeout=1 --save
```

PMM SERVER FAILS WHILE UPGRADING

A bug in PMM Server ansible scripts caused PMM to upgrade Nginx's dependencies without updating Nginx itself. Due to this, PMM throws an error while upgrading and cannot upgrade to a newer version.

> **Important**
>
> This issue has been resolved for PMM version 2.33.0. However, the issue persists on all the versions prior to 2.33.0.

Solution

While PMM is being upgraded, log in to the PMM server and run the following command:

```
sed -i 's/- nginx/- nginx*/' /usr/share/pmm-update/ansible/playbook/tasks/update.yml
```

PMM CANNOT ACESS ADMIN USER AFTER UPGRADING

After upgrading PMM from version 2.39.0 to 2.40.0 (not el7) using Docker, the `admin` user cannot access the PMM UI.

Solution: To fix the problem and gain back admin access to the PMM interface execute the following:

```
# psql -U grafana
grafana=> update "user" set id='1' where login='admin';
UPDATE 1
grafana=> \q

# grafana cli --homepath=/usr/share/grafana --config=/etc/grafana/grafana.ini admin reset-admin-password <PASS>
```

Configuration issues

This section focuses on configuration issues, such as PMM-agent connection, adding and removing services for monitoring, and so on.

CLIENT-SERVER CONNECTIONS

There are many causes of broken network connectivity.

The container is constrained by the host-level routing and firewall rules when using using Docker. For example, your hosting provider might have default `iptables` rules on their hosts that block communication between PMM Server and PMM Client, resulting in *DOWN* targets in VictoriaMetrics. If this happens, check the firewall and routing settings on the Docker host.

PMM can also generate diagnostics data that can be examined and/or shared with our support team to help solve an issue. You can get collected logs from PMM Client using the pmm-admin summary command.

Logs obtained in this way include PMM Client logs and logs received from the PMM Server, and stored separately in the `client` and `server` folders. The `server` folder also contains its `client` subfolder with the self-monitoring client information collected on the PMM Server.

Percona LLC and/or its affiliates ©2024

Beginning with PMM 2.4.0, there is a flag that enables the fetching of `pprof` debug profiles and adds them to the diagnostics data. To enable, run `pmm-admin summary --pprof`.

You can get PMM Server logs with either of these methods:

Direct download

In a browser, visit `https://<address-of-your-pmm-server>/logs.zip`.

From Help menu

1. Select *Help* → *PMM Logs*.
2. Click *PMM Logs* to retrieve PMM diagnostics data which can be examined and shared with our support team should you need help.

CONNECTION DIFFICULTIES

Passwords

When adding a service, the host might not be detected if the password contains special symbols (e.g., `@`, `%`, etc.).

In such cases, you should convert any password, replacing special characters with their escape sequence equivalents.

One way to do this is to use the `encodeURIComponent` JavaScript function in your browser's web console (commonly found under a *Development Tools* menu). Run the function with your password as the parameter. For example:

```
> encodeURIComponent("s3cR#tpa$$worD")
```

will give:

```
"s3cR%23tpa%24%24worD"
```

Password change

When adding clients to the PMM server, you use the `admin` user. However, if you change the password for the admin user from the PMM UI, then the clients will not be able to access PMM due to authentication issues. Also, Grafana will lock out the admin user due to multiple unsuccessful login attempts.

In such a scenario, use API key for authentication. You can use API keys as a replacement for basic authentication.

5.11.3 Percona Alerting

No Alert rule templates tab on the Alerting page

Percona Alerting option isn't active.

1. Go to *Configuration* → *Settings* → *Advanced Settings*.
2. Enable *Alerting*.

Custom alert rule templates not migrated to Percona Alerting

If you have used Integrated Alerting in previous PMM versions, and had custom templates under `/srv/ia/templates`, make sure to transfer them to `/srv/alerting/templates`. PMM is no longer sourcing templates from the `ia` folder, since we have deprecated Integrated Alerting with the 2.31 release.

Percona LLC and/or its affiliates ©2024

UNREACHABLE EXTERNAL IP ADDRESSES

If you get an email or page from your system that the IP is not reachable from outside my organization, do the following:

To configure your PMM Server's Public Address, select *Configuration* → *Settings* → *Advanced Settings*, and supply an address to use in your alert notifications.

ALERT RULE TEMPLATES ARE DISABLED

Built-In alerts are not editable, but you can copy them and edit the copies. (In PMM 2.14.0 and above).

If you create a custom alert rule template, you will have access to edit.

QAN issues

This section focuses on problems with QAN, such as queries not being retrieved so on.

MISSING DATA

Why don't I see any query-related information?

There might be multiple places where the problem might come from:

- Connection problem between pmm-agent and pmm-managed
- PMM-agent cannot connect to the database.
- Data source is not properly configured.

Why don't I see the whole query?

Long query examples and fingerprints can be truncated to 1024 symbols to reduce space usage. In this case, the query explains section will not work.

Plugins issues

PMM does not allow to install, upgrade or remove plugins

Users have encountered issues with installing, updating and removing plugins from PMM. The cause of this issue is the incorrect permissions assigned to the `/srv/grafana/plugins` directory. These permissions are preventing the grafana component from writing to the directory.

Solution

Set the ownership on the directory `/srv/grafana/plugins` to `grafana:grafana`.

5.11.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

Percona LLC and/or its affiliates ©2024

5.12 Export PMM data with PMM Dump

PMM data dumps are compressed tarball files containing a comprehensive export of your PMM metrics and QAN data collected by PMM Server.

You can download these dataset files locally, or share them with Percona Support via an SFTP server. This enables you to share PMM data securely, which is especially useful when you need you troubleshoot PMM issues without without providing access to your PMM instance.

Starting with 2.41, PMM enables you to generate PMM Datasets straight from PMM. If you are using an older PMM version, you can use the standalone PMM Dump utility instead.

5.12.1 Dump contents

The **dump.tar.gz** dump file is a .TAR archive compressed via Gzip. Here's what's inside the folders it contains:

- **meta.json**: metadata about the data dump
- **vm**: Victoria Metrics data chunks in native VM format, organized by timeframe
- **ch**: Query Analytics (QAN) data stored in ClickHouse, organized by rows count
- **log.json**: logs detailing the export and archive creation process

5.12.2 Create a data dump

To create a dump of your dataset:

1. From the main menu on the left, go to **Help > PMM Dump**.
2. Click **Create dataset** to go to the **Export new dataset** page.
3. Choose the service for which you want to create the dataset or leave it empty to export all data.
4. Define the time range for the dataset.
5. Enable **Export QAN** to include Query Analytics (QAN) metrics alongside the core metrics.
6. Enable **Ignore load** to export the dump bypassing the default resource limit restrictions.
7. Click **Create dataset**. This will generate a data dump file and automatically record an entry in the PMM Dump table. From there, you can use the options available in the **Options** menu to send the dump file to Percona Support or download it locally for internal usage.

5.12.3 Send a data dump to Percona Support

If you are a Percona Customer, you can securely share PMM data dumps with Percona Support via SFTP.

1. From the main menu on the left, go to **Help > PMM Dump**.
2. Select the PMM dump entry which you want to send to Support.
3. In the **Options** column, expand the table row to check the PMM Service associated with the dataset, click the ellipsis (three vertical dots) and select **Send to Support**.
4. Fill in the details of the SFTP server, then click **Send**.
5. Update your Support ticket to let Percona know that you've uploaded the dataset on the SFTP server.

5.12.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2023-12-12

5.13 Integrate with Percona Platform

5.13.1 Integrate PMM with Percona Platform

Percona Platform brings together database distributions, support expertise, services, management, and automated insights.

Connect your PMM servers to Percona Platform to boost the monitoring capabilities of your PMM installations and manage database deployments easier. In addition, you get access to PMM updates, automated insights, advanced advisor checks and more alert rule templates.

CONNECT PMM TO PERCONA PLATFORM

You can connect to Percona Platform with a Percona Account or via Google or GitHub authentication. If Percona Support has enabled a custom identity provider for your account, you can also log in using your company's credentials.

We recommend that you connect with a Percona Account, as this gives you access to other Percona services, including Percona Platform, Percona Customer Portal, and Community Forum. If you don't have a Percona Account, you can create one on the Percona Platform homepage using the **Don't have an account? Create one?** link.

Prerequisites

To ensure that PMM can establish a connection to Percona Platform:

UPGRADE TO PMM 2.27.0 OR LATER

Before connecting your PMM server to Percona Platform, make sure you are using PMM version 2.27 or newer. Otherwise, upgrade your PMM installation beforehand.

This is required because, starting with PMM 2.27, Percona Platform has replaced username/password authentication with access token authentication. Access-token authentication increases security and enables federated identity.

This change did not affect existing connections to PMM Platform, which were not automatically terminated.

For more information, see Install and set up PMM.

CHECK THAT YOU ARE A MEMBER OF AN EXISTING PLATFORM ORGANIZATION

1. Log in to Percona Platform using your Percona Account. If you are connecting via GitHub, make sure you set your email address as **public** in your GitHub account. If your email address is private instead, Percona Platform cannot access it to authenticate you.
2. On the **Getting Started** page, check that the **Create organization** step shows an option to view your organization.

Contact your account administrator or create a new organization for your Percona Account if this is the case.

SET THE PUBLIC ADDRESS OF YOUR PMM SERVER

PMM automatically detects and populates the public address of the PMM server when this is not set up. If you need to set it differently, go to **Settings > Advanced Settings** and edit the **Public Address** field.

Percona LLC and/or its affiliates ©2024

Connect PMM to Percona Platform

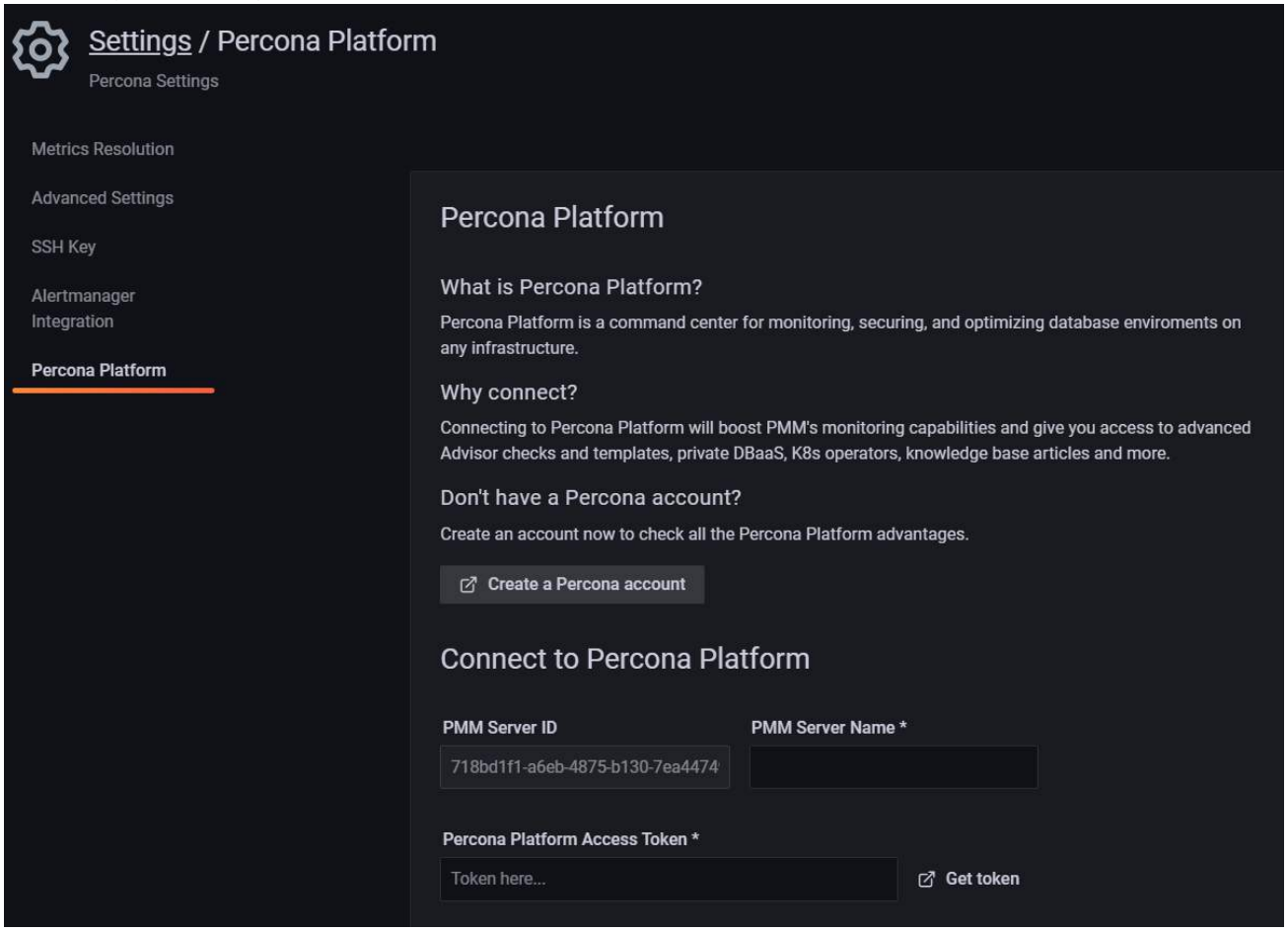
To connect your PMM server to Percona Platform, copy your personal access token from Platform Portal and paste it into PMM. You will find your access token in Platform Portal as part of your user profile page.

Token validity

For security reasons, access tokens expire after 30 minutes. Make sure to paste the code before that, or generate a new one if it expires.

To connect your PMM server to Percona Platform:

1. In PMM, go to **Settings > Percona Platform** tab to fill in the **Connect PMM to Percona Portal** form:



2. The **PMM server ID** field is automatically populated with the ID identified for your PMM instance. Enter the name of your PMM instance and click **Get token** to go to Percona Platform Portal and generate your access token.
3. Log into Percona Platform using your Percona Account (if you don’t have an active current session).
4. On the **Profile Settings page**, copy the code from the **Percona Platform Access Token** field.
5. Back into PMM, paste the Access Token into the **Percona Platform Access Token** field, and click **Connect**.

To confirm that you have successfully connected the server and check the list of all servers currently connected to an organization, go to Percona Platform > **Dashboard** tab and click **View Instances** next to the **Connect your PMM** step.

Percona LLC and/or its affiliates ©2024

Check Percona Portal entitlements

After connecting to the Percona Platform, PMM has access to additional alert templates, Advisor checks, and account information. See Check Percona Portal account information.

DISCONNECT A PMM INSTANCE

Disconnect a PMM instance when you want to unlink it from your Percona Platform organization or stop monitoring it there.

To disconnect a PMM server, go to **Configuration > Settings > Percona Platform** and click **Disconnect**.

Disconnecting instances as an Admin

In situations where you are not able to disconnect servers yourself, ask your PMM Admin to disconnect the server for you. For example, you may not be able to disconnect servers when PMM is moved to a network segment without outbound connections to public networks.

> **Availability**
>
> This feature is available starting with PMM 2.29.0.

If you cannot disconnect servers yourself, ask your PMM Admin to disconnect the server for you. For example, you may not be able to disconnect servers when PMM is moved to a network segment without outbound connections to public networks.

If you are a PMM Admin, you can terminate any connections to Percona Platform, even if you are not logged into PMM with a Percona Account. However, we recommend logging in with a Percona Account before disconnecting servers, as this will automatically remove the disconnected servers from Percona Platform as well.

If you do disconnect servers without being connected with a Percona Account, you'll have to manually remove the unavailable servers from Percona Platform. This ensures that your list of connected PMM instances stays up-to-date in Percona Platform.

To do this, go to PMM instances, and remove any servers that you have already disconnected from PMM.

Sign into PMM with your Percona Account

Once you've successfully connected your PMM instance to the Percona Platform, you can also sign into PMM using your Percona Account:

1. Log out of your existing PMM session.
2. On the PMM login screen, click *Sign in with Percona Account*. If you have an active Percona Account session on the same browser, PMM will log you in automatically. Otherwise, enter your Percona Account credentials to start a new session.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

Percona LLC and/or its affiliates ©2024

5.13.2 Check Percona Portal account information

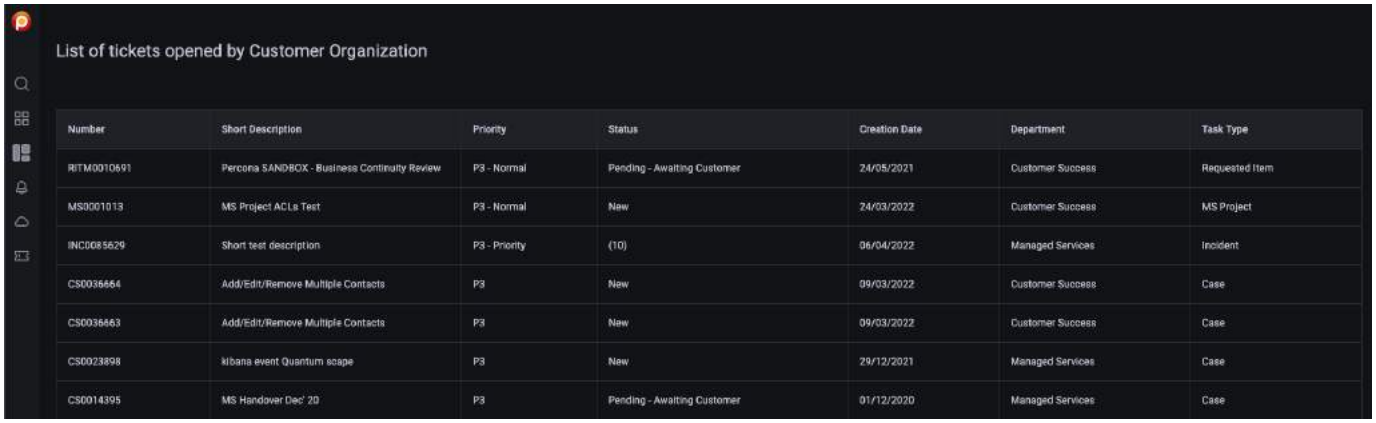
When you connect your PMM instances to Percona Platform, PMM gets access to:

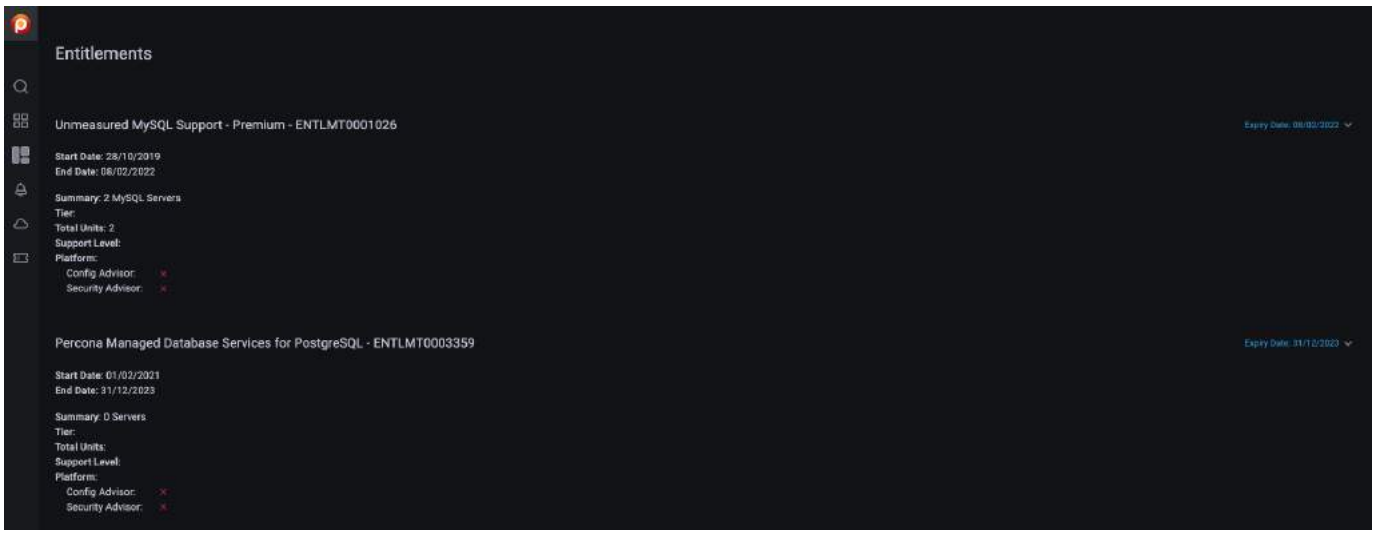
- more alert templates
- Basic Advisor Checks for additional database checks
- Standard/Premium Advisor Checks for more advanced database health checks.

Standard/Premium checks are available when you connect to Percona Platform with a customer account.

You can check the list of available Standard/Premium Advisor checks in the Percona Platform documentation or on the Percona Platform Subscription plans page.

When you connect with a customer account, PMM reveals two new tabs on the main menu, where you can check all the information available for your customer accounts: **Entitlements** and **Support tickets**:





Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2023-03-24

Percona LLC and/or its affiliates ©2024

6. Details

6.1 Details

- Architecture: High-level architecture and main components.
- User interface components: Descriptions of the main menus and icons.
- PMM components and versions: PMM components and their version used in PMM.
- Data handling in PMM: Personal and confidential data handling in PMM.
- QAN under the hood: The Query Analytics dashboard provides insights into query execution and time allocation, helping you analyze database queries over time. It enables you to optimize database performance and identify and resolve issues at their source.
- Developing Advisor checks: Database health assessments.
- Dashboards reference: A complete list of dashboards by category, with screenshots.
- Commands:
- pmm-admin: The manual page for the PMM administration tool.
- pmm-agent: The manual page for the PMM Client agent program.
- API: How to access the Swagger API.
- VictoriaMetrics: The monitoring solution and time-series database that replaced Prometheus in PMM 2.12.0.
- ClickHouse: A third-party column-oriented database management system (DBMS) that facilitates the Query Analytics functionality.
- PostgreSQL: An open source object-relational database management system used as the primary data store.
- Glossary: A list of obscure terms and definitions.

6.1.1 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

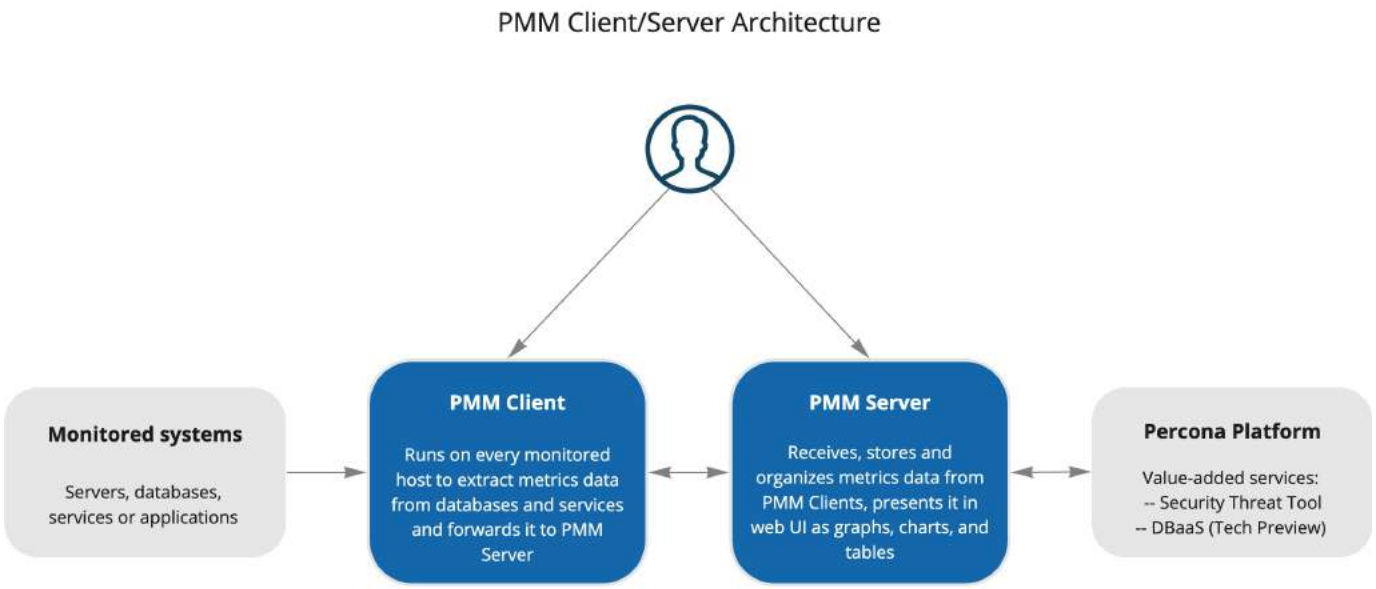
Community Forum Get a Percona Expert

2024-06-03

6.2 Architecture

PMM is a client/server application built by Percona comprising its own and third-party components and tools.

Percona LLC and/or its affiliates ©2024



6.2.1 PMM Server

PMM Server is the heart of PMM. It receives data from clients, collects it, and stores it. Metrics are drawn as tables, charts and graphs within *dashboards*, each a part of the web-based user interface.

6.2.2 PMM Client

PMM Client is a collection of agents and exporters that run on the host being monitored.

PMM Client runs on every database host or node you want to monitor. The client collects server metrics, general system metrics, query analytics and sends it to the server. Except when monitoring AWS RDS instances, a PMM Client must be running on the host to be monitored.

6.2.3 Percona Platform

Percona Platform provides value-added services for PMM.

6.2.4 PMM context

The PMM Client package provides:

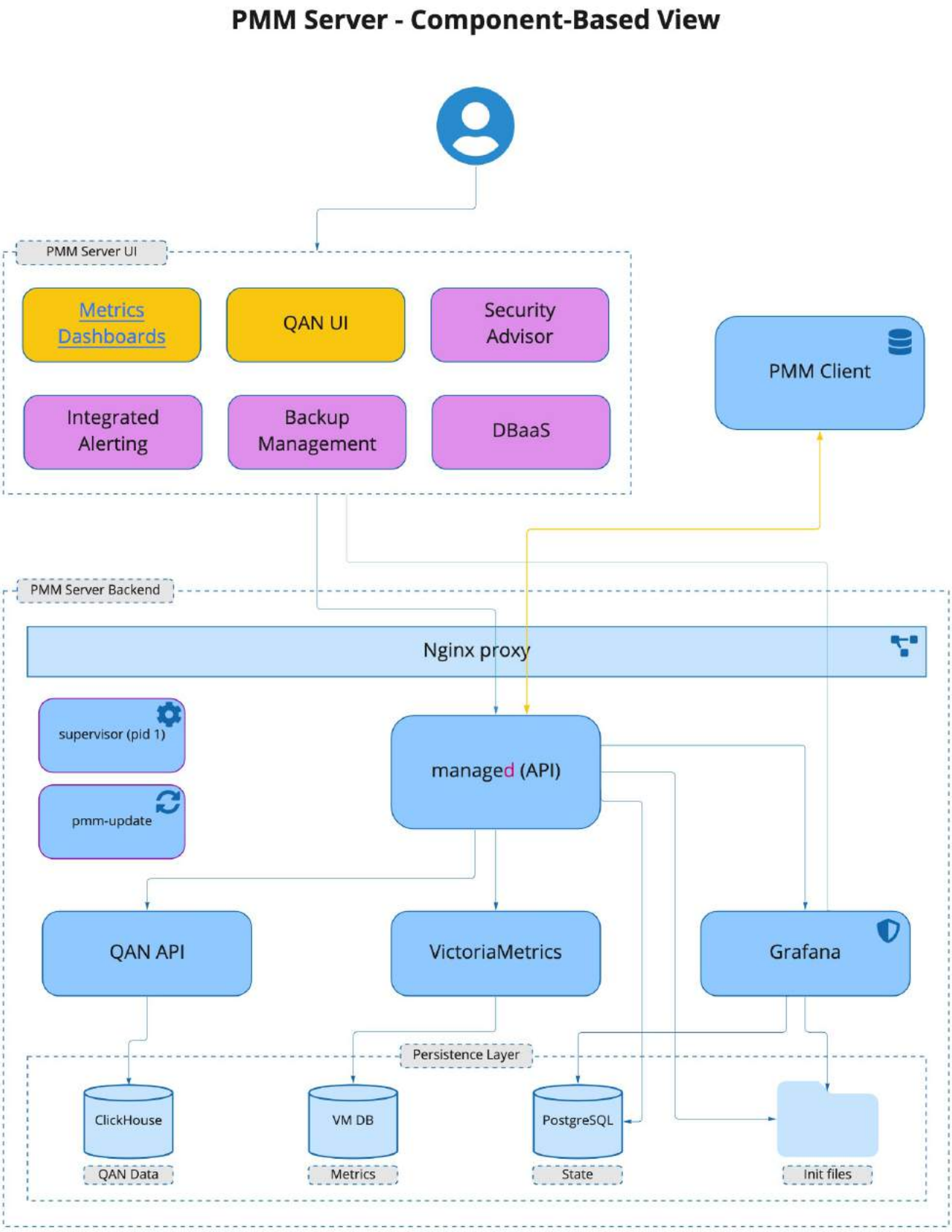
- Exporters for each database and service type. When an exporter runs, it connects to the database or service instance, runs the metrics collection routines, and sends the results to PMM Server.
- `pmm-agent`: Run as a daemon process, it starts and stops exporters when instructed.
- `vmagent`: A VictoriaMetrics daemon process that sends metrics data (*pushes*) to PMM Server.

The PMM Server package provides:

- `pmm-managed`
- Query Analytics
- Grafana
- VictoriaMetrics

Percona LLC and/or its affiliates ©2024

PMM Server



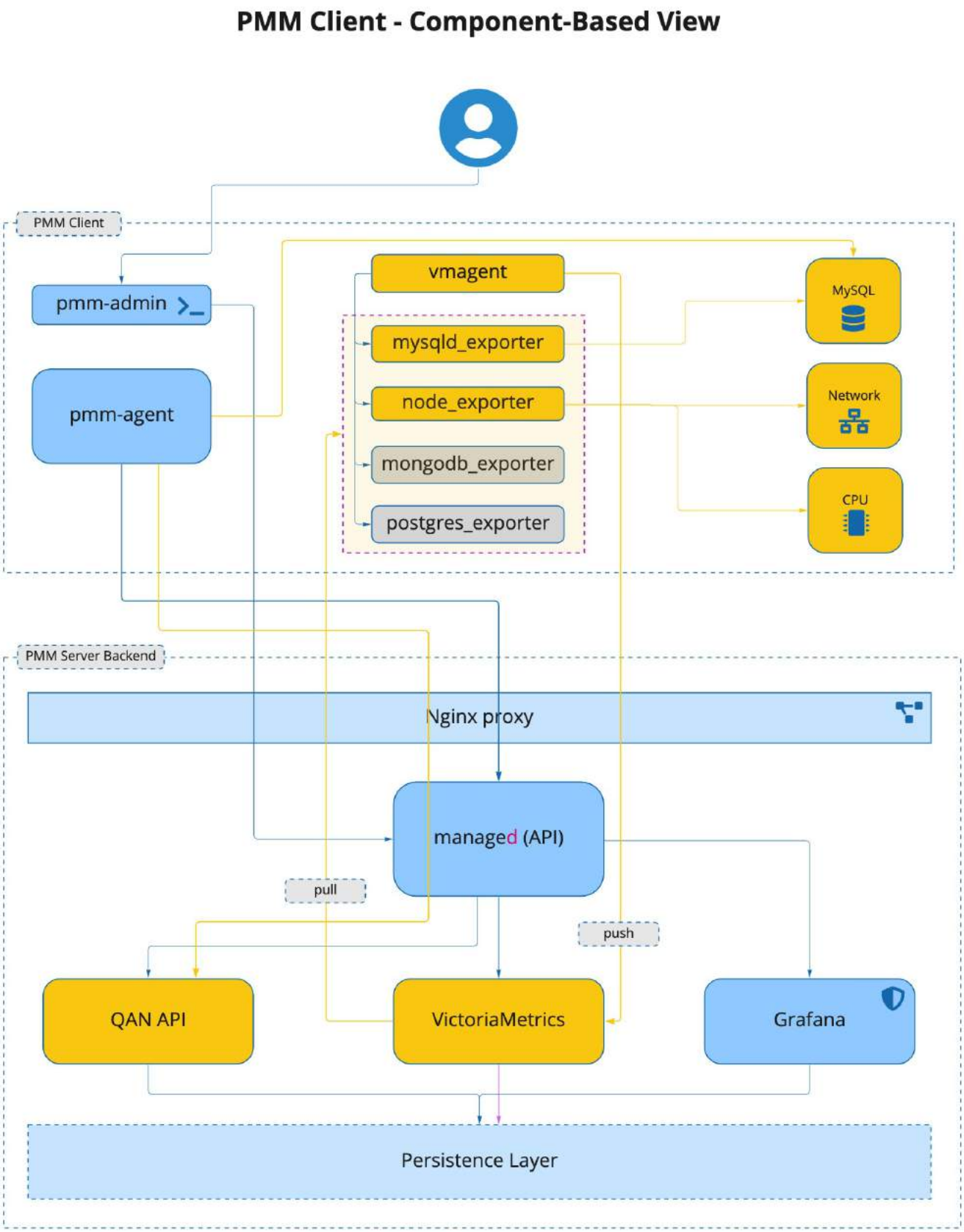
Percona LLC and/or its affiliates ©2024

PMM Server includes the following tools:

- Query Analytics (QAN) enables you to analyze database query performance over periods of time. In addition to the client-side QAN agent, it includes the following:
  - QAN API is the back-end for storing and accessing query data collected by the QAN agent running on a PMM Client.
  - QAN App is a web application for visualizing collected Query Analytics data, which is part of the PMM Server's UI.
- Metrics Monitor provides a historical view of metrics that are critical to a MySQL or MongoDB server instance. It includes the following:
- VictoriaMetrics, a scalable time-series database. (Replaced Prometheus in PMM 2.12.0.)
- ClickHouse is a third-party column-oriented database that facilitates the Query Analytics functionality.
- Grafana is a third-party dashboard and graph builder for visualizing data aggregated (by VictoriaMetrics or Prometheus) in an intuitive web interface.
- Percona Dashboards is a set of dashboards for Grafana developed by Percona.

Percona LLC and/or its affiliates ©2024

PMM Client

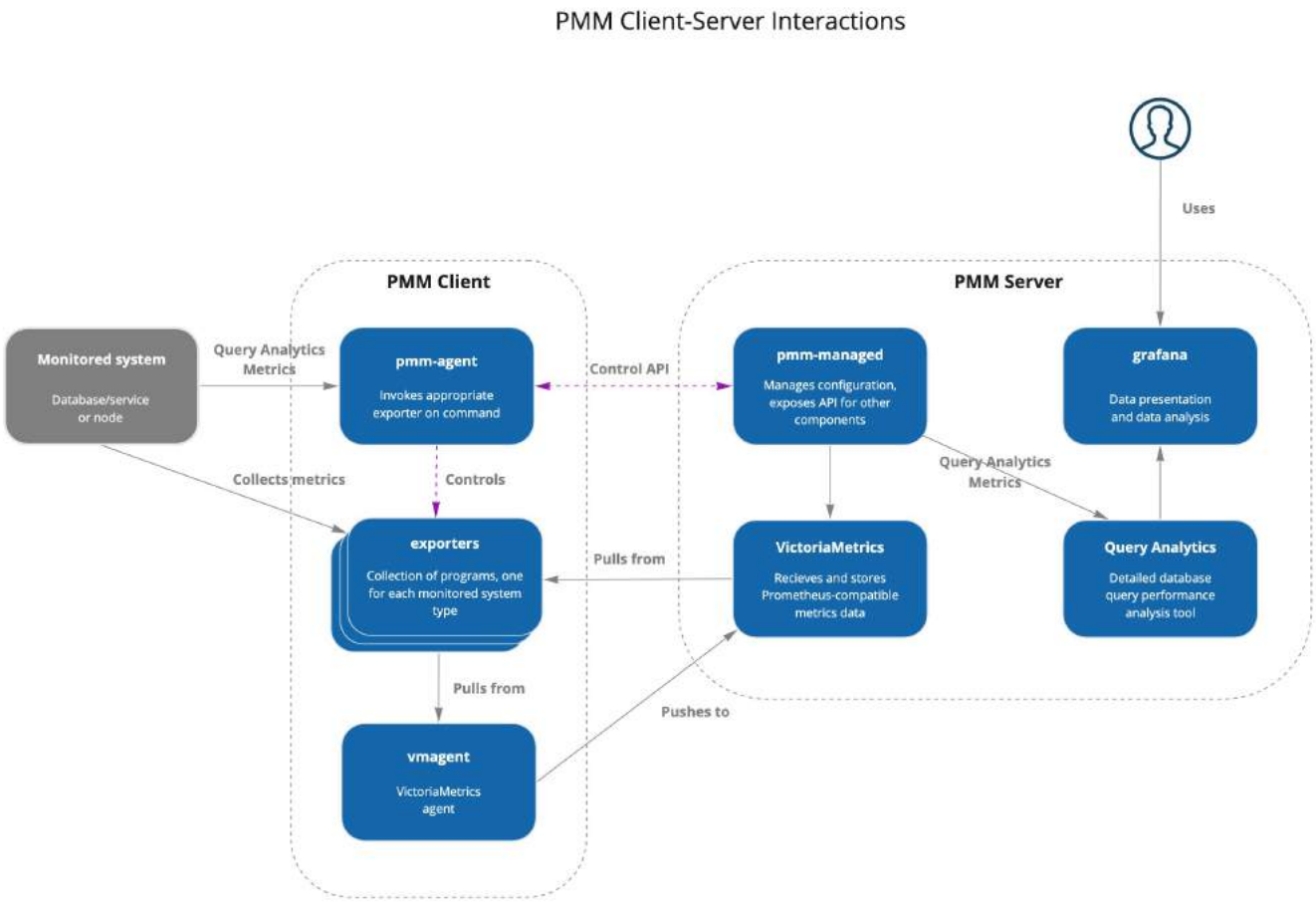


Percona LLC and/or its affiliates ©2024

The PMM Client package consists of the following:

- `pmm-admin` is a command-line tool for managing PMM Client, for example, adding and removing database instances that you want to monitor. (Read more).
- `pmm-agent` is a client-side component of a minimal command-line interface, which is a central entry point in charge of bringing the client functionality: it carries on client's authentication, gets the client configuration stored on the PMM Server, manages exporters and other agents.
- `node_exporter` is an exporter that collects general system metrics.
- `mysqld_exporter` is an exporter that collects MySQL server metrics.
- `mongodb_exporter` is an exporter that collects MongoDB server metrics.
- `postgres_exporter` is an exporter that collects PostgreSQL performance metrics.
- `proxysql_exporter` is an exporter that collects ProxySQL performance metrics.
- `rds_exporter` is an exporter that collects Amazon RDS performance metrics.
- `azure_database_exporter` is an exporter that collects Azure database performance metrics.

To make data transfer from PMM Client to PMM Server secure, all exporters are able to use SSL/TLS encrypted connections, and their communication with PMM server is protected by the HTTP basic authentication.



6.2.5 Get expert help

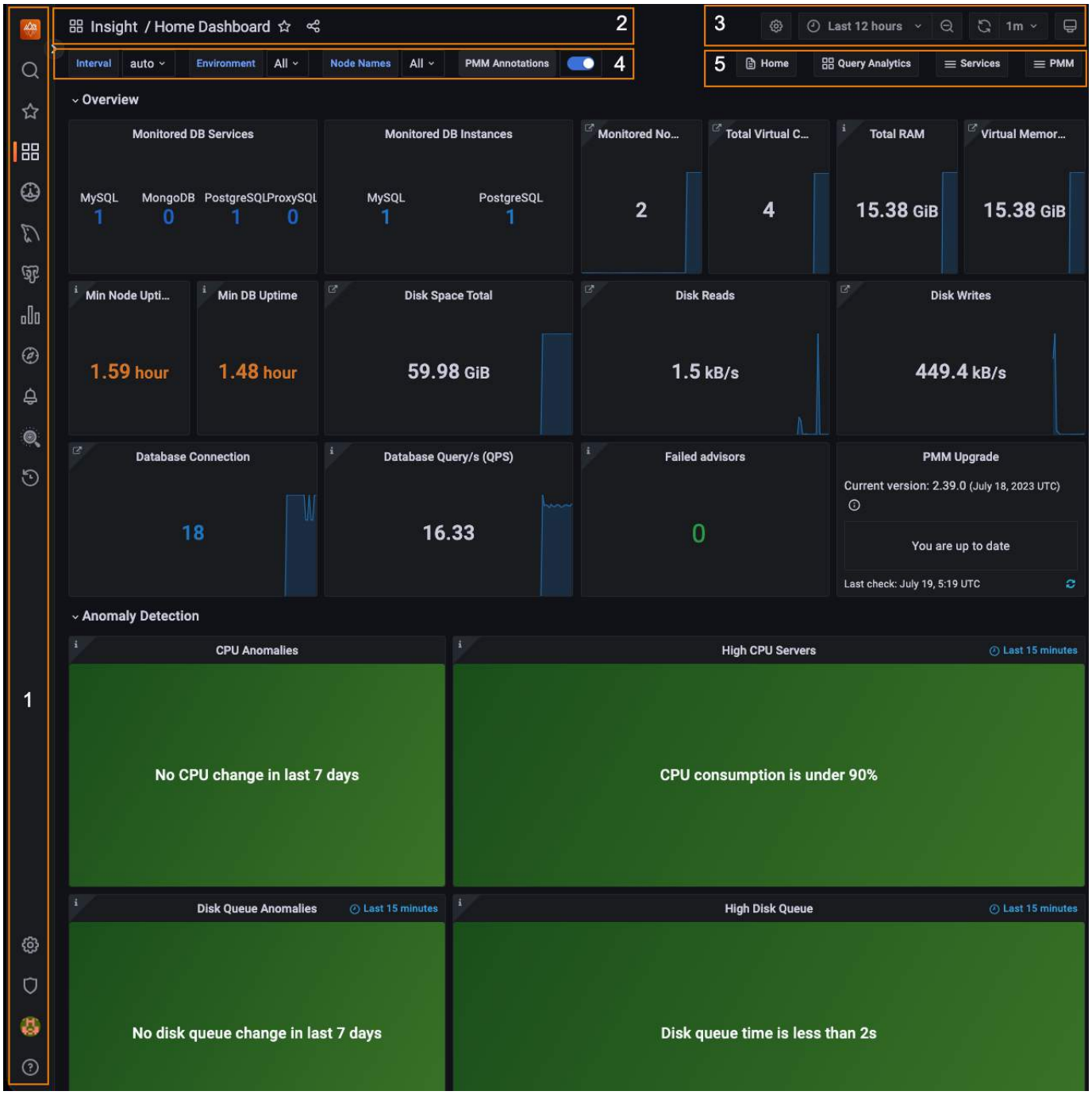
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2023-03-15

6.3 UI components



Percona LLC and/or its affiliates ©2024

Key

1. Main menu (also *Grafana menu*, *side menu*)
2. Navigation bar
3. View controls
4. View selectors (dynamic contents)
5. Shortcut menu (dynamic contents)

6.3.1 Main menu

The main menu is part of the Grafana framework and is visible on every page.

Item (Top)	Name	Description
	Home	Link to home dashboard.
	Search	Search dashboards by name.
	Starred	Mark your favorite dashboards.
	Dashboards	Create dashboards or folders, manage dashboards, import dashboards, create playlists, manage snapshots.
	Operating System (OS)	Operating System dashboard
	MySQL	MySQL dashboard
	MongoDB	MongoDB dashboard
	PostgreSQL	PostgreSQL dashboard
	ProxySQL	ProxySQL dashboard
	HAproxy	HAproxy dashboard
	Query Analytics (QAN)	Query Analytics
	Explore	Run queries with PromQL.
	Percona Alerting	Alerting, Create new alerts and manage your alert rules and alert templates.
	Configuration	
	Entitlements	This tab is displayed after connecting PMM to Percona Portal, and shows all your Percona Platform account information.
	Support Tickets	Shows the list of tickets opened across your organization. This tab is only available after you connect PMM to Percona Platform.
	Environment Overview	This tab is displayed after connecting PMM to Percona Portal. Shows the name and email of the Customer Success Manager assigned to your organization, who can help with any PMM queries. This tab will soon be populated with more useful information about your PMM environment.
	Server Admin	
	Backup Management	Backup management and storage location configuration.

Percona LLC and/or its affiliates ©2024

Item (Top)	Name	Description
	PMM Advisor Checks	Run health assessment checks against your connected databases and check any failed checks.
	DBaaS	

Tip

The DBaaS icon appears only if a server feature flag has been set.

The Backup Management icon appears when *Backup Management* is activated in *Configuration → Settings → Advanced Settings*.

Icon (Bottom)	Description
(Profile icon)	User menu
	Help

6.3.2 Navigation bar

Item (left)	Description
	(Display only.)
(Name) /	(Optional) Folder name.
(Name)	Dashboard name.
	Mark as favorite.
	Share dashboard.

6.3.3 View controls

Item (right)	Description
	Dashboard settings.
	Cycle view mode.
(time range)	Time range selector.
	Time range zoom out.
	Refresh dashboard.
(Time interval)	Refresh period.

6.3.4 View selectors

This menu bar is context-sensitive; it changes according to the page you are on. (With wide menus on small screens, items may wrap to the next row.)

Percona LLC and/or its affiliates ©2024

Item	Description
Interval	Data interval.
Region	Filter by region.
Environment	Filter by environment.
Cluster	Filter by cluster.
Replication Set	Filter by replication set.
Node Name	Filter by node name.
Service Name	Filter by service name.
PMM Annotations	View annotations.

6.3.5 Shortcut menu

This menu contains shortcuts to other dashboards. The list changes according to the page you're on.

> This menu will be removed in future releases. Its function will be replaced by the *PMM Dashboards* main menu entry.

Item	Description
Home	Home dashboard.
Query Analytics	Query Analytics.
Compare	Nodes compare.
(Service Type)	Service type menu (see Services menu).
HA	HA dashboards.
Services	Services menu.
PMM	PMM menu.

> **Tip**
>
> The *Compare* menu links to the Instances Overview dashboard for the current service type.

Services menu

The *Services* menu choice determines the Service Type menu.

Menu	Item	Service type menu	Description
Services			
	MongoDB Instances Overview	MongoDB	MongoDB dashboards.
	MySQL Instances Overview	MySQL	MySQL dashboards.
	Nodes Overview	OS	OS dashboards.
	PostgreSQL Instances Overview	PostgreSQL	PostgreSQL dashboards.

Percona LLC and/or its affiliates ©2024

Note

- You can easily access all dashboards for monitored services from the *Main menu* via *Other Dashboards* in the *Services* sub menu.
- Only the services being monitored by PMM will appear on the main menu.

PMM menu

This item lists shortcuts to utility pages.

Menu	Item
PMM	
	PMM Add Instance
	PMM Advisor Checks
	PMM Inventory
	PMM Settings

6.3.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

6.4 PMM components and versions

The following table lists all the PMM client/server components and their versions:

PMM client/server component	Version	Documentation	Location on GitHub
Grafana	9.2.20*	Grafana Documentation	Github Grafana
VictoriaMetrics	1.93.4	VictoriaMetrics Documentation	Github VictoriaMetrics
Nginx	1.20.1	Nginx Documentation	Github Nginx
Percona Distribution for PostgreSQL	14.5	Percona Distribution for PostgreSQL 14 Documentation	
Clickhouse	23.8.2.7	ClickHouse Documentation Documentation	Github ClickHouse
PerconaToolkit	3.4.0	Percona Toolkit Documentation	Github Percona Toolkit
Alertmanager	0.22.0	Alertmanager Documentation	Github Alertmanager
MongoDB exporter	0.37.0		Github MongoDB Exporter
MySQL exporter	v0.14.0*	MySQL Server Exporter Documentation	Github MySQL Server Exporter

Percona LLC and/or its affiliates ©2024

PMM client/server component	Version	Documentation	Location on GitHub
PostgreSQL exporter	v0.14.0*		Github PostgreSQL Server Exporter
RDS exporter	0.7.2		Github RDS Exporter
Node exporter	v1.4.0*	Node Exporter Documentation	Github Node Exporter
Azure exporter	2.30.0		Github Azure Metrics Exporter

* - Original upstream version along with some changes authored by Percona

6.4.1 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-12-12

6.5 Data handling in PMM

The following questions are being answered related to personal and confidential data handling in PMM:

1. Which type of data is transmitted?

Data collection source	Data collected
DB host to PMM	Database performance metrics SQL query examples for query analytics (optional).
PMM to DB Host	DSN and credentials for database access. A separate DB user is used (limited access) to retreive metrics from the database.
DB Host to S3 compatible storage location	Database backup - optional if PMM Administrator configures it with Public Cloud (AWS, GCP, etc) as a possible storage location.
PMM Server to Percona Cloud	Telemetry data is collected. PMM Server collects varying amounts of data from version to version, and no personal or confidential information is collected. See here for details on the data being transmitted.

2. Where is the data obtained from the DB host transmitted?

   All data gathered from the DB Host is transmitted to the PMM Server. It is possible to transmit DB backups to Cloud S3 storage (optional).

   Telemetry data is sent to Percona Cloud. This does not contain any sensitive or personally identifiable information.

3. What is the purpose and nature of data processing?

   As per our Privacy Policy, the data collection purposes are to provide the services and product enhancements.

   Although, PMM does not collect nor transfer personal data explicitly, in case query analytics is enabled and query examples collection is not disabled, we gather SQL query examples with real data and personal data may appear there if it is stored in DB. All QAN data always remains within the PMM server, and is never transmitted anywhere else.

Percona LLC and/or its affiliates ©2024

4. What is the frequency and volume of processed data?

   By default, metrics data is gathered every 5, 10 or 60 minutes. In case Query Analytics is enabled and SQL query examples are gathered every minute, we don't use any special processing for personal or confidential data. PMM server has no clue about the meaning of the data inside the SQL query.

   So it is processed as usual, which is to store inside the PMM server and present on the PMM UI by request.

   Other than email addresses for Grafana users, PMM does not directly ask or collect any other personal data. For more information about the telemetry data that is collected, please refer to the Percona Privacy Policy.

5. What applications or third parties can access the data created and processed by the cloud service?

   Third parties or other applications are not able to access the data gathered by the PMM server.

6. Is Personal Data processed for other applications or parties, and should the data that is processed in the cloud service be available to other applications or 3rd parties?

   PMM server doesn't pass any gathered, personal or confidential data to any third party or other applications nor to Percona Cloud.

7. How safe is the encryption?

   It's a must to encrypt all connections to and from the cloud including the data in the cloud storage and PMM does so by default.

   We use TLS (v1.2 at least) for connections between:

   - Database host to PMM Server (optionally, depending on user configuration)
   - PMM Server to Percona Cloud
   - PMM Server to remote database (optionally, depending on user configuration)
   - End-user to PMM Server web interface/api (self-signed by default)

   For more information about Percona security posture, please refer to our Trust Center here.

6.5.1 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-05-22

6.6 Develop Advisor checks

6.6.1 Developing Advisor checks

PMM offers sets of checks that can detect common security threats, performance degradation, data loss and data corruption.

As a developer, you can create custom checks to cover additional use cases, relevant to your specific database infrastructure.

Percona LLC and/or its affiliates ©2024

Check components

A check is a combination of:

- A query for extracting data from the database.
- Python script for converting extracted data into check results. This is actually a Starlark script, which is a Python dialect that adds more imperative features than Python. The script's execution environment is sandboxed, and no I/O can be done from it.

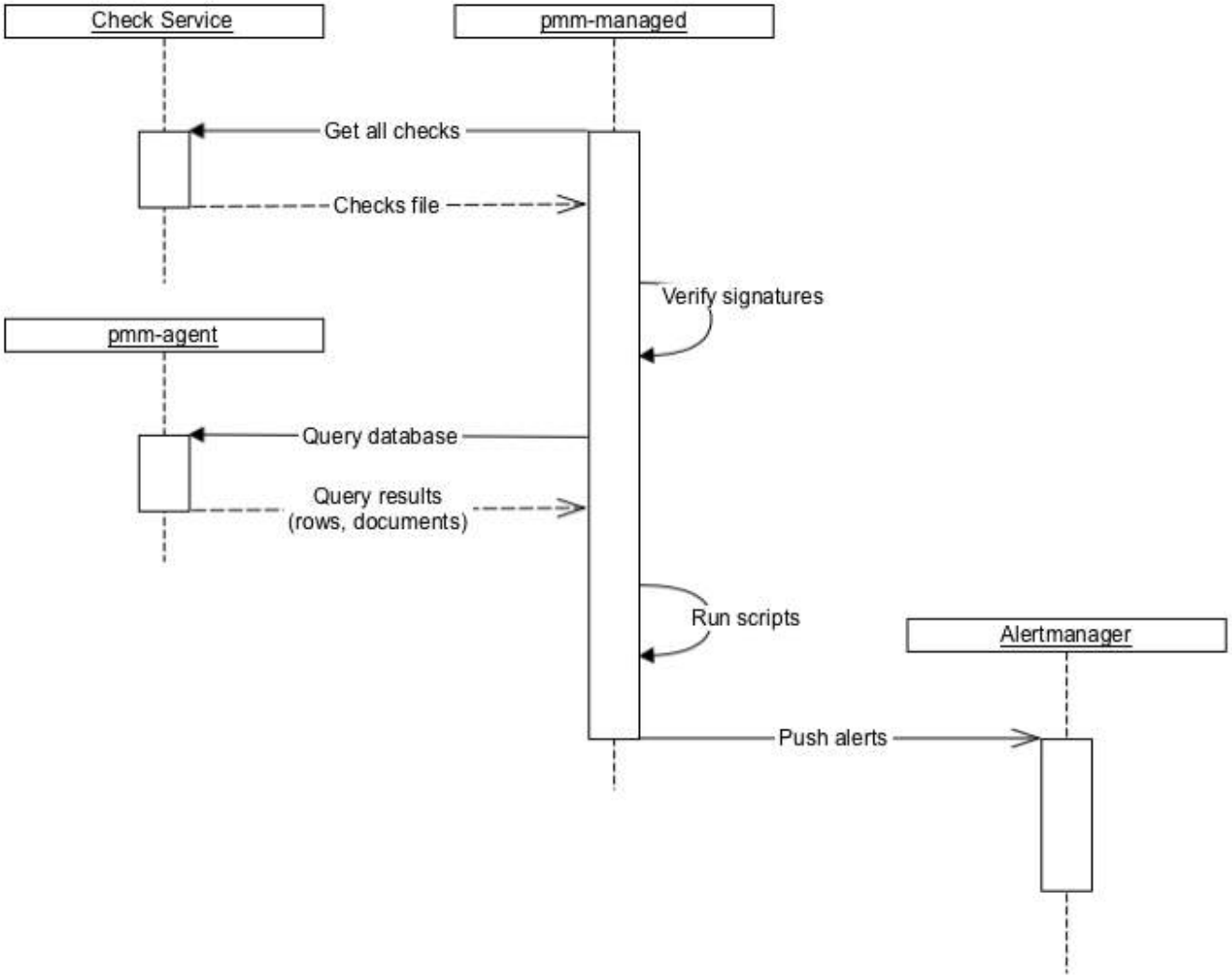
All checks are self-contained in the first phase, as well as in most of the planned phases.

This means that extracted data is processed on the PMM side and not sent back to Percona Platform.

Backend

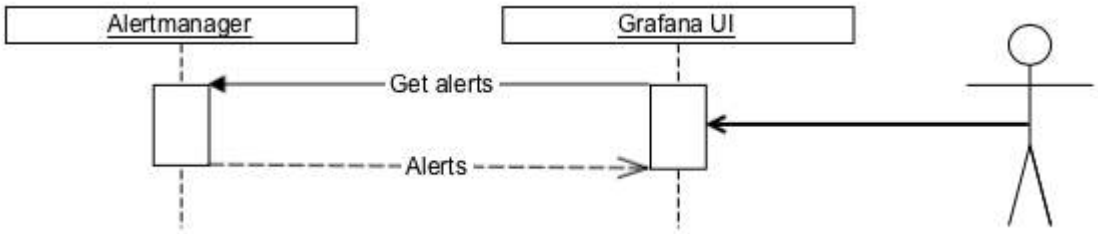
1. pmm-managed checks that the installation is opted-in for checks.
2. pmm-managed downloads files with checks from Percona Platform.
3. pmm-managed verifies file signatures using a list of hard-coded public keys. At least one signature should be correct.
4. pmm-managed sends queries to pmm-agent and gathers results.
5. pmm-managed executes check scripts that produce alert information.
6. pmm-managed sends alerts to Alertmanager.
7. Due to Alertmanager design, pmm-managed has to send and re-send alerts to it much more often than the frequency with which checks are executed. This expected behavior is not important for using checks but is important for understanding how checks work.
8. Currently, Prometheus is not involved.

Percona LLC and/or its affiliates ©2024



Frontend

PMM uses Alertmanager API to get information about failed checks and show them on the UI:



Check format versions

Starting with the 2.28 release, PMM uses Advisor checks format version 2. Format version 1 is deprecated.

Percona LLC and/or its affiliates ©2024

6.6.2 Version 2 advisor checks for PMM 2.28 and newer

PMM 2.28 upgraded Advisor Checks to version 2, which uses a slightly different structure than version 1 checks, created in 2.7 and earlier. This is because, compared to version 1 checks, checks created in 2.28 and later offer additional support for:

- Multiple queries
- Victoria Metrics as a data source
- Database **Family** field

Format for v.2 checks

Advisor checks for PMM 2.28 and later use the following format:

Percona LLC and/or its affiliates ©2024

Version 2 Checks Format

```
---
checks:
  - version: 2
    name: exampleV2
    summary: Check format V2
    description: Checks something important
    interval: standard
    family: MYSQL
    category: configuration ## Deprecated since PMM 2.36
    advisor: dev            ## Required since PMM 2.36
    queries:
      - type: MYSQL_SHOW
        query: VARIABLES

      - type: METRICS_INSTANT
        query: mysql_global_status_uptime{service_name=~"{{.ServiceName}}"}

      - type: METRICS_INSTANT
        query: mysql_global_status_uptime{service_name=~"{{.ServiceName}}"}
        parameters:
          lookback: 5m

      - type: METRICS_RANGE
        query: avg by (node_name) (avg_over_time(node_load1{node_name=~"{{.NodeName}}"}[5m]))
        parameters:
          range: 15m
          step: 5m

      - type: METRICS_RANGE
        query: avg by (node_name) (avg_over_time(node_load1{node_name=~"{{.NodeName}}"}[5m]))
        parameters:
          lookback: 5m
          range: 15m
          step: 5m

    script: |
      def check_context(docs, context):
          # `docs` is a frozen (deeply immutable) list where each item represents single query results. Order of results
          # matches order of queries in check file. Each query result is list of dicts where each item where each dict
          # represents a single document in result set.
          #
          # `context` is a dict with additional functions.
          #
          # Global `print` and `fail` functions are available.
          #
          # `check_context` function is expected to return a list of dicts that are then converted to alerts;
          # in particular, that list can be empty.
          # Any other value (for example, string) is treated as a script execution failure
          # (Starlark does not support Python exceptions);
          # it is recommended to use global function `fail` for that instead.

          results = []

          for row in docs[0]:
              name, value = row["Variable_name"], row["Value"]
              if name == "version":
                  results.append({
                      "summary": "MySQL has version {}".format(value),
                      "description": "Current version is {}".format(value),
                      "read_more_url": "",
                      "severity": "warning",
                      "labels": {},
                  })

          uptimeNow = int(int(docs[1][0]["value"][1])/60)
          results.append({
              "summary": "MySQL uptime {} min".format(uptimeNow),
              "description": "Current uptime is {} min".format(uptimeNow),
              "read_more_url": "",
              "severity": "warning",
              "labels": {},
          })

          uptimeFiveMinAgo = int(int(docs[2][0]["value"][1])/60)
          results.append({
              "summary": "MySQL uptime 5 min ago was {} min".format(uptimeFiveMinAgo),
              "description": "5 min ago uptime was {} min".format(uptimeFiveMinAgo),
              "read_more_url": "",
              "severity": "warning",
              "labels": {},
          })

          dataPoints = []
          for row in docs[3][0]["values"]:
            dataPoints.append(row[1])

          results.append({
              "summary": "Node has load average for last 15 minutes {}".format(dataPoints),
              "description": "Data points {}".format(dataPoints),
```


Percona LLC and/or its affiliates ©2024

```
        "read_more_url": "",
        "severity": "warning",
        "labels": {},
    })

    dataPoints = []
    for row in docs[4][0]["values"]:
        dataPoints.append(row[1])

    results.append({
        "summary": "Five minutes ago node had load average for 15 minutes {}".format(dataPoints),
        "description": "Data points {}".format(dataPoints),
        "read_more_url": "",
        "severity": "warning",
        "labels": {},
    })

    return results
```

Checks script

The check script assumes that there is a function with `check_context`, that accepts a *list* where each item represents the result of a single query specified in the check. Each result itself is a *list* of *docs* containing returned rows for SQL databases and documents for MongoDB. It returns zero, one, or several check results that are then converted to alerts.

Check severity levels

You can label your advisor checks with one of the following available severity levels: **Emergency**, **Alert**, **Critical**, **Error**, **Warning**, **Notice**, **Info**, **Debug**. PMM groups failed checks by their severity, and displays them under **Advisors Checks > Failed Checks**.

Check fields

Checks can include the following fields:

- **Version** (integer, required): defines what other properties are expected, what types are supported, what is expected from the script and what it can expect from the execution environment, etc.
- **Name** (string, required): defines machine-readable name (ID).
- **Summary** (string, required): defines short human-readable description.
- **Description** (string, required): defines long human-readable description.
- **Family** (string, required): specifies one of the supported database families: MYSQL, POSTGRESQL, MONGODB. This field is only available for Advisor checks v.2, created for PMM 2.28 and later.
- **Advisor** (string, required): specifies the advisor to which this check belongs. For local environments, specify **dev**.
- **Interval** (string/enum, optional): defines running interval. Can be one of the predefined intervals in the UI: Standard, Frequent, Rare.
- **Queries** (array, required): contains items that specify queries.
  - **Type** (string/enum, required): defines the query type. Check the list of available types in the table below.
  - **Query** (string, can be absent if the type defines the whole query by itself): The query is executed on the PMM Client side and can contain multiple queries specific to the target DBMS.
  - **Parameters** (key-value, can be absent if query doesn’t have required parameters)
- **Script** (string, required): contains a small Starlark script that processes query results, and returns check results. It is executed on the PMM Server side.

Percona LLC and/or its affiliates ©2024

Query types

Expand the table below for the list of checks types that you can use to define your query type and the PMM Service type for which the check will run.

Check Types table

Check type	Description	"query" required (must be empty if "No")
MYSQL_SHOW	Executes 'SHOW ...' clause against MySQL database.	Yes
MYSQL_SELECT	Executes 'SELECT ...' clause against MySQL database.	Yes
POSTGRESQL_SHOW	Executes 'SHOW ALL' command against PosgreSQL database.	No
POSTGRESQL_SELECT	Executes 'SELECT ...' clause against PosgreSQL database.	Yes
MONGODB_GETPARAMETER	Executes db.adminCommand({ getParameter: "*" }) against MongoDB's "admin" database. For more information, see getParameter	No
MONGODB_BUILDINFO	Executes db.adminCommand({ buildInfo: 1 }) against MongoDB's "admin" database. For more information, see buildInfo	No
MONGODB_GETCMDLINEOPTS	Executes db.adminCommand({ getCmdLineOpts: 1 }) against MongoDB's "admin" database. For more information, see getCmdLineOpts	No
MONGODB_REPLSETGETSTATUS	Executes db.adminCommand({ replSetGetStatus: 1 }) against MongoDB's "admin" database. For more information, see replSetGetStatus	No
MONGODB_GETDIAGNOSTICDATA	Executes db.adminCommand({ getDiagnosticData: 1 }) against MongoDB's "admin" database. For more information, see MongoDB Performance	No
METRICS_INSTANT	Executes instant MetricsQL query. Query can use placeholders in query string **{{.NodeName}}** and **{{.ServiceName}}** . Both match target service/node names. To read more about instant queries, check out the Prometheus docs.	Yes
METRICS_RANGE	Executes range MetricsQL query. Query can use placeholders in query string **{{.NodeName}}** and **{{.ServiceName}}** . Both match target service/node names. To read more about range queries, check out the Prometheus docs.	Yes
CLICKHOUSE_SELECT	Executes 'SELECT ...' statements against PMM's Query Analytics Clickhouse database. Queries can use the **{{.ServiceName}}** and **{{.ServiceID}}** placeholders in query string. They match the target service name and service ID respectively.	Yes

Percona LLC and/or its affiliates ©2024

Query parameters

- `METRICS_INSTANT`
  - **lookback** (duration, optional): specifies how far in past to look back to metrics history. If this parameter is not specified, then query executed on the latest data. Example values: `30s`, `5m`, `8h`.
- `METRICS_RANGE`
  - **lookback** (duration, optional): specifies how far in past to look back to metrics history. If this parameter is not specified, then query executed on the latest data. Example values: `30s`, `5m`, `8h`.
  - **range** (duration, required): specifies time window of the query. This parameter is equal to Prometheus API.
  - **step** (duration, required): query resolution. This parameter is equal to Prometheus API.
- `POSTGRESQL_SELECT`
  - **all_dbs** (boolean, optional): execute query on all available databases in PostgreSQL instance. If this parameter is not specified, then query executed on the default database (the one that was specified when service was added to PMM).

Develop checks

> **Development / Debugging Only**

Note that check development in PMM is currently for **debugging only** and **NOT for production use!** Future releases plan to include the option to run custom local checks in addition to hosted Percona Platform checks.

To develop custom checks for PMM:

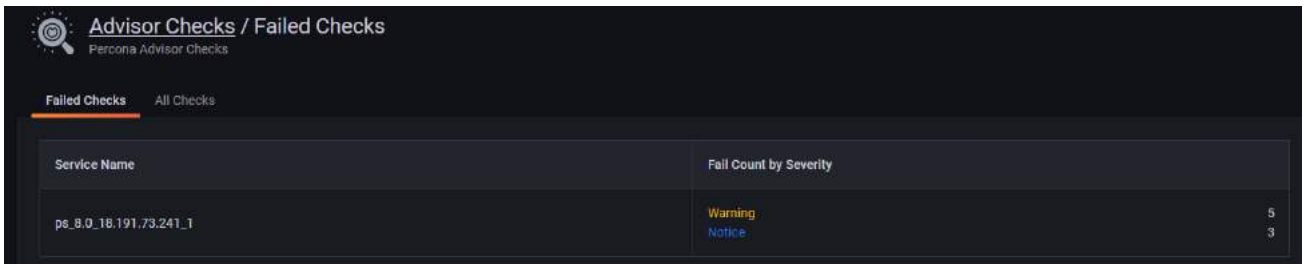
1. Install the latest PMM Server and PMM Client builds following the installation instructions.
2. Run PMM Server with special environment variables:
   - `PERCONA_TEST_CHECKS_FILE=/srv/custom-checks.yml` to use checks from the local files instead of downloading them from Percona Platform.
   - `PERCONA_TEST_CHECKS_DISABLE_START_DELAY=true` to disable the default check execution start delay. This is currently set to one minute, so that checks run upon system start.
   - `PERCONA_TEST_CHECKS_RESEND_INTERVAL=2s` to define the frequency for sending the SA-based alerts to Alertmanager.

```
docker run -p 80:80 -p 443:443 --name pmm-server \
-e PERCONA_TEST_CHECKS_FILE=/srv/custom-checks.yml \
-e PERCONA_TEST_CHECKS_DISABLE_START_DELAY=true \
-e PERCONA_TEST_CHECKS_RESEND_INTERVAL=2s \
perconalab/pmm-server:dev-latest
```

3. Log in to Grafana with credentials **admin/admin**.
4. Go to **Configuration > Settings > Advanced Settings** and make sure the **Advisors** option is enabled.
5. Create `/srv/custom-checks.yml` inside the `pmm-server` container with the content of your check. Specify **dev** advisor in your check.
6. The checks will run according to the time interval defined on the UI. You can see the result of running the check on the home dashboard:

Percona LLC and/or its affiliates ©2024

7. Click on the number of failed checks to open the Failed Checks dashboard:



8. Check out pmm-managed logs:

```
docker exec -it pmm-server supervisorctl tail -f pmm-managed
```

6.6.3 Troubleshooting and tips

- in Debug mode, PMM generates a lot of redundant information in the log files, information that is not useful for developing checks. If debug logging is enabled, you can disable it with the following environment variable: `PMM_DEBUG=0`.
- All logs from checks subsystem has `component=checks` tag, so you can just filter `pmm-managed` logs with grep.
- Local check file should always be linked to fake dev advisor: `advisor: dev`. If PMM does not display the **Development** tab on the **Advisors** page, make sure that you specify dev advisor in the check file.
- If this still doesn't display the **Development** tab, probably PMM could not load your file due to formatting issues. Check pmm-managed logs for details.
- There are to ways to reload the check file:
- Click `Run check` button (but it's unavailable if you don't have any tabs on advisors page and most likely that is the case during development)
- Reload managed: `supervisorctl restart pmm-managed` (execute inside PMM server)

Submit feedback

We welcome your feedback on the current process for developing and debugging checks. Send us your comments over Slack or post a question on the Percona Forums.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-05-09

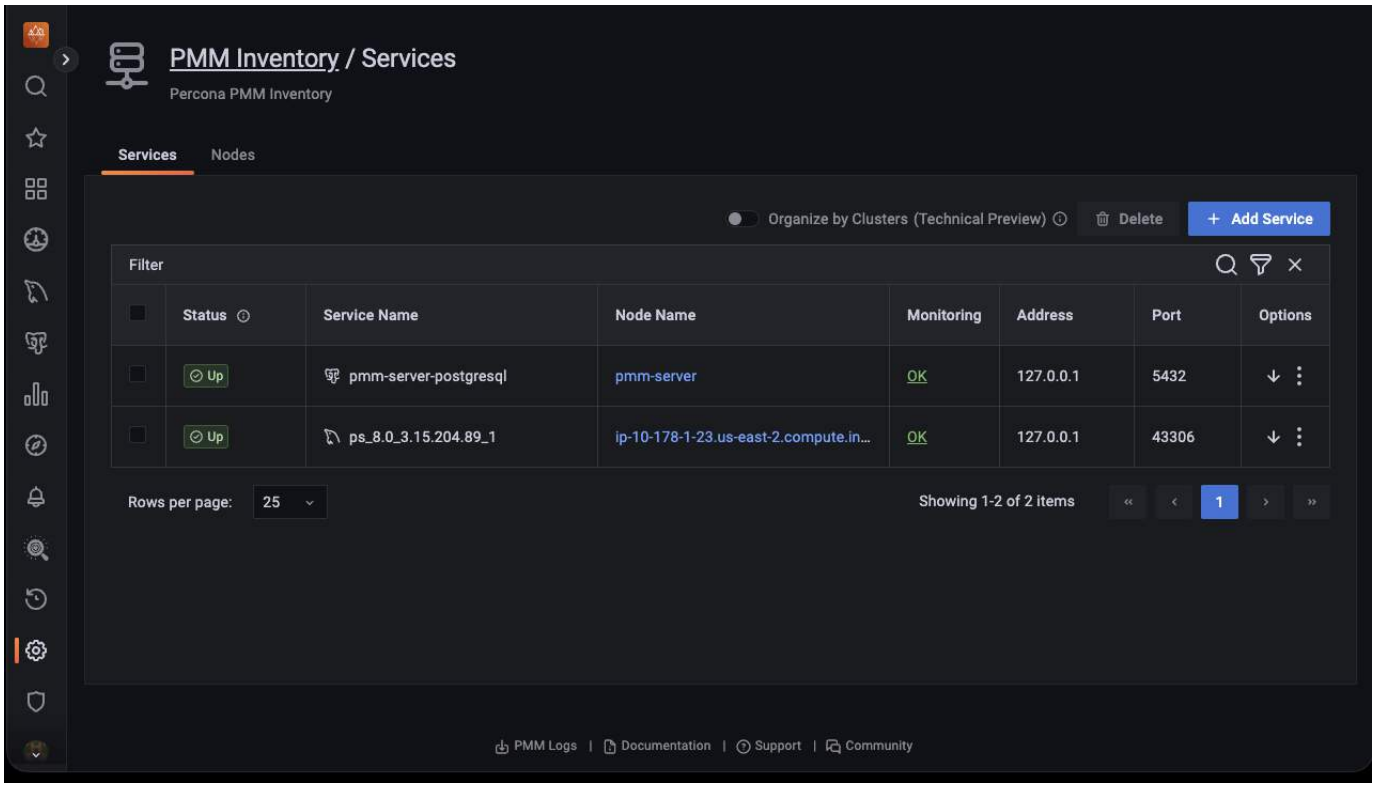
Percona LLC and/or its affiliates ©2024

6.7 PMM Inventory

6.7.1 PMM Inventory

The **Inventory** dashboard is a high-level overview of all objects registered in PMM.

To check your inventory list, go to **Configuration → Inventory**.



Inventory objects form a hierarchy with Node at the top, then Service and Agents assigned to a Node. This information is detailed in the two tabs available on this page.

SERVICES TAB

The **Services** tab displays the individual services, the nodes on which they run, and the Agents that help collect the service metrics along with the following information:

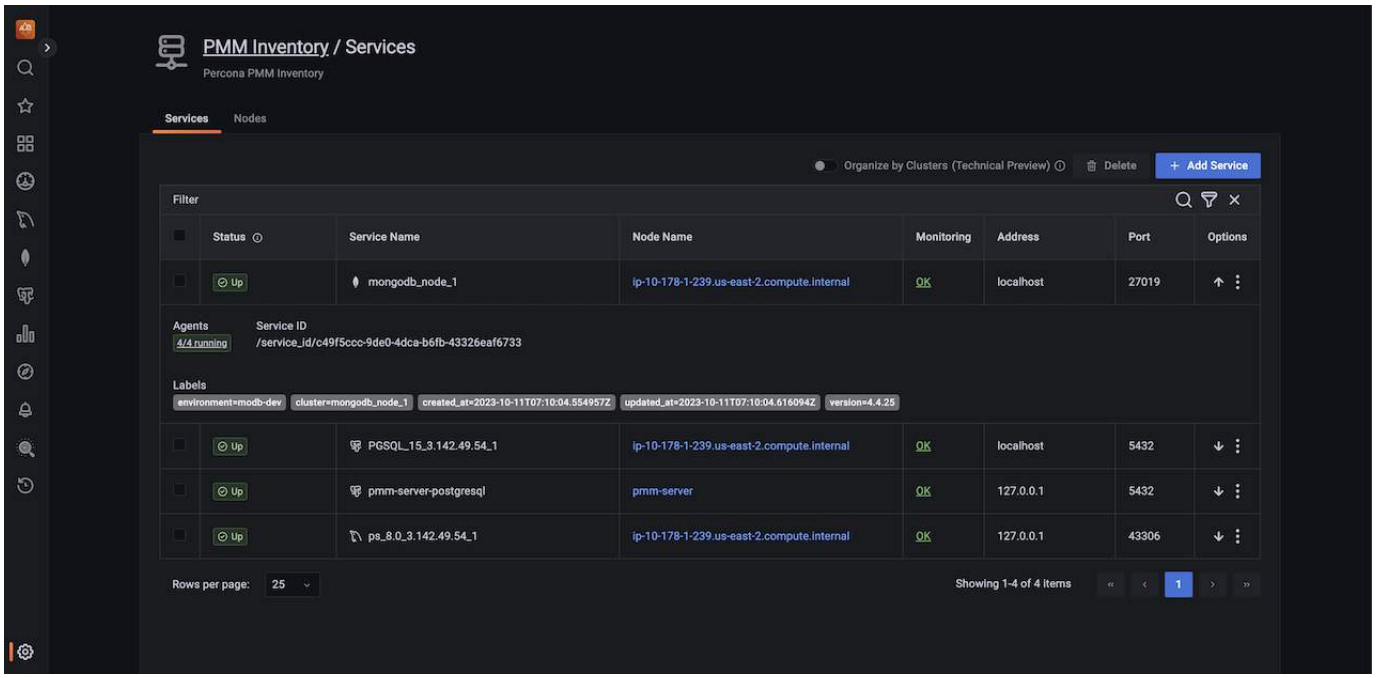
You can check Query Analytics information and the Service Overview Dashboard for each service by clicking on the icon in the **Options** column.

From here you can also check additional information about the service, by clicking on the icon. This expands the service entry to show reference information like service labels and IDs.

Column Name	Description
Service name	The name or identifier associated with the service being monitored.
Node name	Name or identifier associated with a specific node.
Monitoring status	The **Monitoring** column summarizes the status of all the Agents assigned to the service.
Address	The IP address or DNS where the service is currently running.
Port	The port number on which the service is running.

Percona LLC and/or its affiliates ©2024

Column Name	Description
Options	* You can check **QAN** information and the **Dashboard** for each service by clicking on the icon * You can also check additional information about the service, by clicking on the icon. This expands the service entry to show reference information like service labels and IDs.



Attributes

These are some of the atributes for a service:

- Each instance of a service gets a `service_type` attribute so one can clearly tell what type of database it is, for instance: `mysql`, `postgresql`, `mongodb`, etc.
- Every service is related to a certain node via its `node_id` attribute. This feature allows to support monitoring of multiple instances on a single node, with different service names, e.g. `mysql1-3306`, and `mysql1-3307`.
- Starting with PMM 2.41.0, each instance of a service gets a `version` attribute to the response of the endpoint that provides a list of services being monitored by PMM. This makes it easy to visualize the database server version.

  However, following are the imitations:

  - The version is not captured for the internal PostgreSQL database.
  - The version is only captured when a new service is being added to PMM and the agent installed on the client side is equal to or greater than v2.41.0.
  - When a database is upgraded, you will not see the database version updated automatically. It will be updated if you remove and then re-add the service.

Agents

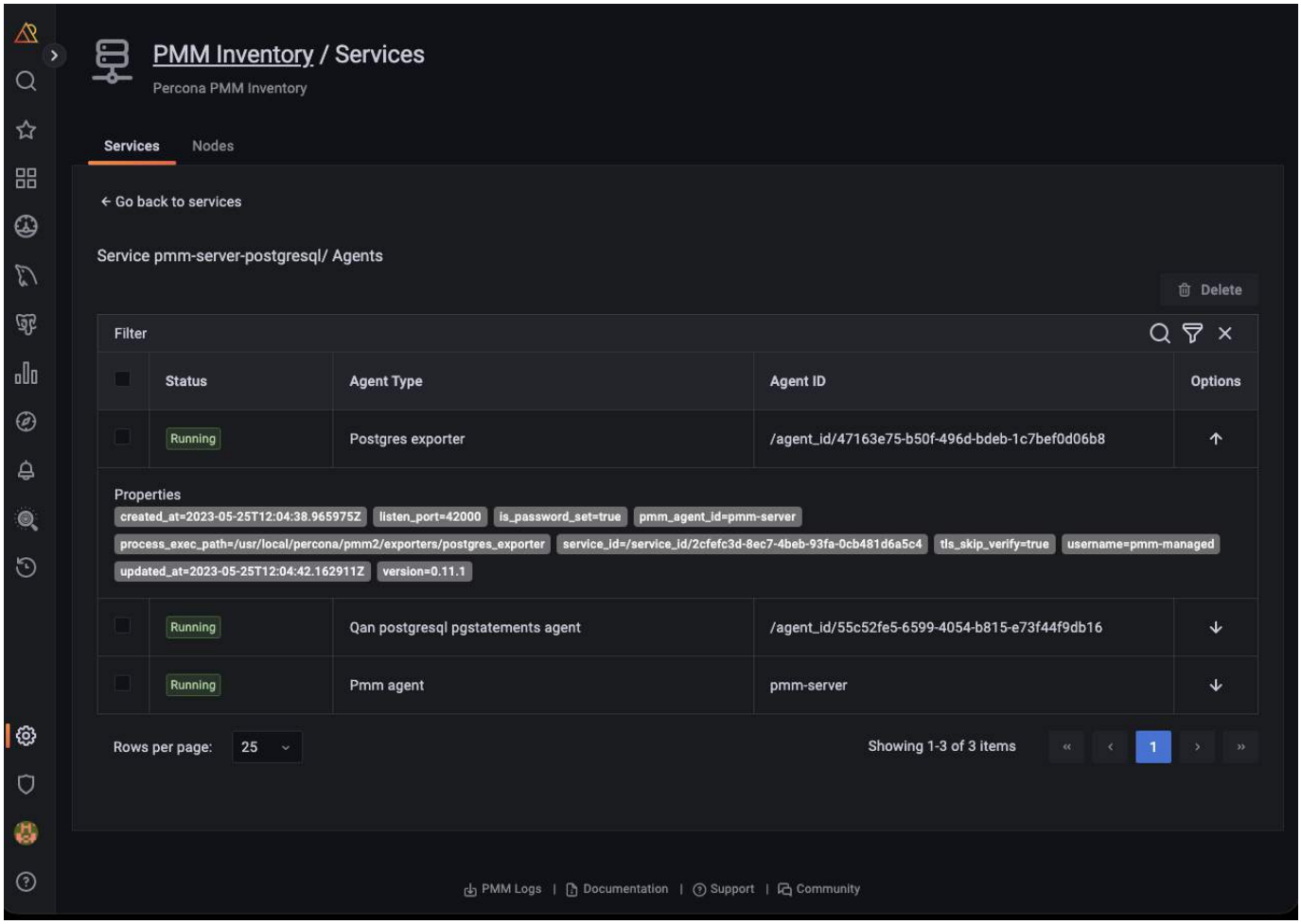
Each binary (exporter, agent) running on a client will get an `agent_type` value.

Example

- `pmm-agent` is at the top of the tree, assigned to PMM Agent itself
- `node_exporter` is assigned to an agent that extracts the node metrics
- `mysqld_exporter` and `qan-mysql-perfschema-agent` are assigned to agents that extract metrics from mysql and its performance schema respectively.

Percona LLC and/or its affiliates ©2024

To view the agents running on a service and their health status, click **OK** or **Failed** under the **Monitoring** column. Furthermore, you can also check the properties of a particular agent by clicking the icon under the **Options** column.



Node-service relationship

Starting with PMM 2.40.0, you can click on the link in the **Node Name** column to view the node on which a specific service is running and analyze how node-level resource utilization impacts the performance of those services.

Understanding the relationship between nodes and services is key to gaining insights into the distribution and performance of individual services across nodes.

- **Deployment**: Services within PMM are deployed on nodes and rely on them for resources, such as CPU, memory, and storage, to execute tasks.
- **Resource allocation**: It is essential to know which nodes host which services to allocate resources appropriately to avoid underuse or overload.
- **Performance optimization**: By analyzing node and service-level metrics, you can pinpoint and resolve issues that impede service performance, such as resource limitations and performance bottlenecks.
- **Incident response**: When an issue or incident occurs, understanding the node-service relationship helps in troubleshooting. You can quickly identify which nodes and services are affected and focus your efforts on resolving the problem.

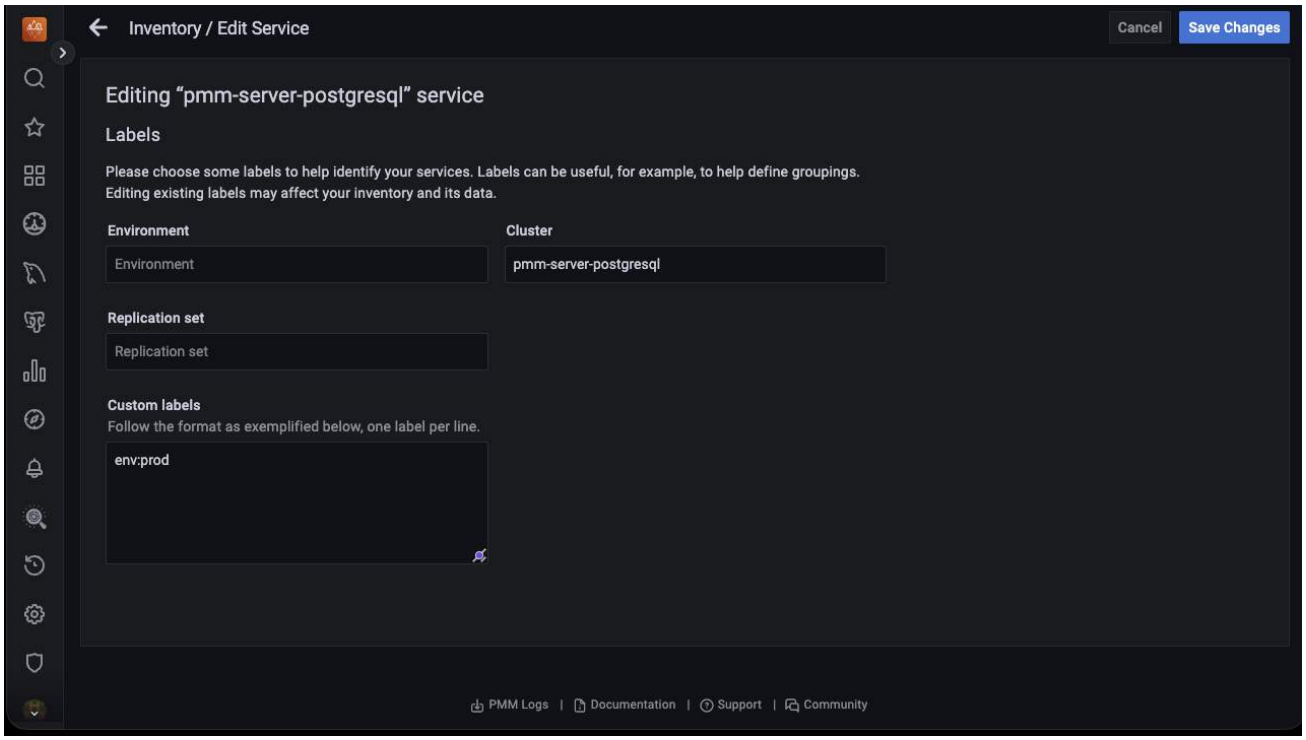
EDITING LABELS FOR A SERVICE

You can edit the labels as follows:

1. From the **Main** menu, navigate to **Configuration → Inventory**.

Percona LLC and/or its affiliates ©2024

2. Click on the three dots next to the service you want to edit labels for.
3. Click **Edit**. The **Edit Service** page opens.
4. Edit the labels as per your requirement and click **Save Changes**. The editing service dialogue box opens.



5. Click **Confirm and save changes**. You will be taken back to the **Inventory/Services** page.

Effect of editing labels for a service

Editing existing labels can impact the following PMM functions:

- **Alerting**

  Editing labels without updating alerting rules can lead to missed alerts. If an alert rule is based on specific labels that are changed or no longer apply, the alert may not trigger when it should.

  Update the alert rules promptly after editing the labels for a smooth alerting experience.

- **Scheduled backup**s: Editing the cluster label will remove all scheduled backups for the imapcted service or cluster.

  To prevent any issues, make sure to recreate your backups once you’ve configured the cluster.

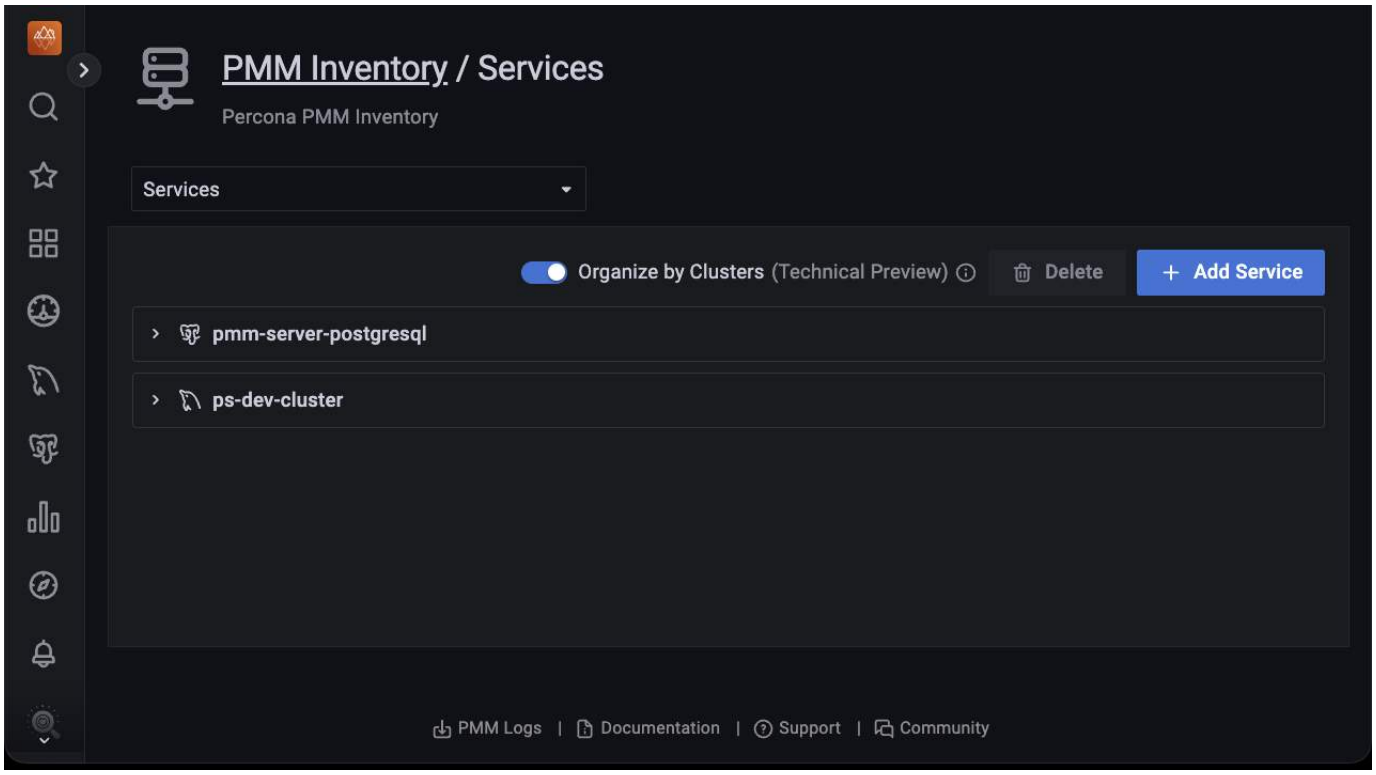
- **Dashboard data**: Edited labels do not affect the existing time-series(metrics). It will only affect the new time-series(metrics).

Cluster view

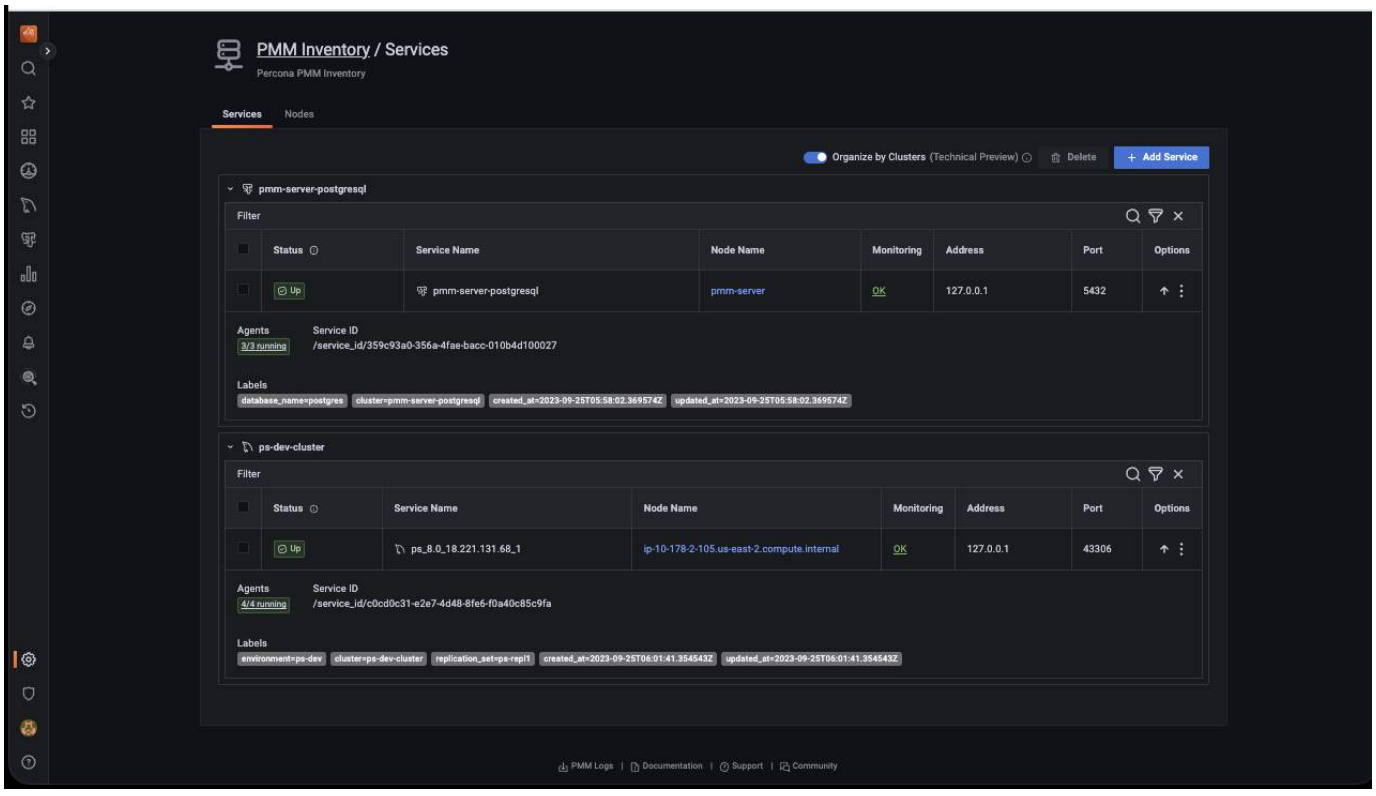
> **Disclaimer**
>
> This feature is still technical preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

Starting with PMM 2.40.0, you can choose to view a group of services as a single cluster with the **Organize by Clusters** toggle. PMM uses the `cluster` label to display services under the same cluster.

Percona LLC and/or its affiliates ©2024

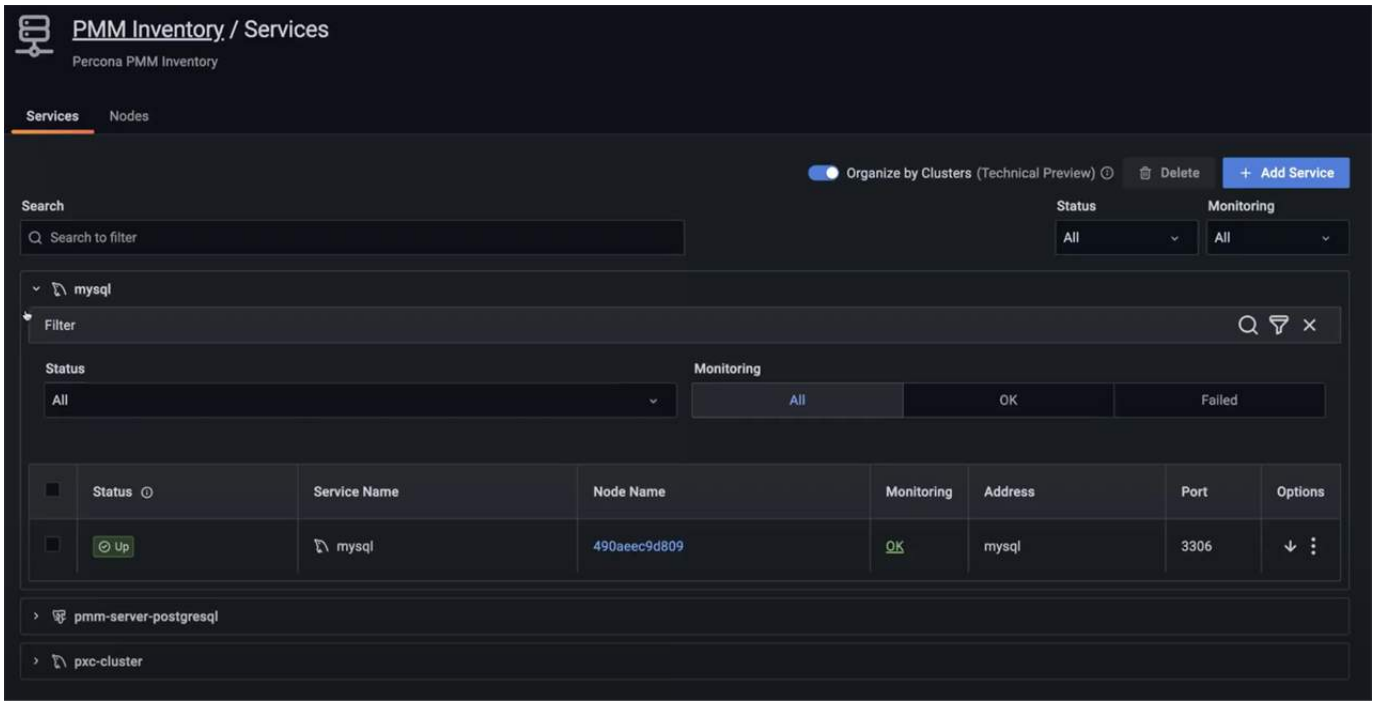


Click the downward arrow to view cluster details, including the services running on that cluster, agents, and labels.



Furthermore, you can filter the clusters by criteria such as Cluster name, Status, Service name, Node name, Monitoring, Address, and Port.

Percona LLC and/or its affiliates ©2024



NODES TAB

Shows where the service and agents run.

Each `node_id` is associated with a `machine_id` (from `/etc/machine-id`). Nodes also have `node_type` attributes, which give an idea about their nature. Some examples are: generic, container, remote, remote_rds, etc.

By expanding the entry from the options column, you can check the node labels and attributes.

Starting with PMM 2.38.0, you can see the number of agents running on any particular node. When you click on any node, the UI navigates to the view of agents, which is filtered to display only agents related to that specific node.

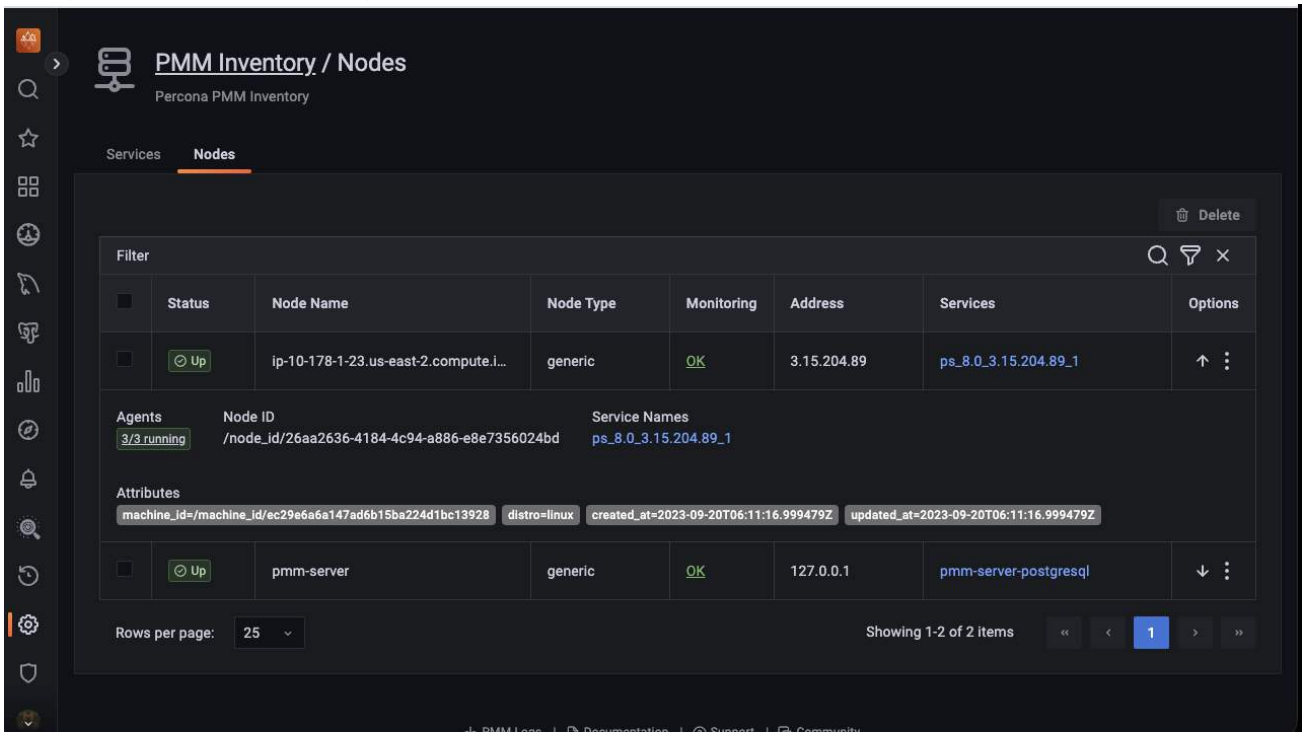
Furthermore, starting with PMM 2.40.0, you can see the service running on that specific node when you click on the link in the **Services** column.

To see the details of the agents running, do the following:

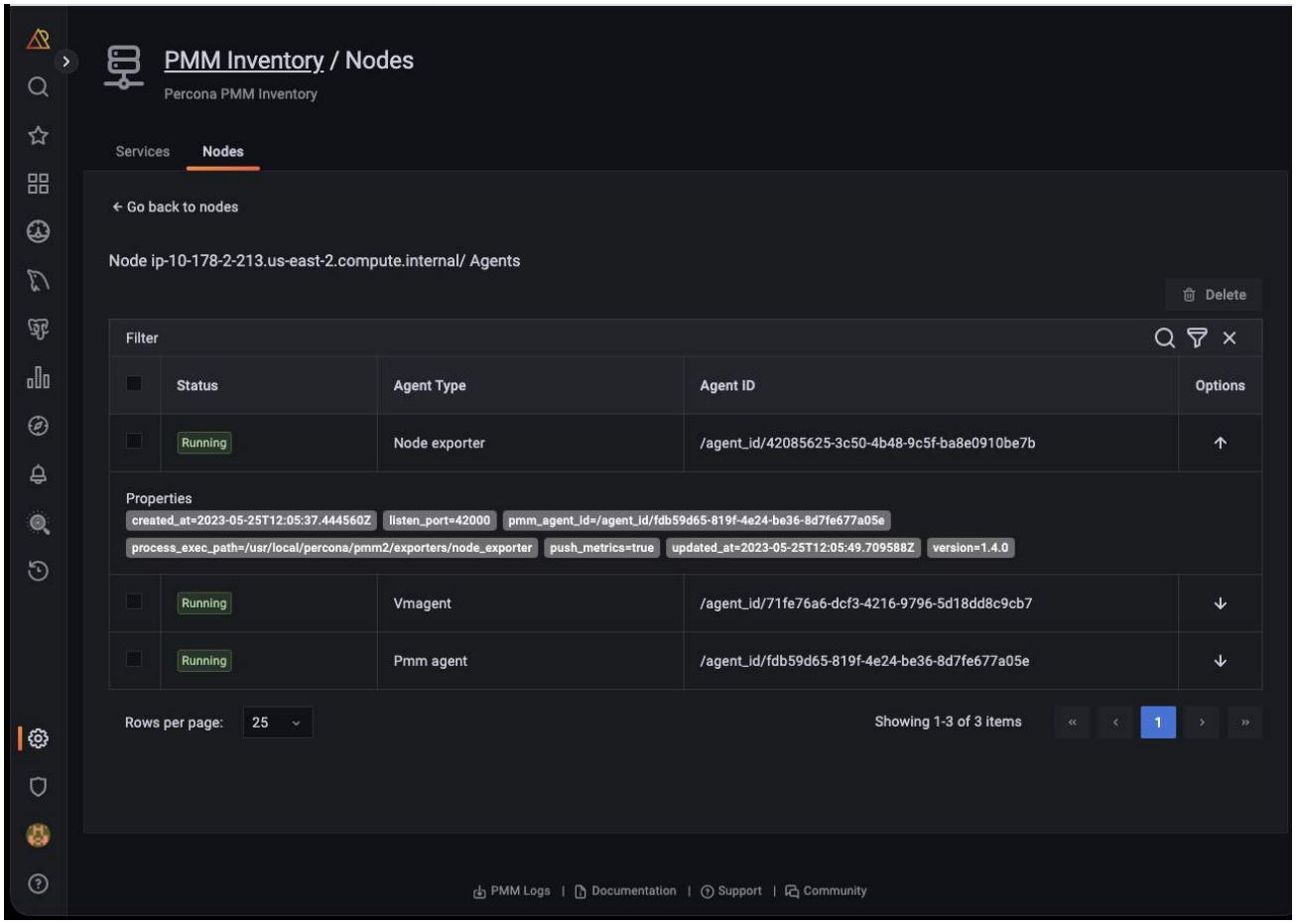
1. On the **Nodes** tab, under the **Monitoring** column, click **OK** or **Failed** depending on the status of the node that you have selected. A page that provides the user with crucial information regarding the total number of agents

Percona LLC and/or its affiliates ©2024

deployed on that node is displayed.



2. Click on the icon under the **Options** column to view the properties of a specific agent.
3. On the **Nodes** tab, under the **Options** column, click on the icon for the selected node to check the properties and the current health status of an agent.

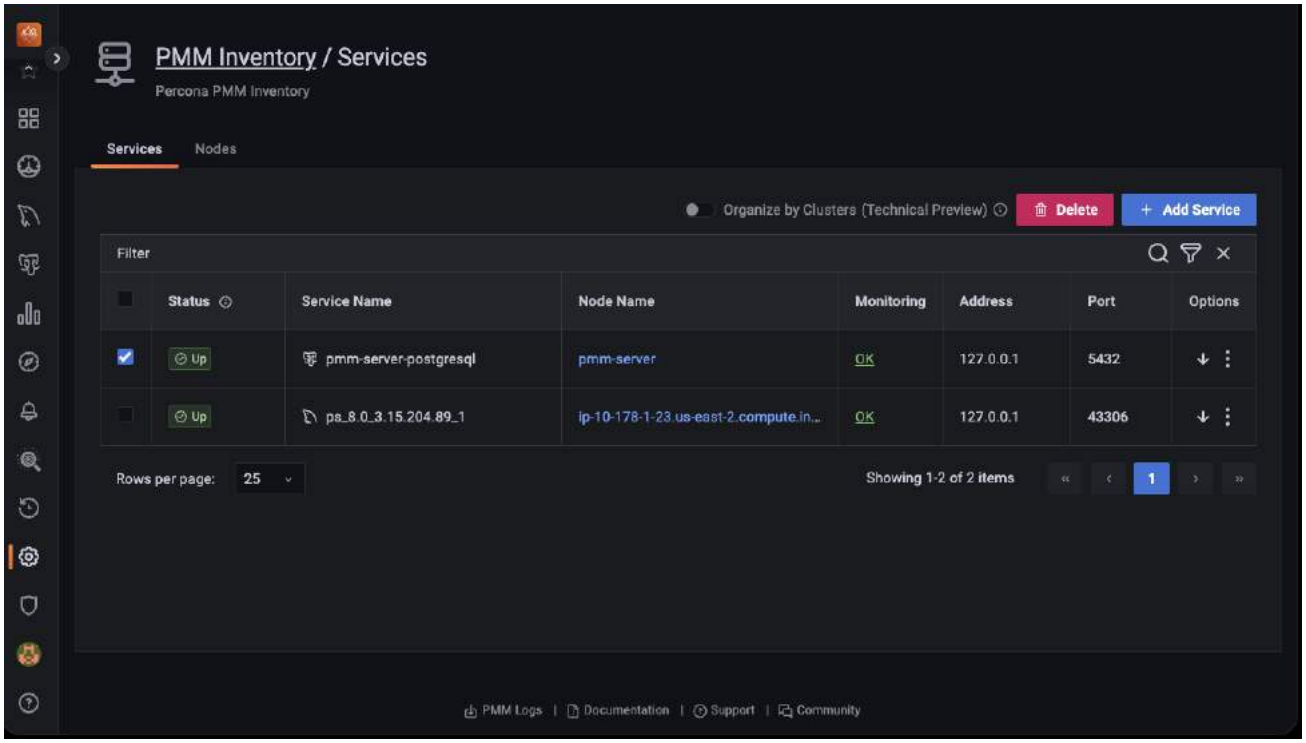


Percona LLC and/or its affiliates ©2024

Removing items from the inventory

To remove items from the inventory:

1. Go to **Configuration** → **Inventory**.
2. In the first column, select the items to be removed.



3. Click **Delete** and confirm the removal.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

6.8 Dashboards

6.8.1 Dashboards

Category	Dashboard	Elements
Kubernetes	Kubernetes Cluster Summary	7
Kubernetes	Kubernetes Pods Status	26
Kubernetes	Kubernetes Volumes	57
Insight	Advanced Data Exploration	7

Percona LLC and/or its affiliates ©2024

Category	Dashboard	Elements
Insight	Home Dashboard	26
Insight	Prometheus Exporter Status	57
Insight	Prometheus Exporters Overview	27
Insight	VictoriaMetrics	52
Insight	VictoriaMetrics Agents Overview	58
PMM	Environment Overview	0
PMM	Environment Summary	0
DBaaS	DB Cluster Summary	0
OS	CPU Utilization Details	21
OS	Disk Details	34
OS	Network Details	70
OS	Memory Details	116
OS	Node Temperature Details	6
OS	Nodes Compare	74
OS	Nodes Overview	115
OS	Node Summary	67
OS	NUMA Details	72
OS	Processes Details	35
Prometheus	Prometheus Exporter Status	57
Prometheus	Prometheus Exporters Overview	27
MySQL	MySQL Amazon Aurora Details	20
MySQL	MySQL Command/Handler Counters Compare	11
MySQL	MySQL InnoDB Compression Details	41
MySQL	MySQL InnoDB Details	339
MySQL	MySQL MyISAM/Aria Details	55
MySQL	MySQL MyRocks Details	101
MySQL	MySQL Instance Summary	90
MySQL	MySQL Instances Compare	70
MySQL	MySQL Instances Overview	96
MySQL	MySQL Wait Event Analyses Details	42
MySQL	MySQL Performance Schema Details	48
MySQL	MySQL Query Response Time Details	49
MySQL	MySQL Replication Summary	50
MySQL	MySQL Group Replication Summary	18
MySQL	MySQL Table Details	45
MySQL	MySQL User Details	62
MongoDB	Experimental MongoDB Collection Overview	100

Percona LLC and/or its affiliates ©2024

Category	Dashboard	Elements
MongoDB	Experimental MongoDB Collection Details	100
MongoDB	Experimental MongoDB Oplog Details	100
MongoDB	MongoDB Cluster Summary	55
MongoDB	MongoDB Instance Summary	42
MongoDB	MongoDB Instances Compare	19
MongoDB	MongoDB ReplSet Summary	130
MongoDB	MongoDB InMemory Details	46
MongoDB	MongoDB MMAPv1 Details	52
MongoDB	MongoDB WiredTiger Details	54
PostgreSQL	PostgreSQL Instances Overview	114
PostgreSQL	Experimental PostgreSQL Vacuum Monitoring	114
PostgreSQL	PostgreSQL Instance Summary	67
PostgreSQL	PostgreSQL Instances Compare	89
ProxySQL	ProxySQL Instance Summary	55
High-availability	PXC/Galera Node Summary	32
High-availability	PXC/Galera Cluster Summary	19
High-availability	Experimental PXC/Galera Cluster Summary	7
High-availability	PXC/Galera Nodes Compare	55
High-availability	HAProxy Instance Summary	113

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-07-03

6.8.2 Manage dashboards

Manage dashboards

This section describes how to manage your PMM dashboards and the widgets on those dashboards, including:

- Creating dashboard folders
- Managing dashboard folders
- Setting custom Home dashboard

Percona LLC and/or its affiliates ©2024

CREATE DASHBOARD FOLDERS

Folders help you organize and group PMM dashboards, which is crucial when you have multiple dashboards or teams using the same PMM instance.

> **Note**
>
> To create a dashboard folder, you must have PMM's *Admin* privileges.

To create a dashboard folder:

1. On the PMM dashboards page, from the side menu, go to *Dashboards* > *New folder*.
2. Enter a unique name for your folder and click *Create*.

MANAGING DASHBOARD FOLDERS

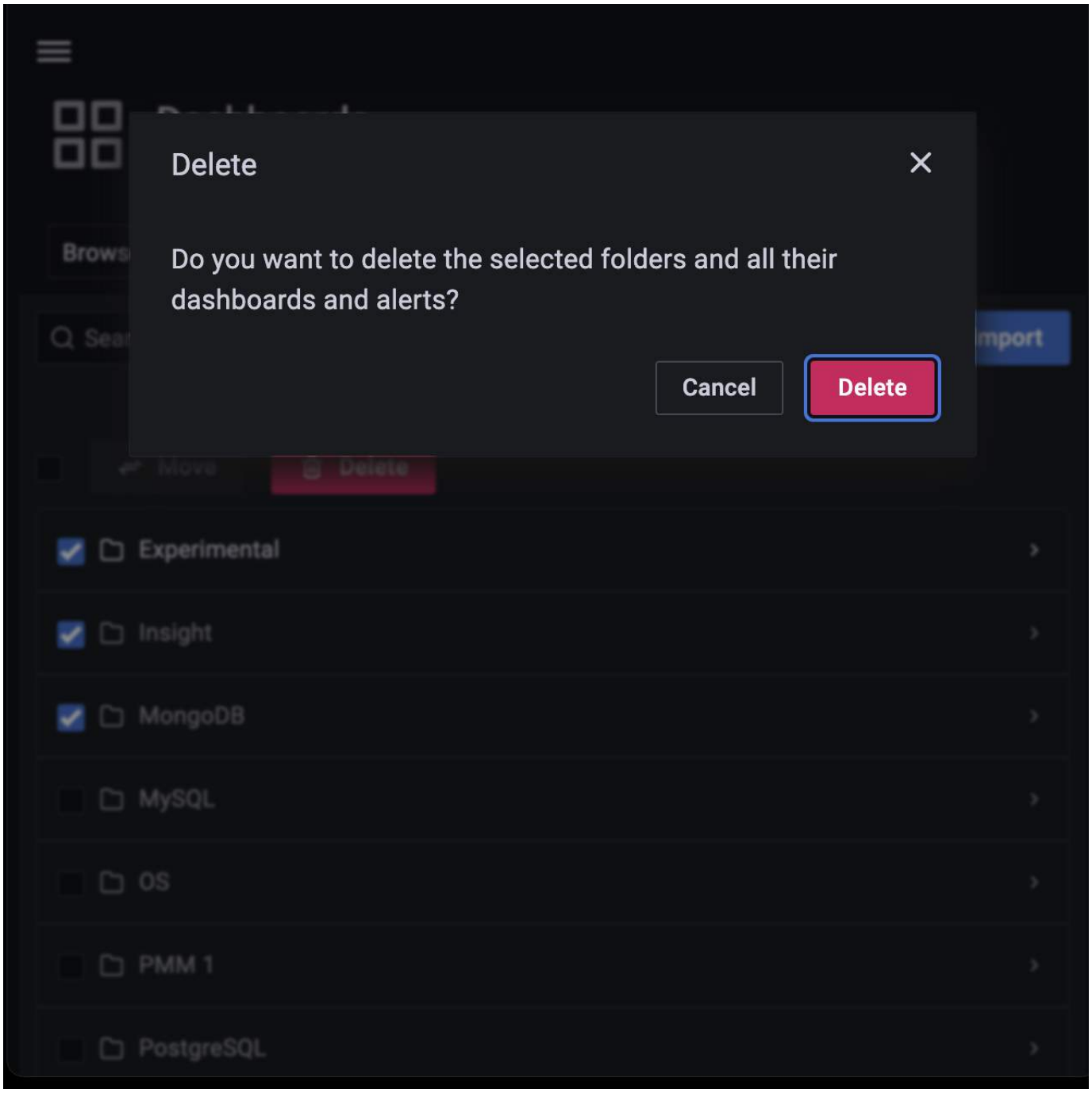
This section describes how to delete multiple dashboards, move dashboards from one folder to another and navigate to a folder page where you can assign folder and dashboard permissions.

Delete multiple dashboards

To delete multiple dashboards at once:

From the side menu, go to *Dashboards* > *Browse* and check the dashboards that you want to delete, and click *Delete*.

Percona LLC and/or its affiliates ©2024

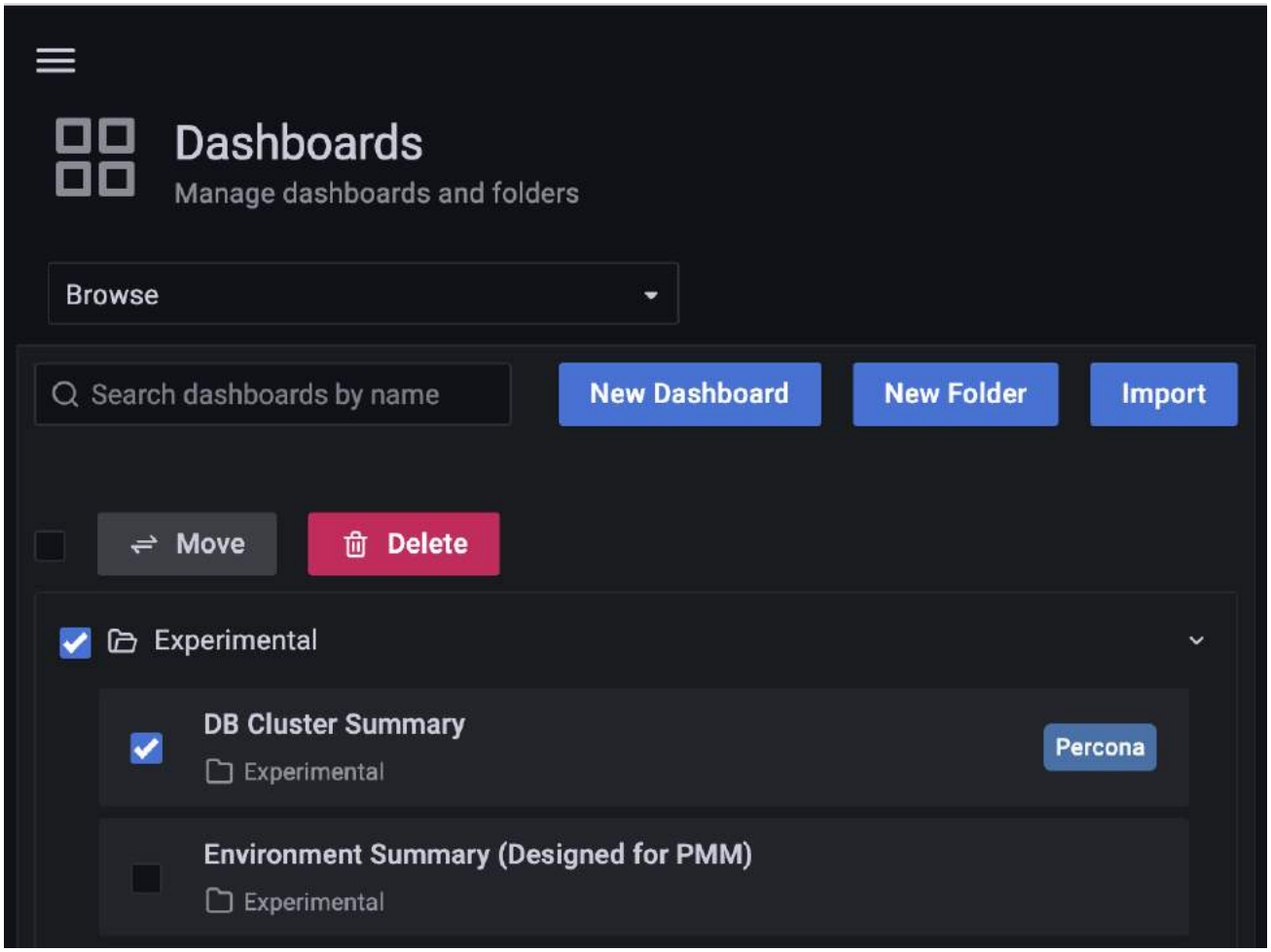


Move dashboards from one folder to another

You can move dashboards from one folder to another in the following two ways:

1. From the side menu, go to *Dashboards* > *Browse* and check the dashboards that you want to move. Click *Move*.

Percona LLC and/or its affiliates ©2024

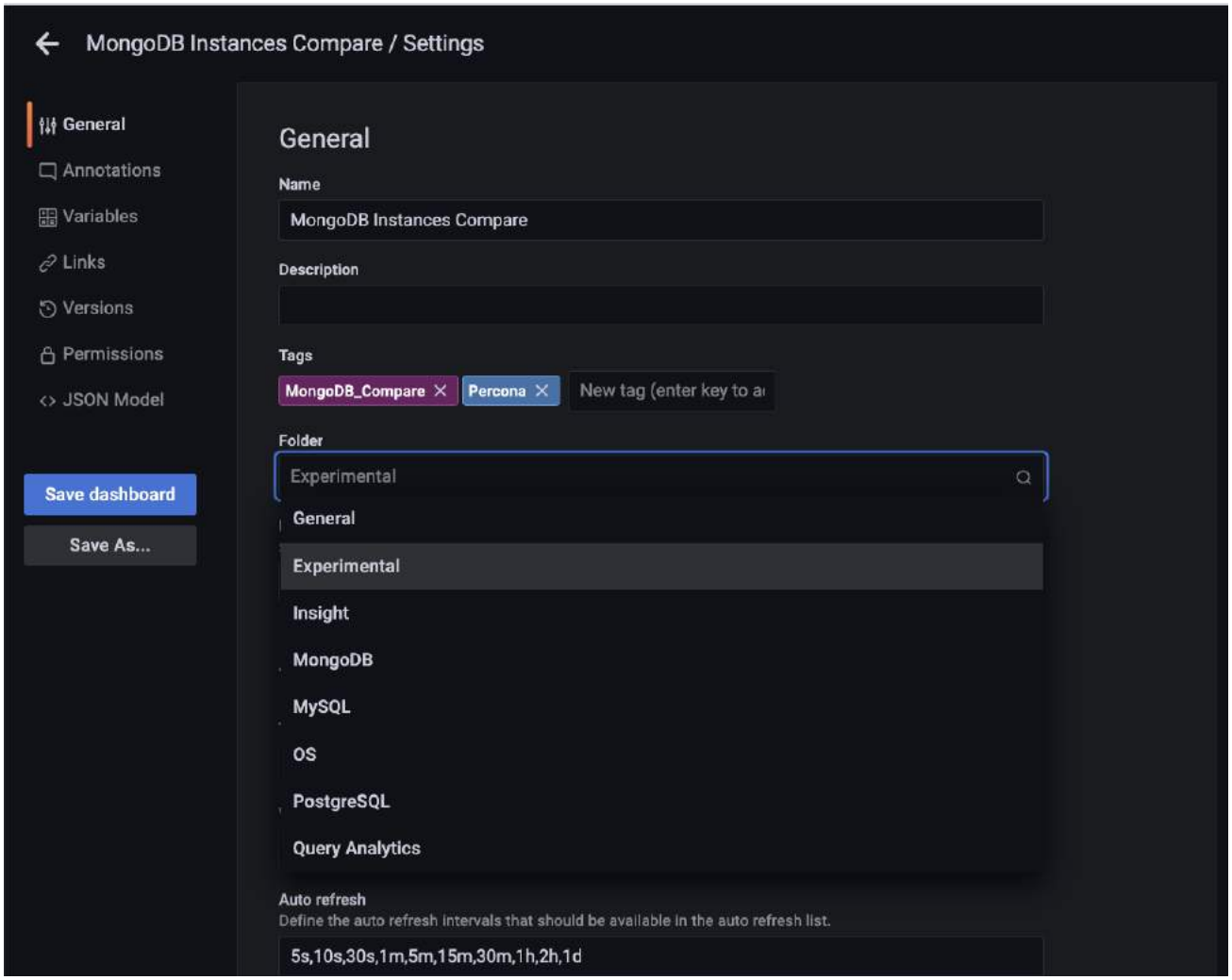


2. On the *Choose Dashboard Folder* dialog box select the dashboards that you want to move from the drop-down. Click *Move*.

The other way of moving dashboards from one folder to another is:

1. Open the dashboard that you want to move to another folder.
2. Click on icon to open *Dashboard Settings*.
3. On the *General* page, under *Folder* select the folder name that you want to move from the dropdown.

Percona LLC and/or its affiliates ©2024



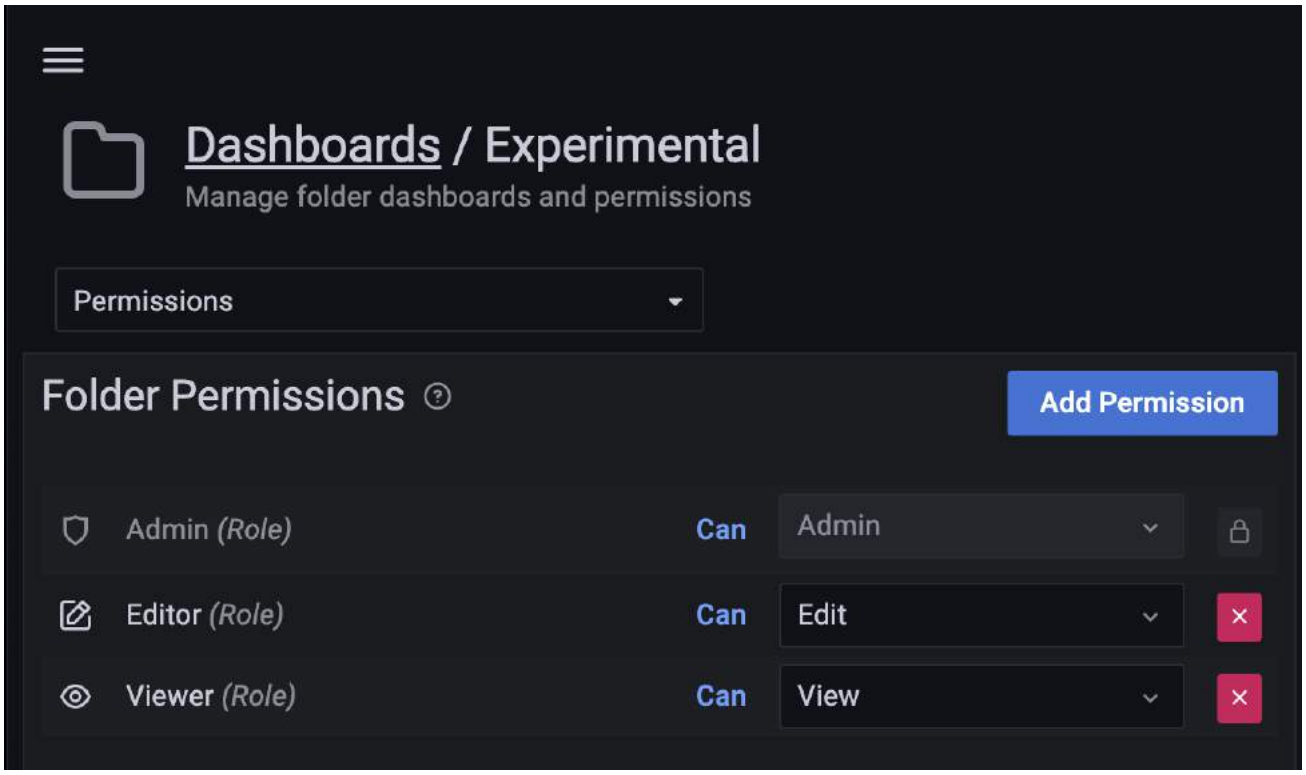
4. Click *Save Dashboard* on the the left to save the change.

> **Note**
>
> You should have atleast an *Editor* role to move a dashboard.

Navigate to a dashboard folder page to assign permissions

1. From the side menu, go to *Dashboards > Browse* and hover over the dashboard folder whose permissions you want to set. Click *Go to Folder*.
2. Go to the *Permissions* tab and select the requisite permission from the drop-down for the various roles.

Percona LLC and/or its affiliates ©2024



SETTING CUSTOM HOME DASHBOARD

The home dashboard you set is the dashboard all the users will see after logging in to PMM UI. You can set the home dashboard for a server, an organization, a team, or your user account.

Set home dashboard for your organization

Organization Admins can set the home dashboard for their organization. For information on managing users in an organization, see Manage Users

1. Navigate to the dashboard that you want to set as the home dashboard.
2. Click the star next to the dashboard title to mark the dashboard as a favorite.
3. Hover your cursor over *Configuration*
4. Click *Preferences*.
5. In the Home Dashboard field, select the dashboard that you want to set as your home dashboard.
6. Click *Save*.

Set home dashboard for your team

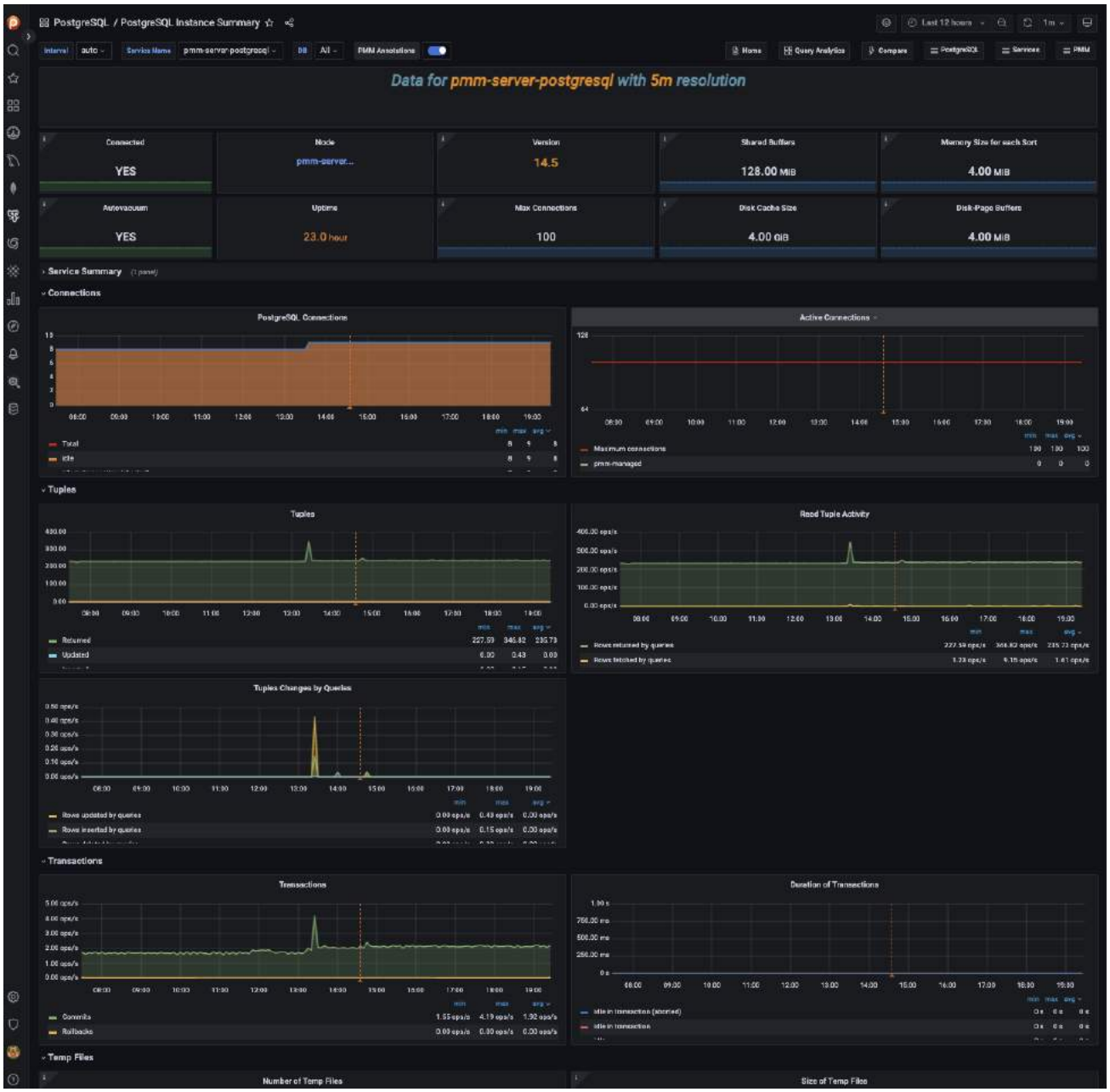
Organization and team Admins can set the home dashboard for their team as follows:

1. Navigate to the dashboard that you want to set as your home dashboard.
2. Click star next to the dashboard to mark the dashboard as a favorite.
3. On the main menu, hover your cursor over *Configuration*.
4. Click *Teams*. Grafana displays the team list.
5. Click on the team for whom you want to set the home dashboard and then navigate to the *Settings* tab.
6. In the Home Dashboard field, select the dashboard that you want to use for your home dashboard.
7. Click *Save*.

Percona LLC and/or its affiliates ©2024

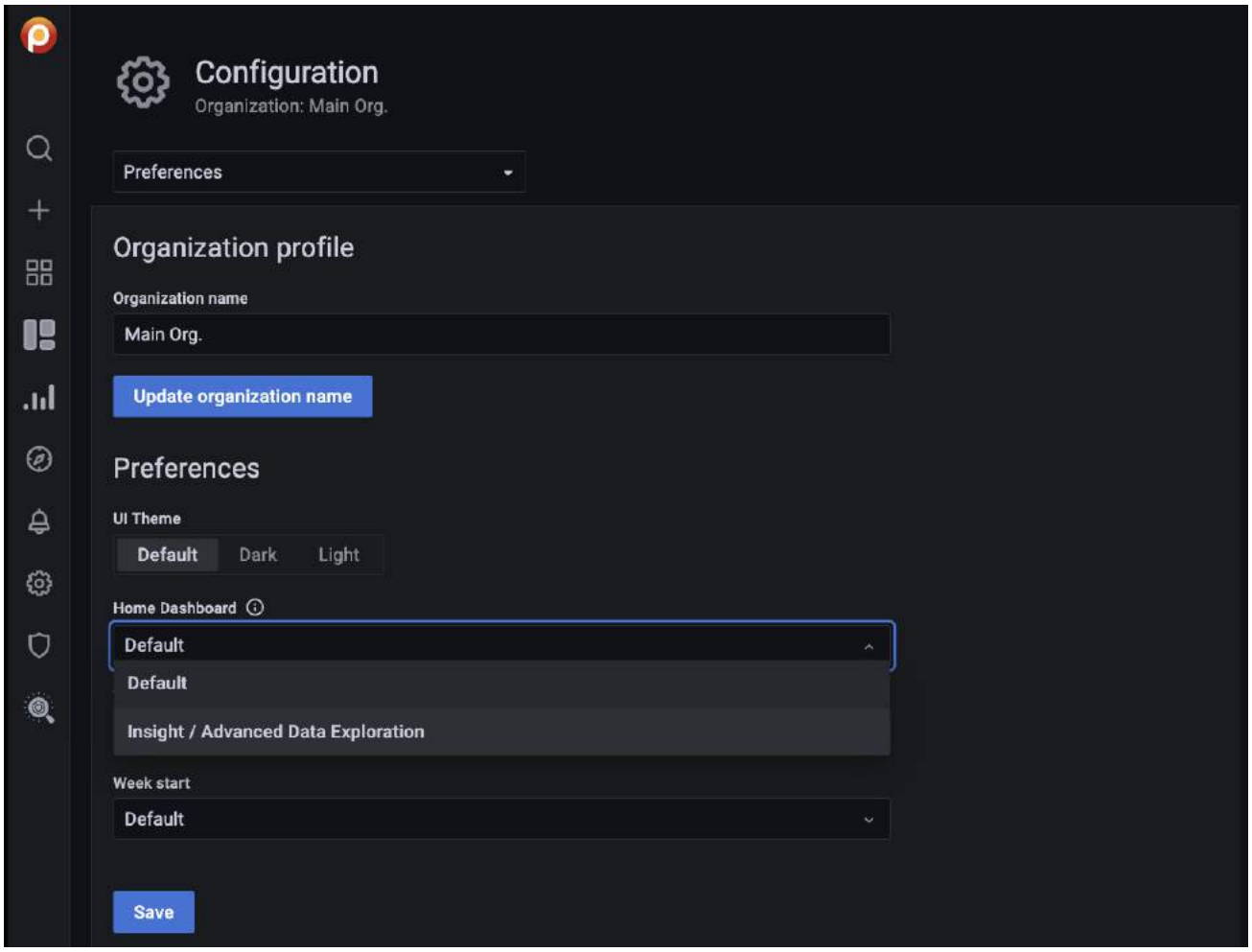
Set your Personal Home Dashboard

1. From the main menu, go to *Dashboards* > *Browse* and select the dashboard you want to set as your home dashboard.
2. Click the star next to the dashboard title to mark it as a favorite.



3. From the side menu go to *Configuration* > *Preferences*. In the *Home Dashboard* field, select the dashboard that you want to set as your home dashboard.

Percona LLC and/or its affiliates ©2024



4. Click *Save*.

GET EXPERT HELP

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

6.8.3 PMM dashboards

Insight

HOME DASHBOARD

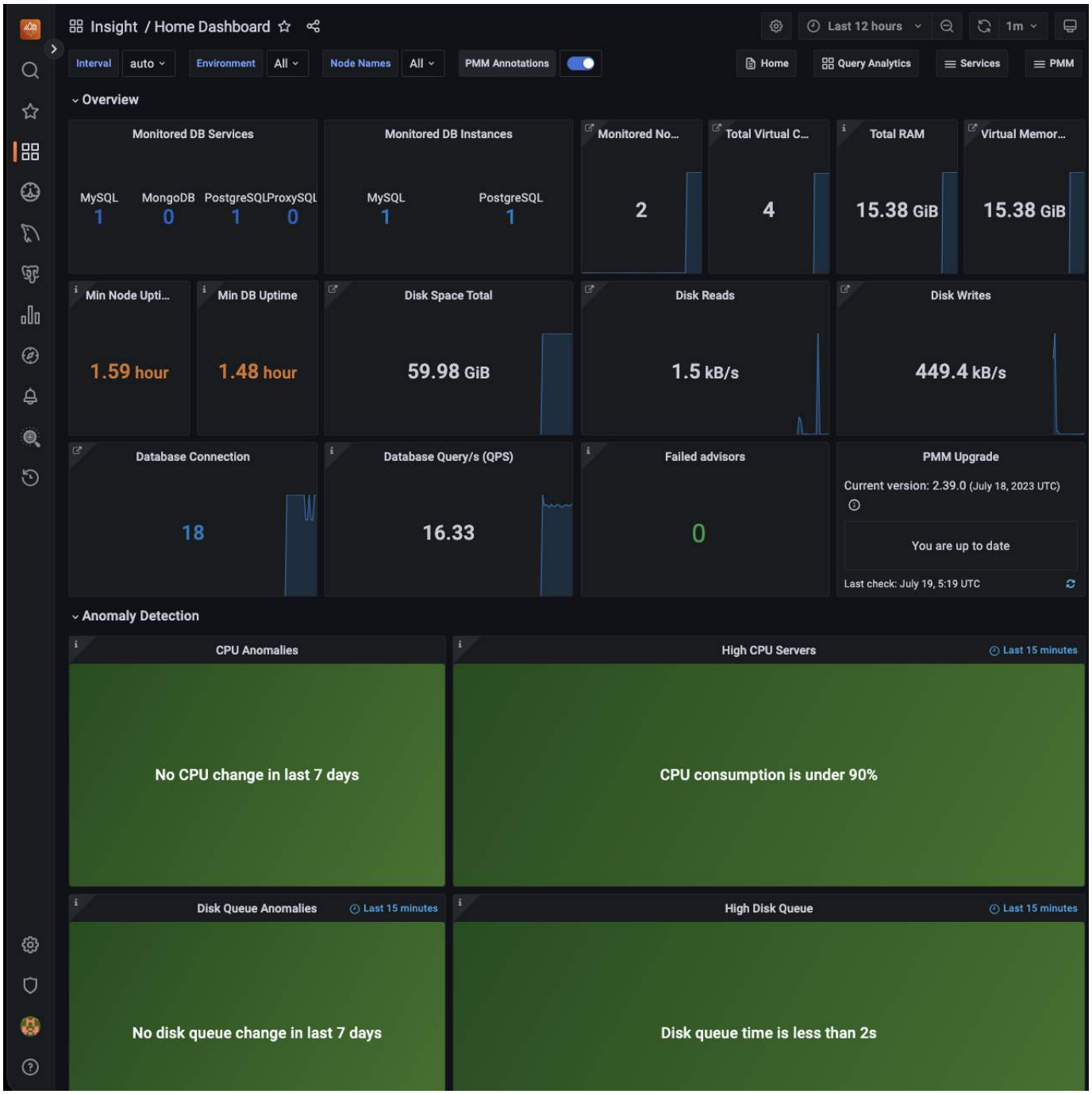
Availability

The new Home dashboard is available starting with PMM 2.32.0.

Percona LLC and/or its affiliates ©2024

The Home Dashboard provides a high-level overview of your environment, such as the services, infrastructure, and critical issues (if any). It is the starting page of PMM from which you can open the tools of PMM and browse online resources.

This Home Dashboard displays data that is organized in panels as given below.



Percona LLC and/or its affiliates ©2024

Overview

This panel lists all added hosts along with essential information about their performance. For each host, you can find the current values of the following metrics:

- Monitored DB Services
- Monitored DB Instances
- Monitored Nodes
- Memory Available
- Disk Reads
- Disk Writes
- Network IO
- DB Connections
- DB QPS
- Virtual CPUs
- RAM
- Host Uptime
- DB Uptime
- Advisors check

This panel also displays the current version number. Use **Upgrade to X.X.X version** to upgrade to the most recent version of PMM.

Anomaly Detection

The **Anomaly Detection** panel lists all the anomalies in your environment. Color-coded states on the panels provide a quick visual representation of the problem areas.

The following anomalies are displayed on this panel:

- CPU anomalies (high as well as low)
- High CPU servers
- Low CPU servers
- Disk Queue anomalies
- High disk queue
- High Memory Used

Command Center

You can find critical information such as CPU utlization, memory utilization, anomalies, read and write latency, etc., about your environment on the **Command Center** panel.

The information is represented graphically on the **Command Center** panel. In this panel, the graphs for the last hour and the previous week are displayed adjacently, making it easy to identify the trends.

Percona LLC and/or its affiliates ©2024

The following information is displayed on the **Command Center** for the **Top 20** nodes:

- CPU usage
- Disk queue
- Disk Write latency
- Disk Read latency
- Memory usage

Command Center lists the

Service Summary

The Service Summary panel provides the following information for the services being monitored:

- DB connections
- DB QPS (Query per sec)
- DB uptime

Get expert help

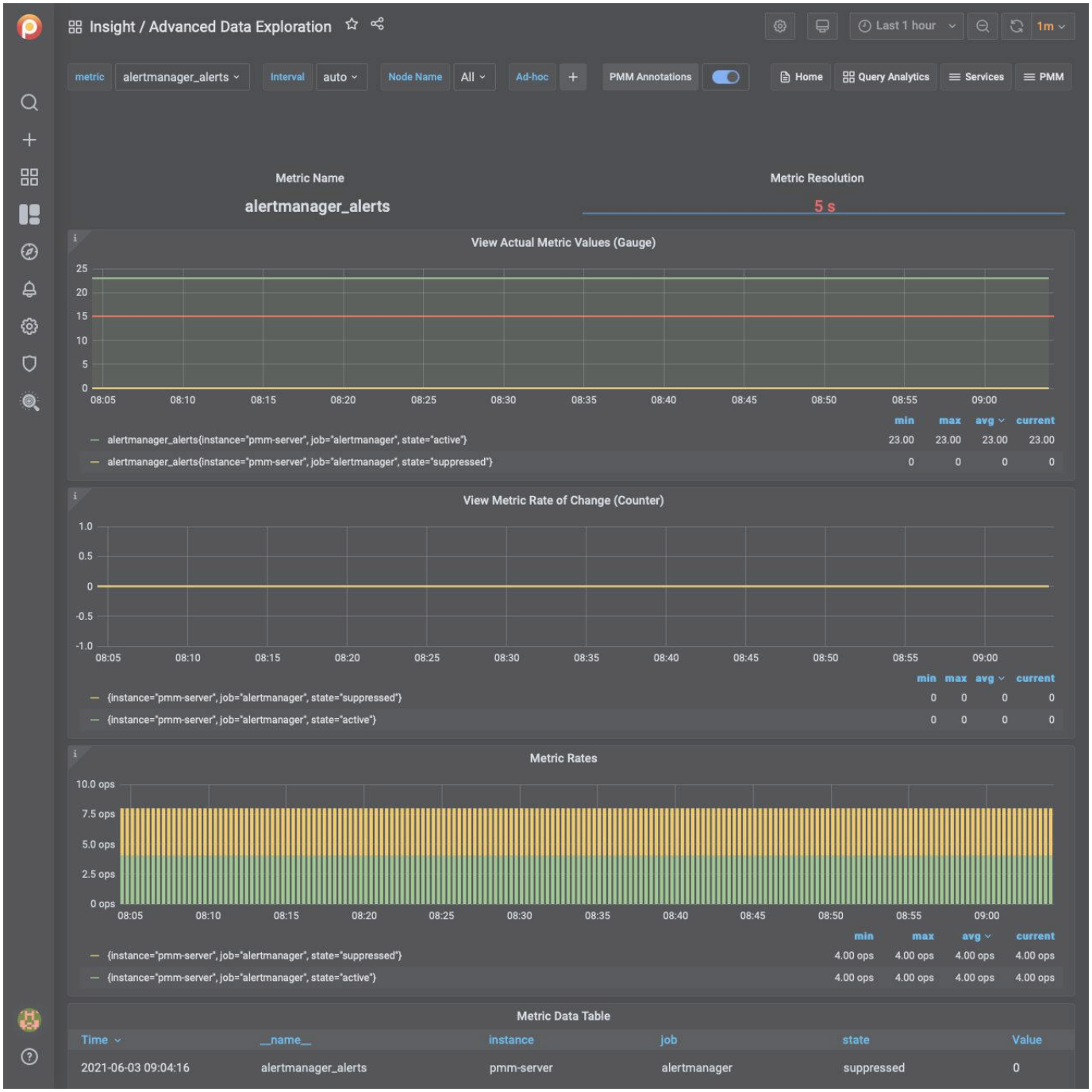
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-08

Percona LLC and/or its affiliates ©2024

ADVANCED DATA EXPLORATION



The *Advanced Data Exploration* dashboard provides detailed information about the progress of a single Prometheus metric across one or more hosts.

View actual metric values (Gauge)

A gauge is a metric that represents a single numerical value that can arbitrarily go up and down.

Gauges are typically used for measured values like temperatures or current memory usage, but also “counts” that can go up and down, like the number of running goroutines.

View Metric Rate of Change (Counter)

A counter is a cumulative metric that represents a single numerical value that only ever goes up. A counter is typically used to count requests served, tasks completed, errors occurred, etc. Counters should not be used to expose

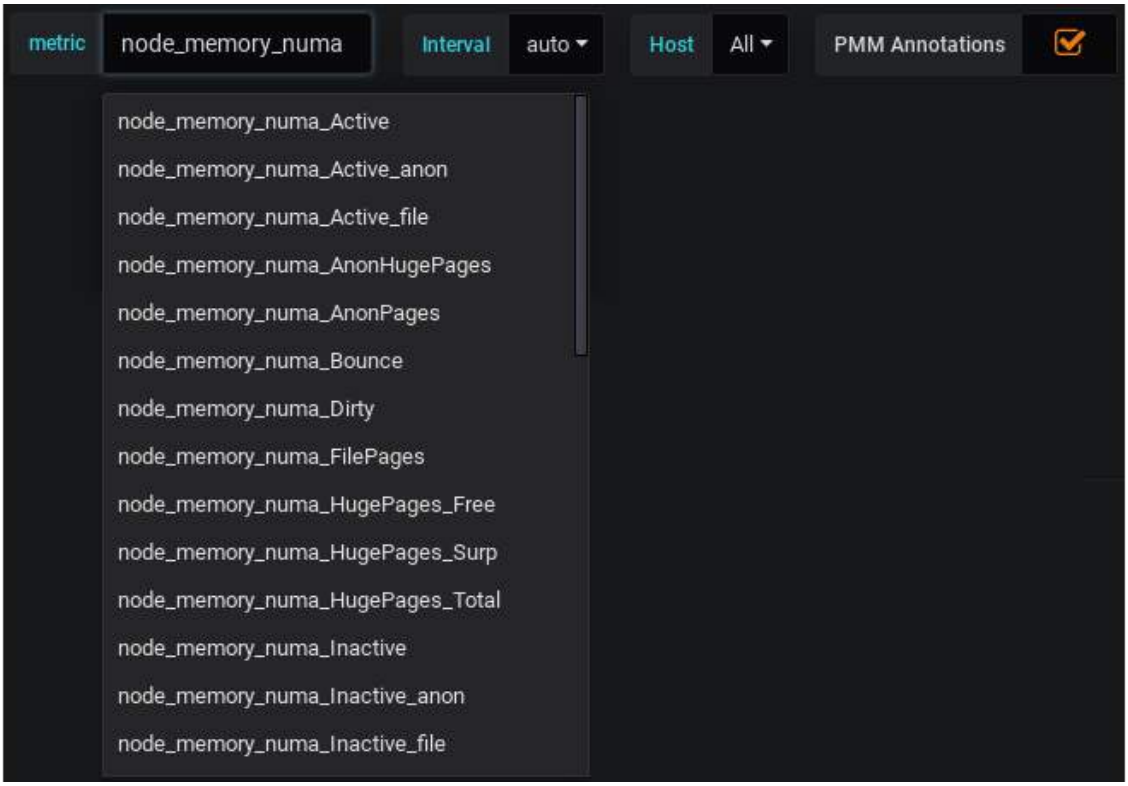
Percona LLC and/or its affiliates ©2024

current counts of items whose number can also go down, e.g. the number of currently running goroutines. Use gauges for this use case.

Metric Rates

Shows the number of samples Per second stored for a given interval in the time series.

This dashboard supports metrics related to NUMA. The names of all these metrics start with `node_memory_numa`.



Get expert help

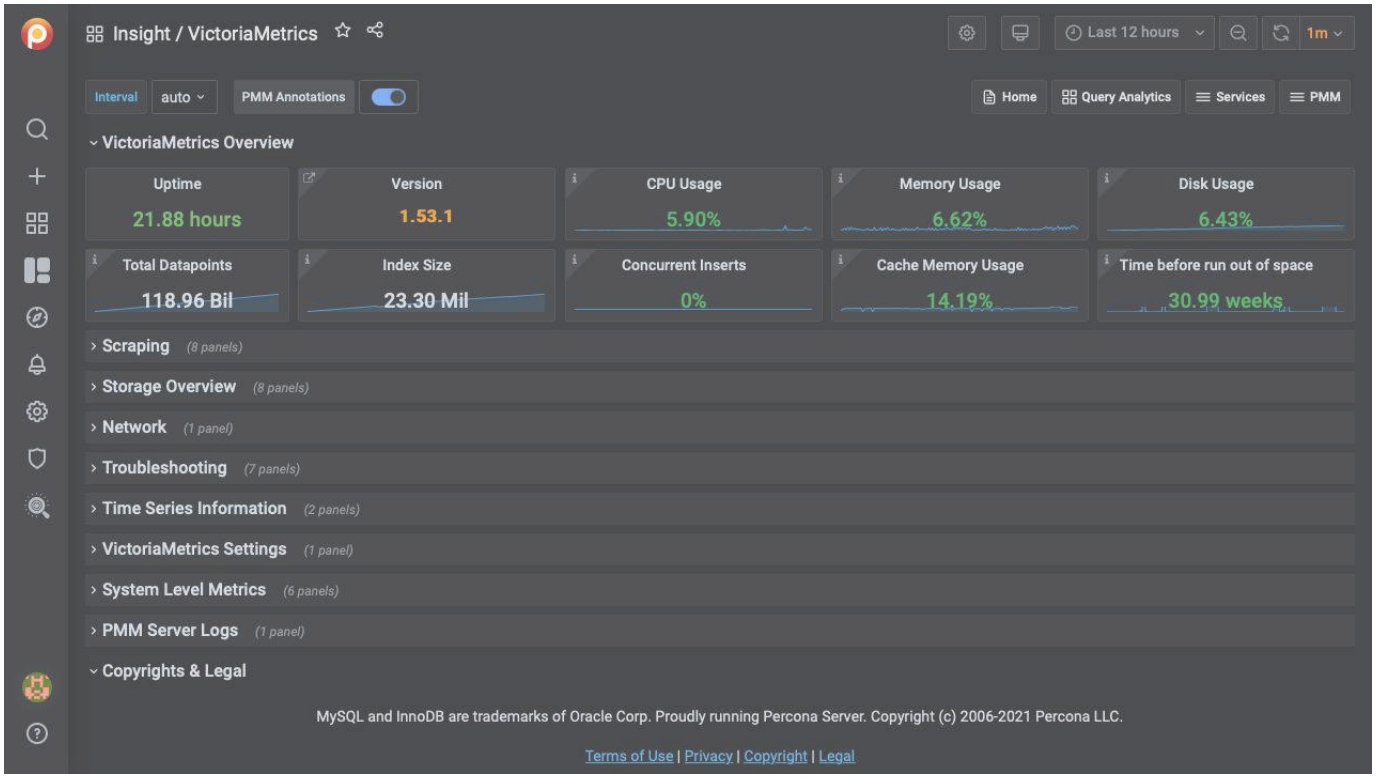
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-03

Percona LLC and/or its affiliates ©2024

VICTORIAMETRICS



No description

Get expert help

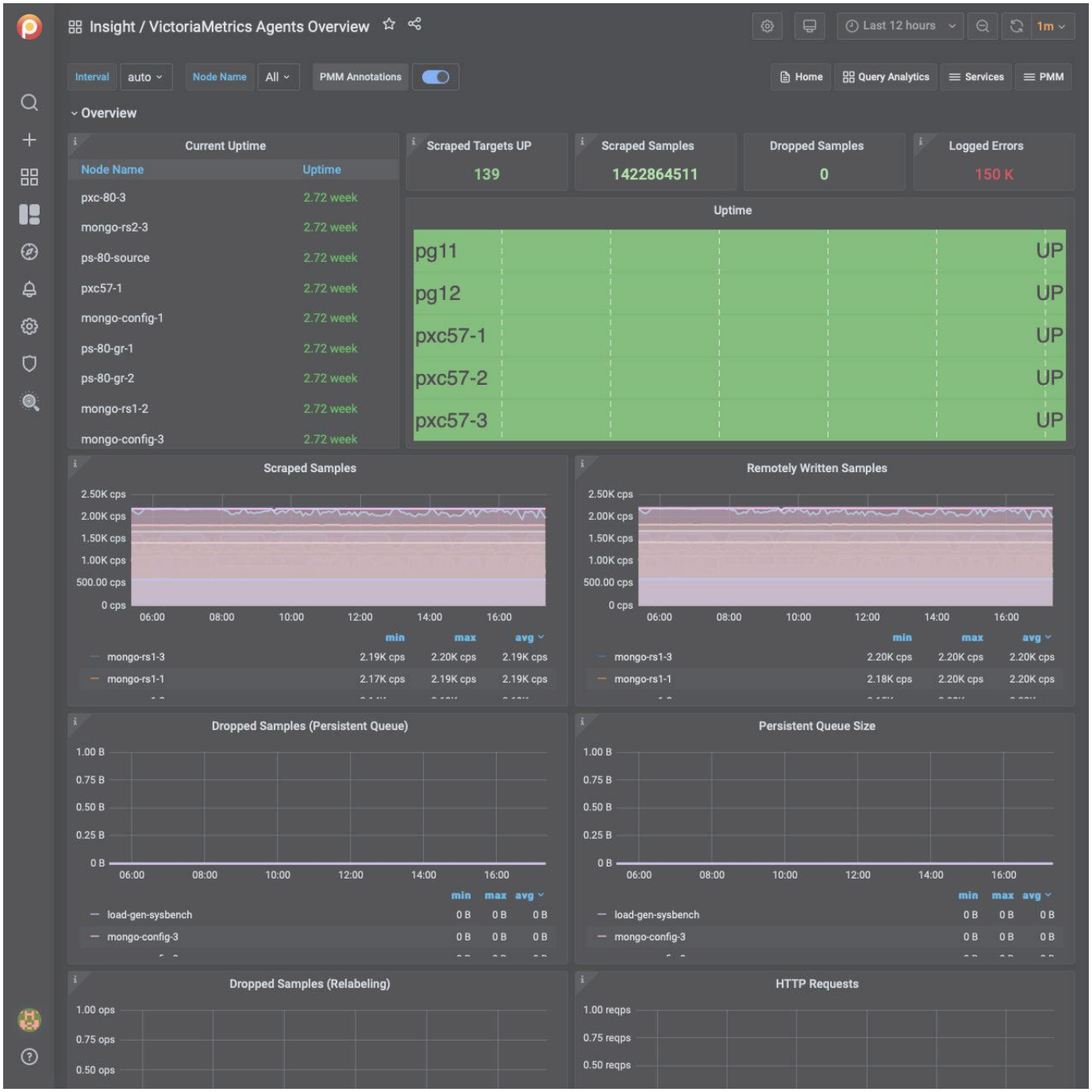
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

VICTORIAMETRICS AGENTS OVERVIEW



No description

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

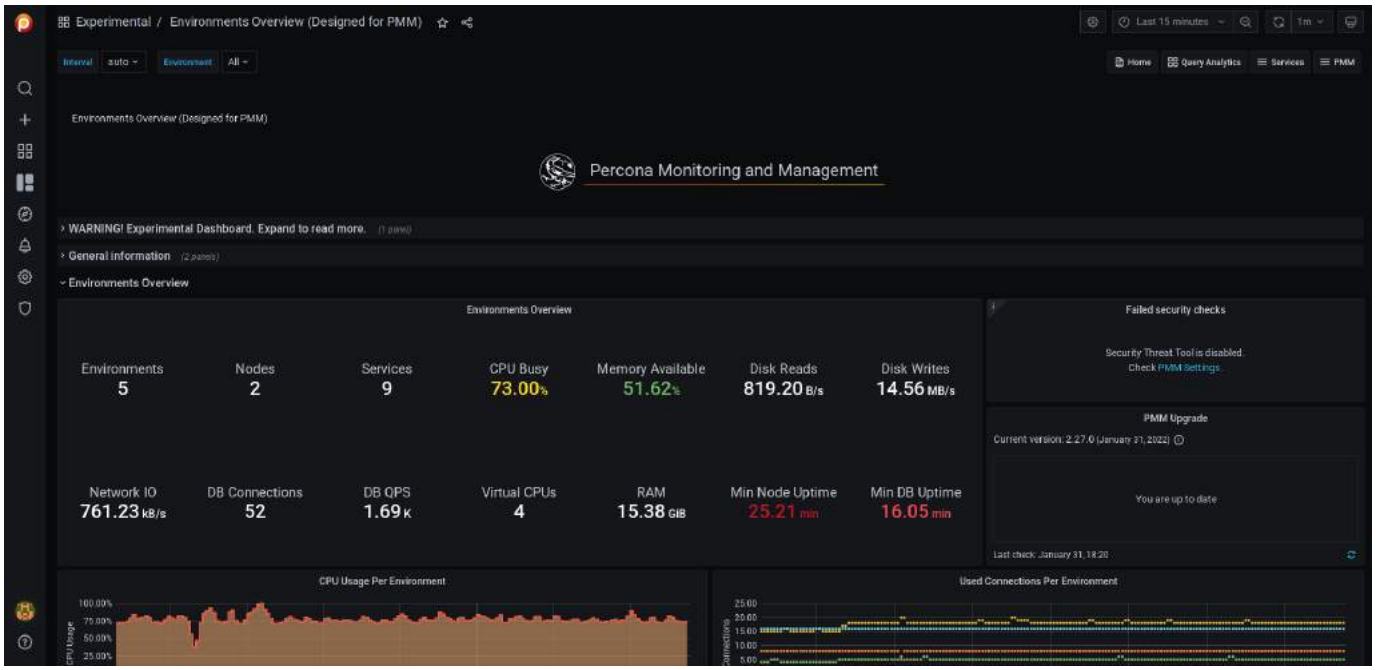
Percona LLC and/or its affiliates ©2024

2021-05-11

Environments

ENVIRONMENT OVERVIEW

> **Disclaimer**
>
> This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.



The Dashboard provides the user with a high-level view of all the environments in PMM.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

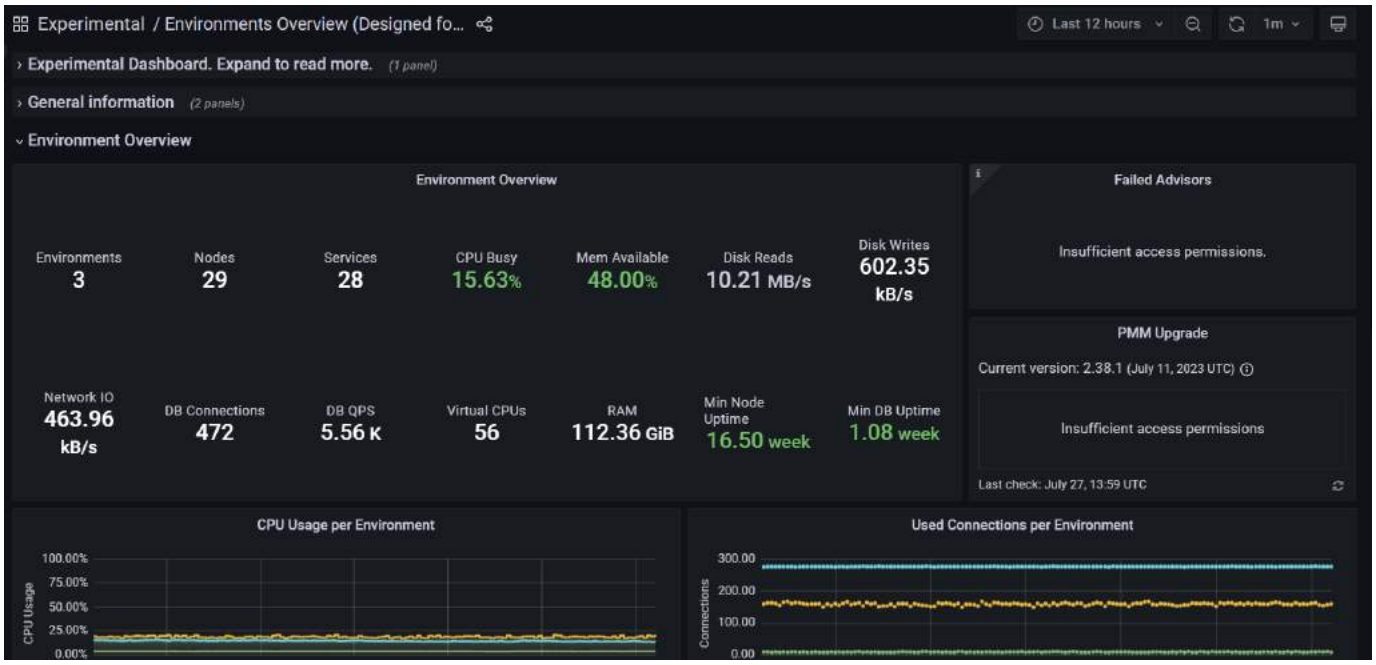
Community Forum Get a Percona Expert

2022-02-08

ENVIRONMENT SUMMARY

> **Disclaimer**
>
> This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.

Percona LLC and/or its affiliates ©2024



The Environment Summary Dashboard provides an at-a-glance view specific to the selected environment in PMM, including an overview of the services and nodes running in that environment.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-02-08

Kubernetes

KUBERNETES CLUSTER SUMMARY

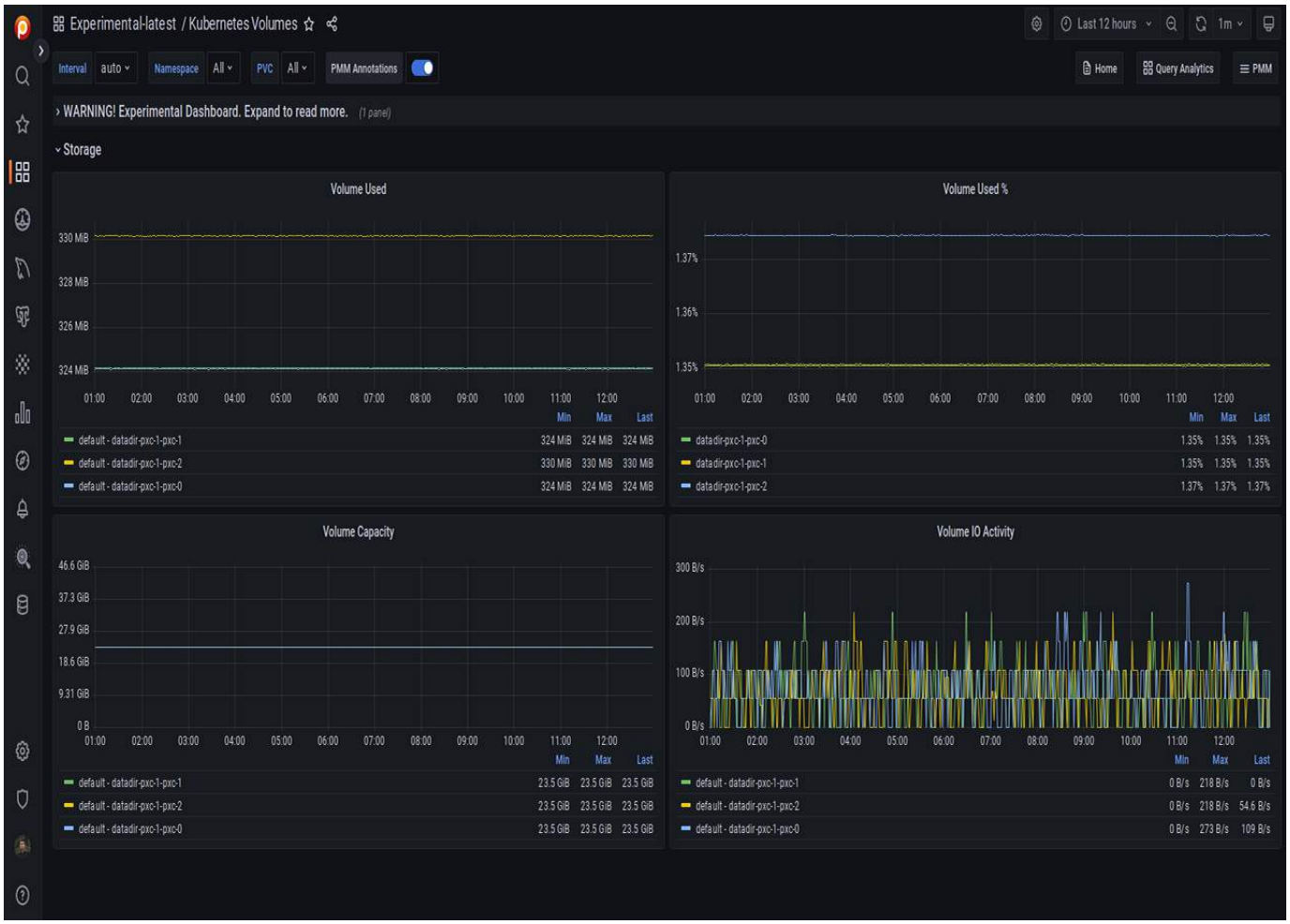
Disclaimer

This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.

Availability

This experimental dashboard is available starting with PMM 2.30.0.

Percona LLC and/or its affiliates ©2024



Kubernetes Volumes dashboard provides a comprehensive overview of your Kubernetes cluster, including:

- Resources
- Node Status
- Pod Status
- PVC status
- CPU Overview
- Kubernetes Resource Count
- Memory Overview and more

With this dashboard, you can view all workloads running in the cluster and optimize their performance.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-05-02

Percona LLC and/or its affiliates ©2024

KUBERNETES PODS STATUS

Disclaimer

This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.

Availability

This experimental dashboard is available starting with PMM 2.37.0.

Percona LLC and/or its affiliates ©2024

Kubernetes Pods Status dashboard provides detailed information about the state and performance of your pods, including CPU, Memory, and Network metrics.

This dashboard can help you quickly pinpoint any issues affecting your pods and ensure they continue to operate smoothly.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-05-02

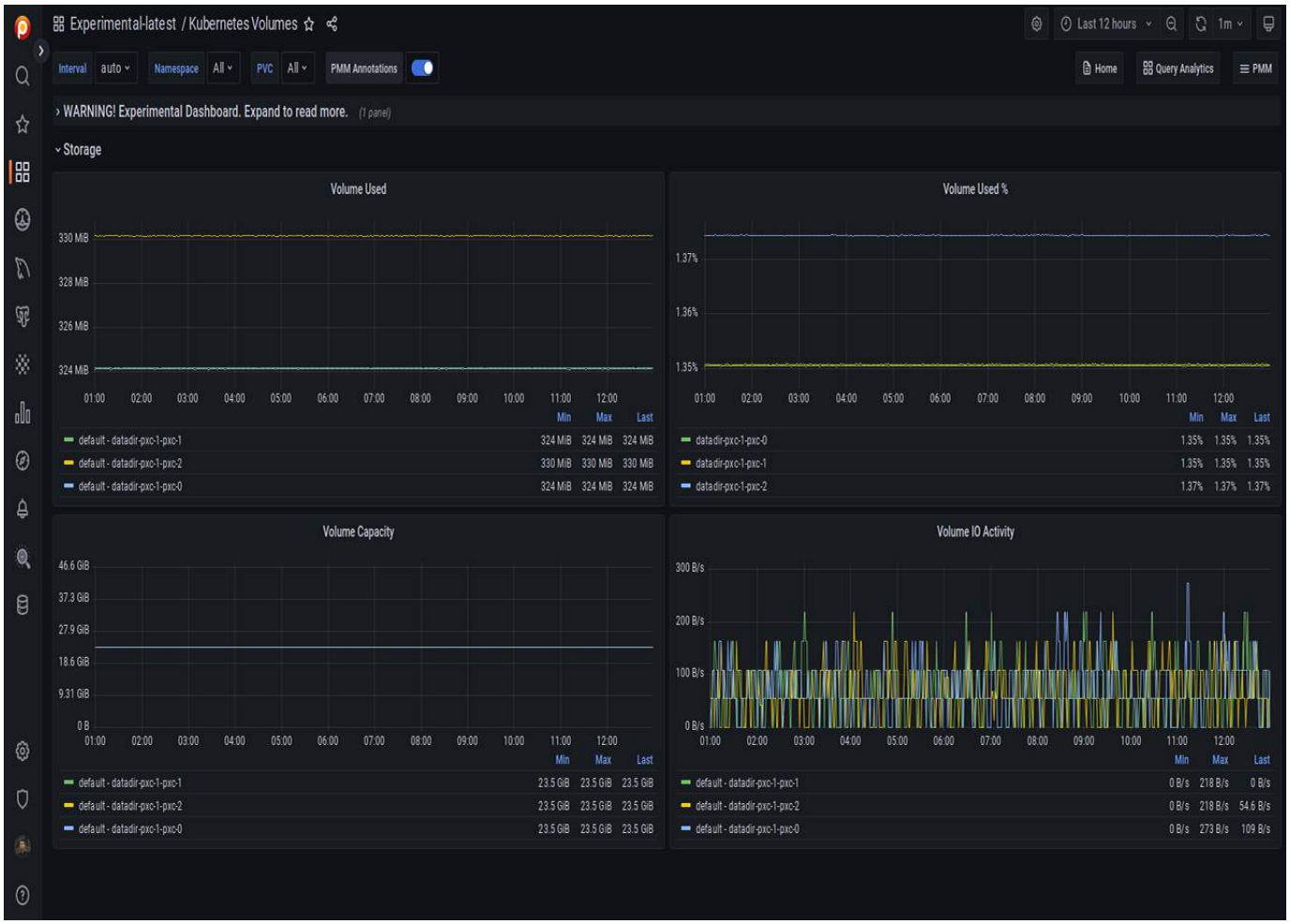
KUBERNETES VOLUMES

Disclaimer

This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.

This dashboard will not work for minikube cluster as there are no drivers for it.

Percona LLC and/or its affiliates ©2024



Starting with PMM 2.37.0 we have introduced *Kubernetes Volumes* dashboard. This dashboard provides insights into your Kubernetes volumes, including capacity and usage, in real-time.

With this dashboard, you can easily monitor the performance and usage of your volumes and take proactive measures to ensure their performance.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-05-02

KUBERNETES MONITORING FOR PERCONA OPERATORS

Important

This feature is still in Technical Preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

Percona LLC and/or its affiliates ©2024

Monitoring the state of the database is crucial to timely identify and react to performance issues. Percona Monitoring and Management (PMM) solution enables you to do just that.

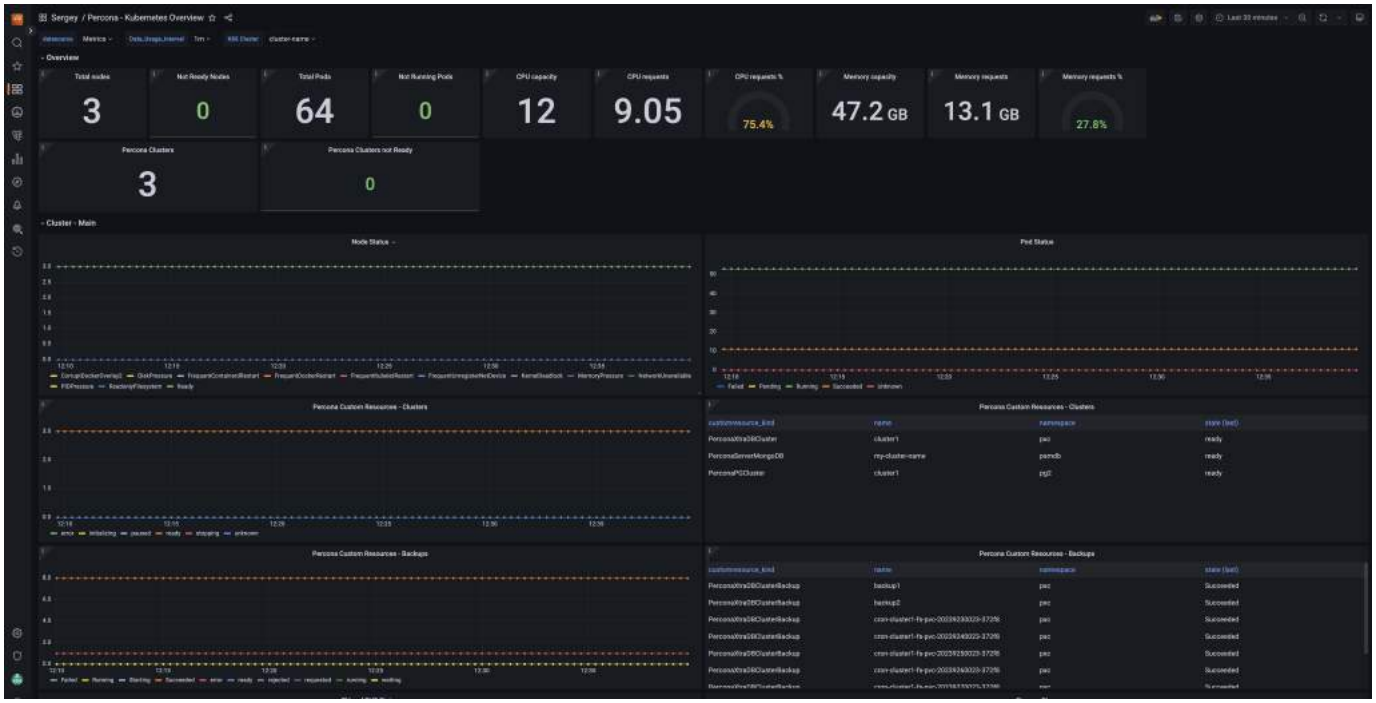
However, the database state also depends on the state of the Kubernetes cluster itself. Hence it's important to have metrics that can depict the state of the Kubernetes cluster.

For inforamtion on setting up monitoring for the Kubernetes cluster health, see documentation.

This setup has been tested with the PMM server as the centralized data storage and the Victoria Metrics Kubernetes monitoring stack as the metrics collector. These steps may also apply if you use another Prometheus-compatible storage.

Kubernetes overview

The Kubernetes Cluster overview dashboard gives you an overview of Kubernetes health and its objects, including Percona custom resources.



Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-10-05

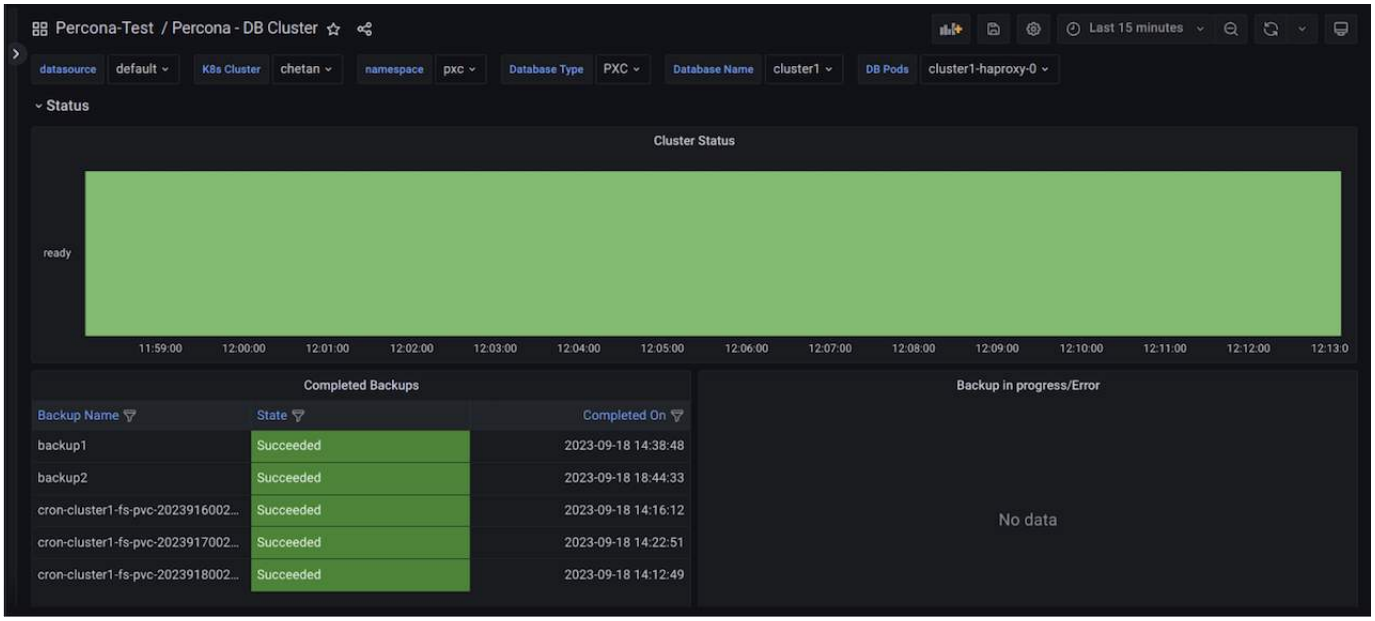
DB CLUSTERS MANAGED WITH PERCONA KUBERNETES OPERATORS

Important

This feature is still in Technical Preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

Percona LLC and/or its affiliates ©2024

This dashboard displays the primary parameters of database clusters created by Percona Operators for various databases and helps identify the performance issues.



Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-10-05

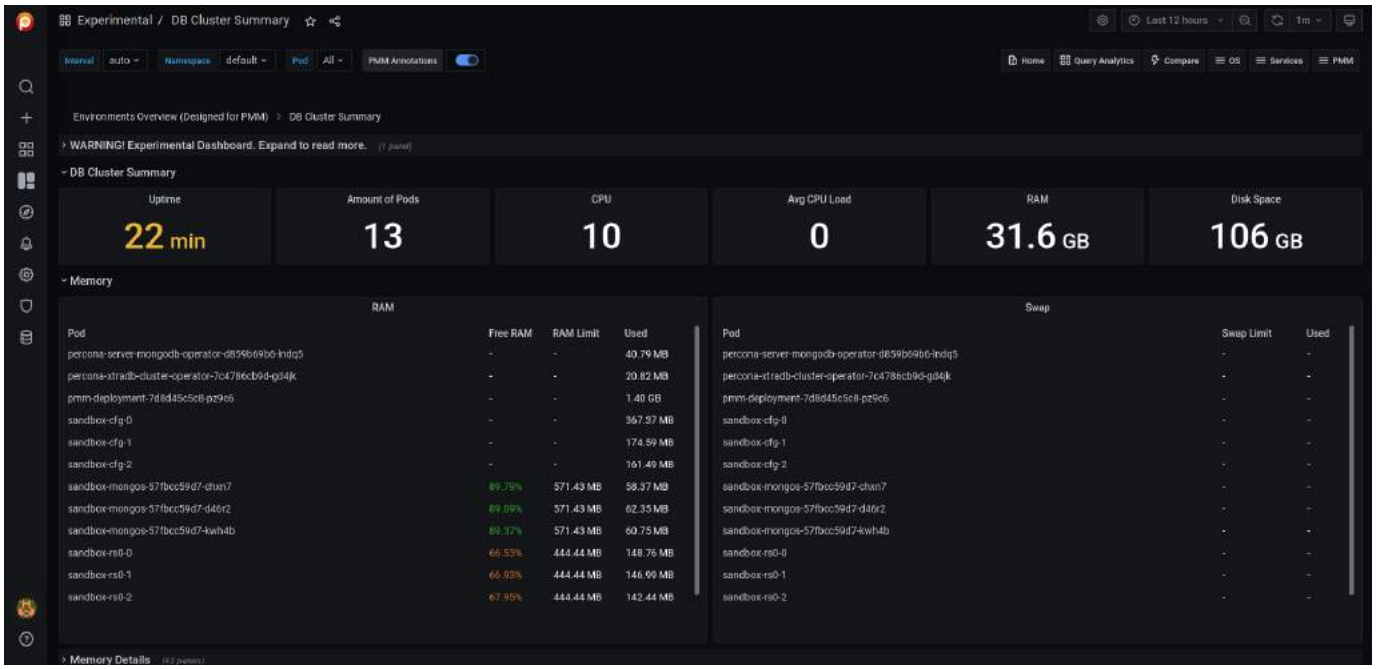
DBaas

DB CLUSTER SUMMARY

Disclaimer

This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.

Percona LLC and/or its affiliates ©2024



This Dashboard is a part of DBaaS solution inside PMM.

This Dashboard is designed to show the resource consumption inside Kubernetes (K8s) Cluster.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

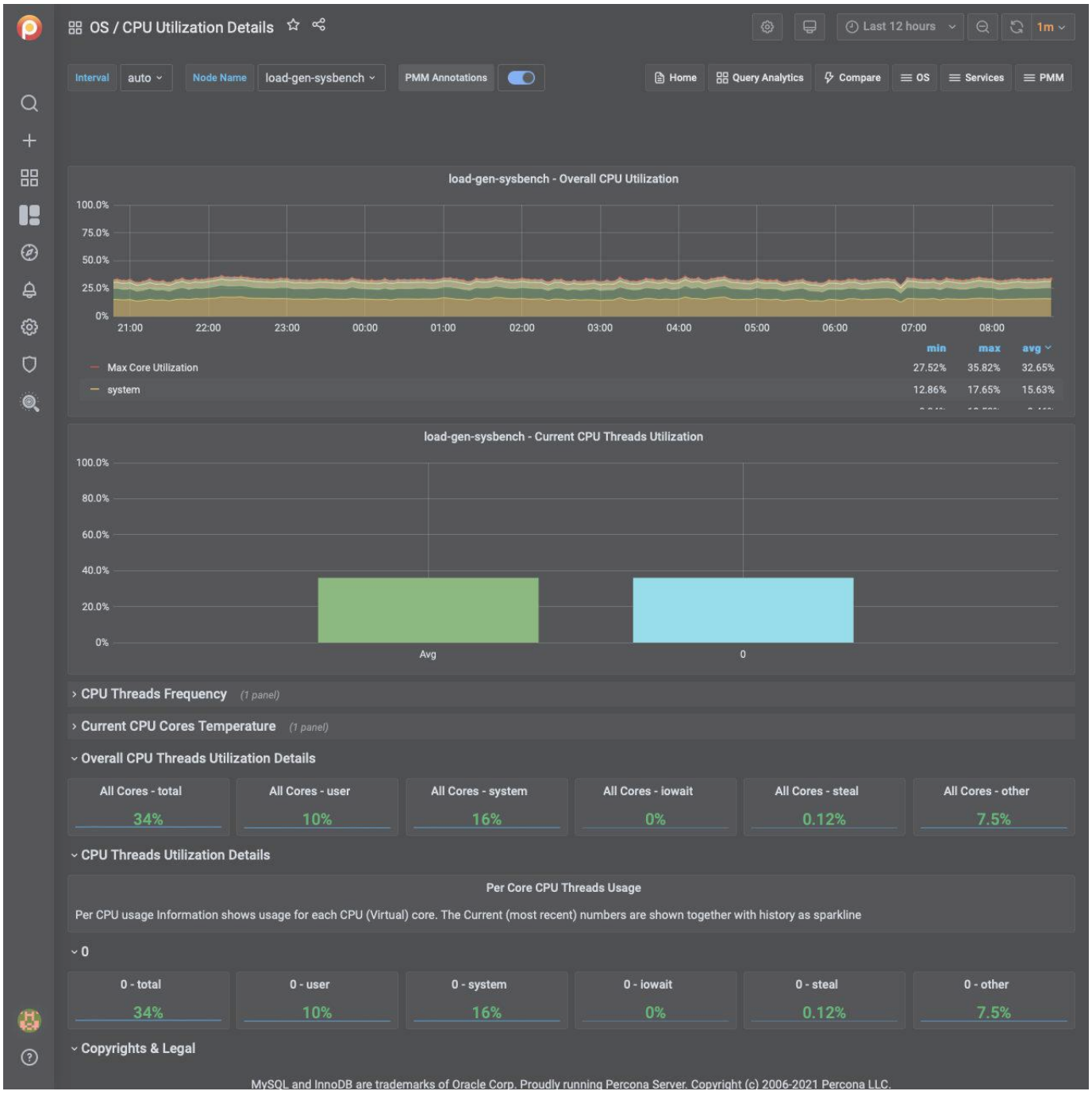
Community Forum Get a Percona Expert

2022-12-19

Percona LLC and/or its affiliates ©2024

OS Dashboards

CPU UTILIZATION DETAILS



Overall CPU Utilization

The Overall CPU Utilization metric shows how much of the overall CPU time is used by the server. It has these components:

Max Core Utilization

No description

Percona LLC and/or its affiliates ©2024

System

This component the proportion of time the CPUs spent inside the Linux kernel for operations like context switching, memory allocation and queue handling.

User

This component is the time spent in the user space. Normally, most of the MySQL CPU time is in user space. A high value of user time indicates a CPU bound workload.

Softirq

This component is the portion of time the CPU spent servicing software interrupts generated by the device drivers. A high value of softirq may indicates a poorly configured device. The network devices are generally the main source of high softirq values.

Steal

When multiple virtual machines share the same physical host, some virtual machines may be allowed to use more of their share of CPU and that CPU time is accounted as Steal by the virtual machine from which the time is taken.

Iowait

This component is the time the CPU spent waiting for disk IO requests to complete. A high value of iowait indicates a disk bound load.

Nice

No description

In addition, sampling of the Max utilization of a single core is shown.

This metric presents global values: while there may be a lot of unused CPU, a single core may be saturated. Look at the Max Core Utilization to see if any core is reaching close to 100%.

Current CPU Threads Utilization

This shows the total utilization of each CPU core along with the average utilization of all CPU cores. Watch for any core close to 100% utilization and investigate the root cause.

CPU Threads Frequency

No description

Current CPU Cores Temperature

No description

Overall CPU Threads Utilization Details

No description

Get expert help

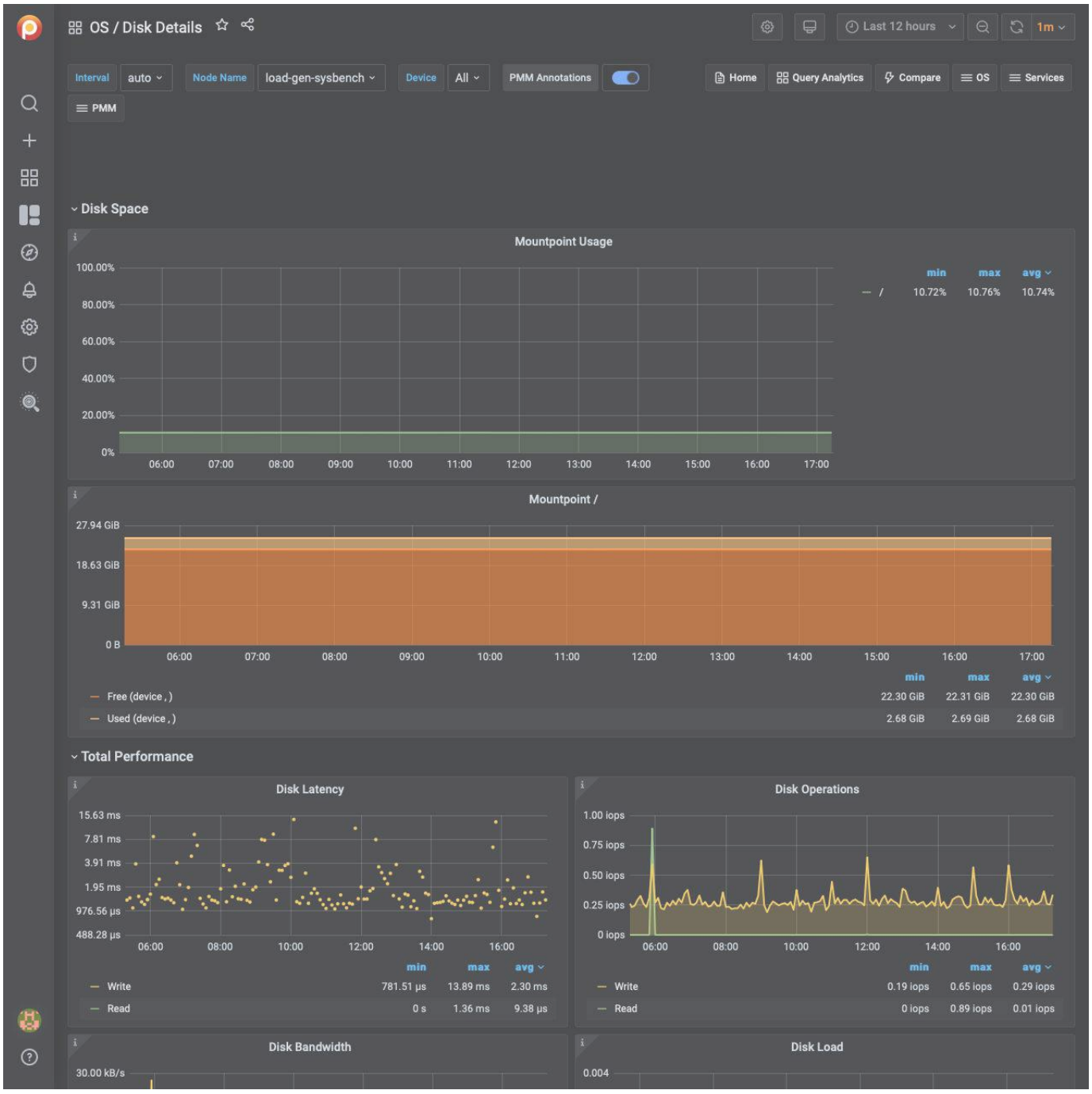
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2021-06-03

DISK DETAILS



mount point Usage

Shows the percentage of disk space utilization for every mount point defined on the system. Having some of the mount points close to 100% space utilization is not good because of the risk of a “disk full” error that can block one of the services or even cause a crash of the entire system.

In cases where the mount point is close to 100% consider removing unused files or expanding the space allocated to the mount point.

Percona LLC and/or its affiliates ©2024

mount point

Shows information about the disk space usage of the specified mount point.

Used is the amount of space used.

Free is the amount of space not in use.

Used+Free is the total disk space allocated to the mount point.

Having *Free* close to 0 B is not good because of the risk of a “disk full” error that can block one of the services or even cause a crash of the entire system.

In cases where Free is close to 0 B consider removing unused files or expanding the space allocated to the mount point.

Disk Latency

Shows average latency for Reads and Writes IO Devices. Higher than typical latency for highly loaded storage indicates saturation (overload) and is frequent cause of performance problems. Higher than normal latency also can indicate internal storage problems.

Disk Operations

Shows amount of physical IOs (reads and writes) different devices are serving. Spikes in number of IOs served often corresponds to performance problems due to IO subsystem overload.

Disk Bandwidth

Shows volume of reads and writes the storage is handling. This can be better measure of IO capacity usage for network attached and SSD storage as it is often bandwidth limited. Amount of data being written to the disk can be used to estimate Flash storage life time.

Disk Load

Shows how much disk was loaded for reads or writes as average number of outstanding requests at different period of time. High disk load is a good measure of actual storage utilization. Different storage types handle load differently - some will show latency increases on low loads others can handle higher load with no problems.

Disk IO Utilization

Shows disk Utilization as percent of the time when there was at least one IO request in flight. It is designed to match utilization available in iostat tool. It is not very good measure of true IO Capacity Utilization. Consider looking at IO latency and Disk Load Graphs instead.

Avg Disks Operations Merge Ratio

Shows how effectively Operating System is able to merge logical IO requests into physical requests. This is a good measure of the IO locality which can be used for workload characterization.

Disk IO Size

Shows average size of a single disk operation.

Get expert help

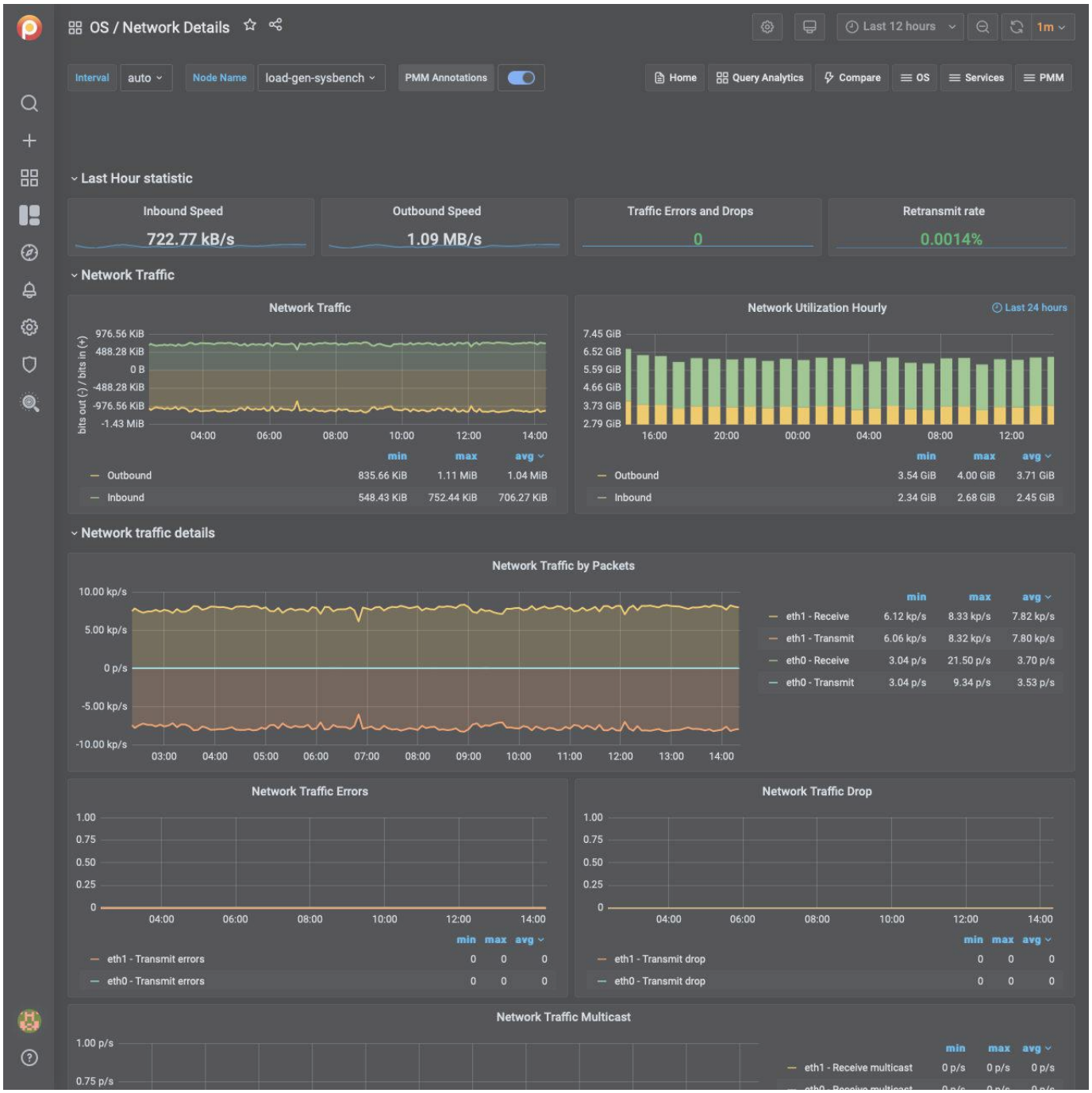
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2021-05-11

NETWORK DETAILS



Last Hour Statistic

This section reports the *inbound speed*, *outbound speed*, *traffic errors and drops*, and *retransmit rate*.

Network Traffic

This section contains the *Network traffic* and *network utilization hourly* metrics.

Percona LLC and/or its affiliates ©2024

Network Traffic Details

This section offers the following metrics:

- Network traffic by packets
- Network traffic errors
- Network traffic drop
- Network traffic multicast

Network Netstat TCP

This section offers the following metrics:

- Timeout value used for retransmitting
- Min TCP retransmission timeout
- Max TCP retransmission timeout
- Netstat: TCP
- TCP segments

Network Netstat UDP

In this section, you can find the following metrics:

- Netstat: UDP
- UDP Lite

The graphs in the *UDP Lite* metric give statistics about:

`InDatagrams`

: Packets received

`OutDatagrams`

: Packets sent

`InCsumErrors`

: Datagrams with checksum errors

`InErrors`

: Datagrams that could not be delivered to an application

`RcvbufErrors`

: Datagrams for which not enough socket buffer memory to receive

`SndbufErrors`

: Datagrams for which not enough socket buffer memory to transmit

`NoPorts`

: Datagrams received on a port with no listener

Percona LLC and/or its affiliates ©2024

ICMP

This section has the following metrics:

- ICMP Errors
- Messages/Redirects
- Echos
- Timestamps/Mask Requests

ICMP Errors

`InErrors`

Messages which the entity received but determined as having ICMP-specific errors (bad ICMP checksums, bad length, etc.)

`OutErrors`

Messages which this entity did not send due to problems discovered within ICMP, such as a lack of buffers

`InDestUnreachs`

Destination Unreachable messages received

`OutDestUnreachs`

Destination Unreachable messages sent

`InType3`

Destination unreachable

`OutType3`

Destination unreachable

`InCsumErrors`

Messages with ICMP checksum errors

`InTimeExcds`

Time Exceeded messages received

Messages/Redirects

`InMsgs`

Messages which the entity received. Note that this counter includes all those counted by `icmpInErrors`

`InRedirects`

Redirect messages received

`OutMsgs`

Messages which this entity attempted to send. Note that this counter includes all those counted by `icmpOutErrors`

Percona LLC and/or its affiliates ©2024

`OutRedirects`

Redirect messages sent. For a host, this object will always be zero, since hosts do not send redirects

Echos

`InEchoReps`

Echo Reply messages received

`InEchos`

Echo (request) messages received

`OutEchoReps`

Echo Reply messages sent

`OutEchos`

Echo (request) messages sent

Timestamps/Mask Requests

`InAddrMaskReps`

Address Mask Reply messages received

`InAddrMasks`

Address Mask Request messages received

`OutAddrMaskReps`

Address Mask Reply messages sent

`OutAddrMasks`

Address Mask Request messages sent

`InTimestampReps`

Timestamp Reply messages received

`InTimestamps`

Timestamp Request messages received

`OutTimestampReps`

Timestamp Reply messages sent

`OutTimestamps`

Timestamp Request messages sent

Percona LLC and/or its affiliates ©2024

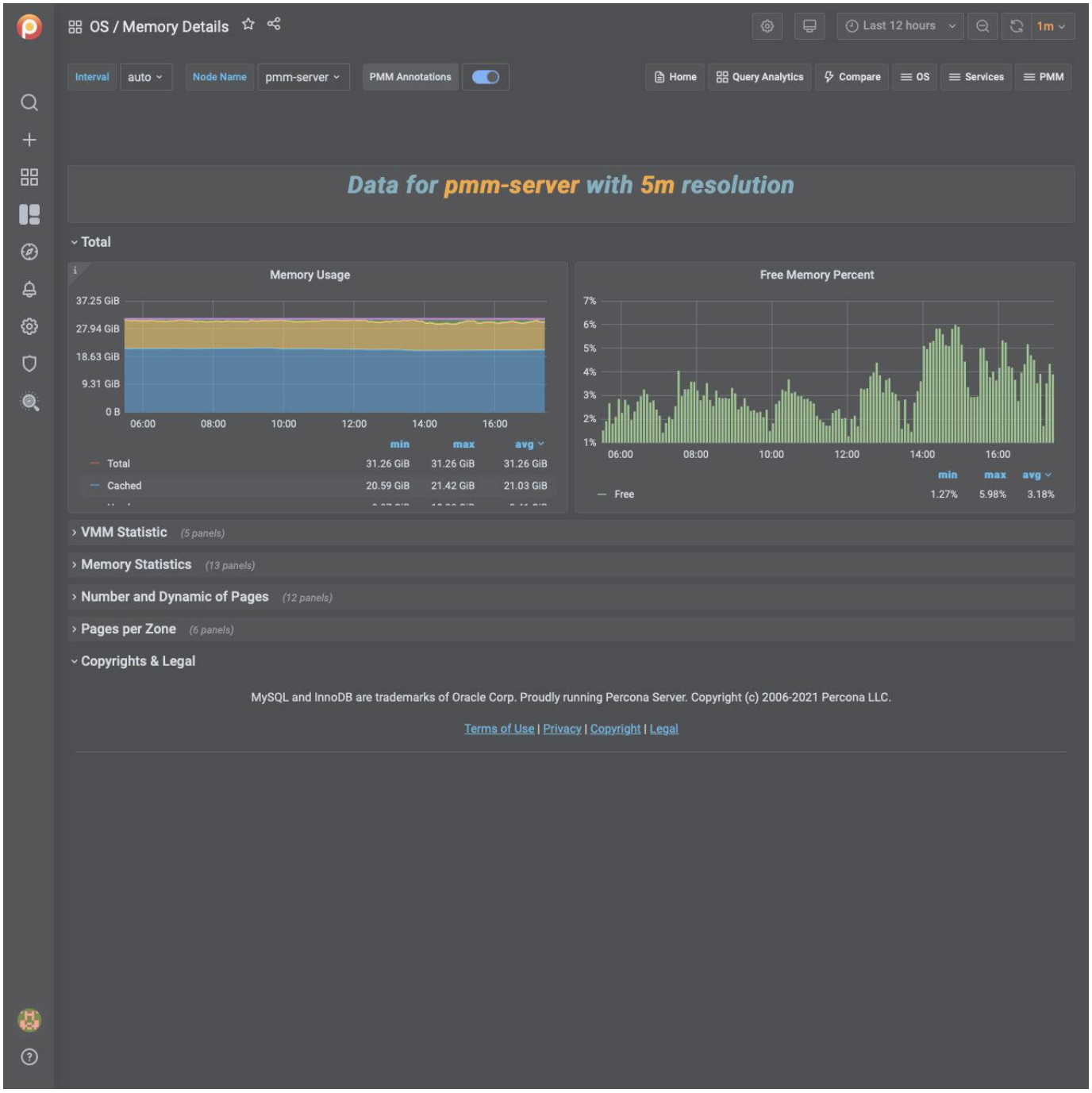
Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

MEMORY DETAILS



Percona LLC and/or its affiliates ©2024

Memory Usage

No description

Get expert help

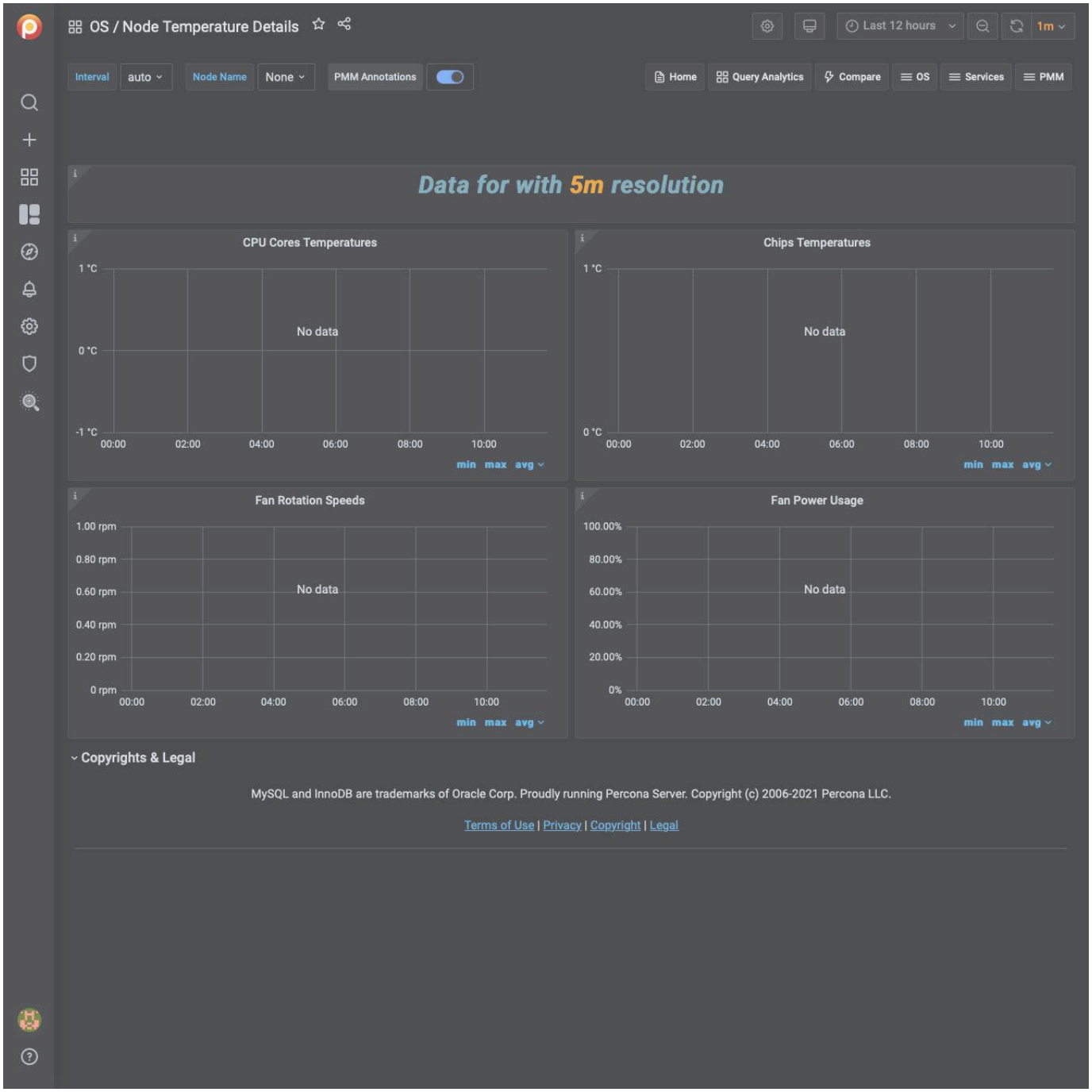
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-31

Percona LLC and/or its affiliates ©2024

NODE TEMPERATURE DETAILS



The Node Temperature Details dashboard exposes hardware monitoring and sensor data obtained through the `sysfs` virtual file system of the node.

Hardware monitoring devices attached to the CPU and/or other chips on the motherboard let you monitor the hardware health of a system. Most modern systems include several of such devices. The actual list can include temperature sensors, voltage sensors, fan speed sensors, and various additional features, such as the ability to control the rotation speed of the fans.

CPU Cores Temperatures

Presents data taken from the temperature sensors of the CPU

Percona LLC and/or its affiliates ©2024

Chips Temperatures

Presents data taken from the temperature sensors connected to other system controllers

Fan Rotation Speeds

Fan rotation speeds reported in RPM (rotations per minute).

Fan Power Usage

Describes the pulse width modulation of the PWN-equipped fans. PWM operates like a switch that constantly cycles on and off, thereby regulating the amount of power the fan gains: 100% makes it rotate at full speed, while lower percentage slows rotation down proportionally.

Get expert help

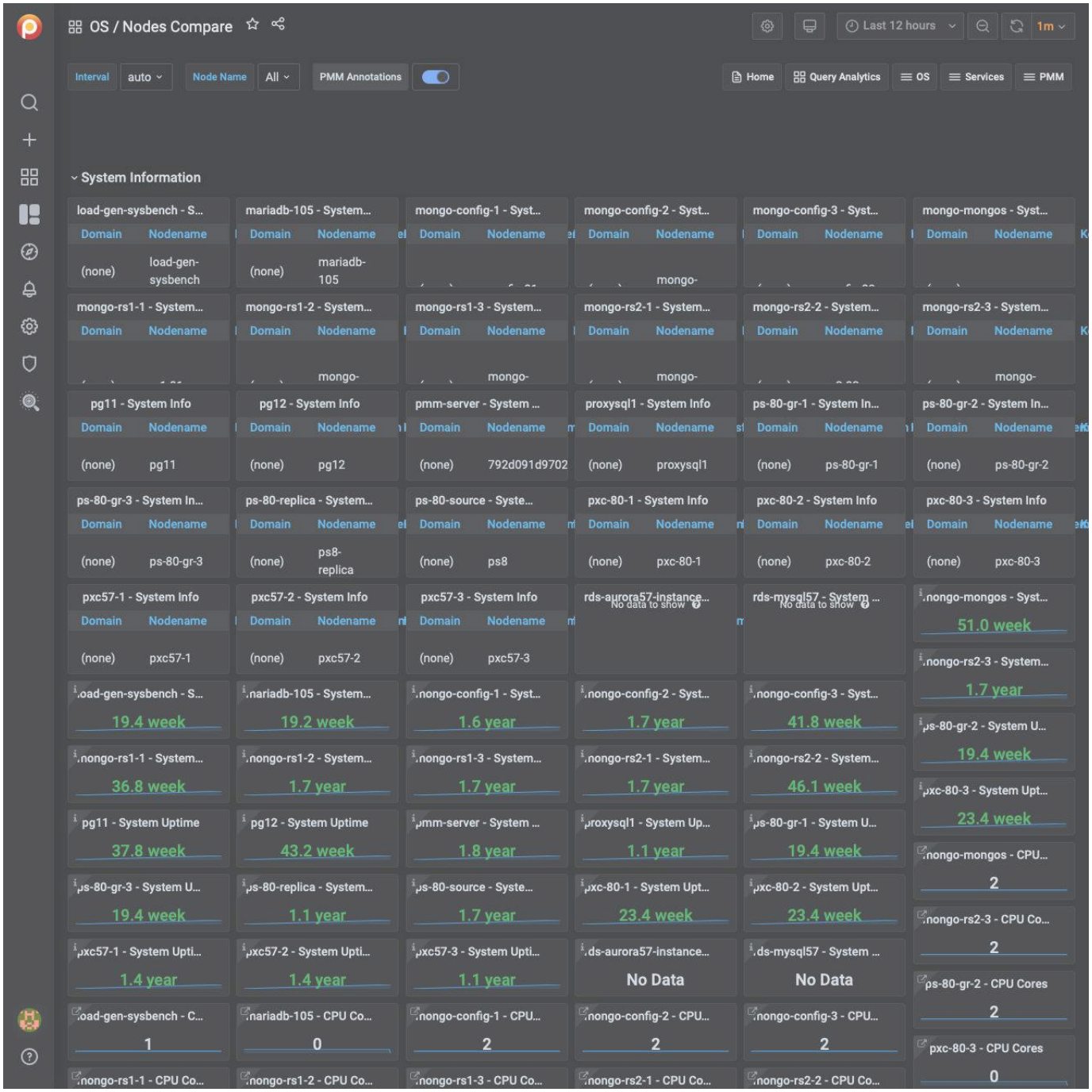
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-03

Percona LLC and/or its affiliates ©2024

NODES COMPARE



This dashboard lets you compare a wide range of parameters. Parameters of the same type are shown side by side for all servers, grouped into the following sections:

- System Information
- CPU
- Memory
- Disk Partitions
- Disk Performance
- Network

Percona LLC and/or its affiliates ©2024

The *System Information* section shows the *System Info* summary of each server, as well as *System Uptime*, *CPU Cores*, *RAM*, *Saturation Metrics*, and *Load Average* gauges.

The *CPU* section offers the *CPU Usage*, *Interrupts*, and *Context Switches* metrics.

In the *Memory* section, you can find the *Memory Usage*, *Swap Usage*, and *Swap Activity* metrics.

The *Disk Partitions* section encapsulates two metrics, *Mountpoint Usage* and *Free Space*.

The *Disk Performance* section contains the *I/O Activity*, *Disk Operations*, *Disk Bandwidth*, *Disk IO Utilization*, *Disk Latency*, and *Disk Load* metrics.

Finally, *Network* section shows *Network Traffic*, and *Network Utilization Hourly* metrics.

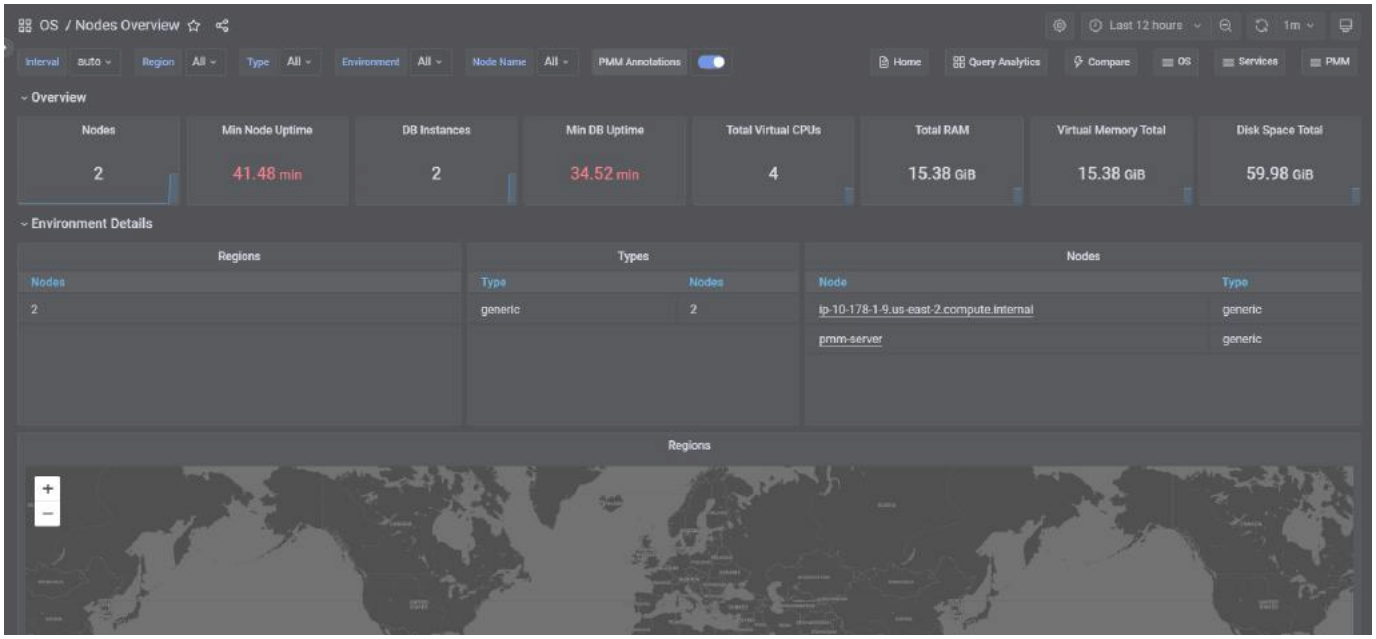
Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

NODES OVERVIEW



The Nodes Overview dashboard provides details about the efficiency of work of the following components. Each component is represented as a section in the dashboard.

- CPU
- Memory & Swap
- Disk
- Network

The *CPU* section offers the *CPU Usage*, *CPU Saturation and Max Core Usage*, *Interrupts and Context Switches*, and *Processes* metrics.

Percona LLC and/or its affiliates ©2024

In the *Memory* section, you can find the *Memory Utilization*, *Virtual Memory Utilization*, *Swap Space*, and *Swap Activity* metrics.

The *Disk* section contains the *I/O Activity*, *Global File Descriptors Usage*, *Disk IO Latency*, and *Disk IO Load* metrics.

In the *Network* section, you can find the *Network Traffic*, *Network Utilization Hourly*, *Local Network Errors*, and *TCP Retransmission* metrics.

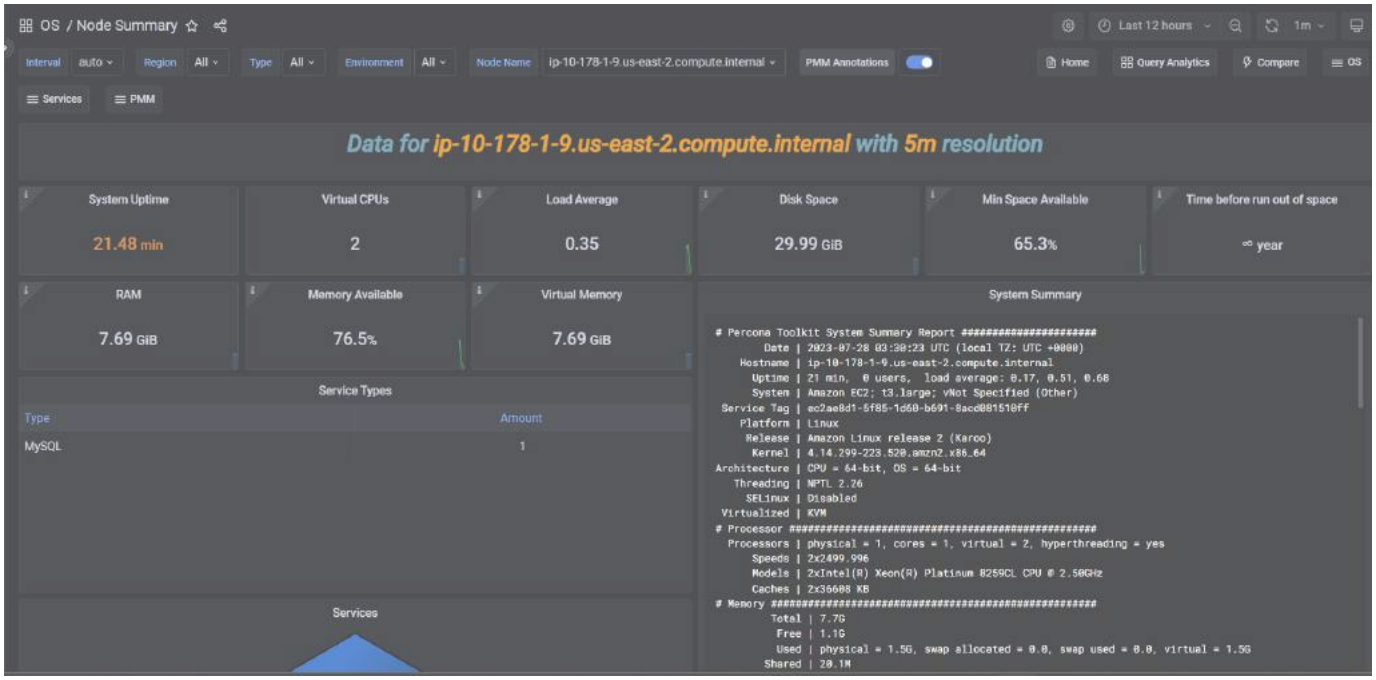
Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

NODE SUMMARY



System Summary

The output from `pt-summary`, one of the Percona Toolkit utilities.

CPU Usage

The CPU time is measured in clock ticks or seconds. It is useful to measure CPU time as a percentage of the CPU's capacity, which is called the CPU usage.

CPU Saturation and Max Core Usage

When a system is running with maximum CPU utilization, the transmitting and receiving threads must all share the available CPU. This will cause data to be queued more frequently to cope with the lack of CPU. CPU Saturation may be measured as the length of a wait queue, or the time spent waiting on the queue.

Percona LLC and/or its affiliates ©2024

Interrupts and Context Switches

Interrupt is an input signal to the processor indicating an event that needs immediate attention. An interrupt signal alerts the processor and serves as a request for the processor to interrupt the currently executing code, so that the event can be processed in a timely manner.

Context switch is the process of storing the state of a process or thread, so that it can be restored and resume execution at a later point. This allows multiple processes to share a single CPU, and is an essential feature of a multitasking operating system.

Processes

No description

Memory Utilization

No description

Virtual Memory Utilization

No description

Swap Space

No description

Swap Activity

Swap Activity is memory management that involves swapping sections of memory to and from physical storage.

I/O Activity

Disk I/O includes read or write or input/output operations involving a physical disk. It is the speed with which the data transfer takes place between the hard disk drive and RAM.

Global File Descriptors Usage

No description

Disk IO Latency

Shows average latency for Reads and Writes IO Devices. Higher than typical latency for highly loaded storage indicates saturation (overload) and is frequent cause of performance problems. Higher than normal latency also can indicate internal storage problems.

Disk IO Load

Shows how much disk was loaded for reads or writes as average number of outstanding requests at different period of time. High disk load is a good measure of actual storage utilization. Different storage types handle load differently - some will show latency increases on low loads others can handle higher load with no problems.

Network Traffic

Network traffic refers to the amount of data moving across a network at a given point in time.

Network Utilization Hourly

No description

Local Network Errors

Total Number of Local Network Interface Transmit Errors, Receive Errors and Drops. Should be Zero

Percona LLC and/or its affiliates ©2024

TCP Retransmission

Retransmission, essentially identical with Automatic repeat request (ARQ), is the resending of packets which have been either damaged or lost. Retransmission is one of the basic mechanisms used by protocols operating over a packet switched computer network to provide reliable communication (such as that provided by a reliable byte stream, for example TCP).

Get expert help

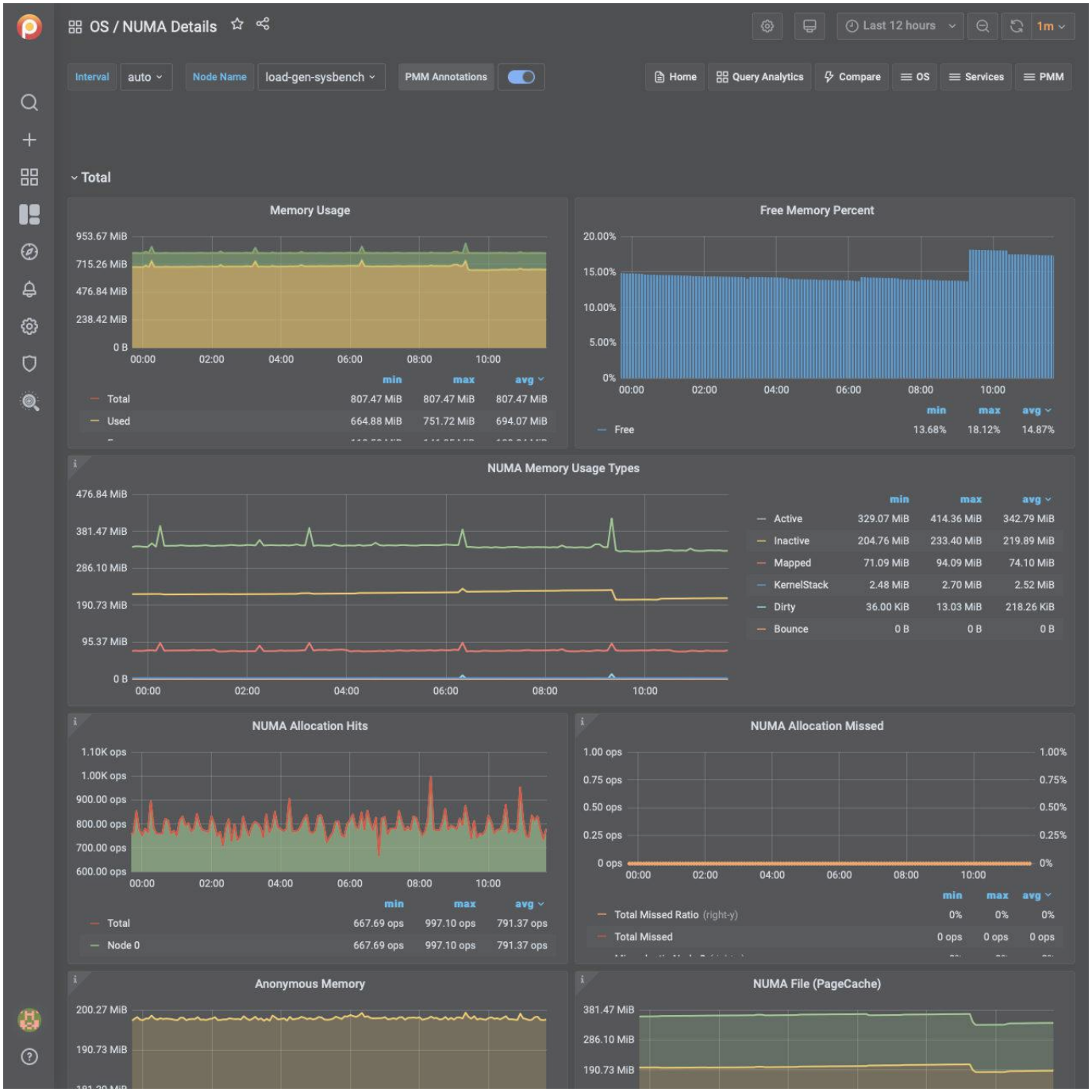
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

NUMA DETAILS



For each node, this dashboard shows metrics related to Non-uniform memory access (NUMA).

Memory Usage

Remotes over time the total, used, and free memory.

Free Memory Percent

Shows the free memory as the ratio to the total available memory.

NUMA Memory Usage Types

`Dirty`

Memory waiting to be written back to disk

Percona LLC and/or its affiliates ©2024

`Bounce`

Memory used for block device bounce buffers

`Mapped`

Files which have been mapped, such as libraries

`KernelStack`

The memory the kernel stack uses. This is not reclaimable.

NUMA Allocation Hits

Memory successfully allocated on this node as intended.

NUMA Allocation Missed

Memory missed is allocated on a node despite the process preferring some different node.

Memory foreign is intended for a node, but actually allocated on some different node.

Anonymous Memory

Active

Anonymous memory that has been used more recently and usually not swapped out.

Inactive

Anonymous memory that has not been used recently and can be swapped out.

NUMA File (PageCache)

Active(file) Pagecache memory that has been used more recently and usually not reclaimed until needed.

Inactive(file) Pagecache memory that can be reclaimed without huge performance impact.

Shared Memory

Shmem Total used shared memory (shared between several processes, thus including RAM disks, SYS-V-IPC and BSD like SHMEM).

HugePages Statistics

Total

Number of hugepages being allocated by the kernel (Defined with `vm.nr_hugepages`).

Free

The number of hugepages not being allocated by a process

`Surp`

The number of hugepages in the pool above the value in `vm.nr_hugepages`. The maximum number of surplus hugepages is controlled by `vm.nr_overcommit_hugepages`.

Local Processes

Memory allocated on a node while a process was running on it.

Percona LLC and/or its affiliates ©2024

Remote Processes

Memory allocated on a node while a process was running on some other node.

Slab Memory

`Slab`

Allocation is a memory management mechanism intended for the efficient memory allocation of kernel objects.

`SReclaimable`

The part of the Slab that might be reclaimed (such as caches).

`SUnreclaim`

The part of the Slab that can’t be reclaimed under memory pressure

Get expert help

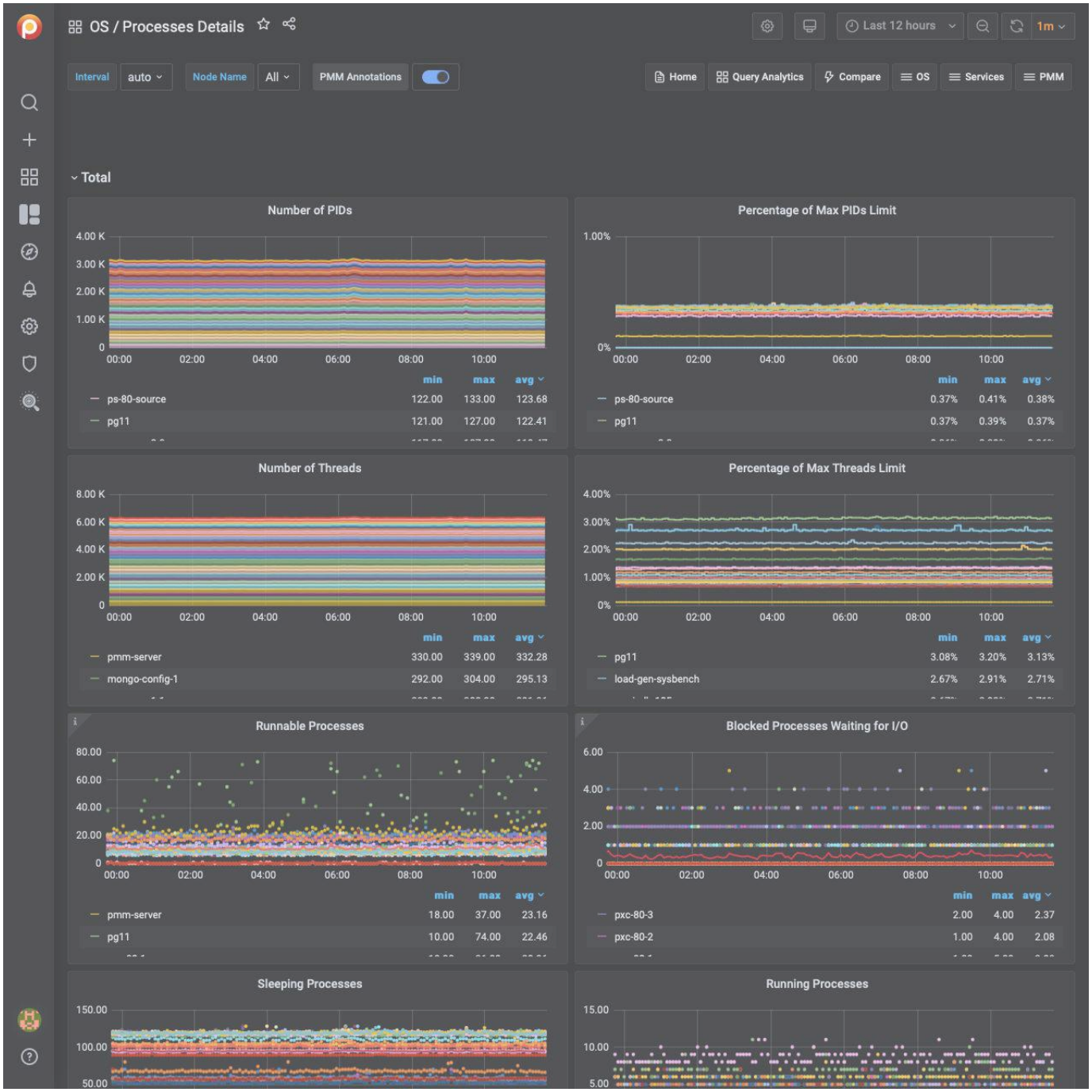
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

PROCESSES DETAILS



The Processes Details dashboard displays Linux process information - PIDs, Threads, and Processes. The dashboard shows how many processes/threads are either in the kernel run queue (runnable state) or in the blocked queue (waiting for I/O). When the number of process in the runnable state is constantly higher than the number of CPU cores available, the load is CPU bound. When the number of process blocked waiting for I/O is large, the load is disk bound. The running average of the sum of these two quantities is the basis of the `loadavg` metric.

The dashboard consists of two parts: the first section describes metrics for all hosts, and the second part provides charts for each host.

Percona LLC and/or its affiliates ©2024

Charts for all hosts, available in the first section, are the following ones:

- States of Processes
- Number of PIDs
- Percentage of Max PIDs Limit
- Number of Threads
- Percentage of Max Threads Limit
- Runnable Processes
- Blocked Processes Waiting for I/O
- Sleeping Processes
- Running Processes
- Disk Sleep Processes
- Stopped Processes
- Zombie Processes
- Dead Processes

The following charts are present in the second part, available for each host:

- Processes
- States of Processes
- Number of PIDs
- Percentage of Max PIDs Limit
- Number of Threads
- Percentage of Max Threads Limit

Number of PIDs

No description

Percentage of Max PIDs Limit

No description

Number of Threads

No description

Percentage of Max Threads Limit

No description

Runnable Processes

Processes

The Processes graph shows how many processes/threads are either in the kernel run queue (runnable state) or in the blocked queue (waiting for I/O). When the number of process in the runnable state is constantly higher than the number of CPU cores available, the load is CPU bound. When the number of process blocked waiting for I/O is large, the load is disk bound. The running average of the sum of these two quantities is the basis of the `loadavg` metric.

Blocked Processes Waiting for I/O

Processes

Percona LLC and/or its affiliates ©2024

The Processes graph shows how many processes/threads are either in the kernel run queue (runnable state) or in the blocked queue (waiting for I/O). When the number of process in the runnable state is constantly higher than the number of CPU cores available, the load is CPU bound. When the number of process blocked waiting for I/O is large, the load is disk bound. The running average of the sum of these two quantities is the basis of the `loadavg` metric.

Sleeping Processes

No description

Running Processes

No description

Disk Sleep Processes

No description

Stopped Processes

No description

Zombie Processes

No description

Dead Processes

No description

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

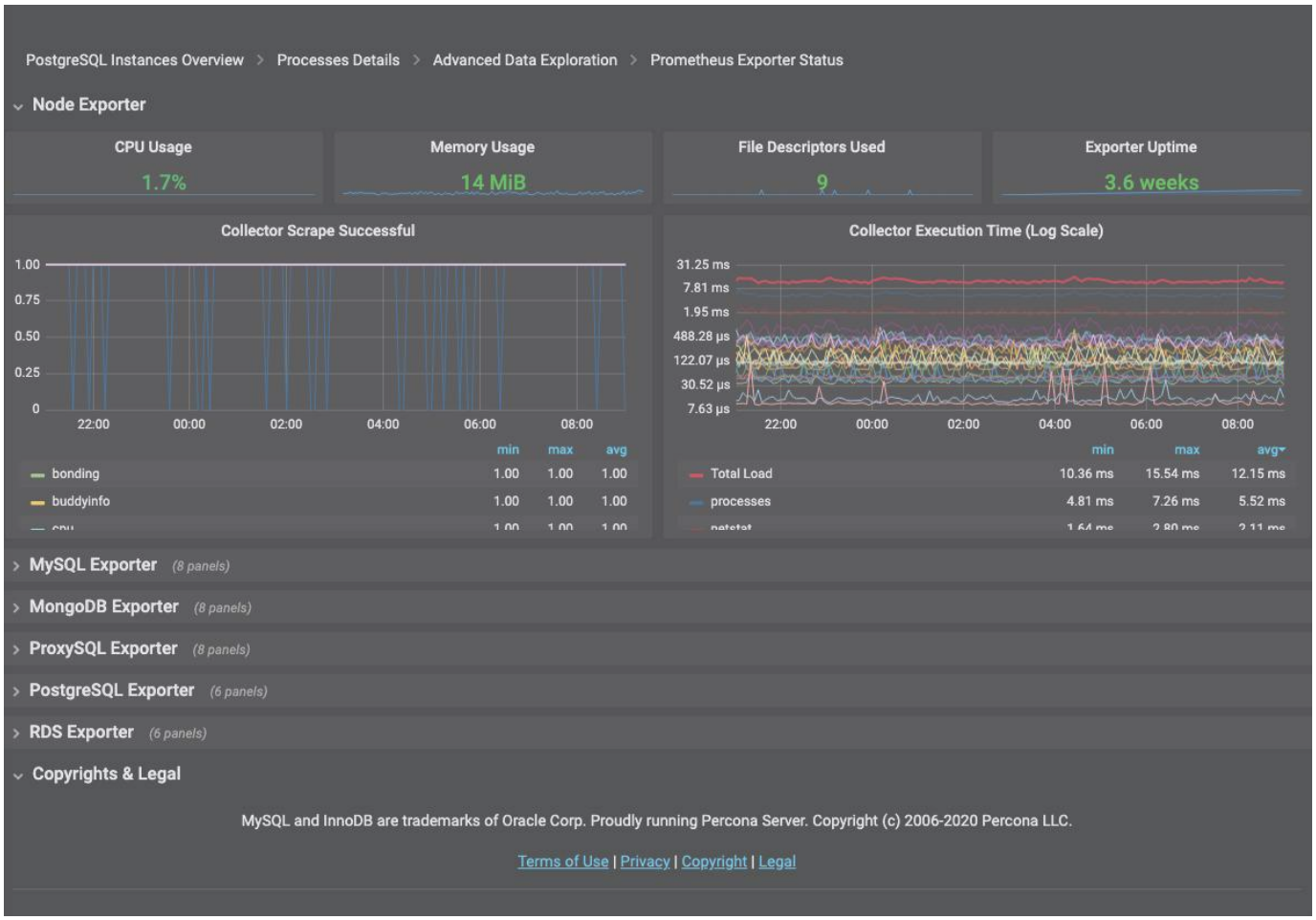
Community Forum Get a Percona Expert

2021-06-25

Percona LLC and/or its affiliates ©2024

Prometheus Dashboards

PROMETHEUS EXPORTER STATUS



The Prometheus Exporter Status dashboard reports the consumption of resources by the Prometheus exporters used by PMM. For each exporter, this dashboard reveals the following information:

- CPU usage
- Memory usage
- File descriptors used
- Exporter uptime

Get expert help

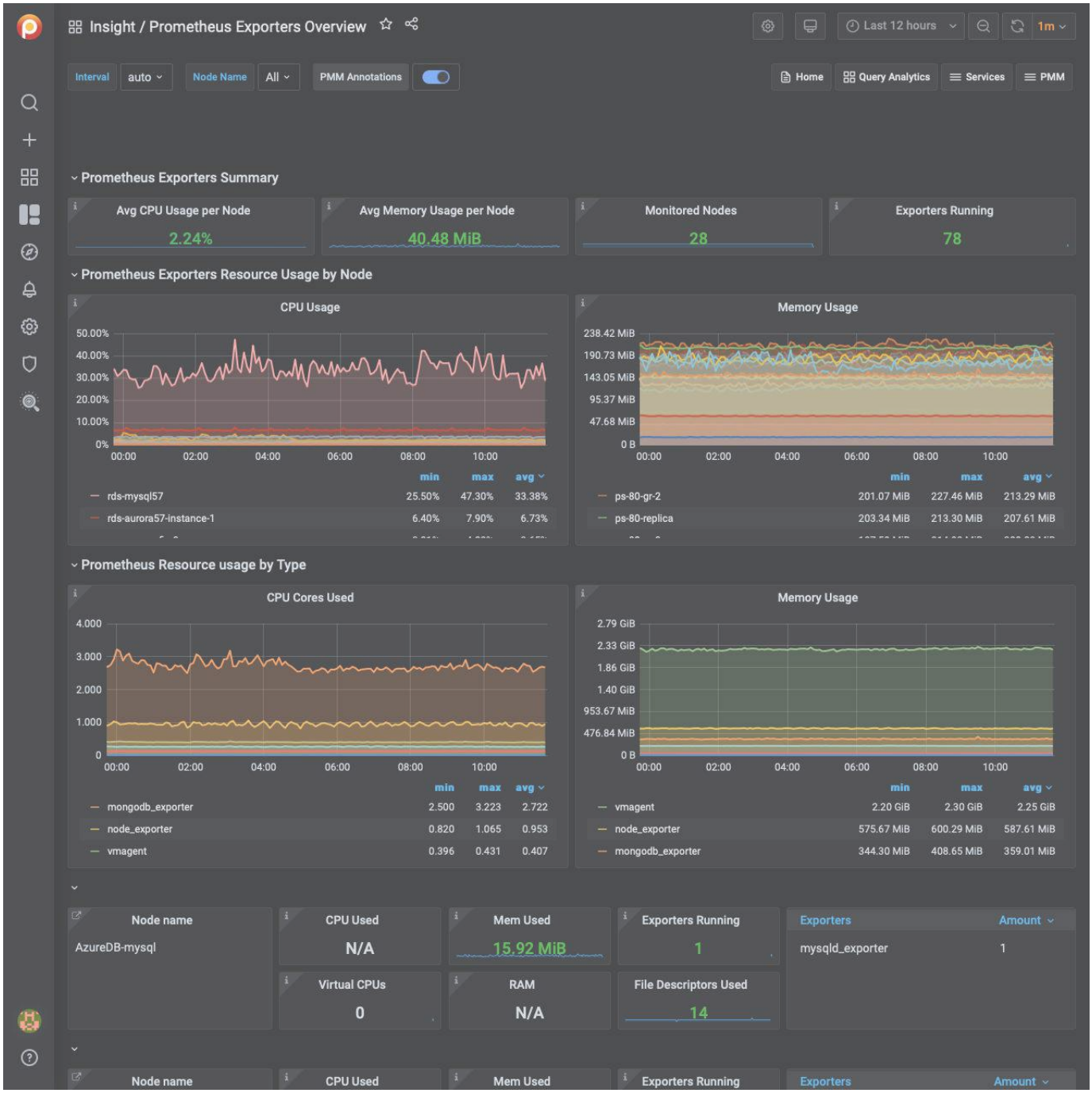
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

Percona LLC and/or its affiliates ©2024

PROMETHEUS EXPORTERS OVERVIEW



Prometheus Exporters Summary

This section provides a summary of how exporters are used across the selected hosts. It includes the average usage of CPU and memory as well as the number of hosts being monitored and the total number of running exporters.

Avg CPU Usage per Host

Shows the average CPU usage in percent per host for all exporters.

Avg Memory Usage per Host

Shows the Exporters average Memory usage per host.

Percona LLC and/or its affiliates ©2024

Monitored Hosts

Shows the number of monitored hosts that are running Exporters.

Exporters Running

Shows the total number of Exporters running with this PMM Server instance.

The CPU usage and memory usage do not include the additional CPU and memory usage required to produce metrics by the application or operating system.

Prometheus Exporters Resource Usage by Node

This section shows how resources, such as CPU and memory, are being used by the exporters for the selected hosts.

CPU Usage

Plots the Exporters' CPU usage across each monitored host (by default, All hosts).

Memory Usage

Plots the Exporters' Memory usage across each monitored host (by default, All hosts).

Prometheus Exporters Resource Usage by Type

This section shows how resources, such as CPU and memory, are being used by the exporters for host types: MySQL, MongoDB, ProxySQL, and the system.

CPU Cores Used

Shows the Exporters' CPU Cores used for each type of Exporter.

Memory Usage

Shows the Exporters' memory used for each type of Exporter.

List of Hosts

At the bottom, this dashboard shows details for each running host.

CPU Used

Show the CPU usage as a percentage for all Exporters.

Mem Used

Shows total Memory Used by Exporters.

Exporters Running

Shows the number of Exporters running.

RAM

Shows the total amount of RAM of the host.

Virtual CPUs

Shows the total number of virtual CPUs on the host.

Percona LLC and/or its affiliates ©2024

You can click the value of the *CPU Used*, *Memory Used*, or *Exporters Running* columns to open the Prometheus Exporter Status dashboard for further analysis.

See also

Percona blog: Understand Your Prometheus Exporters with Percona Monitoring and Management (PMM)

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

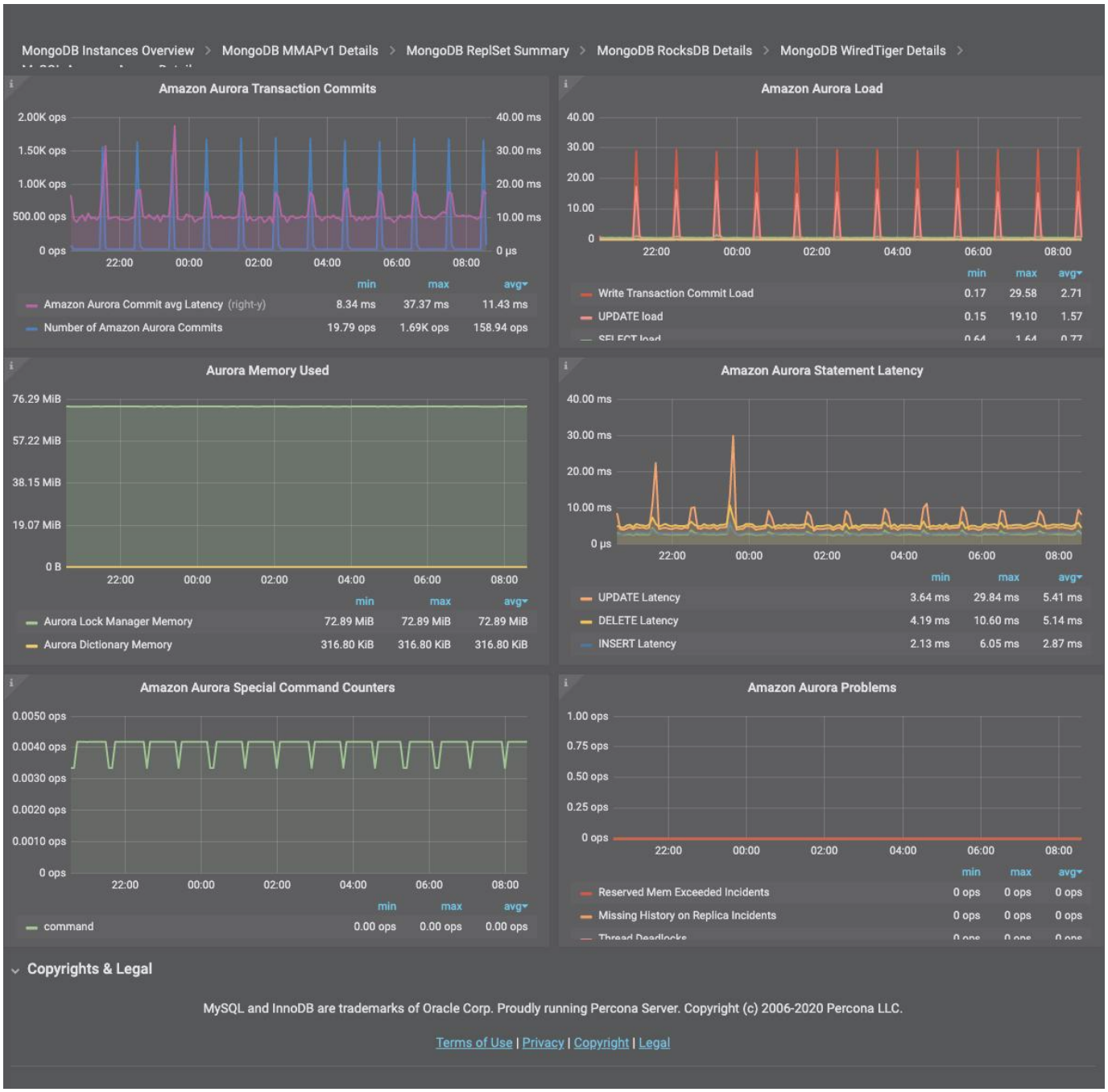
Community Forum Get a Percona Expert

2021-06-10

Percona LLC and/or its affiliates ©2024

MySQL Dashboards

MYSQL AMAZON AURORA DETAILS



Amazon Aurora Transaction Commits

This graph shows the number of Commits which Amazon Aurora engine performed as well as average commit latency. Graph Latency does not always correlate with the number of performed commits and can be quite high in certain situations.

- **Number of Amazon Aurora Commits**: The average number of commit operations per second.
- **Amazon Aurora Commit avg Latency**: The average amount of latency for commit operations

Percona LLC and/or its affiliates ©2024

Amazon Aurora Load

This graph shows us what statements contribute most load on the system as well as what load corresponds to Amazon Aurora transaction commit.

- **Write Transaction Commit Load**: Load in Average Active Sessions per second for COMMIT operations
- **UPDATE load**: Load in Average Active Sessions per second for UPDATE queries
- **SELECT load**: Load in Average Active Sessions per second for SELECT queries
- **DELETE load**: Load in Average Active Sessions per second for DELETE queries
- **INSERT load**: Load in Average Active Sessions per second for INSERT queries

An *active session* is a connection that has submitted work to the database engine and is waiting for a response from it. For example, if you submit an SQL query to the database engine, the database session is active while the database engine is processing that query.

Aurora Memory Used

This graph shows how much memory is used by Amazon Aurora lock manager as well as amount of memory used by Amazon Aurora to store Data Dictionary.

- **Aurora Lock Manager Memory**: the amount of memory used by the Lock Manager, the module responsible for handling row lock requests for concurrent transactions.
- **Aurora Dictionary Memory**: the amount of memory used by the Dictionary, the space that contains metadata used to keep track of database objects, such as tables and indexes.

Amazon Aurora Statement Latency

This graph shows average latency for the most important types of statements. Latency spikes are often indicative of the instance overload.

- **DDL Latency:** Average time to execute DDL queries
- **DELETE Latency**: Average time to execute DELETE queries
- **UPDATE Latency**: Average time to execute UPDATE queries
- **SELECT Latency**: Average time to execute SELECT queries
- **INSERT Latency**: Average time to execute INSERT queries

Amazon Aurora Special Command Counters

Amazon Aurora MySQL allows a number of commands which are not available in standard MySQL. This graph shows usage of such commands. Regular `unit_test` calls can be seen in default Amazon Aurora install, the rest will depend on your workload.

- `show_volume_status`: The number of executions per second of the command SHOW VOLUME STATUS. The SHOW VOLUME STATUS query returns two server status variables, Disks and Nodes. These variables represent the total number of logical blocks of data and storage nodes, respectively, for the DB cluster volume.
- `awslambda`: The number of AWS Lambda calls per second. AWS Lambda is an event-drive, server-less computing platform provided by AWS. It is a compute service that run codes in response to an event. You can run any kind of code from Aurora invoking Lambda from a stored procedure or a trigger.
- `alter_system`: The number of executions per second of the special query ALTER SYSTEM, that is a special query to simulate an instance crash, a disk failure, a disk congestion or a replica failure. It’s a useful query for testing the system.

Percona LLC and/or its affiliates ©2024

Amazon Aurora Problems

This graph shows different kinds of Internal Amazon Aurora MySQL Problems which general should be zero in normal operation.

Anything non-zero is worth examining in greater depth.

Get expert help

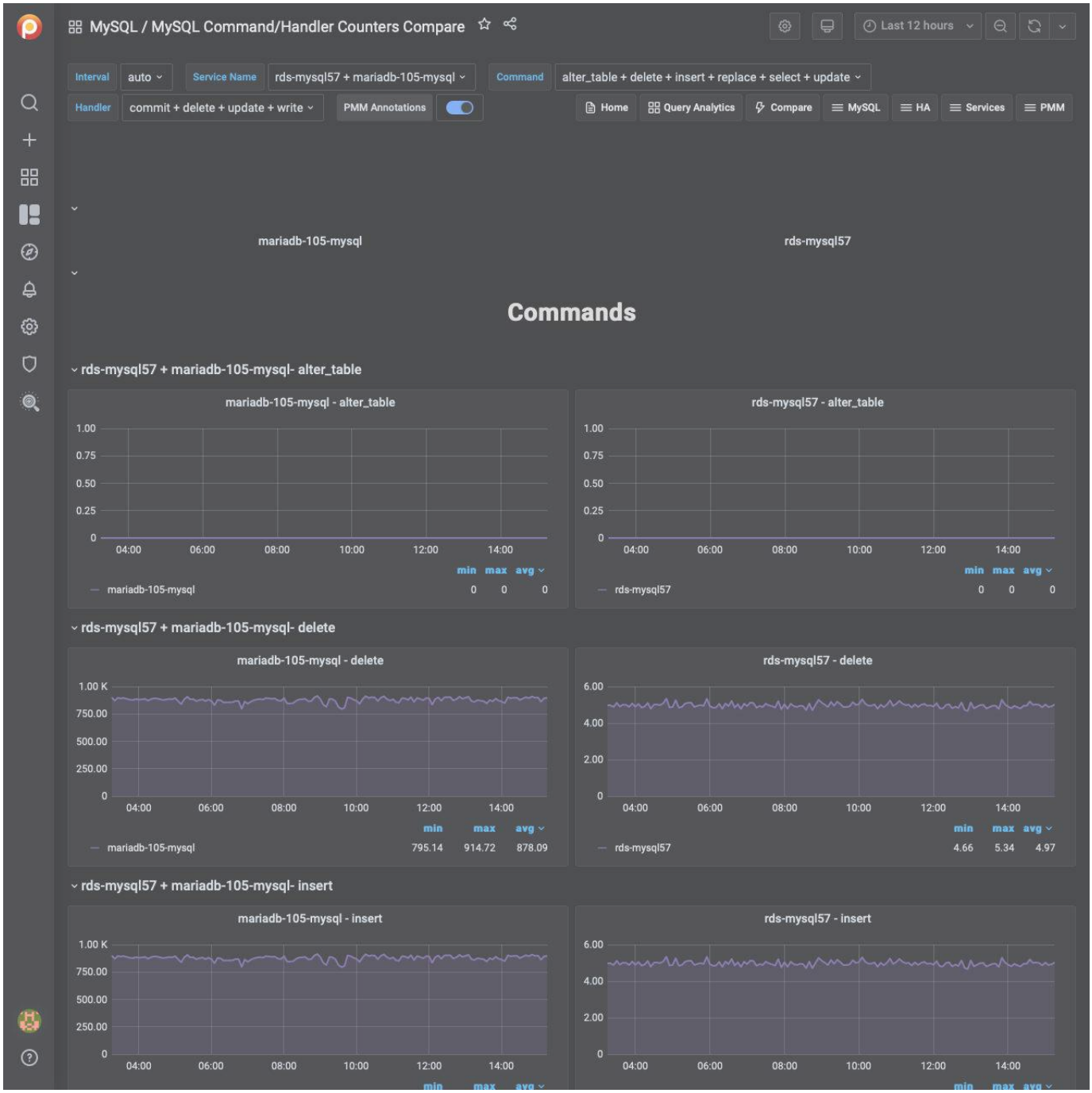
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

MYSQL COMMAND/HANDLER COUNTERS COMPARE



This dashboard shows server status variables. On this dashboard, you may select multiple servers and compare their counters simultaneously.

Server status variables appear in two sections: *Commands* and *Handlers*. Choose one or more variables in the *Command* and *Handler* fields in the top menu to select the variables which will appear in the *COMMANDS* or *HANDLERS* section for each host. Your comparison may include from one up to three hosts.

By default or if no item is selected in the menu, PMM displays each command or handler respectively.

Percona LLC and/or its affiliates ©2024

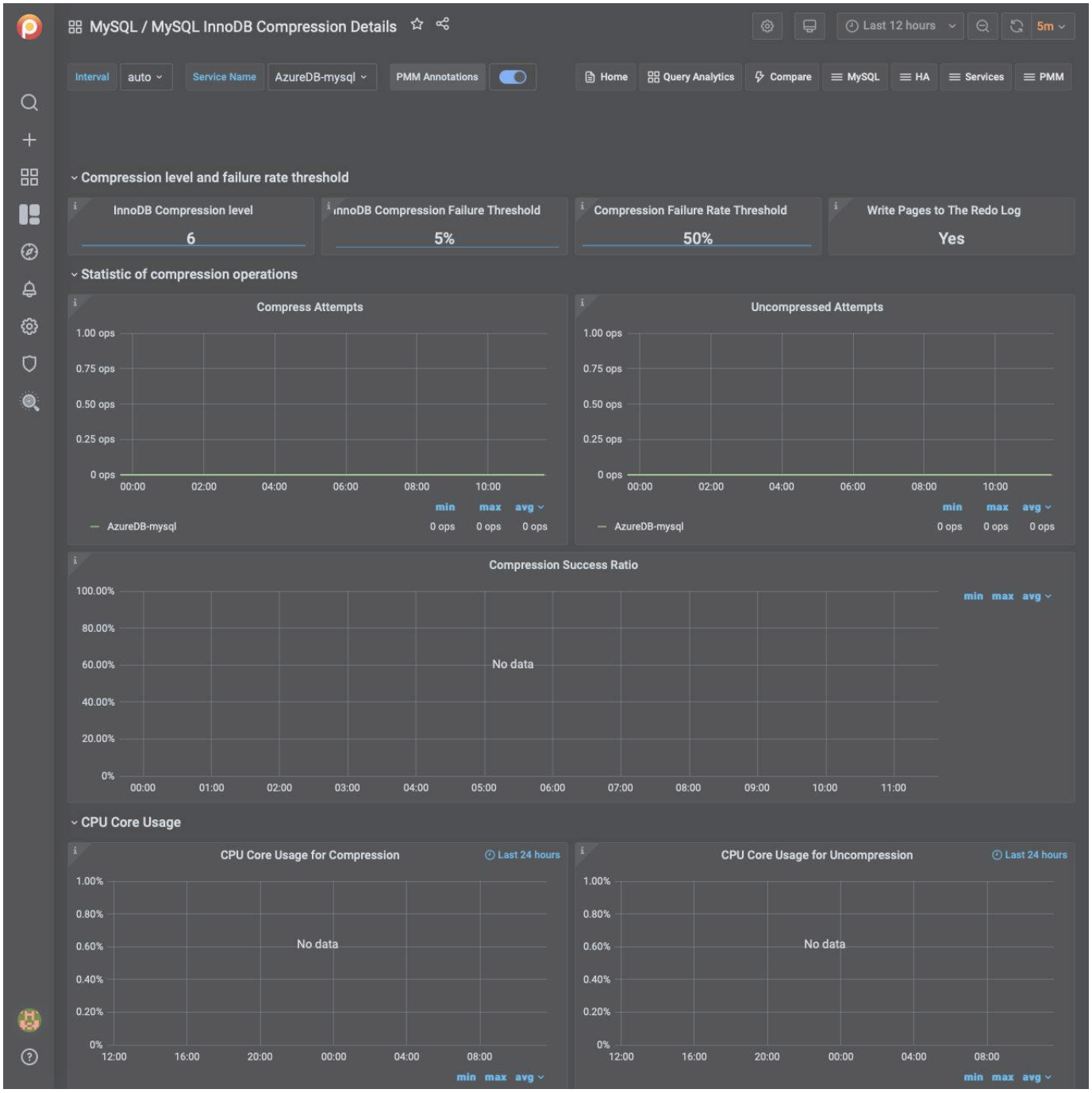
Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

MYSQL INNODB COMPRESSION DETAILS



This dashboard helps you analyze the efficiency of InnoDB compression.

Percona LLC and/or its affiliates ©2024

Compression level and failure rate threshold

InnoDB Compression Level

The level of zlib compression to use for InnoDB compressed tables and indexes.

InnoDB Compression Failure Threshold

The compression failure rate threshold for a table.

Compression Failure Rate Threshold

The maximum percentage that can be reserved as free space within each compressed page, allowing room to reorganize the data and modification log within the page when a compressed table or index is updated and the data might be recompressed.

Write Pages to the Redo Log

Specifies whether images of re-compressed pages are written to the redo log. Re-compression may occur when changes are made to compressed data.

Statistic of compression operations

Compress Attempts

Number of compression operations attempted. Pages are compressed whenever an empty page is created or the space for the uncompressed modification log runs out.

Uncompressed Attempts

Number of uncompression operations performed. Compressed InnoDB pages are uncompressed whenever compression fails, or the first time a compressed page is accessed in the buffer pool and the uncompressed page does not exist.

CPU Core Usage

CPU Core Usage for Compression

Shows the time in seconds spent by InnoDB Compression operations.

CPU Core Usage for Uncompression

Shows the time in seconds spent by InnoDB Uncompression operations.

Buffer Pool Total

Total Used Pages

Shows the total amount of used compressed pages into the InnoDB Buffer Pool split by page size.

Total Free Pages

Shows the total amount of free compressed pages into the InnoDB Buffer Pool split by page size.

Get expert help

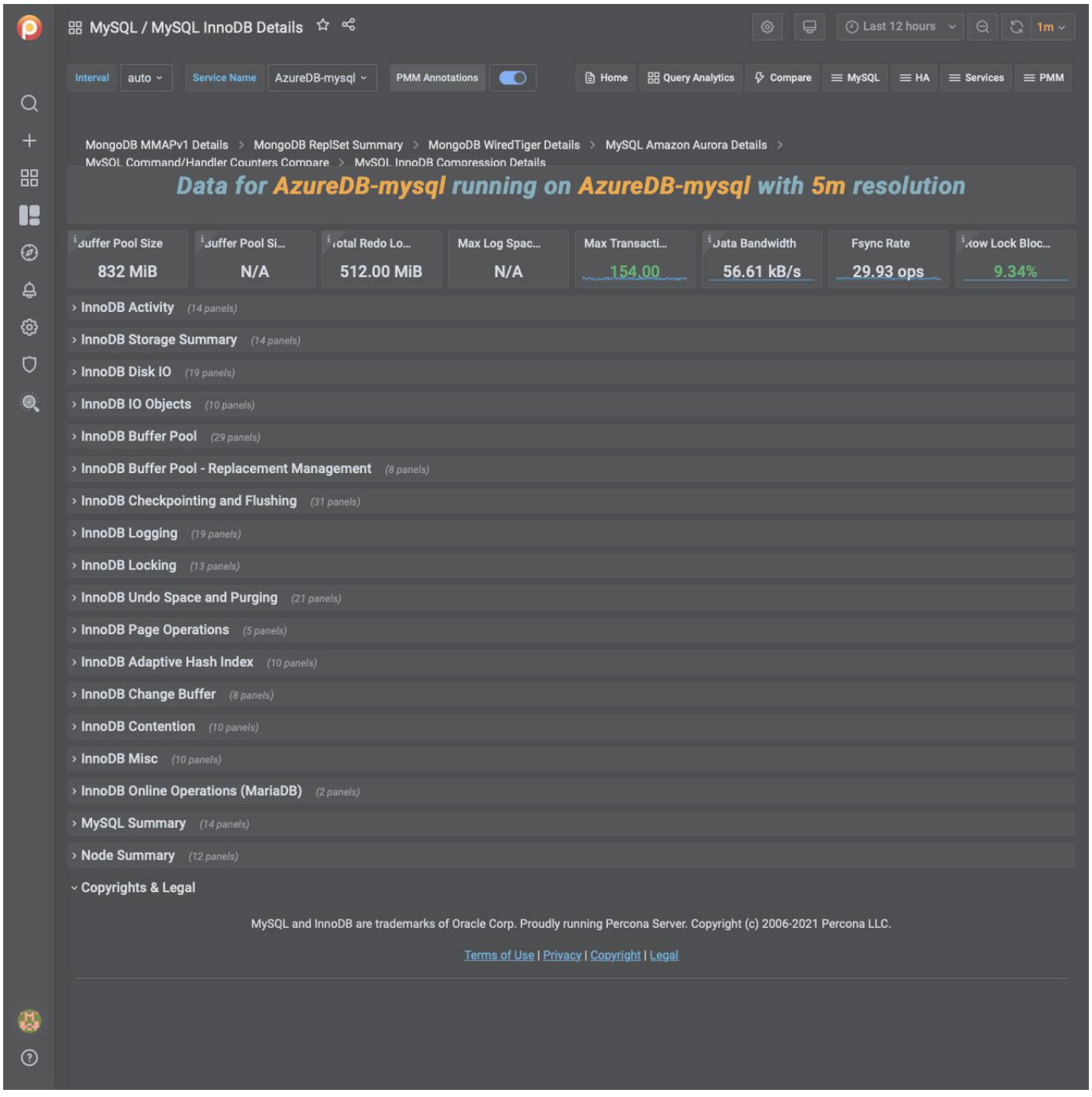
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2021-05-11

MYSQL INNODB DETAILS



Tip

If metrics are missing, try running: `SET GLOBAL innodb_monitor_enable=all;` in the MySQL client.

Percona LLC and/or its affiliates ©2024

InnoDB Activity

Writes (Rows) Writes (Transactions) Row Writes per Trx

Rows Written Per Transactions which modify rows. This is better indicator of transaction write size than looking at all transactions which did not do any writes as well.

Rows Read Per Trx Log Space per Trx Rollbacks

Percent of Transaction Rollbacks (as portion of read-write transactions).

BP Reqs Per Row

Number of Buffer Pool requests per Row Access. High numbers here indicate going through long undo chains, deep trees and other inefficient data access. It can be less than zero due to several rows being read from single page.

Log Fsync Per Trx

Log Fsync Per Transaction.

InnoDB Row Reads InnoDB Row Operations

This graph allows you to see which operations occur and the number of rows affected per operation. A graph like Queries Per Second will give you an idea of queries, but one query could effect millions of rows.

InnoDB Row Writes InnoDB Row Operations

This graph allows you to see which operations occur and the number of rows affected per operation. A graph like Queries Per Second will give you an idea of queries, but one query could effect millions of rows.

InnoDB Read-Only Transactions InnoDB Read-Write Transactions InnoDB Transactions Information (RW)

The InnoDB Transactions Information graph shows details about the recent transactions. Transaction IDs Assigned represents the total number of transactions initiated by InnoDB. RW Transaction Commits are the number of transactions not read-only. Insert-Update Transactions Commits are transactions on the Undo entries. Non Locking RO Transaction Commits are transactions commit from select statement in auto-commit mode or transactions explicitly started with “start transaction read only”.

Misc InnoDB Transactions Information

Additional InnoDB Transaction Information

InnoDB Storage Summary

InnoDB Tables

Current Number of InnoDB Tables in database

Data Buffer Pool Fit

Buffer Pool Size as Portion of the Data

Avg Row Size

Amount of Data Per Row

Index Size Per Row

Index Size Per Row shows how much space we’re using for indexes on per row basics

InnoDB Data Summary Space Allocated

Percona LLC and/or its affiliates ©2024

Total Amount of Space Allocated. May not exactly match amount of space used on file system but provided great guidance.

Space Used

Space used in All InnoDB Tables. Reported Allocated Space Less Free Space.

Data Length

Space Used by Data (Including Primary Key).

Index Length

Space Used by Secondary Indexes.

Estimated Rows

Estimated number of Rows in InnoDB Storage Engine. It is not exact value and it can change abruptly as information is updated.

Indexing Overhead

How Much Indexes Take Compared to Data.

Free Space Percent

How Much Space is Free. Too high value wastes space on disk.

Free

Allocated Space not currently used by Data or Indexes.

InnoDB File Per Table

If Enabled, By Default every Table will have its own Tablespace represented as its own `.idb` file rather than all tables stored in single system tablespace.

InnoDB Disk IO

InnoDB Page Size Avg Data Read Rq Size Avg Data Write Rq Size Avg Log Write Rq Size Data Written Per Fsync Log Written Per Fsync Data Read Per Row Read Data Written Per Row Written

Due to difference in timing of Row Write and Data Write the value may be misleading on short intervals.

InnoDB Data I/O

InnoDB I/O

- Data Writes - The total number of InnoDB data writes.
- Data Reads - The total number of InnoDB data reads (OS file reads).
- Log Writes - The number of physical writes to the InnoDB redo log file.
- Data Fsyncs - The number of fsync() operations. The frequency of `fsync()` calls is influenced by the setting of the `innodb_flush_method` configuration option.

InnoDB Data Bandwidth InnoDB Log IO

Percona LLC and/or its affiliates ©2024

InnoDB I/O

- Data Writes - The total number of InnoDB data writes.
- Data Reads - The total number of InnoDB data reads (OS file reads).
- Log Writes - The number of physical writes to the InnoDB redo log file.
- Data Fsyncs - The number of `fsync()` operations. The frequency of `fsync()` calls is influenced by the setting of the `innodb_flush_method` configuration option.

InnoDB FSyncs InnoDB Pending IO InnoDB Pending Fsyncs InnoDB Auto Extend Increment

When Growing InnoDB System Tablespace extend it by this size at the time.

InnoDB Double Write

Whether InnoDB Double Write Buffer is enabled. Doing so doubles amount of writes InnoDB has to do to storage but is required to avoid potential data corruption during the crash on most storage subsystems.

InnoDB Fast Shutdown

Fast Shutdown means InnoDB will not perform complete Undo Space and Change Buffer cleanup on shutdown, which is faster but may interfere with certain major upgrade operations.

InnoDB Open Files

Maximum Number of Files InnoDB is Allowed to use.

InnoDB File Use

Portion of Allowed InnoDB Open Files Use.

InnoDB IO Objects

InnoDB IO Targets Write Load

Write Load Includes both Write and fsync (referred as misc).

InnoDB Buffer Pool

Buffer Pool Size

InnoDB Buffer Pool Size

InnoDB maintains a storage area called the buffer pool for caching data and indexes in memory. Knowing how the InnoDB buffer pool works, and taking advantage of it to keep frequently accessed data in memory, is one of the most important aspects of MySQL tuning. The goal is to keep the working set in memory. In most cases, this should be between 60%-90% of available memory on a dedicated database host, but depends on many factors.

Buffer Pool Size of Total RAM

InnoDB Buffer Pool Size % of Total RAM

InnoDB maintains a storage area called the buffer pool for caching data and indexes in memory. Knowing how the InnoDB buffer pool works, and taking advantage of it to keep frequently accessed data in memory, is one of the most important aspects of MySQL tuning. The goal is to keep the working set in memory. In most cases, this should be between 60%-90% of available memory on a dedicated database host, but depends on many factors.

NUMA Interleave

Interleave Buffer Pool between NUMA zones to better support NUMA systems.

Buffer Pool Activity

Percona LLC and/or its affiliates ©2024

Combined value of Buffer Pool Read and Write requests.

BP Data

Percent of Buffer Pool Occupied by Cached Data.

BP Data Dirty

Percent of Data which is Dirty.

BP Miss Ratio

How often buffer pool read requests have to do read from the disk. Keep this percent low for good performance.

BP Write Buffering

Number of Logical Writes to Buffer Pool Per logical Write.

InnoDB Buffer Pool LRU Sub-Chain Churn Buffer Pool Chunk Size

Size of the “Chunk” for buffer pool allocation. Allocation of buffer pool will be rounded by this number. It also affects the performance impact of online buffer pool resize.

Buffer Pool Instances

Number of Buffer Pool Instances. Higher values allow to reduce contention but also increase overhead.

Read Ahead IO Percent

Percent of Reads Caused by InnoDB Read Ahead.

Read Ahead Wasted

Percent of Pages Fetched by Read Ahead Evicted Without Access.

Dump Buffer Pool on Shutdown Load Buffer Pool at Startup Portion of Buffer Pool To Dump/Load

Larger Portion increases dump/load time but get more of original buffer pool content and hence may reduce warmup time.

Include Buffer Pool in Core Dump

Whenever to Include Buffer Pool in Crash Core Dumps. Doing so may dramatically increase core dump file slow down restart. Only makes a difference if core dumping on crash is enabled.

InnoDB Old Blocks

Percent of The Buffer Pool To be Reserved for “Old Blocks” - which has been touched repeatedly over period of time.

InnoDB Old Blocks Time

The Time which has to pass between multiple touches for the block for it to qualify as old block.

InnoDB Random Read Ahead

Is InnoDB Random ReadAhead Enabled.

InnoDB Random Read Ahead

The Threshold (in Pages) to trigger Linear Read Ahead.

InnoDB Read IO Threads

Number of Threads used to Schedule Reads.

Percona LLC and/or its affiliates ©2024

InnoDB Write IO Threads

Number of Threads used to Schedule Writes.

InnoDB Native AIO Enabled

Whether Native Asynchronous IO is enabled. Strongly recommended for optimal performance.

InnoDB Buffer Pool - Replacement Management

LRU Scan Depth

InnoDB LRU Scan Depth

This variable defines InnoDB Free Page Target per buffer pool. When number of free pages falls below this number this number page cleaner will make required amount of pages free, flushing or evicting pages from the tail of LRU as needed.

LRU Clean Page Searches

When Page is being read (or created) the Page need to be allocated in Buffer Pool.

Free List Miss Rate

The most efficient way to get a clean page is to grab one from free list. However if no pages are available in Free List the LRU scan needs to be performed.

LRU Get Free Loops

If Free List was empty LRU Get Free Loop will be performed. It may perform LRU scan or may use some other heuristics and shortcuts to get free page.

LRU Scans

If Page could not be find any Free list and other shortcuts did not work, free page will be searched by scanning LRU chain which is not efficient.

Pages Scanned in LRU Scans

Pages Scanned Per Second while doing LRU scans. If this value is large (thousands) it means a lot of resources are wasted.

Pages scanned per LRU Scan

Number of pages scanned per LRU scan in Average. Large number of scans can consume a lot of resources and also introduce significant addition latency to queries.

LRU Get Free Waits

If InnoDB could not find a free page in LRU list and had to sleep. Should be zero.

InnoDB Checkpointing and Flushing

Pages Flushed from Flush List

Number of Pages Flushed from “Flush List” This combines Pages Flushed through Adaptive Flush and Background Flush.

Page Flush Batches Executed

InnoDB Flush Cycle typically Runs on 1 second intervals. If too far off from this number it can indicate an issue.

Pages Flushed Per Batch

Percona LLC and/or its affiliates ©2024

How many pages are flushed per Batch. Large Batches can “choke” IO subsystem and starve other IO which needs to happen.

Neighbor Flushing Enabled

Neighbor Flushing is Optimized for Rotational Media and unless you’re Running spinning disks you should disable it.

InnoDB Checkpoint Age

InnoDB Checkpoint Age

The maximum checkpoint age is determined by the total length of all transaction log files (`innodb_log_file_size`).

When the checkpoint age reaches the maximum checkpoint age, blocks are flushed synchronously. The rules of the thumb is to keep one hour of traffic in those logs and let the check-pointing perform its work as smooth as possible. If you don’t do this, InnoDB will do synchronous flushing at the worst possible time, i.e., when you are busiest.

Pages Flushed (Adaptive)

Adaptive Flush Flushes pages from Flush List based on the need to advance Checkpoint (driven by Redo Generation Rate) and by maintaining number of dirty pages within set limit.

Adaptive Flush Batches Executed Pages Per Batch (Adaptive)

Pages Flushed Per Adaptive Batch.

Neighbor Flushing

To optimize IO for rotational Media InnoDB may flush neighbor pages. It can cause significant wasted IO for flash storage. Generally for flash you should run with `innodb_flush_neighbors=0` but otherwise this shows how much IO you’re wasting.

Pages Flushed (LRU)

Flushing from the tail of the LRU list is needed to keep readily-available free pages for new data to be read when data does not fit in the buffer pool.

LRU Flush Batches Executed Pages Per Batch (LRU)

Pages Flushed Per Neighbor.

LSN Age Flush Batch Target

Target for Pages to Flush due to LSN Age.

Pages Flushed (Neighbor)

Number of Neighbor pages flushed (If neighbor flushing is enabled) from Flush List and LRU List Combined.

Neighbor Flush Batches Executed Pages Per Batch (Neighbor)

Pages Flushed Per Neighbor.

Sync Flush Waits

If InnoDB could not keep up with Checkpoint Flushing and had to trigger Sync flush. This should never happen.

Pages Flushed (Background)

Pages Flushed by Background Flush which is activated when server is considered to be idle.

Background Flush Batches Executed Pages Per Batch (Background)

Percona LLC and/or its affiliates ©2024

Pages Flushed Per Background Batch.

Redo Generation Rate

Rate at which LSN (Redo) is Created. It may not match how much data is written to log files due to block size rounding.

InnoDB Flushing by Type Pages Evicted (LRU)

This correspond to number of clean pages which were evicted (made free) from the tail of LRU buffer.

Page Eviction Batches Pages Evicted per Batch Max Log Space Used Single Page Flushes

Single Page flushes happen in rare case, then clean page could not be found in LRU list. It should be zero for most workloads.

Single Page Flush Pages Scanned Pages Scanned Per Single Page Flush InnoDB IO Capacity

Estimated number of IOPS storage system can provide. Is used to scale background activities. Do not set it to actual storage capacity.

InnoDB IO Capacity Max

InnoDB IO Capacity to use when falling behind and need to catch up with Flushing.

InnoDB Logging

Total Log Space

Number of InnoDB Log Files Multiplied by Their Size.

Log Buffer Size

InnoDB Log Buffer Size

The size of buffer InnoDB uses for buffering writes to log files.

At Transaction Commit

What to do with Log file At Transaction Commit. Do nothing and wait for timeout to flush the data from Log Buffer, Flush it to OS Cache but not FSYNC or Flush only.

Flush Transaction Log Every

Every Specified Number of Seconds Flush Transaction Log.

InnoDB Write Ahead Block Size

This variable can be seen as minimum IO alignment InnoDB will use for Redo log file. High Values cause waste, low values can make IO less efficient.

Log Write Amplification

How much Writes to Log Are Amplified compared to how much Redo is Generated.

Log Fsync Rate Redo Generated per Trx

Amount of Redo Generated Per Write Transaction. This is a good indicator of transaction size.

InnoDB Log File Usage Hourly

InnoDB Log File Usage Hourly

Percona LLC and/or its affiliates ©2024

Along with the buffer pool size, `innodb_log_file_size` is the most important setting when we are working with InnoDB. This graph shows how much data was written to InnoDB's redo logs over each hour. When the InnoDB log files are full, InnoDB needs to flush the modified pages from memory to disk.

The rules of the thumb is to keep one hour of traffic in those logs and let the checkpointing perform its work as smooth as possible. If you don't do this, InnoDB will do synchronous flushing at the worst possible time, i.e., when you are busiest.

This graph can help guide you in setting the correct `innodb_log_file_size`.

Log Padding Written

Amount of Log Padding Written.

InnoDB Log File Size InnoDB Log Files

Number of InnoDB Redo Log Files.

Log Bandwidth Redo Generation Rate

Rate at which LSN (Redo) is Created. It may not match how much data is written to log files due to block size rounding.

InnoDB Group Commit Batch Size

The InnoDB Group Commit Batch Size graph shows how many bytes were written to the InnoDB log files per attempt to write. If many threads are committing at the same time, one of them will write the log entries of all the waiting threads and flush the file. Such process reduces the number of disk operations needed and enlarge the batch size.

InnoDB Locking

Lock Wait Timeout

InnoDB Lock Wait Timeout

How long to wait for row lock before timing out.

InnoDB Deadlock Detection

If Disabled InnoDB Will not detect deadlocks but rely on timeouts.

InnoDB Auto Increment Lock Mode

Will Define How much locking will come from working with Auto Increment Columns.

Rollback on Timeout

Whenever to rollback all transaction on timeout or just last statement.

Row Lock Blocking

Percent of Active Sections which are blocked due to waiting on InnoDB Row Locks.

Row Writes per Trx

Rows Written Per Transactions which modify rows. This is better indicator of transaction write size than looking at all transactions which did not do any writes as well.

Rollbacks

Percent of Transaction Rollbacks (as portion of read-write transactions).

InnoDB Row Lock Wait Activity InnoDB Row Lock Wait Time InnoDB Row Lock Wait Load

Percona LLC and/or its affiliates ©2024

Average Number of Sessions blocked from proceeding due to waiting on row level lock.

InnoDB Row Locks Activity InnoDB Table Lock Activity Current Locks

InnoDB Undo Space and Purging

Undo Tablespaces Max Undo Log Size InnoDB Undo Log Truncate Purge Threads Max Purge Lag

Maximum number of Unpurged Transactions, if this number exceeded delay will be introduced to incoming DDL statements.

Max Purge Lag Delay Current Purge Delay

The Delay Injected due to Purge Thread(s) unable to keep up with purge progress.

Rollback Segments InnoDB Purge Activity

The InnoDB Purge Performance graph shows metrics about the page purging process. The purge process removed the undo entries from the history list and cleanup the pages of the old versions of modified rows and effectively remove deleted rows.

Transactions and Undo Records InnoDB Undo Space Usage

The InnoDB Undo Space Usage graph shows the amount of space used by the Undo segment. If the amount of space grows too much, look for long running transactions holding read views opened in the InnoDB status.

Transaction History InnoDB Purge Throttling Records Per Undo Log Page

How Many Undo Operations Are Handled Per Each Undo Log Page.

Purge Invoked

How Frequently Purge Operation is Invoked.

Ops Per Purge

Home Many Purge Actions are done Per invocation.

Undo Slots Used

Number of Undo Slots Used.

Max Transaction History Length Purge Batch Size Rseg Truncate Frequency

InnoDB Page Operations

InnoDB Page Splits and Merges

The InnoDB Page Splits graph shows the InnoDB page maintenance activity related to splitting and merging pages. When an InnoDB page, other than the top most leaf page, has too much data to accept a row update or a row insert, it has to be split in two. Similarly, if an InnoDB page, after a row update or delete operation, ends up being less than half full, an attempt is made to merge the page with a neighbor page. If the resulting page size is larger than the InnoDB page size, the operation fails. If your workload causes a large number of page splits, try lowering the `innodb_fill_factor` variable (5.7+).

Page Merge Success Ratio InnoDB Page Reorg Attempts

The InnoDB Page Reorgs graph shows information about the page reorganization operations. When a page receives an update or an insert that affect the offset of other rows in the page, a reorganization is needed. If the reorganization process finds out there is not enough room in the page, the page will be split. Page reorganization can only fail for compressed pages.

InnoDB Page Reorgs Failures

Percona LLC and/or its affiliates ©2024

The InnoDB Page Reorgs graph shows information about the page reorganization operations. When a page receives an update or an insert that affect the offset of other rows in the page, a reorganization is needed. If the reorganization process finds out there is not enough room in the page, the page will be split. Page reorganization can only fail for compressed pages.

InnoDB Fill Factor

The portion of the page to fill then doing sorted Index Build. Lowering this value will worsen space utilization but will reduce need to split pages when new data is inserted in the index.

InnoDB Adaptive Hash Index

Adaptive Hash Index Enabled

Adaptive Hash Index helps to optimize index Look-ups but can be severe hotspot for some workloads.

Adaptive Hash Index Partitions

How many Partitions Used for Adaptive Hash Index (to reduce contention).

Percent of Pages Hashed

Number of Pages Added to AHI vs Number of Pages Added to Buffer Pool.

AHI Miss Ratio

Percent of Searches which could not be resolved through AHI.

Rows Added Per Page

Number of Rows “Hashed” Per Each Page which needs to be added to AHI.

AHI ROI

How Many Successful Searches using AHI are performed per each row maintenance operation.

InnoDB AHI Usage

The InnoDB AHI Usage graph shows the search operations on the InnoDB adaptive hash index and its efficiency. The adaptive hash index is a search hash designed to speed access to InnoDB pages in memory. If the Hit Ratio is small, the working data set is larger than the buffer pool, the AHI should likely be disabled.

InnoDB AHI Miss Ratio InnoDB AHI Churn - Rows InnoDB AHI Churn - Pages

InnoDB Change Buffer

Change Buffer Max Size

The Maximum Size of Change Buffer (as Percent of Buffer Pool Size).

Change Buffer Max Size

The Maximum Size of Change Buffer (Bytes).

InnoDB Change Buffer Merge Load

Number of Average of Active Merge Buffer Operations in Process.

InnoDB Contention

InnoDB Thread Concurrency

If Enabled limits number of Threads allowed inside InnoDB Kernel at the same time.

Percona LLC and/or its affiliates ©2024

InnoDB Commit Concurrency

If Enabled limits number of Threads allowed inside InnoDB Kernel at the same time during Commit Stage.

InnoDB Thread Sleep Delay

The Time the thread will Sleep before Re-Entering InnoDB Kernel if high contention.

InnoDB Adaptive Max Sleep Delay

If Set to Non-Zero Value InnoDB Thread Sleep Delay will be adjusted automatically depending on the load up to the value specified by this variable.

InnoDB Concurrency Tickets

Number of low level operations InnoDB can do after it entered InnoDB kernel before it is forced to exit and yield to another thread waiting.

InnoDB Spin Wait Delay InnoDB Spin Wait Pause Multiplier InnoDB Sync Spin Loops InnoDB Contention - OS Waits

The InnoDB Contention - OS Waits graph shows the number of time an OS wait operation was required while waiting to get the lock. This happens once the spin rounds are exhausted.

InnoDB Contention - Spin Rounds

The InnoDB Contention - Spin Rounds graph shows the number of spin rounds executed to get a lock. A spin round is a fast retry to get the lock in a loop.

InnoDB Misc

InnoDB Main Thread Utilization

The InnoDB Main Thread Utilization graph shows the portion of time the InnoDB main thread spent at various task.

InnoDB Activity

The InnoDB Activity graph shows a measure of the activity of the InnoDB threads.

InnoDB Dedicated Server

InnoDB automatically optimized for Dedicated Server Environment (auto scaling cache and some other variables).

InnoDB Sort Buffer Size

This Buffer is used for Building InnoDB Indexes using Sort algorithm.

InnoDB Stats Auto Recalc Update Stats when Metadata Queried

Refresh InnoDB Statistics when meta-data queries by `SHOW TABLE STATUS` or `INFORMATION_SCHEMA` queries. If Enabled can cause severe performance issues.

Index Condition Pushdown (ICP)

Index Condition Pushdown (ICP) is an optimization for the case where MySQL retrieves rows from a table using an index. Without ICP, the storage engine traverses the index to locate rows in the base table and returns them to the MySQL server which evaluates the WHERE condition for the rows. With ICP enabled, and if parts of the WHERE condition can be evaluated by using only columns from the index, the MySQL server pushes this part of the WHERE condition down to the storage engine. The storage engine then evaluates the pushed index condition by using the index entry and only if this is satisfied is the row read from the table. ICP can reduce the number of times the storage engine must access the base table and the number of times the MySQL server must access the storage engine.

InnoDB Persistent Statistics InnoDB Persistent Sample Pages

Percona LLC and/or its affiliates ©2024

Number of Pages To Sample if Persistent Statistics are Enabled.

InnoDB Transient Sample Pages

Number of Pages To Sample if Persistent Statistics are Disabled.

InnoDB Online Operations (MariaDB)

InnoDB Defragmentation

The InnoDB Defragmentation graph shows the status information related to the InnoDB online defragmentation feature of MariaDB for the optimize table command. To enable this feature, the variable `innodb-defragment` must be set to 1 in the configuration file.

> Currently available only on a MariaDB server.

InnoDB Online DDL

The InnoDB Online DDL graph shows the state of the online DDL (alter table) operations in InnoDB. The progress metric is estimate of the percentage of the rows processed by the online DDL.

> Currently available only on a MariaDB server.

MySQL Summary

MySQL Uptime

MySQL Uptime

The amount of time since the last restart of the MySQL server process.

Current QPS

Current QPS

Based on the queries reported by MySQL's `SHOW STATUS` command, it is the number of statements executed by the server within the last second. This variable includes statements executed within stored programs, unlike the Questions variable. It does not count `COM_PING` or `COM_STATISTICS` commands.

File Handlers Used Table Open Cache Miss Ratio Table Open Cache Size Table Definition Cache Size MySQL Connections

Max Connections

Max Connections is the maximum permitted number of simultaneous client connections. By default, this is 151. Increasing this value increases the number of file descriptors that mysqld requires. If the required number of descriptors are not available, the server reduces the value of Max Connections.

mysqld actually permits Max Connections + 1 clients to connect. The extra connection is reserved for use by accounts that have the SUPER privilege, such as root.

Max Used Connections is the maximum number of connections that have been in use simultaneously since the server started.

Connections is the number of connection attempts (successful or not) to the MySQL server.

MySQL Client Thread Activity

MySQL Active Threads

Percona LLC and/or its affiliates ©2024

Threads Connected is the number of open connections, while Threads Running is the number of threads not sleeping.

MySQL Handlers

MySQL Handlers

Handler statistics are internal statistics on how MySQL is selecting, updating, inserting, and modifying rows, tables, and indexes.

This is in fact the layer between the Storage Engine and MySQL.

- `read_rnd_next` is incremented when the server performs a full table scan and this is a counter you don't really want to see with a high value.
- `read_key` is incremented when a read is done with an index.
- `read_next` is incremented when the storage engine is asked to 'read the next index entry'. A high value means a lot of index scans are being done.

Top Command Counters

Top Command Counters

The `Com_{{ xxx }}` statement counter variables indicate the number of times each `xxx` statement has been executed. There is one status variable for each type of statement. For example, `Com_delete` and `Com_update` count `DELETE` and `UPDATE` statements, respectively. `Com_delete_multi` and `Com_update_multi` are similar but apply to `DELETE` and `UPDATE` statements that use multiple-table syntax.

MySQL Network Traffic

MySQL Network Traffic

Here we can see how much network traffic is generated by MySQL. Outbound is network traffic sent from MySQL and Inbound is network traffic MySQL has received.

Node Summary

System Uptime

The parameter shows how long a system has been up and running without a shut down or restart.

Load Average

The system load is a measurement of the computational work the system is performing. Each running process either using or waiting for CPU resources adds 1 to the load.

RAM

RAM (Random Access Memory) is the hardware in a computing device where the operating system, application programs and data in current use are kept so they can be quickly reached by the device's processor.

Memory Available

Percent of Memory Available

> On Modern Linux Kernels amount of Memory Available for application is not the same as Free+Cached+Buffers.

Virtual Memory

RAM + SWAP

Disk Space

Percona LLC and/or its affiliates ©2024

Sum of disk space on all partitions.

> It can be significantly over-reported in some installations.

Min Space Available

Lowest percent of the disk space available.

CPU Usage

The CPU time is measured in clock ticks or seconds. It is useful to measure CPU time as a percentage of the CPU's capacity, which is called the CPU usage.

CPU Saturation and Max Core Usage

When a system is running with maximum CPU utilization, the transmitting and receiving threads must all share the available CPU. This will cause data to be queued more frequently to cope with the lack of CPU. CPU Saturation may be measured as the length of a wait queue, or the time spent waiting on the queue.

Disk I/O and Swap Activity

Disk I/O includes read or write or input/output operations involving a physical disk. It is the speed with which the data transfer takes place between the hard disk drive and RAM.

Swap Activity is memory management that involves swapping sections of memory to and from physical storage.

Network Traffic

Network traffic refers to the amount of data moving across a network at a given point in time.

Get expert help

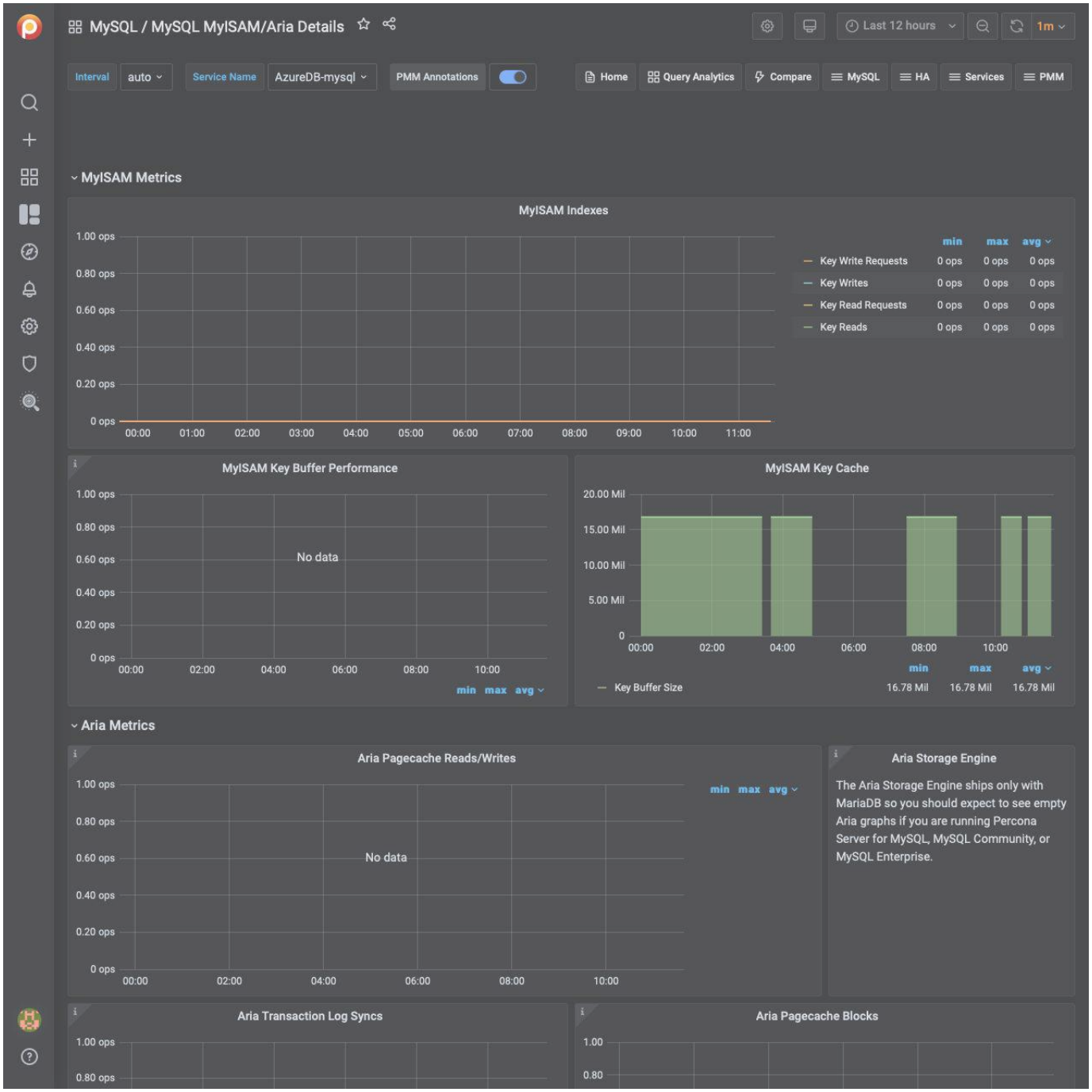
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

Percona LLC and/or its affiliates ©2024

MYSQL MYISAM/ARIA DETAILS



MyISAM Key Buffer Performance

The `Key Read Ratio` (`Key_reads` / `Key_read_requests`) ratio should normally be less than 0.01.

The `Key Write Ratio` (`Key_writes` / `Key_write_requests`) ratio is usually near 1 if you are using mostly updates and deletes, but might be much smaller if you tend to do updates that affect many rows at the same time or if you are using the `DELAY_KEY_WRITE` table option.

Aria Pagecache Reads/Writes

This graph is similar to InnoDB buffer pool reads/writes. `aria-pagecache-buffer-size` is the main cache for the Aria storage engine. If you see high reads/writes (physical IO), i.e. reads are close to read requests and/or writes are

Percona LLC and/or its affiliates ©2024

close to write requests you may need to increase the `aria-pagecache-buffer-size` (may need to decrease other buffers: `key_buffer_size`, `innodb_buffer_pool_size`, etc.)

Aria Transaction Log Syncs

This is similar to InnoDB log file syncs. If you see lots of log syncs and want to relax the durability settings you can change `aria_checkpoint_interval` (in seconds) from 30 (default) to a higher number. It is good to look at the disk IO dashboard as well.

Aria Pagecache Blocks

This graph shows the utilization for the Aria pagecache. This is similar to InnoDB buffer pool graph. If you see all blocks are used you may consider increasing `aria-pagecache-buffer-size` (may need to decrease other buffers: `key_buffer_size`, `innodb_buffer_pool_size`, etc.)

Get expert help

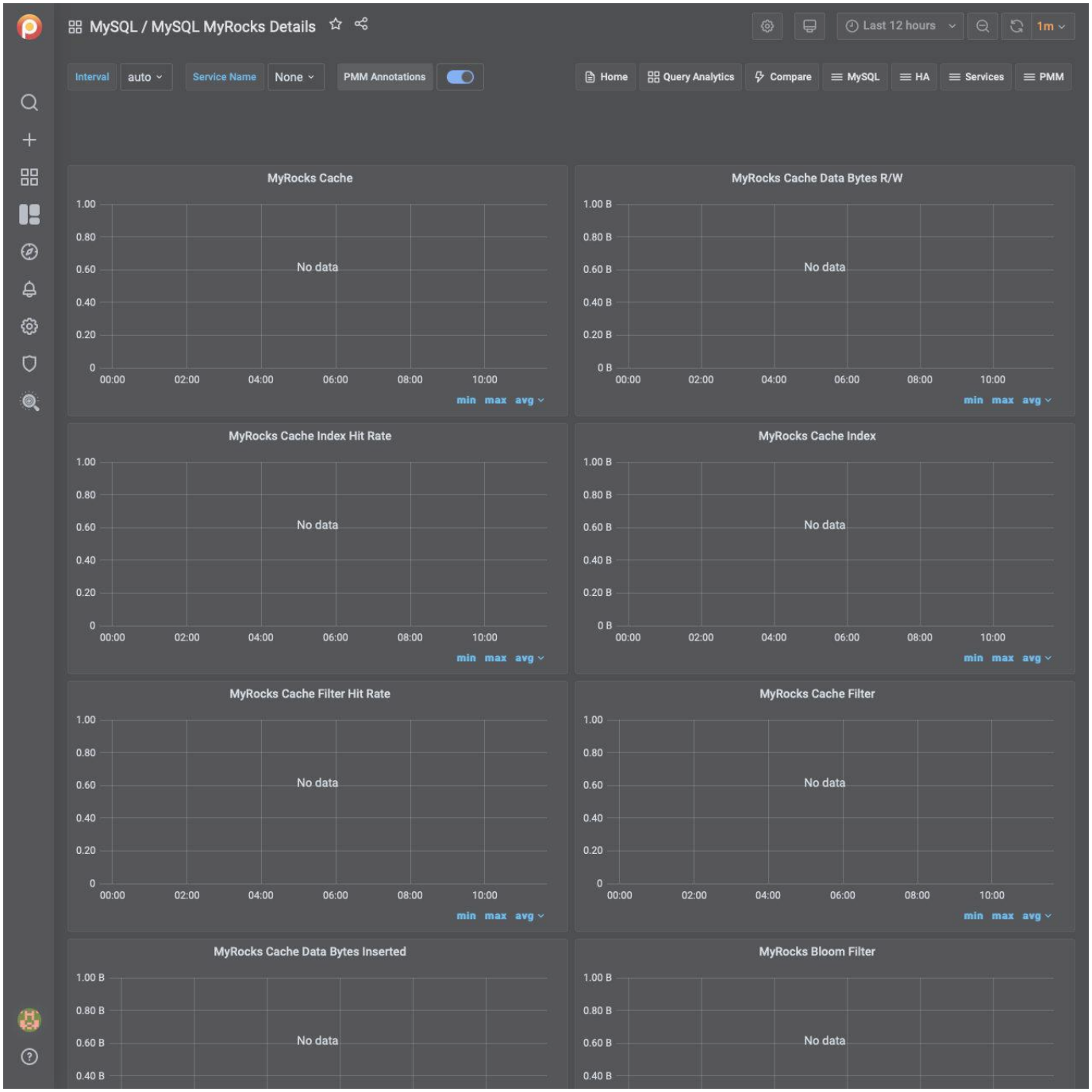
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

MYSQL MYROCKS DETAILS



The MyRocks storage engine developed by Facebook based on the RocksDB storage engine is applicable to systems which primarily interact with the database by writing data to it rather than reading from it. RocksDB also features a good level of compression, higher than that of the InnoDB storage engine, which makes it especially valuable when optimizing the usage of hard drives.

PMM collects statistics on the MyRocks storage engine for MySQL in the Metrics Monitor information for this dashboard comes from the *Information Schema* tables.

Metrics

- MyRocks cache
- MyRocks cache data bytes R/W

Percona LLC and/or its affiliates ©2024

- MyRocks cache index hit rate
- MyRocks cache index
- MyRocks cache filter hit rate
- MyRocks cache filter
- MyRocks cache data bytes inserted
- MyRocks bloom filter
- MyRocks memtable
- MyRocks memtable size
- MyRocks number of keys
- MyRocks cache L0/L1
- MyRocks number of DB ops
- MyRocks R/W
- MyRocks bytes read by iterations
- MyRocks write ops
- MyRocks WAL
- MyRocks number reseeks in iterations
- RocksDB row operations
- MyRocks file operations
- RocksDB stalls
- RocksDB stops/slowdowns

Get expert help

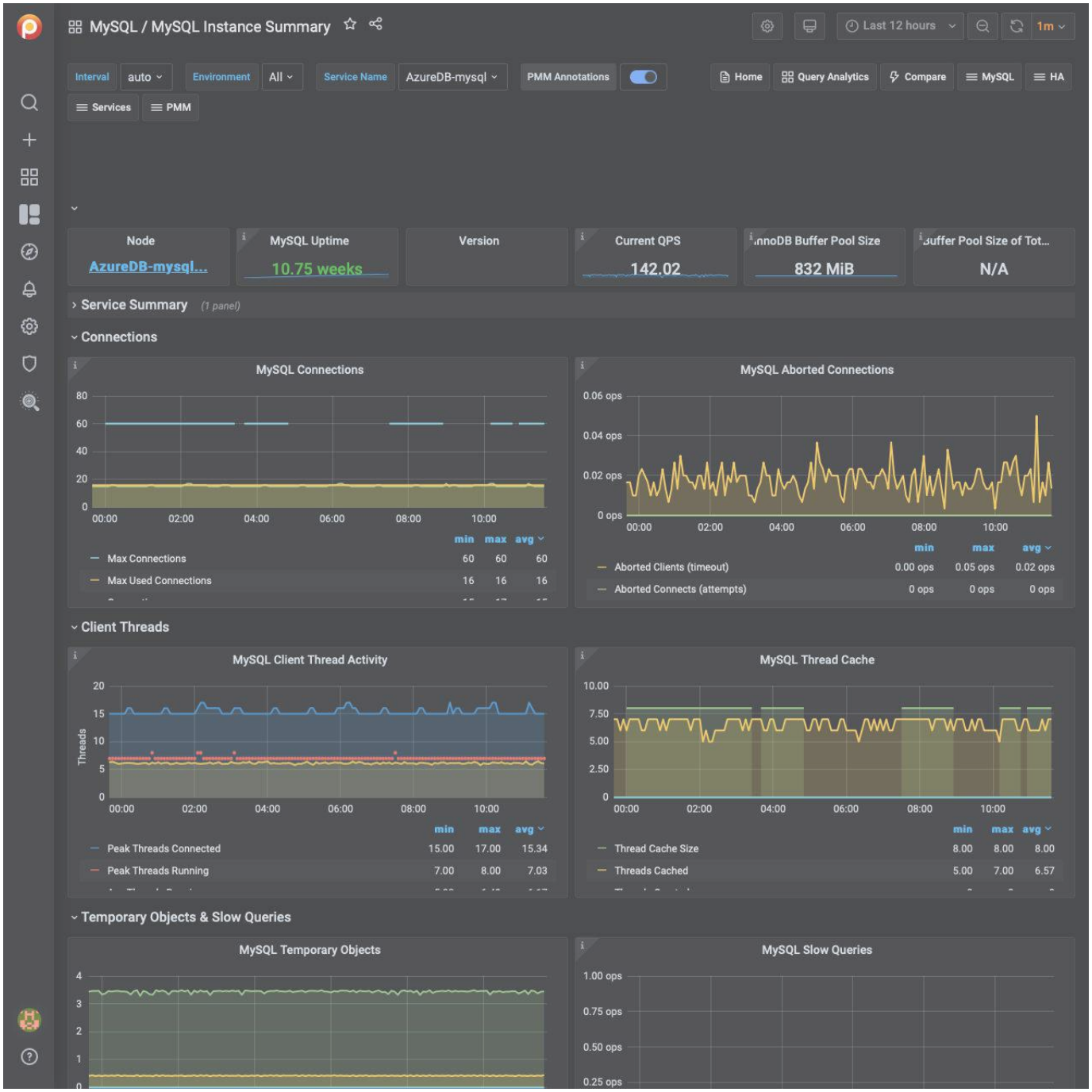
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

Percona LLC and/or its affiliates ©2024

MYSQL INSTANCE SUMMARY



MySQL Connections

Max Connections

Max Connections is the maximum permitted number of simultaneous client connections. By default, this is 151. Increasing this value increases the number of file descriptors that `mysqld` requires. If the required number of descriptors are not available, the server reduces the value of Max Connections.

`mysqld` actually permits Max Connections + 1 clients to connect. The extra connection is reserved for use by accounts that have the SUPER privilege, such as root.

Max Used Connections is the maximum number of connections that have been in use simultaneously since the server started.

Percona LLC and/or its affiliates ©2024

Connections is the number of connection attempts (successful or not) to the MySQL server.

MySQL Aborted Connections

Aborted Connections

When a given host connects to MySQL and the connection is interrupted in the middle (for example due to bad credentials), MySQL keeps that info in a system table (since 5.6 this table is exposed in `performance_schema`).

If the amount of failed requests without a successful connection reaches the value of `max_connect_errors`, `mysqld` assumes that something is wrong and blocks the host from further connection.

To allow connections from that host again, you need to issue the `FLUSH HOSTS` statement.

MySQL Client Thread Activity

MySQL Active Threads

Threads Connected is the number of open connections, while Threads Running is the number of threads not sleeping.

MySQL Thread Cache

MySQL Thread Cache

The `thread_cache_size` variable sets how many threads the server should cache to reuse. When a client disconnects, the client's threads are put in the cache if the cache is not full. It is auto-sized in MySQL 5.6.8 and above (capped to 100). Requests for threads are satisfied by reusing threads taken from the cache if possible, and only when the cache is empty is a new thread created.

- `threads_created`: The number of threads created to handle connections.
- `threads_cached`: The number of threads in the thread cache.

MySQL Slow Queries

MySQL Slow Queries

Slow queries are defined as queries being slower than the `long_query_time` setting. For example, if you have `long_query_time` set to 3, all queries that take longer than 3 seconds to complete will show on this graph.

MySQL Select Types

MySQL Select Types

As with most relational databases, selecting based on indexes is more efficient than scanning an entire table's data. Here we see the counters for selects not done with indexes.

- ***Select Scan*** is how many queries caused full table scans, in which all the data in the table had to be read and either discarded or returned.
- ***Select Range*** is how many queries used a range scan, which means MySQL scanned all rows in a given range.
- ***Select Full Join*** is the number of joins that are not joined on an index, this is usually a huge performance hit.

MySQL Sorts

MySQL Sorts

Due to a query's structure, order, or other requirements, MySQL sorts the rows before returning them. For example, if a table is ordered 1 to 10 but you want the results reversed, MySQL then has to sort the rows to return 10 to 1.

This graph also shows when sorts had to scan a whole table or a given range of a table to return the results and which could not have been sorted via an index.

Percona LLC and/or its affiliates ©2024

MySQL Table Locks

Table Locks

MySQL takes a number of different locks for varying reasons. In this graph we see how many Table level locks MySQL has requested from the storage engine. In the case of InnoDB, many times the locks could actually be row locks as it only takes table level locks in a few specific cases.

It is most useful to compare Locks Immediate and Locks Waited. If Locks waited is rising, it means you have lock contention. Otherwise, Locks Immediate rising and falling is normal activity.

MySQL Questions

MySQL Questions

The number of statements executed by the server. This includes only statements sent to the server by clients and not statements executed within stored programs, unlike the Queries used in the QPS calculation.

This variable does not count the following commands:

- `COM_PING`
- `COM_STATISTICS`
- `COM_STMT_PREPARE`
- `COM_STMT_CLOSE`
- `COM_STMT_RESET`

MySQL Network Traffic

MySQL Network Traffic

Here we can see how much network traffic is generated by MySQL. Outbound is network traffic sent from MySQL and Inbound is network traffic MySQL has received.

MySQL Network Usage Hourly

MySQL Network Usage Hourly

Here we can see how much network traffic is generated by MySQL per hour. You can use the bar graph to compare data sent by MySQL and data received by MySQL.

MySQL Internal Memory Overview

System Memory: Total Memory for the system.

InnoDB Buffer Pool Data: InnoDB maintains a storage area called the buffer pool for caching data and indexes in memory.

TokuDB Cache Size: Similar in function to the InnoDB Buffer Pool, TokuDB will allocate 50% of the installed RAM for its own cache.

Key Buffer Size: Index blocks for MyISAM tables are buffered and are shared by all threads. `key_buffer_size` is the size of the buffer used for index blocks.

Adaptive Hash Index Size: When InnoDB notices that some index values are being accessed very frequently, it builds a hash index for them in memory on top of B-Tree indexes.

Query Cache Size: The query cache stores the text of a SELECT statement together with the corresponding result that was sent to the client. The query cache has huge scalability problems in that only one thread can do an operation in the query cache at the same time.

Percona LLC and/or its affiliates ©2024

InnoDB Dictionary Size: The data dictionary is InnoDB's internal catalog of tables. InnoDB stores the data dictionary on disk, and loads entries into memory while the server is running.

InnoDB Log Buffer Size: The MySQL InnoDB log buffer allows transactions to run without having to write the log to disk before the transactions commit.

Top Command Counters

Top Command Counters

The `Com_xxx` statement counter variables indicate the number of times each `xxx` statement has been executed. There is one status variable for each type of statement. For example, `Com_delete` and `Com_update` count DELETE and UPDATE statements, respectively. `Com_delete_multi` and `Com_update_multi` are similar but apply to DELETE and UPDATE statements that use multiple-table syntax.

Top Command Counters Hourly

Top Command Counters Hourly

The `Com_xxx` statement counter variables indicate the number of times each `xxx` statement has been executed. There is one status variable for each type of statement. For example, `Com_delete` and `Com_update` count DELETE and UPDATE statements, respectively. `Com_delete_multi` and `Com_update_multi` are similar but apply to DELETE and UPDATE statements that use multiple-table syntax.

MySQL Handlers

MySQL Handlers

Handler statistics are internal statistics on how MySQL is selecting, updating, inserting, and modifying rows, tables, and indexes.

This is in fact the layer between the Storage Engine and MySQL.

- `read_rnd_next` is incremented when the server performs a full table scan and this is a counter you don't really want to see with a high value.
- `read_key` is incremented when a read is done with an index.
- `read_next` is incremented when the storage engine is asked to 'read the next index entry'. A high value means a lot of index scans are being done.

MySQL Query Cache Memory

MySQL Query Cache Memory

The query cache has huge scalability problems in that only one thread can do an operation in the query cache at the same time. This serialization is true not only for SELECTs, but also for INSERT/UPDATE/DELETE.

This also means that the larger the `query_cache_size` is set to, the slower those operations become. In concurrent environments, the MySQL Query Cache quickly becomes a contention point, decreasing performance. MariaDB and AWS Aurora have done work to try and eliminate the query cache contention in their flavors of MySQL, while MySQL 8.0 has eliminated the query cache feature.

The recommended settings for most environments is to set:

- `query_cache_type=0`
- `query_cache_size=0`

> **Tip**
>
> While you can dynamically change these values, to completely remove the contention point you have to restart the database.

Percona LLC and/or its affiliates ©2024

MySQL Query Cache Activity

MySQL Query Cache Activity

The query cache has huge scalability problems in that only one thread can do an operation in the query cache at the same time. This serialization is true not only for SELECTs, but also for INSERT/UPDATE/DELETE.

This also means that the larger the `query_cache_size` is set to, the slower those operations become. In concurrent environments, the MySQL Query Cache quickly becomes a contention point, decreasing performance. MariaDB and AWS Aurora have done work to try and eliminate the query cache contention in their flavors of MySQL, while MySQL 8.0 has eliminated the query cache feature.

The recommended settings for most environments is to set:

- `query_cache_type=0`
- `query_cache_size=0`

> **Tip**
>
> While you can dynamically change these values, to completely remove the contention point you have to restart the database.

MySQL Table Open Cache Status

MySQL Table Open Cache Status

The recommendation is to set the `table_open_cache_instances` to a loose correlation to virtual CPUs, keeping in mind that more instances means the cache is split more times. If you have a cache set to 500 but it has 10 instances, each cache will only have 50 cached.

The `table_definition_cache` and `table_open_cache` can be left as default as they are auto-sized MySQL 5.6 and above (i.e., do not set them to any value).

MySQL Open Tables

MySQL Open Tables

The recommendation is to set the `table_open_cache_instances` to a loose correlation to virtual CPUs, keeping in mind that more instances means the cache is split more times. If you have a cache set to 500 but it has 10 instances, each cache will only have 50 cached.

The `table_definition_cache` and `table_open_cache` can be left as default as they are auto-sized MySQL 5.6 and above (i.e., do not set them to any value).

MySQL Table Definition Cache

MySQL Table Definition Cache

The recommendation is to set the `table_open_cache_instances` to a loose correlation to virtual CPUs, keeping in mind that more instances means the cache is split more times. If you have a cache set to 500 but it has 10 instances, each cache will only have 50 cached.

The `table_definition_cache` and `table_open_cache` can be left as default as they are auto-sized MySQL 5.6 and above (i.e., do not set them to any value).

Get expert help

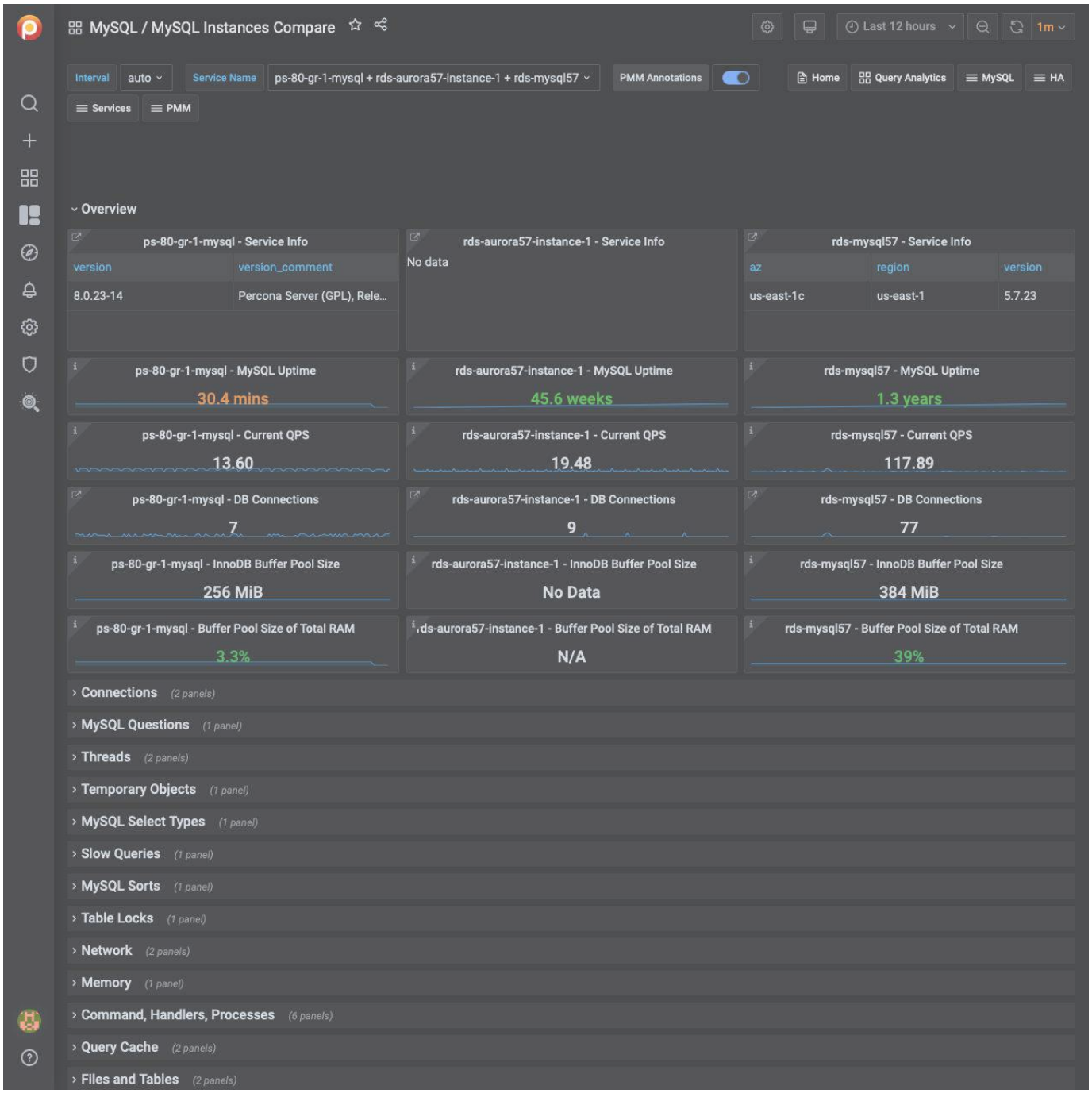
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2021-06-25

MYSQL INSTANCES COMPARE



No description

Percona LLC and/or its affiliates ©2024

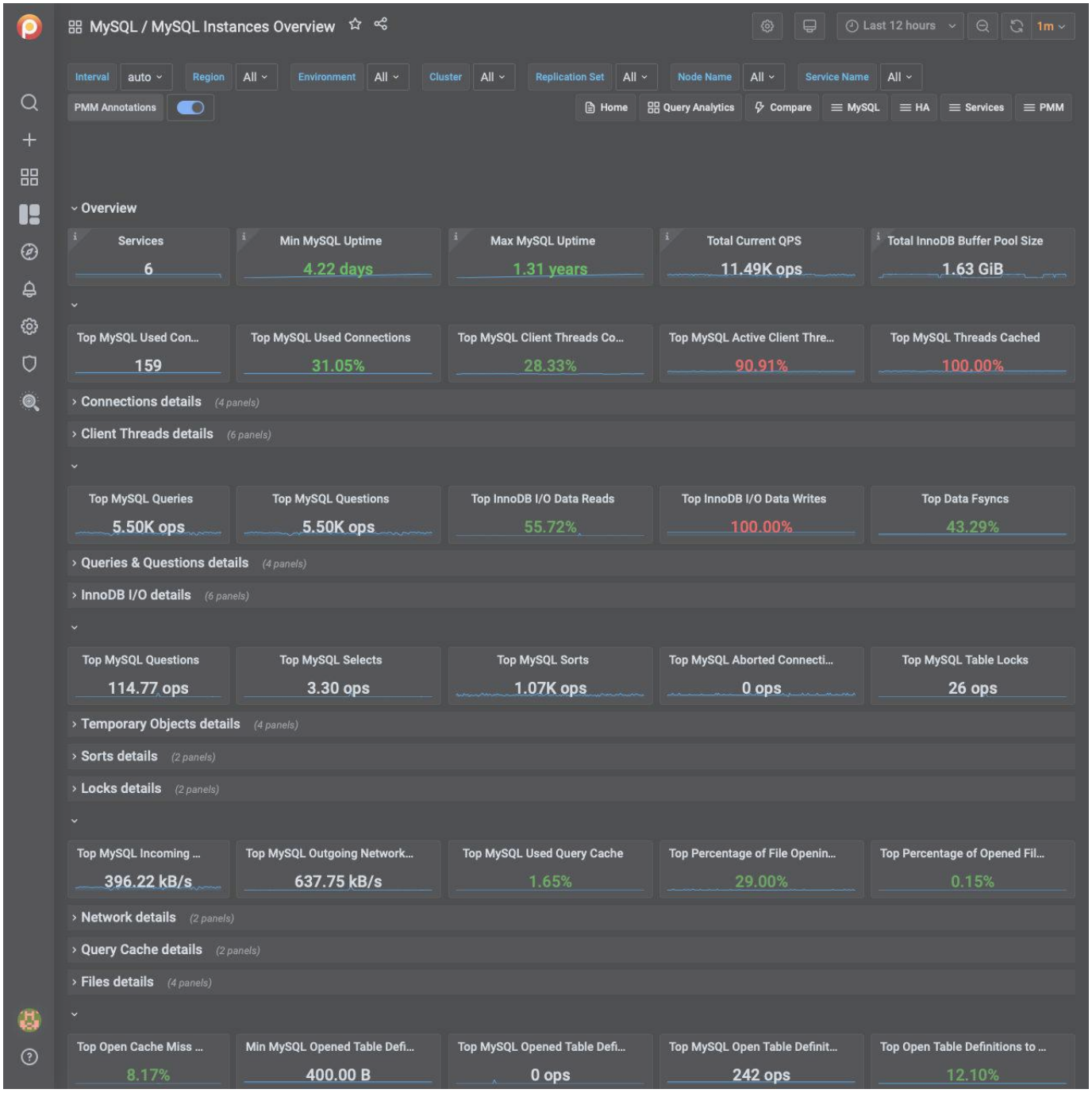
Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

MYSQL INSTANCES OVERVIEW



No description

Percona LLC and/or its affiliates ©2024

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-03

MYSQL WAIT EVENT ANALYSES DETAILS

Percona LLC and/or its affiliates ©2024

This dashboard helps to analyze *Performance Schema* wait events. It plots the following metrics for the chosen (one or more) wait events:

- Count - Performance Schema Waits
- Load - Performance Schema Waits
- Avg Wait Time - Performance Schema Waits

Get expert help

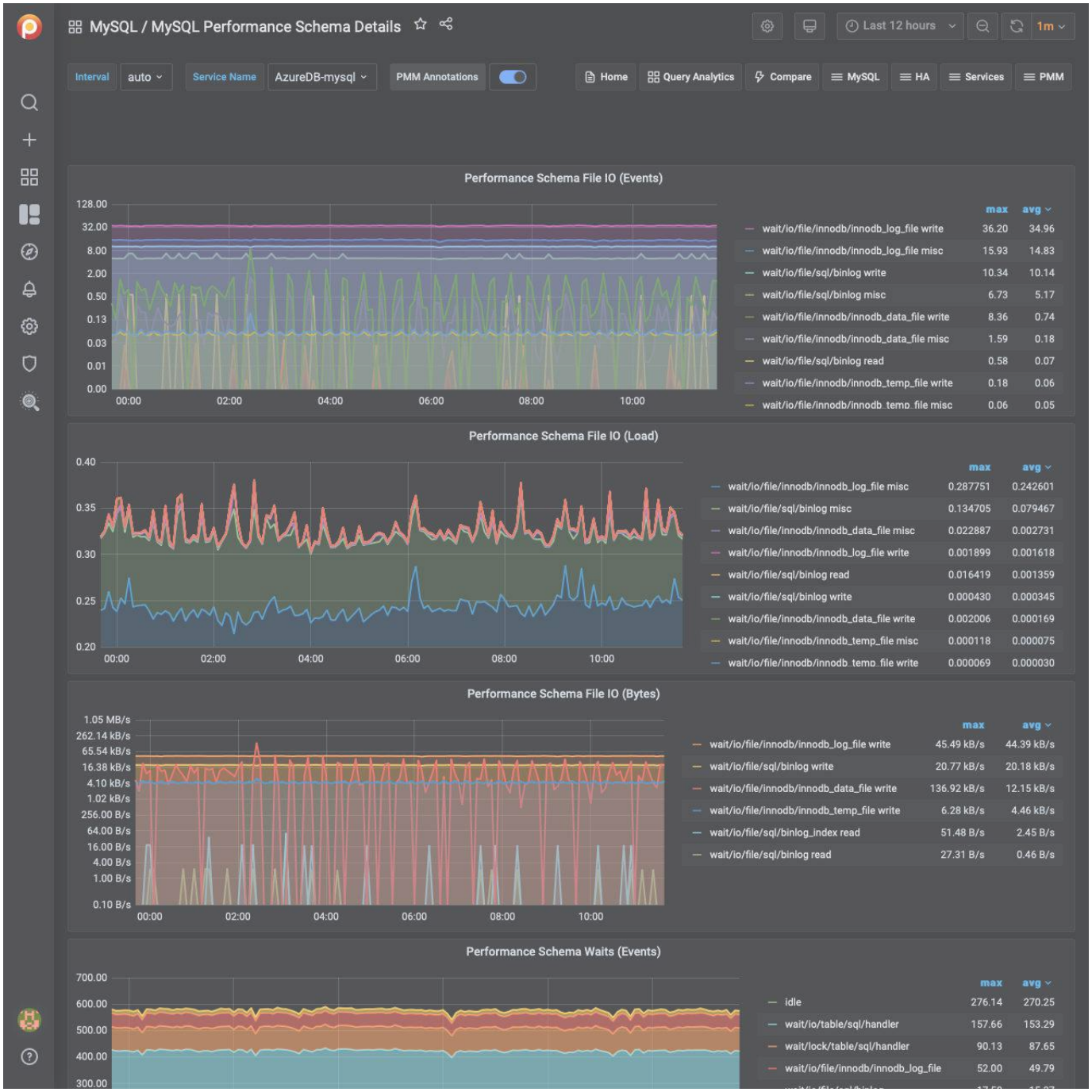
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

Percona LLC and/or its affiliates ©2024

MYSQL PERFORMANCE SCHEMA DETAILS



Percona LLC and/or its affiliates ©2024

The MySQL Performance Schema dashboard helps determine the efficiency of communicating with Performance Schema. This dashboard contains the following metrics:

- Performance Schema file IO (events)
- Performance Schema file IO (load)
- Performance Schema file IO (Bytes)
- Performance Schema waits (events)
- Performance Schema waits (load)
- Index access operations (load)
- Table access operations (load)
- Performance Schema SQL and external locks (events)
- Performance Schema SQL and external locks (seconds)

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

Percona LLC and/or its affiliates ©2024

MYSQL QUERY RESPONSE TIME DETAILS

Average Query Response Time

The Average Query Response Time graph shows information collected using the Response Time Distribution plugin sourced from table `INFORMATION_SCHEMA.QUERY_RESPONSE_TIME`. It computes this value across all queries by taking the sum of seconds divided by the count of queries.

Percona LLC and/or its affiliates ©2024

Query Response Time Distribution

Query response time counts (operations) are grouped into three buckets:

- 100 ms - 1 s
- 1 s - 10 s
- > 10 s

Average Query Response Time

Available only in Percona Server for MySQL, provides visibility of the split of READ vs WRITE query response time.

Read Query Response Time Distribution

Available only in Percona Server for MySQL, illustrates READ query response time counts (operations) grouped into three buckets:

- 100 ms - 1 s
- 1 s - 10 s
- > 10 s

Write Query Response Time Distribution

Available only in Percona Server for MySQL, illustrates WRITE query response time counts (operations) grouped into three buckets:

- 100 ms - 1 s
- 1 s - 10 s
- > 10 s

Get expert help

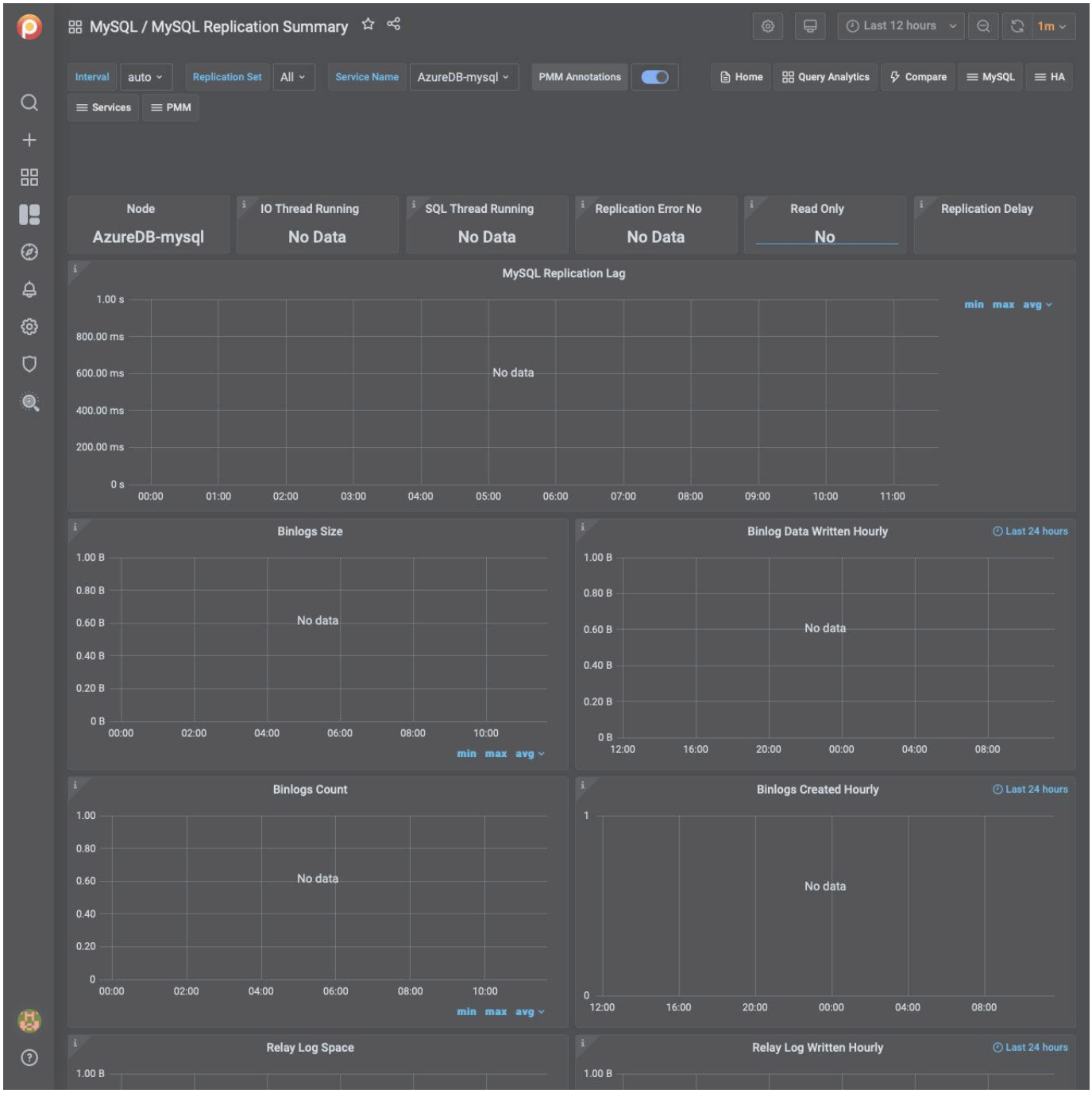
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

Percona LLC and/or its affiliates ©2024

MYSQL REPLICATION SUMMARY



IO Thread Running

This metric shows if the IO Thread is running or not. It only applies to a secondary host.

SQL Thread is a process that runs on a secondary host in the replication environment. It reads the events from the local relay log file and applies them to the secondary server.

Depending on the format of the binary log it can read query statements in plain text and re-execute them or it can read raw data and apply them to the local host.

Possible values

Percona LLC and/or its affiliates ©2024

Yes

The thread is running and is connected to a replication primary

No

The thread is not running because it is not launched yet or because an error has occurred connecting to the primary host

Connecting

The thread is running but is not connected to a replication primary

No value

The host is not configured to be a replication secondary

IO Thread Running is one of the parameters that the command `SHOW SLAVE STATUS` returns.

SQL Thread Running

This metric shows if the SQL thread is running or not. It only applies to a secondary host.

Possible values

Yes

SQL Thread is running and is applying events from the relay log to the local secondary host

No

SQL Thread is not running because it is not launched yet or because of an error occurred while applying an event to the local secondary host

Replication Error No

This metric shows the number of the last error in the SQL Thread encountered which caused replication to stop.

One of the more common errors is *Error: 1022 Duplicate Key Entry*. In such a case replication is attempting to update a row that already exists on the secondary. The SQL Thread will stop replication to avoid data corruption.

Read only

This metric indicates whether the host is configured to be in *Read Only* mode or not.

Possible values

Yes

The secondary host permits no client updates except from users who have the SUPER privilege or the REPLICATION SLAVE privilege.

This kind of configuration is typically used for secondary hosts in a replication environment to avoid a user can inadvertently or voluntarily modify data causing inconsistencies and stopping the replication process.

No

The secondary host is not configured in *Read Only* mode.

Percona LLC and/or its affiliates ©2024

MySQL Replication Delay

This metric shows the number of seconds the secondary host is delayed in replication applying events compared to when the primary host applied them, denoted by the `Seconds_Behind_Master` value, and only applies to a secondary host.

Since the replication process applies the data modifications on the secondary asynchronously, it could happen that the secondary replicates events after some time. The main reasons are:

- **Network round trip time** - high latency links will lead to non-zero replication lag values.
- **Single threaded nature of replication channels** - primary servers have the advantage of applying changes in parallel, whereas secondary ones are only able to apply changes in serial, thus limiting their throughput. In some cases Group Commit can help but is not always applicable.
- **High number of changed rows or computationally expensive SQL** - depending on the replication format (`ROW` vs `STATEMENT`), significant changes to the database through high volume of rows modified, or expensive CPU will all contribute to secondary servers lagging behind the primary.

Generally adding more CPU or Disk resources can alleviate replication lag issues, up to a point.

Binlog Size

This metric shows the overall size of the binary log files, which can exist on both primary and secondary servers.

The binary log (also known as the *binlog*) contains events that describe database changes: `CREATE TABLE`, `ALTER TABLE`, updates, inserts, deletes and other statements or database changes.

The binlog file is read by secondaries via their IO Thread process to replicate database changes modification on the data and on the table structures. There can be more than one binlog file depending on the binlog rotation policy (for example using the configuration variables `max_binlog_size` and `expire_logs_days`) or because of server reboots.

When planning the disk space, take care of the overall dimension of binlog files and adopt a good rotation policy or think about having a separate mount point or disk to store the binlog data.

Binlog Data Written Hourly

This metric shows the amount of data written hourly to the binlog files during the last 24 hours. This metric can give you an idea of how big is your application in terms of data writes (creation, modification, deletion).

Binlog Count

This metric shows the overall count of binary log files, on both primary and secondary servers.

Binlogs Created Hourly

This metric shows the number of binlog files created hourly during the last 24 hours.

Relay Log Space

This metric shows the overall size of the relay log files. It only applies to a secondary host.

The relay log consists of a set of numbered files containing the events to be executed on the secondary host to replicate database changes.

The relay log has the same format as the binlog.

There can be multiple relay log files depending on the rotation policy adopted (using the configuration variable `max_relay_log_size`).

As soon as the SQL thread completes to execute all events in the relay log file, the file is deleted.

If this metric contains a high value, the variable `max_relay_log_file` is high too. Generally, this not a serious issue. If the value of this metric is constantly increased, the secondary is delaying too much in applying the events.

Percona LLC and/or its affiliates ©2024

Treat this metric in the same way as the MySQL Replication Delay metric.

Relay Log Written Hourly

This metric shows the amount of data written hourly into relay log files during the last 24 hours.

Get expert help

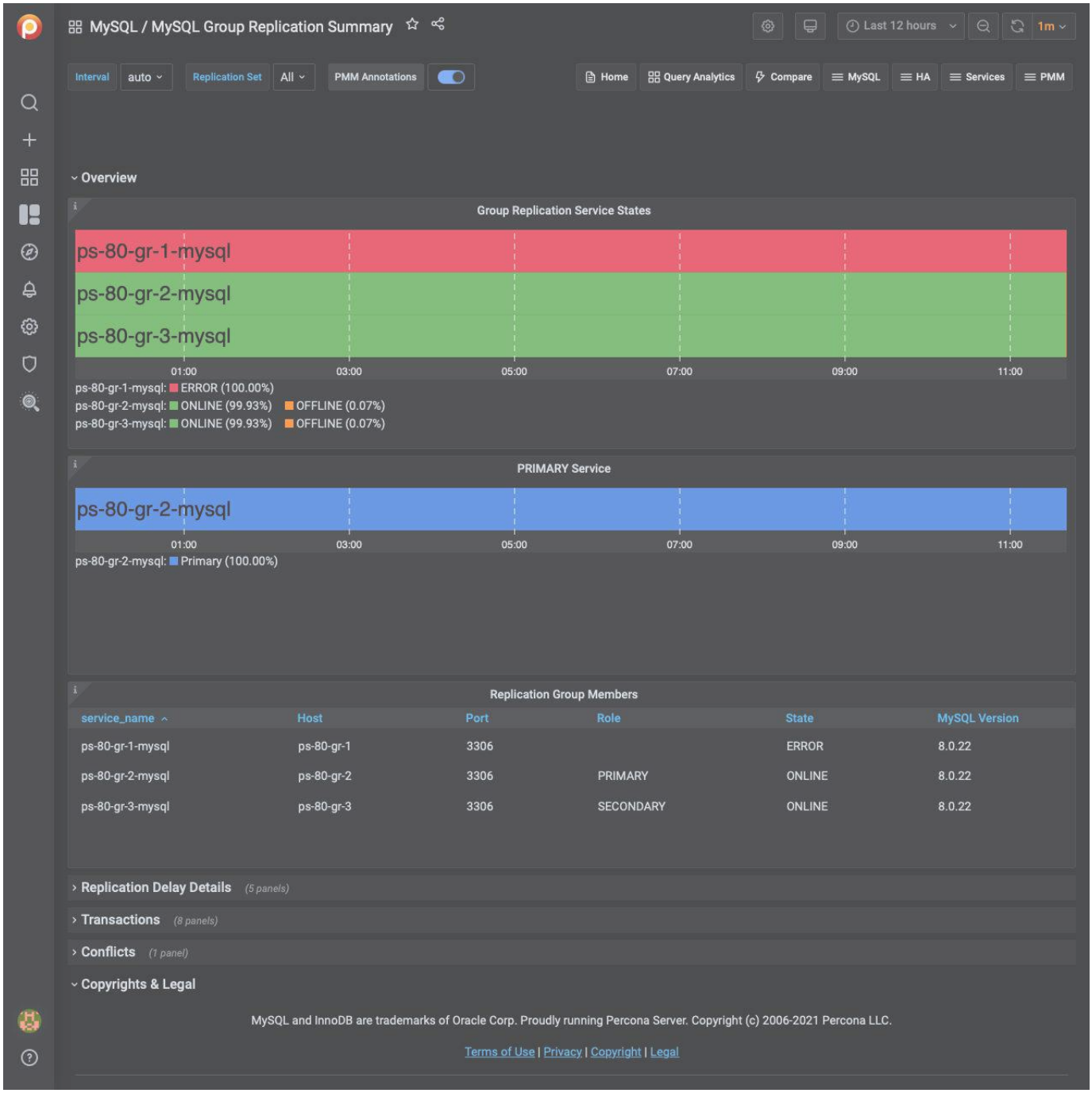
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

Percona LLC and/or its affiliates ©2024

MYSQL GROUP REPLICATION SUMMARY



Overview

- PRIMARY Service
- Group Replication Service States
- Replication Group Members
- Replication Lag
- Replication Delay
- Transport Time

Percona LLC and/or its affiliates ©2024

Transactions

- Transaction Details
- Applied Transactions
- Sent Transactions
- Checked Transactions
- Rolled Back Transactions
- Transactions Row Validating
- Transactions in the Queue for Checking
- Received Transactions Queue

Conflicts

- Detected Conflicts

Get expert help

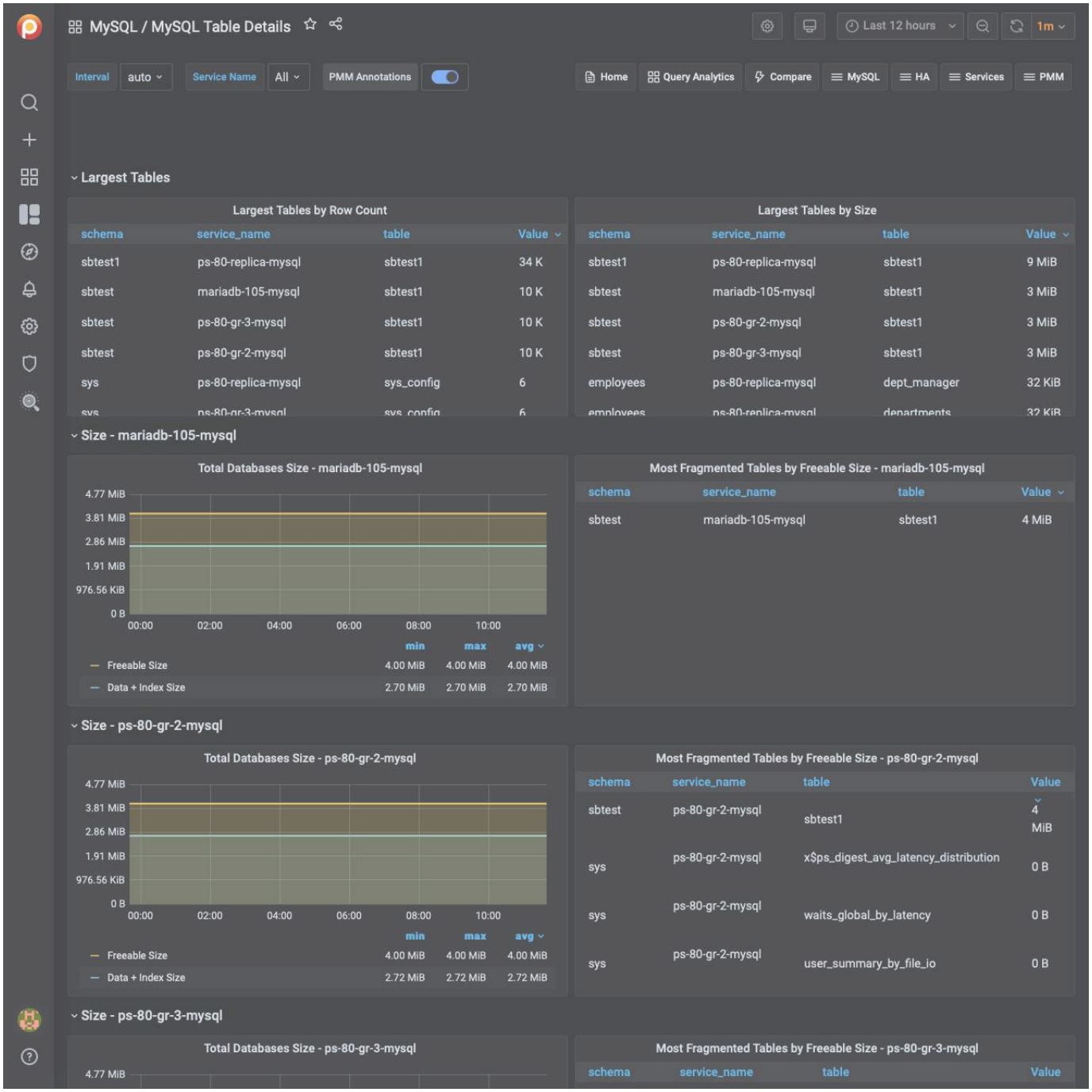
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-03

Percona LLC and/or its affiliates ©2024

MYSQL TABLE DETAILS



Largest Tables

Largest Tables by Row Count

The estimated number of rows in the table from `information_schema.tables`.

Largest Tables by Size

The size of the table components from `information_schema.tables`.

Percona LLC and/or its affiliates ©2024

Pie

Total Database Size

The total size of the database: as data + index size, so freeable one.

Most Fragmented Tables by Freeable Size

The list of 5 most fragmented tables ordered by their freeable size

Table Activity

The next two graphs are available only for Percona Server and MariaDB and require `userstat` variable turned on.

Rows read

The number of rows read from the table, shown for the top 5 tables.

Rows Changed

The number of rows changed in the table, shown for the top 5 tables.

Auto Increment Usage

The current value of an `auto_increment` column from `information_schema`, shown for the top 10 tables.

Get expert help

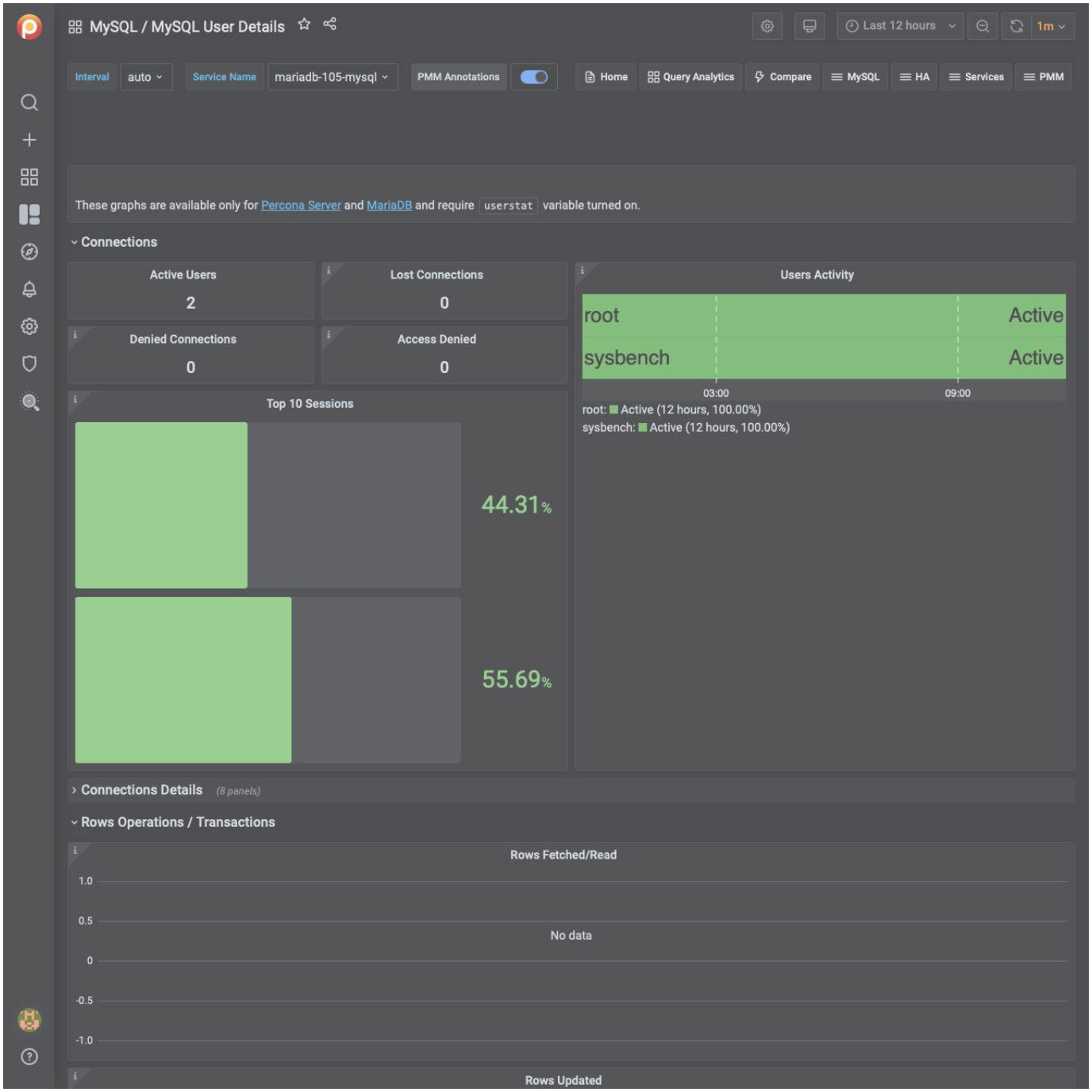
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

MYSQL USER DETAILS



This dashboard requires Percona Server for MySQL 5.1+ or MariaDB 10.1/10.2 with XtraDB. Also `userstat` should be enabled, for example with the `SET GLOBAL userstat=1` statement. See Setting up MySQL.

Data is displayed for the 5 top users.

Top Users by Connections Created

The number of times user's connections connected using SSL to the server.

Top Users by Traffic

The number of bytes sent to the user's connections.

Percona LLC and/or its affiliates ©2024

Top Users by Rows Fetched/Read

The number of rows fetched by the user’s connections.

Top Users by Rows Updated

The number of rows updated by the user’s connections.

Top Users by Busy Time

The cumulative number of seconds there was activity on connections from the user.

Top Users by CPU Time

The cumulative CPU time elapsed, in seconds, while servicing connections of the user.

Get expert help

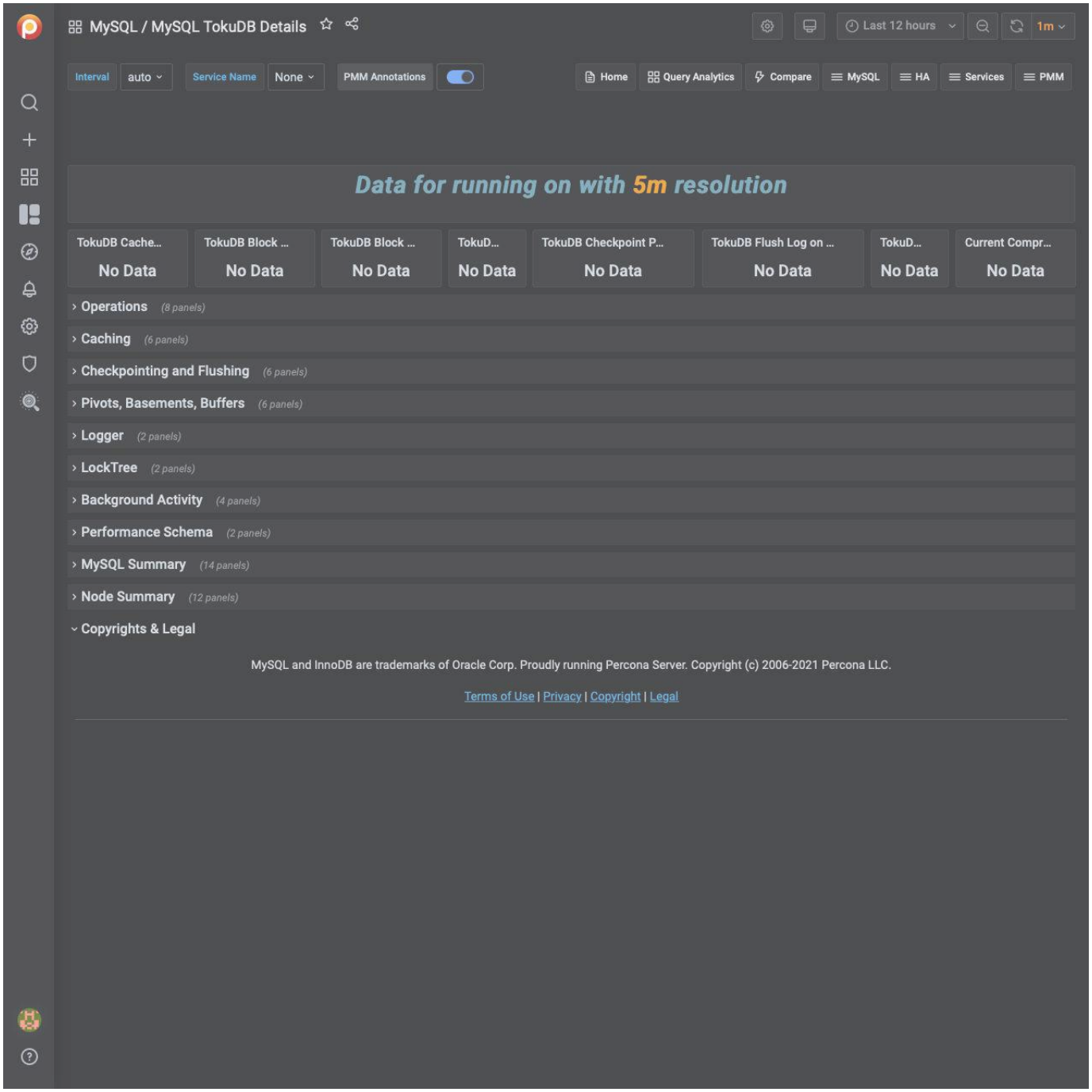
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-10

Percona LLC and/or its affiliates ©2024

MYSQL TOKUDB DETAILS



No description

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2021-05-11

MongoDB Dashboards

EXPERIMENTAL MONGODB COLLECTION OVERVIEW

Disclaimer

This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.

Availability

This experimental dashboard is available starting with PMM 2.30.0.

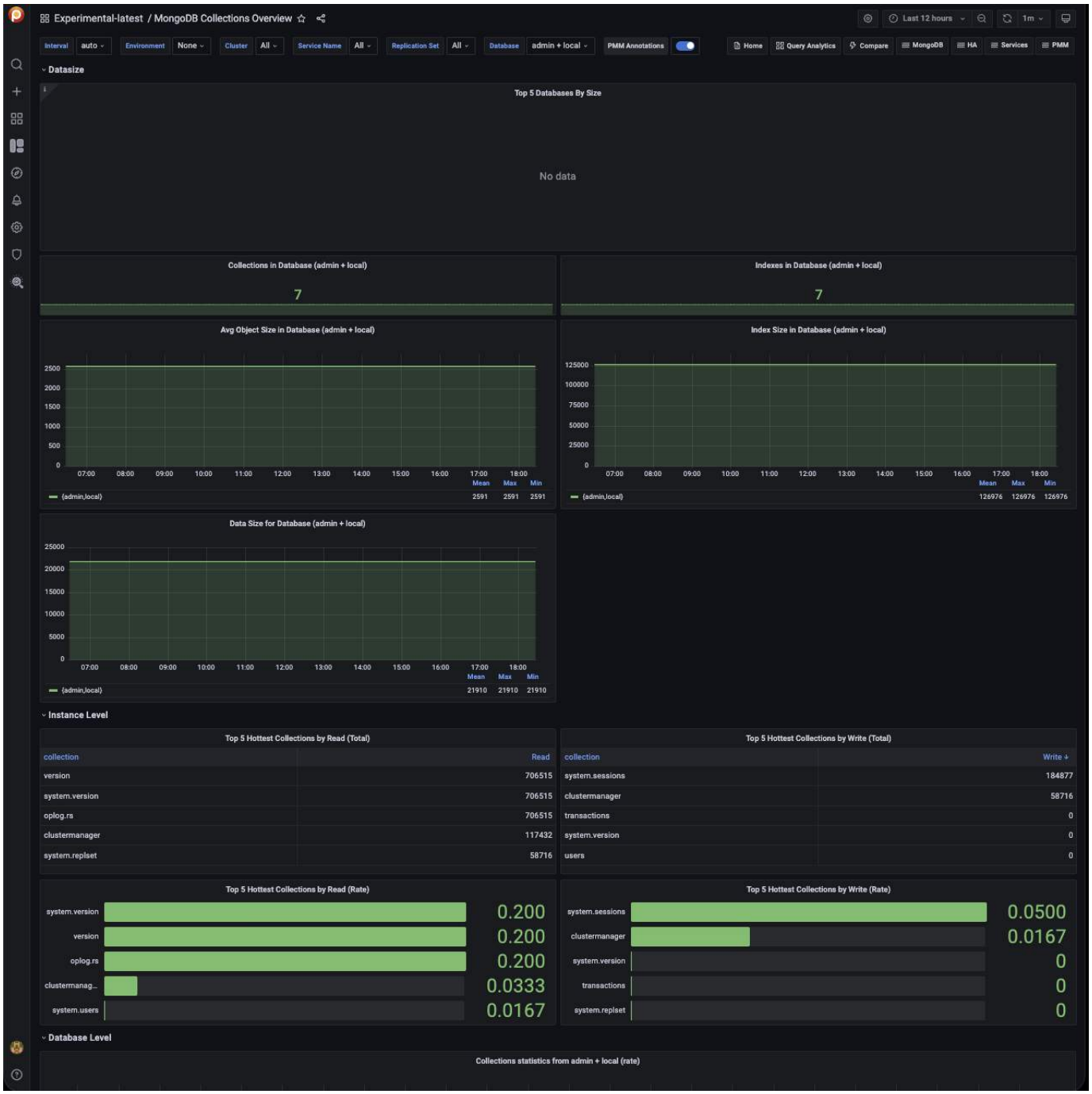
This realtime dashboard contains panels of data about the Hottest Collections in the MongoDB database.

The Instance level includes two panels, one for the *Hottest Collections by Read (Total)* and the *Hottest Collections by Write (total)*.

The next panel displays data at the *Database Level*, where you can view MongoDB metrics such as *Commands*, *Inserts*, *Updates*, *Removes*, and *Getmore*.

The last panel shows the number of operations in the chosen database.

Percona LLC and/or its affiliates ©2024



Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-08-24

Percona LLC and/or its affiliates ©2024

EXPERIMENTAL MONGODB COLLECTION DETAILS

Disclaimer

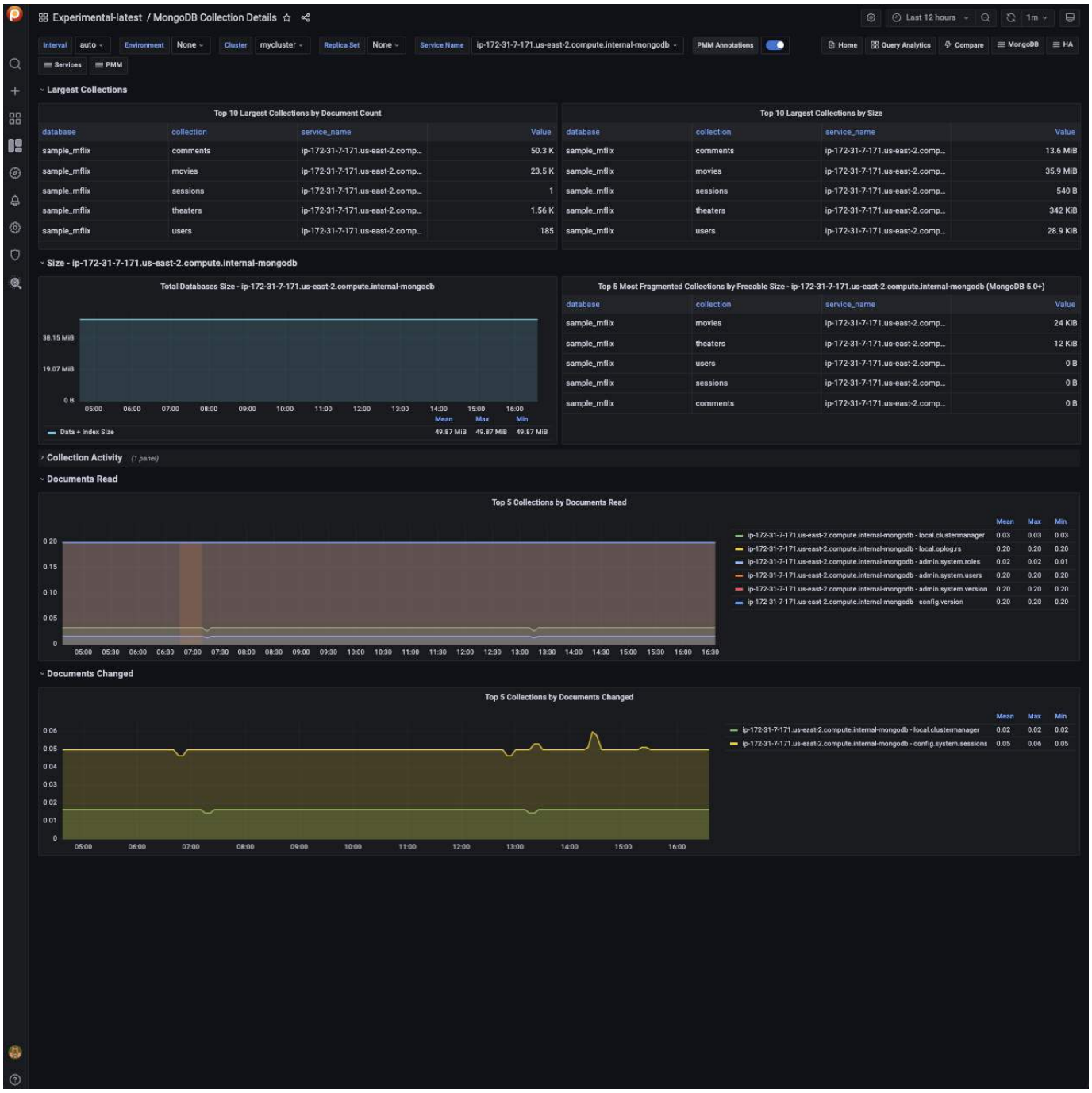
This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.

Availability

This experimental dashboard is available starting with PMM 2.30.0.

This realtime experimental dashboard provides detailed information about the top collections by document count, size, and document read for MongoDB databases.

Percona LLC and/or its affiliates ©2024



Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-08-24

Percona LLC and/or its affiliates ©2024

EXPERIMENTAL MONGODB OPLOG DETAILS

Disclaimer

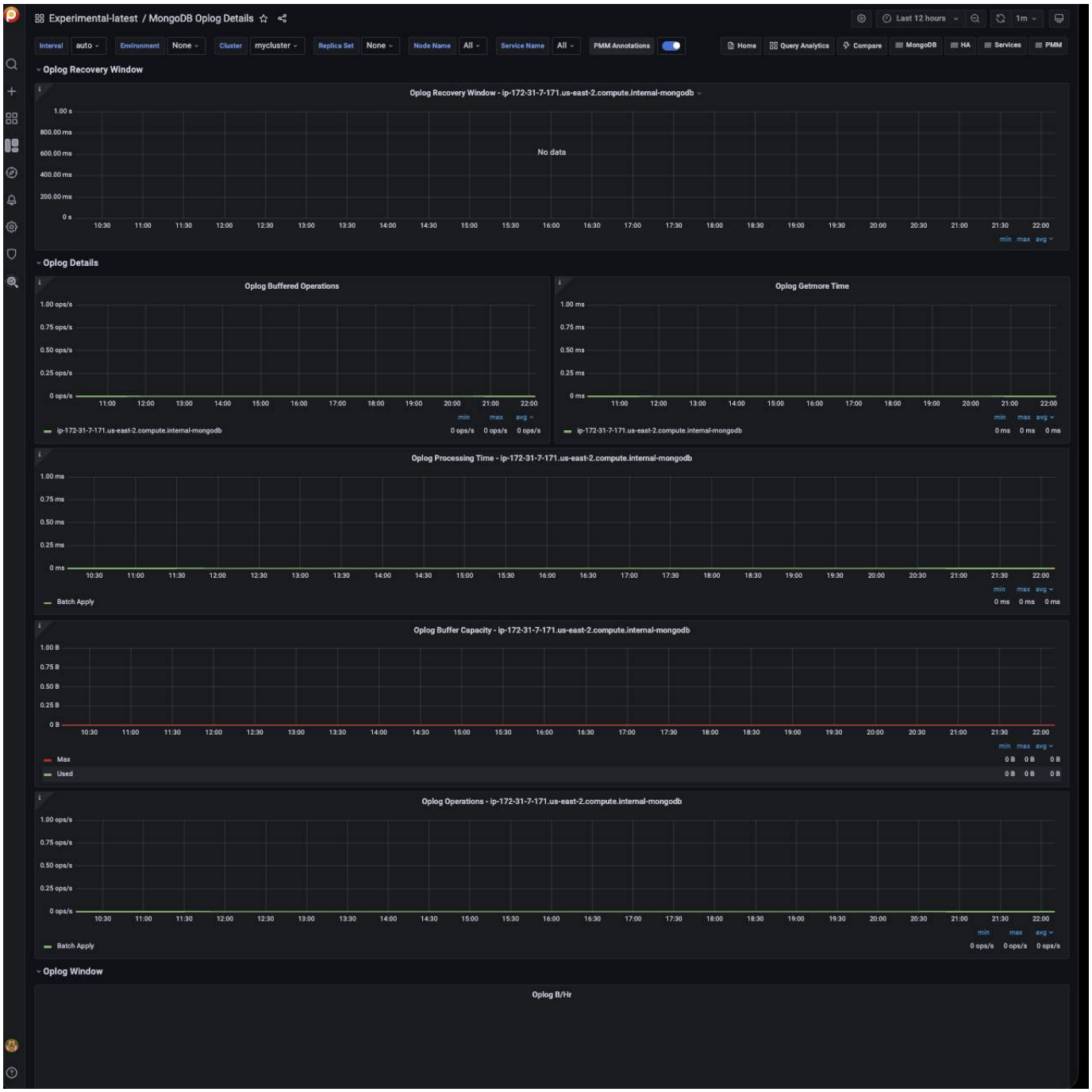
This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.

Availability

This experimental dashboard is available starting with PMM 2.30.0.

This realtime dashboard contains Oplog details such as Recovery Window, Processing Time, Buffer Capacity, and Oplog Operations.

Percona LLC and/or its affiliates ©2024



Get expert help

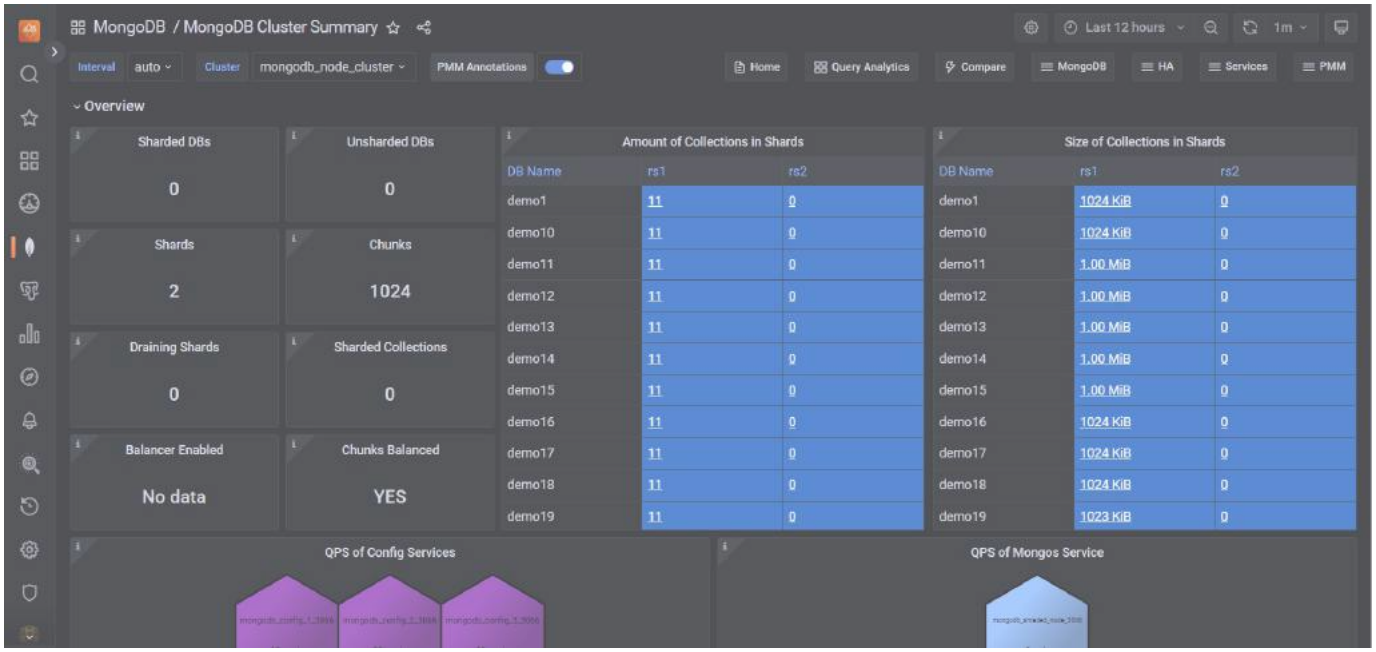
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-08-24

Percona LLC and/or its affiliates ©2024

MONGODB CLUSTER SUMMARY



Current Connections Per Shard

TCP connections (Incoming) in mongod processes.

Total Connections

Incoming connections to mongos nodes.

Cursors Per Shard

The Cursor is a MongoDB Collection of the document which is returned upon the find method execution.

Mongos Cursors

The Cursor is a MongoDB Collection of the document which is returned upon the find method execution.

Operations Per Shard

Ops/sec, classified by legacy wire protocol type (`query`, `insert`, `update`, `delete`, `getmore`).

Total Mongos Operations

Ops/sec, classified by legacy wire protocol type (`query`, `insert`, `update`, `delete`, `getmore`).

Change Log Events

Count, over last 10 minutes, of all types of configuration db changelog events.

Oplog Range by Set

Timespan ‘window’ between oldest and newest ops in the Oplog collection.

Get expert help

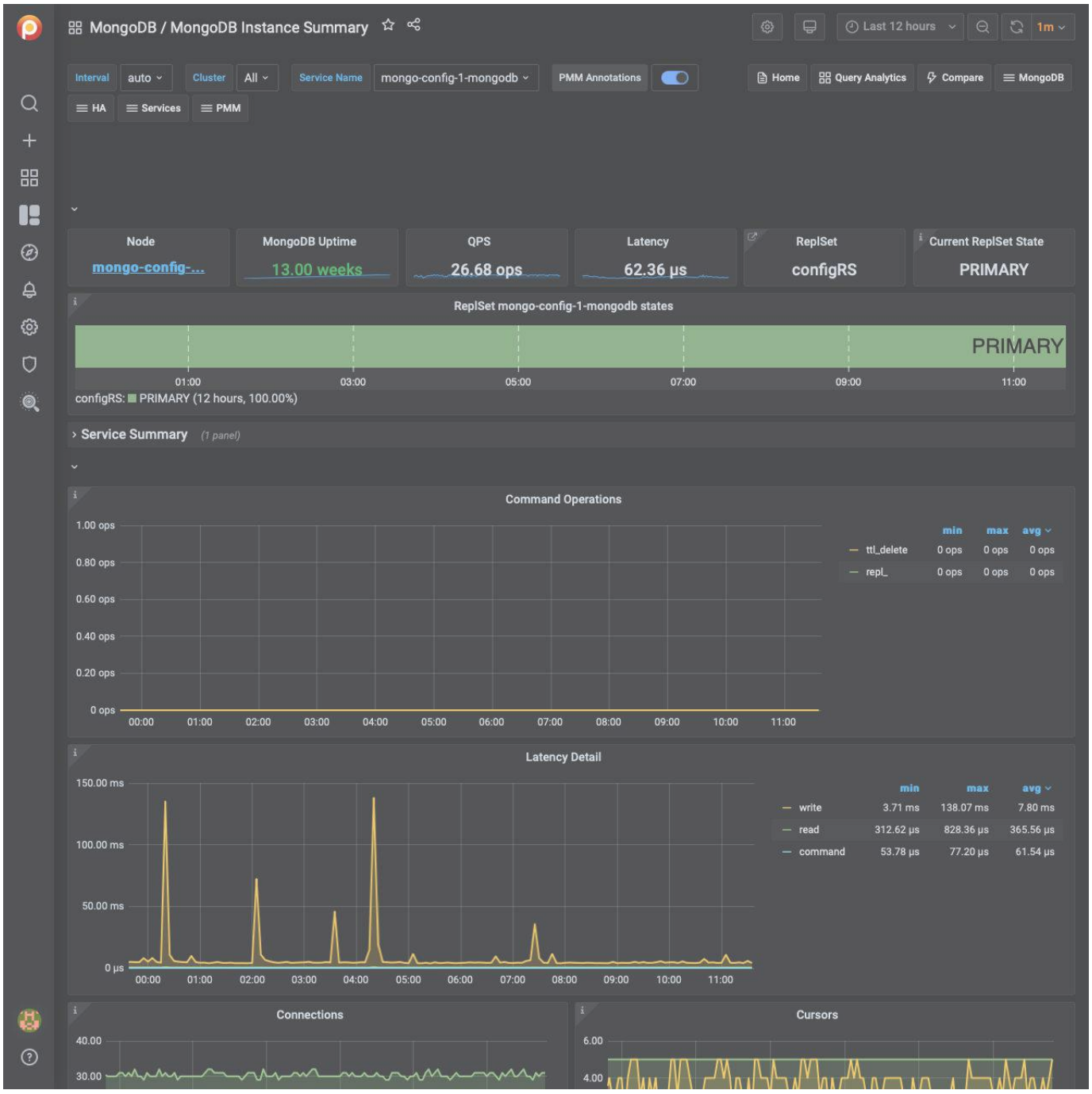
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2021-05-11

MONGODB INSTANCE SUMMARY



Command Operations

Ops or Replicated Ops/sec classified by legacy wire protocol type (`query`, `insert`, `update`, `delete`, `getmore`). And (from the internal TTL threads) the docs deletes/sec by TTL indexes.

Latency Detail

Average latency of operations (classified by read, write, or (other) command)

Percona LLC and/or its affiliates ©2024

Connections

TCP connections (Incoming)

Cursors

Open cursors. Includes idle cursors.

Document Operations

Docs per second inserted, updated, deleted or returned. (not 1-to-1 with operation counts.)

Queued Operations

Operations queued due to a lock.

Query Efficiency

Ratio of Documents returned or Index entries scanned / full documents scanned

Scanned and Moved Objects

This panel shows the number of objects (both data (`scanned_objects`) and index (`scanned`)) as well as the number of documents that were moved to a new location due to the size of the document growing. Moved documents only apply to the MMAPv1 storage engine.

`getLastError` Write Time

Legacy driver operation: Number of, and Sum of time spent, per second executing `getLastError` commands to confirm write concern.

`getLastError` Write Operations

Legacy driver operation: Number of `getLastError` commands that timed out trying to confirm write concern.

Assert Events

This panel shows the number of assert events per second on average over the given time period. In most cases assertions are trivial, but you would want to check your log files if this counter spikes or is consistently high.

Page Faults

Unix or Window memory page faults. Not necessarily from MongoDB.

Get expert help

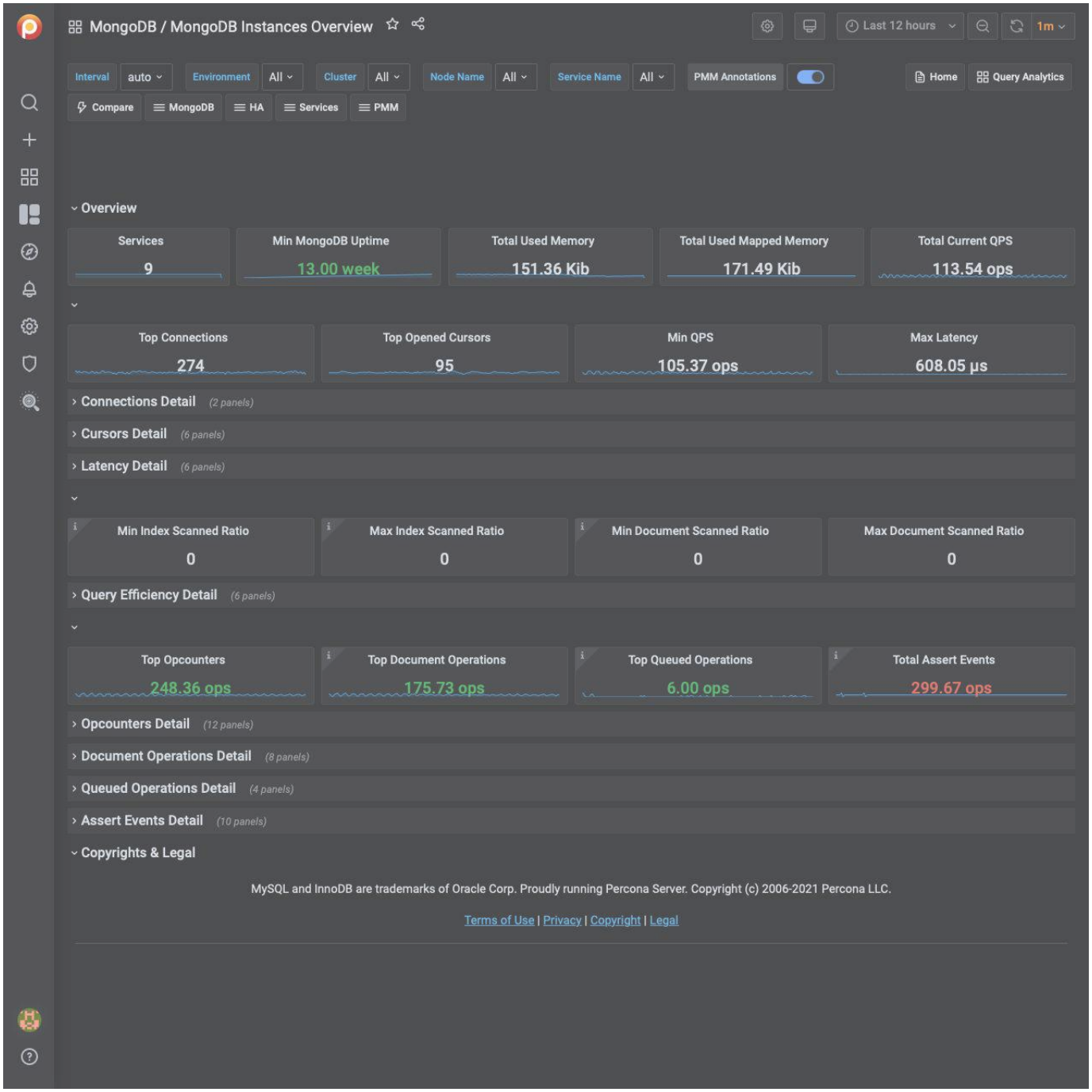
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

MONGODB INSTANCES OVERVIEW



This dashboard provides basic information about MongoDB instances.

Command Operations

Shows how many times a command is executed per second on average during the selected interval.

Look for peaks and drops and correlate them with other graphs.

Connections

Keep in mind the hard limit on the maximum number of connections set by your distribution.

Anything over 5,000 should be a concern, because the application may not close connections correctly.

Percona LLC and/or its affiliates ©2024

Cursors

Helps identify why connections are increasing. Shows active cursors compared to cursors being automatically killed after 10 minutes due to an application not closing the connection.

Document Operations

When used in combination with *Command Operations*, this graph can help identify *write amplification*. For example, when one `insert` or `update` command actually inserts or updates hundreds, thousands, or even millions of documents.

Queued Operations

Any number of queued operations for long periods of time is an indication of possible issues. Find the cause and fix it before requests get stuck in the queue.

`getLastError` Write Time, `getLastError` Write Operations

This is useful for write-heavy workloads to understand how long it takes to verify writes and how many concurrent writes are occurring.

Asserts

Asserts are not important by themselves, but you can correlate spikes with other graphs.

Memory Faults

Memory faults indicate that requests are processed from disk either because an index is missing or there is not enough memory for the data set. Consider increasing memory or sharding out.

Get expert help

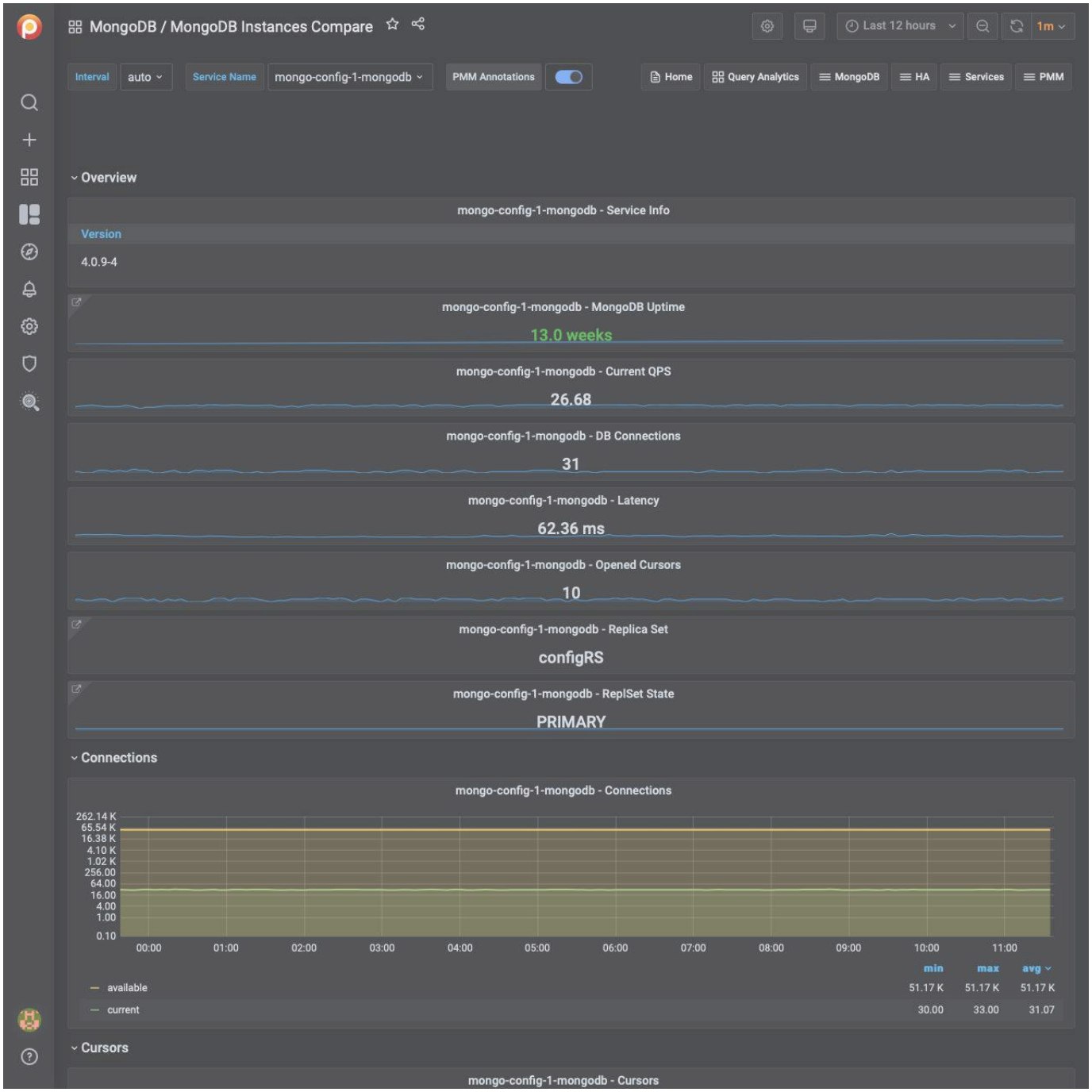
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-03

Percona LLC and/or its affiliates ©2024

MONGODB INSTANCES COMPARE



Connections

No description

Cursors

No description

Latency

Average latency of operations (classified by read, write, or (other) command)

Scan Ratios

Ratio of index entries scanned or whole docs scanned / number of documents returned

Percona LLC and/or its affiliates ©2024

Index Filtering Effectiveness

No description

Requests

Ops/sec (classified by (legacy) wire protocol request type)

Document Operations

Documents inserted/updated/deleted or returned per sec

Queued Operations

The number of operations that are currently queued and waiting for a lock

Used Memory

No description

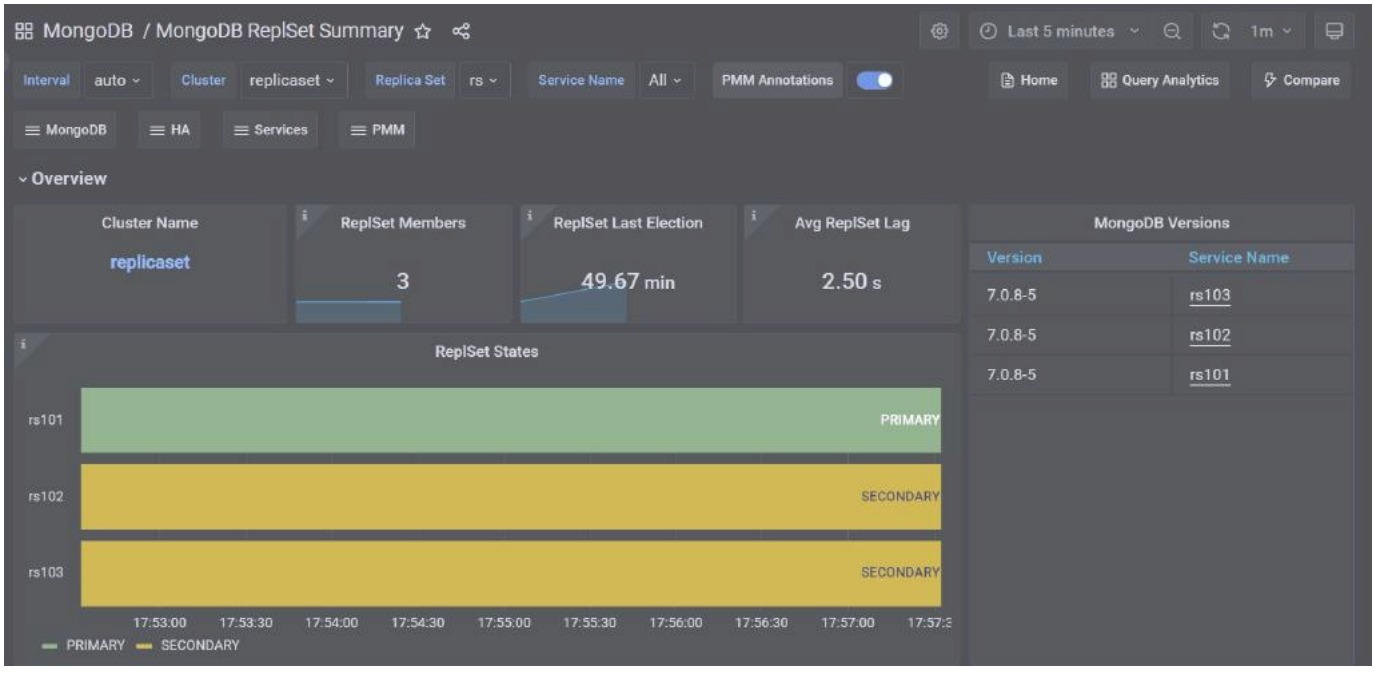
Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

MONGODB REPLSET SUMMARY



Replication Lag

MongoDB replication lag occurs when the secondary node cannot replicate data fast enough to keep up with the rate that data is being written to the primary node. It could be caused by something as simple as network latency, packet loss within your network, or a routing issue.

Percona LLC and/or its affiliates ©2024

Operations - by service name

Operations are classified by legacy wire protocol type (insert, update, and delete only).

Max Member Ping Time - by service name

This metric can show a correlation with the replication lag value.

Max Heartbeat Time

Time span between now and last heartbeat from replicaset members.

Elections

Count of elections. Usually zero; 1 count by each healthy node will appear in each election. Happens when the primary role changes due to either normal maintenance or trouble events.

Oplog Recovery Window - by service name

Timespan ‘window’ between newest and the oldest op in the Oplog collection.

Get expert help

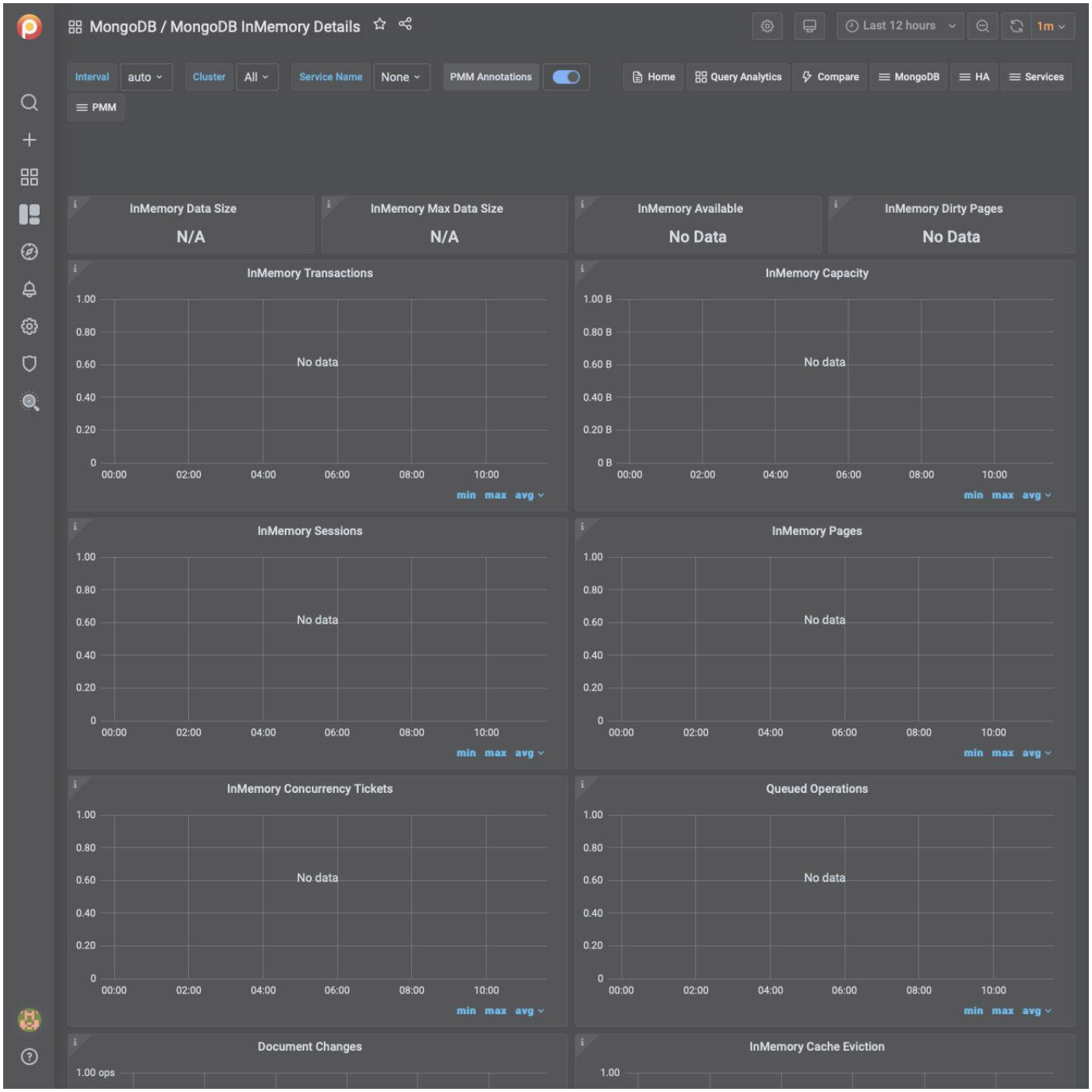
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

MONGODB INMEMORY DETAILS



InMemory Transactions

WiredTiger internal transactions

InMemory Capacity

Configured max and current size of the WiredTiger cache.

InMemory Sessions

Internal WiredTiger storage engine cursors and sessions currently open.

InMemory Pages

Pages in the WiredTiger cache

Percona LLC and/or its affiliates ©2024

InMemory Concurrency Tickets

A WT ‘ticket’ is assigned out for every operation running simultaneously in the WT storage engine. “Tickets available” = hard coded high value - “Tickets Out”.

Queued Operations

Operations queued due to a lock

Document Changes

Mixed metrics: Docs per second inserted, updated, deleted or returned on any type of node (primary or secondary); + replicated write Ops/sec; + TTL deletes per second.

InMemory Cache Eviction

This panel shows the number of pages that have been evicted from the WiredTiger cache for the given time period. The InMemory storage engine only evicts modified pages which signals a compaction of the data and removal of the dirty pages.

Scanned and Moved Objects

This panel shows the number of objects (both data (`scanned_objects`) and index (`scanned`)) as well as the number of documents that were moved to a new location due to the size of the document growing. Moved documents only apply to the MMAPv1 storage engine.

Page Faults

Unix or Window memory page faults. Not necessarily from MongoDB.

Get expert help

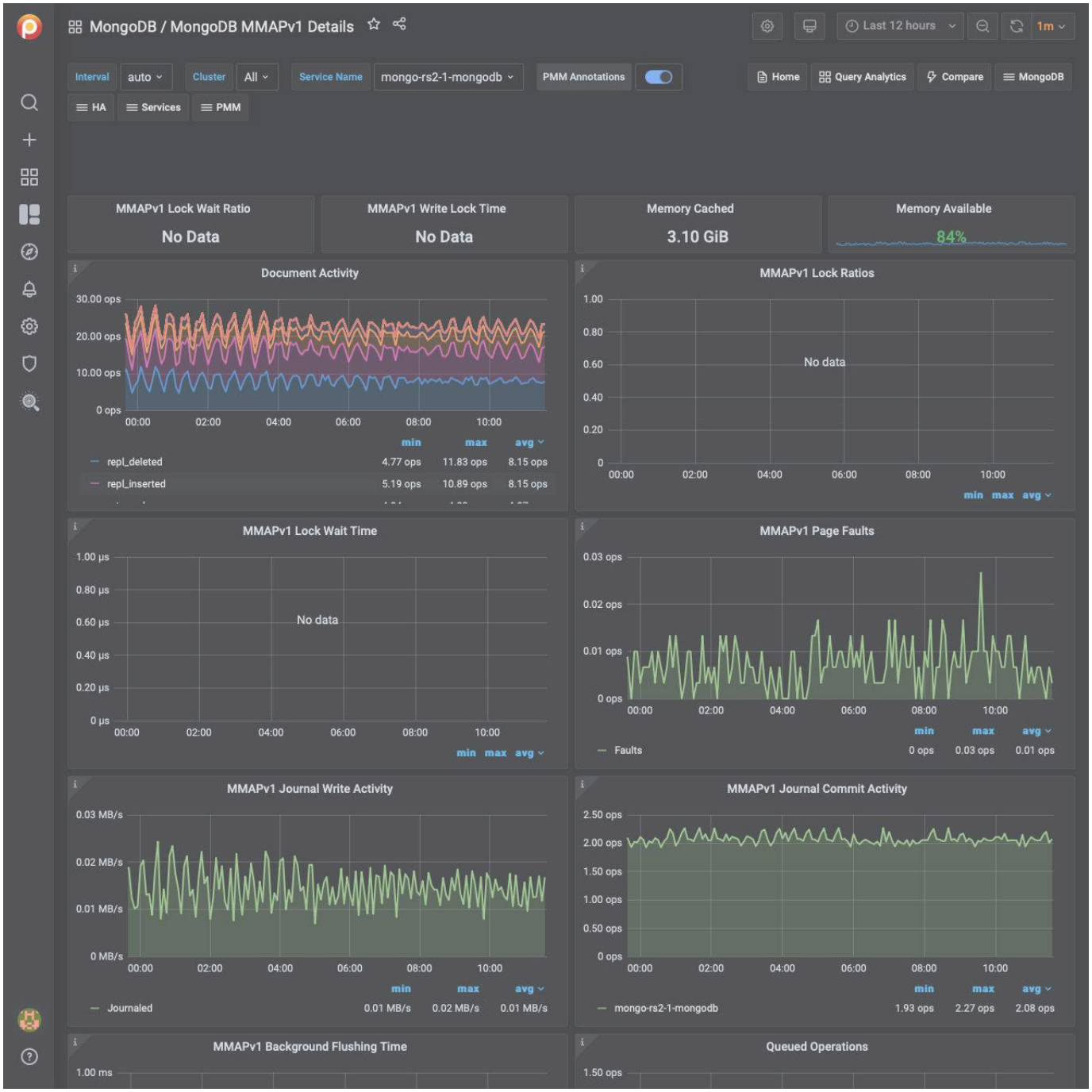
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

MONGODB MMAPV1 DETAILS



Document Activity

Docs per second inserted, updated, deleted or returned. Also showing replicated write ops and internal TTL index deletes.

MMAPv1 Lock Wait Time

Time spent per second waiting to acquire locks.

MMAPv1 Page Faults

Unix or Window memory page faults. Not necessarily from MongoDB.

Percona LLC and/or its affiliates ©2024

MMAPv1 Journal Write Activity

MB processed through the journal in memory.

MMAPv1 Journal Commit Activity

MB committed to disk for the journal.

MMAPv1 Background Flushing Time

Average time in ms, over full uptime of `mongod` process, the MMAP background flushes have taken.

Queued Operations

Queue size of ops waiting to be submitted to storage engine layer. (see WiredTiger concurrency tickets for number of ops being processed simultaneously in storage engine layer.)

Client Operations

Ops and Replicated Ops/sec, classified by legacy wire protocol type (`query`, `insert`, `update`, `delete`, `getmore`).

Scanned and Moved Objects

This panel shows the number of objects (both data (`scanned_objects`) and index (`scanned`)) as well as the number of documents that were moved to a new location due to the size of the document growing. Moved documents only apply to the MMAPv1 storage engine.

Get expert help

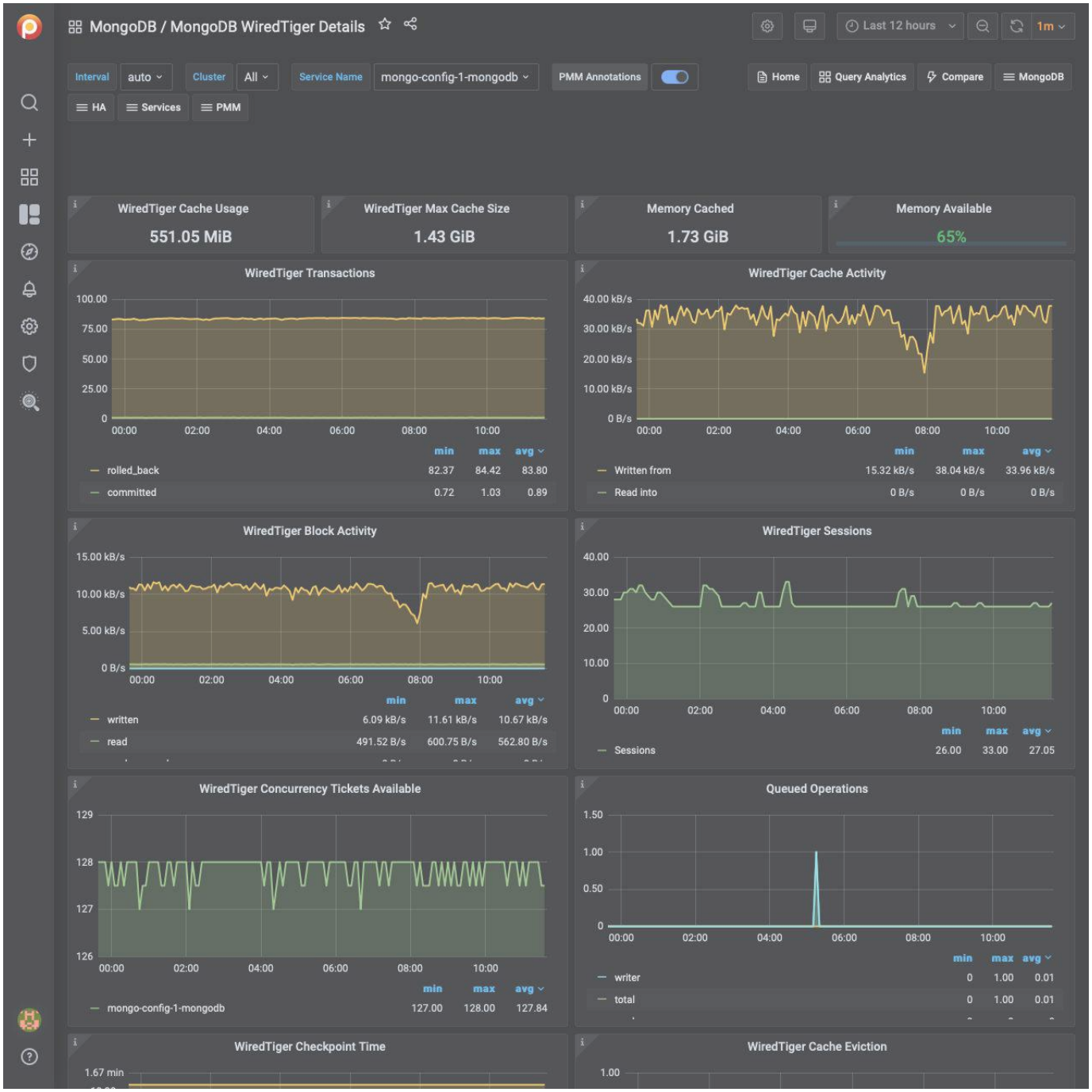
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

MONGODB WIREDTIGER DETAILS



WiredTiger Transactions

WiredTiger internal transactions

WiredTiger Cache Activity

Data volume transferred per second between the WT cache and data files. Writes out always imply disk; Reads are often from OS file buffer cache already in RAM, but disk if not.

WiredTiger Block Activity

Data volume handled by the WT block manager per second

Percona LLC and/or its affiliates ©2024

WiredTiger Sessions

Internal WT storage engine cursors and sessions currently open

WiredTiger Concurrency Tickets Available

A WT ‘ticket’ is assigned out for every operation running simultaneously in the WT storage engine. “Available” = hard-coded high value - “Out”.

Queued Operations

Operations queued due to a lock.

WiredTiger Checkpoint Time

The time spent in WT checkpoint phase. Warning: This calculation averages the cyclical event (default: 1 min) execution to a per-second value.

WiredTiger Cache Eviction

Least-recently used pages being evicted due to WT cache becoming full.

WiredTiger Cache Capacity

Configured max and current size of the WT cache.

WiredTiger Cache Pages

WiredTiger Log Operations

WT internal write-ahead log operations.

WiredTiger Log Activity

Data volume moved per second in WT internal write-ahead log.

WiredTiger Log Records

Number of records appended per second in WT internal log.

Document Changes

Mixed metrics: Docs per second inserted, updated, deleted or returned on any type of node (primary or secondary); + replicated write Ops/sec; + TTL deletes per second.

Scanned and Moved Objects

This panel shows the number of objects (both data (`scanned_objects`) and index (`scanned`)) as well as the number of documents that were moved to a new location due to the size of the document growing. Moved documents only apply to the MMAPv1 storage engine.

Page Faults

Unix or Window memory page faults. Not necessarily from MongoDB.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

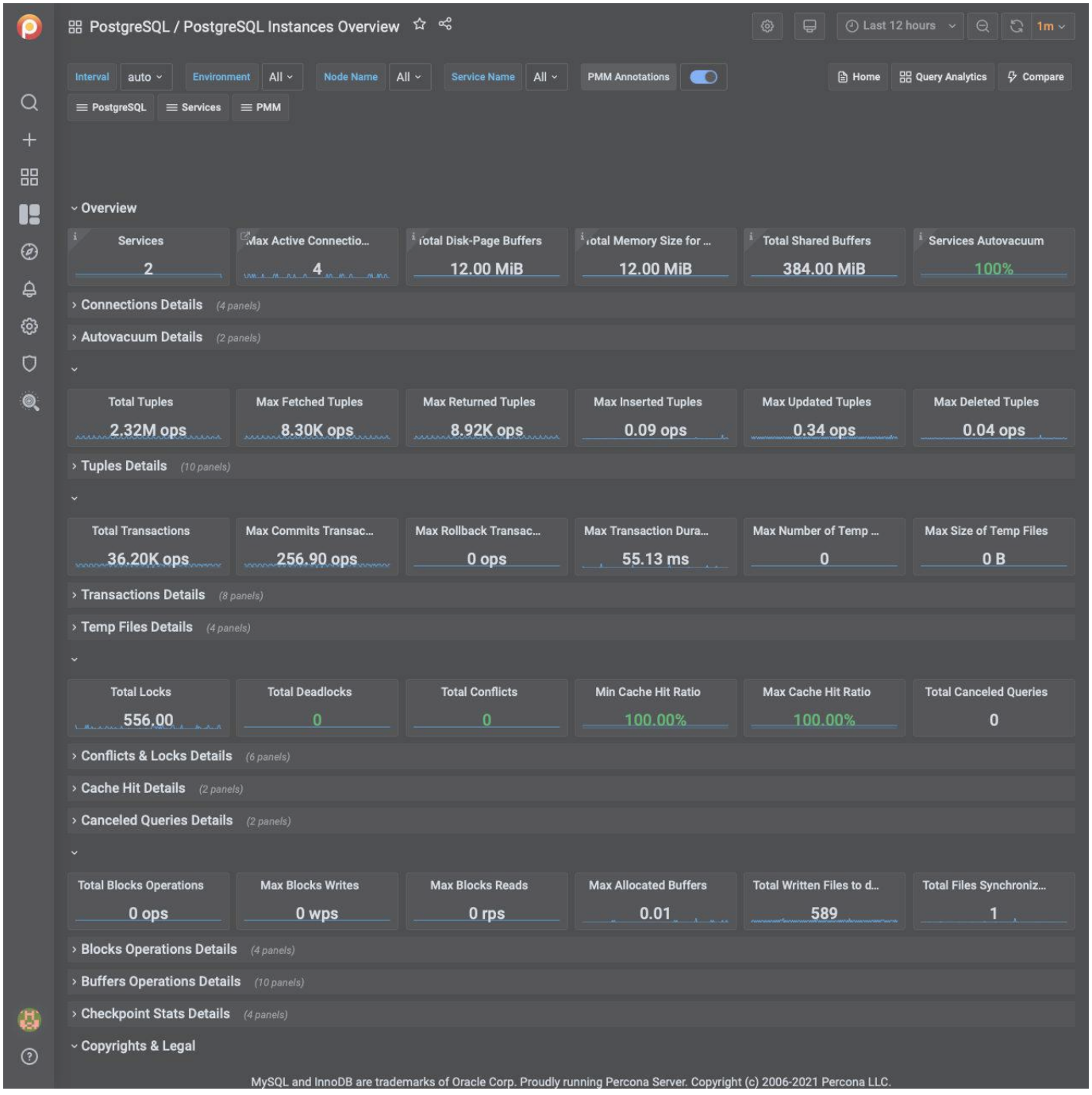
Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2021-05-11

PostgreSQL Dashboards

POSTGRESQL INSTANCES OVERVIEW



Connected

Reports whether PMM Server can connect to the PostgreSQL instance.

Version

The version of the PostgreSQL instance.

Percona LLC and/or its affiliates ©2024

Shared Buffers

Defines the amount of memory the database server uses for shared memory buffers. Default is `128MB`. Guidance on tuning is `25%` of RAM, but generally doesn't exceed `40%`.

Disk-Page Buffers

The setting `wal_buffers` defines how much memory is used for caching the write-ahead log entries. Generally this value is small (`3%` of `shared_buffers` value), but it may need to be modified for heavily loaded servers.

Memory Size for each Sort

The parameter `work_mem` defines the amount of memory assigned for internal sort operations and hash tables before writing to temporary disk files. The default is `4MB`.

Disk Cache Size

PostgreSQL's `effective_cache_size` variable tunes how much RAM you expect to be available for disk caching. Generally adding Linux free+cached will give you a good idea. This value is used by the query planner whether plans will fit in memory, and when defined too low, can lead to some plans rejecting certain indexes.

Autovacuum

Whether autovacuum process is enabled or not. Generally the solution is to vacuum more often, not less.

PostgreSQL Connections

Max Connections

The maximum number of client connections allowed. Change this value with care as there are some memory resources that are allocated on a per-client basis, so setting `max_connections` higher will generally increase overall PostgreSQL memory usage.

Connections

The number of connection attempts (successful or not) to the PostgreSQL server.

Active Connections

The number of open connections to the PostgreSQL server.

PostgreSQL Tuples

Tuples

The total number of rows processed by PostgreSQL server: fetched, returned, inserted, updated, and deleted.

Read Tuple Activity

The number of rows read from the database: as returned so fetched ones.

Tuples Changed per 5 min

The number of rows changed in the last 5 minutes: inserted, updated, and deleted ones.

PostgreSQL Transactions

Transactions

The total number of transactions that have been either been committed or rolled back.

Percona LLC and/or its affiliates ©2024

Duration of Transactions

Maximum duration in seconds any active transaction has been running.

Temp Files

Number of Temp Files

The number of temporary files created by queries.

Size of Temp files

The total amount of data written to temporary files by queries in bytes.

All temporary files are taken into account by these two gauges, regardless of why the temporary file was created (e.g., sorting or hashing), and regardless of the `log_temp_files` setting.

Conflicts and Locks

Conflicts/Deadlocks

The number of queries canceled due to conflicts with recovery in the database (due to dropped tablespaces, lock timeouts, old snapshots, pinned buffers, or deadlocks).

Number of Locks

The number of deadlocks detected by PostgreSQL.

Buffers and Blocks Operations

Operations with Blocks

The time spent reading and writing data file blocks by back ends, in milliseconds.

Tip

Capturing read and write time statistics is possible only if `track_io_timing` setting is enabled. This can be done either in configuration file or with the following query executed on the running system:

```
ALTER SYSTEM SET track_io_timing=ON;
SELECT pg_reload_conf();
```

Buffers

The number of buffers allocated by PostgreSQL.

Canceled Queries

The number of queries that have been canceled due to dropped tablespaces, lock timeouts, old snapshots, pinned buffers, and deadlocks.

Data shown by this gauge are based on the `pg_stat_database_conflicts` view.

Cache Hit Ratio

The number of times disk blocks were found already in the buffer cache, so that a read was not necessary.

Percona LLC and/or its affiliates ©2024

This only includes hits in the PostgreSQL buffer cache, not the operating system’s file system cache.

Checkpoint Stats

The total amount of time that has been spent in the portion of checkpoint processing where files are either written or synchronized to disk, in milliseconds.

PostgreSQL Settings

The list of all settings of the PostgreSQL server.

System Summary

This section contains the following system parameters of the PostgreSQL server: CPU Usage, CPU Saturation and Max Core Usage, Disk I/O Activity, and Network Traffic.

Get expert help

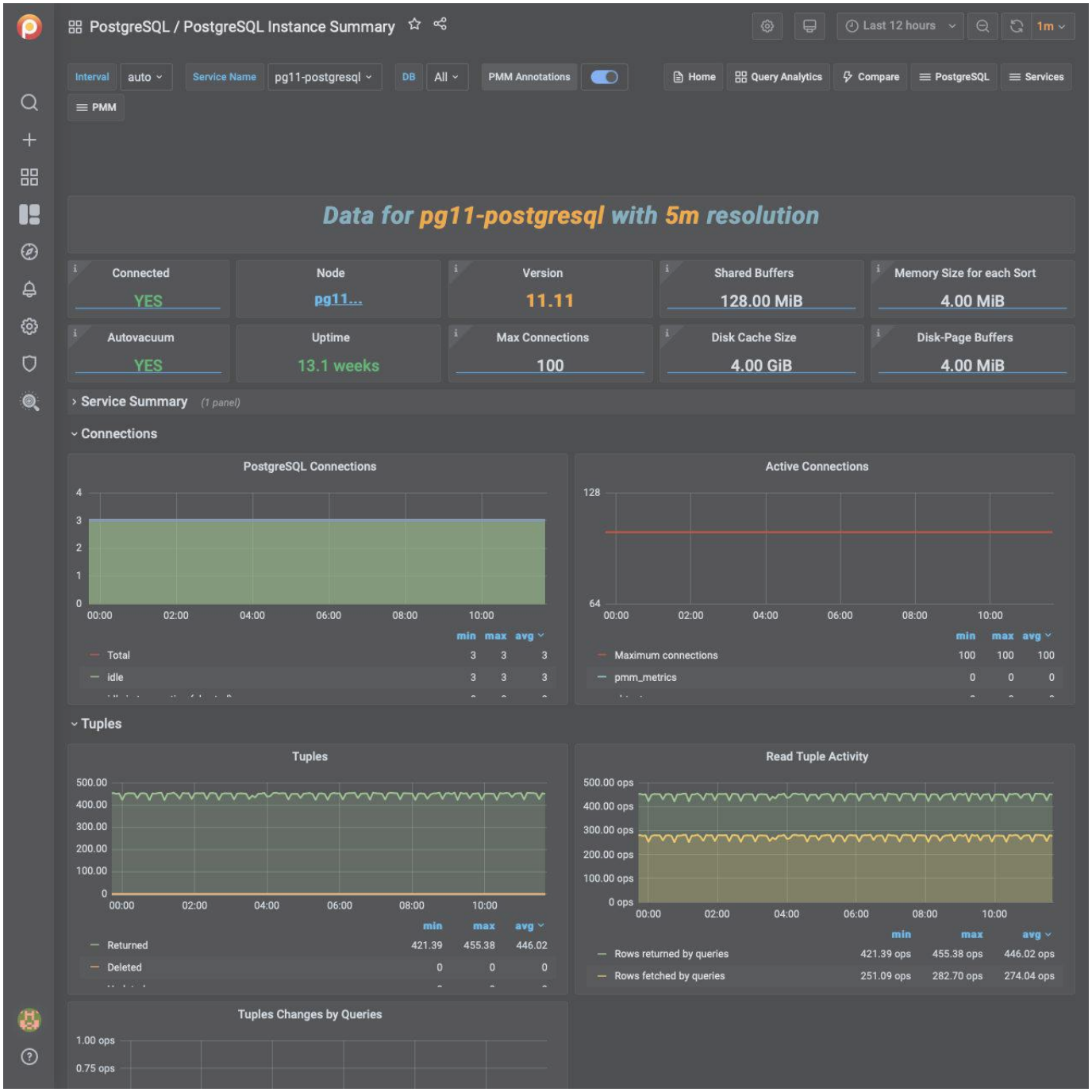
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-15

Percona LLC and/or its affiliates ©2024

POSTGRESQL INSTANCE SUMMARY



Number of Temp Files

Cumulative number of temporary files created by queries in this database since service start. All temporary files are counted, regardless of why the temporary file was created (e.g., sorting or hashing), and regardless of the `log_temp_files` setting.

Size of Temp Files

Cumulative amount of data written to temporary files by queries in this database since service start. All temporary files are counted, regardless of why the temporary file was created, and regardless of the `log_temp_files` setting.

Percona LLC and/or its affiliates ©2024

Temp Files Activity

Number of temporary files created by queries in this database. All temporary files are counted, regardless of why the temporary file was created (e.g., sorting or hashing), and regardless of the `log_temp_files` setting.

Temp Files Utilization

Total amount of data written to temporary files by queries in this database. All temporary files are counted, regardless of why the temporary file was created, and regardless of the `log_temp_files` setting.

Canceled Queries

Based on `pg_stat_database_conflicts` view

Get expert help

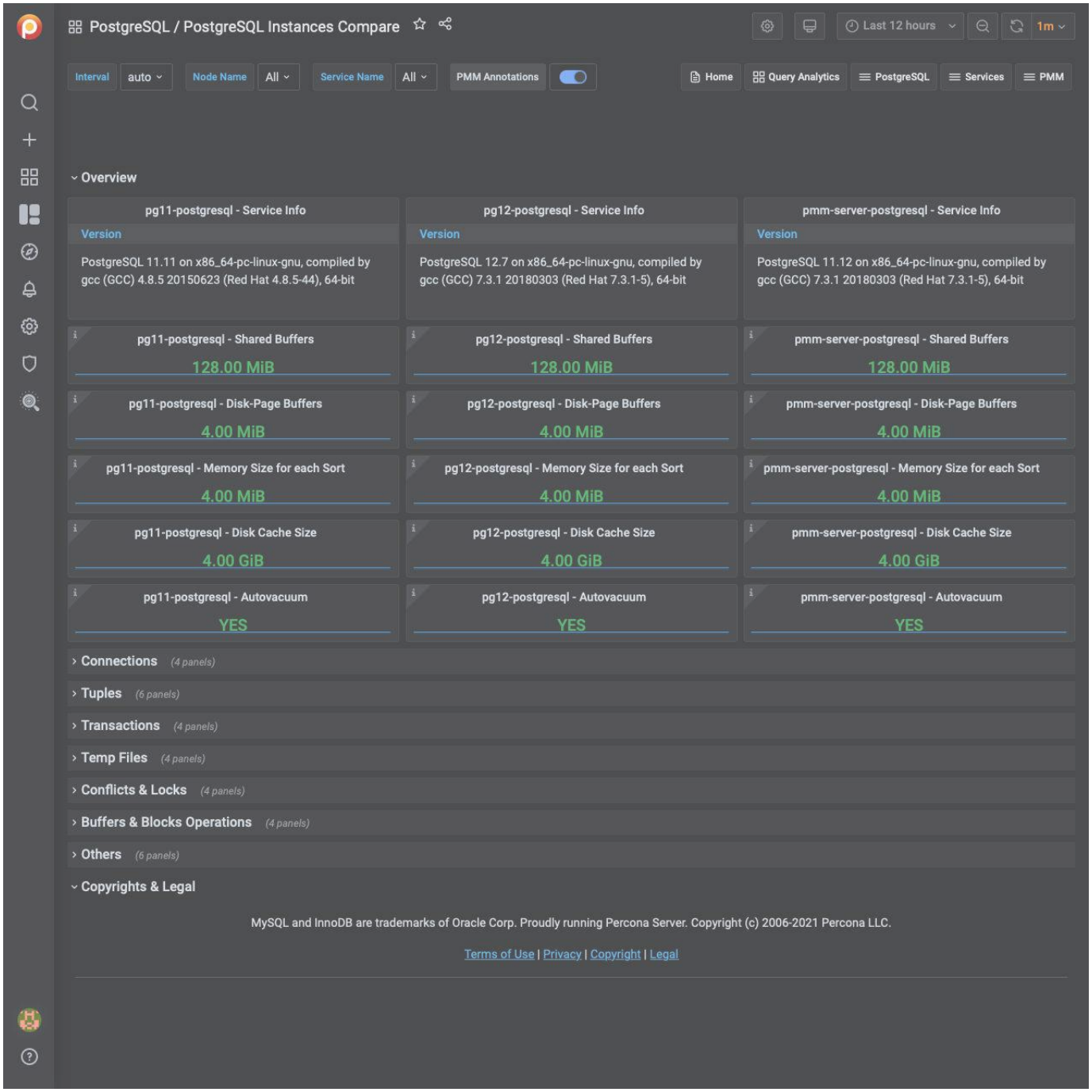
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

POSTGRESQL INSTANCES COMPARE



No description

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

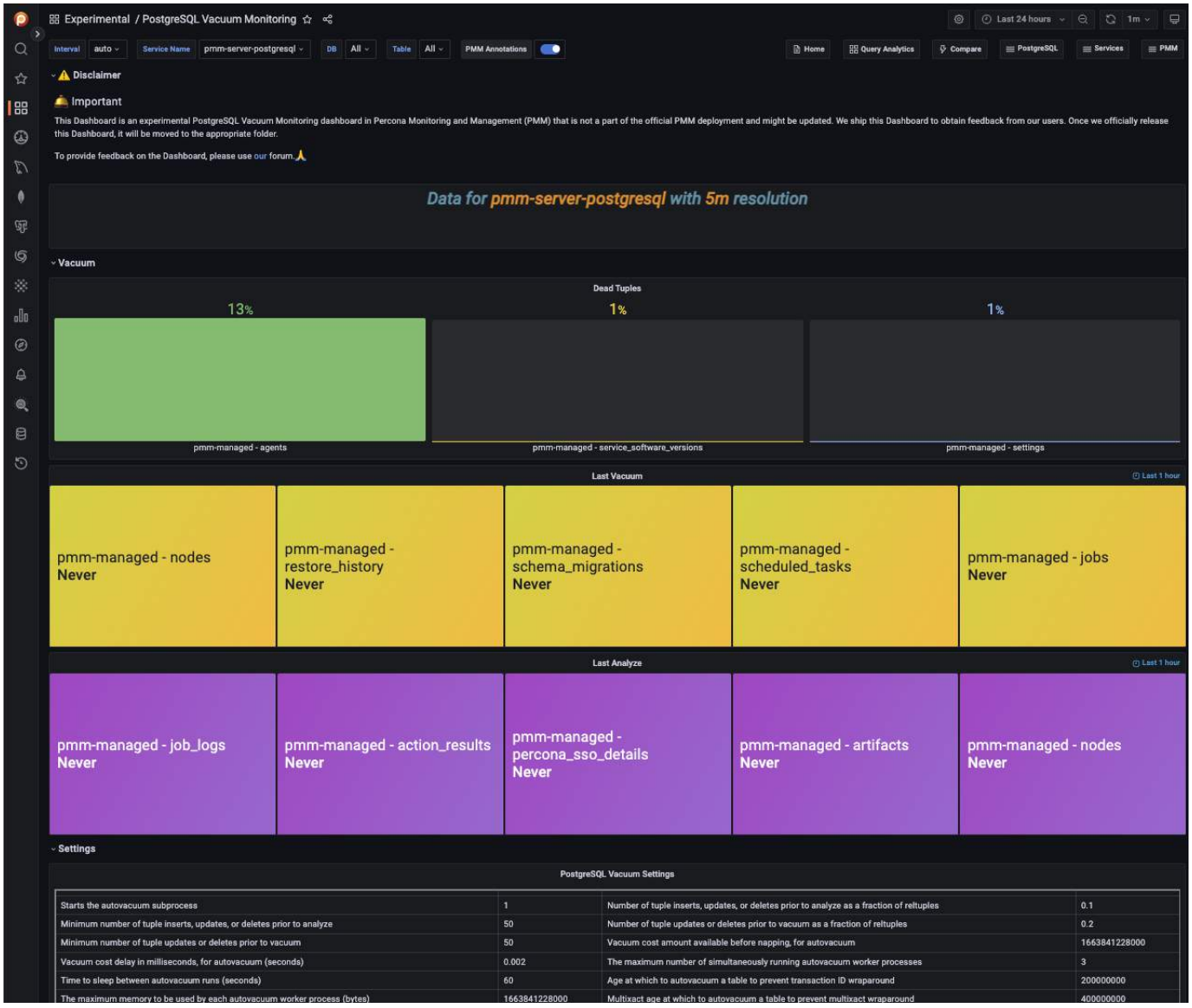
Percona LLC and/or its affiliates ©2024

2021-06-03

EXPERIMENTAL POSTGRESQL VACUUM MONITORING

Disclaimer

This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.



This dashbaord provides timely insights into the autovacuum process in PostgreSQL.

Percona LLC and/or its affiliates ©2024

This dashboard contains the following:

- Dead tuples - Identifies the number of dead rows in each table even though the rows are physically removed from the table.
- Last time vacuum ran - Tracks the last time a vacuum or autovacuum process successfully ran on each of your tables.
- Number of rows modified since last Analyze - The number of rows changed since the last time ANALYZE ran.
- Manual vacuum events - Tracks the number of times a manual vacuum was run on each table.
- Table disk usage - Tracking the disk space used by each table is crucial as it enables you to gauge expected changes in the query performance over time - but it can also help you detect potential vacuuming-related issues.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

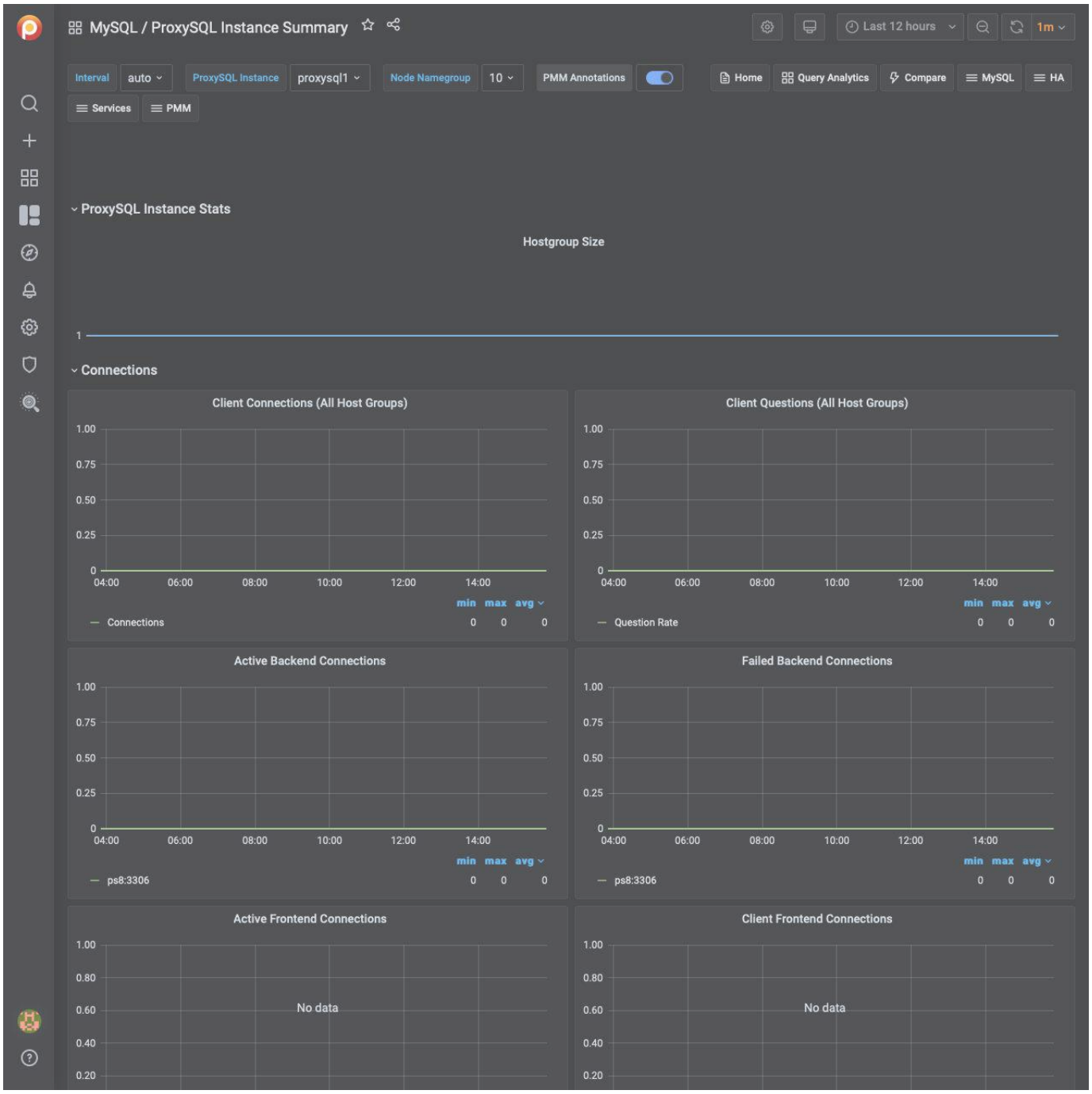
Community Forum Get a Percona Expert

2022-09-26

Percona LLC and/or its affiliates ©2024

ProxySQL Dashboards

PROXYSQL INSTANCE SUMMARY



Network Traffic

Network traffic refers to the amount of data moving across a network at a given point in time.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

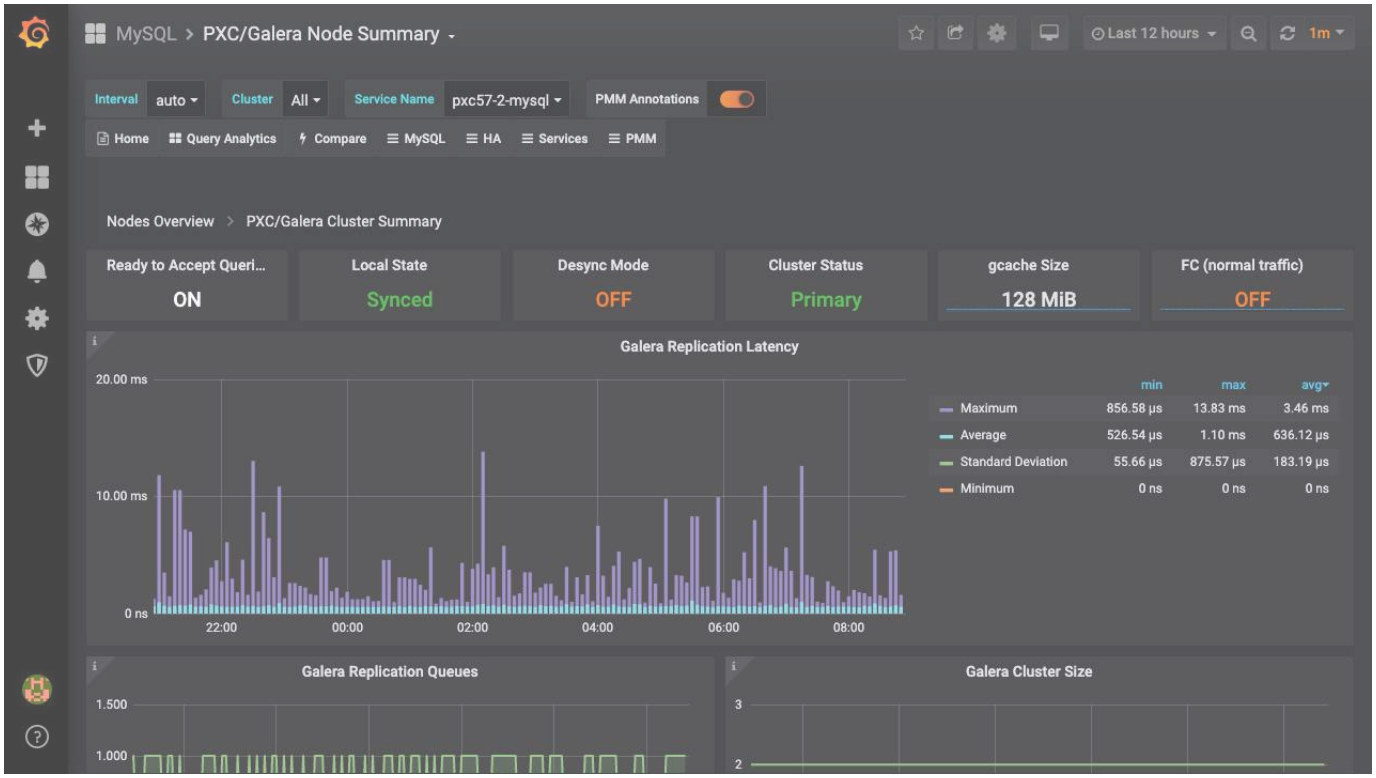
Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2021-05-11

HA Dashboards

PXC/GALERA NODE SUMMARY



Galera Replication Latency

Shows figures for the replication latency on group communication. It measures latency from the time point when a message is sent out to the time point when a message is received. As replication is a group operation, this essentially gives you the slowest ACK and longest RTT in the cluster.

Galera Replication Queues

Shows the length of receive and send queues.

Galera Cluster Size

Shows the number of members currently connected to the cluster.

Galera Flow Control

Shows the number of `FC_PAUSE` events sent/received. They are sent by a node when its replication queue gets too full. If a node is sending out FC messages it indicates a problem.

Galera Parallelization Efficiency

Shows the average distances between highest and lowest seqno that are concurrently applied, committed and can be possibly applied in parallel (potential degree of parallelization).

Galera Writing Conflicts

Shows the number of local transactions being committed on this node that failed certification (some other node had a commit that conflicted with ours) – client received deadlock error on commit and also the number of local transactions in flight on this node that were aborted because they locked something an applier thread needed –

Percona LLC and/or its affiliates ©2024

deadlock error anywhere in an open transaction. Spikes in the graph may indicate writing to the same table potentially the same rows from 2 nodes.

Available Downtime before SST Required

Shows for how long the node can be taken out of the cluster before SST is required. SST is a full state transfer method.

Galera Writeset Count

Shows the count of transactions received from the cluster (any other node) and replicated to the cluster (from this node).

Galera Writeset Size

Shows the average transaction size received/replicated.

Galera Writeset Traffic

Shows the bytes of data received from the cluster (any other node) and replicated to the cluster (from this node).

Galera Network Usage Hourly

Shows the bytes of data received from the cluster (any other node) and replicated to the cluster (from this node).

Get expert help

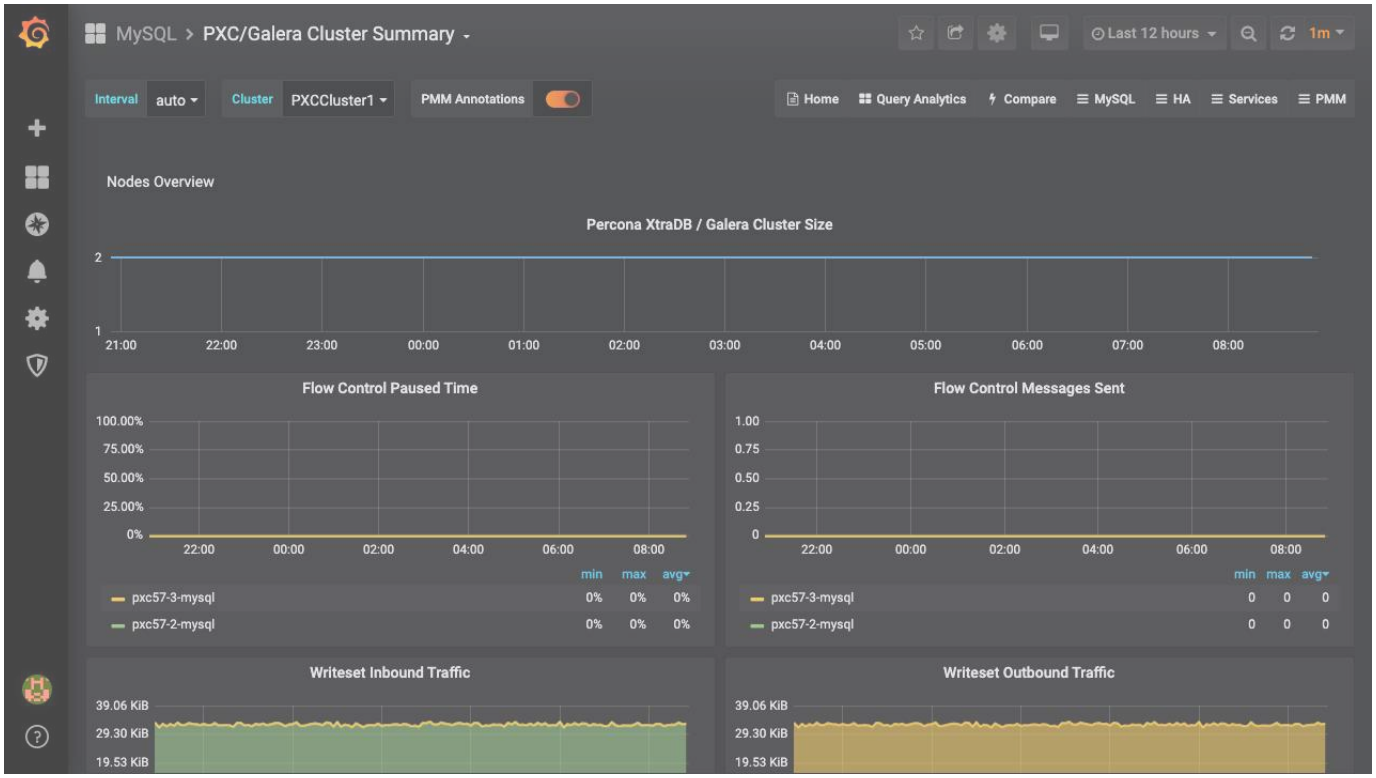
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

PXC/GALERA CLUSTER SUMMARY



No description

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

EXPERIMENTAL PXC/GALERA CLUSTER SUMMARY

Disclaimer

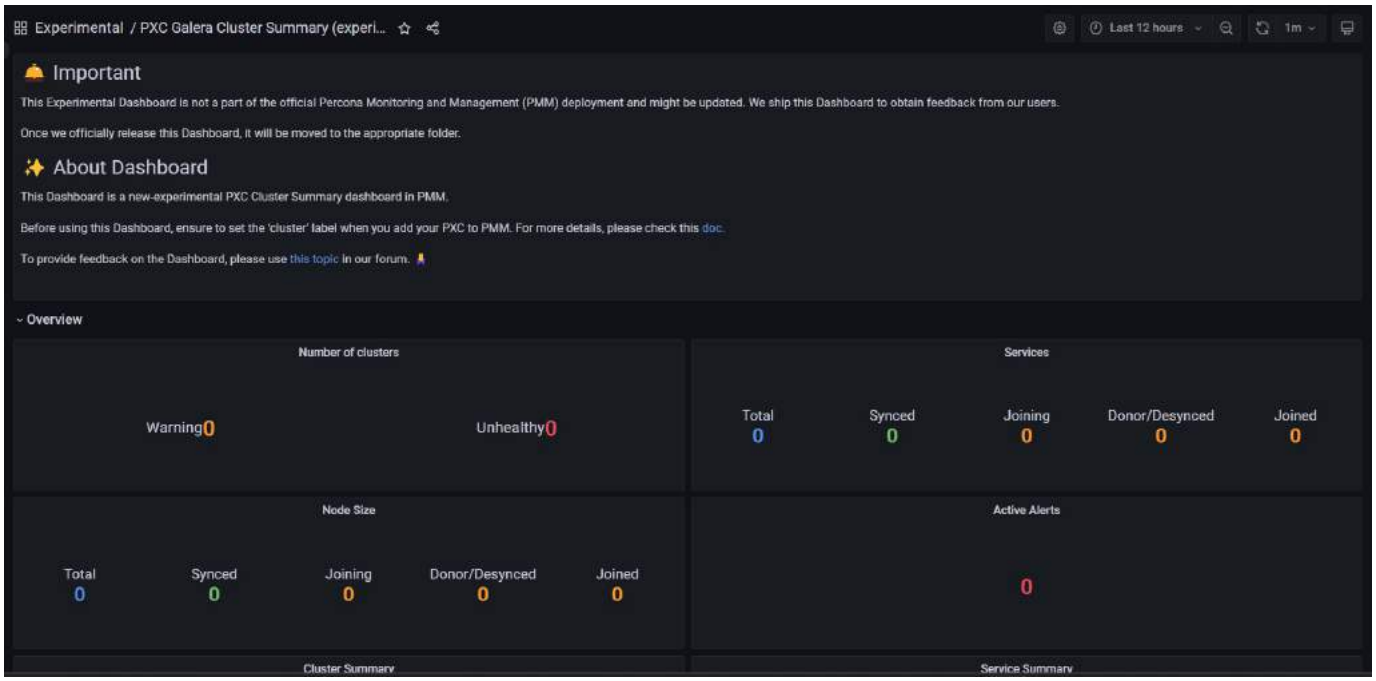
This is an Experimental Dashboard that is not part of the official Percona Monitoring and Management (PMM) deployment and might be updated. We ship this Dashboard to obtain feedback from our users.

Availability

This experimental dashboard is available starting with PMM 2.29.0.

The experimental PXC/Galera Cluster Summary dashboard provides a high level information about the clusters, resource utilization and its state for MySQL databases.

Percona LLC and/or its affiliates ©2024



Get expert help

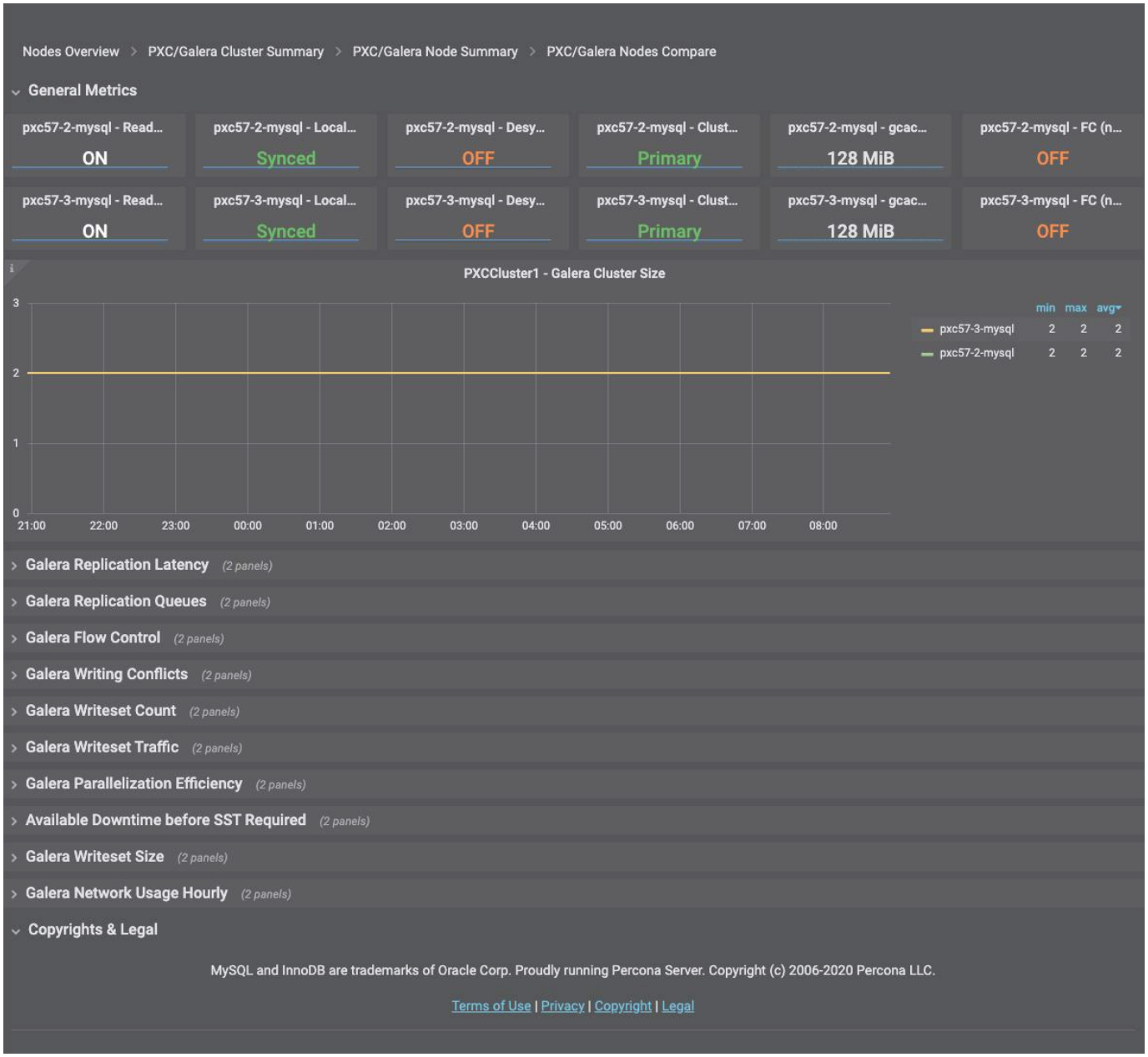
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-07-19

Percona LLC and/or its affiliates ©2024

PXC/GALERA NODES COMPARE



$cluster – Galera Cluster Size

Shows the number of members currently connected to the cluster.

Get expert help

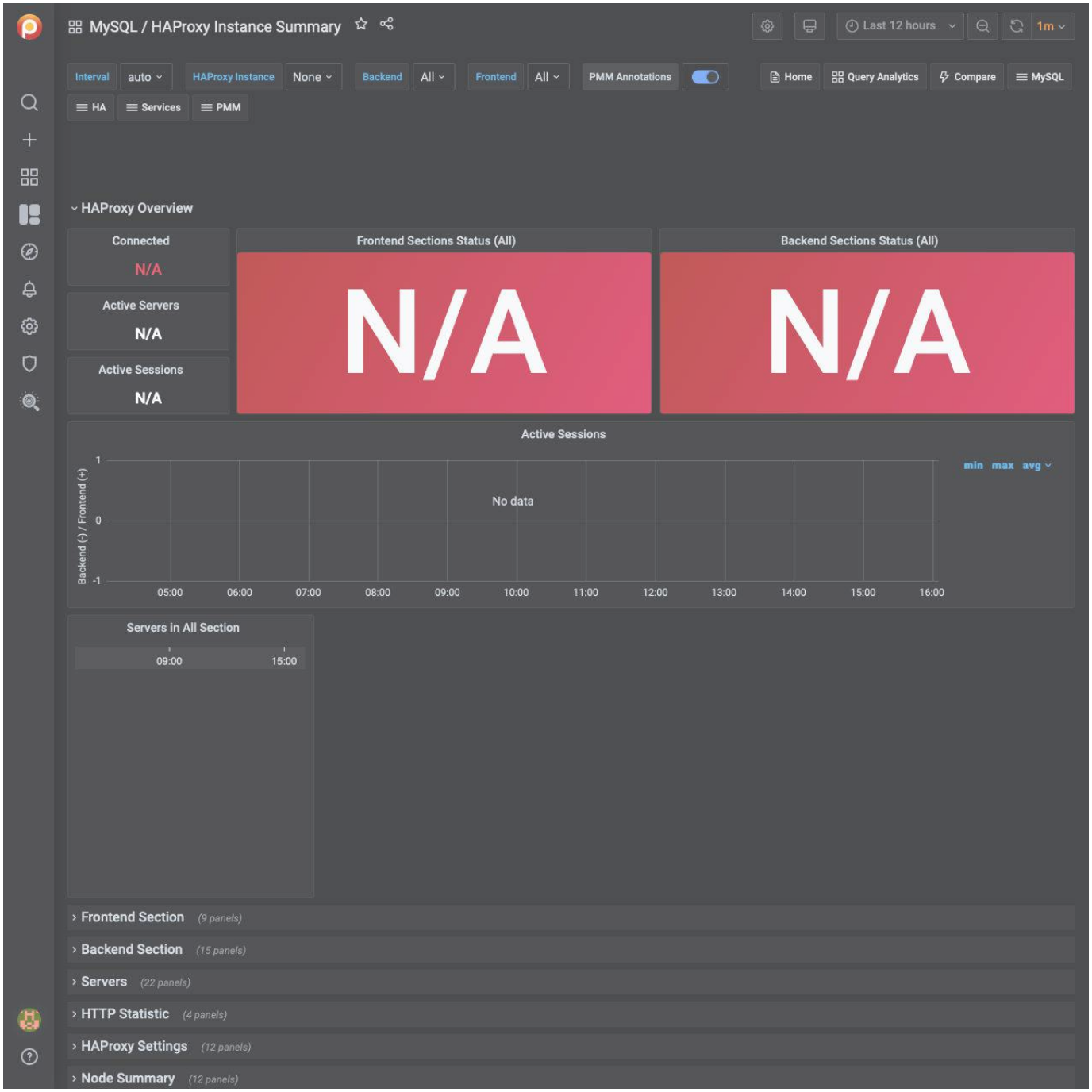
If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-05-11

Percona LLC and/or its affiliates ©2024

HAPROXY INSTANCE SUMMARY



No description.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2021-06-03

6.9 Commands

6.9.1 Commands

- `pmm-admin` – Command line tool for configuring and administering PMM
- `pmm-agent` – Daemon process, communicating between PMM Client and PMM Server

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-03-11

6.9.2 pmm-admin - PMM Administration Tool

NAME

`pmm-admin` - Administer PMM

SYNOPSIS

`pmm-admin [FLAGS]`

`pmm-admin config [FLAGS] --server-url=server-url`

`pmm-admin add DATABASE [FLAGS] [NAME] [ADDRESS]`

DATABASE:= [MongoDB | MySQL | PostgreSQL | ProxySQL]

`pmm-admin add --pmm-agent-listen-port=LISTEN_PORT DATABASE [FLAGS] [NAME] [ADDRESS]`

`pmm-admin add haproxy [FLAGS] [NAME]`

`pmm-admin add external [FLAGS] [NAME] [ADDRESS]`

`pmm-admin add external-serverless [FLAGS] [NAME] [ADDRESS]`

`pmm-admin remove [FLAGS] service-type [service-name]`

`pmm-admin register [FLAGS] [node-address] [node-type] [node-name]`

`pmm-admin list [FLAGS] [node-address]`

`pmm-admin status [FLAGS] [node-address]`

`pmm-admin summary [FLAGS] [node-address]`

`pmm-admin annotate [--node|--service] [--tags <tags>] [node-name|service-name]`

Percona LLC and/or its affiliates ©2024

```
pmm-admin help [COMMAND]
```

DESCRIPTION

`pmm-admin` is a command-line tool for administering PMM using a set of COMMAND keywords and associated FLAGS.

PMM communicates with the PMM Server via a PMM agent process.

COMMON FLAGS

`-h`, `--help`

Show help and exit.

`--help-long`

Show extended help and exit.

`--help-man`

Generate `man` page. (Use `pmm-admin --help-man | man -l -` to view.)

`--debug`

Enable debug logging.

`--trace`

Enable trace logging (implies debug).

`--log-level` (This parameter is available starting with PMM 2.29.0.)

Set the level for the logs as per your requirement such as INFO, WARNING, ERROR, and FATAL.

`--json`

Enable JSON output.

`--version`

Show the application version and exit.

`--server-url=server-url`

PMM Server URL in `https://username:password@pmm-server-host/` format.

`--server-insecure-tls`

Skip PMM Server TLS certificate validation.

`--group=<group-name>`

Group name for external services. Default: `external`

Percona LLC and/or its affiliates ©2024

`--expose-exporter` (This flag is availble starting with PMM 2.41.0.)

If you enable this flag, any IP address on the local network and anywhere on the internet can access exporter endpoints. If the flag is disabled/not present, exporter endpoints can be accessed only locally. The flag is disabled by default

COMMANDS

GENERAL COMMANDS

`pmm-admin help [COMMAND]`

Show help for `COMMAND`.

INFORMATION COMMANDS

`pmm-admin list --server-url=server-url [FLAGS]`

Show Services and Agents running on this Node, and the agent mode (push/pull).

`pmm-admin status --server-url=server-url [FLAGS]`

Show the following information about a local pmm-agent, and its connected server and clients:

- Agent: Agent ID, Node ID.
- PMM Server: URL and version.
- PMM Client: connection status, time drift, latency, `vmagent` status, `pmm-admin` version.
- Agents: Agent ID path and client name.

FLAGS:

`--wait=<period><unit>`

Time to wait for a successful response from pmm-agent. *period* is an integer. *unit* is one of `ms` for milliseconds, `s` for seconds, `m` for minutes, `h` for hours.

`pmm-admin summary --server-url=server-url [FLAGS]`

Creates an archive file in the current directory with default file name `summary_<hostname>_<year>_<month>_<date>_<hour>_<minute>_<second>.zip`. The contents are two directories, `client` and `server` containing diagnostic text files.

FLAGS:

`--filename="filename"`

The Summary Archive filename.

`--skip-server`

Skip fetching `logs.zip` from PMM Server.

`--pprof` (This parameter is available starting with PMM 2.29.0)

Include performance profiling data in the summary.

Percona LLC and/or its affiliates ©2024

CONFIGURATION COMMANDS

pmm-admin config

`pmm-admin config [FLAGS] [node-address] [node-type] [node-name]`

Configure a local `pmm-agent`.

FLAGS:

`--node-id=node-id`

Node ID (default is auto-detected).

`--node-model=node-model`

Node model.

`--region=region`

Node region.

`--az=availability-zone`

Node availability zone.

`--metrics-mode=mode`

Metrics flow mode for agents node-exporter. Allowed values: - `auto`: chosen by server (default). - `push`: agent will push metrics. - `pull`: server scrapes metrics from agent.

`--paths-base=dir`

Base path where all binaries, tools and collectors of PMM client are located

`--agent-password=password` (This parameter i available starting with PMM 2.29.0.)

Custom agent password.

pmm-admin register

`pmm-admin register [FLAGS] [node-address] [node-type] [node-name]`

Register the current Node with the PMM Server.

`--server-url=server-url`

PMM Server URL in `https://username:password@pmm-server-host/` format.

`--machine-id="/machine_id/9812826a1c45454a98ba45c56cc4f5b0"`

Node machine-id (default is auto-detected).

`--distro="linux"`

Node OS distribution (default is auto-detected).

`--container-id=container-id`

Container ID.

Percona LLC and/or its affiliates ©2024

`--container-name=container-name`

Container name.

`--node-model=node-model`

Node model.

`--region=region`

Node region.

`--az=availability-zone`

Node availability zone.

`--custom-labels=labels`

Custom user-assigned labels.

`--agent-password=password` (This parameter is available starting with PMM 2.29.0.)

Custom agent password.

`pmm-admin add --pmm-agent-listen-port=LISTEN_PORT`

`pmm-admin add --pmm-agent-listen-port=LISTEN_PORT DATABASE [FLAGS] [NAME] [ADDRESS]`

Configure the PMM agent with a listen port.

`--pmm-agent-listen-port=LISTEN_PORT`

The PMM agent listen port.

DATABASE:= [MongoDB | MySQL | PostgreSQL | ProxySQL]

`pmm-admin remove`

`pmm-admin remove [FLAGS] service-type [service-name]`

Remove Service from monitoring.

`--service-id=service-id`

Service ID.

`--force`

Remove service with that name or ID and all dependent services and agents.

When you remove a service, collected data remains on PMM Server for the specified retention period.

`pmm-admin annotate`

`pmm-admin annotate [--node|--service] <annotation> [--tags <tags>] [--node-name=<node>] [--service-name=<service>]`

Annotate an event. (Read more)

`<annotation>`

The annotation string. If it contains spaces, it should be quoted.

Percona LLC and/or its affiliates ©2024

`--node`

Annotate the current node or that specified by `--node-name`.

`--service`

Annotate all services running on the current node, or that specified by `--service-name`.

`--tags`

A quoted string that defines one or more comma-separated tags for the annotation. Example: `"tag 1,tag 2"`.

`--node-name`

The node name being annotated.

`--service-name`

The service name being annotated.

Combining flags

Flags may be combined as shown in the following examples.

`--node`

Current node.

`--node-name`

Node with name.

`--node --node-name=NODE_NAME`

Node with name.

`--node --service-name`

Current node and service with name.

`--node --node-name --service-name`

Node with name and service with name.

`--node --service`

Current node and all services of current node.

`-node --node-name --service --service-name`

Service with name and node with name.

`--service`

All services of the current node.

Percona LLC and/or its affiliates ©2024

`--service-name`

Service with name.

`--service --service-name`

Service with name.

`--service --node-name`

All services of current node and node with name.

`--service-name --node-name`

Service with name and node with name.

`--service --service-name -node-name`

Service with name and node with name.

> **Tip**
>
> If node or service name is specified, they are used instead of other parameters.

DATABASE COMMANDS

MongoDB

`pmm-admin add mongodb [FLAGS] [node-name] [node-address]`

Add MongoDB to monitoring.

FLAGS:

`--node-id=node-id`

Node ID (default is auto-detected).

`--pmm-agent-id=pmm-agent-id`

The pmm-agent identifier which runs this instance (default is auto-detected).

`--username=username`

MongoDB username.

`--password=password`

MongoDB password.

`--agent-password=password`

Override the default password for accessing the `/metrics` endpoint. (Username is `pmm` and default password is the agent ID.)

> Avoid using special characters like ‘‘, ‘;’ and ‘$’ in the custom password.

Percona LLC and/or its affiliates ©2024

`--query-source=profiler`

Source of queries, one of: `profiler`, `none` (default: `profiler`).

`--environment=environment`

Environment name.

`--cluster=cluster`

Cluster name.

`--replication-set=replication-set`

Replication set name.

`--custom-labels=custom-labels`

Custom user-assigned labels.

`--skip-connection-check`

Skip connection check.

`--tls`

Use TLS to connect to the database.

`--tls-skip-verify`

Skip TLS certificates validation.

`--tls-certificate-key-file=PATHTOCERT`

Path to TLS certificate file.

`--tls-certificate-key-file-password=IFPASSWORDTOCERTISSET`

Password for TLS certificate file.

`--tls-ca-file=PATHTOCACERT`

Path to certificate authority file.

`--metrics-mode=mode`

Metrics flow mode for agents node-exporter. Allowed values: - `auto`: chosen by server (default). - `push`: agent will push metrics. - `pull`: server scrapes metrics from agent.

`--max-query-length=NUMBER` (This parameter is available starting with PMM 2.32.0.)

Limit query length in QAN. Allowed values: - -1: No limit. - 0: Default value. The default value is 2048 chars. - >0: Query will be truncated after chars.

> Ensure you do not set the value of `max-query-length` to 1, 2, or 3. Otherwise, the PMM agent will get terminated.

Advanced Options

Percona LLC and/or its affiliates ©2024

PMM starts the MongoDB exporter by default only with `diagnosticdata` and `replicasetstatus` collectors enabled.

FLAGS:

`--enable-all-collectors`

Enable all collectors.

`--disable-collectors`

Comma-separated list of collector names to exclude from exporter.

`--max-collections-limit=-1`

Disable collstats, dbstats, topmetrics and indexstats if there are more than collections. 0: No limit. Default is -1, PMM automatically sets this value.

> A very high limit of `max-collections-limit` could impact the CPU and Memory usage. Check `--stats-collections` to limit the scope of collections and DB's metrics to be fetched.

`--stats-collections=db1,db2.col1`

Collections for collstats & indexstats.

Enable all collectors

To enable all collectors, pass the parameter `--enable-all-collectors` in the `pmm-admin add mongodb` command. This will enable `collstats`, `dbstats`, `indexstats`, and `topmetrics` collectors.

Disable some collectors

To enable only some collectors, pass the parameter `--enable-all-collectors` along with the parameter `--disable-collectors`.

For example, if you want all collectors except `topmetrics`, specify:

```
--enable-all-collectors --disable-collectors=topmetrics
```

Limit `dbStats`, `collStats` and `indexStats`

By default, PMM decides the limit for the number of collections to monitor the `collStats` and `indexStats` collectors.

You can also set an additional limit for the `collStats`, `indexStats`, `dbStats`, and `topmetrics` collectors with the `--max-collections-limit` parameter.

Set the value of the parameter `--max-collections-limit` to:

- 0: which indicates that `collStats` and `indexStats` can handle unlimited collections.
- n, which indicates that `collStats` and `indexStats` can handle <=n collections. If the limit is crossed - exporter stops collecting monitoring data for the `collStats` and `indexStats` collectors.
- -1 (default) doesn't need to be explicitly set. It indicates that PMM decides how many collections it would monitor, currently <=200 (subject to change).

To further limit collections to monitor, enable `collStats` and `indexStats` for some databases or collections:

- Specify the databases and collections that `collStats` and `indexStats` will use to collect data using the parameter `--stats-collections`. This parameter receives a comma-separated list of name spaces in the form `database[.collection]`.

Examples

Percona LLC and/or its affiliates ©2024

To add MongoDB with all collectors (`diagnosticdata`, `replicasetstatus`, `collstats`, `dbstats`, `indexstats`, and `topmetrics`) with default limit detected by PMM (currently <=200 collections, but subject to change):

```
pmm-admin add mongodb --username=admin --password=admin_pass --enable-all-collectors mongodb_srv_1 127.0.0.1:27017
```

To add MongoDB with all collectors (`diagnosticdata`, `replicasetstatus`, `collstats`, `dbstats`, `indexstats`, and `topmetrics`) with `max-collections-limit` set to 1000:

```
pmm-admin add mongodb --username=admin --password=admin_pass --enable-all-collectors --max-collections-limit=1000 mongodb_srv_1 127.0.0.1:27017
```

To enable all the collectors with an unlimited number of collections monitored:

```
pmm-admin add mongodb --username=admin --password=admin_pass --enable-all-collectors --max-collections-limit=0 mongodb_srv_1 127.0.0.1:27017
```

To add MongoDB with default collectors (`diagnosticdata` and `replicasetstatus`):

```
pmm-admin add mongodb --username=admin --password=admin_pass mongodb_srv_1 127.0.0.1:27017
```

Disable `collstats` collector and enable all the others without limiting `max-collections-limit`:

```
pmm-admin add mongodb --username=admin --password=admin_pass --enable-all-collectors --max-collections-limit=0 --disable-collectors=collstats mongodb_srv_1 127.0.0.1:27017
```

If `--stats-collections=db1,db2.col1` then the collectors are run as follows:

Database	Collector is run on
`db1`	All the collections
`db2`	**Only** for collection `col1`

Enable all collectors and limit monitoring for `dbstats`, `indexstats`, `collstats` and `topmetrics` for all collections in `db1` and `col1` collection in `db2`, without limiting `max-collections-limit` for a number of collections in `db1`:

```
pmm-admin add mongodb --username=admin --password=admin_pass --enable-all-collectors --max-collections-limit=0 --stats-collections=db1,db2.col1 mongodb_srv_1 127.0.0.1:27017
```

Resolutions

PMM collects metrics in two resolutions to decrease CPU and Memory usage: high and low resolutions.

In high resolution we collect metrics from collectors which work fast: - `diagnosticdata` - `replicasetstatus` - `topmetrics`

In low resolution we collect metrics from collectors which could take some time: - `dbstats` - `indexstats` - `collstats`

MySQL

```
pmm-admin add mysql [FLAGS] node-name node-address | [--name=service-name] --address=address[:port] | --socket
```

Add MySQL to monitoring.

FLAGS:

`--address`

MySQL address and port (default: 127.0.0.1:3306).

`--socket=socket`

Path to MySQL socket. (Find the socket path with `mysql -u root -p -e "select @@socket"`.)

Percona LLC and/or its affiliates ©2024

`--node-id=node-id`

Node ID (default is auto-detected).

`--pmm-agent-id=pmm-agent-id`

The pmm-agent identifier which runs this instance (default is auto-detected).

`--username=username`

MySQL username.

`--password=password`

MySQL password.

`--agent-password=password`

Override the default password for accessing the `/metrics` endpoint. (Username is `pmm` and default password is the agent ID.)

> Avoid using special characters like ‘‘, ‘;’ and ‘$’ in the custom password.

`--query-source=slowlog`

Source of SQL queries, one of: `slowlog`, `perfschema`, `none` (default: `slowlog`). For `slowlog` query source, you need change permissions for specific files. Root permissions are needed for this.

`--size-slow-logs=N`

Rotate slow log file at this size. If `0`, use server-defined default. Negative values disable log rotation. A unit suffix must be appended to the number and can be one of:

- `KiB`, `MiB`, `GiB`, `TiB` for base 2 units (1024, 1048576, etc).

`--disable-queryexamples`

Disable collection of query examples.

`--disable-tablestats`

Disable table statistics collection.

Excluded collectors for low-resolution time intervals:

- `--collect.auto_increment.columns`
- `--collect.info_schema.tables`
- `--collect.info_schema.tablestats`
- `--collect.perf_schema.indexiowaits`
- `--collect.perf_schema.tableiowaits`
- `--collect.perf_schema.file_instances`

Excluded collectors for medium-resolution time intervals:

- `--collect.perf_schema.tablelocks`

Percona LLC and/or its affiliates ©2024

`--disable-tablestats-limit=disable-tablestats-limit`

Table statistics collection will be disabled if there are more than specified number of tables (default: server-defined). 0=no limit. Negative value disables collection.

`--environment=environment`

Environment name.

`--cluster=cluster`

Cluster name.

`--replication-set=replication-set`

Replication set name.

`--custom-labels=custom-labels`

Custom user-assigned labels.

`--skip-connection-check`

Skip connection check.

`--tls`

Use TLS to connect to the database.

`--tls-skip-verify`

Skip TLS certificates validation.

`--tls-cert-file=PATHTOCERT`

Path to TLS client certificate file.

`--tls-key=PATHTOCERTKEY`

Key for TLS client certificate file.

`--tls-ca-file=PATHTOCACERT`

Path to certificate authority file.

`--ssl-ca=PATHTOCACERT`

The path name of the Certificate Authority (CA) certificate file. If used, must specify the same certificate used by the server. (-ssl-capath is similar, but specifies the path name of a directory of CA certificate files.)

`--ssl-cert=PATHTOCERTKEY`

The path name of the client public key certificate file.

`--ssl-key`

The path name of the client private key file.

Percona LLC and/or its affiliates ©2024

`--ssl-skip-verify`

Skip SSL certificate verification.

`--metrics-mode=mode`

Metrics flow mode for agents node-exporter. Allowed values: - `auto`: chosen by server (default). - `push`: agent will push metrics. - `pull`: server scrapes metrics from agent.

`--max-query-length=NUMBER` (This parameter is available starting with PMM 2.32.0.)

Limit query length in QAN. Allowed values: - -1: No limit. - 0: Default value. The default value is 2048 chars. - >0: Query will be truncated after chars.

> Ensure you do not set the value of `max-query-length` to 1, 2, or 3. Otherwise, the PMM agent will get terminated.

`--comments-parsing=off/on`

Enable/disable parsing comments from queries into QAN filter groups: - off: Disabled. - on: Enabled.

PostgreSQL

`pmm-admin add postgresql [FLAGS] [node-name] [node-address]`

Add PostgreSQL to monitoring.

FLAGS:

`--node-id=<node id>`

Node ID (default is auto-detected).

`--pmm-agent-id=<pmm agent id>`

The pmm-agent identifier which runs this instance (default is auto-detected).

`--username=<username>`

PostgreSQL username.

`--password=<password>`

PostgreSQL password.

`--database=<database>`

PostgreSQL database (default: postgres).

`--agent-password=password`

Override the default password for accessing the `/metrics` endpoint. (Username is `pmm` and default password is the agent ID.)

> Avoid using special characters like ‘', ‘;’ and ‘$’ in the custom password.

Percona LLC and/or its affiliates ©2024

`--query-source=<query source>`

Source of SQL queries, one of: `pgstatements`, `pgstatmonitor`, `none` (default: `pgstatements`).

`--disable-queryexamples`

Disable collection of query examples. Applicable only if `query-source` is set to `pgstatmonitor`.

`--environment=<environment>`

Environment name.

`--cluster=<cluster>`

Cluster name.

`--replication-set=<replication set>`

Replication set name.

`--custom-labels=<custom labels>`

Custom user-assigned labels.

`--skip-connection-check`

Skip connection check.

`--tls`

Use TLS to connect to the database.

`--tls-skip-verify`

Skip TLS certificates validation.

`--tls-ca-file`

TLS CA certificate file.

`--tls-cert-file`

TLS certificate file.

`--tls-key-file`

TLS certificate key file.

`--metrics-mode=mode`

Metrics flow mode for agents node-exporter. Allowed values: - `auto`: chosen by server (default). - `push`: agent will push metrics. - `pull`: server scrapes metrics from agent.

`--max-query-length=NUMBER` (This parameter is available starting with PMM 2.32.0.)

Limit query length in QAN. Allowed values: - -1: No limit. - 0: Default value. The default value is 2048 chars. - >0: Query will be truncated after chars.

Percona LLC and/or its affiliates ©2024

> Ensure you do not set the value of `max-query-length` to 1, 2, or 3. Otherwise, the PMM agent will get terminated.

`--comments-parsing=off/on`

Enable/disable parsing comments from queries into QAN filter groups: - off: Disabled. - on: Enabled.

ProxySQL

`pmm-admin add proxysql [FLAGS] [node-name] [node-address]`

Add ProxySQL to monitoring.

FLAGS:

`--node-id=node-id`

Node ID (default is auto-detected).

`--pmm-agent-id=pmm-agent-id`

The pmm-agent identifier which runs this instance (default is auto-detected).

`--username=username`

ProxySQL username.

`--password=password`

ProxySQL password.

`--agent-password=password`

Override the default password for accessing the `/metrics` endpoint. (Username is `pmm` and default password is the agent ID.)

> Avoid using special characters like ‘', ‘;’ and ‘$’ in the custom password.

`--environment=environment`

Environment name.

`--cluster=cluster`

Cluster name.

`--replication-set=replication-set`

Replication set name.

`--custom-labels=custom-labels`

Custom user-assigned labels.

`--skip-connection-check`

Skip connection check.

Percona LLC and/or its affiliates ©2024

`--tls`

Use TLS to connect to the database.

`--tls-skip-verify`

Skip TLS certificates validation.

`--metrics-mode=mode`

Metrics flow mode for agents node-exporter. Allowed values: - `auto`: chosen by server (default). - `push`: agent will push metrics. - `pull`: server scrapes metrics from agent.

`--disable-collectors`

Comma-separated list of collector names to exclude from exporter.

HAProxy

`pmm-admin add haproxy [FLAGS] [NAME]`

Add HAProxy to monitoring.

FLAGS:

`--server-url=SERVER-URL`

PMM Server URL in `https://username:password@pmm-server-host/` format.

`--server-insecure-tls`

Skip PMM Server TLS certificate validation.

`--username=USERNAME`

HAProxy username.

`--password=PASSWORD`

HAProxy password.

`--scheme=SCHEME`

Scheme to generate URI to exporter metrics endpoints (http or https).

`--metrics-path=METRICS-PATH`

Path under which metrics are exposed, used to generate URI (default: /metrics).

`--listen-port=LISTEN-PORT`

Listen port of haproxy exposing the metrics for scraping metrics (Required).

`--service-node-id=SERVICE-NODE-ID`

Node ID where service runs (default is auto-detected).

`--environment=ENVIRONMENT`

Environment name like ‘production’ or ‘qa’.

Percona LLC and/or its affiliates ©2024

`--cluster=CLUSTER`

Cluster name.

`--replication-set=REPLICATION-SET`

Replication set name.

`--custom-labels=CUSTOM-LABELS`

Custom user-assigned labels. Example: region=east,app=app1.

`--metrics-mode=MODE`

Metrics flow mode for agents node-exporter. Allowed values: - `auto`: chosen by server (default). - `push`: agent will push metrics. - `pull`: server scrapes metrics from agent.

`--skip-connection-check`

Skip connection check.

OTHER COMMANDS

`pmm-admin add external [FLAGS]`

Add External source of data (like a custom exporter running on a port) to be monitored.

FLAGS:

`--service-name="current-hostname"`

Service name (autodetected defaults to the hostname where `pmm-admin` is running).

`--agent-node-id=AGENT-NODE-ID`

Node ID where agent runs (default is autodetected).

`--username=USERNAME`

External username.

`--password=PASSWORD`

External password.

`--scheme=http or https`

Scheme to generate URI to exporter metrics endpoints.

`--metrics-path=/metrics`

Path under which metrics are exposed, used to generate URI.

`--listen-port=LISTEN-PORT`

Listen port of external exporter for scraping metrics. (Required.)

`--service-node-id=SERVICE-NODE-ID`

Node ID where service runs (default is autodetected).

Percona LLC and/or its affiliates ©2024

`--environment=prod`

Environment name like ‘production’ or ‘qa’.

`--cluster=east-cluster`

Cluster name.

`--replication-set=rs1`

Replication set name.

`--custom-labels=CUSTOM-LABELS`

Custom user-assigned labels. Example: `region=east,app=app1`.

`--metrics-mode=auto`

Metrics flow mode, can be `push`: agent will push metrics, `pull`: server scrape metrics from agent or `auto`: chosen by server.

`--group="external"`

Group name of external service. (Default: `external`.)

`pmm-admin add external-serverless [FLAGS]`

Add External Service on Remote node to monitoring.

Usage example: `pmm-admin add external-serverless --url=http://1.2.3.4:9093/metrics`.

Also, individual parameters can be set instead of `--url` like: `pmm-admin add external-serverless --scheme=http --host=1.2.3.4 --listen-port=9093 --metrics-path=/metrics --container-name=ddd --external-name=e125`.

Note that some parameters are mandatory depending on the context. For example, if you specify `--url`, `--schema` and other related parameters are not mandatory. But if you specify `--host` you must provide all other parameters needed to build the destination URL, or you can specify `--address` instead of host and port as individual parameters.

FLAGS:

`--url=URL`

Full URL to exporter metrics endpoints.

`--scheme=https`

Scheme to generate URL to exporter metrics endpoints.

`--username=USERNAME`

External username.

`--password=PASSWORD`

External password.

`--address=1.2.3.4:9000`

External exporter address and port.

Percona LLC and/or its affiliates ©2024

`--host=1.2.3.4`

External exporters hostname or IP address.

`--listen-port=9999`

Listen port of external exporter for scraping metrics.

`--metrics-path=/metrics`

Path under which metrics are exposed, used to generate URL.

`--environment=testing`

Environment name.

`--cluster=CLUSTER`

Cluster name.

`--replication-set=rs1`

Replication set name.

`--custom-labels='app=myapp,region=s1'`

Custom user-assigned labels.

`--group="external"`

Group name of external service. (Default: `external`.)

`--machine-id=MACHINE-ID`

Node machine-id.

`--distro=DISTRO`

Node OS distribution.

`--container-id=CONTAINER-ID`

Container ID.

`--container-name=CONTAINER-NAME`

Container name.

`--node-model=NODE-MODEL`

Node model.

`--region=REGION`

Node region.

`--az=AZ`

Node availability zone.

Percona LLC and/or its affiliates ©2024

EXAMPLES

```
pmm-admin add mysql --query-source=slowlog --username=pmm --password=pmm sl-mysql 127.0.0.1:3306
```

```
MySQL Service added.
Service ID  : /service_id/a89191d4-7d75-44a9-b37f-a528e2c4550f
Service name: sl-mysql
```

```
pmm-admin add mysql --username=pmm --password=pmm --service-name=ps-mysql --host=127.0.0.1 --port=3306
```

```
pmm-admin status
pmm-admin status --wait=30s
```

```
Agent ID: /agent_id/c2a55ac6-a12f-4172-8850-4101237a4236
Node ID : /node_id/29b2cc24-3b90-4892-8d7e-4b44258d9309
PMM Server:
 URL : https://x.x.x.x:443/
 Version: 2.5.0
PMM Client:
 Connected : true
 Time drift: 2.152715ms
 Latency : 465.658µs
 pmm-admin version: 2.5.0
 pmm-agent version: 2.5.0
Agents:
 /agent_id/aeb42475-486c-4f48-a906-9546fc7859e8 mysql_slowlog_agent Running
```

DISABLE COLLECTORS

```
pmm-admin add mysql --disable-collectors='heartbeat,global_status,info_schema.innodb_cmp' --username=pmm --password=pmm --service-name=db1-mysql --host=127.
0.0.1 --port=3306
```

For other collectors that you can disable with the `--disable-collectors` option, please visit the official repositories for each exporter:

- `node_exporter`
- `mysqld_exporter`
- `mongodb_exporter`
- `postgres_exporter`
- `proxysql_exporter`

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-12-12

6.9.3 pmm-agent - PMM Client agent

NAME

`pmm-agent` - The PMM Client daemon program.

Percona LLC and/or its affiliates ©2024

SYNOPSIS

```
pmm-agent [command] [options]
```

DESCRIPTION

`pmm-agent`, part of the PMM Client package, runs as a daemon process on all monitored hosts.

COMMANDS

`pmm-agent run`

Run pmm-agent (default).

`pmm-agent setup [node-address] [node-type] [node-name]`

Configure local pmm-agent (requires root permissions)

`pmm-agent help [command]`

Show help (for command) and exit.

OPTIONS AND ENVIRONMENT

Most options can be set via environment variables (shown in parentheses).

Option	Environment variable	Description
`--server-password=SERVER-PASSWORD`	`PMM_AGENT_SERVER_PASSWORD`	Password to connect to PMM Server.
`--server-username=SERVER-USERNAME`	`PMM_AGENT_SERVER_USERNAME`	Username to connect to PMM Server.
`--server-address=host:port`	`PMM_AGENT_SERVER_ADDRESS`	PMM Server address and port number.
`--server-insecure-tls`	`PMM_AGENT_SERVER_INSECURE_TLS`	Skip PMM Server TLS certificate validation.
`--az=AZ`	`PMM_AGENT_SETUP_AZ`	Node availability zone.
`--config-file=path_to/pmm-agent.yaml`	`PMM_AGENT_CONFIG_FILE`	Configuration file path and name.
`--container-id=CONTAINER-ID`	`PMM_AGENT_SETUP_CONTAINER_ID`	Container ID.
`--container-name=CONTAINER-NAME`	`PMM_AGENT_SETUP_CONTAINER_NAME`	Container name.
`--debug`	`PMM_AGENT_DEBUG`	Enable debug output.
`--distro=distro`	`PMM_AGENT_SETUP_DISTRO`	Node OS distribution (default is auto-detected).
`--force`	`PMM_AGENT_SETUP_FORCE`	Remove Node with that name and all dependent Services and Agents (if existing).
`--id=/agent_id/...`	`PMM_AGENT_ID`	ID of this pmm-agent.
`--listen-address=LISTEN-ADDRESS`	`PMM_AGENT_LISTEN_ADDRESS`	Agent local API address.
`--listen-port=LISTEN-PORT`	`PMM_AGENT_LISTEN_PORT`	Agent local API port.
`--machine-id=machine-id`	`PMM_AGENT_SETUP_MACHINE_ID`	Node machine ID (default is auto-detected).
`--metrics-mode=auto`	`PMM_AGENT_SETUP_METRICS_MODE`	Metrics flow mode for agents node-exporter. Can be `push` (agent will push metrics), `pull` (server scrapes metrics from agent) or `auto` (chosen by server).

Percona LLC and/or its affiliates ©2024

Option	Environment variable	Description
`--node-model=NODE-MODEL`	`PMM_AGENT_SETUP_NODE_MODEL`	Node model.
`--paths-base=PATH`	`PMM_AGENT_PATHS_BASE`	Base path for PMM client, where all binaries, tools and collectors are located. If not set, default is `/usr/local/percona/pmm2`.
`--paths-exporters_base=PATH`	`PMM_AGENT_PATHS_EXPORTERS_BASE`	Base path for exporters to use. If not set, or set to a relative path, uses value of `--paths-base` prepended to it.
`--paths-mongodb_exporter=PATH`	`PMM_AGENT_PATHS_MONGODB_EXPORTER`	Path to `mongodb_exporter`.
`--paths-mysqld_exporter=PATH`	`PMM_AGENT_PATHS_MYSQLD_EXPORTER`	Path to `mysqld_exporter`.
`--paths-node_exporter=PATH`	`PMM_AGENT_PATHS_NODE_EXPORTER`	Path to `node_exporter`.
`--paths-postgres_exporter=PATH`	`PMM_AGENT_PATHS_POSTGRES_EXPORTER`	Path to `postgres_exporter`.
`--paths-proxysql_exporter=PATH`	`PMM_AGENT_PATHS_PROXYSQL_EXPORTER`	Path to `proxysql_exporter`.
`--paths-pt-summary=PATH`	`PMM_AGENT_PATHS_PT_SUMMARY`	Path to `pt-summary`.
`--paths-pt-mysql-summary=PATH`	`PMM_AGENT_PATHS_PT_MYSQL_SUMMARY`	Path to `pt-mysql-summary`.
`--paths-pt-pg-summary=PATH`	`PMM_AGENT_PATHS_PT_PG_SUMMARY`	Path to `pt-pg-summary`.
`--paths-tempdir=PATH`	`PMM_AGENT_PATHS_TEMPDIR`	Temporary directory for exporters.
`--ports-max=PORTS-MAX`	`PMM_AGENT_PORTS_MAX`	Highest allowed port number for listening sockets.
`--ports-min=PORTS-MIN`	`PMM_AGENT_PORTS_MIN`	Lowest allowed port number for listening sockets.
`--region=REGION`	`PMM_AGENT_SETUP_REGION`	Node region.
`--skip-registration`	`PMM_AGENT_SETUP_SKIP_REGISTRATION`	Skip registration on PMM Server.
`--trace`	`PMM_AGENT_TRACE`	Enable trace output (implies `--debug`).
`-h`, `--help`		Show help (synonym for `pmm-agent help`).
`--version`		Show application version, PMM version, time-stamp, git commit hash and branch.
`--expose-exporter` (This flag is available starting with PMM 2.41.0.)		If you enable this flag, any IP address on the local network and anywhere on the internet can access node exporter endpoints. If the flag is disabled, node exporter endpoints can be accessed only locally.

USAGE AND EXAMPLES OF `paths-base` FLAG

Since 2.23.0 this flag could be used for easier setup of pmm agent. With this flag, the root permissions for PMM client aren’t needed anymore, and it will be

Since 2.23.0 this flag could be used for easier setup of PMM agent. With this flag the root permissions for PMM client aren’t needed anymore and it will be fully working.

Percona LLC and/or its affiliates ©2024

Examples:

Percona LLC and/or its affiliates ©2024

- **Case 1:** There are no root permissions for `/usr/local/percona/pmm2` folder or there is a need to change default folder for PMM files. Command:

```
pmm-agent setup --paths-base=/home/user/custom/pmm2 --config-file=pmm-agent-dev.yaml --server-insecure-tls --server-address=127.0.0.1:443 --server-
username=admin --server-password=admin
```

Config output:

```
# Updated by `pmm-agent setup`.
---
id: /agent_id/be568008-b1b4-4bd9-98c7-392d1f4b724e
listen-address: 127.0.0.1
listen-port: 7777
server:
    address: 127.0.0.1:443
    username: admin
    password: admin
    insecure-tls: true
paths:
    paths_base: /home/user/custom/pmm2
    exporters_base: /home/user/custom/pmm2/exporters
    node_exporter: /home/user/custom/pmm2/exporters/node_exporter
    mysqld_exporter: /home/user/custom/pmm2/exporters/mysqld_exporter
    mongodb_exporter: /home/user/custom/pmm2/exporters/mongodb_exporter
    postgres_exporter: /home/user/custom/pmm2/exporters/postgres_exporter
    proxysql_exporter: /home/user/custom/pmm2/exporters/proxysql_exporter
    rds_exporter: /home/user/custom/pmm2/exporters/rds_exporter
    azure_exporter: /home/user/custom/pmm2/exporters/azure_exporter
    vmagent: /home/user/custom/pmm2/exporters/vmagent
    tempdir: /tmp
    pt_summary: /home/user/custom/pmm2/tools/pt-summary
    pt_pg_summary: /home/user/custom/pmm2/tools/pt-pg-summary
    pt_mysql_summary: /home/user/custom/pmm2/tools/pt-mysql-summary
    pt_mongodb_summary: /home/user/custom/pmm2/tools/pt-mongodb-summary
ports:
    min: 42000
    max: 51999
debug: false
trace: false
```

As could be seen above, base for all exporters and tools was changed only by setting `--paths-base`. With this tag the folder for PMM that doesn't require root access could be specified.

- **Case 2:** The older `--paths-exporters_base` flag could be passed along with the `--paths-base` Command:

```
pmm-agent setup --paths-base=/home/user/custom/pmm2 --paths-exporters_base=/home/user/exporters --config-file=pmm-agent-dev.yaml --server-insecure-tls --
server-address=127.0.0.1:443 --server-username=admin --server-password=admin
```

Config output:

```
# Updated by `pmm-agent setup`.
---
id: /agent_id/afce1917-8836-4857-b3e5-ad372c2ddbe5
listen-address: 127.0.0.1
listen-port: 7777
server:
    address: 127.0.0.1:443
    username: admin
    password: admin
    insecure-tls: true
paths:
    paths_base: /home/user/custom/pmm2
    exporters_base: /home/user/exporters
    node_exporter: /home/user/exporters/node_exporter
    mysqld_exporter: /home/user/exporters/mysqld_exporter
    mongodb_exporter: /home/user/exporters/mongodb_exporter
    postgres_exporter: /home/user/exporters/postgres_exporter
    proxysql_exporter: /home/user/exporters/proxysql_exporter
    rds_exporter: /home/user/exporters/rds_exporter
    azure_exporter: /home/user/exporters/azure_exporter
    vmagent: /home/user/exporters/vmagent
    tempdir: /tmp
    pt_summary: /home/user/custom/pmm2/tools/pt-summary
    pt_pg_summary: /home/user/custom/pmm2/tools/pt-pg-summary
    pt_mysql_summary: /home/user/custom/pmm2/tools/pt-mysql-summary
```


Percona LLC and/or its affiliates ©2024

```
    pt_mongodb_summary: /home/user/custom/pmm2/tools/pt-mongodb-summary
ports:
    min: 42000
    max: 51999
debug: false
trace: false
```

As could be seen above the behavior for the `--paths-base` was the same, but paths for all exporters were overwritten by the `--paths-exporter_base` flag.

Summary: Flag `--paths-base` will set path for all exporters and tools, but each one could be overridden by specific flag (like `--paths-mongodb_exporter`, `--paths-pt-mysql-summary` and etc).

LOGGING

By default, pmm-agent sends messages to stderr and to the system log (`syslogd` or `journald` on Linux).

To get a separate log file, edit the `pmm-agent` start-up script.

`systemd`-based systems

- Script file: `/usr/lib/systemd/system/pmm-agent.service`
- Parameter: `StandardError`
- Default value: `file:/var/log/pmm-agent.log`

Example:

```
StandardError=file:/var/log/pmm-agent.log
```

`initd`-based systems

- Script file: `/etc/init.d/pmm-agent`
- Parameter: `pmm_log`
- Default value: `/var/log/pmm-agent.log`

Example:

```
pmm_log="/var/log/pmm-agent.log"
```

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-12-12

6.10 Query Analytics under the hood

6.10.1 Components

QAN consists of three main components:

Percona LLC and/or its affiliates ©2024

1. Filters Panel
2. Overview Panel
3. Details Panel

6.10.2 How QAN collects data?

- QAN collects data every minute, on the minute (e.g. 8:15:00 and next at 8:16:00, etc.).
- The collected data is represented by “buckets.”

6.10.3 Understanding buckets

- A bucket contains all the data captured during a one-minute interval.
- Once a bucket is created, it is sent to the PMM Server, where it is parsed and saved in the ClickHouse database. All QAN-related data is stored in ClickHouse, which can be either part of the PMM Server or an external instance.
- Queries in buckets are aggregated by query ID, which means that all queries with the same query ID are represented by a single row in the Overview panel.
- Query IDs are calculated differently depending on the database technology and query source.

MySQL

- For MySQL, the query ID is based on the DIGEST value (for Performance Schema) or the MD5 checksum of the fingerprint (for slow log).
- DIGEST (Performance Schema) and fingerprint (Slow Log) are both methods of representing a query without its sensitive data. In the case below, both queries will have the **same query ID** because the DIGEST value is generated from the query text after removing the sensitive data.

```
INSERT INTO people VALUES ('Joe', 'Doe');
INSERT INTO people VALUES ('John', 'Smith');
```

PERFSCHEMA

- The query ID is based on the DIGEST (fingerprint) value from the `events_statements_summary_by_digest` table in the mysql database.
- The DIGEST value for the same query may differ across MySQL versions.
- The DIGEST is generated from the query text without sensitive data (DIGEST_TEXT).
- In MySQL 8.0 and higher, you can use the STATEMENT_DIGEST(“your query”) function to obtain the DIGEST (query ID) for a given query. See the MySQL official documentation for more details.

SLOWLOG

- The query ID is the MD5 checksum of the query fingerprint.

6.10.4 Sources for data

- MySQL Performance Schema: The `events_statements_summary_by_digest` and `events_statements_history` tables in the `mysql` database.
- MySQL Slow Log: The slow log file specified during MySQL configuration.
- PostgreSQL pg_stat_statements (PGSS): The `pg_stat_statements` view in the required database.
- PostgreSQL pg_stat_monitor (PGSM): The `pg_stat_monitor` view in the required database.

Percona LLC and/or its affiliates ©2024

6.10.5 Examples

Example 1: MySQL with Performance Schema or Slow Log

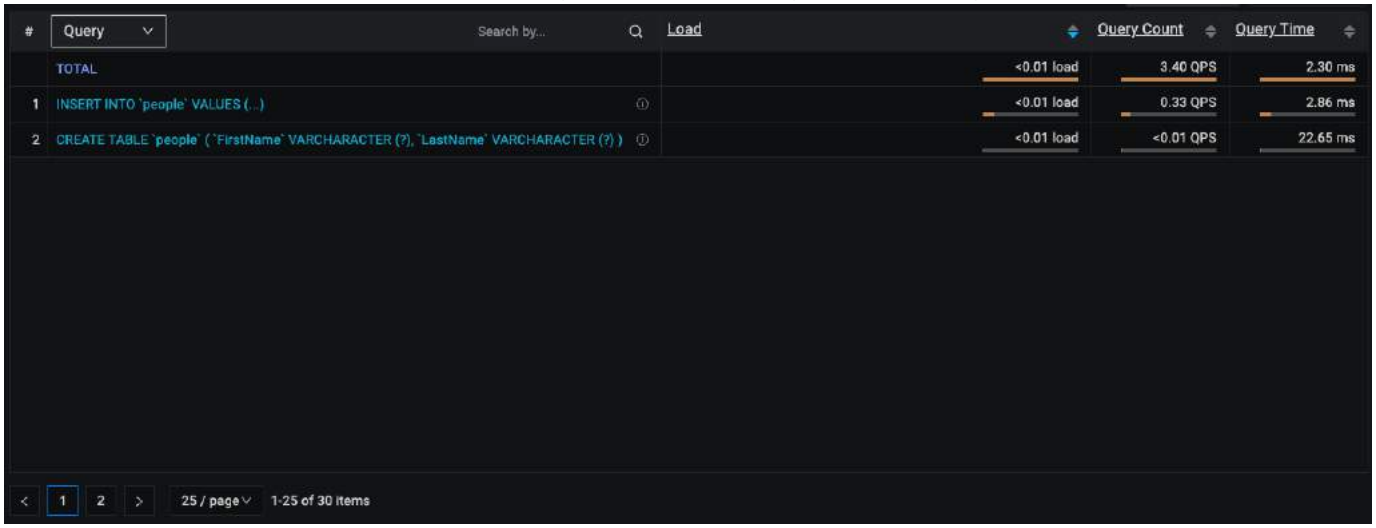
Timeline
8:05:05: You start the pmm-agent.

8:05:20: You execute the following queries:

```
CREATE TABLE people (FirstName varchar(255), LastName varchar(255));
INSERT INTO people VALUES ('Joe', 'Doe');
INSERT INTO people VALUES ('John', 'Smith');
```

8:05:25: The queries finish executing.

8:06:00: The buckets are collected and sent to the PMM Server. **8:06:10:** Head back to QAN. Here, you should see two rows in the QAN Overview Panel (depending on the filter settings and time range):



Let’s answer some questions about the image above.
Question: Why does the query look slightly different in the Overview list?
Answer: The query is the same, but sensitive data (e.g., ‘Joe’, ‘John’) is replaced by placeholders like “?”, “?+”, or “…” in the Overview list.

Question: We triggered two INSERT queries, but there is only one row in the list overview. Why?
Answer: For the executed queries below, query ID will be the same which is why they are aggregated into a single row in the Overview list. However, in the Details panel, you should still see the actual count of how many times the query was executed. As you can see in the image below, the count for the INSERT query is 2 in the Details panel, which matches the number of times we executed the INSERT query.

```
INSERT INTO people VALUES ('Joe', 'Doe');
INSERT INTO people VALUES ('John', 'Smith');
```

Percona LLC and/or its affiliates ©2024

Example 2: MySQL with Performance Schema or Slow Log, Query Execution Split Across Two Buckets

Timeline
8:05:55: You start pmm-agent.

8:05:59: You execute the following queries:

```
CREATE TABLE people (FirstName varchar(255), LastName varchar(255));
INSERT INTO people VALUES ('Joe', 'Doe');
INSERT INTO people VALUES ('John', 'Smith');
```

8:06:00: The first bucket is collected and sent to the PMM Server. The queries executed so far are included in this bucket and can be found in QAN for the time range 8:05:00-8:06:00.

8:06:01: The queries finish executing.

8:07:00: The second bucket is collected and sent to the PMM Server. The remaining executed queries are included in this bucket and are now visible in QAN.

8:07:10: You should see two rows in the QAN Overview Panel (depending on the filter settings and time range): Now let's take a look at results with different time range filters.

1. QAN with default time range filter (Last 12 hours) - two rows in overview (one for CREATE query, another for INSERT). In the Details panel, the count for the INSERT query is 2.
2. Range 8:05:00 - 8:06:00 - one row in the Overview list for the CREATE query, because none of the INSERT queries were executed by that time
3. Range 8:06:00 - 8:07:00 - One row in the Overview panel for the INSERT query because the CREATE query was sent in the previous bucket. In the Details Panel, the count for the INSERT query is 2 since both queries were executed in this range. ß Remember that it may take up to 2 minutes for all the data to be visible in QAN after the queries have finished executing.

In the future, we will include examples covering more technologies (e.g., PostgreSQL, MongoDB) in this topic.

6.10.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-06-03

6.11 API

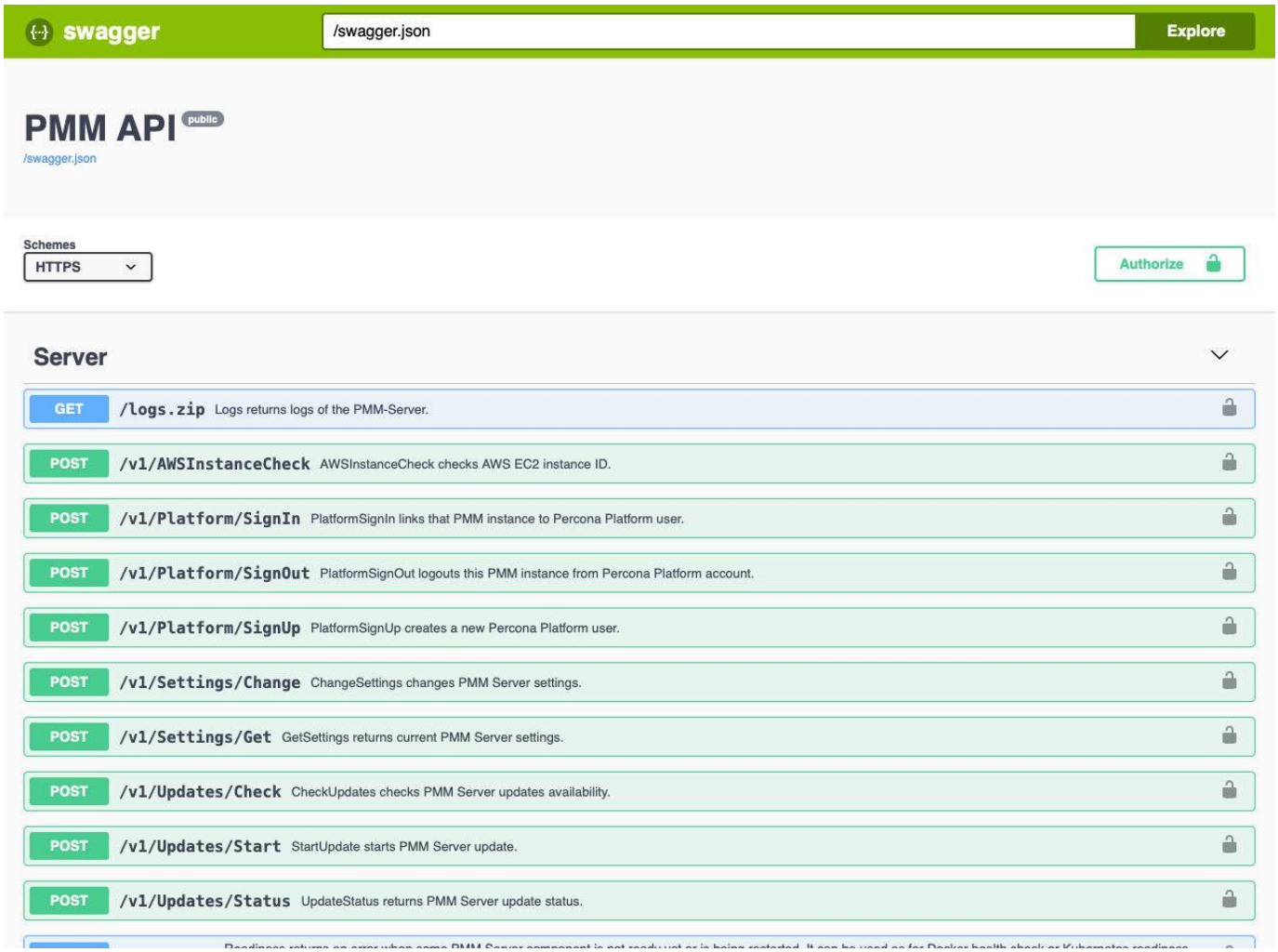
6.11.1 Interactive API documentation

To integrate your applications or CI/CD, you can use our online interactive documentation.

6.11.2 Swagger API documentation

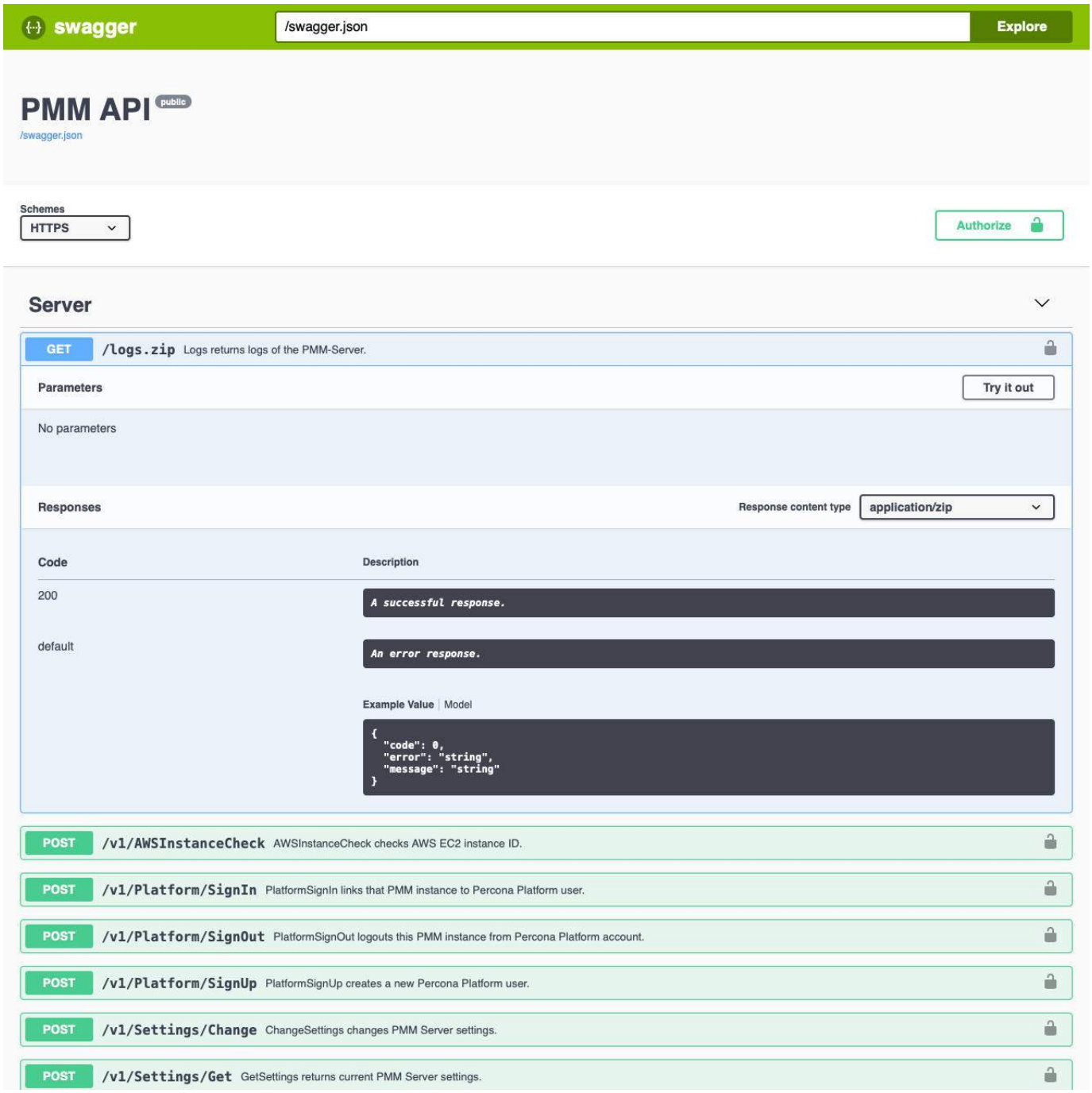
PMM Server lets you visually interact with API resources representing all objects within PMM. You can browse the API using the Swagger UI, accessible at the `/swagger` endpoint URL:

Percona LLC and/or its affiliates ©2024



Clicking on an object lets you examine objects and execute requests on them:

Percona LLC and/or its affiliates ©2024



The objects visible are nodes, services, and agents:

- A **Node** can be a bare metal server, a virtual machine, a Docker container, or a more specific type such as an Amazon RDS Node. A node runs zero or more Services and Agents, and has zero or more Agents providing insights for it.
- A **Service** is generally a database running on the Node: Amazon Aurora MySQL, MySQL, MongoDB, PostgreSQL, etc. It runs on zero (Amazon Aurora Serverless), single (MySQL), or several (Percona XtraDB Cluster) Nodes. It also has zero or more Agents providing insights for it.
- An **Agent** is an executable binary, that runs on the Node. It is not useful by itself, but instead provides insights (metrics, query performance data, etc.) about Nodes and/or Services. An agent always runs on the single Node (except External Exporters), and provides insights for zero or more Services and Nodes.

Nodes, Services, and Agents have **Types** which define their specific properties, and their specific logic.

Percona LLC and/or its affiliates ©2024

Nodes and Services are external by nature – we do not manage them (create, destroy), but merely maintain their inventory (add to inventory, remove from inventory) in `pmm-managed`. Most Agents are started and stopped by `pmm-agent`. One exception is the External Exporter Type which is started externally.

API Keys and authentication

API keys are used to control access to the PMM server components and resources. With an API key, you are authenticated to the PMM server, have access to PMM server components and resources, and perform various actions on them. You can use API keys as a replacement for basic authentication.

GENERATE API KEYS

PMM uses the Grafana API keys for authentication. Following are the steps to generate the API keys:

1. Login to PMM.
2. From the side menu, click *Configuration → API keys*.
3. On the Configuration page, click *Add API Key*.
4. *Add API key* dialog box opens.
5. Enter the following to generate an API key:
    - key name (you can give any desired name)
    - Select the Role from the drop-down
    - Enter a value in the Time to live text box (hover on the tooltip for more information).
    - Click Add.
6. *API Key Created* window displays your newly created key. Make sure to copy your key and keep it secure.

AUTHENTICATE

You can authenticate your request using the HTTPS header.

> **Important**
>
> Use the `-k` or `--insecure` parameter to force cURL to ignore invalid and self-signed SSL certificate errors. The option will skip the SSL verification process, and you can bypass any SSL errors while still having SSL-encrypted communication. However, using the `--insecure` parameter is not recommended. Although the data transfer is encrypted, it is not entirely secure. For enhanced security of your PMM installation, you need valid SSL certificates. For information on validating SSL certificates, refer to: SSL certificates.

```
curl -H "Authorization: Bearer <api_key>" https://127.0.0.1/v1/version
```

USE AN API KEY IN BASIC AUTH

You can use the API key for basic authentication in a REST API call in the following format. Replace `API_KEY` with your API key.

Example

```
curl -X GET https://api_key:API_KEY@127.0.0.1/v1/version
```

6.11.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2023-08-23

6.12 VictoriaMetrics

VictoriaMetrics is a third-party monitoring solution and time-series database that replaced Prometheus in PMM 2.12.0.

6.12.1 Push/Pull modes

VictoriaMetrics metrics data can be both 'pushed' to the server and 'pulled' by the server. When setting up services, you can decide which mode to use.

> The 'push' mode is now default for newly-added services. (In PMM 2.12.0 the default mode was 'pull'.)

The mode (push/pull) is controlled by the `--metrics-mode` flag for the `pmm-admin config` and `pmm-admin add` commands.

If you need to change the metrics mode for an existing Service, you must remove it and re-add it with the same name and the required flags. (You cannot update a service.)

6.12.2 Remapped targets for direct Prometheus paths

Direct Prometheus paths return structured information directly from Prometheus, bypassing the PMM application.

They are accessed by requesting a URL of the form `<PMM SERVER URL>/prometheus/<PATH>`.

As a result of the move to VictoriaMetrics some direct Prometheus paths are no longer available.

Prometheus path	VictoriaMetrics equivalent
`/prometheus/alerts`	No change.
`/prometheus/config`	No equivalent, but there is some information at `/prometheus/targets`.
`/prometheus/flags`	The `flag` metrics at `/prometheus/metrics`.
`/prometheus/graph`	`/graph/explore` (Grafana) or `graph/d/prometheus-advanced/advanced-data-exploration` (PMM dashboard).
`/prometheus/rules`	No change.
`/prometheus/service-discovery`	No equivalent.
`/prometheus/status`	Some information at `/prometheus/metrics`. High cardinality metrics information at `/prometheus/api/v1/status/tsdb`.
`/prometheus/targets`	`/victoriametrics/targets`

6.12.3 Environment variables

PMM predefines certain flags that allow users to set all other VictoriaMetrics parameters as environment variables:

The environment variable must be prepended with `VM_`.

Example

Percona LLC and/or its affiliates ©2024

To set downsampling, use the `downsampling.period` parameter as follows:

```
-e VM_downsampling_period=20d:10m,120d:2h
```

This instructs VictoriaMetrics to deduplicate samples older than 20 days with 10 minute intervals and samples older than 120 days with two hour intervals.

6.12.4 Using VictoriaMetrics external database instance

> **Important/Caution**
>
> This feature is still in Technical Preview and is subject to change. We recommend that early adopters use this feature for evaluation purposes only.

Starting with PMM 2.40.0, you can now use an external VictoriaMetrics database for monitoring in PMM.

The environment variable `PMM_VM_URL` has been added, which should point to the external VictoriaMetrics database and should have the following format:

```
http(s)://hostname:port/path.
```

If the external VictoriaMetrics database requires basic authentication, the following environment variables should be used:

```
VMAGENT_remoteWrite_basicAuth_username={username}
VMAGENT_remoteWrite_basicAuth_password={password}
```

If other authentication methods are used on the VictoriaMetrics side, users can use any of the `vmagent` environment variables by prepending `VMAGENT_ prefix`.

When external VictoriaMetrics is configured, internal VictoriaMetrics stops. In this case, VM Agent on PMM Server pulls metrics from agents configured in the `pull metrics mode` and from remote nodes. Data is then pushed to external VictoriaMetrics.

> **Note**
>
> VM Agents run by PMM Clients push data directly to external VictoriaMetrics.
>
> Ensure that they can connect to external VictoriaMetrics.

6.12.5 Troubleshooting

To troubleshoot issues, see the VictoriaMetrics troubleshooting documentation.

You can also contact the VictoriaMetrics team via:

- Google Groups
- Slack
- Reddit
- Telegram

Percona LLC and/or its affiliates ©2024

6.12.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-10-05

6.13 ClickHouse

You can use an external ClickHouse database instance outside the PMM Server container running on other hosts.

6.13.1 Environment variables

PMM predefines certain flags that allow you to use ClickHouse parameters as environment variables:

> **Warning**
>
> The `PERCONA_TEST_*` environment variables are experimental and subject to change. It is recommended that you use these variables for testing purposes only and not on production.

To use ClickHouse as an external database instance, use the following environment variables:

`PERCONA_TEST_PMM_CLICKHOUSE_ADDR` -> hostname:port

Name of the host and port of the external ClickHouse database instance.

Optional environment variables

`PERCONA_TEST_PMM_CLICKHOUSE_DATABASE` -> database name

Database name of the external ClickHouse database instance.

`PERCONA_TEST_PMM_CLICKHOUSE_POOL_SIZE` -> pool size

The maximum number of threads in the current connection thread pool. This value cannot be bigger than max_thread_pool_size.

`PERCONA_TEST_PMM_CLICKHOUSE_BLOCK_SIZE` -> max block size

The number of rows to load from tables in one block for this connection.

Example

To use ClickHouse as an external database instance, start the PMM docker with the specified variables for external ClickHouse:

```
-e PERCONA_TEST_PMM_CLICKHOUSE_ADDR=$ADDRESS:$PORT
-e PERCONA_TEST_PMM_CLICKHOUSE_DATABASE=$DB
-e PERCONA_TEST_PMM_CLICKHOUSE_POOL_SIZE=1
-e PERCONA_TEST_PMM_CLICKHOUSE_BLOCK_SIZE=65000
```


Percona LLC and/or its affiliates ©2024

6.13.2 Troubleshooting

To troubleshoot issues, see the ClickHouse troubleshooting documentation.

6.13.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-07-19

6.14 PostgreSQL

You can use an external PostgreSQL database instance outside the PMM Server container running on the same or other hosts.

6.14.1 Environment variables

PMM predefines certain flags that allow you to use PostgreSQL parameters as environment variables:

> **Warning**
>
> The `PERCONA_TEST_*` environment variables are experimental and subject to change. It is recommended that you use these variables for testing purposes only and not on production. The minimum supported PostgreSQL server version is 14.

To use PostgreSQL as an external database instance, use the following environment variables:

Environment variable	Flag	Description
PERCONA_TEST_POSTGRES_ADDR	postgres-addr	Hostname and port for external PostgreSQL database.
PERCONA_TEST_POSTGRES_DBNAME	postgres-name	Database name for external or internal PostgreSQL database.
PERCONA_TEST_POSTGRES_USERNAME	postgres-username	PostgreSQL user name to connect as.
PERCONA_TEST_POSTGRES_DBPASSWORD	postgres-password	Password to be used for database authentication.
PERCONA_TEST_POSTGRES_SSL_MODE	postgres-ssl-mode	This option determines whether or with what priority a secure SSL TCP/IP connection will be negotiated with the database. Currently supported: `disable`, `require`, `verify-ca`, `verify-full`.
PERCONA_TEST_POSTGRES_SSL_CA_PATH	postgres-ssl-ca-path	This parameter specifies the name of a file containing SSL certificate authority (CA) certificate(s).
PERCONA_TEST_POSTGRES_SSL_KEY_PATH	postgres-ssl-key-path	This parameter specifies the location for the secret key used for the client certificate.

Percona LLC and/or its affiliates ©2024

Environment variable	Flag	Description
PERCONA_TEST_POSTGRES_SSL_CERT_PATH	postgres-ssl-cert-path	This parameter specifies the file name of the client SSL certificate.
PERCONA_TEST_PMM_DISABLE_BUILTIN_POSTGRES		Environment variable to disable built-in PMM server database. Note that Grafana depends on built-in PostgreSQL. And if the value of this variable is “true”, then it is necessary to pass all the parameters associated with Grafana to use external PostgreSQL.

By default, communication between the PMM server and the database is not encrypted. To secure a connection, follow PostgeSQL SSL instructions and provide `POSTGRES_SSL_*` variables.

To use grafana with external PostgreSQL add `GF_DATABASE_*` environment variables accordingly.

Example

To use PostgreSQL as an external database:

1. Generate all necessary SSL certificates.
2. Deploy PMM Server with certificates under read-only permissions and Grafana user and Grafana group.

```
/pmm-server-certificates# la -la
drwxr-xr-x 1 root    root    4096 Apr  5 12:43 .
drwxr-xr-x 1 root    root    4096 Apr  5 12:43 ..
-rw------- 1 grafana grafana 1391 Apr  5 12:38 certificate_authority.crt
-rw------- 1 grafana grafana 1257 Apr  5 12:38 pmm_server.crt
-rw------- 1 grafana grafana 1708 Apr  5 12:38 pmm_server.key
```

3. Attach `pg_hba.conf` and certificates to the PostgreSQL image.

```
/external-postgres-configuration# cat pg_hba.conf
local     all         all                                 trust
hostnossl all         example_user all                    reject
hostssl   all         example_user all                    cert
```

```
/external-postgres-certificates# ls -la
drwxr-xr-x 1 root     root     4096 Apr  5 12:38 .
drwxr-xr-x 1 root     root     4096 Apr  5 12:43 ..
-rw------- 1 postgres postgres 1391 Apr  5 12:38 certificate_authority.crt
-rw------- 1 postgres postgres 1407 Apr  5 12:38 external_postgres.crt
-rw------- 1 postgres postgres 1708 Apr  5 12:38 external_postgres.key
```

4. Create `user` and `database` for pmm-server to use. Set appropriate rights and access.
5. Install `pg_stat_statements` in PostgreSQL in order to have all metrics according to this handy document.
6. Run PostgreSQL server.

```
docker run
--name external-postgres
-e POSTGRES_PASSWORD=secret
<image_id>
postgres
-c shared_preload_libraries=pg_stat_statements
-c pg_stat_statements.max=10000
-c pg_stat_statements.track=all
-c pg_stat_statements.save=off
-c ssl=on
-c ssl_ca_file=$CA_PATH
-c ssl_key_file=$KEY_PATH
-c ssl_cert_file=$CERT_PATH
-c hba_file=$HBA_PATH
```

7. Run PMM server.

Percona LLC and/or its affiliates ©2024

```
docker run
--name pmm-server
-e PERCONA_TEST_POSTGRES_ADDR=$ADDRESS:$PORT
-e PERCONA_TEST_POSTGRES_DBNAME=$DBNAME
-e PERCONA_TEST_POSTGRES_USERNAME=$USER
-e PERCONA_TEST_POSTGRES_DBPASSWORD=$PASSWORD
-e PERCONA_TEST_POSTGRES_SSL_MODE=$SSL_MODE
-e PERCONA_TEST_POSTGRES_SSL_CA_PATH=$CA_PATH
-e PERCONA_TEST_POSTGRES_SSL_KEY_PATH=$KEY_PATH
-e PERCONA_TEST_POSTGRES_SSL_CERT_PATH=$CERT_PATH
-e PERCONA_TEST_PMM_DISABLE_BUILTIN_POSTGRES=true
-e GF_DATABASE_URL=$GF_DATABASE_URL
-e GF_DATABASE_SSL_MODE=$GF_SSL_MODE
-e GF_DATABASE_CA_CERT_PATH=$GF_CA_PATH
-e GF_DATABASE_CLIENT_KEY_PATH=$GF_KEY_PATH
-e GF_DATABASE_CLIENT_CERT_PATH=$GF_CERT_PATH
percona/pmm-server:2
```

6.14.2 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-05-15

6.15 Glossary

6.15.1 Annotation

A way of showing a mark on dashboards signifying an important point in time.

6.15.2 Dimension

In the Query Analytics dashboard, to help focus on the possible source of performance issues, you can group queries by *dimension*, one of: Query, Service Name, Database, Schema, User Name, Client Host

6.15.3 EBS

Amazon’s Elastic Block Store.

6.15.4 Fingerprint

A *normalized statement digest*—a query string with values removed that acts as a template or typical example for a query.

6.15.5 IAM

Identity and Access Management (for Amazon AWS).

6.15.6 MM

Metrics Monitor.

Percona LLC and/or its affiliates ©2024

6.15.7 NUMA

Non-Uniform Memory Access.

6.15.8 PEM

Privacy Enhanced Mail.

6.15.9 QPS

Queries Per Second. A measure of the rate of queries being monitored.

6.15.10 Query Analytics

Component of PMM Server that enables you to analyze MySQL query performance over periods of time.

6.15.11 Advisors

Automated checks that you can run against connected databases to identify any potential security threats, configuration problems, performance concerns, policy non-compliance issues etc.

6.15.12 Technical Preview

Releases intended for public preview and feedback, but with no support or service level agreement (SLA). Should not be used on production or business-critical systems. May contain breaking changes to UI, API, CLI. (Read more.)

6.15.13 VG

Volume Group.

6.15.14 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-08

Percona LLC and/or its affiliates ©2024

7. DBaaS

7.1 Introduction to Database as a service (DBaaS)

Do not use for mission-critical workloads

- DBaaS feature is deprecated. We encourage you to use Percona Everest instead. Check our Migration guide.
- DBaaS feature is available for PMM Admin users.

Database as a service (DBaaS) feature of Percona Monitoring and Management (PMM) is an open source solution to run MySQL and MongoDB clusters on Kubernetes.

It allows you to utilize the benefits of Kubernetes and Percona's operators to run and manage database clusters.

7.1.1 Start here

- Architecture and how DBaaS works
- Setting up
- Getting started with DBaaS
- Registering and managing Kubernetes clusters
- Creating and managing database clusters
- Creating or updating a database cluster from a DBaaS template

7.1.2 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-12-12

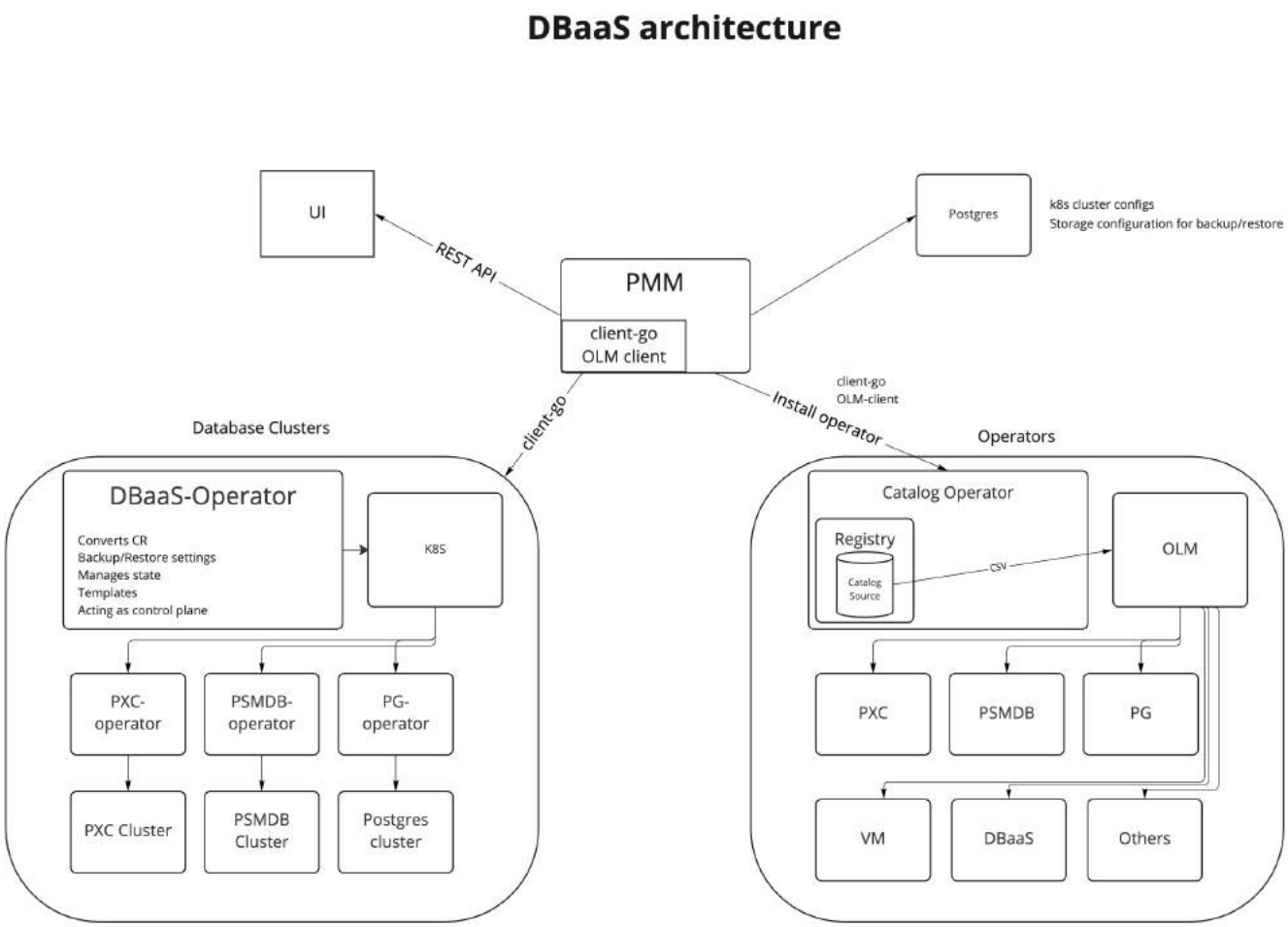
7.2 DBaaS architecture

Do not use for mission-critical workloads

DBaaS feature is deprecated. We encourage you to use Percona Everest instead. Check our Migration guide.

DBaaS is built on top of PMM and Kubernetes and the high-level architecture is shown below:

Percona LLC and/or its affiliates ©2024



In DBaaS, the role of PMM is as follows:

1. Expose Public REST API
2. Provision Kubernetes cluster and install the following operators:
    a. OLM (Operator Lifecycle Manager)
    b. Percona Operator for MongoDB
    c. Percona Operator for MySQL
    d. DBaaS operator

7.2.1 Operator Lifecycle Manager (OLM)

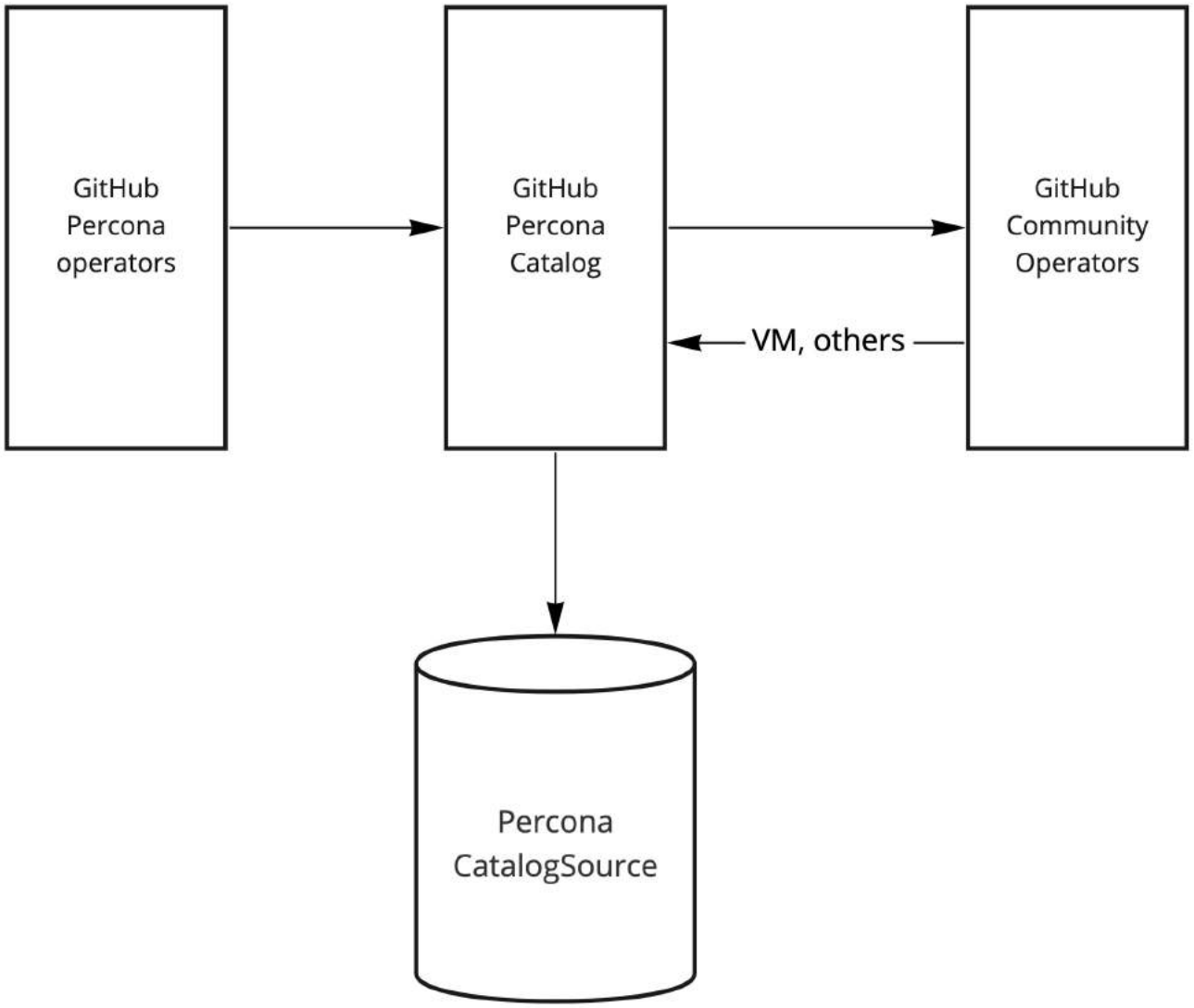
DBaaS uses OLM to install and update operators. PMM installs OLM and Operator Catalog during the registration of the Kubernetes cluster.

An Operator catalog is a repository of metadata that Operator Lifecycle Manager (OLM) can query to discover and install Operators and their dependencies on a cluster. OLM always installs Operators from the latest version of a catalog and if a new version of an operator becomes available, OLM can upgrade it once the user has accepted the upgrade. DBaaS uses its own catalog for OLM that has the following operators:

1. DBaaS operator
2. PXC operator
3. PSMDB operator

Percona LLC and/or its affiliates ©2024

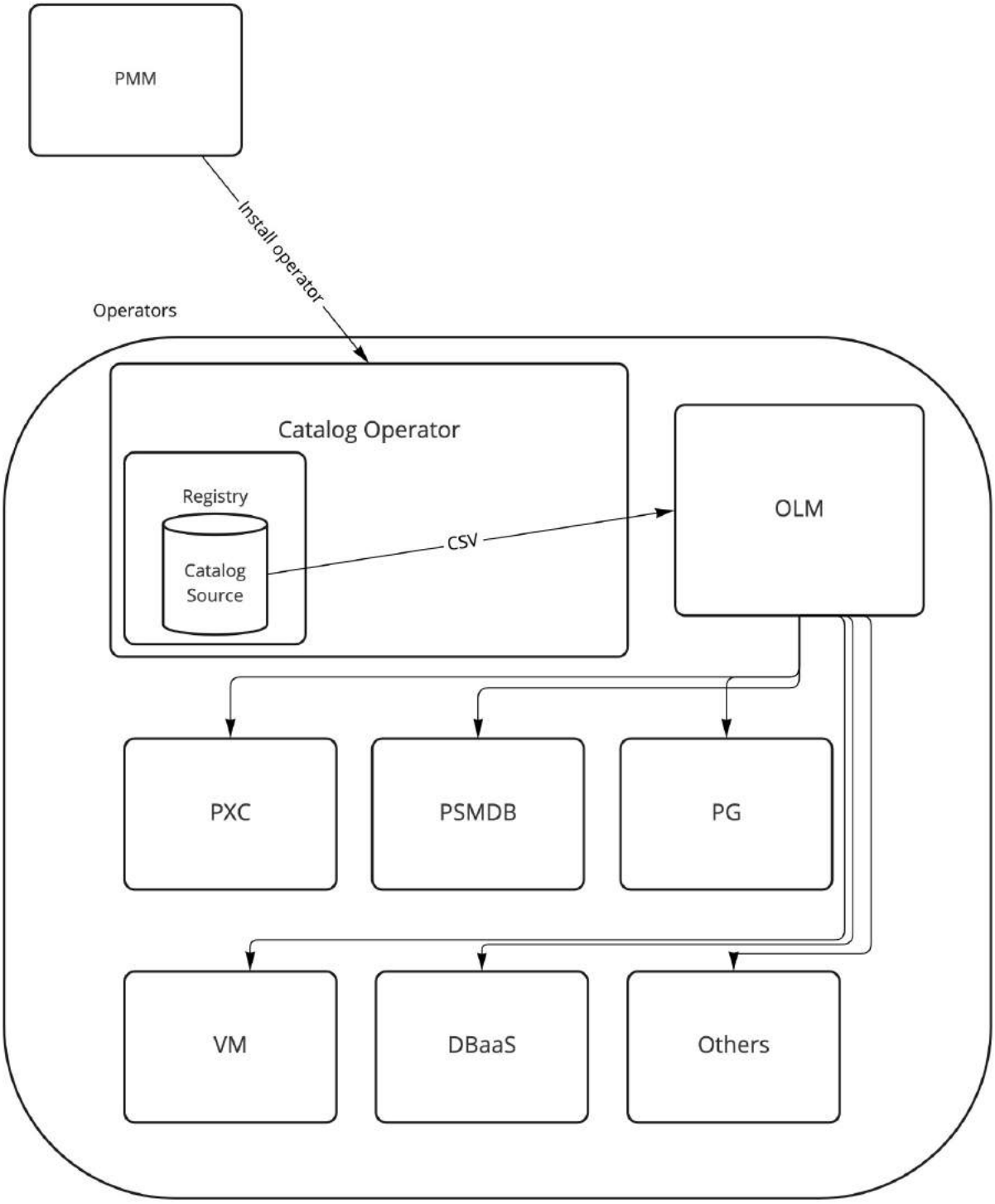
4. Victoria Metrics operator



Percona Catalog is an OLM catalog that stores ClusterServiceVersions and CustomResourceDefinitions for creation in a cluster, and stores metadata about packages and channels. It’s a source of truth for the available versions of operators ready to use in DBaaS

The installation of operators looks the following way

Percona LLC and/or its affiliates ©2024



7.2.2 DBaaS operator

DBaaS operator is responsible for creating and managing databases following operator pattern and depends on underlying operators for running database clusters. It provides a simplified API to manage database clusters via `kubectl`.

Percona LLC and/or its affiliates ©2024

7.2.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-12-12

7.3 Managing Database clusters

7.3.1 Database cluster Templates

> **Do not use for mission-critical workloads**
>
> DBaaS feature is deprecated. We encourage you to use Percona Everest instead. Check our Migration guide.

Database clusters can be created from templates using PMM. Database cluster Template allows operators to customize Database Clusters based on their requirements, environments, or infrastructure.

Examples

- **Data clusters**: Different load patterns, such as simple reads, heavy writes, 50%/50% read/write, and the number of connections.
- **Infrastructure** - Different parameters and tuning for the resulting cluster: such as network configuration (load balancing, exposure), storage classes/types, etc.
- **Environment**: Combination of database cluster and infrastructure that will affect the database cluster configuration.

7.3.2 Customize PXC DB configuration

The following **example** shows how DBaaS users can create PXC DB clusters from a template that sets a custom MySQL configuration. The default MySQL configuration is the following:

```
[mysqld]
wsrep_provider_options="gcache.size=600M"
```

> **Note**
>
> PXC DB configuration can be customized based on your needs. This can be accomplished by creating a template and changing that field.

Create Custom Resource Definition (CRD) template

To create a template, do the following:

1. Identify the field(s) of interest by reading the PXC operator documentation and the PXC CRD. In this case, you have to change the `spec.pxc.configuration` field.

Percona LLC and/or its affiliates ©2024

2. Create a template CRD `pxctpl-crd-pxc-configuration.yaml` with just those small subset of fields.

```
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
annotations:
    controller-gen.kubebuilder.io/version: v0.8.0
creationTimestamp: null
name: pxctemplatepxcconfiguration.dbaas.percona.com
labels:
    dbaas.percona.com/template: "yes"
    dbaas.percona.com/engine: "pxc"
spec:
group: dbaas.percona.com
names:
    kind: PXCTemplatePXCConfiguration
    listKind: PXCTemplatePXCConfigurationList
    plural: pxctemplatepxcconfiguration
    singular: pxctemplatepxcconfiguration
scope: Namespaced
versions:
- name: v1
    schema:
    openAPIV3Schema:
        properties:
        apiVersion:
            type: string
        kind:
            type: string
        metadata:
            type: object
        spec:
            properties:
            pxc:
                properties:
                configuration:
                    type: string
                type: object
            type: object
        status:
            type: object
        type: object
    served: true
    storage: true
```

3. Run the following command.

```
kubectl apply -f pxctpl-crd-upgrade-options.yaml
```

For more information, see DatabaseCluster templates.

ADD READ PERMISSIONS FOR DBAAS-OPERATOR

In order for the dbaas-operator to apply the template it needs access to the template CRs:

```
DBAAS_OPERATOR_MANAGER_ROLE=$(kubectl get clusterroles | grep dbaas-operator | grep -v metrics | grep -v proxy | cut -f 1 -d ' '); kubectl get
clusterroles/"$DBAAS_OPERATOR_MANAGER_ROLE" -o yaml > dbaas-operator-manager-role.yaml
cat <<EOF >>dbaas-operator-manager-role.yaml
- apiGroups:
  - dbaas.percona.com
  resources:
  - pxctemplatepxcconfiguration
  verbs:
  - get
  - list
EOF
```

Run the following command:

```
kubectl apply -f dbaas-operator-manager-role.yaml
```


Percona LLC and/or its affiliates ©2024

CREATE PXCTEMPLATEPXCCONFIGURATION TEMPLATE CR

1. Create a corresponding CR `pxctpl-pxc-config-max-connection-789.yaml` with the required values.

```
apiVersion: dbaas.percona.com/v1
kind: PXCTemplatePXCConfiguration
metadata:
name: pxc-config-max-connections-789
labels:
    dbaas.percona.com/template: "yes"
    dbaas.percona.com/engine: "pxc"
spec:
pxc:
    configuration: |
    [mysqld]
    max_connections = 789
```

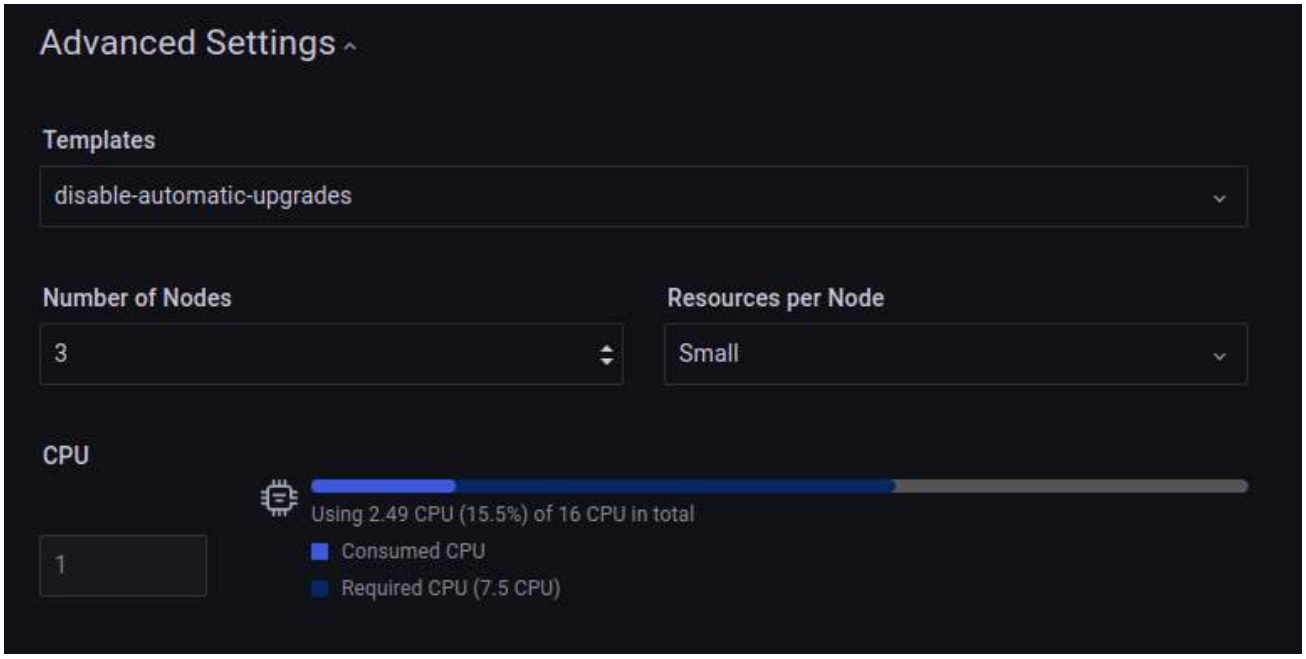
2. Run the following command:

```
kubectl apply -f pxctpl-pxc-config-max-connection-789.yaml
```

Create a DB cluster from template

To create a DB cluster from a template, do the following:

1. From the main menu navigate to *DBaaS → Create DB Cluster*.
2. On the *Advanced Settings* panel, select the template from the *Templates* drop-down.



3. Click `Create`.

Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2023-12-12

7.4 DBaaS backup and restore

> **Do not use for mission-critical workloads**
>
> DBaaS feature is deprecated. We encourage you to use Percona Everest instead. Check our Migration guide.

You can add a backup schedule while creating DB clusters in DBaaS. This feature is a fusion of backup management and DBaaS in PMM. Currently, DBaaS only supports scheduled backups, which can only be enabled when a database cluster is created.

To create a scheduled backup do the following:

Prerequisites

1. Enable Backup management:

   From the main menu, navigate to *Configuration → Settings → Advanced Settings → Backup Management* to enable *Backup management*.

2. Add a backup storage location:

   > **Note**
   >
   > Currently, only S3 is supported.

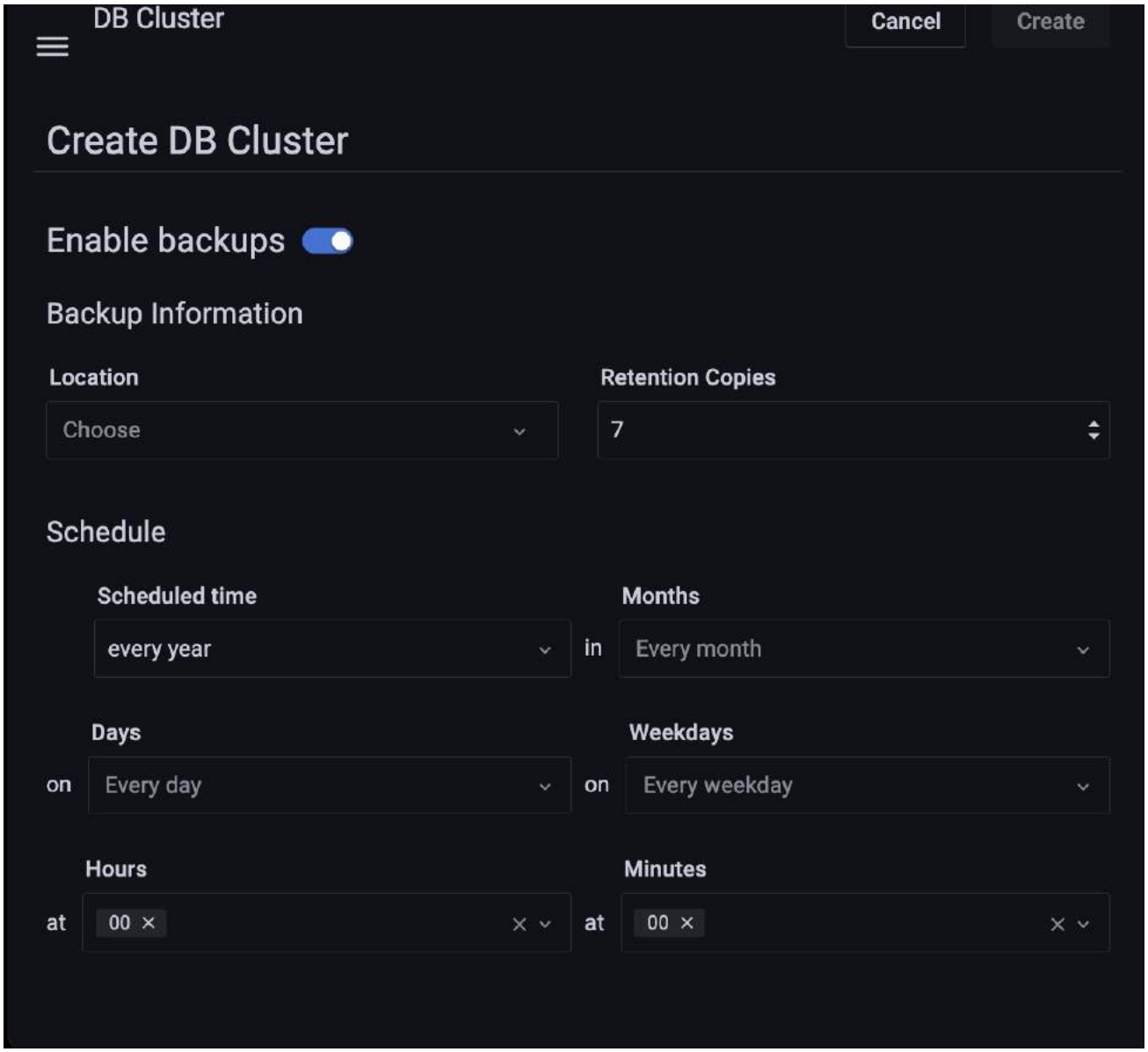
   From the main menu, navigate to *Backups → Storage Locations → Add storage location* and enter all the required information to add a backup storage location. For details, see Prepare a storage location.

7.4.1 Create a backup schedule

To create a backup schedule do the following:

1. From the main menu navigate to *DBaaS → DB Cluster*.
2. Click toggle in the *Enable backups* panel.

Percona LLC and/or its affiliates ©2024



3. In the *Backup Information* panel, enter the backup details.
4. Set the schedule for when scheduled backups should take place:

   From the drop-down select the *Scheduled time* when you want the backup to take place.
5. Click *Create*.

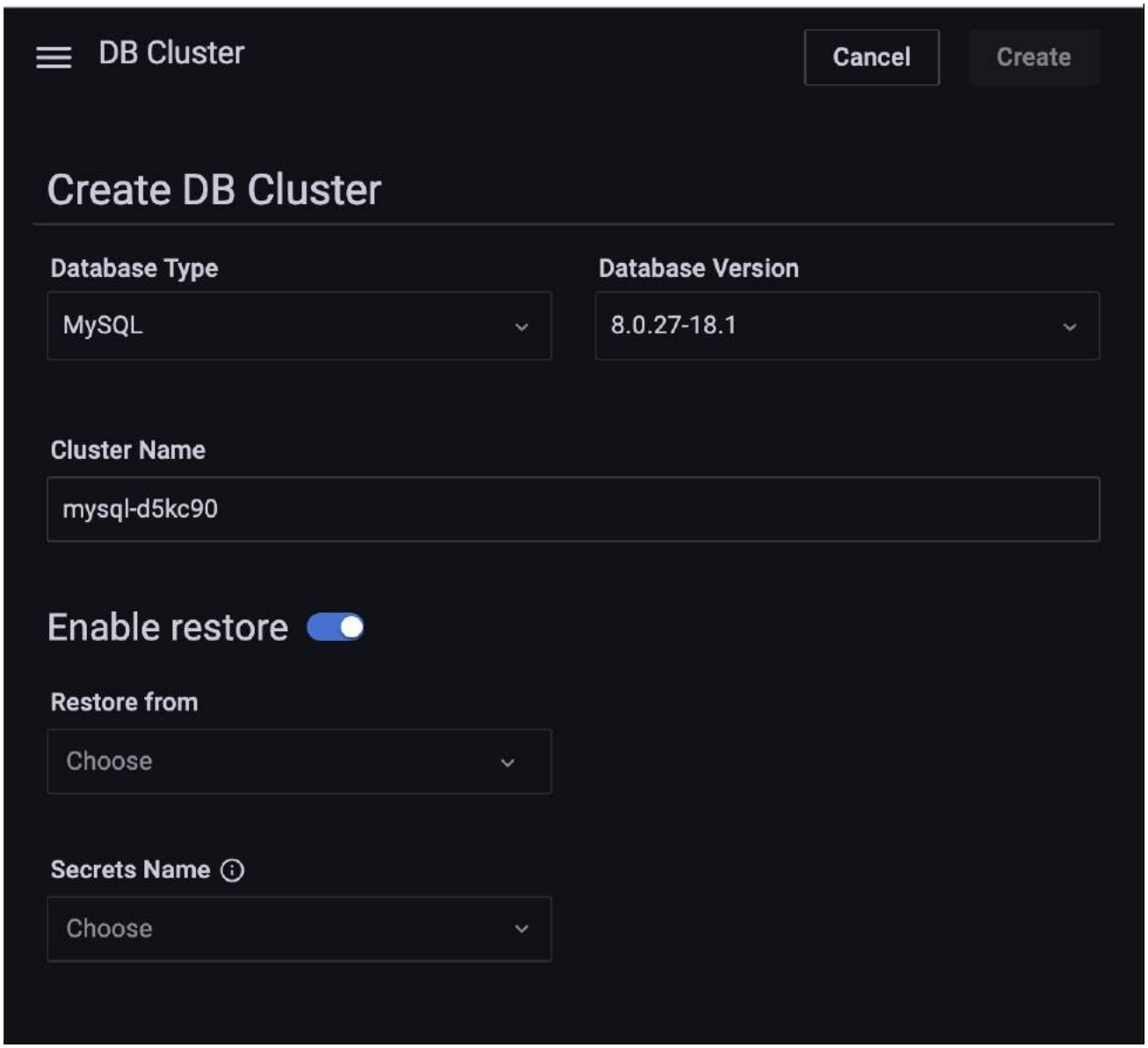
7.4.2 Restore backup

You can create a DBaaS cluster from a backup stored on S3. You can use backups from an existing cluster to spin up a new database cluster from this backup.

To restore backup, do the following:

1. From the main menu navigate to *DBaaS → DB Cluster*.
2. Click toggle in the *Enable restore* panel.

Percona LLC and/or its affiliates ©2024



3. Enter the information on the *Enable restore* panel.
4. Click *Create*.

7.4.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

Percona LLC and/or its affiliates ©2024

8. FAQ

8.1 How can I contact the developers?

- Community forum.
- PMM project in JIRA.

8.2 What are the minimum system requirements?

- Server:
  - Disk: 1 GB per monitored database (1 week data retention)
  - Memory: 2 GB per monitored database
  - CPU: Supports `SSE4.2`
- Client:
  - Disk: 100 MB

> **See also**
>
> - Setting up PMM Server
> - Setting up PMM Client

8.3 How can I upgrade from version 1?

There is no direct software upgrade path.

You must set up PMM 2 and connect your existing clients to it.

When all data is registered in PMM2 and expired in PMM1, decommission your PMM1 instance.

> **See also**
>
> - Upgrade from PMM1
> - Percona blog: Running PMM1 and PMM2 Clients on the Same Host

8.4 How to control data retention?

Go to *Configuration* → *Settings* → *Advanced Settings* → *Data retention* to adjust the value in days.

> **See also**
>
> Configure data retention

8.5 How are PMM Server logs rotated?

PMM Server embeds multiple components, like Victoria Metrics, Query Analytics, Grafana, `managed`, PostgreSQL, ClickHouse, etc. (components). All PMM Server component logs are rotated by `supervisord`. The components' log rotation settings are stored in `*.ini` files within the `/etc/supervisord.d` directory. Those settings define both the

Percona LLC and/or its affiliates ©2024

maximum size of a log file and the number of log files to keep. The log rotation takes place once the log file reaches its maximum size.

8.6 What privileges are required to monitor a MySQL instance?

```
SELECT, PROCESS, SUPER, REPLICATION CLIENT, RELOAD
```

See also

Setting Up/Client/MySQL.

8.7 Can I monitor multiple service instances?

Yes.

You can add multiple instances of MySQL or any other service to be monitored from the same PMM Client.

To do this, you provide a unique port and IP address, or a socket for each instance, and specify a unique name for each. (If a name is not provided, PMM uses the name of the PMM Client host.)

For example, to add MySQL monitoring for two local MySQL servers:

```
pmm-admin add mysql --username root --password root instance-01 127.0.0.1:3001
pmm-admin add mysql --username root --password root instance-02 127.0.0.1:3002
```

See also

`pmm-admin add mysql`

8.8 Can I rename instances?

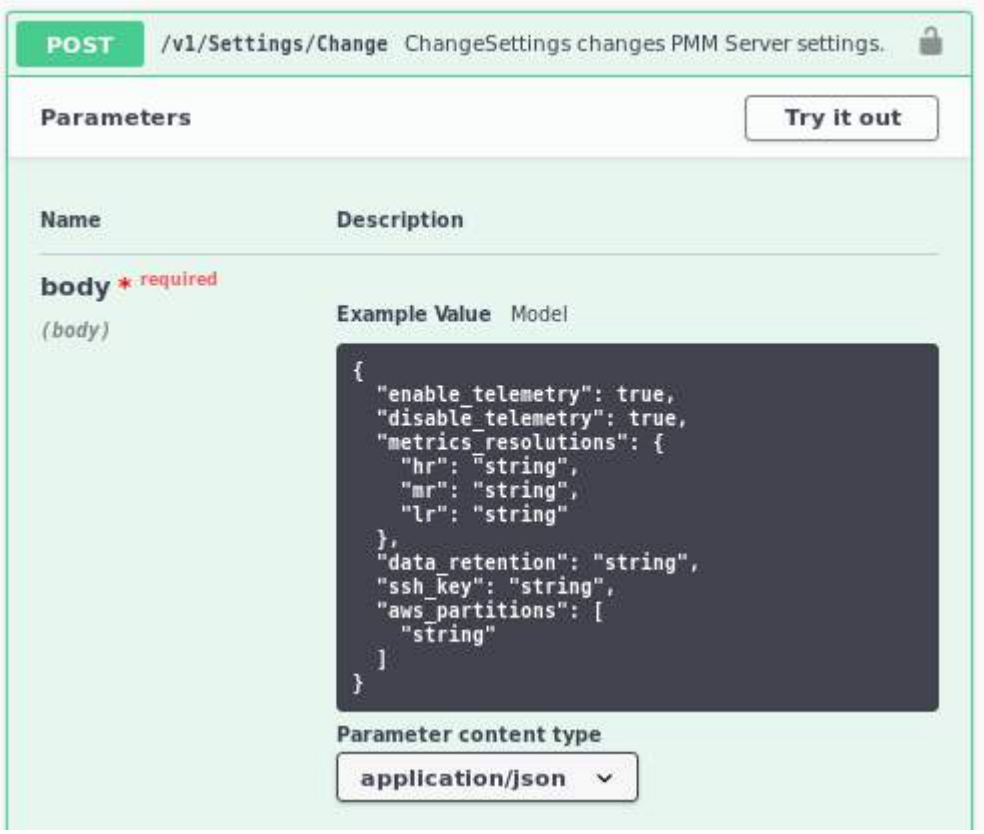
Yes, by removing and re-adding with a different name.

When you remove a monitoring service, previously collected data remains available in Grafana. However, the metrics are tied to the instance name. So if you add the same instance back with a different name, it will be considered a new instance with a new set of metrics. So if you are re-adding an instance and want to keep its previous data, add it with the same name.

8.9 Can I add an AWS RDS MySQL or Aurora MySQL instance from a non-default AWS partition?

By default, the RDS discovery works with the default `aws` partition. But you can switch to special regions, like the GovCloud one, with the alternative AWS partitions (e.g. `aws-us-gov`) adding them to the *Settings* via the PMM Server API.

Percona LLC and/or its affiliates ©2024



To specify other than the default value, or to use several, use the JSON Array syntax: `["aws", "aws-cn"]`.

8.10 What resolution is used for metrics?

The default values (in seconds):

Preset	Low	Medium	High
Rare	300	180	60
Standard	60	10	5
Frequent	30	5	1
Custom (defaults)	60	10	5

See also

Metrics resolution

8.11 How do I set up Alerting?

When a monitored service metric reaches a defined threshold, PMM Server can trigger alerts for it using embedded Grafana Alerting functionality.

For this, you must configure alerting rules that define conditions under which an alert should be triggered, and the contact points used to send the alert (e.g. email).

Percona templated alerts enable you to create alerts based on built-in or custom templates to simplify the alert setup process. Grafana managed alerts allows attaching rules to your dashboard panel and enables you to create more

Percona LLC and/or its affiliates ©2024

sophisticated alerting rules. In addition, it can be easier to manage installations with a large number of hosts. This additional flexibility comes at the expense of simplicity.

See also

Grafana Alerting

8.12 How do I use a custom Prometheus configuration file?

Normally, PMM Server fully manages the Prometheus configuration file.

However, some users may want to change the generated configuration to add additional scrape jobs, configure remote storage, etc.

From version 2.4.0, when `pmm-managed` starts the Prometheus file generation process, it tries to load the `/srv/prometheus/prometheus.base.yml` file first, to use it as a base for the `prometheus.yml` file.

The `prometheus.yml` file can be regenerated by restarting the PMM Server container, or by using the `SetSettings` API call with an empty body.

See also

- API
- Percona blog: Extending PMM's Prometheus Configuration

8.13 How to troubleshoot an Update?

See Troubleshoot update.

8.14 What are my login credentials when I try to connect to a Prometheus Exporter?

- User name: `pmm`
- Password: Agent ID

PMM protects an exporter's output from unauthorized access by adding an authorization layer. To access an exporter, you can use `pmm` as a user name and the Agent ID as a password. You can find the Agent ID corresponding to a given exporter by running `pmm-admin list`.

See also

`pmm-admin list`

8.15 How to provision PMM Server with non-default admin password?

Currently, there is no API available to change the `admin` password. If you're deploying through Docker, you can use the following code snippet to change the password after starting the Docker container:

```
PMM_PASSWORD="mypassword"
echo "Waiting for PMM to initialize to set password..."
until [ "`docker inspect -f {{.State.Health.Status}} pmm-server`" = "healthy" ]; do sleep 1; done
docker exec -t pmm-server bash -c  "grafana-cli --homepath /usr/share/grafana admin reset-admin-password $PMM_PASSWORD"
```


Percona LLC and/or its affiliates ©2024

(This example assumes your Docker container is named `pmm-server`.)

8.16 How to change the PMM password for a default admin user?

If you're deploying through Docker, you can change the default password for an admin user after starting the Docker container as follows:

- For PMM versions 2.27.0 and later:

```
docker exec -t pmm-server change-admin-password <new_password>
```

- For PMM versions prior to 2.27.0:

```
docker exec -t pmm-server bash -c 'grafana-cli --homepath /usr/share/grafana --configOverrides cfg:default.paths.data=/srv/grafana admin reset-admin-password
newpass'
```

8.17 How to use a non-default listen-port for pmm-admin?

If you configure the PMM agent to use a non-default listen-port, for pmm-admin to communicate with the agent, use the global flag `--pmm-agent-listen-port=LISTEN_PORT`.

```
--pmm-agent-listen-port=LISTEN_PORT
```

Example: To use the listen-port 8000

```
pmm-admin --pmm-agent-listen-port=8000 add postgresql --username=pmm-agent --password=pmm-agent-password --query-source=pgstatmonitor nameofpostgres
```

If you are using OVF/AMI, you can change the default password through SSH by using the following command:

```
change-admin-password <new_password>
```

8.18 How does PMM handle personal and confidential data?

Read our Privacy Policy to learn how PMM manages personal and confidential data. More technical details can be found in Data handling in PMM.

8.19 Why am I getting a "User already exists" error when logging back into PMM?

Following CVE fix 2023-3128 in the 2.38 release, PMM increases security by only allowing authentications based on the unique user ID provided by the identity provider.

If you are trying to log into PMM via a third-party authentication provider which doesn't support a unique ID field, PMM 2.38 and later will show this error on second and subsequent authentications.

Solution: we recommend logging into PMM using a Percona Account, as this is a highly secure authentication method. **Workaround**: if you need to log into PMM via a third-party authentication provider which doesn't support a unique ID field, you can use the following workaround to log into PMM:

- pass the `GF_AUTH_OAUTH_ALLOW_INSECURE_EMAIL_LOOKUP=1` environment variable to the PMM container **OR**
- set the `oauth_allow_insecure_email_lookup` config key in the auth section of the `grafana.ini` file. Keep in mind that any changes you make to this file are lost when upgrading PMM, so make sure to manually update this file after each upgrade.

Percona LLC and/or its affiliates ©2024

Important

We do not recommend using the above workaround for an extended period. Instead, ensure user uniqueness across multiple identity providers, while also encouraging your identity provider to support a unique ID field, or choose a provider who does.

8.20 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

Percona LLC and/or its affiliates ©2024

9. Release notes

9.1 Release Notes

- Percona Monitoring and Management 2.42.0
- Percona Monitoring and Management 2.41.2
- Percona Monitoring and Management 2.41.1
- Percona Monitoring and Management 2.41.0
- Percona Monitoring and Management 2.40.1
- Percona Monitoring and Management 2.40.0
- Percona Monitoring and Management 2.39.0
- Percona Monitoring and Management 2.38.1
- Percona Monitoring and Management 2.38.0
- Percona Monitoring and Management 2.37.1
- Percona Monitoring and Management 2.37.0
- Percona Monitoring and Management 2.36.0
- Percona Monitoring and Management 2.35.0
- Percona Monitoring and Management 2.34.0
- Percona Monitoring and Management 2.33.0
- Percona Monitoring and Management 2.32.0
- Percona Monitoring and Management 2.31.0
- Percona Monitoring and Management 2.30.0
- Percona Monitoring and Management 2.29.1
- Percona Monitoring and Management 2.29.0
- Percona Monitoring and Management 2.28.0
- Percona Monitoring and Management 2.27.0
- Percona Monitoring and Management 2.26.0
- Percona Monitoring and Management 2.25.0
- Percona Monitoring and Management 2.24.0
- Percona Monitoring and Management 2.23.0
- Percona Monitoring and Management 2.22.0
- Percona Monitoring and Management 2.21.0
- Percona Monitoring and Management 2.20.0
- Percona Monitoring and Management 2.19.0
- Percona Monitoring and Management 2.18.0
- Percona Monitoring and Management 2.17.0
- Percona Monitoring and Management 2.16.0
- Percona Monitoring and Management 2.15.1
- Percona Monitoring and Management 2.15.0
- Percona Monitoring and Management 2.14.0

Percona LLC and/or its affiliates ©2024

- Percona Monitoring and Management 2.13.0
- Percona Monitoring and Management 2.12.0
- Percona Monitoring and Management 2.11.1
- Percona Monitoring and Management 2.11.0
- Percona Monitoring and Management 2.10.1
- Percona Monitoring and Management 2.10.0
- Percona Monitoring and Management 2.9.1
- Percona Monitoring and Management 2.9.0
- Percona Monitoring and Management 2.8.0
- Percona Monitoring and Management 2.7.0
- Percona Monitoring and Management 2.6.1
- Percona Monitoring and Management 2.6.0
- Percona Monitoring and Management 2.5.0
- Percona Monitoring and Management 2.4.0
- Percona Monitoring and Management 2.3.0
- Percona Monitoring and Management 2.2.2
- Percona Monitoring and Management 2.2.1
- Percona Monitoring and Management 2.2.0
- Percona Monitoring and Management 2.1.0
- Percona Monitoring and Management 2.0.1
- Percona Monitoring and Management 2.0.0

9.1.1 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-06-11

9.2 Percona Monitoring and Management 2.42.0

Release date	**June 11th, 2024**
Installation	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

It enables you to observe the health of your database systems, explore new patterns in their behavior, troubleshoot them and execute database management operations regardless of whether your databases are located on-premises or in the cloud.

Percona LLC and/or its affiliates ©2024

9.2.1 What's new in this release

This release introduces support for Ubuntu 24.04, configurable metrics resolutions per service, improved connection management for PostgreSQL and PMM Agent, and experimental dashboards for PMM self-monitoring and MongoDB.

The release also brings enhancements to MySQL Query Response Time Details, adds new metrics and labels for PostgreSQL, and addresses several issues related to MongoDB monitoring, upgrades, and Query Analytics and High Availability documentation.

Experimental dashboards in PMM are new or redesigned dashboards that are released as a preview feature for you to test and provide feedback, as we hope to include them as defaults in a future release.

Keep in mind that these dashboards are not considered fully stable or complete, and their design, functionality, and metrics may change in future releases based on user feedback and further development.

9.2.2 Release highlights

9.2.3 Ubuntu 24.04 Noble Numbat support

PMM now officially supports monitoring of databases running on the recently released Ubuntu 24.04 Noble Numbat operating system.

With this addition, PMM continues to provide comprehensive coverage across a wide range of Linux distributions commonly used in database deployments.

Check the full list of supported platforms on the Percona Software Support Lifecycle webpage.

For information on installing the PMM Client on Linux, see Set up PMM Client.

9.2.4 Configurable metrics resolutions per service

You can now configure metrics resolutions on a per-service basis by setting Low-Resolution (LR), Medium-Resolution (MR), and High-Resolution (HR) settings for each exporter individually.

Customizing the resolution settings for individual services means that you can fine-tune your PMM setup to balance data granularity and resource consumption. This will enable you to:

- allocate resources efficiently by focusing on high-resolution data for key services
- reduce storage requirements by adjusting resolution for less critical components
- align your monitoring setup with the specific needs of your environment

This feature is currently accessible via the PMM API and will be integrated into the user interface in a future release.

API call example:

```
curl --request POST \
    --url https://<your-pmm-address>/v1/inventory/Agents/ChangePostgresExporter \
    --header 'accept: application/json' \
    --header 'content-type: application/json' \
    --data '
{
"common": {
    "metrics_resolutions": {
    "hr": "60s",
    "mr": "300s",
    "lr": "3600s"
    }
},
"agent_id": "<agent-id>"
}  '
```


Percona LLC and/or its affiliates ©2024

9.2.5 Improved connection management for PostgreSQL Services and pmm-agent

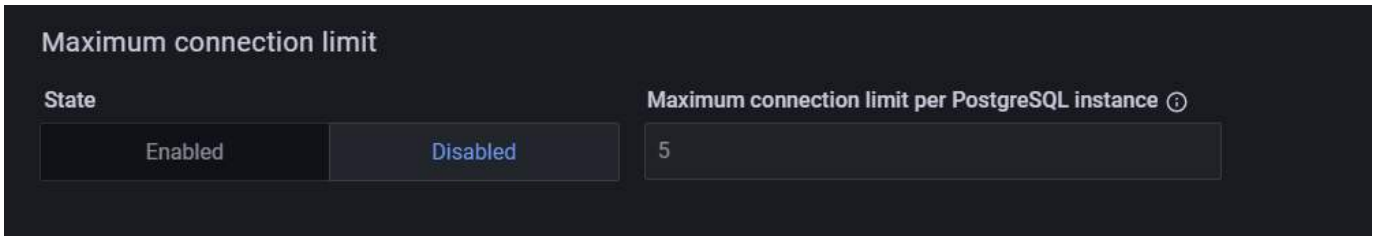
PMM now offers enhanced connection management capabilities for PostgreSQL services and the PMM Agent, to optimize resource utilization and prevent potential performance issues caused by excessive connections.

PostgreSQL services

When adding a new PostgreSQL services, you can now set a maximum limit on the number of connections that the PostgreSQL exporter can open to the same PostgreSQL instance.

This feature, previously available only through the CLI, is now also accessible through the PMM web interface as well:

- PostgreSQL service: **PMM Configuration > Add Service > PostgreSQL > Configuring PostgreSQL Service**
- RDS PostgreSQL service: **PMM Configuration > Add Service > Amazon RDS > Discover with Amazon Credentials > RDS PostgreSQL**



By setting a maximum connection limit, you can prevent excessive connections during concurrent operations, and ensure that connections are closed promptly to avoid idle connections.

When adjusting the maximum number of connections, consider the following:

- higher values might be needed for larger or busier instances.
- setting the limit too high can impact performance.
- if no limit is specified or the option is disabled, the server will manage the connection limits automatically.

PMM Agent

Similarly, you can now limit the number of connections that the PMM Agent opens to monitored databases for QAN, Advisors, and other systems.

Starting with this release, pmm-agent tracks all required resources (primarily connections), and opens no more than two connections per database instance simultaneously.

You can change this default value by setting a parameter or environment variable when starting the pmm-agennt.

For more information on configuring the PMM Agent and available parameters, see the PMM Agent help documentation by running `pmm-agent --help`.

9.2.6 Experimental PMM self-monitoring dashboard

We've added a new experimental **PMM Health** dashboard to provide detailed insights into the health and performance of PMM itself.

The dashboard reflects the PMM architecture and covers key components such as overall component status, node health, Query Analytics, PMM-managed, Grafana, VictoriaMetrics, ClickHouse, and PostgreSQL metrics.

This initial version is available in the **Experimental** folder after updating PMM and contains a starter set of key metrics to help identify potential issues and ensure optimal operation.

Percona LLC and/or its affiliates ©2024

For more information about this new dashboard, see the Keeping an eye on the eye blog post.

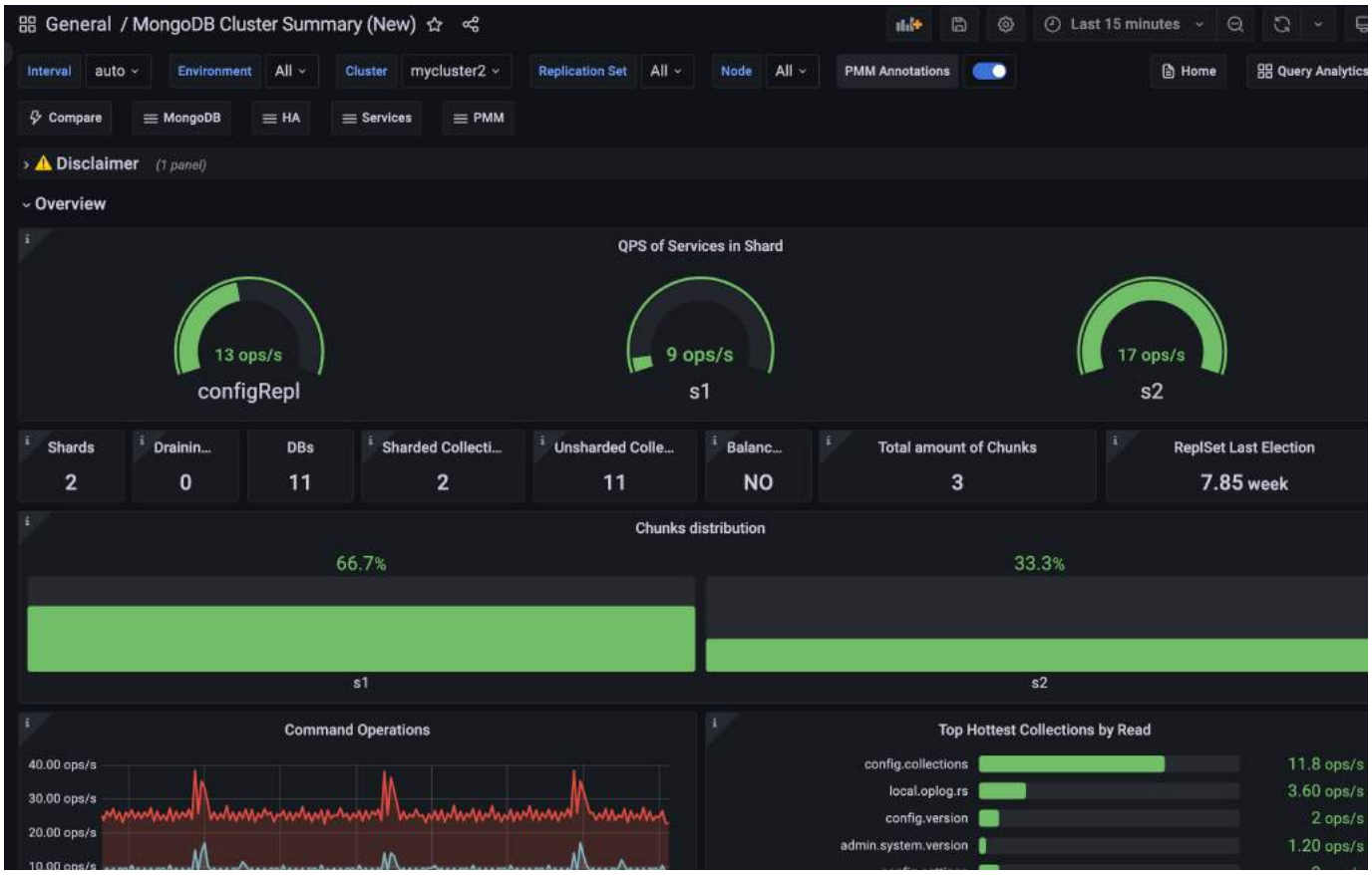
9.2.7 Experimental MongoDB dashboards: Sharded Cluster and Replica Set

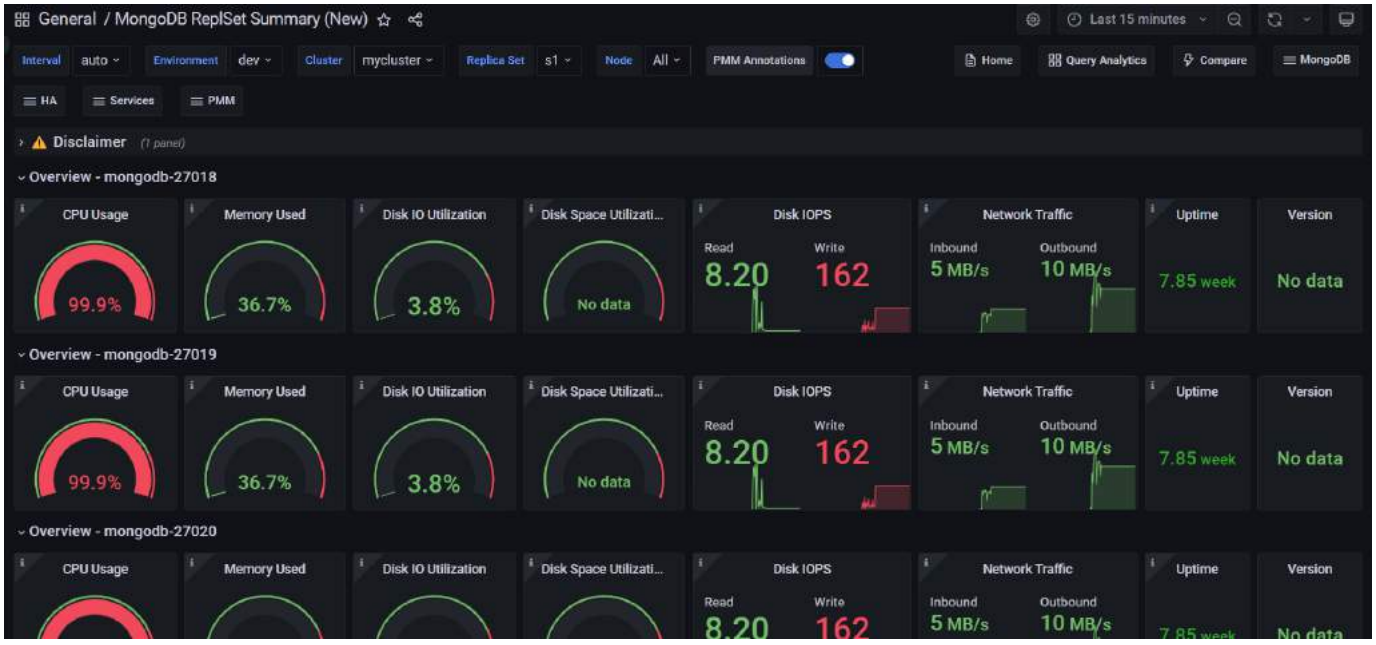
Along with the PMM Health Dashboard, we're also introducing experimental updates to two key MongoDB dashboards: the **Cluster Summary** and **ReplicaSet** dashboards. These new versions, accessible from the MongoDB section within PMM, have been redesigned to address feedback regarding readability and data density.

The design and navigation are simplified here, to focus on the essential parameters and metrics most relevant to MongoDB performance monitoring.

We encourage you to explore these experimental dashboards and provide feedback in the PMM forum to help us refine them.

Percona LLC and/or its affiliates ©2024





Improvements

- PMM-3303 - We've introduced an experimental PMM Health dashboard in the Experimental folder, offering detailed insights into PMM's health and performance by covering key components such as node health, Query Analytics, Grafana, VictoriaMetrics, ClickHouse, and PostgreSQL metrics.
- PMM-13123 - Enhanced the **Query Response Time Distribution** graphs in the **MySQL Query Response Time Details** dashboard to provide more granular insights into query performance. Previously, the graphs did not

Percona LLC and/or its affiliates ©2024

include buckets for queries with response times less than 100 milliseconds, the range which many queries typically fall into. This limitation made it difficult to analyze the performance of sub-100ms queries, especially those in the sub-1ms range.

- PMM-13075 - Added support for Ubuntu 24.04 support.
- PMM-12896, PMM-12895, PMM-12971 - Added per-database metrics collection and control of parameters, enabling more granular data gathering without overusing connections.
- PMM-12994 - Added labels to the `pg_replication_slot_slot_is_active` and `slot_current_wal_lsn` metrics in the postgres_exporter to identify the replication type (logical or physical) and the plugin used for the slot.
- PMM-12753 - Added experimental versions of the MongoDB Cluster Summary and ReplicaSet dashboards with a streamlined design and simplified navigation, focusing on essential metrics for monitoring MongoDB performance.
- PMM-11583 - Added support for the `innodb_redo_log_capacity` variable introduced in MySQL 8.0, ensuring accurate and consistent data representation in the InnoDB Logging graphs.
- PMM-11278 - Improved the Query Analytics documentation to explain how QAN collects data. For more information, see the **Query Analytics under the hood** section in the Query Analytics topic.
- PMM-13119 - Improved the High Availability documentation Improved the High Availability documentation to clearly outline the available options for HA in PMM. For more information, see the Set up PMM for HA topic.

Fixed issues

- PMM-12349 - Fixed an issue in the MongoDB RepSet Summary dashboard where incorrect data was displayed when a node in the replica set was down, ensuring that the dashboard accurately reflects the status and version information of the nodes even when they are unavailable.
- PMM-12522 - Resolved an issue where high resolution collection of MongoDB sharding information caused timeouts.
- PMM-12880 - Fixed an issue where the `pmm-admin --tls-skip-verify` flag did not work correctly when using x509 authentication for MySQL, ensuring that the flag now properly skips certificate verification when connecting to MySQL instances with self-signed or untrusted certificates.
- PMM-12962 - Updated the balancer metrics in MongoDB exporter to accurately calculate and report the `mongodb_mongos_sharding_balancer_enabled` metric, providing a correct indication of whether the balancer is currently enabled or disabled in the sharded cluster. Additionally, the `mongodb_mongos_sharding_chunks_is_balanced` metric has been renamed to `mongodb_mongos_sharding_chunks_is_balancer_running`. Update any alerts or dashboards using the old metric name to ensure they continue working correctly.
- PMM-12989 - Fixed an issue where incorrect log entries persisted when monitoring Arbiter nodes in a MongoDB replica set.
- PMM-12998 - Fixed an issue where upgrading PMM Server from versions older than 2.41.0 using Amazon Machine Image (AMI) or Open Virtualization Format (OVF) was unstable, causing the upgrade process to fail.
- PMM-9403 - Fixed an issue where the MongoDB Replication Lag graph displayed incorrect and identical values for all nodes, by updating the metric to differentiate between instances and accurately represent the replication lag of each node in the cluster.

9.2.8 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2024-06-11

9.3 Percona Monitoring and Management 2.41.2

Release date	**March 20th, 2024**
Installation	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

It enables you to observe the health of your database systems, explore new patterns in their behavior, troubleshoot them and execute database management operations—regardless of whether your databases are located on-premises or in the cloud.

9.3.1 Release highlights

Debian 12(Bookworm) pmm-client packages

Starting with PMM 2.41.2, we now offer pmm-client packages for the latest version of Debian. You can install these packages by following the instructions in our documentation.

Experimental dashboards

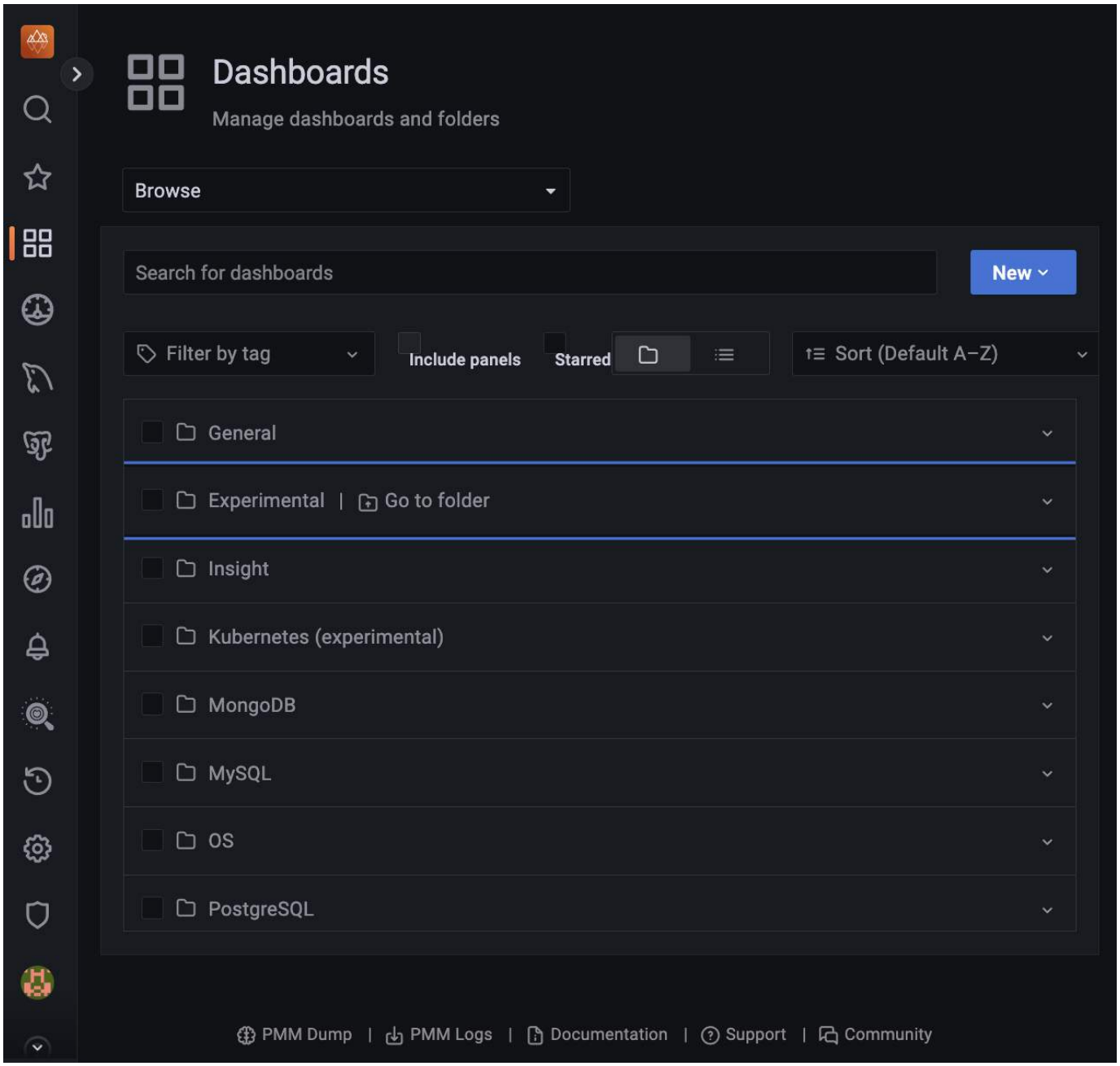
> **warning**
>
> These **experimental** dashboards are subject to change. It is recommended to use these dashboards for testing purposes only.

As part of PMM 2.41.2, we have added the following experimental dashboards.

- Reworked Kubernetes dashboards
- Databases overview dashboard
- PostgreSQL Instance Summary dashboard
- PostgreSQL Checkpoints, Buffers and WAL Usage dashboard
- PostgreSQL Patroni dashboard

To access the experimental dashboards, go to the PMM home page and then navigate to **Dashboards > Experimental** from the side menu.

Percona LLC and/or its affiliates ©2024



Alternatively, go to the PMM home page, navigate to **Dashboards** from the side menu, and enter the dashboard name in the **Search for dashboards** field.

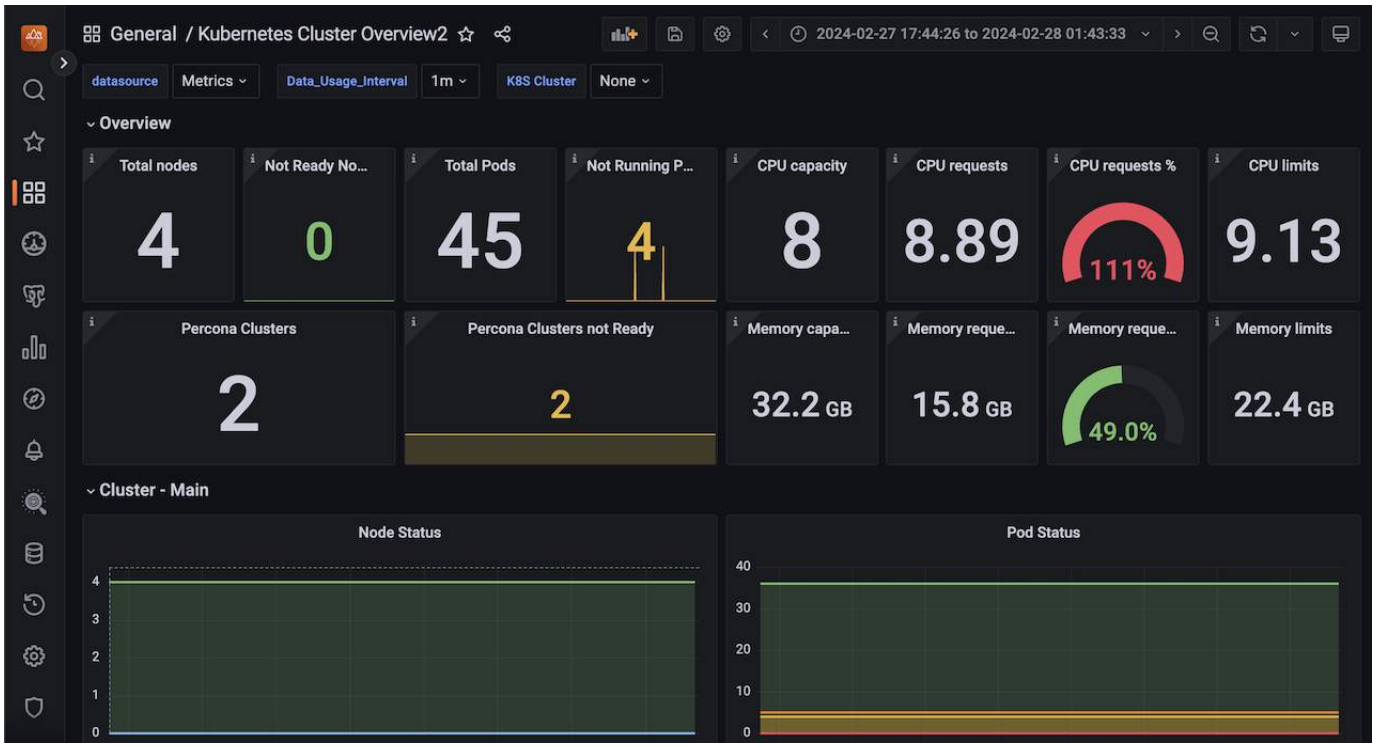
REWORKED KUBERNETES DASHBOARDS

PMM 2.41.2 now features redesigned Kubernetes dashboards that offer improved clarity and usability.

- The redundant information has been removed from some of these dashboards.
- The **Overview** dashboard, which has now been renamed as **Kubernetes Cluster Overview**, has been reorganized and improved for better functionality.
- The **DB Cluster** dashboard has been renamed as **Databases on Kubernetes** and now displays dynamic information. Furthermore, it separates database/proxy details and shows resource limits.

These changes aim to improve user experience while adhering to PMM standards, particularly for managing multiple clusters.

Percona LLC and/or its affiliates ©2024



DATABASES OVERVIEW DASHBOARD

The latest release introduces a new dashboard that offers a simplified overview of the databases being monitored by PMM. This dashboard aims to provide a centralized and unified place where the crucial parameters for database performance can be easily accessed.

If you're looking for more information on this dashboard, see the blog post.

POSTGRESQL INSTANCE SUMMARY DASHBOARD

The PostgreSQL Instance Summary dashboard is a newly designed feature that displays the critical PostgreSQL metrics. This dashboard aims to assist DBAs and developers in identifying PostgreSQL issues quickly.

Percona LLC and/or its affiliates ©2024

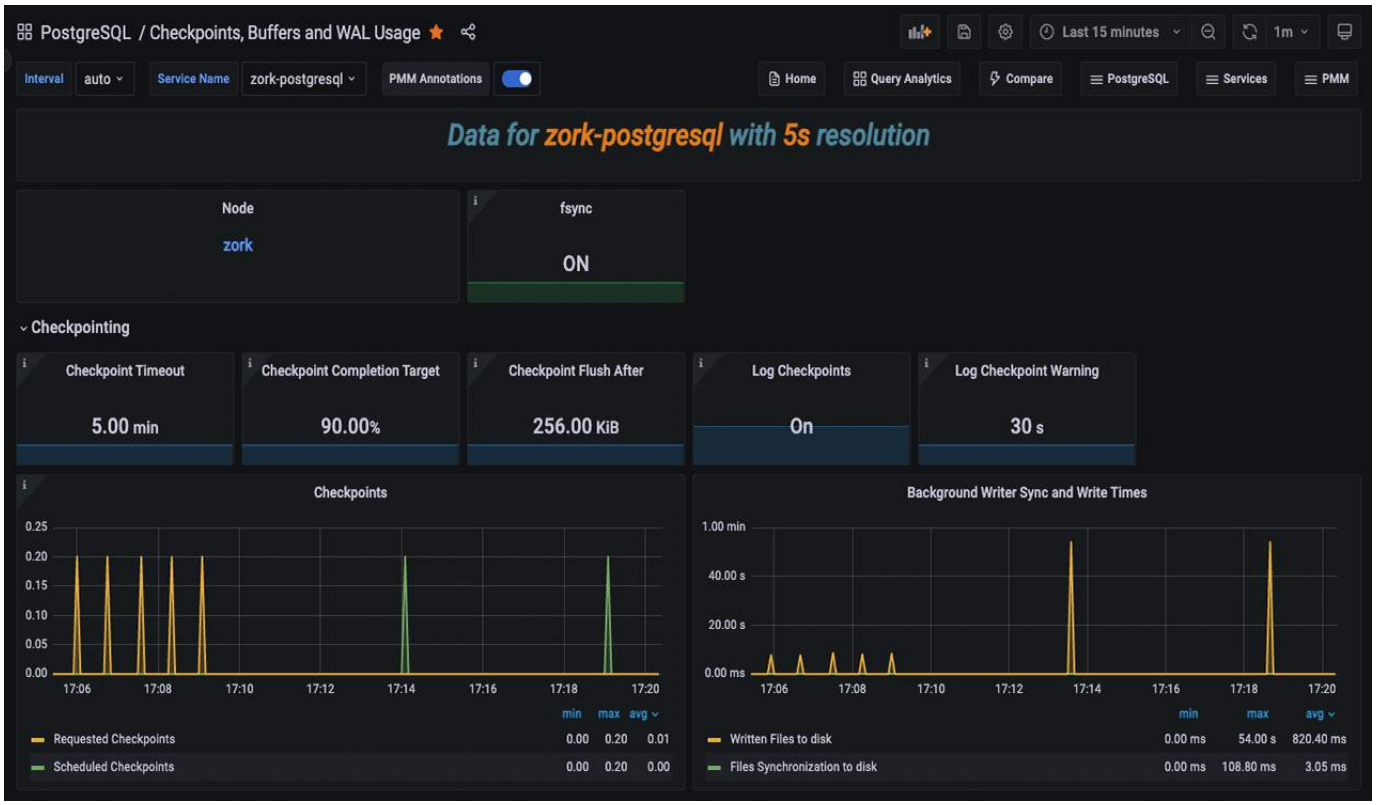
POSTGRESQL CHECKPOINTS, BUFFERS AND WAL USAGE DASHBOARD

In our continuous effort to enhance our dashboards, we have introduced this new experimental dashboard that provides more data about your PostgreSQL instances.

By leveraging this data, administrators can gain insights into their PostgreSQL servers and fine-tune their performance. Understanding and managing checkpoints, buffers, and WAL usage contribute to a well-performing and reliable PostgreSQL environment.

If you’re looking for in-depth insights into this dashboard, refer to our blog post.

Percona LLC and/or its affiliates ©2024



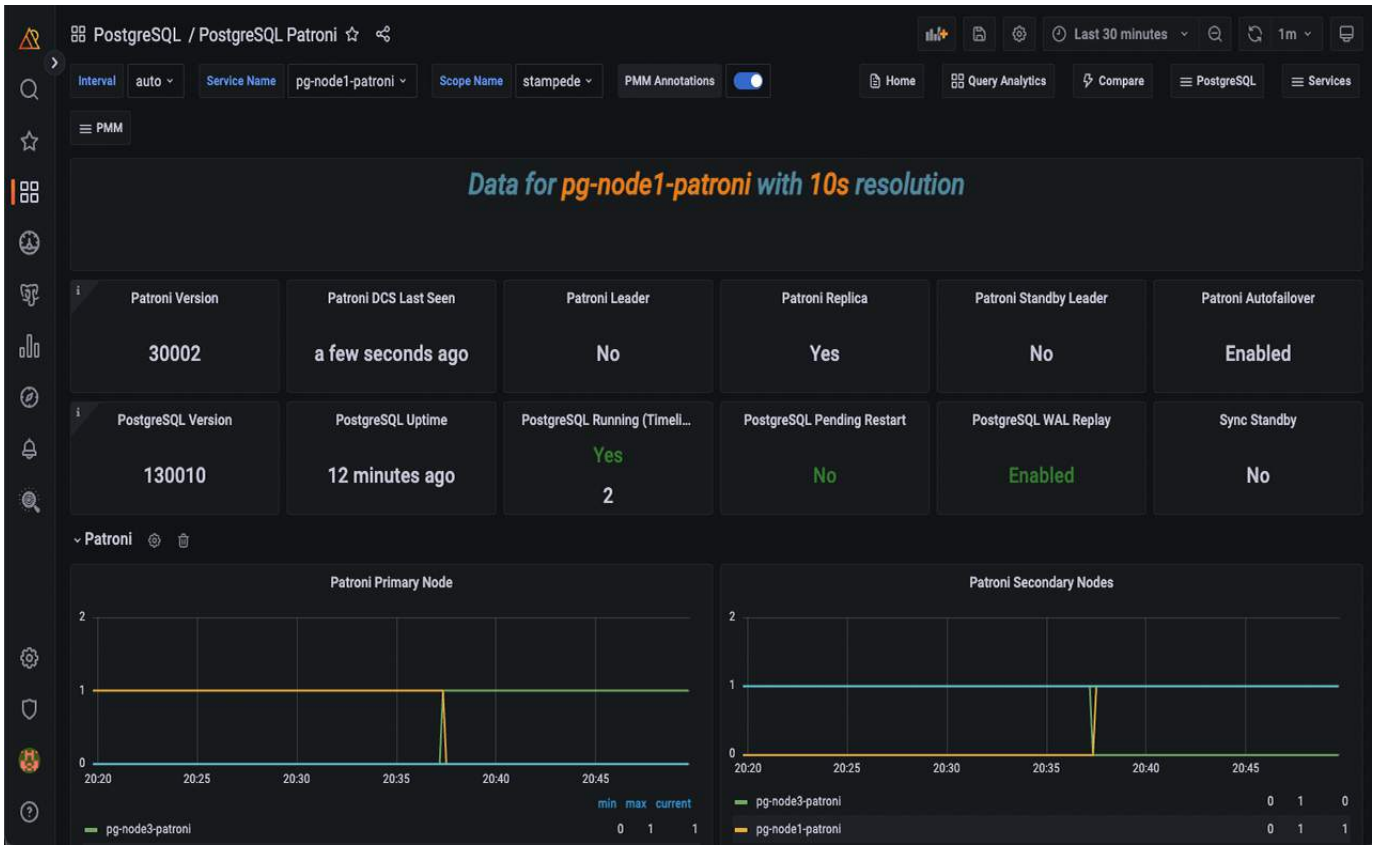
POSTGRESQL PATRONI DASHBOARD

Starting with PMM 2.41.2, we have included a new dashboard designed to monitor Patroni as an external service. For more information, see our documentation on external services.

This dashboard facilitates gathering more data for your PostgreSQL cluster inside PMM.

Read our blog post to learn more about this dashboard and how to add Patroni monitoring.

Percona LLC and/or its affiliates ©2024



Important

We would appreciate your feedback on these experimental dashboards to help us enhance it further.

Percona LLC and/or its affiliates ©2024

9.3.2 Improvements

- PMM-10974 - Starting from PMM version 2.41.2, PMM has improved its stat tracking feature. In case `pg_stat_monitor` is not available, PMM will now use `pg_stat_statements` as a fallback option.
- PMM-12884 - In PMM dump, the **Address** field on the **Send to Support** screen now comes with the address and port pre-filled, making it easier to send logs to support.
- PMM-12887 - [Tech Preview] This new experimental dashboard has been added in PMM 2.41.2 for monitoring PostgreSQL checkpoints, buffers and WAL usage.
- PMM-12888 - [Tech Preview] This new experimental dashboard has been added in PMM 2.41.2 to monitor Patroni as an external service.
- PMM-12960 - [Tech Preview] The latest PMM release includes redesigned Kubernetes dashboards with improved clarity and usability.
- PMM-12981 - Starting with PMM 2.41.2, PMM can now collect wraparound metrics for PostgreSQL. By monitoring these wraparound-related metrics, PostgreSQL administrators can proactively manage wraparound risks and prevent potential data corruption and downtime.
- PMM-12894 - We have limited the number of connections used by postgres_exporter. Additionally, we have fixed an issue where connections are closed as soon as they are no longer needed, thereby preventing any hanging or idle connections.
- PMM-12897 - Previously, two heavy queries were running for each database in the PostgreSQL server at medium resolution, which impacted the performance. These queries have now been moved to low resolution to improve performance.

9.3.3 Fixed issues

- PMM-12806 - In previous versions, PMM Server used certain non-configurable Victoria Metrics settings with default values. With this release, we are updating this behavior to enable you to modify default values for Victoria Metrics settings using environment variables. Previously, environment variables were overlooked as Victoria metrics prioritized specific command line flags passed by PMM over environment variables.
- PMM-12348 - The ClickHouse engine was updated from the ordinary engine to the Atomic engine, which is the most robust engine, after upgrading PMM to version 2.41.0.
- PMM-12785 - Fixed an issue where VictoriaMetrics scrape configuration contained unsupported parameters.
- PMM-12805 - While monitoring MongoDB servers, the disk space was being consumed due to the `CommandNotSupportedOnView` message in the logs. This issue has now been resolved.
- PMM-12809 - Fixed several CVEs in PMM versions v2.40.1 and above.
- PMM-12852 - An internal server error was thrown while attempting to modify any setting in PMM. This issue has been resolved now.
- PMM-12986 - The `PostgreSQL connections in use` alert was triggering false positive notifications, but the issue has been resolved now.
- PMM-12948 - Fixed an issue where the wrong information was being displayed regarding the uptime of the PostgreSQL instance.
- PMM-12997 - The `pmm-agent` output was changed from `stdout` to `stderr`, which caused the automation to break. The issue has been resolved now.

9.3.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2024-03-22

9.4 Percona Monitoring and Management 2.41.1

Release date	**Feb 1st, 2024**
Installation	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

It enables you to observe the health of your database systems, explore new patterns in their behavior, troubleshoot them and execute database management operations—regardless of whether your databases are located on-premises or in the cloud.

9.4.1 Release Highlights

Improved support for Arbiter nodes

We've made it easier to add and monitor MongoDB Arbiter nodes in PMM, as this no longer requires workarounds for cases where authorization is enabled on the MongoDB replica set. Additionally, Arbiter nodes are now displayed correctly on the dashboards:

New metric for sharded MongoDB insights

We've added a new metric to our MongoDB exporters: `mongodb_shards_collection_chunks_count`. This metric is available for sharded Mongo instances, offering insights into the distribution of collection chunks across different shards.

While this version is not currently integrated into any dashboard, you have the flexibility to incorporate it into your custom dashboards. For comprehensive instructions on managing dashboards, check out this video tutorial and the documentation.

New experimental MongoDB Instance Summary dashboard

This release also introduces a new MongoDB dashboard that offers a simplified view of the most critical MongoDB metrics. You can find the new **MongoDB Instance Summary** dashboard in the **Experimental** folder of your PMM installation. This dashboard is currently in Technical preview so we encourage you to evaluate its performance with your MongoDB instance and share your valuable feedback on our forum.

Percona LLC and/or its affiliates ©2024

9.4.2 Improvements

- PMM-12390 - [Backup Management]: The **Service name** field on the Create Backup pages is now case-insensitive, allowing for more comprehensive and accurate results retrieval.
- PMM-12712 - MongoDB sharded cluster metrics now include a new metric to represent database and collection distribution across different shards. This helps gain insights into the overall balance of shards within MongoDB setups.
- PMM-12510 - Improved workflow for adding MongoDB Arbiter nodes to PMM, which also ensures that Arbiter nodes are now displayed correctly on the dashboards.
- PMM-12750 - [Dashboards]: Introduced an experimental MongoDB Instance Summary dashboard to provide a rapid overview of key MongoDB metrics.
- PMM-12866 - [Dashboards]: Minor UX improvements to the K8s experimental dashboard (Tech Preview).

9.4.3 Bugs Fixed

- PMM-12652 - Fixed issue where PMM failed to start after an upgrade, due to SELinux restrictions on accessing a PMM file created with root ownership on Podman. This fix applies to upgrades from PMM 2.41.0 to newer versions.
- PMM-9825 - [QAN]: PostgreSQL databases, monitored using the pg_stat_monitor extension v2.0+, were previously displayed with User IDs rather than actual User Names in the **Service Name** column on the **Query Analytics (QAN)** page. This issue is now fixed.
- PMM-12290 - [QAN]: Fixed bugs with SQL comments parsed in queries when they were not intentionally enabled for remotely added PostgreSQL and MySQL.
- PMM-12621 - [Dashboards]: Added support for the Anonymous mode to show all Service types on the main menu. This ensures users can access the dashboards, even when PMM is configured in Anonymous mode.
- PMM-12425 - [QAN]: The **Absolute time range** time picker on the **Query Analytics (QAN)** page no longer displays an error when using 'now' instead of a specific timestamp for filtering dashboard data.
- PMM-12473 - Fixed monitoring of external services when query string parameters are passed to `--metrics-path`.
- PMM-9407 - Fixed missing custom `queries.yaml` file for medium-resolution in Debian packages for PostgreSQL.
- PMM-12350 - Fixed issue of flood logs in `mongod_exporter` when connected to Mongos.
- PMM-12738 - Fixed issue that prevented PMM from running when using Helm with customer certificates.
- PMM-12781 - Corrected a permission error that was flooding PostgreSQL logs by ensuring proper execution permissions for the `pg_ls_waldir` function.

9.4.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-02-02

9.5 Percona Monitoring and Management 2.41.0

Release date	**Dec 12, 2023**
Installation	Installing Percona Monitoring and Management

Percona LLC and/or its affiliates ©2024

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

9.5.1 Release Highlights

Streamlined database problem reporting to Percona

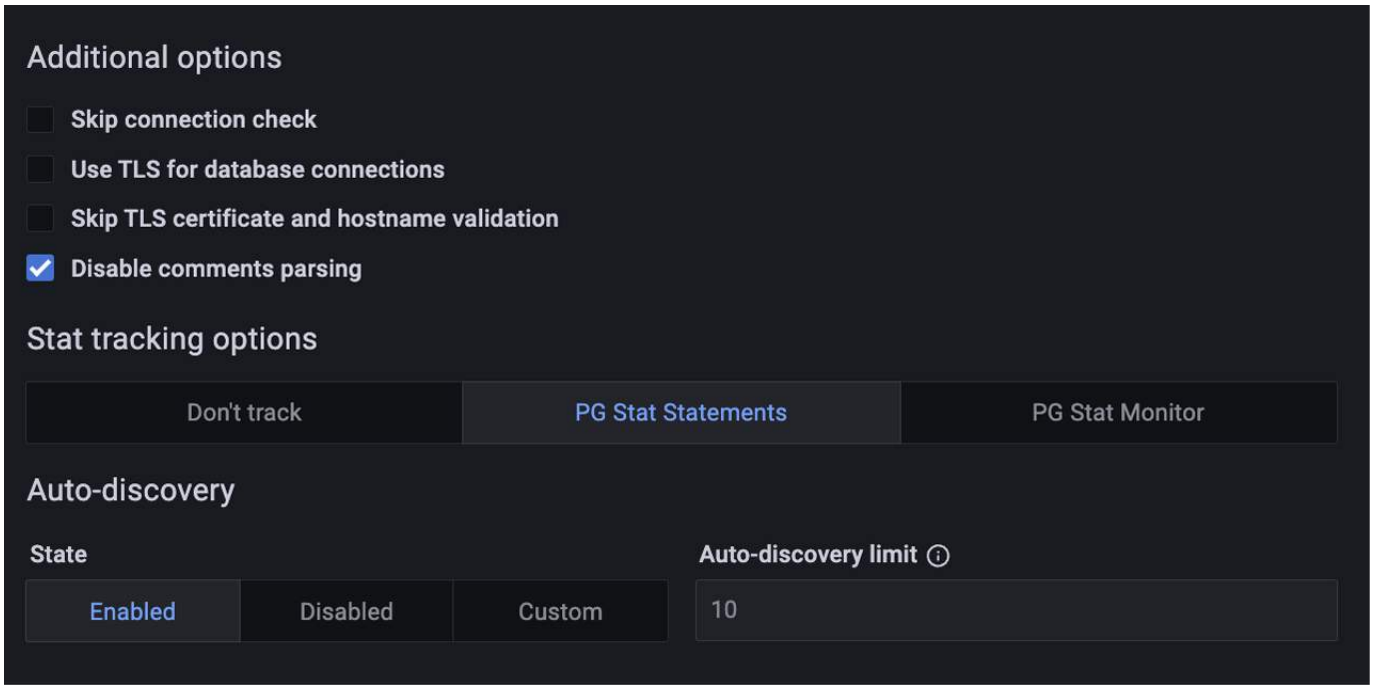
To improve the gathering and sharing of PMM metrics and data, we've now integrated the pmm_dump client utility into PMM. Initially a standalone client for PMM Server, PMM Dump is now accessible in the PMM user interface.

This integration enables you to collect PMM data to share with our Support team.

To get started, in the main menu, go to **Help > PMM Dump** and select either to export a dataset locally or upload it to our SFTP servers using the credentials generated through your Percona Support ticket.

PostgreSQL monitoring: optimizing performance

PMM 2.41.0 introduces limit for Auto-discovery in PostgreSQL, a feature that dynamically discovers all databases in your PostgreSQL instance. Limiting Auto-discovery reduces connections and prevents high CPU and RAM usage caused by multiple databases, thus optimizing performance.



For details, see documentation.

PMM DBaaS functionality evolution into Percona Everest

We have decided to separate our DBaaS offering into an independent product. Consequently, we are discontinuing the DBaaS functionality in PMM and offering a migration path to Everest.

While the DBaaS functionality will remain available in PMM versions 2.x, all future updates and enhancements will be exclusively accessible through the Percona Everest interface. For a more streamlined and robust database deployment experience, try Percona Everest.

Percona LLC and/or its affiliates ©2024

Technical Preview of PMM in high availability (HA) mode

> **Important**
>
> Disclaimer: This feature has been added in PMM 2.41.0 and is currently in Technical Preview. Early adopters are advised to use this feature for testing purposes only as it is subject to change.

Starting with this release, PMM can now be run in high availability mode with several PMM server applications and independent data sources.

Currently, PMM servers provide high availability, but users are responsible for maintaining the high availability of the data sources used by PMM.

These data sources are:

- ClickHouse: An open-source, fast analytical database.
- VictoriaMetrics: A scalable, long-term storage solution for time series data.
- PostgreSQL: A powerful open-source relational database management system used in this setup to store PMM data like inventory, settings, and other feature-related data.

If you're looking to dive deep into this feature, see our comprehensive documentation.

9.5.2 New Features

- PMM-12459 - The pmm_dump client utility previously available as a standalone client for PMM Server is now readily accessible within the PMM user interface.

9.5.3 Improvements

- PMM-11341 - PMM 2.41.0 introduces limit for Auto-discovery in PostgreSQL, a feature that dynamically discovers all databases in your PostgreSQL instance. Limiting Auto-discovery reduces connections and prevents high CPU and RAM usage caused by multiple databases, thus optimizing performance.
- PMM-12375 - Starting with PMM 2.41.0, each instance of a service gets a `version` attribute in the PMM Inventory UI.
- PMM-12422 - PMM 2.41.0 introduces a new flag called `--expose-exporter`. When you enable this flag any IP address, either from a local system or from anywhere on the internet, can access exporter endpoints. If the flag is not enabled, the exporter will be available only locally.
- PMM-12544 - Added deprecation notices to the PMM documentation DBaaS pages. For a more streamlined and robust database deployment experience, try Percona Everest.
- PMM-12549 - Added support for the latest MongoDB version. You can now use PMM to monitor MongoDB 7 databases.

9.5.4 Components upgrade

- PMM-12154 - Updated `postgres_exporter` to version 0.14.0. With this update, we have resolved several performance issues and eliminated the creation of multiple connections per database.
- PMM-12223 - Clickhouse has been updated to version 23.8.2.7, which optimizes memory and CPU usage to improve system performance.

Percona LLC and/or its affiliates ©2024

9.5.5 Bugs Fixed

- PMM-4712 - We have addressed the issue where the pprof heap reports for postgres_exporter were missing.
- PMM-12626 - Due to the packages being built on an outdated Go version, there was a potential vulnerability. We have updated Go to the latest version to mitigate this risk.
- PMM-12414 - Fixed an issue with an unexpected behavior (502 response) when accessing the `logs.zip` endpoint. This was caused by the `group_by` parameter being included in the Alertmanager configuration. Additionally, we removed AlertManager-related files from `logs.zip` since we stopped using AlertManager.
- PMM-11714 - Registering a node with the Grafana Admin flag enabled but a non-admin role was failing. This issue has now been resolved.
- PMM-12660 - Prior to version 2.41.0 of PMM, the endpoint `/v1/management/Agent/List` could deliver database certificates to the PMM UI, allowing an authenticated admin user to view the output of TLS certificates. This posed a security issue since certificates should be consumed by the backend only. We have resolved this issue now.
- PMM-12630 - When users attempted to upgrade PMM versions lower than or equal to 2.37.1, the upgrade process got stuck in a loop and failed. The issue has been resolved now.
- PMM-12725 - Fixed the pagination for QAN.
- PMM-12658 - Corrected a typo in the MongoDB cluster summary dashboard.

9.5.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-12-12

9.6 Percona Monitoring and Management 2.40.1

Release date:	**Oct 20, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open-source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

9.6.1 Fixed issues

- PMM-12592 - Fixed an issue where the **Library Panels** for the PMM dashboard were not working after upgrade to PMM 2.40.0.
- PMM-12576 - After upgrading to PMM 2.40.0, changing the Admin user's password from the terminal was not functioning. The issue has been resolved now.
- PMM-12587 - After upgrading to PMM 2.40.0, some users may experience incorrect mappings between dashboards, folders, users, and groups. This can result in either a successful upgrade or a `500 internal server error`. The issue has now been resolved.
- PMM-12590 CVE-2023-4911 is a vulnerability in the OS that PMM is based on. It has been fixed in the base OS, and the fix is available in PMM.

Percona LLC and/or its affiliates ©2024

9.6.2 How to upgrade from 2.40.0 to PMM 2.40.1

Important

The steps below are only applicable to users who have upgraded to 2.40.0 and have been experiencing the following issues:

- PMM-12592
- PMM-12587

To upgrade to PMM 2.40.1:

1. Copy the file from `/srv/backup/grafana/grafana.db` to `/srv/grafana/grafana.db`

   Caution

   This step will result in the loss of all changes made to Grafana upon upgrading to version 2.40.0.

   ```
   docker exec -t pmm-server cp /srv/backup/grafana/grafana.db /srv/grafana/grafana.db
   ```

2. Set permissions:

   ```
   chmod 640 /srv/grafana/grafana.db

   chown grafana:grafana /srv/grafana/grafana.db
   ```

3. Upgrade to 2.40.1 as usual.

9.6.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-10-25

9.7 Percona Monitoring and Management 2.40.0

Release date:	**Oct 05, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

Caution

If you are using the Library Panels on the PMM dashboard, avoid upgrading to PMM 2.40.0 due to a known issue. Instead, we recommend waiting for the release of PMM 2.40.1.

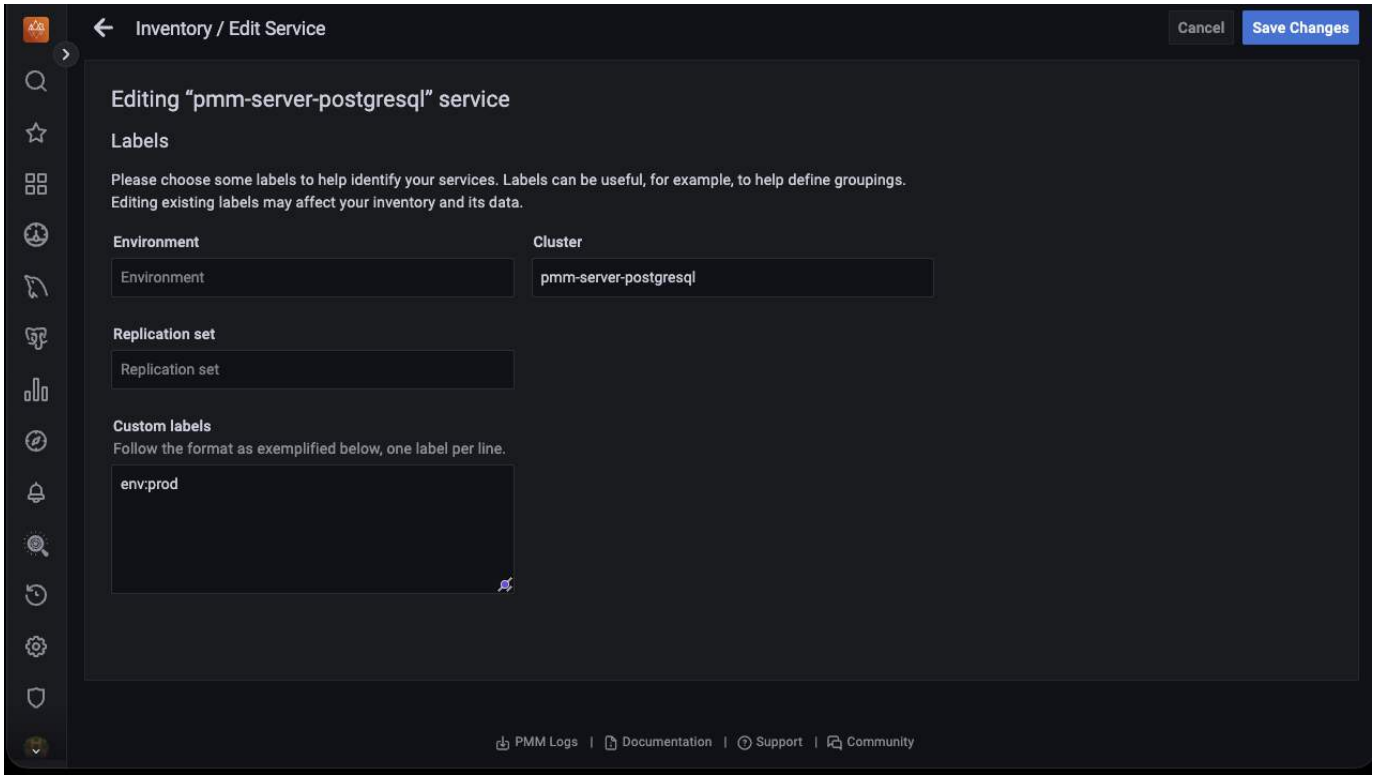
Percona LLC and/or its affiliates ©2024

9.7.1 Release Highlights

Inventory improvements

ABILITY TO UPDATE LABELS FOR EXISTING SERVICES TO PMM

Starting with PMM 2.40.0, editing labels for existing services is easier and much more efficient. You can edit a service label directly from the PMM UI without removing and re-adding it. For more information on editing labels, see documentation.



Furthermore, in our continued effort to enhance user experience, we have redesigned the following pages to make them more user-friendly and intuitive:

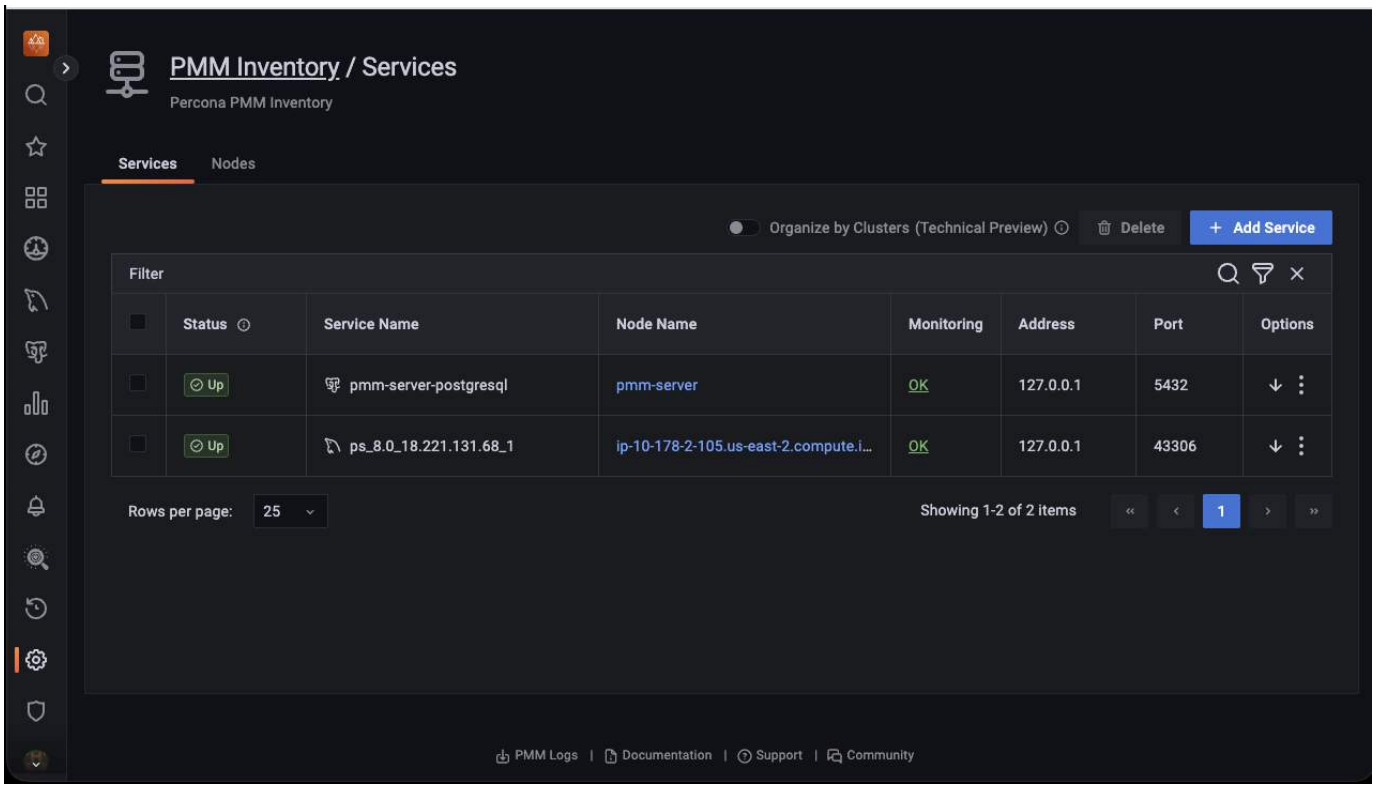
- Select service type page
- Add service page

CONNECTING SERVICES AND NODES

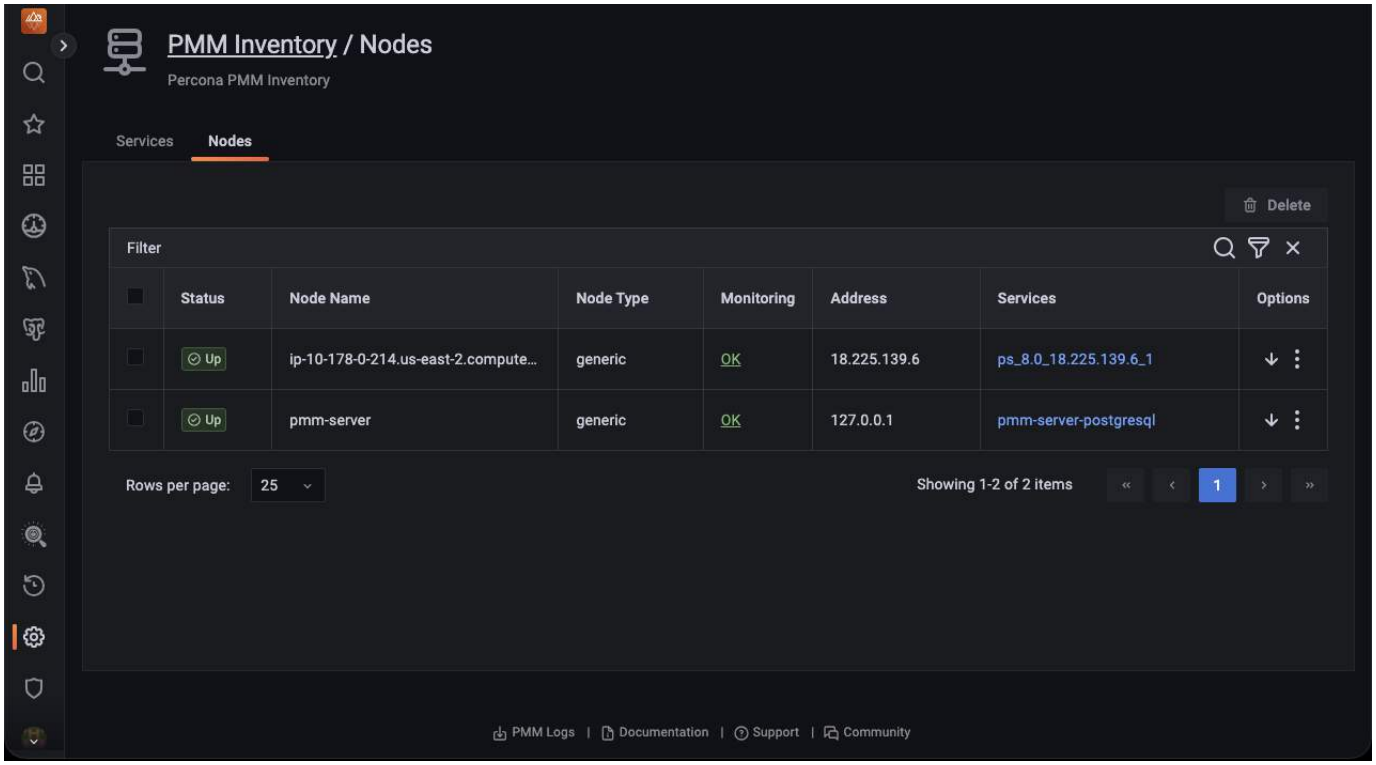
Starting with PMM 2.40.0, you can click on the link in the **Node Name** column to view the node on which a specific service is running and analyze how node-level resource utilization impacts the performance of those services.

Before introducing this feature, locating the running services on a node was cumbersome. However, with this new feature, you can effortlessly access a list of services running on a specific node and identify the node name where the service is being utilized. Additionally, the filters implemented make navigation a lot simpler.

Percona LLC and/or its affiliates ©2024



Furthermore, you can also see the service running on that specific node when you click on the link in the **Services** column.



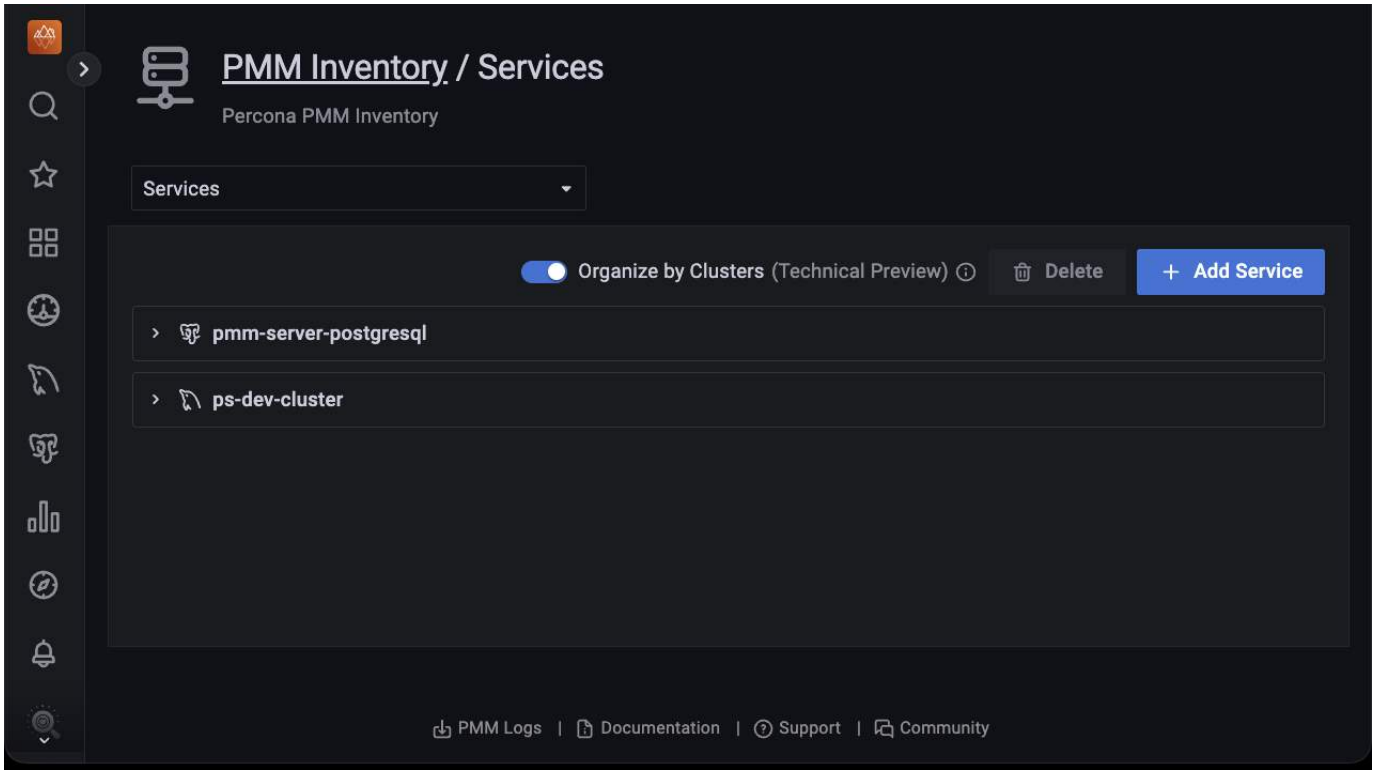
Percona LLC and/or its affiliates ©2024

CLUSTER VIEW FOR INVENTORY

Important

This feature is still in Technical Preview and is subject to change. We recommend that early adopters use this feature for evaluation purposes only.

Understanding the structure of your inventory is crucial. With the release of PMM 2.40.0, we've introduced an experimental feature that categorizes all your Services by Cluster, making it easier for you to understand your inventory. By using the **Organize by Clusters** toggle, you can view a group of services as a single cluster. PMM utilizes the cluster label to display services that belong to the same cluster.



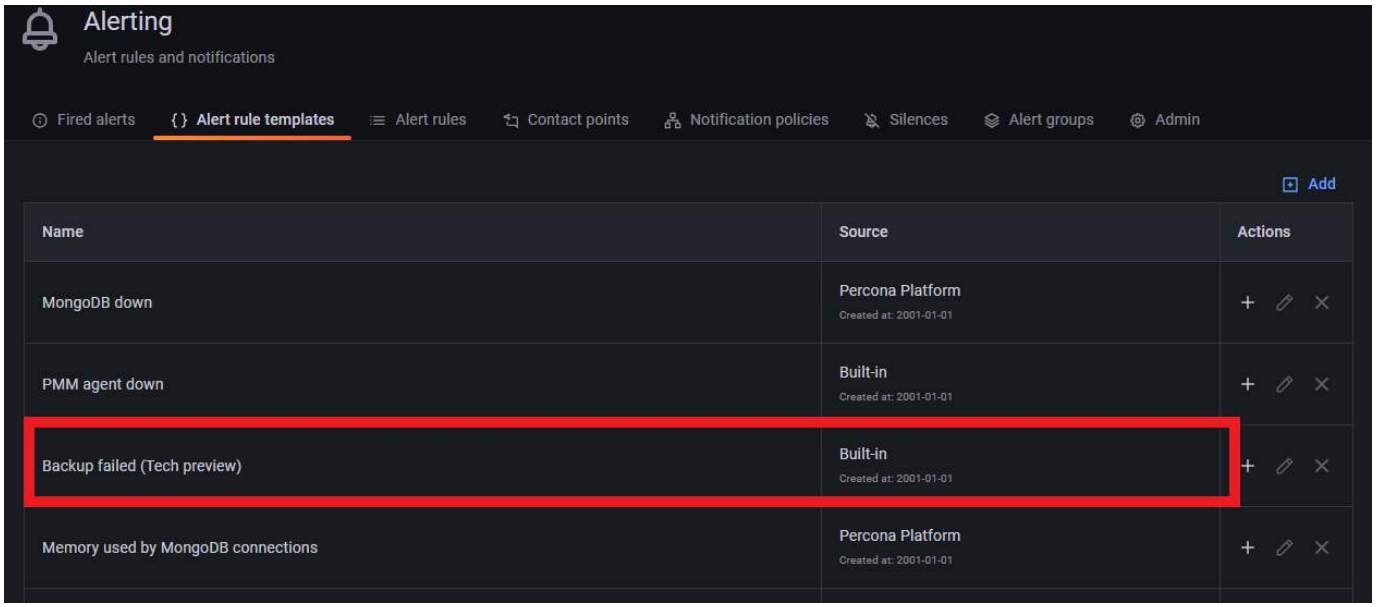
For detailed information on this feature, see documentation.

MongoDB Backup Monitoring

With this release, we are shipping the first version of a the **Backup failed** alert template which notifies of any failed MongoDB backups. This uses a new, dedicated metric, `pmm_managed_backups_artifacts`, for checking the state of backup artifacts.

This new template is currently in Technical Preview and we are keen to get your feedback around this change before we move it to General Availability.

Percona LLC and/or its affiliates ©2024



For information on working with this new template, see the Percona Alerting topic.

Components Upgrade

VictoriaMetrics has been upgraded v1.93.4.

9.7.2 New Features

- PMM-11963 - Starting with PMM 2.40.0, you can click on the link in the **Node Name** column to view the node on which a specific service is running.
- PMM-11148 - [Inventory]: We have redesigned the **Select service type** and **Add service** pages to make them more user-friendly and intuitive.
- PMM-11423 - [Inventory]: Starting with PMM 2.40.0, you can now edit service labels directly from the PMM UI without having to remove and re-add them.
- PMM-12378 - [Inventory]: We have introduced an experimental feature that categorizes all your Services by Cluster, making it easier for you to understand your inventory. By using the **Organize by Clusters** toggle, you can view a group of services as a single cluster.
- PMM-12384 - [Alerting]: Added a new built-in alert template, **Backup failed** which you can use to create alert rules that notify of failed MongoDB backups.
- PMM-9374 - [Technical Preview]: Starting with PMM 2.40.0, you can now use an external VictoriaMetrics database for monitoring in PMM. This provides multiple benefits, including scalability, resource isolation, reduced load on the PMM server, customization, etc.

9.7.3 Improvements

- PMM-4466 - Grafana now uses PostgreSQL instead of SQLite, resulting in improved performance, stability for user auth requests, data integrity, reliability, security, etc.
- PMM-12310 - Links from PMM to documentation may change, causing “broken links” in older PMM versions due to document structure changes. We have replaced all links with Short.io links specific to each document to address this. This way, we can maintain the PMM-to-doc links using a URL shortener. This ensures that the links remain accessible even if the document structure changes.
- PMM-12457 - We have added a new `node_name` property to the **PMM agent down** alert template in PMM. This allows users to easily identify the node where the failure occurred.

Percona LLC and/or its affiliates ©2024

- PMM-12488 - VictoriaMetrics has been updated to v1.93.4.
- PMM-12500 - [Technical Preview]: For Percona Operator users, we have introduced a new dashboard for K8s monitoring with PMM 2.40.0.
- PMM-11770 - PMM now automatically sets the server domain value in `grafana.ini` to the public address specified in PMM's Advanced Settings. This ensures that links generated by Grafana, such as links in alert emails, also carry that address.

9.7.4 Bugs Fixed

- PMM-10145 - When we installed an AMI image in AWS and added an Amazon RDS instance in the creating state, it caused errors. The issue has been resolved now.
- PMM-12173 - On adding several clients to PMM, unexpected and unauthorized errors caused PMM to stop responding. The issue has since been resolved now.
- PMM-12344 - An error was displayed on the **Explain** tab after selecting a PostgreSQL query. The issue has been resolved now.
- PMM-12361 - The command `pmm-admin inventory` returned GO errors. The issue has been reolved now.
- PMM-12382 - Fixed an issue where the upper-case custom labels were not being accepted.
- PMM-11371 - Usage of `pmm-admin` config subcommand shutdown pmm-agent. The issue has been resolved now.
- PMM-11603 - When running `pmm-agent` outside of systemd, adding an agent causes errors. The issue has been resolved now.
- PMM-11651 - When a user has both full access and limited access roles, only the limited role's metrics are displayed. The issue has been resolved now.
- PMM-12146 - Dead Tuples graph on PostgreSQL Vacuum Monitoring Dashboard displayed invalid percentage. The issue has been resolved now.
- PMM-12448 - Can't start backup if artifact with empty `service_id` exists.

9.7.5 Known issues

- PMM-12517 - If you have set custom dashboards as your Home dashboard in version 2.26.0, they may be lost after upgrading to version 2.40.0. To prevent this, we recommend that you back up your custom dashboard before upgrading and recover it after the upgrade.
- PMM-12576 - After upgrading PMM from version 2.39.0 to 2.40.0 (not el7) using Docker, the admin user cannot access the PMM UI. Refer to the documentation for information on the solution.
- PMM-12592 - If you are using the **Library Panels** on the PMM dashboard, avoid upgrading to PMM 2.40.0 due to a known issue.

9.7.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-10-17

Percona LLC and/or its affiliates ©2024

9.8 Percona Monitoring and Management 2.39.0

Release date:	**Aug 14, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

9.8.1 Release Highlights

MongoDB backups: UI-based restore on a new replica set cluster

Building on the recent MongoDB backup management enhancements, we have now added the option to restore backups on different clusters with the same configuration. This enables you to test restore data before deploying it on Dev environments.

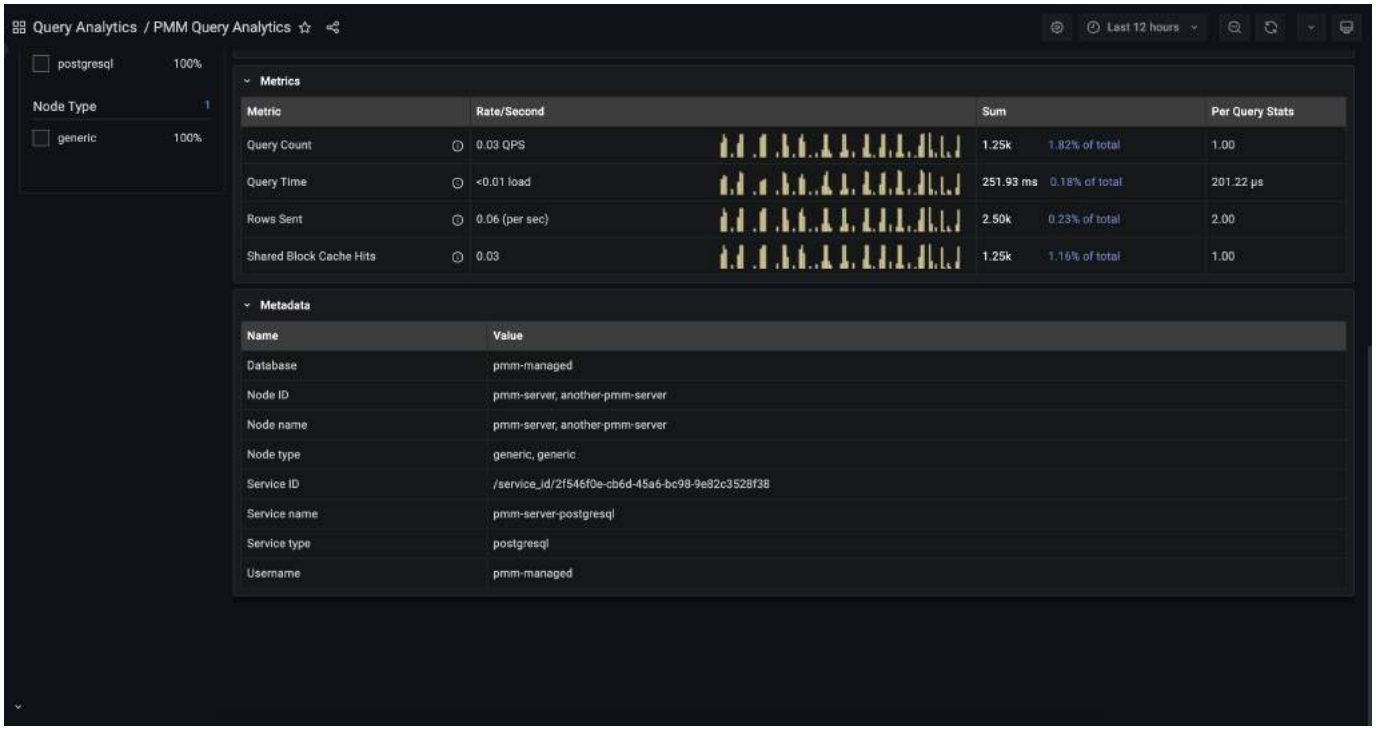
Restoring on different clusters is not available for sharded clusters. This is a complicated process that still requires manual intervention. For more information, see Restoring from a sharded cluster

Query Analytics - Meta information

QAN is an important tool that can help filter data and identify problematic queries. However, it can be challenging to determine which services, nodes, and other labels are linked to these queries.

PMM 2.39.0 now includes a metadata table for QAN to address this issue. This table contains information such as Node name, Node name, Service name, Service type, and user name for each selected query. With this feature, you can easily identify the services, nodes, and labels associated with your identified queries.

To view the metadata table, navigate to the **Main** menu and click on **Query Analytics (QAN)**. This will bring you to the **Query Analytics** page. Next, select the desired query and expand the **Metadata** tab. The metadata for that specific query will then be displayed.



Percona LLC and/or its affiliates ©2024

9.8.2 Improvements

- PMM-6279 - [QAN]: PMM 2.39.0 now includes a metadata table for each query. This table contains information such as Node name, Node name, Service name, Service type, and user name for each selected query.
- PMM-10270 - [Backup Management]: Enhanced MongoDB Backup Management to support restoring to different replica set clusters with the same configuration.
- PMM-11999, PMM-12257 - [Advisors] - The **Advisors** page now contains a new **Technology** column that shows the database type for each advisor. This helps you distinguish between MongoDB, MySQL, and PostgreSQL advisors.
- PMM-11206 - [Alerting]: You can now easily add multiple **Alert rule templates** at once, whether through the API or UI. When added in bulk, each template will be displayed separately on the **Alert rule templates** page. This makes it quicker and more efficient to manage your alert rules.
- PMM-11637 - PMM UI now reflects the new PMM logo as per the new branding.
- PMM-12070 - We have updated the deprecated Boomtable plugin to the Grafana table plugin.
- PMM-12389 - The postgres_exporter now provides more comprehensive logs that display the specific namespace and error message for any issue.

9.8.3 Bugs Fixed

- PMM-12151 - [QAN]: The Explain plan was not displayed in a few scenarios. The issue has been resolved now.
- PMM-11658 - Fixed issue that caused the pmm-agent to crash on AMI and OVF distributions.
- PMM-11992 - Postgres databases with names other than **Postgres** could not be added. The issue has been resolved now.
- PMM-12263 - On the MongoDB Cluster Summary dashboard, the **Replication Lag by Set** graph showed the wrong value if **rs** was in the primary-secondary-arbiter (PSA) configuration. It calculated the arbiter's lag from the beginning of the UNIX time (currently >54 years instead of a few seconds). The issue has been resolved now.
- PMM-12280 - Fixed an issue where the **Service Summary Panel** on Home Dashboard used the incorrect units to display `Available Memory`.
- PMM-10665 - Fixed issue that caused QAN to show incorrect values when setting non-default values for performance schema. Make sure to restart pmm-agent after changing the settings for perfschema in MySQL.
- PMM-10391 - PMM did not work with external Clickhouse versions greater than 22.5.2. The issue has been resolved now.
- PMM-12231 - In the recent versions of PMM, users have encountered issues with installing, updating and removing plugins from PMM. For more information on this issue, see Troubleshooting Plugin issues.

9.8.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-08-14

Percona LLC and/or its affiliates ©2024

9.9 Percona Monitoring and Management 2.38.1

Release date:	**July 13, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open-source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

9.9.1 Fixed issue

PMM-12293 - With this update, we are fixing an authentication issue that occurred when trying to relog into PMM using a Percona Account. Second and subsequent logins no longer result in a “User already exists” error.

9.9.2 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-07-18

9.10 Percona Monitoring and Management 2.38.0

Release date:	**Jul 3, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open-source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

9.10.1 Release Highlights

Important security updates

ORACLE LINUX 9 AS NEW BASE OS FOR PMM

As security is paramount to us, PMM 2.38.0 adds a solid and modernized enterprise-grade foundation to PMM. This release migrates our base operating system on which we build our PMM image from CentOS 7 to Oracle Linux 9. Thus, starting with PMM 2.38.0, PMM is publicly available on Oracle Linux 9.

This transition brings many benefits:

- a large number of CVEs automatically fixed
- improved upstream response time to security-related issues
- better compatibility with new technologies
- modern libraries for our team to build on

Since not all users will be able to seamlessly make the move, we will be rolling this out in phases:

Percona LLC and/or its affiliates ©2024

Automated migration paths

The following users will be able to take advantage of this immediately:

- New installations of PMM:
  - Docker containers from our Docker hub: `percona/pmm-server:2`
  - OVF: Download directly from our website
  - AMI: Loaded from AWS Marketplace
- Docker/Podman users who do full container replacement upgrades to newer versions of PMM. Your existing `docker pull` or `docker run` commands will use the new image.

Manual upgrade options

For users upgrading AMI, OVF, or Docker-based installations via the UI, the **Upgrade Now** button on the Home dashboard will upgrade your PMM version to the latest PMM 2 release. This upgrade will provide access to all the new features and bug fixes available in this version. However, the PMM base operating system will remain on EL7 (CentOS 7) and will not be upgraded to EL9 (Oracle Linux 9). This means that certain CVEs related to the base operating system will remain unpatched.

While we work on providing automated migration paths for these deployment types, you have the option to manually upgrade your PMM installation to take advantage of the updated Oracle Linux 9 base operating system and associated security enhancements.

To manually upgrade your PMM installation to the new Oracle Linux 9 base, check out the instructions in the PMM is getting a modernized enterprise-grade foundation blog post.

GRAFANA UPGRADE

PMM now uses Grafana to 9.2.20, which fixes some critical vulnerabilities. For more information, see Grafana's release blog.

EXPORTERS

Updated the mysqld_exporter, node_exporter, postgres_exporter, and mongo_exporter to fix the CVEs.

CLICKHOUSE PLUGIN

We have migrated from the Altinity Clickhouse datasource to the Grafana Clickhouse datasource due to the vulnerabilities discovered.

Due to this migration, we have also updated our experimental Environment Overview Dashboard.

> **Important**
>
> Though we have removed the datasource from PMM, we are still reusing the same name (ClickHouse) to keep the changes minimal. Thus, if you are using the old datasource in your dashboards, ensure that you upgrade the underlying queries that will pull the data from the new datasource.

Example

The following example shows the difference in the query before and after migration.

Before migration:

```
// Altinity ClickHouse datasource
SELECT $timeSeries as t,
    sum(m_query_time_sum)/sum(m_query_time_cnt) as QLatency,
    environment
FROM $table
WHERE $timeFilter and environment != ''
```


Percona LLC and/or its affiliates ©2024

```
GROUP BY t,environment
ORDER BY t
```

After migration:

```
// Grafana ClickHouse datasource
SELECT period_start as t,
    sum(m_query_time_sum)/sum(m_query_time_cnt) as QLatency,
    environment
FROM pmm."metrics"
WHERE $__timeFilter(period_start) AND environment != ''
GROUP BY t, environment
ORDER BY t
```

Altinity data sources used macros like `$timeSeries`, `$timeFilter`, and `$table`, which are no longer present in the Grafana datasource. As a result, the query needs to be rewritten to use the new macros instead of the old ones. You will notice that in the example above, the macro `$timeFilter` has changed to `$__timeFilter`, and now it requires a parameter, which should be a column of type **datetime**.

Depending on which macros you have used, your migration effort will vary. However, we recommend checking the plugin documentation to apply the new syntax correctly, including macros and template variables.

Improved nodes view on the Inventory page

In our continuing endeavor to enhance user experience, we further improved the **Inventory** page by improving the **Nodes** tab. With this change, you can get more context on your inventory objects, such as nodes, and more actionable information for exploring and resolving possible issues.

Percona LLC and/or its affiliates ©2024

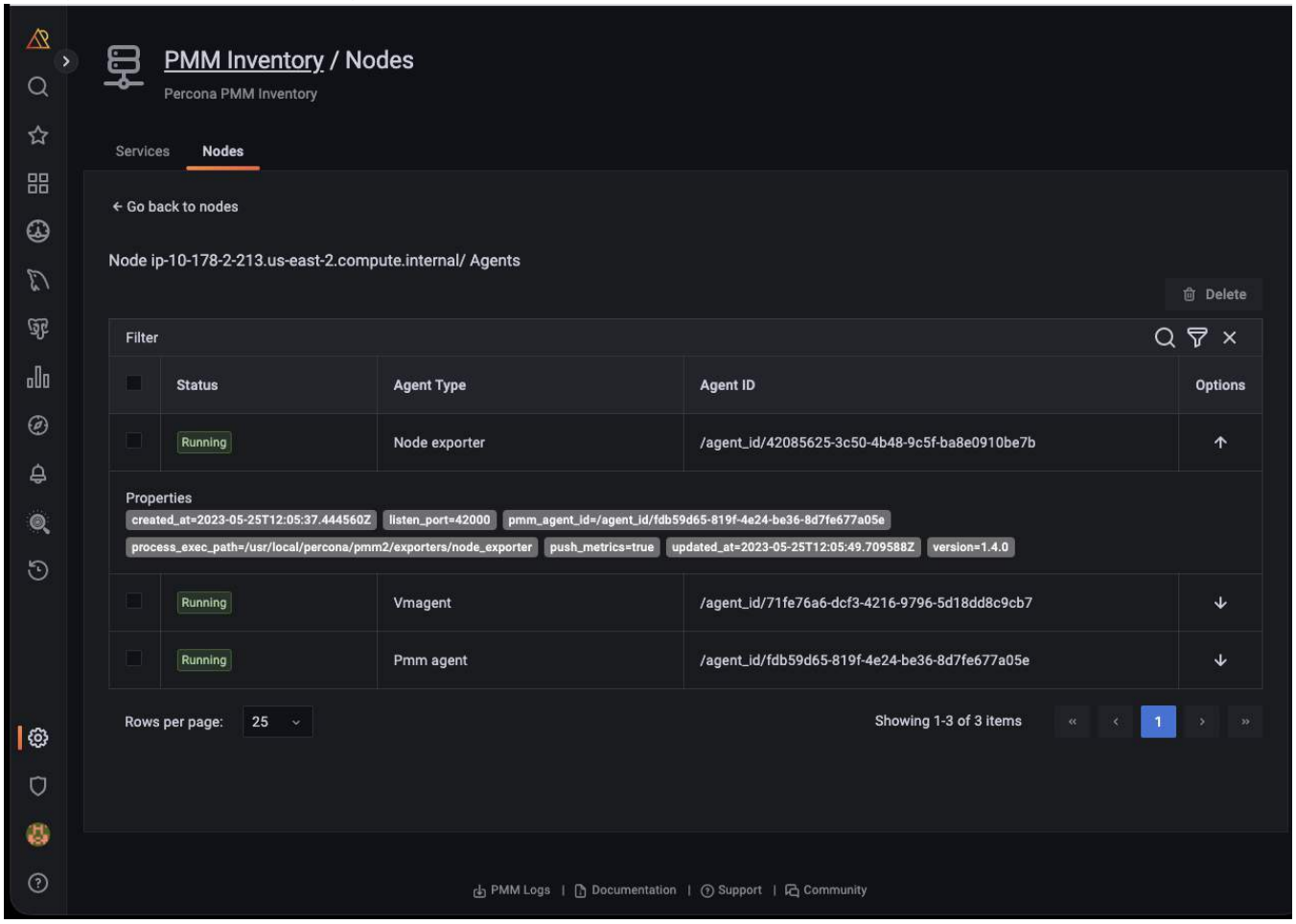
Here is the change as part of PMM 2.38.0:

Percona LLC and/or its affiliates ©2024

- Check the number of agents running on any particular node. When you click on any node, the UI navigates to the view of agents, which is filtered to display only agents related to that specific node.

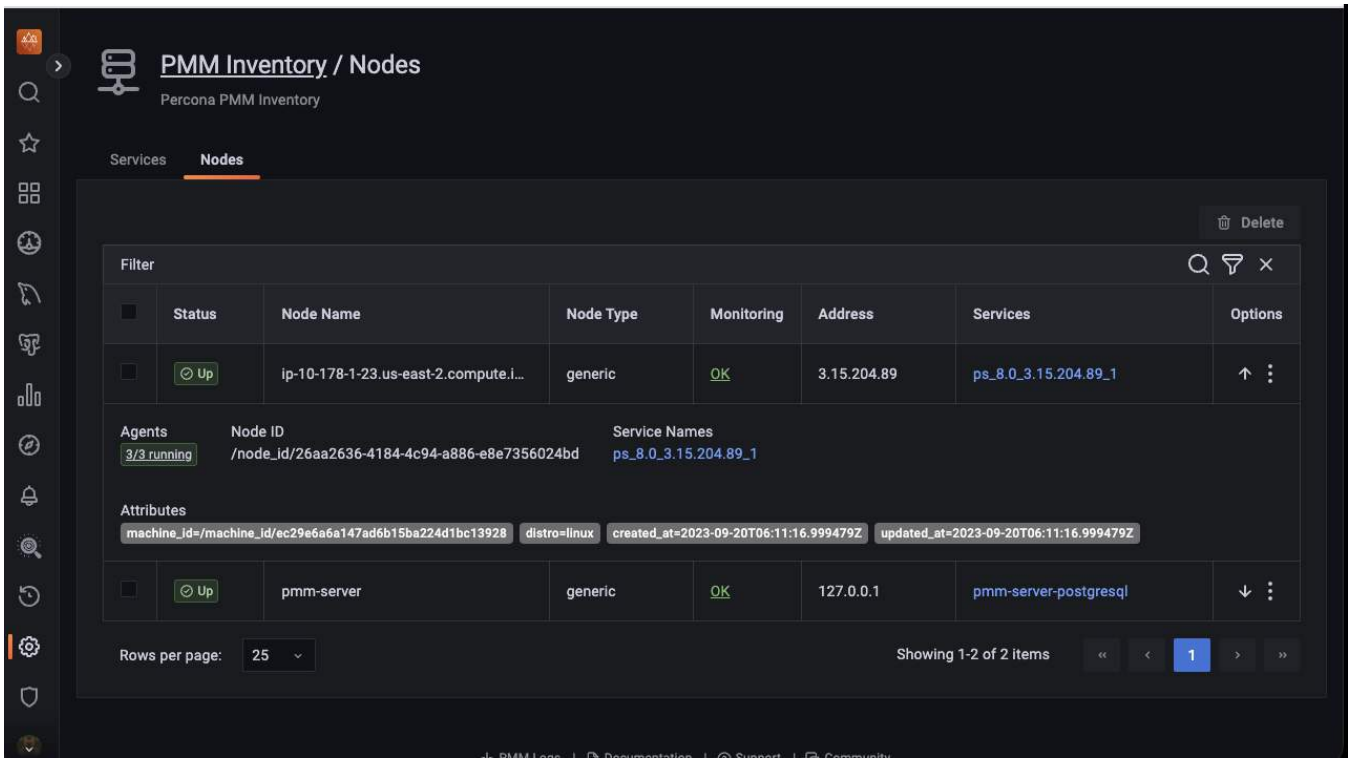
  To see the details of the agents running:

  On the **Nodes** tab, under the **Monitoring** column, click **OK** or **Failed**, depending on the status of the node that you have selected. A page that provides the user with crucial information regarding the total number of agents deployed on that node is displayed.



- View comprehensive information about each agent, including key attributes such as node type, IP address, and associated services. This gives you a complete overview of the nodes at a glance.

Percona LLC and/or its affiliates ©2024



- Check the health status of the agents to see if they are running or have failed.

Backup enhancements

BETTER FOLDER MANAGEMENT FOR ENHANCED PBM COMPATIBILITY

We've added a new **Folder** field to the Create Backup pages. Use this to set a specific target directory within the selected local or S3-compatible location for the backup.

Organizing backups in folders makes grouping PBM backups for clusters easier, and it improves PMM-PBM (Percona Backup for MongoDB) integration workflows.

The **Folder** field is automatically populated with the value of the cluster label. You can change this default folder from PMM's Advanced Settings, but make sure you understand how your custom folder will impact PBM integration workflows.

SHARDED CLUSTER MONGODB BACKUP SUPPORT

This release introduces support for working with backups for sharded clusters. PMM handles the backup process end-to-end, but restoring such artifacts is currently possible only via the CLI, using Percona Backup for MongoDB.

For information on restoring sharded backups, check the PBM documentation.

SQL comments support in QAN

> **Important/Caution**
>
> This feature is still in Technical Preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

You can now collect labels from the comments placed in SQL queries. You can tag the query by key value in the query comment. This helps in query management and optimization. This feature is currently supported for MySQL and PostgreSQL.

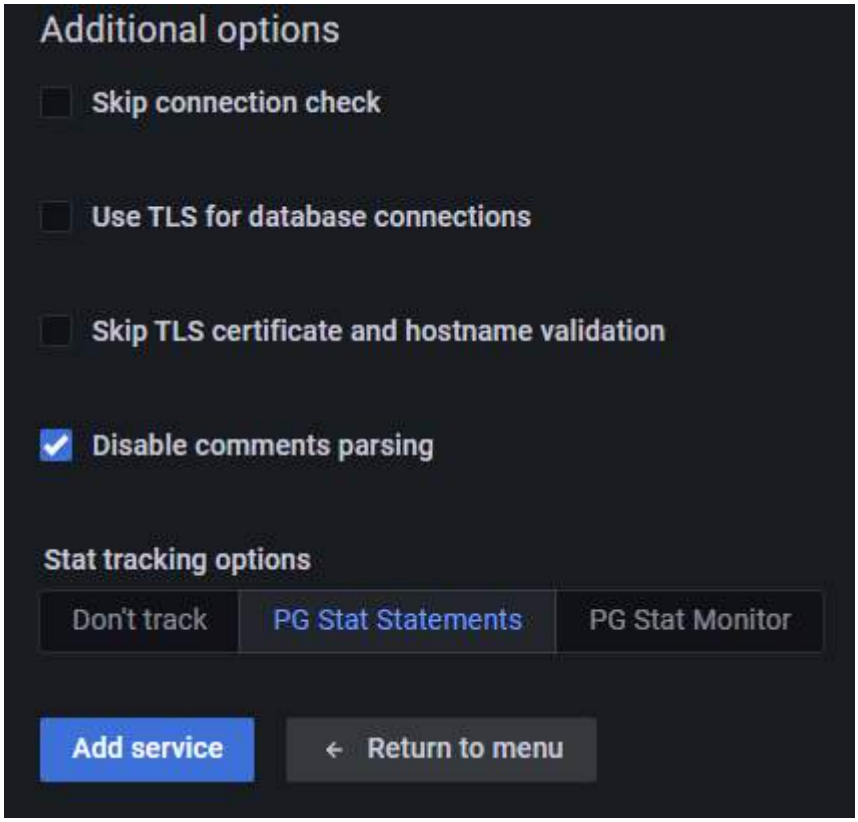
Percona LLC and/or its affiliates ©2024

For example:

```
SELECT * /* cluster='east' */ FROM city;
```

You can enable this feature as follows:

- CLI - Pass the parameter `comments-parsing`
- UI - It is enabled by default. To disable, uncheck the **Disable comments parsing** checkbox in the **Additional Options** section. This is supported for PostgreSQL and MySQL.



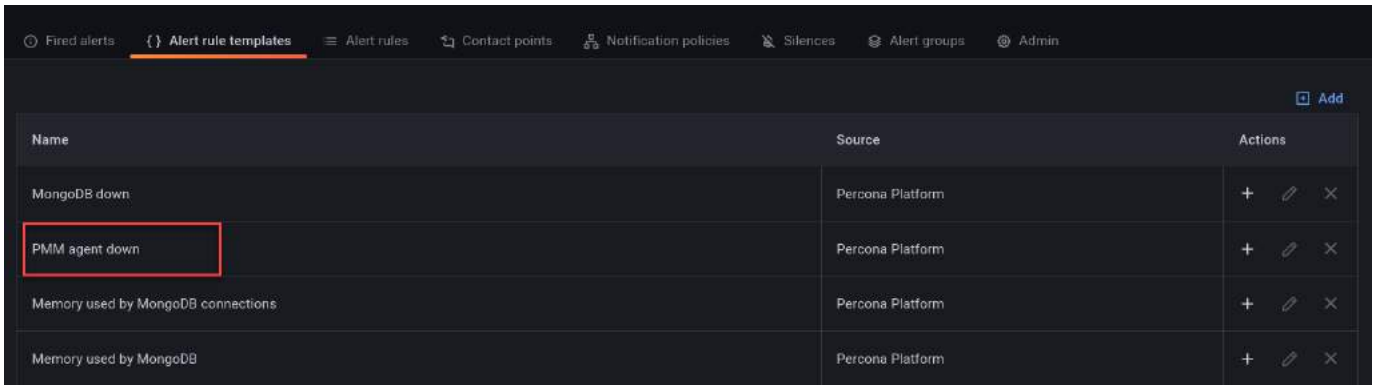
New PMM Agent Down alert template

We've added a new default alert template to Percona Alerting.

The new **PMM Agent down** alert template monitors your Node status to notify if the Agent is down. Before using this template:

- ensure the PMM Agent is healthy and present on the Database Node.
- check that PMM Client is on version 2.38 version or later.

Percona LLC and/or its affiliates ©2024



Grafana upgrade

PMM now uses Grafana v9.2.20, which includes fixes for some critical vulnerabilities. For more information, see Grafana's release blog.

9.10.2 New Features

- PMM-2852 - [QAN]: Starting with PMM 2.38.0, MySQL and PostgreSQL supports collection of labels from the comments placed in the SQL queries.
- PMM-11961 - [Inventory]: Starting with PMM 2.38.0, we have enhanced the **Inventory** page by improving the **Nodes** tab. With this change, you can get more context on your inventory objects, such as node, and more actionable information for exploring and resolving possible issues.
- PMM-11962 - [Inventory]: Starting with PMM 2.38.0, you can filter the nodes and services on the **Inventory** page using the parameters such as ID, names, status, etc., instead of scrolling through an exhaustive list of nodes or services.
- PMM-12087 - [Inventory]: Added tooltips to explain the meaning of the different service statuses in the Inventory view.

9.10.3 Improvements

- PMM-9367 - [Backups]: Added support for working with backups for sharded clusters.
- PMM-11250 - [Backup]: Added option to specify target directory for the selected backup location to improve PMM-PBM integration workflows.
- PMM-12038 - [Backup]: Extended support for MongoDB backups to cover all authentication mechanisms available for MongoDB (including LDAP, x509 certs).
- PMM-9544 - [Alerting]: Added new default alert template to Node status and notify if the Agent is down.
- PMM-12114 - Upgraded Grafana to 9.2.18 to fix to CVEs.
- PMM-12136 - We have migrated our base operating system on which we build our base image from CentOS 7 to Oracle Linux 9 to fix the CVEs.

9.10.4 Bugs Fixed

- PMM-11578 - [Alerting]: The **Last notified** column under **Alerting > Fired Alerts** has been renamed to **Last triggered**. This better illustrates the fact that triggered alerts may not always generate a notification. This can happen when the contact points are not correctly configured. In addition, the **Suppressed** status in the **State** column has been renamed to **Silenced** to keep consistency with how this status is displayed on other **Alerting** pages.
- PMM-12067 - [QAN]: Fixed vulnerability in Vitess package.

Percona LLC and/or its affiliates ©2024

- PMM-12105 - [QAN]: Fixed an issue where QAN was broken due to wrong pgsm_query_ids.
- PMM-10063 - [QAN]: The QAN filter section could display the available list of services/nodes. But the load time was too much, with the UI being unresponsive. This issue has been resolved now. The entire list is loaded with the count displayed on the upper right side.
- PMM-11386 - [QAN]: MongoDB QAN query time metrics was showing the wrong unit. It should be ms. The issue has been resolved now.
- PMM-12024 - [QAN]: Fixed an issue where a QAN agent couldn't start due to an expected return value type.
- PMM-12091 - [QAN]: The tooltip for QAN > MongoDB > Query Details was displayed in seconds instead of decimals. The issue has been fixed now.
- PMM-9844 - [QAN]: Data for a large number of partitions or possible indexes was shown in the QAN Details tab in one scrolling column making it practically impossible to read. The issue has been resolved now.
- PMM-11938 - [QAN]: Fixed the problem when PMM tried to create `pg_stat_monitor_settings` view all over again.
- PMM-11645 - Fixed mysql service detection when this is not loaded in systemd.
- PMM-9541 - PostgreSQL exporter now works as expected with `--socket`.
- PMM-10799 - Now in the **EXPLAIN/TABLE** tab you can get results even for queries with table alias.
- PMM-11692 - On the MySQL Instances Overview dashboard, the `Top MySQL Active Client Threads` panel color was misleading. The issue has been resolved now.
- PMM-11715 - CPU Utilization Graph for RDS instances was not matching what Cloudwatch reported. The issue has been resolved now.
- PMM-11950 - Non-admin roles are calling restricted endpoints after login or page refresh.
- PMM-12085 - Fixed an issue where SSH access to AMI and OVF images for Oracle Linux 9-based images did not work. The issue has been resolved now.
- PMM-12118 - Addressed the Grafana ClickHouse DS plugin CVEs.
- PMM-12119 - When a user successfully adds an RDS service to PMM, the RDS exporter's status does not change from "UNKNOWN" (the initial status of all exporters) to "RUNNING." This issue has been resolved.
- PMM-9224 - Data from Arbiter nodes is now correctly displayed on the **MongoDB ReplSet Summary** dashboard.

9.10.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-04-18

9.11 Percona Monitoring and Management 2.37.1

Release date:	**Jun 5, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

Percona LLC and/or its affiliates ©2024

Important

We recommend using the latest version of PMM. This ensures that you have access to the latest PMM features and that your environment runs on the latest version of the underlying components, such as VictoriaMetrics, with all the bug fixes in place.

9.11.1 Release Highlights

- We have identified and fixed CVE-2023-34409 in PMM 2.37.1:

  PMM-12182: PMM authentication bypass vulnerability

  Workaround

  If you are unable to update PMM you can resolve this issue as follows:

  a. Make changes to the NGINX configuration on the running PMM instance. To do so, create a Bash script with the code from this script on GitHub.

  b. Apply the code using this `docker` command on a server running the PMM Docker container (as root or using sudo):

  ```
  docker exec -it pmm-server bash -c 'curl -fsSL https://raw.githubusercontent.com/percona/pmm/main/scripts/authfix.sh  | /bin/bash '
  ```

  c. If you are running PMM via a virtual appliance (OVF or AMI), use SSH to shell into the PMM server and run this command:

  ```
  curl -fsSL https://raw.githubusercontent.com/percona/pmm/main/scripts/authfix.sh  | /bin/bash
  ```

  For more details see, blogpost.

- An Enterprise Linux 9 (el9) version of PMM was released through oversight. This was intended to be a technical preview. There are several known issues with this el9-based version. Thus, we do not recommend running it in production until the official GA announcement.

  The images from the `percona` Docker repository have been removed. However, those who want to test them can locate them in the perconalab Docker repository.

9.11.2 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-06-15

9.12 Percona Monitoring and Management 2.37.0

Release date:	**May 02, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

Percona LLC and/or its affiliates ©2024

9.12.1 Release Highlights

Improved Inventory view

This release of PMM starts the series of enhancements that will improve the way you work with Services and Nodes in PMM. With this first set of changes, we're revamping the **Inventory** page to give you more context on your inventory objects, and more actionable information for exploring and fixing possible issues.

Here are all the specific changes:

- Improved design of the **Inventory** page for a clearer presentation of all the objects registered by PMM.
- Services now show their database status, which is based on the monitoring metrics. This enables you to quickly check the status of the databases when you are working with the inventory.
- New **Monitoring** column showing the live status for the monitoring Agents assigned to a Service. This enables you to quickly spot any possible issues with the Agents.
- Quick access to the Service Summary **Dashboard** and to **QueryAnalytics** results so that you can easily explore important related information.
- New **Details** section grouping together all the relevant parameters and labels for a Service. To keep things clean, this section is collapsed by default, so that you can view extra information only when you need it.

Here's what's coming:

Building up on these changes, subsequent releases will further improve PMM Inventory with:

- More useful features to filter and better connect Services and Nodes.
- Services organized in clusters so that you can capture the bigger picture, instead of working at of individual Service-level.

New Experimental Dashboards for Kubernetes Monitoring

> **Important**
>
> These experimental dashboards are subject to change. It is recommended to use these dashboards for testing purposes only.

We are pleased to announce the release of PMM 2.37, which introduces three new experimental dashboards:

- Kubernetes Cluster Summary
- Kubernetes Pods Status
- Kubernetes Volumes

These dashboards are designed to provide valuable insights into the status and performance of your Kubernetes cluster, pods, and volumes, helping you to identify and troubleshoot issues quickly and easily.

We welcome your feedback as we continue to enhance PMM with these new dashboards.

Experimental Kubernetes Cluster Summary Dashboard

Percona LLC and/or its affiliates ©2024

Kubernetes Cluster Summary provides a comprehensive overview of your Kubernetes cluster, including:

- Components
- Node
- Pod
- PVC status
- CPU
- Memory overview, and more.

This dashboard displays all workloads running in the cluster, enabling you to take action and optimize its performance.

Experimental Kubernetes Pods Status Dashboard

Kubernetes Pods Status dashboard provides detailed information about the state and performance of your pods, including CPU, Memory, and Network metrics.

This dashboard can help you quickly pinpoint any issues affecting your pods and ensure they continue to operate smoothly.

Experimental Kubernetes Volume Dashboard

Kubernetes Volumes dashboard provides insights into your Kubernetes volumes, including capacity and usage, in real time. With this dashboard, you can easily monitor the performance and usage of your volumes and take proactive measures to ensure their performance.

Refer to the documentation to learn more about these new experimental dashboards and how to use them.

Here are the steps to create a new folder and move all experimental dashboards to the new folder for quick access and internal use:

> **Note**
>
> You should have at least an *Editor* role to create a new folder and move all experimental dashboards.

1. Navigate to the **Main** menu and hover on the *Dashboards* icon.
2. Click **New folder**.
3. Provide a name for your folder, and then select **Create**.
4. Navigate to *Dashboards* from the **Main** menu and click **Browse**.
5. Select the dashboard that you want to move and click **Move**.
6. On the **Choose Dashboard** dialogue box, from the dropdown under **Folder** option, choose the folder where you want to move the dashboard.
7. To apply your changes, select **Save Dashboard**.

9.12.2 Components upgrade

VictoriaMetrics has been upgraded to version [1.89.1].

Percona LLC and/or its affiliates ©2024

9.12.3 New Features

- PMM-10913 - Starting with PMM 2.37.0, you can use an external PostgreSQL server as data storage for PMM. Thus, even if PMM fails, your data will be stored in an external source, reducing the risk associated with data loss. Furthermore, clustered PostgreSQL instances also enhance performance.
- PMM-11281 - We have written a query for the postgres exporter to deal with lock conflicts in databases that block several transactions, thus impacting performance. With this query, you can find the PID of the blocking session and how many sessions it blocked. The proactive approach enables customers to address non-responsive databases quickly by understanding the underlying causes.
- PMM-11384 and PMM-11834 - PMM 2.37.0 now supports PSMDB operator versions 1.13 and 1.14.
- PMM-11438 - Starting with PMM 2.37.0, PMM administrators can disable the internal PostgreSQL server, which helps to fine-tune the server.
- PMM-11439 - Starting with PMM 2.37.0, a PMM administrator can now view metrics from an external PostgreSQL server.

9.12.4 Improvements

- PMM-11412 - We have enhanced our dashboards to be more user-oriented, displaying the DB cluster's status along with its components, such as metrics of stateful, pods, and so on. Also, you can apply filters based on DB name or DB type.
- PMM-11832 - When hovering over the **Backup** option on the main menu, PMM now displays the subcategories for easier navigation.
- PMM-11522 - With the newly added support for Clickhouse as a datasource for Advisors, Advisors can now use data produced by Query Analytics (QAN) to run checks on databases.
- PMM-11544 - Improved functionality for EXPLAIN query in QAN.
- PMM-11926 - Updated the format of the **Upgrade to a Premium plan** URL on the **Advisor Insights** page.
- PMM-11078 - Following the replacement of Integrated Alerting with Percona Alerting in PMM 2.311, we have now phased out all the APIs related to this deprecated feature.

9.12.5 Bugs Fixed

- PMM-8690 - Fixed an issue where the Replica set lag displayed incorrect values on the MongoDB ReplSet Summary dashboard for an ARBITER node.
- PMM-9775 - QAN: PMM failed to display the set timezone for QAN despite it being set to UTC.
- PMM-10687 - DBaaS: Fixed an issue where DB cluster creation failed when using an unreleased version of pmm-server.
- PMM-11121 - DBaaS: Fixed an issue where the list of DB clusters did not load if one of the K8s clusters was unresponsive.
- PMM-11226 - Fixed an issue where different operator versions and names were displayed for each K8s cluster.
- PMM-11313 - Rows Fetched/Read and Users by Rows Read dashboards are not working in the MySQL User Details dashboard
- PMM-11555 - For thousands of client connections for ProxySQL, the graph for Active Frontend connections did not render due to sluggish performance. Furthermore, this could affect the rendering and usage of the ProxySQL Instance summary dashboard. The issue has been resolved now.
- PMM-11829 - PMM Agent does not work with PG 12 and PGSM 2.0. This issue has been resolved now.
- PMM-11844 - Fixed an issue where a user failed to add PSMDB with pmm-admin using native LDAP authentication and TLS.

Percona LLC and/or its affiliates ©2024

- PMM-11862 - Fixed inconsistent naming of the Advisors feature across PMM UI.
- PMM-11875 - Fixed an issue where the users could not select and compare multiple nodes on the Summary dashboard.
- PMM-11904 - Index Size and Concurrent Inserts panels were broken on the Insight / VictoriaMetrics dashboard after updating to version 1.89.1.
- PMM-10795 - Node summary was not visible under MongoDB ReplSet Summary when the node name contained a dot (.) in the name.
- PMM-11465 - Fixed problem with empty output for EXPLAIN in QAN.
- PMM-11729 - Fixed syntax issues with placeholders for EXPLAIN in QAN.
- PMM-11849 - Fixed issue that affected Advisors, where Actions and Jobs would always be executed on the primary node in a MongoDB cluster.
- PMM-11934 - Product tour is now correctly displaying the Advisors information.

9.12.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-06-08

9.13 Percona Monitoring and Management 2.36.0

Release date:	**Mar 23, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

9.13.1 Release Highlights

pg_stat_monitor 2.0 support in QAN

We are excited to announce PMM 2.36 now supports pg_stat_monitor 2.0 (PGSM 2.0) in QAN, a powerful PostgreSQL query performance monitoring tool. By downloading this update, you will have access to the latest improvements and fixes covered by PGSM2, including:

- Improved internal architecture that results in fewer lock acquisitions and increases performance by approximately 20%.
- Support for PostgreSQL 15
- Enhanced consistency with `pg_stat_statements` so that the `pg_stat_monitor` view has identical column names, columns, and data types as `pg_stat_statements` for every major version of PostgreSQL from versions 11 to 15.
- A bucket status indication (done vs. current) eliminates the need for the tool to evaluate bucket status and facilitates accurate data display.
- The generation of a unique ID for a query makes it easier to monitor query planning, execution, and performance regardless of version, database, user, or schema.

Percona LLC and/or its affiliates ©2024

For more information on the enhancements in PGSM2, refer to the full list.

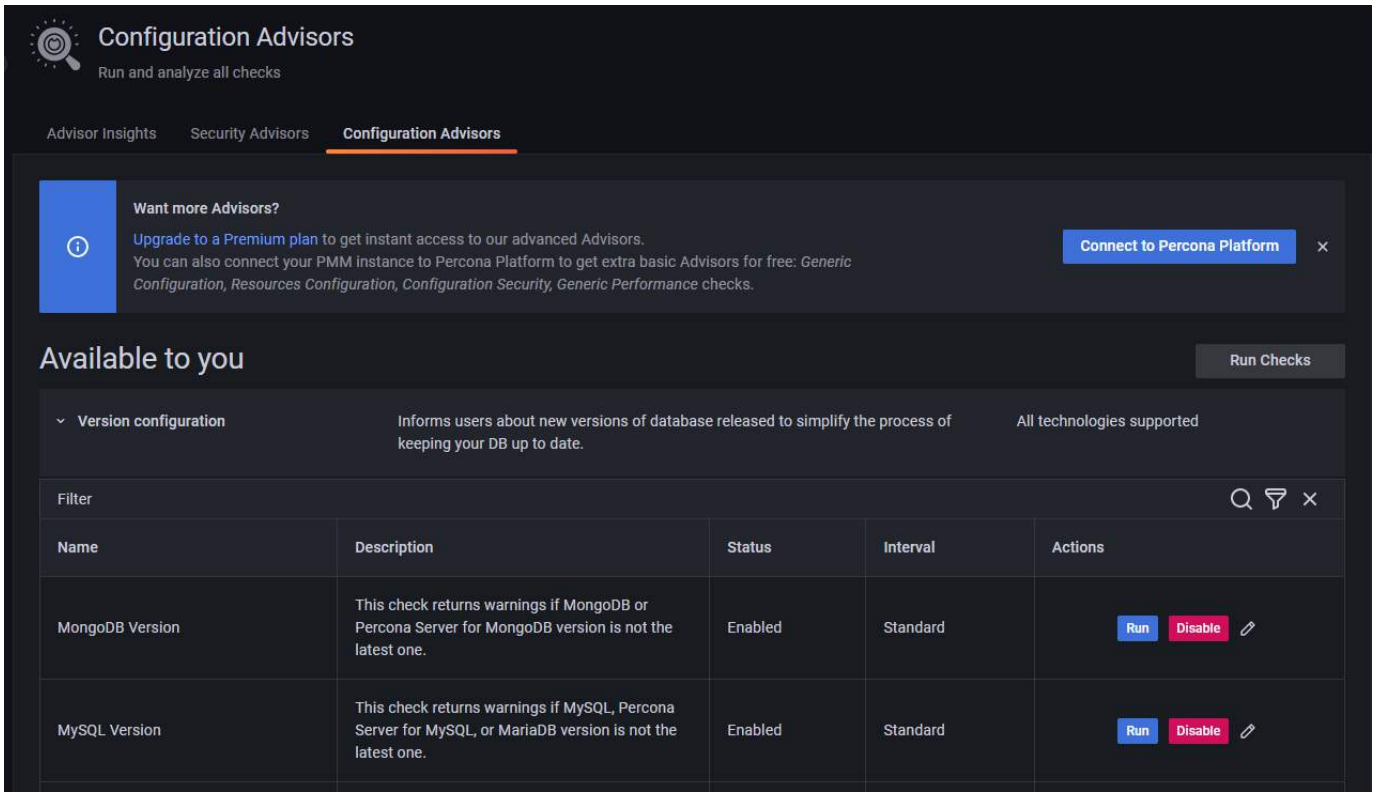
Important

PMM 2.36 and PGSM 2.0 now support PG 13, 14, and 15. Support for older versions (PG 11 and 12) will be introduced in PMM 2.37.

Revamped Advisors page

We've improved the structure of the **Advisors** page to group Advisors by categories and clearly show the Advisors available for your current subscription.

With this change, we have renamed all the tabs on the page, and added an option to easily integrate with Percona Platform for extra Advisors for free:



PostgreSQL Advisors executed against more databases (DBs)

To give you additional insights on your PostgreSQL DBs, PostgreSQL Advisors now try to perform checks on all the databases accessible with the credentials used during pmm-client setup. In most cases, these are the databases available for the PMM database account user. For more information, see the Create a database account for PMM topic in the PMM online Help."

Backup Management enabled by default

Since the PMM 2.32 release, we have continuously improved the Backup Management functionality in terms of backup locations, restore options and user experience.

As a mature and generally useful feature, this option is now enabled by default for easier access to your backups and restore artifacts.

Percona LLC and/or its affiliates ©2024

Note

- Upgrading to PMM will automatically enable this feature for existing PMM instances. You can disable it at any time from your PMM dashboard on the **Advanced Settings** page or using the Docker environment variable `DISABLE_BACKUP_MANAGEMENT=1`.
- While we have GA'ed Backup Management for MongoDB since PMM 2.32, Backup Management for MySQL still in technical preview, and, therefore, subject to change. For more information, see the Percona Release Lifecycle.

9.13.2 Components upgrade

Grafana has been upgraded to version 9.2.13.

9.13.3 New Features

- PMM-9474 - PostgreSQL Advisors are now executed against all the available databases (DBs), not only the one connected to PMM
- PMM-11363 - Starting with PMM 2.36.0, you no longer need to unassign a role before deleting it. If you try to delete a role assigned to a user, you will see a dropdown with replacement roles. As a result, the selected role will be assigned to the user.

9.13.4 Improvements

- PMM-11640 - [Backup Management]: The Backup Management feature is now enabled by deault forf easier access to your backups and restore artifacts.
- PMM-11182 - [Platform integration]: Renamed the **Disconnect** button on the confirmation message that pop-ups when you are terminating Percona Platform connections as an Administrator. The button is now called **Force disconnect** to better differentiate between situations where you are disconnecting using a Percona Platform Account.

  Force-disconnecting only clears the connection in PMM without automatically removing the disconnected servers from Percona Platform. We recommend that you always disconnect while logged in with your Percona Account. Otherwise, make sure to manually remove the force-disconnected servers from your list of PMM instances in Percona Platform.
- PMM-11381 - Starting with PMM 2.36.0, you can scrape Kubelet metrics with DBaaS or VictoriaMetrics operator. Using Kubelet metrics, you can gather volume-related information.
- PMM-11566 - You now have Admin Access to PMM Demo instances so that you can experiment with all the features of PMM.
- PMM-11699 - Improved documentation and descriptions for Advisor checks.
- PMM-10974 - Usage of pg_stat_monitor by default.

9.13.5 Bugs Fixed

- PMM-11291 - [DBaaS]: There was a difference in the shape and color of the DB Cluster Status box compared to the Kubernetes Cluster status box. The issue has been resolved now.
- PMM-11296 - [DBaaS]: Fixed an issue where the DB Cluster creation page did not load when DBaaS was disabled.
- PMM-11358 - [DBaaS]: Connection to the DB cluster displayed a loader when the DB cluster was suspended/ paused. The issue has been resolved now.
- PMM-11613 - [DBaaS]: Backups could not be created when uppercase letters were used in the storage name. The issue has been resolved now.
- PMM-11568 - [DBaaS]: Disabled the selection of MySQL versions less than 8.0.27.

Percona LLC and/or its affiliates ©2024

- PMM-11766 - [DBaaS]: The status of an unavailable cluster was displayed as Active. When this cluster was unregistered, an error was thrown. The issue has been resolved now.
- PMM-11428 - [DBaaS]: Fixed an issue where editing DB clusters was broken.
- PMM-10254 - [Dashboards]: On the MySQL/Mongo dashboards, the Memory Available graph showed incorrect values. The issue has been resolved now.
- PMM-11209 - [Backup Management]: Improved error handling for partially successful restores. Restore jobs that finished successfully on some nodes and failed on others are no longer locked in a **Pending** state. Instead, PMM now shows a **PartlyDone** error and populates the logs with specific details on the restore status for the replica set and nodesup restores.
- PMM-10606 - Fixed an issue for the incorrect link for node name on the mysql instance summary dashboard.
- PMM-11001 - Fixed an error that occurred in certain scenarios when expanding alert rules nested in folders.
- PMM-11004 - Fixed issue that ignored the database name provided when adding PG server to PMM.
- PMM-11408 - Navigation from the New Home dashboard panel to certain dashboards is broken. The issue has been resolved now.
- PMM-11421 - Improved error message that is displayed when trying to connect PMM to Percona Platform over HTTP Proxies: “PMM Platform connection does not support proxy”.
- PMM-11526 - On the Summary dashboard, the error message exit status 1 has been replaced with a more meaningful message.
- PMM-11588 - Fixed an issue where the database instance count on the OS Overview dashboard was incorrect.
- PMM-11636 - The Grafana Admin page crashed when filtering users. The issue has been resolved now.
- PMM-11654 - There was no data displayed on the PXC Galera replication latency dashboard. The issue has been resolved now.
- PMM-11664 - Fixed an issue for the environment filter missing from the OS dashboards.
- PMM-11717 - Mysql exporter agent stopped working after upgrading the server to PMM 2.36.0. The issue has been resolved now.

9.13.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-04-06

9.14 Percona Monitoring and Management 2.35.0

Release date:	**Feb 23, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open-source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

Percona LLC and/or its affiliates ©2024

9.14.1 Release Highlights

Access Control: Limit access to metrics

> **Important/Caution**
>
> Disclaimer: PMM Access Control is still in Technical Preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

We are pleased to announce that PMM 2.35.0 marks the first step towards introducing PMM Access Control. Label-based access control in PMM allows you to manage who has access to metrics based on labels.

By creating roles, you can specify which data can be queried based on specific label criteria, for instance, allowing the QA team to view data related to test environments. Using this feature, you can associate multiple labels with a role, ensuring only data from series that match your defined labels is returned.

This feature provides a standardized method for granting, changing, and revoking access to metrics, giving you a powerful and flexible way to manage access to metrics and ensure secure access to the information you need. For more information, see documentation.

GA: Deployment of PMM in Kubernetes with Helm Chart

We are happy to announce the General Availability of PMM deployment in Kubernetes with the Helm chart. PMM deployment via Helm chart in Kubernetes has been available as Tech Preview since PMM v2.29, but now we're delighted to offer it to users as a fully supported feature. For more information, see documentation.

DBaaS: Add a backup schedule while creating DB clusters

Starting with PMM 2.35.0, you can add a backup schedule while creating DB clusters in DBaaS. This feature is a fusion of backup management and DBaaS in PMM. Currently, DBaaS only supports scheduled backups, which can only be enabled when a database cluster is created.

DBaaS **does not** support on-demand backups. For more information, see documentation.

DBaaS: Restore cluster from backup stored on S3

You can now create a DBaaS cluster from a backup stored on S3. You can use backups taken from existing cluster and use them to spin-up a new database cluster from this backup. Attempts to use it to restore a previous version of the cluster will result in downtime since the system will treat it as a creation of a new cluster.

DBaaS **does not** support on-demand restore. For more information, see documentation.

More pre-run checks for MongoDB backups

With this release, PMM runs additional verifications to ensure infrastructure compliance and prevent backup execution errors:

TOOL COMPATIBILITY CHECK

Before initiating backups for MongoDB, PMM now examines the current MongoDB node to ensure that the connected PMM Agents have the tools to execute and finalize the backup.

If the requirements are not met, PMM reports what tools or tool versions are missing, similarly to the existing prerequisites check for MySQL backups.

Percona LLC and/or its affiliates ©2024

OVERLAPPING JOBS CHECK

To avoid conflicting jobs, PMM proactively disables the creation of Point-in-time Recovery (PITR), or snapshot backups, when other backups are scheduled for running.

Similarly, creating PITR backups for different services of the same cluster is now automatically restricted.

CLUSTER MEMBERSHIP CHECK

To avoid execution errors, PMM no longer allows scheduling backups on MongoDB Services that are not managed as a cluster.

You can check which of your current Services are not members of a cluster on the **PMM Inventory > Services** page. Services that do not specify a cluster name in the **Other Details** column should be removed and re-added using the following set of commands:

```
pmm-admin add mongodb
username=pmm_mongodb --password=password
query-source=profiler --cluster=mycluster
```

Backup restore logs accessible from PMM

With this release, we are adding the option to view the logs for restores from a MongoDB database backup. This complements the existing option to view the logs for the backup process and makes it easier to gather data for compliance, auditing, and troubleshooting purposes.

Instead of having to look through PBM log files, you can now access the **Log** option in PMM, which is now automatically displayed next to each MongoDB backup restore entry on the **Backup > Restores** page.

Components Upgrade

MongoDB exporter has been upgraded to 0.37.0.

9.14.2 New Features

- PMM-10325 - Starting with PMM 2.35.0, for enhanced security, you can change the admin password even after `pmm-agent` is registered on the PMM Server.
- PMM-10550 - [Backup Management]: PMM now automatically displays a `Logs` option next to each MongoDB backup restore entry on the **Backup > Restores** page.
- PMM-11302 - [DBaaS]: Starting with PMM 2.35.0, you can add a backup schedule while creating DB clusters in DBaaS. You can also restore the cluster from backup stored on S3.
- PMM-10962 - Starting with PMM 2.35.0, we have introduced Access Control. Roles are a vital part of Access Control that provide the users with access to specified metrics. You can create, edit, and delete user roles. The roles can be assigned to the users with the **Configuration > Users** tab. For more information, see documentation.

9.14.3 Improvements

- PMM-10471 - [Backup Management]: PMM now automatically checks the current MongoDB node to ensure that the connected PMM Agents have the tools to execute and finalize the backup.
- PMM-11113 - Enhanced user experience for the **Percona Platform** page under **Configuration > Settings**. The new version of this page highlights the advantages of connecting your PMM instance to Percona Platform and makes it easier to generate and use a connection token.

Percona LLC and/or its affiliates ©2024

9.14.4 Bugs Fixed

- PMM-9446 - Fixed an issue with the MySQL/MySQL Details dashboard, where the users' names were overlapping.
- PMM-10016 - [Advisors]: The list of checks in the **Failed Checks** table would randomly rearrange after clicking on any of the checks in the list. This issue is now resolved.
- PMM-10109 - [DBaaS]: Fixed an issue that prevented the creation of DB clusters when performing a `List` query.
- PMM-11290 - [DBaaS] Fixed an issue where the Advance option in Edit DB cluster disabled the Edit/Save option.
- PMM-11474 - [DBaaS]: An internal server error was thrown on the UI while upgrading PMM to 2.34.0, and DB clusters were not visible in the UI. The issue has been resolved now.
- PMM-11062 - Fixed an issue where the Check update button was not visible.
- PMM-11116 - Despite PMM-gent being connected, the status of all other agents was `Unknown`. The issue has been resolved now.
- PMM-11154 - Fixed an issue where all agents except `pmm-agent` display NULL values in `version` and `runs_on_node_id` fields in the `agents` table of internal PostgreSQL database.
- PMM-11186 - Check that service is a member of a cluster before MongoDB backup/restore
- PMM-11274 - "Postgresql exporter error" was thrown for all nodes after upgrading from 2.29.0 to 2.33.0. The issue has been resolved now.
- PMM-11298 - While monitoring the performance of VictoriaMetrics with PMM 2.33.0, the CPU usage shown on Node Summary or VictoriaMetrics dashboard overview panel did not match the value shown on the system Level Metrics panel. The issue has been resolved now.
- PMM-11357 - [Percona Alerting]: Fixed an issue where the name of an alert rule was not inherited correctly from the alert template.
- PMM-11399 - [Percona Alerting]: The pop-up message that confirmed saving an alert rule would not display the name of the rule correctly. This issue is now resolved.
- PMM-11517 - [Backup Management]: Fixed a PBM synchronization issue that made PMM unable to restore local physical backups after the first attempt.

9.14.5 Coming Soon

Access Management

The following features/use cases will be covered as part of the various stages of PMM Access Management:

Role-based access control (RBAC), including access limitations to the features like **Alerting** and **Backup** with read-edit-delete permissions.

PMM High Availability

PMM High Availability (HA) feature will be available soon!

The planned phases will be launched soon.

9.14.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2023-03-15

9.15 Percona Monitoring and Management 2.34.0

Release date:	**January 30, 2023**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open-source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

9.15.1 Release Highlights

Various UX enhancements to Backup Management

With this release, we are updating the Backup and Restore functionality with key usability enhancements, including:

- Filtering on services at backup creation time. This makes it considerably easier to find the right service for a backup when working with multiple servers.
- Automatic redirection to the **Restores** page after initiating a restore. From here you can immediately monitor the restore process, and see the restore history for your PMM instance.
- Added information about the type of storage for the backup artifacts on the **All backups** page. This helps you get a more comprehensive overview of your current backups.
- Design and usability improvements on the **Create Backup** and **All Backups** page to increase consistency with other PMM UI controls.

DBaaS migration

Starting with PMM 2.34.0, PMM installs and manages operators using Operator Lifecycle Manager (OLM) and database clusters using `dbaas-operator`. Nonetheless, these changes are backward incompatible, and you must perform these additional steps:

1. Force unregister the Kubernetes cluster from PMM (select Ignore errors checkbox) and register it back. PMM will install OLM and `dbaas-catalog` to manage the lifecycle of PXC, PSMDB, and DBaaS operators.
2. Create a database cluster for each cluster you have created in PMM. You can use the script to do it automatically during the maintenance window. Migration scripts use the default configuration to connect to Kubernetes clusters and run migrations against them.

New components

We have a newly added component, `VMProxy` starting with PMM 2.34.0. `VMProxy` is a stateless reverse proxy that proxies requests to VictoriaMetrics and optionally adds `extra_filters` query based on the provided configuration.

Percona LLC and/or its affiliates ©2024

9.15.2 New Features

PMM-11031 - To enhance the DBaaS experience, the UI has been extended to include the following:

- Network and Security - DBaaS users can set a source range from which they can connect to their DB clusters with the **Expose** field. You can modify the exposure level using the **Source Range** field. Additionally, the **Source Range** field allows you to choose the network from which you can connect.
- Configuring the database engine - Copy-paste the config string into the **Configuration** text box to configure the database engine.
- Configuring storage classes - You can choose from the available classes.

9.15.3 Improvements

- PMM-11024 - [DBaaS]: Move from dbaas-controller to dbaas-operator
- PMM-10873 - [DBaaS]: Create DB Cluster: We have modified the UI to include a separate window rather than a popup so that creating a DB cluster is easier.
- PMM-11132 - [DBaaS]: During K8s cluster registration, a new state called **Provisioning** is added. **Provisioning** is the default state, and once that is completed, the state will change to **Active**.
- PMM-11134 - [DBaaS]: Edit DB Cluster: We have modified the UI to include a separate window rather than a popup so that editing a DB cluster is easier.
- PMM-9262 - [Backups]: After starting a backup restore from the **All Backups** page, PMM now automatically redirects you to the **Restores** tab, where you can check the status of your new restore.
- PMM-11086 - [Backups]: Improved design for the **On Demand** and **Schedule Backup** cards on the **Create Backup** page. In addition, the **Created** column on the **Restores** tab has been replaced with **Started at** and **Ended** columns to provide more insight into the restore process.
- PMM-10924 - [Backups]: The **Location** column on the **All Backups** page has been enhanced to include information about the storage type used for the backup: Local or S3-compatible.
- PMM-11171 - [Backups]: The **Service name** field on the **Create Backup** page is now searchable. This makes it easier to find the right service for a new backup when working with multiple servers. Just paste the name of the server you're looking for, or start typing to see the list of matching servers.
- PMM-9079 - [Backups]: Added **Target Service** column under **Backup > Restores** to show the service to which a backup was restored.
- PMM-10773 - [Alerting]: Added option to unsilence alerts from the **Actions** menu under **Alerting > Fired Alerts**.
- PMM-11230 - [Alerting]: The **Fired alerts** page under **Alerting** now auto-refreshes more frequently to ensure that any new alerts are displayed in real-time.
- PMM-11255 - [Alerting]: PMM no longer transfers the name of the current alert rule when switching from **Percona templated alert** to **Grafana managed alert**.
- PMM-11090 - The Logs feature has been moved from the `dbaas-controller` to PMM, providing users with the ability to see the events for each pod and the logs for every container properly populated within PMM.
- PMM-11179 - Starting with PMM 2.34.0, we support push metrics mode while updating labels.
- PMM-11189 - Starting With PMM 2.34.0, we have implemented vmproxy integration and frontend API changes.
- PMM-10705 - [DBaaS]: As of PMM 2.34.0, Operator Lifecycle Management (OLM) is used for handling the installation and updating of operators for PMM in an effective, automated, and scalable manner.
- PMM-11069 - [Backups]: PMM no longer allows creating backup tasks with a name that already exists. After upgrading to PMM 2.34, any backup tasks with duplicate names are automatically renamed to apply the new naming constraints.

Percona LLC and/or its affiliates ©2024

9.15.4 Bugs Fixed

- PMM-7350 - [DBaaS]: Fixed an issue where DB Cluster did not restart all PXC servers and proxy SQL servers on restarting.
- PMM-11333 - [DBaaS]: Fixed an issue where the API key was missing on K8s cluster registration after moving to Operator Lifecycle Management (OLM).
- PMM-9505 - [DBaaS]: The options to update, restart, and view logs were enabled for paused clusters. This issue has now been resolved by disabling these options.
- PMM-11221 - [DBaaS]: Fixed an issue where the page did not load when DBaaS was disabled.
- PMM-7958 - During Postgres monitoring, databases were not getting deleted. This issue has been resolved now.
- PMM-9977 - [Backups]: After deleting an expanded row from the **Backup > All Backups** page, PMM would automatically expand the next row in the table. This issue is now fixed and PMM refreshes the list of backups artifacts without automatically expanding any rows.
- PMM-10942 - [Backups] - The scheduler for PITR backups now use clearer text labels to better define the Start time and the Frequency for the backups being created.
- PMM-10853 - [Alerting]: Removed **Save** button from the **Create alert rule** page as this was duplicating the functionality of the **Save and exit** button.
- PMM-10871 - [Alerting]: Fixed broken **Percona Alerting** link on the **Settings > Alertmanager integration** page. The link now correctly redirects to the **Alerting** page.
- PMM-11049 - Fixed a typo that was present in enhanced metrics scraper for `rds_exporter`.
- PMM-11137 - An issue with the MongoDB exporter not showing the version for MongoDB instance has been resolved.
- PMM-11210 - Whenever a cluster was unavailable or deleted, the K8s cluster list was empty. The issue has been resolved now.
- PMM-11239 - Despite removing pmm-server packages before upgrading PMM server, the packages were still present during upgrade. The issue has been resolved now.
- PMM-11372 - The user interface was broken after PMM was upgraded. This issue has been resolved now.

9.15.5 Known issues

- Starting with PMM 2.34.0, the recommended version for running Percona XtraDB cluster is 8.0.27. It’s impossible to create Percona XtraDB clusters using a version older than 8.0.27. All previously created clusters are not affected and will be migrated without any issues.
- PMM-11474: If more than two Kubernetes clusters are registered with PMM, and database cluster credentials are requested, an internal server error may occur.

9.15.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-02-14

Percona LLC and/or its affiliates ©2024

9.16 Percona Monitoring and Management 2.33.0

Release date:	**Dec 13, 2022**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

> **Important**
>
> PMM upgrade **fails** for versions prior to 2.33.0. Although there is a workaround, we highly recommend using the latest version of PMM. This ensures that you have access to the latest PMM features and that your environment runs on the latest version of the underlying components with all the bug fixes in place.

9.16.1 Release Highlights

Offline metric collection

Starting with PMM 2.33.0, we have introduced a new solution to prevent data collection interruption in PMM, via offline metric collection. A network outage between the PMM server and the client prevented metrics from being collected. As a result, the historical data was lost, which can be useful when analyzing trends, diagnosing problems, or investigating issues from a previous period.

In order to address the problem mentioned above, for the PMM server and PMM client running version 2.33.0, in case of a brief outage or network issue, metrics are stored on the monitored instance until the PMM server is up and running. At this point, the metrics will be pushed to the server. It only applies to PMM Clients that run in push mode, which is the default mode. The same holds when a brief network outage occurs on the client side.

> **Caution**
>
> The client node can currently store only 1 GB of offline data. So, for example, if your instance is down for three days, all the data will not be retrieved.

With the implementation of the above feature, we have taken the first step toward PMM HA. Furthermore, we are happy to provide a roadmap for PMM HA implementation, a much-anticipated solution by our users. In the event of a failover, your PMM setup will experience near-zero downtime, and monitoring will continue with the offline metric collection.

To ensure a smooth rollout, PMM HA will be introduced in phases, in line with your requirements and feedback.

> **Important**
>
> PMM HA will be rolled out in three stages. Stage one of PMM HA is the implementation to prevent data collection interruption with VictoriaMetrics integration for short outages **included** as part of **PMM 2.33.0**. HA stages two and three will be rolled out next year.

The following features/use cases will be covered as part of the various stages of PMM HA:

Stages of PMM HA	Solutions Provided
Stage one (included in PMM 2.33.0)	As an initial step towards preventing data collection interruption, we have developed the following: **Prevent data collection interruption with VM integration for short outages**

Percona LLC and/or its affiliates ©2024

Stages of PMM HA	Solutions Provided
Stage two (will be rolled out in 2023)	As part of PMM HA stage two, we plan to implement the following: **HA data sources** As part of stage two we will let the users use external data sources, thereby decreasing dependency on the file system.
Stage three (will be rolled out in 2023)	As part of PMM HA stage three, we plan to implement the following: **Clusterized HA** Clustered PMM will be the focus of stage three. Detailed information will be included in the upcoming release notes.

Guided tour of Percona Alerting

Following the recent improvements to PMM Alerting, we have added a short tutorial to help you explore our streamlined alerting system based on Grafana. The in-app tutorial automatically pops up when you first open the **Alerting** page to help you uncover the fundamentals of Percona Alerting.

Restore MongoDB backups easier

Building on the significant improvements for MongoDB Backup Management introduced in the previous release. We are now simplifying the process for restoring physical MongoDB backups. Starting with this release, you can restore physical backups straight from the UI, and PMM will handle the process end-to-end. Prior to this, you would require to perform additional manual steps to restart your MongoDB database service so that your applications could make use of the restored data.

Components Upgrade

- VictoriaMetrics has been upgraded from 1.77.1 to 1.82.1.
- As part of PMM 2.33.0, Grafana has been updated to 9.2.5, which fixes some critical vulnerabilities. For more information, see Grafana 9.2.5

9.16.2 New Features

- PMM-10889 - Starting with PMM2.33.0, while using the pmm-admin CLI command, you do not have to know the entire command. Just type the initial command and press Tab. The rest of the command will be autocompleted.

  Depending on the shell you are using, add the following to your shells .rc file for autocomplete to work:

  ```
  source <(pmm-admin completion -c <your shell>)
  E.g. source <(pmm-admin completion -c bash)
  ```

- PMM-10955: [Backup Management]: After restoring a backup, you can now check the time when the restore was finalized. For PITR backups, you can also check the time to which the restore was performed. This information is available on the tab **Backup > Restores** page, in the new **Finished at** column, and in the **PITR timestamp** field under the **Actions** menu.

9.16.3 Improvements

- PMM-7000: [HA]: Staring with PMM 2.33.0, if the PMM Server goes down, metrics will be buffered on the monitored instance until PMM Server goes back online, which is when metrics will get pushed to the PMM Server.
- PMM-10901: [Documentation]: To enable the users to use High Availability easily, we have added exclusive documentation for HA.

Percona LLC and/or its affiliates ©2024

- PMM-8516: [Backup Management]: When creating a backup, the Vendor field was not automatically populated in the Back-up on demand dialog. This issue is now fixed.
- PMM-10627: [DBaaS]: You can now create a single node cluster for PSMDB.
- PMM-10855: Avoid massive gRPC messages formatting for disabled log level
- PMM-10903: [DBaaS]: To simplify DBaas, you no longer have to manually enter the Public Address on the **PMM Settings > Advanced Settings** page. With DBaaS enabled, PMM will automatically detect and populate the **Public Address**.
- PMM-11085: [Backup Management]: We've removed the **Create backup** option under **All backups** page > **Actions** menu because this option can only create on-demand backups. You can continue to create both on-demand and scheduled backups using the **Create backup** button on this page.
- PMM-10881: [CVE]: As part of PMM 2.33.0, Grafana was updated to 9.2.5, which fixes some critical vulnerabilities. For more information, see Grafana 9.2.5.
- PMM-10385: Starting with PMM 2.33.0, for MySQL, the *Explain* tab is supported without the *Examples* tab. If a query in the *Explain* tab contains sensitive data, placeholders will replace them.
- PMM-8655: Latest state of agents is now available on the PMM server after the connection is re-established between the client and server.
- PMM-10969: VictoriaMetrics operator has been upgraded from 0.24 to 0.29.0.
- PMM-10554: Postgres collector now provides runtime duration for scrapes at the collector level. As a result, we can identify collectors who take too long. By utilizing this data, corrective action can be taken.
- PMM-10629: [Components Upgrade]: VictoriaMetrics has been upgraded from 1.77.1 to 1.82.1.

9.16.4 Bugs Fixed

- PMM-10858: In specific environments, including AWS, some EPEL repository mirrors did not respond within the time limit defined by `pmm-update`, currently set to 30 seconds. This was causing `supervisord` to kill `pmm-update-checker`, the component responsible for verifying if a newer version of the PMM Server is available for upgrade. This issue has been resolved now.
- PMM-10683: [DBaaS]: While registering a K8s cluster on the **Manage Components Versions** modal, the versions were not getting unchecked. The issue has now been resolved.
- PMM-11082: When adding RDS MySQL instances to PMM, some queries pertaining to table stats were executed, despite table stats being disabled.
- PMM-10637: Fixed an issue where the queries for some panels on the PostgreSQL Instance Summary dashboards were broken.
- PMM-11029: Whenever the `max-query-length` parameter was set to 3, 2, or 1, the PMM agent would be terminated. The issue has now been resolved.
- PMM-10732: Checking the select all checkbox on the **Services**, **Agents**, and **Nodes** modal in the **PMM Inventory** window deleted all entries on all the pages. The issue has been resolved now.
- PMM-10062: While monitoring a large number of nodes, some dashboards had incredibly long load times, resulting in an unresponsive UI. The issue has been fixed now.
- PMM-10983: Some panels on the dashboards were showing only short names. The issue has been resolved now.
- PMM-11041: There was typo on the Home dashboard: **COMMAND CENTER** graphs. This has been fixed now.
- PMM-10996: [Backup Management]: Fixed an issue where the initial description text for a storage location could not be updated.
- PMM-10718: [Backup Management]: When taking backups, PMM now shows a clearer error if the pmm-agent is incompatible with the specified backup operation.

Percona LLC and/or its affiliates ©2024

- PMM-10558: Top 5 Collections by Documents Changed panel on the MongoDB Collection Details dashboard did not display data correctly based on the selected filters. It has now been resolved.
- PMM-10493: [Portal Integration]: In some specific scenarios, disconnecting from Platform Portal as an Administrator would sometimes show a token error. This issue is now fixed.
- PMM-10845: [Advisors] Fixed specific error that occurred when working with Advisors that do not specify a frequency for executing the advisor check.
- PMM-10029: There was a potential issue with the network request exceeding the maximum message size. The issue has been resolved now.

9.16.5 Known issues

- PMM-11126: **PMM server cannot be upgraded to version 2.32.0**

  A bug in PMM Server ansible scripts caused PMM to upgrade Nginx's dependencies without updating Nginx itself. Due to this, PMM throws an error while upgrading and cannot upgrade to a newer version.

  > **Important**
  >
  > This issue has been resolved for PMM version 2.33.0. However, the issue persists on all the versions prior to 2.33.0.

  Solution

  While PMM is being upgraded, log in to the PMM server and run the following command:

  ```
  sed -i 's/- nginx/- nginx*/' /usr/share/pmm-update/ansible/playbook/tasks/update.yml
  ```

- PMM-10858: **PMM server does not show latest versions available with the instances created with AWS**

  For PMM versions prior to 2.33.0, in specific environments, including AWS, some EPEL repository mirrors did not respond within the time limit defined by pmm-update (currently set to 30 seconds). It was causing supervisord to kill pmm-update-checker, which determines if a newer PMM Server is available for upgrade.

  Solution

  Log in to the PMM Server and run the following command as a root user:

  ```
  $ yum-config-manager --setopt=epel.timeout=1 --save
  ```

9.16.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-02-09

9.17 Percona Monitoring and Management 2.32.0

Release date:	**Nov 8, 2022**
Installation:	Installing Percona Monitoring and Management

Percona LLC and/or its affiliates ©2024

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

> **Important**
>
> We recommend using the latest version of PMM. This ensures that you have access to the latest PMM features and that your environment runs on the latest version of the underlying components, such as VictoriaMetrics, with all the bug fixes in place.

9.17.1 Release Highlights

The release highlights are also available in video format. Watch the What's new in PMM 2.32 video

Backup Management for MongoDB is now generally available

We are happy to announce that the Backup Management for MongoDB has reached General Availability and is no longer in Technical Preview.

This change comes with significant functionality enhancements and added UX improvements:

- **Point-in-time recovery (PITR) restores for MongoDB backups**: This enables you to restore Amazon S3-compatible backups from any checkpoint in the past.
- **Local client storage**: In addition to Amazon S3-compatible, we have now also enabled the **Local Client** location option, which saves backups on the remote folder mounted to PMM client host.
- **Key UX improvements**: You can now create on-demand and scheduled backups from the same window instead of using separate modals. In addition, we have added more icons and renamed tabs columns to better label the actions available behind important options.

For more information, see the Backup and Restore topic in the documentation.

State-of-the-art new Home dashboard

In our continued effort to enhance user experience, we have added another feather to our cap by replacing the existing home dashboard with a brand new Home dashboard in PMM 2.32.0.

We are releasing this dashboard as the default Home dashboard. This dashboard was released in PMM 2.30.0 as an experimental dashboard. After considering your feedback and working meticulously on this dashboard, we have developed an enhanced version in PMM 2.32.0. Read more in the blogpost.

Highlights of the new Home dashboard include:

- Optimized performance to handle a large number of nodes concurrently
- Scalable to accommodate varying loads at all times
- Easy anomaly detection to check the health of your nodes from the Home dashboard
- Color-coded states on panels for a quick visual representation of the problem areas
- Comprehensive and easy-to-use dashboard to reduce cognitive load

For detailed information, see documentation.

Enhanced main menu

- In PMM 2.32.0, you can easily access all dashboards for monitored services due to the contemporized navigation structure. All the dashboards in the services folder that were not accessible from the main menu are now

Percona LLC and/or its affiliates ©2024

accessible from the main menu via **Other Dashboards**. Also, your custom dashboards can be accessed easily from the main menu via **Other Dashboards**.

For example, to access the dashboards for MySQL, go to **MySQL > Other dashboards**.

- Only monitored services are now included in the main menu, enhancing the user experience and minimizing cognitive load to maximize usability.

  For example, you will only see MongoDB dashboards in the main menu if you monitor the MongoDB database rather than all the other dashboards.

9.17.2 New Features

- PMM-9366: [Backup Management]: Added possibility to restore MongoDB PITR backups. This option is available under **Backup > All Backups > Actions > Restore this backup** for any PITR-type backups available in the table.
- PMM-9368: [Backup Management]: Added support for local client storage: When adding storage locations under **Backup > Storage Location**, you can now also select **Local Client** option as the location type for storing future backups.
- PMM-10599, PMM-10503: [Backup Management]: Refactored User Interface for creating backups to enable you to create on-demand and scheduled backups from the same window instead of using separate modals.
- PMM-10470: [Backup Management]: Following the support for MongoDB physical backups introduced in PMM 2.31, we have now added the possibility to restore these backups. Restoring MongoDB physical backups requires additional manual operations after the restore and is only available with Percona Server for MongoDB.
- PMM-10502, PMM-10831: [Backup Management]: Improved User Experience and labels on all four tabs of the Backup section:
  - Added icons to better illustrate failed and successful backups.
  - Added **Service Name** column to show the service from where the backup was retrieved.
  - Renamed **DB Vendor** tab to **DB Technology**.
  - Improved **Actions** menu with clearer labeled options.
- PMM-10742: [Backup Management]: Extended PMM compatibility with Percona Backup for MongoDB (PBM) to support the latest version, v. 2.0 PBM.
- PMM-10829: [Backup Management]: The backups table under **Backup > All Backups** now includes a new **Location** column which displays the service where the backup came from (S3 or Local Storage).
- PMM-8792, PMM-9055, PMM-9056 - PMM 2.32.0 introduces the parameter `--max-query-length` in the `pmm-admin` command for MySQL, PostreSQL, and MongoDB databases. It can also be set up through the UI while adding the service. You can see a new input field on the UI, where you can set the value for `max-query-length`. With these changes, queries longer than 2048 characters are not truncated if this parameter is set to `-1` (which means no limit), and the **Explain** and **Examples** tabs on the QAN dashboard do not throw an error (PMM-1899). These tabs provide contextual information pertaining to the query, thus enhancing user experience. For more information, see documentation.
- PMM-9545: To monitor Kubernetes clusters, PMM now collects Kubernetes metrics from kube-state-metrics. For more information on kube-state-metrics, see documentation.

  > **Disclaimer**
  >
  > This feature is still technical preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

- PMM-10579: [Dashboard]: Only monitored services are displayed on the main menu for a clean and simple User Interface.

Percona LLC and/or its affiliates ©2024

- PMM-10512: [Alerting]: When creating new alert rules. you can now use two new labels (**application_name** and **usename**) for **pg_stat_activity_** metrics.

9.17.3 Improvements

- PMM-9946: [DBaaS]: To enhance user experience, once PMM is deployed to a Kubernetes cluster and a user wants to use DBaaS, PMM automatically registers that Kubernetes cluster to DBaaS.
- PMM-10733: [DBaaS]: We have moved another step forward in simplifying the use of DBaaS by simplifying the selection of the database version.
- PMM-10723: [DBaaS]: To make the content legible, we have changed the color of the notification block.
- PMM-10156: [Dashboard]: Starting with PMM 2.32.0, we have an enhanced PMM dashboard that gives a birds-eye view of the services, infrastructure, and critical issues (if any).
- PMM-10612: [Dashboard]: You can now easily access all dashboards for monitored services. Dashboards in the services folder (MySQL dashboards) that were not accessible from the main menu are now accessible from **Other Dashboards**.
- PMM-10695: [Dashboard]: For precision, we have added a unit (%) for the CPU Anomalies panel.
- PMM-10487: [Dashboard]: Our Experimental MongoDB dashboards are now equipped with tooltips to help you understand the information provided in the panels.
- PMM-10692: [Dashboard]: Enhanced the High CPU Usage, High Disk Queue, and High Memory Usage panels to make the troubleshooting process easier and more actionable. Just click the node name on the panel to see the details of the problem.
- PMM-10751: [Dashboard]: For the Vacuum dashboard, when filtering by Service name, the first panel that shows the service name and resolution was hidden due to the Disclaimer. It has now been moved above the Disclaimer to make it readable.
- PMM-10690: [Dashboard]: Starting with PMM 2.32.0, we have enabled sparklines for Disk Read and Write panels to visually represent and show trends in the data.
- PMM-10697: [Dashboard]: We have improved the visualization of the Used Memory Anomaly Panel on the Home Dashboard so that all the anomalies are visible.
- PMM-10694: [Dashboard]: Added and refined tooltips for all the panels on the dashboard.
- PMM-10721: On the **Explore dashboard**, **Metrics panel**, removed the links pointing to the Grafana discussion board.
- PMM-10662, PMM-10633, PMM-10631, PMM-10600: An additional MySQL and MongoDB related datapoints have been added to Telemetry.
- PMM-10926: [Backup Management]: Updated the documentation with information about required user permissions for creating Mongo BM backups. For more information, see the MongoDB topic.
- PMM-10393: The panel for total time taken for scraping metrics by Postgres exporter is improved. Instead of showing coarse-grained measurements of the total time taken by the exporter, it now shows the total time taken per collector.
- PMM-10555: Runtime for collectors in MongoDB exporter is pushed via Telemetry to PMM. This information detects considerable scrape time for collectors and improves collector performance in future releases.

9.17.4 Bugs Fixed

- PMM-10944: [Backup Management]: Fixed an issue that was preventing QAN to retrieve data after a restore for MongoDB with authentication enabled.

Percona LLC and/or its affiliates ©2024

- PMM-10840: [Alerting]: Fixed embedded templates that are using expressions like **x == 0** or **x != 0**. PMM servers connected to Percona Platform already include this fix, since Percona Platform delivered it via Automatic Template Updates, a couple of weeks ago.
- PMM-10759: [Alerting]: Renamed the **Alerting** option in the PMM Settings to **Percona Alerting**. Since this option refers to Percona Alert Rule Templates, this new label reflects better the fact that this option is different from the basic, built-in Alerting functionality.
- PMM-10810: [Alerting]: Fired alerts now show the correct summary, which includes the node or service name specified in the template annotations.
- PMM-10500: [DBaaS]: DBaaS: Fixed an issue where a database cluster for an unregistered k8s cluster was still being displayed on the UI.
- PMM-10640: [DBaaS 1.10, 1.11]: Fixed an issue where a MongoDB 5.0.x cluster was not getting created.
- PMM-9874: [Dashboard]: Fixed an issue where the current ReplSet State stat is unreadable after modifying or restarting the database cluster for MongoDB Dashboard.
- PMM-10339: [Dashboard]: Fixed the duplicate values displayed on the CPU usage charts for MongoDB and MySQL dashboards.
- PMM-10802: [Advisors]: Fixed **MongoDBGetDiagnosticData** and **MongoDBReplSetGetStatus** query type for Advisors Checks V2.
- PMM-10688: This bug fix restores the ability to copy existing kubeconfig and register a new cluster by provided copy. However, AWS secrets will be hidden in the UI.

9.17.5 Known Issues

PMM-11029: **pmm-agent: killed when –max-query-length <=3**

If the value of the parameter `max-query-length` is set to 3, 2 or 1, the PMM agent will get terminated.

Solution

The value of the parameter should be greater than 3.

PMM-11126: **PMM server cannot be upgraded to version 2.32.0**

A bug in PMM Server ansible scripts caused PMM to upgrade Nginx's dependencies without updating Nginx itself. Due to this, PMM throws an error while upgrading and cannot upgrade to a newer version.

> **Important**
>
> This issue persists on versions prior to PMM 2.32.0.

Solution

Log in to PMM server and execute the following command:

```
sed -i 's/- nginx/- nginx*/' /usr/share/pmm-update/ansible/playbook/tasks/update.yml

[PMM-10858](https://jira.percona.com/browse/PMM-10858): **PMM server doesnt show the latest versions available with the instances created from AWS**


In specific environments, including AWS, some EPEL repository mirrors did not respond within the time limit defined by `pmm-update,` currently set to 30
seconds. It was causing `supervisord `to kill `pmm-update-checker`, the component responsible for determining if a newer PMM Server is available for upgrade.

!!! caution alert alert-warning "Important"
    This issue persists on PMM 2.32.0 and all versions prior to PMM 2.32.0.

**Solution**

Log in to the PMM Server and run the following command as a root user:
```


Percona LLC and/or its affiliates ©2024

```sh
$ yum-config-manager --setopt=epel.timeout=1 --save
```

9.17.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-02-12

9.18 Percona Monitoring and Management 2.31.0

Release date:	**Sep 26, 2022**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

Important

We recommend using the latest version of PMM. This ensures that you have access to the latest PMM features and that your environment runs on the latest version of the underlying components, such as VictoriaMetrics, with all the bug fixes in place.

9.18.1 Release Highlights

Enhanced query building, navigation and Alerting

We have upgraded Grafana to version 9.1 in PMM 2.31.0. With this upgrade, you can get a simplified user interface, enhanced visualizations, and a default unified Alerting experience. For more information, see What's new in Grafana.

Percona LLC and/or its affiliates ©2024

The following features have been introduced with PMM 2.31.0:

- **Expandable main menu**

  For an enhanced user experience, we have expanded and designed the main menu by moving the dashboard navigation from the PMM Dashboards menu to the main menu for quick access. So you can now browse dashboards like Operating System (OS), MySQL, MongoDB, PostgreSQL, ProxySQL, and HAProxy directly from the navbar.

- **Prometheus Query Builder**

  We have introduced a new UI query builder using which you can write Prometheus queries effortlessly. Prometheus queries can be complex to grasp, but with this Grafana release, it becomes easier.

- **Starred dashboards**

  To enhance user experience further, we have introduced **Starred**, where you can access your favorite dashboards directly from the main menu.

- **Explore to dashboard**

  You can now create panels and dashboards directly from **Explore**. You can also create a panel in a new or existing dashboard by clicking **Add to dashboard** in the **Explore** toolbar. The generated panel contains all the pane's queries, and the default visualization automatically picked from the current results shown in **Explore**.

- **Improvements to dashboard search**

  For a more effective search experience, we have introduced search by title as well as panel. With this, the overall performance has been enhanced as querying and sorting have become much faster.

- **Command Palette**

  You can now pull up a palette of commands for easier navigation using **cmd+k** (macOS) or **ctrl+k** (Linux/Windows).

General availability of Percona Alerting

With this release, we are streamlining the PMM alert setup process with an overhauled, unified alerting system based on Grafana.

All previous Integrated Alerting and Grafana Alerting functionality are now consolidated on the **Alerting** page. From here, you can configure, create and monitor alerts based on Percona or Grafana templates.

The **Alert Rules** tab has also evolved into a more efficient interface with an added layer for simplifying complex Grafana rules. You'll find that the new **Percona templated alert** option here offers the same functionality available in the old Integrated Alerting, but uses a friendlier interface, powered by Grafana's advanced alerting capabilities.

As an important and generally useful feature, this new Alerting feature is now enabled by default and ready to use in production!

For more information about Percona Alerting, check out Percona Alerting.

Deprecated Integrated Alerting

The new Percona Alerting feature fully replaces the old Integrated Alerting available in previous PMM versions. The new alerting brings full feature parity with Integrated Alerting, along with additional benefits like Grafana-based alert rules and a unified alerting command center.

If you have been using Integrated Alerting in the previous PMM version, your custom alert rule templates will be automatically migrated to PMM 2.31. After upgrading to this new version, you will find all your alert templates under **Alerting > Alert Templates**. These templates are also available for creating new alert rules using the new **Percona templated alerts** option on the Alert Rules tab.

Percona LLC and/or its affiliates ©2024

However, alert rules created with Integrated Alerting are not automatically migrated to Percona Alerting. After upgrading, make sure to manually migrate any custom alert rules that you want to transfer to PMM 2.31 using the Integrated Alerting Migration Script.

9.18.2 Vacuum monitoring dashboard

> **Important**
>
> This experimental dashboard is subject to change. It is recommended to use this dashboard for testing purposes only.

A new experimental dashboard has been released to help our users gain timely insights into the autovacuum process in PostgreSQL. This dashboard is designed to help fine-tune the vacuum configuration and to prevent a XID wraparound - both of which directly contribute to improved performance of the database.

This dashboard contains the following:

- Dead tuples - Identifies the number of dead rows in each table even though the rows are physically removed from the table.
- Last time vacuum ran - Tracks the last time a vacuum or autovacuum process successfully ran on each of your tables.
- Number of rows modified since last Analyze - The number of rows changed since the last time ANALYZE ran.
- Manual vacuum events - Tracks the number of times a manual vacuum was run on each table.
- Table disk usage - Tracking the disk space used by each table is crucial as it enables you to gauge expected changes in the query performance over time - but it can also help you detect potential vacuuming-related issues.

For more information, see documentation

9.18.3 New official (GA) deployment method - Podman

We are excited to announce the General Availability (GA) of Podman support for Deploying PMM 2.31.0. We had introduced it in 2.29.0 as a preview feature, but now we are production ready with this feature.

Simplied deployment with Database as a Service (DBaaS)

In our constant endeavor and focus on an enhanced user experience, in PMM 2.31.0, we have simplified the deployment and configuration of DBaaS as follows:

- With PMM 2.31.0, you can easily add a DB cluster from the newly created K8s cluster. All the **DB cluster** window fields are auto-populated with the values based on the existing K8s cluster.
- For PMM 2.31.0, while accessing DbaaS, if you have an existing Kubernetes cluster configured for DBaaS, you will be automatically redirected to the **DB Cluster** page. Otherwise, you would be redirected to the **Kubernetes Cluster** page.

9.18.4 New Features

- PMM-10089: We have upgraded Grafana to 9.1 in PMM 2.31.0.
- PMM-10092: We have expanded and improved the main menu for PMM 2.31.0 based on Grafana 9.1. The main menu contains crucial elements, one of them being starred dashboards.

Percona LLC and/or its affiliates ©2024

- PMM-10467: Integrated Alerting: Alert rules created with the Integrated Alerting feature in previous PMM versions can be migrated to the new Percona Alerting using the script that will be provided for the public release of PMM 2.31.
- PMM-10443: Dashboards: We have released a new experimental dashboard for collecting metrics for vacuum monitoring to help our users gain insight into the autovacuum process in PostgreSQL.

9.18.5 Improvements

- PMM-10560: Dashboard: We have improved the MongoDB Collection Details Dashboard. The Collection Activity panel is now collapsed by default, and the graphs are visible only when the `--enable-all-collectors` parameter is passed in pmm-admin command.
- PMM-10349: DBaaS: For Simplified DBaaS experience, you will be automatically redirected to the *DB Cluster* page if you have an existing Kubernetes cluster configured for DBaaS. Otherwise, you would be redirected to the *Kubernetes Cluster* page.
- PMM-10064: Introduced UI changes on the **PMM Inventory** pages for better readability and user experience.
- PMM-10076: For an improved UI experience, we have enforced the enter button for the date picker, which performs the same action as clicking **apply time range**.
- PMM-10018: Lowered the default values of CPU and memory for haproxy while deploying a DBaaS Percona XtraDB Cluster (PXC) cluster to avoid wasting resources.
- PMM-9118: Refined the error message thrown when a secure connection cannot be established while adding a monitoring database to PMM, thus making it much more contextual and easy to troubleshoot.
- PMM-10516: Added support for MongoDB physical backups. With this, PMM backup management now supports both logical and physical backups for MongoDB services that are based on Percona Server for MongoDB.

  Percona Backup for MongoDB (PBM) compatibility: MongoDB backups in PMM require Percona Backup for MongoDB version 1.8.x. Make sure not to install the latest BPM version via the percona-release tool, as this automatically installs BPM v.2.0 which is not yet compatible with PMM.
- PMM-10095: While removing the service with pmm-admin, now you do not have to specify the service name for a single service. PMM will automatically look up this service and remove it.
- PMM-10495: By default, Swagger UI attempts to validate specs against swagger.io's online validator. For PMM 2.31.0, we have disabled it to safeguard your privacy.
- PMM-7806: Upgraded `postgres_exporter` version used in pmm from 0.8.0 to 0.10.1. With this upgrade, you can access the latest features with all the bug fixes in place.

9.18.6 Bugs Fixed

- PMM-10628: While adding MongoDB or PostgreSQL database via socket, PMM was throwing an error. This has been fixed in PMM 2.31.0.
- PMM-10624: Empty dataname label after PostgreSQL upgraded in PMM.
- PMM-10608: Fixed issues pertaining to the new Home Dashboard, such as high CPU panel unreadable for a higher number of nodes (more than ten nodes), filtering for the environment as well as node name not working correctly, time range not working as expected, some panels on the dashboard not displaying data for selected filters, and so on.
- PMM-10587: Fixed an issue where if you hover over the telemetry icon stacktrace with error appears.
- PMM-10513: Fixed an issue where pmm-admin summary stored PMM password in client/status.json, thus posing a security risk.
- PMM-10065: Fixed an issue encountered while monitoring a large number of nodes (1000) and services. Here, the changing scrape frequency led to data not being captured on the dashboard and throwing a templating error.

Percona LLC and/or its affiliates ©2024

- PMM-9973: Fixed an issue where the CPU utilization for Postgres Exporter was on the higher side.
- PMM-10044: Fixed an issue where QAN displayed enormously high values.
- PMM-10218: Fixed an issue for corrupted pprof file on concurrent pprof request.

9.18.7 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-01-30

9.19 Percona Monitoring and Management 2.30.0

Release date:	**Aug 24, 2022**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

Important

We recommend using the latest version of PMM. This ensures that you have access to the latest PMM features and that your environment runs on the latest version of the underlying components, such as VictoriaMetrics, with all the bug fixes in place.

Percona LLC and/or its affiliates ©2024

9.19.1 Release Highlights

NEW EXPERIMENTAL DASHBOARDS

> **Important**
>
> These experimental dashboards are subject to change. It is recommended to use these dashboards for testing purposes only.

- We have introduced a more user-friendly and insightful experimental Home Dashboard in PMM version 2.30.0. With this new Home Dashboard, you can experience a boost in performance while monitoring a large number of nodes or services in PMM. Repeated rows (for a large number of nodes) on the dashboard can take a considerable time to load the Home Dashboard UI. Our new experimental dashboard ensures fast retrieval of data on the Home Dashboard. For more information, see documentation.
- We have introduced the following new real-time experimental dashboards to collect more metrics and data for MongoDB:
  - **Experimental MongoDB Collection Overview**

    This dashboard contains panels for data about the Hottest Collections in the MongoDB database. For more information, see documentation.
  - **Experimental MongoDB Collection Details**

    This experimental dashboard provides detailed information about the top collections by document count, size, and document read for MongoDB databases. For more information, see documentation.
  - **Experimental MongoDB Oplog Details**

    This real-time dashboard contains Oplog details such as Recovery Window, Processing Time, Buffer Capacity, and Oplog Operations. For more information, see documentation.

Database as a Service (DBaaS)

Enhanced the user experience by simplifying the configuration and deployment of DBaaS database clusters:

- With PMM 2.30.0, you can add a DB cluster literally at the click of a button. All the fields will be automatically populated with the default value. Click *Register*, and your DB cluster is added. For more information, see documentation.
- Starting with PMM 2.30.0, we have simplified the DBaaS configuration and deployment of database clusters by providing suggestions on the k8s cluster name based on the kubeconfig file.
- We have simplified DBaaS and Percona Platform connection configuration by suggesting and automatically setting the public address of the PMM server if it's not set up. This happens when connecting to Percona Platform, adding a Kubernetes (K8s) cluster, or adding a DB cluster.

Operators support

PMM now supports Percona Operator for MySQL based on Percona XtraDB Cluster (1.11).

9.19.2 New Features

- PMM-9652: Dashboard: New experimental dashboards for MongoDB are now available with more MongoDB metrics and data like Hottest collections, Data Size, Recovery Window, Processing Time, Buffer Capacity, Oplog Operations, and so on.

Percona LLC and/or its affiliates ©2024

- PMM-10367: Dashboard: We have introduced a new user-friendly and insightful experimental Home Dashboard in PMM version 2.30.0. This Home Dashboard improves the performance while monitoring a large number of nodes or services in PMM.
- PMM-10327: You can add a free K8s cluster via Percona Platform Portal and use it for testing DBaaS in PMM. Read more in the Private DBaaS with Free Kubernetes Cluster blogpost.

> This feature will be available on https://portal.percona.com/ with the PMM 2.30.0 GA release.

9.19.3 Improvements

- PMM-10340: DBaaS: Enhanced user experience to simplify DBaaS configuration and deployment of Database clusters with suggestions on k8s cluster name based on k8s config file.
- PMM-10368: Simpler DBaaS and Percona Platform connection configuration by suggesting the public address of the PMM server.
- PMM-5680: Logs: For an enhanced and simplified troubleshooting experience, we have added the following logs in a ZIP file in the **pmm-agent** directory, which you can obtain from the pmm-admin summary command:
  - exporter logs
  - qan agent logs
  - pmm-agent log
- PMM-5492: Logs: We have added the profiling data files to the logs directory to boost the troubleshooting process. You can obtain this by executing the `pmm-admin summary --pprof` command.
- PMM-5005: Logs: Added a more pronounced error message for the PMM agent to help you distinguish between potential problems, such as connectivity issues, unconfigured agent issues, connection time-out issues, etc.
- PMM-10147: To minimize resource consumption and DB connections, we have limited the number of concurrent jobs and actions that PMM Agent can execute concurrently. By default, PMM Agent will now run maximum 32 jobs and actions in parralel. You can configure this limit with the flag `--runner-capacity=<some number>` or environment variable `PMM_AGENT_RUNNER_CAPACITY=<some number>`.
- PMM-9914: PMM generates config files into the `/tmp` directory when a service is added to monitoring. Files in `/tmp` can get deleted by the Operating system. With PMM 2.30.0, we have ensured that the pmm-agent successfully restarts the child agents even though the config files in the `/tmp` directory are deleted and the scraper/exporter processes are killed, thereby ensuring that the exporter is not broken.
- PMM-9994: QAN: Improved the description of the QAN filter in the user guide for better clarity.
- PMM-9632: For flexibility, we have added the ability to change some of the pg_stat_monitor settings. Previously, this was hardcoded.
- PMM-10122: Applied the pmm-agent `--log-level` to the exporter's output so that each message from the exporter with `log-level=WARN` will be logged in pmm-agent with the same `log-level=WARN`.
- PMM-9921: Migrated pmm2-client Docker image to ubi-micro to reduce the number of packages and the image size.
- PMM-10085: Support for Percona XtraDB Cluster (PXC) 1.11.0 in PMM.
- PMM-10351: DBaaS: Full form of DBaaS (Database as a Service) is displayed on the Dashboard for clarity for the new users.

9.19.4 Bugs Fixed

- PMM-10474: Fixed an issue where the PMM tour gets in the way when sharing a dashboard via a rendered image with an error message at the top (HTTP 404).

Percona LLC and/or its affiliates ©2024

- PMM-10483: Fixed an issue where the QAN dashboard was not reloading after modifying the filters, refresh rate, sorting, etc.
- PMM-10299: Fixed an issue where the MongoDB ReplSet Dashboard overview section shows incorrect information for the Cluster and ReplSet members count and does not change even after selecting a different cluster name.
- PMM-10494: Fixed an issue where the Confirm action modal is stuck until the cluster is registered.
- PMM-9987: Fixed an issue in QAN where the metrics values did not match those on the Query details panel.
- PMM-10159: Fixed an issue where the `logs.zip` directory contained the credenatials.
- PMM-9398: Fixed an issue where an error was thrown on the MySQL InnoDB Details dashboard for too many data points.

9.19.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-18

9.20 Percona Monitoring and Management 2.29.1

Release date:	**July 28, 2022**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

Important

We recommend using the latest version of PMM. This ensures that you have access to the latest PMM features and that your environment runs on the latest version of the underlying components, such as VictoriaMetrics, with all the bug fixes in place.

9.20.1 Release Highlights

Fixed the following CVE:

PMM-10338: We have updated Grafana to fix critical CVE’s impacting Alerting in PMM.

9.20.2 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2022-07-28

9.21 Percona Monitoring and Management 2.29.0

Release date:	**July 19, 2022**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

> **Important**
>
> We recommend using the latest version of PMM. This ensures that you have access to the latest PMM features and that your environment runs on the latest version of the underlying components, such as VictoriaMetrics, with all the bug fixes in place.

9.21.1 Release Highlights

Guided tour of PMM

PMM now includes a short in-app tutorial that automatically pops up when you first launch the application.

If you are a new user, this is your virtual unboxing of key PMM features to help you get started easier.

If you are an Intermediate or Advanced user, use the tour as a checklist of features worth exploring to ensure you're making the most out of your PMM.

Deploying PMM in various environments

K8S

> **Disclaimer**
>
> Deploying PMM on Kubernetes is still technical preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

Starting with PMM 2.29.0, we have introduced the helm chart supported by Percona to seamlessly deploy your PMM instances on Kubernetes (k8s), a prominent container orchestration system. With the Helm chart, you can customize your deployment. You can tweak your PMM installation by setting these various parameters. For more information, see the documentation.

Before introducing the helm chart for PMM, if you wanted to deploy PMM, you would need a separate virtual machine to run PMM, but with the helm chart, that is not required. Helm Charts provide the ability to leverage Kubernetes packages at the click of a button or a single CLI command.

PODMAN

> **Disclaimer**
>
> Podman support is still technical preview and is subject to change. We recommend that early adopters use this feature for testing purposes only.

Percona LLC and/or its affiliates ©2024

Starting with PMM 2.29.0, get your PMM instances up and running quickly with the Podman. Using Podman makes it easy to run, build, and deploy PMM.

Deployment of PMM with the Podman is far more secure as it does not require root privileges to create, run and manage containers. It lets you run containers as a non-root user, so you never have to give a user root permission on the host. Thus it adds a new security layer, which is crucial in the event of a security breach. Podman also allows multiple unprivileged users to run containers on the same machine. For more information on deploying PMM on Podman, see the documentation.

LOCAL FILE SYSTEM

You can start your PMM instances without a data container using your local file system.

You can use docker volume or host directory with the `-v` docker-cli option for storing PMM state (databases, configs, etc.)

RHEL 9 SUPPORT (CLIENT):

PMM client now supports RHEL 9.

Monitoring

META METRICS

We have added some metrics for Mongodb exporters to monitor the time it takes to collect data for different collectors. With these metrics, you can identify the monitoring cost of various collectors by the MongoDB exporters.

PXC CLUSTER DASHBOARD

> **Important**
>
> This experimental dashboard is subject to change. It is recommended to use this dashboard for testing purposes only.

Created a new experimental PXC/Galera Cluster Summary dashboard to make it simple, intuitive, and provide at-a-glance visibility for better decision making. Critical data for the PXC cluster is put together in this dashboard to obtain feedback from our users. For more information, refer to the documentation.

Earlier, the PXC Cluster data was distributed over different dashboards. The users had to browse the MySQL compare dashboard to check for the data like slow queries and network overviews that were not in the PXC nodes compare dashboard. This made it time-consuming to identify any possible issues with the database. With the new PXC dashboard, we aim to solve this problem and gain insightful data about the cluster and services. Now the users can have an overview of the PXC clusters from this dashboard. Also, users with beginner to intermediate levels of expertise can effortlessly analyze the data on this dashboard.

Troubleshooting

MANAGING LOGS

As a PMM user, you can explicitly set a detailed level of logging so that, by default, you will have meaningful logs to see, thereby enhancing your troubleshooting experience.

If you want detailed logging, you can set `--log-level` to INFO. By default, INFO is not logged. Thus, you have more control over the logs you want to view with the `--log-level` parameter with the flexibility to choose options such as warning, error, fatal, and info.

For example, to set the log level to `fatal` for a MySQL database:

```
pmm-admin add mysql --username=XXX --password=XXX --log-level=fatal
```


Percona LLC and/or its affiliates ©2024

ADDITIONAL PARAMETERS ADDED IN COMMANDS

- Added the agent (exporter) port to the output of the `pmm-admin` commands, which is crucial during debugging.

```
pmm-admin list
pmm-admin status
pmm-admin inventory list agents
```

- Added new flag `--agent-password` to the following commands to support a custom password via the CLI to make it more secure:

```
pmm-admin register --agent-password=AGENT-PASSWORD
pmm-admin config --agent-password=AGENT-PASSWORD`
```

- You can use the `pprof` parameter in the `pmm-admin` summary command.

```
pmm-admin summary --pprof
```

`pprof` reads a collection of profiling data from the PMM server, enabling you to generate reports in textual and graphical formats to visualize and analyze data.

You can analyze CPU, Memory usage, heap size, and thread count for your PMM components with the data obtained.

Percona Platform Integration

TERMINATE PERCONA PLATFORM CONNECTIONS AS AN ADMIN

When regular users are not able to disconnect a server (for example, when PMM was moved to a network segment without outbound connections to public networks), PMM Admins can now terminate such connections instead.

The **Disconnect** option under **Settings > Percona Platform** now enables PMM Admins to retire PMM instances from Percona Platform even if they are not logged in with a Percona Account.

When disconnecting servers without a Percona Account, the lingering servers are not automatically removed from the list of PMM instances in Percona Platform. As a Percona Platform Admin, you can ensure that your list of PMM instances stays up-to-date by manually removing those unavailable servers via PMM instances.

EASIER ACCESS TO YOUR PERCONA SUPPORT SPECIALIST

If you are a Percona customer, you can now find the contact details of your dedicated Percona Support Specialist on the new **Environment Overview** tab on the main menu.

The **Percona Contacts** section here shows the name and email of the Customer Success Manager assigned to your organization, who can help with any PMM queries. This new tab will soon be populated with more useful information about your PMM environment.

Advisors

EXTENDED SEVERITIES FOR ADVISOR CHECKS AND ALERTS

We have extended and unified the list of severity levels for the notifications coming from Integrated Alerting and Advisor Checks. When you create a new alert rule, you can now choose from a broader range of severity labels: **Emergency**, **Alert**, **Critical**, **Error**, **Warning**, **Notice**, **Info**, **Debug**.

The same severities are now also available for notifications coming from failed checks under **Advisor Checks > Failed Checks**. We've also made a small UX improvement to the way failed checks are grouped by severity in the table.

Percona LLC and/or its affiliates ©2024

USE FILTERS TO EXPLORE AVAILABLE ADVISORS CHECKS

As we're constantly adding more advisors for checking the health of your connected databases, you may find it useful to now be able to drill-down through this constantly growing list.

Use the filters on the **Advisors Checks > All Checks** page to search advisors by **Name**, **Description**, **Status** or **Interval**. If you need to share the filtered results with your team members, send them the PMM URL, which saves your search criteria and results.

DBaaS

Simplified the API such that it requires fewer parameters to run the API.

With this new implementation, the only mandatory field is the Kubernetes cluster name. All other fields such as resource parameters, image field, params field, and name filed are optional. The API sets these to default values if these fields are not specified.

Also, the documentation has been enhanced for these APIs.

Components upgrade

- Upgraded VictoriaMetrics operator to version 0.24.
- Upgraded the ClickHouse plugin to 2.4.4, which fixes some of the CVEs.
- With this release, we are upgrading grpc-gateway to version 2.0, which deprecates the error field in error responses. If you are using the PMM API, ensure to replace this with the new message field from the Google Cloud API Error model.
- Upgraded the mysqld exporter to the upstream version 0.14.0.

Important bug fixes

- PMM-9981: Fixed excessive space usage for Group replication monitoring. Services with enabled group replication and custom queries now take up less disk space and perform faster. To achieve this, we removed the **Transactions Details** table from the **Group Replication** dashboard, as these were based on labels that were generating high-cardinality data.

  If you still have issues with performance in Group Replication dashboard, log into PMM as an admin, and use *https://pmm_server_url/prometheus/api/v1/admin/tsdb/delete_series?match[]=* to remove all time series for the following metrics:

  - mysql_perf_schema_replication_group_worker_rep_delay_seconds
  - mysql_perf_schema_replication_group_worker_transport_time_seconds
  - mysql_perf_schema_replication_group_worker_time_RL_seconds
  - mysql_perf_schema_replication_group_worker_apply_time_seconds
  - Mysql_perf_schema_replication_group_worker_lag_in_secondds

  For example, to remove all time series for the *mysql_perf_schema_replication_group_worker* metric use the following URL: *https://PMM_SERVER_URL/prometheus/api/v1/admin/tsdb/delete_series?match[]=mysql_perf_schema_replication_group_worker_rep_delay_seconds*
- PMM-9510: Fixed incorrect and duplicate information displayed on the Dashboard for MongoDB Replica Set.

9.21.2 New Features

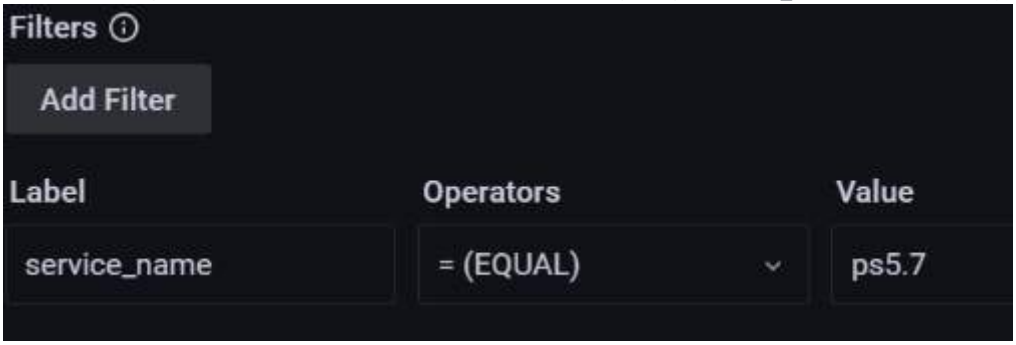
- PMM-10133: New User Onboarding: Added a guided tour that highlights the main PMM components on your first onboarding journey.

Percona LLC and/or its affiliates ©2024

- PMM-10059: Advisors: Extended and unified list of severity levels for the notifications coming from Integrated Alerting and Advisor Checks.
- PMM-7110: You can now obtain profiling data with the pprof tool that helps analyze CPU, Memory usage, heap size, and thread count for your PMM components.
- PMM-8660: Percona customers can now find the contact details of their dedicated Percona Support Specialist on the new **Environment Overview** tab on the main menu.
- PMM-7925: [Technical Preview]: Starting with PMM 2.29.0, you can now deploy your PMM instances using Podman, which is considered far more secure as it does not require root access.
- PMM-9613: [Technical Preview]: You can now scale and deploy your PMM instances faster using the Kubernetes clusters.
- PMM-9919: We have added some additional metrics to Mongodb exporters to monitor the time it takes to collect data for different collectors that would allow us to identify the monitoring cost of various collectors by MongoDB exporters.

9.21.3 Improvements

- PMM-9766: Advisors: Added filters on the **Advisors Checks > All Checks** page to narrow the list of available advisor checks.
- PMM-7491: Alerting: We've redefined the Filters option under **Integrated Alerting > Alert Rules > New Rule** to simplify the way you target specific services or nodes for your alert rules.
- Instead of typing an exact filter pattern, you can now intuitively create a filter by choosing from the available operators and filling in the predefined fields. So instead of typing: *service_name=name123*, just fill in the following fields:



- PMM-9704: With the implementation of a simplified API for DBaaS, you only have to specify the Kubernetes cluster name. All the other fields are optional, and the API will set the default values for these fields if not specified.
- PMM-8222: DBaaS: Enhanced the documentation for the API.
- PMM-9611: PMM now warns you if your current domain is different than the one specified in the Grafana .INI configuration file. Since this can generate incorrect short URLs when sharing the link to a PMM dashboard, make sure to correct this mismatch if you see a warning on the **Share Panel > Link page**.
- PMM-7326: You now have more control over the logs you want to view with the **–log-level** parameter. With this enhancement, you can experience more pronounced logging with the flexibility to choose options such as warning, error, and fatal.
- PMM-8566: For enhanced security, you can now specify the custom Basic Auth password for agents when adding a Node.
- PMM-9362: PMM Admins can now disconnect servers from Percona Platform even if they are not logged in with a Percona Account Percona Account.
- PMM-6592: PMM now displays the ports used by the exporters in the output of the pmm- admin list and status commands, which are crucial while debugging.

Percona LLC and/or its affiliates ©2024

- PMM-7110: You can now use the pprof parameter in the pmm-admin summary command to obtain profiling data from the PMM server, thereby enabling you to generate reports in textual and graphical formats to visualize and analyze data.
- PMM-7186: The pmm-admin summary command now retrieves a list of targets scraped by Victoriametrics on the client-side.
- PMM-8308: To enhance the troubleshooting experience, in addition to the information summary and archive, you can now view the paths for all the exporters managed by the pmm-agent in the diagnostic data.
- PMM-9650: Created a new experimental dashboard for PXC Cluster Summary to make it simple and intuitive.
- PMM-10039: VictoriaMetrics operator has been upgraded to 0.24.
- PMM-10083: Upgraded the ClickHouse plugin to 2.4.4, which fixes some of the CVEs.
- PMM-10103: Added RHEL 9 support for PMM client.
- PMM-10155: Access QAN from the main menu instead of the PMM dashboard for better reach and visibility.
- PMM-2038: Starting with PMM 2.29.0, we have upgraded the mysqld exporter to the upstream version.
- PMM-9913: PMM removes all the unnecessary temporary files created while adding a monitoring service, which might impact your PMM server's performance if not deleted.
- PMM-10067: Updated the version of Percona Toolkit to 3.4.0.

9.21.4 Bugs Fixed

- PMM-10127: Alerting: Fixed issue where changing the page under **Integrated Alerting > Alerts** would not update the list of results.
- PMM-10110: DBaas: Fixed an issue where an incorrect operator version was installed.
- PMM-9965: DBaas: Fixed an issue where re-registering a K8s cluster returned an error.
- PMM-7143: DBaaS: Fixed an issue where proper node metrics were not exposed for database instances created with DBaaS.
- PMM-8677: DBaaS: Fixed the names of the operators used in PMM on the DBaaS UI to reflect their official names rather than short forms.
- PMM-9998: Fixed an issue where 'Unknown column…' errors are thrown when monitoring MariaDB from the Debian package (10.2, 10.3, 10.4, 10.5).
- PMM-10144: Fixed an issue where if an already existing service was added via CLI, incorrect output was being displayed.
- PMM-9320: Fixed an issue where it was impossible to add MongoDB to PMM when the password contained "+."
- PMM-9909: Fixed an issue where the redirection from Home Dashboard Stats Panel to Node Summary did not work.
- PMM-10049: Removed unnecessary error messages from the pmm-agent log.
- PMM-8819: Fixed an issue where the metric data displayed in the table was not appropriately aligned.
- PMM-10087: PMM would show an "Internal server error" for custom alert rules created with a **"ia=1"** label. This issue is now fixed.
- PMM-7462: Fixed an issue where you can update the PMM settings via the UI if they are not provided as an Environment variable.
- PMM-9510: Fixed incorrect and duplicate information displayed on the Dashboard for MongoDB Replica Set.
- PMM-9910: Fixed an issue where upgrading PMM from 2.26.0 to 2.27.0 via the UI failed due to a GRPC error.
- PMM-9981: Enormous space usage and slow queries for Group replication monitoring
- PMM-10069: Fixed a typo in the tooltip for InnoDB Random Read Ahead.

Percona LLC and/or its affiliates ©2024

- PMM-10161: Removed unnecessary breadcrumb navigation panel from all dashboards.

9.21.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-07-19

9.22 Percona Monitoring and Management 2.28.0

Release date:	**May 12, 2022**
Installation:	Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL and MongoDB.

> **Important**
>
> We recommend using the latest version of PMM. This ensures that you have access to the latest PMM features and that your environment runs on the latest version of the underlying components, such as VictoriaMetrics, with all the bug fixes in place.

9.22.1 Release Highlights

Advisors

- **Advisor checks enabled by default**

  Starting with the previous release and continuing with this one, we have added significant improvements to the Advisors Checks functionality in performance, assessment coverage, and user experience.

  As a mature and generally useful feature, this option is now enabled by default for easier access to automatic checks and better insight into database health and performance, delivered by Percona Platform.

  > **Note**
  >
  > Upgrading to PMM will automatically enable this feature for existing PMM instances. You can disable it at any time from your PMM dashboard on the **Advanced Settings** page.

- **Run individual advisor checks**

  In addition to running all available advisors at once, you now have the option to run each advisor check individually.

  This gives you more granular control over the checks on your connected databases. Running checks individually also means that you get the results for relevant advisors faster and that you can focus on resolving failed checks one at a time. For more information, see Working with Advisor checks.

- **Enhanced Advisor checks**

Percona LLC and/or its affiliates ©2024

PMM 2.28 includes a new major version of Advisors that features some important enhancements. The most significant changes are:

- Support for multiple queries
- Support for Victoria Metrics as a data source

In a nutshell, these changes will allow experts to create more intelligent advisor checks to continue delivering more value to your connected PMM instances. The file format in which Advisors checks are written has been updated to support the new functionality provided by the Advisors service part of Percona Platform.

This is a breaking change, so we recommend upgrading your PMM instance to benefit from these enhancements. For more information, see Develop Advisors.

Ubuntu 22.04 LTS support

We are providing binaries for the recently released version of Ubuntu from this release.

Components upgrade

- **VictoriaMetrics**: VictoriaMetrics has been upgraded to 1.76.1.
- **Node exporter**: Node Exporter has now been updated to 1.3.1.

> **Important**
>
> If you customized the disabled collectors, the list could change. Check the available collectors in Documentation.

9.22.2 New Features

- PMM-9749: Advisors: Possibility to run individual advisor checks separately.
- PMM-9469: Advisors: Ability to have multiple queries in a single check.
- PMM-9468: Advisors: Ability to query VictoriaMetrics as a data source.

9.22.3 Improvements

- PMM-9841: Advisors: Advisor checks are now enabled by default.
- PMM-8326: Advisors: Changed the icon for the **Edit Check Rule** option with a more suggestive one that better reflects this functionality.
- PMM-9907: `pmm2-client` now supports Ubuntu 22.04 LTS.
- PMM-9780: VictoriaMetrics has been upgraded to 1.76.1.
- PMM-5871: Node Exporter has now been updated to 1.3.1.
- PMM-9958: The PMM logs button, which is used to download PMM logs for troubleshooting, is added to the help panel for better accessibility and enhanced user experience.
- PMM-9672: Minor UI improvements to the visual elements in the breadcrumb trails to visually align them to the look-and-feel of Grafana pages and improve overall UI consistency.

9.22.4 Bugs Fixed

- PMM-9854: Advisors: In some scenarios, PMM was not displaying the complete list of advisors available for instances connected to Percona Platform. This issue is now fixed.

Percona LLC and/or its affiliates ©2024

- PMM-9848: Advisors: Fixed text contrast issue on the **Failed Advisor Checks** page that was visible when navigating the list of results while using PMM with the Light theme.
- PMM-9426: DBaaS: Fixed an issue related to K8s monitoring where the K8s monitoring failed with K8s version 1.22 and higher.
- PMM-9885: Dashboard: Fixed the documentation links on the **Advanced settings** page on the PMM dashboard.
- PMM-9828: Fixed an issue with the QAN dashboard navigator/explorer where if you open QAN from a dashboard and try to navigate to a different dashboard, the explorer keeps closing/refreshing, making it impossible to navigate.
- PMM-9363: PMM users logged in via SSO would still have access to PMM after disconnecting. This issue is now fixed and PMM correctly terminates SSO sessions after disconnecting.
- PMM-9415: Backup Management: Fixed an issue where initial data restore on AWS instances fails. However, consecutive data restore attempts were successful.

9.22.5 Known Issues

PMM-9992: **Error while using reverse proxy (like Nginx)**

While using a reverse proxy (for example, Nginx) in front of PMM, you can run into the error `origin not allowed` after upgrading to PMM 2.27.0 or newer versions.

Solution

Add the Host header to the reverse proxy configuration file.

Example

For Nginx, add the following:

```
proxy_set_header Host $http_host;
```

9.22.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-18

9.23 Percona Monitoring and Management 2.27.0

Date: April 14, 2022
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is an open source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

Percona LLC and/or its affiliates ©2024

Important

We recommend using the latest version of the PMM instance. This ensures that you access the latest features of the underlying components, such as VictoriaMetrics, with all the bug fixes in place. One of the critical features of VictoriaMetrics is stream parsing mode, which enhances the performance of PMM and saves memory when scraping targets expose millions of metrics.

9.23.1 Release Highlights

PMM and Percona Portal Integration

- **Enhanced UX for connecting PMM server to Percona Portal**

  To leverage Percona Platform's recent support for federated identity, PMM now uses access-token authorization for connecting PMM instances to Percona Platform.

  This replaces the former username/password authentication used in PMM 2.26.0 and older versions.

  For more information, see Integrate PMM with Percona Platform.

- **Access to account information for Percona customers**

  When you connect your PMM instances to Percona Platform as a Percona customer, you can now check all your Percona Platform account information on the new **Entitlements** and **Support Tickets** tab on the main menu.

  For more information, see Check your account information in the online Help.

Advisors

- **Broader database health assessments with Advisors**

  With this release, we are renaming Security Checks to Advisors and grouping them according to the functionality and recommendations they provide. To reflect these changes, the old **Security Threat Tool** option is now called **Advisors**.

  In addition, we have added new checks for MySQL and MongoDB. To check the list of checks and the tier for which they are available, see Advisor checks for PMM in the Percona Platform online Help.

- **New Paid tier and special Advisor checks for customers**

  In addition to Registered Checks, Percona customers who connect PMM to Percona Platform now also have access to Paid Advisor checks, which offer more advanced database health information.

  For more information, see Working with Advisor checks.

- Advisor checks have been extended with two new query types: `GetDiagnosticData` and `replSetGetStatus`.
- Improved the documentation around developing checks.

DBaaS

Simplified experience in registering EKS cluster with kubeconfig generated by `eksctl`.

Components upgrade

- **Grafana**: PMM 2.27.0 has now migrated to Grafana 8.3.5. This version of Grafana is loaded with a gamut of exciting features. For more information, see What's new in Grafana v8.0.
- **VictoriaMetrics**: VictoriaMetrics has been upgraded to 1.72.0.

Percona LLC and/or its affiliates ©2024

9.23.2 New Features

- PMM-9718: PMM and Percona Portal Integration: Federated connections to Percona Platform.
- PMM-9305, PMM-8661: PMM and Percona Portal Integration: Visibility over Percona Platform Entitlements and Support Tickets.
- PMM-9473: Advisors: Additional abilities for MongoDB Advisor Checks.
- PMM-8800: DBaaS: With PMM, you can now have a simplified experience in registering your EKS cluster with kubeconfig generated by `eksctl`. Copy-paste the configuration by selecting Using Amazon Elastic Kubernetes Service (EKS) check-box, and your K8s cluster is registered. For more information, see Documentation.
- PMM-8434: Support for passing PMM Server Public Address as an environment variable while starting the PMM server. For more information, see Documentation.

9.23.3 Improvements

- PMM-9319: PMM and Percona Portal Integration: Synchronized Platform and PMM roles: We have updated PMM permissions to ensure that Administrators of Percona Portal organizations are also granted Admin role in PMM. -
- PMM-9339: Integrated Alerting: The Use TLS option in webhook settings has been renamed to Show TLS setting to better reflect its functionality.
- PMM-9182: Integrated Alerting: Added **Silence All** option for when you want to stop notifications from all alerting rules at once.
- PMM-9164: Integrated Alerting: You can now use an existing rule as a source for new ones instead of using a template.
- PMM-9635: Advisors: Extended security checks to Advisors to cover broader database health checks.
- PMM-9148: QAN: You can now share a link for Query Analytics at the click of a button with the *Copy Link*.
- PMM-8045: DBaaS: With this version of PMM, we have added a warning about the deletion of API keys so that the user is forewarned before deleting the API key.
- PMM-9452: With this release of PMM, we have implemented a simplified password change method for the default admin user using the command line parameter `change-admin-password`.
- PMM-9542: PMM now predefines certain flags that allow users to set all other VictoriaMetrics parameters as environment variables. For more information see Documentation.
- PMM-8794: For consistency, we have implemented a unified 24 hours time format for backup management.
- PMM-9306: VictoriaMetrics has been upgraded to 1.72.0.
- PMM-8412: Grafana has been upgraded to 8.x.
- PMM-9648: With PMM 2.27.0 migrating to Grafana 8.0, the Singlestat Panel has been deprecated and replaced with Stat Panel in Grafana for an enhanced user experience.

9.23.4 Bugs Fixed

- PMM-9797: Fixed an issue where the data on the Home Dashboard was represented incorrectly on the Stat Panel, which could potentially confuse the user.
- PMM-9757: Fixed an issue where the metrics for MongoDB were not being exposed when the database connection was getting lost.
- PMM-9603: PMM and Percona Portal Integration: Fixed “Insufficient access rights” error that was displayed for admin user after connecting PMM server connect to Percona Portal.
- PMM-9323: PMM and Percona Portal Integration: Ensured PMM no longer downloads checks and templates from Percona Portal when the Telemetry option is disabled in the PMM Advanced Settings.

Percona LLC and/or its affiliates ©2024

- PMM-8986: Advisors: Fixed an issue where the Advisors check on the PMM servers monitoring a large number of database services was causing a timeout.
- PMM-9570: DBaaS: Create DB Cluster dialog box was closing automatically while trying to create a database cluster and had to be opened twice. This issue has been fixed now.
- PMM-9496: DBaaS: Fixed an issue where the host and database were not being monitored in DBaaS.
- PMM-9783: QAN: Fixed an issue where QAN failed to work after an upgrade.
- PMM-9661: QAN: Fixed an issue where QAN layout breaks while resizing the window.
- PMM-9797: Dashboard: Fixed an issue where the data on the Home Dashboard was represented incorrectly on the Stat Panel, which could potentially confuse the user.
- PMM-9757: Fixed an issue where the metrics for MongoDB were not being exposed when the database connection was getting lost.
- PMM-9671: Fixed an issue where an upgrade to 2.26.0 failed as PMM crashed.
- PMM-9413: Fixed an issue where the PMM management daemon was getting deadlocked when a PMM agent was getting connected with a duplicate `agent_id`.
- PMM-9015: Fixed an issue where PMM does not display when PostgreSQL is down, thus failing to capture the metrics for PostgreSQL.
- PMM-8203: Fixed an issue where the `pmm-agent.log` is cluttered with unnecessary errors when MariaDB database versions 10.2, 10.3, 10.4, and 10.5 are added for monitoring.
- PMM-5831: Fixed an issue where `pmm-admin` uses the default value of 'listen port' rather than picking up the value from the agent configuration file.
- Fixed the following CVE's:
  - PMM-9726: Fixed a critical CVE that was affecting some versions of Go.
  - PMM-9722: Fixed a CVE for ClickHouse DBMS.
  - PMM-9327: Fixed a vulnerability in the Network Security Services (NSS) package.
  - PMM-9502: Fixed multiple JavaScript Common Vulnerabilities and Exposures (CVE) for PMM AMI setup.

9.23.5 Known Issues

PMM-9992: **Error while using reverse proxy (like Nginx)**

While using a reverse proxy (for example, Nginx) in front of PMM, you can run into the error `origin not allowed` after upgrading to PMM 2.27.0 or newer versions.

Solution

Add the `Host` header to the reverse proxy configuration file.

Example

For Nginx, add the following:

```
proxy_set_header Host $http_host;
```

9.23.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2022-12-19

9.24 Percona Monitoring and Management 2.26.0

Date: February 8, 2022
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.24.1 Release Highlights

- **Experimental Dashboards**

New experimental dashboards are introduced that will be shipped to the PMM users. These dashboards will be uploaded to the **Experimental** folder to enable the users to test them in their environment and provide feedback.

The following Dashboards are being shipped as part of this release:

- K8s monitoring dashboard
- Environment Overview dashboard
- Environment Summary dashboard

> **Important**
>
> These experimental dashboards are subject to change. It is recommended to use these dashboards for testing purposes only.

- **SMTP Configuration Verification**

When configuring an SMTP email server for sending out email notifications, you now have the option to test that your specified settings are correct. The **Email** tab under **Configuration > Settings > Communication** now includes a **Test** button to send a test alert through the specified server. For more information about setting up an SMTP email server, see Configure.

- **Breaking change for Integrated Alerting (Technical Preview)**

This release introduces major changes to the core structure of rule templates. As a result, alert rules and templates created in previous PMM versions are not compatible with PMM 2.26.0 and cannot be migrated to this new version. After upgrading, make sure to manually recreate any custom alert rules and rule templates that you want to transfer to PMM 2.26.0.

> **Disclaimer**
>
> Integrated Alerting is still a Preview functionality and, as such, subject to change. We recommend that early adopters use this feature for testing purposes only.

9.24.2 New Features

- PMM-9059: Following the recently introduced support for connecting PMM to Percona Platform, you now have the option to also unlink any servers that are no longer relevant to your Platform organization. To disconnect a PMM server, go to **Configuration > Settings > Percona Platform** and click **Disconnect**. You can check the list of

Percona LLC and/or its affiliates ©2024

servers connected to an organization in Percona Platform by clicking **View instances** on the Dashboard page. For more information, see Configure for more details.

- PMM-9312: Tech Preview Feature: PMM now captures the MongoDB metrics such as `dbStats`, `collStats`, `indexStats`, and `topmetrics`. See Documentation for more details.

9.24.3 Improvements

- PMM-9176: DBaaS - PMM now supports Percona Distribution for MongoDB Operator 1.10.0.
- PMM-9159: DBaaS - PMM now supports Percona Kubernetes Operator for Percona XtraDB Cluster 1.10.0.
- PMM-9180: Integrated Alerting > Add Alert Rule - Added the Template expression in a collapsible panel for an enhanced user experience (default view as collapsed) as the technical message could confuse the users.
- PMM-7781: Integrated Alerting - Alert rules no longer depend on their source rule template after creation. This means that you can now update or delete rule templates without impacting existing rules that are based on that template. For more information, see Integrated Alerting.
- PMM-9356: Added new experimental Environment dashboards in PMM.
- PMM-9296: Disclaimer about Technical Preview feature added to Percona Platform - Connect PMM to Percona portal page.

9.24.4 Bugs Fixed

- PMM-9416: Upgrading to PMM 2.25.0 using docker (replacing the image) fails when upgrading from versions less than or equal to 2.23.0.

> **Caution**
>
> It is recommended to upgrade directly to PMM 2.26.0 instead of 2.25.0 when updating from versions less than or equal to PMM 2.23.0 if your PMM doesn't have external access (access to `repo.percona.com`).

- PMM-8867: Fixed an issue for PMM Client installation using the tarball script (without using RPMs) where the configuration was getting lost due to the configuration file `pmm-agent.yml` being recreated.
- PMM-8094: DBaaS - Fixed an issue for paused clusters that froze with PSMDB v1.8 operators when all the pods were terminated, providing no cluster resumption option.
- PMM-8535: DBaaS - Repeating error after force unregister
- PMM-9144: Dashboard - Fixed the Add inventory page issue that indicated AWS RDS/Aurora supported only MySQL.
- PMM-9289: **Get from Browser** on settings page does not fetch port in Public address field breaking the integration for Platform authentication.
- PMM-9255: On connecting the PMM server to Percona Platform for an admin user insufficient access rights error message is thrown.
- PMM-9049: Eliminated confusion around the current and available version date by adding a tooltip with an explanation for these dates.
- PMM-9181: Integrated Alerting - Modified the label for the enable/disable button in order to avoid confusion.
- PMM-5405: Fixed an issue where the `pmm-admin summary` command fails if a null value is passed for the `--filename` parameter.
- PMM-8141: Fixed an issue where the metrics were not captured as the cleanup of the temporary folder on the client node deleted the requisite configuration file.

Percona LLC and/or its affiliates ©2024

9.24.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-12-19

9.25 Percona Monitoring and Management 2.25.0

Date: December 14, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

Important note for users considering docker way upgrade to PMM 2.25.0

If you upgrade from a PMM version less than or equal to 2.23.0 using docker, it will fail if your PMM does not have external access (access to `repo. percona.com`). Thus, it is recommended to upgrade to PMM 2.26.0 instead. See PMM-9416 for more details.

9.25.1 Release Highlights

- **Percona Platform (Technology Preview):**
  - Connect Percona Monitoring and Management (PMM) to Percona Platform to boost the monitoring capabilities of your PMM installations and access all your Percona accounts and services from one single, centralized location. For more information, see the Percona Platform Portal documentation
- **Enhanced PostgreSQL monitoring**
  - You can now specify custom database names when adding PostgreSQL Servers for monitoring. Previous PMM versions always used the default `postgres` name instead.
  - Added support for the new version of `pg_stat_monitor` extension. Release Candidate v1.0.0-rc.1 brings many new PostgreSQL metrics, Dashboards and Query Analytics! To find out about all the features available in the new `pg_stat_monitor` version, see the pg_stat_monitor User guide
  - Added compatibility for the latest Percona Distributions for PostgreSQL 14, 13, 12, 11 updated on December 7, 2021, which includes the newest version of the `pg_stat_monitor` extension.
- **Grafana usability enhancements**
  - PMM is now using the native Grafana provisioning mechanism for adding dashboards, plug-ins, and data sources. This ensures faster and more reliable upgrading to newer PMM versions. Your existing plug-ins and dashboard changes are preserved during upgrades, but always make sure to back them up before upgrading and check that everything transferred correctly afterward.
  - Added option to change the time zone on dashboards. This selection is preserved while you navigate over Dashboards. If you need to change this setting permanently for your account, change it in your preferences by the URL: `https://YOUR_PMM_SERVER/graph/profile`

Percona LLC and/or its affiliates ©2024

- **DBaaS (Technical Preview)**
  - You can now deploy and update your DBaaS created PXC clusters to the latest version of Percona Distribution for MySQL Operator 1.9.0. This enables you to take advantage of the latest features and fixes.
- **PMM environment enhancements**
  - The pmm-client docker container can now be started as a sidecar. For users that use PMM client in Kubernetes or build automation around it, you can now start the client as a sidecar container simply by passing a flag. The client will also gracefully handle any instances where the connection to DB is not available. For more details, see the PMM Client documentation
  - Removed support for Ubuntu 16.04. With the support of new products and new versions of already supported products, we also removed old, unsupported software. As of this release, we are no longer supporting Ubuntu 16.04 in PMM according to recent announcements

9.25.2 New Features

- PMM-9050: Connect PMM Server to Percona Platform for additional account info in PMM and value added content

9.25.3 Improvements

- PMM-8545: DBaaS: Support of Percona Distribution for MySQL Operator 1.9.0 in PMM
- PMM-7677: Docker container for `pmm-client`: Option to change behavior and follow sidecar pattern The flags `PMM_AGENT_SIDECAR` and `PMM_AGENT_SIDECAR_SLEEP` does this. Read more in documentation
- PMM-3516: Optimize provisioning mechanism of plugins, dashboards, and data sources in PMM
- PMM-8674: Integrated Alerting: Add Tooltips to `Add Alert Rule` fields to make it easier to understand what information is needed
- PMM-8505: Integrated Alerting: Clarify description of the ‘Low memory’ Alert Template
- PMM-8503: Integrated Alerting: Field validation in Email and Slack tabs when updating settings
- PMM-7527: Integrated Alerting: Improvements to overall user experience for action buttons in Alerting
- PMM-7079: Integrated Alerting: New ‘information’ icon to give additional details about Alerts without cluttering screen
- PMM-8259: Better clarification of error messages in `pmm-admin` when PMM server can’t be unregistered
- PMM-8972: Add ability to specify custom base path to exporters and tools using `pmm-admin` command
- PMM-8282: Improved messaging for TLS option when adding Remote instances in PMM over UI

9.25.4 Bugs Fixed

- PMM-9169: Security Advisor Checks are not working for MongoDB instances
- PMM-8982: Backup Management: User is not able to see MongoDB backup logs if backup was taken on older version of pmm-server
- PMM-9157: Dashboards: Changing the timezone on dashboards does not persist navigation
- PMM-7116: Dashboards: Incorrect STARTUP state on MongoDB ReplSet Summary dashboard
- PMM-8993: Integrated Alerting: Sending email using Gmail fails
- PMM-7802: PMM can’t monitor MongoDB arbiter nodes (Thanks to Artem Meshcheryakov for reporting this issue)
- PMM-6937: Can’t add PostgreSQL instances to PMM without `postgres` DB in PostgreSQL server (Thanks to Daniel Kowalewski for reporting this issue)
- PMM-7447: Can’t add into PMM instances of PostgreSQL with `SCRAM-SHA-256` authentication

Percona LLC and/or its affiliates ©2024

- PMM-9085: PMM Server crashes after upgrading to 2.22 every 4 hours
- PMM-9156: `pmm-agent` paths-base option not working for `pmm2-client` binary installation in PMM 2.23.0
- PMM-8461: DBaaS: Confusing error when accessing DBaaS pages when it's disabled
- PMM-8110: DBaaS: Registering K8s cluster with operators already installed can cause error
- PMM-8694: Query Analytics: URLs in Query Analytics with a selected query and a timestamp range does not select the query
- PMM-9227: Pagination Reset on QAN after Time Range change doesn't work, results in wrong results
- PMM-9298: PMM AMI image in 2.24.0 has only 8GB space for data and Volume Size Check fails while upgrading to 2.25.0

Important note for users of PMM 2.24.0

2.24.0 AMI image has only 8GB available for the data, it is a bug (see PMM-9298). To resize a disk to full size you need to login to AMI instance with SSH and use the following command:

```
curl https://raw.githubusercontent.com/percona/pmm-update/main/ansible/playbook/tasks/create-lvm.yml -o lvn-fix.yml && sudo ansible-playbook lvn-fix.yml
```

For instructions about how to access your instances by using an SSH client, see Connecting to Your Linux Instance Using SSH Make sure to replace the user name ec2-user used in this document with admin.

What this command does:

1. Downloads Ansible playbook and runs it
2. Copy your data from /srv to the temporary directory
3. Create lvm partition
4. Copy data from system disk to a new LVM partition

9.25.5 Known issues

- PMM-9255: After connecting PMM to Percona Platform, PMM occasionally shows a false permission issue notification, incorrectly suggesting that the connection could not be established due to missing permissions. Reload the page to remove the incorrect notification and confirm the connection.
- PMM-9312: It's not possible to enable `collStats`, `indexStats` and `-max-collections-limit` for MongoDB

9.25.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-18

9.26 Percona Monitoring and Management 2.24.0

Date: November 18, 2021
Installation: Installing Percona Monitoring and Management

Percona LLC and/or its affiliates ©2024

Important note for users of PMM 2.24.0

2.24.0 AMI image has only 8GB available for the data, it is a bug (see PMM-9298). To resize a disk to full size you need to login to AMI instance with SSH and use the following command:

```
curl https://raw.githubusercontent.com/percona/pmm-update/main/ansible/playbook/tasks/create-lvm.yml -o lvn-fix.yml && sudo ansible-playbook lvn-fix.yml
```

For instructions about how to access your instances by using an SSH client, see Connecting to Your Linux Instance Using SSH Make sure to replace the user name ec2-user used in this document with admin.

What this command does:

1. Downloads Ansible playbook and runs it
2. Copy your data from /srv to the temporary directory
3. Create lvm partition
4. Copy data from system disk to a new LVM partition

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.26.1 Release Highlights

- **Enhanced PostgreSQL monitoring**
- Beginning with this release, PMM now supports monitoring of PostgreSQL 14, both - Community edition and Percona Distribution for PostgreSQL.
- We've made significant improvements in exposed data and added additional features if you monitor PostgreSQL together with the pg_stat_monitor extension (Part of Percona distribution for PostgreSQL). These features include:
  - The PostgreSQL queries will have complete Query Execution Plan information. This will help with future optimization of queries and give a clear understanding of query performance
  - Query execution histograms collection and presentation inside Query Analytics for a much more detailed understanding of query executions.
  - Query analytics will both show and let the user drill down to the Top Query of the particular query if it's a subquery and have this parent query. This feature will allow users to see the dependencies between queries better and understand the impact of subqueries.
  - Query Analytics can filter PostgreSQL queries by query commands like SELECT, UPDATE, etc., and by Application Name if it's set for PostgreSQL connection from the application.
- **Integrated Alerting (Technical preview):**
  - Alerting in PMM now has an additional notification channel type - webhooks. So now, users can integrate Alerting with any tool they use for Incident management. Read more about new notification channels and how to set them up in our documentation

9.26.2 New Features

- PMM-8027: Integrated Alerting: New notification channel added: Webhooks
- PMM-8301: Add data collection support and visualization for newly added dimensions in pg_stat_monitor such as Application Name,Top Query, Plan in Query Analytics
- PMM-8588: PostgreSQL Histograms added to QAN when using pg_stat_monitor extension
- PMM-8632: New Filter: "Command Type" allows filtering queries based on type (SELECT, INSERT, UPDATE, DELETE, n/a) when pg_stat_monitor extension enabled

Percona LLC and/or its affiliates ©2024

9.26.3 Improvements

- PMM-8803: Backup Management: Improved error messages to indicate incompatible versions of software
- PMM-8636: Integrated Alerting: Additional context to alerts to better convey issue detected
- PMM-8644: Integrated Alerting: API should allow textual TLS configurations for webhooks
- PMM-8122: Integrated Alerting: UI does not indicate a port is needed in configuration for SMTP communication channel
- PMM-8484: Added support for PostgreSQL 14 and Percona Distribution for PostgreSQL 14
- PMM-7297: Updated plugin for ClickHouse data source from 2.1.0 to 2.3.1. This fixes some bugs and eliminates noise from warnings in logs as well as adding support of new types (`DateTime64`) and improved ARRAY JOIN parsing

9.26.4 Bugs Fixed

- PMM-8975: Backup Management: long presentation of recurrent intervals in Backup scheduling
- PMM-8541: Navigating through PMM Settings link at Failed security checks panel takes more than 30 seconds
- PMM-8387: MySQL InnoDB Details dashboard is not in the left menu
- PMM-8858: Dashboards: No Host uptime on Homepage for RDS instances
- PMM-8611: Dashboards: PMM Agents status presented as DOWN while there is no recent data yet on the status
- PMM-8393: Integrated Alerting: Alert rules not executed after upgrading PMM Server running as Docker container
- PMM-8058: Integrated Alerting: Firing alerts disappear after PMM server restart
- PMM-8089: PMM is not exposing data for memory used by MongoDB when it's mapped with the journal. This was inconsistent behavior compared to older versions of PMM.
- PMM-9100: Dashboards: Binary Log related metrics on MySQL Replication dashboard are not prevented and not collected for MySQL 8
- PMM-8633: Unfinished queries are included in Query Analytics for PostgreSQL with pg_stat_monitor usage because of incorrect use of `state_code`.
- PMM-8859: Increased memory consumption on Client-side for PostgreSQL monitoring when executing either too many custom queries or some queries against too many tables
- PMM-9046: Incorrect link to instructions about installing Image Rendering Plugin
- PMM-8952: Query Analytics: No table/indexes information for Views when PostgreSQL server monitored with pg_stat_monitor

9.26.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-12-19

9.27 Percona Monitoring and Management 2.23.0

Date: October 21, 2021

Percona LLC and/or its affiliates ©2024

Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.27.1 Release Highlights

- **Grafana Upgrade** Embedded Grafana version was upgraded from 7.5.7 to 7.5.11 to improve some Grafana instrumentation metrics. Also, the upgrade is delivering a security fix that impacted Grafana. Please upgrade to the latest version of PMM ASAP. Read more about CVE issue here
- **Backup Management (Technical preview):**
  - An ability to view logs of the backup process for better visibility over the backup process
  - An ability to schedule Point-In-Time-Recoverable backup from MongoDB clusters with the correct configuration. Note: there is no UI to restore PITR for MongoDB at the moment. It will come with future releases, but it is possible to restore a PITR backup with Percona Backup for MongoDB manually
- **DBaaS (Technical preview):** From this release on, PMM users who are using the DBaaS feature will be able to update versions of their DBaaS controlled Databases by the push of a button according to each DB's compatibility matrix. Please note that we recently found a bug PMM-8723 that was causing significant problems with DBaaS usage. This bug was fixed in this release, and no additional actions will be required.

9.27.2 New Features

- PMM-8269: Backup Management: Ability to schedule Point-In-Time-Recoverable backups for MongoDB
- PMM-8159: Backup Management: Ability to see a logs of backup process for MongoDB
- PMM-7519: Backup Management: Version compatibility check prior to attempted MySQL data restoration
- PMM-8200: DBaaS: Admin can now initiate a DB version upgrade with just a button click
- PMM-8273: Integrated Alerting: Alert templates delivery from Percona.com for anonymous PMM servers

9.27.3 Improvements

- PMM-8973: Grafana upgrade from 7.5.7 to 7.5.11 Includes better Grafana instrumentation metrics and fix for CVE-2021-39226 (read more on Grafana blog)
- PMM-8653: Added titles to Home Dashboard panels for better readability
- PMM-8669: Integrated Alerting: Create a clearer distinction about using PMM Alerting as preferred method vs using an external Alertmanager
- PMM-8539: Wrong Cluster Role presentation on MongoDB Cluster Summary
- PMM-7559: Integrated Alerting: Improve error message when trying to delete a channel that is used by a rule
- PMM-6763: Better color contrast in Time distribution in QAN details
- PMM-5669: New flag –paths-base in pmm-agent to avoid problems with hard-coded paths. Please note: this is possible if you run pmm-agent separately from pmm-admin. The ability to specify base paths over pmm-admin is not yet implemented

9.27.4 Bugs Fixed

- PMM-7985: Users were losing manually installed Grafana plugins after upgrade via Docker

Percona LLC and/or its affiliates ©2024

> **Caution**
>
> The issue is fixed automatically since 2.23.0 version forward. For the upgrades from versions before 2.23.0 please backup plugins first.

- PMM-8767: Copied dashboards with tags were ending up in unexpected folder after upgrade
- PMM-8635: MyRocks WAL panel from MySQL MyRocks Details Dashboard presented data in wrong units
- PMM-8527: Dashboards: ProxySQL/HAProxy DB Conns, DB QPS, DB uptime metrics were missing on Home dashboard panels
- PMM-8749: Adding more than 1 `mongos` was breaking MongoDB Cluster Summary dashboard
- PMM-8004: Fixed broken metrics reporting in case of lost connection to MongoDB. (Thanks to Álvaro López López for reporting this issue)
- PMM-8489: Failed to get topology labels when target server is `mongos`
- PMM-6877: Fixed error flooding from when monitoring mongos (Thanks to Clyde Shillingford for reporting this issue)
- PMM-8851: Can't monitor GCP Cloud SQL or other PostgreSQL with custom SSL certificates (Thanks to Jyoti Prakash for reporting this issue)
- PMM-8646: PostgreSQL services monitoring was stalled after intermittent connection latency
- PMM-8723: PMM wouldn't restart DBaaS functionality and would break it after upgrade via UI. Affecting versions starting from 2.17.0

9.27.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-18

9.28 Percona Monitoring and Management 2.22.0

Date: September 23, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.28.1 Release Highlights

- **DBaaS (Technical preview):** DBaaS users can now use the PMM UI to upgrade existing Clusters to the newer version of the operator without interacting directly with Kubernetes.

9.28.2 New Features

- PMM-8396: Backup Management: Add an ability to specify the number of retries for Scheduled backups before operation fails

Percona LLC and/or its affiliates ©2024

- PMM-8079: DBaaS: Percona Operators inside Kubernetes cluster managed by PMM can now be upgraded
- PMM-8077: DBaaS: Show the current version of Operators used in Kubernetes Cluster
- PMM-7924: MySQL Performance Details dashboard: Add “Performance Schema Status Monitoring” chart

9.28.3 Improvements

- PMM-8688: Backup Management: When adding a new Backup, the action button now reads “Create Backup” instead of “Edit” to reflect the action happening
- PMM-8311: Integrated Alerting: Disable edit/delete buttons for Percona-sourced Templates
- PMM-8509: Management of ability to update PMM Server in the same way as this implemented to other Settings for PMM. Users can use API, UI, or docker Environment Variables to change the setting responsible for the Update process. As with all PMM settings, environment variables have higher priority and can’t be changed with the API or in the UI.
- PMM-7392: DBaaS: Change Number of Nodes when editing Topology

9.28.4 Bugs Fixed

- PMM-8613: Backup Management: Unable to restore MySQL backup
- PMM-8463: Backup Management: State stuck on “Pending” when creating backup with already existing name
- PMM-8408: DBaaS: Development version of PMM Client was used for monitoring DB Clusters created by DBaaS
- PMM-8584: Wrong CPU metric labels in dashboards for RDS instances
- PMM-8421: Listen-port ignored/removed for external services after server update to PMM 2.19 and higher (Thanks to Rainer Plischke for reporting this issue). **Please make sure to upgrade PMM Server to avoid loss of external exporter listen port (PMM-8829) and always upgrade PMM Server before PMM Client (PMM-8854).**
- PMM-8703: Custom dashboard prevents PMM Server Docker update from 2.20 to 2.21 (Thanks to Hubertus Krogmann for reporting this issue)
- PMM-7690: AWS discovery and monitoring based on IAM roles is not working

If you have problems upgrading your `pmm-2-client` packages, try clearing caches with:

```
sudo apt-get clean
```

and remove files manually with:

```
cd /var/cache/apt/archives && sudo rm -rf ./*
```

9.28.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-18

Percona LLC and/or its affiliates ©2024

9.29 Percona Monitoring and Management 2.21.0

Date: August 26, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.29.1 Release Highlights:

- **Custom certificates support**: We have added support for custom TLS certificates for remote monitoring of PostgreSQL and MongoDB services, configurable on the command line with `pmm-admin` or through the UI.
- **Backup scheduling with retention** (Technical Preview): When scheduling a backup you can now specify how many of the most recent backups to keep. Backups not in this range are automatically deleted to free space for new ones.
- New supported versions:
  - DBaaS functionality now supports Kubernetes Operator for MongoDB version 1.9.0.
  - PMM Client packages now support Debian 11 (“Bullseye”).

9.29.2 New Features

- PMM-8158: Backup Management: Delete option — When deleting a backup from Backup Management inventory a new option lets you also delete the data file from storage.
- PMM-8156: Backup Management: Retention — You can now define how many of the most recent scheduled backups to keep.
- PMM-8214: Ability to collect Kubernetes cluster metrics — Collection only, metrics are not currently presented on any PMM dashboard.
- PMM-7477: Support custom TLS certificates when monitoring remote MongoDB instances
- PMM-7888: Custom TLS certificates now allow SSL connections to PostgreSQL instances (Thanks to Jyoti Prakash for reporting this issue)

9.29.3 Improvements

- PMM-8267: Backup Management: Active progress indicator
- PMM-8549: Backup Management: Show loading status on delete window
- PMM-8542: Backup Management: Inform that times should be entered in UTC timezone format
- PMM-8316: DBaaS: PSMDB 1.9 operator support —For what’s new see release notes.
- PMM-7612: Integrated Alerting: Validate communication settings ‘From’ email address format
- PMM-7570: Specify Custom Basic Auth password for Agents when adding Services
- PMM-8560: Add support for Debian 11 (“Bullseye”) to `pmm-client` package
- PMM-7087: Rename custom query file to `example-queries-postgres.yml` and include warning that the file will be overwritten on upgrade; user should create a copy with a new name to prevent losing metrics collection on future upgrades. (Thanks to Daniel Guzman Burgos for reporting this issue)
- PMM-8568: Use latest CentOS patches for creating OVA, AMI and Azure images
- PMM-5291: Update ClickHouse version from 19.7.5.27 to 21.3-lts
- PMM-8091: Collect and present additional ProxySQL metrics taken from `runtime_mysql_servers` table

Percona LLC and/or its affiliates ©2024

9.29.4 Bugs Fixed

- PMM-8616: Backup Management: No ‘Delete from storage’ action on backup inventory
- PMM-8543: Backups are not visible after PMM Server upgrade
- PMM-8458: Backup Management: Inconsistent auto-fill of ‘Vendor’ field with on-demand backup
- PMM-8404: Dashboard image rendering plugin renders image that includes error message
- PMM-7286: Query Analytics can’t handle colon character (`:`) in service names (Thanks to Daniel Guzman Burgos for reporting this issue)
- PMM-7278: `mongo_exporter` fails to authorize when MongoDB running with `authMechanism=MONGODB-X509` (Thanks to Lena D for reporting this issue)
- PMM-8307: Default configuration limits for allowed connections prevents monitoring large numbers (500+) of DB servers
- PMM-2168: `rds_exporter` not stopped when all RDS instances are removed or disabled
- PMM-8219: PMM Server update panel “Current version” empty if no internet connectivity
- PMM-8559: Unauthorized error appears while logging in

9.29.5 Known Issues

- Failure to upgrade when using a dashboard with custom tags.

Important

In some cases users may not be able to complete the upgrade to 2.21.0 and we have linked this back to dashboards with custom tags. This is to be fixed in our upcoming 2.22.0 release but there are steps (more in the ticket) that you can take if you’re already impacted by this:

```
curl -LJOs https://raw.githubusercontent.com/percona/pmm-server/c2e92bc3aec123affda5f1992c96c95ac74f4a2d/import-dashboards.py
docker cp import-dashboards.py pmm-server:/usr/share/percona-dashboards/
docker exec -it pmm-server chmod a+x /usr/share/percona-dashboards/import-dashboards.py
```

9.29.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-18

9.30 Percona Monitoring and Management 2.20.0

Date: August 3, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

Percona LLC and/or its affiliates ©2024

9.30.1 Release Highlights

- The Easy-install script for PMM Server is checksum verified.
- PMM will use Grafana 7.5 instead of 7.3. We are also preparing for Grafana 8.
- PostgreSQL monitoring with the `pg_stat_monitor` plugin enabled exposes new metrics such as Plan Times, WAL Fpi/Bytes/Records.
- For users who deploy PMM Server through the AWS Marketplace, AWS RDS service discovery will be executed without AWS credentials and tuning IAM roles.
- For Backup Management (Technical Preview), we added the ability to schedule backups so you can schedule and see already scheduled backups in the UI.

9.30.2 Important note for users of PMM who started out using the Docker image of 2.16.0

If you installed PMM version 2.16 as a new Docker image and have since used the home dashboard upgrade widget to upgrade to any of 2.17, 2.18, or 2.19, you might experience problems with monitoring the PMM server itself, Remote Monitoring, or RDS/Azure monitoring. If you experience any of these problems, you can simply run the following commands to get your instance working and it will be automatically resolved in the next release:

1. Enter the container: `docker exec -it pmm-server bash`
2. Roll back `pmm2-client` package to stable version: `yum downgrade -y pmm2-client`

Alternatively, you can replace the existing Docker container with a fresh install of the latest release by following the official instructions for an upgrade. (This will guide you through taking a backup of your PMM Server and restoring it after installing a fresh docker image of PMM Server.)

9.30.3 New Features

- PMM-8157: Backup Management: Show scheduled backups – A new view that shows the list of scheduled backups with quick actions to enable/disable, edit, copy, and delete.
- PMM-8155: Backup Management: Scheduling of backups – Support for Backup Scheduling has been added so that users can define backup jobs to run automatically in the future with the option of making the schedules recurring.
- PMM-7010: Option to unregister current node (`pmm-admin unregister`)

9.30.4 Improvements

- PMM-7552: PMM Server Docker image: Add labels to PMM Server Docker image (Thanks to Nicolas for reporting this issue)
- PMM-8266: PMM Server Docker image: Decommission and remove Prometheus binaries and configuration
- PMM-8040: PMM Server Docker image: Remove yum cache to reduce size of image
- PMM-7809: Grafana upgrade from 7.3.7 to 7.5.7 – Read more at grafana.com
- PMM-8386: Overview text on the Home page is missing PostgreSQL as a supported technology
- PMM-7722: DBaaS: Announce new supported version of operator – Shows that a new version of the operator is available.
- PMM-6278: Modification of MySQL ‘Delete’ queries to provide ‘Explain’ information
- PMM-8468: Forbid the use of outdated ciphers for HTTPS protocol on exporters
- PMM-7649: Security Checks: Show “Insufficient access permissions” in UI for non admin users
- PMM-8059: Update Query Analytics UI to clarify estimated results on MySQL ‘explain’ response where we modified original query

Percona LLC and/or its affiliates ©2024

- PMM-8043: Return Service Name in `GetCheckResults` API response
- PMM-8000: Expose new numbered metrics available in `pg_stat_monitor` 0.9

9.30.5 Bugs Fixed

- PMM-8299: Backup Management: Remove storage location shows wrong notification – When removing a storage location, PMM presents misleading information to the user in the form of two notification messages for both error and success.
- PMM-8283: Backup Management: Error when removing location with 'force delete' option
- PMM-8064: Dashboards: Size of Temp Files Report Metric value has wrong unit on PostgreSQL Instance Summary Dashboard
- PMM-6981: Dashboards: Wrong version is shown for MariaDB services
- PMM-7738: Integrated Alerting: Alerts for some built-in templates missing service name label
- PMM-6877: `mongodb_exporter` doesn't recognize being on a mongos host and fills the syslog with `replSetGetStatus` errors (Thanks to Clyde Shillingford for reporting this issue)
- PMM-7627: Consistent PMM Server log management – Adds consistency to the log management of `nginx`, `postgresql` and `clickhouse-server`, which is now delegated to `supervisord`. Removes the `logrotate` daemon from the image.
- PMM-8492: PMM Client version is 2.21.0 inside PMM Server after upgrade from 2.16.0

9.30.6 Known Issues (unfixed problems that you should be aware of)

- PMM-8414: Backup Scheduler not working if user specifies explicit job start time

9.30.7 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-12-19

9.31 Percona Monitoring and Management 2.19.0

Date: June 30, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

Visit our forum to comment on this release.

Percona LLC and/or its affiliates ©2024

9.31.1 Release Highlights

- Backup Management can now be enabled from the UI. Go to *Configuration → Settings → Advanced Settings*, and you will see it in the *Technical Preview* section. We also added support for MongoDB services on-demand backup and restore. For now, it only supports ReplicaSet on S3-compatible storage.
- Dashboards improvements
  - There are several community-driven improvements to ProxySQL data collection, with new dashboards to expose such metrics like: *Queries Latency histograms* and `SHUNNED_REPLICATION_LAG` state.
  - Fixes for Amazon Aurora service detection on the dashboard, MongoDB ReplicaSet Summary, and other MongoDB memory-related panels.
- Improvements to DBaaS secrets by generating strong passwords for operators. This is an improvement to the Automated Operator Installation released in PMM 2.18, which will greatly enhance security.

9.31.2 New Features

- PMM-7639: Backup Management: Ability to remove Backup Location even if there are some backup artifacts on it
- PMM-7567: Backup Management: Simple backup for MongoDB ReplicaSet
- PMM-7568: Backup Management: Simple restore for MongoDB ReplicaSet

9.31.3 Improvements

- PMM-8112: Dashboards: Collect and present histograms from ProxySQL on Queries Latency dashboard (Thanks to foosinn for helping with this improvement)
- PMM-8081: Dashboards: Add collection and presentation for `SHUNNED_REPLICATION_LAG` state to `proxysql_connection_pool_status` (Thanks to spaceform for helping with this improvement)
- PMM-7584: Components Upgrade: VictoriaMetrics 1.53.1 to 1.60.0
- PMM-8001: Better error handling when `pg_stat_monitor` is an unsupported version
- PMM-7659: DBaaS: Ability to specify the type of connection for DBaaS cluster during DB Cluster creation
- PMM-7828: DBaaS: Select Database Type by default if only one operator is installed
- PMM-8153: Backup Management: Disable ‘Restore’ button for backups whose service has been deleted

9.31.4 Bugs Fixed

- PMM-7194: ‘Share with Percona’ option doesn’t export data from collapsed panels
- PMM-8060: User cannot add remote instances after OVF/AMI upgrade (previously was Known Issue)
- PMM-7104: Slowlog rotation by pmm-agent causing additional unexpected rotation of binary logs (Thanks to Arthur Tokarchuk for reporting this issue)
- PMM-8125: Error of monitoring GCP based PostgreSQL because of internal database `cloudsqladmin`
- PMM-6778: Can’t specify custom labels during node addition/configuration
- PMM-8090: Multi-request protection breaks metrics gathering (Thanks to Francisco Miguel Biete for fixing this bug)
- PMM-5248: InnoDB TableSpace data is not collected for Percona Server 8
- PMM-4665: User’s log file contains error messages about TokuDB and heartbeat despite not being used
- PMM-8014: Error when adding Amazon RDS MySQL with TLS over API
- PMM-3352: Low file descriptors limit (1024) with AMI or OVF images causes errors

Percona LLC and/or its affiliates ©2024

- PMM-7948: `pmm-admin list` reports the wrong Node for External Services
- PMM-6295: Unclear/Incorrect statuses of the pmm-agent when Agent or PMM server went down (Thanks to Mikhail Solovyev for reporting this issue)
- PMM-5917: pmm-agent moves slow logs without checking privileges
- PMM-8021: “Query Analytics” misspelled on left side menu
- PMM-5283: Inconsistency with lengths of Example and Fingerprints in Query Analytics
- PMM-8121: Error message and help for “Remove Service” command is not helpful
- PMM-8196: Additional spaces in email/passwords fields on Sign up/Login pages causes Authentication problems
- PMM-8009: Long First/Last names causes errors when used in Register/Login form
- PMM-8220: Dashboards: Active Time Series Changes on Victoria Metrics dashboard report no data
- PMM-8202: Dashboards: No Amazon Aurora services are available on MySQL Amazon Aurora Details dashboard for selection
- PMM-8085: Dashboards: Wrong units are used in MongoDB dashboards on memory-related panels
- PMM-7154: Dashboards: No data on some panels from MongoDB ReplSet Summary dashboard
- PMM-8115: DBaaS: Delete PSMDB cluster action takes too long
- PMM-7737: DBaaS: Replace all default passwords in operator secrets during installation
- PMM-7970: DBaaS: Confusing message for Cluster name pattern on DB Cluster creation screen
- PMM-7755: DBaaS: Clusters with longer name not initializing
- PMM-7528: DBaaS: Error after Kubernetes cluster destroyed or removed externally to DBaaS
- PMM-8013: Backup Management: Unable to get Backup Artifact list after Service removal

9.31.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-01-11

9.32 Percona Monitoring and Management 2.18.0

Date: June 1, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

Percona LLC and/or its affiliates ©2024

9.32.1 Release Highlights

The goal for this small release was to eliminate a lot of bugs and complete some features.

- DBaaS

  Added the ability for PMM to install the latest versions of the K8s operator into the K8s cluster. There is no longer any need to install the operator manually. Just connect PMM to your K8s cluster and let PMM do the rest.

- Backup Management

  Backup functionality was released as a Technical Preview feature and will require specific prerequisites from the user side to be installed. Currently, PMM will allow you to:

  - manage storage for backups (S3 only);
  - execute a backup for MySQL and Mongo instances;
  - restore a MySQL backup to the same instance from where it was taken (via the UI).

  Restore in other cases is not yet implemented on the UI.

9.32.2 New Features

- PMM-7509: Integrated Alerting: Pagination for Alerts list

9.32.3 Improvements

- PMM-8029: DBaaS: PSMDB 1.8 operator support
- PMM-7548: Integrated Alerting: Disable edit and delete buttons for templates manually staged by user directly on the file system

9.32.4 Bugs Fixed

- PMM-8053: Better error handling for non-admins who try to access the settings page
- PMM-7941: Wrong replication status for MongoDB Replica Set
- PMM-7302: Webhook usage with `max_alerts` attribute for Alertmanager configuration causes errors for PMM
- PMM-7224: Instance Overview dashboards behave inconsistently
- PMM-6864: MongoDB Oplog Recovery Window dashboard is broken (Thanks to Clyde Shillingford for reporting this issue)
- PMM-7910: MongoDB Query metrics stops being collected if the cursor is failed once (Thanks to Yann Rouillard for reporting this issue)
- PMM-6451: Passing parameters between Query Analytics and Dashboards is broken
- PMM-5368: Unclear message “Failed to get PMM Server parameter” after configuration (Thanks to Martin Wittwer for reporting this issue)
- PMM-5135: Query Example is often empty for MySQL 8+ (Thanks to Mikhail Solovyev for reporting this issue)
- PMM-8083: Better configuration file checking during configuration
- PMM-7958: Databases cannot be deleted while PostgreSQL is being monitored
- PMM-6553: Slow log size units are not defined in help
- PMM-5931: Graph and values in Query Analytics are identical for TOTAL and case when data is Not Available
- PMM-5538: Heavy Load with Distinct Queries on Slowlog enabled could cause no data being reported
- PMM-8095: The link to the Community section in the PMM footer is broken

Percona LLC and/or its affiliates ©2024

- PMM-7982: Query Analytics: Sorting element for the columns is hard to access
- PMM-6676: Terms and Privacy pages opened in the current tab complicates registration process
- PMM-6552: Do not register with server if configuration file fails to create on client
- PMM-6505: Inconsistent style for error messages on Add RDS instance page.
- PMM-8069: Integrated Alerting: Alert template now accepts `.yaml` extension in addition to `.yml` when manually staging on the file system
- PMM-7673: Integrated Alerting: Actions column is transparent
- PMM-7916: DBaaS: Wrong required resources when editing a cluster
- PMM-7753: DBaaS: Edit DB Cluster shows wrong values by default
- PMM-7184: DBaaS: Connection column showing different values after deleting DB cluster
- PMM-8088: DBaaS: In case of error, the kubeconfig file is left in the system

9.32.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.33 Percona Monitoring and Management 2.17.0

Date: May 11, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.33.1 Release Highlights

- Custom certificates help define proper security levels for remotely monitored MySQL instances, including Google Cloud SQL.
- Usability improvements to the External Monitoring UI. When filling parameters, you can enter the parts of an endpoint (scheme, host, path) or let PMM automatically extract them from a URL.
- pg_stat_monitor 0.9.0 support. This change will give you compatibility with the latest version. Support for new features will be in an upcoming release.
- Single-line install of PMM Server on supported Linux distributions (this feature is in Technical Preview).
- DBaaS Changes: (this feature is in Technical Preview)
  - It is easier to experience DBaaS functionality; you can quickly turn it ON/OFF in *Advanced settings* on the *Settings* page. (Read more)
  - Database components management will enable PMM administrators to limit users in your organization to specific (admin-approved) database versions in their DBaaS DB Clusters.
  - For PXC clusters created using DBaaS, HAProxy will now be used by default. Please note: Monitoring of the HAProxy in DBaaS will be enabled in an upcoming release.

Percona LLC and/or its affiliates ©2024

- Changes to *Sign in to Percona Platform*. From this release, Registration of the Percona account will be more secure and require additional confirmation.

9.33.2 New Features

- PMM-7863: DBaaS: Ability to specify in K8s configuration the version of HAProxy to be used for DB creation
- PMM-7848, PMM-7847, PMM-7421: Add support for using SSL certificates between pmm-admin and monitored MySQL databases
- PMM-7883: Single-line install of PMM Server on supported Linux distributions - [Technical Preview]
- PMM-7013, PMM-7819: DBaaS: Use HAProxy by default instead of ProxySQL for MySQL DB clusters
- PMM-7356, PMM-7581: DBaaS: Management of available versions of DB components
- PMM-7358, PMM-7576: DBaaS: Management of default versions of DB components

9.33.3 Improvements

- PMM-7572: Add TLS options to mysqld_exporter
- PMM-7783: Support of pg_stat_monitor 0.9.0
- PMM-7064: Integrated Alerting: Presenting severity of the Alert Rule using different colors
- PMM-7946: Better error message on PMM client if server doesn’t support HAProxy
- PMM-7932: Usability improvements on UI for adding External Services
- PMM-7641, PMM-7820: Add DBaaS to Technical Preview section and allow user to Enable/Disable via UI
- PMM-7966: Telemetry: Collect enabled/disabled status for Integrated Alerting and Security Threat Tool features

9.33.4 Bugs Fixed

- PMM-7911: DBaaS: Invalid Number of Nodes results in an annoying error message pop-up
- PMM-7884: DBaaS: Fix DB Cluster tab loading
- PMM-7917: PostgreSQL exporter has high CPU usage during Restart
- PMM-8037: User can create a Percona Platform account without proper confirmation
- PMM-7702: DBaaS: Cannot edit already-created PSMDB clusters
- PMM-7991: MySQL Summary panel doesn’t exist on MySQL Summary dashboard
- PMM-7939: Inconsistent format of version reporting in pmm-admin
- PMM-7920: PostgreSQL Exporter has increased memory usage with pmm-client 2.15.1 & pmm-server 2.16.0
- PMM-7700: Integrated Alerting: Rule API crashing with more than two parameters or invalid values
- PMM-7616: Integrated Alerting: Incorrect title of the page in a browser
- PMM-7396: Integrated Alerting: Alerts tab error if user deletes Alert Rule which has Firing alerts

9.33.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2021-06-25

9.34 Percona Monitoring and Management 2.16.0

Date: April 15, 2021

Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.34.1 Important note for users of PMM 2.16.0

If you started using PMM from version 2.16 and have already upgraded to 2.17, 2.18, or 2.19, you might have some problems with PMM Server monitoring, Remove Monitoring, or RDS/Azure monitoring. If you experience a problem, we recommend you upgrade and replace the Docker container by following the official instructions for an upgrade here: https://www.percona.com/doc/percona-monitoring-and-management/2.x/setting-up/server/docker.html#upgrade.

If you can't do this, then you need to perform additional steps after upgrading to 2.20.

1. Enter the container: `docker exec -it pmm-server bash`
2. Roll back `pmm2-client` package to stable version: `yum downgrade -y pmm2-client`

9.34.2 Release Highlights

- **Amazon RDS PostgreSQL monitoring**

  AWS monitoring in PMM now covers PostgreSQL RDS and PostgreSQL Aurora types. PMM will include them in a Discovery UI where they can be added which will result in node related metrics as well as PostgreSQL database performance metrics. Before this release, this was available only to MySQL-related instances from Amazon RDS.

- **Azure Discovery and Node metrics extraction**

  Technical Preview: PMM will have the same level of support for Microsoft Azure Database as a Service (DBaaS) as we have for AWS's DBaaS (RDS/Aurora on MySQL or PostgreSQL). You will be able to easily discover and add Azure databases for monitoring by PMM complete with node-level monitoring. This feature is available only if you explicitly activate it on the *PMM Settings* page. Deactivating it will not remove added services from monitoring, but will just hide the ability to discover and add new Microsoft Azure Services.

  (This is a feature technical preview because we want to release it as soon as possible to get feedback from users. We are expecting to do more work on this feature to make it more API and resource efficient.)

- **Security Threat Tool Scheduling - Manage execution and execution intervals**

  Security Threat Tool users are now able to control the Security Check execution time intervals for groups of checks, move checks between groups, and disable individual checks if necessary.

- **Support for pg_stat_monitor 0.8**

  Added compatibility with pg_stat_monitor plugin v 0.8.0. This is not exposing the new features for the plugin in PMM yet, but ensures Query Analytics metrics are collected to the same degree it was with version 0.6.0 of the plugin.

- **Consistent support of Technical Preview Features**

  Reworked the PMM Settings page to make it clear what features are in Technical Preview vs General Availability (GA) and to simplify activation/deactivation of technical preview features. We also provide a better definition of what a Technical Preview is.

Percona LLC and/or its affiliates ©2024

- **Migration of Settings and other service pages in PMM from Grafana dashboards**

  The *PMM Settings* page and several others (including *Add Instance* and *Inventory*) are being converted to Grafana pages and will no longer be presented as dashboards. Additionally, we're moving the menu to the sidebar navigation for consistency and more flexibility compared to the older menu structure.

- **Integrated Alerting improvements**

  We released the next stage of improvements in Integrated Alerting functionality of PMM to simplify the usage of the feature. Together with improvements, we continue fixing known bugs in this feature.

- **[DBaaS] Resource planning and prediction (Resource calculator)**

  Technical preview: While creating a DB cluster a user can see a prediction of the resources this cluster will consume with all components as well as the current total and available resources in the K8s. Users will be warned that if they attempt to create a DB cluster it may be unsuccessful because of available resources in the K8s.

- **[DBaaS] PSMDB 1.7.0 operator support**

  DBaaS in PMM will be using the recently-released Percona Kubernetes Operator for Percona Server for MongoDB 1.7.0 to create MongoDB clusters.

9.34.3 New Features

- PMM-7313, PMM-7610: Ability to discover and monitor Amazon RDS PostgreSQL instances with collecting PostgreSQL and RDS node metrics (Thanks to Daniel Guzman Burgos for reporting this issue).
- PMM-7345: Expose metrics for all available databases on a PMM monitored PostgreSQL server.
- PMM-7344: Update postgres_exporter version from 0.4.6 to 0.8.0. (See the full list of improvements in the changelog.)
- PMM-7767, PMM-7696: Implement feature flag to enable Microsoft Azure monitoring. Users can use the UI or set an environment variable (`ENABLE_AZUREDISCOVER=1`) during container creation.
- PMM-7684, PMM-7498: Ability to discover running and supported Microsoft Azure Databases instances in a provided account.
- PMM-7678, PMM-7679, PMM-7676, PMM-7499, PMM-7691: Prepare, modify and use `azure_exporter` to collect Node related metrics.
- PMM-7681: Use Microsoft Azure metrics on Node/OS-related dashboards to show the metrics on panels.
- PMM-7339: Security Threat Tool: Ability to execute security checks individually and on-demand.
- PMM-7451, PMM-7337: Security Threat Tool: Ability to change intervals for security checks on the *PMM Settings* page.
- PMM-7772, PMM-7338: Security Threat Tool: Ability to change default execution interval per check.
- PMM-7336: Security Threat Tool: Execute checks based on execution interval they belong to.
- PMM-7335: Security Threat Tool: Ship security check files with predefined execution interval.
- PMM-7748: Add an additional experimental menu for Dashboards on the left side panel.
- PMM-7688: Unify UX and layout of all PMM specific pages like Settings, Add Instance etc.
- PMM-7687: Modify links in menus to ensure both menus are working as expected after dashboard URL change.
- PMM-7705: Simplify display of features in technical preview to easily identify them and their current state.
- PMM-7522, PMM-7511: Integrated Alerting: Improve Notification Channels UX by Pagination for the Notification list.
- PMM-7521, PMM-7510: Integrated Alerting: Improve Alert Rule Templates UX by Pagination on Rule Templates list.
- PMM-7652, PMM-7674, PMM-7503, PMM-7486: DBaaS: While creating the DB cluster see all and available resources in the K8s cluster, such as Disk, CPU & Memory.

Percona LLC and/or its affiliates ©2024

- PMM-7508, PMM-7488: DBaaS: See predicted resource usage for selected DB Cluster configuration.
- PMM-7364: DBaaS: Show warning before starting creating the cluster if there are not enough resources in the K8s cluster to create DB Cluster with requested configuration.
- PMM-7580, PMM-7359: DBaaS: Users can select the database version to use during DB Cluster creation.

9.34.4 Improvements

- PMM-7506: Security Threat Tool: Reduce False Positives due to Roles automatically created in PXC with no password but cannot be used to login.
- PMM-7569: Make PMM compatible with pg_stat_monitor 0.8 release.
- PMM-7571: Modified Percona Platform Account registration flow from PMM server UI.
- PMM-7513: Integrated Alerting: Ability to see default values and Threshold values during the Alert Rule creation.
- PMM-7461: Integrated Alerting: Improve UX of tables presentation and loading on UI.
- PMM-7375: Integrated Alerting: Inform users about the template that they are editing and warn them about the limitations.
- PMM-7260: Integrated Alerting: Make it clearer what rule is being edited.

9.34.5 Bugs Fixed

- PMM-7131, PMM-7555: QAN for PostgreSQL attempts to connect to a database with the same name as the username. (Thanks to Daniel Guzman Burgos for reporting this issue)
- PMM-7481: Query Analytics is not showing “Query with Errors” in the Profile section.
- PMM-7464: NGINX misconfiguration leads to log storm in push mode
- PMM-7434: Integrated Alerting: Unknown parameters [threshold] error during Add/Update Alert Rule.
- PMM-7231: Integrated Alerting: Disabling channels does nothing.
- PMM-7379: Integrated Alerting: Can not edit Alert Rule Name through API.
- PMM-7232: Integrated Alerting: Disabling IA does not disable rules evaluation and notifications sending.
- PMM-7119: Integrated Alerting: Update error notification for adding/update Alert rule Template – There was inconsistent behavior if you tried to add a new Rule Template with an already-used name.
- PMM-7543: Integrated Alerting: selected section disappears from a breadcrumb after clicking the tab for a second time.
- PMM-7766: DBaaS: PMM Upgrade breaks DBaaS get credentials method.
- PMM-7351: DBaaS: Safari does not accept float numbers as a custom option in the “Create Cluster” dialogue.
- PMM-7701: DBaaS: PSMDB clusters stuck in initializing due to special characters in secrets.

9.34.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-08-03

Percona LLC and/or its affiliates ©2024

9.35 Percona Monitoring and Management 2.15.1

Date: March 18, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.35.1 Release Highlights

This patch release fixes performance issues discovered in systems, together with other small fixes.

9.35.2 Bugs Fixed

- PMM-7635: Fix high CPU consumption by Grafana server after upgrade by docker replacement to 2.15.0 with large numbers of services in ‘push’ mode.
- PMM-7713: Fix high CPU and Memory consumption by Victoria Metrics after upgrade by docker replacement to 2.15.0 with large numbers of services in ‘pull’ mode.
- PMM-7470: MongoDB exporter `IndexStatsCollections` is assigned values from wrong flag (intended for 2.15.0, omitted due to missing merge cutoff) (Thanks to Tim for reporting this issue).
- PMM-1531: Metrics not being collected due to rename of MySQL 8 information schema tables.

9.35.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.36 Percona Monitoring and Management 2.15.0

Date: March 01, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.36.1 Release Highlights

- PMM 1 vs. 2 Parity

  Disable collectors during adding node/service to monitoring

  With this feature users can disable any collector used by PMM to get metrics. When metrics cannot be collected or are no longer needed, disabling the collector(s) prevents PMM from flooding logs and saves infrastructure resources.

Percona LLC and/or its affiliates ©2024

Our vision for PMM collectors is to provide “stop from collecting” functionality to prevent possible harm to the user environment. This “disable” feature is an initial step towards the ideal functionality. The full and flexible management for “What metrics to collect and in what resolution” is slated for future releases.

External services monitoring

Since PMM 1.4.0, users had the ability to monitor external services Percona didn’t currently support (e.g., Redis). This blog article from 2018 nicely described external services monitoring at that time. (At that time Percona was not natively supporting a PostgreSQL monitoring service and so this was listed as an external service. Today, PostgreSQL is natively supported by PMM.)

Until now, PMM 2.x didn’t support external services monitoring. With this release, any non-natively supported by PMM service will now become supported with external services monitoring. You can see the list of possible exporters to be used in https://prometheus.io/docs/instrumenting/exporters/. Natively-supported services will continue to deliver an expanded set of metrics and insights.

Provide summary information for systems (`pt-*-summary actions`)

With the addition of `pt-*-summary` in PMM 2, users can now view summary information about services and nodes on PMM’s dashboard. This summary information is in the industry common format of `pt-*-summary` tools output to simplify portability of this data. This format will also be preserved in the snapshot of the dashboard shared with Percona Support to simplify investigations of issues.

Note: `pt-*-summary` includes formats for:

- `pt-mysql-summary`
- `pt-mongodb-summary`
- `pt-pg-summary`
- `pt-summary`

- HAProxy support by PMM

  Users are able to add HAProxy Services for monitoring in PMM2. The support level of them in PMM will be the same we have for ProxySQL, so they will be presented in Inventory and on Dashboard. This will allow users who use HAProxy in their HA configuration to also have this component monitored by PMM. In future releases PMM will start use HAProxy by default for the DBaaS feature and will also use this functionality to monitor HAProxy.

- DBaaS Preview improvements (Technical Preview)

  From now you will be able to see the progress of internal steps the system makes when executing some operations with DBaaS. The Progress Bar will not be time-related and will present only steps. The Progress Bar component will also reflect the K8s/Operator-related errors to the user, so in the case of errors, you will have the error text on the UI, and no need to use K8s tools to see the error. With the same UI, you will be able to see the latest logs from K8s so they will have even more information about why the error happened.

  Known Limitations: The progress bar will not provide valuable information for the Delete operation (will be in a later version when we’ll change the API with Operators Team), Operation of DB Cluster Modification will have “strange” behavior and will start changes from non-zero values of steps. (This will be modified after API changes.)

9.36.2 New Features

- PMM-4172, PMM-4306, PMM-5784, PMM-7177: Services and Nodes Summary presentation. Present information about DB’s and Node status using `pt-mysql-summary`, `pt-mongodb-summary`, `pt-pg-summary` outputs (in API and on Dashboards).
- PMM-7123: Ability to add External Services via the UI in PMM server.
- PMM-6711: Add `external-group` flag for `pmm-admin inventory` commands for simpler work with External services.
- PMM-7405: Check connection response format when adding External Service to monitoring.
- PMM-6797: HAProxy monitoring: Ability to add HAProxy services with `pmm-admin [inventory] add [service] haproxy` command.

Percona LLC and/or its affiliates ©2024

- PMM-7487: HAProxy monitoring: Check connection to HAProxy services when adding them for monitoring.
- PMM-7496: HAProxy monitoring: New HAProxy PXC dashboards.
- PMM-6943: HAProxy monitoring: Show HAProxy type services in PMM Inventory.
- PMM-6924: Integrated Alerting: Show 'breadcrumbs' navigation aid on non-dashboard pages as well as Grafana dashboard pages.
- PMM-7294: Integrated Alerting: Pagination for viewing large numbers of Alert Rules.
- PMM-7417: Security Threat Tool: Show list of all available security checks.
- PMM-7418: Security Threat Tool: Ability to disable specific security checks.
- PMM-7419: DBaaS: Ability to see DB Cluster creation/modification logs.
- PMM-7266: DBaaS: Cluster creation progress bar – You can now see the progress of DBaaS DB cluster creation. (The progress bar is based on the number of back-end technical steps, not the time required to perform the tasks.)

9.36.3 Improvements

- PMM-4679: Docker: `:latest` tag for `pmm-server` and `pmm-client` images has been moved from v1 latest release to v2 latest release. **Note**: use of the latest tag is not recommended in production environments, instead use `:2` tag.
- PMM-7472: Remove Prometheus data source – If you were using custom dashboards with a specified data source (not using empty to use default one) you may need to edit your dashboards to use the proper data source. PMM is no longer using Prometheus but uses compatible storage for metrics from VictoriaMetrics. We renamed the data source to be more technology-agnostic.
- PMM-6695: Software update: Grafana 7.1.3 to 7.3.7 (See What's new in Grafana 7.2 and What's new in Grafana 7.3.)
- PMM-7471: Software update: VictoriaMetrics 1.52.0 to 1.53.1 (See VictoriaMetrics 1.53.0 and VictoriaMetrics 1.53.1.)
- PMM-6693: API keys usage – PMM users can now use API keys (generated in Grafana UI) for interaction with PMM server instead of username/password pairs. The API key should have the same level of access (Admin or Viewer) as is required for username/password pairs.
- PMM-7240: DBaaS: Change from Dashboard to Grafana Page – We changed the DBaaS page from a Grafana Dashboard to a Grafana Page to be better aligned with the DBaaS enable/disable status and avoid confusion when DBaaS is disabled.
- PMM-7328: Security Threat Tool: Download and run checks when activated, immediately, repeating every 24 hours thereafter (Previously, downloading and running new checks happened every 24 hours but the cycle didn't begin when STT was activated.)
- PMM-7329: Security Threat Tool: Hide check results tab if STT is disabled.
- PMM-7331: Security Threat Tool: Failed checks have 'Read more' links with helpful content.
- PMM-7422: Security Threat Tool: View all active and silenced alerts.
- PMM-7257, PMM-7433: Integrated Alerting: Easier-to-read rule details in Alert Rules list (API and UI presentation).
- PMM-7259: Integrated Alerting: Better UI error reporting for disabled Integrated Alerting. (Hint to users how to enable it.)
- PMM-5533: Better indentation of columns in `pmm-admin list` output.
- PMM-5888: Improve `pmm-admin --help` descriptions for external services.

9.36.4 Bugs Fixed

- PMM-5837: `pmm-agent` reports "Malformed DSN" error when adding PostgreSQL instance with a PMM user password containing `=` (equals sign) (Thanks to Alexandre Barth for reporting this issue).

Percona LLC and/or its affiliates ©2024

- PMM-5969: Removing Services or Nodes with `pmm-admin ... --force` mode does not stop running agents, VictoriaMetrics continues collecting data from exporters.
- PMM-6685: In low screen resolutions Services submenu wraps, becomes obscured, and can't be accessed.
- PMM-6681: Not all PMM admin users can download diagnostic logs, only those with Grafana admin rights.
- PMM-7227: Table stats metrics not being collected in instances with millions of tables.
- PMM-7426: `vmagent` continually restarts, blocking comms between `pmm-agent` & `pmm-managed` – Users running multiple services on the same PMM agent in 'push' mode could face this issue when restarting the agent after bulk-adding services.
- PMM-6636: Dashboards: MySQL Replication Summary: 'Binlog Size', 'Binlog Data Written Hourly', 'Node' not being charted when the instance is RDS.
- PMM-7325: Dashboards: MySQL User Details: user labels unreadable with high number (>20) of users (Thanks to Andrei Fedorov for reporting this issue).
- PMM-7416: Dashboards: PostgreSQL Instance Summary: Some panels (e.g. Tuple) not using selected database.
- PMM-7235: Integrated Alerting: Filtered out alerts are shown in the UI as firing.
- PMM-7324: Integrated Alerting: Add Pager Duty Notification Channel: after user pastes copied key Add button is not enabled.
- PMM-7346: Integrated Alerting: It is possible to create Alert Rule with negative duration time.
- PMM-7366: Integrated Alerting: Entities (e.g. templates, channels, rules) are in inconsistent states.
- PMM-7467: Integrated Alerting: `<` (less-than symbol) wrongly interpreted by Alert templates (as `<`).
- PMM-7591: Integrated Alerting: User can not receive notifications on email after password update.
- PMM-7343: Security Threat Tool: Check results show previously failed checks after STT re-enabled.
- PMM-7250: DBaaS: Confusing error "Cannot get PSMDB/PXC cluster" appears after removing DB cluster.
- PMM-7193: DBaaS: Number of Nodes can be set as float.
- PMM-7349: DBaaS: Host and Password occasionally disappearing from Connection column.

9.36.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.37 Percona Monitoring and Management 2.14.0

Date: January 28, 2021
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.37.1 Release Highlights

- Switch to push metrics by default

Percona LLC and/or its affiliates ©2024

In PMM 2.12.0, Percona replaced its metrics collection engine (formerly Prometheus) with VictoriaMetrics. Historically, PMM used a *pull* method with Prometheus while VictoriaMetrics can operate in either a *pull* or *push* method. When PMM 2.12.0 was released, Percona kept the default method as *pull*. Now with PMM 2.14.0, Percona is shifting the default to *push* for all newly-added instances. This blog post describes the two methods and why *push* benefits users. Also, here is a post by Peter Zaitzev of FAQs relating to the move to VictoriaMetrics and the push model. Documentation on the push method is here.

Note: Installing the 2.14.0 or newer PMM server will change the default behavior on 2.12.0 and 2.13.0 clients from “pull” method to “push” for any newly added services. Existing services will remain in whatever mode they were prior to upgrade.

- DBaaS Preview phase 1.0 (Technical Preview)

  In 2.13.0 we introduced Percona’s Database as a Service (DBaaS) which enables non-DBAs (software architects, developers, site reliability engineers, etc.) to perform typical DBA tasks to manage an organization’s database environment via user interfaces and automation orchestration. This release contains several enhancements and fixes, many directly from user feedback.

  Note: This capability is feature-flagged and turned off by default. Users require a variable to be passed to PMM to expose this functionality.

- External services presentation on node summary dashboard

  Improvements to the user experience for adding and viewing external services (any data that can be monitored by a Prometheus exporter such as: non-Percona supported databases like Redis, ElasticSearch, Cassandra, etc. or an organization’s external application) on the Node Summary dashboard of PMM.

9.37.2 New Features

- PMM-5765: Ability to monitor External Services for situations where PMM Client can’t be installed – Uses a new command `pmm-admin add external-serverless`. (See pmm-admin.) (This is a **Technical Preview** feature)
- PMM-7015: DBaaS Preview: Create DB cluster with randomly-generated password
- PMM-7007: Integrated Alerting: Ability to copy (duplicate) alert rules
- PMM-7006: Integrated Alerting: Ability to delete alert rules
- PMM-6941: Integrated Alerting: Ability to delete alert rule templates

9.37.3 Improvements

- PMM-6985: DBaaS: Ability to force unregister Kubernetes cluster
- PMM-7145: ‘Push’ metrics mode is default when adding services and nodes (All agents collecting data from Services and Nodes will now use PUSH model if not specified explicitly. You will still be able to use `--metrics-mode` flag to use Pull metrics if needed. All previously set up agents will keep their existing mode. To change these you need to remove and re-add them.)
- PMM-7282: Integrated Alerting: Ability to create rule without channels and filters
- PMM-7226: Integrated Alerting: Validate parameters during rule creation/update
- PMM-7082: Integrated Alerting: Severity levels are color-coded
- PMM-7065: Integrated Alerting: Show rule details for items in Alert Rules list
- PMM-7048: DBaaS: Simplify Cluster creation by moving Create Cluster button to earlier steps
- PMM-6993: Protect against possible problems with EXPLAIN of stored functions in MySQL – We are fixing possible problems caused by an attempt to analyze queries covered in https://bugs.mysql.com/bug.php?id=67632.

Percona LLC and/or its affiliates ©2024

9.37.4 Bugs Fixed

- PMM-7312: Error when accessing Metrics data on Dashboards for large installations
- PMM-7310: VictoriaMetrics consuming 100's Gb's of disk in `/tmp/searchResults` in PMM 2.13.0
- PMM-5137: Swagger page redirect isn't working
- PMM-7144: DBaaS: Creating DB cluster with same name (Thanks to Beata Handzelova for reporting this issue)
- PMM-7323: DBaaS: 'Remove DB Cluster from Kubernetes Cluster' removes wrong one
- PMM-7251: Integrated Alerting: Error `Rule with ID "mysql_version" not found` if both Security Threat Tool and Integrated Alerting enabled
- PMM-7247: DBaaS: Disk size is always 0 for Percona XtraDB cluster
- PMM-7178: `pg_stat_monitor` integration is broken with version 0.6.0 of the plugin
- PMM-7169: Old data (from Prometheus) not deleted when Retention period expires
- PMM-7105: Query Analytics: no 'Example' or 'Explain' data for MariaDB
- PMM-7239: Integrated Alerting: Validate Slack channel names in Notification Channels
- PMM-7213: MySQL InnoDB Details dashboard: remove color-coding on 'Data Buffer Pool Fit' element
- PMM-7167: Some panels not visible when using long time intervals (e.g. 30 days)
- PMM-7133: Incorrect descriptions for data links in dashboards
- PMM-7103: VictoriaMetrics build logs not deleted from PMM Server Docker image
- PMM-6904: `pmm-admin annotate` command crashes for non-generic node types
- PMM-6902: No query Examples on PostgreSQL 12 with pg_stat_monitor
- PMM-6838: ProxySQL Instance Summary dashboard: Incorrect "Hostgroup Size" formula
- PMM-6490: `rds_exporter` crashes when more than 100 AWS RDS instances added (Thanks to https://github.com/vlinevych for fixing this)
- PMM-6096: `pmm-agent` connection checker does not check authentication for MongoDB
- PMM-7303: Disk Details, Nodes Compare dashboards: 'Disk Utilization' description is confusing

9.37.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.38 Percona Monitoring and Management 2.13.0

Date: December 29, 2020

Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

Percona LLC and/or its affiliates ©2024

9.38.1 Release Highlights

1. Ability to monitor SSL-enabled MongoDB

   Allows PMM administrators to set up configured SSL certificate “keys” to authenticate the connection to PMM, specifically for setting up MongoDB. This is a critical security requirement especially in large enterprise infrastructure environments.

2. Technical Previews

   > **Caution**
   >
   > We do not recommend the use of technical preview features in enterprise or production environments until the functionality is released as general availability (GA). While in Technical Preview status, these features are not supported by Percona Support SLA, except by Product/Engineering on a best-efforts basis.

   a. Integrated Alerting MVP

      A new feature in PMM to set up parameters and revive alerts about the Services and Nodes monitored by PMM.

      Read more on our blog and in our documentation.

   b. Node Summary/Nodes Overview dashboards: Show External services on dashboards

      Improves the user experience for adding and viewing external services on the Node Summary dashboard of PMM. External services means any data that can be monitored by a Prometheus exporter, for example, non-Percona supported databases like Redis, ElasticSearch, Cassandra, etc. or an organization’s external application.

   c. DBaaS Preview phase 1.0

      We are also releasing the first preview of DBaaS functionality; when combined with a compatible Kubernetes environment and Percona Operators, you can create Percona XtraDB or MongoDB clusters with just a few clicks. (Read more about configuration and usage.)

9.38.2 Improvements

- PMM-5364: Ability to monitor SSL-enabled MongoDB by passing certificate parameters in `pmm-admin add` command (Thanks to Hubertus Krogmann for reporting this issue)
- PMM-7086: Re-mapped `/prometheus/<end-point>` to `/victoriametrics/<end-point>` but created aliases for users that still rely on the `/prometheus/<end-point>` in bookmarks and scripts (Thanks to Daniel Guzman Burgos for reporting this issue)
- PMM-6713: Node Summary/Nodes Overview dashboards: External exporters can now be added to dashboard and shown as part of grouping of a broader service
- PMM-7173: VictoriaMetrics updated to 1.50.2: Includes HTML pages vs JSON output and new functions available for alerting rules (see all tags)

9.38.3 Bugs Fixed

- PMM-7054: ProxySQL Instance Summary dashboard: no Node Metrics
- PMM-7092: PMM Server Docker update from 2.11.1 to 2.12.0 leaves container in unhealthy state (Thanks to Hubertus Krogmann for reporting this issue)
- PMM-7208: Confusing “Access denied” message for ‘Viewer’ users on many dashboards
- PMM-6987: No IP address shown in log file of OVF appliance running in headless mode

Percona LLC and/or its affiliates ©2024

- PMM-7146: MongoDB Instance Summary dashboard: `ReplSet` element showing metric name instead of replication set
- PMM-6992: Administrators can't see user's actual IP address in Grafana profile-Preferences-Sessions
- PMM-6865: Rendered dashboard images partly obscured by error message

9.38.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-12-19

9.39 Percona Monitoring and Management 2.12.0

Date: December 1, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.39.1 Release Highlights

- VictoriaMetrics replaces Prometheus and is now the default data source. VictoriaMetrics supports both PUSH (client to server) and PULL metrics collection modes. (Read more.)
- PMM Client can be run as a Docker image.
- The 'Add Instance' page and forms have been redesigned and look much better.

9.39.2 New Features

- PMM-5799: PMM Client now available as docker image in addition to RPM, DEB and `.tgz`
- PMM-6968: Integrated Alerting: Basic notification channels actions API Create, Read, Update, Delete
- PMM-6842: VictoriaMetrics: Grafana dashboards to monitor `VictoriaMetricsDB` as replacement for dashboards that used to monitor Prometheus DB
- PMM-6395: Replace Prometheus with VictoriaMetrics in PMM for better performance and additional functionality

9.39.3 Improvements

- PMM-6744: Prevent timeout of low resolution metrics in MySQL instances with many tables (~1000's)
- PMM-6504: MySQL Replication Summary: MySQL Replication Delay graph not factoring in value of intentionally set `SQL_Delay` thus inflating time displayed
- PMM-6820: `pmm-admin status --wait` option added to allow for configurable delay in checking status of `pmm-agent`
- PMM-6710: `pmm-admin`: Allow user-specified custom 'group' name when adding external services
- PMM-6825: Allow user to specify 'listen address' to `pmm-agent` otherwise default to 127.0.0.1
- PMM-6793: Improve user experience of 'add remote instance' workflow

Percona LLC and/or its affiliates ©2024

- PMM-6759: Enable Kubernetes startup probes to get status of `pmm-agent` using 'GET HTTP' verb
- PMM-6736: MongoDB Instance Summary dashboard: Ensure colors for ReplSet status matches those in MongoDB ReplSet Summary dashboard for better consistency
- PMM-6730: Node Overview/Summary Cleanup: Remove duplicate service type 'DB Service Connections'
- PMM-6542: PMM Add Instance: Redesign page for more intuitive experience when adding various instance types to monitoring
- PMM-6518: Update default data source name from 'Prometheus' to 'Metrics' to ensure graphs are populated correctly after upgrade to VictoriaMetrics
- PMM-6428: Query Analytics dashboard - Ensure user-selected filter selections are always visible even if they don't appear in top 5 results
- PMM-5020: PMM Add Remote Instance: User can specify 'Table Statistics Limit' for MySQL and AWS RDS MySQL to disable table stat metrics which can have an adverse impact on performance with too many tables

9.39.4 Bugs Fixed

- PMM-6811: MongoDB Cluster Summary: when secondary optime is newer than primary optime, lag incorrectly shows 136 years
- PMM-6650: Custom queries for MySQL 8 fail on 5.x (on update to `pmm-agent` 2.10) (Thanks to user debug for reporting this issue)
- PMM-6751: PXC/Galera dashboards: Empty service name with MySQL version < 5.6.40
- PMM-5823: PMM Server: Timeout when simultaneously generating and accessing logs via download or API
- PMM-4547: MongoDB dashboard replication lag count incorrect (Thanks to user `vvol` for reporting this issue)
- PMM-7057: MySQL Instances Overview: Many monitored instances (~250+) gives 'too long query' error
- PMM-6883: Query Analytics: 'Reset All' and 'Show Selected' filters behaving incorrectly
- PMM-6686: Query Analytics: Filters panel blank on Microsoft Edge 44.18362.449.0
- PMM-6007: PMM Server virtual appliance's IP address not shown in OVF console
- PMM-6754: Query Analytics: Bad alignment of percentage values in Filters panel
- PMM-6752: Query Analytics: Time interval not preserved when using filter panel dashboard shortcuts
- PMM-6664: Query Analytics: No horizontal scroll bar on Explain tab
- PMM-6632: Node Summary - Virtual Memory Utilization chart: incorrect formulas
- PMM-6537: MySQL InnoDB Details - Logging - Group Commit Batch Size: giving incorrect description
- PMM-6055: PMM Inventory - Services: 'Service Type' column empty when it should be 'External' for external services

9.39.5 Known Issues

- PMM-7092: Update docker `pmm-server` 2.11.1 to 2.12.0 results in an unhealthy container.

  Workaround: A folder is not created on container upgrade and will need to be created manually for one of the components. Before starting the new pmm-server 2.12.0, execute:

```
docker exec -ti pmm-server mkdir -p /srv/victoriametrics/data
docker exec -ti pmm-server chown -R pmm:pmm /srv/victoriametrics/
docker restart pmm-server
```


Percona LLC and/or its affiliates ©2024

9.39.6 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.40 Percona Monitoring and Management 2.11.1

Date: October 19, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.40.1 Bugs Fixed

- PMM-6782: High CPU usage after update to 2.11.0

9.40.2 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.41 Percona Monitoring and Management 2.11.0

Date: October 14, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.41.1 New Features

- PMM-6567: Technical preview of new PostgreSQL extension `pg_stat_monitor`
- PMM-6515: Link added directly to Node/Service page from Query Analytics filters, opens in new window

9.41.2 Improvements

- PMM-6727: Grafana plugin updates: `grafana-polystat-panel=1.2.2`, `grafana-piechart-panel=1.6.1`
- PMM-6625: Default sort to “Average - descending” on all dashboards

Percona LLC and/or its affiliates ©2024

- PMM-6609: MySQL Instances Compare & Summary dashboards: Changed metric in ‘MySQL Internal Memory Overview’
- PMM-6598: Dashboard image sharing (Share Panel): Improved wording with link to configuration instructions
- PMM-6557: Update Prometheus to 2.21.0
- PMM-6554: MySQL InnoDB Details dashboard: Add “sync flushing” to “InnoDB Flushing by Type”

9.41.3 Bugs Fixed

- PMM-4547: MongoDB dashboard replication lag count incorrect (Thanks to user `vvol` for reporting this issue)
- PMM-6639: Integrated update does not detect all container types
- PMM-6765: Tables information tab reports ‘table not found’ with new PostgreSQL extension `pg_stat_monitor`
- PMM-6764: Query Analytics: cannot filter items that are hidden - must use “Show all”
- PMM-6742: Upgrade via PMM UI stalls (on `yum update pmm-update`)
- PMM-6689: No PostgreSQL queries or metrics in Query Analytics with PostgreSQL 13 (`postgresql_pgstatements_agent` in Waiting status)
- PMM-6738: PostgreSQL examples shown despite `--disable-queryexamples` option
- PMM-6535: Unable to open ‘Explore’ in new window from Grafana menu
- PMM-6532: Click-through URLs lose time ranges when redirecting to other dashboards
- PMM-6531: Counter-intuitive coloring of element “Update Stats when Metadata Queried”
- PMM-6645: Clean up unnecessary errors in logs (`vertamedia-clickhouse-datasource` plugin)
- PMM-6547: Hexagonal graph tooltip text overflows bounding box

9.41.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.42 Percona Monitoring and Management 2.10.1

Date: September 22, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.42.1 Bugs Fixed

- PMM-6643: New MongoDB exporter has higher CPU usage compared with old

Percona LLC and/or its affiliates ©2024

9.42.2 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.43 Percona Monitoring and Management 2.10.0

Date: September 15, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.43.1 New Features

- PMM-2045: New dashboard: MySQL Group Replication Summary
- PMM-5738: Enhanced exporter: replaced original `mongodb-exporter` with a completely rewritten one with improved functionality
- PMM-5126: Query Analytics Dashboard: Search by query substring or dimension (Thanks to user `debug` for reporting this issue)
- PMM-6360: Grafana Upgrade to 7.1.3
- PMM-6355: Upgrade Prometheus to 2.19.3
- PMM-6597: Documentation: Updated Image rendering instructions for PMM
- PMM-6568: Reusable user interface component: Pop-up dialog. Allows for more consistent interfaces across PMM
- PMM-6375, PMM-6373, PMM-6372: Sign in, Sign up and Sign out UI for Percona Account inside PMM Server
- PMM-6328: Query Analytics Dashboard: Mouse-over crosshair shows value on sparklines
- PMM-3831: Node Summary Dashboard: Add `pt-summary` output to dashboard to provide details on system status and configuration

9.43.2 Improvements

- PMM-6647: MongoDB dashboards: RocksDB Details removed, MMAPv1 & Cluster Summary changed
- PMM-6536: Query Analytics Dashboard: Improved filter/time search message when no results
- PMM-6467: PMM Settings: User-friendly error message
- PMM-5947: Bind services to internal address for containers

9.43.3 Bugs Fixed

- PMM-6336: Suppress sensitive data: honor `pmm-admin` flag `--disable-queryexamples` when used in conjunction with `--query-source=perfschema`
- PMM-6244: MySQL InnoDB Details Dashboard: Inverted color scheme on “BP Write Buffering” panel

Percona LLC and/or its affiliates ©2024

- PMM-6294: Query Analytics Dashboard doesn't resize well for some screen resolutions (Thanks to user `debug` for reporting this issue)
- PMM-5701: Home Dashboard: Incorrect metric for `DB uptime` (Thanks to user `hubi_oediv` for reporting this issue)
- PMM-6427: Query Analytics dashboard: Examples broken when switching from MongoDB to MySQL query
- PMM-5684: Use actual data from `INFORMATION_SCHEMA` vs relying on cached data (which can be 24 hrs old by default)
- PMM-6500: PMM Database Checks: Unwanted high-contrast styling
- PMM-6440: MongoDB ReplSet Summary Dashboard: Primary shows more lag than replicas
- PMM-6436: Query Analytics Dashboard: Styles updated to conform with upgrade to Grafana 7.x
- PMM-6415: Node Summary Dashboard: Redirection to database's Instance Summary dashboard omits Service Name
- PMM-6324: Query Analytics Dashboard: Showing stale data while fetching updated data for query details section
- PMM-6316: Query Analytics Dashboard: Inconsistent scrollbar styles
- PMM-6276: PMM Inventory: Long lists unclear; poor contrast & column headings scroll out of view
- PMM-6529: Query Analytics filter input margin disappears after scrolling

9.43.4 Known Issues

- PMM-6643: High CPU usage for new MongoDB exporter (fixed in Percona Monitoring and Management 2.10.1)

9.43.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.44 Percona Monitoring and Management 2.9.1

Date: August 4, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.44.1 Improvements

- PMM-6230: Custom dashboards set as Home remain so after update
- PMM-6300: Query Analytics Dashboard: Column sorting arrows made easier to use (Thanks to user debug for reporting this issue)
- PMM-6208: Security Threat Tool: Temporarily silence viewed but un-actioned alerts
- PMM-6315: Query Analytics Dashboard: Improved metrics names and descriptions
- PMM-6274: MySQL User Details Dashboard: View selected user's queries in Query Analytics Dashboard
- PMM-6266: Query Analytics Dashboard: Pagination device menu lists 25, 50 or 100 items per page

Percona LLC and/or its affiliates ©2024

- PMM-6262: PostgreSQL Instance Summary Dashboard: Descriptions for all ‘Temp Files’ views
- PMM-6253: Query Analytics Dashboard: Improved SQL formatting in Examples panel
- PMM-6211: Query Analytics Dashboard: Loading activity spinner added to Example, Explain and Tables tabs
- PMM-6162: Consistent sort order in dashboard drop-down filter lists
- PMM-5132: Better message when filter search returns nothing

9.44.2 Bugs Fixed

- PMM-5783: Bulk failure of SHOW ALL SLAVES STATUS scraping on PS/MySQL distributions triggers errors
- PMM-6294: Query Analytics Dashboard doesn’t resize well for some screen resolutions (Thanks to user debug for reporting this issue)
- PMM-6420: Wrong version in successful update pop-up window
- PMM-6319: Query Analytics Dashboard: Query scrolls out of view when selected
- PMM-6302: Query Analytics Dashboard: Unnecessary EXPLAIN requests
- PMM-6256: Query Analytics Dashboard: `InvalidNamespace` EXPLAIN error with some MongoDB queries
- PMM-6329: Query Analytics Dashboard: Unclear origin of sparkline tool-tip on mouse-over
- PMM-6259: Query Analytics Dashboard: Slow appearance of query time distribution graph for some queries
- PMM-6189: Disk Details Dashboard: Disk IO Size chart larger by factor of 512
- PMM-6269: Query Analytics Dashboard: Metrics drop-down list obscured when opened
- PMM-6247: Query Analytics Dashboard: Overview table not resizing on window size change
- PMM-6227: Home Dashboard redirection to Node Summary Dashboard not working

9.44.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.45 Percona Monitoring and Management 2.9.0

Date: July 14, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.45.1 Release Highlights

This release brings a major rework of the Query Analytics (QAN) component, completing the migration from Angular to React, and adding new UI functionality and features.

Percona LLC and/or its affiliates ©2024

For details, see:

- PMM-5125: Implement new version of QAN
- PMM-5516: QAN migration to React and new UI implementation

You can read more in the accompanying blog post (here).

9.45.2 New Features

- PMM-6124: New dashboards: MongoDB Replica Set Summary and MongoDB Cluster Summary
- PMM-1027: New dashboard: MySQL User Details (`INFORMATION_SCHEMA.CLIENT_STATISTICS`)
- PMM-5604: User interface for MongoDB EXPLAIN
- PMM-5563: Per-Service and per-Node Annotations (This completes the work on improvements to the Annotation functionality.)

9.45.3 Improvements

- PMM-6114: Sort Agents, Nodes, and Services alphabetically by name in Inventory page (Thanks to user `debug` for reporting this issue)
- PMM-6147: Update Grafana plugins to latest versions

9.45.4 Bugs Fixed

- PMM-5800: QAN explain and tables tabs not working after removing MySQL metrics agent
- PMM-5812: Prometheus relabeling broken (`relabel_configs` un-marshal errors) (Thanks to user `b4bufr1k` for reporting this issue)
- PMM-6184: MongoDB Instances Compare dashboard shows MySQL metric
- PMM-5941: Stacked Incoming/Outgoing Network Traffic graphs in MySQL Instances Overview dashboard prevents comparison
- PMM-6194: Missing UID for Advanced Data Exploration dashboard
- PMM-6191: Incorrect computation for Prometheus Process CPU Usage panel values in Prometheus dashboard
- PMM-6175: Node Overview dashboard shows unit for unit-less value ‘Top I/O Load’

9.45.5 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.46 Percona Monitoring and Management 2.8.0

Date: June 25, 2020
Installation: Installing Percona Monitoring and Management

Percona LLC and/or its affiliates ©2024

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.46.1 Improvements

- PMM-544: Agents, Services and Nodes can now be removed via the 'PMM Inventory' page
- PMM-5706: User-installed Grafana plugins unaffected by PMM upgrade

9.46.2 Bugs Fixed

- PMM-6153: PMM 2.7.0 inoperable when no Internet connectivity
- PMM-5365: Client fails to send non-UTF-8 query analytics content to server (Thanks to user `romulus` for reporting this issue)
- PMM-5920: Incorrect metric used in formula for "Top Users by Rows Fetched/Read" graph
- PMM-6084: Annotations not showing consistently on dashboards
- PMM-6011: No data in MongoDB Cluster summary, RocksDB & MMAPv1 details
- PMM-5987: Incorrect total value for virtual memory utilization

9.46.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.47 Percona Monitoring and Management 2.7.0

Date: June 9, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

In this release, we have updated Grafana to version 6.7.4 to fix CVE-2020-13379. We recommend updating to the latest version of PMM as soon as possible.

9.47.1 New Features

- PMM-5257, PMM-5256, & PMM-5243: `pmm-admin` socket option (`--socket`) to specify UNIX socket path for connecting to MongoDB, PostgreSQL, and ProxySQL instances

9.47.2 Improvements

- PMM-2244: `pmm-admin status` command output shows both `pmm-admin` and `pmm-agent` versions
- PMM-5968: Disallow PMM Server node or agent removal via API
- PMM-5946: MySQL Table Details dashboard filter on Service Name prevents display of services without data

Percona LLC and/or its affiliates ©2024

- PMM-5926: Expose PMM agent version in `pmm-admin status` command
- PMM-5891: PMM Home page now includes News panel
- PMM-5906: Independent update of PMM components deactivated

9.47.3 Bugs Fixed

- PMM-6004: MySQL exporter reporting wrong values for cluster status (`wsrep_cluster_status`)
- PMM-4547: MongoDB dashboard replication lag count incorrect
- PMM-5524: Prometheus alerting rule changes needs docker restart to activate
- PMM-5949: Unwanted filters applied when moving from QAN to Add Instance page
- PMM-5870: MySQL Table Details dashboard not showing separate service names for tables
- PMM-5839: PostgreSQL metrics disparity between query time and block read/write time
- PMM-5348: Inventory page has inaccessible tabs that need reload to access
- PMM-5348: Incorrect access control vulnerability fix (CVE-2020-13379) by upgrading Grafana to 6.7.4

9.47.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.48 Percona Monitoring and Management 2.6.1

Date: May 18, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.48.1 Improvements

- PMM-5936: Improved Summary dashboard for Security Threat Tool ‘Failed Checks’
- PMM-5937: Improved Details dashboard for Security Threat Tool ‘Failed Database Checks’

9.48.2 Bugs Fixed

- PMM-5924: Alertmanager not running after PMM Server upgrade via Docker
- PMM-5915: `supervisord` not restarting after restart of PMM Server virtual appliances (OVF/AMI)
- PMM-5945: ‘Updates’ dashboard not showing available updates
- PMM-5870: MySQL Table Details dashboard not showing separate service names for tables

Percona LLC and/or its affiliates ©2024

9.48.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.49 Percona Monitoring and Management 2.6.0

Date: May 11, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.49.1 New Features

- PMM-5728: Technical preview of External Services monitoring feature. A new command provides integration with hundreds of third-party systems (https://prometheus.io/docs/instrumenting/exporters/) via the Prometheus protocol so that you can monitor external services on a node where PMM agent is installed.
- PMM-5822: PMM now includes a Security Threat Tool to help users avoid the most common database security issues.
- PMM-5559: Global annotations can now be set with the `pmm-admin annotate` command.
- PMM-4931: PMM now checks Docker environment variables and warns about invalid ones.

9.49.2 Improvements

- PMM-1962: The PMM Server API (via `/v1/readyz`) now also returns Grafana status information in addition to that for Prometheus.
- PMM-5854: The Service Details dashboards were cleaned up and some unused selectors were removed.
- PMM-5775: It is now clearer which nodes are Primary and which are Secondary on MongoDB Instance dashboards.
- PMM-5549: PMM's Grafana component is now the latest, 6.7.3.
- PMM-5393: There's a new 'Node Summary' row in the services Summary and Details dashboards summarizing the system update, load average, RAM and memory.
- PMM-4778: `mongodb_exporter` is now the latest version, 0.11.0.
- PMM-5734: Temporary files activity and utilization charts (`rate` & `irate`) were added to the PostgreSQL Instance overview.
- PMM-5695: The error message explains better when using the `--socket` option incorrectly.

9.49.3 Bugs Fixed

- PMM-4829: The MongoDB Exporter wasn't able to collect metrics from hidden nodes without either the latest driver or using the `connect-direct` parameter.
- PMM-5056: The average values for Query time in the Details and Profile sections were different.

Percona LLC and/or its affiliates ©2024

- PMM-2717: Updating MongoDB Exporter resolves an error (`Failed to execute find query on 'config.locks': not found.`) when used with `shardedCluster` 3.6.4.
- PMM-4541: MongoDB exporter metrics collection was including system collections from `collStats` and `indexStats`, causing “log bloat”.
- PMM-5913: Only totals were shown in QAN when filtering on `Cluster=MongoDB`.
- PMM-5903: When applying a filter the QAN Overview was being refreshed twice.
- PMM-5821: The Compare button was missing from HA Dashboard main menus.
- PMM-5687: Cumulative charts for Disk Details were not showing any data if metrics were returning `NaN` results.
- PMM-5663: The ‘version’ value was not being refreshed in various MySQL dashboards.
- PMM-5643: Advanced Data Exploration charts were showing ‘N/A’ for Metric Resolution and ‘No data to show’ in the Metric Data Table.
- PMM-4756: Dashboards were not showing services with empty environments.
- PMM-4562: MongoDB and MySQL registered instances with empty cluster labels (`-environment=<label>`) were not visible in the dashboard despite being added instances.
- PMM-4906: The MongoDB exporter for MongoDB 4.0 and above was causing a “log bloat” condition.

Help us improve our software quality by reporting any bugs you encounter using our bug tracking system.

9.49.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2022-11-18

9.50 Percona Monitoring and Management 2.5.0

Date: April 14, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.50.1 New Features

- PMM-5042 and PMM-5272: PMM can now connect to MySQL instances by specifying a UNIX socket. This can be done with a new `--socket` option of the `pmm-admin add mysql` command. (Note: Updates to both PMM Client and PMM Server were done to allow UNIX socket connections.)
- PMM-4145: Amazon RDS instance metrics can now be independently enabled/disabled for Basic and/or Enhanced metrics.

9.50.2 Improvements

- PMM-5581: PMM Server Grafana plugins can now be updated on the command line with the `grafana-cli` command-line utility.

Percona LLC and/or its affiliates ©2024

- PMM-5536: Three Grafana plugins were updated to the latest versions: `vertamedia-clickhouse-datasource` to 1.9.5, `grafana-polystat-panel` to 1.1.0, and `grafana-piechart-panel` to 1.4.0.
- PMM-4252: The resolution of the PMM Server `favicon` image has been improved.

9.50.3 Bugs Fixed

- PMM-5547: PMM dashboards were failing when presenting data from more than 100 monitored instances (error message `proxy error: context canceled`).
- PMM-5624: Empty charts were being shown in some Node Temperature dashboards.
- PMM-5637: The Data retention value in Settings was incorrectly showing the value as minutes instead of days.
- PMM-5613: Sorting data by Query Time was not working properly in Query Analytics.
- PMM-5554: Totals in charts were inconsistently plotted with different colors across charts.
- PMM-4919: The force option (`--force`) in `pmm-admin config` was not always working.
- PMM-5351: The documentation on MongoDB user privileges has been corrected.

Help us improve our software quality by reporting any bugs you encounter using our bug tracking system.

9.50.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.51 Percona Monitoring and Management 2.4.0

Date: March 18, 2020
Installation: Installing Percona Monitoring and Management

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

9.51.1 New Features

- PMM-3387: Prometheus custom configuration is now supported by PMM Server. The feature is targeted at experienced users and is done by adding the base configuration file into the PMM Server container to be parsed and included into the managed Prometheus configuration.
- PMM-5186: Including `--pprof` option in the `pmm-admin summary` command adds `pprof` debug profiles to the diagnostics data archive
- PMM-5102: The new “Node Details” dashboard now displays data from the hardware monitoring sensors in `hwmon`. The new dashboard is based on the `hwmon` collector data from the `node_exporter`. Please note that data may be unavailable for some nodes because of the configuration or virtualization parameters.

Percona LLC and/or its affiliates ©2024

9.51.2 Improvements

- PMM-4915: The Query Analytics dashboard now shows Time Metrics in the Profile Section as “AVG per query” instead of “AVG per second”
- PMM-5470: ClickHouse query optimized for Query Analytics to improve its speed and reduce the load on the back-end
- PMM-5448: The default high and medium metrics resolutions were changed to 1-5-30 and 5-10-60 sec. To reduce the effect of this change on existing installations, systems having the “old” high resolution chosen on the PMM Settings page (5-5-60 sec.) will be automatically re-configured to the medium one during an upgrade. If the resolution was changed to some custom values via API, it will not be affected
- PMM-5531: A health check indicator was implemented for the PMM Server Docker image. It is based on the Docker HEALTHCHECK. This feature can be used as follows:

```
docker inspect -f {{.State.Health.Status}}
until [ "`docker inspect -f {{.State.Health.Status}} pmm-server`" == "healthy" ]; do sleep 1; done
```

- PMM-5489: The “Total” line in all charts is now drawn with the same red color for better consistency
- PMM-5461: Memory graphs on the node-related dashboards were adjusted to have fixed colors that are more distinguishable from each other
- PMM-5329: Prometheus in PMM Server was updated to version 2.16.0. This update has brought several improvements. Among them are significantly reduced memory footprint of the loaded TSDB blocks, lower memory footprint for the compaction process (caused by the more balanced choice of what to buffer during compaction), and improved query performance for the queries that only touch the most recent 2 hours of data.
- PMM-5210: Data Retention is now specified in days instead of seconds on the PMM Settings page. Please note this is a UI-only change, so the actual data retention precision is not changed
- PMM-5182: The `logs.zip` archive available on the PMM Settings page now includes additional self-monitoring information in a separate `client` subfolder. This subfolder contains information collected on the PMM Server and is equivalent to the one collected on a node by the `pmm-admin summary` command.
- PMM-5112: The Inventory API List requests now can be filtered by the Node/Service/Agent type

9.51.3 Bugs Fixed

- PMM-5178: Query Detail Section of the Query Analytics dashboard didn’t show tables definitions and indexes for the internal PostgreSQL database
- PMM-5465: MySQL Instance related dashboards had row names not always matching the actual contents. To fix this, elements were re-ordered and additional rows were added for better matching of the row name and the corresponding elements
- PMM-5455: Dashboards from the Insight menu were fixed to work correctly when the low resolution is set on the PMM Settings page
- PMM-5446: A number of the Compare Dashboards were fixed to work correctly when the low resolution is set on the PMM Settings page
- PMM-5430: MySQL Exporter section on the Prometheus Exporter Status dashboard now collapsed by default to be consistent with other database-related sections
- PMM-5445, PMM-5439, PMM-5427, PMM-5426, PMM-5419: Labels change (which occurs e.g. when the metrics resolution is changed on the PMM Settings page) was breaking dashboards
- PMM-5347: Selecting queries on the Query Analytics dashboard was generating errors in the browser console
- PMM-5305: Some applied filters on the Query Analytics dashboard were not preserved after changing the time range
- PMM-5267: The Refresh button was not working on the Query Analytics dashboard

Percona LLC and/or its affiliates ©2024

- PMM-5003: pmm-admin list and status use different JSON naming for the same data
- PMM-5526: A typo was fixed in the Replication Dashboard description tooltip

Help us improve our software quality by reporting any bugs you encounter using our bug tracking system.

9.51.4 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.52 Percona Monitoring and Management 2.3.0

Date: February 19, 2020

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

For *PMM* install instructions, see Installing PMM Server and Installing PMM Client.

> **Caution**
>
> PMM 2 is designed to be used as a new installation — please don't try to upgrade your existing PMM 1 environment.

9.52.1 Improvements and new features

- PMM-5064 and PMM-5065: Starting from this release, users will be able to integrate PMM with an external Alertmanager by specifying the Alertmanager URL and the Alert Rules to be executed inside the PMM server (**This feature is for advanced users only at this point**)
- PMM-4954: Query Analytics dashboard now shows units both in the list of queries in a summary table and in the Details section to ease understanding of the presented data
- PMM-5179: Relations between metrics are now specified in the Query Analytics Details section
- PMM-5115: The CPU frequency and temperature graphs were added to the CPU Utilization dashboard
- PMM-5394: A special treatment for the node-related dashboards was implemented for the situations when the data resolution change causes new metrics to be generated for existing nodes and services, to make graphs show continuous lines of the same colors

9.52.2 Fixed bugs

- PMM-4620: The high CPU usage by the pmm-agent process related to MongoDB Query Analytics was fixed
- PMM-5377: `singlestats` showing percentage had sparklines scaled vertically along with the graph swing, which made it difficult to visually notice the difference between neighboring `singlestats`.
- PMM-5204: Changing resolution on the PMM settings page was breaking some `singlestats` on the Home and MySQL Overview dashboards

Percona LLC and/or its affiliates ©2024

- PMM-5251: Vertical scroll bars on the graph elements were not allowed to do a full scroll, making last rows of the legend unavailable for some graphs
- PMM-5410: The “Available Downtime before SST Required” chart on the PXC/Galera Node Summary dashboard was not showing data because it was unable to use metrics available with different scraping intervals

9.52.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.53 Percona Monitoring and Management 2.2.2

Date: February 4, 2020

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

For *PMM* install instructions, see Installing PMM Server and Installing PMM Client.

Caution

PMM 2 is designed to be used as a new installation — please don’t try to upgrade your existing PMM 1 environment.

9.53.1 Improvements and new features

- PMM-5321: The optimization of the Query Analytics parser code for PostgreSQL queries allowed us to reduce the memory resources consumption by 1-5%, and the parsing time of an individual query by 30 to 40%
- PMM-5184: The `pmm-admin summary` command have gained a new `--skip-server` flag which makes it operating in a local-only mode, creating summary file without contacting the PMM Server

9.53.2 Fixed bugs

- PMM-5340: The Scraping Time Drift graph on the Prometheus dashboard was showing wrong values because the actual metrics resolution wasn’t taken into account
- PMM-5060: Query Analytics Dashboard did not show the row with the last query of the first page, if the number of queries to display was 11

9.53.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

Percona LLC and/or its affiliates ©2024

2021-06-25

9.54 Percona Monitoring and Management 2.2.1

Date: January 23, 2020

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance.

For *PMM* install instructions, see Installing PMM Server and Installing PMM Client.

> **Caution**
>
> PMM 2 is designed to be used as a new installation — please don't try to upgrade your existing PMM 1 environment.

PMM Server version 2.2.0 suffered an unauthenticated denial of service vulnerability (CVE-2020-7920). Any other PMM versions do not carry the same code logic, and are thus unaffected by this issue. **Users who have already deployed PMM Server 2.2.0 are advised to upgrade to version 2.2.1 which resolves this issue.**

9.54.1 Improvements and new features

- PMM-5229: The new RDS Exporter section added to the Prometheus Exporter Status dashboard shows singlestats and charts related to the `rds_exporter`
- PMM-5228 and PMM-5238: The Prometheus dashboard and the Exporters Overview dashboard were updated to include the `rds_exporter` metrics in their charts, allowing better understanding of the impacts of monitoring RDS instances
- PMM-4830: The consistency of the applied filters between the Query Analytics and the Overview dashboards was implemented, and now filters selected in QAN will continue to be active after the switch to any of the Overview dashboards available in the Services menu
- PMM-5235: The DB `uptime` `singlestats` in node rows on the Home dashboard were changed to show minimal values instead of average ones to be consistent with the top row
- PMM-5127: The "Search by" bar on the Query Analytics dashboard was renamed to "Filter by" to make its purpose more clear
- PMM-5131: The Filter panel on the Query Analytics dashboard now shows the total number of available Labels within the "See all" link, which appears if the Filter panel section shows only top 5 of its Labels

9.54.2 Fixed bugs

- PMM-5232: The `pmm-managed` component of the PMM Server 2.2.0 is vulnerable to DoS attacks, that could be carried out by anyone who knows the PMM Server IP address (CVE-2020-7920). Versions other than 2.2.0 are not affected.
- PMM-5226: The handlebars package was updated to version 4.5.3 because of the Prototype Pollution vulnerability in it (CVE-2019-19919). Please note PMM versions were not affected by this vulnerability, as handlebars package is used as a build dependency only.
- PMM-5206: Switching to the Settings dashboard was breaking the visual style of some elements on the Home dashboard
- PMM-5139: The breadcrumb panel, which shows all dashboards visited within one session starting from the root, was unable to fully show breadcrumb longer than one line

Percona LLC and/or its affiliates ©2024

- PMM-5212: The explanatory text was added to the Download PMM Server Logs button in the Diagnostic section of the PMM Settings dashboard, and a link to it was added to the Prometheus dashboard which was the previous place to download logs
- PMM-5215: The unneeded `mariadb-libs` package was removed from the PMM Server 2.2.0 OVF image, resulting in both faster updating with the `yum update` command and avoiding dependency conflict messages in the update logs
- PMM-5216: PMM Server Upgrade to 2.2.0 was showing Grafana Update Error page with the Refresh button which had to be clicked to start using the updated version
- PMM-5211: The “Where do I get the security credentials for my Amazon RDS DB instance” link in the Add AWS RDS MySQL or Aurora MySQL instance dialog was not targeted at the appropriate instruction
- PMM-5217: PMM 2.x OVF Image memory size was increased from 1 Gb to 4 Gb with the additional 1 Gb swap space because the previous amount was hardly housing the PMM Server, and it wasn’t enough in some cases like performing an upgrade
- PMM-5271: LVM logical volumes were wrongly resized on AWS deployment, resulting in “no space left on device” errors
- PMM-5295: InnoDB Transaction Rollback Rate values on the MySQL InnoDB Details dashboard were calculated incorrectly
- PMM-5270: PXC/Galera Cluster Summary dashboard was showing empty Cluster drop-down list, making it impossible to choose the cluster name
- PMM-4769: The wrongly named “Timeout value used for retransmitting” `singlestat` on the Network Details dashboard was renamed to “The algorithm used to determine the timeout value” and updated to show the algorithm name instead of a digital code
- PMM-5260: Extensive resource consumption by `pmm-agent` took place in case of Query Analytics for PostgreSQL; it was fixed by a number of optimizations in the code, resulting in about 4 times smaller memory usage
- PMM-5261: CPU usage charts on all dashboards which contain them have undergone colors update to make `softIRQ` and Steal curves better differentiated
- PMM-5244: High memory consumption in the PMM Server with a large number of agents sending data simultaneously was fixed by improving bulk data insertion to the ClickHouse database

9.54.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.55 Percona Monitoring and Management 2.2.0

Date: December 24, 2019

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance. You can run *PMM* in your own environment for maximum security and reliability. It provides thorough time-based analysis for MySQL, MongoDB, and PostgreSQL servers to ensure that your data works as efficiently as possible.

Percona LLC and/or its affiliates ©2024

Main improvements in this release are:

- Alternative installation methods available for PMM 1.x are re-implemented for PMM 2: now PMM Server can be installed as a virtual appliance, or run using AWS Marketplace
- AWS RDS and remote instances monitoring re-added in this release include AWS RDS MySQL / Aurora MySQL instances, and remote PostgreSQL, MySQL, MongoDB, and ProxySQL ones
- The new Settings dashboard allows configuring PMM Server via the graphical interface

For *PMM* install instructions, see Installing PMM Server and Installing PMM Client.

> **Caution**
>
> PMM 2 is designed to be used as a new installation — please don't try to upgrade your existing PMM 1 environment.

9.55.1 Improvements and new features

- PMM-4575: The new PMM Settings dashboard allows users to configure various PMM Server options: setting metrics resolution and data retention, enabling or disabling send usage data statistics back to Percona and checking for updates; this dashboard is now the proper place to upload your public key for the SSH login and to download PMM Server logs for diagnostics
- PMM-4907 and PMM-4767: The user's AMI Instance ID is now used to setup running PMM Server using AWS Marketplace as an additional verification on the user, based on the Amazon Marketplace rules
- PMM-4950 and PMM-3094: Alternative AWS partitions are now supported when adding an AWS RDS MySQL or Aurora MySQL Instance to PMM
- PMM-4976: Home dashboard clean-up: "Systems under monitoring" and "Network IO" `singlestats` were refined to be based on the `host` variable; also avoiding using color as an indicator of state; "All" row elements were relinked to the "Nodes Overview" dashboard with regards to the selected host.
- PMM-4800: The `pmm-admin add mysql` command has been modified to make help text more descriptive: now when you enable `tablestats` you will get more detail on if they're enabled for your environment and where you stand with respect to the auto-disable limit
- PMM-4969: Update Grafana to version 6.5.1
- PMM-5053: A tooltip was added to the Head Block graph on the Prometheus dashboard
- PMM-5068: Drill-down links were added to the Node Summary dashboard graphs
- PMM-5050: Drill-down links were added to the graphs on all Services Compare dashboards
- PMM-5037: Drill-down links were added to all graphs on the Services Overview dashboards
- PMM-4988: Filtering in Query Analytics have undergone improvements to make group selection more intuitive: Labels unavailable under the current selection are shown as gray/disabled, and the percentage values are dynamically recalculated to reflect Labels available within the currently applied filters
- PMM-4966: All passwords are now substituted with asterisk signs in the exporter logs for security reasons when not in debug mode
- PMM-527: `node_exporter` is now providing hardware monitoring information such as CPU temperatures and fan statuses; while this information is being collected by PMM Server, it will not be shown until a dedicated dashboard is added in a future release
- PMM-3198: Instead of showing All graphs for all services by default, MySQL Command/Handler Counters Compare dashboard now shows the predefined set of ten most informative ones, to reduce load on PMM Server at its first open

Percona LLC and/or its affiliates ©2024

9.55.2 Fixed bugs

- PMM-4978: The “Top MySQL Questions” `singlestat` on the MySQL Instances Overview dashboard was changed to show ops instead of percentage
- PMM-4917: The “Systems under monitoring” and “Monitored DB Instances” `singlestats` on the Home dashboard now have a sparkline to make situation more clear with recently shut down nodes/instances
- PMM-4979: Set decimal precision `2` for all the elements, including charts and `singlestats`, on all dashboards
- PMM-4980: Fix “Load Average” `singlestat` on the Node Summary dashboard to show decimal value instead of percent
- PMM-4981: Disable automatic color gradient in filled graphs on all dashboards
- PMM-4941: Some charts were incorrectly showing empty fragments with high time resolution turned on
- PMM-5022: Fix outdated drill-down links on the Prometheus Exporters Overview and Nodes Overview dashboards
- PMM-5023: Make the All instances `uptime singlestat` on the Home dashboard to show `Min` values instead of `Avg`
- PMM-5029: Option to upload dashboard snapshot to Percona was disappearing after upgrade to 2.1.x
- PMM-4946: Rename singlestats on the Home dashboard for better clarity: “Systems under monitoring” to “Nodes under monitoring” and “Monitored DB Instances” to “Monitored DB Services”, and make the last one to count remote DB instances also
- PMM-5015: Fix format of Disk Page Buffers singlestat on the Compare dashboard for PostgreSQL to have two digits precision for the consistency with other singlestats
- PMM-5014: LVM logical volumes were wrongly sized on a new AWS deployment, resulting in “no space left on device” errors.
- PMM-4804: Incorrect parameters validation required both `service-name` and `service-id` parameters of the `pmm-admin remove` command to be presented, while the command itself demanded only one of them to identify the service.
- PMM-3298: Panic errors were present in the `rds_exporter` log after adding an RDS instance from the second AWS account
- PMM-5089: The `serialize-javascript` package was updated to version 2.1.1 because of the possibility of regular expressions cross-site scripting vulnerability in it (CVE-2019-16769). Please note PMM versions were not affected by this vulnerability, as `serialize-javascript` package is used as a build dependency only.
- PMM-5149: Disk Space `singlestat` was unable to show data for RDS instances because of not taking into account sources with unknown file system type

9.55.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.56 Percona Monitoring and Management 2.1.0

Date: November 11, 2019

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance. You can run PMM in your own environment for maximum security

Percona LLC and/or its affiliates ©2024

and reliability. It provides thorough time-based analysis for MySQL, MongoDB, and PostgreSQL servers to ensure that your data works as efficiently as possible.

For install instructions, see Installing Percona Monitoring and Management.

> **Caution**
>
> PMM 2 is designed to be used as a new installation — please don't try to upgrade your existing PMM 1 environment.

9.56.1 Improvements and new features

- PMM-4063: Update QAN filter panel to show only labels available for selection under currently applied filters
- PMM-815: Latency Detail graph added to the MongoDB Instance Summary dashboard
- PMM-4768: Disable heavy-load collectors automatically when there are too many tables
- PMM-4821: Use color gradient in filled graphs on all dashboards
- PMM-4733: Add more log and configuration files to the downloadable `logs.zip` archive
- PMM-4672: Use integer percentage values in QAN filter panel
- PMM-4857: Update tooltips for all MongoDB dashboards
- PMM-4616: Rename column in the Query Details section in QAN from Total to Sum
- PMM-4770: Use Go 1.12.10
- PMM-4780: Update Grafana to version 6.4.1
- PMM-4918: Update Grafana plugins to newer versions, including the `clickhouse-datasource` plugin

9.56.2 Fixed bugs

- PMM-4935: Wrong instance name displayed on the MySQL Instance Summary dashboard due to the incorrect string crop
- PMM-4916: Wrong values are shown when changing the time range for the Node Summary Dashboard in case of remote instances
- PMM-4895 and PMM-4814: The update process reports completion before it is actually done and therefore some dashboards, etc. may not be updated
- PMM-4876: PMM Server access credentials are shown by the `pmm-admin status` command instead of hiding them for security reasons
- PMM-4875: PostgreSQL error log gets flooded with warnings when `pg_stat_statements` extension is not installed in the database used by PMM Server or when PostgreSQL user is unable to connect to it
- PMM-4852: Node name has an incorrect value if the Home dashboard opened after QAN
- PMM-4847: Drill-downs from the Environment Overview dashboard doesn't show data for the preselected host
- PMM-4841 and PMM-4845: `pg_stat_statement` QAN Agent leaks database connections
- PMM-4831: Clean-up representation of selectors names on MySQL-related dashboards for a better consistency
- PMM-4824: Incorrectly calculated singlestat values on MySQL Instances Overview dashboard
- PMM-4819: In case of the only one monitored host, its uptime is shown as a smaller value than the all hosts uptime due to the inaccurate rounding
- PMM-4816: Set equal thresholds to avoid confusing singlestat color differences on a Home dashboard
- PMM-4718: Labels are not fully displayed in the filter panel of the Query Details section in QAN
- PMM-4545: Long queries are not fully visible in the Query Examples section in QAN

Percona LLC and/or its affiliates ©2024

Help us improve our software quality by reporting any Percona Monitoring and Management bugs you encounter using our bug tracking system.

9.56.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.57 Percona Monitoring and Management 2.0.1

Date: October 9, 2019

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance. You can run PMM in your own environment for maximum security and reliability. It provides thorough time-based analysis for MySQL, MongoDB, and PostgreSQL servers to ensure that your data works as efficiently as possible.

For install instructions, see Installing Percona Monitoring and Management.

Caution

PMM 2 is designed to be used as a new installation — please don't try to upgrade your existing PMM 1 environment.

9.57.1 Improvements

- PMM-4779: Securely share dashboards with Percona
- PMM-4735: Keep one old slowlog file after rotation
- PMM-4724: Alt+click on check updates button enables force-update
- PMM-4444: Return "what's new" URL with the information extracted from the pmm-update package change log

9.57.2 Fixed bugs

- PMM-4758: Remove Inventory rows from dashboards
- PMM-4757: `qan_mysql_perfschema_agent` failed querying `events_statements_summary_by_digest` due to data types conversion
- PMM-4755: Fixed a typo in the InnoDB AHI Miss Ratio formula
- PMM-4749: Navigation from Dashboards to QAN when some Node or Service was selected now applies filtering by them in QAN
- PMM-4742: General information links were updated to go to PMM 2 related pages
- PMM-4739: Remove request instances list
- PMM-4734: A fix was made for the collecting `node_name` formula at MySQL Replication Summary dashboard
- PMM-4729: Fixes were made for formulas on MySQL Instances Overview
- PMM-4726: Links to services in MongoDB singlestats didn't show Node name

Percona LLC and/or its affiliates ©2024

- PMM-4720: `machine_id` could contain trailing `\\n`
- PMM-4640: It was not possible to add MongoDB remotely if password contained a `#` symbol

Help us improve our software quality by reporting any Percona Monitoring and Management bugs you encounter using our bug tracking system.

9.57.3 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2021-06-25

9.58 Percona Monitoring and Management 2.0.0

Date: September 19, 2019

Percona Monitoring and Management (PMM) is a free and open-source platform for managing and monitoring MySQL, MongoDB, and PostgreSQL performance. You can run *PMM* in your own environment for the best security and reliability. It provides thorough time-based analysis for MySQL, MongoDB, and PostgreSQL servers to ensure that your data works as efficiently as possible.

For installation instructions, see Installing Percona Monitoring and Management.

Caution

PMM 2 is designed to be used as a new installation — please don't try to upgrade your existing PMM 1 environment.

The new PMM2 introduces a number of enhancements and additional feature improvements, including:

- Detailed query analytics and filtering technologies enable you to identify issues faster than ever before.
- A better user experience: Service-level dashboards give you immediate access to the data you need.
- The new addition of PostgreSQL query tuning.
- Enhanced security protocols to ensure your data is safe.
- Our new API allows you to extend and interact with third-party tools.

More details about new and improved features available within the release can be found in the corresponding blog post.

Help us improve our software quality by reporting any Percona Monitoring and Management bugs you encounter using our bug tracking system.

9.58.1 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Percona LLC and/or its affiliates ©2024

Community Forum Get a Percona Expert

2023-02-23

Percona LLC and/or its affiliates ©2024

10. Reference

10.1 Copyright and licensing information

10.1.1 Documentation licensing

Percona Monitoring and Management documentation is (C)2009-2024 Percona LLC and/or its affiliates and is distributed under the Creative Commons Attribution 4.0 International License.

10.1.2 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2024-02-02

10.2 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.

Percona LLC and/or its affiliates ©2024

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.

10.2.1 Get expert help

If you need assistance, visit the community forum for comprehensive and free database knowledge, or contact our Percona Database Experts for professional support and services.

Community Forum Get a Percona Expert

2023-06-27

Percona LLC and/or its affiliates ©2024