# Percona Monitoring & Management

## Installing and Basic Architecture

**Max Bubenick**

April 2019 @ Percona University BA

# What is PMM?

Percona Monitoring and Management (PMM) is an open-source platform for managing and monitoring MySQL, MongoDB, PostgreSQL and ProxySQL performance. It is developed by Percona in collaboration with experts in the field of managed database services, support and consulting.

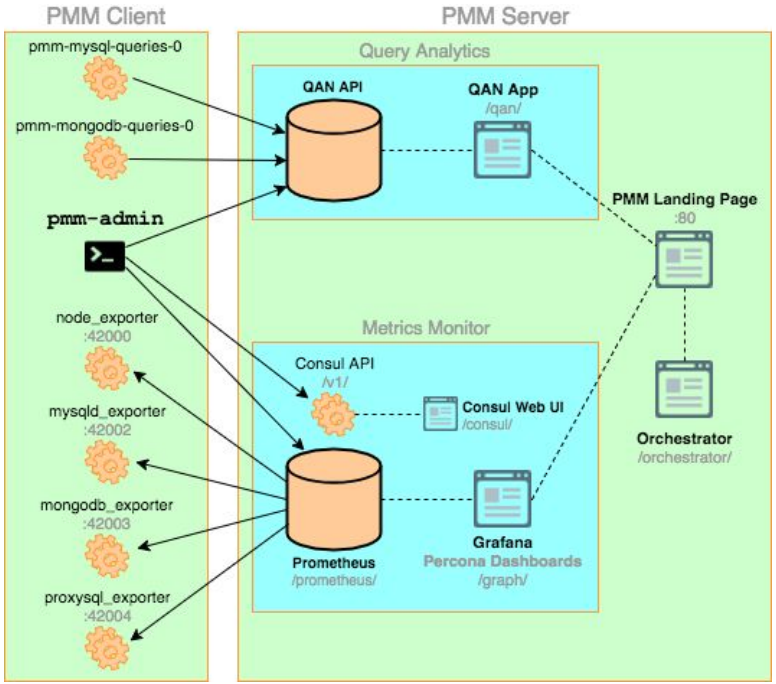
PMM is a free and open-source solution that you can run in your own environment for maximum security and reliability. It provides thorough time-based analysis for DB servers to ensure that your data works as efficiently as possible.

PERCONA
Monitoring and Management

# PMM Architecture

- PMM Client installed on every database host that you want to monitor. It collects server metrics, general system metrics, and Query Analytics data for a complete performance overview.

- PMM Server is the central part of PMM that aggregates collected data and presents it in the form of tables, dashboards, and graphs in a web interface.



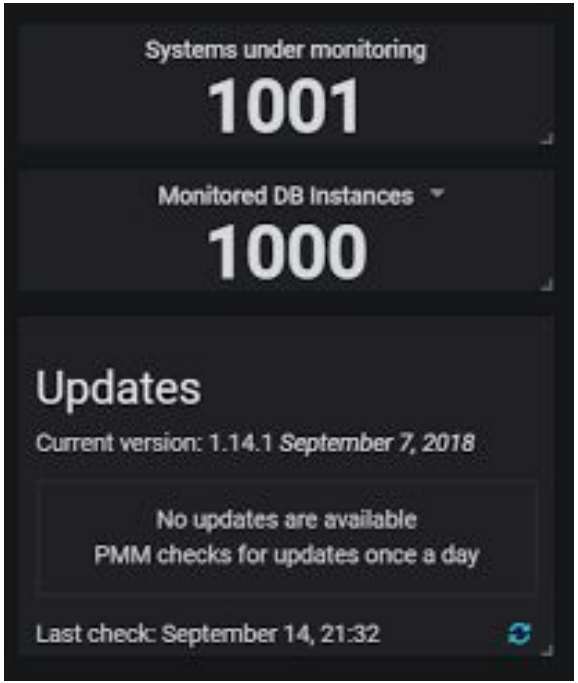



PERCONA

# Deploying PMM

PMM is designed to be scalable for various environments. If you have just one MySQL or MongoDB server, you can install and run both PMM server and PMM clients on one database host.

It is more typical to have several MySQL and MongoDB server instances distributed over different hosts. In this case, you need to install the PMM client package on each database host that you want to monitor. In this scenario, the PMM server is set up on a dedicated monitoring host.



# PMM Requirements

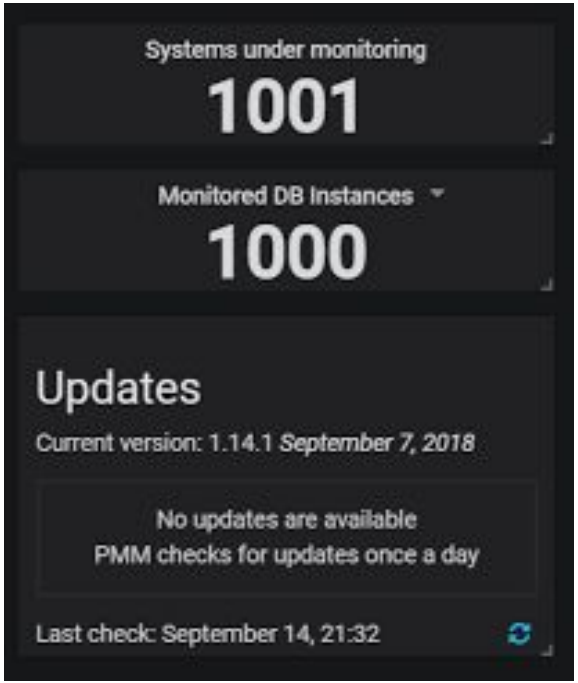
**Ports**
The following ports must be open to enable communication between the PMM Server and PMM clients.

**PMM Client -> PMM Server**

PMM Server should keep ports 80 or 443 ports open for computers where PMM Client is installed to access the PMM web interface.

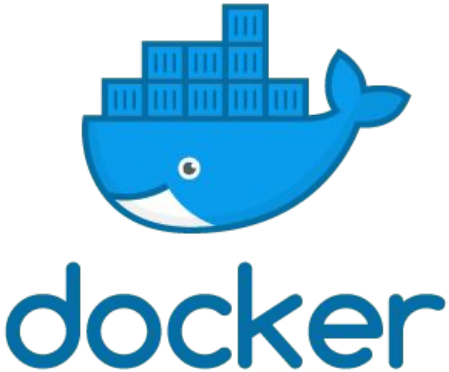
**PMM Server -> PMM Client**

**42000** For PMM to collect general linux system metrics.
**42001** This port is used by a service which collects query performance data and makes it available to QAN.
**42002** For PMM to collect MySQL server metrics.
**42003** For PMM to collect MongoDB server metrics.
**42004** For PMM to collect ProxySQL server metrics.
**42005** For PMM to collect PostgreSQL server metrics.



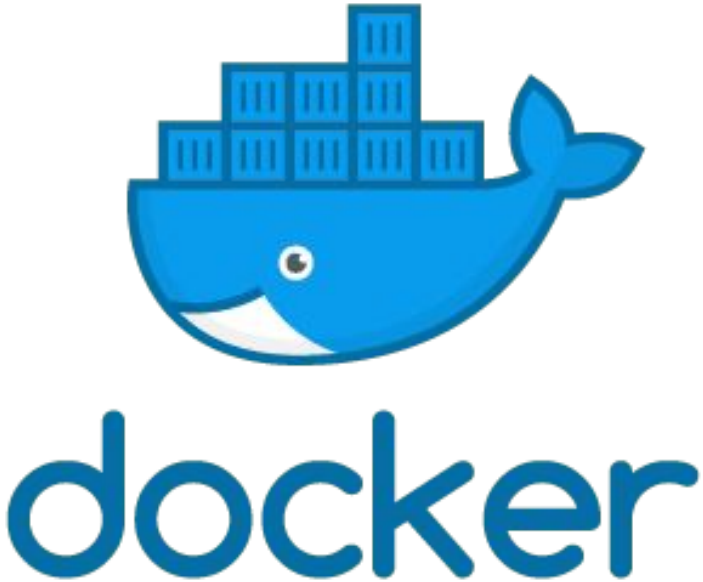
# Installing PMM Server

3 Different options:





# Installing PMM Server



# Installing Docker PMM Server

## Requirements

### Specs

1GB storage for each node and week retention
2GB memory per node, not linear (16GB can handle 20 nodes)

### Software

Docker >= 1.12.6

### Networking

https://hub.docker.com/r/percona/pmm-server/tags/

# Installing Docker PMM Server

## Pull docker image

```
$ docker pull percona/pmm-server:1
```

If you want always the latest version

Or

```
$ docker pull percona/pmm-server:1.17.1
```

If you want a specific version

# Installing Docker PMM Server

## Create data container

```
$ docker create \
   -v /opt/prometheus/data \
   -v /opt/consul-data \
   -v /var/lib/mysql \
   -v /var/lib/grafana \
   --name pmm-data \
   percona/pmm-server:1.17.1 /bin/true
```

Persists pmm data so you can retain history when upgrading to a new version

# Installing Docker PMM Server

Create and launch pmm-server container

```
$ docker run -d \
   -p 8080:80 \
   --volumes-from pmm-data \
   --name pmm-server \
   --restart always \
    percona/pmm-server:1.17.1
```

- Run detached
- Maps port (8080 landing page)
- Mount volumes from pmm-data container
- Docker daemon to [re]start pmm-server
- Image from 1.17.1

# Installing Docker PMM Server

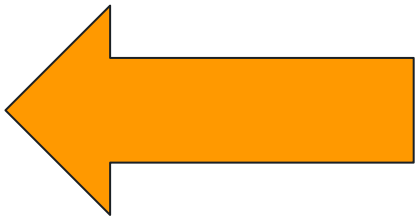
## Some PMM-Server Parameters

- **DISABLE_UPDATES:** removes the Update button from the interface
- **METRICS_MEMORY:** Prometheus memory, deprecated since 1.13 due to prometheus 2
- **METRICS_RESOLUTION:** need to be set if latency is higher than 1s
- **METRICS_RETENTION:** combination of hours, minutes, and seconds (i.e 192h30m0s)
- **SERVER_PASSWORD:** server frontend password
- **SERVER_USER:** server frontend user
- **QUERIES_RETENTION**: how many days queries are stored

# Installing Docker PMM Server

## Passing PMM-Server Parameters

```
$ docker run -d \
   -p 8080:80 \
   --volumes-from pmm-data \
   --name pmm-server \
   --restart always \
   -e SERVER_USER=pmm \
   -e SERVER_PASSWORD=change_me \
   -e METRICS_RETENTION=720h \
   -e METRICS_RESOLUTION=1s \
   -e DISABLE_UPDATES=1 \
   -e QUERIES_RETENTION=8 \
    percona/pmm-server:1.17.1
```



# Installing PMM Client

PMM Client is a package of agents and exporters installed on a database host that you want to monitor.

### DEB based GNU/Linux distros

```
$ wget https://repo.percona.com/apt/percona-release_latest.generic_all.deb
$ sudo dpkg -i percona-release_latest.generic_all.deb
# apt-get install pmm-client
```

### RPM based GNU/Linux distros

```
$ sudo yum install https://repo.percona.com/yum/percona-release-latest.noarch.rpm
# yum install pmm-client
```

# Installing PMM Client

Connecting pmm-client to server

```
$ sudo pmm-admin config --server 192.168.2.11:8080 --server-password change_me
```

```
$ sudo pmm-admin info
pmm-admin 1.17.1

PMM Server      | 192.168.2.11:8080 (password-protected)
Client Name     | mysql-01
Client Address  | 192.168.2.12
Service Manager | linux-systemd

Go Version      | 1.10.1
Runtime Info    | linux/amd64
```

# Installing PMM Client

Add monitoring services

```
Usage:
  pmm-admin add [command]

Available Commands:
  linux:metrics      Add this system to metrics monitoring.
  mysql              Add complete monitoring for MySQL instance (linux and mysql metrics, queries).
  mysql:metrics      Add MySQL instance to metrics monitoring.
  mysql:queries      Add MySQL instance to Query Analytics.
  mongodb            Add complete monitoring for MongoDB instance (linux and mongodb metrics, queries).
  mongodb:metrics    Add MongoDB instance to metrics monitoring.
  mongodb:queries    Add MongoDB instance to Query Analytics.
  postgresql         Add complete monitoring for PostgreSQL instance (linux and postgresql metrics).
  postgresql:metrics Add PostgreSQL instance to metrics monitoring.
  proxysql           Add complete monitoring for ProxySQL instance (linux and proxysql metrics).
  proxysql:metrics   Add ProxySQL instance to metrics monitoring.
```

# Installing PMM Client

## Configuring MySQL for PMM

Creating pmm MySQL user

```
mysql> GRANT SELECT, PROCESS, SUPER, REPLICATION CLIENT, RELOAD ON *.* TO 'pmm'@' localhost' IDENTIFIED BY 'changeM3!'
WITH MAX_USER_CONNECTIONS 10;
mysql> GRANT SELECT, UPDATE, DELETE, DROP ON performance_schema.* TO 'pmm'@'localhost';
```

# Installing PMM Client

## Configuring MySQL for PMM QAN

Configs for QAN (Slowlog recommended for Percona Server)

```
#QAN
log_output='file'
log_slow_rate_type='query'
log_slow_rate_limit=100
long_query_time=0
log_slow_verbosity='full'
log_slow_admin_statements=ON
log_slow_slave_statements=ON
slow_query_log_use_global_control='all'
slow_query_log_always_write_time=1
slow_query_log=ON
```

# Installing PMM Client

## Configuring MySQL for PMM QAN

Configs for QAN (Perf Schema)

```
#QAN
performance_schema=ON
```

```
mysql> select * from performance_schema.setup_consumers where NAME='statements_digest';
+------------------+---------+
| NAME             | ENABLED |
+------------------+---------+
| statements_digest | YES     |
+------------------+---------+
```

# Installing PMM Client

Adding linux:metrics service

```
$ sudo pmm-admin add linux:metrics
```

```
Flags:
      --disable-ssl   disable ssl mode on exporter (default true)
      --force         force to add another linux:metrics instance with different name for testing purposes
  -h, --help          help for linux:metrics
```

# Installing PMM Client

Adding mysql:metrics service

```
$ sudo pmm-admin add mysql:metrics --defaults-file /usr/local/percona/.my.cnf

 Flags:
       --create-user                          create a new MySQL user
       --create-user-maxconn uint16           max user connections for a new user (default 10)
       --create-user-password string          optional password for a new MySQL user
       --defaults-file string                 path to my.cnf
       --disable-binlogstats                  disable binlog statistics
       --disable-processlist                  disable process state metrics
       --disable-ssl                          disable ssl mode on exporter
       --disable-tablestats                   disable table statistics
       --disable-tablestats-limit uint16      number of tables after which table stats are disabled automatically
       --disable-userstats                    disable user statistics
       --force                                force to create/update MySQL user
       --host string                          MySQL host
       --password string                      MySQL password
       --port string                          MySQL port
       --socket string                        MySQL socket
       --user string                          MySQL username
```

# Installing PMM Client

Adding mysql:queries service (QAN)

```
$ sudo pmm-admin add mysql:queries --defaults-file /usr/local/percona/.my.cnf --query-source slowlog
```

```
Flags:
      --create-user                    create a new MySQL user
      --create-user-maxconn uint16     max user connections for a new user (default 10)
      --create-user-password string    optional password for a new MySQL user
      --defaults-file string           path to my.cnf
      --disable-queryexamples          disable collection of query examples
      --disable-ssl                    disable ssl mode on exporter
      --force                          force to create/update MySQL user
      --host string                    MySQL host
      --password string                MySQL password
      --port string                    MySQL port
      --query-source string            source of SQL queries: auto, slowlog, perfschema (default "auto")
      --retain-slow-logs int           number of slow logs to retain after rotation (default 1)
      --slow-log-rotation              enable slow log rotation (default true)
      --socket string                  MySQL socket
      --user string                    MySQL username
```

# Troubleshooting PMM Client

```
$ sudo pmm-admin info
pmm-admin 1.17.1

PMM Server      | 192.168.2.11:8080 (password-protected)
Client Name     | mysql
Client Address  | 192.168.2.12
Service Manager | linux-systemd

Go Version      | 1.10.1
Runtime Info    | linux/amd64

$ sudo pmm-admin show-passwords
HTTP basic authentication
User     | pmm
Password | change_me

MySQL new user creation
Password |
```

# Troubleshooting PMM Client

```
$ sudo pmm-admin check-network
PMM Network Status

Server Address | 192.168.2.11:8080
Client Address | 192.168.2.12

* System Time
NTP Server (0.pool.ntp.org)         | 2019-04-23 16:49:38 +0000 UTC
PMM Server                          | 2019-04-23 02:08:08 +0000 GMT
PMM Client                          | 2019-04-23 02:08:01 +0000 UTC
PMM Server Time Drift               | 52890s
Time is out of sync. Please make sure the server time is correct to see the metrics.
PMM Client Time Drift               | 52897s
Time is out of sync. Please make sure the client time is correct to see the metrics.
PMM Client to PMM Server Time Drift | OK
```

# Troubleshooting PMM Client

```
* Connection: Client --> Server
-------------------- -------
SERVER SERVICE       STATUS
-------------------- -------
Consul API           OK
Prometheus API       OK
Query Analytics API  OK

Connection duration | 1.669109ms
Request duration    | -563.149µs
Full round trip     | 1.10596ms


* Connection: Client <-- Server
-------------- ------ ------------------- ------- ---------- ---------
SERVICE TYPE   NAME   REMOTE ENDPOINT     STATUS  HTTPS/TLS  PASSWORD
-------------- ------ ------------------- ------- ---------- ---------
linux:metrics  mysql  192.168.2.12:42000  OK      YES        YES
mysql:metrics  mysql  192.168.2.12:42002  OK      YES        YES
```

# Troubleshooting PMM Client

```
$ sudo pmm-admin list
pmm-admin 1.17.1

PMM Server      | 192.168.2.11:8080 (password-protected)
Client Name     | mysql
Client Address  | 192.168.2.12
Service Manager | linux-systemd

-------------- ------ ----------- -------- ----------------------------------------
------------------------------------------------------------------------------------
SERVICE TYPE   NAME   LOCAL PORT  RUNNING  DATA SOURCE                              OPTIONS
-------------- ------ ----------- -------- ----------------------------------------
------------------------------------------------------------------------------------
mysql:queries  mysql  -           YES      pmm:***@unix(/var/lib/mysql/mysql.sock)  query_source=slowlog,
query_examples=true, slow_log_rotation=true, retain_slow_logs=1
linux:metrics  mysql  42000       YES      -
mysql:metrics  mysql  42002       YES      pmm:***@unix(/var/lib/mysql/mysql.sock)
```

# Live demo

# Thank you!

# Q&Y

Links:

https://github.com/maxbube/pmm-vagrant-demo
https://pmmdemo.percona.com

Contact Me:

E-mail: max.bubenick@percona.com
Twitter: @maxbube