Percona

Working with Percona

Step 1 — Install Percona

- Update the package index

```
sudo apt update
```

- Install Curl packages

```
sudo apt install -y curl
```

- Download the latest release of percona

```
curl -O https://repo.percona.com/apt/percona-release_latest.generic_all.deb
```

- Install packages and dependencies

```
sudo apt install -y gnupg2 lsb-release ./percona-release_latest.generic_all.deb
```

- Update the packages

```
sudo apt update
```

- Setup specific percona server version

```
sudo percona-release setup ps80
```

- Enable percona server

```
sudo percona-release enable ps-80 release
```

```
sudo apt update
```

- Install percona-server package (setup password)

```
sudo apt install -y percona-server-server
```

- After installing percona, check service status

```
sudo systemctl status mysql
```

- Output:

> ● mysql.service - Percona Server
>   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
>   Active: active (running) since Thu 2024-09-19 08:18:53 +0530; 7h ago
>   Process: 674 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
>   Main PID: 862 (mysqld)
>   Status: "Server is operational"
>   Tasks: 54 (limit: 18891)
>   Memory: 404.5M
>   CPU: 3min 24.976s
>   CGroup: /system.slice/mysql.service
>   └─862 /usr/sbin/mysqld

- Install toolkit

```
apt list | grep percona
```

```
libperconaserverclient21-dev/unknown 8.0.37-29-1.focal amd64
libperconaserverclient21-zenfs-dev/unknown 8.0.26-16-1.focal amd64
libperconaserverclient21-zenfs/unknown 8.0.26-16-1.focal amd64
libperconaserverclient21/unknown 8.0.37-29-1.focal amd64
percona-backup-mongodb/unknown 2.6.0-1.focal amd64
percona-mysql-router-zenfs/unknown 8.0.26-16-1.focal amd64
percona-mysql-router/unknown 8.0.37-29-1.focal amd64
percona-mysql-shell/unknown 8.0.37-1-1.focal amd64
percona-release/stable,now 1.0-29.generic all [installed]
percona-server-client-zenfs/unknown 8.0.26-16-1.focal amd64
percona-server-client/unknown,now 8.0.37-29-1.focal amd64 [installed,automatic]
percona-server-common-zenfs/unknown 8.0.26-16-1.focal amd64
percona-server-common/unknown,now 8.0.37-29-1.focal amd64 [installed,automatic]
percona-server-dbg-zenfs/unknown 8.0.26-16-1.focal amd64
percona-server-dbg/unknown 8.0.37-29-1.focal amd64
percona-server-rocksdb-zenfs/unknown 8.0.26-16-1.focal amd64
percona-server-rocksdb/unknown 8.0.37-29-1.focal amd64
percona-server-server-zenfs/unknown 8.0.26-16-1.focal amd64
percona-server-server/unknown,now 8.0.37-29-1.focal amd64 [installed]
percona-server-source-zenfs/unknown 8.0.26-16-1.focal amd64
percona-server-source/unknown 8.0.37-29-1.focal amd64
percona-server-test-zenfs/unknown 8.0.26-16-1.focal amd64
```

```
percona-server-test/unknown 8.0.37-29-1.focal amd64

percona-server-tokudb-zenfs/unknown 8.0.26-16-1.focal amd64

percona-server-tokudb/unknown 8.0.27-18-1.focal amd64

percona-telemetry-agent/stable,now 1.0.2-2.focal amd64 [installed,automatic]

percona-toolkit/unknown 3.6.0-1.focal amd64

percona-xtrabackup-24/unknown 2.4.29-1.focal amd64

percona-xtrabackup-80/unknown 8.0.35-31-1.focal amd64

percona-xtrabackup-81/unknown 8.1.0-1-1.focal amd64

percona-xtrabackup-82/unknown 8.2.0-1-1.focal amd64

percona-xtrabackup-83/unknown 8.3.0-1-1.focal amd64

percona-xtrabackup-dbg-24/unknown 2.4.29-1.focal amd64

percona-xtrabackup-dbg-80/unknown 8.0.35-31-1.focal amd64

percona-xtrabackup-dbg-81/unknown 8.1.0-1-1.focal amd64

percona-xtrabackup-dbg-82/unknown 8.2.0-1-1.focal amd64

percona-xtrabackup-dbg-83/unknown 8.3.0-1-1.focal amd64

percona-xtrabackup-test-24/unknown 2.4.29-1.focal amd64

percona-xtrabackup-test-80/unknown 8.0.35-31-1.focal amd64

percona-xtrabackup-test-81/unknown 8.1.0-1-1.focal amd64

percona-xtrabackup-test-82/unknown 8.2.0-1-1.focal amd64

percona-xtrabackup-test-83/unknown 8.3.0-1-1.focal amd64
```

- Install -80 for MySQL 8.0 & -24 for MySQL 5.7

```
sudo apt install percona-xtrabackup-80
```

Step 2 — Replication Configuration on Primary Server

- install nano

```
sudo apt-get install nano
```

- Edit my.cnf file

```
sudo nano /etc/mysql/my.cnf
```

```
[mysqld]
server-id = 1
log_bin = /var/log/mysql/mysql-bin.log
binlog_format = ROW
innodb_flush_log_at_trx_commit = 1
sync_binlog = 1
```

- Restart MySQL server

```
sudo systemctl restart mysql
```

-
  ```
  mysql -u root -p
  ```

-
  ```
  CREATE USER 'replica_user'@'%' IDENTIFIED BY 'strongpassword';
  GRANT REPLICATION SLAVE ON *.* TO 'replica_user'@'%';
  FLUSH PRIVILEGES;
  ```

- FLUSH TABLES WITH READ LOCK;

  SHOW MASTER STATUS;

- ```
  +------------------+----------+--------------+------------------+-------------------+
  | File             | Position | Binlog_Do_DB | Binlog_Ignore_DB | Executed_Gtid_Set |
  +------------------+----------+--------------+------------------+-------------------+
  | mysql-bin.000007 |   291136 |              |                  |                   |
  +------------------+----------+--------------+------------------+-------------------+
  ```

- UNLOCK TABLES;

- Go to the temp folder and export the database

```
mysqldump -u root -p --all-databases --master-data > db_dump.sql
```

- Export the specific table

```
mysqldump -u root -p your_database your_table > /tmp/your_table.sql
```

Unlock tables

```
UNLOCK TABLES;
```

- Transfer the database/table to the slave server

```
scp /tmp/your_table.sql user@slave_server:/tmp/
```

Step 3 — Replication Configuration on Replication Server

- On Replication Server

- ```
  sudo nano /etc/mysql/my.cnf
  ```

- ```
  [mysqld]
  server-id = 2
  relay_log = /var/log/mysql/mysql-relay-bin.log
  replicate_do_db = your_database
  replicate_do_table = your_database.your_table
  ```

-

Step 4 — Master Replication over SSL

Check following attributes on both master and replication servers.

```
sudo nano /etc/mysql/my.cnf
```

On Master server

```
[mysqld]

server-id = 1
replicate-same-server-id = 0
auto-increment-increment = 2
auto-increment-offset = 1

relay-log = /var/lib/mysql/relay-bin
relay-log-index = /var/lib/mysql/relay-bin.index
relay-log-info-file = /var/lib/mysql/relay-log.info

log-error = /var/lib/mysql/mysql.err
master-info-file = /var/lib/mysql/mysql-master.info
log-bin = /var/lib/mysql/bin
```

On Replication Server

```
[mysqld]

server-id = 2
replicate-same-server-id = 0
auto-increment-increment = 2
auto-increment-offset = 1

relay-log = /var/lib/mysql/relay-bin
relay-log-index = /var/lib/mysql/relay-bin.index
relay-log-info-file = /var/lib/mysql/relay-log.info
```

```
log-error = /var/lib/mysql/mysql.err
master-info-file = /var/lib/mysql/mysql-master.info
log-bin = /var/lib/mysql/bin
```

Restart MySQL on both servers

```
systemctl restart mysql
```

On Master server create a directory to save certificates

```
sudo chmod -R 777 /var/lib/
mkdir /var/lib/mysql-certs
cd /var/lib/mysql-certs/
```

CA cert

```
openssl genrsa 2048 > MySQLCAKey.pem
```

```
Generating RSA private key, 2048 bit long modulus (2 primes)
..........+++++
.................................................................+++++
e is 65537 (0x010001)
```

```
openssl req -new -x509 -nodes -days 3600 -key MySQLCAKey.pem -out MySQLCA-Cert.pem
```

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:MySQL Replication CA
```

Email Address []:

Percona Toolkit

Experimenting each percona toolkit tools to efficiently replicate from the master to slave server

1. Prerequestics

Test Database

- Download sakila db

  https://dev.mysql.com/doc/index-other.html

- Extract the installation archive to a temporary location
- Connect to the MySQL server

  mysql -uroot -p

- Execute the sakila-schema.sql script to create the database structure, and execute the sakila-data.sql script

  SOURCE /home/<usr>/Downloads/sakila-db/sakila-schema.sql;

  SOURCE /home/<usr>/Downloads/sakila-db/sakila-data.sql;

2. pt-archive

In this test case, data exchange from one table to another in the same db on the same server

- Connect to the MySQL server

  ```
  mysql -uroot -p
  ```

- Create a new table

  ```
  USE sakila;

  CREATE TABLE old_rental LIKE rental;

  SELECT COUNT(*) FROM rental WHERE rental_date < "2006-01-01";

  SELECT * FROM old_rental;
  ```

- Locate **pt-archiver**

  ```
  find / -name pt-archiver 2>/dev/null
  ```

- This will show the location of the pt-archiver

  ```
  /usr/bin/pt-archiver
  ```

- using pt-archiver command

  ```
  /usr/bin/pt-archiver --source D=sakila,t=rental --dest D=sakila,t=old_rental --where "rental_date <
  '2006-01-01'" --statistics --user <user> --password <password> --host <host>
  ```

- user and hosts examples

User	From Host
cosm	%
cusm	%
debian-sys-ma	localhost
dsm	%
lmsv1	%
mysql.infosche	localhost
mysql.session	localhost
mysql.sys	localhost
replica_user	%
root	localhost
sakila	%
srr	%

- output of the above pt-archiver command

Started at 2024-10-08T10:02:00, ended at 2024-10-08T10:07:33

Source: D=sakila,p=...,t=rental,u=sakila

Dest: D=sakila,p=...,t=old_rental,u=sakila

SELECT 15861

INSERT 15861

DELETE 15861

Action	Count	Time	Pct
commit	31724	311.6698	93.55
select	15862	6.5446	1.96
deleting	15861	6.3438	1.90
inserting	15861	5.2972	1.59
other	0	3.3111	0.99

- **This will remove the data from the source table and insert them in the target table**

pt-summary

- Summarizes the status and configuration of a server
- Locate the pt-summary

```
find / -name pt-summary 2>/dev/null
```

```
usr/bin/pt-summary
```

- Usage

```
sudo /usr/bin/pt-summary | less
```

- Output

```
# Percona Toolkit System Summary Report ####################
        Date | 2024-10-08 05:36:02 UTC (local TZ: +0530 +0530)
    Hostname | dev2-Latitude-E7270
      Uptime |  2:57,  2 users,  load average: 0.89, 1.44, 1.61
      System | Dell Inc.; Latitude E7270; vNot Specified (Laptop)
 Service Tag | BFKTR72
    Platform | Linux
     Release | Ubuntu 22.04.5 LTS (jammy)
      Kernel | 6.8.0-45-generic
Architecture | CPU = 64-bit, OS = 64-bit
   Threading | NPTL 2.35
      SELinux | No SELinux detected
  Virtualized | No virtualization detected
# Processor #################################################
  Processors | physical = 1, cores = 2, virtual = 4, hyperthreading = yes
      Speeds | 1x2898.822, 1x2900.040, 1x2914.082, 1x2986.415
      Models | 4xIntel(R) Core(TM) i5-6300U CPU @ 2.40GHz
      Caches | 4x3072 KB
 Designation        Configuration        Size    Associativity
 ========================= ============================
 ======== =====================
  L1 Cache        Enabled, Not Socketed, Level 1 64 kB   8-way Set-associative
  L1 Cache        Enabled, Not Socketed, Level 1 64 kB   8-way Set-associative
```

L2 Cache	Enabled, Not Socketed, Level 2 512 kB 4-way Set-associative
L3 Cache	Enabled, Not Socketed, Level 3 3 MB 12-way Set-associative

Memory

Total	15.4G
Free	1.5G
Used	physical = 5.9G, swap allocated = 2.0G, swap used = 0.0, virtual = 5.9G
Shared	1.1G
Buffers	8.1G
Caches	8.1G
Dirty	1344 kB
UsedRSS	13.7G
Swappiness	60

: