Marcelo Altmann

- Senior Software Engineer @ Percona
- Working on Xtrabackup Project
- Author of key features
  - FIFO Datasink / xbcloud multi-thread
  - Memory Estimation
  - ZSTD Compression Support
  - Xbcloud exponential Backoff
  - Xbcloud instance profile
  - KMIP & KMS Keyring Component Integration

Vinicius Grippa

- Senior Support Engineer @ Percona

Content

Before we start

- Launch tutorial docker container:

```
Unset
docker run -d -t -i --privileged --name percona_live -p 9091:9091 \
altmannmarcelo/pxb_pl:latest
```

- Check container is up and running:

```
Unset
docker logs -n 1 percona_live
```

We expect to see: **Everything is up and running.**

- Enter container (launch at least two sessions):

```
Unset
docker exec -it percona_live bash
```

- MySQL **root** user is installed with **auth_socket** (no password)
- There is a **run_load.sh** script to simulate load during backups
- We will make usage of debug sync points:
  - Way of stopping the program at a specific point in the code:
  - Widely used on test suite to make certain conditions deterministically
  - Only available on Debug builds
  - **NOT SUITABLE FOR PRODUCTION**
  - Example:
    https://github.com/percona/percona-xtrabackup/blob/percona-xtrabackup-8.0.32-26/storage/innobase/xtrabackup/src/backup_mysql.cc#L1663

Full Backup

- Check there is no employee named **Roy**:

```
mysql> SELECT * FROM employees.employees WHERE first_name = 'Roy'\G
Empty set (0.09 sec)
```

- Start to simulate load:

```
run_load.sh run
```

- Take a full backup:

```
xtrabackup --backup --target-dir=/backups/full \
--register-redo-log-consumer --debug_sync='log_status_get'
```

We should see the backup has been stopped at **log_status_get** sync point and a kill command to resume the backup:

```
2023-04-28T15:22:40.613320-00:00 0 [Note] [MY-011825] [Xtrabackup] Selecting LSN and
binary log position from p_s.log_status
2023-04-28T15:22:40.613388-00:00 0 [Note] [MY-011825] [Xtrabackup] DEBUG: Suspending
at debug sync point 'log_status_get'Resume with 'kill -SIGCONT 173'

[1]+  Stopped                 xtrabackup --backup --target-dir=/backups/full
--debug_sync='log_status_get'
```

- Check table employees has already been copied:

```
Unset
ls -l /backups/full/employees/employees.ibd
```

- Roy was hired. Add him to the database:

```
Unset
mysql> INSERT INTO employees.employees VALUES (500000, '1988-05-21',
'Roy', 'Trenneman', 'M', '2023-05-22');
```

- Resume backup:

```
Unset
kill -SIGCONT 173
```

Wait for Xtrabackup to complete

- Stop load script
- Stop MySQL and remove datadir:

```
Unset
mysql -e shutdown
rm -rf /var/lib/mysql
```

- Prepare backup

```
Unset
xtrabackup --prepare --target-dir=/backups/full
```

- Copy backup to datadir and adjust folder ownership:

```
Unset
xtrabackup --copy-back --target-dir=/backups/full
chown -R mysql.mysql /var/lib/mysql
```

- Start MySQL

```
Unset
mysqld --user=mysql &
```

- Check if Roy is an employee:

```
Unset
mysql> SELECT * FROM employees.employees WHERE first_name = 'Roy'\G
```

```
*************************** 1. row ***************************
     emp_no: 500000
birth_date: 1988-05-21
first_name: Roy
 last_name: Trenneman
     gender: M
 hire_date: 2023-05-22
1 row in set (0.09 sec)
```

Incremental Backup

- Install page tracking component

```
Unset
mysql> INSTALL COMPONENT "file://component_mysqlbackup";
```

- Check there is no employee named **Maurice** nor **Jen**

```
Unset
mysql> SELECT * FROM employees.employees WHERE first_name = 'Maurice'
OR first_name = 'Jen'\G
Empty set (0.09 sec)
```

- Start to simulate load:

```
Unset
run_load.sh run
```

- Take a full backup

```
Unset
xtrabackup --backup --page-tracking --target-dir=/backups/inc_full
```

Wait for backup to complete

- Maurice was hired. Add him to the database:

```
Unset
mysql> INSERT INTO employees.employees VALUES (500001, '1988-05-21',
'Maurice', 'Moss', 'M', '2023-05-22');
```

- Take the first incremental backup

```
Unset
xtrabackup --backup --page-tracking --target-dir=/backups/inc_1 \
--incremental-basedir=/backups/inc_full
```

- Jen was hired. Add her to the database

```
Unset
mysql> INSERT INTO employees.employees VALUES (500002, '1988-05-21',
'Jen', 'Barber', 'F', '2023-05-22');
```

- Take the second incremental backup

```
xtrabackup --backup --page-tracking --target-dir=/backups/inc_2 \
--incremental-basedir=/backups/inc_1
```

- Stop load script
- Stop MySQL and remove datadir:

```
mysql -e shutdown
rm -rf /var/lib/mysql
```

- Prepare **inc_full** backup

```
xtrabackup --prepare --apply-log-only --target-dir=/backups/inc_full
```

- Prepare **inc_1** backup

```
xtrabackup --prepare --apply-log-only --target-dir=/backups/inc_full \
--incremental-dir=/backups/inc_1
```

- Prepare **inc_2** backup

```
Unset
xtrabackup --prepare --target-dir=/backups/inc_full \
--incremental-dir=/backups/inc_2
```

- Copy backup to datadir and adjust folder ownership:

```
Unset
xtrabackup --copy-back --target-dir=/backups/inc_full
chown -R mysql.mysql /var/lib/mysql
```

- Start MySQL

```
Unset
mysqld --user=mysql &
```

- Check there is employee named **Maurice** nor **Jen**

```
Unset
mysql> SELECT * FROM employees.employees WHERE first_name = 'Maurice'
OR first_name = 'Jen'\G
```

Streaming

- Check MinIO console access: http://127.0.0.1:9091 (U: **admin** P: **password**)
- Check S3 API access:

```
Unset
aws --endpoint-url http://127.0.0.1:9090 s3 ls
2023-05-03 00:59:18 perconalive
```

- Check there is no employee named **Douglas**

```
Unset
mysql> SELECT * FROM employees.employees WHERE first_name =
'Douglas'\G
Empty set (0.09 sec)
```

- Start to simulate load:

```
Unset
run_load.sh run
```

- Take a backup streaming to S3:

```
Unset
xtrabackup --backup --page-tracking --stream=xbstream \
--extra-lsndir=/backups/stream_full | xbcloud put --storage=s3 \
--s3-endpoint=http://127.0.0.1:9090 --s3-access-key=admin \
--s3-secret-key=password --s3-bucket=perconalive full_backup
```

- Douglas was hired. Add her to the database

```
Unset
mysql> INSERT INTO employees.employees VALUES (500003, '1988-05-21',
'Douglas', 'Reynholm', 'M', '2023-05-22');
```

- Take an incremental backup streaming to S3:

```
Unset
xtrabackup --backup --stream=xbstream \
--extra-lsndir=/backups/stream_inc \
--incremental-basedir=/backups/stream_full | xbcloud put \
--storage=s3 --s3-endpoint=http://127.0.0.1:9090 \
--s3-access-key=admin --s3-secret-key=password \
--s3-bucket=perconalive inc_backup
```

- Stop load script
- Stop MySQL and remove datadir:

```
Unset
mysql -e shutdown
rm -rf /var/lib/mysql
```

- List folder on **perconalive** bucket:

```
Unset
aws --endpoint-url http://127.0.0.1:9090 s3 ls s3://perconalive
                        PRE full_backup/
                        PRE inc_backup/
```

- Create folder, download and extract full backup

```
Unset
mkdir /backups/stream_full
xbcloud get --storage=s3 --s3-endpoint=http://127.0.0.1:9090 \
--s3-access-key=admin --s3-secret-key=password \
--s3-bucket=perconalive full_backup | xbstream -x -C \
/backups/stream_full
```

- Create folder, download and extract full backup

```
Unset
mkdir /backups/stream_inc
xbcloud get --storage=s3 --s3-endpoint=http://127.0.0.1:9090 \
--s3-access-key=admin --s3-secret-key=password \
--s3-bucket=perconalive inc_backup | xbstream -x -C \
/backups/stream_inc
```

- Prepare full backup

```
Unset
xtrabackup --prepare --apply-log-only \
--target-dir=/backups/stream_full
```

- Prepare incremental backup

```
Unset
xtrabackup --prepare --target-dir=/backups/stream_full \
--incremental-dir=/backups/stream_inc
```

- Copy backup to datadir and adjust folder ownership:

```
Unset
xtrabackup --copy-back --target-dir=/backups/stream_full
chown -R mysql.mysql /var/lib/mysql
```

- Start MySQL

```
Unset
mysqld --user=mysql &
```

- Check if Douglas is an employee:

```
Unset
mysql> SELECT * FROM employees.employees WHERE first_name =
'Douglas'\G
```

RENAME EXTRALSN DIR AS NAME IS COLIDING