

Лекция  
05.03.2019

# DBA1. p11.

Резервное копирование

Ильшат Каразбаев  
руководитель группы DBA  
АО ТК Центр

# Немного обо мне

Вместе со своей командой администрирую:

СУБД MySQL, Mariadb, galeracluster, Postgres

Главный по базам в ТК Центр

# Повестка дня:

1. Вводная
2. Виды резервного копирования
3. Инструменты
4. Best practices
5. Демонстрация
6. Задания
7. Литература

# Вводная.

Для чего нужны бекапы?

Восстановление после сбоев

Человеческий фактор (кто-то нечаянно все удалил)

Аудит и ревизия кода

Тестирование

# Виды резервного копирования

1. Логическое копирование
2. Физическое копирование

# Логические копии.

Генерируется простой файл: SQL, CSV, tab-separated

Можно использовать команды SQL для загрузки данных:

- INSERT INTO ...
- LOAD DATA INFILE ...

Можно восстановить даже одну строку.

Актуальность версии сервера уходит на второй план, иногда можно переносить данные между разными СУБД.

# Логические копии. Утилиты

## mysqldump

- Один из лучших инструментов логического копирования
- Можно дампить структуры бд, таблиц
- Можно восстановить даже одну строку
- Блокировки. Не такая уж и большая проблема с InnoDB.
- Может генерировать большие файлы SQL, дампить таблицы по отдельности, а не всю структуру БД с данными целиком.

# Логические копии. Утилиты

## mysqldump

- Подходит для создания реплик (--master-data)
- Долгое время снятия/восстановления (нужно парсить весь файл SQL, почти всегда - в один поток)
- Однопоточность (можно параллелить на таблицы, но необходимо пошаманить)
- Character set могут поехать, нужно быть начеку
- Даты могут поехать, если сервера в различных временных зонах



# Логические копии. Утилиты

mysqldump

Версии

с MariaDB 5.3: --start-transaction (start transaction with consistent snapshot)

с MariaDB 10.3 поддержка system versioned tables (несовместим со старыми версиями mysqldump)

# Логические копии. Утилиты

## SELECT INTO OUTFILE

- Доступен в виде опции mysqldump
- Необходимо меньше ресурсов при восстановлении
- Немного сложнее восстанавливать
- Можно генерировать вывод в CSV для переноса между разными СУБД

# Логические копии. Утилиты

mysdumper/myloader

- Похож на mysqldump, поддерживает параллельность
- Поддерживается в том числе и Перконой
- Поддержка GTID (в mysqldump тоже есть)
- Может дампит структуры таблиц и БД, триггеры, events, routines (в mysqldump тоже есть)
- В целом плох для mariadb, много багов и медленнее mysqldump

# Физические копии

- Точная копия БД со всеми сущностями
- Восстановление всего или ничего
- Быстрый способ создать/восстановить полную копию данных
- Ограничения только в скорости ресурсов (ЦП, диск, сеть)
- Отлично подходит для создания или восстановления всей инфраструктуры

# Физические копии. Утилиты

## xtrabackup

- Поддержка локального и удаленного резервного копирования
- Поддержка инкрементальных бекапов
- Необходима подготовка резервных копий
- Можно задействовать по максимуму ресурсы CPU, RAM
- Начиная с mariadb 10.3 лучше использовать форк - mariabackup (не поддерживает формат лог файлов)

# Физические копии. Утилиты

xtrabackup

- Поддержка частичных бекапов (отдельно сдампить таблицу или БД)
- Поддержка информации о репликации (в том числе и галера)

# Физические копии. Утилиты

## `mariabackup`

- Форк `xtrabackup`
- Для MariaDB 10.3+ необходимо заменить `xtrabackup` на `mariabackup`
- Поддерживает все фичи `xtrabackup`
- `innodb page compression`
- `data at rest encryption`
- полная поддержка фич MariaDB
- Поддержка ОС windows
- Можно использовать в качестве метода SST в галера
- Поддержка MyRocks storage engine

# Физические копии. Утилиты

Снепшоты

LVM, EBS, SAN

- Точная копия данных
- Быстро
- Не так гибко, как xtrabackup/mariabackup
- Плохо подходит для частичного восстановления данных



# Физические копии. Утилиты

Снепшоты

LVM, EBS, SAN

- Копии должны быть согласованными
- `innodb_flush_log_at_trx_commit=1`, необходим `innodb recovery`
- Лучше остановить экземпляр БД, но это не всегда допустимо

# Физические копии. Утилиты

Для amazon EC2

ec2-consistent-snapshot

- Копии остановленных экземпляров БД
- FLUSH TABLE WITH READ LOCK
- freeze файловой системы fsfreeze/xfs\_freeze
- RAID

# Point in time recovery

С помощью бинарных логов (должны быть включены)

- `mysqlbinlog binlog.00001 > data.sql`
- Для RBR: `mysqlbinlog --base-64--output=DECODE-ROWS --verbose binlog.00001 > data.sql`

# Point in time recovery. Hint.

Если на реплике или при переводе мастер сервера в режим “только чтение”

- STOP SLAVE/SET GLOBAL READ\_ONLY=1;
- FLUSH NO\_WRITE\_TO\_BINLOG BINARY LOGS;
- снимаем бекап любым способом
- START SLAVE/SET GLOBAL READ\_ONLY=0;
- начиная с этого момента у нас будет отсечка по бинлогам и будет проще ВОССТАНОВИТЬ

# Point in time recovery. Восстановление

1. Смотрим позицию бекапа: `cat xtrabackup_binlog_info`
2. `mysqlbinlog --start-position=xxx`
3. найти время, до которого нужно восстановиться, просмотрев бинарный лог
4. `mysqlbinlog --start-position=xxx --stop-position=yyy`
5. найти позицию бинлога, после которой нам снова нужны данные
6. `mysqlbinlog --start-position=zzz`
7. Если использовать hint с предыдущего слайда, то пункт 1 и 2 необязателен
8. Если нам не нужна конечная позиция (мы просто донакатим бинлоги как инкрементальный бекап), то кроме накатки бинлогов после отсечки ничего не нужно
9. Если нам нужно догнать до актуальных данных, то и отсечка не нужна, при наличии мастер сервера
10. Если умер мастер сервер, то тоже не беда (при использовании GTID), мы можем использовать другую реплику в качестве мастера (она должна быть 🤖)

# Best practices. Мониторинг и автоматизация

1. Мониторинг создания бекапов
2. Мониторинг размера бекапов
3. Мониторинг ошибок при создании бекапов
4. Автоматизация создания бекапов

# Best practices. Тестирование резервных копий.

1. Состояние бекапа неизвестно, пока он не восстановлен
2. Поэтому - регулярно восстанавливайте резервные копии
3. Проводить тестирование также и после изменения процедур резервного копирования
4. Тестирование бекапов - это полный тест от восстановления данных до разворачивания репликации. (распаковать, расшифровать, создание нового инстанса, старт репликации)

# Best practices. Восстановление после сбоев.

- Создайте план восстановления после сбоев
- Бекапы должны храниться отдельно от основного датацентра
- Необходимо постоянно измерять время полного восстановления (копирование копии, разворачивание, бутстрап реплики, актуализация данных)
- Автоматизируйте процесс восстановления. Это особенно важно для больших данных, когда на копирование и разворачивание уходят часы работы. Без БД сервис недоступен и бизнес теряет деньги.



# Best practices. Восстановление после сбоев.

- Полезно иметь горячие реплики в другом датацентре для исключения в ряде случаев использования бекапов и всех шагов по восстановлению
- Полезно иметь отстающую реплику
- Необходимо, чтобы работать с восстановлением реплик умели все те, кто будет работать во время сбоев, а не один человек, который когда-то настроил резервное копирование

# Best practices. Security

Если кто-то сможет проникнуть на мастер сервер, сможет ли он дотянуться и до бекап сервера?

# Best practices. Что бекапить

1. Код процедур, events, triggers
2. Конфигурацию репликации
3. Конфигурацию сервера
4. Данные БД
5. Конфигурацию системных пользователей (мониторинг, ротирование slow логов, чекеры статуса мариадб для кластерных решений)
6. Бинарные логи (отдельно от политики хранения бинарных логов `expire_logs_days`)

# Демонстрация

# Задания

1. Создайте реплику с помощью `mysqldump`
2. Создайте реплику с помощью `myiabackup`
3. Настройте автоматическое резервное копирование на реплике раз в день полное и раз в 4 часа инкрементальное с помощью `myiabackup`, автоматическое удаление копий старше месяца.
4. Восстановитесь из резервной копии, которая создастся в п.3, восстановите инкрементальный бекап поверх, а потом настройте репликацию.

# Вопросы?

Все анонсы здесь:

- telegram чат: [t.me/mariadb\\_course](https://t.me/mariadb_course)

Материалы курса:

- видео: [https://www.youtube.com/channel/UCGsmu6YDpcR\\_kWcXzeQkWrA](https://www.youtube.com/channel/UCGsmu6YDpcR_kWcXzeQkWrA)
- слайды лекций и примеры: [git@github.com:barazbay/mariadb\\_course.git](https://github.com:barazbay/mariadb_course.git)

Меня можно найти:

- vk, instagram: barazbay
- twitter: karazbay

# Литература

1. Почти вся презентация отсюда:  
<https://www.youtube.com/watch?v=zE5pNzO7Unk>
2. High performance mysql, Baron Schwartz
3. <https://mariadb.com/kb/en/library/mysqldump/>
4. <https://mariadb.com/kb/en/library/select-into-outfile/>
5. <https://mariadb.com/kb/en/library/mariabackup-overview/>
6. <https://mariadb.com/resources/blog/convert-galera-node-to-async-slave-and-vice-versa-with-galera-cluster/>