

MySQL | MariaDB

- Tworzenie usera na potrzeby backupu CLI
- Ustawienia archiwizacji logów transakcyjnych
- Purge logów starszych niż 3 dni
- Wykonanie backupu wszystkich baz, każda do osobnego pliku

Tworzenie usera na potrzeby backupu CLI

Wykonujemy zalogowani do CLI mysql/mariadb jako root

Tworzenie usera pod backup:

```
CREATE USER 'NazwaUzytkownika'@'localhost' IDENTIFIED BY 'TrudneHaslo';  
GRANT EVENT, LOCK TABLES, SELECT, RELOAD, SHOW VIEW ON *.* TO 'NazwaUzytkownika'@'localhost';  
FLUSH PRIVILEGES;
```

Oraz nadajemy uprawnienia do tworzenia backupu:

```
GRANT PROCESS ON *.* TO 'NazwaUzytkownika'@'localhost';  
GRANT LOCK TABLES ON *.* TO 'NazwaUzytkownika'@'localhost';  
GRANT SHOW VIEW ON *.* TO 'NazwaUzytkownika'@'localhost';  
FLUSH PRIVILEGES;
```

Ustawienia archiwizacji logów transakcyjnych

W MariaDB / MySQL logi transakcyjne nazywane są **binlogami** i z automatu nie odkładają się na dysk, dlatego by to włączyć musimy zedytować pliki my.cnf / my.ini

Lokalizacja dla MySQL

```
C:\ProgramData\MySQL\MySQL Server X.Y\my.ini
```

Lokalizacja dla MariaDB

```
C:\Program Files\MariaDB X.Y\data\my.ini
```

W pliku wyszukujemy sekcję:

```
# ***** Group Replication Related *****  
# Specifies the base name to use for binary log files. With binary logging  
# enabled, the server logs all statements that change data to the binary  
# log, which is used for backup and replication.
```

I usuwamy znak komentarza "#" bądź dopisujemy poniższe parametry:

```
binlog-format = ROW  
expire-logs-days = 2  
log-bin="LokalnaŚcieżkaDoFolderuLogów"
```

Purge logów starszych niż 3 dni

Wykonujemy z poziomu CLI SQL

```
PURGE BINARY LOGS BEFORE NOW() - INTERVAL 3 DAY;
```

Wykonanie backupu wszystkich baz, każda do osobnego pliku

Na potrzeby backupu baz danych będziemy potrzebować następującego skryptu:

Wersja dla systemów linux

Tworzymy plik:

```
vim NazwaSkryptu.sh
```

```
#!/bin/bash

USER="NazwaUseraBackupu"
PASSWORD="HasłoBackup"
#OUTPUT="Ścieżka zapisu plików"

#Ewentulne usuwanie starych backupów
#rm "$OUTPUTDIR/*gz" > /dev/null 2>&1

databases=`mysql -u $USER -p$PASSWORD -e "SHOW DATABASES;" | tr -d " | " | grep -v Database`

for db in $databases; do
    if [[ "$db" != "information_schema" ]] && [[ "$db" != "performance_schema" ]] && [[ "$db" != "mysql" ]] && [[ "$db" != "_" ]] ; then
        echo "Dumping database: $db"
        mysqldump -u $USER -p$PASSWORD --databases $db > `date +%Y%m%d`. $db.sql

        # Ewentualne packowanie plików SQL
        # gzip $OUTPUT/`date +%Y%m%d`. $db.sql
    fi
done
```

Należy pamiętać by nadać uprawnienia do uruchamiania (atrybut x) dla pliku:

```
chmod +x NazwaSkryptu.sh
```

Wykonujemy poprzez

```
./NazwaSkryptu.sh
```

Można też oczywiście dodać to do **crontab**