

TWORZENIE KOPII REZERWOWYCH BAZ ORAZ ICH NAPRAWA

- Serwer MySQL podczas uruchamiania posiada możliwość sprawdzania tabel w celu odszukania i naprawiania problemów.
- Niektóre programy (zewnętrzne) do naprawy lub robienia kopii zapasowej współpracują z serwerem. Łączą się z nim i wydają instrukcje SQL, które informują serwer o rodzaju operacji sprawdzania tabel lub tworzenia kopii zapasowej, jaką ma wykonać. Podczas, gdy taki program pracuje z tabelą, serwer nie powinien z niej jednocześnie korzystać. Jeżeli będzie miał taką możliwość, współzawodnictwo o dostęp do tabeli może ją uszkodzić i spowodować, że nie będzie nadawać się do użytku.

Blokowanie tabel

a) Wewnętrzne

- W celu sprawdzenia:
 - Po zalogowaniu:
 - **mysql>use baza;**
 - **mysql>LOCK TABLE tabela READ;**
 - **FLUSH TABLE tabela;**
To polecenie każe serwerowi zamknąć pliki tabel, co usuwa z bufora wszystkie niezapisane jeszcze zmiany w tabelach.
 - Wtedy można z drugiego okna wywołać program do sprawdzenia (patrz dalsza część instrukcji)
 - Po zakończeniu należy zwolnić blokadę poleceniem
mysql>UNLOCK TABLE;
- W celu naprawy:
 - Różnica w blokowaniu:
mysql>LOCK TABLE tabela WRITE;
mysql>FLUSH TABLE tablea;
 - Odblokowywanie:
mysql>UNLOCK TABLE;

Można od razu wszystkie tabele zablokować (blokada odczytu):

mysql>FLUSH TABLES WITH READ LOCK;

Wyłączenie blokady:

mysql>UNLOCK TABLES;

Ta metoda jest dobra na unieruchomienie serwera podczas kopiowania wszystkich katalogów baz danych. Jest ona jednak nieprzyjemna klientom, którzy muszą wykonywać aktualizację, dlatego taką blokadę serwera należy używać tylko, gdy jest konieczna.

b) Zewnętrzne

W celu sprawdzenia, czy serwer może korzystać z blokowania zewnętrznego trzeba sprawdzić zmienną serwera, która od MySQL v.4 nosi nazwę `skip_external_locking`, a wcześniej `skip_locking`.

mysql>SHOW VARIABLES LIKE 'skip%locking';

Jeżeli jest włączona (ON), to blokada zewnętrzna nie działa i serwer nie będzie wiedział, kiedy programy sprawdzają lub naprawiają tabelę. Trzeba wtedy zastosować blokowanie wewnętrzne, aby poinformować serwer, żeby nie korzystał w tej chwili z tabeli i nie pozwalał sięgać do niej klientom.

Tworzenie kopii zapasowych

1. mysqldump

Współdziała ściśle z serwerem MySQL. Należy się upewnić, że programy klienckie nie modyfikują kopiowanych w tym momencie tabel (program uruchamiamy z wiersza poleceń bez logowania do mysql).

mysqldump --opt -u root -p baza > nazwa_pliku

Jest to plik tekstowy, składa się z instrukcji min. INSERT i CREATE TABLE.

Opcje:

--opt - zabezpiecza przed modyfikowaniem kopiowanych tabel, blokując je wszystkie jednocześnie

--databases - pozwala na jednoczesne kopiowanie kilku baz

Np. --databases baza1 baza2

Jak się nie da tej opcji, to pierwszy wpis będzie traktowany jako baza, a drugi jako tabela, czyli np.

mysqldump -u root -p --opt baza1 tabela1

Skopiuje tylko tabelę1 z bazy

Jak chcemy parę tabel z bazy skopiować:

mysqldump -u root -p --opt baza tabela1 tabela2

Aby utworzyć kopię zapasową wszystkich baz danych serwera, należy użyć opcji --all --databases.

W tym przypadku nie podaje się żadnych argumentów w postaci nazwy bazy danych lub tabeli. Z tej opcji należy korzystać ostrożnie, jeżeli kopie rezerwowe mają być przeniesione do innego serwera. Będą one zawierać tabele uprawnień z bazy danych mysql, a niekoniecznie chodzi o wymianę także tabel uprawnień tego serwera.

Domyślnie program mysqldump kopiuje strukturę tabel (instrukcje CREATE TABLE) i ich zawartość (instrukcje INSERT). Użycie opcji --no-create-info spowoduje skopiowanie tylko zawartości tabel, zaś instrukcji --no-data tylko ich struktury.

2. mysqlhotcopy

szybszy od mysqldump,

automatycznie zarządza protokołem blokowania

mysqlhotcopy -u -p=hasło baza katalog_gdzie_ma_skopiowac

żeby w poleceniu nie podawać hasła, trzeba w /etc/mysql/my.cnf wpisać password=hasło i wtedy uruchamia się bez opcji „p”.

program ten kopiuje cały katalog z bazą.

Odzyskiwanie całej bazy danych

W celu odzyskania bazy danych należy ją ponownie załadować z ostatniej kopii zrobionej programem **mysqldump** lub **mysqlhotcopy**.

1. w przypadku **mysqldump** plik trzeba przesłać jako dane do programu mysql:

```
#mysql -u root -p --one-database baza<plik
```

--one-database pozwala programowi mysql wykonać tylko zapytania dla pojedynczej odtwarzanej bazy danych.

2. w przypadku **mysqlhotcopy** wystarczy katalog z kopią zapasową skopiować bezpośrednio do katalogu danych (/var/lib/mysql).

Sprawdzanie i naprawa tabel

1. sprawdzanie

```
#myisamchk /var/lib/mysql/baza/tabela
```

Domyślna opcja to --check

Rozszerzony test:

```
--extend-check
```

Jeżeli test z tą opcją nie wykazał błędów można być pewnym, że z tabelą jest w porządku.

2. naprawa

```
#myisamchk --recover --quick ...
```

--quick powoduje próbę nagrania na podstawie tylko zawartości pliku indeksu.

Jeżeli problem nie zostanie rozwiązany:

```
#myisamchk --recover ...
```

Wtedy program może też zmodyfikować pliki danych.

Jeśli to nie zadziała:

```
#myisamchk --safe-recover ...
```

Jest to tryb wolniejszy, ale usuwa problemy, z którymi tryb --recover sobie nie poradzi.

UWAGA! Jeżeli, w którymś momencie program przerwie procedurę naprawy wyświetlając komunikat:

„can't create New temp file:nazwa pliku”

należy powtórzyć polecenie dodając opcję --force, która zmusi go do usunięcia tymczasowego pliku. Plik ten może być pozostałością poprzedniej nieudanej próby naprawienia tabeli.

Odzyskiwanie kontroli nad serwerem, z którym nie można się połączyć.

Jak się zapomni hasła na konto root:

Wyłączyć serwer
/etc/init.d/mysql stop

Uruchomić serwer z opcją --skip-grant-tables w następujący sposób:

mysqld_safe --skip-grant-tables &

Opcja ta informuje serwer, aby podczas weryfikacji połączenia nie korzystał z tabel uprawnień. Pozwoli to uzyskać połączenie bez podawania hasła z wszystkimi uprawnieniami.

Potem w innym oknie zalogować się bez hasła

#mysql -u root

Potem trzeba wydać instrukcję

mysql>FLUSH PRIVILEGES;

każe serwerowi ponownie odczytać tabele uprawnień, aby znów ich używać (bo inaczej serwer jest zupełnie otwarty)

potem można ustawić nowe hasło.

UWAGA!!! - do wykonanych ćwiczeń wykonaj sprawozdanie. W sprawozdaniu mają się znajdować polecenia, które użyłeś do wykonania danego ćwiczenia. Ponadto zanim cokolwiek usuniesz, zmodyfikujesz itp. zwołaj prowadzącego zajęcia.

Zadanie 1

Sprawdzić, czy działa blokada zewnętrzna tabel.

Zadanie 2

Utworzyć bazę danych o nazwie **zjazd**.

Zadanie 3

Utworzyć konto **kierownik** o hasle **123** i nadać mu pełne uprawnienia do bazy **zjazd**.

Zadanie 4

W drugim terminalu zalogować się jako kierownik i w bazie zjazd utworzyć tabelę **uczestnicy** o polach:

Id (autonumerownie, klucz podstawowy)

imie text(20)

nazwisko text(40)

Dodać do tabeli uczestnicy 3 rekordy.

Zadanie 5

Jako **root** zablokować bazę danych **zjazd**;

Zadanie 6

Jako **kierownik** dodać do bazy **zjazd** kolejny rekord. Co się dzieje?

Zadanie 7

Odblokować jako **root** dostęp do bazy **zjazd** i zaobserwować co się dzieje na koncie **kierownik**.

Zadanie 8

Utworzyć kopię zapasową bazy **zjazd** za pomocą polecenia **mysqldump** do pliku **zjazd_kopia**.

Zadanie 9

Jako **kierownik** usunąć z bazy **zjazd** uczestnika o **id=3**.

Zadanie 10

Jako **root**, korzystając z pliku **zjazd_kopia**, przywrócić bazę **zjazd**.

Zadanie 11

Jako **kierownik** sprawdzić, czy usunięty rekord został przywrócony.

Zadanie 12

Utworzyć kopię bazy **zjazd** za pomocą **mysqldhotcopy** do katalogu **kopia_zjazd** (utworzonym wcześniej w katalogu domowym).

Zadanie 13

Jako **kierownik** usunąć z bazy **zjazd** uczestnika o **id=3**.

Zadanie 14

Odzyskaj bazę **zjazd** danych skopiowaną wcześniej poleceniem **mysqldhotcopy**.

Zadanie 15

Jako **kierownik** sprawdzić, czy usunięty rekord został przywrócony.

Zadanie 16

Sprawdź tabelę **uczestnicy** (korzystając z polecenia **mysqlcheck**).

Zadanie 17

Opis sytuacji: Zapomniałeś hasła na konto **root**, zaloguj się na serwer **mysql** bez znajomości hasła i następnie nadaj kontu **root** nowe hasło.