
mysqlbinlog

Exécuter des requêtes dans les logs binaire

OPTIONS

- database=db_name, -d db_name** Limite les lignes à cette base de données (log local uniquement).
- force-read, -f** Continue même si vous obtenez une erreur SQL.
- host=host_name, -h host_name** Lit le log binaire depuis le serveur MySQL distant.
- local-load=path, -l path** Prépare les fichiers temporaires destinés aux commandes **LOAD DATA INFILE** dans le dossier spécifié.
- offset=N, -o N** Ignore les N première lignes.
- password [=password], -p [password]** Le mot de passe utilisé lors de la connexion sur le serveur. S'il n'est pas donné en ligne de commande, il sera demandé interactivement. Notez que si vous utilisez la forme courte **-p**, vous ne devez pas laisser d'espace entre l'option et le mot de passe.
- port=port_num, -P port_num** Le numéro de port TCP/IP pour la connexion.
- position=N, -j N** Commence la lecture dans le log binaire à la position N.
- protocol=TCP | SOCKET | PIPE | MEMORY** Spécifie le protocole de connexion à utiliser.
- read-from-remote-server, -R** Lit le log binaire depuis un serveur MySQL. Les options de connexion distantes seront ignorées à moins que cette option ne soit donné. Ces options sont **-host, -password, -port, -protocol, -socket** et **-user**.
- result-file=name, -r name** Export direct vers le fichier spécifié.
- short-form, -s** Affiche uniquement les commandes du log, sans les informations supplémentaires.
- socket=path, -S path** Le fichier de socket à utiliser pour la connexion.
- user=user_name, -u user_name** Le nom d'utilisateur MySQL lors de la connexion à distance.
- version, -V** Affiche la version et quitte.

Exemples

Vous pouvez envoyer le résultat de mysqlbinlog vers un client mysql avec un pipe : c'est une technique pour restaurer le serveur après un crash

```
shell> mysqlbinlog hostname-bin.000001 | mysql
```

ou :

```
shell> mysqlbinlog hostname-bin.[0-9]* | mysql
```

Si vous avez plus d'un fichier de log binaire à exécuter sur le serveur MySQL, la méthode sûre est de tout faire avec la même connexion MySQL.

```
shell> mysqlbinlog hostname-bin.000001 hostname-bin.000002 | mysql
```

La seconde méthode :

```
shell> mysqlbinlog hostname-bin.000001 > /tmp/statements.sql
```

```
shell> mysqlbinlog hostname-bin.000002 » /tmp/statements.sql
```

```
shell> mysql -e "source /tmp/statements.sql"
```