

## **Objectif de ce tutoriel**

Ce tutoriel va vous expliquer comment paramétrer le serveur MySQL de votre diskstation Synology DS211 afin de résoudre le problème de limitation sur la taille des requêtes SQL.

## **Description du problème**

Voici une exception Java qui surprend toujours lorsqu'on la rencontre pour la première fois :

*com.mysql.jdbc.PacketTooBigException: Packet for query is too large (9729274 > 8388608). You can change this value on the server by setting the max\_allowed\_packet' variable.*

Vous rencontrerez ce problème lorsque vous commencerez à vouloir insérer dans vos tables des données volumineuses comme des fichiers (type SQL : BLOB pour Binary Large Object). De quoi s'agit-il exactement ? La taille des requêtes SQL est limitée par défaut à 8,38 Mo mais cette limitation peut être changée en allant modifier la variable « max\_allowed\_packet » du serveur MySQL.

## **Comment résoudre le problème ?**

J'ai dû tâtonner quelques temps avant de trouver une solution. Je tiens à préciser qu'elle n'est certainement pas la solution la plus « officielle » mais elle a le mérite de marcher et j'ai pensé qu'elle pourrait intéresser d'autres néophytes qui comme moi n'ont pas eu de solide formation en administration serveur.

Voici à présent un descriptif rapide des différentes étapes qui constituent ma solution :

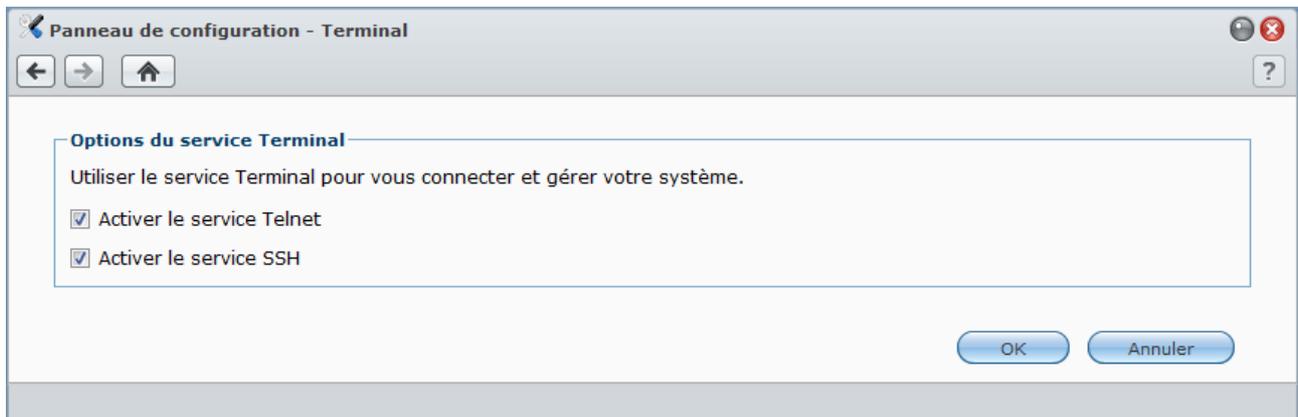
1. Activer le service SSH sous le DSM du diskstation Synology
2. Installer putty.exe sur votre Windows
3. Se connecter en « root » avec SSH sur le diskstation
4. Retrouver le fichier /usr/syno/etc/rc.d/S21mysql.sh
5. Éditer le fichier avec vi et modifier la valeur de max\_allowed\_packet
6. Redémarrer MySQL avec la commande : /usr/syno/etc/rc.d/S21mysql.sh restart

Les plus expérimentés n'auront sans doute pas besoin de plus de détails pour comprendre ma démarche. Pour les autres, voici ci-après la description de chaque étape.

## Étape 1 : Activer le service SSH

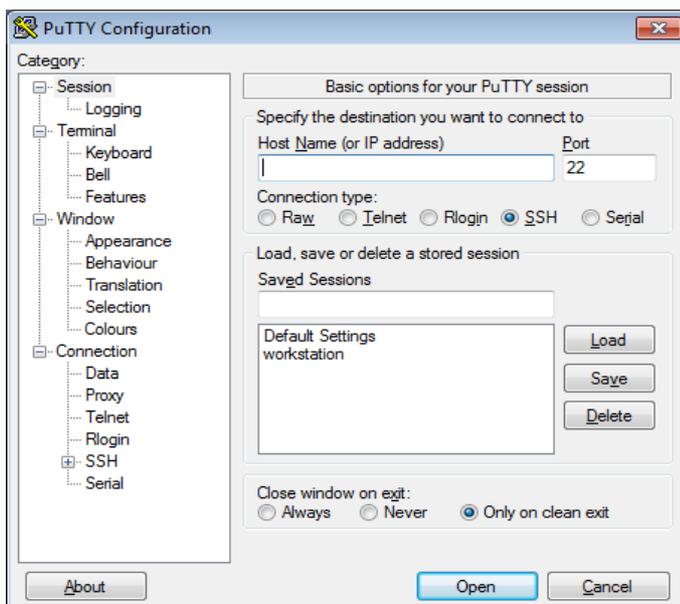
Vous devez vous connecter sur votre diskstation avec l'interface Web d'administration Diskstation Manager (url : <http://<ip-de-votre-serveur>:5000/webman/index.cgi>)

Ensuite, ouvrez : Panneau de configuration / Terminal, et vérifiez que l'option « Activer le service SSH » est bien cochée. Le port 22 sera désormais ouvert sur votre serveur pour vous permettre d'y accéder par SSH (Secure Shell) qui est un équivalent sécurisé de telnet.



## Étape 2 : Installer putty.exe

Ensuite, il va vous falloir un client SSH pour vous permettre de vous connecter à votre serveur. Comme Windows n'en possède pas par défaut, il m'a fallu aller en chercher un sur Internet, et j'ai trouvé putty (tapez « putty ssh » sous google). Bien sûr libre à vous d'en utiliser un autre si vous voulez.



PuTTY permet de faire du SSH, du telnet et a priori plein d'autres choses. En ce qui nous concerne, nous allons ouvrir une session ssh sur le host de notre diskstation Synology avec le port 22.

### Étape 3 : se connecter en root

Après avoir cliqué sur le bouton Open, la console noire apparaît, c'est le début des réjouissances.

```
login as: █
```

Quel login/password choisir ?

Instinctivement, j'ai tout d'abord pris le login « admin » et mon mot de passe administrateur Synology, puis je me suis rendu compte que je n'avais pas tous les droits nécessaires pour faire ce que je voulais, et j'ai finalement pris le login « root » avec le même mot de passe que mon compte admin.

```
BusyBox v1.16.1 (2011-10-24 18:58:05 CST) built-in shell (ash)
Enter 'help' for a list of built-in commands.

DiskStation> █
```

Cette fois, nous y sommes. J'espère que vous avez quelques notions en commande en ligne DOS ou UNIX.

### Étape 4 : Retrouver le fichier `/usr/syno/etc/rc.d/S21mysql.sh`

Nous devons retrouver le fichier `S21mysql.sh` qui se trouve dans le répertoire suivant :  
`/usr/syno/etc/rc.d/`

Pour vous rendre dans ce répertoire, tapez la commande :

```
cd /usr/syno/etc/rc.d/
```

Ensuite, vous pouvez taper la commande `dir` pour lister son contenu et ainsi vérifier que le fichier recherché est bien là.

```
-FWXR-XR-X 1 root root 1231 Oct 24 13:40 S08snmpd.sh
-FWXR-XR-X 1 root root 548 Oct 24 13:39 S09DDNS.sh
-FWXR-XR-X 1 root root 539 Oct 24 13:39 S10UPnPportmap.sh
-FWXR-XR-X 1 root root 2299 Oct 24 13:39 S10slapd.sh
-FWXR-XR-X 1 root root 1240 Oct 24 13:39 S11nslcd.sh
-FWXR-XR-X 1 root root 964 Oct 24 13:40 S12upsmon.sh
-FWXR-XR-X 1 root root 3955 Oct 24 13:39 S20pgsql.sh
-FWXR-XR-X 1 root root 8172 Oct 24 13:39 S21mysql.sh
-FWXR-XR-X 1 root root 380 Oct 24 13:39 S23ntpdate.sh
-FWXR-XR-X 1 root root 3710 Oct 24 13:39 S23synologd.sh
-FWXR-XR-X 1 root root 4329 Oct 24 13:40 S25download.sh
-FWXR-XR-X 1 root root 5535 Oct 24 13:39 S55cupsd.sh
-FWXR-XR-X 1 root root 1421 Oct 24 13:39 S56gcpd.sh
-FWXR-XR-X 1 root root 1801 Oct 24 13:39 S66S2S.sh
-FWXR-XR-X 1 root root 2568 Oct 24 13:39 S66fileindexd.sh
```

## Etape 5 : Editer le fichier avec vi et modifier la valeur de max\_allowed\_packet

Vi qu'est ce que c'est ? C'est un éditeur de texte sous Linux qui fonctionne par ligne de commande. J'assumerai que comme moi, vous n'êtes pas familier avec cet outil, et que vous avez donc besoin d'être guidé pas à pas.

Voici un petit tutorial sur le vi :

<http://www.commentcamarche.net/contents/tutlinux/linvi.php3>

Je suis d'ailleurs actuellement en train de le consulter pour me rafraichir la mémoire avant d'écrire la suite...

Tout d'abord, lancez la commande : **vi S21mysql.sh**

```
#!/bin/sh
PID_FILE=/tmp/mysqld.pid
MYSQL_VERSION="5.1.49"
MYSQL_DIR="/usr/syno/mysql"
MYSQL_SERVICE_PATH="/var/services/mysql"
DO_NETBKP_CANCEL_TMP_FILE="/tmp/do_netbkp_cancel.chk"
LOCALBKP_CANCEL_TMP_FILE="/tmp/do_localbkp_cancel.chk"
MYSQL_RCVR_DB_CONFLICT_LIST="/tmp/mysql_rcvr_db_conflict_list.tmp"
MYSQL_RCVR_DB_NEW_LIST="/tmp/mysql_rcvr_db_new_list.tmp"
MYSQL_FAIL=1
CANCEL_RET=2
MYSQL_PROG_PATH=/usr/syno/mysql/bin/mysql

MYSQL_ARGS="--max_allowed_packet=8M"

UpgradeDatabase ()
{
    # Upgrade MySQL database
    Ret=0

    # Backup root password and reset password to empty
    echo "USE mysql;" > /tmp/mysql_init.$$
    echo "DELETE FROM user WHERE user='upgrade_root_tmp';" >> /tmp/mysql_ini

- S21mysql.sh 1/326 0%
```

La valeur que vous allez devoir modifier est indiquée en bleu. Vous pouvez faire passer la limite de 8M à celle que vous souhaitez. Pour ma part, j'ai pris l'habitude de la fixer à 128M.

Pour éditer cette valeur, placez d'abord le curseur au niveau du 8 de 8M

```
CANCEL_RET=2
MYSQL_PROG_PATH=/usr/syno/mysql/bin/mysql

MYSQL_ARGS="--max_allowed_packet=8M"

UpgradeDatabase ()
{
```

Puis appuyez sur la touche Inser de votre clavier pour passer en mode d'insertion. Editez votre valeur et quand vous avez fini, appuyez sur la touche Echap

```
MYSQL_PROG_PATH=/usr/syno/mysql/bin/mysql

MYSQL_ARGS="--max_allowed_packet=128M"

UpgradeDatabase ()
```

Enfin, tapez la commande suivante pour sauvegarder les modifications et quitter l'éditeur :

```
:wq
```

Si vous voulez quitter sans sauvegarder, la commande est la suivante :

```
:q
```

## Etape 6 : Redémarrer MySQL avec `/usr/syno/etc/rc.d/S21mysql.sh restart`

Tapez la commande : `/usr/syno/etc/rc.d/S21mysql.sh restart`

Le serveur MySQL s'arrête puis redémarre en prenant en compte la bonne valeur de `max_allowed_packet`.

```
DiskStation> /usr/syno/etc/rc.d/S21mysql.sh restart
Stopping MySQL...
Shutting down MySQL.. SUCCESS!
Starting MySQL...
Starting MySQL. SUCCESS!
MYSQL data dir = /volume1/@database/mysql...
DiskStation>
```

Voilà, le tour et joué ! Quittez putty avec la commande `exit`. Vous n'avez plus qu'à vérifier que le nouveau paramétrage est bien opérationnel en lançant votre requête SQL qui posait problème, et qui devrait normalement passer comme sur des roulettes.

## Pour aller plus loin

[phpMyAdmin](#) :

Utiliser l'outil `phpmyadmin` pour vérifier la valeur de la variable « `max_allowed_packet` » avant et après la manipulation décrite ci-dessus. Voici ce que l'on obtient après avoir fait sa manipulation :



Variable	Valeur pour la session / Valeur globale
auto increment increment	1
auto increment offset	1
autocommit	ON
automatic sp privileges	ON
back log	50
....	....
lower case table names	0
max allowed packet	134 217 728
max binlog cache size	4 294 963 200
max binlog size	1 073 741 824

Modification non pérenne :

Je tiens à signaler un défaut de taille concernant cette méthode : Il faut penser à la refaire après chaque mise à jour du serveur Synology.

Quelques infos en plus :

Les fichiers de donnée de vos bases de données sont stockés dans le répertoire suivant :  
/volume1/@database/mysql

Le répertoire d'installation de MySQL :  
/usr/syno/mysql