**MySQL over SSH**

Ghislain RABETRANO

**MySQL over SSH**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Auteur** | **Date** | **Note** |
| [1.0](file:///C:\Users\ghislain\OneDrive\switzernet\training\170912-mysql-over-ssh\index.docx) | Ghislain RABETRANO | 2017-07-12 | Création du document |
|  |  |  |  |

Table des matières

[À propos 1](#_Toc493059673)

[2. Training session 1](#_Toc493059674)

[Logiciels 1](#_Toc493059675)

[Préparation 1](#_Toc493059676)

[3. Connexion SSL 1](#_Toc493059677)

[4. MySQL over SSH 1](#_Toc493059678)

[5. Utilisation d’un client lourd (software) 1](#_Toc493059679)

[6. Mise en pratique 1](#_Toc493059680)

[7. Validation 1](#_Toc493059681)

# 

# À propos

**Durée estimée: 1 heure**

En général, une base de données est un sur un serveur distant. De ce fait, le seul moyen de pouvoir lui adresser des requêtes est de suivre les étapes suivantes :

Connexion SSH au serveur

Connexion au serveur MySQL

Enfin, on peut faire des requêtes SQL

Cela est très bien si on connaît parfaitement la structure de la base de données et si la requête est simple (pas trop de lignes de retour, pas trop de colonnes à afficher et la requête tient sur 2 ou 3 lignes).

Maintenant, on est vite limité si on veut rapidement faire des requêtes complexes et temporaires …

Dans ce training, nous allons voir :

Comment se connecter à une base de données distante au travers de SSH.

Comment exploiter cette connexion en utilisant un client lourd (software) pour MySQL.

# Training session

## Logiciels

* Cygwin [[lien](https://www.cygwin.com/)]
* MySQL Workbench [[lien](https://www.mysql.com/fr/products/workbench/)]

## Préparation

* Avoir une connexion sur le serveur www-dev.switzernet.com (Utilisateur : trainee)
* Avoir une connexion à la base de données « training » sur le serveur www-dev.switzernet.com (Même utilisateur : trainee)

# Connexion SSL

Une fois que vous avez récupéré le mot de passe de l’utilisateur « trainee », nous allons commencer par donner au serveur Linux notre clé publique pour ne pas à avoir à entrer à chaque fois un login/password.

Si ce n’est pas encore fait, nous devons créer une clé SSH publique.

Comme dans l’exemple si dessous, saisissez « ssh-keygen » et validez. Gardez les valeurs par défaut validant directement en appuyant la touche « entrée »

|  |
| --- |
| # ssh-keygen  Generating public/private rsa key pair.  Enter file in which to save the key (/home/trainee/.ssh/id\_rsa):  Enter passphrase (empty for no passphrase):  Enter same passphrase again:  Your identification has been saved in /home/trainee/.ssh/id\_rsa.  Your public key has been saved in /home/trainee/.ssh/id\_rsa.pub.  The key fingerprint is:  bf:7f:6e:14:25:0f:d0:bc:04:2d:27:58:d7:1a:d3:d0 trainee@www-dev  The key's randomart image is:  +---[RSA 2048]----+  | o+\*o= |  | . o.@ E|  | = X |  | + .|  | A . |  -------------------------+ |

Une fois la clé générée, nous allons déclarer notre clé publique au serveur Linux. Cela nous permettra de ne plus à avoir entré un login/mot de passe pour les futures connexions SSH.

|  |
| --- |
| # ssh-copy-id -i ~/.ssh/id\_rsa.pub trainee@www-dev.switzernet.com |

# MySQL over SSH

Maintenant, nous allons exploiter notre connexion SSH « sans mot de passe ».

Le protocole SSH permet d’exécuter une commande à distance.

Par exemple, nous pouvons lister le contenu d’un répertoire sur le serveur : au lieu de faire un « ls -ail ~ » sur le serveur, on peut le faire depuis sa machine locale.

|  |
| --- |
| # ssh trainee@www-dev.switzernet.com "ls -ail ~"  total 32  10621137 drwxr-xr-x 3 trainee trainee 4096 Sep 12 11:33 .  10616833 drwxr-xr-x 6 root root 4096 Sep 12 11:30 ..  10621143 -rw------- 1 trainee trainee 101 Sep 12 11:50 .bash\_history  10621138 -rw-r--r-- 1 trainee trainee 220 Sep 12 11:30 .bash\_logout  10621139 -rw-r--r-- 1 trainee trainee 3515 Sep 12 11:30 .bashrc  10621144 -rw------- 1 trainee trainee 122 Sep 12 11:33 .mysql\_history  10621140 -rw-r--r-- 1 trainee trainee 675 Sep 12 11:30 .profile  10621141 drwx------ 2 trainee trainee 4096 Sep 12 11:40 .ssh |

Le fait d’avoir déclaré sa clé publique sur le serveur permet aussi de faire de la redirection de port. SSH permet de spécifier qu’un port sur sa machine correspond au port du serveur distant.

Voyons l’exemple suivant :

ssh -L 3307:127.0.0.1:3306 user@monserveurdistant.com -N

Dans cet exemple, nous voyons que grace au à la connexion SSH « user@monserveurdistant.com », le port 3307 est équivalent à écouter depuis la machine distant la socket «127.0.0.1:3306 ».

D’un point de vue concret, dans notre cas, nous pouvons saisir la commande suivante :

|  |
| --- |
| # ssh -L 3307:127.0.0.1:3306 trainee@www-dev.switzernet.com -N & |

Nous pouvons donc tester que la connexion est bien établie :

|  |
| --- |
| # mysql -P3307 -utrainee -h127.0.0.1 -p  Enter password:  Welcome to the MySQL monitor. Commands end with ; or \g.  Your MySQL connection id is 62  Server version: 5.5.57-0+deb8u1 (Debian)  Copyright (c) 2000, 2017, Oracle and/or its affiliates. All rights reserved.  Oracle is a registered trademark of Oracle Corporation and/or its  affiliates. Other names may be trademarks of their respective  owners.  Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  mysql> |

Quelques précisions :

* -P3307 : Nous disons à notre client mysql local que le port est 3307 (comme spécifié pour notre tunnel)
* -h127.0.0.1 : Nous disons à notre client mysql que le serveur est sur 127.0.0.1 par rapport au serveur distant !

# Utilisation d’un client lourd (software)

Nous avons ici l’exemple avec le logiciel MySQL Workbench.

|  |  |
| --- | --- |
|  | Page d’accueil.  C’est à partir d’ici que nous ferons notre première connexion. |
|  | Une fois les données saisies, il faut tester la connexion |
|  | Nous voyons que la connexion est bien établie |
|  | Pour créer une table, cela devient très simple |
|  | Cela est également simple d’ajouter des données |
|  | Maintenant, nous pouvons faire des requêtes dans un environnement plus agréable à la vue. |

# Mise en pratique

À l’aide des exemples ci-dessus, établis une connexion vers la base de données MySQL de training.

Supprime toutes tables (drop table …)

Créer les tables « group » et « user » qui sont reliées par group.id = user.group\_id

Faites une requête avec une jointure

Pour aller plus loin :

Faites une extraction des données vers un fichier CSV

# Validation

Créez un fichier « validation.docx » et mettez une impression-écran de chaque étape

\* \* \*