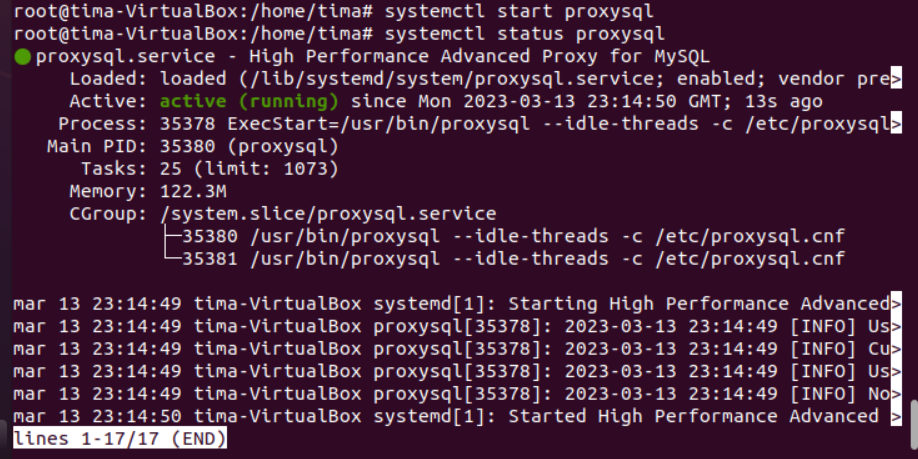
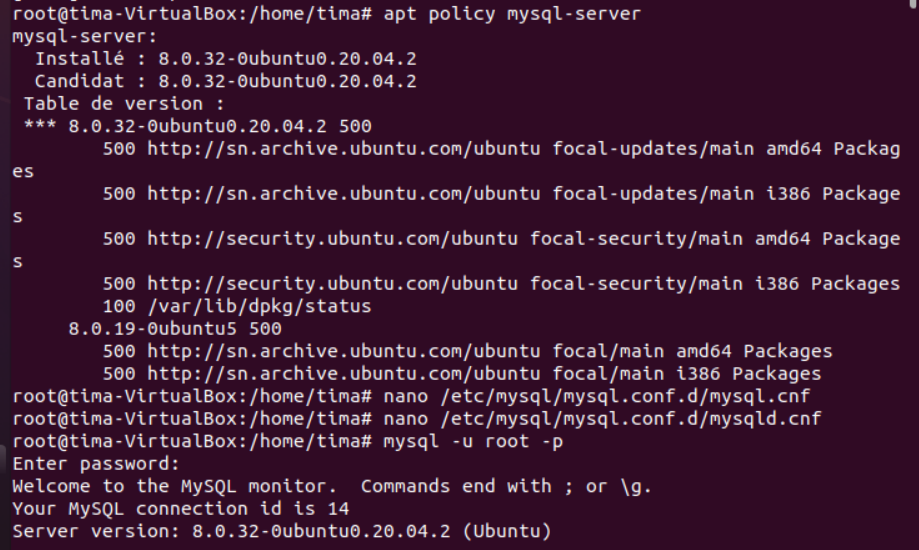
**Étape 1 - Installation de ProxySQL**



Ensuite, nous aurons besoin d'une application cliente MySQL pour se connecter à l'instance ProxySQL. En effet, ProxySQL utilise en interne une interface compatible avec MySQL pour les tâches administratives. Nous utiliserons l' mysqloutil de ligne de commande, qui fait partie du mysql-clientpackage disponible dans les référentiels Ubuntu.

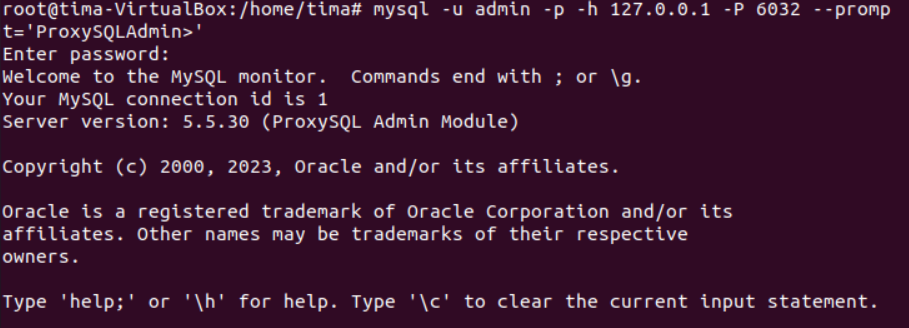


## Étape 2 - Définition du mot de passe de l'administrateur ProxySQL

Après son installation, ProxySQL utilise un fichier de configuration fourni par le package pour initialiser les valeurs par défaut de toutes ses variables de configuration. Après cette initialisation, ProxySQL stocke sa configuration dans une base de données qu’on peut gérer et modifier via la ligne de commande.

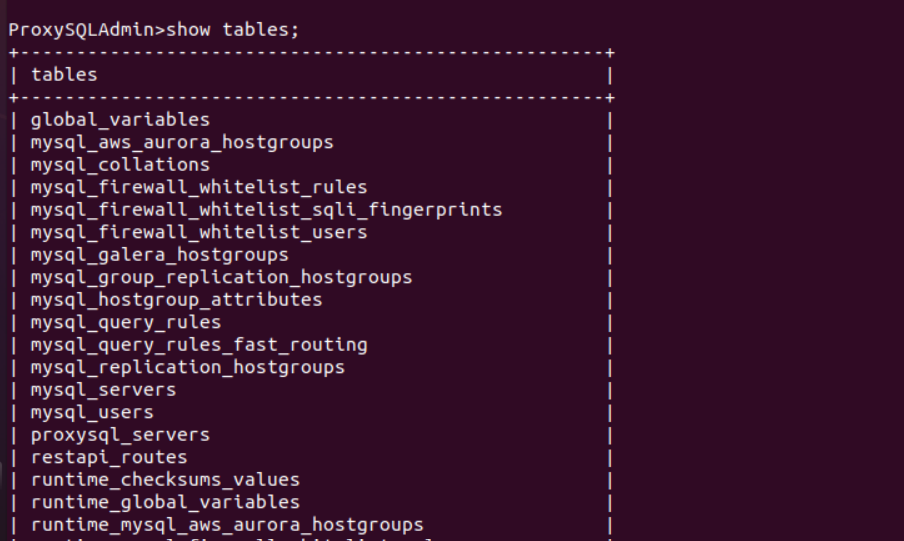
Pour définir le mot de passe administrateur dans ProxySQL, nous allons nous connecter à cette base de données de configuration et mettre à jour les variables appropriées.

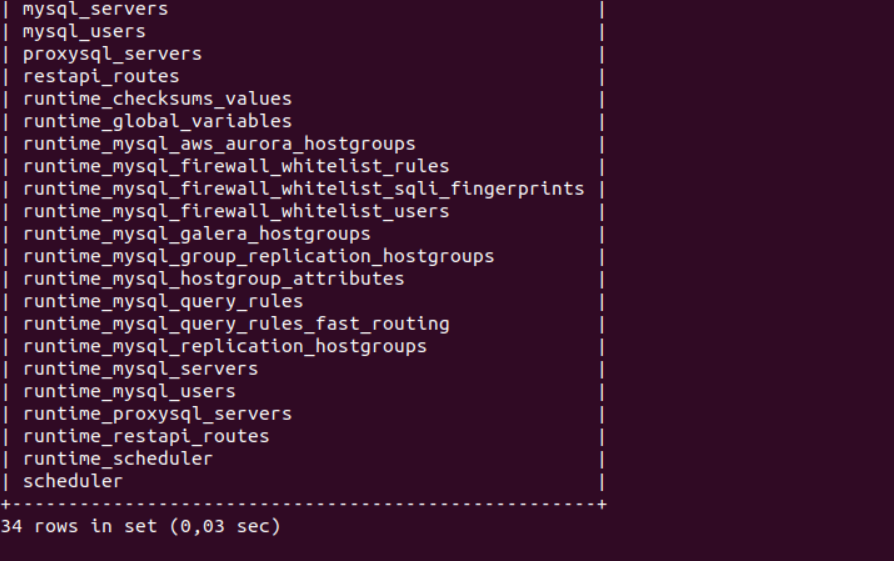
Tout d'abord, pour accéder à l'interface d'administration. Nous serons invités à entrer le mot de passe qui, sur une installation par défaut, est admin.



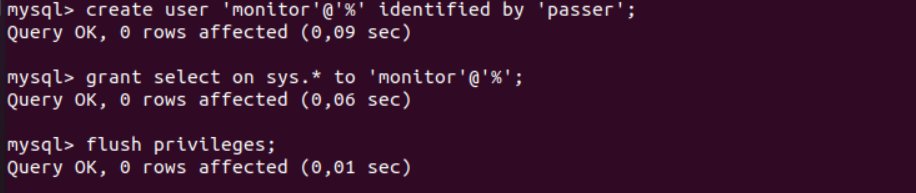
* -u spécifie l'utilisateur avec lequel nous voulons nous connecter, ici **admin** , l'utilisateur par défaut pour les tâches administratives telles que la modification des paramètres de configuration.
* -h 127.0.0.1indique mysqlde se connecter à l'instance ProxySQL locale. Nous devons définir cela explicitement car ProxySQL n'écoute pas le fichier socket qui mysql suppose par défaut.
* -P spécifie le port auquel se connecter. L'interface d'administration de ProxySQL écoute sur 6032.
* --prompt est un indicateur facultatif qui modifie l'invite par défaut, qui est normalement mysql>. Ici, nous le remplaçons par ProxySQLAdmin> pour indiquer clairement que nous sommes connectés à l'interface d'administration de ProxySQL. Cela sera utile pour éviter toute confusion plus tard lorsque nous nous connecterons également aux interfaces MySQL sur les serveurs de bases de données répliquées…

Lorsqu’on se connecte sur l'interface d'administration de ProxySQL, nous pouvons voir une nouvelle table : mysql\_group\_replication\_hostgroups





C'est le tableau que nous utiliserons pour configurer à quel groupe d'hôtes un nœud appartiendra.



## Étape 5 - Ajout de nœuds MySQL au pool de serveurs ProxySQL

Pour que ProxySQL connaisse nos trois nœuds MySQL, nous devons indiquer à ProxySQL comment les répartir sur ses *groupes d'hôtes* , qui sont des ensembles de nœuds désignés. Chaque groupe d'hôtes est identifié par un nombre positif, comme 1ou 2. Les groupes d'hôtes peuvent acheminer différentes requêtes SQL vers différents ensembles d'hôtes lors de l'utilisation du routage de requête ProxySQL.

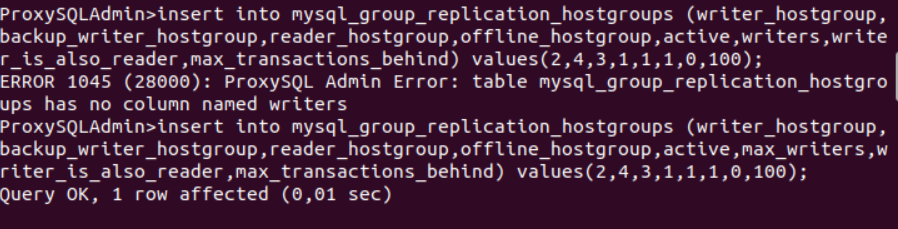
Dans les configurations de réplication statique, les groupes d'hôtes peuvent être définis arbitrairement. Cependant, la prise en charge de la réplication de groupe de ProxySQL divise automatiquement tous les nœuds d'un groupe de réplication en quatre états logiques :

* **writer** s, qui sont des nœuds MySQL pouvant accepter des requêtes qui modifient les données. ProxySQL s'assure de maintenir tous les nœuds principaux jusqu'à la quantité maximale définie dans ce groupe.
* **les rédacteurs de sauvegarde** , qui sont également des nœuds MySQL qui peuvent accepter des requêtes qui modifient les données. Cependant, ces nœuds ne sont pas désignés comme écrivains ; les nœuds principaux dépassant le nombre défini d'enregistreurs maintenus sont conservés dans ce groupe et sont promus en enregistreurs si l'un des enregistreurs échoue.
* **Les lecteurs** sont des nœuds MySQL qui ne peuvent pas accepter les requêtes modifiant les données et doivent être utilisés comme nœuds en lecture seule. ProxySQL ne place ici que des nœuds esclaves.
* **offline** , qui concerne les nœuds qui se comportent mal en raison de problèmes tels qu'un manque de connectivité ou un trafic lent.

Chacun de ces quatre états a des groupes d'hôtes correspondants, mais les identificateurs de groupe numériques ne sont pas attribués automatiquement.

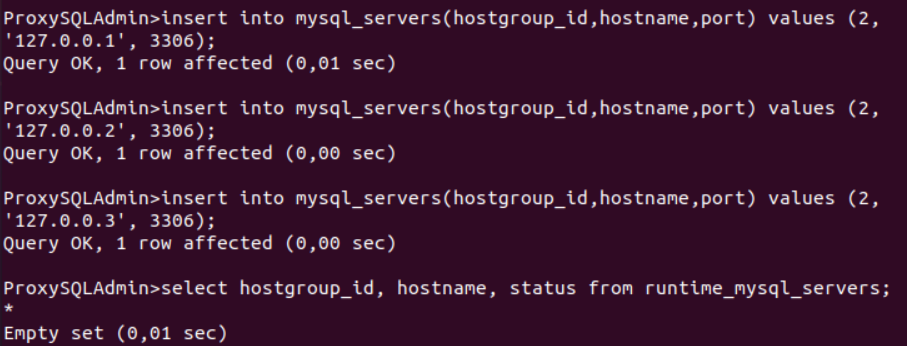
En mettant tout cela ensemble, nous devons dire à ProxySQL quels identifiants il doit utiliser pour chaque état. Ici, nous utilisons 1pour le groupe d'hôtes **hors ligne**2 , pour le groupe d'hôtes d' **écriture**3 , pour le groupe d'hôtes **de lecture** et 4pour le groupe d'hôtes d' **écriture de sauvegarde .**

Pour définir ces identifiants, créons une nouvelle ligne avec ces variables et valeurs dans la mysql\_group\_replication\_hostgroupstable de configuration.

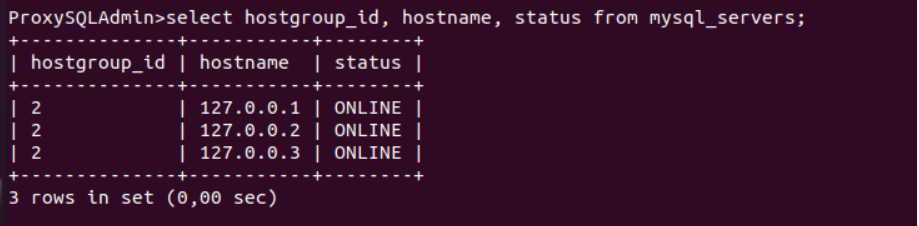


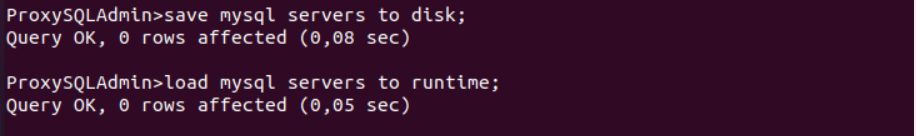
Maintenant que ProxySQL sait comment répartir les nœuds entre les groupes d'hôtes, nous pouvons ajouter nos serveurs MySQL au pool. Pour ce faire, nous avons besoin de INSERTl'adresse IP et du groupe d'hôtes initial de chaque serveur dans la mysql\_servers table, qui contient la liste des serveurs avec lesquels ProxySQL peut interagir.

Ajoutons chacun des trois serveurs MySQL, en veillant à remplacer les exemples d'adresses IP dans les commandes ci-dessous.



ProxySQL devrait maintenant distribuer nos nœuds sur les groupes d'hôtes comme spécifié. Vérifions cela en exécutant une SELECTrequête sur la mysql\_servers table, qui expose l'état actuel des serveurs que ProxySQL utilise.



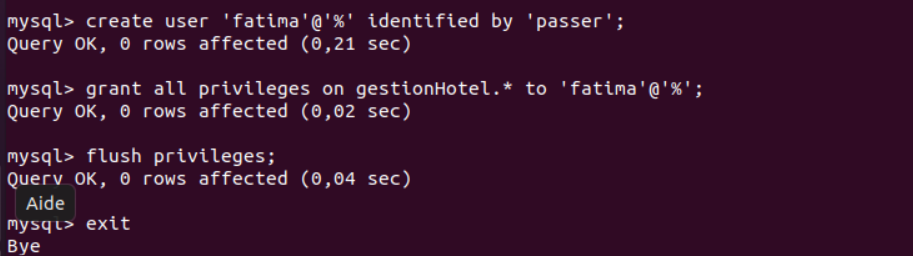


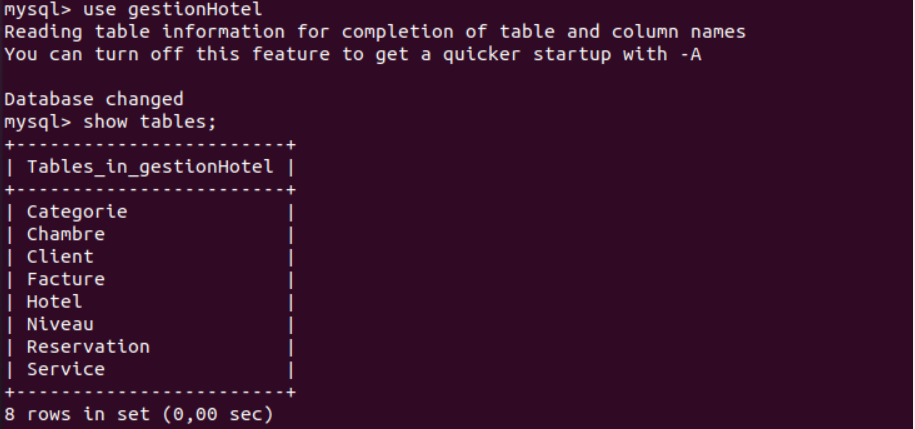
## Étape 6 - Création des utilisateurs MySQL

ProxySQL agit comme un équilibreur de charge ; les utilisateurs finaux se connectent à ProxySQL, et ProxySQL transmet à son tour la connexion au nœud MySQL choisi. Pour se connecter à un nœud individuel, ProxySQL réutilise les informations d'identification avec lesquelles il a été accédé.

Pour autoriser l'accès aux bases de données situées sur les nœuds de réplication, nous devons créer un compte utilisateur avec les mêmes informations d'identification que ProxySQL et accorder à cet utilisateur les privilèges nécessaires.

Créons un nouvel utilisateur appelé **fatima** identifié avec le mot de passe passer.



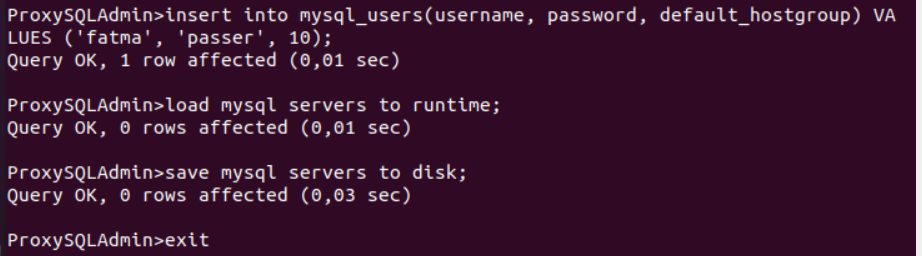


La liste visible des tables dans la base de données montrant la gestionHotel table créée dans le didacticiel de réplication d'origine confirme que l'utilisateur a été créé correctement sur les nœuds.

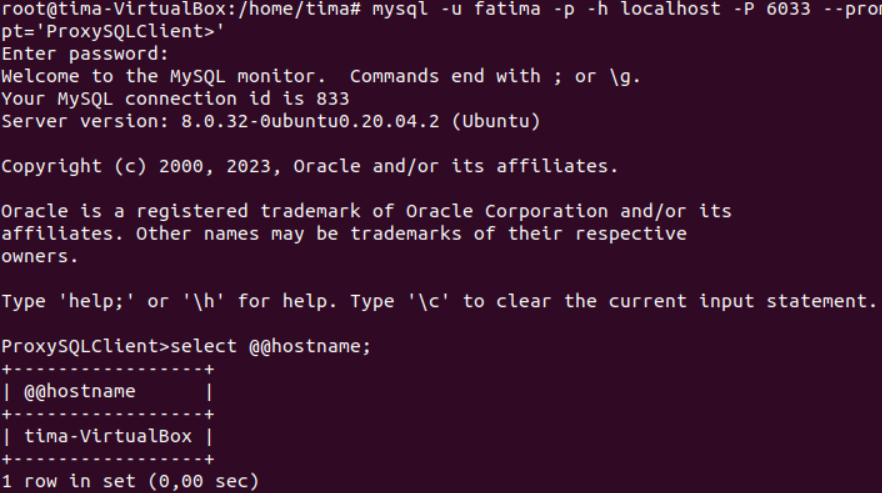
## Étape 7 - Création de l'utilisateur ProxySQL

La dernière étape de configuration consiste à autoriser les connexions à ProxySQL avec l' utilisateur **fatima** et à transmettre ces connexions aux nœuds.

Pour ce faire, nous devons définir des variables de configuration dans la mysql\_userstable, qui contient les informations d'identification de l'utilisateur. Dans l'interface ProxySQL, ajoutons le nom d'utilisateur, le mot de passe et le groupe d'hôtes par défaut à la base de données de configuration (qui est 2, pour le groupe d'hôtes **de l'écrivain** )



ProxySQL écoute sur le port 6033 les connexions client entrantes, essayons donc de nous connecter à la vraie base de données (pas à l'interface d'administration) en utilisant **fatima** et port 6033.



## Étape 8 - Vérification de la configuration ProxySQL