

# AWS RDS to Azure DB for MySQL

## 1 前言

本文主要通过主从复制机制，将 AWS 的 RDS 数据库数据复制到 Azure DB for MySQL，实现 PaaS to PaaS 的数据复制。

## 2 阅读对象

技术部门全体成员

文档类型说明：

**Solution** 对某项具体需求做出的解决方案

**KB** 解决某个具体问题的文档/手册

**FAQ** 使用自问自答的形式对常见问题的解答

本次我们将用 2 个 Azure DB for MySQL 作为测试环境  
前提条件：

## 要求

- 源服务器版本必须至少是 MySQL 5.6 版。
- 源服务器版本和副本服务器版本必须相同。例如，两者必须同时是 MySQL 5.6 版或 MySQL 5.7 版。
- 每个表都必须有主键。
- 源服务器应使用 MySQL InnoDB 引擎。
- 用户必须有权在源服务器上配置二进制日志记录和创建新用户。
- 如果源服务器启用了 SSL，请确保为域提供的 SSL CA 证书已包含在 `mysql.az_replication_change_master` 或 `mysql.az_replication_change_master_with_gtid` 存储过程中。请参阅以下[示例](#)和 `master_ssl_ca` 参数。
- 请确保源服务器的 IP 地址已添加到 Azure Database for MySQL 副本服务器的防火墙规则中。使用 [Azure 门户](#) 或 [Azure CLI](#) 更新防火墙规则。
- 请确保托管源服务器的计算机在端口 3306 上允许入站和出站流量。
- 请确保源服务器具有公共 IP 地址，DNS 可公开访问，或者源服务器具有完全限定的域名 (FQDN)。

### 📌 重要

必须在“常规用途”或“内存优化”定价层中创建 Azure Database for MySQL 服务器，因为数据传入复制只在这些层中受支持。

## 创建并配置源 MySQL

主页 > Azure Database for MySQL 服务器 > 选择 Azure Database for MySQL 部署选项 >

### 创建 MySQL 服务器

Microsoft

⚠️ 更改“基本”选项可能会重置你所做的选择。创建资源前，请先审阅所有选项。

**项目详细信息**  
选择订阅以管理已部署资源和成本。使用资源组(如文件夹)以组织和管理所有资源。

订阅 \*

资源组 \*  [新建](#)

**服务器详细信息**  
输入此服务器的必需设置，包括选取位置以及配置计算和存储资源。

服务器名称 \*  ✓

数据源 \*

位置 \*

版本 \*

创建完后可看到

工具 / Azure Database for MySQL 服务器 ...

Poc4CN (poc4halal.en.microsoft.com)

+ 创建 | 管理视图 | 刷新 | 导出至 CSV | 打开查询 | 分配标记 | 反馈

筛选任何字段... 订阅 == 全部 | 资源组 == 全部 | 位置 == 全部 | 添加过滤器

正在显示 1 到 2 条记录(共 2 条)。 不分组 | 列表视图

名称 ↑	类型 ↑	状态 ↑	资源组 ↑	位置 ↑	订阅 ↑
<input type="checkbox"/> seanmysql1	Azure Database for MySQL 服务器	可用	MySql	东亚	Microsoft Azure

## 创建 MySQL 服务器 ...

Microsoft

### 服务器详细信息

输入此服务器的必需设置，包括选取位置以及配置计算和存储资源。

服务器名称 \*

数据源 \*

位置 \*

版本 \*

计算 + 存储   
2 个 vCore, 100 GB 存储空间  
[配置服务器](#)

管理员帐户

管理员用户名 \*

密码 \*

确认密码 \*   密码和确认密码必须匹配。

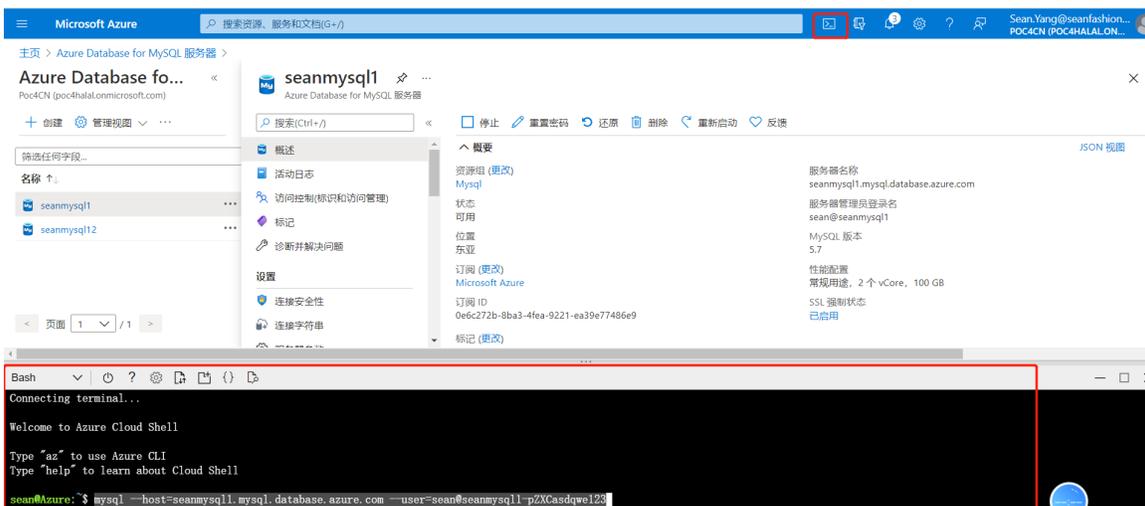
- 确保源服务器允许端口 3306 上的入站和出站流量，并且源服务器具有公共 IP 地址，DNS 可供公开访问，或者 DNS 具有完全限定的域名 (FQDN)。
- 通过尝试从工具（例如其他计算机上托管的 MySQL 命令行）或 Azure 门户中提供的 Azure Cloud Shell 连接，测试与源服务器的连接。
- 如果你的组织有严格的安全策略，并且不允许源服务器上的所有 IP 地址都能进行从 Azure 到源服务器的通信，那么你可能可以使用以下命令来确定 MySQL 服务器的 IP 地址。
- 使用 MySQL 命令行之类的工具登录 Azure Database for MySQL 服务器。



测试阶段设置为允许所有连接，正常环境请还是相关对应的 ip 地址



## 通过 azure CLI 连接到数据库



mysql --host=xxxx.mysql.database.azure.com --user=sean@seanmysql1-pZXCasdqwe123

```

Your MySQL connection id is 65174
Server version: 5.6.47.0 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>

```

## 创建一个数据库及表作为后续测试数据

```

CREATE DATABASE replication default charset=utf8;
create table tb_student(id int not null auto_increment,name varchar(20),
age tinyint unsigned default 0,
gender enum('male','female'),
subject enum('ui','java','yunwei','python'),
primary key(id)) engine=innodb default charset=utf8;

```

```

insert into tb_student values (null,'刘备',33,'male','java');
insert into tb_student values (null,'关羽',34,'male','python');
insert into tb_student values (null,'张飞',35,'male','yunwei');
insert into tb_student values (null,'曹操',30,'male','java');

```

```

MySQL [replication]> insert into tb_student values (null,'曹操',30,'male','java');
Query OK, 1 row affected (0.042 sec)

```

```

MySQL [replication]>
MySQL [replication]>
MySQL [replication]>
MySQL [replication]>
MySQL [replication]> select * from tb_student;
+----+-----+-----+-----+-----+
| id | name  | age  | gender | subject |
+----+-----+-----+-----+-----+
| 1  | 刘备  | 33   | male   | java    |
| 2  | 关羽  | 34   | male   | python  |
| 3  | 张飞  | 35   | male   | yunwei  |
| 4  | 曹操  | 30   | male   | java    |
+----+-----+-----+-----+-----+
4 rows in set (0.035 sec)

```

## 启用二进制日志记录

```
SHOW VARIABLES LIKE 'log_bin';
```

```

MySQL [replication]> SHOW VARIABLES LIKE 'log_bin';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| log_bin       | ON    |
+-----+-----+
1 row in set (0.037 sec)

```

如果没有启用，请去/etc/my.cnf 添加一行  
log-bin=mysql-bin.log  
然后重启 mysql 服务

## 配置源服务器设置

```
SET GLOBAL lower_case_table_names = 1;
```



## 创建新的复制角色并设置权限

```
CREATE USER 'syncuser'@'%' IDENTIFIED BY 'yourpassword';  
GRANT REPLICATION SLAVE ON *.* TO 'syncuser'@'%';
```

## 导出源数据库及表

## 将源服务器设置为只读模式

```
FLUSH TABLES WITH READ LOCK;  
SET GLOBAL read_only = ON; (这一步我们需要在 portal 去设置)
```



## 导出数据库及表

```
mysqldump --opt -h seanmysql1.mysql.database.azure.com -u sean@seanmysql -pZXCasdqwe123 replication > replication.sql
```

```
sean@Azure: ~$ mysqldump -h seanmysql1.mysql.database.azure.com -u sean@seanmysql -pZXCasdqwe123 replication > replication.sql
```

连接到 slave 数据库（如果连不上，请根据前面操作将防火墙权限给最大，禁用 SSL）

```
sean@Azure: ~$ mysql --host=seanmysql12.mysql.database.azure.com -u sean@seanmysql12 -pZXCasdqwe123
```

```
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MySQL connection id is 65138
Server version: 5.6.47.0 MySQL Community Server (GPL)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>
```

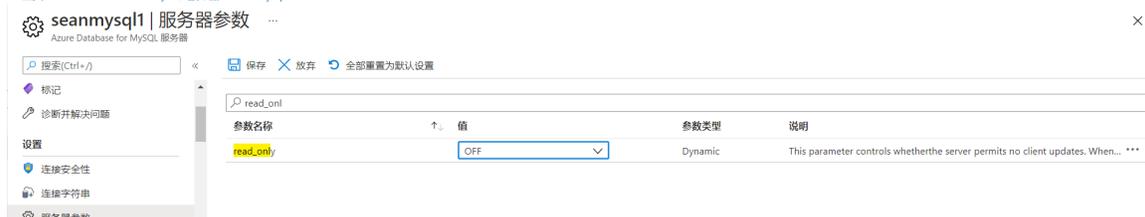
创建一个 database，可以同名，也可以是另外名称的  
CREATE DATABASE replication default charset=utf8;

## 将源库导出的 sql 文件导入到 slave 库里

```
mysql -h seanmysql12.mysql.database.azure.com -u sean@seanmysql12 -pZXCasdqwe123 replication2 < replication.sql
```

```
sean@Azure: ~$ mysql -h seanmysql12.mysql.database.azure.com -u sean@seanmysql12 -pZXCasdqwe123 replication2 < replication.sql
```

转储数据库后，将 MySQL 源服务器改回读/写模式。



```
UNLOCK TABLES;
```

```
Database changed
MySQL [(replication)]>
MySQL [(replication)]> UNLOCK TABLES;
Query OK, 0 rows affected (0.035 sec)

MySQL [(replication)]>
```

## 配置服务器启用复制

在源 DB 里执行

```
SHOW MASTER STATUS;
```

```
MySQL [replication]> show master status;
```

File	Position	Binlog_Do_DB	Binlog_Ignore_DB	Executed_Gtid_Set
mysql-bin.000001	3854			

1 row in set (0.035 sec)

记录下 (File 值和 Position 值)

在 slave 数据库中执行一下命令 (里面涉及的用户是刚刚在源服务器中创建的 syncuser 用来执行复制的)

```
CALL mysql.az_replication_change_master('seanmysql1.mysql.database.azure.com', 'syncuser@seanmysql1', 'yourpassword', 3306, 'mysql-bin.000001', 3854);
```

```
CALL mysql.az_replication_start;
```

```
show slave status\G;
```

```
      Last_IO_Error:
      Last_SQL_Errno: 0
      Last_SQL_Error:
  Replicate_Ignore_Server_Ids:
    Master_Server_Id: 2767040204
           Master_UUID: f866c341-dd67-11eb-a0ae-217e16f0757a
    Master_Info_File: mysql.slave_master_info
           SQL_Delay: 0
  SQL_Remaining_Delay: NULL
  Slave_SQL_Running_State: Slave has read all relay log; waiting for more updates
    Master_Retry_Count: 86400
    Master_Bind:
  Last_IO_Error_Timestamp:
```

去源服务器插入一条信息

```
insert into tb_student values (null,'貂蝉',24,'female','ui');
```

然后观察 slave 服务器，发现该数据已同步过来

```
MySQL [replication]> select * from tb_student;
```

id	name	age	gender	subject
1	刘备	33	male	java
2	关羽	34	male	python
3	张飞	35	male	yunwei
4	曹操	30	male	java
5	貂蝉	24	female	ui

```
5 rows in set (0.035 sec)
```