

设置数据库

历史

在本单元中，我们将使用免费数据库 MariaDB，它是 MySQL 的克隆版本，由同一作者 - 来自芬兰的 Michael“Monty”Widenius 编写。这个命名背后的历史是 Monty 首先创建了 MySQL（以他的女儿 My 命名），该数据库后来成为最流行的开源数据库。他将 MySQL 卖给了 Sun，Sun 又被以商业数据库软件而闻名的 Oracle 收购。结果，Monty 创建了一个名为 MariaDB 的 MySQL 克隆（以他的另一个女儿的名字命名）。MySQL 和 MariaDB 在很多方面都是兼容的，我们在本单元中学到的大部分内容同样适用于它们，尽管 Oracle（对于 MySQL）和开源社区（对于 MariaDB）都为其数据库添加了新功能。

尽管我们将使用 MariaDB，但在某些地方，名称 MySQL 仍然会出现 - 最明显的是作为用于访问数据库的命令程序的名称。

安装数据库

打开您的虚拟机并使用适当的命令安装数据库。在 Debian 上是：

```
$ sudo apt install mariadb-server mariadb-client
```

Advanced note

那个时髦的波浪形括号在做什么？Bash 将其扩展为 `mariadb-server mariadb-client`。同样，如果您输入：`cc -o hello{,.c}` 它将扩展为 `cc -o hello hello.c`。有很多速记技巧可以让您在 shell 中更快地输入内容：尝试边走边拿起它们！

问：打印什么 `echo {a,b,c}-{1,2,3}`？在运行之前尝试猜测。

该 `mariadb-server` 软件包包含存储数据并让客户端登录的服务器。`mariadb-client` 是一个命令行客户端（命令名称为 `mysql`）；稍后我们还将使用 Java 客户端从 Java 应用程序访问数据库。

数据库服务器现已安装，但除非您要求，否则我们的系统不会启动它。在 Debian 上，服务管理器是 `SystemD`。我们可以使用以下命令启动服务器运行：

```
$ sudo systemctl start mariadb
```

检查服务是否正在运行

```
$ sudo systemctl status mariadb
$ sudo journalctl -u mariadb
```

将其设置为默认运行：

```
$ sudo systemctl enable mariadb
```

Advanced note

虽然 Linux或多或少地标准化了使用 SystemD 作为服务管理器.....但它在某些方面并不受欢迎。还存在其他系统！*Alpine Linux*（这是我们以前使用的VM 映像）使用 OpenRC。BSD 主要使用大杂烩的 shell 脚本来完成此任务。Mac 使用一种称为 `launchctl`。

反对 SystemD 的论点是它破坏了与 POSIX OS 标准的兼容性并且违背了 *UNIX Way*（做好一件事）；开发人员 Lennart Poettering 有点争议（软技能很重要！）；坦率地说，该项目的过度扩张令人难以置信。除了管理服务之外，它还可以加密您的主文件夹、管理 WiFi 连接、管理您的日志文件和系统名称等等。

支持它的理由是它速度快，摆脱了一堆糟糕的 shell 脚本，并以对 Linux 有意义的方式标准化了事物。Linux 发行版过去更加多样化，但现在的选择主要是您想要使用什么包管理器以及默认情况下您想要什么桌面。

目前，SystemD 似乎是我们为 Linux 选择的。一旦你学会了它就基本上没问题了，但是一定要尝试一下 BSD 系统，看看它以前是什么样子，如果你喜欢它的话！

安全

要登录数据库，您通常需要一个用户帐户和一个身份验证因素（例如密码或私钥）。但是，在最新版本中，mysql 用户帐户链接到系统用户帐户。不过你可能应该保护它。在没有安全性的情况下运行面向公众的数据库将导致数据泄露和罚款，速度比您输入的速度还要快 *metasploit*。

默认设置将允许任何人登录并查看某些数据库，例如表，`test` 但这并不是特别安全。大多数发行版都附带一个 `mysql_secure_installation` 脚本，可以设置更安全的访问权限。运行它并为 root 用户设置密码（否则它将是默认的 root 密码或空白）。

创建数据库

是的，你有一个正在运行的 mysql 服务器！让我们将它连接到数据库！创建数据库：

```
mysqladmin -u root -p create mydatabase
```

要连接到它：

```
mysql -u root -p mydatabase
```

这应该会让你进入提示！恭喜！您有一个正在运行的数据库。

导入样本数据

我们准备了一些示例数据，将在本周和接下来的几周内使用。

首先，下载以下两个文件并将它们放在与 Vagrantfile 相同的文件夹中。该文件夹很重要，因为脚本包含从导入某些数据 `sample-data.sql` 开始的硬编码绝对路径。 `/vagrant/` 您可以像之前使用安全安装文件一样下载文件。

```
cd /vagrant
wget https://raw.githubusercontent.com/cs-
uob/COMSM0085/master/code/databases/sample-data.sql
wget https://raw.githubusercontent.com/cs-
uob/COMSM0085/master/code/databases/sampledatab.tar
tar -xvf sampledatab.tar
```

这将创建一个文件夹 `sampledata`，其中包含我们需要的文件。

如果您使用的是此存储库的本地副本，您还可以在 `/code/databases`。

该 `tar` 文件是磁带存档：包含更多文件和文件夹的文件，就好像它本身就是一个文件夹一样。

Advanced note

这里的选项是 `x=`提取文件，`v=`验证（将每个处理的文件的名称打印到标准输出），`f=`文件名位于以下参数中。有时您可能会看到此命令没有这些选项的“-”——这是有效的，因为它 `tar` 支持较旧的约定，其中选项不以破折号为前缀，但为了安全起见，您应该坚持现代约定（它也理解）。

要自己创建 `tar` 文件，命令将是 `tar -cvf ARCHIVE.tar FILE1 FILE2...` `c=`create the archive（如果存档不存在），并假设其他标志未使用的所有参数都引用要添加的文件。事实上，`tar -xvf ARCHIVE.tar FILES...` 它也可以工作，并且只从存档中提取指定的文件。

使用以下命令加载示例数据：

```
mysql -u root -p -e 'source /vagrant/sample-data.sql'
```

这会提取 CSV 文件中的一些数据（如果需要，请查看脚本）并创建一个默认用户“`vagrant`”，该用户可以在没有密码的情况下登录数据库，但只能读取而不能写入两个示例数据库“人口普查”和“选举”。还有另一个名为“`data`”的数据库，它一开始是空的，“`vagrant`”既可以读取也可以写入它。

Advanced note

它似乎还抛出了一堆错误，并且是以前运行该单元的人留给我们的。它在我们需要修复的工作清单上，但它似乎仍然有效？哟洛。拉取请求表示赞赏。

您现在可以登录数据库并尝试以下操作：

- `mysql` 当您登录时，您现在会收到数据库提示，`MariaDB [(none)]>` 表明您尚未加载特定数据库。
- `SHOW DATABASES;` 数据库提示符上会显示您也可以访问的数据库列表。
- 使用 `USE` 命令选择一个，例如 `USE elections;`。您的提示现在应该显示为 `MariaDB [elections]>`。
- `SHOW TABLES;` 将显示所选数据库中的表。
- 选举数据库中有一个政党表，因此 `SELECT * FROM Party;` 将向您显示数据。
- 您还可以用来 `DESCRIBE Party;` 显示 `Party` 表中的列和类型的列表。
- 最后，`exit` 或单独一行上的 `Control+D` 可让您返回 `shell`。

您可以打开下面的 SQL 和 CSV 文件，`/vagrant/sampleddata` 与从 MariaDB 获得的输出进行比较。研究这个直到它对你有意义为止。这些 `setup.sql` 文件包含数据库架构，文件 `import.sql` 提取示例数据。

在实验室机器上

在实验室计算机上，为了节省磁盘空间，您的虚拟机在重新启动之间可能不会保留 - 并且因为它们不是托管在网络文件系统上，如果您下次登录到不同的计算机，您的更改也不会被保存，但您会从头开始重新安装虚拟机。为了确保数据库在您需要时随时准备就绪，请 `Vagrantfile` 在文本编辑器中打开并将设置命令添加到配置脚本中。这些命令与我们刚才手动执行的命令相同。