How to install Firebird on Ubuntu 18.04?

By **angeloma** - October 28, 2019

Database management systems are sufficient for many kinds of projects. Of course, they abound with MySQL / MariaDB or PostgreSQL popular but there are also other very good and efficient ones like SQLite or Firebird. The latter is less known but equally efficient and sophisticated. Of course, its operation is more similar to SQLite than to the very robust MySQL / MariaDB. So in this post, I will show you how to install Firebird on Ubuntu 18.04.

Firebird is a relational database management system written in C++, open–source and based on Interbase version 6. It uses SQL language which makes its learning curve quite low. It is less known than the highly popular MySQL / MariaDB but also used in large projects for its efficiency.

On the other hand, some of the features of the application are the following:

- It is scalable
- Support the Client/Server architecture by TCP protocol.
- Many drivers are available for various programming languages such as PHP, Java or C++.
- Quite secure in user management.
- Cross-platform. With installable binaries for Windows, Linux, BSD, Solaris, and others.

So I will teach you how to install it and we will take the opportunity to do some small tests with the application.

Install Firebird on Ubuntu 18.04

The version that is available in the official Ubuntu 18.04 repositories include Firebird. But the included version is quite old and does not take advantage of the new features of the recent versions as the correction of quite critical bugs.

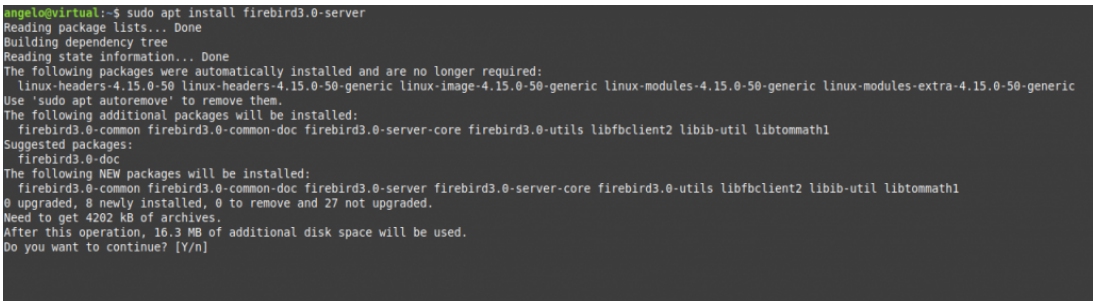
Fortunately, we have a dedicated repository for it, so let us get to work.

Open a terminal session or connect to your server using SSH and add the repository.

```
:~$ sudo add-apt-repository ppa:mapopa/firebird3.0
```

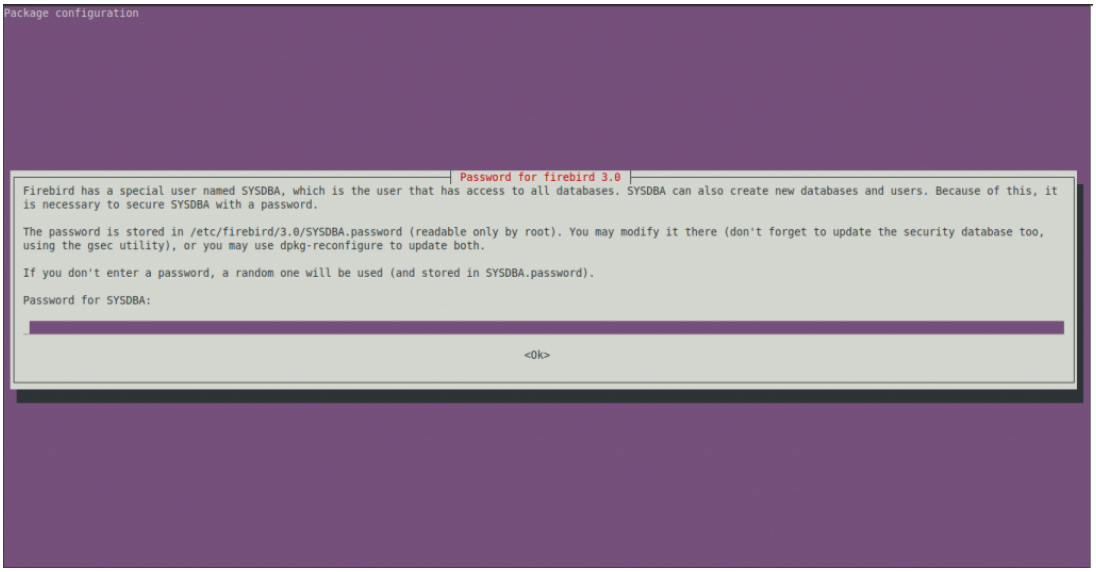
After it is added. Refresh the APT cache and install the necessary packages:

```
:~$ sudo apt update
:~$ sudo apt install firebird3.0-server
```



1.- Install Firebird on Ubuntu 18.04

During installation, you will be informed that the default user is “SYSDBA”. Define a password for this main user. It has to be quite secure.



2.- Set a password for the default user

Once you have defined the password. All you have to do is reconfigure the package so that all the changes are properly applied.

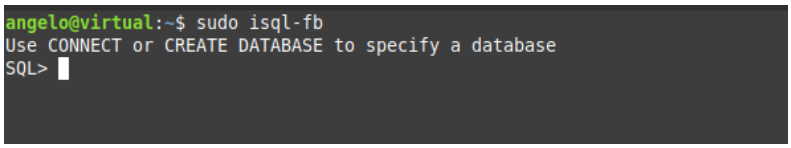
```
:~$ sudo dpkg-reconfigure firebird3.0-server
```

And it is ready to use.

Working with Firebird on Ubuntu 18.04

To enter the Firebird administration console and start using commands, just execute:

```
:~$ sudo isql-fb
```



3.- Using Firebird

With this we will be able to execute commands. And to get out just use this other one:

```
> quit;
```

Before starting to create databases and tables, it is necessary to talk about one thing. Firebird can store and access your databases from anywhere SQLite style, but can also restrict the location of them.

To make it accessible from anywhere or from a specific directory, just edit its configuration file:

```
:~$ sudo nano /etc/firebird/3.0/firebird.conf
```

Leave it as it is in the image to allow access to the database from any directory. There you can set a restriction to a particular directory.

```
# ----------------------------
# Database Paths/Directories
#
# DatabaseAccess may be None, Full or Restrict. If you choose Restrict,
# provide ';'-separated trees list, where database files are stored.
# Relative paths are treated relative to the root directory of firebird.
# Default value 'Full' gives full access to all files on your site.
# To specify access to specific trees, enum all required paths
# (for Windows this may be something like 'C:\DataBase;D:\Mirror',
# for unix - '/db;/mnt/mirrordb'). If you choose 'None', then only
# databases listed in databases.conf can be attached.
#
# Note: simple quotation marks shown above should *NOT* be used when
# specifying values and directory path names. Examples:
#
# DatabaseAccess = None
# DatabaseAccess = Restrict C:\DataBase
# DatabaseAccess = Restrict C:\DataBase;D:\Mirror
# DatabaseAccess = Restrict /db
# DatabaseAccess = Restrict /db;/mnt/mirrordb
DatabaseAccess = Full
#
```

4.- Making some changes in the configuration file

Save the changes and close the file. Then, to apply the changes, just restart the service.

```
:~$ sudo systemctl restart firebird3.0
```

Creating databases and tables

Now we can start creating new databases and tables.

First we connect again to the Firebird shell.

:~$ sudo isql-fb

And then, we proceed to create the new database with the following syntax:

```
:> create database ["database_path.fdb"] user ['user']password '[password]';
```

For example:

```
:> create database "/var/lib/firebird/3.0/data/first_database.fdb" user 'SYSDBA' passwo
```

Then, to use it or rather to connect to it we use this other command:

```
:> connect "/var/lib/firebird/3.0/data/first_database.fdb" user 'SYSDBA' password 'ange
```

Now we're ready to start creating tables.

For this example I will create a table with only two fields.

```
:> CREATE TABLE STUDENT (ID INT NOT NULL PRIMARY KEY, NAME VARCHAR(25));
```

So, you can show all the tables by using this command:

```
:> show tables;
```

```
SQL> CREATE TABLE STUDENT (ID INT NOT NULL PRIMARY KEY, NAME VARCHAR(25));
SQL> show tables;
       STUDENT
SQL>
```

6.- Creating table with two columns

Now, I am going to insert some data. Just use the following command:

```
:> INSERT INTO STUDENT VALUES (1, 'Angelo');
```

And get the data:

```
:> select * from STUDENT;
```

```
SQL> INSERT INTO STUDENT VALUES (1, 'Angelo');
SQL> select * from STUDENT;

          ID NAME
============ ========================
           1 Angelo

SQL>
```

7.- Using Firebird on Ubuntu 18.04

As you can see being SQL language, the commands and operations are similar to MySQL / MariaDB and PostgreSQL.

Finally, you can show the installed version:

```
:> show version;
```

```
SQL> show version;
ISQL Version: LI-V3.0.5.33100 Firebird 3.0
Server version:
Cannot get server version without database connection
SQL>
```

8.- Firebird on Ubuntu 18.04

And that is it.

Conclusion

Firebird may not be popular but is considered by many projects for its efficiency. Being of the SQL family its commands are not so different from those of other better known programs such as MySQL or MariaDB. So we only have to work with it.

angeloma

Senior Writer and partner