

Installing and setting up Mosquitto on Debian 10

This document's purpose is to explain how to install and set up mosquitto to use it on a debian. Unless specified otherwise all commands on the document have to be executed by a superuser of the system.

I - Getting Mosquitto

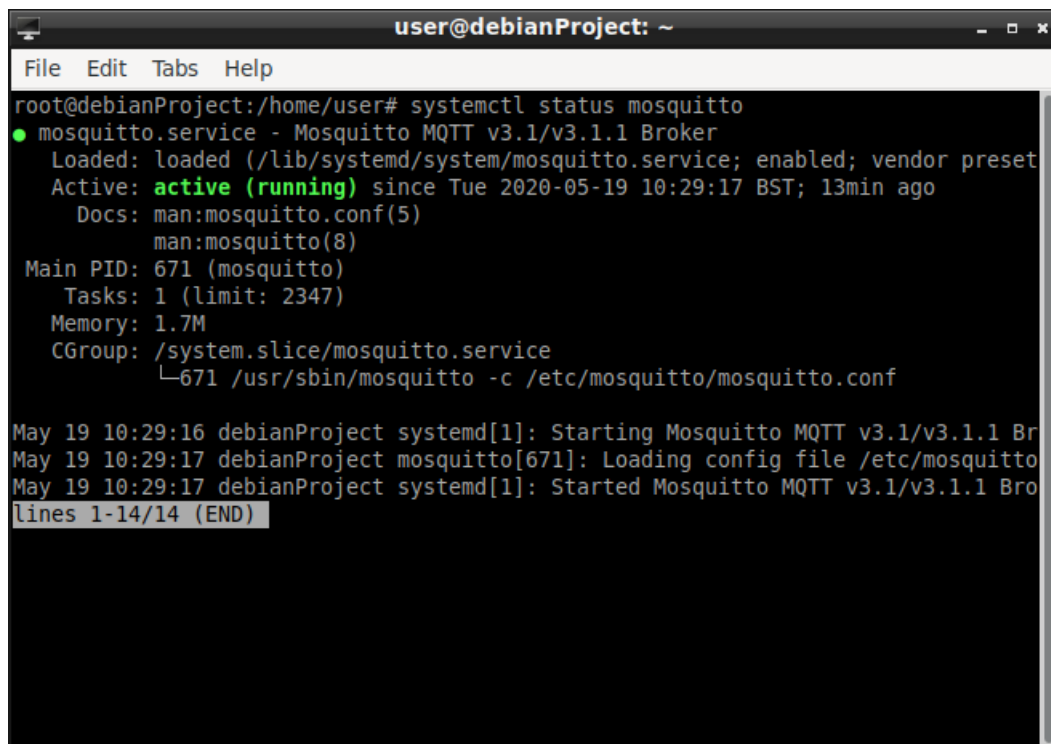
To get mosquitto you need to type in console :

1. `sudo apt update`
2. `sudo apt install mosquitto mosquitto-clients`

Debian start the service immediately, for testing if mosquitto is well installed you just need to ask the system with :

1. `systemctl status mosquitto`

You must obtain something similar to figure 1.

A terminal window titled 'user@debianProject: ~' showing the command 'systemctl status mosquitto' and its output. The output indicates that the mosquitto.service is loaded, enabled, and active (running) since Tue 2020-05-19 10:29:17 BST. It also shows the main PID as 671, tasks as 1, memory as 1.7M, and the CGroup as /system.slice/mosquitto.service. Log messages at the bottom show the service starting and loading the config file.

```
user@debianProject: ~
File Edit Tabs Help
root@debianProject:/home/user# systemctl status mosquitto
● mosquitto.service - Mosquitto MQTT v3.1/v3.1.1 Broker
   Loaded: loaded (/lib/systemd/system/mosquitto.service; enabled; vendor preset
   Active: active (running) since Tue 2020-05-19 10:29:17 BST; 13min ago
     Docs: man:mosquitto.conf(5)
           man:mosquitto(8)
   Main PID: 671 (mosquitto)
     Tasks: 1 (limit: 2347)
    Memory: 1.7M
     CGroup: /system.slice/mosquitto.service
            └─671 /usr/sbin/mosquitto -c /etc/mosquitto/mosquitto.conf

May 19 10:29:16 debianProject systemd[1]: Starting Mosquitto MQTT v3.1/v3.1.1 Br
May 19 10:29:17 debianProject mosquitto[671]: Loading config file /etc/mosquitto
May 19 10:29:17 debianProject systemd[1]: Started Mosquitto MQTT v3.1/v3.1.1 Br
lines 1-14/14 (END)
```

Figure 1: Status of mosquitto

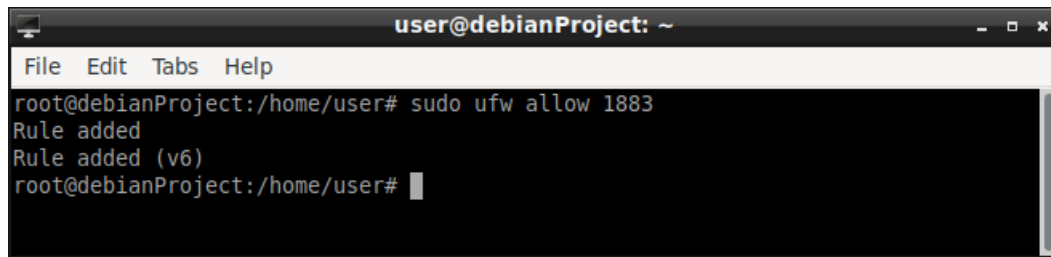
By default all access ports are closed on Debian, but we need to allow access on port 1883, the default port for MQTT communication. We will use ufw, if it's not installed simply use

1. `apt install ufw`

and then

1. `sudo ufw allow 1883`

You should get the same message as on figure 2.

A terminal window titled 'user@debianProject: ~' with a menu bar containing 'File', 'Edit', 'Tabs', and 'Help'. The terminal shows the command 'root@debianProject:/home/user# sudo ufw allow 1883' being executed. The output consists of two lines: 'Rule added' and 'Rule added (v6)'. The prompt returns to 'root@debianProject:/home/user#' with a cursor at the end.

```
user@debianProject: ~
File Edit Tabs Help
root@debianProject:/home/user# sudo ufw allow 1883
Rule added
Rule added (v6)
root@debianProject:/home/user#
```

Figure 2: allowing port 1883

Your MQTT server is now online. You can use any client on your local network to access it. We used two clients at this stage :

MyMQTT on android : <https://play.google.com/store/apps/details?id=at.tripwire.mqtt.client>
and

MQTTExplorer on Windows : <http://mqtt-explorer.com/>

MQTTExplorer is also available on Mac and Linux.

II - Defining users

At this point anyone can connect to the broker to publish or see any of the information on it. We don't want that, it's too dangerous and will put the stability of our solution at risk. To counter this we will add account and the necessity of login to access information.

First we create the password file with the information of the first user

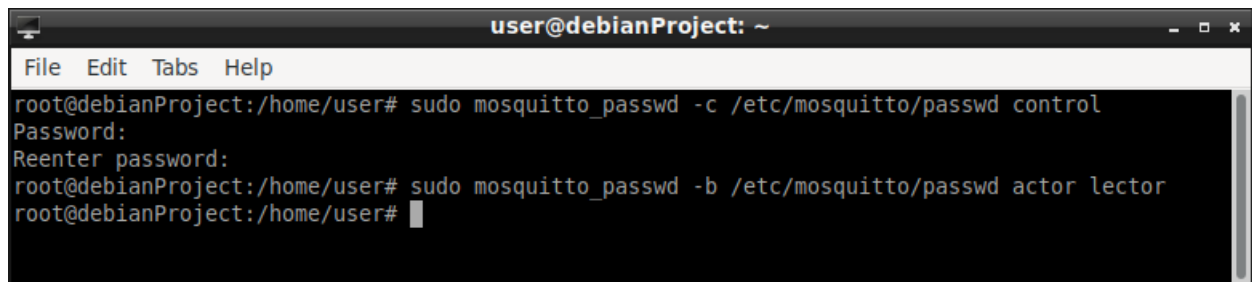
```
1. sudo mosquitto_passwd -c /etc/mosquitto/passwd control
```

In this example, -c create the file, “/etc/mosquitto/passwd” is the name of the file and “control” is the user ID, you are then asked to enter and confirm the password for “control”.

You only need to create the file once then you can add user with the following command :

```
1. sudo mosquitto_passwd -b /etc/mosquitto/passwd actor lector
```

Here -b is to add user, “actor” is the identifiant and “lector” is the password.
The console will look like figure 3.



```
user@debianProject: ~  
File Edit Tabs Help  
root@debianProject:/home/user# sudo mosquitto_passwd -c /etc/mosquitto/passwd control  
Password:  
Reenter password:  
root@debianProject:/home/user# sudo mosquitto_passwd -b /etc/mosquitto/passwd actor lector  
root@debianProject:/home/user#
```

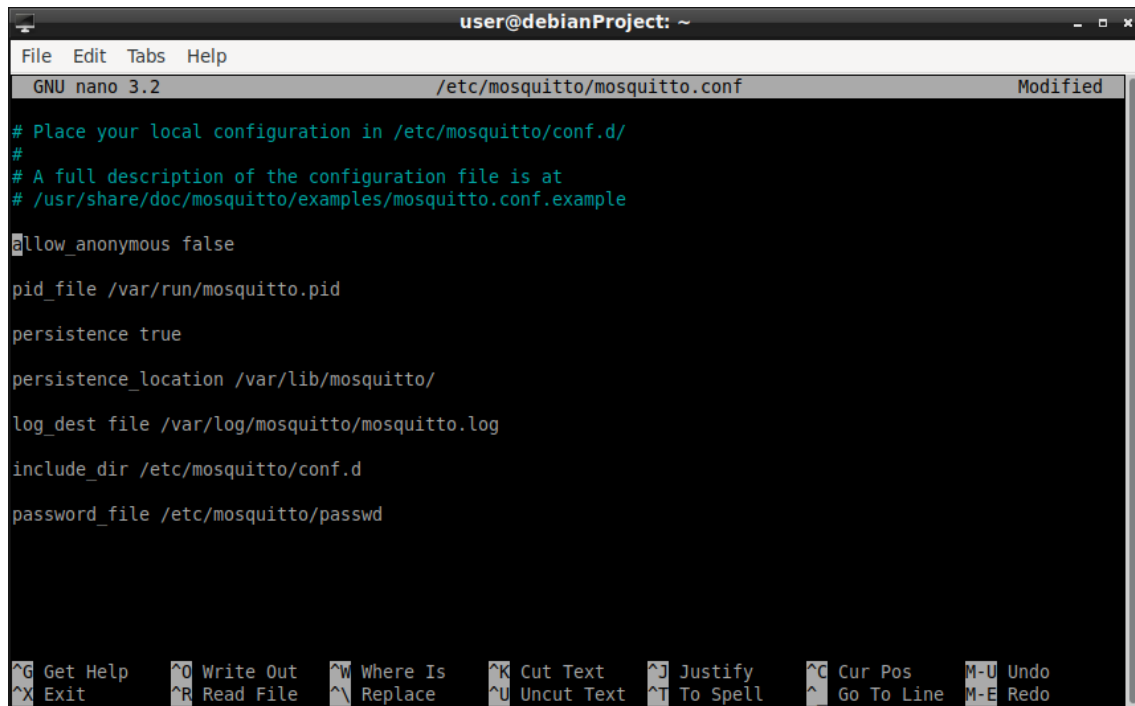
Figure 3: Definition of users

You can add as many user as you want or need with the second command.

Now the users are defined but we need to let the server know where they are defined. To do so we modify the mosquitto.conf file. It's located at /etc/mosquitto/mosquitto.conf, we will use nano to add “allow anonymous false” and “password file /etc/mosquitto/passwd”.

```
1. nano /etc/mosquitto/mosquitto.conf
```

Your mosquitto.conf file should be similar to figure 4.



```
user@debianProject: ~
File Edit Tabs Help
GNU nano 3.2 /etc/mosquitto/mosquitto.conf Modified

# Place your local configuration in /etc/mosquitto/conf.d/
#
# A full description of the configuration file is at
# /usr/share/doc/mosquitto/examples/mosquitto.conf.example

allow_anonymous false

pid_file /var/run/mosquitto.pid

persistence true

persistence_location /var/lib/mosquitto/

log_dest file /var/log/mosquitto/mosquitto.log

include_dir /etc/mosquitto/conf.d

password_file /etc/mosquitto/passwd

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo
^X Exit      ^R Read File  ^\ Replace  ^U Uncut Text ^T To Spell  ^_ Go To Line M-E Redo
```

Figure 4: mosquitto.conf

It is needed to reload the service by using :

1. systemctl restart mosquitto

Now it's mandatory to dispose of a valid login and password to access the broker

III – Adding Access Control

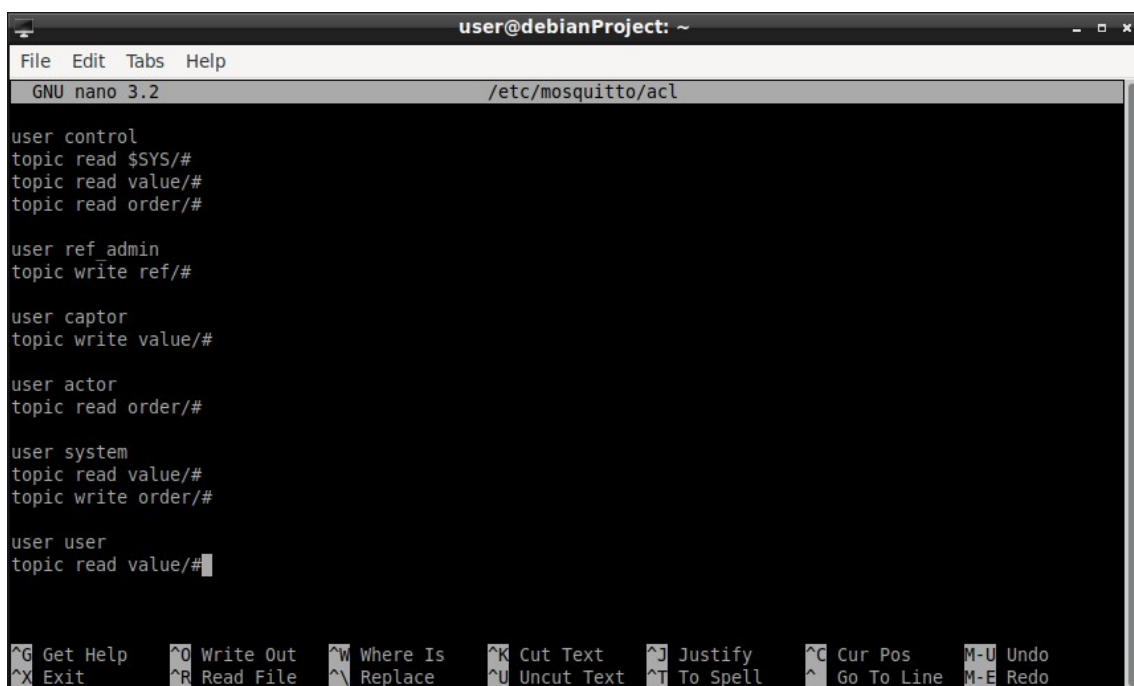
At this point only users defined in passwd can access to the server, but any user can access any information, if you want to restrict the access of each user you need to define an acl file. We will use nano again.

1. `nano /etc/mosquitto/acl`

When using the acl file everything is inaccessible unless specified otherwise, at the beginning of the file you can specify what anonymous client can see, it doesn't interest us because they are banned. Then you can define authorization for a specific user as follow :

1. user system
2. topic read value/#
3. topic write order/#

“system” is the username of the new user, topic is necessary if you write/read .The client can only subscribe, for write it can only publish. If you write nothing it can do both, then “value/” is the subject concerned, if you write “value/#” it also gave access to sub-subject. Then you obtain a file like figure 5.

The image shows a terminal window with the nano text editor open. The title bar reads 'user@debianProject: ~'. The editor's menu bar includes 'File', 'Edit', 'Tabs', and 'Help'. The status bar at the bottom shows various keyboard shortcuts like '^G Get Help', '^O Write Out', '^W Where Is', etc. The main text area shows the content of the /etc/mosquitto/acl file, which defines permissions for several users: 'control', 'ref_admin', 'captor', 'actor', 'system', and 'user'. Each user entry is followed by a 'topic' and a permission (read or write) for a specific topic or all topics (indicated by #).

```
user control
topic read $SYS/#
topic read value/#
topic read order/#

user ref_admin
topic write ref/#

user captor
topic write value/#

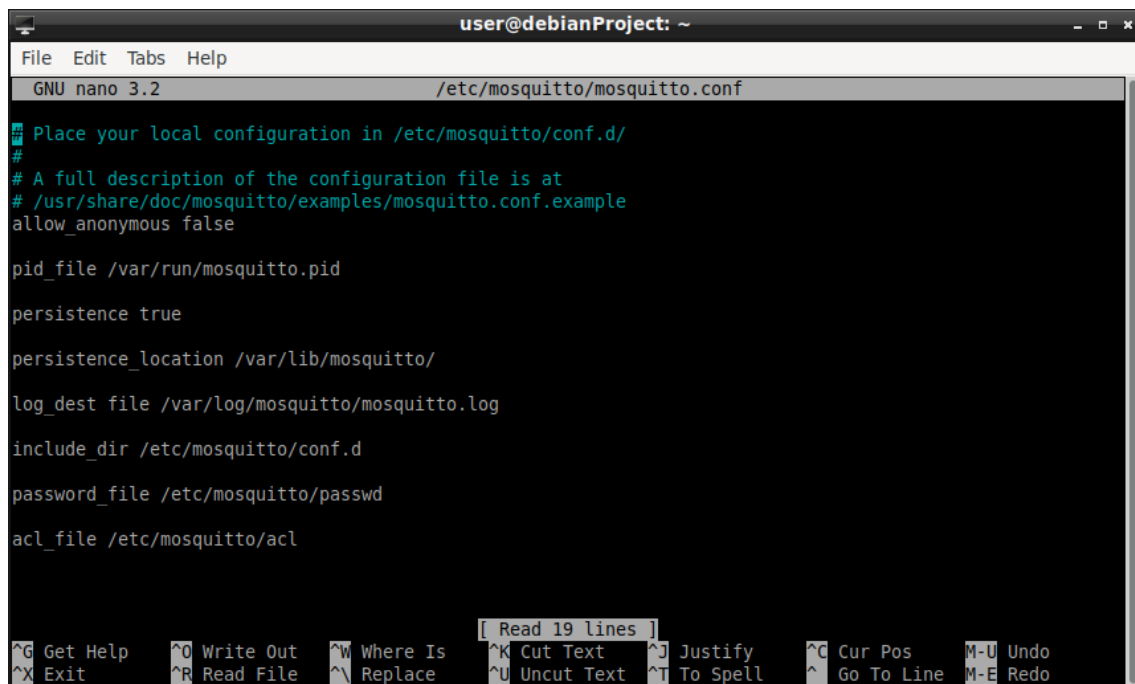
user actor
topic read order/#

user system
topic read value/#
topic write order/#

user user
topic read value/#
```

Figure 5: acl file

The file is now ready to be used. We need to specify to mosquitto to use it. We go back to modifying the mosquitto.conf with nano by adding “acl_file /etc/mosquitto/acl”.



```
user@debianProject: ~  
File Edit Tabs Help  
GNU nano 3.2 /etc/mosquitto/mosquitto.conf  
# Place your local configuration in /etc/mosquitto/conf.d/  
#  
# A full description of the configuration file is at  
# /usr/share/doc/mosquitto/examples/mosquitto.conf.example  
allow_anonymous false  
  
pid_file /var/run/mosquitto.pid  
  
persistence true  
  
persistence_location /var/lib/mosquitto/  
  
log_dest file /var/log/mosquitto/mosquitto.log  
  
include_dir /etc/mosquitto/conf.d  
  
password_file /etc/mosquitto/passwd  
  
acl_file /etc/mosquitto/acl  
  
[ Read 19 lines ]  
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo  
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line M-E Redo
```

Figure 6: *mosquitto.conf* with access control

All we need to do is to reload the server. The server is now configured with a login feature restricting access to specific subjects.