

Jungletronics · [Follow publication](#)

Mosquitto — User Access Configurations Setups

Editing mosquitto.conf File to Configure Simple Authentication— MQTT — Episode #02

6 min read · Jan 13, 2020



J3

[Follow](#)

Listen

Share

*Hi, let's see first the mosquitto **anonymous access** communications. This post continues [MQTT — Episode #01](#):*

In the end, we will force access to mosquitto's broker via login/password only.



Fig 1. What are the procedures to set **login/password access** to mosquitto's broker?

Let us begin by **not** editing this file(I'm using mosquitto v 1.6.8):

C:\Program Files\mosquitto**mosquitto.conf**

The **mosquitto.conf** is the configuration file for mosquitto. In this first configuration, a similar internal file will be loaded by mosquito broker automatically, and its default configuration authorizes anonymous access:/

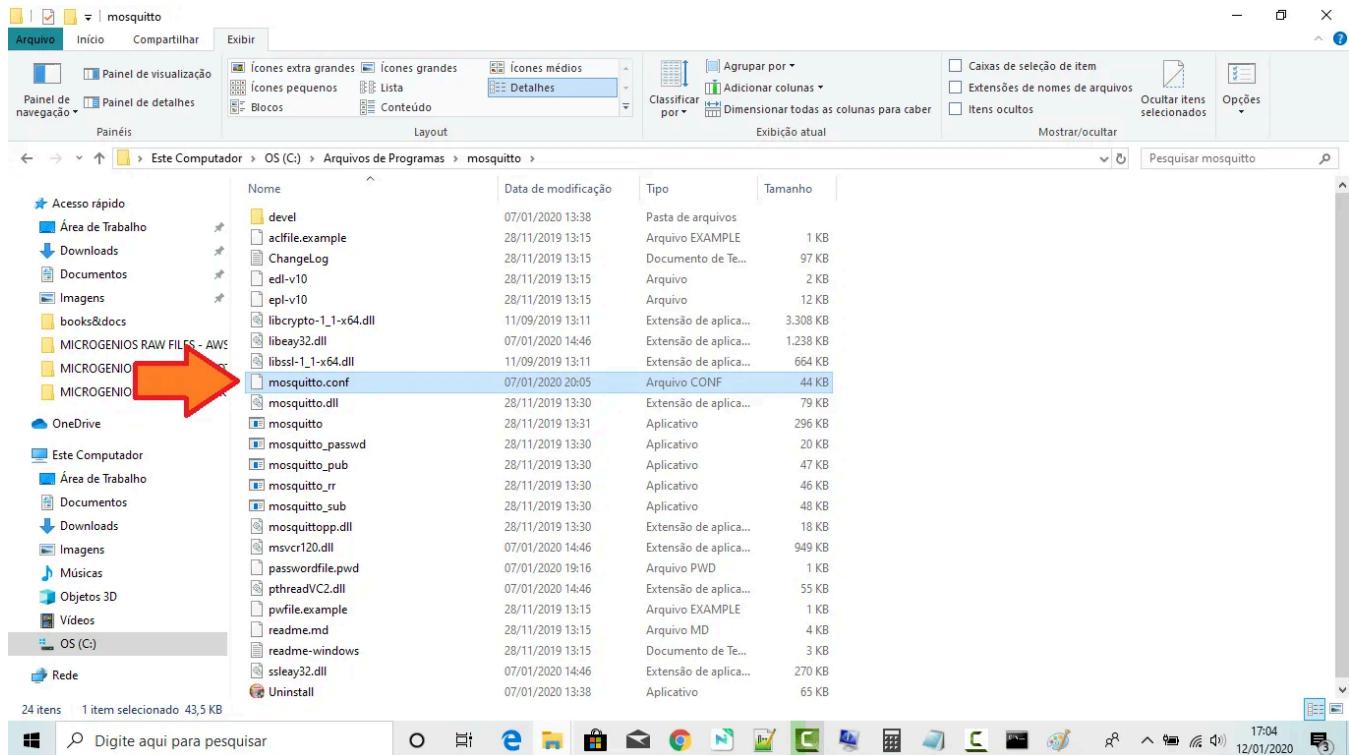


Fig 2. On Windows mosquitto.conf are located at **C:\Program Files\mosquitto**

In anonymous access, we will now up the default broker configuration, remember, without modification of mosquitto.conf :)

Here are the steps:

0 ° Step – Install mosquito broker in your machine & reconfigure its service in Win10 so we can start & stop it manually;

Consider following [this](#) post:)

Preparing 3 Prompters Terminals

Open Three Terminals; type these commands in each one:

```
cd.. (2x) // go to c:/> directory
cd C:\Program Files\mosquitto // change to mosquito directory
cls // clear the screen;)
```

First, anonymous access:

Anonymous Acess

1 ° Step –Let's run the server:

On Term1, in C:\Program Files\mosquitto\ directory, type:

mosquitto -v

```
C:\Program Files\mosquitto>mosquitto -v
1578866791: mosquitto version 1.6.8 starting
1578866791: Using default config.
1578866791: Opening ipv6 listen socket on port 1883.
1578866791: Opening ipv4 listen socket on port 1883.
1578866795: New connection from ::1 on port 1883.
1578866795: New client connected from ::1 as mosq-vOTNPVcbuC8biGn9fT (p2, c1, k60).
1578866795: No will message specified.
1578866795: Sending CONNACK to mosq-vOTNPVcbuC8biGn9fT (0, 0)
1578866795: Received SUBSCRIBE from mosq-vOTNPVcbuC8biGn9fT
1578866795:     temperature (QoS 0)
1578866795: mosq-vOTNPVcbuC8biGn9fT 0 temperature
1578866795: Sending SUBACK to mosq-vOTNPVcbuC8biGn9fT
1578866799: New connection from ::1 on port 1883.
1578866799: New client connected from ::1 as mosq-QkuFDL8tbQyAS10Bky (p2, c1, k60).
1578866799: No will message specified.
1578866799: Sending CONNACK to mosq-QkuFDL8tbQyAS10Bky (0, 0)
1578866799: Received PUBLISH from mosq-QkuFDL8tbQyAS10Bky (d0, q0, r0, m0, 'temperature', ... (2 bytes))
1578866799: Sending PUBLISH to mosq-vOTNPVcbuC8biGn9fT (d0, q0, r0, m0, 'temperature', ... (2 bytes))
1578866799: Received DISCONNECT from mosq-QkuFDL8tbQyAS10Bky
1578866799: Client mosq-QkuFDL8tbQyAS10Bky disconnected.
1578866855: Received PINGREQ from mosq-vOTNPVcbuC8biGn9fT
1578866855: Sending PINGRESP to mosq-vOTNPVcbuC8biGn9fT
1578866915: Received PINGREQ from mosq-vOTNPVcbuC8biGn9fT
1578866915: Sending PINGRESP to mosq-vOTNPVcbuC8biGn9fT
```

Fig 3. As we run the server with **-v (verbose)** all event will be dump to the terminal

2 Step — At others two terminals in sequence for **sub & pub** clients, type:

Term_2, For **_sub**, in C:\Program Files\mosquitto\ directory, type:

mosquitto_sub -h localhost -p 1883 -t temperature

Term_3, For **_pub**, in C:\Program Files\mosquitto\ directory, type:

mosquitto_pub -h localhost -p 1883 -t temperature -m 45

The screenshot shows three separate command-line windows running simultaneously on a Windows desktop. The top window displays the Mosquitto broker log output, showing connections from two clients (p2, c1) and their corresponding messages. The middle window shows a client publishing a 'temperature' message to the broker. The bottom window shows another client subscribing to the 'temperature' topic and receiving the published message.

```

C:\Program Files\mosquitto>mosquitto -v
1578866791: mosquitto version 1.6.8 starting
1578866791: Using default config.
1578866791: Opening ipv6 listen socket on port 1883.
1578866791: Opening ipv4 listen socket on port 1883.
1578866795: New connection from ::1 on port 1883.
1578866795: New client connected from ::1 as mosq-vOTNPVcbuC8biGn9fT (p2, c1, k60).
1578866795: No will message specified.
1578866795: Sending CONNACK to mosq-vOTNPVcbuC8biGn9fT (0, 0)
1578866795: Received SUBSCRIBE from mosq-vOTNPVcbuC8biGn9fT
1578866795:     temperature (QoS 0)
1578866795: mosq-vOTNPVcbuC8biGn9fT 0 temperature
1578866795: Sending SUBACK to mosq-vOTNPVcbuC8biGn9fT
1578866799: New connection from ::1 on port 1883.
1578866799: New client connected from ::1 as mosq-QkuFDL8tbQyAS10Bky (p2, c1, k60).
1578866799: No will message specified.
1578866799: Sending CONNACK to mosq-QkuFDL8tbQyAS10Bky (0, 0)
1578866799: Received PUBLISH from mosq-QkuFDL8tbQyAS10Bky (d0, q0, r0, m0, 'temperature', ... (2 bytes))

C:\Program Files\mosquitto>mosquitto_sub -h localhost -p 1883 -t temperature
45

C:\Program Files\mosquitto>mosquitto_pub -h localhost -p 1883 -t temperature -m 45
C:\Program Files\mosquitto>
```

Fig 4. All three terminals running together!

See the dump file in the admin broker terminal above. See that the clients are **anonymous** — this is very dangerous on the internet:/

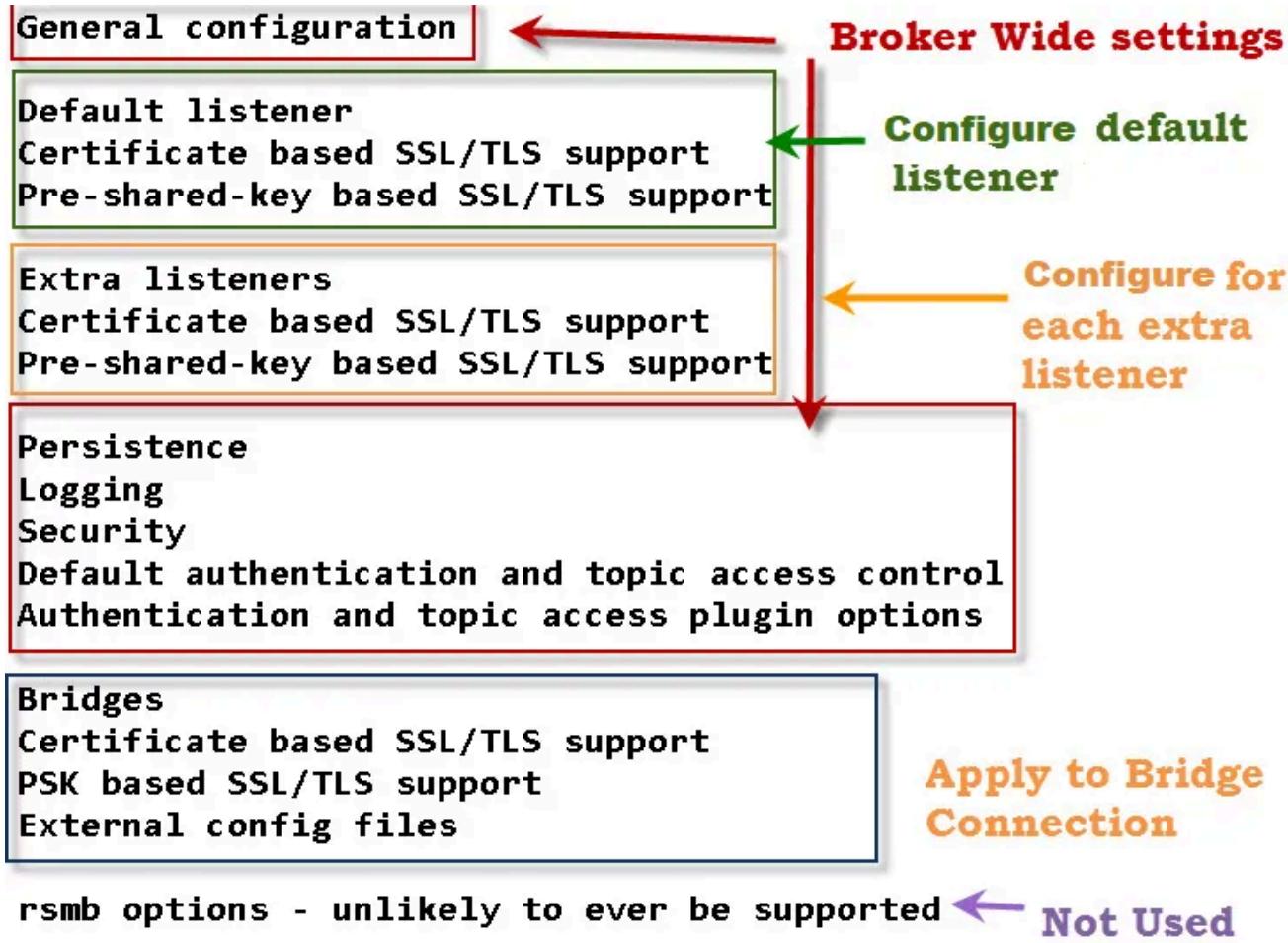
1578781497: New client connected from ::1 as mosq-5eKhup3oVmPBaNaZnW (**p2, c1, k60**).

Simple Authentication Access

Now let's fix anonymous access by setting a `login/passwd` connection.

Anonymous clients will be refused to connect.

For this, we will have to edit, as administrator credentials, the `mosquitto.conf` file. Here is the structure of it:



Mosquitto.conf Structure

Fig 5. Here is the global structure of the mosquitto.conf file (from [Quick Guide to The Mosquitto.conf File With Examples](#))

As you know, this file is located at **system file**, so we need to open this file in **notepad++** editor like administrator:

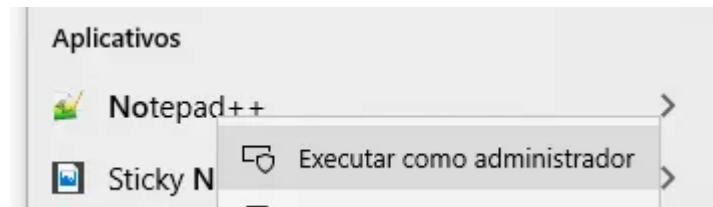


Fig 6. We will use notepad ++ for mosquitto.conf editing — Note: You can configure a broker to listen on a port and **require SSL** and also to listen on another port and **not use SSL**.

To create a password file you need to use the **mosquitto_passwd** utility that comes with the client tools when installing the mosquitto broker.

To initialize the configuration of authenticated access, at **Term_01**, we will need to stop the server (ctrl + C) and use the command below to create **passwordfile.pwd**

file that will be located at C:\Program Files\mosquitto directory; our administrator will be admin user and our password will be 123;

1 ° Step —On Term 1, on C:\Program Files\mosquitto\ directory, type:

```
mosquitto_passwd -c "C:\Program Files\mosquitto\passwordfile.pwd"  
admin
```

and hit <enter> and enter 2 x the chosen password (123);

This command creates passwordfile.pwd file and set as our administrator **admin** user, password 123;

Your file for user configuration is ready at C:\ProgramFiles\mosquitto\ directory!

Admin is the name of our first and our main user;

2 ° Step — Go to C:\Program Files\mosquitto directory and see the file created there:

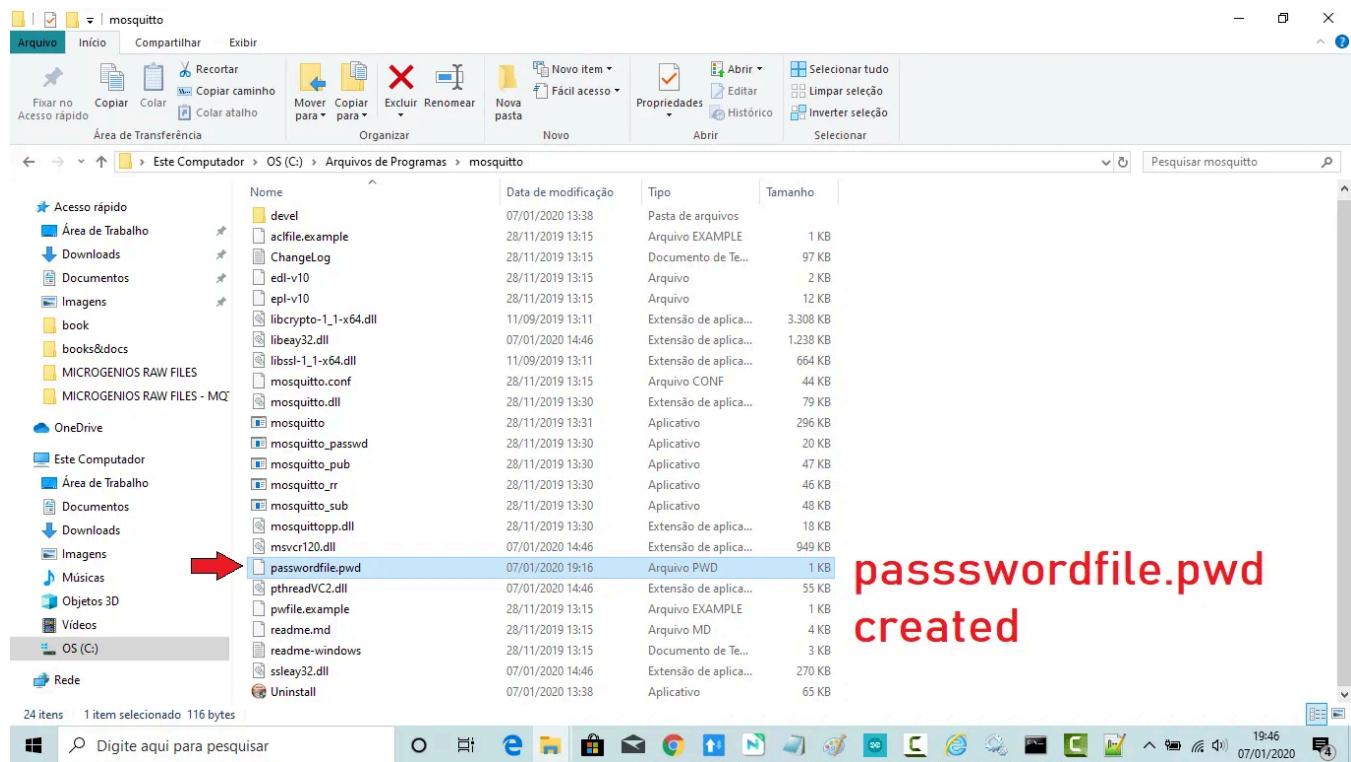


Fig 7. the password file will be created in the C:\Program Files\mosquitto\ directory

If we open this new file as text, **passwordfile.pwd**, we will see the **admin** user credentials:

```
admin:$6$sEorGWHKkOfEI8qJ$nxEMynuvKuguXqbYq7TWBsSAxEDon/MuK0pFo4Cm0y
0K29m/I0yi6y3zFzuJeFXRT9DgyVVLDS/w072CADlIaw==
```

3

°Step — Now open C:\Program Files\mosquitto\mosquitto.conf.

At line 651, uncomment it and set it to false, like this:

```
allow_anonymous false
```

```
638 # client" could connect but another with clientid "mqtt" couldn't.
639 #clientid_prefixes
640
641 # Boolean value that determines whether clients that connect
642 # without providing a username are allowed to connect. If set to
643 # false then a password file should be created (see the
644 # password_file option) to control authenticated client access.
645 #
646 # Defaults to true if no other security options are set. If 'password_file' or
647 # 'psk_file' is set, or if an authentication plugin is loaded which implements
648 # username/password or TLS-PSK checks, then 'allow_anonymous' defaults to
649 # false.
650 #
651 allow_anonymous false
```

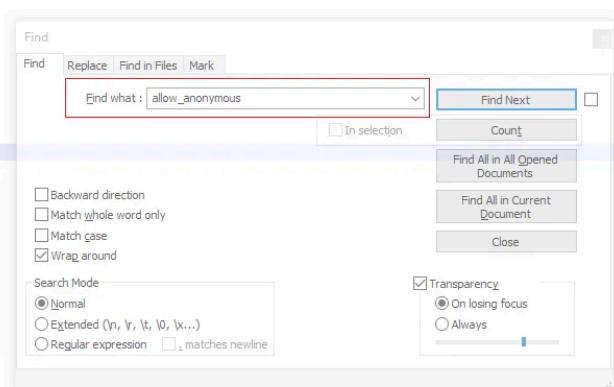


Fig 8. in the conf file, please comment this line and type: **allow_anonymous false**

4

°Step — Scrolling down enough until line 669, uncomment it and set it to the directory location of password file config, like this:

```
password_file "C:\Program Files\mosquitto\passwordfile.pwd"
```

```

648 # username/password or TLS-PSK checks, then `allow_anonymous` defaults to
649 # false.
650 #
651 allow_anonymous false
652 #
653 # -----
654 # Default authentication and topic access control
655 # -----
656
657 # Control access to the broker using a password file. This file can be
658 # generated using the mosquitto_passwd utility. If TLS support is not compiled
659 # into mosquitto (it is recommended that TLS support should be included) then
660 # plain text passwords are used, in which case the file should be a text file
661 # with lines in the format:
662 # username:password
663 # The password (and colon) may be omitted if desired, although this
664 # offers very little in the way of security.
665 #
666 # See the TLS client require_certificate and use_identity_as_username options
667 # for alternative authentication options. If an auth_plugin is used as well as
668 # password file, the auth plugin check will be made first.
669 password_file C:\Program Files\mosquitto\passwordfile.pwd
670
671 # Access may also be controlled using a pre-shared-key file. This requires
672 # TLS-PSK support and a listener configured to use it. The file should be text
673 # lines in the format:
674 # identity:key
675 # The key should be in hexadecimal format without a leading "0x".
676 # If an auth_plugin is used as well, the auth_plugin check will be made first.
677 #psk_file
678
679 # Control access to topics on the broker using an access control list
680 # file. If this parameter is defined then only the topics listed will
681 # have access.
682 # If the first character of a line of the ACL file is a # it is treated as a
683 # comment.
684 # Topic access is added with lines of the format:
685 #

```

Normal text file length: 44.598 lines: 989 Ln: 669 Col: 1 Sel: 0 | 0 Unix (LF) UTF-8 INS

Digitate aqui para pesquisar 0 e f m o s c n d i r t l v u b w h g p y k j z ^ = < > & < > >> << >>> <<< >>> <<<< >>>>

Fig 9. Tell the mosquito server where to find the password file

Save the file and now start the server again.

5 °Step – At Terminal 1, type (run as Administrator, please):

```
<Ctrl + C> // To stop the server
mosquitto -c mosquito.conf -v // to run it with loaded file
```

This will break the connection of the anonymous client:

```
C:\Program Files\mosquitto>mosquitto -c mosquitto.conf -v
1578868128: mosquitto version 1.6.8 starting
1578868128: Config loaded from mosquitto.conf.
1578868128: Opening ipv6 listen socket on port 1883.
1578868128: Opening ipv4 listen socket on port 1883.
1578868128: New connection from ::1 on port 1883.
1578868128: Sending CONNACK to ::1 (0, 5)
1578868128: Socket error on client <unknown>, disconnecting.

C:\Program Files\mosquitto>mosquitto_sub -h localhost -p 1883 -t temperature
45
Connection error: Connection Refused: not authorised.

C:\Program Files\mosquitto>
```

The screenshot shows three separate command-line windows. The top window displays the Mosquitto broker starting up and accepting a connection from the local host. The middle window shows an attempt to subscribe to the 'temperature' topic, which fails with a 'Connection Refused: not authorised' error. The bottom window is currently empty, awaiting further input.

Fig 10. See this message in the second terminals: **Connection error: Connection Refused: not authorised.**
(sic)

6 °Step – Now let's test if the admin user can **subscribe** topic:

At Terminal 2, type:

```
mosquitto_sub -h localhost -p 1883 -u admin -P 123 -t temperature
```

```
C:\Program Files\mosquitto>mosquitto -c mosquitto.conf -v
1578868128: mosquitto version 1.6.8 starting
1578868128: Config loaded from mosquitto.conf.
1578868128: Opening ipv6 listen socket on port 1883.
1578868128: Opening ipv4 listen socket on port 1883.
1578868128: New connection from ::1 on port 1883.
1578868128: Sending CONNACK to ::1 (0, 5)
1578868128: Socket error on client <unknown>, disconnecting.
1578868846: New connection from ::1 on port 1883.
1578868846: New client connected from ::1 as mosq-Ne2cx001c2WTp8eEU7 (p2, c1, k60, u'admin').
1578868846: No will message specified.
1578868846: Sending CONNACK to mosq-Ne2cx001c2WTp8eEU7 (0, 0)
1578868846: Received SUBSCRIBE from mosq-Ne2cx001c2WTp8eEU7
1578868846:   temperature (QoS 0)
1578868846: mosq-Ne2cx001c2WTp8eEU7 0 temperature
1578868846: Sending SUBACK to mosq-Ne2cx001c2WTp8eEU7

C:\Program Files\mosquitto>mosquitto_sub -h localhost -p 1883 -u admin -P 123 -t temperature
```

```
C:\Program Files\mosquitto>mosquitto_pub -h localhost -p 1883 -t temperature -m 45
```

The screenshot shows four command-line windows. The first two windows show the Mosquitto broker starting and accepting a connection from the local host. The third window shows the 'admin' user successfully subscribing to the 'temperature' topic. The fourth window shows a message being published to the 'temperature' topic with payload '45'. The windows are arranged vertically, with the third and fourth windows sharing the same terminal window.

Fig 11. Admin user **subscribing** at the second terminal

7

°Step — Now let's test if the admin user can **publish** to the same topic.

At Terminal 3, type (run as Administrator, please):

```
mosquitto_pub -h localhost -p 1883 -u admin -P 123 -t temperature -m 45
```

The screenshot shows three separate Command Prompt windows side-by-side. The top window has a light gray background and displays log messages from a Mosquitto broker. The middle window has a white background and shows the command being run. The bottom window has a white background and also shows the command being run. A red rectangular box highlights the command in the bottom window.

```
Prompt de Comando - mosquitto -c mosquitto.conf -v
1578869217: New client connected from ::1 as mosq-EE0o2SjU00f6iGxaTP (p2, c1, k60, u'admin').
1578869217: No will message specified.
1578869217: Sending CONNACK to mosq-EE0o2SjU00f6iGxaTP (0, 0)
1578869217: Received SUBSCRIBE from mosq-EE0o2SjU00f6iGxaTP
1578869217:    temperature (QoS 0)
1578869217: mosq-EE0o2SjU00f6iGxaTP 0 temperature
1578869217: Sending SUBACK to mosq-EE0o2SjU00f6iGxaTP
1578869249: New connection from ::1 on port 1883.
1578869249: New client connected from ::1 as mosq-hSua0zhgXx1VkfWGlK (p2, c1, k60, u'admin').
1578869249: No will message specified.
1578869249: Sending CONNACK to mosq-hSua0zhgXx1VkfWGlK (0, 0)
1578869249: Received PUBLISH from mosq-hSua0zhgXx1VkfWGlK (d0, q0, r0, m0, 'temperature', ... (2 bytes))
1578869249: Sending PUBLISH to mosq-EE0o2SjU00f6iGxaTP (d0, q0, r0, m0, 'temperature', ... (2 bytes))
1578869249: Received DISCONNECT from mosq-hSua0zhgXx1VkfWGlK
1578869249: Client mosq-hSua0zhgXx1VkfWGlK disconnected.

Prompt de Comando - mosquitto_sub -h localhost -p 1883 -u admin -P 123 -t temperature
C:\Program Files\mosquitto>mosquitto_sub -h localhost -p 1883 -u admin -P 123 -t temperature
45

Prompt de Comando
C:\Program Files\mosquitto>mosquitto_pub -h localhost -p 1883 -u admin -P 123 -t temperature -m 45
C:\Program Files\mosquitto>
```

Fig 12. Admin user **publishing** at the third terminal

And there you have it! **no more anonymous user!**

```
1578789217: New client connected from ::1 as mosq-EE0o25jU00f6IGxaTP
(p2, c1, k60, u'admin').
```

In the next MQTT episode, let's use **ACL (Access Control List)**.

Thanks and bye, for now, o/

[Download All Files For This Project](#)

MQTT Related Posts

- 01** # Episode — Mosquitto — Intro To MQTT — It is Suitable for the Internet of Things Applications — MQTT
- 02** # Episode — Mosquitto — User Access Configurations Setups — Editing mosquito.conf File to Configure SSL Authentications — MQTT (this one)
- 03** # Episode — Mosquitto — Mosquitto — ACLs — Wildcards & ACL — access control lists — MQTT
- 04** # Episode — Mosquitto — MQTT QoS — How To Set QoS at Mosquitto Broker — MQTT
- 05** # Episode — Mosquitto — Bulletproof TLS & SSL Mosquitto — How To Set Up Mosquitto Broker/Client Keys & Certificates — MQTT
- 06** # Episode — Mosquitto — Mosquitto Bridge — How To Bridge Two Mosquitto Brokers — MQTT
- 07** ...be tuned for the upcoming post about MQTT and IoT o/

References & Credits

Eclipse Mosquitto™ An open source MQTT broker

Microgênicos — Treinamento em Sistemas Embarcados — Microchip Regional Partner — Microchip Certified Brazilian Training Education Company & a Simplício-Owned enterprise o/

How to change your computer's name in Windows 10

There are plenty of reasons to change your computer's name -- especially if you bought it directly from a manufacturer...

www.cnet.com

Quick Guide to The Mosquitto.conf File With Examples

You can configure the mosquito broker using a configuration file. The default configuration file is called...

www.steves-internet-guide.com

Notes About mosquitto

- After installation and every PC restart, the mosquitto program starts automatically;
- Mosquitto runs as a service and operates in the background;
- If a new configuration is made in mosquitto it is common to STOP and START the mosquitto MQTT service;
- Through shell, it is possible to reconfigure mosquitto without restarting the service;
- The clients are constantly sending PING to the server, that respond at about each **60 seconds** intervals (); follow an extract:

157878**1078**: Sending PINGRESP to mosq-nSDsiy1MZKG4ShY5D0

157878**1138**: Received PINGREQ from mosq-nSDsiy1MZKG4ShY5D0

- This is because HTPP protocol, which mosquitto is under, tends to disconnect the clients;

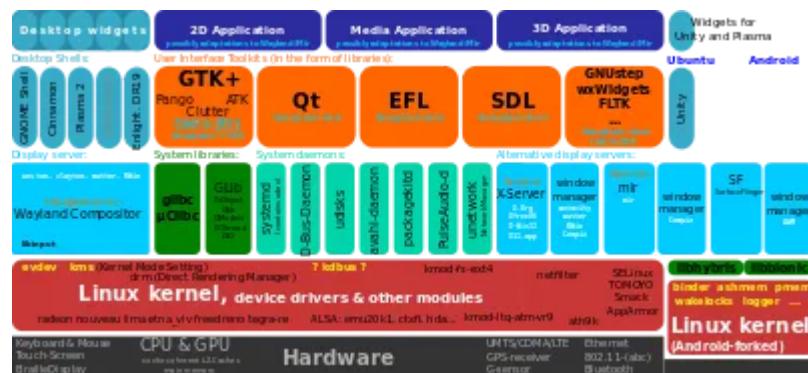
Sintaxe

Fig 4. Mosquitto syntax complete

Notes About This Post

Daemon (computing)

From Wikipedia, the free encyclopedia



Components of some Linux desktop environments that are daemons include D-Bus, NetworkManager (here called unetwork), PulseAudio (usound), and Avahi.

In multitasking computer operating systems, a daemon (/di:mən/ or /deɪmən/)^[1] is a computer program that runs as a background process, rather than being under the direct control of an interactive user. Traditionally, the process names of a daemon end with the letter *d*, for clarification that the process is in fact a daemon, and for differentiation between a daemon and a normal computer program. For example, syslogd is the daemon that implements the system logging facility, and sshd is a daemon that serves incoming SSH connections. (from [https://en.wikipedia.org/wiki/Daemon_\(computing\)](https://en.wikipedia.org/wiki/Daemon_(computing)))

Review

All commands in this post: [Open 3 terminals as Administrator]

Subtitles: [terminal number/file] command description

For Anonymous access:

```
[1][2][3] cd C:\Program Files\mosquitto
[1][2][3] cls
[1] mosquitto -v
[2] mosquitto_sub -h localhost -p 1883 -t temperature
[3] mosquitto_pub -h localhost -p 1883 -t temperature -m 45
```

For Secure Access:

```
[1] mosquitto_passwd -c C:\ProgramFiles\mosquitto\passwordfile.pwd
admin
```

```
[mosquitto.conf] allow_anonymous false
[mosquitto.conf] password_file C:\Program
Files\mosquitto\passwordfile.pwd
```

```
[1] mosquitto -c mosquitto.conf -v
[2] mosquitto_sub -h localhost -p 1883 -u admin -P 123 -t
temperature
[3] mosquitto_pub -h localhost -p 1883 -u admin -P 123 -t
temperature -m 45
```

Mqtt

Mqtt Broker

Mqtt Secure Broker

How To

Step By Step Guide