Mosquitto MQTT broker and clients

START HERE

mosquitto_sub -t *<topic>* **-v**
Listen, show topic

mosquitto_sub -t '$SYS/#' -v
Broker statistics

mosquitto_pub -t *<topic>* **-m** *<msg>*
Publish a message

mosquitto_pub -q 1 -t *<topic>* **-m** *<msg>*
At-least-once

mosquitto_pub -r -t *<topic>* **-m** *<msg>*
Retain last value

mosquitto_sub -h *<host>* **-p** *<port>* **-u** *<user>* **-P** *<pw>* **-t** *<topic>*
Remote, with login

mosquitto_sub -t '#' -v
Everything (careful)

mosquitto_passwd -c *<file>* *<user>*
Create a user file

Publisher mosquitto_pub → **topic + payload** → **Broker** mosquitto → **matching filters** → **Subscriber** mosquitto_sub

CONCEPTS

Clients never talk to each other: publishers and subscribers meet at a topic on the broker.

Topic wildcards

Filter	Matches
+	exactly one level: a/+/c
#	all remaining levels, last only: a/#
$SYS/#	broker stats; not matched by a lone #

Quote '#' and '$SYS' in the shell, or it expands or becomes a comment.

PUBLISH

mosquitto_pub -t *<topic>* **-m** *<msg>*
Publish one message

mosquitto_pub -t *<topic>* **-f** *<file>*
Send a file as the payload

mosquitto_pub -t *<topic>* **-s** *< <file>*
Send all of stdin as one message

<cmd> **| mosquitto_pub -t** *<topic>* **-l**
Send each stdin line as a message

mosquitto_pub -t *<topic>* **-n**
Send an empty payload
With -r this clears the retained message on the topic.

mosquitto_pub -r -t *<topic>* **-m** *<msg>*
Retain message on the broker
New subscribers get the last retained value immediately.

mosquitto_pub -q *<0|1|2>* **-t** *<topic>* **-m** *<msg>*
Set delivery guarantee
Default is 0 (at most once); 2 is exactly once.

mosquitto_pub --repeat *<n>* **--repeat-delay** *<s>* **-t** *<topic>* **-m** *<msg>*
Publish repeatedly

SUBSCRIBE

mosquitto_sub -t *<topic>*
Print each payload

SUBSCRIBE (cont.)

mosquitto_sub -v -t *<topic>*
Prefix payloads with the topic

mosquitto_sub -t *<a>* **-t** *<b>*
Subscribe to several filters

mosquitto_sub -q *<0|1|2>* **-t** *<topic>*
Set subscription QoS

mosquitto_sub -C *<n>* **-t** *<topic>*
Exit after n messages

mosquitto_sub -W *<s>* **-t** *<topic>*
Exit after s seconds
Exit status 27 means it timed out waiting.

mosquitto_sub -R -t *<topic>*
Skip retained messages

mosquitto_sub --retained-only -t *<topic>*
Only retained messages, then exit

mosquitto_sub -T *<topic>* **-t** *<filter>*
Hide matching topics from output
Unsubscribing with -U cannot punch holes in a wildcard.

mosquitto_sub -F "%t %p" -t *<topic>*
Custom output format
%t topic, %p payload, %x hex payload, %r retain flag, %q QoS, %l length, @Y-@m-@d... timestamps.

mosquitto_sub -E -t *<topic>*
Exit once subscribed

mosquitto_sub -t *<topic>* **--remove-retained --retained-only**
Delete retained messages
Run it on a narrow filter first; '#' clears everything.

CONNECTION & AUTH

-h *<host>* **-p** *<port>*
Broker address (localhost:1883)

-u *<user>* **-P** *<password>*
Username and password
-P shows in the process list and shell history.

CONNECTION & AUTH (cont.)

-i *<id>*
Fixed client ID
Two clients with the same ID kick each other off.

-I *<prefix>*
Client ID prefix plus PID

-L mqtt://<u>*<url>*</u>
Everything in one URL
Form: mqtt://user:pw@host:port/topic.

-V *<5|311|31>*
MQTT protocol version
Default is 3.1.1.

-c -i *<id>*
Persistent session
Needs -i; the broker queues QoS 1/2 messages while you are offline.

-k *<seconds>*
Keepalive interval

--ws
Connect over WebSockets

-d
Print protocol debug output

--will-topic *<topic>* **--will-payload** *<msg>*
Last will on unexpected disconnect
Add --will-retain and --will-qos as needed.

TLS

--cafile *<ca.pem>*
Trust this CA certificate
Port 8883 turns TLS on automatically.

--cert *<cert.pem>* **--key** *<key.pem>*
Client certificate login

--capath *<dir>*
Directory of trusted CAs

--tls-use-os-certs
Trust the system CA store

--insecure
Skip hostname check, testing only

--tls-version *<tlsv1.2|tlsv1.3>*
Pin the TLS version

TLS (cont.)

--psk *<hex>* **--psk-identity** *<id>*
Pre-shared key instead of certificates

USERS & PASSWORDS

mosquitto_passwd -c *<file>* *<user>*
Create file with a user
-c overwrites an existing file.

mosquitto_passwd *<file>* *<user>*
Add a user or change password

mosquitto_passwd -b *<file>* *<user>* *<password>*
Set password non-interactively
Password lands in shell history.

mosquitto_passwd -D *<file>* *<user>*
Delete a user

mosquitto_passwd -U *<file>*
Hash a plain-text password file
Never run it on an already hashed file.

mosquitto_passwd -H sha512 *<file>* *<user>*
Hash for Mosquitto 1.6 and older

Point password_file at the file in mosquitto.conf and set allow_anonymous false; the broker needs a reload (SIGHUP) for changes.

RECIPES

Watch broker status

```
mosquitto_sub -v -t '$SYS/broker/#'
```

Authenticated TLS publish

```
mosquitto_pub -h <host> -p 8883 --cafile ca.pem \
  -u <user> -P <pw> -t home/door -m open
```

Capture one reading

```
mosquitto_sub -t sensors/+/temp -C 1 -W 10
```

Retained status with last will

```
mosquitto_pub -r -t dev/<id>/status -m online \
  --will-topic dev/<id>/status --will-payload offline --will-retain
```