MQTT-TLS Profile of ACE

draft-sengul-ace-mqtt-tls-profile-03

Cigdem Sengul, Nominet

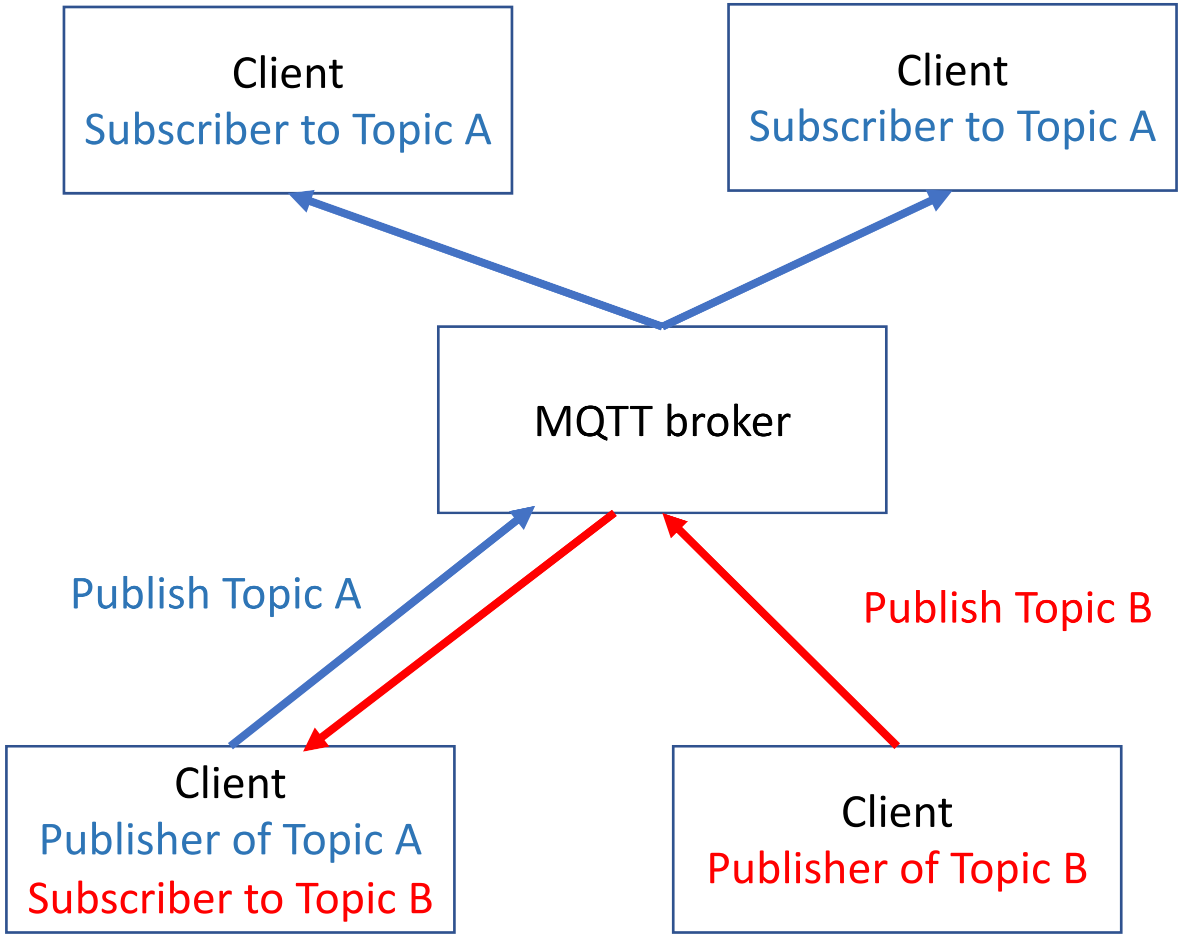
Anthony Kirby, Nominet,

Paul Fremantle, University of Portsmouth

IETF 104 ACE WG meeting

March 29, 2019

MQTT recap



- MQTT is a publish/subscribe protocol, with a broker managing the data exchange
- Runs over TCP and supports TLS
- Sample messages
  - CONNECT: First message an MQTT client sends to the broker
  - PUBLISH: Can be sent by a publisher or broker
  - SUBSCRIBE
  - PINGREQUEST: Keep alive from clients
- Topic subscriptions: Topic Filter (including wild cards) + QoS (Quality of Service)

MQTT-TLS profile version updates

- draft-sengul-ace-mqtt-tls-profile-00: Basic: MQTT v3.1 – OASIS standard
- draft-sengul-ace-mqtt-tls-profile-01: Extended: MQTT v5 – Candidate OASIS standard
- draft-sengul-ace-mqtt-tls-profile-02: Added PINGREQ support
  - Response to review from Dominik Obermaier (enterprise MQTT, and participant from Oasis MQTT Workgroup)
- draft-sengul-ace-mqtt-tls-profile-03: Clarified token structure and encoding, added privacy and security considerations
  - Response to review from Ludwig Seitz

Profile checklist

Profile identifier	mqtt_tls
Communication/security protocol	MQTT-TLS
AS discovery	~~Not supported~~ Can be supported by MQTT v5
Client & RS mutual authentication	RS: certificate in TLS handshake Client: Token and MAC in MQTT Connect message + Several methods for token transport and verification enabled with MQTT v5
PoP protocols	Symmetric/asymmetric
Token transport	MQTT Connect message + Several methods for token transport and verification enabled with MQTT v5
Token introspection	/instrospect (HTTPS)
Token request	/token (HTTPS)
/authz-info	May be supported (draft Appendix B)

Passing tokens to the broker

The connection between client and broker is secured by TLS.

- After TLS session set-up:
  - Token is transported in the CONNECT message –different methods may be supported for MQTT v3.1 and v5

- Before TLS session set-up:

  Raw Public Keys (RPK) and Preshared Keys (PSK) modes

  Token may need to be published to “authz-info” topic unauthorized (Described in Appendix B)

Token in CONNECT

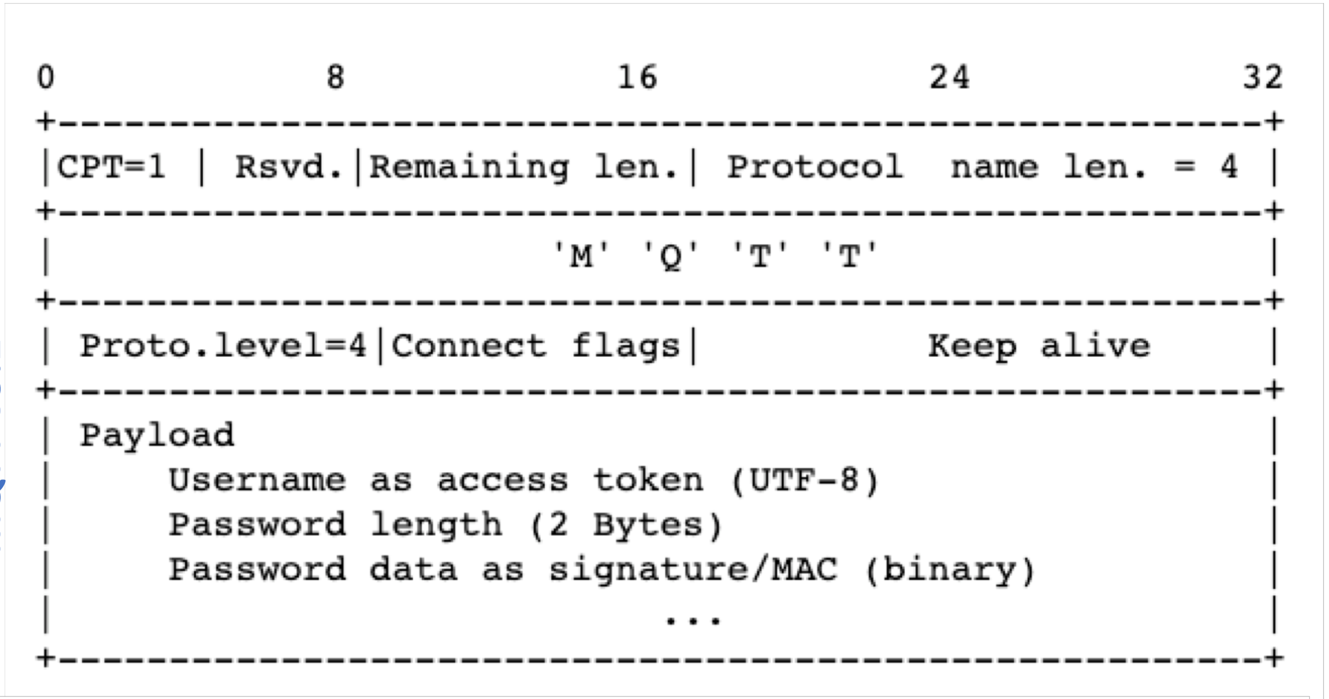
Method 1 (MQTT v3.1)

- Default auth method: ace_mqtt_tls (not in packet)
- Username: Access token
- Password: Signature/MAC for PoP

Method 2 (MQTT v5)

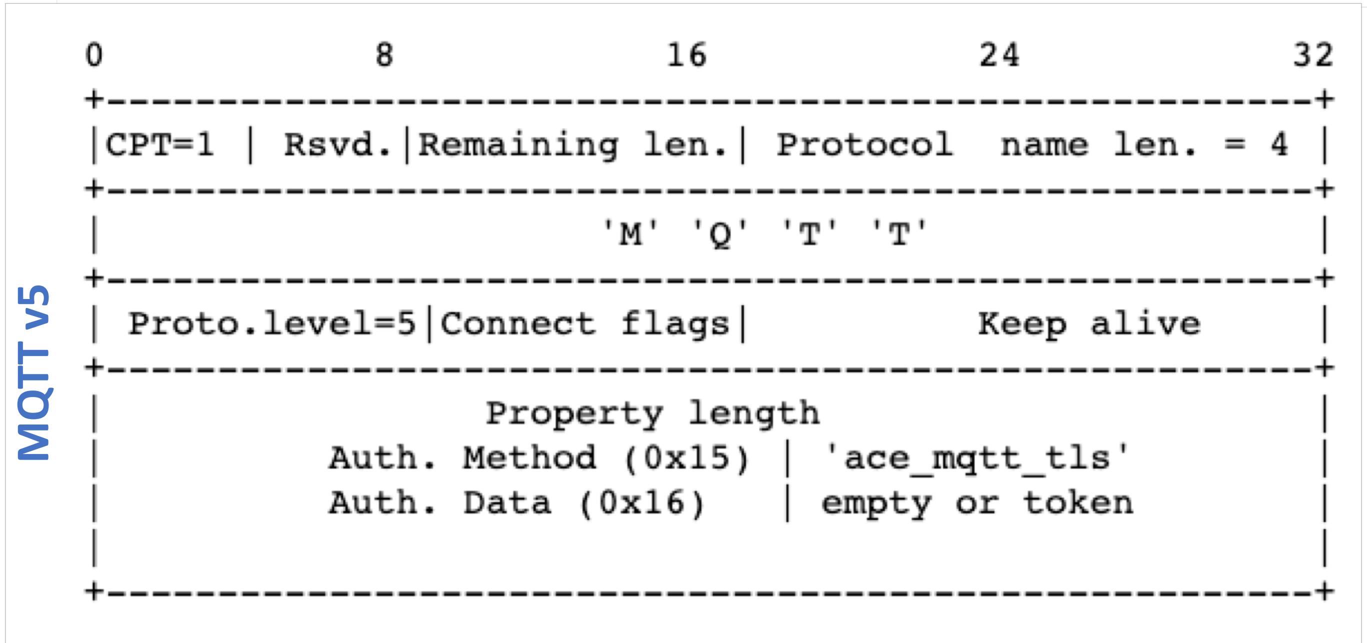
- Auth method: ace_mqtt_tls
- Auth data: empty or token(+mac)
- If empty: AS discovery
- Else token verification:
- If Token + signature/MAC, just verify
- If only Token, use the challenge protocol

MQTT v3.1



```
0               8               16              24              32
+-------------------------------------------------------------+
|CPT=1 | Rsvd.|Remaining len.| Protocol  name len. = 4 |
+-------------------------------------------------------------+
|                       'M'  'Q'  'T'  'T'                    |
+-------------------------------------------------------------+
| Proto.level=4|Connect flags|          Keep alive            |
+-------------------------------------------------------------+
| Payload                                                     |
|     Username as access token (UTF-8)                        |
|     Password length (2 Bytes)                               |
|     Password data as signature/MAC (binary)                 |
|                           ...                               |
+-------------------------------------------------------------+
```

MQTT v5



```
0               8               16              24              32
+-------------------------------------------------------------+
|CPT=1 | Rsvd.|Remaining len.| Protocol  name len. = 4 |
+-------------------------------------------------------------+
|                       'M'  'Q'  'T'  'T'                    |
+-------------------------------------------------------------+
| Proto.level=5|Connect flags|          Keep alive            |
+-------------------------------------------------------------+
|                    Property length                          |
|         Auth. Method (0x15) | 'ace_mqtt_tls'                |
|         Auth. Data (0x16)   | empty or token                |
|                                                             |
+-------------------------------------------------------------+
```

Error handling

MQTT v 3.1

- On token expiry, kill the connection as server disconnect not possible
- Better than silently failing, because there is no other way to tell the client it has to renew its token

MQTT v5

- On token expiry, send DISCONNECT message with error code ‘Not Authorized’
- If QoS >= 1, then PUBACK/SUBACK messages can return error ’Not Authorized’
- AUTH packet for ‘Re-authentication’ avoiding disconnection.

Summary

- Draft covers both versions of MQTT
- A single AS controls who can publish/subscribe to a certain topic
- Broker is the trusted party, ends the TLS connections to pub/sub clients
  - Topic and QoS are the only data needed by the broker to dispatch messages
  - May be extended to support draft-ietf-ace-key-groupcomm for payload protection