MQTT is a messaging protocol based on publish-subscribe pattern. It works on top of the TCP/IP protocol and is used in IoT.

This implementation of MQTT is from eclipse.org. Platform specific initializations and functions are provided under *platform/* directory. The APIs under *platform/* provide MQTT initialization and connect functionalities for connection over TCP, SSL (TLS) and Websocket.

# Features and Limitations

Here is the list of salient features of this MQTT implementation:

1. Supports MQTT v3.1
2. MQTT Subscribe and Publish are supported
3. Supports Publish with QOS 2
4. Supports MQTT keepalive
5. Supports Secured MQTT (MQTT over TLS connection)
6. Supports MQTT over Websockets (Both Secured and Non-secured)

# Header file/s

1. Components/mqtt/include/ MQTTClient.h.
2. Components/mqtt/platform/ mqtt\_nw.h.

# API Reference

## MQTTNetworkInit

### Overview

This API needs to be called if the intended MQTT connection is over TCP. This initializes the connection handle passed to the API.

### Definition

|  |
| --- |
| void MQTTNetworkInit(MQTTNetwork\* handle) |

### Parameter

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *handle* | Pointer to connection handle data structure of type MQTTNetwork |

Table : MQTTNetworkInit - parameters

### Return

None.

## MQTTNetworkConnect

### Overview

This function is used to connect to the broker over TCP. This connection will be non-secured.

### Definition

|  |
| --- |
| int  MQTTNetworkConnect(MQTTNetwork\* handle, char\* addr, int port) |

### Parameters

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *handle* | Pointer to connection handle data structure of type MQTTNetwork that was passed to MQTTNetworkInit() |
| *addr* | DNS name of the Broker |
| *port* | Broker port number |

Table : MQTTNetworkConnect - parameters

### Return

Success: 0

Error: -1

## MQTTNetworkDisconnect

### Overview

This function disconnects the MQTT connection.

### Definition

|  |
| --- |
| void MQTTNetworkDisconnect(MQTTNetwork \*handle) |

### Parameters

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *handle* | Pointer to connection handle data structure of type MQTTNetwork that was passed to MQTTNetworkInit() |

Table : MQTTNetworkDisconnect - parameters

### Return

None.

## MQTTNetworkInit\_Tls

### Overview

This API needs to be called if the intended MQTT connection is over SSL(TLS). This initializes the connection handle passed to the API.

### Definition

|  |
| --- |
| void MQTTNetworkInit\_Tls(MQTTNetwork\* handle) |

### Parameters

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *handle* | Pointer to connection handle data structure of type MQTTNetwork |

Table : MQTTNetworkInit\_Tls – parameters

### Return

None.

## MQTTNetworkConnect\_Tls

### Overview

This function is used for connecting to Broker over SSL (TLS). This is a secured connection.

### Definition

|  |
| --- |
| int  MQTTNetworkConnect\_Tls(MQTTNetwork \*n, char \*host, int port,  ssl\_wrap\_cfg\_t \*cfg) |

### Parameters

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *handle* | Pointer to connection handle data structure of type MQTTNetwork that was passed to MQTTNetworkInit\_Tls() |
| *host* | DNS name of the Broker or the IP address |
| *port* | Broker port number |
| *cfg* | Pointer to data structure of type ssl\_wrap\_cfg\_t. This is used to pass the SSL related configurations |

Table : MQTTNetworkConnect\_Tls - parameters

### Return

Success: 0

Error: -1

## MQTTNetworkDisconnect\_Tls

### Overview

This function disconnects the MQTT connection done using MQTTNetworkConnect\_Tls.

### Definition

|  |
| --- |
| void MQTTNetworkDisconnect\_Tls(MQTTNetwork \*handle) |

### Parameters

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *handle* | Pointer to connection handle data structure of type MQTTNetwork that was passed to MQTTNetworkInit\_Tls() |

Table : MQTTNetworkDisconnect\_Tls - parameters

### Return

None.

## MQTTNetworkInit\_Ws

### Overview

This API needs to be called if the intended MQTT connection is over Websocket. This initializes the connection handle passed to the API.

### Definition

|  |
| --- |
| void MQTTNetworkInit\_Ws(MQTTNetwork\* handle) |

### Parameters

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *handle* | Pointer to connection handle data structure of type MQTTNetwork |

Table : MQTTNetworkInit\_Ws - parameters

### Return

None.

## MQTTNetworkConnect\_Ws

### Overview

This function is used for connecting to Broker over Websocket. The connection can be secured or non-secured.

### Definition

|  |
| --- |
| int MQTTNetworkConnect\_Ws(MQTTNetwork\* n, websock\_config\_t \* ws\_cfg) |

### Parameters

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *handle* | Pointer to connection handle data structure of type MQTTNetwork that was passed to MQTTNetworkInit\_Ws() |
| *ws* | Pointer to data structure of type websock\_config\_t. This is used to pass the Websocket related configurations |

### Return

Success: 0

Error: -1

## MQTTNetworkDisconnect\_Ws

### Overview

This API disconnects the MQTT connection established using MQTTNetworkConnect\_Ws.

### Definition

|  |
| --- |
| void MQTTNetworkDisconnect\_Tls(MQTTNetwork \*handle) |

### Parameters

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| *handle* | Pointer to connection handle data structure of type MQTTNetwork that was passed to MQTTNetworkInit\_Ws() |

Table : MQTTNetworkDisconnect\_Ws - parameters

### Return

None.

# Application Example

For the example code, refer: *examples/mqtt application*.