



## **TI CC32XX SIMPLELINK : MQTT PACKAGE**

---

### **Release Notes**

---

Sep 14, 2016

Table of Contents

1 Introduction ..... 3

2 Release Content ..... 3

3 Directory Structure ..... 3

4 License and Manifest ..... 3

5 Change Log..... 4

## 1 Introduction

This document describes the MQTT Package for use with the CC32xx SimpleLink Launchpad evaluation kit and CC32xx SimpleLink Software Development Kit (SDK).

## 2 Release Content

The package provides the source code for the server and client MQTT libraries for TI's SimpleLink WiFi platform. This revision of the MQTT package supports the CC3200 device. Example applications have been included in the package. For the contents included in the SDK, please refer to the section about the directory structure.

## 3 Directory Structure

Directory Path	Contents Description
cc3200-sdk/netapps/mqtt	<ul style="list-style-type: none"><li>Source code for the MQTT server and client along with platform &amp; network adaptors to work the CC3200 SimpleLink SDK</li></ul>
cc3200-sdk/example/mqtt_client	<ul style="list-style-type: none"><li>Sample application to illustrate MQTT client functionalities</li></ul>
cc3200-sdk/example/mqtt_server	<ul style="list-style-type: none"><li>Sample application to illustrate MQTT server + client features</li></ul>

## 4 License and Manifest

Please see the accompanying documents “CC3200-MQTT\_license.pdf” for the licensing terms and the “CC3200-MQTT\_manifest.html” file for the manifest.

## 5 Change Log

#	MQTT PKG Version	Date of Release	Comments
1	mqtt_client: 1.4.0 mqtt_server: 1.3.0 mqtt_common: 1.5.0	14 Sep 2016	<b>Updates in the netapps/mqtt</b> <ul style="list-style-type: none"> <li>mqtt_client.c: mqtt_client_lib_init(), additional re-initialization of the global parameters.</li> <li>mqtt_common.h: macros MQTT_MALLOC and MQTT_FREE introduced. These macros are being used by the following files: server_util.c, cc32xx_sl_net.c and server_core.c</li> <li>sl_mqtt_server.h: introduced sl_ExtLib_MqttServerActivate(void) and sl_ExtLib_MqttServerPause(void) API(s)</li> <li>cc32xx_sl_net.c: incorporated compile time macros to interwork with both Gen 1 and Gen 2 version of SL API(s)</li> <li>server_pkts.h: introduced mqtt_server_pause(void) and mqtt_server_pause_locked(void) API(s)</li> <li>server_pkts.c: implementation of mqtt_server_pause(void) and mqtt_server_pause_locked(void) API(s) and updates to post message to the loopback port to relinquish NWP resources</li> <li>server_util.h: removed macros my_malloc and my_free</li> <li>sl_mqtt_client.c: Default value of BUF_LEN updated to 2048 bytes from 1024 bytes and corresponding check introduced in the sl_ExtLib_MqttClientSend( ) API</li> <li>sl_mqtt_server.c: implementation of sl_ExtLib_MqttServerActivate(void) and sl_ExtLib_MqttServerPause(void) API(s)</li> <li>docs/: updated revision have been made available</li> </ul>
2	mqtt_client: 1.2.2 mqtt_server: 1.1.2	27 June 2016	File name: mqtt_client.c ----- 1. mqp_free => mqp_free_locked, added critical section for freeing up the 'mqp' packet memory 2. Added new error code "MQP_ERR_REMLSTN" to indicate failure to open the socket 3. In function "connect_msg_send", avoided double freeing up of packet on successful send

		<p>4. In function "mqp_setup_proc_data_rcv", initializing member (rx_mqp) of data structure.</p> <p>5. In cl_ctx_rcv function, "single_ctx_rx_prep" =&gt; "single_ctx_rx_prep_locked", added critical section for ping request message</p> <p>6. net_connect(): returning a new error code, if client is not able to open socket connection to the server</p> <p>7. Added compile time flag 'MQTT_PART_TX_XPORT_EXPERIMENTAL' to highlight that the MQTT Client LIB does not support segmented transmission of the MQTT packets over the network. The flag MQTT_PART_TX_XPORT_EXPERIMENTAL is disabled by default.</p> <p>8. tx_part_addup(): Added an assert check (infinite for loop) to highlight that the MQTT Client LIB does not support segmented transmission of the MQTT packets over the network.</p> <p>File name: mqtt_client.h</p> <p>-----</p> <p>1. mqtt_client_send_progress(): incorporated the routine under the compile time option / flag MQTT_PART_TX_XPORT_EXPERIMENTAL to indicate that the Client LIB does not support segmented transmission of the MQTT packets over the network.</p> <p>File name: server_pkts.c</p> <p>-----</p> <p>1. In function ka_sequence(), fix for paho compliance test "keep alive test"</p> <p>2. Support of data structures to enable secure connections from the clients.</p> <p>Filename: sl_mqtt_client.c</p> <p>-----</p> <p>1. Fixed synchronisation issues between Application Task and SL MQTT Task</p> <p>2. Clean-up will registration</p> <p>3. Return error codes to applications</p> <p>Filename: sl_mqtt_server.c</p> <p>-----</p> <p>1. Added to initialization parameters of mqtt_server_lib_cfg</p> <p>Filename: client_mgmt.c</p> <p>-----</p> <p>1. Updates to ensure that there is no overflow of the storage beyond the number MAX_PUBREL_INFLT messages for the outgoing QOS-2 PUBLISH and its follow-up messages.</p> <p>2. Session resume(): bug fix and updated the order of</p>
--	--	--

			<p>dispatch to (re-send) PUBREL (ack2) messages, (re-send) PUBLISH (to get ack1 i.e. PUBREC / PUBACK) messages and PUBLISH messages (not sent as yet).</p> <p>3. cl_mgmt_qos2_pub_rx_update(): Updates to forward QOS2 packet "exactly once"</p> <p>Filename: server_core.c</p> <p>-----</p> <p>1. SUB_node_create(): Fixed the upkeep of the count of the subscription topics made by each client</p> <p>2. node_data_update(): Updated to ensure that retained message for a topic is deleted, when a client sends message having zero length and retain flag as true - bug detected whilst running the paho compliance test for "retained messages"</p> <p>3. _proc_pub_msg_rx(): Updated code to store messages that have QOS 0 and retain flag 1</p> <p>4. on_connack_send(): updated to ensure the client state is updated prior to topic management</p> <p>5. _proc_pub_msg_rx(): Updates to forward QOS2 packet "exactly once".</p> <p>Filename: server_pkts.h</p> <p>-----</p> <p>1. Add "nw_security" field in the struct mqtt_server_lib_cfg</p> <p>Filename: mqtt_common.c</p> <p>-----</p> <p>1. initialization of new members of the struct secure_conn</p> <p>Filename: mqtt_common.h</p> <p>-----</p> <p>1. Added new error code MQP_ERR_REMLEN to indicate that no remote server is listening</p> <p>2. Added new members remote_dn_str (remote domain name string) and dev_listed_ca in the struct secure_conn</p> <p>Filename: sl_mqtt_client.h</p> <p>-----</p> <p>1. Streamlined type for the qos parameter across various API(s)</p> <p>2. Added security related parameters to SIMqttServer_t</p> <p>3. Introduced error codes</p> <p>Filename: sl_mqtt_server.h</p> <p>-----</p> <p>1. Streamlined values of SL_MQTT_SRVR_NETCONN macros client and server</p>
--	--	--	--

			<p>Renamed following:</p> <ol style="list-style-type: none"><li>1. platform\cc31xx_sl_net.c =&gt; platform\cc32xx_sl_net.c</li><li>2. platform\cc31xx_sl_net.h =&gt; platform\cc32xx_sl_net.h</li><li>3. platform\cc3200_platform.c =&gt; platform\cc32xx_platform.c</li></ol> <p>Paho Tests (snapshot, 13 June 2016, master branch)</p> <p>-----</p> <p>Ran paho tests for both the server and client libs and have made relevant updates to the libraries.</p>
--	--	--	--