Google Summer of Code 2014

MQTT-SN implementation for Wiselib

Functional requirements

Note: Below requirements are based and consistent with MQTT-SN specification available here: http://www.mqtt.org/new/wp-content/uploads/2009/06/MQTT-SN_spec_v1.2.pdf

Gateway Advertisement and Discovery	
REQ 1	If client receives GWINFO or ADVERTISE message with gateway id, which is not known, gateway information is saved.
REQ 2	Client always choose a gateway with the strongest signal.
REQ 3	If client receives ADVERTISE and saves gateway information (or gateway information is already known because of previously receive of GWINFO message), it must start a process of gateway monitoring and check if next ADVERTISE message will come after time indicated by 'duration' field. If it does not receive ADVERTISE from certain gateway after N_ADV consecutive times, client removes it from a list of active gateways.
REQ 4	If no gateway is known by client, it starts process of searching gateway and broadcast SEARCHGW messages. To prevent broadcast storms, the sending of SEARCHGW message is delayed by a random time between 0 and T_SEARCHGW and increased twice after each send.
REQ 5	If clients receives SEARCHGW message sent by another client during searching gateway process, it behaves like message was sent by itself.
REQ 6	If client receives SEARCHGW message sent by another client and has at least one active gateway on its list, it sends GWINFO message to sender after T_GWINFO delay. If client will receive GWINFO before T_GWINFO interval, it will cancel a transmission.

Client's Connection Setup	
REQ 7	To setup a connection, client sends CONNECT message to a gateway. Client can send CONNECT message with or without setting 'Will flag'.
REQ 8	Client responds with WILLTOPIC message for WILLTOPICREQ message.
REQ 9	Client responds with WILLMSG message for WILLMSGREQ message.
REQ 10	If client receives CONACK message with 'return code' equal to: ACCEPTED it assumes that connection with gateway is established.

REQ 11	If client receives CONACK message with ‘return code’ equal to: REJECTED CONGESTION it resends CONNECT message after T_WAIT interval.

Procedure of updating Will data	
REQ 12	Client can update its Will topic by sending WILLTOPICUPD message.
REQ 13	Client can update its Will message by sending WILLMSGUPD message.
REQ 14	Client can delete its Will data (topic and message) by sending empty WILLTOPICUPD message.

Topic Name Registration Procedure	
REQ 15	If client is connected with a gateway, it can register a topic by sending REGISTER message.
REQ 16	If clients receives REGACK message with 'return code' equal to: ACCEPTED it assumes that topic is successfully registered and data can published.
REQ 17	If client receives REGACK message with ‘return code’ equal to: REJECTED CONGESTION it resends REGACK message after T_WAIT interval.
REQ 18	At any point in time a client may have only one REGISTER message outstanding, it has to wait for a REGACK message before it can register another topic name

Publish Procedure	
REQ 19	If client has successfully registered a topic name and is connected with a gateway, it can send PUBLISH messages for that topic
REQ 20	If clients receives PUBACK message with 'return code' equal to: ACCEPTED it assumes that PUBLISH message has been successfully sent to a gateway
REQ 21	If client receives REGACK message with ‘return code’ equal to: REJECTED CONGESTION it resends PUBLISH message after T_WAIT interval.
REQ 22	At any point in time a client may have only one PUBLISH message outstanding, it has to

	wait for a PUBACK message before it can send another PUBLISH message
REQ 23	Client is able to register a global callback for servicing data provided in PUBLISH message. This callback will be used for all received PUBLISH messages if there will not be registered any callback for specific topic
REQ 24	Client is able to unregister a global callback
REQ 25	Client is able to register a callback for servicing data prived in PUBLISH message which is connected with specific topic. This type of callback has higher priority then a global callback
REQ 26	Client is able to unregister a callback provided for specific topic

Topic Subscribe/Unsubscribe Procedure	
REQ 27	If client is connected with a gateway, it can send SUBSCRIBE messages in purpose of receiving PUBLISH messages which are connected with particular topic
REQ 28	If clients receives SUBACK message with 'return code' equal to: ACCEPTED it assumes that SUBSCRIBE message has been successfully sent to a gateway
REQ 29	If client receives SUBACK message with ‘return code’ equal to: REJECTED CONGESTION it resends SUBSCRIBE message after T_WAIT interval.
REQ 30	At any point in time a client may have only one SUBSCRIBE message outstanding, it has to wait for a SUBACK message before it can send another SUBSCRIBE message
REQ 31	If client is connected with a gateway and has successfully subscribed for certain topic, it can send UNSUBSCRIBE messages in purpose of stopping receiving PUBLISH messages connected with that topic.
REQ 32	If clients receives UNSUBACK message with 'return code' equal to: ACCEPTED it assumes that SUBSCRIBE message has been successfully sent to a gateway
REQ 33	If client receives UNSUBACK message with ‘return code’ equal to: REJECTED CONGESTION it resends SUBSCRIBE message after T_WAIT interval.
REQ 34	At any point in time a client may have only one UNSUBSCRIBE message outstanding, it has to wait for a UNSUBACK message before it can send another UNSUBSCRIBE message

<table>
<tr><th colspan="2">Keep alive and PING Procedure</th></tr>
<tr><td>REQ 35</td><td>Client should send a PINGREQ message within each time period indicated by the field ‘duration’ in CONNECT message</td></tr>
<tr><td>REQ 35</td><td>Client should respond with a PINGRESP message when it receives a PINGREQ message form the gateway which it is connected</td></tr>
<tr><td>REQ 37</td><td>If client will not receive PINGRESO from certain gateway after multiple retransmissions it should first try to connect to another gateway before trying to re-connect to this gateway</td></tr>
</table>