|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_001 |
| **Test Objective** | The IUT MUST close the network connection if fixed header flags in CONNECT Control Packet are invalid |
| **Reference** | [MQTT-2.2.2-1], [MQTT-2.2.2-2], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '1111'B;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_002 |
| **Test Objective** | If the Protocol Name is incorrect the IUT MAY disconnect the Client or it MAY continue processing the CONNECT packet. |
| **Reference** | [MQTT-3.1.2-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "TTQM",  protocol\_level indicating value 0x04;  }  then {  // TODO: response is not clearly defined  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_003 |
| **Test Objective** | The IUT MUST respond to Protocol Levels which it supports (in scope: MQTT-3.1.1) with return code 0x00 |
| **Reference** | [MQTT-3.1.2-2], [MQTT-3.1.4-4] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04;  }  then {  the IUT sends a CONNACK message containing  connect\_return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_004 |
| **Test Objective** | The IUT MUST validate that the reserved flag in the CONNECT Control Packet is set to zero and disconnect the Client if it is not zero |
| **Reference** | [MQTT-3.1.2-3], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  reserved\_field indicating value '1'B;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_005 |
| **Test Objective** | If the Will Flag is set to 1, the Will QoS and Will Retain fields in the Connect Flags will be used by the IUT, and the Will Topic and Will Message fields MUST be present in the payload. |
| **Reference** | [MQTT-3.1.2-9], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  will\_flag indicating value '1'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  will\_topic indicating value omit,  will\_message indicating value omit;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_006 |
| **Test Objective** | If the Will Flag is set to 0 the Will QoS and Will retain fields in the Connect Flags MUST be set to zero and the Will Topic and Will Message fields MUST NOT be present in the payload. |
| **Reference** | [MQTT-3.1.2-11], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  will\_flag indicating value '0'B,  will\_qos indicating value 1,  will\_retain indicating value '1'B,  reserved\_field indicating value '0'B;  ,  payload containing  // will\_topic indicating value "$WILL\_TOPIC",  // will\_message indicating value "$WILL\_MESSAGE";  will\_topic corresponding to WILL\_TOPIC,  will\_message corresponding to WILL\_MESSAGE;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_007 |
| **Test Objective** | If the Will Flag is set to 0, then the Will QoS MUST be set to 0 |
| **Reference** | [MQTT-3.1.2-13], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  will\_flag indicating value '0'B,  will\_qos indicating value 1,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_008 |
| **Test Objective** | If the Will Flag is set to 1, the value of Will QoS can be 0, 1 or 2. It MUST NOT be 3 |
| **Reference** | [MQTT-3.1.2-14], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  will\_flag indicating value '1'B,  will\_qos indicating value 3,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_009 |
| **Test Objective** | If the Will Flag is set to 1, the value of Will QoS can be 0, 1 or 2. |
| **Reference** | [MQTT-3.1.2-14], [MQTT-3.1.4-4] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  will\_flag indicating value '1'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ;  }  then {  the IUT sends a CONNACK message containing  connect\_return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_010 |
| **Test Objective** | If the Will Flag is set to 0, then the Will Retain Flag MUST be set to 0 |
| **Reference** | [MQTT-3.1.2-15], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '1'B,  reserved\_field indicating value '0'B;  ,  payload containing  will\_topic indicating value omit,  will\_message indicating value omit;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_011 |
| **Test Objective** | If the Will Flag is set to 0, then the Will Retain Flag MUST be set to 0 |
| **Reference** | [MQTT-3.1.2-15], [MQTT-3.1.4-4] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  will\_topic indicating value omit,  will\_message indicating value omit;  ;  }  then {  the IUT sends a CONNACK message containing  connect\_return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_012 |
| **Test Objective** | If the User Name Flag is set to 0, the Password Flag MUST be set to 0 |
| **Reference** | [MQTT-3.1.2-22], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  user\_name\_flag indicating value '0'B,  password\_flag indicating value '1'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_013 |
| **Test Objective** | If the user name flag is set to 0, a user name MUST NOT be present in the payload |
| **Reference** | [MQTT-3.1.2-18], [MQTT-3.1.2-22], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  user\_name corresponding to MQTT\_USER\_NAME,  password indicating value omit;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_014 |
| **Test Objective** | If the User Name Flag is set to 1, a user name MUST be present in the payload |
| **Reference** | [MQTT-3.1.2-19], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  user\_name\_flag indicating value '1'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  user\_name indicating value omit,  password indicating value omit;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_015 |
| **Test Objective** | If the Password Flag is set to 0, a password MUST NOT be present in the payload |
| **Reference** | [MQTT-3.1.2-20], [MQTT-3.1.2-22], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  user\_name indicating value omit,  password indicating value "$MQTT\_PASSWORD";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_016 |
| **Test Objective** | If the Password Flag is set to 1, a user password MUST be present in the payload |
| **Reference** | [MQTT-3.1.2-21], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  user\_name\_flag indicating value '1'B,  password\_flag indicating value '1'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  user\_name indicating value "$MQTT\_USER\_NAME",  password indicating value omit;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_017 |
| **Test Objective** | The IUT MUST allow ClientIDs which are between 1 and 23 UTF-8 encoded bytes in length |
| **Reference** | [MQTT-3.1.3-5] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '1'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$24\_BYTE\_CLIENT\_ID";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_018 |
| **Test Objective** | The IUT MUST allow ClientIDs that contain only alphanumeric characters [0-9a-zA-Z]. |
| **Reference** | [MQTT-3.1.3-5] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '1'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$INVALID\_CHAR\_CLIENT\_ID";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_019 |
| **Test Objective** | The IUT MAY allow a Client to supply a Client Identifier that has a length of zero bytes, however if it does so the IUT MUST treat this as a special case and assign a unique Client Identifier to that Client. |
| **Reference** | [MQTT-3.1.3-6], [MQTT-3.1.3-7], [MQTT-3.1.4-4] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '1'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$ZERO\_BYTE\_CLIENT\_ID";  ;  }  then {  // TODO: Standards says: MAY allow  the IUT sends a CONNACK message containing  connect\_return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_020 |
| **Test Objective** | The ClientId MUST be a well-formed UTF-8 encoded String. |
| **Reference** | [MQTT-1.5.3-1], [MQTT-3.1.3-4], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "U+D800";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_021 |
| **Test Objective** | The ClientId MUST be a UTF-8 encoded String. |
| **Reference** | [MQTT-1.5.3-2], [MQTT-3.1.3-4], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "U+0000";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_022 |
| **Test Objective** | The Will Topic MUST be a UTF-8 encoded string |
| **Reference** | [MQTT-1.5.3-1], [MQTT-3.1.3-10], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '1'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  will\_topic indicating value "U+D800",  will\_message indicating value "$WILL\_MESSAGE";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_023 |
| **Test Objective** | The Will Topic MUST be a UTF-8 encoded string |
| **Reference** | [MQTT-1.5.3-2], [MQTT-3.1.3-10], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '1'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "U+0000",  will\_message indicating value "$WILL\_MESSAGE";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_024 |
| **Test Objective** | The User Name MUST be a UTF-8 encoded string. |
| **Reference** | [MQTT-1.5.3-1], [MQTT-3.1.3-11], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '1'B,  password\_flag indicating value '1'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  user\_name indicating value "U+D800 U+0000 U+D8FFF",  password indicating value "$MQTT\_PASSWORD";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNECT\_025 |
| **Test Objective** | The User Name MUST be a UTF-8 encoded string. |
| **Reference** | [MQTT-1.5.3-2], [MQTT-3.1.3-11], [MQTT-3.1.4-1], [MQTT-3.2.2-6] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '1'B,  password\_flag indicating value '1'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  user\_name indicating value "U+0000",  password indicating value "$MQTT\_PASSWORD";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNACK\_001 |
| **Test Objective** | The IUT accepts only valid fixed header flags for CONNECT Control Packet and responds with CONNACK |
| **Reference** | [MQTT-2.2.2-1], [MQTT-3.1.4-4] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ;  }  then {  the IUT sends a CONNACK message containing  header\_flags indicating value '0000'B,  connect\_return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNACK\_002 |
| **Test Objective** | If the IUT accepts a connection with CleanSession set to 1, the IUT MUST set SessionPresent to 0 in the CONNACK packet in addition to setting a zero-return code. |
| **Reference** | [MQTT-3.2.2-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '1'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$CLIENT\_ID";  ;  }  then {  the IUT sends a CONNACK message containing  header\_flags indicating value '0000'B,  session\_present\_flag indicating value '0'B,  connect\_return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNACK\_003 |
| **Test Objective** | If the IUT has stored session state, it MUST set session present flag to 1 in the CONNACK control packet. |
| **Reference** | [MQTT-3.2.2-2] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the IUT having a present session for the CLIENT\_ID  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$CLIENT\_ID";  ;  }  then {  the IUT sends a CONNACK message containing  header\_flags indicating value '0000'B,  session\_present\_flag indicating value '1'B,  connect\_return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNACK\_004 |
| **Test Objective** | If the IUT does not have a stored session state, it MUST set session present flag to 0 in the CONNACK control packet. |
| **Reference** | [MQTT-3.2.2-3] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the IUT having no present session for the CLIENT\_ID  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$CLIENT\_ID";  ;  }  then {  the IUT sends a CONNACK message containing  header\_flags indicating value '0000'B,  session\_present\_flag indicating value '0'B,  connect\_return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNACK\_005 |
| **Test Objective** | The IUT MUST respond to Protocol Levels which it does not support (in scope: MQTT-3.1.1) with return code 0x01 |
| **Reference** | [MQTT-3.1.2-2], [MQTT-3.2.2-4], [MQTT-3.2.2-5] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x05,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$CLIENT\_ID";  ;  }  then {  the IUT sends a CONNACK message containing  header\_flags indicating value '0000'B,  session\_present\_flag indicating value '0'B,  connect\_return\_code indicating value 0x01;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNACK\_006 |
| **Test Objective** | If the Client supplies a zero-byte Client Identifier with Clean Session set to 0, the IUT MUST respond to the CONNECT Control Packet with a CONNACK return code 0x02 and then close the Network Connection. |
| **Reference** | [MQTT-3.1.3-8], [MQTT-3.1.3-9], [MQTT-3.2.2-4], [MQTT-3.2.2-5] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$ZERO\_BYTE\_CLIENT\_ID";  ;  }  then {  the IUT sends a CONNACK message containing  header\_flags indicating value '0000'B,  session\_present\_flag indicating value '0'B,  connect\_return\_code indicating value 0x02;  and the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNACK\_007 |
| **Test Objective** | The Network Connection has been made but the MQTT service is unavailable |
| **Reference** | [MQTT-3.2.2-4], [MQTT-3.2.2-5] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the IUT having no available service for the MQTT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '0'B,  password\_flag indicating value '0'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$CLIENT\_ID";  ;  }  then {  the IUT sends a CONNACK message containing  header\_flags indicating value '0000'B,  session\_present\_flag indicating value '0'B,  connect\_return\_code indicating value 0x03;  and the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_CONNACK\_008 |
| **Test Objective** | The data in the username or password is malformed |
| **Reference** | [MQTT-3.1.2-19], [MQTT-3.1.2-21], [MQTT-3.2.2-4], [MQTT-3.2.2-5] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
|  | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a CONNECT message containing  header\_flags indicating value '0000'B,  protocol\_name indicating value "MQTT",  protocol\_level indicating value 0x04,  connect\_flags containing  clean\_session indicating value '0'B,  user\_name\_flag indicating value '1'B,  password\_flag indicating value '1'B,  will\_flag indicating value '0'B,  will\_qos indicating value 0,  will\_retain indicating value '0'B,  reserved\_field indicating value '0'B;  ,  payload containing  client\_identifier indicating value "$CLIENT\_ID",  user\_name indicating value "$MQTT\_USER\_NAME\_INVALID",  password indicating value "$MQTT\_PASSWORD\_INVALID";  ;  }  then {  the IUT sends a CONNACK message containing  header\_flags indicating value '0000'B,  session\_present\_flag indicating value '0'B,  connect\_return\_code indicating value 0x04;  and the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBSCRIBE\_001 |
| **Test Objective** | The IUT closes the network connection if fixed header flags in SUBSCRIBE Control Packet are invalid |
| **Reference** | [MQTT-2.2.2-1], [MQTT-2.2.2-2], [MQTT-3.8.1-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '1101'B;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBSCRIBE\_002 |
| **Test Objective** | SUBSCRIBE Control Packet MUST contain a non-zero 16-bit Packet Identifier |
| **Reference** | [MQTT-2.3.1-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value 0,  payload containing  topic\_filter indicating value "$TOPIC\_FILTER",  requested\_qos indicating value 1;  ;  }  then {  // TODO: response/behavior is not described in the standard  // Note: Mosquitto answers here with SUBACK and packet\_id = 0  the IUT sends a SUBACK message containing  packet\_identifier indicating value 0;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBSCRIBE\_003 |
| **Test Objective** | The Topic Filters in a Subscribe packet payload MUST be UTF-8 encoded strings. This data MUST NOT include encodings of code points between U+D800 and U+DFFF |
| **Reference** | [MQTT-1.5.3-1], [MQTT-3.8.3-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "U+D800",  requested\_qos indicating value 1;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBSCRIBE\_004 |
| **Test Objective** | The Topic Filters in a Subscribe packet payload MUST be UTF-8 encoded strings. A UTF-8 encoded string MUST NOT include an encoding of the null character U+0000. |
| **Reference** | [MQTT-1.5.3-2], [MQTT-3.8.3-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "U+0000",  requested\_qos indicating value 1;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBSCRIBE\_005 |
| **Test Objective** | The payload of a SUBSCRIBE packet MUST contain at least one Topic Filter /QoS pair. |
| **Reference** | [MQTT-3.8.3-3], [MQTT-4.8.0-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  omit;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBSCRIBE\_006 |
| **Test Objective** | The upper 6 bits of the Requested QoS byte are reserved. Reserved bits for Requested QoS in the payload must be set to 0. |
| **Reference** | [MQTT-3.8.3-4] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "$TOPIC\_FILTER",  requested\_qos indicating value 0,  requested\_qos\_flags indicating value '111111'B;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBSCRIBE\_007 |
| **Test Objective** | The Requested QoS in the payload MUST be 0, 1 or 2 |
| **Reference** | [MQTT-3.8.3-4] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "$TOPIC\_FILTER",  requested\_qos indicating value 3;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBACK\_001 |
| **Test Objective** | The IUT accepts only valid fixed header flags for SUBSCRIBE Control Packet and responds with SUBACK |
| **Reference** | [MQTT-2.2.2-1], [MQTT-3.8.1-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B;  }  then {  the IUT sends a SUBACK message containing  header\_flags indicating value '0000'B;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBACK\_002 |
| **Test Objective** | SUBSCRIBE (in case where QoS > 0) Control Packet MUST contain a non-zero 16-bit Packet Identifier |
| **Reference** | [MQTT-2.3.1-1], [MQTT-2.3.1-7], [MQTT-3.8.4-1], [MQTT-3.8.4-2] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "$TOPIC\_FILTER",  requested\_qos indicating value 1;  ;  }  then {  the IUT sends a SUBACK message containing  header\_flags indicating value '0000'B,  packet\_identifier indicating value "$PACKET\_ID";  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBACK\_003 |
| **Test Objective** | SUBACK return codes other than 0x00, 0x01, 0x02 and 0x80 are reserved and MUST NOT be used. |
| **Reference** | [MQTT-3.9.3-1], [MQTT-3.9.3-2] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "$TOPIC\_FILTER",  requested\_qos indicating value 0;  ;  }  then {  the IUT sends a SUBACK message containing  header\_flags indicating value '0000'B,  packet\_identifier indicating value "$PACKET\_ID",  return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBACK\_004 |
| **Test Objective** | SUBACK return codes other than 0x00, 0x01, 0x02 and 0x80 are reserved and MUST NOT be used. |
| **Reference** | [MQTT-3.9.3-1], [MQTT-3.9.3-2] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "$TOPIC\_FILTER",  requested\_qos indicating value 1;  ;  }  then {  the IUT sends a SUBACK message containing  header\_flags indicating value '0000'B,  packet\_identifier indicating value "$PACKET\_ID",  return\_code indicating value 0x01;  // Note: if the IUT supports only QoS 0  or the IUT sends a SUBACK message containing  header\_flags indicating value '0000'B,  packet\_identifier indicating value "$PACKET\_ID",  return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBACK\_005 |
| **Test Objective** | SUBACK return codes other than 0x00, 0x01, 0x02 and 0x80 are reserved and MUST NOT be used. |
| **Reference** | [MQTT-3.9.3-1], [MQTT-3.9.3-2] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "$TOPIC\_FILTER",  requested\_qos indicating value 2;  ;  }  then {  the IUT sends a SUBACK message containing  header\_flags indicating value '0000'B,  packet\_identifier indicating value "$PACKET\_ID",  return\_code indicating value 0x02;  // Note: if the IUT supports only up to QoS 1  or the IUT sends a SUBACK message containing  header\_flags indicating value '0000'B,  packet\_identifier indicating value "$PACKET\_ID",  return\_code indicating value 0x01;  // Note: if the IUT supports only QoS 0  or the IUT sends a SUBACK message containing  header\_flags indicating value '0000'B,  packet\_identifier indicating value "$PACKET\_ID",  return\_code indicating value 0x00;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_SUBACK\_006 |
| **Test Objective** | SUBACK return codes other than 0x00, 0x01, 0x02 and 0x80 are reserved and MUST NOT be used. |
| **Reference** | [MQTT-3.9.3-1], [MQTT-3.9.3-2] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a SUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "$TOPIC\_FILTER\_INVALID",  requested\_qos indicating value 2;  ;  }  then {  // Note: Specification not clear: not explicitly stating what "failed subscription" is.  // see: https://github.com/erlio/vernemq/issues/72  // see: https://groups.google.com/forum/#!topic/mqtt/ppdmL4-v2Yk  the IUT sends a SUBACK message containing  header\_flags indicating value '0000'B,  packet\_identifier indicating value "$PACKET\_ID",  return\_code indicating value 0x80;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_UNSUBSCRIBE\_001 |
| **Test Objective** | The IUT closes the network connection if fixed header flags in UNSUBSCRIBE Control Packet are invalid |
| **Reference** | [MQTT-2.2.2-1], [MQTT-2.2.2-2], [MQTT-3.10.1-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a UNSUBSCRIBE message containing  header\_flags indicating value '1101'B;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_UNSUBSCRIBE\_002 |
| **Test Objective** | UNSUBSCRIBE Control Packet MUST contain a non-zero 16-bit Packet Identifier |
| **Reference** | [MQTT-2.3.1-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a UNSUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value 0;  }  then {  the IUT closes the TCP\_CONNECTION  // Note: response/behavior not explicitly described in the specification  or the IUT sends a UNSUBACK message containing  packet\_identifier indicating value 0;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_UNSUBSCRIBE\_003 |
| **Test Objective** | The Topic Filters in an Unsubscribe packet payload MUST be UTF-8 encoded strings. This data MUST NOT include encodings of code points between U+D800 and U+DFFF |
| **Reference** | [MQTT-1.5.3-1], [MQTT-3.10.3-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a UNSUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  topic\_filter indicating value "U+D800";  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_UNSUBSCRIBE\_004 |
| **Test Objective** | The payload of a UNSUBSCRIBE packet MUST contain at least one Topic Filter. |
| **Reference** | [MQTT-3.10.3-2], [MQTT-4.8.0-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a UNSUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID",  payload containing  omit;  ;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_UNSUBACK\_001 |
| **Test Objective** | The IUT accepts only valid fixed header flags for UNSUBSCRIBE Control Packet and responds with SUBACK |
| **Reference** | [MQTT-2.2.2-1], [MQTT-3.10.1-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a UNSUBSCRIBE message containing  header\_flags indicating value '0010'B;  }  then {  the IUT sends a UNSUBACK message containing  header\_flags indicating value '0000'B;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_UNSUBACK\_002 |
| **Test Objective** | The IUT MUST respond to an UNSUBSCRIBE request by sending an UNSUBACK control packet. The UNSUBACK control packet MUST have the same Packet Identifier as the UNSUBSCRIBE Control Packet. |
| **Reference** | [MQTT-3.10.4-4], [MQTT-3.10.4-5] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a UNSUBSCRIBE message containing  header\_flags indicating value '0010'B,  packet\_identifier indicating value "$PACKET\_ID";  }  then {  the IUT sends a UNSUBACK message containing  header\_flags indicating value '0000'B,  packet\_identifier indicating value "$PACKET\_ID";  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_PINGREQ\_001 |
| **Test Objective** | The IUT closes the network connection if fixed header flags in PINGREQ Control Packet are invalid |
| **Reference** | [MQTT-2.2.2-1], [MQTT-2.2.2-2] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a PINGREQ message containing  header\_flags indicating value '1111'B;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_PINGRESP\_001 |
| **Test Objective** | The IUT accepts only valid fixed header flags for PINGREQ Control Packet and responds with a PINGRESP |
| **Reference** | [MQTT-2.2.2-1], [MQTT-3.12.4-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a PINGREQ message containing  header\_flags indicating value '0000'B;  }  then {  the IUT sends a PINGRESP message containing  header\_flags indicating value '0000'B;  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_DISCONNECT\_001 |
| **Test Objective** | The IUT MUST validate that reserved bits are set to zero and disconnect the Client if they are not zero. |
| **Reference** | [MQTT-2.2.2-1], [MQTT-3.14.1-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a DISCONNECT message containing  header\_flags indicating value '0000'B;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |

|  |  |
| --- | --- |
| **TP Id** | TP\_MQTT\_Broker\_DISCONNECT\_002 |
| **Test Objective** | The IUT MUST validate that reserved bits are set to zero and disconnect the Client if they are not zero. |
| **Reference** | [MQTT-2.2.2-2], [MQTT-3.14.1-1] |
| **PICS Selection** | PIC\_Broker |
| **Initial Conditions** | |
| with {  the CLIENT having a "MQTT connection" to the IUT  } | |
| **Expected Behaviour** | |
| ensure that {  when {  the IUT receives a DISCONNECT message containing  header\_flags indicating value '1111'B;  }  then {  the IUT closes the TCP\_CONNECTION  }  } | |
| **Final Conditions** | |
|  | |