Test description

|  |  |
| --- | --- |
| **Test ID** | MIP4SLT1\_1132 |
| **Test Title** | Subscribe using a filter on a multiple topics, using ConcreteTopicExpression |
| **Execution Priority** | 1 |
| **Objective** | The objective is for the Consumer to subscribe to a multiple topics provided by the Provider. |
| **Scenario** | The Consumer is interested in receiving regular updates on multiple topics in an asynchronous way (i.e. without polling the Producer). |
| **Environment** | Internet or Co-located. Exchange Pattern: P/S. |
| **Participation** | 2 |
| **MTRS** | N/A |
| **Pre-test Conditions** | MIP4SLT1\_1211, MIP4SLT1\_1213, MIP4SLT1\_1215, MIP4SLT1\_1130  Prior to initiating exchange of MIP4.0 messages, each participant in the exchange will complete the *MIP4.0 Joining Questionnaire [REF-MIP-5]* and share this questionnaire with their exchange partners. The exchange of completed questionnaires will:   * Inform exchange partners of the capabilities of a partner system * Allow partners to exchange source identifiers * Aid in the early identification of potential issues   The exchange partners are expected to resolve any identified issues prior to the actual exchange of MIP4.0 messages.   |  |  | | --- | --- | | **ID** | **Rule** | | Exigence-1 | Prior to initiating MIP4.0 Information Exchange, the exchange partners shall complete and share the *MIP4.0 Joining Questionnaire*.[REF-MIP-5] |   Both Consumer and Producer systems conform to MIP4.0 Publish/Subscribe Exchange Pattern Specification.  Both Consumer and Producer are able to process and exchange information represented according to MIP4.0 Information Schemas.  Consumer successfully discovered the target Producer. The system on the Consumer side has been properly configured and is ready to interact with the Producer system. The Producer system broadcasts at least some of the topics requested. |
| **Test Inputs** | N/A. |
| **Conclusion** | The test is concluded after the Consumer is receiving regular asynchronous updates of the requested topics from the Producer. The data received in these updates is valid and complete. The subscription with the termination time will have timed out and the Consumer will have unsubscribed from the subscription without a specified duration. Once the Producer has successfully destroyed the subscriptions, the test is complete. |
| **Test Outputs** | N/A. |
| **Traceability** | REQ\_EM\_0008, REQ\_EM\_0010. |

Test Procedure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Action | Expected Result | Src | Dst |
|  | The Consumer subscribes to the Producer in order to advertise its interest in a subset of the data space.  rpw:GetResourceProperty  with parameters wsn:ConcreteTopicExpression, wsn:FixedTopicSet, wsn:TopicExpressionDialect and wstop:TopicSet. ~~ | The Producer receives a valid request for resource properties using ConcreteTopicExpression ~~ | 1 | 2 |
|  | Producer sends the topics requested to the Consumer ~~ | Consumer receives the requested topics. ~~ | 2 | 1 |
|  | The Consumer processes the list of exposed topics, prepares a filter set, and requests a subscription on multiple topics to the Producer using the wsn-b:Subscribe operation. Filter is specified using topic “contains” items.  The Consumer requests a subscription on multiple topics to the Producer using the wsn-b:Subscribe operation but specifies his filter using topic “does not contain” items.  The Consumer requests a subscription on multiple different topics to the Producer using the wsn-b:Subscribe operation and requests a termination time for the subscription. ~~ | Producer receives two subscription requests for multiple topics (one using topic “contains” and one using topic “does not contain”) and a third subscription request for multiple topics with a desired termination time. All use ConcreteTopicExpression. ~~ | 1 | 2 |
| 4. | Producer accepts or rejects the subscription requests.  Producer checks subscription filters and determines if data matches specified filters. If it does, the Producer sends a wsn-b:Notify to the address specified in the wsn-b:ConsumerReference for EACH matching subscription. ~~ | The Producer accepts the requests.  The Producer sends notification to the Consumer using ConcreteTopicExpression.  The Consumer processes the data contained in the messages using ConcreteTopicExpression. ~~ | 2 | 1 |
| 5. | Consumer renews subscription with termination time prior to expiration using wsn-b:Renew with the SubscriptionReference and a new expected termination time. ~~ | Producer successfully renews subscription to new termination time. ~~ | 1 | 2 |
| 6. | Producer confirms subscription renewed to Consumer.  Producer continuously sends notifications to the Consumer until the Consumer unsubscribes or the subscription expires.  Subscription with termination time expires ~~ | Subscription renewed to new termination time.  Producer sends notifications.  Producer raises ResourceUnknownFault according to WS-BaseNotification[Ref-6] ~~ | 2 | 1 |
| 7. | Consumer unsubscribes using wsn-b:Unsubscribe operation with the SubscriptionReference issued from the initial Subscribe responses. ~~ | Producer tries to destroy the corresponding subscriptions. If Producer cannot destroy the subscriptions it sends a fault message to the Consumer. ~~ | 1 | 2 |

**Configuration**

|  |  |  |
| --- | --- | --- |
| Item | Value | Comment |
| EventGeneration | 1 | 0 🡪 Combine steps to one event in the MTMT, generate new MTMT events on every source - destination change. 1 🡪 Every step will be added to the MTMT as a separate event. |