**3GPP TSG-SA WG6 Meeting #60 S6-241499**

**Changsha, China 15th -19th April 2024 (revision of S6-241191 was 0707)**

**Source: Convida Wireless, LLC**

**Title:** **Policy based multi-modal SEALDD flow establishment**

**Spec: 3GPP TR 23.700-23 V0.1.0**

**Agenda item: 8.5**

**Document for: Approval**

**Contact: Catalina Mladin,** [**mladin.catalina@convidawireless.com**](mailto:mladin.catalina@convidawireless.com)

**1. Introduction**

This contribution provides a solution for KI #2 - E2E Multi-Modal Communication Flows.

**2. Reason for Change**

SEALDD currently supports policy based SEALDD flow establishment functionality in clause 9.2.2.3 of TS 23.433. As shown in Figure 9.2.2.3-1, this fucntionality is limited to a SEALDD server establishing a single SEALDD flow for an individual application traffic flow between a VAL client and server.



Figure 9.2.2.3-1: Policy enforced by SEALDD server for connectivity

To effectively support multi-modal application traffic flows within SEALDD, SEALDD servers and clients require enhancements to support policy based multi-modal SEALDD flow establishment functionality. This contribution defines a policy based multi-modal SEALDD flow establishment procedure which enables multi-modal application traffic exhcange between multiple VAL clients and servers.

**3. Proposal**

It is proposed to agree the following changes to 3GPP 23.700-23 V0.2.0.

\*\*\* All new text \*\*\*

## 7.x Solution #X: Policy based Multi-modal SEALDD flow establishment

### 7.x.1 Architecture Impacts

This solution describes policy-based multi-modal SEALDD flow establishment functionality.

### 7.x.2 Solution description

This solution is targeting KI #2: E2E Multi-Modal Communication Flows and specifies a policy based multi-modal SEALDD flow establishment procedure.

NOTE 1: In the normative phase, this solution is to be harmonized with the corresponding solution for multi-modal aplication data delivery initiated by VAL server, as well as with clause 9.2.2.3 of TS 23.433.

NOTE 2: A corresponding procedure for deletion of a multi-modal SEALDD flow is to be defined during the normative phase.

#### 7.x.2.1 Multi-modal SEALDD flow establishment based on policy

Figure 7.x.2.1 illustrates the procedure for a SEALDD server to establish a multi-modal SEALDD flow based on a configured multi-modal SEALDD policy. The multi-modal SEALDD flow is established by the SEALDD server before multi-modal application communication between VAL clients and VAL servers starts.

Pre-conditions:

1. The SEALDD server has been configured with a multi-modal SEALDD policy.



Figure 7.x.2.1-1: Policy based multi-modal SEALDD flow establishment

1. The VAL servers subscribe to SEALDD event exposure for connection status using the procedure defined in TS 23.433 clause 9.2.2.6.

2. Based on the configured multi-modal SEALDD policy information, the SEALDD server determines the applicable VAL flows and triggers the establishment of required individual SEALDD connections as described in TS 23.433 clause 9.2.2.3. Next, the SEALDD server triggers the establishment of a multi-modal SEALDD flow and associates the individual SEALDD connections with it. To maintain this association, the SEALDD server assigns a multi-modal SEALDD flow identifier and associates it with the identifiers of each individual SEALDD connection.

3. During establishment of the multi-modal SEALDD flow, the SEALDD server interacts with the 3GPP CN to configure multi-modal QoS flow parameters based on the multi-modal SEALDD policy information . The SEALDD server utilizes the NEF/PCF/NRM/EES service for multi-modal QoS adjustment. Specifically, the SEALDD server relies on the northbound Policy Authorization Service API exposed by the PCF as specified in 3GPP TS 23.502 [6] and 3GPP TS 23.503 [7], if the SEALDD server is connected to the PCF via the N5 reference point, or the northbound AF Session with QoS Service API and/or the PFD Management northbound APIs exposed by the NEF as specified in 3GPP TS 23.502 [6] and 3GPP TS 23.503 [7], if the SEALDD server is connected to the PCF via NEF. SEALDD relies also upon the EES Session with QoS API as specified in 3GPP TS 23.558 [10] and/or the NRM QoS functionality as described in 3GPP TS 23.434 [4].

4. The SEALDD server allocates an IP address(s) and port(s) for sending and receiving packet over the SEAL-S reference point. The SEALDD server then sends SEALDD connection establishment notification with VAL service ID, the IP address and port to each of the VAL servers.

5. The SEALDD server allocates an IP address and port for sending and receiving packet over the SEAL-Uu reference point. The SEALDD server then sends regular data transmission connection establishment requests to each SEALDD client with SEALDD flow ID, VAL service ID, the IP address and port. Each request is responded to by the SEALDD clients. UE IP address (and port) can be included by the SEALDD client in the response or sent in a separate update message by SEALDD client if a different UE IP address is to be used in SEALDD connection user plane.

NOTE 1: Step 4 and step 5 can be done in parallel.

NOTE 2: Step 5 can be sent via PDU session (if exist) or via application triggering (if no PDU session exists).

6. The SEALDD clients further notify the VAL clients about the SEALDD connections being established.

Upon receiving multi-modal application traffic from VAL clients (not shown in the figure), the SEALDD clients send it to SEALDD server in SEALDD traffic. The SEALDD server identifies multi-modal application traffic based on the VAL service ID and further sends the multi-modal application traffic to the VAL servers. The downlink multi-modal application traffic sent from VAL server to VAL client is processed similarly.

### 7.x.3 Solution evaluation

This solution enables end-to-end multi-modal connection between VAL client and VAL server for application data delivery, with the connection initiation being based on policy, and it is well-aligned with the existing functionality in clause 9.2.2.3 of TS 23.433. The solution has minimal architectural impacts.

\*\*\* End of Changes \*\*\*