**3GPP TSG-SA5 Meeting #158*****S5-247143***

Orlando, USA, 18 -22 November 2024 revision of S5-246916

**Source: Ericsson Hungary Ltd.**

**Title: Rel-19 pCR TR 28.871 Reliable notification handling improvements**

**Document for: Approval**

**Agenda Item: 6.19.8**

# 1 Decision/action requested

***Approve the proposal.***

# 2 References

[1] 3GPP TR 28.871 Study on Service Based Management Architecture enhancement phase 3

[2] 3GPP TS 32.158 Design rules for REpresentational State Transfer (REST) Solution Sets (SS)

[3] 3GPP TS 28.622

# 3 Rationale

It is possible to create two NtfSubscriptionControl IOCs with the same notificationRecipientAddress which include (some) identical notificationTypes. Depending on the scope and notificationFilter the exact same notification message should be sent to two subscriptions. It is not clarified in TS 28.622 whether in this case the notification message would be sent once or twice in such cases.

It is proposed to specify that the notification shall be sent separately for each subscription. This simplifies processing on the consumer side. As the situation is assumed to be rare, the additional network, processing load is insignificant.

The above situation is critical for the sequence numbering scheme proposed in section TR 28.871 “5.13.3.1 Part#1 Add sequence number”. If the multiple subscriptions send notifications to the same notificationRecipientAddress the sequence numbers will be mixed up. For this reason the NtfSubscriptionControl.id shall be added to each notification.

An alternative solution would be to require that notificationRecipientAddress shall be unique for all subscriptions, but that is seen as a too strong restriction. A single consumer might like to separate the needed subscription into multiple NtfSubscriptionControl IOC settings.

# 4 Detailed proposal

**First change**

#### 5.13.3.1 Part#1 Add sequence number (REQ-rel-notif-1)

Delivery of notifications can be prevented by many reasons (overload, bugs, firewalls, proxy or concentrator nodes, etc.) some of which cannot be foreseen at the moment. The smartest approach to ensure reliability has been identified as adding a sequence number to all notifications so that the consumer can identify ‘holes’ in the sequence of the notifications and out-of-order notifications. Sequence numbering is a robust mechanism that can detect problems both for the current HTTP 1.1 protocol (see 3GPP TS 28.622 [2]) or any future protocols (e.g. HTTP 3).

Notifications can arrive out-of-order. If the HTTP connection is closed after sending a notification and a new connection is opened for the subsequent notification, the order of delivery is not guaranteed. The sequence number can be used to re-order the notifications.

Every notification should carry a monotonically increasing sequence number, that will be separate for each notification subscription. The sequence number should be a large unsigned integer that is reset to zero at start or restart of the producer. This will allow the consumer to detect missing or out-of-order notifications. Heartbeat notifications should also carry a new parameter lastEventTime that contains the eventTime parameter from the last (non-heartbeat) notification.

As multiple NtfSubscriptionControl MOIs may contain the same notificationRecipientAddress the sequence numbering for the different subscriptions may get mixed up. To avoid this problem every notification shall carry an identifier that uniquely identifies the subscription e.g. the DN of the NtfSubscriptionControl MOI.

To simplify consumer-side notification handling and sequence numbering in cases when the same notification would be sent to the same recipient for multiple subscriptions, it shall be clarified that such notifications shall be sent separately for each subscription.

Impacted specifications are: 3GPP TS 28.532 [14] and 3GPP TS 28.111 [8].

**End of changes**