**3GPP TSG-WG SA4 #126 RTC**

**Chicago, 13th -17th November 2023**

|  |
| --- |
| *CR-Form-v12.2* |
| **PSEUDO CHANGE REQUEST** |
|  |
|  | **26.113** | **CR** |  | **rev** | **1** | **Current version:** | **0.7.0** |  |
|  |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME | **X** | Radio Access Network |  | Core Network | **X** |

|  |
| --- |
|  |
| ***Title:***  | [iRTCW] Definition of iRTCW Procedures |
|  |  |
| ***Source to WG:*** | Qualcomm Incorporated |
| ***Source to TSG:*** | S4 |
|  |  |
| ***Work item code:*** | iRTCW |  | ***Date:*** | 7th November 2023 |
|  |  |  |  |  |
| ***Category:*** | B |  | ***Release:*** | Rel-18 |
|  | *Use one of the following categories:****F*** *(correction)****A*** *(mirror corresponding to a change in an earlier release)****B*** *(addition of feature),* ***C*** *(functional modification of feature)****D*** *(editorial modification)*Detailed explanations of the above categories canbe found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)…Rel-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19)* |
|  |  |
| ***Reason for change:*** | We propose the definition of a set of APIs to match the procedures defined in TS26.506.  |
|  |  |
| ***Summary of change:*** |  |
|  |  |
| ***Consequences if not approved:*** |  |
|  |  |
| ***Clauses affected:*** |  |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** |  |  |  Other core specifications  | TS/TR ... CR ...  |
| ***affected:*** |  |  |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  |  |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
|  |  |
| ***This CR's revision history:*** |  |

|  |
| --- |
| **First Change** |

## 4.1 General

### 4.1.1 HTTP resource URIs and paths

The resource URI used in each HTTP request to the API provider shall have the structure defined in subclause 4.4.1 of TS 29.501 [22], i.e.:

{apiRoot}/{apiName}/{apiVersion}/{apiSpecificResourceUriPart}

with the following components:

- {apiRoot} shall be set as described in TS 29.501 [22].

- {apiName}shall be set as defined by the following clauses.

- {apiVersion} shall be set to "v2" in this release of the specification.

- {apiSpecificResourceUriPart} shall be set as described in the following clauses.

### 4.1.2 Usage of HTTP

#### 4.1.2.1 HTTP protocol version

##### 4.1.2.1.1 RTC AF

Implementations of the RTC AF shall expose both HTTP/1.1 [24] and HTTP/2 [31] endpoints at interfaces RTC-1 and RTC-5, including support for the HTTP/2 starting mechanisms specified in section 3 of RFC 7540 [31]. In both protocol versions, TLS [29] shall be supported and HTTPS interactions should be used on these interfaces in preference to cleartext HTTP.

The Application Service Provider may use any supported HTTP protocol version at interface RTC-1.

The Media Session Handler and the RTC AS may use any supported HTTP protocol version at interface RTC-5.

All responses from the RTC AF that carry a message body shall include a strong entity tag in the form of an ETag response header and a modification timestamp in the form of a Last-Modified response header.

All endpoints shall support the conditional HTTP requests If-none-Match and If-Modified-Since.

NOTE: RTC AF can reuse the RTC-5 APIs for RTC-3.

#### 4.1.2.2 HTTP message bodies for API resources

The OpenAPI [23] specification of HTTP messages and their content bodies is contained in Annex X.

#### 4.1.2.3 Usage of HTTP headers

##### 4.1.2.3.1 General

Standard HTTP headers shall be used in accordance with clause 5.2.2 of TS 29.500 [21] for both HTTP/1.1 and HTTP/2 messages.

##### 4.1.2.3.2 Media Session Handler identification

The Media Session Handler in the RTC Client shall identify itself to the RTC AF at interface RTC-5 using a User-Agent request header (see section 5.3.3 of RFC 7231 [25]) in which the first element shall be a product identified by the token RTCMediaSessionHandler and optionally suffixed with a product-version.

The Media Session Handler may additionally supply a comment element in the User-Agent request header containing a vendor-specific identification string.

##### 4.1.2.3.3 RTC AF identification

The RTC AF shall identify itself using a Server response header (see section 7.4.2 of RFC 7231 [25]) of the following form:

RTCAF-{FQDN}/{implementationSpecificSuffix}

where {FQDN} shall be the Fully-Qualified Domain Name of the RTC AF exposed to the requesting client, and {implementationSpecificSuffix} shall be determined by the implementation.

##### 4.1.2.3.4 Support for conditional HTTP GET requests

All responses from the RTC AF that carry a resource message body shall include:

- a strong entity tag for the resource, conveyed in an ETag response header,

- a resource modification timestamp, conveyed in a Last-Modified response header, and

- a predicted time-to-live period for the resource, conveyed in a Cache-Control: max-age response header.

All API endpoints on the RTC AF that expose the HTTP GET method shall support conditional requests using the If-None-Match and If-Modified-Since request headers. API clients should not attempt to revalidate their cached copy of a resource using a conditional GET request before the indicated time-to-live period has elapsed.

##### 4.1.2.3.5 Support for conditional HTTP POST, PUT, PATCH and DELETE requests

All API endpoints on the RTC AF that expose the HTTP POST, PUT, PATCH or DELETE methods shall support conditional requests using the If-Match request header. The API client should supply a strong entity tag in an ETag request header when invoking any of these HTTP methods.

### 4.1.3 HTTP response codes

Guidelines for error responses to the invocation of APIs of NF services are specified in clause 4.8 of TS 29.501 [22]. API-specific error responses are specified in the respective technical specifications.

|  |
| --- |
| **Second Change** |

## 5.1 Provisioning (RTC-1) API

### 5.1.1 Overview

This clause defines the provisioning API used by the Application Provider to provision resources for their real-time communication sessions. The Provisioning API is an extension of the Provisioning API as defined in TS26.512 clause 7.

The following M1 procedures are relevant for RTC sessions:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Procedure** | **Inherited** | **Extended** | **Not Relevant** | **New** |
| Provisioning Sessions API |  | X |  |  |
| Server Certificates Provisioning |  |  | X (but can be reused) |  |
| Content Preparation Templates |  |  | X |  |
| Content Protocols Discovery |  |  | X |  |
| Content Hosting Provisioning |  |  | X |  |
| Consumption Reporting Provisioning | X |  |  |  |
| Metrics Reporting Provisioning |  | X |  |  |
| Policy Templates Provisioning |  | X (RTCQoSSpecification) |  |  |
| Edge Resources Provisioning | X |  |  |  |
| Event Data Processing Provisioning |  |  | X (but can be reused) |  |
| Configuration Provisioning |  |  |  | X |

### 5.1.2 Data Model

The data model from TS26.512 clause 7.2.3.1 is extended to support the additional RTC procedures. Table 5.1-1 specifies the modifications and extensions to the ProvisioningSession resource.

Table 5.2-1: Definition of ProvisioningSession resource

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Property name | Type | Cardinality | Usage | Description | Applicability |
| provisioningSessionId | ResourceId | 1..1 | C: RR: RO | A unique identifier for this Provisioning Session. | All types. |
| provisioningSession‌Type | Provisioning‌Session‌Type | 1..1 | C: RWR: ROU: – | The type of Provisioning Session. | All types. |
| aspId | AspId | 0..1 | C: WR: RO | The identity of the Application Service Provider responsible for this Provisioning Session, as specified in clause 5.6.2.3 of TS 29.514 [34]. | All types. |
| externalApplicationId | ApplicationId | 1..1 | C: RWR: ROU: RO | The external application identifier (see TS 29.571 [12]), nominated by the 5GMS Application Provider, to which this Provisioning Session pertains. | All types. |
| serverCertificateIds | Array(ResourceId) | 0..1 | C: –R: RO | A list of Server Certificate identifiers currently associated with this Provisioning Session. | downlink |
| contentPreparation‌TemplateIds | Array(ResourceId) | 0..1 | C: –R: RO | A list of Content Preparation Template identifiers currently associated with this Provisioning Session. | downlink,uplink |
| metricsReporting‌ConfigurationIds | Array(ResourceId) | 0..1 | C: –R: RO | A list of Metrics Reporting Configuration identifiers currently associated with this Provisioning Session. | downlink,uplink,rtc |
| policyTemplateIds | Array(ResourceId) | 0..1 | C: –R: RO | A list of Policy Template identifiers currently associated with this Provisioning Session. | downlink,uplink,rtc |
| edgeResources‌ConfigurationIds | Array(ResourceId) | 0..1 | C: –R: RO | A list of Edge Resources Configuration identifiers currently associated with this Provisioning Session. | downlink,uplink,rtc |
| eventDataProcessing‌ConfigurationIds | Array(ResourceId) | 0..1 | C: –R: RO | A list of Event Data Processing Configuration identifiers currently associated with this Provisioning Session. | downlink,uplink |
| provisionedConfigurationIds | Array(ResourceId) | 0..1 | C: -R: RO | A list of the provisioned configuration identifiers that are currently associated with this Provisioning Session. | rtc |

## 5.2 Configuration Provisioning API

### 5.2.1 Overview

The Configuration Provisioning API is used by the Application Provider to provision configuration that will be relayed to the MSH for usage with RTC sessions of that Application Provider.

### 5.2.2 Resource Structure

The Configuration Provisioning API is accessible through the following URL base path:

{apiRoot}/3gpp-maf*/{apiVersion}/*provisioning-sessions/{provisioningSessionId}/

Table 5.2.2-1 below specifies the operations and the corresponding HTTP methods that are supported by this API. In each case, the Provisioning Session identifier shall be substituted into {provisioningSessionId} in the above URL template and the sub-resource path specified in the second column of the table shall be appended to the URL base path.

Table 5.2.2‑1: Operations supported by the Configuration Provisioning API

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Sub‑resource path | Allowed HTTP method(s) | Description |
| Retrieve Configuration Provisioning | configuration-provisioning | POST | Used to create a Configuration resource. |
| Create Configuration Provisioning | GET | Used to retrieve an existing RTC1Configuration resource. |
| Update Configuration Provisioning | PUT, PATCH | Used to modify an existing RTC1Configuration resource. |
| Destroy Configuration Provisioning | DELETE | Used to delete an existing RTC1Configuration resource. |

### 5.2.3 Data Model

#### 5.2.3.1 ProvisionedConfiguration resource

The data model for the ProvisionedConfiguration resource is specified in Table 5.2.3.1-1 below:

Table 5.2.3.1-1: Definition of ProvisionedConfiguration resource

| Property name | Data Type | Cardinality | Description |
| --- | --- | --- | --- |
| offerTrustedStunServers | boolean | 0..1 | Indicates if the AF should provide a list of trusted STUN servers to the UE for usage with RTC sessions of this application provider. |
| stunServers | array(URL) | 0..1 | An array of trusted STUN servers that the application can use as ICE candidates. |
| offerTrustedTurnServers | boolean | 0..1 | Indicates if the RTC AF should provide a list of trusted TURN servers to the UE for usage with RTC sessions of this application provider. |
| turnServers | array(URL) | 0..1 | An array of trusted TURN servers that the application can use as ICE candidates. |
| offerTrustedSwapServers | boolean | 0..1 | Indicates if the AF should provide a list of trusted SWAP servers to the UE for usage with RTC sessions of this application provider. |
| swapServers | array(URL) | 0..1 | An array of trusted WebRTC signaling servers that support the SWAP protocol. If provided, the application shall use one of the listed servers for RTC sessions of this application provider. |

### 5.2.4 Operations

The RTC AF shall relay the configuration information to the MSH using the Configuration procedure, if requested by the Provisioning information.

|  |
| --- |
| **3rd Change** |

## 5.3 Policy Template Provisioning API

### 5.3.1 Overview

The Policy Template Provisioning API is an extension of the same procedure in TS26.512 clause 7.9. The data model is extended to support policies for real-time communication.

### 5.3.2 Data model

#### 5.3.2.1 PolicyTemplate resource

The data model for the PolicyTemplate resource is specified in table 5.3.2‑1 below:

Table 5.3.2-1: Definition of PolicyTemplate resource

| Property | Type | Cardinality | Usage | Description |
| --- | --- | --- | --- | --- |
| policyTemplateId | ResourceId | 1..1 | C: ROR: ROU: RO | Identifier of this Policy Template assigned by the 5GMS AF that is unique within the scope of the Provisioning Session. |
| state | string enum | 1..1 | C: ROR: ROU: RO | A Policy Template may be in the PENDING, INVALID, READY, or SUSPENDED state.Only a Policy Template in the READY state may be instantiated as a Dynamic Policy Instance and applied to media streaming sessions. |
| stateReason | Problem‌Details | 1..1 | C: ROR: ROU: – | Additional details about the current state of this Policy Template exposed to the 5GMS Application Provider by the 5GMS AF.The instance sub-property shall be present and shall indicate the URL of this Policy Template resource.The title sub-property shall be present and shall indicate a human-readable representation of the state property specified above, e.g. "Policy Template ready for use" or "Policy Template invalid".The detail sub-property shall be present and shall indicate a human-readable status/error message.All other properties shall be omitted. |
| externalReference | string | 1..1 | C: RWR: ROU: RW | Additional identifier for this Policy Template, unique within the scope of its Provisioning Session, that can be cross-referenced with external metadata about the media streaming session. |
| qoSSpecification | M1‌QoS‌Specification | 0..1 | C: RWR: ROU: RW | Specifies the network quality of service to be applied to media streaming sessions at this Policy Template. |
| rtcQosSpecification | array(RTCQoSSpecification) | 0..1 | C: RWR: ROU: RW | Specifies the network quality of service to be applied to the different media streams of the RTC session. |
| application‌Session‌Context | Object | 1..1 |  | Specifies information about the application session context to which this Policy Template can be applied. |
|  sliceInfo | Snssai | 0..1 | C: RWR: RWU: RW | As defined in clause 5.4.4.2 of TS 29.571 [12]. |
|  dnn | Dnn | 0..1 | C: RWR: RWU: RW | As defined in clause 5.3.2 of TS 29.571 [12]. |
| charging‌Specification | Charging‌Specification | 0..1 | C: RWR: RWU: RW  | Provides information about the charging policy to be used for this Policy Template. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |
| --- |
| **4th Change** |

## 7.1 Dynamic Policies API

### 7.1.1 Overview

An RTC AF shall support the Dynamic Policy API over the RTC-5 interface, as defined in the present sub-clause. The Dynamic Policy API allows both the MSH and the trusted ICE or WebRTC Signaling Server AS to request a specific QoS and charging policy to be applied to the data flows of an RTC session. The API defines a set of data models, resources, and the related procedures for the creation and management of the dynamic policy request.

The RTC AF uses either the Npcf\_PolicyAuthorization API over N5, or the Nnef\_AFSessionWithQoS over N33 interface to request the allocation of the dynamic policy with the PCF.

The security procedures as described in clause 13.4 of TS33.501 shall apply for the Dynamic Policy API over RTC-5 interface.

This procedure is an extension of the Dynamic Policies API as specified in TS26.512 clause 11.5.

NOTE: RTC AF can reuse the RTC-5 APIs for RTC-3.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |
|  |  |
|  |  |

### 7.1.2 Data model

#### 7.1.2.1 DynamicPolicy resource

The DynamicPolicy resource for RTC is define in the following table.

Table 7.1-1: Definition of Dynamic Policy resource

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property name | Data type | Cardinality | Usage | Description |
| dynamicPolicyId | ResourceId | 1..1 | RO | Unique identifier for this Dynamic Policy. |
| policyTemplateId | ResourceId | 1..1 | C: RWR: ROU: RW | Identifies the Policy Template which should be applied to the application flow(s). |
| serviceDataFlowDescriptions | array(Service‌Data‌Flow‌Description) | 1..1 | C: RWR: ROU: RW | Describes the service data flows managed by this Dynamic Policy. |
| mediaType | MediaType | 0..1 | C: RWR: ROU: RW | The type of media carried by the application flows listed in service‌DataFlow‌Descriptions. |
| provisioningSessionId | ResourceId | 1..1 | C: RWR: ROU: RW | Provisioning Session identifier obtained from Service Access Information (see clause 11.2.3).Uniquely identifies Provisioning Session, which is linked to the Application Service Provider. |
| qosSpecification | M5‌QoS‌Specification | 0..1 | C: RWR: ROU: RW | Describes the network Quality of Service properties of this Dynamic Policy. |
| rtcQoSSpecification | array(RTCQoSSpecification) | 0..1 | C: RWR: ROU: RW | For RTC sessions, this provides a list of QoS specification that are associated with the streams of the RTC session. The RTCQoSSpecification is defined in clause 5.3.2.1. |
| enforcementMethod | string | 0..1 | C: ROR: ROU: RO | Description of the Policy Enforcement Method. The parameter is set by the 5GMSd AF. |
| enforcementBitRate | integer | 0..1 | C: ROR: ROU: RO | Description of the enforcement bit rate. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

### 7.1.4 Operations

The policyTemplateId uniquely identifies the Policy Template, to which the Dynamic Policy Instance is associated. An RTC session is composed of multiple components that correspond to RTP streams or data channels. Multiple RTP media sessions may be multiplexed into a single RTP stream, thus producing a single QoS flow.

The Dynamic Policy resource contains a serviceDataFlowDescription property which contains the service data flow template according to TS 23.503. The ServiceDataFlowDescription shall contain a flowDescription object (including 5-Tuples, Type of Service, Security Parameter Index, etc.).

The qosSpecifcation object shall be present and it shall contain the following properties:

- marBwDlBitRate or marBwUlBitRate, indicating the maximum requested bit rate by the Media Session Handler.

- mirBwDlBitRate or mirBwUlBitRate, indicating the minimum requested bit rate by the Media Session Handler.

- minDesBwDlBitRate or minDesBwUlBitrate, indicating the minimum bit rate desired by the Media Session Handler.

The requested QoS parameters shall be within the limits of the provisioning QoS Policy Template.

For RTP and data channel sessions, the mediaIdentifier shall be associated with the media identifier as signaled by the “mid=” attribute of the corresponding RTP session in the SDP.

|  |
| --- |
| **5th Change** |

## 7.2 Configuration Information API

### 7.2.1 Overview

The Configuration Information API is used by the Media Session Handler to obtain configuration information for use with RTC sessions of a specific Application Service Provider from the RTC AF.

### 7.2.2 Resource Structure

The Configuration Information API is accessible through the following URL base path:

{apiRoot}/3gpp-maf*/{apiVersion}/*configuration-information/

Table 7.2.2-1 below specifies the operations and the corresponding HTTP methods that are supported by this API.

Table 7.2.2‑1: Operations supported by the Configuration Information API

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Sub‑resource path | Allowed HTTP method(s) | Description |
| Retrieve Configuration Information  | configuration-information | POST | Used to create a Configuration Information resource. |
| Create Configuration Information | GET | Used to retrieve an existing Configuration Information resource. |
| Update Configuration Information | PUT, PATCH | Used to modify an existing Configuration Information resource. |
| Destroy Configuration Information | DELETE | Used to delete an existing Configuration Information resource. |

### 7.2.3 Data Model

#### 7.2.3.1 ConfigurationInformation resource

The data model for the ConfigurationInformation resource is specified in Table 7.2.3.1-1 below:

Table 7.2.3.1-1: Definition of *ConfigurationInformation* resource

| Property name | Data Type | Cardinality | Description |
| --- | --- | --- | --- |
| stunServers | array(URL) | 0..1 | An array of trusted STUN servers that the application can use as ICE candidates. |
| turnServers | array(URL) | 0..1 | An array of trusted TURN servers that the application can use as ICE candidates. |
| swapServers | array(URL) | 0..1 | An array of trusted WebRTC signaling servers that support the SWAP protocol. If provided, the application shall use one of the listed servers for RTC sessions of this application provider. |