**Agenda item:** 10.5

**Source:** Qualcomm Inc.

**Title: 5G-RTC STUN functionality**

**Document for** Discussion andAgreement

# Introduction

In this contribution, we propose procedures for offering 5G system support to WebRTC sessions through 5G-RTC STUN and TURN servers.

# STUN functionality

## 2.1 General

In the second collaboration scenario, a more integrated support by the 5G system for WebRTC sessions comes through offering 5G-RTC STUN functionality. A 5G-RTC STUN AF is a STUN server (compatible with RFC8489). In addition, the server offers 5G functionality to support the WebRTC session. The STUN server receives bind requests as part of the ICE negotiation. These requests allow the STUN server to discover the public IP address and port number of the connection, the so-called reflexive transport address. The request and response may contain STUN attributes, which can be marked as comprehension-required or comprehension-optional. A STUN server that doesn’t understand a comprehension-required attribute must reply with an error message. IANA maintains a registry of the STUN attributes.

We propose to introduce a set of new STUN attributes to trigger the 5G support for a WebRTC application. These would allow the 5G-RTC STUN server to request QoS allocation and charging for the media connection. The following attributes are proposed as comprehension-optional attributes:

* 3GPP-PRIVATE-ADDRESS: consists of a protocol family indicator, an IP address and a port number, the DNN and possible S-NSSAI, that correspond to the private transport address as seen by the UE. This information may be needed to identify the NAT association and ensure proper QoS allocation for the media session.
* 3GPP-QOS: consists of an indication of the QoS attributes that are recommended for the connection associated with this request. It consists of an average bitrate, a max bitrate, a max latency, and a max PLR indicator.

Other attributes, e.g. to convey codec recommendations or configurations may also be defined.

A 5G-RTC STUN server shall support these attributes. The 5G-RTC STUN server uses the information in a successful binding to request QoS allocation and charging policy. It may also respond with a recommendation about the target QoS parameters and/or a recommended codec using the STUN attributes.

## 2.2 Call Flow

The call flow for the invocation of a 5G-RTC STUN server is provided as follows:

The different steps are as follows:

1. The ASP that desires to offer better 5G support for its WebRTC-based applications, creates a Provisioning session with the 5G-RTC Provisioning AF. This step is optional and the MNO may decide to offer support for WebRTC sessions without an associated Provisioning session.
2. The Provisioning AF shares the QoS and media configuration templates with all associated 5G-RTC AFs. This can for example be done through storing this information in the UDF.
3. The WebRTC configuration, including the STUN and TURN server list, is sent to the MSH as part of the service access information
4. The application fetches the list of pre-configured STUN and TURN servers from the local configuration. The configuration indicates for each server if it is 5G-RTC enabled.
5. The App submits a binding request to the 5G-RTC STUN server with the additional attributes to trigger the ICE negotiation.
6. The 5G-RTC STUN AF retrieves the associated QoS template and media configuration
7. The 5G-RTC STUN AF invokes the Npcf\_PolicyAuthorization to allocate QoS to the media session of the STUN/TURN binding.
8. The 5G-RTC STUN AF creates the binding response and sends the additional information back to the application together with the address binding.
9. The application updates the offer/answer SDP based on the received STUN information

## 2.3 Configuration Procedure

The MSH receives a list of 5G-RTC STUN and TURN servers that are provided by the MNO for 5G system integration of WebRTC sessions. It also receives recommendations about the QoS templates for WebRTC sessions. The MSH makes this information available to the application through the RTC-6 interface.

The information may be formatted as follows:

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Description** |
| ice\_af\_list | array(Object) | A list of ICE AFs that an application may use for the ICE negotiation. |
|  type | enumeration  | the type of the ICE server, which may either be STUN or TURN. |
|  url | string | the URL of the ICE server. |
|  5g\_enabled | boolean | indicates if the current ICE server is 5G-RTC enabled. A 5G-RTC ICE server supports the STUN attributes and is able to perform tasks such as QoS allocation. |
| media\_recommendations | array(Object) | A list of media recommendations that the application should use to create its offer/answer. |
|  type | enumeration | indicates the media type, which can be AUDIO, VIDEO, TEXT. |
|  codec | string | the recommended codec configuration for the media type |
|  average\_bitrate | integer | the recommended average bitrate |
|  max\_bitrate | integer | the recommended peak bitrate |
|  |  |  |

For Web applications, the configuration information may be accessible through standardized W3C APIs such as the Indexed Database API or the File API.

## 2.4 5G-RTC STUN Procedures

The 5G-RTC STUN server shall use the N5 or N33 interfaces to request QoS allocation for the associated STUN binding. Upon determining the 3-Tuple for the connection (public IP address, port number, and protocol), the 5G-RTC STUN server invokes the Nnef\_AFsessionWithQoS or the Npcf\_PolicyAuthorization methods to request QoS for the identified QoS flow.

# Proposal

We propose to agree the collaboration scenarios and document them in the PD.

# References

[1] IETF RFC8489, Session Traversal Utilities for NAT (STUN)

[2] IETF RFC 8656, Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)

[3] IETF RFC 8445, Interactive Connectivity Establishment (ICE):

 A Protocol for Network Address Translator (NAT) Traversal

[4] IANA, Session Traversal Utilities for NAT (STUN) Parameters, https://www.iana.org/assignments/stun-parameters/stun-parameters.xhtml#stun-parameters-4