TRAM
Internet-Draft
Intended status: Standards Track
Expires: August 16, 2015

P. Martinsen
H. Wildfeuer
Cisco

February 12, 2015

Differentiated prIorities and Status Code-points Using Stun Signalling (DISCUSS)
draft-martinsen-tram-discuss-02

Abstract

This draft describes a mechanism for information exchange between an application and the network. The information provided from the application to the network MAY be used by a network element in the path to modify its behavior to improve application quality of experience (QoE). Likewise, the information provided by the network to the application MAY be used by the application to modify its behavior to optimize for QoE.

The information provided from the application to the network can also be useful for middleboxes that are responsible for security at edges of network (e.g. firewalls) or other middleboxes in determining how to treat the packets delivered from this application.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

Table of Contents

1. Introduction

In the context of Content, Mobile, Fixed Service, Service Providers, Enterprise and Private networks have a need to prioritize packet flows end-to-end. These flows are often dynamic, time-bound, encrypted, peer-to-peer, possibly asymmetric, and might have different priorities depending on network conditions, direction, time of the day, dynamic user preferences and other factors. These

factors may be time variant, and thus need to be signalled. Moreover, in many cases of peer-to-peer communication, flow information is known only to the endpoint. These considerations, coupled with the trend to use encryption for browser-to-browser communication [I-D.ietf-rtcweb-security-arch], imply that access lists, deep packet inspection and other static prioritization methods cannot be employed successfully to prioritize packet flows.

The lack of congestion control in UDP may lead to problems as described in [I-D.eggert-tsvwg-rfc5405bis]. The mechanism described in this document can be used to introduce fairness and congestion control for UDP streams.

There is a need for a solution that is easy for the application developer to use. That means consistent behavior on all supported platforms and preferably without need of administrative privileges to set and read values. The solution also needs to be able to cross administrative domains without the risk of being rewritten. [[Q1: This draft will only offer tamper detection of some of the values. Further discussion regarding the incentive to lie is needed. --palmarti]]

This draft describes how these problems can be solved by defining a few strictly defined STUN [RFC5389] attributes which can be added to any STUN message the client wants to send. STUN messages are typically sent during the ICE [RFC5245] connectivity check phase when the media session is established, or when keep-alive STUN messages are sent after the session is established. The application is not limited to those two scenarios, if some communication between application and network is needed it can choose to do so at any time.

Devices on the media path can use the information in the STUN attributes to prioritize the flow, perform traffic engineering, provide network analytics or as a gateway to existing methods for prioritizing flows (DSCP [RFC2474]). Applications can use information in network status attribute to influence rate stating points or rate adaption mechanisms.

2. General Usage

This draft defines several attributes that can be added to a STUN message; STREAM-TYPE, BANDWIDTH-USAGE, STREAM-PRIORITY and NETWORK-STATUS. See Section 6 for the formal description. It is RECOMMENDED to add them to a STUN request response pair, especially if the NETWORK-STATUS attribute is in use. This allows the information gathered to be sent back to the requesting agent in the the STUN response.

The STREAM-TYPE, BANDWIDTH-USAGE, STREAM-PRIORITY attributes MUST be added before any INTEGRITY attribute. It is RECOMMENDED to only add these attributes to STUN messages containing a INTEGRITY attribute as this prevents tampering with the content of the attribute.

If the client wants to receive feedback from the network it must add a null NETWORK-STATUS attribute. A null NETWORK_STATUS attribute is created by filling in the all the fields in the attribute with 0x0 values. This attribute MUST be added after the INTEGRITY attribute, as on-path devices may write information into this attribute. Having a readily available attribute to write into will save the the on-path device from growing buffers to add their own attribute. On path devices SHOULD not add their own NETWORK-STATUS attribute (or any other STUN attribute).

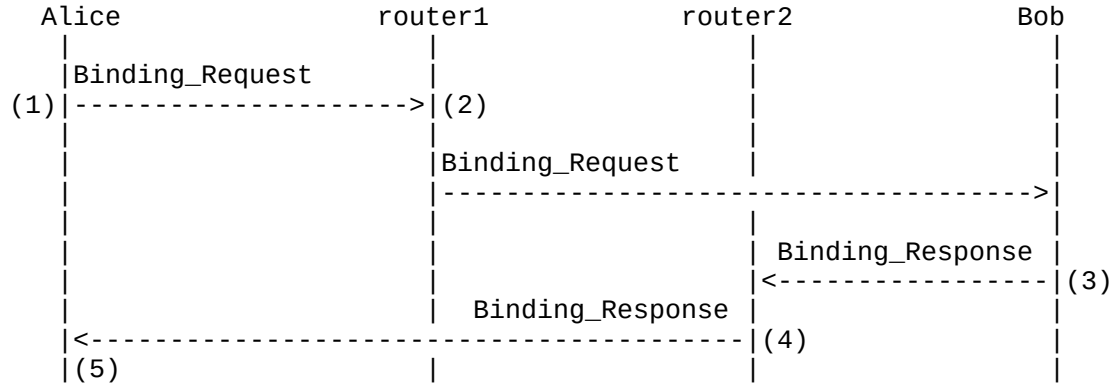
If an agent receives a STUN request with a NETWORK-STATUS attribute after the INTEGRITY attribute, it should copy the content into a new NETWORK-STATUS attribute and add it before the INTEGRITY attribute when sending the STUN response. A new null NETWORK-STATUS attribute can be added after the INTEGRITY attribute. New STUN attributes described in this draft can also be added describing the stream in this direction.

If an agent receives a STUN response with a NETWORK-STATUS attribute before the INTEGRITY attribute, this describes the stream in the upstream direction. A NETWORK-STATUS attribute after the INTEGRITY attribute describes the stream in the downstream direction.

It might make sense to distinguish DISCUSS packets from normal STUN packets. This would prevent unnecessary processing of normal STUN packets on the network nodes.

[[Q2: A few alternatives (Needs discussion): ---1: Alter the STUN magic cookie. (But than i would not be a STUN packet anymore, and that raises a new set of problems) ---2: Add a special this is DISCUSS attribute. This must be the first attribute in the message. This allows for network node to look for DISCUSS packets at a fixed offset without needing to parse the entire packet. ---3: Alter the transaction id. This might be problematic if using it in conjunction with ICE connectivity checks. But probably fine in other scenarios. ---4: Define a new STUN Method. Also brakes ICE, makes it harder to tag onto attributes to already in use messages. --palmarti]]

[[Q3: Do we want to restrict this to req/resp or do we want to allow for the attributes to be added in this fashion for indications as well? --palmarti]]



DISCUSS example flow

1. Alice creates a Binding Request, adds STREAM-TYPE, BANDWIDTH-USAGE, STREAM-PRIORITY attributes before the INTEGRITY attribute and a single null NETWORK-STATUS attribute after the INTEGRITY attribute.

2. Router1 inspects the STUN Request message and reads any STREAM-TYPE, BANDWIDTH-USAGE, or STREAM-PRIORITY attributes and the information they contain. It then updates the NETWORK-STATUS attribute with any information the router have. It then forwards the request.

3. Bob processes the STUN Request. When Bob builds the response, it copies the NETWORK-STATUS attribute into the STUN Response before the INTEGRITY check and adds new null NETWORK-STATUS attribute after the INTEGRITY attribute. Bob then transmits the message.

4. Router2 (first DISCUSS network element for the downstream direction) inspects the Response message, reads the STREAM-TYPE, BANDWIDTH-USAGE, or STREAM-PRIORITY attributes and MAY alter the NETWORK-STATUS attribute located after the INTEGRITY attribute. It then transmits the message.

5. When Alice receives the STUN Response, she can extract the STREAM-TYPE, BANDWIDTH-USAGE, or STREAM-PRIORITY attributes and the two NETWORK-STATUS attributes to get a complete picture of what the remote agent is sending and how the status of both the upstream and downstream path.

3. Network Processing

This section describes the processing of DISCUSS packets by network devices.

3.1. Packet Processing by Network Device

Network devices are said to support DISCUSS if they perform inspection of packets being forward or switched in order to identify an DISCUSS STUN packet. These devices will also be able to read/ write STUN attributes to/from this packet. It is not required that every network device in the path support DISCUSS. It is expected that DISCUSS will have the most value being implemented at certain points in the network (PIN's) such as WAN edge devices, wireless access devices, and Internet gateways.

Network devices that support DISCUSS MAY utilize the information provided by the application in the STUN attributes to modify their behavior. These include the attributes defined in this document with the exception of the NETWORK-STATUS attribute. The NETWORK-STATUS attribute SHOULD NOT be used by the DISCUSS capable network device to modify its behavior. The intent of the NETWORK-STATUS attribute is for the application to modify its behavior.

If the NETWORK-STATUS attributes exists in a DISCUSS packet after an INTEGRITY attribute, the DISCUSS capable network device MUST process it as described in this section. NETWORK-STATUS attributes that exist before the INTEGRITY attribute MUST NOT be modified by the network device. The modifications to the NETWORK-STATUS attribute are:

- Update the Node Cnt field in the attribute. The device SHALL increment this field by one unless it is at its maximum (saturated) value. If the field is at its maximum value, the device SHALL NOT modify this field.

- Overwrite the attribute CS bit if the value at this device is "worst" than the current value. In other words, only write to this bit if the device is experiencing congestion on relevant queues/interfaces for this flow AND the current value of this field is 0 (Off).

The determination of congestion at a device is out of the scope of this document. Setting of CS bit to On by the device is meant to provide direct feedback to the application of potential or current loss of packets in its flow (s). The application can then react to this indication by altering its encoding of information in the packet to deal with congestion/packet loss, e.g. reduce its encoding rate or

switch to embedded encoding. Devices SHOULD ensure that the DISCUSS capable applications that do react to congestion notification by reducing their transmission rate be treated properly to ensure fairness with non reacting applications (i.e. ensure fairness for well behaving applications).

The DISCUSS STUN packet SHOULD experience minimal extra processing delay through the DISCUSS capable network device relative to non-DISCUSS packets in the flow. The DISCUSS STUN packet MAY be placed out of order in the packet flow, but SHOULD NOT be delayed more than a few packet interval times.

3.2. Interaction with DSCP

One of the attributes that may be added to the STUN packet by the application is the STREAM-PRIORITY attribute. This attribute indicates the relative priority of streams inside of an application session. This attribute is compatible with the use of DSCP (or other priority markings) at the networking layer as described in this section.

Since transport layer markings may be modified by middle boxes or devices in the path or at the interface of the application itself due to the lack of support in the OS network stack, the STREAM-PRIORITY attribute can be used as a mechanism for ensuring proper QoS treatment through multiple domains. DISCUSS capable device may use the STREAM-PRIORITY attribute to remark the DSCP value to the appropriate value. DSCP re-marking based on STREAM-PRIORITY attribute may make sense at certain PIN’s, e.g. gateway between network domains (e.g. managed network to/from Internet), access switches in managed network, etc. The translation from the Priority number in the STREAM-PRIORITY attribute to the correct transport layer marking (e.g. DSCP) is implementation specific and out of the scope of this document.

[I-D.dhesikan-tsvwg-rtcweb-qos] provides the recommended DSCP values for webrtc enabled browsers to use for various classes of traffic.

4. Interaction with ICE

An ICE connectivity check is performed by sending a STUN Binding indication. Prior to sending the agent can add one STREAM-TYPE attribute. If added, only one MUST be added. This is to avoid unnecessary large STUN packets during the connectivity checks. If the connectivity check is sent on a 5-tuple that multiplexes different types of media and more detailed information wants to be signalled it should be done after the connectivity check phase is finished.

This limits the information the STUN messages are able to convey during the connectivity checks, but also avoids adding network confusion with BANDWIDTH-USAGE attributes describing different paths that never going to be utilized.

[[Q4: Problem with consent freshness if not based on STUN. --palmarti]]

5. Multiplexed Streams

In some scenarios a 5-tuple can be used to transport several media streams. BUNDLE [I-D.ietf-mmusic-sdp-bundle-negotiation] describes such a mechanism.

At times, the different "streams" carried in this bundle require very different treatment from the network, including the ability to prioritize some of these "streams" over others. For example, the application my bundle video and audio in the same 5-tuple flow, but would like the network to prioritize the delivery of audio over that of video in the case of congestion. Another example is the use of embedded (or scalable) coding for video. Per RFC 6190 [RFC6190], using Multi-session Transmission (MST) the layers are transported in seperate sub-flows (RTP sessions) within the bundled flow. Using the STREAM-TYPE attribute with the extension to identify the sub-flow and its priority would allow network elements, if capable, to provide differentiated services even in the case of bundling.

For RTP/SRTP based flows, the existence and attributes for sub-flows in the flow MAY be indicated by the application via the SUB-STREAM-xxx attributes. These attributes MUST only be included if the equivalent STREAM-xxxx attributes are included. It is expected that only a sub-set of network elements representing bottleneck Points in Network (PIN) will be able to inspect the higher layer protocols to differentiate sub-flows, so it is important to describe the agregate flow, and then the sub-flows. The SUB-STREAM-xxxx attributes are similar to the corresponding STREAM-xxxx attributes with the addition of the application layer identifier field. For the case of RTP/SRTP, this field is the SSRC assigned to the flow. Note that this will only work for non-header encrypted SRTP.

When describing the aggregate stream with a STREAM-TYPE attribute there are two possibilities to describe the streams that are multiplexed. Adding one attribute for each type (Audio, Video,++), or to save a few bits on the wire it is also possible to construct the STREAM-TYPE so a one type value describes several types. For example audio have the value of 1 and application data have the value of 4. If the STREAM_TYPE value is set to 5 the only combination that gives that is audio and application data. As previously discussed,

in the case of bundling, the agregate stream attribute MUST be included before the optional sub-stream attributes

The other attributes BANDWIDTH-USAGE, STREAM-PRIORITY and NETWORK-STATUS SHOULD only be added once as they describe the behavior of the 5-tuple and not individual streams.

6. New STUN attributes

This STUN extension defines the following new attributes:

0xXXX0: STREAM-TYPE
0xXXX1: BANDWIDTH-USAGE
0xXXX2: STREAM-PRIORITY
0xXXX8: SUB-STREAM-TYPE
0xXXX9: SUB-BANDWIDTH-USAGE
0xXXXA: SUB-STREAM-PRIORITY
0xYYYY: NETWORK-STATUS

6.1. STREAM-TYPE

This attribute have a length that are multiples of 4 (32) so no padding is necessary.

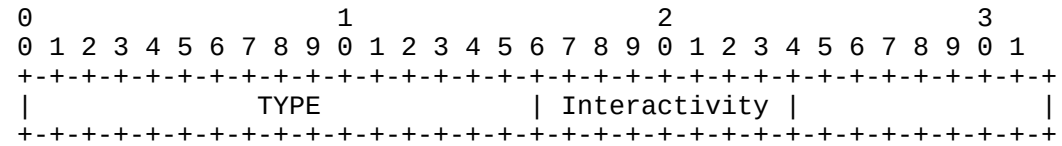


Figure 1: STREAM TYPE Attribute

TYPE: STREAM TYPE is a 16 bit value defined in Figure 2 below describing the flow.

0x0001 Audio
0x0002 Video
0x0004 Application Data
0x0008 Other

Figure 2: STREAM Types

Interactivity: Is a 8 bit value defined in Figure 3 below describing the flow.

```
0x00 Undef
0x01 Stream (Broadcast? Oneway?)
0x02 Interactive
```

Figure 3: Interactivity Types

It is possible to combine the stream types if a stream contains more than one type.

If a 5-tuple is used to send both a audio and video stream, the stream type can be set to 0x0006. This can be useful if the application wants to hint that the 5-tuple contains several streams, This is useful if the attribute is added to STUN binding requests during ICE connectivity checks. If more information regarding multiplexed streams is needed it is possible to add more than one attribute to a STUN message (See section ??). This can be done to STUN messages that are being sent after the connectivity check phase is finished (Keep-alive, consent freshness). During this phase the added size of the STUN messages pose no security threat.

6.2. BANDWIDTH-USAGE

This attribute have a length that are multiples of 4 (32) so no padding is necessary.

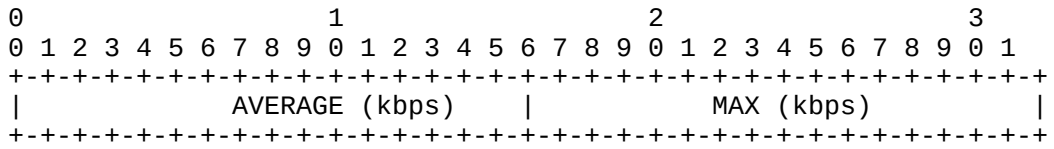


Figure 4: BANDWIDTH USAGE Attribute

AVERAGE: Expected sustained bandwidth usage for this stream. Note that for elastic types of streams like video, sudden large movements in the picture my lead to this value being inaccurate.

MAX: The maximum bandwidth usage for this stream. If the sustained and max value differ greatly it might be safe to assume that an elastic encoder is in use. (Would it be useful to say something about expected BURST lengths?)

6.3. STREAM-PRIORITY

This attribute have a length that are multiples of 4 (32) so no padding is necessary.

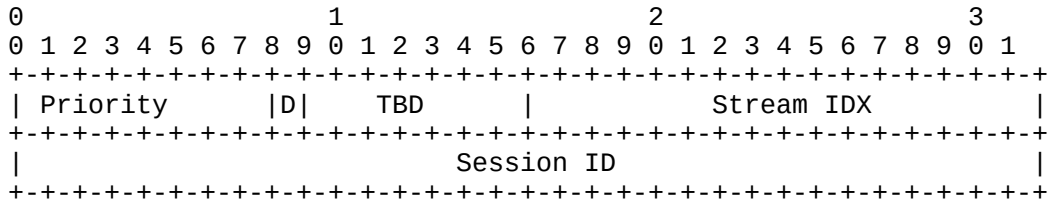


Figure 5: STREAM PRIORITY Attribute

Priority: Describes this streams priority among other streams coming from this endpoint/application (With same session ID). Values range from 255 (0xFF) to 0.

D: Delay sensitive. The application can set this bit as a hint to the network element that the stream is delay sensitive. (Unsure if this is useful)

Stream IDX: Application can choose set this to ease debugging in the network. A reasonable value can foe example be the index have in the SDP.

Session ID: Identification to distinguish what session this stream is part of. This MUST have the same value for all the media streams the application wants to give differentiated services. (Note that this ID may overlap with other streams that originates from a different IP address. The network element MUST only prioritize among streams with the same Session Id originating from the same IP)

6.4. NETWORK-STATUS

This attribute have a length that are multiples of 4 (32) so no padding is necessary. The values are kept in the same attribute to make it easier for the network element to process it. Only one attribute, with static placement of the fields. [[Q5: Does this matter? Could we have several attributes with possible different ordering without any problem for the network element? --palmarti]]

This attribute MUST be added after any INTEGRITY attribute in the STUN message. Values in this attribute can be updated along the network path by nodes that are not able to regenerate a correct INTEGRITY attribute.

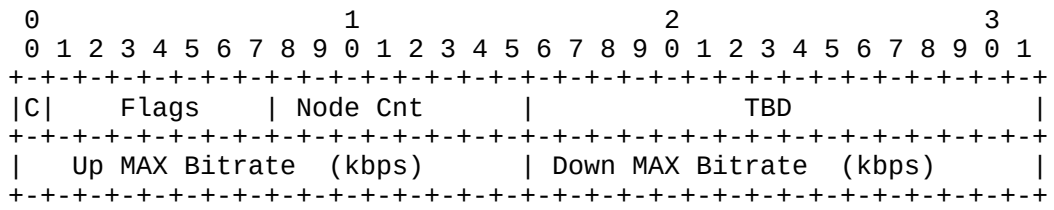


Figure 6: NETWORK-STATUS Attribute

C: Congestion Status. This bit is set to indicate that there is congestion at the network device's relevant queues/interfaces for this flow. The network element should set this bit to 1 (On) if it is experiencing congestion. This bit is set to 0 (off) when the attribute is created by the application. The application that sees this bit set might act on it by doing some rate adaption or similar action.

Flags: 7 more bits available for flags.

Node Cnt: Numbers of nodes that supports DISCUSS in the network path. Any router on the path that understands DISCUSS should increase this number. This field is set to 0 when the attribute is created by the application.

TBD: 16 more bits available for useful info.

Up MAX Bitrate: Available MAX bit-rate the router is able to handle for the 5-tuple in the UP direction. (Same direction as the packet is moving)

Down MAX Bitrate: Available MAX bit-rate the router is able to handle for the 5-tuple in the DOWN direction. (Opposite direction as the packet is moving)

6.5. SUB-STREAM-TYPE, SUB-STREAM-PRIORITY

These attributes are identical format to their aggregate stream version (STREAM-TYPE, STREAM-PRIORITY) with the addition of a transport layer identifier. The transport layer identifier is a 64 bit field which contains the unique identifier of the sub-stream for which the attribute applies.

Currently, only RTP transport is supported with the identifier being the SSRC currently used by the sub-stream.

7. IANA Considerations

IANA is requested to add the following attributes to the STUN attribute registry [iana-stun],

- 0xXXX0: STREAM-TYPE (0xXXX0, in the comprehension-optional range)
- 0xXXX1: BANDWIDTH-USAGE (0xXXX1, in the comprehension-optional range)
- 0xXXX2: STREAM-PRIORITY (0xXXX2, in the comprehension-optional range)
- 0xYYYY: NETWORK-STATUS (0xYYYY, in the comprehension-optional range)

8. Implementation Status

[Note to RFC Editor: Please remove this section and reference to [RFC6982] prior to publication.]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [RFC6982]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [RFC6982], "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

8.1. NATtools

Organization: Cisco

Description: Open-Source ICE, TURN and STUN implementation.

Implementation: https://github.com/cisco/NATTools

Level of maturity: Code is stable. Tests being run to learn more on how to leverage the information shared between client and network.

Coverage: Implements the DISCUSS attributes

Licensing: BSD

Implementation experience: Draft was implemented with internal video test clients. Wireless access point implemented STUN detection in the media path and acted on the information in the DISCUSS attributes. After running tests in different congestion scenarios it is clear that sharing information between endpoint and network can help with congestion and end-user experience. This approach required little effort to implement on the client side.

Contact: Paal-Erik Martinsen <palmarti@cisco.com>.

9. Security Considerations

Due to the security implications described in [I-D.thomson-mmusic-ice-webrtc] where large STUN packet are used to amplify an attack, keeping the added STUN attributes small is a important design consideration.

To avoid unwanted information leakage the new defined STUN attributes defined in this draft are strictly defined. No more information should be leaked that an on-path device could learn by observing the stream over time or do some deep packet analysis. This draft would benefit from more discussions on this topic.

It is also worth noticing that the STUN attributes defined should be treated as hints, and more work is needed regarding how to deal with misbehaving clients or network devices.

10. Acknowledgements

Authors would like to thank Dan Wing, Anca Zamfir, Jon Snyder and Cullen Jennings for their comments and review.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010.

[RFC6190] Wenger, S., Wang, Y., Schierl, T., and A. Eleftheriadis, "RTP Payload Format for Scalable Video Coding", RFC 6190, May 2011.

[RFC6982] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", RFC 6982, July 2013.

11.2. Informational References

[I-D.ietf-rtcweb-security-arch]
Rescorla, E., "WebRTC Security Architecture", draft-ietf-rtcweb-security-arch-09 (work in progress), February 2014.

[I-D.thomson-mmusic-ice-webrtc]
Thomson, M., "Using Interactive Connectivity Establishment (ICE) in Web Real-Time Communications (WebRTC)", draft-thomson-mmusic-ice-webrtc-01 (work in progress), October 2013.

[I-D.dhesikan-tsvwg-rtcweb-qos]
Dhesikan, S., Druta, D., Jones, P., and J. Polk, "DSCP and other packet markings for RTCWeb QoS", draft-dhesikan-tsvwg-rtcweb-qos-06 (work in progress), March 2014.

[I-D.ietf-mmusic-sdp-bundle-negotiation]
Holmberg, C., Alvestrand, H., and C. Jennings, "Multiplexing Negotiation Using Session Description Protocol (SDP) Port Numbers", draft-ietf-mmusic-sdp-bundle-negotiation-05 (work in progress), October 2013.

[I-D.eggert-tsvwg-rfc5405bis]
Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage Guidelines", draft-eggert-tsvwg-rfc5405bis-01 (work in progress), June 2014.

[iana-stun]
IANA, , "IANA: STUN Attributes", April 2011, <http://www.iana.org/assignments/stun-parameters/stun-parameters.xml>.

Authors' Addresses

Paal-Erik Martinsen
Cisco Systems, Inc.
Philip Pedersens vei 20
Lysaker, Akershus 1366
Norway

Email: palmarti@cisco.com

Herb Wildfeuer
Cisco Systems, Inc.
821 Alder Drive
Milpitas, California 95035
United States

Email: hwildfeu@cisco.com