

RTCWEB
Internet-Draft
Intended status: Standards Track
Expires: November 26, 2013

R. Penno
T. Reddy
D. Wing
Cisco
M. Boucadair
France Telecom
May 25, 2013

PCP Considerations for WebRTC Usage
draft-penno-rtcweb-pcp-00

Abstract

This document describes the motivations for WebRTC applications to be PCP-aware and the benefits provided by PCP-capable NATs and Firewalls.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on November 26, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](http://trustee.ietf.org/license-info) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
2.	Notational Conventions	3
3.	Advantages of using PCP with WebRTC	3
3.1.	Firewalls Blocking UDP	3
3.2.	Firewalls permit specific WebRTC servers	5
3.3.	ICE Lite	5
3.4.	Reducing Call Set-Up Time	6
3.4.1.	ICE Speedup	6
3.4.2.	Pre-allocating ports to speed call setup time	6
3.5.	NAT	7
3.6.	Optimizing NAT and Firewall Keepalives	7
3.7.	Faster Flow Failure Detection	8
3.8.	3GPP Selective IP Traffic Offload (SIPTO)	8
3.9.	Auditing	9
3.10.	NAT64	10
4.	Usage of PCP with STUN and TURN	10
4.1.	STUN	10
4.2.	TURN	11
5.	Security Considerations	11
6.	IANA Considerations	11
7.	Acknowledgments	12
8.	References	12
8.1.	Normative References	12
8.2.	Informative References	13
	Authors' Addresses	15

[1.](#) Introduction

Port Control Protocol (PCP, [[RFC6887](#)]) provides a mechanism to describe a flow to the network. The primary driver for PCP has been creating port mappings on NAT and firewall devices. When doing this,

PCP pushes flow information from the host into the network (specifically to the network's NAT or firewall device), and receives information back from the network (from the NAT or firewall device).

The Web Real-Time communication (WebRTC) framework [[I-D.ietf-rtcweb-overview](#)] provides the protocol building blocks to support direct, interactive, real-time communication using audio, video, collaboration, games, etc., between peer web-browsers. WebRTC application use Interactive Connectivity Establishment (ICE) protocol [[RFC5245](#)] for gathering candidates, prioritizing them, choosing default ones, exchanging them with the remote party, pairing them and ordering them into check lists. Once all of the above steps have been completed the participating ICE agents can begin a phase of connectivity checks and eventually select a pair of candidates that will be used for real-time communication.

This specification describes the reasons for WebRTC applications to be PCP-aware and use PCP along side with STUN and TURN. It also explains the benefits for a network that deploy PCP-controlled NATs and Firewalls.

[2.](#) Notational Conventions

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 [[RFC2119](#)].

This document uses terms defined in [[RFC5389](#)] and [[RFC5766](#)].

eNodeB: The eNodeB is a base station entity that supports the Long-Term Evolution (LTE) air interface [[RFC6459](#)].

[3.](#) Advantages of using PCP with WebRTC

The below sections explain the problems with NAT and Firewall, current techniques used to solved them and the PCP solution in these scenarios.

[3.1.](#) Firewalls Blocking UDP

Enterprise networks may deploy firewalls with restrictive policies configured to block UDP traffic. These firewalls may be configured to permit TCP or HTTP(s) traffic only. One of the reasons for blocking UDP could be that there is no way for the firewall to determine when the endpoints have terminated the call, in which case the firewall has to close the dynamic mapping based on firewall UDP mapping timer value. [\[RFC4787\]](#) mandates that the UDP mapping timer for NAT must not expire in less than 2 minutes and recommends a default value of five minutes or more. Firewalls are likely to follow the same recommendation for their UDP mapping timer, which would be applicable to both IPv4 and IPv6 firewalls. The behavioural requirements for IPv6 firewalls is explained in [section 3.2.3](#) of

[\[RFC6092\]](#). [\[I-D.hutton-rtcweb-nat-firewall-considerations\]](#) gives details of other organization e.g. a public service agency or university that deploy firewall which may have restrictive firewall policy to block UDP traffic.

Modern firewalls may also have application-layer gateways (ALGs) perform policy enforcement to permit peer-to-peer UDP media session. Using the ALG, a firewall can determine when the call is terminated and close any dynamic mappings created for the media session. But the problem is the session signaling between the WebRTC application running in the browser and the web server could be using TLS, in which case the ALG no longer has access to the signaling. Moreover, WebRTC does not enforce a particular session signaling protocol to be used, so firewalls using ALGs would fail to inspect the signaling to identify the 5-tuple used for each media stream. Furthermore, the session signaling and the peer-to-peer media may traverse different Firewalls.

Using TURN for all such communication to by-pass firewall causes the following problems:

- o TURN server could increase media latency as explained in [section 4.1.2.2 of \[RFC5245\]](#). Using a reliable and ordered protocol like TCP instead of UDP to transfer real-time media is problematic as delays would be directly noticeable and may be unacceptable to the user.
- o High-end TURN server would be needed (For example when TLS-over-TCP transport is used between the client and the server) to cater

to all such calls.

- o TURN server could either be located in the DMZ of the enterprise network or located in the public Internet. If the TURN server is located in the public Internet it comes at a high cost to the provider of the TURN server, since the server typically needs a high-bandwidth connection to the Internet as explained in the Introduction of [\[RFC5766\]](#). As a consequence, it is best to use a TURN server only when a direct communication path cannot be found. When the client and a peer use ICE to determine the communication path, ICE will use hole punching techniques to search for a direct path first and only use a TURN server when a direct path cannot be found.
- o Some of the other limitations of TURN explained in [section 2.6 of \[RFC5766\]](#) are, the value of the Diffserv field may not be preserved, the Explicit Congestion Notification (ECN) field may be reset etc.

PCP resolves the above problems by restricting firewall traversal to authorized PCP clients and communicating mapping lifetimes and call termination between the PCP client and the PCP-controlled firewall. A PCP Server can also enforce per-host quotas for mappings.

[3.2.](#) Firewalls permit specific WebRTC servers

When an enterprise uses a trusted WebRTC server deployed in a 3rd party network for communication, the enterprise firewall could have granular policies to permit peer-to-peer UDP media session only when the call is initiated using the selected WebRTC server (Dr. Good) it trusts and block others (Dr. Evil). Firewall policy has a white-list of permitted outside applications/sites and can blacklist HTTP(S) connections via various forms of detections (destination DNS lookup, HTTP URL Filtering, DPI proxy that at least performs HTTPS inspection of URL in certificate, Subject Name of TLS exchange and validates SSL records etc). Firewall in this configuration would also block TCP connection to arbitrary TURN servers in the Internet. 3GPP networks may also have a similar configuration where IMS services of certain other operators are permitted and others are blocked [[\[TR33.830\]](#)].

With PCP, this problem is solved by associating the media session with the signaling session. This is done by sending a cryptographic token in the signaling which authorizes the firewall mapping for the media session.

3.3. ICE Lite

For scenarios where the client is connected to the public Internet and has public IP address at which it can receive packets from the remote peer and uses ICE LITE implementation explained in [section 2.7 of \[RFC5245\]](#), the ICE Lite endpoint will not generate its own ICE connectivity checks, by definition. Thus, if an ICE Lite endpoint is behind a firewall that blocks unsolicited incoming traffic then ICE Lite will fail.

This workaround for solving the problem is by using full ICE or by changing the filtering policy on the firewall to permit unsolicited incoming UDP traffic which would effectively disable the purpose of firewall. Full ICE will take more time to be adapted especially with legacy VoIP equipment which will initially start with ICE-Lite implementation as discussed in section 6 of [\[I-D.cbran-rtcweb-nat\]](#).

With PCP, a firewall can filter incoming UDP traffic and PCP client can communicate exceptions to the firewall to permit specific mappings when a call is active. In this way, the ICE Lite endpoint and its network are protected from unsolicited incoming UDP traffic, and can still operate using ICE Lite (rather than full ICE).

3.4. Reducing Call Set-Up Time

There are initiatives to speedup ICE processing in order to reduce call setup time using techniques such as Trickle ICE [\[I-D.rescorla-mmusic-ice-trickle\]](#) and RTP multiplexing Section 4.4 of [\[I-D.ietf-rtcweb-rtp-usage\]](#). Trickle ICE can begin connectivity checks while the endpoint is still gathering candidates and can considerably shorten the time necessary for ICE processing to complete. RTP multiplexing suggests to bind interactive audio and

interactive video to the same 5-tuple {dest addr, source addr, protocol, dest port, source port} to optimize NAT resource usage and shorten the call setup time.

PCP can help reduce call set-up time by speeding up ICE and, if appropriate, at the same time allowing each media for flow over a different 5-tuple.

[3.4.1.](#) ICE Speedup

ICE requires time to perform its setup operations. This time grows in proportion to the number of transport sessions which must be opened in order to support the call. If using a different IP addresses and/or ports for audio versus video streams, call setup time will increase. The precise amount of this increase depends on the type of NAT and other factors like packet loss. The use of RTP Multiplexing technique introduces some QoS challenges in many networks, e.g., In Mobile Networks the QoS considerations are explained in Section 4.1 of [[I-D.reddy-rtcweb-mobile](#)].

Fast call setup time and QoS can both be retained by using PCP. External IP addresses and ports can be learnt faster using PCP than other techniques because the PCP client is communicating only with PCP servers in the Home and Service Provider network. In contrast, STUN and TURN servers may be located halfway around the world from the endpoint adding delay to learn server-reflexive and relayed candidates. Trickle ICE can begin connectivity checks using the candidates learnt from PCP, while the endpoint is still gathering other candidate types and thus can considerably shorten the time necessary for ICE processing to complete.

[3.4.2.](#) Pre-allocating ports to speed call setup time

The external IP:port allocated through PCP belong to the client for duration of the lifetime of the mapping. This means that connectivity checks for a new call can begin immediately using the already allocated external IP:port and if necessary the client can extend the lifetime of the mapping. TURN allocations can also be extended using Refresh transaction to update the time-to-expiry of existing allocation and thus can be used for a new call immediately. Server Reflexive candidates learnt using STUN can also be maintained

for a new call but requires the endpoint to send frequent keepalives to prevent the NAT and firewall mappings from expiring.

The PCP client for fast call setup can also use `PORT_SET` option [[I-D.ietf-pcp-port-set](#)] requesting the PCP server to pre-allocate contiguous ports with port parity preservation.

[3.5.](#) NAT

Direct peer-to-peer communication is not possible if both NATs are of a certain type that changes the outside port number when connecting to new hosts (NAT behaviour "address-dependent mapping" or "address and-port-dependent mapping" as described in [[RFC4787](#)]).

When such NAT devices are encountered, communication can be established using a media relay (TURN) server. But using TURN servers is expensive as explained in [section 4.1.1.2 of \[RFC5245\]](#) and other challenges of using TURN are discussed in [Section 3.1](#). Relayed candidates should only be used as last-resort when connectivity checks using other candidate types are not successful.

PCP improves this situation by creating explicit bindings on PCP-controlled NATs and can adjust their mapping and filtering behavior so that connections can be successfully created. PCP can also recursively communicate with multiple layers of NATs using [[I-D.ietf-pcp-proxy](#)]. Usage of STUN and PCP for learning candidates, prioritization, encoding them in offer or answer is explained in [Section 4.1](#).

[3.6.](#) Optimizing NAT and Firewall Keepalives

Applications like WebRTC need to keep their Network Address Translator (NAT) and firewall mappings alive for long periods of time, even when they are otherwise not sending or receiving any traffic. The signaling protocol used for WebRTC would want to keep the client-server connection alive for as long as the application is running. When the WebRTC application has otherwise no traffic to send, specific keep-alive messages are sent periodically to ensure that the NAT/Firewall state in the middle does not expire. The endpoint would also have to send keepalives for the media session to

keep NAT/Firewall bindings alive. As NAT/firewall mapping timers may

be short and unknown to the endpoint, the keepalive messages are sent frequently.

In cellular mobile networks, frequent keepalive messages make the radio transition between active and power-save states causing signaling congestion. The excessive time spent on the active state due to keepalives also greatly reduces the battery life of the cellular connected devices such as smartphones or tablets.

PCP is useful to reduce NAT and firewall keepalive messages (e.g., Section 3.4 of [[I-D.reddy-pcp-optimize-keepalives](#)]) for both signaling protocol and media session.

[3.7.](#) Faster Flow Failure Detection

If a NAT device has rebooted, lost its mappings or has its external IP address changed then it may take few minutes before the endpoint realizes that the connectivity is lost, that would result in disruption of signaling and media traffic. Application can find that the signaling session is broken by using TCP keepalive probes, the time taken to detect that the connection is broken depends on the frequency of keepalive probes. If the endpoint is using sendonly media streams, it may take few minutes based on RTCP reports to realize that the connectivity is lost. WebRTC client will then have to re-establish connection with the WebRTC server and initiate ICE restart.

Using the Rapid Recovery procedure explained in [Section 14 of \[RFC6887\]](#), the PCP client upon receiving a PCP ANNOUNCE from a PCP server, becomes aware that the PCP server has rebooted or lost its mapping state. The PCP client issues new PCP requests to recreate any lost mapping state and thus reconstructs lost mappings fast enough that existing media streams do not break and re-establish connectivity with its WebRTC server.

If for some reason PCP server determines that some or all of its mappings have become unusable (e.g., when a home gateway is assigned a different external IPv4 address by the upstream DHCP server) then the PCP server automatically repairs its mappings and notifies its clients about the new External IP address and port as part of the Rapid Recovery techniques explained in [Section 14.2 of \[RFC6887\]](#). The client based on this notification can use MICE [[I-D.wing-mmusic-ice-mobility](#)] or ICE Restart to achieve RTP Mobility.

[3.8.](#) 3GPP Selective IP Traffic Offload (SIPTO)

Given the exponential growth in the mobile data traffic, Mobile Operators are looking for ways to offload some of the IP traffic flows at the nearest access edge that has an Internet peering point. This approach results in efficient usage of the mobile packet core and helps lower the transport cost. Since Release 10, 3GPP starts supporting of Selected IP Traffic Offload (SIPTO) function defined in [TS23.060][TS23.060], [TS23.401]. The SIPTO function allows an operator to offload certain types of traffic at a network node close to the UE's point of attachment to the access network. Limited Mobility support available with SIPTO is explained in section 2.3.3 of [I-D.zuniga-dmm-gap-analysis].

If SIPTO is carried out in a Traffic offload Function (TOF) entity in the path between the Radio stations and the Mobile Gateway (MGW) as explained in [I-D.reddy-rtcweb-mobile] and the Mobile Node (MN) roams from one eNodeB and changes its point of attachment to a new eNodeB NAT changes. In this case host candidates for the MN will not change but MN will be behind a new NAT after roaming. It may take few minutes before the MN realizes that the connectivity is lost, resulting in disruption of signalling and media traffic. Application can find that the signaling session is broken by using TCP keepalive probes, the time taken to detect that connection is broken depends on the frequency of the keepalive probes. If the endpoint is using sendonly media streams, it may take few minutes based on RTCP reports to realize that the connectivity is lost. WebRTC client will then have to re-establish connection with the WebRTC server and initiate ICE restart.

The problem can be mitigated by the following mechanism using PCP:

When TOF receives the SIPTO rules for the MN, the PCP-controlled NAT at TOF sends unicast PCP ANNOUNCE response to the MN informing it that the NAT has changed. WebRTC application using PCP can verify that external IP addresses and ports have changed for the media streams and proceed accordingly (e.g., MICE [I-D.wing-mmusic-ice-mobility] or ICE Restart to achieve RTP Mobility).

3.9. Auditing

On certain networks, it is necessary to audit communications across the network firewall and attribute those communications to certain users or users running certain applications. The use case for auditing is also explained in Section 4.2.5.1 of [I-D.ietf-rtcweb-use-cases-and-requirements].

Today, this is done by tracking IP address assignment on the network and auditing lots of mappings created by firewalls.

PCP improves that auditing by PCP Authentication [[I-D.ietf-pcp-authentication](#)]. A PCP server can audit all traffic including media sessions from inside an enterprise premises to any external peer. An enterprise that uses an WebRTC based web application for communication and desires to audit all WebRTC based application sessions used from inside the company towards any external peer can deploy a PCP-controlled firewall and enforce a policy on the PCP-controlled firewall to mandate PCP client authentication. Only after successful authentication, PCP client will be permitted to create dynamic mappings on the firewalls and NATs.

[3.10.](#) NAT64

For the IPv6-only WebRTC client to establish media session with IPv4-only WebRTC client it must learn prefix64(s).

The workaround for solving the problem is by using heuristics is explained in [[I-D.ietf-behave-nat64-discovery-heuristic](#)]. Various other solutions including STUN for discovery based on heuristics are discussed in [[I-D.ietf-behave-nat64-learn-analysis](#)].

PCP allows to learn PREFIX64 when a NAT64 is in the path [[I-D.ietf-pcp-nat64-prefix64](#)]. PCP client can directly communicate with PCP-controlled NAT64 device to learn the Prefix64(s). This feature is useful to help establishing successful media session between an IPv6-only WebRTC client and an IPv4-only WebRTC client. The other advantages of using PCP is that endpoint will be notified whenever the Network Specific Prefix (NSP) is changed and endpoint will also learn multiple NSPs configured in the network.

Experimental results related to the use of this feature for SIP-based applications in general are provided in Section 4.2 of [[I-D.boucadair-pcp-nat64-experiments](#)].

[4.](#) Usage of PCP with STUN and TURN

[4.1.](#) STUN

This section explains the procedure to use STUN and PCP with ICE [\[RFC5245\]](#):

The ICE agent learns external IP addresses and ports using the PCP MAP opcode. If server reflexive candidates and external IP addresses learnt using PCP are different than the candidates learnt through STUN, the PCP discovered candidates are encoded in the ICE offer and answer just like the server reflexive candidates learnt using STUN [\[RFC5389\]](#). When using the recommended formula explained in

[Section 4.1.2.1 of \[RFC5245\]](#) to compute priority for the candidate learnt through PCP, the ICE agent should use a preference value greater than the server reflexive candidate and hence they are tested before the server reflexive candidates.

The recommended type preference value is 105 for candidates discovered using PCP and is explained in [section 4.2 of \[RFC6544\]](#).

During connectivity checks the ICE agent SHOULD check if the XOR-MAPPED-ADDRESS from the STUN Binding response matches the external address and port provided by PCP MAP response.

- o If the match is successful, then it indicates that only PCP-aware NATs exist between the peers. PCP can further be used to keep the NAT bindings alive and close the mappings.
- o If the match is not successful then it indicates PCP unaware NATs exist between the peers.

[4.2.](#) TURN

TURN server may be used for the following reasons even if PCP capable Firewalls and NATs exist:

- o Users of WebRTC based web application may choose to use TURN so as to not expose the host candidate addresses to the remote peer for privacy reasons.
- o IPv6 support in TURN includes IPv4-to-IPv6 and IPv6-to-IPv4 relaying [\[RFC6156\]](#).
- o ICE connectivity checks using the candidates provided by STUN and

PCP could fail because the endpoint is behind PCP-unaware NAT that performs address-dependent mapping and thus only relayed candidate allocated from the TURN server gets selected for media.

- o TURN server could also be used for RTP Mobility [[I-D.wing-mmusic-ice-mobility](#)], etc.

[5.](#) Security Considerations

Security considerations discussed in [[RFC6887](#)] are to be taken into account. PCP authentication [[I-D.ietf-pcp-authentication](#)] MAY also be used.

[6.](#) IANA Considerations

This document does not require any action from IANA.

Penno, et al.

Expires November 26, 2013

[Page 11]

Internet-Draft

PCP with WebRTC

May 2013

[7.](#) Acknowledgments

The authors would like to thank Charles Eckel for review and comments.

[8.](#) References

[8.1.](#) Normative References

[[I-D.ietf-pcp-authentication](#)]

Wasserman, M., Hartman, S., and D. Zhang, "Port Control Protocol (PCP) Authentication Mechanism", [draft-ietf-pcp-authentication-01](#) (work in progress), October 2012.

[[I-D.ietf-pcp-proxy](#)]

Boucadair, M., Penno, R., and D. Wing, "Port Control Protocol (PCP) Proxy Function", [draft-ietf-pcp-proxy-02](#) (work in progress), February 2013.

[[I-D.ietf-rtcweb-rtp-usage](#)]

Perkins, C., Westerlund, M., and J. Ott, "Web Real-Time Communication (WebRTC): Media Transport and Use of RTP", [draft-ietf-rtcweb-rtp-usage-06](#) (work in progress), February 2013.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", [RFC 5389](#), October 2008.
- [RFC5766] Mahy, R., Matthews, P., and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", [RFC 5766](#), April 2010.
- [RFC6092] Woodyatt, J., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", [RFC 6092](#), January 2011.
- [RFC6156] Camarillo, G., Novo, O., and S. Perreault, "Traversal Using Relays around NAT (TURN) Extension for IPv6", [RFC 6156](#), April 2011.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", [RFC 6887](#), April 2013.

[8.2.](#) Informative References

- [I-D.boucadair-mmusic-altc]
Boucadair, M., Kaplan, H., Gilman, R., and S. Veikkolainen, "Session Description Protocol (SDP) Alternate Connectivity (ALTC) Attribute", [draft-boucadair-mmusic-altc-09](#) (work in progress), January 2013.
- [I-D.boucadair-pcp-nat64-experiments]
Abdesselam, M., Boucadair, M., Hasnaoui, A., and J. Queiroz, "PCP NAT64 Experiments", [draft-boucadair-pcp-nat64-experiments-00](#) (work in progress), September 2012.
- [I-D.cbran-rtcweb-nat]
Bran, C., Kaufman, M., Jennings, C., and J. Rosenberg, "WebRTC Network Address Translation", [draft-cbran-rtcweb-nat-02](#) (work in progress), October 2011.

- [I-D.hutton-rtcweb-nat-firewall-considerations]
 Stach, T., Hutton, A., and J. Uberti, "RTCWEB Considerations for NATs, Firewalls and HTTP proxies", [draft-hutton-rtcweb-nat-firewall-considerations-00](#) (work in progress), March 2013.
- [I-D.ietf-behave-nat64-discovery-heuristic]
 Savolainen, T., Korhonen, J., and D. Wing, "Discovery of the IPv6 Prefix Used for IPv6 Address Synthesis", [draft-ietf-behave-nat64-discovery-heuristic-17](#) (work in progress), April 2013.
- [I-D.ietf-behave-nat64-learn-analysis]
 Korhonen, J. and T. Savolainen, "Analysis of solution proposals for hosts to learn NAT64 prefix", [draft-ietf-behave-nat64-learn-analysis-03](#) (work in progress), March 2012.
- [I-D.ietf-pcp-nat64-prefix64]
 Boucadair, M., "Learn NAT64 PREFIX64s using PCP", [draft-ietf-pcp-nat64-prefix64-02](#) (work in progress), May 2013.
- [I-D.ietf-pcp-port-set]
 Sun, Q., Boucadair, M., Sivakumar, S., Zhou, C., Tsou, T., and S. Perreault, "Port Control Protocol (PCP) Extension for Port Set Allocation", [draft-ietf-pcp-port-set-01](#) (work in progress), May 2013.
- [I-D.ietf-rtcweb-overview]

Alvestrand, H., "Overview: Real Time Protocols for Browser-based Applications", [draft-ietf-rtcweb-overview-06](#) (work in progress), February 2013.

- [I-D.ietf-rtcweb-use-cases-and-requirements]
 Holmberg, C., Hakansson, S., and G. Eriksson, "Web Real-Time Communication Use-cases and Requirements", [draft-ietf-rtcweb-use-cases-and-requirements-10](#) (work in progress), December 2012.

- [I-D.reddy-pcp-optimize-keepalives]

Reddy, T., Isomaki, M., Wing, D., and P. Patil,
"Optimizing NAT and Firewall Keepalives Using Port Control
Protocol (PCP)", [draft-reddy-pcp-optimize-keepalives-01](#)
(work in progress), January 2013.

[I-D.reddy-rtcweb-mobile]

Reddy, T., Kaippallimalil, J., R, R., and R. Ejzak,
"Considerations with WebRTC in Mobile Networks", [draft-reddy-rtcweb-mobile-03](#) (work in progress), May 2013.

[I-D.rescorla-mmusic-ice-trickle]

Rescorla, E., Uberti, J., and E. Iovov, "Trickle ICE:
Incremental Provisioning of Candidates for the Interactive
Connectivity Establishment (ICE) Protocol", [draft-rescorla-mmusic-ice-trickle-01](#) (work in progress), October
2012.

[I-D.wing-mmusic-ice-mobility]

Wing, D., Patil, P., Reddy, T., and P. Martinsen,
"Mobility with ICE (MICE)", [draft-wing-mmusic-ice-mobility-03](#) (work in progress), January 2013.

[I-D.zuniga-dmm-gap-analysis]

Zuniga, J., Bernardos, C., Melia, T., and C. Perkins,
"Mobility Practices and DMM Gap Analysis", [draft-zuniga-dmm-gap-analysis-03](#) (work in progress), December 2012.

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation
(NAT) Behavioral Requirements for Unicast UDP", [BCP 127](#),
[RFC 4787](#), January 2007.

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

[RFC6459] Korhonen, J., Soinen, J., Patil, B., Savolainen, T.,
Bajko, G., and K. Iisakkila, "IPv6 in 3rd Generation
Partnership Project (3GPP) Evolved Packet System (EPS)",
[RFC 6459](#), January 2012.

- [RFC6544] Rosenberg, J., Keranen, A., Lowekamp, B.B., and A.B. Roach, "TCP Candidates with Interactive Connectivity Establishment (ICE)", [RFC 6544](#), March 2012.
- [RFC6544] Rosenberg, J., Keranen, A., Lowekamp, B.B., and A.B. Roach, "TCP Candidates with Interactive Connectivity Establishment (ICE)", [RFC 6544](#), March 2012.
- [TR33.830] 3GPP, , "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Feasibility study on IMS firewall traversal (Release 12).", September 2012.
- [TS23.060] 3GPP, , ""General Packet Radio Service (GPRS); Service description; Stage 2", June 2012.", September 2012.
- [TS23.401] 3GPP, , "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access (Release 11), 3GPP TS 23.401, V11.2.0 (2012-06).", September 2012.

Authors' Addresses

Reinaldo Penno
Cisco Systems, Inc.
170 West Tasman Drive
San Jose 95134
USA

Email: repenno@cisco.com

Tirumaleswar Reddy
Cisco Systems, Inc.
Cessna Business Park, Varthur Hobli
Sarjapur Marathalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: tiredddy@cisco.com

Dan Wing
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134
USA

Email: dwing@cisco.com

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: mohamed.boucadair@orange.com

