

PCP Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: November 29, 2013

M. Boucadair  
France Telecom  
T. Reddy  
P. Patil  
D. Wing  
Cisco  
May 28, 2013

Using PCP to Reveal a Host behind NAT  
draft-boucadair-pcp-nat-reveal-01

Abstract

This document describes how to use PCP to retrieve the identity of a host behind a NAT. Two use cases are discussed and the PCP applicability is analyzed. This document extends PCP with a new OpCode called QUERY OpCode.

The proposed mechanism is valid for all NAT flavors including NAT44, NAT64 or NPTv6.

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 29, 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 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. Requirements Language and Terminology . . . . .	3
3. Problem Space . . . . .	3
3.1. Policy and Charging Control Architecture . . . . .	3
3.2. NAT between the PCEF and AF . . . . .	4
3.3. NAT before PCEF . . . . .	5
4. PCP Applicability . . . . .	6
4.1. NAT between the PCEF and AF . . . . .	6
4.2. NAT before PCEF . . . . .	7
5. QUERY OpCode . . . . .	9
5.1. QUERY Request Format . . . . .	9
5.2. QUERY Response Format . . . . .	10
5.3. Generating a QUERY Request . . . . .	11
5.4. Processing a QUERY Request . . . . .	12
5.5. Processing a QUERY Response . . . . .	13
6. Applicability Scope of QUERY OpCode . . . . .	13
7. IANA Considerations . . . . .	13
8. Security Considerations . . . . .	13
9. References . . . . .	14
9.1. Normative References . . . . .	14
9.2. Informative References . . . . .	14
Authors' Addresses . . . . .	15

## 1. Introduction

As reported in [RFC6269], several issues are encountered when an IP address is shared among several subscribers. These issues are encountered in various deployment contexts: e.g., Carrier Grade NAT (CGN), application proxies or A+P [RFC6346].

This document extends Port Control Protocol [RFC6887] to identify a host among those sharing the same IP address in certain scenarios.

The proposed technique can be used independently or combined with the host identifier, denoted as HOST\_ID defined in [I-D.ietf-intarea-nat-reveal-analysis].

Additional scenarios requiring host identification are listed in [I-D.boucadair-intarea-host-identifier-scenarios].

## 2. Requirements Language and Terminology

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 note uses terminology defined in [RFC6887].

## 3. Problem Space

### 3.1. Policy and Charging Control Architecture

Figure 1 depicts a reference architecture of a mobile network [RFC6342].

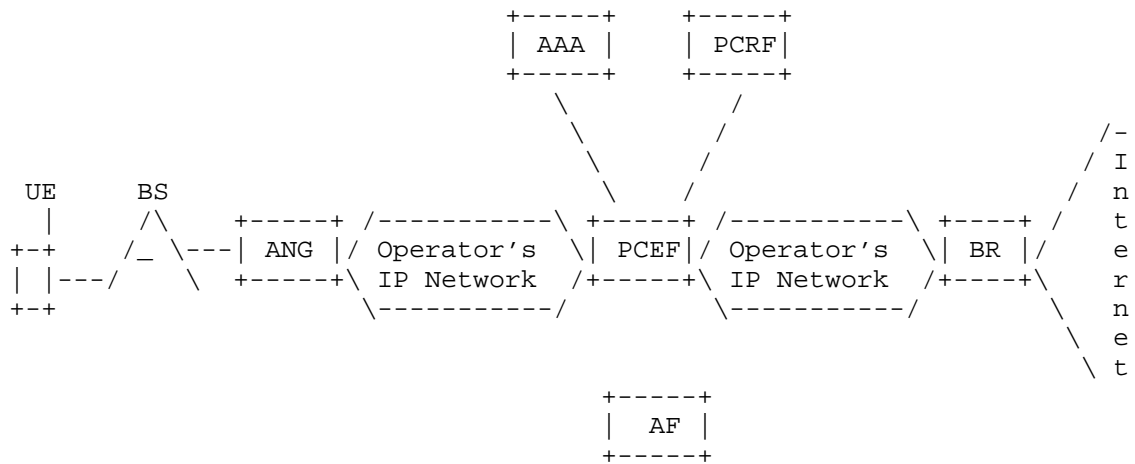


Figure 1: Mobile Network Architecture

The main involved functional elements are:

- o UE (User Equipment) is a mobile node.
- o Policy and Charging Rule Function (PCRF) which is responsible for determining which policy and charging control rules are to be applied [TS.23203].
- o Policy and Charging Enforcement Function (PCEF) which performs policy enforcement (e.g., Quality of Service (QoS)) and flow-based charging [TS.23203].
- o Application Function (AF) is an element offering applications that require dynamic policy and/or charging control [TS.23203].

- o Access Network Gateway (ANG), Base Station (BS) and Border Router (BR) are defined in [RFC6342].

Section 3.2 and Section 3.3 explain the encountered problems to identify individual UEs when a NAT is involved in the data path. The use of PCP to solve those problems is analyzed in Section 4.

### 3.2. NAT between the PCEF and AF

Figure 2 shows a scenario where a NAT function is located between the PCEF and AF.

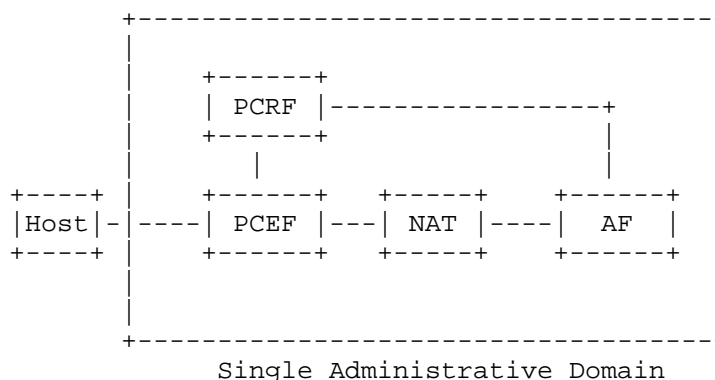


Figure 2: NAT between PCEF and AF

The main issue in this case is that PCEF, PCRF and AF all receive information bound to the same UE but cannot correlate between the piece of data visible for each entity. Concretely,

- o PCEF is aware of the IMSI (International Mobile Subscriber Identity) and an internal IP address assigned to the UE.
- o AF receives an external IP address and port as assigned by the NAT function.
- o PCRF is not able to correlate between the external IP address/port assigned by the NAT and the internal IP address and IMSI of the UE. For instance, the offered QoS on internal IPv4 address and the (shared) external IPv4 address may need to be correlated for accounting purposes.
- o The IP address seen by the AF is shared among multiple UEs using NAT, the PCRF needs to be able to inspect the NAT binding to disambiguate among the individual UEs. AF will not be able to treat UE traffic based on policy provided by PCRF.

This scenario can be generalized as follows (Figure 3):

- o Policy Enforcement Point (PEP, [RFC2753])
- o Policy Decision Point (PDP, [RFC2753])

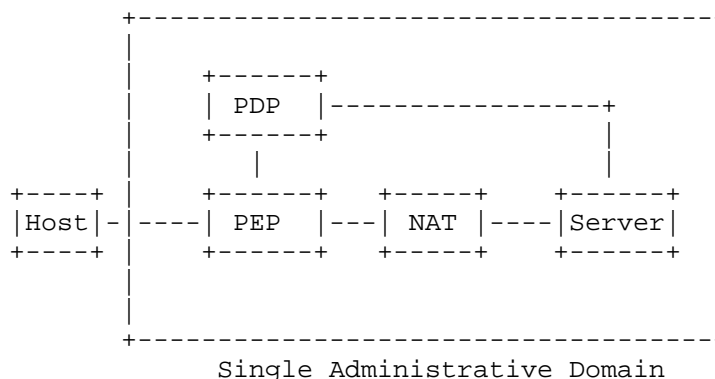


Figure 3: NAT between PEP and Server

### 3.3. NAT before PCEF

Figure 4 shows an alternative scenario in which a NAT function is located before PCEF. The main issue is that PCEF and AF are only aware of the external IP address and the external port number assigned by the NAT function. PCEF/AF will fail to identify each UE behind NAT since multiple UEs share the same external IP address and thus are unable to treat incoming connections differently.

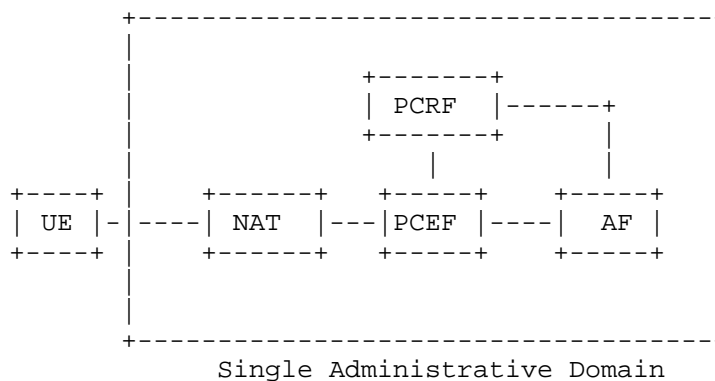


Figure 4: NAT before PCEF

This scenario can be generalized as follows (Figure 5):

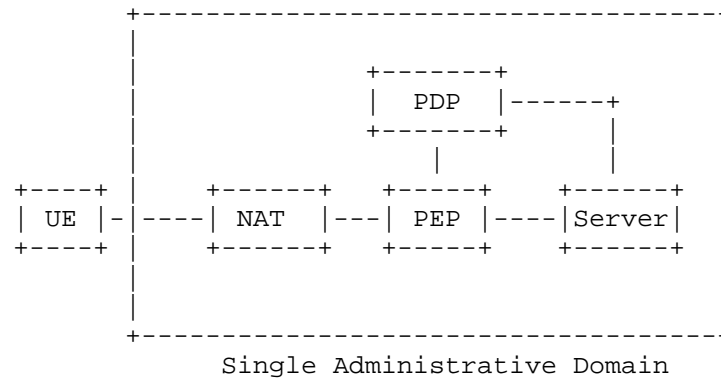


Figure 5: NAT before PEP

#### 4. PCP Applicability

This section discusses how PCP can be used to solve the problems described in Section 3.2 and Section 3.3.

##### 4.1. NAT between the PCEF and AF

A solution to solve the problem discussed in Section 3.2 is to enable a PCP Server to control the NAT and enable a PCP Client in the PCRF (see Figure 6).

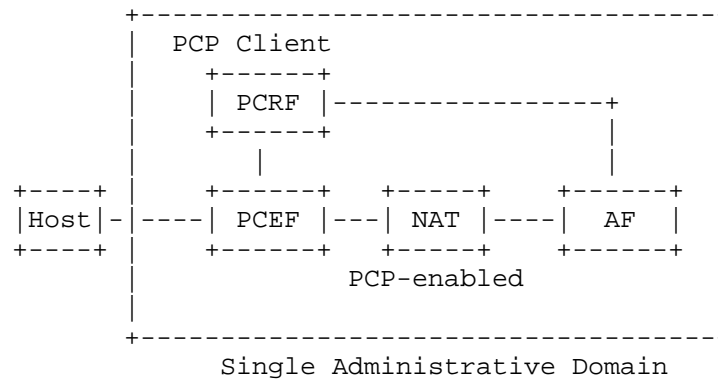


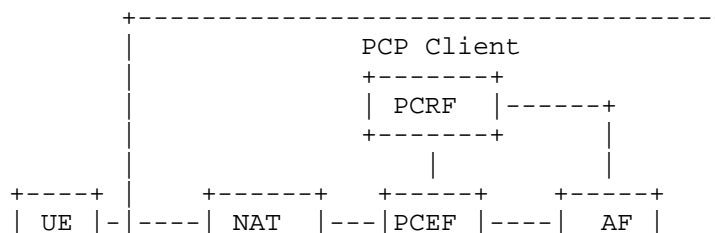
Figure 6: NAT between PCEF and AF

The updated interaction between PCRF, PCEF and AF is detailed below.

- o The PCP server controlling the NAT is configured to accept PCP requests with THIRD\_PARTY Option from authorized PCP clients (i.e., PCRF).
- o PCRF is configured with the IP address of the PCP Server.
- o The PCRF is aware of the following 5-tuple of each flow {Internal IP address of UE, Internal Port, Protocol, Remote Peer IP address, Remote Port number} learnt from PCEF. PCRF is also aware of the following 5-tuple of each flow {External IP address, External Port, Protocol, Remote Peer IP address, Remote Port number} learnt from AF.
- o The PCRF generates PCP PEER request with THIRD\_PARTY option which has Internal IP Address set to the UE's Internal IP address provided by the PCEF.
  - \* The Internal Port, Protocol, Remote Peer Port, Remote Peer IP Address fields of the PEER request are set by the PCRF according to the 5-tuple flow information provided by PCEF.
  - \* Suggested External Port and Suggested External IP Address are set to zero.
  - \* Requested Lifetime field is set to a very low value (see Section 12.3 of [RFC6887]).
- o Upon the receipt of the PEER response, the PCRF compares the External IP Address and Port learnt with the 5-tuple flow information provided by the AF to correlate external IP address/port associated with the mapping and the internal IP address/port of the flow.
- o PCRF notifies PCEF/AF to enforce relevant policies for the UE.

#### 4.2. NAT before PCEF

A solution to solve the problem discussed in Section 3.3 is to extend PCP with a new OpCode called QUERY (see Section 5).



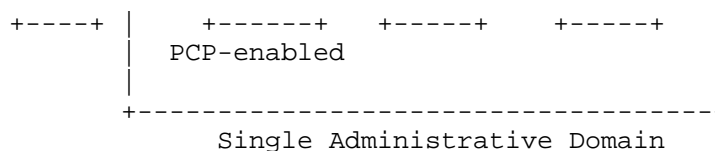
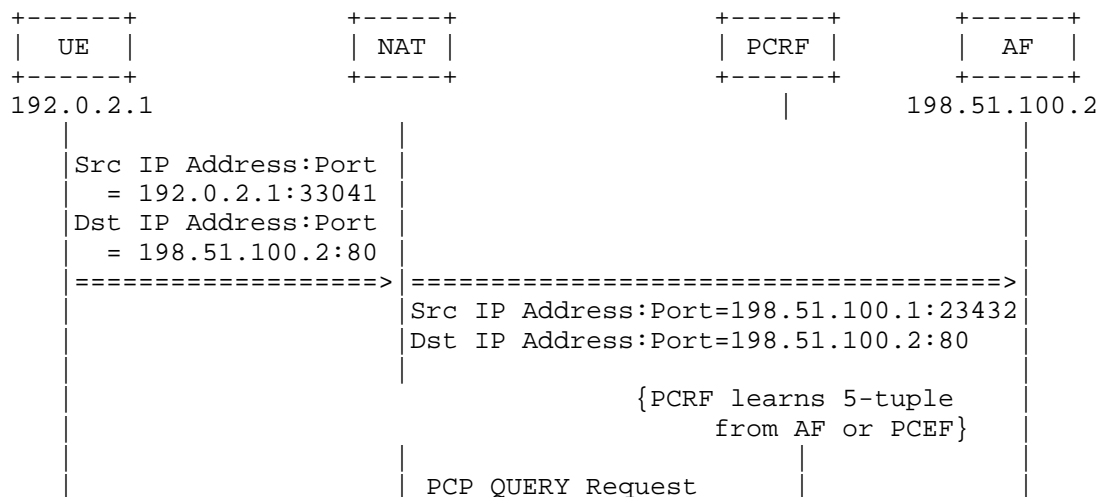


Figure 7: NAT before PCEF

The updated interaction between PCRF, PCEF and AF is detailed below:

- o The PCP server controlling the NAT is configured to accept QUERY requests Section 5 from authorized PCP clients such as PCRF. Query requests must not be received in the Internet-facing interface but from an internal interface (e.g., dedicated management interface).
- o PCRF generates a PCP QUERY request with External IP Address, External Port, Remote Peer IP address, Remote Peer Port and Protocol fields for the flow learnt from PCEF or AF.
- o PCRF learns the internal IP address and internal port number in the QUERY response. This correlation is used by the PCRF to retrieve the UE's policy to be passed to the PCEF.

Figure 8 shows an example of the use of QUERY OpCode. In this example, an HTTP connection is initiated by the UA (192.0.2.1:33041) to an HTTP server (198.51.100.2:80). The NAT assigns 198.51.100.1/23432 as external IP Address/Port. PCRF learns Internal IP Address and Port associated with the NAT mapping using PCP QUERY request/response.





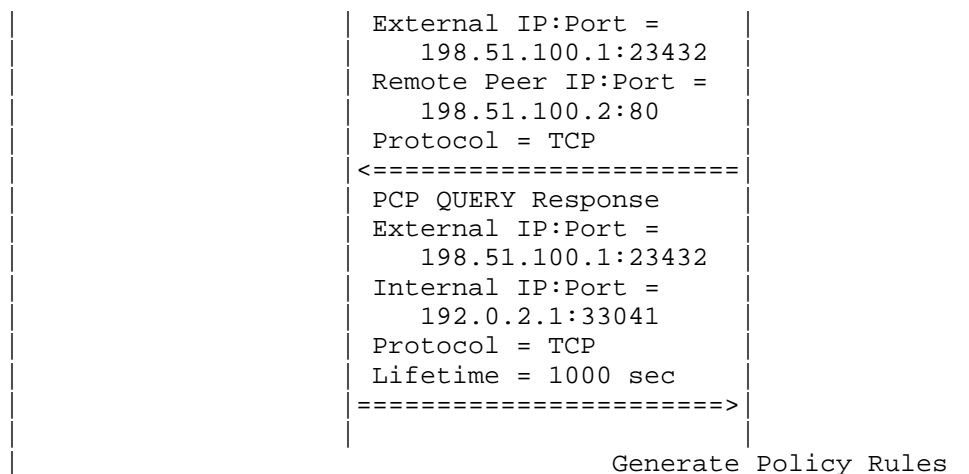


Figure 8: Usage Example

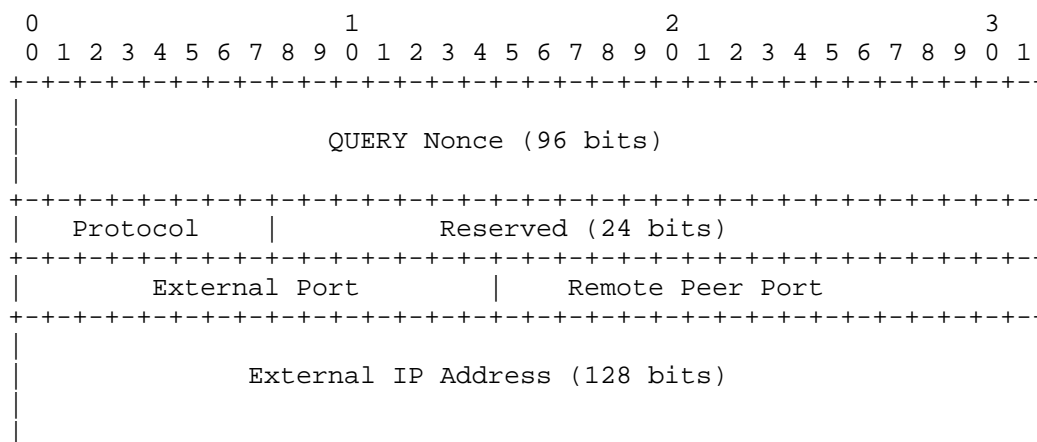
## 5. QUERY OpCode

This section defines a new PCP OpCode which can be used to query PCP-aware NAT to retrieve the Internal IP Address and Internal Port of a given mapping.

The PCP Server MUST provide a configuration option to allow administrators to enable/disable QUERY OpCode.

### 5.1. QUERY Request Format

The following diagram shows the format of the OpCode-specific information in a request for the QUERY OpCode.



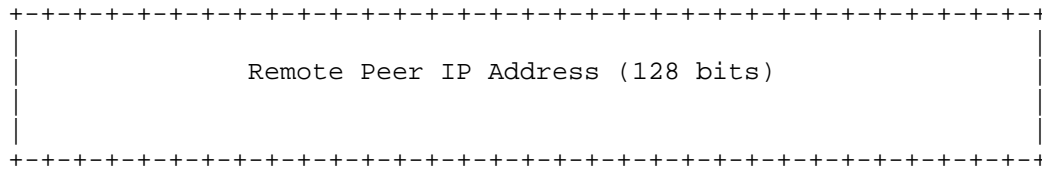


Figure 9: Query Opcode Request

These fields are described below:

Requested Lifetime (in common header): This field is positioned to 0.

Mapping Nonce: Random value chosen by the PCP client. See Section 12.2 of [RFC6887]

Protocol: Upper-layer protocol associated with this OpCode. Values are taken from the IANA protocol registry [proto\_numbers]. For example, this field contains 6 (TCP) if the OpCode is describing a TCP mapping. Protocol MUST NOT be zero.

Reserved: 24 reserved bits, MUST be set to 0 on transmission and MUST be ignored on reception.

External Port: External port allocated by NAT for the flow. External Port MUST NOT be zero

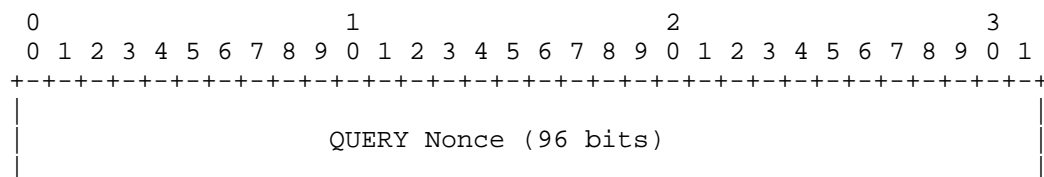
Remote Peer Port: Remote peer's port for the flow. Remote Peer Port MUST NOT be zero

External IP address: External IP address allocated by NAT for the flow. External IP address MUST NOT be zero

Remote Peer IP address: Remote peer IP address for the flow. Remote Peer IP address MUST NOT be zero.

## 5.2. QUERY Response Format

The following diagram shows the format of OpCode-specific information in a response packet for the QUERY OpCode:



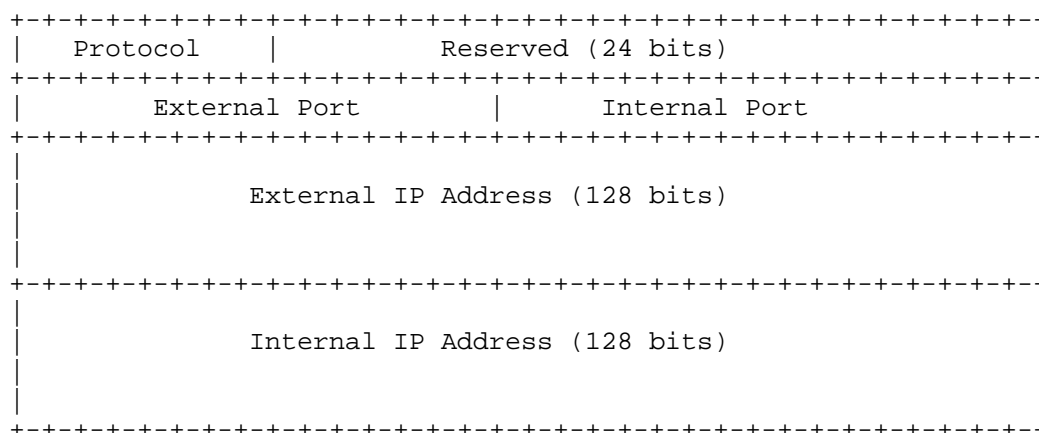


Figure 10: Query Opcode Response

These fields are described below:

Lifetime (in common header): On a success response, this indicates the lifetime for this mapping, in seconds. On an error response, this indicates that mapping does not exist.

Mapping Nonce: Copied from the request.

Protocol: Copied from the request.

Reserved: 24 reserved bits, MUST be set to 0.

External Port: Copied from the request.

External IP address: Copied from the request.

Internal Port: Internal Port as assigned by the PCP-controlled device.

Internal IP address: Internal IP address as assigned by the PCP-controlled controlled device.

### 5.3. Generating a QUERY Request

This section describes the operation of a PCP client when sending requests with the QUERY OpCode.

PCP QUERY request is used by an authorized third party PCP client that is only aware of the 5-tuple {External IP address and Port, Protocol, Remote Peer IP address and Port} and needs to learn the Internal IP address and Port associated with the NAT mapping. The request MUST contain non-zero values of Protocol, External Port, Remote Peer Port, External IP address and Remote Peer IP address. The Requested Lifetime MUST be set to zero.

#### 5.4. Processing a QUERY Request

This section describes the operation of a PCP server when processing a QUERY request.

For EIM/EIF port-mapping NAT, the processing of the QUERY request is as follows:

- o If any of the values Protocol, External Port and External IP address are equal to zero, the request is invalid and the PCP server MUST return a MALFORMED\_REQUEST to the client.
- o If Protocol, External Port and External IP address do not match any existing implicit dynamic mapping, then the PCP server MUST return NONEXIST\_MAP error response (also needed in [I-D.boucadair-pcp-failure]).
- o If Protocol, External Port and External IP address match an existing implicit dynamic mapping, then the PCP server MUST build a QUERY response with the Internal IP address, Internal Port and the lifetime associated with the mapping.

For EDM port-mapping NAT, the processing of the QUERY request is as follows:

- o If any of the values Protocol, External Port, Remote Peer Port, External IP address and Remote Peer IP Address are zero, the request is invalid and PCP server MUST return a MALFORMED\_REQUEST to the client.
- o If Protocol, External Port, Remote Peer Port, External IP address and Remote Peer IP address do not match any existing implicit dynamic mapping then the PCP server MUST return NONEXIST\_MAP error response (also needed in [I-D.boucadair-pcp-failure]).
- o If Protocol, External Port, Remote Peer Port, External IP address and Remote Peer IP address matches an existing implicit dynamic mapping then the PCP server builds a QUERY response with the Internal IP address, Internal Port and the lifetime associated with the mapping.

PCP QUERY requests received on the Internet-facing interface MUST be silently dropped.

In DS-Lite context [RFC6333], the Internal IP address returned in the QUERY response MUST be the IPv6 address of the remote tunnel endpoint and not the internal private IPv4 address.

#### 5.5. Processing a QUERY Response

After performing common PCP response processing by the PCP Client, the response is further matched with a previously-sent QUERY request by comparing the QUERY Nonce, External IP Address, External Port and Protocol. On a SUCCESS response, the PCP Client can use the Internal IP Address and Port in the QUERY response as needed.

#### 6. Applicability Scope of QUERY OpCode

The PCP-Reveal solution is designed for needs within one single administrative domain (i.e., the PCP Client and PCP Server are managed by the same entity). Considerations related to the activation of the PCP-Reveal solution in an inter-domain context is out of scope of this document.

#### 7. IANA Considerations

Authors of this document request IANA to assign the following OpCode:

- o QUERY

The following error code is requested:

- o NONEXIST\_MAP

#### 8. Security Considerations

Security considerations discussed in [RFC6887] are to be taken into account. In particular, QUERY OpCode MUST NOT be implemented or used unless the network on which the PCP QUERY messages are to be sent is fully trusted. For example if Access Control Lists (ACLs) are installed on the PCP server, and the network between the PCP client and the PCP server, so those ACLs allow only communications from a trusted PCP client to the PCP server.

QUERY OpCode may be generated by non legitimate PCP Clients; the PCP Server MUST enforce some policies such as rate limit QUERY messages. QUERY requests received from non legitimate PCP Clients are silently dropped.

PCP authentication [I-D.ietf-pcp-authentication] MAY be used.

## 9. References

### 9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.
- [proto\_numbers] IANA, , "Protocol Numbers", 2010, <<http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xml>>.

### 9.2. Informative References

- [I-D.boucadair-intarea-host-identifier-scenarios] Boucadair, M., Binet, D., Durel, S., Chatras, B., Reddy, T., and B. Williams, "Host Identification: Use Cases", draft-boucadair-intarea-host-identifier-scenarios-03 (work in progress), March 2013.
- [I-D.boucadair-pcp-failure] Boucadair, M. and R. Penno, "Analysis of Port Control Protocol (PCP) Failure Scenarios", draft-boucadair-pcp-failure-06 (work in progress), May 2013.
- [I-D.ietf-intarea-nat-reveal-analysis] Boucadair, M., Touch, J., Levis, P., and R. Penno, "Analysis of Solution Candidates to Reveal a Host Identifier (HOST\_ID) in Shared Address Deployments", draft-ietf-intarea-nat-reveal-analysis-10 (work in progress), April 2013.
- [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.
- [RFC2753] Yavatkar, R., Pendarakis, D., and R. Guerin, "A Framework for Policy-based Admission Control", RFC 2753, January 2000.

- [RFC6269] Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", RFC 6269, June 2011.
- [RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, August 2011.
- [RFC6342] Koodli, R., "Mobile Networks Considerations for IPv6 Deployment", RFC 6342, August 2011.
- [RFC6346] Bush, R., "The Address plus Port (A+P) Approach to the IPv4 Address Shortage", RFC 6346, August 2011.
- [TS.23203] 3GPP, , "Policy and charging control architecture", September 2012.

## Authors' Addresses

Mohamed Boucadair  
France Telecom  
Rennes 35000  
France

Email: mohamed.boucadair@orange.com

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

Email: tiredddy@cisco.com

Prashanth Patil  
Cisco Systems, Inc.  
Cessna Business Park, Varthur Hobli  
Sarjapur Marthalli Outer Ring Road  
Bangalore, Karnataka 560103  
India

Email: praspatti@cisco.com

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

Email: [dwing@cisco.com](mailto:dwing@cisco.com)



Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: May 29, 2016

M. Boucadair  
France Telecom  
R. Parthasarathi  
Nokia Networks  
November 26, 2015

Port Control Protocol (PCP) for SIP Deployments in Managed Networks  
draft-boucadair-pcp-sip-ipv6-07

Abstract

This document discusses how PCP (Port Control Protocol) can be used in SIP deployments in managed networks. This document applies for both IPv4 and IPv6.

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 May 29, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 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. PCP Features . . . . .	4
2.1. Learn External IP Address and Port Number . . . . .	4
2.2. Learn and Set the Lifetime of Mapping Entries . . . . .	6
2.3. Allow Unidirectional Media Flows . . . . .	6
2.4. Preserve Port Parity . . . . .	7
2.5. Preserve Port Contiguity . . . . .	7
2.6. Learn PREFIX64 . . . . .	8
2.7. Compliant with "a=rtcp" Attribute . . . . .	10
2.8. DSCP Marking Policy . . . . .	10
3. Avoid Crossing CGNs . . . . .	11
3.1. Avoid NAT64 . . . . .	11
3.2. Avoid Crossing DS-Lite AFTR . . . . .	12
4. Security Considerations . . . . .	12
5. IANA Considerations . . . . .	12
6. Acknowledgements . . . . .	12
7. References . . . . .	13
7.1. Normative References . . . . .	13
7.2. Informative References . . . . .	13
Authors' Addresses . . . . .	16

## 1. Introduction

The base Port Control Protocol (PCP, [RFC6887]) specification allows to retrieve the external IP address and external port to be conveyed in the SIP signaling messages [RFC3261]. Therefore, SIP Proxy Servers do not need to support means to ease the NAT traversal of SIP messages (e.g., [RFC5626], [RFC6223], etc.). Another advantage of using the external IP address and port is this provides a hint to the proxy server there is no need to return a small expire timer (e.g., 60s). In addition, the outbound proxy does not need any further feature to be supported in order to assist the remote endpoint to successfully establish media sessions. In particular, ALGs are not required in the NAT for this purpose and no dedicated functions at the media gateway are needed.

This document discusses how PCP can be used in SIP deployments (including IPv6 considerations).

The benefits of using PCP for SIP deployments are listed below:

- o Avoid embedding an ALG in the middleboxes. Note, ALGs are not recommended since the evolution of the service would depend on the ALG maintenance.

- o Not require any Hosted NAT Traversal function (e.g., [RFC7362]) to be embedded in the SIP server. Intermediate NATs and firewalls are transparent to the SIP service platform.
- o Avoid overloading the network with keepalive message to maintain the mapping in intermediate middleboxes.

Note, mechanisms such as STUN do not allow to discover the lifetime assigned by the middleboxes; frequent keepalive messages are therefore generated to maintain binding entries on those middleboxes. PCP is superior to those mechanisms as it allows to retrieve the assigned lifetime, and to provide hints to the middleboxes in order to decide which lifetime value is to be assigned for that particular flow.

- o Work without requiring symmetric RTP/RTCP [RFC4961].
- o Not require symmetric SIP to work (i.e., rport [RFC3581]).
- o Easily support unidirectional sessions.
- o Does not encounter issues with early media.
- o The combination of PCP and ALTC [RFC6947] allows to optimize IPv4-IPv6 interworking function resources.
- o Because there is no need for connectivity checks, session establishment delay is not impacted (pairs of ports can be pre-reserved).
- o The binding entries maintained by a flow-aware device (NAT/Firewall) can be associated with a textual description ([RFC7220]).

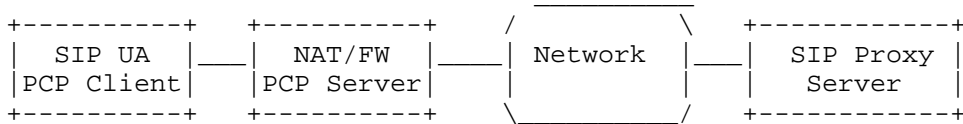
Experimentation results, including SIP flow examples, are documented in [I-D.boucadair-pcp-nat64-experiments].

In deployments where ICE [RFC5245] is required, PCP can be of great help as discussed in [I-D.penno-rtcweb-pcp] for the WebRTC case. ICE can be used in the context of SIP over WebSocket [RFC7118] and WebRTC when deployed within managed networks. Because TURN suffers from limitations in traversing NAT and firewalls over UDP, PCP is a promising solution that can complement ICE in those deployment contexts to soften the experienced high failure rate [ICEFailure].

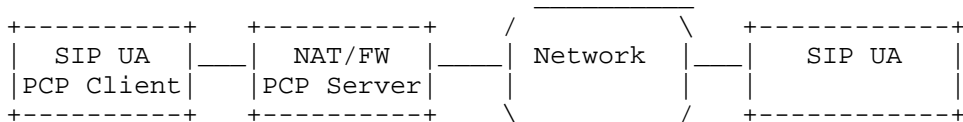
The document targets SIP deployments in managed networks. It can also be used as part of SIP-based services delivery in the context of

network-located residential gateway effort [WT-317]. Typical deployment scenarios are shown in Figure 1.

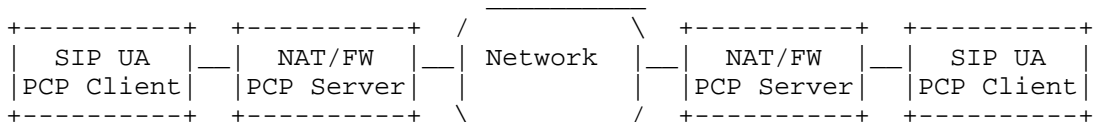
(a) SIP UA behind a NAT/FW communicating with a Proxy Server



(b) SIP UA behind a NAT/FW communicating with a remote SIP UA



(c) SIP UAs behind a NATs/FWs



(d) SIP UA behind a CPE: PCP Proxy

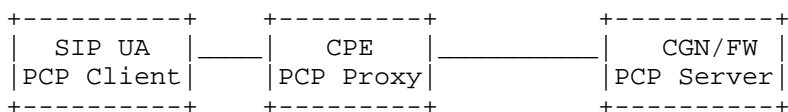


Figure 1: Typical deployment scenarios

The PCP server can be provisioned using a variety of means (e.g., [RFC7291]) or rely on the discovery method specified in [RFC6887].

This document does not make any assumption whether the PCP client is implemented as an OS service or whether it is integrated in the SIP User Agent (UA). Those considerations are implementation-service.

## 2. PCP Features

### 2.1. Learn External IP Address and Port Number

The PCP base specification allows to create mappings in PCP-controlled devices and therefore prepare for receiving incoming

packets. A SIP UA can use PCP to create one mapping for SIP signalling messages and other mappings for media session purposes.

The SIP UA uses the external IP address and port number to build SIP headers. In particular, this information is used to build the VIA and CONTACT headers.

Figure 2 shows an example of the flow exchange that occurs to retrieve the external IP address and an external IP address assigned by the NAT, while Figure 2 provides an excerpt of the SIP REGISTER message issued by the SIP UA; only the assigned IP address and port number are present in the SIP headers.

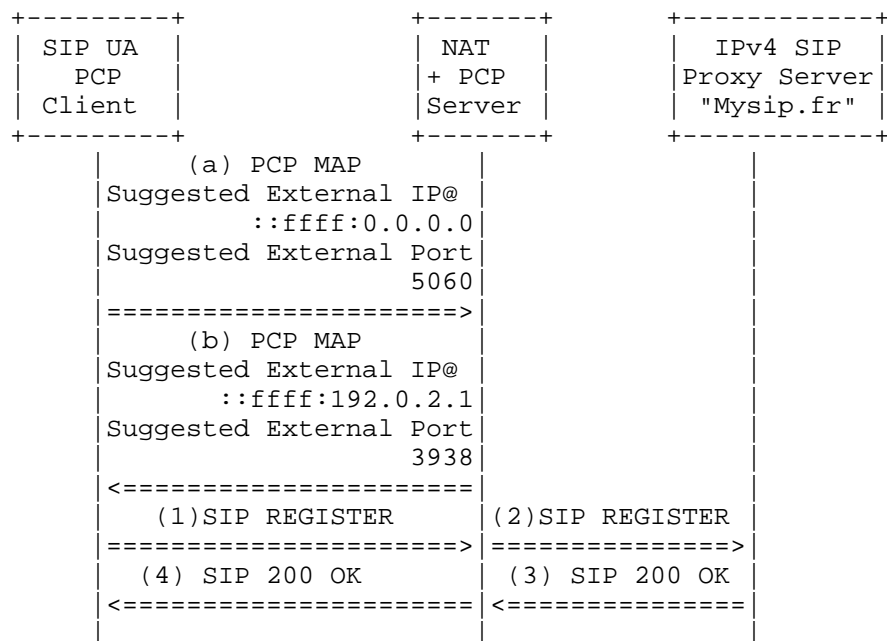


Figure 2: SIP REGISTER Call Flow

```
SIP Message:
REGISTER sip:mysip.fr SIP/2.0
Via: SIP/2.0/UDP 192.0.2.1:3938;branch=z9hG4bK1572043597
From: <sip:client4@mysip.fr:5070>;tag=893886783
To: <sip:client4@mysip.fr:5070>
Call-ID: 1271173454
CSeq: 2 REGISTER
Contact: <sip:client4@192.0.2.1:3938;line=b3433a7df33282d>
    Authorization: Digest username="client4", realm="asterisk",
    nonce="09f75e47", uri="sip:mysip.fr",
    response="826fcff4c6e84ee45fbfa52c351e6316", algorithm=MD5
Max-Forwards: 70
User-Agent: Linphone/3.4.0 (eXosip2/unknown)
Expires: 3600
```

Figure 3: Example of REGISTER messenger

The external IP address and port(s) instantiated for media streams, are used to build the SDP offer/answer. In particular, the "c" line and "m" lines.

## 2.2. Learn and Set the Lifetime of Mapping Entries

PCP allows to discover and to set the lifetime of mapping instantiated in intermediate middleboxes.

The discovery of the lifetime of a mapping avoids overloading the network and SIP servers with frequent messages. This is in particular important for cellular devices. According to [Power], the consumption of a cellular device with a keep-alive interval equal to 20 seconds (that is the default value in [RFC3948] for example) is 29 mA (2G)/34 mA (3G). This consumption is reduced to 16 mA (2G)/24 mA (3G) when the interval is increased to 40 seconds, to 9.1 mA (2G)/16 mA (3G) if the interval is equal to 150 seconds, and to 7.3 mA (2G)/14 mA (3G) if the interval is equal to 180 seconds. When no keep-alive is issued, the consumption would be 5.2 mA (2G)/6.1 mA (3G). The impact of keepalive messages would be more severe if multiple applications are issuing those messages (e.g., SIP, IPsec, etc.).

## 2.3. Allow Unidirectional Media Flows

As a consequence of instantiating mappings for media/session flows, incoming packets can be successfully forwarded to the appropriate SIP UA. Particularly, unidirectional media flows (e.g., announcement server) will be forwarded accordingly.

## 2.4. Preserve Port Parity

For deployments relying on classic RTP/RTCP odd/even port numbers assignment scheme, PORT\_SET option [I-D.ietf-pcp-port-set] can be used by a SIP UA to request port parity be preserved by the PCP server.

An example is depicted in Figure 4.

## 2.5. Preserve Port Contiguity

For deployments assuming RTCP port number can be deduced from the RTP port number, PORT\_SET option [I-D.ietf-pcp-port-set] can be used by a SIP UA to retrieve a pair of contiguous ports from the PCP server.

A flow example is shown in Figure 4.

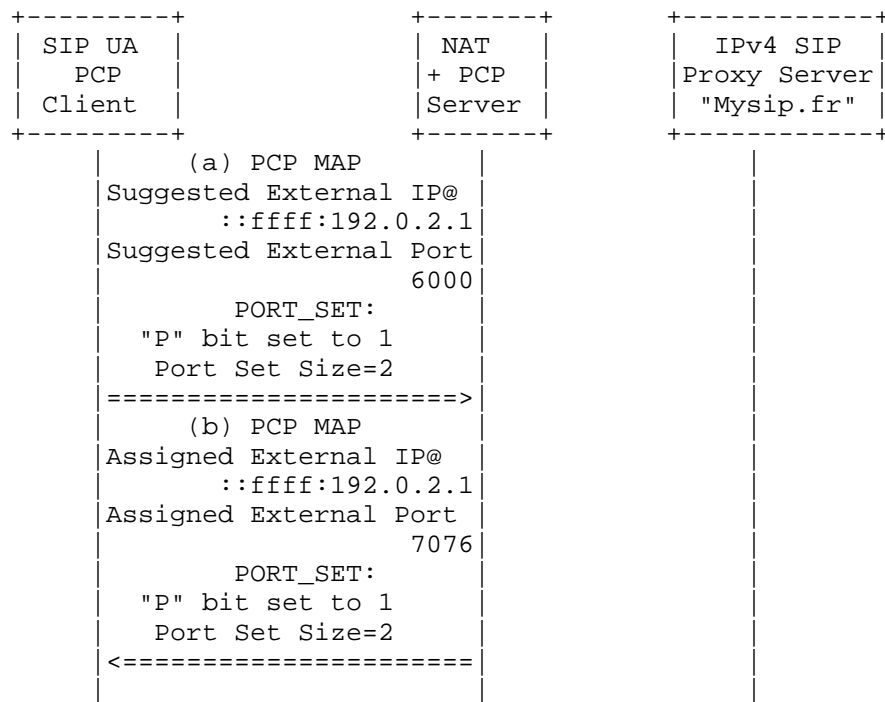


Figure 4: Retrieve a pair of ports that preserves port parity

## 2.6. Learn PREFIX64

If the SIP UA is located behind a NAT64 device [RFC6146], the option defined in [RFC7225] can be used to retrieve the PREFIX64 used by that NAT64 device.

The retrieved prefix will be used to locally build an IPv6-converted IPv4 address ([RFC6052]) corresponding to the IPv4 address included in the SDP message received from a remote IPv4-enabled SIP UA; the SDP message can be an SDP offer or an answer.



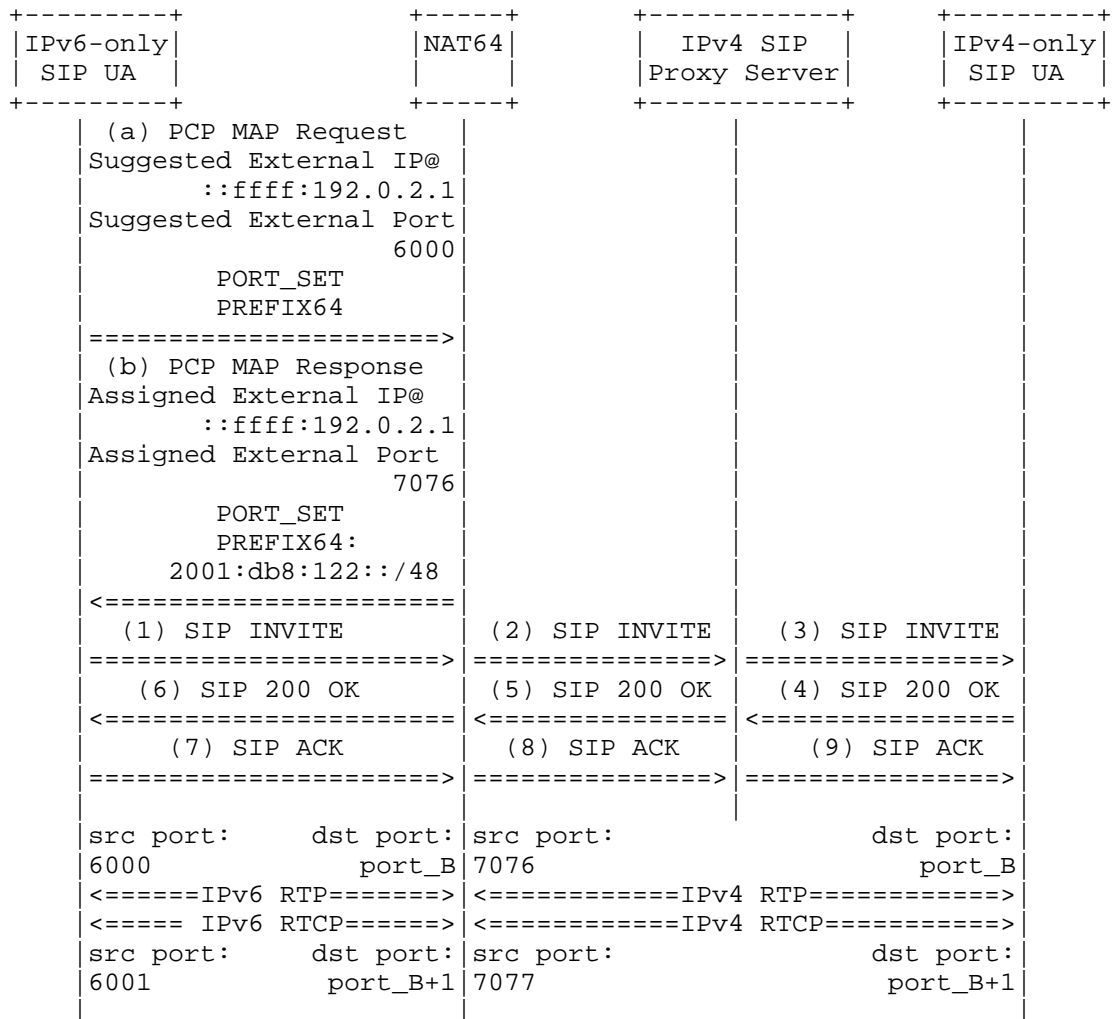


Figure 5: Example of IPv6 to IPv4 SIP-Initiated Session

Figure 6 shows the content of the SIP INVITE message sent by the IPv6-only SIP UA. This message uses the retrieved external IP address and external port numbers in SIP headers and SDP lines. This message is translated by the NAT64 without altering the SIP/SDP content.

```
INVITE sip:13@mysip.fr:5070 SIP/2.0
Via: SIP/2.0/UDP 192.0.2.1:56252;branch=z9hG4bK1876803184
From: <sip:client4@mysip.fr:5070>;tag=631384602
To: <sip:13@mysip.fr:5070> Call-ID: 1377792765 CSeq: 21 INVITE
Contact: <sip:client4@192.0.2.1:56252>
Authorization: Digest username="client4", realm="asterisk",
  nonce="3358d80b", uri="sip:13@mysip.fr:5070",
  response="41442e94f6610e6f383a355albdf3e48", algorithm=MD5
Content-Type: application/sdp Allow: INVITE, ACK, CANCEL, OPTIONS,
  BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO
Max-Forwards: 70
User-Agent: Linphone/3.4.0 (eXosip2/unknown)
Subject: Phone call Content-Length: 443

v=0
o=client4 2487 2487 IN IP4 192.0.2.1
s=Talk c=IN IP4 192.0.2.1
b=AS:256
t=0 0
m=audio 7076 RTP/AVP 111 110 3 101
a=rtpmap:111 speex/16000
a=fmtp:111 vbr=on a=rtpmap:110 speex/8000
a=fmtp:110 vbr=on a=rtpmap:3 GSM/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-11
m=video 9076 RTP/AVP 102 99
a=rtpmap:102 H264/90000
a=fmtp:102 profile-level-id=428014
a=rtpmap:99 MP4V-ES/90000
a=fmtp:99
profile-level-id=3
```

Figure 6: Content of the INVITE message

## 2.7. Compliant with "a=rtcp" Attribute

The base PCP specification can be used to retrieve the port number to be singled if "a=rtcp" attribute is in use [RFC3550].

## 2.8. DSCP Marking Policy

PCP can be used to discover the DSCP value to be used when sending real-time flows or to create a mapping that matches a DSCP marking. This can be achieved using the DSCP option defined in [I-D.boucadair-pcp-extensions]. DSCP setting value is configured by the network and not the SIP UA.

This feature can be used as an input for DSCP marking in some deployments such as [I-D.ietf-tsvwg-rtcweb-qos].

### 3. Avoid Crossing CGNs

#### 3.1. Avoid NAT64

Because an IPv6-only SIP UA is not aware of the connectivity capabilities of the remote UA, the IPv6-only SIP UA uses the ALTC attribute [RFC6947] to signal the assigned IPv6 address and the IPv4 address learned via PCP.

If the remote SIP UA is IPv6-enabled, IPv6 transfer capabilities will be used to place the session. If the remote SIP UA is IPv4-only, IPv4 transfer capabilities will be used. NAT64 devices will be crossed only if the remote UA is IPv4-only.

Figure 7 provides an except of a SIP INVITE message that encloses both the local IPv6 address and the IPv4 address/port number assigned by a NAT64 device.

```
INVITE sip:13@mysip.fr:5070 SIP/2.0
Via: SIP/2.0/UDP 192.0.2.1:35011;branch=z9hG4bK702695557
From: <sip:client4@mysip.fr:5070>;tag=641336337
To: <sip:13@mysip.fr:5070>
Call-ID: 1532307201
CSeq: 20 INVITE
Contact: <sip:client4@192.0.2.1:35011>
Content-Type: application/sdp
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY,
      MESSAGE, SUBSCRIBE, INFO
Max-Forwards: 70
User-Agent: Linphone/3.4.0 (eXosip2/unknown)
Subject: Phone call
Content-Length: 538

v=0
o=client4 3867 3867 IN IP4 192.0.2.1
s=Talk
c=IN IP4 192.0.2.1
b=AS:256
t=0 0
m=audio 7056 RTP/AVP 111 110 3 101
a=altc:1 IP6 2001:db8:1f94:3000:6c73:ea54:cef:2730 45678
a=altc:2 IP4 192.0.2.1 7056
```

Figure 7: Content of the INVITE message (with ALTC Attribute)

### 3.2. Avoid Crossing DS-Lite AFTR

SIP UAs co-located with the B4 [RFC6333] or located behind the CPE can behave as dual-stack UAs:

- o Native IPv6 address is assigned locally.
- o The external IPv4 address and port is retrieved using PCP.

To avoid unnecessary invocation of AFTR resources, ALTC attribute is used to signal both IPv4 and IPv6 addresses. If the remote SIP UA is IPv6-enabled, IPv6 transfer capabilities will be used to place the session (i.e., the flows will avoid crossing the DS-Lite AFTR device). If the remote SIP UA is IPv4-only, IPv4 transfer capabilities will be used. AFTR devices will be crossed only if the remote UA is IPv4-only.

## 4. Security Considerations

PCP-related security considerations are discussed in [RFC6887].

Security considerations related to the discovery of PREFIX64 are discussed in Section 7 of [RFC7225] and those related to retrieving a set of ports are discussed in Section 7 of [I-D.ietf-pcp-port-set].

An attacker that wants to intercept media flows, without requiring intercepting SIP signalling message, can insert a fake PCP server that will influence the content of SIP messages so that an illegitimate node is inserted in the media path. Such behavior is not desirable. Means to prevent the PCP client from discovering illegitimate PCP servers must be enforced. Within the context of this document, the network on which the PCP messages are to be sent is fully trusted. For example, access control lists (ACLs) can be installed on the PCP client, PCP server, and the network between them, so those ACLs allow only communications from a trusted PCP client to the PCP server.

## 5. IANA Considerations

This document does not require any action from IANA.

## 6. Acknowledgements

Many thanks for T. Reddy and S. Kiesel for their review.

## 7. References

### 7.1. Normative References

- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <<http://www.rfc-editor.org/info/rfc3261>>.
- [RFC3581] Rosenberg, J. and H. Schulzrinne, "An Extension to the Session Initiation Protocol (SIP) for Symmetric Response Routing", RFC 3581, DOI 10.17487/RFC3581, August 2003, <<http://www.rfc-editor.org/info/rfc3581>>.
- [RFC6887] Wing, D., Ed., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, DOI 10.17487/RFC6887, April 2013, <<http://www.rfc-editor.org/info/rfc6887>>.

### 7.2. Informative References

- [I-D.boucadair-pcp-extensions]  
Boucadair, M., Penno, R., and D. Wing, "Some Extensions to Port Control Protocol (PCP)", draft-boucadair-pcp-extensions-03 (work in progress), April 2012.
- [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.ietf-pcp-port-set]  
Qiong, 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-13 (work in progress), October 2015.
- [I-D.ietf-tsvwg-rtcweb-qos]  
Dhesikan, S., Jennings, C., Druta, D., and P. Jones, "DSCP and other packet markings for WebRTC QoS", draft-ietf-tsvwg-rtcweb-qos-05 (work in progress), October 2015.
- [I-D.penno-rtcweb-pcp]  
Penno, R., Reddy, T., Wing, D., and M. Boucadair, "PCP Considerations for WebRTC Usage", draft-penno-rtcweb-pcp-00 (work in progress), May 2013.

- [ICEFailure] Telemetry Dashboard, "WEBRTC\_ICE\_SUCCESS\_RATE", March 2015, <[http://telemetry.mozilla.org/#filter=beta%2F36%2FWEBrTC\\_ICE\\_SUCCESS\\_RATE%2Fsaved\\_session%2FFirefox&aggregates=multiselect-all!Submissions&evoOver=Builds&locked=true&sanitize=true&renderhistogram=Graph](http://telemetry.mozilla.org/#filter=beta%2F36%2FWEBrTC_ICE_SUCCESS_RATE%2Fsaved_session%2FFirefox&aggregates=multiselect-all!Submissions&evoOver=Builds&locked=true&sanitize=true&renderhistogram=Graph)>.
- [Power] Haverinen, H., Siren, J., and P. Eronen, "Energy Consumption of Always-On Applications in WCDMA Networks", April 2007, <<http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4212635>>.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <<http://www.rfc-editor.org/info/rfc3550>>.
- [RFC3948] Huttunen, A., Swander, B., Volpe, V., DiBurro, L., and M. Stenberg, "UDP Encapsulation of IPsec ESP Packets", RFC 3948, DOI 10.17487/RFC3948, January 2005, <<http://www.rfc-editor.org/info/rfc3948>>.
- [RFC4961] Wing, D., "Symmetric RTP / RTP Control Protocol (RTCP)", BCP 131, RFC 4961, DOI 10.17487/RFC4961, July 2007, <<http://www.rfc-editor.org/info/rfc4961>>.
- [RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, DOI 10.17487/RFC5245, April 2010, <<http://www.rfc-editor.org/info/rfc5245>>.
- [RFC5626] Jennings, C., Ed., Mahy, R., Ed., and F. Audet, Ed., "Managing Client-Initiated Connections in the Session Initiation Protocol (SIP)", RFC 5626, DOI 10.17487/RFC5626, October 2009, <<http://www.rfc-editor.org/info/rfc5626>>.
- [RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, DOI 10.17487/RFC6052, October 2010, <<http://www.rfc-editor.org/info/rfc6052>>.
- [RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, DOI 10.17487/RFC6146, April 2011, <<http://www.rfc-editor.org/info/rfc6146>>.

- [RFC6223] Holmberg, C., "Indication of Support for Keep-Alive", RFC 6223, DOI 10.17487/RFC6223, April 2011, <<http://www.rfc-editor.org/info/rfc6223>>.
- [RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, DOI 10.17487/RFC6333, August 2011, <<http://www.rfc-editor.org/info/rfc6333>>.
- [RFC6947] Boucadair, M., Kaplan, H., Gilman, R., and S. Veikkolainen, "The Session Description Protocol (SDP) Alternate Connectivity (ALTC) Attribute", RFC 6947, DOI 10.17487/RFC6947, May 2013, <<http://www.rfc-editor.org/info/rfc6947>>.
- [RFC7118] Baz Castillo, I., Millan Villegas, J., and V. Pascual, "The WebSocket Protocol as a Transport for the Session Initiation Protocol (SIP)", RFC 7118, DOI 10.17487/RFC7118, January 2014, <<http://www.rfc-editor.org/info/rfc7118>>.
- [RFC7220] Boucadair, M., Penno, R., and D. Wing, "Description Option for the Port Control Protocol (PCP)", RFC 7220, DOI 10.17487/RFC7220, May 2014, <<http://www.rfc-editor.org/info/rfc7220>>.
- [RFC7225] Boucadair, M., "Discovering NAT64 IPv6 Prefixes Using the Port Control Protocol (PCP)", RFC 7225, DOI 10.17487/RFC7225, May 2014, <<http://www.rfc-editor.org/info/rfc7225>>.
- [RFC7291] Boucadair, M., Penno, R., and D. Wing, "DHCP Options for the Port Control Protocol (PCP)", RFC 7291, DOI 10.17487/RFC7291, July 2014, <<http://www.rfc-editor.org/info/rfc7291>>.
- [RFC7362] Ivov, E., Kaplan, H., and D. Wing, "Latching: Hosted NAT Traversal (HNT) for Media in Real-Time Communication", RFC 7362, DOI 10.17487/RFC7362, September 2014, <<http://www.rfc-editor.org/info/rfc7362>>.
- [WT-317] Broadband Forum, "Network Enhanced Residential Gateway (NERG)", 2015, <<https://www.broadband-forum.org/technical/technicalwip.php>>.

Authors' Addresses

Mohamed Boucadair  
France Telecom  
Rennes 35000  
France

Email: mohamed.boucadair@orange.com

Parthasarathi Ravindran  
Nokia Networks  
Manyata Embassy Business park  
Bangalore, Karnataka 560045  
India

Email: partha@parthasarathi.co.in



PCP working group  
Internet-Draft  
Updates: 6887 (if approved)  
Intended status: Standards Track  
Expires: April 24, 2014

S. Cheshire  
Apple  
S. Perreault  
Viagenie  
October 21, 2013

Updates to the PCP Specification  
draft-cheshire-pcp-unsupp-family-06

Abstract

The Port Control Protocol (PCP) allows clients to request mappings in NAT gateways and firewalls. This document specifies the PCP UNSUPP\_FAMILY error code, which enables PCP servers to inform clients when the requested external address family is not supported. This document also removes the requirement for the PCP server to validate the mapping nonce, which proved to be unhelpful in practice.

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 April 24, 2014.

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 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.

## 1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

## 2. PCP Unsupported Family Error

Port Control Protocol [RFC6887] MAP requests allow clients to request inbound mappings in NAT gateways and firewalls. A client can request a MAP mapping to an external IPv6 address or to an external IPv4 address. The client signifies which family of external address it desires by the type of address it puts into the Suggested External Address field.

If the client wants an external IPv6 address, then it populates the Suggested External Address field with a native IPv6 address. In the overwhelmingly common case where the client doesn't know the external address when it makes its initial request, this will be the all-zeros IPv6 address (:::).

If the client wants an external IPv4 address, then it populates the Suggested External Address field with an IPv4-mapped IPv6 address (the first 80 bits set to zero, the next 16 set to one). In the overwhelmingly common case where the client doesn't know the NAT's external address when it makes its initial request, this will be the all-zeros IPv4 address (::ffff:0:0).

The PCP specification [RFC6887] is somewhat vague about whether the address family is a firm requirement, or merely a hint that the PCP server is free to ignore. This update clarifies that issue: The specific address placed in the Suggested External Address field is merely a suggestion that the PCP server is free to ignore, but the address family is not. If the specific suggested address cannot be provided, another address of the same family SHOULD be provided if possible, but if the suggested address \*family\* cannot be provided by this PCP server, it MUST return a PCP error reply containing the UNSUPP\_FAMILY error code.

Many gateway devices, particularly early ones, may not be able to provide both external address families. For example, an IPv4-only NAT cannot provide an external IPv6 address.

Even with gateway devices that can support both external address families, the ability to provide an external address of the requested family may depend on the family of the client's internal address. For example, a gateway that supports native IPv6, and traditional NAT44, but not NAT64, can provide mappings from an internal IPv6 address to an external IPv6 address (typically the same address when no address translation is being performed), and can provide mappings from an internal IPv4 address to an external IPv4 address, but not mappings from an internal IPv6 address to an external IPv4 address. When such a gateway receives a request to map an internal IPv6 address to an external IPv4 address it MUST return the UNSUPP\_FAMILY error code.

Note that it is possible and valid for a given internal address and port to have two mappings simultaneously, one to an external IPv4 address and one to an external IPv6 address. The handling of outbound packets is determined by the outbound destination address; for example, an outbound IPv6 packet addressed to an IPv6 address in the NAT64 gateway's IPv6 address pool is translated to the corresponding IPv4 packet before forwarding; an outbound IPv6 packet addressed to some other routable IPv6 address is forwarded unmodified.

A client that can handle both IPv6 and IPv4 external addresses MAY send two requests, and then determine its behavior based on the responses it receives. For example, if the client requests and receives an IPv6 external address, it might create a DNS AAAA record giving that IPv6 address. If the client requests and receives an IPv4 external address, it might create a DNS address record giving that IPv4 address. If the client requests and receives both families of external address, it might create both DNS records. Or, if one external address is sufficient for the client, then it MAY first request its preferred address family, and only if that fails with an UNSUPP\_FAMILY error, request the other family.

## 2.1. Implications for RFC 6887

Various sections of the PCP specification [RFC6887] describe clients and servers identifying a MAP mapping by examining the three-tuple of { protocol, internal address, internal port } in a request or reply. For example:

If the internal port, protocol, and internal address match an existing static mapping (which will have no nonce), then a PCP reply is sent giving the external address and port of that static mapping, using the nonce from the PCP request. The server does not record the nonce.

It is possible that a mapping might already exist for a requested internal address, protocol, and port. If so, the PCP server takes the following actions...

If no mapping exists for the internal address, protocol, and port, and the PCP server is able to create a mapping using the suggested external address and port, it SHOULD do so.

After performing common PCP response processing, the response is further matched with a previously sent MAP request by comparing the internal IP address (the destination IP address of the PCP response, or other IP address specified via the THIRD\_PARTY option), the protocol, the internal port, and the mapping nonce. Other fields are not compared, because the PCP server sets those fields.

Everywhere that the PCP specification [RFC6887] refers to using the "protocol, internal address, and internal port," to identify a particular inbound mapping, it should be read to mean the four-tuple of { protocol, internal address, internal port, external address family }.

PCP clients and servers that only support one external address family can continue to use the previous three-tuple { protocol, internal address, internal port } to identify inbound mappings, since they only support one external address family, and unilaterally reject MAP requests and responses containing the unsupported family. For PCP servers this means rejecting MAP requests containing the unsupported address family via the UNSUPP\_FAMILY error code. For PCP clients this should be a non-issue because a PCP client should never receive a reply containing an external address family it didn't request, but should a client receive such a reply from a misbehaving PCP server offering an external address family the client did not request, the client MUST silently ignore the erroneous reply.

An implication of this update to the PCP specification is that when renewing a MAP mapping, a PCP client **MUST** include a suggested external address of the correct family, so that the gateway device can identify which mapping is being renewed. Ideally a PCP client **SHOULD** record the previously-granted external address and use that as the suggested external address in its renewal request, to facilitate recovery in the event of gateway state loss, but at the very least a PCP client **MUST** provide an all-zeroes suggested external address of the correct family (just as it must have indicated the desired address family in its initial request that created the mapping).

These considerations apply only to MAP requests. With PEER requests, the five-tuple of { protocol, internal address, internal port, remote peer address, remote peer port } uniquely identifies the intended mapping. When technologies like NAT64 are used the external address family need not be the same as the remote peer address family, but the external address family is still uniquely determined by the remote peer address, and does not need to be specified separately.

### 3. New Nonce Check Behavior

The PCP specification [RFC6887] states that if a client requests a mapping (or renews a mapping, which is the same thing, from the server's point of view) and the requested mapping already exists, but with a different nonce, then the server returns a NOT\_AUTHORIZED error.

This has proved to be problematic. The nonce exists to guard against off-path attackers. It helps a client have confidence that the PCP responses it receives are really from the server that processed its PCP request. And it helps a PCP server validate that a client requesting a mapping is the same client that previously requested a mapping for that internal address and port. In some circumstances a legitimate client may not know the correct nonce to renew its own mappings.

For example, if a host reboots or otherwise suffers a loss of state, it may not have a record of nonces it previously used. Suppose this host then requests a mapping from an external IPv4 address to its internal IP address at TCP port 22, so that it can receive ssh logins. If the same internal host had previously requested such a mapping using a different nonce, then the new request will fail with a NOT\_AUTHORIZED error. This is unhelpful and misleading. The client does in fact have a mapping. Incoming connection requests to its external address and port will in fact be forwarded to it at port 22. The PCP server is simply refusing to tell the client what the external address and port are, hindering the client's ability to use the mapping that it actually already has.

The same scenario also exists in the case where (i) a different internal host had previously requested a mapping to its internal port 22, (ii) that host then left the network, and (iii) the newly vacated internal IP address is then assigned to new host. When this happens, the new host will be unable to usefully request a mapping to its internal port 22 until the old mapping expires, or is deleted through some other means (e.g. via the DHCP server informing the PCP server that the IP address has been reassigned, or via manual intervention by an administrator, or via some other out-of-band mechanism). Note that the new host will actually have a working mapping to its internal port 22, and will actually receive incoming connection requests arriving at the external address and port, but the PCP server will refuse to tell the client what the external address and port are, thereby hindering the new host from communicating that external address and port to the peer it wishes to receive connections from. This is not helpful.

This PCP security check does not prevent the new host from learning

the external address and port by other circuitous means. For example, the new host could discover the external address and port by sending outbound traffic a destination it controls, and having that destination report back the source address and port.

Furthermore, this PCP security check is inconsistent with other PCP behavior. It makes PCP behave differently for explicit dynamic versus other kinds of mappings. Indeed, requests matching static mappings are not subjected to the nonce check and will result in a response containing the static mapping's current state. There is no reason that MAP requests matching a dynamic mapping should return less information.

Therefore, the nonce check behavior described below **MUST** be implemented instead.

### 3.1. Nonce Check for MAP Requests

If operating in the Simple Threat Model (Section 18.1 of the PCP specification [RFC6887]), and the internal port, protocol, internal address, and external address family match an existing explicit dynamic mapping, but the mapping nonce does not match, then the existing mapping is not modified in any way, and a valid PCP reply is returned to the client, using the client-specified nonce, reporting the external address, port, and remaining lifetime of the existing mapping.

This specification makes no statement about mapping nonce with the Advanced Threat Model.

### 3.2. Nonce Check for PEER Requests

If operating in the Simple Threat Model (Section 18.1 of the PCP specification [RFC6887]), and the protocol, internal address, internal port, remote peer address, and remote peer port match a mapping that already exists, but the mapping nonce does not match (that is, a previous PEER request was processed), then the existing mapping is not modified in any way, and a valid PCP reply is returned to the client, using the client-specified nonce, reporting the external address, port, and remaining lifetime of the existing mapping.

This specification makes no statement about mapping nonce with the Advanced Threat Model.

### 3.3. Returning NOT\_AUTHORIZED error

A NOT\_AUTHORIZED error should still be returned, as described in Section 15.1 of the PCP specification [RFC6887], when a PCP client attempts to delete a static mapping (i.e., a mapping created outside of PCP itself) or an outbound (implicit or PEER-created) mapping.

### 3.4. Discussion

The behavior described above in Sections 3.1 and 3.2 is what is currently being considered by the working group. An implication of this behavior is that if a client forgets its previous nonce (through reboot or similar lost of state), then when it tries to recreate its previous mappings, it will learn about its existing mappings, but it will be unable to extend their lifetimes. This means that a mapping with a one-hour lifetime will be renewed after roughly half an hour, at which point its remaining lifetime will be about half an hour. It will then be renewed after roughly fifteen minutes, then seven minutes, then three minutes, and so on, increasingly rapidly, until the old mapping finally expires and is immediately replaced with a new one with a new nonce.

The lower limit on the retry interval of four seconds implies that after a mapping expires, there will be a window of up to four seconds where no mapping exists, before the legitimate client re-tries its request and recreates the intended mapping (this time with the new nonce).

As an alternative to returning the current port and lifetime information about the mapping, the PCP server could instead return a NOT\_AUTHORIZED error. However, were the PCP server to do this, the user is likely to perceive the gateway as "broken" and power-cycle it to fix the problem. Such forced reboot would clear out NAT state, thereby allowing a subsequent request to succeed, thereby (apparently) solving the problem. A pattern of habitual rebooting of the gateway to make it work gives the impression that the software is buggy and unreliable, and does not result in a positive user experience.



#### 4. IANA Considerations

IANA should allocate the following PCP Result Code:

14 UNSUPP\_FAMILY: Unsupported external address family, e.g., IPv6 in a NAT that handles only IPv4. This is a long lifetime error.

#### 5. Security Considerations

The UNSUPP\_FAMILY error code leaks no sensitive information and creates no new security vulnerabilities.

Allowing a client to learn the parameters of an existing mapping without knowing the mapping nonce used to create it could leak mapping information to an on-path attacker.

Having the PCP server refuse to renew or delete mappings if the request nonce doesn't match the existing nonce allows an off-path attacker to preemptively poison a NAT gateway with bogus mappings, which the legitimate holder of the internal address will then be unable to renew or delete because it doesn't know the nonce the attacker used when creating the bogus mappings.

#### 6. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

#### Authors' Addresses

Stuart Cheshire  
Apple Inc.  
1 Infinite Loop  
Cupertino, California 95014  
USA

Phone: +1 408 974 3207  
Email: cheshire@apple.com

Simon Perreault  
Viagenie  
246 Aberdeen  
Quebec, QC G1R 2E1  
Canada

Phone: +1 418 656 9254  
Email: [simon.perreault@viagenie.ca](mailto:simon.perreault@viagenie.ca)  
URI: <http://viagenie.ca>



PCP Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: October 16, 2014

M. Boucadair  
France Telecom  
R. Penno  
D. Wing  
Cisco  
April 14, 2014

DHCP Options for the Port Control Protocol (PCP)  
draft-ietf-pcp-dhcp-13

Abstract

This document specifies DHCP (IPv4 and IPv6) options to configure hosts with Port Control Protocol (PCP) server IP addresses. The use of DHCPv4 or DHCPv6 depends on the PCP deployment scenarios. The set of deployment scenarios to which use of DHCPv4 or DHCPv6 apply are outside the scope of this document.

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

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 October 16, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 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. Terminology . . . . .	3
3. DHCPv6 PCP Server Option . . . . .	3
3.1. Format . . . . .	3
3.2. DHCPv6 Client Behavior . . . . .	4
4. DHCPv4 PCP Option . . . . .	5
4.1. Format . . . . .	5
4.2. DHCPv4 Client Behavior . . . . .	6
5. DHCP Server Configuration Guidelines . . . . .	6
6. Dual-Stack Hosts . . . . .	8
7. Hosts with Multiple Interfaces . . . . .	8
8. Security Considerations . . . . .	8
9. IANA Considerations . . . . .	8
9.1. DHCPv6 Option . . . . .	8
9.2. DHCPv4 Option . . . . .	8
10. Acknowledgements . . . . .	9
11. References . . . . .	9
11.1. Normative References . . . . .	9
11.2. Informative References . . . . .	10
Authors' Addresses . . . . .	10

## 1. Introduction

This document defines DHCPv4 [RFC2131] and DHCPv6 [RFC3315] options that can be used to configure hosts with PCP server [RFC6887] IP addresses.

This specification assumes a PCP server is reachable with one or multiple IP addresses. As such, a list of IP addresses can be returned in the DHCP PCP server option.

This specification allows returning one or multiple lists of PCP server IP addresses. This is used as a hint to guide the PCP client when determining whether to send PCP requests to one or multiple PCP servers. Concretely, the PCP client needs an indication to decide whether entries need to be instantiated in all PCP servers (e.g.,

multi-homing, multiple PCP-controlled devices providing distinct services , etc.) or using one IP address from the list (e.g., redundancy group scenario, proxy-based model, etc.). Refer to [I-D.boucadair-pcp-deployment-cases] for a discussion on PCP deployment scenarios.

For guidelines on how a PCP client can use multiple IP addresses and multiple PCP servers, see [I-D.ietf-pcp-server-selection].

## 2. Terminology

This document makes use of the following terms:

- o PCP server denotes a functional element that receives and processes PCP requests from a PCP client. A PCP server can be co-located with or be separated from the function (e.g., NAT, Firewall) it controls. Refer to [RFC6887].
- o PCP client denotes a PCP software instance responsible for issuing PCP requests to a PCP server. Refer to [RFC6887].
- o DHCP refers to both DHCPv4 [RFC2131] and DHCPv6 [RFC3315].
- o DHCP client denotes a node that initiates requests to obtain configuration parameters from one or more DHCP servers.
- o DHCP server refers to a node that responds to requests from DHCP clients.

## 3. DHCPv6 PCP Server Option

### 3.1. Format

The DHCPv6 PCP server option can be used to configure a list of IPv6 addresses of a PCP server.

The format of this option is shown in Figure 1.

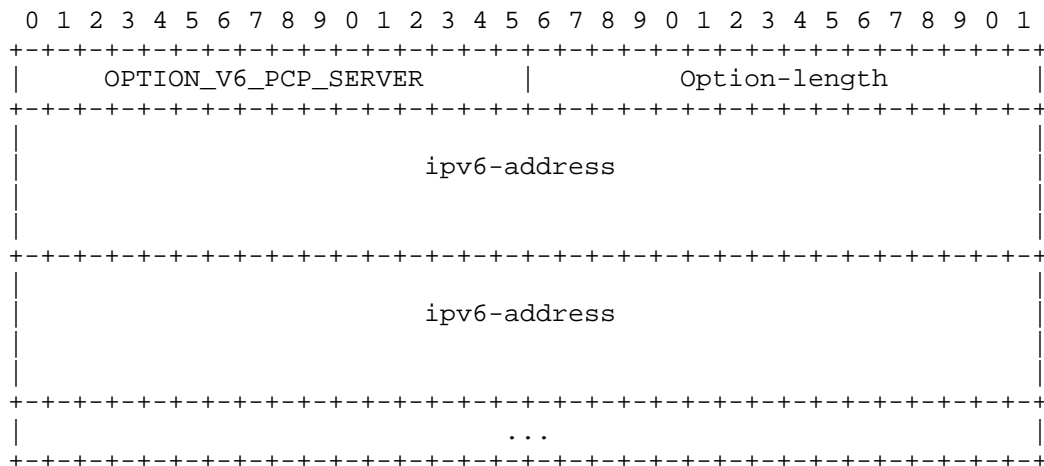


Figure 1: DHCPv6 PCP server option

The fields of the option shown in Figure 1 are as follows:

- o Option-code: OPTION\_V6\_PCP\_SERVER (TBA, see Section 9.1)
- o Option-length: Length of the 'PCP server IP Address(es)' field in octets. MUST be a multiple of 16.
- o PCP server IPv6 Addresses: Includes one or more IPv6 addresses [RFC4291] of the PCP server to be used by the PCP client. Note, IPv4-mapped IPv6 addresses (Section 2.5.5.2 of [RFC4291]) are allowed to be included in this option.

To return more than one PCP server to the DHCPv6 client (as opposed to more than one address for a single PCP server), the DHCPv6 server returns multiple instances of `OPTION_V6_PCP_SERVER`.

### 3.2. DHCPv6 Client Behavior

To discover one or more PCP servers, the DHCPv6 client requests PCP server IP addresses by including `OPTION_V6_PCP_SERVER` in an Option Request Option (ORO), as described in Section 22.7 of [RFC3315].

The DHCPv6 client MUST be prepared to receive multiple instances of OPTION\_V6\_PCP\_SERVER; each instance is to be treated as a separate PCP server.

If an IPv4-mapped IPv6 address is received in `OPTION_V6_PCP_SERVER`, it indicates that the PCP server has the corresponding IPv4 address.

Note: When presented with the IPv4-mapped prefix, current versions of Windows and Mac OS generate IPv4 packets, but will not send IPv6 packets [RFC6052]. Representing IPv4 addresses as IPv4-mapped IPv6 addresses follows the same logic as in section 5 of [RFC6887].

The DHCPv6 client MUST silently discard multicast and host loopback addresses [RFC6890] conveyed in OPTION\_V6\_PCP\_SERVER.

#### 4. DHCPv4 PCP Option

##### 4.1. Format

The DHCPv4 PCP server option can be used to configure a list of IPv4 addresses of a PCP server. The format of this option is illustrated in Figure 2.

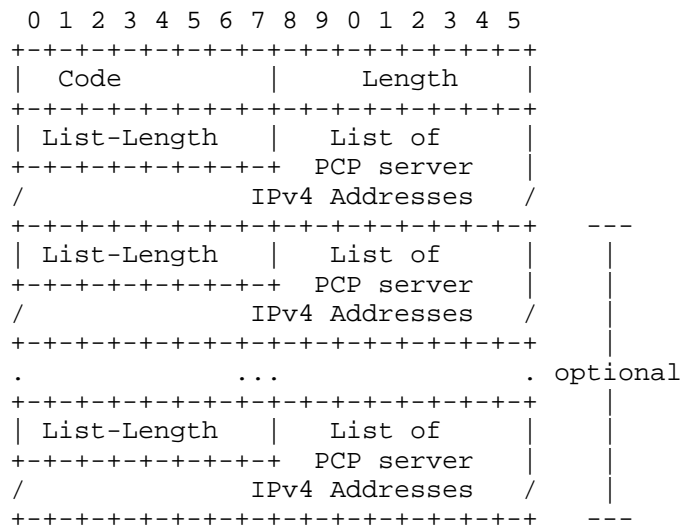


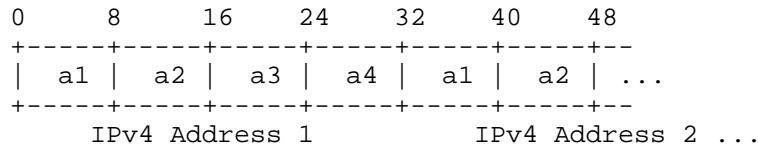
Figure 2: DHCPv4 PCP server option

The description of the fields is as follows:

- o Code: OPTION\_V4\_PCP\_SERVER (TBA, see Section 9.2);
- o Length: Length of all included data in octets. The minimum length is 5.
- o List-Length: Length of the "List of PCP server IPv4 Addresses" field in octets; MUST be a multiple of 4.



- o List of PCP server IPv4 Addresses: Contains one or more IPv4 addresses of the PCP server to be used by the PCP client. The format of this field is shown in Figure 3.
- o OPTION\_V4\_PCP\_SERVER can include multiple lists of PCP server IPv4 addresses; each list is treated as a separate PCP server. When several lists of PCP server IPv4 addresses are to be included, "List-Length" and "PCP server IPv4 Addresses" fields are repeated.



This format assumes that an IPv4 address is encoded as a1.a2.a3.a4.

Figure 3: Format of the List of PCP server IPv4 Addresses

OPTION\_V4\_PCP\_SERVER is a concatenation-requiring option. As such, the mechanism specified in [RFC3396] MUST be used if OPTION\_V4\_PCP\_SERVER exceeds the maximum DHCPv4 option size of 255 octets.

#### 4.2. DHCPv4 Client Behavior

To discover one or more PCP servers, the DHCPv4 client requests PCP server IP addresses by including OPTION\_V4\_PCP\_SERVER in a Parameter Request List Option [RFC2132].

The DHCPv4 client MUST be prepared to receive multiple lists of PCP server IPv4 addresses in the same DHCPv4 PCP server option; each list is to be treated as a separate PCP server.

The DHCPv4 client MUST silently discard multicast and host loopback addresses [RFC6890] conveyed in OPTION\_V4\_PCP\_SERVER.

#### 5. DHCP Server Configuration Guidelines

DHCP servers supporting the DHCP PCP server option can be configured with a list of IP addresses of the PCP server(s). If multiple IP addresses are configured, the DHCP server MUST be explicitly configured whether all or some of these addresses refer to:

1. the same PCP server: the DHCP server returns multiple addresses in the same instance of the DHCP PCP server option.
2. distinct PCP servers: the DHCP server returns multiple lists of PCP server IP addresses to the requesting DHCP client (encoded as

multiple `OPTION_V6_PCP_SERVER` or in the same `OPTION_V4_PCP_SERVER`); each list is referring to a distinct PCP server. For example, multiple PCP servers may be configured to a PCP client in some deployment contexts such as multi-homing. It is out of scope of this document to enumerate all deployment scenarios that require multiple PCP servers to be returned.

Precisely how DHCP servers are configured to separate lists of IP addresses according to which PCP server they address is out of scope for this document. However, DHCP servers **MUST NOT** combine the IP addresses of multiple PCP servers and return them to the DHCP client as if they belong to a single PCP server, and DHCP servers **MUST NOT** separate the addresses of a single PCP server and return them as if they belonged to distinct PCP servers. For example, if an administrator configures the DHCP server by providing a Fully Qualified Domain Name (FQDN) for a PCP server, even if that FQDN resolves to multiple addresses, the DHCP server **MUST** deliver them within a single server address block.

DHCPv6 servers that implement this option and that can populate the option by resolving FQDNs will need a mechanism for indicating whether to query for A records or only AAAA records. When a query returns A records, the IP addresses in those records are returned in the DHCPv6 response as IPv4-mapped IPv6 addresses.

Discussion: The motivation for this design is to accommodate deployment cases where an IPv4 connectivity service is provided while only DHCPv6 is in use (e.g., an IPv4-only PCP server in a DS-Lite context [RFC6333]).

Since this option requires support for IPv4-mapped IPv6 addresses, a DHCPv6 server implementation will not be complete if it does not query for A records and represent any that are returned as IPv4-mapped IPv6 addresses in DHCPv6 responses. This behavior is neither required nor suggested for DHCPv6 options in general: it is specific to `OPTION_V6_PCP_SERVER`. The mechanism whereby DHCPv6 implementations provide this functionality is beyond the scope of this document.

For guidelines on providing context-specific configuration information (e.g., returning a regional-based configuration), and information on how a DHCP server might be configured with FQDNs that get resolved on demand, see [I-D.ietf-dhc-topo-conf].

## 6. Dual-Stack Hosts

A Dual-Stack host might receive PCP server option via both DHCPv4 and DHCPv6. For guidance on how a DHCP client can handle PCP server IP lists for the same network but obtained via different mechanisms, see [I-D.ietf-pcp-server-selection].

## 7. Hosts with Multiple Interfaces

A host may have multiple network interfaces (e.g, 3G, IEEE 802.11, etc.); each configured differently. Each PCP server learned MUST be associated with the interface via which it was learned.

Refer to [I-D.ietf-pcp-server-selection] and Section 8.4 of [RFC6887] for more discussion on multi-interface considerations.

## 8. Security Considerations

The security considerations in [RFC2131] and [RFC3315] are to be considered. PCP-related security considerations are discussed in [RFC6887].

The PCP Server option targets mainly the simple threat model (Section 18.1 of [RFC6887]). It is out of scope of this document to discuss potential implications of the use of this option in the advanced threat model (Section 18.2 of [RFC6887]).

## 9. IANA Considerations

### 9.1. DHCPv6 Option

IANA is requested to assign the following new DHCPv6 Option Code in the registry maintained in <http://www.iana.org/assignments/dhcpv6-parameters>:

Option Name	Value
OPTION_V6_PCP_SERVER	TBA

### 9.2. DHCPv4 Option

IANA is requested to assign the following new DHCPv4 Option Code in the registry maintained in <http://www.iana.org/assignments/bootp-dhcp-parameters/>:

Option Name	Value	Data length	Meaning
OPTION_V4_PCP_SERVER	TBA	Variable; the minimum length is 5.	Includes one or multiple lists of PCP server IP addresses; each list is treated as a separate PCP server.

## 10. Acknowledgements

Many thanks to C. Jacquenet, R. Maglione, D. Thaler, T. Mrugalski, T. Reddy, S. Cheshire, M. Wasserman, C. Holmberg, A. Farrel, S. Farrel, B. Haberman, and P. Resnick for their review and comments.

Special thanks to T. Lemon and B. Volz for the review and their effort to enhance this specification.

## 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.
- [RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.
- [RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, March 1997.
- [RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.
- [RFC3396] Lemon, T. and S. Cheshire, "Encoding Long Options in the Dynamic Host Configuration Protocol (DHCPv4)", RFC 3396, November 2002.
- [RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.
- [RFC6890] Cotton, M., Vegoda, L., Bonica, R., and B. Haberman, "Special-Purpose IP Address Registries", BCP 153, RFC 6890, April 2013.

## 11.2. Informative References

- [I-D.boucadair-pcp-deployment-cases]  
Boucadair, M., "PCP Deployment Models", draft-boucadair-pcp-deployment-cases-01 (work in progress), December 2013.
- [I-D.ietf-dhc-topo-conf]  
Lemon, T. and T. Mrugalski, "Customizing DHCP Configuration on the Basis of Network Topology", draft-ietf-dhc-topo-conf-01 (work in progress), February 2014.
- [I-D.ietf-pcp-server-selection]  
Boucadair, M., Penno, R., Wing, D., Patil, P., and T. Reddy, "PCP Server Selection", draft-ietf-pcp-server-selection-02 (work in progress), January 2014.
- [RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.
- [RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, August 2011.

## Authors' Addresses

Mohamed Boucadair  
France Telecom  
Rennes 35000  
France

Email: mohamed.boucadair@orange.com

Reinaldo Penno  
Cisco  
USA

Email: repenno@cisco.com

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

Email: dwing@cisco.com

Internet Engineering Task Force  
Internet-Draft  
Intended status: Standards Track  
Expires: April 24, 2016

Q. Sun  
China Telecom  
M. Boucadair  
France Telecom  
S. Sivakumar  
Cisco Systems  
C. Zhou  
Huawei Technologies  
T. Tsou  
Huawei Technologies (USA)  
S. Perreault  
Jive Communications  
October 22, 2015

Port Control Protocol (PCP) Extension for Port Set Allocation  
draft-ietf-pcp-port-set-13

Abstract

In some use cases, e.g., Lightweight 4over6, the client may require not just one port, but a port set. This document defines an extension to the Port Control Protocol (PCP) allowing clients to manipulate sets of ports as a whole. This is accomplished by a new MAP option: PORT\_SET.

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 April 24, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 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 . . . . .	3
1.1. Applications Using Port Sets . . . . .	3
1.2. Lightweight 4over6 . . . . .	3
1.3. Firewall Control . . . . .	3
1.4. Discovering Stateless Port Set Mappings . . . . .	4
2. The need for PORT_SET . . . . .	4
3. Terminology . . . . .	5
4. The PORT_SET Option . . . . .	5
4.1. Client Behavior . . . . .	7
4.2. Server Behavior . . . . .	7
4.3. Absence of Capability Discovery . . . . .	8
4.4. Port Set Renewal and Deletion . . . . .	9
4.4.1. Overlap Conditions . . . . .	9
5. Examples . . . . .	9
5.1. Simple Request on NAT44 . . . . .	9
5.2. Stateless Mapping Discovery . . . . .	10
5.3. Resolving Overlap . . . . .	11
6. Operational Considerations . . . . .	12
6.1. Limits and Quotas . . . . .	12
6.2. High Availability . . . . .	12
6.3. Idempotence . . . . .	12
6.4. What Should a PCP Client Do When It Receives Fewer Ports than Requested? . . . . .	13
7. Security Considerations . . . . .	14
8. IANA Considerations . . . . .	14
9. Contributors . . . . .	14
10. Acknowledgements . . . . .	16
11. References . . . . .	16
11.1. Normative References . . . . .	16
11.2. Informative References . . . . .	16
Authors' Addresses . . . . .	17

## 1. Introduction

This document extends Port Control Protocol (PCP) [RFC6887] with the ability to retrieve a set of ports using a single request. It does so by defining a new PORT\_SET option.

This section describes a few (and non-exhaustive) envisioned use cases. Note that the PCP extension defined in this document is generic and is expected to be applicable to other use cases.

### 1.1. Applications Using Port Sets

Some applications require not just one port, but a port set. One example is a Session Initiation Protocol (SIP) User Agent Server (UAS) [RFC3261] expecting to handle multiple concurrent calls, including media termination. When it receives a call, it needs to signal media port numbers to its peer. Generating individual PCP MAP requests for each of the media ports during call setup would introduce unwanted latency and increased signaling load. Instead, the server can pre-allocate a set of ports such that no PCP exchange is needed during call setup.

### 1.2. Lightweight 4over6

In the Lightweight 4over6 (lw4o6) [RFC7596] architecture, shared global addresses can be allocated to customers. It allows moving the Network Address Translation (NAT) function, otherwise accomplished by a Carrier-Grade NAT (CGN) [RFC6888], to the Customer-Premises Equipment (CPE). This provides more control over the NAT function to the user, and more scalability to the Internet Service Provider (ISP).

In the lw4o6 architecture, the PCP-controlled device corresponds to the Lightweight AFTR (lwAFTR), and the PCP client corresponds to the Lightweight B4 (lwB4). The PCP client sends a PCP MAP request containing a PORT\_SET option to trigger shared address allocation on the Lightweight AFTR (lwAFTR). The PCP response contains the shared address information, including the port set allocated to the Lightweight B4 (lwB4).

### 1.3. Firewall Control

Port sets are often used in firewall rules. For example, defining a range for Real-time Transport Protocol (RTP) [RFC3550] traffic is common practice. The PCP MAP request can already be used for firewall control. The PORT\_SET option brings the additional ability to manipulate firewall rules operating on port sets instead of single ports.



#### 1.4. Discovering Stateless Port Set Mappings

A PCP MAP request can be used to retrieve a mapping from a stateless device (i.e., one that does not establish any per-flow state, and simply rewrites the address and/or port in a purely algorithmic fashion, including no rewriting). Similarly, a PCP MAP request with a PORT\_SET request can be used to discover a port set mapping from a stateless device. See Section 5.2 for an example.

#### 2. The need for PORT\_SET

Multiple PCP MAP requests can be used to manipulate a set of ports, having roughly the same effect as a single use of a PCP MAP request with a PORT\_SET option. However, use of the PORT\_SET option is more efficient when considering the following aspects:

**Network Traffic:** A single request uses less network resources than multiple requests.

**Latency:** Even though PCP MAP requests can be sent in parallel, we can expect the total processing time to be longer for multiple requests than a single one.

**Server-side efficiency:** Some PCP-controlled devices can allocate port sets in a manner such that data passing through the device is processed much more efficiently than the equivalent using individual port allocations. For example, a CGN having a "bulk" port allocation scheme (see [RFC6888], Section 5) often has this property.

**Server-side scalability:** The number of state table entries in PCP-controlled devices is often a limiting factor. Allocating port sets in a single request can result in a single mapping entry being used, therefore allowing greater scalability.

Therefore, while it is functionally possible to obtain the same results using plain MAP, the extension proposed in this document allows greater efficiency, scalability, and simplicity, while lowering latency and necessary network traffic.

In addition, PORT\_SET supports parity preservation. Some protocols (e.g., RTP [RFC3550]) assign meaning to a port number's parity. When mapping sets of ports for the purpose of using such kind of protocol, preserving parity can be necessary.

### 3. Terminology

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].

### 4. The PORT\_SET Option

Option Name: PORT\_SET

Number: TBD (see Section 8)

Purpose: To map sets of ports.

Valid for Opcodes: MAP

Length: 5 bytes

May appear in: Both requests and responses

Maximum occurrences: 1

The PORT\_SET option indicates that the PCP client wishes to reserve a set of ports. The requested number of ports in that set is indicated in the option.

The maximum occurrences of the PORT\_SET option MUST be limited to 1. The reason is that the suggested external port set depends on the data contained in the MAP Opcode header. Having two PORT\_SET options with a single MAP Opcode header would imply having two overlapping suggested external port sets.

Note that the option number is in the "optional to process" range (128-191), meaning that a PCP MAP request with a PORT\_SET option will be interpreted by a PCP server that does not support PORT\_SET as a single-port PCP MAP request, as if the PORT\_SET option was absent.

The PORT\_SET Option is formatted as shown in Figure 1.

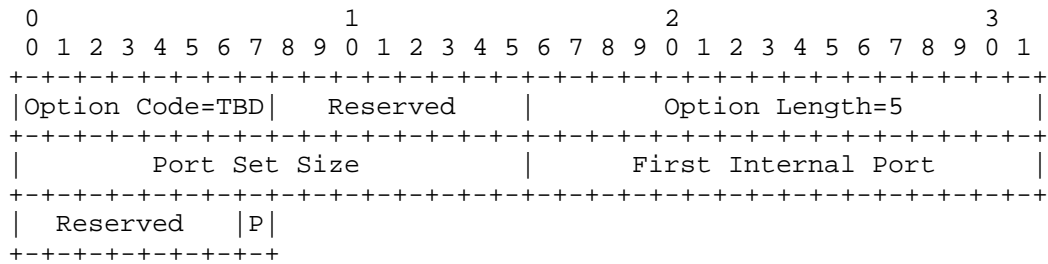


Figure 1: PORT\_SET Option

The fields are as follows:

**Port Set Size:** A 16-bit unsigned integer. Number of ports requested. MUST NOT be zero.

**First Internal Port:** In a request, this field MUST be set equal to the Internal Port field in the MAP opcode by the PCP client. In a response, this field indicates the first internal port of the port set mapped by the PCP server, which may differ from the value sent in the request. That is to be contrasted to the Internal Port field, which by necessity is always identical in matched requests and responses.

**Reserved:** MUST be set to zero when sending, MUST be ignored when receiving.

**P:** 1 if parity preservation is requested, 0 otherwise. See [RFC4787], Section 4.2.2.

The Internal Port Set is defined as being the range of Port Set Size ports starting from the First Internal Port. The Suggested External Port Set is defined as being the range of Port Set Size ports starting from the Suggested External Port. Similarly, the Assigned External Port Set is defined as being the range of Port Set Size ports starting from the Assigned External Port. The Internal Port Set returned in a response and the Assigned External Port Set have the same size.

The Suggested External Port corresponds to the first port in the suggested External Port Set. Its purpose is for clients to be able to regenerate previous mappings after state loss. When such an event happens, clients may attempt to regenerate identical mappings by suggesting the same External Port Set as before the state loss. Note that there is no guarantee that the allocated External Port Set will be the one suggested by the client.

#### 4.1. Client Behavior

To retrieve a set of ports, the PCP client adds a PORT\_SET option to its PCP MAP request. If parity preservation is required (i.e., an even port to be mapped to an even port, and an odd port to be mapped to an odd port), the PCP client MUST set the parity bit (to 1) to ask the PCP server to preserve the port parity.

The PCP client MUST NOT include more than one PORT\_SET option in a PCP MAP request. If several port sets are needed, the PCP client MUST issue separate PCP MAP requests, each potentially including a PORT\_SET option. These individual PCP MAP requests MUST include distinct Internal Ports.

If the PCP client does not know the exact number of ports it requires, it MAY then set the Port Set Size to 0xffff, indicating that it is willing to accept as many ports as the PCP server can offer.

A PCP client SHOULD NOT send a PORT\_SET option for single-port PCP MAP requests (including creation, renewal, and deletion), because that needlessly increases processing on the server.

PREFER\_FAILURE MUST NOT appear in a request with PORT\_SET option. As a reminder PREFER\_FAILURE was specifically designed for the Universal Plug and Play (UPnP) Internet Gateway Device - Port Control Protocol Interworking Function (IGD-PCP IWF) [RFC6970]. The reasons for not recommending the use of PREFER\_FAILURE are discussed in Section 13.2 of [RFC6887].

When the PCP-controlled device supports multiple port-sets delegation for a given PCP client, the PCP client MAY re-initiate a PCP request to get another port set when it has exhausted all the ports within the port-set.

#### 4.2. Server Behavior

In addition to regular PCP MAP request processing, the following checks are made upon receipt of a PORT\_SET option with non-zero Requested Lifetime:

- o If multiple PORT\_SET options are present in a single PCP MAP request, a MALFORMED\_OPTION error is returned.
- o If the Port Set Size is zero, a MALFORMED\_OPTION error is returned.

- o If PREFER\_FAILURE option is present, a MALFORMED\_OPTION error is returned.

The PCP server MAY map fewer ports than the value of Port Set Size from the request. It MUST NOT map more ports than the PCP client asked for. Internal ports outside the range of Port Set Size ports starting from the Internal Port MUST NOT be mapped by the PCP server.

If the requested port set cannot be fully satisfied, the PCP server SHOULD map as many ports as possible, and SHOULD map at least one port (which is the same behavior as if Port Set Size is set to 1).

If the PCP server ends up mapping only a single port, for any reason, the PORT\_SET option MUST NOT be present in the response. In particular, if the PCP server receives a single-port PCP MAP request that includes a PORT\_SET option, the PORT\_SET option is silently ignored and the request is handled as a single-port PCP MAP request.

If the port parity preservation is requested ( $P = 1$ ), the PCP server MAY preserve port parity. In that case, the External Port is set to a value having the same parity as the First Internal Port.

If the mapping is successful, the MAP response's Assigned External Port is set to the first port in the External Port Set, and the PORT\_SET option's Port Set Size is set to number of ports in the mapped port set. The First Internal Port field is set to the first port in the Internal Port Set.

#### 4.3. Absence of Capability Discovery

A PCP client that wishes to make use of a port set includes the PORT\_SET option. If no PORT\_SET option is present in the response, the PCP client cannot conclude that the PCP server does not support the PORT\_SET option. It may just be that the PCP server does support PORT\_SET but decided to allocate only a single port, for reasons that are its own. If the client wishes to obtain more ports, it MAY send additional PCP MAP requests (see Section 6.4), which the PCP server may or may not grant according to local policy.

If port set capability is added to or removed from a running PCP server, the server MAY reset its Epoch time and send an ANNOUNCE message as described in the PCP specification ([RFC6887], Section 14.1). This causes PCP clients to retry, and those using PORT\_SET will now receive a different response.

#### 4.4. Port Set Renewal and Deletion

Port set mappings are renewed and deleted as a single entity. That is, the lifetime of all port mappings in the set is set to the Assigned Lifetime at once.

A PCP client attempting to refresh or delete a port set mapping MUST include the PORT\_SET option in its request.

##### 4.4.1. Overlap Conditions

Port set PCP MAP requests can overlap with existing single port or port set mappings. This can happen either by mistake or after a PCP client becomes out of sync with server state.

If a PCP server receives a PCP MAP request, with or without a PORT\_SET option, that tries to map one or more internal ports or port sets belonging to already existing mappings, then the request is considered to be a refresh request applying those mappings. Each of the matching port or port set mappings is processed independently, as if a separate refresh request had been received. The processing is as described in Section 15 of [RFC6887]. The PCP server sends a Mapping Update message for each of the mappings.

#### 5. Examples

##### 5.1. Simple Request on NAT44

An application requires a range of 100 IPv4 UDP ports to be mapped to itself. The application running on the host has created sockets bound to IPv4 UDP ports 50,000 to 50,099 for this purpose. It does not care about which external port numbers are allocated. The PCP client sends a PCP request with the following parameters over IPv4:

- o MAP opcode

Mapping Nonce: <a random nonce>

Protocol: 17

Internal Port: 50,000

Suggested External Port: 0

Suggested External IP Address: ::ffff:0.0.0.0

- o PORT\_SET Option

Port Set Size: 100

First Internal Port: 50,000

P: 0

The PCP server is unable to fulfill the request fully: it is configured by local policy to only allocate 32 ports per user. Since the `PREFER_FAILURE` option is absent from the request, it decides to map UDP ports 37,056 to 37,087 on external address 192.0.2.3 to internal ports 50,000 to 50,031. After setting up the mapping in the NAT44 device it controls, it replies with the following PCP response:

- o MAP opcode

Mapping Nonce: <copied from the request>

Protocol: 17

Internal Port: 50,000

Assigned External Port: 37,056

Assigned External IP Address: ::ffff:192.0.2.3

- o PORT\_SET Option

Port Set Size: 32

First Internal Port: 50,000

P: 0

Upon receiving this response, the host decides that 32 ports is good enough for its purposes. It closes sockets bound to ports 50,032 to 50,099, sets up a refresh timer, and starts using the port range it has just been assigned.

## 5.2. Stateless Mapping Discovery

A host wants to discover a stateless NAT44 mapping pointing to it. To do so, it sends the following request over IPv4:

- o MAP opcode

Mapping Nonce: <a random nonce>

Protocol: 0

Internal Port: 1

Suggested External Port: 0

Suggested External IP Address: ::ffff:0.0.0.0

- o PORT\_SET Option

Port Set Size: 65,535

First Internal Port: 1

P: 0

The PCP server sends the following response:

- o MAP opcode

Mapping Nonce: <copied from the request>

Protocol: 0

Internal Port: 1

Assigned External Port: 26,624

Assigned External IP Address: ::ffff:192.0.2.5

- o PORT\_SET Option

Port Set Size: 2048

First Internal Port: 26,624

P: 0

From this response, the host understands that a 2048-port stateless mapping is pointing to itself, starting from port 26,624 on external IP address 192.0.2.5.

### 5.3. Resolving Overlap

This example relates to Section 4.4.1.

Suppose internal port 100 is mapped to external port 100 and port set 101-199 is mapped to external port set 201-299. The PCP server receives a PCP MAP request with Internal Port = 100, External Port = 0, and a PORT\_SET option with Port Set Size = 100. The request's



Mapping Nonce is equal to those of the existing single port and port set mappings. This request is therefore treated as two refresh requests, the first one applying to the single port mapping and the second one applying to the port set mapping. The PCP server updates both mapping's lifetimes as usual then sends two responses: the first one contains Internal Port = 100, External Port = 100, and no PORT\_SET option, while the second one contains Internal Port = 101, External Port = 201, and a PORT\_SET option with Port Set Size = 99.

## 6. Operational Considerations

### 6.1. Limits and Quotas

It is up to the PCP server to determine the port-set quota, if any, for each PCP client.

If the PCP server is configured to allocate multiple port-set allocations for one subscriber, the same Assigned External IP Address SHOULD be assigned to the subscriber in multiple port-set responses.

To optimize the number of mapping entries maintained by the PCP server, it is RECOMMENDED to configure the PCP server to assign the maximum allowed port set size in a single response. This policy SHOULD be configurable.

### 6.2. High Availability

The failover mechanism in MAP (Section 14 in [RFC6887]) can also be applied to port sets.

### 6.3. Idempotence

A core, desirable property of the PCP protocol is idempotence. In a nutshell, requests produce the same results whether they are executed once or multiple times. This property is preserved with the PORT\_SET attribute, with the following caveat: the order in which the PCP server receives requests with overlapping Internal Port Sets will affect the mappings being created and the responses received.

For example suppose these two requests are sent by a PCP client:

Request A: Internal Port Set 1-10

Request B: Internal Port Set 5-14

The PCP server's actions will depend on which request is received first. Suppose that A is received before B:

Upon reception of A: Internal ports 1-10 are mapped. A success response containing the following fields is sent:

Internal Port: 1

First Internal Port: 1

Port Set Size: 10

Upon reception of B: The request matches mapping A. The request is interpreted as a refresh request for mapping A, and a response containing the following fields is sent:

Internal Port: 5

First Internal Port: 1

Port Set Size: 10

If the order of reception is reversed (B before A), the created mapping will be different, and the First Internal Port in both responses would then be 5.

To avoid surprises, PCP clients MUST ensure that port set mapping requests do not inadvertently overlap. For example, a host's operating system could include a central PCP client process through which port set mapping requests would be arbitrated. Alternatively, individual PCP clients running on the same host would be required to acquire the internal ports from the operating system (e.g., a call to the `bind()` function from the BSD API) before trying to map them with PCP.

#### 6.4. What Should a PCP Client Do When It Receives Fewer Ports than Requested?

Suppose a PCP client asks for 16 ports and receives 8. What should it do? Should it consider this a final answer? Should it try a second request, asking for 8 more ports? Should it fall back to 8 individual PCP MAP requests? This document leaves the answers to be implementation-specific, but describes issues to be considered when answering them.

First, the PCP server has decided to allocate 8 ports for some reason. It may be that allocation sizes have been limited by the PCP server's administrator. It may be that the PCP client has reached a quota. It may be that these 8 ports were the last contiguous ones available. Depending on the reason, asking for more ports may or may

not be likely to actually yield more ports. However, the PCP client has no way of knowing.

Second, not all PCP clients asking for N ports actually need all N ports to function correctly. For example, a DNS resolver could ask for N ports to be used for source port randomization. If fewer than N ports are received, the DNS resolver will still work correctly, but source port randomization will be slightly less efficient, having fewer bits to play with. In that case, it would not make much sense to ask for more ports.

Finally, asking for more ports could be considered abuse. External ports are a resource that is to be shared among multiple PCP clients. A PCP client trying to obtain more than its fair share could trigger countermeasures according to local policy.

In conclusion, it is expected that for most applications, asking for more ports would not yield benefits justifying the additional costs.

## 7. Security Considerations

The security considerations discussed in [RFC6887] apply to this extension.

As described in Section 4.4.1, a single PCP request using the PORT\_SET option may result in multiple responses. For this to happen it is necessary that the request contain the nonce associated to multiple mappings on the server. Therefore, an on-path attacker could use an eavesdropped nonce to mount an amplification attack. Use of PCP authentication ([RFC6887], Section 18) eliminates this attack vector.

In order to prevent a PCP client from controlling all ports bound to a shared IP address, port quotas should be configured on the PCP server (Section 17.2 of [RFC6887]).

## 8. IANA Considerations

IANA has allocated value TBD (note to IANA: to be allocated from the range 128-191) in the "PCP Options" registry at <http://www.iana.org/assignments/pcp-parameters> for the new PCP option defined in Section 4.

## 9. Contributors

The following are extended authors who contributed to the effort:

Yunqing Chen

China Telecom  
Room 502, No.118, Xizhimennei Street  
Beijing 100035  
P.R.China  
Chongfeng Xie  
China Telecom  
Room 502, No.118, Xizhimennei Street  
Beijing 100035  
P.R.China  
Yong Cui  
Tsinghua University  
Beijing 100084  
P.R.China  
Phone: +86-10-62603059  
Email: yong@csnet1.cs.tsinghua.edu.cn  
Qi Sun  
Tsinghua University  
Beijing 100084  
P.R.China  
Phone: +86-10-62785822  
Email: sunqibupt@gmail.com  
Gabor Bajko  
Mediatek Inc.  
Email: gabor.bajko@mediatek.com

Xiaohong Deng

France Telecom

Email: xiaohong.deng@orange-ftgroup.com

## 10. Acknowledgements

The authors would like to show sincere appreciation to Alain Durand, Cong Liu, Dan Wing, Dave Thaler, Peter Koch, Reinaldo Penno, Sam Hartman, Stuart Cheshire, Ted Lemon, Yoshihiro Ohba, Meral Shirazipour, Jouni Korhonen, and Ben Campbell for their useful comments and suggestions.

## 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.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

### 11.2. Informative References

- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.
- [RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.
- [RFC6888] Perreault, S., Yamagata, I., Miyakawa, S., Nakagawa, A., and H. Ashida, "Common Requirements for Carrier-Grade NATs (CGNs)", BCP 127, RFC 6888, April 2013.

- [RFC6970] Boucadair, M., Penno, R., and D. Wing, "Universal Plug and Play (UPnP) Internet Gateway Device - Port Control Protocol Interworking Function (IGD-PCP IWF)", RFC 6970, DOI 10.17487/RFC6970, July 2013, <<http://www.rfc-editor.org/info/rfc6970>>.
- [RFC7596] Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the Dual-Stack Lite Architecture", RFC 7596, DOI 10.17487/RFC7596, July 2015, <<http://www.rfc-editor.org/info/rfc7596>>.

## Authors' Addresses

Qiong Sun  
China Telecom  
P.R.China

Phone: 86 10 58552936  
Email: [sunqiong@ctbri.com.cn](mailto:sunqiong@ctbri.com.cn)

Mohamed Boucadair  
France Telecom  
Rennes 35000  
France

Email: [mohamed.boucadair@orange.com](mailto:mohamed.boucadair@orange.com)

Senthil Sivakumar  
Cisco Systems  
7100-8 Kit Creek Road  
Research Triangle Park, North Carolina 27709  
USA

Phone: +1 919 392 5158  
Email: [ssenthil@cisco.com](mailto:ssenthil@cisco.com)

Cathy Zhou  
Huawei Technologies  
Bantian, Longgang District  
Shenzhen 518129  
P.R. China

Email: [cathy.zhou@huawei.com](mailto:cathy.zhou@huawei.com)

Tina Tsou  
Huawei Technologies (USA)  
2330 Central Expressway  
Santa Clara, CA 95050  
USA

Phone: +1 408 330 4424  
Email: Tina.Tsou.Zouting@huawei.com

Simon Perreault  
Jive Communications  
Quebec, QC  
Canada

Email: sperreault@jive.com

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: January 15, 2016

S. Perreault  
Jive Communications  
M. Boucadair  
France Telecom  
R. Penno  
D. Wing  
Cisco  
S. Cheshire  
Apple  
July 14, 2015

Port Control Protocol (PCP) Proxy Function  
draft-ietf-pcp-proxy-09

Abstract

This document specifies a new PCP functional element denoted as a PCP Proxy. The PCP Proxy relays PCP requests received from PCP clients to upstream PCP server(s). A typical deployment usage of this function is to help establish successful PCP communications for PCP clients that can not be configured with the address of a PCP server located more than one hop away.

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 January 15, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 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
1.1. Use Case: the NAT Cascade . . . . .	3
1.2. Use Case: the PCP Relay . . . . .	4
2. Terminology . . . . .	5
3. Operation of the PCP Proxy . . . . .	5
3.1. Optimized Hairpin Routing . . . . .	8
3.2. Termination of Recursion . . . . .	8
3.3. Source Address for PCP Requests Sent Upstream . . . . .	9
3.4. Unknown OpCodes and Options . . . . .	9
3.4.1. No NAT is Co-located with the PCP Proxy . . . . .	9
3.4.2. PCP Proxy Co-located with a NAT Function . . . . .	10
3.5. Mapping Repair . . . . .	10
3.6. Multiple PCP Servers . . . . .	11
4. IANA Considerations . . . . .	11
5. Security Considerations . . . . .	11
6. Acknowledgements . . . . .	12
7. References . . . . .	12
7.1. Normative References . . . . .	12
7.2. Informative References . . . . .	12
Authors' Addresses . . . . .	13

## 1. Introduction

This document defines a new PCP [RFC6887] functional element: the PCP Proxy. As shown in Figure 1, the PCP proxy is logically equivalent to a PCP client back-to-back with a PCP server. The "glue" between the two is what is specified in this document. Other than that "glue", the server and the client behave exactly like their regular counterparts.

The PCP Proxy is responsible for relaying PCP messages received from PCP clients to upstream PCP servers and vice versa.

Whether the PCP Proxy is co-located with a flow-aware function (e.g., NAT, firewall) is deployment-specific.

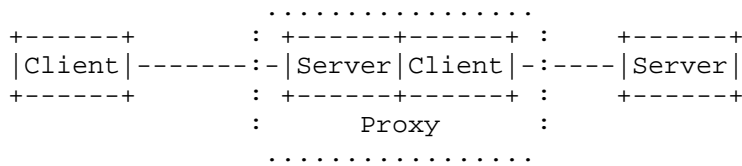


Figure 1: Reference Architecture

This document assumes a hop-by-hop PCP authentication scheme. That is, in reference to Figure 1, the left-most PCP client authenticates with the PCP Proxy, while the PCP Proxy authenticates with the upstream server. Note that in some deployments, PCP authentication may only be enabled between the PCP Proxy and an upstream PCP server (e.g., a customer premises host may not authenticate with the PCP Proxy but the PCP Proxy may authenticate with the PCP server). The hop-by-hop authentication scheme is more suitable from a deployment standpoint. Furthermore, it allows to easily support a PCP Proxy that alters PCP messages (e.g., strip a PCP option, modify a PCP field, etc.).

#### 1.1. Use Case: the NAT Cascade

In today's world, with public routable IPv4 addresses becoming less readily available, it is increasingly common for customers to receive a private address from their Internet Service Provider (ISP), and the ISP uses a NAT gateway of its own to translate those packets before sending them out onto the public Internet. This means that there is likely to be more than one NAT on the path between client machines and the public Internet:

- o If a residential customer receives a translated address from their ISP, and then installs their own residential NAT gateway to share that address between multiple client devices in their home, then there are at least two NAT gateways on the path between client devices and the public Internet.
- o If a mobile phone customer receives a translated address from their mobile phone carrier, and uses "Personal Hotspot" or "Internet Sharing" software on their mobile phone to make Wireless LAN (WLAN) Internet access available to other client devices, then there are at least two NAT gateways on the path between those client devices and the public Internet.
- o If a hotel guest connects a portable WLAN gateway to their hotel room Ethernet port to share their room's Internet connection between their phone and their laptop computer, then packets from the client devices may traverse the hotel guest's portable NAT,

the hotel network's NAT, and the ISP's NAT before reaching the public Internet.

While it is possible, in theory, that client devices could somehow discover all the NATs on the path, and communicate with each one separately using Port Control Protocol [RFC6887], in practice it's not clear how client devices would reliably learn this information. Since the NAT gateways are installed and operated by different individuals and organizations, no single entity has knowledge of all the NATs on the path. Also, even if a client device could somehow know all the NATs on the path, requiring a client device to communicate separately with all of them imposes unreasonable complexity on PCP clients, many of which are expected to be simple low-cost devices.

In addition, this goes against the spirit of NAT gateways. The main purpose of a NAT gateway is to make multiple downstream client devices to appear, from the point of view of everything upstream of the NAT gateway, to be a single client device. In the same spirit, it makes sense for a PCP-capable NAT gateway to make multiple downstream client devices requesting port mappings to appear, from the point of view of everything upstream of the NAT gateway, to be a single client device requesting port mappings.

## 1.2. Use Case: the PCP Relay

Another envisioned use case of the PCP Proxy is to help establish successful PCP communications for PCP clients that can not be configured with the address of a PCP server located more than one hop away. A PCP Proxy can be for instance embedded in a CPE (Customer Premises Equipment) while the PCP server is located in a network operated by an ISP. This is illustrated in Figure 2.

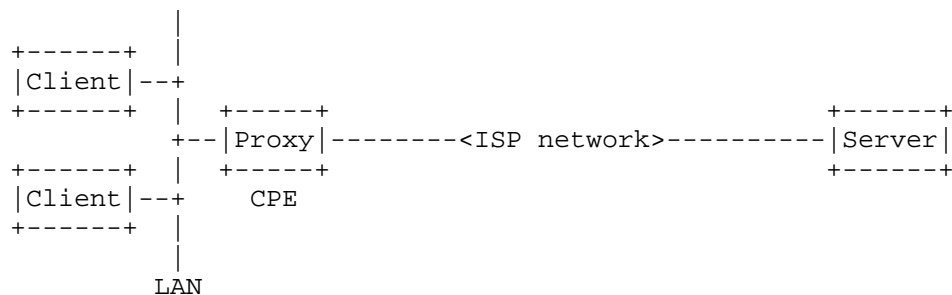


Figure 2: PCP Relay Use Case

This works because the proxy's server side is listening on the address used as a default gateway by the clients. The clients use

that address as a fallback when discovering the PCP server's address. The proxy picks up the requests and forwards them upstream to the ISP's PCP server, with whose address it has been provisioned through regular PCP client provisioning means.

This particular use case assumes that provisioning the server's address on the CPE is feasible while doing it on the clients in the LAN is not, which is what makes the PCP proxy valuable.

Note that [I-D.ietf-pcp-anycast] documents an alternate solution to the PCP proxy. Nevertheless, as discussed in [I-D.boucadair-pcp-deployment-cases], the anycast solution may be problematic when multiple PCP servers are to be contacted.

## 2. Terminology

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

Where this document uses the terms "upstream" and "downstream", the term "upstream" refers to the direction outbound packets travel towards the public Internet, and the term "downstream" refers to the direction inbound packets travel from the public Internet towards client systems. Typically when a home user views a web site, their computer sends an outbound TCP SYN packet upstream towards the public Internet, and an inbound downstream TCP SYN ACK reply comes back from the public Internet.

## 3. Operation of the PCP Proxy

Upon receipt of a PCP mapping-creation request from a downstream PCP client, a PCP proxy first examines its local mapping table to see if it already has a valid active mapping matching the Internal Address and Internal Port (and in the case of PEER requests, remote peer) given in the request.

If the PCP proxy does not already have a valid active mapping for this mapping-creation request, then it allocates an available port on its external interface. We assume for the sake of this description that the address of its external interface is itself a private address, subject to translation by an upstream NAT. The PCP proxy then constructs an appropriate corresponding PCP request of its own (described below), and sends it to its upstream NAT, and the newly-created local mapping is considered temporary until a confirming reply is received from the upstream PCP server.

If the PCP proxy does already have a valid active mapping for this mapping-creation request, and the lifetime remaining on the local mapping is at least 3/4 of the lifetime requested by the PCP client, then the PCP proxy SHOULD send an immediate reply giving the outermost External Address and Port (previously learned using PCP recursively, as described below), and the actual lifetime remaining for this mapping. If the lifetime remaining on the local mapping is less than 3/4 of the lifetime requested by the PCP client, then the PCP proxy MUST generate an upstream request as described below.

For mapping-deletion requests (Lifetime = 0), the local mapping, if any, is deleted, and then (regardless of whether a local mapping existed) a corresponding upstream request is generated.

The PCP proxy knows the destination IP address for its upstream PCP request using the same means that are available for provisioning a PCP client. In particular, the PCP proxy MUST follow the procedure defined in Section 8.1 of [RFC6887] to discover its PCP server. This does not preclude other means from being used in addition.

In the upstream PCP request:

- o The PCP Client's IP Address and Internal Port are the PCP proxy's own external address and port just allocated for this mapping.
- o The Suggested External Address and Port in the upstream PCP request SHOULD be copied from the original PCP request.
- o The Requested Lifetime is as requested by the client if it falls within the acceptable range for this PCP server; otherwise it SHOULD be capped to appropriate minimum and maximum values configured for this PCP server.
- o The Mapping Nonce is copied from the original PCP request.
- o For PEER requests, the Remote Peer IP Address and Port are copied from the original PCP request.

Upon receipt of a PCP reply giving the outermost (i.e., publicly routable) External Address, Port and Lifetime, the PCP proxy records this information in its own mapping table and relays the information to the requesting downstream PCP client in a PCP reply. The PCP proxy therefore records, among other things, the following information in its mapping table:

- o Client's Internal Address and Port.
- o External Address and Port allocated by this PCP proxy.

- o Outermost External Address and Port allocated by the upstream PCP server.
- o Mapping lifetime (also dictated by the upstream PCP server).
- o Mapping nonce.

In the downstream PCP reply:

- o The Lifetime is as granted by the upstream PCP server, or less, if the granted lifetime exceeds the maximum lifetime this PCP server is configured to grant. If the downstream Lifetime is more than the Lifetime granted by the upstream PCP server (which is NOT RECOMMENDED) then this PCP proxy MUST take responsibility for renewing the upstream mapping itself.
- o The Epoch Time is this PCP proxy's Epoch Time, not the Epoch Time of the upstream PCP server. Each PCP server has its own independent Epoch Time. However, if the Epoch Time received from the upstream PCP server indicates a loss of state in that PCP server, the PCP proxy can either recreate the lost mappings itself, or it can reset its own Epoch Time to cause its downstream clients to perform such state repairs themselves. A PCP proxy MUST NOT simply copy the upstream PCP server's Epoch Time into its downstream PCP replies, since if it suffers its own state loss it needs the ability to communicate that state loss to clients. Thus each PCP server has its own independent Epoch Time. However, as a convenience, a downstream PCP proxy may simply choose to reset its own Epoch Time whenever it detects that its upstream PCP server has lost state. Thus, in this case, the PCP proxy's Epoch Time always resets whenever its upstream PCP server loses state; it may also reset at other times too.
- o The Mapping Nonce is copied from the reply received from the upstream PCP server.
- o The Assigned External Port and Assigned External IP Address are copied from the reply received from the upstream PCP server (i.e., they are the outermost External IP Address and Port, not the locally-assigned external address and port.)
- o For PEER requests, the Remote Peer IP Address and Port are copied from the reply received from the upstream PCP server.

### 3.1. Optimized Hairpin Routing

A PCP proxy SHOULD implement Optimized Hairpin Routing. What this means is the following:

- o If a PCP proxy observes an outgoing packet arriving on its internal interface that is addressed to an External Address and Port appearing in the NAT gateway's own mapping table, then the NAT gateway SHOULD (after creating a new outbound mapping if one does not already exist) rewrite the packet appropriately and deliver it to the internal client currently allocated that External Address and Port.
- o If a PCP proxy observes an outgoing packet arriving on its internal interface which is addressed to an Outermost External Address and Port appearing in the NAT gateway's own mapping table, then the NAT gateway SHOULD do likewise: create a new outbound mapping if one does not already exist, and then rewrite the packet appropriately and deliver it to the internal client currently allocated that Outermost External Address and Port. This is not necessary for successful communication, but for efficiency. Without this Optimized Hairpin Routing, the packet will be delivered all the way to the outermost NAT gateway, which will then perform standard hairpin translation and send it back. Using knowledge of the Outermost External Address and Port, this rewriting can be anticipated and performed locally, which will typically offer higher throughput and lower latency than sending it all the way to the outermost NAT gateway and back.

Note that traffic counters maintained by an upstream PCP server will differ from the ones of a PCP Proxy implementing the optimized hairpin routing.

### 3.2. Termination of Recursion

Any recursive algorithm needs a mechanism to terminate the recursion at the appropriate point. This termination of recursion can be achieved in a variety of ways. The following (non exhaustive) examples are provided for illustration purposes:

- o An ISP's PCP-controlled gateway (that may embed a NAT, firewall or any function that can be controlled with PCP) could be configured to know that it is the outermost PCP-controlled gateway, and consequently does not need to relay PCP requests upstream.
- o A PCP-controlled gateway could determine automatically that if its external address is not one of the known private addresses [RFC1918][RFC6598], then its external address is a public routable

IP address, and consequently it does not need to relay PCP requests upstream.

- o Recursion may be terminated if there is no explicit list of PCP servers configured to the PCP Proxy (e.g., [RFC7291]) or if its default router is not responsive to PCP requests.
- o Recursion may also be terminated if the upstream PCP-controlled device does not embed a PCP Proxy.

### 3.3. Source Address for PCP Requests Sent Upstream

As with a regular PCP server, the PCP-controlled device can be a NAT, a firewall, or even some sort of hybrid. In particular, a PCP proxy that simply relays all requests upstream can be thought of as the degenerate case of a PCP server controlling a wide-open firewall back-to-back with a regular PCP client.

One important property of the PCP-controlled device will affect the PCP proxy's behaviour: when the proxy's server part instructs the device to create a mapping, that mapping's external address may or may not be one that belongs to the proxy node.

- o When the mapping's external address belongs to the proxy node, as would presumably be the case for a NAT, then the proxy's client side sends out an upstream PCP request using the mapping's external IP address as source.
- o When the mapping's external address does not belong to the proxy node, as would presumably be the case for a firewall, then the proxy's client side needs to install upstream mappings on behalf of its downstream clients. To do this, it MUST insert a THIRD\_PARTY Option in its upstream PCP request carrying the mapping's external address.

Note that hybrid PCP-controlled devices may create NAT-like mappings in some circumstances and firewall-like mappings in others. A proxy controlling such a device would adjust its behavior dynamically depending on the kind of mapping created.

### 3.4. Unknown OpCodes and Options

#### 3.4.1. No NAT is Co-located with the PCP Proxy

When no NAT is co-located with the PCP Proxy, the port numbers included in received PCP messages (from the PCP server or PCP client(s)) are not altered by the PCP Proxy. The PCP Proxy relays to



the PCP server unknown Options and OpCodes because there is no reachability failure risk.

#### 3.4.2. PCP Proxy Co-located with a NAT Function

By default, the proxy MUST relay unknown OpCodes and mandatory-to-process unknown Options. Rejecting unknown Options and OpCodes has the drawback of preventing a PCP client to make use of new capabilities offered by the PCP server but not supported by the PCP Proxy even if no IP address and/or port is included in the Option/OpCode.

Because PCP messages with an unknown OpCode or mandatory-to-process unknown Options can carry a hidden internal address or internal port that will not be translated, a PCP Proxy MUST be configurable to disable relaying unknown OpCodes and mandatory-to-process unknown Options. If the PCP Proxy is configured to disable relaying unknown OpCodes and mandatory-to-process unknown Options, the PCP Proxy MUST behave as follows:

- o a PCP Proxy co-located with a NAT MUST reject by an UNSUPP\_OPCODE error response a received request with an unknown OpCode.
- o a PCP Proxy co-located with a NAT MUST reject by an UNSUPP\_OPTION error response a received request with a mandatory-to-process unknown Option.

#### 3.5. Mapping Repair

ANNOUNCE requests received from PCP clients are handled locally; as such these requests MUST NOT be relayed to the provisioned PCP server.

Upon receipt of an unsolicited ANNOUNCE response from a PCP server, the PCP Proxy proceeds to renew the mappings and checks whether there are changes compared to a local cache if it is maintained by the PCP Proxy. If no change is detected, no unsolicited ANNOUNCE is generated towards PCP clients. If a change is detected, the PCP Proxy MUST generate unsolicited ANNOUNCE message(s) to appropriate PCP clients. If the PCP Proxy does not maintain a local cache for the mappings, unsolicited multicast ANNOUNCE messages are sent to PCP clients.

Upon change of its external IP address, the PCP Proxy SHOULD renew the mappings it maintained. If the PCP server assigns a different external port, the PCP Proxy SHOULD follow the mapping repair procedure defined in [RFC6887]. This can be achieved only if a full state table is maintained by the PCP Proxy.

### 3.6. Multiple PCP Servers

A PCP Proxy MAY handle multiple PCP servers at the same time. Each PCP server is associated with its own epoch value. PCP clients are not aware of the presence of multiple PCP servers.

According to [RFC7488], if several PCP Names are configured to the PCP Proxy, it will contact in parallel all these PCP servers.

In some contexts (e.g., PCP-controlled CGNs), the PCP Proxy MAY load balance the PCP clients among available PCP servers. The PCP Proxy MUST ensure requests of a given PCP client are relayed to the same PCP server.

The PCP Proxy MAY rely on some fields (e.g., Zone ID [I-D.penno-pcp-zones]) in the PCP request to redirect the request to a given PCP server.

### 4. IANA Considerations

This document makes no request of IANA.

### 5. Security Considerations

The PCP Proxy MUST follow the security considerations elaborated in [RFC6887] for both the client and server side.

Section 3.3 specifies the cases where a THIRD\_PARTY option is inserted by the PCP Proxy. In those cases, means to prevent a malicious user from creating mappings on behalf of a third party must be enabled as discussed in Section 13.1 of [RFC6887]. In particular, THIRD\_PARTY options MUST NOT be enabled unless the network on which the PCP messages are to be sent is fully trusted. For example if access control lists (ACLs) are installed on the PCP Proxy, PCP server, and the network between them, so those ACLs allow only communications from a trusted PCP Proxy to the PCP server.

A received request carrying an unknown OpCode or Option SHOULD be dropped (or in the case of an unknown Option which is not mandatory-to-process the Option SHOULD be removed) if it is not compatible with security controls provisioned to the PCP Proxy.

The device embedding the PCP Proxy MAY block PCP requests directly sent to the PCP server. This can be enforced using access control lists.

## 6. Acknowledgements

Many thanks to C. Zhou, T. Reddy, and D. Thaler for their review and comments.

Special thanks to F. Dupont who contributed to this document.

## 7. References

### 7.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

### 7.2. Informative References

- [I-D.boucadair-pcp-deployment-cases] Boucadair, M., "Port Control Protocol (PCP) Deployment Models", draft-boucadair-pcp-deployment-cases-03 (work in progress), July 2014.
- [I-D.ietf-pcp-anycast] Kiesel, S., Penno, R., and S. Cheshire, "Port Control Protocol (PCP) Anycast Addresses", draft-ietf-pcp-anycast-06 (work in progress), May 2015.
- [I-D.penno-pcp-zones] Penno, R., "PCP Support for Multi-Zone Environments", draft-penno-pcp-zones-01 (work in progress), October 2011.
- [RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.
- [RFC6598] Weil, J., Kuarsingh, V., Donley, C., Liljenstolpe, C., and M. Azinger, "IANA-Reserved IPv4 Prefix for Shared Address Space", BCP 153, RFC 6598, April 2012.
- [RFC7291] Boucadair, M., Penno, R., and D. Wing, "DHCP Options for the Port Control Protocol (PCP)", RFC 7291, July 2014.
- [RFC7488] Boucadair, M., Penno, R., Wing, D., Patil, P., and T. Reddy, "Port Control Protocol (PCP) Server Selection", RFC 7488, March 2015.

Authors' Addresses

Simon Perreault  
Jive Communications  
Quebec, QC  
Canada

Email: sperreault@jive.com

Mohamed Boucadair  
France Telecom  
Rennes 35000  
France

Email: mohamed.boucadair@orange.com

Reinaldo Penno  
Cisco  
USA

Email: repenno@cisco.com

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

Email: dwing@cisco.com

Stuart Cheshire  
Apple Inc.  
1 Infinite Loop  
Cupertino, California 95014  
USA

Phone: +1 408 974 3207  
Email: cheshire@apple.com

PCP  
Internet-Draft  
Intended status: Standards Track  
Expires: February 20, 2014

S. Kiesel  
University of Stuttgart  
R. Penno  
Cisco Systems  
August 19, 2013

PCP Server Discovery based on well-known IP Address  
draft-kiesel-pcp-ip-based-srv-disc-01

Abstract

The Port Control Protocol (PCP) provides a mechanism to control how incoming packets are forwarded by upstream devices such as Network Address Translator IPv6/IPv4 (NAT64), Network Address Translator IPv4/IPv4 (NAT44), IPv6 and IPv4 firewall devices, and a mechanism to reduce application keep alive traffic.

This document establishes a well-known IP address for the PCP Server and documents how PCP clients embedded in endpoints can use it during the discovery and regular operation phases.

## 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

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 February 20, 2014.

## 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 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 . . . . .	4
2. PCP Server Discovery based on well-known IP Address . . . . .	5
2.1. Well-Known PCP Server IP Address (WkPsdIPa) . . . . .	5
2.2. PCP Discovery Client behavior . . . . .	5
2.3. PCP Discovery Server behavior . . . . .	5
3. Deployment Considerations . . . . .	6
3.1. Multiple PCP Servers, Symmetric Routing . . . . .	6
3.2. Multiple PCP Servers, Assymetric Routing . . . . .	6
4. IANA Considerations . . . . .	8
4.1. Registration of IPv4 Special Purpose Address . . . . .	8
4.2. Registration of IPv6 Special Purpose Address . . . . .	9
4.3. PCP Option . . . . .	10
5. Security Considerations . . . . .	11
6. Acknowledgements . . . . .	12
7. References . . . . .	13
7.1. Normative References . . . . .	13
7.2. Informative References . . . . .	13
Appendix A. Problems with Other Discovery methods . . . . .	15
A.1. DHCP PCP Options . . . . .	15
A.2. Default Router . . . . .	15
A.3. User Input . . . . .	15
A.4. Domain Name System Based . . . . .	16
Authors' Addresses . . . . .	17

## 1. Introduction

The Port Control Protocol (PCP) [I-D.ietf-pcp-base] provides a mechanism to control how incoming packets are forwarded by upstream devices such as Network Address Translator IPv6/IPv4 (NAT64), Network Address Translator IPv4/IPv4 (NAT44), IPv6 and IPv4 firewall devices, and a mechanism to reduce application keep alive traffic.

But before a PCP client can perform any of these tasks it needs to discover one or more PCP servers. Several algorithms have been specified that produce a suitable PCP Server address given PCP client (i.e., the address may vary for different clients or different points of network attachment, etc.). These approaches are based on user input, DHCP [I-D.ietf-pcp-dhcp] or default router, which is the one detailed in the PCP base document [I-D.ietf-pcp-base].

But unfortunately in many deployments, the first-hop router does not run a PCP server, or DHCP cannot be used. These and other problems are described in detail in the Appendix. Appendix A.

This document follows a different approach: it establishes a well-known address for the PCP Server (TBD: this approach could easily be generalized in order to discover other services as well. But this is for further study). PCP clients are expected to send requests to this address during the PCP Server discovery process. A PCP Server configured with the anycast address could optionally redirect or return a list of unicast PCP Servers to the client.



## 2. PCP Server Discovery based on well-known IP Address

### 2.1. Well-Known PCP Server IP Address (WkPsdIPa)

IANA is requested to register a single IPv4 address 192.0.0.X (TBD) and a single IPv6 address 2001:YYYY::ZZZZ (TBD) within the respective Special Purpose Address Registries as the well-known IP anycast addresses for PCP Servers. These addresses are called WkPsdIPa (well-known PCP server discovery IP address(es)) in this document.

### 2.2. PCP Discovery Client behavior

PCP Clients that need to discover PCP servers should first send a PCP request to its default router. This is important because in the case of cascaded PCP Servers, all of them need to be discovered in order of hop distance from the client. The PCP client then SHOULD send a PCP request to the WkPsdIPa. PCP Clients must be prepared to receive an error and try other discovery methods.

### 2.3. PCP Discovery Server behavior

PCP Server can be configured to listen on the WkPsdIPa for incoming PCP requests.

PCP responses are sent from that same IANA-assigned address (see Page 5 of [RFC1546]).

### 3. Deployment Considerations

Network operators should install one or more PCP Servers as specified above. Depending on the network deployment scenario they may use IP routing tables, or other suitable mechanisms to direct PCP requests to one of these servers.

[TBD: explain in more detail] This works fine even with cascaded access routers with NATs. After each router hop the operator may decide whether to handle the discovery requests, e.g., using a static routing table entry, or whether let them flow "automatically" towards the Internet backbones using the default routing table entry.

#### 3.1. Multiple PCP Servers, Symmetric Routing

In the case of symmetric routing all inbound and outbound packets from a PCP client traverse the same PCP Server or controlled device. Multiple PCP Servers sharing an anycast address in a symmetric routing scenario are used for two purposes: ease of network configuration and redundancy. In the case of redundancy, If there is a network or routing change a PCP client might start interacting with a different PCP Server sharing the same anycast address. From a PCP Client point of view this would be the same as a PCP Server reboot and a PCP Client could find out about it by examining the Epoch field during the next PCP request or ANNOUNCE message.

#### 3.2. Multiple PCP Servers, Assymetric Routing

In the case of asymmetric routing inbound packets from a PCP client traverse a different PCP Server or controlled device than outbound packets. If these PCP Servers are firewalls, the PCP client would need to create mappings on both of them in order to properly communicate with other hosts. But if these PCP Servers share an anycast address the PCP Client will create mappings in only on, when in fact should create mapping on both of them.

Therefore in order to support this scenario we propose a new option for the ANNOUNCE opcode. This will allow a PCP Client to request from a PCP Server a list of unicast IP addresses associated with other PCP Servers. The client can then proceed to create mappings on these PCP Servers using their unicast addresses.

This Option:

Option Name: LIST\_PCP\_SRVS

Number: TBA (IANA)

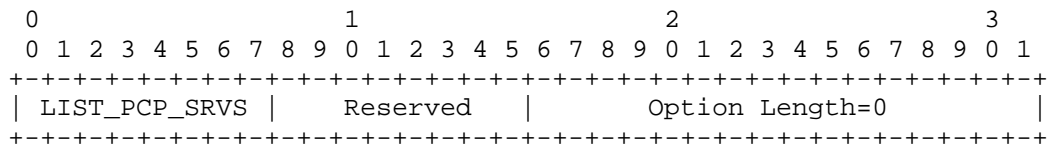
Purpose: Allows a PCP Client to request from a PCP Server a list of  
all PCP Servers configured

Valid for Opcodes: ANNOUNCE

Length: 0x0

May appear in: request and reply

Maximum occurrences in request: 1



The Reply from the PCP Server would be a list of IP addresses

Length in reply: 128 bits \* number of IP addresses

Maximum occurrences in reply: as many as fit within maximum PCP message size

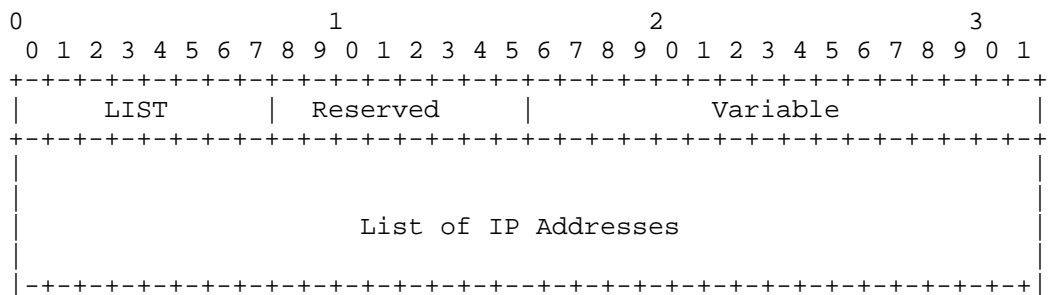


Figure 1: List of PCP Servers

#### 4. IANA Considerations

##### 4.1. Registration of IPv4 Special Purpose Address

IANA is requested to register a single IPv4 address in the IANA IPv4 Special Purpose Address Registry [RFC5736].

[RFC5736] itemizes some information to be recorded for all designations:

1. The designated address prefix.

Prefix: TBD by IANA. Prefix length: /32

2. The RFC that called for the IANA address designation.

This document.

3. The date the designation was made.

TBD.

4. The date the use designation is to be terminated (if specified as a limited-use designation).

Unlimited. No termination date.

5. The nature of the purpose of the designated address (e.g., unicast experiment or protocol service anycast).

protocol service anycast.

6. For experimental unicast applications and otherwise as appropriate, the registry will also identify the entity and related contact details to whom the address designation has been made.

N/A.

7. The registry will also note, for each designation, the intended routing scope of the address, indicating whether the address is intended to be routable only in scoped, local, or private contexts, or whether the address prefix is intended to be routed globally.

Typically used within a network operator's network domain, but in principle globally routable.

8. The date in the IANA registry is the date of the IANA action, i.e., the day IANA records the allocation.

TBD.

#### 4.2. Registration of IPv6 Special Purpose Address

IANA is requested to register a single IPv6 address in the IANA IPv6 Special Purpose Address Block [RFC4773].

[RFC4773] itemizes some information to be recorded for all designations:

1. The designated address prefix.

Prefix: TBD by IANA. Prefix length: /128

2. The RFC that called for the IANA address designation.

This document.

3. The date the designation was made.

TBD.

4. The date the use designation is to be terminated (if specified as a limited-use designation).

Unlimited. No termination date.

5. The nature of the purpose of the designated address (e.g., unicast experiment or protocol service anycast).

protocol service anycast.

6. For experimental unicast applications and otherwise as appropriate, the registry will also identify the entity and related contact details to whom the address designation has been made.

N/A.

7. The registry will also note, for each designation, the intended routing scope of the address, indicating whether the address is intended to be routable only in scoped, local, or private contexts, or whether the address prefix is intended to be routed globally.

Typically used within a network operator's network domain, but in principle globally routable.

8. The date in the IANA registry is the date of the IANA action, i.e., the day IANA records the allocation.

TBD.

#### 4.3. PCP Option

The following PCP Option should be allocated:

LIST\_PCP\_SRVS

## 5. Security Considerations

TBD

## 6. Acknowledgements

Ted Lemon for insightful DHCP discussions and Dave Thaler for pointing out the asymmetric routing case.



## 7. References

### 7.1. Normative References

- [RFC1546] Partridge, C., Mendez, T., and W. Milliken, "Host Anycasting Service", RFC 1546, November 1993.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.
- [RFC2732] Hinden, R., Carpenter, B., and L. Masinter, "Format for Literal IPv6 Addresses in URL's", RFC 2732, December 1999.
- [RFC3958] Daigle, L. and A. Newton, "Domain-Based Application Service Location Using SRV RRs and the Dynamic Delegation Discovery Service (DDDS)", RFC 3958, January 2005.
- [RFC4773] Huston, G., "Administration of the IANA Special Purpose IPv6 Address Block", RFC 4773, December 2006.
- [RFC5736] Huston, G., Cotton, M., and L. Vegoda, "IANA IPv4 Special Purpose Address Registry", RFC 5736, January 2010.

### 7.2. Informative References

- [DhcpRequestParams] OpenFlow, "OpenFlow Switch Specification", February 2011, <<http://msdn.microsoft.com/en-us/library/windows/desktop/aa363298%28v=vs.85%29.aspx>>.
- [I-D.chen-pcp-mobile-deployment] Chen, G., Cao, Z., Boucadair, M., Ales, V., and L. Thiebaut, "Analysis of Port Control Protocol in Mobile Network", draft-chen-pcp-mobile-deployment-04 (work in progress), July 2013.
- [I-D.ietf-dhc-container-opt] Droms, R. and R. Penno, "Container Option for Server Configuration", draft-ietf-dhc-container-opt-07 (work in progress), April 2013.
- [I-D.ietf-pcp-base] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)",

draft-ietf-pcp-base-29 (work in progress), November 2012.

[I-D.ietf-pcp-dhcp]

Boucadair, M., Penno, R., and D. Wing, "DHCP Options for the Port Control Protocol (PCP)", draft-ietf-pcp-dhcp-08 (work in progress), August 2013.

## Appendix A. Problems with Other Discovery methods

Several algorithms have been specified that allows PCP Client to discover the PCP Servers on a network . However, each of this approaches has technical or operational issues that will hinder the fast deployment of PCP.

### A.1. DHCP PCP Options

There are two problems with DHCP Options: DHCP Server on Home Gateways (HGW) and Operating Systems DHCP clients

Currently what the HGW does with the options it receives from the ISP is not standardized in any general way. As a matter of practice, the HGW is most likely to use its own customer-LAN-facing IP address for the DNS server address. As for other options, it's free to offer the same values to the client, offer no value at all, or offer its own IP address if that makes sense, as it does (sort of) for DNS.

In scenarios where PCP Server resides on ISP network and is intended to work with arbitrary home gateways that don't know they are being used in a PCP context, that won't work, because there's no reason to think that the HGW will even request the option from the DHCP server, much less offer the value it gets from the server on the customer-facing LAN. There is work on the DHC WG to overcome some of these limitations [I-D.ietf-dhc-container-opt] but in terms of deployment it also needs HGW to be upgraded.

The problems with Operating Systems is that even if DHCP PCP Option were made available to customer-facing LAN, host stack DHCP enhancements are required to process or request new DHCP PCP option. One exception is Windows [DhcpRequestParams]

Finally, in the case of IPv6 there are networks where there is DHCPv6 infrastructure at all or some hosts do not have a DHCPv6 client.

### A.2. Default Router

If PCP server does not reside in first hop router, whether because subscriber has a existing home router or in the case of Wireless Networks (3G, LTE) [I-D.chen-pcp-mobile-deployment], trying to send a request to default router will not work.

### A.3. User Input

A regular subscriber can not be expected to input IP address of PCP Server or network domain name. Moreover, user can be at a Wi-Fi hotspot, Hotel or related. Therefore relying on user input is not

reliable.

#### A.4. Domain Name System Based

There are three separate category of problems with NAPTR [RFC3958]

1. End Points: It relies on PCP client determining the domain name and supporting certain DNS queries
2. DNS Servers: DNS server need to be provisioned with the necessary records
3. CPEs: CPEs might interfere with DNS queries and the DHCP domain name option conveyed by ISP that could be used to bootstrap NAPTR might not be relayed to home network.

Authors' Addresses

Sebastian Kiesel  
University of Stuttgart Computing Center  
Allmandring 30  
Stuttgart 70550  
Germany

Email: [ietf-pcp@skiesel.de](mailto:ietf-pcp@skiesel.de)  
URI: <http://www.rus.uni-stuttgart.de/nks/>

Reinaldo Penno  
Cisco Systems  
170 West Tasman Dr  
San Jose CA  
USA

Email: [repenno@cisco.com](mailto:repenno@cisco.com)



Network Working Group  
Internet-Draft  
Intended status: Best Current Practice  
Expires: October 15, 2013

T. Lemon  
Nominum, Inc.  
April 13, 2013

Customizing DHCP Configuration on the Basis of Network Topology  
draft-lemon-dhc-topo-conf-01

Abstract

DHCP servers have evolved over the years to provide significant functionality beyond that which is described in the DHCP base specifications. One aspect of this functionality is support for context-specific configuration information. This memo describes some such features and makes recommendations as to how they can be used.

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 October 15, 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 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. Locality . . . . .	2
3. Simple Subnetted Network . . . . .	5
4. Regional Configuration Example . . . . .	6
5. Dynamic Lookup . . . . .	7
6. Acknowledgments . . . . .	8
7. Security Considerations . . . . .	8
8. IANA Considerations . . . . .	8
9. References . . . . .	8
9.1. Normative References . . . . .	8
9.2. Informative References . . . . .	9
Author's Address . . . . .	9

## 1. Introduction

The DHCPv4 [RFC2131] and DHCPv6 [RFC3315] protocol specifications describe how addresses can be allocated to clients based on network topology information provided by the DHCP relay infrastructure. Address allocation decisions are integral to the allocation of addresses and prefixes in DHCP.

The DHCP protocol also describes mechanisms for provisioning devices with additional configuration information; for example, DNS [RFC1034] server addresses, default DNS search domains, and similar information.

Although it was the intent of the authors of these specifications that DHCP servers would provision devices with configuration information appropriate to each device's location on the network, this practice was never documented, much less described in detail.

Existing DHCP server implementations do in fact provide such capabilities; the goal of this document is to describe those capabilities for the benefit both of operators and of protocol designers who may wish to use DHCP as a means for configuring their own services, but may not be aware of the capabilities provided by modern DHCP servers.

## 2. Locality

Figure 1 illustrates a simple hierarchy of network links with Link D serving as a backbone to which the DHCP server is attached.



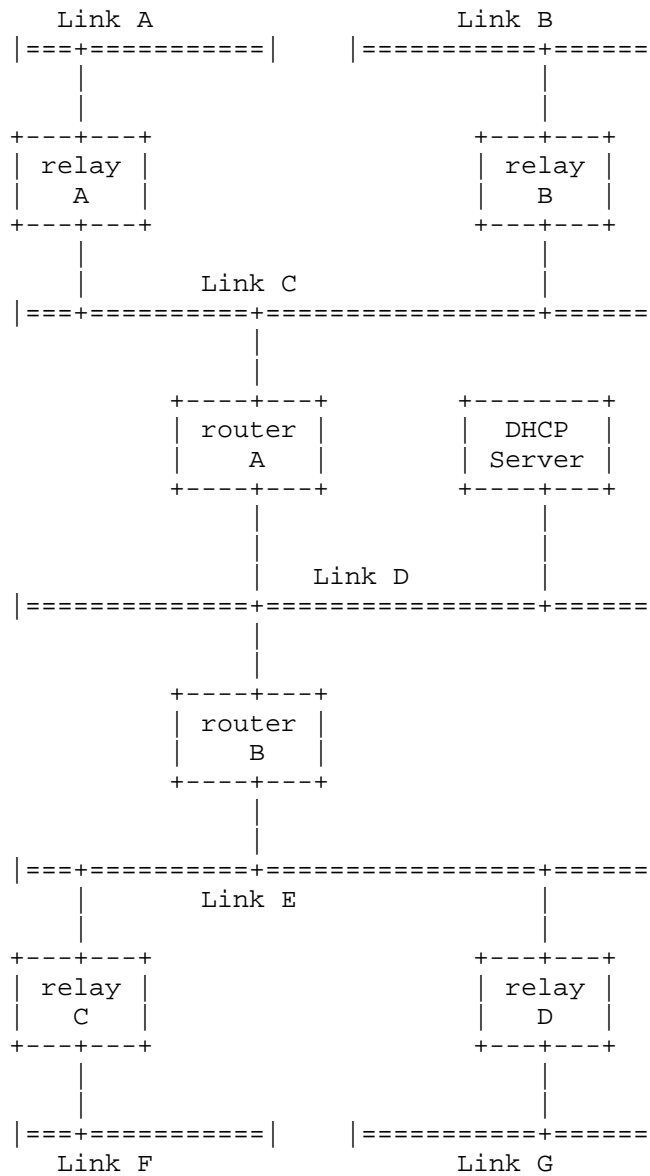


Figure 1

This diagram allows us to represent a variety of different network configurations and illustrate how existing DHCP servers can provide configuration information customized to the particular location from which a client is making its request.

It's important to understand the background of how DHCP works when considering this diagram. DHCP clients are assumed not to have routable IP addresses when they are attempting to obtain configuration information.

The reason for making this assumption is that one of the functions of DHCP is to bootstrap the DHCP client's IP address configuration; if the client does not yet have an IP address configuration, it cannot route packets to an off-link DHCP server, and so some kind of relay mechanism is required.

The details of how this works are different between DHCPv4 and DHCPv6, but the essence is the same: whether or not the client actually has an IP configuration, it generally communicates with the DHCP server by sending its requests to a DHCP relay agent on the local link; this relay agent, which has a routable IP address, then forwards the DHCP requests to the DHCP server. In some cases in DHCPv4, when a DHCP client has a routable IPv4 address.

In either case, the DHCP server is able to obtain an IP address that it knows is on-link for the link to which the DHCP client is connected: either the DHCPv4 client's routable IPv4 address, or the relay agent's IP address on the link to which the client is connected.

DHCPv6 also has support for more finely grained link identification, using Lightweight DHCPv6 Relay Agents [RFC6221] (LDRA). In this case, in addition to receiving an IPv6 address that is on-link for the link to which the client is connected, the DHCPv6 server also receives an Interface Identifier option from the relay agent that can be used to more precisely identify the client's location on the network.

What this means in practice is that the DHCP server in all cases has sufficient information to pinpoint, at the very least, the layer 3 link to which the client is connected, and in some cases which layer 2 link the client is connected to, when the layer 3 link is aggregated out of multiple layer 2 links.

In all cases, then, the DHCP server will have a link-identifying IP address, and in some cases it may also have a link-specific identifier. It should be noted that there is no guarantee that the link-specific identifier will be unique outside the scope of the link-identifying IP address.

It is also possible for link-specific identifiers to be nested, so that the actual identifier that identifies the link is an aggregate of two or more link-specific identifiers sent by a set of LDRA's in a

chain; in general this functions exactly as if a single identifier were received from a single LDRA, so we do not treat it specially in the discussion below, but sites that use chained LDRA configurations will need to be aware of this when configuring their DHCP servers.

Routable IP address: an IP address with a scope of use wider than the local link.

So let's examine the implications of this in terms of how a DHCP server can deliver targeted supplemental configuration information to DHCP clients.

### 3. Simple Subnetted Network

Consider Figure 1 in the context of a simple subnetted network. In this network, there are four leaf subnets: links A, B, F and G, on which DHCP clients will be configured. In a simple network like this, there may be no need for link-specific configuration in DHCPv6, since local routing information is delivered through router advertisements.

However, in IPv4, it is very typical to configure the default route using DHCP; in this case, the default route will be different on each link. In order to accomplish this, the DHCP server will need a link-specific configuration for the default route.

To illustrate, we will use an example from a hypothetical DHCP server that uses a simple JSON notation for configuration. Although we know of no DHCP server that uses this specific syntax, every commercial DHCP server provides similar functionality.

```
{ "prefixes":  
  { "10.0.0.0/24": { "options": { "routers": ["10.0.0.1"] }  
    "on-link": ["a"] } }  
  { "10.0.1.0/24": { "options": { "routers": ["10.0.1.1"] }  
    "on-link": ["b"] } }  
  { "10.0.2.0/24": { "options": { "routers": ["10.0.2.1"] }  
    "on-link": ["f"] } }  
  { "10.0.3.0/24": { "options": { "routers": ["10.0.3.1"] }  
    "on-link": ["g"] } } }
```

Figure 2

In figure 2, we see a configuration example for this scenario: a set of prefixes, each of which has a set of options and a list of links for which it is on-link. We have defined one option for each prefix:

a routers option. This option contains a list of values; each list only has one value, and that value is the IP address of the router specific to the prefix.

When the DHCP server receives a request, it searches the list of prefixes for one that encloses the link-identifying IP address provided by the client or relay agent. The DHCP server then examines the options list associated with that prefix and returns those options to the client.

So for example a client connected to link A in the example would have a link-identifying IP address within the 10.0.0.0/24 prefix, so the DHCP server would match it to that prefix. Based on the configuration, the DHCP server would then return a routers option containing a single IP address: 10.0.0.1. A client on link F would have a link-identifying address in the 10.0.2.0/24 prefix, and would receive a routers option containing the IP address 10.0.2.1.

#### 4. Regional Configuration Example

In this example, link C is a regional backbone for an ISP. Link E is also a regional backbone for that ISP. Relays A, B, C and D are PE routers, and Links A, B, F and G are actually link aggregators with individual layer 2 circuits to each customer—for example, the relays might be DSLAMs or cable head-end systems. At each customer site we assume there is a single CPE device attached to the link.

We further assume that links A, B, F and G are each addressed by a single prefix, although it would be equally valid for each CPE device to be numbered on a separate prefix.

In a real-world deployment, there would likely be many more than two PE routers connected to each regional backbone; we have kept the number small for simplicity.

In this example, the goal is to configure all the devices within a region with server addresses local to that region, so that service traffic does not have to be routed between regions unnecessarily.

```
{ "prefixes":
  { "2001:DB8:0:0::/40": { "on-link": ["A"] } },
  { "2001:DB8:100:0::/40": { "on-link": ["B"] } },
  { "2001:DB8:200:0::/40": { "on-link": ["F"] } },
  { "2001:DB8:300:0::/40": { "on-link": ["G"] } } },

{ "links":
  { "A": { "region": "omashu" },
    "B": { "region": "omashu" },
```

```
"F": {"region": "gaoling"},
"G": {"region": "gaoling"}}

{"regions":
  {"omashu": {"options": {"sip-servers": ["sip.omashu.example.org"],
    "dns-servers": ["dns1.omashu.example.org",
      "dns2.omashu.example.org"]}},
    "gaoling": {"options": {"sip-servers": ["sip.gaoling.example.org"],
      "dns-servers": ["dns1.gaoling.example.org",
        "dns2.gaoling.example.org"]}}}}
```

Figure 3

In this example, when a request comes in to the DHCP server with a link-identifying IP address in the 2001:DB8:0:0::/40 prefix, it is identified as being on link A. The DHCP server then looks on the list of links to see what region the client is in. Link A is identified as being in omashu. The DHCP server then looks up omashu in the set of regions, and discovers a list of region-specific options.

The DHCP server then resolves the domain names listed in the options and sends a sip-server option containing the IP addresses that the resolver returned for sip.omashu.example.org, and a dns-server option containing the IP addresses returned by the resolver for dns1.omashu.example.org and dns2.omashu.example.org.

Similarly, if the DHCP server receives a request from a DHCP client where the link-identifying IP address is contained by the prefix 2001:DB8:300:0::/40, then the DHCP server identifies the client as being connected to link G. The DHCP server then identifies link G as being in the gaoling region, and returns the sip-servers and dns-servers options specific to that region.

As with the previous example, the exact configuration syntax and structure shown above does not precisely match what existing DHCP servers do, but the behavior illustrated in this example can be accomplished with all existing commercial DHCP servers.

## 5. Dynamic Lookup

In the Regional example, the configuration listed several domain names as values for the sip-servers and dns-servers options. The wire format of both of these options contains one or more IPv6 addresses--there is no way to return a domain name to the client.

This was understood to be an issue when the original DHCP protocol was defined, and historical implementations even from the very early days would accept domain names and resolve them. Some early DHCP implementations, particularly those based on earlier BOOTP implementations, had very limited capacity for reconfiguration.

However, all modern commercial DHCP servers handle name resolution by querying the resolver each time a DHCP packet comes in. This means that if DHCP servers and DNS servers are managed by different administrative entities, there is no need for the administrators of the DHCP servers and DNS servers to communicate when changes are made. When changes are made to the DNS server, these changes are immediately and automatically adopted by the DHCP server. Similarly, when DHCP server configurations change, DNS server administrators need not be aware of this.

## 6. Acknowledgments

Thanks to Dave Thaler for suggesting that even though "everybody knows" how DHCP servers are deployed in the real world, it might be worthwhile to have an IETF document that explains what everybody knows, because in reality not everybody is an expert in how DHCP servers are administered.

## 7. Security Considerations

This document explains existing practice with respect to the use of Dynamic Host Configuration Protocol [RFC2131] and Dynamic Host Configuration Protocol Version 6 [RFC3315]. The security considerations for these protocols are described in their specifications and in related documents that extend these protocols. This document introduces no new functionality, and hence no new security considerations.

## 8. IANA Considerations

The IANA is hereby absolved of any requirement to take any action in relation to this document.

## 9. References

### 9.1. Normative References

- [RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

- [RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

## 9.2. Informative References

- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, November 1987.
- [RFC6221] Miles, D., Ooghe, S., Dec, W., Krishnan, S., and A. Kavanagh, "Lightweight DHCPv6 Relay Agent", RFC 6221, May 2011.

## Author's Address

Ted Lemon  
Nominum, Inc.  
2000 Seaport Blvd  
Redwood City, CA 94063  
USA

Phone: +1-650-381-6000  
Email: Ted.Lemon@nominum.com

PCP Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: January 11, 2014

T. Reddy  
P. Patil  
D. Wing  
R. Penno  
Cisco  
July 10, 2013

PCP Authentication Requirements  
draft-reddy-pcp-auth-req-04

Abstract

In an attempt to reach consensus on a PCP authentication mechanism, this document describes requirements for PCP authentication. It is hoped this can serve as the basis for a comparison of PCP authentication mechanisms.

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 January 11, 2014.

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 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 . . . . .	3
2. Terminology . . . . .	3
3. Requirements . . . . .	3
4. Third Party Authorization . . . . .	5
5. Other recommendations . . . . .	7
6. IANA Considerations . . . . .	7
7. Security Considerations . . . . .	7
8. References . . . . .	7
8.1. Normative References . . . . .	7
8.2. Informative References . . . . .	7
Appendix A. Change History . . . . .	8
A.1. Change from -01 to -02 . . . . .	8
A.2. Change from -02 to -03 . . . . .	8
A.3. Change from -03 to -04 . . . . .	8
Authors' Addresses . . . . .	8

## 1. Introduction

This document derives requirements for PCP Authentication from PCP deployment scenarios and scope described in [RFC6887] and other PCP drafts. The document focuses on requirements and does not make a suggestion on the authentication mechanism to be used to satisfy requirements.

## 2. Terminology

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 note uses terminologies defined in [RFC4949] such as realm, security association, identity, credential etc.

## 3. Requirements

REQ-1: PCP MUST provide client authentication. PCP client and server MUST also be able to mutually authenticate. Mutual authentication is especially necessary when the PCP server is located in a different administrative domain from the PCP client. Credentials to gain access to the network could be different from the credentials used to authenticate with the PCP server.

- \* The identity details of the client could be used by the PCP server to grant access to certain PCP opcodes or PCP options. For example GUESTS might not be permitted to use the MAP opcode and only ADMINISTRATOR might be permitted to use the THIRD\_PARTY option.

- \* The identity details of the client could be used for auditing.

REQ-2: PCP Authentication MUST generate security association for integrity protection of PCP request and response. This and all subsequent requirements are not applicable to multicast PCP responses like ANNOUNCE.

REQ-3: A PCP server MUST be able to indicate that a request will not be processed without authentication.

REQ-4: If a PCP client authenticates with a PCP server,

- A. The client MUST be able to verify the integrity and origin of responses from the server.

- B. The server MUST be able to send authenticated unsolicited responses.
- C. If a PCP response does not include integrity related to a current security association, then those messages MUST NOT be trusted without soliciting an integrity protected version.
- D. The server MUST be able to trigger reauthentication with the client. The unsolicited message for authentication trigger MUST be integrity protected if there is a valid unexpired SA.

REQ-5: It is important that PCP not leak privacy information between the PCP client and PCP server,

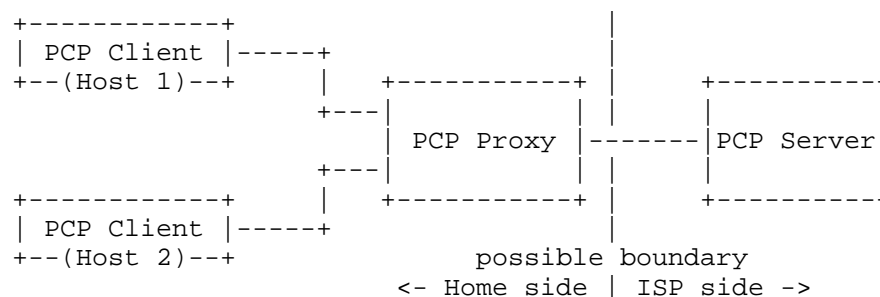
- A. The authentication mechanism MUST be able to keep credentials hidden from eavesdroppers on path between the client and server.
- B. Confidentiality of the PCP messages is OPTIONAL for PCP request and response of opcodes MAP, PEER, ANNOUNCE and options THIRD\_PARTY, PREFER\_FAILURE and FILTER as explained in [RFC6887]. Other PCP drafts MUST evaluate if confidentiality is OPTIONAL for new PCP opcodes and options introduced.
- C. PCP authentication SHOULD be immune to passive dictionary attacks.
- D. PCP Authentication MUST ensure that an attacker snooping PCP messages cannot guess the SA.

REQ-6: To ease troubleshooting and ensure fate sharing, PCP authentication and PCP messages MUST be multiplexed over the same port.

REQ-7: PCP authentication MUST accommodate authentication between administrative domains. For example, a PCP client may wish to communicate directly to an ISP's PCP server, even though the in-home CPE router does not support PCP. In this scenario the PCP client needs to directly authenticate with the ISP's PCP server.

REQ-8: For the scenarios described in REQ-7, the PCP authentication mechanism MUST be functional across address and port translation, including NAT64 and NAT44.

REQ-9: A PCP proxy that modifies PCP messages SHOULD have the ability to independently authenticate with the PCP client and PCP server. The presence of a PCP proxy hence requires two separately authenticates SAs. As a consequence, the PCP proxy:



- A. MUST be able to validate message integrity of PCP messages from the PCP server and client respectively.
- B. MUST be able to ensure message integrity after updating the PCP message for cases described in sections 6 and 7 of [I-D.ietf-pcp-proxy].

The PCP proxy MUST also permit authentication on only one side of the proxy. For example, a customer premises host may not authenticate with the PCP proxy but the PCP proxy may authenticate with the PCP server.

REQ-10: It is RECOMMENDED that PCP authentication support a mechanism where authentication on one port MUST be usable on other ports without requiring another authentication exchange for other ports. For example, there could multiple applications on the host like BitTorrent [BitTorrent], WebRTC[I-D.ietf-rtcweb-overview]/SIP [RFC3261] using PCP. Multiple authentication exchanges increase load on the PCP server and chatter on the network. For example, if 'N' messages are to be exchanged for PCP authentication and 'M' independent applications implement their own PCP client, a total of N\*M messages have to be exchanged and 'M' number of SAs maintained for each host.

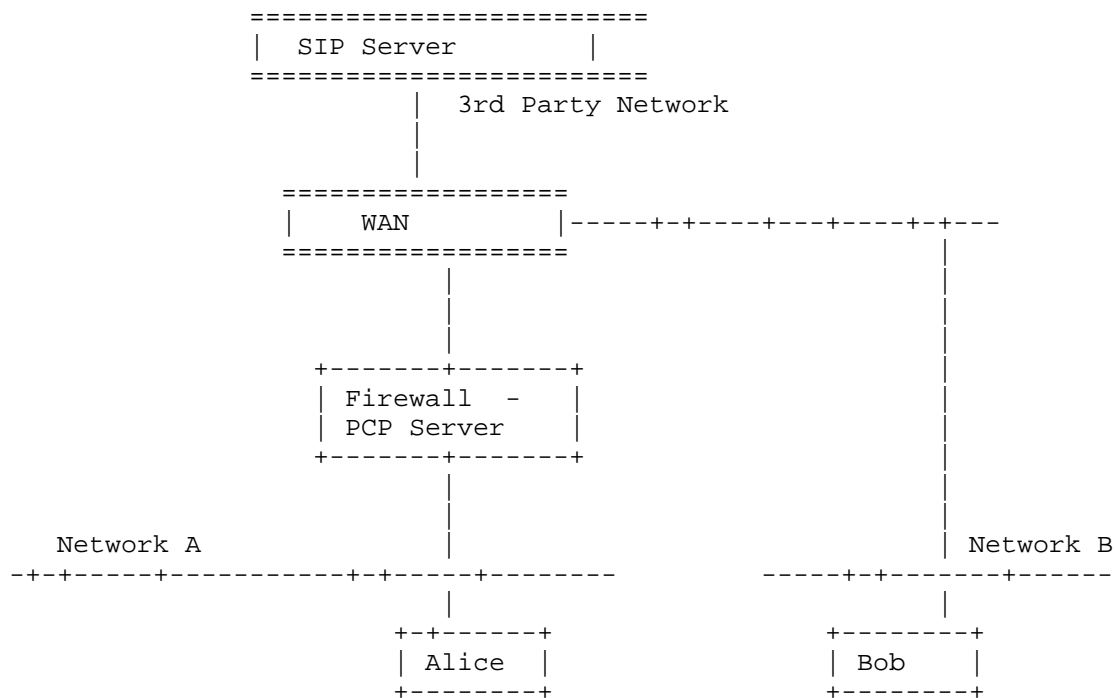
REQ-11: It is RECOMMENDED to choose a widely deployed authentication technique with known security properties rather than inventing a new authentication mechanism.

REQ-12: Changes in PCP to accommodate authentication SHOULD be minimal so that updates and additions to the authentication mechanism have minimal bearing on modifying PCP.

#### 4. Third Party Authorization

REQ-13: In addition to a two party authentication that has been discussed in this draft, a mechanism for third party authorization

MUST also be supported. This is applicable in cases where a third party authorizes the use of a resource on a PCP server for a desired PCP client. For example, as depicted in Figure 1 , a PCP request to a PCP capable firewall authorized by a SIP proxy rather than by virtue of the end user making the PCP request. The PCP server is to permit a PCP MAP request from the PCP client if the user is making a SIP call with the Enterprise or a trusted SIP server in 3rd party network, otherwise do not allow MAP request from that particular user. In this scenario the first party is the user, second party is the PCP server (which is also the firewall) and the third party is the SIP server, where the user is authorized to use MAP request only when making a call using the trusted SIP Server.



Users : Alice, Bob

Figure 1: WebRTC server in a different administrative domain

## 5. Other recommendations

REQ-14: There SHOULD be support for a means to provide integrity protection without user authentication, i.e., an anonymous client should be able to verify a PCP server using server-side-only auth and as a consequence obtain an SA which will be used for PCP message integrity. For example, a client visiting foreign networks such as a hotel, hot spot etc where the client may gain access to the network but does not know the credentials to authenticate with the PCP server. The negotiation of SA should be secure such that the SA is only known to the anonymous client and PCP server.

## 6. IANA Considerations

This document does not require any action from IANA.

## 7. Security Considerations

This entire document is about security considerations for PCP.

## 8. References

### 8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4949] Shirey, R., "Internet Security Glossary, Version 2", RFC 4949, August 2007.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

### 8.2. Informative References

- [BitTorrent] "Cohen, B., "The BitTorrent Protocol Specification Version 11031", February 2008.", September 2012.
- [I-D.ietf-pcp-proxy] Boucadair, M., Penno, R., and D. Wing, "Port Control Protocol (PCP) Proxy Function", draft-ietf-pcp-proxy-03 (work in progress), June 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.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

## Appendix A. Change History

### A.1. Change from -01 to -02

- o Requirements reorganized based on commonality
- o New requirement 3(c(2)) added.

### A.2. Change from -02 to -03

- o Merged REQ-1 and REQ-7
- o Updated Section 5 "Other recommendations"

### A.3. Change from -03 to -04

- o Updated REQ-4, REQ-9 and REQ-14.

## Authors' Addresses

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

Email: tiredddy@cisco.com

Prashanth Patil  
Cisco Systems, Inc.  
Bangalore  
India

Email: [praspati@cisco.com](mailto:praspati@cisco.com)

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

Email: [dwing@cisco.com](mailto:dwing@cisco.com)

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

Email: [repenno@cisco.com](mailto:repenno@cisco.com)





PCP Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: January 04, 2014

D. Wing  
R. Penno  
T. Reddy  
Cisco  
July 03, 2013

PCP Flowdata Option  
draft-wing-pcp-flowdata-00

Abstract

This document defines a mechanism for a host to signal flow characteristics to the network, and the network to signal its ability to accommodate that flow back to the host. The mechanism defines a new PCP option for the existing MAP and PEER opcodes.

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 January 04, 2014.

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 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. Terminology . . . . .	3
3. PCP FLOWDATA Option . . . . .	3
3.1. Usage and Processing . . . . .	4
3.2. Generating a PCP Request with FLOWDATA Option . . . . .	5
3.3. Processing a Request with FLOWDATA Option . . . . .	6
3.4. Processing a Response with FLOWDATA Option . . . . .	7
3.5. Link or State Changes on PCP Server . . . . .	7
3.6. Conflict Resolution . . . . .	8
4. PCP FLOWDATA Option Data Fields . . . . .	9
5. FLOWDATA Interaction with PCP Proxy . . . . .	14
6. Network Authorization . . . . .	15
7. Scaling Considerations . . . . .	15
8. Security Considerations . . . . .	15
9. IANA Considerations . . . . .	15
10. Acknowledgements . . . . .	15
11. References . . . . .	15
11.1. Normative References . . . . .	15
11.2. Informative References . . . . .	16
Authors' Addresses . . . . .	16

## 1. Introduction

Access networks often have insufficient bandwidth or other characteristics that prevent some applications from functioning as well as desired. Although the quality of wireless and wired access networks continue to improve, those access networks are often constrained for various reasons. This document provides a mechanism to signal the application's network requirements to the access network, so that certain network flows can receive service that is differentiated from other network flows. With this mechanism, a host can request the network provide certain characteristics for a flow in both the upstream and downstream directions. The network authorizes the request and signals back to the host that it can (fully or partially) accommodate the flow. This sort of signaling is useful for long-lived flows such as interactive audio/video, streaming video, and network control traffic (call signaling, routing protocols).

In order to obtain such differentiated service from a network, many previous mechanisms have been created for hosts to convey flow information to the network. The mechanism described in this document has several useful properties:

- o Usable at the application level, without needing operating system support;

- o Abstracts layer 2 specifics, so host and applications can avoid layer 2-specific signaling;
- o Robust metadata support, to convey sufficient information to the network about the flow;
- o Differentiates service on the local network and the immediately adjacent access network, which is typically bandwidth constrained;
- o Deployable on a local network and its adjacent access link, without needing support of the remote host's network or support of the remote host;
- o Provides differentiated service for both directions of a flow, including flows that cross administrative boundaries (such as the Internet).

The mechanism described in this specification defines an extension to Port Control Protocol (PCP [RFC6887]). This may be surprising at first because PCP is considered as a protocol for managing mappings in NATs and firewalls. However, PCP does not require the network implement a NAT or to implement a firewall. This is an important point: this specification does not require the network operate a NAT, and does not require the network operate a firewall. At a high level, PCP provides bi-directional communication a flow to the network. PCP can recursively communicate flow information to a number of on-path devices using PCP itself ([I-D.cheshire-recursive-pcp], [I-D.ietf-pcp-proxy]) or using an SDN protocol. Such recursion provides the flow information to more devices on the path, allowing each of them to optimize the flow over their respective links.

## 2. Terminology

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].

## 3. PCP FLOWDATA Option

The FLOWDATA option described in this document allows a host to signal the bi-directional characteristics of a flow to its PCP server. After signaling, the PCP server determines if it can accommodate that flow, making configuration changes if necessary to accommodate the flow, and returns information in the FLOWDATA option indicating its ability to accommodate the described flow.

### 3.1. Usage and Processing

A host may want to indicate to the network the priority of a flow after the flow has been established (typical if the host is operating as a client) or before the flow has been established (typical if the host is operating as a server). Both of these are supported and depicted in the following diagrams.

The following diagram shows how a connection is first established and then the flow is prioritized. This allows for the fastest connection setup time with the server.

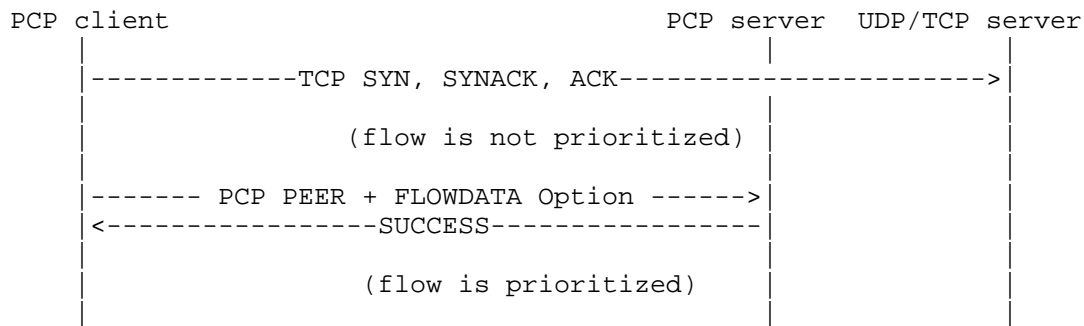


Figure 1: Message diagram, client connects first

The following diagram shows first asking the network to prioritize a flow, then establishing a flow. This is useful if the priority of the flow is more important than establishing the flow quickly.

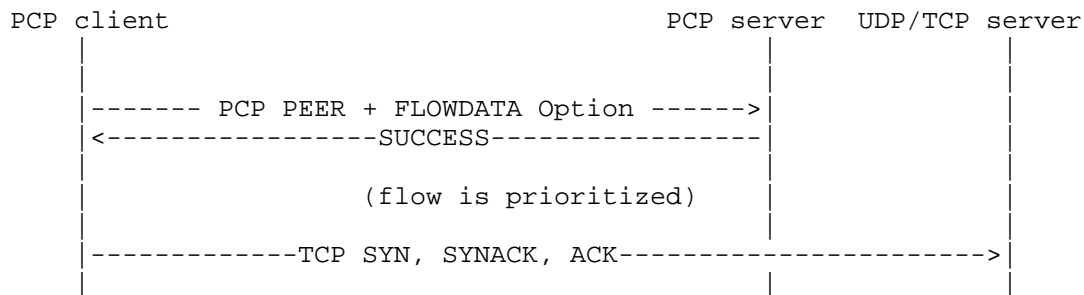


Figure 2: Message diagram, client sets priority first

The following diagram shows a PCP client getting a PCP MAP mapping for incoming flows with priority. This ensures that the PCP client has a mapping and all packets associated with the incoming TCP connections matching that mapping are prioritized. The PCP Client in this case could be a video server in a data center.

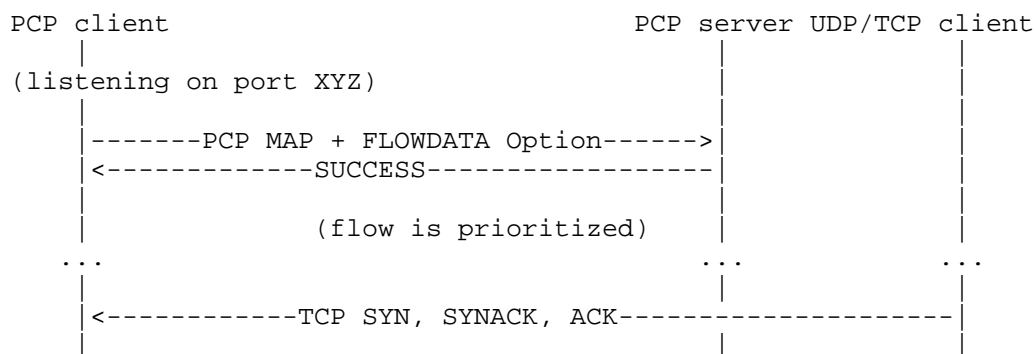


Figure 3: Message diagram, operating a server

The following diagram shows how two separate connections, where only one is active at a time, use the same instance identifier.

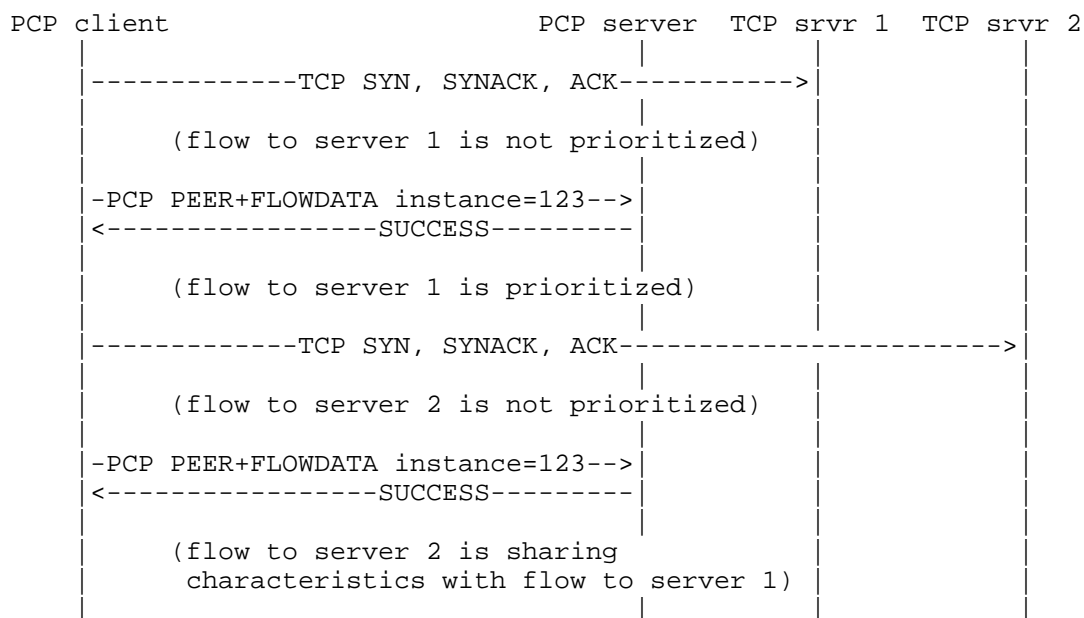


Figure 4: Message diagram with Instance Identifier

### 3.2. Generating a PCP Request with FLOWDATA Option

The PCP client first does all the processing described in Sections 8.1, 11.2, and 12.3 of [RFC6887] as appropriate for generating a MAP or PEER opcode request. Included in that request is a FLOWDATA option formatted as described in this document. For flows

established by the PCP client, the MAP or PEER request with FLOWDATA option can be sent before or after the PCP client has established any flows. For flows terminated by the PCP client (that is, when operating a server), the FLOWDATA option can be received and processed by the PCP server together with a MAP request or later during a MAP refresh request as shown in Figure 3.

### 3.3. Processing a Request with FLOWDATA Option

The PCP server performs processing in the order of the paragraphs below.

Upon receiving a PCP Request with FLOWDATA option first does the processing described in Section 8.2, 11.3, and 12.2 of [RFC6887], as appropriate for processing a MAP or PEER opcode request. If the MAP or PEER request contains the FLOWDATA option, the PCP server determines if the flow characteristics described in the FLOWDATA option can be accommodated by the network element controlled by the PCP server (that is, the router, NAT, or firewall controlled by the PCP server). To determine this, the PCP server might examine its static configuration and do bandwidth counting, or it might reconfigure the underlying network so that additional bandwidth is made available for this particular flow, or might perform other actions. If the PCP server determines the flow can only be partially accommodated, it returns values in the FLOWDATA fields that it can accommodate or returns 0 in those FLOWDATA fields where it has no information. In other words if the request indicated a low tolerance for delay but the PCP server and its controlled device determine that only high delay is available, the FLOWDATA response indicates high delay is available. The same sort of processing occurs on all of the FLOWDATA fields of the response (upstream and downstream delay tolerance, loss tolerance, jitter tolerance, minimum bandwidth, maximum bandwidth).

A PCP server that processes the FLOWDATA option is likely to create state for that flow (e.g., for bandwidth counting so that the bandwidth is returned to the bandwidth pool when the flow lifetime expires). Because Memory and other resources limit how much state can be created, the PCP server MUST implement a policy limit so that all state is not consumed by one host. It MAY also implement other limits, such as rate limits. The PCP server can implement its own policy to remove flows from its memory, such as FIFO. If a host has exceeded its quota, the existing error `USER_EX_QUOTA` SHOULD be returned.

If the PCP server can accommodate the flow as described in the FLOWDATA option, and can create the mapping as described in the MAP or PEER opcode, it sends a PCP response with the `SUCCESS` response

code, and includes the FLOWDATA option filled in according to Section 4.

After performing the above steps, the router creates state (if necessary for its implementation) and sends SUCCESS response code to the client with the data fields in the FLOWDATA option properly filled out.

### 3.4. Processing a Response with FLOWDATA Option

The PCP client performs processing in the order of the paragraphs below.

Upon receiving a PCP response, the PCP client performs the normal processing described in Section 8.3 of [RFC6887].

If the PCP response was SUCCESS (0), the PCP server has created a mapping. If the PCP response contains the FLOWDATA option, the FLOWDATA fields indicate if the network could accommodate the requested flow characteristics. The PCP client can use that information to influence the traffic it sends and receives on the network. For example, if the FLOWDATA response indicates the network can accommodate a flow of a certain downstream bandwidth, the PCP client will likely achieve the best result if it does not initiate a flow that exceeds that bandwidth.

Note to implementers: PCP allows the server to send multiple responses to a single request. This means that after sending a request and receiving a (positive) response, a subsequent response might be sent updating the information about the flow, should the network conditions change. The response could carry a FLOWDATA option where the data fields contain different values from the first response. This might occur, for example, if a competing high-bandwidth flow has finished, more bandwidth is available for this host; the DSL line rate might have improved (or degraded); the link speed may have been dynamically increased (or decreased). Thus, a PCP client should expect these subsequent responses and react accordingly.

### 3.5. Link or State Changes on PCP Server

After the PCP server has sent a SUCCESS response code including the FLOWDATA option, link characteristics might change causing a flow to no longer be accommodated by the network (e.g., link speed degrades) or for the PCP server to flush a flow from its list of prioritized flows (e.g., due to memory constraints). Whenever the network can no longer accommodate a flow, the PCP server MUST inform the PCP client by sending a mapping update response including an updated FLOWDATA



option, following the same procedure as a Mapping Update (Section 14.2 of [RFC6887]). As with PCP without FLOWDATA, if the PCP server loses all its state it will alert the PCP clients using rapid recovery (Section 14 of [RFC6887]) which also indicates loss of FLOWDATA state in the network.

Note: it is also possible that originally-requested flowdata could be accommodated (e.g., link speed improved). We might want to signal to endpoints that they should ask again for their originally-requested flowdata. This is for future study.

### 3.6. Conflict Resolution

It is possible that two hosts send requests with different thresholds for delay or jitter or different values for bandwidth in each direction, and their requests arrive at the same PCP server. An example is a media streamer and a television within the same home where one indicates its sending bandwidth is higher than the other indicates its receiving bandwidth. As another example, the indicated tolerance for delay might be different.

If this occurs, it is RECOMMENDED that the PCP server use the smaller bandwidth and stricter delay/loss tolerance (that is, the lower tolerance to delay or jitter), and issue a FLOWDATA update so both PCP clients receive the same information. The diagram below depicts a conflict message flow.

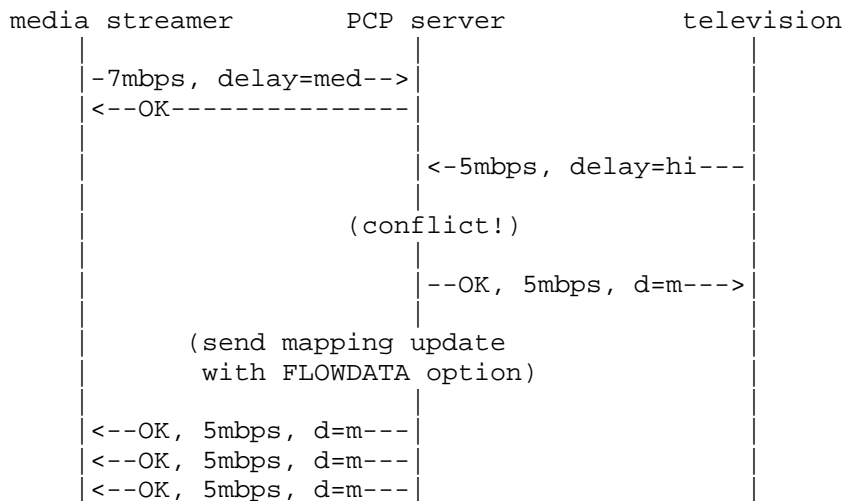


Figure 5: Message diagram, resolving conflict

It is also possible for one PCP client to think two flows should use the same instance identifier but the other PCP client to use different instance identifiers for those two flows. In this case, the operation of the PCP server (and the device it controls) is implementation specific.

#### 4. PCP FLOWDATA Option Data Fields

The FLOWDATA option has the following characteristics:

Option Name: FLOWDATA  
 Number: (to be assigned by IANA)  
 Purpose: Describe flow characteristics to the network  
 Valid for Opcodes: MAP, PEER  
 Length: 24 octets  
 May appear in: request. May appear in response only if it  
                   appeared in the associated request.  
 Maximum occurrences: 1

The FLOWDATA option request has the following format.

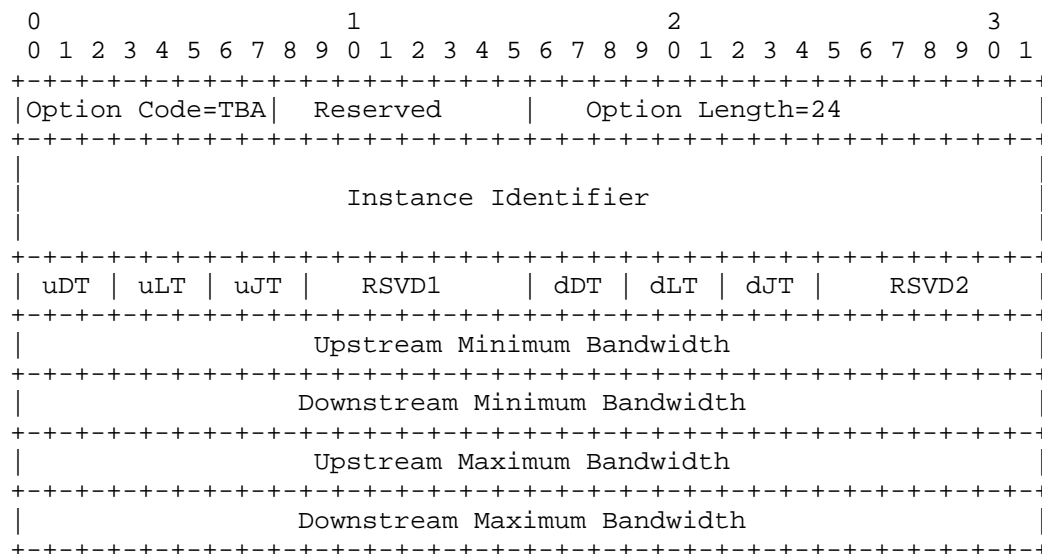


Figure 6: FLOWDATA Option

Description of the fields:

Instance Identifier: 96 bit identifier, unique to each simultaneously-active flow. This is a pseudo random number that MUST be generated following the procedures described in [RFC4086].

uDT: Upstream Delay Tolerance, 0=no information available, 1=very low, 2=low, 3=medium, 4=high.

uLT: Upstream Loss Tolerance, 0=no information available, 1=very low, 2=low, 3=medium, 4=high.

uJT: Upstream Jitter Tolerance, 0=no information available, 1=very low, 2=low, 3=medium, 4=high.

RSVD1: Reserved (7 bits), MUST be ignored on reception and MUST be 0 on transmission.

dDT: Downstream Delay Tolerance, 0=no information available, 1=very low, 2=low, 3=medium, 4=high.

dLT: Downstream Loss Tolerance, 0=no information available, 1=very low, 2=low, 3=medium, 4=high.

dJT: Downstream Jitter Tolerance, 0=no information available, 1=very low, 2=low, 3=medium, 4=high.

RSVD2: Reserved (7 bits), MUST be ignored on reception and MUST be 0 on transmission.

Upstream Minimum Bandwidth Measures bandwidth sent by the PCP client. Value is in octets per second. The value 0 means no information is available.

Downstream Minimum Bandwidth Measures bandwidth sent to the PCP client. Value is in octets per second. The value 0 means no information is available.

Upstream Maximum Bandwidth: Measures bandwidth sent by the PCP client. Value is in octets per second. The value 0 means no information is available.

Downstream Maximum Bandwidth Measures bandwidth sent to the PCP client. Value is in octets per second. The value 0 means no information is available.

The instance identifier accommodates network traffic where multiple 5-tuples exist for a particular data flow, but the bandwidth flows only over the aggregate of the multiple 5-tuples. A use-case for this identifier is TCP video streaming which retrieves short pieces

of the movie, often over separate TCP connections for load balancing, which would use the same Instance Identifier for each TCP connection. An instance is considered unique if the combination of the PCP client's IP address and the instance identifier are unique.

Discussion point: Minimum and maximum value of bandwidth is 1 byte per second to 4 gigaBYTES per second. We probably need to express higher bandwidth, and maybe also lower bandwidth?

Different applications have different needs for their flows. The following table is derived from [RFC4594] to serve as a guideline for tolerance to loss, delay and jitter for some sample applications.

Service Class Name	Traffic Characteristics	Tolerance to		
		Loss	Delay	Jitter
Network Control	Variable size packets, mostly inelastic short messages, but traffic can also burst (e.g., OSPF)	Low	Low	High
Telephony	Fixed-size small packets, constant emission rate, inelastic and low-rate flows (e.g., G.711, G.729)	Very Low	Very Low	Very Low
Signaling	Variable size packets, some what bursty short-lived flows	Low	Low	High
Multimedia Conferencing	Variable size packets, constant transmit interval, rate adaptive, reacts to loss	Low - Medium	Very Low	Low
Real-Time Interactive	RTP/UDP streams, inelastic, mostly variable rate	Low	Very Low	Low
Multimedia Streaming	Variable size packets, elastic with variable rate	Low - Medium	Medium	High
Broadcast Video	Constant and variable rate, inelastic, non-bursty flows	Very Low	Medium	Low
Low-Latency Data	Variable rate, bursty short-lived elastic flows	Low	Low - Medium	High
OAM	Variable size packets, elastic & inelastic flows	Low	Medium	High
High-Throughput Data	Variable rate, bursty long-lived elastic flows	Low	Medium - High	High
Standard	A bit of everything	0	0	0
Low-Priority Data	Non-real-time and elastic (e.g., network backup)	High	High	High

The FLOWDATA Option response has the following format. The fields indicate what the network can accommodate of the request.

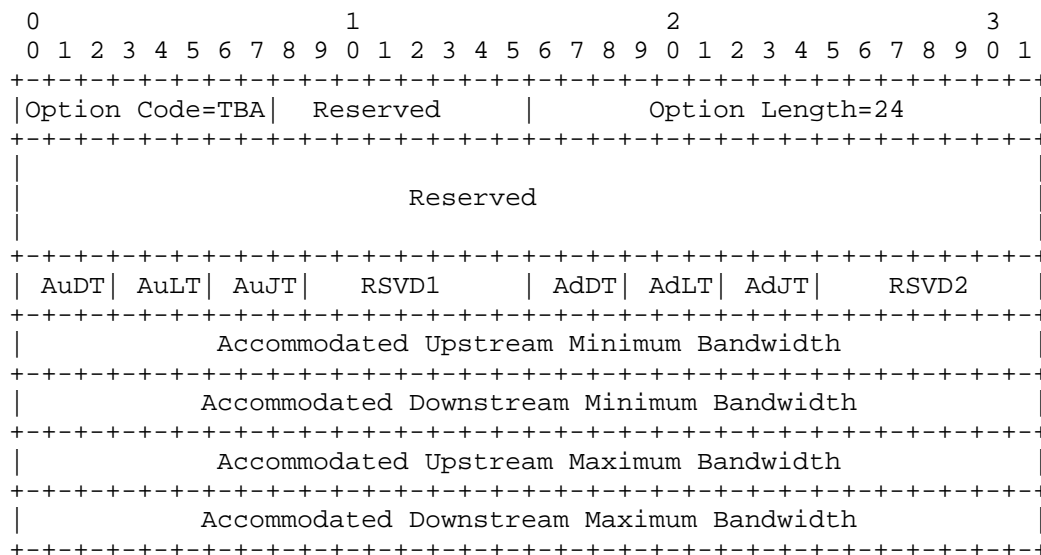


Figure 7: FLOWDATA Option

Description of the fields:

Reserved: 96 bits, MUST be ignored on reception and MUST be 0 on transmission.

AuDT: Accommodated Upstream Delay Tolerance, 0=no information available, 1=able to accommodate very low, 2=able to accommodate low, 3=able to accommodate medium, 4=able to accommodate high.

AuLT: Accommodated Upstream Loss Tolerance, 0=no information available, 1=able to accommodate very low, 2=able to accommodate low, 3=able to accommodate medium, 4=able to accommodate high.

AuJT: Accommodated Upstream Jitter Tolerance, 0=no information available, 1=able to accommodate very low, 2=able to accommodate low, 3=able to accommodate medium, 4=able to accommodate high.

RSVD1: Reserved (7 bits), MUST be ignored on reception and MUST be 0 on transmission.

AdDT: Accommodated Downstream Delay Tolerance, 0=no information available, 1=able to accommodate very low, 2=able to accommodate low, 3=able to accommodate medium, 4=able to accommodate high..

AdLT: Accommodated Downstream Loss Tolerance, 0=no information available, 1=able to accommodate very low, 2=able to accommodate low, 3=able to accommodate medium, 4=able to accommodate high.

AdJT: Accommodated Downstream Jitter Tolerance, 0=no information available, 1=able to accommodate very low, 2=able to accommodate low, 3=able to accommodate medium, 4=able to accommodate high.

RSVD2: Reserved (7 bits), MUST be ignored on reception and MUST be 0 on transmission.

Accommodated Upstream Minimum Bandwidth Bandwidth the network can accommodate for this flow, sent by the PCP client. Value in bytes per second. 0 means no information is available.

Accommodated Downstream Minimum Bandwidth Bandwidth the network can accommodate for this flow, sent to the PCP client. Value in bytes per second. 0 means no information is available.

Accommodated Upstream Maximum Bandwidth: Bandwidth the network can accommodate for this flow, sent by the PCP client. Value in bytes per second. 0 means no information is available.

Accommodated Downstream Maximum Bandwidth Bandwidth the network can accommodate for this flow, sent to the PCP client. Maximum Downstream bandwidth in bytes per second, 0 means no information is available.

## 5. FLOWDATA Interaction with PCP Proxy

The FLOWDATA option is optional to process. A PCP Proxy performs the functions described in [I-D.ietf-pcp-proxy], and if the PCP request contains the FLOWDATA option it also performs the functions described in this section.

The PCP request containing the FLOWDATA option SHOULD be proxied normally, so that the upstream PCP server can be aware of the entire request. The PCP proxy MAY have its own policies specific to the FLOWDATA option which require it to modify the FLOWDATA values request (e.g., reduce bandwidth for a certain PCP client).

After proxying the message containing FLOWDATA, when the PCP proxy receives the associated PCP response, the PCP proxy MAY reduce the bandwidth values or use worse (higher) values for delay, loss, or jitter tolerance. It MUST NOT increase the bandwidth or use better (lower) values for the delay, loss, or jitter tolerance.

## 6. Network Authorization

Oftentimes the endpoints themselves are not authorized to request network resources, but instead authorization has to first be obtained from a network element such as a call controller or policy element. To accommodate such deployments, third party authorization can be used with FLOWDATA . At a high level, this authorization works by the PCP client first obtaining a cryptographic token from the authorizing network element (e.g., call controller) and includes that token in the PCP request. The PCP server in the network validates the token and grants access.

## 7. Scaling Considerations

The network elements need only act upon those flows explicitly signaled by a PCP client, instead of all possible flows that a host generates.

Short lived flows (e.g., HTTP/1.0) or best-effort flows would receive little to no benefit from the signaling described in this document. As explained in Section 3.3, the PCP server will limit excessive flowdata requests, so hosts are encouraged to be conservative in how many flows are signaled with flowdata.

## 8. Security Considerations

On some networks, only certain users or certain applications are authorized to signal the priority of a flow. This authorization can be achieved with PCP client authentication [I-D.ietf-pcp-authentication].

## 9. IANA Considerations

IANA is requested to assign a new PCP option called FLOWDATA from the optional to process range (128-255) in the [pcp-iana] registry.

## 10. Acknowledgements

Thanks to Anca Zamfir for review comments.

## 11. References

### 11.1. 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-03 (work in progress), June 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC4086] Eastlake, D., Schiller, J., and S. Crocker, "Randomness Requirements for Security", BCP 106, RFC 4086, June 2005.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

## 11.2. Informative References

- [I-D.cheshire-recursive-pcp]  
Cheshire, S., "Recursive PCP", draft-cheshire-recursive-pcp-02 (work in progress), March 2013.
- [RFC4594] Babiarz, J., Chan, K., and F. Baker, "Configuration Guidelines for DiffServ Service Classes", RFC 4594, August 2006.
- [pcp-iana]  
IANA, "Port Control Protocol (PCP) Parameters", May 2013, <<http://www.iana.org/assignments/pcp-parameters/pcp-parameters.xml#options>>.

## Authors' Addresses

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

Email: [dwing@cisco.com](mailto:dwing@cisco.com)

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

Email: [repenno@cisco.com](mailto:repenno@cisco.com)

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

Email: [tiredy@cisco.com](mailto:tiredy@cisco.com)

PCP  
Internet-Draft  
Intended status: Standards Track  
Expires: October 4, 2014

D. Wing  
T. Reddy  
P. Patil  
R. Penno  
Cisco  
April 2, 2014

PCP Extension for Third Party Authorization  
draft-wing-pcp-third-party-authz-03

Abstract

It is often desirable for an application server to permit a flow across a firewall, as happens today when a firewall includes an Application Layer Gateway (ALG) function. However, an ALG has several weaknesses.

This document describes a cryptographic technique for an application server to permit a flow across a firewall. This technique uses OAuth and a new PCP option.

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 October 4, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 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. Problem Statement . . . . .	3
4. Solution Overview . . . . .	5
5. Obtaining a Token Using OAuth . . . . .	6
5.1. ACCESS_TOKEN Option . . . . .	7
5.2. Generating the ACCESS_TOKEN option . . . . .	9
5.3. PCP server processing ACCESS_TOKEN option . . . . .	10
5.4. Processing the PCP response . . . . .	11
6. PCP Server and Proxy behavior . . . . .	11
7. Usage with PCP Authentication mechanism . . . . .	12
8. Security Considerations . . . . .	12
9. IANA Considerations . . . . .	13
10. Acknowledgements . . . . .	13
11. References . . . . .	13
11.1. Normative References . . . . .	13
11.2. Informative References . . . . .	14
Authors' Addresses . . . . .	15

## 1. Introduction

It is desirable for a third party to permit flows across a firewall. A typical use-case is a SIP proxy (which is aware of legitimate calls) which is not co-located with a firewall. Today, this functionality is provided by a firewall implementing a SIP-aware Application Layer Gateway function, which examines the SIP signaling to that SIP proxy and opens the appropriate pinholes for the RTP media. This has disadvantages, as described in detail in section Section 3.

This document addresses requirement "Third Party Authorization" explained in section 4 of [I-D.reddy-pcp-auth-req].

This document proposes that a PCP [RFC6887] client communicate with an OAuth Authorization Server to obtain a cryptographic token for its media flow. That token is included in the PCP request and validated by the PCP server.

Note: There is no relationship with the THIRD\_PARTY option defined in [RFC6887], which serves a different purpose. THIRD\_PARTY Option for

MAP and PEER Opcodes described in [RFC6887] is only applicable when all entities i.e the PCP client, PCP server and Application Server, are deployed within the same administrative domain. Since PCP server does not listen on a public interface, an Application Server outside the site will not be able to use THIRD\_PARTY option to request services on behalf of the client.

## 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].

WebRTC Server: A web server that supports WebRTC [I-D.ietf-rtcweb-overview].

## 3. Problem Statement

To protect networks using real-time communications, firewalls or session border controllers [RFC5853] are typically deployed. Firewalls usually implement Application Layer Gateway functionality, which intercepts and analyzes session signaling traffic such as Session Initiation Protocol (SIP) [RFC3261] messages and creates a dynamic mapping to permit the corresponding media traffic. In particular, a firewall extracts media transport addresses, transport protocol and ports from session description and creates a dynamic mapping for media to flow through. This model will not work in the following cases:

1. Session signaling is end-to-end encrypted (say, using TLS).
2. Firewall does not understand the session signaling protocol, or extensions to the protocol, used by the endpoints.
3. Session signaling and media traverse different firewalls (e.g., signaling exits a network via one firewall whereas media exits a network via a different firewall)

When an enterprise deploys WebRTC, the above problems are relevant because:

1. Session signaling between WebRTC application running in a browser and a web server will use TLS.
2. WebRTC does not enforce a particular session signaling protocol; therefore, a firewall is unlikely to understand the signaling protocol.

3. Session signaling and peer-to-peer media may traverse different firewalls.

As a result firewalls block media traffic.

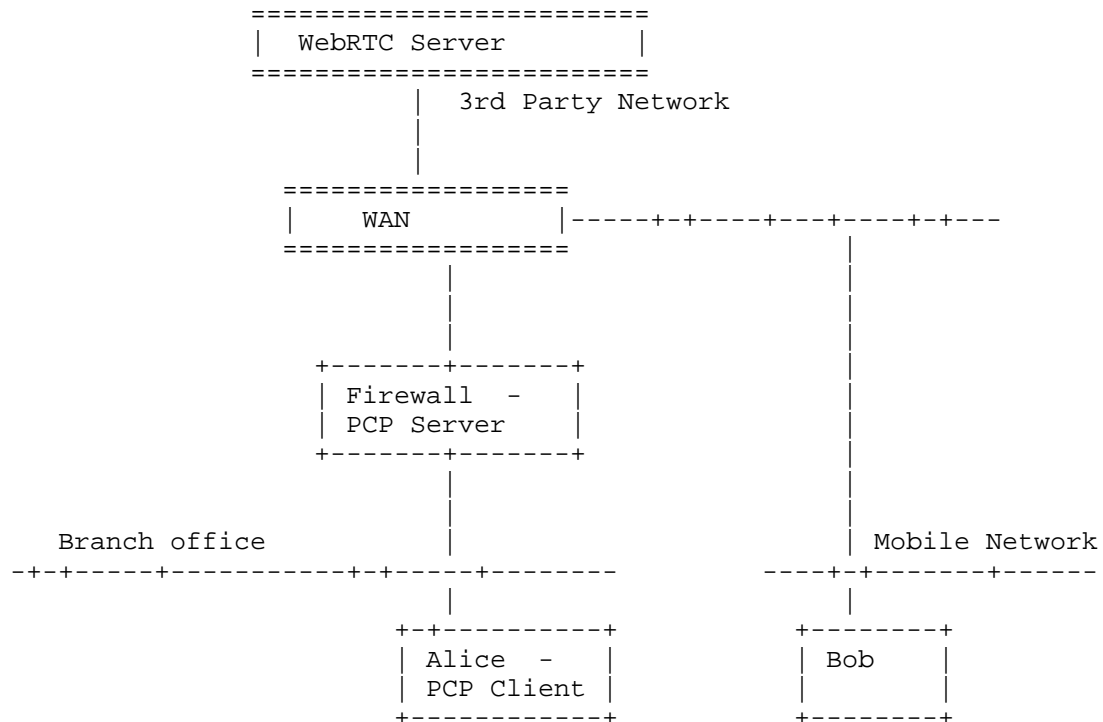
A mitigation to the problems above is for an enterprise to deploy a TURN server in the DMZ and have WebRTC clients use the TURN server. The use-case explained in Section 4.2.5.1 of [I-D.ietf-rtcweb-use-cases-and-requirements] refers to deploying a TURN [RFC5766] server to audit all media sessions from inside the company premises to any external peer.

However, using TURN for all such communication causes some problems for an enterprise network administrator :

- o Enterprise firewalls would typically have granular policies to permit calls initiated using selected WebRTC servers (Dr. Good) it trusts and block the rest (Dr. Evil).
- o A TURN server just provides a 5-tuple (source IP address, destination IP address, protocol number, source port number, and destination port number) for auditing and no other details of the WebRTC or SIP server being used to establish the call.
- o A TURN server could increase media latency as explained in section 4.1.2.2 of [RFC5245].
- o A 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 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 Other limitations of TURN are explained in section 2.6 of [RFC5766]. For example the value of Diffserv field may not be preserved, Explicit Congestion Notification (ECN) field may be reset etc.

#### 4. Solution Overview

In the below topology, the main functional elements involved are :



Users : Alice, Bob

WebRTC Server : OAuth 2.0 Authorization server

Figure 1: WebRTC server in a different administrative domain

In the topology, a WebRTC Server is deployed in a third party network trusted by the Enterprise. For the two endpoints to successfully establish media sessions, a firewall needs to permit ICE [RFC5245] connectivity checks and subsequent media traffic.

In such a scenario this specification proposes that a PCP client follows the steps described below:

1. The PCP client makes a PCP request without any authorization. If the PCP server returns an `AUTHORIZATION_REQUIRED` error message,

the PCP client concludes that the PCP server is mandating the use of third party authorization.

2. The PCP client then obtains a cryptographic token from an OAuth 2.0 Authorization server.
3. The PCP client sends a PCP request including the cryptographic token in the TOKEN\_ACCESS option, defined below. Alternatively, the PCP client could first obtain a cryptographic token from the OAuth 2.0 Authorization server and send the PCP request with the TOKEN\_ACCESS option by default.
4. The PCP server uses the TOKEN\_ACCESS option to perform third party authorization.

The technique proposed in the specification can be used by any other Application Function trusted by the network to permit time-bound, encrypted, peer-to-peer traffic.

#### 5. Obtaining a Token Using OAuth

This section explains OAuth 2.0 authorization framework [RFC6749] to solve the "Third Party Authorization" requirement explained in section 4 of [I-D.reddy-pcp-auth-req].

The following mapping of OAuth concepts to PCP is used :

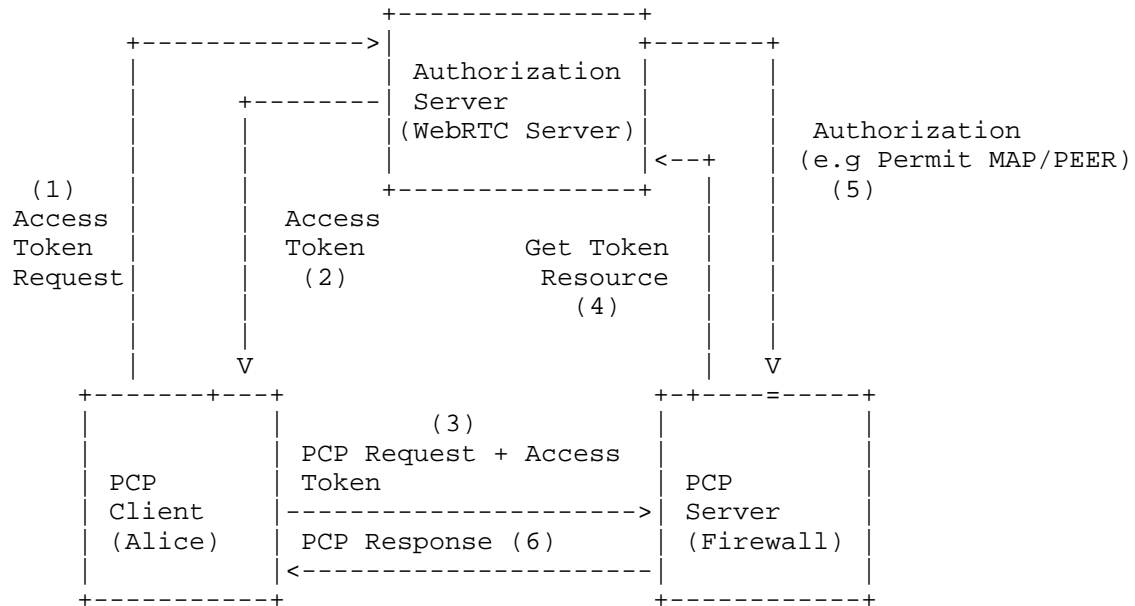
OAuth	PCP
Client	PCP Client
Resource owner	Authorization Server. For example the WebRTC server
Authorization server	Authorization server.
Resource server	PCP Server

Figure 2: OAuth terminology mapped to PCP terminology

Using the OAuth 2.0 authorization framework, a PCP client (third-party application) obtains limited access to a PCP server (resource server) on behalf of the WebRTC server (resource owner or authorization server). The PCP client requests access to resources controlled by the resource owner (WebRTC server) and hosted by the resource server (PCP server). The PCP client obtains an access



token, lifetime, and other access attributes like the PCP options and opcodes that the PCP client is permitted to use from the authorization server. The PCP client conveys the token in the PCP ACCESS\_TOKEN option to access the protected resources hosted by the resource server (PCP server). The PCP server validates the token and takes appropriate action e.g., allows the PCP request to create mappings on the PCP server.



User : Alice

Figure 3: Interactions

OAuth in [RFC6749] defines four grant types. This specification uses the OAuth grant type "Implicit" explained in section 1.3.2 of [RFC6749] where the PCP client is issued an access token directly. The scope of the access token explained in section 3.3 of [RFC6749] MUST be PCP.

#### 5.1. ACCESS\_TOKEN Option

This specification defines a new PCP ACCESS\_TOKEN Option that is described in Figure 4.

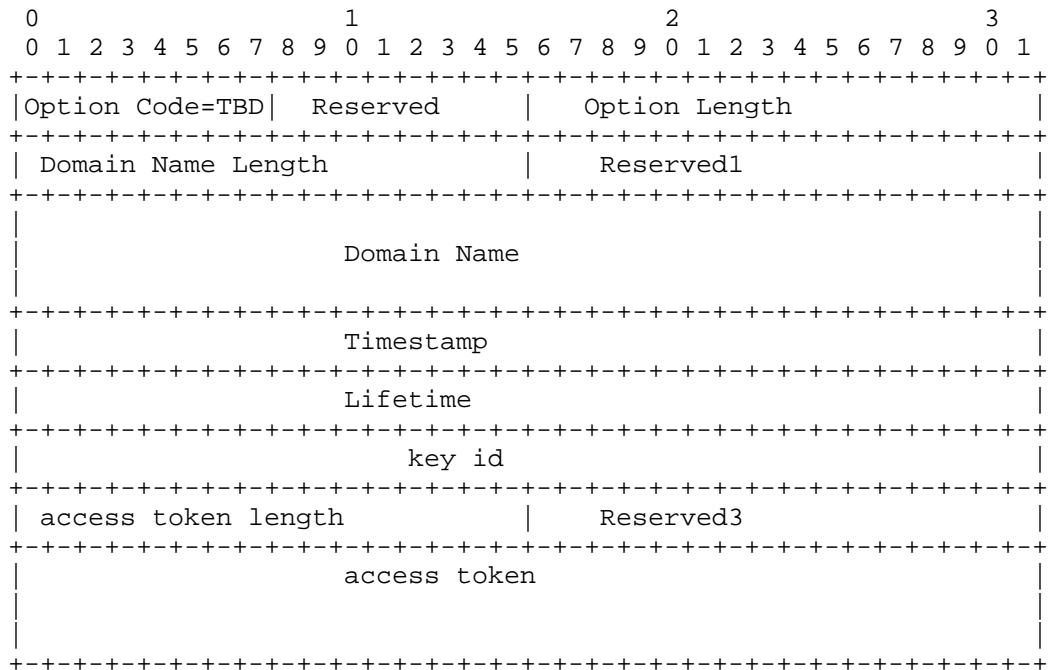


Figure 4: PCP ACCESS\_TOKEN Option

The fields are described below:

Option Length: 16 bits. Indicates the length of the enclosed data, in octets. Variable, but MUST NOT be 0.

Domain Name Length: Length of the 'Domain Name' field in octets.

Reserved1: set to 0 by sender and ignored by the receiver.

Server Domain Name: The domain name of the Authorized Server that generated the access token.

Timestamp: 64-bit unsigned integer field containing a timestamp. The value indicates the time since January 1, 1970, 00:00 UTC, by using a fixed point format. In this format, the integer number of seconds is contained in the first 48 bits of the field, and the remaining 16 bits indicate the number of 1/64K fractions of a second (Native format - Unix).

Lifetime: The lifetime of the access token since the response was generated, in seconds. For example, the value 3600 indicates one

hour. The Lifetime value SHOULD be equal to the "expires\_in" parameter defined in section 4.2.2 of [RFC6749].

key id: An ephemeral and unique key identifier generated by the authorization server. The authorization server MUST NOT generate the same key identifier twice within the lifetime of the access token.

access token length: Length of the access token field in octets. OAuth does not impose any limitation on the length of the access token but since PCP messages cannot exceed 1100 octets (Section 7 of [RFC6887]), access token length needs to be restricted to fit within the maximum PCP message size. The access token is defined in section 1.4 of [RFC6749]. TBD : what is the recommended/maximum token length for PCP. We need a discussion of this maximum length and analysis of what that means

Reserved3: set to 0 by sender and ignored by the receiver.

access token: The access token issued by the authorization server.

Option Name: ACCESS\_TOKEN

Number: TBA in the mandatory-to-process range (IANA)

Purpose: This option conveys the token granted by the authorization server for third party authorization.

Valid for Opcodes: MAP, PEER

May appear in : request.

Maximum occurrences : 1

## 5.2. Generating the ACCESS\_TOKEN option

The mechanism used by an OAuth client to obtain a token from the OAuth authorization server is outside the scope of this document. The OAuth client could obtain the token via in-band signaling or an exclusive out-of-band protocol. This specification uses the token type Handle described in [RFC6819]. A handle token is a reference to some internal data structure within the OAuth authorization server; the internal data structure contains the attributes of the token such as allowed PCP Opcode or PCP Option, etc. The PCP client, after receiving the access token from the OAuth authorization server, generates the ACCESS\_TOKEN option which is included in the PCP request to the PCP server.

### 5.3. PCP server processing ACCESS\_TOKEN option

A PCP server performs processing in the order described below.

When a PCP server receives a PCP request with an ACCESS\_TOKEN option, it will verify that the access token is valid. To address replay attacks, the PCP server MUST perform the following check :

When a PCP request with an ACCESS\_TOKEN Option is received, the received timestamp (TSnew in the Timestamp field) is checked and the cryptographic token is accepted if the timestamp is recent enough to the reception time of the PCP request, RDnew :

$$\text{Lifetime} + \text{Delta} > \text{abs}(\text{RDnew} - \text{TSnew})$$

The RECOMMENDED value for the allowed Delta is 5 seconds. If the timestamp is NOT within the boundaries then discard the PCP request with AUTHORIZATION-FAILED error response defined in [I-D.ietf-pcp-authentication].

After the validation described above, the PCP server communicates with the authorization server in order to validate the token and obtain token-bound data. The mechanism for communication is outside the scope of this document. The PCP server makes a request to the authorization server to validate the token but produces no other data with the request. If the token is successfully validated, the authorization server just returns the token bound authorization data in the response. The PCP server then matches this authorization data with what is requested in the PCP request sent by the PCP client. If the authorization sets match, the PCP server honors the PCP request made by the PCP client.

If the token is invalid or the request exceeds what is authorized by the token then the PCP server generates an AUTHORIZATION-FAILED error response. An example might be that an OAuth authorization server permits creating 5 mappings, and the PCP request made by the client is trying to create a 6th mapping.

Handle token type was selected for the following reasons :

1. The Authorization Server can inform the PCP server to revoke the access token after the call is terminated. This mechanism ensures that even if the PCP client does not close the dynamic mapping created, the PCP server based on the revocation notification from the Authorization Server can close the dynamic mapping.

2. A PCP-controlled Firewall with restrictive policies may also want to validate with the Authorization Server if the selected candidate pairs in the final offer/answer match the 5-tuple {dest addr, source addr, protocol, dest port, source port} sessions traversing the Firewall. This validation ensures that the PCP client is using the token only to send and receive the media streams finalized in the call to the remote peer. Thus the PCP server can make sure that the token cannot be used for anything else.
3. If PCP authentication [I-D.ietf-pcp-authentication] is used then the PCP server may also validate with the authorization server if the access token is issued and used by the same user or not.

Another approach, not discussed in this document, is a self-contained token where all the information necessary to authenticate the validity of the token is contained within the token itself. This approach has the benefit of avoiding a protocol between the PCP server and the OAuth authentication server for token validation, thus reducing latency. However, this approach has the drawback of needing a large PCP packet to accommodate the token. Because PCP messages are limited to 1100 octets, using the handle approach is more flexible and the trade-off for additional latency is reasonable. The other disadvantages of self-contained tokens, such as difficulties with revocation etc., are discussed in[RFC6819].

#### 5.4. Processing the PCP response

Upon receiving a PCP response, the PCP client performs the normal processing described in Section 8.3 of [RFC6887]. If the PCP response was SUCCESS (0), the PCP server has determined that the token is valid. If the PCP response was AUTHORIZATION-FAILED, it indicates that the token could be invalid, expired or the PCP request exceeded what is authorized by the token.

#### 6. PCP Server and Proxy behavior

The ACCESS\_TOKEN option is mandatory-to-process (its most significant bit is clear). Thus, per existing behavior described in [RFC6887], a PCP server receiving this option MUST return the error MALFORMED\_OPTION if the option contents are malformed, or UNSUPP\_OPTION if the option is unrecognized, unimplemented, or disabled, or if the client is not authorized to use the option.

A PCP Proxy MUST follow the rules mentioned in section of 3.4 of [I-D.ietf-pcp-proxy] when processing the ACCESS\_TOKEN option.

## 7. Usage with PCP Authentication mechanism

The following steps MUST be followed when PCP third party authorization is used with PCP authentication mechanism.

- o PCP client MUST send the access token after successful EAP authentication. This provides integrity protection for ACCESS\_TOKEN option.
- o If PCP Auth session lifetime expires before the authorization token expires and the PCP client, PCP server fail to trigger re-authentication then dynamic mappings created because of third party authorization MUST be deleted.

## 8. Security Considerations

Security considerations discussed in [RFC6887] and PCP authentication [I-D.ietf-pcp-authentication] are to be taken into account. If left unprotected the Authorization server could present a means for an attacker to poll a series of possible token values, fishing for a valid token. Therefore, the Authorization Server SHOULD issue special credentials to PCP server to access it and the communication between PCP server and Authorization server MUST be protected using TLS.

A PCP server will delete explicit dynamic mappings after the lifetime of the cryptographic token expires. The PCP client must obtain a new cryptographic token from the authorization server before the current token becomes invalid or expires. The PCP client must propagate the new cryptographic token to the PCP server to refresh lifetime of mappings before the current token becomes invalid or expires. The PCP server in addition to timestamp checking can also maintain a cache of used key id as an effective countermeasure against replay attacks.

Discussion: If the additional latency needs to be avoided and it is permissible to create a PCP mapping briefly for PCP clients, an implementation could create PCP mappings while the token is being validated. The PCP server could create a mapping immediately, send a PCP response and in parallel start verification of the token. If the verification request times out or returns a failure response, the PCP mapping can be destroyed and a PCP mapping update is sent to the PCP client. The PCP server while waiting for the validation response to arrive from Authorization server can either drop or buffer the traffic matching the mapping created.

## 9. IANA Considerations

We request IANA register the PCP option ACCESS\_TOKEN and the result code AUTHORIZATION\_REQUIRED in [pcp-registry].

## 10. Acknowledgements

Authors would like to thank Dave Thaler, Charles Eckel, Paul Jones, Dacheng Zhang, Anca Zamfir, Parthasarathi R and Suresh kumar for their comments and review.

## 11. References

### 11.1. Normative References

- [I-D.ietf-pcp-authentication]  
Wasserman, M., Hartman, S., and D. Zhang, "Port Control Protocol (PCP) Authentication Mechanism", draft-ietf-pcp-authentication-03 (work in progress), February 2014.
- [I-D.ietf-pcp-proxy]  
Perreault, S., Boucadair, M., Penno, R., Wing, D., and S. Cheshire, "Port Control Protocol (PCP) Proxy Function", draft-ietf-pcp-proxy-05 (work in progress), February 2014.
- [I-D.reddy-pcp-auth-req]  
Reddy, T., Patil, P., Wing, D., and R. Penno, "PCP Authentication Requirements", draft-reddy-pcp-auth-req-04 (work in progress), July 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.
- [RFC6407] Weis, B., Rowles, S., and T. Hardjono, "The Group Domain of Interpretation", RFC 6407, October 2011.
- [RFC6749] Hardt, D., "The OAuth 2.0 Authorization Framework", RFC 6749, October 2012.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

[pcp-registry]

IANA, , "Port Control Protocol (PCP) Parameters", May 2013, <<http://www.iana.org/assignments/pcp-parameters/pcp-parameters.xml>>.

## 11.2. Informative References

[I-D.ietf-rtcweb-overview]

Alvestrand, H., "Overview: Real Time Protocols for Brower-based Applications", draft-ietf-rtcweb-overview-09 (work in progress), February 2014.

[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-14 (work in progress), February 2014.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

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

[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.

[RFC5853] Hautakorpi, J., Camarillo, G., Penfield, R., Hawrylyshen, A., and M. Bhatia, "Requirements from Session Initiation Protocol (SIP) Session Border Control (SBC) Deployments", RFC 5853, April 2010.

[RFC6342] Koodli, R., "Mobile Networks Considerations for IPv6 Deployment", RFC 6342, August 2011.

[RFC6819] Lodderstedt, T., McGloin, M., and P. Hunt, "OAuth 2.0 Threat Model and Security Considerations", RFC 6819, January 2013.



Authors' Addresses

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

Email: [dwing@cisco.com](mailto:dwing@cisco.com)

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

Email: [tiredy@cisco.com](mailto:tiredy@cisco.com)

Prashanth Patil  
Cisco Systems, Inc.  
Bangalore  
India

Email: [praspati@cisco.com](mailto:praspati@cisco.com)

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

Email: [repenno@cisco.com](mailto:repenno@cisco.com)