

PCP Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 22, 2014

G. Chen
China Mobile
T. Reddy
P. Patil
Cisco
M. Boucadair
France Telecom
September 18, 2013

PCP Server Discovery in Mobile Networks with SIPTO
draft-chen-pcp-sipto-serv-discovery-00

Abstract

This document proposes an extension to DHCPv4 Relay information so that a PCP client learns the relevant PCP server deployed in the context of traffic offload.

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 March 22, 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. DHCPv4 Relay Agent | 3 |
| 3.1. Format | 3 |
| 3.2. Relay Agent behavior | 4 |
| 3.3. DHCPv4 Server behavior | 4 |
| 4. Security Considerations | 4 |
| 5. IANA Considerations | 5 |
| 6. Acknowledgements | 5 |
| 7. Change History | 5 |
| 8. References | 5 |
| 8.1. Normative References | 5 |
| 8.2. Informative References | 5 |
| Authors' Addresses | 6 |

1. Introduction

Given the exponential growth in the mobile data traffic, Mobile Operators are investigating solutions to offload some of the IP traffic flows at the nearest access edge that has an Internet interconnection point. This approach results in efficient usage of the mobile packet core and helps lower the transport cost. Since Release 10, 3GPP starts supporting of Selected IP Traffic Offload (SIPTO) function defined in [TS23.060], [TS23.401].

The SIPTO function described in [TS23.060] allows an operator to offload certain types of traffic at a network node close to the UE's point of attachment to the access network. Traffic Offload Function(TOF) has been defined to make such decisions and enforces NAT for those traffic. The traffic would go through the Mobile Packet Core only if the flow identification doesn't match TOF filters. SIPTO architecture is also explained in [I-D.chen-pcp-mobile-deployment].

[I-D.ietf-pcp-dhcp] specifies DHCP (IPv4 and IPv6) options to communicate Port Control Protocol (PCP) Server addresses to hosts. However, PCP Client on the mobile node will not know whether a flow will traverse the Mobile Packet Core or will get offloaded at the TOF and hence will not know which PCP server to send its requests to. Even if the mobile node learns its PCP server using DHCP, it will only learn about the PCP server in the Mobile Packet Core since the source of information is the DHCP server in the Mobile Packet Core. The mobile node may never learn the presence of the PCP server at

TOF. This requires TOF to act as a PCP Proxy for the PCP server in the Mobile Packet Core and as a PCP server for the offloaded traffic at the TOF. However, this alone does not solve this problem since the mobile node needs to be informed of the PCP proxy on the TOF.

This document proposes an extension to DHCPv4 Relay Information Option to achieve this objective. This will also ensure that the PCP client will only learn the PCP server address of the TOF.

Note:

- o The SIPTO problem can be addressed for IPv6 either by using NPTv6 [RFC6296] or associating the mobile device with multiple IPv6 prefixes (one prefix to offload the traffic and other one provided by the Mobile Packet Core for IP Mobility, access to Application Servers in Mobile Packet Core etc). New DHCPv6 Relay Agent PCP Server option will only be required if NPTv6 is used to offload the traffic. However if multiple IPv6 prefixes are assigned to the mobile device then it can use the mechanism explained in [I-D.ietf-pcp-server-selection] to contact multiple PCP servers.
- o The proposed extension to DHCPv4 Relay Information Option in this document is also useful to solve problems in other deployments like PMIPv6 [RFC5213] where mobile access gateway can selectively offload some of the IPv4 traffic flows in the access network instead of tunneling back to the local mobility anchor in the home network [RFC6909].

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. DHCPv4 Relay Agent

When DHCPv4 Relay Agent [RFC3046] is co-located with the TOF, the proposal is for the relay agent to influence the DHCPv4 Server to opt for the PCP server address proposed by the Relay Agent over the one configured on the DHCPv4 Server. The DHCPv4 Relay Agent will insert a new suboption under relay agent information option indicating the IP address of the appropriate PCP server/proxy. For this to happen, the UE MUST ensure that it includes OPTION_PCP_SERVER in the Parameter Request List Option in the DHCPv4 Discover/Request message.

3.1. Format

To realize the mechanism described above, the document proposes a new PCP Server suboption for the DHCPv4 relay agent information option that carries the IP address of PCP Server. If a PCP server is associated with more than one IP address, all those IP addresses can be listed as part of this option. If there is more than one PCP server, there will be multiple instances of this option each corresponding to a PCP server.

| Code | Length | PCP IP address | | | | | | | | |
|------|--------|----------------|----|----|----|----|----|----|----|-----|
| TBA | n | a1 | a2 | a3 | a4 | a1 | a2 | a3 | a4 | ... |

Code: TBA

Length: Includes the length of the "PCP Server IP address" field in octets; The maximum length is 255 octets. The length should be multiple of 4.

PCP Server IP address: The IP address of the PCP Server to be used by the PCP Client when issuing PCP messages.

3.2. Relay Agent behavior

A DHCPv4 relay agent MAY be configured to include a PCP Server suboption in relayed DHCPv4 messages. If the source IP address in the DHCPv4 request matches the TOF filter configuration then the PCP Server IP address SHOULD be inserted into the PCP Server suboption. The PCP Server IP address is determined through mechanisms outside the scope of this document.

3.3. DHCPv4 Server behavior

The proposed suboption provides additional information to the DHCP server. Upon receiving a DHCPv4 Discover/Request containing the suboption, the DHCPv4 server, if configured to support this suboption, MUST populate the DHCPv4 Offer/Ack with the suggested PCP server IP address overriding any other PCP server IP address configuration that it may already have. There is no special additional processing for this suboption.

4. Security Considerations

The security considerations in [RFC6887], [I-D.ietf-pcp-proxy] and section 5 of [RFC3046] also apply to this use.

5. IANA Considerations

Authors of this document request IANA to assign a suboption number for the PCP Server Suboption from the DHCP Relay Agent Information Option [RFC3046] suboption number space.

6. Acknowledgements

TODO

7. Change History

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

8. References

8.1. Normative References

- [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.
- [I-D.ietf-pcp-proxy]
Boucadair, M., Penno, R., and D. Wing, "Port Control Protocol (PCP) Proxy Function", draft-ietf-pcp-proxy-04 (work in progress), July 2013.
- [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-01 (work in progress), May 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3046] Patrick, M., "DHCP Relay Agent Information Option", RFC 3046, January 2001.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

8.2. Informative References

- [I-D.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.

[RFC5213] Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., and B. Patil, "Proxy Mobile IPv6", RFC 5213, August 2008.

[RFC6296] Wasserman, M. and F. Baker, "IPv6-to-IPv6 Network Prefix Translation", RFC 6296, June 2011.

[RFC6909] Gundavelli, S., Zhou, X., Korhonen, J., Feige, G., and R. Koodli, "IPv4 Traffic Offload Selector Option for Proxy Mobile IPv6", RFC 6909, April 2013.

[TS23.060] 3GPP, ., "General Packet Radio Service (GPRS); Service description; Stage 2", June 2012.", September 2012.

[TS23.401] 3GPP, ., "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access (Release 11)", 3GPP TS 23.401, V11.2.0 (2012-06).", September 2012.

Authors' Addresses

Gang Chen
China Mobile
No.32 Xuanwumen West Street
Xicheng District
Beijing 100053
China

Email: phdgang@gmail.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.
Bangalore
India

Email: praspati@cisco.com

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: mohamed.boucadair@orange.com

PCP working group
Internet-Draft
Intended status: Standards Track
Expires: April 8, 2016

S. Kiesel
University of Stuttgart
R. Penno
Cisco Systems, Inc.
S. Cheshire
Apple
October 6, 2015

Port Control Protocol (PCP) Anycast Addresses
draft-ietf-pcp-anycast-08

Abstract

The Port Control Protocol (PCP) Anycast Addresses enable PCP clients to transmit signaling messages to their closest PCP-aware on-path NAT, Firewall, or other middlebox, without having to learn the IP address of that middlebox via some external channel. This document establishes one well-known IPv4 address and one well-known IPv6 address to be used as PCP Anycast Addresses.

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 8, 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 |
| 2. PCP Server Discovery based on well-known IP Address | 4 |
| 2.1. PCP Discovery Client behavior | 4 |
| 2.2. PCP Discovery Server behavior | 4 |
| 3. Deployment Considerations | 5 |
| 4. IANA Considerations | 6 |
| 4.1. Registration of IPv4 Special Purpose Address | 6 |
| 4.2. Registration of IPv6 Special Purpose Address | 6 |
| 5. Security Considerations | 7 |
| 5.1. Information Leakage through Anycast | 7 |
| 5.2. Hijacking of PCP Messages sent to Anycast Addresses | 7 |
| 6. Acknowledgments | 9 |
| 7. References | 10 |
| 7.1. Normative References | 10 |
| 7.2. Informative References | 10 |
| Authors' Addresses | 11 |

1. Introduction

The Port Control Protocol (PCP) [RFC6887] 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), and IPv6 and IPv4 firewall devices. Furthermore, it provides a mechanism to reduce application keep alive traffic [I-D.ietf-pcp-optimize-keepalives]. The PCP base protocol document [RFC6887] specifies the message formats used, but the address to which a client sends its request is either assumed to be the default router (which is appropriate in a typical single-link residential network) or has to be configured otherwise via some external mechanism, such as a configuration file or a DHCP option [RFC7291].

This document follows a different approach: it establishes two well-known anycast addresses for the PCP Server, one IPv4 address and one IPv6 address. PCP clients usually send PCP requests to these well-known addresses if no other PCP server addresses are known or after communication attempts to such other addresses have failed. The anycast addresses are allocated from pools of special-purpose IP addresses (see Section 4), in accordance with Section 3.4 of [RFC4085]. Yet, a means to disable or override these well-known addresses (e. g., a configuration file option) should be available in implementations.

Using an anycast address is particularly useful in larger network topologies. For example, if the PCP-enabled NAT/firewall function is not located on the client's default gateway, but further upstream in a Carrier-grade NAT (CGN), sending PCP requests to the default gateway's IP address will not have the desired effect. When using a configuration file or the DHCP option to learn the PCP server's IP address, this file or the DHCP server configuration must reflect the network topology, and the router and CGN configuration. This may be cumbersome to achieve and maintain. If there is more than one upstream CGN and traffic is routed using a dynamic routing protocol such as OSPF, this approach may not be feasible at all, as it cannot provide timely information on which CGN to interact with. In contrast, when using the PCP anycast address, the PCP request will travel through the network like any other packet, without any special support from DNS, DHCP, other routers, or anything else, until it reaches the PCP-capable device, which receives it, handles it, and sends back a reply. A further advantage of using an anycast address instead of a DHCP option is, that the anycast address can be hard-coded into the application. There is no need for an application programming interface for passing the PCP server's address from the operating system's DHCP client to the application. For further discussion of deployment considerations see Section 3.

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

2.1. PCP Discovery Client behavior

PCP clients can add the PCP anycast addresses, which are defined in Sections 4.1 and 4.2, after the default router list (for IPv4 and IPv6) to the list of PCP server(s) (see Section 8.1, step 2. of [RFC6887]). This list is processed as specified in [RFC7488].

Note: If, in some specific scenario, it was desirable to use only the anycast address (and not the default router), this could be achieved by putting the anycast address into the configuration file, or DHCP option, etc.

2.2. PCP Discovery Server behavior

PCP Servers can be configured to listen on the anycast addresses for incoming PCP requests. When a PCP server receives a PCP requests destined for an anycast address it supports, it sends the corresponding PCP replies using that same anycast address as the source address (see Page 6 of [RFC1546] for further discussion).

3. Deployment Considerations

For general recommendations regarding operation of anycast services see [RFC4786]. Architectural considerations of IP anycast are discussed in [RFC7094].

In some deployment scenarios, using PCP anycasting may have certain limitations, which can be overcome by using additional mechanisms or by using other PCP server discovery methods instead, such as DHCP [RFC7291] or a configuration file.

One important example is a network topology, in which a network is connected to one or more upstream network(s) via several parallel firewalls, each individually controlled by its own PCP server. Even if all of these PCP servers are configured for anycasting, only one will receive the messages sent by a given client, depending on the state of the routing tables.

As long as routing is always symmetric, i.e., all upstream and downstream packets from/to that client are routed through this very same firewall, communication will be possible as expected. If there is a routing change, a PCP client using PCP anycasting might start interacting with a different PCP server. From the PCP client's point of view this would be the same as a PCP server reboot and the client could detect it by examining the Epoch field during the next PCP response or ANNOUNCE message. The client would re-establish the firewall rules and packet flows could resume.

If, however, routing is asymmetric, upstream packets from a client traverse a different firewall than the downstream packets to that client. Establishing policy rules in only one of these two firewalls by means of PCP anycasting will not have the desired result of allowing bi-directional connectivity. One solution approach to overcome this problem is an implementation-specific mechanism to synchronize state between all firewalls at the border of a network, i.e., a PEER message sent to any of these PCP servers would establish rules in all firewalls. Another approach would be to use a different discovery mechanism (e.g., DHCP or a configuration file) that allows a PCP client to acquire a list of all PCP servers controlling the parallel firewalls and configure each of them individually.

4. IANA Considerations

4.1. Registration of IPv4 Special Purpose Address

IANA is requested to assign a single IPv4 address from the 192.0.0.0/24 prefix and register it in the IANA IPv4 Special-Purpose Address Registry [RFC6890].

| Attribute | Value |
|----------------------|---|
| Address Block | 192.0.0.???/32 (??? = TBD by IANA) |
| Name | Port Control Protocol Anycast |
| RFC | This document, if approved (TBD) |
| Allocation Date | Date of approval of this document (TBD) |
| Termination Date | N/A |
| Source | True |
| Destination | True |
| Forwardable | True |
| Global | True |
| Reserved-by-Protocol | False |

4.2. Registration of IPv6 Special Purpose Address

IANA is requested to assign a single IPv6 address from the 2001:0000::/23 prefix and register it in the IANA IPv6 Special-Purpose Address Registry [RFC6890].

| Attribute | Value |
|----------------------|---|
| Address Block | 2001:0????????/128 (??? = TBD by IANA) |
| Name | Port Control Protocol Anycast |
| RFC | This document, if approved (TBD) |
| Allocation Date | Date of approval of this document (TBD) |
| Termination Date | N/A |
| Source | True |
| Destination | True |
| Forwardable | True |
| Global | True |
| Reserved-by-Protocol | False |

5. Security Considerations

In addition to the security considerations in [RFC6887], [RFC4786], and [RFC7094], two further security issues are considered here.

5.1. Information Leakage through Anycast

In a network without any border gateway, NAT or firewall that is aware of the PCP anycast address, outgoing PCP requests could leak out onto the external Internet, possibly revealing information about internal devices.

Using an IANA-assigned well-known PCP anycast address enables border gateways to block such outgoing packets. In the default-free zone, routers should be configured to drop such packets. Such configuration can occur naturally via BGP messages advertising that no route exists to said address.

Sensitive clients that do not wish to leak information about their presence can set an IP TTL on their PCP requests that limits how far they can travel towards the public Internet. However, methods for choosing an appropriate TTL value, e.g., based on the assumed radius of the trusted network domain, is beyond the scope of this document.

Before sending PCP requests with possibly privacy-sensitive parameters (e.g., IP addresses and port numbers) to the PCP anycast addresses, PCP clients can send an ANNOUNCE request (without parameters; see Section 14.1 of [RFC6887]), in order to probe whether a PCP server consumes and processes PCP requests sent to that anycast address.

5.2. Hijacking of PCP Messages sent to Anycast Addresses

The anycast addresses are treated by normal host operating systems just as normal unicast addresses, i.e., packets destined for an anycast address are sent to the default router for processing and forwarding. Hijacking such packets in the first network segment would effectively require the attacker to impersonate the default router, e.g., by means of ARP spoofing in an Ethernet network. Once an anycast message is forwarded closer to the core network, routing will likely become subject to dynamic routing protocols such as OSPF or BGP. Anycast messages could be hijacked by announcing counterfeited messages in these routing protocols. When analyzing the risk and possible consequences of such attacks in a given network scenario, the probable impacts on PCP signaling need to be put into proportion with probable impacts on other protocols such as the actual application protocols.

In addition to following best current practices in first hop security and routing protocol security, PCP authentication [RFC7652] may be useful in some scenarios. However, the effort needed for a proper setup of this authentication mechanism (e.g., installing the right shared secrets or cryptographic keys on all involved systems) may thwart the goal of fully automatic configuration by using PCP anycast. Therefore, this approach may be less suitable for scenarios with high trust between the operator of the PCP-controlled middlebox and all users (e.g., a residential gateway used only by family members) or if there is anyway rather limited trust that the middlebox will behave correctly (e.g., the Wifi in an airport lounge). In contrast, this scheme may be highly useful in scenarios with many users and a trusted network operator, such as a large corporate network or a university campus network, which uses several parallel NATs or firewalls to connect to the Internet. Therefore, a thorough analysis of the benefits and costs of using PCP authentication in a given network scenario is recommended.

6. Acknowledgments

The authors would like to thank the members of the PCP working group for contributions and feedback, in particular Mohamed Boucadair, Charles Eckel, Simon Perreault, Tirumaleswar Reddy, Markus Stenberg, Dave Thaler, and Dan Wing.

7. References

7.1. Normative References

- [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.
- [RFC7488] Boucadair, M., Penno, R., Wing, D., Patil, P., and T. Reddy, "Port Control Protocol (PCP) Server Selection", RFC 7488, March 2015.

7.2. Informative References

- [I-D.ietf-pcp-optimize-keepalives] Reddy, T., Patil, P., Isomaki, M., and D. Wing, "Optimizing NAT and Firewall Keepalives Using Port Control Protocol (PCP)", draft-ietf-pcp-optimize-keepalives-06 (work in progress), May 2015.
- [RFC1546] Partridge, C., Mendez, T., and W. Milliken, "Host Anycasting Service", RFC 1546, November 1993.
- [RFC4085] Plonka, D., "Embedding Globally-Routable Internet Addresses Considered Harmful", BCP 105, RFC 4085, DOI 10.17487/RFC4085, June 2005, <<http://www.rfc-editor.org/info/rfc4085>>.
- [RFC4786] Abley, J. and K. Lindqvist, "Operation of Anycast Services", BCP 126, RFC 4786, December 2006.
- [RFC7094] McPherson, D., Oran, D., Thaler, D., and E. Osterweil, "Architectural Considerations of IP Anycast", RFC 7094, DOI 10.17487/RFC7094, January 2014, <<http://www.rfc-editor.org/info/rfc7094>>.
- [RFC7291] Boucadair, M., Penno, R., and D. Wing, "DHCP Options for the Port Control Protocol (PCP)", RFC 7291, July 2014.
- [RFC7652] Cullen, M., Hartman, S., Zhang, D., and T. Reddy, "Port Control Protocol (PCP) Authentication Mechanism", RFC 7652, DOI 10.17487/RFC7652, September 2015, <<http://www.rfc-editor.org/info/rfc7652>>.

Authors' Addresses

Sebastian Kiesel
University of Stuttgart Information Center
Networks and Communication Systems Department
Allmandring 30
Stuttgart 70550
Germany

Email: ietf-pcp@skiesel.de

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

Email: repenno@cisco.com

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

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

Network Working Group
Internet-Draft
Updates: 6887 (if approved)
Intended status: Standards Track
Expires: January 21, 2016

M. Wasserman
S. Hartman
Painless Security
D. Zhang
Huawei
T. Reddy
Cisco
July 20, 2015

Port Control Protocol (PCP) Authentication Mechanism
draft-ietf-pcp-authentication-14

Abstract

An IPv4 or IPv6 host can use the Port Control Protocol (PCP) to flexibly manage the IP address and port mapping information on Network Address Translators (NATs) or firewalls to facilitate communication with remote hosts. However, the un-controlled generation or deletion of IP address mappings on such network devices may cause security risks and should be avoided. In some cases the client may need to prove that it is authorized to modify, create or delete PCP mappings. This document describes an in-band authentication mechanism for PCP that can be used in those cases. The Extensible Authentication Protocol (EAP) is used to perform authentication between PCP devices.

This document updates RFC6887.

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 21, 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 |
| 2. Terminology | 4 |
| 3. Protocol Details | 5 |
| 3.1. Session Initiation | 5 |
| 3.1.1. Authentication triggered by the client | 6 |
| 3.1.2. Authentication triggered by the server | 7 |
| 3.1.3. Authentication using EAP | 7 |
| 3.2. Recovery from lost PA session | 9 |
| 3.3. Session Termination | 10 |
| 3.4. Session Re-Authentication | 11 |
| 4. PA Security Association | 12 |
| 5. Packet Format | 13 |
| 5.1. Packet Format of PCP Auth Messages | 13 |
| 5.2. Opcode-specific information of AUTHENTICATION Opcode | 15 |
| 5.3. NONCE Option | 16 |
| 5.4. AUTHENTICATION_TAG Option | 16 |
| 5.5. PA_AUTHENTICATION_TAG option | 18 |
| 5.6. EAP_PAYLOAD Option | 19 |
| 5.7. PRF Option | 19 |
| 5.8. MAC_ALGORITHM Option | 20 |
| 5.9. SESSION_LIFETIME Option | 20 |
| 5.10. RECEIVED_PAK Option | 21 |
| 5.11. ID_INDICATOR Option | 21 |
| 6. Processing Rules | 22 |
| 6.1. Authentication Data Generation | 22 |
| 6.2. Authentication Data Validation | 23 |
| 6.3. Retransmission Policies for PA Messages | 24 |
| 6.4. Sequence Numbers for PCP Auth Messages | 24 |
| 6.5. Sequence Numbers for Common PCP Messages | 25 |
| 6.6. MTU Considerations | 26 |
| 7. IANA Considerations | 27 |

| | | |
|-------|--|----|
| 7.1. | NONCE | 28 |
| 7.2. | AUTHENTICATION_TAG | 28 |
| 7.3. | PA_AUTHENTICATION_TAG | 28 |
| 7.4. | EAP_PAYLOAD | 29 |
| 7.5. | PRF | 29 |
| 7.6. | MAC_ALGORITHM | 29 |
| 7.7. | SESSION_LIFETIME | 30 |
| 7.8. | RECEIVED_PAK | 30 |
| 7.9. | ID_INDICATOR | 30 |
| 8. | Security Considerations | 31 |
| 9. | Acknowledgements | 31 |
| 10. | Change Log | 32 |
| 10.1. | Changes from wasserman-pcp-authentication-02 to ietf- pcp-authentication-00 | 32 |
| 10.2. | Changes from wasserman-pcp-authentication-01 to -02 | 32 |
| 10.3. | Changes from ietf-pcp-authentication-00 to -01 | 32 |
| 10.4. | Changes from ietf-pcp-authentication-01 to -02 | 32 |
| 10.5. | Changes from ietf-pcp-authentication-02 to -03 | 33 |
| 10.6. | Changes from ietf-pcp-authentication-03 to -04 | 33 |
| 10.7. | Changes from ietf-pcp-authentication-04 to -05 | 33 |
| 10.8. | Changes from ietf-pcp-authentication-05 to -06 | 33 |
| 11. | References | 34 |
| 11.1. | Normative References | 34 |
| 11.2. | Informative References | 35 |
| | Authors' Addresses | 35 |

1. Introduction

Using the Port Control Protocol (PCP) [RFC6887], an application can flexibly manage the IP address mapping information on its network address translators (NATs) and firewalls, and control their policies in processing incoming and outgoing IP packets. Because NATs and firewalls both play important roles in network security architectures, there are many situations in which authentication and access control are required to prevent un-authorized users from accessing such devices. This document defines a PCP security extension that enables PCP servers to authenticate their clients with Extensible Authentication Protocol (EAP). The EAP messages are encapsulated within PCP messages during transportation.

The following issues are considered in the design of this extension:

- o Loss of EAP messages during transportation
- o Reordered delivery of EAP messages
- o Generation of transport keys

- o Integrity protection and data origin authentication for PCP messages
- o Algorithm agility

The mechanism described in this document meets the security requirements to address the Advanced Threat Model described in the base PCP specification [RFC6887]. This mechanism can be used to secure PCP in the following situations:

- o On security infrastructure equipment, such as corporate firewalls, that do not create implicit mappings for specific traffic.
- o On equipment (such as CGNs or service provider firewalls) that serve multiple administrative domains and do not have a mechanism to securely partition traffic from those domains.
- o For any implementation that wants to be more permissive in authorizing applications to create mappings for successful inbound communications destined to machines located behind a NAT or a firewall.

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 RFC 2119 [RFC2119].

Most of the terms used in this document are introduced in [RFC6887].

PCP Client: A PCP software instance that is responsible for issuing PCP requests to a PCP server. In this document, a PCP client is also a EAP peer [RFC3748], and it is the responsibility of a PCP client to provide the credentials when authentication is required.

PCP Server: A PCP software instance that resides on the PCP-Controlled Device that receives PCP requests from the PCP client and creates appropriate state in response to that request. In this document, a PCP server is integrated with an EAP authenticator [RFC3748]. Therefore, when necessary, a PCP server can verify the credentials provided by a PCP client and make an access control decision based on the authentication result.

PCP-Authentication (PA) Session: A series of PCP message exchanges transferred between a PCP client and a PCP server. The PCP messages involved within a session includes the PA messages used to perform EAP authentication, key distribution and session management, and the common PCP messages secured with the keys distributed during

authentication. Each PA session is assigned a distinctive Session ID.

Session Partner: A PCP implementation involved within a PA session. Each PA session has two session partners (a PCP server and a PCP client).

PCP device: A PCP client or a PCP server.

Session Lifetime: The lifetime associated with a PA session, which decides the lifetime of the current authorization given to the PCP client.

PCP Security Association (PCP SA): A PCP security association is formed between a PCP client and a PCP server by sharing cryptographic keying material and associated context. The formed duplex security association is used to protect the bidirectional PCP signaling traffic between the PCP client and PCP server.

Master Session Key (MSK): A key derived by the partners of a PA session, using an EAP key generating method (e.g., the one defined in [RFC5448]).

PCP-Authentication (PA) message: A PCP message containing an AUTHENTICATION Opcode. Particularly, a PA message sent from a PCP server to a PCP client is referred to as a PA-Server message, while a PA message sent from a PCP client to a PCP server is referred to as a PA-Client message. Therefore, a PA-Server message is actually a PCP response message specified in [RFC6887], and a PA-Client message is a PCP request message. This document specifies an option, the PA_AUTHENTICATION_TAG Option defined in Section 5.5 for PCP authentication, to provide integrity protection and message origin authentication for PA messages.

Common PCP message: A PCP message which does not contain an AUTHENTICATION Opcode. This document specifies an AUTHENTICATION_TAG Option to provide integrity protection and message origin authentication for the common PCP messages.

3. Protocol Details

3.1. Session Initiation

At the beginning of a PA session, a PCP client and a PCP server need to exchange a series of PA messages in order to perform an EAP authentication process. Each PA message MUST contain an AUTHENTICATION Opcode and may optionally contain a set of Options for various purposes (e.g., transporting authentication messages and

session management). The opcode-specific information in a AUTHENTICATION Opcode consists of two fields : Session ID and Sequence Number. The Session ID field is used to identify the PA session to which the message belongs. The sequence number field is used to detect whether reordering or duplication occurred during message delivery.

3.1.1. Authentication triggered by the client

When a PCP client intends to proactively initiate a PA session with a PCP server, it sends a PA-Initiation message (a PA-Client message with the result code "INITIATION") to the PCP server. Section 5.1 updates the PCP request message format with result codes for the PCP Authentication mechanism. In the opcode-specific information of the message, the Session ID and Sequence Number fields are set as 0. The PA-Client message MUST also contain a NONCE option defined in Section 5.3 which consists of a random nonce.

After receiving the PA-Initiation, if the PCP server agrees to initiate a PA session with the PCP client, it will reply with a PA-Server message which contains an EAP Request and the result code field of this PA-Server message is set to AUTHENTICATION_REQUEST. In addition, the server MUST assign a unique session identifier to distinctly identify this session, and fill the identifier into the Session ID field in the opcode-specific information of the PA-Server message. The Sequence Number field of the message is set as 0. The PA-Server message MUST contain a NONCE option so as to send the nonce value back. The nonce will then be used by the PCP client to check the freshness of this message. Subsequent PCP messages within this PA session MUST contain this session identifier.

| PCP client | PCP server |
|--|---------------|
| <pre>-- PA-Initiation-----> (Seq=0, rc=INITIATION, Session ID=0)</pre> | <pre> </pre> |
| <pre><-- PA-Server ----- (Seq=0, Session ID=X, EAP request, rc=AUTHENTICATION_REQUEST)</pre> | <pre> </pre> |
| <pre>-- PA-Client -----> (Seq=1, Session ID=X, EAP response, rc=AUTHENTICATION_REPLY)</pre> | <pre> </pre> |
| <pre><-- PA-Server ----- (Seq=1, Session ID=X, EAP request, rc=AUTHENTICATION_REQUEST)</pre> | <pre> </pre> |

3.1.2. Authentication triggered by the server

In the scenario where a PCP server receives a common PCP request message from a PCP client which needs to be authenticated, the PCP server rejects the request with a `AUTHENTICATION_REQUIRED` error code and can reply with a unsolicited PA-Server message to initiate a PA session. The result code field of this PA-Server message is set to `AUTHENTICATION_REQUEST`. In addition, the PCP server **MUST** assign a Session ID for the session and transfer it within the PA-Server message. The Sequence Number field in the PA-Server message is set as 0. If the PCP client retries the common request before EAP authentication is successful then it will receive `AUTHENTICATION_REQUIRED` error code from the PCP server. In the PA messages exchanged afterwards in this session, the Session ID will be used in order to help session partners distinguish the messages within this session from those not within. When the PCP client receives this initial PA-Server message from the PCP server, it can reply with a PA-Client message or silently discard the request message according to its local policies. In the PA-Client message, a `NONCE` option which consists of a random nonce **MAY** be appended. If so, in the next PA-Server message, the PCP server **MUST** forward the nonce back within a `NONCE` option.

| PCP client | PCP server |
|---|---------------|
| -- Common PCP request-----> | |
| <- Common PCP response----- rc=AUTHENTICATION_REQUIRED) | |
| <-- PA-Server ----- (Seq=0, Session ID=X, EAP request) rc=AUTHENTICATION_REQUEST) | |
| -- PA-Client -----> (Seq=0, Session ID=X, EAP response) rc=AUTHENTICATION_REPLY) | |
| <-- PA-Server ----- (Seq=1, Session ID=X, EAP request, rc=AUTHENTICATION_REQUEST) | |

3.1.3. Authentication using EAP

In a PA session, an EAP request message is transported within a PA-Server message and an EAP response message is transported within a PA-Client message. EAP relies on the underlying protocol to provide

reliable transmission; any reordered delivery or loss of packets occurring during transportation must be detected and addressed. Therefore, after sending out a PA-Server message, the PCP server will not send a new PA-Server message in the same PA session until it receives a PA-Client message with a proper sequence number from the PCP client, and vice versa. If a PCP client receives a PA message containing an EAP request and cannot generate an EAP response immediately due to certain reasons (e.g., waiting for human input to construct a EAP message or due to EAP message fragmentation waiting for the additional PA messages in order to construct a complete EAP message), the PCP device MUST reply with a PA-Acknowledgement message (PA message with a RECEIVED_PAK Option) to indicate that the message has been received. This approach not only can avoid unnecessary retransmission of the PA message but also can guarantee the reliable message delivery in conditions where a PCP device needs to receive multiple PA messages carrying the fragmented EAP request before generating an EAP response. The number of EAP messages exchanged between the PCP client and PCP server depends on the EAP method used for authentication.

In this approach, PCP client and a PCP server MUST perform a key-generating EAP method in authentication. Particularly, a PCP authentication implementation MUST support EAP-TTLS [RFC5281] and SHOULD support TEAP [RFC7170]. Therefore, after a successful authentication procedure, a Master Session Key (MSK) will be generated. If the PCP client and the PCP server want to generate a transport key using the MSK, they need to agree upon a Pseudo-Random Function (PRF) for the transport key derivation and a MAC algorithm to provide data origin authentication for subsequent PCP messages. In order to do this, the PCP server needs to append a set of PRF Options and MAC_ALGORITHM Options to the initial PA-Server message. Each PRF Option contains a PRF that the PCP server supports, and each MAC_ALGORITHM Option contains a MAC (Message Authentication Code) algorithm that the PCP server supports. Moreover, in the first PA-Server message, the server MAY also attach an ID_INDICATOR Option defined in Section 5.11 to direct the client to choose correct credentials. After receiving the options, the PCP client MUST select the PRF and the MAC algorithm which it would like to use, and then adds the associated PRF and MAC Algorithm Options to the next PA-Client message.

After the EAP authentication, the PCP server sends out a PA-Server message to indicate the EAP authentication and PCP authorization results. If the EAP authentication succeeds, the result code of the PA-Server message is AUTHENTICATION_SUCCEEDED. In this case, before sending out the PA-Server message, the PCP server MUST update the PCP SA with the MSK and transport key, and use the derived transport key to generate a digest for the message. The digest is transported

within an PA_AUTHENTICATION_TAG Option for PCP Auth. A more detailed description of generating the authentication data can be found in Section 6.1. In addition, the PA-Server message MUST also contain a SESSION_LIFETIME Option defined in Section 5.9 which indicates the lifetime of the PA session (i.e., the lifetime of the MSK). After receiving the PA-Server message, the PCP client then needs to generate a PA-Client message as response. If the PCP client also authenticates the PCP server, the result code of the PA-Client message is AUTHENTICATION_SUCCEEDED. In addition, the PCP client needs to update the PCP SA with the MSK and transport key, and uses the derived transport key to secure the message. From then on, all the PCP messages within the session are secured with the transport key and the MAC algorithm specified in the PCP SA. The first secure PA-client message from the client MUST include the set of PRF and MAC_ALGORITHM options received from the PCP server. The PCP server determines if the set of algorithms conveyed by the client matches the set it had initially sent, to detect an algorithm downgrade attack. If the server detects a downgrade attack then it MUST send a PA-Server message with result code DOWNGRADE_ATTACK_DETECTED and terminate the session. If the PCP client sends common PCP request within the PA session without AUTHENTICATION_TAG option then the PCP server rejects the request by returning AUTHENTICATION_REQUIRED error code.

If a PCP client/server cannot authenticate its session partner, the device sends out a PA message with the result code, AUTHENTICATION_FAILED. If the EAP authentication succeeds but authorization fails, the device making the decision sends out a PA message with the result code, AUTHORIZATION_FAILED. In these two cases, after the PA message is sent out, the PA session MUST be terminated immediately. It is possible for independent PCP clients on the host to create multiple PA sessions with the PCP server.

3.2. Recovery from lost PA session

If a PCP server resets or loses the PCP SA due to reboot, power failure, or any reason then it sends unsolicited ANNOUNCE response as explained in section 14.1.3 of [RFC6887] to the PCP client. Upon receiving the ANNOUNCE response with an anomalous Epoch time, PCP client deduces that the server may have lost state. The ANNOUNCE is either bogus (an attack), legitimate, or not seen by the client. These three cases are described below:

- o PCP client sends integrity-protected unicast ANNOUNCE request to the PCP server to check if the PCP server has indeed lost the state or an attacker has sent the ANNOUNCE response.

- * If integrity-protected success response is received from the PCP server then the PCP client determines that the PCP server has not lost the PA session, and the unsolicited ANNOUNCE response was sent by an attacker.
- * If the PCP server responds to the ANNOUNCE request with UNKNOWN_SESSION_ID error code then the PCP client MUST initiate full EAP authentication with the PCP server as explained in Section 3.1.1. After EAP authentication is successful PCP client updates the PCP SA and issues new common PCP requests to recreate any lost mapping state.
- o In a scenario where the PCP server has lost the PCP SA but did not inform the PCP client, if the PCP client sends PCP request integrity-protected then the PCP server rejects the request with UNKNOWN_SESSION_ID error code. The PCP client then initiates full EAP authentication with the PCP server as explained in Section 3.1.1 and updates the PCP SA after successful authentication.

If the PCP client resets or loses the PCP SA due to reboot, power failure, or any reason and sends common PCP request then the PCP server rejects the request with AUTHENTICATION_REQUIRED error code. The PCP client MUST authenticate with the PCP server and after EAP authentication is successful retry the common PCP request with AUTHENTICATION_TAG option. The PCP server MUST update the PCP SA after successful EAP authentication.

3.3. Session Termination

A PA session can be explicitly terminated by either session partner. A PCP Server may explicitly request termination of the session by sending an unsolicited termination-indicating PA response (a PA response with a result code "SESSION-TERMINATED"). Upon receiving a termination-indicating message, the PCP client MUST respond with a termination-indicating PA message, and MUST then remove the associated PCP SA. To accommodate packet loss, the PCP server MAY transmit the termination-indicating PA response up to ten times (with an appropriate Epoch Time value in each to reflect the passage of time between transmissions) provided that the interval between the first two notifications is at least 250 ms, and the interval between subsequent notification at least doubles.

A PCP client may explicitly request termination of the session by sending a termination-indicating PA request (a PA request with a result code "SESSION-TERMINATED"). After receiving a termination-indicating message from the PCP client, a PCP server MUST respond with a termination-indicating PA response and remove the PCP SA

immediately. When the PCP client receives the termination-indicating PA response, it MUST remove the associated PCP SA immediately.

3.4. Session Re-Authentication

A session partner may select to perform EAP re-authentication if it would like to update the PCP SA without initiating a new PA session. For example a re-authentication procedure could be triggered for the following reasons:

- o The session lifetime needs to be extended.
- o The sequence number is going to reach the maximum value. Specifically, when the sequence number reaches $2^{32} - 2^{16}$, the session partner MUST trigger re-authentication.

When the PCP server would like to initiate a re-authentication, it sends the PCP client a PA-Server message. The result code of the message is set to "RE-AUTHENTICATION", which indicates the message is for a re-authentication process. If the PCP client would like to start the re-authentication, it will send a PA-Client message to the PCP server, with the result code of the PA-Client message set to "RE-AUTHENTICATION". Then, the session partners exchange PA messages to transfer EAP messages for the re-authentication. During the re-authentication procedure, the session partners protect the integrity of PA messages with the key and MAC algorithm specified in the current PCP SA; the sequence numbers associated with the message will continue to keep increasing according to Section 6.3. The result code for PA-Server message carrying EAP request will be set to AUTHENTICATION_REQUIRED and PA-Client message carrying EAP response will be set to AUTHENTICATION_REPLY.

If the EAP re-authentication succeeds, the result code of the last PA-Server message is "AUTHENTICATION_SUCCEEDED". In this case, before sending out the PA-Server message, the PCP server MUST update the SA and use the new key to generate a digest for the PA-Server message and subsequent PCP messages. In addition, the PA-Server message MUST be appended with a SESSION_LIFETIME Option which indicates the new lifetime of the PA session. PA and PCP message sequence numbers must also be reset to zero.

If the EAP authentication fails, the result code of the last PA-Server message is "AUTHENTICATION_FAILED". If the EAP authentication succeeds but authorization fails, the result code of the last PA-Server message is "AUTHORIZATION_FAILED". In the latter two cases, the PA session MUST be terminated immediately after the last PA message exchange. If for some unknown reason re-authentication is

not performed and session lifetime has expired then PA session MUST be terminated immediately.

During re-authentication, the session partners can also exchange common PCP messages in parallel. The common PCP messages MUST be protected with the current SA until the new SA has been generated. The sequence of EAP messages exchanged for re-authentication will not change, regardless of the PCP device triggering re-authentication. If the PCP server receives re-authentication request from the PCP client after it had signaled re-authentication request then it should discard its request and respond to the re-authentication request from the PCP client.

4. PA Security Association

At the beginning of a new PA session, each PCP device must create and initialize state information for a new PA Security Association (PCP SA) to maintain its state information for the duration of the PA session. The parameters of a PCP SA are listed as follows:

- o IP address and UDP port number of the PCP client
- o IP address and UDP port number of the PCP server
- o Session Identifier
- o Sequence number for the next outgoing PA message
- o Sequence number for the next incoming PA message
- o Sequence number for the next outgoing common PCP message
- o Sequence number for the next incoming common PCP message
- o Last outgoing message payload
- o Retransmission interval
- o The master session key (MSK) generated by the EAP method.
- o The MAC algorithm that the transport key should use to generate digests for PCP messages.
- o The pseudo random function negotiated in the initial PA-Server and PA-Client message exchange for the transport key derivation
- o The transport key derived from the MSK to provide integrity protection and data origin authentication for the messages in the

PA session. The lifetime of the transport key SHOULD be identical to the lifetime of the session.

- o The nonce selected by the PCP client at the initiation of the session.
- o The Key ID associated with Transport key.

Particularly, the transport key is computed in the following way: Transport key = prf(MSK, "IETF PCP" || Session ID || Nonce || key ID), where:

- o prf: The pseudo-random function assigned in the Pseudo-random function parameter.
- o MSK: The master session key generated by the EAP method.
- o "IETF PCP": The ASCII code representation of the non-NULL terminated string (excluding the double quotes around it).
- o '||' : is the concatenation operator.
- o Session ID: The ID of the session which the MSK is derived from.
- o Nonce: The nonce selected by the client and transported in the Initial PA-Client message.
- o Key ID: The ID assigned for the transport key.

5. Packet Format

5.1. Packet Format of PCP Auth Messages

The format of the PA-Server message is identical to the response message format specified in Section 7.2 of [RFC6887]. The result code for PA-Sever message carrying EAP request MUST be set to AUTHENTICATION_REQUEST.

As illustrated in Figure 1, this document updates the reserved field in the request header specified in Section 7.1 of [RFC6887] to carry Opcode-specific data.

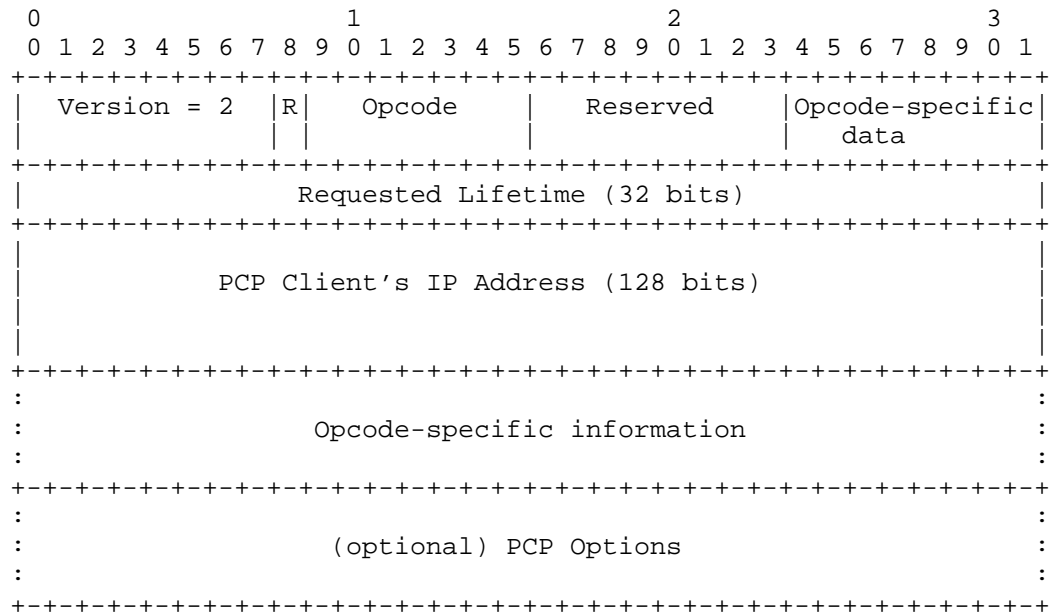


Figure 1. Request Packet Format

As illustrated in Figure 2, the PA-Client messages use the request header specified in Figure 1. The Opcode-specific data is used to transfer the result codes (e.g., "INITIATION", "AUTHENTICATION_FAILED"). Other fields in Figure 2 are described in Section 7.1 of [RFC6887]. The result code for PA-Client message carrying EAP response MUST be set to AUTHENTICATION_REPLY.

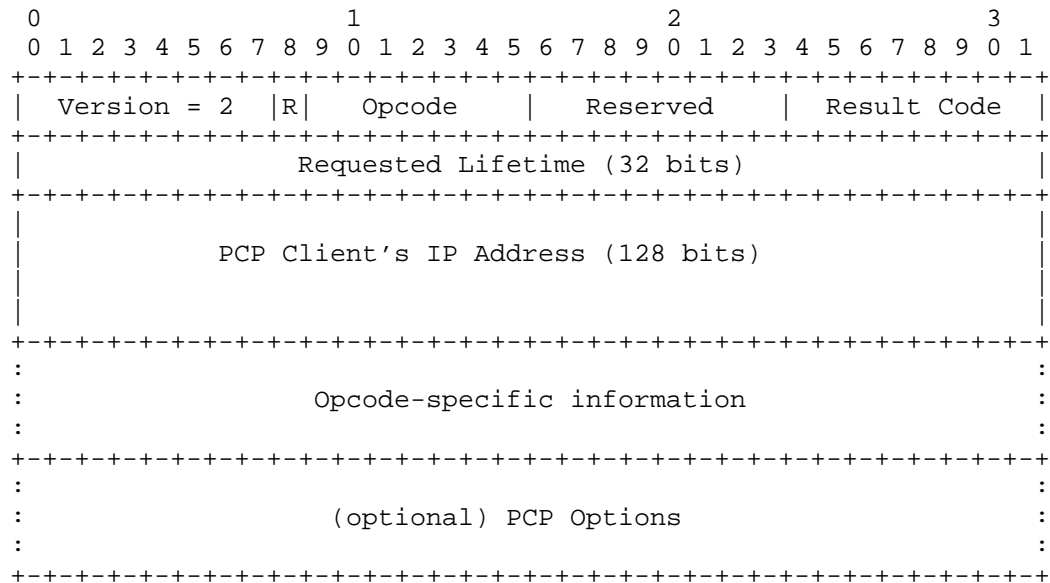
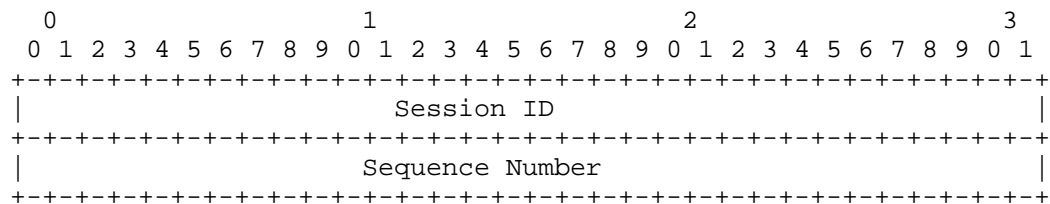


Figure 2. PA-Client message Format

The Requested Lifetime field of PA-Client message and Lifetime field of PA-Server message are both set to 0 on transmission and ignored on reception.

5.2. Opcode-specific information of AUTHENTICATION Opcode

The following diagram shows the format of the Opcode-specific information for the AUTHENTICATION Opcode.

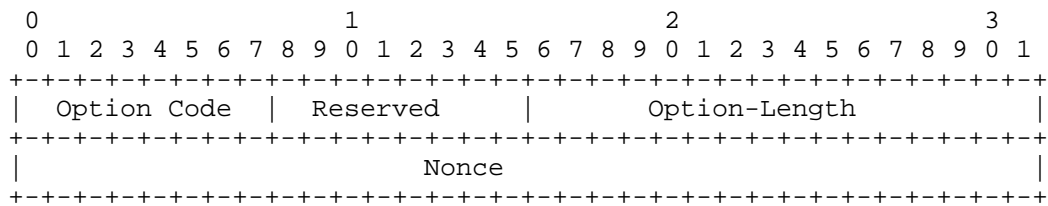


Session ID: This field contains a 32-bit PA session identifier.

Sequence Number: This field contains a 32-bit sequence number. A sequence number needs to be incremented on every new (non-retransmission) outgoing PA message in order to provide an ordering guarantee for PA messages.

5.3. NONCE Option

Because the session identifier of a PA session is determined by the PCP server, a PCP client does not know the session identifier which will be used when it sends out a PA-Initiation message. In order to prevent an attacker from interrupting the authentication process by sending off-line generated PA-Server messages, the PCP client needs to generate a random number as a nonce in the PA-Initiation message. The PCP server will append the nonce within the initial PA-Server message. If the PA-Server message does not carry the correct nonce, the message MUST be discarded silently.



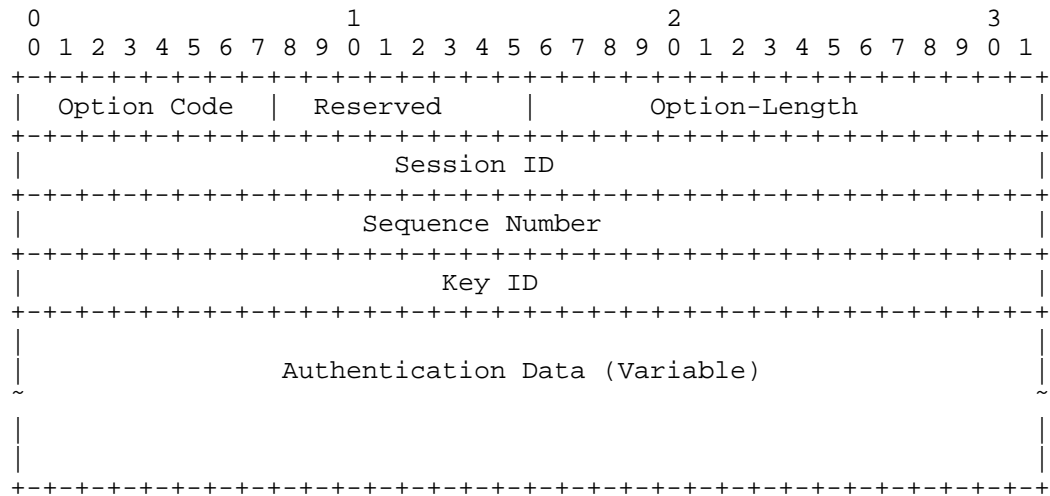
Option Code: TBA-130.

Reserved: 8 bits. MUST be set to 0 on transmission and MUST be ignored on reception.

Option-Length: 4 octets.

Nonce: A random 32 bit number which is transported within a PA-Initiation message and the corresponding reply message from the PCP server.

5.4. AUTHENTICATION_TAG Option



Because there is no authentication Opcode in common PCP messages, the authentication tag for common PCP messages needs to carry the Session ID and Sequence Number.

Option Code: TBA-131.

Reserved: 8 bits. MUST be set to 0 on transmission and MUST be ignored on reception.

Option-Length: The length of the AUTHENTICATION_TAG Option for Common PCP message (in octets), including the 12 octet fixed header and the variable length of the authentication data.

Session ID: A 32-bit field used to identify the session to which the message belongs and identify the secret key used to create the message digest appended to the PCP message.

Sequence Number: A 32-bit sequence number. In this solution, a sequence number needs to be incremented on every new (non-retransmission) outgoing common PCP message in order to provide ordering guarantee for common PCP messages.

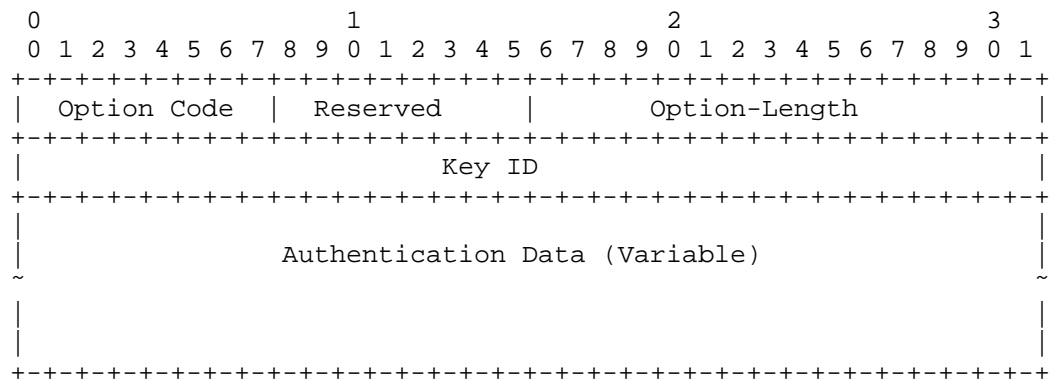
Key ID: The ID associated with the transport key used to generate authentication data. This field is filled with zero if the MSK is directly used to secure the message.

Authentication Data: A variable-length field that carries the Message Authentication Code for the Common PCP message. The generation of the digest varies according to the algorithms

specified in different PCP SAs. This field MUST end on a 32-bit boundary, padded with 0's when necessary.

5.5. PA_AUTHENTICATION_TAG option

This option is used to provide message authentication for PA messages. Compared with the AUTHENTICATION_TAG Option for Common PCP Messages, the Session ID field and the Sequence Number field are removed because such information is provided in the Opcode-specific information of AUTHENTICATION Opcode.



Option Code: TBA-132.

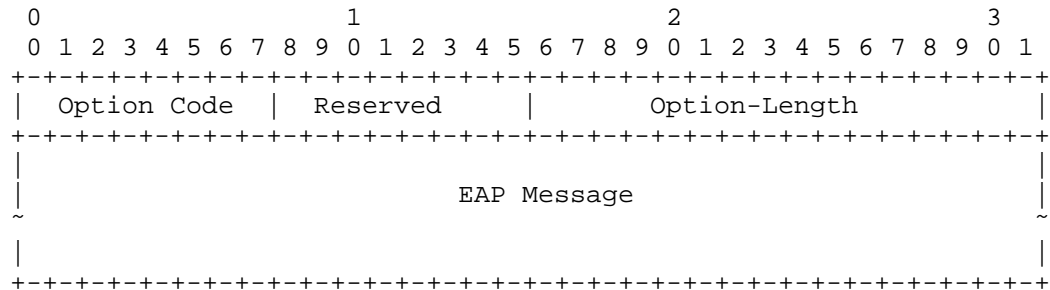
Reserved: 8 bits. MUST be set to 0 on transmission and MUST be ignored on reception.

Option-Length: The length of the PA_AUTHENTICATION Option for PCP Auth message (in octet), including the 4 octet fixed header and the variable length of the authentication data.

Key ID: The ID associated with the transport key used to generate authentication data. This field is filled with zero if the MSK is directly used to secure the message.

Authentication Data: A variable-length field that carries the Message Authentication Code for the PCP Auth message. The generation of the digest varies according to the algorithms specified in different PCP SAs. This field MUST end on a 32-bit boundary, padded with null characters when necessary.

5.6. EAP_PAYLOAD Option



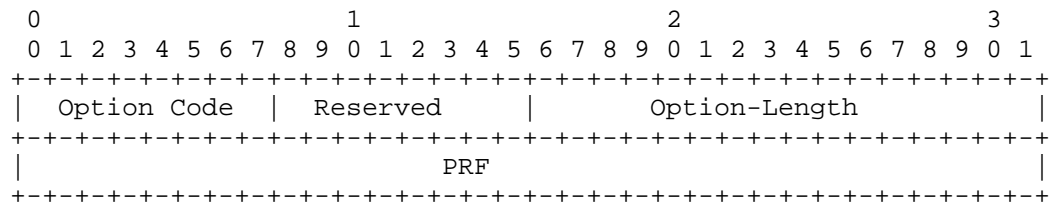
Option Code: TBA-133.

Reserved: 8 bits. MUST be set to 0 on transmission and MUST be ignored on reception.

Option-Length: Variable

EAP Message: The EAP message transferred. Note this field MUST end on a 32-bit boundary, padded with 0's when necessary.

5.7. PRF Option



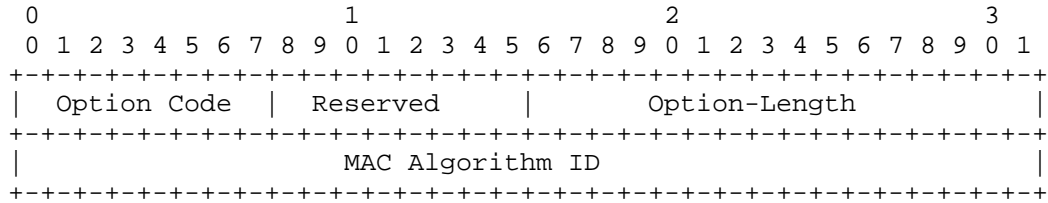
Option Code: TBA-134.

Reserved: 8 bits. MUST be set to 0 on transmission and MUST be ignored on reception.

Option-Length: 4 octets.

PRF: The Pseudo-Random Function which the sender supports to generate an MSK. This field contains an IKEv2 Transform ID of Transform Type 2 [RFC7296][RFC4868]. A PCP implementation MUST support PRF HMAC_SHA2_256 (5).

5.8. MAC_ALGORITHM Option



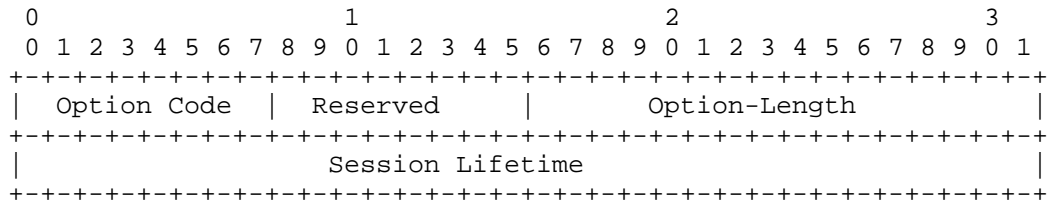
Option Code: TBA-135.

Reserved: 8 bits. MUST be set to 0 on transmission and MUST be ignored on reception.

Option-Length: 4 octets.

MAC Algorithm ID: Indicate the MAC algorithm which the sender supports to generate authentication data. The MAC Algorithm ID field contains an IKEv2 Transform ID of Transform Type 3 [RFC7296][RFC4868]. A PCP implementation MUST support AUTH_HMAC_SHA2_256_128 (12).

5.9. SESSION_LIFETIME Option



Option Code: TBA-136.

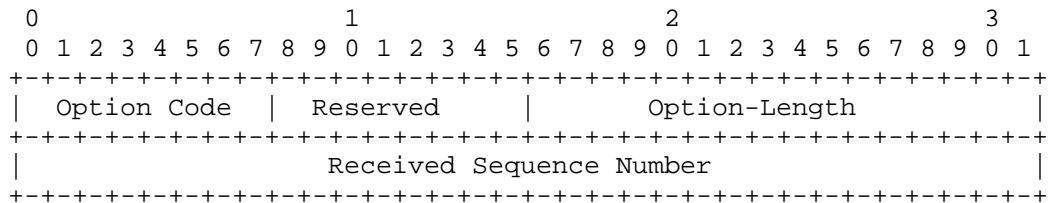
Reserved: 8 bits. MUST be set to 0 on transmission and MUST be ignored on reception.

Option-Length: 4 octets.

Session Lifetime: An unsigned 32-bit integer, in seconds, ranging from 0 to $2^{32}-1$ seconds. The lifetime of the PA Session, which is decided by the authorization result.

5.10. RECEIVED_PAK Option

This option is used in a PA-Acknowledgement message to indicate that a PA message with the contained sequence number has been received.



Option Code: TBA-137.

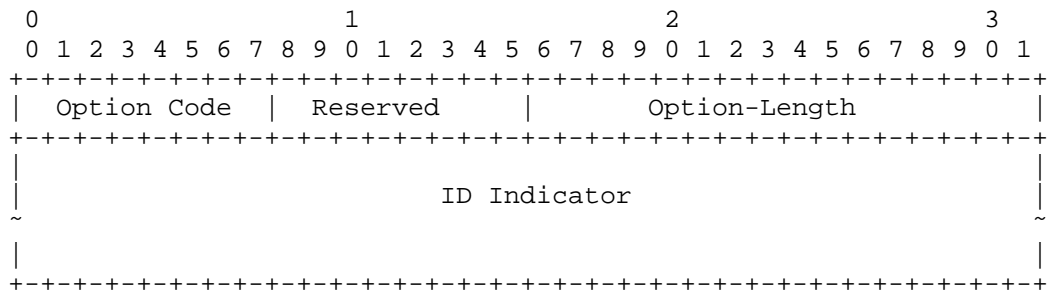
Reserved: 8 bits. MUST be set to 0 on transmission and MUST be ignored on reception.

Option-Length: 4 octets.

Received Sequence Number: The sequence number of the last received PA message.

5.11. ID_INDICATOR Option

The ID_INDICATOR option is used by the PCP client to determine which credentials to provide to the PCP server.



Option Code: TBA-138.

Reserved: 8 bits. MUST be set to 0 on transmission and MUST be ignored on reception.

Option-Length: Variable.

ID Indicator: The identity of the authority that issued the EAP credentials to be used to authenticate the client. The field MUST

NOT be null terminated and its length is indicated by the Option-Length field. In particular when a client receives a ID_INDICATOR option, it MUST NOT rely on the presence of a NUL character in the wire format data to identify the end of the ID Indicator field.

The field MUST end on a 32-bit boundary, padded with 0's when necessary. The ID indicator field is UTF-8 encoded [RFC3629] Unicode string conforming to the "UsernameCaseMapped" profile of the PRECIS IdentifierClass [I-D.ietf-precis-saslprepbis]. The PCP client validates that the ID indicator field conforms to the "UsernameCaseMapped" profile of the PRECIS IdentifierClass. The PCP client enforces the rules specified in section 3.2.2 of [I-D.ietf-precis-saslprepbis] to map the ID indicator field. The PCP client compares the resulting string with the ID indicators stored locally on the PCP client to pick the credentials for authentication. The two indicator strings are to be considered equivalent by the client if and only if they are an exact octet-for-octet match.

6. Processing Rules

6.1. Authentication Data Generation

After successful EAP authentication process, every subsequent PCP message within the PA session MUST carry an authentication tag which contains the digest of the PCP message for data origin authentication and integrity protection.

- o Before generating a digest for a PA message, a device needs to first locate the PCP SA according to the session identifier and then get the transport key. Then the device appends an PA_AUTHENTICATION_TAG Option for PCP Auth at the end of the PCP Auth message. The length of the Authentication Data field is decided by the MAC algorithm adopted in the session. The device then fills the Key ID field with the key ID of the transport key, and sets the Authentication Data field to 0. After this, the device generates a digest for the entire PCP message (including the PCP header and PA_AUTHENTICATION_TAG Option) using the transport key and the associated MAC algorithm, and inserts the generated digest into the Authentication Data field.
- o Similar to generating a digest for a PA message, before generating a digest for a common PCP message, a device needs to first locate the PCP SA according to the session identifier and then get the transport key. Then the device appends the AUTHENTICATION_TAG Option at the end of common PCP message. The length of the Authentication Data field is decided by the MAC algorithm adopted in the session. The device then uses the corresponding values

derived from the SA to fill the Session ID field, the Sequence Number field and the Key ID field, and sets the Authentication Data field to 0. After this, the device generates a digest for the entire PCP message (including the PCP header and AUTHENTICATION_TAG Option) using the transport key and the associated MAC algorithm, and inputs the generated digest into the Authentication Data field.

6.2. Authentication Data Validation

When a device receives a common PCP message with an AUTHENTICATION_TAG Option for Common PCP Messages, the device needs to use the Session ID transported in the option to locate the proper SA, and then find the associated transport key (using the key ID in the option) and the MAC algorithm. If no proper SA or transport key is found or the sequence number is invalid (see Section 6.5), the PCP device stops processing the PCP message and discards the message silently. After storing the value of the Authentication field of the AUTHENTICATION_TAG Option, the device fills the Authentication field with zeros. Then, the device generates a digest for the message (including the PCP header and Authentication Tag Option) with the transport key and the MAC algorithm. If the value of the newly generated digest is identical to the stored one, the device can ensure that the message has not been tampered with, and the validation succeeds. Otherwise, the PCP device stops processing the PCP message and silently discards the message.

Similarly, when a device receives a PA message with an PA_AUTHENTICATION_TAG Option for PCP Authentication, the device needs to use the Session ID transported in the Opcode to locate the proper SA, and then find the associated transport key (using the key ID in the option) and the MAC algorithm. If no proper SA or transport key is found or the sequence number is invalid (see Section 6.4), the PCP device stops processing the PCP message and discards the message. After storing the value of the Authentication field of the PA_AUTHENTICATION_TAG Option, the device fills the Authentication field with zeros. Then, the device generates a digest for the message (including the PCP header and PA_AUTHENTICATION_TAG Option) with the transport key and the MAC algorithm. If the value of the newly generated digest is identical to the stored one, the device can ensure that the message has not been tampered with, and the validation succeeds. Otherwise, the PCP device stops processing the PCP message and silently discards the message.

6.3. Retransmission Policies for PA Messages

Because EAP relies on the underlying protocols to provide reliable transmission, after sending a PA message, a PCP client/server MUST NOT send out any subsequent messages until receiving a PA message with a proper sequence number from the peer. If no such a message is received the PCP device will re-send the last message according to retransmission policies. This work reuses the retransmission policies specified in the base PCP protocol (Section 8.1.1 of [RFC6887]). In the base PCP protocol, such retransmission policies are only applied by PCP clients. However, in this work, such retransmission policies are also applied by the PCP servers. If Maximum retransmission duration seconds have elapsed and no expected response is received, the device will terminate the session and discard the current SA.

As illustrated in Section 3.1.3, in order to avoid unnecessary retransmission, the device receiving a PA message MUST send a PA-Acknowledgement message to the sender of the PA message when it cannot send a PA response immediately. The PA-Acknowledgement message is used to indicate the receipt of the PA message. When the sender receives the PA-Acknowledgement message, it will stop the retransmission.

Note that the last PA messages transported within the phases of session initiation, session re-authentication, and session termination do not have to follow the above policies since the devices sending out those messages do not expect any further PA messages.

When a device receives a re-transmitted last incoming PA message from its session partner, it MUST try to answer it by sending the last outgoing PA message again. However, if the duplicate message has the same sequence number but is not bit-wise identical to the original message then the device MUST discard it. In order to achieve this function, the device may need to maintain the last incoming and the associated outgoing messages. In this case, if no outgoing PA message has been generated for the received duplicate PA message yet, the device needs to send a PA-Acknowledgement message. The rate of replying to duplicate PA messages MUST be limited to provide robustness against denial of service (DoS) attacks. The details of rate limiting are outside the scope of this specification.

6.4. Sequence Numbers for PCP Auth Messages

PCP uses UDP to transport signaling messages. As an un-reliable transport protocol, UDP does not guarantee ordered packet delivery and does not provide any protection from packet loss. In order to

ensure the EAP messages are exchanged in a reliable way, every PCP message exchanged during EAP authentication must carry a monotonically increasing sequence number. During a PA session, a PCP device needs to maintain two sequence numbers for PA messages, one for incoming PA messages and one for outgoing PA messages. When generating an outgoing PA message, the device adds the associated outgoing sequence number to the message and increments the sequence number maintained in the SA by 1. When receiving a PA message from its session partner, the device will not accept it if the sequence number carried in the message does not match the incoming sequence number the device maintains. After confirming that the received message is valid, the device increments the incoming sequence number maintained in the SA by 1.

The above rules are not applicable to PA-Acknowledgement messages (i.e., PA messages containing a RECEIVED_PAK Option). A PA-Acknowledgement message does not transport any EAP message and only indicates that a PA message is received. Therefore, reliable transmission of PA-Acknowledgement messages is not required. For instance, after sending out a PA-Acknowledgement message, a device generates an EAP response. In this case, the device need not have to confirm whether the PA-Acknowledgement message has been received by its session partner or not. Therefore, when receiving or sending out a PA-Acknowledgement message, the device MUST NOT increase the corresponding sequence number stored in the SA. Otherwise, loss of a PA-Acknowledgement message will cause a mismatch in sequence numbers.

Another exception is the message retransmission scenario. As discussed in Section 6.3, when a PCP device does not receive any response from its session partner it needs to retransmit the last outgoing PA message following the retransmission procedure specified in section 8.1.1 of [RFC6887]. The original message and duplicate messages MUST be bit-wise identical. When the device receives such a duplicate PA message from its session partner, it MUST send the last outgoing PA message again. In such cases, the maintained incoming and outgoing sequence numbers will not be affected by the message retransmission.

6.5. Sequence Numbers for Common PCP Messages

When transporting common PCP messages within a PA session, a PCP device needs to maintain a sequence number for outgoing common PCP messages and a sequence number for incoming common PCP messages. When generating a new outgoing PCP message, the PCP device updates the Sequence Number field in the AUTHENTICATION_TAG option with the outgoing sequence number maintained in the SA and increments the outgoing sequence number by 1.

When receiving a PCP message from its session partner, the PCP device will not accept it if the sequence number carried in the message is smaller than the incoming sequence number the device maintains. This approach can protect the PCP device from replay attacks. After confirming that the received message is valid, the PCP device will update the incoming sequence number maintained in the PCP SA with the sequence number of the incoming message.

Note that the sequence number in the incoming message may not exactly match the incoming sequence number maintained locally. As discussed in the base PCP specification [RFC6887], if a PCP client is no longer interested in the PCP transaction and has not yet received a PCP response from the server then it will stop retransmitting the PCP request. After that, the PCP client might generate new PCP requests for other purposes using the current SA. In this case, the sequence number in the new request will be larger than the sequence number in the old request and so will be larger than the incoming sequence number maintained in the PCP server.

Note that in the base PCP specification [RFC6887], a PCP client needs to select a nonce in each MAP or PEER request, and the nonce is sent back in the response. However, it is possible for a client to use the same nonce in multiple MAP or PEER requests, and this may cause a potential risk of replay attacks. This attack is addressed by using the sequence number in the PCP response.

6.6. MTU Considerations

EAP methods are responsible for MTU handling, so no special facilities are required in PCP to deal with MTU issues. Particularly, EAP lower layers indicate to EAP methods and AAA servers the MTU of the lower layer. EAP methods such as EAP-TLS [RFC5216], TEAP [RFC7170], and others that are likely to exceed reasonable MTUs provide support for fragmentation and reassembly. Others, such as EAP-GPSK [RFC5433] assume they will never send packets larger than the MTU and use small EAP packets.

If an EAP message is too long to be transported within a single PA message, it will be divided into multiple sections and sent within different PA messages. Note that the receiver may not be able to know what to do in the next step until it has received all the sections and reconstructed the complete EAP message. In this case, in order to guarantee reliable message transmission, after receiving a PA message, the receiver replies with a PA-Acknowledgement message to notify the sender to send the next PA message.

7. IANA Considerations

The following PCP Opcode is to be allocated in the mandatory-to-process range from the standards action range (the registry for PCP Opcodes is maintained in <http://www.iana.org/assignments/pcp-parameters>):

TBA AUTHENTICATION Opcode.

The following PCP result codes are to be allocated in the mandatory-to-process range from the standards action range (the registry for PCP result codes is maintained in <http://www.iana.org/assignments/pcp-parameters>):

TBA INITIATION: The client indication to the server for authentication.

TBA AUTHENTICATION_REQUIRED: The error response is signaled to the client that EAP authentication is required.

TBA AUTHENTICATION_FAILED: This error response is signaled to the client if EAP authentication had failed.

TBA AUTHENTICATION_SUCCEEDED: This success response is signaled to the client if EAP authentication had succeeded.

TBA AUTHORIZATION_FAILED: This error response is signaled to the client if the EAP authentication had succeeded but authorization failed.

TBA SESSION_TERMINATED: This PCP result code indicates to the partner that the PA session must be terminated.

TBA UNKNOWN_SESSION_ID: The error response is signaled from the PCP server that there is no known PA session associated with the Session ID signaled in the PA request or common PCP request from the PCP client.

TBA DOWNGRADE_ATTACK_DETECTED: This error response is signaled to the client if the server detects downgrade attack.

TBA AUTHENTICATION_REQUEST: The server indication to the client that EAP request is signaled in the PA message.

TBA AUTHENTICATION_REPLY: The client indication to the server that EAP response is signaled in the PA message.

The following PCP Option Codes are to be allocated in the mandatory-to-process range from the standards action range (the registry for PCP Options is maintained in <http://www.iana.org/assignments/pcp-parameters>):

7.1. NONCE

Option Name: NONCE

option-code: TBA-130 in the mandatory-to-process range (IANA).

Purpose: See Section 5.3.

Valid for Opcodes: Authentication Opcode.

option-len: Option Length is 4 octets.

May appear in: request and response.

Maximum occurrences: 1.

7.2. AUTHENTICATION_TAG

Option Name: AUTHENTICATION_TAG

option-code: TBA-131 in the mandatory-to-process range (IANA).

Purpose: See Section 5.4.

Valid for Opcodes: MAP, PEER and ANNOUNCE Opcodes.

option-len: Variable length.

May appear in: request and response.

Maximum occurrences: 1.

7.3. PA_AUTHENTICATION_TAG

Option Name: PA_AUTHENTICATION_TAG

option-code: TBA-132 in the mandatory-to-process range (IANA).

Purpose: See Section 5.5.

Valid for Opcodes: Authentication Opcode.

option-len: Variable length.

May appear in: request and response.

Maximum occurrences: 1.

7.4. EAP_PAYLOAD

Option Name: EAP_PAYLOAD.

option-code: TBA-133 in the mandatory-to-process range (IANA).

Purpose: See Section 5.6.

Valid for Opcodes: Authentication Opcode.

option-len: Variable length.

May appear in: request and response.

Maximum occurrences: 1.

7.5. PRF

Option Name: PRF.

option-code: TBA-134 in the mandatory-to-process range (IANA).

Purpose: See Section 5.7.

Valid for Opcodes: Authentication Opcode.

option-len: Option Length is 4 octets.

May appear in: request and response.

Maximum occurrences: as many as fit within maximum PCP message size.

7.6. MAC_ALGORITHM

Option Name: MAC_ALGORITHM.

option-code: TBA-135 in the mandatory-to-process range (IANA).

Purpose: See Section 5.8.

Valid for Opcodes: Authentication Opcode.

option-len: Option Length is 4 octets.

May appear in: request and response.

Maximum occurrences: as many as fit within maximum PCP message size.

7.7. SESSION_LIFETIME

Option Name: SESSION_LIFETIME.

option-code: TBA-136 in the mandatory-to-process range (IANA).

Purpose: See Section 5.9.

Valid for Opcodes: Authentication Opcode.

option-len: Option Length is 4 octets.

May appear in: response.

Maximum occurrences: 1.

7.8. RECEIVED_PAK

Option Name: RECEIVED_PAK.

option-code: TBA-137 in the mandatory-to-process range (IANA).

Purpose: See Section 5.10.

Valid for Opcodes: Authentication Opcode.

option-len: Option Length is 4 octets.

May appear in: request and response.

Maximum occurrences: 1.

7.9. ID_INDICATOR

Option Name: ID_INDICATOR.

option-code: TBA-138 in the mandatory-to-process range (IANA).

Purpose: See Section 5.11.

Valid for Opcodes: Authentication Opcode.

option-len: Variable length.

May appear in: response.

Maximum occurrences: 1.

8. Security Considerations

In this work, after a successful EAP authentication process is performed between two PCP devices, an MSK will be exported. The MSK will be used to derive the transport keys to generate MAC digests for subsequent PCP message exchanges. However, before a transport key has been generated, the PA messages exchanged within a PA session have little cryptographic protection, and if there is no already established security channel between two session partners, these messages are subject to man-in-the-middle attacks and DOS attacks. For instance, the initial PA-Server and PA-Client message exchange is vulnerable to spoofing attacks as these messages are not authenticated and integrity protected. In addition, because the PRF and MAC algorithms are transported at this stage, an attacker may try to remove the PRF and MAC options containing strong algorithms from the initial PA-Server message and force the client choose the weakest algorithms. Therefore, the server needs to guarantee that all the PRF and MAC algorithms it provides support for are strong enough.

In order to prevent very basic DOS attacks, a PCP device SHOULD generate state information as little as possible in the initial PA-Server and PA-Client message exchanges. The choice of EAP method is also very important. The selected EAP method must be resilient to the attacks possible in an insecure network environment, provide user-identity confidentiality, protection against dictionary attacks, and support session-key establishment.

When a PCP proxy [I-D.ietf-pcp-proxy] is located between a PCP server and PCP clients, the proxy may perform authentication with the PCP server before it processes requests from the clients. In addition, re-authentication between the PCP proxy and PCP server will not interrupt the service that the proxy provides to the clients since the proxy is still allowed to send common PCP messages to the PCP server during that period.

9. Acknowledgements

Thanks to Dan Wing, Prashanth Patil, Dave Thaler, Peter Saint-Andre, Carlos Pignataro, Brian Haberman, Paul Kyzivat, Jouni Korhonen, Stephen Farrell and Terry Manderson for the valuable comments.

10. Change Log

[Note: This section should be removed by the RFC Editor upon publication]

10.1. Changes from wasserman-pcp-authentication-02 to ietf-pcp-authentication-00

- o Added discussion of in-band and out-of-band key management options, leaving choice open for later WG decision.
- o Removed support for fragmenting EAP messages, as that is handled by EAP methods.

10.2. Changes from wasserman-pcp-authentication-01 to -02

- o Add a nonce into the first two exchanged PCP-Auth message between the PCP client and PCP server. When a PCP client initiate the session, it can use the nonce to detect offline attacks.
- o Add the key ID field into the authentication tag option so that a MSK can generate multiple transport keys.
- o Specify that when a PCP device receives a PCP-Auth-Server or a PCP-Auth-Client message from its partner the PCP device needs to reply with a PCP-Auth-Acknowledge message to indicate that the message has been received.
- o Add the support of fragmenting EAP messages.

10.3. Changes from ietf-pcp-authentication-00 to -01

- o Editorial changes, added use cases to introduction.

10.4. Changes from ietf-pcp-authentication-01 to -02

- o Add the support of re-authentication initiated by PCP server.
- o Specify that when a PCP device receives a PCP-Auth-Server or a PCP-Auth-Client message from its partner the PCP device MAY reply with a PCP-Auth-Acknowledge message to indicate that the message has been received.
- o Discuss the format of the PCP-Auth-Acknowledge message.
- o Remove the redundant information from the Auth Opcode, and specify new result codes transported in PCP packet headers

- o

10.5. Changes from ietf-pcp-authentication-02 to -03

- o Change the name "PCP-Auth-Request" to "PCP-Auth-Server"
- o Change the name "PCP-Auth-Response" to "PCP-Auth-Client"
- o Specify two new sequence numbers for common PCP messages in the PCP SA, and describe how to use them
- o Specify a Authentication Tag Option for PCP Common Messages
- o Introduce the scenario where a EAP message has to be divided into multiple sections and transported in different PCP-Auth messages (for the reasons of MTU), and introduce how to use PCP-Auth-Acknowledge messages to ensure reliable packet delivery in this case.

10.6. Changes from ietf-pcp-authentication-03 to -04

- o Change the name "PCP-Auth" to "PA".
- o Refine the retransmission policies.
- o Add more discussion about the sequence number management .
- o Provide the discussion about how to instruct a PCP client to choose proper credential during authentication, and an ID Indicator Option is defined for that purpose.

10.7. Changes from ietf-pcp-authentication-04 to -05

- o Add contents in IANA considerations.
- o Add discussions in fragmentation.
- o Refine the PA messages retransmission policies.
- o Add IANA considerations.

10.8. Changes from ietf-pcp-authentication-05 to -06

- o Added mechanism to handle algorithm downgrade attack.
- o Updated Security Considerations section.
- o Updated ID Indicator Option.

11. References

11.1. Normative References

- [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-09 (work in progress), July 2015.
- [I-D.ietf-precis-saslprepbis]
Saint-Andre, P. and A. Melnikov, "Preparation, Enforcement, and Comparison of Internationalized Strings Representing Usernames and Passwords", draft-ietf-precis-saslprepbis-18 (work in progress), May 2015.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, DOI 10.17487/RFC3629, November 2003, <<http://www.rfc-editor.org/info/rfc3629>>.
- [RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowitz, Ed., "Extensible Authentication Protocol (EAP)", RFC 3748, DOI 10.17487/RFC3748, June 2004, <<http://www.rfc-editor.org/info/rfc3748>>.
- [RFC4868] Kelly, S. and S. Frankel, "Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec", RFC 4868, DOI 10.17487/RFC4868, May 2007, <<http://www.rfc-editor.org/info/rfc4868>>.
- [RFC5281] Funk, P. and S. Blake-Wilson, "Extensible Authentication Protocol Tunneled Transport Layer Security Authenticated Protocol Version 0 (EAP-TTLSv0)", RFC 5281, DOI 10.17487/RFC5281, August 2008, <<http://www.rfc-editor.org/info/rfc5281>>.
- [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>>.

- [RFC7170] Zhou, H., Cam-Winget, N., Salowey, J., and S. Hanna, "Tunnel Extensible Authentication Protocol (TEAP) Version 1", RFC 7170, DOI 10.17487/RFC7170, May 2014, <<http://www.rfc-editor.org/info/rfc7170>>.
- [RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <<http://www.rfc-editor.org/info/rfc7296>>.

11.2. Informative References

- [RFC5216] Simon, D., Aboba, B., and R. Hurst, "The EAP-TLS Authentication Protocol", RFC 5216, DOI 10.17487/RFC5216, March 2008, <<http://www.rfc-editor.org/info/rfc5216>>.
- [RFC5433] Clancy, T. and H. Tschofenig, "Extensible Authentication Protocol - Generalized Pre-Shared Key (EAP-GPSK) Method", RFC 5433, DOI 10.17487/RFC5433, February 2009, <<http://www.rfc-editor.org/info/rfc5433>>.
- [RFC5448] Arkko, J., Lehtovirta, V., and P. Eronen, "Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA')", RFC 5448, DOI 10.17487/RFC5448, May 2009, <<http://www.rfc-editor.org/info/rfc5448>>.

Authors' Addresses

Margaret Wasserman
Painless Security
356 Abbott Street
North Andover, MA 01845
USA

Phone: +1 781 405 7464
Email: mrw@painless-security.com
URI: <http://www.painless-security.com>

Sam Hartman
Painless Security
356 Abbott Street
North Andover, MA 01845
USA

Email: hartmans@painless-security.com
URI: <http://www.painless-security.com>

Dacheng Zhang
Huawei
Beijing
China

Email: zhang_dacheng@hotmail.com

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

Email: tiredddy@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.

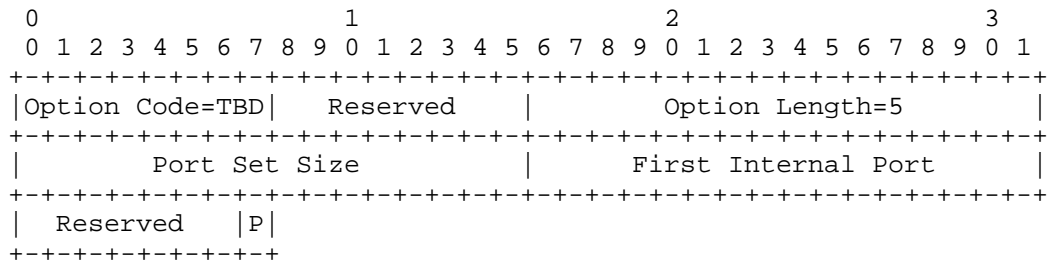


Figure 1: PORT_SET Option

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

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: 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

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

Email: 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.

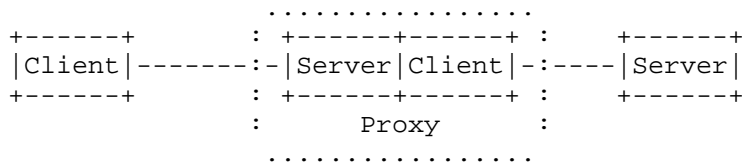


Figure 1: Reference Architecture

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.

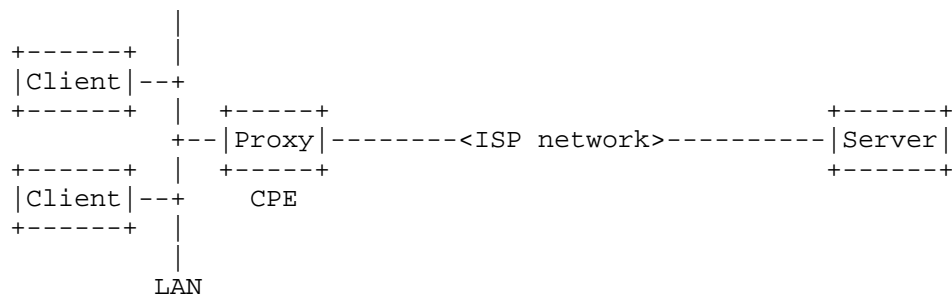


Figure 2: PCP Relay Use Case

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 Working Group
Internet-Draft
Updates: 6887 (if approved)
Intended status: Standards Track
Expires: November 23, 2013

M. Boucadair
France Telecom
R. Penno
D. Wing
P. Patil
T. Reddy
Cisco
May 22, 2013

PCP Server Selection
draft-ietf-pcp-server-selection-01

Abstract

This document specifies the behavior to be followed by the PCP client to contact its PCP server(s) when one or several PCP server names are configured. Multiple names may be configured to a PCP client in some deployment contexts such as multi-homing.

This document updates RFC6887.

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 November 23, 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. Terminology | 3 |
| 3. Name Resolution | 3 |
| 4. IP Address Selection | 3 |
| 4.1. Serial Queries | 4 |
| 5. Examples | 4 |
| 5.1. Example 1 | 4 |
| 5.2. Example 2 | 5 |
| 5.3. Example 3 | 5 |
| 6. Security Considerations | 6 |
| 7. IANA Considerations | 6 |
| 8. Acknowledgements | 6 |
| 9. References | 6 |
| 9.1. Normative References | 6 |
| 9.2. Informative References | 6 |
| Appendix A. Multi-homing | 7 |
| A.1. IPv6 Multi-homing | 7 |
| A.2. IPv4 Multi-homing | 8 |
| A.3. Multiple interfaces and Servers | 9 |
| Authors' Addresses | 9 |

1. Introduction

This document specifies the behavior to be followed by the PCP client [RFC6887] to contact its PCP server(s) [RFC6887] when receiving one or several PCP server names (e.g., using DHCP [I-D.ietf-pcp-dhcp]). This document is not specific to DHCP; any other mechanism can be used to configure PCP server names.

Multiple names may be configured to a PCP client in some deployment contexts such as multi-homing (see Appendix A). It is out of scope of this document to enumerate all deployment scenarios which require multiple names to be configured.

This document assumes appropriate name resolution means (e.g., Section 6.1.1 of [RFC1123]) are available on the host client.

2. Terminology

This document makes use of the following terms:

- o PCP server denotes a functional element which 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., Network Address Translation (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 Name is a string that can be passed to getaddrinfo (Section 6.1 of [RFC3493]), such as a DNS name, address literals, etc. A name may be a fully qualified domain name (e.g., "myservice.example.com."), IPv4 address in dotted-decimal form (e.g., 192.0.2.33) or textual representation of an IPv6 address (e.g., 2001:db8::1).

3. Name Resolution

Each configured name is passed to the name resolution library (e.g., Section 6.1.1 of [RFC1123] or [RFC6055]) to retrieve the corresponding IP address(es) (IPv4 or IPv6). Then, the PCP client follows the procedure specified in Section 4 to contact its PCP server(s).

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.

4. IP Address Selection

This section specifies the behavior to be followed by the PCP client to contact its PCP server(s) when receiving one or several PCP server names:

1. If only one PCP server name is configured: if a list of IP addresses is returned as a result of resolving the PCP server name, the PCP client follows the procedure specified in Section 4.1.

2. If several PCP server names are configured: each name is treated as a separate PCP server. Moreover, each name may be resolved into one IP address or a list of IP addresses. The PCP client contacts in parallel the first IP address of each name and follows the procedure specified in Section 4.1 for the list of IP addresses returned for each name. Section 5 provides some examples to illustrate this procedure.

This procedure does not require any knowledge of the capabilities of the PCP-controlled device(s). Instead, the PCP client assumes each configured name refer to a separate PCP server.

This procedure may result in a PCP client instantiating multiple mappings maintained by distinct PCP servers. The decision to use all these mappings or delete some of them is deployment-specific. Only the PCP client can decide whether all the mappings are needed or only a subset of them.

4.1. Serial Queries

The PCP client initializes its Maximum Retransmission Count (MRC) to 4.

The PCP client sends its PCP message to the PCP server following the retransmission procedure specified in Section 8.1.1 of [RFC6887]. If no response is received after MRC attempts, the PCP client tries with the next IP address in its list of PCP server addresses. If it has exhausted its list, the procedure is repeated every fifteen minutes until the PCP request is successfully answered. If, when sending PCP requests the PCP client receives an ICMP error (e.g., port unreachable, network unreachable) it SHOULD immediately try the next IP address in the list. Once the PCP client has successfully received a response from a PCP server address on that interface, it sends subsequent PCP requests to that same server address until that PCP server becomes non-responsive, which causes the PCP client to attempt to re-iterate the procedure starting with the first PCP server address on its list.

5. Examples

The following sub-sections provide three examples to illustrate the procedure.

For all these examples, let's suppose pcpservice-x, pcpservice-y and pcpservice-z are configured as PCP server names.

5.1. Example 1

Let's also suppose:

- * IPx1 and IPx2 are returned for pcpserver-x; IPx1 is not reachable.
- * IPy1 and IPy2 are returned for pcpserver-y; IPy1 is reachable
- * IPz1 and IPz2 are returned for pcpserver-z; IPz1 is reachable

The procedure to contact the PCP servers is as follows:

- * Send PCP requests to all servers: IPx1, IPy1 and IPz1
- * Responses are received from IPy1 and IPz1 but not from IPx1
 - The request is re-sent to IPx1
 - If no response is received after four attempts, the request is sent to IPx2

5.2. Example 2

Now, if the following conditions are made:

- * IPx1 and IPx2 are returned for pcpserver-x; IPx1 is not reachable.
- * IPy1 and IPy2 are returned for pcpserver-y; IPy1 is reachable
- * IPz1 and IPz2 are returned for pcpserver-z; IPz1 is not reachable

The procedure to contact the PCP servers lead to the following:

- * Send PCP requests to all servers: IPx1, IPy1 and IPz1
- * A response is received from IPy1 but not from IPx1 and IPz1
 - the requests are re-sent to IPx1 and IPz1
 - If no response is received after four attempts, the request is then sent to IPx2 and IPz2

5.3. Example 3

Let's suppose now that:

- * IPx1 and IPx2 are returned for pcpserver-x; IPx1 is not reachable.
- * IPy1 and IPy2 are returned for pcpserver-y; IPy1 is not reachable
- * IPz1 and IPz2 are returned for pcpserver-z; IPz1 is not reachable

The procedure to contact the PCP servers is as follows:

- * Send PCP requests to all servers: IPx1, IPy1 and IPz1
- * No answer is received for all requests
 - the requests are re-sent to IPx1, IPy1 and IPz1

- If no response is received after four attempts, the request is then sent to IPx2, IPy2 and IPz2

6. Security Considerations

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

This document does not specify how PCP server names are provisioned to the PCP client. It is the responsibility of PCP server provisioning document(s) to elaborate on the security considerations to discover a legitimate PCP server.

7. IANA Considerations

This document does not request any action from IANA.

8. Acknowledgements

Many thanks to D. Thaler for the review and comments.

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.

9.2. Informative References

- [I-D.ietf-pcp-dhcp] Boucadair, M., Penno, R., and D. Wing, "DHCP Options for the Port Control Protocol (PCP)", draft-ietf-pcp-dhcp-07 (work in progress), March 2013.
- [RFC1123] Braden, R., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, October 1989.
- [RFC3493] Gilligan, R., Thomson, S., Bound, J., McCann, J., and W. Stevens, "Basic Socket Interface Extensions for IPv6", RFC 3493, February 2003.

- [RFC4116] Abley, J., Lindqvist, K., Davies, E., Black, B., and V. Gill, "IPv4 Multihoming Practices and Limitations", RFC 4116, July 2005.
- [RFC6055] Thaler, D., Klensin, J., and S. Cheshire, "IAB Thoughts on Encodings for Internationalized Domain Names", RFC 6055, February 2011.

Appendix A. Multi-homing

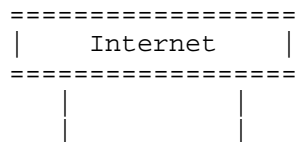
The main problem of a PCP multi-homing situation can be succinctly described as 'one PCP client, multiple PCP servers'. As described in Section 4, if a PCP client discovers multiple PCP server names, it should send requests to all of them in parallel with the following assumptions:

- o There is no requirement that multiple PCP servers have the same capabilities.
- o PCP requests to different servers are independent, the result of a PCP request to one server does not influence another.
- o If PCP servers provide NAT, it is out of scope how the client manages ports across PCP servers. For example, whether PCP client requires all external ports to be the same or whether there are ports available at all.

The following sub-sections describe multi-homing examples to illustrate PCP client behavior.

A.1. IPv6 Multi-homing

In this example of an IPv6 multi-homed network, two or more routers co-located with firewalls are present on a single link shared with the host(s). Each router is in turn connected to a different service provider network and the host in this environment would be offered multiple prefixes and advertised multiple DNS/NTP servers. Consider a scenario in which firewalls within an IPv6 multi-homing environment also implement a PCP server. PCP client learns of the available PCP servers using DHCP [I-D.ietf-pcp-dhcp] or any other PCP server discovery technique defined in future specifications. The PCP client will send PCP requests in parallel to each of the PCP servers.



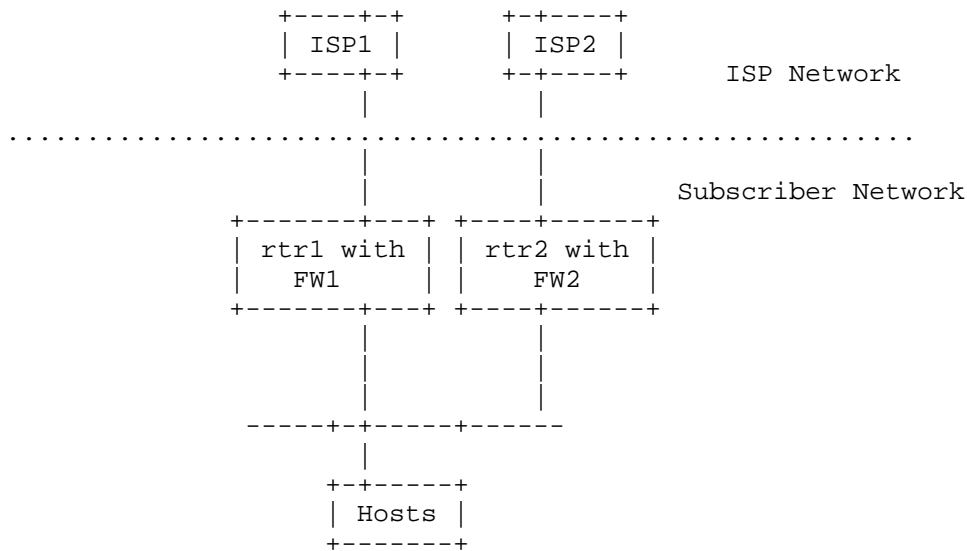
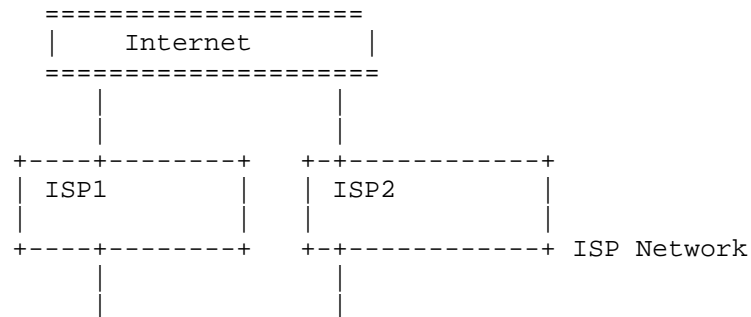


Figure 1: IPv6 Multihoming

A.2. IPv4 Multi-homing

In this example an IPv4 multi-homed network described in 'NAT- or RFC2260-based multi-homing' (Section 3.3 of[RFC4116]), the gateway router is connected to different service provider networks. This method uses PA addresses assigned by each transit provider to which the site is connected. The site uses NAT to translate the various provider addresses into a single set of private-use addresses within the site. In such a case, two PCP servers have to be present to control NAT to each of the transit providers. PCP client learns of the available PCP servers using DHCP [I-D.ietf-pcp-dhcp] or any other PCP server discovery technique defined in future specifications. The PCP client will send PCP requests in parallel to each of the PCP servers.



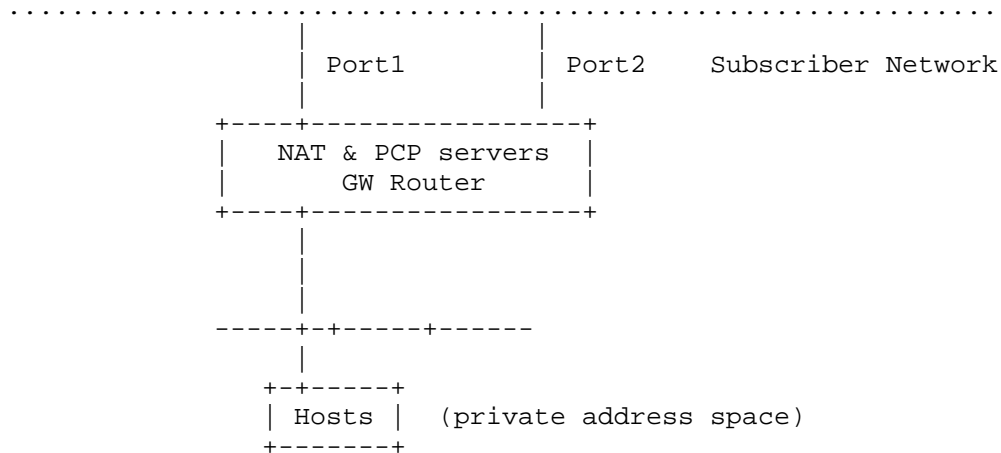


Figure 2: IPv4 Multihoming

A.3. Multiple interfaces and Servers

In case for multi-homing when an end host such as a mobile terminal has multiple interfaces concurrently active (e.g., IEEE 802.11 and 3G), a PCP client would discover different PCP servers over different interfaces. Although multiple interfaces are available, an application might choose to use just one based on, for example, cost and bandwidth requirements, and therefore would need to send PCP requests to just one PCP server.

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

Prashanth Patil
Cisco Systems, Inc.
Bangalore
India

Email: praspati@cisco.com

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

Email: tiredddy@cisco.com

PCP Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 08, 2014

H. Moustafa
D. Moses
Intel Corporation
M. Boucadair
France Telecom
November 04, 2013

PCP Extension for Signaling Feedback Information from the End-User
Application to the Application Server and to the Network
draft-mou-pcp-application-network-feedback-00

Abstract

Nowadays users consumption style for video and multimedia applications is strongly changing. Users are heavily counting on wireless and mobile devices for video streaming and interactive video and multimedia applications. This can be implemented for instance by having more intelligence in the service and the network infrastructure to better suit a differentiated users consumption style. This can be achieved through having: (i) a knowledge in the network and service platform about the available device and network conditions for the end-user and (ii) a knowledge in the network about the content requirements in terms of devices capabilities and network resources for content stored either in the network or in the application server. To obtain such knowledge with no need for changing the current infrastructure and in a generalized way to all applications, feedback/notification mechanisms between the end-user application, the network and the service platform is needed to provide information helping the content delivery and adaptation decisions. This document is investigating such application-agnostic track.

This document extends the Port Control Protocol (PCP) RFC 6887 [RFC6887] to allow: (i) the users application to notify the network and application server about its available resources in terms of devices capabilities and network conditions as well as information about the user (e.g., location, mobility status) and (ii) the application server to notify the network about the requirements of the content it stores in terms of devices and network resources. A new PCP option, denoted the FEEDBACK, is specified to allow such feedback notification signaling. This option is used with both PEER and MAP 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 May 08, 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. Use Cases | 3 |
| 3.1. Optimized Content Delivery by the Network | 4 |
| 3.2. Optimized Content Delivery by the Service Platform | 4 |
| 3.3. Network-based Video Session Seamless Experience across Devices | 5 |
| 3.4. Network-based User-Centric Video Content Adaptation | 5 |
| 3.5. Optimization Examples | 5 |
| 4. Why PCP? | 6 |
| 5. FEEDBACK PCP Option | 7 |
| 5.1. PCP Request and Responses | 9 |
| 6. Security Consideration | 11 |
| 7. IANA Consideration | 11 |
| 8. Acknowledgements | 11 |
| 9. References | 12 |
| 9.1. Normative References | 12 |
| 9.2. Informative References | 12 |
| Appendix A. Existing Protocols of Relevance to User's Feedback | |

| | |
|---|----|
| Information and their Limitations | 13 |
|---|----|

1. Introduction

Nowadays, Internet traffic includes huge amount of video traffic which is expected to dominate the world's mobile traffic in the coming years (66% of mobile video traffic according to Cisco forecast). The users' consumption style for video services is strongly evolving towards a user-centric model enabling video services access based on users' profiles and making receiving device (e.g., Laptops, tablets, smartphones, and future devices models) constitute the majority of end-user devices for video services through plenty of services over the Internet.

Although this big evolution, the network and service infrastructure are not optimized enough to handle the content delivery and adaptation function of the network resources, devices resources, application and content requirements. As a result, Quality of Experience (QoE) for video services and multimedia applications cannot be guaranteed in some situations.

This document defines an extension to the Port Control Protocol (PCP) RFC 6887 [RFC6887] allowing: (i) the end-user application to signal in real-time to the network and application server information about its available device capabilities and network resources (mainly device characteristics, buffering status as an indication of the network conditions as well as other useful context information (e.g., location, environment light/noise, mobility status) and (ii) the application server to signal in real-time to the network the requirement of the content it stores in terms of devices and network resources. The extension defines a new PCP option for the existing PEER and MAP OpCodes: FEEDBACK Option for signaling information between the end-users application, the network and the application server.

Motivations to undertake this effort are discussed in Section 3 through a number of use cases, while justifications for the use of PCP are elaborated in Section 4.

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 RFC 2119 [RFC2119].

3. Use Cases

The new model for video services consumption over mobile and wireless networks creates the need for feedback information between the end-user application, the network and the service platform in real-time. This helps optimizing the content for device/network resources on one hand and end-user QoE and battery experience on the other hand. In this view more intelligence needs to be considered in the network. A way to do that is through having several service functions in the network middle boxes on the path between the end-user and the application server, as shown in Draft [ServiceFunctions], that could be collocated with the NAT function or not. Examples of service functions that we consider to better manage the video delivery are: video optimization function (including transcoding), caching capabilities, TCP optimizer, video controller allowing for example content recommendation or intelligent scheduling in the case of networks. The following subsections present some related use-cases:

3.1. Optimized Content Delivery by the Network

In this case, the end-user's device is the host implementing the PCP client and a middle box network node in the network (e.g. edge router, home gateway or any cache node close to the user) is the PCP server. This middle box node can store a copy of the content or can be a content relay from the application server to the end-user. If the middle box network node is only a relay, then it receives the feedback information sent to the network by the end-user application and by the application server and makes use of this information for optimizing the content it relays to the end-user. If the middle box network node stores the content, then it makes use of the feedback information sent to the network by the end-user application to optimize the stored content when requested by the end-user. This use-case is beneficial in adaptive video streaming and server-based video conferencing applications. In the former, the middle box network node can be an edge router, home gateway or any network node caching the content. In the latter, the Multi-Point Control Unit (MCU), with which video conferencing clients communicate, is the middle box network node.

3.2. Optimized Content Delivery by the Service Platform

In this case, the end-user application is the host implementing the PCP client and the application server implements a PCP server. Feedback information is sent from the end-user application to the application server allowing the server to make intelligence decisions on the content adaptation. This use-case applies also in a Content Delivery Network (CDN) case, where several application servers exist and an application server controller, who plays the role of a PCP proxy, directs the end-user request to the appropriate application server (function of proximity for instance or load balancing).

3.3. Network-based Video Session Seamless Experience across Devices

Device awareness in terms of the device capabilities enables application content to be transferred from device to device or to be shared across multiple devices. In this case, the devices are located behind the same residential gateway, and therefore be reachable from outside with the same IPv4 address or IPv6 prefix. Each end-user device is a host implementing PCP client and sends feedback information on its devices resources to the middle box network node (e.g. CPE or home gateway) playing the role of PCP server. The functionalities related to the session transfer from device to device or sharing the same content across multiple devices is outside the scope of this document. Our focus is how to benefit from the feedback information during these cases as shown below:

If a video session started on device1 (e.g. smartphone) and is transferred to device2 (e.g. a tablet or TV screen) when it becomes available in the proximity of device1, the end-user application feedback information sent by device 2 will be used by the middle box network node to adapt the application content to match the device resources and network conditions of device 2.

If the same video is watched by multiple users with different devices resources and network conditions (e.g., video for a lecture in a classroom), the end-user application feedback information sent by each user device will be used by the middle box network node to adapt the application content to match the device resources and network conditions for each user device.

3.4. Network-based User-Centric Video Content Adaptation

In this case, the end-user device is the host implementing the PCP client, the application server storing the content is a PCP client, and a middle box network node that stores and/or relays the content implements a PCP server and receives feedback from both the end-user application and the application server. The feedback information (mainly on the location, battery level and mobility status) sent by the end-user application to the middle box network node is used in the decision of content adaptation (mainly the insertion of targeted advertisements). If the middle box network node does not store the content, then it can receive the native content from the application server then adapts it making use of the feedback information. If the middle box network node stores the content then it adapts it directly for each user.

3.5. Optimization Examples

The following are some examples on how to optimize the content for device and network resources on one hand and end-user QoE and battery experience on the other hand.

- o For a high display resolution, a video of high bitrate/resolution is sent if the users application buffering level and the global network bandwidth conditions are sufficient and the battery level is sufficient.
- o If the battery level degrades during the session even if the users application buffering level remains sufficient, then the video is continued to be streamed in lesser bitrate/resolution to save battery and prevent video session interruption due to battery failure (keeping a threshold on the quality based on the remaining resources). In this case, lesser resources can be allocated for cellular users to match the bandwidth required for the low bitrate video.
- o If the battery level is fine but the users application buffering level and global network bandwidth conditions degrades, the video switches to lower bitrate (following the ordinary adaptive streaming approach), which in turns save in the battery level.
- o For small size display, the transmission errors are less observed by the user especially if the user is mobile and is in a noisy environment and so video can be displayed with lesser quality (bitrate) to save battery resources and network resources even if the buffering level and global network bandwidth conditions are not poor. This is much useful for content categories as news and non-live content, in which transmission errors have lesser impact on the users perception.
- o If the user is in a bright/sunny environment and the battery level is not high, content can be sent with higher brightness but with lower bitrate to compensate the power consumed from content brightness.
- o If the user is in a dimly environment, no matter of the battery level and the global network bandwidth, the user can receive content with lower brightness. This has a dual benefit: allowing better perceived quality for the user and saving in battery resources.
- o Users can receive content including targeted advertisement for regional services according to each user location, mobility status and battery level suitability. This saves network bandwidth and devices resources (mainly battery resources) happening from the reception of bigger traffic volume if each user receives all the advertisements.

4. Why PCP?

The use of PCP to signal feedback/notification between the end-users application, the network and service platform in real-time is

motivated by the following: In the Port Control Protocol (PCP) specification RFC 6887 [RFC6887], PCP is viewed as a request/response protocol and also as a hint/notification protocol between a PCP client and a PCP server. Message flows in PCP are viewed as independent streams carrying information between the PCP client and the PCP server, in which the PCP client sends a stream of hints indicating to its server its mapping status and the PCP server sends to the client a notification informing the client on the actual state of its mapping. In this view of the protocol, PCP can be extended to carry more mapping information than the IP internal versus external addresses. Draft [Flowdata] is an example of the use of PCP for signaling by the client the flow characteristics to the network and signaling by the network its ability to accommodate that flow back to the client.

In addition, PCP allows learning and influencing the mapping lifetime, which helps reducing network bandwidth, overload on application servers and middle boxes and battery resources for wireless and mobile devices. This makes PCP suitable for conveying feedback information in our use-cases in real-time and resource wise way.

Further advantages for PCP motivating its use are as follows:

- o PCP can be used to install state in upstream devices such as NAT, firewalls or other flow-aware devices.
- o PCP can be used to notify a failure that may occur at an upstream PCP-controlled device, and therefore the PCP client can react accordingly.
- o PCP allows learning the lifetime of installed mappings and would therefore avoid overloading the network and service platform with keepalive messages. This also saves the battery resources for wireless and mobile end-user devices.
- o PCP can be used to notify the network with the flow characteristics so as to enforce policies at the access segment.
- o PCP can be used to receive informative information from the network so that client may use them to select the interface to use to place a session.
- o PCP can be extended easily to allow reporting capabilities to a remote server.
- o Extending PCP with the FEEDBACK feature avoids making assumptions on how media streams are exchanged (e.g., RTP, IAX mini-frames, etc.).
- o PCP extension does not require an OS support. The feature can be managed at the application level.

5. FEEDBACK PCP Option

This document presents an extension to PCP through the definition of FEEDBACK option in PCP. The FEEDBACK option aims to signal feedback information between the end-user application, the network and service platform to help optimizing the network and devices resources and enhancing the users experience. This feedback mechanism makes also use of the PCP FLOWDATA option [Flowdata]. This means that the PCP client sends both FLOWDATA and the FEEDBACK options in the same requests.

The information signaled in the FEEDBACK Option may include the screen size, screen resolution and battery level as device capabilities information and the users location, environment type and mobility status as context information on the user side. This information is signaled once at the beginning of the session and can be updated upon variation. Following the information signaling, the PCP server uses the FEEDBACK option to signal its ability of providing content with characteristics matching the device and network resources as well as the user context. This information will be used by the application server and by the network (through the middle box network nodes having service functions) for optimized content adaptation as explained in Section 3.

The FEEDBACK option does not make any assumption on the presence of NAT and/or firewall. In particular, PCP-based mechanism to instantiate state in an upstream NAT, firewall and any other flow-aware device are not impacted by the use of FEEDBACK option.

The PCP client is implemented at the end-user device and the application server (content server), and the PCP server is implemented at the middle box network node storing the content and the application server. The PCP client may be configured with multiple IP addresses; each of them pointing to a distinct PCP server. The PCP client will contact all these PCP server in parallel as discussed in [I-D.ietf-pcp-server-selection].

A PCP Proxy [PCPProxy] functionality can be enabled in intermediate nodes (e.g., Customer Premise Equipment (CPE)) on the path between the PCP client and the PCP server.

Upon receipt of the FEEDBACK option by the PCP Server, its content is passed to the Application Server that decides whether an action is needed to serve the requesting client to better accommodate its resources and conditions.

The procedure for generating a request that includes the FEEDBACK option, handling a request that includes the FEEDBACK option, and generating a response to a request that includes the FEEDBACK option is similar to the behavior specified in [Flowdata].

Triggers to generate a request that capture the network conditions and device resources in a FLOWDATA and FEEDBACK options are local to the application. How the application server or network middle boxes makes use of the content of the FLOWDATA and FEEDBACK options is also local to the PCP Server and to each the decision-making process at the Application Server side or network middle box node.

5.1. PCP Request and Responses

The PCP client follows the steps of generating the PEER and MAP opcodes request and response as described in [Flowdata]. The FEEDBACK option is included in the request and response according to the format described in this section.

Option Name: FEEDBACK
 Number: to be assigned by IANA
 Purpose: Describe to the network information on the end-user application and user context (e.g., device characteristics, buffering status as indication of the network conditions, location, environment light/noise, mobility status), and information on the content that can be provided by the application server.
 Valid for opcodes: PEER and MAP
 Length: 16 Octets
 May appear in: request. May appear in response only if it appeared in the associated request
 Maximum occurrences: 1

The FEEDBACK option request has the following format:

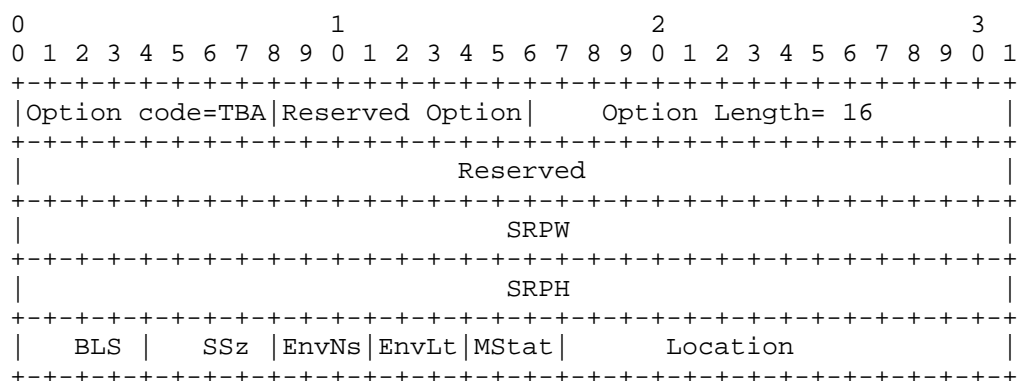


Figure 1: FEEDBACK Option Request

The fields are described as follows:

- o SRPW: Screen Resolution Pixels in Width, giving the number of pixels in width for the screen. This field takes a value 0 if no information/update is required to be sent.
- o SRPH: Screen Resolution Pixels in Height, giving the number of pixels in height for the screen. This field takes a value 0 if no information/update is required to be sent.
- o BLS: Battery Level Status. The following values are defined:
 - * 0 if no information/update is required to be sent
 - * 1= fading,
 - * 2= weak,
 - * 3= medium,
 - * 4 = high,
 - * 5= very high.
- o SSz: Screen Size of the device. The following values are defined:
 - * 0 if no information/update is required to be sent,
 - * 1= very small,
 - * 2= small,
 - * 3= medium,
 - * 4= big,
 - * 5= very big.
- o EnvNs: Environment noise level. The following values are defined:
 - * 0 if no information/update is required to be sent
 - * 1= very small,
 - * 2= small,
 - * 3= medium,
 - * 4= noisy,
 - * 5= very noisy.
- o EnvLt: Environment light level. The following values are defined:
 - * 0 if no information/update is required to be sent
 - * 1= poor,
 - * 2= dim,
 - * 3= good,
 - * 4= bright,
 - * 5= very bright
- o MStat: User activity and mobility status. The following values are defined:
 - * 0 if no information/update is required to be sent
 - * 1 = static,
 - * 2 = weak mobility (e.g., walking),
 - * 3= regular mobility (e.g., running),
 - * 4 = high mobility (e.g., in train)
- o Location: Gives the user location. This field takes a value 0 if no information/update is required to be sent.

The FEEDBACK option response has the following format:

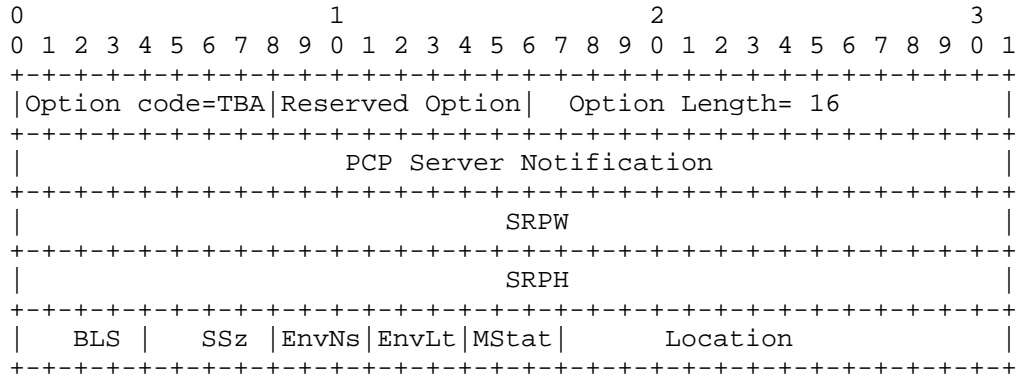


Figure 2: FEEDBACK Option Response

The field PCP Server Notification shows the response status for the sent request:

- o 0= request cannot be satisfied,
- o 1= the request can be partially satisfied,
- o 2= the request can be satisfied,
- o 3= the request can be fully satisfied.

The content of the remaining fields are echoed from the request.

6. Security Consideration

The security consideration for PCP in RFC 6887 [RFC6887] and the PCP client authentication [PCPAAuthentication] are sufficient to ensure security and host authorization for the proposed PCP extension in this document.

7. IANA Consideration

IANA is requested to assign a new PCP option called FEEDBACK in the IANA registry for PCP [pcp-iana].

8. Acknowledgements

Thanks for colleagues from Intel who provided valuable comments on this draft: Jeffrey Foerster, Somayazulu Srinivasa, Wu-Chi Feng and Muthaiah Venkatachalam.

Thanks also for Saso Stojanovski from Intel for the discussion and exchange on PCP within 3GPP.

9. References

9.1. Normative References

- [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-01 (work in progress), May 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6887] Wing, D., "Port Control Protocol (PCP)", RFC 6887, April 2013.

9.2. Informative References

- [Flowdata]
Wing, D., Penno, R., and T. Reddy, "PCP Flowdata Option", draft-wing-pcp-flowdata-00 (work in progress), July 2013.
- [I-D.ietf-httpbis-http2]
Belshe, M., Peon, R., Thomson, M., and A. Melnikov, "Hypertext Transfer Protocol version 2.0", draft-ietf-httpbis-http2-07 (work in progress), October 2013.
- [PCPAAuthentication]
Wasserman, M., Hartman, S., and D. Zhang, "Port Control Protocol (PCP) Authentication Mechanism", draft-ietf-pcp-authentication-02 (work in progress), October 2013.
- [PCPPProxy]
Boucadair, M., Penno, R., and D. Ding, "Port Control Protocol (PCP) Proxy Function", draft-ietf-pcp-proxy-03 (work in progress), June 2013.
- [RFC2616] Fielding, R., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.
- [RFC3840] Rosenberg, J., "Indicating User Agent Capabilities in Session Initiation Protocol (SIP)", RFC 3840, August 2004.
- [RFC4566] Handley, M., "SDP: Session Description Protocol", RFC 4566, July 2006.
- [ServiceFunctions]
Liu, W., Li, H., Huang, O., Boucadair, M., Leymann, N., Cao, Z., and J. Hu, "Service Function Chaining Use Cases",

draft-liu-sfc-use-cases-00 (work in progress), October 2013.

Appendix A. Existing Protocols of Relevance to User's Feedback Information and their Limitations

Some existing protocols can convey devices characteristics and information on the user, however with limited applicability. Examples are:

- o SIP [RFC3840]
 - * Provides a specification for capabilities and characteristics in SIP User Agent (UA). Capabilities and characteristics information is carried as parameters of the Contact header field and can be used within REGISTER requests and responses, OPTIONS responses, and requests and responses creating dialogs (such as INVITE).
 - * SIP limitation for the explained use-cases: i) the need of adding the SIP protocol stack in video streaming servers, ii) SIP does not rely on network entities and is mainly an application specific protocol, and iii) SIP is not appropriate for "live" real-time control with no network priority to SIP controls.
 - * Other Signalling protocols such as IAX are used to establish media sessions.
- o HTTP 1.1 [RFC2616] and HTTP2.0 [I-D.ietf-httpbis-http2]
 - * HTTP can incorporate information on the device and the user in its headers (e.g. Accept or Use-Agent headers), however this will not be a generic solution to any underlying transport protocol.
 - * Also, HTTP by its nature is a stateless protocol and activating HTTP proxy functionality impacts the performance of the network which limits the communication through proxies.
- o SDP [RFC4566]
 - * SDP is meant to provide a standard representation for session description without incorporating a transport protocol, without a focus on user's feedback information

- * SDP is used for the session negotiation at the beginning of the session and so for the devices characteristics and the user information could not be directly signaled in real-time
- * SDP uses protocols as SIP and RTSP that are not intercepted by middle nodes in the network which limits its applicability.
- * SDP is not used in some non-SIP deployment contexts.

Authors' Addresses

Hassnaa Moustafa
Intel Corporation
Hillsboro, OR 97124
USA

EMail: hassnaa.moustafa@intel.com

Danny Moses
Intel Corporation

EMail: danny.moses@intel.com

Mohamed Boucadair
France Telecom
Rennes 35000
France

EMail: mohamed.boucadair@orange.com