

PCP working group
Internet-Draft
Intended status: Standards Track
Expires: August 18, 2014

S. Kiesel
University of Stuttgart
R. Penno
Cisco Systems, Inc.
S. Cheshire
Apple
February 14, 2014

PCP Anycast Address
draft-ietf-pcp-anycast-01

Abstract

The Port Control Protocol (PCP) Anycast Address enables PCP clients to transmit signaling messages to their closest 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 Address.

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 August 18, 2014.

Copyright Notice

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

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	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	7
5. Security Considerations	9
6. References	10
6.1. Normative References	10
6.2. Informative References	10
Appendix A. Discussion of other PCP Discovery methods	11
A.1. Default Router	11
A.2. DHCP PCP Options	11
A.3. User Input	12
A.4. Domain Name System Based	12
A.5. Addressing only based on Destination Port	12
Appendix B. Discussion of IP Anycast Address usage for PCP	14
B.1. Motivation	14
B.2. Scenarios	14
B.3. Historical Objections to Anycast	14
Authors' Addresses	16

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), IPv6 and IPv4 firewall devices, and a mechanism to reduce application keep alive traffic.

The PCP 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 DHCP. The properties and drawbacks of various mechanisms are discussed in Appendix A.

This document follows a different approach: it establishes a well-known anycast address for the PCP Server. PCP clients are expected to send requests to this address during the PCP Server discovery process. A PCP Server configured with the anycast address could optionally redirect or return a list of unicast PCP Servers to the client. For a more extensive discussion on anycasting see Appendix B.

The benefit of using an anycast address is simplicity and reliability. In an example deployment scenario:

1. A network administrator installs a PCP-capable NAT.
2. An end user (who may be the same person) runs a PCP-enabled application. This application can implement PCP with purely user-level code -- no operating system support is required.
3. This PCP-enabled application sends its PCP request to the PCP anycast address. This packet travels through the network like any other, without any special support from DNS, DHCP, other routers, or anything else, until it reaches the PCP-capable NAT, which receives it, handles it, and sends back a reply.

Using the PCP anycast address, the only two things that need to be deployed in the network are the two things that actually use PCP: The PCP-capable NAT, and the PCP-enabled application. Nothing else in the network needs to be changed or upgraded, and nothing needs to be configured, including the PCP client.

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

2.1. PCP Discovery Client behavior

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

2.2. PCP Discovery Server behavior

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

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

3. Deployment Considerations

There are known limitations when there is more than one PCP server and asymmetric routing, or similar scenarios. Mechanisms to deal with those situations, such as state synchronization between PCP servers, are beyond the scope of this document.

4. IANA Considerations

4.1. Registration of IPv4 Special Purpose Address

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

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

1. The designated address prefix.

Prefix: TBD by IANA. Prefix length: /32

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

This document.

3. The date the designation was made.

TBD.

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

Unlimited. No termination date.

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

protocol service anycast.

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

N/A.

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

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

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

TBD.

4.2. Registration of IPv6 Special Purpose Address

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

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

1. The designated address prefix.

Prefix: TBD by IANA. Prefix length: /128

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

This document.

3. The date the designation was made.

TBD.

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

Unlimited. No termination date.

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

protocol service anycast.

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

N/A.

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

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

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

TBD.

5. Security Considerations

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 into the public Internet.

6. References

6.1. Normative References

- [RFC1546] Partridge, C., Mendez, T., and W. Milliken, "Host Anycasting Service", RFC 1546, November 1993.
- [RFC3958] Daigle, L. and A. Newton, "Domain-Based Application Service Location Using SRV RRs and the Dynamic Delegation Discovery Service (DDDS)", RFC 3958, January 2005.
- [RFC4773] Huston, G., "Administration of the IANA Special Purpose IPv6 Address Block", RFC 4773, December 2006.
- [RFC5736] Huston, G., Cotton, M., and L. Vegoda, "IANA IPv4 Special Purpose Address Registry", RFC 5736, January 2010.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

6.2. Informative References

- [DNSDisc] Hagino, J. and D. Thaler, "Analysis of DNS Server Discovery Mechanisms for IPv6", draft-ietf-ipngwg-dns-discovery-01 (work in progress), November 2001.
- [DhcpRequestParams] OpenFlow, "OpenFlow Switch Specification", February 2011, <<http://msdn.microsoft.com/en-us/library/windows/desktop/aa363298%28v=vs.85%29.aspx>>.
- [I-D.chen-pcp-mobile-deployment] Chen, G., Cao, Z., Boucadair, M., Ales, V., and L. Thiebaut, "Analysis of Port Control Protocol in Mobile Network", draft-chen-pcp-mobile-deployment-04 (work in progress), July 2013.
- [I-D.ietf-dhc-container-opt] Droms, R. and R. Penno, "Container Option for Server Configuration", draft-ietf-dhc-container-opt-07 (work in progress), April 2013.
- [I-D.ietf-pcp-dhcp] Boucadair, M., Penno, R., and D. Wing, "DHCP Options for the Port Control Protocol (PCP)", draft-ietf-pcp-dhcp-09 (work in progress), November 2013.

Appendix A. Discussion of other PCP Discovery methods

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

A.1. Default Router

The PCP specification allows one mode of operation in which the PCP client sends its requests to the default router. This approach is appropriate in a typical single-link residential network but has limitations in more complex network topologies.

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

A.2. DHCP PCP Options

One general drawback of relying on external configuration mechanisms, such as DHCP [I-D.ietf-pcp-dhcp], is that it creates an external dependency on another piece of network infrastructure which must be configured with the right address for PCP to work. In some environments the staff managing the DHCP servers may not be the same staff managing the NAT gateways, and in any case, constantly keeping the DHCP server address information up to date as NAT gateways are added, removed, or reconfigured, is burdensome.

Another drawback of relying on DHCP for configuration is that at least one significant target deployment environments for PCP -- namely 3GPP for mobile telephones -- does not use DHCP.

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

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

In scenarios where PCP Server resides on ISP network and is intended to work with arbitrary home gateways that don't know they are being used in a PCP context, that won't work, because there's no reason to think that the HGW will even request the option from the DHCP server,

much less offer the value it gets from the server on the customer-facing LAN. There is work on the DHC WG to overcome some of these limitations [I-D.ietf-dhc-container-opt] but in terms of deployment it also needs HGW to be upgraded.

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

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

A.3. User Input

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

A.4. Domain Name System Based

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

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

A.5. Addressing only based on Destination Port

One design option that was considered for Apple's NAT gateways was to have the NAT gateway simply handle and respond to all packets addressed to UDP port 5351, regardless of the destination address in the packet. Since the device is a NAT gateway, it already examines every packet in order to rewrite port numbers, so also detecting packets addressed to UDP port 5351 is not a significant additional burden. Also, since this device is a NAT gateway which rewrites port numbers, any attempt by a client to talk *though* this first NAT gateway to create mappings in some second upstream NAT gateway is futile and pointless. Any mappings created in the second NAT gateway are useful to the client only if there are also corresponding mappings created in the first NAT gateway. Consequently, there is no

case where it is useful for PCP requests to pass transparently through the first PCP-aware NAT gateway on their way to the second PCP-aware NAT gateway. In all cases, for useful connectivity to be established, the PCP request must be handled by the first NAT gateway, and then the first NAT gateway generates a corresponding new upstream request to establish a mapping in the second NAT gateway. (This process can be repeated recursively for as many times as necessary for the depth of nesting of NAT gateways; this is transparent to the client device.)

Appendix B. Discussion of IP Anycast Address usage for PCP

B.1. Motivation

The two issues identified in Appendix A.5 result in the following related observations: the PCP client may not **know** what destination address to use in its PCP request packets; the PCP server doesn't **care** what destination address is in the PCP request packets.

Given that the devices neither need to know nor care what destination address goes in the packet, all we need to do is pick one and use it. It's little more than a placeholder in the IP header. Any globally routable unicast address will do. Since this address is one that automatically routes its packet to the closest on-path device that implements the desired functionality, it is an anycast address.

B.2. Scenarios

In the simple case where the first-hop router is also the NAT gateway (as is common in a typical single-link residential network), sending to the PCP anycast address is equivalent to sending to the client's default router, as specified in the PCP base document [RFC6887].

In the case of a larger corporate network, where there may be several internal routed subnets and one or more border NAT gateway(s) connecting to the rest of the Internet, sending to the PCP anycast address has the interesting property that it magically finds the right border NAT gateway for that client. Since we posit that other network infrastructure does not need (and should not have) any special knowledge of PCP (or its anycast address) this means that to other non-NAT routers, the PCP anycast address will look like any other unicast destination address on the public Internet, and consequently the packet will be forwarded as for any other packet destined to the public Internet, until it reaches a NAT or firewall device that is aware of the PCP anycast address. This will result in the packet naturally arriving the NAT gateway that handles this client's outbound traffic destined to the public Internet, which is exactly the NAT gateway that the client wishes to communicate with when managing its port mappings.

B.3. Historical Objections to Anycast

In March 2001 a draft document entitled "Analysis of DNS Server Discovery Mechanisms for IPv6" [DNSDisc] proposed using anycast to discover DNS servers, a proposal that was subsequently abandoned in later revisions of that draft document.

There are legitimate reasons why using anycast to discover DNS

servers is not compelling, mainly because it requires explicit configuration of routing tables to direct those anycast packets to the desired DNS server. However, DNS server discovery is very different to NAT gateway discovery. A DNS server is something a client explicitly talks to, via IP address. The DNS server may be literally anywhere on the Internet. Various reasons make anycast an unconvincing technique for DNS server discovery:

- o DNS is a pure application-layer protocol, running over UDP.
- o On an operating system without appropriate support for configuring anycast addresses, a DNS server would have to use something like Berkeley Packet Filter (BPF) to snoop on received packets to intercept DNS requests, which is inelegant and inefficient.
- o Without appropriate routing changes elsewhere in the network, there's no reason to assume that packets sent to that anycast address would even make it to the desired DNS server machine. This places an additional configuration burden on the network administrators, to install appropriate routing table entries to direct packets to the desired DNS server machine.

In contrast, a NAT gateway is something a client's packets stumble across as they try to leave the local network and head out onto the public Internet. The NAT gateway has to be on the path those packets naturally take or it can't perform its NAT functions. As a result, the objections to using anycast for DNS server discovery do not apply to PCP:

- o No routing changes are needed (or desired) elsewhere in the local network, because the whole *point* of using anycast is that we want the client's PCP request packet to take the same forwarding path through the network as a TCP SYN to any other remote destination address, because we want the *same* NAT gateway that would have made a mapping in response to receiving an outbound TCP SYN packet from the client to be the one that makes a mapping in response to receiving a PCP request packet from the client.
- o A NAT engine is already snooping on (and rewriting) every packet it forwards. As part of that snooping it could trivially look for packets addressed to the PCP UDP port and process them locally (just like the local processing it already does when it sees an outbound TCP SYN packet).

Authors' Addresses

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

Email: ietf-pcp@skiesel.de
URI: <http://www.rus.uni-stuttgart.de/nks/>

Reinaldo Penno
Cisco Systems, Inc.
San Jose, CA
US

Phone:
Fax:
Email: repenno@cisco.com
URI:

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
Intended status: Experimental
Expires: August 11, 2014

M. Wasserman
S. Hartman
Painless Security
D. Zhang
Huawei
February 7, 2014

Port Control Protocol (PCP) Authentication Mechanism
draft-ietf-pcp-authentication-03

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 communications 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 proposes 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.

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 August 11, 2014.

Copyright Notice

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

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

Table of Contents

1. Introduction	3
2. Terminology	4
3. Protocol Details	5
3.1. Session Initiation	5
3.2. Session Termination	8
3.3. Session Re-Authentication	8
4. PA Security Association	9
5. Result Code	10
6. Packet Format	10
6.1. Packet Format of PCP Auth Messages	10
6.2. Authentication OpCode	11
6.3. Nonce Option	12
6.4. Authentication Tag Option for Common PCP	12
6.5. Authentication Tag Option for PCP Auth Messages	13
6.6. EAP Payload Option	14
6.7. PRF Option	15
6.8. MAC Algorithm Option	15
6.9. Session Lifetime Option	15
6.10. Received Packet Option	16
7. Processing Rules	16
7.1. Authentication Data Generation	16
7.2. Authentication Data Validation	17
7.3. Retransmission Policies for PCP Auth Messages	18
7.4. Sequence Numbers for PCP Auth Messages	18
7.5. Sequence Numbers for Common PCP Messages	19
7.6. MTU Considerations	20
8. IANA Considerations	20
9. Security Considerations	20
10. Acknowledgements	21
11. Change Log	21
11.1. Changes from wasserman-pcp-authentication-02 to ietf- pcp-authentication-00	21
11.2. Changes from wasserman-pcp-authentication-01 to -02	21
11.3. Changes from ietf-pcp-authentication-00 to -01	21
11.4. Changes from ietf-pcp-authentication-01 to -02	21
11.5. Changes from ietf-pcp-authentication-02 to -03	22

12. References	22
12.1. Normative References	22
12.2. Informative References	22
Authors' Addresses	23

1. Introduction

Using the Port Control Protocol (PCP) [RFC6887], an IPv4 or IPv6 host 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 proposes a PCP security extension which enables PCP servers to authenticate their clients with Extensible Authentication Protocol (EAP). The EAP messages are encapsulated within PCP packets during transportation.

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

- o Loss of EAP messages during transportation
- o Disordered 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 does not create implicit mappings.
- 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 explicit mappings than it is in authorizing implicit mappings.

- o For implementations that support the THIRD_PARTY Option (unless they can meet the constraints outlined in Section 14.1.2.2).
- o For implementations that wish to support any deployment scenario that does not meet the constraints described in Section 14.1.

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 device (e.g., a host) which 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 device (e.g., a NAT or a firewall) that implements the server-side of the PCP protocol, via which PCP clients request and manage explicit mappings. 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 (PCP-Auth) Session: A series of PCP message exchanges transferred between a PCP client and a PCP server. The PCP message involved within a session includes the PCP-Auth messages used to perform EAP authentication, key distribution and session management, and the common PCP messages secured with the keys distributed during authentication. Each PCP-Auth session is assigned a distinctive Session ID.

Session Partner: A PCP device involved within a PCP-Auth session. Each PCP-Auth session has two session partners (a PCP server and a PCP client).

Session Lifetime: The life period associated with a PCP-Auth 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 PCP-Auth session, using an EAP key generating method (e.g., the one defined in [RFC5448]).

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

Common PCP message: A PCP message which does not contain an Authentication OpCode. This document specifies an option, the Authentication Tag Option for Common PCP, 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 PCP-Auth session, a PCP client and a PCP server need to exchange a series of PCP-Auth messages in order to perform an EAP authentication process. Each PCP-Auth message is attached with an Authentication OpCode and may optionally contain a set of Options for various purposes (e.g., transporting authentication messages and session managements). The Authentication OpCode consists of two fields: Session ID and Sequence Number. The Session ID field is used to identify the session to which the message belongs. The sequence number field is used to detect the disorder or the duplication occurred during packet delivery.

When a PCP client intends to proactively initiate a PCP-Auth session with a PCP server, it sends a PCP-Auth-Initiation message (a PCP-Auth-Client message with the result code "INITIATION") to the PCP server. In the message, the Session ID and Sequence Number fields of the Authentication OpCode are set as 0. The PCP client MAY also optionally append a nonce option which consists of a random nonce with the message.

After receiving the PCP-Auth-Initiation, if the PCP server agrees to initiate a PCP-Auth session with the PCP client, it will reply with a PCP-Auth-Server message which contains an EAP Identity Request, and the result code field of this PCP-Auth-Server message is set as AUTHENTICATION-REQUIRED. In addition, the server MUST assign a

session identifier which can distinctly identify this session, and fill the identifier into the Session ID field of the Authentication OpCode in the PCP-Auth-Server message. The Sequence Number field of the Authentication OpCode is set as 0. If there is a nonce option in the received PCP-Auth-Initiation message, the PCP-Auth-Server MUST be attached with 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 the PCP-Auth-Server message. From now on, every PCP message within this session will be attached with this session identifier. When receiving a PCP-Auth message from an unknown session, a PCP device MUST discard the message silently. If the PCP client intends to simplify the authentication process, it MAY append an EAP Identity Response message within the PCP-Auth-Initiation message so as to inform the PCP server that it would like to perform EAP authentication and skip the step of waiting for the EAP Identity Request.

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 can reply with a PCP-Auth-Server message to initiate a PCP-Auth session. The result code field of this PCP-Auth-Server message is set as AUTHENTICATION-REQUIRED. In addition, the PCP server MUST assign a session ID for the session and transfer it within the PCP-Auth-Server message. The Sequence Number field in the PCP-Auth-Server is set as 0. In the PCP-Auth messages exchanged afterwards in this session, the session ID MUST 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 PCP-Auth-Server message from the PCP server, it can reply with a PCP-Auth-Client message or silently discard the request message according to its local policies. In the PCP-Auth-Client message, a nonce option which consists of a random nonce MAY be appended. If so, in the next PCP-Auth-Server message, the PCP sever MUST forward the nonce back within a nonce option.

In a PCP-Auth session, an EAP request message is transported within a PCP-Auth-Server message, and an EAP answer message is transported within a PCP-Auth-Client message. EAP relies on the underlying protocol to provide reliable transmission; any disordered delivery or loss of packets occurred during transportation must be detected and addressed. Therefore, after sending out a PCP-Auth-Server message, the PCP server will not send a new PCP-Auth-Server message until it receives a PCP-Auth-Client message with a proper sequence number from the PCP client, and vice versa. If a PCP device receives a PCP-Auth message from its partner and cannot generate a EAP response within a pre-specified period due to certain reasons (e.g., waiting for human input to construct a EAP message or waiting for the additional PCP-Auth messages in order to construct a complete EAP message), the PCP

device MUST reply with a PCP-Auth-Acknowledge message (PCP-Auth messages with a Received Packet Option) to notify the packet has been received. This approach not only can avoid un-necessarily retransmission of the PCP-Auth message but also can guarantee the reliable packet delivery in the conditions where a PCP device needs to receive multiple PCP-Auth messages before generating an EAP response.

In this approach, it is mandated for a PCP client and a PCP server to perform a key-generating EAP method in authentication. 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 traffic 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 packets. In order to do this, the PCP server needs to append a set of PRF Options and MAC Algorithm Options to the initial PCP-Auth-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. After receiving the options, the PCP client selects the PRF and the MAC algorithm which it would like to use, and then attach the associated PRF and MAC Algorithm Options to the next PCP-Auth-Client message.

After the EAP authentication, the PCP server sends out a PCP-Auth-Server message to indicate the EAP authentication and PCP authorization results. If the EAP authentication succeeds, the result code of the PCP-Auth-Server message is AUTHENTICATION-SUCCEED. In this case, before sending out the PCP-Auth-Server message, the PCP server MUST generate a PCP SA and use the derived transport key to generate a digest for the message. The digest is transported within an Authentication Tag Option for PCP Auth. A more detailed description of generating the authentication data can be found in Section 7.1. In addition, the PCP-Auth-Server MAY also contain a Session Lifetime Option which indicates the life-time of the PCP-Auth session (i.e., the life-time of the MSK). After receiving the PCP-Auth-Server message, the PCP client then needs to generate a PCP-Auth-Client message as response. If the PCP client also authenticates the PCP server, the result code of the PCP-Auth-Client is AUTHENTICATION-SUCCEED. In addition, the PCP client needs to generate a PCP SA and uses the derived traffic key to secure the message. From then on, all the PCP messages within the session are secured with the traffic key and the MAC algorithm specified in the PCP SA, unless a re-authentication is performed.

If a PCP client/server cannot authenticate its session partner, the device sends out a PCP-Auth message with the result code, AUTHENTICATION-FAILED. If the EAP authentication succeeds but

Authorization fails, the device making the decision sends out a PCP-Auth message with the result code, AUTHORIZATION-FAILED. In these two cases, after the PCP-Auth message is sent out, the PCP-Auth session MUST be terminated immediately.

3.2. Session Termination

A PCP-Auth session can be explicitly terminated by sending a termination-indicating PCP-Auth message (a PCP-Auth message with a result code "SESSION-TERMINATION") from either session partner. After receiving a Termination-Indicating message from the session partner, a PCP device MUST respond with a Termination-Indicating PCP-Auth message and remove the PCP-Auth SA immediately. When the session partner initiating the termination process receives the PCP-Auth message, it will remove the associated PCP-Auth SA immediately.

3.3. Session Re-Authentication

A session partner may select to perform EAP re-authentication if it would like to update the PCP SA (e.g., update the MSK and rollback the sequence numbers, or extend the session life period) without initiating a new PCP-Auth session.

When the PCP server would like to initiate a re-authentication, it sends the PCP client a PCP-Auth-Server message. The result code of the message is set to "RE-AUTHENTICATION", which indicates the message is for an re-authentication process. If the PCP client would like to start the re-authentication, it will send an PCP-Auth-Client message to the PCP server, the result code of the PCP-Auth-Client message is set to "RE-AUTHENTICATION". Then, the session partners exchange PCP-Auth messages to transfer EAP messages for the re-authentication. During the re-authentication procedure, the session partners protect the integrity of PCP-Auth messages with the key and MAC algorithm specified in the current PCP SA; the sequence numbers associated with the packet will never be rolled back and keep increasing according to Section 7.3.

If the EAP re-authentication succeeds, the result code of the last PCP-Auth-Server is "AUTHENTICATION-SUCCEED". In this case, before sending out the PCP-Auth-Server, the PCP server must update the SA and use the new key to generate digests to protect the integrity and authenticity of the PCP-Auth-Server and any subsequent PCP message. In addition, the PCP-Auth-Server MAY be appended with a Session Lifetime Option which indicates the new life-time of the PCP-Auth session.

If the EAP authentication fails, the result code of the last PCP-Auth-Server is "AUTHENTICATION-FAILED". If the EAP authentication

succeeds but Authorization fails, the result code of the last PCP-Auth-Server is "AUTHORIZATION-FAILED". In the latter two cases, the PCP-Auth session MUST be terminated immediately after the last PCP-Auth message exchange.

4. PA Security Association

At the beginning of a PCP-Auth session, a session SHOULD generate a PCP-Auth SA to maintain its state information during the session. The parameters of a PCP-Auth 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 PCP-Auth message
- o Sequence number for the next incoming PCP-Auth message
- o Sequence number for the next outgoing common PCP message (included in the SA for PCP client)
- o Sequence number for the next incoming common PCP message (included in the SA for PCP client)
- o Last outgoing message payload
- o Retransmission interval
- o MSK: The master session key generated by the EAP method.
- o MAC algorithm: The algorithm that the transport key should use to generate digests for PCP messages.
- o Pseudo-random function: The pseudo random function negotiated in the initial PCP-Auth-Server and PCP-Auth-Client exchange for the transport key derivation
- o Transport key: the key derived from the MSK to provide integrity protection and data origin authentication for the messages in the PCP-Auth session. The life-time of the transport key SHOULD be identical to the life-time of the session.
- o The nonce selected by the PCP client at the initiation of the session.

- o Key ID: the 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 The 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 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 PCP-Auth-Client packet. If the PCP client does not select one, this value is set as 0.
- o Key ID: The ID assigned for the traffic key.

5. Result Code

This message use the result code field specified in the PCP headers to transport the information for authentication and session management. Particularly, the values of following result codes are specified.

TBD INITIATION

TBD AUTHENTICATION-REQUIRED

TBD AUTHENTICATION-FAILED

TBD AUTHENTICATION-SUCCEED

TBD AUTHORIZATION-FAILED

TBD SESSION-TERMINATION

6. Packet Format

6.1. Packet Format of PCP Auth Messages

The format of PCP-Auth-Server messages is identical to the response packet format specified in Section 7.2 of [RFC6887].

As illustrated in Figure 1, the PCP-Auth-Client messages use the requester header specified in Section 7.1 of [RFC6887]. The only difference is that eight reserved bits are used to transfer the result codes (e.g., "INITIATION", "AUTHENTICATION-FAILED"). Other fields in Figure 1 are described in Section 7.1 of [RFC6887].

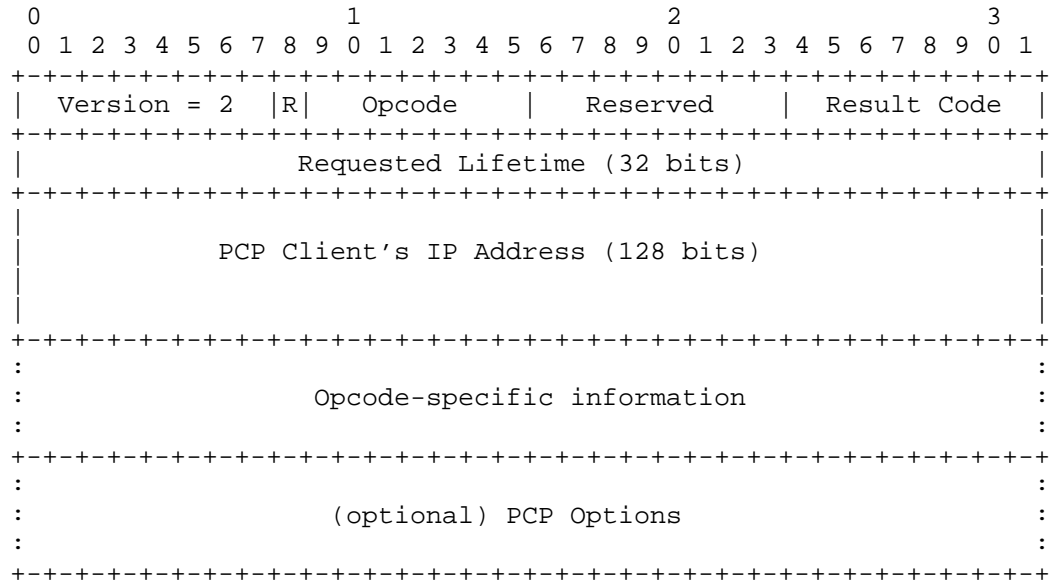
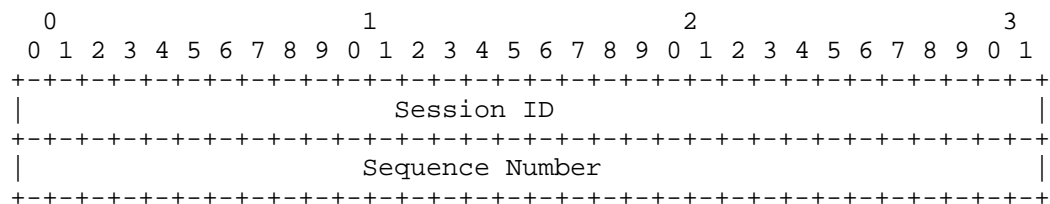


Figure 1. PCP-Auth-Client message Format

6.2. Authentication OpCode

The following figure illustrates the format of an authentication OpCode:



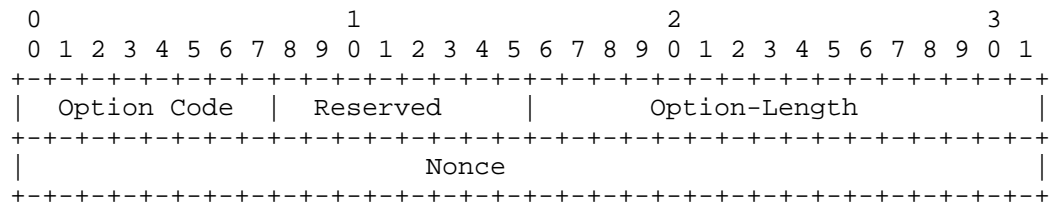
Session ID: This field contains a 32-bit PCP-Auth session identifier.

Sequence Number: This field contains a 32-bit sequence number. In this solution, a sequence number needs to be incremented on every

new (non-retransmission) outgoing packet in order to provide ordering guarantee for PCP.

6.3. Nonce Option

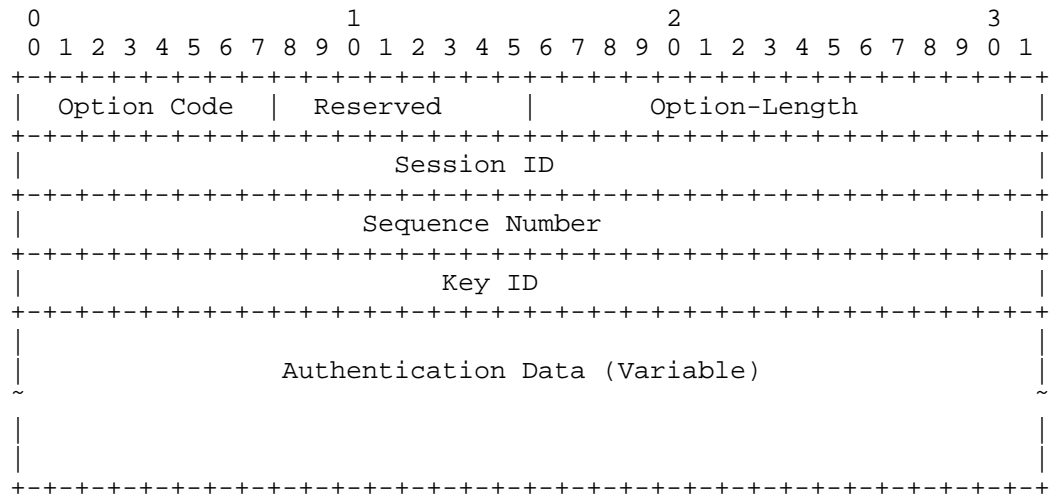
Because the session identifier of PCP-Auth session is determined by the PCP server, a PCP client does not know the session identifier which will be used when it sends out a PCP-Auth-Initiation message. In order to prevent an attacker from interrupting the authentication process by sending off-line generated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



Option-Length: The length of the Nonce Option (in octet), including the 4 octet fixed header and the variable length of the authentication data.

Nonce: A random 32 bits number which is transported within a PCC-Initiate message and the corresponding reply message from the PCP server.

6.4. Authentication Tag Option for Common PCP



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

Session ID: A 32-bit field used to indicates the identifier of the session that the message belongs to and identifies the secret key used to create the message digest appended to the PCP message.

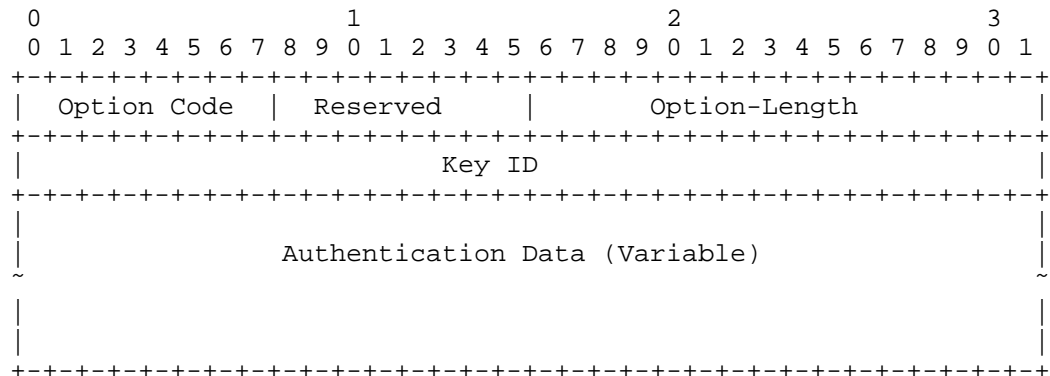
Sequence Number: This field contains a 32-bit sequence number. In this solution, a sequence number needs to be incremented on every new (non-retransmission) outgoing packet in order to provide ordering guarantee for common PCP messages.

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

Authentication Data: A variable-length field that carries the Message Authentication Code for the PCP packet. The generation of the digest can be various according to the algorithms specified in different PCP SAs. This field **MUST** end on a 32-bit boundary, padded with 0's when necessary.

6.5. Authentication Tag Option for PCP Auth Messages

This option is used to provide message authentication for PCP-Auth messages. Compared with the Authentication Tag Option for Common PCP, the session ID field and the sequence number field are removed because such information is provided in the Authentication OpCode.

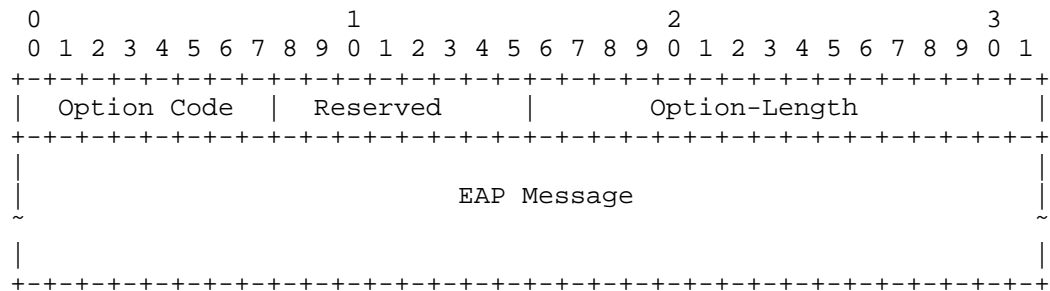


Option-Length: The length of the Authentication Tag Option for PCP Auth (in octet), including the 12 octet fixed header and the variable length of the authentication data.

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

Authentication Data: A variable-length field that carries the Message Authentication Code for the PCP packet. The generation of the digest can be various according to the algorithms specified in different PCP SAs. This field MUST end on a 32-bit boundary, padded with 0's when necessary.

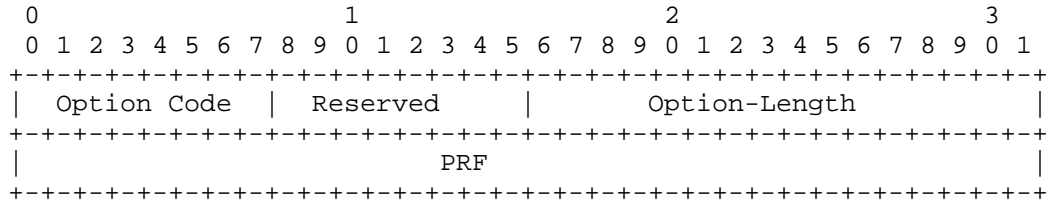
6.6. EAP Payload Option



Option-Length: The length of the EAP Payload Option (in octet), including the 4 octet fixed header and the variable length of the EAP message.

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

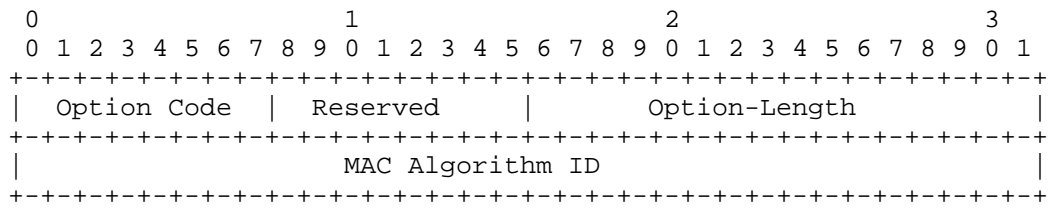
6.7. PRF Option



Option-Length: The length of the PRF Option (in octet), including the 4 octet fixed header and the variable length of the EAP message.

PRF: The Pseudo-Random Function which the sender supports to generate an MSK. This field contains an IKEv2 Transform ID of Transform Type 2 [RFC4306][RFC4868]. A PCP implementation MUST support PRF_HMAC_SHA2_256 (5).

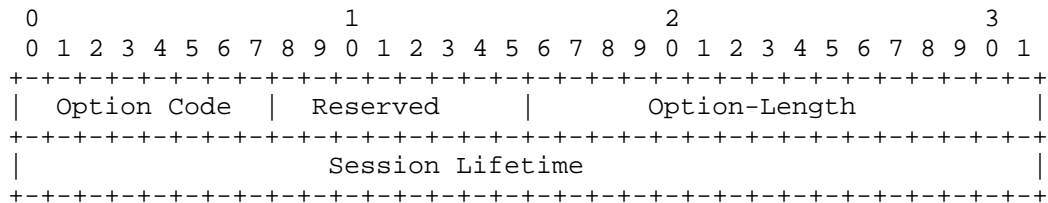
6.8. MAC Algorithm Option



Option-Length: The length of the MAC Algorithm Option (in octet), including the 4 octet fixed header and the variable length of the EAP message.

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 [RFC4306][RFC4868]. A PCP implementation MUST support AUTH_HMAC_SHA2_256_128 (12).

6.9. Session Lifetime Option

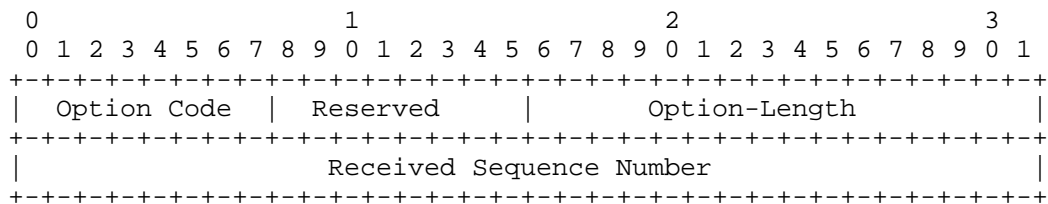


Option-Length: The length of the Session Lifetime Option (in octet), including the 4 octet fixed header and the variable length of the EAP message.

Session Lifetime: The life time of the PCP-Auth Session, which is decided by the authorization result.

6.10. Received Packet Option

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



Option-Length: The length of the Received Packet Option (in octet), including the 4 octet fixed header and the variable length of the EAP message.

Received Sequence Number: The sequence number of the last received PCP packet.

7. Processing Rules

7.1. Authentication Data Generation

If a PCP SA is generated as the result of a successful EAP authentication process, every subsequent PCP message within the session MUST carry an Authentication Tag Option which contains the digest of the PCP message for data origin authentication and integrity protection.

Before generating a digest for a PCP-Auth message, a device needs to first locate the PCP SA according to the session identifier and then get the traffic key. Then the device appends an 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 traffic 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 Authentication Tag Option) using the traffic key and the associated MAC algorithm, and insert the generated digest into the Authentication Data field.

Similar to generating a digest for a PCP-Auth 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 traffic key. Then the device appends the Authentication Tag Option for common PCP at the end of the message. The length of the Authentication Data field is decided by the MAC algorithm adopted in the session. The device then use the corresponding values derived from the SA to fills 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 traffic key and the associated MAC algorithm, and inputs the generated digest into the Authentication Data field.

7.2. Authentication Data Validation

When a device receives a common PCP packet with an Authentication Tag Option for Common PCP, 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 key ID in the option) and the MAC algorithm. If no proper SA or traffic key is found, the PCP packet MUST be discarded 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 packet (including the PCP header and Authentication Tag Option) with the transport key and the MAC algorithm found in the first step. If the value of the newly generated digest is identical to the stored one, the device can ensure that the packet has not been tampered with, and the validation succeeds. Otherwise, the packet MUST be discarded.

Similarly, when a device receives a PCP Auth packet with an Authentication Tag Option for PCP Auth, 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 key ID in the option) and the MAC algorithm. If no proper SA or traffic key is found, the PCP packet MUST be discarded 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 packet (including the PCP header and Authentication Tag Option) with the transport key and the MAC algorithm found in the first step. If the value of the newly generated digest is identical to the stored one, the device can ensure that the packet has not been tampered with, and the validation succeeds. Otherwise, the packet MUST be discarded.

7.3. Retransmission Policies for PCP Auth Messages

Because EAP relies on the underlying protocols to provide reliable transmission, after sending a PCP-Auth message, a PCP client/server MUST NOT send out any subsequent messages until receiving an expected PCP-Auth message (the PCP-Auth message with a proper sequence number) from the peer. If no such a message is received in a certain period, the PCP device will re-send the last message according to certain 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.

Note that the last PCP-Auth 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 PCP-Auth messages.

When a device receives such a duplicate PCP-Auth message from its session partner, it MUST try to answer it by sending the last outgoing PCP-Auth message again. The rate of replying the duplicate PCP-Auth messages MUST be limited.

7.4. Sequence Numbers for PCP Auth Messages

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

The above rules are not applied to PCP-Auth-Acknowledgement messages (i.e., PCP-Auth messages containing a Received Packet Option). A PCP-Auth-Acknowledgement message does not transport any EAP message and only indicate that a PCP-Auth message is received. Therefore, the

reliable transmission of PCP-Auth-Acknowledgement message does not have to be guaranteed. Therefore, when receiving or sending out a PCP-Auth-Acknowledgement message, the device MUST not increase the corresponding sequence number stored in the SA. Otherwise, the lost of a PCP-Auth-Acknowledgement message during transportation will cause the mismatching issues with the sequence numbers.

Another exception is in the message retransmission scenarios. When a device does not receive any response from its session partner in a certain period, it needs to retransmit the last outgoing PCP-Auth message with a limited rate. The duplicate messages and the original message MUST use the identical sequence number. When the device receives such a duplicate PCP-Auth message from its session partner, it MUST try to answer it by sending the last outgoing PCP-Auth message again. Note the rate of replying the duplicate PCP-Auth messages must be limited. In such cases, the maintained incoming and outgoing sequence numbers will not be affected by the message retransmission.

7.5. Sequence Numbers for Common PCP Messages

When transporting common PCP messages within a PCP-Auth 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 messages, the PCP device attaches the outgoing sequence number for common PCP messages to the messages and increments the sequence number maintained in the SA by 1.

When receiving a PCP packet from its session partner, the PCP device will not accept it if the sequence number carried in the packet is smaller than the incoming sequence number the server maintains. This approach can protect the PCP server from replay attacks. After confirming that the received packet is valid, the PCP server will use the sequence number in the incoming packet to take place the incoming sequence number for common PCP messages maintained in the SA.

Note that the sequence number in the incoming packet may not exactly match the incoming sequence number maintained locally. In the base PCP specification [RFC6887], a PCP client may stop retransmitting a PCP request without receiving any expected PCP answer when the client is no longer interested in the PCP transaction. After that, the PCP client will try to generate new PCP requests for other purposes. In this case, the sequence number in the new request will be larger than the incoming sequence number maintained in the PCP server.

7.6. MTU Considerations

EAP methods are responsible for MTU handling, so no special facilities are required in this protocol to deal with MTU issues. If an EAP message is too long for a single PCP-Auth message to transport, it will be divided into multiple sections and transport them within different PCP-Auth messages. Note that the receiver may not be able to know what to do in the next step until receiving all the sections and constructing the complete EAP message. In this case, in order to guarantee reliable message transmission, after receiving a PCP-Auth message, the receiver **MUST** reply with a PCP-Auth-Acknowledgement message until all the sections have been received.

8. IANA Considerations

TBD

9. Security Considerations

This section applies only to the in-band key management mechanism. It will need to be updated if the WG choose to pursue the out-of-band key management mechanism discussed above.

In this work, after a successful EAP authentication process performed between two PCP devices, a MSK will be exported. The MSK can 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 PCP-Auth messages exchanged within a PCP-Auth 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 PCP-Auth-Server and PCP-Auth-Client 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 PCP-Auth-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 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 PCP-Auth-Server and PCP-Auth-Client exchanges. The choice of EAP method is also very important. The selected EAP method must be resilient to the attacks possibly in an insecure network environment, and the user-identity confidentiality, protection against dictionary attacks, and session-key establishment must be supported.

10. Acknowledgements

11. Change Log

11.1. 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.

11.2. 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 traffic 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.

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

- o Editorial changes, added use cases to introduction.

11.4. 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

11.5. 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.

12. References

12.1. Normative References

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

12.2. Informative References

- [RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowitz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.
- [RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005.
- [RFC4868] Kelly, S. and S. Frankel, "Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec", RFC 4868, May 2007.
- [RFC5191] Forsberg, D., Ohba, Y., Patil, B., Tschofenig, H., and A. Yegin, "Protocol for Carrying Authentication for Network Access (PANA)", RFC 5191, May 2008.
- [RFC5448] Arkko, J., Lehtovirta, V., and P. Eronen, "Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA')", RFC 5448, May 2009.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

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: zhangdacheng@huawei.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 02, 2014

S. Perreault, Ed.
Viagenie
M. Boucadair
France Telecom
R. Penno
D. Wing
Cisco
S. Cheshire
Apple
January 29, 2014

Port Control Protocol (PCP) Proxy Function
draft-ietf-pcp-proxy-05

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 August 02, 2014.

Copyright Notice

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

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents

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

Table of Contents

1. Introduction	2
1.1. Use Case: the NAT Cascade	3
1.2. Use Case: the PCP Relay	4
2. Terminology	4
3. Operation of the PCP Proxy	5
3.1. Optimized Hairpin Routing	7
3.2. Termination of Recursion	8
3.3. Source Address for PCP Requests Sent Upstream	8
3.4. Unknown OpCodes and Options	9
3.5. Mapping Repair	9
3.6. Multiple PCP Servers	10
4. IANA Considerations	10
5. Security Considerations	10
6. Acknowledgements	11
7. References	11
7.1. Normative References	11
7.2. Informative References	11
Authors' Addresses	11

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.

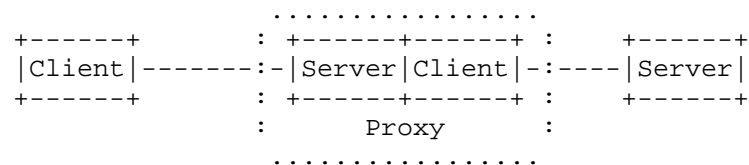


Figure 1: Reference Architecture

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 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 Wi-Fi 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 Wi-Fi gateway, such as an Apple AirPort Express, to their hotel room Ethernet port to share their room's Internet connection between their phone, their iPad, 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 [PCP] (NAT-PMP's IETF Standards Track successor), 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 making outgoing TCP connections to appear, from the point of view of everything upstream of the NAT gateway, to be a single client device making outgoing TCP connections. 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 (Internet Service Provider). This is illustrated in Figure 2.

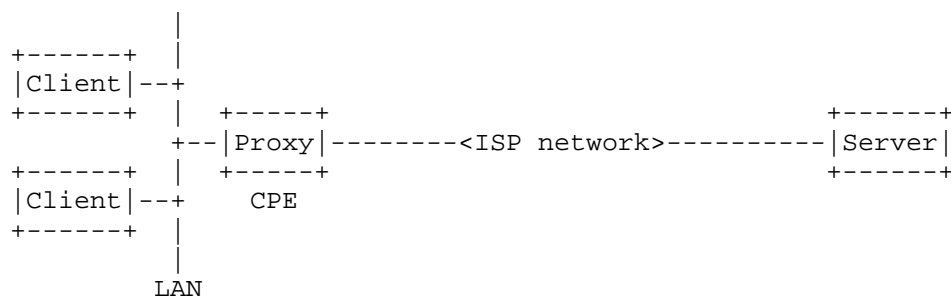


Figure 2: PCP Relay Use Case

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.

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].

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.

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:

- o An ISP's NAT gateway could be configured to know that it is the outermost NAT gateway, and consequently does not need to relay PCP requests upstream. In fact, it may be the case that many large-scale NATs of the kind used by ISPs may simply not implement Recursive PCP, thereby naturally terminating the recursion at that point.
- o A NAT 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.

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

[Editor's note: I think this section is severely broken. I'll leave it as-is for this revision and will start discussion on the list.]

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 It returns an UNSUPP_OPCODE error response when it receives a request with an unknown OpCode.
- o It returns an UNSUPP_OPTION error response when it receives a 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 [I-D.ietf-pcp-server-selection], 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 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 option 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 be removed) if it is not compatible with security controls provisionned 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.ietf-pcp-server-selection] Boucadair, M., Penno, R., Wing, D., Patil, P., and T. Reddy, "PCP Server Selection", draft-ietf-pcp-server-selection-02 (work in progress), January 2014.
- [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.

Authors' Addresses

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

Phone: +1 418 656 9254
Email: simon.perreault@viagenie.ca
URI: <http://viagenie.ca>

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

Port Control Protocol
Internet-Draft
Intended status: Standards Track
Expires: April 22, 2012

R. Penno
Juniper Networks
October 20, 2011

PCP Support for Multi-Zone Environments
draft-penno-pcp-zones-01

Abstract

A zone is a notion which denotes a routing instance, a set interfaces or prefixes characterized by having a different address realm and/or security policy. A NAT device can route packets with the same source IP address to different zones depending on configuration policies such as destination IP address. This functionality has been present for many years in NAT devices from multiple vendors. PCP allows a host to interact with a PCP-controlled NAT device and request an external IP and port. Therefore a PCP Server that controls the NAT device and receives a PCP request from a host needs to know from which NAT pool to allocate an external IP address and port. This document specifies an extension to PCP to support the zone concept.

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 April 22, 2012.

Copyright Notice

Copyright (c) 2011 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. Terminology	3
1.2. Problem Statement	3
1.3. Scope	4
2. PCP Base Support for Multiple Zones	4
2.1. PCP PEER Request	4
2.2. PCP MAP Request	5
3. PCP Extension for Multiple Zones	5
4. IANA Considerations	6
5. Security Considerations	6
6. Acknowledgements	6
7. References	6
7.1. Normative References	6
7.2. Informative References	7
Author's Address	8

1. Introduction

A zone is a routing instance, set interfaces or prefixes characterized by having a different address domain or security policy. A NAT device is present on each zone through NAT pools which are used to translate packet to and from a zone. The PCP protocol allows a host to interact with a NAT device and request a external IP and port. Since a NAT Device can route packets with the same source IP address to different Zones depending on policy or packet match conditions, the PCP Server that interacts with the NAT device and receives a PCP request from a host needs to know from which NAT pool to allocate an IP address and port.

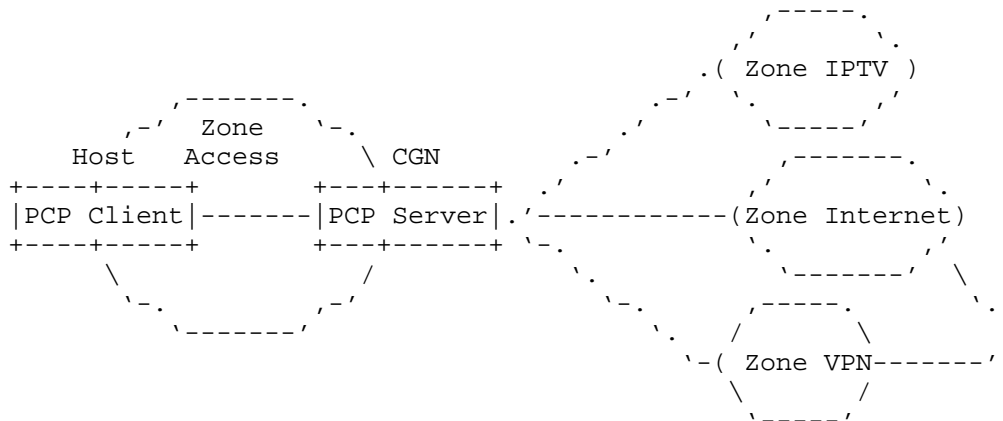
1.1. Terminology

This document uses PCP terminology defined in [I-D.ietf-pcp-base]]. In addition the following terms are defined in this document:

- o Zone: A routing instance, set of interfaces or network prefixes that has a separate addressing domain or security policy.
- o Address Domain: A collection of IP addresses. A NAT device is present on each domain through one or more NAT pools associated with each Zone.

1.2. Problem Statement

A PCP Server can control a NAT attached to distinct zones; each zone is characterised by one or several address pools. In such environment the NAT must rely on a pre-configured policy to determine which address pool to use when handling an IP packet coming from an internal host. An example of such policy may be to rely on the destination IP address, DSCP value(s), protocol (e.g., SIP, RTP, RTSP), etc.



The core of the problem is that packets from the same source IP address can be routed to any of the zones depending on match conditions based on the 5-tuple. Moreover, sessions could be initiated from any of these zones toward the host. These zones many times have different addressing domains and therefore different NAT pools. This means that packets from the host will use a different NAT pool depending on the destination zone.

It is important to notice that zones (or similar concept) has been present in Enterprise NAT and CGN from multiple vendors for many years. It is the advent and interaction with PCP that has created a need for a standardized approach.

1.3. Scope

The matching conditions that ultimately decide where to route a packet can be very elaborate including even application layer information. But the scope of this document is to abstract such implementation specific approaches behind the concept of a Zone-ID.

2. PCP Base Support for Multiple Zones

Before discussing extensions to the PCP protocol in the following sections we discuss how to support multiple zones with the current methods present in the base PCP protocol.

2.1. PCP PEER Request

A PCP PEER request could contains the destination IP address, port and Transport protocol of the peer the host will be trying to communicate . In that case, if the NAT device maintains a mapping of

zones (and associated NAT pools) to network prefixes it can choose the appropriate NAT pool. It is important to understand that this will only work if the policy that decides to which Zone to route packets is only based on the information present on the PCP PEER request.

Therefore if the PCP Client knows it is behind a NAT with zone support, it is RECOMMENDED that it includes the remote peer's 5-tuple in the PCP PEER request in the connect-then-lifetime case. If the peer's 5-tuple is not present in the PCP request, the external IP and port returned in the message is non-deterministic.

2.2. PCP MAP Request

In the case of PCP MAP request the NAT device does not know from which zone to install a mapping and consequently from which NAT pool to choose an external IP address and port. A FILTER Option may be included to allow the PCP Server select the external address pool to use. If other information than the destination IP address is used to drive the selection of the external address pool, additional information is required to be conveyed in the PCP MAP request (e.g., DSCP marking policy (see <http://tools.ietf.org/html/draft-boucadair-pcp-extensions-01#section-3>)).

3. PCP Extension for Multiple Zones

The proposed PCP extension is a new PCP Option that would convey the Zone-ID. The Zone-ID is an opaque identifier that is known by the PCP Client and the PCP-controlled NAT device. The procedure to provision the Zone-ID is out of scope.

When the NAT device receives a PCP request with a Zone-ID, it will use that or a derivative of it to determine the NAT pool from which to allocate an IP address and port.

Option Name: ZONEID

Number: TBA (IANA); Mandatory to process

Purpose: It allows the client request and server indicate from which Zone-ID the external IP:port were allocated.

Valid for Opcodes: MAP, PEER

Length: Variable

May appear in: both

Maximum occurrences: 1

4. IANA Considerations

TBD

5. Security Considerations

Subscribers can only request ports for the specific Zone-IDs allowed in their security profile. For example, in a typical Wireless deployment, mobile terminals could request mappings in zones 'Internet', 'HTTP Proxy Farm', and 'Video Farm'. A PCP request that contains a zone-id considered a security violation would be silently dropped.

6. Acknowledgements

Thanks to Mohamed Boucadair for early review comments

7. References

7.1. Normative References

- [RFC0959] Postel, J. and J. Reynolds, "File Transfer Protocol", STD 9, RFC 959, October 1985.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2766] Tsirtsis, G. and P. Srisuresh, "Network Address Translation - Protocol Translation (NAT-PT)", RFC 2766, February 2000.
- [RFC2960] Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L., and V. Paxson, "Stream Control Transmission Protocol", RFC 2960, October 2000.
- [RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

- [RFC4966] Aoun, C. and E. Davies, "Reasons to Move the Network Address Translator - Protocol Translator (NAT-PT) to Historic Status", RFC 4966, July 2007.
- [RFC5382] Guha, S., Biswas, K., Ford, B., Sivakumar, S., and P. Srisuresh, "NAT Behavioral Requirements for TCP", BCP 142, RFC 5382, October 2008.
- [RFC5508] Srisuresh, P., Ford, B., Sivakumar, S., and S. Guha, "NAT Behavioral Requirements for ICMP", BCP 148, RFC 5508, April 2009.

7.2. Informative References

- [I-D.ietf-behave-address-format]
 Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", draft-ietf-behave-address-format-10 (work in progress), August 2010.
- [I-D.ietf-behave-dns64]
 Bagnulo, M., Sullivan, A., Matthews, P., and I. Beijnum, "DNS64: DNS extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", draft-ietf-behave-dns64-11 (work in progress), October 2010.
- [I-D.ietf-behave-ftp64]
 Beijnum, I., "An FTP ALG for IPv6-to-IPv4 translation", draft-ietf-behave-ftp64-12 (work in progress), July 2011.
- [I-D.ietf-behave-v6v4-framework]
 Baker, F., Li, X., Bao, C., and K. Yin, "Framework for IPv4/IPv6 Translation", draft-ietf-behave-v6v4-framework-10 (work in progress), August 2010.
- [I-D.ietf-behave-v6v4-xlate-stateful]
 Bagnulo, M., Matthews, P., and I. Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", draft-ietf-behave-v6v4-xlate-stateful-12 (work in progress), July 2010.
- [I-D.ietf-pcp-base]
 Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", draft-ietf-pcp-base-16 (work in progress), October 2011.

- [I-D.wing-behave-dns64-config]
Wing, D., "IPv6-only and Dual Stack Hosts on the Same Network with DNS64", draft-wing-behave-dns64-config-03 (work in progress), February 2011.
- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.
- [RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010.
- [RFC5853] Hautakorpi, J., Camarillo, G., Penfield, R., Hawrylyshen, A., and M. Bhatia, "Requirements from Session Initiation Protocol (SIP) Session Border Control (SBC) Deployments", RFC 5853, April 2010.

Author's Address

Reinaldo Penno
Juniper Networks
1194 N Mathilda Avenue
Sunnyvale, California 94089
USA

Email: rpenno@juniper.net

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: August 17, 2014

A. Ripke
T. Dietz
J. Quittek
NEC
R. da Silva
Telefonica I+D
February 13, 2014

PCP Tunnel-ID Option
draft-ripke-pcp-tunnel-id-option-00

Abstract

This document describes a new Port Control Protocol (PCP) option called TUNNEL_ID. It serves for identifying a Third Party in addition to the means that PCP's THIRD_PARTY option already provides for that purpose.

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 August 17, 2014.

Copyright Notice

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

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
3. Target Scenario	3
4. Format	4
5. Behavior	5
5.1. Generating a Request	5
5.2. Processing a Request	6
5.3. Processing a Response	6
6. Alternative	6
7. IANA Considerations	6
8. Security Considerations	6
9. References	7
9.1. Normative References	7
9.2. Informative References	7
Authors' Addresses	7

1. Introduction

The IETF has specified the Port Control Protocol (PCP) ([RFC6887]) to control how packets are translated and forwarded by a PCP-controlled device such as a network address translator (NAT) or firewall.

This draft focuses on the application of PCP's THIRD_PARTY option that is used when the PCP client sends requests that concern other internal hosts than the host of the PCP client. This is, for example, the case if port mapping requests for a carrier grade NAT (CGN) are not sent from PCP clients at the subscribers, but from a portal of the carrier at which subscribers can request port mappings.

The issue addressed by the TUNNEL_ID option is that there are CGN deployments that do not distinguish internal hosts by their IP address only, but use further identifiers for unique subscriber identification. This is, for example, the case if a CGN supports overlapping private IP address spaces according to [RFC1918] for internal hosts of different subscribers. Then different internal hosts are identified and mapped at the CGN by their IP address and an additional ID, for example, the ID of a tunnel between the CGN and the subscriber. In such cases, the IP address contained in the THIRD_PARTY option is not sufficient. An additional identifier needs to be carried by the PCP protocol in order to uniquely identify the Internal Host. The TUNNEL_ID option serves this purpose.

The TUNNEL_ID option is defined for use in combination with the THIRD_PARTY option for the PCP opcodes MAP and PEER.

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].

The terminology defined in the specification of PCP [RFC6887] applies.

3. Target Scenario

This section illustrates the use of the TUNNEL_ID option in a scenario for a port mapping requests via a carrier portal.

The scenario shown in Figure 1 has a carrier operating a CGN and a portal for subscribers to request port mappings at the CGN. The portal communicates with the CGN using PCP. For this purpose the portal is co-located with a PCP client and the CGN is co-located with a PCP server. The way subscribers interact with the portal for requesting port mapping for their internal hosts is not specified in this scenario.

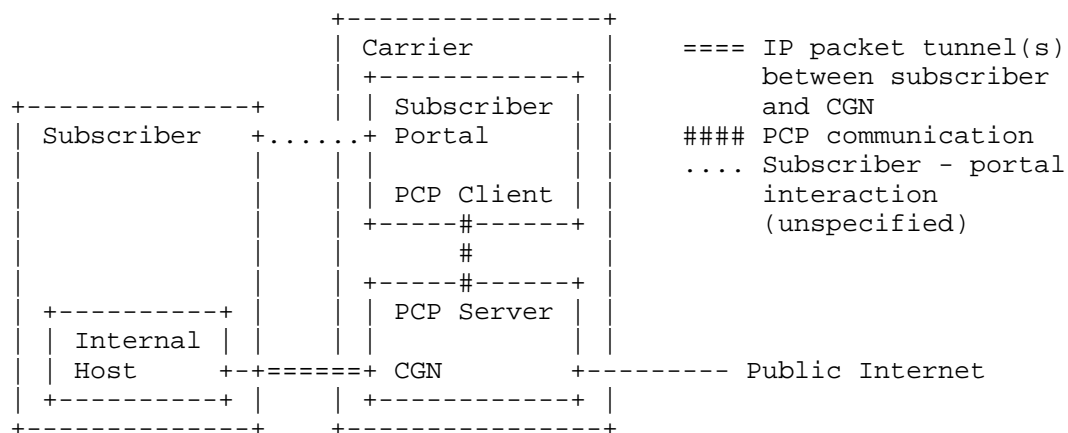


Figure 1: Carrier portal for port mapping requests

The internal hosts use private IP addresses as specified in [RFC1918]. Since there is no NAT between the internal host and the CGN, there is an overlap of addresses used by internal hosts at different subscribers. That is why the CGN needs more than just the

internal host's IP address to distinguish internal hosts at different subscribers. A commonly deployed method for solving this issue is using an additional identifiers for this purpose. A very good candidate for this additional identifier at the CGN is the ID of the tunnel that connects the CGN to the subscriber's network.

Requests for port mappings from the portal to the CGN need to uniquely identify the internal host for which a port mapping is to be established or modified. Already existing for this purpose is the THIRD_PARTY option that can be used to specify the internal host's IP address. The TUNNEL_ID option is introduced for carrying the additional (tunnel) information needed to identify the internal host in this scenario.

The additional identifier for internal hosts needs to be included in MAP requests from the PCP client in order to uniquely identify the internal host that should have its address mapped. This is the purpose that the new TUNNEL_ID serves in this scenario. It carries the additional identifier, that is the tunnel ID, that serves for identifying an internal host in combination with the internal host's (private) IP address. The IP address of the internal host is included in the PCP client's mapping requests by using the THIRD_PARTY option.

The information carried by the TUNNEL_ID is not just needed to identify an internal host in a PCP request. The CGN needs this information in its internal mapping tables for translating packet addresses and for forwarding packets to subscriber-specific tunnels.

4. Format

The TUNNEL_ID option is formatted as shown in Figure 2.

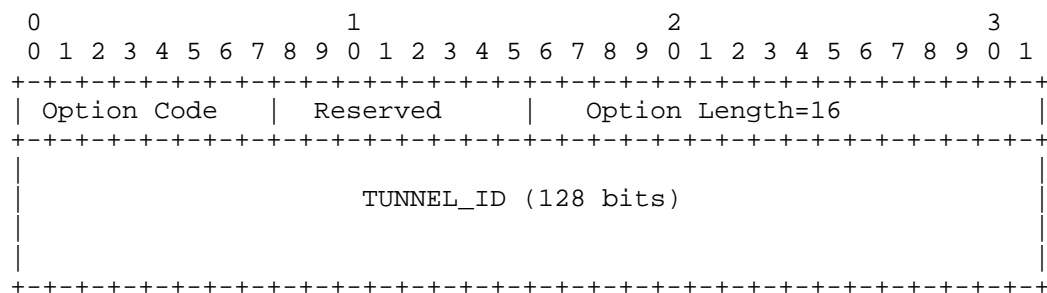


Figure 2: TUNNEL_ID Option

- o Option Name: TUNNEL_ID

- o Number: TBD
- o Purpose: Identifies a request of an external IP address and port.
- o Valid for opcodes: MAP, PEER, and all other for which the THIRD_PARTY option is valid for.
- o Length: 16 octets
- o May appear in: Request. Must appear in response if it appeared in the associated request.
- o Maximum occurrences: 1

The fields are as follows:

- o TUNNEL_ID: A vendor specific tunnel identifier that can be used to identify a subscriber's CGN session and the port ranges to apply this request to.

The tunnel identifier field can contain any vendor specific value to identify a tunnel. The option number is in the mandatory-to-process range (0-127), meaning that a request with a TUNNEL_ID option is executed by the PCP server if and only if the TUNNEL_ID option is supported by the PCP server.

5. Behavior

The following sections describe the operations of a PCP client and a PCP server when generating the request and processing the request and response.

5.1. Generating a Request

In addition to generating a PCP request that is described in [RFC6887] the following has to be applied. The TUNNEL_ID option can be used together either with a PCP MAP or PEER opcode. It MUST be used in combination with the THIRD_PARTY option which provides an IP address and port entered by the subscriber. The TUNNEL_ID option holds the respective tunnel identifier to allow the CGN to uniquely identify the internal host (specified in the THIRD_PARTY option) for which the port mapping is to be established or modified. If the tunnel identifier is shorter than 128 bits then the TUNNEL_ID option field is to be filled up with leading zeros up to 128 bits.

5.2. Processing a Request

The TUNNEL_ID option is in the mandatory-to-process range and if the PCP server does not support this option it MUST return an UNSUPP_OPTION response. If the provided TUNNEL_ID is unknown/unavailable the PCP server MUST return a TUNNEL_ID_UNKNOWN response.

5.3. Processing a Response

If the PCP client receives a TUNNEL_ID_UNKNOWN response back for its previous request it SHOULD report an error message. To where to report an error message is implementation dependent.

6. Alternative

An alternative to identify a tunnel affiliation in the given scenario could be using the DESCRIPTION ([I-D.ietf-pcp-description-option]) option to carry a tunnel ID option. The DESCRIPTION option is to allow a text description to be attached to a port mapping. But using the DESCRIPTION option for a tunnel ID might not be appropriate because it specifies using UTF-8 and another requirement is that the description text must not be null terminated, which cannot always be met.

7. IANA Considerations

The following PCP Option Code is to be allocated in the mandatory-to-process range:

TUNNEL_ID

[NOTE for IANA: Please allocate a PCP Option Code at <http://www.iana.org/assignments/pcp-parameters/pcp-parameters.xml#option-rules>]

The following PCP Result Code is to be allocated:

TUNNEL_ID_UNKNOWN

[NOTE for IANA: Please allocate a PCP Result Code at <http://www.iana.org/assignments/pcp-parameters/pcp-parameters.xml#result-codes>]

8. Security Considerations

As this option is related to the use of the THIRD_PARTY option the corresponding security considerations apply. Especially, the network on which the PCP messages are sent must be fully trusted.

9. References

9.1. Normative References

- [RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.
- [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-description-option]
Boucadair, M., Penno, R., and D. Wing, "PCP Description Option", draft-ietf-pcp-description-option-04 (work in progress), February 2014.

Authors' Addresses

Andreas Ripke
NEC
Heidelberg
Germany

Email: ripke@neclab.eu

Thomas Dietz
NEC
Heidelberg
Germany

Email: dietz@neclab.eu

Juergen Quittek
NEC
Heidelberg
Germany

Email: quittek@neclab.eu

Rafael Lopez da Silva
Telefonica I+D
Madrid
Spain

Email: ralds@tid.es

PCP
Internet-Draft
Intended status: Standards Track
Expires: August 9, 2014

D. Wing
T. Reddy
P. Patil
R. Penno
Cisco
February 5, 2014

PCP Extension for Third Party Authorization
draft-wing-pcp-third-party-authz-02

Abstract

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

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

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

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

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

This Internet-Draft will expire on August 9, 2014.

Copyright Notice

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

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Notational Conventions	3
3. Problem Statement	3
4. Solution Overview	5
5. Obtaining a Token Using OAuth	6
5.1. ACCESS_TOKEN Option	7
5.2. Generating the ACCESS_TOKEN option	9
5.3. PCP server processing ACCESS_TOKEN option	10
5.4. Processing the PCP response	11
6. PCP Server and Proxy behavior	11
7. Usage with PCP Authentication mechanism	12
8. Security Considerations	12
9. IANA Considerations	13
10. Acknowledgements	13
11. References	13
11.1. Normative References	13
11.2. Informative References	14
Authors' Addresses	15

1. Introduction

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

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

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

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

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

2. Notational Conventions

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

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

3. Problem Statement

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

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

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

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

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

As a result firewalls block media traffic.

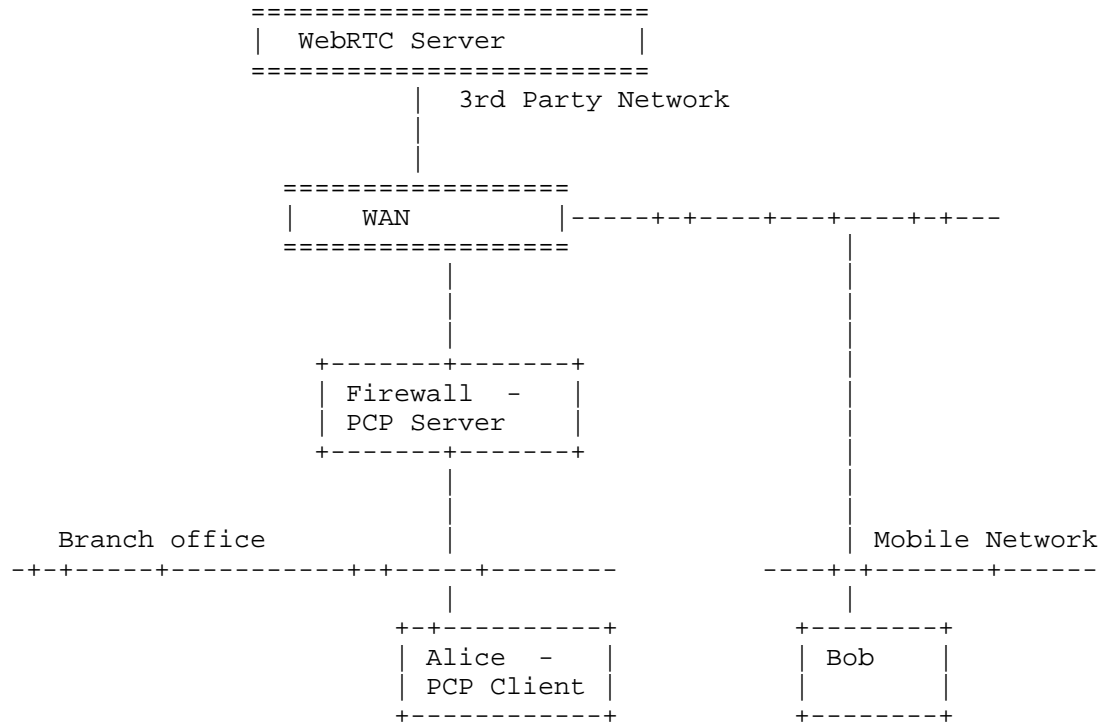
A mitigation to the problems above is for an enterprise to deploy a TURN server in the DMZ and have WebRTC clients use the TURN server. The use-case explained in Section 4.2.5.1 of [I-D.ietf-rtcweb-use-cases-and-requirements] refers to deploying a TURN [RFC5766] server to audit all media sessions from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

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

- o Enterprise firewalls would typically have granular policies to permit calls initiated using selected WebRTC servers (Dr. Good) it trusts and block the rest (Dr. Evil).
- o A TURN server just provides a 5-tuple (source IP address, destination IP address, protocol number, source port number, and destination port number) for auditing and no other details of the WebRTC or SIP server being used to establish the call.
- o A TURN server could increase media latency as explained in section 4.1.2.2 of [RFC5245].
- o A TURN server could either be located in the DMZ of the enterprise network or located in the public Internet. If the TURN server is located in the public Internet it comes at a high cost to the provider of the TURN server, since the server typically needs a high-bandwidth connection to the Internet as explained in the Introduction of [RFC5766]. As a consequence, it is best to use a TURN server only when a direct communication path cannot be found. When the client and a peer use ICE to determine communication path, ICE will use hole punching techniques to search for a direct path first and only use a TURN server when a direct path cannot be found.
- o Other limitations of TURN are explained in section 2.6 of [RFC5766]. For example the value of Diffserv field may not be preserved, Explicit Congestion Notification (ECN) field may be reset etc.

4. Solution Overview

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



Users : Alice, Bob

WebRTC Server : OAuth 2.0 Authorization server

Figure 1: WebRTC server in a different administrative domain

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

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

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

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

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

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

5. Obtaining a Token Using OAuth

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

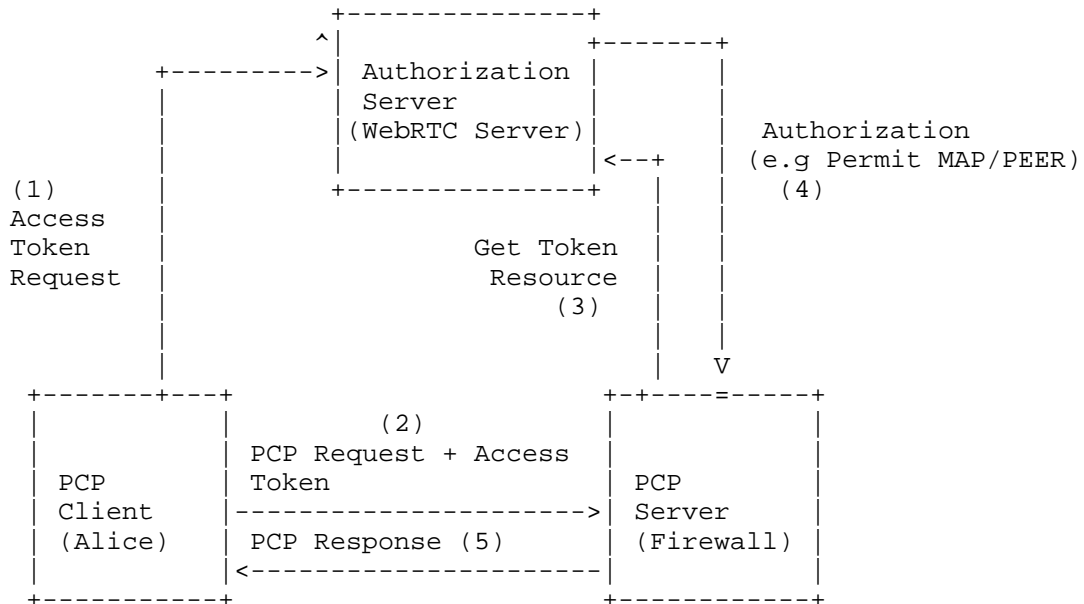
The following mapping of OAuth concepts to PCP is used :

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

Figure 2: OAuth terminology mapped to PCP terminology

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

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



User : Alice

Figure 3: Interactions

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

5.1. ACCESS_TOKEN Option

This specification defines a new PCP ACCESS_TOKEN Option that is described in Figure 4.

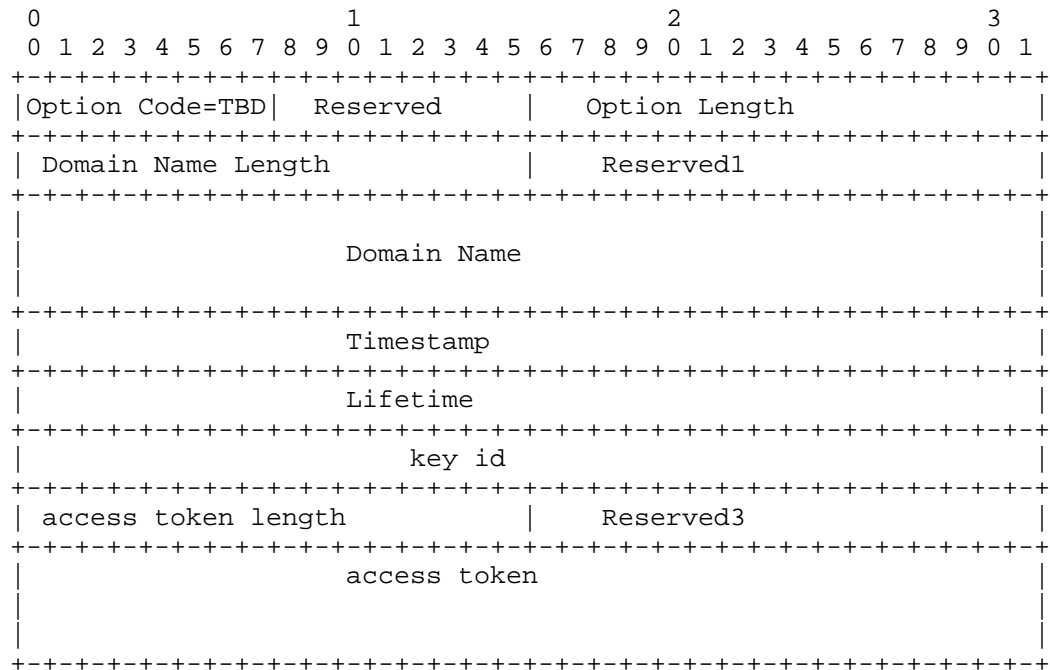


Figure 4: PCP ACCESS_TOKEN Option

The fields are described below:

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

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

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

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

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

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

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

key id: key id, which is an identifier generated by the authorization server. It generates this key id by computing a hash over the access token using SHA-1 and truncating the hash to 96 bits (retaining the left most bits).

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

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

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

Option Name: ACCESS_TOKEN

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

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

Valid for Opcodes: MAP, PEER

May appear in : request. May appear in response only if it appeared in the associated request.

Maximum occurrences : 1

5.2. Generating the ACCESS_TOKEN option

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

5.3. PCP server processing ACCESS_TOKEN option

A PCP server performs processing in the order described below.

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

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

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

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

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

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

Handle token type was selected for the following reasons :

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

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

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

5.4. Processing the PCP response

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

6. PCP Server and Proxy behavior

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

A PCP Proxy MUST follow the rules mentioned in section of 7 of [I-D.ietf-pcp-proxy] when the processing the ACCESS_TOKEN option.

7. Usage with PCP Authentication mechanism

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

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

8. Security Considerations

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

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

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

9. IANA Considerations

We request IANA register the PCP option ACCESS_TOKEN and the result code AUTHORIZATION_REQUIRED in [pcp-registry].

10. Acknowledgements

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

11. References

11.1. Normative References

- [I-D.ietf-pcp-authentication]
Wasserman, M., Hartman, S., and D. Zhang, "Port Control Protocol (PCP) Authentication Mechanism", draft-ietf-pcp-authentication-02 (work in progress), October 2013.
- [I-D.ietf-pcp-proxy]
Perreault, S., Boucadair, M., Penno, R., Wing, D., and S. Cheshire, "Port Control Protocol (PCP) Proxy Function", draft-ietf-pcp-proxy-05 (work in progress), February 2014.
- [I-D.reddy-pcp-auth-req]
Reddy, T., Patil, P., Wing, D., and R. Penno, "PCP Authentication Requirements", draft-reddy-pcp-auth-req-04 (work in progress), July 2013.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.
- [RFC6407] Weis, B., Rowles, S., and T. Hardjono, "The Group Domain of Interpretation", RFC 6407, October 2011.
- [RFC6749] Hardt, D., "The OAuth 2.0 Authorization Framework", RFC 6749, October 2012.
- [RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013.

[pcp-registry]

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

11.2. Informative References

[I-D.ietf-rtcweb-overview]

Alvestrand, H., "Overview: Real Time Protocols for Brower-based Applications", draft-ietf-rtcweb-overview-08 (work in progress), September 2013.

[I-D.ietf-rtcweb-use-cases-and-requirements]

Holmberg, C., Hakansson, S., and G. Eriksson, "Web Real-Time Communication Use-cases and Requirements", draft-ietf-rtcweb-use-cases-and-requirements-12 (work in progress), October 2013.

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

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

[RFC5766] Mahy, R., Matthews, P., and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", RFC 5766, April 2010.

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

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

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

Authors' Addresses

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

Email: dwing@cisco.com

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

Email: tiredy@cisco.com

Prashanth Patil
Cisco Systems, Inc.
Bangalore
India

Email: praspati@cisco.com

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

Email: repenno@cisco.com