PCP Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 15, 2012

M. Boucadair
France Telecom
R. Penno
Juniper Networks
D. Wing
Cisco
R. Dupont
Internet Systems Consortium

September 12, 2011

Port Control Protocol (PCP) Proxy Function
draft-bpw-pcp-proxy-02

Abstract

This document specifies the behavior of a PCP Proxy element, for instance embedded in Customer Premise routers.

Status of this Memo

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

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

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

This Internet-Draft will expire on March 15, 2012.

Copyright Notice

Table of Contents

1. Introduction 3
2. PCP Server Discovery and Provisioning 3
3. PCP Proxy as a PCP Server 4
4. Control of the Firewall 4
5. Embedded NAT in the CP Router 4
6. Simple PCP Proxy 6
7. Smart Proxy 7
7.1. Multiple PCP Servers 7
7.2. Epoch Handling 8
7.3. Request/Response Caching 8
7.4. Retransmission Handling 9
7.5. Full State 9
8. IANA Considerations 9
9. Security Considerations 9
10. References 10
10.1. Normative References 10
10.2. Informative References 11
Authors' Addresses 11

1. Introduction

PCP [I-D.ietf-pcp-base] discusses the implementation of NAT control features that rely upon Carrier Grade NAT (CGN) devices such as DS-Lite AFTR [RFC6333].

The Customer Premise router, the B4 element in DS-Lite, is in charge to enforce some security controls on PCP requests so implements a PCP Proxy function: it acts as a PCP server receiving PCP requests on internal interfaces, and as a PCP client forwarding accepted PCP requests on an external interface to a CGN PCP server. The CGN PCP server in turn send replies (PCP responses) to the PCP Proxy external interface which are finally forwarded to PCP clients.

The PCP Proxy can be simple, i.e., implement as transparent/minimal processing as possible, or it can be smart, i.e., handle multiple CGN PCP servers, cache requests/responses, etc. A smart Proxy can be associated with UPnP IGD [I-D.bpw-pcp-upnp-igd-interworking] or/and NAT-PMP [I-D.bpw-pcp-nat-pmp-interworking] Interworking Function (IWF).

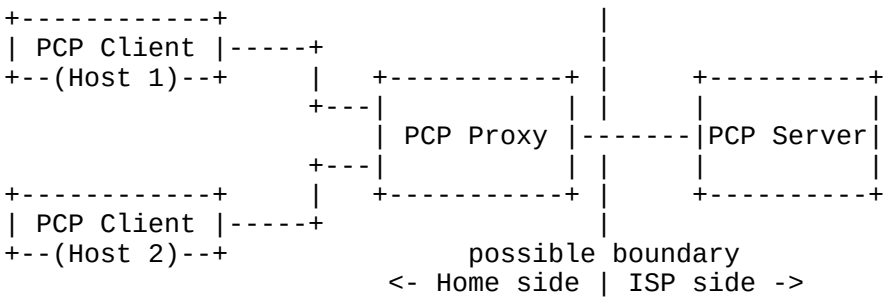

Figure 1: Reference Architecture

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].

2. PCP Server Discovery and Provisioning

The PCP Proxy MUST implement one of the discovery methods listed in [I-D.ietf-pcp-base] (e.g., DHCP [I-D.bpw-pcp-dhcp]).

The address of the PCP Proxy is provisioned to local PCP Clients as their default PCP Server: If the PCP DHCP option is supported by an

internal PCP Client, it will retrieve the PCP Server IP address to use from its local DHCP server (usually embedded on the CP router); otherwise internal PCP Clients will assume their default router being the PCP Server.

3. PCP Proxy as a PCP Server

The PCP Proxy acts as a PCP server for internal hosts and accepts PCP requests on the interface(s) facing them, e.g., it creates servicing socket(s) and bound them to each address of this (these) interface(s) on UDP port 44323.

When the topology makes a routing loop possible, the PCP Proxy MAY check it is not the source of a PCP message it’s received.

4. Control of the Firewall

A security policy to accept PCP messages from the provisioned PCP Server is to be enabled on the CP router. This policy can be for instance triggered by DHCP configuration or by outbound PCP requests issued from the PCP Proxy to the provisioned PCP Server.

In order to accept inbound and outbound traffic associated with PCP mappings instantiated in the upstream PCP Server, appropriate security policies are to be configured on the firewall.

For instance if the firewall rules have a lifetime, PCP response can be snooped in order to instantiate the corresponding firewall rules with the same lifetime. If they have no lifetime, an explicit dynamic mapping table can be kept in the PCP Proxy state in order to instantiate and remove corresponding firewall rules. This is in fact an easy subcase of Section 5.

REMOTE_PEER_FILTER Options can be installed into the local firewall, forwarded to the PCP Server so installed into the remote NAT/firewall or both.

> [Ed. Note: should we say the firewall function is already handled by the PCP controlled device so it is useless at the local level?]

5. Embedded NAT in the CP Router

When no NAT is embedded in the CP router, the port number included in received PCP messages (from the PCP Server or PCP Client(s)) are not altered by the PCP Proxy.

[Ed. Note: NAT444 seems to be the only exception?]

When the PCP Proxy is co-located with a NAT function in the CP router, it MUST update the content of received requested messages with the mapped port number and the address belonging to the external interface of the CP router (i.e., after the NAT operation) and not as initially positioned by the PCP Client. For the reverse path, PCP response messages MUST be updated by the PCP Proxy to replace the target port number to what has been initially positioned by the PCP Client. For this purpose the PCP Proxy has an access to the local NAT state. Note PCP messages with an unknown OpCode or Option can carry a hidden target address or internal port which will not be translated:

- a PCP Proxy co-located with a NAT SHOULD reject by an UNSUPP_OPCODE error response a received request with an unknown OpCode;

- a PCP Proxy co-located with a NAT SHOULD reject by an UNSUPP_OPTION error response a received request with a mandatory-to-process unknown Option;

- a PCP Proxy co-located with a NAT SHOULD remove any optional-to-process unknown Options from received requests before forwarding them.

When a PCP request is received and accepted by the PCP Proxy the corresponding mapping (explicit dynamic mapping for a MAP request, implicit dynamic mapping for a PEER request) is looked for in the local NAT state and temporary created if it does not exist. Temporary means it is deleted if no SUCCESS response is received, either explicitly or because of its short lifetime at creation.

If the local NAT associates explicit dynamic mappings to a lifetime, the requested lifetime in MAP requests SHOULD be adjusted to be in the accepted range of the local NAT, and the assigned lifetime copied from MAP responses to the corresponding mapping in the local NAT. The same processing applies to implicit dynamic mappings and PEER requests/responses (but the valid requested lifetime range begins by zero in this case).

Otherwise explicit dynamic mappings have an undefined lifetime in the local NAT and the PCP Proxy SHOULD maintain an explicit dynamic mapping table and SHOULD delete corresponding explicit dynamic mappings in the local NAT when they expire or are deleted by the MAP request with a zero requested lifetime.

6. Simple PCP Proxy

A simple PCP Proxy performs minimal modifications to PCP requests and responses, in particular it does not change the Epoch value in responses. So it does not handle more than one PCP server.

The detailed behavior at the reception of a PCP request on an internal interface is as follows:

- check if the source IP address and the PCP target address are the same.
- apply security controls, including with the result of the previous item.
- if the request is rejected, build a synthetic error response and send it back to the PCP client.
- if the request is accepted, adjust it (e.g., adding a THIRD_PARTY Option, updating the internal address and port to their translated values as specified in Section 5 and forward it on a fresh UDP socket connected to the PCP server.
- Wait for the response during a reasonable delay.
- when the response is received from the PCP server, adjust it back (e.g., removing the THIRD_PARTY Option added previously, updating the internal address and port to their initial values as specified in Section 5), forward it to the source PCP client and close the socket to the PCP server.

  [Ed. Note: is there extra validation useful? The response comes from the PCP server and the PCP client will validated it anyway.]

- on a hard error on the UDP socket, build a synthetic ICMP error and send it to the source PCP client.

The reasonable delay minimum value is 20 seconds, request retransmission is handled by PCP clients.

For each pending request, the proxy MUST maintain in a data record:

- the request payload
- the interface where the request was received

- the source IP address of the request
- the source UDP port of the request
- the UDP socket connected to the PCP server
- an expire timeout

Receiving interfaces can be implemented by a set of servicing sockets, each socket bound to an address of an internal interface. Interface, source address and port are used to send back packets to the source PCP client. The request payload is used to generate synthetic ICMP. Responses are received on the UDP socket.

There is no (not yet) standardized way to build a synthetic error response, in particular no way to determine which Epoch value to put into it. This is why it is better to build a synthetic ICMP error than a synthetic error response with NETWORK_FAILURE on a socket hard error.

Too large requests SHOULD be forwarded to the PCP server in order to relay back the error response, i.e., the PCP Proxy is not in charge to enforce the message size limit and in general the PCP Proxy SHOULD NOT generate error response for a reason other than security controls. No behavior is specified in the case the PCP Proxy processing (e.g., adding a THIRD_PARTY Option) makes a valid request too large when it is sent to the PCP Server.

7. Smart Proxy

When a simple PCP Proxy uses as global variables only the CGN PCP server IP address, a set of servicing sockets and a list of pending request handlers, a smart PCP Proxy implements more services.

Even if most services rely on the Epoch handling one Section 7.2, services are described below in a natural order.

7.1. Multiple PCP Servers

A smart PCP Proxy MAY offer to handle multiple PCP servers at the same time, each PCP server is associated to each own handled Epoch value according to Section 7.2.

The only constraint is to maintain a reasonable coherency as PCP clients cannot be assumed to be prepared to this, i.e., this has to be transparent for / hidden to them.

[Ed. Note: we propose to require a partition of clients, clients on the same host or sharing a target address SHOULD be in the same subset, i.e., the same PCP server and the same Epoch.]

[Ed. Note: the Proxy can get per PCP server capabilities, for instance from the error responses.]

7.2. Epoch Handling

With Epoch handling the Epoch value is related to internal timers and not blindly copied from PCP responses. There should be no advantages to have more than one managed Epoch per PCP server.

The Epoch MUST be reset when explicit dynamic mappings are lost, i.e.:

- at startup if the PCP proxy can’t recover the state.

  [Ed. Note: as it is very optional to manage state in the Proxy it should be the default.]

- when the WAN address is changed or any similar events which show any previous state is no longer valid.

- when the Epoch value in a PCP response is too small (cf. Epoch value validation rules in [I-D.ietf-pcp-base]).

- when the External Address has changed

The last two rules are per PCP server, a PCP Proxy MAY check these conditions in all received responses for a PCP server, including when the PCP Proxy is a part of an IWF [I-D.bpw-pcp-upnp-igd-interworking] [I-D.bpw-pcp-nat-pmp-interworking].

7.3. Request/Response Caching

A PCP Proxy providing request/response caching checks each time it receives a PCP request if it has already seen the same request recently and got the corresponding PCP response. In this case, it sends back directly the cached response with the proper Epoch value and not forward the request to the PCP server.

[Ed. Note: this is an easy optimization, the only difficult point can be solved by the Epoch handling.]

7.4. Retransmission Handling

An extension of the previous service is to manage the retransmission of pending requests to the server internally, i.e., no longer driven by the PCP client. A cache entry SHOULD be expired after a delay short enough to keep it easy to distinguish it from a replay.

> [Ed. Note: this allows smart retransmission scheduling as the Proxy "sees" all PCP exchanges with the PCP server.]

7.5. Full State

A smart PCP Proxy can keep the full state: an image of all active explicit dynamic mappings is kept in memory. This service is not interesting by itself but it can be necessary to support embedded firewall or NAT Section 5 and if the PCP Proxy is integrated in an IWF (e.g., to support UPnP IGD [I-D.bpw-pcp-upnp-igd-interworking]).

In conclusion this service MAY be supported. Note when it is supported the state SHOULD be recovered in case of failures according to [I-D.boucadair-pcp-failure].

8. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

9. Security Considerations

The security controls are applied on PCP requests and are about:

- authorized target addresses, in particular in case of a third party.
- authorized internal and external ports (note the external port is in general assigned by the CGN PCP server).

The default policy for requests for a third party when such a policy exists is be to not allow them. The exact rule is: PCP requests including a THIRD_PARTY option enclosing an IP address distinct than the source IP address of the request MUST be rejected (by a NOT_AUTHORIZED error response).

When a PCP Proxy is at the boundary of two trust domains (named

"internal" and "external" sides), it MUST provide at least these two security controls:

- split horizon anti-spoofing: requests from the external side and responses from the internal side MUST be dropped.
- a policy about requests on the behalf of a third party MUST be enforced.

A PCP Proxy MAY implement only the simple rule about third party: all received requests including a THIRD_PARTY option are rejected.

> [Ed. Note: this is stricter than the default but keeps the minimal implementation as simple as possible.]

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 a priori compatible with security controls or correct processing. This includes at least all cases where received requests are scanned for elements like the protocol, an address or a port.

> [Ed. Note: magically a minimal implementation in favorable environments (no embedded NAT!) MAY accept unknown Opcodes and Options. There is no need for a similar rule for responses as the proxy can do nothing with a "bad" response anyway...]

10. References

10.1. Normative References

[I-D.bpw-pcp-dhcp] Boucadair, M., Penno, R., and D. Wing, "DHCP and DHCPv6 Options for the Port Control Protocol (PCP)", draft-bpw-pcp-dhcp-04 (work in progress), April 2011.

[I-D.ietf-pcp-base] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", draft-ietf-pcp-base-13 (work in progress), July 2011.

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

10.2. Informative References

[I-D.boucadair-pcp-failure]
Boucadair, M., Dupont, F., and R. Penno, "Port Control Protocol (PCP) Failure Scenarios", draft-boucadair-pcp-failure-01 (work in progress), March 2011.

[I-D.bpw-pcp-nat-pmp-interworking]
Boucadair, M., Penno, R., Wing, D., and F. Dupont, "Port Control Protocol (PCP) NAT-PMP Interworking Function", draft-bpw-pcp-nat-pmp-interworking-00 (work in progress), March 2011.

[I-D.bpw-pcp-upnp-igd-interworking]
Boucadair, M., Penno, R., Wing, D., and F. Dupont, "Universal Plug and Play (UPnP) Internet Gateway Device (IGD)-Port Control Protocol (PCP) Interworking Function", draft-bpw-pcp-upnp-igd-interworking-02 (work in progress), February 2011.

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, August 2011.

Authors' Addresses

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: mohamed.boucadair@orange-ftgroup.com

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

Email: rpenno@juniper.net

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

Email: dwing@cisco.com

Francis Dupont
Internet Systems Consortium

Email: fdupont@isc.org