Network Working Group   S. Perreault, Ed.
Internet-Draft   Viagenie
Intended status: Standards Track   M. Boucadair
Expires: August 02, 2014   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.

Table of Contents

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.

```
                        .................
            +------+       : +------+------+ :     +------+
            |Client|-------:-|Server|Client|-:----|Server|
            +------+       : +------+------+ :     +------+
                        :       Proxy     :
                        .................
```

                    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 on
   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.

```
                  |
        +------+  |
        |Client|--+
        +------+  |  +-----+                          +------+
                  +--|Proxy|--------<ISP network>----------|Server|
        +------+  |  +-----+                          +------+
        |Client|--+    CPE
        +------+  |
                  |
               LAN
```
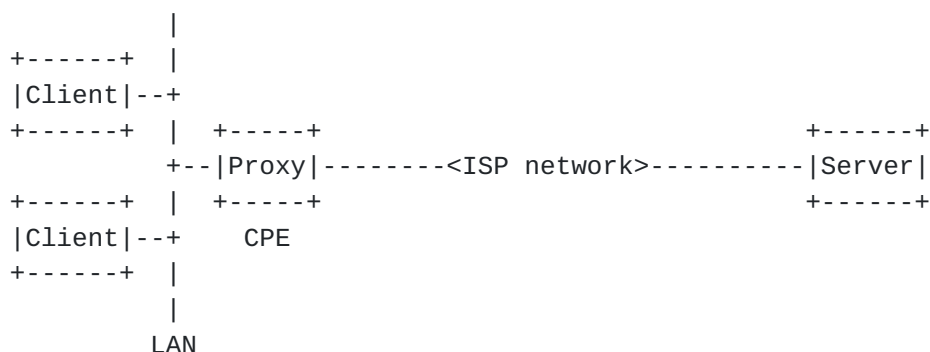
                  Figure 2: PCP Relay Use Case

   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 provioning 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