Network Working Group
Internet-Draft
Intended status: Experimental
Expires: January 1, 2016

M. Boucadair
C. Jacquenet
France Telecom
June 30, 2015

Discovering the Capabilities of Flow-Aware Service Functions (a.k.a. Middleboxes): A PCP-based Approach
draft-boucadair-hops-capability-discovery-00

Abstract

This document specifies a solution to discover the capabilities of a flow-aware service function. The solution relies upon the use of the Port Control Protocol (PCP).

This solution allows for applications to anticipate connectivity failures and to proceed with countermeasures (e.g., create a mapping for incoming connections, discover a mapping lifetime, discover an external IP address, avoid injecting some options in the outgoing packets, etc.). The propsoed approach allows, for example, to discover whether an upstream flow-aware service function is MPTCP-friendly (that is, it does not strip MPTCP signals) or SCTP-compliant, whether it embeds a firewall function, etc. or a combination thereof.

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 January 1, 2016.

Table of Contents

1. Introduction

Advanced service functions (e.g., Performance Enhancement Proxies ([RFC3135]), NATs [RFC3022][RFC6333][RFC6146], firewalls [I-D.ietf-opsawg-firewalls], etc.) are required to achieve various objectives such as IP address sharing, firewalling, to avoid covert channels, to detect and protect against DDoS attacks, etc.

Removing those functions is not an option because they are used to address constraints that are often typical of the current yet protean Internet situation (global IPv4 address depletion comes to mind, but also the plethora of services with different QoS/security/robustness requirements, etc.), and this is even exacerbated by environment-specific designs (e.g., the nature and the number of service functions that need to be invoked at the Gi interface of a mobile infrastructure).

Moreover, these sophisticated service functions are located in the network but also in service platforms, or other structures like

Content Delivery Networks. Some of these service functions can be controlled by hosts (e.g., NAT) to avoid connectivity complications ([RFC6269]) while others are hidden to customers.

This document proposes a solution that can be used by hosts to discover the capabilities of flow-aware service functions that are visible to them but it can also be used by an administrator responsible for the management of such (hidden) service functions, e.g., to inform an SFC controller ([I-D.ww-sfc-control-plane]) about the nature and the status of these service functions. Obviously, exposing this information to hosts/applications is deployment-specific.

Customer-facing flow-aware service functions can announce their capabilities to hosts. This information can be used by a host to select a service function instance (e.g., include the external address of that service function in a referral will involve that service function in the communication path). For example, a host that discovers that the Residential Gateway it is connected to does not support Stream Control Transmission Protocol (SCTP, [RFC4960]), won't even try to use SCTP as a transport protocol; TCP/SCTP happy eyeball proposals are useless in such case.

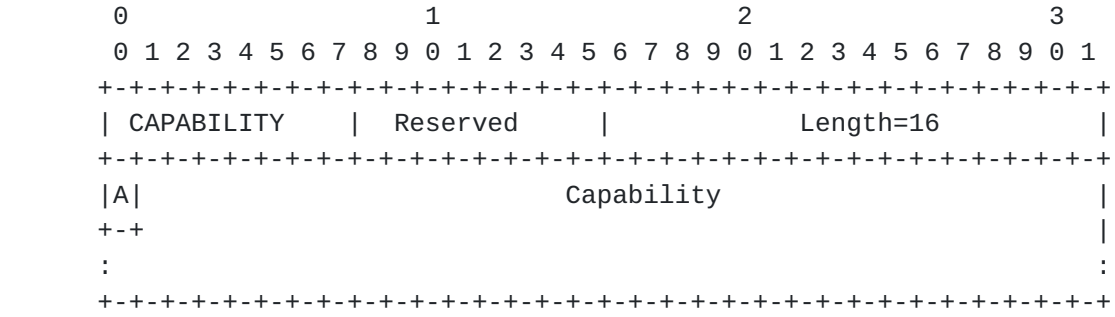
This document extends the base PCP [RFC6887] with a new option, called CAPABILITY (Section 2), to discover the capabilities of one or several service functions typically embedded in middleboxes. Retrieving the capabilities of these middleboxes is meant to facilitate fault management (e.g., provide a hint about why some applications fail, help select the required actions to instruct the middlebox to handle incoming connections, etc.). This option, when received from a PCP server, is used by a host (and the PCP client) to better adapt the traffic it may send according to the perceived network conditions as exposed in the PCP option (including tweaking PCP requests to instruct mappings).

This specification can also be used to help the introduction of new transport protocols. For example, CPE devices managed by a service provider can include this feature. Also, a service provider that Introduces additional service nodes that support new features (e.g., SCTP-aware CGN) in the network can select the set of CPEs that will be serviced by these nodes. It can do so by setting the SCTP bit when sending the capability information to the selected CPEs. Additional sample use cases are discussed in Section 5.

2. PCP CAPABILITY Option

The CAPABILITY option (Code: TBA, Figure 1) is used by a flow-aware service function to explicitly inform a host about the capabilities that pertain to the said flow-aware service function, especially as far as IP forwarding operations are concerned.

One single CAPABILITY option is conveyed in the same PCP message even if several functions are co-located in the same device (e.g., NAT44 and NAT64, NAT44 and port set assignment capability, etc.).



This Option:

Option Name: PCP Capabilities option (CAPABILITY)
Number: TBA (IANA)
Purpose: Retrieve the capabilities of a PCP-controlled device
Valid for Opcodes: ANNOUNCE, MAP, PEER
Length: 16
May appear in: both request and response
Maximum occurrences: 1

Figure 1: Capability option

When set, the A bit indicates the PCP server supports authentication ([I-D.ietf-pcp-authentication]). If this bit is unset, it indicates that plain PCP is supported.

The Capability Field is encoded in 127 bits. Each bit in the Capability bit mask is used to represent a PCP-controlled device capability. Whenever a bit of the Capability Field is set, this means that the corresponding capability is enabled/supported. Several bits can be set simultaneously if several functions are co-located. The default value for each capability bit is '0'. The meaning associated with the following Capability bits is (other values can be added to the list):

Bit #: Description
1: NAT44 [RFC3022]

2: Stateless NAT64 [RFC6145].
4: Stateful NAT64 [RFC6146].
5 : Dual-Stack Lite (DS-Lite) AFTR [RFC6333]
6: Dual-Stack Extra Lite [RFC6619]
8: A+P Port Range Router (PRR) [RFC6346]
9: Supports PORT_SET option [I-D.ietf-pcp-port-set].
16: IPv4 firewall.
32: IPv6 Firewall [RFC6092].
64: IPv6-to-IPv6 Network Prefix Translation (NPTv6) [RFC6296].
119: TCP [RFC0793].
120: User Datagram Protocol (UDP) [RFC0768].
121: UDP-Lite compliant [RFC3828]
122: Datagram Congestion Control Protocol (DCCP) [RFC4340]
123: SCTP [RFC4960]
124: Multipath TCP (MPTCP) [RFC6824].
125: DSCP re-marking function.
126: FLOWDATA-aware function ([I-D.wing-pcp-flowdata]).
127: ILNP Translator [RFC6740].

3. PCP Client & Host Behavior

The PCP client includes a CAPABILITY option in a MAP or ANNOUNCE request to learn the capabilities of an upstream PCP-controlled device. When conveyed in a PCP request, the Capability field MUST be set to 0. The CAPABILITY option can be inserted in a MAP request that is used to learn the external IP address, as detailed in Section 11.6 of [RFC6887].

The PCP client MUST be prepared to receive multiple CAPABILITY options (e.g., if several PCP servers are deployed and each of them is configured with a distinct set of capabilities). The PCP client MUST associate each received set of capabilities and suffix with the PCP server from which the information was retrieved.

Upon receipt of an unsolicited PCP ANNOUNCE message, the PCP client replaces the former set of capabilities as received from the same PCP server with the new set of capabilities, as indicated in the CAPABILITY option.

Based on the received capabilities, the host/application/PCP client may decide to tune its requests (e.g., Section 5). For example, a PCP client can use the returned information to decide whether all PCP servers need to be contacted in parallel or only a subset of them , or which service function to solicit in order to establish some sessions (e.g., SCTP).

4. PCP Server Behavior

Activating this feature on the PCP server is subject to administrative authorization procedures.

The PCP server that controls a flow-aware service function SHOULD be configured to provide requesting PCP clients with the supported capabilities whose corresponding bit in the CAPABILITY option will therefore be set. When enabled, the CAPABILITY option conveys the set of capabilities supported by the PCP-controlled device.

If the PCP server is configured to honor the CAPABILITY option but has no means to determine the set of capabilities supported by the local device, the PCP server MUST NOT include any CAPABILITY option in its PCP messages.

The PCP server MAY be configured to include a CAPABILITY option in all MAP responses, even if the CAPABILITY option is not listed in the associated request. The PCP server MAY be configured to include a CAPABILITY option in its ANNOUNCE messages.

In the event of any change of the capabilities supported by the PCP-controlled device (e.g., the activation of a new service function), the PCP server SHOULD issue an unsolicited PCP ANNOUNCE message to inform the PCP client about the updated set of capabilities.

Upon receipt of a PCP request from a PCP client that requires the PCP server to proceed with an operation that is beyond its capabilities, the PCP server MAY return an error code together with the CAPABILITY option.

When a new PCP server joins the network, it MAY then send an ANNOUNCE Opcode with its capabilities (i.e., CAPABILITY option).

5. Sample Use Cases

Below is provided a non-exhaustive list of use cases to illustrate the benefits of the proposed solution:

- A middlebox may be configured to strip MPTCP options or to let them pass without any modification. Explicitly advertising such capability to the hosts will avoid extra delays to establish successful TCP sessions. In reference to Figure 2, the host won't use MPTCP because the firewall it is connected to does not support MPTCP. This information is useful for the application since it can use the TCP option space more efficiently, so as to insert options that couldn't be inserted if MPTCP options were included.

Figure 2: MPTCP Example

- A host that supports both TCP and SCTP can decide which transport to use when establishing transport sessions. For example, an application that is designed to be transported over TCP or SCTP can avoid sending SCTP packets if an upstream device in the path announces that it is not compliant with SCTP. SCTP can be used if that upstream device announces it supports SCTP. Furthermore, if that upstream device is also a NAT, appropriate (SCTP) explicit dynamic mappings can be instantiated by the application so that incoming connections can be forwarded appropriately. Figure 3 shows an example of two NAT devices; one of them supports SCTP. Owing to the CAPABILITY option, SCTP sessions can be forced by the host to cross the SCTP-enabled NAT by including for instance the external IP address (@IP_Ext2) in a referral).

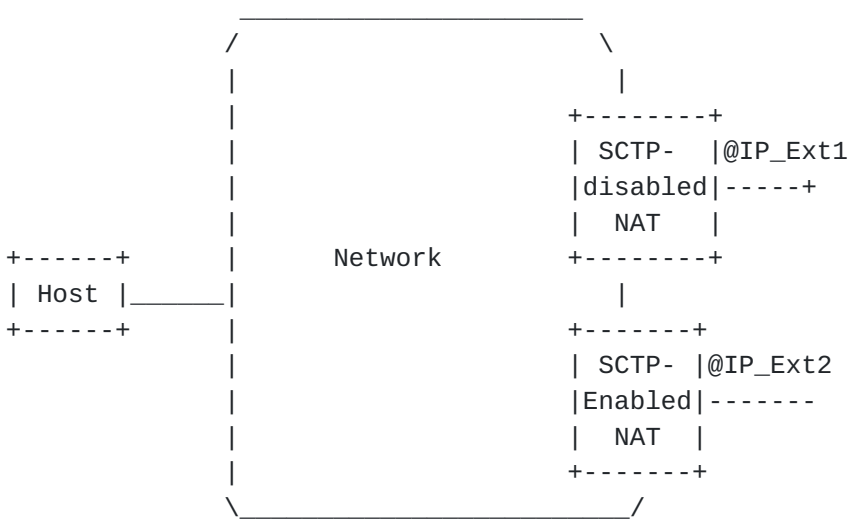


Figure 3: SCTP Example

- In an IPv6 network that runs NPTv6 [RFC6296] functions, firewalls controlled by a PCP server are embedded in different devices: the PCP client learns about the available PCP servers by means of DHCP [RFC7291] or any other PCP server discovery technique. The PCP client learns about the PCP server capabilities by using the CAPABILITY option. The PCP client sends MAP PCP request to a PCP-controlled NPTv6 device with Internal Port=0 and Protocol=0 (which means 'all ports for all protocols') to find the external IP

address. This PCP request has to be sent only once since NPTv6 is stateless and provides a 1:1 relationship between addresses that belong to the "inside" and "outside" prefixes, respectively. The PCP client will send PCP requests only to the PCP server that controls the NPTv6 device before the Assigned Lifetime of the MAP response expires or when the host that embeds the PCP client acquires a new IPv6 address that belongs to the "inside" prefix. However, the PCP client will send MAP/PEER requests to the PCP server that controls the firewall device to create/delete dynamic outbound mappings, or use PCP instead of its default application keep-alives to maintain the firewall-maintained states alive.

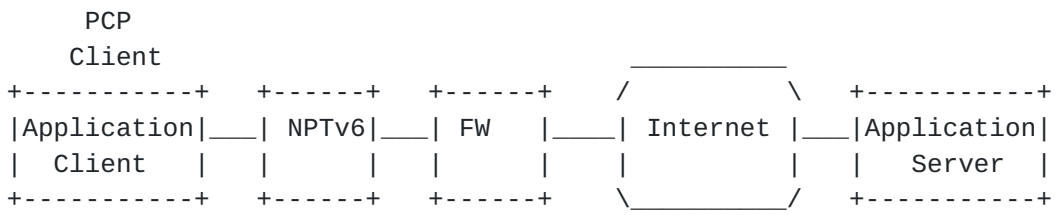


Figure 4: NPTv6 and Firewall not collocated with PCP server Capability

- In a network that embeds NAT64 [RFC6146] devices, the PCP-controlled firewall service functions are embedded in different devices: The IPv6-only PCP client can send the PREFIX64 PCP option [RFC7225] only to the PCP-controlled NAT64 device to learn the Prefix64(s) used to build IPv4-embedded IPv6 addresses.

- Multiple PCP-controlled devices: See Figure 5 the example of a network deploying several techniques to connect with the IPv4 Internet, to provide IPv6-only connectivity, etc. The discovered capabilities can be used to trigger the selection of the appropriate PCP server [RFC7488].

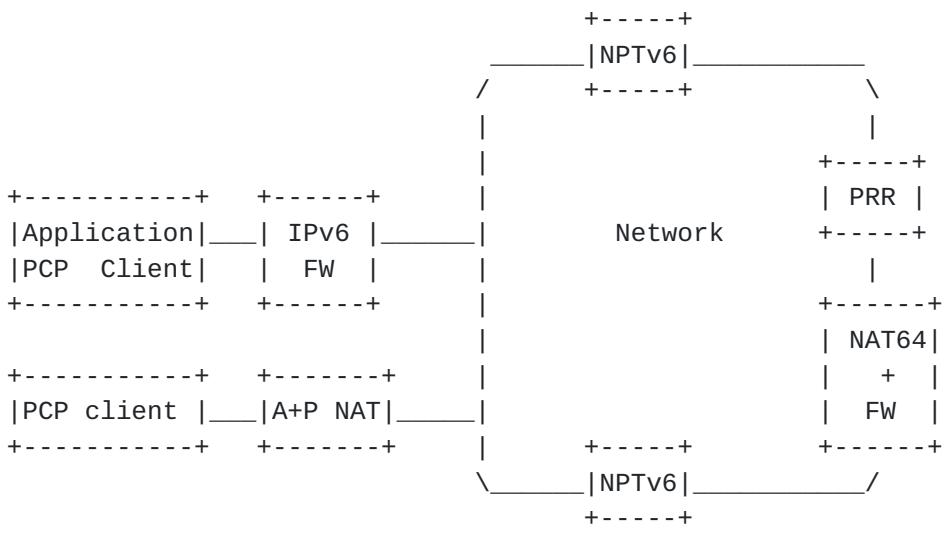


Figure 5: Multiple PCP-controlled devices

- In a IPv6 network that supports a PCP-controlled ILNP translator [RFC6740], the PCP-controlled firewall service functions are embedded in different devices. The PCP client needs to send PCP requests only to the PCP-controlled ILNP translator to find Global Locators associated with Internal Locators.

- When the PCP-controlled device is a Port Range Router (PRR, see Section 3.2 of [RFC6346]), the PCP client should use the PORT_SET [I-D.ietf-pcp-port-set] option.

6. Security Considerations

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

An attacker may generate a PCP message with a fake CAPABILITY option to switch off some features that would have been used by a host. Means to authenticate the PCP server SHOULD be supported.

7. IANA Considerations

The following PCP option Code is to be allocated in the optional-to-process range (the registry is maintained in http://www.iana.org/assignments/pcp-parameters/pcp-parameters.xml#options):

CAPABILITY

A sub-registry is required to track the set of capabilities of PCP-controlled devices.

8. References

8.1. Normative References

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

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

8.2. Informative References

[I-D.ietf-opsawg-firewalls] Baker, F. and P. Hoffman, "On Firewalls in Internet Security", draft-ietf-opsawg-firewalls-01 (work in progress), October 2012.

[I-D.ietf-pcp-authentication] Wasserman, M., Hartman, S., Zhang, D., and T. Reddy, "Port Control Protocol (PCP) Authentication Mechanism", draft-ietf-pcp-authentication-12 (work in progress), June 2015.

[I-D.ietf-pcp-port-set] Qiong, Q., Boucadair, M., Sivakumar, S., Zhou, C., Tsou, T., and S. Perreault, "Port Control Protocol (PCP) Extension for Port Set Allocation", draft-ietf-pcp-port-set-09 (work in progress), May 2015.

[I-D.wing-pcp-flowdata] Wing, D., Penno, R., and T. Reddy, "PCP Flowdata Option", draft-wing-pcp-flowdata-00 (work in progress), July 2013.

[I-D.ww-sfc-control-plane] Li, H., Wu, Q., Boucadair, M., Jacquenet, C., Haeffner, W., Lee, S., Parker, R., Dunbar, L., Malis, A., Halpern, J., Reddy, T., and P. Patil, "Service Function Chaining (SFC) Control Plane Components & Requirements", draft-ww-sfc-control-plane-06 (work in progress), June 2015.

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768, August 1980.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.

[RFC3135] Border, J., Kojo, M., Griner, J., Montenegro, G., and Z. Shelby, "Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations", RFC 3135, June 2001.

[RFC3828] Larzon, L-A., Degermark, M., Pink, S., Jonsson, L-E., and G. Fairhurst, "The Lightweight User Datagram Protocol (UDP-Lite)", RFC 3828, July 2004.

[RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", RFC 4340, March 2006.

[RFC4960] Stewart, R., "Stream Control Transmission Protocol", RFC 4960, September 2007.

[RFC6092] Woodyatt, J., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092, January 2011.

[RFC6145] Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", RFC 6145, April 2011.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, April 2011.

[RFC6269] Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", RFC 6269, June 2011.

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

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

[RFC6346] Bush, R., "The Address plus Port (A+P) Approach to the IPv4 Address Shortage", RFC 6346, August 2011.

[RFC6619] Arkko, J., Eggert, L., and M. Townsley, "Scalable Operation of Address Translators with Per-Interface Bindings", RFC 6619, June 2012.

[RFC6740] Atkinson,, RJ., "Identifier-Locator Network Protocol (ILNP) Architectural Description", RFC 6740, November 2012.

[RFC6824] Ford, A., Raiciu, C., Handley, M., and O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses", RFC 6824, January 2013.

[RFC7225] Boucadair, M., "Discovering NAT64 IPv6 Prefixes Using the Port Control Protocol (PCP)", RFC 7225, May 2014.

[RFC7291] Boucadair, M., Penno, R., and D. Wing, "DHCP Options for the Port Control Protocol (PCP)", RFC 7291, July 2014.

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

Authors' Addresses

Mohamed Boucadair
France Telecom
Rennes 35000
France

Email: mohamed.boucadair@orange.com

Christian Jacquenet
France Telecom
Rennes
France

Email: christian.jacquenet@orange.com