Network Working Group
Internet-Draft
Intended status: Experimental
Expires: December 29, 2012

M. Wasserman
S. Hartman
Painless Security
D. Zhang
Huawei
June 27, 2012

# Port Control Protocol (PCP) Authentication Mechanism
# draft-ietf-pcp-authentication-00.txt

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.

Table of Contents

## 1. Introduction

Using the Port Control Protocol (PCP) [I-D.ietf-pcp-base], 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 following issues are considered in the design of this extension:

- Loss of EAP messages during transportation
- Disordered delivery of EAP messages
- Generation of transport keys
- Integrity protection and data origin authentication for PCP messages
- Algorithm agility

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

- On security infrastructure equipment, such as corporate firewalls, that does not create implicit mappings.
- 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.
- For any implementation that wants to be more permissive in authorizing explicit mappings than it is in authorizing implicit mappings.
- For implementations that support the THIRD_PARTY Option (unless they can meet the constraints outlined in Section 14.1.2.2).
- 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 [I-D.ietf-pcp-base].

PCP Client (PCC): A PCP device (e.g., a host) which is responsible for issuing PCP requests to a PCP server. In this document, a PCC is also a EAP peer [RFC3748], and it is the responsibility of a PCC to provide the credentials when authentication is required.

PCP Server (PCS): A PCP device (e.g., a NAT or a firewall) that implements the server-side of the PCP protocol, via which PCCs request and manage explicit mappings. In this document, a PCS is integrated with an EAP authenticator [RFC3748]. Therefore, when necessary, a PCS can verify the credentials provided by a PCC and make an access control decision based on the authentication result.

PCP Authentication (PA) Session: A series of PCP message exchanges transferred between a PCC and a PCS in order to perform authentication, authorization, key distribution and secured PCP communication. Each PA session is assigned a distinctive Session ID. The PCP devices involved within a PA session are called session partners. A typical PA session has two session partners.

Session Lifetime: The life period associated with a PA session, which decided the lifetime of the current authorization given to the PCC.

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

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

PA (PCP for Authentication) message: A PCP message containing an Authentication OpCode for EAP authentication.

non-PA message: A PCP message which is not a PA message.

## 3. Separate vs. Inline Key Management

There is an open question in the working group regarding what approach should be used for PCP key management. The precursor to this document originally proposed an inline key management approach using EAP directly over PCP. There was an alternative proposal on the list to standardize a separate key management approach using PANA [RFC5191] (with EAP). The WG will need to make a decision between these two approaches before this document can be completed.

Both approaches for key management could be used with the integrity protection mechanism and options described later in this document.

## 4. Separate Key Management

The separate key management proposal involves running PANA between the end-points to dynamically generate a security association, and then using that security association to authenticate PCP message exchanges.

For this approach we would define an AVP for PANA to indicate that the PANA session was being used for PCP authentication, not for network access purposes.

A PANA server would be implemented on each PCP server that support authenticated requests, or another mechanism would need to be specified to locate a PANA server that can be used for PCP-related PANA requests. It may be possible to define a subset of the PANA protocol that can be run on PCP Servers if the same PANA server will not be used for network access. For example, it would not be necessary for these servers to support IP Address Reconfiguration.

Once a secure session has been established using PANA, the Secure OpCode option described in this draft could be used to associate PCP requests with a particular PANA session. Some discussion may be needed on how the PCP session will be securely bound to the PANA session initiation.

Although a separate key management approach using PANA has been discussed on the PCP mailing list, this approach would require further documentation if the WG decides to pursue it.

## 5. Inline Key Management

The inline key management approach is described in this document in the sections Section 6.1 and Section 6.2.

## 6. Protocol Details

### 6.1. Session Initiation

To carry out an EAP authentication process between two PCP devices, a set of PA messages need to be exchanged. A PA message contains an Authentication OpCode and associated Options. The Authentication OpCode consists of three fields: Session ID, Flag, and Sequence Number. The Session ID field is used to identify the session to which the message belongs. The Flag field indicates the type of the PCP message. The sequence number field is used to detect the disorder or the duplication occurred during packet delivery.

The message exchanges conveyed within an PA session is introduced in the remainder section.

When a PCC intends to initiate a PA session with a PCS, it sends a PCC-Initiation message to the PCS. In the message, the Session ID and Sequence Number fields of the Authentication OpCode are set as 0; the I bit is set. The PCC-Initiation message is also attached with a nonce option which consists of a random nonce selected by the PCC to tolerate off-line attacks. After receiving the PCC-Initiation, if the PCS would like to initiate a PA session, it will reply with a PA-Request which contains an EAP Identity Request. The Sequence Number field in the PA-Request is set as 0, and the Session ID field MUST be filled with the session identifier assigned by the PCS for this session. The PA-Request also needs to be attached with a nonce option which is learned from the PCC. Form now on, every PA message within this session must be attached with the session identifier. When receiving a PA message from an unknown session, a PCP device MUST discard the message silently. If the PCC intends to simplify the authentication process, it can append an EAP Identity Response message within the PCC-Initiation request so as to inform the PCS that it would like to perform EAP authentication andskip over the step of waiting for the EAP Identity Request.

In the scenario where a PCS receives a non-PA PCP message from a PCC which needs to be authenticated, the PCS can reply with a PA-Request to initiate a PA session; the result code field of the PA-Request is set as AUTHENTICATION-REQUIRED. In addition, the PCS MUST assign a session ID for the session and transfer it within the PA-Request. In the PA messages exchanged afterwards in this session, the session ID MUST be appended. Therefore, in the subsequent communication, the PCC can distinguish the messages in this session from those in other sessions through the PCS IP address and the session ID. When the PCC receives the initial PA-Request message from the PCS, it can reply with a PA-Answer message to continue the session or silently discards the request message according to its local policies.

In a PA session, PA-Request messages are sent from PCSs to PCCs while PA-Answer messages are only sent from PCCs to PCSs. Correspondently, an EAP request messages MUST be transported within a PA-Request message, and an EAP answer messages MUST be transported within a PA-Answer message. Particularly, when a PCP device receives a PA-Request or a PA-Answer message from its partner, the PCP device needs to reply with a PA-Acknowledge message to indicate that the message has been received. This solution is used to deal with the conditions where the device cannot generate a response within a pre-specified period due to certain reasons (e.g., waiting for human input to construct a EAP message). Therefore, the partner does not have to un-necessarily retransfer the PCP message.

In this work, it is mandated for a PCC and a PCS 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 PCC and the PCS 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. On this occasion, the PCS needs to append the initial PA-Request message with a set of PRF Options and MAC Algorithm Options. Each PRF Option contains a PRF that the PCS supports. Similarly, each MAC Algorithm Option contains a MAC (Message Authentication Code) algorithm that the PCS supports. After receiving the request, the PCC selects a PRF and a MAC algorithm which it would like to use, and sends back a PA-Answer with a PRF Option and a MAC Algorithm Option for the selected algorithms.

The last PA-Request message transported within a PA session carries the EAP authentication and PCP authorization results. The last PA-Request and PA-Answer messages MUST have their the 'C' (Complete) bit set.

If the EAP authentication succeeds, the result code of the last PA-Request is AUTHENTICATION-SUCCESS. In this case, before sending out the PA-Request, the PCS must derive a transport key and use it to generate digests to protect the integrity and authenticity of the PA-Request and any subsequent PCP message. Such digests are transported within Authentication Tag Options. In addition, the PA-Request needs to be appended with a Session Lifetime Option which indicates the life time of the PA session (i.e., the life time of the MSK).

If the EAP authentication fails, the result code of the last PA-Request is AUTHENTICATION-FAILED. If the EAP authentication successes but Authorization fails, the result code of the last PA-Request is AUTHORIZATION-FAILED. In the latter two cases, the PA session MUST be terminated immediately after the last PCP

authentication message exchange.

### 6.2. Session Termination

A PA session can be explicitly terminated by sending a termination-indicating PA acknowledge message from either session partner. After receiving a termination-indicating message from the session partner, a PCP device MUST response with a termination-indicating PA Acknowledge message and remove the PA SA immediately. When the session partner initiating the termination process receives the acknowledge message, it will remove the associated PA SA immediately.

## 7. PA Security Association

At the beginning a PA session, a session SHOULD generate a PA SA to maintain its state information during the session. The parameters of a PA SA is listed as follows:

- IP address and UDP port number of the PCC
- IP address and UDP port number of the PCS
- Session Identifier
- Sequence number for the next outgoing PCP message
- Sequence number for the next incoming PCP message
- Last outgoing message payload
- Retransmission interval
- MSK
- MAC algorithm: The algorithm that the transport key should use to generate digests for PCP messages.
- Pseudo-random function: The pseudo random function negotiated in the initial PA-Request and PA-Answer exchange for the transport key derivation
- Transport key: the key derived from the MSK to provide integrity protection and data origin authentication for the messages in the PA session. The life time of the transport key SHOULD be identical to the life time of the session.

Particularly, the transport key is computed in the following way:

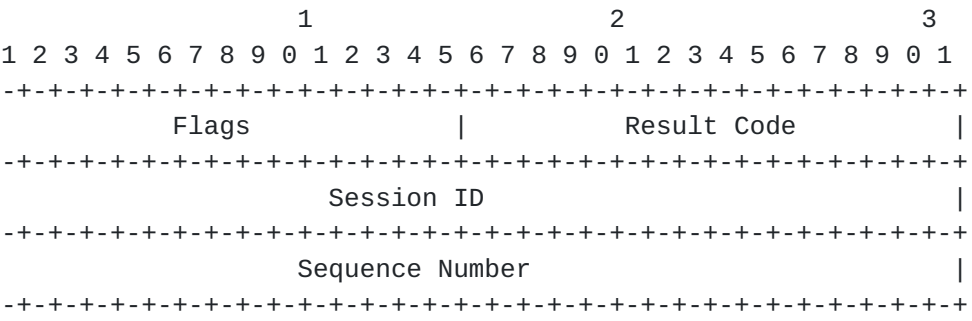
Transport key = prf(MSK, "IETF PCP"| Session_ID, key ID), where:

- The prf: The pseudo-random function assigned in the Pseudo-random function parameter.
- MSK: The master session key generated by the EAP method.
- "IETF PCP": The ASCII code representation of the non-NULL terminated string (excluding the double quotes around it).
- Session_ID: The ID of the session which the MSK is derived from
- Key ID: The ID assigned for the traffic key

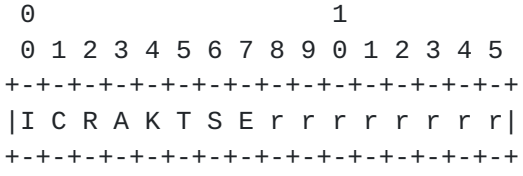
## 8. Packet Format

### 8.1. Authentication OpCode Format

The following figure illustrates the format of an authentication Opcode:




Flags: The Flags field is two octets. The following bits are assigned:



* I (Initiation): This bit is set in a PCC-Initiation message.

* C (Complete): If the message is the last PA-Request or PA-Answer message in the session, this bit MUST be set. For other messages, this bit MUST be cleared.

* R (Request): This bit is set in a PA-Request message.

* A (Answer): This bit is set in a PA-Answer message.

* K (acKnowledgement): This bit is set and only set in a PA-Acknowledgement message.

* T (Termination): If this bit is set in a PA-Acknowledgement message, the message is used for session-termination indication.

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

Result Code: This field is two octets.Following result code values are defined:

1 AUTHENTICATION-REQUIRED

2 AUTHENTICATION-FAILED

3 AUTHENTICATION-SUCCESS

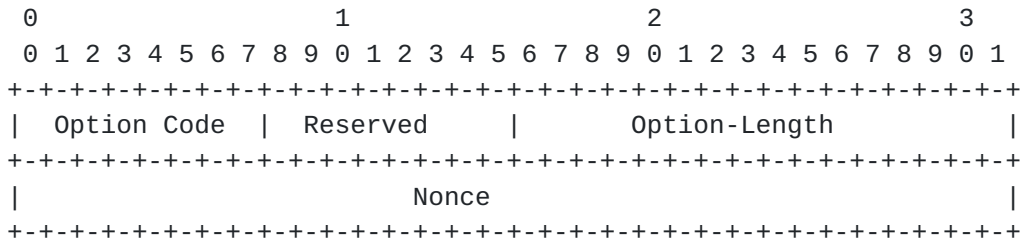
4 AUTHORIZATION-FAILED

## 8.2. Nonce Option

Question: Would it be possible to remove this option from the PCP authentication draft, and use the nonce from the main PCP header instead?

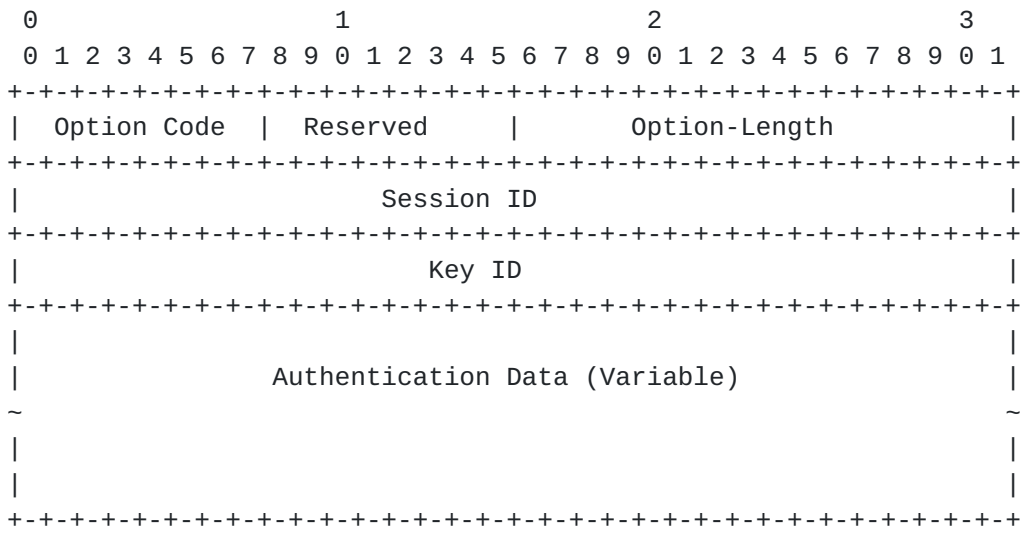
Because the session identifier of PA session is determined by the PCS, a PCS does not know the session identifier which will be used when it sends out a PCC-Initiation message. In order to prevent an attacker from interrupting the authentication process by sending off-line generated PA-Request messages, the PCS needs to generate a

random number as nonce in the PCC-Initiation message. The PCS will append the nonce within the initial PA-Request message. if the PA-Request message does not carry the correct nonce, the message will be discarded silently.




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

## 8.3. Authentication Tag Option



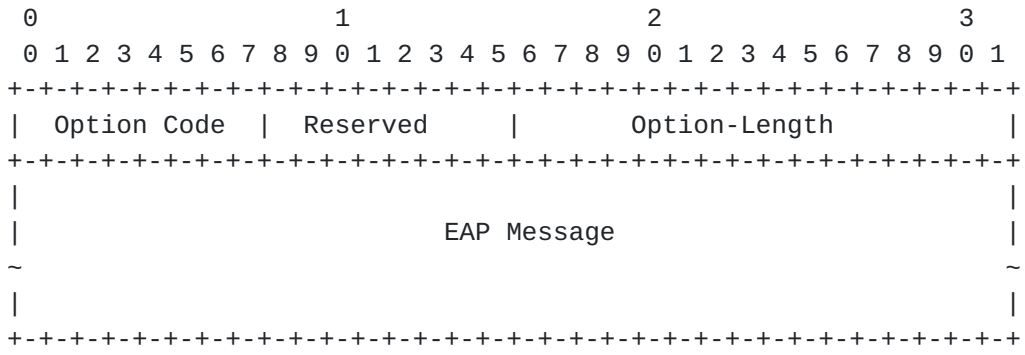

Option-Length: The length of the Authentication Tag Option (in octet), including the 8 octet fixed header and the variable length of the authentication data.

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.

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.

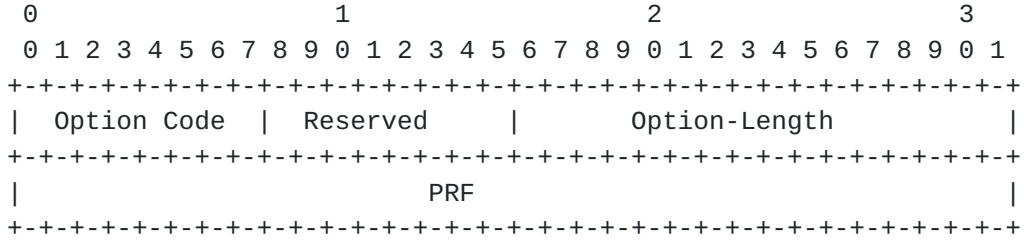
Authentication Data: A Variable length field that carries the Message Authentication Code for the PCP packet. The generation of the digest can be various according to the algorithms specified in different PCP SAs. This field MUST end on a 32-bit boundary, padded with 0's when necessary

## 8.4. EAP Payload Option



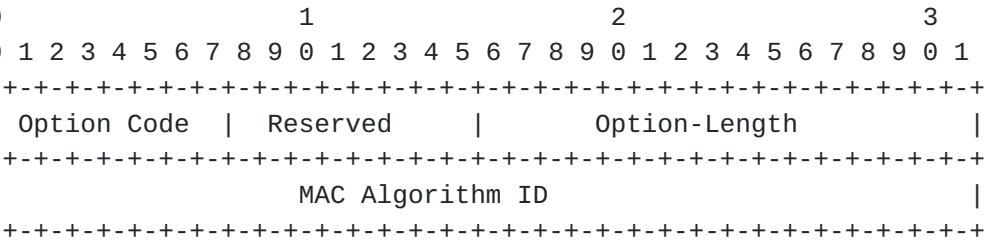

EAP Message: The EAP message transferred. Note this field MUST end on a 32-bit boundary, padded with 0's when necessary.

## 8.5. PRF Option



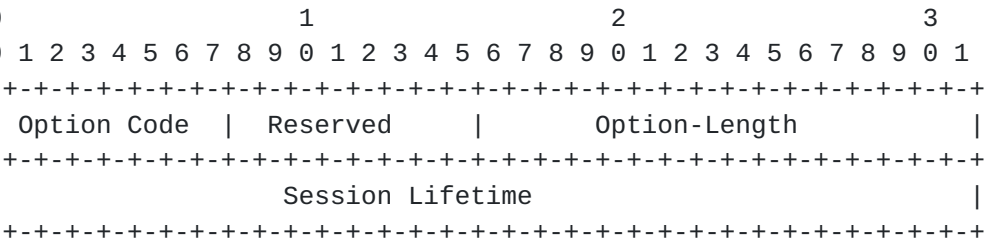

PRF: The pseudo-random Function which the sender supports to generate a MSK. This filed contains an IKEv2 Transform ID of Transform Type 2 [RFC4306].

### 8.6. Hash Algorithm Option




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].

### 8.7. Session Lifetime Option




Session Lifetime: The life period of the PA Session, which is decided by the authorization result.

## 9. Processing Rules

### 9.1. Authentication Data Generation

If a PCP SA is generated as the result of an 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 message, a device needs to first select a traffic key in the session and append the Authentication Tag Option at the end of the protected PCP message. The length of the Authentication Data field is decided by the MAC algorithm adopted in the session. The device then fills the Session ID field and the PCP SA ID field, and sets the Authentication Data field as 0. After this, the device generates a digest for the entire PCP message (including the PCP header and Authentication Tag Option) with the MAC algorithm and the selected traffic key, and input the generated digest into the Authentication Data field.

### 9.2. Authentication Data Validation

When a device receives a PCP packet with an Authentication Tag Option, it needs to use the session ID transported in the option to locate the proper SA, and then find out the associated transport key (using key ID) and the MAC algorithm. If no proper SA 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 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 during the transportation. The validation successes. Otherwise, the packet MUST be discarded.

### 9.3. Sequence Number

PCP adopts UDP to transport signaling messages. As an un-reliable transporting protocol, UDP does not guarantee the 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 carries an monotonically increased sequence number. During a PA session, a PCP device needs to maintain two sequence numbers, one for incoming packets and one for outgoing packets. When generating an outgoing PCP packet, the device attaches the outgoing sequence number to the packet and increments the sequence number maintained in the SA by 1. When receiving a PCP 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. However, the above rules are not applied to PA-Acknowledgement messages. When receiving or sending out a PA-Acknowledgement message, the device does not inicrease the correspondent sequence number stored in the SA. Another exception is message retransmission. When a device does not receive any response message from its session partner in a certain period, it needs to retransmit the last sent message with a limited rate. The duplicate messages and the original message MUST use the identical sequence number. When the device receives such duplicate messages from its session partner, it MUST tries to answer them by sending the last outgoing message with a limited rate unless it has received another valid message with a larger sequence number from its session. In such cases, the maintained incoming and outgoing sequence numbers will not

be affected by the message retransmission.

### 9.4. Retransmission Policies

This work provides a retransmission mechanism for reliable PA message delivery. The timer, the variables, and the rules used in this mechanism is mostly brought from PANA.

The retransmission behavior is controlled and described by the following variables:

RT: Retransmission timeout from the previous (re)transmission

IRT: Base value for RT for the initial retransmission

MRC: Maximum retransmission count

MRT: Maximum retransmitting time interval

RAND: Randomization factor

With each message transmission or retransmission, the sender sets RT according to the rules given below.

If RT expires before receiving any reply, the sender re-calculates RT and retransmits the message. Each of the computations of a new RT include a randomization factor (RAND), which is a random number chosen with a uniform distribution between -0.1 and +0.1. The randomization factor is included to minimize the synchronization of messages. The algorithm for choosing a random number does not need to be cryptographically sound. The algorithm SHOULD produce a different sequence of random numbers from each invocation. RT for the first message retransmission is based on IRT:

RT = IRT

RT for each subsequent message retransmission is based on the previous value of RT (RTprev):

RT = (2+RAND) * RTprev

MRT specifies an upper bound on the value of RT (disregarding the randomization added by the use of RAND). If MRT has a value of 0, there is no upper limit on the value of RT. Otherwise:

if (RT > MRT)

RT = (1+RAND) * MRT

MRC specifies an upper bound on the number of times a sender may retransmit a message. Unless MRC is zero, the message exchange fails once the sender has transmitted the message MRC times. In this case, the sender needs to start a session termination process illustrated in Section 3.2.

### 9.5. MTU Considerations

EAP methods are responsible for MTU handling, so no special facilities are required in this protocol to deal with MTU issues.

## 10. IANA Considerations

TBD

## 11. 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. This work does not exclude the possibility of using the MSK to generate keys for different security protocols to enable per-packet cryptographic protection. The methods of deriving the transport key for the security protocols is out of scope of this document.

However, before a transport key has been generated, the PA messages exchanged within a PA session have little cryptographic protection, and if there is no already established security channel between two session partners, these messages are subject to man-in-the-middle attacks and DOS attacks. For instance, the initial PA-Request and PA-Answer exchange is vulnerable to spoofing attacks as these messages are not authenticated and integrity protected. In order to prevent very basic DOS attacks, a PCP device SHOULD generate state information as little as possible in the initial PA-Request and PA-Answer exchanges. The choice of EAP method is also very important. The selected EAP method must be resilient to the attacks possibly occurred in a insecure network environment, and the user-identity confidentiality, protection against dictionary attacks, and session-key establishment must be supported.

## 12. Acknowledgements

This document was written using the xml2rfc tool described in RFC 2629 [RFC2629].

## 13. Change Log

### 13.1. Changes from wasserman-pcp-authentication-02 to ietf-pcp-authentication-00

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

### 13.2. Changes from wasserman-pcp-authentication-01 to -02

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

### 13.3. Changes from wasserman-pcp-authentication-00 to -01

- Editorial changes, added use cases to introduction.

## 14. References

### 14.1. Normative References

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

## 14.2. Informative References

[I-D.ietf-pcp-base]
Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", draft-ietf-pcp-base-26 (work in progress), June 2012.

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.

[RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005.

[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.

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

Phone:
Fax:
Email: zhangdacheng@huawei.com
URI: