PANA Working Group
Internet-Draft
Expires: November 5, 2004

D. Forsberg
Nokia
Y. Ohba (Ed.)
Toshiba
B. Patil
Nokia
H. Tschofenig
Siemens
A. Yegin
Samsung

May 7, 2004

Protocol for Carrying Authentication for Network Access (PANA)
draft-ietf-pana-pana-04

Status of this Memo

Copyright Notice

Abstract

This document defines the Protocol for Carrying Authentication for Network Access (PANA), a link-layer agnostic transport for Extensible Authentication Protocol (EAP) to enable network access authentication between clients and access networks. PANA can carry any authentication method that can be specified as an EAP method, and can

be used on any link that can carry IP. PANA covers the client-to-network access authentication part of an overall secure network access framework, which additionally includes other protocols and mechanisms for service provisioning, access control as a result of initial authentication, and accounting.

Table of Contents

1. Introduction

Providing secure network access service requires access control based on the authentication and authorization of the clients and the access networks. Initial and subsequent client-to-network authentication provides parameters that are needed to police the traffic flow through the enforcement points. A protocol is needed to carry authentication methods between the client and the access network.

Currently there is no standard network-layer solution for authenticating clients for network access. [I-D.ietf-pana-usage-scenarios] describes the problem statement that led to the development of PANA.

Scope of this work is identified as designing a link-layer agnostic transport for network access authentication methods. The Extensible Authentication Protocol (EAP) [I-D.ietf-eap-rfc2284bis] provides such authentication methods. In other words, PANA will carry EAP which can carry various authentication methods. By the virtue of enabling transport of EAP above IP, any authentication method that can be carried as an EAP method is made available to PANA and hence to any

link-layer technology. There is a clear division of labor between PANA, EAP and EAP methods.

Various environments and usage models for PANA are identified in the [I-D.ietf-pana-usage-scenarios] Internet-Draft. Potential security threats for network-layer access authentication protocol are discussed in [I-D.ietf-pana-threats-eval] draft. These two drafts have been essential in defining the requirements [I-D.ietf-pana-requirements] on the PANA protocol. Note that some of these requirements are imposed by the chosen payload, EAP [I-D.ietf-eap-rfc2284bis].

There are components that are part of a complete secure network solution but are outside of the PANA protocol specification, including IP address configuration, authentication method choice, filter rule installation, data traffic protection and PAA-EP protocol. These components are described in separate documents [I-D.ietf-pana-framework][I-D.ietf-pana-snmp].

1.1 Specification of Requirements

In this document, several words are used to signify the requirements of the specification. These words are often capitalized. 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].

2. Terminology

This section describes some terms introduced in this document:

PANA Session:

PANA session is defined as the exchange of messages between the PANA Client (PaC) and the PANA Authentication Agent (PAA) to authenticate a user (PaC) for network access. If the authentication is unsuccessful, the session is terminated. The session is considered as active until there is a disconnect indication by the PaC or the PAA terminates it. A distinct PANA session is associated with a pair of device identifiers of PaC and PAA. For example, if the PaC has two interfaces connected to the

same IP link with different IP addresses and IP address is used as a device identifier, a distinct PANA session will be created per interface if both interfaces addresses need to be authorized for network access.

Session Identifier:

This identifier is used to uniquely identify a PANA session on the PAA and PaC. It is included in PANA messages to bind the message to a specific PANA session.

PANA Security Association:

A PANA security association is a relationship between the PaC and PAA, formed by the sharing of cryptographic keying material and associated context. Security associations are duplex. That is, one security association is needed to protect the bidirectional traffic between the PaC and the PAA.

PANA Client (PaC):

The client side of the protocol that resides in the host device which is responsible for providing the credentials to prove its identity for network access authorization.

Device Identifier (DI):

The identifier used by the network as a handle to control and police the network access of a client. Depending on the access technology, this identifier might contain any of IP address, link-layer address, switch port number, etc. of a connected device.

PANA Authentication Agent (PAA):

The access network side entity of the protocol whose responsibility is to verify the credentials provided by a PANA client and grant network access service to the device associated with the client and identified by a DI.

Enforcement Point (EP):

A node on the access network where per-packet enforcement policies (i.e., filters) are applied on the inbound and outbound traffic of client devices. Information such as DI and (optionally) cryptographic keys are provided by PAA per client for constructing filters on the EP.

Network Access Provider (NAP):

A service provider that provides physical and link-layer connectivity to an access network it manages.

AAA-Key:

A key derived by the EAP peer and EAP server and transported to the authenticator [I-D.ietf-eap-keying].

3. Protocol Overview

The PANA protocol involves two functional entities namely the PaC and the PAA. The protocol resides above the transport layer and the details are explained in Section 4.

The placement of the entities used in PANA largely depends on a certain architecture. The PAA may optionally interact with a AAA backend to authenticate the user (PaC). The EP, mentioned in the context with PANA, is a logical entity. There is, however, the option that the EP is not physically co-located with the PAA. In case that the PAA and the EP are co-located only an API is required for intercommunication instead of a separate protocol. In the case where the PAA is separated from the EP, a separate protocol will be used between the PAA and the EP for managing access control. The protocol and messaging between the PAA and EP for access authorization is outside the scope of this draft and will be dealt separately. Figure 1 illustrates the interactions in a simplified manner:

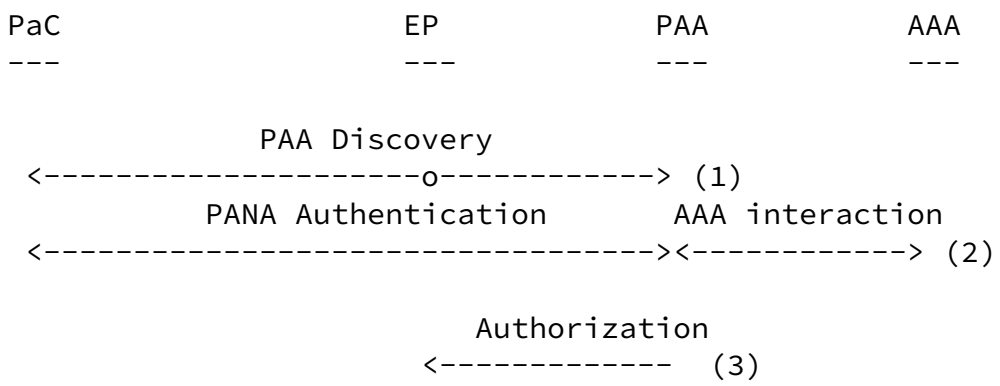


Figure 1: PANA Framework

PANA supports authentication of a PaC using various EAP methods. The EAP method used depends on the level of security required for the EAP messaging itself. PANA does not secure the data traffic itself. However, EAP methods that enable key exchange may allow other protocols to be bootstrapped for securing the data traffic [I-D.ietf-pana-ipsec].

From a state machine aspect, PANA protocol consists of three phases

1. Discovery and initial handshake phase

2. Authentication phase

3. Termination phase

In the first phase, an IP address of PAA is discovered and a PANA

session is established between PaC and PAA. EAP messages are exchanged and a PANA SA is established in the second phase. The established PANA session as well as a PANA SA is deleted in the third phase.

4. Protocol Details

4.1 Common Processing Rules

4.1.1 Payload Encoding

The payload of any PANA message consists of zero or more AVPs (Attribute Value Pairs). A brief description of the AVPs defined in this document is listed below:

- Cookie AVP: contains a random value that is used for making initial handshake robust against blind resource consumption DoS attacks.
- Protection-Capability AVP: contains information which protection should be initiated after the PANA exchange (e.g., link-layer or network layer protection).
- Device-Id AVP: contains a device identifier of the sender of the message. A device identifier is represented as a pair of device identifier type and device identifier value. Either a layer-2 address or an IP address is used for the device identifier value.
- EP-Device-Id AVP: contains the device identifier of an EP.
- EAP AVP: contains an EAP PDU.
- MAC AVP: contains a Message Authentication Code that protects a PANA message PDU.
- Termination-Cause AVP: contains the reason of session termination.
- Result-Code AVP: contains information about the protocol execution results.
- Session-Id AVP: contains the session identifier value.

- Session-Lifetime AVP: contains the duration of authorized access.
- Failed-AVP: contains the offending AVP that caused a failure.
- NAP-Information AVP, ISP-Information AVP: contains the information on a NAP and an ISP, respectively.
- Key-Id AVP: contains a AAA-Key identifier.
- PPAC AVP: Post-PANA-Address-Configuration AVP. Conveys the list of IP address configuration methods available when sent by the

PAA, and the chosen method when sent by the PaC.

- Nonce AVP: contains a randomly chosen value.

4.1.2 Transport Layer Protocol

PANA uses UDP as its transport layer protocol. The UDP port number is TBD. All messages except for PANA-PAA-Discover are always unicast. PANA-PAA-Discover MAY be unicasted when the PaC knows the IP address of the PAA.

4.1.3 Fragmentation

PANA does not provide fragmentation of PANA messages. Instead, it relies on fragmentation provided by EAP methods and IP layer when needed.

4.1.4 Sequence Number and Retransmission

PANA uses sequence numbers to provide ordered delivery of EAP messages. The design involves use of two sequence numbers to prevent some of the DoS attacks on the sequencing scheme. Every PANA packet include one transmitted sequence number (tseq) and one received sequence number (rseq) in the PANA header. See [1] for detailed explanation on why two sequence numbers are needed.

The two sequence number fields have the same length of 32 bits and appear in PANA header. tseq starts from initial sequence number

(ISN) and is monotonically increased by 1. The serial number arithmetic defined in [RFC1982] is used for sequence number operation. The ISNs are exchanged between PaC and PAA during the discovery and initial handshake phase (see Section 4.2). The rules that govern the sequence numbers in other phases are described as follows.

- When a message is sent, a new sequence number is placed on the tseq field of message regardless of whether it is sent as a result of retransmission or not. When a message is sent, rseq is copied from the tseq field of the last accepted message.

- When a message is received, it is considered valid in terms of sequence numbers if and only if (i) its tseq is greater than the tseq of the last accepted message and (ii) its rseq falls in the range between the tseq of the last acknowledged message + 1 and the tseq of the last transmitted message.

PANA relies on EAP-layer retransmissions, or for example NAS

functionality [I-D.ietf-aaa-eap], for retransmitting EAP Requests based on timer. Other PANA layer messages that require a response from the communicating peer are retransmitted based on timer at PANA-layer until a response is received (in which case the retransmission timer is stopped) or the number of retransmission reaches the maximum value (in which case the PANA session MUST be deleted immediately). For PANA-layer retransmission, the retransmission timer SHOULD be calculated as described in [RFC2988] to provide congestion control. See Section 7 for default timer and maximum retransmission count parameters.

4.1.5 PANA Security Association

A PANA SA is created as an attribute of a PANA session when EAP authentication succeeds with a creation of a AAA-Key. A PANA SA is not created when the PANA authentication fails or no AAA-Key is produced by any EAP authentication method. In the case where two EAP authentications are performed in sequence in a single PANA authentication phase, it is possible that two AAA-Keys are derived. If this happens, the PANA SA MUST be generated from both AAA-Keys. When a new AAA-Key is derived as a result of EAP-based re-authentication, any key derived from the old AAA-Key MUST be

updated to a new one that is derived from the new AAA-Key. In order to distinguish the new AAA-Key from old ones, one Key-Id AVP MUST be carried in PANA-Bind-Request and PANA-Bind-Answer messages or PANA-FirstAuth-End-Request and PANA-FirstAuth-End-Answer messages at the end of the EAP authentication which resulted in deriving a new AAA-Key. The Key-Id AVP is of type Unsigned32 and MUST contain a value that uniquely identifies the AAA-Key within the PANA session. The PANA-Bind-Answer message (or the PANA-FirstAuth-End-Answer message) sent in response to a PANA-Bind-Request message (or a PANA-FirstAuth-End-Request message) with a Key-Id AVP MUST contain a Key-Id AVP with the same AAA-Key identifier carried in the request. PANA-Bind-Request, PANA-Bind-Answer, PANA-FirstAuth-End-Request and PANA-FirstAuth-End-Answer messages with a Key-Id AVP MUST also carry a MAC AVP whose value is computed by using the new PANA-MAC-Key derived from the new AAA-Key (or the new pair of AAA-Keys when the PANA_MAC_KEY is derived from two AAA-Keys). Although the specification does not mandate a particular method for calculation of Key-Id AVP value, a simple method is to use monotonically increasing numbers."

The created PANA SA is deleted when the corresponding PANA session is deleted. The lifetime of the PANA SA is the same as the lifetime of the PANA session for simplicity.

PANA SA attributes as well as PANA session attributes are listed below:

PANA Session attributes:

* Session-Id
* Device-Id of PaC
* Device-Id of PAA
* List of device identifiers of EPs
* Initial tseq of PaC (ISN_pac)
* Initial tseq of PAA (ISN_paa)
* Last transmitted tseq value

* Last received rseq value

* Last transmitted message payload

* Retransmission interval

* Session lifetime

* Protection-Capability

* PANA SA attributes:

  + AAA-Key

  + AAA-Key Identifier

  + PANA_MAC_Key

The PANA_MAC_Key is used to integrity protect PANA messages. When the PANA_MAC_Key is derived from a single AAA-Key, it is computed in the following way:

```
PANA_MAC_KEY = The first N bits of
               HMAC_SHA1(AAA-Key, ISN_pac | ISN_paa | Session-ID)
```

where the value of N depends on the integrity protection algorithm in use, i.e., N=160 for HMAC-SHA1.

When the PANA_MAC_Key is derived from two AAA-Keys, it is computed in the following way:


```
PANA_MAC_KEY = The first N bits of
               HMAC_SHA1(AAA-Key1 | AAA-Key2, ISN_pac | ISN_paa |
                         Session-ID)
```

where AAA-Key1 and AAA-Key2 are AAA-Keys for the first and second EAP authentication in a single PANA authentication phase, respectively.

The length of AAA-Key, AAA-Key1 and AAA-Key2 MUST be N bits or

longer. See Section 4.1.6 for the detailed usage of the PANA_MAC_Key.

4.1.6 Message Authentication Code

A PANA message can contain a MAC (Message Authentication Code) AVP for cryptographically protecting the message.

When a MAC AVP is included in a PANA message, the value field of the MAC AVP is calculated by using the PANA_MAC_Key in the following way:

```
MAC AVP value = PANA_MAC_PRF(PANA_MAC_Key, PANA_PDU)
```

where PANA_PDU is the PANA message including the PANA header, with the MAC AVP value field first initialized to 0. PANA_MAC_PRF represents the pseudo random function corresponding to the MAC algorithm specified in the MAC AVP. In this version of draft, PANA_MAC_PRF is HMAC-SHA1. The PaC and PAA MUST use the same algorithm to calculate a MAC AVP they originate and receive. The algorithm is determined by the PAA when a PANA-Bind-Request with a MAC AVP is sent. When the PaC does not support the MAC algorithm specified in the PANA-Bind-Request message, it MUST silently discard the message. The PAA MUST NOT change the MAC algorithm throughout the continuation of the PANA session.

4.1.7 Message Validity Check

When a PANA message is received, the message is considered to be invalid at least when one of the following conditions are not met:

- The IP Hop Limit (or TTL) field has a value of 255, i.e., the packet could not possibly have been forwarded by a router.

- Each field in the message header contains a valid value including sequence number, message length, message type, version number, flags, etc.

- When a device identifier of the communication peer is bound to the PANA session, it matches the device identifier carried in MAC and/or IP header(s), or other auxiliary indetifier provided by the

lower-layers (e.g., circuit ID).

- The message type is one of the expected types in the current state.

- The message payload contains a valid set of AVPs allowed for the message type and there is no missing AVP that needs to be included in the payload.

- Each AVP is decoded correctly.

- When a MAC AVP is included, the AVP value matches the MAC value computed against the received message.

- When a Device-Id AVP is included, the AVP is valid if the device identifier type contained in the AVP is supported (this check is for both PaC and PAA) and is the requested one (this check is for PAA only) and the device identifier value contained in the AVP matches the value extracted from the lower-layer encapsulation header corresponding to the device identifier type contained in the AVP. Note that a Device-Id AVP carries the PaC's device identifier in messages from PaC to PAA and PAA's device identifier in messages from PAA to PaC.

Invalid messages MUST be discarded in order to provide robustness against DoS attacks and an unprotected. In addition, a non-acknowledged error notification message MAY be returned to the sender. See Section 4.1.8 for details.

4.1.8 Error Handling

PANA-Error message MAY be sent by either PaC or PAA when a badly formed PANA message is received or in case of other errors. If the cause of this error message was a request message (e.g., PANA-PAA-Discover or *-Request), then the request MAY be retransmitted immediately without waiting for its retransmission timer to go off. If the cause of the error was a response message, the receiver of the PANA-Error message SHOULD NOT resend the same response until it receives the next request.

To defend against DoS attacks a timer MAY be used. One (1) error notification is sent to each different sender each N seconds. N is a configurable parameter.

When an error message is sent unprotected with MAC AVP and the lower-layer is insecure, the error message is treated as an informational message. The receiver of such an error message MUST NOT change its state unless the error persists and the PANA session

is not making any progress.

4.2 Discovery and Initial Handshake Phase

When a PaC attaches to a network, and knows that it has to discover PAA for PANA, it SHOULD send a PANA-PAA-Discover message to a well-known link local multicast address (TBD) and UDP port (TBD). PANA PAA discovery assumes that PaC and PAA are one hop away from each other. If PaC knows the IP address of the PAA (some pre-configuration), it MAY unicast the PANA discovery message to that address. PAA SHOULD answer to the PANA-PAA-Discover message with a PANA-Start-Request message.

When the PAA receives such a request, or upon receiving some lower layer indications of a new PaC, PAA SHOULD unicast a PANA-Start-Request message.

There can be multiple PAAs on the link. The authentication and authorization result does not depend on which PAA is chosen by the PaC. By default the PaC MAY choose the PAA that sent the first response.

PaC MAY also choose to start sending packets before getting authenticated. In that case, the network MAY detect this and send an unsolicited PANA-Start-Request message to PaC in addition to filtering the unauthorized traffic. EP is the node that can detect such activity. PAA-to-EP protocol MAY be used for this purpose.

A PANA-Start-Request message MAY carry a Cookie AVP that contains a cookie. The rseq field of the header is set to zero (0). The tseq field of the header contains the initial sequence number. The cookie is used for preventing the PAA from resource consumption DoS attacks by blind attackers. The cookie is computed in such a way that it does not require any per-session state maintenance on the PAA in order to verify the cookie returned in a PANA-Start-Answer message. The exact algorithms and syntax used for generating cookies does not affect interoperability and hence is not specified here. An example algorithm is described below.

```
Cookie =
  <secret-version> | HMAC_SHA1( <Device-Id of PaC> | <secret> )
```

where <secret> is a randomly generated secret known only to the PAA, <secret-version> is an index used for choosing the secret for generating the cookie and '|' indicates concatenation. The secret-version should be changed frequently enough to prevent replay

attacks. The secret key is locally known to the PAA only and valid for a certain time frame.

Protection-Capability and Post-PANA-Address-Configuration AVPs MAY be optionally included in the PANA-Start-Request in order to indicate required and available capabilities for the network access. These AVPs MAY be used by the PaC for assessing the capability match even before the authentication takes place. But these AVPs are provided during the insecure discovery phase, there are certain security risks involved in using the provided information. See Section 9 for further discussion on this.

PAA MAY enable NAP-ISP authentication separation by setting the S-flag of the message header of the PANA-Start-Request. Also, the PANA-Start-Request MAY contain zero or one NAP-Information AVP and zero or more ISP-Information AVPs to advertise the information on the NAP and/or ISPs.

When a PaC receives the PANA-Start-Request message in response to the PANA-PAA-Discover message, it responds with a PANA-Start-Answer message if it wishes to enter the authentication phase. The PANA-Start-Answer message contains the initial sequence numbers in the tseq and rseq fields of the PANA header, a copy of the received Cookie (if any) as the PANA payload.

If the S-flag of the received PANA-Start-Request message is not set, PaC MUST NOT set the S-flag in the PANA-Start-Answer message sent back to the PAA. In this case, PaC MAY indicate its choice of ISP by including an ISP-Information AVP in the PANA-Start-Answer message. When a AAA backend is used, the identity of the destination AAA server or realm MUST be determined based on the explicitly chosen ISP. When the ISP-Information AVP is not present, the access network MAY rely on the client identifier carried in the EAP authentication method to make this determination.

If the S-flag of the received PANA-Start-Request message is set, PaC can indicate its desire to perform separate EAP authentication for NAP and ISP by setting the S-flag in the PANA-Start-Answer message. If the S-flag in the PANA-Start-Answer message is not set, only one authentication is performed and the processing occurs as described earlier. If the S-flag in the PANA-Start-Answer message is set, the determination of the destination AAA server or realm for ISP

authentication is performed as described earlier. In addition, where backend AAA servers are used for NAP authentication, the NAP is considered the ultimate AAA realm, and the destination AAA server for this authentication is determined entirely by the local configuration on the access server hosting PAA (NAS).

The PaC can choose an ISP and contain an ISP-Information AVP for the chosen ISP in a PANA-Start-Answer message even when there is no ISP-Information AVP contained in the PANA-Start-Request message.

When the PAA receives the PANA-Start-Answer message from the PaC, it verifies the cookie. The cookie is considered as valid if the received cookie has the expected value. If the computed cookie is valid, the protocol enters the authentication phase. Otherwise, it MUST silently discard the received message.

Initial EAP Request MAY be optionally carried by the PANA-Start-Request (as opposed to by a later PANA-Auth-Request) message in order to reduce the number of round-trips. This optimization SHOULD NOT be used if the PAA discovery is desired to be stateless.

When the S-flag is set in a PANA-Start-Request message, the initial EAP Request MUST NOT be carried in the PANA-Start-Request message. (If the initial EAP Request were contained in the PANA-Start-Request message during the S-flag negotiation, the PaC cannot tell whether the EAP Request is for NAP authentication or ISP authentication.)

If the initial EAP Request message is carried in the PANA-Start-Request message, an EAP Response message MUST be carried in the PANA-Start-Answer message returned to the PAA.

In any case, PANA MUST NOT generate an EAP message on behalf of EAP peer or EAP (pass-through) authenticator.

The PANA-Start-Request/Answer exchange is needed before entering authentication phase even when the PaC is pre-configured with PAAs IP address and the PANA-PAA-Discover message is unicast.

A PANA-Start-Request message that carries a Cookie AVP is never retransmitted. A PANA-Start-Request message that does not carry a Cookie AVP is retransmitted based on timer. A PANA-Start-Answer

message that carries a Cookie AVP is retransmitted based on timer. A PANA-Start-Answer message that does not carry a Cookie AVP is never retransmitted based on timer.

It is possible that both PAA and PaC initiate the discovery and initial handshake procedure at the same time, i.e., the PAA sends a PANA-Start-Request message while the PaC sends a PANA-PAA-Discover message. To resolve the race condition, the PAA SHOULD silently discard the PANA-PAA-Discover message received from the PaC after it has sent a PANA-Start-Request message with creating a state (i.e., no Cookie AVP included) for the PaC. In this case PAA will retransmit PANA-Start-Request based on a timer, if PaC doesn't respond in time (message was lost for example). If PAA had sent stateless PANA-Start-Request message (i.e., a Cookie AVP was included), then it SHOULD answer to the PANA-PAA-Discover message.

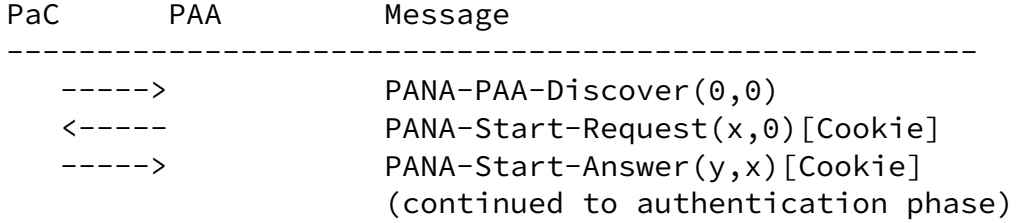


Figure 2: Example Sequence for Discovery and Initial Handshake Phase when PANA-PAA-Discover is sent by PaC

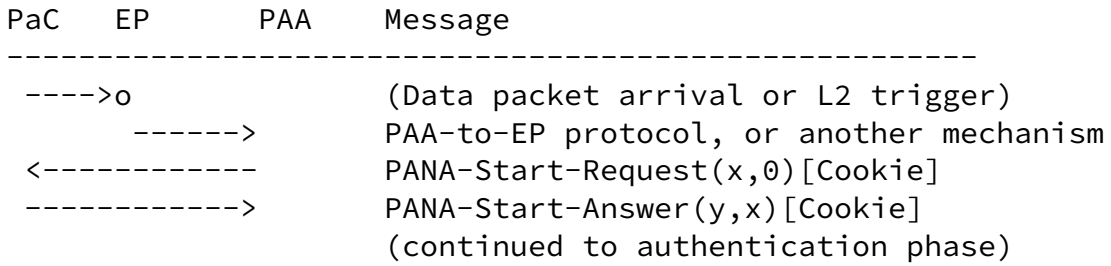


Figure 3: Example Sequence for Discovery and Initial Handshake when discovery is triggered by data traffic

4.3 Authentication Phase

The main task in authentication phase is to carry EAP messages between PaC and PAA. EAP Request messages are carried in PANA-Auth-Request messages and optionally carried in PANA-Start-Request messages. EAP Response messages are carried in PANA-Auth-Answer messages and optionally carried in PANA-Start-Answer messages. When an EAP Success/Failure message is sent from a PAA, the message is carried in a PANA-Bind-Request (PBR) or PANA-FirstAuth-End-Request (PFER) message. The PANA-FirstAuth-End-Reques message MUST be used at the end of the first EAP when the PaC and PAA have negotiated during the discovery and initial handshake phase to perform separate NAP and ISP authentications in a single PANA authentication phase. Otherwise, the PANA-Bind-Request message MUST be used. The PANA-Bind-Request and PANA-FirstAuth-End-Request messages MUST be acknowledged with a PANA-Bind-Answer (PBA) and a PANA-FirstAuth-End-Answer (PFEA) messages, respectively.

When the PaC and PAA have negotiated during the discovery and initial handshake phase to perform separate NAP and ISP authentications, the S-flag of PANA-Auth-Request and PANA-Auth-Answer messages MUST be set. Otherwise, the S-flag MUST NOT be set.

When separate NAP and ISP authentications are performed, the PAA determines the execution order of NAP authentication and ISP authentication. In this case, the PAA can indicate which EAP authentication is currently occurring by using N-flag in the PANA message header. When NAP authentication is performed, the N-flag MUST be set. When ISP authentication is performed, the N-flag MUST NOT be set. The N-flag MUST NOT be set when S-flag is not set.

When separate NAP and ISP authentications are performed, if the first EAP authentication has failed, the PAA can choose not to perform the second EAP authentication by clearing the S-flag of the PANA-FirstAuth-End-Request message. In this case, the S-flag of the PANA-FirstAuth-End-Answer message sent by the PaC MUST be cleared. If the S-flag of the PANA-FirstAuth-End-Request message is set when the first EAP authentication has failed, the PaC can choose not to perform the second EAP authentication by clearing the S-flag of the PANA-FirstAuth-End-Answer message. If the first EAP authentication failed and the S-flag is not set in the PANA-FirstAuth-End-Answer

message as a result of those operations, the PANA session MUST be immediately deleted. Otherwise, the second EAP authentication MUST be performed.

Currently, use of multiple EAP methods in PANA is designed only for NAP-ISP authentication separation. It is not for arbitrary EAP method sequencing, or giving the PaC another chance when an authentication method fails. The NAP and ISP authentication are considered completely independent. Presence or success of one should not effect the other. Making a network access authorization decision based on the success or failure of each authentication is a network policy issue.

When an EAP method that is capable of deriving keys is used during the authentication phase and the keys are successfully derived, the PANA-FirstAuth-End-Request and PANA-FirstAuth-End-Answer and/or PANA-Bind-Request and PANA-Bind-Answer messages, and all subsequent PANA messages MUST contain a MAC AVP.

When separate NAP and ISP authentications are performed and the lower-layer is insecure, the two EAP methods MUST be capable of deriving keys. In this case, if the first EAP authentication is successful, the PANA-FirstAuth-End-Request and PANA-FirstAuth-End-Answer messages as well as PANA-Auth-Request and PANA-Auth-Answer messages in the second EAP authentication MUST be protected with the key derived from the AAA-Key for the first EAP authentication. The PANA-Bind-Request and PANA-Bind-Answer messages and all subsequent PANA messages MUST be protected either with the AAA-Key for the first EAP authentication if the first EAP authentication succeeds and the second EAP authentication fails, or

with the AAA-Key for the second EAP authentication if the first EAP authentication fails and the second EAP authentication succeeds, or with the compound key derived from the two AAA-Keys, one for the first EAP authentication and the other from the second EAP authentication, if both the first and second EAP authentications succeed (see Section 4.1.5 for how the compound key is derived).

The PANA-Bind-Request and the PANA-Bind-Answer message exchange is also used for binding device identifiers of the PaC and the PAA to the PANA SA when the identifiers are either IP or MAC addresses. To achieve this, the PANA-Bind-Request and the PANA-Bind-Answer SHOULD

contain a device identifier of the PAA and the PaC, respectively, in a Device-Id AVP. Device identifier exchange that is protected by a MAC AVP prevents man-in-the-middle attacks. The PaC MUST use the same type of device identifier as contained in the PANA-Bind-Request message. The PANA-Bind-Request message MAY also contain a Protection-Capability AVP to indicate if link-layer or network-layer ciphering should be initiated after PANA. No link layer or network layer specific information is included in the Protection-Capability AVP. When the information is preconfigured on the PaC and the PAA this AVP can be omitted. It is assumed that at least PAA is aware of the security capabilities of the access network. The PANA protocol does not specify how the PANA SA and the Protection-Capability AVP will be used to provide per-packet protection for data traffic.

Additionally, PANA-Bind-Request MUST include a Post-PANA-Address-Configuration AVP, which helps PAA to inform PaC about whether a new IP address MUST be configured and the available methods to do so. PaC MUST include a PPAC AVP in order to indicate its choice of method when there is a match between the methods offered by the PAA and the methods available on the PaC. When there is no match, a PPAC AVP MUST NOT be included and the Result-Code AVP MUST be set to PANA_PPAC_CAPABILITY_UNSUPPORTED in the PANA-Bind-Answer message.

PANA-Bind-Request and PANA-Bind-Answer messages MUST be retransmitted based on the retransmission rule described in Section 4.1.4.

EAP authentication can fail at a pass-through authenticator without sending an EAP-Failure message [I-D.ietf-eap-statemachine]. When this occurs, the PAA SHOULD send a PANA-Error message to the PaC with using PANA_UNABLE_TO_COMPLY result code. The PaC SHOULD ignore the message unless it is secured by PANA or lower layer. In any case, a more appropriate way is to rely on a timeout on the PaC.

There is a case where EAP authentication succeeds with producing an EAP-Success message but network access authorization fails due to, e.g., authorization rejected by a AAA proxy or authorization locally

rejected by a PAA. When this occurs, the PAA MUST send PANA-Bind-Request with a result code PANA_AUTHORIZATION_REJECTED. If a AAA-Key is established between PaC and PAA by the time when the EAP-Success is generated by the EAP server (this is the case when the

EAP method provides protected success indication), the this PANA-Bind message exchange MUST be protected with a MAC AVP and with carrying a Key-Id AVP. The AAA-Key and the PANA session MUST be deleted after the PANA-Bind message exchange.

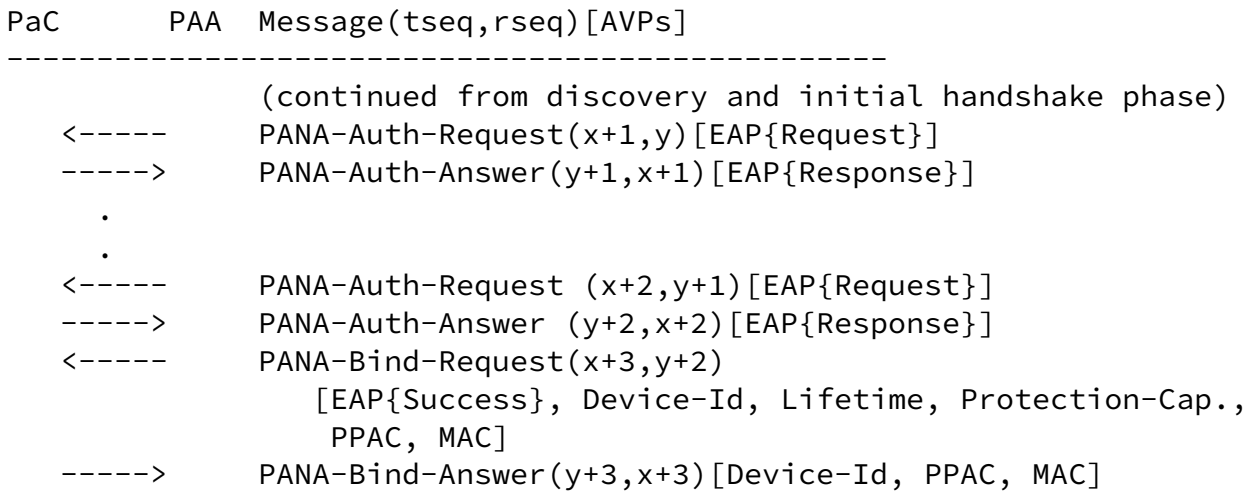


Figure 4: Example Sequence in Authentication Phase

4.4 Re-authentication

There are two types of re-authentication supported by PANA.

The first type of re-authentication is based on EAP by entering an authentication phase. In this case, some or all message exchanges for discovery and initial handshake phase MAY be omitted in the following way. When a PaC wants to initiate EAP-based re-authentication, it sends a unicast PANA-PAA-Discovery message to the PAA. This message MUST contain a Session-Id AVP which is used for identifying the PANA session on the PAA. If the PAA already has an established PANA session for the PaC with the matching identifier, it sends a PANA-Auth-Request message containing the same identifier to start an authentication phase. If the PAA can not recognize the session identifier, it proceeds with regular authentication by sending back PANA-Start-Request. When the PAA initiates EAP-based re-authentication, it sends a PANA-Auth-Request message containing the session identifier for the PaC to enter an authentication phase. PAA SHOULD initiate EAP authentication before the current session lifetime expires. In both cases, the tseq and rseq values are inherited from the previous (re-)authentication. For any EAP-based re-authentication, if there is an established PANA SA,

PANA-Auth-Request and PANA-Auth-Answer messages MUST be protected by adding a MAC AVP to each message.

The second type of re-authentication is based on a single protected message exchange without entering the authentication phase. PANA-Reauth-Request and PANA-Reauth-Answer messages are used for this purpose. If there is an established PANA SA, both the PaC and the PAA are allowed to send a PANA-Reauth-Request message to the communicating peer whenever it needs to make sure the availability of the PANA SA on the peer and expect the peer to return a PANA-Reauth-Answer message. Both PANA-Reauth-Request/ PANA-Reauth-Answer messages MUST be protected with a MAC AVP.

Implementations MUST limit the rate of performing re-authentication for both types of re-authentication. The PaC and the PAA can handle rate limitation on their own, they don't have to perform any coordination with each other. There is no negotiation of timers for this purpose.

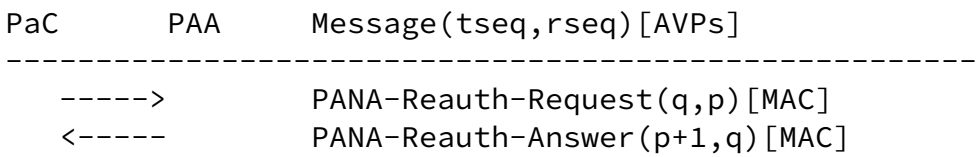


Figure 5: Example Sequence for PaC-initiated second type Re-authentication

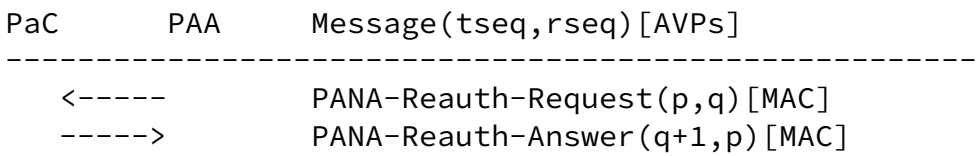


Figure 6: Example Sequence for PAA-initiated second type Re-authentication

4.5 Termination Phase

A procedure for explicitly terminating a PANA session can be initiated either from PaC (i.e., disconnect indication) or from PAA (i.e., session revocation). The PANA-Termination-Request and the PANA-Termination-Answer message exchanges are used for disconnect indication and session revocation procedures.

The reason for termination is indicated in the Termination-Cause AVP.

When there is an established PANA SA established between the PaC and the PAA, all messages exchanged during the termination phase MUST be protected with a MAC AVP. When the sender of the PANA-Termination-Request receives a valid acknowledgment, all states maintained for the PANA session MUST be deleted immediately.

```
PaC      PAA     Message(tseq,rseq)[AVPs]
-----------------------------------------------------
   ----->        PANA-Termination-Request(q,p)[MAC]
   <-----        PANA-Termination-Answer(p+1,q)[MAC]
```

Figure 7: Example Sequence for Session Termination

4.6 Illustration of a Complete Message Sequence

A complete PANA message sequence is illustrated in Figure 8. The example assumes the following scenario:

- PaC multicasts PANA-PAA-Discover message
- The ISNs used by the PAA and the PaC are x and y, respectively.
- A single EAP sequence is used in authentication phase.
- An EAP authentication method with a single round trip is used in the EAP sequence.
- The EAP authentication method derives keys. The PANA SA is established based on the unique and fresh session key provided by the EAP method.
- After PANA SA is established, all messages are integrity and replay protected with the MAC AVP.
- Re-authentication based on the PANA-Reauth-Request/ PANA-Reauth-Answer exchange is performed.
- The PANA session is terminated as a result of the PANA-

Termination-Request indication from the PaC.


```
PaC      PAA  Message(tseq,rseq)[AVPs]
-----------------------------------------------------
// Discovery and initial handshake phase
   ----->     PANA-PAA-Discover (0,0)
   <-----     PANA-Start-Request (x,0)[Cookie]
   ----->     PANA-Start-Request-Answer (y,x)[Cookie]

// Authentication phase
   <-----     PANA-Auth-Request(x+1,y)[EAP]
   ----->     PANA-Auth-Answer(y+1,x+1)[EAP]
   <-----     PANA-Auth-Request(x+2,y+1)[EAP]
   ----->     PANA-Auth-Answer(y+2,x+2)[EAP]
   <-----     PANA-Bind-Request(x+3,y+2)
                [EAP{Success}, Device-Id, Lifetime, Protection-Cap., MAC]
   ----->     PANA-Bind-Answer(y+3,x+3)[Device-Id, MAC]

// Re-authentication
   <-----     PANA-Reauth-Request (x+4,y+3)[MAC]
   ----->     PANA-Reauth-Answer (y+4,x+4)[MAC]

// Termination phase
   ----->     PANA-Termination-Request(y+5,x+4)[MAC]
   <-----     PANA-Termination-Answer (x+5,y+5)[MAC]
```

Figure 8: A Complete Message Sequence

Another PANA message sequence is illustrated in Figure 9. The example assumes the following scenario:

- PaC multicasts PANA-PAA-Discover message
- The ISNs used by the PAA and the PaC are x and y, respectively.

- PAA offers NAP and ISP separate authentication, as well as a choice of ISP from "ISP1" and "ISP2". PaC accepts the offer from PAA, with choosing "ISP1" as the ISP.

- An EAP sequence for NAP authentication and an EAP sequence for ISP authentication is performed in this order in authentication phase.

- An EAP authentication method with a single round trip is used in the EAP sequence.

- The EAP authentication methods derive keys. Once the two EAP authenticatioins are successful, the PANA_MAC_KEY is derived from the two AAA-Keys.

- After PANA SA is established, all messages are integrity and replay protected with the MAC AVP.

- Re-authentication based on the PANA-Reauth-Request/ PANA-Reauth-Answer exchange is performed.

- Re-authentication and termination phase are not shown.

```
PaC      PAA  Message(tseq,rseq)[AVPs]
-----------------------------------------------------
// Discovery and initial handshake phase
   ----->    PANA-PAA-Discover (0,0)
   <-----     PANA-Start-Request (x,0)            // S-flag set
              [Cookie, ISP-Information("ISP1"),
               ISP-Information("ISP2"),
               NAP-Information("MyNAP")]
   ----->     PANA-Start-Request-Answer (y,x)     // S-flag set
              [Cookie, ISP-Information("ISP1")]   // PaC chooses "ISP1"

// Authentication phase
   <-----     PANA-Auth-Request(x+1,y)[EAP]       // NAP authentication
                                                  // S- and N-flags set
   ----->     PANA-Auth-Answer(y+1,x+1)[EAP]      // S- and N-flags set
   <-----     PANA-Auth-Request(x+2,y+1)[EAP]     // S- and N-flags set
   ----->     PANA-Auth-Answer(y+2,x+2)[EAP]      // S- and N-flags set
```

```
        <-----     PANA-FirstAuth-End-Request(x+3,y+2) // S- and N-flags set
                     [EAP{Success}, Key-Id, MAC]
        ----->     PANA-FirstAuth-End-Answer(y+3,x+3)  // S- and N-flags set
                     [Key-Id, MAC]
        <-----     PANA-Auth-Request(x+3,y+4)[EAP, MAC]// ISP authentication
                                                       // S-flag set
        ----->     PANA-Auth-Answer(y+4,x+4)[EAP, MAC] // S-flag set
        <-----     PANA-Auth-Request(x+4,y+5)[EAP, MAC]// S-flag set
        ----->     PANA-Auth-Answer(y+5,x+5)[EAP, MAC] // S-flag set
        <-----     PANA-Bind-Request(x+5,y+6)          // S-flag set
                     [EAP{Success}, Device-Id, Key-Id,
                      Lifetime, Protection-Cap., PPAC, MAC]
        ----->     PANA-Bind-Answer(y+6,x+5)           // S-flag set
                     [Device-Id, Key-Id, PPAC, MAC]
```

Figure 9: A Complete Message Sequence for NAP and ISP Separate Authentications

4.7 Device ID Choice

The device identifiers used in the context of PANA can be an IP address, a MAC address, or an identifier that is not carried on data packets but has local significance in identifying a connected host (e.g., circuit ID). The last type of identifiers are commonly used in physically secured point-to-point links where MAC addresses are not available.

It is assumed that PAA knows the link type and the security mechanisms being provided or required on the access network (e.g., based on physical security, link-layer ciphers enabled before or after PANA, or IPsec). Based on that information, the PAA can decide what type of device ID will be used when running PANA with the client. When IPsec-based mechanism [I-D.ietf-pana-ipsec] is the choice of access control, PAA SHOULD provide an IP address as device ID, and expect the PaC to provide its IP address in return. In case IPsec is not used, MAC addresses are used as device IDs when available. If non-IPsec access control is enabled, and a MAC address

is not available, device ID exchange does not occur within PANA. Instead, peers rely on lower-layers to provide locally-significant identifiers along with received PANA packets.

4.8 Session Lifetime

The authentication phase determines the PANA session lifetime when the network access authorization succeeds. The Session-Lifetime AVP MAY be optionally included in the PANA-Bind-Request message to inform PaC about the valid lifetime of the PANA session. It MUST be ignored when included in other PANA messages. When there are multiple EAP authentication taking place, this AVP SHOULD be included after the final authentication.

The lifetime is a non-negotiable parameter that can be used by PaC to manage PANA-related state. PaC does not have to perform any actions when the lifetime expires, other than optionally purging local state. PAA SHOULD initiate EAP authentication before the current session lifetime expires.

PaC and PAA MAY optionally rely on lower-layer indications to expedite the detection of a disconnected peer. Availability and reliability of such indications depend on the specific access technologies. PANA peer can use PANA-Reauth-Request message to verify the disconnection before taking an action.

The session lifetime parameter is not related to the transmission of PANA-Reauth-Request messages. These messages can be used for asynchronously verifying the liveness of the peer and enabling

mobility optimizations. The decision to send PANA-Reauth-Request message is taken locally and does not require coordination between the peers.

When separate EAP authentications are performed for ISP and NAP in a single PANA session, it is possible that different authorization lifetime values are associated with the two authentications. In this case, the smaller authorization lifetime value MUST be used for calculating the PANA Session-Lifetime value. As a result, when EAP-based re-authentication occurs, both NAP and ISP authentications will be performed in the same re-authentication procedure.

4.9 Mobility Handling

A mobile PaC's AAA performance can be enhanced by deploying a context-transfer-based mechanism, where some session attributes are transferred from the previous PAA to the current one in order to avoid performing a full EAP authentication (reactive approach). Additional mechanisms that are based on the proactive AAA state establishment at one or more candidate PAAs may be developed in the future [I-D.irtf-aaaarch-handoff]. The details of a context-transfer-based mechanism is provided in this section.

Upon changing its point of attachment, a PaC that wants to quickly resume its ongoing PANA session without running EAP MAY send its unexpired PANA session identifier in its PANA-Start-Answer message. Along with the Session-Id AVP, MAC and Nonce AVPs MUST be included in this message. Nonce AVP carries a randomly chosen value (PaC_Nonce), and MAC AVP is computed by using the PANA_MAC_Key shared between the PaC and its previous PAA that has an unexpired PANA session with the PaC. This action signals PaC's desire to perform the mobility optimization. In the absence of Session-Id AVP in this message, PANA session takes its usual course (i.e., EAP-based authentication is performed).

If PAA receives a session identifier in the PANA-Start-Answer message, and it is configured to enable this optimization, it SHOULD retrieve the PANA session attributes from the previous PAA. Current PAA determines the identity of the previous PAA by looking at the DiameterIdentity part of the PANA session identifier. The MAC AVP can only be verified by the previous PAA, therefore a copy of the PANA message SHOULD be provided to the previous PAA. The mechanism required to send a copy of the PANA-Start-Answer message from current PAA to the previous PAA, and retrieve the session attributes is outside the scope of PANA protocol. Seamoby Context Transfer Protocol [I-D.ietf-seamoby-ctp] might be useful for this purpose.

When the previous or current PAA is not configured to enable this

optimization, the current PAA can not retrieve the PANA session attributes, or the PANA session has already expired (i.e., session lifetime is zero), the PAA MUST send the PANA-Auth-Request message with a new session identifier and let the PANA exchange take its usual course. This action will engage EAP-based authentication and

create a fresh PANA session from scratch.

In case the current PAA can retrieve the on-going PANA session attributes from the previous PAA, the PANA session continues with a PANA-Bind exchange.

As part of the context transfer, an intermediate AAA-Key material is provided by the previous PAA to the current PAA.

```
AAA-Key-int = The first N bits of
              HMAC-SHA1(AAA-Key, DiameterIdentity | Session-ID)
```

In case there are two AAA-Keys generated from a NAP-ISP authentication, the AAA-Key-int computation is:

```
AAA-Key-int = The first N bits of
              HMAC-SHA1(AAA-Key1 | AAA-Key2, DiameterIdentity |
                        Session-ID)
```

The value of N depends on the integrity protection algorithm in use, i.e., N=160 for HMAC-SHA1. DiameterIdentity is the identifier of the current PAA. Session-ID is the identifier of the PaC's PANA session with the previous PAA.

The current PAA and PaC compute the new AAA-Key by using the nonce values and the AAA-Key-int. PAA_Nonce is the randomly chosen value that MUST be carried in a Nonce AVP in the PANA-Bind-Request message.

```
AAA-Key-new = The first N bits of
              HMAC-SHA1(AAA-Key-int, PaC_nonce | PAA_nonce)
```

New PANA_MAC_Key is computed based on the algorithm described in Section 4.1.5, by using the new AAA-Key and the new Session-ID assigned by the current PAA. The MAC AVP contained in the PANA-Bind-Request and PANA-Bind-Answer messages MUST be generated and verified by using the new PANA_MAC_Key. The Session-ID AVP MUST include a new session identifier assigned by the current PAA. A new PANA session is created upon successful completion of this exchange.

Note that correct operation of this optimization relies on many factors, including applicability of authorization state from one network attachment to another. [I-D.ietf-eap-keying] identifies this operation as "fast handoff" and provides deployment considerations.

Operators are recommended to take those guidelines into account when using this optimization in their networks.

4.10 Support for Separate EP

PANA allows PAA and EP to be separate entities. In this case, if data traffic protection needs to be initiated after successful PANA authentication phase, PaC needs to know the device identifier of EP(s) so that it is able to establish a security association with each EP to protect data traffic.

To this end, when a Protection-Capability AVP with either L2_PROTECTION or IPSEC_PROTECTION in the AVP payload is carried in a PANA-Bind-Request message and if there is an EP that has a different device identifier than that of the PAA, one or more EP-Device-Id AVPs MUST also be carried in the PANA-Bind-Request message. In this case, if one EP has the same device identifier as the PAA, an EP-Device-Id AVP that contains the device identifier of the EP (i.e., the PAA) MUST also be included in the PANA-Bind-Request.

Aside from provisioning EP, the same PAA-to-EP protocol MAY be used for triggering the PAA upon detecting the need to authenticate a new client.

5. PANA Security Association Establishment

When PANA is used over an already established secure channel, such as physically secured wires or ciphered link-layers, we can reasonably assume that man-in-the-middle attack or service theft is not possible [I-D.ietf-pana-threats-eval].

Anywhere else where there is no secure channel prior to PANA, the protocol needs to protect itself against such attacks. The device identifier that is used during the authentication needs to be verified at the end of the authentication to prevent service theft and DoS attacks. Additionally, a free loader should be prevented from spoofing data packets by using the device identifier of an already authorized legitimate client. Both of these requirements necessitate generation of a security association between the PaC and the PAA at the end of the authentication. This can only be done when the authentication method used can generate cryptographic keys. Use of secret keys can prevent attacks which would otherwise be very easy to launch by eavesdropping on and spoofing traffic over an insecure link.

PANA relies on EAP and the EAP methods to provide a session key in order to establish a PANA security association. An example of such a method is EAP-TLS [RFC2716], whereas EAP-MD5 [I-D.ietf-eap-rfc2284bis] is an example of a method that cannot create such keying material. The choice of EAP method becomes important, as discussed in the next section.

This keying material is already used within PANA during the final handshake. This handshake ensures that the device identifier that is bound to the PaC at the end of the authentication process is not coming from a man-in-the-middle, but from the legitimate PaC. Knowledge of the same keying material on both PaC and the PAA helps prove this. The other use of the keying material is discussed in [I-D.ietf-pana-framework].

6. Message Formats

This section defines message formats for PANA protocol.

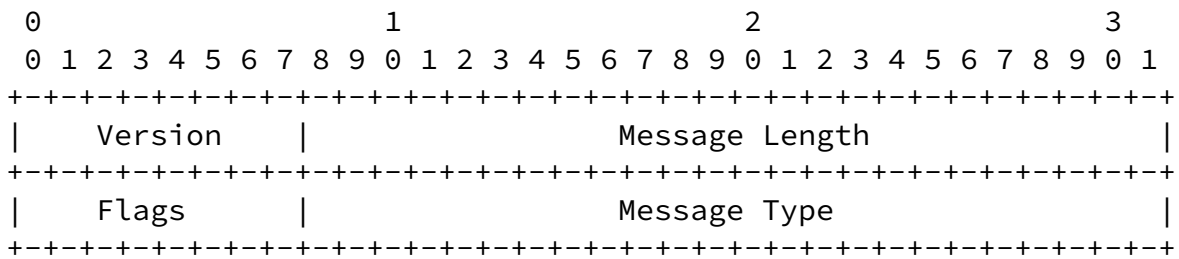
6.1 IP and UDP Headers

The Hop Limit (or TTL) field of the IP header MUST be set to 255. When a PANA-PAA-Discover message is multicast, IP destination address of the message is set to a well-known link-local multicast address (TBD). A PANA-PAA-Discover message MAY be unicast in some cases as specified in Section 4.2. Any other PANA packet is unicasted between the PaC and the PAA. The source and destination addresses SHOULD be set to the addresses on the interfaces from which the message will be sent and received, respectively.

When the PANA packet is sent in response to a request, the UDP source and destination ports of the response packet MUST be copied from the destination and source ports of the request packet, respectively. The destination port of an unsolicited PANA packet MUST be set to an assigned value (TBD), and the source port MUST be set to a value chosen by the sender.

6.2 PANA Header

A summary of the PANA header format is shown below. The fields are transmitted in network byte order.



```
|                  Transmitted Sequence Number                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Received Sequence Number                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  AVPs ...
+-+-+-+-+-+-+-+-+-+-+-+-
```

Version

This Version field MUST be set to 1 to indicate PANA Version 1.

Message Length

The Message Length field is three octets and indicates the length of the PANA message including the header fields.

Flags

The Flags field is eight bits. The following bits are assigned:

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|R r r r S N r r|
+-+-+-+-+-+-+-+-+
```

R(equest)

If set, the message is a request. If cleared, the message is an answer.

S(eparate)

When the S-flag is set in a PANA-Start-Request message it indicates that PAA is willing to offer separate EAP authentications for NAP and ISP. When the S-flag is set in a PANA-Start-Answer message it indicates that PaC accepts on performing separate EAP authentications for NAP and ISP. When the S-flag is set in a PANA-Auth-Request/Answer, PANA-FirstAuth-End-Request/Answer and PANA-Bind-Request/Answer

messages it indicates that separate authentications are being performed in the authentication phase.

N(AP authentication)

When the N-flag is set in a PANA-Auth-Request message, it indicates that PAA is performing NAP authentication. When the N-flag is unset in a PANA-Auth-Request message, it indicates that PAA is performing ISP authentication. The N-flag MUST NOT be set when S-flag is not set.

r(eserved)

these flag bits are reserved for future use, and MUST be set to zero, and ignored by the receiver.

Message Type

The Message Type field is three octets, and is used in order to communicate the message type with the message. The 24-bit address space is managed by IANA [ianaweb]. PANA uses its own address space for this field.

Transmitted Sequence Number

The Transmitted Sequence Number field contains the monotonically increasing 32 bit sequence number that the message sender increments every time a new PANA message is sent.

Received Sequence Number

The Received Sequence Number field contains the 32 bit transmitted sequence number that the message sender has last received from its peer.

AVPs

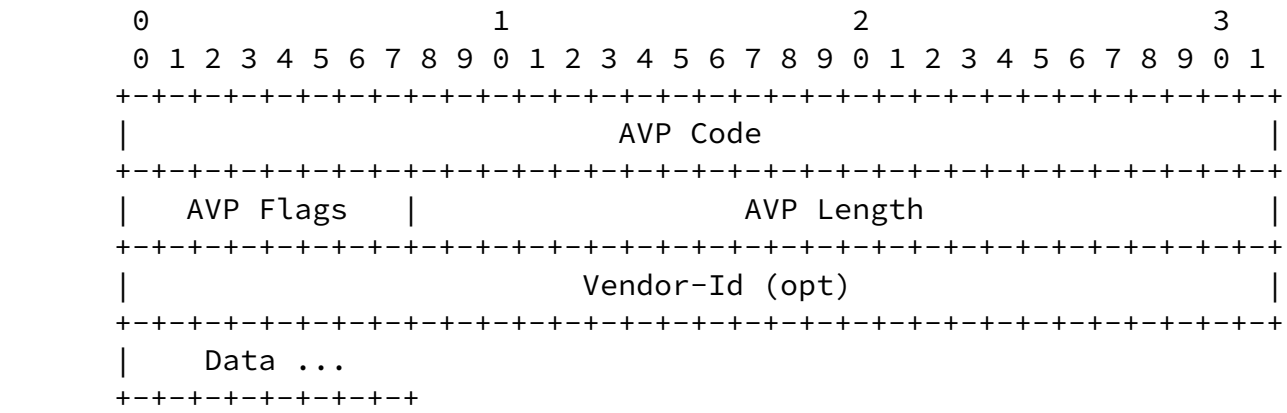
AVPs are a method of encapsulating information relevant to the PANA message. See section Section 6.3 for more information on

AVPs.

6.3 AVP Header

Each AVP of type OctetString MUST be padded to align on a 32-bit boundary, while other AVP types align naturally. A number of zero-valued bytes are added to the end of the AVP Data field till a word boundary is reached. The length of the padding is not reflected in the AVP Length field [RFC3588].

The fields in the AVP header MUST be sent in network byte order. The format of the header is:

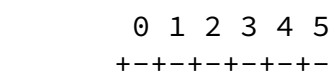


AVP Code

The AVP Code, combined with the Vendor-Id field, identifies the attribute uniquely. AVP numbers are allocated by IANA [ianaweb]. PANA uses its own address space for this field although some of the AVP formats are borrowed from Diameter protocol [RFC3588].

AVP Flags

The AVP Flags field is eight bits. The following bits are assigned:



```
|V M r r r r r r|
+-+-+-+-+-+-+-+-+
```

M(andatory)

The 'M' Bit, known as the Mandatory bit, indicates whether support of the AVP is required.

V(endor)

The 'V' bit, known as the Vendor-Specific bit, indicates whether the optional Vendor-Id field is present in the AVP header.

r(eserved)

these flag bits are reserved for future use, and MUST be set to zero, and ignored by the receiver.

AVP Length

The AVP Length field is three octets, and indicates the number of octets in this AVP including the AVP Code, AVP Length, AVP Flags, and the AVP data

Vendor-Id

The Vendor-Id field is present if the 'V' bit is set in the AVP Flags field. The optional four-octet Vendor-Id field contains the uniquely assigned id value, encoded in network byte order. Any vendor wishing to implement a vendor-specific PANA AVP MUST use their own Vendor-Id along with their privately managed AVP address space, guaranteeing that they will not collide with any other vendor's vendor-specific AVP(s), nor with future IETF

applications.

Data

The Data field is zero or more octets and contains information specific to the Attribute. The format and length of the Data field is determined by the AVP Code and AVP Length fields.

6.4 PANA Messages

Figure 10 lists all PANA messages defined in this document

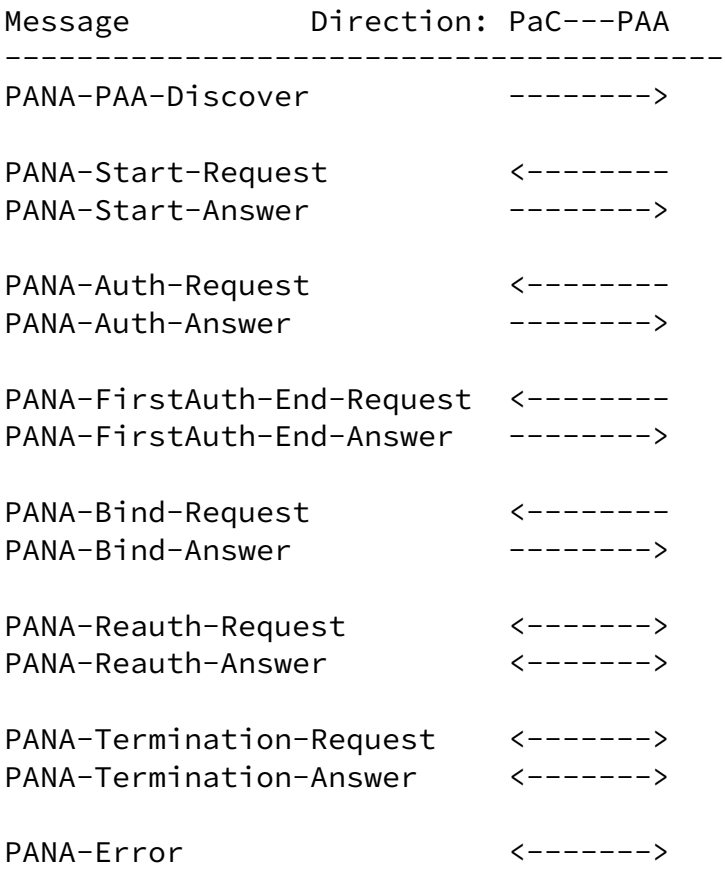


Figure 10: PANA Message Overview

6.4.1 Message Specifications

Every PANA message MUST include a corresponding ABNF [RFC2234] specification found in [RFC3588]. Note that PANA messages have a different header format compared to Diameter.

Example:

```
message ::= < PANA-Header: <Message type>, [REQ] [SEP] >
          * [ AVP ]
```

6.4.2 PANA-PAA-Discover (PDI)

The PANA-PAA-Discover (PDI) message is used to discover the address of PAA(s). Both sequence numbers in this message are set to zero (0).

```
PANA-PAA-Discover ::= < PANA-Header: 1 >
          0*1 < Session-Id >
           *  [ AVP ]
```

6.4.3 PANA-Start-Request (PSR)

PANA-Start-Request (PSR) is sent by the PAA to the PaC. The PAA sets the transmission sequence number to an initial random value. The received sequence number is set to zero (0).

```
PANA-Start-Request ::= < PANA-Header: 2, REQ [SEP] >
              [ Cookie ]
              [ EAP-Payload ]
              [ NAP-Information ]
           *  [ ISP-Information ]
              [ Protection-Capability]
              [ PPAC ]
           *  [ AVP ]
```

6.4.4 PANA-Start-Answer (PSA)

PANA-Start-Answer (PSA) is sent by the PaC to the PAA in response to a PANA-Start-Request message. The PANA_start message transmission sequence number field is copied to the received sequence number field. The transmission sequence number is set to initial random value.

```
PANA-Start-Answer ::= < PANA-Header: 2 [SEP] >
              [ Session-Id ]
              [ Cookie ]
              [ Nonce ]
              [ EAP-Payload ]
              [ ISP-Information ]
           *  [ AVP ]
```

6.4.5 PANA-Auth-Request (PAR)

PANA-Auth-Request (PAR) is sent by the PAA to the PaC.

```
      PANA-Auth-Request ::= < PANA-Header: 3, REQ [SEP] [NAP] >
                            < Session-Id >
                            < EAP-Payload >
                       *    [ AVP ]
                      0*1   < MAC >
```

(Both NAP-Information and ISP-Information MUST NOT be included at the same time)

6.4.6 PANA-Auth-Answer (PAN)

PANA-Auth-Answer (PAN) is sent by the PaC to the PAA in response to a PANA-Auth-Request message.

```
      PANA-Auth-Answer ::= < PANA-Header: 3 [SEP] [NAP] >
                           < Session-Id >
                           < EAP-Payload >
                      *    [ AVP ]
                     0*1   < MAC >
```

6.4.7 PANA-Bind-Request (PBR)

PANA-Bind-Request (PBR) is sent by the PAA to the PaC.

```
      PANA-Bind-Request ::= < PANA-Header: 4, REQ [SEP] [NAP] >
                            < Session-Id >
                            { Result-Code }
                            { PPAC }
                            [ EAP-Payload ]
                            [ Device-Id ]
                            [ Session-Lifetime ]
                            [ Protection-Capability ]
                            [ Key-Id ]
                            [ Nonce ]
                       *    [ EP-Device-Id ]
                       *    [ AVP ]
                      0*1   < MAC >
```

6.4.8 PANA-Bind-Answer (PBA)

PANA-Bind-Answer (PBA) is sent by the PaC to the PAA in response to a PANA-Result-Request message.


```
PANA-Bind-Answer ::= < PANA-Header: 4 [,SEP] [NAP] >
                     < Session-Id >
                     { Result-Code }
                     [ PPAC ]
                     [ Device-Id ]
                     [ Key-Id ]
                   *  [ AVP ]
                  0*1 < MAC >
```

6.4.9 PANA-Reauth-Request (PRAR)

PANA-Reauth-Request (PRAR) is either sent by the PaC or the PAA.

```
PANA-Reauth-Request ::= < PANA-Header: 5, REQ >
                        < Session-Id >
                        [ Device-Id ]
                      *  [ AVP ]
                     0*1 < MAC >
```

6.4.10 PANA-Reauth-Answer (PRAA)

PANA-Reauth-Answer (PRAA) is sent in response to a PANA-Reauth-Request.

```
PANA-Reauth-Answer ::= < PANA-Header: 5 >
                       < Session-Id >
                       [ Device-Id ]
                     *  [ AVP ]
                    0*1 < MAC >
```

6.4.11 PANA-Termination-Request (PTR)

PANA-Termination-Request (PTR) is sent either by the PaC or the PAA.

```
PANA-Termination-Request ::= < PANA-Header: 6, REQ >
                             < Session-Id >
                             < Termination-Cause >
                           *  [ AVP ]
                          0*1 < MAC >
```

6.4.12 PANA-Termination-Answer (PTA)

PANA-Termination-Answer (PTA) is sent either by the PaC or the PAA in response to PANA-Termination-Request.


```
PANA-Termination-Answer ::= < PANA-Header: 6 >
                            < Session-Id >
                          *  [ AVP ]
                         0*1 < MAC >
```

6.4.13 PANA-Error (PER)

PANA-Error is sent either by the PaC or the PAA.

```
PANA-Error ::= < PANA-Header: 7 >
               < Session-Id >
               < Result-Code >
               { Failed-AVP }
             *  [ AVP ]
            0*1 < MAC >
```

6.4.14 PANA-FirstAuth-End-Request (PFER)

PANA-FirstAuth-End-Request (PFER) is sent by the PAA to the PaC.

```
PANA-FirstAuth-End-Request ::= < PANA-Header: 8, REQ [SEP] [NAP] >
                               < Session-Id >
                               < Device-Id >
                               { EAP-Payload }
                               { Result-Code }
                               [ Key-Id ]
                             *  [ AVP ]
```

```
                0*1 < MAC >
```

6.4.15 PANA-FirstAuth-End-Answer (PFEA)

PANA-FirstAuth-End-Answer (PFEA) is sent by the PaC to the PAA in response to a PANA-FirstAuth-End-Request message.

```
      PANA-FirstAuth-End-Answer ::= < PANA-Header: 8, REQ [SEP] [NAP] >
                    < Session-Id >
                    < Device-Id >
                    [ Key-Id ]
                *   [ AVP ]
                0*1 < MAC >
```

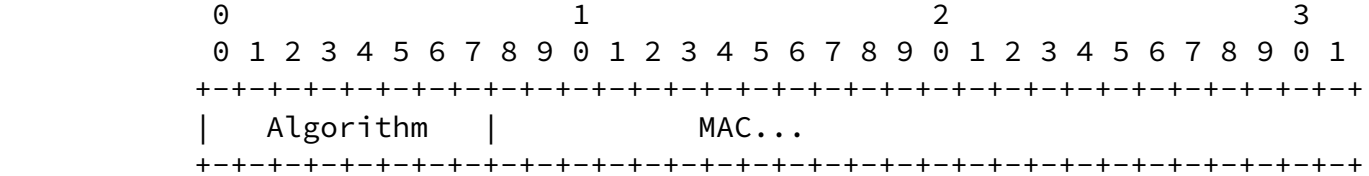
6.5 AVPs in PANA

Some of the used AVPs are defined in this document and some of them

are defined in other documents like [RFC3588]. PANA proposes to use the same name space with [RFC3588]. For temporary allocation, PANA uses AVP type numbers starting from 1024.

6.5.1 MAC AVP

The first octet (8 bits) of the MAC (Code 1024) AVP data contains the MAC algorithm type. Rest of the AVP data payload contains the MAC encoded in network byte order. The Algorithm 8 bit name space is managed by IANA [ianaweb]. The AVP length varies depending on the used algorithm.



Algorithm

```
              1             HMAC-SHA1 (20 bytes)
```

MAC

The Message Authentication Code is encoded in network byte order.

6.5.2 Device-Id AVP

The Device-Id AVP (Code 1025) is of Address type [RFC3588]. IPv4 and IPv6 addresses are encoded as specified in [RFC3588]. The content and format of data (including byte and bit ordering) for link-layer addresses is expected to be specified in specific documents that describe how IP operates over different link-layers. For instance, [RFC2464]. Address families other than that are defined for link-layer or IP addresses MUST NOT be used for this AVP.

6.5.3 Session-Id AVP

All messages pertaining to a specific PANA session MUST include a Session-Id AVP (Code 1026) which carries a PAA-assigned fix value throughout the lifetime of a session. When present, the Session-Id SHOULD appear immediately following the PANA header.

The Session-Id MUST be globally and eternally unique, as it is meant to identify a PANA Session without reference to any other information, and may be needed to correlate historical authentication information with accounting information. The PANA Session-Id AVP has

the same format as the Diameter Session-Id AVP [RFC3588].

6.5.4 Cookie AVP

The Cookie AVP (Code 1027) is of type OctetString. The data is opaque and the exact content is outside the scope of this protocol.

6.5.5 Protection-Capability AVP

The Protection-Capability AVP (Code 1028) is of type Unsigned32. The AVP data indicates the cryptographic data protection capability supported by the EPs. Below is a list of specified data protection capabilities:

0 L2_PROTECTION

1 IPSEC_PROTECTION

6.5.6 Termination-Cause AVP

The Termination-Cause AVP (Code 1029) is of type of type Enumerated, and is used to indicate the reason why a session was terminated on the access device. The AVP data is used as a collection of flags The following Termination-Cause AVP defined in [RFC3588] are used for PANA.

LOGOUT 1 (PaC -> PAA)

The client initiated a disconnect

ADMINISTRATIVE 4 (PAA -> Pac)

The client was not granted access, or was disconnected, due to administrative reasons, such as the receipt of a Abort-Session-Request message.

SESSION_TIMEOUT 8 (PAA -> PaC)

The session has timed out, and service has been terminated.

6.5.7 Result-Code AVP

The Result-Code AVP (AVP Code 1030) is of type Unsigned32 and indicates whether an EAP authentication was completed successfully or whether an error occurred. Here are Result-Code AVP values taken from [RFC3588] and adapted for PANA.

6.5.7.1 Authentication Results Codes

These result code values inform the PaC about the authentication and authorization result. The authentication result and authorization result can be different as described below, but only one result that corresponds to the one detected first is returned.

PANA_SUCCESS 2001

Both the authentication and authorization processes are successful.

PANA_AUTHENTICATION_REJECTED 4001

The authentication process failed. When this error is returned, the authorization process also fails.

PANA_AUTHORIZATION_REJECTED 5003

The authorization process failed. This error could occur when authorization is rejected by a AAA proxy or rejected locally by a PAA, even if the authentication procedure succeeds.

6.5.7.2 Protocol Error Result Codes

Protocol error result code values.

PANA_MESSAGE_UNSUPPORTED 3001

Error code from PAA to PaC or from PaC to PAA. Message type not recognized or supported.

PANA_UNABLE_TO_DELIVER 3002

Error code from PAA to PaC. PAA was unable to deliver the EAP payload to the authentication server.

PANA_INVALID_HDR_BITS 3008

Error code from PAA to PaC or from PaC to PAA. A message was received whose bits in the PANA header were either set to an invalid combination, or to a value that is inconsistent with the message type's definition.

PANA_INVALID_AVP_BITS 3009

Error code from PAA to PaC or from PaC to PAA. A message was

received that included an AVP whose flag bits are set to an unrecognized value, or that is inconsistent with the AVP's definition.

PANA_AVP_UNSUPPORTED 5001

Error code from PAA to PaC or from PaC to PAA. The received message contained an AVP that is not recognized or supported and was marked with the Mandatory bit. A PANA message with this error MUST contain one or more Failed-AVP AVP containing the AVPs that caused the failure.

PANA_UNKNOWN_SESSION_ID 5002

Error code from PAA to PaC or from PaC to PAA. The message contained an unknown Session-Id. PAA MUST NOT send this error result code value to PaC if PaC sent an unknown Session-Id in the PANA-Start-Answer message (session resumption).

PANA_INVALID_AVP_VALUE 5004

Error code from PAA to PaC or from PaC to PAA. The message contained an AVP with an invalid value in its data portion. A PANA message indicating this error MUST include the offending AVPs within a Failed-AVP AVP.

PANA_MISSING_AVP 5005

Error code from PAA to PaC or from PaC to PAA. The message did not contain an AVP that is required by the message type definition. If this value is sent in the Result-Code AVP, a Failed-AVP AVP SHOULD be included in the message. The Failed-AVP AVP MUST contain an example of the missing AVP complete with the Vendor-Id if applicable. The value field of the missing AVP should be of correct minimum length and contain zeroes.

PANA_RESOURCES_EXCEEDED 5006

Error code from PAA to PaC. A message was received that cannot be authorized because the client has already expended allowed resources. An example of this error condition is a client that is restricted to one PANA session and attempts to establish a second session.

PANA_CONTRADICTING_AVPS 5007

Error code from PAA to PaC. The PAA has detected AVPs in the message that contradicted each other, and is not willing to provide service to the client. One or more Failed-AVP AVPs MUST be present, containing the AVPs that contradicted each other.

PANA_AVP_NOT_ALLOWED 5008

Error code from PAA to PaC or from PaC to PAA. A message was received with an AVP that MUST NOT be present. The Failed-AVP AVP MUST be included and contain a copy of the offending AVP.

PANA_AVP_OCCURS_TOO_MANY_TIMES 5009

Error code from PAA to PaC or from PaC to PAA. A message was received that included an AVP that appeared more often than permitted in the message definition. The Failed-AVP AVP MUST be included and contain a copy of the first instance of the offending AVP that exceeded the maximum number of occurrences.

PANA_UNSUPPORTED_VERSION 5011

Error code from PAA to PaC or from PaC to PAA. This error is returned when a message was received, whose version number is unsupported.

PANA_UNABLE_TO_COMPLY 5012

This error is returned when a request is rejected for unspecified reasons. For example, when an EAP authentication fails at an EAP pass-through authenticator without passing an EAP-Failure message to the PAA, a Result-Code AVP with this error code is carried in PANA-Error message.

PANA_INVALID_AVP_LENGTH 5014

Error code from PAA to PaC or from PaC to PAA. The message contained an AVP with an invalid length. The PANA-Error message indicating this error MUST include the offending AVPs within a Failed-AVP AVP.

PANA_INVALID_MESSAGE_LENGTH 5015

Error code from PAA to PaC or from PaC to PAA. This error is returned when a message is received with an invalid message length.

PANA_PROTECTION_CAPABILITY_UNSUPPORTED 5016

Error code from PaC to PAA. This error is returned when the PaC receives a PANA-Bind-Request is received with an Protection-Capability AVP and a valid MAC AVP but does not support the protection capability specified in the Protection-Capability AVP.

PANA_PPAC_CAPABILITY_UNSUPPORTED 5017

Error code from PaC to PAA. This error is returned in a PANA-Bind-Answer message when there is no match between the list of PPAC methods offered by the PAA and the ones available on the PaC.

6.5.8 EAP-Payload AVP

The EAP-Payload AVP (AVP Code 1031) is of type OctetString and is used to encapsulate the actual EAP packet that is being exchanged between the EAP peer and the EAP authenticator.

6.5.9 Session-Lifetime AVP

The Session-Lifetime AVP (Code 1032) data is of type Unsigned32. It contains the number of seconds remaining before the current session is considered expired.

6.5.10 Failed-AVP AVP

The Failed-AVP AVP (AVP Code 1033) is of type Grouped and provides debugging information in cases where a request is rejected or not fully processed due to erroneous information in a specific AVP. The format of the Failed-AVP AVP is defined in [RFC3588].

6.5.11 NAP-Information AVP

The NAP-Information AVP (AVP Code: 1034) is of type Grouped, and contains zero or one Provider-Identifier AVP which carries the

identifier of the NAP and one Provider-Name AVP which carries the name of the NAP. Its Data field has the following ABNF grammar:

```
NAP-Information ::= < AVP Header: 1034 >
                    0*1 { Provider-Identifier }
                        { Provider-Name }
                     *  [ AVP ]
```


6.5.12 ISP-Information AVP

The ISP-Information AVP (AVP Code: 1035) is of type Grouped, and contains zero or one Provider-Identifier AVP which carries the identifier of the ISP and one Provider-Name AVP which carries the name of the ISP. Its Data field has the following ABNF grammar:

```
ISP-Information ::= < AVP Header: 1035 >
                    0*1 { Provider-Identifier }
                        { Provider-Name }
                     *  [ AVP ]
```

6.5.13 Provider-Identifier AVP

The Provider-Identifier AVP (AVP Code: 1036) is of type Unsigned32, and contains an IANA assigned "SMI Network Management Private Enterprise Codes" [ianaweb] value, encoded in network byte order.

6.5.14 Provider-Name AVP

The Provider-Name AVP (AVP Code: 1037) is of type UTF8String, and contains the UTF8-encoded name of the provider.

6.5.15 EP-Device-Id AVP

The EP-Device-Id AVP (AVP Code: 1038) contains the device identifier of an EP. The payload format of the EP-Device-Id AVP is the same as that of the Device-Id AVP (see See section Section 6.5.2).

6.5.16 Key-Id AVP

The Key-Id AVP (AVP Code: 1039) is of type Integer32, and contains an AAA-Key identifier. The AAA-Key identifier is assigned by PAA and MUST be unique within the PANA session.

6.5.17 Post-PANA-Address-Configuration (PPAC) AVP

The data field of PPAC AVP (Code 1040) is of type Unsigned32. The AVP data is used to carry a set of flags which maps to various IP address configuration methods. When sent by the PAA, the AVP MUST have at least one of the flags set, and MAY have more than one set. When sent by the PaC, only one of the flags MUST be set.

The format of the AVP data is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
```



```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|N|D|A|T|I|                   Reserved                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

PPAC Flags

N (No configuration)

The PaC does not have to (if sent by PAA) or will not (if sent by PaC) configure a new IP address after PANA.

D (DHCP)

The PaC can (if sent by PAA) or will (if sent by PaC) use DHCP [RFC2131][RFC3315] to configure a new IP address after PANA.

A (stateless autoconfiguration)

The PaC can/will use stateless IPv6 address autoconfiguration [RFC2462] to configure a new IP address after PANA.

T (DHCP with IPsec tunnel mode)

The PaC can/will use [RFC3456] to configure a new IP address after PANA.

I (IKEv2)

The PaC can/will use [I-D.ietf-ipsec-ikev2] to configure a new IP address after PANA.

Reserved

These flag bits are reserved for future use, and MUST be set to zero, and ignored by the receiver.

Unless the N-flag is set, the PaC MUST configure a new IP address using one of the methods indicated by the other flags. Refer to [I-D.ietf-pana-framework] for a detailed discussion on when these methods can be used.

6.5.18 Nonce AVP

The Nonce AVP (Code 1041) is of type OctetString. The data contains a randomly generated value in opaque format. The data length MUST be between 8 and 256 bytes inclusive.

6.6 AVP Occurrence Table

The following tables lists the AVPs used in this document, and specifies in which PANA messages they MAY, or MAY NOT be present.

The table uses the following symbols:

0 The AVP MUST NOT be present in the message.

0+ Zero or more instances of the AVP MAY be present in the message.

0-1 Zero or one instance of the AVP MAY be present in the message. It is considered an error if there are more than one instance of the AVP.

1 One instance of the AVP MUST be present in the message.

1+ At least one instance of the AVP MUST be present in the message.

	Message Type						
Attribute Name	PSR	PSA	PAR	PAN	PBR	PBA	PDI
Result-Code	0	0	0	0	1	1	0
Session-Id	0	0-1	1	1	1	1	0-1
Termination-Cause	0	0	0	0	0	0	0
EAP-Payload	0-1	0-1	1	1	0-1	0	0
MAC	0	0-1	0-1	0-1	0-1	0-1	0
Nonce	0	0-1	0	0	0-1	0	0
Device-Id	0	0	0	0	0-1	0-1	0
Cookie	0-1	0-1	0	0	0	0	0
Protection-Cap.	0-1	0	0	0	0-1	0	0
PPAC	0-1	0	0	0	1	0-1	0
Session-Lifetime	0	0	0	0	0-1	0	0
Failed-AVP	0	0	0	0	0	0	0
ISP-Information	0+	0-1	0	0	0	0	0
NAP-Information	0-1	0	0	0	0	0	0
EP-Device-Id	0	0	0	0	0+	0	0
Key-Id	0	0	0	0	0-1	0-1	0

Figure 11: AVP Occurrence Table (1/2)

	Message Type						
Attribute Name	PRAR	PRAA	PTR	PTA	PER	PFER	PFEA
Result-Code	0	0	0	0	1	1	0
Session-Id	1	1	1	1	1	1	1
Termination-Cause	0	0	1	0	0	0	0
EAP-Payload	0	0	0	0	0	1	0
MAC	0-1	0-1	0-1	0-1	0-1	0-1	0-1

Nonce	0	0	0	0	0	0	0	
Device-Id	0-1	0-1	0	0	0	0	0	
Cookie	0	0	0	0	0	0	0	
Protection-Cap.	0	0	0	0	0	0	0	
PPAC	0	0	0	0	0	0	0	
Session-Lifetime	0	0	0	0	0	0	0	
Failed-AVP	0	0	0	0	1	0	0	
ISP-Information	0	0	0	0	0	0	0	
NAP-Information	0	0	0	0	0	0	0	
EP-Device-Id	0	0	0	0	0	0	0	
Key-Id	0	0	0	0	0	0-1	0-1	

Figure 12: AVP Occurrence Table (2/2)

7. PANA Protocol Message Retransmissions

The PANA protocol provides retransmissions for all the message exchanges except PANA-Auth-Request/Answer. PANA-Auth-Request

messages carry EAP requests which are retransmitted by the EAP protocol entities when needed. The messages that need PANA-level retransmissions are listed below:

```
PANA-PAA-Discover (PDI)
PANA-Start-Request (PSR)*
PANA-Start-Answer (PSA)**
PANA-Bind-Request (PBR)
PANA-FirstAuth-End-Request (PFER)
PANA-Reauth-Request (PRAR)
PANA-Termination-Request (PTR)

*)  PSR that carries a Cookie AVP is not retransmitted.
**) PSA that does not carry a Cookie AVP is not retransmitted.
```

The PDI and PSA messages are always sent by the PaC. PBR is sent by PAA. The last two messages, PRAR and PTR are sent either by PaC or PAA.

The rule is that the sender of the request message retransmits the request if the corresponding answer is not received in time. Answer messages are sent as answers to the request messages, not based on a timer. Exception to this rule is the PSA message. Because of the stateless nature of the PAA in the beginning PaC provides retransmission also for the PSA message. PANA-Error messages MUST not be retransmitted. See Section 4.1.8 for more details of PANA error handling.

PANA retransmission timers are based on the model used in DHCPv6 [RFC3315]. Variables used here are also borrowed from this specification. PANA is a request response like protocol. The message exchange terminates when either the request sender successfully receives the appropriate answer, or when the message exchange is considered to have failed according to the retransmission mechanism described below.

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

RT	Retransmission timeout
IRT	Initial retransmission time
MRC	Maximum retransmission count

MRT Maximum retransmission time

MRD Maximum retransmission duration

RAND Randomization factor

With each message transmission or retransmission, the sender sets RT according to the rules given below. If RT expires before the message exchange terminates, the sender recomputes 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 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 transmission is based on IRT:

```
RT = IRT + RAND*IRT
```

RT for each subsequent message transmission is based on the previous value of RT:

```
RT = 2*RTprev + 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 = MRT + 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.

MRD specifies an upper bound on the length of time a sender may retransmit a message. Unless MRD is zero, the message exchange fails once MRD seconds have elapsed since the client first transmitted the message.

If both MRC and MRD are non-zero, the message exchange fails whenever either of the conditions specified in the previous two paragraphs are

met.

If both MRC and MRD are zero, the client continues to transmit the message until it receives a response.

7.1 Transmission and Retransmission Parameters

This section presents a table of values used to describe the message retransmission behavior of request and PANA-Start-Answer messages marked with REQ_*. PANA-PAA-Discover message retransmission values are marked with PDI_*. The table shows default values.

Parameter	Default	Description
PDI_IRT	1 sec	Initial PDI timeout.
PDI_MRT	120 secs	Max PDI timeout value.
PDI_MRC	0	Configurable.
PDI_MRD	0	Configurable.
REQ_IRT	1 sec	Initial Request timeout.
REQ_MRT	30 secs	Max Request timeout value.
REQ_MRC	10	Max Request retry attempts.
REQ_MRD	0	Configurable.

So for example the first RT for the PBR message is calculated using REQ_IRT as the IRT:

RT = REQ_IRT + RAND*REQ_IRT

8. IANA Considerations

This section provides guidance to the Internet Assigned Numbers Authority (IANA) regarding registration of values related to the Diameter protocol, in accordance with BCP 26 [IANA]. The following policies are used here with the meanings defined in BCP 26: "Private Use", "First Come First Served", "Expert Review", "Specification Required", "IETF Consensus", "Standards Action".

This section explains the criteria to be used by the IANA for assignment of numbers within namespaces defined within this document.

For registration requests where a Designated Expert should be consulted, the responsible IESG area director should appoint the Designated Expert. For Designated Expert with Specification Required, the request is posted to the PANA WG mailing list (or, if it has been disbanded, a successor designated by the Area Director) for comment and review, and MUST include a pointer to a public specification. Before a period of 30 days has passed, the Designated Expert will either approve or deny the registration request and publish a notice of the decision to the PANA WG mailing list or its successor. A denial notice must be justified by an explanation and, in the cases where it is possible, concrete suggestions on how the request can be modified so as to become acceptable.

8.1 PANA UDP Port Number

TBD.

8.2 PANA Multicast Address

TBD.

8.3 PANA Header

8.3.1 Message Type

TBD.

8.3.2 Flags

TBD.

8.4 AVP Header

8.4.1 AVP Code

TBD.

8.4.2 Flags

TBD.

8.4.3 Vendor Id

TBD.

8.5 AVP Values

8.5.1 MAC AVP Values

TBD.

8.5.2 Device-Id AVP Values

TBD.

8.5.3 Protection-Capability AVP Values

TBD.

8.5.4 Result-Code AVP Values

TBD.

8.5.5 Termination-Cause AVP Values

TBD.

8.5.6 Provider-Identifier AVP Values

TBD.

8.5.7 Post-PANA-Address-Configuration AVP Values

TBD.

9. Security Considerations

The PANA protocol provides ordered delivery for EAP messages. If an EAP method that provides session keys is used, a PANA SA is created. The EAP Success/Failure message is one of the signaling messages which is integrity protected with this PANA SA. The PANA protocol does not provide security protection for the initial EAP message exchange. Integrity protection can only be provided after the PANA SA has been established. Thus, PANA re-authentication, revocation and disconnect notifications can be authenticated, integrity and replay protected. In certain environments (e.g., on a shared link) the EAP method selection is an important issue.

The PANA framework described in this document covers the discussion of different protocols which are of interest for a protocol between the PaC and the PAA (typically referred as the PANA protocol).

The PANA itself consists of a sequence of steps which are executed to complete the network access authentication procedure. Some of these

steps are optional.

The following execution steps have been identified as being relevant for PANA. They security considerations will be discussed in detail subsequently.

a) Discovery message exchange

In general it is difficult to prevent a vulnerabilities of the discovery protocol since the initial discovery are unsecured. To prevent very basic attacks an adversary should not be able to cause state creation with discovery messages at the PAA. This is prevented by re-using a cookie concept (see [RFC2522] which allows the responder to be stateless in the first message exchange. Because of the architectural assumptions made in PANA (i.e., the PAA is the on the same link as the PaC) the return-routability concept does not provide additional protection. Hence it is difficult to prevent this threat entirely. Furthermore it is not possible to shift heavy cryptographic operations to the PaC at the first few messages since the computational effort depends on the EAP method. The usage of client-puzzles as introduced by [jb99] is under investigation.

Resistance against blind DoS attacks (i.e., attacks by off-path adversaries) is achieved with sequence numbers and cookies.

Since PAA and PaC are supposed to be one IP hop away, a simple TTL check can prevent off-link attacks. Furthermore, additional filtering can be enabled on the EPs. An EP may be able to filter unauthorized PAA advertisements when they are received on the access

side of the network where only PaCs are connected.

In networks where lower-layers are not secured prior to running PANA, the capability discovery enabled through inclusion of Protection-Capability and Post-PANA-Address-Configuration AVPs in PANA-Start-Request message is susceptible to spoofing. Therefore, usage of these AVPs during the discovery phase in such insecure networks is NOT RECOMMENDED. The same AVPs are delivered via an integrity-protected PANA-Bind-Request upon successful authentication.

b) EAP over PANA message exchange

The EAP derived session key is used to create a PANA security association. Since the execution of an EAP method might require a large number of roundtrips and no other session key is available it is not possible to secure the EAP message exchange itself. Hence an adversary can both eavesdrop the EAP messages and is also able to inject arbitrary messages which might confuse both the EAP peer on PaC and the EAP authenticator or authentication server on the PAA. The threats caused by this ability heavily depend on the EAP state machine. Since especially the PAA is not allowed to discard packets and packets have to be stored or forwarded to an AAA infrastructure some risk of DoS attacks exists.

Eavesdropping EAP packets might cause problems when (a) the EAP method is weak and enables dictionary or replay attacks or even allows an adversary to learn the long-term password directly. Furthermore, if the optional EAP Identity payload is used then it allows the adversary to learn the identity of the PaC. In such a case a privacy problem is prevalent.

To prevent these threats, [I-D.ietf-pana-framework] suggests using proper EAP methods for particular environments. Depending on the usage environment an EAP authentication has to be used for example which supports user identity confidentiality, protection against dictionary attacks and session key establishment. It is therefore the responsibility of the network operators and end users to choose the proper EAP method.

PANA does not protect the EAP method exchange, but provides ordered delivery with sequence numbers. Sequence numbers and cookies provide resistance against blind DoS attacks.

c) PANA SA establishment

Once the EAP message authentication is finished a fresh and unique session key is available to the PaC and the PAA. This assumes that the EAP method allows session key derivation and that the generated

session key has a good quality. For further discussion about the importance of the session key generation refer to the next subsection (d) about compound authentication. The session key available for the PaC is established as part of the authentication and key exchange procedure of the selected EAP method. The PAA obtains the session

key via the AAA infrastructure (if used). Security issues raised with this session key transport are described in [I-D.ietf-eap-keying].

The establishment of a PANA SA is required in environments where no physical or link layer security is available. The PANA SA allows subsequently exchanged messages to experience cryptographic protection. For the current version of the document an integrity object (MAC AVP) is defined which supports data-origin authentication, replay protection based on sequence numbers and integrity protection based on a keyed message digest. Confidentiality protection is not provided. The session keys used for this object have to be provided by the EAP method. For this version of the document it is assumed that no negotiation of algorithms and parameters takes place. Instead HMAC-SHA1 is used by default. A different algorithm may be chosen by default in a future version of the PANA protocol specification. The used algorithm is indicated in the header of the Integrity object. To select the security association for signaling message protection the Session ID is conveyed. The keyed message digest included in the Integrity object will include all fields of the PANA signaling message including the sequence number field of the packet.

The protection of subsequent signaling messages prevents an adversary from acting as a man-in-the-middle adversary, from injecting packets, from replaying messages and from modifying the content of the exchanged packets. This prevents subsequently described threats.

If an entity (PAA or PaC) loses its state (especially the current sequence number) then the entire PANA protocol has to be restarted. No re-synchronization procedure is provided.

The lifetime of the PANA SA has to be bound to the AAA-authorized session lifetime with an additional tolerance period. Unless PANA state is updated by executing another EAP authentication, PANA SA is removed when the current session expires. The lifetime of the PANA SA has to be bound to the AAA-authorized session lifetime with an additional tolerance period. Unless PANA state is updated by executing another EAP authentication, PANA SA is removed when the current session expires.

d) Enabling weak legacy authentication methods in insecure networks

Some of the authentication methods are not strong enough to be used in insecure networks where attackers can easily eavesdrop and spoof on the link. They may not be able to produce much needed keying material either. An example would be using EAP-MD5 over wireless links. Use of such legacy methods can be enabled by carrying them over a secure channel. There are EAP methods which are specifically designed for this purpose, such as EAP-TTLS [I-D.ietf-pppext-eap-ttls],PEAP [I-D.josefsson-pppext-eap-tls-eap] or EAP-IKEv2 [I-D.tschofenig-eap-ikev2]. PANA can carry these EAP tunneling methods which can carry the legacy methods. PANA does not do anything special for this case. The EAP tunneling method will have to produce keying material for PANA SA when needed. There are certain MitM vulnerabilities with tunneling EAP methods [mitm]. Solving these problems is outside the scope of PANA. The compound authentication problem described in [I-D.puthenkulam-eap-binding] is likely to be solved in EAP itself rather than in PANA.

e) Device Identifier exchange

As part of the authorization procedure a Device Identifier has to be installed at the EP by the PAA. The PaC provides the Device Identifier information to the PAA secured with the PANA SA. Section 6.2.4 of [I-D.ietf-pana-threats-eval] describes a threat where an adversary modifies the Device Identifier to gain unauthorized access to the network.

The installation of the Device Identifier at the EP (independently whether the EP is co-located with the PAA or not) has to be accomplished in a secure manner. These threats are, however, not part of the PANA protocol itself since the protocol is not PANA specific.

f) Triggering a data protection protocol

Recent activities in the EAP working group try to create a common framework for key derivation which is described in [I-D.ietf-eap-keying]. This framework is also relevant for PANA in various ways. First, a PANA security association needs to be created. Additionally it might be necessary to trigger a protocol which allows link layer and network layer data protection to be established. As an example see Section 1 of [I-D.ietf-eap-keying] with [802.11i] and [802.11] as an example. Furthermore, a derived session key might help to create the pre-requisites for network-layer protection (for example IPsec [I-D.ietf-pana-ipsec]).

As motivated in Section 6.4 of [I-D.ietf-pana-threats-eval] it might be necessary to establish either a link layer or a network layer protection to prevent certain thefts in certain scenarios.

Threats specific to the establishment of a link layer or a network layer security association are outside the scope of PANA. The interested reader should refer to the relevant working groups such as IPsec or Midcom.

g) Liveness test

Network access authentication is done for a very specific purpose and often charging procedures are involved which allow restricting network resource usage based on some policies. In mobile environments it is always possible that an end host suddenly disconnects without transmitting a disconnect message. Operators are generally motivated to detect a disconnected end host as soon as possible in order to release resources (i.e., garbage collection). The PAA can remove per-session state information including installed security association, packet filters, etc.

Different procedures can be used for disconnect indication. PANA cannot assume link-layer disconnect indication. Hence this functionality has to be provided at a higher layer. With this version of the draft we suggest to apply the soft-state principle found at other protocols (such as RSVP). Soft-state means that session state is kept alive as long as refresh messages refresh the state. If no new refresh messages are provided then the state automatically times out and resources are released. This process includes stopping accounting procedures.

A PANA session is associated with a session lifetime. The session is terminated unless it is refreshed by a new round of EAP authentication before it expires. Therefore, at the latest a disconnected client can be detected when its lifetime expires. A disconnect may also be detected earlier by using PANA reauthentication messages. A request message can be generated by either PaC or PAA at any time and the peer must respond with an answer message. A successful round-trip of this exchange is a simple verification that the peer is alive. This test can be engaged when there is a possibility that the peer might have disconnected (e.g., after discontinuation of data traffic). Periodic use of this exchange as a keep-alive requires additional care as it might result in congestion and hence false alarms. This exchange is cryptographically protected when PANA SA is available in order to prevent threats associated with the abuse of this functionality.

h) Tear-Down message

The PANA protocol supports the ability for both the PaC and the PAA to transmit a tear-down message. This message causes state removal, a stop of the accounting procedure and removes the installed packet

filters.

It is obvious that such a message must be protected to prevent an adversary from deleting state information and thereby causing denial of service attacks.

i) Mobility optimization

The mobility optimization described in Section 4.9 involves the previous PAA providing a AAA-Key to the current PAA of the PaC. There are security risks stemming from potential compromise of PAAs. Compromise of the current PAA does not yield compromise of the previous PAA, as AAA-Key cannot be computed from a compromised AAA-Key-new. But a compromised previous PAA along with the intercepted nonce values leads to the compromise of AAA-Key-new. Operators should be aware of the potential risk of using this optimization. An operator can reduce the risk exposure by forcing the PaC to perform an EAP-based authentication immediately after the optimized PANA execution.

10. Open Issues and Change History

A list of open issues is maintained at [2].

Issues incorporated in PANA-01 June 2003: 1, 3, 10, 5, 6, 7 and 11.

Issues incorporated in PANA-02 October 2003: 8, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 30, 31, 32 and 33.

Issues incorporated in PANA-03 February 2004: 2, 16, 34, 35, 36, 38, 39, 40, 42, 43, 44, 50, 51 and 60.

Issues incorporated in PANA-04 May 2004: 28, 52, 53, 56, 57, 58, 59, 61, 62, 63, 64, 65, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80 and 83.

11. Acknowledgments

We would like to thank Jari Arkko, Mohan Parthasarathy, Julien Bournelle, Rafael Marin Lopez and all members of the PANA working group for their valuable comments to this document.

Normative References

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

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC1982] Elz, R. and R. Bush, "Serial Number Arithmetic", RFC 1982, August 1996.

[RFC2988] Paxson, V. and M. Allman, "Computing TCP's Retransmission Timer", RFC 2988, November 2000.

[I-D.ietf-eap-rfc2284bis]
Blunk, L., "Extensible Authentication Protocol (EAP)", draft-ietf-eap-rfc2284bis-09 (work in progress), February 2004.

[RFC2716] Aboba, B. and D. Simon, "PPP EAP TLS Authentication Protocol", RFC 2716, October 1999.

[RFC2234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G. and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.

[RFC2462] Thomson, S. and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462, December 1998.

[RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, December 1998.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C. and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3456] Patel, B., Aboba, B., Kelly, S. and V. Gupta, "Dynamic Host Configuration Protocol (DHCPv4) Configuration of IPsec Tunnel Mode", RFC 3456, January 2003.

[I-D.ietf-eap-keying]
Aboba, B., "EAP Key Management Framework", draft-ietf-eap-keying-01 (work in progress), October 2003.

[IANA] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434,

October 1998.

Informative References

[I-D.ietf-pana-requirements]
Yegin, A. and Y. Ohba, "Protocol for Carrying Authentication for Network Access (PANA)Requirements", draft-ietf-pana-requirements-07 (work in progress), June 2003.

[I-D.ietf-aaa-eap]
Eronen, P., Hiller, T. and G. Zorn, "Diameter Extensible Authentication Protocol (EAP) Application", draft-ietf-aaa-eap-05 (work in progress), April 2004.

[I-D.puthenkulam-eap-binding]
Puthenkulam, J., "The Compound Authentication Binding Problem", draft-puthenkulam-eap-binding-04 (work in progress), October 2003.

[RFC2522] Karn, P. and W. Simpson, "Photuris: Session-Key Management Protocol", RFC 2522, March 1999.

[I-D.ietf-pana-usage-scenarios]
Ohba, Y., "Problem Statement and Usage Scenarios for PANA", draft-ietf-pana-usage-scenarios-06 (work in progress), April 2003.

[I-D.ietf-pana-threats-eval]
Parthasarathy, M., "PANA Threat Analysis and security requirements", draft-ietf-pana-threats-eval-04 (work in progress), May 2003.

[I-D.ietf-pana-ipsec]
Parthasarathy, M., "PANA enabling IPsec based Access Control", draft-ietf-pana-ipsec-03 (work in progress), May 2004.

[I-D.ietf-pana-framework]
Jayaraman, P., "PANA Framework", draft-ietf-pana-framework-00 (work in progress), May 2004.

[I-D.ietf-pana-snmp]
Mghazli, Y., Ohba, Y. and J. Bournelle, "SNMP usage for PAA-2-EP interface", draft-ietf-pana-snmp-00 (work in progress), April 2004.

[I-D.irtf-aaaarch-handoff]
Arbaugh, W. and B. Aboba, "Experimental Handoff Extension to RADIUS", draft-irtf-aaaarch-handoff-04 (work in

progress), November 2003.

[I-D.ietf-eap-statemachine]
Vollbrecht, J., Eronen, P., Petroni, N. and Y. Ohba, "State Machines for Extensible Authentication Protocol (EAP) Peer and Authenticator", draft-ietf-eap-statemachine-03 (work in progress), March 2004.

[I-D.ietf-seamoby-ctp]
Loughney, J., "Context Transfer Protocol", draft-ietf-seamoby-ctp-08 (work in progress), January 2004.

[I-D.ietf-ipsec-ikev2]
Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", draft-ietf-ipsec-ikev2-13 (work in progress), March 2004.

[I-D.josefsson-pppext-eap-tls-eap]
Josefsson, S., Palekar, A., Simon, D. and G. Zorn, "Protected EAP Protocol (PEAP)", draft-josefsson-pppext-eap-tls-eap-07 (work in progress), October 2003.

[I-D.ietf-pppext-eap-ttls]
Funk, P. and S. Blake-Wilson, "EAP Tunneled TLS Authentication Protocol (EAP-TTLS)", draft-ietf-pppext-eap-ttls-04 (work in progress), April 2004.

[I-D.tschofenig-eap-ikev2]
Tschofenig, H. and D. Kroeselberg, "EAP IKEv2 Method (EAP-IKEv2)", draft-tschofenig-eap-ikev2-03 (work in progress), February 2004.

[ianaweb] IANA, "Number assignment", http://www.iana.org.

[jb99] Juels, A. and J. Brainard, "Client Puzzles: A Cryptographic Defense Against Connection Depletion Attacks", Proceedings of NDSS '99 (Networks and Distributed Security Systems), pages 151-165, 1999.

[mitm] Asokan, N., Niemi, V. and K. Nyberg, "Man-in-the-middle in

tunnelled authentication", In the Proceedings of the 11th International Workshop on Security Protocols, Cambridge, UK, April 2003.

[802.11i] Institute of Electrical and Electronics Engineers, "Draft

supplement to standard for telecommunications and information exchange between systems - lan/man specific requirements - part 11: Wireless medium access control (mac) and physical layer (phy) specifications: Specification for enhanced security", IEEE 802.11i/D10.0, 2004.

[802.11] Institute of Electrical and Electronics Engineers, "Information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications", IEEE Standard 802.11, 1999(R2003).

URIs

[1] <http://www.toshiba.com/tari/pana/sequence-number.txt>

[2] <http://danforsberg.info:8080/pana-issues/>

Authors' Addresses

Dan Forsberg
Nokia Research Center
P.O. Box 407
FIN-00045 NOKIA GROUP
Finland

Phone: +358 50 4839470
EMail: dan.forsberg@nokia.com

Yoshihiro Ohba
Toshiba America Research, Inc.
1 Telcordia Drive
Piscataway, NJ 08854
USA

Phone: +1 732 699 5305
EMail: yohba@tari.toshiba.com

Basavaraj Patil
Nokia
6000 Connection Dr.
Irving, TX 75039
USA

Phone: +1 972-894-6709
EMail: Basavaraj.Patil@nokia.com

Hannes Tschofenig
Siemens Corporate Technology
Otto-Hahn-Ring 6
81739 Munich
Germany

EMail: Hannes.Tschofenig@siemens.com

Alper E. Yegin
Samsung Advanced Institute of Technology
75 West Plumeria Drive
San Jose, CA 95134
USA

Phone: +1 408 544 5656
EMail: alper.yegin@samsung.com

Intellectual Property Statement

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

Full Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.