Internet Engineering Task Force
Internet Draft
draft-forsberg-pana-secure-network-access-auth-01.txt
Expires: March 2003

Dan Forsberg
Jarno Rajahalme
Nokia
September 2002

Secure Network Access Authentication (SeNAA)
<draft-forsberg-pana-secure-network-access-auth-01.txt>

Status of this Memo

Copyright Notice

Abstract

This draft describes how reliable Secure Network Access Authentication (SeNAA) protocol over UDP carries Transport Layer Security (TLS) protocol. SeNAA messages are formatted like Diameter messages and contain Attribute Value Pairs (AVPs) that are protected with TLS Record Layer. SeNAA provides secure transport for Extensible Authentication Protocol (EAP) between PANA Client (PaC) and PANA Authentication Agent (PAA). PANA stands for Protocol for carrying Authentication for Network Access.

Table of contents

1.0 Introduction

Terminal's network access authentication in different network technologies has become an important issue in the Internet. Different authentication methods already exist but are more or less link layer dependent. Mobile terminals utilize different link layer technologies and roam between them. Generic link layer independent authentication and authorization method is needed to support smooth interaction between mobile terminals and access networks while roaming. A link layer agnostic solution for network access authentication is proposed.

This draft describes how reliable Secure Network Access Authentication (SeNAA) protocol over UDP carries Transport Layer Security (TLS) protocol. SeNAA messages are formatted like Diameter messages and contain AVPs that are protected with TLS Record Layer. SeNAA provides secure transport for EAP between PANA Client (PaC) and PANA Authentication Agent (PAA).

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119].

2.0 Terminology

This document uses the same terminology as has been described in PANA requirements draft [PANAREQ]. Additionally the following terms are used.

SeNAA
: Secure Network Access Authentication

3.0 Protocol overview

SeNAA uses UDP [UDP] as the transport protocol. UDP is lightweight and allows application level implementations with port numbers. UDP carries Diameter [DIAM] formatted SeNAA messages. SeNAA provides reliable request and response style message delivery (re-transmission and duplicate packet detection).

SeNAA does not assume a secure channel between PaC and PAA. Thus, on top of SeNAA protocol Transport Layer Security [TLS] protocol is used to negotiate a Local Security Association (LSA) between PaC and PAA. TLS provides authentication, privacy, integrity, and replay protection. It is used to protect SeNAA message AVPs and EAP [EAP] between PaC and PAA. AVPs that need protection are fed to the TLS Record layer and the resulting encrypted and compressed data is stored into a TLS-Payload AVP. EAP protocol is carried inside an EAP-

Payload AVP [NASREQ]. SeNAA messages after succesfull TLS handshake are integrity protected with a checksum stored in the Msg-Checksum AVP. The AVP is protected with TLS Record layer.

TLS is also used for re-authentication between PaC and PAA. TLS supports mutual authentication and can optionally be used instead of EAP for user authentication. In all cases TLS is used for access network authentication. SeNAA messages carry information such as the PaC's Device Identifier (DI), that MUST be integrity protected [PANAREQ]. If PAA supports DIAMETER and/or RADIUS AAA back-end, signaling between PaC and PAA can easily be extended to the back-end.

SeNAA is designed in such a way that the TLS protocol can be left out from the protocol stack. SeNAA messages can be carried over UDP without AVP encryption if the PaC and PAA already share an adequate secure channel (i.e. L2 encryption and authentication).

SeNAA doesn't rely on any modifications to the EAP protocol. It provides secure transport up to the PAA for EAP. Thus, any existing EAP methods can be used securely with SeNAA between PaC and PAA. Security after PAA is out of scope of SeNAA. PAA is assumed to get user authentication answer (Success or Failure) from the authenticator.

SeNAA utilizes protocols like EAP, TLS, UDP and IP that are assumed to exist in the PaC terminal already even without SeNAA. Diameter like message formatting and request/response style reliability transport is one additional requirement for the PaC terminal and is provided with SeNAA protocol. TCP [TCP] and SCTP [SCTP] are considered too heavy weight transport protocols for SeNAA purposes (i.e. more message round trips needed).

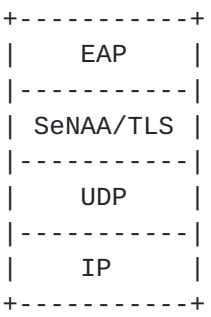


Figure 1. SeNAA Protocol stack in PaC

Data protection, such as IP datagrams, is out of the scope of SeNAA. One possibility for further studies is to use the key material produced in the TLS handshake process with IPSec [IPSEC].

3.1 SeNAA

Successful mutual authentication is divided into two phases. In phase 1 the network is authenticated and the user in phase 2.

Phase 1 consists of a TLS handshake as is shown in Figure 2a. Local re-authentication, where PaC authenticates to PAA, belongs to the phase 1 and is handled with TLS Session Resumption (Figure 2b). Access network authentication is based on access network certificates. How certificates are created, processed and verified is out of the scope of this document.

Phase 2 uses EAP for authenticating the user (Figure 3). User authentication is bound to the DI, which is used to control access to the network.

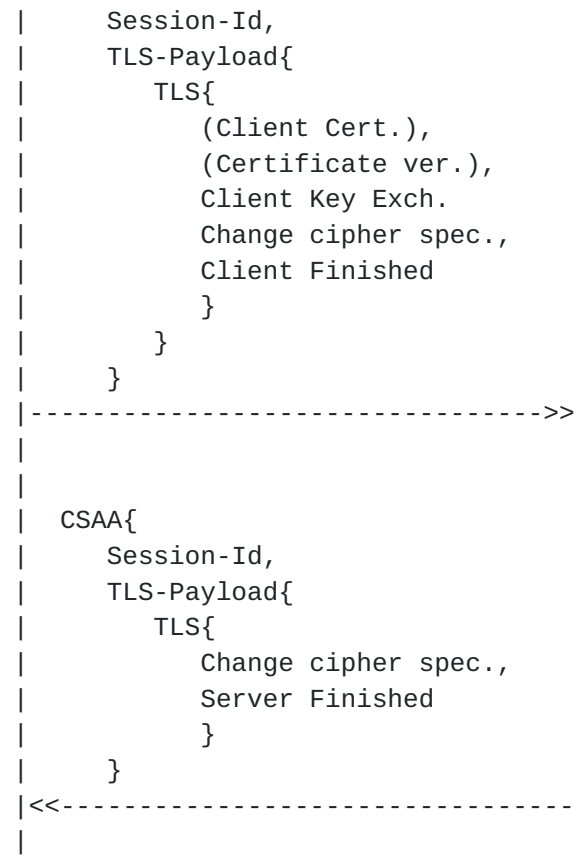
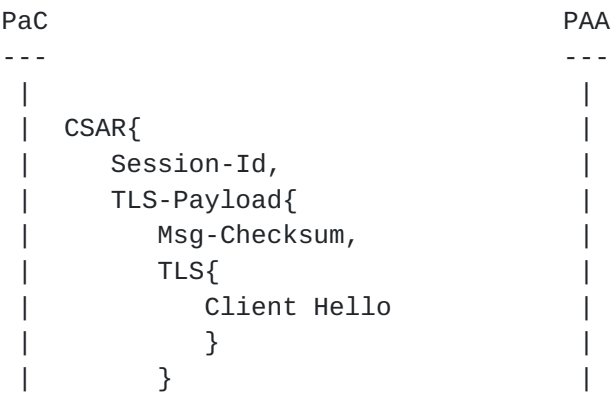


Figure 2a. Phase 1: Initial authentication with TLS.

In phase 1, Server-Certificate-Request (SCR) message carries Client-Hello TLS message. Server-Certificate-Answer (SCA) carries the TLS answer which contains the Access Network certificate. PaC verifies the certificate. PAA also adds a Session-Id AVP into the SCA message. This Session-Id is different from the TLS session-Id. The next messages MUST have this AVP included during the whole session. To finish the TLS handshake PaC sends Client-Security-Association-Request (CSAR) message to the PAA. PAA answers with Client-Security-Association-Answer (CSAA).



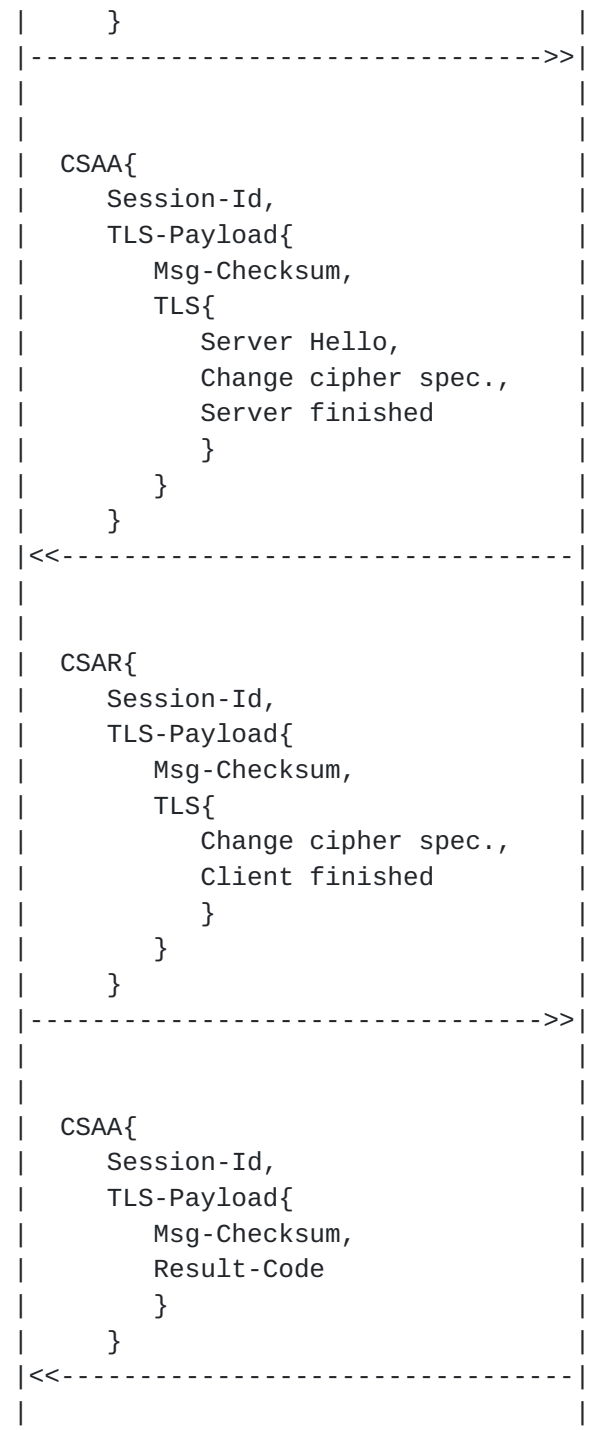


Figure 2b. Phase 1: Re-authentication with TLS.

When TLS is not used, a different UDP port number (PAA UDP port <UDP-port3>, PaC UDP port <UDP-port4>) MUST be used for plaintext CLR/CLA message delivery. PaC can decide not to use phase 1 authentication but MUST use phase 1 authentication if <UDP-port3> is not reachable. Similarly if UDP port <UDP-port1> is not reachable, PaC SHOULD try to

use UDP port <UDP-port3>.

In phase 2, after a successful TLS handshake, PaC uses AAA-Client-Request (CLR) message to start user authentication and DI authorization. CLR carries EAP-Payload AVP to PAA. PAA answers with AAA-Client-Answer (CLA) message with a Result-Code AVP [DIAM]. Result-Code informs PaC if multiple round trips are needed (DIAMETER_MULTI_ROUND_AUTH) for completing the EAP authentication method (DIAMETER_SUCCESS) or if the authentication (authorization) succeeded or failed (DIAMETER_AUTHENTICATION_REJECTED) [DIAM].

PaC adds it's DI(s) into the CLR message so that PAA can verify the integrity of the DI and optionally provide it for the enforcement point.

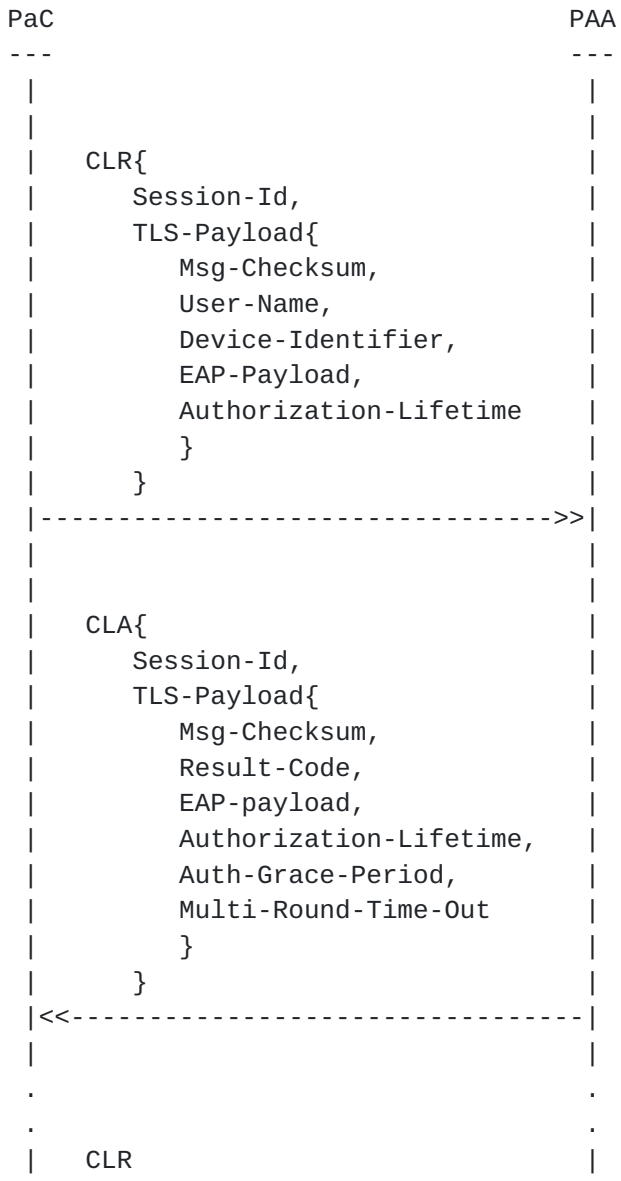


Figure 3. Phase 2: User authentication signaling. EAP is carried inside EAP-Payload AVP.

3.2 PAA location and discovery

PAA can be co-located in the AR or separated from the ARs. In case of separation, ARs act as relay agents to PAA for MN. PAA is in the same subnet as the AR and MN. When PAA receives a message from MN through an AR, it MUST send the reply directly to the MN. When PAA is separated from ARs, an AR SHOULD relay MN's SeNAA messages to the PAA. When MN receives an answer from PAA, it MUST send further requests to the PAA directly.

3.3 Reliable request and response style transactions

SeNAA uses request and response style transactions. For each request a response is sent. If response is not received in time, the request is re-sent. This mechanism is used for packet loss recovery. The received response works as an acknowledgment.

SeNAA uses Diameter message formatting. The Diameter message header provides packet duplication (End-to-End Id) detection and request/response mapping (Hop-by-Hop Id).

To protect integrity of the Diameter header and the parts that are not protected with TLS, a MAC is calculated over the Diameter message. The MAC is put into an AVP called Msg-Checksum.

3.4 Re-authentication

PaC needs to be authenticated before network access is granted through the enforcement point (EP) in the access network. Authentication and re-authentication initiator can be PaC or PAA.

PaC starts re-authentication by sending CSAR message with TLS Client Hello in the TLS-Payload AVP to PAA to UDP port <UDP-port1> (Figure 2b). Similarly, PAA initiates re-authentication by sending CSAA message with TLS Server Hello message in the TLS-Payload AVP to PaC to UDP port <UDP-port2>. The hello message contains the current TLS session specific id, which is used to detect session resumption from initial authentication [TLS]. Re-authentication involves multiple CSAR/CSAA round trips.

After TLS handshake or session resumption is done and the SA is established PaC uses the TLS Record Layer to encrypt SeNAA message AVPs.

3.5 Disconnect indication and session termination

SeNAA doesn't assume connection-oriented links [PANAREQ]. Thus, TLS re-authentication is used for notifying PAA of PaC's presence. Re-authentication interval is implementation specific <TBD?>.

PaC can explicitly terminate a session with SeNAA-Session-Termination-Request message (SSTR) sent to PAA. PAA answers with a SeNAA-Session-Termination-Answer. DIAMETER_LOGOUT [DIAM] is used in the Termination-Cause AVP in the SSTR message. The Result-Code value in SSTA is DIAMETER_SUCCESS [DIAM]if session was succesfully terminated or DIAMETER_UNKNOWN_SESSION_ID [DIAM], if session was not found.

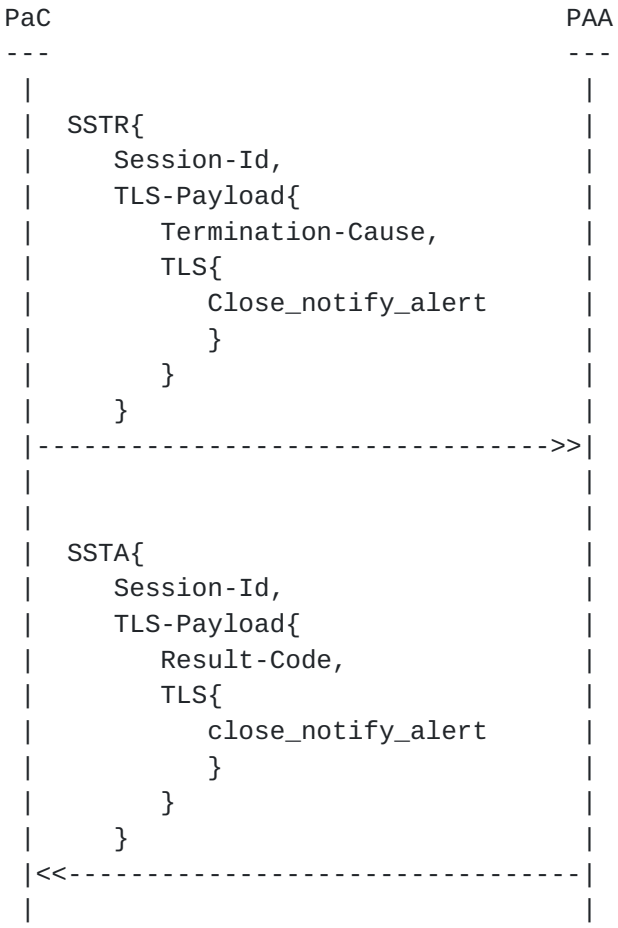


Figure 4. Session termination from PaC to PAA

PAA can also terminate the session with SeNAA-Abort-Session-Request (SASR). PaC answers with SeNAA-Abort-Session-Answer (SASA). Termination-Cause AVP contains DIAMETER_ADMINISTRATIVE, if the session was terminated due to the administrative reasons [DIAM], DIAMETER_SESSION_TIMEOUT, if the session timeout timer expired [DIAM].

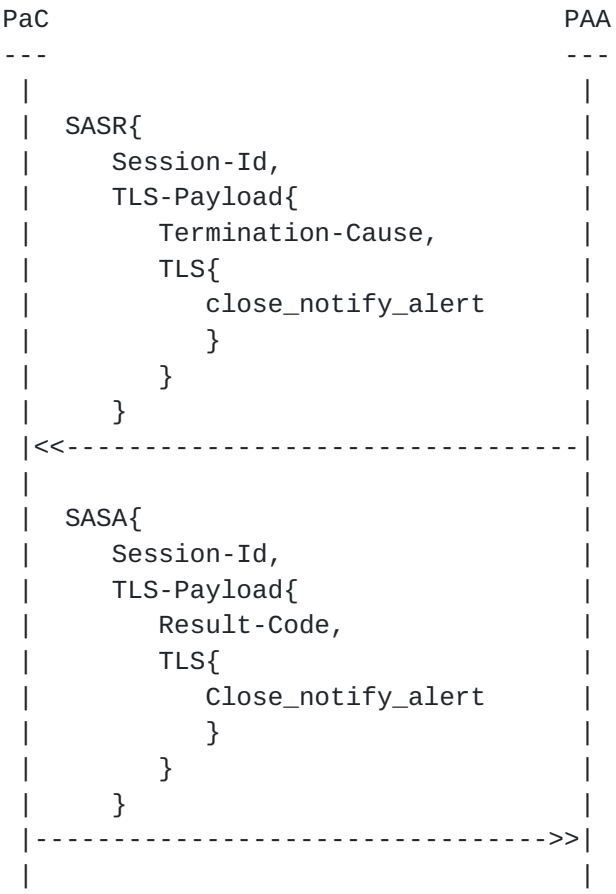


Figure 5. Session termination from PAA to PaC

3.6 Error handling with TLS

TLS Alert, Handshake and Change cipher spec. protocols are carried inside TLS Record Layer. For example if TLS Alert protocol reports a fatal error it is delivered with the next TLS-Payload AVP or with separate CSAR/CSAA messages. The SeNAA application MUST understand the return codes from TLS Record Layer API functions. When fatal error is received, the TLS session is torn down. The SeNAA session MUST be re-negotiated.

4.0 Message Formats

TLS message formats can be found from [TLS]. [DIAM] explains Diameter message header format and AVP format. This document also uses the ABNF notation for Diameter message format descriptions [DIAM]. SCA and CSAA message don't contain Result-Code AVPs.

4.1 Server-Certificate-Request (SCR)

Format of the SCR message:

```
< Server-Certificate-Request > ::= < Diameter Header: <TBD>,
                                    REQ, PXY >
                { User-Name }
                { TLS-payload }
```

TLS-Payload AVP contains the TLS handshake messages in the AVP data area as specified in [TLS]. The TLS-Payload AVP contains TLS-Client-hello.

4.2 Server-Certificate-Answer (SCA)

Format of the SCA message:

```
< Server-Certificate-Answer > ::= < Diameter Header: <TBD>, PXY >
                < Session-Id >
                { TLS-payload }
```

A Session-Id is generated for the PaC and delivered in the Session-Id AVP.

The TLS-Payload AVP contains TLS Server-hello, TLS Server-Certificate, TLS server-Key-Exchange, and TLS Server-Hello-Done messages.

4.3 Client-Security-Association-Request (CSAR)

Format of the CSAR message:

```
< Client-Security-Association-Request > ::= < Diameter Header: <TBD>,
                                              REQ >
                < Session-Id >
                { TLS-payload }
```

The Session-Id AVP is used in every SeNAA message between PaC and PAA.

The TLS-Payload contains TLS-Client-Key-Exchange, TLS-Change-Cipher-

Spec, and TLS-Client-Finished messages.

4.4 Client-Security-Association-Answer (CSAA)

Format of the CSAA message:

```
< Client-Security-Association-Answer > ::= < Diameter Header: <TBD>,
                                             PXY >
                < Session-Id >
                { TLS-Payload }
```

The TLS-Payload contains TLS-Change-Cipher-Spec and TLS-Server-Finished messages.

4.5 AAA-Client-Request (CLR)

Format of the initial CLR message:

```
< AAA-Client-Request > ::= < Diameter Header: <TBD>, REQ, PXY >
                < Session-Id >
                [ TLS-Payload:
                  { Msg-Checksum }
                  { User-Name }
                  { Device-Identifier }
                  [ EAP-Payload ]
                  [ Authorization-Lifetime ]
                ]
```

The TLS-Payload AVP data area contains encrypted AVPs through TLS Record layer.

4.6 AAA-Client-Answer (CLA)

Format of the CLA message from PAA to PaC:

```
< AAA-Client-Answer > ::= < Diameter Header: <TBD>, PXY >
                < Session-Id >
                [ TLS-Payload:
                  { Msg-Checksum }
                  { Result-Code }
                  [ EAP-payload ]
                  [ Authorization-Lifetime ]
                  [ Auth-Grace-Period ]
                  [ Multi-Round-Time-Out ]
                ]
```

The TLS-Payload AVP data area contains encrypted AVPs through TLS Record layer.

4.7 SeNAA-Session-Termination-Request (SSTR)

```
< SeNAA-Session-Termination-Request > ::= < Diameter Header: 275,
REQ,PXY >
                < Session-Id >
                [ TLS-Payload:
                 [ Msg-Checksum ]
                 { Termination-Cause }
                ]
```

4.8 SeNAA-Session-Termination-Answer (SSTA)

```
< SeNAA-Session-Termination-Answer > ::= < Diameter Header: 275,
REQ,PXY >
                < Session-Id >
                [ TLS-Payload:
                 [ Msg-Checksum ]
                 { Result-Code }
                ]
```

4.9 SeNAA-Abort-Session-Request (SASR)

4.10 SeNAA-Abort-Session-Answer (SASA)

5.0 AVP Formats

This section defines used AVPs that are not defined in [DIAM].

5.1 TLS-Payload AVP

The TLS-Payload AVP data contains encapsulated AVPs that are encrypted and compressed by TLS Record layer. Upon receive of TLS-Payload AVP the data area is first fed to the TLS Record layer to get the plain text AVP list for further processing.

5.2 Msg-Checksum AVP

The Msg-Checksum AVP data contains checksum of the Diameter message header and AVPs that are not protected with TLS. The TLS-Payload AVP MUST be the last AVP in the message. This makes it possible to calculate the MAC before creating the TLS-Payload AVP. Checksum algorithm is <TBD>.

5.3 Device-Identifier AVP

Device-Identifier AVP may contain one or more device identifiers, for example a layer 2 MAC address and an IPv6 address. Each identifier in the AVP data payload has the format described in Figure 6.

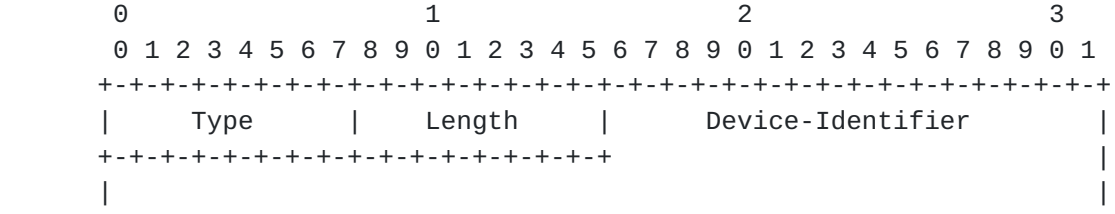


Figure 6. Device-Identifier format

Type is one of the following: <TBD>

6.0 Message re-transmission timers

SeNAA provides reliable request and response style transactions. Peer which initiates the transaction is responsible for re-transmission if the corresponding response is not received in <TBD-msec1> milliseconds. Maximum number of retries is <TBD-retries1>.

If Multi-Round-Time-Out AVP is included in a SeNAA message from PAA to PaC, the re-transmission (same Hop-by-Hop Id and End-to-End Id) MUST not exceed this time limit.

7.0 Security Considerations

If an EAP method is used in an unsecure environment, TLS with SeNAA doesn't provide adequate protection for the man-in-the-middle attack with that EAP method.

If TLS is not used, a secure enough link layer MUST be used between PaC and PAA.

8.0 IANA Considerations

UDP Port number(s) must be defined. SCR/SCA, CSAR/CSAA and CLR/CLA message command codes must be assigned. Msg-Checksum and TLS-Payload AVP codes must be assigned.

9.0 References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC2119, BCP0014, March 1997.

[PANAREQ] R. Penno (ed.), A. Yegin, Y. Ohba, G. Tsirtsis, C. Wang. "Protocol for Carrying Authentication for Network Access (PANA) Requirements and Terminology", Internet draft <draft-ietf-pana-requirements-04.txt>, April 2003, Work in progress.

[UDP] J. Postel, "User Datagram Protocol", RFC768, STD0006, Aug 1980.

[EAP] L. Blunk, J. Vollbrecht, Bernard Aboba, "Extensible Authentication Protocol (EAP)", Internet draft <draft-ietf-pppext-rfc2284bis-04.txt>, April 2002, Work in progress.

[TLS] T. Dierks, C. Allen., "The TLS Protocol Version 1.0.", RFC2246, January 1999.

[DIAM] Pat R. Calhoun, John Loughney, Erik Guttman, Glen Zorn, Jari Arkko, "Diameter Base Protocol", Internet draft, <draft-ietf-aaa-diameter-14.txt>, October 2002, Work in progress.

[NASREQ] P. Calhoun, W. Bulley, A. Rubens, J. Haag, "Diameter NASREQ Application", IETF work in progress.

[TCP] J. Postel, "Transmission Control Protocol.", STD0007, RFC793, Sep 1981.

[SCTP] L. Ong, J. Yoakum, "An Introduction to the Stream Control Transmission Protocol (SCTP)." RFC3286, May 2002.

[IPSEC] S. Kent, R. Atkinson, "IP Encapsulating Security Payload (ESP)." RFC2406, November 1998.

10.0 Full Copyright Statement

11.0 Acknowledgments

The authors of this document would like to thank N. Asokan for giving security information related to this draft.

12.0 Authors' Addresses

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

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

Jarno Rajahalme
Nokia Research Center
P.O. Box 407
FIN-00045 NOKIA GROUP, Finland

Phone: +358 50 4839470
EMail: jarno.rajahalme@nokia.com