

MIP6 Working Group
Internet Draft
Expires: April 2005

G. Giaretta
I. Guardini
E. Demaria
TILab
J. Bournelle
M. Laurent-Maknavicius
GET/INT
October 2004

MIPv6 Authorization and Configuration based on EAP
<[draft-giaretta-mip6-authorization-eap-02.txt](#)>

Status of this Memo

This document is an Internet-Draft and is subject to all provisions of [section 3 of RFC 3667](#). By submitting this Internet-Draft, I certify that any applicable patent or other IPR claims of which I am aware have been disclosed, and any of which I become aware will be disclosed, in accordance with [RFC 3668](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Abstract

This draft defines an architecture, and related protocols, for performing dynamic Mobile IPv6 authorization and configuration relying on a AAA infrastructure. The necessary interaction between the AAA server of the home provider and the mobile node is realized using EAP, exploiting the capability of some EAP methods to convey generic information items together with authentication data. This approach has the advantage that the access equipment acts as a simple pass-through for EAP messages and therefore does not play any active role in the Mobile IPv6 negotiation procedure, which makes the solution easier to deploy and maintain.

Internet-Draft

MIPv6 Authorization based on EAP

October 2004

Table of Contents

1.	Introduction.....	3
2.	Terminology.....	4
3.	Protocol Overview.....	5
4.	Requirements on EAP methods.....	10
5.	Detailed description of the Protocol.....	12
5.1	Mobile node bootstrapping.....	12
5.2	Management of re-authentication events.....	17
6.	Home Agent considerations.....	19
6.1	Requirements on AAAH-HA communication.....	19
6.2	Management of MIPv6 authorization state.....	20
7.	The MIPv6-Authorization container.....	22
7.1	PEAPv2.....	22
7.2	EAP-FAST.....	23
7.3	EAP-SIM.....	23
7.4	EAP-AKA.....	24
7.5	EAP-TTLS.....	24
7.6	EAP-IKEv2.....	25
8.	New TLVs.....	26
8.1	Service-Status-TLV.....	26
8.2	Service-Selection-TLV.....	27
8.3	Service-Options-TLV.....	27
8.4	Home-Agent-Address-TLV.....	28
8.5	Home-Address-TLV.....	28
8.6	IKE-Authentication-Options-TLV.....	29
8.7	IKE-Bootstrap-Information-TLV.....	30
8.8	Negotiation-Result-TLV.....	31
8.9	Authorization-Lifetime-TLV.....	32
9.	Security Considerations.....	33
10.	References.....	34
10.1	Normative References.....	34
10.2	Informative References.....	34
	Acknowledgments.....	36
	Authors' Addresses.....	36
	Intellectual Property Statement.....	37

1. Introduction

Mobile IPv6 [[RFC3775](#)] requires that Mobile Nodes (MNs) and Home Agents (HAs) share a set of configuration parameters: the MN must know its Home Address, the Home Agent Address and the cryptographic material needed to protect MIPv6 signaling (e.g. shared keys or certificates to setup an IPsec security association). MIPv6 base protocol does not specify any method to automatically acquire this information; which means that network administrators are normally required to manually set configuration data on MNs and HAs.

Manual configuration of Home Agents and Mobile Nodes also works as an implicit method for Mobile IPv6 authorization, because only the users that have been administratively enabled on a specific Home Agent are allowed to exploit Mobile IPv6 and its features.

However, in a large network (e.g. the network of a mobile operator), which may include millions of users and many Home Agents, the operational and administrative burden of this procedure may easily become overwhelming. In addition, the extensive use of manual and static configurations limits the flexibility and reliability of the system, in that it is not possible to dynamically assign the HA when the user enters the network, which would help to optimize performance and resource utilization (e.g. assignment of the HA closest to the MN's point of attachment).

This is generally referred to as the Mobile IPv6 bootstrapping problem. As discussed in [[MIPv6PS](#)], several bootstrapping scenarios can be identified depending on the relationship between the network operator providing IP services to the MN (Access Service Provider, ASP) and the service provider managing the HA (Mobility Service Provider, MSP). This document describes a solution to the bootstrapping problem that is applicable in a scenario where the ASP and the MSP are the same provider (Integrated ASP, IASP).

The proposed solution performs dynamic Mobile IPv6 authorization and configuration together with MN authentication for network access. MIPv6 negotiation and bootstrapping is controlled by the AAA server of the home provider (IASP), that interacts with the mobile node relying on AAA routing and EAP, exploiting the capability of some EAP methods (e.g. PEAPv2 [[PEAPv2](#)], EAP-FAST [[EAP-FAST](#)]) to convey generic information items together with authentication data.

[2](#). Terminology

General mobility terminology can be found in [[RFC3753](#)]. The following additional terms are used here:

ASP	Access Service Provider
IASP	Integrated Access Service Provider
MSP	Mobility Service Provider
AAA	Authentication Authorization Accounting
AAAH	AAA server of the Home domain

[3.](#) Protocol Overview

The basic idea behind the solution proposed herewith is to perform Mobile IPv6 bootstrapping during the authentication procedure undertaken by the Mobile Node to gain network access.

In particular, this draft defines a method to:

- explicitly authorize the use of Mobile IPv6 based on the service profile of the user, its position within the network, etc.
- dynamically allocate a Home Agent to the Mobile Node;
- dynamically configure Mobile IPv6 start-up parameters (i.e. MIPv6 bootstrapping) on the Mobile Node. These parameters include the Home Address and the cryptographic material needed to set-up the IPsec Security Association used to protect Mobile IPv6 signaling (i.e. Binding Updates and Binding Acknowledgements).

Figure 1 shows the overall architecture of the solution proposed in this draft. The central element of the architecture is the AAA server of the Home Domain (i.e. AAAH), which interacts with both the MN and

the selected HA to perform service authorization and configuration.

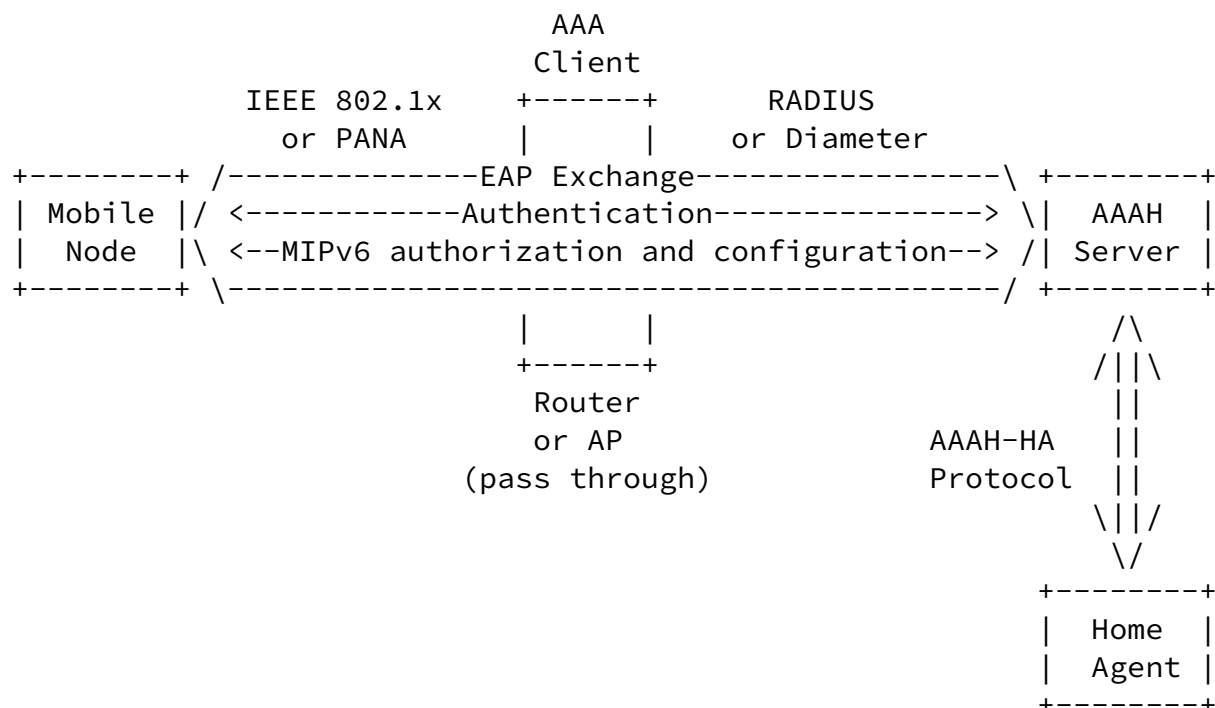


Figure 1 - Solution architecture

The solution is applicable to any access network relying on EAP [RFC3748] for user authentication and works with all EAP methods supporting the exchange of general purpose information elements, in any form (e.g. TLVs or AVPs), between EAP peers. Exploiting this

capability, MN and AAAH can piggyback Mobile IPv6 negotiation messages within the same EAP conversation used to carry out user authentication.

This kind of operation is already supported by several tunneled (e.g. PEAPv2 [PEAPv2]) and non tunneled (e.g. EAP-IKEv2 [EAP-IKEv2]) EAP methods, that also include native support for encryption, authentication and integrity protection of exchanged configuration data (e.g. HA address).

Figure 2 shows an overview of the procedure defined to handle MIPv6 bootstrap on the Mobile Node. For the sake of simplicity it is assumed that the employed AAA protocol is Diameter, but obviously RADIUS is suitable as well.

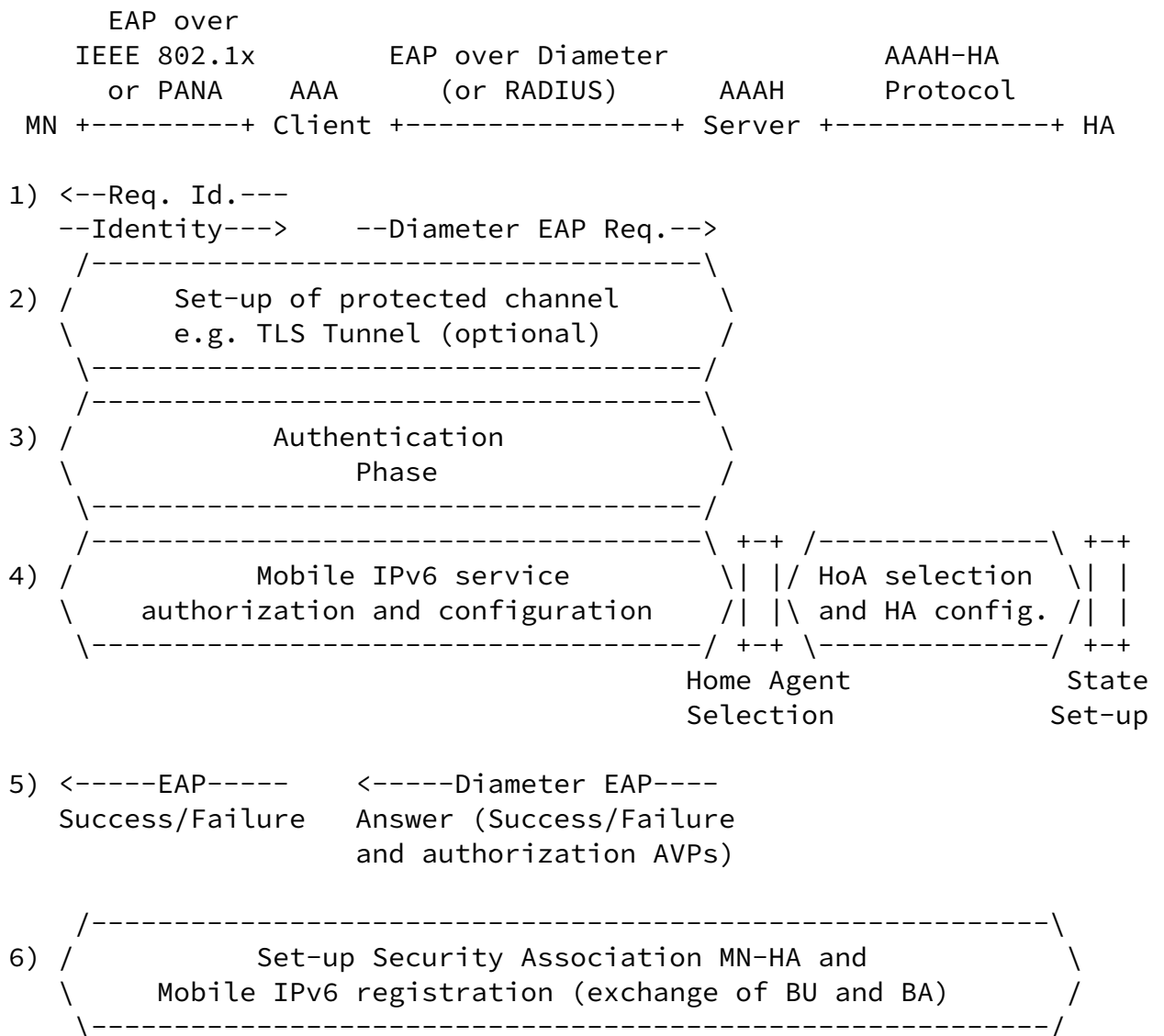


Figure 2 - Overview of Mobile IPv6 bootstrap procedure

The whole procedure can be divided in six steps:

1. EAP identity exchange (i.e. exchange of EAP Request Identity and EAP Response Identity messages);
2. set-up of a protected channel (e.g. TLS tunnel) for the delivery of subsequent EAP signaling. This is an optional step that is

present only if the EAP method provides confidentiality support. It is mandatory only if the MIPv6 negotiation procedure involves the exchange of sensitive information;

3. authentication phase. The actual authentication procedure and its security properties depend on the selected EAP method. In tunneled EAP methods (e.g. PEAPv2) this step may involve one or more complete EAP conversations occurring within a previously negotiated TLS session. Each EAP conversation may accomplish user authentication relying on any available EAP method (e.g. EAP-MD5, EAP-SIM, EAP-AKA);
4. Mobile IPv6 service authorization and configuration. MN and AAAH exchange a sequence of signaling messages to authorize and configure Mobile IPv6. Those messages are encapsulated as requested by the employed EAP method (e.g. TLVs or AVPs) and delivered as part of the on-going EAP session. If the EAP method provides confidentiality this protocol handshake is encrypted using the previously negotiated ciphersuite. During this phase, AAAH selects a suitable Home Agent for the MN and exchanges authorization and configuration data with it using a AAAH-HA protocol, whose specification is out of the scope of the present document. Further analysis on the definition of such an interface can be found in [[AAAH-HA](#)] and [[AAAMIPFWK](#)]. At the end of this phase, the MN knows its own Home Address, the address of the correspondent Home Agent, the peer authentication method (i.e. certificates or pre-shared key) and the cryptographic material (e.g. pre-shared key) needed to set-up an IPsec security association with IKE [[RFC2409](#)]. The IKE pre-shared key can be either constructed by AAAH and then delivered to MN in a proper TLV (or AVP) or independently derived by MN and AAAH from the EAP key hierarchy;
5. EAP session termination. Assuming the mobile node has been successfully authenticated, the AAAH server sends a Diameter EAP Answer message with Result-Code equal to SUCCESS. The AAA client extracts the EAP Success message from the Diameter EAP Answer and forwards it to the MN terminating the EAP session;
6. set-up of IPsec Security Association and MIPv6 registration. At the end of the EAP communication, the MN gains network access and acquires a valid Care-of Address within the visited subnet (e.g.

via stateless autoconfiguration); then it performs an IKE exchange to establish the IPsec Security Association with the HA, using the authentication method and the cryptographic material negotiated during the MIPv6 service configuration phase (step 4). Finally, the MN performs MIPv6 registration, sending a Binding Update (protected with IPsec) to the HA.

This draft also defines the procedures to handle re-authentication events and to manage the termination of the Mobile IPv6 session.

In summary, the proposed architecture has the following advantages:

- allows the MSP to maintain a centralized management (on the AAA server) of the user profiles and the authentication, authorization and accounting procedures for any type of service, including Mobile IPv6;
- improves the reliability and performance of the Mobile IPv6 protocol, in that the HA to be dynamically assigned to the MN can be freely chosen among those that are closest to the user's point of attachment, thus optimizing network usage and reducing the transfer delay for data traffic in bi-directional tunneling;
- can be deployed, or extended with new features, without having to update the access equipment and the AAA protocols in use. Only minor changes in the AAA servers, the Home Agents and the mobile terminals are required, in that the AAA client does not play any active role in MIPv6 negotiation (i.e. it is a pass-through for EAP signaling). This reduces the deployment costs and makes the solution easy to use even when a Mobile Node is roaming with a provider different from its own;
- allows the usage of any AAA protocol supporting the transport of EAP messages for the communication between the AAA client and server (i.e. not just Diameter, but also RADIUS). This significantly simplifies the deployment of MIPv6 in existing communication networks, where support for Diameter protocol in access equipment is not so extensive.
- allows the operator to dynamically choose the authentication method for IKE bootstrapping and to automatically distribute the pre-shared key eventually needed; in this way the pre-shared key must not be pre-configured and can be frequently changed increasing resistance to attacks. In the case of an EAP method providing dynamic generation of keying material the pre-shared key can be derived from EAP hierarchy avoiding the need to explicitly send it to the MN [[MIPv6AMSK](#)].

As a whole, the solution adds a maximum of 2 RTTs (see the detailed protocol description in [section 5](#)) to the EAP conversation carried out by the mobile node to authenticate itself and gain network access. The number of extra RTTs can be reduced if the employed EAP method has the capability of transporting MIPv6 negotiation TLVs (or AVPs) together with authentication data. Nonetheless, it should be noted that the full negotiation procedure can be undertaken by the MN only during its initial bootstrapping, and therefore the performance requirements are not so strict.

[4.](#) Requirements on EAP methods

In EAP methods, the EAP peer and EAP server exchange data in order to authenticate the EAP peer and eventually the EAP server (mutual authentication). This draft proposes the use of these exchanges to transport MIPv6 parameters, as part of an handshake that requires at maximum 2 RTTs. Thus, the main requirement for the applicability of the solution is:

- the EAP method must provide a way to carry arbitrary parameters during or after the authentication phase. This implies that the EAP method must provide messages and mechanisms for this purpose.

Then, for security reasons, the EAP method must provide the following properties:

- mutual authentication: the EAP method must provide mutual authentication. The access network must authenticate users before granting them Mobile IPv6 service and the EAP peer should authenticate the access network before delivering sensitive data;
- integrity: the exchanged MIPv6 parameters must be protected against any alteration and thus the EAP method must provide integrity protection;
- replay protection: the EAP messages containing MIPv6 parameters must be protected against Replay Attack, so that an attacker is not able to get previous given data by replaying an old request;
- confidentiality: depending on which data the AAA server provides to the mobile node (e.g. an IKE pre-shared key), it may be necessary to protect the message exchange against eavesdropping.

The table below checks some existing EAP methods against the requirements listed above.

Internet-Draft

MIPv6 Authorization based on EAP

October 2004

M-A: Mutual Authentication

R-P: Replay Protection

	M-A	Integrity	R-P	Confidentiality	Exchange of arbitrary Parameters
PEAPv2	x	x	x	x	x
EAP-FAST	x	x	x	x	x
EAP-TTLS	x	x	x	x	x
EAP-IKEv2	x	x	x	x	x
EAP-SIM	x	x	x	x	x
EAP-AKA	x	x	x	x	x
EAP-TLS	x	x	x	x	
EAP-MD5					

In summary, it is possible to note that the procedure described in this draft can be successfully undertaken with several tunneled (PEAPv2, EAP-FAST and EAP-TTLS) and non tunneled EAP methods (EAP-

IKEv2, EAP-SIM, EAP-AKA), that all support the transport of arbitrary parameters.

[5.](#) Detailed description of the Protocol

This section details the procedures and message exchanges that can be adopted by the network operator to explicitly authorize the activation of Mobile IPv6 support for a specific user as well as enable dynamic bootstrapping of MIPv6 protocol parameters (e.g. Home Address, Home Agent Address).

[5.1](#) Mobile node bootstrapping

If EAP is used for access control, when the MN enters the network it is immediately polled for its identity, by means of an EAP Request Identity message. This message is used to start the EAP communication. The MN replies sending its identity, in the form of a NAI (Network Access Identifier), within an EAP Response Identity message, that is received by a AAA client (e.g. the Access Point in the case of a Wireless LAN) and forwarded via AAA routing to the AAAH server using the Diameter EAP Application (or the RADIUS EAP extensions). Then the AAAH server selects an EAP method (e.g. based on the user service profile) and proposes it to the MN in subsequent EAP messages. In order to enable the Mobile IPv6 negotiation procedure defined in this document, the EAP method chosen by the AAAH

server must be an EAP method supporting the transport of general purpose and variable length information elements, in the form of TLVs (or AVPs), together with authentication data (see [section 4](#)).

After this initial handshake, MN and AAAH undertake the actual authentication phase, that may require the exchange of a variable number of EAP Request/Response messages. In many EAP methods, like PEAPv2 or EAP-IKEv2, the authentication phase is preceded by the establishment of an encrypted channel between MN and AAAH that provides protection capabilities (i.e. privacy, integrity protection, etc.) for all the messages exchanged during the rest of the EAP conversation.

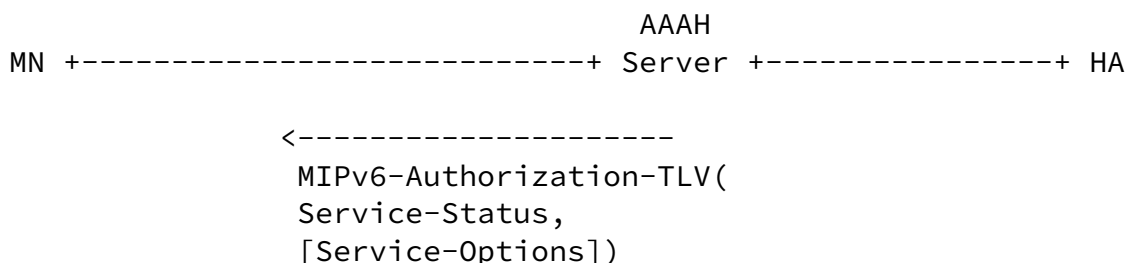
As soon as the authentication phase is completed, the procedure for MIPv6 bootstrapping may take place. For that purpose, the MN and the AAAH server exploit the on-going EAP communication to exchange a sequence of signaling messages transporting configuration parameters.

All the messages used for MIPv6 bootstrapping are encoded in TLVs carried by a generic MIPv6-Authorization container. This choice simplifies a lot the deployment of the procedure with any EAP method satisfying the requirements listed in [section 4](#). In fact, only the structure of the MIPv6-Authorization container needs to be adapted to the actual message format of the employed EAP method.

For the sake of simplicity, in this section it is assumed that the EAP method used for network access authentication supports the

transport of arbitrary parameters in TLV format. In this case the MIPv6-Authorization container becomes a MIPv6-Authorization-TLV. Nonetheless, in [section 7](#) the format of the container is defined for all the EAP methods identified in [section 4](#).

The whole bootstrapping procedure is depicted in Figure 3.



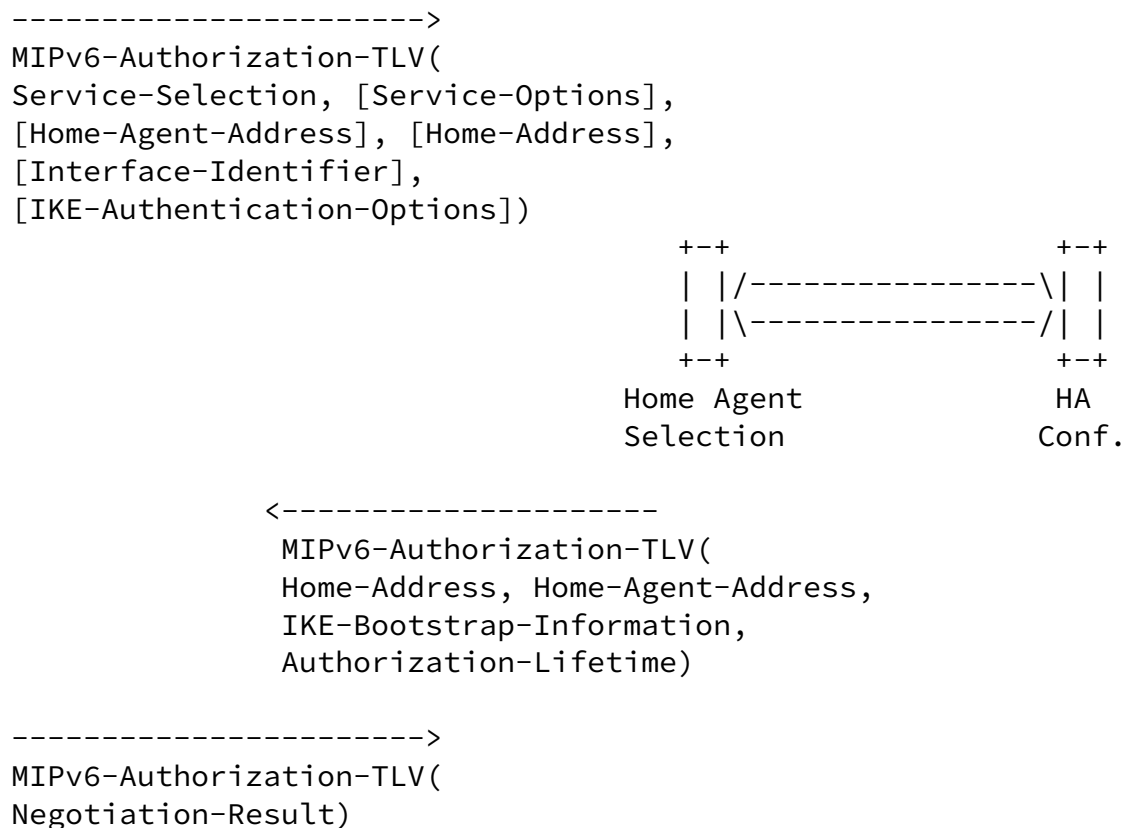


Figure 3 - MIPv6 bootstrapping procedure

AAAH starts the MIPv6 negotiation phase sending to the MN a MIPv6-Authorization-TLV including the following TLVs:

- Service-Status-TLV: used to communicate whether the home domain is willing to provide Mobile IPv6 service to the MN. This might depend on the user service profile or on other administrative rules (e.g. service accountability);

- Service-Options-TLV (optional): used to specify other service options the MN can ask for (e.g. allocation of a HA in the visited domain).

MN replies to this first message confirming its intention to start Mobile IPv6 and, optionally, providing a set of hints on the desired service capabilities; this is achieved delivering a MIPv6-Authorization-TLV including the following TLVs:

- Service-Selection-TLV: used by the MN to specify if it is willing to activate Mobile IPv6 protocol operation;
- Service-Options-TLV (optional): used by the MN to communicate which service options, among those previously advertised by AAAH, it would like make use of;
- Home-Agent-Address-TLV (optional): used by the MN to suggest a preferred Home Agent. This can be a HA with whom the MN has a pre-configured Security Association or a HA acquired through dynamic HA address discovery. The AAAH server treats this indication just as a hint, which means that, for administrative reasons, the MN may be assigned a Home Agent different from the one previously requested;
- Home-Address-TLV (optional): used by the MN to suggest a preferred Home Address (e.g. an address pre-configured on one of its network interfaces); like the previous TLV, this indication is considered only as a hint by the AAAH server;
- Interface-Identifier-TLV (optional): through this TLV, the MN can suggest a preferred Interface Identifier (selected according to [\[RFC3041\]](#) or following other criteria) to be used by the AAA infrastructure to build the Home Address starting from the selected home prefix. Also in this case, this information, if present, is treated as a pure hint by the AAAH server.
- IKE-Authentication-Options-TLV (optional): through this TLV, the MN communicates to the AAAH server in order of preference the peer authentication methods it supports for IKE bootstrapping. The lack of this TLV means that the MN supports only the default method.

The solution described in this document supports the following methods for peer authentication in IKE phase 1:

- use of a pre-shared key (PSK) derived by the AAAH server and sent to the MN and the HA; in this case confidentiality must be provided by both the AAAH-HA protocol and the EAP session between the MN and the AAAH server. This is the default method.

- use of a pre-shared key independently derived by the MN and the

AAAH server from the keying material exported by the employed EAP method. This key can be derived from an Application Master Session Key (AMSK) specific to Mobile IPv6, which can be constructed as explained in [[MIPv6AMSK](#)]. The PSK is then delivered by the AAAH server to the HA by means of a AAAH-HA protocol providing confidentiality;

- use of digital certificates. This solution involves the employment of digital signatures and public key encryption as stated in [[RFC2409](#)]. This method requires the availability of a PKI.

If in the Service-Selection-TLV the MN has chosen not to make use of Mobile IPv6, AAAH terminates the EAP communication sending an EAP Success message, since the authentication procedure has been accomplished successfully.

Otherwise, if the MN has confirmed its willingness to start MIPv6 service, AAAH selects a suitable Home Agent through a Home Agent Selection Algorithm. Possible parameters to be taken into account by this algorithm include: current load of available HAs (e.g. number of active bindings), location of the MN and, eventually, the preferences provided by the MN itself in the previous message exchange (i.e. Service-Options-TLV, Home-Agent-Address-TLV, Home-Address-TLV, IKE-Authentication-Options-TLV). For example, based on the knowledge of the MN's current point of attachment (i.e. the current AAA client), the AAAH server may select, among the HAs available in the home domain, the one that is closest to the MN in terms of IP routing hops. This approach is normally expected to improve performance. However, the detailed definition of a Home Agent Selection Algorithm is out of the scope of this document.

After a suitable HA has been identified, the AAAH server selects the peer authentication method to be used in IKE phase 1. The peer authentication methods supported by the MN are known from the IKE-Authentication-Options-TLV received during the previous exchange. If possible, the AAAH server should choose the method on top of the list provided by the MN (i.e. the one with the highest preference).

As soon as the peer authentication method has been identified, the AAAH server interacts with the HA to dynamically configure the state needed to enable subsequent MIPv6 protocol operations, including the authorization lifetime of the MIPv6 service granted to the MN and the necessary security parameters (e.g. pre-shared key). Possible protocols that can be used for this purpose include Diameter (through a new Diameter Application), SNMPv3 or COPS-PR. Further details about this communication are provided in [section 6](#). Anyway, the detailed specification of the interface between AAAH and HA is out of the

scope of this document. Additional considerations on the nature and goals of such an interface can be found in [\[AAAH-HA\]](#) and [\[AAAMIPFWK\]](#).

The security parameters that the AAAH server delivers to the HA may vary depending on the peer authentication method chosen for IKE bootstrapping:

- if the AAAH server selects pre-shared key as peer authentication method it needs to send the chosen PSK (randomly generated or derived from the EAP key hierarchy) to the HA together with the associated lifetime;
- if the AAAH server selects a peer authentication method based on certificates it does not need to derive keys nor to send security parameters to the HA.

After the AAAH server has configured the state on the HA, it continues the EAP session communicating to the MN all the MIPv6 configuration data it is waiting for. This is achieved delivering to the MN an EAP Request containing a MIPv6-Authorization-TLV and the following sub-TLVs: Home-Address-TLV (i.e. the home address), Home-Agent-Address-TLV (i.e. the address of the HA), IKE-Bootstrap-Information-TLV (i.e. the peer authentication method to be used in IKE phase 1 and associated cryptographic material) and Authorization-Lifetime-TLV (i.e. the lifetime granted to the MN for this session).

After the MN has received all the configuration data from the AAAH server (i.e. HA address, Home Address and IKE bootstrap information), it sends back an EAP Response containing a Negotiation-Result-TLV, stating whether it accepts, or refuses, the proposed arrangement. If the MN refuses the configuration, the AAAH server should immediately release the resources previously allocated on the Home Agent.

After the completion of the EAP session, MN holds all data needed to perform Mobile IPv6 registration: the MN knows its Home Address, the address of the correspondent Home Agent and all cryptographic data needed to establish the IPsec security association with it; furthermore, since it has been successfully authenticated, the MN can acquire an IPv6 address to be used as Care-of Address.

The first operation carried out by the MN after the acquisition of the Care-of Address is the establishment of the IPsec Security Association with the HA, that is mandated by [\[RFC3775\]](#) to protect MIPv6 location update signaling. Set-up of the IPsec SA can be accomplished following the procedure detailed in [\[RFC3776\]](#).

As soon as the IPsec Security Association is established, MN can send a Binding Update to the HA, thus starting up Mobile IPv6 service.

[5.2](#) Management of re-authentication events

At the expiration of AAA session time-outs or after a change in the point of attachment to the network (e.g. change of Access Point), a re-authentication procedure is performed leading to the user identity to be checked again along with its right to continue exploiting network resources. To that purpose the AAAH server may repeat a full authentication or, alternatively, decide to use optimizations in order to make the procedure faster. Once this phase is completed the AAAH server also undertakes the re-negotiation of the MIPv6 service.

Since the MIPv6 bootstrapping procedure is assumed to be completely stateless, when a re-authentication event occurs the AAAH server may not know the state of the MIPv6 service on the MN. For this reason the AAAH server starts the MIPv6 negotiation like in the bootstrapping case: it delivers a MIPv6-Authorization-TLV containing a Service-Status-TLV and optionally a Service-Options-TLV.

If the MIPv6 service is not active on the MN the procedure continues as described in [section 5.1](#). Otherwise, the MN replies with a MIPv6-Authorization-TLV containing a Service-Selection-TLV indicating that the MIPv6 service is already in use. Furthermore, the MN inserts the Home-Agent-Address-TLV, the Home-Address-TLV and the IKE-Authentication-Options-TLV to inform the AAAH server about its current state. The AAAH server can then get in touch with the HA to check the integrity of the state, renew the MIPv6 authorization lifetime and eventually refresh the security parameters.

If the peer authentication method used by the MN in IKE phase 1 is pre-shared key, the AAAH server must derive a new PSK and send it to the HA together with the associated lifetime. In case the PSK is not derived from the EAP key hierarchy (i.e. it is randomly generated by the AAAH server), the AAAH server must communicate it also to the MN. Instead, in case of authentication based on certificates, the AAAH server does not need to derive keys nor deliver additional security data to the HA and the MN.

If the state on the HA is successfully updated, the AAAH server terminates the EAP communication sending an EAP Success message.

Otherwise, the AAAH server should continue the EAP communication renegotiating the MIPv6 service (i.e. allocation of a new HA and related Home Address).

This solution can be easily deployed even within a network including several AAA servers, each one managing a subset of the available Network Access Servers (NASs). This is because the re-negotiation procedure does not assume to have any long term state related to Mobile IPv6 stored on the AAAH server. In this way, everything works correctly even if, due to MN's movements within the network, the AAAH

server that handles the re-authentication is not the same server that authenticated the MN for the first time and performed the MIPv6 bootstrapping procedure.

As explained above, the re-authentication events are normally triggered by the network. Nonetheless, if the EAP lower layer offers a way to trigger EAP re-authentications (e.g. PANA supports this feature), it may be possible for the MN to re-negotiate the MIPv6 service at any time.

[6.](#) Home Agent considerations

This section provides further thoughts about the AAAH-HA communication and lists specific features that have to be supported by the Home Agent to allow dynamic negotiation of Mobile IPv6 protocol parameters.

[6.1](#) Requirements on AAAH-HA communication

This draft details only the message exchange between the MN and the AAAH server. Obviously a complete Mobile IPv6 bootstrapping solution requires also the definition of a mechanism for the communication between the AAAH server and the Home Agent. Possible protocols that may be used for this purpose include SNMPv3, COPS-PR or Diameter but a formal definition of such a protocol is out of scope of this document.

A detailed analysis of the goals for a generic AAAH-HA protocol can be found in [[AAAH-HA](#)]; in this section some requirements, specific to this scenario, are highlighted.

The selected protocol should allow the AAAH server to:

- use a Network Access Identifier (NAI) to identify the mobile node in the communication with the HA;

- send an authorization lifetime to the HA to limit Mobile IPv6 session duration for the MN;
- send to the HA a set of hints for the construction of the Home Address (e.g. a preferred Home Address or a preferred Interface Identifier);
- poll the designated HA for the allocation of a Home Address to the MN;
- force the HA to terminate an active Mobile IPv6 session for authorization policy reasons (e.g. credit exhaustion or reallocation of a new HA to the MN);
- enforce explicit operational limitations and authorization restrictions on the HA (e.g. packet filters, QoS parameters);
- retrieve the Mobile IPv6 state associated to a specific MN from the correspondent HA. This may be useful to periodically verify the Mobile IPv6 service status;

- send to the HA the security data needed to setup the IPsec SA with the MN. Possible security data are the authentication method and the cryptographic material to be used for IKE bootstrapping.

Moreover, the protocol selected to implement the communication between the AAAH server and the HA should fulfill the following general requirements:

- the AAAH server and the HA must be able to authenticate each other (mutual authentication) in order to prevent the installation of unauthorized state on the HA;
- the AAA-HA interface must provide integrity protection in order to prevent any alteration of exchanged data (e.g. Mobile IPv6 configuration parameters);
- the AAA-HA interface must provide replay protection;

- the AAA-HA interface should provide confidentiality since it may be used to transfer security parameters (e.g. IKE pre-shared key);
- the AAA-HA interface should support inactive peer detection. This functionality can be used by the AAAH server to maintain a list of active HAs (e.g. useful for HA selection);

6.2 Management of MIPv6 authorization state

The Home Agent is required to store some authorization data for each of the MNs it is serving. A new data structure may be used for this purpose and it should include at least the following fields:

- NAI of the user;
- Home Address assigned to the MN;
- Cryptographic Data: this field includes the peer authentication method to be used in IKE and, if needed, the pre-shared key and its lifetime;
- Authorization Lifetime: it is the lifetime of the Mobile IPv6 service granted to the MN;

At the expiration of the Authorization Lifetime the HA should check if there is an active entry for the MN in its Binding Cache in order to verify if the MN is still using Mobile IPv6. If the entry is available the Home Agent should negotiate with the AAAH server an extension of the Authorization Lifetime granted to the MN. Otherwise, the HA should immediately release the authorization state associated

to that MN and eventually notify the session termination to the AAAH server (e.g. by means of a Session Termination Request if the employed AAAH-HA protocol is Diameter).

Moreover, the release of the resources previously allocated on the Home Agent can be undertaken at any time by the AAAH server. Typically this happens at credit exhaustion or when the MN disconnects from the network.

The policies adopted by the AAAH server to release the resources allocated to the MN may vary depending on the user service profile.

For instance, the AAAH server may decide to postpone the release of the resources on the HA in order to allow the MN to continue using the Mobile IPv6 service even if it has moved to an access network for which no roaming agreements are in place (e.g. a corporate network or a network providing cost-free access). In that case, the MN can continue to rely on the IPsec SA previously negotiated with the HA and the respective authorization is managed through the Mobile IPv6 Authorization Lifetime.

[7.](#) The MIPv6-Authorization container

All the messages used for MIPv6 bootstrapping are encoded in TLVs carried by a generic MIPv6-Authorization container. In this way, only

the structure of the container needs to be adapted to the actual message format of the employed EAP method.

The MIPv6-Authorization container can be implemented as a TLV, as an AVP or in some other way depending on the specific characteristics of the EAP method used for network access authentication (see Figure 4).

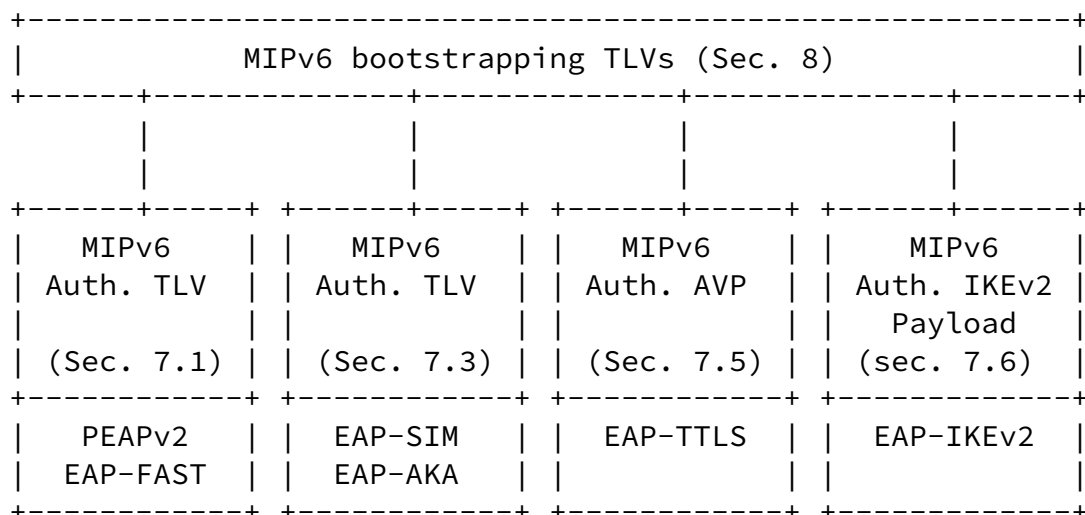
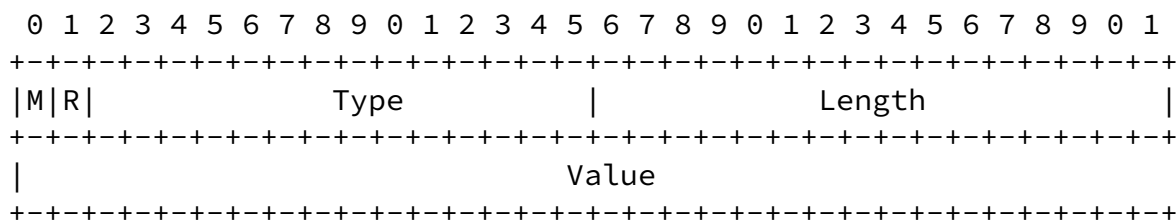


Figure 4 - Transport of MIPv6 bootstrapping messages

In the following the format of the MIPv6-Authorization container is defined for each EAP method identified in [section 4](#). This list is not exhaustive and does not prevent the use of other EAP methods satisfying all the requirements listed in this document.

[7.1](#) PEAPv2

The exchange of arbitrary information in PEAPv2 is based on EAP-TLVs. In this case the MIPv6-Authorization container is encoded as an EAP-TLV and has the structure depicted below:



M

0 - Non-mandatory TLV

R

Reserved, set to zero (0)

Type

TBD - MIPv6-Authorization

Length

The length of the Value field in octets

Value

This field carries the subsequent TLVs

[7.2](#) EAP-FAST

The format of the messages for EAP-FAST [[EAP-FAST](#)] is the same as PEAPv2.

[7.3](#) EAP-SIM

EAP-SIM [[EAP-SIM](#)] allows the transport of additional information in form of TLVs. The format of the MIPv6-Authorization container is depicted below:

```

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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           Type           |      Length      |           Value           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Type

TBD - MIPv6-Authorization

Length

Indicates the length of this attribute in multiples of four

bytes. The maximum length of an attribute is 1024 bytes. The length includes the Type and Length bytes.

Internet-Draft

MIPv6 Authorization based on EAP

October 2004

Value

This field carries the subsequent TLVs

7.4 EAP-AKA

The format of the messages for EAP-AKA [[EAP-AKA](#)] is the same as EAP-SIM.

7.5 EAP-TTLS

EAP-TTLS messages [[EAP-TTLS](#)] allow the exchange of arbitrary data in the form of AVPs encapsulated in the TLS record. The MIPv6-Authorization container is encoded as depicted below:

[illegible]

AVP Code

TBD - MIPv6-Authorization

Flag 'V' (Vendor-Specific)

0

Flag 'M' (Mandatory)

0

Flag 'r' (reserved)

must be set to 0

AVP Length

the length of this AVP including the AVP Code, AVP Length, flags, Vendor-ID (if present) and Data.

Data

this field carries subsequent TLVs

7.6 EAP-IKEv2

EAP-IKEv2 [[EAP-IKEv2](#)] allows the transport of generic data in the form of IKEv2 payloads. The MIPv6-Authorization container is encoded as depicted below:

```

 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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Next Payload |C|  RESERVED   |          Payload Length          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     Data                                     ~
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

Next Payload (1 octet)

TBD - MIPv6-Authorization

Critical (1 bit)

must be set to zero

RESERVED (7 bits)

must be sent as zero; must be ignored on receipt.

Payload Length (2 octets)

Length in octets of the current payload, including the generic payload header

Data

It contains subsequent TLVs

Giaretta, et al.

Expires - April 2005

[Page 25]

Internet-Draft

MIPv6 Authorization based on EAP

October 2004

[8.](#) New TLVs

Independently from the EAP method used for network access authentication, the MIPv6-Authorization container enables to transport a series of TLVs. This gives more flexibility to the whole solution and permits the definition of new TLVs that do not need to be bound to a specific EAP method.

The following TLVs have been defined so far:

- Service-Status-TLV
- Service-Selection-TLV
- Service-Options-TLV
- Home-Agent-Address-TLV
- Home-Address-TLV
- IKE-Authentication-Options-TLV
- IKE-Bootstrap-Information-TLV
- Negotiation-Result-TLV

[8.1](#) Service-Status-TLV

This TLV is sent by the AAAH to inform the MN about the status of Mobile IPv6 service. It is defined as follows:

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

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Type=Service-Status   |   Length   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Code   |
+---+---+---+---+---+---+

```

Type

TBD - Service-Status

Length

1

Code

0 = MIPv6 service is available
1 = MIPv6 service is not available

[8.2](#) Service-Selection-TLV

This TLV is sent by the MN to inform the AAAH whether it wants to activate MIPv6 service or whether the service is already active.

```

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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Type=Service-Selection   |   Length   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Code   |
+---+---+---+---+---+---+

```

Type

TBD - Service-Selection

Length

1

Code

- 0 = activate MIPv6 service
- 1 = MIPv6 service already active
- 2 = do not activate MIPv6 service

8.3 Service-Options-TLV

This TLV is used by the AAAH server to advertise the service options the MN can ask for. It is also used by the MN to communicate its selection to the AAAH. So far only the HA in visited domain option has been defined. The TLV has the following format:

```

  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
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|          Type=Service-Options          |          Length          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|V|          Reserved                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Type

TBD - Service-Options

Length

2

V

from AAAH to MN:

- 0 = AAAH cannot provide a HA in the visited domain
- 1 = AAAH can provide a HA in the visited domain

from MN to AAAH:

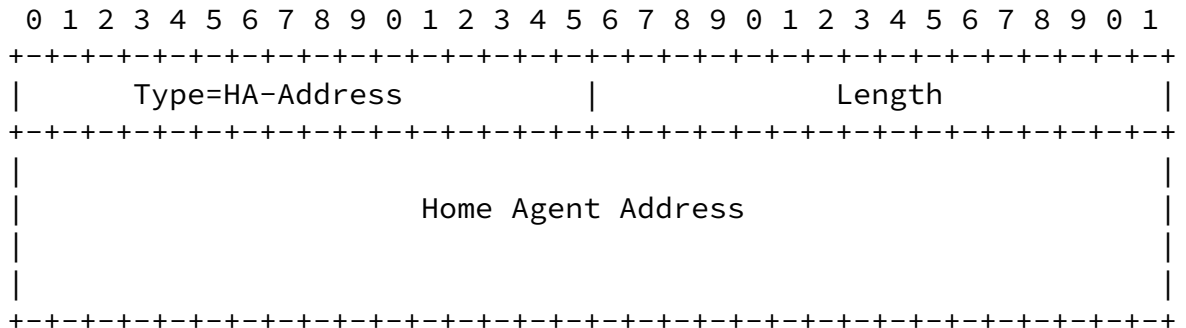
- 0 = MN does not specify any preference on HA location
- 1 = MN is requesting a HA in the visited domain

Reserved

15 bit reserved set to 0

[8.4](#) Home-Agent-Address-TLV

This TLV carries the Home Agent's address and it's defined as follows:



Type

TBD - Home-Agent-Address

Length

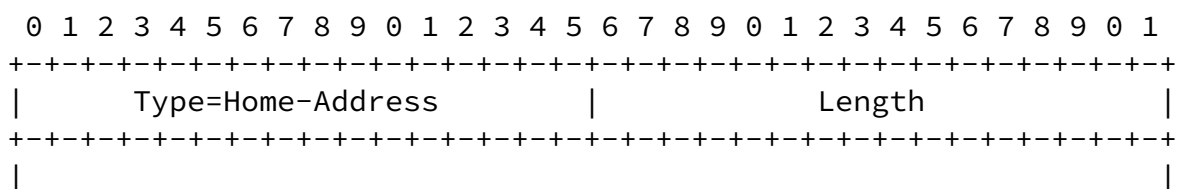
16

Value

Home Agent Address

[8.5](#) Home-Address-TLV

This TLV carries the Home Address assigned to the MN. It is defined as follows:



Home Address

Type

TBD - Home-Address

Length

16

Value

Home Address

8.6 IKE-Authentication-Options-TLV

This TLV carries data related to IKE bootstrapping. If used during the initial MIPv6 bootstrapping procedure, the value field contains the list of peer authentication methods supported by the MN. Otherwise, if used during re-authentication events, the value field contains only the peer authentication method currently in use.

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
Type=IKE-Authentication-Options Length
AuthMethod-1 AuthMethod-2 ...

Type

TBD - IKE-Authentication-Options

Length

Length of this TLV.

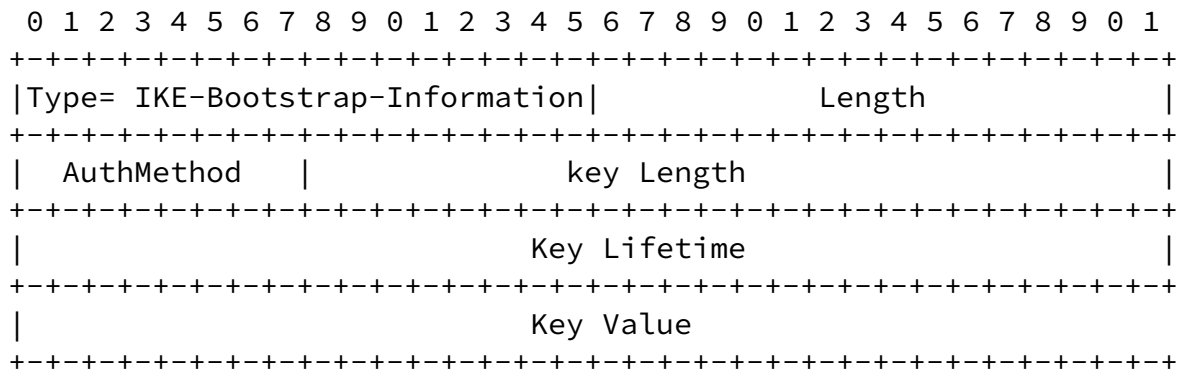
Value

AuthMethod - code corresponding to the authentication method supported for IKE phase 1. All the methods supported by the MN are inserted in order of preference. The following values are defined:

Authentication Method	AuthMethod
PSK (pre-shared key generated by AAAH)	0
AMSK (pre-shared key derived from EAP)	1
CERT (use of certificates)	2

[8.7](#) IKE-Bootstrap-Information-TLV

This TLV carries data related to the set-up of an IPsec Security Association with the Home Agent. It contains the peer authentication method to be used for IKE phase 1 and, eventually, the related cryptographic material (e.g. pre-shared key).



Type

TBD - IKE-Bootstrap-Information

Length

Length of this TLV.

Value

AuthMethod - the authentication method to be used in IKE phase 1. This field can assume a value among those defined in [section 8.6](#) (i.e. PSK, AMSK or CERT).

Internet-Draft

MIPv6 Authorization based on EAP

October 2004

Key Length - the length of the key to be used as pre-shared key for IKE phase 1 authentication. This field must be present if AuthMethod is set to PSK and may be present if AuthMethod is set to AMSK.

Key Lifetime - the lifetime of the key in seconds. A value of all ones means infinite. This field is present only if the AuthMethod field is set to PSK or AMSK.

Key Value - the value of the key. This field is present only if the AuthMethod field is set to PSK.

[8.8](#) Negotiation-Result-TLV

It is defined as follows:

```

  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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Type=Negotiation-Result   |           Length           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Result-Code   |
+---+---+---+---+---+

```

Type

TBD - Result

Length

1

Value

0 = Success
128 = Failure

[8.9](#) Authorization-Lifetime-TLV

It is defined as follows:

```

  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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|  Type= Authorization-Lifetime  |                Length                |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|      Authorization-Lifetime      |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Type

TBD - Authorization-Lifetime

Length

2

Value

The lifetime granted to the MN (in seconds)

[9.](#) Security Considerations

The Mobile IPv6 bootstrapping procedure described in this document assumes the MN and the AAA server of the home domain exchange the necessary parameters exploiting the EAP communication established for network access authentication. Therefore, to secure the bootstrapping procedure, the employed EAP method must support mutual authentication as well as integrity and replay protection.

Moreover, if the pre-shared key needed to bootstrap the IPsec SA with the Home Agent is not derived from the EAP key hierarchy but explicitly delivered to the MN by the AAAH server, the EAP method must also provide confidentiality. Several tunneled and non tunneled EAP methods, like PEAPv2 and EAP-IKEv2, fulfill all of these security requirements.

[10](#). References

[10.1](#) Normative References

- [RFC3775] Johnson, D., Perkins, C. and J. Arkko, "Mobility Support in IPv6", [RFC 3775](#), June 2004.
- [RFC3776] Arkko, J., Devarapalli, V., Dupont, F., "Using IPsec to Protect Mobile IPv6 Signaling between Mobile Nodes and Home Agents", [RFC 3776](#), June 2004.
- [RFC3748] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, H. Levkowetz, "Extensible Authentication Protocol (EAP)", [RFC 3748](#), June 2004.
- [RFC2409] Harkins, D., Carrel, D., "The Internet Key Exchange (IKE)", [RFC 2409](#), November 1998.
- [PEAPv2] Palekar, A. et al., "Protected EAP Protocol (PEAP) Version 2", [draft-josefsson-pppext-eap-tls-eap-08](#) (work in progress), July 2004.

- [EAPKEYFWK] Aboba, B., Simon, D., Arkko, J., Levkowetz, H., "EAP Key Management Framework", [draft-ietf-eap-keying-03](#) (work in progress), July 2004.
- [MIPv6AMSK] Giaretta, G., Guardini, I., Demaria, E., Bournelle, J., Laurent-Maknavicius, M., "Application Master Session Key (AMSK) for Mobile IPv6", [draft-giaretta-mip6-amsk-00](#) (work in progress), September 2004

10.2 Informative References

- [MIPv6PS] Patel, A. et al. "Problem Statement for bootstrapping Mobile IPv6", [draft-ietf-mip6-bootstrap-ps-00](#) (work in progress), July 2004.
- [RFC3753] Manner, J., Kojo, M. "Mobility Related Terminology", [RFC 3753](#), June 2004.
- [RFC3041] Narten, T., Draves, R., "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", [RFC 3041](#), January 2001.
- [AAAH-HA] Giaretta, G., Guardini, I., Demaria, E., Bournelle, J., Lopez, R., "Goals for AAA-HA interface", [draft-giaretta-mip6-aaa-ha-goals-00](#) (work in progress), September 2004

Giaretta, et al. Expires - April 2005 [Page 34]

Internet-Draft MIPv6 Authorization based on EAP October 2004

- [AAAMIPFWK] Yegin, A., "AAA Mobile IPv6 Application Framework", [draft-yegin-mip6-aaa-fwk-00](#) (work in progress), August 2004
- [RFC3084] K. Chan, D. Durham, S. Gai, S. Herzog, K. McCloghrie, F. Reichmeyer, J. Seligson, A. Smith, R. Yavatkar, "COPS Usage for Policy Provisioning", [RFC 3084](#), March 2001.
- [MIPv6APP] Faccin, S., Perkins, C., Le, F., Patil, B., "Diameter Mobile IPv6 Application", [draft-le-aaa-diameter-mobileipv6-03](#) (expired), April 2003.
- [PANA] Forsberg, D. et al., "Protocol for Carrying Authentication for Network Access (PANA)", [draft-ietf-](#)

[pana-pana-04](#) (work in progress), May 2004.

- [RFC3410] Case, J., Mundy, R., Partain, D. and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", [RFC 3410](#), December 2002.
- [EAP-TTLS] Funk, P., Blake-Wilson, S., "EAP Tunneled TLS Authentication Protocol (EAP-TTLS)", [draft-ietf-pppext-eap-ttls-05](#) (work in progress), July 2004.
- [EAP-IKEv2] Tschofenig, H., Kroeselberg, D., Ohba, Y., "EAP IKEv2 Method", [draft-tschofenig-eap-ikev2-03](#), February 2004.
- [EAP-SIM] Haverinen, H. and J. Salowey, "Extensible Authentication Protocol Method for GSM Subscriber Identity Modules (EAP-SIM)", [draft-haverinen-pppext-eap-sim-13](#) (work in progress), April 2004.
- [EAP-AKA] Arkko, J. and H. Haverinen, "EAP-AKA Authentication", [draft-arkko-pppext-eap-aka-12](#) (work in progress), April 2004.
- [EAP-FAST] N.Cam-Winget, D. McGrew, J. Salowey, H.Zhou, "EAP Flexible Authentication via Secure Tunneling (EAP-FAST)", [draft-cam-winget-eap-fast-00.txt](#) (expired), February 2004

Acknowledgments

The authors would like to thank Simone Ruffino, Tom Hiller, Hannes Tschofenig, Rafael Marin Lopez, Hiroyuki Ohnishi, Mayumi Yanagiya, James Kempf and Yoshihiro Ohba for their valuable comments.

Authors' Addresses

Gerardo Giaretta
Telecom Italia Lab
via G. Reiss Romoli, 274
10148 TORINO
Italy
Phone: +39 011 2286904
Email: gerardo.giaretta@tilab.com

Ivano Guardini
Telecom Italia Lab
via G. Reiss Romoli, 274
10148 TORINO
Italy
Phone: +39 011 2285424
Email: ivano.guardini@tilab.com

Elena Demaria
Telecom Italia Lab
via G. Reiss Romoli, 274
10148 TORINO
Italy
Phone: +39 011 2285403
Email: elena.demaria@tilab.com

Julien Bournelle
GET/INT
9 rue Charles Fourier
Evry 91011
France
Email: julien.bournelle@int-evry.fr

Maryline Laurent-Maknavicius
GET/INT
9 rue Charles Fourier
Evry 91011
France
Email: maryline.maknavicius@int-evry.fr

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2004). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

Acknowledgement

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

