Submitted to AAA Working Group
INTERNET DRAFT
1 May 2003

Charles E. Perkins
Ernie Tacsik
Nokia
Thomas Eklund
Xelerated Networks

AAA for IPv6 Network Access

draft-perkins-aaav6-06.txt

Status of This Memo

Abstract

IPv6 nodes (clients) need a way to offer credentials to the AAA infrastructure in order to be granted access to the local network. For IPv6, it will be more efficient and thus reasonable to expect such access controls to be exerted by IPv6 routers, possibly as part of performing their role as DHCPv6 relays. Routers and DHCPv6 servers are expected to work in conjunction with AAA servers to determine whether or not the client's credentials are valid. Routers can exert such network access control by the device of carefully controlling entries in their packet filter and Neighbor Cache.

Contents

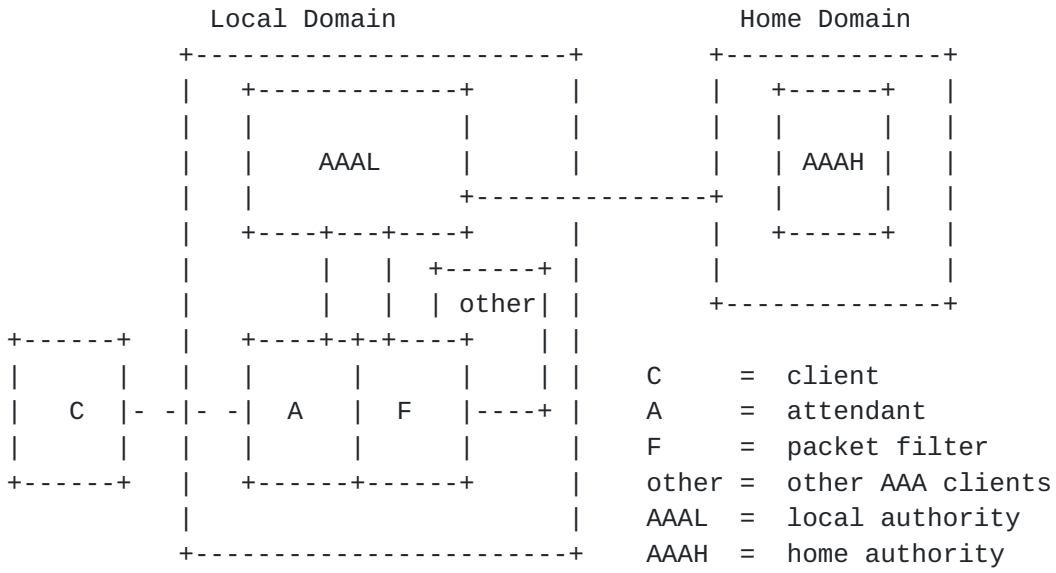
1. Introduction

This document proposes a way for IPv6 nodes (clients) to offer credentials to a local AAA server in order to be granted access to the local network. Whereas for IPv4 it is not clear that routers and DHCP will be equipped to handle such functions, we believe that it is more efficient and thus reasonable to expect such access controls to be exerted by IPv6 routers, possibly as part of performing their role as DHCPv6 relays. Routers and DHCPv6 servers are expected to work in conjunction with AAA servers to determine whether or not the client's credentials are valid.

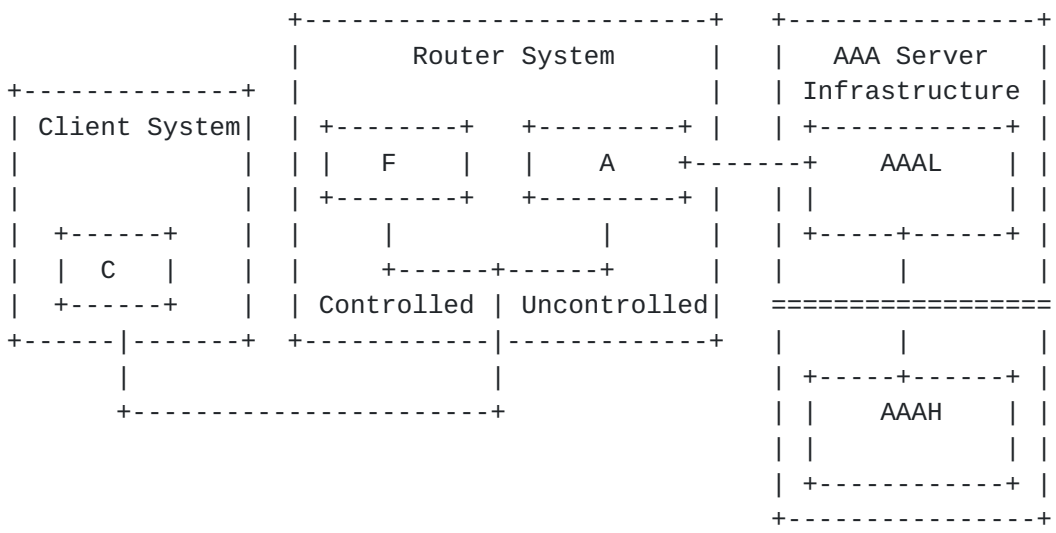
Routers can exert such network access control by carefully controlling entries in their packet filter and Neighbor Cache [8]. If a client does not supply verifiable credentials, then the router SHOULD NOT forward packets to that client. Furthermore, such uncredentialed devices should have no access (or perhaps only very limited access) to the other network links adjacent to the router. Only in this way can a new client be prevented from abusing network connectivity before its authorization is complete.

2. Terminology

This document makes use of many terms defined in recent AAA requirements documents (for example, [6], [5]). The general framework consists of nodes in the following general relationship:



From a system point of view:



The entities in the pictures above are defined as follows:

Client System:
: The client system is the node requesting access to the network.

Client:
: The client is the entity whose authorization is

checked. The client resides on the client system.

Router System:
The router is the node that provides network access to the client. In addition to the usual packet forwarding functionality, the router system typically consists of other functional blocks like the attendant and the packet filter.

Attendant:
The attendant is the entity that extracts identification and authorization data sent by the client and forwards them to AAAL for verification. It is also responsible for making the necessary configuration updates (e.g., to the packet filter, and the router's Neighbor Cache) so that only authorized clients can access the network.

Packet filter:
A packet filter/firewall/security gateway is the entity responsible for disallowing unauthorized datagram traffic. When a client is authorized, the access control list of the filter is updated with the corresponding client's IP address(es).

Controlled and uncontrolled access:
Each network interface of the router can be configured to provide AAA services. When an interface is so configured, all transiting packets are subject to controlled access. If a packet does not pass access control, but is an AAA message addressed to the router, it is given to the Attendant in the uncontrolled access part.

AAAL:
The AAA server in the foreign domain that mediates local access to the AAA infrastructure.

AAAH:
The AAA server in the home domain which is able to authorize each of its clients.

Other nodes:
Other clients that perform some function as a result of the policy received from AAAH, e.g. accounting, QoS, etc.

AAA credential:
Data provided by a client to the AAA server in an authorization request. For example, this can be a

message authentication code constructed using a secret shared between the client and AAAH.

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 [4].

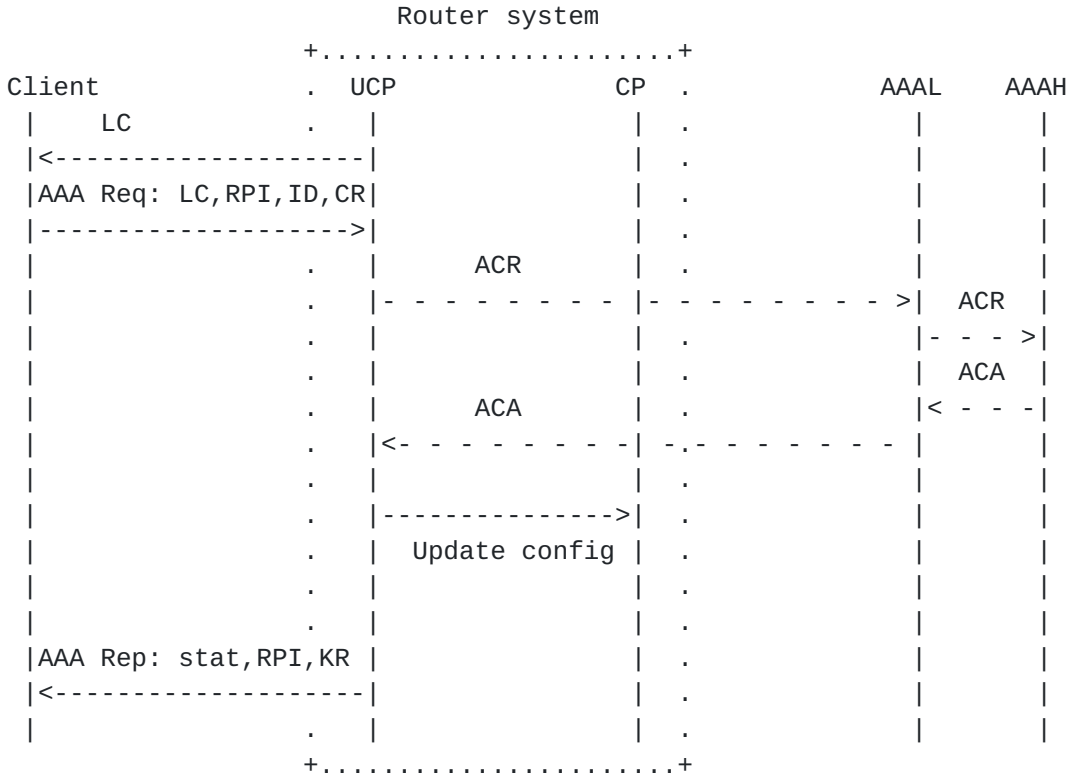
3. General Framework

Using the terminology just introduced, in this section we describe the general framework for our proposals.

3.1. Protocol Description

The client solicits access to the network in conjunction with some protocol. Protocols considered in this document include Stateless Address Autoconfiguration [10] Mobile IPv6 [7], and DHCPv6 [3], as an example of a stateful autoconfiguration service.

If the router has AAA service enabled, a packet received on the interface is made available to the controlled part. The controlled part will only forward traffic that corresponds to an authorized client. All other packets will be dropped except for upstream AAA authorization protocol packets (sent by the client to the router). Such packets are made available to the attendant in the uncontrolled part. The attendant may extract the relevant AAA data and forward them to AAAL. The overall protocol is depicted below.



LC = Local AAA Challenge
RPI = Replay Protection Indicator used between client and AAAH
CR = AAA Credential
ID = Client Identifier
KR = Key Reply
UCP = Uncontrolled part
CP = Controlled part
ACR = AAA Client Request (using an AAA protocol)
ACA = AAA Client Answer (using an AAA protocol)

The figure describes the authorization protocol exchanges in the generic case. The operation of this protocol is initiated by either the attendant or the client. First, the attendant (uncontrolled part in the router) sends a local challenge to the client. The client then constructs an AAA credential which securely binds the local challenge, the client identifier, any replay protection indicator used between the client and AAAH, and other necessary information. The credential is such that it can be verified by AAAH. The client then sends an ``AAA Request'' message containing the credential and the information necessary to verify it. The attendant checks that the local challenge included in the AAA Request is valid, extracts the AAA related information and sends them to AAAL using an appropriate AAA protocol. We label this message, AAA Client Request (ACR). AAAL forwards ACR to AAAH, which verifies the credential

and sends back the result, and any necessary keys. We label this

reply message AAA Client Answer (ACA). The AAAL forwards ACA to the attendant. AAAL may also forward any necessary keys. The attendant extracts the relevant data from ACA and forwards them to the client. The attendant updates configuration of the packet filter (controlled part in the router) so that traffic to and from the client is allowed to pass through.

3.2. Client Identifier

The client identifier has to be in some format that will enable AAAL to identify a suitable AAAH for carrying out all necessary authorization steps.

A Network Access Identifier (NAI) [1] is often used to convey user identity, but IPv6 networks may well be constructed to determine the user's identity based only on the IPv6 address of the user's host. Therefore, the client identifier MAY be a NAI or an IPv6 address. The NAI MAY also identify the user to AAAL, although this is not necessary (e.g., the user part of the NAI may be intelligible only to AAAH).

3.3. Replay Protection

Each participant in the protocol SHOULD verify the freshness of the protocol messages in order to protect itself from replay attacks. Replay protection between AAAL and AAAH, and between AAAL and attendant are handled by the AAA protocol. Therefore, we only need to consider replay protection between the client and the other entities.

The attendant ensures freshness of an AAA Request message from the client by verifying that the local challenge included in the Request is a recent one.

The client and AAAH may use either timestamps or random challenges (nonces) for replay protection. The former is straightforward. The latter is as follows. In the AAA Reply, AAAH sends an AAAH challenge. When the client makes the next AAA Request, it includes this AAAH challenge. It also includes its own client challenge. When AAAH receives this request, it verifies that the AAAH challenge is current. In the reply, AAAH copies the client challenge, and includes a new AAAH challenge. This way, the client can verify the freshness of the reply from AAAH.

If the AAAH challenge in an AAA Request is not valid, or if the client sends an explicit request for an AAAH challenge, AAAH will

reply with a new AAAH challenge. This operation is similar to that for nonces as specified for Mobile IP [9].

3.4. AAA Credential

An AAA credential is created by the client and is verified by AAAH. The creation and verification is based on a security association shared between the client and AAAH. The credential SHOULD securely bind the following pieces of information:

- Client identifier,
- Local AAA challenge, if one was provided by the attendant, and
- Depending on the style of replay protection being used between the client and AAAH, either a timestamp or a pair of challenges.

In certain applications, additional data may be included in the computation of the AAA Credential.

The exact algorithm used to compute the AAA Credential depends on the security association between the client and AAAH. HMAC_MD5 is a suitable algorithm, based on a shared secret between the client and AAAH.

4. Instantiation with Stateless Address Autoconfiguration

In this section, we describe how the general protocol sketched in Section 3 can be used with Stateless Address Autoconfiguration [10].

4.1. Structure of Protocol Messages

We define new ICMPv6 messages to transport AAA data between the client and the attendant. In addition, we defined several options that can be embedded in a AAAv6 Protocol Message. Detailed definitions of these messages are given in Section 9. Here we give a brief description of each AAAv6 protocol message type, and each AAAv6 option. In addition, we also defined an AAAv6 Challenge option to Router Advertisement, enabling the attendant to send a challenge to the client.

4.1.1. AAAv6 Protocol Message types

From client to attendant:

AAA Request: Request for client authorization.

AAA Home Challenge Request: Request for a new challenge from AAAH.

From attendant to client:

AAA Reply: Reply to AAA Request.

AAA Teardown: Indication of termination of the currently active AAA registration. This message is always sent unsolicited to the registered AAA client.

4.1.2. AAA Protocol Message options

Each AAA Protocol Message specifies the AAA options that may accompany it. Currently, the following options are defined.

Security Data: This option is intended to carry security data. Currently, two subtypes are defined.

AAA Credential: Sent by the client; used by AAAH to verify the authorization of the client.

AAAH Authenticator: Sent by AAAH; used by the client to verify the authenticity of AAA Reply.

Client Identifier: This option should enable AAAL to determine the AAAH to which an AAA Request is to be forwarded. Currently, two subtypes are defined: NAI, and IPv6 address.

Generalized Key Reply: This option is used to distribute session keys to be used by the client. Currently several subtypes are defined for both stateless and stateful operation (see sections 9.3.4, 10.6).

Challenge: This option is used to carry nonces used for replay protection. Currently three subtypes are defined:

Local Challenge: Challenge issued by the attendant to the client.

Home Challenge: Challenge issued by AAAH to the client.

Client Challenge: Challenge issued by the client to AAAH.

Timestamp: This option is used to carry timestamp information used for replay protection.

IPv6 Address: This option is used to carry IP address information.

Lifetime: This option indicates the lifetime of an AAA authorization.

5. Protocol Overview

5.1. Basic operation

The basic operation follows the model described in Section 3.1. When an IPv6 client starts up or enter a new subnet, it receives a Router Advertisement with a AAA Challenge option. As is usual, this Router Advertisement is either broadcast periodically, or MAY be sent in response to a Router Solicitation by the client.

The client will construct a tentative IP address and MAY reply with an AAA Request ICMPv6 message with the following options:

- Local Challenge option into which the challenge from the Router Advertisement is copied.
- Either Timestamp option or both AAAH Challenge and Client Challenge options
- Client Identifier option consisting of the client's NAI or some long term IPv6 address, such as the client's home address.
- AAA Credential option constructed by concatenating all of the preceding options and applying the algorithm specified by the security association between the client and AAAH.

When challenges are used for replay protection, the client MUST include the currently advertised AAAH challenge (perhaps as received from AAAH via a previous AAA Reply message) in the AAAH Challenge option, and a random number in the Client Challenge option. If the client does not have an AAAH challenge, it SHOULD send an AAA Home Challenge Request message first (see Section 5.2).

The client MUST perform Duplicate Address Detection (DAD) before sending the AAA Request. The source address of AAA Request MUST be the chosen IPv6 address.

On receiving the Request, the attendant MUST check if the chosen address is already in use. If it is, the attendant MUST send an AAA Reply with code ADDRESS_IN_USE.

Otherwise, the attendant will extract the AAA field values and forward them to AAAL in an ACR message using an AAA protocol, which

is then forwarded to AAAH. The data in each AAA option MUST be conveyed to AAAH by the ACR message. In return, AAAH will construct an ACA message containing information in a suitable form that can be be extracted by the attendant and conveyed to the client in an AAA Reply message with appropriate options. The following options are discussed in this document:

- Either Timestamp option or both AAAH Challenge and Client Challenge options

- One or more Key Reply options

- Lifetime option

- AAAH Authenticator option

For error conditions other than those specifically identified in this document, the attendant or the AAA servers can cause the AAAv6 Request to be denied, by returning the code AAAV6_FAILURE in the AAA Reply message.

We describe AAAH behavior in terms of what the client should eventually receive in the AAA Reply. If the AAA Credential is incorrect, the client MUST receive an AAA Reply with code INVALID_CREDENTIAL. If challenges are used for replay protection, and if the AAAH challenge is absent or invalid, the AAA Reply SHOULD have a code NEW_CHALLENGE, and SHOULD contain an AAAH Challenge option. If timestamps are used for replay protection, and the Timestamp option is absent or invalid, the AAA Reply SHOULD have code INVALID_TIMESTAMP.

AAAH SHOULD choose a validity period for the verification which should be included in the Lifetime option of AAA Reply. If AAAL proposes its own lifetime value (in the ACR message), then the Lifetime option MUST contain the lower of the two values. If AAAH chooses a key to be used between the attendant and the client, that key SHOULD be encoded in a Client-Attendant Key Reply option. If timestamps are used for replay protection, there MUST be a timestamp option. If challenges are used for replay protection, AAAH MUST copy the Client Challenge, and include a new random number in the AAAH Challenge. Finally, AAAH should compute an authenticator, to be included in an AAAH Authenticator option, by concatenating all the preceding options intended for the client, and applying the algorithm specified by the security association between the client and AAAH. In addition, AAAH MAY include information in the ACA message intended for AAAL.

If the status of the request is successful, AAAH will send back an ACA message indicating success to AAAL. AAAL will forward this to

the attendant. If there are any keys distributed by AAAH, AAAL MUST re-encode those keys for the attendant.

The attendant MUST add an entry for the client in its Neighbor Cache and at the same time update the packet filter with the client's IPv6 address when the AAA verification for the client has been successful. The lifetime of these entries MUST be set to the value specified in the ACA message. The attendant will extract all information in the ACA message intended for the client and send them back in an AAA Reply ICMPv6 message. If, in the case of stateful address allocation (e.g., DHCPv6 [3]; see section 7), the source address of the AAA Request was the unspecified address, the corresponding AAA Reply MUST be sent to the all-nodes multicast address. Otherwise the AAA Reply MUST be sent to the source address of the corresponding AAA Request.

The attendant MUST create security associations for the client corresponding to any keys distributed to it by AAAL.

When the client receives an AAA Reply indicating success, it MUST verify the AAAH authenticator and the validity of the replay protection indicator. If verification succeeds, and key reply extensions have been included in the Reply, the client MUST create security associations for the attendant. The client MUST associate the lifetime specified in the Lifetime option with the address that was authorized. When the lifetime is close to expiration, the client SHOULD re-initiate the AAA process.

5.2. Challenge Request

If the client does not have a valid AAAH challenge, it SHOULD send an AAA Home Challenge Request message. This SHOULD include the Client Challenge option and MAY include the Client Identifier option. The AAA Reply SHOULD have code NEW_CHALLENGE, and SHOULD include an AAAH Authenticator option.

5.3. Initiation of the AAA Process

The AAA process can be initiated either from the client or from the attendant. The attendant can initiate the process by sending a Router Advertisement with the AAA Challenge option. The client can initiate the process by sending a Router Solicitation.

5.4. Termination

It is also possible to terminate valid sessions. To terminate a session, the attendant clears the packet filter and sends a AAA

Teardown message to the client which invalidates the IP address. A typical scenario for termination would be in a pre-paid service when the pre-paid amount is used up. The client may request termination by sending an AAA Request message with a zero lifetime.

6. Instantiation with Mobile IPv6

There are two ways to handle Mobile IPv6. First, the client could do the AAA processing when it obtains a care-of address, and then it could send a binding update to the home agent, and possibly to the previous router and other correspondents.

If the home agent and AAAH belong to the same domain, it may be more efficient to bundle the binding update to the home agent in the AAA Request message so that the delay is minimized. We support this possibility by defining a new option called Embedded Data option. The client generates an IPv6 packet containing the binding update to the home agent, but instead of sending it directly, it includes it in the AAA Request as the payload of an Embedded Data option. AAAH will extract the binding update IPv6 packet and send it to the home agent. The home agent SHOULD send the binding acknowledgement back to AAAH so that it can be similarly transported to the client as part of the AAA Reply.

In addition, we define new subtypes to the AAA Generalized Key Reply option so that AAAH could distribute authentication keys for use between the home agent and the mobile node.

7. Instantiation with DHCPv6

In this section we describe how the general protocol sketched in Section 3 can be used with DHCPv6 [3].

Between the client and the server there may also be a DHCP Relay which together with the DHCP server MAY be used to restrict access. The exact behavior of the relay is described in Section 7.4

7.1. Mapping the general protocol

The general protocol messages in Section 3 and the instantiation with stateless autoconfiguration in Section 4 are mapped to DHCP in the following fashion.

- The Local Challenge is sent as an option in the DHCP Advertise message.

- The AAA Request and the Home Challenge Request are sent as options in the DHCP Request message.

- The AAA Reply is sent as options in the DHCP Reply message.

- The AAA Teardown messages is not explicitly sent in any message. Instead the DHCP Reconfigure-init and the DHCP Release messages will be used.

In addition to these two new error values called AAA_Failed indicating failure of the AAA registration attempt and AAA_New_Home_Challenge indicating that a new challenge has been sent to the DHCP client will be defined.

7.2. Mapping the message options

Most of the Protocol Message options in Section 4.1.2 are mapped to DHCP options. The following list specifies which options are needed for DHCP.

- Security Data: As described in Section 4.1.2.

- Client Identifier: This option contains the NAI used by the client. The IPv6 address subtype, since that information is already supplied by DHCPv6.

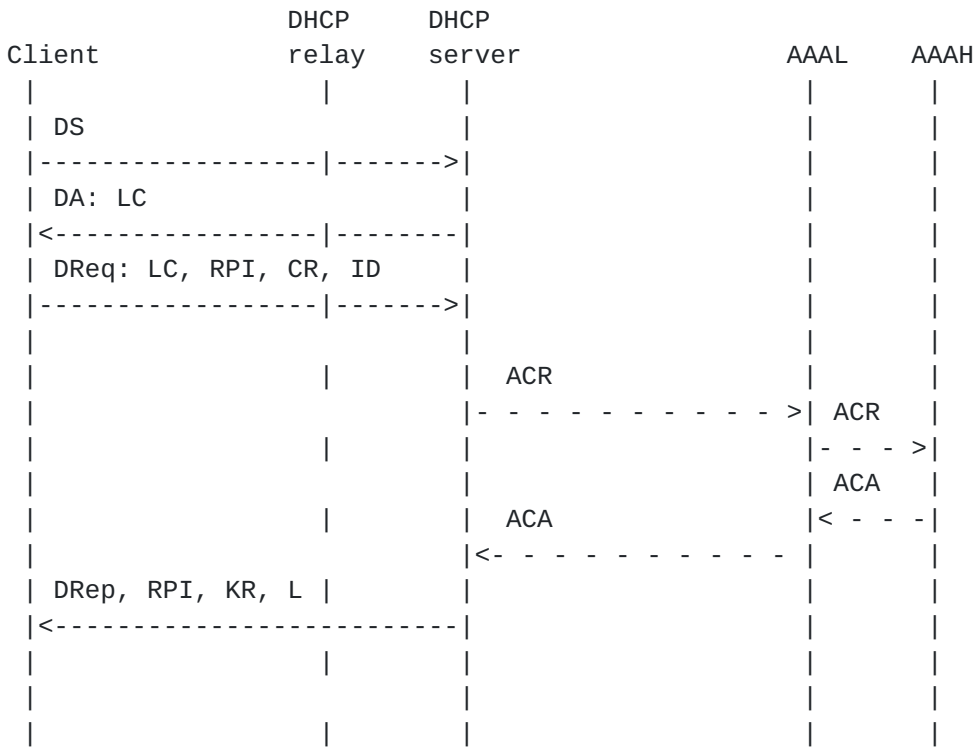
- Generalized Key Reply: As described in Section 4.1.2.

- Challenge: As described in Section 4.1.2.

- Timestamp: As described in Section 4.1.2.

- Lifetime: As described in Section 4.1.2.

The detailed option formats are described in Section 10.



DS = DHCP Solicit
DA = DHCP Advertise
DReq = DHCP Request
DRep = DHCP Reply
LC = Local AAA Challenge
RPI = Replay Protection Indicator used between client and AAAH
CR = AAA Credential
ID = Client Identifier
KR = Key Reply
L = Lifetime
ACR = AAA Client Request (using an AAA protocol)
ACA = AAA Client Answer (using an AAA protocol)

7.3. Protocol Overview

7.3.1. Basic operation

The basic operation follows the model outlined in Section 3.

When the DHCP client starts up in the subnet, it will send a DHCP Solicit as described in the DHCPv6 draft [3]. The DHCP servers receiving the DHCPV6 Solicit reply by sending DHCP Advertise messages containing the Local Challenge option. Either all or none of the DHCP servers MUST include a Local Challenge option in order to avoid

any ambiguities.

The DHCP client will construct a DHCP Request message with the following options added before any authentication option:

- Local Challenge option into which the challenge in the DHCP Advertise message is copied.
- Either Timestamp option or both Home Challenge and Client Challenge options.
- Client Identifier option.
- AAA Credential option.

When challenges are used for replay protection, the DHCP client MUST include its current home challenge in the Home Challenge option, and a random number in the Client Challenge option. If the DHCP client does not have an Home Challenge, it SHOULD request a Home Challenge first as described in Section 8.

On receiving a valid DHCP Request the DHCP server will extract the relevant data and forward them to the AAAL in the ACR message using the AAA protocol. The ACR is then forwarded to the AAAH via the AAA network. In addition to the options relating to AAA sent in the DHCP Request message, the IPv6 address that will be assigned to the DHCP client might be relevant to the AAAH.

In return, AAAH will construct an ACA message and send it to the AAAL via the AAA infrastructure. The AAAL will then forward the ACA message to the DHCP server.

If the ACA message indicates failure the value of the DHCP Reply will be set to AAA_Failed and the DHCP server denies the DHCP address acquisition.

If the ACA message indicates success it will contain information to allow the following DHCP options to be attached to the DHCP Reply message:

- Either Timestamp option or both Home Challenge and Client Challenge options.
- One or more Key Reply options.
- Lifetime option.
- AAAH Authenticator option.

In addition to these options the DHCP server will attach those options needed to satisfy the DHCP client's request.

7.3.2. Termination

The lease can be terminated either by the DHCP client or the DHCP server. The DHCP client terminates the lease by sending a DHCP Release message and waiting for a DHCP Reply. Alternatively, the DHCP server MAY terminate the address lease by sending an Reconfigure-init message by unicast to the DHCP client. The DHCP client will try to reacquire its address lease which the DHCP server then will deny.

7.4. Access Control

The access to the controlled part (CP) of the network can be carried out in three different ways.

In a subnetwork using a DHCP Relay to forward messages between the client and the server the access control MAY be located in the DHCP Relay if the default router and the DHCP relay are also co-located. The DHCP Relay MUST add an entry in its Neighbor Cache when forwarding a DHCP Replay indicating successful allocation of an address. In addition to adding an entry in the Neighbor Cache subnet or site specific filtering rules MAY also be added.

In small sites where there is one DHCP server co-located in the default router the DHCP server MUST add the entries in its neighbor cache and MAY also add subnet or site specific filtering rules.

The filtering rules MAY also be added to suitable locations in the accessed network by some other means, e.g. through AAA interaction.

8. Requesting a Home Challenge

If the AAAv6 client does not have a Home Challenge, it SHOULD request one by sending a AAAv6 Home Challenge Request message. This request message contains authentication data so that the AAAH can eventually ensure that the request comes from an authorized client. The attendant SHOULD relay this request to the AAAL in an extension to a ACR message. The AAAL, if it does not have any challenge values buffered for the AAAv6 client, SHOULD relay the ACR and the request extension to the AAAH.

If the AAAH decides to honor the request, it formulates one or more random challenges, each of which MAY be required to meet certain test conditions agreed upon beforehand between the AAAv6 client and the AAAH. The random challenges are gathered into an extension to an ACA message which is sent to the AAAL. The AAAL then transmits (to the attendant) an ACA containing no more than one of the Home Challenge

values in an ACA challenge reply extension. Finally, the attendant transmits an AAAv6 Home Challenge Reply message to the AAAv6 client.

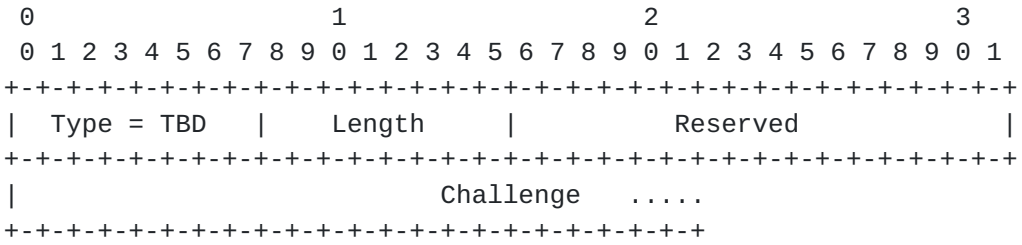
This additional mechanism is intended to enable use with EAP, which requires a challenge value to be acted on by the EAP client. This challenge value has to be generated by the AAAH, and is not available to the client until requested. Thus, it is not possible with this mechanism to handle all authentication signaling needs in a single round trip.

However, depending on the security association between the AAAv6 client and the AAAH, the Home Challenge Request may be used to acquire challenge values for security protocols other than EAP.

9. Message Formats for Stateless Address Autoconfiguration

9.1. AAA Challenge Option

The AAA challenge option is applied to Router Advertisements. If the only purpose is to indicate support for AAAv6 in the router advertisement, the option contains only the type and a length field set to zero.



Type — The Type field identifies the option as being an AAA challenge option and has a value of TBD.

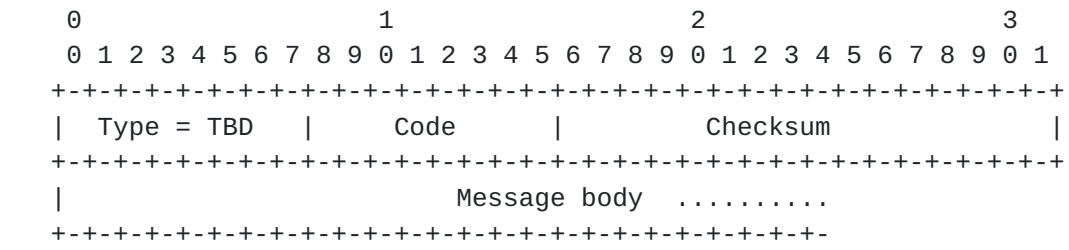
Length — The Length field gives the length measured in octets of this option, including the Type and Length fields.

Reserved — Ignored on reception; sent as 0.

Challenge — This field contains a challenge.

9.2. AAA Protocol Messages

AAA Messages are new ICMPv6 messages. They have the following general structure.



Type The following new types are defined:

AAA Request: TBD

AAA Home Challenge Request: TBD

AAA Reply: TBD

AAA Teardown: TBD

Code The code field depends on the message type. Currently the following Code fields are defined:

For AAA Reply

SUCCESS: 0

NEW_CHALLENGE: 1

ADDRESS_IN_USE: 20

INVALID_CREDENTIAL: 50

INVALID_TIMESTAMP: 51

AAAV6_FAILURE: 52

For AAA Teardown

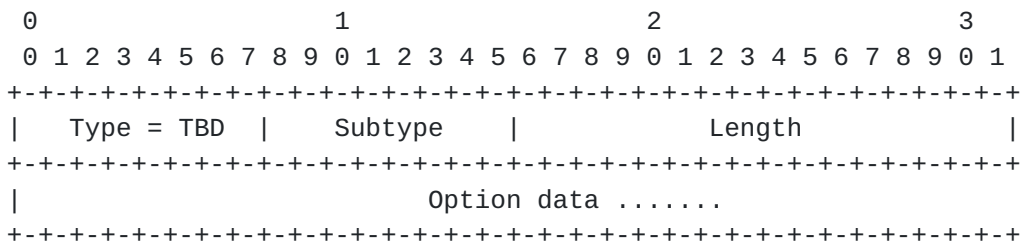
SUCCESS: 0

No Code values are defined for the remaining AAA message types. The Code field MUST be set to zero.

Message body The message body may consist of one or more options.

9.3. AAA Protocol Message Options

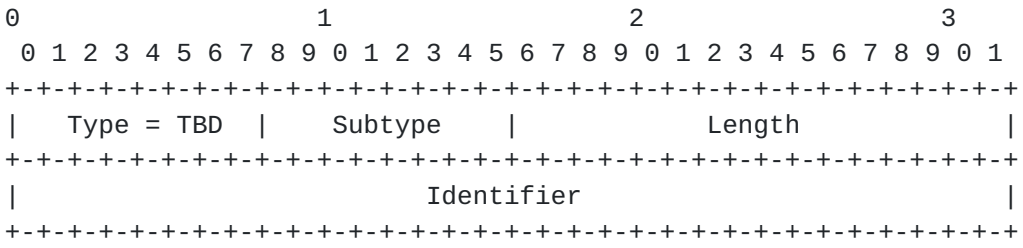
The general structure of an AAA Protocol Message Option is as follows.



Type	The Type field identifies the option. Currently, the following types are supported. The most significant bit of the Type indicates if the option is unskippable (0) or skippable (1).
Subtype	Each option type may be further subdivided. The Subtype field identifies option at the next level of granularity.
Length	The Length field indicates the size of the Option data in octets.
Option data	The format of option data is depends on the type and subtype, and is defined below.

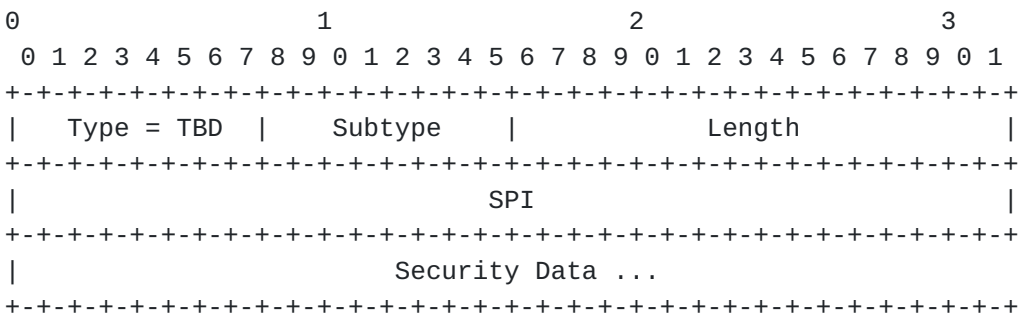
9.3.1. Client Identifier option

The Client Identifier option is used by AAAL to determine the appropriate realm towards which to route the AAA request, and by AAAH in verifying the AAA Credential.



Type	1 (unskippable)
Subtype	Currently two subtypes are defined: NAI (0) and IPv6 address (1)
Length	For subtype 1, the Length should be 16.
Identifier	For subtype 0, this field contains a NAI formatted according to RFC2486 [1]. For subtype 1, this field contains an IPv6 address.

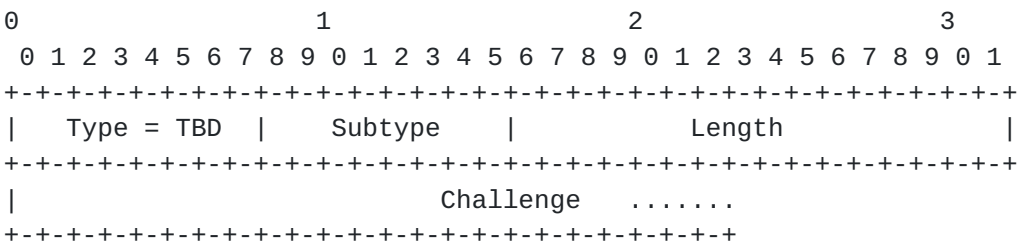
9.3.2. Security Data



Type	2 (unskippable)
Subtype	Currently two subtypes are defined: AAA Credential (0) and AAAH Authenticator (1)
Length	Length of the option in octets, not including the first four octets.
SPI	The security parameter index to be used in interpreting the Security Data.
Security Data	The actual payload.

9.3.3. Challenge

The Challenge option is used to convey nonces for replay protection between various pairs of entities.



Type 3 (unskippable)

Subtype Currently three subtypes are defined:

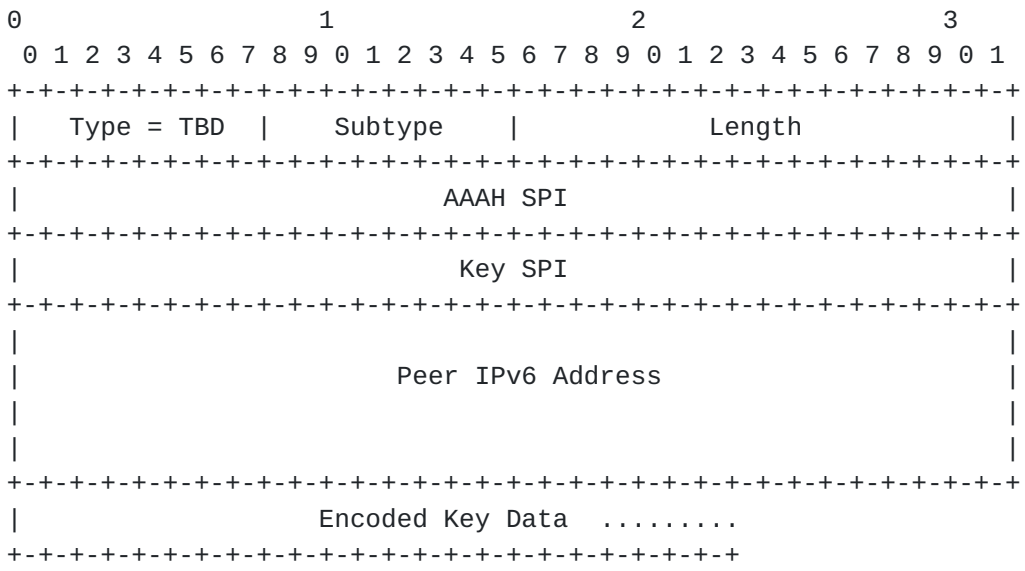
- Local Challenge: Challenge issued by the attendant to the client (0)
- Home Challenge: Challenge issued by AAAH to the client (1)
- Client Challenge: Challenge issued by the client to AAAH (2)

Length Length of the challenge in octets.

Challenge The actual challenge data.

9.3.4. Generalized Key Reply

This option is used to convey keys distributed by AAAH for use between the client and other entities.



Type 4 (unskippable)

Subtype Currently subtypes are defined for three entity pairs:

- Client-Attendant authentication key: Key to be used between the current attendant and the client for IPSec authentication (1)

- Client-Attendant encryption key: Key to be used between the current attendant and the client for IPsec encryption (2)

- MN-HA authentication key: Key to be used between the home agent and the client for IPsec authentication (4)

If the most significant bit of the Subtype value is 1, the ``Peer IPv6 Address'' field is present. Otherwise, it is absent.

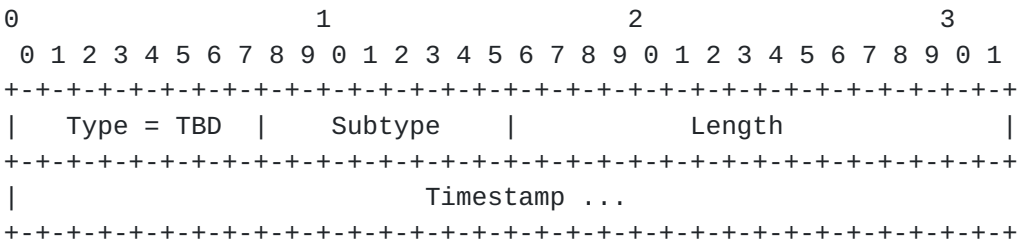
Length Length of the option in octets except the first four octets.

AAAH SPI This field indicates the security association between the client and AAAH which should be used by the client to interpret the Encoded Key Data field.

Key SPI	This field indicates the SPI value for the new security association into which the key should be inserted.
Peer IPv6 Address	When present, this field indicates the IPv6 address of the peer. This is useful when the client does not already know the address to be used. This field is present in subtypes 128 and above.
Encoded Key Data	This field contains the key, along with any other information required by the client to create the security association. The contents of the field MUST be encrypted by AAAH as specified by AAAH SPI.

9.3.5. Timestamp

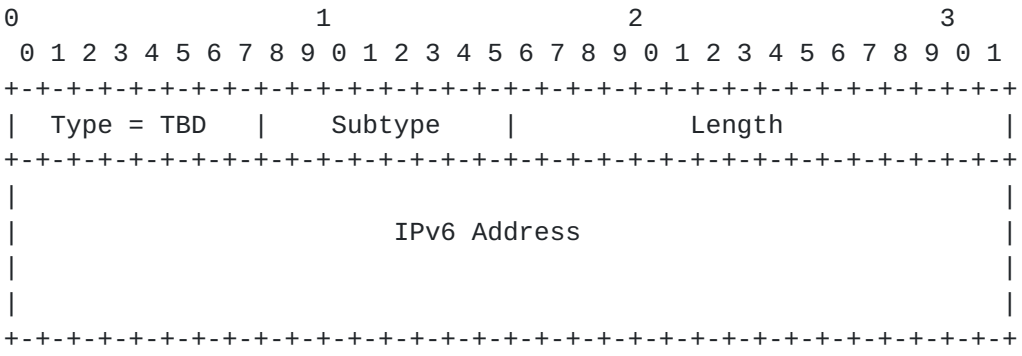
Timestamp field may be used for replay protection between the client and AAAH.



Type	5 (unskippable)
Subtype	Currently no subtypes are defined. Should be zero.
Length	Length of the Timestamp in octets.
Timestamp	Timestamp value in some format mutually intelligible to the client and AAAH

9.3.6. IPv6 Address

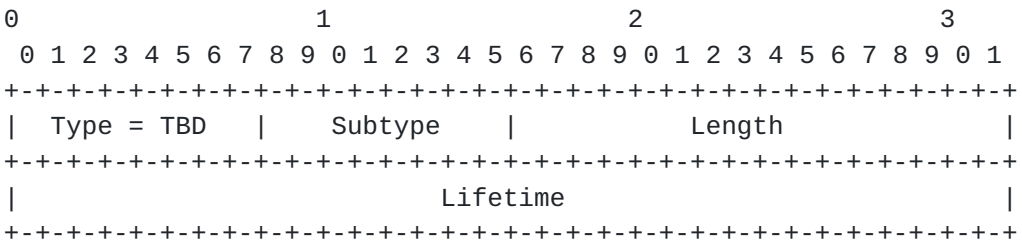
This option is used by the client to convey the IPv6 address it has chosen. There may be other uses for this option, too.



Type	6 (unskippable)
Subtype	Currently no subtypes are defined. Should be zero.
Length	16
IPv6 Address	A valid IPv6 address.

9.3.7. Lifetime

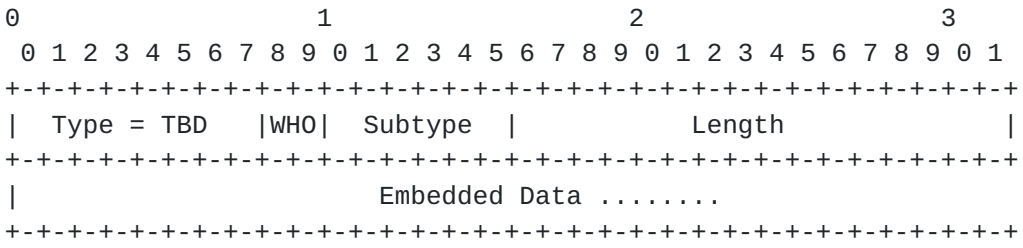
This option is used to indicate the validity period of a successful AAA verification.



Type	7 (unskippable)
Subtype	Currently no subtypes are defined. Should be zero.
Length	4
Lifetime	Lifetime in seconds.

9.3.8. Embedded Data

This option is used to transport specific kinds of embedded data in the AAA messages.



Type 8 (unskippable)

WHO This field indicates who should process the embedded data. It is interpreted as follows (values in binary)

00 - Recipient of the AAA Protocol message (i.e., either the client or the attendant)
01 - AAAL
10 - AAAH
11 - reserved

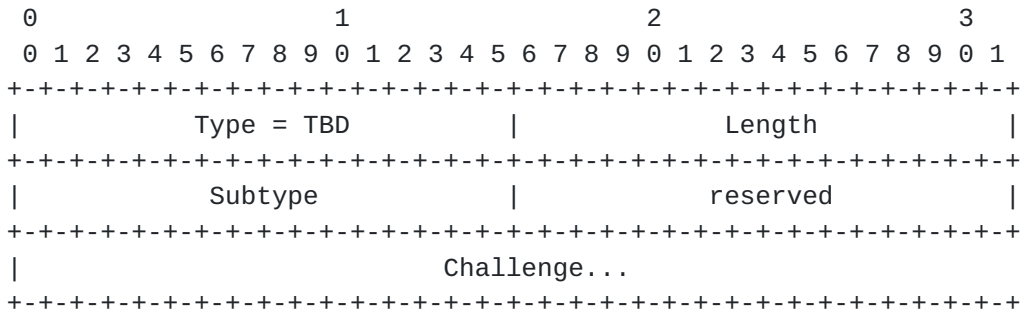
Subtype Currently two subtypes are defined:
(0) -- MN-HA binding update
(1) -- HA-MN binding acknowledgement
(2) -- EAP Request [2]
(3) -- EAP Response [2]

Data The actual embedded data itself. For subtype 0, this MUST be an IPv6 packet addressed to the HA, and containing a binding update. For subtype 1, this MUST be an IPv6 packet addressed to the CoA, and containing a binding acknowledgement from the HA.

For example, to bundle the HA binding update with AAA processing, the client will first generate a binding update, and insert it into an embedded data option of the AAA Request message, with WH = 10 (binary) and Subtype = 0. Based on the value of WH, the attendant will extract the Embedded Data and forward it to AAAH via AAAL. Based on the Subtype, AAAH will forward the binding update to the home agent, and will receive a binding acknowledgement in reply. The attendant will forward the binding acknowledgement in an Embedded Data option to the AAA Reply message, with WH - 00 (binary) and Subtype = 1.

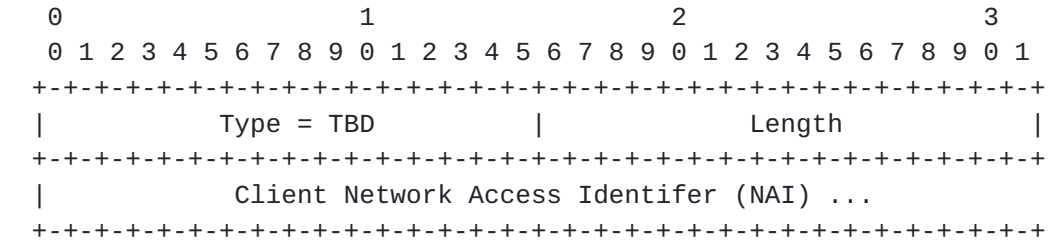
10. Message Formats for Stateful Address Autoconfiguration with DHCPv6

10.1. Challenge option



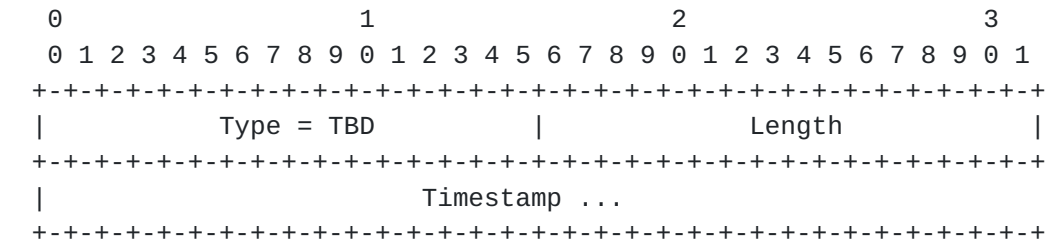
Type	The Type field identifies the option as a Local Challenge option.
Length	The Length of the option in octets not including the Type and Length fields.
Subtype	The Subtype field identifies the type of the challenge. Currently defined subtypes are: - Local Challenge: Challenge issued by the DHCP server to the client, 0 - Home Challenge: Challenge issued by AAAH to the client, 1 - Client Challenge: Challenge issued by the client to AAAH, 2
reserved	0, used for aligning the Challenge field.
Challenge	The actual challenge data.

10.2. Client NAI option



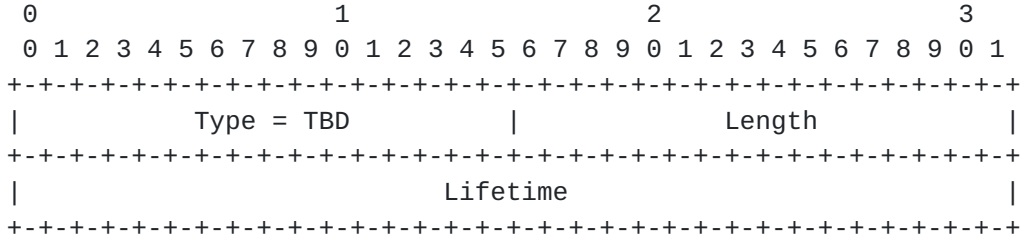
Type	The Type field identifies the option as a Client Identifier option.
Length	The Length of the option in octets not including the Type and Length fields.
Client NAI	The client NAI formatted according to RFC2486 [1]

10.3. Timestamp option



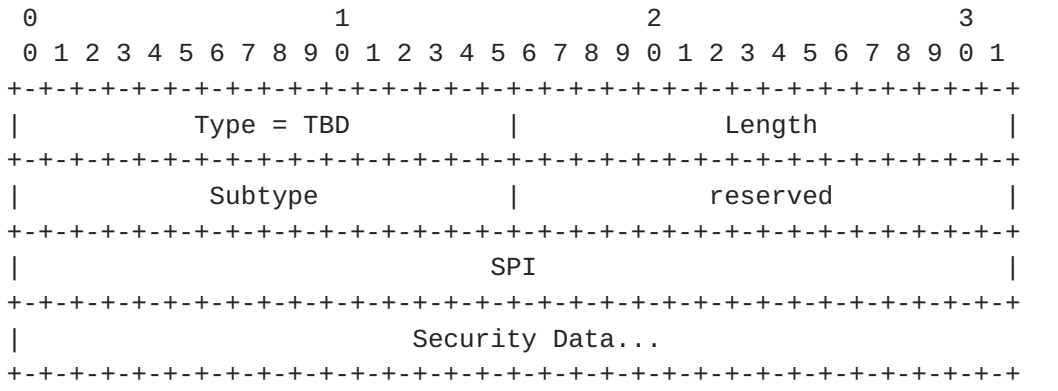
Type	The Type field identifies the option as a Timestamp option.
Length	The Length of the option in octets not including the Type and Length fields.
Timestamp	Timestamp value in some format mutually intelligible to the client and AAAH.

10.4. Lifetime option



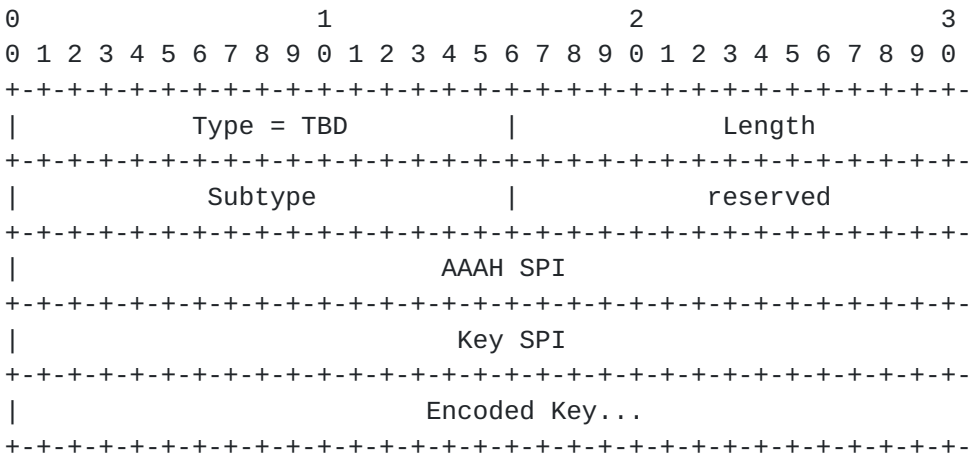
Type	The Type field identifies the option as a Timestamp option.
Length	4
Lifetime	Lifetime in seconds indicating the validity period of a successful AAA verification.

10.5. Security Data option



Type	The Type field identifies the option as a Timestamp option.
Length	The Length of the option in octets not including the Type and Length fields.
Subtype	Currently two subtypes are defined: AAA Credential (0) and AAAH Authenticator (1).
reserved	0, used for aligning the Challenge field.
Security Data	The actual payload.

10.6. Generalized Key Reply option



Type	The Type field identifies the option as a Timestamp option.
Length	The Length of the option in octets not including The Type and Length fields.
Subtype	Currently three pairs of subtypes are defined: Client-Attendant authentication key: Key to be used between the current attendant and the client for IPSec authentication (1) Client-Attendant encryption key: Key to be used between the current attendant and the client for IPsec encryption (2) MN-HA authentication key: Key to be used between The home agent and the client for Ipsec authentication (4)
reserved	0, used for aligning the next field.
AAAH SPI	This field indicates the security association between the client and AAAH which should be used by the client to interpret the Encoded Key Data field.
Key SPI	This field indicates the SPI value for the new security association into which the key should be inserted.
Encoded Key	This field contains the key, along with any other information required by the client to create the security association. The contents of the field

MUST be encrypted by AAAH as specified by AAAH SPI.

11. Security Considerations

Source address based packet filtering does not guarantee that only authorized clients will be able to send out traffic through the router. An attacker can fake the source address of IP packets. In situations where strong protection is needed, clients should be required to use an IPSec AH tunnel to the router.

12. Open Issues and Discussion

12.1. Packet Service Filter

Future work may be needed to determine how services can be updated dynamically based on the authorized services. A typical service filter will permit/deny a filter rule based on upper layer information for the authenticated IP address.

The attendant could prepare an ACL with service filters based on the AAA response for the authenticated services for the different UDP/TCP ports.

12.2. Use of Destination Options

An alternative to defining new ICMPv6 messages and associated options will be to define new IPv6 destination options for all AAA related payload. One disadvantage is that the length of destination options is limited by the 8-bit length field. An advantage is that embedded data may be transported more naturally using either a Routing Header or IP-in-IP encapsulation, thereby avoiding the need for something like the Embedded Data option.

12.3. AAAL

If QoS or other traffic parameters affecting the whole site are received from the AAAH, the AAAL SHOULD have some means to enforce these. In this case the AAAL SHOULD also enforce some form of filtering separate from the DHCP infrastructure.

12.4. Other

How are the AAA Credentials computed?

Do we need the padding in the DHCPv6 options?

Contributors

Patrik Flykt (Nokia)

References

[1] B. Aboba and M. Beadles. The Network Access Identifier. Request for Comments (Proposed Standard) 2486, Internet Engineering Task Force, January 1999.

[2] L. Blunk and J. Vollbrecht. PPP Extensible Authentication Protocol (EAP). Request for Comments (Proposed Standard) 2284, Internet Engineering Task Force, March 1998.

[3] J. Bound, C. Perkins, M. Carney, and R. Droms. Dynamic Host Configuration Protocol for IPv6 (DHCPv6) (work in progress). Internet Draft, Internet Engineering Task Force, January 2001.

[4] S. Bradner. Key words for use in RFCs to Indicate Requirement Levels. Request for Comments (Best Current Practice) 2119, Internet Engineering Task Force, March 1997.

[5] B. Aboba et al. Criteria for Evaluating AAA Protocols for Network Access. Request for Comments (Proposed Standard) 2989, Internet Engineering Task Force, November 2000.

[6] S. Glass, T. Hiller, S. Jacobs, and C. Perkins. Mobile IP Authentication, Authorization, and Accounting Requirements. Request for Comments (Proposed Standard) 2977, Internet Engineering Task Force, October 2000.

[7] D. Johnson and C. Perkins. Mobility Support in IPv6 (work in progress). draft-ietf-mobileip-ipv6-15.txt, October 2001.

[8] T. Narten, E. Nordmark, and W. Simpson. Neighbor Discovery for IP Version 6 (IPv6). Request for Comments (Draft Standard) 2461, Internet Engineering Task Force, December 1998.

[9] C. Perkins. IP Mobility Support. Request for Comments (Proposed Standard) 3220, Internet Engineering Task Force, December 2001.

[10] S. Thomson and T. Narten. IPv6 Stateless Address Autoconfiguration. Request for Comments (Draft Standard) 2462, Internet Engineering Task Force, December 1998.

Addresses

Questions about this memo can be directed to the authors:

Ernie Tacsik
Nokia Inc.
6000 Connection Drive 3:1000
Irving, Texas 75039
USA

Phone: +1 972 894 4044
E-mail: ernie.tacsik@nokia.com
Fax: +1 972 894 5525

Thomas Eklund
Xelerated Networks
Regeringsgatan 67

103 86 Stockholm
Sweden
Phone: +46 8 50625753
E-mail: thomas.eklund@xelerated.com
Fax: +46 8 54553211

Charles E. Perkins
Communications Systems Lab
Nokia Research Center
313 Fairchild Drive
Mountain View, California 94043
USA
Phone: +1-650 625-2986
EMail: charliep@iprg.nokia.com
Fax: +1 650 625-2502