Mobile Ad hoc Networking (MANET)	U. Herberg
Internet-Draft	Fujitsu Laboratories of America
Intended status: Standards Track	T. Clausen
Expires: March 09, 2012	LIX, Ecole Polytechnique
	September 06, 2011

MANET Cryptographical Signature TLV Definition
draft-ietf-manet-packetbb-sec-06

Abstract

This document describes general and flexible TLVs (type-length-value structure) for representing cryptographic signatures as well as timestamps, using the generalized MANET packet/message format [RFC5444]. It defines two Packet TLVs, two Message TLVs, and two Address Block TLVs, for affixing cryptographic signatures and timestamps to a packet, message and address, respectively.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

This document specifies: Section 12:

- *two TLVs for carrying cryptographic signatures and timestamps in packets, messages, and address blocks as defined by [RFC5444],
- *a generic framework for calculating cryptographic signatures, accounting (for Message TLVs) for mutable message header fields (<msg-hop-limit> and <msg-hop-count>), where these fields are present in messages.

This document requests from IANA:

- *allocations for these Packet, Message, and Address Block TLVs from the 0-223 Packet TLV range, the 0-127 Message TLV range and the 0-127 Address Block TLV range from [RFC5444],
- *creation of two IANA registries for recording code points for hash function and signature calculation, respectively.

Finally, this document defines, in

- *one common method for generating signatures as a cryptographic function, calculated over the hash value of the content to be signed.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].
This document uses the terminology and notation defined in [RFC5444]. In particular, the following TLV fields from [RFC5444] are used in this specification:

<msg-hop-limit> - hop limit of a message, as specified in Section 5.2 of [RFC5444].

<msg-hop-count> - hop count of a message, as specified in Section 5.2 of [RFC5444].

<length> - length of a TLV in octets, as specified in Section 5.4.1 of [RFC5444].

3. Applicability Statement

MANET routing protocols using the format defined in [RFC5444] are accorded the ability to carry additional information in control messages and packets, through inclusion of TLVs. Information so included MAY be used by a MANET routing protocol, or by an extension of a MANET routing protocol, according to its specification.
This document specifies how to include a cryptographic signature for a packet, a message, and addresses in address blocks within a message, by way of such TLVs. This document also specifies how to treat "mutable" fields, specifically the <msg-hop-count> and <msg-hop-limit> fields, if present in the message header when calculating signatures, such that the resulting signature can be correctly verified by any recipient, and how to include this signature.
This document describes a generic framework for creating signatures, and how to include these signatures in TLVs. In Section 12, an example method for calculating such signatures is given, using a cryptographic function over the hash value of the content to be signed.

4. Security Architecture

Basic MANET routing protocol specifications are often "oblivious to security", however have a clause allowing a control message to be rejected as "badly formed" prior to it being processed or forwarded. MANET routing protocols such as [RFC6130] and [OLSRv2] recognize external reasons (such as failure to verify a signature) for rejecting a message as "badly formed", and therefore "invalid for processing". This architecture is a result of the observation that with respect to security in MANETs, "one size rarely fits all" and that MANET routing protocol deployment domains have varying security requirements ranging from "unbreakable" to "virtually none". The virtue of this approach is

that MANET routing protocol specifications (and implementations) can remain "generic", with extensions providing proper deployment-domain specific security mechanisms.
The MANET routing protocol "security architecture", in which this specification situates itself, can therefore be summarized as follows:

- Security-oblivious MANET routing protocol specifications, with a clause allowing an extension to reject a message (prior to processing/forwarding) as "badly formed".

- MANET routing protocol security extensions, rejecting messages as "badly formed", as appropriate for a given deployment-domain specific security requirement.

- Code-points and an exchange format for information, necessary for specification of such MANET routing protocol security extensions.

This document addresses the last of these issues, by specifying a common exchange format for cryptographic signatures, making reservations from within the Packet TLV, Message TLV, and Address Block TLV registries of [RFC5444], to be used (and shared) among MANET routing protocol security extensions.
For the specific decomposition of a signature into a cryptographic function over a hash value, specified in Section 12, this document establishes two IANA registries for code-points for hash functions and cryptographic functions adhering to [RFC5444].
With respect to [RFC5444], this document:

- is intended to be used in the non-normative, but intended, mode of use described in Appendix B of [RFC5444].

- is a specific example of the Security Considerations section of [RFC5444] (the authentication part).

5. Overview and Functioning

This document specifies a syntactical representation of security related information for use with [RFC5444] addresses, messages, and packets, as well as establishes IANA registrations and registries.
Moreover, this document provides guidelines how MANET routing protocols and MANET routing protocol extensions, using this specification, should treat Signature and Timestamp TLVs, and mutable fields in messages.
This specification does not represent a stand-alone protocol; MANET routing protocols and MANET routing protocol extensions, using this specification, MUST provide instructions as to how to handle packets, messages and addresses with security information, associated as specified in this document.
This document requests assignment of TLV types from the registries defined for Packet, Message and Address Block TLVs in [RFC5444]. When a TLV type is assigned from one of these registries, a registry for "Type

Extensions" for that TLV type is created by IANA. This document utilizes these "Type Extension" registries so created, in order to specify internal structure (and accompanying processing) of the <value> field of a TLV.
For example, and as defined in this document, a SIGNATURE TLV with Type Extension = 0 specifies that the <value> field has no pre-defined internal structure, but is simply a sequence of octets. A SIGNATURE TLV with Type Extension = 1 specifies that the <value> field has a pre-defined internal structure, and defines its interpretation (specifically, the <value> field consists of a cryptographic operation over a hash value, with fields indicating which hash function and cryptographic operation has been used, specified in Section 12).
Other documents may request assignments for other Type Extensions, and must if so specify their internal structure (if any) and interpretation.

6. General Signature TLV Structure

The value of the Signature TLV is:

```
<value> := <signature-value>
```

where:

<signature-value> is a field, of <length> octets, which contains the information, to be interpreted by the signature verification process, as specified by the Type Extension.

Note that this does not stipulate how to calculate the <signature-value>, nor the internal structure hereof, if any; such MUST be specified by way of the Type Extension for the SIGNATURE TLV type, see Section 13. This document specifies two such type-extensions, for signatures without pre-defined structures, and for signatures constructed by way of a cryptographic operation over a hash-value.

7. General Timestamp TLV Structure

The value of the Timestamp TLV is:

```
<value> := <time-value>
```

where:

<time-value> is an unsigned integer field, of length <length>, which contains the timestamp.

Note that this does not stipulate how to calculate the <time-value>, nor the internal structure hereof, if any; such MUST be specified by

way of the Type Extension for the TIMESTAMP TLV type, see Section 13.

A timestamp is essentially "freshness information". As such, its setting and interpretation is to be determined by the MANET routing protocol, or MANET routing protocol extension, that uses the timestamp, and may, e.g., correspond to a UNIX-timestamp, GPS timestamp or a simple sequence number.

8. Packet TLVs

Two Packet TLVs are defined, for including the cryptographic signature of a packet, and for including the timestamp indicating the time at which the cryptographic signature was calculated.

8.1. Packet SIGNATURE TLV

A Packet SIGNATURE TLV is an example of a Signature TLV as described in Section 6.
The following considerations apply:

- As packets defined in [RFC5444] are never forwarded by routers, no special considerations are required regarding mutable fields (e.g. <msg-hop-count> and <msg-hop-limit>), if present, when calculating the signature.

- Any Packet SIGNATURE TLVs already present in the Packet TLV block MUST be removed before calculating the signature, and the Packet TLV block size MUST be recalculated accordingly. The TLVs can be restored after having calculated the signature value.

The rationale for removing any Packet SIGNATURE TLV already present prior to calculating the signature is that several signatures may be added to the same packet, e.g., using different signature functions.

8.2. Packet TIMESTAMP TLV

A Packet TIMESTAMP TLV is an example of a Timestamp TLV as described in Section 7. If a packet contains a TIMESTAMP TLV and a SIGNATURE TLV, the TIMESTAMP TLV SHOULD be added to the packet before any SIGNATURE TLV, in order that it be included in the calculation of the signature.

9. Message TLVs

Two Message TLVs are defined, for including the cryptographic signature of a message, and for including the timestamp indicating the time at which the cryptographic signature was calculated.

9.1. Message SIGNATURE TLV

A Message SIGNATURE TLV is an example of a Signature TLV as described in Section 6. When determining the <signature-value> for a message, the following considerations must be applied:

- The fields <msg-hop-limit> and <msg-hop-count>, if present, MUST both be assumed to have the value 0 (zero) when calculating the signature.

- Any Message SIGNATURE TLVs already present in the Message TLV block MUST be removed before calculating the signature, and the message size as well as the Message TLV block size MUST be recalculated accordingly. Removed SIGNATURE TLVs SHOULD be restored after having calculated the signature value.

The rationale for removing any Message SIGNATURE TLV already present prior to calculating the signature is that several signatures may be added to the same message, e.g., using different signature functions.

9.2. Message TIMESTAMP TLV

A Message TIMESTAMP TLV is an example of a Timestamp TLV as described in Section 7. If a message contains a TIMESTAMP TLV and a SIGNATURE TLV, the TIMESTAMP TLV SHOULD be added to the message before the SIGNATURE TLV, in order that it be included in the calculation of the signature.

10. Address Block TLVs

Two Address Block TLVs are defined, for associating a cryptographic signature to an address, and for including the timestamp indicating the time at which the cryptographic signature was calculated.

10.1. Address Block SIGNATURE TLV

An Address Block SIGNATURE TLV is an example of a Signature TLV as described in Section 6. The signature is calculated over the address, concatenated with any other values, for example, any other TLV value that is associated with that address. A MANET routing protocol or MANET routing protocol extension using Address Block SIGNATURE TLVs MUST specify how to include any such concatenated attribute of the address in the verification process of the signature.

10.2. Address Block TIMESTAMP TLV

An Address Block TIMESTAMP TLV is an example of a Timestamp TLV as described in Section 7. If both a TIMESTAMP TLV and a SIGNATURE TLV are associated with an address, the timestamp value should be considered when calculating the value of the signature.

11. Signature: Basic

The basic signature proposed, represented by way of a SIGNATURE TLV with Type Extension = 0, is a simple bit-field containing the cryptographic signature. This assumes that the mechanism stipulating how signatures are calculated and verified is established outside of this specification, e.g., by way of administrative configuration or external out-of-band signaling. Thus, the <signature-value> for when using Type Extension = 0 is:

```
<signature-value> := <signature-data>
```

where:

<signature-data> is an unsigned integer field, of length <length>, which contains the cryptographic signature.

12. Signature: Cryptographic Function over a Hash Value

One common way of calculating a signature is applying a cryptographic function on a hash value of the content. This decomposition is specified in the following, using a Type Extension = 1 in the Signature TLVs.

12.1. General Signature TLV Structure

The following data structure allows representation of a cryptographic signature, including specification of the appropriate hash function and cryptographic function used for calculating the signature:

```
<signature-value> := <hash-function>
                     <cryptographic-function>
                     <key-index>
                     <signature-data>
```

where: Section 13.

<hash-function> is an 8-bit unsigned integer field specifying the hash function.

<cryptographic-function> is an 8-bit unsigned integer field specifying the cryptographic function.

<key-index> is an 8-bit unsigned integer field specifying the key index of the key which was used to sign the message, which allows unique identification of different keys with the same originator. It is the responsibility of each key originator to make sure that actively used keys that it issues have distinct key indices and that

all key indices have a value not equal to 0x00. The value 0x00 is reserved for a pre-installed, shared key.

<signature-data> is an unsigned integer field, whose length is <length> - 3, and which contains the cryptographic signature.

The version of this TLV, specified in this section, assumes that calculating the signature can be decomposed into:

signature-value = cryptographic-function(hash-function(content))

The hash function and the cryptographic function correspond to the entries in two IANA registries, set up by this specification in

12.1.1. Rationale

The rationale for separating the hash function and the cryptographic function into two octets instead of having all combinations in a single octet - possibly as TLV type extension - is twofold: First, if further hash functions or cryptographic functions are added in the future, the number space might not remain continuous. More importantly, the number space of possible combinations would be rapidly exhausted. As new or improved cryptographic mechanism are continuously being developed and introduced, this format should be able to accommodate such for the foreseeable future.
The rationale for not including a field that lists parameters of the cryptographic signature in the TLV is, that before being able to validate a cryptographic signature, routers have to exchange or acquire keys (e.g. public keys). Any additional parameters can be provided together with the keys in that bootstrap process. It is therefore not necessary, and would even entail an extra overhead, to transmit the parameters within every message. One implicitly available parameter is the length of the signature, which is <length> - 3, and which depends on the choice of the cryptographic function.

12.2. Considerations for Calculating the Signature

In the following, considerations are listed, which MUST be applied when calculating the signature for Packet, Message and Address SIGNATURE TLVs, respectively.

12.2.1. Packet SIGNATURE TLV

When determining the <signature-value> for a Packet, the signature is calculated over the three fields <hash-function>, <cryptographic-function> and <key-index> (in that order), concatenated with the entire Packet, including the packet header, all Packet TLVs (other than Packet SIGNATURE TLVs) and all included Messages and their message headers, in accordance with Section 8.1.

12.2.2. Message SIGNATURE TLV

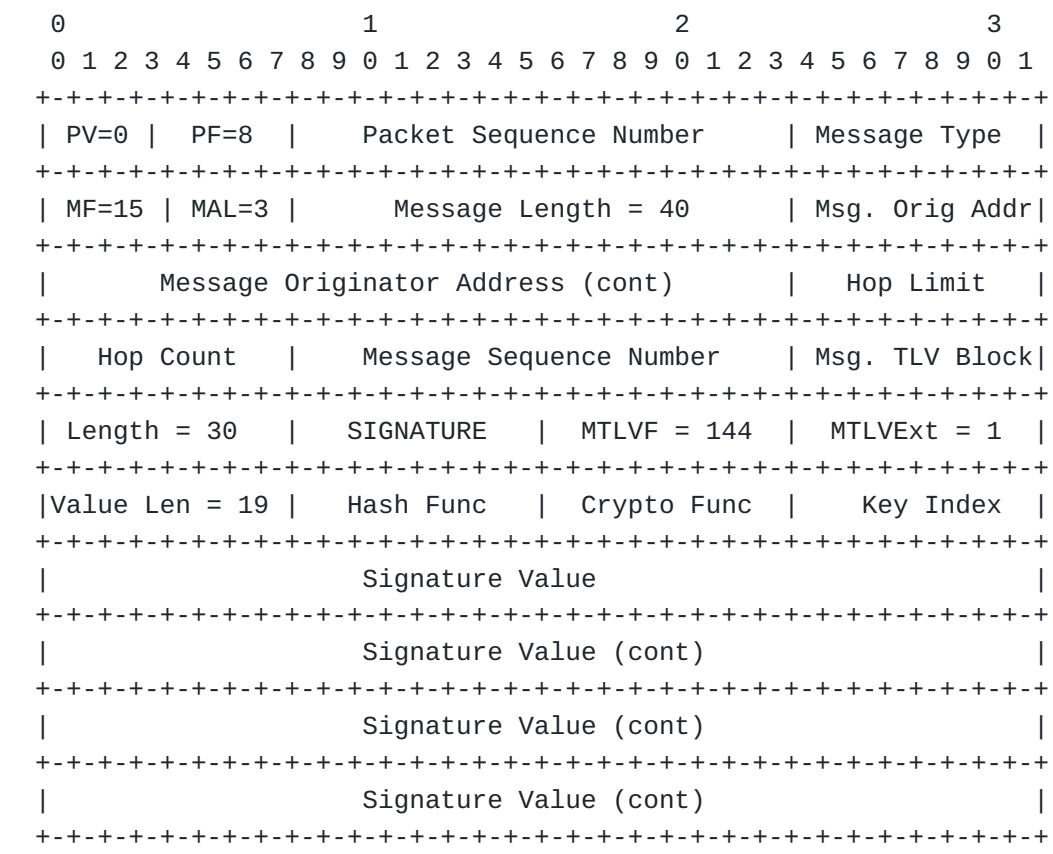
When determining the <signature-value> for a message, the signature is calculated over the three fields <hash-function>, <cryptographic-function>, and <key-index> (in that order), concatenated with the entire message. The considerations in Section 9.1 MUST be applied.

12.2.3. Address Block SIGNATURE TLV

When determining the <signature-value> for an address, the signature is calculated over the three fields <hash-function>, <cryptographic-function>, and <key-index> (in that order), concatenated with the address, concatenated with any other values, for example, any other TLV value that is associated with that address. A MANET routing protocol or MANET routing protocol extension using Address Block SIGNATURE TLVs MUST specify how to include any such concatenated attribute of the address in the verification process of the signature. The considerations in Section 10.2 MUST be applied.

12.3. Example of a Signed Message

The sample message depicted in Figure 5 is derived from appendix D of [RFC5444]. The message contains a SIGNATURE Message TLV, with the value representing a 16 octet long signature of the whole message. The type extension of the Message TLV is 1, for the specific decomposition of a signature into a cryptographic function over a hash value, as specified in Section 12.



13. IANA Considerations

This specification defines:

- two Packet TLV types, which MUST be allocated from the 0-223 range of the "Assigned Packet TLV Types" repository of [RFC5444] as specified in Table 1,

- two Message TLV types, which MUST be allocated from the 0-127 range of the "Assigned Message TLV Types" repository of [RFC5444] as specified in Table 2,

- two Address Block TLV types, which MUST be allocated from the 0-127 range of the "Assigned Address Block TLV Types" repository of [RFC5444] as specified in Table 3.

This specification requests:

- creation of type extension registries for these TLV types with initial values as in Table 1 to Table 3.

IANA is requested to assign the same numerical value to the Packet TLV, Message TLV and Address Block TLV types with the same name.

The following terms are used with the meanings defined in [BCP26]: "Namespace", "Assigned Value", "Registration", "Unassigned", "Reserved", "Hierarchical Allocation", and "Designated Expert".
The following policies are used with the meanings defined in [BCP26]: "Private Use", "Expert Review", and "Standards Action".

13.1. Expert Review: Evaluation Guidelines

For the registries for TLV type extensions where an Expert Review is required, the designated expert SHOULD take the same general recommendations into consideration as are specified by [RFC5444].
For the Timestamp TLV, the same type extensions for all Packet, Message and Address TLVs SHOULD be numbered identically.

13.2. Packet TLV Type Registrations

IANA is requested to make allocations from the "Packet TLV Types" namespace of [RFC5444] for the Packet TLVs specified in Table 1.

Name	Type	Type Extension	Description
SIGNATURE	TBD1	0	Signature of a packet
		1	Signature, decomposed into cryptographic function over a hash value, as specified in Section 12 in this document.
		2-223	Expert Review
		224-255	Experimental Use
TIMESTAMP	TBD2	0	Unsigned timestamp of arbitrary length, given by the TLV length field. The MANET routing protocol has to define how to interpret this timestamp
		1-223	Expert Review
		224-255	Experimental Use

Packet TLV types

13.3. Message TLV Type Registrations

IANA is requested to make allocations from the "Message TLV Types" namespace of [RFC5444] for the Message TLVs specified in Table 2.

Name	Type	Type Extension	Description
SIGNATURE	TBD3	0	Signature of a message
		1	

Name	Type	Type Extension	Description
			Signature, decomposed into cryptographic function over a hash value, as specified in Section 12 in this document.
		2-223	Expert Review
		224-255	Experimental Use
TIMESTAMP	TBD4	0	Unsigned timestamp of arbitrary length, given by the TLV length field.
		1-223	Expert Review
		224-255	Experimental Use

Message TLV types

13.4. Address Block TLV Type Registrations

IANA is requested to make allocations from the "Address Block TLV Types" namespace of [RFC5444] for the Packet TLVs specified in Table 3.

Name	Type	Type Extension	Description
SIGNATURE	TBD5	0	Signature of an object (e.g. an address)
		1	Signature, decomposed into cryptographic function over a hash value, as specified in Section 12 in this document.
		2-223	Expert Review
		224-255	Experimental Use
TIMESTAMP	TBD6	0	Unsigned timestamp of arbitrary length, given by the TLV length field.
		1-223	Expert Review
		224-255	Experimental Use

Address Block TLV types

13.5. Hash Function

IANA is requested to create a new registry for hash functions that can be used when creating a signature, as specified in Section 12 of this document. The initial assignments and allocation policies are specified in Table 4.

Hash function value	Algorithm	Description
0	none	The "identity function": the hash value of an object is the object itself
1-223		Expert Review
224-255		Experimental Use

Hash-Function registry

13.6. Cryptographic Algorithm

IANA is requested to create a new registry for the cryptographic function, as specified in Section 12 of this document. Initial assignments and allocation policies are specified in Table 5.

Cryptographic function value	Algorithm	Description
0	none	The "identity function": the value of an encrypted hash is the hash itself
1-223		Expert Review
224-255		Experimental Use

Cryptographic function registry

14. Security Considerations

This document does not specify a protocol. It provides a syntactical component for cryptographic signatures of messages and packets as defined in [RFC5444]. It can be used to address security issues of a MANET routing protocol or MANET routing protocol extension. As such, it has the same security considerations as [RFC5444].
In addition, a MANET routing protocol or MANET routing protocol extension that uses this specification MUST specify the usage as well as the security that is attained by the cryptographic signatures of a message or a packet.
As an example, a MANET routing protocol that uses this component to reject "badly formed" messages if a control message does not contain a valid signature, SHOULD indicate the security assumption that if the signature is valid, the message is considered valid. It also SHOULD indicate the security issues that are counteracted by this measure (e.g. link or identity spoofing) as well as the issues that are not counteracted (e.g. compromised keys).

15. Acknowledgements

The authors would like to thank Bo Berry (Cisco), Alan Cullen (BAE), Justin Dean (NRL), Christopher Dearlove (BAE), Paul Lambert (Marvell), Jerome Milan (Ecole Polytechnique) and Henning Rogge (FGAN) for their constructive comments on the document.

16. References

16.1. Normative References

[BCP26]	Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, BCP 26, May 2008.
[RFC2119]	

	Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, BCP 14, March 1997.
[RFC5444]	Clausen, T.H., Dearlove, C.M., Dean, J.W. and C. Adjih, "Generalized MANET Packet/Message Format", RFC 5444, February 2009.

16.2. Informative References

[RFC6130]	Clausen, T.H., Dean, J.W. and C.M. Dearlove, "MANET Neighborhood Discovery Protocol (NHDP)", RFC 6130, March 2011.
[OLSRv2]	Clausen, T.H., Dearlove, C.M. and P. Jacquet, "The Optimized Link State Routing Protocol version 2", work in progress draft-ietf-manet-olsrv2-12.txt, July 2011.

Authors' Addresses

Ulrich Herberg Herberg Fujitsu Laboratories of America 1240 E. Arques Ave. M/S 345 Sunnyvale, CA, 94085 USA EMail: ulrich@herberg.name URI: http://www.herberg.name/

Thomas Heide Clausen Clausen LIX, Ecole Polytechnique 91128 Palaiseau Cedex, France Phone: +33 6 6058 9349 EMail: T.Clausen@computer.org URI: http://www.thomasclausen.org/