

Mobile Ad hoc Networking (MANET)  
Internet-Draft  
Intended status: Standards Track  
Expires: January 28, 2010

U. Herberg  
T. Clausen  
LIX, Ecole Polytechnique  
July 27, 2009

MANET Cryptographical Signature TLV Definition  
draft-herberg-manet-packetbb-sec-02

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79. This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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>.

This Internet-Draft will expire on January 28, 2010.

Copyright Notice

Copyright (c) 2009 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal

Provisions Relating to IETF Documents in effect on the date of publication of this document (<http://trustee.ietf.org/license-info>). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

## Abstract

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

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Applicability Statement . . . . .	4
4. Protocol Overview and Functioning . . . . .	5
5. General Signature TLV Structure . . . . .	5
6. General Timestamp TLV Structure . . . . .	7
7. Message TLVs . . . . .	7
7.1. Message SIGNATURE TLV . . . . .	7
7.2. Message TIMESTAMP TLV . . . . .	8
8. Packet TLVs . . . . .	8
8.1. Packet SIGNATURE TLV . . . . .	8
8.1.1. Packet TIMESTAMP TLV . . . . .	8
9. IANA Considerations . . . . .	8
9.1. TLV Registrations . . . . .	8
9.1.1. Expert Review: Evaluation Guidelines . . . . .	9
9.1.2. Message TLV Type Registrations . . . . .	9
9.1.3. Packet TLV Type Registrations . . . . .	9
9.2. New IANA registries . . . . .	10
9.2.1. Expert Review: Evaluation Guidelines . . . . .	10
9.2.2. Hash-Function Registry . . . . .	10
9.2.3. Cryptographic Algorithm Registry . . . . .	11
10. Security Considerations . . . . .	11
11. Acknowledgements . . . . .	12
12. References . . . . .	12
12.1. Normative References . . . . .	12
12.2. Informative References . . . . .	12
Appendix A. Examples . . . . .	13
A.1. Example of a Signed Message . . . . .	13
Authors' Addresses . . . . .	15

## 1. Introduction

This document:

- o specifies two TLVs for carrying cryptographic signatures and timestamps in packets and messages as defined by [RFC5444],
- o requests IANA allocations for these Packet and Message TLVs from the 0-127 Message TLV range and the 0-223 Packet TLV range from [RFC5444],
- o describes how cryptographic signatures are calculated, taking into account the mutable message header fields (<msg-hop-limit> and <msg-hop-count>) for messages where these fields are present,
- o requests creation of two IANA registries for recording code points for hash function and signature calculation, respectively.

This document does not stipulate how to sign, validate, or encrypt messages. A specification of a routing protocol or routing protocol extension, using the security representation of this document, MUST specify appropriate interpretation of the TLVs. This document does specifically not suggest specific cryptographic algorithms or hash functions, but rather establishes IANA registries for such.

## 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]. Additionally, it defines the following terminology:

### o Hash-Function

A hash function is an algorithm that takes a message of any length as input and produces a fixed-length string as output. Hash functions are used in cryptography for authentication and message integrity.

### o Signature:

An electronic signature authenticates the signer (who often is the originator) of a message. In addition, it can be verified whether the message has been changed after it has been signed. In many cases, a signature is calculated by encrypting a hash of a message, whis is the basic assumption of this document.

- o Timestamp

The timestamp indicates the time when the timestamp has been created. If a timestamp is added to a message before signing the message, this information can be useful to determine the "freshness" of the signed message. "Old" messages can indicate replayed messages.

### 3. Applicability Statement

The packet and message format defined in [RFC5444] accords MANET routing protocols using this format the ability to carry additional information in control messages, through inclusion of TLVs. Information so included in a control message MAY be used by the routing protocol, or an extension of the routing protocol, according to its specification.

This document specifies how to include a cryptographic signature for a packet or message by way of TLVs, as specified in [RFC5444]. This document also specifies how to treat "mutable" fields (<msg-hop-count> and <msg-hop-limit>) in the message header when calculating the signature, such that the resulting signature can be correctly verified by any recipient, and how to include this signature. A MANET routing protocol, or an extension of a MANET routing protocol, MAY use such included cryptographic signatures for, for example, rejecting messages where signature verification fails.

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. Protocols such as [NHDP] recognize external reasons (such as failure to verify a signature) as being reasons 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:

- o Security-oblivious MANET routing protocol specification, with a clause allowing an extension to reject a message (prior to processing/forwarding) as "badly formed".
- o MANET routing protocol security extensions, rejecting messages as "badly formed", as appropriate for a given deployment-domain.
- o Code-points and an exchange format for information necessary for specification of such security extensions.

This document addresses the last of these issues, by specifying a common exchange format for cryptographic signatures. This document also makes reservations from within the Message TLV and Packet TLV registries of [RFC5444], to be used (and shared) among MANET routing protocol security extensions. Finally, this document establishes two IANA registries for code-points for hash functions and cryptographic functions for use by protocols adhering to [RFC5444].

With respect to [RFC5444], this document:

- o is intended to be used in the non-normative but intended mode of use of [RFC5444] as described in its Appendix B.
- o is a specific example of the Security Considerations section of [RFC5444] (the authentication part).

#### 4. Protocol Overview and Functioning

This specification does not describe a protocol, nor does it mandate specific router or protocol behavior. It represents a purely syntactical representation of security related information for use with [RFC5444] messages and packets, as well as sets up IANA registrations and registries.

#### 5. General Signature TLV Structure

The following data structure allows the representation of a cryptographic signature, including specification of the appropriate hash function and cryptographic algorithm used for calculating the signature. This <signature> data structure is specified, using the regular expression syntax of [RFC5444], as:

```
<signature> := <hash-function>
               <cryptographic-algorithm>
               <signature-value>
```

where:

<hash-function> is an 8-bit unsigned integer field specifying the hash function according to Table 3.

<cryptographic-algorithm> is an 8-bit unsigned integer field specifying the cryptographic function according to Table 4.

<signature-value> is an unsigned integer field, whose length is <tlv-length>-2, and which contains the cryptographic signature.

The basic version of this TLV assumes that calculating the signature can be decomposed into:

```
signature-value = cryptographic-function(hash-function(message))
```

with cryptographic-function and hash-function being selected from Table 3 and Table 4 respectively (where either of them can be the identity function -- indicated by "none" in the registry). The type extension 0 is assumed to indicate this decomposition. Otherwise, if a signature is not decomposable in that way, the type extension field can be used for indication how signatures are to be calculated.

The algorithm that is used for calculating the hash function is selected from one of those listed in Table 3. Furthermore, <hash-function> corresponds to the number in that table assigned by IANA.

The algorithm that is used for calculating the cryptographic algorithm is selected from one of those listed in Table 4. Furthermore, <cryptographic-algorithm> corresponds to the number in that table assigned by IANA. If the selected hash function is "none" (0), the cryptographic function SHOULD NOT be "none" (0).

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 or cryptographic functions are added in the future, the number space might not be continuous any more. More importantly, the number space of 256 possible combinations is rapidly exhausted. For example, having only 16 different hash functions and 16 different cryptographic functions would lead to exhaustion. 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 the following: Before being able to validate a cryptographic signature, routers have to exchange keys (e.g. public keys). Any additional parameters can be exchanged together with the keys in this bootstrap process. It is therefore not necessary, and would even entail an extra overhead, to transmit the parameters within every message. One inherently included parameter is the length of the signature, which is `tlv-length - 2` and which depends on the choice of the cryptographic function.

## 6. General Timestamp TLV Structure

The following data structure allows the representation of a timestamp. This `<timestamp>` data structure is specified as:

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

where:

`<time-value>` is an unsigned integer field, whose length is `<tlv-length>`, and which contains the timestamp. The value of this variable is to be interpreted by the routing protocol as specified by the type extension of the Timestamp TLV (refer to Table 2).

A timestamp is essentially "freshness information". As such, its setting and interpretation is to be determined by the routing protocol (or the extension to a routing protocol) that uses it, and may e.g. correspond to a UNIX-timestamp, GPS timestamp or a simple sequence number. This is out of the scope of this specification.

## 7. 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.

### 7.1. Message SIGNATURE TLV

A Message SIGNATURE TLV is an example of a SIGNATURE TLV as described in Section 5. When determining the `<signature-value>` for a message, the signature is calculated over the entire message with the following considerations:

- o the fields `<msg-hop-limit>` and `<msg-hop-count>` MUST be both assumed to have the value 0 (zero).

- o all Message SIGNATURE TLVs MUST be removed before calculating the signature, and the message size as well as the Message TLV block size MUST be recalculated accordingly. The TLVs can be restored after having calculated the signature value.

## 7.2. Message TIMESTAMP TLV

A Message TIMESTAMP TLV is an example of a TIMESTAMP TLV as described in Section 6. If a message contains a TIMESTAMP TLV and a SIGNATURE TLV, the TIMESTAMP TLV SHOULD be added first to the message, in order to include it in the calculation of the signature.

## 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 5. When calculating the <signature-value> for a Packet, the signature is calculated over the entire Packet, including the packet header, all Packet TLVs (other than Packet SIGNATURE TLVs) and all included Messages and their message headers.

#### 8.1.1. Packet TIMESTAMP TLV

A Packet TIMESTAMP TLV is an example of a TIMESTAMP TLV as described in Section 6.

## 9. IANA Considerations

### 9.1. TLV Registrations

This specification defines 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 1 and 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 2.

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

## 9.1.1.1. 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].

## 9.1.1.2. Message TLV Type Registrations

The Message TLVs as specified in Table 1 must be allocated from the "Message TLV Types" namespace of [RFC5444].

Name	Type	Type Extension	Description
SIGNATURE	TBD1	0	Signature of a message
		1-223	Expert Review
		224-255	Experimental Use
TIMESTAMP	TBD2	0	Unsigned timestamp of arbitrary length, given by the tlv-length field. The timestamp is assumed to increase strictly monotonously by steps of 1. The MANET routing protocol has to define how to interpret this timestamp
		1	Unsigned 32-bit timestamp as specified in [POSIX]
		2	NTP timestamp format as defined in [RFC4330]
		3	Signed timestamp of arbitrary length with no constraints such as monotonicity. In particular, it may represent any random value
		4-223	Expert Review
		224-255	Experimental Use

Table 1: Message TLV types

## 9.1.1.3. Packet TLV Type Registrations

The Packet TLVs as specified in Table 2 must be allocated from the "Packet TLV Types" namespace of [RFC5444].

Name	Type	Type Extension	Description
SIGNATURE	TBD3	0	Signature of a packet
		1-223	Expert Review
		224-255	Experimental Use
TIMESTAMP	TBD4	0	Unsigned timestamp of arbitrary length, given by the tlv-length field. The timestamp is assumed to increase strictly monotonously by steps of 1. The MANET routing protocol has to define how to interpret this timestamp
		1	Unsigned 32-bit timestamp as specified in [POSIX]
		2	NTP timestamp format as defined in [RFC4330]
		3	Signed timestamp of arbitrary length with no constraints such as monotonicity. In particular, it may represent any random value
		4-223	Expert Review
		224-255	Experimental Use

Table 2: Packet TLV types

## 9.2. New IANA registries

This document specifies some values where IANA registries are required.

### 9.2.1. Expert Review: Evaluation Guidelines

For the registries for the following tables where an Expert Review is required, the designated expert SHOULD take the same general recommendations into consideration as are specified by [RFC5444].

### 9.2.2. Hash-Function Registry

IANA is requested to create a new registry for the hash functions that can be used when creating a signature. The initial assignments and allocation policies are specified in Table 3.

Hash function value	Algorithm	Description
0	none	The "identity function": the hash value of a message is the message itself
1	MD5	The hash function as specified in [RFC1321]
2	SHA1	The hash function as specified in [RFC3174]
3	SHA256	The hash function as specified in [SHA256]
4-223		Expert Review
224-255		Experimental Use

Table 3: Hash-Function registry

### 9.2.3. Cryptographic Algorithm Registry

IANA is requested to create a new registry for the cryptographic function. Initial assignments and allocation policies are specified in Table 4.

Cryptographic algorithm value	Algorithm	Description
0	none	The "identity function": the value of an encrypted hash is the hash itself
1	RSA	RSA as specified in [RFC2437]
2	DSA	DSA as specified in [DSA]
3	HMAC	HMAC as specified in [RFC2104]
4	3DES	3DES as specified in [3DES]
5	AES	AES as specified in [AES]
6-223		Expert Review
224-255		Experimental Use

Table 4: Cryptographic algorithm registry

## 10. Security Considerations

This document does not specify a protocol itself. However, 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 protocol or extension that uses the component specified in this document. As such, it has the same security considerations as [RFC5444].

In addition, a protocol that includes this component 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 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).

## 11. Acknowledgements

The authors would like to thank Jerome Milan (Ecole Polytechnique) for his advice as cryptographer. In addition, many thanks to Alan Cullen (BAE), Justin Dean (NRL), Christopher Dearlove (BAE), and Henning Rogge (FGAN) for their constructive comments on the document.

## 12. References

### 12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, BCP 14, March 1997.
- [RFC5444] Clausen, T., Dearlove, C., Dean, J., and C. Adjih, "Generalized MANET Packet/Message Format", RFC 5444, February 2009.

### 12.2. Informative References

- [3DES] American National Standards Institute, "Triple Data Encryption Algorithm Modes of Operation", ANSI X9.52-1998, 1998.
- [AES] National Institute of Standards & Technology, "Advanced Encryption Standard (AES)", FIPS 197, November 2001.
- [DSA] National Institute of Standards & Technology, "Digital Signature Standard", NIST, FIPS PUB 186, May 1994.

- [NHDP] Clausen, T., Dean, J., and C. Dearlove, "MANET Neighborhood Discovery Protocol (NHDP)", work in progress draft-ietf-manet-nhdp-10.txt, July 2009.
- [POSIX] IEEE Computer Society, "1003.1-2008 Standard for Information Technology - Portable Operating System Interface (POSIX)", Base Specifications Issue 7, December 2008.
- [RFC1321] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.
- [RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.
- [RFC2437] Kaliski, B. and J. Staddon, "PKCS #1: RSA Cryptography Specifications Version 2.0", RFC 2437, October 1998.
- [RFC3174] Eastlake, D. and P. Jones, "US Secure Hash Algorithm 1 (SHA1)", RFC 3174, September 2001.
- [RFC4330] Mills, D., "Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI", RFC 4330, January 2006.
- [SHA256] National Institute of Standards and Technology, "Secure Hash Algorithm", NIST FIPS 180-2, August 2002.

## Appendix A. Examples

### A.1. Example of a Signed Message

The sample message depicted in Figure 1 is taken from the appendix of [RFC5444]. However, a SIGNATURE Message TLV has been added. It is assumed that the SIGNATURE TLV type is lesser than the TLV type of the second message TLV (i.e. it comes first in the order of Message TLVs). The TLV value represents a 16 octet long signature of the whole message.

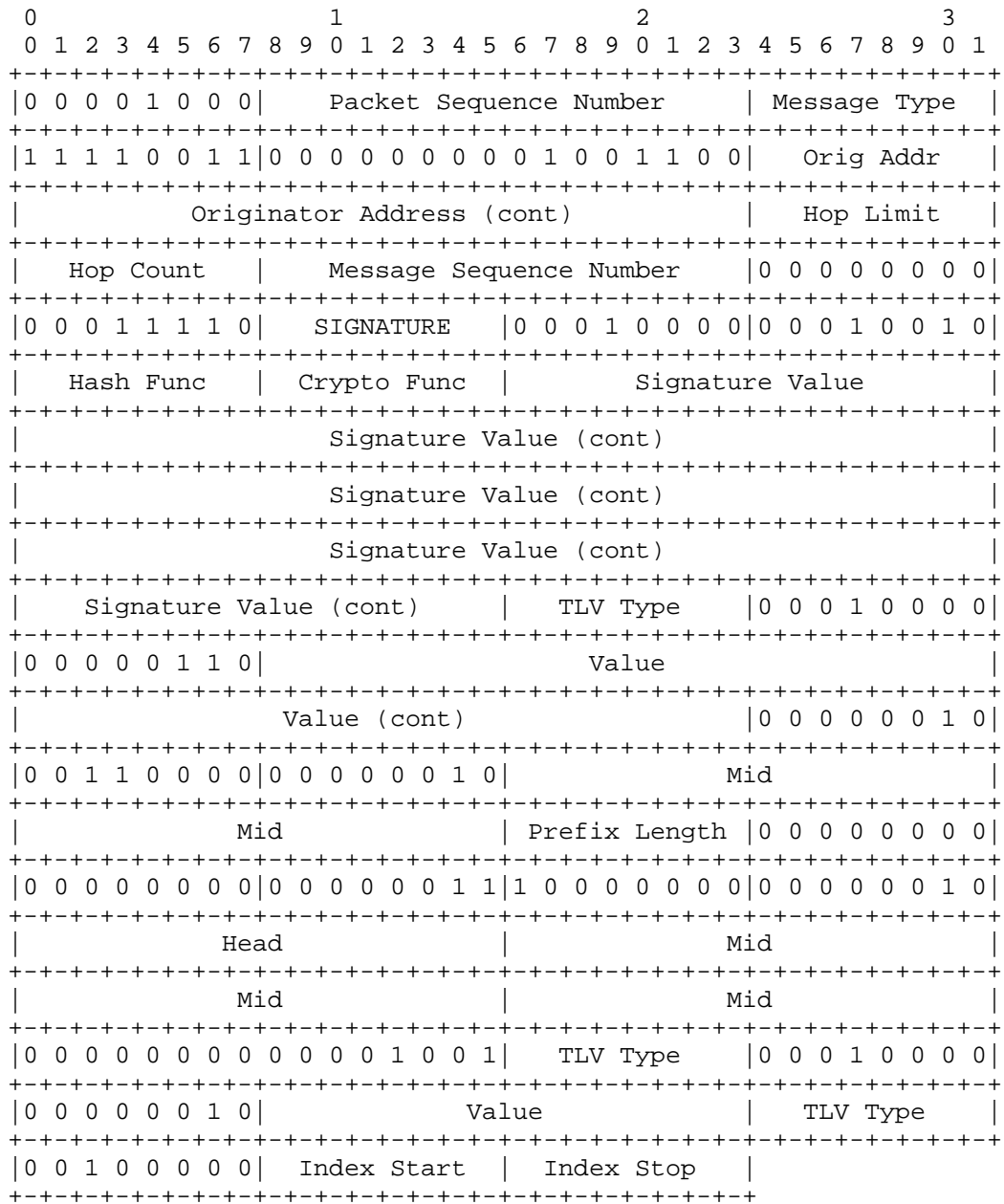


Figure 1: Example message with signature

Authors' Addresses

Ulrich Herberg  
LIX, Ecole Polytechnique  
91128 Palaiseau Cedex,  
France

Phone: +33-1-6933-4126  
Email: [ulrich@herberg.name](mailto:ulrich@herberg.name)  
URI: <http://www.herberg.name/>

Thomas Heide Clausen  
LIX, Ecole Polytechnique  
91128 Palaiseau Cedex,  
France

Phone: +33 6 6058 9349  
Email: [T.Clausen@computer.org](mailto:T.Clausen@computer.org)  
URI: <http://www.thomasclausen.org/>

