

Mobile Ad hoc Networking (MANET)  
Internet-Draft  
Intended status: Standards Track  
Expires: December 22, 2010

U. Herberg  
T. Clausen  
LIX, Ecole Polytechnique  
June 20, 2010

MANET Cryptographical Signature TLV Definition  
draft-ietf-manet-packetbb-sec-00

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 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

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on December 22, 2010.

Copyright Notice

Copyright (c) 2010 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 (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as

described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Applicability Statement . . . . .	4
4. Protocol Overview and Functioning . . . . .	5
5. General Signature TLV Structure . . . . .	6
5.1. Rationale . . . . .	6
6. General Timestamp TLV Structure . . . . .	7
7. Packet TLVs . . . . .	7
7.1. Packet SIGNATURE TLV . . . . .	7
7.2. Packet TIMESTAMP TLV . . . . .	8
8. Message TLVs . . . . .	8
8.1. Message SIGNATURE TLV . . . . .	8
8.2. Message TIMESTAMP TLV . . . . .	8
9. Address Block TLVs . . . . .	8
9.1. Address Block SIGNATURE TLV . . . . .	9
9.2. Address Block TIMESTAMP TLV . . . . .	9
10. IANA Considerations . . . . .	9
10.1. TLV Registrations . . . . .	9
10.1.1. Expert Review: Evaluation Guidelines . . . . .	9
10.1.2. Packet TLV Type Registrations . . . . .	9
10.1.3. Message TLV Type Registrations . . . . .	10
10.1.4. Address Block TLV Type Registrations . . . . .	11
10.2. New IANA Registries . . . . .	12
10.2.1. Expert Review: Evaluation Guidelines . . . . .	12
10.2.2. Hash Function . . . . .	12
10.2.3. Cryptographic Algorithm . . . . .	13
11. Security Considerations . . . . .	13
12. Acknowledgements . . . . .	14
13. References . . . . .	14
13.1. Normative References . . . . .	14
13.2. Informative References . . . . .	14
Appendix A. Examples . . . . .	15
A.1. Example of a Signed Message . . . . .	15
Authors' Addresses . . . . .	17

## 1. Introduction

This document:

- o specifies two TLVs for carrying cryptographic signatures and timestamps in packets, messages and address blocks as defined by [RFC5444],
- o requests 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],
- o describes how cryptographic signatures are calculated, taking (for Message TLVs) into account the mutable message header fields (<msg-hop-limit> and <msg-hop-count>) where these fields are present in messages,
- 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 or validate 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 Object

An object, here, is any sequence of bytes that is used to calculate the signature over (e.g. a packet, a message, an address as defined in [RFC5444], a timestamp, or a combination of these).

- o Signature

A digital signature can be used to (i) authenticate the originator and (ii) to assure that the object, which has been signed, has not been altered in transit. In many cases, a signature is calculated by encrypting a hash of the object, which is the basic assumption of this specification.

- o Timestamp

The timestamp indicates the time when the timestamp has been created. If a timestamp is added to an object before signing the object, this information can be useful to determine the "freshness" of the signed object. "Old" objects can indicate replayed objects. The minimal requirement for a timestamp is to provide a logical representation of time (e.g. Lamport time). Using timestamps may require - at least roughly - synchronized clocks among the routers in the network.

### 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 by an extension of the routing protocol, according to its specification.

This document specifies how to include a cryptographic signature for a packet, message or address block by way of such TLVs. This document also specifies how to treat "mutable" fields (<msg-hop-count> and <msg-hop-limit>) 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. 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] and [OLSRv2] 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 specifications, 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 specific security requirement.
- 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 Packet TLV, Message TLV and Address Block 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 algorithms 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 establishes IANA registrations and registries.

## 5. General Signature TLV Structure

The following data structure allows 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.

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

<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))
```

The hash function and the cryptographic algorithm correspond to the IANA registry in the two registries set up by this specification, see Section 10.

### 5.1. Rationale

The rationale for separating the hash function and the cryptographic algorithm 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 algorithms are added in the future, the number space might not remain continuous. More importantly, the number space of 256 possible combinations would be rapidly exhausted: 16 different hash functions and 16 different cryptographic algorithms 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 algorithm.

## 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 1).

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. 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.

### 7.1. 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.

## 7.2. Packet TIMESTAMP TLV

A Packet TIMESTAMP TLV is an example of a Timestamp TLV as described in Section 6. If a packet contains a TIMESTAMP TLV and a SIGNATURE TLV, the TIMESTAMP TLV SHOULD be added to the packet before the SIGNATURE TLV, in order that it be included in the calculation of the signature.

## 8. 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.

### 8.1. 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.

### 8.2. 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 to the message before the SIGNATURE TLV, in order that it be included in the calculation of the signature.

## 9. 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.

### 9.1. Address Block SIGNATURE TLV

An Address Block SIGNATURE TLV is an example of a Signature TLV as described in Section 5. The signature can be calculated over any object, including, for example, the address to which this TLV is associated to.

### 9.2. Address Block TIMESTAMP TLV

An Address Block TIMESTAMP TLV is an example of a Timestamp TLV as described in Section 6. 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.

## 10. IANA Considerations

### 10.1. TLV Registrations

This specification defines:

- o 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,
- o 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,
- o and 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.

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

#### 10.1.1. 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].

#### 10.1.2. Packet TLV Type Registrations

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

Name	Type	Type Extension	Description
SIGNATURE	TBD3	0 1-223 224-255	Signature of a packet Expert Review Experimental Use
TIMESTAMP	TBD4	0  1 2 3  4-223 224-255	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 Unsigned 32-bit timestamp as specified in [POSIX] NTP timestamp format as defined in [RFC4330] Signed timestamp of arbitrary length with no constraints such as monotonicity. In particular, it may represent any random value Expert Review Experimental Use

Table 1: Packet TLV types

## 10.1.3. Message TLV Type Registrations

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

Name	Type	Type Extension	Description
SIGNATURE	TBD1	0 1-223 224-255	Signature of a message Expert Review 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 2: Message TLV types

## 10.1.4. Address Block TLV Type Registrations

The Address Block TLVs as specified in Table 3 must be allocated from the "Address Block TLV Types" namespace of [RFC5444].

Name	Type	Type Extension	Description
SIGNATURE	TBD1	0	Signature of an object (e.g. an address)
		1-223 224-255	Expert Review 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
		4-223 224-255	Expert Review Experimental Use

Table 3: Address Block TLV types

## 10.2. New IANA Registries

This document introduces three namespaces that have been registered: Packet TLV Types, Message TLV Types, and Address Block TLV Types. This section specifies IANA registries for these namespaces and provides guidance to the Internet Assigned Numbers Authority regarding registrations in these namespaces.

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".

### 10.2.1. 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].

### 10.2.2. Hash Function

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 4.

Hash function value	Algorithm	Description
0	none	The "identity function": the hash value of an object is the object 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 4: Hash-Function registry

## 10.2.3. Cryptographic Algorithm

IANA is requested to create a new registry for the cryptographic algorithm. Initial assignments and allocation policies are specified in Table 5.

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 5: Cryptographic algorithm registry

## 11. 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).

## 12. 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.

## 13. References

### 13.1. 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., Dearlove, C., Dean, J., and C. Adjih, "Generalized MANET Packet/Message Format", RFC 5444, February 2009.

### 13.2. 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-12.txt, March 2010.
- [OLSRv2] Clausen, T., Dearlove, C., and P. Jacquet, "The Optimized Link State Routing Protocol version 2", work in progress draft-ietf-manet-olsrv2-11.txt, April 2010.
- [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.

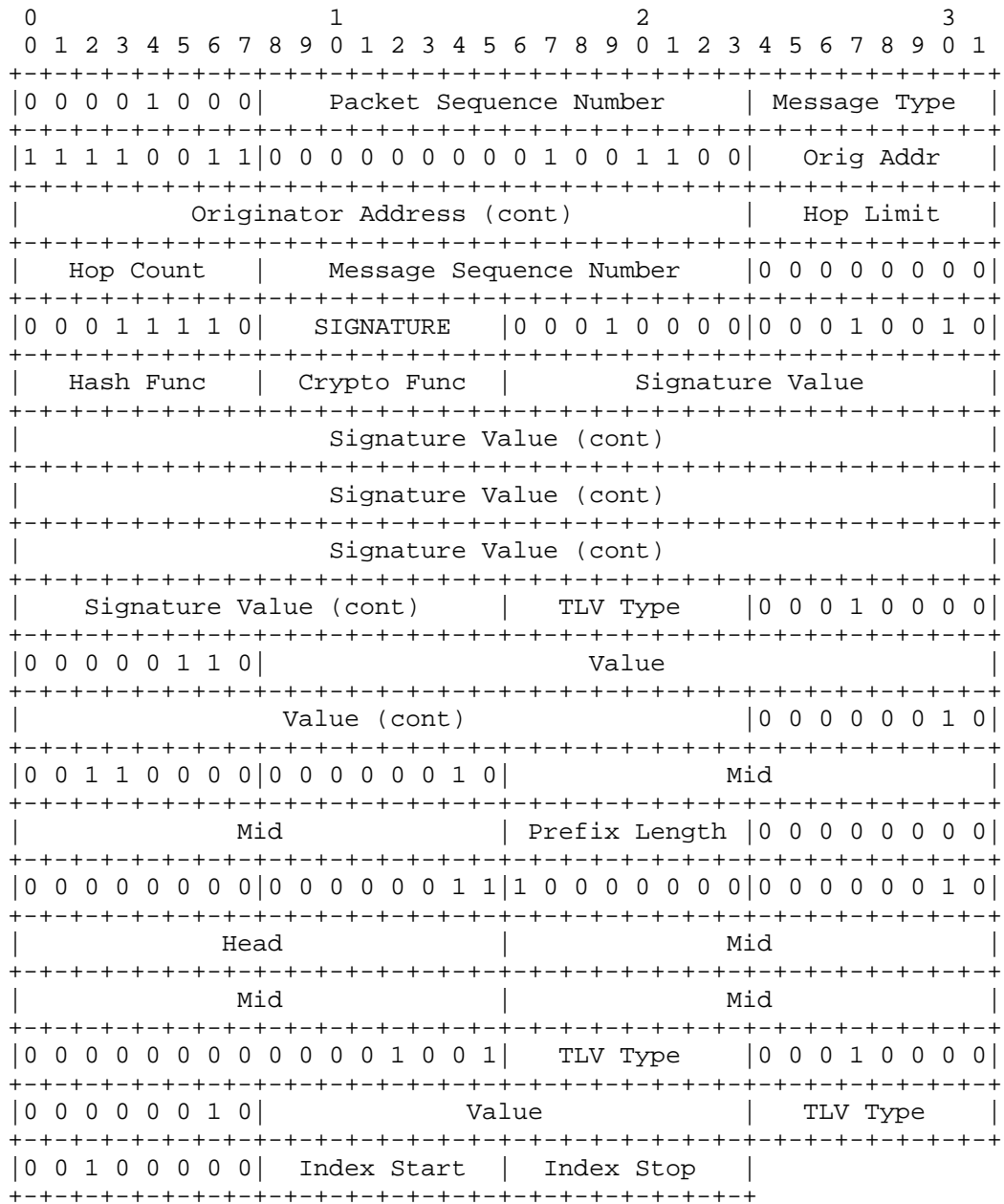


Figure 1: Example message with signature

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/>

