Network Working Group   L. Zheng
Internet-Draft   M. Chen
Intended status: Standards Track   Huawei Technologies
Expires: September 30, 2014   M. Bhatia
                                                         Alcatel-Lucent
                                                         March 29, 2014

                 LDP Hello Cryptographic Authentication
               draft-ietf-mpls-ldp-hello-crypto-auth-04.txt

Abstract

   This document introduces a new optional Cryptographic Authentication
   TLV that LDP can use to secure its Hello messages. It secures the
   Hello messages against spoofing attacks and some well known attacks
   against the IP header. This document describes a mechanism to secure
   the LDP Hello messages using National Institute of Standards and
   Technology (NIST) Secure Hash Standard family of algorithms.

Requirements Language

   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 RFC 2119 [RFC2119].

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 September 30, 2014.

Copyright Notice

Table of Contents

1. Introduction

   The Label Distribution Protocol (LDP) [RFC5036] sets up LDP sessions
   that run between LDP peers. The peers could either be directly
   connected at the link level or could be multiple hops away. An LDP
   Label Switching Router (LSR) could either be configured with the
   identity of its peers or could discover them using LDP Hello
   messages. These messages are sent encapsulated in UDP addressed to
   "all routers on this subnet" or to a specific IP address. Periodic
   Hello messages are also used to maintain the relationship between LDP
   peers necessary to keep the LDP session active.

   Since the Hello messages are sent using UDP and not TCP, these
   messages cannot use the security mechanisms defined for TCP
   [RFC5926]. While some configuration guidance is given in [RFC5036]
   to help protect against false discovery messages, it does not provide
   an explicit security mechanism to protect the Hello messages

   Spoofing a Hello packet for an existing adjacency can cause the valid
   adjacency to time out and in turn can result in termination of the
   associated session. This can occur when the spoofed Hello specifies
   a smaller Hold Time, causing the receiver to expect Hellos within
   this smaller interval, while the true neighbor continues sending
   Hellos at the previously agreed lower frequency. Spoofing a Hello
   packet can also cause the LDP session to be terminated directly,
   which can occur when the spoofed Hello specifies a different
   Transport Address, other than the previously agreed one between
   neighbors. Spoofed Hello messages have been observed and reported as
   a real problem in production networks [RFC6952].

   [RFC5036] states that the threat of spoofed Hellos can be reduced by
   accepting Hellos only on interfaces to which LSRs that can be trusted
   are directly connected, and ignoring Hellos not addressed to the "all
   routers on this subnet" multicast group. Spoofing attacks via
   Targeted Hellos are a potentially more serious threat. An LSR can
   reduce the threat of spoofed Targeted Hellos by filtering them and
   accepting only those originating at sources permitted by an access
   list. However, filtering using access lists requires LSR resource,
   and does not prevent IP-address spoofing.

   This document introduces a new Cryptographic Authentication TLV which
   is used in LDP Hello messages as an optional parameter. It enhances
   the authentication mechanism for LDP by securing the Hello message
   against spoofing attack. It also introduces a cryptographic sequence
   number carried in the Hello messages that can be used to protect
   against replay attacks. The LSRs could be configured to only accept
   Hello messages from specific peers when authentication is in use.

Using this Cryptographic Authentication TLV, one or more secret keys
(with corresponding key IDs) are configured in each system. For each
LDP Hello message, the key is used to generate and verify a HMAC Hash
that is stored in the LDP Hello message. For cryptographic hash
function, this document proposes to use SHA-1, SHA-256, SHA-384, and
SHA-512 defined in US NIST Secure Hash Standard (SHS) [FIPS-180-3].
The HMAC authentication mode defined in NIST FIPS 198 is used
[FIPS-198]. Of the above, implementations MUST include support for
at least HMAC-SHA-256 and SHOULD include support for HMAC-SHA-1 and
MAY include support for either of HMAC-SHA-384 or HMAC-SHA-512.

2. Cryptographic Authentication TLV

2.1. Optional Parameter for Hello Message

[RFC5036] defines the encoding for the Hello message. Each Hello
message contains zero or more Optional Parameters, each encoded as a
TLV. Three Optional Parameters are defined by [RFC5036]. This
document defines a new Optional Parameter: the Cryptographic
Authentication parameter.

```
Optional Parameter              Type
------------------------------  --------
IPv4 Transport Address          0x0401 (RFC5036)
Configuration Sequence Number   0x0402 (RFC5036)
IPv6 Transport Address          0x0403 (RFC5036)
Cryptographic Authentication    0x0404 (this document, TBD by IANA)
```

The Cryptographic Authentication TLV Encoding is described in section
2.2.

2.2. LDP Security Association

An LDP Security Association (SA) contains a set of parameters shared
between any two legitimate LDP speakers.

Parameters associated with an LDP SA are as follows:

o   Security Association Identifier (SA ID)

    This is a 32-bit unsigned integer used to uniquely identify an LDP
    SA, as manually configured by the network operator.

    The receiver determines the active SA by looking at the SA ID
    field in the incoming Hello message.

    The sender, based on the active configuration, selects an SA to
    use and puts the correct Key ID value associated with the SA in
    the LDP Hello message. If multiple valid and active LDP SAs exist
    for a given interface, the sender may use any of those SAs to
    protect the packet.

    Using SA IDs makes changing keys while maintaining protocol
    operation convenient. Each SA ID specifies two independent parts,
    the authentication algorithm and the authentication key, as
    explained below.

    Normally, an implementation would allow the network operator to

configure a set of keys in a key chain, with each key in the chain
having fixed lifetime. The actual operation of these mechanisms
is outside the scope of this document.

Note that each SA ID can indicate a key with a different
authentication algorithm. This allows the introduction of new
authentication mechanisms without disrupting existing LDP
sessions.

o Authentication Algorithm

This signifies the authentication algorithm to be used with the
LDP SA. This information is never sent in clear text over the
wire. Because this information is not sent on the wire, the
implementer chooses an implementation specific representation for
this information.

Currently, the following algorithms are supported:

HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512.

o Authentication Key

This value denotes the cryptographic authentication key associated
with the LDP SA. The length of this key is variable and depends
upon the authentication algorithm specified by the LDP SA.

o KeyStartAccept

The time that this OSPFv3 router will accept packets that have
been created with this OSPFv3 Security Association.

o KeyStartGenerate

The time that this LDP router will begin using this LDP Security
Association for LDP Hello message generation.

o KeyStopGenerate

The time that this LDP router will stop using this LDP Security
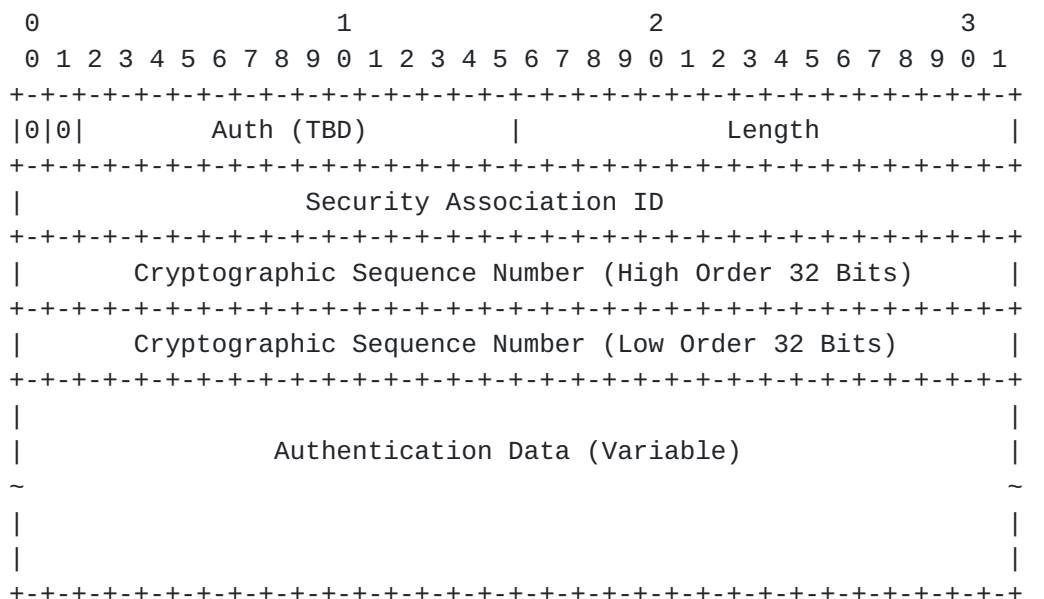Association for LDP Hello message generation.

o KeyStopAccept

The time that this LDP router will stop accepting packets
generated with this LDP Security Association.

In order to achieve smooth key transition, KeyStartAccept SHOULD be

less than KeyStartGenerate and KeyStopGenerate SHOULD be less than
KeyStopAccept. If KeyStartGenerate or KeyStartAccept are left
unspecified, the time will default to 0 and the key will be used
immediately. If KeyStopGenerate or KeyStopAccept are left
unspecified, the time will default to infinity and the key's lifetime
will be infinite. When a new key replaces an old, the
KeyStartGenerate time for the new key MUST be less than or equal to
the KeyStopGenerate time of the old key.

Key storage SHOULD persist across a system restart, warm or cold, to
avoid operational issues. In the event that the last key associated
with an interface expires, it is unacceptable to revert to an
unauthenticated condition, and not advisable to disrupt routing.
Therefore, the router SHOULD send a "last Authentication Key
expiration" notification to the network manager and treat the key as
having an infinite lifetime until the lifetime is extended, the key
is deleted by network management, or a new key is configured

2.3. Cryptographic Authentication TLV Encoding

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0|0|        Auth (TBD)        |              Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Security Association ID                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Cryptographic Sequence Number (High Order 32 Bits)      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Cryptographic Sequence Number (Low Order 32 Bits)       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                              |
|                 Authentication Data (Variable)               |
~                                                              ~
|                                                              |
|                                                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Type: TBD, Cryptographic Authentication

- Length: Specifying the length in octets of the value field.

- Security Association ID: 32 bit field that maps to the
authentication algorithm and the secret key used to create the
message digest carried in LDP payload.

Though the SA ID implies the algorithm, the HMAC output size should
not be used by implementers as an implicit hint, because additional

   algorithms may be defined in the future that have the same output
   size.

   - Cryptographic Sequence Number: 64-bit strictly increasing sequence
   number that is used to guard against replay attacks. The 64-bit
   sequence number MUST be incremented for every LDP Hello packet sent
   by the LDP router. Upon reception, the sequence number MUST be
   greater than the sequence number in the last LDP Hello packet
   accepted from the sending LDP neighbor. Otherwise, the LDP packet is
   considered a replayed packet and dropped.

   LDP routers implementing this specification MUST use existing
   mechanisms to preserve the sequence number's strictly increasing
   property for the deployed life of the LDP router (including cold
   restarts). One mechanism for accomplishing this could be to use the
   high-order 32 bits of the sequence number as a boot count that is
   incremented anytime the LDP router loses its sequence number state.
   Techniques such as sequence number space partitioning described above
   or non-volatile storage preservation can be used but are beyond the
   scope of this specification. Sequence number wrap is described in
   Section 2.4.

   - Authentication Data:

   This field carries the digest computed by the Cryptographic
   Authentication algorithm in use. The length of the Authentication
   Data varies based on the cryptographic algorithm in use, which is
   shown as below:

   Auth type   Length
   ---------------   ----------
   HMAC-SHA1   20 bytes
   HMAC-SHA-256   32 bytes
   HMAC-SHA-384   48 bytes
   HMAC-SHA-512   64 bytes

2.4. Sequence Number Wrap

   When incrementing the sequence number for each transmitted LDP
   packet, the sequence number should be treated as an unsigned 64-bit
   value. If the lower order 32-bit value wraps, the higher order 32-
   bit value should be incremented and saved in non-volatile storage.
   If the LDP router is deployed long enough that the 64-bit sequence
   number wraps, all keys, independent of key distribution mechanism
   MUST be reset. This is done to avoid the possibility of replay
   attacks. Once the keys have been changed, the higher order sequence
   number can be reset to 0 and saved to non-volatile storage.

[3](#). Cryptographic Authentication Procedure

   As noted earlier, the Security Association ID maps to the
   authentication algorithm and the secret key used to generate and
   verify the message digest. This specification discusses the
   computation of LDP Cryptographic Authentication data when any of the
   NIST SHS family of algorithms is used in the Hashed Message
   Authentication Code (HMAC) mode.

   The currently valid algorithms (including mode) for LDP Cryptographic
   Authentication include:

   HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 and HMAC-SHA-512

   Of the above, implementations of this specification MUST include
   support for at least HMAC-SHA-256 and SHOULD include support for
   HMAC-SHA-1 and MAY also include support for HMAC-SHA-384 and HMAC-
   SHA-512.

   Implementations of this standard MUST use HMAC-SHA-256 as the default
   authentication algorithm.

4. **Cross Protocol Attack Mitigation**

   In order to prevent cross protocol replay attacks for protocols
   sharing common keys, the two octet LDP Cryptographic Protocol ID is
   appended to the authentication key prior to use. Other protocols
   using cryptographic authentication as specified herein MUST similarly
   append their respective Cryptographic Protocol IDs to their keys in
   this step. Refer to IANA Considerations (Section 8).

5. **Cryptographic Aspects**

   In the algorithm description below, the following nomenclature, which
   is consistent with [FIPS-198], is used:

   H is the specific hashing algorithm (e.g. SHA-256).

   K is the Authentication Key from the LDP security association.

   Ks is a Protocol Specific Authentication Key obtained by appending
   Authentication Key (K) with the two-octet LDP Cryptographic Protocol
   ID appended.

   Ko is the cryptographic key used with the hash algorithm.

   B is the block size of H, measured in octets rather than bits.

   Note that B is the internal block size, not the hash size.

      For SHA-1 and SHA-256: B == 64

      For SHA-384 and SHA-512: B == 128

   L is the length of the hash, measured in octets rather than bits.

   XOR is the exclusive-or operation.

   Opad is the hexadecimal value 0x5c repeated B times.

   Ipad is the hexadecimal value 0x36 repeated B times.

   Apad is a value which is the same length as the hash output or
   message digest. In case of IPv4, the first 4 octets contain the IPv4
   source address followed by the hexadecimal value 0x878FE1F3 repeated
   (L-4)/4 times. In case of IPv6, the first 16 octets contain the IPv6
   source address followed by the hexadecimal value 0x878FE1F3 repeated
   (L-16)/4 times. This implies that hash output is always a length of
   at least 16 octets.

5.1. **Preparing the Cryptographic Key**

   The LDP Cryptographic Protocol ID is appended to the Authentication
   Key (K) yielding a Protocol Specific Authentication Key (Ks). In
   this application, Ko is always L octets long. While [RFC2104]
   supports a key that is up to B octets long, this application uses L
   as the Ks length consistent with [RFC4822], [RFC5310], [RFC5709] and
   [RFC7166]. According to [FIPS-180-3], Section 3, keys greater than L

octets do not significantly increase the function strength. Ks is
computed as follows:

If the Protocol Specific Authentication Key (Ks) is L octets long,
then Ko is equal to Ks. If the Protocol Specific Authentication Key
(Ks) is more than L octets long, then Ko is set to H(Ks). If the
Protocol Specific Authentication Key (Ks) is less than L octets long,
then Ko is set to the Protocol Specific Authentication Key (Ks) with
zeros appended to the end of the Protocol Specific Authentication Key
(Ks) such that Ko is L octets long.

5.2. Computing the Hash

First, the Authentication Data field in the Cryptographic
Authentication TLV is filled with the value Apad. Then, to compute
HMAC over the Hello message it performs:

H(Ko XOR Opad || H(Ko XOR Ipad || (Hello Message)))

Hello Message refers to the LDP Hello message excluding the IP
header.

When XORing Ko and Ipad, or XORing Ko and Opad, Ko must be padded
with zeros to the length of Ipad and the Opad.

5.3. Result

The resultant Hash becomes the Authentication Data that is sent in
the Authentication Data field of the Cryptographic Authentication
TLV. The length of the Authentication Data field is always identical
to the message digest size of the specific hash function H that is
being used.

This also means that the use of hash functions with larger output
sizes will also increase the size of the LDP message as transmitted
on the wire.

6. **Processing Hello Message Using Cryptographic Authentication**

6.1. **Transmission Using Cryptographic Authentication**

   Prior to transmitting the Hello message, the Length in the
   Cryptographic Authentication TLV header is set as per the
   authentication algorithm that is being used. It is set to 24 for
   HMAC-SHA-1, 36 for HMAC-SHA-256, 52 for HMAC-SHA-384 and 68 for HMAC-
   SHA-512.

   The Security Association ID field is set to the ID of the current
   authentication key. The HMAC Hash is computed as explained in
   Section 3. The resulting Hash is stored in the Authentication Data
   field prior to transmission. The authentication key MUST NOT be
   carried in the packet.

6.2. **Receipt Using Cryptographic Authentication**

   The receiving LSR applies acceptability criteria for received Hellos
   using cryptographic authentication. If the Cryptographic
   Authentication TLV is unknown to the receiving LSR, the received
   packet MUST be discarded according to Section 3.5.1.2.2 of [RFC5036].

   The receiving LSR locates the LDP SA using the Security Association
   ID field carried in the message. If the SA is not found, or if the
   SA is not valid for reception (i.e., current time < KeyStartAccept or
   current time >= KeyStopAccept), LDP Hello message MUST be discarded.

   If the cryptographic sequence number in the LDP packet is less than
   or equal to the last sequence number received from the same neighbor,
   the LDP message MUST be discarded, and an error event SHOULD be
   logged.

   Before the receiving LSR performs any processing, it needs to save
   the values of the Authentication Data field. The receiving LSR then
   replaces the contents of the Authentication Data field with Apad,
   computes the Hash, using the authentication key specified by the
   received Security Association ID field, as explained in Section 3.
   If the locally computed Hash is equal to the received value of the
   Authentication Data field, the received packet is accepted for other
   normal checks and processing as described in [RFC5036]. Otherwise,
   if the locally computed Hash is not equal to the received value of
   the Authentication Data field, the received packet MUST be discarded,
   and an error event SHOULD be logged.

   After the LDP Hello message has been successfully authenticated,
   implementations MUST store the 64-bit cryptographic sequence number
   for the Hello message received from the neighbor. The saved

cryptographic sequence numbers will be used for replay checking for
subsequent packets received from the neighbor.

7. **Security Considerations**

   Section 1 of this document describes the security issues arising from
   the use of unauthenticated LDP Hello messages. In order to address
   those issues, it is RECOMMENDED that all deployments use the
   Cryptographic Authentication TLV to authenticate the Hello messages.

   The quality of the security provided by the Cryptographic
   Authentication TLV depends completely on the strength of the
   cryptographic algorithm in use, the strength of the key being used,
   and the correct implementation of the security mechanism in
   communicating LDP implementations. Also, the level of security
   provided by the Cryptographic Authentication TLV varies based on the
   authentication type used.

   It should be noted that the authentication method described in this
   document is not being used to authenticate the specific originator of
   a packet but is rather being used to confirm that the packet has
   indeed been issued by a router that has access to the Authentication
   Key.

   Deployments SHOULD use sufficiently long and random values for the
   Authentication Key so that guessing and other cryptographic attacks
   on the key are not feasible in their environments. Furthermore, it
   is RECOMMENDED that Authentication Keys incorporate at least 128
   pseudo-random bits to minimize the risk of such attacks. In support
   of these recommendations, management systems SHOULD support
   hexadecimal input of Authentication Keys.

   The mechanism described herein is not perfect and does not need to be
   perfect. Instead, this mechanism represents a significant increase
   in the effort required for an adversary to successfully attack the
   LDP Hello protocol while not causing undue implementation,
   deployment, or operational complexity.

8. IANA Considerations

   The IANA is requested to as assign a new TLV from the "Multiprotocol
   Label Switching Architecture (MPLS) Label Switched Paths (LSPs)
   Parameters - TLVs" registry, "TLVs and sub-TLVs" sub- registry.

   Value   Meaning   Reference
   -----   ------------------------------   ---------
   TBD   Cryptographic Authentication TLV   this document (sect 3.2)

   The IANA is also requested to as assign value from the
   "Authentication Cryptographic Protocol ID", registry under the
   "Keying and Authentication for Routing Protocols (KARP) Parameters"
   category.

   Value   Meaning   Reference
   -----   ------------------------------   ---------
   TBD   LDP Cryptographic Protocol ID   this document (sect 4)

9. Acknowledgements

The authors would like to thank Liu Xuehu for his work on background and motivation for LDP Hello authentication. The authors also would like to thank Adrian Farrel, Eric Rosen, Sam Hartman, Eric Gray, Kamran Raza and Acee Lindem for their valuable comments.

We would also like to thank the authors of RFC 5709 and RFC 7166 from where we have taken most of the cryptographic computation procedures from.

10. References

10.1. Normative References

   [FIPS-180-3]
            "Secure Hash Standard (SHS), FIPS PUB 180-3",
            October 2008.

   [FIPS-198]
            "The Keyed-Hash Message Authentication Code (HMAC), FIPS
            PUB 198", March 2002.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
            Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC4822] Atkinson, R. and M. Fanto, "RIPv2 Cryptographic
            Authentication", RFC 4822, February 2007.

   [RFC5036] Andersson, L., Minei, I., and B. Thomas, "LDP
            Specification", RFC 5036, October 2007.

   [RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R.,
            and M. Fanto, "IS-IS Generic Cryptographic
            Authentication", RFC 5310, February 2009.

   [RFC5709] Bhatia, M., Manral, V., Fanto, M., White, R., Barnes, M.,
            Li, T., and R. Atkinson, "OSPFv2 HMAC-SHA Cryptographic
            Authentication", RFC 5709, October 2009.

   [RFC7166] Bhatia, M., Manral, V., and A. Lindem, "Supporting
            Authentication Trailer for OSPFv3", RFC 7166, March 2014.

10.2. Informative References

   [RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-
            Hashing for Message Authentication", RFC 2104,
            February 1997.

   [RFC5926] Lebovitz, G. and E. Rescorla, "Cryptographic Algorithms
            for the TCP Authentication Option (TCP-AO)", RFC 5926,
            June 2010.

   [RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of
            BGP, LDP, PCEP, and MSDP Issues According to the Keying
            and Authentication for Routing Protocols (KARP) Design
            Guide", RFC 6952, May 2013.

Authors' Addresses

    Lianshu Zheng
    Huawei Technologies
    China

    Email: vero.zheng@huawei.com

    Mach(Guoyi) Chen
    Huawei Technologies
    China

    Email: mach.chen@huawei.com

    Manav Bhatia
    Alcatel-Lucent
    India

    Email: manav.bhatia@alcatel-lucent.com