Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 16, 2011

M. Bhatia
Alcatel-Lucent

April 14, 2011

Analysis of Protocol Independent Multicast Sparse Mode (PIM-SM) Security According to KARP Design Guide
draft-bhatia-karp-pim-gap-analysis-00

Abstract

This document analyzes Protocol Independent Multicast Sparse Mode (PIM-SM) according to the guidelines set forth in the KARP Design Guide.

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 October 16, 2011.

1. Introduction

This document performs the initial analysis of the current state of Protocol Independent Multicast Sparse Mode (PIM-SM) [RFC4601] according to the requirements of [I-D.ietf-karp-design-guide]

[RFC5796] describes mechanisms to authenticate the PIM-SM link-local messages using the IP security (IPsec) Encapsulating Security Payload (ESP) [RFC4303] or (optionally) the Authentication Header (AH) [RFC4302] .

This document specifies manual key management as mandatory to implement, i.e., that all implementations MUST support, and provides the necessary structure for an automated key management protocol that the PIM routers may use.

However, some gaps remain between the current state and the requirements for manually keyed routing security expressed in the [I-D.ietf-karp-threats-reqs] document. This document explores these gaps and proposes directions for addressing the gaps.

2. Current State and Gap Analysis

[RFC5796] describes how IPsec can be used to secure and authenticate PIM-SM protocol packets. It mandates the use of manual keying and optionally provides support for an automated group key management mechanism. However, it leaves the procedures for implementing automated group key management to other documents and does not discuss how this can be done.

[RFC5796] uses manually configured keys, rather than some automated key management protocol , since no suitable key management mechanism is available at this time. This is because PIM-SM adjacencies are formed on a one-to-many basis and most key management mechanisms are designed for a one-to-one communication model. Since [RFC5796] uses manual keying it clearly states that it provides no protection against both inter-session and intra-session replay attacks. This can be exploited in several ways.

Since multiple PIM-SM routers can exist on a single link, it would be

worth noting that setting up IPsec Security Associations (SAs) manually can be a very tedious process. The routers might not even support IPsec, rendering automatic key negotiation either impractical (in those platforms where an extra license has to be obtained for using IPsec) or infeasible (in those platforms where IPsec support is not available at all).

While I don't yet see a need to prioritize certain PIM-SM packets over the others, it should be noted that this would be extremely difficult to achieve since PIM-SM uses IPsec for its security and authentication.

[RFC4601] requires all PIM-SM routers to configure an IPsec Security Association (SA) when sending PIM Register packets to each Rendezous Point (RP). This can become highly unscalable as the number of RPs increase or in case of Anycast-RP [RFC4610] deployment where each PIM-SM router close to the source will need to establish IPsec tunnels to all PIM-SM routers in the Anycast-RP set.

Similarly, the Security Policy Database at each Rendezvous Point should be configured to choose an SA to use when sending Register-Stop messages. Because Register-Stop messages are unicast to the destination DR, a different SA and a potentially unique SPI are required for each DR.

In order to simplify the management problem, [RFC4601] suggests using the same authentication algorithm and authentication parameters, regardless of the sending RP and regardless of the destination DR. While this alleviates the management problem by some extent it still requires a unique SA on each DR which can result in a significant scaling issue as the size of the PIM-SM network grows.

In order to encourage deployment of PIM-SM security, an authentication option is required that does not have the deployment challenges of IPsec. We thus need an authentication mechanism alternate to IPsec as part of the first phase of the KARP design guide where we secure the routing protocols using manual keying.

The new mechanism should work for both the Unicast and Multicast PIM-SM routing exchanges. It should also provide both inter-session and intra-session replay protection that has been spelled out in the [I-D.ietf-karp-threats-reqs] document.

3. Security Considerations

TBD

4. IANA Considerations

This document places no new request to IANA

5. Acknowledgements

I would like to thank Stig Venaas and Bill Atwood for reviewing and providing feedback on this draft.

6. References

6.1. Normative References

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

[RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

[RFC5796] Atwood, W., Islam, S., and M. Siami, "Authentication and Confidentiality in Protocol Independent Multicast Sparse Mode (PIM-SM) Link-Local Messages", RFC 5796, March 2010.

6.2. Informative References

[I-D.ietf-karp-design-guide] Lebovitz, G. and M. Bhatia, "Keying and Authentication for Routing Protocols (KARP) Design Guidelines", draft-ietf-karp-design-guide-02 (work in progress), March 2011.

[I-D.ietf-karp-threats-reqs] Lebovitz, G., Bhatia, M., and R. White, "The Threat Analysis and Requirements for Cryptographic Authentication of Routing Protocols' Transports", draft-ietf-karp-threats-reqs-01 (work in progress), October 2010.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, December 2005.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

[RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol",

RFC 4306, December 2005.

[RFC4610] Farinacci, D. and Y. Cai, "Anycast-RP Using Protocol Independent Multicast (PIM)", RFC 4610, August 2006.

Author's Address

Manav Bhatia
Alcatel-Lucent
India

Email: manav.bhatia@alcatel-lucent.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 31, 2011

M. Bhatia
Alcatel-Lucent
D. Zhang
Huawei Technologies co., LTD.

April 29, 2011

Analysis of Bidirectional Forwarding Detection (BFD) Security According to KARP Design Guide
draft-bhatia-zhang-karp-bfd-analysis-01

Abstract

This document analyzes the Bidirectional Forwarding Detection protocol (BFD) according to the guidelines set forth in section 4.2 of draft-ietf-karp-design-guide.

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

Copyright Notice

Table of Contents

1. Introduction

This document performs a gap analysis of the current state of BFD [RFC5880] according to the requirements of [I-D.ietf-karp-design-guide]. Previously, the OPSEC working group has provided an analysis of cryptographic issues with BFD in [RFC6039].

The existing BFD specifications provide a very initial security solution. Key ID is provided so that the key used in securing a packet can be changed on demand. Two cryptographic algorithms (MD5 and SHA-1) are supported for Integrity protection of the control packets; the algorithms are both demonstrated to be subject to collision attacks. While these attacks will not necessarily affect BFD, other routing protocols like RIPv2 [RFC4822], IS-IS [RFC5310] and OSPFv2 [RFC5709] have moved to stronger algorithms and it is imperative that BFD also does that as it does not make much sense to secure these routing protocols with a stronger authentication algorithm if BFD continues using a weaker security algorithm.

While BFD uses a non-decreasing per-packet sequence number to protect itself from intra-connection replay attacks, it still leaves the protocol vulnerable to the inter-session replay attacks.

2. Requirements to Meet

There are several requirements described in section 3 of [I-D.ietf-karp-threats-reqs] that BFD does not currently meet:

Replay Protection: BFD provides an incomplete intra-session and no inter-session replay attack protection; this creates significant denial-of-service opportunities.

Strong Algorithms: the cryptographic algorithms adopted for message authentication in BFD are MD5 or SHA-1 based. However, both algorithms are known to be vulnerable to collision attacks. [I-D.bhatia-bfd-crypto-auth] proposes a solution to support HMAC with the SHA-1 and SHA-2 family of hash functions for BFD.

DoS Attacks: BFD packets can be sent at millisecond intervals (the protocol uses timers at microsecond intervals). When malicious packets are sent at short intervals, with the authentication bit set, it can cause a DoS attack.

The remainder of this document explains the details of how these requirements fail to be met and proposes mechanisms for addressing them.

3. Current State

[RFC5880] describes five authentication mechanisms for the integrity protection of BFD control packets: Simple Password, Keyed MD5 [RFC1321], Meticulous Keyed MD5, Keyed SHA-1 and Meticulous SHA-1. In the simple password mechanism, every control packet is associated with a password transported in plain text; attacks eavesdropping the network traffic can easily learn the password and compromise the security of the corresponding BFD session. In the Keyed MD5 and the Meticulous Keyed MD5 mechanisms, BFD nodes use share secret keys to generate keyed MD5 digests for control packets. Similarly, in the Keyed SHA-1 and the Meticulous Keyed SHA-1 mechanisms, BFD nodes use shared secret keys to generate keyed SHA-1 digests for control packets. Note that in the keyed authentication mechanisms, every BFD control packet is associated with a non-decreasing 32-bit sequence number to resist replay attacks. In the Keyed MD5 and the Keyed SHA-1 mechanisms, the sequence member is only required to increase occasionally. However, in the Meticulous Keyed MD5 and the Meticulous Keyed SHA-1 mechanisms, the sequence member is required to monotonically increase with each succesive packet.

Additionally, limited key updating functionality is provided. There is a Key ID in every authenticated BFD control packet, indicating the key used to hash the packet. However, there is no mechanism described to provide a smooth key rollover that the BFD routers can use when moving from one key to the other.

The BFD session timers are defined with the granularity of microseconds, and it is common in practice to send BFD packets at millisecond intervals. Since the cryptographic sequence number space is only 32 bits, a sequence number used in a BFD session may reach its maximum value and roll over within limited period. For instance, if a sequence number is increased by one every millisecond, then it will reach its maximum value in less than 8 weeks. This can result in potential inter-session replay attacks especially when BFD uses the non-meticulous authentication modes.

Note that when using authentication mechanisms, BFD requests the sequence of a received BFD packets drops with a limited range (3* Detection time multiplier). Therefore, when meticulous authentication modes are used, a replayed BFD packet will be rejected if it cannot fit into a relatively short window (3 times of the detect interval of the session). This introduces some difficulties for replaying packets. However, in a non-meticulous authentication mode, such windows can be large as sequence numbers are only increased occassionally, thus making it easier to perform replay attacks .

In a BFD session, each node needs to select a 32-bit discriminator to identify itself. Therefore, a BFD session is identified by two discriminators. If a node will randomly select a new discriminator for a new session and use authentication mechanism to secure the control packets, inter-session replay attacks can be mitigated to some extent. However, in existing BFD demultiplexing mechanisms, the discriminators used in a new BFD session may be predictable. In some deployment scenarios, the discriminators of BFD routers may be decided by the destination and source addresses. So, if the sequence number of a BFD router rolls over for some reasons (e.g., reboot), the discriminators used to identify the new session will be identical to the ones used in the previous session. This makes performing a reply attack relatively simple.

BFD allows a mode called the echo mode. Echo packets are not defined in the BFD specification, though they can keep the BFD session up. The format of the echo packet is local to the sending side and there are no guidelines on the properties of these packets beyond the choice of the source and destination addresses. While the BFD specification recommends applying security mechanisms to prevent spoofing of these packets, there are no guidelines on what type of mechanisms are appropriate.

4. Impacts of BFD Replays

As discussed, BFD cannot meet the requirements of inter-session or intra-session replay protection. This section discusses the impacts of BFD replays.

When cryptographic authentication mechanisms are adopted for BFD, a non-decreasing 32-bit long sequence number is used. In the Keyed MD5 and the Keyed SHA-1 mechanisms, the sequence member is not required to increase for every packet. Therefore an attacker can keep replaying the packets with the latest sequence number until the sequence number is updated. This issue is eliminated in the Meticulous Keyed MD5 and the Meticulous Keyed SHA-1 mechanisms. However, note that a sequence number may reach its maximum and be rolled over in a session. In this case, without the support from a automatic key management mechanism, the BFD session will be vulnerable to replay attacks performed by sending the packets before the roll over of the sequence number. For instance, an attacker can replay a packet with a sequence number which is larger than the current one. If the replayed packet is accepted, the victim will reject the legal packets whose sequence members are less than the one in the replayed packet. Therefore, the attacker can get a good chance to bring down the BFD session.

Additionally, the BFD specification allows for the change of authentication state based on the state of a received packet. For instance, according to [RFC5880], if the state of a accepted packet is down, the receiver of the packet needs to transfer its state to down as well. Therefore, an elaborately selected replayed packet can cause a serious denial-of-service attack.

BFD does not provide any solution to deal with inter-session replay attacks. If two subsequent BFD sessions adopt an identical discriminator pair and use the same cryptographic key to secure the control packets, it is intuitive to use a malicious authenticated packet (stored from the past session) to perform inter-connection replay attacks.

Any security issues in the BFD echo mode will directly affect the BFD protocol and session states, and hence the network stability. For instance, any replay attacks would be indistinguishable from normal forwarding of the tested router. An attack would still cause a faulty link to be believed to be up, but there is little that can be done about it. However, if the echo packets are guessable, it may be possible to spoof from an external source and cause BFD to believe that a one-way link is really bidirectional. As a result, it is important that the echo packets contain random material that is also checked upon reception.

5. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

6. Security Considerations

7. Acknowledgements

We would like to thank Alexander Vainshtein for his comments on this document.

8. References

8.1. Normative References

[RFC1321] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.

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

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, June 2010.

[RFC6039] Manral, V., Bhatia, M., Jaeggli, J., and R. White, "Issues with Existing Cryptographic Protection Methods for Routing Protocols", RFC 6039, October 2010.

8.2. Informative References

[I-D.bhatia-bfd-crypto-auth] Bhatia, M. and V. Manral, "BFD Generic Cryptographic Authentication", draft-bhatia-bfd-crypto-auth-03 (work in progress), January 2011.

[I-D.ietf-karp-design-guide] Lebovitz, G. and M. Bhatia, "Keying and Authentication for Routing Protocols (KARP) Design Guidelines", draft-ietf-karp-design-guide-02 (work in progress), March 2011.

[I-D.ietf-karp-threats-reqs] Lebovitz, G., Bhatia, M., and R. White, "The Threat Analysis and Requirements for Cryptographic Authentication of Routing Protocols' Transports", draft-ietf-karp-threats-reqs-02 (work in progress), April 2011.

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

Authors' Addresses

Manav Bhatia
Alcatel-Lucent
Bangalore,
India

Phone:
Email: manav.bhatia@alcatel-lucent.com

Dacheng Zhang
Huawei Technologies co., LTD.
Beijing,
China

Phone:
Fax:
Email: zhangdacheng@huawei.com
URI:

Routing Working Group
Internet-Draft
Intended status: Informational
Expires: December 29, 2011

M. Jethanandani
K. Patel
Cisco Systems, Inc
L. Zheng
Huawei

June 27, 2011

Analysis of BGP, LDP, PCEP, and MSDP Security According to KARP Design Guide

draft-ietf-karp-routing-tcp-analysis-00.txt

Abstract

This document analyzes BGP, LDP, PCEP and MSDP according to guidelines set forth in section 4.2 of [draft-ietf-karp-design-guide].

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 December 29, 2011.

Copyright Notice

Table of Contents

1. Introduction

In March 2006 the Internet Architecture Board (IAB) in its "Unwanted Internet Traffic" workshop described an attack on core routing infrastructure as an ideal attack with the most amount of damage. It called for the tightening the security of the core routing infrastructure.

This document performs the initial analysis of the current state of BGP, LDP, PCEP and MSDP according to the requirements of [draft-ietf-karp-design-guide]. This draft builds on several previous analysis efforts into routing security. The OPSEC working group put together Issues with existing Cryptographic Protection Methods for Routing Protocols [draft-ietf-opsec-routing-protocols-crypto-issues] an analysis of cryptographic issues with routing protocols and draft-hartman-ospf-analysis-01 which has a analysis for OSPF.

Section 2 looks at the current state of the four routing protocols. Section 3 goes into what the optimal state would be for the three routing protocols according to KARP Design Guidelines [draft-ietf-karp-design-guide] and Section 4 does a analysis of the gap between the existing state and the optimal state of the protocols and suggest some areas where we need to improve.

1.1. Contributing Authors

Anantha Ramaiah, Mach Chen

1.2. Abbreviations

BGP - Border Gateway Protocol

DoS - Denial of Service

KARP - Key and Authentication for Routing Protocols

KDF - Key Derivation Function

KEK - Key Encrypting Key

KMP - Key Management Protocol

LDP - Label Distribution Protocol

LSR - Label Switch Routers

MAC - Message Authentication Code

MKT - Master Key Tuple

MSDP - Multicast Source Distribution Protocol

MD5 - Message Digest algorithm 5

OSPF - OPen Shortest Path First

PCEP - Path Computation Element Protocol

TCP - Transmission Control Protocol

UDP - User Datagram Protocol

2. Current State of BGP, LDP, PCEP and MSDP

This section looks at the underlying transport protocol and key mechanisms built for the protocol. It describes the security mechanisms built into BGP, LDP, PCEP and MSDP.

2.1. Transport level

At a transport level, routing protocols are subject to a variety of DoS attacks. Such attacks can cause the routing protocol to become congested with the result that routing updates are supplied too slowly to be useful or in extreme case prevent route convergence after a change.

Routing protocols use several methods to protect themselves. Those that run on TCP use access list to permit packets only from know sources. These access lists also help edge routers from attacks originating from outside the protected cloud. In addition for edge routers running eBGP, TCP LISTEN is run only on interfaces on which its peers have been discovered or that are configured to expect sessions on.

GTSM [RFC5082] describes a generalized Time to Live (TTL) security mechanism to protect a protocol stack from CPU-utilization based attacks.TCP Robustness [RFC5961] recommends some TCP level mitigations against spoofing attacks targeted towards long lived routing protocol sessions.

Even when BGP, LDP, PCEP and MSDP sessions use access list they are subject to spoofing and man in the middle attacks. Authentication and integrity checks allow the receiver of a routing protocol update to know that the message genuinely comes from the node that purports to have sent it and to know whether the message has been modified.

TCP MD5 [RFC2385] specifies such a mechanism to protect BGP and other TCP based routing protocols via the TCP MD5 option. TCP MD5 option provides a way for carrying an MD5 digest in a TCP segment. This digest acts like a signature for that segment, incorporating information known only to the connection end points. The MD5 key used to compute the digest is stored locally on the router. This option is used by routing protocols to provide for session level protection against the introduction of spoofed TCP segments into any existing TCP streams, in particular TCP Reset segments. TCP MD5 does not provide a generic mechanism to support key roll-over.

However, the Message Authentication Codes (MACs) used by MD5 to compute the signature are considered to be too weak. TCP-AO [RFC5925] and its companion documentCrypto Algorithms for TCP-AO

[RFC5926] is a step towards correcting both the MAC weakness and KMP. For MAC it specifies two MAC algorithms that MUST be supported. They are HMAC-SHA-1-96 as specified in HMAC [RFC2104] and AES-128-CMAC-96 as specified in NIST-SP800-38B [NIST-SP800-38B]. Cryptographic research suggests that both these MAC algorithms defined are fairly secure and are not known to be broken in any ways. It also provides for additional MACs to be added in the future.

2.2. Keying mechanisms

For TCP-AO [RFC5925] there is no Key Management Protocol (KMP) used to manage the keys that are used for generating the Message Authentication Code (MAC). It allows for a master key to be configured manually or for it to be managed from a out of band mechanism. Most routers are configured with a static key that does not change over the life of the session.

For point-to-point key management IKE [RFC2409] tries to solve the issue of key exchange under a SA.

2.3. LDP

Section 5 of LDP [RFC5036] states that LDP is subject to three different types of attacks. It talks about spoofing, protection of privacy of label distribution and denial of service attacks.

2.3.1. Spoofing attacks

Spoofing attack for LDP occur both during the discovery phase and during the session communication phase.

2.3.1.1. Discovery exchanges using UDP

Label Switching Routers (LSRs) indicate their willingness to establish and maintain LDP sessions by periodically sending Hello messages. Receipt of a Hello message serves to create a new "Hello adjacency", if one does not already exist, or to refresh an existing one.

Unlike all other LDP messages, the Hello messages are sent using UDP not TCP. This means that they cannot benefit from the security mechanisms available with TCP. LDP [RFC5036] does not provide any security mechanisms for use with Hello messages except to note that some configuration may help protect against bogus discovery events.

Spoofing a Hello packet for an existing adjacency can cause the adjacency to time out and that can result in termination of the associated session. This can occur when the spoofed Hello message

specifies a small Hold Time, causing the receiver to expect Hello messages within this interval, while the true neighbor continues sending Hello messages at the lower, previously agreed to, frequency.

Spoofing a Hello packet can also cause the LDP session to be terminated directly. This can occur when the spoofed Hello specifies a different Transport Address from the previously agreed one between neighbors. Spoofed Hello messages are observed and reported as real problem in production networks.

2.3.1.2. Session communication using TCP

LDP like other TCP based routing protocols specifies use of the TCP MD5 Signature Option to provide for the authenticity and integrity of session messages. As stated above, some assert that MD5 authentication is now considered by some to be too weak for this application. A stronger hashing algorithm e.g SHA1, could be deployed to take care of the weakness.

2.3.2. Privacy Issues

LDP provides no mechanism for protecting the privacy of label distribution. The security requirements of label distribution are similar to other routing protocols that need to distribute routing information.

2.3.3. Denial of Service Attacks

LDP is subject to Denial of Service (DoS) attacks both in its discovery mode as well as during the session mode.

The discovery mode attack is similar to the spoofing attack except that when the spoofed Hello messages are sent with a high enough frequency can cause the adjacency to time out.

2.4. PCEP

Attacks on PCEP [RFC5440] may result in damage to active networks. This may include computation responses, which if changed can cause protocols like LDP to setup sub-optimal or inappropriate LSPs. In addition, PCE itself can be attacked by a variety of DoS attacks. Such attacks can cause path computations to be supplied too slowly to be of any value particularly as it relates to recovery or establishment of LSPs.

As the RFC states, PCEP could be the target of the following attacks.

- Spoofing (PCC or PCE implementation)
- Snooping (message interception)
- Falsification
- Denial of Service

According to the RFC, inter-AS scenarios when PCE-to-PCE communication is required, attacks may be particularly significant with commercial as well as service-level implications.

Additionally, snooping of PCEP requests and responses may give an attacker information about the operation of the network. Simply by viewing the PCEP messages someone can determine the pattern of service establishment in the network and can know where traffic is being routed, thereby making the network susceptible to targeted attacks and the data within specific LSPs vulnerable.

Ensuring PCEP communication privacy is of key importance, especially in an inter-AS context, where PCEP communication end-points do not reside in the same AS, as an attacker that intercepts a PCE message could obtain sensitive information related to computed paths and resources.

2.5. MSDP

Similar to BGP and LDP, TCP MD5 [RFC2385] specifies a mechanism to protect TCP sessions via the TCP MD5 option. But with a weak MD5 authentication, TCP MD5 is considered too weak for this application.

MSDP also advocates imposing a limit on number of source address and group addresses (S,G) that can be stored within the protocol and thereby mitigate state explosion due to any denial of service and other attacks.

3. Optimal State for BGP, LDP, PCEP, and MSDP

The ideal state for BGP, LDP and MSDP protocols are when they can withstand any of the known types of attacks.

Additionally, Key Management Protocol (KMP) for the routing sessions should help negotiate unique, pair wise random keys without administrator involvement. It should also negotiate Security Association (SA) parameter required for the session connection, including key life times. It should keep track of those lifetimes and negotiate new keys and parameters before they expire and do so without administrator involvement. In the event of a breach, the keys should be changed immediately.

The DoS attacks for BGP, LDP, PCEP and MSDP are attacks to the transport protocol, TCP in this case. TCP should be able to withstand any of DoS scenarios by dropping packets that are attack packets in a way that does not impact legitimate packets.

The routing protocols should provide a mechanism to determine authenticate and validate the routing information carried within the payload.

3.1. LDP

For the spoofing kind of attacks that LDP is vulnerable to during the discovery phase, it should be able to determine the authenticity of the neighbors sending the Hello message.

There is currently no requirement to protect the privacy of label distribution as labels are carried in the clear like other routing information.

4. Gap Analysis for BGP, LDP, PCEP and MSDP

This section outlines the differences between the current state of the routing protocol and the desired state as outlined in section 4.2 of KARP Design Guidelines [draft-ietf-karp-design-guide]. It covers issues that are common to the four protocols leaving protocol specific issues to sub-sections.

At a transport level the routing protocols are subject to some of the same attacks that TCP applications are subject to. These include but are not limited to DoS attacks. Recommendations to make the transport protocol should be followed and implemented. An example of such a draft is Improving TCP’s Robustness to Blind In-Window Attacks. [RFC5961]

From a security perspective we lack comprehensive KMP. As an example TCP-AO [RFC5925] talks about coordinating keys derived from MKT between endpoints, but the MKT itself has to be configured manually or through a out of band mechanism. Even when keys are configured manually, a method for their rollover has not been defined. This leads to keys not being updated regularly which in itself increases the security risk. Also TCP-AO does not address the issue of connectionless reset.

Authentication, tamper protection, and encryption all require the use of keys by sender and receiver. An automated KMP therefore has to include a way to distribute MKT between two end points with little or no administration overhead. It has to cover automatic key rollover.

There are two methods of automatic key rollover. Implicit key rollover can be initiated after certain volume of data gets exchanged or when a certain time has elapsed. This does not require explicit signaling. On the other hand, explicit key rollover requires a out of band key signaling mechanism. An example of this is IKE [RFC2409] but it could be any other new mechanisms also.

There is a need to protect authenticity and validity of the routing/ label information that is carried in the payload of the sessions. However, we believe that is outside the scope of this document at this time and is being addressed by SIDR WG. Similar mechanisms could be used for intra-domain protocols.

4.1. LDP

As described in LDP [RFC5036], the threat of spoofed Basic Hellos can be reduced by accepting Basic Hellos on interfaces that LSRs trust, employing GTSM [RFC5082] and ignoring Basic Hellos not addressed to the "all routers on this subnet" multicast group. Spoofing attacks

via Extended Hellos are potentially a more serious threat. An LSR can reduce the threat of spoofed Extended Hellos by filtering them and accepting Hellos from sources permitted by an access list. However, performing the filtering using access lists requires LSR resource, and the LSR is still vulnerable to the IP source address spoofing. Spoofing attacks can be solved by being able to authenticate the Hello messages, and an LSR can be configured to only accept Hello messages from specific peers when authentication is in use.

LDP Hello Cryptographic Authentication [draft-zheng-mpls-ldp-hello-crypto-auth-01] suggest a new Cryptographic Authentication TLV that can be used as an authentication mechanism to secure Hello messages.

4.2. PCEP

PCE discovery according to its RFC is a significant feature for the successful deployment of PCEP in large networks. This mechanism allows a PCC to discover the existence of suitable PCEs within the network without the necessity of configuration. It should be obvious that, where PCEs are discovered and not configured, the PCC cannot know the correct key to use. There are different approaches to retain some aspect of security, but all of them require use of a keys and a keying mechanism, the need for which has been discussed above.

5. Security Requirements

This section describes requirements for BGP, LDP, PCEP and MSDP security that should be met within the routing protocol.

As with all routing protocols, they need protection from both on-path and off-path blind attacks. A better way to protect them would be with per-packet protection using a cryptographic MAC. In order to provide for the MAC, keys are needed.

Once keys are used, mechanisms are required to support key rollover. This should cover both manual and automatic key rollover. Multiple approaches could be used. However since the existing mechanisms provide a protocol field to identify the key as well as management mechanisms to introduce and retire new keys, focusing on the existing mechanism as a starting point is prudent.

Finally, replay protection is required. The replay mechanism needs to be sufficient to prevent an attacker from creating a denial of service or disrupting the integrity of the routing protocol by replaying packets. It is important that an attacker not be able to disrupt service by capturing packets and waiting for replay state to be lost.

6. Acknowledgements

We would like to thank Brian Weis for encouraging us to write this draft and providing comments on it.

7. References

7.1. Normative References

[RFC2385] Heffernan, A., "Protection of BGP Sessions via the TCP MD5 Signature Option", RFC 2385, August 1998.

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

[draft-ietf-karp-design-guide] Lebovitz, G., "KARP Design Guidelines", September 2010.

7.2. Informative References

[NIST-SP800-38B] Dworking, "Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication", May 2005.

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

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

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[RFC3547] Baugher, M., Weis, B., Hardjono, T., and H. Harney, "The Group Domain of Interpretation", RFC 3547, July 2003.

[RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

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

[RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, October 2007.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP

Authentication Option", RFC 5925, June 2010.

[RFC5961] Ramaiah, A., Stewart, R., and M. Dalal, "Improving TCP's Robustness to Blind In-Window Attacks", RFC 5961, August 2010.

[draft-ietf-opsec-routing-protocols-crypto-issues] Manral, "Issues with existing Cryptographic Protection Methods for Routing Protocols", September 2010.

[draft-zheng-mpls-ldp-hello-crypto-auth-01] Zheng, "LDP Hello Cryptographic Authentication", March 2011.

Authors' Addresses

Mahesh Jethanandani
Cisco Systems, Inc
170 Tasman Drive
San Jose, CA 95134
USA

Phone: +1 (408) 527-8230
Email: mjethanandani@gmail.com

Keyur Patel
Cisco Systems, Inc
170 Tasman Drive
San Jose, CA 95134
USA

Phone: +1 (408) 526-7183
Email: keyupate@cisco.com

Lianshu Zheng
Huawei
No. 3 Xinxi Road, Hai-Dian District
Beijing, 100085
China

Phone: +86 (10) 82882008
Fax:
Email: verozheng@huawei.com
URI:

OSPF Working Group
Internet-Draft
Intended status: Standards Track
Expires: November 7, 2011

M. Bhatia
Alcatel-Lucent
S. Hartman
Painless Security
D. Zhang
Huawei Technologies co., LTD.
A. Lindem
Ericsson
May 6, 2011

Security Extension for OSPFv2 when using Manual Key Management
draft-ietf-ospf-security-extension-manual-keying-00

Abstract

The current OSPFv2 cryptographic authentication mechanism as defined in the OSPF standards is vulnerable to both inter-session and intra-session replay attacks when its uses manual keying. Additionally, the existing cryptographic authentication schemes do not cover the IP header. This omission can be exploited to carry out various types of attacks.

This draft proposes changes to the authentication sequence number mechanism that will protect OSPFv2 from both inter-session and intra-session replay attacks when its using manual keys for securing its protocol packets. Additionally, we also describe some changes in the cryptographic hash computation so that we eliminate most attacks that result because OSPFv2 does not protect the IP header.

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 November 7, 2011.

Copyright Notice

Table of Contents

1. Introduction

The OSPFv2 cryptographic authentication mechanism as described in [[RFC2328]] uses per-packet sequence numbers to provide protection against replay attacks. The sequence numbers increase monotonically so that the attempts to replay the stale packets can be thwarted. The sequence number values are maintained as a part of adjacency states. Therefore, if an adjacency is broken down, the associated sequence numbers get reinitialized and the neighbors start all over again. Additionally, the cryptographic authentication mechanism does not specify how to deal with the rollover of a sequence number when its value would wrap. These omissions can be taken advantage of by attackers to implement various replay attacks ([RFC6039]). In order to address these issues, we propose extensions to the authentication sequence number mechanism. Compared with the cryptographic authentication mechanism proposed in [RFC5709], the solution proposed does not impose any more security presumption.

The cryptographic authentication as described in [RFC2328] and later updated in [RFC5709] does not include the IP header. This also can be exploited to launch several attacks as the source address in the IP header is no longer protected. The OSPF specification, for broadcast and NBMA (Non-Broadcast Multi-Access Networks), requires the implementations to look at the source address in the IP header to determine the neighbor from witch the packet was received. Changing the IP source address of a packet which can confuse the receiver and can be exploited to produce a number of denial of service attacks [RFC6039]. If the packet is interpreted as coming from a different neighbor, the sequence number received from the neighbor may be updated. This may disrupt communication with the legitimate neighbor. Hello packets may be reflected to cause a neighbor to appear to have one-way communication. Old Database descriptions may be reflected in cases where the per-packet sequence numbers are sufficiently divergent in order to disrupt an adjacency [I-D.ietf-karp-ospf-analysis]. This is referred to as the IP layer issue in [I-D.ietf-karp-threats-reqs].

[RFC2328] states that implementations MUST offer keyed MD5 authentication. It is likely that this will be deprecated in favor of the stronger algorithms described in [RFC5709] in future deployments [I-D.ietf-opsec-igp-crypto-requirements].

This draft proposes a simple change in the cryptographic authentication mechanism, as currently described in [RFC5709], to prevent such IP layer attacks.

1.1. Requirements Section

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 RFC2119 [RFC2119].

When used in lowercase, these words convey their typical use in common language, and are not to be interpreted as described in RFC2119 [RFC2119].

2. Replay Protection using Extended Sequence Numbers

In order to provide replay protection against both inter-session and intra-session replay attacks, the OSPFv2 sequence number is expanded to 64-bits with the least significant 32-bit value containing a strictly increasing sequence number and the most significant 32-bit value containing the boot count. OSPFv2 implementations are required to retain the boot count in non-volatile storage for the deployment life the OSPF router. The requirement to preserve the boot count is also placed on SNMP agents by the SNMPv3 security architecture (refer to snmpEngineBoots in [RFC4222].

Since there is no room in the OSPFv2 packet for a 64-bit sequence number, it will occupy the 8 octets following the OSPFv2 packet and MUST be included when calculating the OSPFv2 packet digest. These additional 8 bytes are not included in the OSPFv2 packet header length but are included in the OSPFv2 header Authentication Data length and the IPv4 packet header length.

The lower order 32-bit sequence number MUST be incremented for every OSPF packet sent by the OSPF router. Upon reception, the sequence number MUST be greater than the sequence number in the last OSPF packet of that type accepted from the sending OSPF neighbor. Otherwise, the OSPF packet is considered a replayed packet and dropped. OSPF packets of different types may arrive out of order if they are priorized as recommended in [RFC3414].

OSPF routers implementing this specification MUST use available mechanisms to preserve the sequence number’s strictly increasing property for the deployed life of the OSPFv3 router (including cold restarts). This is achieved by maintaining a boot count in non-volatile storage and incrementing it each time the OSPF router loses its prior sequence number state. The SNMPv3 snmpEngineBoots variable [RFC4222] MAY be used for this purpose. However, maintaining a separate boot count solely for OSPF sequence numbers has the advantage of decoupling SNMP reinitialization and OSPF reinitialization. Also, in the rare event that the lower order 32-

bit sequence number wraps, the boot count can be incremented to preserve the strictly increasing property of the aggregate sequence number. Hence, a separate OSPF boot count is RECOMMENDED.

3. OSPF Packet Extensions

The OSPF packet header includes an authentication type field, and 64-bits of data for use by the appropriate authentication scheme (determined by the type field). Authentication types 0, 1 and 2 are defined [RFC2328]. This section of this defines Authentication type TBD (3 is recommended).

When using this authentication scheme, the 64-bit Authentication field in the OSPF packet header as defined in section D.3 of [RFC2328] is changed as shown below. The sequence number is removed and the Key ID is extended to 32 bits and moved to the former position of the sequence number.

Additionally, the 64-bit sequence number is moved to the first 64-bits following the OSPFv2 packet and is protected by the authentication digest. These additional 64 bits or 8 octets are included in the IP header length but not the OSPF header packet length.

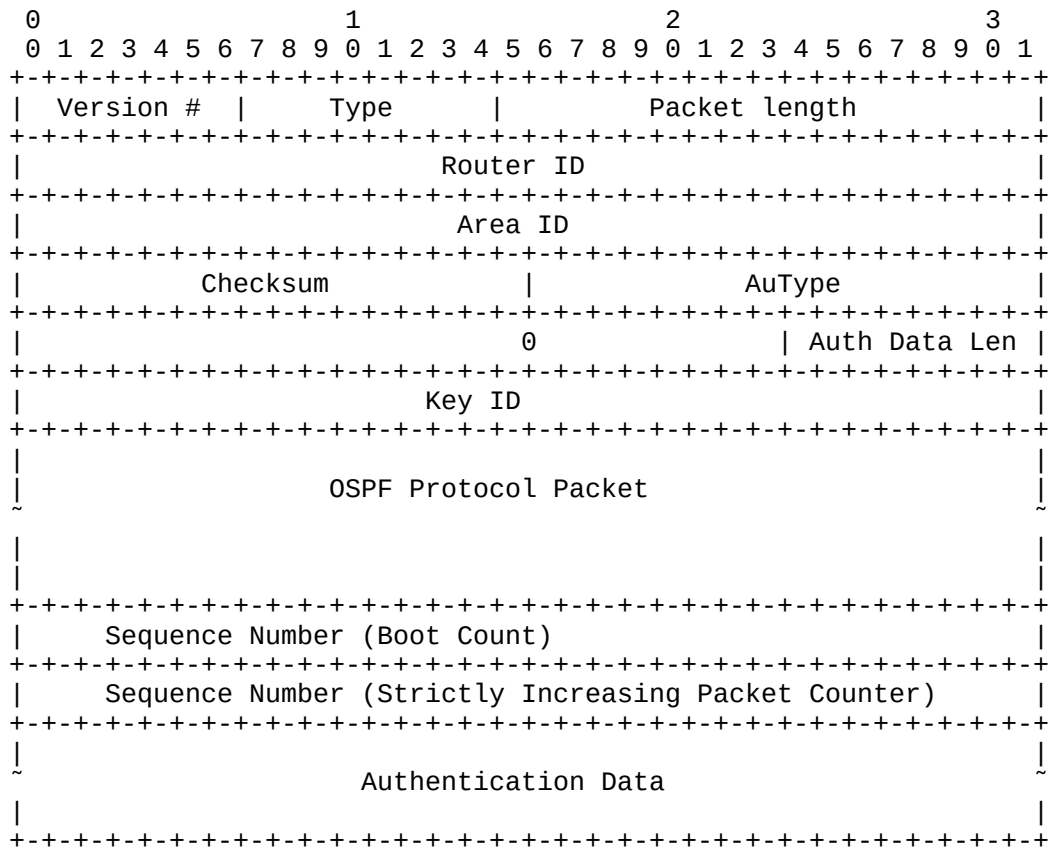


Figure 7 - Extended Sequence Number Packet Extensions

4. OSPF Packet Key Selection

This section describes how the proposed security solution selects long-lived keys from key tables. [I-D.ietf-karp-crypto-key-table]. Generally, a key used for OSPFv2 packet authentication should satisfy the following requirements:

- o The key time period as defined by NotBefore and NotAfter must include the current time.
- o The key can be used for the desired security algorithm.

In the remainder of this section, additional requirements for keys are enumerated for different scenarios.

4.1. Key Selection for Unicast OSPF Packet Transmission

Assume that a router R1 tries to send a unicast OSPF packet from its interface I1 to the interface R2 of a remote router R2 using security protocol P via interface I at time T. Firstly consider the circumstances where R1 and R2 are not connected with a virtual link. R1 then needs to select a long long-lived symmetric key from its key table. Because the key should be shared by the by both R1 and R2 to protect the communication between I1 and I2, the key should satisfy the following requirements:

- The Peer field includes the router ID of R2.
- the PeerKeyID field is not "unknown".
- The Interfaces field includes I1.
- the Direction field is either "out" or "both".

When R1 and R2 are connected to a virtual link, the third condition is a little more complex. Because the virtual link can be regarded as an unnumbered point-to-point network, the IP address of the interface actually used to send the packet (i.e., I1) is discovered during routing table calculation. Therefore, when the system operator configures keys to protect the virtual link, I1 is unknown and can be any OSPF interface in the OSPF virtual link's transit area. Therefore, the key should be identified solely by the local and remote router IDs rather than by the interface on which the packet is sent. The third requirement list above should be changed to "the Interface field includes the router ID".

4.2. Key Selection for Multicast OSPF Packet Transmission

If a router R1 sends an OSPF packet from its interface I1 to a multicast address (e.g., AllSPFRouters, AllDRouters), it needs to select a key according to the following requirements:

- The Peer field includes the multicast address.
- The PeerKeyID field is "group".
- The Interfaces field includes I1.
- The Direction field is either "out" or "both".

4.3. Key Selection for OSPF Packet Reception

When Cryptographic Authentication is employed, the ID of the authentication key is included in the authentication field of the OSPF packet header. Using this key ID, it is relatively easy for a receiver to locate the key. The simple requirements are:

- The Peer field includes the router ID of the sender.
- The PeerKeyID field includes the key ID obtained from the authentication field.
- The Direction field is either "in" or "both".

5. Mechanism to secure the IP header

This document updates the definition of Apad which is currently a constant defined in [RFC5709] to the source address from the IP header of the OSPFv2 protocol packet. The overall cryptographic authentication process defined in [RFC5709] remains unchanged. To reduce the potential for confusion, this section minimizes the repetition of text from RFC 5709 and is incorporated here by reference [RFC5709].

RFC 5709, Section 3.3, describes how the cryptographic authentication must be computed. It requires OSPFv2 packet's Authentication Trailer (which is the appendage described in RFC 2328, Section D.4.3, Page 233, items (6)(a) and (6)(d)) to be filled with the value Apad where Apad is a hexadecimal constant value 0x878FE1F3 repeated (L/4) times, where L is the length of the hash being used and is measured in octets rather than bits.

Routers at the sending side must initialize Apad to a value of the source address that would be used when sending out the OSPFv2 packet, repeated L/4 times, where L is the length of the hash, measured in octets. The basic idea is to incorporate the source address from the IP header in the cryptographic authentication computation so that any change of IP source address in a replayed packet can be detected.

At the receiving end, implementations MUST initialize Apad as the source address from IP Header of the incoming OSPFv2 packet, repeated L/4 times, instead of the constant that's currently defined in [RFC5709]. Besides changing the value of Apad, this document does not introduce any other changes to the authentication mechanism described in [RFC5709]. This would prevent all attacks where a rogue OSPF router changes the IP source address of an OSPFv2 packet and replays it on the same multi-access interface or another interface

since the IP source address is now protected and such changes would cause the authentication check to fail and the replayed packet to be rejected.

6. Security Considerations

This document attempts to fix the manual key management procedure that currently exists within OSPFv2, as part of the Phase 1 of the KARP Working Group. Therefore, only the OSPFv2 manual key management mechanism is considered. Any solution that takes advantage of the automatic key management mechanism is beyond the scope of this document.

The proposed sequence number extension offers most of the benefits of of more complicated mechanisms involving challenges. There are, however, a couple drawbacks to this approach. First, it requires the OSPF implementation to be able to save its boot count in non-volatile storage. If the non-volatile storage is ever repaired or upgraded such that the contents are lost or the OSPFv2 router is replaced with a model, the keys MUST be changed to prevent replay attacks.

Second, if a router is taken out of service completely (either intentionally or due to a persistent failure), the potential exists for reestablishment of an OSPFv2 adjacency by replaying the entire OSPFv2 session establishment. This scenario is however, extremely unlikely, since it would imply an identical OSPFv2 adjacency formation packet exchange. The replay of OSPFv2 hello packets alone for an OSPFv2 router that has been taken out of service should not result in any serious attack as the only consequence is superfluous processing. Of course, this attack could also be thwarted by changing the relevant manual keys.

This document also provides a solution to prevent certain denial of service attacks that can be launched by changing the source address in the IP header of the OSPFv2 protocol packet.

7. IANA Considerations

This document requests a new code point from the "OSPF Shortest Path First (OSPF) Authentication Codes" registry:

- TBD - Cryptographic Authentication with Extended Sequence Numbers. The value 3 is recommended.

8. References

8.1. Normative References

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

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.

[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.

8.2. Informative References

[I-D.ietf-karp-crypto-key-table] Housley, R. and T. Polk, "Database of Long-Lived Symmetric Cryptographic Keys", draft-ietf-karp-crypto-key-table-00 (work in progress), November 2010.

[I-D.ietf-karp-ospf-analysis] Hartman, S. and D. Zhang, "Analysis of OSPF Security According to KARP Design Guide", draft-ietf-karp-ospf-analysis-00 (work in progress), March 2011.

[I-D.ietf-karp-threats-reqs] Lebovitz, G., Bhatia, M., and R. White, "The Threat Analysis and Requirements for Cryptographic Authentication of Routing Protocols' Transports", draft-ietf-karp-threats-reqs-02 (work in progress), April 2011.

[I-D.ietf-opsec-igp-crypto-requirements] Bhatia, M. and V. Manral, "Summary of Cryptographic Authentication Algorithm Implementation Requirements for Routing Protocols", draft-ietf-opsec-igp-crypto-requirements-04 (work in progress), October 2010.

[RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, RFC 3414, December 2002.

[RFC4222] Choudhury, G., "Prioritized Treatment of Specific OSPF Version 2 Packets and Congestion Avoidance", BCP 112, RFC 4222, October 2005.

[RFC6039] Manral, V., Bhatia, M., Jaeggli, J., and R. White, "Issues with Existing Cryptographic Protection Methods for Routing

Protocols", RFC 6039, October 2010.

Authors' Addresses

Manav Bhatia
Alcatel-Lucent
Bangalore,
India

Phone:
Email: manav.bhatia@alcatel-lucent.com

Sam Hartman
Painless Security

Email: hartmans@painless-security.com

Dacheng Zhang
Huawei Technologies co., LTD.
Beijing,
China

Phone:
Fax:
Email: zhangdacheng@huawei.com
URI:

Acee Lindem
Ericsson
102 Carric Bend Court
Cary, NC 27519
USA

Phone:
Email: acee.lindem@ericsson.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 2, 2012

M. Jethanandani
B. Weis
K. Patel
Cisco Systems
July 1, 2011

Key Management for Pairwise Routing Protocol
draft-mahesh-karp-kmprp-00

Abstract

This document defines an automated method of Key Management for routing protocol that need pair-wise keys.

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 January 2, 2012.

Copyright Notice

Table of Contents

1. Introduction

Key management today is limited to statically configuring master keys in individual routers. This document extends currently defined IKEv2 protocol to define an automated method of Key Management for Pairwise Routing Protocol (KMPRP) that allows network devices to automatically exchange key material related information between the network devices.

2. Protocol Exchanges

The exchange of private keying material between two network devices using a dedicated key management protocol is a requirement as articulated in [I-D.ietf-karp-routing-tcp-analysis]. There is no need to define an entirely new protocol for this purpose, when existing mature protocol exchanges and methods have been vetted. This draft makes use of the IKEv2 protocol exchanges, state machine, and policy definitions to define a dedicated key management protocol. However, as IKEv2 was developed exclusively for the use of IPsec, these protocol exchanges are incorporated by reference into the present key protocol definitions, and are exchanged using a dedicated UDP port number (TDB - IANA). The use of a dedicated UDP port will clearly differentiate this protocol from IKEv2.

In the following figures, the notations contained in the message are defined as follows.

Notation	Payload
AUTH	Authentication
CERT	Certificate
CERTREQ	Certificate Request
D	Delete
HDR	KMPRP Header (not a payload)
IDi	Identification - Initiator
IDr	Identification - Responder
KE	Key Exchange
Ni, Nr	Nonce
N	Notify
SA	Security Association
SK	Encrypted and Authenticated
TSi	Traffic Selector - Initiator
TSr	Traffic Selector - Responder

Acronyms Used in Protocol Exchange

2.1. RP_INIT

The RP Initial Exchange (RP_INIT) is identical to the IKE_SA_INIT exchange defined in Internet Key Exchange Protocol Version 2 [RFC5996]. The RP_INIT exchange is a two-message exchange that allows the network devices to negotiate cryptographic algorithms, exchange nonces, and do a Diffe-Hellman (DH) [DH] exchange, for their routing protocols, after which protocols on these network devices can communicate privately. Note that at this point the network devices have not identified their peer. For the details of this exchange, refer to IKE_SA_INIT in Internet Key Exchange Protocol Version 2 [RFC5996].

```
   Peer (Initiator)                    Peer (Responder)
   --------------------                ------------------
   HDR, SAi1, KEi, Ni          -->
                               <--     HDR, SAr1, KEr, Nr, [CERTREQ,]
```

RP_INIT

2.2. RP_AUTH

Next, the network devices perform a RP Authentication exchange (RP_AUTH), which is substantially the same as the IKE_AUTH exchange defined in RFC 5996, except that the SA payload contains policy specific to the routing protocol security policy (labeled SArpi and SArpr) rather than IPsec policy (SAi2, SAr2 defined in RFC 5996). The SArpi and SArpr payloads are described in Section 3; for the details of the rest of the exchange please refer to IKE_AUTH in RFC 5996.

```
  Peer (Initiator)                      Peer (Responder)
  --------------------                  ------------------
  HDR, SK {IDi, [CERT,] [CERTREQ,]
      [IDr,] AUTH, SArpi,
      TSi, TSr}               -->
                              <--       HDR, SK {IDr, [CERT,] AUTH,
                                            SArpr, TSi, TSr}
```

RP_AUTH

In the RP_AUTH exchange, the Initiator proposes one or more sets of policies for one routing protocol in the SArpi. The Responder returns the one policy contained in SArpi that it accepts. Based on this policy, appropriate keying material is derived from the existing shared keying material. At the successful conclusion of the RP_AUTH exchange, the initiator and responder have agreed upon a single set of policy and keying material for a particular routing protocol.

2.3. RP_ADD

The network devices may then destroy the state associated with the RP SA, continuing to use the RP policy and keying material, or they may choose to retain them for the further use. If both the network devices choose to retain them, they may use the RP SA to subsequently agree upon replacement policy for the same RP, or agree upon policy and keying material for another routing protocol. Either case will require the use of the RP Additional Exchange (RP_ADD), similar to the IKEv2 CREATE_CHILD_SA exchange as defined in RFC 5996.

```
  Peer (Initiator)                      Peer (Responder)
  --------------------                  ------------------
  HDR, SK {SArpi, Ni, [KEi ],
           TSi, TSr}             -->
                                 <--    HDR, SK {SArpr, Nr, [KEr ],
                                                 TSi, TSr}
```

RP_ADD

In the RP_ADD exchange, the SA payloads in the RP_ADD exchange are used identically as in the RP_AUTH exchange. For details on the rest of the exchange, refer to the CREATE_CHILD_SA exchange as defined in RFC 5996.

2.4. INFORMATIONAL

The IKEv2 INFORMATIONAL exchange is also useful for deleting specific RP SAs or sending status information. The Notify (N) and Delete (D) payloads are as those defined by IKEv2 [IKEV2-PARAMS]. For example, if the Responder refused to accept one of Proposals sent by the Initiator, it would return an INFORMATIONAL exchange of type NO_PROPOSAL_CHOSEN instead of the response to RP_ADD.

```
     Peer (Initiator)                   Peer (Responder)
     -------------------                ------------------
     HDR, SK {[N,] [D,] ... }    -->
                                 <--    HDR, SK {[N,] [D,] ... }
```

INFORMATIONAL

3. Header and Payload Formats

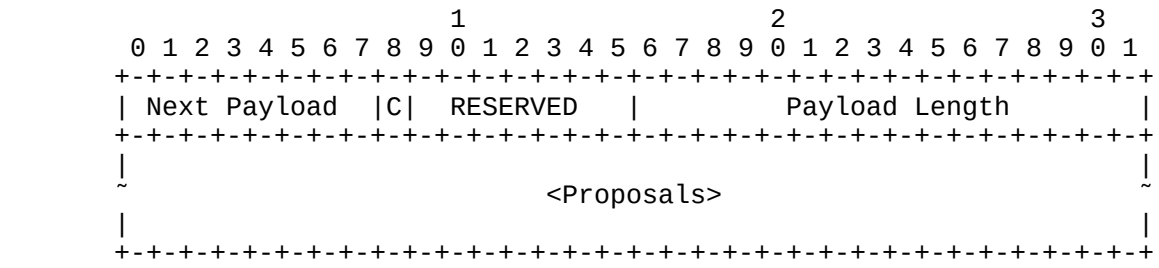
The protocol defined in this memo uses a HDR identical to the Generic Payload Header defined in section 3.2 of RFC 5996. The new exchanges defined in this memo are not used with IKEv2. A new IANA registry is to be created to identify the RP exchange types and payloads described in this section.

3.1. Security Association Payload

The Security Association (SA) payload contains a list of Proposals, which describe one or more sets of policy that a router is willing to use to protect a routing protocol. It is identical to the SA payload described in RFC 5996, and the details of the fields are described there.

In the Initiator's message, the SArpi payload contains a list of Proposal payloads (as defined in the next section), each of which contains a single set of policy that can be applied to the packets described in the Traffic Selector (TS) payloads in the same exchange. For example, the TS payloads may describe a set of IP addresses and ports which are a BGP connection, and the SA payload contains a list of proposals describing what policy the router is willing to use to protect that BGP traffic. Each set of policy is given a particular "Proposal Number" uniquely identifying this set of policy.

The responder includes a single Proposal payload in it's SA policy, which denotes the choice it has made amongst the initiator's list of Proposals. Any attributes of a selected transform MUST be returned unmodified as explained in IKEv2 [RFC5996] section 3.3.6. The initiator of an exchange MUST check that the accepted offer is consistent with one of its proposals, and if not MUST terminate the exchange.

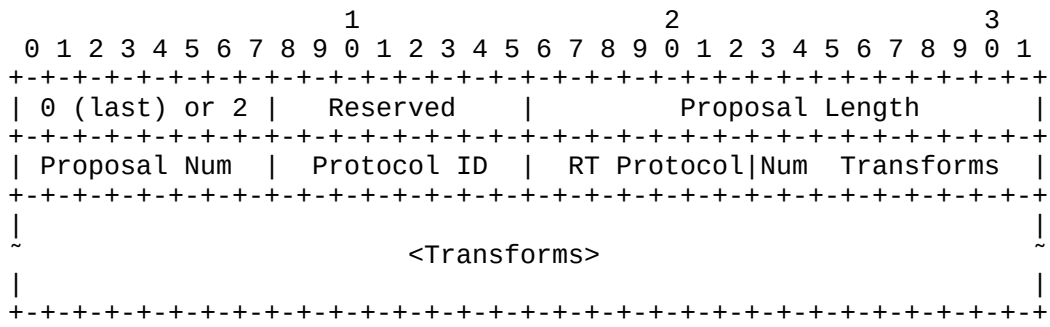


Security Association Payload

The Security Association Payload fields are defined as in RFC 5996.

3.1.1. Proposal Substructure

The Proposal (P) substructure of the Security Association Payload contains an identification for the set of policy choices, the security protocol offered in the proposal, and details of the cryptographic choices offered.



Proposal Payload

o 0 (last) or 2 (more) (1 octet) - Specifies whether this is the last Proposal Substructure in the SA.

o RESERVED (1 octet) - MUST be sent as zero; MUST be ignored on receipt.

o Proposal Length (2 octets, unsigned integer) - Length of this proposal, including all transforms and attributes that follow.

o Proposal Num (1 octet) - When a proposal is made, the first proposal in an SA payload MUST be 1, and subsequent proposals MUST be one more than the previous proposal (indicating an OR of the two proposals). When a proposal is accepted, the proposal number in the SA payload MUST match the number on the proposal sent that was accepted.

o Protocol ID (1 octet) - Specifies the protocol identifier for the current negotiation.

Protocol	Protocol ID	Reference
KMPRP	1	
TCP-AO	2	RFC 5925
LDP Discovery Key	3	TBD
Standards Action	4-128	
Private Use	129-255	

Protocol ID

o RT Protocol (1 octet) - Specifies the routing protocol identifier for the current negotiation.

Routing (RT) Protocol	Protocol ID	Reference
BGP	1	RFC 4271
LDP	2	RFC 5036
MSDP	3	RFC 3618
PIM PORT	4	
PCEP	5	RFC 5440

Routing Protocol

o Num Transforms (1 octet) - Specifies the number of transforms in this proposal.

o Transforms (variable) - One or more transform substructures.

3.1.1.1. Transforms Substructures

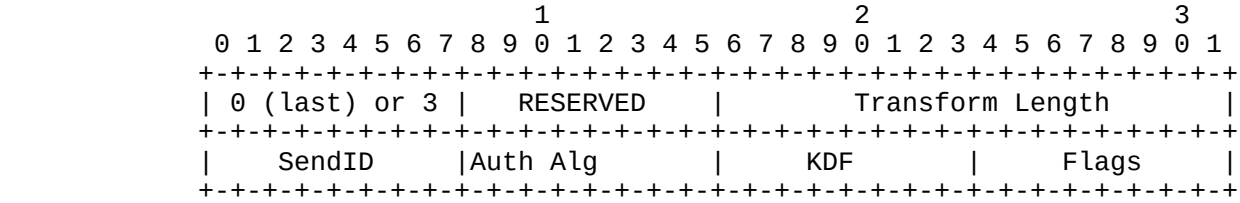
Each Proposal has a list of Transform (T) substructures, each of which describe a particular set of cryptographic policy choices. This is useful for an initiator to propose multiple cryptographic choices for the same policy described in its associated Proposal payload.

3.1.1.1.1. KMPRP

This transform payload is used negotiate policy to protect the KMPRP exchanges. The Transforms are identical to the Transforms specified to negotiate IKE policy in Section 3.3.2 of IKEv2 [RFC5996].

3.1.1.1.2. TCP-AO Transforms

The TCP-AO [RFC5925] transform payload contains the following fields.



TCP-AO Transforms

o 0 (last) or 3 (more) (1 octet) - Specifies whether this is the last Transform Substructure in the Proposal.

o RESERVED (1 octet) - MUST be sent as zero; MUST be ignored on

receipt.

o Transform Length (2 octets) - The length (in octets) of the Transform Substructure including Header and Attributes.

o SendID (1 octet) - The TCP-AO KeyID that the sender will use to represent this Transform. The KeyID will be used to generate the keys independently on each network device at the end of the exchange.

o Auth Alg (1 octet) - The Authentication algorithm defined as a part of this Transform. Values are defined in Cryptographic Algorithms for the TCP Authentication Option [RFC5926].

Auth Alg	ID
HMAC-SHA-1-96	1
AES-128-CMAC-96	2
Standards Action	3-128
Private Use	129-255

Authentication Algorithm

o KDF (1 octet) - The KDF defined as a part of this Transform. Values are defined in Cryptographic Algorithms for the TCP Authentication Option [RFC5926].

KDF	ID
KDF_HMAC_SHA1	1
KDF_AES_128_CMAC	2
Standards Action	3-128
Private Use	129-255

Key Derivation Functions

o Flags (1 octet) - Indicates specific options for TCP-AO. The bits are as follows:

In the description below, a bit being 'set' means its value is '1', while 'cleared' means its value is '0'. 'X' bits MUST be cleared when sending and MUST be ignored on receipt.

- o O (Options) - This bit indicates whether or not TCP Options are to be included in the bytes protected by the authentication calculation. This bit is set to indicate that TCP Options are to be ignored and cleared to indicate that TCP Options are protected.

When a TCP-AO transform is chosen, keying material for the TCP-AO master key is generated as follows, where Ni and Nr are unique to this exchange. The value SK_D is defined in RFC 5996, and refers to the value derived from SKEYSEED that is used to derive new keys (e.g., for TCP-AO).

<TCP-AO master key> = prf+(SK_d, Ni | Nr)

4. Operation Details

4.1. General

KMPRP is used to dynamically derive key material information between the two network devices trying to establish or maintain a routing protocol neighbor adjacency. Typically network devices running the routing protocols establish neighbor adjacencies at the routing protocol level. These routing protocols may run different security algorithms that provide transport level security for the protocol neighbor adjacencies. Depending on the security algorithm used, the routing protocols are configured with security algorithm specific keys that are either long term keys or short term session keys. These keys are specific to the security algorithms used to enforce transport level security for the routing protocols.

A routing protocol causes KMPRP to execute when it needs key material to establish neighbor adjacency. This can be as a result of the routing protocol neighbor being configured, neighbor changed or updated, a local rekey policy decision, or some other event dictated by the implementation. The key material would allow the network devices to then independently generate the same key and establish a KMPRP neighbor adjacency between them. This is typically done by the Initiator (KMPRP speaker) initiating a KMPRP RP_INIT exchange mentioned in the section 2.1 towards its KMPRP peer. As part of RP_INIT exchange, KMPRP will send a message to the KMPRP peer’s well known KMPRP UDP port [TBD] by IANA. The format of the message is explained in section 3. The procedure to exchange key information is explained in section 3. Once the key material information is successfully exchanged by both the KMPRP speaker, the KMPRP neighbor adjacency may be torn down.

The master key data received from KMPRP peers are stored in the separate Key Management Database known as KMDB. KMDB follows the

guidelines in[I-D.ietf-karp-crypto-key-table], and each entry consists of Key specific information, Security algorithm to which the Key is applicable to, Routing Protocol Clients of interest, and the announcing KMPRP Peer. KMDB is also used to notify the routing protocols about the key updates. Typically key material information is exchanged whenever a routing protocol is about to create a new neighbor adjacency. This is considered as an Initial Key exchange mode. Key material information is also exchanged to refresh existing key data on an already existing neighbor adjacency. This is considered as Key rollover exchange mode. The following sections describes their detail behavior.

4.2. Initial Key Specific Data Exchange

Routing protocols informs KMPRP of its new neighbor adjacency. It does so by creating a local entry in KMDB which consists of a Security algorithm, Key specific information, routing protocol client and the routing protocol neighbor. Upon a successful creation of such an entry KMPRP initiates KMPRP peering with the neighbor and starts initial KMPRP RP_INIT exchange explained in section 2.1 followed by the RP_AUTH exchanged explained in section 2.2. Once the key related information is successfully exchanged, KMDB may invoke the routing protocol client to provide key specific information updates if any.

4.3. Key Specific Data Rollover Exchange

Key rollover exchange may be initiated at a pre-configured time interval or as part of a manual configuration and is outside the scope of this document. The procedure of Key Rollover exchange is exactly same as the Initial Key specific data exchange described above.

5. Key Management Database (KMDB)

Protocol interaction between KMPRP and its client routing protocols is typically done using KMDB. Routing protocols update KMDB by installing a new Key related information or purging an existing Key specific information. As part of the KMDB update, KMPRP initiates peering connections with its appropriate KMPRP peers to announce the updated key related information. KMPRP may also receive an updated key related information from its peers which gets installed in KMDB. Whenever KMPRP updates KMDB with updated key information from its peers, it notifies client routing protocols of its updates.

6. Protocol Interaction

Routing protocols could end up with multiple keys when updated by KMDB. Typically, routing protocols should use the keys till the point its peers have transitioned to a new key. Once the peers have transitioned to a new key, routing protocols could put the old keys on timers and eventually free them. The reason to put them on timer and not free them right away is to ensure that all out of order packets in TCP are handled correctly.

7. IANA Considerations

A new UDP port number will need to be assigned for systems that want to implement this protocol.

A new IANA registry is to be created to identify the RP exchange types and payloads.

Note to RFC Editor: this section may be removed on publication as an RFC.

8. Security Considerations

TBD

9. Acknowledgements

During the development of TCP-AO, Gregory Lebovitz noted that a protocol based on an IKEv2 exchange would be a good automated key management method for deriving a TCP-AO master key.

Many protocol definitions and protocol formats come from RFC 5996, either by reference or inclusion.

10. References

10.1. Normative References

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

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, June 2010.

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

[RFC5996] Kaufman, C., Hoffman, P., Nir, Y., and P. Eronen, "Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 5996, September 2010.

10.2. Informative References

[DH] Diffie, W. and M. Hellman, "New Directions in Cryptography", IEEE Transactions on Information Theory, V.IT-22 n. 6, June 1977.

[I-D.ietf-karp-crypto-key-table] Housley, R. and T. Polk, "Database of Long-Lived Symmetric Cryptographic Keys", draft-ietf-karp-crypto-key-table-01 (work in progress), May 2011.

[I-D.ietf-karp-routing-tcp-analysis] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Security According to KARP Design Guide", draft-ietf-karp-routing-tcp-analysis-00 (work in progress), June 2011.

[IKEV2-PARAMS] "Internet Key Exchange Version 2 (IKEv2) Parameters", <http://www.iana.org/assignments/ikev2-parameters/ikev2-parameters.xml>.

Authors' Addresses

Mahesh Jethanandani
Cisco Systems
170 Tasman Drive
San Jose, California CA
USA

Phone: +1 (408) 527-8230
Fax:
Email: mjethanandani@gmail.com
URI:

Brian Weis
Cisco Systems
170 W. Tasman Drive
San Jose, California 95134
USA

Phone: +1 (408) 526-4796
Fax:
Email: bew@cisco.com
URI:

Keyur Patel
Cisco Systems
170 Tasman Drive
San Jose, California 95134
USA

Phone: _1 (408) 526-7183
Fax:
Email: keyupate@cisco.com
URI:

Network Working Group
Internet-Draft
Intended status: Experimental
Expires: January 4, 2012

D. Zhang
Huawei
S. Hartman
Painless Security
July 3, 2011

Unicast Router Key Management Protocol (RKMP)
draft-zhang-karp-rkmp-00.txt

Abstract

When running routing protocols such as BGP or RSVP-TE, two routers need to exchange routing messages in a unicast (one-to-one) fashion. In order to authenticate these messages using symmetric cryptography, a secret key needs to be established. This document defines a Router Key Management Protocol (RKMP) for establishing and managing such keys for routing protocols.

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 January 4, 2012.

Copyright Notice

Table of Contents

1. Introduction

Unicast and multicast both are common communication models adopted by routing protocols in exchanging routing messages. Using unicast, a message is expected to be sent to a single receiver identified by a unique address, while using multicast the same message is sent to a number of recipients.

In [I-D.hartman-karp-mrkmp], an automatic group key management mechanism is proposed for securing multicast routing message exchanges (MRKMP). This draft propose a complementary Router Key Management Protocol for securing unicast routing messages (RKMP).

Existing routing protocols using unicast (e.g., BGP, LDP, RSVP-TE) have cryptographic authentication mechanisms that use a key shared between the routers on the both sides of the communication to protect unicast routing message exchanges between the routers.

RKMP assumes that routers need to be provisioned with some credentials for a one-to-one authentication protocol. Preshared keys or asymmetric keys and an authorization list are expected to be common deployments.

If two routers running a routing protocol have not authenticated each other yet, before sending out any routing protocol packets two routers needs to perform mutual authentication using their provisioned credentials. If successful, two routers negotiate the key material to securing the routing protocol execution.

1.1. Terminology

1.2. Relationship to IKEv2

IKEv2 [RFC4306] provides a protocol for authenticating IPsec security associations between two peers. It currently provides no group keying. IKEv2 is attractive as a basis for this protocol because while it is much simpler than IKE [RFC2409], it provides all the needed flexibility in one-to-one authentication.

Unlike IKE, IKEv2 is explicitly designed for IPsec. The document does not separate handling of aspects of the protocol that would be needed for IPsec from those that apply to general key management. IPsec specific rules are combined with more general requirements. While concepts and protocol payloads can be used in a different key management protocol, the current structure of IKEv2 does not provide a mechanism for applying IKEv2 to a domain of interpretation other than IPsec. In addition, the complexity required in the IKE specification when compared to IKEv2 suggests that the generality of

IKE may not be worth the complexity cost.

So this protocol borrows concepts and payloads from IKEv2 but does not normatively depend on the IKEv2 specification.

1.3. Multicast as an Additional Feature

The base RKMP proposed in this draft aims for automatically generating keys to secure unicast routing messages. However, it can be easily extended to support authenticating multicast communications among routers. In [I-D.hartman-karp-mrkmp], the extension of RKMP in supporting multicast called MRKMP is introduced. RKMP and MRKMP can be combined to construct a integrated key management solution supporting both unicast and multicast.

2. Overview

2.1. Types of Keys

The keys adopted in RKMP is listed as follows:

PSK (Pre-Shared Key) : PSKs are pair-wise unique keys used for authenticating one router to the other one during the initial exchange. These keys are configured by some mechanism such as manual configuration or a management application outside of the scope of RKMP.

Protocol master key: A protocol master key is the key exported by RKMP for use by a routing protocol such as BGP. This is the key that is shared in the key table between the routing protocol and RKMP.

Transport key: A transport key is the key used to integrity protect routing messages in a protocol such as BGP. In today's routing protocol cryptographic authentication mechanisms the transport key can be the same as the protocol master key.

2.2. Initial Exchanges

When a router intends to send an routing message to a remote one but there is no valid RKMP_SA shared between the router and its partner, the router will perform initial exchanges with its partner to derive .

The initial exchanges is based on IKEv2's IKE_SA_INIT and IKE_SA_AUTH exchanges, which are referred to as RKMP_SA_INIT and RKMP_SA_AUTH exchanges respectively. During the initial exchanges, an initiating router attempts to authenticate to the router which it intends to

exchange unicast routing messages with. Messages are unicast from the initiator to the responding router. Unicast RKMP messages form a request/response protocol; the party sending the messages is responsible for retransmissions.

The initial exchanges provide capability negotiation, specifically including supported cryptographic suites for the key management protocol. Identification of the initiator and responder is also exchanged. A symmetric key is established to provide integrity, confidenality and authenticity protection for key management messages. These negotiation results compose a RKMP SA. While routing security does not typically require confidentiality, the key management protocol does because keys are exchanged and these must be protected.

During authentcation, the identity of each party is cryptographically verified. This can be done using, e.g., a preshared key, asymmetric keys or self-signing certificates. Other mechanisms may be added as a future extension.

The authentication exchange can also generate a SA for a routing protocol (called a child SA generally) . In the typical case, a router can obtain the needed key material (e.g., protocol master keys and maybe transport keys) for securing the desired routing protocol which in two round-trips.

2.3. Child SA Exchange

The child SA exchange is analogous to the CREATE_CHILD_SA exchange in IKEv2 and consists of a single request/response pair. However, the CREATE_CHILD_SA exchange in IKEv2 is designated for IPsec while the child SA exchange can be used to generate SAs to secure various routing protocols.

A child SA exchange can be triggered in order to 1) rekey an antique protocol master key and establish a new equivalent one, 2) generate needed key material for a newly executed routing protocol based on an existing RKMP_SA, or 3) rekey an RMKP_SA and establish a new equivalent RMKP_SA.

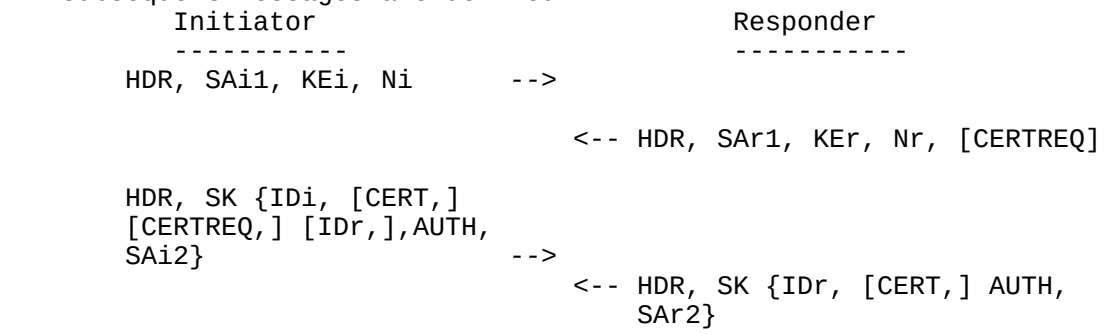
A child SA exchange MAY be initiated by either end of the RKMP_SA after the initial exchanges are completed. All messages in a child SA exchange are cryptographically protected using the cryptographic algorithms and keys negotiated in the the initial exchange.

3. Initial Exchange Details

In the remainder of this document, the notations of the payloads contained in the messages are consistent with what have defined in Section 1.2 of [RFC4306].

The initial exchanges are decrypted as follows:

The payloads included in the first pair of exchanged messages (i.e., the RKMP_SA_INIT exchange) are identical to what have been specified in the IKE_SA_INIT exchange [RFC4306]. During the RKMP_SA_INIT exchange, the two communicating partners needs to identify the cryptographic suite they both support, exchange nonces in order to check each other's aliveness, and exchange their public keys. After the exchange, both partners can use the Diffie-Hellman algorithm to agree upon a shared secret from which all keys for securing subsequent messages are derived.



The second pair of exchanged messages (i.e., the RKMP_SA_AUTH exchange) employ most of the payload specified in the IKE_SA_AUTH exchange. However, the traffic selector payloads in the original IKE_SA_AUTH exchange is removed. The objective of exchanging of traffic selector payloads is to guarantee the consistence of the Security Policy Databases (SPD) on the communicating partners. Therefore, when an IP packet is received by an IPsec subsystem and matches a "protect" selector in its Security Policy Database (SPD), the subsystem will have to protect that packet with IPsec. However, this is not the scenario that RKMP needs to consider. In addition, because RKMP is designed for cryptographic keys for routing protocols instead of IPsec, more values of the protocol ID field in the Security Association payload needs to be defined to represent different routing protocols.

4. Child SA Exchange Details

The Child SA exchange takes advantage of the payloads of the CREATE_CHILD_SA exchange while removing the traffic selector payloads. In addition, in order to support different routing protocols more values of the protocol ID field in the Security Association payload needs to be defined.

```
 Initiator                              Responder
-----------                            -----------
HDR, SK {[N], SA, Ni,
[KEi]}                  -->
                           <--    HDR, SK {SA, Nr, [KEr]}
```

Note that in IPsec the value used to identify a particular SA is referred to as a Security Parameter Index (SPI). However, the values identifying a SA in other routing protocols may be named differently. For example, in RIPv2, OSPFv2 and IS-IS, such values are denoted as key identifiers. RKMP follows IKEV2 and uses SPIs to denote the values identifying SAs in different routing protocols.

5. Interfaces

This section introduces three groups of interfaces: the interface to routing protocols, the interface to RKMP, and the interface to the key table.

The interface to RKMP includes following methods:

RKMP_generateSA: This method is called when a routing protocol expects RKMP to generate a new routing protocol SA and store it into the key table. As parameters, the protocol ID, the addresses of the Interfaces that two routers will be used to exchange messages need to be inputed. RKMP will send the SPI of the SA back to the routing protocol. After getting the SPI, the routing protocol can use it to derive the correspondent key material from the key table.

RKMP_rekeySA: This method can be called when a routing protocol intends to proactively rekey an child SA which is still in its valid period. The protocol ID and the SPI of the SA which intends to be rekeyed are inputted as parameters. If the child SA is found, RKMP will return the SPI of the new generated equivalent SA to the routing protocol. If there is no correspondent child SA being found, RKMP will return zero back.

The interface to the key table includes following methods:

Keytable_getSA: This method is called when a routing protocol intends to get key material to secure a routing message sent to a remote router. As parameters, the protocol ID, the addresses of the Interfaces that two router will be used to exchange messages need to be inputted. (If the SPI of the SA is available, the routing protocol can also input the SPI to indentify the desired SA. It is assumed here that an SA can be uniquely identified by its SPI and the associated routing protocol ID.) The contents of the associated routing protocol SA will be returned.

Keytable_delete: This method is called when a routing protocol intends to delete un-useful child SAs to release occupied resources. The protocol ID and the SPI of the SA to be deleted are inputted as parameters to identify the child SA which will be deleted. If the inputted SPI is zero, all the child SAs used by the routing protocol will be deleted.

Keytable_insertSA: This method is called when RKMP have generated a new routing protocol SA and intends to store it into the key table. If there is already a SA with the identical SPI, the inserting operation will be failed.

Keytable_rekeySA: This method is called when RKMP have generated a equivalent SA and intends to use it take place of the existing one maintained in the key table.

The interface to a routing protocol includes following methods:

RP_revokeSA: This method is called when RKMP deams that the RKMP security association has failed and then discards all state associated with the RKMP SA and any child SAs negotiated using that RKMP SA. After being invoked, the routing protocol will not use existing SAs to secure routing protocols messages.

6. Security Considerations

7. IANA Considerations

The values of the protocol ID fields in the payloads need to be assigned by IANA to present various routing protocols.

8. Acknowledgements

9. References

9.1. Normative References

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

[RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005.

9.2. Informative References

[I-D.hartman-karp-mrkmp] Hartman, S. and D. Zhang, "Multicast Router Key Management Protocol (MRKMP)", draft-hartman-karp-mrkmp-01 (work in progress), March 2011.

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

Authors' Addresses

Dacheng Zhang
Huawei
Beijing
China

Email: zhangdacheng@huawei.com

Sam Hartman
Painless Security

Email: hartmans@painless-security.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 30, 2011

L. Zheng
M. Chen
Huawei Technologies
M. Bhatia
Alcatel-Lucent

June 28, 2011

LDP Hello Cryptographic Authentication
draft-zheng-mpls-ldp-hello-crypto-auth-02.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 how the National Institute of Standards and Technology (NIST) Secure Hash Standard family of algorithms should be used to secure LDP Hello messages.

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 December 30, 2011.

Copyright Notice

Table of Contents

1. Introduction

The Label Distribution Protocol (LDP) [RFC5036] utilizes LDP sessions that run between LDP peers. The peers may be directly connected at the link level or may be remote. A label switching router (LSR) that speaks LDP may be configured with the identity of its peers or may discover them using the LDP Hello message sent encapsulated in UDP that may be 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.

Unlike all other LDP messages, the Hello messages are sent using UDP not TCP. This means that they cannot benefit from the security mechanisms available with TCP. [RFC5036] does not provide any security mechanisms for use with Hello messages except to note that some configuration may help protect against bogus discovery events.

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 is observed and reported as real problem in production networks. Spoofed Hello attack has been identified in [I-D.ietf-karp-routing-tcp-analysis] and need to be addressed.

As described in [RFC5036], the threat of spoofed Basic Hellos can be reduced by accepting Basic Hellos only on interfaces to which LSRs that can be trusted, and ignoring Basic Hellos not addressed to the "all routers on this subnet" multicast group. Spoofing attacks via Extended Hellos are potentially more serious threat. An LSR can reduce the threat of spoofed Extended Hellos by filtering them and accepting only those originating at sources permitted by an access list. However, performing the filtering using access lists requires LSR resource, and the LSR is still vulnerable to the IP source address spoofing.

This document introduces a new Cryptographic Authentication TLV which is used in LDP Hello message as an optional parameter. It enhances the authentication mechanism for LDP by securing the Hello message against spoofing attack, and an LSR can 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 packet, the key is used to generate and verify a HMAC Hash that is stored in the LDP Hello packet. 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

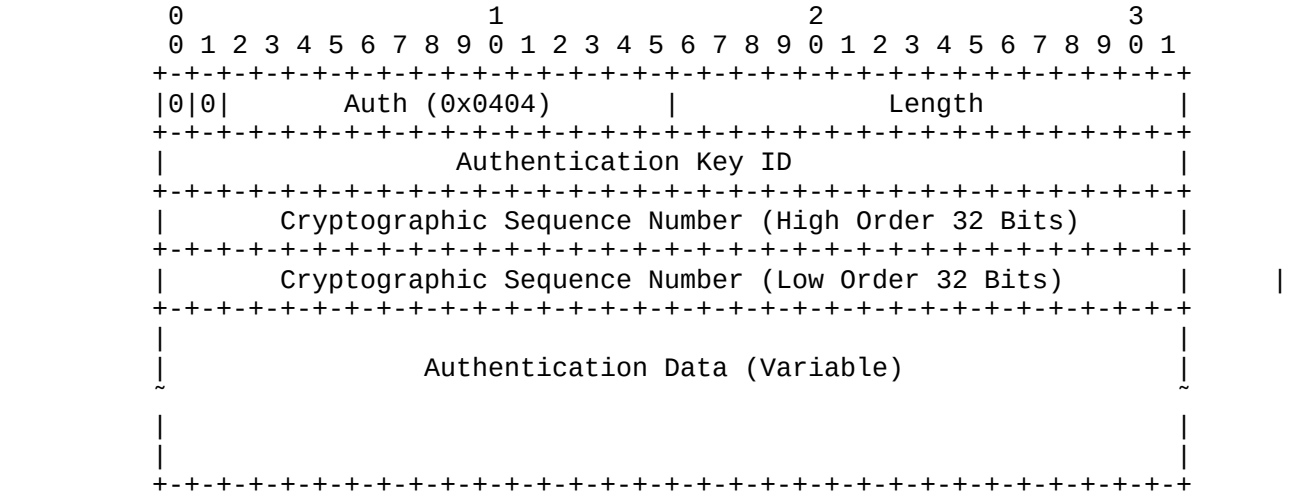
2.1. Optional Parameter for Hello Message

[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. Cryptographic Authentication TLV Encoding



- Type: 0x0404 (TBD by IANA), Cryptographic Authentication

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

- Auth Key ID: 32 bit field that identifies the algorithm and the secret key used to create the message digest carried in LDP payload.

- 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 packet sent by the LDP router. Upon reception, the sequence number MUST be greater than the sequence number in the last LDP packet accepted from the sending LDP neighbor. In case it isnt, the LDP packet is considered a replayed packet and is dropped.

LDP routers implementing this specification SHOULD use available mechanisms to preserve the sequence number's strictly increasing property for the deployed life of the LDP router (including cold restarts). Techniques such as sequence number space partitioning and non-volatile storage preservation can be used but are beyond the scope of this specification.

- 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 used, 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

3. Cryptographic Aspects

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

- H is the specific hashing algorithm specified by Auth Type (e.g. SHA-256).

- K is the Authentication Key for the Hello packet.

- Ko is the cryptographic key used with the hash algorithm.

- B is the block size of H, in octets.

    For SHA-1 and SHA-256: B == 64
    For SHA-384 and SHA-512: B == 128

- L is the length of the hash outputs, in octets.

- XOR is the exclusive-or operation.

- Ipad is the byte 0x36 repeated B times.

- Opad is the byte 0x5c repeated B times.

- Apad is source IP address that the would be used when sending out the LDP packet, repeated L/4 times, where L is the length of the hash, measured in octets.

3.1. Cryptographic Key

As described in [RFC2104], the authentication key K can be of any length up to B. Applications that use keys longer than B bytes will first hash the key using H and then use the resultant L byte string as the actual key to HMAC.

In this application, Ko is always L octets long. If the Authentication Key (K) is L octets long, then Ko is equal to K. If the Authentication Key (K) is more than L octets long, then Ko is set to H(K). If the Authentication Key (K) is less than L octets long, then Ko is set to the Authentication Key (K) with trailing zeros such that Ko is L octets long.

3.2. Hash

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

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

Hello Packet refers to the LDP Hello packet excluding the IP header.

3.3. 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.

4. Processing Hello Message Using Cryptographic Authentication

4.1. Transmission Using Cryptographic Authentication

Prior to transmitting 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 Auth Key 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.

4.2. 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].

If the Auth Key ID field does not match the ID of a configured authentication key, the received packet 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 packet MUST be discarded.

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 Auth Key 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, the received packet MUST be discarded.

5. Security Considerations

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

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.

6. IANA Considerations

IANA maintains a registry of LDP message parameters with a sub-registry to track LDP TLV Types. This document requests IANA to assign a new TLV type as follows for Cryptographic Authenticatio. This document suggests 0x0404 to foster pre-standard implementations.

7. 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, Thomas Nadeau, So Ning, Eric Rosen and Sam Hartman for their valuable comments.

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

8. References

8.1. 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.

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

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

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

8.2. References

8.3. Informative References

[I-D.ietf-karp-routing-tcp-analysis]
Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Security According to KARP Design Guide", draft-ietf-karp-routing-tcp-analysis-00 (work in progress), June 2011.

[RFC2385] Heffernan, A., "Protection of BGP Sessions via the TCP MD5 Signature Option", RFC 2385, August 1998.

[RFC4634] Eastlake, D. and T. Hansen, "US Secure Hash Algorithms (SHA and HMAC-SHA)", RFC 4634, July 2006.

[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.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, June 2010.

Authors' Addresses

Lianshu Zheng
Huawei Technologies
China

Email: verozheng@huawei.com

Mach(Guoyi) Chen
Huawei Technologies
China

Email: mach@huawei.com

Manav Bhatia
Alcatel-Lucent
India

Email: manav.bhatia@alcatel-lucent.com