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:

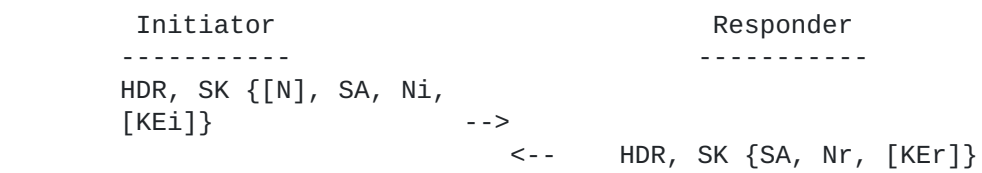
The payloads included in the first pair of exchanged messages (i.e., the RKMP_SA_INIT exchange) are identical to what have been specified in the IKE_SA_INIT exchange [RFC4306]. During the RKMP_SA_INIT exchange, the two communicating partners needs to identify the cryptographic suite they both support, 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.

```
       Initiator                          Responder
       -----------                        -----------
    HDR, SAi1, KEi, Ni      -->

                                <-- HDR, SAr1, KEr, Nr, [CERTREQ]

    HDR, SK {IDi, [CERT,]
    [CERTREQ,] [IDr,],AUTH,
    SAi2}                   -->
                                <-- HDR, SK {IDr, [CERT,] AUTH,
                                    SAr2}
```

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.



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