Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 21, 2011
S. Hartman
Painless Security
D. Zhang
Huawei
October 18, 2010

Multicast Router Key Management Protocol (MRKMP)
draft-hartman-karp-mrkmp-00.txt

Abstract

Several routing protocols engage in one-to-many communication. In order to authenticate these communications using symmetric cryptography, a group key needs to be established. This specification defines a group protocol for establishing and managing such keys.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Many routing protocols such as OSPF and IS-Is use a one-to-many or multicast model of communications. The same message is sent to a number of recipients.

These protocols have cryptographic authentication mechanisms that use a key shared among all members of a communicating group in order to protect messages sent within that group. From a security standpoint, all routers in a group are considered equal. Protecting against a misbehaving router that is part of the group is out of scope for this protocol.

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.

The members of a group elect a Group Controller/Key Server (GCKS). Potentially any member of the group may act as a GCKS. Since protecting against misbehaving routers is out of scope, there is no need to protect against a node that is not currently the GCKS impersonating the GCKS.

To prove membership in the group, a router authenticates using its provisioned credentials to the current GCKS. If successful, the router is given the current key material for the group. Group size is relatively small and need for forced eviction of members is rare. If a GCKS needs to evict a member, then it can simply re-authenticate with the existing members and provide them new key material.

1.1. Terminology

One key terminology question to answer is the definition of group. It appears that as used in this document, the term group corresponds to a routing protocol instance on a single link. However, this needs to be confirmed with TE routing protocols and with PIM. If that works out then a more precise term than group should be used in this document.

1.2. Relationship to IKEv2

IKEv2 [RFC5996]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, 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.

For these reasons, this protocol borrows concepts and payloads from IKEv2 but does not normatively depend on the IKEv2 specification.

1.3. Relationship to GDOI

The IPsec Group Domain of Interpretation (GDOI) [RFC3547] provides a protocol that is structurally very similar to this one. As specified, IKE can be used to provide phase 1 authentication to a GCKS. After that, GDOI provides phase 2 messages to establish key-encryption keys and traffic keys. Key management operations can be accomplished via GDOI messages sent to the group after the phase 2 exchange.

GDOI is defined for IKE not for IKEv2. In addition, GDOI's phase 2 uses its own hashing mechanism and nonce mechanism to provide integrity protection and replay protection. Like IKE, GDOI has significant complexity to support phase 2 identities that are different than the phase 1 identity. GDOI requires a GCKS to have a signature key used to sign GDOI messages when the rekey protocol is used. Since attacks caused by members of the group masquerading as the GCKS are out of scope, this is significant unnecessary complexity in the protocol.

This protocol can be thought of as a simplified GDOI based on IKEv2 rather than IKE. However, integrity and replay mechanisms are taken from IKEv2. Support for phase 2 identities is removed as unneeded complexity. Security for the group key management messages is provided using symmetric primitives rather than asymmetric signatures. Phase 1 authentication will often still involve asymmetric signatures.

2. Overview

MRKMP is composed of several parts. There is an initial exchange used to establish a shared key with a GCKS and authenticate the identities of both parties. Unicast key management exchanges provide the ability to join a group or request updates to the group; group joins can also be combined with the initial exchange. There is an election protocol used by routers to determine which router will act as the GCKS; this protocol is not integrity protected, but a GCKS confirms its role when a member uses the unicast exchange to join the group. Finally, a GCKS uses multicast exchanges to update parameters of the group. This section briefly describes each of these parts of MRKMP. The later sections in the document describe the details of the protocols.

2.1. Types of Keys

MRKMP manipulates several different types of symmetric keys:

preshared: Preshared keys are one mechanism for authenticating one router to another during the initial exchange. These keys are configured by some mechanism such as manual configuration or a management application outside of the scope of MRKMP. A single preshared key can be used for all members of a group. Alternatively each pair of routers can have a different preshared key.

peer key management key: Routers share a key with the GCKS that is a result of the mrkmp_init exchange.

KEK: A Key encryption Key (KEK) is a key used to encrypt group key management messages to the current members of a group. A KEK is learned as the product of establishing an MRKMP association or through a group key management message encrypted in a previous KEK. A KEK has an explicit expiration but may also be retired by a message encrypted in the KEK sent by the GCKS.

protocol master key: A protocol master key is the key exported by MRKMP for use by a routing protocol such as OSPF or IS-IS. The Protocol master key is the key that would be manually configured if a routing protocol is used without key management.

transport key: The transport key is the key used to integrity protect routing messages in a protocol such as IS-IS or OSPF. In today's routing protocol cryptographic authentication mechanisms the transport key is the same as the protocol master key. A disadvantage of this approach is that replay prevention is challenging with this architecture. Ideally some key derivation step would be used to establish a fresh transport key among all the participants in the group.

2.1.1. Key Encryption Key

When a router wishes to join a group, the router performs the mrkmp_init and mrkmp_auth exchange with a GCKS. During this process the router can establish an association with a specific group. Part of that association will be delivery of a KEK and associated parameters.

Group key management messages are sent to a group address not unicast to an individual peer. The group key management messages are protected using the KEK. The group key management messages need to provide both integrity and confidentiality protection using the KEK.

As part of establishing the association, the router joining the group is given an expiration time for the KEK. A group key management message may establish a new KEK with new parameters.

From time to time, a GCKS may wish to either force early expiration of a KEK or allow a KEK to expire. Protocol master keys are permitted to be valid for somewhat longer than the KEK that created them so as to avoid disrupting routing when this happens. When a KEK is retired or expires without being replaced by a new KEK announced in the old KEK, group members need to perform a new initial exchange to the GCKS. This is useful for example if a router is no longer authorized to be part of the group.

Other mechanisms such as LKH (section 5.4 [RFC2627]) could be used to permit removal of a group member while avoiding new initial authentications. However these mechanisms come at a complexity cost that is not justified for a small number of routers participating in a single multicast link.

2.1.2. Protocol Keys

Current routing protocols directly use the protocol master key to integrity protect messages. One advantage for this approach is that the initial hello messages used for discovery and capability exchange can be protected using the same mechanism as other messages. Typically a sequence number is used for replay detection. Without changing the key, the existing protocols are vulnerable to a number of serious denial of service attacks from replays.

The MRKMP can solve this replay problem by changing the protocol master key whenever a peer is about to exhaust its sequence number space or whenever a peer loses information about what sequence numbers it used. This could potentially involve changing the protocol master key whenever a router reboots that was part of the group using the current protocol master key. Since key changes will not disrupt active adjacencies and can be accomplished relatively quickly, this is not expected to be a huge problem. Note that after one key change, others routers can boot without causing additional key changes; a flurry of key changes would not be required if several routers reboot near each other.

Another approach would be to separate the protocol master key from the transport keys. For example the transport key used by a given peer could be a fresh key derived from the protocol master key and nonces announced by that peer. Some mechanism would need to make sure that the peer's announcement of its nonce was fresh; this mechanism would almost certainly involve some form of interaction with the router wishing to guarantee freshness. There are two key advantages of this separation between transport keys and protocol master keys. The first is that the interaction between the MRKMP and routing protocol can be simplified significantly. The second is that even when manually configured protocol master keys are used, replay and adequate DOS protection can be achieved.

2.2. GCKS Election

Before a MRKMP system actually starts working, the routers in the multicast group need to select a GCKS so that they can obtain cryptographic keys to secure subsequent exchanges of routing information. MRKMP specifies an election protocol that dynamically assigns the responsibility of key management to one of the group members. Note that there are already announcer-electing mechanisms provided in some routing protocols (e.g., OSPF and IS-IS). However, much involvement between a MRKMP system and a routing protocol implementation will be introduced if the MRKMP system reuses the announcer-electing mechanism for the election of the GCKS. The state machine of the routing protocol also has to be modified. For instance, in OSPF, after a DR has been elected, routers need to halt their OSPF executions, and carry out the initial exchange to authenticate the DR and collect the keys for subsequent communications. After this step, the routers need to re-start their OSPF state machines so as to exchange routing information. As a consequence of such cases, an individual GCKS electing solution within MRKMP is preferable.

Each router has a GCKS priority. Higher priorities are more preferred GCKSes. As discussed in Section 8, the routing protocol can influence the GCKS election protocol by manipulating the priority so that it is likely that the same router will be the announcer for the routing protocol and the GCKS. Even if two different routers are elected as the announcer and GCKS, then the routing protocol and MRKMP will function correctly.

2.3. Initial Exchange

The initial exchange is based on IKEv2's IKE_SA_INIT and IKE_SA_AUTH exchanges. During this exchange, an initiating router attempts to authenticate to the router it believes is a GCKS for a group that the initiating router wants to join. Messages are unicast from the initiator to the responding GCKS. Unicast MRKMP P messages form a request/response protocol; the party sending the messages is responsible for retransmissions.

The initial exchange provides 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 integrity protect and encrypt key management messages. While routing security does not typically require confidentiality, the key management protocol does because keys are exchanged and these must be protected.

Then the identities of each party are cryptographically verified. This can be done using a preshared key or symmetric keys. Other mechanisms may be added as a future extension.

The authentication exchange also provides an opportunity to join a group as part of the initial exchange. In the typical case, a router can obtain the needed key material for a group in two round-trips.

2.4. Group Join Exchange

The primary purpose of the unicast MRKMP messages is to get an initiator the information it needs to join a group and participate in a routing protocol. The initiator indicates what group it wants to join. XXX we need to discuss group naming--if MRKMP is limited to a subnet this may be as simple as saying that initiator wants to join the OSPF group or the IS-IS group.

The responder performs several checks. First, the responder confirms that the responder is currently acting as GCKS for the group in question. Then, the responder confirms that the initiator is permitted to join the group. If these checks pass, then the responder provides a key download payload to the initiator encrypted in the peer key management key. As discussed in Section 2.1.2, the GCKS MUST change the protocol master key if a router was part of the group under the current protocol master key and reboots. In this case, the GCKS SHOULD provide the new and old protocol master key to the initiator, setting the validity times for the old key to permit reception but not transmission. The GCKS MUST use the mechanism in the next section to flood the new key to the rest of the group.

A group association created by this exchange may last beyond the unicast MRKMP association used to create it. Once membership in a group is established, resources are not required to maintain the unicast association with the GCKS.

A member of a group can also use the unicast exchange to request a GCKS to change the protocol master key because that group has exhausted its available sequence space. For protocols where the protocol master key is the same as the transport key, it is critical that no two messages be sent by the same router with the same sequence number and protocol master key. The sequence number space is finite. So if a router is running low on available sequence space it needs to request a new protocol master key be generated.

2.5. Group Key Management

The GCKS shares a KEK with all members of a group. The GCKS can send a multicast message to the group to update the set of protocol master keys, update the KEK, or retire the KEK and request new group join exchanges.

Typically the protocol master key is changed only when needed to provide replay protection or when the KEK changes. The KEK changes whenever a new GCKS is elected or whenever it is administratively desirable to change the keys. For example if an employee leaves an organization it might be desirable to change the KEKs. A KEK is retired whenever forward security is desired: whenever the authorization of who is permitted to be in a group changes and the GCKS needs to make sure that the router is no longer participating. Most authorization changes such as removing a router from service do not require forward security in practical deployments.

3. GCKS Election

The GCKS election process selects a single router on a link to act as GCKS for a group.Similar with other popular announcer electing mechanisms (e.g., VRRP, HSRP), in MRKMP, only GCKSes use multicast to periodically send Advertisement messages. Such advertisements can be used as heart beat packets to indicate the aliveness of GCKSes. In addition, a state machine with three states (Initial, GCKS, and Member) is specified for GCKS election. When a router is initially connected to a multicast network, its state is set as Initial. The router then sends a multicast initial advertisement, if a GCKS is working on the network, it will reply the router with an advertisement using unicast. After receiving the advertisement from the GCKS, the router will try to register with the GCKS using the initial exchange, and then the state of the router is transferred to Member. Note that when the router receives the advertisement it does not have the traffic distributed in the group. Thus, the integrity of the unicast advertisement does not have to be protected. After a certain period, if the router still does not receive any advertisement from a GCKS or other group members, the router then believe there is no other group member on the network and set its state as GCKS. If during the period the router does not receive any advertisement from a GCKS but receives advertisements from other routers on the network, router believes that the group is involved in a GCKS election process. Apart from the initialization of a multicast network, the fail-over of a GCKS can also trigger an election process. For instance, if a router does not receive the heart beat advertisement for a certain period, it will transfer its state to Initial and try to elect a new one. In a GCKS electing process, a router has to stay in the Initial state until a new GCKS is allocated. Particularly, the router first sends its initial advertisement with its priority and waits for a certain period. During the period, if a router receives an initial advertisement which consists of a lower priority, the router then sends the advertisement again with a limited rate. After period, if the router does not find any router with a higher priority, it announces itself as the GCKS. If two routers have the same priority, the one with the lowest IP source address used for messages on the link will be the GCKS. After a router transfer its state to GCKS, it will reply to the initial advertisements from other routers with GCKS advertisements, even when the initial advertisements consist of properties priorities than its priority. This approach guarantees that a GCKS will not be changed frequently after it has been elected. After receiving the GCKS advertisement of the new elected GCKS, other routers transfer their states to Member. However, if a GCKS G1 receives a GCKS advertisement from another router G2 and G2 is a more preferred GCKS, G1 follows the procedure in Section 3.2.

If a node in state member fails to perform an initial exchange with the router it believes to be GCKS, it resets its state to initial but ignores advertisements from that router. This way an attacker cannot disrupt communications indefinitely by masquerading as a GCKS.

If a node transitions to GCKS state, it performs the procedure in Section 3.1.

3.1. A new GCKS is Elected

3.2. Merging Partitioned Networks

Whenever a GCKS finds that a more preferred router is also acting as a GCKS for the same group, then the group is partitioned. Typically if there is already an active GCKS for a group, even if a more preferred GCKS joins, the GCKS will not change. Two situations can result in multiple GCKSes active for a group. The first is that members of the group do not share common authentication credentials. The second is that the group was previously partitioned so that some nodes could not see election messages from other nodes. After the problem resulting in the partition is fixed, then both active GCKSes will see each others election announcements. The group needs to merge.

The less preferred GCKS performs a unicast mrkmp_merge_sa unicast key management message to the more preferred GCKS. In this message the less preferred GCKS includes its key download payload, so the more preferred GCKS learns the protocol master keys of the less preferred GCKS.

The more preferred GCKS generates a new key download payload including a KEK and the union of all the protocol master keys. The GCKS SHOULD mark the existing protocol master keys as expiring for usage in transmitted packets in a relatively short time. The GCKS SHOULD introduce a new protocol master key. This key download payload is returned to the less preferred GCKS and is sent out in the current KEK using a group key management message.

The less preferred GCKS sends the received key download payload encrypted in its existing KEK. XXX how many retransmits. After all retransmissions of this payload the less preferred GCKS sets its state to member.

As a result of this procedure, members learn the protocol master keys of both GCKSes and converge on a single KEK and GCKS. Changing the protocol master keys during a merge is important for protocols that use the protocol master key as a transport key. The new GCKS does not know which routers have joined the group with the other GCKS.

Therefore, it could not correctly detect one of these routers rebooting and change the protocol master key at that point. If the key is changed as part of the merge, replays are handled.

4. Key Download Payload

What all is actually in the message you get at the end of phase 2 and that is sent out periodically during group key management

For the KEK, this needs to include the key itself, the algorithm (presumably drawn from the IKEv2 symmetric algorithms), key ID, group ID and the four lifetimes.

The protocol master keys include the key, an algorithm ID, the key ID and the four lifetimes.

By four lifetimes we mean receive start, send start, send end and receive end. It’s important that a key can be flooded out to all potential receivers before it is used for sending.

5. Initial Exchange Details

6. Group Management Unicast Exchanges

6.1. Group Join Exchange

If a router receives a group join exchange for a group for which it is not the GCKS, it MUST return a notification. If it knows the GCKS for the group then it returns MRKMP_WRONG_GCKS including the address of the GCKS in the notification payload. The initiator tries the group join exchange (probably with a new initial exchange) with the indicated router. If the responder does not know the GCKS for the group, either because it is not a member of the group or because its GCKS election state is initial, it returns the MRKMP_GCKS_UNKNOWN notification. If the responder is not trying to be a member of this group or has seen a more preferred GCKS advertisement in the election process then the potential_candidate bit is clear, otherwise it is set. The initiator sets its GCKS election state to initial when receiving this notification. If the potential candidate bit is set in the notification then the initiator will accept GCKS election advertisements from the responder. If the potential candidate bit is clear, then the initiator will discard GCKS election advertisements from the responder until BLACKLIST_TIMEOUT seconds have elapsed or until the initiator successfully joins the group.

7. Group Key Management Operation

Group key management messages are multicast from the GCKS to the group. The message contains the key identifier of a KEK, as well as encrypted/integrity-protected payloads. Inside the encrypted/integrity-protected payloads is a monotonically increasing sequence number, and payloads specific to the message being sent. Group members MUST ignore a message with a sequence number that is the same or less than the sequence number of the most recent message they have received.

7.1. General operation

Periodically the GCKS will send out an update message encrypted in the current KEK including the current group key download payload and parameters. If a new KEK is about to be valid for receiving messages, this is included. Any protocol master keys that are valid for sending or receiving SHOULD be included.

If a previous KEK is still valid for sending, then an update message is sent encrypted in the old KEK. This message MUST include the new KEK. This message SHOULD include the protocol master keys.

7.2. Out of Sequence Space

7.3. Changing the Active GCKS

8. Interface to Routing Protocol

This section describes signaling between MRKMP and the routing protocol. The primary communication between these protocols is that MRKMP populates rows in the key table making protocol master keys available to the routing protocol. However additional signaling is also required from the routing protocol to MRKMP. This section discusses that signaling. All required communication from MRKMP to the routing protocol can be accomplished by manipulating the key table. However an implementation MAY wish to signal MRKMP failures to the routing protocol in order to provide consistent management feedback.

8.1. Joining a Group

When a routing protocol instance wishes to begin communicating on a multicast group, it signals a group join event to MRKMP. This event includes the identity of the group as well as this router’s priority for being a GCKS for the group. When MRKMP receives this event, it starts MRKMP for this group and attempts to find a GCKS.

8.2. Priority Adjustment

It is desirable that the GCKS function track the functions within a routing protocol. For example for protocols such as OSPF that designate a router on a link to manage adjacencies for that link, it would be desirable for the GCKS role to be assigned to that router. The routing protocol provides a priority input to the GCKS election process. Initially the routing protocol should map any priority mechanism within the routing protocol to the GCKS election procedure so that routers favored as announcer for a link will also be favored as a GCKS.

However, the routing protocol SHOULD also dynamically manipulate the GCKS election priority based on what happens within the routing protocol. The router actually elected as the announcer SHOULD have a GCKS election priority higher than any other group member. Typically, by the time the routing protocol is able to elect an announcer, a GCKS will already be chosen. However, if a GCKS election is triggered when the routing protocol is already operational, then the election can choose the routing protocol’s announcer.

8.3. Leaving a Group

If a routing protocol terminates on an interface, MRKMP needs to be notified that group is no longer joined. MRKMP MUST stop participating in the GCKS election process, stop monitoring for key management messages and if the current router is a GCKS, stop acting in that role.

9. Security Considerations

An attacker who can suppress packets sent to the group can create a denial of service condition. One attack is to suppress GCKS election packets and cause two routers to believe they are both the GCKS for the group. If the least preferred router never hears the GCKS advertisement from the more preferred router, then the group will remain partitioned. Such an attacker is likely to be able to mount more direct denial of service, for example suppressing the actual routing protocol packets.

The security of the system as a whole depends on the pair-wise security between the router currently in the GCKS role and the other routers in the group. Since any router can potentially act as GCKS, the pair-wise security between all members of the group is critical to the security of the system. In practical deployments, information used by the router acting as GCKS to authorize a member joining the group will be configured by some management application. In these deployments, the security of the system depends on the management application correctly maintaining this information on all routers potentially in the group.

10. Acknowledgements

This draft is the result of a design discussion held after the IETF 78 KARMP meeting. The authors, David Mcgrew, Brian Weis and Gregory Lebovitz all contributed to the design meeting.

11. Informative References

[RFC2627] Wallner, D., Harder, E., and R. Agee, "Key Management for Multicast: Issues and Architectures", RFC 2627, June 1999.

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

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

Authors' Addresses

Sam Hartman
Painless Security

Email: hartmans-ietf@mit.edu

Dacheng Zhang
Huawei

Email: zhangdacheng@huawei.com