PF_KEY Extension as an Interface between Mobile IPv6 and IPsec/IKE
draft-sugimoto-mip6-pfkey-migrate-00

Status of this Memo

This document is an Internet-Draft and is subject to all provisions of Section 3 of RFC 3667. By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she become aware will be disclosed, in accordance with RFC 3668.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on August 18, 2005.

Copyright Notice

Copyright (C) The Internet Society (2005).

Abstract

This document describes need for interface between Mobile IPv6 and IPsec/IKE and show how two protocols can interwork each other. We propose a mechanism to realize this interaction by extending PF_KEY framework. Proposed mechanism makes it possible for Mobile IPv6 to inform IPsec/IKE about the movement. Receiving the message, IPsec
components can take necessary actions to update corresponding entry in SPD and SADB. In addition, IKE can use the notification for keeping its IKE connection alive.

Table of Contents

1. Introduction ............................................. 3
2. Terminology ............................................. 4
3. Needs for Interactions between Mobile IPv6 and IPsec/IKE ... 5
4. Requirements ............................................. 7
5. PF_KEY Extension for Mobile IPv6 .......................... 8
   5.1 Overview ............................................. 8
   5.2 Message Sequence ................................... 9
   5.3 Issuing PF_KEY MIGRATE Message ..................... 10
   5.4 Processing PF_KEY MIGRATE Message ................. 11
   5.5 Applicability of PF_KEY MIGRATE to Other Systems .... 12
   5.6 Limitation of PF_KEY MIGRATE ....................... 12
6. Necessary Modifications to Mobile IPv6 and IPsec/IKE .... 13
7. Security Considerations ................................... 14
8. Conclusion ................................................. 15
9. References ............................................... 15
   Authors’ Addresses ...................................... 16
A. PF_KEY MIGRATE Message Format .......................... 17
B. Acknowledgements ....................................... 20
   Intellectual Property and Copyright Statements ........... 21
1. Introduction

In Mobile IPv6 [RFC3775], the MN and HA may use IPsec tunnel to protect mobility signals and payload traffic. From security point of view, it is preferable to protect mobility signals that are transmitted on the path between MN and HA. In addition, mobile user may want to have payload traffic protected by IPsec tunnel. Hence, leveraging IPsec tunnel is essential in Mobile IPv6 operation. Since the MN may change its attachment point to the Internet, it is necessary to update endpoint address of the IPsec tunnel. This indicates that corresponding entry in IPsec database (SPD and SADB) should be updated when MN performs movement. In addition, IKE requires treatment to keep its IKE connection alive in Mobile IPv6 environment.

This document describes need for interface between Mobile IPv6 and IPsec/IKE and show how two protocols can interwork each other. We propose a mechanism to realize this interaction by extending PF_KEY framework [RFC2367]. Proposed mechanism makes it possible for Mobile IPv6 to inform IPsec/IKE about the movement. Receiving the message, IPsec components can take necessary actions to update corresponding entry in SPD and SADB. In addition, IKE can use the notification for keeping its IKE connection alive. Proposed mechanism has been implemented on both Linux and BSD platform and confirmed to work well with existing Mobile IPv6 and IPsec/IKE stack.
2. Terminology

There is no term newly introduced in this document. Definitions of the terms relevant to Mobile IPv6 can be found in [RFC3775] and [RFC3776]. Definitions of the terms relevant to IPsec can be found in [RFC2401] and [I-D.ietf-ipsec-rfc2401bis]. Some clarifications are provided below for the terms that are frequently used in this document.

Care-of address (CoA) - Topologically correct IPv6 address assigned for MN. MN registers its primary CoA to the HA, which is called home registration. CoA should be an endpoint address of the tunnel established between the MN and HA. The tunnel could be either IP-in-IP tunnel or IPsec tunnel.

Movement - An event in which MN moves from one subnet to another changing its primary CoA. In Mobile IPv6, movement can be categorized into following three types: Home-to-Foreign, Foreign-to-Foreign, and Foreign-to-Home. In terms of necessary actions taken upon movement, each case should be differentiated.

Return Routability procedure - A mechanism for authorizing Binding Update (BU) message sent by the MN to CN. There are four Mobility Header (MH) messages involved in the procedure: Home Test Init (HoTI), Home Test (HoT), Care-of Test Init (CoTI), and Care-of Test (CoT). In order to minimize a chance for malicious node to eavesdrop contents of the Home Test, it is strongly recommended to protect HoTI and HoT with IPsec tunnel.
3. Needs for Interactions between Mobile IPv6 and IPsec/IKE

First of all we try to figure out needs for interactions between Mobile IPv6 and IPsec/IKE. Ideally, there should be no interactions needed. However, throughout prototyping activity and operation of Mobile IPv6, we have identified following needs.

MN may change its attachment point to the Internet whenever it performs movement. Consequently, its local address of tunnel established between its HA should be updated. Since CoA is used as an endpoint for the tunnel, both MN and HA should take necessary update. On MN, local address of the tunnel should be updated with newly assigned CoA. On HA, remote address of the tunnel should be updated with newly registered CoA. As for normal IP-in-IP tunnel, Mobile IPv6 should be able to update the endpoint address easily. However, Mobile IPv6 may not have full control on updating endpoint of IPsec tunnel since it requires direct access to the IPsec database. This indicates that Mobile IPv6 should request IPsec for updating specific entry in SADB.

In practical scenario, key management protocol such as IKE would be used to establish security association between the MN and HA. IKE daemon should always be aware of up-to-date SPD in order to perform key negotiations and subsequent rekeying. Since IKE runs according to the SPD, tunnel information (endpoint addresses) should be somehow included in security policy entry. Although specific data structure of a security policy is implementation specific, most of the existing IPsec implementations conceptually have such information in security policy. Considering the fact that tunnel endpoint address could be dynamically updated, corresponding entry in SPD should also be updated.

Mobile IPv6 specifies a flag named Key Management Mobility Capability bit (K-bit) in Binding Update and Binding Acknowledgement message (see section 10.3.1 of [RFC3775]), which indicates an ability of IKE to survive movement. When both MN and HA agree to use this functionality, the IKE daemons dynamically update its IKE connection when the MN performs movement. In order to realize this, IKE daemon should be notified by Mobile IPv6 and necessary information to migrate the IKE connection should be provided.

Mobile IPv6 may need to make an access to SPD not only for updating endpoint address but also for deletion/insertion of specific security policy entry. When the MN performs Foreign-to-Home movement, IPsec tunnel established between the MN and HA should be destroyed, which means that the SPD entry should have no effect any more. On the other hand, when the MN performs Home-to-Foreign movement, the IPsec tunnel should newly be created. Hence security policy entries that
are associated with tunnel mode security association may dynamically be added/removed in along with MN’s movement.

It should be noted that NEMO Basic Support [RFC3963] may have similar requirements for the Mobile Router (MR) and MR’s HA (MRHA). In NEMO, MR works just as same as MN registering its location information to the MRHA and establishes tunnel (IP-in-IP or IPsec tunnel). If IPsec tunnel is established between MR and MRHA, the MR serves as a Security Gateway (SGW) for the nodes inside the mobile network. MR is responsible for handling its tunnel endpoint properly.
4. Requirements

Given the needs for interface between Mobile IPv6 and IPsec/IKE, there should be minimum interface between the two protocols. Followings are the requirements for the interface from software engineering point of view.

- Necessary modifications to the existing software, namely Mobile IPv6 and IPsec/IKE should be kept minimum in order to realize proposed mechanism.
- Proposed mechanism should not be platform dependent. The mechanism should be based on technology which is commonly available on various platform. This seems to be essential for achieving high portability of the implementation which supports proposed mechanism.
5. PF_KEY Extension for Mobile IPv6

In order to fulfil the needs and requirements described in Section 3 and Section 4 we propose to extend PF_KEY framework so that Mobile IPv6 and IPsec/IKE could interact each other. The extension is primarily for migrating endpoint address of IPsec tunnel from one to another, which results in updating IPsec database. A new PF_KEY message named MIGRATE was introduced for the mechanism.

5.1 Overview

The figure below illustrates how Mobile IPv6 and IPsec/IKE components interact each other with PF_KEY MIGRATE message in dynamic keying scenario. On left top, there is a Mobile IPv6 entity. It may be possible that Mobile IPv6 component is completely implemented inside the kernel. In any case, Mobile IPv6 should be the one who issues the MIGRATE message. On right top, there is an IKE daemon who is responsible for establishing security associations required for Mobile IPv6 operation. In manual keying scenario, the difference is only that there is no IKE daemon running on the system.

+-------------+           +------------+
|             |           |            |
| Mobile IPv6 |           | IKE Daemon |
|             |           |            |
+-------------+           +------------+

1. PF_KEY MIGRATE + A 4. Update SPD & SADB

+-----------+ +-----------+
| |         | |
Userland V | |
[PF_KEY Socket]-------------------------
Kernel     |
+-----------+ +-----------+

2. Update SPD 3. Update SADB

The primary role of PF_KEY MIGRATE message is to migrate endpoint address of tunnel mode SA requesting IPsec to update its database (SPD and SADB). In addition, the message can be used by IKE to enhance its mobility capability. If the PF_KEY MIGRATE message is properly processed by the kernel, it will be sent to all open socket as normal PF_KEY messages. A sequence of processing MIGRATE message
is done in following steps:

- Mobile IPv6 issues a PF_KEY MIGRATE message to the PF_KEY socket.
- Operating system (kernel) validates the message and checks if corresponding security policy entry exists in SPD.
- If the message is confirmed to be valid, the target security policy entry is updated according to the MIGRATE message. If there is any target security association found that are also target of the update, those should also be updated.
- After the MIGRATE message is successfully processed inside the kernel, it will be sent to all open PF_KEY socket.
- IKE daemon receives the MIGRATE message from PF_KEY socket and updates its SPD and SADB. IKE daemon may also update its IKE connection to keep it alive.

Note that the way IKE maintains its local copy of SPD is implementation specific issue since there is no standard interface to access SPD. Some IKE implementation may continuously monitor SPD inside the kernel. Some IKE implementation may expect notification from the kernel when SPD is updated. In either way, the proposed mechanism gives a chance for IKE to keep its SPD up-to-date which is significant in Mobile IPv6 operation.

5.2 Message Sequence

Next, we will see how migration takes place in along with home registration. The figure below shows sequence of mobility signals and PF_KEY MIGRATE messages while the MN roams around subnets. It is assumed that in initial state the tunnel endpoint address for given MN is set as its home address. In initial home registration, the MN and HA migrate the tunnel endpoint address from HoA to CoA1. It should be noted that no migration takes place when the MN performs re-registration since the care-of address remains same. Accordingly, the MN performs movement and changes its primary care-of address from CoA1 to CoA2. PF_KEY MIGRATE message is issued on both MN and HA. When the MN returns home, migration takes place updating the endpoint address with MN’s home address.
5.3 Issuing PF_KEY MIGRATE Message

Mobile IPv6 entity (MN or HA) triggers the migration by sending PF_KEY MIGRATE message to the PF_KEY socket. Conceptually, the PF_KEY MIGRATE message should contain following information:

- Selector information
  * source address/port
  * destination address/port
  * upper layer protocol (i.e. Mobility Header)
  * direction (inbound/outbound)

- Old SA information
  * old source address of IPsec tunnel
  * old destination address of IPsec tunnel
  * IPsec protocol (ESP/AH)
  * mode (Tunnel)

- New SA information
  * new source address of IPsec tunnel
  * new destination address of IPsec tunnel
Selector information is required for specifying target security policy entry to be updated. It should contain information required for selector as specified in [RFC2401]. Plus, direction of the security policy (inbound/outbound) should be provided. Old SA information is used to specify target security association to be updated. Source and destination address of the target entry should be overwritten with the ones included in new SA information. Note that IPsec protocol and mode are not updated by the PF_KEY MIGRATE message.

The PF_KEY MIGRATE message should be formed based on security policy configuration and binding record. Selector information and some parts of the SA information (IPsec protocol and mode) should be taken from policy configuration. The rest of the information should be taken from sequential binding information. For example, in the case where the MN updates its inbound security policy and corresponding tunnel mode security association, the old source address of the IPsec tunnel should be set as its previous CoA, and the new source address of the IPsec tunnel should be set as its current CoA. Hence, MN should sequentially keep track of its CoA record. Such information shall be stored in binding update list entry. For the same reason, HA should keep track of previous CoA of MN. Such information shall be stored in binding cache entry.

Additionally, a piece of information which indicates mobility capability of IKE (K-bit) should be provided by any means. This makes it possible for IKE to see if there is a need to update its IKE endpoint in accordance with PF_KEY MIGRATE message.

Detailed message format of PF_KEY MIGRATE is provided in Appendix A.

5.4 Processing PF_KEY MIGRATE Message

Since a PF_KEY MIGRATE message is applied to single security policy entry, kernel should first check validity of the message. If the message is invalid, EINVAL error MUST be returned as a return value for the write() operation made to the PF_KEY socket. After the validation, kernel checks if target security policy entry really exists in SPD. If there is no entry found, ESRCH error MUST be returned. In any error case, the message must not have any effect on SPD and SADB and migration becomes incomplete.

With respect to the behavior of normal process (including IKE daemon) who receives PF_KEY MIGRATE message from PF_KEY socket, it SHOULD first check if the message does not include error information. If
there is any error indicated, the process MUST silently discard the PF_KEY MIGRATE message. Otherwise, processing of the message may continue.

5.5 Applicability of PF_KEY MIGRATE to Other Systems

It should be noted that PF_KEY MIGRATE is also applicable to other systems than Mobile IPv6. For example, it can be used in a scenario where IPsec/IKE enabled node establishes tunnel mode security association with its Security Gateway while it roams around the network (aka "road warrior"). Security policy is set as such that all traffic should bi-directionally go through the IPsec tunnel. In such case, migration of the tunnel endpoint address can be handled by PF_KEY MIGRATE message. Each time the node changes its attachment point to the Internet, PF_KEY MIGRATE should be issued to the system. Consequently, the IPsec database (SPD and SADB) shall be properly updated.

It is also essential to keep design of the mechanism protocol independent. More specifically, PF_KEY MIGRATE should be able to work on both IPv4 and IPv6. In order to achieve this, IP addresses to be stored in selector and security association information should be handled in a protocol-independent manner.

5.6 Limitation of PF_KEY MIGRATE

Currently, Security Parameter Index (SPI) is not included in the old SA information to specify target security association entry. This helps to lessen operational burden of Mobile IPv6. However, this simplification can produce ambiguity in searching for the target security association entry. Further considerations and improvements needed.

It should be noted that delivery of PF_KEY MIGRATE message cannot be guaranteed, which is common to other PF_KEY messages. It may be possible that the MIGRATE message is lost. In such case, there will be inconsistency between the binding record managed by Mobile IPv6 and IPsec database inside the kernel. Once a PF_KEY MIGRATE message is lost, it would not be possible for the receiver to process subsequent MIGRATE message properly. Initialization of Mobile IPv6 stack and IPsec database may be needed for recovery. Further improvements should be made.
6. Necessary Modifications to Mobile IPv6 and IPsec/IKE

In order to realize the proposed mechanism, there are some necessary modifications to Mobile IPv6 and IPsec/IKE. Following are the summary of necessary modifications, which could be of interest to implementors of Mobile IPv6/IPsec/IKE.

- Modifications to Mobile IPv6:
  - Mobile IPv6 makes an access to PF_KEY socket. In particular, Mobile IPv6 should have privilege to write data to PF_KEY socket.
  - Issue PF_KEY MIGRATE message. In order to form the MIGRATE message, it is required that Mobile IPv6 should be aware of its IPsec (security policy) configuration and precise binding record.

- Modifications to IPsec:
  - Processing of PF_KEY MIGRATE message. Kernel should be able to process the PF_KEY MIGRATE sent by Mobile IPv6. Unless the message is invalid, it should be sent to all open PF_KEY socket.

- Modifications to IKE:
  - Processing of PF_KEY MIGRATE message. IKE may update its local copy of IPsec database (SPD and SADB) in accordance with received PF_KEY MIGRATE message. In addition, it may update its IKE connection with new endpoint address indicated by the PF_KEY MIGRATE message.
7. Security Considerations

There is no security considerations newly raised by the mechanism proposed in this document.
8. Conclusion

- There are needs for Mobile IPv6 and IPsec/IKE to interact each other to provide full support of IPsec security functions.
- An extension to PF_KEY framework (PF_KEY MIGRATE message) was proposed, which makes it possible for the IPsec/IKE to migrate endpoint address of the IPsec tunnel from one to another.
- PF_KEY MIGRATE message also makes it possible for IKE to survive movements by updating its IKE connection.
- In order for the IKE to perform key negotiations and rekeying, effort should be made to keep its SPD up-to-date.
- The proposed mechanism was implemented on both Linux and BSD platform and confirmed to be working well.
- Currently, large portion of the proposed mechanism is implementation dependent due to lack of standard interface to access SPD (PF_POLICY?).

9. References

[I-D.ietf-ipsec-rfc2401bis]


 RFC 2401, November 1998.

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


Authors' Addresses

Shinta Sugimoto  
Ericsson Research Japan  
Koraku Mori Building  
1-4-14, Koraku, Bunkyo-ku  
Tokyo  112-0004  
Japan  

Phone: +81 3 3830 2241  
Email: shinta.sugimoto@ericsson.com

Francis Dupont  
Point6  
c/o GET/ENST Bretagne  
2 rue de la Chataigneraie  
CS 17607  
35576 Cesson-Sevigne Cedex  
France  

Fax: +33 2 99 12 70 30  
Email: Francis.Dupont@enst-bretagne.fr
Appendix A. PF_KEY MIGRATE Message Format

The figure below shows the message format of PF_KEY MIGRATION. The message consists of 6 parts (boundary of each part is marked with ">"). The message starts with PF_KEY base message header followed by two address extensions. A pair of address extensions hold source and destination address of the selector. Rest of the message are specific to IPsec implementation of BSD. sadb_x_policy{} structure holds additional information of security policy. The last part of the message is a pair of sadb_x_ipsecrequest{} structures that hold old and new SA information.
Following is a structure of PF_KEY base message header specified in [RFC2367]. A new message type for PF_KEY MIGRATE (i.e., SADB_X_MIGRATE) should be specified in member sadb_msg_type.

```c
struct sadb_msg {
    uint8_t sadb_msg_version;
    uint8_t sadb_msg_type;
    uint8_t sadb_msg_errno;
    uint8_t sadb_msg_satype;
    uint16_t sadb_msg_len;
    uint16_t sadb_msg_reserved;
    uint32_t sadb_msg_seq;
    uint32_t sadb_msg_pid;
};
```

Following is a structure of address extension header specified in [RFC2367]. Upper layer protocol should be specified in member sadb_address_proto.

```c
struct sadb_address {
    uint16_t sadb_address_len;
    uint16_t sadb_address_exttype;
    uint8_t sadb_address_proto;
    uint8_t sadb_address_prefixlen;
    uint16_t sadb_address_reserved;
};
```

Following is a structure for holding attributes that are relevant to security policy, which is available on BSD IPsec implementation.
Direction of the target security policy should be specified in member sadb_x_policy_dir.

```c
struct sadb_x_policy {
    uint16_t    sadb_x_policy_len;
    uint16_t    sadb_x_policy_exttype;
    uint16_t    sadb_x_policy_type;
    uint8_t     sadb_x_policy_dir;
    uint8_t     sadb_x_policy_reserved;
    uint32_t    sadb_x_policy_id;
    uint32_t    sadb_x_policy_priority;
};
```

Following is a structure for holding attributes that are relevant to security association, which is available on BSD IPsec implementation. IPsec protocol (ESP or AH) and mode (Tunnel) of the target security association should be provided in member sadb_x_ipsecrequest_proto and sadb_x_ipsecrequest_mode, respectively.

```c
struct sadb_x_ipsecrequest {
    uint16_t    sadb_x_ipsecrequest_len;
    uint16_t    sadb_x_ipsecrequest_proto;
    uint8_t     sadb_x_ipsecrequest_mode;
    uint8_t     sadb_x_ipsecrequest_level;
    uint16_t    sadb_x_ipsecrequest_reserved1;
    uint32_t    sadb_x_ipsecrequest_reqid;
    uint32_t    sadb_x_ipsecrequest_reserved2;
};
```
Appendix B. Acknowledgements

The authors gratefully acknowledges the contribution of: Masahide Nakamura, Kazunori Miyazawa, Noriaki Takamiya.
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.