Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2009, Article ID 716480, 15 pages
doi:10.1155/2009/716480

Research Article

Secure Media Independent Handover Message Transport in Heterogeneous Networks

Jeong-Jae Won,[1] Murahari Vadapalli,[1] Choong-Ho Cho,[2] and Victor C. M. Leung[3]

[1] *Telecommunication and Network R&D Center, Samsung Electronics Co., LTD., 416 Maetan-3dong, Yeongtong-gu, Suwon-si, Gyeonggi-do 443-742, South Korea*
[2] *Department of Computer & Information Science, Korea University, Chung-Nam 339-700, South Korea*
[3] *Department of Electric & Computer Engineering, The University of British Columbia, 2332 Main Mall, Vancouver, BC, Canada V6T 1Z4*

Correspondence should be addressed to Victor C. M. Leung, velung@ece.ubc.ca

Received 31 January 2009; Accepted 21 September 2009

Recommended by Yang Xiao

The IEEE 802.21 framework for Media Independent Handover (MIH) provides seamless vertical handover support for multimode mobile terminals. MIH messages are exchanged over various wireless media between mobile terminals and access networks to facilitate seamless handover. This calls for the need to secure MIH messages against network security threats in the wireless medium. In this paper, we first analyze IPSec/IKEv2 and DTLS security solution for secure MIH message transport. We show that handover latency can be an impediment to the use of IPSec and DTLS solutions. To overcome the handover overhead and hence minimize authentication time, a new secure MIH message transport solution, referred as MIHSec in this paper, is proposed. Experimental results are obtained for MIH between WLAN and Ethernet networks and the impacts of MIH message security on the handover latency are evaluated for IPSec, DTLS, and MIHSec security solutions. The effectiveness of MIHSec is demonstrated.

1. Introduction

Modern access systems have the capability to fulfill a specific quality-of-service (QoS) to the user, which leads to a requirement for seamless transitions from one access network to another in the presence of terminal mobility. Thus, it is anticipated that seamless interradio access technology (inter-RAT) mobility will be widely deployed in modern heterogeneous networks such as IEEE 802.11 (Wi-Fi), Global System for Mobile Communications (GSM), code-division multiple access (CDMA), and Mobile WiMAX. The growing importance of these issues has attracted the attention of standard groups including the IEEE 802.21 work group. The IEEE 802.21 standard defines Media Independent Handover (MIH) mechanisms that enable the optimization of inter-RAT handovers in heterogeneous networks [1–4].

The emerging IEEE 802.21 standard enables seamless, inter-RAT handover between IEEE 802 and non-IEEE 802 (e.g., 3GPP, 3GPP2) access technologies with the MIH function (MIHF) in the terminal and network sides. The role of MIHF is to provide media independent services to multi-RAT mobile terminals (MMTs) through a common interface to the mobility management and handover processes.

Related to this work, handover provisioning between GPRS and WiMAX is suggested in [2], which utilizes the potential of IEEE 802.21 to efficiently support inter-RAT handovers with full description of MIH services such as information service for providing network information, event service to trigger layer 2 (L2) events, and command service for handover execution like resource reservation and handover request. Reducing the authentication time over heterogeneous access networks involving interdomain mobility is a very critical criterion for seamless handover. In [3], Media independent preauthentication (MPA) provision is suggested. MPA provides a significant reduction in handover delays for both network-layer and application-layer mobility management protocols. However, the MPA scheme [3] does not address secure transport of media independent messages.

In addition to authentication as described in MPA [3], confidentiality and message integrity of MIH messages is another necessary requirement.

The requirements for MIH message level security are described in the 802.21 Security Study Group proposals [4]. The following security issues are identified.

(i) *MIH Access Control*. MIH service access should be controlled based on authentication and authorization.

(ii) *Replay Protection*. An MIH packet for an event or command can be replayed later to the same node.

(iii) *Denial of Service*.

(iv) *Message Integrity*. An MIH message may be altered on the way.

The available solutions for supporting authentication and access security are IP Security (IPSec) [5] and Datagram Transport Layer Security (DTLS) [6]. IPSec is a security solution at the network layer and is commonly used for most Internet applications. DTLS is a security solution at the transport layer, used for applications that operate over the User Datagram Protocol (UDP) or Transmission Control Protocol (TCP). In contrast to these existing security solutions, an MIH Security (MIHSec) solution is proposed and analyzed in this paper. Unlike IPSec and DTLS, MIHSec operates at the application layer.

The following MIH message protection issues are considered in this paper:

(i) communications between MIHF in MMT and any MIH Points of Service (PoS) in the access network,

(ii) communications between MIHF in MMT and MIH Information Server,

(iii) communications between MIHF in MMT and MIH IWF Broker. IWF provides the proprietary function between MIH services and a specific access network,

(iv) communications between MIHF in access routers (ARs).

In this paper, we first analyze IPSec with Internet Key Exchange version 2 (IKEv2) and DTLS security solutions for secure MIH message transport. We show that handover latency is an impediment to the use of IPSec and DTLS solutions. To overcome the handover overhead and hence minimize authentication time, a new secure MIH message transport solution, referred as MIHSec in this paper, is proposed.

IPSec and DTLS are off the shelf security solutions and software for them is readily available as GNU source. However, MIHSec is a newly defined security solution for providing security to MIH Messages. MIHSec operates at the application layer and utilizes Extensible Authentication Protocol (EAP) and MIH header TLV extensions to provide security to MIH messages.

Prototypes of MIH security methods with IPSEc/IKEv2, DTLS, and the new MIHSec mechanism are developed and the results are compared based on IEEE 802.21 Draft 11 for handover scenarios between Wi-Fi and Ethernet networks. The impacts on signaling latency, message transport latency, message overhead, and configurations are analyzed.

The rest of this paper is organized as follows. In Section 2, we provide background information on the IEEE 802.21 standard. In Section 3, we define the secure MIH transport models. In Section 4, the feasible methods for secure MIH transport with existing solutions such as IPSec/IKE and DTLS are analyzed. In Section 5, we present the design of our new secure MIH message transport protocol called MIHSec. In Sections 6 and 7, we exemplify the prototype by implementing and testing with MIHF implementation between Wi-Fi and Ethernet networks. Section 8 concludes the paper.

2. Related Work

2.1. IEEE 802.21 Standard. IEEE 802.21 [1] is a recent effort of IEEE that aims at enabling seamless service continuity among heterogeneous networks including 3GPP, 3GPP2, and the IEEE 802 family of standards. The standard defines a logical entity, MIHF, which is located between the lower layer (L2 and below) and upper layer. At the lower layer, MMT has multiple radio interfaces for different access technologies such as WLAN, WiMAX, and 3GPP. Upper layer entities that use the services provided by MIHF are referred as MIH Users. The role of MIHF is providing media independent services to MIH Users through a common interface to facilitate mobility management and handover processes.

Figure 1 shows the overview of MIH framework outlined by IEEE 802.21 standard. There are three primary services: Media Independent Event Service (MIES), Media Independent Command Service (MICS), and Media Independent Information Service (MIIS). MIES may indicate or predict changes in a state and transmission behavior of the physical and link layers. Common MIES provided through MIHF are “Link Up,” “Link Down,” “Link Parameters Change,” and “Link Going Down.” MICS enables higher layers to configure, control, and obtain information from the lower layers including physical and link layers. The information provided by MICS is dynamic information comprised of link parameters, whereas information provided by MIIS is comprised of static parameters.

MIIS provides a unified framework for obtaining neighboring network information that exists within a geographical area. It helps the higher layer mobility protocol to acquire a global view of available heterogeneous networks to conduct effective seamless handover. The information may be present in MMT locally but is usually stored in some external information server, which may be accessed by the MIHF in the MMT. For MIIS, the IEEE 802.21 standard defines information structures called Information Elements (IEs) that are classified into two groups: access network specific information (type of network, roaming agreements, cost of connecting, and QoS capabilities) and Point of Attachment (PoA) specific information (channel range, location, and supported data rates).

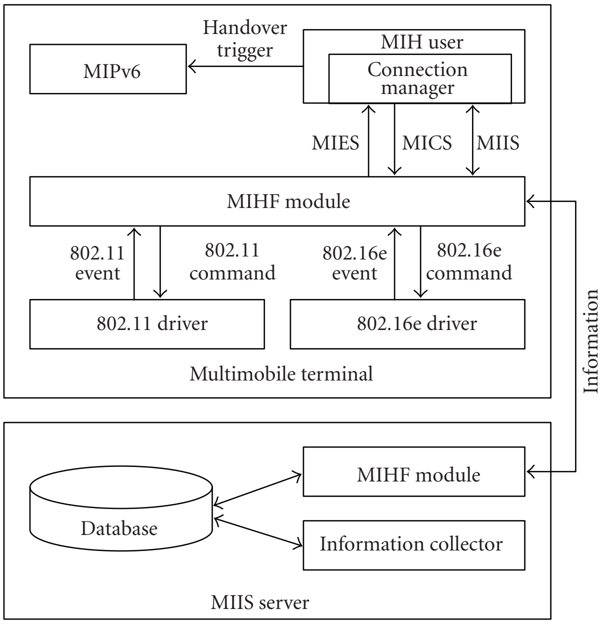

Figure 1: Overview of MIH framework.

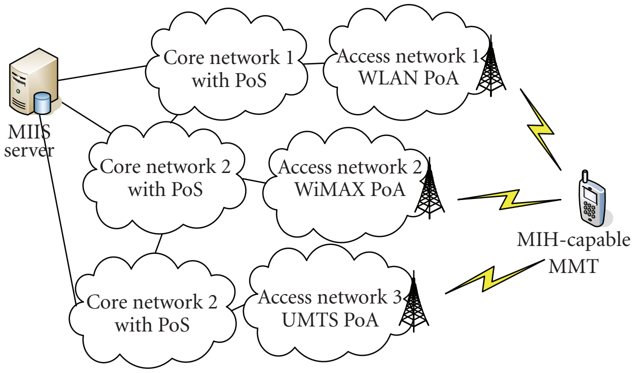

Figure 2: Network model with MIH services.

Figure 2 shows an example of the network model including MIH services. An MIH-capable MMT has multiple wireless interfaces based on different access technologies. It can connect concurrently to multiple PoAs, which are network side endpoints of L2 links. Each access network provides one or more MIH PoS nodes. To provide MIIS, an MIIS server can be located on the network side. The server maintains information of neighboring access networks in its local database.

Figure 3 shows the MIH-based handover message exchange involved in a mobile initiated handover from the serving network to the target network. The detailed explanation of the messages and procedures are as follows. The MIH procedure starts with the MMT querying about the surrounding networks. This query is forwarded by the information server located in the operator network and answered to MMT with available candidate network information (message 1-2). As the answer contains information regarding a possible network, the MMT switches on its target network interface and starts to measure the candidate networks. Just after measuring the candidate network, MMT will generate an MIH_MN_Candidate_Query message asking for the list of resources available in candidate networks and including the QoS requirements of the user (message 3–6).

At this point, the MMT has enough information about the surrounding networks to decide on the network to which it will hand over. Once the MMT has decided the target network to hand over, it delivers a handover commit command to the MIHF (message 7–10), which will be used for resource reservation in the target network before switching from the serving network to the target network (L2 and L3 handover). After completion of resource reservation in the target network, the MMT starts to establish the connection in the target network. Once the connection is established, a higher-layer handover procedure can start. In this case Mobile IP has been selected, although any other mobility management protocol would be equally suited. When the handover is completed at the higher layers, the MMT sends an MIH_HO_Complete message to the MIHF, which will inform the target PoS that it is now the new serving PoS. At this point the target PoS informs all the involved network elements of the handover finalization (message 11–14). Specifically, the target PoS has to inform the serving PoS of the handover completion so that it can release any resources.

2.2. Existing Secure Transport Methods. IP Security [5] and DTLS [6] are the existing secure transport methods currently available in the market, which support authentication and access security for the MIH messages. Figure 4 shows the integration of the security framework in the existing MIH framework.

2.2.1. IPSec/IKEv2. IPSec [5] provides a standard mechanism for data security for protocols running over IP. Since the MIH messages (in the prototype implementation of MIHF) use UDP over IPv6 for transport, IPSec can be an automatic choice for message protection. However, since IPSec needs a preconfigured trust relationship between the communicating end points, the feasibility and efficiency of this method needs to be examined in the context of handover to different access networks.

Figure 5 shows the messages exchanged between MIH enabled nodes, to setup the IPSec tunnel using IKEv2.

2.2.2. Datagram Transport Layer Security. The DTLS [6] protocol provides communication privacy for datagram protocols. It is designed to run in the application space, without requiring any kernel modifications. The basic design philosophy of DTLS is to construct "TLS over datagram." The reason that TLS cannot be used directly in datagram environments is simply that packets may be lost or reordered. TLS has no internal facilities to handle this kind of unreliability, and therefore TLS can break when hosted on datagram transport. The purpose of DTLS is to make only the minimal changes to TLS required to fix this problem. To the greatest extent possible, DTLS is identical to TLS.

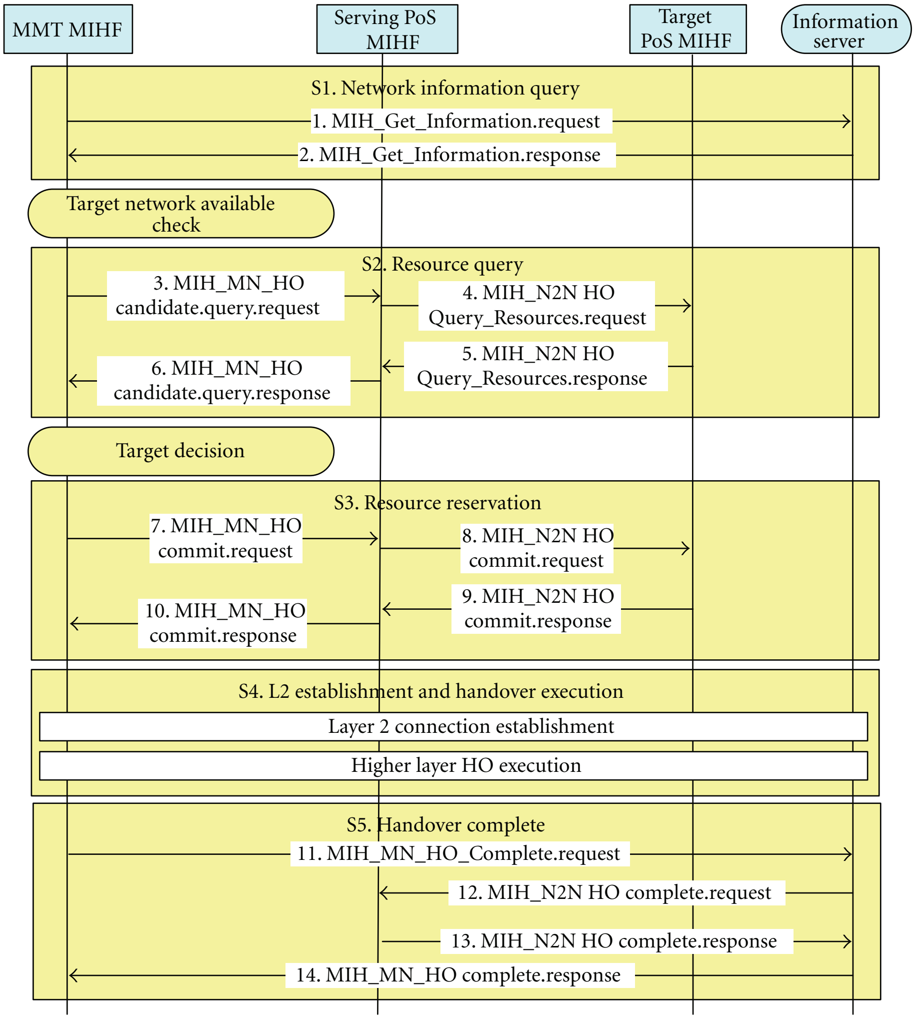
MMT MIHF
Serving PoS MIHF
Target PoS MIHF
Information server
S1. Network information query
1. MIH_Get_Information.request
2. MIH_Get_Information.response
Target network available check
S2. Resource query
3. MIH_MN_HO candidate.query.request
4. MIH_N2N HO Query_Resources.request
5. MIH_N2N HO Query_Resources.response
6. MIH_MN_HO candidate.query.response
Target decision
S3. Resource reservation
7. MIH_MN_HO commit.request
8. MIH_N2N HO commit.request
9. MIH_N2N HO commit.response
10. MIH_MN_HO commit.response
S4. L2 establishment and handover execution
Layer 2 connection establishment
Higher layer HO execution
S5. Handover complete
11. MIH_MN_HO_Complete.request
12. MIH_N2N HO complete.request
13. MIH_N2N HO complete.response
14. MIH_MN_HO complete.response

Figure 3: MIH-based handover—call flow.

Figure 6 shows the DTLS protocol messages exchanged between client and server for establishing a DTLS association.

2.2.3. New MIHSec Transport Method. In the above section, the current secure transport methods like IPSec and DTLS are discussed. In contrast to these two methods, a new method known as MIHSec is proposed in this paper. MIHSec provides solutions to the problems that arise in using IPSec and DTLS for MIH-based handover applications. The details of the problems and solutions are presented in the subsequent sections.

3. Secure MIH Transport Models

This paper discusses two secure transport models that are commonly used in general security architectures [7] like IPSec.

The end-to-end security model provides protection to the messages on an end-to-end basis; that is, packets encrypted at source is decrypted at the end point. And the other model is the end point-to-security gateway model, wherein packets are encrypted between the endpoint and the gateway, which is to say that the packets should be encrypted/decrypted multiple times on its transmission to the destination node. Elaborate descriptions of these two models, when applied to the MIH solution, are given in the subsequent paragraphs.

3.1. End-to-End Protection. In this model, a secure channel is established from the MMT to each MIH service end-point in the network, before any MIH message exchange can take place. The secure channel source is MMT and the destination is Interworking Function (IWF), MIH Information Service (IS) server, and PoS. IWF provides the proprietary function between MIH services and a specific access network. This is out of the scope of IEEE 802.21.

The secured path shall provide data integrity, authenticity, and confidentiality as desired. The MIH on MMT will be responsible for setting up and terminating the secure channel. An encrypted packet sent from MMT can be decrypted at IWF, IS server, and PoS only. Other than the

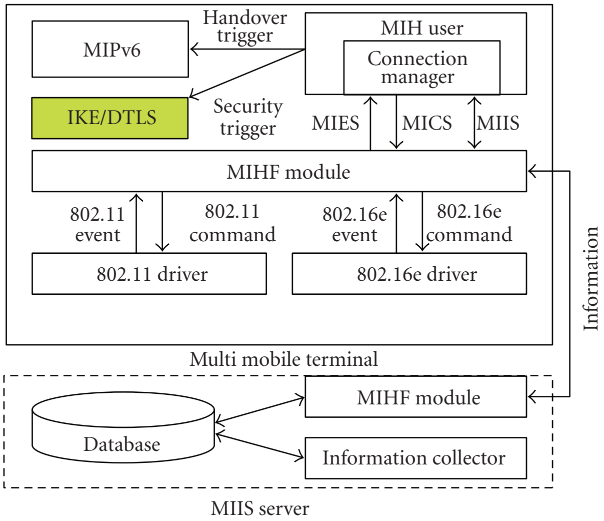

Figure 4: Secure transport module in MIH framework.

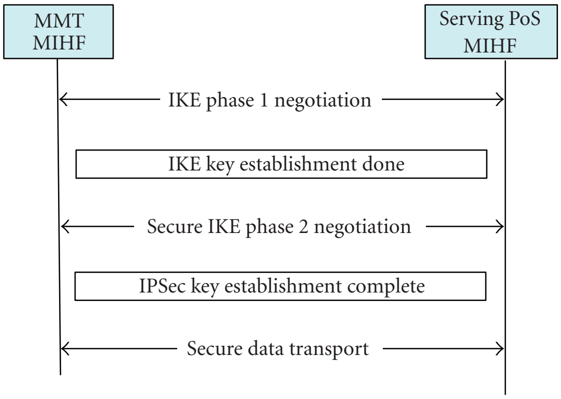
MMT MIHF
Serving PoS MIHF
IKE phase 1 negotiation
IKE key establishment done
Secure IKE phase 2 negotiation
IPSec key establishment complete
Secure data transport

Figure 5: IPSec tunnel establishment.

destination node, the nodes on the path cannot decrypt the packet. This model provides security between the nodes that are residing in the end point of the transmission paths.

For example, during handover to a new access network, the MIH entity in MMT should trigger the IKEv2 daemon to establish an IPSec security association (SA) with MIH PoS for MIH command and event service in the new access network, before sending the MIH-MN-HO-Complete message. It should also establish IPSec SA with the MIH IS server in the same way, before sending any MIH_Get_Information request message to the IS server. Similarly, a secure channel has to be established between MMT and IWF Proxy before transmitting any packet between the MMT and IWF Proxy nodes. The tunnel between MMT and AR is identified as T2, the tunnel between MMT and IWF Proxy is identified as T1, and the tunnel between MMT and IS server is identified as T3. This is illustrated in Figure 7.

3.2. Endpoint-to-Security Gateway Protection. In this model, a secure channel is established from the MMT to the AR in the access network, before any MIH message exchange

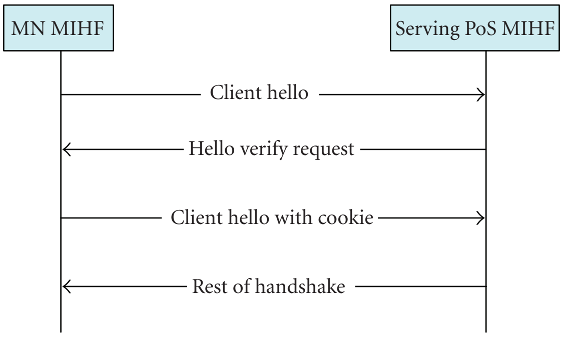

Figure 6: DTLS client server message exchange.

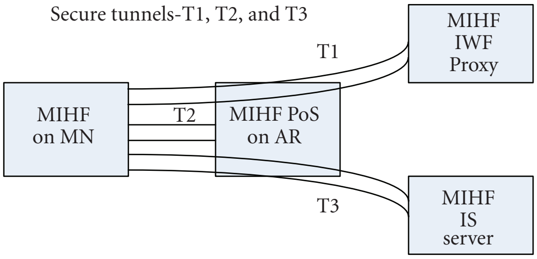

Figure 7: MIH message security through end-to-end tunnels.

can take place between MMT and AR. The source is the MMT and the destination is AR. And similarly when the packet is sent from AR, the source is AR and the destination is the MMT. The secured path shall provide data integrity, authenticity, and confidentiality as desired. The MIH on MMT will be responsible for setting up and terminating the secure channel with the AR. The AR will be responsible for establishing a secure channel between itself and each MIH node in the network, like IWF Proxy or IS server.

For example, during handover to a new access network, the MIH entity in MMT should trigger the IKEv2 daemon to establish an IPSec SA with the new AR, before sending the MIH_MN_HO_Complete Message. Establishment of a secure channel is done before transmitting any MIH packet.

In this method, the destination end point may or may not be the logical end point of the tunnel. For example, when MMT sends an MIH_Get_Information request message to the IS server, the packet traverses through tunnel T1 and tunnel T3 to reach the destination—IS server. As shown in Figure 8, the tunnel between MMT and AR is known as T1, the tunnel between AR and IS server is T3, and the tunnel between AR and IWF Proxy is T2.

The analysis in this paper focuses on security through end-to-end tunnels, as illustrated in Figure 7, and the experimental results are based on that model only. However, similar results are expected in the endpoint to gateway tunnel method also, as illustrated in Figure 8.

The endpoint to gateway approach would have an advantage when it is assumed that the secure channel T1 is not required as this path will be protected by L2 security. In such a case the overhead of security will be avoided in the wireless link.

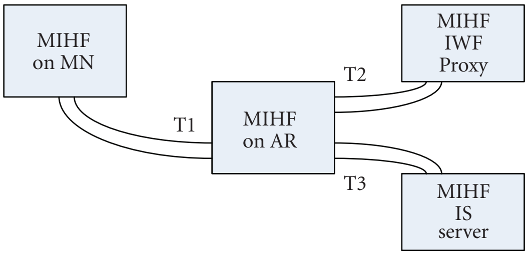

Figure 8: MIH message security through endpoint to gateway tunnels.

Hence in this paper, the endpoint to endpoint tunnel method is considered.

4. Analysis of Secure MIH Message Transport with Existing Solutions

4.1. Requirement of Secure MIH Message Transport. The MIH-enabled nodes in the network have the capability to handle the Event Service (ES), Command Service (CS), and Information Service (IS) requests. These service messages carry manifold information, which is helpful to the decision process in MIHF to perform the handover functionality in the network and node elements.

The MIH messages are transmitted over the Internet between the MIH enabled access node, the IS server, and IWF proxy. For MMTs these messages are sent over the wireless network and the wired infrastructure that make up the access domain.

As an MIH message is transmitted over insecure channels on its path to the destination, it becomes an obligation to secure these messages from hackers who are trying to hijack the channels, spoof the packets, or snoop in the network.

This section discusses the list of security features that are required to be incorporated in the MIH messages.

4.1.1. MIH Access Control. Based on policies, an MIH PoS in the operator network may want to allow only certain MIH services to the MIH entity in the MMT. The access control can be enforced through IPSec/IKEv2, DTLS or by defining new information elements as a part of the MIH protocol.

4.1.2. MIH Replay Protection and Denial of Service. MIH packets may be spoofed or packets may be replayed by an attacker. By using IPSec SA or DTLS session for all MIH message exchanges, these attacks can be prevented. An MIH protocol level method may also be considered for protection against this attack by including timestamp/sequence number in the MIH messages.

4.1.3. MIH Data Integrity and Confidentiality. MIH data integrity and confidentiality can be achieved through IPSec and DTLS. A sufficiently strong encryption and integrity algorithm, for example, aes-cbc/256-bit and hmac-sha1/128-bit, can be negotiated between MIH peers during IKEv2 [8] signaling or DTLS handshake to ensure protection.

An MIH protocol-based approach can be used for message integrity. For example, a message authentication code information element may be included in each MIH message, which needs to be protected for data integrity.

All three methods for MIH message protection are analyzed in this paper to identify the scope of prototyping and experimentation. Based on the prototyping and experimentation results, the IPSec, DTLS, and new MIHSec methods will be evaluated for ease of configuration, efficiency, and handover latency.

4.2. Methods of Securing MIH Message Transport with Existing Solutions

4.2.1. IPSec/IKEv2. In Figure 9, MIHF will trigger the IKEv2 daemon to establish an IPSec SA with the MIH endpoint before any MIH message exchange can take place.

Each MIH end-point shall perform the following steps:

(1) get X.509 Certificate from a trusted certificate authority (CA) by supplying the MIHF ID,

(2) install the CA certificate and the host certificate,

(3) exchange the credentials with the other MIHF end point and verify the other end-point's certificate and MIHF ID,

(4) update the IPSec policy database (SPD) and IPSec association database (SAD) for protection of MIH Message (UDP/MIH_PORT) sent to and received from the other MIH endpoint.

The credentials are exchanged and verified by the IKEv2 daemon in IKE_SA_INIT and IKE_SA_AUTH. This method requires that the MIHF endpoints know the MIHF ID of the other MIH end point. How the MIHF IDs of MIH PoS in the target network are obtained is the topic of "MIHF Discovery Analysis". Table 1 lists various scenarios in this regard and the possible ways to get the MIHF ID.

(a) IPSec/IKEv2 Pros and Cons.

Pros has the following.

(1) IPSec provides the most standard solution for data security for protocols running over IP. Even the IP header can be protected by using IPSec in tunnel mode.

(2) IPSec support is readily available in all standard operating systems.

(3) Using IKEv2, security keys can be configured automatically.

(4) Using IKEv2 with EAP allows the security credentials to be verified by the authentication, authorization, and accounting (AAA) server for the access network.

Cons has the following.

(1) IKEv2 signaling adds to latency in handover.

(2) IPSec header adds overhead to packets send over the air interface.

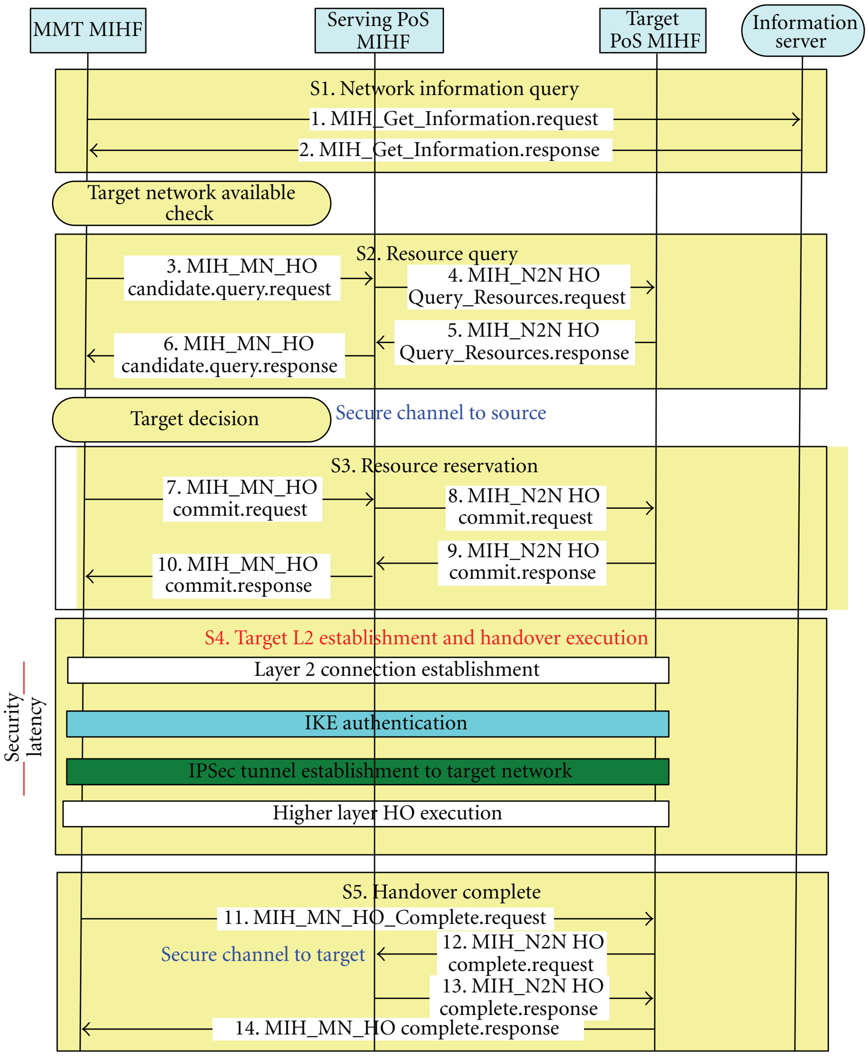
MMT MIHF
Serving PoS MIHF
Target PoS MIHF
Information server
S1. Network information query
1. MIH_Get_Information.request
2. MIH_Get_Information.response
Target network available check
S2. Resource query
3. MIH_MN_HO candidate.query.request
4. MIH_N2N HO Query_Resources.request
5. MIH_N2N HO Query_Resources.response
6. MIH_MN_HO candidate.query.response
Target decision
Secure channel to source
S3. Resource reservation
7. MIH_MN_HO commit.request
8. MIH_N2N HO commit.request
9. MIH_N2N HO commit.response
10. MIH_MN_HO commit.response
S4. Target L2 establishment and handover execution
Layer 2 connection establishment
IKE authentication
IPSec tunnel establishment to target network
Higher layer HO execution
Security latency
S5. Handover complete
11. MIH_MN_HO_Complete.request
Secure channel to target
12. MIH_N2N HO complete.request
13. MIH_N2N HO complete.response
14. MIH_MN_HO complete.response

Figure 9: Securing MIH with IPSec.

(3) IPSec ciphering algorithm execution adds to latency in handover.

(4) Integration of MIH with IKE is an issue with handover as IP address changes in MMT.

4.2.2. Datagram Transport Layer Security. In Figure 10, DTLS is used for secure MIH transport, which uses all of the same handshake messages and flows as TLS, with three principal changes:

(1) a stateless cookie exchange has been added to prevent denial of service attacks,

(2) modifications to the handshake header to handle message loss, reordering, and fragmentation,

(3) retransmission timers to handle message loss.

(a) DTLS Pros and Cons.

Pros has the following.

(1) DTLS is an application layer protocol.

(2) No kernel modification is required.

(3) It does not depend on any underlying reliable transport protocol.

(4) It can be implemented with lesser modification of existing TLS.

(5) It is closer to functionalities of IPSec but cheaper.

Cons has the following.

(1) DTLS signaling which involves multiple handshake messages between client and server adds to latency.

(2) DTLS is not independent protocol. DTLS will internally use TLS library. So TLS library support is required.

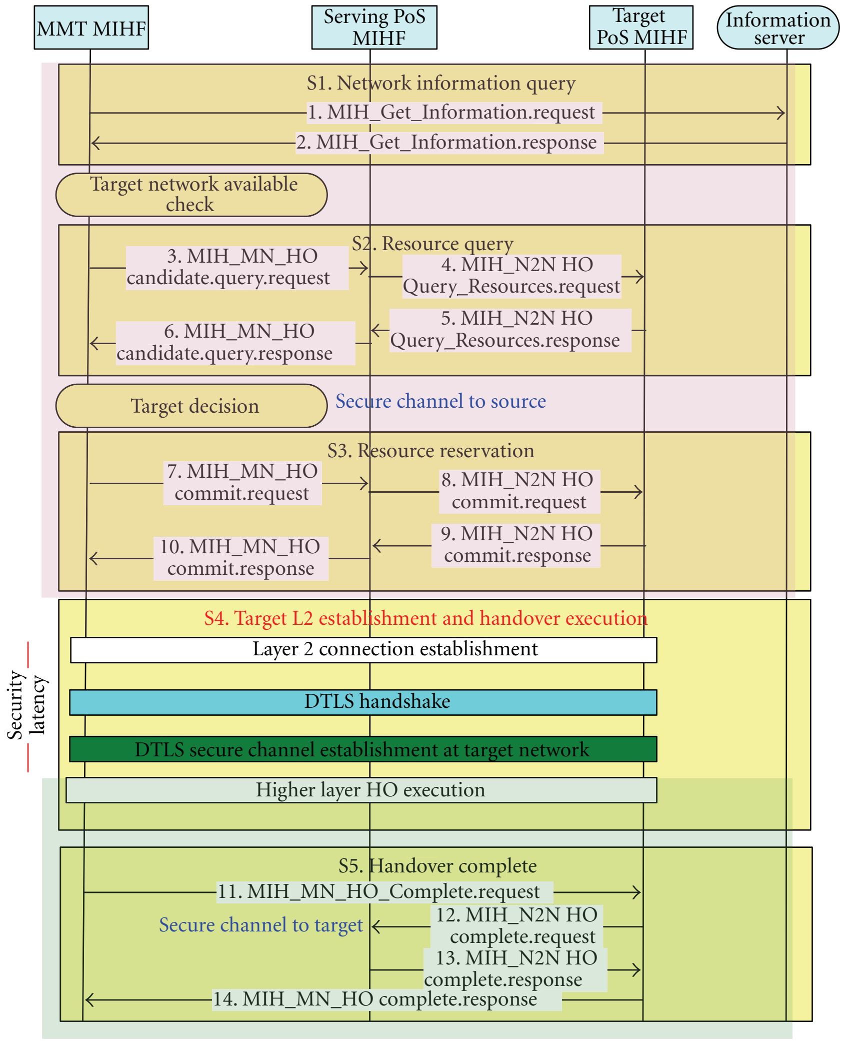
MMT MIHF
Serving PoS MIHF
Target PoS MIHF
Information server
S1. Network information query
1. MIH_Get_Information.request
2. MIH_Get_Information.response
Target network available check
S2. Resource query
3. MIH_MN_HO candidate.query.request
4. MIH_N2N HO Query_Resources.request
5. MIH_N2N HO Query_Resources.response
6. MIH_MN_HO candidate.query.response
Target decision
Secure channel to source
S3. Resource reservation
7. MIH_MN_HO commit.request
8. MIH_N2N HO commit.request
9. MIH_N2N HO commit.response
10. MIH_MN_HO commit.response
S4. Target L2 establishment and handover execution
Layer 2 connection establishment
DTLS handshake
DTLS secure channel establishment at target network
Higher layer HO execution
Security latency
S5. Handover complete
11. MIH_MN_HO_Complete.request
Secure channel to target
12. MIH_N2N HO complete.request
13. MIH_N2N HO complete.response
14. MIH_MN_HO complete.response

Figure 10: Securing MIH with DTLS.

Table 1: Methods for getting MIHF ID's.

MIHF Host	Scenario	Solution
MIHF on MMT	To get PAR MIHF ID during start up	MIHF Discovery methods. Listen to MIHF Capability Discover Broadcast
MIHF on MMT	To get NAR MIHF ID during HO	MIHF Discovery methods (DHCP/DNS). Listen to MIHF Capability Discover Broadcast
MIHF on MMT	To get IS server MIHF ID	MIHF Discovery methods (DHCP/DNS)
MIHF on PAR	To get NAR MIHF ID	Listen to MIHF Capability Discover Broadcast
MIHF on PAR	To get IS server MIHF ID	MIHF Discovery methods (DHCP/DNS)

5. Method for Securing MIH Messages with Protocol Extensions to MIH (MIHSec)

5.1. Motivation for a New Secure MIH Messages Transport Protocol. In the previous sections we discussed IPSec and DTLS solution to provide security to the MIH messages. The IPSec operates at IP layer and the DTLS at the application layer to provide security to the MIH messages.

The IPSec and DTLS could suffice the requirements for providing security to the MIH messages. The steps carried out to provide secure transmission of MIH messages are provided in Figure 11.

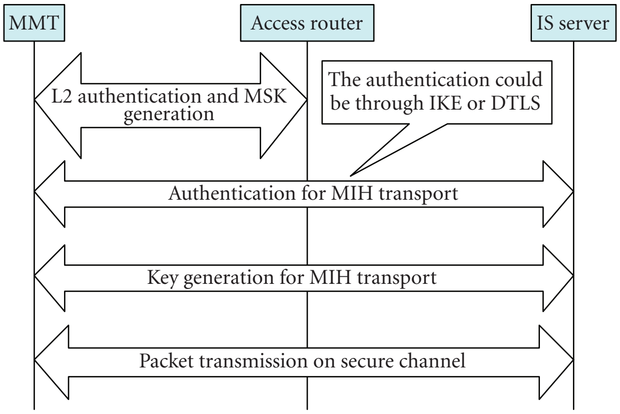
MMT
Access router
IS server
L2 authentication and MSK generation
The authentication could be through IKE or DTLS
Authentication for MIH transport
Key generation for MIH transport
Packet transmission on secure channel

FIGURE 11: IPSec/DTLS key generations at IS server.

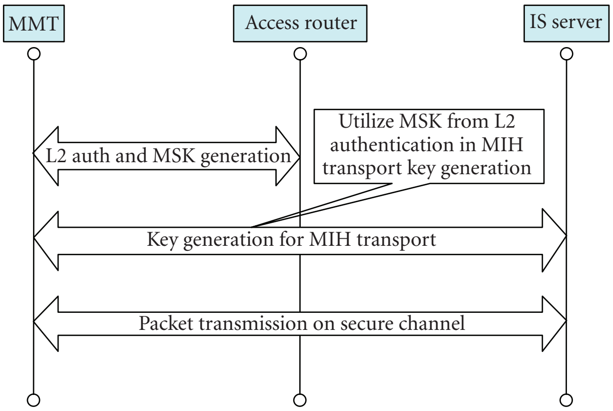
MMT
Access router
IS server
L2 auth and MSK generation
Utilize MSK from L2 authentication in MIH transport key generation
Key generation for MIH transport
Packet transmission on secure channel

FIGURE 12: MIH transport key generation at IS server using L2 authentication MSK Key.

The L2 authentication is performed between the MMT and AR. This provides a secure communication channel on the air interface between MMT and AR.

The MIH Transport Authentication—which can be IKEv2 or DTLS—is carried out next to authenticate MMT with the MIH network entity. In Figure 11, IS server is considered as an MIH entity, for example, illustration. Upon completion of the authentication with the IS server, the MIH IK and the CK keys are generated. These keys are used by the MIH layer to provide the secure communication channel between the MMT and IS server.

The inherent problem with IPSec/DTLS security method is multiple authentications (L2 authentication and Authentication for MIH Transport) that occur in the flow. The additional MIH transport authentication would add to the latency during the handover, which in turn degrades the performance of handover. If MIH transport authentication can be eliminated, the handover latency time will be minimized. This section discusses basic idea to provide the MIH Security at the application layer by providing enhancements to the 802.21 standard.

5.2. Enhancements to 802.21 to Support MIH Security (MIHSec)

5.2.1. The Concept of MIHSec. The inherent disadvantages of DTLS and IPSec in the handover scenarios would support the need for developing a new integrated security feature in MIH messages. The important requirement is minimization of handover latency and support of confidentiality and integrity protection to the MIH messages.

The idea here is to eliminate the MIH transport authentication and utilize the Master Shared Key generated by the L2 authentication procedure, for generating the MIH keys. Avoiding MIH transport authentication step would enhance the handover latency and hence better performance during the handover as shown in Figure 12.

The solution that is proposed here would utilize the authentication provided at the L2 layer. In most of the access networks, available in today's market, the authentication is provided by using the EAP standard.

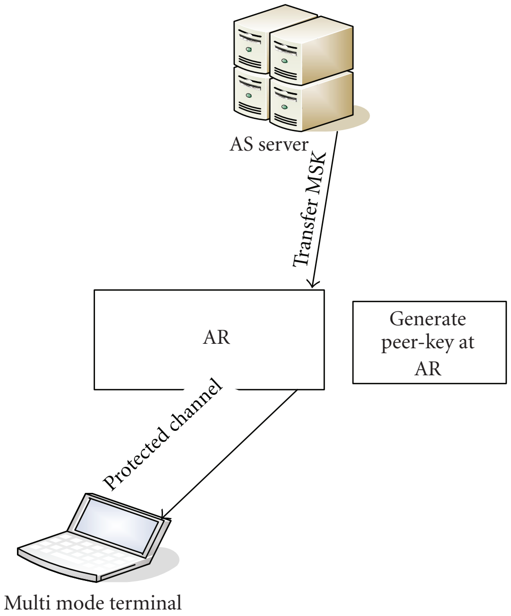

FIGURE 13: Generation of peer-key.

MIH protocol would utilize the MSK generated by the EAP, to generate its own CK and IK. The advantage of using MSK of L2 authentication is (a) low latency and (b) maintenance of key hierarchy—in security parlance, its also known as perfect forward secrecy.

Upon completion of L2 authentication, MSK is sent to AR in the Access Network. The AR sends the MSK and MAC address of the MMT to the IS server.

The MSK is utilized by MIHF in AR to generate a Peer-Key in MIHF node in AR. And also the MSK is utilized by MIHF in IS server to generate IS-Key.

To summarize, between the MMT and AR nodes, Peer-Key is generated and between MMT and IS server nodes IS-Key is generated. Peer-Key is the key hierarchy between MMT and AR and IS-Key is the key hierarchy between MMT and IS server.

5.2.2. MIH Key Generation Procedure. In Figure 13, the multimode mobile terminal performs authentication with the access network. This is done using the EAP protocol. The result of the authentication is the generation of the MSK key. The peer MIH function in AR uses the MSK key, along with other parameters to generate a Peer-Key. The algorithm for generating the keys is described in the following section.

(a) Algorithm for Security Keys Generation between Mobile Terminal and PoA. The Peer-Key is used to establish secure channel between MMT and PoA. The pseudocode for generating the security keys is described as follows:

Algorithm 1. Key_generation_algorithm_in_MIHPeer().

```
Begin:
Get the MSK key of EAP
    Use the keyed-md5 as Pseudo Random Function for generating the Peer-Key
    Peer-Key = Keyed-md5(MSK, MAC-Peer, MAC-PoA)
    // The inputs to the prf are MAC address of MMT and MAC address of PoA
    The result of keyed-md5 is Peer-Key
    Peer-Key is a 128 bit hash value
    Use Peer-Key to generate the CK and IK
        Cipher Key = prf(Peer-Key, "Peer", 0)
    Integrity Key = prf(Peer-Key, "Peer", 1)
// The 0 and 1 in the prf function indicate whether the key generated is the CK or the IK
End:
```

CK(Ciphering Key) and IK(Integrity Key) generated are used to secure the MIH Data, along with the MIH headers

(b) Algorithm for Generating Security Keys between MMT and IS Server. IS-Key is used to establish secure channel between the mobile terminal and the IS server. The algorithm for generating security keys between IS server and MMT is mentioned here in after.

The pseudo code for generating security keys is described as follows:

Algorithm 2. Key_generation_algorithm_in_MIHServer().

```
Begin:
Get the MSK key of EAP
    Use the keyed-md5 as Pseudo Random Function for generating the Peer-Key
    IS-Key = Keyed-md5(MSK, ISServer-IPAddress,
MAC-Peer)
    // The inputs to the prf are IP Address of the IS server and MAC address of MMT
    The result of keyed-md5 is IS-Key
    Peer-Key is a 128 bit hash value
    Use IS-Key to generate the CK and IKs between the MMT and the IS server
        Cipher Key = prf(IS-Key, "IS-Server", 0)
    Integrity Key = prf(IS-Key, "IS-Server", 1)
        // The 0 and 1 in the prf function indicate whether the key generated is the CK or the IK
End:
```

5.2.3. Extensions to MIH Header. IP Security operates at IP layer. An extension to the IP header has been provided to incorporate security features in IP. Similarly there is a need to provide security extension headers to the current MIH standard for providing security features in 802.21. The objective of these extension headers is to carry message digest between tunnel end points, to enable the end points to validate the packet data and header information.

In order to support security at the MIH, extensions need to be provided at MIH Header as illustrated in Figure 14. This is due to the fact that the MIH layer at the destination has to identify if the MIH packet is security protected or not. Hence, two new TLVs are added to support the security feature in MIH. An encryption TLV and integrity TLV are provided as an extension for MIHSec. The illustration of the same is provided in Figure 11.

And as illustrated in Figure 15, encryption is provided over MIH data and confidentiality is provided over MIH header and MIH data.

When a secure MIH packet is to be transmitted from MMT to IS server, MIHF in MMT performs confidentiality protection first and then applies integrity protection on header and data. At the destination node, the MIHF in the IS server performs integrity checking initially and if the integrity check is passed, confidentiality check is done. If either of integrity check or the confidentiality check fails, that packet is dropped.

Integrity protection checking is done first, before performing the deciphering functionality.

5.2.4. Benefits of MIH Security Solution.

(i) A separate authentication mechanism (like IKE authentication or DTLS authentication) is not necessary as the MSK keys from the L2 authentication are utilized in maintaining key hierarchy and also for generating the MIH CK and IK keys.

(ii) The handover latency is minimized due to elimination of IKE/DTLS authentication procedure.

(iii) Changes to the MIH code are minimal to support confidentiality and integrity protection and hence the ease of integration with the present code.

(iv) Available PRF algorithms can be reused.

(v) The last one is the protection against Denial of Service.

MIH-type (confidentiality/integrity)	MIH length	MIH value (128 bit cipher or 128 bit hash)

Figure 14: MIH extension header.

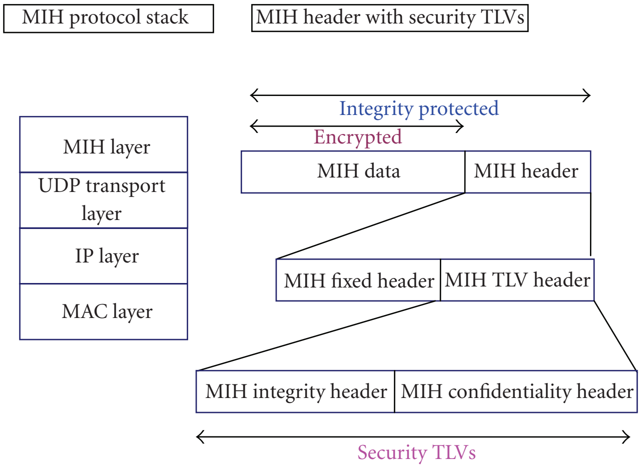

Figure 15: MIH with security TLV.

5.3. Performance Evaluation Parameters

5.3.1. Security Signaling Latency. Security signaling latency is defined as time taken to perform the authentication and security key generation, along with the tunnel establishment time:

$$\begin{aligned}\text{Security Signaling Latency} &= \text{Authentication Time}\\ &\quad + \text{Key generation Time}\\ &\quad + \text{Tunnel Establishment Time}.\end{aligned} \tag{1}$$

The authentication time is the time taken to authenticate the MIHF-enabled network entity. Key generation time is the time taken to generate the CK and IK keys from the MSK. Tunnel establishment time is the time taken to populate the IKs, CKs, and MIHF entity MAC address information in the table.

5.3.2. Message Transport Latency. Message transport latency is defined as the time taken to apply the integrity protection or confidentiality protection on the MIH packet that is exchanged between the MIHF entities:

$$\begin{aligned}&\text{Message Transport Latency}\\ &\quad = \text{Time taken to apply protection to MIH packet}.\end{aligned} \tag{2}$$

5.3.3. Message Overhead. Message overhead is the amount of additional information that has to be carried in the MIH packet to carry the message digest. The message digest is carried as a part of TLV in the MIH packet.

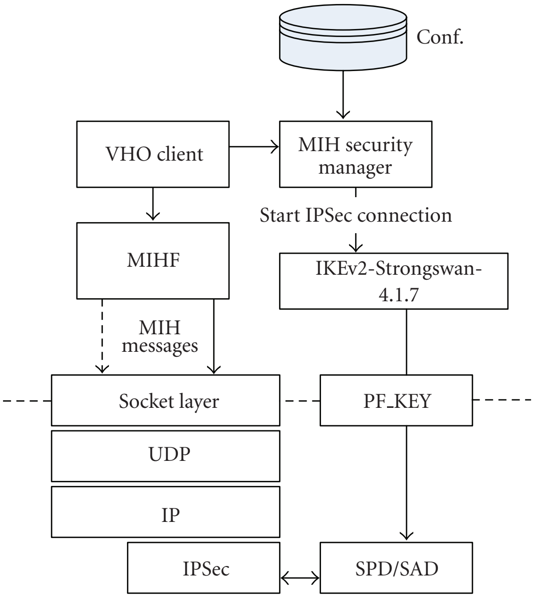

Figure 16: System software architecture in MMT and AR with MIHF and IPSec entities.

6. Prototype Implementation

6.1. Software Architecture for IPSec/IKEv2. Figure 16 shows system software architecture in MMT and AR with MIHF and IPSec functions integrated. The following entities are added to the MIHF/VHO-Client implementation.

Security Configuration Settings. MIHF shall be configured manually to use appropriate security methods (IPSec-IKEv1/v2, encryption/authentication algorithms, etc.).

MIHF Security Manager. The MIHF security manager module shall read the security settings from the configuration file.

It will generate the connection settings (/etc/ipsec.d/mihfsec.conf) dynamically for the new MIHF peer with which the IPSec SA need to be established, reload the settings in IKEv2 daemon, and trigger the IKEv2 daemon to establish IPSec SA with target MIHF peer.

Openssl. The IPSec modules in this solution use the openssl library version 0.9.8 g [9].

The prototype implementation is tested with different security algorithms for encryption and integrity check to measure the latency in handover due to IKEv2 signaling messages as well as MIH message transaction delay added by the security algorithms.

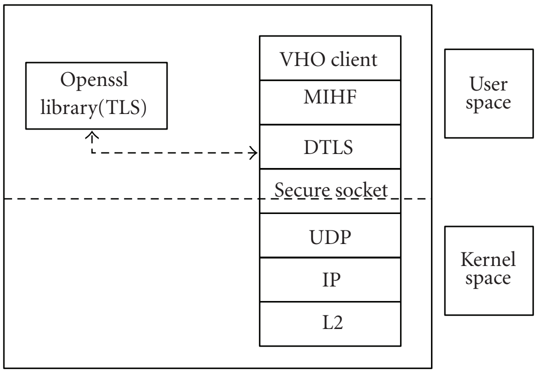

FIGURE 17: System software architecture in MMT/AR with MIHF and DTLS entities.

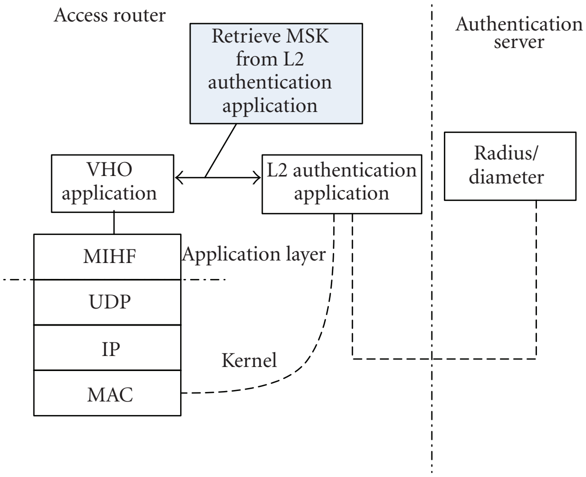

FIGURE 18: Software architecture for MIHSec.

6.2. Software Architecture for DTLS. Figure 17 shows the system software architecture in MMT and AR with MIHF and DTLS functions being integrated. The following entities are added to the MIHF/VHO Client implementation.

Security Configuration Settings. MIHF shall be configured manually to use appropriate security methods (DTLS, encryption/authentication algorithms, etc.).

DTLS. This layer is responsible for enforcing MIH message-transport security. This module creates a DTLS client socket for initiating MIH message exchange with MIH peers and a DTLS server socket which listens to MIHF message from MIH peers. DTLS connection will be established between the peer sockets before any MIH exchange can take place.

Secure Socket Layer. This is implemented using openssl 0.9.8 g library

The DTLS client initiates the communication by sending HELLO SERVER packet by using *SSL_write* API. This initiates the DTLS handshake message sequence, where the messages are processed by the Openssl library. The DTLS client and server authenticate each other, negotiate the algorithms for encryption and integrity, and install the security keys.

Asymmetric key cryptography with RSA (Rivest, Shamir, and Adleman) algorithm is used for authentication between the peer entities.

The client MIH peer sends the all MIH request messages through *SSL_Write* API, which results in the message to be encrypted with the established security key and sent to the server. The server MIH peer decrypts the data and sends to the SSL_read API for passing the message to the MIHF/VHO-Client module.

The prototype implementation is tested with different security algorithms for encryption and integrity check to measure the latency in handover due to DTLS signaling messages as well as MIH message transport delay due to security algorithm processing.

6.3. Software Architecture for MIHSec. Figure 18 shows the software architecture for MIHSec. An AR example is considered to elaborate the architecture concepts. The same design would apply to the MMT also.

MIHF is a media independent handover function. It operates at application layer and interfaces with UDP layer in the kernel, VHO application at the user land space. MIHF handles the event service, command service, and information service messages.

VHO application interacts with MIHF in AR and authentication application. The job of VHO application is to make handover decisions and to maintain the key hierarchy. The key hierarchy is utilized to generate the CKs and IKs for the MIH sessions. The IK and CK are maintained in the table, which is indexed by MAC address of the Peer. The MAC address of the peer acts as a security parameter index for the secure channel.

The changes required to VHO application code for incorporating the MIH security are minimal and hence ease of integration with the current MIH code for providing security enhancements.

A brief patch for the MIH security is provided as follows:

Note that the patch is shown in italics font and current code in regular font.

```
Vho_application_main()
Being:
  New MMT has made an attach with the AR
  Receive MSK from authentication application
  Generate Peer_Key and IS_Key
  Generate CK and IK keys from the key hierarchy
  Maintain the keys information as shown in Figure 20
  ...
  ...
   Decision Process
  ...
```

```
    ...
    etc
End

MIH_LookUp_in AR()
Being:
    Handle the received packet
    Extract MMT-MAC from MIH TLV
    Index into the Key Table (Figure 20) based on Peer-MAC
    Extract CK and IK
    Perform Integrity Check to the packet
    If the integrity check fails, drop the packet
    Else perform the Ciphering Check
    If the ciphering check fails, drop the packet
    Else
    ...
    Perform the normal operations
    ...
    ...
End

MIH_LookUp_In_MMT()
Being:
    Handle the received packet
    Extract MMT-MAC and AR-MAC from MIH TLV
    Index into the Key Table based on Peer-MAC
    Extract CK and IK
    Perform Integrity Check to the packet
    If the integrity check fails, drop the packet
    Else perform the Ciphering Check
    If the ciphering check fails, drop the packet
    Else
    ...
    Perform the normal operations
    ...
    ...
End

MIH_Secure_Packet_Transmission()
Being:
    Decide on packets to be transmitted
    Check the Security YES/NO Flag. If the flag value is NO, transmit the normal MIH the packet (It implies that Security is not mandatory) else
    Index into the key table using ID as identifier to retrieve the IK and CK keys
    (The ID here is MIH Identifier. EAP uses this identifier in it's initial messages for identifying itself with the peer)
    Perform Confidentiality protection on MIH Data
    Perform Integrity Protection on MIH Data and MIH
    Headers (leaving MIH-Integrity TLV header, but including MIH-Encryption TLV header)
    Transmit the security protected packet
End
```

Table 2: TEST configurations.

Security Methods	Settings
IPSec/IKEv2	(1) ESP/Transport, - ENC=3des-cbc/192-bit, AUTH=hmac-md5/128-bit - ESP/Transport, ENC=aes-cbc/256-bit, AUTH=hmac-sha1/128-bit (2) IKEv2 Settings: - Strongswan 1.4.7 daemon [10] - X.509 certificates with RSA (1024-bit private key)
DTLS	- Openssl 0.9.8g library [11] - X.509 certificates with RSA (1024-bit private key)
MIHSec	- EAP Protocol for Authentication - Extensions to 802.21 to support MIHSec in MIHF

The security keys are maintained as shown in Figure 19.

On receiving the MSK from authentication application, this table is configured by VHO application.

A provision could be provided to configure this table manually. However, at present, this option is not being considered and could be investigated later.

7. Experimental Results

7.1. Test Environment. Figure 20 illustrates the test environment used for testing the prototype implementations with the test configuration in Table 2.

7.2. Test Settings for Security Methods. The IPSec/IKEv2 connection settings for PAR (and IS server) are statically configured, while the connection settings for NAR are dynamically generated.

EAP stack integration with MIHF is performed to enable EAP to carry MIHF identifier as EAP identifier. MIHF is extended to support security headers.

ID of MN1	MAC of MN1	Integrity key 1	Cipher key 1	Peer_Key-1 IS_Key-1	Security (Yes/No) flag
ID of MN2	MAC of MN2	Integrity key 2	Cipher key 2	Peer_Key-2 IS_Key-2	Security (Yes/No) flag
ID of MN3	MAC of MN3	Integrity key 3	Cipher key 3	Peer_Key-3 IS_Key-3	Security (Yes/No) flag

Figure 19: Format of security keys table in AR.

Table 3: Result analysis and comparison summary.

Parameters	Security method		
	IPSec/IKEv2	DTLS	MIHSec
Security signaling latency	High (100 s of msec)	Moderate (10 s of msec)	Low (1 s of msec)
Message transport latency	Negligible	Negligible	Negligible
Configuration and setup	Difficult	Moderate	Easy
Message overhead per MIH exchange	Less than 25%	Above 25%	Less than 25%

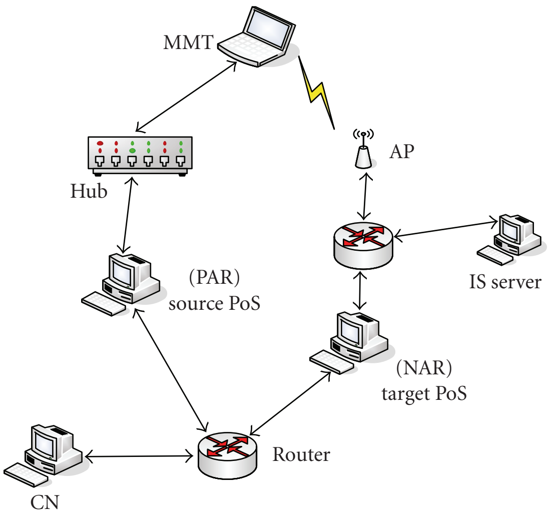

Figure 20: Environment for MIH security prototype testing.

7.3. Result Analysis. Test results are analyzed and compared with respect to Signaling latency, Message transport latency, Message overhead, Configurations, and setup. The results are shown in Table 3.

7.3.1. Security Signaling Latency. In the test setup we used, it is found that IPSec/IKEv2 takes about 230 milliseconds for IKEv2 signaling involving 2 IKE_SA_INIT messages and 2 IKE_AUTH messages and key generation /installation.

In the DTLS method, only about 30 milliseconds are taken for signaling and installation of security keys.

In the MIHSec method, since the authentication is directly integrated with L2 authentication, it leads to an efficient security signaling time.

7.3.2. Message Transport Latency. Our experimental results showed that IPSec transforms (Encryption and Decryption) do not add much confidentiality latency to MIH message exchange. In this experiment we used general purpose machines with security algorithms implemented in software. In more practical scenarios, sophisticated hardware will be used for implementing security algorithms and then the latency will be negligible.

Our experimental results showed that DTLS transforms (Encryption and Decryption) do not add much confidentiality latency to MIH message exchange.

The latency of the message transport in MIHSec is comparable to the IPSec and DTLS latency times.

7.3.3. Message Overhead. To an MIH message exchange (Request and Response), about 70 bytes are added as overhead in 3des-cbc/192-bits and about 90 bytes are added as overhead in the case aes-cbc/256-bit. This is applicable in both IPSec and MIHSec case.

To an MIH message exchange, about 100 bytes are added as overhead in the case of DTLS based message protection.

7.3.4. Configuration and Setup. MIHF configuration for IPSec/IKEv2 is fairly complex. This is due to the fact that the IKE is inherently a key authentication protocol with complex configurations, and which expects peer to configure the security information in advance. In addition when the handovers are performed, with the changes in IP address to the mobile terminal, configuration becomes a challenging task in IPSec/IKEv2. When end-to-end secure tunnels are used (as in this experimented), the MIHF should be configured to establish IPSec SA with each end-point. Manual configuration of this is impractical.

Also the IPSec support is required in the kernel.

DTLS is an Application Layer protocol and the DTLS Client/Server requires lesser configuration effort.

However, the use of the following approach will simplify configuration process.

(1) Use MIH Discovery method for automatically discovering the target MIH endpoint.

(2) Use MIH ID as the unique identifier to generate X.509.

Configurations that are made for the L2 security should be sufficient for the MIHSec. No additional configurations are required for MIHSec, hence simplifying configuration operations when compared to DTLS or IKEv2/IPSec.

8. Conclusion

This paper analyses different security methods which could be used for MIH message protection. Prototype of MIH security methods with IPSEc/IKEv2, DTLS, and MIHSec methods are developed and the results are compared. The experiments showed better results in terms of message overhead for MIHSec and IPSec methods compared to DTLS. However in terms of signaling latency, MIHSec showed better results. Also, since the MIH messages are transported over UDP (in this implementation of MIHF), security at transport layer might be sufficient, and hence MIHSec method is a strong candidate. We have presented numerical results to show that 802.21 with MIHSec security extensions provides good handover latency, compared to DTLS and IPSec. This shows that MIHSec is a better solution to support secure MIH message transport.

Acknowledgments

This work was supported by the National Research Foundation of Korea Grant funded by the Korean Government (KRF-2008-D00273-101053) and V. Leung's work was supported by Grants from TELUS and the Canadian Natural Sciences and Engineering Research Council under Grant CRDPJ 341254-06.

References

[1] LAN MAN Standards Committee of the IEEE Computer Society, Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services, IEEE P802.21/D11.00, May 2008.

[2] E. Gustafsson and A. Jonsson, "Always best connected," *IEEE Wireless Communications*, vol. 10, no. 1, pp. 49–55, 2003.

[3] G. Lampropoulos, A. K. Salkintzis, and N. Passas, "Media-independent handover for seamless service provision in heterogeneous networks," *IEEE Communications Magazine*, vol. 46, no. 1, pp. 64–71, 2008.

[4] A. Dutta, D. Famolari, S. Das, et al., "Media-independent pre-authentication supporting secure interdomain handover optimization," *IEEE Wireless Communications*, vol. 15, no. 2, pp. 55–64, 2008.

[5] S. Kent and K. Seo, "Security architecture for the Internet protocol," RFC 4301, December 2005.

[6] E. Rescorla and N. Modadugu, "Datagram transport layer security," RFC 4347, April 2006.

[7] S. Kent and R. Atkinson, "Security architecture for the Internet protocol," RFC 2401, November 1998.

[8] C. Kaufman, Ed., "Internet key exchange (IKEv2) protocol," RFC 4306, December 2005.

[9] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz, "Extensible authentication protocol (EAP)," RFC 3748, June 2004.

[10] Strongswan, "The OpenSource IPsec-based VPN solution for Linux," http://strongswan.org.

[11] OpenSSL Project, http://www.openssl.org.