Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	**Privacy key management for BSs and BSISs in 802.16 LE Systems**
Date Submitted	**2005-09-14**
Source(s)	Hung-Lin Chou, Chi-Chen Lee, Industrial Technology Research Institute, Computer and Communications Research Labs, Taiwan Bldg. 11, 195 Sec. 4, Chung Hsing Rd. Chutung, HsinChu, Taiwan 310, R.O.C. Voice: +886-3-5912042 Fax: +886-3-5829733 mailto: hunglinchou@itri.org.tw
Re:	Call for Contributions, IEEE 802.16h Task Group on License-Exempt Coexistence, IEEE 802.16h-05/014, 2005/06/09
Abstract	Propose the PKM protocol for intercommunications in 802.16 LE.
Purpose	Provide PKM procedures to enhance the security connection between BS and BS/BSIS
Notice	This document has been prepared to assist IEEE 802.16. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE's name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE's sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16.
Patent Policy and Procedures	The contributor is familiar with the IEEE 802.16 Patent Policy and Procedures <http://ieee802.org/16/ipr/patents/policy.html>, including the statement "IEEE standards may include the known use of patent(s), including patent applications, provided the IEEE receives assurance from the patent holder or applicant with respect to patents essential for compliance with both mandatory and optional portions of the standard." Early disclosure to the Working Group of patent information that might be relevant to the standard is essential to reduce the possibility for delays in the development process and increase the likelihood that the draft publication will be approved for publication. Please notify the Chair <mailto:chair@wirelessman.org> as early as possible, in written or electronic form, if patented technology (or technology under patent application) might be incorporated into a draft standard being developed within the IEEE 802.16 Working Group. The Chair will disclose this notification via the IEEE 802.16 web site <http://ieee802.org/16/ipr/patents/notices>.

Privacy Key Management for BSs and BSISs in 802.16 LE Systems

Hung-Lin Chou, Chi-Chen Lee
Computer & Communications Research Labs, ITRI, Taiwan

1. Introduction

This document proposes an enhanced network architecture which is distributed and more flexible. Besides, the related Privacy Key Management protocol for 802.16 LE systems is also introduced.

2. Background

In session#37, the architecture proposed in [1] was accepted. However, the accepted architecture requires a regional centralized RADIUS server and BSIS(in session #38, the CIS(Coexistence Identification Server) is renamed to BSIS(Base Station Identification Server)) which may lack for flexibility and scalability. Moreover, it is more reasonable that each operator has its own RADIUS server for authentication. This proposal intends to reduce the key management complexity of the RADIUS server and the maintenance overhead of BSIS. In the enhanced architecture, only the global RADIUS server (here may be more than one global server) called "root" RADIUS server remains and the BSISs will be distributed. All RADIUS servers and BSISs of the 802.16 LE operators shall register IP addresses of RADIUS servers and BSISs as well as the country code of the operator to the root RADIUS server(s). An 802.16 LE system learns other existing 802.16 LE systems by querying the root RADIUS server(s) using its own country code and neighboring country code.
A new re-key mechanism is proposed, as the previous re-key procedures rely on Radius-Server to generate security blocks and Security Parameters Index (SPIs) (a field of ESP header which identifies the security parameters in combination with IP address) and Keys for the BSs/BSISs. The loading of SPIs/Keys update of Radius-Server will be an issue as the number of BSs increases. For multiple Radius-Servers environment, the new PKM protocol provides an easier way to regenerate the session-key that secures the communications between BSs/BSISs based on the Master-Key, which is for generating session-key. The original IAPP-based solution relies on RADIUS Server to keep security information parameters and BSs mapping (ex: SPI、Security Association and Supporting Transform/Authentication Algorithm…etc). While BSs need to re-key or to create a new SPI/SA, RADIUS Server must involve and handle message exchange between BSs/BSIS. The proposed mechanism resolves the issue of SPIs/Keys mapping in multiple RADIUS-Servers environment by avoid the SPI/SA mapping, i.e. the RADIUS Server will not involve the re-key procedures.

In session#38, we discuss the different security issues of 802.16e and 802.16h. For 802.16e, the encrypted data packets just transmit between SS and BS in wireless interface, and the authentication/authorization/accounting procedures adopt EAP (Extensible Authentication Protocol). For 802.16h, it needs different secure thinking for packet passing through different network equipments (ex: routers/firewalls). IPSec is a common secure connection solution for IP-network and also applied to IPv4 and IPv6 environment. General firewalls also know how to check the header of IPSec packet (ESP header) and have the filtering rules to decide whether the IPSec packets could be allowed to pass through or not.

Acronyms

BSIS – Base Station Identification Server

PKM – Private Key Management

IPsec – Internet Protocol Security

ESP – IP Encapsulating Security Payload

AH – Authentication Header

3. Suggested remedy

(1) Proposed enhancement of general architecture for inter-network communication

[insert the following section into 2.1.2.1 Architecture]

Considering the IP network firewalls and different filtering rules, we should find a common security solution to make BSs/BSISs data connection transparent under almost common network management cases. IPSec is used to IPv4 and also included in IPv6 for the IP-Layer security solution. And all BSs/BSISs don't just reside in the same network environment. The data connections should go through some routers/firewalls and need to follow a common security rules.

Figure 1 shows the BSs/BSISs connections encrypted in IPSec. Based on IPSec, all data connections between BSs/BSISs could pass through firewalls and routers unless some firewalls block IPSec connections.

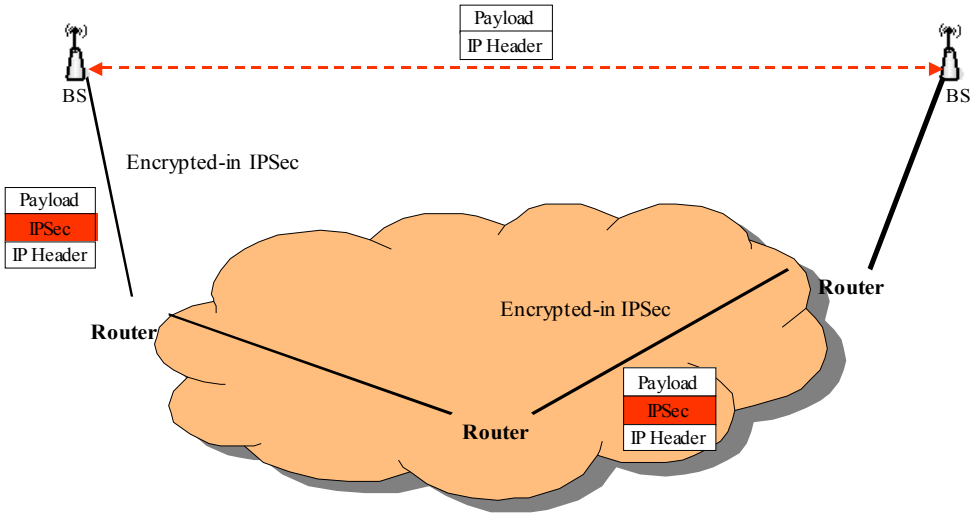

Figure 1 BSs/BSISs connection encrypted in IPSec

Figure 2 demonstrates the IEEE 802.16 LE inter-network communication architecture under multi-Operators with multi-RADIUS Servers.

If BS-1 wants to communicate with BS-2, it must get BS-2's Country's Code, Operator ID and BSID from local BSIS first. And then work as the following steps

(1)BS-1 send RADIUS-Access-Request frame with BS-2's Country's Code, Operator ID and BSID to local RADIUS-Server
(2)Local RADIUS-Server will act as RADIUS-Proxy and transfer this RADIUS-Access-Request to the target RADIUS-Server
(3)Target RADIUS-Server will response RADIUS-Access-Accept with Pairwise-Master-Key and PMK-index for BS-1 and Security-Block for BS-2
(4)Local RADIUS-Server will generate Security-Block including Pairwise-Master-Key and PMK-index from target Raidus-Server

(5)BS-1 will receive RADIUS-Access-Accept from its local RADIUS-Server and get the Pairwise-Master-Key、 PMK-index and ESP Authentication/Transform IDs in Security-Block for BS-1
(6)BS-1 will act as a PKM-initiator to send Session-Key-Start to BS-2 with Security-Block for BS-2
(7)BS-2 will calculate the ESP-Key-Stuffs with Pairwise-Master-Key, choose the ESP Authentication/Transform IDs supported by BS-2 and response Session-Key-Request to BS-1
(8)BS-1 will also calculate the ESP-Key-Stuffs with Pairwise-Master-Key to verify Key-Signature, compare ESP Authentication/Transform IDs support by BS-2 with current settings supported by BS-1 and response Session-Key-Response to BS-2
(9)BS-2 will verify Key-Signature and response Session-Key-Accept to BS-1
(10)After the above procedures, BS-1 and BS-2 could communicate in IPSec with the ESP-Key-Stuffs generated dynamically

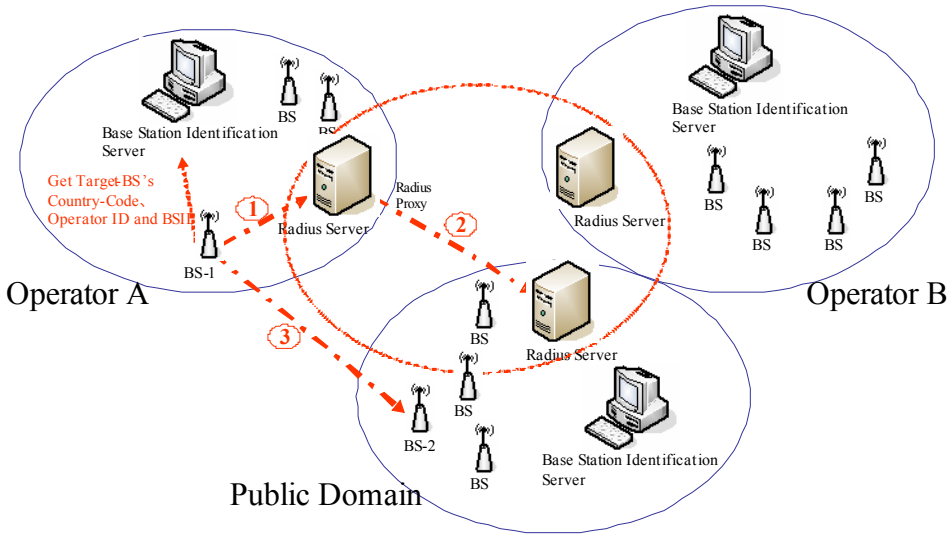

Figure 2 Network Architecture under multi-Operators with multi-RADIUS Servers

The following figure shows the each connection of BSs/BSISs will be encrypted in individual Session-Key in IPsec

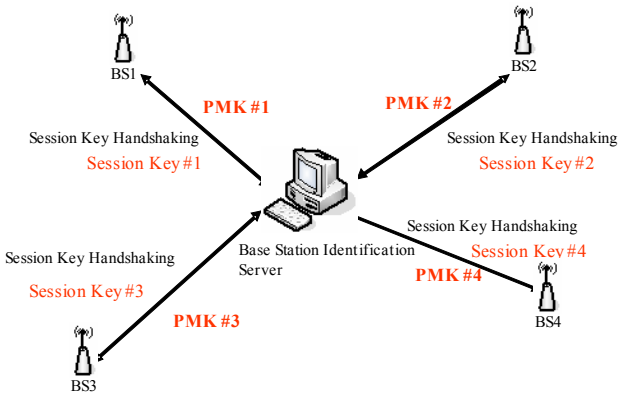

Figure 3 Individual Session-Key

For the BSs/BSISs, each connection with different BSs/BSISs will use individual Session-Key in IPsec. Those Session Keys would be generated from PKM-Handshaking with Pairwise-Master-Keys between each pair BSs/BSISs. The re-key procedures also don't need RADIUS-Servers and just use Pairwise-Master-Keys.

(2) Proposed enhancement of RADIUS protocol usage

[delete original text of section 3.2.4.2 and replace with the following text]

3.2.4.2 RADIUS Protocol usage

For future interoperability consideration, similar mechanisms in [2] are maintained. Secure exchange of 802.16 LE signaling information can be achieved after successful procedures of the RADIUS protocol. To include RADIUS support, the RADIUS server and the BS/BSIS RADIUS client must be configured with the shared secret key and with each other's IP address. Each BS/BSIS acts as a RADIUS client and has its own shared secret key with the RADIUS server. The shared secret key may be different from that of any other BS/BSIS.

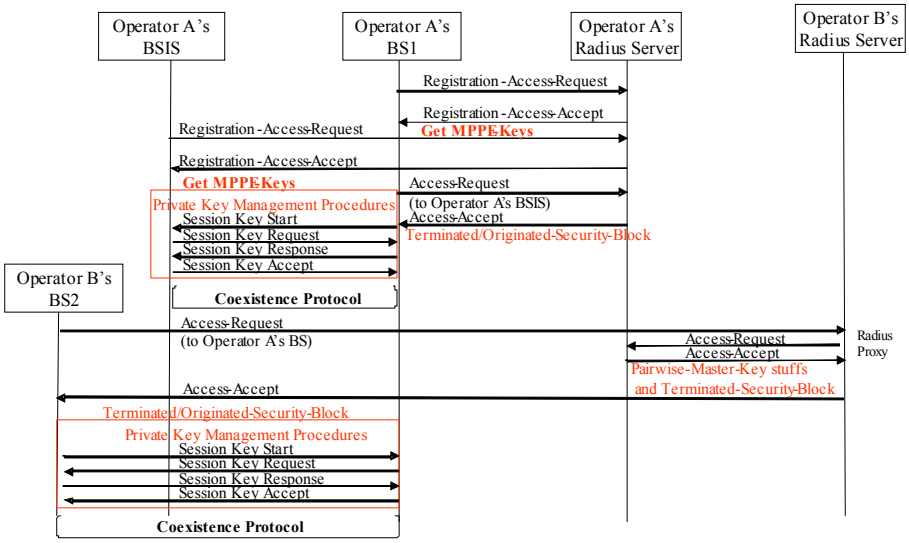

Figure 4 RADIUS protocol example

Figure 4 shows the RADIUS protocol message exchange sequence. At starting up, each BS or BSIS must send a RADIUS-BS/BSIS-Registration-Access-Request (shown in table 2) to the RADIUS server for authentication purpose and leave the address mapping (BSID to IP) information in the server. At this time, the RADIUS server will retain the following information of registered BS or BSIS:

(a) Wireless medium address of BS (BSID) or medium address of BSIS,
(b) MPPE-Keys in RADIUS-BS/BSIS-Registration-Access-Request/Accept Procedures
(c) IP address or DNS name,
(d) *Cipher suites supported by the BS or BSIS for the protection of Coexistence Protocol communications,*
(e) *and Pairwise-Master-Key for BS or BSIS to establish Session-Key-Handshaking procedures*

Same as [2], Microsoft Point-to-Point Encryption (MPPE) (RFC 2548:1999) key is introduced. The MS-MPPE-Send-Key, which could be got in the RADIUS-BS/BSIS-Registration-Access-Accept message (shown in table 3) and RADIUS-BS/BSIS-Access-Accept message (shown in table 5), is used for encrypting the security blocks in the RADIUS-BS/BSIS-Access-accept message for PKM-target and PKM-initiator. A registration access reject message may be issued due to a BS not supporting the ESP Transform or ESP Authentication algorithm selected for use in securing the following intercommunication, or for other RADIUS configuration reasons not discussed here.

Once a BS wants to get the knowledge of neighbor topology, it must first send RADIUS-BS/BSIS-Access-Request message (shown in table 4) to the RADIUS server in order to acquire the regional BSIS's IP address. The wireless medium addresses of regional BSIS, similar to BSID, well known by all BSs supporting LE operation, is sent in the RADIUS-BS/BSIS-Access-Request message to the RADIUS server for looking up IP address of the BSIS. Upon receiving the request message, the RADIUS server will respond with a RADIUS-BS/BSIS-Access-Accept message (shown in table 5) if the BS is a valid member which is allowed to perform inter-communication. The RADIUS-BS/BSIS-Access-Accept message would contain Originated-BS-Security-Block(for BS encrypted in MPPE-Send-Key from current RADIUS-BS/BSIS-Access-Request/Accept message) and Terminated-BS/BSIS-Security-Block(for BSIS encrypted in MPPE-Send-Key from BSIS's RADIUS-BS/BSIS-Registration-Access-Request/Accept message). Security-Block (shown in table 1) contains Pairwise Master Key Index、Pairwise-Master-KEY、Key Lifetime、the list of ESP Authentication/Transform IDs for initiator-send/receive for establishing a secure connection with the BSIS .

After querying process between the BS and the regional BSIS in Coexistence Protocol, the BSIS will respond to the BS with possible neighbor BSs candidates and their BSIDs. The BS, then, tries to establish secure connections with the neighbor BSs after evaluating the coexistence relationships with these candidates. The BS sends RADIUS-BS/BSIS-Access-Request message to local RADIUS server for Originated/Terminated-BS/BSIS-Security-Blocks. After getting Security-Blocks from RADIUS-BS/BSIS-Access-Accept messages, the BS establishes secure connections with each evaluated neighbor BS.

An access reject message may be issued due to a BS or the regional BSIS not supporting the ESP Transform or ESP Authentication algorithm selected for the following intercommunication, or for other RADIUS configuration reasons not discussed here.

Table 1 Security Block Format

Element ID	*Length*	*Information*
1	*1*	*Pairwise Master Key Index for BS/BSIS (0-255)*
2	*32*	*Pairwise-Master-KEY*
3	*4 * number*	*The list of ESP Authentication IDs corresponding to the ESP Authentication algorithms for initiator-send*
4	*4 * number*	*The list of ESP Transform IDs corresponding to the ESP transforms for initiator-send*
5	*4 * number*	*The list of ESP Authentication IDs corresponding to the ESP Authentication algorithms for initiator-receive*
6	*4 * number*	*The list of ESP Transform IDs corresponding to the ESP transforms for initiator-receive*
7	*4*	*Pairwise-Master-KEY Lifetime*

The Security-Block would be encrypted in 32-bytes MPPE-Send-Key with the following manner ('+' indicates

concatenation):

b(1) = MD5(MPPE-Send-Key+BSID) c(1) = p(1) xor b(1) C = c(1)
b(2) = MD5(MPPE-Send-Key+BSID + c(1)) c(2) = p(2) xor b(2) C = C + c(2)
. .
. .
. .
b(i) = MD5(MPPE-Send-Key+BSID + c(i-1)) c(i) = p(i) xor b(i) C = C + c(i)

Break plain text into 16 octet chunks p(1), p(2)...p(i), where i = len(P)/16. Call the ciphertext blocks c(1), c (2)...c(i) and the final ciphertext C. Intermediate values b(1), b(2)...c(i) are required. The resulting encrypted String field will contain c(1)+c(2)+...+c(i).

For Originated Security Block, the encrypted MPPE-Send-Key is from "RADIUS-Access-Request/Accept".For Terminated Security Block, the encrypted MPPE-Send-Key is from "RADIUS-Registration-Access-Request/Accept".

(3) Proposed RADIUS protocol messages

[delete original text of section 6.3 and replace with the following text]

6.2 RADIUS protocol messages

The following messages are listed to support RADIUS protocol:
Note that TBD means To Be Defined.

- *RADIUS-BS/BSIS-Registration-Request (BS/BSIS → RADIUS server): A startup BS/BSIS sends this message for authentication purpose.*

Table 2 RADIUS-BS/BSIS-Registration-Access-Request

Attribute number	***Attribute name***	***Value***
1	*User-Name*	*BSID. The BSID should be represented in ASCII format, with octet values separated by a "-". Example: "00-10-A4-23-19-C0".*
4	*NAS-IP-Address*	*BS's IP Address*
6	*Service-Type*	*Coexistence-Protocol-Register (value = TBD, ex. IAPP-Register, value = 15)*

26	*Vendor-Specific-Attribute (VSA)*	
26-TBD	*Supported-ESP-Authentication-Algorithms*	*The list of ESP Authentication IDs corresponding to the ESP Authentication algorithms supported by this BS (See Table 6)*
26-TBD	*Supported-ESP-Transforms*	*The list of ESP Transform IDs corresponding to the ESP transforms supported by this BS (See Table 5)*
32	*NAS-Identifier*	*BS's NAS Identifier*
80	*Message-Authenticator*	*The RADIUS message's authenticator*

According to RFC 2865:2000, other RADIUS attributes may be included in the RADIUS-BS/BSIS-Registration-Access-Request packet in addition to the ones listed in Table 2.

- *RADIUS-BS/BSIS-Registration-Accept (RADIUS server → BS/BSIS): After RADIUS server verifies the valid membership, it will respond with this accept message.*

Table 3 RADIUS-BS/BSIS-Registration-Access-Accept

Attribute number	***Attribute name***	***Value***
1	*User-Name*	*BSID.*
6	*Service-Type*	*Coexistence-Protocol -Register (value = TBD, ex. IAPP-Register, value = 15)*
26	*Vendor-Specific-Attribute (VSA)*	
26-TBD	*Supported-ESP-Authentication-Algorithms*	*The list of ESP Authentication IDs corresponding to the ESP Authentication algorithms approved by Radius Server*
26-TBD	*Supported-ESP-Transforms*	*The list of ESP Transform IDs corresponding to the ESP transforms approved by Radius Server*
27	*Session-Timeout*	*Number of seconds until the BS should re-issue the registration Access-Request to the RADIUS server to obtain new key information.*
80	*Message-Authenticator*	*The RADIUS message's authenticator*

According to RFC 2865:2000, other RADIUS attributes may be included in the RADIUS-BS/BSIS-Registration-Access-Accept packet in addition to the ones listed in Table 3.

- *RADIUS-BS/BSIS-Access-Request (BS/BSIS → RADIUS server): The BS sends this message to request for inter-communication with another neighbor BS or a regional BSIS.*

Table 4 RADIUS-BS/BSIS- Access-Request

Attribute number	*Attribute name*	*Value*
1	*User-Name*	*User-Name must include Country-Code、Operator ID and Regional BSIS's WM address or neighbor BS's BSID*
4	*NAS-IP-Address*	*Original BS's IP Address (the BS sending this request message)*
6	*Service-Type*	*CS/CIS-Check (value = TBD, ex. IAPP-AP-Check, value = 16)*
61	*NAS-Port-Type*	*Wireless – Other (value = 18)*
80	*Message-Authenticator*	*The RADIUS message's authenticator*

According to RFC 2865:2000, other RADIUS attributes may be included in the RADIUS-BS/BSIS-Access-Request packet in addition to the ones listed in Table 4.

- *RADIUS-BS/BSIS-Access-Accept (RADIUS server → BS/BSIS): After verifying that the neighbor BS is valid member, RADIUS server will respond with the security blocks necessary for establishing a secure connection between the neighbor BS and requesting BS or between BSIS and requesting BS.*

Table 5 RADIUS-BS/BSIS- Access-Accept

Attribute number	*Attribute name*	*Value*
1	*User-Name*	*User-Name must include Country-Code、Operator ID and Regional BSIS's WM address or neighbor BS's BSID*
8	*Framed-IP-Address*	*IP Address of Regional BSIS or neighbor BS.*
26	*Vendor-Specific-Attribute (VSA)*	
26-TBD	*Originated-BS-Security-Block*	*Security Block encrypted using originated BS's MPPE-SEND-KEY, to be decrypted and used by the original BS*
26-TBD	*Terminated-BS/BSIS-Security-Block*	*Security Block encrypted using neighbor BS's MPPE-SEND-KEY (or BSIS's), to be decrypted and used by the neighbor BS (or BSIS)*
80	*Message-Authenticator*	*The RADIUS message's authenticator*

According to RFC 2865:2000, other RADIUS attributes may be included in the RADIUS-BS/BSIS-Access-Accept packet in addition to the ones listed in Table 5.

Table 6 ESP Transform identifiers

Transform identifier	*Value*	*Reference*
RESERVED	*0*	*[RFC2407]*
ESP_DES_IV64	*1*	*[RFC2407]*
ESP_DES	*2*	*[RFC2407]*
ESP_3DES	*3*	*[RFC2407]*
ESP_RC5	*4*	*[RFC2407]*
ESP_IDEA	*5*	*[RFC2407]*
ESP_CAST	*6*	*[RFC2407]*
ESP_BLOWFISH	*7*	*[RFC2407]*
ESP_3IDEA	*8*	*[RFC2407]*
ESP_DES_IV32	*9*	*[RFC2407]*
ESP_RC4	*10*	*[RFC2407]*
ESP_NULL	*11*	*[RFC2407]*
ESP_AES-CBC	*12*	*[RFC3602]*
Reserved for privacy use	*249-255*	*[RFC2407]*

Table 7 ESP Authentication algorithm identifiers

Transform identifier	*Value*	*Reference*
RESERVED	*0*	*[RFC2407]*
HMAC-MD5	*1*	*[RFC2407]*
HMAC-SHA	*2*	*[RFC2407]*
DES-MAC	*3*	*[RFC2407]*
KPDK	*4*	*[RFC2407]*
HMAC-SHA2-256	*5*	*[Leech]*
HMAC-SHA2-384	*6*	*[Leech]*
HMAC-SHA2-512	*7*	*[Leech]*
HMAC-RIPEMD	*8*	*[RFC2857]*
RESERVED	*9-61439*	
Reserved for privacy use	*61440-65535*	

(4) Proposed enhancement of Privacy Key Management protocol usage

[add the new section 3.2.4.4 Privacy Key Management protocol usage with the following text]

The PKM protocol would provide a flexible and easy-to-maintain key exchange mechanism. The PKM is based on the Pairwise-Master-Key to provide a symmetric key for the PKM-Initiator and PKM-Target side.

The following figure shows the PKM Session-Key-Handshaking procedures

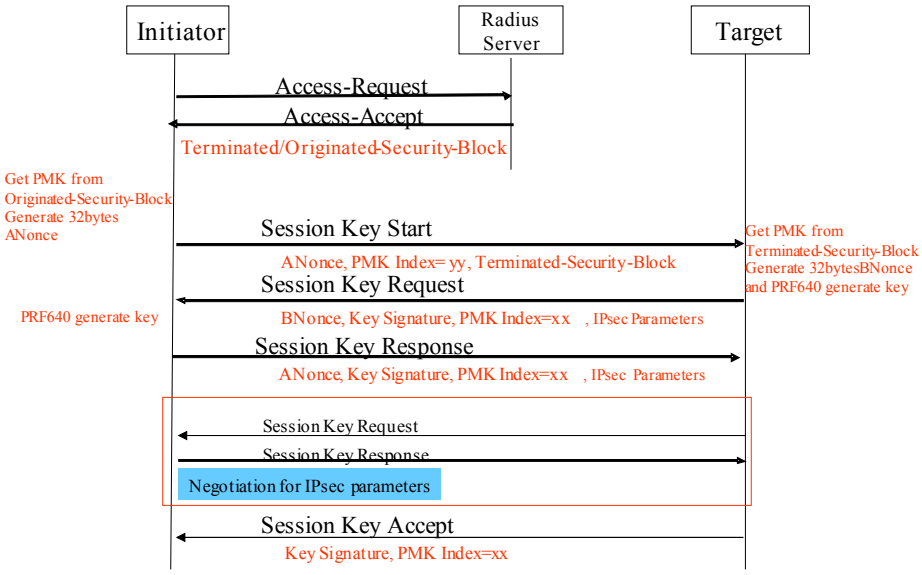

Figure 5 PKM Session-Key-Handshaking procedures

The PKM-Initiator will need to get the Pairwise-Master Key in Originated-BS-Security-Block from RADIUS-Server. And then perform the following steps

(1) *PKM-Initiator would get Pairwise-Master-Key-Index、Pairwise -Master-Key、ESP Authentication/Transform IDs and Key-Lifetime in originated Security-Block in RADIUS-BS/BSIS- Access-Accept message and then generate a random 32-bytes ANonce.*

(2) *PKM-Initiator would will send Session-Key-Start message to PKM-Target with "ANonce"、"Pairwise-Master-Key-Index" and "Terminated Security-Block".*

(3) *After receiving Session-Key-Start message, PKM-Target would generate a random 32-bytes BNonce. And perform the PRF640 algorithm to generate the 640-bits Key. Keep the first 512-bits ESP-Transform/Authentication Keys and use the last 128-bits M-Key as the HMAC-MD5 key to generate 16-bytes Key-Signature.*

(4) *PKM-Target would will send Session-Key-Request message to PKM-Initiator with "BNonce"、"Pairwise-Master-Key-Index" and " ESP Authentication/Transform IDs"(PKM-Target chosen).*

(5) *After receiving Session-Key-Request message, PKM-Initiator would perform the PRF640 algorithm to*

generate the 640-bits Key. Keep the first 512-bits ESP-Transform/Authentication Keys and use the last 128-bits M-Key as the HMAC-MD5 key to generate 16-bytes Key-Signature to verify the Key-Signature field on the Session-Key-Request message. If it is wrong, PKM-Initiator would perform silent-drop and doesn't response any message. If it is correct, PKM-Initiator would prepare the Session-Key-Response message and use HMAC-MD5 generate Key-Signature filed.

(6) *PKM-Initiator would will send Session-Key-Response message to PKM-Target with "ANonce"、"Pairwise-Master-Key-Index" and " ESP Authentication/Transform IDs"(PKM-Initiator chosen) .*

(7) *After receiving Session-Key- Response message, PKM-Target would check the ANonce value if equal to the previous ANonce value in Session-Key-Start message and use HMAC-MD5 generate Key-Signature filed to verify the Key-Signature field. Compare the values of "ESP Authentication/Transform IDs" to make sure the security parameters.*

(8) After the above, PKM-Target will send Session-Key-Accept with Key-Signature filed to PKM-Initiator to verify.

(9) *The following IPsec connection will use the first 512-bits ESP-Transform/Authentication Keys from PRF640 as keys and perform the ESP-Transform/Authentication algorithms from chosen ESP Authentication/Transform IDs.*

The following figure shows the PKM Session-Key Re-Key procedures

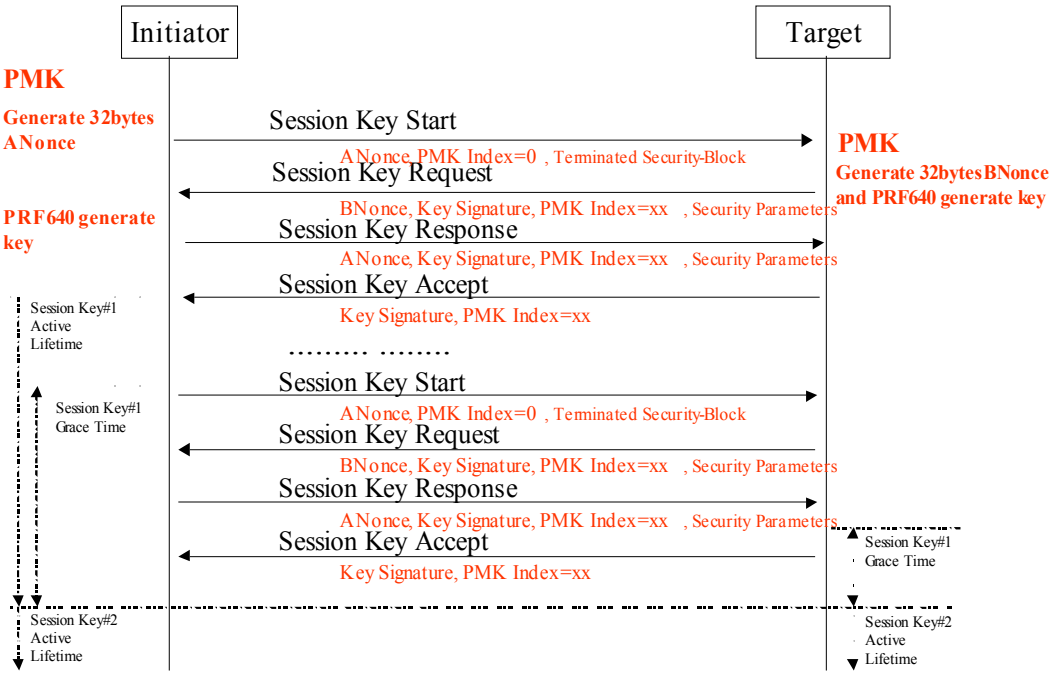

Figure 6 PKM Session-Key Re-Key procedures

Each Session-Key would set a Key-Lifetime, and PKM-Initiator could set a Session-Key grace time to perform Session-Key-Handshaking for the next new Session-Key#2 to be generated until the end of the key lifetime. The Session-Key#1 could use up its lifetime and then activate the Session-Key#2. If each side use the Session-Key#2 first in IPsec connection, it could also activate the Session-Key#2. If the lifetime of Session-Key#1 use up, the PKM-Initiator doesn't perform the Session-Key Re-Key procedures. PKM-Target would disconnect the IP connection until the Session-Key#2 generated.

The following figure shows the PKM Session-Key Re-Key procedures with the PMK update

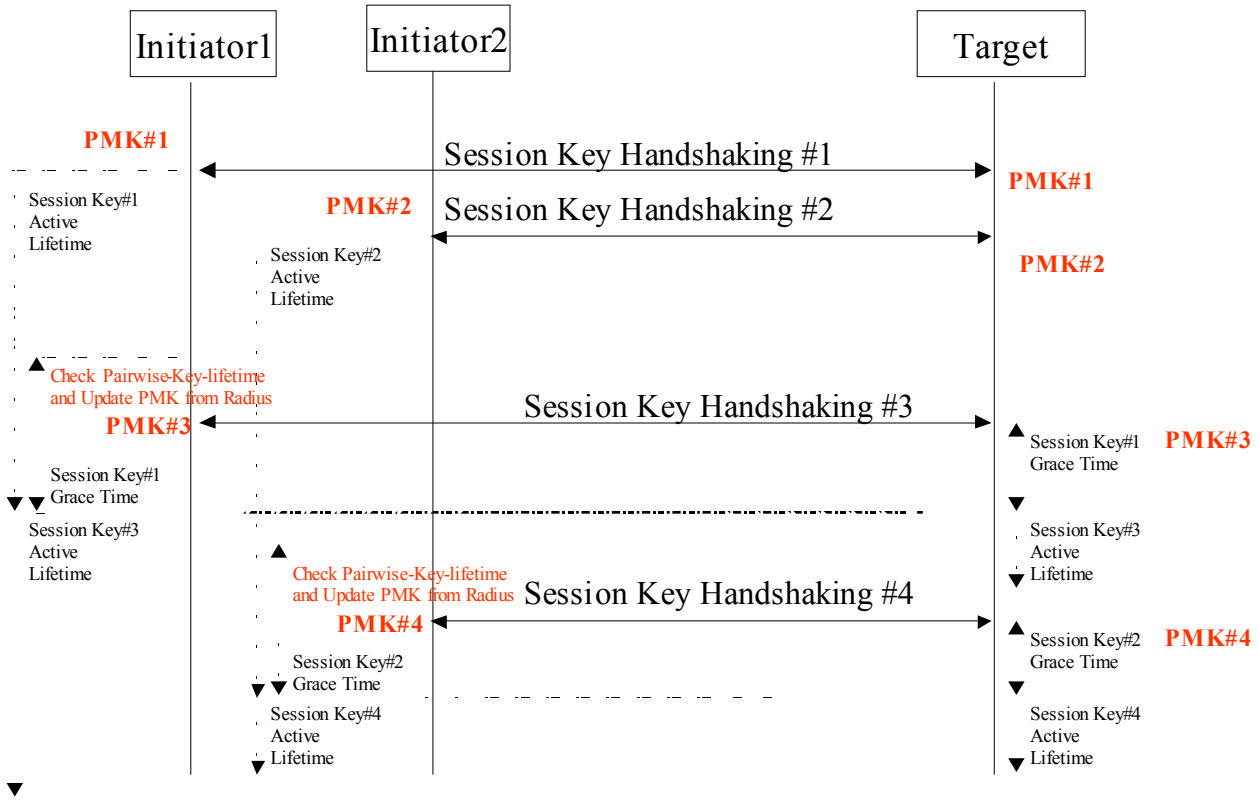

Figure 7 PKM Session-Key Re-Key procedures with the MK update of PKM-Target

The PKM-Initiator will check the current Pairwise-Key-Lifetime if still valid. If the PKM-Initiator detects the Pairwise-Key-Lifetime used up, it would perform RADIUS-BS/BSIS- Access-Request/Accept procedures to get the latest Pairwise-Master-Key in Security-Blocks from RADIUS-Server.

Each Pairwise-Master-Key would set a Pairwise-Master-Key-Lifetime, and BSs/BSISs could set a Pairwise-Master-Key grace time to perform Access-Request/Accept procedures for the new Pairwise-Master-Key until the end of the Pairwise-Master-Key lifetime. If the lifetime of Pairwise-Master-Key use up, the originated

BSs/BSISs don't perform the Access-Request/Accept procedures, the terminated BSs/BSISs should discard the connections.

The following figure shows the 640-bits Key generated by PRF640

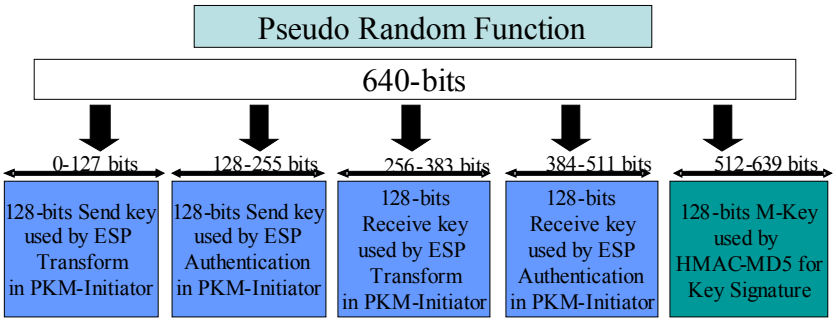

Figure 8 the 640-bits Key generated by PRF640

The BSs/BSISs get Pairwise-Master-Key from RADIUS-Servers and generate 32-bytes Nonce value to derive 640-bits key as follows

PRF-640(PMK, "BS-BSIS key expansion", Min(BS1ID,BS2ID) || ***Max (BS1ID,BS2ID)*** || ***Min (ANonce,BNonce)*** || ***Max(ANonce,BNonce))***

Where

PRF-640 (K,A,B) =
for i=0 to 4 do
R=R|HMAC-SHA-1(K, A|0|B|I)
return LeastSignificant-640-bits(R)

and "|" denotes bitstring concatenation

(5)Proposed Privacy Key Management protocol messages

[insert the following section into 6.3 Privacy Key Management protocol messages]

The PKM protocol procedures contain 4 message actions, and each-side could check the code value of the begin of PKM message to recognize which action need to perform this moment. The meaning of codes for PKM message as follows

- *0 = Session Key Start*
- *1 = Session Key Request*
- *2 = Session Key Response*
- *3 = Session Key Accept*

The PKM message uses TLV format to add the following attributes

Table 8 Session Key frame TLV

Type	*Length*	*Value Information*
1	*32*	*Nonce*
2	*8*	*Replay Counter*
3	*8*	*Key lifetime in seconds*
4	*16*	*Key Signature*
5	*4*	*Security Parameter Index*
6	*4 * number*	*The list of ESP Authentication IDs corresponding to the ESP Authentication algorithms for initiator-send supported by this BS*
7	*4 * number*	*The list of ESP Transform IDs corresponding to the ESP transforms for initiator-send supported by this BS*
8	*4 * number*	*The list of ESP Authentication IDs corresponding to the ESP Authentication algorithms for initiator-receive supported by this BS*
9	*4 * number*	*The list of ESP Transform IDs corresponding to the ESP transforms for initiator-receive supported by this BS*
10	*33 + 4*n*	*Security Block*

The Length field contains a 16-bits value to record the whole frames size starting from Code field, with the ESP-Transforms-and-Authentication-Algorithms-Codes field filled in if present.

The PMK-Index field contains a 8-bits value to record the current Pairwise-Master-Key-Index each PKM-side used. If the PKM-Target detects the PMK-Index different of PKM-Initiator, it must update the latest Pairwise-Master-Key.

The Replay-Counter field contains a 64-bits random number (such as 64-bit NTP timestamp) and does not repeat within the life of the Master-Key material.

The Key-Lifetime field contains a 64-bits value to record the Session-Key lifetime in seconds.

The Key-Signature field contains an HMAC-MD5 message integrity check computed over the Session-Key-Frame starting from Code field, with the ESP-Transforms-and-Authentication-Algorithms-Codes field filled in if present, but with the Key Signature field set to zero. The M-Key is used as the HMAC-MD5 key.

The Security-Parameters-Index field contains a 32-bits value to assign to the IPsec Security Association (including the encryption and authentication keys, the authentication algorithm for AH and ESP, the encryption algorithm for ESP, the lifetime of encryption keys…etc in this session). PKM-Initiator/Target could check the SPI value in ESP-Header to detect to use which SA for this IPsec connection.

The following figure shows the Session-Key-Start message format

Code(1) =0	**Length(2)**	**PMK Index**(1)	Source_BSSID(6)	Destination_BSSID(6)
TLV Attributes…… NONCE (**32**) Security Parameters Index **(4)** Terminated Security Block (**33 + 4*n**)				

Figure 9 Session-Key-Start message format

The following figure shows the Session-Key-Request message format

Code(1) =1	**Length(2)**	**PMK Index**(1)	Source_BSSID(6)	Destination_BSSID(6)
TLV Attributes……				
NONCE (**32**)				
Replay Counter (**8**)				
Key Lifetime (**8**)				
Key Signature (**16**)				
Security Parameters Index **(4)**				
ESP Authentication IDs for initiator-send supported by this BS **(Codes Number(1) + Codes Number *4)**				
ESP Transform IDs for initiator-send supported by this BS **(Codes Number(1) + Codes Number *4)**				
ESP Authentication IDs for initiator-receive supported by this BS **(Codes Number(1) + Codes Number *4)**				
ESP Transform IDs for initiator-receive supported by this BS **(Codes Number(1) + Codes Number *4)**				

Figure 10 Session-Key-Request message format

The following figure shows the Session-Key-Response message format

Code(1) =2	**Length(2)**	**PMK Index**(1)	Source_BSSID(6)	Destination_BSSID(6)
TLV Attributes……				
NONCE (**32**) Replay Counter (**8**) Key Lifetime (**8**) Key Signature (**16**) Security Parameters Index **(4)** ESP Authentication IDs for initiator-send supported by this BS **(Codes Number(1) + Codes Number *4)** ESP Transform IDs for initiator-send supported by this BS **(Codes Number(1) + Codes Number *4)** ESP Authentication IDs for initiator-receive supported by this BS **(Codes Number(1) + Codes Number *4)** ESP Transform IDs for initiator-receive supported by this BS **(Codes Number(1) + Codes Number *4)**				

Figure 11 Session-Key-Response message format

The following figure shows the Session-Key-Accept message format

Code(1) =3	**Length(2)**	**PMK Index**(1)	Source_BSSID(6)	Destination_BSSID(6)
TLV Attributes…… Replay Counter (**8**) Key Signature (**16**)				

Figure 12 Session-Key-Accept message format

References

[1] IEEE C802.16h – 05/012r1 –General Architecture for Inter-network Communication Across 802.16 LE Systems, 2005-04-29

[2] IEEE Std 802.11F-2003, IEEE Trial-Use Recommended Practice for Multi-Vendor Access Point Interoperability via an Inter-Access Point Protocol Across Distribution Systems Supporting IEEE 802.11™ Operation.