CHAPTER 3
SECURITY ENHANCEMENT IN WiMAX KEY MANAGEMENT

3.1 INTRODUCTION

Although Mobile WiMAX is in the process of competing as a 4G candidate, it is still challenged by security threats and the proposed security enhancement in initial network entry process is discussed in chapter 2. Even though key generation is one of the functions of network entry process, it plays an incredible role in the goal of securing mobile WiMAX. This process not only generates key but also maintains and refreshes the key after lifetime upon request and hence discussed separately in this chapter. For managing those keys, the Privacy Key Management (PKM) Protocol is employed in WiMAX key management (Louafi Nuaymi 2007).

PKM protocol experiences security threats like replay, DoS, MITM, impersonation etc. The keys of MS have their own life time after which they will be no more valid (Syed & Mohammad Ilyas 2007). When the MS moves out from the coverage of BS, it has to switch to another BS and has to generate the keys with that BS. But when the MS resides in the same BS or moves within the coverage of the same BS, then the keys can be used till its lifetime. Before the expiry of key lifetime, the key should be updated to avoid connection disruption. This chapter proposes an Automatic Key Refreshing Technique which is ascertained for generating TEKs mutually. This technique reduces the unnecessary message exchanges and also avoids the transmission
of keys. Thus it reduces the bandwidth requirement for message exchange and also avoids impersonation, MITM & replay attacks thus augment the WiMAX key management security in an elevated manner.

3.2 MOBILE WIMAX KEY MANAGEMENT

As already mentioned in chapter 1, the security sublayer of WiMAX comprises two component protocols namely the encapsulation protocol and key management protocol. Here the focus is on key management performed by the protocol PKM version 2 which delineates the creation of keys, availability of them and intention of those keys (Frank Ohrtman 2005 and Seok-Yee Tang et al 2010). The key generation procedure based 802.1X/Extensible Authentication Protocol (EAP) ends with a Master Session Key (MSK). In addition, PKM is used to apply conditional access to network services, making it the authentication protocol, defending them from theft of service and providing a secure key exchange (Loufí Nuaymi 2007).

3.2.1 Privacy Key Management Protocol

The PKM protocol is used by the BS to control the distribution of keys needed for the security services like privacy, authenticity, data integrity and access control to the MSs. Distinctively, by using this key management protocol, the MS and the BS harmonize keying material (Altaf et al 2008). Security Association (SA) is the vital element of PKM protocol which is described as the set of security information shared between BS and its client MSs in order to support secure communications across the network (Andrews et al 2007). It is adverted here that within the MAC layer, there should be two kinds of connection namely management connections and transport connections. The transport connections are used for user data traffic and can be prerequisite or established on demand. Such data traffic in mobile WiMAX is
carried over by PKMv2 protocol instead of PKMv1 to achieve mutual authentication and to update the keys.

PKMv1 is a subset of PKMv2 in its functionality and the latter supports both mutual authentication & unilateral authentication, facilitates periodic re-authentication/reauthorization and key updates (Aboba et al 2004, Fan Yang 2011). In contrast to its predecessor, PKMv2 offers strong encryption algorithms to execute key exchanges between an MS and the corresponding BS (Sen Xu & Chin-Tser Huang 2006). After establishing a shared secret between the MS and the BS, PKMv2 utilizes it to secure subsequent exchanges of TEKs between the two parties. The summary of PKMv2 provided in this section embraces security negotiation, authorization/ authentication, key derivation, handshake and key transportation.

```plaintext
result = null;
Kin = truncate (key, 160);
for (i=0; i <= int( (keylength-1)/160 ); i++)
  if MAC mode is HMAC
    result = result $|$ SHA-1( ilstring $|$ keylength $|$ Kin);
  else if MAC mode is CMAC
    result = result $|$ CMAC(Kin, i $|$ ilstring $|$ keylength);
end
return truncate (result, keylength);

// truncate(x, y) is the rightmost y bits of a value x only if y ≤ x
```

**Figure 3.1 Dot16KDF Function**

The PKMv2 key hierarchy defines specifically the generation of various keys and their role in the system. There are two primary sources of keying material corresponding to the authentication schemes supported namely RSA-based and EAP-based. All PKMv2 key derivations are based on the Dot16 Key Derivation Function (Dot16KDF) which is an AES counter (CTR)
mode construction used to derive an arbitrary amount of keying material from source keying material (Seok-Yee Tang et al 2010) is detailed in Figure 3.1. The algorithm is defined differently depending on which MAC mode is negotiated during the security negotiation phase, whether HMAC or CMAC. PKMv2 uses a little different key hierarchy, due to the fact that two authentication systems are used: one is based on RSA and the other on EAP.

3.2.2 PKMv2 – EAP

Every WiMAX implementation, unilateral authentication is compulsory and the mutual authentication as optional. Experience has shown that mutual authentication is also extremely useful to have (Fan Yang 2011). Authentication is achieved using a public key interchange protocol which ensures not only authentication but also the generation of encryption keys. The IEEE 802.16e-2005 standard describes PKMv2 which permits three types of authentication such as RSA based authentication, EAP based authentication and RSA based authentication followed by EAP authentication. The proposed work uses EAP based authentication for mutual derivation of keys by MS and BS as shown in Figure 3.2.

At the initial entry, the MS and the Authentication Server (AS) mutually authenticate each other using an EAP-based authentication method. According to the mobile WiMAX specifications, the EAP authentication should follow the guidelines such as the mutual authentication support and security against the MITM attack. The result of the EAP exchange is the 512-bit Master Session Key (MSK), known to both the ASN server and the MS. After getting mutual authentication, the MSK is securely shifted from the AS to the authenticator (BS). The other keys derived are
**Pairwise Master Key (PMK):** The rightmost 160 bits of 512-bit MSK is truncated to get PMK which serves as the main session related root key for access security and it also plays its role in re-authentication.

**Authentication Key (AK):** Using PMK and the identities of MS and BS, AK of length 160 bits is derived.

\[
\begin{align*}
\text{MSK (512 bits)} & \quad \Downarrow \\
\text{Truncate (MSK, 160)} & \quad \Downarrow \\
\text{PMK} & \quad \Downarrow \\
\text{Dot16KDF (PMK, MS MAC Address |BSID| AK, 160)} & \quad \Downarrow \\
\text{AK (160 bits)} & \quad \Downarrow \\
\text{HMAC} & \quad \Downarrow \\
\text{Dot16KDF (AK, MS MAC Address |BSID| ‘HMAC_KEYs+KEK’, 448)} & \\
\text{HMAC_KEY_U (160 BITS)} & \quad \text{HMAC_KEY_D (160 BITS)} & \quad \text{KEK (128 BITS)}
\end{align*}
\]

*Figure 3.2 EAP based PKMv2*

**HMAC Keys:** AK in turn is used to generate HMAC Uplink Key (HMAC_KEY_U) and HMAC downlink Key (HMAC_KEY_U) each of length 160 bits that are utilized to sign & authenticate downlink and uplink managed messages respectively.

**Key Encryption Key (KEK):** AK also generates KEK of length 128 bits used to encrypt the messages such as TEK and KEK for group (GKEK) which are sent from the BS to the MS.
Traffic Encryption Key (TEK): Finally TEK of length 128 bits is generated by deriving Dot16KDF (KEK, SAID, TEK_SeqNo, 128) and is used to encrypt all messages between MS and BS. The TEK sequence number (TEK_SeqNo) is a part of the TEK parameter with a length of 2 bits used to avoid replay attacks with TEK (Seok-Yee Tang et al 2010).

3.3 REVIEW OF PREVIOUS RESEARCH WORK

Johnston & Walker (2004), Jacob (2011) review the IEEE802.16 security in many aspects, such as vulnerability in authentication and key management protocols, failure in data encryption and lack of explicit definition. Mutual authentication is the major contribution proposed by them, which enables MS to authenticate BS as well. Lang Wei-min et al (2008) scrutinized the physical layer threat and MAC layer threats of WiMAX and then listed the security needs of a WiMAX system.

Hur et al (2008) gives an overview of the EAP-based handover procedures of the latest IEEE 802.16e standard and their security flaws are evaluated and guarantees a backward/forward secrecy. Also AK is generated iteratively with pre-authentication to reduce the latency and for secure handover. Johnston & Walker (2004), Mandin (2004) analyze the need for PKM v2 to replace PKM v1. Sen Xu & Chin-Tser Huang (2006) discusses the IEEE 802.16 standard especially the security sublayer, and then investigates possible attacks on the basic PKM version as well as the PKMv2 and also gives possible solutions to counter those attacks. The uncertainty of cryptographic keys is essential for the security of cryptographic applications. Due to the stumpy entropy and possibly poor randomness of the passwords, they are not appropriate to be used directly as cryptographic keys.

Meltem Sonmez Turan et al (2010) propose the password based key derivation function which generates password based on the salt and length of
key values. Sen Xu et al (2008) focuses on the MAC layer security defects in the standard and the possible attacks to the authentication and key management protocols are also analyzed. A security handover protocol is also proposed which should be propped in the future 802.16e for mobility. Altaf et al (2008) examines the authorization protocol for both versions of PKM in WiMAX. Existing solutions tackle these problems using timestamp or nonce. A hybrid approach of nonce and timestamp has been proposed to prevent the authorization protocol from such attacks. Fan Yang (2011) provides a comparative analysis of TEK Exchange in both PKMv1 and PKMv2. Location based services in the WiMAX architecture are examined by Venkatachalam et al (2009) which are used for various applications like navigation, tracking of assets and personnel and emergency services. Jeremy Brown et al (2009), Adnan Shahid Khan et al (2011) proposed rekeying algorithms which are mainly needed for group authentication key update.

The review of research discussed the various possible attacks in the key management process and it is clear that security in the key management process still needs to be enhanced by focusing towards key exchanges during generation as well as refreshing before expiry.

3.4 PROPOSED METHODOLOGY

3.4.1 Issues in Existing Method

In the existing key exchange as shown in Figure 3.3, the MS originates the initial key exchange by sending an authentication message to BS which holds an X.509 certificate as explained below.

BS → MS (SA-TEK-Challenge message): During initial network entry or reauthorization, the BS will send this message towards the MS. The MS should respond with a PKMv2 SA-TEK-Request within a pre-defined time
limit. If this is not the case, the BS retransmits the same challenge message for a pre-defined number of times before it initiates another full authentication, or simply drops the MS.

**MS → BS (SA-TEK-Request message):** Upon acknowledgment of the previous message, the MS validates its contents by checking the HMAC/CMAC digest. After that, the MS sends the current message to the BS.

**BS→MS (SA-TEK-Reply message):** The BS also authenticates the message by inspecting its HMAC/CMAC. The BS verifies that the BS_Random and MS’s security capabilities reported in the ‘Security_ Neg’ field that matches with that of the MS details provided by ASN. If false, the BS informs this inconsistency to higher layers. On yielding true for both conditions, the BS responds to the corresponding MS using the current message. After the handshake, the MS should run a TEK delivery protocol instance, for example a TEK state machine, for each authorized Security Association Identity (SAID) having a data flow that needs traffic encryption (Seok-Yee Tang et al, 2010). The protocol uses the PKMv2 Key-Request/Key-Reply/Key-Reject messages.

![Figure 3.3 Initial Key Exchange](image-url)
**MS → BS (Key-Request message):** It contains the Authentication key Sequence number (AKSeqNo), SAID, MS_Random and HMAC/CMAC MIC over the contents of this message. This first message is optional and is sent only if the BS accepts to refresh the key before the MS requests it. The BS responds to the current message with a Key-Reply message that carries the BS’s active keying material for a specific SAID.

**BS → MS (Key-Reply message):** It contains a list of SAID to identify the security association. If the MS is authorized, the BS has to send the AK to MS. After that, key request message is initialized from MS to BS in the key exchange phase. Once MS is authorized, BS has to send the key response message that contains a sequence number, SAID, the HMAC along with old TEK and new TEK (Lang Wei-min et al 2008). At a speck of time two active keys have overlapping lifetimes as shown in Figure 3.4. The BS is able to support two active TEKs simultaneously for each MS during TEK transition period. After expiration of the old key, the key transition is terminated and when the time allotted for a particular user expires during key exchange, MS has to enter the network as a new client.

![Figure 3.4 TEK Exchange in Existing Key Management](image)
The observations about inadequacies in existing work are as follows:

- Huge amount of bandwidth is utilized since BS has to send 2 TEKs during each key request message from MS (Deiningger et al 2007).
- Large storage is required since both BS & MS have to store two keys.
- Time consumption for key exchange is more because MS has to wait for TEK from BS.
- Impersonation and MITM attack occur due to lack of security in key exchange (Sen Xu et al 2008).

For surmounting the above discussed inadequacies, an automated mutual key refreshing technique is launched in the WiMAX key management.

3.4.2 Proposed Key Refreshing Function

According to PKM protocol, the MS requests SA’s keying material from the corresponding BS. Every SA can be accessed by the authorized MS. And such authorization is provided by the BS in charge. When MS resides or moves within the coverage of BS without moving to neighbour BS, then the SA’s keying data expires after a particular time. So, upon delivering SA keying data to an MS, the BS notifies the user about the expiration times (Loutfi Nuaymi 2007). The MS tracks the keying material lifetimes constantly and before expiration updates them by querying the corresponding BS. Otherwise, the MS should repeat the network entry and initialization procedure. As already mentioned, key synchronization is also managed by the PKMv2 protocol.
Table 3.1 Lifetime of Keys

<table>
<thead>
<tr>
<th>Key</th>
<th>Minimum Lifetime</th>
<th>Maximum Lifetime</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>AK</td>
<td>1 day</td>
<td>70 days</td>
<td>7 days</td>
</tr>
<tr>
<td>TEK</td>
<td>30 minutes</td>
<td>7 days</td>
<td>12 hours</td>
</tr>
</tbody>
</table>

The keys, explicitly the Authorization Key (AK) and Traffic Encryption Key (TEK) play a significant part in key exchange. Each key has a specific key lifetime as depicted in Table 3.1. Once the lifetime of a particular key gets expired, the MS sends the request to the BS for the next key (Altatif et al 2008, Kahya et al 2012). To evade this circumstance in the proposed methodology, the new keys are generated simultaneously in both MS and BS before the expiration of old keys.

The genuine method for mobile WiMAX access control is Extensible Authentication Protocol (EAP)-based authentication, since the interaction with Authorization Authentication Accounting (AAA) server is very flexible and secured (Hur et al 2008). The proposed automatic key refreshing algorithm employs the EAP based authentication scheme which is already discussed in the previous section. Initially, the Master Session Key (MSK) is generated and further keys are derived from it. Before the termination of old TEK lifetime, the new TEKs are generated simultaneously in the proposed work.

The new key is generated based on Previous Key and Iteration Based Key Refreshing Function (PKIBKRF) as shown in Figure 3.5. An HMS (Hybrid MD5 SHA1) algorithm is used within this automatic key generation function.
Figure 3.5 PKIBKRF Technique

Figure 3.6 HMS Technique
At each of the 20 steps in a round of the SHA –1 compression function additional steps from MD5 compression function are combined as shown in Figure 3.6. New TEK is generated from the old TEK key with the help of MSK, MS_MAC, BSID and CID (Meltem Sonmez Turan et al 2010). For each SS, a 16-bit CID is provided by BS during network entry and initialization phase. A 48-bit BSID is used for operator identification and it is different from the MAC address.

3.4.3 Automated Key Refreshing

The initial authorization phase and key exchange phase are handled as similar to the existing system as in Figure 3.3. Once the key lifetime is expired, MS wants the new key from BS for further communication (Lang Wei-min et al 2008). At each time the MS performs the authentication & authorization phase. Lin & Chen (2003) discusses about reducing the message exchanges for wireless networks in the motive of minimizing time taken for key management. A similar economical message exchange is ensured by proposed method but in a secured way.

The total time required for key management is specified as $T_{KM}$.

$$T_{KM} = T_{AUTHEN}+T_{AUTHOR}+T_{KEYEX} \quad (3.1)$$

To avoid both phases and hence reduce the bandwidth utilization, the keys are generated simultaneously by both BS & MS mutually in the proposed work. Thus only one third of the total key management time as mentioned in Equation 3.1 is required. MS has to send a key update request to BS for automatic mutual key generation shown in Figure 3.7.
Figure 3.7 Mutual Key Generation Technique

Table 3.2 Terminologies used in Algorithm

<table>
<thead>
<tr>
<th>Terms</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS_MAC</td>
<td>Mobile Station MAC Address</td>
</tr>
<tr>
<td>KLT</td>
<td>Key Lifetime</td>
</tr>
<tr>
<td>KRF</td>
<td>Key Refresh Time</td>
</tr>
<tr>
<td>SST</td>
<td>Session Start Time</td>
</tr>
<tr>
<td>E_{AK}()</td>
<td>Encryption with Authorization Key</td>
</tr>
<tr>
<td>D_{AK}()</td>
<td>Decryption with Authorization Key</td>
</tr>
</tbody>
</table>
Start Timer and Initialize it to key lifetime

if Timer == 400 seconds

//key update request Optional during call drop
Send E_{AK}(CID|MS_MAC|TS|N1) to BS
Wait for acceptance from BS

//key generation
K_{input}=CID|oldTEK|MS_MAC|BSID|MSK
temp=HMS(K_{input})
newTEK=truncate(temp,128)
//For key verification in BS
MS_kv= truncate(HMAC(newTEK),16)

end

Figure 3.8 Key Refreshing in MS

The timer is synchronized in both BS and MS and the SST is set as 400 seconds which is the maximum time needed to update the key successfully in spite of collisions and network delays during message exchanges. Hence the key lifetime is initialized (in the range of 30 mins to 7 days as mentioned in Table 3.1) and key update request is sent by MS to BS if the timer is 400 seconds. To ensure security, the key update request with timestamp and nonce is encrypted with AK.

After the acceptance of MS by BS, the key generation algorithm is processed as explained in Figure 3.8. The key update request is made optional only during call drops to retain communication (Pero Latkoski & Borislav Popovski 2012). This phase can be bypassed for periodical key update and the key generation is ensued automatically. The keys are generated iteratively and the key updating count is set to 3 to avoid the bandwidth utilization by limited users. Once the key is generated, MS has to send rightmost 2 bytes of the
HMAC of TEK to BS for key verification. Then, the BS verifies the key generated by MS. After the verification process, the data encryption takes place. If the iteration is completed, the user will enter the network again as a new user. In Table 3.2, the terminologies used in algorithms are described.

The proposed algorithm for key refreshing and verification in BS is illustrated in Figure 3.9. After receiving the key update request from MS, BS checks its validity. If the MS is legitimate, BS checks whether the key update request is advanced or on time. The signal pattern “01” is sent to reject the user if the key updating exceeds the count value 3. Also BS sends “10” signal pattern for illegitimate MS proved while entering the network.

It is accomplished by calculating the difference in time between key lifetime and current clock time. If it is more than 400 seconds, it is regarded as an advanced request and the chances are more for this type of request to be initiated by attackers. The signal pattern “01” is sent by BS to indicate the rejection of the MS. If the difference is less than 400 seconds, then the key update count of MS is checked. If it is less than 4, MS is accepted by the BS and sends “00” as the signal pattern after decrypting the message, which is sent by the MS. The new TEK is generated by the same procedure as in MS.

To ensure that the same key is used in both MS and BS, verification is done by deriving the rightmost HMAC (newTEK) and matching it with that of the same derived in MS and sent to BS. Timestamp and the received nonce N1 are sent to ensure mutual authentication and to avoid replay, impersonation and MITM attacks. After the successful completion of this message exchange, oldTEK is replaced by newTEK by both MS and BS. Thus, the periodical updation of key is automatically done with less message exchanges, less bandwidth and more security.
Figure 3.9 Key Refreshing and Verification in BS

3.5 RESULTS AND DISCUSSION

3.5.1 Initial Key Generation Process

The previous section explained the proposed automated mutual key updation technique with the help of proposed algorithms. This section depicts the results of simulation process according to the proposed algorithms. MS
derives its AK and TEK that are derived initially using any one of the PKM-v2 protocols explained key derivation and hierarchy of this chapter. Simulation scenario uses PKM V2-EAP for the generation of initial key. Figure 3.10 shows the generation of MSK and Figure 3.11 shows the generation of other keys like PMK, AK, KEK, HMAC keys and TEK.

![Figure 3.10 MSK generation for MS](image)

3.5.2 Key Refreshing Process

As soon as the MS received the key, the timer starts with the key lifetime 1800 second. When the timer reaches 400 seconds, key update request is initiated from MS. The GUI window shown in Figure 3.12 illustrates the automated mutual key updating algorithm. It demonstrates the key exchange between the MS and BS.
Figure 3.11 Generation of other keys for MS

Figure 3.12 GUI implementation of Key update
Figure 3.13 Key Refreshing by MS

Figure 3.13 shows the separate key refreshing process for a valid MS. Key update process of an authorized MS results in a new TEK. The encrypted TEK and the update count is shown in Figure 3.14. Number of key update request sent from the MS is counted to avoid requesting beyond 3 times.

Figure 3.14 Key Update Request sent to BS by MS
On receiving this, the BS checks the validity of MS by accessing the ASN database. Key refreshing process is allowed only when the user is proved as valid. When the user is valid BS sends its acknowledgement with its MAC, BSID, timestamp and nonce to ensure mutual authentication and to avoid security threats. For a valid user, the reply is the accept message with the corresponding timer value as shown in Figure 3.15.

![Figure 3.15 Accept Message for Valid MS](image1)

![Figure 3.16 Key verification Request by MS](image2)

Once the user is accepted, then the calculation for new TEK is performed in BS similar to MS. In the meantime MS sends the key verification request to BS. Key verification message reduces the bandwidth requirement by
sending only the rightmost 2 bytes of HMAC of new TEK to the BS as shown in Figure 3.16. If the key verification message is fulfilled, then the key accept message as in Figure 3.17 is sent by the BS. The signal pattern ‘00’ is given for the accept message and ‘01’ is given for the reject message.

Figure 3.17 Acceptance of key update by BS

Figure 3.18 Limitation of Key update

To evade the utilization of bandwidth by the same user, the BS allows the key refreshment only for 3 times. Even though the user is valid within the network, the BS rejects the key update request message if the user
exceeds the key request count. As in Figure 3.18, the BS shows the message box as key update count exceeds.

**Figure 3.19 Identification of invalid MS**

**Figure 3.20 Rejection of invalid MS**

In Figure 3.19, identification of the invalid user is explained. The BS identifies the unauthorized MS and only 2 bits of signal pattern ‘01’ is sent as the reject message without responding any details of BS.
**Figure 3.21 Key update Request in Advance**

When any fraudulent MS sends the encrypted request impersonating a valid MS, it can be identified by the verification of key by the HMAC value and this type of user is rejected as shown in Figure 3.20. Key update request is also not allowed before its concerned time. Hence key update request of MS coming to BS when the key lifetime of MS is more than 400 seconds is rejected as shown in Figure 3.21 Because this type of request has more chances to be initiated from MITM.

### 3.5.3 Comparative Analysis

A comparative analysis is carried out to to point the merits of proposed work. In Figure 3.22, comparison of bandwidth usage by the users in existing key management and in proposed methodology are analyzed using the results shown in the Table 3.3. The observed information from the comparison is that the bandwidth is reduced in proposed technique compared to existing methodology. Simulation is experimented for the existing and proposed key updation for 10 MSs. The key updation request count of MS may be a random number from 1 to 5. When the count is greater than 3, then the MS is not allowed for key updation. MS has to request for new key as like initial entry. Figure 3.22 shows the bandwidth utilisation by proposed methods.
Table 3.3 Bandwidth (BW) requirement of existing and proposed methods

<table>
<thead>
<tr>
<th></th>
<th>Sim1</th>
<th>Sim2</th>
<th>Sim3</th>
<th>Sim4</th>
<th>Sim5</th>
<th>Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>BW for Existing method in Kbits</td>
<td>5.76</td>
<td>5.76</td>
<td>5.76</td>
<td>5.76</td>
<td>5.76</td>
<td>5.76</td>
</tr>
<tr>
<td>BW for TEK by Proposed method update in Kbits</td>
<td>4.752</td>
<td>5.004</td>
<td>4.626</td>
<td>4.752</td>
<td>4.878</td>
<td>4.8024</td>
</tr>
<tr>
<td>BW for TEK &amp;AK by Proposed method Kbits</td>
<td>3.602</td>
<td>4.204</td>
<td>3.316</td>
<td>3.602</td>
<td>3.918</td>
<td>3.7284</td>
</tr>
</tbody>
</table>

Average Bandwidth requirement for updating TEK using the proposed method saves 16.63% compared to the existing method. If the proposed method is also pertained to AK, the bandwidth requirement can save 22.36% compared to existing method. If the key update count of all the MSs who request the key update are assumed to be less than or equal to 3, then the TEK updation saves the bandwidth of 21.87% compared to existing method. Key updation when adapted for both AK and TEK saves the bandwidth of 49.65% compared to existing method.

![Bandwidth Consumption](image)

Figure 3.22 Bandwidth Consumption
The communication is discarded if any collision occurs in existing key exchange, and the authentication & authorization steps are repeated for new request. But in proposed methodology, it is made easy by just resending the key update request. In Table 3.4, the Key Updation Time (KUT) is calculated and listed for both the existing and proposed key exchange. Hence the proposed key updation saves the average time of 51.9% for collisions of range 1 to 5 when compared to the existing key exchange as shown in Figure 3.23.

Table 3.4 Calculation of Key Updation Time (KUT)

<table>
<thead>
<tr>
<th>No. of Collisions</th>
<th>KUT of Proposed method in secs</th>
<th>KUT of Existing method in secs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7.95</td>
<td>20.71</td>
</tr>
<tr>
<td>2</td>
<td>20.27</td>
<td>41.79</td>
</tr>
<tr>
<td>3</td>
<td>32.00</td>
<td>62.28</td>
</tr>
<tr>
<td>4</td>
<td>39.90</td>
<td>80.94</td>
</tr>
<tr>
<td>5</td>
<td>46.68</td>
<td>99.48</td>
</tr>
<tr>
<td>Total</td>
<td>146.8</td>
<td>305.2</td>
</tr>
</tbody>
</table>

Figure 3.23 Traffic Analysis with collision
3.6 CONCLUSION

This chapter discussed about the provisioning of security in key management of network entry process which contributes the following:

- By the proposed automatic key refreshing technique, the bandwidth used for key exchange is saved and it plummets the traffic congestion.
- Security enhancement by avoiding Man in the middle attack, Replay attack and Impersonation attack.
- It requires the storage of only one key, hence the petite amount of memory is required. Also it saves the key updation time when the calls dropped due to collisions.
- Avoids service interruption greatly and saves channel resources by less message exchanges.

If the proposed method is applied to TEK management, then the consumption of bandwidth is average of 16.63% and maximum of 21.8% less than the existing technique while considering the key storage. In this chapter, the proposed work concentrates on TEK management because the lifetime of TEK is less compare to AK management. If it is applied for both AK and TEK exchange, then the bandwidth consumption is an average of 22.36% and maximum of 49.65% compared with the existing methodology. The key updation time is saved up to 51.9% although the presence of collision contrasts to the existing key management time.

Extension to this work can be focussed on time consuming mutual key generation algorithms with the condition of not compromising the security. The Proposed work can also be analysed implementing in other protocols like PKM v2-RSA and PKM v2-RSA followed by EAP. Time consumption can be evaluated by implementing the proposed work in a real time mobile WiMAX test bed.