Strong Password Based EAP-TLS Authentication Protocol for WiMAX

Anjani K.Rai,Shivendu Mishra and Vimal Kumar
Computer Science and Engineering Department
Motilal Nehru National Institute of Technology
Allahabad - 211 004, India

***Abstract*—Security becomes more important in wireless network due to lack of physical boundary. Worldwide interoperability for microwave access (WiMAX) is a wireless communications technology, also known as IEEE802.16 that is intended for wireless "metropolitan area network". IEEE802.16e (WiMAX amendment) is a standard for mobile application in WiMAX. IEEE 802.16e supports EAP (Extensible authentication protocol) for authentication. EAP-TLS which is a recommendation of IEEE802.16e provides strongest security but is inefficient and costly due to use of PKI to handle the certificate. This paper proposed a strong password based EAP-TLS authentication protocol which uses simple common password for mutual authentication in place of certificate and hence reduces the problem associated with EAP-TLS and satisfy the entire requirement for IEEE 802.16e authentication.**

Keywords—IEEE802.16; WiMAX; extensible authentication protocol; EAP-TLS

I. INTRODUCTION

IEEE 802.16 is the standard for broadband wireless access (BWA) [1]. IEEE 802.16(e) is a WiMAX amendment that deals mobility in WiMAX network [2]. The wireless system is less secure than wired system due to lack of physical boundary. RFC 3748 defines EAP (Extensible Authentication Protocol) which is universal authentication framework and is supported by IEEE802.16e [7]. Initially EAP was developed for use with Point-to-Point Protocol links and was later adapted for use by wired and then wireless IEEE 802 networks. In all of these situations an attacker may spoof EAP packets, launch denial of service attacks, recover passwords using a dictionary attack, initiate a man-in-the-middle attack as well as other types of attacks by accessing the link over which EAP packets are transmitted [7].

The explicit authentication mechanism supported by EAP is called EAP methods. There have been a number of EAP methods proposed, each of them having some advantage and disadvantage. EAP-TLS is one of the strongest secure EAP method which provides mutual authentication and exchange the key information so that encryption can be established between the supplicant (client) and authenticator (server) and secure communication can continue from this point until the connection is broken.

The drawback of EAP-TLS is that it requires both the supplicant (client) and the authenticator (server) to have public key certificates. The issuing of public key certificates requires a lot of additional administrative work as well as overhead.

In this paper, we have proposed a strong password based EAP-TLS authentication protocol which is fully compliant with RFC 4017 (an unofficial standard for an EAP method to be use in wireless network) and better satisfy the security goals under the WiMAX security architecture.

The paper is organized as follows. In Section II, we identify the authentication method requirement for WiMAX. In Section III, we have examined some EAP methods. Section IV describes the propose strong password based EAP-TLS authentication protocol for WiMAX network. Section V presents the performance analysis and Section VI concludes the paper

II. AUTHENTICATION METHOD REQUIREMENT FOR WIRELESS NETWORK

RFC 4017[9] identify the requirement for authentication in wireless network, which are classified into three levels: Mandatory Requirements, Recommended Requirements and Optional Requirements

A. Mandatory Requirements

The mandatory criteria for an EAP method to be use for wireless network are:

- During authentication, a strong master session key with128-bits of effective key strength must be generated by an EAP method. Also, the key must contain at least 64 octets [9].
- An EAP method for use with wireless network must provide mutual authentication meaning that the user authenticates authenticator and the authenticator authenticates the user in a single interlocked communication as defined by RFC 3748[9]
- An EAP method must be resistant to dictionary attacks, which means that when a password is used as a secret, the method does not allow guessing the password by observing pair of challenge and replying message [8].
- An EAP method must protect against man-in-the-middle attacks by using the mutual authentication [9].
- An EAP method must provide Credential security for the confidentiality and integrity protection of user's data.

- An EAP method must protect the EAP conversation by negotiating ciphersuite.

These are the mandatory requirements that are established in RFC 4017. Any EAP method must satisfy these requirements to be used for wireless communication.

B. Recommended Requirement

The Recommended Requirements criteria for an EAP method to be used for wireless communication are:

- If an EAP packet exceeds the minimum MTU of 1020 octets then method should support fragmentation and reassembly [9].
- An EAP methods used for wireless communication should also support end-user identity hiding.

C. Optional Requirement

Following are the optional requirement for an EAP method to be used for wireless communication:

- An EAP methods used for wireless communication may support channel binding as defined in RFC 3748.
- An EAP methods used for wireless communication may also support fast reconnection as defined in RFC 3748[9].

III. EXTENSIBLE AUTHENTICATION METHODS OVERVIEW

Extensible Authentication Protocol (EAP) is a flexible authentication framework which supports multiple authentication methods. In WiMAX network, EAP allow to exchange complex authentication protocol between MS (end user) and BS (authenticator).

There are two types of EAP model specified in [7], they are: pass through behavior model and multiplexing model. In pass through behavior model, there are three entities involved in EAP authentication, i.e. Supplicant, Authenticator, and Authentication Authenticator, all of them reside in three separated devices. Supplicant resides in wireless client station, authenticator resides in access points, and authentication authenticator resides in AAA (Authentication, Authorization, and Accounting) Authenticators, such as RADIUS and DIAMETER. The authenticator will act only as a pass-through device.

In multiplexing model, there are only two separated devices, where authenticator and authentication authenticator entities reside in a single device. The authenticator will implement all authentication services. For simplicity we will consider multiplexing model where supplicant resides in MS/SS and authenticator resides in BS for WiMAX network as shown in figure 1.

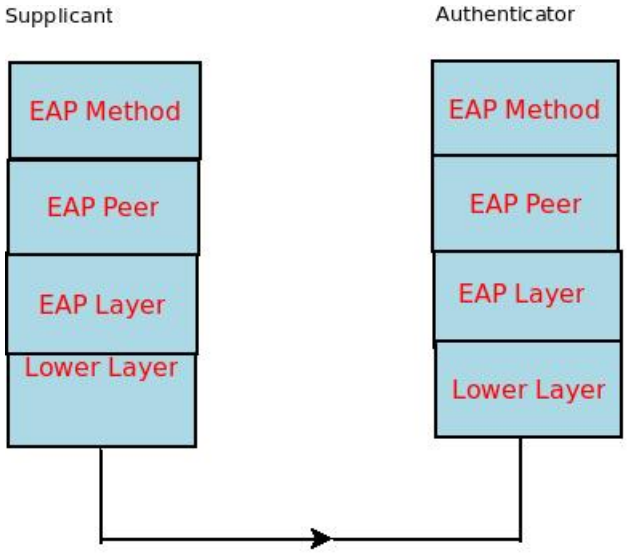

Figure 1: Multiplexing model implementation

Currently there have been a number of EAP methods proposed including the following.

A. EAP-MD5

EAP-MD5 (EAP-message digest 5) is the very fast EAP method and was developed by RSA. It uses a three way handshake in order to authenticate the user and hash function to hide the secret or password [7]. EAP-MD5 uses a 128-bit generated number string, or hash, to verify the authenticity of a user.

There are several disadvantage of EAP-MD5 like birthday attack, lack of mutual authentication, dictionary attack. Method does not provide any means by which to establish a master session key which is also a big disadvantage [8]

EAP-MD5 does not satisfy the mandatory requirement of RFC 4017 and hence should not be used for wireless communication authentication [8].

B. EAP-TLS

EAP-Transport Layer Security (EAP-TLS) [3] includes support for certificate-based mutual authentication and key derivation. The EAP-TLS conversation will typically begin with the authenticator (server) and the supplicant (client) negotiating EAP. The authenticator typically sends n EAP Request/Identity packet to the supplicant, supplicant will respond with an EAP-Response/Identity packet to the authenticator, containing the supplicant's user-Id. Assuming that supplicant and authenticator have already negotiated the EAP Request/Identity and EAP Response/Identity, the Overall EAP-TLS procedure is shown in figure 2 and can be summarized as:

- The client and the server exchange version number, session id, and random number (used for replay protection) and negotiate a common cipher suite and hash Functions with the Client Hello and the Server Hello messages.

- Server sends Server Certificate Handshake message which contains public key certificate for a key exchange public key (such as an RSA or Diffie-Hellman key exchange public key) or a signature public key (such as an RSA or Digital Signature Standard (DSS) signature public key). If Server Certificate Handshake message contains a signature public key then a TLS Server Key Exchange handshake message must also be included to allow the client for key exchange. If Server requests client authentication, it sends a Certificate Request message listing acceptable certificate types and certificate authorities. Server finishes the Hello procedure with the Server Hello Done message.
- Client verifies the validity of Servers certificate. If public key certificate is requested by Server, Client sends it in the Certificate message. the client encrypts a random number with the server's public key, and sends it in the Client Key Exchange message to finish the agreement on the session key and sends the Certificate Verify message (if necessary) to be verified by Server. Client also generates key material from the random number for encryption and decryption.
- Server decrypts the message with its private key and gets the random number and generates key material for encryption and decryption. This concludes the handshake and the PMK is now use to encrypt and decrypt the rest of the session [3].

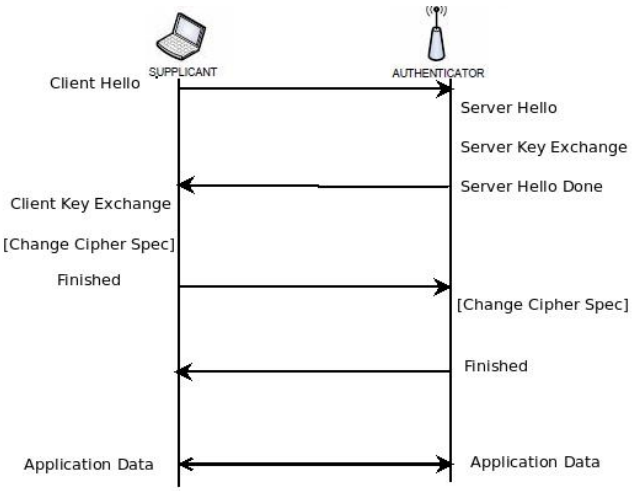

Figure 2: Message flow in EAP-TLS Protocol

There is several advantage to use EAP-TLS, this method provides mutual authentication. This method also provides for the exchange of key information so that encryption can be established between the supplicant (client) and authenticator (server) and secure communication can continue from this point until the connection is broken. The biggest advantage of EAP-TLS is that it supports fast reconnect as defined by RFC 3748. This means that once a connection is established between supplicant and authenticator, to create a new connection or association requires less round-trips than the original connection took to establish [3]

The largest disadvantage of EAP-TLS is that it requires both the supplicant (client) and the authenticator (server) to have public key certificates. The issuing of public key certificates requires a lot of additional administrative work as well as overhead. Another disadvantage of EAP-TLS is that the exchange of certificate information requires additional time so the method is not quick and efficient [2].

EAP-TTLS (EAP-tunneled TLS) minimizes the overhead associated with EAP-TLS by using only authenticator side certificate. In this method the authenticator (server) is authenticated by supplicant (client) using public key certificate of the authenticator. Supplicant (client) may be authenticated by authenticator (server) using the secure connection established during handshake. The authentication of the supplicant may itself be EAP, or it may be another authentication protocol such as PAP, CHAP, MS- CHAP or MS-CHAP-V2 [5].

Thus EAP-TTLS allows legacy password-based authentication protocols to be used against existing authentication databases, while protecting the security of these legacy protocols against eavesdropping, man-in-the-middle and other attacks [5].

Since there is many more supplicants than the authenticator therefore using only authenticator side public key certificate method greatly cuts down the overhead associated with EAP-TLS. As with EAP-TLS, EAP-TTLS supports fast reconnect and it supports a key strength of 384 bits [8].

Asokan et al [10] discovered that EAP-TTLS protocol is vulnerable to the Man-in-the-Middle attack. Although method reduces overhead by using only server side certificate but the method is still more expensive than other password based EAP methods.

There are some other EAP methods that are based on password for authentication rather than certificate for example EAP-LEAP, EAP-FAST. One of the biggest advantage of password based EAP methods are that they provides the security level similar to that of certificate base EAP method without overhead associated with maintaining PKI infrastructure .

IV. PROPOSED STRONG PASSWORD BASED EAP-TLS AUTHENTICATION PROTOCOL

From aforementioned discussion we have concluded that EAP-TLS which is also recommended by IEEE 802.16e is the strongest security techniques but is inefficient and costly due to use of PKI to handle the certificate. Strong password based EAP methods like EAP-SPEKE (Simple Password-Authenticated Exponential Key Exchange) provides mutual authentication between supplicant (client) and authenticator (server) by using a simple common password. EAP-SPEKE may be consider as an option of EAP-TLS which provides strong security and authenticate both user and device but it does not support fast reconnection which is the one of important requirement of EAP.

The proposed strong password based EAP-TLS authentication protocol overcomes the problem associated with EAP-TLS and EAP-SPEKE by taking the advantage from both. Proposed Protocol authenticates the supplicant (client) and authenticator (server) mutually by designing TLS handshake message using SPEKE.

In multiplexing model where only two devices supplicant (client) and authenticator (server) exist, proposed extensible authentication method use the following variable name: N is 1024 bit prime number of the form kq+1. q is large prime factor of N-1. k=N-1/q and all divisor of k/2 must be greater than q (excluding 1). x is password, g is a suitable Diffie-Hellman base such that **$g= hash(x)^k \bmod N$** (SHA1 may be use for hashing) and is stored at server. a and b are the random number selected by client and server respectively.

Overall procedure is given in fig 3, as shown in the figure, it is supposed that both supplicant and authenticator have already finished the EAP-Request/identity and EAP-Response/Identity message. Both the authenticator and supplicant didn't have to send the Certificate message comparing with the certificate-based Schemes because public key can be derived from the identity (username).

1) Supplicant sends username or identifier (I) in extended Client Hello message.

2) Authenticator reply by sending extended Server Hello handshake message followed by Server Key Exchange, and server Hello Done. Server Key Exchange message consists N, g and server's public key B where **$B=g^b \bmod N$.** Supplicant MUST abort the handshake with an alert if B is not in the range [2, (N-2)].

3) Supplicant gets the value of N, g from the Server Key Exchange message and computes his public key **$A= g^a \bmod N$**. Supplicant sends his public Key in Client Key Exchange message. Authenticator MUST abort the handshake with an alert if A is not in the range [2, (N-2)].

4) Supplicant and Authenticator uses Change Cipher Spec message to notify each other that they are switching to new cipher algorithm.

5) Both supplicant and authenticator calculates secret key **$S=B^a \bmod N$** and **$S=A^b \bmod N$** respectively

6) Finished message are use to ensure that both supplicant and authenticator are possession of same secret key. Supplicant sends k_1=KCF (kcfParam1) in Finish message and authenticator sends k_2=KCF (kcfParam2) in Finish message. Where (kcfParam1)= hex byte 04, (kcfParam2) = hex byte 03, KCF1-SHA1(kcfParam)=SHA1 ((kcfParam) | A | B | S | g) [6] (kcf)= KCF1-SHA1.

The authenticator computes its value for K1 as a hash of its concatenated values for (kcfParam1), A, B, S, and g, and then it compares its K1 value to the value of K1 received from the supplicant. If they are not equal, the authenticator must abort and sends an error message. The supplicant does the same procedure for verification of authenticator's K2 [6]. Now both Supplicant and authenticator have authenticated to each other and a shared secret key is established between them.

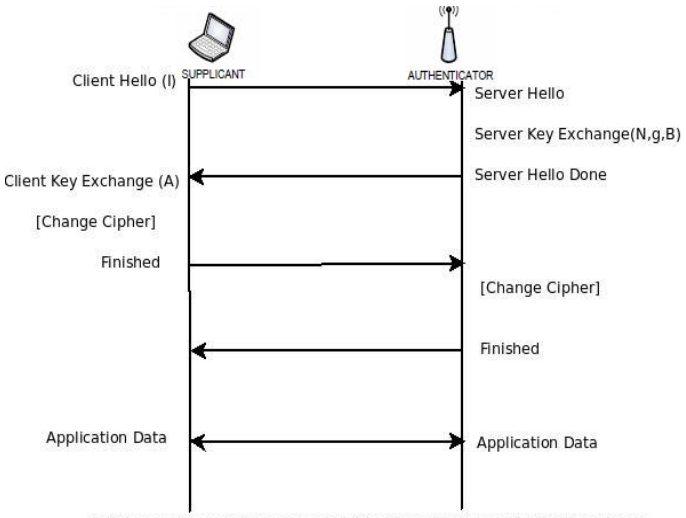

Figure 3: Message flow in proposed strong password based EAP-TLS protocol

V. PERFORMANCE ANALYSIS

The security of proposed strong password based EAP-TLS authentication protocol depends upon hardness of DH problem. Since N is 1024 bit prime defining large subgroup of order q therefore an attacker can not calculate discrete logarithm using the method in [11] and hence supplicant's password could not be compromised.

In the proposed protocol any passwords or password-crackable data does not send over the network; although the password is too small but methods prevent active network attacks (man-in-the-middle, replay, etc.) as well as passive attack. The random number a and b selected by supplicant and authenticator consist more than 160 random bit which avoid the chance of brute-force attack and also computation of discrete logarithms. Since the group parameter that Supplicant receives from authenticator is properly checked by looking all divisor of k/2 which assures that the parameter are not altered.

The method provides a zero-knowledge proof of password which does not depend upon success of authentication and prevents unconstrained guessing from network attackers.

The propose method provides mutual authentication and incorporation of Diffie-Hellman ensures that if an attacker compromise the current key then any future key could not be compromised by him which is important for strong key management

Table 1: Comparison of EAP methods against security requirement

EAP Type	EAP-MD5	EAP-TLS	EAP-TTLS	EAP-SPEKE	Proposed Protocol
Mutual Authentication	No	Yes	Yes	Yes	Yes
Credential security	Weak	Strong	Strong	Strong	Strong
Man-in-the-middle attack protection	No	Yes	Yes	Yes	Yes
Strong session key	No	Yes	Yes	Yes	Yes
Dictionary-attack protection	No	Yes	Yes	Yes	Yes
Replay protection	No	Yes	Yes	Yes	Yes
User Authentication	Yes	Not, if certificate is store on disk	Not, if certificate is store on disk	Yes	Yes
Efficient	Yes	No	No	Yes	Yes
Low cost	Yes	No	No	Yes	Yes
Fast reconnect	No	Yes	Yes	No	Yes
Ease implementation	Yes	if certificate is store on disk	if certificate is store on disk	Yes	Yes

In this method compromise of password at any stage does not allow to decrypt any past message since message is encrypted by key which was derived by password as well as two large random number a, and b.

The method prevents password-sniffing and all the active and passive network attacks (man- in-the-middle, replay, etc). [6].Since the method requires only software packages in place of PKI infrastructure which causes long handshake latency and certificate management overheads, therefore cost of implementation is greatly reduces compared to EAP-TLS [8].

The proposed protocol also improved the performance of re-authentication (reconnection) by reducing number of message between supplicant and authenticator using small password in place of PKI infrastructure of EAP-TLS.

Table 1 compares the proposed method with others extensible authentication protocol against the various security properties of EAP method. As shown in table the propose method provides strongest level of security than any other EAP method. Propose protocol fulfills all the Mandatory, Recommended and Optional Requirements that are outlined in RFC 4017 and hence is one of the best suitable authentication method for IEEE 802.16(WIMAX) type of network.

VI. CONCLUSION

In this paper we have proposed strong password based EAP-TLS authentication protocol which supports mutual authentication and prevents password-sniffing and all the active and passive network attacks (man- in-the-middle, replay, etc). The method is simple password based, uses only one modulo exponential during protocol and does not require PKI infrastructure and hence maintains low cost.

The proposed protocol support fast reconnection by reducing number of round trip which occurs due to presence of PKI infrastructure. The comparative study shows that the protocol is more promising since it provides strong security, high efficiency and easy deployment.

REFERENCES

[1] IEEE Std. 802.16-2004, Air Interface for Fixed Broadband Wireless Access Systems, IEEE, Oct. 2004.

[2] IEEE Std. 802.16e-2005, Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems—Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands, IEEE, Feb. 2006.

[3] Simon, D., Aboba, B., and Hurst, R. 2008. The EAP-TLS authentication protocol. RFC 2716.

[4] Cam-Winget, N., Mcgrew, D., Salowey, J., Zhou, H. 2007. The flexible authentication via secure tunneling extensible authentication protocol method (EAP-FAST). RFC 4851.

[5] Extensible Authentication Protocol Tunneled Transport Layer Security Authenticated Protocol Version 0 (EAP-TTLSv0)

[6] D. Jablon: The SPEKE Password-Based Key Agreement Methods, IETF draftjablon-speke-02.txt(2003).

[7] Adoba, B., Blunk, L., Vollbrecht, J., Carlson, J. and Levkowetz, E. 2004. Extensible authentication protocol (EAP). RFC 3748.

[8] David Q. Liu,Mark Coslow, Extensible Authentication Protocols for IEEE Standards 802.11 and 802.16.

[9] Stanley, D., Walker, J., and Aboba, B. 2005. Extensible authentication protocol (EAP) method requirements for wireless LANs. RFC 4017

[10] N. Asokan, V. Niemi and K. Nyberg, “Man-in-the-middle in tunneled authentication protocols”, in the 11th Security Protocols Workshop, Cambridge(UK), April 2003, Springer-Verlag, 2003

[11] Gordon, D., "Designing and Detecting Trapdoors for Discrete Log Cryptosystems", Springer-Verlag Advances in Cryptology - Crypto '92, pp.66-75,1993