ISSN:0975-9646

A New Lightweight EAP-PK Authentication Method for IEEE 802.11 standard Wireless Network

*Dhiraj M. Londe[1], Prof. Nitin Chopde[2]
[1]*ME (Scholar) G.H. Raisoni College of Engineering & Management, Amravati*
[2]*ME (CSE) G.H. Raisoni College of Engineering & Management, Amravati*

Abstract- **Wireless Local Area Network in IEEE802.11 standard is facing more complicated problem related security thread, which expose legitimated users to increased risk. Therefore, the security in WLAN network is very challenge in authentication. Extensible Authentication Protocol (EAP) is an authentication framework that is frequently used in wireless networks and Point to Point connections. For authentication purpose EAP used different methods to performed authentication. In this paper, we will explain each existing EAP authentication method, show the flow of every one and restriction. That way we will propose new method of authentication: Lightweight Extensible Authentication Protocol-Public Key "EAP-PK". This method of authentication combines between the simplicity of deployment and management of password method and robustness of certificated ones. We will check the EAP-PA security properties (like security and authentication) by using the specialized model checker AVISPA, which provides formal proofs of the security protocol.**

Key words: **Wireless Local Area Network, security protocol, Access control, EAP, AVISPA.**

I. INTRODUCTION

Consider the last era, the use of wireless communication technologies has been growing very fastly. This improvement happen due to the new web application (like facebook, twitter, youtube, what apps etc.) which introduce in the multiple latest solution like laptop, tablet, Notepad and mobile phone. All this such type of devices use unsecured public network to transmit confidential information such as user name, password, private data (such as bank account number, ATM pin) that require high security level.

II. WIRELESS LOCAL AREA NETWORK

Securing communication in WLANs network is a very complex challenge in network security, due to vastly introduce smart phone. The mobile client's need a way for both mutually prove their identity between them and verity the contents of their data traffic that manipulated between them is free of tampering or sniffing. In communication, there are some goals must be achieved to have successful security in wireless networks that is mutual authentication, identity privacy and data integrity in communication.

The first generation of wireless technologies had a bad reputation, due to their poorly designed security strategy by using WEP (Wired Equivalent Privacy) protocol [5]. WEP weakness include lack of protection against malicious tampering of message, incorrect usage of an encryption algorithm, a repayable authentication method (that is, an eavesdropper can sniff a valid user's authentication and replay it to gain the access to the network) [5, 6], key generation that is secret key is too small, only 40 bits and it is very susceptible to brute force attacks and secret key are accessible to user, therefore key is not secret.

The Wi-Fi alliance, the international association of wireless device manufacturers, responded to these weaknesses with Wi-Fi Protected Access (WPA).

IEEE has also finalized a draft of IEEE 802.11i standard, also known as Robust Security Network (RSN) or WPA2, which is specially designed to address WEP's weakness [5]. This new release standard provides an intelligent authentication mechanism based on EAP (Extensible Authentication Protocol) protocol and the success of EAP protocol also provide some methods and some principal that are used in authentication [1]. The principal function of EAP protocol is a framework to encapsulate the confidential data (such as username, password etc.) used for authentication purpose. The EAP protocol is not attached to particular EAP methods and in some case a security flow in one method are discovered due to some problem, we can simply change this particular method without changing all the platform or protocol. Recently many EAP methods exist, but few of them are standardized in the Internet Engineering Task Force (IETE) organization. Most of these methods suffer from several problems that make them vulnerable to several types of attacks [1].

In this paper, we will analyze the existing EAP methods, such as password methods (EAP-MD5, EAP-LEAP), certificated methods (EAP-TLS), tunnel and protected methods (EAP- TTLS, EAP-PEAP) and will proposes a new EAP method LEAP-PK (Lightweight Extensible Authentication Protocol-Public Key), which combines the simplicity of deployment and management of password methods and robustness of certificated ones.

Rest of this paper is arranged as follows- Section 2 describe the EAP framework, section 3 describe the possible wireless attacks, section 4 describe different EAP methods, section 5 and 6 introduce and describe the new propose method "LEAP-PK" and section 7 illustrates the validation results of the LEAP-PK by using the tool AVISPA. At last will concluding the remark.

III. EXTENSIBLE AUTHENTICATION PROTOCOL

The IEEE approved the 802.11b (2.4 Gigahertz (GHz) range, 11 Mbps throughout) and 802.11a (5 GHz range, 54 Mbps throughout) extensions in September 1999 [7]. That time the wireless LAN is widely adopted and accelerated in vertical (retail, education, health care, transportation and many more) and horizontal markets. As standardized by the

IEEE, security for 802.11 networks can be simplified into 2 main components that are authentication and encryption. These two components have been proven the security [7].

An alternative WLAN security approach focuses on developing a framework for providing centralized authentication and dynamic key distribution. For such system development a proposal jointly submitted to the IEEE by Cisco System, Microsoft and other organizations introduced an end to end framework using 802.1x and EAP to provide the enhanced functionality [7]. The Extensible Authentication Protocol (EAP) (B. Aboba et al. (2004)) is a protocol designed by the Internet Engineering Task Force (IETF) that permits the use of different types of authentication mechanisms through the so called EAP methods, these are performed between an EAP Peer, EAP Server through EAP authenticator which forwards EAP packets back between EAP peer and EAP server (shown in figure 1).

EAP (RFC 2284) allows wireless client adapters, such type of adapter support authentication types for communicate with different back and user server such as Remote Access Dial In User Service (RADIUS) [7]. Cisco employees designed and implemented new protocol that supports all operating that is Lightweight Extensible Authentication Protocol (LEAP). EAP protocol based on 802.1x authentication framework on Cisco Aironet WLAN products and solutions [7].

The RFC 4017 and RFC 3748 that defined by Internet Engineering Task Force (IETF), listed some mandatory, recommended and optional requirements for EAP methods used In IEEE 802.11 wireless [1].

I. Mutual Authentication
II. Identity Privacy
III. Dictionary Attack Resistance
IV. Replay Attack Resistance
V. Derivation of strong and dynamic session keys
VI. Tested Implementation
VII. Delegation & Fast Reconnect.

In EAP protocol communication 3 principal entities are as follows [1]:

1. EAP peer- It is the client to authenticate.
2. EAP authenticator (Access Point) – It corresponds to the entity that has control of the service (such as access point).
3. EAP server(Authenticator Server)- It is the entity capable of authenticating the EAP client

Following figure 1, show the EAP communication (message exchange) between EAP Peer (Client) and Authencation Server. The procedure starts by EAP client who start packet frame. After that EAP authenticator asking EAP client for his identity through identity message. After that EAP client identifies himself through identity message. An identity message of EAP client, EAP authenticator sends this identity to authenticator server. After that, by using specific authentication method, EAP authenticator server sends a request message to EAP client, and then EAP client reply with a response message for agreement to uses that same specific authentication method and then authentication process stars [1].

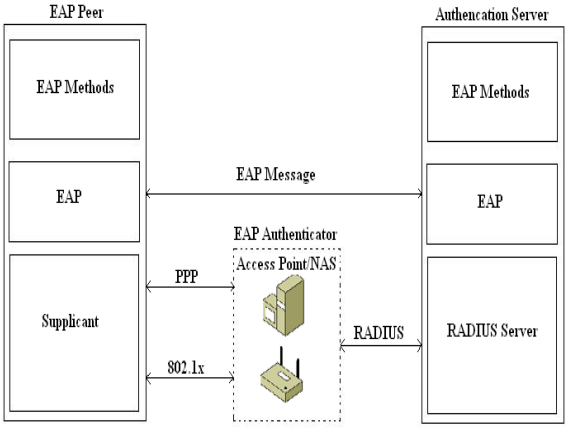


Figure 1. EAP Communication

After end of authentication method procedure, EAP authenticator server sends success message or failure message to EAP client via EAP authenticator according to the status of the authentication process success or fails [1].

IV. EAP METHODS POSSIBLE ATTACKS

The possible attacks on the EAP method as follows:

A. Denial of Service

Spoofing EAP packets in order to collect such information as SSID and the channel of the AP, Denial of Service (DOS) attacks can use and modify the spoofed authentication responses, replay attacks and can also cut the session between client and legal access point or packets with overlapping identifies [1,2].

B. Man-In-The-Middle (MITM)

MITM attack in which an attacker mount a rouge access point between the client and authentication into a trusted network [1, 2].

C. Dictionary Attack

Dictionary attack or using a list of common passwords in order to attempt to gain access by simulating the authentication exchange offline [1, 2].

D. Interfering Encryption

Interfering with negotiation of encryption parameters including the encryption type used in order to negotiate a less secure type which is easier to launch a subsequent attack [1, 2].

V. EAP METHODS

A Wi-Fi Protected Access (WPA) or the 802.11i standard referred to as WPA2, both used for modern wireless networks security. This authentication technique based on the IEEE 802.1 x standards [8]. This standard use of an Extensible Authentication Protocol (EAP) in a point to point network. EAP definition in 802.1 xs does not specify an exact method, algorithm or any procedure for the authentication but rather specifies a framework into which a particular method can be plugged [8].

Some EAP methods specifically developed for wireless networks and some EAP methods are also developed for wired networks. This method based on public key encryption and the use of certificates as well as class of methods that use not certificates but passwords for their authentication methods [8].

*1. **Legacy based methods***

1.1 EAP-MD5

In this methods for user authentication purpose collect username and password, encrypt that via the MD5 message hashing algorithm and pass that data on to a RADIUS server. If the session of user start after authentication, the key of session is not change at that time of session expire. In addition, MD5 encryption cannot fulfill the requirement for symmetric authentication between client and access point and access point and client as specified in the RFC pertaining to EAP over wireless network. This makes it susceptible to man-in-the middle attacks as well. Due to these disadvantages, it should not be considered to be secure EAP method [1, 2, 8].

*2. **Certificate based methods***

1. EAP-TLS

EAP-TLS (Transport Level Security) is an EAP method based on RFC 2716 using public key certificate authentication procedure within the EAP framework. This method provides the mutual authentication between client and authenticator and vice versa. It has been authenticate each entity, including client and access point, possess a public key certificate singed by a mutually trusted certificate authority. Such type of solution is very strong authentication process and is very secure. However, it does require key infrastructure to be in place in order to work. This is necessary to purchase of certificates from an outside central authority or the added deployment of the infrastructure for the enterprise itself to become a certificate authority. For this reason it is more costly to implement than password based methods and distributing the certificates to all the entities on the network has very big issue. For such type of disadvantages it is not mostly used in authentication [1, 2, 8].

2. EAP-TTLS

EAP-TTLS (Tunneled Transport Level Security) is an extension on transport layer security. By using public key algorithm and certificates in this method secure tunnel is established between client and server, issued by a mutually trusted certificate authority. Once this tunnel is established, another authentication method is employed and that transaction is communicated via the secure tunnel. In this method authentication exchange take place by using secured tunnel but it used less secure authentication method (like MD5 or PAP or CHAP) for authentication. This method provide benefit of mutual authentication, secured cipher suite negotiation, the ability to use both passwords and certificates and to keep the user's identify private since any password authentication would occur inside of a certificate secured tunnel [1, 2, 8].

*3. **EAP PEAP***

Protected EAP (PEAP) is an EAP method that is works same as TTLS. It carries an authentication transaction in an encrypted fashion to create a TLS session. Within the encrypted tunnel, it also used less secure authentication method. Unlike in TTLS, PEAP authenticates the authenticator to the client, but not in the other direction. This reduces the complexity and cost by only requiring certificates to the present on the authenticators, not on the clients. But by using PEAP some benefits also include message authentication and encryption, secure key exchange, fragmentation and reassembly ability and fast reconnect. PEAP is one of the most secure EAP methods, but has not gained universal acceptance because Microsoft and Cisco each support differing implementations of the method [1, 2, 8].

*4. **Password based methods***

The cost and ease of use are advantages of this method as compare to certificate based method. The cost is potentially less due to no certificate purchases or self certificate authority setup being necessary for the enterprise and ease of use is enhanced by allowing users to have an easy to remember password rather than a cryptic key. However, unlike certificate based methods, password based methods can be susceptible to dictionary attacks [1, 2, 8].

*5. **SPEKE***

Simple Password Exponential Key Exchange (SPEKE) is an EAP method form Interlink Networks. The SPEKE method uses mutual knowledge of a password in both the authenticator and client to generate a series of messages to be exchanged of apparently random contents. Once both client and authenticator are in agree that the password is correct then a master session key will be shared between the devices for subsequent use. In this method effectively giving a one way function due to the relative difficulty of performing the discrete logarithmic function required to reverse it [1, 2, 8].

This method used public key encryption methods for key transfer and authentication procedures without the expense and complexity of deploying certificates. In addition, the mechanism is not a s sensitive to dictionary attacks as other password based methods.

Requirement\Methods	EAP-MD5	EAP-TLS	EAP-TTLS	EAP-PEAP	EAP-SPEKE
Mandatory Generation of keying material	No	Not required	Yes	Yes	Yes
Mutual authentication	No	Yes	Yes	Yes	Yes
Self protecting	Yes	Yes	Yes	Yes	Yes
Resistance to dictionary attack	Only with long password	Yes	Yes	Yes	Yes
Protection to MITM attack	No	Yes	Yes	Yes	Yes
Protected cipher suite negotiation	No	Not required	Yes	Yes	Yes
Recommended User identity hiding	No	No	Yes	Yes	No
Faster reconnect	No	Yes	Yes	Yes	No

Table 1. EAP Method Requirements for Wireless LAN

EAP Methods	Encryption Technologies	Possible Attacks
EAP-MD5	One way message digest	Dictionary attack, Man-In-The-Middle attack
EAP-TLS	Digital certificates	Strong authentication, resistant to attacks
EAP-TTLS	Digital certificates or Diffie Hellman algorithm to generate keying material, Symmetric key for data encryption	Strong authentication, resistant to attacks
EAP-PEAP	Digital certificates or Diffie Hellman algorithm to generate keying material, Symmetric key for data encryption	Strong authentication, resistant to attacks

Table 2. EAP Methods Comparison of encryption technologies and possible attacks

VI. THE PROPOSE METHODS: LEAP-PK

To provide a mutual authentication and data integrity. This method uses an asymmetric cipher algorithm like RSA, Pohlig-Hellman etc. The propose method based on EAP protocol since it does not require any change in the 802.x/802.11 standards. The working of our propose method as follows.

A. In LEAP-PK, we generate a special random number at client side and id used with shared secret key as input to one way hash function to create new key (kc) at output side, after that send this random number to authenticate server to create same key (kc) as similar at the client.

B. Now the key (kc) is randomly generated that is each time dynamically changes the key (kc) on response pairs exchanged between client and authenticator server. Such type of authentication is very beneficial for mutual authentication process. Suppose the attacker gets the key (kc) at this exchanged between client and authenticator server, present access time, so attacker has nothing to do at next access time because both client and authentication server create a new key (kc) in next access time. Due to changed key (kc) in each access time, the attacker face very difficult to work.

C. In this method, uses key (kc) to encrypt client information (username and password) when exchanged between client and authentication server. The encrypt information is not read by any attacker easily since we maintain the privacy and avoid client tracking by its username. Generate a strong session encryption key by increasing the varying value input {S,C} for one way hash function that used individually at each side of connection to generate the session key by the new varying key (kc).

D. The random number (like 10, 11, 12, 13) which hashed with pre shared static secret key and the content of this key is 128 bits describe as follows.
0 to 31 particular flag used to indicate that random number.
32 to 47 indicate which encryption algorithm (A5/1, A5/2) will be used to encrypt/ decrypt the data traffic.
48 to 128 representation of random number.

E. The random number is generated on client machine, so it is depend on client which encryption algorithm (such as RSA, Pohlig-Hellman, A5/1, A5/2, A5/3 etc.) they used for random number generation and it used any different encryption algorithm that sending this random number to the authentication server. So that, there exist ability to change the encryption algorithm at each access time, since privacy and integrity data traffic is maintain between client and authentication server. This is the responsibility of client used such type of algorithm that required very strong security and encryption/decryption processing.

F. Authenticate server create a session ID for each client for identification of this session, then authentication server sends this session ID record to the client. When session disconnected, client resend session ID record to authentication server, which check existence of session ID record in its database. If exist, it will be reestablish that session with client using same parameters but only both generates a new one dynamic shared secret key (kc). Then they go to direct to generate a new one session encryption key regardless all procedures in between, instead of execute whole protocol again to reduce consumed time and power. The generation of new session encryption key will be done by hashing old one session encryption key with new one dynamic shared secret key (kc) using one way hash function.

VII. LEAP-PK ANALYSIS

The LEAP-PK does not only address all the mandatory features required by the RFC 4017 [2], it also has several advantages in comparison to other EAP methods

A. *Mutual authentication:* LEAP-PK offers to the supplicant the possibility to authenticate the server compared to EAP-MD5. We can now detect more easily the rouge access point and check the message integrity by decrypting the received challenge and comparing the calculated and the received MAC.

B. *Quick authentication:* Unlike the EAP-TLS method, EAP-PK is based on challenge response mechanism with a reduced number of exchanged packets, which offers a quick authentication process.

C. Confidentiality: The confidentiality is guaranteed by using a strong encryption algorithm like RSA, Pohlig-Hellman, A5/1, A5/2, A5/3 etc.

The proposed method is evaluated by using the formal security verification platform AVISPA. In the next section of this paper, we will present the AVISPA tools and discuss the validation results of the LEAP-PK.

VIII. AVISPA DESCRIPTION AND ARCHITECTURE

Network security protocols, such as key-exchange and key-management protocols, are difficult to design and to debug.

The formal verification is logic for proving security properties of network. In the last decade the formal verification of security protocols has been booming and was the subject of intense research. This gave birth to a number of verification tools, like Murphi, CSP, FDR, NRL protocol analyzer, Isabelle and AVISPA. The main goal of this section is to briefly describe the Automated Validation of Internet Security Protocol and Applications tool (AVISPA) [1, 2].

AVISPA takes as input a High Level Protocol Specification Language (HLPSL) for describing security protocols and specifying their intended security properties. HLPSL is an explicit and intuitive language to model a protocol; its semantics is based on Lamport's Temporal Logic of Actions (TLA). The HLPSL is based on roles; each protocol is divided into a set of Basic Roles representing the actions of one single agent in a run of the protocol, and Composition Roles which represent the entire protocol and instantiate the Basic Roles. Each role is modeled as a 'state'. Each state has variables which are responsible for the state transitions, retrieves its initial information by parameters, and communicates synchronously with other roles by channel. The security goal is the most important feature of this tool. It allows the model checkers to find the possible attacks. In general, authentication is modeled by these words: witness, request, wrequest and secret. The figure 2 [2] shows the structure of the AVISPA Tool.

Once the protocol is modeled in HLPSL, AVISPA translates them into a lower-level language Intermediate Format (IF) by a translator called hlpsl2if. IF is executed directly by the back-ends tools (OFMC, CL-AtSe, SATMC and TA4SP) to verify if the security goals are satisfied or violated. The AVISPA tools and HLPSL language are a very popular formal verification pack. However, the differences between the specification language and the notation User and Server, particularly the definitions role by role and not message by message, make this pack difficult to use. For this reason, a new tool "Security Protocol Animator" (SPAN) was created to facilitate the specification phase by allowing the animation of the language HLPSL [1, 2].

SPAN can be used to design and to verify the rightness of the formally modeled protocol; it helps to simulate the designed protocol using HLPSL specifications and to build Message Sequence Charts (MSC) of the protocol. SPAN also allows checking the generation of nonce values and message texts. Since SPAN implements an active intruder, it can also be used to interactively find and build attacks on protocols [1, 2]

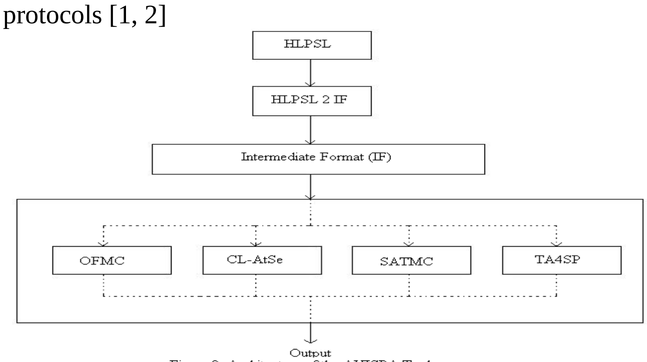


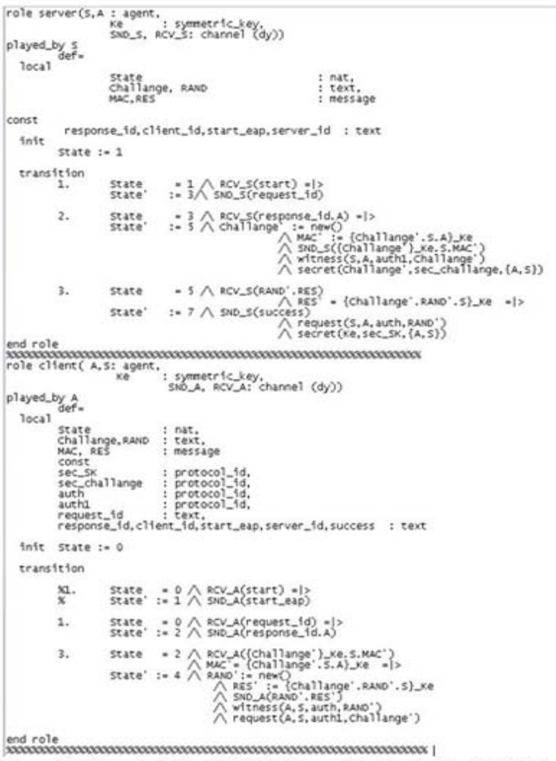
Figure 2. Architecture of the AVISPA Tool

IX. SPECIFICATION AND VALIDATION

This section presents the validation results of the EAPPK, obtained by using the tools AVISPA and SPAN. Since the authenticator only passes through the authentication messages between the peer and authentication sever, the authenticator can be omitted in the formal verification. EAP-PK protocol is defined in User (user station) and Server (authentication server) Model and then is coded in the formal language HLPSL used in AVISPA. The correctness of the written HLPSL code is checked using the protocol animation tool SPAN. Then, the protocol is analyzed by executing the AVISPA tools. The figure 3 [2] presents an extract of the EAP-PK specification in the HLPSL language.

We assume that the user station and the authentication server had a pre-shared pair of key (Ke, Kd) in advance. The server then generates a nonce value (challenge) while the client generates RAND. After protocol verification with SPAN, the intruder simulation was done to check the robustness and whether it makes any abnormal flaws in the protocol run. The HPSL code and the follow goals were verified:

A. *Mutual Authentication:* The supplicant authenticates the server by comparing the calculated MAC value with the received MAC value, which proves that the server knows the pre-shared key Ke [2]. The verification is done by:
   MAC'= {Challange'.server_id.client_id} _Ke
   In the other side the server authenticates the supplicant by checking with the received value RES:
   RES' = {Challange'.RandA'.server_id}_Ke
B. *Key secrecy:* The instruction secret (Ke, sec_SK, {A, S}) asserts that the Ke should be kept secret between the A (client) and the S (server) [2].



```
role server(S,A : agent,
            Ke       : symmetric_key,
            SND_S, RCV_S: channel (dy))
played_by S
     def=
  local
            State                          : nat,
            Challange, RAND                : text,
            MAC,RES                        : message
const
        response_id,client_id,start_eap,server_id  : text
  init
        State := 1
  transition
     1.     State    = 1 /\ RCV_S(start) =|>
            State'  := 3 /\ SND_S(request_id)

     2.     State    = 3 /\ RCV_S(response_id.A) =|>
            State'  := 5 /\ Challange' := new()
                               /\ MAC' := {Challange'.S.A}_Ke
                               /\ SND_S({Challange'}_Ke.S.MAC')
                               /\ witness(S,A,auth1,Challange')
                               /\ secret(Challange',sec_challange,{A,S})

     3.     State    = 5 /\ RCV_S(RAND'.RES)
                               /\ RES' = {Challange'.RAND'.S}_Ke  =|>
            State'  := 7 /\ SND_S(success)
                               /\ request(S,A,auth,RAND')
                               /\ secret(Ke,sec_SK,{A,S})
end role
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
role client( A,S: agent,
             Ke      : symmetric_key,
                       SND_A, RCV_A: channel (dy))
played_by A
     def=
  local
        State             : nat,
        Challange,RAND    : text,
        MAC, RES          : message
        const
        sec_SK            : protocol_id,
        sec_challange     : protocol_id,
        auth              : protocol_id,
        auth1             : protocol_id,
        request_id        : text,
        response_id,client_id,start_eap,server_id,success  : text

  init  State := 0

  transition

     %1.    State   = 0 /\ RCV_A(start) =|>
     %      State' := 1 /\ SND_A(start_eap)

     1.     State   = 0 /\ RCV_A(request_id) =|>
            State' := 2 /\ SND_A(response_id.A)

     3.     State   = 2 /\ RCV_A({Challange'}_Ke.S.MAC')
                        /\ MAC' = {Challange'.S.A}_Ke  =|>
            State' := 4 /\ RAND' := new()
                           /\ RES' := {Challange'.RAND'.S}_Ke
                           /\ SND_A(RAND'.RES')
                           /\ witness(A,S,auth,RAND')
                           /\ request(A,S,auth1,Challange')

end role
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 3. Extract of the LEAP-PK specification in the HLPSL language

C. *Attack robustness:* The witness and request events' goal is to authenticate the source of the message. Witness (S, A, auth1, Challange') signifies "agent S asserts that he wants to be the peer of agent A, agreeing on the challenge value". Request (A, S, auth1, Challange') means "agent A accepts the challenge value and now relies on the guarantee that agent S exists and agrees on this value". This means that the supplicant and the authentication server have the correct and same encryption and decryption keys (Ke, Kd). The supplicant is able to authenticate the server on the challenge value and the server is able to authenticate the supplicant on the RAND value. These roles permit to detect several types of attacks such as: Man in the Middle and dictionary attack [2].

D. *Replay attacks protection:* One time use of challenge and RAND values allows the EAP-PK method to be robust to the replay attack in which the intruder replays old message from a previous protocol run or by specifying multiple parallel sessions between the same agents [2].

The figure 4 [2] presents a simulation part of intruder attacks simulation obtained by SPAN with 2 parallel sessions. And figure 11 shows the result of the OFMC protocol verification. As we can see, no attacks were detected by the OFMC and all the stated security goals were satisfied.

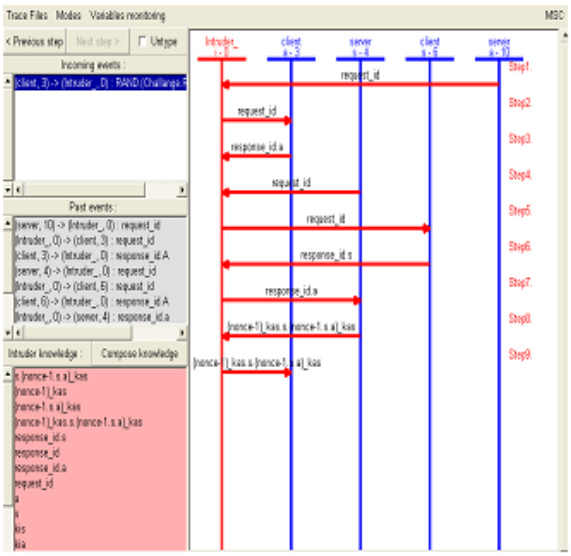


Figure 4. LEAP-PK SPAN attack Simulation

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  C:\progra~1\SPAN\testsuite\results\EAP-XX.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.09s
  visitedNodes: 55 nodes
  depth: 8 plies
```

Figure 5. LEAP-PK OFMC Protocol verification

CONCLUSION

The EAP protocol gives dynamicity and flexibility to the IP networks. However the existed EAP methods do not offer the expected properties foe a secure authentication and easy implementation. In this paper we proposed a new EAP method called LEAP-PK which offers interesting properties of fast and mutual authentication, simplicity of use and robustness to man in the middle, DOS, and offline attacks. The proposed method can be deployed inside wireless networks without using a PKI infrastructure or changing the existed network hardware. To simplify the use of this method, the pre-shared key can be generated from a password shared between the client and the authentication server.

REFERENCES

[1] AHMED M. EL- NAGAR, AHMED A. ABD EL-HAFEZ and ADEL ELHNAWY, *"A Moderate Weight EAP Authentication Method (EAP- MEAP) for Wireless Local Area Network"*, March 2012

[2] Younes El Hajjaji El Idrissi, Noureddine Zahid, Mohamed Jedra,*"A New EAP Authentication Method for IEEE 802.11 Wireless"*, June 2011

[3] R. Dantu, G. Clothier, A. Atri,*"EAP methods for wireless networks"*, Computer Standards Interfaces29

[4] D. Stanley, J. Walker, B. Aboba,*"Extensible Authentication Protocol Method Requirements for Wireless LANs"*, RFC 4017, March 2005.

[5] Kwang-Hyun Baek, Sean W. Smith, David Kotz, *"A Survey of WPA and 802.11i RSN Authentication Protocols"*, November 2004.

[6] Riad Lemhachheche, Jumnit Hong,*"WEP Protocol Weaknesses and Vulnerabilities"*.

[7] "*Extensible Authentication Protocol Transport Layer Security Deployment Guide for Wireless LAN Networks"*, CISCO SYSTEM White Paper.

[8] Ram Dantu, Gabriel Clothier, Anuj Atri, *"EAP Methods for Wireless Networks"*, September 2006.